

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

.REM &

IDENTIFICATION

-----  
PRODUCT CODE: AC-T704B-MC  
PRODUCT NAME: CZKDJ80 KDJ11 CPU DIAGNOSTIC  
PRODUCT DATE: 15-MAR-84  
MAINTAINER: DIAGNOSTIC ENGINEERING  
AUTHORS: HENRY ENMAN, JIM PITTMAN, BARRY IRRGANG

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

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

COPYRIGHT (C) 1983, 1984 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL	PDP	UNIBUS	MA'SBUS
DEC	DECUS	DECTAPE	

&

39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65

HISTORY

.REM 6

-----

OCT-83 REV. A  
FEB-84 REV. B

- FIRST RELEASE
- CORRECTIONS MADE TO:
1. CORRECT VECTOR AREA MAINTENANCE PROBLEM
  2. SET APT SWR TO 2000 SO THAT DEFAULT IS TO NOT TEST BEVENT WHEN IN APT ENVIRONMENT.
  3. PREVENT #TESTN FROM GETTING OUT OF SYNC WHEN SKIPPING DESELECTED TESTS.
  4. PREVENT EXECUTION OF RESET INSTRUCTION TEST WHEN IN APT ENVIRONMENT.
  5. CHANGE MARK INSTRUCTION TEST.
  6. TURN CACHE MEMORY SYSTEM OFF DURING NON-CACHE TESTS.
  7. ENSURE THAT CPU ERROR REGISTER IS CLEARED AFTER COMPLETION OF TEST THAT MIGHT CAUSE IT TO BE SET.
  8. SAVE PC AND CONTENTS OF R6 ON UNEXPECTED INTERRUPTS
- ADDITIONAL TESTS TO IMPROVE TEST COVERAGE INCLUDE:
1. RED ZONE TRAP TEST
  2. I/O TIME OUT TRAP TEST
  3. OOO ADDRESS TRAP TEST
  4. PRE-FETCH BUFFER INVALIDATION TEST
  5. TEST FOR SLOW C BIT ON ROR, ROL AND SXT INSTRUCTIONS
  6. WAIT INSTRUCTION TEST WHEN BEVENT TEST IS SELECTED

6

D1

66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83

.REM &

TABLE OF CONTENTS

1.0	GENERAL INFORMATION
1.1	PROGRAM ABSTRACT
1.2	SYSTEM REQUIREMENTS
1.3	RELATED DOCUMENTS AND STANDARDS
1.4	DIAGNOSTIC HIERARCHY PREREQUISITES
1.5	ASSUMPTIONS
2.0	OPERATING INSTRUCTIONS
3.0	ERROR INFORMATION

&

84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100  
101  
102  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139

.REM 8

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

THIS IS AN APT COMPATIBLE VERSION OF THE KDJ11 CPU DIAGNOSTIC.  
IT FOCUSES ON TESTING THE KDJ11 BASIC INSTRUCTION SET INCLUDING  
EIS, TRAPS AND THE ALTERNATE REGISTER SET.

1.2 SYSTEM REQUIREMENTS

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

1.3 RELATED DOCUMENTS AND STANDARDS

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

1.4 DIAGNOSTIC HIERARCY PREREQUISITES

NONE

1.5 ASSUMPTIONS

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

2.0 OPERATING INSTRUCTIONS

2.1 LOADING AND STARTING PROCEEDURE

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

2.2 PROGRAM OPTIONS

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

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



140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161  
162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192

\*\*\*\*\*

DEFAULT SETTINGS ARE TO TEST BEVENT, THE OTHER BITS HAVE NO EFFECT ON THE OPERATION OF THE CPU TEST.

TO CHANGE THE SWITCH REGISTER, HALT THE PROGRAM, LOAD THE SOFTWARE SWITCH REGISTER (ADDRESS 176) WITH THE DESIRED OPTIONS AND RESTART THE PROGRAM USING THE J11 MICRO OOT P COMMAND.

#### 2.2 OPERATION UNDER APT

OPERATION IN THE APT ENVIRONMENT REQUIRES SOME SPECIAL CONSIDERATIONS DUE TO THE ASYNCHRONOUS HALTS OF THE DIAGNOSTIC BY THE APT MONITOR. IF THE EFFECTS OF THESE HALTS ARE NOT ANTICIPATED, FALSE ERRORS MAY BE REPORTED. THEREFORE, WHEN OPERATING IN THE APT ENVIRONMENT THE FOLLOWING DIFFERENCES IN THE EXECUTION OF THE PROGRAM SHOULD BE NOTED:

1. THE RESET INSTRUCTION TEST IS NOT EXECUTED
2. BIT 10 IN THE SOFTWARE SWITCH REGISTER IS SET SO THAT BEVENT IS NOT TESTED UNDER APT.
3. THE SERIAL LINE UNIT INTERRUPT TEST IS EXECUTED ONLY ON THE FIRST PASS OF THE PROGRAM.
4. RED ZONE TRAP TEST CHECKS FOR APT ENVIRONMENT BEFORE CALLING ERROR ROUTINE. IF IN APT MODE AND AN ERROR OCCURS, IT WILL RETRY TEST ONE MORE TIME. IF IT PASSES ON SECOND ATTEMPT, THEN FIRST ERROR WILL BE CONSIDERED TO BE APT INDUCED.

#### 3.0 ERROR INFORMATION

ALL ERRORS WILL HALT AFTER REPORTING TO APT. ERRORS RELATING TO BOARD TESTS WILL PRINT THE FOLLOWING ERROR MESSAGE:

ERROR WHILE TESTING BOARD FUNCTIONS  
ERROR # - (UNIQUE ERROR NUMBER)  
ERROR PC - (PC AT TIME OF ERROR)

ERRORS RELATED TO CPU TESTS WILL PRINT THE FOLLOWING MESSAGE.

ERROR DURING CPU TESTS  
ERROR # - (UNIQUE ERROR NUMBER)  
ERROR PC - (PC AT TIME OF ERROR)

#### 4.0 PROGRESS REPORT

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

PROGRAM HEADER AND TABLES  
 KDJ11A.MAC 22-FEB-84 15:12

MACY11 30A(1052) 15-MAR-84 13:28 PAGE 6

SEQ 0006

```

193 .TITLE PROGRAM HEADER AND TABLES
194 .SBITL PROGRAM HEADER
195
196 .MCALL NEWTST,ERRDEF,,EQUAT,,KT11,,%OCAT,,%EOP,,%APTBL5,SETUP
197 .MCALL .%TYPE,,%TYPDEC,ERRDF,BGNTST,%ENDTST,BGNMOD,ENMOD,CKLOOP
198 .MCALL .HEADER,,SETUP,,%TRAP,BGNSUB,ENDSUB,,%ACT11,,%APTHOR
199 .MCALL .%APTYPE,,%ERROR,,%TYPOCT,,%FEAD
200
201
202
203 .TITLE KDJ11-A CPU DIAGNOSTIC
204 ;*COPYRIGHT (C) OCTOBER,1983
205 ;*DIGITAL EQUIPMENT CORP.
206 ;*MAYNARD, MASS. 01754
207 ;*
208 ;*
209 ;*THIS PROGRAM WAS ASSEMBLED USING THE PDP-11 MAINDEC SYSMAC
210 ;*PACKAGE (MAINDEC-11-DZQAC-C3), JAN 19, 1977.
211 ;*
212 000001 $TJ=1
213 160000 $SWR=160000 ;;HALT ON ERROR, LOOP ON TEST, INHIBIT ERROR TYP0UT

```

```

214 .TITLE GLOBAL AREAS
215 .SBTTL GLOBAL EQUATES SECTION
216
217 ***
218 ; THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
219 ; ARE USED IN MORE THAN ONE TEST.
220 ;
221 .SBTTL BASIC DEFINITIONS
222
223 ;*INITIAL ADDRESS OF THE STACK POINTER *** 1000 ***
224 001000 STACK= 1000
225 .EQUIV EMT,ERROR ;BASIC DEFINITION OF ERROR CALL
226 .EQUIV IOT,SCOPE ;BASIC DEFINITION OF SCOPE CALL
227
228 ;*MISCELLANEOUS DEFINITIONS
229 000011 HT= 11 ;CODE FOR HORIZONTAL TAB
230 000012 LF= 12 ;CODE FOR LINE FEED
231 000015 CR= 15 ;CODE FOR CARRIAGE RETURN
232 000200 CRLF= 200 ;CODE FOR CARRIAGE RETURN-LINE FEED
233 177776 PS= 177776 ;PROCESSOR STATUS WORD
234 .EQUIV PS,PSW
235 177774 STKLMT= 177774 ;STACK LIMIT REGISTER
236 177772 PIRQ= 177772 ;PROGRAM INTERRUPT REQUEST REGISTER
237 177570 DSWR= 177570 ;HARDWARE SWITCH REGISTER
238 177570 DDISP= 177570 ;HARDWARE DISPLAY REGISTER
239
240 ;*GENERAL PURPOSE REGISTER DEFINITIONS
241 000000 R0= #0 ;GENERAL REGISTER
242 000001 R1= #1 ;GENERAL REGISTER
243 000002 R2= #2 ;GENERAL REGISTER
244 000003 R3= #3 ;GENERAL REGISTER
245 000004 R4= #4 ;GENERAL REGISTER
246 000005 R5= #5 ;GENERAL REGISTER
247 000006 R6= #6 ;GENERAL REGISTER
248 000007 R7= #7 ;GENERAL REGISTER
249 000006 SP= #6 ;STACK POINTER
250 000007 PC= #7 ;PROGRAM COUNTER
251
252 ;*PRIORITY LEVEL DEFINITIONS
253 000000 PR0= 0 ;PRIORITY LEVEL 0
254 000040 PR1= 40 ;PRIORITY LEVEL 1
255 000100 PR2= 100 ;PRIORITY LEVEL 2
256 000140 PR3= 140 ;PRIORITY LEVEL 3
257 000200 PR4= 200 ;PRIORITY LEVEL 4
258 000240 PR5= 240 ;PRIORITY LEVEL 5
259 000300 PR6= 300 ;PRIORITY LEVEL 6
260 000340 PR7= 340 ;PRIORITY LEVEL 7
261
262 ;*"SWITCH REGISTER" SWITCH DEFINITIONS
263 100000 SW15= 100000
264 040000 SW14= 40000
265 020000 SW13= 20000
266 010000 SW12= 10000
267 004000 SW11= 4000
268 002000 SW10= 2000
269 001000 SW09= 1000

```

GLOBAL AREAS MACY11 30A(1052) 15-MAR-84 13:28 PAGE 8  
 KDJ11A.MAC 22-FEB-84 15:12 BASIC DEFINITIONS

SEQ 0008

270	000400	SW08=	400	
271	000200	SW07=	200	
272	000100	SW06=	100	
273	000040	SW05=	40	
274	000020	SW04=	20	
275	000010	SW03=	10	
276	000004	SW02=	4	
277	000002	SW01=	2	
278	000001	SW00=	1	
279		.EQUIV	SW09,SW9	
280		.EQUIV	SW08,SW8	
281		.EQUIV	SW07,SW7	
282		.EQUIV	SW06,SW6	
283		.EQUIV	SW05,SW5	
284		.EQUIV	SW04,SW4	
285		.EQUIV	SW03,SW3	
286		.EQUIV	SW02,SW2	
287		.EQUIV	SW01,SW1	
288		.EQUIV	SW00,SW0	
289				
290		;*DATA BIT DEFINITIONS (BIT00 TO BIT15)		
291	100000	BIT15=	100000	
292	040000	BIT14=	40000	
293	020000	BIT13=	20000	
294	010000	BIT12=	10000	
295	004000	BIT11=	4000	
296	002000	BIT10=	2000	
297	001000	BIT09=	1000	
298	000400	BIT08=	400	
299	000200	BIT07=	200	
300	000100	BIT06=	100	
301	000040	BIT05=	40	
302	000020	BIT04=	20	
303	000010	BIT03=	10	
304	000004	BIT02=	4	
305	000002	BIT01=	2	
306	000001	BIT00=	1	
307		.EQUIV	BIT09,BIT9	
308		.EQUIV	BIT08,BIT8	
309		.EQUIV	BIT07,BIT7	
310		.EQUIV	BIT06,BIT6	
311		.EQUIV	BIT05,BIT5	
312		.EQUIV	BIT04,BIT4	
313		.EQUIV	BIT03,BIT3	
314		.EQUIV	BIT02,BIT2	
315		.EQUIV	BIT01,BIT1	
316		.EQUIV	BIT00,BIT0	
317				
318		;*BASIC "CPU" TRAP VECTOR ADDRESSES		
319	000004	ERRVEC=	4	;; TIME OUT AND OTHER ERRORS
320	000010	RESVEC=	10	;; RESERVED AND ILLEGAL INSTRUCTIONS
321	000014	TBITVEC=	14	;; "T" BIT
322	000014	TRTVEC=	14	;; TRACE TRAP
323	000014	BPTVEC=	14	;; BREAKPOINT TRAP (BPT)
324	000020	IOTVEC=	20	;; INPUT/OUTPUT TRAP (IOT) **SCOPE**
325	000024	PWRVEC=	24	;; POWER FAIL

```

326      000030      EMTVEC= 30      ;EMULATOR TRAP (EMT) **ERROR**
327      000034      TRAPVEC=34      ;"TRAP" TRAP
328      000060      TKVEC= 60      ;TTY KEYBOARD VECTOR
329      000064      TPVEC= 64      ;TTY PRINTER VECTOR
330      000240      PIRQVEC=240    ;PROGRAM INTERRUPT REQUEST VECTOR
331
332      .SBTTL MEMORY MANAGEMENT DEFINITIONS
333
334      ;*KT11 VECTOR ADDRESS
335      000250      MMVEC= 250
336
337      ;*KT11 STATUS REGISTER ADDRESSES
338
339      177572      SR0= 177572
340      177574      SR1= 177574
341      177576      SR2= 177576
342      172516      SR3= 172516
343
344      ;*USER "I" PAGE DESCRIPTOR REGISTERS
345
346      177600      UIPDR0= 177600
347      177602      UIPDR1= 177602
348      177604      UIPDR2= 177604
349      177606      UIPDR3= 177606
350      177610      UIPDR4= 177610
351      177612      UIPDR5= 177612
352      177614      UIPDR6= 177614
353      177616      UIPDR7= 177616
354
355      ;*USER "D" PAGE DESCRIPTOR REGISTERS
356
357      177620      UDPDR0= 177620
358      177622      UDPDR1= 177622
359      177624      UDPDR2= 177624
360      177626      UDPDR3= 177626
361      177630      UDPDR4= 177630
362      177632      UDPDR5= 177632
363      177634      UDPDR6= 177634
364      177636      UDPDR7= 177636
365
366      ;*USER "I" PAGE ADDRESS REGISTERS
367
368      177640      UIPAR0= 177640
369      177642      UIPAR1= 177642
370      177644      UIPAR2= 177644
371      177646      UIPAR3= 177646
372      177650      UIPAR4= 177650
373      177652      UIPAR5= 177652
374      177654      UIPAR6= 177654
375      177656      UIPAR7= 177656
376
377      ;*USER "D" PAGE ADDRESS REGISTERS
378
379      177660      UOPAR0= 177660
380      177662      UOPAR1= 177662
381      177664      UOPAR2= 177664
    
```

382	177666	UDPAR3= 177666
383	177670	UDPAR4= 177670
384	177672	UDPAR5= 177672
385	177674	UDPAR6= 177674
386	177676	UDPAR7= 177676
387		
388		;*SUPERVISOR "I" PAGE DESCRIPTOR REGISTERS
389		
390	172200	SIPDR0= 172200
391	172202	SIPDR1= 172202
392	172204	SIPDR2= 172204
393	172206	SIPDR3= 172206
394	172210	SIPDR4= 172210
395	172212	SIPDR5= 172212
396	172214	SIPDR6= 172214
397	172216	SIPDR7= 172216
398		
399		;*SUPERVISOR "D" PAGE DESCRIPTOR REGISTERS
400		
401	172220	SDPDR0= 172220
402	172222	SDPDR1= 172222
403	172224	SDPDR2= 172224
404	172226	SDPDR3= 172226
405	172230	SDPDR4= 172230
406	172232	SDPDR5= 172232
407	172234	SDPDR6= 172234
408	172236	SDPDR7= 172236
409		
410		;*SUPERVISOR "I" PAGE ADDRESS REGISTERS
411		
412	172240	SIPAR0= 172240
413	172242	SIPAR1= 172242
414	172244	SIPAR2= 172244
415	172246	SIPAR3= 172246
416	172250	SIPAR4= 172250
417	172252	SIPAR5= 172252
418	172254	SIPAR6= 172254
419	172256	SIPAR7= 172256
420		
421		;*SUPERVISOR "D" PAGE ADDRESS REGISTERS
422		
423	172260	SDPAR0= 172260
424	172262	SDPAR1= 172262
425	172264	SDPAR2= 172264
426	172266	SDPAR3= 172266
427	172270	SDPAR4= 172270
428	172272	SDPAR5= 172272
429	172274	SDPAR6= 172274
430	172276	SDPAR7= 172276
431		
432		;*KERNEL "I" PAGE DESCRIPTOR REGISTERS
433		
434	172300	KIPDR0= 172300
435	172302	KIPDR1= 172302
436	172304	KIPDR2= 172304
437	172306	KIPDR3= 172306

GLOBAL AREAS  
KDJ11A.MAC

MACY11 30A(1052)  
22-FEB-84 15:12

15-MAR-84 13:28 PAGE 11

MEMORY MANAGEMENT DEFINITIONS

SEQ 0011

438	172310	KIPDR4=	172310
439	172312	KIPDR5=	172312
440	172314	KIPDR6=	172314
441	172316	KIPDR7=	172316
442			
443		; *KERNEL "D" PAGE DESCRIPTOR REGISTERS	
444			
445	172320	KDPDR0=	172320
446	172322	KDPDR1=	172322
447	172324	KDPDR2=	172324
448	172326	KDPDR3=	172326
449	172330	KDPDR4=	172330
450	172332	KDPDR5=	172332
451	172334	KDPDR6=	172334
452	172336	KDPDR7=	172336
453			
454		; *KERNEL "I" PAGE ADDRESS REGISTERS	
455			
456	172340	KIPAR0=	172340
457	172342	KIPAR1=	172342
458	172344	KIPAR2=	172344
459	172346	KIPAR3=	172346
460	172350	KIPAR4=	172350
461	172352	KIPAR5=	172352
462	172354	KIPAR6=	172354
463	172356	KIPAR7=	172356
464			
465		; *KERNEL "D" PAGE ADDRESS REGISTERS	
466			
467	172360	KDPAF0=	172360
468	172362	KDPAF1=	172362
469	172364	KDPAF2=	172364
470	172366	KDPAF3=	172366
471	172370	KDPAF4=	172370
472	172372	KDPAF5=	172372
473	172374	KDPAF6=	172374
474	172376	KDPAF7=	172376
475			
476		; THESE ARE FLOATING POINT ACCUMULATOR EQUATES	
477	000000	ACO=	#0
478	000001	AC1=	#1
479	000002	AC2=	#2
480	000003	AC3=	#3
481	000004	AC4=	#4
482	000005	AC5=	#5
483	000006	AC6=	#6
484	000007	AC7=	#7
485			
486	000244	FPVEC=	244
487			
488		; THESE ARE CACHE REGISTER EQUATES	
489	177746	CCR=	177746
490	177744	MSE=	177744
491	177752	HITHIS=	177752
492	177766	CPEREG=	177766
493			

; CACHE CONTROL REGISTER  
; MEMORY SYSTEM ERROR REGISTER  
; HIT/MISS REGISTER  
; CPU ERROR REGISTER

```

494                                     ;MISCELLANEOUS DEFINITIONS
495      177546      BEVENT= 177546      ;BEVENT CONTROL REGISTER
496      177560      RCSR= 177560
497      177562      RBUF= 177562
498      177564      XCSR= 177564
499      177566      XBUF= 177566
500      000000      ERRTN= HALT
501      000001      $TSTNC=1
502      000001      ERRNUM= 1      ;INITIALIZE ERROR NUMBER COUNTER
503      002000      AUSWR= 2000      ;SWR FOR APT--NO BEVENT TESTING
504
505
506                                     ;THIS EQUATE DEFINES THE BOTTOM OF THE PROGRAM STACK POINTER
507      001000      STBOT= 1000
508      000000      .ASECT
509      .SBTTL TRAP CATCHER
510
511      000000      . =0
512                                     ;*ALL UNUSED LOCATIONS OF THE VECTOR AREA CONTAIN
513                                     ;*A ".+2, IOT" SEQUENCE TO CATCH AND PROCESS ILLEGAL
514                                     ;*TRAPS AND INTERRUPTS THAT MIGHT OCCUR.
515                                     ;*THE IOT TRAP WHICH IS TAKEN ON THE ILLEGAL TRAP/INT
516                                     ;*TRAPS TO THE $SCOPE ROUTINE WHICH (IF THE RETURN PC IS
517                                     ;*LESS THAN 1002) JUMPS TO THE $ERROR ROUTINE.
518                                     ;*THE $ERROR ROUTINE WILL REPORT THE ERROR AS FOLLOWS:
519                                     ;*   PC=YYYYYY UNEXPECTED TRAP TO XXX
520                                     ;*AND RETURN TO THE PROGRAM AT PC=YYYYYY+2
521                                     ;*WHERE XXX=LOCATION OF ILLEGAL TRAP
522                                     ;*   YYYYYY=PC AT TIME OF TRAP
523                                     ;*NOTE: IF THE PROCESSOR IS NOT AN 11/05 THE PROGRAM
524                                     ;*   CAN BE STARTED AT ADDRESS 0 AS WELL AS ADDRESS 200.
525
526      000000      000000      $40CAT: HALT      ;;HALT
527      000002      000737      BR      .-100      ;;BRANCH TO 177700 & TIME OUT (NOT ON
528                                     ;;11/05)
529      000004      001266      .WORD START      ;;VECTOR TO STARTING ADDRESS
530      000006      000340      .WORD 340      ;;WITH PRIORITY LEVEL 7
531      000174      000174      .-174
532      000174      000000      DISPREG: .WORD 0      ;;SOFTWARE DISPLAY REGISTER
533      000176      000000      SWREG: .WORD 0      ;;SOFTWARE SWITCH REGISTER
534      .SBTTL STARTING ADDRESS)
535      000200      000137      001266      JMP $START ;;GO TO START OF PROGRAM
536      .SBTTL ACT11 HOOKS
537
538                                     ;*****
539                                     ;HOOKS REQUIRED BY ACT11
540      000204      $SVPC=.      ;SAVE PC
541      000046      . =46
542      000046      042016      $ENDAD      ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN . $EUP
543      000052      000052      . =52
544      000052      000000      .WORD 0      ;;2)SET LOC.52 TO ZERO
545      000204      .-$SVPC      ;;RESTORE PC
546      .SBTTL APT PARAMETER BLOCK
547
548                                     ;*****
549      ;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT

```



```

550      ;*****
551      . $X= .      ;SAVE CURRENT LOCATION
552      . =24      ;SET POWER FAIL TO POINT TO START OF PROGRAM
553      000024      000200      200      ;FOR APT START UP
554      . =44      ;POINT TO APT INDIRECT ADDRESS PNTR.
555      000044      000204      $APTHOR ;POINT TO APT HEADER BLOCK
556      . = $X      ;RESET LOCATION COUNTER
557      ;*****
558      ;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
559      ;INTERFACE SPEC.
560
561      000204      $APTHD:
562      000204      000000      $HIBTS: .WORD 0      ;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
563      000206      001000      $MBADR: .WORD $MAIL ;ADDRESS OF APT MAILBOX (BITS 0-15)
564      000210      000001      $TSTM: .WORD 1      ;RUN TIM OF LONGEST TEST
565      000212      000002      $PASTM: .WORD 2      ;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
566      000214      000000      $UNITM: .WORD 0      ;ADDITIONAL RUN TIME (SECS) OF 1 PASS FOR EACH ADDITIONAL UNIT
567      000216      000014      .WORD $ETEND-$MAIL/2 ;LENGTH MAILBOX-ETABLE(WORDS)
568      . = $X      ;SAVE CURRENT LOCATION COUNT
569      . =2
570      000002      000000      0
571      000004      000006      6
572      000006      000004      4      ;SET UP SOME VECTORS
573      . = $X      ;RESTORE LOCATION COUNT
574      . =1000

```

```

575 .SBTTL GLOBAL DATA SECTION
576
577
578 ; THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
579 ; IN MORE THAN ONE TEST.
580
581 .SBTTL APT MAILBOX-ETABLE
582
583 ;|*****
584 .EVEN
585 001000 $MAIL: ; APT MAILBOX
586 001000 000000 $MSGTY: .WORD AMSGTY ; MESSAGE TYPE CODE
587 001002 000000 $FATAL: .WORD AFATAL ; FATAL ERROR NUMBER
588 001004 000000 $TESTN: .WORD ATESTN ; TEST NUMBER
589 001006 000000 $PASS: .WORD APASS ; PASS COUNT
590 001010 000000 $DEVCT: .WORD ADEVCT ; DEVICE COUNT
591 001012 000000 $UNIT: .WORD AUNIT ; I/O UNIT NUMBER
592 001014 000000 $MSGAD: .WORD AMSGAD ; MESSAGE ADDRESS
593 001016 000000 $MSGLG: .WORD ANSGLG ; MESSAGE LENGTH
594 001020 $ETABLE: ; APT ENVIRONMENT TABLE
595 001020 000 $ENV: .BYTE .ENV ; ENVIRONMENT BYTE
596 001021 000 $ENVM: .BYTE AENVM ; ENVIRONMENT MODE BITS
597 001022 000000 $SWREG: .WORD ASWREG ; APT SWITCH REGISTER
598 001024 002000 $USMR: .WORD AUSMR ; USER SWITCHES
599 001026 000000 $CPUOP: .WORD ACPJOP ; CPU TYPE, OPTIONS
600 ;*
601 ;*
602 ;* 11/04=01,11/05=02,11/20=03,11/40=04,11/45=05
603 ;* 11/70=06,POQ=07,Q=10
604 ;* BIT 10-REAL TIME CLOCK
605 ;* BIT 9-FLOATING POINT PROCESSOR
606 ;* BIT 8-MEMORY MANAGEMENT
606 001030 $ETEND:
607 .MEXIT
608
609 ; THESE LOCATIONS ARE USED IN MORE THAN ONE TEST TO STORE VECTOR DATA
610 ; WHEN THE TEST NEEDS TO HAVE AN ERROR CONDITION RESPOND DIFFERENTLY
611 ; FROM THE DEFAULT RESPONSE.
612 001030 000000 SLOC00: .WORD 0
613 001032 000000 SLOC01: .WORD 0
614
615 ; THESE LOCATIONS ARE USED IN MORE THAN ONE TEST TO STORE WORKING DATA.
616 001034 000000 EXPDAT: .WORD 0 ; STORES EXPECTED (GOOD) DATA FOR COMPARISONS
617 001036 000000 RECDAT: .WORD 0 ; STORES RECIEVED DATA TO BE VERIFIED
618 001040 000000 COUNT: .WORD 0 ; ERROR INDICATOR FOR FLOATING POINT TESTS
619 001042 000000 FLAG: .WORD 0 ; USED TO STORE "FLAG" CONDITIONS
620 001044 000000 ERRCNT: .WORD 0 ; STORAGE FOR ERROR COUNT
621 001046 177570 SWR: .WORD DSWR ; STORAGE FOR SWITCH REGISTER ADDRESS
622 001050 177570 DISPLAY: .WORD DDISP ; STORAGE FOR DISPLAY REGISTER ADDRESS
623 001052 000000 $ERFLG: .WORD 0 ; ERROR FLAG
624
625 ; THESE LOCATIONS ARE USED BY MORE THAN ONE TEST AS LOOP COUNTERS
626 001054 000000 DCOUNT: .WORD 0
627 001056 000000 ALLCTR: .WORD 0
628 001060 000000 LOOPIN: .WORD 0
629 001062 000000 SAVSP1: .WORD 0 ; STORAGE FOR UNEXPECTED TRAP DATA
630 001064 000000 SAVSP2: .WORD 0 ; " " " " " "

```

C2

```
631  
632  
633  
634  
635 001066 000000      SEQ:      .WORD  0      ;STORES SEQUENCE NUMBER FOR JUMP TESTS  
636 001070 000000      SPS:      .WORD  0      ;STORES STACK POINTER FOR JUMP TESTS  
637 001072 000000      SPSJ:     .WORD  0      ;STORES STACK POINTER FOR JUMP TESTS  
638  
639  
640  
641 001074 177777      WAITIN:   .WORD 177777  
642  
643 ;!!!!!!THIS IS IT. THE PROGRAM TEST LOCATION AND WRITE BUFFER!!!!!!!!!!!!!!!!!!!!  
644 001076  
645 001076 000002      TSTLOC:  .BLKW  2
```

```

646 .SBTTL GLOBAL TEXT SECTION
647
648 ;**
649 ; THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
650 ; MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
651 ; MORE THAN ONE TEST.
652 ;**
653
654 ;
655 ; FORMAT STATEMENTS USED IN PRINT CALLS
656 ;
657
658 001102 005015 040503 044103 ERRMSG: .ASCIZ <CR><LF>/CACHE SYSTEM ERROR/
659 001110 020105 054523 052123
660 001116 046505 042440 051122
661 001124 051117 000
662 001127 015 042412 051122 CPUERR: .ASCIZ <CR><LF>/ERROR DURING CPU TESTS/
663 001134 051117 042040 051125
664 001142 047111 021107 050103
665 001150 020125 042524 052123
666 001156 000123
667 001160 005015 051105 047522 BRDERR: .ASCIZ <CR><LF>/ERROR WHILE TESTING BOARD FUNCTIONS/
668 001166 020122 044127 046111
669 001174 020105 042524 052123
670 001202 047111 020107 047502
671 001210 051101 020104 052506
672 001216 041516 044524 047117
673 001224 000123
674 001226 005015 051105 047522 ERR1: .ASCIZ <CR><LF>/ERROR # =/
675 001234 020122 020043 000075
676 001242 005015 051105 047522 ERR2: .ASCIZ <CR><LF>/ERROR PC =/
677 001250 020122 041520 036440
678 001256 000
679 001257 015 020012 020040 #CRLF: .ASCIZ <CR><LF>/ /
680 001264 000
681 001266 .EVEN

```

E2

GLOBAL AREAS      MACY11 30A(1052) 15-MAR-84 13:28 PAGE 17  
KDJ11A.MAC      22-FEB-84 15:12      GLOBAL ERROR REPORT SECTION

SEQ 0017

682  
683  
684  
685  
686  
687  
688

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

F2

GLOBAL AREAS      MACY11 30A(1052) 15-MAR-84 13:28 PAGE 18  
KDJ11A.MAC      22-FEB-84 15:12      GLOBAL SUBROUTINES SECTION

SEQ 0018

689  
690  
691  
692  
693  
694

.SBTTL GLOBAL SUBROUTINES SECTION  
; ++  
; THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES  
; THAT ARE USED IN MORE THAN ONE TEST  
; --

```

695 001266          START:
696 001266 012737 000014 177746      MOV     #14,0#CCR          ;SET CACHE TO FORCE MISS
697                                     .SBTTL INITIALIZE THE COMMON TAGS
698 001274 012706 001000          MOV     #STACK,SP        ;;SETUP THE STACK POINTER
699                                     ;;INITIALIZE A FEW VECTORS
700 001300 012737 043470 000030      MOV     #ERROR,0#EMTVEC  ;;EMT VECTOR FOR ERROR ROUTINE
701 001306 012737 000340 000032      MOV     #340,0#EMTVEC+2 ;;LEVEL 7
702 001314 012737 043152 000034      MOV     #TRAP,0#TRAPVEC ;;TRAP VECTOR FOR TRAP CALLS
703 001322 012737 000340 000036      MOV     #340,0#TRAPVEC+2;LEVEL 7
704 001330 005067 177452          CLR     #PASS            ;;CLEAR THE PASS COUNT
705 001334 016767 040424 040414      MOV     #ENDCT,#EOPCT   ;;SETUP END-OF-PROGRAM COUNTER
706 001342 105067 177504          CLR     #ERFLG          ;;CLEAR THE ERROR FLAG
707                                     ;;SIZE FOR A HARDWARE SWITCH REGISTER. IF NOT FOUND OR IT IS
708                                     ;;EQUAL TO A "-1", SETUP FOR A SOFTWARE SWITCH REGISTER.
709 001346 013746 000004          MOV     #ERRVEC,-(SP)   ;;SAVE ERROR VECTOR
710 001352 012737 001406 000004      MOV     #64#,R#ERRVEC   ;;SET UP ERROR VECTOR
711 001360 012767 177570 177460      MOV     #DSWR,SWR       ;;SETUP FOR A HARDWARE SWICH REGISTER
712 001366 012767 177570 177454      MOV     #DDISP,DISPLAY  ;;AND A HARDWARE DISPLAY REGISTER
713 001374 022777 177777 177444      CMP     #-1,DSWR        ;;TRY TO REFERENCE HARDWARE SWR
714 001402 001012          BNE     66#             ;;BRANCH IF NO TIMEOUT TRAP OCCURRED
715                                     ;;AND THE HARDWARE SWR IS NOT = -1
716 001404 000403          BR      65#             ;;BRANCH IF NO TIMEOUT
717 001406 012716 001414          64#:  MOV     #65#,(SP)    ;;SET UP FOR TRAP RETURN
718 001412 000002          RTI
719 001414 012767 000176 177424      65#:  MOV     #SWREG,SWR   ;;POINT TO SOFTWARE SWR
720 001422 012767 000174 177420      MOV     #DISPREG,DISPLAY
721 001430 012637 000004          66#:  MOV     (SP)+,#ERRVEC ;;RESTORE ERROR VECTOR
722
723                                     .MACRO  ##SETMAIL      ?#ARG1
724                                     CLR     #PASS            ;;CLEAR PASS COUNT
725                                     BITB    #APTSIZE,#ENVM   ;;TEST USER SIZE UNDER APT
726                                     BEQ     #ARG1            ;;YES,USE NON-APT SWITCH
727                                     MOV     #SWREG,SWR      ;;NO,USE APT SWITCH REGISTER
728                                     #ARG1:
729                                     .ENDM   ##SETMAIL
730 001434 005067 177346          CLR     #PASS            ;;CLEAR PASS COUNT
731 001440 132767 000200 177353      BITB    #APTSIZE,#ENVM   ;;TEST USER SIZE UNDER APT
732 001446 001403          BEQ     67#             ;;YES,USE NON-APT SWITCH
733 001450 012767 001022 177370      MOV     #SWREG,SWR      ;;NO,USE APT SWITCH REGISTER
734 001456
735 001456 012737 043470 000020      67#:  MOV     #ERROR,0#IOTVEC ;;SET UP IOT VECTORS
736 001464 012737 000340 000022      MOV     #340,0#IOTVEC+2 ;;TO GO TO ERROR ROUTINE
737 001472 005037 177366          CLR     #177766         ;;CLEAR CPU ERROR REGISTER
738 001476 005067 177302          .RESTART: CLR #TESTN    ;;RESET #TESTN TO ZERO
739 001502 012737 000014 177746      MOV     #14,0#CCR       ;;SET CACHE TO FORCE MISS
740
741                                     .SBTTL BASE INSTRUCTION SET TESTS
742                                     ;*****
743                                     ;*****
744                                     ; BEGIN BASE INSTRUCTION SET TESTING
745                                     ;*****
746                                     ;*****
747 001510          FRSTST:
748                                     ;*****
749                                     ;*TEST 1      TEST BEQ BNE INSTRUCTIONS
750                                     ;*****

```

```

751      ;THESE TWO INSTRUCTIONS ARE FUNDAMENTAL TO RECOGNIZING ERROR CONDITIONS
752      ;*****
753      TST1:
754      001510      005267      177270      INC      $TESTN      ;INCREMENT TEST NUMBER
755      001514      000277      SCC
756      001516      000244      CLZ      ;CC=0100 - Z BIT CLEARED
757      001520      001401      BEQ      1$      ;*TEST INSTR - TRY TO CAUSE A BEQ ERROR
758      001522      001003      BNE      2$      ;BRANCH IF GOOD
759      ;THE Z FLAG DIDNT CLEAR OR BRANCH FAILED.
760      ;FAILURE AT THIS LOCATION
761      ;COULD MEAN A BUS PROBLEM, MICRO-CODE PROBLEM
762      ;CONDITION CODE PROBLEM OR JUST ABOUT ANYTHING
763      ;ELSE.
764      001524      104000      1$:      ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
765      001524      000001      .WORD      1      ;UNIQUE ERROR NUMBER
766      001526      001127      .WORD      CPUERR      ;ADDRESS OF ERROR MESSAGE
767      001530      000257      ;
768      001532      000257      CCC
769      001534      000264      SEZ      ;COND CODES = 0100 (ZERO)
770      001536      001001      BNE      3$      ;*TEST INSTR* TRY TO BRANCH ON ZERO FLAG
771      001540      001403      BEQ      4$      ;*TEST INSTR* BRANCH IF GOOD
772      ;BRANCH FAILURE WITH Z BIT SET
773      001542      104000      3$:      ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
774      001542      000002      .WORD      2      ;UNIQUE ERROR NUMBER
775      001544      001127      .WORD      CPUERR      ;ADDRESS OF ERROR MESSAGE
776      001546      001550      4$:
777      001550      M2:
778      ;*****
779      ;*TEST 2      TEST BRANCH ON CARRY
780      ;*****
781      ;THIS IS A TEST TO SEE IF THE MODULE FORM ANTICIPATED IS FEASIBLE.
782      ;*****
783      TST2:
784      001550      005267      177230      INC      $TESTN      ;INCREMENT TEST NUMBER
785      001550      000257      CCC      ;CC=0000
786      001554      103003      BCC      2$      ;*TEST INSTR*
787      ;BRANCH CARRY CLEAR FAILED
788      001560      104000      1$:      ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
789      001560      000003      .WORD      3      ;UNIQUE ERROR NUMBER
790      001562      001127      .WORD      CPUERR      ;ADDRESS OF ERROR MESSAGE
791      001564      000261      2$:      SEC      ;CC=1111
792      001566      103403      BCS      4$      ;*TEST INSTR*
793      ;BRANCH CARRY SET FAILED
794      001570      104000      3$:      ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
795      001572      000004      .WORD      4      ;UNIQUE ERROR NUMBER
796      001574      001127      .WORD      CPUERR      ;ADDRESS OF ERROR MESSAGE
797      001576      001600      4$:
798      001600      M3:
799      ;*****
800      ;*TEST 3      TEST DATA PATHS
801      ;*****
802
803
804
805
806

```



```

807 001600          TST3:
808 001600 005267 177200      INC      $TESTN      ; INCREMENT TEST NUMBER
809 001604 005000          CLR      R0          ; TRY TO INSURE WE ARE TESTING
810 001606 001005          BNE      1$          ; THE DATA PATH AND NOT THE "CLR R0" INSTRUCTION
811                                     ; FORCE LOCATION TO ZERO
812 001610 005010          CLR      (R0)        ; TRY TO INSURE 0=0
813 001612 001003          BNE      1$          ; AGAIN, TRY TO INSURE THAT 0=0
814 001614 005737 000000    TST      @0          ; BRANCH IF GOOD
815 001620 001403          BEQ      2$          ; LOCATION 0 NOT SETUP PROPERLY
816
817 001622          1$:
818 001622 104000          ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
819 001624 000005          .WORD      5          ; UNIQUE ERROR NUMBER
820 001626 001127          .WORD      CPUERR       ; ADDRESS OF ERROR MESSAGE
821 001630          2$:
822
823 001630          M4:
824                                     ; *****
825                                     ; *TEST 4      TEST DATA PATHS - ONES AND ZEROS
826                                     ; *****
827 001630          TST4:
828 001630 005267 177150    INC      $TESTN      ; INCREMENT TEST NUMBER
829 001634 012737 125252 000000  MOV      @125252,@0  ; 0=125252
830 001642 022737 125252 000000  CMP      @125252,@0  ; SEE IF DATA MADE IT
831 001650 001403          BEQ      2$          ; BRANCH IF IF DATA IS GOOD
832                                     ; ERROR! EITHER THE BUS IS BAD,
833                                     ; OR THE MOV OR COMPARE
834                                     ; INSTRUCTIONS FAILED
835
836 001652          1$:
837 001652 104000          ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
838 001654 000006          .WORD      6          ; UNIQUE ERROR NUMBER
839 001660 001127          .WORD      CPUERR       ; ADDRESS OF ERROR MESSAGE
840                                     ; END OF TEST
841
842 001660          ;
843          M5:
844                                     ; *****
845                                     ; *TEST 5      TEST DATA PATHS - DATA 0'S AND 1'S
846                                     ; *****
847 001660 005267 177120    TST5:
848 001664 012737 052525 000000  INC      $TESTN      ; INCREMENT TEST NUMBER
849 001672 023727 000000 052525  MOV      @052525,@0  ; SETUP DATA
850 001700 001403          CMP      @0,@052525  ; TEST FOR CORRECT DATA
851                                     ;
852 001702          1$:
853 001704 000007          ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
854 001706 001127          .WORD      7          ; UNIQUE ERROR NUMBER
855 001710          .WORD      CPUERR       ; ADDRESS OF ERROR MESSAGE
856
857
858 001710          ;
859          M6:
860                                     ; *****
861                                     ; *TEST 6      TEST DATA PATHS - 1'S
862 001710          TST6:

```

J2

```

863 001710 005267 177070      INC      $TESTN      ;INCREMENT TEST NUMBER
864 001714 005037 000000      CLR      0#0
865 001720 005137 000000      COM      0#0
866 001724 023727 000000 177777  CMP      0#0,0177777 ;SET UP MEMORY LOCATION 0 = 111111
867 001732 001403              BEQ      2#          ;TEST DATA
868 001734              1#:              ;BRANCH IF NO ERROR
869 001734 104000              ERROR
870 001736 000010              .WORD   10          ;ALL ERRORS TO TRAP TO EMT VECTOR
871 001740 001127              .WORD   CPUERR     ;UNIQUE ERROR NUMBER
872 001742              2#:              ;ADDRESS OF ERROR MESSAGE
873
874
875 001742              ;
876              ;GPROTS:
877              ;*****
878              ;*TEST 7      R0 BIT TESTS
879              ;*****
880 001742 005267 177036      TST7:      INC      $TESTN      ;INCREMENT TEST NUMBER
881 001746 012700 177777      MOV      #177777,R0 ;R0=177777
882 001752 020027 177777      CMP      R0,#177777 ;DOES R0=177777
883 001756 001403              BEQ      1#          ;YES GO ON
884              ;NO GO TO ERROR
885 001760 104000              ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
886 001762 000011              .WORD   11          ;UNIQUE ERROR NUMBER
887 001764 001127              .WORD   CPUERR     ;ADDRESS OF ERROR MESSAGE
888 001766 005000              1#:      CLR      R0          ;R0=0
889 001770 020027 000000      CMP      R0,#0      ;DOES R0=0
890 001774 001403              BEQ      2#          ;YES GO ON
891              ;NO GO TO ERROR
892 001776 104000              ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
893 002000 000012              .WORD   12          ;UNIQUE ERROR NUMBER
894 002002 001127              .WORD   CPUERR     ;ADDRESS OF ERROR MESSAGE
895 002004 012700 125252      2#:      MOV      #125252,R0 ;R0=125252
896 002010 020027 125252      CMP      R0,#125252 ;DOES R0=125252
897 002014 001403              BEQ      3#          ;YES GO ON
898              ;NO GO TO ERROR
899 002016 104000              ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
900 002020 000013              .WORD   13          ;UNIQUE ERROR NUMBER
901 002022 001127              .WORD   CPUERR     ;ADDRESS OF ERROR MESSAGE
902 002024 012700 052525      3#:      MOV      #52525,R0  ;R0=52525
903 002030 020027 052525      CMP      R0,#52525  ;DOES R0=52525
904 002034 001403              BEQ      4#          ;YES GO ON
905              ;NO GO TO ERROR
906 002036 104000              ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
907 002040 000014              .WORD   14          ;UNIQUE ERROR NUMBER
908 002042 001127              .WORD   CPUERR     ;ADDRESS OF ERROR MESSAGE
909 002044              4#:
910
911              ;
912 002044              ;GPR1TS:
913              ;*****
914              ;*TEST 10     R1 BIT TESTS
915              ;*****
916 002044              TST10:
917 002044 005267 176734      INC      $TESTN      ;INCREMENT TEST NUMBER
918 002050 012701 177777      MOV      #177777,R1 ;R1=177777

```

```

919 002054 020127 177777      CMP      R1,#177777      ;DOES R1=177777
920 002060 001403              BEQ      1$              ;YES GO ON
921                          ;NO GO TO ERROR
922 002062 104000              ERROR     ;ALL ERRORS TO TRAP TO EMT VECTOR
923 002064 000015              .WORD    15              ;UNIQUE ERROR NUMBER
924 002066 001127              .WORD    CPUERR          ;ADDRESS OF ERROR MESSAGE
925 002070 005001              1$:     CLR      R1              ;R1=0
926 002072 020127 000000      CMP      R1,#0           ;DOES R1=0
927 002076 001403              BEQ      2$              ;YES GO ON
928                          ;NO GO TO ERROR
929 002100 104000              ERROR     ;ALL ERRORS TO TRAP TO EMT VECTOR
930 002102 000016              .WORD    16              ;UNIQUE ERROR NUMBER
931 002104 001127              .WORD    CPUERR          ;ADDRESS OF ERROR MESSAGE
932 002106 012701 125252      2$:     MOV      #125252,R1    ;R1=125252
933 002112 020127 125252      CMP      R1,#125252     ;DOES R1=125252
934 002116 001403              BEQ      3$              ;YES GO ON
935                          ;NO GO TO ERROR
936 002120 104000              ERROR     ;ALL ERRORS TO TRAP TO EMT VECTOR
937 002122 000017              .WORD    17              ;UNIQUE ERROR NUMBER
938 002124 001127              .WORD    CPUERR          ;ADDRESS OF ERROR MESSAGE
939 002126 012701 052525      3$:     MOV      #52525,R1    ;R1=52525
940 002132 020127 052525      CMP      R1,#52525     ;DOES R1=52525
941 002136 001403              BEQ      4$              ;YES GO ON
942                          ;NO GO TO ERROR
943 002140 104000              ERROR     ;ALL ERRORS TO TRAP TO EMT VECTOR
944 002142 000020              .WORD    20              ;UNIQUE ERROR NUMBER
945 002144 001127              .WORD    CPUERR          ;ADDRESS OF ERROR MESSAGE
946 002146              4$:
947
948
949 002146              ;
950              GPR2TS:
951              ;*****
952              ;*TEST 11      R2 BIT TESTS
953              ;*****
954 002146 005267 176632      ;ST11:
955 002152 012702 177777      INC      $TESTN          ;INCREMENT TEST NUMBER
956 002156 020227 177777      MOV      #177777,R2     ;R2=177777
957 002162 001403              CMP      R2,#177777     ;DOES R2=177777
958                          BEQ      1$              ;YES GO ON
959                          ;NO GO TO ERROR
960 002164 104000              ERROR     ;ALL ERRORS TO TRAP TO EMT VECTOR
961 002166 000021              .WORD    21              ;UNIQUE ERROR NUMBER
962 002170 001127              .WORD    CPUERR          ;ADDRESS OF ERROR MESSAGE
963 002172 005002              1$:     CLR      R2              ;R2=0
964 002174 020227 000000      CMP      R2,#0           ;DOES R2=0
965 002200 001403              BEQ      2$              ;YES GO ON
966                          ;NO GO TO ERROR
967 002202 104000              ERROR     ;ALL ERRORS TO TRAP TO EMT VECTOR
968 002204 000022              .WORD    22              ;UNIQUE ERROR NUMBER
969 002206 001127              .WORD    CPUERR          ;ADDRESS OF ERROR MESSAGE
970 002210 012702 125252      2$:     MOV      #125252,R2    ;R2=125252
971 002214 020227 125252      CMP      R2,#125252     ;DOES R2=125252
972 002220 001403              BEQ      3$              ;YES GO ON
973                          ;NO GO TO ERROR
974 002222 104000              ERROR     ;ALL ERRORS TO TRAP TO EMT VECTOR
975 002224 000023              .WORD    23              ;UNIQUE ERROR NUMBER

```

```

975 002226 001127          .WORD  CPUERR          ;ADDRESS OF ERROR MESSAGE
976 002230 012702 052525 3$:  MOV    #52525,R2      ;R2=52525
977 002234 020227 052525    CMP    R2,#52525     ;DOES R2=52525
978 002240 001403          BEQ    4$            ;YES GO ON
979                                ;NO GO TO ERROR
980 002242 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
981 002244 000024          .WORD  24          ;UNIQUE ERROR NUMBER
982 002246 001127          .WORD  CPUERR      ;ADDRESS OF ERROR MESSAGE
983 002250          4$:
984
985
986 002250          ;
987          GPR3TS:
988          ;*****
989          ;*TEST 12      R3 BIT TESTS
990          ;*****
991 002250 005267 176530  TST12:  INC    $TESTN      ;INCREMENT TEST NUMBER
992 002254 012703 177777    MOV    #177777,R3   ;R3=177777
993 002260 020327 177777    CMP    R3,#177777  ;DOES R3=177777
994 002264 001403          BEQ    1$            ;YES GO ON
995                                ;NO GO TO ERROR
996 002266 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
997 002270 000025          .WORD  25          ;UNIQUE ERROR NUMBER
998 002272 001127          .WORD  CPUERR      ;ADDRESS OF ERROR MESSAGE
999 002274 005003          1$:  CLR    R3          ;R3=0
1000 002276 020327 000000    CMP    R3,#0       ;DOES R3=0
1001 002302 001403          BEQ    2$            ;YES GO ON
1002                                ;NO GO TO ERROR
1003 002304 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
1004 002306 000026          .WORD  26          ;UNIQUE ERROR NUMBER
1005 002310 001127          .WORD  CPUERR      ;ADDRESS OF ERROR MESSAGE
1006 002312 012703 125252 2$:  MOV    #125252,R3   ;R3=125252
1007 002316 020327 125252    CMP    R3,#125252 ;DOES R3=125252
1008 002322 001403          BEQ    3$            ;YES GO ON
1009                                ;NO GO TO ERROR
1010 002324 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
1011 002326 000027          .WORD  27          ;UNIQUE ERROR NUMBER
1012 002330 001127          .WORD  CPUERR      ;ADDRESS OF ERROR MESSAGE
1013 002332 012703 052525 3$:  MOV    #52525,R3   ;R3=52525
1014 002336 020327 052525    CMP    R3,#52525  ;DOES R3=52525
1015 002342 001403          BEQ    4$            ;YES GO ON
1016                                ;NO GO TO ERROR
1017 002344 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
1018 002346 000030          .WORD  30          ;UNIQUE ERROR NUMBER
1019 002350 001127          .WORD  CPUERR      ;ADDRESS OF ERROR MESSAGE
1020 002352          4$:
1021
1022
1023 002352          ;
1024          GPR4TS:
1025          ;*****
1026          ;*TEST 13      R4 BIT TESTS
1027          ;*****
1028 002352 005267 176426  TST13:  INC    $TESTN      ;INCREMENT TEST NUMBER
1029 002356 012704 177777    MOV    #177777,R4   ;R4=177777
1030 002362 020427 177777    CMP    R4,#177777  ;DOES R4=177777

```

```

1031 002366 001403      BEQ      1#      ;YES GO ON
1032                                     ;NO GO TO ERROR
1033 002370 104000      ERROR                                     ;ALL ERRORS TO TRAP TO EMT VECTOR
1034 002372 000031      .WORD    31      ;UNIQUE ERROR NUMBER
1035 002374 001127      .WORD    CPUERR   ;ADDRESS OF ERROR MESSAGE
1036 002376 005004      CLR      R4      ;R4=0
1037 002400 020427 000000 1#:      CMP      R4,#0    ;DOES R4=0
1038 002404 001403      BEQ      2#      ;YES GO ON
1039                                     ;NO GO TO ERROR
1040 002406 104000      ERROR                                     ;ALL ERRORS TO TRAP TO EMT VECTOR
1041 002410 000032      .WORD    32      ;UNIQUE ERROR NUMBER
1042 002412 001127      .WORD    CPUERR   ;ADDRESS OF ERROR MESSAGE
1043 002414 012704 125252 2#:      MOV      #125252,R4 ;R4=125252
1044 002420 020427 125252      CMP      R4,#125252 ;DOES R4=125252
1045 002424 001403      BEQ      3#      ;YES GO ON
1046                                     ;NO GO TO ERROR
1047 002425 104000      ERROR                                     ;ALL ERRORS TO TRAP TO EMT VECTOR
1048 002430 000033      .WORD    33      ;UNIQUE ERROR NUMBER
1049 002432 001127      .WORD    CPUERR   ;ADDRESS OF ERROR MESSAGE
1050 002434 012704 052525 3#:      MOV      #52525,R4 ;R4=52525
1051 002440 020427 052525      CMP      R4,#52525 ;DOES R4=52525
1052 002444 001403      BEQ      4#      ;YES GO ON
1053                                     ;NO GO TO ERROR
1054 002446 104000      ERROR                                     ;ALL ERRORS TO TRAP TO EMT VECTOR
1055 002450 000034      .WORD    34      ;UNIQUE ERROR NUMBER
1056 002452 001127      .WORD    CPUERR   ;ADDRESS OF ERROR MESSAGE
1057 002454                                     ;
1058                                     ;
1059                                     ;
1060 002454      |
1061      GPR5TS:
1062      |;*****
1063      |;*TEST 14      R5 BIT TESTS
1064      |;*****
1065      |TST14:
1065 002454 005267 176324      INC      #TESTN   ;INCREMENT TEST NUMBER
1066 002460 012705 177777      MOV      #177777,R5 ;R5=177777
1067 002464 020527 177777      CMP      R5,#177777 ;DOES R5=177777
1068 002470 001403      BEQ      1#      ;YES GO ON
1069                                     ;NO GO TO ERROR
1070 002472 104000      ERROR                                     ;ALL ERRORS TO TRAP TO EMT VECTOR
1071 002474 000035      .WORD    35      ;UNIQUE ERROR NUMBER
1072 002476 001127      .WORD    CPUERR   ;ADDRESS OF ERROR MESSAGE
1073 002500 005005      CLR      R5      ;R5=0
1074 002502 020527 000000 1#:      CMP      R5,#0    ;DOES R5=0
1075 002506 001403      BEQ      2#      ;YES GO ON
1076                                     ;NO GO TO ERROR
1077 002510 104000      ERROR                                     ;ALL ERRORS TO TRAP TO EMT VECTOR
1078 002512 000036      .WORD    36      ;UNIQUE ERROR NUMBER
1079 002514 001127      .WORD    CPUERR   ;ADDRESS OF ERROR MESSAGE
1080 002516 012705 125252 2#:      MOV      #125252,R5 ;R5=125252
1081 002522 020527 125252      CMP      R5,#125252 ;DOES R5=125252
1082 002526 001403      BEQ      3#      ;YES GO ON
1083                                     ;NO GO TO ERROR
1084 002530 104000      ERROR                                     ;ALL ERRORS TO TRAP TO EMT VECTOR
1085 002532 000037      .WORD    37      ;UNIQUE ERROR NUMBER
1086 002534 001127      .WORD    CPUERR   ;ADDRESS OF ERROR MESSAGE

```

```

1087 002536 012705 052525      3$:  MOV      #52525,R5      ;R5=52525
1088 002542 020527 052525      CMP      R5,#52525      ;DOES R5=52525
1089 002546 001403              BEQ      4$             ;YES GO ON
1090                                ;NO GO TO ERROR
1091 002550 104000              ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
1092 002552 000040              .WORD    40            ;UNIQUE ERROR NUMBER
1093 002554 001127              .WORD    CPUERR       ;ADDRESS OF ERROR MESSAGE
1094 002556              4$:
1095
1096
1097 002556              ;
1098                                GPR6TS:
1099                                ;*****
1100                                ;*TEST 15      R6 BIT TESTS
1101                                ;*****
1101 002556              TST15:
1102 002556 005267 176222          INC      $TESTN        ;INCREMENT TEST NUMBER
1103 002562 012706 177777          MOV      #177777,R6   ;R6=177777
1104 002566 020627 177777          CMP      R6,#177777  ;DOES R6=177777
1105 002572 001403              BEQ      1$           ;YES GO ON
1106                                ;NO GO TO ERROR
1107 002574 104000              ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
1108 002576 000041              .WORD    41            ;UNIQUE ERROR NUMBER
1109 002600 001127              .WORD    CPUERR       ;ADDRESS OF ERROR MESSAGE
1110 002602 005006              1$:  CLR      R6          ;R6=0
1111 002604 020627 000000          CMP      R6,#0        ;DOES R6=0
1112 002610 001403              BEQ      2$           ;YES GO ON
1113                                ;NO GO TO ERROR
1114 002612 104000              ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
1115 002614 000042              .WORD    42            ;UNIQUE ERROR NUMBER
1116 002616 001127              .WORD    CPUERR       ;ADDRESS OF ERROR MESSAGE
1117 002620 012706 125252          2$:  MOV      #125252,R6  ;R6=125252
1118 002624 020627 125252          CMP      R6,#125252  ;DOES R6=125252
1119 002630 001403              BEQ      3$           ;YES GO ON
1120                                ;NO GO TO ERROR
1121 002632 104000              ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
1122 002634 000043              .WORD    43            ;UNIQUE ERROR NUMBER
1123 002636 001127              .WORD    CPUERR       ;ADDRESS OF ERROR MESSAGE
1124 002640 012706 052525          3$:  MOV      #52525,R6   ;R6=52525
1125 002644 020627 052525          CMP      R6,#52525   ;DOES R6=52525
1126 002650 001403              BEQ      4$           ;YES GO ON
1127                                ;NO GO TO ERROR
1128 002652 104000              ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
1129 002654 000044              .WORD    44            ;UNIQUE ERROR NUMBER
1130 002656 001127              .WORD    CPUERR       ;ADDRESS OF ERROR MESSAGE
1131 002660 012706 001000          4$:  MOV      #STBOT,R6   ;RESTORE SP
1132
1133
1134 002664              ;
1135                                PSWBTS:
1136                                ;*****
1137                                ;*TEST 16      PSW LOW BYTE BIT TESTS
1138                                ;*****
1138 002664              TST16:
1139 002664 005267 176114          INC      $TESTN        ;INCREMENT TEST NUMBER
1140 002670 012737 000377 177776      MOV      #377,#0177776 ;PS=357 T BIT SHOULDN'T SET
1141 002676 022737 000357 177776      CMP      #357,#0177776 ;DOES PS=357
1142 002704 001403              BEQ      1$           ;YES GO ON

```

```

1143
1144 002706 104000          ERROR
1145 002710 000045          .WORD 45
1146 002712 001127          .WORD CPUERR
1147 002714 005037 177776 1#: CLR 0#177776
1148 002720 022737 000000 177776 1#: CMP 00,0#177776
1149 002726 001403          BEQ 2#
1150
1151 002730 104000          ERROR
1152 002732 000046          .WORD 46
1153 002734 001127          .WORD CPUERR
1154 002736 012737 000105 177776 2#: MOV 0105,0#177776
1155 002744 022737 000105 177776 2#: CMP 0105,0#177776
1156 002752 001403          BEQ 3#
1157
1158 002754 104000          ERROR
1159 002756 000047          .WORD 47
1160 002760 001127          .WORD CPUERR
1161 002762 012737 000252 177776 3#: MOV 0252,0#177776
1162 002770 022737 000252 177776 3#: CMP 0252,0#177776
1163 002776 001403          BEQ 4#
1164
1165 003000 104000          ERROR
1166 003002 000050          .WORD 50
1167 003004 001127          .WORD CPUERR
1168 003006          4#:
1169
1170
1171 003006          MSP0:
1172          |*****
1173          |*TEST 17          TEST SINGLE OPERAND INSTRUCTIONS- MODE 0
1174          |*****
1175          |THE INC, COM, CLR, AND DECREMENT INSTRUCTIONS ARE VERIFIED.
1176          |*****
1177          |TST17:
1178 003006 005267 175772          INC 0TESTN          ;INCREMENT TEST NUMBER
1179 003012 005004          CLR R4              ;INITIALIZE R4 WITH DATA
1180 003014 005104          COM R4              ;
1181 003016 005004          CLR R4              ;*TEST INSTRUCTION
1182 003020 001403          BEQ 2#              ;BRANCH IF R4 CLEARED
1183 003022          1#:
1184 003022 104000          ERROR
1185 003024 000051          .WORD 51
1186 003026 001127          .WORD CPUERR
1187 003030 005104          2#: COM R4
1188 003032 005204          INC R4
1189 003034 001403          BEQ 4#
1190
1191 003036          3#:
1192 003036 104000          ERROR
1193 003040 000052          .WORD 52
1194 003042 001127          .WORD CPUERR
1195 003044          4#:
1196
1197          |
1198 003044          MSPB:

```

```

1199
1200
1201
1202 003044
1203 003044 005267 175734
1204 003050 005004
1205 003052 003104
1206 003054 105004
1207 003056 001403
1208
1209 003060
1210 003060 104000
1211 003062 000053
1212 003064 001127
1213 003066 105304
1214 003070 100002
1215 003072 105104
1216 003074 001403
1217
1218 003076
1219 003076 104000
1220 003100 000054
1221 003102 001127
1222 003104
1223
1224
1225 003104
1226
1227
1228
1229 003104
1230 003104 005267 175674
1231 003110 005004
1232 003112 005014
1233 003114 005114
1234 003116 005014
1235 003120 001403
1236
1237 003122
1238 003122 104000
1239 003124 000055
1240 003126 001127
1241 003130 005114
1242 003132 001403
1243 003134 100002
1244 003136 005214
1245 003140 001403
1246
1247 003142
1248 003142 104000
1249 003144 000056
1250 003146 001127
1251 003150
1252
1253
1254 003150

```

```

*****
; *TEST 20 TEST SINGLE OPS - EVEN BYTE OF CLRB, DECB, AND COMB
*****
TST20:
      INC      #TESTN          ; INCREMENT TEST NUMBER
      CLR      R4
      COM      R4              ; SETUP TEST REGISTER
      CLRB    R4              ; *TEST CLEAR BYTE INSTRUCTION
      BEQ     2#              ; BRANCH IF GOOD
                               ; CLEAR EVEN BYTE FAILED
1# :
      ERROR   53              ; ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD   CPUERR         ; UNIQUE ERROR NUMBER
      .WORD   000000         ; ADDRESS OF ERROR MESSAGE
2# :
      DECB   R4              ; *TEST DECREMENT BYTE
      BPL   3#              ; DECREMENT BYTE FAILED
      COMB   R4              ; *TEST COMPLIMENT BYTE
      BEQ   4#              ; BRANCH IF GOOD
                               ; COMPLIMENT OR DECREMENT FAILED TO WORK
3# :
      ERROR   54              ; ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD   CPUERR         ; UNIQUE ERROR NUMBER
      .WORD   000000         ; ADDRESS OF ERROR MESSAGE
4# :
      MSPC:
*****
; *TEST 21 TEST SINGLE OPS - MODE 1 CLRB, COMB, AND INCB
*****
TST21:
      INC      #TESTN          ; INCREMENT TEST NUMBER
      CLR      R4
      CLR      (R4)
      COM      (R4)           ; SETUP TEST DATA
      CLR      (R4)           ; *TEST INSTRUCTION
      BEQ     2#              ; BRANCH IF GOOD
                               ; MODE 1 FAILED
1# :
      ERROR   55              ; ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD   CPUERR         ; UNIQUE ERROR NUMBER
      .WORD   000000         ; ADDRESS OF ERROR MESSAGE
2# :
      COM     (R4)
      BEQ     3#              ; *TEST INSTRUCTION
                               ; (0) SHOULD = -1
      BPL   3#
      INC     (R4)
      BEQ     4#              ; *TEST INSTRUCTION
                               ; BRANCH IF GOOD
                               ; COM OR INC FAILED TO ALTER LOC 0 CORRECTLY
3# :
      ERROR   56              ; ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD   CPUERR         ; UNIQUE ERROR NUMBER
      .WORD   000000         ; ADDRESS OF ERROR MESSAGE
4# :
      MSPD:

```



```

1255
1256
1257
1258 003150
1259 003150 005267 175630
1260 003154 005004
1261 003156 005014
1262 003160 005114
1263 003162 105014
1264 003164 105014
1265 003166 001403
1266
1267 003170
1268 003170 104000
1269 003172 000057
1270 003174 001127
1271 003176 105214
1272 003200 100405
1273 003202 001404
1274 003204 105114
1275 003206 105214
1276 003210 105214
1277 003212 001403
1278
1279 003214
1280 003214 104000
1281 003216 000060
1282 003220 001127
1283 003222
1284
1285
1286 003222
1287
1288
1289
1290 003222
1291 003222 005267 175556
1292 003226 005004
1293 003230 005014
1294 003232 005114
1295 003234 005204
1296 003236 105014
1297 003240 001403
1298
1299 003242 104000
1300 003244 000061
1301 003246 001127
1302 003250 005304
1303 003252 005214
1304 003254 005204
1305 003256 105114
1306 003260 105214
1307 003262 100003
1308 003264 001402
1309 003266 105214
1310 003270 001403

```

```

*****
;+TEST 22 TEST SINGLE OPS MODE1-EVEN BYTE-CLRB,COMB,INCB
*****
TST22:
      INC      $TESTN          ;INCREMENT TEST NUMBER
      CLR      R4
      CLR      (R4)
      COM      (R4)           ;SETUP TEST DATA
      CLRB    (R4)           ;+TEST INSTRUCTION
      CLRB    (R4)           ;+TEST INSTRUCTION
      BEQ     2$             ;BRANCH IF GOOD
                          ;CLEAR (0) EVEN BYTE FAILED
1$:
      ERROR   57             ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD   CPUERR        ;UNIQUE ERROR NUMBER
                          ;ADDRESS OF ERROR MESSAGE
2$:
      INCB    (R4)           ;+TEST INSTRUCTION
      BMI     3$             ; TEST FLAGS
      BEQ     3$
      COMB   (R4)           ;+TEST INSTRUCTION
      INCB   (R4)
      INCB   (R4)
      BEQ    4$             ;BRANCH IF GOOD
                          ;COMB INCB FAILED
3$:
      ERROR   60             ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD   CPUERR        ;UNIQUE ERROR NUMBER
                          ;ADDRESS OF ERROR MESSAGE
4$:
      MSPE0:
*****
;+TEST 23 TEST SINGLE OPS - ODD BYTE - CLRB, COMB, DECB
*****
TST23:
      INC      $TESTN          ;INCREMENT TEST NUMBER
      CLR      R4
      CLR      (R4)
      COM      (R4)           ;SETUP TEST DATA
      INC      R4             ;POINT TO ODD BYTE
      CLRB    (R4)           ;+TEST INSTRUCTION
      BEQ     1$             ;BRANCH IF GOOD
                          ;CLEAR ODD BYTE FAILED
                          ;ALL ERRORS TO TRAP TO EMT VECTOR
1$:
      ERROR   61             ;UNIQUE ERROR NUMBER
      .WORD   CPUERR        ;ADDRESS OF ERROR MESSAGE
      DEC     R4             ;POINT TO EVEN BYTE
      INC     (R4)          ;LOC 0-1 0
      INC     R4             ;POINT TO ODD BYTE
      COMB   (R4)           ;+TEST INSTRUCTION
      INCB   (R4)          ;LOC 0--1 0
      EPL    2$             ;BRANCH IF ERROR
      BEQ    2$
      INCB   (R4)           ;+TEST INSTRUCTION
      BEQ    3$             ;BRANCH IF GOOD

```



```

1367 003402 000065          .WORD 65          ;UNIQUE ERROR NUMBER
1368 003404 001127          .WORD CPUERR      ;ADDRESS OF ERROR MESSAGE
1369 003406 005304          1#: DEC R4
1370 003410 105324          DECB (R4)+       ;*TEST INSTRUCTION
1371 003412 100003          BPL 2#           ;BRANCH IF BAD
1372 003414 005304          DEC R4           ;POINT TO EVEN BYTE
1373 003416 105124          COMB (R4)+      ;*TEST INSTRUCTION
1374 003420 001403          BEQ 3#          ;BRANCH IF GOOD
1375                                     ;MODE 2, EVEN BYTE FAILED
1376 003422
1377 003422 104000          2#: ERROR        ;ALL ERRORS TO TRAP TO EMT VECTOR
1378 003424 000066          .WORD 66          ;UNIQUE ERROR NUMBER
1379 003426 001127          .WORD CPUERR      ;ADDRESS OF ERROR MESSAGE
1380 003430          3#:
1381
1382 ;
1383 003430          ;MSPH:
1384          ;*****
1385          ;*TEST 26      TEST CLRB, COMB, INCB MODE 2 - ODD BYTE
1386          ;*****
1387 003430          TST26:
1388 003430 005267 175350      INC $TESTN       ;INCREMENT TEST NUMBER
1389 003434 005004          CLR R4
1390 003436 105104          COMB R4
1391 003440 005204          INC R4           ;R4=400
1392 003442 005014          CLR (R4)
1393 003444 005114          COM (R4)        ;400=-1 -1
1394 003446 005214          INC (R4)        ;POINT TO ODD BYTE
1395 003450 105024          CLRB (R4)+     ;*TEST INSTRUCTION
1396 003452 001403          BEQ 1#          ;BRANCH IF GOOD
1397                                     ;MODE 2, ODD BYTE FAILED
1398 003454 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
1399 003456 000067          .WORD 67          ;UNIQUE ERROR NUMBER
1400 003460 001127          .WORD CPUERR      ;ADDRESS OF ERROR MESSAGE
1401 003462 005304          1#: DEC R4
1402 003464 005304          DEC R4
1403 003466 105224          INCB (R4)+     ;400=1 0
1404 003470 105124          COMB (R4)+     ;*TEST INSTRUCTION
1405 003472 100003          BPL 2#         ;BRANCH IF MODE 2 FAILED
1406 003474 005304          DEC R4         ;POINT TO ODD BYTE
1407 003476 105224          INCB (R4)+
1408 003500 001403          BEQ 3#         ;BRANCH IF GOOD
1409                                     ;MODE 2, ODD BYTE FAILED
1410 003502
1411 003502 104000          2#: ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
1412 003504 000070          .WORD 70          ;UNIQUE ERROR NUMBER
1413 003506 001127          .WORD CPUERR      ;ADDRESS OF ERROR MESSAGE
1414 003510          3#:
1415
1416 ;
1417 003510          ;MSPI:
1418          ;*****
1419          ;*TEST 27      TEST CLR, COM, INC - MODE 3
1420          ;*****
1421 003510          TST27:
1422 003510 005267 175270      INC $TESTN       ;INCREMENT TEST NUMBER

```

```

1423 003514 005004 CLR R4 ;
1424 003516 005014 CLR (R4) ;0=0
1425 003520 105114 COMB (R4) ;
1426 003522 005214 INC (R4) ;0=400
1427 003524 005034 CLR B(R4) ;*TEST INSTRUCTION
1428 003526 001403 BEQ 1# ;BRANCH IF GOOD
1429 ;MODE 3 FAILED, 400 SHOULD=0
1430 003530 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1431 003532 000071 .WORD 71 ;UNIQUE ERROR NUMBER
1432 003534 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
1433 003536 005304 1# : DEC R4 ;
1434 003540 005304 DEC R4 ;R4=0
1435 003542 005134 COM B(R4)+ ;*TEST INSTRUCTION
1436 003544 100004 BPL 2# ;BRANCH IF BAD
1437 003546 005304 DEC R4 ;
1438 003550 005304 DEC R4 ;REPOSITION POINTER
1439 003552 005234 INC B(R4)+ ;*TEST INSTRUCTION
1440 003554 001403 BEQ 3# ;BRANCH IF GOOD
1441 ;MODE 3 FAILED
1442 003556 2# :
1443 003556 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1444 003560 000072 .WORD 72 ;UNIQUE ERROR NUMBER
1445 003562 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
1446 003564 3# :
1447 ;
1448 ;
1449 003564 MSPJ:
1450 ;*****
1451 ;*TEST 30 TEST CLRB, COMB, INCB - MODE 3, EVEN/ODD BYTE
1452 ;*****
1453 003564 TST30:
1454 003564 005267 175214 INC #TESTN ;INCREMENT TEST NUMBER
1455 003570 005004 CLR R4 ;R4=0
1456 003572 005001 CLR R1 ;
1457 003574 105101 COMB R1 ;
1458 003576 005201 INC R1 ;R1=400
1459 003600 005011 CLR (R1) ;
1460 003602 005121 COM (R1)+ ;400=1
1461 003604 005011 CLR (R1) ;
1462 003606 105111 COMB (R1) ;402=000 377
1463 003610 005014 CLR (R4) ;
1464 003612 105114 COMB (R4) ;
1465 003614 005214 INC (R4) ;0=400
1466 003616 105034 CLRB B(R4)+ ;*TEST INSTRUCTION 400=377 000
1467 003620 001403 BEQ 1# ;BRANCH IF MODE 3 EVEN BYTE CLEARED
1468 ;TEST INSTRUCTION FAILED
1469 003622 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1470 003624 000073 .WORD 73 ;UNIQUE ERROR NUMBER
1471 003626 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
1472 003630 005304 1# : DEC R4 ;REPOSITION POINTER
1473 003632 005304 DEC R4 ;
1474 003634 105134 COMB B(R4)+ ;*TEST INSTRUCTION
1475 003636 005304 DEC R4 ;
1476 003640 005304 DEC R4 ;REPOSITION POINTER
1477 003642 105234 INCB B(R4)+ ;*TEST INSTRUCTION
1478 003644 001403 BEQ 3# ;BRANCH IF GOOD

