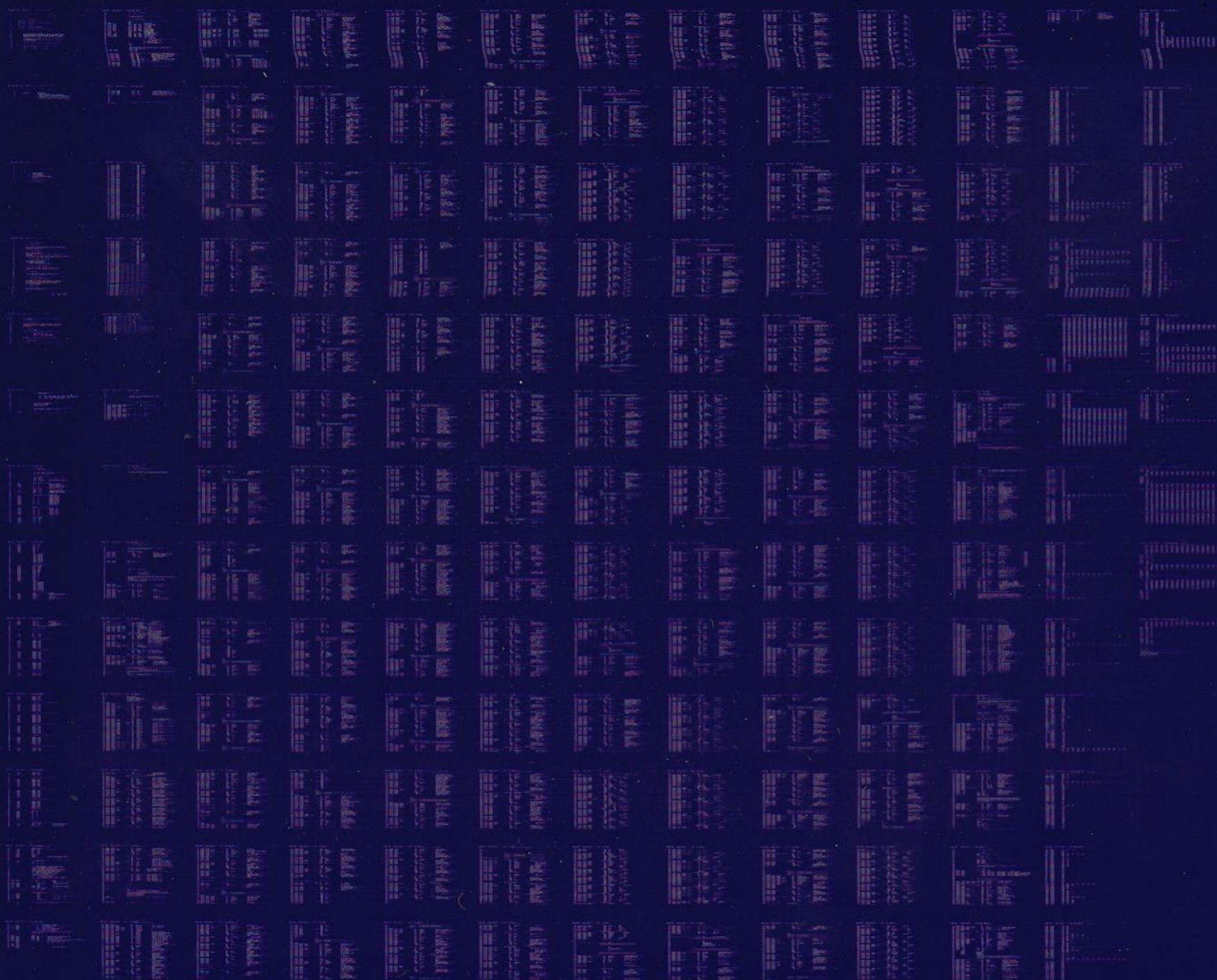


KDJ11-A

KDJ11 FPP DIAG
CZKDLBO

COPYRIGHT (c) 1983-84
AH-T709B-MC
FICHE 01 OF 01

JUL 1984
digital
Made In USA



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	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	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90
91	92	93	94	95	96	97	98	99	100

.REM &

IDENTIFICATION

PRODUCT CODE: AC-T708B-MC
PRODUCT NAME: CZKDLBO KDJ11 FLOATING POINT 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	MASSBUS
DEC	DECUS	DECTAPE	

&

39
40
41
42
43
44
45
46
47
48
49
50
51

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

HISTORY

.REM &

FIRST RELEASE
CORRECTIONS MADE TO:
1. CORRECT VECTOR AREA MAINTENANCE PROBLEM
2. TURN CACHE MEMORY SYSTEM OFF DURING NON-CACHE TESTS.
3. ENSURE THAT CPU ERROR REGISTER IS CLEARED AFTER
COMPLETION OF TEST THAT MIGHT CAUSE IT TO BE SET.
4. SAVE PC AND CONTENTS OF R6 ON UNEXPECTED INTERRUPTS

&

52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70

.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

&

.REM 6

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

THIS IS AN APT COMPATIBLE VERSION OF THE KDJ11 FLOATING POINT INSTRUCTION TESTS. IT FOCUSES ON TESTING THE KDJ11 FLOATING POINT INSTRUCTION FUNCTIONALITY.

1.2 SYSTEM REQUIREMENTS

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

1.3 RELATED DOCUMENTS AND STANDARDS

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

1.4 DIAGNOSTIC HIERARCY PREREQUISITES

THE KDJ11 CPU AND MEMORY MANAGEMENT DIAGNOSTICS MUST RUN SUCCESSFULLY PRIOR TO RUNNING THE FLOATING POINT TESTS.

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 XXDP+ 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

127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155

THE SOFTWARE SWITCH REGISTER HAS NO EFFECT ON THE OPERATION OF
THIS DIAGNOSTIC.

2.3 OPERATION UNDER APT

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

3.0 ERROR INFORMATION

ERRORS WILL CAUSE THE FOLLOWING ERROR MESSAGE TO BE PRINTED:

ERROR DURING FPP TESTING
ERROR # = (UNIQUE ERROR NUMBER)
ERROR PC = (PC AT TIME OF ERROR)

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

4.0 PROGRESS REPORT

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

E

156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176

.TITLE PROGRAM HEADER AND TABLES
.SBTTL PROGRAM HEADER

.MCALL NEWTST,ERRDEF,.EQUAT,.KT11,.\$40CAT,.\$EOP,.\$APTBL5,SETUP
.MCALL . \$TYPE,.\$TYPDEC,ERRDF,BGNTST,ENDTST,BGNMOD,ENDMOD,CKLOOP
.MCALL .HEADER,.SETUP,.\$TRAP,BGNSUB,ENDSUB,.\$ACT11,.\$APTHDR
.MCALL . \$ATYPE,.\$ERROR,.\$TYPOCT,.\$READ

.TITLE KDJ11-A FLOATING POINT DIAGNOSTIC
: *COPYRIGHT (C) OCTOBER,1983
: *DIGITAL EQUIPMENT CORP.
: *MAYNARD, MASS. 01754

: *
: *
: *THIS PROGRAM WAS ASSEMBLED USING THE PDP-11 MAINDEC SYSMAC
: *PACKAGE (MAINDEC-11-DZQAC-C3), JAN 19, 1977.
: *

000001
160000

\$TN=1
\$SWR=160000 ;:HALT ON ERROR, LOOP ON TEST, INHIBIT ERROR TYP0UT

```
177 .TITLE GLOBAL AREAS
178 .SBTTL GLOBAL EQUATES SECTION
179
180 ***
181 ; THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
182 ; ARE USED IN MORE THAN ONE TEST.
183 ;--
184 .SBTTL BASIC DEFINITIONS
185
186 ;*INITIAL ADDRESS OF THE STACK POINTER *** 1000 ***
187 001000 STACK= 1000
188 .EQUIV EMT,ERROR ;;BASIC DEFINITION OF ERROR CALL
189 .EQUIV IOT,SCOPE ;;BASIC DEFINITION OF SCOPE CALL
190
191 ;*MISCELLANEOUS DEFINITIONS
192 000011 MT= 11 ;;CODE FOR HORIZONTAL TAB
193 000012 LF= 12 ;;CODE FOR LINE FEED
194 000015 CR= 15 ;;CODE FOR CARRIAGE RETURN
195 000200 CRLF= 200 ;;CODE FOR CARRIAGE RETURN-LINE FEED
196 177776 PS= 177776 ;;PROCESSOR STATUS WORD
197 .EQUIV PS,PSW
198 177774 STKLM= 177774 ;;STACK LIMIT REGISTER
199 177772 PIRQ= 177772 ;;PROGRAM INTERRUPT REQUEST REGISTER
200 177570 DSWR= 177570 ;;HARDWARE SWITCH REGISTER
201 177570 DDISP= 177570 ;;HARDWARE DISPLAY REGISTER
202
203 ;*GENERAL PURPOSE REGISTER DEFINITIONS
204 000000 R0= #0 ;;GENERAL REGISTER
205 000001 R1= #1 ;;GENERAL REGISTER
206 000002 R2= #2 ;;GENERAL REGISTER
207 000003 R3= #3 ;;GENERAL REGISTER
208 000004 R4= #4 ;;GENERAL REGISTER
209 000005 R5= #5 ;;GENERAL REGISTER
210 000006 R6= #6 ;;GENERAL REGISTER
211 000007 R7= #7 ;;GENERAL REGISTER
212 000006 SP= #6 ;;STACK POINTER
213 000007 PC= #7 ;;PROGRAM COUNTER
214
215 ;*PRIORITY LEVEL DEFINITIONS
216 000000 PR0= 0 ;;PRIORITY LEVEL 0
217 000040 PR1= 40 ;;PRIORITY LEVEL 1
218 000100 PR2= 100 ;;PRIORITY LEVEL 2
219 000140 PR3= 140 ;;PRIORITY LEVEL 3
220 000200 PR4= 200 ;;PRIORITY LEVEL 4
221 000240 PR5= 240 ;;PRIORITY LEVEL 5
222 000300 PR6= 300 ;;PRIORITY LEVEL 6
223 000340 PR7= 340 ;;PRIORITY LEVEL 7
224
225 ;*"SWITCH REGISTER" SWITCH DEFINITIONS
226 100000 SW15= 100000
227 040000 SW14= 40000
228 020000 SW13= 20000
229 010000 SW12= 10000
230 004000 SW11= 4000
231 002000 SW10= 2000
232 001000 SW09= 1000
```



```
233      000400      SW08= 400
234      000200      SW07= 200
235      000100      SW06= 100
236      000040      SW05= 40
237      000020      SW04= 20
238      000010      SW03= 10
239      000004      SW02= 4
240      000002      SW01= 2
241      000001      SW00= 1
242      .EQUIV      SW09,SW9
243      .EQUIV      SW08,SW8
244      .EQUIV      SW07,SW7
245      .EQUIV      SW06,SW6
246      .EQUIV      SW05,SW5
247      .EQUIV      SW04,SW4
248      .EQUIV      SW03,SW3
249      .EQUIV      SW02,SW2
250      .EQUIV      SW01,SW1
251      .EQUIV      SW00,SW0
252
253      ;*DATA BIT DEFINITIONS (BIT00 TO BIT15)
254      100000      BIT15= 100000
255      040000      BIT14= 40000
256      020000      BIT13= 20000
257      010000      BIT12= 10000
258      004000      BIT11= 4000
259      002000      BIT10= 2000
260      001000      BIT09= 1000
261      000400      BIT08= 400
262      000200      BIT07= 200
263      000100      BIT06= 100
264      000040      BIT05= 40
265      000020      BIT04= 20
266      000010      BIT03= 10
267      000004      BIT02= 4
268      000002      BIT01= 2
269      000001      BIT00= 1
270      .EQUIV      BIT09,BIT9
271      .EQUIV      BIT08,BIT8
272      .EQUIV      BIT07,BIT7
273      .EQUIV      BIT06,BIT6
274      .EQUIV      BIT05,BIT5
275      .EQUIV      BIT04,BIT4
276      .EQUIV      BIT03,BIT3
277      .EQUIV      BIT02,BIT2
278      .EQUIV      BIT01,BIT1
279      .EQUIV      BIT00,BIT0
280
281      ;*BASIC "CPU" TRAP VECTOR ADDRESSES
282      000004      ERRVEC= 4      ;;TIME OUT AND OTHER ERRORS
283      000010      RESVEC= 10     ;;RESERVED AND ILLEGAL INSTRUCTIONS
284      000014      TBITVEC= 14    ;; "T" BIT
285      000014      TRTVEC= 14     ;;TRACE TRAP
286      000014      BPTVEC= 14    ;;BREAKPOINT TRAP (BPT)
287      000020      IOTVEC= 20    ;;INPUT/OUTPUT TRAP (IOT) **SCOPE**
288      000024      PWRVEC= 24    ;;POWER FAIL
```

```
289      000030      EMTVEC= 30      ;;EMULATOR TRAP (EMT) **ERROR**
290      000034      TRAPVEC=34      ;; "TRAP" TRAP
291      000060      TKVEC= 60      ;;TTY KEYBOARD VECTOR
292      000064      TPVEC= 64      ;;TTY PRINTER VECTOR
293      000240      PIRQVEC=240      ;;PROGRAM INTERRUPT REQUEST VECTOR
294      .SBTTL      MEMORY MANAGEMENT DEFINITIONS
295
296      ;*KT11 VECTOR ADDRESS
297
298      000250      MMVEC= 250
299
300      ;*KT11 STATUS REGISTER ADDRESSES
301
302      177572      SR0= 177572
303      177574      SR1= 177574
304      177576      SR2= 177576
305      172516      SR3= 172516
306
307      ;*USER "I" PAGE DESCRIPTOR REGISTERS
308
309      177600      UIPDR0= 177600
310      177602      UIPDR1= 177602
311      177604      UIPDR2= 177604
312      177606      UIPDR3= 177606
313      177610      UIPDR4= 177610
314      177612      UIPDR5= 177612
315      177614      UIPDR6= 177614
316      177616      UIPDR7= 177616
317
318      ;*USER "D" PAGE DESCRIPTOR REGISTORS
319
320      177620      UDPDR0= 177620
321      177622      UDPDR1= 177622
322      177624      UDPDR2= 177624
323      177626      UDPDR3= 177626
324      177630      UDPDR4= 177630
325      177632      UDPDR5= 177632
326      177634      UDPDR6= 177634
327      177636      UDPDR7= 177636
328
329      ;*USER "I" PAGE ADDRESS REGISTERS
330
331      177640      UIPAR0= 177640
332      177642      UIPAR1= 177642
333      177644      UIPAR2= 177644
334      177646      UIPAR3= 177646
335      177650      UIPAR4= 177650
336      177652      UIPAR5= 177652
337      177654      UIPAR6= 177654
338      177656      UIPAR7= 177656
339
340      ;*USER "D" PAGE ADDRESS REGISTERS
341
342      177660      UDPAR0= 177660
343      177662      UDPAR1= 177662
344      177664      UDPAR2= 177664
```


345	177666	UDPAR3= 177666
346	177670	UDPAR4= 177670
347	177672	UDPAR5= 177672
348	177674	UDPAR6= 177674
349	177676	UDPAR7= 177676
350		
351		;*SUPERVISOR "I" PAGE DESCRIPTOR REGISTERS
352		
353	172200	SIPDR0= 172200
354	172202	SIPDR1= 172202
355	172204	SIPDR2= 172204
356	172206	SIPDR3= 172206
357	172210	SIPDR4= 172210
358	172212	SIPDR5= 172212
359	172214	SIPDR6= 172214
360	172216	SIPDR7= 172216
361		
362		;*SUPERVISOR "D" PAGE DESCRIPTOR REGISTERS
363		
364	172220	SDPDR0= 172220
365	172222	SDPDR1= 172222
366	172224	SDPDR2= 172224
367	172226	SDPDR3= 172226
368	172230	SDPDR4= 172230
369	172232	SDPDR5= 172232
370	172234	SDPDR6= 172234
371	172236	SDPDR7= 172236
372		
373		;*SUPERVISOR "I" PAGE ADDRESS REGISTERS
374		
375	172240	SIPAR0= 172240
376	172242	SIPAR1= 172242
377	172244	SIPAR2= 172244
378	172246	SIPAR3= 172246
379	172250	SIPAR4= 172250
380	172252	SIPAR5= 172252
381	172254	SIPAR6= 172254
382	172256	SIPAR7= 172256
383		
384		;*SUPERVISOR "D" PAGE ADDRESS REGISTERS
385		
386	172260	SDPAR0= 172260
387	172262	SDPAR1= 172262
388	172264	SDPAR2= 172264
389	172266	SDPAR3= 172266
390	172270	SDPAR4= 172270
391	172272	SDPAR5= 172272
392	172274	SDPAR6= 172274
393	172276	SDPAR7= 172276
394		
395		;*KERNEL "I" PAGE DESCRIPTOR REGISTERS
396		
397	172300	KIPDR0= 172300
398	172302	KIPDR1= 172302
399	172304	KIPDR2= 172304
400	172306	KIPDR3= 172306

GLOBAL AREAS
KDJ11A.MACMACY11 30A(1052)
15-MAR-84 15:51

15-MAR-84 16:58 PAGE 11

MEMORY MANAGEMENT DEFINITIONS

SEQ 0011

401	172310	KIPDR4= 172310
402	172312	KIPDR5= 172312
403	172314	KIPDR6= 172314
404	172316	KIPDR7= 172316
405		
406		;*KERNEL "D" PAGE DESCRIPTOR REGISTERS
407		
408	172320	KDPDR0= 172320
409	172322	KDPDR1= 172322
410	172324	KDPDR2= 172324
411	172326	KDPDR3= 172326
412	172330	KDPDR4= 172330
413	172332	KDPDR5= 172332
414	172334	KDPDR6= 172334
415	172336	KDPDR7= 172336
416		
417		;*KERNEL "I" PAGE ADDRESS REGISTERS
418		
419	172340	KIPAR0= 172340
420	172342	KIPAR1= 172342
421	172344	KIPAR2= 172344
422	172346	KIPAR3= 172346
423	172350	KIPAR4= 172350
424	172352	KIPAR5= 172352
425	172354	KIPAR6= 172354
426	172356	KIPAR7= 172356
427		
428		;*KERNEL "D" PAGE ADDRESS REGISTERS
429		
430	172360	KDPAR0= 172360
431	172362	KDPAR1= 172362
432	172364	KDPAR2= 172364
433	172366	KDPAR3= 172366
434	172370	KDPAR4= 172370
435	172372	KDPAR5= 172372
436	172374	KDPAR6= 172374
437	172376	KDPAR7= 172376
438		
439		;THESE ARE FLOATING POINT ACCUMULATOR EQUATES
440	000000	AC0= #0
441	000001	AC1= #1
442	000002	AC2= #2
443	000003	AC3= #3
444	000004	AC4= #4
445	000005	AC5= #5
446	000006	AC6= #6
447	000007	AC7= #7
448		
449	000244	FPVEC= 244
450		
451		;THESE ARE CACHE REGISTER EQUATES
452	177746	CCR= 177746 ;CACHE CONTROL REGISTER
453	177744	MSER= 177744 ;MEMORY SYSTEM ERROR REGISTER
454	177752	HITMIS= 177752 ;HIT/MISS REGISTER
455	177766	CPEREG= 177766 ;CPU ERROR REGISTER
456		


```

457      ;MISCELLANEOUS DEFINITIONS
458      177546      BEVENT= 177546      ;BEVENT CONTROL REGISTER
459      177560      RCSR= 177560
460      177562      RBUF= 177562
461      177564      XCSR= 177564
462      177566      XBUF= 177566
463      000000      ERRTN= HALT
464      000001      $TSTNU=1
465      000001      ERRNUM= 1      ;INITIALIZE ERROR NUMBER COUNTER
466      002000      AUSWR= 2000      ;SWR FOR APT--NO BEVENT TESTING
467
468
469      ;THIS EQUATE DEFINES THE BOTTOM OF THE PROGRAM STACK POINTER
470      001000      STBOT= 1000
471      000000      .ASECT
472      .SBTTL TRAP CATCHER
473
474      000000      .=0
475      ;*ALL UNUSED LOCATIONS OF THE VECTOR AREA CONTAIN
476      ;*A ".+2, IOT" SEQUENCE TO CATCH AND PROCESS ILLEGAL
477      ;*TRAPS AND INTERRUPTS THAT MIGHT OCCUR.
478      ;*THE IOT TRAP WHICH IS TAKEN ON THE ILLEGAL TRAP/INT
479      ;*TRAPS TO THE $SCOPE ROUTINE WHICH (IF THE RETURN PC IS
480      ;*LESS THAN 1002) JUMPS TO THE $ERROR ROUTINE.
481      ;*THE $ERROR ROUTINE WILL REPORT THE ERROR AS FOLLOWS:
482      ;*      PC=YYYYYY UNEXPECTED TRAP TO XXX
483      ;*AND RETURN TO THE PROGRAM AT PC=YYYYYY+2
484      ;*WHERE XXX=LOCATION OF ILLEGAL TRAP
485      ;*      YYYYYY=PC AT TIME OF TRAP
486      ;*NOTE: IF THE PROCESSOR IS NOT AN 11/05 THE PROGRAM
487      ;*      CAN BE STARTED AT ADDRESS 0 AS WELL AS ADDRESS 200.
488
489      000000      000000      $40CAT: HALT      ;;HALT
490      000002      000737      BR      .-100      ;;BRANCH TO 177700 & TIME OUT (NOT ON
491      ;;11/05)
492      000004      002200      .WORD START      ;;VECTOR TO STARTING ADDRESS
493      000006      000340      .WORD 340      ;;WITH PRIORITY LEVEL 7
494      000174      000174      .=174
495      000174      000000      DISPREG: .WORD 0      ;;SOFTWARE DISPLAY REGISTER
496      000176      000000      SWREG: .WORD 0      ;;SOFTWARE SWITCH REGISTER
497      .SBTTL STARTING ADDRES(ES)
498      000200      000137      002200      JMP $START ;;GO TO START OF PROGRAM
499      .SBTTL ACT11 HOOKS
500
501      ;*****
502      ;HOOKS REQUIRED BY ACT11
503      000204      $SVPC=.      ;SAVE PC
504      000046      .=46
505      000046      036362      $ENDAD      ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .SEOP
506      000052      000052      .=52
507      000052      000000      .WORD 0      ;;2)SET LOC.52 TO ZERO
508      000204      .=$SVPC      ;; RESTORE PC
509      .SBTTL APT PARAMETER BLOCK
510
511      ;*****
512      ;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT

```

```

513                                     ;;*****
514                                     . $X=.      ;;SAVE CURRENT LOCATION
515                                     . =24      ;;SET POWER FAIL TO POINT TO START OF PROGRAM
516 000024 000200 200      ;;FOR APT START UP
517                                     . =44      ;;POINT TO APT INDIRECT ADDRESS PNTR.
518 000044 000204 $APTHDR ;;POINT TO APT HEADER BLOCK
519                                     . =. $X      ;;RESET LOCATION COUNTER
520                                     ;;*****
521                                     ;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
522                                     ;INTERFACE SPEC.
523
524 000204 $APTHD:
525 000204 000000 $HIBTS: .WORD 0      ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
526 000206 001000 $MBADR: .WORD $MAIL ;;ADDRESS OF APT MAILBOX (BITS 0-15)
527 000210 000001 $TSTM: .WORD 1      ;;RUN TIM OF LONGEST TEST
528 000212 000002 $PASTM: .WORD 2     ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
529 000214 000000 $UNITM: .WORD 0     ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
530 000216 000014 .WORD $ETEND-$MAIL/2 ;;LENGTH MAILBOX-ETABLE(WORDS)
531                                     . =. $X      ;;SAVE CURRENT LOCATION COUNT
532                                     . =2
533 000002 000000 0
534 000004 000006 6
535 000006 000004 4      ;SET UP SOME VECTORS
536 000204 . =. $X      ;RESTORE LOCATION COUNT
537 001000 . =1000

```



```
538 .SBTTL GLOBAL DATA SECTION
539
540
541 ;**
542 ; THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
543 ; IN MORE THAN ONE TEST.
544 ;--
545 .SBTTL APT MAILBOX-ETABLE
546
547 ;*****
548 .EVEN
549 $MAIL:
550 $MSGTY: .WORD   MSGTY   ; APT MAILBOX
551 $FATAL: .WORD   AFATAL  ; MESSAGE TYPE CODE
552 $TESTN: .WORD   ATESTN  ; FATAL ERROR NUMBER
553 $PASS: .WORD   APASS    ; TEST NUMBER
554 $DEVCT: .WORD   ADEVCT  ; PASS COUNT
555 $UNIT: .WORD   AUNIT    ; DEVICE COUNT
556 $MSGAD: .WORD   AMSGAD  ; I/O UNIT NUMBER
557 $MSGLG: .WORD   AMSGLG  ; MESSAGE ADDRESS
558 $ETABLE:
559 $ENV: .BYTE   AENV      ; MESSAGE LENGTH
560 $ENVH: .BYTE   AENVH    ; APT ENVIRONMENT TABLE
561 $SWREG: .WORD   ASWREG   ; ENVIRONMENT BYTE
562 $USWR: .WORD   AUSWR    ; ENVIRONMENT MODE BITS
563 $CPUOP: .WORD   ACPUOP  ; APT SWITCH REGISTER
564 ;
565 ;
566 ;
567 ;
568 ;
569 001030 $ETEND:
570 .MEXIT
571
572 ; THESE LOCATIONS ARE USED IN MORE THAN ONE TEST TO STORE VECTOR DATA
573 ; WHEN THE TEST NEEDS TO HAVE AN ERROR CONDITION RESPOND DIFFERENTLY
574 ; FROM THE DEFAULT RESPONSE.
575 001030 SLOC00: .WORD 0
576 001032 SLOC01: .WORD 0
577
578 ; THESE LOCATIONS ARE USED IN MORE THAN ONE TEST TO STORE WORKING DATA.
579 001034 EXPDAT: .WORD 0 ; STORES EXPECTED (GOOD) DATA FOR COMPARISONS
580 001036 RECDAT: .WORD 0 ; STORES RECIEVED DATA TO BE VERIFIED
581 001040 COUNT: .WORD 0 ; ERROR INDICATOR FOR FLOATING POINT TESTS
582 001042 FLAG: .WORD 0 ; USED TO STORE "FLAG" CONDITIONS
583 001044 ERRCNT: .WORD 0 ; STORAGE FOR ERROR COUNT
584 001046 SWR: .WORD DSWR ; STORAGE FOR SWITCH REGISTER ADDRESS
585 001050 DISPLAY: .WORD DDISP ; STORAGE FOR DISPLAY REGISTER ADDRESS
586 001052 $ERFLG: .WORD 0 ; ERROR FLAG
587
588 001054 ; THESE LOCATIONS ARE USED BY MORE THAN ONE TEST AS LOOP COUNTERS
589 001056 DCOUNT: .WORD 0
590 001060 ALLCTR: .WORD 0
591 001062 LOOPIN: .WORD 0
592 001064 SAVSP1: .WORD 0 ; STORAGE FOR UNEXPECTED TRAP DATA
593 SAVSP2: .WORD 0 ;
```

```

594
595
596 001066 000004      BTEXP: .BLKW 4      ;STORES EXPONENT DURING BIT TESTS
597 001076 000004      BTRES: .BLKW 4      ;STORES RECIEVED DATA FOR BIT TESTS
598 001106 000004      RECFEC: .BLKW 4      ;RECIEVED FLOATING POINT EXCEPTION CODE
599 001116 000004      RECST: .BLKW 4      ;RECIEVED FLOATING POINT STATUS
600 001126 000004      RECDST: .BLKW 4      ;DESTINATION ADDRESS FOR FLOATING POINT TESTS
601
602
603
604
605
606
607
608 001136
609 001136 000020      ;!!!!!!THIS IS IT. THE PROGRAM TEST LOCATION AND WRITE BUFFER!!!!!!!!!!!!!!!!!!!!!!
                        TSTLOC: .BLKW 20
  
```


610	001176	123456			TAB1:	.WORD	123456
611	001200	000000				.WORD	000000
612	001202	000000				.WORD	0
613	001204	000001				.WORD	1
614	001206	055555			TAB2:	.WORD	055555
615	001210	177777				.WORD	177777
616	001212	145671				.WORD	145671
617	001214	100000				.WORD	100000
618	001216	003000			TAB3:	.WORD	003000
619	001220	123456				.WORD	123456
620	001222	000000				.WORD	0
621	001224	000000				.WORD	0
622	001226	055555			TAB4:	.WORD	55555
623	001230	177777				.WORD	-1
624	001232	000000				.WORD	0
625	001234	000000				.WORD	0
626	001236	043243			TAB5:	.WORD	43243
627	001240	000000				.WORD	0
628	001242	000000				.WORD	0
629	001244	000000				.WORD	0
630	001246	162400			TAB5A:	.WORD	162400
631	001250	000000				.WORD	0
632	001252	000000				.WORD	0
633	001254	000000				.WORD	0
634	001256	000000			TAB6:	.WORD	0
635	001260	000000				.WORD	0
636	001262	000000				.WORD	0
637	001264	000000				.WORD	0
638	001266	047050			TAB6A:	.WORD	47050
639	001270	010000				.WORD	10000
640	001272	000000				.WORD	0
641	001274	000000				.WORD	0
642	001276	000200			TAB7:	.WORD	200
643	001300	000000				.WORD	0
644	001302	000000				.WORD	0
645	001304	000000				.WORD	0
646	001306	000200			TAB8:	.WORD	200
647	001310	000000				.WORD	0
648	001312	000000				.WORD	0
649	001314	000001				.WORD	1
650	001316	000400	000000	000000	TAB9:	.WORD	400.0.0.0
651	001324	000000					
652	001326	030000			TAB10:	.WORD	30000
653	001330	003000				.WORD	3000
654	001332	000000				.WORD	0
655	001334	000000				.WORD	0
656	001336	016400			TAB11:	.WORD	16400
657	001340	000000				.WORD	0
658	001342	000000				.WORD	0
659	001344	000000				.WORD	0
660	001346	030000	003000	000002	TAB11A:	.WORD	30000.3000.2.0
661	001354	000000					
662	001356	016100	000000	000000	TAB12:	.WORD	16100.0.0.1
663	001364	000001					
664	001366	016200			TAB13:	.WORD	16200
665	001370	000000				.WORD	0

666	001372	000000				.WORD	0
667	001374	000001				.WORD	1
668	001376	030000	003000	000000	TAB13B:	.WORD	30000,3000,0,140000
669	001404	140000					
670	001406	030000			TAB14:	.WORD	30000
671	001410	000000				.WORD	0
672	001412	000000				.WORD	0
673	001414	000000				.WORD	0
674	001416	024700			TAB15:	.WORD	24700
675	001420	000000				.WORD	0
676	001422	000000				.WORD	0
677	001424	000000				.WORD	0
678	001426	025000			TAB16:	.WORD	25000
679	001430	175363				.WORD	175363
680	001432	123456				.WORD	123456
681	001434	123456				.WORD	123456
682	001436	030000			TAB17:	.WORD	30000
683	001440	007020				.WORD	7020
684	001442	000000	000000			.WORD	0,0
685	001446	023456			TAB18:	.WORD	23456
686	001450	000000				.WORD	0
687	001452	000000				.WORD	0
688	001454	000001				.WORD	1
689	001456	100200	000000	000000	TAB21:	.WORD	100200,0,0,0
690	001464	000000					
691	001466	100400	000000	000000	TAB22:	.WORD	100400,0,0,0
692	001474	000000					
693	001476	000200	000000	000000	TAB23:	.WORD	200,0,0,1
694	001504	000001					
695	001506	062400	000000	000000	TAB24:	.WORD	62400,0,0,0
696	001514	000000					
697	001516	001100	000000	000000	TAB25:	.WORD	1100,0,0,0
698	001524	000000					
699	001526	100600	000000	000000	TAB26:	.WORD	100600,0,0,0
700	001534	000000					
701	001536	001000	000000	000000	TAB27:	.WORD	1000,0,0,0
702	001544	000000					
703	001546	000600	000000	000000	TAB28:	.WORD	600,0,0,0
704	001554	000000					
705	001556	010100	000000	000000	TAB29:	.WORD	10100,0,0,0
706	001564	000000					
707	001566	010100	000000	002000	TAB29A:	.WORD	10100,0,2000,0
708	001574	000000					
709							
710	001576	000500	000000	000000	TAB30:	.WORD	500,0,0,0
711	001604	000000					
712	001606	100400	000000	000000	TAB31:	.WORD	100400,0,0,0
713	001614	000000					
714	001616	016000	000000	000000	TAB32:	.WORD	16000,0,0,0
715	001624	000000					
716	001626	011600	000000	000000	TAB33:	.WORD	11600,0,0,0
717	001634	000000					
718	001636	000640	000000	000000	TAB34:	.WORD	640,0,0,0
719	001644	000000					
720	001646	077600	000000	000000	TAB40:	.WORD	77600,0,0,0
721	001654	000000					

GLOBAL AREAS MACY11 30A(1052) 15-MAR-84 16:58 PAGE 18
KDJ11A.MAC 15-MAR-84 15:51 APT MAILBOX-ETABLE

SEQ 0018

722	001656	100200	000000	000000	TAB41: .WORD	100200,0,0,1
723	001664	000001				
724	001666	000340	000000	000000	TAB42: .WORD	340,0,0,0
725	001674	000000				
726	001676	000077	177777	177777	TAB43: .WORD	77,177777,177777,177776
727	001704	177776				
728	001706	000577	177777	177777	TAB45: .WORD	577,-1,-1,-1
729	001714	177777				
730	001716	000577	177777	000000	TAB46: .WORD	577,-1,0,0
731	001724	000000				
732	001726	173737	124242	052525	TAB47: .WORD	173737,124242,052525,12346
733	001734	012346				
734	001736	000000	000000	052525	TAB47A: .WORD	0,0,052525,12346
735	001744	012346				
736	001746	173737	124242	000000	TAB48: .WORD	173737,124242,0,0
737	001754	000000				
738	001756	000600	000000	000000	TAB49: .WORD	600,0,0,0
739	001764	000000				

GLOBAL AREAS
KDJ11A.MACMACY11 30A(1052)
15-MAR-84 15:5115-MAR-84 16:58 PAGE 19
GLOBAL TEXT SECTION

SEQ 0019

```

740      .SBTTL  GLOBAL TEXT SECTION
741
742      ;++
743      ; THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
744      ; MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
745      ; MORE THAN ONE TEST.
746      ;--
747
748      ;
749      ; FORMAT STATEMENTS USED IN PRINT CALLS
750      ;
751
752      001766 005015 040503 044103 ERRMSG: .ASCIZ <CR><LF>/CACHE SYSTEM ERROR/
753      001774 020105 054523 052123
754      002002 046505 042440 051122
755      002010 051117      000
756      002013      015 042412 051122 FPPERR: .ASCIZ <CR><LF>/ERROR DURING FPP TESTING/
757      002020 051117 042040 051125
758      002026 047111 020107 050106
759      002034 020120 042524 052123
760      002042 047111 000107
761      002046 005015 051105 047522 ERR1:  .ASCIZ <CR><LF>/ERROR # =/
762      002054 020122 020043 000075
763      002062 005015 051105 047522 ERR2:  .ASCIZ <CR><LF>/ERROR PC =/
764      002070 020122 041520 036440
765      002076      000
766      002077      015 020012 020040 $CRLF: .ASCIZ <CR><LF>/ /
767      002104      000
768      002106
      .EVEN

```


GLOBAL AREAS MACY11 30A(1052) 15-MAR-84 16:58 PAGE 20
KDJ11A.MAC 15-MAR-84 15:51 GLOBAL ERROR REPORT SECTION

SEQ 0020

769
770
771
772
773
774
775

.SBTTL GLOBAL ERROR REPORT SECTION

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

```

776      .SBTTL  GLOBAL SUBROUTINES SECTION
777
778      ;**
779      ; THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES
780      ; THAT ARE USED IN MORE THAN ONE TEST.
781      ;--
782      ;FPP COMMON SUBROUTINES
783      002106 012600 WLDTRP: MOV      (SP)+,R0      ;SAVE PC
784      002110 012605      MOV      (SP)+,R5      ;SAVE STATUS AND RESTORE STACK
785      002112 104000      ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
786      002114 000001      .WORD      1      ;UNIQUE ERROR NUMBER
787      002116 002013      .WORD      FPPERR    ;ADDRESS OF ERROR MESSAGE
788      002120 000110      JMP        (R0)      ;GO BACK INLINE
789
790      ;
791      ;
792      002122 000000 TRPFLG: .WORD      0
793      002124 000207 ERRFP:  RTS        R7
794      002126 000207 ERR:   RTS        R7
795
796      ;
797      ;
798      ;
799      ;
800      ;SUBROUTINE DATA VERIFICATION -
801      ;
802      ; CALLED BY      JSR      R7,DATVER
803      ;
804      ;INPUT:          (R4)=EXPECTED DATA
805      ;                (R1)=RECEIVED DATA
806      ;
807      ;THIS ROUTINE VERIFIES THAT THE 4 CONSECUTIVE WORDS STARTING WITH (R4) ARE
808      ;EQUAL TO THE FOUR WORDS ADDRESSED BY (R1). THE CONTENTS OF R4, AND R1 ARE NOT
809      ;DISTURBED.
810      ;LOCATION "COUNT" , IF NOT EQUAL TO 0 SIGNIFIES DATA ERROR
811      ;IF THE STATUS IS FLOATING MODE, THE LAST TWO BYTES OF RECEIVED
812      ;ARE SIMPLY CHECKED FOR ZEROS
813      ;
814      ;
815      002130 010446 DATVFR: MOV      R4,-(SP)      ;SAVE R4
816      002132 010146      MOV      R1,-(SP)      ;SAVE R1
817      002134 012767 000003 176676      MOV      #3,COUNT    ;SET UP ITERATION COUNT
818      002142 000167 000012      JMP        DAT1
819      ;
820      002146 010446 DATVER: MOV      R4,-(SP)      ;SAVE R4
821      002150 010146      MOV      R1,-(SP)      ;SAVE R1
822      002152 012767 000005 176660      MOV      #5,COUNT    ;SET UP ITERATION COUNT
823      002160 005367 176654 DAT1:  DEC      COUNT
824      002164 001402      BEQ        2$
825      002166 022421      CMP        (R4)+,(R1)+
826      002170 001773      BEQ        DAT1
827      002172 012601 2$:  MOV      (SP)+,R1
828      002174 012604      MOV      (SP)+,R4
829      002176 000207      RTS        R7
830
      ;RESTORE R1
      ;RESTORE R4
      ;GO BACK TO CALLING ROUTINE
      ;IF DATA ERROR, COUNT NE 0

```



```
831 002200 START:
832 002200 012737 000014 177746 MOV #14,#CCR ;SET CACHE TO FORCE MISS
833 .SBTTL INITIALIZE THE COMMON TAGS
834 002206 012706 001000 MOV #STACK,SP ;SETUP THE STACK POINTER
835 ;;INITIALIZE A FEW VECTORS
836 002212 012737 040046 000030 MOV #ERROR,#EMTVEC ;EMT VECTOR FOR ERROR ROUTINE
837 002220 012737 000340 000032 MOV #340,#EMTVEC+2 ;LEVEL 7
838 002226 012737 037530 000034 MOV #TRAP,#TRAPVEC ;TRAP VECTOR FOR TRAP CALLS
839 002234 012737 000340 000036 MOV #340,#TRAPVEC+2;LEVEL 7
840 002242 005067 176540 CLR #PASS ;CLEAR THE PASS COUNT
841 002246 016767 034056 034046 MOV #ENDCT,#EOPCT ;SETUP END-OF-PROGRAM COUNTER
842 002254 105067 176572 CLRB #ERFLG ;CLEAR THE ERROR FLAG
843 ;;SIZE FOR A HARDWARE SWITCH REGISTER, IF NOT FOUND OR IT IS
844 ;;EQUAL TO A "-1", SETUP FOR A SOFTWARE SWITCH REGISTER.
845 002260 013746 000004 MOV #ERRVEC,-(SP) ;SAVE ERROR VECTOR
846 002264 012737 002320 000004 MOV #64,#ERRVEC ;SET UP ERROR VECTOR
847 002272 012767 177570 176546 MOV #DSWR,SWR ;SETUP FOR A HARDWARE SWICH REGISTER
848 002300 012767 177570 176542 MOV #DDISP,DISPLAY ;AND A HARDWARE DISPLAY REGISTER
849 002306 022777 177777 176532 CMP #-1,DSWR ;TRY TO REFERENCE HARDWARE SWR
850 002314 001012 BNE 66$ ;BRANCH IF NO TIMEOUT TRAP OCCURRED
851 ;AND THE HARDWARE SWR IS NOT = -1
852 002316 000403 BR 65$ ;BRANCH IF NO TIMEOUT
853 002320 012716 002326 64$: MOV #65$,(SP) ;SET UP FOR TRAP RETURN
854 002324 000002 RTI
855 002326 012767 000176 176512 65$: MOV #SWREG,SWR ;POINT TO SOFTWARE SWR
856 002334 012767 000174 176506 MOV #DISPREG,DISPLAY
857 002342 012637 000004 66$: MOV (SP)+,#ERRVEC ;RESTORE ERROR VECTOR
858
859 .MACRO $$SETMAIL ?$ARG1
860 CLR #PASS ;CLEAR PASS COUNT
861 BITB #APTSIZE,$ENVM ;TEST USER SIZE UNDER APT
862 BEQ $ARG1 ;YES,USE NON-APT SWITCH
863 MOV #SWREG,SWR ;NO,USE APT SWITCH REGISTER
864
865 $ARG1:
866 .ENDM $$SETMAIL
867 002346 005067 176434 CLR #PASS ;CLEAR PASS COUNT
868 002352 132767 000200 176441 BITB #APTSIZE,$ENVM ;TEST USER SIZE UNDER APT
869 002360 001403 BEQ 67$ ;YES,USE NON-APT SWITCH
870 002370 012767 001022 176456 MOV #SWREG,SWR ;NO,USE APT SWITCH REGISTER
871 002370 012737 040046 000020 67$: MOV #ERROR,#IOTVEC ;SET UP IOT VECTORS
872 002376 012737 000340 000022 MOV #340,#IOTVEC+2 ;TO GO TO ERROR ROUTINE
873 002404 005037 177766 CLR #177766 ;CLEAR CPU ERROR REGISTER
874 002410 005067 176370 RESTART: CLR #TESTN ;RESET $TESTN TO ZERO
875 002414 012737 000014 177746 MOV #14,#CCR ;SET CACHE TO FORCE MISS
876
877 ;*****
878 .SBTTL FLOATING POINT TESTS
879 ;*****
880 BEGIN FLOATING POINT TESTING
881 ;*****
882 ;*****
883 002422 MBT1:
884 ;*****
885 ;*TEST 1 FPP REGISTER BIT TESTS
886 ;*****
```

```
887  
888  
889  
890  
891  
892  
893 002422  
894 002422 005267 176356  
895 002426 170011  
896 002430 005005  
897 002432 012702 001066  
898 002436 012703 001076  
899 002442 170400  
900 002444 174012  
901 002446 005004  
902 002450 170400  
903 002452 170401  
904 002454 170402  
905 002456 170403  
906 002460 170404  
907 002462 170405  
908  
909 002464 010501  
910 002466 070127 000014  
911 002472 062701 002500  
912 002476 000111  
913 002500 172467 176362  
914 002504 174067 176366  
915 002510 000167 000074  
916 002514 172567 176346  
917 002520 174167 176352  
918 002524 000167 000060  
919 002530 172667 176332  
920 002534 174267 176336  
921 002540 000167 000044  
922 002544 172767 176316  
923 002550 174367 176322  
924 002554 000167 000030  
925 002560 172467 176302  
926 002564 174004  
927 002566 172404  
928 002570 000167 177710  
929 002574 172467 176266  
930 002600 174005  
931 002602 172405  
932 002604 000167 177674  
933 002610 026767 176252 176260  
934 002616 001014  
935 002620 026767 176244 176252  
936 002626 001010  
937 002630 026767 176236 176244  
938 002636 001004  
939 002640 026767 176230 176236  
940 002646 001403  
941 002650  
942 002650 104000
```

```
;R5=FPP POINTER  
;R1=TEMPORARY COUNTER  
;R2=POINTER TO EXPECTED DATA  
;R3=POINTER TO RECEIVED DATA  
;R4=ODD/EVEN COUNTER  
;*****  
TST1: INC $TESTN ;INCREMENT TEST NUMBER  
SETD  
MBT2: CLR R5 ;SETUP FPP ACC POINTER  
MOV #BTEXP,R2 ;POINT TO TEST DATA  
MOV #BTRES,R3 ;POINT TO RECEIVED DATA  
MBT2A: CLRD ACO ;SETUP FPP REGISTER VALUES  
STD ACO,(R2) ;CLEAR EXPECTED VALUE  
CLR R4  
BTGO: CLRD ACO ;SETUP FPP REGISTER VALUES  
CLRD AC1  
CLRD AC2  
CLRD AC3  
CLRD AC4  
CLRD AC5  
  
MOV R5,R1 ;GET FPP AC NUMBER INTO R1  
MUL #14,R1 ;ALLOW 10 LOCATIONS FOR OPERATION  
ADD #MACO,R1 ;SETUP JMP LOCATION  
JMP (R1)  
MACO: LDD BTEXP,ACO ;LOAD TEST DATA INTO TEST REGISTER  
MACOA: STD ACO,BTRES ;SAVE TEST RESULT  
JMP MACE ;GET OUT  
MAC1: LDD BTEXP,AC1 ;LOAD TEST DATA INTO TEST REGISTER  
STD AC1,BTRES ;SAVE TEST RESULT  
JMP MACE ;GET OUT  
MAC2: LDD BTEXP,AC2 ;LOAD TEST DATA INTO TEST REGISTER  
STD AC2,BTRES ;SAVE TEST RESULT  
JMP MACE ;GET OUT  
MAC3: LDD BTEXP,AC3 ;LOAD TEST DATA INTO TEST REGISTER  
STD AC3,BTRES ;SAVE TEST RESULT  
JMP MACE ;GET OUT  
MAC4: LDD BTEXP,ACO ;LOAD TEST DATA INTO TEST REGISTER  
STD ACO,AC4 ;SAVE TEST RESULT  
LDD AC4,ACO ;GET OUT  
JMP MACOA  
MAC5: LDD BTEXP,ACO ;LOAD TEST DATA INTO TEST XFER REGISTER  
STD ACO,AC5 ;LOAD TEST REGISTER  
LDD AC5,ACO ;STORE RESULT INTO XFER FPP REGISTER  
JMP MACOA ;GET OUT  
MACE: CMP BTEXP,BTRES ;BRANCH IF REGISTER ERROR  
BNE BTER  
CMP BTEXP+2,BTRES+2  
BNE BTER  
CMP BTEXP+4,BTRES+4  
BNE BTER  
CMP BTEXP+6,BTRES+6  
BEQ MBT8  
  
BTER: ERROR ;GOOD RESULT  
;ALL ERRORS TO TRAP TO EMT VECTOR
```


943	002652	000002		.WORD	2		;UNIQUE ERROR NUMBER
944	002654	002013		.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
945							;FPP AC LOADED INCORRECTLY
946							;NOW VERIFY THE OTHER REGISTERS REMAINED ZERO
947	002656			MBT8:			
948	002656	005001		CLR	R1		;CLEAR TEMPORARY COUNTER
949	002660	005705		TST	R5		;SEE IF R0 UNDER TEST
950	002662	001413		BEQ	MBT8A		;BRANCH IF TEST ING R0
951	002664	020527	000004	CMP	R5,#4		;SEE IF TESTING FPP REGISTER >R4
952	002670	100010		BPL	MBT8A		;SKIP R0 TESTING
953	002672	174067	176200	STD	AC0,BTRES		;SAVE AC TEST RESULT
954	002676	004767	000246	JSR	R7,BTTST		;VERIFY THAT CONTENTS REMAINED ZERO
955	002702	001403		BEQ	MBT8A		;BRANCH IF EXPECTED RESULT
956	002704	104000		ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
957	002706	000003		.WORD	3		;UNIQUE ERROR NUMBER
958	002710	002013		.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
959							;BAD AC0
960	002712	020527	000001	MBT8A:	CMP	R5,#1	;SEE IF R1 UNDER TEST
961	002716	001410			BEQ	MBT8B	;BRANCH IF R1 UNDER TEST
962	002720	174167	176152		STD	AC1,BTRES	;SAVE AC TEST RESULT
963	002724	004767	000220		JSR	R7,BTTST	;VERIFY THAT CONTENTS REMAINED ZERO
964	002730	001403			BEQ	MBT8B	;BRANCH IF GOOD
965	002732	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
966	002734	000004			.WORD	4	;UNIQUE ERROR NUMBER
967	002736	002013			.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
968							;BAD AC1
969	002740	020527	000002	MBT8B:	CMP	R5,#2	;SEE IF TESTING FPP REGISTER AC2
970	002744	001410			BEQ	MBT8C	;BRANCH IF R2 UNDER TEST
971	002746	174267	176124		STD	AC2,BTRES	;SAVE AC TEST RESULT
972	002752	004767	000172		JSR	R7,BTTST	;VERIFY THAT CONTENTS REMAINED ZERO
973	002756	001403			BEQ	MBT8C	;BRANCH IF GOOD
974	002760	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
975	002762	000005			.WORD	5	;UNIQUE ERROR NUMBER
976	002764	002013			.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
977							;BAD AC2
978	002766	020527	000003	MBT8C:	CMP	R5,#3	;SEE IF R3 UNDER TEST
979	002772	001410			BEQ	MBT8D	;BRANCH IF R3 UNDER TEST
980	002774	174367	176076		STD	AC3,BTRES	;SAVE AC TEST RESULT
981	003000	004767	000144		JSR	R7,BTTST	;VERIFY THAT CONTENTS REMAINED ZERO
982	003004	001403			BEQ	MBT8D	;BRANCH IF GOOD
983	003006	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
984	003010	000006			.WORD	6	;UNIQUE ERROR NUMBER
985	003012	002013			.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
986							;BAD AC3
987	003014	020527	000004	MBT8D:	CMP	R5,#4	;SEE IF R4 UNDER TEST
988	003020	001411			BEQ	MBT8E	;BRANCH IF R4 UNDER TEST
989	003022	172404			LDD	AC4,AC0	;MOVE REGISTER CONTENT
990	003024	174067	176046		STD	AC0,BTRES	;SAVE AC TEST RESULT
991	003030	004767	000114		JSR	R7,BTTST	;VERIFY THAT CONTENTS REMAINED ZERO
992	003034	001403			BEQ	MBT8E	;BRANCH IF GOOD
993	003036	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
994	003040	000007			.WORD	7	;UNIQUE ERROR NUMBER
995	003042	002013			.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
996							;BAD AC4
997	003044	020527	000005	MBT8E:	CMP	R5,#5	;SEE IF R0 UNDER TEST
998	003050	001411			BEQ	MBT8F	;BRANCH IF R0 UNDER TEST

999	003052	172405		LDD	AC5,ACO	;MOVE REGISTER CONTENTS
1000	003054	174067	176016	STD	ACO,BTRES	;SAVE AC TEST RESULT
1001	003060	004767	000064	JSR	R7,BTTST	;VERIFY THAT CONTENTS REMAINED ZERO
1002	003064	001403		BEG	MBT8F	;BRANCH IF GOOD
1003	003066	104000		ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1004	003070	000010		.WORD	10	;UNIQUE ERROR NUMBER
1005	003072	002013		.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
1006						
1007	003074	005204		MBT8F: INC	R4	;BAD AC5
1008	003076	000241		CLC		;INCREMENT PATTERN COUNTER
1009	003100	042704	177776	BIC	#177776,R4	
1010	003104	001401		BEG	MBT8FG	; TEST FOR ODD /EVEN
1011	003106	000261		SEC		;BRANCH IF EVEN
1012	003110	006112		MBT8FG: ROL	(R2)	;SET CARRY FOR TEST PATTERN SHIFT
1013	003112	006162	000002	ROL	2(R2)	;ROTATE LSW OF TEST PATTERN
1014	003116	006162	000004	ROL	4(R2)	;ROTATE 2 WORD OF TEST PATTERN
1015	003122	006162	000006	ROL	6(R2)	;ROTATE 3 WORD OF TEST PATTERN
1016	003126	103402		BCS	MBT8I	;ROTATE 4 WORD OF TEST PATTERN
1017	003130	000167	177314	JMP	BTGO	;JUMP IF THROUGH WITH TEST PATTERN
1018						;CONTINUE WITH NEW TEST PATTERN
1019	003134	005205		MBT8I: INC	R5	
1020	003136	020527	000006	CMP	R5,#6	;GO TO NEXT REGISTER TEST
1021	003142	100016		BPL	MBTE	;SEE IF THROUGH TESTING
1022	003144	000167	177272	JMP	MBT2A	;JUMP IF THROUGH
1023						;CONTINUE TESTING WITH NEW PATTERN
1024						
1025	003150	005767	175722	BTTST: TST	BTRES	;VERIFY CONTENTS AS ZERO
1026	003154	001010		BNE	BTTSTE	;EXIT IF NOT ZERO
1027	003156	005767	175716	TST	BTRES*2	;VERIFY CONTENTS AS ZERO
1028	003162	001005		BNE	BTTSTE	;EXIT IF NOT ZERO
1029	003164	005767	175712	TST	BTRES*4	;VERIFY CONTENTS AS ZERO
1030	003170	001002		BNE	BTTSTE	;EXIT IF NOT ZERO
1031	003172	005767	175706	TST	BTRES*6	;VERIFY CONTENTS AS ZERO
1032	003176	000207		BTTSTE: RTS	R7	;GO BACK TO CALLING ROUTINE
1033						
1034						
1035						
1036						
1037	003200			MBTE:		
1038						
1039						
1040	003200			MFACU:		
1041						
1042						
1043						
1044						
1045						
1046						
1047						
1048	003200			TST2:		
1049	003200	005267	175600	INC	#TESTN	;INCREMENT TEST NUMBER
1050						
1051						
1052						
1053	003204	170011		SETD		
1054	003206	005000		MFA: CLR	R0	;SETUP FPP ACC POINTER

1055	003210	005004		CLR	R4	
1056	003212	012702	001066	MOV	#BTEXP,R2	;POINT TO TEST DATA
1057	003216	012703	001076	MOV	#BTRES,R3	;POINT TO RECEIVED DATA
1058	003222	012767	000051	MOV	#51,BTEXP	;SETUP EXPECTED DATA
1059	003230	012767	000052	MOV	#52,BTEXP+2	
1060	003236	012767	000053	MOV	#53,BTEXP+4	
1061	003244	012767	000054	MOV	#54,BTEXP+6	
1062	003252	172467	175610	LDD	BTEXP,ACO	;MOVE DATA TEMPORARILY
1063	003256	174005		STD	ACO,AC5	;PUT DATA INTO TEST REGISTER
1064	003260	004567	000240	JSR	R5,SUBT	;SUBTRACT TEN FROM EACH EXPECTED DATA
1065	003264	172467	175576	LDD	BTEXP,ACO	;MOVE DATA TEMPORARILY
1066	003270	174004		STD	ACO,AC4	;MOVE DATA INTO TEST REGISTER
1067	003272	004567	000226	JSR	R5,SUBT	;SUBTRACT 10 FROM TEST DATA WORDS
1068	003276	172767	175564	LDD	BTEXP,AC3	;STORE INTO TEST REGISTER
1069	003302	004567	000216	JSR	R5,SUBT	;GET NEXT SET OF UNIQUE DATA WORDS
1070	003306	172667	175554	LDD	BTEXP,AC2	;STORE INTO TEST REGISTER
1071	003312	004567	000206	JSR	R5,SUBT	;GET NEXT SET OF TEST DATAS
1072	003316	172567	175544	LDD	BTEXP,AC1	;LOAD TEST REGISTER
1073	003322	004567	000176	JSR	R5,SUBT	;GET NEXT SET OF TEST WORDS
1074	003326	172467	175534	LDD	BTEXP,ACO	;LOAD FINAL TEST REGISTER
1075						;ALL REGISTER CONTAIN UNIQUE TEST WORDS
1076	003332	174067	175540	STD	ACO,BTRES	;STORE ACO,RESULT
1077	003336	004567	000246	JSR	R5,BFA	;CHECK RESULT
1078	003342	001403		BEQ	BFAC1	;BRANCH IF GOOD
1079	003344	104000		ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1080	003346	000011		.WORD	11	;UNIQUE ERROR NUMBER
1081	003350	002013		.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
1082						;BAD ACO
1083	003352	004567	000200	BFAC1: JSR	R5,ADDT	;UPDATE EXPECTED RESULT
1084	003356	174167	175514	STD	AC1,BTRES	;STORE AC1 RESULT
1085	003362	004567	000222	JSR	R5,BFA	;CHECK RESULT
1086	003366	001403		BEQ	BFAC2	;BRANCH IF GOOD
1087	003370	104000		ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1088	003372	000012		.WORD	12	;UNIQUE ERROR NUMBER
1089	003374	002013		.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
1090						;BAD RESULT AC1
1091	003376	004567	000154	BFAC2: JSR	R5,ADDT	;UPDATE EXPECTED RESULT
1092	003402	174267	175470	STD	AC2,BTRES	;STORE AC2 RESULT
1093	003406	004567	000176	JSR	R5,BFA	;CHECK RESULT
1094	003412	001403		BEQ	BFAC3	;BRANCH IF GOOD
1095	003414	104000		ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1096	003416	000013		.WORD	13	;UNIQUE ERROR NUMBER
1097	003420	002013		.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
1098						;BAD AC2 RESULT
1099	003422	004567	000130	BFAC3: JSR	R5,ADDT	;UPDATE EXPECTED RESULT
1100	003426	174367	175444	STD	AC3,BTRES	;SAVE TEST RESULT
1101	003432	004567	000152	JSR	R5,BFA	;CHECK RESULT
1102	003436	001403		BEQ	BFAC4	;BRANCH IF GOOD
1103	003440	104000		ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1104	003442	000014		.WORD	14	;UNIQUE ERROR NUMBER
1105	003444	002013		.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
1106						;BAD AC3 RESULT
1107	003446	004567	000104	BFAC4: JSR	R5,ADDT	;UPDATE EXPECTED RESULT
1108	003452	172704		LDD	AC4,AC3	;SAVE TEMPORARY
1109	003454	174367	175416	STD	AC3,BTRES	;STORE AC4 RESULT
1110	003460	004567	000124	JSR	R5,BFA	;CHECK RESULT

1111	003464	001403			BEQ	BFAC5			
1112	003466	104000			ERROR				;BRANCH IF GOOD
1113	003470	000015			.WORD	15			;ALL ERRORS TO TRAP TO EMT VECTOR
1114	003472	002013			.WORD	FPPERR			;UNIQUE ERROR NUMBER
1115									;ADDRESS OF ERROR MESSAGE
1116	003474	004567	000056		BFAC5:	JSR	R5,ADDT		;BAD AC4 RESULT
1117	003500	172605			LDD	AC5,AC2			;UPDATE EXPECTED RESULT
1118	003502	174267	175370		STD	AC2,BTRES			;SAVE TEMPORARY COPY
1119	003506	004567	000076		JSR	R5,BFA			;MOVE AC5 RESULT
1120	003512	001456			BEQ	BFAE			;CHECK RESULT
1121	003514	104000			ERROR				;BRANCH IF GOOD
1122	003516	000016			.WORD	16			;ALL ERRORS TO TRAP TO EMT VECTOR
1123	003520	002013			.WORD	FPPERR			;UNIQUE ERROR NUMBER
1124									;ADDRESS OF ERROR MESSAGE
1125	003522	000452			BR	BFAE			;BAD AC5 RESULT
1126									;EXIT MODULE
1127	003524	162767	000010	175334	SUBT:	SUB	#10,BTEXP		;UPDATE EXPECTED CONTENTS
1128	003532	162767	000010	175330		SUB	#10,BTEXP.2		;UPDATE EXPECTED CONTENTS
1129	003540	162767	000010	175324		SUB	#10,BTEXP.4		;UPDATE EXPECTED CONTENTS
1130	003546	162767	000010	175320		SUB	#10,BTEXP.6		;UPDATE EXPECTED CONTENTS
1131	003554	000205				RTS	R5		
1132	003556	062767	000010	175302	ADDT:	ADD	#10,BTEXP		;UPDATE EXPECTED CONTENTS
1133	003564	062767	000010	175276		ADD	#10,BTEXP.2		;UPDATE EXPECTED CONTENTS
1134	003572	062767	000010	175272		ADD	#10,BTEXP.4		;UPDATE EXPECTED CONTENTS
1135	003600	062767	000010	175266		ADD	#10,BTEXP.6		;UPDATE EXPECTED CONTENTS
1136	003606	000205				RTS	R5		
1137									
1138	003610	026767	175252	175260	BFA:	CMP	BTEXP,BTRES		;VERIFY CONTENTS
1139	003616	001013				BNE	BFB		;EXIT IF NOT ZERO
1140	003620	026767	175244	175252		CMP	BTEXP.2,BTRES.2		;VERIFY CONTENTS
1141	003626	001007				BNE	BFB		;EXIT IF NOT ZERO
1142	003630	026767	175236	175244		CMP	BTEXP.4,BTRES.4		;VERIFY CONTENTS
1143	003636	001003				BNE	BFB		;EXIT IF NOT ZERO
1144	003640	026767	175230	175236		CMP	BTEXP.6,BTRES.6		;VERIFY CONTENTS
1145	003646	000205			BFB:	RTS	R5		;GO BACK TO CALLING ROUTINE
1146									
1147									
1148	003650				BFAE:				
1149									
1150									
1151									
1152	003650				TSFP1:				
1153					*****				
1154					;TEST 3		TEST LDFPS AND STFPS MODE 0		
1155					*****				
1156	003650				TST3:				
1157	003650	005267	175130		INC	#TESTN			;INCREMENT TEST NUMBER
1158	003654	005037	002122		CLR	#TRPFLG			;CLEAR TRAP FLAG
1159	003660	012704	147757		MOV	#147757,R4			;SETUP DATA TO BE LOADED
1160	003664	004767	000032		JSR	PC,LOST			;LOAD AND STORE FPS WITH DATA
1161	003670	012704	105252		MOV	#105252,R4			;SETUP DATA TO BE LOADED
1162	003674	004767	000022		JSR	PC,LOST			;LOAD AND STORE FPS WITH DATA
1163	003700	012704	042505		MOV	#42505,R4			;SETUP DATA TO BE LOADED
1164	003704	004767	000012		JSR	PC,LOST			;LOAD AND STORE FPS WITH DATA
1165	003710	005004			CLR	R4			;SETUP DATA TO BE LOADED
1166	003712	004767	000004		JSR	PC,LOST			;LOAD AND STORE FPS WITH DATA


```

1167
1168
1169 003716 000167 000020      ;      JMP      FIN1
1170
1171 003722 170104      LOST:  LDFPS  R4      ;LOAD FPS WITH DATA
1172 003724 170201      STFPS  R1      ;LOAD R1 WITH (FPS)
1173 003726 020401      CMP     R4,R1    ;DID THE INSTRUCTIONS WORK
1174 003730 001403      BEQ     1$      ;YES GO ON
1175 003732 104000      ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
1176 003734 000017      .WORD    17      ;UNIQUE ERROR NUMBER
1177 003736 002013      .WORD    FPPERR ;ADDRESS OF ERROR MESSAGE
1178
1179 003740 000207      1$:  RTS     PC      ;NO GO TO ERROR
1180 003742      FIN1:      ;RETURN
1181
1182 003742      TSFP2:
1183      ;*****
1184      ;*TEST 4      TEST CFCC
1185      ;*****
1186 003742      TST4:
1187 003742 005267 175036      INC     $TESTN      ;INCREMENT TEST NUMBER
1188 003746 005037 002122      CLR     @TRPFLG    ;CLEAR TRAP FLAG
1189 003752 012704 000017      MOV     @17,R4      ;SETUP DATA TO BE LOADED
1190 003756 004767 000032      JSR     PC,TSF2    ;LOAD FPS AND COPY CONDITION CODES TO PS
1191 003762 012704 000012      MOV     @12,R4      ;SETUP DATA TO BE LOADED
1192 003766 004767 000022      JSR     PC,TSF2    ;LOAD FPS AND COPY CONDITION CODES TO PS
1193 003772 012704 000005      MOV     @5,R4      ;SETUP DATA TO BE LOADED
1194 003776 004767 000012      JSR     PC,TSF2    ;LOAD FPS AND COPY CONDITION CODES TO PS
1195 004002 005004      CLR     R4      ;SETUP DATA TO BE LOADED
1196 004004 004767 000004      JSR     PC,TSF2    ;LOAD FPS AND COPY CONDITION CODES TO PS
1197
1198
1199 004010 000167 000030      ;      JMP      FIN2
1200
1201 004014 170104      TSF2:  LDFPS  R4      ;LOAD FPS
1202 004016 170000      CFCC      ;COPY CONDITION CODES TO PS
1203 004020 013701 177776      MOV     @177776,R1    ;SAVE PS TO R1
1204 004024 042701 177760      BIC     @177760,R1    ;MASK OUT UNWANTED BITS
1205 004030 020401      CMP     R4,R1    ;WAS CONDITION CODE BITS TRANSFERRED
1206
1207      BEQ     1$      ;CORRECTLY
1208 004034 104000      ERROR      ;YES GO ON
1209 004036 000020      .WORD    20      ;ALL ERRORS TO TRAP TO EMT VECTOR
1210 004040 002013      .WORD    FPPERR ;UNIQUE ERROR NUMBER
1211
1212 004042 000207      1$:  RTS     PC      ;ADDRESS OF ERROR MESSAGE
1213 004044      FIN2:      ;NO GO TO ERROR
1214
1215 004044      TSFP3:
1216      ;*****
1217      ;*TEST 5      TEST SETF, SETD, SETI, SETL
1218      ;*****
1219 004044      TST5:
1220 004044 005267 174734      INC     $TESTN      ;INCREMENT TEST NUMBER
1221 004050 005037 002122      CLR     @TRPFLG    ;CLEAR TRAP FLAG
1222 004054 012704 000200      MOV     @200,R4    ;SETUP DATA TO BE LOADED

```

1223	004060	170104			LDFPS	R4		;LOAD FPS
1224	004062	170001			SETF			;MAKE FD=0
1225	004064	170201			STFPS	R1		;STORE FPS
1226	004066	020127	000000		CMP	R1, #0		;IS FD=0
1227	004072	001403			BEQ	18		;YES GO ON
1228	004074	104000			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
1229	004076	000021			.WORD	21		;UNIQUE ERROR NUMBER
1230	004100	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
1231								;NO GO TO ERROR
1232	004102	170011		18:	SETD			;MAKE FD=1
1233	004104	170201			STFPS	R1		;STORE FPS
1234	004106	020104			CMP	R1, R4		;IS FD=1
1235	004110	001403			BEQ	21		;YES GO ON
1236	004112	104000			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
1237	004114	000022			.WORD	22		;UNIQUE ERROR NUMBER
1238	004116	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
1239								;NO GO TO ERROR
1240	004120	012704	000100	21:	MOV	#100, R4		;SETUP DATA TO BE LOADED
1241	004124	170104			LDFPS	R4		;LOAD FPS
1242	004126	170002			SETI			;MAKE FL=0
1243	004130	170201			STFPS	R1		;STORE FPS
1244	004132	020127	000000		CMP	R1, #0		;IS FL=0
1245	004136	001403			BEQ	31		;YES GO ON
1246	004140	104000			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
1247	004142	000023			.WORD	23		;UNIQUE ERROR NUMBER
1248	004144	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
1249								;NO GO TO ERROR
1250	004146	170012		31:	SETL			;MAKE FL=1
1251	004150	170201			STFPS	R1		;STORE FPS
1252	004152	020104			CMP	R1, R4		;IS FL=1
1253	004154	001403			BEQ	41		;YES GO ON
1254	004156	104000			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
1255	004160	000024			.WORD	24		;UNIQUE ERROR NUMBER
1256	004162	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
1257								;NO GO TO ERROR
1258	004164			41:				
1259								
1260								
1261	004164							
1262								
1263								
1264								
1265	004164							
1266	004164	005267	174614		INC	#TESTN		;INCREMENT TEST NUMBER
1267	004170	005037	002122		CLR	#TRPFLG		;CLEAR TRAP FLAG
1268	004174	012705	170003		MOV	#170003, R5		;INIT OP CODE
1269	004200	013746	000244		MOV	#0244, -(SP)		;SAVE FP VECTOR
1270	004204	012737	004340	000244	MOV	#ILLOP1, #0244		;SETUP NEW VECTOR
1271	004212	013746	000004		MOV	#04, -(SP)		;SAVE TIME OUT VECTOR
1272	004216	012737	004424	000004	MOV	#TIMEOU, #04		;SETUP NEW VECTOR
1273	004224	013746	000010		MOV	#010, -(SP)		;SAVE ILLEGAL VECTOR
1274	004230	012737	004434	000010	MOV	#ILLOP2, #010		;SETUP NEW VECTOR
1275	004236	005003			CLR	R3		
1276	004240	170103		D1:	LDFPS	R3		;CLEAR FPS
1277	004242	005002			CLR	R2		
1278	004244	010537	004250		MOV	R5, #0D2		;SETUP THE ILLEGAL INST

1279	004250	000000		D2:	.WORD	0		
1280	004252	170000		D3:	CFCC			
1281	004254	005202			INC	R2		; MEMORY WORDS TO BE USED WITH
1282	004256	005202			INC	R2		; EXECUTION OF ILLEGAL OP CODE
1283	004260	170201			STFPS	R1		; SAVE FPS
1284	004262	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
1285	004264	000025			.WORD	25		; UNIQUE ERROR NUMBER
1286	004266	002013			.WORD	FPPERR		; ADDRESS OF ERROR MESSAGE
1287								; GO TO ERROR
1288	004270	022705	170010	D4:	CMP	#170010,R5		; COMPUTE NEXT OP CODE
1289	004274	001003			BNE	D5		
1290	004276	012705	170013		MOV	#170013,R5		
1291	004302	000755			BR	D1		
1292	004304	022705	170077	D5:	CMP	#170077,R5		
1293	004310	001001			BNE	D6		
1294	004312	000402			BR	D7		
1295	004314	005205		D6:	INC	R5		
1296	004316	000747			BR	D1		
1297	004320	012637	000010	D7:	MOV	(SP)+,B#10		; RESTORE VECTORS
1298	004324	012637	000004		MOV	(SP)+,B#4		
1299	004330	012637	000244		MOV	(SP)+,B#244		
1300								
1301								
1302	004334	000167	000104		JMP	FIN4		
1303								
1304	004340	022716	004252	ILLOP1:	CMP	#D3,(SP)		; DID TRAP OCCUR ON TEST INST
1305	004344	001403			BEQ	1:		; YES GO ON
1306	004346	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
1307	004350	000026			.WORD	26		; UNIQUE ERROR NUMBER
1308	004352	002013			.WORD	FPPERR		; ADDRESS OF ERROR MESSAGE
1309								; NO GO TO ERROR
1310	004354	022626		1:	CMP	(SP)+,(SP)+		; CLEAN UP STACK
1311	004356	170201			STFPS	R1		; STORE FPS
1312	004360	022701	100000		CMP	#100000,R1		; IS FPS CORRECT
1313	004364	001403			BEQ	2:		; YES GO ON
1314	004366	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
1315	004370	000027			.WORD	27		; UNIQUE ERROR NUMBER
1316	004372	002013			.WORD	FPPERR		; ADDRESS OF ERROR MESSAGE
1317								; NO GO TO ERROR
1318	004374	005004		2:	CLR	R4		; INT R4 TO A KNOWN STATE
1319	004376	170304			STST	R4		; STORE FEC AT R4
1320								; IF THE DESTINATION MODE IS IMPROPERLY
1321								; DECODED AN ODD ADDRESS TRAP TO 4
1322								; SHOULD OCCUR
1323	004400	022704	000002		CMP	#2,R4		; IS FEC CORRECT
1324	004404	001002			BNE	3:		; NO GO TO ERROR
1325	004406	000167	177656		JMP	D4		; YES GO ON
1326	004412			3:				
1327	004412	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR
1328	004414	000030			.WORD	30		; UNIQUE ERROR NUMBER
1329	004416	002013			.WORD	FPPERR		; ADDRESS OF ERROR MESSAGE
1330								; GO TO ERROR
1331	004420	000167	177644		JMP	D4		; THEN GO ON
1332								
1333	004424			TIMEOU:				
1334	004424	104000			ERROR			; ALL ERRORS TO TRAP TO EMT VECTOR

Address	Offset	Value	Instruction	Comment
1335	004426	000031	.WORD 31	UNIQUE ERROR NUMBER
1336	004430	002013	.WORD FPPERR	ADDRESS OF ERROR MESSAGE
1337				ERROR BECAUSE OF TRAP TO 4
1338	004432	000006	RTT	RETURN
1339				
1340	004434		ILL0P2:	
1341	004434	104000	ERROR	ALL ERRORS TO TRAP TO EMT VECTOR
1342	004436	000032	.WORD 32	UNIQUE ERROR NUMBER
1343	004440	002013	.WORD FPPERR	ADDRESS OF ERROR MESSAGE
1344				ERROR BECAUSE OF TRAP TO 10
1345	004442	000006	RTT	RETURN
1346	004444		FIN4:	
1347				
1348	004444		TSFP5:	
1349			*****	
1350			*TEST 7	TEST FID (INTERRUPT DISABLE BIT)
1351			*****	
1352	004444		TST7:	
1353	004444	005267	INC \$TESTN	INCREMENT TEST NUMBER
1354	004450	005037	CLR \$TRPFLG	CLEAR TRAP FLAG
1355	004454	013746	MOV \$244, -(SP)	SAVE FP VECTOR
1356	004460	012737	MOV \$ILL, \$244	SETUP NEW VECTOR
1357	004466	012703	MOV \$40000, R3	SETUP DATA TO BE LOADED
1358	004472	170103	LDFPS R3	LOAD FPS, FID=1
1359	004474	170020	.WORD 170020	ILLEGAL FP INSTRUCTION
1360	004476	170000	CFCC	
1361	004500	170201	STFPS R1	SEE IF ERROR WAS RECORDED IN FPS
1362	004502	022701	CMP \$140000, R1	
1363	004506	001403	BEQ 1\$	YES GO ON
1364	004510	104000	ERROR	ALL ERRORS TO TRAP TO EMT VECTOR
1365	004512	000033	.WORD 33	UNIQUE ERROR NUMBER
1366	004514	002013	.WORD FPPERR	ADDRESS OF ERROR MESSAGE
1367				NO GO TO ERROR
1368	004516	170304	1\$: STST R4	SEE IF FEC=2
1369	004520	022704	CMP \$2, R4	
1370	004524	001403	BEQ 2\$	YES GO ON
1371	004526	104000	ERROR	ALL ERRORS TO TRAP TO EMT VECTOR
1372	004530	000034	.WORD 34	UNIQUE ERROR NUMBER
1373	004532	002013	.WORD FPPERR	ADDRESS OF ERROR MESSAGE
1374				NO GO TO ERROR
1375	004534	012637	2\$: MOV (SP)+, \$244	RESTORE VECTOR
1376				
1377				
1378	004540	000167	JMP FIN5	
1379		000010		
1380	004544		ILL:	
1381	004544	104000	ERROR	ALL ERRORS TO TRAP TO EMT VECTOR
1382	004546	000035	.WORD 35	UNIQUE ERROR NUMBER
1383	004550	002013	.WORD FPPERR	ADDRESS OF ERROR MESSAGE
1384				FID ERROR
1385	004552	000006	RTT	RETURN
1386	004554		FIN5:	
1387				
1388	004554		TSFP6:	
1389			*****	
1390			*TEST 10	TEST LOD, STD FSRC AND FDST MODE 1


```
1391
1392 004554
1393 004554 005267 174224
1394 004560 005037 002122
1395 004564 005004
1396 004566 170104
1397 004570 170011
1398 004572 013746 000004
1399 004576 012737 004750 000004
1400 004604 012704 004740
1401 004610 172414
1402 004612 020427 004740
1403 004616 001403
1404 004620 104000
1405 004622 000036
1406 004624 002013
1407
1408 004626 012701 004730
1409 004632 012703 000004
1410 004636 022421
1411 004640 001403
1412 004642 104000
1413 004644 000037
1414 004646 002013
1415
1416 004650 077306
1417 004652 012704 001136
1418 004656 174014
1419 004660 020427 001136
1420 004664 001403
1421 004666 104000
1422 004670 000040
1423 004672 002013
1424
1425 004674 012701 004730
1426 004700 012703 000004
1427 004704 022421
1428 004706 001403
1429 004710 104000
1430 004712 000041
1431 004714 002013
1432
1433 004716 077306
1434 004720 012637 000004
1435
1436
1437 004724 000167 000030
1438
1439 004730 177777
1440 004732 000000
1441 004734 052525
1442 004736 125252
1443 004740 177777
1444 004742 000000
1445 004744 052525
1446 004746 125252

;*****
TST10:
INC $TESTN ;INCREMENT TEST NUMBER
CLR $TRPFLG ;CLEAR TRAP FLAG
CLR R4 ;SETUP TO LOAD DATA
LDFPS R4 ;CLEAR FPS
SETD ;SET FD TO 1
MOV $4,-(SP) ;SAVE TIMEOUT VECTOR
MOV $TSF6,$4 ;SETUP NEW VECTOR
MOV $TS6DAT,R4 ;SETUP POINTER TO DATA
LDD (R4),ACO ;TEST INSTRUCTION
CMP R4,$TS6DAT ;IS R4 CORRECT
BEQ 1$ ;YES GO ON
ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 36 ;UNIQUE ERROR NUMBER
.WORD FPPERR ;ADDRESS OF ERROR MESSAGE
;NO GO TO ERROR
MOV $TS6DA,R1 ;SETUP POINTER TO DATA
MOV $4,R3 ;INIT COUNTER
2$: CMP (R4)+,(R1)+ ;WAS SOURCE DATA ALTERED
BEQ 3$ ;NO GO ON
ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 37 ;UNIQUE ERROR NUMBER
.WORD FPPERR ;ADDRESS OF ERROR MESSAGE
;YES GO TO ERROR
SOB R3,2$ ;ARE WE DONE
MOV $TSTLOC,R4 ;SETUP POINTER FOR DATA
STD ACO,(R4) ;TEST INSTRUCTION
CMP R4,$TSTLOC ;IS R4 CORRECT
BEQ 4$ ;YES GO ON
ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 40 ;UNIQUE ERROR NUMBER
.WORD FPPERR ;ADDRESS OF ERROR MESSAGE
;NO GO TO ERROR
MOV $TS6DA,R1 ;SETUP POINTER TO DATA
MOV $4,R3 ;INIT COUNTER
5$: CMP (R4)+,(R1)+ ;IS DESTINATION DATA CORRECT
BEQ 6$ ;YES GO ON
ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 41 ;UNIQUE ERROR NUMBER
.WORD FPPERR ;ADDRESS OF ERROR MESSAGE
;NO GO TO ERROR
SOB R3,5$ ;ARE WE DONE
MOV (SP)+,$4 ;RESTORE VECTOR
;
JMP FIN6
;
TS6DA: .WORD 177777
.WORD 000000
.WORD 052525
.WORD 125252
TS6DAT: .WORD 177777
.WORD 000000
.WORD 052525
.WORD 125252
```

```

1447
1448 004750
1449 004750 104000
1450 004752 000042
1451 004754 002013
1452
1453 004756 000006
1454 004760
1455
1456 004760
1457
1458
1459
1460 004760
1461 004760 005267 174020
1462 004764 005037 002122
1463 004770 012704 000200
1464 004774 170104
1465 004776 013746 000004
1466 005002 012737 005154 000004
1467 005010 012704 005164
1468 005014 172414
1469 005016 012701 005174
1470 005022 172511
1471 005024 172401
1472 005026 012704 001136
1473 005032 174114
1474 005034 004767 000072
1475 005040 012704 001136
1476 005044 012701 005174
1477 005050 174014
1478 005052 004767 000054
1479 005056 012701 005164
1480 005062 172511
1481 005064 170001
1482 005066 172401
1483 005070 170011
1484 005072 012704 001136
1485 005076 174114
1486 005100 004767 000026
1487 005104 012704 005204
1488 005110 012701 001136
1489 005114 174011
1490 005116 004767 000010
1491 005122 012637 000004
1492
1493
1494 005126 000167 000062
1495
1496 005132 012703 000004
1497 005136 022421
1498 005140 001403
1499 005142 104000
1500 005144 000043
1501 005146 002013
1502

;SF6:
ERROR
.WORD 42
.WORD FPPERR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;ODD ADDRESS TRAP
;RETURN

FIN6:
;TSFP7:
;*****
;TEST 11 TEST LDD, LDF FSRC MODE 0
;*****
TST11:
INC $TESTN ;INCREMENT TEST NUMBER
CLR $TRPFLG ;CLEAR TRAP FLAG
MOV $200,R4 ;SETUP TO LOAD FPS
LDFPS R4 ;LOAD FPS, FD=1
MOV $04,-(SP) ;SAVE TIMEOUT VECTOR
MOV $TSF7,$04 ;SETUP NEW VECTOR
MOV $TS7DA1,R4 ;SETUP POINTER TO DATA
LDD (R4),AC0 ;CLEAR AC0
MOV $TS7DA2,R1 ;SETUP POINTER TO DATA
LDD (R1),AC1 ;LOAD AC1 WITH DATA
LDD AC1,AC0 ;TEST INSTRUCTION
MOV $TSTLOC,R4 ;
STD AC1,(R4) ;CHECK IF AC1 HAS BEEN ALTERED
JSR PC,CHECK7 ;
MOV $TSTLOC,R4 ;SETUP POINTERS FOR DATA
MOV $TS7DA2,R1 ;
STD AC0,(R4) ;CHECK IF ACO RECEIVED CORRECT DATA
JSR PC,CHECK7 ;
MOV $TS7DA1,R1 ;SETUP POINTER TO DATA
LDD (R1),AC1 ;CLEAR AC1
SETF AC1,AC0 ;SET FD=0
LDF AC1,AC0 ;TEST INSTRUCTION
SETD ;SET FD=1
MOV $TSTLOC,R4 ;SETUP POINTER TO DATA
STD AC1,(R4) ;CHECK IF AC1 HAS BEEN ALTERED
JSR PC,CHECK7 ;
MOV $TS7DA4,R4 ;SETUP POINTERS FOR DATA
MOV $TSTLOC,R1 ;
STD ACO,(R1) ;CHECK IF ACO HAS CORRECT DATA
JSR PC,CHECK7 ;
MOV (SP)+,$04 ;RESTORE VECTOR

;
; JMP FIN7
;
CHECK7: MOV $4,R3 ;INIT COUNTER
CHECK7: CMP (R4)+,(R1)+ ;IS DATA OK
BEQ CHK7 ;YES GO ON
ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 43 ;UNIQUE ERROR NUMBER
.WORD FPPERR ;ADDRESS OF ERROR MESSAGE
;NO GO TO ERROR