```

```

1479                                     ;MODE 3, EVEN BYTE FAILED
1480 003646                               2#:
1481 003646 104000                       ERROR
1482 003650 000074                       .WORD 74
1483 003652 001127                       .WORD CPUERR
1484 003654 005304                       3#: DEC R4
1485 003656 005304                       DEC R4
1486 003660 005214                       INC (R4)
1487 003662 105234                       INCB B(R4)+
1488 003664 001004                       BNE 4#
1489 003666 005304                       DEC R4
1490 003670 005304                       DEC R4
1491 003672 105034                       CLR B B(R4)+
1492 003674 001403                       BEQ 5#
1493
1494 003676                               4#:
1495 003676 104000                       ERROR
1496 003700 000075                       .WORD 75
1497 003702 001127                       .WORD CPUERR
1498 003704 005304                       5#: DEC R4
1499 003705 005304                       DEC R4
1500 003710 105134                       COM B(R4)+
1501 003712 005304                       DEC R4
1502 003714 005304                       DEC R4
1503 003716 105234                       INCB B(R4)+
1504 003720 001403                       BEQ 7#
1505
1506 003722                               6#:
1507 003722 104000                       ERROR
1508 003724 000076                       .WORD 76
1509 003726 001127                       .WORD CPUERR
1510 003730                               7#:
1511
1512
1513 003730                               |
1514                                     ;*****
1515                                     ;*TEST 31 TEST CLR, COM, DEC - MODE 4
1516                                     ;*****
1517 003730                               TST31:
1518 003730 005267 175050                 INC $TESTN
1519 003734 005004                       CLR R4
1520 003736 105104                       COMB R4
1521 003740 005204                       INC R4
1522 003742 005014                       CLR (R4)
1523 003744 005124                       COM (R4)+
1524 003746 005014                       CLR (R4)
1525 003750 005224                       INC (R4)+
1526 003752 005044                       CLR -(R4)
1527 003754 001403                       BEQ 1#
1528
1529 003756 104000                       ERROR
1530 003760 000077                       .WORD 77
1531 003762 001127                       .WORD CPUERR
1532 003764 005344                       1#: DEC -(R4)
1533 003766 005114                       COM (R4)
1534 003770 001405                       BEQ 2#

```

```

;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;R4=401
;*TEST INSTRUCTION
;BRANCH IF 402 NEQ 0
;R4=401
;401=0
;BRANCH IF GOOD
;ODD BYTE FAILED
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;R4=401
;403=377
;*TEST INSTRUCTION
;BRANCH IF GOOD
;MODE3 ODD BYTE FAILED.
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;INCREMENT TEST NUMBER
;R4=400
;
;400=-1
;
;402=1
;*TEST INSTRUCTION
;BRANCH IF GOOD
;MODE 4 FAILED
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;*TEST INSTRUCTION 400=-2
;400=1
;BRANCH IF BAD

```

```

1535 003772 100404      BMI      2#           ;BRANCH IF BAD
1536 003774 005204      INC      R4
1537 003776 005204      INC      R4           ;R4=400
1538 004000 005344      DEC      -(R4)       ;*TEST INSTRUCTION
1539 004002 001403      BEQ      3#           ;BRANCH IF GOOD
1540
1541 004004              2#:
1542 004004 104000      ERROR           ;ALL ERRORS TO TRAP TO EMT VECTOR
1543 004006 000100      .WORD      100      ;UNIQUE ERROR NUMBER
1544 004010 001127      .WORD      CPUERR   ;ADDRESS OF ERROR MESSAGE
1545 004012
1546
1547
1548 004012              3#:
1549
1550
1551
1552 004012              |
1553 004012 005267 174766      MSPM:
1554 004016 005004      |*****
1555 004020 105104      |*TEST 32      TEST COMB, INCB, CLR8 - MODE 4, ODD BYTE
1556 004022 005204      |*****
1557 004024 005044      TST32:
1558 004026 105114      INC      #TESTN     ;INCREMENT TEST NUMBER
1559 004030 005224      CLR      R4
1560 004032 005014      CO#B     R4
1561 004034 005124      INC      R4           ;R4=400
1562 004036 005204      CLR      -(R4)       ;376=0
1563 004040 105044      COMB     (R4)
1564 004042 001403      INC      (R4)+       ;376=001 000
1565
1566 004044              1#:
1567 004044 104000      ERROR           ;ALL ERRORS TO TRAP TO EMT VECTOR
1568 004046 000101      .WORD      101      ;UNIQUE ERROR NUMBER
1569 004050 001127      .WORD      CPUERR   ;ADDRESS OF ERROR MESSAGE
1570 004052 005204      2#:
1571 004054 005204      INC      R4           ;R4=403
1572 004056 105144      INC      R4           ; TEST INST. 401=377
1573 004060 005304      COMB     -(R4)
1574 004062 005304      DEC      R4
1575 004064 105244      DEC      R4
1576 004066 001403      INCB     -(R4)       ; TEST INST. 401=0
1577
1578 004070              3#:
1579 004070 104000      ERROR           ;ALL ERRORS TO TRAP TO EMT VECTOR
1580 004072 000102      .WORD      102      ;UNIQUE ERROR NUMBER
1581 004074 001127      .WORD      CPUERR   ;ADDRESS OF ERROR MESSAGE
1582 004076 105344      4#:
1583 004100 001403      DECB     -(R4)       ;*TEST INST.
1584
1585 004102              5#:
1586 004102 104000      ERROR           ;ALL ERRORS TO TRAP TO EMT VECTOR
1587 004104 000103      .WORD      103      ;UNIQUE ERROR NUMBER
1588 004106 001127      .WORD      CPUERR   ;ADDRESS OF ERROR MESSAGE
1589 004110              6#:
1590

```

1591  
1592 004110  
1593  
1594  
1595  
1596 004110  
1597 004110 005267 174670  
1598 004114 005004  
1599 004116 005014  
1600 004120 105114  
1601 004122 005224  
1602 004124 005054  
1603 004126 001403  
1604  
1605 004130 104000  
1606 004132 000104  
1607 004134 001127  
1608 004136 005204  
1609 004140 005204  
1610 004142 005154  
1611 004144 001407  
1612 004146 005204  
1613 004150 005204  
1614 004152 005354  
1615 004154 001403  
1616 004156 005224  
1617 004160 105254  
1618 004162 001403  
1619  
1620 004164  
1621 004164 104000  
1622 004166 000105  
1623 004170 001127  
1624 004172  
1625  
1626  
1627 004172  
1628  
1629  
1630  
1631 004172  
1632 004172 005267 174606  
1633 004176 005004  
1634 004200 105104  
1635 004202 005204  
1636 004204 005001  
1637 004206 105101  
1638 004210 005301  
1639 004212 005002  
1640 004214 005012  
1641 004216 005014  
1642 004220 005114  
1643 004222 005011  
1644 004224 005454  
1645 004226 001403  
1646

```

MSPN:
;*****
;*TEST 33      TEST CLR, COM, INC - MODE 5
;*****
TST33:
    INC      #TESTN          ;INCREMENT TEST NUMBER
    CLR      R4
    CLR      (R4)
    COMB     (R4)
    INC      (R4)+          ;O=400
    CLR      @-(R4)        ;*TEST INST. 400=0
    BEQ      1$            ;BRANCH IF GOOD
                                ;MODE 5 FAILED
    ERROR    104           ;ALL ERRORS TO TRAP TO EMT VECTOR
    .WORD   CPUERR        ;UNIQUE ERROR NUMBER
    .WORD   CPUERR        ;ADDRESS OF ERROR MESSAGE
1$:
    INC      R4
    INC      R4
    COM      @-(R4)
    BEQ      2$
    INC      R4
    INC      R4
    DEC      @-(R4)        ;REPOSITION POINTER
    BEQ      2$           ;*TEST INST. 376=0
    INC      (R4)+        ;BRANCH IF BAD
    INCB     @-(R4)       ;O=401 R4=2
    BEQ      3$           ;*TEST INST. 400= 0 376
                                ;BRANCH IF GOOD
                                ;MODE 5 FAILED
2$:
    ERROR    105           ;ALL ERRORS TO TRAP TO EMT VECTOR
    .WORD   CPUERR        ;UNIQUE ERROR NUMBER
    .WORD   CPUERR        ;ADDRESS OF ERROR MESSAGE
3$:
MSP0:
;*****
;*TEST 34      TEST NEG MODE 5
;*****
TST34:
    INC      #TESTN          ;INCREMENT TEST NUMBER
    CLR      R4
    COMB     R4
    INC      R4              ;R4=400
    CLR      R1
    COMB     R1
    DEC      R1              ;R1=376
    CLR      R2              ;R2=0
    CLR      (R2)           ;O=0
    CLR      (R4)
    COM      (R4)           ;400=-1
    CLR      (R1)           ;376=0
    NEG      @-(R4)        ;O=0
    BEQ      2$            ;BRANCH IF GOOD
                                ;NEG FAILED
2$:

```

K3

```

1647 004230
1648 004230 104000
1649 004232 000106
1650 004234 001127
1651 004236 005334
1652 004240 005454
1653 004242 001403
1654 004244 102402
1655 004246 100401
1656 004250 103403
1657
1658 004252
1659 004252 104000
1660 004254 000107
1661 004256 001127
1662 004260 005334
1663 004262 001403
1664
1665 004264
1666 004264 104000
1667 004266 000110
1668 004270 001127
1669 004272 105212
1670 004274 005454
1671 004276 001403
1672 004300 102402
1673 004302 103001
1674 004304 100403
1675
1676 004306
1677 004306 104000
1678 004310 000111
1679 004312 001127
1680 004314 105212
1681 004316 001403
1682 004320
1683 004320 104000
1684 004322 000112
1685 004324 001127
1686 004326
1687
1688
1689
1690 004326
1691
1692
1693
1694 004326
1695 004326 005267 174452
1696 004332 005004
1697 004334 005204
1698 004336 005204
1699 004340 005001
1700 004342 105101
1701 004344 005201
1702 004346 005011

```

```

1#:
ERROR
.WORD 106
.WORD CPUERR
DEC 8(R4)+
NEG 8-(R4)
BEQ 3#
EVS 3#
BMI 3#
BCS 4#
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;0=-1
;0=1
;BRANCH IF BAD
;BRANCH IF BAD
;BRANCH IF BAD
;BRANCH IF GOOD
;NEG FAILED

2#:
ERROR
.WORD 107
.WORD CPUERR
DEC 8(R4)+
BEQ 6#
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;TEST RESULT OF NEGATE
;BRANCH IF GOOD
;RESULT OF NEGATE BAD

3#:
ERROR
.WORD 110
.WORD CPUERR
INCB (R2)
NEG 8-(R4)
BEQ 7#
BVS 7#
BCC 7#
BMI 8#
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;0=1
;0=-1
;BRANCH IF GOOD
;BAD NEGATE

4#:
ERROR
.WORD 111
.WORD CPUERR
INCB (R2)
BEQ 10#
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;0=0
;BRANCH IF GOOD

5#:
ERROR
.WORD 112
.WORD CPUERR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE

```

```

NSPP:
;*****
;*TEST 35 TEST CLR, COM, INC - MODE 6
;*****
TST35:
INC #TESTN ;INCREMENT TEST NUMBER
CLR R4
INC R4
INC R4 ;R4+2
CLR R1
COMB R1
INC R1 ;R1=400
CLR (R1)

```



```

1703 004350 005121 COM (R1)+ ;400=-1
1704 004352 005011 CLR (R1) ;R1=402
1705 004354 005211 INC (R1) ;402=1
1706 004356 005002 CLR R2 ;R2=0
1707 004360 005012 CLR (R2) ;0=0
1708 004362 005064 000376 CLR 376(R4) ;400=0
1709 004366 001403 BEQ 2# ;BRANCH IF GOOD
1710 004370 1# :
1711 004370 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1712 004372 000113 .WORD 113 ;UNIQUE ERROR NUMBER
1713 004374 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
1714 004376 005364 000376 2# : DEC 376(R4) ;400=-1
1715 004402 005164 000400 COM 400(R4) ;402=-1
1716 004406 001405 BEQ 3# ;BRANCH IF BAD
1717 004410 005264 000400 INC 400(R4) ;402=-2
1718 004414 005264 000400 INC 400(R4)
1719 004420 001403 BEQ 4# ;BRANCH IF GOOD
1720 ;MODE 6 FAILED
1721 004422 3# :
1722 004422 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1723 004424 000114 .WORD 114 ;UNIQUE ERROR NUMBER
1724 004426 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
1725 004430 005261 177776 4# : INC -2(R1) ;400=0
1726 004434 001403 BEQ 6# ;BRANCH IF GOOD
1727 ;ERROR! INC MODE 6 FAILED
1728 004436 5# :
1729 004436 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1730 004440 000115 .WORD 115 ;UNIQUE ERROR NUMBER
1731 004442 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
1732 004444 6# :
1733 ;
1734 ;
1735 ;
1736 ;
1737 004444 MSPQ:
1738 ;*****
1739 ;*TEST 36 TEST NEG MODE 6
1740 ;*****
1741 004444 TST36:
1742 004444 005267 174334 INC #TESTN ;INCREMENT TEST NUMBER
1743 004450 005001 CLR R1 ;R1=0
1744 004452 005004 CLR R4
1745 004454 105104 COMB R4
1746 004456 005204 INC R4 ;R4=100
1747 004460 005014 CLR (R4)
1748 004462 005114 COM (R4) ;400=-1
1749 004464 005044 CLR -(R4) ;376=0
1750 004466 005044 CLR -(R4)
1751 004470 005224 INC (R4)+ ;374=1 R4=376
1752 004472 005464 000002 NEG 2(R4) ;400=1
1753 004475 001403 BEQ 1# ;NEGATE FAILED
1754 004500 102402 BVS 1#
1755 004502 100401 BMT 1#
1756 004504 103403 BBS 2# ;BRANCH IF GOOD
1757 ;NEGATE FAILED
1758 004506 1# :

```

```

1759 004506 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
1760 004510 000116          .WORD        116          ;UNIQUE ERROR NUMBER
1761 004512 001127          .WORD        CPUERR      ;ADDRESS OF ERROR MESSAGE
1762 004514 005364 000002  2$: DEC        2(R4)      ; TEST RESULT OF NEGATE
1763 004520 001403          BEQ         4$          ;BRANCH IF GOOD
1764                                     ;RESULT OF NEGATE FAILED
1765 004522                                     ;
1766 004522 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
1767 004524 000117          .WORD        117          ;UNIQUE ERROR NUMBER
1768 004526 001127          .WORD        CPUERR      ;ADDRESS OF ERROR MESSAGE
1769 004530 005464 000000  4$: NEG        0(R4)      ;+0=0
1770 004534 001403          BEQ         5$          ; BRANCH IF GOOD
1771                                     ;NEGATE FAILED
1772 004536 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
1773 004540 000120          .WORD        120          ;UNIQUE ERROR NUMBER
1774 004542 001127          .WORD        CPUERR      ;ADDRESS OF ERROR MESSAGE
1775 004544 105461 000374  5$: NEGR       374(R1)    ;374=0 377
1776 004550 102403          BVS         6$          ;
1777 004552 001402          BEQ         6$          ;
1778 004554 100001          BPL         6$          ;
1779 004556 103403          BCS         7$          ;BRANCH IF GOOD
1780                                     ;NEGATE FAILED
1781 004560                                     ;
1782 004560 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
1783 004562 000121          .WORD        121          ;UNIQUE ERROR NUMBER
1784 004564 001127          .WORD        CPUERR      ;ADDRESS OF ERROR MESSAGE
1785 004566 105261 000374  7$: INCB       374(R1)    ;374=0
1786 004572 001403          BEQ         9$          ;BRANCH IF GOOD
1787                                     ;NEGATE FAILED
1788 004574                                     ;
1789 004574 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
1790 004576 000122          .WORD        122          ;UNIQUE ERROR NUMBER
1791 004600 001127          .WORD        CPUERR      ;ADDRESS OF ERROR MESSAGE
1792 004602                                     ;
1793                                     ;
1794                                     ;
1795 004602                                     ;MSPR:
1796                                     ;*****
1797                                     ;*TEST 37      TEST CLR, COM, INC - MODE 7
1798                                     ;*****
1799 004602                                     ;TST37:
1800 004602 005267 174176  INC         $TESTN      ;INCREMENT TEST NUMBER
1801 004606 005001          CLR         R1          ;R1=0
1802 004610 005004          CLR         R4          ;
1803 004612 105104          COMB       R4          ;
1804 004614 005204          INC         R4          ;R4=400
1805 004616 005011          CLR         (R1)
1806 004620 105111          COMB       (R1)
1807 004622 005211          INC         (R1)
1808 004624 005211          INC         (R1)
1809 004626 005211          INC         (R1)
1810 004630 005011          CLR         (R4)      ;;0=402
1811 004632 005064 000002  CLR         2(R4)      ;400=0
1812 004636 005164 000002  COM         2(R4)      ;
1813 004642 005074 177400  CLR         B-400(R4)  ;402=-1
1814 004646 001403          BEQ         2$          ;BRANCH IF GOOD

```

```

1815 004650 1$:
1816 004650 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1817 004652 000123 .WORD 123 ;UNIQUE ERROR NUMBER
1818 004654 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
1819 ;INSTRUCTION FAILED
1820 004656 005171 000000 2$: COM 80(R1) ;402=-1
1821 004662 100403 BMI 4$ ;BRANCH IF GOOD
1822 004664 3$:
1823 004664 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1824 004666 000124 .WORD 124 ;UNIQUE ERROR NUMBER
1825 004670 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
1826
1827 004672 005104 4$: COM R4
1828 004674 005274 000401 IFC 8401(R4) ;402=0
1829 004700 001403 BEQ 6$ ;BRANCH IF GOOD
1830 004702 5$:
1831 004702 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1832 004704 000125 .WORD 125 ;UNIQUE ERROR NUMBER
1833 004706 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
1834 ;MODE 7 FAILED
1835 004710 6$:
1836
1837 ;
1838 004710 ;ISPS:
1839 ;:*****
1840 ;*TEST 40 TEST NEG MODE 7
1841 ;:*****
1842 ;TST40:
1843 004710 005267 174070 INC ;TESTN ;INCREMENT TEST NUMBER
1844 004714 005004 CLR R4
1845 004716 005014 CLR (R4) ;0=0
1846 004720 005002 CLR R2 ;
1847 004722 105102 COMB R2
1848 004724 005202 INC R2 ;R2=400
1849 004726 005012 CLR (R2) ;400=0
1850 004730 005472 177400 NEG 8-400(R2) ;NEG OF 0=0
1851 004734 103401 BCS 2$ ;****
1852 004736 001403 BEQ 2$ ;BRANCH IF GOOD
1853 004740 1$:
1854 004740 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1855 004742 000126 .WORD 126 ;UNIQUE ERROR NUMBER
1856 004744 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
1857 ;
1858 004746 005314 2$: DEC (R4) ;0=-1
1859 004750 005474 000400 NEG 8+400(R4) ;0=1
1860 004754 001403 BEQ 3$ ;BRANCH IF ERROR
1861 004756 102402 BVS 3$ ;
1862 004760 100401 BMI 3$ ;
1863 004762 103403 BCS 4$ ;BRANCH IF GOOD
1864 004764 3$:
1865 004764 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1866 004766 000127 .WORD 127 ;UNIQUE ERROR NUMBER
1867 004770 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
1868 ;NEGATE MODE 7 FAILED
1869 004772 4.:
1870

```

```

1871
1872 004772
1873
1874
1875
1876 004772
1877 004772 005267 174006
1878 004776 005004
1879 005000 105104
1880 005002 005204
1881 005004 005027
1882 005006 177777
1883 005010 001403
1884 005012
1885 005012 104000
1886 005014 000130
1887 005016 001127
1888 005020
1889
1890
1891
1892 005020
1893
1894
1895
1896 005020
1897 005020 005267 173760
1898 005024 005004
1899 005026 000277
1900 005030 000244
1901 005032 005704
1902 005034 103403
1903 005036 102402
1904 005040 100401
1905 005042 001403
1906 005044
1907 005044 104000
1908 005046 000131
1909 005050 001127
1910
1911 005052 005304
1912 005054 000277
1913 005056 000250
1914 005060 005704
1915 005062 103403
1916 005064 102402
1917 005066 001401
1918 005070 100403
1919 005072
1920 005072 104000
1921 005074 000132
1922 005076 001127
1923
1924 005100
1925
1926

;
MSPT:
;*****
; *TEST 41 TEST SINGLE OPERAND MODE 2 REG 7
;*****
TST41:
      INC      $TESTN          ;INCREMENT TEST NUMBER
      CLR      R4
      COMB    R4
      INC      R4              ;R4=400
      CLR     (R7)+           ;CLEAR NEXT LOCATION
10:   .WORD    -1              ;SETUP INITIAL DATA
      BEQ     30              ;BRANCH IF GOOD
20:   ERROR                   ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD   130              ;UNIQUE ERROR NUMBER
      .WORD   CPUERR           ;ADDRESS OF ERROR MESSAGE
30:
;
;
MSPU:
;*****
; *TEST 42 TEST TST MODE 0
;*****
TST42:
      INC      $TESTN          ;INCREMENT TEST NUMBER
      CLR      R4              ;R4=0
      SCC     R4              ;CONDITION CODES =1111
      CLZ     R4              ;CC=1011
      TST     R4              ;*TEST INSTRUCTION
      BCS    10
      BVS    10
      BMI    10              ;BRANCH IF ERROR
      BEQ    20              ;BRANCH IF GOOD
10:   ERROR                   ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD   131              ;UNIQUE ERROR NUMBER
      .WORD   CPUERR           ;ADDRESS OF ERROR MESSAGE
      TST    MODE 0 FAILED
      R4    -1
20:   DEC      R4
      SCC     R4
      CLN    R4              ;CC=0111
      TST     R4              ;*TEST INSTRUCTION MODE 0
      BCS    30
      BVS    30
      BEQ    30              ;BRANCH IF ERROR
      BMI    40              ;BRANCH IF GOOD
30:   ERROR                   ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD   132              ;UNIQUE ERROR NUMBER
      .WORD   CPUERR           ;ADDRESS OF ERROR MESSAGE
      TST    FAILED
40:
;

```

```

1927 005100 MSPV0:
1928 ;*****
1929 ;*TEST 43 TEST TST MODE 0 BYTE
1930 ;*****
1931 005100 TST43:
1932 005100 005267 173700 INC #TESTN ;INCREMENT TEST NUMBER
1933 005104 005004 CLR R4
1934 005106 105104 COMB R4 ;0=000 377
1935 005110 000277 SCC
1936 005112 000250 CLN ;CC=0111
1937 005114 105704 TSTB R4 ;*TEST INSTRUCTION ON EVEN BYTE
1938 005116 102403 BVS 1# ;BRANCH IF ERROR
1939 005120 103402 BCS 1#
1940 005122 102401 BVS 1#
1941 005124 100403 BMI 2# ;BRANCH IF GOOD
1942 005126
1943 005126 104000 1#: ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1944 005130 000133 .WORD 133 ;UNIQUE ERROR NUMBER
1945 005132 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
1946
1947 005134 005204 2#: INC R4 ;POINT TO 1
1948 005136 105704 TSTB R4 ;TEST INSTRUCTION
1949 005140 001403 BEQ 4# ;BRANCH IF GOOD
1950
1951 005142 104000 3#: ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1952 005144 000134 .WORD 134 ;UNIQUE ERROR NUMBER
1953 005146 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
1954 ;TST FAILED ON BYTE
1955 005150 4#:
1956
1957
1958 005150 ;
1959 MSPV:
1960 ;*****
1961 ;*TEST 44 TEST TST MODE 1
1962 ;*****
1963 005150 TST44:
1964 005154 005267 173630 INC #TESTN ;INCREMENT TEST NUMBER
1965 005156 005014 CLR R4
1966 005160 000277 CLR (R4) ;0=0
1967 005162 000244 SCC
1968 005164 005714 CLZ ;CC=1011
1969 005166 103403 TST (R4) ;*TEST INSTRUCTION IN MODE 1
1970 005170 102402 BCS 1# ;BRANCH IF ERROR
1971 005172 100401 BVS 1#
1972 005174 001403 BMI 1#
1973 005176 001403 BEQ 2# ;BRANCH IF GOOD
1974 005176 104000 1#: ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1975 005200 000135 .WORD 135 ;UNIQUE ERROR NUMBER
1976 005202 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
1977
1978 005204 005214 2#: INC (R4) ;0=1
1979 005206 000277 SCC
1980 005210 005714 TST (R4) ;TEST INSTRUCTION
1981 005212 001403 BEQ 3# ;BRANCH IF ERROR
1982 005214 102402 BVS 3#

```

```

1983 005216 103401          BCS      3#
1984 005220 100003          BPL      4#          ;BRANCH IF GOOD
1985 005222                3#;
1986 005222 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
1987 005224 000136          .WORD    136          ;UNIQUE ERROR NUMBER
1988 005226 001127          .WORD    CPUERR       ;ADDRESS OF ERROR MESSAGE
1989                                ;TST FAILED MODE 1
1990 005230                4#;
1991
1992
1993 005230                ;
1994                                ;MSPX:
1995                                ;*****
1996                                ;*TEST 45      TEST TST MODE 1 BYTE
1997                                ;*****
1998 005230                TST45:
1999 005234 005267 173550          INC      #TESTN          ;INCREMENT TEST NUMBER
2000 005236 005014          CLR      R4              ;R4=0
2001 005240 105114          CLR      (R4)
2002 005242 005214          COMB    (R4)
2003 005244 000277          INC      (R4)          ;O=001 000
2004 005246 000244          SCC
2005 005250 105714          CLZ          ;CC=1011
2006 005252 103403          TSTB    (R4)          ;*TEST INSTRUCTION
2007 005254 102402          BCS     1#              ;BRANCH IF ERROR
2008 005256 100401          BVS     1#
2009 005260 001403          BMI     1#
2010 005262                BEQ     2#              ;BRANCH IF GOOD
2011 005262 104000          1#;
2012 005264 000137          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
2013 005266 001127          .WORD    137          ;UNIQUE ERROR NUMBER
2014                                .WORD    CPUERR       ;ADDRESS OF ERROR MESSAGE
2015 005270 005204          2#;
2016 005272 000277          INC      R4              ;R4=1
2017 005274 105714          SCC
2018 005276 001403          TSTB    (R4)          ; TEST INSTRUCTION
2019 005300 100402          BEQ     3#              ;BRANCH IF ERROR
2020 005302 102401          BMI     3#
2021 005304 103003          BVS     3#
2022 005306                BEQ     4#              ;BRANCH IF GOOD
2023 005306 104000          3#;
2024 005310 000140          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
2025 005312 001127          .WORD    140          ;UNIQUE ERROR NUMBER
2026                                .WORD    CPUERR       ;ADDRESS OF ERROR MESSAGE
2027                                ;
2028
2029
2030 005314                ;
2031                                ;COPY:
2032                                ;*****
2033                                ;*TEST 46      TEST TST MODE 2
2034                                ;*****
2035 005314 005267 173464          TST46:
2036 005320 005004          INC      #TESTN          ;INCREMENT TEST NUMBER
2037 005322 005024          CLR      R4              ;
2038 005324 005014          CLR      (R4)          ;O=0
                                CLR      (R4)

```

```

2039 005326 005114 COM (R4) ;2=-1
2040 005330 005004 CLR R4 ;R4=0
2041 005332 000277 SCC ;
2042 005334 000244 CLZ ;CC=1011
2043 005336 005724 TST (R4)+ ;TEST INSTRUCTION
2044 005340 103403 BCS 1# ;BRANCH IF ERROR
2045 005342 102402 BVS 1#
2046 005344 100401 BMI 1#
2047 005346 001403 BEQ 2# ;BRANCH IF GOOD
2048 005350 1# ;
2049 005350 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
2050 005352 000141 .WORD 141 ;UNIQUE ERROR NUMBER
2051 005354 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
2052 ;MODE 2 TEST FAILED
2053 005356 005724 2# TST (R4)+ ;TST LOC2
2054 005360 103403 BCS 3#
2055 005362 102402 BVS 3#
2056 005364 001401 BEQ 3#
2057 005366 100403 BMI 4#
2058 005370 3# ;
2059 005370 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
2060 005372 000142 .WORD 142 ;UNIQUE ERROR NUMBER
2061 005374 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
2062 ;MODE 2 FAILED
2063 005376 4# ;
2064 ;
2065 ;
2066 005376 ;
2067 MSPZ:
2068 ;*****
2069 ;TEST 47 TEST TST MODE 2 BYTE
2070 ;*****
2070 005376 TST47:
2071 005376 005267 173402 INC #TESTN ;INCREMENT TEST NUMBER
2072 005402 005004 CLR R4
2073 005404 005024 CLR (R4)+ ;
2074 005406 105144 COMB -(R4) ;0=377 000
2075 005410 005304 DEC R4 ;R4=0
2076 005412 000277 SCC ;
2077 005414 000244 CLZ ;CC=1011
2078 005416 105724 TSTB (R4)+ ;
2079 005420 102403 BVS 1# ;BRANCH IF ERROR
2080 005422 103402 BCS 1#
2081 005424 100401 BMI 1#
2082 005426 001403 BEQ 2# ;BRANCH IF GOOD
2083 005430 1# ;
2084 005430 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
2085 005432 000143 .WORD 143 ;UNIQUE ERROR NUMBER
2086 005434 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
2087 ;MODE 2 EVEN BYTE FAILED
2088 005436 000277 2# SCC ;
2089 005440 000250 CLN ;CC=0111
2090 005442 105724 TSTB (R4)+ ;
2091 005444 001403 BEQ 3# ;
2092 005446 103402 BCS 3#
2093 005450 102401 BVS 3#
2094 005452 100403 BMI 4# ;BRANCH IF GOOD

```

```

2095 005454
2096 005454 104000 3$: ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
2097 005456 000144 .WORD 144 ;UNIQUE ERROR NUMBER
2098 005460 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
2099 ;MODE 2 ODD BYTE FAILED
2100 005462
2101
2102
2103 005462
2104 MSPAA:
2105 *****
2106 TEST 50 TEST TST MODE 3
2107 *****
2108 005462 005267 173316 TST50: INC ;TESTN ;INCREMENT TEST NUMBER
2109 005466 005034 CLR R4
2110 005470 005014 CLR (R4)
2111 005472 105114 COMB (R4)
2112 005474 005214 INC (R4) ;0=400
2113 005476 005034 CLR B(R4)+ ;400=0
2114 005500 005004 CLR R4 ;R4=0
2115 005502 000277 SCC
2116 005504 000244 CLZ ;CC=1011
2117 005506 005734 TST B(R4)+ ; TEST MODE 3
2118 005510 103403 BCS 1# ;BRANCH IF ERROR
2119 005512 102402 BVS 1#
2120 005514 100401 BMI 1#
2121 005516 001403 BEQ 2# ;BRANCH IF GOOD
2122 005520
2123 005520 104000 3$: ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
2124 005522 000145 .WORD 145 ;UNIQUE ERROR NUMBER
2125 005524 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
2126 ;MODE 3 FAILED
2127 005526 005304 2$: DEC R4
2128 005530 005304 DEC R4 ;R4=0
2129 005532 005334 DEC B(R4)+ ;400=-1
2130 005534 005004 CLR R4
2131 005536 000277 SCC
2132 005540 000250 CLN ;CC=0111
2133 005542 005734 TST B(R4)+ ; TEST INSTRUCTION
2134 005544 103403 BCS 3#
2135 005546 001402 BEQ 3# ;BRANCH IF ERROR
2136 005550 102401 BVS 3#
2137 005552 100403 BMI 4# ;BRANCH IF GOOD
2138 ;ERROR MODE 3
2139 005554
2140 005554 104000 3$: ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
2141 005556 000146 .WORD 146 ;UNIQUE ERROR NUMBER
2142 005560 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
2143 005562
2144
2145
2146 005562
2147 MSPAA:
2148 *****
2149 TEST 51 TEST TST MODE 3 AUTO-INC
2150 *****
2151 TST51:

```



```

2151 005562 005267 173216      INC      $TESTN      ;INCREMENT TEST NUMBER
2152 005566 005004      CLR      R4
2153 005570 005014      CLR      (R4)        ;0=0
2154 005572 105114      COMB    (R4)        ;
2155 005574 005214      INC      (R4)        ;0=400
2156 005576 005001      CLR      R1
2157 005600 105101      COMB    R1
2158 005602 005201      INC      R1          ;R1=400
2159 005604 005011      CLR      (R1)       ;400=0
2160 005606 000277      SCC
2161 005610 005734      YST      B(R4)+     ;400=0
2162 005612 103403      BCS     1#          ;ERROR IF CARRY
2163 005614 102402      BVS     1#          ;ERROR IF OVERFLOW
2164 005616 100401      BMI     1#          ;ERROR IF MINUS
2165 005620 001403      BEQ     2#          ;ERROR IF NOT EQUAL
2166
2167 005622                1#:
2168 005622 104000      ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
2169 005624 000147      .WORD  147                          ;UNIQUE ERROR NUMBER
2170 005626 001127      .WORD  CPUERR                        ;ADDRESS OF ERROR MESSAGE
2171 005630 005304      2#:  DEC      R4
2172 005632 005304      DEC      R4
2173 005634 005704      TST      R4          ;SEE IF AUTO-INC WORKED
2174 005636 001403      BEQ      4#          ;ERROR IF R4 NE 0
2175
2176 005640                3#:
2177 005640 104000      ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
2178 005642 000150      .WORD  150                          ;UNIQUE ERROR NUMBER
2179 005644 001127      .WORD  CPUERR                        ;ADDRESS OF ERROR MESSAGE
2180 005646                4#:
2181
2182
2183 005646      ;
2184      ;
2185      ;
2186      ;
2187 005646      ;
2188 005646 005267 173132      TSTB2:  INC      $TESTN      ;INCREMENT TEST NUMBER
2189 005652 005004      CLR      R4
2190 005654 005014      CLR      (R4)
2191 005656 105114      COMB    (R4)
2192 005660 005214      INC      (R4)
2193 005662 005214      INC      (R4)        ;0=401
2194 005664 005001      CLR      R1
2195 005666 105101      COMB    R1
2196 005670 005201      INC      R1          ;R1=400
2197 005672 005011      CLR      (R1)
2198 005674 005111      COMB    (R1)
2199 005676 105011      CLRB    (R1)        ;400=377 000
2200 005700 105734      TSTB    B(R4)+     ;** TEST INSTRUCTION
2201 005702 001403      BEQ     1#          ;ERROR IF EQUAL
2202 005704 103402      BCS     1#          ;ERROR IF CARRY SET
2203 005706 102401      BVS     1#          ;ERROR IR OVERFLOW
2204 005710 100403      BMI     2#          ;BRANCH IF MINUS
2205
2206 005712                1#:

```

```

2207 005712 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
2208 005714 000151          .WORD      151          ;UNIQUE ERROR NUMBER
2209 005716 001127          .WORD      CPUERR       ;ADDRESS OF ERROR MESSAGE
2210 005720 005304          2#: DEC      R4
2211 005722 005304          DEC      R4
2212 005724 001403          BEQ      4#             ;BRANCH IF AUTO-INC WORKED
2213                                     ;AUTO-INC FAILED
2214 005726          3#:
2215 005726 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
2216 005730 000152          .WORD      152          ;UNIQUE ERROR NUMBER
2217 005732 001127          .WORD      CPUERR       ;ADDRESS OF ERROR MESSAGE
2218 005734          4#:
2219
2220
2221 005734          ;
2222          ;*****
2223          ;*TEST 53          TEST TST MODE 4
2224          ;*****
2225          TST53:
2226 005734 005267 173044      INC      #TESTN         ;INCREMENT TEST NUMBER
2227 005740 005004          CLR      R4
2228 005742 005014          CLR      (R4)          ,0=0
2229 005744 005204          INC      R4
2230 005746 005204          INC      R4             ;R4=2
2231 005750 000277          SCC
2232 005752 000244          CLZ
2233 005754 005744          YST      -(R4)        ;CC=1011
2234 005756 103403          BCS      1#           ;**TEST INTRUCTION
2235 005760 102402          BVS      1#           ;ERROR IF CARRY
2236 005762 100401          BMI      1#           ;ERROR IF OVERFLOW
2237 005764 001403          BEQ      2#           ;ERROR IF MINUS
2238                                     ;BRANCH IF GOOD
2239                                     ;ERROR! CC WRONG
2240 005766          1#:
2241 005770 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
2242 005772 000153          .WORD      153          ;UNIQUE ERROR NUMBER
2243 005774 001127          .WORD      CPUERR       ;ADDRESS OF ERROR MESSAGE
2244 005776 005704          2#: TST      R4
2245          BEQ      4#             ;INSURE CORRECT AUTO-DEC
2246                                     ;BRANCH IF GOOD AUTO-DEC
2247                                     ;BAD AUTO-DEC
2248 006000          3#:
2249 006000 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
2250 006002 000154          .WORD      154          ;UNIQUE ERROR NUMBER
2251 006004 001127          .WORD      CPUERR       ;ADDRESS OF ERROR MESSAGE
2252
2253 006006          ;
2254          ;*****
2255          ;*TEST 54          TEST TST MODE 4 BYTE
2256          ;*****
2257          TST54:
2258 006006 005267 172772      INC      #TESTN         ;INCREMENT TEST NUMBER
2259 006012 005004          CLR      R4
2260 006014 005014          CLR      (R4)
2261 006016 005114          COM      (R4)
2262 006020 105114          COMB     (R4)          ;0=377 000

```

```

2263 006022 000277          SCC
2264 006024 005204          INC      R4
2265 006026 005204          INC      R4          ;R4=2
2266 006030 105744          TSTB    -(R4)      ;**TEST INSTRUCTION
2267 006032 001403          BEQ     1#          ;ERROR IF EQUAL TO 0
2268 006034 103402          BCS     1#          ;ERROR IF CARRY
2269 006036 102401          BVS     1#          ;ERROR IF OVERFLOW
2270 006040 100403          BMI     2#          ;BRANCH IF MINUS
2271
2272          006042          1#:          ;ERROR! CC SHOULD EQUAL 0100
2273 006042 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
2274 006044 000155          .WORD    155      ;UNIQUE ERROR NUMBER
2275 006046 001127          .WORD    CPUERR   ;ADDRESS OF ERROR MESSAGE
2276 006050 105744          2#:          TSTB    -(R4)      ;**TEST EVEN BYTE
2277 006052 001403          BEQ     4#          ;BRANCH IF GOOD
2278
2279          006054          3#:          ;ERROR! CC SHOULD EQUAL 0100 AND R4=-1
2280 006054 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
2281 006056 000156          .WORD    156      ;UNIQUE ERROR NUMBER
2282 006060 001127          .WORD    CPUERR   ;ADDRESS OF ERROR MESSAGE
2283 006062
2284
2285 006062          4#:
2286          NOTE:
2287          *****
2288          TEST 55      TEST TST MODE 5
2289          *****
2289 006062          TST55:
2290 006062 005267 172716          INC     #TESTN    ;INCREMENT TEST NUMBER
2291 006066 005004          CLR     R4
2292 006070 005024          CLR     (R4)+    ;
2293 006072 000277          SCC
2294 006074 000244          CLZ
2295 006076 005754          TST     0-(R4)   ;
2296 006100 103403          BCS     1#          ;ERROR IF CARRY
2297 006102 102402          BVS     1#          ;ERROR IF OVERFLOW
2298 006104 100401          BMI     1#          ;ERROR IF MINUS
2299 006106 001403          BEQ     2#          ;BRANCH IF GOOD
2300
2301          006110          1#:          ;ERROR! CC WRONG, SHOULD = 0100
2302 006110 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
2303 006112 000157          .WORD    157      ;UNIQUE ERROR NUMBER
2304 006114 001127          .WORD    CPUERR   ;ADDRESS OF ERROR MESSAGE
2305 006116 005704          2#:          TST     R4
2306 006120 001403          BEQ     4#          ;BRANCH IF AUTO-DEC WORKED
2307
2308          006122          3#:          ;ERROR! AUTO-DEC FAILED
2309 006122 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
2310 006124 000160          .WORD    160      ;UNIQUE ERROR NUMBER
2311 006126 001127          .WORD    CPUERR   ;ADDRESS OF ERROR MESSAGE
2312 006130          4#:
2313
2314          TST55:
2315 006130          *****
2316          TEST 56      TEST TST MODE 5 BYTE
2317          *****
2318

```





```

2431 006374 001402      BEQ      1#      ;ERROR IF R4=0
2432 006376 102401      BVS      1#      ;ERROR IF OBERFLOW
2433 006400 100403      BMI      2#      ;BRANCH IF MINUS
2434                                     ;ERROR! CC SHOULD = 100-, R4 SHOULD=-1
2435 006402      1#:
2436 006402 104000      ERROR                                     ;ALL ERRORS TO TRAP TO EMT VECTOR
2437 006404 000167      .WORD   167                               ;UNIQUE ERROR NUMBER
2438 006406 001127      .WORD   CPUERR                           ;ADDRESS OF ERROR MESSAGE
2439 006410 005204      2#:   INC      R4                          ;R4 SHOULD =0
2440 006412 001373      BNE      1#      ;ERROR IF R4 NE 0
2441
2442
2443 006414      ;
2444      ;MDA0:
2445      ;*****
2446      ;*TEST 62      TEST ADD MODE 0
2447      ;*****
2448      ;TST62:
2449      ;      INC      #TESTN                ;INCREMENT TEST NUMBER
2450      ;      CLR      R4                      ;R4=0
2451      ;      CLR      R1
2452      ;      COM      R1                      ;R1=-1
2453      ;      ADD      R1,R4                  ;** TEST ADD OF R1 TO R4
2454      ;      BEQ      1#                      ;ERROR IF ZERO
2455      ;      BCS      1#                      ;ERROR IF CARRY
2456      ;      BVS      1#                      ;ERROR IF OVERFLOW
2457      ;      BMI      2#                      ;BRANCH IF MINUS
2458      ;      ;ERROR! CC SHOULD = 1000 , R4=-1
2459      ;
2460      ;      ERROR                                     ;ALL ERRORS TO TRAP TO EMT VECTOR
2461      ;      .WORD   170                               ;UNIQUE ERROR NUMBER
2462      ;      .WORD   CPUERR                           ;ADDRESS OF ERROR MESSAGE
2463      ;      2#:   INC      R4                          ;R4 SHOULD =0
2464      ;      BEQ      4#                      ;BRANCH IF R4=0
2465      ;      ;ERROR! R4 SHOULD = 0
2466      ;
2467      ;      3#:   ERROR                                     ;ALL ERRORS TO TRAP TO EMT VECTOR
2468      ;      .WORD   171                               ;UNIQUE ERROR NUMBER
2469      ;      .WORD   CPUERR                           ;ADDRESS OF ERROR MESSAGE
2470      ;
2471      ;      4#:
2472      ;      ;MDS0:
2473      ;*****
2474      ;*TEST 63      TEST SUB MODE 0
2475      ;*****
2476      ;TST63:
2477      ;      INC      #TESTN                ;INCREMENT TEST NUMBER
2478      ;      CLR      R4
2479      ;      CLR      R1
2480      ;      INC      R1                      ;R1=1 R4=0
2481      ;      SUB      R1,R4                  ;**TEST OF R4-R1, R4=-1
2482      ;      BVS      1#                      ;ERROR IF V SET
2483      ;      BCC      1#                      ;ERROR IF NO CARRY
2484      ;      BEQ      1#                      ;ERROR IF =0
2485      ;      BMI      2#                      ;BRANCH IF MINUS
2486      ;      ;ERROR! CC SHOULD = 1001
    
```

```

2487 006504          1$:
2488 006504 104000      ERROR
2489 006506 000172      .WORD 172
2490 006510 001127      .WORD CPUERR
2491 006512 005101      2$: COM R1
2492 006514 005201      INC R1
2493 006516 160104      SUB R1,R4
2494 006520 001403      BEQ 4$
2495
2496 006522          3$:
2497 006522 104000      ERROR
2498 006524 000173      .WORD 173
2499 006526 001127      .WORD CPUERR
2500 006530          4$:
2501
2502
2503 006530          ;
2504                      ;*****
2505                      ;*TEST 64      TEST MOV MODE 27,00
2506                      ;*****
2507 006530          TST64:
2508 006530 005267 172250  INC $TESTN          ;INCREMENT TEST NUMBER
2509 006534 000257          CCC          ;CC=0000
2510 006536 012704 125252  MOV #125252,R4      ;**TEST MOVE
2511 006542 001401          BEQ 1$          ;ERROR IF = 0
2512 006544 100403          BMI 2$          ;BRANCH IF MINUS
2513
2514 006546          1$:
2515 006546 104000      ERROR
2516 006550 000174      .WORD 174
2517 006552 001127      .WORD CPUERR
2518 006554 012701 052525 2$: MOV #052525,R1      ;**TEST MOVE
2519 006560 100401          BMI 3$          ;ERROR IF MINUS
2520 006562 001003          BNE 4$          ;BRANCH IF NE 0
2521
2522 006564          3$:
2523 006564 104000      ERROR
2524 006566 000175      .WORD 175
2525 006570 001127      .WORD CPUERR
2526 006572 060104      4$: ADD R1,R4          ;R1+R4=-1
2527 006574 100403          BMI 6$          ;BRANCH IF MINUS
2528
2529 006576          5$:
2530 006576 104000      ERROR
2531 006600 000176      .WORD 176
2532 006602 001127      .WORD CPUERR
2533 006604 005204      6$: INC R4          ;R4+1=0
2534 006606 001373          BNE 5$          ;ERROR IF NOT ZERO
2535
2536
2537 006610          ;
2538                      ;MB100:
2539                      ;*****
2540                      ;*TEST 65      TEST BIC, BIS MODE 0,0
2541                      ;*****
2541 006610          TST65:
2542 006610 005267 172170  INC $TESTN          ;INCREMENT TEST NUMBER

```

```

2543 006614 005004 CLR R4
2544 006616 005104 COM R4 ;R4=-1
2545 006620 012701 125252 MOV #125252,R1 ;SETUP R1 TEST DATA
2546 006624 012702 052525 MOV #052525,R2 ;R2=COMPLIMENT OF R1
2547 006630 000261 SEC
2548 006632 040104 BIC R1,R4 ;**TEST BIC WITH CARRY SET
2549 006634 103003 BCC 1# ;ERROR IF NO CARRY
2550 006636 102402 BVS 1# ;ERROR IF OVERFLOW
2551 006640 001401 BEQ 1# ;ERROR IF 0
2552 006642 100003 BPL 2# ;BRANCH IF PLUS
2553 ;ERROR! CC SHOULD = 0001
2554 006644 1#:
2555 006644 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
2556 006646 000177 .WORD 177 ;UNIQUE ERROR NUMBER
2557 006650 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
2558 006652 020402 2#: CMP R4,R2 ;COMPARE CONTENTS OF R4 AND R2
2559 006654 001403 BEQ 4# ;BRANCH IF EQUAL
2560 ;ERROR! R4 AND R2 SHOULD BE EQUAL
2561 006656 3#:
2562 006656 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
2563 006660 000200 .WORD 200 ;UNIQUE ERROR NUMBER
2564 006662 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
2565 006664 005301 4#: DEC R1 ;R1=125251
2566 006666 050201 BIS R2,R1 ;BIS 052525 AND 125251=177775
2567 006670 100403 BMI 6# ;BRANCH IF MINUS VALUE
2568 ;ERROR! BAD BIS OPERATION
2569 006672 5#:
2570 006672 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
2571 006674 000201 .WORD 201 ;UNIQUE ERROR NUMBER
2572 006676 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
2573 006700 005201 6#: INC R1
2574 006702 005201 INC R1
2575 006704 005201 INC R1 ;R1=0
2576 006706 001371 BNE 5# ;ERROR IF NE 0
2577
2578 ;
2579 006710 ;MBC00:
2580 ;*****
2581 ;*TEST 66 TEST BIT, CMP MODE 0,0
2582 ;*****
2583 TS166:
2584 006710 005267 172070 INC #TESTN ;INCREMENT TEST NUMBER
2585 006714 012701 125252 MOV #125252,R1 ;R1=125252
2586 006720 012704 100000 MOV #100000,R4 ;R4=100000
2587 006724 012702 052525 MOV #052525,R2 ;R2=052525
2588 006730 030401 BIT R4,R1 ;**TEST OF BIT ,CC=1000
2589 006732 001401 BEQ 1# ;ERROR IF EQ 0
2590 006734 100403 BMI 2# ;BRANCH IF GOOD
2591 ;ERROR! CC SHOULD = 1000
2592 006736 1#:
2593 006736 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
2594 006740 000202 .WORD 202 ;UNIQUE ERROR NUMBER
2595 006742 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
2596 006744 020401 2#: CMP R4,R1 ;*TEST 100000-125252=25252
2597 006746 001402 BEQ 3# ;ERROR IF EQUAL 0
2598 006750 103001 BCC 3# ;ERROR IF CARRY CLEARED

```



```

2599 006752 100403          BMI      4#          ;BRANCH IF GOOD
2600                                     ;ERROR! CC SHOULD = 0010
2601 006754                3#:
2602 006754 104000          ERROR
2603 006756 000203          .WORD   203          ;ALL ERRORS TO TRAP TO EMT VECTOR
2604 006760 001127          .WORD   CPUERR       ;UNIQUE ERROR NUMBER
2605 006762 020104          4#:  CMP      R1,R4    ;ADDRESS OF ERROR MESSAGE
2606 006764 001403          BEQ      5#          ;125252-100000 = 25252
2607 006766 103402          BCS      5#          ;ERROR IF EQUAL
2608 006770 102401          BVS      5#          ;ERROR IF CARRY
2609 006772 100003          BPL      6#          ;ERROR IF OVERFLOW
2610                                     ;BRANCH IF GOOD
2611 006774                5#:          ;ERROR! CC SHOULD =0001
2612 006774 104000          ERROR
2613 006776 000204          .WORD   204          ;ALL ERRORS TO TRAP TO EMT VECTOR
2614 007000 001127          .WORD   CPUERR       ;UNIQUE ERROR NUMBER
2615 007002 005004          6#:  CLR      R4          ;ADDRESS OF ERROR MESSAGE
2616 007004 005204          INC      R4          ;R4=1
2617 007006 000277          SCC
2618 007010 030401          BIT      R4,R1
2619 007012 001403          BEQ      8#          ;R4 + R1 = 2
2620                                     ;BRANCH IF GOOD
2621 007014                7#:          ;ERROR! CC SHOULD = 0101
2622 007014 104000          ERROR
2623 007016 000205          .WORD   205          ;ALL ERRORS TO TRAP TO EMT VECTOR
2624 007020 001127          .WORD   CPUERR       ;UNIQUE ERROR NUMBER
2625 007022                8#:          ;ADDRESS OF ERROR MESSAGE
2626
2627
2628 007022                ;
2629                ;MM11:
2630                ;*****
2631                ;*TEST 67      TEST MOV, MOVB MODE 1,1 AND SIGN EXT ON MOVB TO GPR
2632                ;*****
2633 007022 005267 171756          TST67:  INC      #TESTN          ;INCREMENT TEST NUMBER
2634 007026 012704 000400          MOV      #400,R4          ;R4=400
2635 007032 012701 000402          MOV      #402,R1          ;R1=402
2636 007036 005014                CLR      (R4)
2637 007040 005114                COM      (R4)
2638 007042 005011                CLR      (R1)
2639 007044 105111                COMB     (R1)
2640 007046 005002                CLR      R2
2641 007050 012703 000405          MOV      #405,R3          ;402=000 377
2642 007054 000277                SCC
2643 007056 011412                MOV      (R4),(R2)       ;R2=0
2644 007060 001403                BEQ      1#          ;R3=405
2645 007062 102402                BVS      1#          ;CC=1111
2646 007064 103001                BCC      1#          ;MOV 400 TO 0 ,0--1
2647 007066 100403                BMI      2#          ;ERROR IF 0
2648                                     ;ERROR IF OVERFLOW
2649 007070                1#:          ;ERROR IF NO CARRY
2650 007070 104000          ERROR          ;BRANCH IF GOOD
2651 007072 000206          .WORD   206          ;ERROR! CC SHOULD =1001
2652 007074 001127          .WORD   CPUERR       ;ALL ERRORS TO TRAP TO EMT VECTOR
2653 007076 005212          2#:  INC      (R2)       ;UNIQUE ERROR NUMBER
2654 007100 001004          BNE      3#          ;ADDRESS OF ERROR MESSAGE
                                     ;0=0
                                     ;ERROR IF NOT 0

```

```

2655 007102 000257          CCC          ;CC=0000
2656 007104 111113          MOVB      (R1),(R3)      ;405=377
2657 007106 001401          BEQ       3#           ;ERROR IF EQUAL
2658 007110 100403          BMI       4#           ;BRANCH IF GOOD
2659 007112 3#:
2660 007112 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
2661 007114 000207          .WORD      207        ;UNIQUE ERROR NUMBER
2662 007116 001127          .WORD      CPUERR     ;ADDRESS OF ERROR MESSAGE
2663 007120 105213          4#:
2664 007122 001373          INCB      (R3)        ;405=0
2665 007124 005002          BNE       3#           ;ERROR IF 405 NOT 0
;CHECK THAT SIGN EXTENSION OCCURS ON A MOVB TO GENERAL REGISTER.
2666 007124 005002          CLR       R2          ;INIT R2 TO ZERO.
2667 007126 111102          MOVB      (R1),R2     ;MOVE 377 TO R2
2668 007130 100005          BPL       5#           ;ERROR! BIT 15 SHOULD BE SET.
2669 007132 102404          BVS       5#           ;V BIT SHOULD BE CLEARED
2670 007134 103403          BCS       5#           ;CARRY BIT SHOULD BE UNAFFECTED
2671 007136 022702 177777          CMP       #177777,R2 ;TEST R2
2672 007142 001403          BEQ       6#           ;SIGN EXTENDED THROUGH UPPER BYTE
2673
2674
2675 007144 5#:
2676 007144 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
2677 007146 000210          .WORD      210        ;UNIQUE ERROR NUMBER
2678 007150 001127          .WORD      CPUERR     ;ADDRESS OF ERROR MESSAGE
2679 007152 6#:
2680
2681
2682 007152 ;
2683 ;MA11:
2684 ;*****
2685 ;*TEST 70 TEST ADD MODE 1,1
2686 ;*****
2687 007152 005267 171626          TST70:
2688 007156 012704 000400          INC       #TESTN      ;INCREMENT TEST NUMBER
2689 007162 012701 000402          MOV       #400,R4     ;R4=400
2690 007166 012714 177753          MOV       #402,R1     ;R1=402
2691 007172 012711 000024          MOV       #-25,(R4)   ;400=-25
2692 007176 061114          MOV       #24,(R1)    ;402=24
2693 007200 001404          ADD      (R1),(R4)    ;-25+24=-1
2694 007202 103403          BEQ       1#          ;ERROR IF 0
2695 007204 100002          BCS       1#          ;ERROR IF CARRY
2696 007206 005214          BPL       1#          ;ERROR IF POSITIVE RESULT
2697 007210 001403          INC      (R4)        ;-1+1=0
2698 007212 2#:
2699 007212 104000          BEQ       2#          ;BRANCH IF GOOD
2700 007212 104000          ERROR          ;ERROR! CC SHOULD = 1000
2701 007214 000211          .WORD      211        ;ALL ERRORS TO TRAP TO EMT VECTOR
2702 007216 001127          .WORD      CPUERR     ;UNIQUE ERROR NUMBER
2703 007220 2#:
2704
2705
2706 007220 ;
2707 ;MS11:
2708 ;*****
2709 ;*TEST 71 TEST SUB MODE 1,1
2710 ;*****
2710 007220 ;TST71:

```

D5

```

2711 007220 005267 171560      INC      #TESTN      ;INCREMENT TEST NUMBER
2712 007224 012704 000400      MOV      #400,R4      ;R4=400
2713 007230 012701 000404      MOV      #404,R1      ;R1=404
2714 007234 012714 000003      MOV      #3,(R4)      ;400=3
2715 007240 012711 000006      MOV      #6,(R1)      ;406=6
2716 007244 000277              SCC              ;CC=1111
2717 007246 161411              SUB      (R4),(R1)    ;6-3=3
2718 007250 001402              BEQ      1#           ;ERROR IF 0
2719 007252 100401              BMI      1#           ;ERROR IF MINUS
2720 007254 103003              BCC      2#           ;BRANCH IF GOOD
2721                                ;ERROR! CC SHOULD = 0000
2722 007256                                1#:
2723 007256 104000              ERROR              ;ALL ERRORS TO TRAP TO EMT VECTOR
2724 007260 000212              .WORD 212          ;UNIQUE ERROR NUMBER
2725 007262 001127              .WORD CPUERR       ;ADDRESS OF ERROR MESSAGE
2726 007264 161411              2#: SUB (R4),(R1)  ;3-3=0
2727 007266 001373              BNE 1#             ;ERROR IF NOT 0
2728
2729
2730 007270      |
2731      |M8811:
2732      |;*****
2733      |;*TEST 72      TEST BIC, HIS MODE 1,1
2734      |;*****
2735      |TST72:
2735 007270 005267 171510      INC      #TESTN      ;INCREMENT TEST NUMBER
2736 007274 012704 000400      MOV      #400,R4      ;R4=400
2737 007300 012701 000402      MOV      #402,R1      ;R1=402
2738 007304 012714 052525      MOV      #052525,(R4) ;400=052525
2739 007310 012711 125252      MOV      #125252,(R1) ;402=125252
2740 007314 051411              BIS      (R4),(R1)    ;R4 V R1 = -1
2741 007316 001401              BEQ      1#           ;ERROR IF 0
2742 007320 100403              BMI      2#           ;BRANCH IF GOOD
2743                                ;ERROR! CC SHOULD = 1000
2744 007322                                1#:
2745 007322 104000              ERROR              ;ALL ERRORS TO TRAP TO EMT VECTOR
2746 007324 000213              .WORD 213          ;UNIQUE ERROR NUMBER
2747 007326 001127              .WORD CPUERR       ;ADDRESS OF ERROR MESSAGE
2748 007330 005211              2#: INC (R1)        ;402=0
2749 007332 001403              BEQ      4#           ;BRANCH IF GOOD
2750                                ;ERROR! CC SHOULD = 0100
2751 007334                                3#:
2752 007334 104000              ERROR              ;ALL ERRORS TO TRAP TO EMT VECTOR
2753 007336 000214              .WORD 214          ;UNIQUE ERROR NUMBER
2754 007340 001127              .WORD CPUERR       ;ADDRESS OF ERROR MESSAGE
2755 007342 005311              4#: DEC (R1)        ;402=-1
2756 007344 041411              BIC      (R4),(R1)    ;R1=125252
2757 007346 001401              BEQ      5#           ;ERROR IF 0
2758 007350 100403              BMI      6#           ;BRANCH IF GOOD
2759                                ;ERROR! CC SHOULD = 1000
2760 007352                                5#:
2761 007352 104000              ERROR              ;ALL ERRORS TO TRAP TO EMT VECTOR
2762 007354 000215              .WORD 215          ;UNIQUE ERROR NUMBER
2763 007356 001127              .WORD CPUERR       ;ADDRESS OF ERROR MESSAGE
2764 007360 005111              6#: COM (R1)        ;402=052525
2765 007362 041114              BIC      (R1),(R4)    ;400=0
2766 007364 001403              BEQ      8#           ;BRANCH IF GOOD

```





```

2879 007642 000225          .WORD 225          ;UNIQUE ERROR NUMBER
2880 007644 001127          .WORD CPUERR       ;ADDRESS OF ERROR MESSAGE
2881 007646
2882
2883
2884 007646
2885
2886
2887
2888
2889 007646 005267 171132
2890 007652 012704 000400
2891 007656 012701 000402
2892 007662 012702 000404
2893 007666 012714 141401
2894 007672 012711 177405
2895 007676 012722 000070
2896 007702 012722 177777
2897 007706 042421
2898 007710 001401
2899 007712 100003
2900
2901 007714
2902 007714 104000
2903 007716 000226
2904 007720 001127
2905 007722 052421
2906 007724 142421
2907 007726 005301
2908 007730 152421
2909 007732 100403
2910
2911 007734
2912 007734 104000
2913 007736 000227
2914 007740 001127
2915 007742 005214
2916 007744 001403
2917
2918 007746
2919 007746 104000
2920 007750 000230
2921 007752 001127
2922 007754
2923
2924
2925 007754
2926
2927
2928
2929 007754
2930 007754 005267 171024
2931 007760 012704 000400
2932 007764 012701 000402
2933 007770 012714 125252
2934 007774 012721 100001

;
MBC22:
;*****
;*TEST 76      TEST BIC, BICB, BIS, BISB MODE 2,2
;*****
TST76:
      INC      $TESTN          ;INCREMENT TEST NUMBER
      MOV      #400,R4          ;R4=400
      MOV      #402,R1          ;R1=402
      MOV      #404,R2          ;R2=404
      MOV      #141401,(R4)     ;400=303 001
      MOV      #177405,(R1)     ;402=377 005
      MOV      #70,(R2)+        ;404=2070
      MOV      #-1,(R2)+        ;406=-1
      BIC      (R4)+,(R1)+      ;402=074004
      BEQ      1#              ;ERROR IF ZERO
      BPL      2#              ;BRANCH IF GOOD
                               ;CC SHOULD = 1000
1#:
      ERROR
      .WORD    226              ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    CPUERR          ;UNIQUE ERROR NUMBER
      .WORD    2#              ;ADDRESS OF ERROR MESSAGE
2#:
      BIS      (R4)+,(R1)+      ;404=074074
      BICB    (R4)+,(R1)+      ;406=074
      DEC     R1                ;R4=405 R1=406
      BISB    (R4)+,(R1)+      ;406=-1 R4=406 R1=407
      BMI     4#                ;BRANCH IF GOOD
                               ;406 SHOULD=-1
3#:
      ERROR
      .WORD    227              ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    CPUERR          ;UNIQUE ERROR NUMBER
      .WORD    4#              ;ADDRESS OF ERROR MESSAGE
4#:
      INC     (R4)              ;406 SHOULD=0
      BEQ     5#                ;BRANCH IF GOOD
                               ;ERROR! 406 NE 0
5#:
      ERROR
      .WORD    230              ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    CPUERR          ;UNIQUE ERROR NUMBER
      .WORD    6#              ;ADDRESS OF ERROR MESSAGE
6#:
;
MBC22:
;*****
;*TEST 77      TEST BIT, CMP MODE 2,2
;*****
TST77:
      INC      $TESTN          ;INCREMENT TEST NUMBER
      MOV      #400,R4          ;R4=400
      MOV      #402,R1          ;R1=402
      MOV      #125252,(R4)     ;400=125252
      MOV      #100001,(R1)+    ;402=100001

```

```

2935 010000 012711 100002      MOV      #100002,(R1)      ;404=100002
2936 010004 005741              TST      -(R1)           ;R1=402
2937 010006 132421              BITB     (R4)+,(R1)+     ;**ANDED RESULT= 000
2938 010010 100401              BMI     1#              ;ERROR IF MINUS
2939 010012 001403              BEQ     2#              ;BRANCH IF GOOD
2940                                ;ERROR! CC SHOULD = 0100
2941 010014              1#:
2942 010014 104000              ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
2943 010016 000231              .WORD   231                ;UNIQUE ERROR NUMBER
2944 010020 001127              .WORD   CPUERR            ;ADDRESS OF ERROR MESSAGE
2945 010022 132124              2#:  BITB     (R1)+,(R4)+     ;** ANDED RESULT = 200
2946 010024 001401              BEQ     3#              ;ERROR IF EQUAL
2947 010026 100403              BMI     4#              ;BRANCH IF GOOD
2948                                ;ERROR! CC SHOULD= 1000 R4=402 R1=404
2949 010030              3#:
2950 010030 104000              ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
2951 010032 000232              .WORD   232                ;UNIQUE ERROR NUMBER
2952 010034 001127              .WORD   CPUERR            ;ADDRESS OF ERROR MESSAGE
2953 010036 022421              4#:  CMP      (R4)+,(R1)+     ;RESULT =+1
2954 010040 001402              BEQ     5#              ;ERROR IF EQUAL
2955 010042 103001              BCC     5#              ;ERROR IF NO CARRY
2956 010044 100403              BMI     6#              ;BRANCH IF GOOD
2957                                ;ERROR! CC SHOULD = 0000
2958 010046              5#:
2959 010046 104000              ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
2960 010050 000233              .WORD   233                ;UNIQUE ERROR NUMBER
2961 010052 001127              .WORD   CPUERR            ;ADDRESS OF ERROR MESSAGE
2962 010054 005341              6#:  DEC      -(R1)           ;404=100001
2963 010056 005741              TST      -(R1)           ;R4=404 R1=402
2964 010060 022124              CMP      (R1)+,(R4)+     ;RESULT =0
2965 010062 001403              BEQ     8#              ;BRANCH IF GOOD
2966                                ;CC SHOULD = 0100 R1=404 R4=406
2967 010064              7#:
2968 010064 104000              ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
2969 010066 000234              .WORD   234                ;UNIQUE ERROR NUMBER
2970 010070 001127              .WORD   CPUERR            ;ADDRESS OF ERROR MESSAGE
2971 010072              8#:
2972
2973
2974 010072              ;
2975                                ;MS33:
2976                                ;*****
2977                                ;*TEST 100      TEST SUB MODE 3,3
2978                                ;*****
2979 010072 005267 170706      TST100:
2980 010076 005004              INC      #TESTN          ;INCREMENT TEST NUMBER
2981 010100 012701 000002      CLR      R4              ;R4=0
2982 010104 012702 000400      MOV      #2,R1           ;R1=2
2983 010110 012714 000400      MOV      #400,R2        ;R2=400
2984 010114 012711 000402      MOV      #400,(R4)       ;R4=400
2985 010120 012722 000200      MOV      #402,(R1)       ;R1=402
2986 010124 012712 054320      MOV      #200,(R2)+      ;400=200
2987 010130 163431              MOV      #54320,(R2)     ;402=54320
2988 010132 001402              SUB     @R4)+,@R1)+     ;54320 - 200=54120
2989 010134 103401              BEQ     1#              ;ERROR IF ZERO
2990 010136 100003              BCS     1#              ;ERROR IF CARRY
                BPL     2#              ;BRANCH IF GOOD

```

```

2991                                     ;ERROR! CC SHOULD =0001
2992 010140                               1#:
2993 010140 104000                       ERROR
2994 010142 000235                       .WORD 235
2995 010144 001127                       .WORD CPUERR
2996 010146 022712 054120               2#: CMP #54120,(R2)
2997 010152 001403                       BEQ 4#
2998
2999 010154                               3#:
3000 010154 104000                       ERROR
3001 010154 000236                       .WORD 236
3002 010160 001127                       .WORD CPUERR
3003 010162 005067 167612               4#: CLR 0
3004 010166 005067 167610               CLR 2
3005
3006
3007
3008 010172                               ;
3009                                     ;MCB44:
3010                                     ;*****
3011                                     ;*TEST 101 TEST CMP, BIT MODE 4,4
3012                                     ;*****
3013 010172 005067 170606               TST101:
3014 010176 012704 000400               INC #TESTN
3015 010202 012701 000402               MOV #400,R4
3016 010206 012721 125366               MOV #402,R1
3017 010212 012724 173001               MOV #125366,(R1)+
3018 010216 024441                       MOV #173001,(R4)+
3019 010220 103401                       CMP -(R4),-(R1)
3020 010222 100003                       BCS 1#
3021                                     BPL 2#
3022 010224                               1#:
3023 010224 104000                       ERROR
3024 010226 000237                       .WORD 237
3025 010230 001127                       .WORD CPUERR
3026 010232 005204                       2#: INC R4
3027 010234 005201                       INC R1
3028 010236 000261                       SEC
3029 010240 134144                       BITB -(R1),-(R4)
3030 010242 103001                       BCC 3#
3031 010244 001403                       BEQ 4#
3032
3033 010246                               3#:
3034 010246 104000                       ERROR
3035 010250 000240                       .WORD 240
3036 010252 001127                       .WORD CPUERR
3037 010254 005724                       4#: TST (R4)+
3038 010256 005201                       INC R1
3039 010260 124441                       CMPB -(R4),-(R1)
3040 010262 001403                       BEQ 6#
3041
3042 010264                               5#:
3043 010264 104000                       ERROR
3044 010266 000241                       .WORD 241
3045 010270 001127                       .WORD CPUERR
3046 010272                               6#:

```



3047  
3048  
3049 010272  
3050  
3051  
3052  
3053 010272  
3054 010272 005267 170506  
3055 010276 012704 000400  
3056 010302 012724 000001  
3057 010306 012724 177776  
3058 010312 012724 000400  
3059 010316 012714 000402  
3060 010322 012701 000410  
3061 010326 065451  
3062 010330 001402  
3063 010332 100001  
3064 010334 103003  
3065  
3066 010336  
3067 010336 104000  
3068 010340 000242  
3069 010342 001127  
3070 010344 062704 000004  
3071 010350 065154  
3072 010352 001403  
3073  
3074 010354  
3075 010354 104000  
3076 010356 000243  
3077 010360 001127  
3078 010362  
3079  
3080  
3081  
3082  
3083  
3084  
3085  
3086  
3087  
3088  
3089 010362  
3090  
3091  
3092  
3093 010362  
3094 010362 005267 170416  
3095 010366 005004  
3096 010370 012701 000400  
3097 010374 012721 125252  
3098 010400 012721 000001  
3099 010404 012721 100000  
3100 010410 036461 000400 177774  
3101 010416 001403  
3102

```

MA55:
;*****
;TEST 102      TEST ADD MODE 5,5
;*****
TST102:
      INC      $TESTN          ;INCREMENT TEST NUMBER
      MOV      #400,R4
      MOV      #1,(R4)+        ;400=1
      MOV      #-2,(R4)+       ;402=-2
      MOV      #400,(R4)+      ;404=400
      MOV      #402,(R4)       ;406=402 R4=406
      MOV      #410,R1         ;R1=410
      ADD      @-(R4),@-(R1)    ;1+ -2= -1
      BEQ      1$              ;ERROR IF ZERO
      BPL      1$              ;ERROR IF PLUS
      BCC      2$              ;BRANCH IF GOOD
      ;ERROR! BAD ADD

1$:
      ERROR
      .WORD    242              ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    CPUERR          ;UNIQUE ERROR NUMBER
      .WORD    CPUERR          ;ADDRESS OF ERROR MESSAGE
2$:
      ADD      #4,R4           ;R4=410
      ADD      @-(R1),@-(R4)   ;-1 + 1 = 0
      BEQ      4$              ;BRANCH IF GOOD
      ;ERROR! CC SHOULD= 0100 R4=406 R1=402

3$:
      ERROR
      .WORD    243              ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    CPUERR          ;UNIQUE ERROR NUMBER
      .WORD    CPUERR          ;ADDRESS OF ERROR MESSAGE
4$:
;-----
;TEST DOP BIT(B) MODE 6,6
;
;
;
MA66:
;*****
;TEST 103      TEST BIT, BITB MODE 6,6
;*****
TST103:
      INC      $TESTN          ;INCREMENT TEST NUMBER
      CLK      R4              ;R4=0
      MOV      #400,R1
      MOV      @125252,(R1)+   ;400=125252
      MOV      #1,(R1)+       ;402=1
      MOV      #100000,(R1)+   ;404=100000 R1=406
      BIT      @00(R4),-4(R1)  ;(400)^(402)=0
      BEQ      2$              ;BRANCH IF GOOD
      ;CC SHOULD = 0100