```


1503	005150	077306		CHK7:	SOB	R3,CHK7		;ARE WE DONE
1504	005152	000207			RTS	PC		;YES RETURN
1505								
1506	005154			TSF7:				
1507	005154	104000			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
1508	005156	000044			.WORD	44		;UNIQUE ERROR NUMBER
1509	005160	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
1510								;ODD ADDRESS TRAP
1511	005162	000006			RTT			;RETURN
1512								
1513	005164	000000		TS7DA1:	.WORD	0		
1514	005166	000000			.WORD	0		
1515	005170	000000			.WORD	0		
1516	005172	000000			.WORD	0		
1517	005174	037641		TS7DA2:	.WORD	37641		
1518	005176	065121			.WORD	65121		
1519	005200	037373			.WORD	37373		
1520	005202	022265			.WORD	22265		
1521	005204	000000		TS7DA4:	.WORD	0		
1522	005206	000000			.WORD	0		
1523	005210	037373			.WORD	37373		
1524	005212	022265			.WORD	22265		
1525	005214			FIN7:				
1526								
1527	005214			TSFP10:				
1528				;;*****				
1529				;;*****				
1530				;;*****				
1531	005214			TS12:				
1532	005214	005267	173564		INC	%TESTN		;INCREMENT TEST NUMBER
1533	005220	005037	002122		CLR	%TRPFLG		;CLEAR TRAP FLAG
1534	005224	012704	000200		MOV	%200,R4		;SETUP TO LOAD FPS
1535	005230	170104			LDFPS	R4		;LOAD FPS, FD=1
1536	005232	013746	000004		MOV	%4,-(SP)		;SAVE TIMEOUT VECTOR
1537	005236	012737	005410	000004	MOV	%TSF10,%4		;SETUP NEW VECTOR
1538	005244	012704	005420		MOV	%TS10D1,R4		;SETUP POINTER TO DATA
1539	005250	172414			LDD	(R4),ACO		;CLEAR ACO
1540	005252	012701	005430		MOV	%TS10D2,R1		;SETUP POINTER TO DATA
1541	005256	172511			LDD	(R1),AC1		;LOAD AC1 WITH DATA
1542	005260	174100			STD	AC1,ACO		;TEST INSTRUCTION
1543	005262	012704	001136		MOV	%TSTLOC,R4		
1544	005266	174114			STD	AC1,(R4)		;CHECK IF AC1 HAS BEEN ALTERED
1545	005270	004767	000072		JSR	PC,CHC10		
1546	005274	012704	001136		MOV	%TSTLOC,R4		;SETUP POINTERS FOR DATA
1547	005300	012701	005430		MOV	%TS10D2,R1		
1548	005304	174014			STD	ACO,(R4)		;CHECK IF ACO RECEIVED CORRECT DATA
1549	005306	004767	000054		JSR	PC,CHC10		
1550	005312	012701	005420		MOV	%TS10D1,R1		;SETUP POINTER TO DATA
1551	005316	172511			LDD	(R1),AC1		;CLEAR AC1
1552	005320	170001			SETF			;SET FD=0
1553	005322	174100			STF	AC1,ACO		;TEST INSTRUCTION
1554	005324	170011			SETD			;SET FD=1
1555	005326	012704	001136		MOV	%TSTLOC,R4		;SETUP POINTER TO DATA
1556	005332	174114			STD	AC1,(R4)		;CHECK IF AC1 HAS BEEN ALTERED
1557	005334	004767	000026		JSR	PC,CHC10		
1558	005340	012704	005440		MOV	%TS10D4,R4		;SETUP POINTERS FOR DATA

```

1559 005344 012701 001136      MOV    #TSTLOC,R1
1560 005350 174011              STD     ACO,(R1)
1561 005352 004767 000010      JSR     PC,CHEC10
1562 005356 012637 000004      MOV     (SP)+,#04
1563
1564
1565 005362 000167 000062      JMP     FIN10
1566
1567 005366 012703 000004      CHEC10: MOV    #4,R3
1568 005372 022421              CH10:  CMP     (R4)+,(R1)+
1569 005374 001403              BEQ     CHK10
1570 005376 104000              ERROR
1571 005400 000045              .WORD   45
1572 005402 002013              .WORD   FPPERR
1573
1574 005404 077306      CHK10: SOB     R3,CH10
1575 005406 000207      RTS     PC
1576
1577 005410
1578 005410 104000      TSF10:  ERROR
1579 005412 000046      .WORD   46
1580 005414 002013      .WORD   FPPERR
1581
1582 005416 000006      RTT
1583
1584 005420 000000      TS10D1: .WORD   0
1585 005422 000000      .WORD   0
1586 005424 000000      .WORD   0
1587 005426 000000      .WORD   0
1588 005430 177777      TS10D2: .WORD   177777
1589 005432 111236      .WORD   111236
1590 005434 100045      .WORD   100045
1591 005436 003651      .WORD   3651
1592 005440 000000      TS10D4: .WORD   0
1593 005442 000000      .WORD   0
1594 005444 100045      .WORD   100045
1595 005446 003651      .WORD   3651
1596 005450
1597
1598 005450
1599
1600
1601
1602 005450
1603 005450 005267 173330
1604 005454 005037 002122
1605 005460 012704 000200
1606 005464 170104
1607 005466 012704 005550
1608 005472 172414
1609 005474 170400
1610 005476 170203
1611 005500 012704 001136
1612 005504 174014
1613 005506 012701 000004
1614 005512 022427 000000

      INC     #TESTN
      CLR     @TRPFLG
      MOV     #200,R4
      LDFPS   R4
      MOV     #TS11D1,R4
      LDD     (R4),ACO
      CLRD    ACO
      STFPS   R3
      MOV     #TSTLOC,R4
      STD     ACO,(R4)
      MOV     #4,R1
      CMP     (R4)+,#0

;CHECK IF ACO HAS CORRECT DATA
;RESTORE VECTOR
;INIT COUNTER
;IS DATA OK
;YES GO ON
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;NO GO TO ERROR
;ARE WE DONE
;YES RETURN
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;ODD ADDRESS TRAP
;
;*****
;*TEST 13      TEST FDST SINGLE OPERAND MODE 0
;*****
TST13:
;INCREMENT TEST NUMBER
;CLEAR TRAP FLAG
;SETUP TO LOAD FPS
;SET FD=1
;SETUP POINTER TO DATA
;LOAD ALL ONES TO ACO
;TEST INSTRUCTION
;GET FPS
;
;CHECK ACO FOR ALL ZEROES
;INIT COUNTER
;

```



```

1615 005516 001403      BEQ      24      ;OK GO ON
1616 005520 104000      ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
1617 005522 000047      .WORD     47      ;UNIQUE ERROR NUMBER
1618 005524 002013      .WORD     FPPERR  ;ADDRESS OF ERROR MESSAGE
1619                                     ;NO GO TO ERROR
1620 005526 077107      24:      SOB      R1,14      ;ARE WE DONE
1621 005530 020327 000204      CMP      R3,#204      ;CHECK FPS
1622 005534 001403      BEQ      34      ;OK GO ON
1623 005536 104000      ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
1624 005540 000050      .WORD     50      ;UNIQUE ERROR NUMBER
1625 005542 002013      .WORD     FPPERR  ;ADDRESS OF ERROR MESSAGE
1626                                     ;NO GO TO ERROR
1627 005544      34:
1628
1629
1630 005544 000167 000010      ;      JMP      FIN11
1631
1632 005550 177777      ;TS11D1: .WORD     177777
1633 005552 177777      .WORD     177777
1634 005554 177777      .WORD     177777
1635 005556 177777      .WORD     177777
1636 005560      FIN11:
1637      ;
1638 005560      TSFP12:
1639      ;*****
1640      ;*TEST 14      TEST FDST SOP MODE 0 WITH ILLEGAL AC7
1641      ;*****
1642 005560      TST14:
1643 005560 005267 173220      INC      $TESTN      ;INCREMENT TEST NUMBER
1644 005564 005037 002122      CLR      @TRPFLG      ;CLEAR TRAP FLAG
1645 005570 012703 040200      MOV      @40200,R3      ;SETUP TO LOAD FPS
1646 005574 170103      LDFPS      R3      ;SET FID=1, AND FD=1
1647 005576 170407      CLRD      AC7      ; TEST INSTRUCTION
1648 005600 170204      STFPS      R4      ;GET FPS
1649 005602 170305      STST      R5      ;GET FEC
1650 005604 022704 140200      CMP      @140200,R4      ;IS FPS CORRECT
1651 005610 001403      BEQ      14      ;YES GO ON
1652 005612 104000      ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
1653 005614 000051      .WORD     51      ;UNIQUE ERROR NUMBER
1654 005616 002013      .WORD     FPPERR  ;ADDRESS OF ERROR MESSAGE
1655                                     ;NO GO TO ERROR
1656 005620 022705 000002      14:      CMP      @2,R5      ;IS FEC CORRECT
1657 005624 001403      BEQ      24      ;YES GO ON
1658 005626 104000      ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
1659 005630 000052      .WORD     52      ;UNIQUE ERROR NUMBER
1660 005632 002013      .WORD     FPPERR  ;ADDRESS OF ERROR MESSAGE
1661                                     ;NO GO TO ERROR
1662 005634      24:
1663
1664
1665 005634      ;
1666      ;TSFP13:
1667      ;*****
1668      ;*TEST 15      TEST FDST SOP MODE 1
1669      ;*****
1670 005634 005267 173144      TST15:      INC      $TESTN      ;INCREMENT TEST NUMBER
  
```

Address	Offset	Hex	Hex	Label	Instruction	Comment
1671	005640	013746	000004		MOV	004, -(SP)
1672	005644	012737	005774	000004	MOV	0TSF13, 004
1673	005652	005037	002122		CLR	00TRPFLG
1674	005656	012702	000200		MOV	0200, R2
1675	005662	170102			LDFPS	R2
1676	005664	012705	000004		MOV	004, R5
1677	005670	012704	001136		MOV	0TSTLOC, R4
1678	005674	012724	177777	100\$:	MOV	0177777, (R4)+
1679	005700	077503			SOB	R5, 100\$
1680	005702	012702	001136		MOV	0TSTLOC, R2
1681	005706	170412			CLRD	(R2)
1682	005710	170203			STFPS	R3
1683	005712	020227	001136		CMP	R2, 0TSTLOC
1684	005716	001403			BEQ	1\$
1685	005720	104000			ERROR	
1686	005722	000053			.WORD	53
1687	005724	002013			.WORD	FPPERR
1688						
1689	005726	012701	000004	1\$:	MOV	004, R1
1690	005732	022227	000000	2\$:	CMP	(R2)+, 00
1691	005736	001403			BEQ	3\$
1692	005740	104000			ERROR	
1693	005742	000054			.WORD	54
1694	005744	002013			.WORD	FPPERR
1695						
1696	005746	077107		3\$:	SOB	R1, 2\$
1697	005750	020327	000204		CMP	R3, 0204
1698	005754	001403			BEQ	4\$
1699	005756	104000			ERROR	
1700	005760	000055			.WORD	55
1701	005762	002013			.WORD	FPPERR
1702						
1703	005764	012637	000004	4\$:	MOV	(SP)+, 004
1704						
1705						
1706	005770	000167	000010		JMP	FIN13
1707						
1708	005774			TSF13:		
1709	005774	104000			ERROR	
1710	005776	000056			.WORD	56
1711	006000	002013			.WORD	FPPERR
1712						
1713	006002	000006			RTT	
1714						
1715	006004			FIN13:		
1716						
1717	006004			TSFP14:		
1718						
1719						
1720						
1721	006004					
1722	006004	005267	172774		INC	0TESTN
1723	006010	013746	000004		MOV	004, -(SP)
1724	006014	012737	006150	000004	MOV	0TSF14, 004
1725	006022	005037	002122		CLR	00TRPFLG
1726	006026	012702	000200		MOV	0200, R2

1727	006032	170102		LDFPS	R2	;SET FD=1
1728	006034	012705	000004	MOV	#4,R5	;INIT COUNTER
1729	006040	012704	001136	MOV	#TSTLOC,R4	;SETUP POINTER TO TEST LOCATION
1730	006044	012724	177777	MOV	#177777,(R4),	;MOVE ALL ONES TO TEST LOCATION
1731	006050	077503		SOB	R5,100	;ARE WE DONE
1732	006052	012702	001136	MOV	#TSTLOC,R2	;SETUP POINTER TO DATA
1733	006056	170422		CLRD	(R2),	;TEST INSTRUCTION
1734	006060	170203		STFPS	R3	;GET FPS
1735	006062	020227	001146	CMP	R2,#TSTLOC+10	;IS R2 CORRECT
1736	006066	001403		BEQ	1	;YES GO ON
1737	006070	104000		ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1738	006072	000057		.WORD	57	;UNIQUE ERROR NUMBER
1739	006074	002013		.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
1740						;NO GO TO ERROR
1741	006076	012702	001136	1\$: MOV	#TSTLOC,R2	;SETUP POINTER TO DATA
1742	006102	012701	000004	MOV	#4,R1	;INIT COUNTER
1743	006106	022227	000000	2\$: CMP	(R2),#0	;CHECK LOCATION FOR 0
1744	006112	001403		BEQ	3	;YES GO ON
1745	006114	104000		ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1746	006116	000060		.WORD	60	;UNIQUE ERROR NUMBER
1747	006120	002013		.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
1748						;NO GO TO ERROR
1749	006122	077107		3\$: SOB	R1,2	;ARE WE DONE
1750	006124	020327	000204	CMP	R3,#204	;CHECK FPS
1751	006130	001403		BEQ	4	;OK GO ON
1752	006132	104000		ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1753	006134	000061		.WORD	61	;UNIQUE ERROR NUMBER
1754	006136	002013		.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
1755						;NO GO TO ERROR
1756	006140	012637	000004	4\$: MOV	(SP),#4	;RESTORE VECTOR
1757						
1758						
1759	006144	000167	000010	JMP	FIN14	
1760						
1761	006150			TSF14:		
1762	006150	104000		ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1763	006152	000062		.WORD	62	;UNIQUE ERROR NUMBER
1764	006154	002013		.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
1765						;ODD ADDRESS TRAP
1766	006156	000006		RTT		;RETURN
1767						
1768	006160			FIN14:		
1769						
1770	006160			TSFP15:		
1771						
1772				*****		
1773				;*TEST 17	TEST FDST SOP MODE 3	
1774				*****		
1775	006160	005267	172620	TST17:		
1776	006164	013746	000004	INC	#TESTN	;INCREMENT TEST NUMBER
1777	006170	012737	006370	MOV	#4,-(SP)	;SAVE TIMEOUT VECTOR
1778	006176	005037	002122	MOV	#TSF15,#4	;SETUP NEW VECTOR
1779	006202	012702	000200	CLR	#TRPFLG	;CLEAR TRAP FLAG
1780	006206	170102		MOV	#200,R2	;SETUP TO LOAD FPS
1781	006210	012705	000011	LDFPS	R2	;SET FD=1
1782	006214	012704	001136	MOV	#9,R5	;INIT COUNTER
				MOV	#TSTLOC,R4	;SETUP POINTER TO TEST LOCATION

1783	006220	012724	177777	100\$:	MOV	#177777,(R4)+	;INIT TEST LOCATION
1784	006224	077503			SOB	R5,100\$	;ARE WE DONE
1785	006226	012737	001150	001136	MOV	#TSTLOC+12,#TSTLOC	;INIT TEST LOCATION
1786	006234	012702	001136		MOV	#TSTLOC,R2	;SETUP POINTER TO DATA
1787	006240	170432			CLRD	#(R2)+	;TEST INSTRUCTION
1788	006242	170203			STFPS	R3	;GET FPS
1789	006244	020227	001140		CMP	R2,#TSTLOC+2	;IS R2 CORRECT
1790	006250	001403			BEQ	1\$	;YES GO ON
1791	006252	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1792	006254	000063			.WORD	63	;UNIQUE ERROR NUMBER
1793	006256	002013			.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
1794							;NO GO TO ERROR
1795	006260	012702	001136	1\$:	MOV	#TSTLOC,R2	;SETUP POINTER TO DATA
1796	006264	022227	001150		CMP	(R2)+,#TSTLOC+12	;IS DATA CORRECT
1797	006270	001403			BEQ	2\$	;YES GO ON
1798	006272	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1799	006274	000064			.WORD	64	;UNIQUE ERROR NUMBER
1800	006276	002013			.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
1801							;NO GO TO ERROR
1802	006300	012701	000004	2\$:	MOV	#4,R1	;INIT COUNTER
1803	006304	022227	177777	3\$:	CMP	(R2)+,#177777	;IS LOCATION ALL ONES
1804	006310	001403			BEQ	4\$	;YES GO ON
1805	006312	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1806	006314	000065			.WORD	65	;UNIQUE ERROR NUMBER
1807	006316	002013			.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
1808							;NO GO TO ERROR
1809	006320	077107		4\$:	SOB	R1,3\$	;ARE WE DONE
1810	006322	012701	000004		MOV	#4,R1	;INIT COUNTER
1811	006326	022227	000000	5\$:	CMP	(R2)+,#0	;IS LOCATION 0
1812	006332	001403			BEQ	6\$	;YES GO ON
1813	006334	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1814	006336	000066			.WORD	66	;UNIQUE ERROR NUMBER
1815	006340	002013			.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
1816							;NO GO TO ERROR
1817	006342	077107		6\$:	SOB	R1,5\$	;ARE WE DONE
1818	006344	020327	000204		CMP	R3,#204	;CHECK FPS
1819	006350	001403			BEQ	7\$	;OK GO ON
1820	006352	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1821	006354	000067			.WORD	67	;UNIQUE ERROR NUMBER
1822	006356	002013			.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
1823							;NO GO TO ERROR
1824	006360	012637	000004	7\$:	MOV	(SP)+,#04	;RESTORE VECTOR
1825							
1826							
1827	006364	000167	000010		JMP	FIN15	
1828							
1829	006370			TSF15:			
1830	006370	104000			ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
1831	006372	000070			.WORD	70	;UNIQUE ERROR NUMBER
1832	006374	002013			.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
1833							;ODD ADDRESS TRAP
1834	006376	000006			RTT		;RETURN
1835							
1836	006400			FIN15:			
1837							
1838	006400			TSFP16:			


```

1839
1840
1841
1842 006400
1843 006400 005267 172400
1844 006404 013746 000004 000004
1845 006410 012737 006562
1846 006416 005037 002122
1847 006422 012702 000200
1848 006426 170102
1849 006430 012705 000010
1850 006434 012704 001136
1851 006440 012724 177777
1852 006444 077503
1853 006446 012702 001146
1854 006452 170442
1855 006454 170203
1856 006456 020227 001136
1857 006462 001403
1858 006464 104000
1859 006466 000071
1860 006470 002013
1861
1862 006472 012701 000004
1863 006476 022227 000000
1864 006502 001403
1865 006504 104000
1866 006506 000072
1867 006510 002013
1868
1869 006512 077107
1870 006514 012701 000004
1871 006520 022227 177777
1872 006524 001403
1873 006526 104000
1874 006530 000073
1875 006532 002013
1876
1877 006534 077107
1878 006536 020327 000204
1879 006542 001403
1880 006544 104000
1881 006546 000074
1882 006550 002013
1883
1884 006552 012637 000004
1885
1886
1887 006556 000167 000010
1888
1889 006562
1890 006562 104000
1891 006564 000075
1892 006566 002013
1893
1894 006570 000006

;*****
;TEST 20 TEST FDST SOP MODE 4
;*****
TST20:
INC 1TESTN ;INCREMENT TEST NUMBER
MOV 004,-(SP) ;SAVE TIMEOUT VECTOR
MOV 0TSF16,004 ;SETUP NEW VECTOR
CLR 00TRPFLG ;CLEAR TRAP FLAG
MOV 0200,R2 ;SETUP TO LOAD FPS
LDFPS R2 ;SET FD=1
MOV 08,R5 ;INIT COUNTER
MOV 0TSTLOC,R4 ;SETUP POINTER TO TEST LOCATION
MOV 0177777,(R4) ;INIT TEST LOCATION
SOB R5,1001 ;ARE WE DONE
MOV 0TSTLOC+10,R2 ;SETUP POINTER TO DATA
CLRD -(R2) ;TEST INSTRUCTION
STFPS R3 ;GET FPS
CMP R2,0TSTLOC ;IS R2 CORRECT
BEQ 11 ;YES GO ON
ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 71 ;UNIQUE ERROR NUMBER
.WORD FPPERR ;ADDRESS OF ERROR MESSAGE
;NO GO TO ERROR
MOV 04,R1 ;INIT COUNTER
CMP (R2)+,00 ;IS LOCATION 0
BEQ 31 ;YES GO ON
ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 72 ;UNIQUE ERROR NUMBER
.WORD FPPERR ;ADDRESS OF ERROR MESSAGE
;NO GO TO ERROR
SOB R1,21 ;ARE WE DONE
MOV 04,R1 ;INIT COUNTER
CMP (R2)+,0177777 ;IS LOCATION UNCHANGED
BEQ 51 ;YES GO ON
ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 73 ;UNIQUE ERROR NUMBER
.WORD FPPERR ;ADDRESS OF ERROR MESSAGE
;NO GO TO ERROR
SOB R1,41 ;ARE WE DONE
CMP R3,0204 ;CHECK FPS
BEQ 61 ;OK GO ON
ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 74 ;UNIQUE ERROR NUMBER
.WORD FPPERR ;ADDRESS OF ERROR MESSAGE
;NO GO TO ERROR
MOV (SP)+,004 ;RESTORE VECTOR
;
JMP FIN16
;
;TSF16:
ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 75 ;UNIQUE ERROR NUMBER
.WORD FPPERR ;ADDRESS OF ERROR MESSAGE
;ODD ADDRESS TRAP
RTT ;RETURN

```

```

1895
1896 006572
1897
1898 006572
1899
1900
1901
1902 006572
1903 006572 005267 172206
1904 006576 013746 000004
1905 006602 012737 006776 000004
1906 006610 005037 002122
1907 006614 012702 000200
1908 006620 170102
1909 006622 012705 000011
1910 006626 012704 001136
1911 006632 012724 177777
1912 006636 077503
1913 006640 012737 001150 001136
1914 006646 012702 001140
1915 006652 170452
1916 006654 170203
1917 006656 020227 001136
1918 006662 001403
1919 006664 104000
1920 006666 000076
1921 006670 002013
1922
1923 006672 022227 001150
1924 006676 001403
1925 006700 104000
1926 006702 000077
1927 006704 002013
1928
1929 006706 012701 000004
1930 006712 022227 177777
1931 006716 001403
1932 006720 104000
1933 006722 000100
1934 006724 002013
1935
1936 006726 077107
1937 006730 012701 000004
1938 006734 022227 000000
1939 006740 001403
1940 006742 104000
1941 006744 000101
1942 006746 002013
1943
1944 006750 077107
1945 006752 020327 000204
1946 006756 001403
1947 006760 104000
1948 006762 000102
1949 006764 002013
1950

      FIN16:
      TSFP17:
      *****
      TEST 21      TEST FDST SOP MODE 5
      *****
      TST21:
      INC      $TESTN      ;INCREMENT TEST NUMBER
      MOV      @4,-(SP)      ;SAVE TIMEOUT VECTOR
      MOV      @TSF17,@4      ;SETUP NEW VECTOR
      CLR      @TRPFLG      ;CLEAR TRAP FLAG
      MOV      @200,R2      ;SETUP TO LOAD FPS
      LDFPS    R2      ;SET FD=1
      MOV      @9,R5      ;INIT COUNTER
      MOV      @TSTLOC,R4      ;SETUP POINTER TO TEST LOCATION
      MOV      @177777,(R4). ;INIT TEST LOCATION
      SOB      R5,100$      ;ARE WE DONE
      MOV      @TSTLOC+12,@TSTLOC      ;INIT TEST LOCATION
      MOV      @TSTLOC+2,R2      ;SETUP POINTER TO DATA
      CLRD     @-(R2)      ; TEST INSTRUCTION
      STFPS    R3      ;GET FPS
      CMP      R2,@TSTLOC      ;IS R2 CORRECT
      BEQ      1$      ;YES GO ON
      ERROR    76      ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    FPPERR      ;UNIQUE ERROR NUMBER
      .WORD    FPPERR      ;ADDRESS OF ERROR MESSAGE
      ;NO GO TO ERROR
      CMP      (R2)+,@TSTLOC+12      ;IS DATA CORRECT
      BEQ      2$      ;YES GO ON
      ERROR    77      ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    FPPERR      ;UNIQUE ERROR NUMBER
      .WORD    FPPERR      ;ADDRESS OF ERROR MESSAGE
      ;NO GO TO ERROR
      MOV      @4,R1      ;INIT COUNTER
      CMP      (R2)+,@177777      ;IS LOCATION ALL ONES
      BEQ      4$      ;YES GO ON
      ERROR    100      ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    FPPERR      ;UNIQUE ERROR NUMBER
      .WORD    FPPERR      ;ADDRESS OF ERROR MESSAGE
      ;NO GO TO ERROR
      SOB      R1,3$      ;ARE WE DONE
      MOV      @4,R1      ;INIT COUNTER
      CMP      (R2)+,@0      ;IS LOCATION 0
      BEQ      6$      ;YES GO ON
      ERROR    101      ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    FPPERR      ;UNIQUE ERROR NUMBER
      .WORD    FPPERR      ;ADDRESS OF ERROR MESSAGE
      ;NO GO TO ERROR
      SOB      R1,5$      ;ARE WE DONE
      CMP      R3,@204      ;CHECK FPS
      BEQ      7$      ;OK GO ON
      ERROR    102      ;ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    FPPERR      ;UNIQUE ERROR NUMBER
      .WORD    FPPERR      ;ADDRESS OF ERROR MESSAGE
      ;NO GO TO ERROR
  
```



```

1951 006766 012637 000004 7: MOV (SP),004 ;RESTORE VECTOR
1952
1953
1954 006772 000167 000010 ;
1955 ;
1956 006776 TSF17:
1957 006776 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1958 007000 000103 .WORD 103 ;UNIQUE ERROR NUMBER
1959 007002 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
1960 ;ODD ADDRESS TRAP
1961 007004 000006 RTT ;RETURN
1962
1963 007006 ;FIN17:
1964 ;
1965 007006 TSFP20:
1966 ;*****
1967 ;*TEST 22 TEST FOST SOP MODE 6
1968 ;*****
1969 007006 TST22:
1970 007006 005267 171772 INC ;TESTN ;INCREMENT TEST NUMBER
1971 007012 005037 002122 CLR ;TRPFLG ;CLEAR TRAP FLAG
1972 007016 013746 000004 MOV ;-(SP) ;SAVE TIMEOUT VECTOR
1973 007022 012737 007176 000004 MOV ;TSF20,004 ;SETUP NEW VECTOR
1974 007030 012702 000200 MOV ;200,R2 ;SETUP TO LOAD FPS
1975 007034 170102 LDFPS R2 ;SET FD=1
1976 007036 012705 000010 MOV ;R5 ;INIT COUNTER
1977 007042 012704 001136 MOV ;TSTLOC,R4 ;SETUP POINTER TO TEST LOCATION
1978 007046 012724 177777 100: MOV ;177777,(R4) ;INIT TEST LOCATION
1979 007052 077503 SOB R5,100: ;ARE WE DONE
1980 007054 012702 001137 MOV ;TSTLOC+1,R2 ;SETUP POINTER TO DATA
1981 007060 170462 000007 CLRD 7(R2) ;TEST INSTRUCTION
1982 007064 170203 STFPS R3 ;GET FPS
1983 007066 020227 001137 CMP R2,TSTLOC+1 ;IS R2 CORRECT
1984 007072 001403 BEQ 1: ;YES GO ON
1985 007074 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1986 007076 000104 .WORD 104 ;UNIQUE ERROR NUMBER
1987 007100 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
1988 ;NO GO TO ERROR
1989 007102 012702 001136 1: MOV ;TSTLOC,R2 ;SETUP POINTER TO DATA
1990 007106 012701 000004 MOV ;R1 ;INIT COUNTER
1991 007112 022227 177777 2: CMP (R2),177777 ;IS DATA CORRECT
1992 007116 001403 BEQ 3: ;YES GO ON
1993 007120 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
1994 007122 000105 .WORD 105 ;UNIQUE ERROR NUMBER
1995 007124 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
1996 ;NO GO TO ERROR
1997 007126 077107 3: SOB R1,2: ;ARE WE DONE
1998 007130 012701 000004 MOV ;R1 ;INIT COUNTER
1999 007134 022227 000000 4: CMP (R2),00 ;IS DATA CORRECT
2000 007140 001403 BEQ 5: ;YES GO ON
2001 007142 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
2002 007144 000106 .WORD 106 ;UNIQUE ERROR NUMBER
2003 007146 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
2004 ;NO GO TO ERROR
2005 007150 077107 5: SOB R1,4: ;ARE WE DONE
2006 007152 020327 000204 CMP R3,204 ;IS FPS CORRECT

```

```

2007 007156 001403      BEQ      6$      ;YES GO ON
2008 007160 104000      ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
2009 007162 000107      .WORD     107    ;UNIQUE ERROR NUMBER
2010 007164 002013      .WORD     FPPERR ;ADDRESS OF ERROR MESSAGE
2011                                     ;NO GO TO ERROR
2012 007166 012637 000004 6$:      MOV      (SP)+,004    ;RESTORE VECTOR
2013
2014
2015 007172 000167 000010      ;
2016                                     ;
2017                                     ;TSF20:
2018 007176 104000      ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
2019 007200 000110      .WORD     110    ;UNIQUE ERROR NUMBER
2020 007202 002013      .WORD     FPPERR ;ADDRESS OF ERROR MESSAGE
2021                                     ;ODD ADDRESS TRAP
2022 007204 000006      RTT          ;RETURN
2023
2024 007206      ;FIN20:
2025      ;
2026 007206      ;TSFP21:
2027      ;*****
2028      ;*TEST 23      TEST FOST SOP MODE 7
2029      ;*****
2030      ;TST23:
2031 007206 005267 171572      INC      $TESTN      ;INCREMENT TEST NUMBER
2032 007212 005037 002122      CLR      @TRPFLG    ;CLEAR TRAP FLAG
2033 007216 013746 000004      MOV      @04,-(SP)    ;SAVE TIMEOUT VECTOR
2034 007222 012737 007420 000004      MOV      @TSF21,004    ;SETUP NEW VECTOR
2035 007230 012702 000200      MOV      @200,R2      ;SETUP TO LOAD FPS
2036 007234 170102      LDFPS      R2              ;SET FD=1
2037 007236 012705 000010      MOV      @8.,R5       ;INIT COUNTER
2038 007242 012704 001136      MOV      @TSTLOC,R4    ;SETUP POINTER TO TEST LOCATION
2039 007246 012724 177777 100$:      MOV      @177777,(R4). ;INIT TEST LOCATION
2040 007252 077503      SOB      R5,100$           ;ARE WE DONE
2041 007254 012737 001136 001146      MOV      @TSTLOC,@TSTLOC+10 ;INIT TEST LOCATION
2042 007262 012702 001141      MOV      @TSTLOC+3,R2  ;SETUP POINTER TO DATA
2043 007266 170472 000005      CLRD      @5(R2)      ;***TEST INSTRUCTION***
2044 007272 170203      STFPS      R3              ;GET FPS
2045 007274 020227 001141      CMP      R2,@TSTLOC+3  ;IS R2 CORRECT
2046 007300 001403      BEQ      1$              ;YES GO ON
2047                                     ;NO GO TO ERROR
2048 007302 104000      ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
2049 007304 000111      .WORD     111    ;UNIQUE ERROR NUMBER
2050 007306 002013      .WORD     FPPERR ;ADDRESS OF ERROR MESSAGE
2051 007310 012702 001136 1$:      MOV      @TSTLOC,R2  ;SETUP POINTER TO DATA
2052 007314 012701 000004      MOV      @4,R1         ;INIT COUNTER
2053 007320 022227 000000 2$:      CMP      (R2)+,00    ;IS DATA CORRECT
2054 007324 001403      BEQ      3$              ;YES GO ON
2055 007326 104000      ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR
2056 007330 000112      .WORD     112    ;UNIQUE ERROR NUMBER
2057 007332 002013      .WORD     FPPERR ;ADDRESS OF ERROR MESSAGE
2058                                     ;NO GO TO ERROR
2059 007334 077107 001136 3$:      SOB      R1,2$       ;ARE WE DONE
2060 007336 022227      CMP      (R2)+,@TSTLOC    ;IS DATA CORRECT
2061 007342 001403      BEQ      4$              ;YES GO ON
2062 007344 104000      ERROR      ;ALL ERRORS TO TRAP TO EMT VECTOR

```



```

2119 007536 077107          2$: SOB      R1,1$      ;ARE WE DONE
2120 007540 012701 000004    MOV      #4,R1      ;INIT COUNTER
2121 007544 022227 177777    3$: CMP      (R2)+,#177777 ;IS DATA CORRECT
2122 007550 001403          BEQ      4$          ;YES GO ON
2123 007552 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
2124 007554 000120          .WORD     120      ;UNIQUE ERROR NUMBER
2125 007556 002013          .WORD     FPPERR   ;ADDRESS OF ERROR MESSAGE
2126                                     ;NO GO TO ERROR
2127 007560 077107          4$: SOB      R1,3$      ;ARE WE DONE
2128 007562 020327 000204    CMP      R3,#204    ;CHECK FPS
2129 007566 001403          BEQ      5$          ;OK GO ON
2130 007570 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
2131 007572 000121          .WORD     121      ;UNIQUE ERROR NUMBER
2132 007574 002013          .WORD     FPPERR   ;ADDRESS OF ERROR MESSAGE
2133                                     ;NO GO TO ERROR
2134 007576 012637 000004    5$: MOV      (SP)+,#04  ;RESTORE VECTOR
2135
2136                                     ;
2137 007602 000167 000010    ; JMP      FIN22
2138                                     ;
2139 007606          ;TSF22:
2140 007606 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
2141 007610 000122          .WORD     122      ;UNIQUE ERROR NUMBER
2142 007612 002013          .WORD     FPPERR   ;ADDRESS OF ERROR MESSAGE
2143                                     ;ODD ADDRESS TRAP
2144 007614 000006          RTT              ;RETURN
2145
2146                                     ;
2147 007616          ;FIN22:
2148 007616          ;
2149          ;TSFP23:
2150          ;*****
2151          ;*TEST 25      TEST FDST SOP MODE 6 GR7
2152          ;*****
2153 007616 005267 171162    TST25: INC      $TESTN      ;INCREMENT TEST NUMBER
2154 007622 005037 002122    CLR      @TRPFLG    ;CLEAR TRAP FLAG
2155 007626 013746 000004    MOV      @4,-(SP)    ;SAVE TIMEOUT VECTOR
2156 007632 012737 007744 000004    MOV      @TSF23,@4    ;SETUP NEW VECTOR
2157 007640 012702 000200    MOV      #200,R2    ;SETUP TO LOAD FPS
2158 007644 170102          LDFPS     R2          ;SET FD=1
2159 007646 012705 000004    MOV      #4,R5      ;INIT COUNTER
2160 007652 012704 001136    MOV      @TSTLOC,R4   ;SETUP POINTER TO TEST LOCATION
2161 007656 012724 177777    100$: MOV      #177777,(R4)+ ;INIT TEST LOCATION
2162 007662 077503          SOB      R5,100$    ;ARE WE DONE
2163 007664 170467 171246    CLRD      TSTLOC    ;TEST INSTRUCTION
2164 007670 170203          STFPS     R3          ;GET FPS
2165 007672 012701 000004    MOV      #4,R1      ;INIT COUNTER
2166 007676 012702 001136    MOV      @TSTLOC,R2   ;SETUP POINTER TO DATA
2167 007702 022227 000000    1$: CMP      (R2)+,#0    ;IS DATA CORRECT
2168 007706 001403          BEQ      2$          ;YES GO ON
2169 007710 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
2170 007712 000123          .WORD     123      ;UNIQUE ERROR NUMBER
2171 007714 002013          .WORD     FPPERR   ;ADDRESS OF ERROR MESSAGE
2172                                     ;NO GO TO ERROR
2173 007716 077107          2$: SOB      R1,1$      ;ARE WE DONE
2174 007720 020327 000204    CMP      R3,#204    ;CHECK FPS