```

```

3103 010420          1#:
3104 010420 104000          ERROR
3105 010422 000244          .WORD 244
3106 010424 001127          .WORD CPLERR
3107 010426 136461 000405 177772 2#: BITB 405(R4),-6(R1)
3108 010434 001401          BEQ 3#
3109 010436 106403          BMT 4#
3110
3111 010440          3#:
3112 010440 104000          ERROR
3113 010442 000245          .WORD 245
3114 010444 001127          .WORD CPUERR
3115 010446          4#:
3116
3117
3118 010446          |
3119          MS77:
3120          |*****
3121          |*TEST 104 TEST SUB MODE 7,7
3122          |*****
3123 010446 005267 170332          TST104:
3124 010452 012704 000400          INC $TESTN
3125 010456 005001          MOV #400,R4
3126 010460 012724 177776          CLR R1
3127 010464 012724 177777          MOV #-2,(R4)+
3128 010470 012724 000400          MOV #-1,(R4)+
3129 010474 012711 000402          MOV #400,(R4)+
3130 010500 005201          MOV #402,(R1)
3131 010502 167471 177372 000403          INC R1
3132 010510 001401          SUB #406(R4),#403(R1)
3133 010512 100403          BEQ 1#
3134          BMT 2#
3135 010514          1#:
3136 010514 104000          ERROR
3137 010516 000246          .WORD 246
3138 010520 001127          .WORD CPUERR
3139 010522 167174 177777 177776 2#: SUB #1(R1),#-2(R4)
3140 010530 001403          BEQ 4#
3141
3142 010532          3#:
3143 010532 104000          ERROR
3144 010534 000247          .WORD 247
3145 010536 001127          .WORD CPUERR
3146 010540 005067 167234          CLR 0
3147 010544 005067 167232          CLR 2
3148
3149
3150
3151 010550          |
3152          MRLO:
3153          |*****
3154          |*TEST 105 TEST ROL, ROLB MODE 0
3155          |*****
3156 010550 005267 170330          TST105:
3157 010554 012704 125252          INC $TESTN
3158 010560 000277          MOV #125252,R4
          SCC
          ; INCREMENT TEST NUMBER
          ; R4=125252
          ; CC=1111

```

```

; ALL ERRORS TO TRAP TO EMT VECTOR
; UNIQUE ERROR NUMBER
; ADDRESS OF ERROR MESSAGE
; (405)+(400)=200
; ERROR IF ZERO
; BRANCH IF GOOD
; CC SHOULD = 1000

```

```

; ALL ERRORS TO TRAP TO EMT VECTOR
; UNIQUE ERROR NUMBER
; ADDRESS OF ERROR MESSAGE

```

```

; INCREMENT TEST NUMBER
; 400=-2
; 402=-1
; 404=400 R4=406
; 0=402
; R1=1
; -2 - -1 = -1
; ERROR IF ZERO
; BRANCH IF GOOD
; CC SHOULD=1000

```

```

; ALL ERRORS TO TRAP TO EMT VECTOR
; UNIQUE ERROR NUMBER
; ADDRESS OF ERROR MESSAGE
; -1 - -1 = 0
; BRANCH IF GOOD
; ERROR! ERROR ON SUBTRACT 400=0

```

```

; ALL ERRORS TO TRAP TO EMT VECTOR
; UNIQUE ERROR NUMBER
; ADDRESS OF ERROR MESSAGE
; RESTORE VECTORS
; "

```

```

3159 010562 006104          ROL      R4          ;R4=052525 WITH CARRY SET
3160 010564 102004          BVC     1#          ;ERROR IF V CLEAR
3161 010566 103003          BCC     1#          ;ERROR IF CARRY CLEAR
3162 010570 022704 052525  CMP     #052525,R4 ;SEE IF R0 = EXPECTED
3163 010574 001403          BEQ     2#          ;ERROR IF R4 NE EXPECTED
3164                                ;ERROR! ROL FAILED, CC SHOULD=0011
3165 010576                1#:
3166 010576 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3167 010600 000250          .WORD    250      ;UNIQUE ERROR NUMBER
3168 010602 001127          .WORD    CPUERR   ;ADDRESS OF ERROR MESSAGE
3169 010604 012704 125252  2#:  MOV     #125252,R4 ;R4=125252
3170 010610 000257          CCC          ;CC=0000
3171 010612 106104          ROLB    R4          ;ROTATE EVEN BYTE
3172 010614 103005          BCC     3#          ;ERROR IF NO CARRY
3173 010616 102004          BVC     3#          ;ERROR IF NO OVERFLOW
3174 010620 100403          BMI     3#          ;ERROR IF MINUS
3175 010622 022704 125124  CMP     #125124,R4 ;SEE IF R4 = EXPECTED
3176 010626 001403          BEQ     4#          ;BRANCH IF GOOD
3177                                ;ERROR! ROLB FAILED, CC SHOULD=1011, R4=125125
3178 010630                3#:
3179 010630 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3180 010632 000251          .WORD    251      ;UNIQUE ERROR NUMBER
3181 010634 001127          .WORD    CPUERR   ;ADDRESS OF ERROR MESSAGE
3182 010636                4#:
3183
3184
3185 010636                ;
3186                                ;MRLB1:
3187                                ;*****
3188                                ;*TEST 106      TEST ROL, ROLB MODE 1
3189                                ;*****
3190                                ;TST106:
3190 010636 005267 170142  INC     #TESTN    ;INCREMENT TEST NUMBER
3191 010642 005004          CLR     R4          ;R4=0
3192 010644 012714 052525  MOV     #52525,(R4) ;O=52525
3193 010650 006114          ROL     (R4)        ;**TEST INSTRUCTION, O=125252
3194 010652 100005          BPL     1#          ;ERROR IF PLUS
3195 010654 102004          BVC     1#          ;ERROR IF NO OVERFLOW
3196 010656 103403          BCS     1#          ;ERROR IF CARRY
3197 010650 021427 125252  CMP     (R4),#125252 ;SEE IF R4=EXPECTED
3198 010664 001403          BEQ     2#          ;BRANCH IF GOOD
3199                                ;BAD ROL ,CC SHOULD=1010
3200 010666                1#:
3201 010666 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3202 010670 000252          .WORD    252      ;UNIQUE ERROR NUMBER
3203 010672 001127          .WORD    CPUERR   ;ADDRESS OF ERROR MESSAGE
3204 010674 012714 125252  2#:  MOV     #125252,(R4) ;O=125252
3205 010700 005204          INC     R4          ;R4=1
3206 010702 000277          SCC          ;CC=1111
3207 010704 106114          ROLB    (R4)        ;**TEST INSTRUCTION
3208 010706 100406          BMI     3#          ;ERROR IF RESULT IS POSITIVE
3209 010710 103005          BCC     3#          ;ERROR IF NO CARRY
3210 010712 102004          BVC     3#          ;ERROR IF V CLEAR
3211 010714 005304          DEC     R4          ;R4=0
3212 010716 022714 052652  CMP     #52652,(R4) ;ERROR IF 0 NE EXPECTED
3213 010722 001403          BEQ     4#          ;BRANCH IF GOOD
3214                                ;ERROR! BAD ROLB ODD BYTE,CC SHOULD=1011

```

```

3215 010724
3216 010724 104000
3217 010726 000253
3218 010730 001127
3219 010732
3220
3221
3222 010732
3223
3224
3225
3226 010732
3227 010732 005267 170046
3228 010736 005004
3229 010740 012714 100000
3230 010744 000257
3231 010746 006124
3232 010750 103002
3233 010752 102001
3234 010754 001403
3235
3236 010756
3237 010756 104000
3238 010760 000254
3239 010762 001127
3240 010764 005304
3241 010766 005304
3242 010770 001012
3243 010772 012714 004040
3244 010776 000241
3245 011000 106124
3246 011002 103405
3247 011004 102404
3248 011006 005304
3249 011010 022714 004100
3250 011014 001403
3251
3252 011016
3253 011016 104000
3254 011020 000255
3255 011022 001127
3256 011024
3257
3258
3259 011024
3260
3261
3262
3263 011024
3264 011024 005267 167754
3265 011030 005004
3266 011032 012714 052525
3267 011036 000277
3268 011040 006137 000000
3269 011044 100005
3270 011046 102004

```

```

3$:      ERROR
        .WORD 253
        .WORD CPUERR
; ALL ERRORS TO TRAP TO EMT VECTOR
; UNIQUE ERROR NUMBER
; ADDRESS OF ERROR MESSAGE

4$:

;
; MRL2:
; *****
; *TEST 107      TEST ROL, ROLB MODE 2
; *****
TST107:
        INC      $TESTN      ; INCREMENT TEST NUMBER
        CLR      R4           ; R4=0
        MOV      #100000,(R4) ; 0=100000
        CCC      ; CC=0000
        ROL      (R4)+       ; **TEST INSTRUCTION
        BCC      1$         ; ERROR IF NO CARRY
        BVC      1$         ; ERROR IF NO OVERFLOW
        BEQ      2$         ; BRANCH IF GOOD
; ROL FAILED ,CCSHOULD= 0100

1$:      ERROR
        .WORD 254
        .WORD CPUERR
; ALL ERRORS TO TRAP TO EMT VECTOR
; UNIQUE ERROR NUMBER
; ADDRESS OF ERROR MESSAGE

2$:      DEC      R4
        DEC      R4
        BNE      3$         ; ERROR IN AUTO-DEC
        MOV      #4040,(R4) ; 0=4040
        CLC
        ROLB     (R4)+       ; **TEST INSTRUCTION
        BCS      3$         ; ERROR IF CARRY SET
        BVS      3$         ; ERROR IF V
        DEC      R4
        CMP      #04100,(R4) ; SEE IF 0= EXPECTED RESULT
        BEQ      4$         ; BRANCH IF GOOD
; ERROR! BAD ROL

3$:      ERROR
        .WORD 255
        .WORD CPUERR
; ALL ERRORS TO TRAP TO EMT VECTOR
; UNIQUE ERROR NUMBER
; ADDRESS OF ERROR MESSAGE

4$:

;
; MRL3:
; *****
; *TEST 110      TEST ROL, ROLB MODE 3
; *****
TST110:
        INC      $TESTN      ; INCREMENT TEST NUMBER
        CLR      R4           ; R4=0
        MOV      #052525,(R4) ; 0=52525
        SCC      ; CC=1111
        ROL      #00
        BPL      1$         ; **TEST INSTRUCTION MODE 3 WITH PC
        BVC      1$         ; ERROR IF PLUS
; ERROR IF NO OVERFLOW

```

```

3271 011050 103403          BCS      1#          ;ERROR IF CARRY
3272 011052 022714 125253  CMP      #125253,(R4) ;COMPARE RESULT WITH EXPECTED
3273 011056 001403          BEQ      2#          ;BRANCH IF GOOD
3274                                ;BAD ROL CC SHOULD=1010
3275 011060          1#:
3276 011060 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3277 011062 000256          .WORD      256          ;UNIQUE ERROR NUMBER
3278 011064 001127          .WORD      CPUERR        ;ADDRESS OF ERROR MESSAGE
3279 011066 012714 125252  2#:  MOV      #125252,(R4) ;O=125252
3280 011072 000261          SEC          ;CC=---1
3281 011074 106137 000000  ROLB     #00          ;**TEST INSTRUCTION
3282 011100 100402          BMI      3#          ;ERROR IF MINUS
3283 011102 103001          BCC      3#          ;ERROR IF NO CARRY
3284 011104 102403          BVS      4#          ;BRANCH IF OVERFLOW
3285                                ;ERROR! BAD ROL, CC SHOULD=1011
3286 011106          3#:
3287 011106 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3288 011110 000257          .WORD      257          ;UNIQUE ERROR NUMBER
3289 011112 001127          .WORD      CPUERR        ;ADDRESS OF ERROR MESSAGE
3290 011114          4#:
3291                                ;
3292                                ;
3293 011114          ;MRL4:
3294                                ;*****
3295                                ;*TEST 111      TEST ROL MODE 4
3296                                ;*****
3297                                ;
3298 011114 005267 167664  TST111:  INC      #TESTN          ;INCREMENT TEST NUMBER
3299 011120 005001          CLR      R1              ;R1=0
3300 011122 012704 000002  MOV      #2,R4           ;R4=2
3301 011126 012711 054321  MOV      #54321,(R1)    ;O=54321
3302 011132 000277          SCC          ;CC=1111
3303 011134 006144          ROL      -(R4)          ;**TEST INSTRUCTION
3304 011136 100007          BPL      1#          ;ERROR IF PLUS
3305 011140 102006          BVC      1#          ;ERROR IF NO OVERFLOW
3306 011142 103105          BCS      1#          ;ERROR IF CARRY
3307 011144 022711 130643  CMP      #130643,(R1)   ;SEE IF EXPECTED RESULT
3308 011150 001002          BNE      1#          ;BRANCH IF ROL FAILED
3309 011152 005704          TST      R4             ;SEE IF AUTO-DEC WORKED
3310 011154 001403          BEQ      2#          ;BRANCH IF GOOD
3311                                ;ERROR! BAD ROL INST
3312 011156          1#:
3313 011156 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3314 011160 000260          .WORD      260          ;UNIQUE ERROR NUMBER
3315 011162 001127          .WORD      CPUERR        ;ADDRESS OF ERROR MESSAGE
3316 011164          2#:
3317                                ;
3318                                ;
3319 011164          ;MRL5:
3320                                ;*****
3321                                ;*TEST 112      TEST ROL MODE 5
3322                                ;*****
3323 011164          ;
3324 011164 005267 167614  TST112:  INC      #TESTN          ;INCREMENT TEST NUMBER
3325 011170 005004          CLR      R4              ;R4=0
3326 011172 012714 000400  MOV      #100,(R4)     ;O=400

```

```

3327 011176 012734 123456      MOV      #123456,B(R4)      ;400=123465, R4=2
3328 011202 000277              SCC                      ;CC=1111
3329 011204 006154              ROL      B-(R4)           ;**TEST INSTRUCTION
3330 011206 100410              BMI      1#              ;ERROR IF RESULT IS MINUS
3331 011210 103007              BCC      1#              ;ERROR IF NO CARRY
3332 011212 102006              BVC      1#              ;ERROR IF NO OVERFLOW
3333 011214 005704              TST      R4              ;SEE IF AUTO-DEC WORKED
3334 011216 001004              BNE      1#              ;ERROR OF AUTO-DEC
3335 011220 022737 047135 000400  CMP      #47135,B#400    ;SEE IF CORRECT RESULT
3336 011226 001403              BEQ      2#              ;BRANCH IF GOOD
3337                                ;BAD ROL MODE 5
3338 011230                                1#:
3339 011230 104000              ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
3340 011232 000261              .WORD   261                        ;UNIQUE ERROR NUMBER
3341 011234 001127              .WORD   CPUERR                     ;ADDRESS OF ERROR MESSAGE
3342 011236                                2#:
3343
3344
3345 011236                                ;
3346                                MRL6:
3347                                ;*****
3348                                ;*TEST 113      TEST ROL MODE 6
3349                                ;*****
3350 011236 005267 167542      TST113:
3351 011242 012704 000400      INC      #TESTN              ;INCREMENT TEST NUMBER
3352 011246 005001              MOV      #400,R4            ;R4=400
3353 011250 012711 032525      CLR      R1                 ;R1=0
3354 011254 000277              MOV      #32525,(R1)        ;O=32525
3355 011256 006164 177400      SCC                      ;**TEST INSTRUCTION
3356 011262 100405              ROL      -400(R4)          ;ERROR IF MINUS
3357 011264 103404              BMI      1#              ;ERROR IF CARRY
3358 011266 102403              BCS      1#              ;ERROR IF OVERFLOW
3359 011270 022711 065253      BVS      1#              ;SEE IF CORRECT RESULT
3360 011274 001403              CMP      #65253,(R1)        ;BRANCH IF GOOD
3361                                BEQ      2#              ;BAD ROL MODE 6
3362 011276                                1#:
3363 011276 104000              ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
3364 011300 000262              .WORD   262                        ;UNIQUE ERROR NUMBER
3365 011302 001127              .WORD   CPUERR                     ;ADDRESS OF ERROR MESSAGE
3366 011304                                2#:
3367
3368
3369 011304                                ;
3370                                MRL7:
3371                                ;*****
3372                                ;*TEST 114      TEST ROL MODE 7
3373                                ;*****
3374 011304 005267 167474      TST114:
3375 011310 012704 000400      INC      #TESTN              ;INCREMENT TEST NUMBER
3376 011314 005037 000402      MOV      #400,R4            ;R4=400
3377 011320 012737 100000 000000  CLR      #402              ;402=0
3378 011326 006174 000002      MOV      #100000,B#0        ;O=100000
3379 011332 100406              ROL      B2(R4)           ;**TEST INSTRUCTION
3380 011334 001005              BMI      1#              ;ERROR IF MINUS
3381 011336 103004              BNE      1#              ;ERROR IF NOT ZERO
3382 011340 102003              BCC      1#              ;ERROR IF NO CARRY
3382 011340 102003              BVC      1#              ;ERROR IF NO OVERFLOW

```

```

3383 011342 005737 000000          TST      B#0          ;CHECK RESULT
3384 011346 001403                    BEQ      2#          ;BRANCH IF GOOD
3385                                     ;BAD ROL MODE 7
3386 011350                    1#:
3387 011350 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3388 011352 000263          .WORD      263          ;UNIQUE ERROR NUMBER
3389 011354 001127          .WORD      CPUERR       ;ADDRESS OF ERROR MESSAGE
3390 011356                    2#:
3391
3392
3393 011356                    |
3394                                     |MSW37:
3395                                     |*****
3396                                     |*TEST 115      TEST SWAB MODE 37
3397                                     |*****
3398 011356                    TST115:
3399 011362 005267 167422          INC      $TESTN          ;INCREMENT TEST NUMBER
3400 011366 012714 040700          MOV      #400,R4         ;R4=400
3401 011372 000337 000400          MOV      #40700,(R4)    ;400= 101 300
3402 011376 100406                    SWAB     B#400          ;400 SHOULD = 300 101
3403 011400 022714 140101          BMI     1#              ;ERROR IF MINUS
3404 011404 001003                    CMP      #140101,(R4)   ;SEE IF EXPECTED RESULT
3405 011406 000337 000400          BNE     1#              ;BRANCH IF BAD
3406 011412 100403                    SWAB     B#400          ;400=101 300
3407                                     ;BRANCH IF GOOD
3408                                     ;ERROR! BAD SWAB MODE 37
3409 011414                    1#:
3410 011416 000264          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3411 011420 001127          .WORD      264          ;UNIQUE ERROR NUMBER
3412 011422          .WORD      CPUERR       ;ADDRESS OF ERROR MESSAGE
3413
3414
3415 011422                    |
3416                                     |MRR0:
3417                                     |*****
3418                                     |*TEST 116      TEST ROR MODE 0
3419                                     |*****
3420 011422 005267 167356                    TST116:
3421 011426 012704 052525          INC      $TESTN          ;INCREMENT TEST NUMBER
3422 011432 000257                    MOV      #52525,R4      ;R4=52525
3423 011434 006004                    CCC          ;CC=0000
3424 011436 103003                    ROR      R4              ;R4 SHOULD = 25252 WITH CARRY
3425 011440 022704 025252          BCC     1#              ;ERROR IF NO CARRY
3426 011444 001403                    CMP      #25252,R4     ;SEE IF R4= EXPECTED
3427                                     ;BRANCH IF GOOD
3428                                     ;ROR MODE 0 FAILED
3429 011446                    1#:
3430 011450 000265          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3431 011452 001127          .WORD      265          ;UNIQUE ERROR NUMBER
3432 011454          .WORD      CPUERR       ;ADDRESS OF ERROR MESSAGE
3433
3434
3435 011454                    |
3436                                     |MRRB1:
3437                                     |*****
3438                                     |*TEST 117      TEST RORB MODE 1
3439                                     |*****

```

```

3439 011454          TST117:
3440 011454 005267 167324      INC      #TESTN      ;INCREMENT TEST NUMBER
3441 011460 005004          CLR      R4          ;R4=0
3442 011462 012714 000001      MOV      #1,(R4)    ;O=1
3443 011466 000277          SCC          ;CC=1111
3444 011470 106014          RORB     (R4)        ;O=000200, NO C
3445 011472 103004          BCC     1#          ;ERROR IF NO CARRY
3446 011474 100003          BPL     1#          ;ERROR IF PLUS
3447 011476 022714 000200      CMP      #200,(R4)  ;CHECK RESULT
3448 011502 001403          BEQ     2#          ;BRANCH IF GOOD
3449
3450 011504          1#:
3451 011504 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3452 011506 000266          .WORD 266      ;UNIQUE ERROR NUMBER
3453 011510 001127          .WORD CPUERR   ;ADDRESS OF ERROR MESSAGE
3454 011512
3455
3456
3457
3458 011512          MJ:
3459
3460
3461
3462 011512          ;*****
3463 011512 005267 167266          ;*TEST 120      TEST JMP - ALL MODES
3464 011516 012737 000001 001066      ;*****
3465 011524 012701 011600          TST120:
3466 011530 000111          INC      #TESTN      ;INCREMENT TEST NUMBER
3467 011532 023727 001066 000002      MOV      #1,#SEQ    ;SETUP TEST SEQUENCER
3468 011540 001403          MOV      #MJU1,R1   ;SET MODE 1 JUMP ADDRESS
3469
3470 011542          MJU2:
3471 011542 104000          JMP      (R1)        ;*JMP MODE 1
3472 011544 000267          CMP      #SEQ,#2    ;CHECK FOR CORRECT SEQUENCE
3473 011546 001127          BEQ     MJU2A       ;BRANCH IF GOOD
3474
3475 011550 020127 011534          1#:
3476 011554 001403          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3477
3478 011556          .WORD 267      ;UNIQUE ERROR NUMBER
3479 011556 104000          .WORD CPUERR   ;ADDRESS OF ERROR MESSAGE
3480 011560 000270          MJU2A:
3481 011562 001127          CMP      R1,#MJU2+2 ;CHECK FOR AUTO-INCREMENT
3482
3483 011564 005237 001066          BEQ     MJU2B       ;BRANCH IF GOOD
3484 011570 012701 011576          2#:
3485 011574 000131          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3486 011576 011630          .WORD 270      ;UNIQUE ERROR NUMBER
3487 011600 023727 001066 000001      .WORD CPUERR   ;ADDRESS OF ERROR MESSAGE
3488 011606 001403          MJU2B:
3489
3490 011610          INC      #SEQ        ;UPDATE TEST SEQUENCER
3491 011610 104000          MOV      #MJ2,R1   ;SETUP MODE 3 JUMP
3492 011612 000271          JMP      B(R1)+     ;JUMP MODE 3
3493 011614 001127          MJ2:
3494
3494          .WORD MJU3      ;MODE 3 DESTINATION
          CMP      #SEQ,#1 ;TEST FOR CORRECT SEQUENCE
          BEQ     MJU1A   ;BRANCH IF GOOD
          ;ERROR! JMP OUT OF SEQUENCE
          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
          .WORD 271      ;UNIQUE ERROR NUMBER
          .WORD CPUERR   ;ADDRESS OF ERROR MESSAGE

```



GLOBAL AREAS MACY11 30A(1052) 15-MAR-84 13:28 PAGE 69  
 KDJ11A.MAC 22-FEB-84 15:12 T120 TEST JMP - ALL MODES

SEQ 0069

3495	011616	005237	001066		MJU1A:	INC	0#SEQ		;UPDATE SEQUENCE
3496	011622	012701	011532			MOV	0MJU2,R1		;SETUP MODE 2 DESTINATION
3497	011626	000121				JMP	(R1)+		;JUMP MODE 2
3498	011630	023727	001066	000003	MJU3:	CMP	0#SEQ,03		;TEST FOR CORRECT SEQUENCE
3499	011636	001403				BEQ	MJU3A		;BRANCH IF GOOD
3500									;ERROR! JMP OUT OF SEQUENCE
3501	011640				4#:				
3502	011640	104000				ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
3503	011642	000272				.WORD	272		;UNIQUE ERROR NUMBER
3504	011644	001127				.WORD	CPUERR		;ADDRESS OF ERROR MESSAGE
3505									
3506	011646	022701	011600		MJU3A:	CMP	0MJ2+2,R1		;TEST AUTO-INCREMENT
3507	011652	001403				BEQ	MJU3B		;BRANCH IF GOOD
3508									;ERROR! AUTO-INCREMENT FAILED MODE 3
3509	011654				5#:				
3510	011654	104000				ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
3511	011656	000273				.WORD	273		;UNIQUE ERROR NUMBER
3512	011660	001127				.WORD	CPUERR		;ADDRESS OF ERROR MESSAGE
3513									
3514	011662	005237	001066		MJU3B:	INC	0#SEQ		;UPDATE SEQUENCER
3515	011666	012701	011746			MOV	0MJU4+2,R1		;SETUP DESTINATION MODE 4
3516	011672	000141				JMP	-(R1)		;EXECUTE JUMP MODE 4
3517	011674	000000			MJU5:	HALT			
3518	011676	022701	012012			CMP	0MJ5,R1		;CHECK AUTO-DECREMENT
3519	011702	001403				BEQ	MJU5A		;BRANCH IF GOOD AUTO-DEC
3520									;ERROR! AUTO-DEC FAILED MODE 5
3521	011704				6#:				
3522	011704	104000				ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
3523	011706	000274				.WORD	274		;UNIQUE ERROR NUMBER
3524	011710	001127				.WORD	CPUERR		;ADDRESS OF ERROR MESSAGE
3525									
3526	011712	023727	001066	000005	MJU5A:	CMP	0#SEQ,05		;TEST CORRECT SEQUENCE
3527	011720	001403				BEQ	MJU5B		;BRANCH IF GOOD SEQUENCE
3528									;ERROR! JMP OUT OF SEQUENCE
3529	011722				7#:				
3530	011722	104000				ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
3531	011724	000275				.WORD	275		;UNIQUE ERROR NUMBER
3532	011726	001127				.WORD	CPUERR		;ADDRESS OF ERROR MESSAGE
3533									
3534	011730	005237	001066		MJU5B:	INC	0#SEQ		;UPDATE SEQUENCE COUNT
3535	011734	012701	012007			MOV	0MJU6-5,R1		;SETUP DESTINATION MODE6
3536	011740	000161	000005			JMP	+5(R1)		;JUMP MODE 6
3537									
3538	011744	000240			8#:				
3539	011746	022701	011744		MJU4:	NOP			;TEST AUTO-DECR
3540	011752	001403				CMP	0MJU4,R1		;BRANCH IF GOOD
3541						BEQ	MJU4A		;ERROR! MODE 4 AUTO-DEC FAILED
3542	011754								
3543	011754	104000				ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
3544	011756	000276				.WORD	276		;UNIQUE ERROR NUMBER
3545	011760	001127				.WORD	CPUERR		;ADDRESS OF ERROR MESSAGE
3546									
3547	011762	023727	001066	000004	MJU4A:	CMP	0#SEQ,04		;TEST FOR CORRECT SEQUENCE
3548	011770	001403				BEQ	MJU4B		;BRANCH IF CORRECT SEQUENCE
3549									;ERROR! INCORRECT JMP SEQUENCE
3550	011772				9#:				

```

3551 011772 104000          ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
3552 011774 000277          .WORD        277          ; UNIQUE ERROR NUMBER
3553 011776 001127          .WORD        CPUERR      ; ADDRESS OF ERROR MESSAGE
3554
3555 012000 005237 001066    MJU4B: INC      @#SEQ      ; UPDATE SEQUENCE
3556 012004 012701 012014    MOV      @MJ5+2,R1      ; SETUP MODE 5 POINTER
3557 012010 000151          JMP      @-(R1)         ; EXECUTE MODE 5 JMP
3558
3559 012012 011676          ; MJ5:  .WORD        MJU5+2 ; POINTER MODE 5
3560
3561 012014 022737 000006 001066    MJU6:  CMP      @6,@#SEQ  ; CHECK FOR CORRECT SEQUENCE
3562 012022 001403          BEQ      MJU6A          ; BRANCH IF GOOD
3563
3564 012024          10#:
3565 012024 104000          ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
3566 012026 000300          .WORD        300        ; UNIQUE ERROR NUMBER
3567 012030 001127          .WORD        CPUERR      ; ADDRESS OF ERROR MESSAGE
3568
3569 012032 005237 001066    MJU6A: INC      @#SEQ      ; UPDATE SEQUENCER
3570 012036 012701 012056    MOV      @MJ7+10,R1     ; SETUP INDEX
3571 012042 000171 177770    JMP      @-10(R1)      ; EXECUTE MODE 7 JUMP
3572 012046 012052          MJ7:  .WORD        MJU7   ; POINTER FOR MODE 7
3573 012050 000000          HALT
3574 012052 022737 000007 001066    MJU7:  CMP      @7,@#SEQ  ; TEST FOR CORRECT SEQUENCE
3575 012060 001403          BEQ      MJU7E          ; BRANCH IF GOOD SEQUENCE
3576
3577 012062          11#:
3578 012062 104000          ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
3579 012064 000301          .WORD        301        ; UNIQUE ERROR NUMBER
3580 012066 001127          .WORD        CPUERR      ; ADDRESS OF ERROR MESSAGE
3581
3582 012070          MJU7E:
3583
3584
3585
3586
3587
3588
3589 012070 005267 166710    TST121: INC      $TESTN      ; INCREMENT TEST NUMBER
3590 012074 012701 012404    MOV      @128,R1       ; SET UP R1 WITH ADDRESS OF ERROR
3591
3592 012100 012717          MOV      (PC)+,(PC)    ; ROUTINE
3593 012102 000240          .WORD        NOP        ; WRITE THE NOP OVER THE JMP INSTRUCTION
3594 012104 000111          .WORD        111       ; NOP INSTRUCTION
3595 012106 012717          64#:  MOV      (PC)+,(PC)  ; WRITE THE NOP OVER THE JMP INSTRUCTION
3596 012110 000240          .WORD        NOP        ; NOP INSTRUCTION
3597 012112 000111          .WORD        111       ; JMP (R1)
3598 012114 012717          MOV      (PC)+,(PC)    ; WRITE THE NOP OVER THE JMP INSTRUCTION
3599 012116 000240          .WORD        NOP        ; NOP INSTRUCTION
3600 012120 000111          .WORD        111       ; JMP (R1)
3601 012122 012717          MOV      (PC)+,(PC)    ; WRITE THE NOP OVER THE JMP INSTRUCTION
3602 012124 000240          .WORD        NOP        ; NOP INSTRUCTION
3603 012126 000111          .WORD        111       ; JMP (R1)
3604 012130 012717          MOV      (PC)+,(PC)    ; WRITE THE NOP OVER THE JMP INSTRUCTION
3605 012132 000240          .WORD        NOP        ; NOP INSTRUCTION
3606 012134 000111          .WORD        111       ; JMP (R1)

```

3607	012136	012717	MOV	(PC)+,(PC)	;WRITE THE NOP OVER THE JMP INSTRUCTION
3608	012140	000240	.WORD	NOP	;NOP INSTRUCTION
3609	012142	000111	69#:	.WORD	111
3610	012144	012717	MOV	(PC)+,(PC)	;WRITE THE NOP OVER THE JMP INSTRUCTION
3611	012146	000240	.WORD	NOP	;NOP INSTRUCTION
3612	012150	000111	70#:	.WORD	111
3613	012152	012717	MOV	(PC)+,(PC)	;WRITE THE NOP OVER THE JMP INSTRUCTION
3614	012154	000240	.WORD	NOP	;NOP INSTRUCTION
3615	012156	000111	71#:	.WORD	111
3616	012160	012717	MOV	(PC)+,(PC)	;WRITE THE NOP OVER THE JMP INSTRUCTION
3617	012162	000240	.WORD	NOP	;NOP INSTRUCTION
3618	012164	000111	72#:	.WORD	111
3619	012166	012717	MOV	(PC)+,(PC)	;WRITE THE NOP OVER THE JMP INSTRUCTION
3620	012170	000240	.WORD	NOP	;NOP INSTRUCTION
3621	012172	000111	73#:	.WORD	111
3622	012174	012717	MOV	(PC)+,(PC)	;WRITE THE NOP OVER THE JMP INSTRUCTION
3623	012176	000240	.WORD	NOP	;NOP INSTRUCTION
3624	012200	000111	74#:	.WORD	111
3625	012202	012717	MOV	(PC)+,(PC)	;WRITE THE NOP OVER THE JMP INSTRUCTION
3626	012204	000240	.WORD	NOP	;NOP INSTRUCTION
3627	012206	000111	75#:	.WORD	111
3628	012210	012717	MOV	(PC)+,(PC)	;WRITE THE NOP OVER THE JMP INSTRUCTION
3629	012212	000240	.WORD	NOP	;NOP INSTRUCTION
3630	012214	000111	76#:	.WORD	111
3631	012216	012717	MOV	(PC)+,(PC)	;WRITE THE NOP OVER THE JMP INSTRUCTION
3632	012220	000240	.WORD	NOP	;NOP INSTRUCTION
3633	012222	000111	77#:	.WORD	111
3634	012224	012717	MOV	(PC)+,(PC)	;WRITE THE NOP OVER THE JMP INSTRUCTION
3635	012226	000240	.WORD	NOP	;NOP INSTRUCTION
3636	012230	000111	78#:	.WORD	111
3637	012232	012717	MOV	(PC)+,(PC)	;WRITE THE NOP OVER THE JMP INSTRUCTION
3638	012234	000240	.WORD	NOP	;NOP INSTRUCTION
3639	012236	000111	79#:	.WORD	111
3640	012240	012717	MOV	(PC)+,(PC)	;WRITE THE NOP OVER THE JMP INSTRUCTION
3641	012242	000240	.WORD	NOP	;NOP INSTRUCTION
3642	012244	000111	80#:	.WORD	111
3643	012246	012717	MOV	(PC)+,(PC)	;WRITE THE NOP OVER THE JMP INSTRUCTION
3644	012250	000240	.WORD	NOP	;NOP INSTRUCTION
3645	012252	000111	81#:	.WORD	111
3646	012254	012717	MOV	(PC)+,(PC)	;WRITE THE NOP OVER THE JMP INSTRUCTION
3647	012256	000240	.WORD	NOP	;NOP INSTRUCTION
3648	012260	000111	82#:	.WORD	111
3649	012262	012717	MOV	(PC)+,(PC)	;WRITE THE NOP OVER THE JMP INSTRUCTION
3650	012264	000240	.WORD	NOP	;NOP INSTRUCTION
3651	012266	000111	83#:	.WORD	111
3652	012270	012717	MOV	(PC)+,(PC)	;WRITE THE NOP OVER THE JMP INSTRUCTION
3653	012272	000240	.WORD	NOP	;NOP INSTRUCTION
3654	012274	000111	84#:	.WORD	111
3655	012276	012717	MOV	(PC)+,(PC)	;WRITE THE NOP OVER THE JMP INSTRUCTION
3656	012300	000240	.WORD	NOP	;NOP INSTRUCTION
3657	012302	000111	85#:	.WORD	111
3658	012304	012717	MOV	(PC)+,(PC)	;WRITE THE NOP OVER THE JMP INSTRUCTION
3659	012306	000240	.WORD	NOP	;NOP INSTRUCTION
3660	012310	000111	86#:	.WORD	111
3661	012312	012717	MOV	(PC)+,(PC)	;WRITE THE NOP OVER THE JMP INSTRUCTION
3662	012314	000240	.WORD	NOP	;NOP INSTRUCTION

```

3663 012316 000111      87#: .WORD 111          ; JMP (R1)
3664 012320 012717      MOV (PC)+,(PC)      ; WRITE THE NOP OVER THE JMP INSTRUCTION
3665 012322 000240      .WORD NOP          ; NOP INSTRUCTION
3666 012324 000111      88#: .WORD 111          ; JMP (R1)
3667 012326 012717      MOV (PC)+,(PC)      ; WRITE THE NOP OVER THE JMP INSTRUCTION
3668 012330 000240      .WORD NOP          ; NOP INSTRUCTION
3669 012332 000111      89#: .WORD 111          ; JMP (R1)
3670 012334 012717      MOV (PC)+,(PC)      ; WRITE THE NOP OVER THE JMP INSTRUCTION
3671 012336 000240      .WORD NOP          ; NOP INSTRUCTION
3672 012340 000111      90#: .WORD 111          ; JMP (R1)
3673 012342 012717      MOV (PC)+,(PC)      ; WRITE THE NOP OVER THE JMP INSTRUCTION
3674 012344 000240      .WORD NOP          ; NOP INSTRUCTION
3675 012346 000111      91#: .WORD 111          ; JMP (R1)
3676 012350 012717      MOV (PC)+,(PC)      ; WRITE THE NOP OVER THE JMP INSTRUCTION
3677 012352 000240      .WORD NOP          ; NOP INSTRUCTION
3678 012354 000111      92#: .WORD 111          ; JMP (R1)
3679 012356 012717      MOV (PC)+,(PC)      ; WRITE THE NOP OVER THE JMP INSTRUCTION
3680 012360 000240      .WORD NOP          ; NOP INSTRUCTION
3681 012362 000111      93#: .WORD 111          ; JMP (R1)
3682 012364 012717      MOV (PC)+,(PC)      ; WRITE THE NOP OVER THE JMP INSTRUCTION
3683 012366 000240      .WORD NOP          ; NOP INSTRUCTION
3684 012370 000111      94#: .WORD 111          ; JMP (R1)
3685 012372 012717      MOV (PC)+,(PC)      ; WRITE THE NOP OVER THE JMP INSTRUCTION
3686 012374 000240      .WORD NOP          ; NOP INSTRUCTION
3687 012376 000111      95#: .WORD 111          ; JMP (R1)
3688 012400 000137 012412  JMP B#129#          ; JMP OVER ERROR CALL
3689 012404      128#:              ; ERROR! PRE-FETCH BUFFER WAS NOT
3690      ; OVER WRITTEN
3691 012404 104000      ERROR              ; ALL ERRORS TO TRAP TO EMT VECTOR
3692 012406 000302      .WORD 302          ; UNIQUE ERROR NUMBER
3693 012410 001127      .WORD CPUERR       ; ADDRESS OF ERROR MESSAGE
3694      ;
3695      ;
3696      ; NOW RESTORE THE OVER WRITTEN JMP INSTRUCTIONS FOR THE NEXT PASS.
3697 012412 012702 000040 129#: MOV #32,R2          ; SET UP R2 AS COUNTER
3698 012416 012703 012104  MOV #64,R3          ; SET UP R3 AS POINTER
3699 012422 012713 000111 130#: MOV #111,(R3)      ; RESTORE OVER WRITTEN JUMPS
3700 012426 062703 000006  ADD #6,R3           ; POINT TO NEXT OVER WRITTEN ADDR.
3701 012432 077205  SOB R2,130#         ; DO UNTIL R2=0
3702      ;
3703      ;
3704 012434      HJP:
3705      ; *****
3706      ; *TEST 122 TEST JMP MODES 17,27,37,67,77
3707      ; *****
3708 012434      TST122:
3709 012434 005267 166344  INC #TESTN          ; INCREMENT TEST NUMBER
3710 012440 012737 000000 001066  MOV #0,B#SEQ        ; SETUP TEST SEQUENCER
3711 012446 000117      JMP (R7)           ; JMP MODE 17(SHOULD BE IN-LINE)
3712      ;
3713 012450 005737 001066  HJP17: TST B#SEQ     ; CHECK SEQUENCE
3714 012454 001403      BEQ 2#            ; BRANCH IF GOOD
3715      ; ERROR! BAD JUMP
3716 012456      1#:
3717 012456 104000      ERROR              ; ALL ERRORS TO TRAP TO EMT VECTOR
3718 012460 000303      .WORD 303          ; UNIQUE ERROR NUMBER
    
```



```

3775 ;
3776 ;
3777 012624 MJSR:
3778 ;*****
3779 ;*TEST 123 TEST JSR ALL MODES
3780 ;*****
3781 012624 TST123:
3782 012624 005267 166154 INC #TESTN ; INCREMENT TEST NUMBER
3783 012630 010637 001070 MOV R6,#SPS ; SAVE STACK POINTER LOCATION
3784 012634 010637 001072 MOV R6,#SPSJ ;
3785 012640 162737 000002 001072 SUB #2,#SPSJ ; SPSJ = R6 AFTER DECRIMENT
3786 012646 012737 000001 001066 MOV #1,#SEQ ; SETUP SEQUENCE COUNTER
3787 012654 012701 012760 MOV #MJSR1,R1 ; SETUP INITIAL JUMP IN MODE 1
3788 012660 005004 CLR R4 ;
3789 012662 005104 COM R4 ; R4=-1 TO BE SAVED ON STACK
3790 012664 004411 JSR R4,(R1) ; JSR MODE 1
3791 ;
3792 012664 022737 000002 001066 MJSR2: CMP #2,#SEQ ; TEST FOR CORRECT SEQUENCE
3793 012674 001403 BEQ MJSR2A ; BRANCH IF GOOD
3794 ;
3795 012676 5#: ; ERROR! MODE 2 JUMPED TO OUT OF SEQUENCE
3796 012676 104000 ; ALL ERRORS TO TRAP TO EMT VECTOR
3797 012700 000310 .WORD 310 ; UNIQUE ERROR NUMBER
3798 012702 001127 .WORD CPUERR ; ADDRESS OF ERROR MESSAGE
3799 ;
3800 012704 023706 001072 MJSR2A: CMP #SPSJ,R6 ; VERIFY STACK DECRIMENT
3801 012710 001006 BNE 6# ; BRANCH IF STACK INCORRECT
3802 012712 021627 125252 CMP (R6),#125252 ; VERIFY CONTENTS OF STACK
3803 012716 001003 BNE 6# ; BRANCH IF DATA ON STACK INCORRECT
3804 012720 022704 013050 CMP #MJSR4,R4 ; SEE IF CORRECT RETURN ADDRESS
3805 012724 001403 BEQ MJSR2B ; BRANCH IF GOOD
3806 ;
3807 012726 6#: ; ERROR! JSR MODE 2 FAILED
3808 012726 104000 ; ALL ERRORS TO TRAP TO EMT VECTOR
3809 012730 000311 .WORD 311 ; UNIQUE ERROR NUMBER
3810 012732 001127 .WORD CPUERR ; ADDRESS OF ERROR MESSAGE
3811 ;
3812 012734 005237 001066 MJSR2B: INC #SEQ ; UPDATE SEQUENCE COUNTER
3813 012740 013706 001070 MOV #SPS,R6 ; RELOAD STACK POINTER
3814 012744 012701 012756 MOV #MJSRA,R1 ; SETUP JSR MODE 3
3815 012750 005004 CLR R4 ; DIFFERENT DATA TO R4
3816 012752 004431 JSR R4,#(R1) ; JSR MODE 3
3817 012754 000000 HALT ;
3818 012756 013144 MJSRA: .WORD MJSR3 ; LITERAL FOR JUMP MODE 3
3819 ;
3820 012760 022737 000001 001066 MJSR1: CMP #1,#SEQ ; TEST FOR CORRECT SEQUENCE
3821 012766 001403 BEQ MJSR1A ; BRANCH IF GOOD
3822 ;
3823 012770 7#: ; ERROR! MODE 1 JUMPED TO OUT OF SEQUENCE
3824 012770 104000 ; ALL ERRORS TO TRAP TO EMT VECTOR
3825 012772 000312 .WORD 312 ; UNIQUE ERROR NUMBER
3826 012774 001127 .WORD CPUERR ; ADDRESS OF ERROR MESSAGE
3827 ;
3828 012776 023706 001072 MJSR1A: CMP #SPSJ,R6 ; VERIFY STACK DECRIMENT
3829 013002 001006 BNE 7# ; BRANCH IF STACK INCORRECT
3830 013004 021627 177777 CMP (R6),#-1 ; VERIFY CONTENT OF STACK

```

3831	013010	001003				BNE	8#			BRANCH IF DATA ON STACK INCORRECT
3832	013012	022704	012666			CMP	#MJSR2,R4			SEE IF CORRECT RETURN ADDRESS
3833	013016	001403				BEQ	MJSR1B			BRANCH IF GOOD
3834										ERROR! JSR MODE 2 FAILED
3835	013020				8#:					
3836	013020	104000				ERROR				ALL ERRORS TO TRAP TO EMT VECTOR
3837	013022	000313				.WORD	313			UNIQUE ERROR NUMBER
3838	013024	001127				.WORD	CPUERR			ADDRESS OF ERROR MESSAGE
3839										
3840	013026	005237	001066			MJSR1B:	INC	0#SEQ		UPDATE SEQUENCE COUNTER
3841	013032	013706	001070				MOV	0#SPS,R6		RELOAD STACK POINTER
3842	013036	012704	125252				MOV	#125252,R4		SETUP R4 DATA
3843	013042	012701	012666				MOV	#MJSR2,R1		SETUP MODE 2 JUMP ADDRESS
3844	013046	004421					JSR	R4,(R1)+		*JUMP MODE 2
3845										
3846										
3847	013050	022737	000004	001066		MJSR4:	CMP	#4,0#SEQ		TEST FOR CORRECT SEQUENCE
3848	013056	001403					BEQ	MJSR4A		BRANCH IF GOOD
3849										ERROR! MODE 4 JUMPED TO OUT OF SEQUENCE
3850	013060				9#:					
3851	013060	104000				ERROR				ALL ERRORS TO TRAP TO EMT VECTOR
3852	013062	000314				.WORD	314			UNIQUE ERROR NUMBER
3853	013064	001127				.WORD	CPUERR			ADDRESS OF ERROR MESSAGE
3854										
3855	013066	023706	001072			MJSR4A:	CMP	0#SPSJ,R6		VERIFY STACK DECRIMENT
3856	013072	001006					BNE	10#		BRANCH IF STACK INCORRECT
3857	013074	021627	052525				CMP	(R6),#052525		VERIFY CONTENTS OF STACK
3858	013100	001003					BNE	10#		BRANCH IF DATA ON STACK INCORRECT
3859	013102	022704	013236				CMP	#MJSR6,R4		SEE IF CORRECT RETURN ADDRESS
3860	013106	001403					BEQ	MJSR4B		BRANCH IF GOOD
3861										ERROR! JSR MODE 4 FAILED
3862	013110				10#:					
3863	013110	104000				ERROR				ALL ERRORS TO TRAP TO EMT VECTOR
3864	013112	000315				.WORD	315			UNIQUE ERROR NUMBER
3865	013114	001127				.WORD	CPUERR			ADDRESS OF ERROR MESSAGE
3866										
3867	013116	005237	001066			MJSR4B:	INC	0#SEQ		UPDATE SEQUENCE COUNTER
3868	013122	013706	001070				MOV	0#SPS,R6		RELOAD STACK POINTER
3869	013126	012704	000377				MOV	#377,R4		SETUP R4 DATA
3870	013132	012701	013144				MOV	#MJSRB+2,R1		SETUP JSR VECTOR
3871	013136	004451					JSR	R4,R-(R1)		JSR MODE 5
3872	013140	000000					HALT			
3873	013142	013332				MJSRB:	.WORD	MJSR5		MODE 5 VECTOR
3874										
3875										
3876	013144	022737	000003	001066		MJSR3:	CMP	#3,0#SEQ		TEST FOR CORRECT SEQUENCE
3877	013152	001403					BEQ	MJSR3A		BRANCH IF GOOD
3878										ERROR! MODE 3 JUMPED TO OUT OF SEQUENCE
3879	013154				11#:					
3880	013154	104000				ERROR				ALL ERRORS TO TRAP TO EMT VECTOR
3881	013156	000316				.WORD	316			UNIQUE ERROR NUMBER
3882	013160	001127				.WORD	CPUERR			ADDRESS OF ERROR MESSAGE
3883										
3884	013162	023706	001072			MJSR3A:	CMP	0#SPSJ,R6		VERIFY STACK DECRIMENT
3885	013166	001006					BNE	12#		BRANCH IF STACK INCORRECT
3886	013170	021627	000000				CMP	(R6),#0		VERIFY CONTENTS OF STACK

```

3887 013174 001003          BNE 12#          ;BRANCH IF DATA ON STACK INCORRECT
3888 013176 022704 012754  CMP  #MJSRA-2,R4 ;SEE IF CORRECT RETURN ADDRESS
3889 013202 001403          BEQ  MJSR3B      ;BRANCH IF GOOD
3890                                     ;ERROR! JSR MODE 3 FAILED
3891 013204          12#:
3892 013204 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3893 013206 000317          .WORD 317      ;UNIQUE ERROR NUMBER
3894 013210 001127          .WORD CPUERR   ;ADDRESS OF ERROR MESSAGE
3895
3896 013212 005237 001066  MJSR3B: INC  #0SEQ ;UPDATE SEQUENCE COUNTER
3897 013216 013706 001070  MOV  #0SPS,R6 ;RELOAD STACK POINTER
3898 013222 012704 052525  MOV  #052525,R4 ;SETUP R4 DATA
3899 013226 012701 013052  MOV  #MJSR4+2,R1 ;SETUP JSR VECTOR
3900 013232 000257          CCC          ;CLEAR CONDITION CODES
3901 013234 004441          JSR  R4,-(R1)   ;JSR MODE 4
3902
3903
3904 013236 022737 000006 001066  MJSR6: CMP  #6,#0SEQ ; TEST FOR CORRECT SEQUENCE
3905 013244 001403          BEQ  MJSR6A     ;BRANCH IF GOOD
3906                                     ;ERROR! MODE 6 JUMPED TO OUT OF SEQUENCE
3907 013246          13#:
3908 013246 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3909 013250 000320          .WORD 320      ;UNIQUE ERROR NUMBER
3910 013252 001127          .WORD CPUERR   ;ADDRESS OF ERROR MESSAGE
3911
3912 013254 023706 001072  MJSR6A: CMP  #0SPSJ,R6 ;VERIFY STACK DECREMENT
3913 013260 001006          BNE  14#       ;BRANCH IF STACK INCORRECT
3914 013262 021627 123456  CMP  (R6),#123456 ;VERIFY CONTENTS OF STACK
3915 013266 001003          BNE  14#       ;BRANCH IF DATA ON STACK INCORRECT
3916 013270 022704 013424  CMP  #MJSR7,R4 ;SEE IF CORRECT RETURN ADDRESS
3917 013274 001403          BEQ  MJSR6B     ;BRANCH IF GOOD
3918                                     ;ERROR! JSR MODE 6 FAILED
3919 013276          14#:
3920 013276 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3921 013300 000321          .WORD 321      ;UNIQUE ERROR NUMBER
3922 013302 001127          .WORD CPUERR   ;ADDRESS OF ERROR MESSAGE
3923
3924 013304 005237 001066  MJSR6B: INC  #0SEQ ;UPDATE SEQUENCE COUNTER
3925 013310 013706 001070  MOV  #0SPS,R6 ;RELOAD STACK POINTER
3926 013314 012704 177773  MOV  #-5,R4    ;SETUP R4 DATA
3927 013320 012701 013340  MOV  #MJSR6+10,R1 ;SETUP JSR VECTOR
3928 013324 004471 177770  JSR  R4,#-10(R1) ;JSR MODE 7
3929 013330 013424  MJSR6: .WORD MJSR7 ;JSR VECTOR
3930
3931
3932 013332 022737 000005 001066  MJSR5: CMP  #5,#0SEQ ; TEST FOR CORRECT SEQUENCE
3933 013340 001403          BEQ  MJSR5A     ;BRANCH IF GOOD
3934                                     ;ERROR! MODE 5 JUMPED TO OUT OF SEQUENCE
3935 013342          15#:
3936 013342 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
3937 013344 000322          .WORD 322      ;UNIQUE ERROR NUMBER
3938 013346 001127          .WORD CPUERR   ;ADDRESS OF ERROR MESSAGE
3939
3940 013350 023706 001072  MJSR5A: CMP  #0SPSJ,R6 ;VERIFY STACK DECREMENT
3941 013354 001006          BNE  16#       ;BRANCH IF STACK INCORRECT
3942 013356 021627 000377  CMP  (R6),#377 ;VERIFY CONTENTS OF STACK

```



```

3943 013362 001003          BNE      16$
3944 013364 022704 013140  CMP      @MJSRB-2,R4
3945 013370 001403          BEQ      MJSR5B
3946
3947 013372          16$:
3948 013372 104000          ERROR
3949 013374 000323          .WORD   323
3950 013376 001127          .WORD   CPUERR
3951
3952 013400 005237 001066  MJSR5B: INC      @#SEQ
3953 013404 013706 001070  MOV      @#SPS,R6
3954 013410 012704 123456  MOV      @123456,R4
3955 013414 012701 013246  MOV      @MJSR6+10,R1
3956 013420 004461 177770  JSR      R4,-10(R1)
3957
3958
3959 013424 022737 000007 001066  MJSR7:  CMP      #7,@#SEQ
3960 013432 001403          BEQ      MJSR7A
3961
3962 013434          17$:
3963 013434 104000          ERROR
3964 013436 000324          .WORD   324
3965 013440 001127          .WORD   CPUERR
3966
3967 013442 023706 001072  MJSR7A: CMP      @#SPSJ,R6
3968 013446 001006          BNE      18$
3969 013450 021627 177773  CMP      (R6),@-5
3970 013454 001003          BNE      18$
3971 013456 022704 013330  CMP      @MJSR5-2,R4
3972 013462 001403          BEQ      MJSR7E
3973
3974 013464          18$:
3975 013464 104000          ERROR
3976 013466 000325          .WORD   325
3977 013470 001127          .WORD   CPUERR
3978
3979 013472          MJSR7E:
3980 013472 013706 001070  MOV      @#SPS,R6
3981
3982
3983 013476          MJRA:
3984
3985
3986
3987 013476          ;*****
3988 013476 005267 165302  TST124: INC      #TESTN
3989 013502 012737 000001 001066  MOV      @1,@#SEQ
3990 013510 010637 001070  MOV      R6,@#SPS
3991 013514 010637 001072  MOV      R6,@#SPSJ
3992 013520 162737 000002 001072  SUB      @2,@#SPSJ
3993 013526 012704 177777  MOV      @-1,R4
3994 013532 004427 000240  JSR      R4,@240
3995
3996 013536 022737 000001 001066  MJR27:  CMP      @1,@#SEQ
3997 013544 001011          BNE      1$
3998 013546 023706 001072  CMP      @#SPSJ,R6

```

```

;BRANCH IF DATA ON STACK INCORRECT
;SEE IF CORRECT RETURN ADDRESS
;BRANCH IF GOOD
;ERROR! JSR MODE 5 FAILED
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;UPDATE SEQUENCE COUNTER
;RELOAD STACK POINTER
;SETUP DATA IN R4
;SETUP JSR VECTOR
;JUMP MODE 6
; TEST FOR CORRECT SEQUENCE
;BRANCH IF GOOD
;ERROR! MODE 7 JUMPED TO OUT OF SEQUENCE
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;VERIFY STACK DECREMENT
;BRANCH IF STACK INCORRECT
;VERIFY CONTENTS OF STACK
;BRANCH IF DATA ON STACK INCORRECT
;SEE IF CORRECT RETURN ADDRESS
;BRANCH IF GOOD
;ERROR! JSR MODE 7 FAILED
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;REPLACE STACK
;*****
;*TEST 124 TEST JSR MODES 27, 37, 67, 77
;*****
;INCREMENT TEST NUMBER
;SETUP SEQUENCER
;SAVE STACK ADDRESS
;SAVE STACK DECREMENT ADDRESS
;
;SETUP R4 DATA
;EXECUTE A JSR MODE 27
;VERIFY CORRECT TEST SEQUENCE
;INCORRECT TEST SEQUENCE
;VERIFY STACK POINTER

```

GLOBAL AREAS MACY11 30A(1052) 15-MAR-84 13:28 PAGE 78  
 KDJ11A.MAC 22-FEB-84 15:12 T124 TEST JSR MODES 27, 37, 67, 77

SEQ 0078

3999	013552	001006			BNE	1*		
4000	013554	021627	177777		CMP	(R6),*-1		;VERIFY R4 GOT LOADED ON THE STACK
4001	013560	001003			BNE	1*		;BRANCH IF INCORRECT STACK CONTENTS
4002	013562	020427	013536		CMP	R4,#MJR27		;VERIFY CORRECT RETURN ADDRESS
4003	013566	001403			BEQ	MJR27A		;BRANCH IF GOOD RETURN ADDRESS ON STACK
4004								;ERROR! MODE 27 FAILED
4005	013570			1*:				
4006	013570	104000			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
4007	013572	000326			.WORD	326		;UNIQUE ERROR NUMBER
4008	013574	001127			.WORD	CPUERR		;ADDRESS OF ERROR MESSAGE
4009								
4010	013576	005237	001066		MJR27A: INC	@SEQ		;UPDATE SEQUENCER
4011	013602	012704	152525		MOV	@152525,R4		;SETUP R4 TEST DATA
4012	013606	013706	001070		MOV	@SPS,R6		;RESET STACK POINTER
4013	013612	004437	013704		JSR	R4,#MJR37		;JSR MODE 37
4014	013616	000000			MJR27B: HALT			
4015								
4016	013620	023727	001066	000003	MJR67: CMP	@SEQ,#3		;VERIFY TEST SEQUENCE
4017	013626	001011			BNE	2*		;INCORRECT TEST SEQUENCE
4018	013630	023706	001072		CMP	@SPSJ,R6		;VERIFY STACK DECREMENT
4019	013634	001006			BNE	2*		;INCORRECT STACK DECREMENT
4020	013636	021627	000125		CMP	(R6),*125		;VERIFY STACK WAS LOADED
4021	013642	001003			BNE	2*		
4022	013644	020427	013764		CMP	R4,#MJR77		;VERIFY RETURN ADDRESS
4023	013650	001403			BEQ	MJR67A		;BRANCH IF GOOD
4024								;ERROR! MODE 67 FAILED
4025	013652							
4026	013652	104000			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
4027	013654	000327			.WORD	327		;UNIQUE ERROR NUMBER
4028	013656	001127			.WORD	CPUERR		;ADDRESS OF ERROR MESSAGE
4029								
4030	013660	005237	001066		MJR67A: INC	@SEQ		;UPDATE SEQUENCER
4031	013664	013706	001070		MOV	@SPS,R6		;RESET STACK
4032	013670	012704	000001		MOV	*1,R4		;SETUP R4 DATA
4033	013674	004477	000002		JSR	R4,#MJR6B		;JSR MODE 77
4034	013700	000000			MJR6A: HALT			
4035	013702	013764			MJR6B: .WORD	MJR77		;DATA FOR MODE 77 JUMP
4036								
4037	013704	023727	001066	000002	MJR37: CMP	@SEQ,#2		;VERIFY TEST SEQUENCE
4038	013712	001011			BNE	2*		;INCORRECT TEST SEQUENCE
4039	013714	023706	001072		CMP	@SPSJ,R6		;VERIFY STACK DECREMENT
4040	013720	001006			BNE	2*		;INCORRECT STACK DECREMENT
4041	013722	021627	152525		CMP	(R6),*152525		;VERIFY STACK WAS LOADED
4042	013726	001003			BNE	2*		
4043	013730	020427	013616		CMP	R4,#MJR27B		;VERIFY RETURN ADDRESS
4044	013734	001403			BEQ	MJR37A		;BRANCH IF GOOD
4045								;ERROR! MODE 37 FAILED
4046	013736							
4047	013736	104000			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
4048	013740	000330			.WORD	330		;UNIQUE ERROR NUMBER
4049	013742	001127			.WORD	CPUERR		;ADDRESS OF ERROR MESSAGE
4050								
4051	013744	005237	001066		MJR37A: INC	@SEQ		;UPDATE SEQUENCER
4052	013750	013706	001070		MOV	@SPS,R6		;RELOAD STACK
4053	013754	012704	000125		MOV	*125,R4		;SETUP R4 TEST DATA
4054	013760	004467	177634		JSR	R4,MJR67		;JSR MODE 6

```

4055
4056 013764 023727 001066 000004 MJR77:  CMP      @#SEQ,#4      ;VERIFY TEST SEQUENCE
4057 013772 001011                BNE      21          ;INCORRECT TEST SEQUENCE
4058 013774 023706 001072                CMP      @#SPSJ,R6   ;VERIFY STACK DECREMENT
4059 014000 001006                BNE      21          ;INCORRECT STACK DECREMENT
4060 014002 021627 000001                CMP      (R6),#1     ;VERIFY STACK WAS LOADED
4061 014006 001003                BNE      21
4062 014010 020427 013700                CMP      R4,#MJR6A   ;VERIFY RETURN ADDRESS
4063 014014 001403                BEQ      MJR77A      ;BRANCH IF GOOD
4064
4065 014016                21:
4066 014016 104000                ERROR
4067 014020 000331                .WORD   331          ;ALL ERRORS TO TRAP TO EMT VECTOR
4068 014022 001127                .WORD   CPUERR       ;UNIQUE ERROR NUMBER
4069
4070 014024                MJR77A:
4071
4072 014024 013706 001070                MOV      @#SPS,R6    ;RESET STACK
4073
4074
4075 014030                |
4076                |MRTS:
4077                |*****
4078                |*TEST 125      TEST RTS AND RTS R6
4079                |*****
4080 014030 005267 164750                TST125:
4081 014034 012706 001000                INC      @TESTN      ;INCREMENT TEST NUMBER
4082 014040 012746 123456                MOV      @STBOT,R6   ;INSURE VALID STACK
4083 014044 012703 014060                MOV      @123456,-(R6) ;SETUP TEST REGISTER
4084 014050 000203                MOV      @RTS1,R3    ;SETUP TEST PC
4085 014052 104000                RTS      R3          ;**TEST INSTRUCTION
4086 014054 000332                ERROR              ;ALL ERRORS TO TRAP TO EMT VECTOR
4087 014056 001127                .WORD   332          ;UNIQUE ERROR NUMBER
4088                .WORD   CPUERR       ;ADDRESS OF ERROR MESSAGE
4089                ;INCORRECT PC ON RTS
4090 014060 022703 123456                RTS1:  CMP      @123456,R3
4091 014064 001403                BEQ      RTS6        ;BRANCH IF GOOD
4092                ;ERROR! REGISTER CONTENTS INCORRECT
4093 014066 104000                ERROR              ;ALL ERRORS TO TRAP TO EMT VECTOR
4094 014070 000333                .WORD   333          ;UNIQUE ERROR NUMBER
4095 014072 001127                .WORD   CPUERR       ;ADDRESS OF ERROR MESSAGE
4096
4097
4098                |
4099                |THIS TEST CHECKS AN UN-TESTED PLA TERM
4100 014074 010601                |
4101 014076 012705 014114                |RTS6:  MOV      R6,R1      ;SAVE STACK IN R1
4102 014102 010506                |      MOV      @1,R5   ;MOVE EXPECTED RETURN ADDR TO R5
4103 014104 000206                |      MOV      R5,R6   ;MOVE RETURN ADDR TO R6
4104                |      RTS      R6      ;>>>>TEST INSTRUCTION<<<<<
4105 014106 104000                |      ERROR              ;ERROR! RTS NOT EXECUTED
4106 014110 000334                |      ERROR              ;ALL ERRORS TO TRAP TO EMT VECTOR
4107 014112 001127                |      .WORD   334          ;UNIQUE ERROR NUMBER
4108 014114 021506                |      .WORD   CPUERR       ;ADDRESS OF ERROR MESSAGE
4109 014116 001403                |11:   CMP      (R5),R6   ;IS R6=31506?
4110                |      BEQ      RTSE      ;IF IT IS THEN GO TO END OF TEST
                |      ;ERROR! WRONG ADDR IN R6

```

```

4111 014120 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
4112 014122 000335          .WORD          335          ;UNIQUE ERROR NUMBER
4113 014124 001127          .WORD          CPUERR        ;ADDRESS OF ERROR MESSAGE
4114 014126 010106          RTSE: MOV          R1,R6        ;RESTORE STACK
4115
4116 014130          TSMUO:
4117          ;*****
4118          ;*TEST 126      SETUP AND TEST KERNEL, SUPERVISOR AND USER STACKS
4119          ;*****
4120 014130          TST126:
4121 014130 005267 164650          INC          $TESTN          ;INCREMENT TEST NUMBER
4122 014134 012737 040000 177776          MOV          #40000,0#177776 ;SET PS TO SUP MODE
4123 014142 012706 177777          MOV          #177777,R6      ;INIT SUP STACK TO ALL ONES
4124 014146 022706 177777          CMP          #177777,R6      ;ARE ALL BITS SET
4125 014152 001403          BEQ          1#              ;YES GO ON
4126          ;NO GO TO ERROR
4127 014154 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
4128 014156 000336          .WORD          336          ;UNIQUE ERROR NUMBER
4129 014160 001127          .WORD          CPUERR        ;ADDRESS OF ERROR MESSAGE
4130
4131 014162 005006          1#: CLR          R6          ;SET SUP STACK TO ALL ZEROES
4132 014164 022706 000000          CMP          #0,R6          ;ARE ALL BITS CLEARED
4133 014170 001403          BEQ          2#              ;YES GO ON
4134          ;NO GO TO ERROR
4135 014172 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
4136 014174 000337          .WORD          337          ;UNIQUE ERROR NUMBER
4137 014176 001127          .WORD          CPUERR        ;ADDRESS OF ERROR MESSAGE
4138
4139 014200 012706 125252          2#: MOV          #125252,R6     ;SET SUP STACK TO ALTERNATING PATTERN
4140 014204 022706 125252          CMP          #125252,R6     ;IS SUP SP CORRECT
4141 014210 001403          BEQ          3#              ;YES GO ON
4142          ;NO GO TO ERROR
4143 014212 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
4144 014214 000340          .WORD          340          ;UNIQUE ERROR NUMBER
4145 014216 001127          .WORD          CPUERR        ;ADDRESS OF ERROR MESSAGE
4146
4147 014220 012706 052525          3#: MOV          #52525,R6     ;SET SUP STACK TO ALTERNATING PATTERN
4148 014224 022706 052525          CMP          #52525,R6     ;IS SUP SP CORRECT
4149 014230 001403          BEQ          4#              ;YES GO ON
4150          ;NO GO TO ERROR
4151 014232 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
4152 014234 000341          .WORD          341          ;UNIQUE ERROR NUMBER
4153 014236 001127          .WORD          CPUERR        ;ADDRESS OF ERROR MESSAGE
4154
4155 014240 012706 000700          4#: MOV          #700,R6      ;SETUP SUP SP
4156 014244 012737 140000 177776          MOV          #140000,0#177776 ;SET PS TO USER MODE
4157 014252 012706 177777          MOV          #177777,R6      ;INIT USER STACK TO ALL ONES
4158 014256 022706 177777          CMP          #177777,R6      ;ARE ALL BITS SET
4159 014262 001403          BEQ          5#              ;YES GO ON
4160          ;NO GO TO ERROR
4161 014264 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
4162 014266 000342          .WORD          342          ;UNIQUE ERROR NUMBER
4163 014270 001127          .WORD          CPUERR        ;ADDRESS OF ERROR MESSAGE
4164
4165 014272 005006          5#: CLR          R6          ;SET USER STACK TO ALL ZEROES
4166 014274 022706 000000          CMP          #0,R6          ;ARE ALL BITS CLEARED

```

4167	014300	001403				BEQ	68		YES GO ON
4168									NO GO TO ERROR
4169	014302	104000				ERROR			ALL ERRORS TO TRAP TO EMT VECTOR
4170	014304	000343				.WORD	343		UNIQUE ERROR NUMBER
4171	014306	001127				.WORD	CPUERR		ADDRESS OF ERROR MESSAGE
4172									
4173	014310	012706	125252		68:	MOV	#125252,R6		SET USER STACK TO ALTERNATING PATTERN
4174	014314	022706	125252			CMP	#125252,R6		IS USER SP CORRECT
4175	014320	001403				BEQ	78		YES GO ON
4176									NO GO TO ERROR
4177	014322	104000				ERROR			ALL ERRORS TO TRAP TO EMT VECTOR
4178	014324	000344				.WORD	344		UNIQUE ERROR NUMBER
4179	014326	001127				.WORD	CPUERR		ADDRESS OF ERROR MESSAGE
4180									
4181	014330	012706	052525		78:	MOV	#52525,R6		SET USER STACK TO ALTERNATING PATTERN
4182	014334	022706	052525			CMP	#52525,R6		IS USER SP CORRECT
4183	014340	001403				BEQ	88		YES GO ON
4184									NO GO TO ERROR
4185	014342	104000				ERROR			ALL ERRORS TO TRAP TO EMT VECTOR
4186	014344	000345				.WORD	345		UNIQUE ERROR NUMBER
4187	014346	001127				.WORD	CPUERR		ADDRESS OF ERROR MESSAGE
4188									
4189	014350	012706	000600		88:	MOV	#600,R6		SETUP USER SP
4190	014354	005037	177776			CLR	#177776		SET PS TO KER MODE
4191	014360	012706	001000			MOV	#STBOT,R6		SETUP KERNEL SP
4192	014364	005037	000700			CLR	#700		CLEAR FIRST WORDS OF SUP, KER, AND USE STACKS
4193	014370	005037	000600			CLR	#600		
4194	014374	005037	001000			CLR	#STBOT		
4195	014400	004767	000070			JSR	PC,CHECK		
4196	014404	012737	040000	177776	RET1:	MOV	#40000,#177776		TEST KER, SUP, AND USE STACKS
4197	014412	022706	000700			CMP	#700,R6		SET PSW TO SUP MODE
4198	014416	001403				BEQ	18		IS SUP SP CORRECT
4199									YES GO ON
4200	014420	104000							NO GO TO ERROR
4201	014422	000346				ERROR			ALL ERRORS TO TRAP TO EMT VECTOR
4202	014424	001127				.WORD	346		UNIQUE ERROR NUMBER
4203						.WORD	CPUERR		ADDRESS OF ERROR MESSAGE
4204	014426	012737	140000	177776	18:	MOV	#140000,#177776		SET PSW TO USE MODE
4205	014434	022706	000500			CMP	#600,R6		IS USE SP CORRECT
4206	014440	001403				BEQ	28		YES GO ON
4207									NO GO TO ERROR
4208	014442	104000				ERROR			ALL ERRORS TO TRAP TO EMT VECTOR
4209	014444	000347				.WORD	347		UNIQUE ERROR NUMBER
4210	014446	001127				.WORD	CPUERR		ADDRESS OF ERROR MESSAGE
4211									
4212	014450	005037	177776		28:	CLR	#177776		SET PSW TO KER MODE
4213	014454	022706	001000			CMP	#STBOT,R6		IS KER SP CORRECT
4214	014460	001403				BEQ	38		YES GO ON
4215									NO GO TO ERROR
4216	014462	104000				ERROR			ALL ERRORS TO TRAP TO EMT VECTOR
4217	014464	000350				.WORD	350		UNIQUE ERROR NUMBER
4218	014466	001127				.WORD	CPUERR		ADDRESS OF ERROR MESSAGE
4219									
4220	014470				38:				
4221									
4222	014470	000167	000252			JMP	MTS0		

```

4223
4224
4225
4226 014474 012737 040000 177776 CHECK: MOV #40000,B#177776 ;SET PSW TO SUP MODE
4227 014502 004767 000060 JSR PC,CHECK1 ; TEST SUP, KER, AND USE STACKS
4228 014506 022716 000000 RET2: CMP #0,(SP) ;IS SUP STACK CLEARED
4229 014512 001403 BEQ 1# ;YES GO ON
4230 ;NO GO TO ERROR
4231 014514 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
4232 014516 000351 .WORD 351 ;UNIQUE ERROR NUMBER
4233 014520 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
4234
4235 014522 012737 140000 177776 1#: MOV #140000,B#177776 ;SET PSW TO USE MODE
4236 014530 022716 000000 CMP #0,(SP) ;IS USE STACK CLEARED
4237 014534 001403 BEQ 2# ;YES GO ON
4238 ;NO GO TO ERROR
4239 014536 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
4240 014540 000352 .WORD 352 ;UNIQUE ERROR NUMBER
4241 014542 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
4242
4243 014544 005037 177776 2#: CLR B#177776 ;SET PSW TO KER MODE
4244 014550 022716 014404 CMP #RET1,(SP) ;DOES KER STACK HAVE CORRECT DATA
4245 014554 001403 BEQ 3# ;YES GO ON
4246 ;NO GO TO ERROR
4247 014556 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
4248 014560 000353 .WORD 353 ;UNIQUE ERROR NUMBER
4249 014562 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
4250
4251 014564 000207 3#: RTS PC ;RETURN
4252
4253 ;ROUTINE TO CHECK STACKS AFTER ONE RTS
4254
4255 014566 012737 140000 177776 CHECK1: MOV #140000,B#177776 ;SET PSW TO USE MODE
4256 014574 004767 000060 JSR PC,CHECK2 ; TEST KER, SUP, AND USE STACKS
4257 014600 022716 000000 RET3: CMP #0,(SP) ;IS USE STACK CLEARED
4258 014604 001403 BEQ 1# ;YES GO ON
4259 ;NO GO TO ERROR
4260 014606 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
4261 014610 000354 .WORD 354 ;UNIQUE ERROR NUMBER
4262 014612 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
4263
4264 014614 005037 177776 1#: CLR B#177776 ;SET PSW TO KER MODE
4265 014620 022716 014404 CMP #RET1,(SP) ;IS KER STACK CORRECT
4266 014624 001403 BEQ 2# ;YES GO ON
4267 ;NO GO TO ERROR
4268 014626 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
4269 014630 000355 .WORD 355 ;UNIQUE ERROR NUMBER
4270 014632 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
4271
4272 014634 012737 040000 177776 2#: MOV #40000,B#177776 ;SET PSW TO SUP MODE
4273 014642 022716 014506 CMP #RET2,(SP) ;IS SUP STACK CORRECT
4274 014646 001403 BEQ 3# ;YES GO ON
4275 ;NO GO TO ERROR
4276 014650 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
4277 014652 000356 .WORD 356 ;UNIQUE ERROR NUMBER
4278 014654 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE

```

```

4279
4280 014656 000207      3#:   RTS      PC           ;RETURN
4281
4282                    ;ROUTINE TO CHECK STACKS AFTER ZERO RTS
4283
4284 014660 022716 014600  CHECK2:  CMP      #RET3,(SP)    ;IS USE STACK CORRECT
4285 014664 001403                BEQ      1#                 ;YES GO ON
4286
4287 014666 104000                ERROR                    ;NO GO TO ERROR
4288 014670 000357                .WORD    357                ;ALL ERRORS TO TRAP TO EMT VECTOR
4289 014672 001127                .WORD    CPUERR             ;UNIQUE ERROR NUMBER
4290
4291 014674 012737 040000 177776 1#:   MOV      #40000,#177776    ;SET PSW TO SUP MODE
4292 014702 022716 014506                CMP      #RET2,(SP)        ;IS SUP STACK CORRECT
4293 014706 001403                BEQ      2#                 ;YES GO ON
4294
4295 014710 104000                ERROR                    ;NO GO TO ERROR
4296 014712 000360                .WORD    360                ;ALL ERRORS TO TRAP TO EMT VECTOR
4297 014714 001127                .WORD    CPUERR             ;UNIQUE ERROR NUMBER
4298
4299 014716 005037 177776                CLR      #177776           ;ADDRESS OF ERROR MESSAGE
4300 014722 022716 014404                2#:   CLR      #177776           ;SET PSW TO KER MODE
4301 014726 001403                CMP      #RET1,(SP)        ;IS KER STACK CORRECT
4302
4303 014730 104000                ERROR                    ;YES GO ON
4304 014732 000361                .WORD    361                ;NO GO TO ERROR
4305 014734 001127                .WORD    CPUERR             ;ALL ERRORS TO TRAP TO EMT VECTOR
4306
4307 014736 012737 140000 177776 3#:   MOV      #140000,#177776    ;UNIQUE ERROR NUMBER
4308 014744 000207                RTS      PC                 ;ADDRESS OF ERROR MESSAGE
4309
4310 014746                    ;
4311 014746                    MTSO:
4312
4313                    MMVCC:
4314
4315 014746                    ;*****
4316 014746 005267 164032                ;*TEST 127      TEST MOV CONDITION CODES - **0-
4317 014752 000277                ;*****
4318 014754 000244                TST127:
4319 014756 012704 000000                INC      #TESTN            ;INCREMENT TEST NUMBER
4320 014762 100403                SCC
4321 014764 102402                CLZ
4322 014766 103001                MOV      #0,R4             ;CC=1011
4323 014770 001403                BMI     1#                 ;CC=0101, R4=0
4324
4325 014772                    1#:   BVS     1#                 ;ERROR IF N FLAG
4326 014772 104000                BCC    1#                 ;ERROR IF V FLAG SET
4327 014774 000362                BEQ     2#                 ;ERROR IF C FLAG CLEAR
4328 014776 001127                BEQ     2#                 ;SKIP IF Z FLAG SET
4329
4330 015000 000277                2#:   SCC
4331 015002 000251                .WORD    251                ;CC SHOULD=0101
4332 015004 012704 100000                MOV      #100000,R4        ;ALL ERRORS TO TRAP TO EMT VECTOR
4333 015010 001403                BEQ     3#                 ;UNIQUE ERROR NUMBER
4334 015012 102402                BVS     3#                 ;ADDRESS OF ERROR MESSAGE

```

```

4335 015014 103401          BCS      3#          ;ERROR IF C SET
4336 015016 100403          BMI      4#          ;EXIT IF N SET
4337                                ;ERROR! CC SHOULD= 1000
4338 015020          3#:
4339 015020 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
4340 015022 000363          .WORD    363          ;UNIQUE ERROR NUMBER
4341 015024 001127          .WORD    CPUERR       ;ADDRESS OF ERROR MESSAGE
4342
4343 015026          4#:
4344
4345
4346 015026          MBTCC:
4347          ;*****
4348          ;*TEST 130      TEST BIT CONDITION CODES - **0-
4349          ;*****
4350 015026          TST130:
4351 015026 005267 163752          INC      $TESTN          ;INCREMENT TEST NUMBER
4352 015032 005004          CLR      R4
4353 015034 005104          COM     R4              ;R4=-1
4354 015036 000277          SCC
4355 015040 000244          CLZ          ;CC=1011
4356 015042 032704 000000          BIT     #0,R4          ;CC=0101
4357 015046 100403          BMI     1#              ;ERROR IF N FLAG
4358 015050 102402          BVS     1#              ;ERROR IF V FLAG SET
4359 015052 103001          BCC     1#              ;ERROR IF C FLAG CLEAR
4360 015054 001403          BEQ     2#              ;SKIP IF Z FLAG SET
4361                                ;ERROR! CC SHOULD=0101
4362 015056          1#:
4363 015056 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
4364 015060 000364          .WORD    364          ;UNIQUE ERROR NUMBER
4365 015062 001127          .WORD    CPUERR       ;ADDRESS OF ERROR MESSAGE
4366
4367 015064 000277          2#:  SCC
4368 015066 000251          .WORD    251          ;CC=0110
4369 015070 032704 100000          BIT     #100000,R4    ;CC=1000
4370 015074 001403          BEQ     3#              ;ERROR IF Z SET
4371 015076 102402          BVS     3#              ;ERROR IF V SET
4372 015100 103401          BCS     3#              ;ERROR IF C SET
4373 015102 100403          BMI     4#              ;EXIT IF N SET
4374                                ;ERROR! CC SHOULD= 1000
4375 015104          3#:
4376 015104 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
4377 015106 000365          .WORD    365          ;UNIQUE ERROR NUMBER
4378 015110 001127          .WORD    CPUERR       ;ADDRESS OF ERROR MESSAGE
4379
4380 015112          4#:
4381
4382
4383 015112          MBCCC:
4384          ;*****
4385          ;*TEST 131      TEST BIC CONDITION CODES - **0-
4386          ;*****
4387 015112          TST131:
4388 015112 005267 163666          INC      $TESTN          ;INCREMENT TEST NUMBER
4389 015116 005004          CLR      R4
4390 015120 005104          COM     R4              ;R4=-1

```



```

4391 015122 000277          SCC
4392 015124 000244          CLZ
4393 015126 042704 177777  BIC      #177777,R4
4394 015132 100403          BMT      1#
4395 015134 102402          BVS      1#
4396 015136 103001          BCC      1#
4397 015140 001403          BEQ      2#
4398
4399          015142          1#:
4400 015142 104000          ERROR
4401 015144 000366          .WORD   366
4402 015146 001127          .WORD   CPUERR
4403
4404 015150 005104          2#:   COM      R4
4405 015152 000277          SCC
4406 015154 000251          .WORD   251
4407 015156 042704 077777  BIC      #77777,R4
4408 015162 001403          BEQ      3#
4409 015164 102402          BVS      3#
4410 015166 103401          BCS      3#
4411 015170 100403          BMT      4#
4412
4413          015172          3#:
4414 015172 104000          ERROR
4415 015174 000367          .WORD   367
4416 015176 001127          .WORD   CPUERR
4417
4418 015200          4#:
4419
4420
4421 015200          MBSCC:
4422          ;*****
4423          ;*TEST 132      TEST BIS CONDITION CODES
4424          ;*****
4425          TST132:
4426 015200 005267 163600  INC      #TESTN
4427 015204 005004          CLR      R4
4428 015206 000277          SCC
4429 015210 000246          .WORD   246
4430 015212 052704 000000  BIS      #0,R4
4431 015216 100403          BMT      1#
4432 015220 102402          BVS      1#
4433 015222 103001          BCC      1#
4434 015224 001403          BEQ      2#
4435
4436 015226          1#:
4437 015226 104000          ERROR
4438 015230 000370          .WORD   370
4439 015232 001127          .WORD   CPUERR
4440
4441          015234          2#:   SCC
4442 015236 000241          CLC
4443 015240 052704 100076  BIS      #100076,R4
4444 015244 001403          BEQ      3#
4445 015246 102402          BVS      3#
4446 015250 103401          BCS      3#

;CC=1011
;CC=0101
;ERROR IF N FLAG
;ERROR IF V FLAG SET
;ERROR IF C FLAG CLEAR
;SKIP IF Z FLAG SET
;ERROR! CC SHOULD=0101

;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE

;R4=-1

;CC=0110
;CC=1000
;ERROR IF Z SET
;ERROR IF V SET
;ERROR IF C SET
;EXIT IF N SET
;ERROR! CC SHOULD= 1000

;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE

;INCREMENT TEST NUMBER
;R4=0

;CC=1001
;R4=0, CC=0101
;ERROR IF MINUS
;ERROR IF V SET
;ERROR IF C CLEAR
;BRANCH IF GOOD
;ERROR! BIS CC FAILED

;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE

;CC=1110
;R4=100076, CC=1000
;ERROR IF Z SET
;ERROR IF V SET
;ERROR IF C SET

```

```

4447 015252 100403          BMI      4#          ;BRANCH IF GOOD
4448                                     ;ERROR! BAD BIS CC
4449 015254                3#:
4450 015254 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
4451 015256 000371          .WORD     371        ;UNIQUE ERROR NUMBER
4452 015260 001127          .WORD     CPUERR     ;ADDRESS OF ERROR MESSAGE
4453
4454 015262                4#:
4455
4456
4457 015262                MDCCC:
4458                ;*****
4459                ;*TEST 133      TEST DEC, INC CONDITION CODES
4460                ;*****
4461 015262                TST133:
4462 015262 005267 163516    INC      #TESTN      ;INCREMENT TEST NUMBER
4463 015266 012704 077777    MOV      #77777,R4   ;R4=77777
4464 015272 000257          CCC
4465 015274 000261          SFC          ;CC=0001
4466 015276 005204          XNC      R4         ;R4=100000, CC=0011
4467 015300 001403          BEQ      1#         ;ERROR IF ZERO
4468 015302 100002          BPL      1#         ;ERROR IF POSITIVE
4469 015304 102001          BVC      1#         ;ERROR IF V CLEAR
4470 015306 103403          BCS      2#         ;BRANCH IF GOOD
4471                                     ;ERROR! INC FAILED
4472 015310                1#:
4473 015310 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
4474 015312 000372          .WORD     372        ;UNIQUE ERROR NUMBER
4475 015314 001127          .WORD     CPUERR     ;ADDRESS OF ERROR MESSAGE
4476
4477 015316 000257          2#:    CCC
4478 015320 005204          INC      R4         ;R4=100001, CC=1000
4479 015322 103413          BCS      3#         ;ERROR IF C SET
4480 015324 102412          BVS      3#         ;ERROR IF V SET
4481 015326 005304          DEC      R4         ;R4=100000, CC=1000
4482 015330 102410          BVS      3#         ;ERROR IF V SET
4483 015332 103407          BCS      3#         ;ERROR IF C SET
4484 015334 000277          SCC
4485 015336 000252          .WORD     252       ;CC=0101
4486 015340 005304          DEC      R4         ;R4=77777, CC=1011
4487 015342 001403          BEQ      3#         ;ERROR IF Z SET
4488 015344 102002          BVC      3#         ;ERROR IF V CLEAR
4489 015346 103001          BCC      3#         ;ERROR IF C CLEAR
4490 015350 100003          BPL      4#         ;BRANCH IF GOOD
4491                                     ;ERROR! BAD CC
4492 015352                3#:
4493 015352 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
4494 015354 000373          .WORD     373        ;UNIQUE ERROR NUMBER
4495 015356 001127          .WORD     CPUERR     ;ADDRESS OF ERROR MESSAGE
4496
4497 015360                4#:
4498
4499
4500 015360                MCTSCC:
4501                ;*****
4502                ;*TEST 134      TEST CLR, TST, SWAB CONDITION CODES

```

```

4503
4504
4505
4506 015360
4507 015360 005267 163420
4508 015364 000277
4509 015366 000244
4510 015370 005004
4511 015372 100403
4512 015374 102402
4513 015376 103401
4514 015400 001403
4515
4516 015402
4517 015402 104000
4518 015404 000374
4519 015406 001127
4520
4521 015410 005104
4522 015412 000277
4523 015414 005704
4524 015416 001403
4525 015420 102402
4526 015422 103401
4527 015424 100403
4528
4529 015426
4530 015426 104000
4531 015430 000375
4532 015432 001127
4533
4534 015434 000277
4535 015436 000304
4536 015440 102402
4537 015442 103401
4538 015444 100403
4539
4540 015446
4541 015446 104000
4542 015450 000376
4543 015452 001127
4544
4545 015454
4546
4547
4548 015454
4549
4550
4551
4552 015454
4553 015454 005267 163324
4554 015460 012704 077777
4555 015464 012701 000001
4556 015470 000257
4557 015472 060401
4558 015474 102003

```

```

;*****
;0100 - **00 - **00
;*****
TST134:
      INC      $TESTN          ;INCREMENT TEST NUMBER
      SCC
      CLZ
      CLR      R4              ;CC=1011
      BMI     1$              ;R4=0, CC=0100
      BVS     1$              ;ERROR IF MINUS
      BCS     1$              ;ERROR IF V SET
      BEQ     2$              ;ERROR IF C SET
      BEQ     2$              ;BRANCH IF GOOD
      BEQ     2$              ;ERROR! BAD CC ON CLR
1$:
      ERROR
      .WORD   374              ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD   CPUERR          ;UNIQUE ERROR NUMBER
      .WORD
2$:
      COM     R4
      SCC
      TST     R4
      BEQ     3$
      BVS     3$
      BCS     3$
      BMI     4$
      BMI     4$
3$:
      ERROR
      .WORD   375              ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD   CPUERR          ;UNIQUE ERROR NUMBER
      .WORD
4$:
      SCC
      SWAB    R4
      BVS     5$
      BCS     5$
      BMI     6$
      BMI     6$
5$:
      ERROR
      .WORD   376              ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD   CPUERR          ;UNIQUE ERROR NUMBER
      .WORD
6$:
MADCC:
;*****
;+TEST 135      TEST ADD CONDITION CODES - ****
;*****
TST135:
      INC      $TESTN          ;INCREMENT TEST NUMBER
      MOV     $77777,R4       ;R4=77777
      MOV     $1,R1          ;R1=1
      CCC
      ADD     R4,R1          ;CC=0000
      BVC     1$              ;77777 + 1 = 100000 IN R1
      BVC     1$              ;ERROR IF V CLEAR

```

```

4559 015476 103402          BCS      1#          ;ERROR IF CARRY
4560 015500 001401          BEQ      1#          ;ERROR IF Z SET
4561 015502 100403          BMI      2#          ;BRANCH IF GOOD
4562                                     ;ERROR! CC SHOULD = 1010
4563 015504                                     1#:
4564 015504 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
4565 015506 000377          .WORD    377          ;UNIQUE ERROR NUMBER
4566 015510 001127          .WORD    CPUERR       ;ADDRESS OF ERROR MESSAGE
4567
4568 015512 005204          2#:  INC      R4          ;R4=100000
4569 015514 060401          ADD      R4,R1        ;100000 + 100000 = 0 IN R1
4570 015516 102002          BVC     3#          ;ERROR IF V CLEAR
4571 015520 103001          BCC     3#          ;ERROR IF CARRY CLEAR
4572 015522 001403          BEQ     4#          ;BRANCH IF GOOD
4573                                     ;ERROR! CC SHOULD = 0111
4574 015524                                     3#:
4575 015524 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
4576 015526 000400          .WORD    400          ;UNIQUE ERROR NUMBER
4577 015530 001127          .WORD    CPUERR       ;ADDRESS OF ERROR MESSAGE
4578
4579 015532 060401          4#:  ADD      R4,R1        ;0 + 100000 = 100000
4580 015534 102402          BVS     5#          ;ERROR IF V SET
4581 015536 103401          BCS     5#          ;ERROR IF C SET
4582 015540 100403          BMI     6#          ;BRANCH IF GOOD
4583                                     ;ERROR! CC SHOULD = 1000
4584 015542                                     5#:
4585 015542 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
4586 015544 000401          .WORD    401          ;UNIQUE ERROR NUMBER
4587 015546 001127          .WORD    CPUERR       ;ADDRESS OF ERROR MESSAGE
4588
4589 015550                                     6#:
4590
4591
4592 015550          MACCC:
4593          ;*****
4594          ;*TEST 136      TEST ADC CONDITION CODES - ****
4595          ;*****
4596 015550          TST136:
4597 015550 005267 163230    INC      $TESTN        ;INCREMENT TEST NUMBER
4598 015554 012704 177777    MOV      $177777,R4   ;R4=177777
4599 015560 000277          SCC          ;CC=1111
4600 015562 005504          ADC      R4          ;R4=0 CC=0101
4601 015564 100403          BMI     1#          ;ERROR IF MINUS
4602 015566 102402          BVS     1#          ;ERROR IF V SET
4603 015570 103001          BCC     1#          ;ERROR IF C SET
4604 015572 001403          BEQ     2#          ;BRANCH IF GOOD
4605                                     ;ERROR! BAD ADC
4606 015574                                     1#:
4607 015574 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
4608 015576 000402          .WORD    402          ;UNIQUE ERROR NUMBER
4609 015600 001127          .WORD    CPUERR       ;ADDRESS OF ERROR MESSAGE
4610
4611 015602 012704 077777    2#:  MOV      $077777,R4   ;R4=077777
4612 015606 000277          SCC          ;
4613 015610 000242          CLV          ;CC=1101
4614 015612 005504          ADC      R4          ;R4=100000 CC=1010

```



```

4671 015730 012704 077777      4$:  MOV      077777,R4          ;R4=77777
4672 015734 012701 170000      MOV      0170000,R1        ;R1=170000
4673 015740 000257              CCC                      ;CC=0000
4674 015742 020401              CMPL     R4,R1            ;77777 - 170000 = 107777 CC= 1011
4675 015744 102003              BVC      5$              ;ERROR IF V CLEAR
4676 015746 103002              BCC      5$              ;ERROR IF C CLEAR
4677 015750 001401              BEQ      5$              ;ERROR IF ZERO
4678 015752 100403              BMI      6$              ;BRANCH IF GOOD
4679                                ;ERROR! BAD CMP
4680 015754                        5$:
4681 015754 104000              ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
4682 015756 000407              .WORD    407                ;UNIQUE ERROR NUMBER
4683 015760 001127              .WORD    CPUERR            ;ADDRESS OF ERROR MESSAGE
4684
4685 015762 000257      6$:  CCC
4686 015764 005101              COM      R1                ;R1=7777
4687 015766 100403              BMI      7$              ;ERROR IF MINUS
4688 015770 001402              BEQ      7$              ;ERROR IF ZERO
4689 015772 103001              BCC      7$              ;ERROR IF CARRY
4690 015774 102003              BVC      8$              ;BRANCH IF GOOD
4691                                ;ERROR! BAD COM
4692 015776                        7$:
4693 015776 104000              ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
4694 016000 000410              .WORD    410                ;UNIQUE ERROR NUMBER
4695 016002 001127              .WORD    CPUERR            ;ADDRESS OF ERROR MESSAGE
4696
4697 016004 000277      8$:  SCC
4698 016006 005101              COM      R1                ;R1=7777
4699 016010 100403              BMI      10$             ;BRANCH IF GOOD
4700                                ;ERROR! BAD COM
4701 016012                        9$:
4702 016012 104000              ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
4703 016014 000411              .WORD    411                ;UNIQUE ERROR NUMBER
4704 016016 001127              .WORD    CPUERR            ;ADDRESS OF ERROR MESSAGE
4705
4706 016020                        10$:
4707
4708
4709 016020      MSBCC:
4710              ;*****
4711              ;*TEST 140      TEST SUB CONDITION CODES - ****
4712              ;*****
4713 016020      TST140:
4714 016020 005267 162760      INC      $TESTN            ;INCREMENT TEST NUMBER
4715 016024 012704 077775      MOV      077775,R4        ;R4=77775
4716 016030 000257              CCC                      ;CC=0000
4717 016032 162704 137757      SUB      0137757,R4       ;77775 - 137757
4718                                ;TRY TO CAUSE AN ARITHMETIC OVERFLOW
4719 016036 102003              BVC      1$              ;ERROR IF V CLEAR
4720 016040 100002              BPL      1$              ;ERROR IF RESULT IS POSITIVE
4721 016042 001401              BEQ      1$              ;ERROR IF Z SET
4722 016044 103003              BCS      2$              ;BRANCH IF GOOD
4723                                ;ERROR! BAD SUBTRACT
4724 016046                        1$:
4725 016046 104000              ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
4726 016050 000412              .WORD    412                ;UNIQUE ERROR NUMBER
    
```

4727	016052	001127		.WORD	CPUERR		; ADDRESS OF ERROR MESSAGE
4728							
4729	016054	012704	000005	2\$:	MOV	#5,R4	;R4=5
4730	016060	000257			CCC		;CC=0000
4731	016062	162704	000012		SUB	#12,R4	;5-12=-5 AND SETS CARRY
4732	016066	103003			BCC	3\$	;ERROR IF CARRY CLEAR
4733	016070	102402			BVS	3\$	;ERROR IF OVERFLOW
4734	016072	001401			BEQ	3\$	;ERROR IF ZERO
4735	016074	100403			BMI	4\$	;BRANCH IF GOOD
4736							;ERROR! SUBTRACT FAILED
4737	016076			3\$:			
4738	016076	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
4739	016100	000413			.WORD	413	;UNIQUE ERROR NUMBER
4740	016102	001127			.WORD	CPUERR	;ADDRESS OF ERROR MESSAGE
4741							
4742	016104			0\$:			
4743							
4744							
4745	016104						
4746							
4747							
4748							
4749	016104						
4750	016104	005267	162674		INC	#TESTN	;INCREMENT TEST NUMBER
4751	016110	012704	100000		MOV	#100000,R4	;R4=100000
4752	016114	000257			CCC		;C=0000
4753	016116	005604			SBC	R4	;TRY TO SET V
4754	016120	100006			BPL	1\$	;ERROR IF N CLEAR
4755	016122	102405			BVS	1\$	;ERROR IF V SET (HAVENT SET C YET)
4756	016124	000261			SEC		;CC SHOULD = 1001
4757	016126	005604			SBC	R4	;TRY AGAIN TO SET V
4758	016130	102002			BVC	1\$	;ERROR IF V CLEAR
4759	016132	103401			BCS	1\$	;ERROR IF C SET
4760	016134	100003			BPL	2\$	;BRANCH IF GOOD
4761							;ERROR! SBC FAILED
4762	016136			1\$:			
4763	016136	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
4764	016140	000414			.WORD	414	;UNIQUE ERROR NUMBER
4765	016142	001127			.WORD	CPUERR	;ADDRESS OF ERROR MESSAGE
4766							
4767	016144	005004		2\$:	CLR	R4	;R4=0
4768	016146	000277			SCC		
4769	016150	000241			CLC		;CC=1110
4770	016152	005604			SBC	R4	;TRY TO CAUSE C FLAG FAILURE
4771	016154	102410			BCS	3\$	;ERROR IF C SET
4772	016156	102407			BVS	3\$	;ERROR IF V SET
4773	016160	001006			BNE	3\$	;ERROR IF NOT ZERO
4774	016162	000261			SEC		;SET CARRY
4775	016164	005604			SBC	R4	;NOW, 0 - CARRY = 177777
4776	016166	103003			BCC	3\$	;ERROR IF CARRY CLEAR
4777	016170	102402			BVS	3\$	;ERROR IF V SET
4778	016172	001401			BEQ	3\$	;ERROR IF ZERO
4779	016174	100403			BMI	4\$	;BRANCH IF GOOD
4780							;ERROR! SBC FAILED
4781	016176			3\$:			
4782	016176	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR

TSBCCC:

\*\*\*\*\*  
TEST 141 TEST SUB CONDITION CODES - \*\*\*\*  
\*\*\*\*\*

TST141:

Address	Word	Value	Condition Code	Description
4783	016200	000415	.WORD 415	UNIQUE ERROR NUMBER
4784	016202	001127	.WORD CPUERR	ADDRESS OF ERROR MESSAGE
4785				
4786	016204		41:	
4787				
4788				
4789	016204		MRLCC:	
4790			*****	
4791			TEST 142 TEST ROL CONDITION CODES - ****	
4792			*****	
4793	016204		TST142:	
4794	016204	005267 162574	INC #TESTN	INCREMENT TEST NUMBER
4795	016210	012704 060000	MOV #60000,R4	R4= 0110000000000000
4796	016214	000257	CCC	CC=0000
4797	016216	006104	ROL R4	R4= 1100000000000000
4798	016220	103402	BCS 11	ERROR IF CARRY
4799	016222	102001	BVC 11	ERROR IF V CLEAR
4800	016224	100403	BMI 21	BRANCH IF GOOD
4801				ERROR! ROL FAILED
4802	016226		11:	
4803	016226	104000	ERROR	ALL ERRORS TO TRAP TO EMT VECTOR
4804	016230	000416	.WORD 416	UNIQUE ERROR NUMBER
4805	016232	001127	.WORD CPUERR	ADDRESS OF ERROR MESSAGE
4806				
4807	016234	006104	21:	
4808	016236	103002	ROL R4	R4= 1000000000000000
4809	016240	102401	BCC 31	ERROR IF CARRY CLEAR
4810	016242	100403	BVS 31	ERROR IF V SET
4811			BMI 41	BRANCH IF GOOD
4812	016244			ERROR! BAD ROL
4813	016244	104000	31:	
4814	016246	000417	ERROR	ALL ERRORS TO TRAP TO EMT VECTOR
4815	016250	001127	.WORD 417	UNIQUE ERROR NUMBER
4816			.WORD CPUERR	ADDRESS OF ERROR MESSAGE
4817	016252	006104	41:	
4818	016254	102003	ROL R4	R4 = 0000000000000001
4819	016256	103002	BVC 51	ERROR IF V CLEAR
4820	016260	100401	BCC 51	ERROR IF C CLEAR
4821	016262	001003	BMI 51	ERROR IF MINUS
4822			BNE 61	BRANCH IF GOOD
4823	016264			ERROR! BAD ROL
4824	016264	104000	51:	
4825	016266	000420	ERROR	ALL ERRORS TO TRAP TO EMT VECTOR
4826	016270	001127	.WORD 420	UNIQUE ERROR NUMBER
4827			.WORD CPUERR	ADDRESS OF ERROR MESSAGE
4828	016272	006104	61:	
4829	016274	102402	ROL R4	R4=0000000000000011
4830	016276	103401	BVS 71	ERROR IF V SET
4831	016300	100003	BCS 71	ERROR IF C SET
4832			BPL 81	BRANCH IF GOOD
4833	016302			ERROR! BAD ROL
4834	016302	104000	71:	
4835	016304	000421	ERROR	ALL ERRORS TO TRAP TO EMT VECTOR
4836	016306	001127	.WORD 421	UNIQUE ERROR NUMBER
4837			.WORD CPUERR	ADDRESS OF ERROR MESSAGE
4838	016310		81:	



```

4839
4840
4841 016310 MRRCC:
4842
4843
4844
4845 016310
4846 016310 005267 162470
4847 016314 012704 000003
4848 016320 000257
4849 016322 006004
4850 016324 103002
4851 016326 102001
4852 016330 100003
4853
4854 016332
4855 016332 104000
4856 016334 000422
4857 016336 001127
4858
4859 016340 006004
4860 016342 103002
4861 016344 102401
4862 016346 100403
4863
4864 016350
4865 016350 104000
4866 016352 000423
4867 016354 001127
4868
4869 016356 006004
4870 016360 102002
4871 016362 103401
4872 016364 100403
4873
4874 016366
4875 016366 104000
4876 016370 000424
4877 016372 001127
4878
4879 016374 006004
4880 016376 102402
4881 016400 103401
4882 016402 100003
4883
4884 016404
4885 016404 104000
4886 016406 000425
4887 016410 001127
4888
4889 016412
4890
4891
4892
4893
4894

MRRCC:
*****
*TEST 143 TEST ROR CONDITION CODES - ****
*****
TST143:
      INC      #TESTN          ; INCREMENT TEST NUMBER
      MOV      #3,R4          ; R4= 0000000000000011
      CCC
      ROR      R4             ; R4= 0000000000000001
      BCC      1#            ; ERROR IF NO CARRY
      BVC      1#            ; ERROR IF V CLEAR
      BPL      2#            ; BRANCH IF GOOD
                               ; ERROR! ROR FAILED
1#:
      ERROR
      .WORD    322           ; ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    CPUERR       ; UNIQUE ERROR NUMBER
                               ; ADDRESS OF ERROR MESSAGE
2#:
      ROR      R4             ; R4= 1000000000000000
      BCC      3#            ; ERROR IF CARRY CLEAR
      BVS      3#            ; ERROR IF V SET
      BMI      4#            ; BRANCH IF GOOD
                               ; ERROR! BAD ROR
3#:
      ERROR
      .WORD    423           ; ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    CPUERR       ; UNIQUE ERROR NUMBER
                               ; ADDRESS OF ERROR MESSAGE
4#:
      ROR      R4             ; R4 = 1100000000000000
      BVC      5#            ; ERROR IF V
      BCS      5#            ; ERROR IF C SET
      BMI      6#            ; BRANCH IF GOOD
                               ; ERROR! BAD ROR
5#:
      ERROR
      .WORD    424           ; ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    CPUERR       ; UNIQUE ERROR NUMBER
                               ; ADDRESS OF ERROR MESSAGE
6#:
      ROR      R4             ; R4= 0110000000000000
      BVS      7#            ; ERROR IF V SET
      BCS      7#            ; ERROR IF C SET
      BPL      8#            ; BRANCH IF GOOD
                               ; ERROR! BAD ROR
7#:
      ERROR
      .WORD    425           ; ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    CPUERR       ; UNIQUE ERROR NUMBER
                               ; ADDRESS OF ERROR MESSAGE
8#:

*****
*TEST 144 TEST C BIT WITH ROR/ROL

```

```

4895
4896
4897
4898
4899 016412
4900 016412 005267 162366
4901 016416 012701 052525
4902 016422 000241
4903 016424 006001
4904 016426 006001
4905 016430 006001
4906 016432 103403
4907 016434 104000
4908 016436 000426
4909 016440 001127
4910 016442 022701 045252 11:
4911 016446 001403
4912 016450 104000
4913 016452 000427
4914 016454 001127
4915 016456 012701 125252 21:
4916 016462 000241
4917 016464 006101
4918 016466 006101
4919 016470 006101
4920 016472 103403
4921 016474 104000
4922 016476 000430
4923 016500 001127
4924 016502 022701 052522 31:
4925 016506 001403
4926 016510 104000
4927 016512 000431
4928 016514 001127
4929 016516
4930
4931
4932 016516
4933
4934
4935
4936 016516
4937 016516 005267 162262
4938 016522 012704 060000
4939 016526 000257
4940 016530 006304
4941 016532 103402
4942 016534 102001
4943 016536 100403
4944
4945 016540 11:
4946 016540 104000
4947 016542 000432
4948 016544 001127
4949
4950 016546 006304 21:

```

```

;*****
;THIS TEST IS TO CHECK FOR A SLOW C BIT PATH INTERNAL TO THE J11 DATA CHIP
;PROBLEM IS ONLY EXHIBITED ON EARLY MASK SETS (1590 AND 1593)
;*****
TST144:
    INC     #TESTN           ;INCREMENT TEST NUMBER
    MOV     #52525,R1       ;INIT R1 WITH DATA
    CLC
    ROR     R1              ;CLEAR THE C BIT
                           ;R1=025252, C BIT =1
    ROR     R1              ;R1=112525, C BIT =0
    ROR     R1              ;R1=045252, C BIT =1
    BCS     1#             ;BRANCH IF CARRY BIT SET
    ERROR   ;ALL ERRORS TO TRAP TO EMT VECTOR
    .WORD   426            ;UNIQUE ERROR NUMBER
    .WORD   CPUERR        ;ADDRESS OF ERROR MESSAGE
    CMP     #45252,R1      ;IS DATA IN R1 = TO EXPECTED DATA?
    BEQ     2#             ;BRANCH IF YES
    ERROR   ;ALL ERRORS TO TRAP TO EMT VECTOR
    .WORD   427            ;UNIQUE ERROR NUMBER
    .WORD   CPUERR        ;ADDRESS OF ERROR MESSAGE
    MOV     #125252,R1    ;SET UP R1
    CLC
    ROL     R1              ;CLEAR THE CARRY BIT
                           ;R1=052524, C BIT =1
    ROL     R1              ;R1=125251, C BIT =0
    ROL     R1              ;R1=052522, C BIT =1
    BCS     3#             ;BRANCH IF CARRY SET
    ERROR   ;ALL ERRORS TO TRAP TO EMT VECTOR
    .WORD   430            ;UNIQUE ERROR NUMBER
    .WORD   CPUERR        ;ADDRESS OF ERROR MESSAGE
    CMP     #052522, R1   ;IS DATA IN R1 = TO EXPECTED DATA?
    BEQ     4#             ;BRANCH IF YES
    ERROR   ;ALL ERRORS TO TRAP TO EMT VECTOR
    .WORD   431            ;UNIQUE ERROR NUMBER
    .WORD   CPUERR        ;ADDRESS OF ERROR MESSAGE
    4#:

```

```

MALCC:
;*****
;TEST 145 TEST ASL CONDITION CODES - ****
;*****
TST145:
    INC     #TESTN           ;INCREMENT TEST NUMBER
    MOV     #60000,R4       ;R4= 0110000000000000
    CCC
    ASL     R4              ;LL=0000
                           ;C=0 R4= 1100000000000000
    BCS     1#             ;ERROR IF CARRY
    BVC     1#             ;ERROR IF V CLEAR
    BMI     2#             ;BRANCH IF GOOD
    1#:
    ERROR   ;ALL ERRORS TO TRAP TO EMT VECTOR
    .WORD   432            ;UNIQUE ERROR NUMBER
    .WORD   CPUERR        ;ADDRESS OF ERROR MESSAGE
    2#:
    ASL     R4              ;C=1 R4= 1000000000000000

```

4951	016550	103002		BCC	3#	ERROR IF CARRY CLEAR
4952	016552	102401		BVS	3#	ERROR IF V SET
4953	016554	100403		BMI	4#	BRANCH IF GOOD
4954						ERROR! BAD ASL
4955	016556		3#:			
4956	016556	104000		ERROR		ALL ERRORS TO TRAP TO EMT VECTOR
4957	016560	000433		.WORD	433	UNIQUE ERROR NUMBER
4958	016562	001127		.WORD	CPUERR	ADDRESS OF ERROR MESSAGE
4959						
4960	016564	006304	4#:	ASL	R4	C=1 R4= 0000000000000000
4961	016566	102003		BVC	5#	ERROR IF V CLEAR
4962	016570	103002		BCC	5#	ERROR IF C CLEAR
4963	016572	100401		BMI	5#	ERROR IF MINUS
4964	016574	001403		BEQ	6#	BRANCH IF GOOD
4965						ERROR! BAD ASL
4966	016576		5#:			
4967	016576	104000		ERROR		ALL ERRORS TO TRAP TO EMT VECTOR
4968	016600	000434		.WORD	434	UNIQUE ERROR NUMBER
4969	016602	001127		.WORD	CPUERR	ADDRESS OF ERROR MESSAGE
4970						
4971	016604	006304	6#:	ASL	R4	C=0 R4= 0000000000000000
4972	016606	102402		BVS	7#	ERROR IF V SET
4973	016610	103401		BCS	7#	ERROR IF C SET
4974	016612	100003		BPL	8#	BRANCH IF GOOD
4975						ERROR! BAD ASL
4976	016614		7#:			
4977	016614	104000		ERROR		ALL ERRORS TO TRAP TO EMT VECTOR
4978	016616	000435		.WORD	435	UNIQUE ERROR NUMBER
4979	016620	001127		.WORD	CPUERR	ADDRESS OF ERROR MESSAGE
4980						
4981	016622		8#:			
4982						
4983						
4984	016622					
4985						
4986						
4987						
4988	016622					
4989	016622	005267	162156	INC	#TESTN	INCREMENT TEST NUMBER
4990	016626	012704	000341	MOV	#341,R4	R4= 0000000011100001
4991	016632	000257		CCC		CC=0000
4992	016634	006204		ASR	R4	R4= 0000000001110000
4993	016636	103002		BCC	1#	ERROR IF NO CARRY
4994	016640	102001		BVC	1#	ERROR IF V CLEAR
4995	016642	100003		BPL	2#	BRANCH IF GOOD
4996						ERROR! ASR FAILED
4997	016644		1#:			
4998	016644	104000		ERROR		ALL ERRORS TO TRAP TO EMT VECTOR
4999	016646	000436		.WORD	436	UNIQUE ERROR NUMBER
5000	016650	001127		.WORD	CPUERR	ADDRESS OF ERROR MESSAGE
5001						
5002	016652	052704	100001	BIS	#100001,R4	R4= 1000000001110001
5003	016656	006204		ASR	R4	R4= 1100000000111000
5004	016660	103002		BCC	3#	ERROR IF CARRY CLEAR
5005	016662	102401		BVS	3#	ERROR IF V SET
5006	016664	100403		BMI	4#	BRANCH IF GOOD

```

MARCC:
|*****
|*TEST 146 TEST ASR CONDITION CODES - ****
|*****
TST146:

```

```

5007
5008 016666          3#:          ;ERROR! BAD ASR
5009 016666 104000   ;ERROR
5010 016670 000437   .WORD 437   ;ALL ERRORS TO TRAP TO EMT VECTOR
5011 016672 001127   .WORD CPUERR ;UNIQUE ERROR NUMBER
5012                                     ;ADDRESS OF ERROR MESSAGE
5013 016674 006204   4#:  ASR    R4      ;R4= 1110000000011100
5014 016676 102002   BVC    5#      ;ERROR IF V
5015 016700 103401   BCS    5#      ;ERROR IF C SET
5016 016702 100403   BMI    6#      ;BRANCH IF GOOD
5017                                     ;ERROR! BAD ASR
5018 016704          5#:          ;ALL ERRORS TO TRAP TO EMT VECTOR
5019 016704 104000   ;ERROR
5020 016706 000440   .WORD 440   ;UNIQUE ERROR NUMBER
5021 016710 001127   .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
5022
5023 016712 006204   6#:  ASR    R4      ;R4= 1111000000001110
5024 016714 102005   BVC    7#      ;ERROR IF V CLEAR
5025 016716 103404   BCS    7#      ;ERROR IF C SET
5026 016720 100003   BPL    7#      ;ERROR IF PLUS
5027 016722 022704 170016  CMP    #170016,R4 ;SEE IF EXPECTED RESULT
5028 016726 001403   BEQ    8#      ;BRANCH IF GOOD
5029                                     ;ERROR! BAD ASR
5030 016730          7#:          ;ALL ERRORS TO TRAP TO EMT VECTOR
5031 016730 104000   ;ERROR
5032 016732 000441   .WORD 441   ;UNIQUE ERROR NUMBER
5033 016734 001127   .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
5034
5035 016736          8#:          ;
5036
5037
5038 016736          MSXTCC:
5039          ;*****
5040          ;*TEST 147      TEST SXT CONDITION CODES / -*0-
5041          ;*****
5042          TST147:
5043 016736 005267 162042  INC    #TESTN      ;INCREMENT TEST NUMBER
5044 016742 012704 123456  MOV    #123456,R4 ;R4=123456
5045 016746 010401          MOV    R4,R1      ;SAVE CONTENTS
5046 016750 000257          CCC          ;CC=0000
5047 016752 006704          SXT    R4          ;R4=0 CC=0100
5048 016754 103403          BCS    1#          ;ERROR IF CARRY
5049 016756 100402          BMI    1#          ;ERROR IF MINUS
5050 016760 102401          BVS    1#          ;ERROR IF OVERFLOW
5051 016762 001403          BEQ    2#          ;BRANCH IF GOOD
5052                                     ;ERROR! BAD SXT
5053 016764          1#:          ;ALL ERRORS TO TRAP TO EMT VECTOR
5054 016764 104000   ;ERROR
5055 016766 000442   .WORD 442   ;UNIQUE ERROR NUMBER
5056 016770 001127   .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
5057
5058 016772 010104   2#:  MOV    P1,R4      ;RESTORE R4
5059 016774 000277   SCC          ;CC=1111
5060 016776 006704          SXT    R4          ;R4=-1 CC=1001
5061 017000 001405          BEQ    3#          ;ERROR IF ZERO
5062 017002 100004          BPL    3#          ;ERROR IF PLUS

```

5063	017004	103003		BCC	3#		;ERROR IF NO CARRY
5064	017006	102402		BVS	3#		;ERROR IF OVERFLOW
5065	017010	005104		COM	R4		;R4=0
5066	017012	001403		BEQ	4#		;BRANCH IF GOOD
5067							;ERROR! BAD SXT
5068	017014		3#:				
5069	017014	104000		ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
5070	017016	000443		.WORD	443		;UNIQUE ERROR NUMBER
5071	017020	001127		.WORD	CPUErr		;ADDRESS OF ERROR MESSAGE
5072							
5073	017022		4#:				
5074							
5075							
5076	017022			MXRCC:			
5077				;*****			
5078				;*TEST 150 TEST XOR CONDITION CODES / **0-			
5079				;*****			
5080	017022			TST150:			
5081	017022	005267	161756	INC	#TESTN		;INCREMENT TEST NUMBER
5082	017026	012704	123456	MOV	#123456,R4		;R4=123456
5083	017032	012701	052525	MOV	#52525,R1		;R1=52525
5084	017036	000257		CCC			;CC=0000
5085	017040	074104		XOR	R1,R4		;*TI* R4=171173
5086	017042	102403		BVS	1#		;ERROR IF OVERFLOW
5087	017044	001402		BEQ	1#		;ERROR IF ZERO
5088	017046	103401		BCS	1#		;ERROR IF CARRY
5089	017050	100403		BMI	2#		;BRANCH IF GOOD
5090							;ERROR! BAD XOR
5091	017052		1#:				
5092	017052	104000		ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
5093	017054	000444		.WORD	444		;UNIQUE ERROR NUMBER
5094	017056	001127		.WORD	CPUErr		;ADDRESS OF ERROR MESSAGE
5095							
5096	017060	012701	125252	2#:	MOV	#125252,R1	;R1=125252
5097	017064	000277		SCC			;CC=1111
5098	017066	074104		XOR	R1,R4		;R4=054321
5099	017070	100403		BMI	3#		;ERROR IF MINUS
5100	017072	001402		BEQ	3#		;ERROR IF ZERO
5101	017074	103001		BCC	3#		;ERROR IF CARRY CLEAR
5102	017076	102003		BVC	4#		;BRANCH IF GOOD
5103							;ERROR! BAD XOR
5104	017100		3#:				
5105	017100	104000		ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
5106	017102	000445		.WORD	445		;UNIQUE ERROR NUMBER
5107	017104	001127		.WORD	CPUErr		;ADDRESS OF ERROR MESSAGE
5108							
5109	017106	074404		4#:	XOR	R4,R4	;R4=0
5110	017110	102406		BVS	5#		;ERROR IF OVREFLOW
5111	017112	100405		BMI	5#		;ERROR IF MINUS
5112	017114	103004		BCC	5#		;ERROR IF NO CARRY
5113	017116	001003		BNE	5#		;ERROR IF NOT ZERO
5114	017120	022704	000000	CMP	#0,R4		;SEE IF EXPECTED RESULT
5115	017124	001403		BEQ	6#		;BRANCH IF GOOD
5116							;ERROR! BAD XOR
5117	017126		5#:				
5118	017126	104000		ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR

```

5119 017130 000446          .WORD 446          ;UNIQUE ERROR NUMBER
5120 017132 001127          .WORD CPUERR       ;ADDRESS OF ERROR MESSAGE
5121
5122 017134                6$:
5123
5124
5125
5126 017134                ;
                    ;MSXT:
5127                ;*****
5128                ;*TEST 151      TEST SXT (SIGN EXTEND INSTRUCTION)
5129                ;*****
5130                ;AN ADDITIONAL TEST IS INCLUDED TO CHECK FOR A SLOW N BIT PATH
5131                ;ON A TRANSITION FROM ZERO TO ONE INTERNAL TO THE J11 DATA CHIP
5132                ;THE PROBLEM IS ONLY EXHIBITED ON EARLY MASK SETS (1590 OR 1593)
5133                ;*****
5134 017134                TST151:
5135 017134 005267 161644    INC      $TESTN      ;INCREMENT TEST NUMBER
5136 017140 005004          CLR      R4          ;TRASH R4
5137 017142 000257          CCC          ;CC=0000
5138 017144 000271          .WORD 271          ;CC=1001
5139 017146 006704          SXT      R4          ;*TEST INSTRUCTION
5140 017150 102405          BVS     1$          ;BRANCH IF OVERFLOW IS NOT CLEARED
5141 017152 100004          BPL     1$          ;BRANCH IF N BIT EFFECTED
5142 017154 001403          BEQ     1$          ;BRANCH IF Z BIT EFFECTED
5143 017156 103002          BCC     1$          ;BRANCH IF C BIT EFFECTED
5144 017160 005204          INC      R4
5145 017162 001403          BEQ     2$          ;BRANCH IF R4 =0
5146
5147 017164                1$:
5148 017164 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
5149 017166 000447          .WORD 447          ;UNIQUE ERROR NUMBER
5150 017170 001127          .WORD CPUERR       ;ADDRESS OF ERROR MESSAGE
5151 017172 000277          2$:
5152 017174 000250          SCC          ;CC=0111
5153 017176 005004          CLN
5154 017200 012714 000055    MOV     $5,(R4)    ;TRASH R4
5155 017204 006714          SXT     (R4)       ;*TEST INSTRUCTION
5156 017206 001005          BNE     3$          ;BRANCH IF BIT EFFECTED
5157 017210 102404          BVS     3$          ;BRANCH IF OVERFLOW
5158 017212 103403          BCS     3$
5159 017214 100402          BMI     3$          ;BRANCH IF N IS SET
5160 017216 005714          TST     (R4)       ;VERIFY INSTRUCTION WORKED
5161 017220 001403          BEQ     4$          ;BRANCH IF R4=0
5162
5163 017222                3$:
5164 017222 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
5165 017224 000450          .WORD 450          ;UNIQUE ERROR NUMBER
5166 017226 001127          .WORD CPUERR       ;ADDRESS OF ERROR MESSAGE
5167
5168                ;
5169                ;NOW TEST FOR SLOW N BIT IN J11 DATA CHIP
5170 017230 012700 177777    4$:
5171 017234 005004          MOV     #-1,R0     ;R0=177777, N BIT = 1
5172 017236 006700          CLR     R4          ;CLEAT THE N BIT
5173                SXT     R0          ;***TEST INSTRUCTION***
5174 017240 005700          TST     R0          ;TEST N BIT TRANSITION 1 TO 0
                    ;R0 SHOULD = 0

```

```

5175 017242 001403      BEQ      5#           ;BRANCH IF OK
5176 017244 104000      ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
5177 017246 000451      .WORD   451             ;UNIQUE ERROR NUMBER
5178 017250 001127      .WORD   CPUERR         ;ADDRESS OF ERROR MESSAGE
5179 017252 005000      5#:   CLR      RO       ;CLEAR RO, N BIT = 0
5180 017254 012704 177777  MOV     #-1,R4        ;SET N BIT
5181 017260 006700      SXT     RO            ;***TEST INSTRUCTION***
5182                                     ;TEST N BIT TRANSITION 0 TO 1
5183 017262 022700 177777  CMP     #-1,RO        ;RO SHOULD = 177777
5184 017266 001403      BEQ     6#           ;BRANCH IF OK
5185 017270 104000      ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
5186 017272 000452      .WORD   452             ;UNIQUE ERROR NUMBER
5187 017274 001127      .WORD   CPUERR         ;ADDRESS OF ERROR MESSAGE
5188 017276                                     6#:
5189
5190
5191 017276      |
5192      |MXOR:
5193      |*****
5194      |*TEST 152      TEST XOR
5195      |*****
5196 017276      TST152:
5197 017302 005267 161502  INC     #TESTN        ;INCREMENT TEST NUMBER
5198 017306 012701 007643  MOV     #7643,R1      ;SETUP DATA
5199 017312 000277 133333  MOV     #133333,R4    ;SETUP DATA
5200 017314 074401      SCC
5201 017316 100006      XOR     R4,R1         ;*TEST INSTRUCTION
5202 017320 001405      BPL     1#           ;BRANCH IF PLUS TO ERROR
5203 017322 103004      BEQ     1#           ;ERROR IF ZERO
5204 017324 102403      BCC     1#           ;ERROR IF CARRY CLEAR
5205 017326 020127 134570  BVS     1#           ;ERROR IF V SET
5206 017332 001403      CMP     R1,#134570    ;VERIFY CORRECT RESULT
5207                                     BEQ     2#           ;BRANCH IF GOOD
5208 017334                                     ;ERROR! BAD XOR
5209 017334 104000      1#:   ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
5210 017336 000453      .WORD   453             ;UNIQUE ERROR NUMBER
5211 017340 001127      .WORD   CPUERR         ;ADDRESS OF ERROR MESSAGE
5212 017342 010102      2#:   MOV     R1,R2
5213 017344 000257      CCC
5214 017346 074402      XOR     R4,R2         ;*TEST INSTRUCTION
5215 017350 100405      BMI     3#           ;ERROR IF MINUS
5216 017352 102404      BVS     3#           ;ERROR IF OVERFLOW
5217 017354 103403      BCS     3#           ;ERROR IF CARRY
5218 017356 020227 007643  CMP     R2,#7643
5219 017362 001403      BEQ     4#           ;BRANCH IF GOOD
5220                                     ;ERROR! BAD XOR
5221 017364      3#:
5222 017364 104000      ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
5223 017366 000454      .WORD   454             ;UNIQUE ERROR NUMBER
5224 017370 001127      .WORD   CPUERR         ;ADDRESS OF ERROR MESSAGE
5225 017372      4#:
5226
5227
5228 017372      |
5229      |MSOB:
5230      |*****
      |*TEST 153      TEST SOB

```

```

5231
5232 017372
5233 017372 005267 161406
5234 017376 012704 000555
5235 017402 000277
5236 017404 103017
5237 017406 102016
5238 017410 100015
5239 017412 001014
5240 017414 077405
5241 017416 103005
5242 017420 102004
5243 017422 100003
5244 017424 001002
5245 017426 000167 000020
5246
5247 017432
5248 017432 104000
5249 017434 000455
5250 017436 001127
5251 017440 000167 000006
5252
5253 017444
5254 017444 104000
5255 017446 000456
5256 017450 001127
5257 017452 020427 000000
5258 017456 001403
5259
5260 017460 104000
5261 017462 000457
5262 017464 001127
5263 017466
5264
5265
5266 017466
5267
5268
5269
5270 017466
5271 017466 005267 161312
5272 017472 012706 000700
5273 017476 012737 125252 000776
5274 017504 012705 017526
5275 017510 012746 006437
5276 017514 000277
5277 017516 000116
5278
5279 017520 104000
5280 017522 000460
5281 017524 001127
5282
5283 017526 101002
5284 017530 100001
5285 017532 102403
5286

;*****
TST153:
INC $TESTN ;INCREMENT TEST NUMBER
MOV #555,R4 ;SETUP TEST COUNTER
SCC ;CC=17
1$: BCC 2$ ;ERROR IF CARRY CLEAR
BVC 2$ ;ERROR IF NO OVERFLOW
BPL 2$ ;ERROR IF PLUS
BNE 2$ ;ERROR IF ZERO
SOB R4,1$ ;*TEST INSTRUCTION
BCC 3$ ;ERROR IF CARRY CLEAR
BVC 3$ ;ERROR IF NO OVERFLOW
BPL 3$ ;ERROR IF PLUS
BNE 3$ ;ERROR IF ZERO
JMP 4$
;ERROR! CC EFFECTED DURING TEST
3$: ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 455 ;UNIQUE ERROR NUMBER
.WORD CPUERR ;ADDRESS OF ERROR MESSAGE
JMP 4$
;ERROR! CC EFFECTED AFTER TEST
2$: ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 456 ;UNIQUE ERROR NUMBER
.WORD CPUERR ;ADDRESS OF ERROR MESSAGE
4$: CMP R4,#0 ;IS R4 CORRECT
BEQ 5$ ;YES GO ON
;ERROR! NO GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 457 ;UNIQUE ERROR NUMBER
.WORD CPUERR ;ADDRESS OF ERROR MESSAGE
5$:
MMARK:
;*****
;*TEST 154 TEST MARK INSTRUCTION
;*****
TST154:
INC $TESTN ;INCREMENT TEST NUMBER
MOV #STBOT-100,SP ;SETUP TEST STACK = 700
MOV #125252,#STBOT-2 ;SET UP NEW R5 VALUE ON STACK
MOV #1$,R5 ;PUT NEW PC IN R5
MOV #MARK+37,-(SP) ;INSERT MARK 37 INSTRUCTION ONTO STACK
SCC
JMP (SP) ;* TEST INSTRUCTION
;MARK INSTRUCTION SHOULD HAVE GONE TO 1$
;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 460 ;UNIQUE ERROR NUMBER
.WORD CPUERR ;ADDRESS OF ERROR MESSAGE
1$: BHI 2$ ;ERROR IF C OR Z BIT CLEAR
BPL 2$ ;ERROR IF N BIT CLEAR
BVS 3$ ;BRANCH IF V BIT SET
;BAD CONDITION CODES ON MARK

```





```

5343 ;*****
5344 ;*TEST 155 TEST CCC (CLEAR CONDITION CODES) INSTRUCTION
5345 ;*****
5346 017672 TST155:
5347 017672 005267 161106 INC $TESTN ; INCREMENT TEST NUMBER
5348 017676 012737 030017 177776 MOV #30017,#0177776 ; SETUP PSW
5349 017704 000257 CCC ; TEST INSTRUCTION
5350 017706 022737 030000 177776 CMP #30000,#0177776 ; DID IT CLEAR ALL CONDITION CODE BITS
5351 017714 001403 BEQ 1$ ; YES GO ON
5352 ; NO GO TO ERROR
5353 017716 104000 ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
5354 017720 000470 .WORD 470 ; UNIQUE ERROR NUMBER
5355 017722 001127 .WORD CPUERR ; ADDRESS OF ERROR MESSAGE
5356 017724 1$:
5357 ;
5358 ;
5359 ;
5360 ;*****
5361 ;*TEST 156 TEST CLEAR C BIT INSTRUCTION
5362 ;*****
5363 017724 TST156:
5364 017724 005267 161054 INC $TESTN ; INCREMENT TEST NUMBER
5365 017730 012737 030017 177776 MOV #30017,#0177776 ; SETUP PSW
5366 017736 000241 CLC ; TEST INSTRUCTION
5367 017740 022737 030016 177776 CMP #30016,#0177776 ; DID IT CLEAR CARRY BIT
5368 017746 001403 BEQ 1$ ; YES GO ON
5369 ; C BIT NOT CLEAR GO TO ERROR
5370 017750 104000 ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
5371 017752 000471 .WORD 471 ; UNIQUE ERROR NUMBER
5372 017754 001127 .WORD CPUERR ; ADDRESS OF ERROR MESSAGE
5373 017756 1$:
5374 ;
5375 ;
5376 017756 TE102:
5377 ;*****
5378 ;*TEST 157 TEST CLN (CLEAR N BIT) INST
5379 ;*****
5380 017756 TST157:
5381 017756 005267 161022 INC $TESTN ; INCREMENT TEST NUMBER
5382 017762 012737 030017 177776 MOV #30017,#0177776 ; SETUP PSW
5383 017770 000250 CLN ; TEST INSTRUCTION
5384 017772 022737 030007 177776 CMP #30007,#0177776 ; DID IT CLEAR NEGATIVE BIT
5385 020000 001403 BEQ 1$ ; YES GO ON
5386 ; NO GO TO ERROR
5387 020002 104000 ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
5388 020004 000472 .WORD 472 ; UNIQUE ERROR NUMBER
5389 020006 001127 .WORD CPUERR ; ADDRESS OF ERROR MESSAGE
5390 020010 1$:
5391 ;
5392 ;
5393 020010 TE103:
5394 ;*****
5395 ;*TEST 160 TEST CLV (CLEAR V BIT) INST
5396 ;*****
5397 020010 TST160:
5398 020010 005267 160770 INC $TESTN ; INCREMENT TEST NUMBER

```



```

5455 020152 04000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
5456 020154 000476        .WORD      476          ;UNIQUE ERROR NUMBER
5457 020156 001127        .WORD      CPUERR       ;ADDRESS OF ERROR MESSAGE
5458 020160              1$:
5459
5460
5461 020160              ;
5462              ;TE107:
5463              ;|*****
5464              ;|*TEST 164      TEST SEN (SET N BIT) INST
5465              ;|*****
5466              ;TST164:
5466 020160 005267 160620    INC          $TESTN      ;INCREMENT TEST NUMBER
5467 020164 012737 030000 177776  MOV          #30000,0#177776 ;SETUP PSW
5468 020172 000270          SEN          ;TEST INSTRUCTION
5469 020174 022737 030010 177776  CMP          #30010,0#177776 ;DID IT SET THE NEGATIVE BIT
5470 020202 001403          BEQ          1$          ;YES GO ON
5471
5472 020204 104000          ERROR          ;NO GO TO ERROR
5473 020206 000477        .WORD      477          ;ALL ERRORS TO TRAP TO EMT VECTOR
5474 020210 001127        .WORD      CPUERR       ;UNIQUE ERROR NUMBER
5475 020212              1$:          ;ADDRESS OF ERROR MESSAGE
5476
5477
5478 020212              ;
5479              ;TE110:
5480              ;|*****
5481              ;|*TEST 165      TEST SEV (SET V BIT) INST
5482              ;|*****
5483              ;TST165:
5483 020212 005267 160566    INC          $TESTN      ;INCREMENT TEST NUMBER
5484 020216 012737 030000 177776  MOV          #30000,0#177776 ;SETUP PSW
5485 020224 000262          SEV          ;TEST INSTRUCTION
5486 020226 022737 030002 177776  CMP          #30002,0#177776 ;DID IT SET THE OVERFLOW BIT
5487 020234 001403          BEQ          1$          ;YES GO ON
5488
5489 020236 104000          ERROR          ;NO GO TO ERROR
5490 020240 000500        .WORD      500          ;ALL ERRORS TO TRAP TO EMT VECTOR
5491 020242 001127        .WORD      CPUERR       ;UNIQUE ERROR NUMBER
5492 020244              1$:          ;ADDRESS OF ERROR MESSAGE
5493
5494
5495 020244              ;
5496              ;TE111:
5497              ;|*****
5498              ;|*TEST 166      TEST SEZ (SET Z BIT) INST
5499              ;|*****
5500              ;TST166:
5500 020244 005267 160534    INC          $TESTN      ;INCREMENT TEST NUMBER
5501 020250 012737 030000 177776  MOV          #30000,0#177776 ;SETUP PSW
5502 020256 000264          SEZ          ;TEST INSTRUCTION
5503 020260 022737 030004 177776  CMP          #30004,0#177776 ;DID IT SET THE ZERO BIT
5504 020266 001403          BEQ          1$          ;YES GO ON
5505
5506 020270 104000          ERROR          ;NO GO TO ERROR
5507 020272 000501        .WORD      501          ;ALL ERRORS TO TRAP TO EMT VECTOR
5508 020274 001127        .WORD      CPUERR       ;UNIQUE ERROR NUMBER
5509 020276              1$:          ;ADDRESS OF ERROR MESSAGE
5510

```

Line	Address	OpCode	Operand 1	Operand 2	Operand 3	Instruction	Comment
5511							
5512	020276					TST167:	
5513						*****	
5514						TEST 167 TEST MULTIPLE CLEARS OF CC [27]	
5515						*****	
5516	020276					TST167:	
5517	020276	005267	160502			INC \$TESTN	INCREMENT TEST NUMBER
5518	020302	012737	030000	177776		MOV #30000, @177776	INIT PSW
5519	020310	000277				SCC	SETUP PSW
5520	020312	000243				.WORD 243	TEST CLC CLV
5521	020314	022737	030014	177776		CMP #30014, @177776	PSW CORRECT?
5522	020322	001403				BEQ 11	YES GO ON
5523							NO GO TO ERROR
5524	020324	104000				ERROR	ALL ERRORS TO TRAP TO EMT VECTOR
5525	020326	000502				.WORD 502	UNIQUE ERROR NUMBER
5526	020330	001127				.WORD CPUERR	ADDRESS OF ERROR MESSAGE
5527	020332	000277			11:	SCC	SETUP PSW
5528	020334	000243				.WORD 243	TEST CLC CLZ
5529	020336	022737	030012	177776		CMP #30012, @177776	PSW CORRECT?
5530	020344	001403				BEQ 21	YES GO ON
5531							NO GO TO ERROR
5532	020346	104000				ERROR	ALL ERRORS TO TRAP TO EMT VECTOR
5533	020350	000503				.WORD 503	UNIQUE ERROR NUMBER
5534	020352	001127				.WORD CPUERR	ADDRESS OF ERROR MESSAGE
5535							
5536	020354	000277			21:	SCC	SETUP PSW
5537	020356	000246				.WORD 246	TEST CLV CLZ
5538	020360	022737	030011	177776		CMP #30011, @177776	PSW CORRECT?
5539	020366	001403				BEQ 31	YES GO ON
5540							NO GO TO ERROR
5541	020370	104000				ERROR	ALL ERRORS TO TRAP TO EMT VECTOR
5542	020372	000504				.WORD 504	UNIQUE ERROR NUMBER
5543	020374	001127				.WORD CPUERR	ADDRESS OF ERROR MESSAGE
5544							
5545	020376	000277			31:	SCC	SETUP PSW
5546	020400	000247				.WORD 247	TEST CLC CLV CLZ
5547	020402	022737	030010	177776		CMP #30010, @177776	PSW CORRECT?
5548	020410	001403				BEQ 41	YES GO ON
5549							NO GO TO ERROR
5550	020412	104000				ERROR	ALL ERRORS TO TRAP TO EMT VECTOR
5551	020414	000505				.WORD 505	UNIQUE ERROR NUMBER
5552	020416	001127				.WORD CPUERR	ADDRESS OF ERROR MESSAGE
5553							
5554	020420	000277			41:	SCC	SETUP PSW
5555	020422	000251				.WORD 251	TEST CLN CLC
5556	020424	022737	030006	177776		CMP #30006, @177776	PSW CORRECT?
5557	020432	001403				BEQ 51	YES GO ON
5558							NO GO TO ERROR
5559	020434	104000				ERROR	ALL ERRORS TO TRAP TO EMT VECTOR
5560	020436	000506				.WORD 506	UNIQUE ERROR NUMBER
5561	020440	001127				.WORD CPUERR	ADDRESS OF ERROR MESSAGE
5562							
5563	020442	000277			51:	SCC	SETUP PSW
5564	020444	000252				.WORD 252	TEST CLN CLV
5565	020446	022737	030005	177776		CMP #30005, @177776	PSW CORRECT?
5566	020454	001403				BEQ 61	YES GO ON

```

5567                                     ;NO GO TO ERROR
5568 020456 104000                       ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
5569 020460 000507                       .WORD 507 ;UNIQUE ERROR NUMBER
5570 020462 001127                       .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
5571
5572 020464 000277                       60: SCC ;SETUP PSW
5573 020466 000253                       .WORD 253 ; TEST CLN CLC CLV
5574 020470 022737 030004 177776        CMP #30004,#177776 ;PSW CORRECT?
5575 020476 001403                       BEQ 70 ;YES GO ON
5576                                     ;NO GO TO ERROR
5577 020500 104000                       ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
5578 020502 000510                       .WORD 510 ;UNIQUE ERROR NUMBER
5579 020504 001127                       .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
5580
5581 020506 000277                       70: SCC ;SETUP PSW
5582 020510 000254                       .WORD 254 ; TEST CLN CLZ
5583 020512 022737 030003 177776        CMP #30003,#177776 ;PSW CORRECT?
5584 020520 001403                       BEQ 80 ;YES GO ON
5585                                     ;NO GO TO ERROR
5586 020522 104000                       ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
5587 020524 000511                       .WORD 511 ;UNIQUE ERROR NUMBER
5588 020526 001127                       .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
5589
5590 020530 000277                       80: SCC ;SETUP PSW
5591 020532 000255                       .WORD 255 ; TEST CLN CLC CLZ
5592 020534 022737 030002 177776        CMP #30002,#177776 ;PSW CORRECT?
5593 020542 001403                       BEQ 90 ;YES GO ON
5594                                     ;NO GO TO ERROR
5595 020544 104000                       ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
5596 020546 000512                       .WORD 512 ;UNIQUE ERROR NUMBER
5597 020550 001127                       .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
5598
5599 020552 000277                       90: SCC ;SETUP PSW
5600 020554 000256                       .WORD 256 ; TEST CLN CLV CLZ
5601 020556 022737 030001 177776        CMP #30001,#177776 ;SETUP PSW
5602 020564 001403                       BEQ 100 ;YES GO ON
5603                                     ;NO GO TO ERROR
5604 020566 104000                       ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
5605 020570 000513                       .WORD 513 ;UNIQUE ERROR NUMBER
5606 020572 001127                       .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
5607
5608 020574                               100:
5609
5610
5611 020574                               ;
5612                               ;TE113:
5613                               ;*****
5614                               ;*TEST 170 TEST MULTIPLE SETS OF CC BITS
5615                               ;*****
5616 020574 005257 160204                 TST170: INC #TESTN ;INCREMENT TEST NUMBER
5617 020600 012737 030000 177776        MOV #30000,#177776 ;INIT PSW
5618 020606 000263                       .WORD 263 ; TEST SEC SEV
5619 020610 022737 030003 177776        CMP #30003,#177776 ;PSW CORRECT?
5620 020616 001403                       BEQ 10 ;YES GO ON
5621                                     ;NO GO TO ERROR
5622 020620 104000                       ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR

```



```

5679
5680 021002 000257      7#:   CCC           ; SETUP PSW
5681 021004 000274      .WORD 274         ; TEST SEN SEZ
5682 021006 022737 030014 177776  CMP #30014,0#177776 ; PSW CORRECT?
5683 021014 001403      BEQ 8#           ; YES GO ON
5684                                     ; NO GO TO ERROR
5685 021016 104000      ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
5686 021020 000523      .WORD 523         ; UNIQUE ERROR NUMBER
5687 021022 001127      .WORD CPUERR      ; ADDRESS OF ERROR MESSAGE
5688
5689 021024 000257      8#:   CCC           ; SETUP PSW
5690 021026 000275      .WORD 275         ; TEST SEN SEC SEZ
5691 021030 022737 030015 177776  CMP #30015,0#177776 ; PSW CORRECT?
5692 021036 001403      BEQ 9#           ; YES GO ON
5693                                     ; NO GO TO ERROR
5694 021040 104000      ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
5695 021042 000524      .WORD 524         ; UNIQUE ERROR NUMBER
5696 021044 001127      .WORD CPUERR      ; ADDRESS OF ERROR MESSAGE
5697
5698 021046 000257      9#:   CCC           ; SETUP PSW
5699 021050 000276      .WORD 276         ; TEST SEN SEV SEZ
5700 021052 022737 030016 177776  CMP #30016,0#177776 ; PSW CORRECT?
5701 021060 001403      BEQ 10#          ; YES GO ON
5702                                     ; NO GO TO ERROR
5703 021062 104000      ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
5704 021064 000525      .WORD 525         ; UNIQUE ERROR NUMBER
5705 021066 001127      .WORD CPUERR      ; ADDRESS OF ERROR MESSAGE
5706
5707 021070      10#:
5708
5709
5710 021070      ;
5711      TE113A:
5712      ;*****
5713      ;*TEST 171      TEST SIGNED AND CONDITIONAL BRANCHES
5714      ;*****
5715      TST171:
5716 021070 005267 157710  INC #TESTN      ; INCREMENT TEST NUMBER
5717 021074 000257      CCC           ; CLEAR ALL CC BITS IN PSW
5718 021076 002003      BGE 1#        ; BGE SHOULD BRANCH
5719                                     ; ERROR! DIDN'T BRANCH
5720 021100 104000      ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
5721 021102 000526      .WORD 526         ; UNIQUE ERROR NUMBER
5722 021104 001127      .WORD CPUERR      ; ADDRESS OF ERROR MESSAGE
5723
5724 021106 003003      1#:   BGT 2#        ; BGT SHOULD BRANCH
5725                                     ; ERROR! DIDN'T BRANCH
5726 021110 104000      ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
5727 021112 000527      .WORD 527         ; UNIQUE ERROR NUMBER
5728 021114 001127      .WORD CPUERR      ; ADDRESS OF ERROR MESSAGE
5729
5729 021116 003401      2#:   BLE 3#        ; BLE SHOULDN'T BRANCH
5730 021120 000403      BR 4#          ; BRANCH TO NEXT TEST
5731                                     ; ERROR; BLE SHOULD NOT HAVE BRANCHED
5732
5732 021122      3#:
5733 021122 104000      ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
5734 021124 000530      .WORD 530         ; UNIQUE ERROR NUMBER

```



5735	021126	001127		.WORD	CPUERR		;ADDRESS OF ERROR MESSAGE
5736							
5737	021130	002401	4#:	BLT	5#		;BLT SHOULD NOT BRANCH
5738	021132	000403		BR	6#		;BRANCH TO NEXT TEST
5739							;ERROR; BLT SHOULD NOT HAVE BRANCHED
5740	021134		5#:				
5741	021134	104000		ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
5742	021136	000531		.WORD	531		;UNIQUE ERROR NUMBER
5743	021140	001127		.WORD	CPUERR		;ADDRESS OF ERROR MESSAGE
5744							
5745	021142	000264	6#:	SEZ			;SET THE Z BIT IN PSW
5746	021142	003403		BLE	7#		;BLE SHOULD BRANCH
5747							;ERROR; BLE DIDN'T BRANCH
5748	021146	104000		ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
5749	021150	000532		.WORD	532		;UNIQUE ERROR NUMBER
5750	021152	001127		.WORD	CPUERR		;ADDRESS OF ERROR MESSAGE
5751							
5752	021154	003001	7#:	BGT	8#		;BGT SHOULD NOT BRANCH
5753	021156	000403		BR	9#		;BRANCH TO NEXT TEST
5754							;ERROR; BGT SHOULD NOT HAVE BRANCHED
5755	021160		8#:				
5756	021160	104000		ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
5757	021162	000533		.WORD	533		;UNIQUE ERROR NUMBER
5758	021164	001127		.WORD	CPUERR		;ADDRESS OF ERROR MESSAGE
5759							
5760	021166	000257	9#:	CCC			;CLEAR ALL CC BITS IN PSW
5761	021170	000270		SEN			;SET N BIT IN PSW
5762	021172	002403		BLT	10#		;SHOULD BRANCH TO NEXT TEST
5763							;ERROR; BLT SHOULD HAVE BRANCHED
5764	021174	104000		ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
5765	021176	000534		.WORD	534		;UNIQUE ERROR NUMBER
5766	021200	001127		.WORD	CPUERR		;ADDRESS OF ERROR MESSAGE
5767							
5768	021202	003403	10#:	BLE	11#		;SHOULD BRANCH TO NEXT TEST
5769							;ERROR; BLE SHOULD HAVE BRANCHED
5770	021204	104000		ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
5771	021206	000535		.WORD	535		;UNIQUE ERROR NUMBER
5772	021210	001127		.WORD	CPUERR		;ADDRESS OF ERROR MESSAGE
5773							
5774	021212	002001	11#:	BGE	12#		;BGE SHOULD NOT BRANCH
5775	021214	000403		BR	13#		;BRANCH TO NEXT TEST
5776							;ERROR; BGE SHOULD NOT HAVE BRANCHED
5777	021216		12#:				
5778	021216	104000		ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
5779	021220	000536		.WORD	536		;UNIQUE ERROR NUMBER
5780	021222	001127		.WORD	CPUERR		;ADDRESS OF ERROR MESSAGE
5781							
5782	021224	003001	13#:	BGT	14#		;BGT SHOULD NOT BRANCH
5783	021226	000403		BR	15#		;BRANCH TO NEXT TEST
5784							;ERROR; BGT SHOULD NOT HAVE BRANCHED
5785	021230		14#:				
5786	021230	104000		ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
5787	021232	000537		.WORD	537		;UNIQUE ERROR NUMBER
5788	021234	001127		.WORD	CPUERR		;ADDRESS OF ERROR MESSAGE
5789							
5790	021236	000257	15#:	CCC			;CLEAR ALL CC BITS