```



```

2175 007724 001403      BEQ      3$
2176 007726 104000      ERROR
2177 007730 000124      .WORD    124
2178 007732 002013      .WORD    FPPERR
2179
2180 007734 012637 000004 3$: MOV      (SP)+,004      ;NO GO TO ERROR
2181
2182
2183 007740 000167 000010      ;
2184
2185 007744      ;TSF23:
2186 007744 104000      ERROR
2187 007746 000125      .WORD    125
2188 007750 002013      .WORD    FPPERR
2189
2190 007752 000006      RTT
2191 007754
2192
2193 007754      FIN23:
2194      ;TSFP24:
2195      ;*****
2196      ;*TEST 26      TEST FDST SOP MODE 7 GR7
2197      ;*****
2198 007754 005267 171024      TST26:
2199 007760 005037 002122      INC      $TESTN
2200 007764 013746 000004      CLR      @TRPFLG
2201 007770 012737 010146 000004      MOV      @4,-(SP)
2202 007776 012702 000200      MOV      @TSF24,@4
2203 010002 170102      MOV      @200,R2
2204 010004 012705 000010      LDFPS   R2
2205 010010 012704 001136      MOV      @8,R5
2206 010014 012724 177777      MOV      @TSTLOC,R4
2207 010020 077503      MOV      @177777,(R4)+
2208 010022 012737 001136 001146      SOB      R5,100$
2209 010030 170477 171112      MOV      @TSTLOC,@TSTLOC+10
2210 010034 170203      CLRD     @TSTLOC+10
2211 010036 012702 001136      STFPS   R3
2212 010042 012701 000004      MOV      @TSTLOC,R2
2213 010046 022227 000000      MOV      @4,R1
2214 010052 001403      1$: CMP      (R2)+,00
2215      BEQ      2$
2216 010054 104000      ERROR
2217 010056 000126      .WORD    126
2218 010060 002013      .WORD    FPPERR
2219 010062 077107      SOB      R1,1$
2220 010064 022227 001136      CMP      (R2)+,@TSTLOC
2221 010070 001403      BEQ      3$
2222
2223 010072 104000      ERROR
2224 010074 000127      .WORD    127
2225 010076 002013      .WORD    FPPERR
2226 010100 012701 000003      3$: MOV      @3,R1
2227 010104 022227 177777      4$: CMP      (R2)+,@177777
2228 010110 001403      BEQ      5$
2229 010112 104000      ERROR
2230 010114 000130      .WORD    130

;OK GO ON
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;RESTORE VECTOR

;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;ODD ADDRESS TRAP
;
;INCREMENT TEST NUMBER
;CLEAR TRAP FLAG
;SAVE TIMEOUT VECTOR
;SETUP NEW VECTOR
;SETUP TO LOAD FPS
;SET FD=1
;INIT COUNTER
;SETUP TEST LOCATION POINTER
;INIT TEST LOCATION
;ARE WE DONE
;INIT TEST LOCATION
;***TEST INSTRUCTION***
;GET FPS
;SETUP POINTER TO DATA
;INIT COUNTER
;IS DATA CORRECT
;YES GO ON
;NO, GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;ARE WE DONE
;IS DATA CORRECT
;YES GO ON
;NO, GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;INIT COUNTER
;IS DATA CORRECT
;YES GO ON
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER

```

```

2231 010116 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
2232 ;NO GO TO ERROR
2233 010120 077107 5$: SOB R1,4$ ;ARE WE DONE
2234 010122 020327 000204 CMP R3,0204 ;CHECK FPS
2235 010126 001403 BEQ 6$ ;OK GO ON
2236 010130 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
2237 010132 000131 .WORD 131 ;UNIQUE ERROR NUMBER
2238 010134 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
2239 ;NO GO TO ERROR
2240 010136 012637 000004 6$: MOV (SP)+,004 ;RESTORE VECTOR
2241 ;
2242 ;
2243 010142 000167 000010 ; JMP FIN24
2244 ;
2245 010146 TSF24: ;
2246 010146 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
2247 010150 000132 .WORD 132 ;UNIQUE ERROR NUMBER
2248 010152 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
2249 ;ODD ADDRESS TRAP
2250 010154 000006 RTT ;
2251 ;
2252 010156 ;FIN24:
2253 ;
2254 010156 ;TSFP25:
2255 ;*****
2256 ;*TEST 27 TEST CLRF
2257 ;*****
2258 010156 TST27:
2259 010156 005267 170622 INC $TESTN ;INCREMENT TEST NUMBER
2260 010162 005037 002122 CLR 00TRPFLG ;CLEAR TRAP FLAG
2261 010166 005002 CLR R2 ;SETUP TO LOAD FPS
2262 010170 170102 LDFPS R2 ;SET FD=0
2263 010172 012705 000004 MOV 04,R5 ;INIT COUNTER
2264 010176 012704 001136 MOV 0TSTLOC,R4 ;SETUP POINTER TO TEST LOCATION
2265 010202 012724 177777 MOV 017777,(R4) ;INIT TEST LOCATION
2266 010206 077503 100$: SOB R5,100$ ;ARE WE DONE
2267 010210 012702 001136 MOV 0TSTLOC,R2 ;SETUP POINTER TO DATA
2268 010214 170422 CLRF (R2) ; TEST INSTRUCTION
2269 010216 170203 STFPS R3 ;GET FPS
2270 010220 020227 001142 CMP R2,0TSTLOC+4 ;IS R2 CORRECT
2271 010224 001403 BEQ 1$ ;YES GO ON
2272 ;NO, GO TO ERROR
2273 010226 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
2274 010230 000133 .WORD 133 ;UNIQUE ERROR NUMBER
2275 010232 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
2276 010234 012702 001136 1$: MOV 0TSTLOC,R2 ;SETUP POINTER TO DATA
2277 010240 012701 000002 MOV 02,R1 ;INIT COUNTER
2278 010244 022227 000000 2$: CMP (R2)+,00 ;IS DATA CORRECT
2279 010250 001403 BEQ 3$ ;YES GO ON
2280 ;NO, GO TO ERROR
2281 010252 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
2282 010254 000134 .WORD 134 ;UNIQUE ERROR NUMBER
2283 010256 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
2284 010260 077107 3$: SOB R1,2$ ;ARE WE DONE
2285 010262 012701 000002 MOV 02,R1 ;INIT COUNTER
2286 010266 022227 177777 4$: CMP (R2)+,017777 ;IS DATA CORRECT

```



```

2287 010272 001403      BEQ      5$
2288
2289 010274 104000      ERROR
2290 010276 000135      .WORD    135
2291 010300 002013      .WORD    FPPERR
2292 010302 077107      5$:      SOB      R1,4$
2293 010304 020327 000004      CMP      R3,4$
2294 010310 001403      BEQ      6$
2295
2296 010312 104000      ERROR
2297 010314 000136      .WORD    136
2298 010316 002013      .WORD    FPPERR
2299 010320
2300
2301
2302 010320      :
2303      TSFP26:
2304      :*****
2305      :*TEST 30      TEST TSTF AND TSTD
2306      :*****
2307 010320 005267 170460      TST30:
2308 010324 005037 002122      INC      $TESTN
2309 010330 005004      CLR      @TRPFLG
2310 010332 170104      CLR      R4
2311 010334 170567 000300      LDFPS   R4
2312 010340 170203      TSTF    TS26D0
2313 010342 020327 000004      STFPS   R3
2314 010346 001403      CMP      R3,4$
2315      BEQ      1$
2316 010350 104000      ERROR
2317 010352 000137      .WORD    137
2318 010354 002013      .WORD    FPPERR
2319 010356 012704 010640      1$:      MOV      @TS26D0,R4
2320 010362 012702 010670      MOV      @TS26D3,R2
2321 010366 004767 000224      JSR      PC,CHEC26
2322 010372 170537 010650      TSTF    @TS26D1
2323 010376 170203      STFPS   R3
2324 010400 020327 000010      CMP      R3,10
2325 010404 001403      BEQ      2$
2326
2327 010406 104000      ERROR
2328 010410 000140      .WORD    140
2329 010412 002013      .WORD    FPPERR
2330 010414 012704 010650      2$:      MOV      @TS26D1,R4
2331 010420 012702 010700      MOV      @TS26D4,R2
2332 010424 004767 000166      JSR      PC,CHEC26
2333 010430 170567 000224      TSTF    TS26D2
2334 010434 170203      STFPS   R3
2335 010436 020327 000000      CMP      R3,0
2336 010442 001403      BEQ      3$
2337
2338 010444 104000      ERROR
2339 010446 000141      .WORD    141
2340 010450 002013      .WORD    FPPERR
2341 010452 012704 010660      3$:      MOV      @TS26D2,R4
2342 010456 012702 010710      MOV      @TS26D5,R2

;YES GO ON
;NO GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;ARE WE DONE
;CHECK FPS
;OK GO ON
;NO, GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE

;INCREMENT TEST NUMBER
;CLEAR TRAP FLAG
;SETUP TO LOAD FPS
;SET FD=0
;***TEST INSTRUCTION***
;GET FPS
;CHECK FPS
;OK GO ON
;NO, GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;SETUP POINTERS TO DATA
;
;CHECK IF DATA IS CORRECT
;***TEST INSTRUCTION***
;GET FPS
;CHECK FPS
;OK GO ON
;NO, GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;SETUP POINTERS TO DATA
;
;CHECK IF DATA IS CORRECT
;***TEST INSTRUCTION***
;GET FPS
;CHECK FPS
;OK GO ON
;NO, GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;SETUP POINTERS TO DATA
;

```

2343	010462	004767	000130		JSR	PC,CHEC26		;CHECK IF DATA IS CORRECT
2344	010466	012704	000200		MOV	#200,R4		;SETUP TO LOAD FPS
2345	010472	170104			LDFPS	R4		;SET FD=1
2346	010474	170537	010640		TSTD	#TS26D0		***TEST INSTRUCTION***
2347	010500	170203			STFPS	R3		;GET FPS
2348	010502	020327	000204		CMP	R3,#204		;CHECK FPS
2349	010506	001403			BEQ	4:		;OK GO ON
2350								;NO, GO TO ERROR
2351	010510	104000			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
2352	010512	000142			.WORD	142		;UNIQUE ERROR NUMBER
2353	010514	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
2354	010516	012704	010640	4:	MOV	#TS26D0,R4		;SETUP POINTERS TO DATA
2355	010522	012702	010670		MOV	#TS26D3,R2		
2356	010526	004767	000064		JSR	PC,CHEC26		;CHECK IF DATA IS CORRECT
2357	010532	170567	000112		TSTD	TS26D1		***TEST INSTRUCTION***
2358	010536	170203			STFPS	R3		;GET FPS
2359	010540	020327	000210		CMP	R3,#210		;CHECK FPS
2360	010544	001403			BEQ	5:		;OK GO ON
2361								;NO, GO TO ERROR
2362	010546	104000			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
2363	010550	000143			.WORD	143		;UNIQUE ERROR NUMBER
2364	010552	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
2365	010554	012704	010650	5:	MOV	#TS26D1,R4		;SETUP POINTERS TO DATA
2366	010560	012702	010700		MOV	#TS26D4,R2		
2367	010564	004767	000026		JSR	PC,CHEC26		;CHECK IF DATA IS CORRECT
2368	010570	170567	000064		TSTD	TS26D2		***TEST INSTRUCTION
2369	010574	170203			STFPS	R3		;GET FPS
2370	010576	020327	000200		CMP	R3,#200		;CHECK FPS
2371	010602	001403			BEQ	6:		;OK GO ON
2372	010604	104000			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
2373	010606	000144			.WORD	144		;UNIQUE ERROR NUMBER
2374	010610	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
2375								;NO GO TO ERROR
2376	010612			6:				
2377								
2378								
2379	010612	000167	000102		JMP	FIN26		
2380								
2381	010616	012701	000004		CHEC26:	MOV	#4,R1	;INIT COUNTER
2382	010622	022422		1:	CMP	(R4),.(R2).		;IS DATA CORRECT
2383	010624	001403			BEQ	2:		;YES GO ON
2384	010626	104000			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
2385	010630	000145			.WORD	145		;UNIQUE ERROR NUMBER
2386	010632	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
2387								;NO GO TO ERROR
2388	010634	077106		2:	SOB	R1,1:		;ARE WE DONE
2389	010636	000207			RTS	PC		;RETURN
2390								
2391	010640	000177			TS26D0:	.WORD	177	
2392	010642	177777				.WORD	177777	
2393	010644	177777				.WORD	177777	
2394	010646	177777				.WORD	177777	
2395	010650	177777			TS26D1:	.WORD	177777	
2396	010652	000000				.WORD	0	
2397	010654	000000				.WORD	0	
2398	010656	000000				.WORD	0	


```

2399 010660 077777
2400 010662 000000
2401 010664 000000
2402 010666 000000
2403 010670 000177
2404 010672 177777
2405 010674 177777
2406 010676 177777
2407 010700 177777
2408 010702 000000
2409 010704 000000
2410 010706 000000
2411 010710 077777
2412 010712 000000
2413 010714 000000
2414 010716 000000
2415 010720
2416
2417 010720
2418
2419
2420
2421 010720
2422 010720 005267 170060
2423 010724 005037 002122
2424 010730 005005
2425 010732 170105
2426 010734 012701 000014
2427 010740 012704 001136
2428 010744 012703 011204
2429 010750 012324
2430 010752 077102
2431 010754 012705 001136
2432 010760 170615
2433 010762 170203
2434 010764 020527 001136
2435 010770 001403
2436
2437 010772 104000
2438 010774 000146
2439 010776 002013
2440 011000 012702 011234
2441 011004 004767 000152
2442 011010 020327 000000
2443 011014 001403
2444
2445 011016 104000
2446 011020 000147
2447 011022 002013
2448 011024 012705 001146
2449 011030 170625
2450 011032 170203
2451 011034 020527 001152
2452 011040 001403
2453
2454 011042 104000

```

```

TS26D2: .WORD 77777
        .WORD 0
        .WORD 0
        .WORD 0
TS26D3: .WORD 177
        .WORD 177777
        .WORD 177777
        .WORD 177777
TS26D4: .WORD 177777
        .WORD 0
        .WORD 0
        .WORD 0
TS26D5: .WORD 77777
        .WORD 0
        .WORD 0
        .WORD 0

```

FIN26:

;TSFP27:

```

;*****
;TEST 31      TEST ABSF
;*****

```

TST31:

```

        INC      $TESTN          ;INCREMENT TEST NUMBER
        CLR      @TRPFLG        ;CLEAR TRAP FLAG
        CLR      R5             ;SETUP TO LOAD FPS
        LDFPS    R5             ;SET FD=0
        MOV      @12.,R1        ;INIT COUNTER
        MOV      @TSTLOC,R4     ;SETUP POINTER TO TEST LOCATION
        MOV      @TS27D0,R3     ;SETUP POINTER TO TEST VALUE
100$:   MOV      (R3)+,(R4)+    ;INIT TEST LOCATION
        SOB      R1,100$        ;ARE WE DONE
        MOV      @TSTLOC,R5     ;SETUP POINTER TO DATA
        ABSF     (R5)           ;***TEST INSTRUCTION***
        STFPS    R3            ;GET FPS
        CMP      R5,@TSTLOC     ;IS R5 CORRECT
        BEQ      1$            ;YES GO ON
                                ;NO, GO TO ERROR
                                ;ALL ERRORS TO TRAP TO EMT VECTOR
        .WORD    146            ;UNIQUE ERROR NUMBER
        .WORD    FPPERR         ;ADDRESS OF ERROR MESSAGE
1$:     MOV      @TS27D3,R2     ;SETUP POINTER TO DATA
        JSR      PC,CHEC27      ;CHECK IF DATA IS CORRECT
        CMP      R3,#0          ;CHECK FPS
        BEQ      2$            ;OK GO ON
                                ;NO, GO TO ERROR
                                ;ALL ERRORS TO TRAP TO EMT VECTOR
        .WORD    147            ;UNIQUE ERROR NUMBER
        .WORD    FPPERR         ;ADDRESS OF ERROR MESSAGE
2$:     MOV      @TSTLOC+10,R5  ;SETUP POINTER TO DATA
        ABSF     (R5)+          ;***TEST INSTRUCTION***
        STFPS    R3            ;GET FPS
        CMP      R5,@TSTLOC+14  ;IS R5 CORRECT
        BEQ      3$            ;YES GO ON
                                ;NO, GO TO ERROR
                                ;ALL ERRORS TO TRAP TO EMT VECTOR
        ERROR

```

2455	011044	000150			.WORD	150			
2456	011046	002013			.WORD	FPPERR			
2457	011050	012705	001146	3\$:	MOV	#TSTLOC+10,R5			
2458	011054	012702	011244		MOV	#TS27D4,R2			
2459	011060	004767	000076		JSR	PC,CHEC27			
2460	011064	020327	000000		CMP	R3,#0			
2461	011070	001403			BEQ	4\$			
2462									
2463	011072	104000			ERROR				
2464	011074	000151			.WORD	151			
2465	011076	002013			.WORD	FPPERR			
2466	011100	012705	001136	4\$:	MOV	#TSTLOC,R5			
2467	011104	170665	000020		ABSF	20(R5)			
2468	011110	170203			STFPS	R3			
2469	011112	020527	001136		CMP	R5,#TSTLOC			
2470	011116	001403			BEQ	5\$			
2471									
2472	011120	104000			ERROR				
2473	011122	000152			.WORD	152			
2474	011124	002013			.WORD	FPPERR			
2475	011126	012705	001156	5\$:	MOV	#TSTLOC+20,R5			
2476	011132	012702	011254		MOV	#TS27D5,R2			
2477	011136	004767	000020		JSR	PC,CHEC27			
2478	011142	020327	000004		CMP	R3,#4			
2479	011146	001403			BEQ	6\$			
2480									
2481	011150	104000			ERROR				
2482	011152	000153			.WORD	153			
2483	011154	002013			.WORD	FPPERR			
2484	011156			6\$:					
2485									
2486									
2487	011156	000167	000102		JMP	FIN27			
2488									
2489	011162	012701	000004		CHEC27: MOV	#4,R1			
2490	011166	022522		1\$:	CMP	(R5),.(R2).			
2491	011170	001403			BEQ	2\$			
2492									
2493	011172	104000			ERROR				
2494	011174	000154			.WORD	154			
2495	011176	002013			.WORD	FPPERR			
2496	011200	077106		2\$:	SOB	R1,1\$			
2497	011202	000207			RTS	PC			
2498									
2499	011204	177777			TS27D0: .WORD	177777			
2500	011206	177777			.WORD	177777			
2501	011210	177777			.WORD	177777			
2502	011212	177777			.WORD	177777			
2503	011214	000377			TS27D1: .WORD	377			
2504	011216	175436			.WORD	175436			
2505	011220	136477			.WORD	136477			
2506	011222	000001			.WORD	1			
2507	011224	000177			TS27D2: .WORD	177			
2508	011226	175436			.WORD	175436			
2509	011230	136477			.WORD	136477			
2510	011232	000001			.WORD	1			

```

;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;SETUP POINTER TO DATA
;
;CHECK IF DATA IS CORRECT
;CHECK FPS
;OK GO ON
;NO, GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;SETUP POINTER TO DATA
;***TEST INSTRUCTION***
;GET FPS
;IS R5 CORRECT
;YES GO ON
;NO, GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;SETUP POINTERS TO DATA
;
;CHECK IF DATA IS CORRECT
;CHECK FPS
;OK GO ON
;NO, GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;
;INIT COUNTER
;IS DATA CORRECT
;YES GO ON
;NO, GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;ARE WE DONE
;RETURN

```


2511 011234 077777
 2512 011236 177777
 2513 011240 177777
 2514 011242 177777
 2515 011244 000377
 2516 011246 175436
 2517 011250 136477
 2518 011252 000001
 2519 011254 000000
 2520 011256 000000
 2521 011260 136477
 2522 011262 000001
 2523 011264
 2524
 2525 011264
 2526
 2527
 2528
 2529 011264
 2530 011264 005267 167514
 2531 011270 005037 002122
 2532 011274 012705 000200
 2533 011300 170105
 2534 011302 012701 000014
 2535 011306 012704 001136
 2536 011312 012703 011552
 2537 011316 012324
 2538 011320 077102
 2539 011322 012705 001136
 2540 011326 170615
 2541 011330 170203
 2542 011332 020527 001136
 2543 011336 001403
 2544
 2545 011340 104000
 2546 011342 000155
 2547 011344 002013
 2548 011346 012702 011602
 2549 011352 004767 000152
 2550 011356 020327 000200
 2551 011362 001403
 2552
 2553 011364 104000
 2554 011366 000156
 2555 011370 002013
 2556 011372 012705 001146
 2557 011376 170625
 2558 011400 170203
 2559 011402 020527 001156
 2560 011406 001403
 2561
 2562 011410 104000
 2563 011412 000157
 2564 011414 002013
 2565 011416 012705 001146
 2566 011422 012702 011612

TS27D3: .WORD 77777
 .WORD 177777
 .WORD 177777
 .WORD 177777
 TS27D4: .WORD 377
 .WORD 175436
 .WORD 136477
 .WORD 1
 TS27D5: .WORD 0
 .WORD 0
 .WORD 136477
 .WORD 1
 FIN27:
 !
 TSFP30:
 ;*****
 ;*TEST 32 TEST ABSD
 ;*****
 TST32:
 INC #TESTN ;INCREMENT TEST NUMBER
 CLR #TRPFLG ;CLEAR TRAP FLAG
 MOV #200,R5 ;SETUP TO LOAD FPS
 LDFPS R5 ;SET FD=1
 MOV #12,R1 ;INIT COUNTER
 MOV #TSTLOC,R4 ;SETUP POINTER TO TEST LOCATION
 MOV #TS30D0,R3 ;SETUP POINTER TO TEST VALUE
 100\$: MOV (R3)+,(R4)+ ;INIT TEST LOCATION
 SOB R1,100\$;ARE WE DONE
 MOV #TSTLOC,R5 ;SETUP POINTER TO DATA
 ABSD (R5) ;***TEST INSTRUCTION***
 STFPS R3 ;GET FPS
 CMP R5,#TSTLOC ;IS R5 CORRECT
 BEQ 1\$;YES GO ON
 ;NO, GO TO ERROR
 ;ALL ERRORS TO TRAP TO EMT VECTOR
 .WORD 155 ;UNIQUE ERROR NUMBER
 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
 1\$: MOV #TS30D3,R2 ;SETUP POINTER TO DATA
 JSR PC,CHEC30 ;CHECK IF DATA IS CORRECT
 CMP R3,#200 ;CHECK FPS
 BEQ 2\$;OK GO ON
 ;NO, GO TO ERROR
 ;ALL ERRORS TO TRAP TO EMT VECTOR
 .WORD 156 ;UNIQUE ERROR NUMBER
 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
 2\$: MOV #TSTLOC+10,R5 ;SETUP POINTER TO DATA
 ABSD (R5)+ ;***TEST INSTRUCTION***
 STFPS R3 ;GET FPS
 CMP R5,#TSTLOC+20 ;IS R5 CORRECT
 BEQ 3\$;YES GO ON
 ;NO, GO TO ERROR
 ;ALL ERRORS TO TRAP TO EMT VECTOR
 .WORD 157 ;UNIQUE ERROR NUMBER
 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
 3\$: MOV #TSTLOC+10,R5 ;SETUP POINTERS TO DATA
 MOV #TS30D4,R2

;

2567	011426	004767	000076		JSR	PC,CHEC30			
2568	011432	020327	000200		CMP	R3,#200			
2569	011436	001403			BEQ	4:			
2570									
2571	011440	104000			ERROR				
2572	011442	000160			.WORD	160			
2573	011444	002013			.WORD	FPPERR			
2574	011446	012705	001136	4:	MOV	#TSTLOC,R5			
2575	011452	170665	000020		ABSD	20(R5)			
2576	011456	170203			STFPS	R3			
2577	011460	020527	001136		CMP	R5,#TSTLOC			
2578	011464	001403			BEQ	5:			
2579									
2580	011466	104000			ERROR				
2581	011470	000161			.WORD	161			
2582	011472	002013			.WORD	FPPERR			
2583	011474	012705	001156	5:	MOV	#TSTLOC+20,R5			
2584	011500	012702	011622		MOV	#TS30D5,R2			
2585	011504	004767	000020		JSR	PC,CHEC30			
2586	011510	020327	000204		CMP	R3,#204			
2587	011514	001403			BEQ	6:			
2588									
2589	011516	104000			ERROR				
2590	011520	000162			.WORD	162			
2591	011522	002013			.WORD	FPPERR			
2592	011524			6:					
2593									
2594	011524	000167	000102		JMP	FIN30			
2595									
2596	011530	012701	000004		CHEC30: MOV	#4,R1			
2597	011534	022522		1:	CMP	(R5),.(R2).			
2598	011536	001403			BEQ	2:			
2599									
2600	011540	104000			ERROR				
2601	011542	000163			.WORD	163			
2602	011544	002013			.WORD	FPPERR			
2603	011546	077106		2:	SOB	R1,1			
2604	011550	000207			RTS	PC			
2605									
2606	011552	177777			TS30D0: .WORD	177777			
2607	011554	177777			.WORD	177777			
2608	011556	177777			.WORD	177777			
2609	011560	177777			.WORD	177777			
2610	011562	000377			TS30D1: .WORD	377			
2611	011564	175436			.WORD	175436			
2612	011566	136477			.WORD	136477			
2613	011570	000001			.WORD	1			
2614	011572	000177			TS30D2: .WORD	177			
2615	011574	175436			.WORD	175436			
2616	011576	136477			.WORD	136477			
2617	011600	000001			.WORD	1			
2618	011602	077777			TS30D3: .WORD	77777			
2619	011604	177777			.WORD	177777			
2620	011606	177777			.WORD	177777			
2621	011610	177777			.WORD	177777			
2622	011612	000377			TS30D4: .WORD	377			

```

;CHECK IF DATA IS CORRECT
;CHECK FPS
;OK GO ON
;NO, GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;SETUP POINTER TO DATA
;***TEST INSTRUCTION***
;GET FPS
;IS R5 CORRECT
;YES GO ON
;NO, GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;SETUP POINTERS TO DATA
;
;CHECK IF DATA IS CORRECT
;CHECK FPS
;OK GO ON
;NO, GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;
;INIT COUNTER
;IS DATA CORRECT
;YES GO ON
;NO, GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;ARE WE DONE
;RETURN

```


Address	Offset	Value	Label	Instruction	Comment
2623	011614	175436		.WORD	175436
2624	011616	136477		.WORD	136477
2625	011620	000001		.WORD	1
2626	011622	000000	TS30D5:	.WORD	0
2627	011624	000000		.WORD	0
2628	011626	000000		.WORD	0
2629	011630	000000		.WORD	0
2630	011632		FIN30:		
2631					
2632	011632		TSFP31:		
2633					
2634					
2635					
2636	011632		TST33:		
2637	011632	005267		INC	\$TESTN ; INCREMENT TEST NUMBER
2638	011636	005037		CLR	\$TRPFLG ; CLEAR TRAP FLAG
2639	011642	013746		MOV	\$04, -(SP) ; SAVE TIMEOUT VECTOR
2640	011646	012737		MOV	\$TSF31, \$04 ; SETUP NEW VECTOR
2641	011654	012702		MOV	\$200, R2 ; SETUP TO LOAD FPS
2642	011660	170102		LDFPS	R2 ; SET FD=1
2643	011662	170527		TSTD	\$5 ; ***TEST INSTRUCTION***
2644	011666	000240		NOP	
2645	011670	000240		NOP	
2646	011672	000240		NOP	
2647	011674	170203		STFPS	R3 ; GET FPS
2648	011676	020327		CMP	R3, \$204 ; CHECK FPS
2649	011702	001403		BEQ	1\$; OK GO ON
2650	011704	104000		ERROR	ALL ERRORS TO TRAP TO EMT VECTOR
2651	011706	000164		.WORD	164 ; UNIQUE ERROR NUMBER
2652	011710	002013		.WORD	FPPERR ; ADDRESS OF ERROR MESSAGE
2653					
2654	011712	012702		MOV	\$TSD31+2, R2 ; NO GO TO ERROR
2655	011716	022227		CMP	(R2), \$5 ; SETUP POINTER TO DATA
2656	011722	001403		BEQ	2\$; IS DATA CORRECT
2657	011724	104000		ERROR	YES GO ON
2658	011726	000165		.WORD	165 ; ALL ERRORS TO TRAP TO EMT VECTOR
2659	011730	002013		.WORD	FPPERR ; UNIQUE ERROR NUMBER
2660					ADDRESS OF ERROR MESSAGE
2661	011732	012701		MOV	\$3, R1 ; NO GO TO ERROR
2662	011736	022227		CMP	(R2), \$240 ; INIT COUNTER
2663	011742	001403		BEQ	4\$; IS DATA CORRECT
2664	011744	104000		ERROR	YES GO ON
2665	011746	000166		.WORD	166 ; ALL ERRORS TO TRAP TO EMT VECTOR
2666	011750	002013		.WORD	FPPERR ; UNIQUE ERROR NUMBER
2667					ADDRESS OF ERROR MESSAGE
2668	011752	077107		SOB	R1, 3\$; NO GO TO ERROR
2669	011754	012637		MOV	(SP), \$04 ; ARE WE DONE
2670					RESTORE VECTOR
2671					
2672	011760	000167		JMP	FIN31
2673					
2674	011764		TSF31:		
2675	011764	104000		ERROR	ALL ERRORS TO TRAP TO EMT VECTOR
2676	011766	000167		.WORD	167 ; UNIQUE ERROR NUMBER
2677	011770	002013		.WORD	FPPERR ; ADDRESS OF ERROR MESSAGE
2678					ODD ADDRESS TRAP

2679	011772	000006		RTT		
2680						;RETURN
2681	011774			FIN31:		
2682						
2683	011774			SFP32:		
2684						
2685				TEST 34	TEST NEGF	
2686						
2687	011774			TST34:		
2688	011774	005267	167004	INC	TESTN	;INCREMENT TEST NUMBER
2689	012000	005037	002122	CLR	TRPFLG	;CLEAR TRAP FLAG
2690	012004	005005		CLR	R5	;SETUP TO LOAD FPS
2691	012006	170105		LDFPS	R5	;SET FD=0
2692	012010	012701	000014	MOV	12,R1	;INIT COUNTER
2693	012014	012704	001136	MOV	TSTLOC,R4	;SETUP POINTER TO TEST LOCATION
2694	012020	012703	012210	MOV	TS32D0,R3	;SETUP POINTER TO TEST VALUE
2695	012024	012324		MOV	(R3),(R4)	;INIT TEST LOCATION
2696	012026	077102		SOB	R1,100	;ARE WE DONE
2697	012030	170767	167102	NEGF	TSTLOC	***TEST INSTRUCTION***
2698	012034	170203		STFPS	R3	;GET FPS
2699	012036	012705	001136	MOV	TSTLOC,R5	;SETUP POINTERS TO DATA
2700	012042	012702	012240	MOV	TS32D3,R2	
2701	012046	004767	000114	JSR	PC,CHEC32	;CHECK IF DATA IS CORRECT
2702	012052	020327	000000	CMP	R3,0	;CHECK FPS
2703	012056	001403		BEQ	1	;YES GO ON
2704						;NO, GO TO ERROR
2705	012060	104000		ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
2706	012062	000170		.WORD	170	;UNIQUE ERROR NUMBER
2707	012064	002013		.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
2708	012066	170767	167054	NEGF	TSTLOC*10	***TEST INSTRUCTION***
2709	012072	170203		STFPS	R3	;GET FPS
2710	012074	012705	001146	MOV	TSTLOC*10,R5	;SETUP POINTERS TO DATA
2711	012100	012702	012250	MOV	TS32D4,R2	
2712	012104	004767	000056	JSR	PC,CHEC32	;CHECK IF DATA IS CORRECT
2713	012110	020327	000010	CMP	R3,010	;CHECK FPS
2714	012114	001403		BEQ	2	;OK GO ON
2715						;NO, GO TO ERROR
2716	012116	104000		ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
2717	012120	000171		.WORD	171	;UNIQUE ERROR NUMBER
2718	012122	002013		.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
2719	012124	170767	167026	NEGF	TSTLOC*20	***TEST INSTRUCTION***
2720	012130	170203		STFPS	R3	;GET FPS
2721	012132	012705	001156	MOV	TSTLOC*20,R5	;SETUP POINTERS TO DATA
2722	012136	012702	012260	MOV	TS32D5,R2	
2723	012142	004767	000020	JSR	PC,CHEC32	;CHECK IF DATA IS CORRECT
2724	012146	020327	000004	CMP	R3,04	;CHECK FPS
2725	012152	001403		BEQ	3	;OK GO ON
2726						;NO, GO TO ERROR
2727	012154	104000		ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
2728	012156	000172		.WORD	172	;UNIQUE ERROR NUMBER
2729	012160	002013		.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
2730	012162					
2731						
2732	012162	000167	000102	JMP	FIN32	
2733						
2734	012166	012701	000004	CHEC32: MOV	04,R1	;INIT COUNTER


```

1$:      CMP      (R5)+,(R2)+
        BEQ      2$

        ERROR
        .WORD    173
        .WORD    FPPERR
2$:      SOB      R1,1$
        RTS      PC

;
TS32D0: .WORD    170000
        .WORD    3541
        .WORD    177777
        .WORD    172710
TS32D1: .WORD    70000
        .WORD    3541
        .WORD    177777
        .WORD    172710
        .WORD    177
        .WORD    100000
        .WORD    177777
        .WORD    177007
TS32D3: .WORD    70000
        .WORD    3541
        .WORD    177777
        .WORD    172710
TS32D4: .WORD    170000
        .WORD    3541
        .WORD    177777
        .WORD    172710
TS32D5: .WORD    0
        .WORD    0
        .WORD    177777
        .WORD    177007

FIN32:
;
TSFP33:
;*****
;*TEST 35          TEST NEGO
;*****
TST35:
        INC      $TESTN
        CLR      @TRPFLG
        MOV      @200,R5
        LDFPS    R5
        MOV      @12.,R1
        MOV      @TSTLOC,R4
        MOV      @TS33D0,R3
100$:   MOV      (R3)+,(R4)+
        SOB      R1,100$
        NEGD     TSTLOC
        STFPS    R3
        MOV      @TSTLOC,R5
        MOV      @TS33D3,R2
        JSR      PC,CHEC33
        CMP      R3,@200
        BEQ      1$

```

```

;IS DATA CORRECT
;YES GO ON
;NO, GO TO ERROR
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;ARE WE DONE
;RETURN

```

```

;INCREMENT TEST NUMBER
;CLEAR TRAP FLAG
;SETUP TO LOAD FPS
;SET FD=1
;INIT COUNTER
;SETUP POINTER TO TEST LOCATION
;SETUP POINTER TO TEST VALUE
;INIT TEST LOCATION
;ARE WE DONE
;***TEST INSTRUCTION***
;GET FPS
;SETUP POINTERS TO DATA
;
;CHECK IF DATA IS CORRECT
;CHECK FPS
;OK GO ON

```

Address	Hex	Dec	Label	Op	Opnd	Comment
2791	012356	104000		ERROR		;NO, GO TO ERROR
2792	012360	000174		.WORD	174	;ALL ERRORS TO TRAP TO EMT VECTOR
2793	012362	002013		.WORD	FPPERR	;UNIQUE ERROR NUMBER
2794	012364	170767	166556	1\$:	TSTLOC+10	;ADDRESS OF ERROR MESSAGE
2795	012370	170203		NEGD	R3	;***TEST INSTRUCTION***
2796	012372	012705	001146	STFPS		;GET FPS
2797	012376	012702	012546	MOV	#TSTLOC+10,R5	;SETUP POINTERS TO DATA
2798	012402	004767	000056	MOV	#TS33D4,R2	
2799	012406	020327	000210	JSR	PC,CHEC33	;CHECK IF DATA IS CORRECT
2800	012412	001403		CMP	R3,#210	;CHECK FPS
2801				BEQ	2\$	;OK GO ON
2802						;NO, GO TO ERROR
2803	012414	104000		ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
2804	012416	000175		.WORD	175	;UNIQUE ERROR NUMBER
2805	012420	002013	166530	2\$:	FPPERR	;ADDRESS OF ERROR MESSAGE
2806	012422	170767		NEGD	TSTLOC+20	;***TEST INSTRUCTION***
2807	012426	170203	001156	STFPS	R3	;GET FPS
2808	012430	012705	012556	MOV	#TSTLOC+20,R5	;SETUP POINTERS TO DATA
2809	012434	012702	000020	MOV	#TS33D5,R2	
2810	012440	004767	000204	JSR	PC,CHEC33	;CHECK IF DATA IS CORRECT
2811	012444	020327		CMP	R3,#204	;CHECK FPS
2812	012450	001403		BEQ	3\$	;OK GO ON
2813						;NO, GO TO ERROR
2814	012452	104000		ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
2815	012454	000176		.WORD	176	;UNIQUE ERROR NUMBER
2816	012456	002013		.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
2817	012460			3\$:		
2818						
2819	012460	000167	000102	JMP	FIN33	
2820						
2821	012464	012701	000004	CHEC33:	MOV #4,R1	;INIT COUNTER
2822	012470	022522		1\$:	CMP (R5),.(R2).	;IS DATA CORRECT
2823	012472	001403			BEQ 2\$	;YES GO ON
2824						;NO, GO TO ERROR
2825	012474	104000		ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
2826	012476	000177		.WORD	177	;UNIQUE ERROR NUMBER
2827	012500	002013		.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
2828	012502	077106		2\$:	SOB R1,1\$	;ARE WE DONE
2829	012504	000207		RTS	PC	;RETURN
2830						
2831	012506	170000		TS33D0:	.WORD 170000	
2832	012510	003541			.WORD 3541	
2833	012512	177777			.WORD 177777	
2834	012514	172710			.WORD 172710	
2835	012516	070000		TS33D1:	.WORD 70000	
2836	012520	003541			.WORD 3541	
2837	012522	177777			.WORD 177777	
2838	012524	172710			.WORD 172710	
2839	012526	000177		TS33D2:	.WORD 177	
2840	012530	100000			.WORD 100000	
2841	012532	177777			.WORD 177777	
2842	012534	177007			.WORD 177007	
2843	012536	070000		TS33D3:	.WORD 70000	
2844	012540	003541			.WORD 3541	
2845	012542	177777			.WORD 177777	
2846	012544	172710			.WORD 172710	

2847 012546 170000
 2848 012550 003541
 2849 012552 177777
 2850 012554 172710
 2851 012556 000000
 2852 012560 000000
 2853 012562 000000
 2854 012564 000000
 2855 012566
 2856
 2857 012566
 2858
 2859
 2860
 2861 012566
 2862 012566 005267 166212
 2863 012572 012704 047600
 2864 012576 170104
 2865 012600 012702 001106
 2866 012604 172407
 2867
 2868 012606 170201
 2869 012610 022701 147600
 2870 012614 001403
 2871 012616 104000
 2872 012620 000200
 2873 012622 002013
 2874
 2875 012624 170312
 2876 012626 022722 000002
 2877 012632 001403
 2878 012634 104000
 2879 012636 000201
 2880 012640 002013
 2881
 2882 012642 022722 012604
 2883 012646 001403
 2884 012650 104000
 2885 012652 000202
 2886 012654 002013
 2887
 2888 012656
 2889
 2890
 2891 012656
 2892
 2893
 2894
 2895 012656
 2896 012656 005267 166122
 2897 012662 012701 001126
 2898 012666 012704 001176
 2899 012672 012702 047750
 2900 012676 170102
 2901 012700 172424
 2902 012702 170203

TS33D4: .WORD 170000
 .WORD 3541
 .WORD 177777
 .WORD 172710
 TS33D5: .WORD 0
 .WORD 0
 .WORD 0
 .WORD 0

FIN33:

MFSRCMO:

 :*TEST 36 TEST LDD MODE 0, ILLEGAL AC7

 TST36:

1\$: INC \$TESTN
 MOV #47600,R4
 LDFPS R4
 MOV #RECFEC,R2
 LDD R7,AC0

2\$: STFPS R1
 CMP #147600,R1
 BEQ 2\$
 ERROR
 .WORD 200
 .WORD FPPERR

3\$: STST (R2)
 CMP #2,(R2)+
 BEQ 3\$
 ERROR
 .WORD 201
 .WORD FPPERR

4\$: CMP #1,(R2)+
 BEQ 4\$
 ERROR
 .WORD 202
 .WORD FPPERR

MLDDM2:

 :*TEST 37 TEST LDD MODE2

 TST37:

INC \$TESTN
 MOV #RECDST,R1
 MOV #TAB1,R4
 MOV #47750,R2
 LDFPS R2
 LDD (R4)+,AC0
 STFPS R3

;INCREMENT TEST NUMBER
 ;SETUP FPP STATUS
 ;LOAD FP STATUS
 ;POINT TO RECEIVED FEC MEMORY
 ;*TEST INSTRUCTION
 ;LOAD AC0 FROM ILLEGAL AC7
 ;SAVE FPP STATUS
 ;VERIFY FER BIT SET
 ;BRANCH IF GOOD ERROR CONDITION
 ;ALL ERRORS TO TRAP TO EMT VECTOR
 ;UNIQUE ERROR NUMBER
 ;ADDRESS OF ERROR MESSAGE
 ;THE FER BIT DIDNT SET
 ;SAVE FEC AND FEA
 ;VERIFY FEC CONTENTS
 ;BRANCH IF GOOD
 ;ALL ERRORS TO TRAP TO EMT VECTOR
 ;UNIQUE ERROR NUMBER
 ;ADDRESS OF ERROR MESSAGE
 ;FEC NE 2 (OPCODE ERROR)
 ;VERFIY FEA CONTENTS
 ;BRANCH FI GOOD
 ;ALL ERRORS TO TRAP TO EMT VECTOR
 ;UNIQUE ERROR NUMBER
 ;ADDRESS OF ERROR MESSAGE
 ;FEA NOT CORRECT ERROR ADDRESS

2903	012704	174021			STD	AC0,(R1),		;SAVE TEST RESULT MODE 2
2904	012706	020203			CMP	R2,R3		;VERIFY FPP STATUS
2905	012710	001403			BEQ	1		;BRANCH IF GOOD
2906	012712	104000			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
2907	012714	000203			.WORD	203		;UNIQUE ERROR NUMBER
2908	012716	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
2909								;BAD FPP STSTATUS
2910	012720	022704	001206	1:	CMP	#TAB1+10,R4		;VERIFY AUTO-INCR
2911	012724	001403			BEQ	2		;BRANCH IF GOOD
2912	012726	104000			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
2913	012730	000204			.WORD	204		;UNIQUE ERROR NUMBER
2914	012732	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
2915								;BAD AUTO-INCR
2916	012734	012704	001176	2:	MOV	#TAB1,R4		;POINT TO RECEIVED DATA
2917	012740	162701	000010		SUB	#10,R1		;RETURN R1 TO PROPER VALUE
2918	012744	004767	167176		JSR	R7,DATVER		;VERIFY DATA FROM FPP
2919	012750	005767	166064		TST	COUNT		;SEE IF COUNTER=0
2920	012754	001403			BEQ	3		;BRANCH IF GOOD COMPARE
2921								;BAD DATA FROM FPP
2922	012756	104000			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
2923	012760	000205			.WORD	205		;UNIQUE ERROR NUMBER
2924	012762	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
2925	012764			3:				
2926								
2927								
2928	012764							
2929								
2930								
2931								
2932	012764							
2933	012764	005267	166014					
2934	012770	012737	001126	001140	INC	#TESTN		;INCREMENT TEST NUMBER
2935	012776	012701	001140		MOV	#RECDST,#TSTLOC+2		;POINT TO RECEIVED DATA LOCATION
2936	013002	012737	001206	001136	MOV	#TSTLOC+2,R1		;SETUP STD IN MODE 3
2937	013010	012704	001136		MOV	#TAB2,#TSTLOC		;POINT TO DATA TABLE
2938	013014	012702	047750		MOV	#TSTLOC,R4		;POINT TO GOOD DATA
2939	013020	170102			MOV	#47750,R2		;LOAD GOOD STATUS
2940	013022	172434			LDFPS	R2		;LOAD FPP STATUS - DOUBLE.ID
2941	013024	170203			LDD	B(R4),AC0		;***TEST INSTRUCTION - MODE 2***
2942	013026	174031			STFPS	R3		;SAVE TEST FPP STATUS
2943	013030	022703	047740		STD	AC0,B(R1),		;SAVE TEST RESULT IN MODE 3
2944	013034	001403			CMP	#47740,R3		;VERIFY FPP STATUS
2945	013036	104000			BEQ	1		;BRANCH IF GOOD
2946	013040	000206			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
2947	013042	002013			.WORD	206		;UNIQUE ERROR NUMBER
2948					.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
2949	013044	022704	001140	1:	CMP	#TSTLOC+2,R4		;BAD FPP STATUS
2950	013050	001403			BEQ	2		;VERIFY AUTO-INCR
2951	013052	104000			ERROR			;BAD AUTO-DEC ON LDD
2952	013054	000207			.WORD	207		;ALL ERRORS TO TRAP TO EMT VECTOR
2953	013056	002013			.WORD	FPPERR		;UNIQUE ERROR NUMBER
2954								;ADDRESS OF ERROR MESSAGE
2955	013060	022701	001142	2:	CMP	#TSTLOC+4,R1		;BAD AUTO-INCR
2956	013064	001403			BEQ	3		;TEST STD AUTO-INCR
2957	013066	104000			ERROR			;BRANCH IF GOOD
2958	013070	000210			.WORD	210		;ALL ERRORS TO TRAP TO EMT VECTOR
								;UNIQUE ERROR NUMBER

MLDDM3:

TEST 40

TEST LDD MODE 3

TST40:

INC #TESTN

MOV #RECDST,#TSTLOC+2

MOV #TSTLOC+2,R1

MOV #TAB2,#TSTLOC

MOV #TSTLOC,R4

MOV #47750,R2

LDFPS R2

LDD B(R4),AC0

STFPS R3

STD AC0,B(R1),

CMP #47740,R3

BEQ 1

ERROR

.WORD 206

.WORD FPPERR

1: CMP #TSTLOC+2,R4

BEQ 2

ERROR

.WORD 207

.WORD FPPERR

2: CMP #TSTLOC+4,R1

BEQ 3

ERROR

.WORD 210

2959	013072	002013		.WORD	FPPERR		; ADDRESS OF ERROR MESSAGE
2960							; BAD AUTO-INCR
2961	013074	012704	001206	3\$:	MOV	@TAB2,R4	; POINT TO RECEIVED DATA
2962	013100	012701	001126		MOV	@RECDST,R1	; POINT TO RECEIVED DATA
2963	013104	004767	167036		JSR	R7,DATVER	; VERFIY DATA FROM FPP
2964	013110	005767	165724		TST	COUNT	; SEE IF COUNTER=0
2965	013114	001403			BEQ	4\$	; BRANCH IF GOOD COMPARE
2966	013116	104000			ERROR		; ALL ERRORS TO TRAP TO EMT VECTOR
2967	013120	000211			.WORD	211	; UNIQUE ERROR NUMBER
2968	013122	002013			.WORD	FPPERR	; ADDRESS OF ERROR MESSAGE
2969							; BAD DATA FROM FPP
2970	013124			4\$:			
2971							
2972	013124			MLDDM4:			
2973				;;*****			
2974				; *TEST 41	TEST LDF, STD MODE 4		
2975				;;*****			
2976	013124			TST41:			
2977	013124	005267	165654		INC	\$TESTN	; INCREMENT TEST NUMBER
2978	013130	012701	001132		MOV	@RECDST+4,R1	; POINT TO RECEIVED DATA LOCATION
2979	013134	012704	001222		MOV	@TAB3+4,R4	; POINT TO GOOD DATA
2980	013140	012705	001256		MOV	@TAB6,R5	; CLEAR OUT ACO
2981	013144	170127	000200		LDFPS	@200	; SET TO DOUBLE
2982	013150	172415			LDD	(R5),ACO	; ACO=0
2983	013152	012702	047550		MOV	@47550,R2	; LOAD GOOD STATUS FLOATING
2984	013156	170102			LDFPS	R2	; LOAD FPP STATUS - DOUBLE, ID
2985	013160	172444			LDF	-(R4),ACO	; *TEST INSTRUCTION - MODE 4
2986	013162	170203			STFPS	R3	; SAVE TEST FPP STATUS
2987	013164	012702	047750		MOV	@47750,R2	; SET TO DOUBLE MODE
2988	013170	170102			LDFPS	R2	; SET FPP TO DOUBLE
2989	013172	174041			STD	ACO, -(R1)	; SAVE TEST RESULT
2990	013174	022703	047540		CHP	@47540,R3	; VERIFY FPP STATUS
2991	013200	001403			BEQ	1\$	; BRANCH IF GOOD
2992	013202	104000			ERROR		; ALL ERRORS TO TRAP TO EMT VECTOR
2993	013204	000212			.WORD	212	; UNIQUE ERROR NUMBER
2994	013206	002013			.WORD	FPPERR	; ADDRESS OF ERROR MESSAGE
2995							; BAD FPP STATUS
2996	013210	022704	001216	1\$:	CHP	@TAB3,R4	; VERFIY AUTO-DEC
2997	013214	001403			BEQ	2\$	; BRANCH IF GOOD
2998	013216	104000			ERROR		; ALL ERRORS TO TRAP TO EMT VECTOR
2999	013220	000213			.WORD	213	; UNIQUE ERROR NUMBER
3000	013222	002013			.WORD	FPPERR	; ADDRESS OF ERROR MESSAGE
3001							; BAD AUTO-INCR
3002	013224	012704	001216	2\$:	MOV	@TAB3,R4	; POINT TO RECEIVED DATA
3003	013230	004767	166712		JSR	R7,DATVER	; VERFIY DATA FROM FPP
3004	013234	005767	165600		TST	COUNT	; SEE IF COUNTER=0
3005	013240	001403			BEQ	3\$	; BRANCH IF GOOD COMPARE
3006	013242	104000			ERROR		; ALL ERRORS TO TRAP TO EMT VECTOR
3007	013244	000214			.WORD	214	; UNIQUE ERROR NUMBER
3008	013246	002013			.WORD	FPPERR	; ADDRESS OF ERROR MESSAGE
3009							; BAD DATA FROM FPP
3010	013250			3\$:			
3011							
3012	013250			MLDDM5:			
3013				;;*****			
3014				; *TEST 42	TEST LDD MODE 5		

```

3015
3016 013250
3017 013250 005267 165530
3018 013254 012701 001126
3019 013260 012704 001140
3020 013264 012737 001176 001136
3021 013272 012702 047750
3022 013276 170102
3023 013300 172454
3024 013302 170203
3025 013304 174011
3026 013306 020203
3027 013310 001403
3028 013312 104000
3029 013314 000215
3030 013316 002013
3031
3032 013320 022704 001136
3033 013324 001403
3034 013326 104000
3035 013330 000216
3036 013332 002013
3037
3038 013334 012704 001176
3039 013340 004767 166602
3040 013344 005767 165470
3041 013350 001403
3042 013352 104000
3043 013354 000217
3044 013356 002013
3045
3046 013360
3047
3048 013360
3049
3050
3051
3052 013360
3053 013360 005267 165420
3054 013364 012701 001326
3055 013370 012704 001006
3056 013374 012702 047750
3057 013400 170102
3058 013402 172464 000200
3059 013406 170203
3060 013410 174061 177600
3061 013414 022703 047740
3062 013420 001403
3063 013422 104000
3064 013424 000220
3065 013426 002013
3066
3067 013430 162701 000200
3068 013434 062704 000200
3069 013440 004767 166502
3070 013444 005767 165370
  
```

```

*****
TST42:
      INC      $TESTN          ; INCREMENT TEST NUMBER
      MOV      @RECDST,R1      ; POINT TO RECEIVED DATA LOCATION
      MOV      @TSTLOC+2,R4    ; POINT TO GOOD DATA
      MOV      @TAB1,@TSTLOC   ; SET UP MODE 5 POINTER TO DATA
      MOV      @47750,R2       ; LOAD GOOD STATUS
      LDFPS    R2              ; LOAD FPP STATUS - DOUBLE.ID
      LDD      @-(R4),AC0      ; *TEST INSTRUCTION - MODE 5
      STFPS    R3              ; SAVE TEST FPP STATUS
      STD      AC0,(R1)        ; SAVE TEST RESULT
      CMP      R2,R3           ; VERIFY FPP STATUS
      BEQ      1$              ; BRANCH IF GOOD
      ERROR    215             ; ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    FPPERR         ; UNIQUE ERROR NUMBER
      .WORD    FPPERR         ; ADDRESS OF ERROR MESSAGE
      .WORD    FPPERR         ; BAD FPP STATUS
      .WORD    FPPERR         ; VERFIY AUTO-DEC
      .WORD    FPPERR         ; BRANCH IF GOOD
      .WORD    FPPERR         ; ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    FPPERR         ; UNIQUE ERROR NUMBER
      .WORD    FPPERR         ; ADDRESS OF ERROR MESSAGE
      .WORD    FPPERR         ; BAD AUTO-DEC
      .WORD    FPPERR         ; POINT TO EXPECTED DATA
      .WORD    FPPERR         ; VERFIY DATA FROM FPP
      .WORD    FPPERR         ; SEE IF COUNTER=0
      .WORD    FPPERR         ; BRANCH IF GOOD COMPARE
      .WORD    FPPERR         ; ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    FPPERR         ; UNIQUE ERROR NUMBER
      .WORD    FPPERR         ; ADDRESS OF ERROR MESSAGE
      .WORD    FPPERR         ; BAD DATA FROM FPP

1$:
      CMP      @TSTLOC,R4
      BEQ      2$
      ERROR    216
      .WORD    FPPERR
      .WORD    FPPERR

2$:
      MOV      @TAB1,R4
      JSR      R7,DATVER
      TST      COUNT
      BEQ      3$
      ERROR    217
      .WORD    FPPERR
      .WORD    FPPERR

3$:
  
```

MLDDM6:

```

*****
; *TEST 43      TEST LDD MODE 6
*****
TST43:
      INC      $TESTN          ; INCREMENT TEST NUMBER
      MOV      @RECDST+200,R1  ; POINT TO RECEIVED DATA LOCATION
      MOV      @TAB2-200,R4    ; SETUP R4 FOR MODE 6
      MOV      @47750,R2       ; LOAD GOOD STATUS
      LDFPS    R2              ; LOAD FPP STATUS - DOUBLE.ID
      LDD      200(R4),AC0     ; LDD MODE 6
      STFPS    R3              ; SAVE TEST FPP STATUS
      STD      AC0,-200(R1)    ; SAVE TEST RESULT
      CMP      @47740,R3       ; VERIFY FPP STATUS
      BEQ      1$              ; BRANCH IF GOOD
      ERROR    220             ; ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD    FPPERR         ; UNIQUE ERROR NUMBER
      .WORD    FPPERR         ; ADDRESS OF ERROR MESSAGE
      .WORD    FPPERR         ; BAD FPP STATUS
      .WORD    FPPERR         ; R1=RECDST
      .WORD    FPPERR         ; POINT TO EXPECTED DATA
      .WORD    FPPERR         ; VERFIY DATA FROM FPP
      .WORD    FPPERR         ; SEE IF COUNTER=0

1$:
      SUB      @200,R1
2$:
      ADD      @200,R4
      JSR      R7,DATVER
      TST      COUNT
  
```


3071	013450	001403		BEQ	3:				
3072	013452	104000		ERROR					
3073	013454	000221		.WORD	221				
3074	013456	002013		.WORD	FPPERR				
3075									
3076	013460								
3077									
3078	013460								
3079									
3080									
3081									
3082	013460								
3083	013460	005267	165320						
3084	013464	012701	001126						
3085	013470	005004							
3086	013472	012727	001176			001136			
3087	013500	012702	047750						
3088	013504	170102							
3089	013506	172474	001136						
3090	013512	170203							
3091	013514	174011							
3092	013516	020203							
3093	013520	001403							
3094	013522	104000							
3095	013524	000222							
3096	013526	002013							
3097									
3098	013530	005704							
3099	013532	001403							
3100	013534	104000							
3101	013536	000223							
3102	013540	002013							
3103									
3104	013542	012704	001176						
3105	013546	004767	166374						
3106	013552	005767	165262						
3107	013556	001403							
3108	013560	104000							
3109	013562	000224							
3110	013564	002013							
3111									
3112	013566								
3113									
3114	013566								
3115									
3116									
3117									
3118	013566								
3119	013566	005267	165212						
3120	013572	012701	001126						
3121	013576	012704	001236						
3122	013602	012702	047750						
3123	013606	005005							
3124	013610	170102							
3125	013612	172427	043243						
3126	013616	005205							

3:									
MLDDM7:									

*TEST 44									

TST44:									
INC	\$TESTN								
MOV	\$RECDST,R1								
CLR	R4								
MOV	\$TAB1,\$TSTLOC								
MOV	\$47750,R2								
LDFPS	R2								
LDD	\$TSTLOC(R4),AC0								
STFPS	R3								
STD	AC0,(R1)								
CMP	R2,R3								
BEQ	1:								
ERROR									
.WORD	222								
.WORD	FPPERR								
1:									
TST	R4								
BEQ	2:								
ERROR									
.WORD	223								
.WORD	FPPERR								
2:									
MOV	\$TAB1,R4								
JSR	R7,DATVER								
TST	COUNT								
BEQ	3:								
ERROR									
.WORD	224								
.WORD	FPPERR								
3:									
MLDM27:									

*TEST 45									

TST45:									
INC	\$TESTN								
MOV	\$RECDST,R1								
MOV	\$TAB5,R4								
MOV	\$47750,R2								
CLR	R5								
LDFPS	R2								
LDD	\$5205,AC0								
INC	R5								

L5

3127	013620	005205		INC	R5	
3128	013622	005205		INC	R5	
3129	013624	022705	000003	CMP	#3,R5	
3130	013630	001403		BEQ	1:	
3131	013632	104000		ERROR		
3132	013634	000225		.WORD	225	
3133	013636	002013		.WORD	FPPERR	
3134						
3135	013640	170203		18: STFPS	R3	
3136	013642	174011		STD	AC0,(R1)	
3137	013644	022703	047740	CMP	#47740,R3	
3138	013650	001403		BEQ	2:	
3139	013652	104000		ERROR		
3140	013654	000226		.WORD	226	
3141	013656	002013		.WORD	FPPERR	
3142						
3143	013660	004767	166262	29: JSR	R7,DATVER	
3144	013664	005767	165150	TST	COUNT	
3145	013670	001403		BEQ	3:	
3146	013672	104000		ERROR		
3147	013674	000227		.WORD	227	
3148	013676	002013		.WORD	FPPERR	
3149						
3150	013700			39:		
3151						
3152	013700					
3153						
3154						
3155						
3156	013700					
3157	013700	005267	165100			
3158	013704	012704	001256			
3159	013710	005067	165216			
3160	013714	005067	165214			
3161	013720	012702	040000			
3162	013724	170102				
3163	013726	172414				
3164	013730	172014				
3165	013732	170203				
3166	013734	022703	040004			
3167	013740	001403				
3168	013742	104000				
3169	013744	000230				
3170	013746	002013				
3171						
3172	013750	012701	001126	19: MOV	#RECDST,R1	
3173	013754	174011		STF	AC0,(R1)	
3174	013756	004767	166164	JSR	R7,DATVER	
3175	013762	005767	165052	TST	COUNT	
3176	013766	001403		BEQ	2:	
3177	013770	104000		ERROR		
3178	013772	000231		.WORD	231	
3179	013774	002013		.WORD	FPPERR	
3180						
3181	013776	012702	040200	29: MOV	#40200,R2	
3182	014002	170102		LDFPS	R2	

TEST 46 TEST ADDF, ADDD, SUBF, SUBD - AC0=0 FSRC=0;

TST46:

INC #TESTN ; INCREMENT TEST NUMBER

MOV #TAB6,R4 ; POINT TO FSRC TEST DATA

CLR RECDST+4 ; CLEAR OUT RECEIVED DATA TABLE

CLR RECDST+6 ;

MOV #40000,R2 ; SET UP GOOD STATUS

LDFPS R2 ; LOAD FPP STATUS, FLOATING

LDF (R4),AC0 ; LOAD AC0 WITH 0

ADDF (R4),AC0 ; 0+0

STFPS R3 ; SAVE STATUS

CMP #40004,R3 ; VERIFY STATUS

BEQ 1: ; BRANCH IF GOOD

ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR

.WORD 230 ; UNIQUE ERROR NUMBER

.WORD FPPERR ; ADDRESS OF ERROR MESSAGE

18: MOV #RECDST,R1 ; BAD FPP STATUS

STF AC0,(R1) ; POINT TO RECEIVED DATA

JSR R7,DATVER ; SAVE DATA

TST COUNT ; VERIFY DATA

BEQ 2: ; BRANCH IF GOOD

ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR

.WORD 231 ; UNIQUE ERROR NUMBER

.WORD FPPERR ; ADDRESS OF ERROR MESSAGE

29: MOV #40200,R2 ; BAD DATA IN AC0

LDFPS R2 ; LOAD FLOATING STATUS

3183	014004	172414			LDD	(R4),ACO		;LOAD ACO WITH 0
3184	014006	172014			ADDD	(R4),ACO		;*TEST INSTRUCTION
3185	014010	174011			STD	ACO,(R1)		;SAVE DATA
3186	014012	170203			STFPS	R3		;SAVE FPS
3187	014014	022703	040204		CMP	#40204,R3		;VERFIY STATUS
3188	014020	001403			BEQ	3:		;BRANCH IF GOOD
3189	014022	104000			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
3190	014024	000232			.WORD	232		;UNIQUE ERROR NUMBER
3191	014026	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
3192								
3193	014030	004767	166112	3:	JSR	R7,DATVER		;BAD FPS
3194	014034	005737	001040		TST	#COUNT		;VERFIY DATA
3195	014040	001403			BEQ	44:		;VERIFY RESULT
3196	014042	104000			ERROR			;BRANCH IF GOOD
3197	014044	000233			.WORD	233		;ALL ERRORS TO TRAP TO EMT VECTOR
3198	014046	002013			.WORD	FPPERR		;UNIQUE ERROR NUMBER
3199								;ADDRESS OF ERROR MESSAGE
3200	014050	172414		44:	LDD	(R4),ACO		;BAD ACO
3201	014052	173014			SUBD	(R4),ACO		;SETUP DATA
3202	014054	170203			STFPS	R3		;*TEST INSTRUCTION
3203	014056	022703	040204		CMP	#40204,R3		;SAVE STATUS
3204	014062	001403			BEQ	4:		;VERFIY STATUS
3205	014064	104000			ERROR			;BRANCH IF GOOD
3206	014066	000234			.WORD	234		;ALL ERRORS TO TRAP TO EMT VECTOR
3207	014070	002013			.WORD	FPPERR		;UNIQUE ERROR NUMBER
3208								;ADDRESS OF ERROR MESSAGE
3209	014072	174011		4:	STD	ACO,(R1)		;BAD FPS
3210	014074	004767	166046		JSR	R7,DATVER		;SAVE ACO DATA
3211	014100	005767	164734		TST	COUNT		;VERFIY DATA
3212	014104	001403			BEQ	5:		
3213	014106	104000			ERROR			;BRANCH IF GOOD
3214	014110	000235			.WORD	235		;ALL ERRORS TO TRAP TO EMT VECTOR
3215	014112	002013			.WORD	FPPERR		;UNIQUE ERROR NUMBER
3216								;ADDRESS OF ERROR MESSAGE
3217	014114	170127	000000	5:	LDFPS	#0		;BAD ACO
3218	014120	172414			LDD	(R4),ACO		;STORE FPP STATUS
3219	014122	173014			SUBF	(R4),ACO		;LOAD ACO
3220	014124	170203			STFPS	R3		;O-0
3221	014126	174011			STD	ACO,(R1)		;SAVE STATUS
3222	014130	022703	000004		CMP	#4,R3		;SAVE ACO
3223	014134	001403			BEQ	6:		;VERFIY STATUS
3224	014136	104000			ERROR			;BRANCH IF GOOD
3225	014140	000236			.WORD	236		;ALL ERRORS TO TRAP TO EMT VECTOR
3226	014142	002013			.WORD	FPPERR		;UNIQUE ERROR NUMBER
3227								;ADDRESS OF ERROR MESSAGE
3228	014144	004767	165776	6:	JSR	R7,DATVER		;BAD FPS
3229	014150	005767	164664		TST	COUNT		;VERIFY DATAT
3230	014154	001403			BEQ	7:		
3231	014156	104000			ERROR			;BRANC IF GOOD
3232	014160	000237			.WORD	237		;ALL ERRORS TO TRAP TO EMT VECTOR
3233	014162	002013			.WORD	FPPERR		;UNIQUE ERROR NUMBER
3234								;ADDRESS OF ERROR MESSAGE
3235	014164			7:				;BAD ACO
3236								
3237								
3238	014164			MNNRM2;				

3239
3240
3241
3242 014164
3243 014164 005267 164614
3244 014170 012701 001126
3245 014174 012705 001256
3246 014200 012704 001206
3247 014204 170127 000200
3248 014210 172415
3249 014212 005002
3250 014214 170102
3251 014216 172414
3252 014220 172015
3253 014222 170203
3254 014224 174011
3255 014226 022703 000000
3256 014232 001403
3257 014234 104000
3258 014236 000240
3259 014240 002013
3260
3261 014242 012704 001226
3262 014246 004767 165674
3263 014252 005767 164562
3264 014256 001403
3265 014260 104000
3266 014262 000241
3267 014264 002013
3268
3269 014266 170127 000200
3270 014272 172414
3271 014274 173015
3272 014276 170203
3273 014300 174011
3274 014302 022703 000200
3275 014306 001403
3276 014310 104000
3277 014312 000242
3278 014314 002013
3279
3280 014316 012704 001226
3281 014322 004767 165620
3282 014326 005767 164506
3283 014332 001403
3284 014334 104000
3285 014336 000243
3286 014340 002013
3287
3288 014342
3289
3290
3291 014342
3292
3293
3294

```
*****
; *TEST 47          TEST ADDF,SUBD - FSRC=0, ACO NE 0
*****
TST47:
      INC          $TESTN          ; INCREMENT TEST NUMBER
      MOV          #RECDST,R1      ; POINT TO RECEIVED DATA TABLE
      MOV          #TAB6,R5        ; POINT TO SOURCE DATA TABLE
      MOV          #TAB2,R4        ; POINT TO ACO DATA
      LDFPS        #200            ; SET TO DOUBLE FOR CLEAR
      LDD          (R5),ACO        ;
      CLR          R2              ; SETUP FPP STATUS
      LDFPS        R2              ; LOAD FPS
      LDF          (R4),ACO        ; LOAD ACO
      ADDF         (R5),ACO        ; *TEST INSTRUCTION
      STFPS        R3              ; SAVE STATUS
      STF          ACO,(R1)        ; SAVE ACO
      CMP          #0,R3           ; VERFIY NEGATIVE RESULT
      BEQ          1$              ; BRANCH IF GOOD
      ERROR        240             ; ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD        FPPERR          ; UNIQUE ERROR NUMBER
      .WORD        FPPERR          ; ADDRESS OF ERROR MESSAGE
; BAD FPS
1$:   MOV          #TAB4,R4        ; POINT TO EXPECTED DATA
      JSR          R7,DATVER      ; VERFIY ACO
      TST          COUNT          ; CHECK RESULT
      BEQ          2$              ; BRANCH IF GOOD
      ERROR        241             ; ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD        FPPERR          ; UNIQUE ERROR NUMBER
      .WORD        FPPERR          ; ADDRESS OF ERROR MESSAGE
; BAD ACO
2$:   LDFPS        #200            ; SET STATUS TO DUOBLE NODE
      LDD          (R4),ACO        ; LOAD ACO WITH A VALUE
      SUBD         (R5),ACO        ; *TEST INSTRUCTION
      STFPS        R3              ; SAVE FPP STATUS
      STD          ACO,(R1)        ; SAVE ACO
      CMP          #200,R3         ; VERIFY RESULT
      BEQ          3$              ; BRANCH IF GOOD
      ERROR        242             ; ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD        FPPERR          ; UNIQUE ERROR NUMBER
      .WORD        FPPERR          ; ADDRESS OF ERROR MESSAGE
; BAD SUBD
3$:   MOV          #TAB4,R4        ; POINT TO EXPECTED
      JSR          R7,DATVER      ; VERFIY ACO
      TST          COUNT          ;
      BEQ          4$              ; BRANCH IF GOOD ACO
      ERROR        243             ; ALL ERRORS TO TRAP TO EMT VECTOR
      .WORD        FPPERR          ; UNIQUE ERROR NUMBER
      .WORD        FPPERR          ; ADDRESS OF ERROR MESSAGE
; BAD ACO
4$:
MNNRM3:
*****
; *TEST 50          TEST ADDD, SUBF - FSRC NE 0, ACO=0
*****
```



```

3295 014342
3296 014342 005267 164436
3297 014346 012701 001126
3298 014352 012705 001256
3299 014356 012704 001176
3300 014362 012702 000200
3301 014366 170102
3302 014370 172415
3303 014372 172014
3304 014374 170203
3305 014376 174011
3306 014400 022703 000210
3307 014404 001403
3308 014406 104000
3309 014410 000244
3310 014412 002013
3311
3312 014414 004767 165526
3313 014420 005767 164414
3314 014424 001403
3315 014426 104000
3316 014430 000245
3317 014432 002013
3318
3319 014434 170127 000200
3320 014440 172415
3321 014442 173014
3322 014444 170203
3323 014446 174011
3324 014450 022703 000200
3325 014454 001403
3326 014456 104000
3327 014460 000246
3328 014462 002013
3329
3330 014464 012704 001446
3331 014470 004767 165452
3332 014474 005767 164340
3333 014500 001403
3334 014502 104000
3335 014504 000247
3336 014506 002013
3337
3338 014510
3339
3340
3341 014510
3342
3343
3344
3345 014510
3346 014510 005267 164270
3347 014514 012702 003240
3348 014520 170102
3349 014522 012704 001276
3350 014526 012705 001306

TST50:
INC $TESTN ; INCREMENT TEST NUMBER
MOV $RECDST,R1 ; POINT TO RECEIVED DATA TABLE
MOV $TAB6,R5 ; POINT TO ACO DATA TABLE
MOV $TAB1,R4 ; POINT TO FSRC DATA
MOV $200,R2 ; SETUP FPP STATUS
LDFPS R2 ; LOAD FPS
LDD (R5),ACO ; LOAD ACO
ADD (R4),ACO ; *TEST INSTRUCTION
STFPS R3 ; SAVE STATUS
STD ACO,(R1) ; SAVE ACO
CMP $210,R3 ; VERIFY NEGATIVE RESULT
BEQ 1$ ; BRANCH IF GOOD
ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 244 ; UNIQUE ERROR NUMBER
.WORD FPPERR ; ADDRESS OF ERROR MESSAGE

1$: JSR R7,DATVER ; BAD FPS
TST COUNT ; VERIFY ACO
BEQ 2$ ; CHECK RESULT
ERROR ; BRANCH IF GOOD
.WORD 245 ; ALL ERRORS TO TRAP TO EMT VECTOR
.WORD FPPERR ; UNIQUE ERROR NUMBER
; ADDRESS OF ERROR MESSAGE

2$: LDFPS $200 ; BAD ACO
LDF (R5),ACO ; SET STATUS TO DOUBLE NODE
SUBD (R4),ACO ; LOAD ACO WITH A VALUE
STFPS R3 ; *TEST INSTRUCTION
STF ACO,(R1) ; SAVE FPP STATUS
CMP $200,R3 ; SAVE ACO
BEQ 3$ ; VERIFY RESULT
ERROR ; BRANCH IF GOOD
.WORD 246 ; ALL ERRORS TO TRAP TO EMT VECTOR
.WORD FPPERR ; UNIQUE ERROR NUMBER
; ADDRESS OF ERROR MESSAGE

3$: MOV $TAB18,R4 ; BAD SUBD
JSR R7,DATVER ; POINT TO EXPECTED DATA
TST COUNT ; VERIFY ACO
BEQ 4$ ; BRANCH IF GOOD ACO
ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 247 ; UNIQUE ERROR NUMBER
.WORD FPPERR ; ADDRESS OF ERROR MESSAGE

4$:

*****
; *TEST 51 TEST ADDF, SUBD - EXP(ACO) - EXP(FSRC)
*****
TST51:
INC $TESTN ; INCREMENT TEST NUMBER
MOV $3240,R2 ; SET FIU,FD,FT
LDFPS R2
MOV $TAB7,R4 ; SET FSRC
MOV $TAB8,R5 ; SETUP ACO

```

C6

```
3351 014532 012701 001126      MOV    #RECDST,R1      ;POINT TO RECEIVED DATA
3352 014536 172415              LDD     (R5),ACO        ;LOAD ACO
3353 014540 172014              ADDD    (R4),ACO        ;*TEST INSTRUCTION
3354 014542 174011              STD     ACO,(R1)        ;SAVE TEST RESULT
3355 014544 012704 001316      MOV    #TAB9,R4        ;POINT TO EXPECTED DATA
3356 014550 004767 165372      JSR     R7,DATVER      ;VERIFY ACO DATA
3357 014554 005767 164260      TST     COUNT        ;
3358 014560 001403              BEQ     1$            ;BRANCH IF GOOD
3359 014562 104000              ERROR    ;ALL ERRORS TO TRAP TO EMT VECTOR
3360 014564 000250              .WORD   250            ;UNIQUE ERROR NUMBER
3361 014566 002013              .WORD   FPPERR        ;ADDRESS OF ERROR MESSAGE
3362                                ;BAD ADDD
3363 014570 012704 001306      1$: MOV    #TAB8,R4        ;
3364 014574 012703 001306      MOV    #TAB8,R3        ;SETUP SAME ACO
3365 014600 012702 003200      MOV    #3200,R2       ;ROUND MODE
3366 014604 170102              LDFPS   R2            ;
3367 014606 172413              LDD     (R3),ACO        ;LOAD ACO
3368 014610 061400              ADD     (R4),ACO        ;*TEST INSTRUCTION
3369 014612 174011              STD     ACO,(R1)        ;SAVE DATA
3370 014614 004767 165326      JSR     R7,DATVER      ;VERIFY ACO
3371 014620 005767 164214      TST     COUNT        ;
3372 014624 001403              BEQ     2$            ;BRANCH IF GOOD
3373 014626 104000              ERROR    ;ALL ERRORS TO TRAP TO EMT VECTOR
3374 014630 000251              .WORD   251            ;UNIQUE ERROR NUMBER
3375 014632 002013              .WORD   FPPERR        ;ADDRESS OF ERROR MESSAGE
3376                                ;BAD ROUND RESULT
3377 014634
3378
3379
3380 014634
3381
3382
3383
3384 014634
3385 014634 005267 164144      MXDF1:
3386 014640 012702 003200      ;*****
3387 014644 170102              ;*TEST 52      TEST ADD - EXP(FSRC) .GT. EXP(ACO)
3388 014646 012704 001336      ;*****
3389 014652 012701 001126      TST52:
3390 014656 012705 001326      INC     #TESTN        ;INCREMENT TEST NUMBER
3391 014662 172415              MOV    #3200,R2       ;R2=FPP STATUS
3392 014664 172014              LDFPS   R2            ;LOAD FPS STATUS
3393 014666 170203              MOV    #TAB11,R4      ;POINT TO FSRC DATA
3394 014670 174011              MOV    #RECDST,R1    ;POINT TO ACO RESULT
3395 014672 022703 003200      MOV    #TAB10,R5     ;POINT TO ACO DATA
3396 014676 001403              LDD     (R5),ACO        ;LOAD ACO DATA
3397 014700 104000              ADDD    (R4),ACO        ;*TEST INSTRUCTIONS
3398 014702 000252              STFPS   R3            ;SAVE FPP STATUS
3399 014704 002013              STD     ACO,(R1)    ;SAVE ACO DATA
3400                                CMP     #3200,R3       ;VERIFY FPP STATUS
3401 014706 012704 001346      BEQ     1$            ;BRANCH IF GOOD
3402 014712 004767 165230      ERROR    ;ALL ERRORS TO TRAP TO EMT VECTOR
3403 014716 005767 164116      .WORD   252            ;UNIQUE ERROR NUMBER
3404 014722 001403              .WORD   FPPERR        ;ADDRESS OF ERROR MESSAGE
3405 014724 104000
3406 014726 000253              ;BAD FPP STATUS
3407                                ;POINT TO EXPECTED DATA
3408                                ;VERIFY CONTENTS OF ACO
3409                                ;
3410                                ;BRANCH IF GOOD ACO
3411                                ;ALL ERRORS TO TRAP TO EMT VECTOR
3412                                ;UNIQUE ERROR NUMBER
```


D6

```
3407 014730 002013          .WORD  FPPERR          ;ADDRESS OF ERROR MESSAGE
3408                                     ;BAD ACO, SHOULD = FSRC
3409 014732 012704 001356    2$: MOV    #TAB12,R4      ;POINT TO FSRC DATA
3410 014736 172415          LDD    (R5),ACO          ;ACO
3411 014740 172014          ADDD   (R4),ACO          ;*TEST INSTRUCTION
3412 014742 012704 001376    MOV    #TAB13B,R4       ;POINT TO EXPECTED RESULT
3413 014746 174011          STD    ACO,(R1)          ;SAVE ACO DATA INTO RECDAT
3414 014750 004767 165172    JSR    R7,DATVER        ;VERIFY DATA
3415 014754 005767 164060    TST    COUNT
3416 014760 001403          BEQ    3$
3417 014762 104000          ERROR
3418 014764 000254          .WORD  254
3419 014766 002013          .WORD  FPPERR
3420                                     ;BRANCH IF GOOD DATA
3421 014770 012702 003000    3$: MOV    #3000,R2      ;ALL ERRORS TO TRAP TO EMT VECTOR
3422 014774 012704 001406    MOV    #TAB14,R4      ;UNIQUE ERROR NUMBER
3423 015000 012705 001416    MOV    #TAB15,R5      ;ADDRESS OF ERROR MESSAGE
3424 015004 172415          LDD    (R5),ACO          ;BAD ACO DATA
3425 015006 170102          LDFPS  R2              ;GET FPP STATUS DATA
3426 015010 172014          ADDF   (R4),ACO          ;POINT TO FSRC DATA
3427 015012 170127 000200    LDFPS  #200           ;POINT TO ACO DATA
3428 015016 174011          STD    ACO,(R1)          ;LOAD ACO
3429 015020 012704 001326    MOV    #TAB10,R4       ;FPP STATUS = FLOAT, INTERRUPTS ENABLE
3430 015024 004767 165116    JSR    R7,DATVER        ;*TEST INSTRUCTION
3431 015030 005767 164004    TST    COUNT           ;RESET TO DOUBLE
3432 015034 001403          BEQ    4$
3433 015036 104000          ERROR
3434 015040 000255          .WORD  255
3435 015042 002013          .WORD  FPPERR
3436                                     ;RECDST=ACO
3437 015044 012705 001426    4$: MOV    #TAB16,R5      ;POINT TO GOOD DATA
3438 015050 170102          LDFPS  R2              ;VERIFY CONTENTS OF ACO
3439 015052 172415          LDF   (R5),ACO          ;BRANCH IF GOOD
3440 015054 172014          ADDF   (R4),ACO          ;ALL ERRORS TO TRAP TO EMT VECTOR
3441 015056 174011          STD    ACO,(R1)          ;UNIQUE ERROR NUMBER
3442 015060 012704 001436    MOV    #TAB17,R4       ;ADDRESS OF ERROR MESSAGE
3443 015064 004767 165056    JSR    R7,DATVER        ;BAD FLOATING ADD
3444 015070 005767 163744    TST    COUNT           ;POINT TO ACO DATA
3445 015074 001403          BEQ    5$
3446 015076 104000          ERROR
3447 015100 000256          .WORD  256
3448 015102 002013          .WORD  FPPERR
3449                                     ;FPP STATUS = FLOAT
3450 015104          5$:                                     ;LOAD ACO
3451                                     ;*TEST INSTRUCTION
3452                                     ;SAVE ACO DATA
3453                                     ;POINT TO GOOD DATA
3454 015104          MNGOP:
3455          ;*****
3456          ;*TEST 53      TEST ADD WITH NEGATIVE OPERANDS
3457          ;*****
3458          TST53:
3459 015104 005267 163674    INC     #TESTN          ;INCREMENT TEST NUMBER
3460 015110 012702 003200    MOV     #3200,R2        ;LOAD FPS VALUE
3461 015114 170102          LDFPS  R2
3462 015116 012704 001456    MOV     #TAB21,R4       ;DATA ADDRESS FOR ACO AND FSR
```

Address	Offset	Value	Label	Instruction	Comment
3463	015122	172414		LDD (R4),ACO	;ACO=100200 0 0 0
3464	015124	172014		ADD (R4),ACO	;*TEST INSTRUCTION
3465	015126	170203		STFPS R3	;SAVE STATUS
3466	015130	012701	001126	MOV #RECDST,R1	;POINT TO RECEIVED DATA TABLE
3467	015134	174011		STD ACO,(R1)	;SAVE ACO DATA
3468	015136	022703	003210	CMP #3210,R3	;VERIFY STATUS
3469	015142	001403		BEQ 1\$	;BRANCH IF GOOD
3470	015144	104000		ERROR	;ALL ERRORS TO TRAP TO EMT VECTOR
3471	015146	000257		.WORD 257	;UNIQUE ERROR NUMBER
3472	015150	002013		.WORD FPPERR	;ADDRESS OF ERROR MESSAGE
3473					
3474	015152	012704	001466	1\$: MOV #TAB22,R4	;POINT TO EXPECTED DATA
3475	015156	004767	164764	JSR R7,DATVER	
3476	015162	005767	163652	TST COUNT	
3477	015166	001403		BEQ 2\$	;VERIFY DATA
3478	015170	104000		ERROR	;BRANCH IF GOOD
3479	015172	000260		.WORD 260	;ALL ERRORS TO TRAP TO EMT VECTOR
3480	015174	002013		.WORD FPPERR	;UNIQUE ERROR NUMBER
3481					;ADDRESS OF ERROR MESSAGE
3482					
3483	015176	012704	001456	2\$: !-FSRC! = !ACO!	
3484	015202	012701	001476	MOV #TAB21,R4	;POINT TO FSRC DATA
3485	015206	012737	015230	MOV #TAB23,R1	;POINT TO ACO DATA
3486	015214	172411	000244	MOV #101\$,B#FPVEC	;SETUP FP VECTOR
3487	015216	172014		LDD (R1),ACO	;LOAD ACO
3488	015220	170000		ADD (R4),ACO	;*TEST INSTRUCTION
3489	015222	104000		100\$: CFCC	;COPY FPP CC
3490	015224	000261		ERROR	;ALL ERRORS TO TRAP TO EMT VECTOR
3491	015226	002013		.WORD 261	;UNIQUE ERROR NUMBER
3492				.WORD FPPERR	;ADDRESS OF ERROR MESSAGE
3493	015230	170203			GO TO ERROR
3494	015232	012701	001126	101\$: STFPS R3	;SAVE FPP STATUS
3495	015236	174011		MOV #RECDST,R1	;POINT TO RECEIVED DATA TABLE
3496	015240	022703	103200	STD ACO,(R1)	;SAVE ACO DATA
3497	015244	001403		CMP #103200,R3	;VERIFY STATUS
3498	015246	104000		BEQ 3\$	;BRANCH IF GOOD
3499	015250	000262		ERROR	;ALL ERRORS TO TRAP TO EMT VECTOR
3500	015252	002013		.WORD 262	;UNIQUE ERROR NUMBER
3501				.WORD FPPERR	;ADDRESS OF ERROR MESSAGE
3502	015254	012605			BAD STATUS
3503	015256	020527	015220	3\$: MOV (SP)+,R5	;GET ERROR PC
3504	015262	001403		CMP R5,#100\$	;VERIFY ERROR ADDRESS ON STACK
3505	015264	104000		BEQ 102\$	;BRANCH IF GOOD
3506	015266	000263		ERROR	;ALL ERRORS TO TRAP TO EMT VECTOR
3507	015270	002013		.WORD 263	;UNIQUE ERROR NUMBER
3508				.WORD FPPERR	;ADDRESS OF ERROR MESSAGE
3509	015272	005726			BAD ERROR RETURN ON STACK
3510	015274	012704	001506	102\$: TST (SP)+	;RESTORE STACK
3511	015300	004767	164642	MOV #TAB24,R4	;POINT TO EXPECTED DATA TABLE
3512	015304	005767	163530	JSR R7,DATVER	;VERIFY DATA
3513	015310	001403		TST COUNT	
3514	015312	104000		BEQ 4\$	;BRANC IF GOOD
3515	015314	000264		ERROR	;ALL ERRORS TO TRAP TO EMT VECTOR
3516	015316	002013		.WORD 264	;UNIQUE ERROR NUMBER
3517				.WORD FPPERR	;ADDRESS OF ERROR MESSAGE
3518					BAD ACO DATA
				!-ACO! = !FSRC!	

3519	015320	012704	001476		4:	MOV	0TAB23,R4		;POINT TO FSRC DATA
3520	015324	012701	001456			MOV	0TAB21,R1		;POINT TO ACO DATA
3521	015330	012737	015360	000244		MOV	0104,00FPVEC		;SETUP FP VECTOR
3522	015336	012702	003200			MOV	03200,R2		;LOAD FPS VALUE
3523	015342	170102				LDFPS	R2		
3524	015344	172411				LDD	(R1),ACO		;LOAD ACO DATA
3525	015346	172014				ADD	(R4),ACO		;*TEST INSTRUCTION
3526	015350	170000			103:	CFCC			;COPY FPP CC
3527	015352	104000				ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
3528	015354	000265				.WORD	265		;UNIQUE ERROR NUMBER
3529	015356	002013				.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
3530									
3531	015360	170203			104:	STFPS	R3		;GO TO ERROR
3532	015362	012701	001126			MOV	0RECDST,R1		;SAVE FPS
3533	015366	174011				STD	ACO,(R1)		;SAVE ACO
3534	015370	022703	103200			CMP	0103200,R3		
3535	015374	001403				BEQ	5:		;VERIFY STATUS
3536	015376	104000				ERROR			;BRANCH IF GOOD
3537	015400	000266				.WORD	266		;ALL ERRORS TO TRAP TO EMT VECTOR
3538	015402	002013				.WORD	FPPERR		;UNIQUE ERROR NUMBER
3539									;ADDRESS OF ERROR MESSAGE
3540	015404	012605			5:	MOV	(SP),R5		;BAD FPS STATUS
3541	015406	020527	015350			CMP	R5,0103:		;GET ERROR PC
3542	015412	001403				BEQ	105:		;VERIFY ERROR ADDRESS ON STACK
3543	015414	104000				ERROR			;BRANCH IF GOOD
3544	015416	000267				.WORD	267		;ALL ERRORS TO TRAP TO EMT VECTOR
3545	015420	002013				.WORD	FPPERR		;UNIQUE ERROR NUMBER
3546									;ADDRESS OF ERROR MESSAGE
3547	015422	005726			105:	TST	(SP),		;BAD ERROR RETURN ON STACK
3548	015424	012704	001506			MOV	0TAB24,R4		;RESTORE STACK
3549	015430	004767	164512			JSR	R7,DATVER		;POINT TO EXPECTED DATA
3550	015434	005767	163400			TST	COUNT		
3551	015440	001403				BEQ	6:		
3552	015442	104000				ERROR			;BRANCH IF GOOD
3553	015444	000270				.WORD	270		;ALL ERRORS TO TRAP TO EMT VECTOR
3554	015446	002013				.WORD	FPPERR		;UNIQUE ERROR NUMBER
3555									;ADDRESS OF ERROR MESSAGE
3556									
3557	015450	012704	001526		6:	MOV	!-FSRC! ,		;BAD ACO
3558	015454	012701	001516			MOV	0TAB26,R4		!ACO!
3559	015460	012702	003200			MOV	0TAB25,R1		;POINT TO FSRC DATA
3560	015464	170102				MOV	03200,R2		;POINT TO ACO DATA
3561	015466	012737	000246	000244		LDFPS	R2		;LOAD FPS VALUE
3562	015474	172411				MOV	0246,00FPVEC		
3563	015476	172014				LDD	(R1),ACO		;SETUP FP VECTOR
3564	015500	170203				ADD	(R4),ACO		;LOAD ACO DATA
3565	015502	012701	001126			STFPS	R3		;*TEST INSTRUCTION
3566	015506	174011				MOV	0RECDST,R1		;SAVE STATUS
3567	015510	020327	003200			STD	ACO,(R1)		;POINT TO RECEIVED DATA TABLE
3568	015514	001403				CMP	R3,03200		;SAVE ACO
3569	015516	104000				BEQ	7:		;VERIFY STATUS
3570	015520	000271				ERROR			;BRANCH IF GOOD
3571	015522	002013				.WORD	271		;ALL ERRORS TO TRAP TO EMT VECTOR
3572						.WORD	FPPERR		;UNIQUE ERROR NUMBER
3573	015524	012704	001536		7:	MOV	0TAB27,R4		;ADDRESS OF ERROR MESSAGE
3574	015530	004767	164412			JSR	R7,DATVER		

Address	Offset	Hex	Label	Instruction	Comment
3575	015534	005767	163300	TST	COUNT
3576	015540	001403		BEQ	8:
3577	015542	104000		ERROR	;
3578	015544	000272		.WORD	272
3579	015546	002013		.WORD	FPPERR
3580					;
3581					;-AC!
3582	015550	012704	001516	MOV	!FSRC! >
3583	015554	012701	001526	MOV	8:; #TAB25,R4
3584	015560	172411		MOV	#TAB26,R1
3585	015562	172014		LDD	(R1),ACO
3586	015564	170203		ADD	(R4),ACO
3587	015566	012701	001126	STFPS	R3
3588	015572	174011		MOV	#RECDST,R1
3589	015574	020327	003200	STD	ACO,(R1)
3590	015600	001403		CMP	R3,#3200
3591	015602	104000		BEQ	9:
3592	015604	000273		ERROR	;
3593	015606	002013		.WORD	273
3594				.WORD	FPPERR
3595	015610	012704	001536	MOV	9:; #TAB27,R4
3596	015614	004767	164326	JSR	R7,DATVER
3597	015620	005767	163214	TST	COUNT
3598	015624	001403		BEQ	10:
3599	015626	104000		ERROR	;
3600	015630	000274		.WORD	274
3601	015632	002013		.WORD	FPPERR
3602					;
3603					;-FSRC! <
3604	015634	012704	001556	MOV	10:; #TAB29,R4
3605	015640	012701	001546	MOV	#TAB28,R1
3606	015644	172411		LDD	(R1),ACO
3607	015646	172014		ADD	(R4),ACO
3608	015650	170203		STFPS	R3
3609	015652	012701	001126	MOV	#RECDST,R1
3610	015656	174011		STD	ACO,(R1)
3611	015660	020327	003200	CMP	R3,#3200
3612	015664	001403		BEQ	11:
3613	015666	104000		ERROR	;
3614	015670	000275		.WORD	275
3615	015672	002013		.WORD	FPPERR
3616					;
3617	015674	012704	001566	MOV	11:; #TAB29A,R4
3618	015700	004767	164242	JSR	R7,DATVER
3619	015704	005767	163130	TST	COUNT
3620	015710	001403		BEQ	12:
3621	015712	104000		ERROR	;
3622	015714	000276		.WORD	276
3623	015716	002013		.WORD	FPPERR
3624					;
3625	015720				;
3626					;
3627					;
3628					;
3629	015720				MSB:
3630					;

3631
3632
3633 015720
3634 015720 005267 163060
3635 015724 012702 003200
3636 015730 170102
3637 015732 012704 001456
3638 015736 012701 001126
3639 015742 172414
3640 015744 173014
3641 015746 170203
3642 015750 174011
3643 015752 022703 003204
3644 015756 001403
3645 015760 104000
3646 015762 000277
3647 015764 002013
3648
3649 015766 012704 001256
3650 015772 004767 164150
3651 015776 005767 163036
3652 016002 001403
3653 016004 104000
3654 016006 000300
3655 016010 002013
3656
3657 016012 012704 001406
3658 016016 172414
3659 016020 173014
3660 016022 170203
3661 016024 174011
3662 016026 022703 003204
3663 016032 001403
3664 016034 104000
3665 016036 000301
3666 016040 002013
3667
3668 016042 012704 001256
3669 016046 004767 164074
3670 016052 005767 162762
3671 016056 001403
3672 016060 104000
3673 016062 000302
3674 016064 002013
3675
3676 016066
3677
3678
3679 016066
3680
3681
3682
3683 016066
3684 016066 005267 162712
3685 016072 012702 003200
3686 016076 170102

;*TEST 54 TEST SUB WITH EXP[ACO]=EXP[FSRC]

TST54:

INC #TESTN ; INCREMENT TEST NUMBER
MOV #3200,R2 ; LOAD FPS DATA
LDFPS R2 ; LOAD FPS
MOV #TAB21,R4 ; POINT TO FSRC DATA
MOV #RECDST,R1 ; POINT TO ACO RECEIVED DATA TABLE
LDD (R4),ACO ; LOAD ACO
SUBD (R4),ACO ; *TEST INSTRUCTION
STFPS R3 ; SAVE STATUS
STD ACO,(R1) ; SAVE ACO INTO RECDST
CMP #3204,R3 ; VERIFY STATUS
BEQ 1\$; BRANCH IF GOOD
ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 277 ; UNIQUE ERROR NUMBER
.WORD FPPERR ; ADDRESS OF ERROR MESSAGE
;BAD FPS STATUS
MOV #TAB6,R4 ; POINT TO EXPECTED DATA
JSR R7,DATVER ; VERIFY ACO
TST COUNT
BEQ 2\$; BRANCH IF GOOD
ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 300 ; UNIQUE ERROR NUMBER
.WORD FPPERR ; ADDRESS OF ERROR MESSAGE
;BAD ACO
MOV #TAB14,R4 ; POINT TO FSRC AND ACO DATA
LDD (R4),ACO ; LOAD ACO DATA
SUBD (R4),ACO ; *TEST INSTRUCTION
STFPS R3 ; SAVE FPS
STD ACO,(R1) ; SAVE ACO INTO RECDST
CMP #3204,R3 ; VERIFY FPS
BEQ 3\$; BRANCH IF GOOD
ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 301 ; UNIQUE ERROR NUMBER
.WORD FPPERR ; ADDRESS OF ERROR MESSAGE
;BAD ACO
MOV #TAB6,R4 ; POINT TO EXPECTED DATA
JSR R7,DATVER ; VERIFY ACO
TST COUNT
BEQ 4\$; BRANCH IF GOOD
ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 302 ; UNIQUE ERROR NUMBER
.WORD FPPERR ; ADDRESS OF ERROR MESSAGE
;BAD ACO
4\$:

; MNRM:

;*TEST 55 TEST NORMALIZE

TST55:

INC #TESTN ; INCREMENT TEST NUMBER
MOV #3200,R2 ; LOAD FPS
LDFPS R2 ;

3687	016100	012705	001606	MOV	#TAB31,R5	;POINT TO FSRC DATA
3688	016104	012701	001576	MOV	#TAB30,R1	;POINT TO ACO DATA
3689	016110	172411		LDD	(R1),ACO	;LOAD ACO
3690	016112	173015		SUBD	(R5),ACO	;*TEST INSTRUCTION
3691						;1 LEFT SHIFT
3692	016114	170203		STFPS	R3	;SAVE STATUS
3693	016116	012704	001126	MOV	#RECDST,R4	;POINT TO RECDATA
3694	016122	174014		STD	ACO,(R4)	;SAVE ACO
3695	016124	012701	001636	MOV	#TAB34,R1	;POINT TO EXPECTED DATA
3696	016130	004767	164012	JSR	R7,DATVER	;VERIFY DATA
3697	016134	005767	162700	TST	COUNT	
3698	016140	001403		BEQ	1:	;BRANCH IF GOOD
3699	016142	104000		ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
3700	016144	000303		.WORD	303	;UNIQUE ERROR NUMBER
3701	016146	002013		.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
3702						
3703	016150	012701	001616	1: MOV	#TAB32,R1	;ACO DATA
3704	016154	012705	001626	MOV	#TAB33,R5	;FSRC DATA
3705	016160	172411		LDD	(R1),ACO	;LOAD ACO
3706	016162	173015		SUBD	(R5),ACO	;*TST INSTRUCTION
3707						;56 LEFT SHIFTS
3708	016164	012701	001126	MOV	#RECDST,R1	;SAVE DATA
3709	016170	174011		STD	ACO,(R1)	
3710	016172	004767	163750	JSR	R7,DATVER	
3711	016176	005767	162636	TST	COUNT	
3712	016202	001403		BEQ	2:	
3713	016204	104000		ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
3714	016206	000304		.WORD	304	;UNIQUE ERROR NUMBER
3715	016210	002013		.WORD	FPPERR	;ADDRESS OF ERROR MESSAGE
3716						
3717	016212			2: MUVAD:		
3718						
3719	016212					
3720						
3721						
3722						
3723	016212					
3724	016212	005267	162566	TST56:	INC	#TESTN
3725	016216	012702	000200	MOV	#200,R2	;INCREMENT TEST NUMBER
3726	016222	170102		LDFPS	R2	;SETUP FLOATING POINT STATUS
3727	016224	012704	001646	MOV	#TAB40,R4	;LOAD FPS
3728	016230	012701	001646	MOV	#TAB40,R1	;POINT TO FSRC DATA
3729	016234	172411		LDD	(R1),ACO	;POINT TO ACO DATA
3730	016236	172014		ADD	(R4),ACO	;LOAD ACO WITH TEST DATA
3731	016240	170203		STFPS	R3	;*TEST INSTRUCTION
3732	016242	012701	001126	MOV	#RECDST,R1	;SAVE FPS
3733	016246	174011		STD	ACO,(R1)	;POINT TO RECEIVED DATA TABLE
3734	016250	022703	000206	CMP	#206,R3	;SAVE ACO RESULT
3735	016254	001403		BEQ	1:	;VERIFY STATUS
3736	016256	104000		ERROR		;BRANCH IF GOOD
3737	016260	000305		.WORD	305	;ALL ERRORS TO TRAP TO EMT VECTOR
3738	016262	002013		.WORD	FPPERR	;UNIQUE ERROR NUMBER
3739						;ADDRESS OF ERROR MESSAGE
3740	016264	012704	001256	1: MOV	#TAB6,R4	;BAD FPS
3741	016270	004767	163652	JSR	R7,DATVER	;POINT TO EXPECTED DATA
3742	016274	005767	162540	TST	COUNT	;VERIFY DATA

3743 016300 001403 BEQ 2# ;BRANCH IF GOOD
3744 016302 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
3745 016304 000306 .WORD 306 ;UNIQUE ERROR NUMBER
3746 016306 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
3747 ;BAD ACO
3748 ;OVERFLOW TRAPS ENABLED
3749 016310 012702 001200 2# : MOV #1200,R2 ;SETUP FLOATING POINT STATUS
3750 016314 170102 LDFPS R2 ;LOAD FPS
3751 016316 012704 001646 MOV #TAB40,R4 ;POINT TO FSRC DATA
3752 016322 012701 001646 MOV #TAB40,R1 ;POINT TO ACO DATA
3753 016326 172411 LDD (R1),ACO ;LOAD ACO WITH TEST DATA
3754 016330 012737 016350 000244 MOV #3#,BFPVEC ;CHANGE TRAP VECTOR
3755 016336 172014 ADDD (R4),ACO ;*TEST INSTRUCTION
3756 016340 170000 23# : CFCC
3757 016342 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
3758 016344 000307 .WORD 307 ;UNIQUE ERROR NUMBER
3759 016346 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
3760 ;FAILED TO TRAP ON OVERFLOW
3761 016350 170203 3# : STFPS R3 ;SAVE FPS
3762 016352 012701 001126 MOV #RECDST,R1 ;POINT TO RECEIVED DATA TABLE
3763 016356 174011 STD ACO,(R1) ;SAVE ACO RESULT
3764 016360 022703 101206 CMP #101206,R3 ;VERIFY STATUS
3765 016364 001403 BEQ 4# ;BRANCH IF GOOD
3766 016366 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
3767 016370 000310 .WORD 310 ;UNIQUE ERROR NUMBER
3768 016372 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
3769 ;BAD FPS
3770 016374 012600 4# : MOV (SP)+,R0 ;CHECK STORED PC
3771 016376 022700 016340 CMP #23#,R0 ;
3772 016402 001403 BEQ 5# ;BRANCH IF RETURN ADDRESS IS GOOD
3773 016404 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
3774 016406 000311 .WORD 311 ;UNIQUE ERROR NUMBER
3775 016410 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
3776 ;BAD RETURN ADDRESS
3777 016412 012600 5# : MOV (SP)+,R0 ;CLEAN UP STACK
3778 016414 012704 001256 MOV #TAB6,R4 ;POINT TO EXPECTED DATA
3779 016420 004767 163522 JSR R7,DATVER ;VERIFY DATA
3780 016424 005767 162410 TST COUNT
3781 016430 001403 BEQ 7# ;BRANCH IF GOOD
3782 016432 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
3783 016434 000312 .WORD 312 ;UNIQUE ERROR NUMBER
3784 016436 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
3785 ;BAD ACO
3786 ;UNDERFLOW TRAPS DISABLED
3787 016440 012702 000200 7# : MOV #200,R2 ;SETUP FLOATING POINT STATUS
3788 016444 170102 LDFPS R2 ;LOAD FPS
3789 016446 012737 002106 000244 MOV #WLDTRP,BFPVEC ;REPLACE WILD TRAP VECTOR
3790 016454 012704 001276 MOV #TAB7,R4 ;POINT TO FSRC DATA
3791 016460 012701 001656 MOV #TAB41,R1 ;POINT TO ACO DATA
3792 016464 172411 LDD (R1),ACO ;LOAD ACO WITH TEST DATA
3793 016466 172014 ADDD (R4),ACO ;*TEST INSTRUCTION
3794 016470 170203 STFPS R3 ;SAVE FPS
3795 016472 012701 001126 MOV #RECDST,R1 ;POINT TO RECEIVED DATA TABLE
3796 016476 174011 STD ACO,(R1) ;SAVE ACO RESULT
3797 016500 022703 000204 CMP #204,R3 ;VERIFY STATUS
3798 016504 001403 BEQ 8# ;BRANCH IF GOOD

```
3799 016506 104000
3800 016510 000313
3801 016512 002013
3802
3803 016514 012704 001256
3804 016520 004767 163422
3805 016524 005767 162310
3806 016530 001403
3807 016532 104000
3808 016534 000314
3809 016536 002013
3810
3811
3812 016540 012702 002200
3813 016544 170102
3814 016546 012737 016600 000244
3815 016554 012704 001276
3816 016560 012701 001656
3817 016564 172411
3818 016566 172014
3819 016570 170000
3820 016572 104000
3821 016574 000315
3822 016576 002013
3823
3824 016600 170203
3825 016602 012701 001126
3826 016606 174011
3827 016610 022703 102210
3828 016614 001403
3829 016616 104000
3830 016620 000316
3831 016622 002013
3832
3833 016624 012605
3834 016626 020527 016570
3835 016632 001403
3836 016634 104000
3837 016636 000317
3838 016640 002013
3839
3840 016642 005726
3841 016644 012704 001246
3842 016650 004767 163272
3843 016654 005767 162160
3844 016660 001403
3845 016662 104000
3846 016664 000320
3847 016666 002013
3848
3849
3850 016670 012702 000200
3851 016674 170102
3852 016676 012737 002106 000244
3853 016704 012704 001656
3854 016710 012701 001666

      ERROR
      .WORD 313
      .WORD FPPERR
      .WORD FPPERR
      8$: MOV #TAB6,R4
      JSR R7,DATVER
      TST COUNT
      BEQ 9$
      ERROR
      .WORD 314
      .WORD FPPERR
      .UNDERFLOW TRAPS ENABLED
      9$: MOV #2200,R2
      LDFPS R2
      MOV #11$,B#FPVEC
      MOV #TAB7,R4
      MOV #TAB41,R1
      LDD (R1),ACO
      ADDD (R4),ACO
      10$: CFCC
      ERROR
      .WORD 315
      .WORD FPPERR
      11$: STFPS R3
      MOV #RECDST,R1
      STD ACO,(R1)
      CMP #102210,R3
      BEQ 12$
      ERROR
      .WORD 316
      .WORD FPPERR
      12$: MOV (SP)+,R5
      CMP R5,#10$
      BEQ 13$
      ERROR
      .WORD 317
      .WORD FPPERR
      13$: TST (SP)+
      MOV #TAB5A,R4
      JSR R7,DATVER
      TST COUNT
      BEQ 14$
      ERROR
      .WORD 320
      .WORD FPPERR
      .UNDERFLOW WITH TRAPS DISABLED - NON-ZERO RESULT
      14$: MOV #200,R2
      LDFPS R2
      MOV #WLDTRP,B#FPVEC
      MOV #TAB41,R4
      MOV #TAB42,R1

;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;BAD FPS
;POINT TO EXPECTED DATA
;VERIFY DATA
;BRANCH IF GOOD
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;BAD ACO
;UNDERFLOW TRAPS ENABLED
9$:
;SETUP FLOATING POINT STATUS
;LOAD FPS
;REPOSITION TRAP VECTOR
;POINT TO FSRC DATA
;POINT TO ACO DATA
;LOAD ACO WITH TEST DATA
;TEST INSTRUCTION
;COPY FPP CC
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;FAILED TO TRAP ON UNDERFLOW
;SAVE FPS
;POINT TO RECEIVED DATA TABLE
;SAVE ACO RESULT
;VERIFY STATUS
;BRANCH IF GOOD
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;BAD FPS
;GET ERROR PC
;VERIFY ERROR ADDRESS ON STACK
;BRANCH IF GOOD
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;BAD ERROR RETURN ON STACK
;RESTORE STACK
;POINT TO EXPECTED DATA
;VERIFY DATA
;BRANCH IF GOOD
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;BAD ACO
;NON-ZERO RESULT
;SETUP FLOATING POINT STATUS
;LOAD FPS
;RESTORE TRAP VECTOR
;POINT TO FSRC DATA
;POINT TO ACO DATA
```


3855	016714	172411			LDD	(R1),ACO		;LOAD ACO WITH TEST DATA
3856	016716	172014			ADD	(R4),ACO		;TEST INSTRUCTION
3857	016720	170203			STFPS	R3		;SAVE FPS
3858	016722	012701	001126		MOV	#RECDST,R1		;POINT TO RECEIVED DATA TABLE
3859	016726	174011			STD	ACO,(R1)		;SAVE ACO RESULT
3860	016730	022703	000204		CMP	#204,R3		;VERIFY STATUS
3861	016734	001403			BEQ	15:		;BRANCH IF GOOD
3862	016736	104000			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
3863	016740	000321			.WORD	321		;UNIQUE ERROR NUMBER
3864	016742	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
3865								;BAD FPS
3866	016744	012704	001256		15:	MOV	#TAB6,R4	;POINT TO EXPECTED DATA
3867	016750	004767	163172			JSR	R7,DATVER	;VERIFY DATA
3868	016754	005767	162060			TST	COUNT	
3869	016760	001403				BEQ	16:	;BRANCH IF GOOD
3870	016762	104000				ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
3871	016764	000322			.WORD	322		;UNIQUE ERROR NUMBER
3872	016766	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
3873								;BAD ACO
3874								;NON-ZERO RESULT
3875	016770	012702	102200		16:	MOV	#102200,R2	;SETUP FLOATING POINT STATUS
3876	016774	170102				LDFPS	R2	;LOAD FPS
3877	016776	012737	017030	000244		MOV	#181,#FPVEC	;RESTORE TRAP VECTOR
3878	017004	012704	001656			MOV	#TAB41,R4	;POINT TO FSRC DATA
3879	017010	012701	001666			MOV	#TAB42,R1	;POINT TO ACO DATA
3880	017014	172411				LDD	(R1),ACO	;LOAD ACO WITH TEST DATA
3881	017016	172014				ADD	(R4),ACO	;TEST INSTRUCTION
3882	017020	170000			17:	CFCC		
3883	017022	104000				ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
3884	017024	000323			.WORD	323		;UNIQUE ERROR NUMBER
3885	017026	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
3886								;NO TRAP ON UNDERFLOW
3887	017030	170203			18:	STFPS	R3	;SAVE FPS
3888	017032	012701	001126			MOV	#RECDST,R1	;POINT TO RECEIVED DATA TABLE
3889	017036	174011				STD	ACO,(R1)	;SAVE ACO RESULT
3890	017040	012600				MOV	(SP),R0	;SAVE STACK CONTENTS
3891	017042	005726				TST	(SP),	;CLEAN UP STACK
3892	017044	022700	017020			CMP	#171,R0	;VERIFY RETURN ADDRESS
3893	017050	001403				BEQ	19:	;BRANCH IF GOOD
3894	017052	104000				ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
3895	017054	000324			.WORD	324		;UNIQUE ERROR NUMBER
3896	017056	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
3897								;BAD RETURN ADDRESS
3898	017060	022703	102204		19:	CMP	#102204,R3	;VERIFY STATUS
3899	017064	001403				BEQ	20:	;BRANCH IF GOOD
3900	017066	104000				ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
3901	017070	000325			.WORD	325		;UNIQUE ERROR NUMBER
3902	017072	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
3903								;BAD FPS
3904	017074	012704	001676		20:	MOV	#TAB43,R4	;POINT TO EXPECTED DATA
3905	017100	004767	163042			JSR	R7,DATVER	;VERIFY DATA
3906	017104	005767	161730			TST	COUNT	
3907	017110	001403				BEQ	21:	;BRANCH IF GOOD
3908	017112	104000				ERROR		;ALL ERRORS TO TRAP TO EMT VECTOR
3909	017114	000326			.WORD	326		;UNIQUE ERROR NUMBER
3910	017116	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE

Address	Offset	Instruction	Comment
3911			
3912	017120		
3913			
3914			
3915	017120		
3916			
3917			
3918			
3919	017120		
3920	017120	005267 161660	
3921			
3922	017124	012702 000300	
3923	017130	170102	
3924	017132	012704 001706	
3925	017136	012701 001256	
3926	017142	172411	
3927	017144	177424	
3928	017146	012701 001126	
3929	017152	174011	
3930	017154	022704 001712	
3931	017160	001403	
3932	017162	104000	
3933	017164	000327	
3934	017166	002013	
3935			
3936	017170	012704 001716	
3937	017174	004767 162746	
3938	017200	005767 161634	
3939	017204	001403	
3940	017206	104000	
3941	017210	000330	
3942	017212	002013	
3943			
3944			
3945	017214	005002	
3946	017216	170102	
3947	017220	012704 001706	
3948	017224	012701 001426	
3949	017230	172411	
3950	017232	177424	
3951	017234	020427 001716	
3952	017240	001403	
3953	017242	104000	
3954	017244	000331	
3955	017246	002013	
3956			
3957	017250	170203	
3958	017252	012701 001126	
3959	017256	174011	
3960	017260	022703 000000	
3961	017264	001403	
3962	017266	104000	
3963	017270	000332	
3964	017272	002013	
3965			
3966	017274	012704 001756	

3967	017300	004767	162642
3968	017304	005767	161530
3969	017310	001403	
3970	017312	104000	
3971	017314	000333	
3972	017316	002013	
3973			
3974			
3975	017320	012702	000200
3976	017324	170102	
3977	017326	005003	
3978	017330	177427	043243
3979	017334	005203	
3980	017336	005203	
3981	017340	005203	
3982	017342	022703	000003
3983	017346	001403	
3984	017350	104000	
3985	017352	000334	
3986	017354	002013	
3987			
3988			
3989	017356	012702	000200
3990	017362	170102	
3991	017364	012704	001726
3992	017370	012701	001706
3993	017374	172411	
3994	017376	177414	
3995	017400	170203	
3996	017402	012701	001126
3997	017406	174011	
3998	017410	022703	000210
3999	017414	001403	
4000	017416	104000	
4001	017420	000335	
4002	017422	002013	
4003			
4004	017424	012704	001746
4005	017430	004767	162512
4006	017434	005767	161400
4007	017440	001403	
4008	017442	104000	
4009	017444	000336	
4010	017446	002013	
4011			
4012			
4013	017450	012702	000200
4014	017454	170102	
4015	017456	012704	001256
4016	017462	012701	001746
4017	017466	172411	
4018	017470	177414	
4019	017472	170203	
4020	017474	012701	001126
4021	017500	174011	
4022	017502	022703	000204

```

JSR    R7,DATVER
TST    COUNT
BEQ    5$
ERROR  333
.WORD  FPPERR
.WORD

;LDCFD GR7
5$:    MOV    #200,R2
      LDFPS  R2
      CLR    R3
      LDCFD  #5203,ACO
      INC    R3
      INC    R3
      INC    R3
      CMP    #3,R3
      BEQ    6$
      ERROR  334
      .WORD  FPPERR
      .WORD

;NEGATIVE OPERANDS
6$:    MOV    #200,R2
      LDFPS  R2
      MOV    #TAB47,R4
      MOV    #TAB45,R1
      LDD    (R1),ACO
      LDCFD  (R4),ACO
      STFPS  R3
      MOV    #RECDST,R1
      STD    ACO,(R1)
      CMP    #210,R3
      BEQ    7$
      ERROR  335
      .WORD  FPPERR
      .WORD

7$:    MOV    #TAB48,R4
      JSR    R7,DATVER
      TST    COUNT
      BEQ    8$
      ERROR  336
      .WORD  FPPERR
      .WORD

;LOAD A ZERO
8$:    MOV    #200,R2
      LDFPS  R2
      MOV    #TAB6,R4
      MOV    #TAB48,R1
      LDD    (R1),ACO
      LDCFD  (R4),ACO
      STFPS  R3
      MOV    #RECDST,R1
      STD    ACO,(R1)
      CMP    #204,R3
  
```

```

;VERIFY DATA
;BRANCH IF GOOD
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;BAD ACO

;SETUP FLOATING POINT STATUS
;LOAD FPS

;*TEST INSTRUCTION

;IF LDCFD WORKED, R3 SHOULD=3
;VERIFY CORRECT PROGRAM FLOW
;BRANCH IF GOOD
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;BAD PROGRAM FLOW

;SETUP FLOATING POINT STATUS
;LOAD FPS
;POINT TO FSRC DATA
;POINT TO ACO DATA
;LOAD ACO WITH TEST DATA
;*TEST INSTRUCTION
;SAVE FPS
;POINT TO RECEIVED DATA TABLE
;SAVE ACO RESULT
;VERIFY STATUS
;BRANCH IF GOOD
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;BAD FPS

;POINT TO EXPECTED DATA
;VERIFY DATA

;BRANCH IF GOOD
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;BAD ACO

;SETUP FLOATING POINT STATUS
;LOAD FPS
;POINT TO FSRC DATA
;POINT TO ACO DATA
;LOAD ACO WITH TEST DATA
;*TEST INSTRUCTION
;SAVE FPS
;POINT TO RECEIVED DATA TABLE
;SAVE ACO RESULT
;VERIFY STATUS
  
```


GLOBAL AREAS MACY11 30A(1052) 15-MAR-84 16:58 PAGE 80
KDJ11A.MAC 15-MAR-84 15:51 T60 TEST CHPD

4135					;*TEST 61	TEST DIVF	
4136					;;*****		
4137	020056				TST61:		
4138	020056	005267	160722		INC	\$TESTN	;INCREMENT TEST NUMBER
4139					;1/EXP[AC]=FSRC=0		
4140	020062	012737	000002	001042	MOV	#2,\$#FLAG	;NO INTERRUPT, BUT FEC
4141	020070	004767	000706		JSR	R7,DVFSUB	;DO TEST
4142	020074	000100	000027		.WORD	100,27	;ACO
4143	020100	000000	000000		.WORD	0,0	;FSRC
4144	020104	000100	000027		.WORD	100,27	;RESULT
4145	020110	040000			.WORD	40000	;TEST FPS
4146	020112	140000			.WORD	140000	;RESULT FPS
4147	020114	000004			.WORD	4	;FEC
4148					;2/AC=EXP[FSRC]=0		
4149					;TRAPS ENABLED		
4150	020116	012737	000001	001042	MOV	#1,\$#FLAG	;INTERRUPT
4151	020124	004767	000652		JSR	R7,DVFSUB	;DO TEST
4152	020130	000000	000000		.WORD	0,0	;ACO
4153	020134	000100	000000		.WORD	100,0	;FSRC
4154	020140	000000	000000		.WORD	0,0	;RESULT
4155	020144	000000			.WORD	0	;TEST FPS
4156	020146	100000			.WORD	100000	;RESULT FPS
4157	020150	000004			.WORD	4	;FEC
4158					;3/FSRC>ACO=0		
4159	020152	005037	001042		CLR	\$#FLAG	;NO INTERRUPT
4160	020156	004767	000620		JSR	R7,DVFSUB	;DO TEST
4161	020162	000177	000234		.WORD	177,234	;ACO
4162	020166	004100	000000		.WORD	4100,0	;FSRC
4163	020172	000000	000000		.WORD	0,0	;RESULT
4164	020176	007400			.WORD	7400	;TEST FPS
4165	020200	007404			.WORD	7404	;RESULT FPS
4166					;4/ACO>EXP[FSRC]=0		
4167	020202	012737	000001	001042	MOV	#1,\$#FLAG	;INTERRUPT
4168	020210	004767	000566		JSR	R7,DVFSUB	;DO TEST
4169	020214	040200	104210		.WORD	40200,104210	;ACO
4170	020220	000125	025252		.WORD	125,25252	;FSRC
4171	020224	040200	104210		.WORD	40200,104210	;RESULT
4172	020230	007557			.WORD	7557	;TEST FPS
4173	020232	107557			.WORD	107557	;RESULT FPS
4174	020234	000004			.WORD	4	;FEC
4175					;5/EXP[AC]=EXP[FSRC]		
4176	020236	005037	001042		CLR	\$#FLAG	;NO INTERRUPT
4177	020242	004767	000534		JSR	R7,DVFSUB	;DO TEST
4178	020246	077760	177777		.WORD	77760,-1	;ACO
4179	020252	077760	000000		.WORD	77760,0	;FSRC
4180	020256	040200	104210		.WORD	40200,104210	;RESULT
4181	020262	007414			.WORD	7414	;TEST FPS
4182	020264	007400			.WORD	7400	;RESULT FPS
4183					;6/AC=FSRC		
4184	020266	005037	001042		CLR	\$#FLAG	;NO INTERRUPT
4185	020272	004767	000504		JSR	R7,DVFSUB	;DO TEST
4186	020276	052525	052525		.WORD	52525,52525	;ACO
4187	020302	052525	052525		.WORD	52525,52525	;FSRC
4188	020306	040200	000000		.WORD	40200,0	;RESULT
4189	020312	007400			.WORD	7400	;TEST FPS
4190	020314	007400			.WORD	7400	;RESULT FPS