5791	021240	000262	SEV					; SET V BIT IN PSW
5792	021242	003403	BLE	16#				; BLE SHOULD BRANCH
5793								; ERROR; BLE DIDN'T BRANCH
5794	021244	104000	ERROR					; ALL ERRORS TO TRAP TO EMT VECTOR
5795	021246	000540	.WORD	540				; UNIQUE ERROR NUMBER
5796	021250	001127	.WORD	CPUERR				; ADDRESS OF ERROR MESSAGE
5797								
5798	021252	002403			16#:	BLT	17#	; BLT SHOULD BRANCH
5799								; ERROR; BLT DIDN'T BRANCH
5800	021254	104000	ERROR					; ALL ERRORS TO TRAP TO EMT VECTOR
5801	021256	000541	.WORD	541				; UNIQUE ERROR NUMBER
5802	021260	001127	.WORD	CPUERR				; ADDRESS OF ERROR MESSAGE
5803								
5804	021262	002001			17#:	BGE	18#	; BGE SHOULDN'T BRANCH
5805	021264	000403				BR	19#	; BRANCH TO NEXT TEST
5806								; ERROR; BGE SHOULD NOT HAVE BRANCHED
5807	021266				18#:			
5808	021266	104000	ERROR					; ALL ERRORS TO TRAP TO EMT VECTOR
5809	021270	000542	.WORD	542				; UNIQUE ERROR NUMBER
5810	021272	001127	.WORD	CPUERR				; ADDRESS OF ERROR MESSAGE
5811								
5812	021274	003001			19#:	BGT	20#	; BGT SHOULD NOT BRANCH
5813	021276	000403				BR	21#	; BRANCH TO NEXT TEST
5814								; ERROR; BGT SHOULD NOT HAVE BRANCHED
5815	021300				20#:			
5816	021300	104000	ERROR					; ALL ERRORS TO TRAP TO EMT VECTOR
5817	021302	000543	.WORD	543				; UNIQUE ERROR NUMBER
5818	021304	001127	.WORD	CPUERR				; ADDRESS OF ERROR MESSAGE
5819								
5820	021306	000257			21#:	CCC		; CLEAR ALL CC BITS
5821	021310	000272	.WORD	272				; SET N AND V BITS IN PSW
5822	021312	002003	BGE	22#				; BGE SHOULD BRANCH
5823								; ERROR; BGE DIDN'T BRANCH
5824	021314	104000	ERROR					; ALL ERRORS TO TRAP TO EMT VECTOR
5825	021316	000544	.WORD	544				; UNIQUE ERROR NUMBER
5826	021320	001127	.WORD	CPUERR				; ADDRESS OF ERROR MESSAGE
5827								
5828	021322	003003			22#:	BGT	23#	; BGT SHOULD BRANCH
5829								; ERROR; BGT DIDN'T BRANCH
5830	021324	104000	ERROR					; ALL ERRORS TO TRAP TO EMT VECTOR
5831	021326	000545	.WORD	545				; UNIQUE ERROR NUMBER
5832	021330	001127	.WORD	CPUERR				; ADDRESS OF ERROR MESSAGE
5833								
5834	021332	003401			23#:	BLE	24#	; BLE SHOULDN'T BRANCH
5835	021334	000403				BR	25#	; BRANCH TO NEXT TEST
5836								; ERROR; BLE SHOULD NOT HAVE BRANCHED
5837	021336				24#:			
5838	021336	104000	ERROR					; ALL ERRORS TO TRAP TO EMT VECTOR
5839	021340	000546	.WORD	546				; UNIQUE ERROR NUMBER
5840	021342	001127	.WORD	CPUERR				; ADDRESS OF ERROR MESSAGE
5841								
5842	021344	002401			25#:	BLT	26#	; BLT SHOULD NOT BRANCH
5843	021346	000403				BR	27#	; BRANCH TO NEXT TEST
5844								; ERROR; BLT SHOULD NOT HAVE BRANCHED
5845	021350				26#:			
5846	021350	104000	ERROR					; ALL ERRORS TO TRAP TO EMT VECTOR

```

5847 021352 000547 .WORD 547 ;UNIQUE ERROR NUMBER
5848 021354 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
5849
5850 021356 27:
5851
5852 021356 TE114:
5853 ;*****
5854 ;*TEST 172 TEST NOP INST
5855 ;*****
5856 021356 TST172:
5857 021356 005267 157422 INC $TESTN ;INCREMENT TEST NUMBER
5858 021362 012737 030000 177776 MOV $30000,$177776 ;INIT PSW
5859 021370 012700 000001 MOV $1,R0 ;INIT R0
5860 021374 012701 000002 MOV $3,R1 ;INIT R1
5861 021400 012702 000003 MOV $3,R2 ;INIT R2
5862 021404 012703 000004 MOV $4,R3 ;INIT R3
5863 021410 012704 000005 MOV $5,R4 ;INIT R4
5864 021414 012705 000006 MOV $6,R5 ;INIT R5
5865 021420 010637 001030 MOV R6,$SLOC00 ;SAVE SP
5866 021424 012737 030017 001034 MOV $30017,$EXPDAT ;SETUP PSW
5867 021432 000277 SCC ;SET ALL CONDITION CODE BITS
5868 021434 000240 NOP ; TEST INSTRUCTION
5869 021436 000240 NOP ; TEST INSTRUCTION
5870 021440 000240 NOP ; TEST INSTRUCTION
5871 021442 004767 000056 JSR PC,T114 ;CHECK PSW, AND GPR'S
5872 021446 020637 001030 CMP R6,$SLOC00 ;CHECK SP
5873 021452 001403 BEQ 1$ ;OK GO ON
5874
5875 021454 104000 ERROR ;ERROR; BLT SHOULD NOT HAVE BRANCHED
5876 021456 000550 .WORD 550 ;ALL ERRORS TO TRAP TO EMT VECTOR
5877 021460 001127 .WORD CPUERR ;UNIQUE ERROR NUMBER
5878 ;ADDRESS OF ERROR MESSAGE
5879 021462 012737 030000 001034 1$: MOV $30000,$EXPDAT ;SETUP PSW
5880 021470 000257 CCC ;CLEAR ALL CONDITION CODE BITS
5881 021472 000260 .WORD 260 ; TEST FOR NOP OPERATION
5882 021474 000260 .WORD 260 ; TEST FOR NOP OPERATION
5883 021476 000260 .WORD 260 ; TEST FOR NOP OPERATION
5884 021500 004767 000020 JSR PC,T114 ;CHECK PSW, AND GPR'S
5885 021504 020637 001030 CMP R6,$SLOC00 ;CHECK SP
5886 021510 001403 BEQ 2$ ;OK GO ON
5887
5888 021512 104000 ERROR ;ERROR; BLT SHOULD NOT HAVE BRANCHED
5889 021514 000551 .WORD 551 ;ALL ERRORS TO TRAP TO EMT VECTOR
5890 021516 001127 .WORD CPUERR ;UNIQUE ERROR NUMBER
5891 ;ADDRESS OF ERROR MESSAGE
5892 021520 2$:
5893
5894 021520 000167 000140 JMP FINNOP
5895
5896 021524 023737 001034 177776 T114: CMF $EXPDAT,$177776 ;CHECK PSW
5897 021532 001407 BEQ TA114 ;OK GO ON
5898 021534 010067 156640 MOV R0,400 ;SAVE R0
5899
5900 021540 104000 ERROR ;ERROR; BLT SHOULD NOT HAVE BRANCHED
5901 021542 000552 .WORD 552 ;ALL ERRORS TO TRAP TO EMT VECTOR
5902 021544 001127 .WORD CPUERR ;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE

```

```

5903
5904 021546 016700 156626          MOV      400,R0          ;RESTORE R0
5905 021552 022700 000001      TA114:  CMP      #1,R0          ;CHECK R0
5906 021556 001403              BEQ      TB114          ;OK GO ON
5907
5908 021560 104000              ERROR
5909 021562 000553              .WORD   553
5910 021564 001127              .WORD   CPUERR
5911
5912 021566 022701 000002      TB114:  CMP      #2,R1          ;CHECK R1
5913 021572 001403              BEQ      TC114          ;OK GO ON
5914
5915 021574 104000              ERROR
5916 021576 000554              .WORD   554
5917 021600 001127              .WORD   CPUERR
5918
5919 021602 022702 000003      TC114:  CMP      #3,R2          ;CHECK R2
5920 021606 001403              BEQ      TD114          ;OK GO ON
5921
5922 021610 104000              ERROR
5923 021612 000555              .WORD   555
5924 021614 001127              .WORD   CPUERR
5925
5926 021616 022703 000004      TD114:  CMP      #4,R3          ;CHECK R3
5927 021622 001403              BEQ      TF114          ;OK GO ON
5928
5929 021624 104000              ERROR
5930 021626 000556              .WORD   556
5931 021630 001127              .WORD   CPUERR
5932
5933 021632 022704 000005      TF114:  CMP      #5,R4          ;CHECK R4
5934 021636 001403              BEQ      TG114          ;OK GO ON
5935
5936 021640 104000              ERROR
5937 021642 000557              .WORD   557
5938 021644 001127              .WORD   CPUERR
5939
5940 021646 022705 000006      TG114:  CMP      #6,R5          ;CHECK R5
5941 021652 001403              BEQ      TH114          ;OK GO ON
5942
5943 021654 104000              ERROR
5944 021656 000560              .WORD   560
5945 021660 001127              .WORD   CPUERR
5946
5947 021662 000207      TH114:  RTS      PC          ;RETURN
5948
5949 021664      ;FINNOP:
5950      ;*****
5951      ;*TEST 173      TEST ATERNATE REGISTER SET
5952      ;*****
5953 021664      TST173:
5954 021664 005267 157114      INC      #TESTN          ;INCREMENT TEST NUMBER
5955 021670 005000              CLR      R0              ;-----CLEAR-----
5956 021672 005001              CLR      R1              ;-----PRIMARY-----
5957 021674 005002              CLR      R2              ;-----GENERAL-----
5958 021676 005003              CLR      R3              ;-----PURPOSE-----

```





6071 022236  
6072  
6073  
6074  
6075 022236  
6076 022236 005267 156542  
6077 022242 012700 177777  
6078 022246 020027 177777  
6079 022252 001403  
6080  
6081 022254 104000  
6082 022256 000577  
6083 022260 001127  
6084 022262 005000  
6085 022264 020027 000000  
6086 022270 001403  
6087  
6088 022272 104000  
6089 022274 000600  
6090 022276 001127  
6091 022300 012700 125252  
6092 022304 020027 125252  
6093 022310 001403  
6094  
6095 022312 104000  
6096 022314 000601  
6097 022316 001127  
6098 022320 012700 052525  
6099 022324 020027 052525  
6100 022330 001403  
6101  
6102 022332 104000  
6103 022334 000602  
6104 022336 001127  
6105 022340  
6106  
6107  
6108 022340  
6109  
6110  
6111  
6112 022340  
6113 022340 005267 156440  
6114 022344 012701 177777  
6115 022350 020127 177777  
6116 022354 001403  
6117  
6118 022356 104000  
6119 022360 000603  
6120 022362 001127  
6121 022364 005001  
6122 022366 020127 000000  
6123 022372 001403  
6124  
6125 022374 104000  
6126 022376 000604

```

ALROTS:
;*****
;*TEST 174 ALTERNATE REGISTER SET R0 BIT TESTS
;*****
TST174:
      INC      $TESTN          ;INCREMENT TEST NUMBER
      MOV      #177777,R0     ;R0=177777
      CMP      R0,#177777     ;DOES R0=177777
      BEQ      1$             ;YES GO ON
                                ;NO GO TO ERROR
                                ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    577            ;UNIQUE ERROR NUMBER
      .WORD    CPUERR        ;ADDRESS OF ERROR MESSAGE
1$:   CLR      R0              ;R0=0
      CMP      R0,#0          ;DOES R0=0
      BEQ      2$             ;YES GO ON
                                ;NO GO TO ERROR
                                ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    600            ;UNIQUE ERROR NUMBER
      .WORD    CPUERR        ;ADDRESS OF ERROR MESSAGE
2$:   MOV      #125252,R0     ;R0=125252
      CMP      R0,#125252    ;DOES R0=125252
      BEQ      3$             ;YES GO ON
                                ;NO GO TO ERROR
                                ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    601            ;UNIQUE ERROR NUMBER
      .WORD    CPUERR        ;ADDRESS OF ERROR MESSAGE
3$:   MOV      #52525,R0     ;R0=52525
      CMP      R0,#52525    ;DOES R0=52525
      BEQ      4$             ;YES GO ON
                                ;NO GO TO ERROR
                                ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    602            ;UNIQUE ERROR NUMBER
      .WORD    CPUERR        ;ADDRESS OF ERROR MESSAGE
4$:

;
ALR1TS:
;*****
;*TEST 175 ALTERNATE REGISTER SET R1 BIT TESTS
;*****
TST175:
      INC      $TESTN          ;INCREMENT TEST NUMBER
      MOV      #177777,R1     ;R1=177777
      CMP      R1,#177777    ;DOES R1=177777
      BEQ      1$             ;YES GO ON
                                ;NO GO TO ERROR
                                ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    603            ;UNIQUE ERROR NUMBER
      .WORD    CPUERR        ;ADDRESS OF ERROR MESSAGE
1$:   CLR      R1              ;R1=0
      CMP      R1,#0          ;DOES R1=0
      BEQ      2$             ;YES GO ON
                                ;NO GO TO ERROR
                                ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    604            ;UNIQUE ERROR NUMBER

```

```

6127 022400 001127          .WORD CPUERR          ;ADDRESS OF ERROR MESSAGE
6128 022402 012701 125252 2$: MOV  #125252,R1      ;R1=125252
6129 022406 020127 125252  CMP  R1,#125252      ;DOES R1=125252
6130 022412 001403          BEQ  3$              ;YES GO ON
6131                                ;NO GO TO ERROR
6132 022414 104000          ERROR                ;ALL ERRORS TO TRAP TO EMT VECTOR
6133 022416 000605          .WORD 605          ;UNIQUE ERROR NUMBER
6134 022420 001127          .WORD CPUERR          ;ADDRESS OF ERROR MESSAGE
6135 022422 012701 052525 3$: MOV  #52525,R1      ;R1=52525
6136 022426 020127 052525  CMP  R1,#52525      ;DOES R1=52525
6137 022432 001403          BEQ  4$              ;YES GO ON
6138                                ;NO GO TO ERROR
6139 022434 104000          ERROR                ;ALL ERRORS TO TRAP TO EMT VECTOR
6140 022436 000606          .WORD 606          ;UNIQUE ERROR NUMBER
6141 022440 001127          .WORD CPUERR          ;ADDRESS OF ERROR MESSAGE
6142 022442          4$:
6143
6144 ;
6145 022442          ;ALR2TS:
6146 ;*****
6147 ;*TEST 176 ALTERNATE REGISTER SET R2 BIT TESTS
6148 ;*****
6149 022442          TST176:
6150 022442 005267 156336  INC  #TESTN          ;INCREMENT TEST NUMBER
6151 022446 012702 177777  MOV  #177777,R2      ;R2=177777
6152 022452 020227 177777  CMP  R2,#177777      ;DOES R2=177777
6153 022456 001403          BEQ  1$              ;YES GO ON
6154                                ;NO GO TO ERROR
6155 022460 104000          ERROR                ;ALL ERRORS TO TRAP TO EMT VECTOR
6156 022462 000607          .WORD 607          ;UNIQUE ERROR NUMBER
6157 022464 001127          .WORD CPUERR          ;ADDRESS OF ERROR MESSAGE
6158 022466 005002          CLR  R2              ;R2=0
6159 022470 020227 000000 1$: CMP  R2,#0          ;DOES R2=0
6160 022474 001403          BEQ  2$              ;YES GO ON
6161                                ;NO GO TO ERROR
6162 022476 104000          ERROR                ;ALL ERRORS TO TRAP TO EMT VECTOR
6163 022500 000610          .WORD 610          ;UNIQUE ERROR NUMBER
6164 022502 001127          .WORD CPUERR          ;ADDRESS OF ERROR MESSAGE
6165 022504 012702 125252 2$: MOV  #125252,R2      ;R2=125252
6166 022510 020227 125252  CMP  R2,#125252      ;DOES R2=125252
6167 022514 001403          BEQ  3$              ;YES GO ON
6168                                ;NO GO TO ERROR
6169 022516 104000          ERROR                ;ALL ERRORS TO TRAP TO EMT VECTOR
6170 022520 000611          .WORD 611          ;UNIQUE ERROR NUMBER
6171 022522 001127          .WORD CPUERR          ;ADDRESS OF ERROR MESSAGE
6172 022524 012702 052525 3$: MOV  #52525,R2      ;R2=52525
6173 022530 020227 052525  CMP  R2,#52525      ;DOES R2=52525
6174 022534 001403          BEQ  4$              ;YES GO ON
6175                                ;NO GO TO ERROR
6176 022536 104000          ERROR                ;ALL ERRORS TO TRAP TO EMT VECTOR
6177 022540 000612          .WORD 612          ;UNIQUE ERROR NUMBER
6178 022542 001127          .WORD CPUERR          ;ADDRESS OF ERROR MESSAGE
6179 022544          4$:
6180
6181 ;
6182 022544          ;ALR3TS:

```



```

6183
6184
6185
6186 022544
6187 022544 005267 156234
6188 022550 012703 177777
6189 022554 020327 177777
6190 022560 001403
6191
6192 022562 104000
6193 022564 000613
6194 022566 001127
6195 022570 005003
6196 022572 020327 000000
6197 022576 001403
6198
6199 022600 104000
6200 022602 000614
6201 022604 001127
6202 022606 012703 125252
6203 022612 020327 125252
6204 022616 001403
6205
6206 022620 104000
6207 022622 000615
6208 022624 001127
6209 022626 012703 052525
6210 022632 020327 052525
6211 022636 001403
6212
6213 022640 104000
6214 022642 000616
6215 022644 001127
6216 022646
6217
6218
6219 022646
6220
6221
6222
6223 022646
6224 022646 005267 156132
6225 022652 012704 177777
6226 022656 020427 177777
6227 022662 001403
6228
6229 022664 104000
6230 022666 000617
6231 022670 001127
6232 022672 005004
6233 022674 020427 000000
6234 022700 001403
6235
6236 022702 104000
6237 022704 000620
6238 022706 001127

```

```

;*****
;*TEST 177 ALTERNATE REGISTER SET R3 BIT TESTS
;*****
TST177:
      INC      $TESTN          ;INCREMENT TEST NUMBER
      MOV      #177777,R3     ;R3=177777
      CMP      R3,#177777    ;DOES R3=177777
      BEQ      1$           ;YES GO ON
                          ;NO GO TO ERROR
                          ;ALL ERRORS TO TRAP TO EMT VECTOR
                          ;UNIQUE ERROR NUMBER
                          ;ADDRESS OF ERROR MESSAGE
      .WORD    613
      .WORD    CPUERR
1$:   CLR      R3             ;R3=0
      CMP      R3,#0         ;DOES R3=0
      BEQ      2$           ;YES GO ON
                          ;NO GO TO ERROR
                          ;ALL ERRORS TO TRAP TO EMT VECTOR
                          ;UNIQUE ERROR NUMBER
                          ;ADDRESS OF ERROR MESSAGE
      .WORD    614
      .WORD    CPUERR
2$:   MOV      #125252,R3    ;R3=125252
      CMP      R3,#125252   ;DOES R3=125252
      BEQ      3$           ;YES GO ON
                          ;NO GO TO ERROR
                          ;ALL ERRORS TO TRAP TO EMT VECTOR
                          ;UNIQUE ERROR NUMBER
                          ;ADDRESS OF ERROR MESSAGE
      .WORD    615
      .WORD    CPUERR
3$:   MOV      #52525,R3     ;R3=52525
      CMP      R3,#52525   ;DOES R3=52525
      BEQ      4$           ;YES GO ON
                          ;NO GO TO ERROR
                          ;ALL ERRORS TO TRAP TO EMT VECTOR
                          ;UNIQUE ERROR NUMBER
                          ;ADDRESS OF ERROR MESSAGE
      .WORD    616
      .WORD    CPUERR
4$:
;
;ALR4TS:
;*****
;*TEST 200 ALTERNATE REGISTER SET R4 BIT TESTS
;*****
TST200:
      INC      $TESTN          ;INCREMENT TEST NUMBER
      MOV      #177777,R4     ;R4=177777
      CMP      R4,#177777    ;DOES R4=177777
      BEQ      1$           ;YES GO ON
                          ;NO GO TO ERROR
                          ;ALL ERRORS TO TRAP TO EMT VECTOR
                          ;UNIQUE ERROR NUMBER
                          ;ADDRESS OF ERROR MESSAGE
      .WORD    617
      .WORD    CPUERR
1$:   CLR      R4             ;R4=0
      CMP      R4,#0         ;DOES R4=0
      BEQ      2$           ;YES GO ON
                          ;NO GO TO ERROR
                          ;ALL ERRORS TO TRAP TO EMT VECTOR
                          ;UNIQUE ERROR NUMBER
                          ;ADDRESS OF ERROR MESSAGE
      .WORD    620
      .WORD    CPUERR

```

```

6239 022710 012704 125252      20:  MOV    #125252,R4      ;R4=125252
6240 022714 020427 125252      CMP    R4,#125252      ;DOES R4=125252
6241 022720 001403              BEQ    30              ;YES GO ON
6242                                ;NO GO TO ERROR
6243 022722 104000              ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
6244 022724 000621              .WORD 621             ;UNIQUE ERROR NUMBER
6245 022726 001127              .WORD CPUERR          ;ADDRESS OF ERROR MESSAGE
6246 022730 012704 052525      30:  MOV    #52525,R4      ;R4=52525
6247 022734 020427 052525      CMP    R4,#52525      ;DOES R4=52525
6248 022740 001403              BEQ    40              ;YES GO ON
6249                                ;NO GO TO ERROR
6250 022742 104000              ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
6251 022744 000622              .WORD 622             ;UNIQUE ERROR NUMBER
6252 022746 001127              .WORD CPUERR          ;ADDRESS OF ERROR MESSAGE
6253 022750
6254
6255
6256 022750
6257
6258
6259
6260 022750
6261 022750 005267 156030
6262 022754 012705 177777
6263 022760 020527 177777
6264 022764 001403
6265
6266 022766 104000              ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
6267 022770 000623              .WORD 623             ;UNIQUE ERROR NUMBER
6268 022772 001127              .WORD CPUERR          ;ADDRESS OF ERROR MESSAGE
6269 022774 005005              CLR    R5              ;R5=0
6270 022776 020527 000000      10:  CMP    R5,#0          ;DOES R5=0
6271 023002 001403              BEQ    20              ;YES GO ON
6272                                ;NO GO TO ERROR
6273 023004 104000              ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
6274 023006 000624              .WORD 624             ;UNIQUE ERROR NUMBER
6275 023010 001127              .WORD CPUERR          ;ADDRESS OF ERROR MESSAGE
6276 023012 012705 125252      20:  MOV    #125252,R5     ;R5=125252
6277 023016 020527 125252      CMP    R5,#125252    ;DOES R5=125252
6278 023022 001403              BEQ    30              ;YES GO ON
6279                                ;NO GO TO ERROR
6280 023024 104000              ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
6281 023026 000625              .WORD 625             ;UNIQUE ERROR NUMBER
6282 023030 001127              .WORD CPUERR          ;ADDRESS OF ERROR MESSAGE
6283 023032 012705 052525      30:  MOV    #52525,R5     ;R5=52525
6284 023036 020527 052525      CMP    R5,#52525    ;DOES R5=52525
6285 023042 001403              BEQ    40              ;YES GO ON
6286                                ;NO GO TO ERROR
6287 023044 104000              ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
6288 023046 000626              .WORD 626             ;UNIQUE ERROR NUMBER
6289 023050 001127              .WORD CPUERR          ;ADDRESS OF ERROR MESSAGE
6290 023052 042767 004000 154716 40:  BIC    #BIT11,P5      ;RETURN TO PRIMARY GEN PURPOSE REGS
6291
6292
6293 023060
6294
    ;
    ;ALR5TS:
    ;*****
    ;*TEST 201 ALTERNATE REGISTER SET R5 BIT TESTS
    ;*****
    ;TST201:
    ;
    ;INC    #TESTN          ;INCREMENT TEST NUMBER
    ;MOV    #177777,R5     ;R5=177777
    ;CMP    R5,#177777    ;DOES R5=177777
    ;BEQ    10            ;YES GO ON
    ;NO GO TO ERROR
    ;ALL ERRORS TO TRAP TO EMT VECTOR
    ;UNIQUE ERROR NUMBER
    ;ADDRESS OF ERROR MESSAGE
    ;R5=0
    ;DOES R5=0
    ;YES GO ON
    ;NO GO TO ERROR
    ;ALL ERRORS TO TRAP TO EMT VECTOR
    ;UNIQUE ERROR NUMBER
    ;ADDRESS OF ERROR MESSAGE
    ;R5=125252
    ;DOES R5=125252
    ;YES GO ON
    ;NO GO TO ERROR
    ;ALL ERRORS TO TRAP TO EMT VECTOR
    ;UNIQUE ERROR NUMBER
    ;ADDRESS OF ERROR MESSAGE
    ;R5=52525
    ;DOES R5=52525
    ;YES GO ON
    ;NO GO TO ERROR
    ;ALL ERRORS TO TRAP TO EMT VECTOR
    ;UNIQUE ERROR NUMBER
    ;ADDRESS OF ERROR MESSAGE
    ;RETURN TO PRIMARY GEN PURPOSE REGS
    ;
    ;TE115:
    ;*****

```

```

6295      |*TEST 202      TEST MPFS (MOVE FROM PROCCESOR STATUS) INST
6296      |*****
6297      |TST202:
6298      |      INC      #TESTN      ;INCREMENT TEST NUMBER
6299      |      CLR      R4          ;SETUP DESTINATION R4
6300
6301      |      MOV      #TE115A,R1   ;SETUP POINTERS TO TABLES
6302      |      MOV      #TE115B,R2   ;
6303      |      MOV      #TE115C,R3   ;
6304      |      MOV      (R1),#177776 ;SETUP PSW
6305      |      MFPS     R4          ; TEST INSTRUCTION
6306      |      CMP      #177776,(R2) ;CHECK PSW
6307      |      BEQ      2#         ;OK GO ON
6308      |      ERROR   627        ;NO GO TO ERROR
6309      |      .WORD   CPUERR     ;ALL ERRORS TO TRAP TO EMT VECTOR
6310      |      .WORD   CPUERR     ;UNIQUE ERROR NUMBER
6311      |      .WORD   CPUERR     ;ADDRESS OF ERROR MESSAGE
6312
6313      |      CMP      R4,(R3)+    ;CHECK R4
6314      |      BEQ      3#         ;OK GO ON
6315      |      ERROR   630        ;NO GO TO ERROR
6316      |      .WORD   CPUERR     ;ALL ERRORS TO TRAP TO EMT VECTOR
6317      |      .WORD   CPUERR     ;UNIQUE ERROR NUMBER
6318      |      .WORD   CPUERR     ;ADDRESS OF ERROR MESSAGE
6319
6320      |      CMP      (R1),#177777 ;ARE WE DONE
6321      |      BNE     1#         ;NO GO TO 1#
6322
6323
6324      |      MOV      #TE115A,R1   ;SETUP POINTERS TO TABLES
6325      |      MOV      #TE115B,R2   ;
6326      |      MOV      #TE115C,R3   ;
6327      |      MOV      R6,R5      ;
6328      |      MOV      (R1),#177776 ;SAVE STACK IN R5
6329      |      MFPS     R6          ;SETUP PSW
6330      |      CMP      #177776,(R2) ; TEST INSTRUCTION
6331      |      BEQ      102#      ;CHECK PSW
6332      |      ERROR   631        ;OK GO ON
6333      |      .WORD   CPUERR     ;NO GO TO ERROR
6334      |      .WORD   CPUERR     ;ALL ERRORS TO TRAP TO EMT VECTOR
6335      |      .WORD   CPUERR     ;UNIQUE ERROR NUMBER
6336      |      .WORD   CPUERR     ;ADDRESS OF ERROR MESSAGE
6337
6338      |      CMP      R6,(R3)    ;CHECK R6
6339      |      BEQ      103#      ;OK GO ON
6340      |      ERROR   632        ;NO GO TO ERROR
6341      |      .WORD   CPUERR     ;ALL ERRORS TO TRAP TO EMT VECTOR
6342      |      .WORD   CPUERR     ;UNIQUE ERROR NUMBER
6343      |      .WORD   CPUERR     ;ADDRESS OF ERROR MESSAGE
6344      |      MOV      R5,R6      ;RESTORE STACK
6345
6346
6347      |      MOV      #TE115A,R1   ;SETUP POINTERS TO TABLES
6348      |      MOV      #TE115B,R2   ;
6349      |      MOV      #TE115D,R3   ;
6350      |      CLR      @#EXPDAT    ;INIT EXPECTED DATA HOLDER.

```

D10

```

6351 023240 012704 001034 4#: MOV #EXPDAT,R4 ; SETUP POINTER TO TEST LOCATION
6352 023244 012137 177776 MOV (R1)+,#177776 ; SETUP PSW
6353 023250 106724 MFPS (R4); ; TEST INSTRUCTION
6354 023252 023722 177776 CMP #177776,(R2)+ ; CHECK PSW
6355 023256 001403 BEQ 5# ; OK GO ON
6356 ; NO GO TO ERROR
6357 023260 104000 ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
6358 023262 000633 .WORD 633 ; UNIQUE ERROR NUMBER
6359 023264 001127 .WORD CPUERR ; ADDRESS OF ERROR MESSAGE
6360
6361 023266 020427 001035 5#: CMP R4,#EXPDAT+1 ; CHECK R4
6362 023272 001403 BEQ 6# ; OK GO ON
6363 ; NO GO TO ERROR
6364 023274 104000 ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
6365 023276 000634 .WORD 634 ; UNIQUE ERROR NUMBER
6366 023300 001127 .WORD CPUERR ; ADDRESS OF ERROR MESSAGE
6367
6368 023302 023723 001034 6#: CMP #EXPDAT,(R3)+ ; CHECK TEST LOCATION
6369 023306 001403 BEQ 7# ; OK GO ON
6370 ; NO GO TO ERROR
6371 023310 104000 ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
6372 023312 000635 .WORD 635 ; UNIQUE ERROR NUMBER
6373 023314 001127 .WORD CPUERR ; ADDRESS OF ERROR MESSAGE
6374 023316 021127 177777 7#: CMP (R1),#177777 ; ARE WE DONE
6375 023322 001346 BNE 4# ; NO GO TO 4#
6376
6377 023324 000167 000032 JMP YE115F
6378
6379
6380
6381 023330 030207 TE115A: .WORD 30207
6382 023332 030000 .WORD 30000
6383 023334 030057 .WORD 30057
6384 023336 177777 .WORD 177777
6385 023340 030211 TE115B: .WORD 30211
6386 023342 030004 .WORD 30004
6387 023344 030041 .WORD 30041
6388 023346 177607 TE115C: .WORD 177607
6389 023350 000000 .WORD 0
6390 023352 000057 .WORD 57
6391 023354 000207 TE115D: .WORD 207
6392 023356 000000 .WORD 0
6393 023360 000057 .WORD 57
6394 023362 000240 TE115F: NOP
6395
6396 023364 TE116:
6397 ;*****
6398 ;*TEST 203 TEST MTPS (MOVE TO PROCESSOR STATUS) INST
6399 ;*****
6400
6401 023364 005267 155414 TST203: INC #TESTN ; INCREMENT TEST NUMBER
6402
6403 023370 012737 030000 177776 MOV #30000,#177776 ; SET PSW TO KERNEL MODE
6404 023376 012701 023722 MOV #TE116D,R1 ; SETUP POINTERS TO TABLES
6405 023402 012702 023700 MOV #TE116B,R2 ;
6406 023406 010103 MOV R1,R3 ;
    
```

GLOBAL AREAS MACY11 30A(1052) 15-MAR-84 13:28 PAGE 121  
 KDJ11A.MAC 22-FEB-84 15:12 T203 TEST MTPS (MOVE TO PROCESSOR STATUS) INST

SEQ 0121

6407	023410	004767	000142		JSR	PC,T116		; TEST INSTRUCTION AND CHECK PSW ; AND SOURCE OPERAND
6408								
6409								
6410								
6411	023414	012737	140000	177776	MOV	#140000,#177776		; SET PSW TO USER MODE
6412	023422	012706	000600		MOV	#600,R6		; SETUP USER STACK
6413	023426	012701	023722		MOV	#TE116D,R1		; SETUP POINTERS TO TABLES
6414	023432	012702	023712		MOV	#TE116C,R2		
6415	023436	010103			MOV	R1,R3		
6416	023440	004767	000112		JSR	PC,T116		; TEST INSTRUCTION AND CHECK PSW ; AND SOURCE OPERAND
6417								
6418								
6419								
6420	023444	012737	140000	177776	MOV	#140000,#177776		; SET PSW TO USER MODE AND CLEAR CC BITS
6421	023452	012701	023674		MOV	#TE116A,R1		; SETUP POINTERS TO TABLES
6422	023456	012702	023712		MOV	#TE116C,R2		
6423	023462	012703	023722		MOV	#TE116D,R3		
6424	023466	010104			MOV	R1,R4		; SAVE A COPY OF R1 INTO R4
6425	023470	004767	000124		JSR	PC,TA116		; TEST INSTRUCTION AND CHECK PSW ; AND SOURCE OPERAND
6426								
6427								
6428								
6429	023474	012737	030000	177776	MOV	#30000,#177776		; SET PSW TO KERNEL MODE
6430	023502	012701	023674		MOV	#TE116A,R1		; SETUP POINTERS TO TABLES
6431	023506	012702	023700		MOV	#TE116B,R2		
6432	023512	012703	023722		MOV	#TE116D,R3		
6433	023516	010104			MOV	R1,R4		; SAVE A COPY OF R1 INTO R4
6434	023520	004767	000074		JSR	PC,TA116		; TEST INSTRUCTION AND CHECK PSW ; AND SOURCE OPERAND
6435								
6436								
6437	023524	005037	177776		CLR	#177776		; SET PSW TO KERNEL MODE
6438	023530	106427	177412		MTPS	#177412		; TEST INSTRUCTION
6439	023534	022737	000012	177776	CMF	#12,#177776		; IS PSW CORRECT
6440	023542	001403			BEQ	100#		; YES GO ON
6441								; NO GO TO ERROR
6442	023544	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
6443	023546	000636			.WORD	636		; UNIQUE ERROR NUMBER
6444	023550	001127			.WORD	CPUERR		; ADDRESS OF ERROR MESSAGE
6445	023552			100#:				
6446								
6447	023552	000167	000154		JMP	FIN116		
6448								
6449	023556	012105		T116:	MOV	(R1)+,R5		; MOVE TEST DATA TO R5
6450	023560	106405			MTPS	R5		; TEST INSTRUCTION
6451	023562	023722	177776		CMF	#177776,(R2)+		; IS PSW CORRECT
6452	023566	001403			BEQ	1#		; YES GO ON
6453								; NO GO TO ERROR
6454	023570	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
6455	023572	000637			.WORD	637		; UNIQUE ERROR NUMBER
6456	023574	001127			.WORD	CPUERR		; ADDRESS OF ERROR MESSAGE
6457								
6458	023576	022305		1#:	CMF	(R3)+,R5		; IS R5 CORRECT
6459	023600	001403			BEQ	2#		; YES GO ON
6460								; NO GO TO ERROR
6461	023602	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
6462	023604	000640			.WORD	640		; UNIQUE ERROR NUMBER

```

6463 023606 001127          .WORD  CPUERR          ;ADDRESS OF ERROR MESSAGE
6464
6465 023610 021227 177777    2#:  CMP      (R2),#177777    ;ARE WE DONE
6466 023614 001360          BNE      T116              ;NO GO TO T116
6467 023616 000207          RTS       PC               ;RETURN
6468
6469 023620 106421          |
6470 023622 023722 177776    TA116: MTPS      (R1)+      ; TEST INSTRUCTION
6471 023626 001403          CMP      @#177776,(R2)+  ; IS PSW CORRECT
6472                                BEQ      1#                ; YES GO ON
6473                                ERROR   ;NO GO TO ERROR
6474 023630 104000          .WORD    641              ;ALL ERRORS TO TRAP TO EMT VECTOR
6475 023632 000641          .WORD    CPUERR          ;UNIQUE ERROR NUMBER
6476                                .WORD    CPUERR          ;ADDRESS OF ERROR MESSAGE
6477 023636 122423          1#:  CMPS     (R4)+,(R3)+  ;CHECK TEST LOCATION
6478 023640 001403          BEQ      2#                ;OK GO ON
6479                                ERROR   ;NO GO TO ERROR
6480 023642 104000          .WORD    642              ;ALL ERRORS TO TRAP TO EMT VECTOR
6481 023644 000642          .WORD    CPUERR          ;UNIQUE ERROR NUMBER
6482 023646 001127          .WORD    CPUERR          ;ADDRESS OF ERROR MESSAGE
6483
6484 023650 020104          2#:  CMPS     R1,R4        ;IS SOURCE OPERAND CORRECT
6485 023652 001403          BEQ      3#                ;YES GO ON
6486                                ERROR   ;NO GO TO ERROR
6487 023654 104000          .WORD    643              ;ALL ERRORS TO TRAP TO EMT VECTOR
6488 023656 000643          .WORD    CPUERR          ;UNIQUE ERROR NUMBER
6489 023660 001127          .WORD    CPUERR          ;ADDRESS OF ERROR MESSAGE
6490
6491 023662 005203          3#:  INC      R3           ;POINT TO NEXT WORD
6492 023664 021227 177777    CMP      (R2),#177777    ;ARE WE DONE
6493 023670 001353          BNE      TA116            ;NO GO TO TA116
6494 023672 000207          RTS       PC               ;RETURN
6495
6496 023674      377          |
6497 023675      000          TE116A: .BYTE    377
6498 023676      252          .BYTE    0
6499 023677      125          .BYTE    252
6500 023700 030357          .BYTE    125
6501 023702 030000          TE116B: .WORD    30357
6502 023704 030252          .WORD    30000
6503 023706 030105          .WORD    30252
6504 023710 177777          .WORD    30105
6505 023712 140017          .WORD    177777
6506 023714 140000          TE116C: .WORD    140017
6507 023716 140012          .WORD    140000
6508 023720 140005          .WORD    140012
6509 023722 177777          TE116D: .WORD    140005
6510 023724 177400          .WORD    177777
6511 023726 177652          .WORD    177400
6512 023730 177525          .WORD    177652
6513                                .WORD    177525
6514 023732          |
6515          |
6516 023732          |
6517          |
6518          |
;*****
;*TEST 204      TEST M'PT (MOVE FROM PROCESSOR TYPE)

```

G10

```

6519
6520 023732
6521 023732 005267 155046
6522 023736 013746 000010
6523 023742 012737 024070 000010
6524 023750 012700 177777
6525 023754 012737 030000 177776
6526 023762 000007
6527 023764 022737 030000 177776
6528 023772 001403
6529
6530 023774 104000
6531 023776 000644
6532 024000 001127
6533 024002 020027 000005 11:
6534 024006 001403
6535
6536 024010 104000
6537 024012 000645
6538 024014 001127
6539
6540 024016 012700 177777 21:
6541 024022 000277
6542 024024 000007
6543 024026 022737 030017 177776
6544 024034 001403
6545
6546 024036 104000
6547 024040 000646
6548 024042 001127
6549
6550 024044 020027 000005 31:
6551 024050 001403
6552
6553 024052 104000
6554 024054 000647
6555 024056 001127
6556
6557 024060 012637 000010 41:
6558
6559 024064 000167 000006
6560
6561
6562 024070
6563 024070 104000
6564 024072 000650
6565 024074 001127
6566
6567
6568 024076
6569
6570 024076
6571
6572
6573
6574 024076
    
```

```

*****
TST204:
    INC      #TESTN          ; INCREMENT TEST NUMBER
    MOV      @#10,-(SP)      ; SAVE VECTOR
    MOV      #TE117A,@#10    ; SETUP VECTOR TO HANDLE POSSIBLE ILLEGAL INST TR
    MOV      #177777,R0      ; INIT R0
    MOV      #30000,@#177776 ; SETUP PSW
    .WORD    7               ; TEST INSTRUCTION
    CMP      #30000,@#177776 ; IS PSW CORRECT
    BEQ      11             ; YES GO ON
                                ; NO GO TO ERROR
                                ; ALL ERRORS TO TRAP TO EMT VECTOR
    .WORD    644            ; UNIQUE ERROR NUMBER
    .WORD    CPUERR         ; ADDRESS OF ERROR MESSAGE
    CMP      R0,#5          ; IS R0 CORRECT
    BEQ      21             ; YES GO ON
                                ; NO GO TO ERROR
                                ; ALL ERRORS TO TRAP TO EMT VECTOR
    .WORD    645            ; UNIQUE ERROR NUMBER
    .WORD    CPUERR         ; ADDRESS OF ERROR MESSAGE
    MOV      #177777,R0      ; INIT R0
    SCC
    .WORD    7               ; TEST INSTRUCTION
    CMP      #30017,@#177776 ; IS PSW CORRECT
    BEQ      31             ; YES GO ON
                                ; NO GO TO ERROR
                                ; ALL ERRORS TO TRAP TO EMT VECTOR
    .WORD    646            ; UNIQUE ERROR NUMBER
    .WORD    CPUERR         ; ADDRESS OF ERROR MESSAGE
    CMP      R0,#5          ; IS R0 CORRECT
    BEQ      41             ; YES GO ON
                                ; NO GO TO ERROR
                                ; ALL ERRORS TO TRAP TO EMT VECTOR
    .WORD    647            ; UNIQUE ERROR NUMBER
    .WORD    CPUERR         ; ADDRESS OF ERROR MESSAGE
    MOV      (SP)+,@#10      ; RESTORE VECTOR
    JMP      FIN117
;
; ERROR! GO TO ERROR IF TRAP TAKES PLACE
TE117A:
    ERROR
    .WORD    650            ; ALL ERRORS TO TRAP TO EMT VECTOR
    .WORD    CPUERR         ; UNIQUE ERROR NUMBER
    .WORD    CPUERR         ; ADDRESS OF ERROR MESSAGE
;
FIN117:
;
TE120:
*****
; *TEST 205 TEST HALT (NOT KERNEL MODE)
*****
TST205:
    
```

```

6575 024076 005267 154702      INC      $TESTN      ; INCREMENT TEST NUMBER
6576 024102 005037 177766      CLR      B#177766    ; INIT CPU ERROR REG
6577 024106 005037 177776      CLR      B#177776    ; INIT PSW-SET KERNEL MODE
6578 024112 013746 000004      MOV      B#4,-(SP)   ; SAVE VECTOR
6579 024116 013746 000006      MOV      B#6,-(SP)   ; SAVE VECTOR
6580 024122 012737 024162 000004  MOV      #TE120A,B#4 ; SET UP VECTOR TO HANDLE ILLEGAL HALT
6581 024130 005037 000006      CLR      B#6         ; SET UP VECTOR TO COME BACK IN KERNEL MODE
6582 024134 012767 140000 153634  MOV      #140000,PS  ; SET IN USER MODE
6583 024142 012706 000600      MOV      #600,R6    ; INITIALIZE THE USER STACK POINTER
6584 024146 000000      HALT              ; TEST INSTRUCTION
6585
6586          024150      PROCNT:          ; ERROR! IF NOTHING HAPPENED GO TO ERROR
6587 024150 104000      ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
6588 024152 000651      .WORD 651      ; UNIQUE ERROR NUMBER
6589 024154 001127      .WORD CPUERR   ; ADDRESS OF ERROR MESSAGE
6590
6591 024156 000167 000064      JMP      FIN120    ;
6592
6593          ;
6594 024162 022737 030000 177776  TE120A: CMP      #30000,B#177776 ; IS PSW CORRECT/PREVIOUS MODE = USER?
6595 024170 001403      BEQ      1#       ; YES GO ON
6596          ; NO, GO TO ERROR
6597 024172 104000      ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
6598 024174 000652      .WORD 652      ; UNIQUE ERROR NUMBER
6599 024176 001127      .WORD CPUERR   ; ADDRESS OF ERROR MESSAGE
6600
6601 024200 022737 000200 177766 1# :  CMP      #200,B#177766 ; TEST CPU ERROR REGISTER
6602 024206 001403      BEQ      2#       ; YES GO ON
6603          ; NO, GO TO ERROR
6604 024210 104000      ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
6605 024212 000653      .WORD 653      ; UNIQUE ERROR NUMBER
6606 024214 001127      .WORD CPUERR   ; ADDRESS OF ERROR MESSAGE
6607
6608 024216 022627 024150      2# :  CMP      (SP)+,#PROCNT ; DOES STACK CONTAIN CORRECT PC
6609 024222 001403      BEQ      3#       ; YES GO ON
6610          ; NO GO TO ERROR
6611 024224 104000      ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
6612 024226 000654      .WORD 654      ; UNIQUE ERROR NUMBER
6613 024230 001127      .WORD CPUERR   ; ADDRESS OF ERROR MESSAGE
6614 024232 022627 140000      3# :  CMP      (SP)+,#140000 ; DOES STACK CONTAIN CORRECT PSW
6615 024236 001403      BEQ      FIN120   ; YES GO ON
6616          ; NO GO TO ERROR
6617 024240 104000      ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
6618 024242 000655      .WORD 655      ; UNIQUE ERROR NUMBER
6619 024244 001127      .WORD CPUERR   ; ADDRESS OF ERROR MESSAGE
6620
6621 024246 005037 177766      FIN120: CLR      B#CPEREG ; CLEAR CPU ERROR REGISTER
6622 024252 012637 000006      MOV      (SP)+,B#6  ; RESTORE VECTOR
6623 024256 012637 000004      MOV      (SP)+,B#4  ; RESTORE VECTOR
6624
6625          ;
6626 024262      TE121:          ;
6627          ; *****
6628          ; *TEST 206      TEST RESET
6629          ; *****
6630 024262      TST206:        ;

```



GLOBAL AREAS MACY11 30A(1052) 15-MAR-84 13:28 PAGE 125  
 KDJ11A.MAC 22-FEB-84 15:12 T206 TEST RESET

SEQ 0125

6631	024262	005267	154516			INC	\$TESTN				; INCREMENT TEST NUMBER
6632	024266	122767	000001	154524		CMPB	#APTENV, #ENV				; ARE WE IN APT MODE?
6633	024274	001002				BNE	1#				; IF NOT: DO THIS TEST
6634	024276	000167	000426			JMP	FIN121				; ELSE SKIP THIS TEST BECAUSE RESETS
6635											; SCREEN UP THE APT MONITOR.
6636	024302	012737	030340	177776	1#:	MOV	#30340, #177776				; SETUP PSW TO KERNEL MODE
6637	024310	012737	160000	177572		MOV	#160000, #177572				; SETUP MMR0
6638	024316	012737	000077	172516		MOV	#77, #172516				; SETUP MMR3
6639	024324	005037	177772			CLR	#177772				; CLEAR PIRQ
6640	024330	023727	177772	000000		CMP	#177772, #0				; IS PIRQ CORRECT
6641	024336	001403				BEQ	C121A				; YES GO ON
6642											; NO GO TO ERROR
6643	024340	104000				ERROR					; ALL ERRORS TO TRAP TO EMT VECTOR
6644	024342	000656				.WORD	656				; UNIQUE ERROR NUMBER
6645	024344	001127				.WORD	CPUERR				; ADDRESS OF ERROR MESSAGE
6646											
6647	024346	012737	025000	177772	C121A:	MOV	#25000, #177772				; MOVE AN ALTERNATING PATTERN TO PIRQ
6648	024354	022737	025252	177772		CMP	#25252, #177772				; IS PIRQ CORRECT
6649	024362	001403				BEQ	C121B				; YES GO ON
6650											; NO GO TO ERROR
6651	024364	104000				ERROR					; ALL ERRORS TO TRAP TO EMT VECTOR
6652	024366	000657				.WORD	657				; UNIQUE ERROR NUMBER
6653	024370	001127				.WORD	CPUERR				; ADDRESS OF ERROR MESSAGE
6654											
6655	024372	012737	077000	177772	C121B:	MOV	#77000, #177772				; SETUP PIRQ
6656	024400	022737	077314	177772		CMP	#77314, #177772				; IS PIRQ CORRECT
6657	024406	001403				BEQ	C121C				; YES GO ON
6658											; NO GO TO ERROR
6659	024410	104000				ERROR					; ALL ERRORS TO TRAP TO EMT VECTOR
6660	024412	000660				.WORD	660				; UNIQUE ERROR NUMBER
6661	024414	001127				.WORD	CPUERR				; ADDRESS OF ERROR MESSAGE
6662											
6663	024416	000277			C121C:	SCC					; SET ALL CC BITS
6664	024420	000005				RESET					; TEST INSTRUCTION
6665	024422	022737	030357	177776		CMP	#30357, #177776				; IS PSW CORRECT
6666	024430	001403				BEQ	1#				; YES GO ON
6667											; NO GO TO ERROR
6668	024432	104000				ERROR					; ALL ERRORS TO TRAP TO EMT VECTOR
6669	024434	000661				.WORD	661				; UNIQUE ERROR NUMBER
6670	024436	001127				.WORD	CPUERR				; ADDRESS OF ERROR MESSAGE
6671											
6672	024440	013701	177572		1#:	MOV	#SRO, R1				; SAVE SRO IN R1.
6673	024444	042701	000176			BIC	#176, R1				; STRIP OFF UNDEFINED BITS 1-6 FROM MMR0
6674	024450	022701	000000			CMP	#0, R1				; IS MMR0 CORRECT
6675	024454	001403				REQ	2#				; YES GO ON
6676											; NO GO TO ERROR
6677	024456	104000				ERROR					; ALL ERRORS TO TRAP TO EMT VECTOR
6678	024460	000662				.WORD	662				; UNIQUE ERROR NUMBER
6679	024462	001127				.WORD	CPUERR				; ADDRESS OF ERROR MESSAGE
6680											
6681	024464	022737	000000	172516	2#:	CMP	#0, #172516				; IS MMR3 CORRECT
6682	024472	001403				BEQ	3#				; YES GO ON
6683											; NO GO TO ERROR
6684	024474	104000				ERROR					; ALL ERRORS TO TRAP TO EMT VECTOR
6685	024476	000663				.WORD	663				; UNIQUE ERROR NUMBER
6686	024500	001127				.WORD	CPUERR				; ADDRESS OF ERROR MESSAGE



```

6743 024706 000002          8$: RTI          ;USER MODE HALT OCCURRED; GO TO ERROR.
6744
6745 024710 005037 177772    9$: CLR          @#177772      ;CLEAR PIRQ
6746 024714 005037 177776    CLR          @#177776      ;CLEAR PSW
6747 024720 010237 000004    MOV          R2,@#4        ;RESTORE VECTORS TO PREVIOUS STATE
6748 024724 010337 000006    MOV          R3,@#6        ;
6749 024730
6750
6751
6752 024730
6753
6754
6755
6756 024730
6757 024730 005267 154050    TST207:      INC          $TESTN      ;INCREMENT TEST NUMBER
6758
6759 024734 012705 000010    MOV          #8.,R5        ;INIT COUNTER
6760 024740 012701 025174    MOV          #T122B,R1     ;SETUP POINTER TO DATA
6761 024744 012737 030000 177776 1$: MOV          #30000,@#177776 ;INIT PSW
6762 024752 004767 000064    JSR          PC,T122A      ; TEST INSTRUCTION
6763 024756 022137 177776    CMP          (R1)+,@#177776 ;IS PSW CORRECT
6764 024762 001403          BEQ          2$            ;YES GO ON
6765
6766 024764 104000          ERROR        ;NO GO TO ERROR
6767 024766 000671          .WORD       671          ;ALL ERRORS TO TRAP TO EMT VECTOR
6768 024770 001127          .WORD       CPUERR      ;UNIQUE ERROR NUMBER
6769
6770 024772 077514          2$: SOB          R5,1$      ;REPEAT UNTIL ALL CASES ARE TESTED
6771
6772
6773 024774 012705 000010    MOV          #8.,R5        ;INIT COUNTER
6774 025000 012737 140000 177776 3$: MOV          #140000,@#177776 ;SETUP PSW TO USER MODE
6775 025006 012706 000600    MOV          #600,R6       ;SETUP USER STACK
6776 025012 004767 000024    JSR          PC,T122A      ; TEST INSTRUCTION
6777 025016 022737 140017 177776  CMP          #140017,@#177776 ;IS PSW CORRECT
6778 025024 001403          BEQ          4$            ;YES GO ON
6779
6780 025026 104000          ERROR        ;NO GO TO ERROR
6781 025030 000672          .WORD       672          ;ALL ERRORS TO TRAP TO EMT VECTOR
6782 025032 001127          .WORD       CPUERR      ;UNIQUE ERROR NUMBER
6783
6784 025034 077517          4$: SOB          R5,3$      ;REPEAT UNTIL ALL CASES ARE TESTED
6785
6786
6787 025036 000167 000152    JMP          FIN122
6788
6789 025042 020527 000010    T122A:      CMP          R5,#8.        ;FIND OUT WHAT COUNTER IS
6790 025046 001003          BNE          1$            ;IF NOT PRIORITY 0 GO TO 1$
6791 025050 000277          SCC          ;SET ALL CC BITS
6792 025052 000230          SPL          0            ;SET PRIORITY TO 0
6793 025054 000446          BR          8$            ;RETURN
6794 025056 020527 000007    1$: CMP          R5,#7      ;FIND OUT WHAT COUNTER IS
6795 025062 001003          BNE          2$            ;IF NOT PRIORITY 1 GO TO 2$
6796 025064 000277          SCC          ;SET ALL CC BITS
6797 025066 000231          SPL          1            ;SET PRIORITY TO 1
6798 025070 000440          BR          8$            ;RETURN

```

```

6799 025072 020527 000006      2$:  CMP      R5,#6      ;FIND OUT WHAT COUNTER IS
6800 025076 001003              BNE      3$          ;IF NOT PRIORITY 2 GO TO 3$
6801 025100 000277              SCC              ;SET ALL CC BITS
6802 025102 000232              SPL      2          ;SET PRIORITY TO 2
6803 025104 000432              BR       8$          ;RETURN
6804 025106 020527 000005      3$:  CMP      R5,#5      ;FIND OUT WHAT COUNTER IS
6805 025112 001003              BNE      4$          ;IF NOT PRIORITY 3 GO TO 4$
6806 025114 000277              SCC              ;SET ALL CC BITS
6807 025116 000233              SPL      3          ;SET PRIORITY TO 3
6808 025120 000424              BR       3$          ;RETURN
6809 025122 020527 000004      4$:  CMP      R5,#4      ;FIND OUT WHAT COUNTER IS
6810 025126 001003              BNE      5$          ;IF NOT PRIORITY 4 GO TO 5$
6811 025130 000277              SCC              ;SET ALL CC BITS
6812 025132 000234              SPL      4          ;SET PRIORITY TO 4
6813 025134 000416              BR       8$          ;RETURN
6814 025136 020527 000003      5$:  CMP      R5,#3      ;FIND OUT WHAT COUNTER IS
6815 025142 001003              BNE      6$          ;IF NOT PRIORITY 5 GO TO 6$
6816 025144 000277              SCC              ;SET ALL CC BITS
6817 025146 000235              SPL      5          ;SET PRIORITY TO 5
6818 025150 000410              BR       8$          ;RETURN
6819 025152 020527 000002      6$:  CMP      R5,#2      ;FIND OUT WHAT COUNTER IS
6820 025156 001003              BNE      7$          ;IF NOT PRIORITY 6 GO TO 7$
6821 025160 000277              SCC              ;SET ALL CC BITS
6822 025162 000236              SPL      6          ;SET PRIORITY TO 6
6823 025164 000402              BR       8$          ;RETURN
6824 025166 000277              7$:  SCC              ;SET ALL CC BITS
6825 025170 000237              SPL      7          ;SET PRIORITY TO 7
6826 025172 000207              8$:  RTS       PC      ;RETURN
6827
6828 025174 030017      Y122B: .WORD    30017
6829 025176 030057      .WORD    30057
6830 025200 030117      .WORD    30117
6831 025202 030157      .WORD    30157
6832 025204 030217      .WORD    30217
6833 025206 030257      .WORD    30257
6834 025210 030317      .WORD    30317
6835 025212 030357      .WORD    30357
6836 025214
6837
6838 025214      FIN122:
6839
6840      TE123:
6841      ;*****
6842      ;*TEST 210      TEST TSTSET INSTRUCTION (MULTI PROCESSING INST)
6843      ;*****
6844      TST210:
6845      INC      $TESTN      ;INCREMENT TEST NUMBER
6846      CLR      @177776     ;INIT PSW
6847      MOV      @10,R3      ;INIT COUNTER
6848      MOV      @400,R1     ;SETUP DESTINATION
6849      MOV      @T123A,R0   ;SETUP SOURCE
6850      MOV      (R0)+,(R1)+ ;RELOCATE TABLES
6851      SOB      R3,100$     ;ARE WE DONE
6852      MOV      @10,-(SP)   ;SAVE VECTOR
6853      MOV      @T123D,@10  ;SETUP NEW VECTOR
6854      CLR      R0          ;INIT R0
6855      MOV      @400,R1     ;SETUP POINTERS TO TABLES
6856      MOV      @410,R2

```

6855	025270	012703	000416		MOV	0415,R3			
6856	025274	010104			MOV	R1,R4			
6857	025276	012737	030000	177776	1\$:	MOV	030000,00177776		; SETUP PSW
6858	025304	000262				SEV			; SET V BIT
6859	025306	007221				.WORD	7221		; TEST INSTRUCTION
6860	025310	022237	177776			CMP	(R2)+,00177776		; IS PSW CORRECT
6861	025314	001403				BEQ	2\$		; YES GO ON
6862									; NO GO TO ERROR
6863	025316	104000				ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
6864	025320	000673				.WORD	673		; UNIQUE ERROR NUMBER
6865	025322	001127				.WORD	CPUERR		; ADDRESS OF ERROR MESSAGE
6866									
6867	025324	020013			2\$:	CMP	R0,(R3)		; IS R0 CORRECT
6868	025326	001403				BEQ	3\$		; YES GO ON
6869									; NO GO TO ERROR
6870	025330	104000				ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
6871	025332	000674				.WORD	674		; UNIQUE ERROR NUMBER
6872	025334	001127				.WORD	CPUERR		; ADDRESS OF ERROR MESSAGE
6873									
6874	025336	005204			3\$:	INC	R4		; SETUP EXPECTED DATA
6875	025340	005204				INC	R4		
6876	025342	020401				CMP	R4,R1		; IS R1 CORRECT
6877	025344	001403				BEQ	4\$		; YES GO ON
6878									; NO GO TO ERROR
6879	025346	104000				ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
6880	025350	000675				.WORD	675		; UNIQUE ERROR NUMBER
6881	025352	001127				.WORD	CPUERR		; ADDRESS OF ERROR MESSAGE
6882									
6883	025354	052713	000001		4\$:	BIS	01,(R3)		; SETUP EXPECTED DATA
6884	025360	022341				CMP	(R3)+,-(R1)		; IS TEST LOCATION CORRECT
6885	025362	001403				BEQ	5\$		; YES GO ON
6886									; NO GO TO ERROR
6887	025364	104000				ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
6888	025366	000676				.WORD	676		; UNIQUE ERROR NUMBER
6889	025370	001127				.WORD	CPUERR		; ADDRESS OF ERROR MESSAGE
6890									
6891	025372	005201			5\$:	INC	R1		; POINT TO NEXT TEST LOCATION
6892	025374	005201				INC	R1		
6893	025376	021127	177777			CMP	(R1),0177777		; ARE WE DONE
6894	025402	001335				BNE	1\$		; NO GO TO 1\$
6895	025404	012737	025460	000010		MOV	0T123E,0010		; SETUP NEW VECTOR
6896	025412	007201				.WORD	7201		; TEST INSTRUCTION ILLEGAL MODE
6897									; ERROR! GO TO ERROR IF DIDN'T TRAP
6898	025414	104000				ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
6899	025416	000677				.WORD	677		; UNIQUE ERROR NUMBER
6900	025420	001127				.WORD	CPUERR		; ADDRESS OF ERROR MESSAGE
6901									
6902	025422	000167	000036			JMP	T123F		
6903									
6904									
6905	025426	167604				T123A:	.WORD	167604	
6906	025430	000000					.WORD	0	
6907	025432	000001					.WORD	1	
6908	025434	177777					.WORD	177777	
6909	025436	030010				T123B:	.WORD	30010	
6910	025440	030004					.WORD	30004	

```

6911 025442 030001
6912 025444 167604
6913 025446 000000
6914 025450 000001
6915
6916 025452
6917 025452 104000
6918 025454 000700
6919 025456 001127
6920
6921 025460 005726
6922 025462 005726
6923 025464 012637 000010
6924
6925
6926 025470
6927
6928
6929
6930 025470
6931 025470 005267 153310
6932 025474 005037 177776
6933 025500 012703 000012
6934 025504 012701 000400
6935 025510 012700 025714
6936 025514 012021
6937 025516 077302
6938 025520 013746 000010
6939 025524 012737 025740 000010
6940 025532 012701 000400
6941 025536 012702 000410
6942 025542 012703 000416
6943 025546 010204
6944 025550 012737 030000 177776 1$:
6945 025556 011100
6946 025560 020327 000416
6947 025564 001401
6948 025566 000402
6949 025570 000261
6950 025572 000401
6951 025574 000241
6952 025576 000262
6953 025600 007322
6954 025602 022337 177776
6955 025606 001403
6956
6957 025610 104000
6958 025612 000701
6959 025614 001127
6960
6961 025616 021100
6962 025620 001403
6963
6964 025622 104000
6965 025624 000702
6966 025626 001127

T123C: .WORD 30001
        .WORD 167604
        .WORD 0
        .WORD 1

T123D:
        ERROR
        .WORD 700
        .WORD CPUERR

T123E: TST (SP)+
        TST (SP)+

T123F: MOV (SP)+,R010

;
T124:
;*****
;*TEST 211 TEST WRTLOCK (WRITE LOCK MULTI PROCESSING INST)
;*****
TST211:
        INC $TESTN ;INCREMENT TEST NUMBER
        CLW R0177776 ;INIT PSW
        MOV R10,R3 ;INIT COUNTER
        MOV R400,R1 ;SETUP DESTINATION
        MOV R124A,R0 ;SETUP SOURCE
100$: MOV (R0)+,(R1)+ ;RELOCATE TABLES
        SOB R3,100$ ;ARE WE DONE
        MOV R010,-(SP) ;SAVE VECTOR
        MOV R124D,R010 ;SETUP NEW VECTOR
        MOV R400,R1 ;SETUP POINTERS TO TABLES
        MOV R410,R2
        MOV R416,R3
        MOV R2,R4
        MOV R30000,R0177776 ;SETUP PSW
        MOV (R1),R0 ;SETUP R0
        CN R3,R416 ;IS THIS THE FIRST TEST CASE
        BR 2$ ;YES GO TO 2$
        BR 3$ ;NO GO TO 3$
2$: SEC ;SET C BIT
        BR 4$
4$: CLC ;CLEAR C BIT
        SEV ;SET V BIT
        ; TEST INSTRUCTION
        ;JS PSW CORRECT
        BEQ 5$ ;YES GO ON
        ;ERROR! NO GO TO ERROR
        ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
        .WORD 701 ;UNIQUE ERROR NUMBER
        .WORD CPUERR ;ADDRESS OF ERROR MESSAGE

5$: CMP (R1),R0 ;IS R0 CORRECT
        BEQ 6$ ;YES GO ON
        ;NO GO TO ERROR
        ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
        .WORD 702 ;UNIQUE ERROR NUMBER
        .WORD CPUERR ;ADDRESS OF ERROR MESSAGE

```

B11

```

6967
6968 025630 001204          60:  INC      R4          ;SETUP EXPECTED DATA
6969 025632 001204          INC      R4          |
6970 025634 020204          CMP      R2,R4       ;IS R2 CORRECT
6971 025636 001403          BEQ      70          ;YES GO ON
6972                                ;NO GO TO ERROR
6973 025640 104000          ERROR    ;ALL ERRORS TO TRAP TO ENT VECTOR
6974 025642 000703          .WORD   703        ;UNIQUE ERROR NUMBER
6975 025644 001127          .WORD   CPUERR     ;ADDRESS OF ERROR MESSAGE
6976
6977 025646 022142          70:  CMP      (R1)+,-(R2) ;IS TEST LOCATION CORRECT
6978 025650 001403          BEQ      80          ;YES GO ON
6979                                ;NO GO TO ERROR
6980 025652 104000          ERROR    ;ALL ERRORS TO TRAP TO ENT VECTOR
6981 025654 000704          .WORD   704        ;UNIQUE ERROR NUMBER
6982 025656 001127          .WORD   CPUERR     ;ADDRESS OF ERROR MESSAGE
6983
6984 025660 001202          80:  INC      R2          ;POINT TO NEXT TEST LOCATION
6985 025662 001202          INC      R2          |
6986 025664 021127 177777    CMP      (R1),#177777 ;ARE WE DONE
6987 025670 001327          BNE     10          ;NO GO TO 10
6988 025672 012737 025746 000010 MOV      #T124E,R#10 ;SETUP NEW VECTOR
6989 025700 007302          .WORD   7302       ;TEST INSTRUCTION ILLEGAL MODE
6990                                ;GO TO ERROR IF DIDN'T TRAP
6991 025702 104000          ERROR    ;ALL ERRORS TO TRAP TO ENT VECTOR
6992 025704 000705          .WORD   705        ;UNIQUE ERROR NUMBER
6993 025706 001127          .WORD   CPUERR     ;ADDRESS OF ERROR MESSAGE
6994
6995 025710 000167 000036    JMP      T124F
6996
6997
6998 025714 167604          T124A: .WORD   167604
6999 025716 000000          .WORD   0
7000 025720 000001          .WORD   1
7001 025722 177777          .WORD   177777
7002 025724 177777          T124B: .WORD   177777
7003 025726 177777          .WORD   177777
7004 025730 177777          .WORD   177777
7005 025732 030011          T124C: .WORD   30011
7006 025734 030004          .WORD   30004
7007 025736 030000          .WORD   30000
7008
7009 025740          T124D:                                ;GO TO ERROR IF TRAPPED
7010 025740 104000          ERROR    ;ALL ERRORS TO TRAP TO ENT VECTOR
7011 025742 000706          .WORD   706        ;UNIQUE ERROR NUMBER
7012 025744 001127          .WORD   CPUERR     ;ADDRESS OF ERROR MESSAGE
7013 025746 005726          T124E: TST      (SP)+ ;CLEAN UP STACK
7014 025750 005726          TST      (SP)+
7015 025752 012637 000010    T124F: MOV      (SP)+,R#10 ;RESTORE VECTOR
7016
7017
7018 025756          TE125:
7019
7020          ;*****
7021          ;*TEST 212      TEST MUL (MULTIPLY INST)
7022          ;*****
TST212:

```

7023	025756	005267	153022			INC	#TESTN		; INCREMENT TEST NUMBER
7024	025762	005037	177776			CLR	#177776		; INIT PS
7025	025766	012701	026266			MOV	#TE125A,R1		; SETUP POINTERS TO TABLES
7026									
7027	025772	010137	001034		1#:	MOV	R1,#EXPDAT		
7028	025776	062737	000002	001034		ADD	#2,#EXPDAT		; POINT TO SOURCE
7029	026004	012703	122222			MOV	#122222,R3		; INIT R3 TO A KNOWN STATE
7030	026010	011102				MOV	(R1),R2		; INIT DESTINATION REG
7031	026012	000277				SCC			; SET ALL CC BITS
7032	026014	070261	000002			MUL	2(R1),R2		; TEST INSTRUCTION
7033	026020	026137	000004	177776		CMP	4(R1),#177776		; IS PS CORRECT
7034	026026	001403				BEQ	2#		; YES GO ON
7035									; NO GO TO ERROR
7036	026030	104000				ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
7037	026032	000707				.WORD	707		; UNIQUE ERROR NUMBER
7038	026034	001127				.WORD	CPUERR		; ADDRESS OF ERROR MESSAGE
7039									
7040	026036	026103	000006		2#:	CMP	6(R1),R3		; IS R3 CORRECT
7041	026042	001403				BEQ	3#		; YES GO ON
7042									; NO GO TO ERROR
7043	026044	104000				ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
7044	026046	000710				.WORD	710		; UNIQUE ERROR NUMBER
7045	026050	001127				.WORD	CPUERR		; ADDRESS OF ERROR MESSAGE
7046									
7047	026052	026102	000010		3#:	CMP	10(R1),R2		; IS R2 CORRECT
7048	026056	001403				BEQ	4#		; YES GO ON
7049									; NO GO TO ERROR
7050	026060	104000				ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
7051	026062	000711				.WORD	711		; UNIQUE ERROR NUMBER
7052	026064	001127				.WORD	CPUERR		; ADDRESS OF ERROR MESSAGE
7053									
7054	026066	026177	000002	152740	4#:	CMP	2(R1),#EXPDAT		; IS SOURCE LOCATION OK
7055	026074	001403				BEQ	5#		; YES GO ON
7056									; NO GO TO ERROR
7057	026076	104000				ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
7058	026100	000712				.WORD	712		; UNIQUE ERROR NUMBER
7059	026102	001127				.WORD	CPUERR		; ADDRESS OF ERROR MESSAGE
7060									
7061	026104	062701	000012		5#:	ADD	#12,R1		; GO TO NEXT TEST
7062	026110	020127	026514			CMP	R1,#FIN125		; ARE WE FINISHED
7063	026114	001326				BNE	1#		; NO GO TO 1#
7064									
7065									
7066									
7067									
7068									
7069	026116	012701	026266		6#:	MOV	#TE125A,R1		; SETUP POINTERS TO TABLES
7070	026122				7#:				
7071									
7072	026122	010102				MOV	R1,R2		
7073	026124	012706	001000			MOV	#5180T,R6		; INIT R6 TO A KNOWN STATE
7074	026130	012704	000004			MOV	#4,R4		; SETUP R4 VALUE
7075	026134	011105				MOV	(R1),R5		; INIT DESTINATION REG
7076	026136	000277				SCC			; SET ALL CC BITS
7077	026140	070561	000002			MUL	2(R1),R5		; TEST INSTRUCTION
7078	026144	026137	000004	177776		CMP	4(R1),#177776		; IS PS CORRECT

; SECOND PART  
 ; USING ODD REGISTER  
 ;  
 6#:  
 7#:



```

7079 026152 001403          BEQ      8#          ; YES GO ON
7080                                ; NO GO TO ERROR
7081 026154 104000          ERROR
7082 026156 000713          .WORD   713          ; ALL ERRORS TO TRAP TO EMT VECTOR
7083 026160 001127          .WORD   CPUERR       ; UNIQUE ERROR NUMBER
7084                                ; ADDRESS OF ERROR MESSAGE
7085 026162 026105 000006    8#:     CMP      6(R1),R5   ; IS R5 CORRECT
7086 026166 001403          BEQ      9#          ; YES GO ON
7087                                ; NO GO TO ERROR
7088 026170 104000          ERROR
7089 026172 000714          .WORD   714          ; ALL ERRORS TO TRAP TO EMT VECTOR
7090 026174 001127          .WORD   CPUERR       ; UNIQUE ERROR NUMBER
7091                                ; ADDRESS OF ERROR MESSAGE
7092 026176 020627 001000    9#:     CMP      R6,#STBOT ; IS R6 CORRECT
7093 026202 001405          BEQ     10#         ; YES GO ON
7094 026204 012706 001000    MOV     #STBOT,R6    ; RESTORE SP
7095                                ; NO GO TO ERROR
7096 026210 104000          ERROR
7097 026212 000715          .WORD   715          ; ALL ERRORS TO TRAP TO EMT VECTOR
7098 026214 001127          .WORD   CPUERR       ; UNIQUE ERROR NUMBER
7099                                ; ADDRESS OF ERROR MESSAGE
7100 026216 005722          10#:    TST      (R2)+    ; POINT TO SOURCE OPERAND
7101 026220 021261 000002    CMP      (R2),2(R1)  ; IS SOURCE LOCATION OK
7102 026224 001403          BEQ     11#         ; YES GO ON
7103                                ; NO GO TO ERROR
7104 026226 104000          ERROR
7105 026230 000716          .WORD   716          ; ALL ERRORS TO TRAP TO EMT VECTOR
7106 026232 001127          .WORD   CPUERR       ; UNIQUE ERROR NUMBER
7107                                ; ADDRESS OF ERROR MESSAGE
7108 026234 020427 000004    11#:    CMP      R4,#4     ; IS R4 CORRECT
7109 026240 001403          BEQ     12#         ; YES GO ON
7110                                ; NO GO TO ERROR
7111 026242 104000          ERROR
7112 026244 000717          .WORD   717          ; ALL ERRORS TO TRAP TO EMT VECTOR
7113 026246 001127          .WORD   CPUERR       ; UNIQUE ERROR NUMBER
7114                                ; ADDRESS OF ERROR MESSAGE
7115 026250 062701 000012    12#:    ADD      #12,R1
7116 026254 020127 026514    CMP     R1,#FIN125  ; ARE WE FINISHED
7117 026260 001320          BNE     7#          ; NO GO TO 7#
7118
7119
7120 026262 000167 000226    JMP     FIN125
7121
7122
7123 026266 177777          ;
TE125A: .WORD   177777   ; MULTIPLICAND
7124 026270 177777          .WORD   177777   ; MULTIPLIER
7125 026272 000000          .WORD   0
7126 026274 000001          .WORD   1
7127 026276 000000          .WORD   0
7128
7129 026300 006772          .WORD   6772
7130 026302 100000          .WORD   100000
7131 026304 000011          .WORD   11
7132 026306 000000          .WORD   0
7133 026310 174403          .WORD   174403
7134

```

GLOBAL AREAS MACY11 30A(1052) 15-MAR-84 13:28 PAGE 134  
 KDJ11A.MAC 22-FEB-84 15:12 T212 TEST MUL (MULTIPLY INST)

SEQ 0134

7135	026312	177777	.WORD	177777	;MULTIPLICAND
7136	026314	077777	.WORD	77777	;MULTIPLIER
7137	026316	000010	.WORD	10	
7138	026320	100001	.WORD	100001	
7139	026322	177777	.WORD	177777	
7140					
7141	026324	077777	.WORD	77777	;MULTIPLICAND
7142	026326	000456	.WORD	456	;MULTIPLIER
7143	026330	000001	.WORD	1	
7144	026332	177322	.WORD	177322	
7145	026334	000226	.WORD	226	
7146					
7147	026336	173210	.WORD	173210	;MULTIPLICAND
7148	026340	000000	.WORD	0	;MULTIPLIER
7149	026342	000004	.WORD	4	
7150	026344	000000	.WORD	0	
7151	026346	000000	.WORD	0	
7152					
7153	026350	000000	.WORD	0	;MULTIPLICAND
7154	026352	003251	.WORD	3251	;MULTIPLIER
7155	026354	000004	.WORD	4	
7156	026356	000000	.WORD	0	
7157	026360	000000	.WORD	0	
7158					
7159	026362	000000	.WORD	0	;MULTIPLICAND
7160	026364	000000	.WORD	0	;MULTIPLIER
7161	026366	000004	.WORD	4	
7162	026370	000000	.WORD	0	
7163	026372	000000	.WORD	0	
7164					
7165	026374	100000	.WORD	100000	;MULTIPLICAND
7166	026376	000001	.WORD	1	;MULTIPLIER
7167	026400	000010	.WORD	10	
7168	026402	100000	.WORD	100000	
7169	026404	177777	.WORD	177777	
7170					
7171	026406	077777	.WORD	77777	;MULTIPLICAND
7172	026410	000001	.WORD	1	;MULTIPLIER
7173	026412	000000	.WORD	0	
7174	026414	077777	.WORD	77777	
7175	026416	000000	.WORD	0	
7176					
7177	026420	000010	.WORD	10	;MULTIPLICAND
7178	026422	010000	.WORD	10000	;MULTIPLIER
7179	026424	000001	.WORD	1	
7180	026426	100000	.WORD	100000	
7181	026430	000000	.WORD	0	
7182					
7183	026432	001452	.WORD	1452	;MULTIPLICAND
7184	026434	034527	.WORD	34527	;MULTIPLIER
7185	026436	000001	.WORD	1	
7186	026440	066506	.WORD	66506	
7187	026442	000265	.WORD	265	
7188					
7189	026444	000007	.WORD	7	;MULTIPLICAND
7190	026446	000400	.WORD	400	;MULTIPLIER

```

7191 026450 000000 .WORD 0
7192 026452 003400 .WORD 3400
7193 026454 000000 .WORD 0
7194
7195 026456 000002 .WORD 2 ;MULTIPLICAND
7196 026460 100000 .WORD 100000 ;MULTIPLIER
7197 026462 000011 .WORD 11
7198 026464 000000 .WORD 0
7199 026466 177777 .WORD 177777
7200
7201 026470 100000 .WORD 100000 ;MULTIPLICAND
7202 026472 077777 .WORD 77777 ;MULTIPLIER
7203 026474 000011 .WORD 11
7204 026476 100000 .WORD 100000
7205 026500 140000 .WORD 140000
7206
7207 026502 000001 .WORD 1 ;MULTIPLICAND
7208 026504 177777 .WORD 177777 ;MULTIPLIER
7209 026506 000010 .WORD 10
7210 026510 177777 .WORD 177777
7211 026512 177777 .WORD 177777
7212 026514
7213
7214 026514
7215
7216
7217
7218 026514
7219 026514 005267 152264
7220 026520 005037 177776
7221 026524 005006
7222 026526 013705 000000
7223 026532 013701 000002
7224 026536 012737 000137 000000
7225 026544 012737 026572 000002
7226 026552 000277
7227 026554 071627 000002
7228 026560 012706 001000
7229
7230 026564 104000
7231 026566 000720
7232 026570 001127
7233
7234 026572 022737 000000 177776
7235 026600 001405
7236 026602 012706 001000
7237
7238 026606 104000
7239 026610 000721
7240 026612 001127
7241
7242 026614 012704 026560
7243 026620 006204
7244 026622 020406
7245 026624 001405
7246 026626 012706 001000

```

```

FIN125:
;
;*****
;#TEST 213 TEST DIV (DIVIDE INST)
;*****
TST213:
INC $TESTN ;INCREMENT TEST NUMBER
CLR @177776 ;INIT PSW
CLR R6 ;INIT SP
MOV @0,R5 ;SAVE VECTORS
MOV @2,R1 ;
MOV @137,@0 ;SETUP NEW VECTORS
MOV @TE126A,@2 ;
;
;SET ALL CC BITS
; TEST INSTRUCTION
;RESTORE SP BEFORE GOING TO ERROR
;IF R7 ISN'T CORRECT GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
A126: MOV @STBOT,R6
;
;IS PS CORRECT
;YES GO ON
;RESTORE SP BEFORE GOING TO ERROR
;NO GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
1#: MOV @A126,R4 ;SETUP EXPECTED DATA
ASR R4 ;
CMP R4,R6 ;IS R6 CORRECT
BEQ 2# ;YES GO ON
MOV @STBOT,R6 ;RESTORE SP BEFORE GOING TO ERROR

```



7303	027016	001127			.WORD	CPUERR		:ADDRESS OF ERROR MESSAGE
7304								
7305	027020	012700	000004	8#:	MOV	#4,R0		:INIT R0
7306	027024	012705	000010		MOV	#10,R5		:INIT R5
7307	027030	005004			CLR	R4		:INIT R4
7308	027032	000277			SCC			:SET ALL CC BITS
7309	027034	071400			DIV	R0,R4		:TEST INSTRUCTION
7310	027036	022737	000000	177776	CMP	#0,#177776		:IS PS CORRECT
7311	027044	001407			BEQ	9#		:YES GO ON
7312	027046	010067	151326		MOV	R0,400		:SAVE R0
7313								:NO GO TO ERROR
7314	027052	104000			ERROR			:ALL ERRORS TO TRAP TO EMT VECTOR
7315	027054	000731			.WORD	731		:UNIQUE ERROR NUMBER
7316	027056	001127			.WORD	CPUERR		:ADDRESS OF ERROR MESSAGE
7317								
7318	027060	016700	151314		MOV	400,R0		:RESTORE R0
7319	027064	022700	000004	9#:	CMP	#4,R0		:IS R0 CORRECT
7320	027070	001403			BEQ	10#		:YES GO ON
7321								:NO GO TO ERROR
7322	027072	104000			ERROR			:ALL ERRORS TO TRAP TO EMT VECTOR
7323	027074	000732			.WORD	732		:UNIQUE ERROR NUMBER
7324	027076	001127			.WORD	CPUERR		:ADDRESS OF ERROR MESSAGE
7325								
7326	027100	022704	000002	10#:	CMP	#2,R4		:IS R4 CORRECT
7327	027104	001403			BEQ	11#		:YES GO ON
7328								:NO GO TO ERROR
7329	027106	104000			ERROR			:ALL ERRORS TO TRAP TO EMT VECTOR
7330	027110	000733			.WORD	733		:UNIQUE ERROR NUMBER
7331	027112	001127			.WORD	CPUERR		:ADDRESS OF ERROR MESSAGE
7332								
7333	027114	022705	000000	11#:	CMP	#0,R5		:IS R5 CORRECT
7334	027120	001403			BEQ	12#		:YES GO ON
7335								:NO GO TO ERROR
7336	027122	104000			ERROR			:ALL ERRORS TO TRAP TO EMT VECTOR
7337	027124	000734			.WORD	734		:UNIQUE ERROR NUMBER
7338	027126	001127			.WORD	CPUERR		:ADDRESS OF ERROR MESSAGE
7339								
7340	027130	012705	000010	12#:	MOV	#10,R5		:INIT R5
7341	027134	005004			CLR	R4		:INIT R4
7342	027136	000277			SCC			:SET ALL CC BITS
7343	027140	071427	000003		DIV	#3,R4		:TEST INSTRUCTION
7344	027144	022737	000000	177776	CMP	#0,#177776		:IS PS CORRECT
7345	027152	001403			BEQ	13#		:YES GO ON
7346								:NO GO TO ERROR
7347	027154	104000			ERROR			:ALL ERRORS TO TRAP TO EMT VECTOR
7348	027156	000735			.WORD	735		:UNIQUE ERROR NUMBER
7349	027160	001127			.WORD	CPUERR		:ADDRESS OF ERROR MESSAGE
7350								
7351	027162	022704	000002	13#:	CMP	#2,R4		:IS R4 CORRECT
7352	027166	001403			BEQ	14#		:YES GO ON
7353								:NO GO TO ERROR
7354	027170	104000			ERROR			:ALL ERRORS TO TRAP TO EMT VECTOR
7355	027172	000736			.WORD	736		:UNIQUE ERROR NUMBER
7356	027174	001127			.WORD	CPUERR		:ADDRESS OF ERROR MESSAGE
7357								
7358	027176	022705	000002	14#:	CMP	#2,R5		:IS R5 CORRECT

7359	027202	001403		BEQ	15#		; YES GO ON
7360							; NO GO TO ERROR
7361	027204	104000		ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
7362	027206	000737		.WORD	737		; UNIQUE ERROR NUMBER
7363	027210	001127		.WORD	CPUERR		; ADDRESS OF ERROR MESSAGE
7364							
7365							
7366							
7367							
7368	027212	012701	027366	15#:	MOV	#TE126B,R1	; SETUP POINTERS TO TABLES
7369							
7370	027216	010137	001034	16#:	MOV	R1,#EXPDAT	; SAVE A COPY OF R1
7371	027222	011104			MOV	(R1),R4	; INIT R4
7372	027224	016103	000004		MOV	4(R1),R3	; SAVE SOURCE
7373	027230	016105	000002		MOV	2(R1),R5	; INIT R5
7374	027234	000277			SCC		; SET ALL CC BITS
7375	027236	071461	000004		DIV	4(R1),R4	; TEST INSTRUCTION
7376	027242	026137	000006	177776	CMP	6(R1),#177776	; IS PS CORRECT
7377	027250	001403			BEQ	17#	; YES GO ON
7378							; NO GO TO ERROR
7379	027252	104000		ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
7380	027254	000740		.WORD	740		; UNIQUE ERROR NUMBER
7381	027256	001127		.WORD	CPUERR		; ADDRESS OF ERROR MESSAGE
7382							
7383	027260	026105	000010	17#:	CMP	10(R1),R5	; IS R5 CORRECT
7384	027264	001403			BEQ	18#	; YES GO ON
7385							; NO GO TO ERROR
7386	027266	104000		ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
7387	027270	000741		.WORD	741		; UNIQUE ERROR NUMBER
7388	027272	001127		.WORD	CPUERR		; ADDRESS OF ERROR MESSAGE
7389							
7390	027274	026104	000012	18#:	CMP	12(R1),R4	; IS R4 CORRECT
7391	027300	001403			BEQ	19#	; YES GO ON
7392							; NO GO TO ERROR
7393	027302	104000		ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
7394	027304	000742		.WORD	742		; UNIQUE ERROR NUMBER
7395	027306	001127		.WORD	CPUERR		; ADDRESS OF ERROR MESSAGE
7396							
7397	027310	023701	001034	19#:	CMP	#EXPDAT,R1	; IS R1 CORRECT
7398	027314	001405			BEQ	20#	; YES GO ON
7399							; NO GO TO ERROR
7400	027316	104000		ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
7401	027320	000743		.WORD	743		; UNIQUE ERROR NUMBER
7402	027322	001127		.WORD	CPUERR		; ADDRESS OF ERROR MESSAGE
7403							
7404	027324	013701	001034	20#:	MOV	#EXPDAT,R1	; RESTORE CORRECT VALUE
7405	027330	026103	000004		CMP	4(R1),R3	; IS SOURCE CORRECT
7406	027334	001405			BEQ	21#	; YES GO ON
7407							; NO GO TO ERROR
7408	027336	104000		ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
7409	027340	000744		.WORD	744		; UNIQUE ERROR NUMBER
7410	027342	001127		.WORD	CPUERR		; ADDRESS OF ERROR MESSAGE
7411							
7412	027344	010361	000004	21#:	MOV	R3,4(R1)	; TRY TO RESTORE CODE
7413	027350	062701	000014		ADD	#14,R1	; POINT TO NEXT LOCATION
7414	027354	021127	000333		CMP	(R1),#333	; ARE WE DONE

```

7415 027360 001316      BNE      16$          ;NO GO TO 16$
7416
7417
7418 027362 000167 000316      JMP      FIN126
7419
7420
7421 027366 177777      ;E126B: .WORD 177777 ;DIVIDEND
7422 027370 177777      .WORD 177777 ;INIT R5
7423 027372 177777      .WORD 177777 ;DIVISOR
7424 027374 000000      .WORD 0      ;PSW
7425 027376 000000      .WORD 0      ;R5 RESULT
7426 027400 000001      .WORD 1      ;R4 RESULT
7427
7428 027402 000000      .WORD 0      ;DIVIDEND
7429 027404 177777      .WORD 177777 ;INIT R5
7430 027406 177777      .WORD 177777 ;DIVISOR
7431 027410 000012      .WORD 12     ;PSW
7432 027412 177777      .WORD 177777 ;R5 RESULT
7433 027414 000000      .WORD 0      ;R4 RESULT
7434
7435 027416 177777      .WORD 177777 ;DIVIDEND
7436 027420 000000      .WORD 0      ;INIT R5
7437 027422 177777      .WORD 177777 ;DIVISOR
7438 027424 000002      .WORD 2      ;PSW
7439 027426 000000      .WORD 0      ;R5 RESULT
7440 027430 177777      .WORD 177777 ;R4 RESULT
7441
7442 027432 000000      .WORD 0      ;DIVIDEND
7443 027434 007642      .WORD 7642   ;INIT R5
7444 027436 007643      .WORD 7643   ;DIVISOR
7445 027440 000004      .WORD 4      ;PSW
7446 027442 007642      .WORD 7642   ;R5 RESULT
7447 027444 000000      .WORD 0      ;R4 RESULT
7448
7449 027446 000000      .WORD 0      ;DIVIDEND
7450 027450 000137      .WORD 137    ;INIT R5
7451 027452 177543      .WORD 177543 ;DIVISOR
7452 027454 000004      .WORD 4      ;PSW
7453 027456 000137      .WORD 137    ;R5 RESULT
7454 027460 000000      .WORD 0      ;R4 RESULT
7455
7456 027462 000000      .WORD 0      ;DIVIDEND
7457 027464 007643      .WORD 7643   ;INIT R5
7458 027466 007643      .WORD 7643   ;DIVISOR
7459 027470 000000      .WORD 0      ;PSW
7460 027472 000000      .WORD 0      ;R5 RESULT
7461 027474 000001      .WORD 1      ;R4 RESULT
7462
7463 027476 100000      .WORD 100000 ;DIVIDEND
7464 027500 004376      .WORD 4376   ;INIT R5
7465 027502 010021      .WORD 10021  ;DIVISOR
7466 027504 000012      .WORD 12     ;PSW
7467 027506 004376      .WORD 4376   ;R5 RESULT
7468 027510 100000      .WORD 100000 ;R4 RESULT
7469
7470 027512 177700      .WORD 177700 ;DIVIDEND

```

7471	027514	170033	.WORD	170033	;INIT R5
7472	027516	010021	.WORD	10021	;DIVISOR
7473	027520	000010	.WORD	10	;PSW
7474	027522	171307	.WORD	171307	;R5 RESULT
7475	027524	176024	.WORD	176024	;R4 RESULT
7476					
7477	027526	177700	.WORD	177700	;DIVIDEND
7478	027530	170033	.WORD	170033	;INIT R5
7479	027532	167757	.WORD	167757	;DIVISOR
7480	027534	000000	.WORD	0	;PSW
7481	027536	171307	.WORD	171307	;R5 RESULT
7482	027540	001754	.WORD	1754	;R4 RESULT
7483					
7484	027542	000000	.WORD	0	;DIVIDEND
7485	027544	177777	.WORD	177777	;INIT R5
7486	027546	000001	.WORD	1	;DIVISOR
7487	027550	000002	.WORD	2	;PSW
7488	027552	177777	.WORD	177777	;R5 RESULT
7489	027554	000000	.WORD	0	;R4 RESULT
7490					
7491	027556	177777	.WORD	177777	;DIVIDEND
7492	027560	045716	.WORD	45716	;INIT R5
7493	027562	000001	.WORD	1	;DIVISOR
7494	027564	000012	.WORD	12	;PSW
7495	027566	045716	.WORD	45716	;R5 RESULT
7496	027570	177777	.WORD	177777	;R4 RESULT
7497					
7498	027572	000000	.WORD	0	;DIVIDEND
7499	027574	000002	.WORD	2	;INIT R5
7500	027576	177770	.WORD	177770	;DIVISOR
7501	027600	000004	.WORD	4	;PSW
7502	027602	000002	.WORD	2	;R5 RESULT
7503	027604	000000	.WORD	0	;R4 RESULT
7504					
7505	027606	177777	.WORD	177777	;DIVIDEND
7506	027610	177776	.WORD	177776	;INIT R5
7507	027612	000010	.WORD	10	;DIVISOR
7508	027614	000004	.WORD	4	;PSW
7509	027616	177776	.WORD	177776	;R5 RESULT
7510	027620	000000	.WORD	0	;R4 RESULT
7511					
7512	027622	000001	.WORD	1	;DIVIDEND
7513	027624	177777	.WORD	177777	;INIT R5
7514	027626	000001	.WORD	1	;DIVISOR
7515	027630	000002	.WORD	2	;PSW
7516	027632	177777	.WORD	177777	;R5 RESULT
7517	027634	000001	.WORD	1	;R4 RESULT
7518					
7519	027636	000001	.WORD	1	;DIVIDEND
7520	027640	000000	.WORD	0	;INIT R5
7521	027642	000002	.WORD	2	;DIVISOR
7522	027644	000002	.WORD	2	;PSW
7523	027646	000000	.WORD	0	;R5 RESULT
7524	027650	000001	.WORD	1	;R4 RESULT
7525					
7526	027652	000001	.WORD	1	;DIVIDEND



```

7527 027654 000000 .WORD 0 ;INIT R5
7528 027656 000003 .WORD 3 ;DIVISOR
7529 027660 000000 .WORD 0 ;PSW
7530 027662 000001 .WORD 1 ;R5 RESULT
7531 027664 052525 .WORD 52525 ;R4 RESULT
7532
7533 027666 000023 .WORD 23 ;DIVIDEND
7534 027670 016054 .WORD 16054 ;INIT R5
7535 027672 016537 .WORD 16537 ;DIVISOR
7536 027674 000000 .WORD 0 ;PSW
7537 027676 010222 .WORD 10222 ;R5 RESULT
7538 027700 000246 .WORD 246 ;R4 RESULT
7539
7540 027702 000333 .WORD 333
7541 027704
7542
7543 027704
7544
7545
7546
7547 027704
7548 027704 005267 151074
7549 027710 005037 177776
7550 027714 012702 000001
7551 027720 000277
7552 027722 072202
7553 027724 022737 000000 177776
7554 027732 001403
7555
7556 027734 104000
7557 027736 000745
7558 027740 001127
7559
7560 027742 020227 000002 1#:
7561 027746 001403
7562
7563 027750 104000
7564 027752 000746
7565 027754 001127
7566
7567 027756 012702 100000 2#:
7568 027762 012703 000001
7569 027766 000257
7570 027770 072203
7571 027772 022737 000007 177776
7572 030000 001403
7573
7574 030002 104000
7575 030004 000747
7576 030006 001127
7577
7578 030010 020327 000001 3#:
7579 030014 001403
7580
7581 030016 104000
7582 030020 000750

```

```

FIN126:
;
TE127:
;*****
; *TEST 214 TEST ASH (ARITHMETIC SHIFT)
;*****
TST214:
INC #TESTN ;INCREMENT TEST NUMBER
CLR #177776 ;INIT PSW
MOV #1,R2 ;SETUP OPERAND
SCC ;SET ALL CC BITS
ASH R2,R2 ;TEST INSTRUCTION
CMP #0,#177776 ;IS PS CORRECT
BEQ 1# ;YES GO ON
;NO GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
1#:
CMP R2,#2 ;IS R2 CORRECT
BEQ 2# ;YES GO ON
;NO GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
2#:
MOV #100000,R2 ;SETUP R2
MOV #1,R3 ;SETUP R3
CCC ;CLEAR ALL CC BITS
ASH R3,R2 ;TEST INSTRUCTION
CMP #7,#177776 ;IS PS CORRECT
BEQ 3# ;YES GO ON
;NO GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
3#:
CMP R3,#1 ;IS R3 CORRECT
BEQ 4# ;YES GO ON
;NO GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
4#:

```

7583	030022	001127			.WORD	CPUERR			ADDRESS OF ERROR MESSAGE
7584									
7585	030024	020227	000000	4\$:	CMP	R2,#0			IS R2 CORRECT
7586	030030	001403			BEQ	5\$			YES GO ON
7587									NO GO TO ERROR
7588	030032	104000			ERROR				ALL ERRORS TO TRAP TO EMT VECTOR
7589	030034	000751			.WORD	751			UNIQUE ERROR NUMBER
7590	030036	001127			.WORD	CPUERR			ADDRESS OF ERROR MESSAGE
7591									
7592	030040	012701	030154	5\$:	MOV	#TE127A,R1			SETUP POINTERS TO TABLES
7593									
7594	030044	010103		6\$:	MOV	R1,R3			
7595	030046	016102	000002		MOV	2(R1),R2			SETUP R2
7596	030052	000277			SCC				SET ALL CC BITS
7597	030054	072211			ASH	(R1),R2			TEST INSTRUCTION
7598	030056	026137	000004	177776	CMP	4(R1),#0177776			IS PS CORRECT
7599	030064	001403			BEQ	7\$			YES GO ON
7600									NO GO TO ERROR
7601	030066	104000			ERROR				ALL ERRORS TO TRAP TO EMT VECTOR
7602	030070	000752			.WORD	752			UNIQUE ERROR NUMBER
7603	030072	001127			.WORD	CPUERR			ADDRESS OF ERROR MESSAGE
7604									
7605	030074	026102	000006	7\$:	CMP	6(R1),R2			IS R2 CORRECT
7606	030100	001403			BEQ	8\$			YES GO ON
7607									NO GO TO ERROR
7608	030102	104000			ERROR				ALL ERRORS TO TRAP TO EMT VECTOR
7609	030104	000753			.WORD	753			UNIQUE ERROR NUMBER
7610	030106	001127			.WORD	CPUERR			ADDRESS OF ERROR MESSAGE
7611									
7612	030110	020301		8\$:	CMP	R3,R1			IS R1 CORRECT
7613	030112	001404			BEQ	9\$			YES GO ON
7614									NO GO TO ERROR
7615	030114	104000			ERROR				ALL ERRORS TO TRAP TO EMT VECTOR
7616	030116	000754			.WORD	754			UNIQUE ERROR NUMBER
7617	030120	001127			.WORD	CPUERR			ADDRESS OF ERROR MESSAGE
7618	030122	010301			MOV	R3,R1			RESTORE R1
7619									
7620	030124	021311		9\$:	CMP	(R3),(R1)			IS SOURCE CORRECT
7621	030126	001403			BEQ	10\$			YES GO ON
7622									NO GO TO ERROR
7623	030130	104000			ERROR				ALL ERRORS TO TRAP TO EMT VECTOR
7624	030132	000755			.WORD	755			UNIQUE ERROR NUMBER
7625	030134	001127			.WORD	CPUERR			ADDRESS OF ERROR MESSAGE
7626									SOURCE LOOKS INCORRECT
7627	030136	062701	000010	10\$:	ADD	#10,R1			INCREMENT POINTER
7628	030142	020127	030414		CMP	R1,#FIN127			ARE WE DONE
7629	030146	001336			BNE	6\$			NO GO TO 6\$
7630									
7631									
7632	030150	000167	000240		JMP	FIN127			
7633									
7634									
7635	030154	177761			TE127A: .WORD	177761			SOURCE
7636	030156	077777			.WORD	77777			DEST
7637	030160	000005			.WORD	5			
7638	030162	000000			.WORD	0			

N11

GLOBAL AREAS MACY11 30A(1052) 15-MAR-84 13:28 PAGE 143  
KD.J11A.MAC 22-FEB-84 15:12 T214 TEST ASH (ARITHMETIC SHIFT)

SEQ 0143

7639					
7640	030164	177700	.WORD	177700	;SOURCE
7641	030166	017777	.WORD	17777	;DEST
7642	030170	000000	.WORD	0	
7643	030172	017777	.WORD	17777	
7644					
7645	030174	177700	.WORD	177700	;SOURCE
7646	030176	100000	.WORD	100000	;DEST
7647	030200	000010	.WORD	10	
7648	030202	100000	.WORD	100000	
7649					
7650	030204	177777	.WORD	177777	;SOURCE
7651	030206	100000	.WORD	100000	;DEST
7652	030210	000010	.WORD	10	
7653	030212	140000	.WORD	140000	
7654					
7655	030214	177737	.WORD	177737	;SOURCE
7656	030216	177777	.WORD	177777	;DEST
7657	030220	000011	.WORD	11	
7658	030222	177777	.WORD	177777	
7659					
7660	030224	177706	.WORD	177706	;SOURCE
7661	030226	102000	.WORD	102000	;DEST
7662	030230	000007	.WORD	7	
7663	030232	000000	.WORD	0	
7664					
7665	030234	177710	.WORD	177710	;SOURCE
7666	030236	017777	.WORD	17777	;DEST
7667	030240	000013	.WORD	13	
7668	030242	177400	.WORD	177400	
7669					
7670	030244	177713	.WORD	177713	;SOURCE
7671	030246	000012	.WORD	12	;DEST
7672	030250	000000	.WORD	0	
7673	030252	050000	.WORD	50000	
7674					
7675	030254	177707	.WORD	177707	;SOURCE
7676	030256	170001	.WORD	170001	;DEST
7677	030260	000002	.WORD	2	
7678	030262	000200	.WORD	200	
7679					
7680	030264	177717	.WORD	177717	;SOURCE
7681	030266	000001	.WORD	1	;DEST
7682	030270	000012	.WORD	12	
7683	030272	100000	.WORD	100000	
7684					
7685	030274	177740	.WORD	177740	;SOURCE
7686	030276	017777	.WORD	17777	;DEST
7687	030300	000004	.WORD	4	
7688	030302	000000	.WORD	0	
7689					
7690	030304	177771	.WORD	177771	;SOURCE
7691	030306	150000	.WORD	150000	;DEST
7692	030310	000010	.WORD	10	
7693	030312	177640	.WORD	177640	
7694					

7695	030314	177742	.WORD	177742	!SOURCE
7696	030316	100000	.WORD	100000	!DEST
7697	030320	000011	.WORD	11	
7698	030322	177777	.WORD	177777	
7699					
7700	030324	177764	.WORD	177764	!SOURCE
7701	030326	100000	.WORD	100000	!DEST
7702	030330	000010	.WORD	10	
7703	030332	177770	.WORD	177770	
7704					
7705	030334	177750	.WORD	177750	!SOURCE
7706	030336	052525	.WORD	52525	!DEST
7707	030340	000004	.WORD	4	
7708	030342	000000	.WORD	0	
7709					
7710	030344	177760	.WORD	177760	!SOURCE
7711	030346	100000	.WORD	100000	!DEST
7712	030350	000011	.WORD	11	
7713	030352	177777	.WORD	177777	
7714					
7715	030354	177770	.WORD	177770	!SOURCE
7716	030356	100000	.WORD	100000	!DEST
7717	030360	000010	.WORD	10	
7718	030362	177600	.WORD	177600	
7719					
7720	030364	177712	.WORD	177712	!SOURCE
7721	030366	004367	.WORD	4367	!DEST
7722	030370	000013	.WORD	13	
7723	030372	156000	.WORD	156000	
7724					
7725	030374	177764	.WORD	177764	!SOURCE
7726	030376	017777	.WORD	17777	!DEST
7727	030400	000001	.WORD	1	
7728	030402	000001	.WORD	1	
7729					
7730	030404	177701	.WORD	177701	!SOURCE
7731	030406	110000	.WORD	110000	!DEST
7732	030410	000003	.WORD	3	
7733	030412	020000	.WORD	20000	
7734					

FIN127:

!TE130:

!\*\*\*\*\*  
!TEST 215 TEST ASHC (ARITHMETIC SHIFT COMBINED)  
!\*\*\*\*\*

!TST215:

7741	030414								
7742	030414	005267	150364	INC	!TESTN	!INCREMENT TEST NUMBER			
7743	030420	005037	177776	CLR	B#177776	!INIT PSW			
7744	030424	012701	000023	MOV	#23,R1	!SETUP R1			
7745	030430	012705	052525	MOV	#52525,R5	!SETUP R5			
7746	030434	005004		CLR	R4	!SETUP R4			
7747	030436	000277		SCC		!SET ALL CC BITS			
7748	030440	073401		ASHC	R1,R4	!TEST INSTRUCTION			
7749	030442	023727	177776 000012	CHP	B#177776,#12	!IS PS CORRECT			
7750	030450	001403		BEQ	11	!YES GO ON			



7807	030622	000765				.WORD	765		;UNIQUE ERROR NUMBER
7808	030624	001127				.WORD	CPUERR		;ADDRESS OF ERROR MESSAGE
7809									
7810									
7811	030626	012701	030762	8#:	MOV		#TE130A,R1		;SETUP POINTERS TO TABLES
7812									
7813	030632	010104		9#:	MOV		R1,R4		;SAVE A COPY OF R1
7814	030634	016102	000002		MOV		2(R1),R2		;SETUP R2
7815	030640	016103	000004		MOV		4(R1),R3		;SETUP R3
7816	030644	000277			SCC				;SET ALL CC BITS
7817	030646	073211			ASHC		(R1),R2		;TEST INSTRUCTION
7818	030650	023761	177776	000006	CMP		#177776,6(R1)		;IS PS CORRECT
7819	030656	001403			BEQ		10#		;YES GO ON
7820									;NO GO TO ERROR
7821	030660	104000			ERROR				;ALL ERRORS TO TRAP TO EMT VECTOR
7822	030662	000766			.WORD		766		;UNIQUE ERROR NUMBER
7823	030664	001127			.WORD		CPUERR		;ADDRESS OF ERROR MESSAGE
7824									
7825	030666	026102	000010	10#:	CMP		10(R1),R2		;IS R2 CORRECT
7826	030672	001403			BEQ		11#		;YES GO ON
7827									;NO GO TO ERROR
7828	030674	104000			ERROR				;ALL ERRORS TO TRAP TO EMT VECTOR
7829	030676	000767			.WORD		767		;UNIQUE ERROR NUMBER
7830	030700	001127			.WORD		CPUERR		;ADDRESS OF ERROR MESSAGE
7831									
7832	030702	026103	000012	11#:	CMP		12(R1),R3		;IS R3 CORRECT
7833	030706	001403			BEQ		12#		;YES GO ON
7834									;NO GO TO ERROR
7835	030710	104000			ERROR				;ALL ERRORS TO TRAP TO EMT VECTOR
7836	030712	000770			.WORD		770		;UNIQUE ERROR NUMBER
7837	030714	001127			.WORD		CPUERR		;ADDRESS OF ERROR MESSAGE
7838									
7839	030716	020401		12#:	CMP		R4,R1		;IS R1 CORRECT
7840	030720	001404			BEQ		13#		;YES GO ON
7841									;NO GO TO ERROR
7842	030722	104000			ERROR				;ALL ERRORS TO TRAP TO EMT VECTOR
7843	030724	000771			.WORD		771		;UNIQUE ERROR NUMBER
7844	030726	001127			.WORD		CPUERR		;ADDRESS OF ERROR MESSAGE
7845	030730	010401			MOV		R4,R1		
7846									
7847	030732	021114		13#:	CMP		(R1),(R4)		;IS SOURCE CORRECT
7848	030734	001403			BEQ		14#		;YES GO ON
7849									;NO GO TO ERROR
7850	030736	104000			ERROR				;ALL ERRORS TO TRAP TO EMT VECTOR
7851	030740	000772			.WORD		772		;UNIQUE ERROR NUMBER
7852	030742	001127			.WORD		CPUERR		;ADDRESS OF ERROR MESSAGE
7853									;POSSIBLE SOURCE CODE CORRUPTION
7854	030744	062701	000014	14#:	ADD		#14,R1		;GO TO NEXT TEST
7855	030750	020127	031372		CMP		R1,#FIN130		;ARE WE DONE
7856	030754	001326			BNE		9#		;NO GO TO 9#
7857									
7858									
7859	030756	000167	000410		JMP		FIN130		
7860									
7861									
7862	030762	177700			TE130A:	.WORD	177700		;SOURCE

GLOBAL AREAS MACY11 30A(1052) 15-MAR-84 13:28 PAGE 147  
 KDJ11A.MAC 22-FEB-84 15:12 T215 TEST ASHC (ARITHMETIC SHIFT COMBINED)

SEQ 0147

7863	030764	100125	.WORD	100125	; DESTINATION WORD 1
7864	030766	177777	.WORD	177777	; DESTINATION WORD 2
7865	030770	000010	.WORD	10	; TEST PSW
7866	030772	100125	.WORD	100125	; RESULT WORD 1
7867	030774	177777	.WORD	177777	; RESULT WORD 2
7868					
7869	030776	177777	.WORD	177777	; SOURCE
7870	031000	000001	.WORD	1	; DESTINATION WORD 1
7871	031002	000000	.WORD	0	; DESTINATION WORD 2
7872	031004	000000	.WORD	0	; TEST PSW
7873	031006	000000	.WORD	0	; RESULT WORD 1
7874	031010	100000	.WORD	100000	; RESULT WORD 2
7875					
7876	031012	177701	.WORD	177701	; SOURCE
7877	031014	047777	.WORD	47777	; DESTINATION WORD 1
7878	031016	100000	.WORD	100000	; DESTINATION WORD 2
7879	031020	000012	.WORD	12	; TEST PSW
7880	031022	117777	.WORD	117777	; RESULT WORD 1
7881	031024	000000	.WORD	0	; RESULT WORD 2
7882					
7883	031026	177706	.WORD	177706	; SOURCE
7884	031030	004256	.WORD	4256	; DESTINATION WORD 1
7885	031032	177700	.WORD	177700	; DESTINATION WORD 2
7886	031034	000002	.WORD	2	; TEST PSW
7887	031036	025677	.WORD	25677	; RESULT WORD 1
7888	031040	170000	.WORD	170000	; RESULT WORD 2
7889					
7890	031042	177711	.WORD	177711	; SOURCE
7891	031044	065700	.WORD	65700	; DESTINATION WORD 1
7892	031046	000012	.WORD	12	; DESTINATION WORD 2
7893	031050	000013	.WORD	13	; TEST PSW
7894	031052	100000	.WORD	100000	; RESULT WORD 1
7895	031054	012000	.WORD	12000	; RESULT WORD 2
7896					
7897	031056	177737	.WORD	177737	; SOURCE
7898	031060	000000	.WORD	0	; DESTINATION WORD 1
7899	031062	000001	.WORD	1	; DESTINATION WORD 2
7900	031064	000004	.WORD	4	; TEST PSW
7901	031066	000000	.WORD	0	; RESULT WORD 1
7902	031070	000000	.WORD	0	; RESULT WORD 2
7903					
7904	031072	177736	.WORD	177736	; SOURCE
7905	031074	000000	.WORD	0	; DESTINATION WORD 1
7906	031076	000001	.WORD	1	; DESTINATION WORD 2
7907	031100	000000	.WORD	0	; TEST PSW
7908	031102	040000	.WORD	40000	; RESULT WORD 1
7909	031104	000000	.WORD	0	; RESULT WORD 2
7910					
7911	031106	177740	.WORD	177740	; SOURCE
7912	031110	100000	.WORD	100000	; DESTINATION WORD 1
7913	031112	000000	.WORD	0	; DESTINATION WORD 2
7914	031114	000011	.WORD	11	; TEST PSW
7915	031116	177777	.WORD	177777	; RESULT WORD 1
7916	031120	177777	.WORD	177777	; RESULT WORD 2
7917					
7918	031122	177725	.WORD	177725	; SOURCE

7919	031124	177777	.WORD	177777	; DESTINATION WORD 1
7920	031126	174000	.WORD	174000	; DESTINATION WORD 2
7921	031130	000007	.WORD	7	; TEST PSW
7922	031132	000000	.WORD	0	; RESULT WORD 1
7923	031134	000000	.WORD	0	; RESULT WORD 2
7924					
7925	031136	177724	.WORD	177724	; SOURCE
7926	031140	177777	.WORD	177777	; DESTINATION WORD 1
7927	031142	174000	.WORD	174000	; DESTINATION WORD 2
7928	031144	000011	.WORD	11	; TEST PSW
7929	031146	100000	.WORD	100000	; RESULT WORD 1
7930	031150	000000	.WORD	0	; RESULT WORD 2
7931					
7932	031152	177733	.WORD	177733	; SOURCE
7933	031154	177777	.WORD	177777	; DESTINATION WORD 1
7934	031156	157023	.WORD	157023	; DESTINATION WORD 2
7935	031160	000012	.WORD	12	; TEST PSW
7936	031162	114000	.WORD	114000	; RESULT WORD 1
7937	031164	000000	.WORD	0	; RESULT WORD 2
7938					
7939	031166	177727	.WORD	177727	; SOURCE
7940	031170	000000	.WORD	0	; DESTINATION WORD 1
7941	031172	177777	.WORD	177777	; DESTINATION WORD 2
7942	031174	000013	.WORD	13	; TEST PSW
7943	031176	177600	.WORD	177600	; RESULT WORD 1
7944	031200	000000	.WORD	0	; RESULT WORD 2
7945					
7946	031202	177717	.WORD	177717	; SOURCE
7947	031204	177777	.WORD	177777	; DESTINATION WORD 1
7948	031206	000001	.WORD	1	; DESTINATION WORD 2
7949	031210	000011	.WORD	11	; TEST PSW
7950	031212	100000	.WORD	100000	; RESULT WORD 1
7951	031214	100000	.WORD	100000	; RESULT WORD 2
7952					
7953	031216	177741	.WORD	177741	; SOURCE
7954	031220	100000	.WORD	100000	; DESTINATION WORD 1
7955	031222	000000	.WORD	0	; DESTINATION WORD 2
7956	031224	000010	.WORD	10	; TEST PSW
7957	031226	177777	.WORD	177777	; RESULT WORD 1
7958	031230	177777	.WORD	177777	; RESULT WORD 2
7959					
7960	031232	177742	.WORD	177742	; SOURCE
7961	031234	037777	.WORD	37777	; DESTINATION WORD 1
7962	031236	177777	.WORD	177777	; DESTINATION WORD 2
7963	031240	000005	.WORD	5	; TEST PSW
7964	031242	000000	.WORD	0	; RESULT WORD 1
7965	031244	000000	.WORD	0	; RESULT WORD 2
7966					
7967	031246	177742	.WORD	177742	; SOURCE
7968	031250	077777	.WORD	77777	; DESTINATION WORD 1
7969	031252	177777	.WORD	177777	; DESTINATION WORD 2
7970	031254	000001	.WORD	1	; TEST PSW
7971	031256	000000	.WORD	0	; RESULT WORD 1
7972	031260	000001	.WORD	1	; RESULT WORD 2
7973					
7974	031262	177711	.WORD	177711	; SOURCE



7975	031264	065600	.WORD	65600	;DESTINATION WORD 1
7976	031266	000012	.WORD	12	;DESTINATION WORD 2
7977	031270	000003	.WORD	3	;TEST PSW
7978	031272	000000	.WORD	0	;RESULT WORD 1
7979	031274	012000	.WORD	12000	;RESULT WORD 2
7980					
7981	031276	177740	.WORD	177740	;SOURCE
7982	031300	077777	.WORD	77777	;DESTINATION WORD 1
7983	031302	177777	.WORD	177777	;DESTINATION WORD 2
7984	031304	000004	.WORD	4	;TEST PSW
7985	031306	000000	.WORD	0	;RESULT WORD 1
7986	031310	000000	.WORD	0	;RESULT WORD 2
7987					
7988	031312	177737	.WORD	177737	;SOURCE
7989	031314	177777	.WORD	177777	;DESTINATION WORD 1
7990	031316	177774	.WORD	177774	;DESTINATION WORD 2
7991	031320	000011	.WORD	11	;TEST PSW
7992	031322	177777	.WORD	177777	;RESULT WORD 1
7993	031324	177777	.WORD	177777	;RESULT WORD 2
7994					
7995	031326	177747	.WORD	177747	;SOURCE
7996	031330	100000	.WORD	100000	;DESTINATION WORD 1
7997	031332	174000	.WORD	174000	;DESTINATION WORD 2
7998	031334	000010	.WORD	10	;TEST PSW
7999	031336	177777	.WORD	177777	;RESULT WORD 1
8000	031340	177700	.WORD	177700	;RESULT WORD 2
8001					
8002	031342	177753	.WORD	177753	;SOURCE
8003	031344	006324	.WORD	6324	;DESTINATION WORD 1
8004	031346	071002	.WORD	71002	;DESTINATION WORD 2
8005	031350	000001	.WORD	1	;TEST PSW
8006	031352	000000	.WORD	0	;RESULT WORD 1
8007	031354	000146	.WORD	146	;RESULT WORD 2
8008					
8009	031356	177765	.WORD	177765	;SOURCE
8010	031360	102351	.WORD	102351	;DESTINATION WORD 1
8011	031362	177231	.WORD	177231	;DESTINATION WORD 2
8012	031364	000011	.WORD	11	;TEST PSW
8013	031366	177760	.WORD	177760	;RESULT WORD 1
8014	031370	116477	.WORD	116477	;RESULT WORD 2

FIN130:  
HSPAU:

\*\*\*\*\*  
;+TEST 216 TEST THAT AUTO DEC/INC OPERATIONS USING SP ARE ON WORD BOUNDRIES  
\*\*\*\*\*

TST216:

INC	TESTN	;INCREMENT TEST NUMBER
CLR	R6	;CLEAR SP
MOVB	(R6)+,COUNT	;TRY AUTOINC ON R6
CMP	#2,R6	;VERIFY AUTO INC BY 2
BEG	SPAU1	;BRANCH IF GOOD
		;BAD AUTO-INC
		;ALL ERRORS TO TRAP TO EMT VECTOR
		;UNIQUE ERROR NUMBER
		;ADDRESS OF ERROR MESSAGE
		;CLEAR R6

SPAU1:

8020	031372		
8021	031372	005267	147406
8022	031376	005006	
8023	031400	112667	147434
8024	031404	022706	000002
8025	031410	001403	
8026			
8027	031412	104000	
8028	031414	000773	
8029	031416	001127	
8030	031420	005006	

```

8031 031422 112667 147412      MOV  (R6)+,COUNT
8032 031426 112667 147406      MOV  (R6)+,COUNT      ;DOUBLE BYTE AUTO-INC
8033 031432 022706 000004      CMP  #4,R6             ;VERIFY RESULT
8034 031436 001403              BEQ  SPAU2              ;BRANCH IF GOOD
8035                          ;BAD DOUBLE AUTO-INC
8036 031440 104000              ERROR                   ;ALL ERRORS TO TRAP TO EMT VECTOR
8037 031442 000774              .WORD 774              ;UNIQUE ERROR NUMBER
8038 031444 001127              .WORD CPUERR            ;ADDRESS OF ERROR MESSAGE
8039 031446 012706 001000      SPAU2: MOV  #STBOT,R6     ;LOAD R6
8040 031452 114667 147362      MOV  -(R6),COUNT      ;TEST AUTO-DEC
8041 031456 022706 000776      CMP  #776,R6           ;VERIFY RESULT
8042 031462 001403              BEQ  SPAU3              ;BRANCH IF GOOD
8043 031464 104000              ERROR                   ;ALL ERRORS TO TRAP TO EMT VECTOR
8044 031466 000775              .WORD 775              ;UNIQUE ERROR NUMBER
8045 031470 001127              .WORD CPUERR            ;ADDRESS OF ERROR MESSAGE
8046 031472 012706 001000      SPAU3: MOV  #STBOT,R6     ;LOAD R6
8047 031476 114667 147336      MOV  -(R6),COUNT      ;TEST AUTO-DEC
8048 031502 114667 147332      MOV  -(R6),COUNT      ;TEST AUTO-DEC
8049 031506 022706 000774      CMP  #774,R6           ;VERIFY RESULT
8050 031512 001403              BEQ  SPAU4              ;BRANCH IF GOOD
8051 031514 104000              ERROR                   ;ALL ERRORS TO TRAP TO EMT VECTOR
8052 031516 000776              .WORD 776              ;UNIQUE ERROR NUMBER
8053 031520 001127              .WORD CPUERR            ;ADDRESS OF ERROR MESSAGE
8054 031522 005006      SPAU4: CLR  R6           ;TEST AUTO-INC ON SOP
8055 031524 105726              TST  (R6)+             ;TEST AUTO-INC
8056 031526 020627 000002      CMP  R6,#2             ;BRANCH IF GOOD
8057 031532 001403              BEQ  SPAU5              ;ALL ERRORS TO TRAP TO EMT VECTOR
8058 031534 104000              ERROR                   ;UNIQUE ERROR NUMBER
8059 031536 000777              .WORD 777              ;ADDRESS OF ERROR MESSAGE
8060 031540 001127              .WORD CPUERR            ;LOAD R6
8061 031542 012706 001000      SPAU5: MOV  #STBOT,R6     ;TEST AUTO-DEC
8062 031546 105746              TST  -(R6)             ;VERIFY RESULT
8063 031550 022706 000776      CMP  #776,R6           ;BRANCH IF GOOD
8064 031554 001403              BEQ  SPAU6              ;ALL ERRORS TO TRAP TO EMT VECTOR
8065 031556 104000              ERROR                   ;UNIQUE ERROR NUMBER
8066 031560 001000              .WORD 1000             ;ADDRESS OF ERROR MESSAGE
8067 031562 001127              .WORD CPUERR            ;LOAD R6
8068 031564 012706 001000      SPAU6: MOV  #STBOT,R6     ;TEST AUTO-DEC
8069                          TST  -(R6)             ;VERIFY RESULT
8070                          BEQ  SPAU6              ;BRANCH IF GOOD
8071 031570                          ;
8072                          ;
8073                          ;*****
8074                          ;*TEST 217      VERIFY YELLOW ZONE TRAP ON AUTO DEC OF R6
8075                          ;*****
8076 031570 005267 147210      TST217: INC  #TESTN      ;INCREMENT TEST NUMBER
8077 031574 005067 146166      CLR  CPERR             ;INIT CPU ERROR REGISTER
8078 031600 012706 000150      MOV  #150,R6           ;LOAD R6 WITH A VALUE THAT WILL
8079                          ;CAUSE A YELLOW STACK TRAP(IE. <400)
8080 031604 016767 146174 147216      MOV  4,SLOC00          ;SAVE VECTOR
8081 031612 012767 031654 146164      MOV  #MTRYA,4          ;SETUP THE STACK OVERFLOW TRAP POINTER
8082 031620 016701 146322      MOV  146,R1           ;SAVE VECTOR
8083 031624 016702 146314      MOV  144,R2           ;SAVE VECTOR
8084 031630 016703 146306      MOV  142,R3           ;SAVE VECTOR
8085 031634 005067 146306      CLR  146              ;JUST AS A PRECAUTION
8086 031640 005046              CLR  -(R6)             ;CAUSE A STACK OVERFLOW TRAP

```

```

8087 031642 012706 001000      MOV      #STBOT,R6      ;RESTORE R6 FOR ERROR CALL
8088                                ;OVERFLOW TRAP FAILED
8089 031646 104000      ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
8090 031650 001001      .WORD    1001                                ;UNIQUE ERROR NUMBER
8091 031652 001127      .WORD    CPUERR                                ;ADDRESS OF ERROR MESSAGE
8092 031654                                MTRYA:
8093 031654 022767 000010 146104      CMP      #BIT03,CPEREG ;WAS CPU ERROR REG SET PROPERLY?
8094 031662 001003      BNE     1$                                ;GO TO ERROR IF NOT
8095 031664 020627 000142      CMP      R6,#142      ;VERIFY CORRECT DECREMENT OF R6
8096 031670 001403      BEQ     MTRYB      ;BRANCH IF GOOD
8097                                ;ERROR! R6 IMPROPERLY DECREMENTED
8098                                ;OR CPU ERROR REGISTER NOT CORRECT
8099 031672                                1$:
8100 031672 104000      ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
8101 031674 001002      .WORD    1002                                ;UNIQUE ERROR NUMBER
8102 031676 001127      .WORD    CPUERR                                ;ADDRESS OF ERROR MESSAGE
8103 031700                                MTRYB:
8104 031700 005067 146062      CLR      CPEREG      ;CLEAR THE CPU ERROR REGISTER
8105 031704 016767 147120 146072      MOV      SLOCOO,4      ;RESTORE VECTOR
8106 031712 010167 146230      MOV      R1,146      ;RESTORE VECTORS
8107 031716 010267 146222      MOV      R2,144      ;
8108 031722 010367 146214      MOV      R3,142      ;
8109 031726 012706 001000      MOV      #STBOT,R6      ;
8110
8111
8112                                ;
8113 031732                                MTRYM:
8114                                ;*****
8115                                ;*TEST 220 TEST STACK OVERFLOW TRAPS IN VARIOUS MODES
8116                                ;*****
8117 031732                                TST220:
8118 031732 005267 147046      INC      #TESTN      ;INCREMENT TEST NUMBER
8119 031736 005067 146024      CLR      CPEREG      ;CLEAR CPU ERROR REGISTER
8120 031742 012706 000400      MOV      #40C,R6      ;SETUP OVERFLOW R6 DATA
8121 031746 016767 146032 147054      MOV      4,SLOCOO      ;SAVE VECTOR
8122 031754 012767 032002 146022      MOV      #TRYMA,4
8123 031762 005067 146410      CLR      376      ;JUST AS A PRECAUTION
8124 031766 005046      CLR      -(R6)      ;CAUSE OVERFLOW TRAP
8125 031770 012706 001000      MOV      #STBOT,R6      ;RESTORE R6 FOR ERROR CALL
8126                                ;NO OVERFLOW TRAP
8127 031774 104000      ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
8128 031776 001003      .WORD    1003                                ;UNIQUE ERROR NUMBER
8129 032000 001127      .WORD    CPUERR                                ;ADDRESS OF ERROR MESSAGE
8130 032002                                TRYMA:
8131 032002 005067 145760      CLR      CPEREG      ;CLEAR CPU ERROR REGISTER
8132 032006 012705 001000      MOV      #1000,R5      ;SETUP R5 DATA
8133 032012 012706 000400      MOV      #400,R6      ;SETUP OVERFLOW R6 DATA
8134 032016 012767 032040 145760      MOV      #TRYMB,4
8135 032024 064645      ADD      -(R6),-(R5)      ;CAUSE OVERFLOW TRAP
8136 032026 012706 001000      MOV      #SYBOT,R6      ;RESTORE R6 FOR ERROR CALL
8137                                ;NO OVERFLOW TRAP
8138 032032 104000      ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
8139 032034 001004      .WORD    1004                                ;UNIQUE ERROR NUMBER
8140 032036 001127      .WORD    CPUERR                                ;ADDRESS OF ERROR MESSAGE
8141 032040                                TRYMB:
8142 032040 005067 145722      CLR      CPEREG      ;CLEAR CPU ERROR REGISTER
    
```

```

8143 032044 012706 000150      MOV      #150,R6      ; SETUP OVERFLOW R6 DATA
8144 032050 012767 032072 145726  MOV      #TRYMC,4
8145 032056 044546      BIC      -(R5),-(R6)  ; CAUSE OVERFLOW TRAP
8146 032060 012706 001000      MOV      #STBOT,R6   ; RESTORE R6 FOR ERROR CALL
8147                                ; NO OVERFLOW TRAP
8148 032064 104000      ERROR    ; ALL ERRORS TO TRAP TO EMT VECTOR
8149 032066 001005      .WORD   1005        ; UNIQUE ERROR NUMBER
8150 032070 001127      .WORD   CPUERR      ; ADDRESS OF ERROR MESSAGE
8151 032072 005067 145670  TRYMC:  CLR      CPREG     ; CLEAR CPU ERROR REGISTER
8152 032076 016767 146726 145700  MOV      SLOCC0,4    ; RESTORE VECTOR
8153 032104 012706 001000      MOV      #STBOT,R6
8154
8155
8156                                ;
8157 032110                                ; MILLO:
8158                                ; *****
8159                                ; *TEST 221 TEST STACK OVERFLOW ON ILLEGAL INST TRAP
8160                                ; *****
8161 032110                                ; TST221:
8162 032110 005267 146670      INC      #TESTN      ; INCREMENT TEST NUMBER
8163 032114 005067 145646      CLR      CPREG     ; CLEAR CPU ERROR REGISTER
8164 032120 012706 000400      MOV      #400,R6    ; SETUP FOR OVERFLOW TRAP
8165 032124 016767 145660 146676  MOV      10,SLOCC0   ; SAVE VECTOR
8166 032132 012767 032160 145650  MOV      #MILLOA,10  ; SETUP ILLEGAL TRAP VECTOR
8167 032140 016767 145640 146664  MOV      4,SLOCC1    ; SAVE VECTOR
8168 032146 012767 032172 145630  MOV      #MILLOB,4   ; SETUP OVERFLOW TRAP VECTOR
8169 032154 000077      77                ; UNUSED INSTRUCTION TRAP
8170 032156 000240      NOP
8171 032160 012706 001000  MILLOA: MOV      #STBOT,R6 ; RESTORE R6 FOR ERROR CALL
8172                                ; UNUSED INSTRUCTION TRAP
8173 032164 104000      ERROR    ; ALL ERRORS TO TRAP TO EMT VECTOR
8174 032166 001006      .WORD   1006        ; UNIQUE ERROR NUMBER
8175 032170 001127      .WORD   CPUERR      ; ADDRESS OF ERROR MESSAGE
8176 032172                                ;
8177 032172 016767 146634 145604  MILLOB: MOV      SLOCC1,4 ; RESTORE VECTOR
8178 032200 016767 146624 145602  MOV      SLOCC0,10   ; RESTORE VECTOR
8179 032206 005067 145554      CLR      CPREG     ; CLEAR CPU ERROR REGISTER
8180 032212 012706 001000      MOV      #STBOT,R6 ; RESTORE R6
8181
8182
8183
8184                                ;
8185 032216                                ; MIOTO:
8186                                ; *****
8187                                ; *TEST 222 TEST STACK OVERFLOW ON IOT TRAP
8188                                ; *****
8189 032216                                ; TST222:
8190 032216 005267 146562      INC      #TESTN      ; INCREMENT TEST NUMBER
8191 032222 005067 145540      CLR      CPREG     ; CLEAR CPU ERROR REGISTER
8192 032226 012706 000400      MOV      #400,R6    ; SETUP STACK FOR OVERFLOW
8193 032232 016767 145562 146570  MOV      20,SLOCC0   ; SAVE OLD IOT VECTOR
8194 032240 012767 032266 145552  MOV      #IOTOA,20   ; SETUP ERROR ACTION ON IOT
8195 032246 016767 145532 146556  MOV      4,SLOCC1    ; SAVE VECTOR
8196 032254 012767 032300 145522  MOV      #IOTOB,4    ; SETUP CORRECT TRAP VECTOR FOR
8197                                ; OVERFLOW
8198 032262 000004      IOT                ; TEST INSTRUCTION
    
```

```

8199 032264 000240
8200 032266 012706 001000 IOTOA: NOP
MOV #STBOT,R6 ;RESTORE R6 FOR ERROR CALL
;FAILURE OF STACK OVERFLOW
8201
8202 032272 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
8203 032274 001007 .WORD 1007 ;UNIQUE ERROR NUMBER
8204 032276 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
8205 032300
8206 032300 005067 145462 IOTOB: CLR CPREG ;CLEAR CPU ERROR REGISTER
8207 032304 012706 001000 MOV #STBOT,R6
8208 032310 016767 146516 145466 MOV SLOC01,4 ;RESTORE VECTOR
8209 032316 016767 146506 145474 MOV SLOC00,20 ;RESTORE TRAP VECTOR
8210
8211
8212
8213 032324 MEMTO:
;*****
;*TEST 223 TEST STACK OVERFLOW ON EMT TRAP
;*****
8214
8215
8216
8217 032324 TST223:
8218 032324 005267 146454 INC #TESTN ;INCREMENT TEST NUMBER
8219 032330 005067 145432 CLR CPREG ;CLEAR CPU ERROR REGISTER
8220 032334 012706 000400 MOV #400,R6 ;SETUP STACK FOR OVERFLOW
8221 032340 016767 145464 146462 MOV 30,SLOC00 ;SAVE OLD EMT VECTOR
8222 032346 012767 032374 145454 MOV #EMTOA,30 ;SETUP ERROR ACTION ON EMT
8223 032354 016767 145424 146450 MOV 4,SLOC01 ;SAVE VECTOR
8224 032362 012767 032406 145414 MOV #EMTOB,4 ;SETUP CORRECT TRAP VECTOR FOR
8225 ;OVERFLOW
8226 032370 104000 EMT ;TEST INSTRUCTION
8227 032372 000240 NOP
8228 032374 012706 001000 EMTOA: MOV #STBOT,R6 ;RESTORE R6 FOR ERROR CALL.
;FAILURE OF STACK OVERFLOW
8229
8230 032400 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
8231 032402 001010 .WORD 1010 ;UNIQUE ERROR NUMBER
8232 032404 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
8233 032406
8234 032406 016767 146416 145414 EMTOB: MOV SLOC00,30 ;RESTORE TRAP VECTOR
8235 032414 016767 146412 145362 MOV SLOC01,4 ;RESTORE VECTOR
8236 032422 005067 145340 CLR CPREG ;CLEAR CPU ERROR REGISTER
8237 032426 012706 001000 MOV #STBOT,R6
8238
8239 032432 MTRPO:
;*****
;*TEST 224 TEST STACK OVERFLOW ON TRAP
;*****
8240
8241
8242
8243 032432 TST224:
8244 032432 005267 146346 INC #TESTN ;INCREMENT TEST NUMBER
8245 032436 005067 145324 CLR CPREG ;CLEAR CPU ERROR REGISTER
8246 032442 012706 000400 MOV #400,R6 ;SETUP STACK FOR OVERFLOW
8247 032446 016767 145362 146354 MOV 34,SLOC00 ;SAVE OLD TRP VECTOR
8248 032454 012767 032502 145352 MOV #TRPOA,34 ;SETUP ERROR ACTION ON TRP
8249 032462 016767 145316 146342 MOV 4,SLOC01 ;SAVE VECTOR
8250 032470 012767 032520 145306 MOV #TRPOB,4 ;SETUP CORRECT TRAP VECTOR FOR
8251 ;OVERFLOW
8252 032476 104400 TRAP ;TEST INSTRUCTION
8253 032500 000240 NOP
8254 032502 010637 001062 TRPUA: MOV SP,#$AVSP1 ;SAVE ERROR DATA

```

```

8255 032506 012706 001000      MOV      #STBOT,R6                ;RESTORE R6 FOR ERROR CALL
8256                                ;FAILURE OF STACK OVERFLOW
8257 032512 104000      ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
8258 032514 001011      .WORD    1011                        ;UNIQUE ERROR NUMBER
8259 032516 001127      .WORD    CPUERR                      ;ADDRESS OF ERROR MESSAGE
8260 032520
8261 032520 016767 146304 145306  TRPUB:  MOV      SLOC00,34                ;RESTORE TRAP VECTOR
8262 032526 016767 146300 145250  MOV      SLOC01,4                ;RESTORE VECTOR
8263 032534 005067 145226      CLR      CPEREG                    ;CLEAR CPU ERROR REGISTER
8264 032540 012706 001000      MOV      #STBOT,R6
8265
8266
8267
8268 032544      ;
8269      ;
8270      ;*TEST 225      TEST STACK OVERFLOW ON BPT
8271      ;*****
8272 032544      TST225:
8273 032544 005267 146234      INC      $TESTN                    ;INCREMENT TEST NUMBER
8274 032550 005067 145212      CLR      CPEREG                    ;CLEAR CPU ERROR REGISTER
8275 032554 012706 000400      MOV      #400,R6                    ;SETUP STACK FOR OVERFLOW
8276 032560 016767 145230 146242  MOV      14,SLOC00                ;SAVE OLD BPT VECTOR
8277 032566 012767 032614 145220  MOV      #BPTOA,14                ;SETUP ERROR ACTION ON BPT
8278 032574 016767 145204 146230  MOV      4,SLOC01                ;SAVE VECTOR
8279 032602 012767 032632 145174  MOV      #BPTOB,4                ;SETUP CORRECT TRAP VECTOR FOR
8280                                ;OVERFLOW
8281 032610 000003      BPT                                ; TEST INSTRUCTION
8282 032612 000240      NOP
8283 032614 010637 001062  BPTOA: MOV      SP,#$SAVSP1                ;SAVE ERROR DATA
8284 032620 012706 001000      MOV      #STBOT,R6                ;RESTORE R6 FOR ERROR CALL
8285                                ;FAILURE OF STACK OVERFLOW
8286 032624 104000      ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
8287 032626 001012      .WORD    1012                        ;UNIQUE ERROR NUMBER
8288 032630 001127      .WORD    CPUERR                      ;ADDRESS OF ERROR MESSAGE
8289 032632
8290 032632 005067 145130      CLR      CPEREG                    ;CLEAR CPU ERROR REGISTER
8291 032636 016767 146166 145150  MOV      SLOC00,14                ;RESTORE TRAP VECTOR
8292 032644 016767 146162 145132  MOV      SLOC01,4                ;RESTORE VECTOR
8293 032652 012706 001000      MOV      #STBOT,R6
8294
8295
8296
8297 032656      ;
8298      ;
8299      ;*TEST 226      TEST STACK OVERFLOW AND ILLEGAL JMP INSTRUCTION
8300      ;*****
8301 032656      TST226:
8302 032656 005267 146122      INC      $TESTN                    ;INCREMENT TEST NUMBER
8303 032662 005067 145100      CLR      CPEREG                    ;CLEAR CPU ERROR REGISTER
8304 032666 012706 000400      MOV      #400,R6                    ;SETUP STACK FOR OVERFLOW
8305 032672 016767 145112 146130  MOV      10,SLOC00                ;SAVE OLD ILLEGAL INST. VECTOR
8306 032700 012767 032730 145102  MOV      #ILAOB,10                ;SETUP ERROR ACTION ILLEGAL OPCODE
8307 032706 016767 145072 146116  MOV      4,SLOC01                ;SAVE VECTOR
8308 032714 012767 032742 145062  MOV      #ILBOB,4                ;SETUP CORRECT TRAP VECTOR FOR
8309                                ;OVERFLOW
8310 032722 005001      CLR      R1

```

```

8311 032724 000101      JMP      R1          ; TEST INSTRUCTION
8312 032726 000240      NOP
8313 032730 012706 001000  ILA0A:  MOV      #STBOT,R6      ;RESTORE R6 FOR ERROR CALL
8314                                     ;FAILURE OF STACK OVERFLOW
8315 032734 104000      ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
8316 032736 001013      .WORD      1013      ;UNIQUE ERROR NUMBER
8317 032740 001127      .WORD      CPUERR    ;ADDRESS OF ERROR MESSAGE
8318 032742
8319 032742 016767 146064 145034  ILB0B:  MOV      SLOC01,4      ;RESTORE VECTOR
8320 032750 016767 146054 145032      MOV      SLOC00,10    ;RESTORE TRAP VECTOR
8321 032756 005067 145004      CLR      CPEREG      ;CLEAR CPU ERROR REGISTER
8322 032762 012706 001000      MOV      #STBOT,R6
8323
8324
8325 032766      ;
8326      ;MILLB0:
8327      ;*****
8328      ;*TEST 227 TEST STACK OVERFLOW ON ILLEGAL JSR INST
8329      ;*****
8329 032766      TST227:
8330 032766 005267 146012      INC      $TESTN      ;INCREMENT TEST NUMBER
8331 032772 012706 000400      MOV      #400,R6     ;SETUP STACK FOR OVERFLOW
8332 032776 016767 145006 146024      MOV      10,SLOC00   ;SAVE OLD VECTOR
8333 033004 012767 033034 144776      MOV      #ILLBOA,10  ;SETUP ERROR ACTION ON ILL. OPCODE
8334 033012 016767 144766 146012      MOV      4,SLOC01    ;SAVE VECTOR
8335 033020 012767 033046 144756      MOV      #ILLB0B,4   ;SETUP CORRECT TRAP VECTOR FOR OVERFLOW
8336 033026 005001      CLR      R1
8337 033030 004501      JSR      R5,R1      ;*** TEST INSTRUCTION***
8338 033032 000240      NOP
8339 033034 012706 001000  ILLBOA: MOV      #STBOT,R6      ;RESTORE R6 FOR ERROR CALL
8340                                     ;ERROR!! FAILURE OF STACK OVERFLOW
8341 033040 104000      ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
8342 033042 001014      .WORD      1014      ;UNIQUE ERROR NUMBER
8343 033044 001127      .WORD      CPUERR    ;ADDRESS OF ERROR MESSAGE
8344 033046 005037 177766      ILLB0B: CLR      #CPEREG    ;CLEAR CPU ERROR REGISTER
8345 033052 016767 145752 144730      MOV      SLOC00,10   ;RESTORE TRAP VECTOR
8346 033060 016767 145746 144716      MOV      SLOC01,4    ;RESTORE VECTOR
8347 033066 012706 001000      MOV      #STBOT,R6
8348
8349
8350
8351 033072      ;
8352      ;MST0:
8353      ;*****
8354      ;*TEST 230 TEST FOR FALSE STACK OVERFLOW
8355      ;*****
8355 033072      TST230:
8356 033072 005267 145706      INC      $TESTN      ;INCREMENT TEST NUMBER
8357 033076 016767 144702 145724      MOV      4,SLOC00    ;SAVE VECTOR
8358 033104 012767 033152 144672      MOV      #MSTOE,4    ;ANTICIPATE OVERFLOW ERROR
8359 033112 012706 001002      MOV      #1002,R6    ;SETUP LEGAL R6
8360 033116 005746      TST      -(R6)      ;TRY TO CAUSE STACK OVERFLOW
8361 033120 012706 002002      MOV      #2002,R6    ;SETUP LEGAL R6
8362 033124 005746      TST      -(R6)      ;TRY TO CAUSE STACK OVERFLOW
8363 033126 012706 004002      MOV      #4002,R6    ;SETUP LEGAL R6
8364 033132 005746      TST      -(R6)      ;TRY TO CAUSE STACK OVERFLOW
8365 033134 012706 010002      MOV      #10002,R6   ;SETUP LEGAL R6
8366 033140 005746      TST      -(R6)      ;TRY TO CAUSE STACK OVERFLOW

```

```

8367 033142 012706 100402      MOV      #100402,R6      ;SETUP LEGAL R6
8368 033146 005746              TST      -(R6)          ;TRY TO CAUSE STACK OVERFLOW
8369 033150 000405              BR       MSTOEE        ;EXIT MODULE
8370 033152 012706 001000      MSTOE:  MOV      #STBOT,R6 ;RESTORE R6 FOR ERROR CALL
8371                                ;ERROR!! INVALID STACK OVERFLOW ERROR
8372 033156 104000              ERROR             ;ALL ERRORS TO TRAP TO EMT VECTOR
8373 033150 001015              .WORD    1015         ;UNIQUE ERROR NUMBER
8374 033162 001127              .WORD    CPUERR       ;ADDRESS OF ERROR MESSAGE
8375 033164 016767 145640 144612 MSTOEE:  MOV      SLOC00,4    ;RESTORE VECTOR
8376 033172 012706 001000      MOV      #STBOT,R6
8377
8378
8379
8380 033176
8381
8382
8383
8384 033176
8385 033176 005267 145602      TST231: INC      #TESTN      ;INCREMENT TEST NUMBER
8386 033202 012706 001000      MOV      #STBOT,R6    ;SETUP STACK
8387 033206 016767 144602 145614 MOV      14,SLOC00     ;SAVE OLD T-BIT VECTOR
8388 033214 012746 000020      MOV      #20,-(R6)    ;PUSH T-BIT
8389 033220 012746 033242      MOV      #MTTA,-(R6)  ;SETUP ERROR TRAP VECTOR
8390 033224 012767 033250 144562 MOV      #MTTB,14     ;SETUP NEW T-BIT VECTOR
8391 033232 000002              RTI                  ;CAUSE A T BIT SET IN PSW
8392                                ;SHOULD NEVER BE EXECUTED
8393 033234 104000              ERROR             ;ALL ERRORS TO TRAP TO EMT VECTOR
8394 033236 001016              .WORD    1016         ;UNIQUE ERROR NUMBER
8395 033240 001127              .WORD    CPUERR       ;ADDRESS OF ERROR MESSAGE
8396                                ;DIDNT TAKE CORRECT TRAP
8397 033242
8398 033242 104000      MTTA:  ERROR             ;ALL ERRORS TO TRAP TO EMT VECTOR
8399 033244 001017              .WORD    1017         ;UNIQUE ERROR NUMBER
8400 033246 001127              .WORD    CPUERR       ;ADDRESS OF ERROR MESSAGE
8401 033250 022706 000774      MTTB:  CMP      #STBOT-4,R6 ;VERIFY SP DECIRMENT
8402 033254 001403              BEQ      MTTD         ;BRANCH IF GOOD
8403                                ;BAD SP
8404 033256 104000      ERROR             ;ALL ERRORS TO TRAP TO EMT VECTOR
8405 033260 001020              .WORD    1020         ;UNIQUE ERROR NUMBER
8406 033262 001127              .WORD    CPUERR       ;ADDRESS OF ERROR MESSAGE
8407 033264 021627 033242      MTTD:  CMP      (R6),#MTTA ;VERIFY PC SAVED ON STACK
8408 033270 001403              BEQ      MTTE        ;BRANCH IF GOOD
8409                                ;INCORRECT PC ON STACK
8410 033272 104000      ERROR             ;ALL ERRORS TO TRAP TO EMT VECTOR
8411 033274 001021              .WORD    1021         ;UNIQUE ERROR NUMBER
8412 033276 001127              .WORD    CPUERR       ;ADDRESS OF ERROR MESSAGE
8413 033300
8414 033300 016767 145524 144506      MTTE:  MOV      SLOC00,14 ;RESTORE VECTOR 14
8415 033306 012706 001000      MOV      #STBOT,R6
8416
8417
8418 033312
8419
8420
8421
8422 033312
MTTS:
;*****
;*TEST 232 TEST T-BIT TRAPS WITH RTI
;*****
TST232:

```



```

8423 033312 005267 145466      INC      #TESTN      ;INCREMENT TEST NUMBER
8424 033316 012706 001000      MOV      #STBOT,R6  ;SETUP STACK
8425 033322 016767 144466 145500  MOV      14,SLOC00  ;SAVE OLD T-BIT VECTOR
8426 033330 012746 000020      MOV      #20,-(R6)  ;PUSH T-BIT
8427 033334 012746 033356      MOV      #MTTSA,-(R6) ;SETUP ERROR TRAP VECTOR
8428 033340 012767 033366 144446  MOV      #MTTJB,14  ;SETUP NEW T-BIT VECTOR
8429 033346 000006      RTT                          ;CAUSE A T BIT SET IN PSW
8430                                ;SHOULD NEVER BE EXECUTED
8431 033350 104000      ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
8432 033352 001022      .WORD 1022                        ;UNIQUE ERROR NUMBER
8433 033354 001127      .WORD CPUERR                       ;ADDRESS OF ERROR MESSAGE
8434 033356 000240      MTTSA: NOP                          ;RTT WILL EXECUTE THIS INSTRUCTION
8435                                ;WITH A T-BIT TRAP
8436                                ;DIDNT TAKE CORRECT TRAP
8437 033360      MTTSQ:
8438 033360 104000      ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
8439 033362 001023      .WORD 1023                        ;UNIQUE ERROR NUMBER
8440 033364 001127      .WORD CPUERR                       ;ADDRESS OF ERROR MESSAGE
8441 033366 022706 000774      MTTSB: CMP      #STBOT-4,R6          ;VERIFY SP DECIRMENT
8442 033372 001403      BEQ      MTTSD                      ;BRANCH IF GOOD
8443                                ;BAD SP
8444 033374 104000      ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
8445 033376 001024      .WORD 1024                        ;UNIQUE ERROR NUMBER
8446 033400 001127      .WORD CPUERR                       ;ADDRESS OF ERROR MESSAGE
8447 033402 021627 033360      MTTSD: CMP      (R6),#MTTSQ          ;VERIFY PC SAVED ON STACK
8448 033406 001403      BEQ      MTTSE                      ;BRANCH IF GOOD
8449                                ;INCORRECT PC ON STACK
8450 033410 104000      ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
8451 033412 001025      .WORD 1025                        ;UNIQUE ERROR NUMBER
8452 033414 001127      .WORD CPUERR                       ;ADDRESS OF ERROR MESSAGE
8453 033416      MTTSE:
8454 033416 016767 145406 144370  MOV      SLOC00,14          ;RESTORE VECTOR 14
8455 033424 012706 001000      MOV      #STBOT,R6
8456
8457 033430      MTTT:
8458      ;*****
8459      ;*TEST 233      TEST OLD STATUS ON T BIT TRAP
8460      ;*****
8461 033430      TST233:
8462 033430 005267 145350      INC      #TESTN      ;INCREMENT TEST NUMBER
8463 033434 012706 001000      MOV      #STBOT,R6  ;SETUP STACK
8464 033440 016767 144350 145362  MOV      14,SLOC00  ;SAVE OLD VECTOR
8465 033446 012746 000020      MOV      #20,-(R6)  ;PUSH T-BIT
8466 033452 012746 033504      MOV      #MTTRA,-(R6) ;SETUP ERROR TRAP VECTOR
8467 033456 012767 033512 144350  MOV      #MTTRB,14  ;SETUP NEW T-BIT VECTOR
8468 033464 012767 000357 144304  MOV      #357,PS    ;SET PRIORITY AND COND C
8469 033472 000277      SCC
8470 033474 000002      RTI
8471                                ;SHOULD NEVER EXECUTE
8472 033476 104000      ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
8473 033500 001026      .WORD 1026                        ;UNIQUE ERROR NUMBER
8474 033502 001127      .WORD CPUERR                       ;ADDRESS OF ERROR MESSAGE
8475                                ;DIDNT TAKE CORRECT TRAP
8476 033504      MTTRA:
8477 033504 104000      ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
8478 033506 001027      .WORD 1027                        ;UNIQUE ERROR NUMBER

```



```

8535 033704 001403          BEQ      MRTF          ;BRANCH IF GOOD
8536                                ;INCORRECT PC ON STACK
8537 033706 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
8538 033710 001036          .WORD    1036          ;UNIQUE ERROR NUMBER
8539 033712 001127          .WORD    CPUERR        ;ADDRESS OF ERROR MESSAGE
8540 033714                                ;
8541 033714 016767 145110 144066 MRTF:      MOV      SLOC00,10      ;RESTORE TRAP VECTOR
8542 033722 012706 001000          MOV      #STBOT,R6
8543
8544                                ;
8545                                ;
8546 033726                                MRT0:
8547                                ;*****
8548                                ;*TEST 235      TEST OLD STATUS ON RESERVED INST TRAP
8549                                ;*****
8550 033726                                YST235:
8551 033726 005267 145052          INC      #TESTN          ;INCREMENT TEST NUMBER
8552 033732 012706 001000          MOV      #STBOT,R6      ;SETUP STACK
8553 033736 016767 144046 145064          MOV      10,SLOC00      ;SAVE OLD VECTOR
8554 033744 012767 033770 144036          MOV      #MRT0B,10      ;SETUP NEW VECTOR
8555 033752 005067 144020          CLR      PS              ;CLEAR PRIORITY AND COND C
8556 033756 000257          CCC
8557 033760 000077          77
8558                                ;DIDNT TAKE CORRECT TRAP
8559 033762                                MRT0A:
8560 033762 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
8561 033764 001037          .WORD    1037          ;UNIQUE ERROR NUMBER
8562 033766 001127          .WORD    CPUERR        ;ADDRESS OF ERROR MESSAGE
8563 033770 026727 145002 000000 MRT0B:  CMP      STBOT-2,#0      ;VERIFY PSW ON STACK
8564 033776 001403          BEQ      MRT0C          ;BRANCH IF CORRECT STATUS
8565                                ;BAD STATUS ON STACK
8566 034000 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
8567 034002 001040          .WORD    1040          ;UNIQUE ERROR NUMBER
8568 034004 001127          .WORD    CPUERR        ;ADDRESS OF ERROR MESSAGE
8569 034006 012706 001000          MOV      #STBOT,R6      ;SETUP STACK
8570 034012 012767 034040 143770          MOV      #MRT0E,10      ;SET UP TRAP VECTOR
8571 034020 012767 000357 143750          MOV      #357,PS
8572 034026 000277          SCC
8573 034030 000077          77
8574                                ;RESERVED INSTRUCTION
8575                                ;DIDNT TAKE CORRECT TRAP
8576 034032                                MRT0D:
8577 034032 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
8578 034034 001041          .WORD    1041          ;UNIQUE ERROR NUMBER
8579 034036 001127          .WORD    CPUERR        ;ADDRESS OF ERROR MESSAGE
8580 034040 026727 144732 000357 MRT0E:  CMP      STBOT-2,#357    ;VERIFY OLD PSW ON STACK
8581 034046 001403          BEQ      MRT0F          ;BRANCH IF GOOD
8582                                ;OLD PSW INCORRECT
8583 034050 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
8584 034052 001042          .WORD    1042          ;UNIQUE ERROR NUMBER
8585 034054 001127          .WORD    CPUERR        ;ADDRESS OF ERROR MESSAGE
8586 034056                                ;
8587 034056 016767 144746 143724 MRT0F:  MOV      SLOC00,10      ;RESOTRE TRAP VECTOR
8588 034064 012706 001000          MOV      #STBOT,R6
8589
8590 034070                                MTP:

```

```

8591 ;*****
8592 ;*TEST 236 TEST TRAP INST
8593 ;*****
8594 034070 TST236:
8595 034070 005267 144710 INC $TESTN ;INCREMENT TEST NUMBER
8596 034074 012706 001000 MOV $STBOT,R6 ;SETUP STACK
8597 034100 016767 143730 144722 MOV 34,SLOCC0 ;SAVE OLD VECTOR
8598 034106 012767 034132 143720 MOV $MTPB,34 ;SETUP NEW TRAP VECTOR
8599 034114 005067 143656 CLR PS ;CLEAR PRIORITY AND COND C
8600 034120 000257 CCC
8601 034122 104400 TRAP
8602 ;DIDNT TAKE CORRECT TRAP
8603 034124 MTPR:
8604 034124 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
8605 034126 001043 .WORD 1043 ;UNIQUE ERROR NUMBER
8606 034130 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
8607 034132 022706 000774 MTPB: CMP $STBOT-4,R6 ;VERIFY SP DECRIMENT
8608 034136 001403 BEQ MTPQ ;BRANCH IF GOOD
8609 ;BAD PC ON STACK
8610 034140 103000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
8611 034142 001044 .WORD 1044 ;UNIQUE ERROR NUMBER
8612 034144 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
8613 034146 021627 034124 MTPQ: CMP (R6),$MTPR ;VERIFY PROPER PC ON STACK
8614 034152 001403 BEQ MTPF ;BRANCH IF GOOD
8615 ;INCORRECT PC ON STACK
8616 034154 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
8617 034156 001045 .WORD 1045 ;UNIQUE ERROR NUMBER
8618 034160 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
8619 034162 MTPF:
8620 034162 016767 144642 143644 MOV SLOCC0,34 ;RESTORE VECTOR
8621 034170 012706 001000 MOV $STBOT,R6
8622
8623
8624
8625 034174 MTPQ:
8626 ;*****
8627 ;*TEST 237 TEST OLD STATUS SAVED ON TRAP
8628 ;*****
8629 034174 TST237:
8630 034174 005267 144604 INC $TESTN ;INCREMENT TEST NUMBER
8631 034200 012706 001000 MOV $STBOT,R6 ;SETUP STACK
8632 034204 016767 143624 144616 MOV 34,SLOCC0 ;SAVE OLD VECTOR
8633 034212 012767 034236 143614 MOV $MTPOB,34 ;SETUP NEW TRAP VECTOR
8634 034220 005067 143552 CLR PS ;CLEAR PRIORITY AND COND C
8635 034224 000257 CCC
8636 034226 104400 TRAP
8637 ;DIDNT TAKE CORRECT TRAP
8638 034230 MTPQA:
8639 034230 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
8640 034232 001046 .WORD 1046 ;UNIQUE ERROR NUMBER
8641 034234 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
8642 034236 026727 144534 000000 MTPQB: CMP STBOT-2,$0 ;VERIFY PSW ON STACK
8643 034244 001403 BEQ MTPQC ;BRANCH IF CORRECT STATUS
8644 ;BAD STATUS ON STACK
8645 034246 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
8646 034250 001047 .WORD 1047 ;UNIQUE ERROR NUMBER

```

```

8647 034252 001127          .WORD  CPUERR          ; ADDRESS OF ERROR MESSAGE
8648 034254 012706 001000  MTPOC:  MOV    #STBOT,R6    ; SETUP STACK
8649 034260 012767 034306 143546  MOV    #MTPOE,34    ; SET UP TRAP VECTOR
8650 034266 012767 000357 143502  MOV    #357,PS     ; SET PRIORITY
8651 034274 000277          SCC          ; SET CONDITION CODES
8652 034276 104400          TRAP          ; ISSUE TRAP
8653                                ; DIDNT TAKE CORRECT TRAP
8654 034300          MTPOD:
8655 034300 104000          ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
8656 034302 001050          .WORD  1050        ; UNIQUE ERROR NUMBER
8657 034304 001127          .WORD  CPUERR      ; ADDRESS OF ERROR MESSAGE
8658 034306 026727 144464 000357  MTPOE:  CMP    STBOT-2,#357 ; VERIFY OLD PSW ON STACK
8659 034314 001403          BEQ    MTPOF       ; BRANCH IF GOOD
8660                                ; OLD PSW INCORRECT
8661 034316 104000          ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
8662 034320 001051          .WORD  1051        ; UNIQUE ERROR NUMBER
8663 034322 001127          .WORD  CPUERR      ; ADDRESS OF ERROR MESSAGE
8664 034324          MTPOF:
8665 034324 016767 144500 143502  MOV    SLOC00,34   ; RESTORE TRAP VECTOR
8666 034332 012706 001000          MOV    #STBOT,R6
8667
8668 ;
8669 ;
8670 034336          MTPA:
8671 ;
8672 ;*****
8673 ;*TEST 240 TEST ALL TRAP OPCODES - SELF MODIFYING
8674 ;*****
8675 034336 005267 144442          TST240:
8676 034342 005003          INC    #TESTN     ; INCREMENT TEST NUMBER
8677 034344 012706 001000          CLR    R3         ; SETUP REGISTER TO INDICATE OPCODE
8678 034350 016767 143460 144452  MOV    #STBOT,R6  ; SETUP STACK
8679 034356 016767 143422 144446  MOV    34,SLOC00 ; SAVE OLD VECTOR
8680 034364 012767 034420 143412  MOV    4,SLOC01  ; SAVE IN CASE OF HALT
8681 034372 012767 034426 143434  MOV    #MTPAH,4  ; SETUP HALT TRAP
8682 034400 000167 000022          MOV    #MTPAA,34 ; SETUP NEW TRAP VECTOR
8683                                ; GO INTO LOOPING CODE
8684 034404 000000          MTPAL:  HALT     ; SET TO A ZERO
8685                                ; TRAP INSTRUCTION FAILED TO TRAP
8686 034406 104000          ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
8687 034410 001052          .WORD  1052        ; UNIQUE ERROR NUMBER
8688 034412 001127          .WORD  CPUERR      ; ADDRESS OF ERROR MESSAGE
8689                                ; EXAMINE OPCODE AT LOCATION MTPAL:
8690
8691 034414 000167 000006          JMP    MTPAA     ; ATTEMPT TO GO ON
8692 ;
8693                                ; ERROR, EITHER CANT MODIFY LOCATION MTPAL
8694 034420          MTPAH:
8695 034420 104000          ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
8696 034422 001053          .WORD  1053        ; UNIQUE ERROR NUMBER
8697 034424 001127          .WORD  CPUERR      ; ADDRESS OF ERROR MESSAGE
8698
8699                                ; OR TRAP INSTRUCTION FAILED
8700 034426          MTPAA:
8701 034426 005203          INC    R3         ; GET NEXT OPCODE
8702
    
```

```

8703 034430 012706 001000      MOV      #STBOT,R6      ;RESTORE STACK
8704 034434 020327 000400      CMP      R3,#400      ;SEE IF LAST OPCODE
8705 034440 001406              BEQ      MTPAE         ;BRANCH IF DONE
8706 034442 012767 104400 177734  MOV      #104400,MTPAL ;TRAP OPCODE INTO LOCATION
8707 034450 060367 177730      ADD      R3,MTPAL     ;FORM TEST OPCODE
8708 034454 000753              BR       MTPAL        ;EXECUTE TEST
8709 034456              MTPAE:
8710
8711 034456 016767 144346 143350      MOV      SLOC00,34
8712 034464 016767 144342 143312      MOV      SLOC01,4      ;RESTORE VECTORS
8713
8714 034472 012706 001000      MOV      #STBOT,R6
8715
8716
8717
8718 034476      MIOT:
8719      ;*****
8720      ;*TEST 241      TEST IOT TRAP
8721      ;*****
8722 034476      TST241:
8723 034476 005267 144302      INC      $TESTN      ;INCREMENT TEST NUMBER
8724 034502 012706 001000      MOV      #STBOT,R6   ;SETUP STACK
8725 034506 016767 143306 144314  MOV      20,SLOC00   ;SAVE OLD VECTOR
8726 034514 012767 034532 143276  MOV      #MIOTB,20   ;SETUP NEW IOT VECTOR
8727 034522 000004              IOT                ;***TEST INSTRUCTION***
8728
8729 034524      MIOTA:
8730 034524 104000      ERROR                   ;ALL ERRORS TO TRAP TO EMT VECTOR
8731 034526 001054      .WORD      1054        ;UNIQUE ERROR NUMBER
8732 034530 001127      .WORD      CPUERR     ;ADDRESS OF ERROR MESSAGE
8733 034532 022706 000774      MIOTB:  CMP      #STBOT-4,R6 ;VERIFY SP DECRIMENT
8734 034536 001403      BEQ      MIOTD        ;BRANCH IF GOOD
8735
8736 034540 104000      ERROR                   ;ALL ERRORS TO TRAP TO EMT VECTOR
8737 034542 001055      .WORD      1055        ;UNIQUE ERROR NUMBER
8738 034544 001127      .WORD      CPUERR     ;ADDRESS OF ERROR MESSAGE
8739 034546 021627 034524      MIOTD:  CMP      (R6),#MIOTA ;VERIFY PROPER PC ON STACK
8740 034552 001403      BEQ      MIOTF        ;BRANCH IF GOOD
8741
8742 034554 104000      ERROR                   ;ALL ERRORS TO TRAP TO EMT VECTOR
8743 034556 001056      .WORD      1056        ;UNIQUE ERROR NUMBER
8744 034560 001127      .WORD      CPUERR     ;ADDRESS OF ERROR MESSAGE
8745 034562 016767 144242 143230  MIOTF:  MOV      SLOC00,20   ;RESTORE VECTOR
8746 034570 012706 001000      MOV      #STBOT,R6
8747
8748 034574      MITO:
8749      ;*****
8750      ;*TEST 242      TEST OLD STATUS ON IOT TRAP
8751      ;*****
8752 034574      TST242:
8753 034574 005267 144204      INC      $TESTN      ;INCREMENT TEST NUMBER
8754 034600 012706 001000      MOV      #STBOT,R6   ;SETUP STACK
8755 034604 016767 143210 144216  MOV      20,SLOC00   ;SAVE OLD VECTOR
8756 034612 012767 034636 143200  MOV      #MITOB,20   ;SETUP NEW IOT VECTOR
8757 034620 005067 143152      CLR      PS          ;CLEAR PRIORITY AND COND C
8758 034624 000257      CCC

```

```

8759 034626 000004 IOT ;DIDNT TAKE CORRECT TRAP
8760 ;
8761 034630 MITOA: ;
8762 034630 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
8763 034632 001057 .WORD 1057 ;UNIQUE ERROR NUMBER
8764 034634 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
8765 034636 026727 144134 000000 MITOB: CMP STBOT-2,#0 ;VERIFY PSW ON STACK
8766 034644 001403 BEQ MITOC ;BRANCH IF CORRECT STATUS
8767 ; ;BAD STATUS ON STACK
8768 034646 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
8769 034650 001060 .WORD 1060 ;UNIQUE ERROR NUMBER
8770 034652 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
8771 034654 012706 001000 MITOC: MOV #STBOT,R6 ;SETUP STACK
8772 034660 012767 034706 143132 MOV #MITOE,20 ;SET UP TRAP VECTOR
8773 034666 012767 000357 143102 MOV #357,PS ;SET PRIORITY
8774 034674 000277 SCC ;SET CONDITION CODES
8775 034676 000004 IOT ;
8776 ; ;DIDNT TAKE CORRECT TRAP
8777 034700 MITOD: ;
8778 034700 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
8779 034702 001061 .WORD 1061 ;UNIQUE ERROR NUMBER
8780 034704 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
8781 034706 026727 144064 000357 MITOE: CMP STBOT-2,#357 ;VERIFY OLD PSW ON STACK
8782 034714 001403 BEQ MITOF ;BRANCH IF GOOD
8783 ; ;OLD PSW INCORRECT
8784 034716 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
8785 034720 001062 .WORD 1062 ;UNIQUE ERROR NUMBER
8786 034722 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
8787 034724 MITOF: ;
8788 034724 016767 144100 143066 MOV SLOC00,20 ;RESTORE VECTOR
8789 034732 012706 001000 MOV #STBOT,R6
8790 ;
8791 ;
8792 034736 ;
8793 ; MET:
8794 ; ;*****
8795 ; *TEST 243 TEST EMULATOR TRAP INSTRUCTION (EMT)
8796 ; ;*****
8797 034736 005267 144042 TST243: INC #TESTN ;INCREMENT TEST NUMBER
8798 034742 012706 001000 MOV #STBOT,R6 ;SETUP STACK
8799 034746 016767 143056 144054 MOV 30,SLOC00 ;SAVE OLD VECTOR
8800 034754 012767 035006 143046 MOV #METB,30 ;SETUP NEW EMT VECTOR
8801 034762 016767 145046 144042 MOV 34,SLOC01 ;SAVE TRAP VECTOR
8802 034770 012767 043470 143036 MOV #ERROR,34 ;SET UP TO HANDLE EMT ERROR
8803 034776 104000 EMT ;
8804 ; ;DIDNT TAKE CORRECT TRAP
8805 035000 META: ;
8806 035000 104400 TRAP ;ERROR TRAP
8807 035002 001063 .WORD 1063 ;ERROR NUMBER
8808 035004 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
8809 035006 022706 000774 METB: CMP #STBOT-4,R6 ;VERIFY SP DECREMENT
8810 035012 001403 BEQ METD ;BRANCH IF GOOD
8811 ; ;BAD PC ON STACK
8812 035014 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
8813 035016 001064 .WORD 1064 ;UNIQUE ERROR NUMBER
8814 035020 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE

```

```

8815 035022 021627 035000 METD: CMP (R6),#META ;VERIFY PROPER PC ON STACK
8816 035026 001403 BEQ METF ;BRANCH IF GOOD
8817 ;INCORRECT PC ON STACK
8818 035030 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
8819 035032 001065 .WORD 1065 ;UNIQUE ERROR NUMBER
8820 035034 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
8821 035036 016767 143770 142770 METF: MOV SLOC01,34 ;RESTORE VECTOR
8822 035044 016767 143760 142756 MOV SLOC00,30 ;RESTORE VECTOR
8823 035052 012706 001000 MOV #STBOT,R6
8824
8825
8826 035056 ;
METO:
8827 ;*****
8828 ;*TEST 244 TEST OLD STATUS ON EMT TRAP
8829 ;*****
8830 035056 TST244:
8831 035056 005267 143722 INC #TESTN ;INCREMENT TEST NUMBER
8832 035062 012706 001000 MOV #STBOT,R6 ;SETUP STACK
8833 035066 016767 142736 143734 MOV 30,SLOC00 ;SAVE OLD VECTOR
8834 035074 012767 035134 142726 MOV #METOB,30 ;SETUP NEW EMT VECTOR
8835 035102 016767 142726 143722 MOV 34,SLOC01 ;SAVE TRAP VECTOR
8836 035110 012767 043470 142716 MOV #ERROR,34 ;SET UP TRAP VECTOR
8837 035116 005067 142654 CLR PS ;CLEAR PRIORITY AND COND C
8838 035122 000257 CCC
8839 035124 104000 EMT
8840 ;DIDNT TAKE CORRECT TRAP
8841 035126 METOA:
8842 035126 104400 TRAP ;ERROR TRAP
8843 035130 001066 .WORD 1066 ;ERROR NUMBER
8844 035132 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
8845 035134 026727 143636 000000 METOB: CMP STBOT-2,#0 ;VERIFY PSW ON STACK
8846 035142 001403 BEQ METOC ;BRANCH IF CORRECT STATUS
8847 ;BAD STATUS ON STACK
8848 035144 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
8849 035146 001067 .WORD 1067 ;UNIQUE ERROR NUMBER
8850 035150 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
8851 035152 012706 001000 METOC: MOV #STBOT,R6 ;SETUP STACK
8852 035156 012767 035204 142644 MOV #METOE,30 ;SET UP TRAP VECTOR
8853 035164 012767 000357 142604 MOV #357,PS ;SET PRIORITY
8854 035172 000277 SCC ;SET CONDITION CODES
8855 035174 104000 EMT
8856 ;DIDNT TAKE CORRECT TRAP
8857 035176 METOD:
8858 035176 104400 TRAP ;ERROR TRAP
8859 035200 001070 .WORD 1070 ;ERROR NUMBER
8860 035202 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
8861 035204 026727 143566 000357 METOE: CMP STBOT-2,#357 ;VERIFY OLD PSW ON STACK
8862 035212 001403 BEQ METOF ;BRANCH IF GOOD
8863 ;OLD PSW INCORRECT
8864 035214 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
8865 035216 001071 .WORD 1071 ;UNIQUE ERROR NUMBER
8866 035220 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
8867 035222 METOF:
8868 035222 016767 143604 142604 MOV SLOC01,34 ;RESTORE VECTOR
8869 035230 016767 143574 142572 MOV SLOC00,30 ;RESTORE VECTOR
8870 035236 012706 001000 MOV #STBOT,R6

```





```

8927 035414 001076 .WORD 1076 ;UNIQUE ERROR NUMBER
8928 035416 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
8929 035420 012706 001000 MBTOC: MOV #STBOT,R6 ;SETUP STACK
8930 035424 012767 035452 142362 MOV #MBTOE,14 ;SET UP TRAP VECTOR
8931 035432 012767 000357 142336 MOV #357,PS ;SET PRIORITY
8932 035440 000277 SCC ;SET CONDITION CODES
8933 035442 000003 BPT
8934
8935 035444 MBTOD: ;DIDNT TAKE CORRECT TRAP
8936 035444 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
8937 035446 001077 .WORD 1077 ;UNIQUE ERROR NUMBER
8938 035450 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
8939 035452 026727 143320 000357 MBTOE: CMP STBOT-2,#357 ;VERIFY OLD PSW ON STACK
8940 035460 001403 BEQ MBTOF ;BRANCH IF GOOD
8941
8942 035462 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
8943 035464 001100 .WORD 1100 ;UNIQUE ERROR NUMBER
8944 035466 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
8945 035470
8946 035470 016767 143334 142316 MBTOF: MOV SLOC00,14 ;RESTORE VECTOR
8947 035476 012706 001000 MOV #STBOT,R6
8948
8949
8950
8951 035502 ;
8952 ;MIL:
8953 ;*****
8954 ;*TEST 247 TEST ILLEGAL JUMP INSTRUCTION TRAP
8955 ;*****
8955 035502 TST247:
8956 035502 005267 143276 INC #TESTN ;INCREMENT TEST NUMBER
8957 035506 012706 001000 MOV #STBOT,R6 ;SETUP STACK
8958 035512 016767 142272 143310 MOV 10,SLOC00 ;SAVE OLD VECTOR
8959 035520 012767 035540 142262 MOV #MILB,10 ;SETUP NEW ILLEGAL VECTOR
8960 035526 005001 CLR R1
8961 035530 000101 JMP R1 ;**TEST INSTRUCTIO
8962 ;DIDNT TAKE CORRECT TRAP
8963 035532
8964 035532 104000 MILA: ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
8965 035534 001101 .WORD 1101 ;UNIQUE ERROR NUMBER
8966 035536 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
8967 035540 022706 000774 MILB: CMP #STBOT-4,R6 ;VERIFY SP DECRIMENT
8968 035544 001403 BEQ MILD ;BRANCH IF GOOD
8969
8970 035546 104000 ERROR ;BAD PC ON STACK
8971 035550 001102 .WORD 1102 ;ALL ERRORS TO TRAP TO EMT VECTOR
8972 035552 001127 .WORD CPUERR ;UNIQUE ERROR NUMBER
8973 035554 021627 035532 MILD: CMP (R6),#MILA ;ADDRESS OF ERROR MESSAGE
8974 035560 001403 BEQ MILF ;VERIFY PROPER PC ON STACK
8975 ;BRANCH IF GOOD
8976 035562 104000 ERROR ;INCORRECT PC ON STACK
8977 035564 001103 .WORD 1103 ;ALL ERRORS TO TRAP TO EMT VECTOR
8978 035566 001127 .WORD CPUERR ;UNIQUE ERROR NUMBER
8979 035570 016767 143234 142212 MILF: MOV SLOC00,10 ;ADDRESS OF ERROR MESSAGE
8980 035576 012706 001000 MOV #STBOT,R6 ;RESTORE VECTOR
8981
8982 035602 MILD:

```

```

8983 ;*****
8984 ;*TEST 250 TEST OLD STATUS ON ILLEGAL JUMP TRAP
8985 ;*****
8986 TST250:
8987 035602 005267 143176 INC #TESTN ;INCREMENT TEST NUMBER
8988 035606 012706 001000 MOV #STBOT,R6 ;SETUP STACK
8989 035612 016767 142172 143210 MOV 10,SLOC00 ;SAVE OLD VECTOR
8990 035620 012767 035646 142162 MOV #MILOB,10 ;SETUP NEW ILLEGAL VECTOR
8991 035626 005067 142144 CLR PS ;CLEAR PRIORITY AND COND C
8992 035632 000257 CCC
8993 035634 005001 CLR R1
8994 035636 000101 JMP R1
8995 ;DIDNT TAKE CORRECT TRAP
8996 035640 MILOA:
8997 035640 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
8998 035642 001104 .WORD 1104 ;UNIQUE ERROR NUMBER
8999 035644 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
9000 035646 026727 143124 000004 MILOB: CMP STBOT-2,#4 ;VERIFY PSW ON STACK
9001 035654 001403 BEQ MILOC ;BRANCH IF CORRECT STATUS
9002 ;BAD STATUS ON STACK
9003 035656 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9004 035660 001105 .WORD 1105 ;UNIQUE ERROR NUMBER
9005 035662 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
9006 035664 012706 001000 MILOC: MOV #STBOT,R6 ;SETUP STACK
9007 035670 012767 035716 142112 MOV #MILOE,10 ;SET UP TRAP VECTOR
9008 035676 012767 000357 142072 MOV #357,PS ;SET PRIORITY
9009 035704 000277 SCC ;SET CONDITION CODES
9010 035706 000101 JMP R1
9011 ;DIDNT TAKE CORRECT TRAP
9012 035710 MILOD:
9013 035710 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9014 035712 001106 .WORD 1106 ;UNIQUE ERROR NUMBER
9015 035714 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
9016 035716 026727 143054 000357 MILOE: CMP STBOT-2,#357 ;VERIFY OLD PSW ON STACK
9017 035724 001403 BEQ MILOF ;BRANCH IF GOOD
9018 ;OLD PSW INCORRECT
9019 035726 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9020 035730 001107 .WORD 1107 ;UNIQUE ERROR NUMBER
9021 035732 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
9022 035734 MILOF:
9023 035734 016767 143070 142046 MOV SLOC00,10 ;RESTORE VECTOR
9024 035742 012706 001000 MOV #STBOT,R6
9025
9026 ;
9027 ;
9028 035746 MIALI:
9029 ;*****
9030 ;*TEST 251 TEST ILLEGAL JSR INSTRUCTION TRAP
9031 ;*****
9032 TST251:
9033 035746 005267 143032 INC #TESTN ;INCREMENT TEST NUMBER
9034 035752 012706 001000 MOV #STBOT,R6 ;SETUP STACK
9035 035756 016767 142026 143044 MOV 10,SLOC00 ;SAVE OLD VECTOR
9036 035764 012767 036004 142016 MOV #MIALI,B,10 ;SETUP NEW ILLEGAL VECTOR
9037 035772 005003 CLR R3
9038 035774 004303 JSR R3,R3

```

```

9039
9040 035776 MIALLA: ;DIDNT TAKE CORRECT TRAP
9041 035776 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9042 036000 001110 .WORD 1110 ;UNIQUE ERROR NUMBER
9043 036002 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
9044 036004 022706 000774 MIALLB: CMP #STBOT-4,R6 ;VERIFY SP DECRIMENT
9045 036010 001403 BEQ MIALLD ;BRANCH IF GOOD
9046 ;BAD PC ON STACK
9047 036012 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9048 036014 001111 .WORD 1111 ;UNIQUE ERROR NUMBER
9049 036016 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
9050 036020 021627 035776 MIALLD: CMP (R6),#MIALLA ;VERIFY PROPER PC ON STACK
9051 036024 001403 BEQ MIALLF ;BRANCH IF GOOD
9052 ;INCORRECT PC ON STACK
9053 036026 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9054 036030 001112 .WORD 1112 ;UNIQUE ERROR NUMBER
9055 036032 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
9056 036034 016767 142770 141746 MIALLF: MOV SLOC00,10 ;RESTORE VECTOR
9057 036042 012706 001000 MOV #STBOT,R6
9058
9059
9060
9061 036046
9062
9063
9064
9065 036046 TST252: ;*****
9066 036046 005267 142732 INC #TESTN ;INCREMENT TEST NUMBER
9067 036052 012706 001000 MOV #STBOT,R5 ;SETUP STACK
9068 036056 016767 141726 142744 MOV 10,SLOC00 ;SAVE OLD VECTOR
9069 036064 012767 036112 141716 MOV #MJSIB,10 ;SETUP NEW VECTOR
9070 036072 005067 141700 CLR PS ;CLEAR PRIORITY AND COND C
9071 036076 000257 ORC
9072 036100 005003 CLR R3
9073 036102 004303 JSR R3,R3
9074
9075 036104 MJSIA: ;DIDNT TAKE CORRECT TRAP
9076 036104 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9077 036106 001113 .WORD 1113 ;UNIQUE ERROR NUMBER
9078 036110 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
9079 036112 026727 142660 000004 MJSIB: CMP STBOT-2,#4 ;VERIFY PSW ON STACK
9080 036120 001403 BEQ MJSIC ;BRANCH IF CORRECT STATUS
9081 ;BAD STATUS ON STACK
9082 036122 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9083 036124 001114 .WORD 1114 ;UNIQUE ERROR NUMBER
9084 036126 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
9085 036130 012706 001000 MJSIC: MOV #STBOT,R6 ;SETUP STACK
9086 036134 012767 036162 141646 MOV #MJSIE,10 ;SET UP TRAP VECTOR
9087 036142 012767 000357 141626 MOV #357,PS ;SET PRIORITY
9088 036150 000277 SCC ;SET CONDITION CODES
9089 036152 004303 JSR R3,R3
9090
9091 036154 MJSID: ;DIDNT TAKE CORRECT TRAP
9092 036154 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9093 036156 001115 .WORD 1115 ;UNIQUE ERROR NUMBER
9094 036160 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE

```

```

9095 036162 026727 142610 000357 MJSIE: CMP STBOT-2,#357 ;VERIFY OLD PSW ON STACK
9096 036170 001403 BEQ MJSIF ;BRANCH IF GOOD
9097 ;OLD PSW INCORRECT
9098 036172 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9099 036174 001116 .WORD 1116 ;UNIQUE ERROR NUMBER
9100 036176 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
9101 036200 MJSIF:
9102 036200 016767 142624 141602 MOV SLOC00,10 ;RESTORE VECTOR
9103 036206 012706 001000 MOV @STBOT,R6
9104
9105 ;
9106 ;
9107 ;*****
9108 ;*TEST 253 I/O TIME OUT TEST
9109 ;*****
9110 036212 TST253:
9111 036212 005267 142566 INC $TESTN ;INCREMENT TEST NUMBER
9112 036216 005067 141544 CLR CPREG ;CLEAR CPU ERROR REGISTER
9113 036222 016767 141556 142600 MOV 4,SLOC00 ;SAVE VECTOR
9114 036230 012767 036256 141546 MOV @2$,4 ;SET UP VECTOR TO HANDLE NXM
9115 036236 012767 030000 141532 MOV @30000,PS ;INIT THE PSW TO A KNOWN STATE
9116 036244 005737 177700 TST @177700 ;TRY TO ACCESS HARDWARE ADDRESS
9117 ;FOR GENERAL PURPOSE REG 0. THIS
9118 ;IS NOT IMPLEMENTED ON KDJ11
9119 ;SHOULD CAUSE TIME OUT.
9120 036250 1$:
9121 036250 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9122 036252 001117 .WORD 1117 ;UNIQUE ERROR NUMBER
9123 036254 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
9124 036256 022767 000020 141502 2$: CMP @BIT04,CPREG ;IS CPU ERROR REGISTER CORRECT?
9125 036264 001403 BEQ 3$
9126 036266 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9127 036270 001120 .WORD 1120 ;UNIQUE ERROR NUMBER
9128 036272 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
9129 036274 022627 036250 3$: CMP (SP)+,#1$ ;CHECK THAT STACK CONTAINS CORRECT ADDR.
9130 036300 001403 BEQ 4$
9131 036302 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9132 036304 001121 .WORD 1121 ;UNIQUE ERROR NUMBER
9133 036306 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
9134 036310 022627 030000 4$: CMP (SP)+,#30000 ;IS THE PSW OK?
9135 036314 001403 BEQ 5$
9136 036316 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9137 036320 001122 .WORD 1122 ;UNIQUE ERROR NUMBER
9138 036322 001127 .WORD CPUERR ;ADDRESS OF ERROR MESSAGE
9139 036324 005067 141436 5$: CLR CPREG ;CLEAR THE CPU ERROR REGISTER
9140 036330 016767 142474 141446 MOV SLOC00,4 ;RESTORE VECTOR
9141
9142 ;*****
9143 ;*TEST 254 ODD ADDRESS/ILLEGAL INST FETCH TRAP TEST
9144 ;*****
9145 ;THIS PROGRAM GENERATES AN ODD ADDRESS IN THE PC. THE KDJ11 SHOULD
9146 ;TRAP THROUGH ADDR 4
9147 ;*****
9148 036336 TST254:
9149 036336 005267 142442 INC $TESTN ;INCREMENT TEST NUMBER
9150 036342 005067 141420 CLR CPREG ;INIT THE CPU ERROR REG

```

```

9151 036346 016767 141432 142454      MOV      4,SLOC00      ;SAVE VECTOR
9152 036354 012767 036414 141422      MOV      #21,4        ;SET UP VECTOR TO HANDLE ODD ADDR TRAP
9153 036362 016767 141420 142442      MOV      6,SLOC01      ;SAVE VECTOR
9154 036370 005067 141412                CLR      6            ;INIT VECTOR
9155 036374 012746 030000                MOV      #30000,-(SP)  ;PUSH A KNOWN PSW ON THE STACK
9156 036400 012746 036407                MOV      #10*1,-(SP)  ;PUSH AN ODD NUMBER ON THE STACK
9157 036404 000002                RTI                    ;POP ODD ADDRESS OFF STACK INTO PC
9158                                     ;SHOULD TRAP HERE
9159 036406                                     10:
9160 036406 104000                ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
9161 036410 001123                .WORD 1123                        ;UNIQUE ERROR NUMBER
9162 036412 001127                .WORD CPUERR                      ;ADDRESS OF ERROR MESSAGE
9163 036414 022767 000100 141344 20:    CMP      #BIT06,CPEREG            ;IS CPU ERROR REGISTER CORRECT?
9164 036422 001403                BEQ      30
9165 036424 104000                ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
9166 036426 001124                .WORD 1124                        ;UNIQUE ERROR NUMBER
9167 036430 001127                .WORD CPUERR                      ;ADDRESS OF ERROR MESSAGE
9168 036432 022726 036407 30:    CMP      #10*1,(SP)+            ;IS STACK CONTENTS CORRECT?
9169 036436 001403                BEQ      40
9170 036440 104000                ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
9171 036442 001125                .WORD 1125                        ;UNIQUE ERROR NUMBER
9172 036444 001127                .WORD CPUERR                      ;ADDRESS OF ERROR MESSAGE
9173 036446 022726 030000 40:    CMP      #30000,(SP)+          ;IS PSW CORRECT ON STACK?
9174 036452 001403                BEQ      50
9175 036454 104000                ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
9176 036456 001126                .WORD 1126                        ;UNIQUE ERROR NUMBER
9177 036460 001127                .WORD CPUERR                      ;ADDRESS OF ERROR MESSAGE
9178 036462 005067 141300 50:    CLR      CPEREG              ;CLEAR CPU ERROR REG
9179
9180                                     ;NOW WE'LL TRY TO FETCH AN INSTRUCTION FROM AN INTERNAL REGISTER.
9181                                     ;THIS SHOULD CAUSE A TRAP TO ADDR 4 AND SET BIT 6 IN THE CPU
9182                                     ;ERROR REGISTER.
9183
9184 036466 012767 036520 141510      MOV      #71,4
9185 036474 012767 030340 141304      MOV      #30340,6
9186 036502 005067 141270                CLR      PS
9187 036506 000167 141264                JMP      PS
9188                                     ;*****TEST INSTRUCTION*****
9189                                     ;TRY INSTRUCTION FETCH FROM INTERNAL
9190                                     ;REGISTER-- SHOULD TRAP VIA ADDR 4
9191 036512 104000                ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
9192 036514 001127                .WORD 1127                        ;UNIQUE ERROR NUMBER
9193 036516 001127                .WORD CPUERR                      ;ADDRESS OF ERROR MESSAGE
9194 036520 016701 141252 70:    MOV      PS,R1                ;SAVE CONTENTS OF PSW IN R1
9195 036524 022767 000100 141234 80:    CMP      #BIT06,CPEREG            ;IS CPU ERROR REGISTER CORRECT?
9196 036532 001403                BEQ      80
9197                                     ;
9198                                     ;ERROR! CPU ERROR REG NOT CORRECT
9199 036534 104000                ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
9200 036536 001130                .WORD 1130                        ;UNIQUE ERROR NUMBER
9201 036540 001127                .WORD CPUERR                      ;ADDRESS OF ERROR MESSAGE
9202 036542 022726 177776 80:    CMP      #PS,(SP)+            ;IS STACK CONTENTS CORRECT?
9203 036546 001403                BEQ      90
9204 036550 104000                ERROR                                ;ALL ERRORS TO TRAP TO EMT VECTOR
9205 036552 001131                .WORD 1131                        ;UNIQUE ERROR NUMBER
9206 036554 001127                .WORD CPUERR                      ;ADDRESS OF ERROR MESSAGE
9206 036556 022726 000000 90:    CMP      #0,(SP)+            ;IS STACK CONTENTS CORRECT?
    
```

9207	036562	001403				BEQ	100		
9208	036564	104000				ERROR			ALL ERRORS TO TRAP TO EMT VECTOR
9209	036566	001132				.WORD	1132		UNIQUE ERROR NUMBER
9210	036570	001127				.WORD	CPUERR		ADDRESS OF ERROR MESSAGE
9211	036572	022701	000340		100:	CMP	#340,R1		WAS PSW LOADED PROPERLY ON TRAP?
9212	036578	001403				BEQ	110		
9213	036600	104000				ERROR			ALL ERRORS TO TRAP TO EMT VECTOR
9214	036602	001133				.WORD	1133		UNIQUE ERROR NUMBER
9215	036604	001127				.WORD	CPUERR		ADDRESS OF ERROR MESSAGE
9216	036606	005067	141154		110:	CLR	CPEREG		CLEAR CPU ERROR REGISTER
9217	036612	016767	142212	141164		MOV	SLOC00,4		RESTORE VECTOR
9218	036620	016767	142206	141160		MOV	SLOC01,6		"
9219									
9220									
9221									
9222									
9223									
9224	036626								
9225	036626	005267	142152			INC	TST255		INCREMENT TEST NUMBER
9226	036632	013767	000004	142170		MOV	#04,SLOC00		SAVE VECTOR
9227	036640	012737	036722	000004		MOV	#21,#04		SET UP VECTOR
9228	036650	005001				CLR	R1		CLEAR R1 FOR USE AS A COUNTER
9229	036650	012706	000777		100:	MOV	#777,R6		SET UP THE STACK WITH ODD ADDRESS.
9230	036654	005067	141106			CLR	CPEREG		CLEAR THE CPU ERROR REGISTER
9231	036660	005067	141112			CLR	PSW		CLEAR THE PSW
9232	036664	005737	177700			TST	#177700		ACCESS NON-EXISTANT I/O ADDRESS
9233	036670	012706	001000		10:	MOV	#STBOT,R6		FIX STACK BEFORE ERROR
9234	036674	122737	000001	001020		CHPB	#APTENV,#0ENV		ARE WE IN APT MODE?
9235	036702	001004				BNE	110		NO GO TO ERROR
9236	036704	005701				TST	R1		HAVE WE BEEN THIS WAY BEFORE?
9237	036706	001002				BNE	110		GO TO ERROR IF R1 NOT= 0
9238	036710	005201				INC	R1		NO, GIVE IT ONE MORE CHANCE IN CASE
9239	036712	000756				BR	100		APT MONITORED AND SCREWED THINGS UP
9240	036714				110:				
9241	036714	104000				ERROR			ALL ERRORS TO TRAP TO EMT VECTOR
9242	036716	001134				.WORD	1134		UNIQUE ERROR NUMBER
9243	036720	001160				.WORD	BRDERR		ADDRESS OF ERROR MESSAGE
9244	036722	022706	000000		20:	CMP	#0,SP		IS R6 CORRECT?
9245	036726	001405				BEQ	30		BRANCH IF YES
9246	036730	012706	001000			MOV	#STBOT,R6		RESTORE R6 BEFORE ERROR
9247	036734	104000				ERROR			ALL ERRORS TO TRAP TO EMT VECTOR
9248	036736	001135				.WORD	1135		UNIQUE ERROR NUMBER
9249	036740	001160				.WORD	BRDERR		ADDRESS OF ERROR MESSAGE
9250	036742	022726	036670		30:	CMP	#10,(SP)+		IS DATA AT ADDR 0 CORRECT?
9251	036746	001405				BEQ	40		BRANCH IF YES
9252	036750	012706	001000			MOV	#STBOT,R6		RESTORE R6 BEFORE ERROR
9253	036754	104000				ERROR			ALL ERRORS TO TRAP TO EMT VECTOR
9254	036756	001136				.WORD	1136		UNIQUE ERROR NUMBER
9255	036760	001160				.WORD	BRDERR		ADDRESS OF ERROR MESSAGE
9256	036762	022716	000000		40:	CMP	#0,(SP)		IS PSW DATA IN ADDR 2 CORRECT?
9257	036766	001405				BEQ	50		BRANCH IF YES
9258	036770	012706	001000			MOV	#STBOT,R6		RESTORE R6 BEFORE ERROR
9259	036774	104000				ERROR			ALL ERRORS TO TRAP TO EMT VECTOR
9260	036776	001137				.WORD	1137		UNIQUE ERROR NUMBER
9261	037000	001160				.WORD	BRDERR		ADDRESS OF ERROR MESSAGE
9262	037002	022767	000124	140756	50:	CMP	#124,CPEREG		IS CPU ERROR REGISTER CORRECT?

```

9263 037010 001403          BEQ      61          ;BRANCH IF YES
9264 037012 104000          ERROR   ;ALL ERRORS TO TRAP TO EMT VECTOR
9265 037014 001140          .WORD  1140        ;UNIQUE ERROR NUMBER
9266 037016 001160          .WORD  BRDERR      ;ADDRESS OF ERROR MESSAGE
9267 037020 005067 140742    61:    CLR      CPEREG    ;CLEAR CPU ERROR REGISTER
9268 037024 012706 001000          MOV     #STBOT,SP  ;RESTORE STACK
9269 037030 016737 141774    000004  MOV     SLOCO0,B#4  ;RESTORE VECTOR
9270 037036 005037 000000          CLR     B#0        ;RESTORE ADDR 0
9271 037042 005037 000002          CLR     B#2        ;RESTORE ADDR 2
9272
9273
9274
9275
9276
9277
9278 037046
9279 037046 005267 141732          INC     TESTN      ;INCREMENT TEST NUMBER
9280 037052 016767 140726 141750  MOV     4,SLOCO0   ;SAVE CONTENTS OF VECTORS
9281 037060 012767 037106 140716  MOV     #PIRQNX,4  ;SETUP INTERRUPT VECTOR
9282 037066 005067 140674          CLR     CPEREG    ;CLEAR CPU ERROR REGISTER
9283 037072 005737 177772          TST     B#PIRQ    ;LOOK FOR REPLY FROM PIRQ
9284 037076 016767 141726 140700  MOV     SLOCO0,4  ;RESTORE VECTORS
9285 037104 000403          BR      PIRTEX    ;IF IT RESPONDS, CONTINUE TESTING.
9286                                     ;ERROR! NO RESPONSE FROM PIRQ REG
9287 037106
9288 037106 104000          PIRQNX: ERROR     ;ALL ERRORS TO TRAP TO EMT VECTOR
9289 037110 001141          .WORD  1141        ;UNIQUE ERROR NUMBER
9290 037112 001160          .WORD  BRDERR      ;ADDRESS OF ERROR MESSAGE
9291 037114
9292
9293
9294 037114
9295
9296
9297
9298 037114
9299 037114 005267 141664          PIR1:  INC     TESTN      ;INCREMENT TEST NUMBER
9300 037120 013767 000240 141702  MOV     B#PIRQVEC,SLOCO0 ;SAVE PIRQ VECTORS
9301 037126 013767 000242 141676  MOV     B#PIRQVEC+2,SLOCO1 ;
9302 037134 012737 037346 000240  MOV     #UNXP1R,B#PIRQVEC ;SET UP PIRQ VECTOR FOR UNEXPECTED INTERRUPT
9303 037142 012737 000340 000242  MOV     #P7,B#PIRQVEC+2 ;
9304 037150 012703 001000          MOV     #1000,R3  ;PUT 1000 IN R3: START TESTING
9305                                     ;BITS IN PIRQ REG BY FLOATING
9306                                     ;A BIT THROUGH BITS 9-15.
9307 037154 012704 037230          MOV     #PIRTBL,R4 ;SET UP R4 AS A POINTER TO EXPECTED
9308                                     ;ENCODED PRIORITY LEVELS IN PIRTL
9309 037160 000237          18:    SPL     7        ;DON'T ALLOW INTERRUPTS.
9310 037162 005037 177772          CLR     B#PIRQ    ;CLEAR OUT THE PIRQ
9311 037166 023724 177772          CMP     B#PIRQ,(R4)+ ;IS PIRQ OK??
9312 037172 001403          BEQ     21        ;BRANCH IF OK
9313                                     ;ERROR! PIRQ REG. WAS NOT CLEAR
9314 037174 104000          ERROR   ;ALL ERRORS TO TRAP TO EMT VECTOR
9315 037176 001142          .WORD  1142        ;UNIQUE ERROR NUMBER
9316 037200 001160          .WORD  BRDERR      ;ADDRESS OF ERROR MESSAGE
9317 037202 010337 177772    21:    MOV     R3,B#PIRQ ;SET A BIT IN PIRQ REGISTER
9318 037206 023724 177772          CMP     B#PIRQ,(R4)+ ;COMPARE THE ENCODED PRIORITY BITS

```



```

9319
9320 037212 001403          BEQ      3#
9321
9322
9323 037214 104000          ERROR
9324 037216 001143          .WORD   1143
9325 037220 001160          .WORD   BRDERR
9326 037222 006303          3#:    ASL      R3
9327 037224 103366          .WORD   2#
9328 037226 000410          .WORD   EXPIR1
9329
9330 037230 000000          PIRTB1: .WORD   0
9331 037232 001042          .WORD   1042
9332 037234 002104          .WORD   2104
9333 037236 004146          .WORD   4146
9334 037240 010210          .WORD   10210
9335 037242 020252          .WORD   20252
9336 037244 040314          .WORD   40314
9337 037246 100356          .WORD   100356
9338
9339 037250          EXPIR1:
9340
9341
9342
9343 037250          PIR2:
9344          ;*****
9345          ;ATEST 250      TEST PIRQ REGISTER LEVEL ENCODING
9346          ;*****
9347          ;THIS TEST IS TO CHECK THAT THE HIGHEST PRIORITY LEVEL SET IN THE
9348          ;PROGRAM INTERRUPT REQUEST BITS IS REFLECTED IN THE ENCODED PROGRAM
9349          ;INTERRUPT ACTIVE BITS.
9350          ;*****
9351 037250          TST260:
9352 037250 005267 141530          INC      #TESTN          ;INCREMENT TEST NUMBER
9353 037254 000237          SPI      7              ;SHUT OFF INTERRUPTS
9354 037256 005037 177772          CLR      @PIRQ          ;CLEAR PIRQ REGISTER
9355 037262 012703 177000          MOV      #177000,R3     ;SET UP DESIRED PATTERN IN R3
9356 037265 012704 037326          MOV      @PITBL1,R4    ;SETUP R4 AS A TABLE POINTER
9357 037272 010337 177772          1#:    MOV      R3,@PIRQ  ;SET PATERN IN PIRQ REGISTER
9358
9359 037276 023714 177772          CMP      @PIRQ,(R4)     ;COMPARE PATTERN IN PIRQ WITH PATTERN IN
9360          ;EXPECTED PATTERN TABLE.
9361 037302 001403          BEQ      2#            ;GO TO ERROR IF NOT THE SAME
9362          ;ERROR; PATTERNS WERE NOT THE SAME
9363 037304 104000          ERROR
9364 037306 001144          .WORD   1144
9365 037310 001160          .WORD   BRDERR
9366 037312 006003          2#:    ROR      R3
9367 037314 042703 000777          BIC      #777,R3
9368 037320 005724          TST     (R4)+
9369 037322 001363          BNE     1#
9370 037324 000413          BR      FIR2EX
9371
9372 037326 177356 077314 037252 PITBL1: .WORD   177356,77314,37252,17210,7146,3104,1042,0
9373 037334 017210 007146 003104
9374 037342 001042 000000

```

```

9375
9376 037346 UNXPIR: ;UNEXPECTED PIRQ INTERRUPT
9377 037346 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9378 037350 001145 .WORD 1145 ;UNIQUE ERROR NUMBER
9379 037352 001160 .WORD BRDERR ;ADDRESS OF ERROR MESSAGE
9380
9381 037354 PIR2EX:
9382
9383
9384 037354 PIR3:
9385 ;*****
9386 ;*TEST 261 TEST PIRQ INTERRUPTS
9387 ;*****
9388 ;THIS TEST CHECKS THAT EACH PROGRAM INTERRUPT OCCURS PROPERLY
9389 ;*****
9390 037354 TST261:
9391 037354 005267 141424 INC #TESTN ;INCREMENT TEST NUMBER
9392 037360 012703 001000 MOV #1000,R3 ;SETUP R3 AS A WORKING REGISTER
9393 037364 012737 037432 000240 MOV #PIRRTN,#PIRQVEC ;SET UP INTERRUPT ROUTINE AT PIR VECTOR
9394 037372 012737 000340 00:0242 MOV #340,#PIRQVEC+2 ;
9395 037400 005004 CLR R4 ;INITIALIZE R4 AS EXPECTED DATA HOLDER.
9396 037402 005724 PI1: TST (R4)+ ;INCREMENT R4 BY 2
9397 037404 050337 177772 BHS R3,#PIRQ ;SET PIRQ TO INTERRUPT
9398 037410 000230 SPL 0 ;ENABLE INTERRUPT TO OCCUR
9399 ;ERROR! INTERRUPT DID NOT OCCUR
9400 037412 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9401 037414 001146 .WORD 1146 ;UNIQUE ERROR NUMBER
9402 037416 001160 .WORD BRDERR ;ADDRESS OF ERROR MESSAGE
9403 037420 062706 000004 PI3: ADD #4,SP ;CLEAN UP THE STACK
9404 037424 006303 ASL R3 ;SHIFT R3 TO LEFT FOR NEXT INTERRUPT
9405 037426 103365 BCC PI1 ;END IF CARRY BIT IS SET
9406 037430 000415 BR PIR3EX ;ON TO THE NEXT TEST
9407
9408 037432 013705 177772 PIRRTN: MOV #PIRQ,R5 ;MOVE THE CONTENTS OF PIRQ REG TO R5
9409 037436 005037 177772 CLR #PIRQ ;KILL PIRQ INTERRUPT.
9410 037442 042705 177761 BIC #177761,R5 ;MASK OFF ALL BITS EXCEPT 1-4.
9411 037446 020405 CMP R4,R5 ;DID THE CORRECT LEVEL INTERRUPT OCCUR?
9412 037450 001403 BEQ PI2 ;IF YES RETURN FROM SUCCESSFUL
9413 ;INTERRUPT,GET READY FOR THE NEXT ONE.
9414 ;ERROR! INTERRUPT OCCURRED IN WRONG SEQUENCE
9415 037452 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9416 037454 001147 .WORD 1147 ;UNIQUE ERROR NUMBER
9417 037456 001160 .WORD BRDERR ;ADDRESS OF ERROR MESSAGE
9418 037460 000167 177734 PI2: JMP PI3 ;GET NEXT INTERRUPT
9419
9420 037464 PIR3EX:
9421
9422
9423
9424 037464 PIR4:
9425 ;*****
9426 ;*TEST 262 TEST PIRQ VS PSW INTERRUPT LEVEL
9427 ;*****
9428 ;THIS TEST IS TO ENSURE THAT PIRQ CANNOT INTERRUPT WHEN PSW INTERRUPT
9429 ;LEVEL IS SET TO BLOCK THEM OUT.
9430 ;*****

```



```

9487 037654 104000          ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
9488 037656 001151          .WORD      1151          ; UNIQUE ERROR NUMBER
9489 037660 001160          .WORD      BRDERR        ; ADDRESS OF ERROR MESSAGE
9490 037662 006303          4#: ASL      R3          ; SHIFT A "1" LEFT UNTIL INTERRUPT LEVEL
9491                                     ; IS REACHED.
9492 037664 103356          BCC      3#          ; BRANCH BACK IF CARRY IS NOT SET.
9493 037666 005337 001054   DEC      0#DCOUNT      ; LOWER INTERRUPT PRIORITY LEVEL
9494 037672 001346          BNE      1#          ; IF DCOUNT=0, WE'RE DONE.
9495 037674 000167 000102   JMP      PIR5EX        ; JUMP TO END OF THIS TEST.
9496
9497                                     ; PIRQ INTERRUPT SERVICE ROUTINE
9498
9499
9500 037700 013746 177772   10#: MOV      0#PIRQ, -(SP) ; SAVE PIRQ DATA ON STACK
9501 037704 005037 177772   CLR      0#PIRQ        ; SHUT OFF PIRQ INTERRUPTS
9502 037710 016601 000004   MOV      4(SP),R1      ; GET OLD PSW, SAVE IN R1.
9503 037714 011602          MOV      (SP),R2       ; GET PIRQ FROM STACK.
9504 037716 042702 177437   BIC      0#177437,R2   ; CLEAR UNWANTED BITS FROM R2
9505 037722 042701 177437   BIC      0#177437,R1   ; CLEAR UNWANTED BITS FROM R1
9506 037726 020102          CMP      R1,R2         ; R2 SHOULD BE > R1.
9507 037730 100403          BMI      20#          ; GO CHECK SEQUENCE OF INTERRUPT.
9508                                     ; ERROR! PRIORITY OF INTERRUPT WAS NOT
9509                                     ; HIGH ENOUGH, SHOULDN'T HAVE OCCURRED.
9510 037732 104000          ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
9511 037734 001152          .WORD      1152          ; UNIQUE ERROR NUMBER
9512 037736 001160          .WORD      BRDERR        ; ADDRESS OF ERROR MESSAGE
9513 037740 012602          20#: MOV      (SP)+,R2   ; POP OLD PIRQ OFF THE STACK.
9514 037742 042702 177761   BIC      0#177761,R2   ; CLEAR OFF EXTRANEIOUS BITS.
9515 037746 020402          CMP      R4,R2         ; SHOULD BE EQUAL.
9516 037750 001403          BEQ      21#          ; IF THEY ARE EQUAL, CLEAN UP STACK AND
9517                                     ; GET READY FOR THE NEXT INTERRUPT
9518                                     ; ELSE
9519                                     ; ERROR! INTERRUPT OCCURRED OUT OF SEQUENCE.
9520 037752 104000          ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
9521 037754 001153          .WORD      1153          ; UNIQUE ERROR NUMBER
9522 037756 001160          .WORD      BRDERR        ; ADDRESS OF ERROR MESSAGE
9523 037760 012716 037662   21#: MOV      0#4,(SP)  ; PUT RETURN ADDRESS ON THE STACK.
9524 037764 000002          RTI                     ; RETURN FROM INTERRUPT, RESTORE PSW.
9525
9526 037766 000300          PITBL2: .WORD      300   ; PRIORITY LEVEL 6
9527 037770 000240          .WORD      240         ; PRIORITY LEVEL 5
9528 037772 000200          .WORD      200         ; PRIORITY LEVEL 4
9529 037774 000140          .WORD      140         ; PRIORITY LEVEL 3
9530 037776 000100          .WORD      100         ; PRIORITY LEVEL 2
9531 040000 000040          .WORD      40          ; PRIORITY LEVEL 1
9532
9533 040002          PIR5EX:
9534
9535
9536
9537 040002          PIR6:
9538          ; *****
9539          ; *TEST 264 TEST THAT PIRQS ARE SERVICED IN CORRECT ORDER
9540          ; *****
9541          ; THIS TEST CHECKS THAT ALL PIRQ INTERRUPTS ARE SERVICED
9542          ; IN THE CORRECT DECENDING ORDER. I.E. IRQ6 IS NOT SERVICED

```

```

9543 ;BEFORE IRQ7 ETC. THE PIRQ IS LOADED WITH ALL INTERRUPTS SIMULTANEOUSLY
9544 ;TURNED ON. THE PSW PRIORITY LEVEL IS THEN LOWERED TO 0. EACH INTERRUPT
9545 ;SHOULD BE SERVICED IN DECENDING ORDER. EACH TIME AN INTERRUPT OCCURS, THE PSW
9546 ;IS LOADED WITH PRIORITY LEVEL 7 WHICH STOPS THE PIRQ FROM FURTHER INTERRUPTS.
9547 ;AFTER A CORRECT INTERRUPT, AN RTI IS EXECUTED, AND THE PSW PRIORITY LEVEL
9548 ;RETRUNS TO ZERO ALLOWING THE NEXT LOWER INTERRUPT TO OCCUR.
9549 ;*****
9550 TST264:
9551 040002 005267 140776 INC $TESTN ;INCREMENT TEST NUMBER
9552 040006 005037 177772 CLR $PIRQ ;CLEAR OUT THE PIRQ
9553 040012 000237 SPL 7 ;NO INTERRUPTS WILL BE SERVICED.
9554 040014 012737 040060 000240 MOV $101,$PIRQVEC ;SET UP VECTORS
9555 040022 012737 000340 000242 MOV $PR7,$PIRQVEC+2 ;
9556 040030 012703 177000 MOV $177000,R3 ;SETUP DESIRED PATTERN IN R3
9557 040034 010337 177772 MOV R3,$PIRQ ;MOVE PATTERN IN R3 TO PIRQ.
9558 040040 012704 000016 MOV $16,R4 ;PRELOAD R4 AS A DECREMENDING COUNTER.
9559 040044 000230 SPL 0 ;LET THE PIRQ INTERRUPT
9560 ;*****INTERRUPTS OCCUR HERE!!!*****
9561 040046 005704 TST R4 ;IS R4 ZERO? IT SHOULD BE. THIS
9562 ;INSTRUCTION SHOULDN'T BE EXECUTED
9563 ;UNTIL ALL INTERRUPTS HAVE OCCURRED.
9564 040050 001423 BEQ PIR6EX ;GO TO NEXT TEST.
9565 ;ERROR! ALL INTERRUPTS DID NOT OCCUR.
9566 040052 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9567 040054 001154 .WORD 1154 ;UNIQUE ERROR NUMBER
9568 040056 001160 .WORD BRDERR ;ADDRESS OF ERROR MESSAGE
9569 040060 013702 177772 10$: MOV $PIRQ,R2 ;SAVE THE PIRQ IN R2
9570 040064 042702 177761 BIC $177761,R2 ;STRIP OFF EXTRANEIOUS BITS.
9571 040070 020402 CMP R4,R2 ;SHOULD BE EQUAL.
9572 040072 001403 BEQ 15$ ;SETUP FOR NEXT INTERRUPT.
9573 ;ERROR! INTERRUPT WAS SERVICED OUT OF ORDER
9574 040074 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9575 040076 001155 .WORD 1155 ;UNIQUE ERROR NUMBER
9576 040100 001160 .WORD BRDERR ;ADDRESS OF ERROR MESSAGE
9577 040102 005744 15$: TST -(R4) ;DECREMENT R4 BY 2
9578 040104 006003 ROR R3 ;CHANGE PATTERN IN R3; LOWER INTERRUPT PRIORITY
9579 040106 042703 000777 BIC $777,R3 ;STRIP OFF UNNEEDED BITS
9580 040112 010337 177772 MOV R3,$PIRQ ;LOWER INTERRUPT PRIORITY LEVEL.
9581 040116 000002 RTI ;
9582
9583 040120 016737 140704 000240 PIR6EX: MOV SLOC00,$PIRQVEC ;RESTORE VECTORS
9584 040126 016737 140700 000242 MOV SLOC01,$PIRQVEC+2 ;
9585
9586
9587
9588
9589
9590 ;*****
9591 ;*TEST 265 SERIAL LINE UNIT INTERRUPT TEST
9592 ;*****
9592 040134 005267 140644 TST265:
9593 040134 005267 140644 INC $TESTN ;INCREMENT TEST NUMBER
9594 040140 122767 000001 140652 1$: CMPB $APTENV,$ENV ;ARE WE IN APT MODE
9595 040146 001003 BNE 2$ ;IF NOT THEN DO THIS TEST
9596 040150 005767 140632 YST $PASS ;ELSE:SEE IF THIS IS THE FIRST PASS
9597 040154 001057 BNE NOINTERRUPT ;IF NOT, SKIP THIS TEST
9598 040156 012767 000340 137612 2$: MOV $PR7,PS ;SET PRIORITY LEVEL 7

```

```

9599 040164 013767 000064 140636      MOV      @TPVEC,SLOC00      ;SAVE TPVEC
9600 040172 013767 000066 140632      MOV      @TPVEC+2,SLOC01  ;"
9601 040200 012767 040236 137656      MOV      @4,TPVEC         ;LOAD TTY PRINTER INTERRUPT VEC
9602 040206 012767 000340 137652      MOV      @PR7,TPVEC+2    ;LOAD TTY PRINTER INTERRUPT VEC
9603 040214 012767 000100 137342      MOV      @100,XCSR       ;ENABLE XMIT CSR INTERRUPT
9604 040222 012737 000000 177776      MOV      @0,@PSW         ;CLEAR CONDITION CODES
9605                                     ;SET PRIORITY LEVEL TO 0
9606                                     ;ERROR! NO INTERRUPT OCCURRED
9607 040230                                     3$:
9608 040230 104000      ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
9609 040232 001156      .WORD     1156          ;UNIQUE ERROR NUMBER
9610 040234 001160      .WORD     BRDERR       ;ADDRESS OF ERROR MESSAGE
9611 040236 005067 137522      4$: CLR      XCSR       ;DON'T ALLOW ANY MORE TTY INTERRUPTS
9612 040242 022716 040230      CMP      @3,(SP)       ;IS OLD PC CORRECT IN SP?
9613 040246 001403      BEQ      5$           ;BRANCH IF YES
9614                                     ;ERROR! INCORRECT PC ON STACK
9615 040250 104000      ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
9616 040252 001157      .WORD     1157          ;UNIQUE ERROR NUMBER
9617 040254 001160      .WORD     BRDERR       ;ADDRESS OF ERROR MESSAGE
9618 040256 022766 000000 000002 5$: CMP      @0,2(SP)      ;IS OLD PSW CORRECT ON STACK?
9619 040264 001403      BEQ      6$           ;BRANCH IF YES
9620                                     ;ERROR! INCORRECT PSW ON STACK
9621 040266 104000      ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
9622 040270 001160      .WORD     1160          ;UNIQUE ERROR NUMBER
9623 040272 001160      .WORD     BRDERR       ;ADDRESS OF ERROR MESSAGE
9624 040274 062706 000004      6$: ADD      @4,R6       ;RESET SP
9625 040300 016737 140524 000064      MOV      SLOC00,@TPVEC   ;RESTORE VECTORS
9626 040306 016737 140520 000066      MOV      SLOC01,@TPVEC+2 ;"
9627
9628 040314      NOINTERRUPT:
9629
9630
9631
9632 040314      EVENT:
9633      ;*****
9634      ;*TEST 266      TEST BEVENT REGISTER RESPONSE
9635      ;*****
9636      ;THIS TEST LOOKS FOR A REPLY FROM THE BEVENT CONTROL REGISTER.
9637      ;IF NO REPLY IS RECEIVED A TIME OUT TRAP OCCURS THRU LOC 000004
9638      ;*****
9639 040314      TST266:
9640 040314 005267 140464      INC      @TESTN        ;INCREMENT TEST NUMBER
9641 040320 013701 000004      1$: MOV      @4,R1       ;SAVE OLD VECTOR IN R1
9642 040324 013702 000006      MOV      @6,R2       ;SAVE LOC 6 IN R2
9643 040330 012737 040366 000004      MOV      @2,@4       ;SETUP TRAP THRU LOC 4
9644 040336 012737 000340 000006      MOV      @340,@6     ;IF BEVENT CTRL REG TIMES OUT
9645 040344 005067 137416      CLR      CPEREG       ;CLEAR CPU ERROR REGISTER
9646 040350 005737 177546      TST      @BEVENT      ;TEST FOR REPLY FROM BEVENT REG
9647                                     ;WILL TRAP THRU LOC 4 IF NO REPLY
9648 040354 010137 000004      MOV      R1,@4       ;RESTORE VECTORS
9649 040360 010237 000006      MOV      R2,@6
9650 040364 000403      BR      3$           ;GO TO NEXT TEST.
9651                                     ;ERROR! NO REPLY FROM BEVENT REGISTER.
9652 040366      2$:
9653 040366 104000      ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
9654 040370 001161      .WORD     1161          ;UNIQUE ERROR NUMBER

```

```

9655 040372 001160          .WORD  BRDERR          ;ADDRESS OF ERROR MESSAGE
9656 040374          3$:
9657
9658
9659
9660
9661
9662
9663
9664
9665 040374          ;*****
9666 040374 005267 140404          ;*TEST 267      TEST BIT 06 IN LKS
9667 040400 032767 000100 137140          ;*****
9668 040406 001411          ;THIS TEST CHECKS TO SEE THAT BIT06 IN THE BEVENT CONTROL REGISTER
9669 040410 005067 137132          ;CAN BE SET. PSW IS SET TO PRIORITY 7 SO THAT NO INTERRUPT OCCURS.
9670 040414 032767 000100 137124          ;*****
9671 040422 001403          TST267:
9672
9673 040424 104000          INC      $TESTN          ;INCREMENT TEST NUMBER
9674 040426 001162          BIT      @BIT06,BEVENT  ;TEST BIT06 TO SEE IF CLEAR
9675 040430 001160          BEQ     1$              ;
9676
9677 040432 002237          CLR     BEVENT          ;CLEAR BEVENT REGISTER
9678 040434 013767 000100 140366          BIT      @BIT06,BEVENT  ;IS IT CLEARED NOW?
9679 040442 013767 000102 140362          BEQ     1$              ;BRANCH IF IT IS
9680 040450 012737 040534 000100          ;ERROR! BIT06 SET AFTER BEING CLEARED
9681 040456 012737 000340 000102          ;ALL ERRORS TO TRAP TO EMT VECTOR
9682 040464 052767 000100 137054          .WORD   1162           ;UNIQUE ERROR NUMBER
9683 040472 032767 000100 137046          .WORD   BRDERR        ;ADDRESS OF ERROR MESSAGE
9684 040500 001003          1$:  SPL      7              ;SET PSW TO PR7 TO BLOCK INTERRUPTS
9685
9686
9687 040502 104000          MOV     @#100,SLOC00   ;SAVE BEVENT VECTORS
9688 040504 001163          MOV     @#102,SLOC01   ;
9689 040506 001160          MOV     @3$,@#100     ;SET UP VECTOR IN CASE WE GET AN
9690 040510 005067 137032          MOV     @PR7,@#102    ;UNEXPECTED INTERRUPT
9691 040514 032767 000100 137024          BIS     @BIT06,BEVENT  ;SET BIT06 IN BEVENT CTRL. REG TO ONE
9692 040522 001407          BIT     @BIT06,BEVENT  ;TEST BIT06 TO BE SURE IT IS SET
9693
9694 040524 104000          BNE    2$              ;
9695 040526 001164          ;ERROR! BIT06 WAS SUPPOSED TO BE SET
9696 040530 001160          ;BUT WASN'T
9697 040532 000403          ;ALL ERRORS TO TRAP TO EMT VECTOR
9698
9699 040534          ;UNIQUE ERROR NUMBER
9700 040534 104000          .WORD   1163           ;ADDRESS OF ERROR MESSAGE
9701 040536 001165          .WORD   BRDERR        ;CLEAR BEVENT REGISTER
9702 040540 001160          2$:  CLR     BEVENT          ;IS IT CLEAR?
9703 040542 013737 001030 000100          BIT     @BIT06,BEVENT  ;BRANCH IF IT IS
9704 040550 013737 001032 000102          BEQ     4$              ;ERROR! BIT06 WOULDN'T CLEAR OUT
9705
9706
9707
9708
9709
9710
9711
9712
9713
9714
9715
9716
9717
9718
9719
9720
9721
9722
9723
9724
9725
9726
9727
9728
9729
9730
9731
9732
9733
9734
9735
9736
9737
9738
9739
9740
9741
9742
9743
9744
9745
9746
9747
9748
9749
9750
9751
9752
9753
9754
9755
9756
9757
9758
9759
9760
9761
9762
9763
9764
9765
9766
9767
9768
9769
9770
9771
9772
9773
9774
9775
9776
9777
9778
9779
9780
9781
9782
9783
9784
9785
9786
9787
9788
9789
9790
9791
9792
9793
9794
9795
9796
9797
9798
9799
9800
9801
9802
9803
9804
9805
9806
9807
9808
9809
9810
9811
9812
9813
9814
9815
9816
9817
9818
9819
9820
9821
9822
9823
9824
9825
9826
9827
9828
9829
9830
9831
9832
9833
9834
9835
9836
9837
9838
9839
9840
9841
9842
9843
9844
9845
9846
9847
9848
9849
9850
9851
9852
9853
9854
9855
9856
9857
9858
9859
9860
9861
9862
9863
9864
9865
9866
9867
9868
9869
9870
9871
9872
9873
9874
9875
9876
9877
9878
9879
9880
9881
9882
9883
9884
9885
9886
9887
9888
9889
9890
9891
9892
9893
9894
9895
9896
9897
9898
9899
9900
9901
9902
9903
9904
9905
9906
9907
9908
9909
9910
9911
9912
9913
9914
9915
9916
9917
9918
9919
9920
9921
9922
9923
9924
9925
9926
9927
9928
9929
9930
9931
9932
9933
9934
9935
9936
9937
9938
9939
9940
9941
9942
9943
9944
9945
9946
9947
9948
9949
9950
9951
9952
9953
9954
9955
9956
9957
9958
9959
9960
9961
9962
9963
9964
9965
9966
9967
9968
9969
9970
9971
9972
9973
9974
9975
9976
9977
9978
9979
9980
9981
9982
9983
9984
9985
9986
9987
9988
9989
9990
9991
9992
9993
9994
9995
9996
9997
9998
9999