4191				;7/FSRC>0<ACO, ROUND		
4192	020316	005037	001042	CLR	B#FLAG	;NO INTERRUPT
4193	020322	004767	000454	JSR	R7,DVFSUB	;DO TEST
4194	020326	077777	125252	.WORD	77777,125252	;ACO
4195	020332	040300	000000	.WORD	40300,0	;FSRC
4196	020336	077652	070707	.WORD	77652,070707	;RESULT
4197	020342	007400		.WORD	7400	;TEST FPS
4198	020344	007400		.WORD	7400	;RESULT FPS
4199				;8/AC>0<FSRC		
4200	020346	005037	001042	CLR	B#FLAG	;NO INTERRUPT
4201	020352	004767	000424	JSR	R7,DVFSUB	;DO TEST
4202	020356	055377	177777	.WORD	55377,-1	;ACO
4203	020362	055300	000000	.WORD	55300,0	;FSRC
4204	020366	040252	125252	.WORD	40252,125252	;RESULT
4205	020372	000000		.WORD	0	;TEST FPS
4206	020374	000000		.WORD	0	;RESULT FPS
4207				;9/FSRC>AC>0		
4208	020376	005037	001042	CLR	B#FLAG	;NO INTERRUPT
4209	020402	004767	000374	JSR	R7,DVFSUB	;DO TEST
4210	020406	064600	000001	.WORD	64600,1	;ACO
4211	020412	066600	000000	.WORD	66600,0	;FSRC
4212	020416	036200	000001	.WORD	36200,1	;RESULT
4213	020422	000000		.WORD	0	;TEST FPS
4214	020424	000000		.WORD	0	;RESULT FPS
4215				;10/AC>FSRC>0		
4216	020426	005037	001042	CLR	B#FLAG	;NO INTERRUPT
4217	020432	004767	000344	JSR	R7,DVFSUB	;DO TEST
4218	020436	012345	156024	.WORD	12345,156024	;ACO
4219	020442	005600	000000	.WORD	05600,0	;FSRC
4220	020446	044745	156024	.WORD	44745,156024	;RESULT
4221	020452	000017		.WORD	17	;TEST FPS
4222	020454	000000		.WORD	0	;RESULT FPS
4223				;11/FSRC<0		
4224	020456	005037	001042	CLR	B#FLAG	;NO INTERRUPT
4225	020462	004767	000314	JSR	R7,DVFSUB	;DO TEST
4226	020466	040422	101010	.WORD	40422,101010	;ACO
4227	020472	140511	101010	.WORD	140511,101010	;FSRC
4228	020476	140072	020167	.WORD	140072,20167	;RESULT
4229	020502	000057		.WORD	57	;TEST FPS
4230	020504	000050		.WORD	50	;RESULT FPS
4231				;12/AC<0		
4232	020506	005037	001042	CLR	B#FLAG	;NO INTERRUPT
4233	020512	004767	000264	JSR	R7,DVFSUB	;DO TEST
4234	020516	160077	000101	.WORD	160077,101	;ACO
4235	020522	040417	177777	.WORD	40417,-1	;FSRC
4236	020526	157651	143527	.WORD	157651,143527	;RESULT
4237	020532	000007		.WORD	7	;TEST FPS
4238	020534	000010		.WORD	10	;RESULT FPS
4239				;13/TRUNCATE TEST		
4240	020536	005037	001042	CLR	B#FLAG	;NO INTERRUPT
4241	020542	004767	000234	JSR	R7,DVFSUB	;DO TEST
4242	020546	060100	000177	.WORD	60100,177	;ACO
4243	020552	040300	000000	.WORD	40300,0	;FSRC
4244	020556	060000	000124	.WORD	60000,124	;RESULT
4245	020562	000040		.WORD	40	;TEST FPS
4246	020564	000040		.WORD	40	;RESULT FPS

```

4247      ;14/ROUND TEST
4248
4249      020566 005037 001042      CLR      @FLAG      ;NO INTERRUPT
4250      020572 004767 000204      JSR      R7,DVFSUB      ;DO TEST
4251      020576 060100 000177      .WORD    60100,177      ;ACO
4252      020602 040300 000000      .WORD    40300,0 ;FSRC
4253      020606 060000 000125      .WORD    60000,125      ;RESULT
4254      020612 000000      .WORD    0      ; TEST FPS
4255      020614 000000      .WORD    0      ;RESULT FPS
4256
4257      020616 012737 000001 001042 ;15/OVERFLOW, INTERRUPTS ENABLED
4258      020624 004767 000152      MOV      @1,@FLAG      ;INTERRUPT
4259      020630 177700 000000      JSR      R7,DVFSUB      ;DO TEST
4260      020634 000200 000000      .WORD    177700,0      ;ACO
4261      020640 137700 000000      .WORD    200,0      ;FSRC
4262      020644 001100      .WORD    137700,0      ;RESULT
4263      020646 101112      .WORD    1100      ; TEST FPS
4264      020650 000010      .WORD    101112      ;RESULT FPS
4265      .WORD    10      ;FEC
4266      020652 012737 000002 001042 ;16/OVERFLOW, TRAPS DISABLED
4267      020660 004767 000116      MOV      @2,@FLAG      ;NO INTERRUPT
4268      020664 000200 000000      JSR      R7,DVFSUB      ;DO TEST
4269      020670 177700 000000      .WORD    200,0      ;ACO
4270      020674 000000 000000      .WORD    177700,0      ;FSRC
4271      020700 041100      .WORD    0,0      ;RESULT
4272      020702 041104      .WORD    41100      ; TEST FPS
4273      020704 000010      .WORD    41104      ;RESULT FPS
4274      .WORD    10      ;FEC OVERFLOW
4275      020706 012737 000001 001042 ;17/UNDERFLOW, TRAPS ENABLED, UV RESULT
4276      020714 004767 000062      MOV      @1,@FLAG      ;INTERRUPT
4277      020720 100200 000000      JSR      R7,DVFSUB      ;DO TEST
4278      020724 040377 177777      .WORD    100200,0      ;ACO
4279      020730 100000 000001      .WORD    40377,-1      ;FSRC
4280      020734 002000      .WORD    100000,1      ;RESULT
4281      020736 102014      .WORD    2000      ; TEST FPS
4282      020740 000012      .WORD    102014      ;RESULT FPS
4283      .WORD    12      ;FEC
4284      020742 012737 000001 001042 ;18/UNDERFLOW, TRAPS ENABLED, ROUND
4285      020750 004767 000026      MOV      @1,@FLAG      ;INTERRUPT
4286      020754 030325 025252      JSR      R7,DVFSUB      ;DO TEST
4287      020760 076777 023456      .WORD    30325,25252      ;ACO
4288      020764 071525 157716      .WORD    76777,23456      ;FSRC
4289      020770 002537      .WORD    71525,157716      ;RESULT
4290      020772 102500      .WORD    2537      ; TEST FPS
4291      020774 000012      .WORD    102500      ;RESULT FPS
4292      .WORD    12      ;FEC
4293
4294      020776 000167 000242      JMP      HOP10      ;GO TO NEXT TEST
4295      ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
4296      ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
4297      ;DIVF SUBROUTINE:
4298      ;
4299      ;      ACO
4300      ;      FSRC
4301      ;      FPS BEFORE EXECUTION
4302      ;      FPS AFTER EXECUTION
      ;      (FEC)
  
```



```

4303
4304
4305
4306
4307 021002 012605
4308 021004 012737 021064 000244
4309 021012 012702 000200
4310 021016 170102
4311 021020 010504
4312 021022 062704 000004
4313 021026 172415
4314 021030 016502 000014
4315 021034 170102
4316
4317 021036 174414
4318 021040 170001
4319
4320
4321 021042 032737 000001 001042
4322 021050 001426
4323 021052 104000
4324 021054 000343
4325 021056 002013
4326
4327 021060 000167 000042
4328
4329
4330 021064 032737 000001 001042
4331 021072 001005
4332 021074 104000
4333 021076 000344
4334 021100 002013
4335
4336 021102 000167 000020
4337 021106 012604
4338 021110 005726
4339 021112 022704 021040
4340 021116 001403
4341 021120 104000
4342 021122 000345
4343 021124 002013
4344
4345
4346
4347 021126 170203
4348 021130 012702 000200
4349 021134 170102
4350 021136 012701 001126
4351 021142 174011
4352 021144 026503 000016
4353 021150 001403
4354 021152 104000
4355 021154 000346
4356 021156 002013
4357
4358 021160 010504

;
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
;
DVFSUB: MOV (SP)+,R5 ; RETURN ADDRESS TO USE AS POINTER
MOV #50$,#FPVEC ; REDIRECT TRAP VECTOR
MOV #200,R2 ; SET TO DOUBLE MODE FOR LOAD
LDFPS R2 ; LOAD FPS
MOV R5,R4 ; POINT TO FSRC DATA
ADD #4,R4
LDD (R5),ACO ; LOAD ACO WITH TEST DATA
MOV 14(R5),R2 ; GET TEST FPS
LDFPS R2 ; LOAD TEST FPS
;
; DIVF (R4),ACO ; *TEST INSTRUCTION
1$: SETF ; WAIT FOR POSSIBLE FPA TRAP.
;
; INSTRUCTION DIDNT TRAP
; BIT #1,#FLAG ; VERIFY A NO TRAP CONDITION
; BEQ 2$ ; BRANCH IF GOOD
; ERROR 2$ ; ALL ERRORS TO TRAP TO EMT VECTOR
; .WORD 343 ; UNIQUE ERROR NUMBER
; .WORD FPPERR ; ADDRESS OF ERROR MESSAGE
; INSTRUCTION SHOULD HAVE TRAPPED
; JMP 2$ ; REJOIN CODE
;
; INSTRUCTION TRAPPED
50$: BIT #1,#FLAG ; SEE IF EXPECTING A TRAP
; BNE 51$ ; BRANCH IF EXPECTING A TRAP
; ERROR 51$ ; ALL ERRORS TO TRAP TO EMT VECTOR
; .WORD 344 ; UNIQUE ERROR NUMBER
; .WORD FPPERR ; ADDRESS OF ERROR MESSAGE
; INSTRUCTION WASNT SUPPOSE TO TRAP
; JMP 2$ ; REJOIN CODE
51$: MOV (SP)+,R4 ; SEE IF PC = INSTRUCTION
TST (SP)+ ; CLEAN UP STACK
CMP #1$,R4 ;
; BEQ 2$ ; BRANCH IF GOOD COMPARE
; ERROR 2$ ; ALL ERRORS TO TRAP TO EMT VECTOR
; .WORD 345 ; UNIQUE ERROR NUMBER
; .WORD FPPERR ; ADDRESS OF ERROR MESSAGE
; PC WAS INCORRECT
;
; COMMON CODE FOR TRAP AND NO TRAP
2$: STFPS R3 ; SAVE FPS
MOV #200,R2 ; SET FPP TO DOUBLE
LDFPS R2
MOV #RECDST,R1 ; POINT TO RECEIVED DATA TABLE
STD ACO,(R1) ; SAVE ACO RESULT
CMP 16(R5),R3 ; VERIFY STATUS
; BEQ 3$ ; BRANCH IF GOOD
; ERROR 3$ ; ALL ERRORS TO TRAP TO EMT VECTOR
; .WORD 346 ; UNIQUE ERROR NUMBER
; .WORD FPPERR ; ADDRESS OF ERROR MESSAGE
; BAD FPS
3$: MOV R5,R4 ; POINT TO EXPECTED DATA

```

4359	021162	062704	000010		ADD	#10,R4		
4360	021166	004767	160736	4\$:	JSR	R7,DATVFR		;VERIFY DATA
4361	021172	005767	157642		TST	COUNT		
4362	021176	001403			BEQ	5\$		;BRANCH IF GOOD
4363	021200	104000			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
4364	021202	000347			.WORD	347		;UNIQUE ERROR NUMBER
4365	021204	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
4366								
4367	021206	005737	001042	5\$:	TST	#0FLAG		;BAD ACO
4368	021212	001002			BNE	6\$		;SEE IF NEED TO CHECK FEC
4369	021214	000165	000020		JMP	20(R5)		;BRANCH IF NEED TO CHECK
4370	021220	170301		6\$:	STST	R1		;RETURN FROM TEST
4371	021222	016504	000020		MOV	20(R5),R4		;SAVE FEC
4372	021226	020401			CMP	R4,R1		;GET FEC
4373	021230	001403			BEQ	7\$		;VERIFY FEC
4374	021232	104000			ERROR			;BRANCH IF GOOD
4375	021234	000350			.WORD	350		;ALL ERRORS TO TRAP TO EMT VECTOR
4376	021236	002013			.WORD	FPPERR		;UNIQUE ERROR NUMBER
4377								;ADDRESS OF ERROR MESSAGE
4378	021240	000165	000022	7\$:	JMP	22(R5)		;BAD FEC
4379								;RETURN FROM TEST
4380	021244							
4381	021244							
4382								
4383								
4384								
4385	021244							
4386	021244	005267	157534					
4387								
4388	021250	012737	000002	001042				
4389	021256	004767	000516					
4390	021262	000000	000000	000000				
4391	021270	000001						
4392	021272	000100	000000	000000				
4393	021300	000000						
4394	021302	000000	000000	000000				
4395	021310	000001						
4396	021312	040000						
4397	021314	140000						
4398	021316	000004						
4399								
4400	021320	012737	000001	001042				
4401	021326	004767	000446					
4402	021332	000402	000000	000000				
4403	021340	000000						
4404	021342	000000	000000	000000				
4405	021350	000000						
4406	021352	000402	000000	000000				
4407	021360	000000						
4408	021362	000200						
4409	021364	100200						
4410	021366	000004						
4411								
4412	021370	005037	001042					
4413	021374	004767	000400					
4414	021400	034300	000000	000000				

4415	021406	000001					
4416	021410	140300	000000	000000	.WORD	140300,0,0,0	;FSRC
4417	021416	000000					
4418	021420	134200	000000	000000	.WORD	134200,0,0,1	;RESULT
4419	021426	000001					
4420	021430	000200			.WORD	200	;TEST FPS
4421	021432	000210			.WORD	210	;RESULT FPS
4422					;4/TRUNCATE		
4423	021434	005037	001042		CLR	00FLAG	;NO INTERRUPT
4424	021440	004767	000334		JSR	R7,DVDSUB	;DO TEST
4425	021444	034300	000000	000000	.WORD	34300,0,0,1	;ACO
4426	021452	000001					
4427	021454	140300	000000	000000	.WORD	140300,0,0,0	;FSRC
4428	021462	000000					
4429	021464	134200	000000	000000	.WORD	134200,0,0,0	;RESULT
4430	021472	000000					
4431	021474	000240			.WORD	240	;TEST FPS
4432	021476	000250			.WORD	250	;RESULT FPS
4433					;5/ROUND NEGATIVE AC, FSRC		
4434	021500	005037	001042		CLR	00FLAG	;NO INTERRUPT
4435	021504	004767	000270		JSR	R7,DVDSUB	;DO TEST
4436	021510	177642	000000	000000	.WORD	177642,0,0,151	;ACO
4437	021516	000151					
4438	021520	166600	000000	000000	.WORD	166600,0,0,123	;FSRC
4439	021526	000123					
4440	021530	051242	000000	000000	.WORD	51242,0,0,0	;RESULT
4441	021536	000000					
4442	021540	000200			.WORD	200	;TEST FPS
4443	021542	000200			.WORD	200	;RESULT FPS
4444					;6/TRUNCATE NEAGTIVE AC, FSRC		
4445	021544	005037	001042		CLR	00FLAG	;NO INTERRUPT
4446	021550	004767	000224		JSR	R7,DVDSUB	;DO TEST
4447	021554	177642	000000	000000	.WORD	177642,0,0,151	;ACO
4448	021562	000151					
4449	021564	166600	000000	000000	.WORD	166600,0,0,123	;FSRC
4450	021572	000123					
4451	021574	051241	177777	177777	.WORD	51241,-1,-1,-1	;RESULT
4452	021602	177777					
4453	021604	000240			.WORD	240	;TEST FPS
4454	021606	000240			.WORD	240	;RESULT FPS
4455					;7/AC=FSRC		
4456	021610	005037	001042		CLR	00FLAG	;NO INTERRUPT
4457	021614	004767	000160		JSR	R7,DVDSUB	;DO TEST
4458	021620	055521	047621	100333	.WORD	55521,47621,100333,-1	;ACO
4459	021626	177777					
4460	021630	055521	047621	100333	.WORD	55521,47621,100333,-1	;FSRC
4461	021636	177777					
4462	021640	040200	000000	000000	.WORD	40200,0,0,0	;RESULT
4463	021646	000000					
4464	021650	007717			.WORD	7717	;TEST FPS
4465	021652	007700			.WORD	7700	;RESULT FPS
4466					;8/UNDERFLOW TRAPS ENABLED, UV RESULT		
4467	021654	012737	000001	001042	MOV	01,00FLAG	;INTERRUPT
4468	021662	004767	000112		JSR	R7,DVDSUB	;DO TEST
4469	021666	100200	000000	000000	.WORD	100200,0,0,0	;ACO
4470	021674	000000					

```

4471 021676 077777 000000 000000 .WORD 77777,0,0,0 ;FSRC
4472 021704 000000
4473 021706 140400 100200 100200 .WORD 140400,100200,100200,100201 ;RESULT
4474 021714 100201
4475 021716 002200 .WORD 2200 ; TEST FPS
4476 021720 102210 .WORD 102210 ;RESULT FPS
4477 021722 000012 .WORD 12 ;FEC
4478 ;9/OVERFLOW TRAPS ENABLED
4479 021724 012737 000001 001042 MOV #1,8#FLAG ; INTERRUPT
4480 021732 004767 000042 JSR R7,DVDSUB ;DO TEST
4481 021736 077000 123465 012346 .WORD 77000,123465,12346,525 ;ACO
4482 021744 000525
4483 021746 000303 000001 140000 .WORD 303,1,140000,140001 ;FSRC
4484 021754 140001
4485 021756 036650 163002 103645 .WORD 36650,163002,103645,64003 ;RESULT
4486 021764 064003
4487 021766 001700 .WORD 1700 ; TEST FPS
4488 021770 101702 .WORD 101702 ;RESULT FPS
4489 021772 000010 .WORD 10 ;FEC
4490 ;
4491 ;
4492 021774 000167 000242 JMP HOP11 ;HOP OVER SUBROUTINE
4493 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
4494 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
4495 ;
4496 ;DIVD SUBROUTINE:
4497 ;
4498 ; ACO
4499 ; FSRC
4500 ; FPS BEFORE EXECUTION
4501 ; FPS AFTER EXECUTION
4502 ; (FEC)
4503 ;
4504 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
4505 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
4506 022000 012605 DVDSUB: MOV (SP),R5 ; RETURN ADDRESS TO USE AS POINTER
4507 022002 012737 022062 000244 MOV #501,8#FPVEC ;REDIRECT TRAP VECTOR
4508 022010 012702 000200 MOV #200,R2 ;SET TO DOUBLE MODE FOR LOAD
4509 022014 170102 LDFPS R2 ;LOAD FPS
4510 022016 010504 MOV R5,R4 ;POINT TO FSRC DATA
4511 022020 062704 000010 ADD #10,R4
4512 022024 172415 LDD (R5),ACO ;LOAD ACO WITH TEST DATA
4513 022026 016502 000030 MOV 30(R5),R2 ;GET TEST FPS
4514 022032 170102 LDFPS R2 ;LOAD TEST FPS
4515 ;
4516 022034 174414 DIVD (R4),ACO ;*TEST INSTRUCTION
4517 022036 170000 1$: CFCC ;WAIT FOR POSSIBLE FPA TRAP.
4518 ;
4519 ;INSTRUCTION DIDNT TRAP
4520 022040 032737 000001 001042 BIT #1,8#FLAG ;VERIFY A NO TRAP CONDITION
4521 022046 001426 BEQ 2$ ;BRANCH IF GOOD
4522 022050 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
4523 022052 000351 .WORD 351 ;UNIQUE ERROR NUMBER
4524 022054 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
4525 ;INSTRUCTION SHOULD HAVE TRAPPED
4526 022056 000167 000042 JMP 2$ ;REJOIN CODE

```



```

4527
4528
4529 022062 032737 000001 001042 50: INSTRUCTION TRAPPED
4530 022070 001005 BIT #1,0#FLAG ;SEE IF EXPECTING A TRAP
4531 022072 104000 BNE 51: ;BRANCH IF EXPECTING A TRAP
4532 022074 000352 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
4533 022076 002013 .WORD 352 ;UNIQUE ERROR NUMBER
4534 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
4535 022100 000167 000020 ;INSTRUCTION WASNT SUPPOSE TO TRAP
4536 022104 012604 51: JMP 2: ;REJOIN CODE
4537 022106 005726 MOV (SP)+,R4 ;SEE IF PC = INSTRUCTION
4538 022110 022704 022036 TST (SP)+ ;CLEAN UP STACK
4539 022114 001403 CMP #1,R4
4540 022116 104000 BEQ 2: ;BRANCH IF GOOD COMPARE
4541 022120 000353 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
4542 022122 002013 .WORD 353 ;UNIQUE ERROR NUMBER
4543 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
4544 ;PC WAS INCORRECT
4545
4546 022124 170203 ;COMMON CODE FOR TRAP AND NO TRAP
4547 022126 012702 000200 2: STFPS R3 ;SAVE FPS
4548 022132 170102 MOV #200,R2 ;SET FPP TO DOUBLE
4549 022134 012701 001126 LDFPS R2
4550 022140 174011 MOV #RECDST,R1 ;POINT TO RECEIVED DATA TABLE
4551 022142 026503 000032 STD ACO,(R1) ;SAVE ACO RESULT
4552 022146 001403 CMP 32(R5),R3 ;VERIFY STATUS
4553 022150 104000 BEQ 3: ;BRANCH IF GOOD
4554 022152 000354 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
4555 022154 002013 .WORD 354 ;UNIQUE ERROR NUMBER
4556 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
4557 022156 010504 3: MOV R5,R4 ;BAD FPS
4558 022160 062704 000020 ADD #20,R4 ;POINT TO EXPECTED DATA
4559 022164 004767 157756 4: JSR R7,DATVER ;VERIFY DATA
4560 022170 005767 156644 TST COUNT
4561 022174 001403 BEQ 5: ;BRANCH IF GOOD
4562 022176 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
4563 022200 000355 .WORD 355 ;UNIQUE ERROR NUMBER
4564 022202 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
4565
4566 022204 005737 001042 5: TST 0#FLAG ;BAD ACO
4567 022210 001002 BNE 6: ;SEE IF NEED TO CHECK FEC
4568 022212 000165 000034 JMP 34(R5) ;BRANCH IF NEED TO CHECK
4569 022216 170301 6: STST R1 ;RETURN FROM TEST
4570 022220 016504 000034 MOV 34(R5),R4 ;SAVE FEC
4571 022224 020401 CMP R4,R1 ;GET FEC
4572 022226 001403 BEQ 7: ;VERIFY FEC
4573 022230 104000 ERROR ;BRANCH IF GOOD
4574 022232 000356 .WORD 356 ;ALL ERRORS TO TRAP TO EMT VECTOR
4575 022234 002013 .WORD FPPERR ;UNIQUE ERROR NUMBER
4576 ;ADDRESS OF ERROR MESSAGE
4577 022236 000165 000036 7: JMP 36(R5) ;BAD FEC
4578 ;RETURN FROM TEST
4579 022242
4580 022242
4581
4582
;*****
;HOP11:
;MMULF:
;TEST 63 TEST MULF

```

4583
 4584 022242
 4585 022242 005267 156536
 4586
 4587 022246 005037 001042
 4588 022252 004767 000564
 4589 022256 000000 000000
 4590 022262 000000 000000
 4591 022266 000000 000000
 4592 022272 007517
 4593 022274 007504
 4594
 4595 022276 005037 001042
 4596 022302 004767 000534
 4597 022306 000200 000000
 4598 022312 000000 000000
 4599 022316 000000 000000
 4600 022322 000013
 4601 022324 000004
 4602
 4603 022326 005037 001042
 4604 022332 004767 000504
 4605 022336 000100 000000
 4606 022342 000300 000000
 4607 022346 000000 000000
 4608 022352 007500
 4609 022354 007504
 4610
 4611 022356 005037 001042
 4612 022362 004767 000454
 4613 022366 040200 000000
 4614 022372 040177 177777
 4615 022376 040177 177777
 4616 022402 000000
 4617 022404 000000
 4618
 4619 022406 005037 001042
 4620 022412 004767 000424
 4621 022416 040177 177777
 4622 022422 040200 000000
 4623 022426 040177 177777
 4624 022432 000040
 4625 022434 000040
 4626
 4627 022436 005037 001042
 4628 022442 004767 000374
 4629 022446 040100 000000
 4630 022452 040100 000000
 4631 022456 040020 000000
 4632 022462 000012
 4633 022464 000000
 4634
 4635 022466 005037 001042
 4636 022472 004767 000344
 4637 022476 017500 000000
 4638 022502 023652 125252

```

*****
TST63:
      INC      $TESTN          ;INCREMENT TEST NUMBER
;1/AC0=FSRC=0 - INTERRUPTS DISABLED
      CLR      $FLAG          ;NO INTERRUPT
      JSR      R7,MLFSUB      ;DO TEST
      .WORD    0,0            ;ACO
      .WORD    0,0            ;FSRC
      .WORD    0,0
      .WORD    7517           ;RESULT
      .WORD    7504           ;TEST FPS
      .WORD    7504           ;RESULTANT FPS
;2/AC>FSRC=0 - INTERRUPTS ON
      CLR      $FLAG          ;NO INTERRUPT
      JSR      R7,MLFSUB      ;DO TEST
      .WORD    200,0          ;ACO
      .WORD    0,0            ;FSRC
      .WORD    0,0
      .WORD    13            ;RESULT
      .WORD    4              ;TEST FPS
      .WORD    4              ;RESULTANT FPS
;3/AC=0 FSRC>0 -
      CLR      $FLAG          ;NO INTERRUPT
      JSR      R7,MLFSUB      ;DO TEST
      .WORD    100,0          ;ACO
      .WORD    300,0          ;FSRC
      .WORD    0,0
      .WORD    7500           ;RESULT
      .WORD    7504           ;TEST FPS
      .WORD    7504           ;RESULTANT FPS
;4/AC=1 >FSRC - ROUND
      CLR      $FLAG          ;NO INTERRUPT
      JSR      R7,MLFSUB      ;DO TEST
      .WORD    40200,0        ;ACO
      .WORD    40177,-1      ;FSRC
      .WORD    40177,-1
      .WORD    0              ;RESULT
      .WORD    0              ;TEST FPS
      .WORD    0              ;RESULTANT FPS
;5/TRUNCATE
      CLR      $FLAG          ;NO INTERRUPT
      JSR      R7,MLFSUB      ;DO TEST
      .WORD    40177,-1      ;ACO
      .WORD    40200,0        ;FSRC
      .WORD    40177,-1
      .WORD    40            ;RESULT
      .WORD    40            ;TEST FPS
      .WORD    40            ;RESULTANT FPS
;6/NORMALIZE
      CLR      $FLAG          ;NO INTERRUPT
      JSR      R7,MLFSUB      ;DO TEST
      .WORD    40100,0        ;ACO
      .WORD    40100,0        ;FSRC
      .WORD    40020,0
      .WORD    12            ;RESULT
      .WORD    0              ;TEST FPS
      .WORD    0              ;RESULTANT FPS
;7/ROUND
      CLR      $FLAG          ;NO INTERRUPT
      JSR      R7,MLFSUB      ;DO TEST
      .WORD    17500,0        ;ACO
      .WORD    23652,125252   ;FSRC
  
```


4639	022506	003177	177777	.WORD	3177,-1	;RESULT
4640	022512	007417		.WORD	7417	;TEST FPS
4641	022514	007400		.WORD	7400	;RESULTANT FPS
4642				;8/AC>0>FSRC ROUND		
4643	022516	005037	001042	CLR	0#FLAG	;NO INTERRUPT
4644	022522	004767	000314	JSR	R7,MLFSUB	;DO TEST
4645	022526	040342	177777	.WORD	40342,-1	;ACO
4646	022532	176543	025252	.WORD	176543,025252	;FSRC
4647	022536	176711	067324	.WORD	176711,67324	;RESULT
4648	022542	007500		.WORD	7500	;TEST FPS
4649	022544	007510		.WORD	7510	;RESULTANT FPS
4650				;9/IAC<FSRC<0, ROUND		
4651	022546	005037	001042	CLR	0#FLAG	;NO INTERRUPT
4652	022552	004767	000264	JSR	R7,MLFSUB	;DO TEST
4653	022556	144600	000000	.WORD	144600,0	;ACO
4654	022562	154000	000000	.WORD	154000,0	;FSRC
4655	022566	060400	000000	.WORD	60400,0	;RESULT
4656	022572	000017		.WORD	17	;TEST FPS
4657	022574	000000		.WORD	0	;RESULT FPS
4658				;10/AC<FSRC, ROUND		
4659	022576	005037	001042	CLR	0#FLAG	;NO INTERRUPT
4660	022602	004767	000234	JSR	R7,MLFSUB	;DO TEST
4661	022606	060000	000000	.WORD	60000,0	;ACO
4662	022612	140377	177776	.WORD	140377,177776	;FSRC
4663	022616	160177	177776	.WORD	160177,177776	;RESULT
4664	022622	000017		.WORD	17	;TEST FPS
4665	022624	000010		.WORD	10	;RESULT FPS
4666				;11/AC>0>FSRC, TRUNCATE		
4667	022626	005037	001042	CLR	0#FLAG	;NO INTERRUPT
4668	022632	004767	000204	JSR	R7,MLFSUB	;DO TEST
4669	022636	060000	000000	.WORD	60000,0	;ACO
4670	022642	140377	177776	.WORD	140377,177776	;FSRC
4671	022646	160177	177776	.WORD	160177,177776	;RESULT
4672	022652	007547		.WORD	7547	;TEST FPS
4673	022654	007550		.WORD	7550	;RESULT FPS
4674				;12/UNDERFLOW, NO INTERRUPTS		
4675	022656	012737	000002	MOV	02,0#FLAG	;NO INTERRUPT
4676	022664	004767	000152	JSR	R7,MLFSUB	;DO TEST
4677	022670	000200	000001	.WORD	200,1	;ACO
4678	022674	000200	000001	.WORD	200,1	;FSRC
4679	022700	040200	000002	.WORD	40200,2	;RESULT
4680	022704	042117		.WORD	42117	;TEST FPS
4681	022706	142100		.WORD	142100	;RESULT FPS
4682	022710	000012		.WORD	12	;FEC
4683				;13/OVERFLOW, TRAP		
4684	022712	012737	000001	MOV	01,0#FLAG	;INTERRUPT
4685	022720	004767	000116	JSR	R7,MLFSUB	;DO TEST
4686	022724	177777	177777	.WORD	177777,-1	;ACO
4687	022730	040300	000000	.WORD	40300,0	;FSRC
4688	022734	100077	177777	.WORD	100077,-1	;RESULT
4689	022740	001117		.WORD	1117	;TEST FPS
4690	022742	101116		.WORD	101116	;RESULT FPS
4691	022744	000010		.WORD	10	;FEC
4692				;14/OVERFLOW NO TRAP		
4693	022746	012737	000002	MOV	02,0#FLAG	;NO INTERRUPT
4694	022754	004767	000062	JSR	R7,MLFSUB	;DO TEST

```

4695 022760 077700 000000 .WORD 77700,0 ;ACO
4696 022764 077700 000000 .WORD 77700,0 ;FSRC
4697 022770 000000 000000 .WORD 0,0 ;RESULT
4698 022774 040117 .WORD 40117 ;TEST FPS
4699 022776 040106 .WORD 40106 ;RESULT FPS
4700 023000 000010 .WORD 10 ;FEC
4701 ;15/UNDEFINED VARIABLE IN FSRC, TRAP ENABLED
4702 023002 012737 000001 001042 MOV #1,0#FLAG ;INTERRUPT
4703 023010 004767 000026 JSR R7,MLFSUB ;DO TEST
4704 023014 123465 000000 .WORD 123465,0 ;ACO
4705 023020 100022 000000 .WORD 100022,0 ;FSRC
4706 023024 123465 000000 .WORD 123465,0 ;RESULT
4707 023030 004000 .WORD 4000 ;TEST FPS
4708 023032 104000 .WORD 104000 ;RESULT FPS
4709 023034 000014 .WORD 14 ;FEC
4710 ;
4711 ;
4712 023036 000167 000242 JMP HOP12
4713 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
4714 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
4715 ;
4716 ; ACO
4717 ; FSRC
4718 ; FPS BEFORE EXECUTION
4719 ; FPS AFTER EXECUTION
4720 ; (FEC)
4721 ;
4722 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
4723 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
4724 ;
4725 023042 012605 MLFSUB: MOV (SP)+,R5 ; RETURN ADDRESS TO USE AS POINTER
4726 023044 012737 023124 000244 MOV #50,0#FPVEC ;REDIRECT TRAP VECTOR
4727 023052 012702 000200 MOV #200,R2 ;SET TO DOUBLE MODE FOR LOAD
4728 023056 170102 LDFPS R2 ;LOAD FPS
4729 023060 172415 LDD (R5),ACO ;LOAD ACO WITH TEST DATA
4730 023062 010504 MOV R5,R4 ;POINT TO FSRC DATA
4731 023064 062704 000004 ADD #4,R4
4732 023070 016502 000014 MOV 14(R5),R2 ;GET TEST FPS
4733 023074 170102 LDFPS R2 ;LOAD TEST FPS
4734 ;
4735 023076 171014 MULF (R4),ACO ;*TEST INSTRUCTION
4736 023100 170001 1$: SETF ;WAIT FOR POSSIBLE FPA TRAP.
4737 ;
4738 ;INSTRUCTION DIDNT TRAP
4739 023102 032737 000001 001042 BIT #1,0#FLAG ;VERIFY A NO TRAP CONDITION
4740 023110 001426 BEQ 2$ ;BRANCH IF GOOD
4741 023112 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
4742 023114 000357 .WORD 357 ;UNIQUE ERROR NUMBER
4743 023116 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
4744 ;INSTRUCTION SHOULD HAVE TRAPPED
4745 023120 000167 000042 JMP 2$ ;REJOIN CODE
4746 ;
4747 ;INSTRUCTION TRAPPED
4748 023124 032737 000001 001042 50$: BIT #1,0#FLAG ;SEE IF EXPECTING A TRAP
4749 023132 001005 BNE 51$ ;BRANCH IF EXPECTING A TRAP
4750 023134 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR

```



```
4751 023136 000360 .WORD 360 ;UNIQUE ERROR NUMBER
4752 023140 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
4753 ;INSTRUCTION WASNT SUPPOSE TO TRAP
4754 023142 000167 000020 JMP 21 ;REJOIN CODE
4755 023146 012604 511: MOV (SP)+,R4 ;SEE IF PC = INSTRUCTION
4756 023150 005726 TST (SP)+ ;CLEAN UP STACK
4757 023152 022704 023100 CMP #11,R4
4758 023156 001403 BEQ 21 ;BRANCH IF GOOD COMPARE
4759 023160 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
4760 023162 000361 .WORD 361 ;UNIQUE ERROR NUMBER
4761 023164 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
4762 ;PC WAS INCORRECT
4763
4764 ;COMMON CODE FOR TRAP AND NO TRAP
4765 023166 170203 21: STFPS R3 ;SAVE FPS
4766 023170 012702 000200 MOV #200,R2 ;SET FPP TO DOUBLE
4767 023174 170102 LDFPS R2
4768 023176 012701 001126 MOV #RECDST,R1 ;POINT TO RECEIVED DATA TABLE
4769 023202 174011 STD ACO,(R1) ;SAVE ACO RESULT
4770 023204 026503 000016 CMP 16(R5),R3 ;VERIFY STATUS
4771 023210 001403 BEQ 31 ;BRANCH IF GOOD
4772 023212 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
4773 023214 000362 .WORD 362 ;UNIQUE ERROR NUMBER
4774 023216 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
4775
4776 023220 010504 31: MOV R5,R4 ;BAD FPS
4777 023222 062704 000010 ADD #10,R4 ;POINT TO EXPECTED DATA
4778 023226 004767 156676 41: JSR R7,DATVFR ;VERIFY DATA
4779 023232 005767 155602 TST COUNT
4780 023236 001403 BEQ 51 ;BRANCH IF GOOD
4781 023240 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
4782 023242 000363 .WORD 363 ;UNIQUE ERROR NUMBER
4783 023244 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
4784
4785 023246 005737 001042 51: TST #FLAG ;BAD ACO
4786 023252 001002 BNE 61 ;SEE IF NEED TO CHECK FEC
4787 023254 000165 000020 JMP 20(R5) ;BRANCH IF NEED TO CHECK
4788 ;RETURN FROM TEST
4789 023260 170301 ;VERIFY ERROR STATUS
4790 023262 016504 000020 61: STST R1 ;SAVE FEC
4791 023266 020401 MOV 20(R5),R4 ;GET FEC
4792 023270 001403 CMP R4,R1 ;VERIFY FEC
4793 023272 104000 BEQ 71 ;BRANCH IF GOOD
4794 023274 000364 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
4795 023276 002013 .WORD 364 ;UNIQUE ERROR NUMBER
4796 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
4797 023300 000165 000022 71: JMP 22(R5) ;BAD FEC
4798 023304 ;RETURN FROM TEST
4799 023304
4800 ;*****
4801 ;*TEST 64 TEST MULF
4802 ;*****
4803 023304 TST64:
4804 023304 005267 155474 INC #TESTN ;INCREMENT TEST NUMBER
4805 023310 005037 001042 ;1/AC=0 CLR #FLAG ;NO INTERRUPT
```

```

4807 023314 004767 000554 JSR R7,MLDSUB ;DO TEST
4808 023320 000100 000000 .WORD 100,0,0,0 ;ACO
4809 023326 000000 .WORD 411,-1,0,1 ;FSRC
4810 023330 000411 177777 000000 .WORD 0,0,0,0 ;RESULT
4811 023336 000001 .WORD 200 ;TEST FPS
4812 023340 000000 000000 000000 .WORD 204 ;RESULTANT FPS
4813 023346 000000 ;2/FSRC=0
4814 023350 000200 CLR B0FLAG ;NO INTERRUPT
4815 023352 000204 JSR R7,MLDSUB ;DO TEST
4816 .WORD 7777,0,0,0 ;ACO
4817 023354 005037 001042 .WORD 0,0,0,0 ;FSRC
4818 023360 004767 000510 .WORD 0,0,0,0 ;RESULT
4819 023364 077777 000000 000000 .WORD 7700 ;TEST FPS
4820 023372 000000 000000 000000 .WORD 7704 ;RESULTANT FPS
4821 023374 000000 000000 000000 ;3/AC=1
4822 023402 000000 CLR B0FLAG ;NO INTERRUPT
4823 023404 000000 000000 000000 JSR R7,MLDSUB ;DO TEST
4824 023412 000000 .WORD 40200,0,0,0 ;ACO
4825 023414 007700 .WORD 277,-1,-1,-1 ;FSRC
4826 023416 007704 .WORD 277,-1,-1,-1 ;RESULT
4827 .WORD 7717 ;TEST FPS
4828 023420 005037 001042 .WORD 7700 ;RESULTANT FPS
4829 023424 004767 000444 ;4/AC>FSRC>0, TRUNCATE
4830 023430 040200 000000 000000 CLR B0FLAG ;NO INTERRUPT
4831 023436 000000 .WORD R7,MLDSUB ;DO TEST
4832 023440 000277 177777 177777 .WORD 65500,0,0,1 ;ACO
4833 023446 177777 .WORD 37577,-1,-1,-2 ;FSRC
4834 023450 000277 177777 177777 .WORD 65077,-1,-1,-1 ;RESULT
4835 023456 177777 .WORD 7717 ;TEST FPS
4836 023460 007717 .WORD 7700 ;RESULTANT FPS
4837 023462 007700 ;5/AC<FSRC<0
4838 CLR B0FLAG ;NO INTERRUPT
4839 023464 005037 001042 JSR R7,MLDSUB ;DO TEST
4840 023470 004767 000400 .WORD 137577,-1,-1,-2 ;ACO
4841 023474 065500 000000 000000 .WORD 165400,0,0,1 ;FSRC
4842 023502 000001 .WORD 65000,0,0,0 ;RESULT
4843 023504 037577 177777 177777 .WORD 7717 ;TEST FPS
4844 023512 177776 .WORD 7700 ;RESULTANT FPS
4845 023514 065077 177777 177777 ;6/AC>FSRC>0
4846 023522 177777 CLR B0FLAG ;NO INTERRUPT
4847 023524 007717 JSR R7,MLDSUB ;DO TEST
4848 023526 007700 .WORD 137577,-1,-1,-2 ;ACO
4849 .WORD 165400,0,0,1 ;FSRC
4850 023530 005037 001042 .WORD 65000,0,0,0 ;RESULT
4851 023534 004767 000334 .WORD 7717 ;TEST FPS
4852 023540 137577 177777 177777 .WORD 7700 ;RESULTANT FPS
4853 023546 177776 .WORD 7717 ;TEST FPS
4854 023550 165400 000000 000000 .WORD 7700 ;RESULTANT FPS
4855 023556 000001 .WORD 7717 ;TEST FPS
4856 023560 065000 000000 000000 .WORD 7700 ;RESULTANT FPS
4857 023566 000000 ;6/AC>FSRC>0
4858 023570 007717 CLR B0FLAG ;NO INTERRUPT
4859 023572 007700 JSR R7,MLDSUB ;DO TEST
4860 .WORD 137577,-1,-1,-2 ;ACO
4861 023574 005037 001042 .WORD 165400,0,0,1 ;FSRC
4862 023600 004767 000270 .WORD 65000,0,0,0 ;RESULT