```

```

9711 ;THIS TEST LOOKS FOR BEVENT TO INTERRUPT WHEN BIT6 IN THE BEVENT
9712 ;CONTROL REGISTER IS SET AND INTERRUPT PRIORITY LEVEL IS SET TO 5
9713 ;|*****|
9714 TST270:
9715 040556 005267 140222 INC $TESTN ;INCREMENT TEST NUMBER
9716 040562 032777 002000 140256 BIT $BIT10,$SWR ;IF BIT 10 IN SWR = 1
9717 040570 001405 BEQ BEVINT ;THEN DON'T DO BEVENT INTERRUPT TESTS
9718 040572 062737 000005 001004 ADD #5,$TESTN ;ELSE: BUMP $TESTN AND
9719 040600 000167 000744 JMP NOBEVENT ;SKIP OVER BEVENT INTERRUPT TESTS
9720 040604 005067 136736 BEVINT: CLR BEVENT ;CLEAR BIT06 IN BEVENT CNTRL REG
9721 040610 000237 SPL 7 ;SET INTERRUPT LEVEL TO PRIORITY 7
9722 040612 013737 000100 001030 MOV $S100,$SLOC00 ;SAVE BEVENT VECTORS
9723 040620 013737 000102 001032 MOV $S102,$SLOC01 ;
9724 040626 012737 040666 000100 MOV $2,$S100 ;SETUP BEVENT INTERRUPT VECTOR
9725 040634 012737 000340 000102 MOV $340,$S102 ;
9726 040642 013701 001074 MOV $WAITIN,R1 ;
9727 ;SET UP A TIMER TO WAIT FOR INTERRUPT
9728 040646 052767 000100 136672 BIS $BIT06,BEVENT ;APPROX. 20 MILLISECS. WORST CASE 50HZ
9729 040654 000235 SPL 5 ;ENABLE BEVENT TO INTERRUPT
9730 040656 077101 1$: SOB R1,1$ ;SET INTERRUPT LEVEL TO PRIORITY 5
9731 ;DO OUR WAITING HERE
9732 040660 104000 ERROR ;INTERRUPT DIDN'T COME. IS LTC ON??
9733 040662 001166 .WORD 1166 ;ALL ERRORS TO TRAP TO EMT VECTOR
9734 040664 001160 .WORD BRDERR ;UNIQUE ERROR NUMBER
9735 ;ADDRESS OF ERROR MESSAGE
9736 040666 005067 136654 2$: CLR BEVENT ;DISABLE BEVENT INTERRUPTS
9737 040672 062706 000004 ADD #4,$SP ;CLEAN UP STACK AFTER INTERRUPT
9738 040676 013737 001030 000100 MOV $SLOC00,$S100 ;RESTORE VECTORS
9739 040704 013737 001032 000102 MOV $SLOC01,$S102 ;
9740 ;
9741 ;|*****|
9742 ;*TEST 271 TEST THAT BIT06 IN LKS CAN BE CLEARED
9743 ;|*****|
9744 ;THIS TEST CHECKS THAT BIT06 OF THE BEVENT CONTROL REGISTER CAN BE
9745 ;CLEARED. NO BEVENT INTERRUPTS SHOULD OCCUR AFTER BIT06 IS CLEARED
9746 ;|*****|
9747 TST271:
9748 040712 005267 140066 INC $TESTN ;INCREMENT TEST NUMBER
9749 040716 000237 SPL 7 ;SHUT OFF INTERRUPTS
9750 040720 052767 000100 136620 BIS $BIT06,BEVENT ;SET BIT06 OF BEVENT CNTRL REG
9751 040726 005067 136614 CLR BEVENT ;CLEAR BIT06 OF BEVENT CNTRL REG
9752 040732 032767 000100 136606 BIT $BIT06,BEVENT ;TEST TO SEE THAT IT CAN BE CLEARED
9753 040740 001403 BEQ 1$ ;RESULT OF BIT TEST SHOULD BE ZERO
9754 ;BIT06 OF BEVENT CNTRL REG WASN'T CLR
9755 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9756 040742 104000 .WORD 1167 ;UNIQUE ERROR NUMBER
9757 040744 001167 .WORD BRDERR ;ADDRESS OF ERROR MESSAGE
9758 040746 001160 1$:
9759 040750 ;
9760 ;|*****|
9761 ;*TEST 272 TEST THAT LKS CAN BLOCK BEVENT INTERRUPTS
9762 ;|*****|
9763 ;THIS TEST IS DESIGNED TO CHECK THAT BEVENT CANNOT INTERRUPT AFTER
9764 ;BIT06 IN THE BEVENT CNTRL REG HAS BEEN CLEARED.
9765
9766

```



```

9767
9768 040750
9769 040750 005267 140030
9770 040754 000237
9771 040756 005067 136564
9772 040752 013737 000100 001030
9773 040770 013737 000102 001032
9774 040776 012737 041024 000100
9775 041004 012737 000340 000102
9776 041012 012701 001074
9777
9778 041016 000235
9779
9780 041020 077101
9781 041022 000403
9782
9783 041024
9784 041024 104000
9785 041026 001170
9786 041030 001160
9787 041032 013737 001030 000100
9788 041040 013737 001032 000102
9789
9790
9791
9792
9793
9794
9795
9796
9797
9798
9799
9800
9801
9802 041046
9803 041046 005267 157732
9804 041052 000237
9805 041054 013737 000100 001030
9806 041062 013737 000102 001032
9807 041070 012737 041122 000100
9808
9809 041076 013701 001074
9810
9811
9812 041102 052767 000100 136436
9813 041110 077101
9814 041112 005067 136430
9815 041116 000235
9816
9817 041120 000403
9818
9819 041122
9820 041122 104000
9821 041124 001171
9822 041126 001160

```

```

;*****
TST272:
INC $TESTN ; INCREMENT TEST NUMBER
SPL 7 ; SHUT OFF INTERRUPTS
CLR BEVENT ; DISABLE BEVENT INTERRUPT
MOV @0100,@SLOC00 ; SAVE BEVENT VECTORS
MOV @0102,@SLOC01 ; " "
MOV @21,@0100 ; SET UP BEVENT INTERRUPT VECTOR IN
MOV @340,@0102 ; CASE WE GET AN UNWANTED INTERRUPT
MOV @WAITIN,R1 ; SET UP A DELAY TO WAIT FOR POSSIBLE
; UNWANTED INTERRUPT.
SPL 5 ; NOW LET'S SEE IF BEVENT CAN INTERRUPT.
; IT SHOULDN'T BE ABLE TO.
1$: SOB R1,1$ ; WE'LL WAIT ABOUT 20 MILLISECOND
BR 3$ ; THEN GO ON TO NEXT TEST.
; ERROR! INTERRUPT SHOULDN'T HAVE OCCURRED
2$:
ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
WORD 1170 ; UNIQUE ERROR NUMBER
WORD BRDERR ; ADDRESS OF ERROR MESSAGE
3$: MOV @SLOC00,@0100 ; RESTORE VECTORS
MOV @SLOC01,@0102 ; "
;*****
;*TEST 273 TEST THAT OLD BEVENT INTERRUPTS NOT SERVICED
;*****
;THIS TEST CHECKS THAT BEVENT INTERRUPTS WHICH OCCUR WHILE THE
;PROCESSOR PRIORITY LEVEL IS SET TO SEVEN WILL NOT BE SERVICED WHEN
;BIT06 IN THE BEVENT CONTROL REGISTER IS CLEARED, AND THE PRIORITY LEVEL
;IS LOWERED. THIS IS DONE BY: FIRST, ENABLING BEVENT INTERRUPT WITH THE
;INTERRUPT PRIORITY LEVEL SET TO 7, ALLOWING TIME FOR BEVENT TO OCCUR,
;THEN DISABLING BEVENT INTERRUPT AND DROPPING THE INTERRUPT
;PRIORITY LEVEL TO 5. BEVENT INTERRUPT SHOULD NOT OCCUR.
;*****
TST273:
INC $TESTN ; INCREMENT TEST NUMBER
SPL 7 ; SET INTERRUPT LEVEL TO 7
MOV @0100,@SLOC00 ; SAVE BEVENT VECTORS
MOV @0102,@SLOC01 ; " "
MOV @21,@0100 ; SET UP BEVENT VECTOR TO HANDLE
; UNEXPECTED INTERRUPT.
MOV @WAITIN,R1 ; SET UP TIMER TO ENSURE
; THAT THERE IS ENOUGH TIME TO
; ALLOW BEVENT TO INTERRUPT
1$: BIS @BIT06,BEVENT ; ENABLE BEVENT INTERRUPT
SOB R1,1$ ; WAIT HERE
CLR BEVENT ; DISABLE BEVENT INTERRUPT
SPL 5 ; DROP INTERRUPT LEVEL TO PRI 5
BR 3$ ; INTERRUPT SHOULD NOT OCCUR
; GO TO END OF TEST
; ERROR! BEVENT INTERRUPTED
2$:
ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
WORD 1171 ; UNIQUE ERROR NUMBER
WORD BRDERR ; ADDRESS OF ERROR MESSAGE

```

```

9823 041130 013737 001030 000100 3$: MOV @SLOC00,@#100 ;RESTORE VECTORS
9824 041136 013737 001032 000102 MOV @SLOC01,@#102 ;
9825
9826
9827 ;*****
9828 ;*TEST 274 TEST BEVENT INTERRUPT VS PSW PRIORITY LEVEL
9829 ;*****
9830 TST274:
9831 041144 005267 137634 INC $TESTN ;INCREMENT TEST NUMBER
9832 041150 005037 177776 CLR @#PS ;CLEAR THE PSW.
9833 041154 013737 000100 001030 MOV @#100,@SLOC00 ;SAVE BEVENT VECTORS
9834 041162 013737 000102 001032 MOV @#102,@SLOC01 ;
9835 041170 012737 041230 000100 MOV @INT4,@#100 ;SET UP INTERRUPT VECTORS
9836 041176 012737 000340 000102 MOV @340,@#102 ;
9837 041204 013701 001074 MOV @WAITIN,R1 ;SET UP R1 AS A COUNTER
9838 041210 000234 SPL 4 ;SET INTERRUPT PRIORITY LEVEL=4
9839 041212 052767 000100 136326 BIS @BIT06,BEVENT ;ENABLE BEVENT INTERRUPT
9840 041220 077101 1$: SOB R1,1$ ;WAIT HERE FOR INTERRUPT TO OCCUR.
9841 ;ERROR! INTERRUPT DIDN'T OCCUR AT PR4
9842 041222 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9843 041224 001172 .WORD 1172 ;UNIQUE ERROR NUMBER
9844 041226 001160 .WORD BRDERR ;ADDRESS OF ERROR MESSAGE
9845 041230 005067 136312 INT4: CLR BEVENT ;DISABLE BEVENT INTERRUPT.
9846 041234 062706 000004 ADD @4,SP ;CLEAN UP THE STACK AFTER INTERRUPT.
9847 041240 012737 041272 000100 BEV5: MOV @INT5,@#100 ;SET UP BEVENT INTERRUPT VECTOR.
9848 041246 013701 001074 MOV @WAITIN,R1 ;SETUP R1 AS A COUNTER.
9849 041252 000235 SPL 5 ;SET INTERRUPT PRIORITY LEVEL=5
9850 041254 052767 000100 136264 BIS @BIT06,BEVENT ;ENABLE BEVENT INTERRUPT
9851 041262 077101 2$: SOB R1,2$ ;WAIT HERE FOR INTERRUPT TO OCCUR.
9852 ;ERROR! INTERRUPT DIDN'T OCCUR AT PR5
9853 041264 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9854 041266 001173 .WORD 1173 ;UNIQUE ERROR NUMBER
9855 041270 001160 .WORD BRDERR ;ADDRESS OF ERROR MESSAGE
9856 041272 005067 135250 INT5: CLR BEVENT ;DISABLE BEVENT INTERRUPT.
9857 041276 062706 000004 ADD @4,SP ;CLEAN UP THE STACK AFTER INTERRUPT.
9858 041302 012737 041330 000100 BEV6: MOV @INT6,@#100 ;SET UP INTERRUPT VECTOR
9859 041310 013701 001074 MOV @WAITIN,R1 ;SETUP R1 AS A COUNTER.
9860 041314 000236 SPL 6 ;SET INTERRUPT PRIORITY LEVEL=6
9861 041316 052767 000100 136222 BIS @BIT06,BEVENT ;ENABLE BEVENT INTERRUPT.
9862 041324 077101 3$: SOB R1,3$ ;WAIT HERE FOR POSSIBLE INTERRUPT.
9863 041326 000403 BR BEV7 ;GOOD! IT DIDN'T INTERRUPT. GO ON.
9864 ;ERROR! BEVENT SHOULDN'T INTERRUPT IF
9865 ;PRIORITY LEVEL=6 OR GREATER.
9866 041330 INT6:
9867 041330 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
9868 041332 001174 .WORD 1174 ;UNIQUE ERROR NUMBER
9869 041334 001160 .WORD BRDERR ;ADDRESS OF ERROR MESSAGE
9870 041336 005067 136204 BEV7: CLR BEVENT ;DISABLE BEVENT INTERRUPT.
9871 041342 012737 041370 000100 MOV @INT7,@#100 ;SETUP INTERRUPT VECTOR.
9872 041350 013701 001074 MOV @WAITIN,R1 ;SETUP R1 AS A COUNTER.
9873 041354 000237 SPL 7 ;SET INTERRUPT PRIORITY LEVEL=7
9874 041356 052767 000100 136162 BIS @BIT06,BEVENT ;ENABLE BEVENT INTERRUPT
9875 041364 077101 4$: SOB R1,4$ ;WAIT HERE FOR POSSIBLE INTERRUPT
9876 041366 000403 BR INT0N ;GOOD! IT DIDN'T INTERRUPT. GO TO DONE.
9877 ;ERROR! BEVENT SHOULDN'T INTERRUPT IF
9878 ;PRIORITY LEVEL=7.

```

```

9879 041370          INT7:
9880 041370 104000          ERROR
9881 041372 001175          .WORD 1175
9882 041374 001160          .WORD BRDERR
9883 041376 005067 136144  INTDN: CLR BEVENT
9884 041402 013737 001030 000100 MOV 00SLOC00,00100
9885 041410 013737 001032 000102 MOV 00SLOC01,00102
9886
9887
9888
9889
9890
9891 041416          ;*****
9892 041416 005267 137362  ;+TEST 275 TEST WAIT INSTRUCTION
9893 041422 013737 000100 001030 ;*****
9894 041430 013737 000102 001032 TST275:
9895 041436 012737 041476 000100 INC #TESTN
9896 041444 012737 000240 000102 MOV 00100,00SLOC00
9897 041452 012701 001074          MOV 00102,00SLOC01
9898 041456 052767 000100 136062 MOV 020,00100
9899 041464 000235          MOV 0240,00102
9900 041466 077101          MOV #WAITIN,R1
9901 041470 104000          BIS #BIT06,BEVENT
9902 041472 001176          SPL 5
9903 041474 001160          SOB R1,10
9904 041476 062706 000004          ERROR
9905 041502 012737 041520 000100 .WORD 1176
9906 041510 000001          .WORD BRDERR
9907
9908          ;HOLD HERE FOR SECOND BEVENT. IF IT DOESN'T COME WE LL BE STUCK HERE.
9909
9910
9911          ;ERROR!
9912          ;SOME THING GOT SCREWED UP WITH WAIT
9913 041512 104000          ;IN ORDER TO GET HERE.
9914 041514 001177          ;ALL ERRORS TO TRAP TO EMT VECTOR
9915 041516 001127          ;UNIQUE ERROR NUMBER
9916 041520 005067 136022 30: CLR BEVENT
9917 041524 005067 136246          CLR PS
9918 041530 062706 000004          ADD #4,SP
9919 041534 016737 137270 000100 MOV 00SLOC00,00100
9920 041542 016737 137264 000102 MOV 00SLOC01,00102
9921 041550          NOBEVENT:
9922
9923
9924
9925 041550          MAINTR:
9926          ;*****
9927          ;+TEST 276 MAINTENANCE REGISTER TEST
9928          ;*****
9929 041550          TST276:
9930 041550 005267 137230          INC #TESTN
9931 041554 013767 000004 137246 MOV 004,SLOC00
9932 041562 013767 000006 137242 MOV 006,SLOC01
9933 041570 012737 041604 000004 MOV 010,004
9934 041576 013701 177750          MOV #0177750,R1

```



```

9968 .MCALL IDMSG,ENOPAS
9969 .SBTTL END OF PASS ROUTINE
9970
9971 ;*****
9972 ;*INCREMENT THE PASS NUMBER (#PASS)
9973 ;*INDICATE END-OF-PROGRAM AFTER 1 PASSES THRU THE PROGRAM
9974 ;*IF THERES A MONITOR GO TO IT
9975 ;*IF THERE ISN'T JUMP TO RESTART
9976
9977 #EOP:
9978 041730 005767 137052 TST #PASS ;ONLY TYPE MESSAGE AT END OF FIRST PASS
9979 041734 001002 BNE SKIPID ;IF >0 THEN SKIP THE ID MESSAGE
9980 041736 104401 042036 TYPE ,MSG1 ;ELSE TYPE THE ID MESSAGE
9981 041742
9982 041742 005267 137040 SKIPID: INC #PASS ;INCREMENT THE PASS NUMBER
9983 041746 042767 100000 137032 BIC #100000,#PASS ;DON'T ALLOW A NEG. NUMBER
9984 041754 005327 DEC (PC)+ ;LOOP?
9985 041756 000001 #EOPCT: .WORD 1
9986 041760 003022 BGT #DOAGN ;YES
9987 041762 012737 MOV (PC)+,R(PC)+ ;RESTORE COUNTER
9988 041764 000001 #ENDCT: .WORD 1
9989 041766 041756 #EOPCT
9990 041770 104401 042077 TYPE ,MSG2
9991 041774 015746 137006 MOV #PASS,-(SP) ;SAVE #PASS FOR TYPEOUT
9992 042000 104405 TYPDS ;GO TYPE--DECIMAL ASCII WITH SIGN
9993 042002 104401 042032 TYPE ,#NULL
9994 042006 013700 000042 #GET42: MOV #42,R0 ;GET MONITOR ADDRESS
9995 042012 001405 BEQ #DOAGN ;BRANCH IF NO MONITOR
9996 042014 000005 RESET ;CLEAR THE WORLD
9997 042016 004710 #ENDAD: JSR PC,(R0) ;GO TO MONITOR
9998 042020 000240 NOP ;SAVE ROOM
9999 042022 000240 NOP ;FOR
10000 042024 000240 NOP ;ACT11
10001 042026
10002 042026 000137 #DOAGN: JMP 2(PC) ;RETURN
10003 042030 001476 #RTNAD: .WORD RESTART
10004 042032 377 377 000 #NULL: .BYTE -1,-1,0 ;NULL CHARACTER STRING
10005 042036
10006 042036 005015 035103 042115 MSG1: .ASCIZ <CR><LF>/CZKDJ-B-0 KDJ11 CPU DIAGNOSTIC/
10007 042044 026512 026502 020060
10008 042052 042113 030512 020061
10009 042060 050103 020125 044504
10010 042066 043501 047516 052123
10011 042074 041511 000
10012 042077 015 041412 045532 MSG2: .ASCIZ <CR><LF>/CZKDJ-B END PASS #/
10013 042104 045104 020102 047105
10014 042112 020104 040520 051523
10015 042120 021440 000
10016 042124
10017 .EVEN
10018 .SBTTL TYPE ROUTINE
10019
10020 ;*****
10021 ;*ROUTINE TO TYPE ASCIZ MESSAGE. MESSAGE MUST TERMINATE WITH A 0 BYTE.
10022 ;*THE ROUTINE WILL INSERT A NUMBER OF NULL CHARACTERS AFTER A LINE FEED.
10023 ;*NOTE1: #NULL CONTAINS THE CHARACTER TO BE USED AS THE FILLER CHARACTER.
;*NOTE2: #FILLS CONTAINS THE NUMBER OF FILLER CHARACTERS REQUIRED.

```

```

10024      ;*NOTE3:          #FILLC CONTAINS THE CHARACTER TO FILL AFTER.
10025      ;*
10026      ;*CALL:
10027      ;+1) USING A TRAP INSTRUCTION
10028      ;*      TYPE      ,MESADR          ;;MESADR IS FIRST ADDRESS OF AN ASCIZ STRING
10029      ;*OR
10030      ;*      TYPE
10031      ;*      MESADR
10032      ;*
10033
10034      042124 105767 000343      ;*TYPE:  TSTB      #TPFLG          ;; IS THERE A TERMINAL?
10035      042130 100002              BPL          1#          ;; BR IF YES
10036      042132 000000              HALT              ;; HALT HERE IF NO TERMINAL.
10037      042134 000430              BR          3#          ;; LEAVE
10038      042136 010046              1#:  MOV        RO,-(SP)      ;; SAVE RO
10039      042140 017600 000002      MOV        #2(SP),RO        ;; GET ADDRESS OF ASCIZ STRING
10040      042144 122767 000001 136646 CMPB      #APTENV,#ENV      ;; RUNNING IN APT MODE
10041      042152 001011              BNE        62#          ;; NO,GO CHECK FOR APT CONSOLE
10042      042154 132767 000100 136637 BITB      #APTSPOOL,#ENVM   ;; SPOOL MESSAGE TO APT
10043      042162 001405              BEQ        62#          ;; NO,GO CHECK FOR CONSOLE
10044      042164 010067 000004      MOV        RO,61#         ;; SETUP MESSAGE ADDRESS FOR APT
10045      042170 004767 001034      JSR        PC,#ATY3       ;; SPOOL MESSAGE TO APT
10046      042174 000000              61#:  .WORD      0          ;; MESSAGE ADDRESS
10047      042176 132767 000040 136615 62#:  BITB      #APTCSUP,#ENVM   ;; APT CONSOLE SUPPRESSED
10048      042204 001003              BNE        60#          ;; YES,SKIP TYPE OUT
10049      042206 112046              2#:  MOVB      (RO)+,-(SP)    ;; PUSH CHARACTER TO BE TYPED ONTO STACK
10050      042210 001005              BNE        4#          ;; BR IF IT ISN'T THE TERMINATOR
10051      042212 005726              TST        (SP)+         ;; IF TERMINATOR POP IT OFF THE STACK
10052      042214 012600              60#:  MOV        (SP)+,RO      ;; RESTORE RO
10053      042216 062716 000002      3#:  ADD        #2,(SP)       ;; ADJUST RETURN PC
10054      042222 000002              RTI              ;; RETURN
10055      042224 122716 000011      4#:  CMPB      #HT,(SP)       ;; BRANCH IF <HT>
10056      042230 001430              BEQ        8#          ;;
10057      042232 122716 000200      CMPB      #CRLF,(SP)      ;; BRANCH IF NOT <CRLF>
10058      042236 001006              BNE        5#          ;;
10059      042240 005726              TST        (SP)+         ;; POP <CR><LF> EQUIV
10060      042242 104401              TYPE              ;; TYPE A CR AND LF
10061      042244 001257              #CRLF
10062      042246 105067 000202      CLRB      #CHARCNT       ;; CLEAR CHARACTER COUNT
10063      042252 000755              BR          2#          ;; GET NEXT CHARACTER
10064      042254 004767 000056      5#:  JSR        PC,#TYPEC      ;; GO TYPE THIS CHARACTER
10065      042260 126726 000206      6#:  CMPB      #FILLC,(SP)+    ;; IS IT TIME FOR FILLER CHARS.?
10066      042264 001350              BNE        2#          ;; IF NO GO GET NEXT CHAR.
10067      042266 016746 000176      MOV        #NULL,-(SP)    ;; GET # OF FILLER CHARS. NEEDED
10068
10069      042272 105366 000001      7#:  DECB      1(SP)          ;; AND THE NULL CHAR.
10070      042276 002770              BLT        6#          ;; DOES A NULL NEED TO BE TYPED?
10071      042300 004767 000032      JSR        PC,#TYPEC      ;; BR IF NO--GO POP THE NULL OFF OF STACK
10072      042304 105367 000144      DECB      #CHARCNT       ;; GO TYPE A NULL
10073      042310 000770              BR          7#          ;; DO NOT COUNT AS A COUNT
10074
10075      ;HORIZONTAL TAB PROCESSOR
10076
10077      042312 112716 000040      8#:  MOVB      #' ,(SP)       ;; REPLACE TAB WITH SPACE
10078      042316 004767 000014      9#:  JSR        PC,#TYPEC      ;; TYPE A SPACE
10079      042322 132767 000007 000124 BITB      #7,#CHARCNT     ;; BRANCH IF NOT AT
    
```

```

10080 042330 001572          BNE      9#          ;;TAB STOP
10081 042332 005726          TST      (SP)+      ;;POP SPACE OFF STACK
10082 042334 000724          BR       2#          ;;GET NEXT CHARACTER
10083 042336                #TYPEPC:
10084 042336 105777 000116          TSTB    @TKS        ;;CHAR IN KYBD BUFFER?      ;MJD001
10085 042342 170022          BPL     10#         ;;BR IF NOT                ;MJD001
10086 042344 017746 000112          MOV     @TKB,(SP)   ;;GET CHAR                ;MJD001
10087 042350 042716 177600          BIC     @177600,(SP) ;;STRIP EXTRANEIOUS BITS ;MJD001
10088 042354 122716 000023          CMPB   @XOFF,(SP)  ;;WAS CHAR XOFF          ;MJD001
10089 042360 001012          BNE     102#       ;;BR IF NOT                ;MJD001
10090 042362                101#:
10091 042362 105777 000072          TSTB    @TKS        ;;WAIT FOR CHAR            ;MJD001
10092 042366 100375          BPL     101#       ;;BR IF NOT                ;MJD001
10093 042370 117716 000066          MOVB   @TKB,(SP)   ;;GET CHAR                ;MJD001
10094 042374 042716 177600          BIC     @177600,(SP) ;;STRIP IT                ;MJD001
10095 042400 122716 000021          CMPB   @XON,(SP)   ;;WAS IT XON?           ;MJD001
10096 042404 001366          BNE     101#       ;;BR IF NOT                ;MJD001
10097 042406                102#:
10098 042406 005726          TST      (SP)+      ;;FIX STACK                ;MJD001
10099 042410                10#:
10100 042410 105777 000050          TSTB    @TPS        ;;WAIT UNTIL PRINTER IS READY ;MJD001
10101 042414 100375          BPL     10#         ;;BR IF NOT                ;MJD001
10102 042416 116677 000002 000042          MOVB   2(SP),@TPB  ;;LOAD CHAR TO BE TYPED INTO DATA REG.
10103 042424 122766 000015 000002          CMPB   @CR,2(SP)   ;;IS CHARACTER A CARRIAGE RETURN?
10104 042432 001003          BNE     1#         ;;BRANCH IF NO
10105 042434 105067 000014          CLRB   #CHARCNT    ;;YES--CLEAR CHARACTER COUNT
10106 042440 000406          BR      #TYPEPC    ;;EXIT
10107 042442 122756 000012 000002 1#:
10108 042450 001402          CMPB   @LF,2(SP)   ;;IS CHARACTER A LINE FEED?
10109 042452 105227          BEQ    #TYPEPC    ;;BRANCH IF YES
10110 042454 000000          INCB   (PC)+      ;;COUNT THE CHARACTER
10111 042456 000207          #CHARCNT: .WORD   0
10112                #TYPEPC: RTS      PC
10113 042460 177560          #TKS:   .WORD    177560  ;;TTY KDB STATUS          ;MJD001
10114 042462 177562          #TKB:   .WORD    177562  ;;TTY KBD BUFFER         ;MJD001
10115 042464 177564          #TPS:   .WORD    177564  ;;TTY PRINTER STATUS REG. ADDRESS
10116 042466 177566          #TPB:   .WORD    177566  ;;TTY PRINTER BUFFER REG. ADDRESS
10117 042470          #NULL:  .BYTE    0       ;;CONTAINS NULL CHARACTER FOR FILLS
10118 042471          #FILLS: .BYTE    2       ;;CONTAINS # OF FILLER CHARACTERS REQUIRED
10119 042472          #FILLC: .BYTE    12      ;;INSERT FILL CHARS. AFTER A "LINE FEED"
10120 042473          #PFLG: .BYTE    0       ;;"TERMINAL AVAILABLE" FLAG (BIT<07>*0=YES)
10121 042474          #QUES: .ASCII   "?"     ;;QUESTION MARK
10122 042475          #LF:   .ASCII   <12>    ;;LINEFEED
10123                .EVEN
10124                .SBTTL  CONVERT BINARY TO DECIMAL AND TYPE ROUTINE:
10125
10126                ;*****
10127                ;*THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 5-DIGIT
10128                ;*SIGNED DECIMAL (ASCII) NUMBER AND TYPE IT. DEPENDING ON WHETHER THE
10129                ;*NUMBER IS POSITIVE OR NEGATIVE A SPACE OR A MINUS SIGN WILL BE TYPED
10130                ;*BEFORE THE FIRST DIGIT OF THE NUMBER. LEADING ZEROS WILL ALWAYS BE
10131                ;*REPLACED WITH SPACES.
10132                ;*CALL:
10133                ;*      MOV     NUM,-(SP)      ;;PUT THE BINARY NUMBER ON THE STACK
10134                ;*      TYPDS                    ;;GO TO THE ROUTINE
10135

```

```

10136 042500                                #TYPDS:
10137 042500 010046                          MOV    R0,-(SP)           ;;PUSH R0 ON STACK
10138 042502 010146                          MOV    R1,-(SP)           ;;PUSH R1 ON STACK
10139 042504 010246                          MOV    R2,-(SP)           ;;PUSH R2 ON STACK
10140 042506 010346                          MOV    R3,-(SP)           ;;PUSH R3 ON STACK
10141 042510 010546                          MOV    R5,-(SP)           ;;PUSH R5 ON STACK
10142 042512 012746 020200                  MOV    #20200,-(SP)      ;;SET BLANK SWITCH AND SIGN
10143 042516 016605 000020                  MOV    20(SP),R5         ;;GET THE INPUT NUMBER
10144 042522 100004                          BPL    1#                ;;BR IF INPUT IS POS.
10145 042524 005405                          NEG    R5                 ;;MAKE THE BINARY NUMBER POS.
10146 042526 112766 000055 000001          MOVVB  #'-,1(SP)         ;;MAKE THE ASCII NUMBER NEG.
10147 042534 005000                          CLR    R0                 ;;ZERO THE CONSTANTS INDEX
10148 042536 012703 042714                  MOV    #DBLK,R5         ;;SETUP THE OUTPUT POINTER
10149 042542 112723 000040                  MOVVB  #'',(R3)+         ;;SET THE FIRST CHARACTER TO A BLANK
10150 042546 005002                          CLR    R2                 ;;CLEAR THE BCD NUMBER
10151 042550 016001 042704                  MOV    @DTBL(R0),R1      ;;GET THE CONSTANT
10152 042554 160105                          SUE   R1,R5              ;;FORM THIS BCD DIGIT
10153 042556 002402                          BLT    4#                ;;BR IF DONE
10154 042560 005202                          INC    R2                 ;;INCREASE THE BCD DIGIT BY 1
10155 042562 000774                          BR     3#                ;;
10156 042564 060105                          ADD    R1,R5             ;;ADD BACK THE CONSTANT
10157 042566 005702                          TST   R2                 ;;CHECK IF BCD DIGIT=0
10158 042570 001002                          BNE   5#                ;;FALL THROUGH IF 0
10159 042572 105716                          TSTB  (SP)               ;;STILL DOING LEADING 0'S?
10160 042574 100407                          BMI   7#                ;;BR IF YES
10161 042576 106316                          ASLB  (SP)               ;;MSD?
10162 042600 103003                          BCC   6#                ;;BR IF NO
10163 042602 116663 000001 177777          MOVVB  1(SP),-1(R3)      ;;YES--SET THE SIGN
10164 042610 052702 000060                  BIS   #'0,R2             ;;MAKE THE BCD DIGIT ASCII
10165 042614 052702 000040                  BIS   #' ,R2             ;;MAKE IT A SPACE IF NOT ALREADY A DIGIT
10166 042620 110223                          MOVVB  R2,(R3)+         ;;PUT THIS CHARACTER IN THE OUTPUT BUFFER
10167 042622 005720                          TST   (R0)+             ;;JUST INCREMENTING
10168 042624 020027 000010                  CMP   R0,#10            ;;CHECK THE TABLE INDEX
10169 042630 002746                          BLT   2#                ;;GO DO THE NEXT DIGIT
10170 042632 003002                          BGT   8#                ;;GO TO EXIT
10171 042634 010502                          MOV   R5,R2              ;;GET THE LSD
10172 042636 000764                          BR    6#                ;;GO CHANGE TO ASCII
10173 042640 105726                          8#:  TSTB  (SP)+         ;;WAS THE LSD THE FIRST NON ZERO?
10174 042642 100003                          BPL   9#                ;;BR IF NO
10175 042644 116663 177777 177776          MOVVB  1(SP), 2(R3)     ;;YES--SET THE SIGN FOR TYPING
10176 042652 105013                          9#:  CLRB  (R3)           ;;SET THE TERMINATOR
10177 042654 012605                          MOV   (SP)+,R5          ;;POP STACK INTO R5
10178 042656 012603                          MOV   (SP)+,R3          ;;POP STACK INTO R3
10179 042660 012602                          MOV   (SP)+,R2          ;;POP STACK INTO R2
10180 042662 012601                          MOV   (SP)+,R1          ;;POP STACK INTO R1
10181 042664 012600                          MOV   (SP)+,R0          ;;POP STACK INTO R0
10182 042666 104401 042714                  TYPE  ,DBLK              ;;NOW TYPE THE NUMBER
10183 042672 016666 000002 000004          MOV   2(SP),4(SP)       ;;ADJUST THE STACK
10184 042700 012616                          MOV   (SP)+,(SP)       ;;
10185 042702 000002                          RTI                      ;;RETURN TO USER
10186 042704 023420                          #DTBL: 10000,
10187 042706 001750                          1000,
10188 042710 000144                          100,
10189 042712 000012                          10,
10190 042714 000004                          #DBLK: ,BLKW 4
10191                                .SBTTL BINARY TO OCTAL (ASCII) AND TYPE
    
```



```

10192
10193
10194
10195
10196
10197
10198
10199
10200
10201
10202
10203
10204
10205
10206
10207
10208
10209
10210
10211
10212
10213
10214
10215
10216 042724 017646 000000
10217 042730 116667 000001 000211
10218 042736 112667 000207
10219 042742 062716 000002
10220 042746 000406
10221 042750 112767 000001 000171
10222 042756 112767 000006 000165
10223 042764 112767 000005 000154
10224 042772 010346
10225 042774 010446
10226 042776 010546
10227 043000 116704 000145
10228 043004 005404
10229 043006 062704 000006
10230 043012 110467 000132
10231 043016 116704 000125
10232 043022 016605 000012
10233 043026 005003
10234 043030 006105
10235 043032 000404
10236 043034 006105
10237 043036 006105
10238 043040 006105
10239 043042 010503
10240 043044 006103
10241 043046 105367 000076
10242 043052 100016
10243 043054 042703 177770
10244 043060 001002
10245 043062 005704
10246 043064 001403
10247 043066 005204

;*****
;THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 6-DIGIT
;OCTAL (ASCII) NUMBER AND TYPE IT.
;#TYPOS---ENTER HERE TO SETUP SUPPRESS ZEROS AND NUMBER OF DIGITS TO TYPE
;CALL:
;# MOV NUM, -(SP) ;NUMBER TO BE TYPED
;# TYPOS ;CALL FOR TYPEOUT
;# .BYTE N ;N=1 TO 6 FOR NUMBER OF DIGITS TO TYPE
;# .BYTE M ;M=1 OR 0
;# ;1=TYPE LEADING ZEROS
;# ;0=SUPPRESS LEADING ZEROS
;#
;#TYPON---ENTER HERE TO TYPE OUT WITH THE SAME PARAMETERS AS THE LAST
;#TYPOS OR #TYPOC
;CALL:
;# MOV NUM, -(SP) ;NUMBER TO BE TYPED
;# TYPON ;CALL FOR TYPEOUT
;#
;#TYPOC---ENTER HERE FOR TYPEOUT OF A 16 BIT NUMBER
;CALL:
;# MOV NUM, -(SP) ;NUMBER TO BE TYPED
;# TYPOC ;CALL FOR TYPEOUT
;#
;#TYPOS: MOV @ (SP), -(SP) ;PICKUP THE MODE
;# MOVB 1(SP), #OFILL ;LOAD ZERO FILL SWITCH
;# MOVB (SP)+, #OMODE+1 ;NUMBER OF DIGITS TO TYPE
;# ADD #2, (SP) ;ADJUST RETURN ADDRESS
;# BR #TYPON
;#TYPOC: MOVB #1, #OFILL ;SET THE ZERO FILL SWITCH
;# MOVB #6, #OMODE+1 ;SET FOR SIX(6) DIGITS
;#TYPON: MOVB #5, #OCNT ;SET THE ITERATION COUNT
;# MOV R3, -(SP) ;SAVE R3
;# MOV R4, -(SP) ;SAVE R4
;# MOV R5, (SP) ;SAVE R5
;# MOVB #OMODE+1, R4 ;GET THE NUMBER OF DIGITS TO TYPE
;# NEG R4
;# ADD #6, R4 ;SUBTRACT IT FOR MAX. ALLOWED
;# MOVB R4, #OMODE ;SAVE IT FOR USE
;# MOVB #OFILL, R4 ;GET THE ZERO FILL SWITCH
;# MOV 12(SP), R5 ;PICKUP THE INPUT NUMBER
;# CLR R5 ;CLEAR THE OUTPUT WORD
;# ROL R5 ;ROTATE MSB INTO "C"
;# BR 3# ;GO DO MSB
;# ROL R5 ;FORM THIS DIGIT
;# ROL R5
;# ROL R5
;# MOV R5, R3
;# ROL R3 ;GET LSB OF THIS DIGIT
;# DECB #OMODE ;TYPE THIS DIGIT?
;# BPL 7# ;BR IF NO
;# BIC #177770, R3 ;GET RID OF JUNK
;# BNE 4# ;TEST FOR 0
;# TST R4 ;SUPPRESS THIS 0?
;# BEQ 5# ;BR IF YES
;# INC R4 ;DON'T SUPPRESS ANYMORE 0'S
    
```

GLOBAL AREAS MACY11 30A(1052) 15-MAR-84 13:28 PAGE 190  
 KDJ11A.MAC 22-FEB-84 15:12 BINARY TO OCTAL (ASCII) AND TYPE

SEQ 0190

```

10248 043070 052703 000060
10249 043074 052703 000040      5$:   BIS      #'0,R3      ;;MAKE THIS DIGIT ASCII
10250 043100 110367 000040      ;   RIS      #' ,R3      ;;MAKE ASCII IF NOT ALREADY
10251 043104 104401 043144      ;   MOVB     R3,R#       ;;SAVE FOR TYPING
10252 043110 105367 000032      ;   TYPE     ,#         ;;GO TYPE THIS DIGIT
10253 043114 003347
10254 043116 002402      7$:   DECB     $OCNT      ;;COUNT BY 1
10255 043120 005204      ;   BGT      2#         ;;BR IF MORE TO DO
10256 043122 000744      ;   BLT      6#         ;;BR IF DONE
10257 043124 012605      ;   INC      R4         ;;INSURE LAST DIGIT ISN'T A BLANK
10258 043126 012604      ;   BR       2#         ;;GO DO THE LAST DIGIT
10259 043130 012603      6$:   MOV      (SP)+,R5    ;;RESTORE R5
10260 043132 016666 000002 000004 ;   MOV      (SP)+,R4    ;;RESTORE R4
10261 043140 012616      ;   MOV      (SP)+,R3    ;;RESTORE R3
10262 043142 000002      ;   MOV      2(SP),4(SP) ;;SET THE STACK FOR RETURNING
10263 043144      000      ;   MOV      (SP)+,(SP)
10264 043145      000      ;   RTI
10265 043146      000      ;   .BYTE    0           ;;RETURN
10266 043147      000      8$:   .BYTE    0           ;;STORAGE FOR ASCII DIGIT
10267 043150 000000      ;   .BYTE    0           ;;TERMINATOR FOR TYPE ROUTINE
10268      ;   $OCNT:   .BYTE    0           ;;OCTAL DIGIT COUNTER
10269      ;   $OFILL:  .BYTE    0           ;;ZERO FILL SWITCH
10270      ;   $OMODE:  .WORD    0           ;;NUMBER OF DIGITS TO TYPE
10271      ;   .SBTTL  TRAP DECODER
10272      ;
10273      ; *****
10274      ; *THIS ROUTINE WILL PICKUP THE LOWER BYTE OF THE "TRAP" INSTRUCTION
10275      ; *AND USE IT TO INDEX THROUGH THE TRAP TABLE FOR THE STARTING ADDRESS
10276      ; *OF THE DESIRED ROUTINE. THEN USING THE ADDRESS OBTAINED IT WILL
10277      ; *GO TO THAT ROUTINE.
10278      ;
10279      ;
10280      ;
10281      ;
10282      ;
10283      ;
10284      ;
10285      ;
10286      ;
10287      ;
10288      ;
10289      ;
10290      ;
10291      ;
10292      ;
10293      ;
10294      ;
10295      ;
10296      ;
10297      ;
10298      ;
10299      ;
10300      ;
10301      ;
10302      ;
10303      ;

```

```

10304      ;          ROUTINE
10305      ;          -----
10306      $TRPAD: .WORD    $TRAP2
10307      .ENDC
10308      .IIF NOF GNS, .NLIST
10309      A=      C
10310      .IIF NOF GNS, .LIST
10311      D          ;CALL=A          TRAP+D(C)          COMNT
10312      .ENDM    ##SET
10313      .MACRO   TRMTRP
10314      $TERM=, - $TRPAD
10315      .ENDM    TRMTRP
10316      .SBTTL   TRAP TABLE
10317
10318      ;*THIS TABLE CONTAINS THE STARTING ADDRESSES OF THE ROUTINES CALLED
10319      ;*BY THE "TRAP" INSTRUCTION.
10320
10321      ;          ROUTINE
10322      ;          -----
10323      $TRPAD: .WORD    $TRAP2
10324      $TYPE   ;CALL=TYPE          TRAP+1(104401)  TTY TYPEOUT ROUTINE
10325      $TYPOC  ;CALL=TYPOC        TRAP+2(104402)  TYPE OCTAL NUMBER (WITH LEADING ZEROS)
10326      $TYPOS  ;CALL=TYPOS        TRAP+3(104403)  TYPE OCTAL NUMBER (NO LEADING ZEROS)
10327      $TYPON  ;CALL=TYPON        TRAP+4(104404)  TYPE OCTAL NUMBER (AS PER LAST CALL)
10328      $TYPDS  ;CALL=TYPDS        TRAP+5(104405)  TYPE DECIMAL NUMBER (WITH SIGN)
10329
10330
10331      .SBTTL   APT COMMUNICATIONS ROUTINE:
10332
10333      ;:*****
10334      043222 112767 000001 000236 $ATY1:  MOVB    #1, $FFLG          ;TO REPORT FATAL ERROR
10335      043230 112767 000001 000226 $ATY3:  MOVB    #1, $HFLG          ;TO TYPE A MESSAGE
10336      043236 000403                    BR      $ATYC
10337      043240 112767 000001 000220 $ATY4:  MOVB    #1, $FFLG          ;TO ONLY REPORT FATAL ERROR
10338      043246                    $ATYC:
10339      043246 010046                    MOV     R0, -(SP)          ;PUSH R0 ON STACK
10340      043250 010146                    MOV     R1, -(SP)          ;PUSH R1 ON STACK
10341      043252 105767 000206                    TSTB   $HFLG             ;SHOULD TYPE A MESSAGE?
10342      043256 001450                    BEQ     5$                ;IF NOT: BR
10343      043260 122767 000001 135532                    CNPB   $APTENV, $ENV      ;OPERATING UNDER APT?
10344      043266 001031                    BNE     3$                ;IF NOT: BR
10345      043270 132767 000100 135523                    BITB   $APTSPOOL, $ENVM   ;SHOULD SPOOL MESSAGES?
10346      043276 001425                    BEQ     3$                ;IF NOT: BR
10347      043300 017600 000004                    MOV     #4(SP), R0        ;GET MESSAGE ADDR.
10348      043304 062766 000002 000004                    ADD     #2, 4(SP)         ;BUMP RETURN ADDR.
10349      043312 005767 135462                    1$:   TST     $MSGTYPE      ;SEE IF DONE W/ LAST XMISSION?
10350      043316 001375                    BNE     1$                ;IF NOT: WAIT
10351      043320 010067 135470                    MOV     R0, $MSGAD        ;PUT ADDR IN MAILBOX
10352      043324 105720                    2$:   TSTB   (R0)+         ;FIND END OF MESSAGE
10353      043326 001375                    BNE     2$                ;IF NOT: BR
10354      043330 166700 135460                    SUB     $MSGAD, R0        ;SUB START OF MESSAGE
10355      043334 006200                    ASR     R0                ;GET MESSAGE LGTH IN WORDS
10356      043336 010067 135454                    MOV     R0, $MSGLG        ;PUT LENGTH IN MAILBOX
10357      043342 012767 000004 135430                    MOV     #4, $MSGTYPE     ;TELL APT TO TAKE MSG.
10358      043350 000413                    BR      5$
10359      043352 017667 000004 000016 3$:   MOV     @4(SP), 4$        ;PUT MSG ADDR IN JSR LINKAGE

```

```

10360 043360 062766 000002 000004      ADD    #2,4(SP)          ;;BUMP RETURN ADDRESS
10361 043366 016746 134404      MOV    177776,-(SP)     ;;PUSH 177776 ON STACK
10362 043372 004767 176526      JSR    PC,#TYPE        ;;CALL TYPE MACRO
10363 043376 000000      4#:   .WORD            0
10364 043400      5#:
10365 043400 105767 000062      10#:  TSTB   #FFLG        ;;SHOULD REPORT FATAL ERROR?
10366 043404 001416      BEQ    12#             ;;IF NOT: BR
10367 043406 005767 135406      TST   #ENV            ;;RUNNING UNDER APT?
10368 043412 001413      BEQ    12#             ;;IF NOT: BR
10369 043414 005767 135360      11#:  TST   #MSGTYPE     ;;FINISHED LAST MESSAGE?
10370 043420 001375      BNE    11#             ;;IF NOT: WAIT
10371 043422 017667 000004 135352      MOV    #4(SP),#FATAL   ;;GET ERROR #
10372 043430 062766 000002 000004      ADD    #2,4(SP)          ;;BUMP RETURN ADDR.
10373 043436 005267 135336      INC   #MSGTYPE        ;;TELL APT TO TAKE ERROR
10374 043442 105067 000020      12#:  CLRB   #FFLG        ;;CLEAR FATAL FLAG
10375 043446 105067 000013      CLRB   #LFLG          ;;CLEAR LOG FLAG
10376 043452 105067 000006      CLRB   #MFLG          ;;CLEAR MESSAGE FLAG
10377 043456 012601      MOV    (SP)+,R1        ;;POP STACK INTO R1
10378 043460 012600      MOV    (SP)+,R0        ;;POP STACK INTO R0
10379 043462 000207      RTS    PC              ;;RETURN
10380 043464      000      #MFLG: .BYTE          0      ;;MESSG. FLAG
10381 043465      000      #LFLG: .BYTE          0      ;;LOG FLAG
10382 043466      000      #FFLG: .BYTE          0      ;;FATAL FLAG
10383      043470      .EVEN
10384      000200      APTSIZE=200
10385      000001      APTENV=001
10386      000100      APTSPOOL=100
10387      000040      APTCSUP=040
10388      ;;*****
10389      ;;THIS ROUTINE WILL INCREMENT THE ERROR COUNT AND THEN PASS THE UNIQUE
10390      ;;ERROR NUMBER TO THE APT ERROR ROUTINE TO BE REPORTED TO THE APT SYSTEM.
10391
10392 043470 005267 135356      #ERROR: INC   #ERFLG    ;;INCREMENT ERROR FLAG
10393 043474 001775      BEQ    #ERROR        ;;DON'T LET IT GO TO ZERO
10394 043476 005267 135342      INC   #ERRCNT        ;;INCREMENT THE ERROR COUNT
10395 043502 021627 001002      CMP    (SP), #1002   ;;IS ERROR FROM VECTOR AREA
10396 043506 101010      BHI    1#            ;;IF YES THEN
10397 043510 012767 007777 000106      MOV    #7777, 3#    ;;REPORT AN UNEXPECTED TRAP
10398 043516 012637 001062      MOV    (SP)+,#$SAVSP1 ;;SAVE UNEXPECTED TRAP DATA
10399 043522 012637 001064      MOV    (SP)+,#$SAVSP2 ;;AND RESTORE SP
10400 043526 000430      BR     2#            ;;ELSE
10401 043530 017667 000000 000066 1#:   MOV    #(SP), 3#    ;;REPORT UNIQUE ERROR NUMBER TO APT
10402 043536 011667 000072      MOV    (SP),101#    ;;SAVE ERROR PC
10403 043542 052716 000002      ADD    #2,(SP)       ;;GET OVER UNIQUE ERROR NUMBER FOR RETURN
10404 043546 017637 000000 043556 100#:  MOV    #(SP),#102#
10405 043554 104401      TYPE   #TYPE          ;;TYPE ERROR MESSAGE
10406 043556 000000      102#: .WORD            0
10407 043560 062716 000002      ADD    #2,(SP)       ;;GET OVER ERROR MESSAGE
10408 043564 104401 001226      TYPE   ,ERR1         ;;
10409 043570 016746 000030      MOV    3#,-(SP)     ;;PUSH UNIQUE ERROR NUMBER ON THE STACK
10410 043574 104402      TYPDC #TYPE          ;;TYPE CTAL. ERROR NUMBER
10411 043576 104401 001242      TYPE   ,ERR2         ;;
10412 043602 016746 000026      MOV    101#,-(SP)   ;;PUSH ERROR PC ON THE STACK
10413 043606 104402      TYPDC #TYPE          ;;TYPE THE ERROR PC
10414 043610 122767 000001 135202 2#:  CMPB   #APTENV,#ENV  ;;CHECK TO MAKE SURE WE'RE IN APT MODE
10415 043616 001004      BNE    5#            ;;IF YES THEN

```

L15

GLOBAL AREAS MACY11 30A(1052) 15-MAR-84 13:28 PAGE 193  
KDJ11A.MAC 22-FEB-84 15:12 APT COMMUNICATIONS ROUTINE

SEQ 0193

10416 043620 004767 177414  
10417 043624 000000  
10418 043626 000777  
10419 043630 000000  
10420 043632 000002  
10421 043634 000000  
10422 043636  
10423 043636 000010  
10424 043636 000001

JSR PC, DATY4  
3\$: .WORD 0  
4\$: BR 4\$  
5\$: HALT  
RTI  
101\$: .WORD 0  
\$PATCH: .BLKW 10  
.END

;GO REPORT ERROR TO APT  
;STORAGE FOR ERROR NUMBER  
;LOOP HERE AFTER REPORTING ERROR TO APT  
;IF NOT APT THEN HALT  
;ALLOW RECOVERY FROM HALT

ABASE = 000000	584	
ACCC = ***** U	4592	
ACDW1 = 000000	584	
ACDW2 = 000000	584	
ACPUOP = 000000	584	599
ADCC = ***** U	4548	
ADDW0 = 000000	584	
ADDW1 = 000000	584	
ADDW10 = 000000	584	
ADDW11 = 000000	584	
ADDW12 = 000000	584	
ADDW13 = 000000	584	
ADDW14 = 000000	584	
ADDW15 = 000000	584	
ADDW2 = 000000	584	
ADDW3 = 000000	584	
ADDW4 = 000000	584	
ADDW5 = 000000	584	
ADDW6 = 000000	584	
ADDW7 = 000000	584	
ADDW8 = 000000	584	
ADDW9 = 000000	584	
ADEVCT = 000000	584	590
ADEVN = 000000	584	595
AENV = 000000	584	596
AENVM = 000000	584	587
AFATAL = 000000	584	
ALCC = ***** U	4932	
ALLCTR = 001056	6260	
ALROTS = 022236	60710	
ALR1TS = 022340	61080	
ALR2TS = 022442	61450	
ALR3TS = 022544	61820	
ALR4TS = 022646	62190	
ALR5TS = 022750	62560	
ALTR0 = ***** U	5950	
ALTR1 = ***** U	6108	
ALTR2 = ***** U	6145	
ALTR3 = ***** U	6182	
ALTR4 = ***** U	6219	
ALTR5 = ***** U	6256	
AMADR1 = 000000	584	
AMADR2 = 000000	584	
AMADR3 = 000000	584	
AMADR4 = 000000	584	
AMAMS1 = 000000	584	
AMAMS2 = 000000	584	
AMAMS3 = 000000	584	
AMAMS4 = 000000	584	
AMSGAD = 000000	584	592
AMSGLG = 000000	584	593
AMSGTY = 000000	584	586
AMTYP1 = 000000	584	
AMTYP2 = 000000	584	
AMTYP3 = 000000	584	
AMTYP4 = 000000	584	







		6038	6044	6050	6056	6062	6067	6083	6090	6097	6104	6120	6127	6134
		6141	6157	6164	6171	6178	6194	6201	6208	6215	6231	6238	6245	6252
		6268	6275	6282	6289	6311	6318	6335	6342	6359	6366	6373	6444	6456
		6163	6475	6482	6489	6532	6538	6548	6555	6565	6589	6599	6606	6613
		6619	6645	6653	6661	6670	6679	6686	6693	6715	6725	6733	6741	6768
		6782	6865	6872	6881	6889	6900	6919	6959	6966	6975	6982	6993	7012
		7038	7045	7052	7059	7083	7090	7098	7106	7113	7232	7240	7250	7264
		7271	7278	7289	7296	7303	7316	7324	7331	7338	7349	7356	7363	7381
		7398	7395	7402	7410	7558	7565	7576	7583	7590	7603	7610	7617	7625
		7754	7761	7768	7775	7787	7794	7801	7808	7823	7830	7837	7844	7852
		8029	8038	8045	8053	8060	8067	8091	8102	8129	8140	8150	8175	8204
		8232	8259	8288	8317	8343	8374	8395	8400	8406	8412	8433	8440	8446
		8452	8474	8479	8485	8496	8501	8507	8527	8533	8539	8562	8568	8579
		8585	8606	8612	8618	8641	8647	8657	8663	8688	8697	8732	8738	8744
		8764	8770	8780	8786	8808	8814	8820	8844	8850	8860	8866	8888	8894
		8900	8922	8928	8938	8944	8966	8972	8978	8999	9005	9015	9021	9043
		9049	9055	9078	9084	9094	9100	9123	9128	9133	9138	9162	9167	9172
		9177	9193	9200	9205	9210	9215	9915						
CPUTST=	***** U	1	203	546	635	662	741	10006						
CR	= 000015	231#	658	662	667	674	676	679	10006	10012	10103	10115		
CKLF	= 000200	232#	10057	10115										
CTSCC	= ***** U	4500												
C121A	024346	6641	6647#											
C121B	024372	6649	6655#											
C121C	024416	6657	6663#											
DATRAM=	000001	1#	738											
DAO	= ***** U	2443												
DCCC	= ***** U	4457												
DCOUNT	001054	625#	9473*	9493*										
DDISP	= 177570	238#	622	712										
DISPLA	001050	622#	712*	720*										
DISPRE	000174	532#	720											
DPO	= ***** U	2421												
DPL27	= ***** U	2503												
DSMR	= 177570	237#	621	711										
DSO	= ***** U	2472												
EMTO	= ***** U	8213												
EMTOA	032374	8222	8228#											
EMTUB	032406	8224	8233#											
EMTVEC=	000050	626#	700*	701*										
ERRCNT	001044	620#	10394*											
ERRMSG	001102 G	658#												
ERRNUM=	001203	502#	764	768#	773	777#	790	794#	797	801#	817	821#	835	839#
		851	855#	868	872#	885	888#	892	895#	899	902#	906	909#	922
		925#	929	932#	936	939#	943	946#	959	962#	966	969#	973	976#
		950	983#	996	999#	1003	1006#	1010	1013#	1017	1020#	1033	1036#	1040
		1043#	1047	1050#	1054	1057#	1070	1073#	1077	1080#	1084	1087#	1091	1094#
		1107	1110#	1114	1117#	1121	1124#	1128	1131#	1144	1147#	1151	1154#	1158
		1161#	1165	1168#	1183	1187#	1191	1195#	1209	1213#	1218	1222#	1237	1241#
		1247	1251#	1267	1271#	1279	1283#	1299	1302#	1312	1316#	1333	1336#	1345
		1349#	1366	1369#	1376	1380#	1398	1401#	1410	1414#	1430	1433#	1442	1446#
		1469	1472#	1480	1484#	1494	1498#	1506	1510#	1529	1532#	1541	1545#	1566
		1570#	1578	1582#	1585	1589#	1605	1608#	1620	1624#	1647	1651#	1658	1662#
		1665	1669#	1676	1680#	1682	1686#	1710	1714#	1721	1725#	1728	1732#	1758
		1762#	1765	1769#	1772	1775#	1781	1785#	1788	1792#	1815	1819#	1822	1826#
		1830	1834#	1853	1857#	1864	1868#	1884	1888#	1906	1910#	1919	1923#	1942

CROSS REFERENCE TABLE -- USER SYMBOLS

1946#	1950	1954#	1973	1977#	1985	1989#	2010	2014#	2022	2026#	2048	2052#
2058	2062#	2083	2087#	2095	2099#	2122	2126#	2139	2143#	2167	2171#	2176
2180#	2206	2210#	2214	2218#	2239	2243#	2246	2250#	2272	2276#	2279	2283#
2301	2305#	2308	2312#	2334	2338#	2342	2346#	2367	2371#	2376	2380#	2404
2408#	2413	2417#	2435	2439#	2451	2462#	2465	2469#	2487	2191#	2496	2500#
2514	2518#	2522	2526#	2529	2533#	2554	2558#	2561	2565#	2569	2573#	2592
2596#	2601	2605#	2611	2615#	2621	2625#	2649	2653#	2659	2663#	2675	2679#
2699	2703#	2722	2726#	2744	2748#	2751	2755#	2760	2764#	2768	2772#	2790
2794#	2800	2804#	2809	2813#	2836	2840#	2845	2849#	2868	2872#	2877	2881#
2901	2905#	2911	2915#	2918	2922#	2941	2945#	2949	2953#	2958	2962#	2967
2971#	2992	2996#	2999	3003#	3022	3026#	3033	3037#	3042	3046#	3066	3070#
3074	3078#	3103	3107#	3111	3115#	3135	3139#	3142	3146#	3165	3169#	3178
3182#	3200	3204#	3215	3219#	3236	3240#	3252	3256#	3275	3279#	3286	3290#
3312	3316#	3338	3342#	3362	3366#	3386	3390#	3408	3412#	3428	3432#	3450
3454#	3470	3474#	3478	3482#	3490	3494#	3501	3505#	3509	3513#	3521	3525#
3529	3533#	3542	3546#	3550	3554#	3564	3568#	3577	3581#	3691	3694#	3716
3720#	3730	3734#	3740	3744#	3756	3760#	3764	3768#	3795	3799#	3807	3811#
3825	3827#	3835	3839#	3850	3854#	3862	3866#	3879	3883#	3891	3895#	3907
3911#	3919	3923#	3935	3939#	3947	3951#	3962	3966#	3974	3978#	4005	4009#
4025	4029#	4046	4050#	4065	4069#	4085	4088#	4093	4096#	4105	4108#	4111
4114#	4127	4130#	4135	4138#	4143	4146#	4151	4154#	4161	4164#	4169	4172#
4177	4180#	4185	4188#	4200	4203#	4208	4211#	4216	4219#	4231	4234#	4239
4242#	4247	4250#	4260	4263#	4268	4271#	4276	4279#	4287	4290#	4295	4298#
4303	4306#	4325	4329#	4338	4342#	4362	4366#	4375	4379#	4399	4403#	4413
4417#	4436	4440#	4449	4453#	4472	4476#	4492	4496#	4516	4520#	4529	4533#
4540	4544#	4563	4567#	4574	4578#	4584	4588#	4606	4610#	4620	4624#	4631
4635#	4653	4657#	4666	4670#	4680	4684#	4692	4696#	4701	4705#	4724	4728#
4737	4741#	4762	4766#	4781	4785#	4802	4806#	4812	4816#	4823	4827#	4833
4837#	4854	4858#	4864	4868#	4874	4878#	4884	4888#	4907	4910#	4912	4915#
4921	4924#	4926	4929#	4945	4949#	4955	4959#	4966	4970#	4976	4980#	4997
5001#	5008	5012#	5018	5022#	5030	5034#	5053	5057#	5068	5072#	5091	5095#
5104	5108#	5117	5121#	5147	5151#	5163	5167#	5176	5179#	5185	5188#	5208
5212#	5221	5225#	5247	5251#	5253	5257#	5260	5263#	5279	5282#	5287	5291#
5293	5296#	5300	5303#	5315	5318#	5327	5327#	5331	5334#	5337	5340#	5353
5356#	5370	5373#	5387	5390#	5404	5407#	5421	5424#	5438	5441#	5455	5458#
5472	5475#	5489	5492#	5506	5509#	5524	5527#	5532	5535#	5541	5544#	5550
5553#	5559	5562#	5568	5571#	5577	5580#	5586	5589#	5595	5598#	5604	5607#
5622	5625#	5631	5634#	5640	5643#	5649	5652#	5658	5661#	5667	5670#	5676
5679#	5685	5688#	5694	5697#	5703	5706#	5719	5722#	5725	5728#	5732	5736#
5740	5744#	5748	5751#	5755	5759#	5764	5767#	5770	5773#	5777	5781#	5785
5789#	5794	5797#	5800	5803#	5807	5811#	5815	5819#	5824	5827#	5830	5833#
5837	5841#	5845	5849#	5875	5878#	5888	5891#	5900	5903#	5908	5911#	5915
5918#	5922	5925#	5929	5932#	5936	5939#	5943	5946#	5965	5968#	5979	5982#
5985	5988#	5991	5994#	5997	6000#	6003	6006#	6009	6012#	6015	6018#	6036
6039#	6042	6045#	6048	6051#	6054	6057#	6060	6063#	6065	6068#	6081	6084#
6088	6091#	6095	6098#	6102	6105#	6118	6121#	6125	6128#	6132	6135#	6139
6142#	6155	6158#	6162	6165#	6169	6172#	6176	6179#	6192	6195#	6199	6202#
6206	6209#	6213	6216#	6229	6232#	6236	6239#	6243	6246#	6250	6253#	6266
6269#	6273	6276#	6280	6283#	6287	6290#	6309	6312#	6316	6319#	6333	6336#
6340	6343#	6357	6360#	6364	6367#	6371	6374#	6442	6445#	6454	6457#	6461
6464#	6473	6476#	6480	6483#	6487	6490#	6530	6533#	6536	6539#	6546	6549#
6553	6556#	6562	6566#	6586	6590#	6597	6600#	6604	6607#	6611	6614#	6617
6620#	6643	6646#	6651	6654#	6659	6662#	6668	6671#	6677	6680#	6684	6687#
6691	6694#	6713	6716#	6723	6726#	6731	6734#	6739	6742#	6766	6769#	6780
6783#	6843	6866#	6870	6873#	6879	6882#	6887	6890#	6898	6901#	6916	6920#
6957	6960#	6964	6967#	6973	6976#	6980	6983#	6991	6994#	7009	7013#	7036







MD4	=	*****	U	823		
MD5	=	*****	U	842		
MD6	=	*****	U	858		
MEMTO		032324		82130		
MET		034736		87920		
META		035000		88050	8815	
METH		035006		8800	88090	
METD		035022		8810	88150	
METF		035036		8816	88210	
METO		035056		88260		
METOA		035126		88410		
METOB		035134		8834	88450	
METOC		035152		8846	88510	
METOD		035176		88570		
METOE		035204		8852	88610	
METOF		035222		8862	88670	
ME100	=	*****	U	5343		
ME101	=	*****	U	5360		
ME102	=	*****	U	5376		
ME103	=	*****	U	5393		
ME104	=	*****	U	5410		
ME105	=	*****	U	5427		
ME106	=	*****	U	5444		
ME107	=	*****	U	5461		
ME110	=	*****	U	5478		
ME111	=	*****	U	5495		
ME112	=	*****	U	5512		
ME113	=	*****	U	5611		
ME113A	=	*****	U	5710		
ME114	=	*****	U	5852		
ME115	=	*****	U	6293		
ME116	=	*****	U	6396		
ME117	=	*****	U	6516		
ME120	=	*****	U	6570		
ME121	=	*****	U	6826		
ME122	=	*****	U	6752		
ME123	=	*****	U	6838		
ME124	=	*****	U	6926		
ME125	=	*****	U	7018		
ME126	=	*****	U	7214		
ME127	=	*****	U	7543		
ME130	=	*****	U	7737		
MIALA		035746		90280		
MIALA		035776		90400	9050	
MIALA		036004		9036	90440	
MIALA		036020		9045	90500	
MIALA		036034		9051	90560	
MILA		035502		89510		
MILA		035532		89630	8973	
MILAO		032656		82970		
MILB		035540		8959	89670	
MILD		035554		8968	89730	
MILF		035570		8974	89790	
MILIBO		032766		83250		
MILLO		032110		81570		
MILLOA		032160		8166	81710	

MILOB	032172	8168	8176*
MILO	035602	8982*	
MILOA	035640	8996*	
MILOB	035646	8990	9000*
MILOC	035664	9001	9006*
MILOD	035710	9012*	
MILOE	035716	9007	9016*
MILOF	035734	9017	9022*
MIOT	034476	8718*	
MIOTA	034524	8729*	8739
MIOTB	034532	8726	8733*
MIOTD	034546	8734	8739*
MIOTF	034562	8740	8745*
MIOTO	032216	8185*	
MITO	034574	8748*	
MITOA	034630	8761*	
MITOB	034636	8756	8765*
MITOC	034654	8766	8771*
MITOD	034700	8777*	
MITOE	034706	8772	8781*
MITOF	034724	8782	8787*
MJ	011512	3458*	
MJP	012434	3704*	
MJP17	012450	3713*	
MJP27	012476	3727*	
MJP27A	012514	3728	3735*
MJP37	012576	3736	3761*
MJP37A	012614	3762	3769*
MJP67	012524	3737*	3770
MJP67A	012542	3738	3745*
MJP67B	012556	3747	3751*
MJP77	012560	3751	3753*
MJP77E	012624	3754	3773*
MJA	013476	3983*	
MJR27	013536	3996*	4002
MJR27A	013576	4003	4010*
MJR27B	013616	4014*	4043
MJR37	013704	4013	4037*
MJR37A	013744	4044	4051*
MJR6A	013700	4034*	4062
MJR6B	013702	4033	4035*
MJR67	013620	4016*	4054
MJR67A	013660	4023	4030*
MJR77	013764	4022	4035
MJR77A	014024	4063	4070*
MJSI	036046	9061*	
MJSIA	036104	9075*	
MJSIB	036112	9069	9079*
MJSIC	036130	9080	9085*
MJSID	036154	9091*	
MJSIE	036162	9086	9095*
MJSIF	036200	9096	9101*
MJSR	012624	3777*	
MJSRA	012756	3814	3818*
MJSRB	013147	3870	3873*
MJSRC	013330	3927	3929*

4056\*

3888  
3944





MRL3	011024	32590	
MRL4	011114	32930	
MRL5	***** U	3319	
MRL6	011236	33450	
MRL7	011304	33690	
MRRB1	011454	34350	
MRRCC	016310	48410	
MRR0	011422	34150	
MRT	033630	85130	
MRTA	033656	85240	8534
MRTB	033664	8521	85280
MRTC	033700	8529	85340
MRTD	033714	8535	85400
MRTU	033726	85460	
MRT0A	033762	85590	
MRT0B	033770	8554	85630
MRT0C	034006	8564	85690
MRT0D	034032	85760	
MRT0E	034040	8570	85800
MRT0F	034056	8581	85860
MRTS	014030	40750	
MSBCC	016020	47090	
MSBCCC	016104	47450	
MSER	177744	4900	
MSG1	042036	9980	100060
MSG2	042077	9990	100120
MS0B	017372	52280	
MSPAA	005462	21030	
MSPAU	031372	80160	
MSPB	003044	11980	
MSPBB	005562	21460	
MSPC	003104	12250	
MSPD	003150	12540	
MSPED	005222	12860	
MSPF	003300	13190	
MSPG	003356	13520	
MSPH	003430	13830	
MSPI	003510	14170	
MSPJ	003564	14490	
MSPK	003730	15130	
MSPM	004012	15480	
MSPN	004110	15920	
MSP0	004172	16270	
MSPP	004326	16900	
MSPQ	004444	17370	
MSPR	004602	17950	
MSPS	004710	18380	
MSPT	004772	18720	
MSPU	005020	18920	
MSPV	005150	19580	
MSPVO	005100	19270	
MSPX	005230	19930	
MSPY	005314	20300	
MSPZ	005376	20660	
MSP0	003006	11710	
MSTB3	005646	21830	

MST0	033072	83510			
MST0E	033152	8358	83700		
MST0EE	033164	8369	83750		
MST4	005734	22210			
MST4B	006006	22530			
MST5	006062	22850			
MST5B	006130	23150			
MST6	006206	23490			
MST7	006270	23830			
MSW37	011356	33930			
MSXT	017134	51260			
MSXTCC	016736	50380			
MS11	007220	27060			
MS22	007562	28520			
MS33	010072	29740			
MS77	010446	31180			
MTP	034070	85900			
MTPA	034336	86700			
MTPAA	034426	8681	8682	8691	87000
MTPAE	034456	8705	87090		
MTPAH	034420	8680	86940		
MTPAL	034404	86840	8706*	8707*	8708
MTPB	034132	8598	86070		
MTPF	034162	8614	86190		
MTPD	034174	86250			
MTPDA	034230	86380			
MTPDB	034236	8633	86420		
MTPDC	034254	8643	86460		
MTPDD	034300	86540			
MTPDE	034306	8649	86580		
MTPDF	034324	8659	86640		
MTPG	034146	8608	86130		
MTPR	034124	86030	8613		
MTRPO	032432	82350			
MTRY	031570	80710			
MTRYA	031654	8081	80920		
MTRYB	031700	8096	81030		
MTRYM	031732	81130			
MTSO	014746	4222	43100		
MTT	033176	83800			
MTTA	033242	8389	83970	8407	
MTTB	033250	8390	84010		
MTTD	033264	8402	84070		
MTTE	033300	8408	84130		
MTRR	033430	84570			
MTRRA	033504	8466	84760		
MTRRB	033512	8467	84800		
MTRRC	033530	8481	84860		
MTRD	033572	8488	84980		
MTRF	033600	8489	85020		
MTRF	033616	8503	85080		
MTTS	033312	84180			
MTTSA	033356	8427	84340		
MTTSB	033366	8428	84410		
MTTSD	033402	8442	84470		
MTTSE	033416	8448	84530		