```



```

4863 023604 017500 000000 000000 .WORD 17500,0,0,0 ;ACO
4864 023612 000000
4865 023614 123652 125252 125252 .WORD 123652,125252,125252,125252 ;FSRC
4866 023622 125252
4867 023624 103177 177777 177777 .WORD 103177,-1,-1,-1 ;RESULT
4868 023632 177777
4869 023634 000200 .WORD 200 ; TEST FPS
4870 023636 000210 .WORD 210 ;RESULTANT FPS
4871
4872 023640 005037 001042 ;7/UNDERFLOW, TRAPS DISABLED
4873 023644 004767 000224 CLR 00FLAG ;NO INTERRUPT
4874 023650 000300 000000 000000 JSR R7,MLDSUB ;DO TEST
4875 023656 000252 .WORD 300,0,0,252 ;ACO
4876 023660 000377 000001 000002 .WORD 377,1,2,3 ;FSRC
4877 023666 000003
4878 023670 000000 000000 000000 .WORD 0,0,0,0 ;RESULT
4879 023676 000000
4880 023700 005740 .WORD 5740 ; TEST FPS
4881 023702 005744 .WORD 5744 ;RESULT FPS
4882
4883 023704 012737 000001 001042 ;8/UNDERFLOW, TRAP ENABLED
4884 023712 004767 000156 MOV 01,00FLAG ;INTERRUPT
4885 023716 100277 000001 000002 JSR R7,MLDSUB ;DO TEST
4886 023724 177777 .WORD 100277,1,2,-1 ;ACO
4887 023726 100300 000001 000001 .WORD 100300,1,1,1 ;FSRC
4888 023734 000001
4889 023736 040417 040001 077403 .WORD 40417,40001,77403,0 ;RESULT
4890 023744 000000
4891 023746 002217 .WORD 2217 ; TEST FPS
4892 023750 102200 .WORD 102200 ;RESULT FPS
4893 023752 000012 .WORD 12 ;FEC
4894
4895 023754 005037 001042 ;9/OVERFLOW, TRAPS DISABLED
4896 023760 004767 000110 CLR 00FLAG ;NO INTERRUPT
4897 023764 177777 177777 177777 JSR R7,MLDSUB ;DO TEST
4898 023772 177777 .WORD -1,-1,-1,-1 ;ACO
4899 023774 040200 177777 177777 .WORD 40200,-1,-1,-1 ;FSRC
4900 024002 177777
4901 024004 000000 000000 000000 .WORD 0,0,0,0 ;RESULT
4902 024012 000000
4903 024014 006740 .WORD 6740 ; TEST FPS
4904 024016 006746 .WORD 6746 ;RESULT FPS
4905
4906 024020 012737 000001 001042 ;10/OVERFLOW, TRAPS ENABLED
4907 024026 004767 000042 MOV 01,00FLAG ;INTERRUPT
4908 024032 157700 025252 025252 JSR R7,MLDSUB ;DO TEST
4909 024040 025252 .WORD 157700,25252,25252,25252 ;ACO
4910 024042 167700 000000 000000 .WORD 167700,0,0,0 ;FSRC
4911 024050 000000
4912 024052 007420 017777 117777 .WORD 7420,017777,117777,117777 ;RESULT
4913 024060 117777
4914 024062 001240 .WORD 1240 ; TEST FPS
4915 024064 101242 .WORD 101242 ;RESULT FPS
4916 024066 000010 .WORD 10 ;FEC
4917
4918

```

```

4919 024070 000167 000242          JMP      HOP13
4920
4921      ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
4922      ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
4923      ;
4924      ;          ACO
4925      ;          FSRC
4926      ;          FPS BEFORE EXECUTION
4927      ;          FPS AFTER EXECUTION
4928      ;          (FEC)
4929      ;
4930      ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
4931      ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
4932      ;
4933 024074 012605          MLDSUB: MOV      (SP),R5          ; RETURN ADDRESS TO USE AS POINTER
4934 024076 012737 024156 000244      MOV      @501,@FPVEC      ; REDIRECT TRAP VECTOR
4935 024104 012702 000200      MOV      @200,R2          ; SET TO DOUBLE MODE FOR LOAD
4936 024110 170102          LDFPS      R2          ; LOAD FPS
4937 024112 172415          LDD        (R5),ACO          ; LOAD ACO WITH TEST DATA
4938 024114 010501          MOV      R5,R1          ; POINT TO FSRC DATA
4939 024116 062701 000010          ADD      @10,R1
4940 024122 016502 000030          MOV      30(R5),R2          ; GET TEST FPS
4941 024126 170102          LDFPS      R2          ; LOAD TEST FPS
4942      ;
4943 024130 171011          ;          MULD      (R1),ACO          ; *TEST INSTRUCTION
4944 024132 170011          1$:      SETD          ; WAIT FOR POSSIBLE FPA TRAP.
4945      ;
4946      ; INSTRUCTION DIDNT TRAP
4947 024134 032737 000001 001042      BIT      @1,@FLAG          ; VERIFY A NO TRAP CONDITION
4948 024142 001426          BEQ        2$          ; BRANCH IF GOOD
4949 024144 104000          ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
4950 024146 000365          .WORD      365          ; UNIQUE ERROR NUMBER
4951 024150 002013          .WORD      FPPERR          ; ADDRESS OF ERROR MESSAGE
4952      ;
4953 024152 000167 000042          JMP      2$          ; INSTRUCTION SHOULD HAVE TRAPPED
4954      ;
4955      ; INSTRUCTION TRAPPED
4956 024156 032737 000001 001042      50$: BIT      @1,@FLAG          ; SEE IF EXPECTING A TRAP
4957 024164 001005          BNE        51$          ; BRANCH IF EXPECTING A TRAP
4958 024166 104000          ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
4959 024170 000366          .WORD      366          ; UNIQUE ERROR NUMBER
4960 024172 002013          .WORD      FPPERR          ; ADDRESS OF ERROR MESSAGE
4961      ;
4962 024174 000167 000020          ; INSTRUCTION WASNT SUPPOSE TO TRAP
4963 024200 012604          ; REJOIN CODE
4964 024202 005726          51$: MOV      (SP),R4          ; SEE IF PC = INSTRUCTION
4965 024204 022704 024132          TST      (SP),R4          ; CLEAN UP STACK
4966 024210 001403          CMP        @1,R4          ;
4967 024212 104000          BEQ        2$          ; BRANCH IF GOOD COMPARE
4968 024214 000367          ERROR          ; ALL ERRORS TO TRAP TO EMT VECTOR
4969 024216 002013          .WORD      367          ; UNIQUE ERROR NUMBER
4970      ;
4971      ; PC WAS INCORRECT
4972      ;
4973 024220 170203          ; COMMON CODE FOR TRAP AND NO TRAP
4974 024222 012702 000200          2$: STFPS      R3          ; SAVE FPS
          MOV      @200,R2          ; SET FPP TO DOUBLE
  
```


4975	024226	170102		LDFPS	R2		
4976	024230	012701	001126	MOV	#RECDST,R1		;POINT TO RECEIVED DATA TABLE
4977	024234	174011		STD	ACO,(R1)		;SAVE ACO RESULT
4978	024236	026503	000032	CMP	32(R5),R3		;VERIFY STATUS
4979	024242	001403		BEG	31		;BRANCH IF GOOD
4980	024244	104000		ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
4981	024246	000370		.WORD	370		;UNIQUE ERROR NUMBER
4982	024250	002013		.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
4983							
4984	024252	010504		31:	MOV	R5,R4	;BAD FPS
4985	024254	062704	000020	ADD	#20,R4		;POINT TO EXPECTED DATA
4986	024260	004767	155662	41:	JSR	R7,DATVER	
4987	024264	005767	154550	TST	COUNT		;VERIFY DATA
4988	024270	001403		BEG	51		
4989	024272	104000		ERROR			;BRANCH IF GOOD
4990	024274	000371		.WORD	371		;ALL ERRORS TO TRAP TO EMT VECTOR
4991	024276	002013		.WORD	FPPERR		;UNIQUE ERROR NUMBER
4992							;ADDRESS OF ERROR MESSAGE
4993	024300	005737	001042	51:	TST	#FLAG	;BAD ACO
4994	024304	001002		BNE	61		;SEE IF NEED TO CHECK FEC
4995	024306	000165	000034	JMP	34(R5)		;BRANCH IF NEED TO CHECK
4996				;VERIFY	ERROR STATUS		;RETURN FROM TEST
4997	024312	170301		61:	STST	R1	
4998	024314	016504	000034	MOV	34(R5),R4		;SAVE FEC
4999	024320	020401		CMP	R4,R1		;GET FEC
5000	024322	001403		BEG	71		;VERIFY FEC
5001	024324	104000		ERROR			;BRANCH IF GOOD
5002	024326	000372		.WORD	372		;ALL ERRORS TO TRAP TO EMT VECTOR
5003	024330	002013		.WORD	FPPERR		;UNIQUE ERROR NUMBER
5004							;ADDRESS OF ERROR MESSAGE
5005	024332	000165	000036	71:	JMP	36(R5)	;BAD FEC
5006	024336			HOP13:			;RETURN FROM TEST
5007	024336			MMODF:			
5008				;;*****			
5009				;+TEST 65	TEST MODF		
5010				;;*****			
5011	024336			TST65:			
5012	024336	005267	154442	INC	#TESTN		;INCREMENT TEST NUMBER
5013				;1/AC=0	FSRC=0		
5014	024342	005037	001042	CLR	#FLAG		;NO INTERRUPT
5015	024346	004767	000554	JSR	R7,MDFSUB		;DO TEST
5016	024352	000100	000000	.WORD	100,0	ACO	
5017	024356	012346	177777	.WORD	12346,-1		;FSRC
5018	024362	000000	000000	.WORD	0,0		;FRACTIONAL RESULT
5019	024366	000000	000000	.WORD	0,0		;INTEGER RESULT
5020	024372	000013		.WORD	13		;TEST FPS
5021	024374	000004		.WORD	4		;RESULTANT FPS
5022				;2/FSRC=0			
5023	024376	005037	001042	CLR	#FLAG		;NO INTERRUPT
5024	024402	004767	000520	JSR	R7,MDFSUB		;DO TEST
5025	024406	012356	177777	.WORD	12356,-1	ACO	
5026	024412	000000	000000	.WORD	0,0	FSRC	
5027	024416	000000	000000	.WORD	0,0		;FRACTIONAL RESULT
5028	024422	000000	000000	.WORD	0,0		;INTEGER RESULT
5029	024426	000003		.WORD	3		;TEST FPS
5030	024430	000004		.WORD	4		;RESULTANT FPS

5031				:3/AC=0		
5032	024432	005037	001042	CLR	B#FLAG	:NO INTERRUPT
5033	024436	004767	000464	JSR	R7,MDFSUB	:DO TEST
5034	024442	000000	000000	.WORD	0.0	:ACO
5035	024446	177777	177777	.WORD	1.-1	:FSRC
5036	024452	000000	000000	.WORD	0.0	:FRACTIONAL RESULT
5037	024456	000000	000000	.WORD	0.0	:INTEGER RESULT
5038	024462	007500		.WORD	7500	:TEST FPS
5039	024464	007504		.WORD	7504	:RESULT FPS
5040				:4/AC>FSRC>0		
5041	024466	005037	001042	CLR	B#FLAG	:NO INTERRUPT
5042	024472	004767	000430	JSR	R7,MDFSUB	:DO TEST
5043	024476	046252	125252	.WORD	46252.125252	:ACO
5044	024502	040300	000000	.WORD	40300.0	:FSRC
5045	024506	000000	000000	.WORD	0.0	:FRACTIONAL RESULT
5046	024512	046377	177777	.WORD	46377.-1	:INTEGER RESULT
5047	024516	000013		.WORD	13	:TEST FPS
5048	024520	000004		.WORD	4	:RESULTANT FPS
5049				:5/AC>FSRC>0		
5050	024522	005037	001042	CLR	B#FLAG	:NO INTERRUPT
5051	024526	004767	000374	JSR	R7,MDFSUB	:DO TEST
5052	024532	077652	125252	.WORD	77652.125252	:ACO
5053	024536	040300	000000	.WORD	40300.0	:FSRC
5054	024542	000000	000000	.WORD	0.0	:FRACTIONAL RESULT
5055	024546	077777	177777	.WORD	77777.-1	:INTEGER RESULT
5056	024552	000000		.WORD	0	:TEST FPS
5057	024554	000004		.WORD	4	:RESULTANT FPS
5058						
5059				:6/AC>0<FSRC, INTEGERS		
5060	024556	005037	001042	CLR	B#FLAG	:NO INTERRUPT
5061	024562	004767	000340	JSR	R7,MDFSUB	:DO TEST
5062	024566	060600	000000	.WORD	60600.0	:ACO
5063	024572	147400	025700	.WORD	147400.25700	:FSRC
5064	024576	000000	000000	.WORD	0.0	:FRACTIONAL RESULT
5065	024602	170000	025700	.WORD	170000.25700	:INTEGER RESULT
5066	024606	007400		.WORD	7400	:TEST FPS
5067	024610	007404		.WORD	7404	:RESULT FPS
5068				:7/AC<0<FSRC, FRACTIONAL		
5069	024612	005037	001042	CLR	B#FLAG	:NO INTERRUPT
5070	024616	004767	000304	JSR	R7,MDFSUB	:DO TEST
5071	024622	100227	177777	.WORD	100227.-1	:ACO
5072	024626	044025	025252	.WORD	44025.25252	:FSRC
5073	024632	104061	021251	.WORD	104061.21251	:FRACTIONAL RESULT
5074	024636	000000	000000	.WORD	0.0	:INTEGER RESULT
5075	024642	000000		.WORD	0	:TEST FPS
5076	024644	000010		.WORD	10	:RESULT FPS
5077				:8/AC<0>FSRC, TRUNCATE		
5078	024646	005037	001042	CLR	B#FLAG	:NO INTERRUPT
5079	024652	004767	000250	JSR	R7,MDFSUB	:DO TEST
5080	024656	046252	125252	.WORD	46252.125252	:ACO
5081	024662	040300	000000	.WORD	40300.0	:FSRC
5082	024666	000000	000000	.WORD	0.0	:FRACTIONAL RESULT
5083	024672	046377	177777	.WORD	46377.-1	:INTEGER RESULT
5084	024676	000053		.WORD	53	:TEST FPS
5085	024700	000044		.WORD	44	:RESULT FPS
5086				:9/ROUND INTEGER		


```

5087 024702 005037 001042      CLR      @FLAG
5088 024706 004767 000214      JSR      R7,MDFSUB
5089 024712 046252 125252      .WORD    46252,125252
5090 024716 040300 000000      .WORD    40300,0
5091 024722 000000 000000      .WORD    0,0
5092 024726 046377 177777      .WORD    46377,-1
5093 024732 000013          .WORD    13
5094 024734 000004          .WORD    4
5095                                ;10/TRUNCATE FRACTION
5096 024736 005037 001042      CLR      @FLAG
5097 024742 004767 000160      JSR      R7,MDFSUB
5098 024746 040777 177777      .WORD    40777,-1
5099 024752 040200 000000      .WORD    40200,0
5100 024756 040177 177770      .WORD    40177,177770
5101 024762 040740 000000      .WORD    40740,0
5102 024766 000000          .WORD    0
5103 024770 000000          .WORD    0
5104                                ;11/ROUND INTEGER
5105 024772 005037 001042      CLR      @FLAG
5106 024776 004767 000124      JSR      R7,MDFSUB
5107 025002 000000 000000      .WORD    0,0
5108 025006 000000 000000      .WORD    0,0
5109 025012 000000 000000      .WORD    0,0
5110 025016 000000 000000      .WORD    0,0
5111 025022 000000          .WORD    0
5112 025024 000004          .WORD    4
5113                                ;12/ROUND FRACTION
5114 025026 005037 001042      CLR      @FLAG
5115 025032 004767 000070      JSR      R7,MDFSUB
5116 025036 040225 125252      .WORD    40225,125252
5117 025042 066652 052525      .WORD    66652,52525
5118 025046 000000 000000      .WORD    0,0
5119 025052 066707 025160      .WORD    66707,25160
5120 025056 007027          .WORD    7027
5121 025060 007004          .WORD    7004
5122                                ;/OVERFLOW
5123 025062 012737 000001 001042  MOV      @1,@FLAG
5124 025070 004767 000032      JSR      R7,MDFSUB
5125 025074 076000 000000      .WORD    76000,0
5126 025100 076000 000000      .WORD    76000,0
5127 025104 000000 000000      .WORD    0,0
5128 025110 033600 000000      .WORD    33600,0
5129 025114 001000          .WORD    1000
5130 025116 101006          .WORD    101006
5131 025120 000010          .WORD    10
5132                                ;
5133 025122 000167 000310      JMP      HOP14
5134
5135      ;*****
5136      ;*****
5137      ;
5138      ;
5139      ;
5140      ;
5141      ;
5142      ;

```

ACO
 FSRC
 FRACTIONAL RESULT
 INTEGER RESULT
 FPS BEFORE EXECUTION

5199	025304	104000			ERROR						
5200	025306	000376			.WORD	376					
5201	025310	002013			.WORD	FPPERR					
5202											
5203	025312	010504			3\$: MOV	R5,R4					
5204	025314	062704	000010		ADD	#10,R4					
5205	025320	004767	154604		4\$: JSR	R7,DATVFR					
5206	025324	005767	153510		TST	COUNT					
5207	025330	001403			BEQ	5\$					
5208	025332	104000			ERROR						
5209	025334	000377			.WORD	377					
5210	025336	002013			.WORD	FPPERR					
5211											
5212											
5213	025340	174111			5\$: SAVE INTEGER RESULT						
5214	025342	010504			STD	AC1,(R1)					
5215	025344	062704	000014		MOV	R5,R4					
5216	025350	004767	154554		ADD	#14,R4					
5217	025354	005767	153460		JSR	R7,DATVFR					
5218	025360	001403			TST	COUNT					
5219	025362	104000			BEQ	6\$					
5220	025364	000400			ERROR						
5221	025366	002013			.WORD	400					
5222					.WORD	FPPERR					
5223	025370	005737	001042		6\$: TST	#0FLAG					
5224	025374	001002			BNE	7\$					
5225	025376	000165	000024		JMP	24(R5)					
5226	025402	170301			7\$: STST	R1					
5227	025404	016504	000024		MOV	24(R5),R4					
5228	025410	020401			CMP	R4,R1					
5229	025412	001403			BEQ	8\$					
5230	025414	104000			ERROR						
5231	025416	000401			.WORD	401					
5232	025420	002013			.WORD	FPPERR					
5233											
5234	025422	000165	000026		8\$: JMP	26(R5)					
5235											
5236	025426	177777	177777	177777	MODGAR: .WORD	-1,-1,-1,-1					
5237	025434	177777									
5238	025436										
5239											
5240	025436										
5241											
5242											
5243											
5244	025436										
5245	025436	005267	153342		TST66:						
5246					INC	\$TESTN					
5247	025442	005037	001042		1/AC>FSRC=0						
5248	025446	004767	001164		CLR	#0FLAG					
5249	025452	012345	177777	177777	JSR	R7,MODSUB					
5250	025460	177777			.WORD	12345,-1,-1,-1					
5251	025462	000100	000000	000000	.WORD	100,0,0,0					
5252	025470	000000			.WORD	0,0,0,0					
5253	025472	000000	000000	000000							
5254	025500	000000									

```

;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;BAD FPS
;POINT TO EXPECTED DATA
;VERIFY DATA
;BRANCH IF GOOD
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;BAD ACO
;SAVE AC1 RESULT
;POINT TO EXPECTED
;VERIFY DATA
;BRANCH IF GOOD
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;BAD AC1
;SEE IF NEED TO CHECK FEC
;BRANCH IF NEED TO CHECK
;RETURN FROM TEST
;SAVE FEC
;GET FEC
;VERIFY FEC
;BRANCH IF GOOD
;ALL ERRORS TO TRAP TO EMT VECTOR
;UNIQUE ERROR NUMBER
;ADDRESS OF ERROR MESSAGE
;BAD FEC
;RETURN FROM TEST
;KNOWN DATA FOR AC1
;INCREMENT TEST NUMBER
;NO INTERRUPT
;DO TEST
;ACO
;FSRC
;FRACTIONAL RESULT

```

```

5255 025502 000000 000000 000000 .WORD 0,0,0,0 ;INTEGER RESULT
5256 025510 000000
5257 025512 000200 .WORD 200 ; TEST FPS
5258 025514 000204 .WORD 204 ;RESULTANT FPS
5259 ;2/AC=0<FSRC
5260 025516 005037 001042 CLR B#FLAG ;NO INTERRUPT
5261 025522 004767 001110 JSR R7,MDDSUB ;DO TEST
5262 025526 000000 000000 000000 .WORD 0,0,0,0 ;ACO
5263 025534 000000
5264 025536 001234 177777 000000 .WORD 1234,-1,0,0 ;FSRC
5265 025544 000000
5266 025546 000000 000000 000000 .WORD 0,0,0,0 ;FRACTIONAL RESULT
5267 025554 000000
5268 025556 000000 000000 000000 .WORD 0,0,0,0 ;INTEGER RESULT
5269 025564 000000
5270 025566 007717 .WORD 7717 ; TEST FPS
5271 025570 007704 .WORD 7704 ;RESULTANT FPS
5272 ;3/AC>FSRC>0
5273 025572 005037 001042 CLR B#FLAG ;NO INTERRUPT
5274 025576 004767 001034 JSR R7,MDDSUB ;DO TEST
5275 025602 056252 125252 125252 .WORD 56252,125252,125252,125250 ;ACO
5276 025610 125250
5277 025612 040300 000000 000000 .WORD 40300,0,0,0 ;FSRC
5278 025620 000000
5279 025622 000000 000000 000000 .WORD 0,0,0,0 ;FRACTIONAL RESULT
5280 025630 000000
5281 025632 056377 177777 177777 .WORD 56377,-1,-1,-4 ;INTEGER RESULT
5282 025640 177774
5283 025642 000213 .WORD 213 ; TEST FPS
5284 025644 000204 .WORD 204 ;RESULTANT FPS
5285 ;4/AC<0>FSRC
5286 025646 005037 001042 CLR B#FLAG ;NO INTERRUPT
5287 025652 004767 000760 JSR R7,MDDSUB ;DO TEST
5288 025656 140240 000000 000000 .WORD 140240,0,0,0 ;ACO
5289 025664 000000
5290 025666 063714 146314 133572 .WORD 63714,146314,133572,167737 ;FSRC
5291 025674 167737
5292 025676 000000 000000 000000 .WORD 0,0,0,0 ;FRACTIONAL RESULT
5293 025704 000000
5294 025706 163777 177777 162531 .WORD 163777,-1,162531,125726 ;INTEGER RESULT
5295 025714 125726
5296 025716 000210 .WORD 210 ; TEST FPS
5297 025720 000204 .WORD 204 ;RESULTANT FPS
5298 ;5/AC>FSRC>0
5299 025722 005037 001042 CLR B#FLAG ;NO INTERRUPT
5300 025726 004767 000704 JSR R7,MDDSUB ;DO TEST
5301 025732 056200 000000 000000 .WORD 56200,0,0,1 ;ACO
5302 025740 000001
5303 025742 040340 000000 000000 .WORD 40340,0,0,0 ;FSRC
5304 025750 000000
5305 025752 000000 000000 000000 .WORD 0,0,0,0 ;FRACTIONAL RESULT
5306 025760 000000
5307 025762 056340 000000 000000 .WORD 56340,0,0,1 ;INTEGER RESULT
5308 025770 000001
5309 025772 000213 .WORD 213 ; TEST FPS
5310 025774 000204 .WORD 204 ;RESULTANT FPS
  
```



```

5311
5312 025776 005037 001042 ;6/TRUNCATE
5313 026002 004767 000630 CLR 0#FLAG ;NO INTERRUPT
5314 026006 056252 125252 125252 JSR R7,MDDSUB ;DO TEST
5315 026014 125252 .WORD 56252,125252,125252,125252 ;ACO
5316 026016 040300 000000 000000 .WORD 40300,0,0,0 ;FSRC
5317 026024 000000 .WORD 0,0,0,0 ;FRACTIONAL RESULT
5318 026026 000000 000000 000000 .WORD 56377,-1,-1,-1 ;INTEGER RESULT
5319 026034 000000 177777 177777 .WORD 253 ;TEST FPS
5320 026036 056377 177777 177777 .WORD 244 ;RESULT FPS
5321 026044 177777 ;7/TRUNCATE FRACTION
5322 026046 000253 CLR 0#FLAG ;NO INTERRUPT
5323 026050 000244 JSR R7,MDDSUB ;DO TEST
5324 .WORD 23252,125252,125252,125252 ;ACO
5325 026052 005037 001042 .WORD 40300,0,0,0 ;FSRC
5326 026056 004767 000554 .WORD 23377,-1,-1,-1 ;FRACTIONAL RESULT
5327 026062 023252 125252 125252 .WORD 0,0,0,0 ;INTEGER RESULT
5328 026070 125252 .WORD 253 ;TEST FPS
5329 026072 040300 000000 000000 .WORD 240 ;RESULT FPS
5330 026100 000000 ;8/ROUND INTEGER
5331 026102 023377 177777 177777 CLR 0#FLAG ;NO INTERRUPT
5332 026110 177777 JSR R7,MDDSUB ;DO TEST
5333 026112 000000 000000 000000 .WORD 76600,0,0,125252 ;ACO
5334 026120 000000 .WORD 40300,0,0,0 ;FSRC
5335 026122 000253 .WORD 0,0,0,0 ;FRACTIONAL RESULT
5336 026124 000240 .WORD 76700,0,0,-1 ;INTEGER RESULT
5337 ;9/ROUND THROUGH FRACTION
5338 026126 005037 001042 CLR 0#FLAG ;NO INTERRUPT
5339 026132 004767 000500 JSR R7,MDDSUB ;DO TEST
5340 026136 076600 000000 000000 .WORD 40300,0,0,0 ;FSRC
5341 026144 125252 .WORD 0,0,0,0 ;FRACTIONAL RESULT
5342 026146 040300 000000 000000 .WORD 76700,0,0,-1 ;INTEGER RESULT
5343 026154 000000 .WORD 200 ;TEST FPS
5344 026156 000000 000000 000000 .WORD 204 ;RESULT FPS
5345 026164 000000 ;/OVERFLOW, TRAPS ENABLED
5346 026166 076700 000000 000000 MOV 01,0#FLAG ;INTERRUPT
5347 026174 177777 JSR R7,MDDSUB ;DO TEST
5348 026176 000200 .WORD -1,-1,-1,-1 ;ACO
5349 026200 000204
5350
5351 026202 005037 001042 ;9/ROUND THROUGH FRACTION
5352 026206 004767 000424 CLR 0#FLAG ;NO INTERRUPT
5353 026212 041525 052525 052525 JSR R7,MDDSUB ;DO TEST
5354 026220 052525 .WORD 41525,052525,52525,52525 ;ACO
5355 026222 040300 000000 000000 .WORD 40300,0,0,0 ;FSRC
5356 026230 000000 .WORD 40177,-1,-1,177740 ;FRACTIONAL RESULT
5357 026232 040177 177777 177777 .WORD 41636,0,0,0 ;INTEGER RESULT
5358 026240 177740 .WORD 7700 ;TEST FPS
5359 026242 041636 000000 000000 .WORD 7700 ;RESULT FPS
5360 026250 000000 ;/OVERFLOW, TRAPS ENABLED
5361 026252 007700 MOV 01,0#FLAG ;INTERRUPT
5362 026254 007700 JSR R7,MDDSUB ;DO TEST
5363 .WORD -1,-1,-1,-1 ;ACO
5364 026256 012737 000001 001042
5365 026264 004767 000346
5366 026270 177777 177777 177777
  
```

```

5367 026276 177777
5368 026300 040400 000000 000000 .WORD 40400,0,0,0 ;FSRC
5369 026306 000000
5370 026310 000000 000000 000000 .WORD 0,0,0,0 ;FRACTIONAL RESULT
5371 026316 000000
5372 026320 100177 177777 177777 .WORD 100177,-1,-1,-1 ;INTEGER RESULT
5373 026326 177777
5374 026330 007700 .WORD 7700 ; TEST FPS
5375 026332 107706 .WORD 107706 ;RESULT FPS
5376 026334 000010 .WORD 10 ;FEC
5377
; /INTEGER CHOPPED TO 56 BITS
5378 026336 005037 001042 CLR 00FLAG ;NO INTERRUPT
5379 026342 004767 000270 JSR R7,MDDSUB ;DO TEST
5380 026346 056700 000000 000000 .WORD 56700,0,0,-1 ;ACO
5381 026354 177777
5382 026356 044440 177777 177777 .WORD 44440,-1,-1,-1 ;FSRC
5383 026364 177777
5384 026366 000000 000000 000000 .WORD 0,0,0,0 ;FRACTIONAL RESULT
5385 026374 000000
5386 026376 063161 100000 000001 .WORD 63161,100000,1,40775 ;INTEGER RESULT
5387 026404 040775
5388 026406 000200 .WORD 200 ; TEST FPS
5389 026410 000204 .WORD 204 ;RESULT FPS
5390
; /OVERFLOW, TRAPS DISABLED
5391 026412 012737 000002 001042 MOV 02,00FLAG ;NO INTERRUPT
5392 026420 004767 000212 JSR R7,MDDSUB ;DO TEST
5393 026424 066600 000000 000000 .WORD 66600,0,0,0 ;ACO
5394 026432 000000
5395 026434 066600 000000 000000 .WORD 66600,0,0,0 ;FSRC
5396 026442 000000
5397 026444 000000 000000 000000 .WORD 0,0,0,0 ;FRACTIONAL RESULT
5398 026452 000000
5399 026454 015200 000000 000000 .WORD 15200,0,0,0 ;INTEGER RESULT
5400 026462 000000
5401 026464 047700 .WORD 47700 ; TEST FPS
5402 026466 147706 .WORD 147706 ;RESULT FPS
5403 026470 000010 .WORD 10 ;FEC
5404
; /UNDERFLOW, TRAPS DISABLED
5405 026472 012737 000002 001042 MOV 02,00FLAG ;NO INTERRUPT
5406 026500 004767 000132 JSR R7,MDDSUB ;DO TEST
5407 026504 100277 000001 000002 .WORD 100277,1,2,-1 ;ACO
5408 026512 177777
5409 026514 100300 000001 000001 .WORD 100300,1,1,1 ;FSRC
5410 026522 000001
5411 026524 000000 000000 000000 .WORD 0,0,0,0 ;FRACTIONAL RESULT
5412 026532 000000
5413 026534 000000 000000 000000 .WORD 0,0,0,0 ;INTEGER RESULT
5414 026542 000000
5415 026544 005200 .WORD 5200 ; TEST FPS
5416 026546 005204 .WORD 5204 ;RESULT FPS
5417 026550 000010 .WORD 10 ;FEC
5418
; /UNDERFLOW TRAPS ENABLED, UV AS RESULT
5419 026552 012737 000001 001042 MOV 01,00FLAG ;INTERRUPT
5420 026560 004767 000052 JSR R7,MDDSUB ;DO TEST
5421 026564 100277 000001 000002 .WORD 100277,1,2,-1 ;ACO
5422 026572 177777

```



```

5423 026574 100300 000001 000001 .WORD 100300,1,1,1 ;FSRC
5424 026602 000001
5425 026604 040417 040001 077403 .WORD 40417,40001,77403,0 ;FRACTIONAL RESULT
5426 026612 000000
5427 026614 000000 000000 000000 .WORD 0,0,0,0 ;INTEGER RESULT
5428 026622 000000
5429 026624 002200 .WORD 2200 ; TEST FPS
5430 026626 102200 .WORD 102200 ;RESULT FPS
5431 026630 000012 .WORD 12 ;FEC
5432
5433
5434 026632 000167 000300 JMP HOP15 ;JUMP OVER SUBROUTINE
5435 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
5436 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
5437
5438
5439
5440
5441
5442
5443
5444
5445
5446
5447
5448
5449 026636 012605 MODSUB: MOV (SP)+,R5 ; RETURN ADDRESS TO USE AS POINTER
5450 026640 012737 026726 000244 MOV #50#,R5 ;REDIRECT TRAP VECTOR
5451 026646 012702 000200 MOV #200,R2 ;SET TO DOUBLE MODE FOR LOAD
5452 026652 170102 LDFPS R2 ;LOAD FPS
5453 026654 172415 LDD (R5),AC0 ;LOAD AC0 WITH TEST DATA
5454 026656 012701 025426 MOV #MODGAR,R1 ;LOAD KNOWN INTO AC1
5455 026662 172511 LDD (R1),AC1
5456 026664 010501 MOV R5,R1 ;POINT TO FSRC DATA
5457 026666 062701 000010 ADD #10,R1
5458 026672 016502 000040 MOV 40(R5),R2 ;GET TEST FPS
5459 026676 170102 LDFPS R2 ;LOAD TEST FPS
5460
5461 026700 171411
5462 026702 170011
5463
5464
5465 026704 032737 000001 001042 ; INSTRUCTION DIDNT TRAP
5466 026712 001426 BIT #1,#FLAG ;VERIFY A NO TRAP CONDITION
5467 026714 104000 BEQ 24 ;BRANCH IF GOOD
5468 026716 000402 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
5469 026720 002013 .WORD 402 ;UNIQUE ERROR NUMBER
5470 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
5471 026722 000167 000042 JMP 24 ;INSTRUCTION SHOULD HAVE TRAPPED
5472 ;REJOIN CODE
5473
5474 026726 032737 000001 001042 ; INSTRUCTION TRAPPED
5475 026734 001005 50$: BIT #1,#FLAG ;SEE IF EXPECTING A TRAP
5476 026736 104000 BNE 51$ ;BRANCH IF EXPECTING A TRAP
5477 026740 000403 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
5478 026742 002013 .WORD 403 ;UNIQUE ERROR NUMBER
5479 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE

```


C9

```

5535
5536 027136
5537 027136
5538
5539
5540
5541 027136
5542 027136 005267 151642
5543
5544 027142 004767 000170
5545 027146 000177 000000 000000
5546 027154 000001
5547 027156 000000 000000 000000
5548 027164 000000
5549 027166 047557
5550 027170 047544
5551
5552 027172 004767 000140
5553 027176 077577 177777 177777
5554 027204 177777
5555 027206 077577 177777 000000
5556 027214 000000
5557 027216 007540
5558 027220 007540
5559
5560 027222 004767 000110
5561 027226 100377 177777 100000
5562 027234 000000
5563 027236 100377 177777 000000
5564 027244 000000
5565 027246 007517
5566 027250 007510
5567
5568 027252 004767 000060
5569 027256 100000 000000 000000
5570 027264 000000
5571 027266 000000 000000 000000
5572 027274 000000
5573 027276 007757
5574 027300 007744
5575
5576 027302 004767 000030
5577 027306 125252 125252 125252
5578 027314 125252
5579 027316 125252 125252 000000
5580 027324 000000
5581 027326 000000
5582 027330 000010
5583
5584
5585 027332 000167 000120
5586
5587
5588
5589
5590
  
```

```

; HOP15:
; MSFD:
; *****
; *TEST 67      TEST STCFD
; *****
; TST67:
; INC      $TESTN      ; INCREMENT TEST NUMBER
; 1/AC=0
; JSR      R7,SFDSUB    ; DO TEST
; .WORD    0177,0,0,1   ; ACO
; .WORD    0,0,0,0      ; RESULT
; .WORD    47557        ; TEST FPS
; .WORD    47544        ; RESULT FPS
; 2/AC>0, TRUNCATE
; JSR      R7,SFDSUB    ; DO TEST
; .WORD    77577,-1,-1,-1 ; ACO
; .WORD    77577,-1,0,0   ; RESULT
; .WORD    7540          ; TEST FPS
; .WORD    7540          ; RESULT FPS
; 3/AC<0, ROUND
; JSR      R7,SFDSUB    ; DO TEST
; .WORD    100377,-1,100000,0 ; ACO
; .WORD    100377,-1,0,0   ; RESULT
; .WORD    7517          ; TEST FPS
; .WORD    7510          ; RESULT FPS
; 4/AC=-0
; JSR      R7,SFDSUB    ; DO TEST
; .WORD    100000,0,0,0   ; ACO
; .WORD    0,0,0,0      ; RESULT
; .WORD    7757          ; TEST FPS
; .WORD    7744          ; RESULT FPS
; 5/AC<0
; JSR      R7,SFDSUB    ; DO TEST
; .WORD    125252,125252,125252,125252 ; ACO
; .WORD    125252,125252,0,0   ; RESULT
; .WORD    0              ; TEST FPS
; .WORD    10             ; RESULT FPS
;
; JMP      HOP16      ; GET OVER SUBROUTINE
; *****
; *****
; STCFD
;
; ACO
; RESULT
  
```



```

5647 027466 004767 000220 JSR R7,SDFSUB ;DO TEST
5648 027472 000177 000000 000000 .WORD 177,0,0,0 ;ACO
5649 027500 000000 .WORD 0,0 ;RESULT
5650 027502 000000 000000 .WORD 200 ;TEST FPS
5651 027506 000200 .WORD 204 ;RESULT FPS
5652 027510 000204 ;2/AC=-0
5653 CLR @FLAG ;NO INTERRUPT
5654 027512 005037 001042 JSR R7,SDFSUB ;DO TEST
5655 027516 004767 000170 .WORD 100000,300,200,100 ;ACO
5656 027522 100000 000300 000200
5657 027530 000100 .WORD 0,0 ;RESULT
5658 027532 000000 000000 .WORD 7777 ;TEST FPS
5659 027536 007777 .WORD 7744 ;RESULT FPS
5660 027540 007744 ;3/AC>0, TRUNCATE
5661 CLR @FLAG ;NO INTERRUPT
5662 027542 005037 001042 JSR R7,SDFSUB ;DO TEST
5663 027546 004767 000140 .WORD 55555,55555,-1,-1 ;ACO
5664 027552 055555 055555 177777
5665 027560 177777 .WORD 55555,55555 ;RESULT
5666 027562 055555 055555 .WORD 240 ;TEST FPS
5667 027566 000240 .WORD 240 ;RESULT FPS
5668 027570 000240 ;4/AC<0, ROUND TO UNDEFINED VARIABLE
5669 MOV @1,@FLAG ;INTERUPT
5670 027572 012737 000001 001042 JSR R7,SDFSUB ;DO TEST
5671 027600 004767 000106 177777 100000 .WORD 77777,-1,100000,0 ;ACO
5672 027604 077777 177777
5673 027612 000000 .WORD 0,0 ;RESULT
5674 027614 000000 000000 .WORD 1200 ;TEST FPS
5675 027620 001200 .WORD 101206 ;RESULT FPS
5676 027622 101206 ;5/AC<0, ROUND
5677 CLR @FLAG ;NO INTERRUPT
5678 027624 005037 001042 JSR R7,SDFSUB ;DO TEST
5679 027630 004767 000056 .WORD 125252,125252,125252,125252 ;ACO
5680 027634 125252 125252 125252
5681 027642 125252 .WORD 125252,125253 ;RESULT
5682 027644 125252 125253 .WORD 7700 ;TEST FPS
5683 027650 007700 .WORD 7710 ;RESULT FPS
5684 027652 007710 ;6/ROUND TO UV, TRAPS DISABLED
5685 MOV @2,@FLAG ;INTERUPT
5686 027654 012737 000002 001042 JSR R7,SDFSUB ;DO TEST
5687 027662 004767 000024 177777 177777 .WORD 77777,-1,-1,0 ;ACO
5688 027666 077777 177777
5689 027674 000000 .WORD 0,0 ;RESULT
5690 027676 000000 000000 .WORD 6700 ;TEST FPS
5691 027702 006700 .WORD 6706 ;RESULT FPS
5692 027704 006706
5693 ;
5694 ;
5695 027706 000167 000232 JMP HOP17 ;GET OVER SUBROUTINE
5696 ;
5697 ;
5698 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
5699 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
5700 STCDF
5701 ;
5702 ; ACO
; RESULT

```

```
5703      ;
5704      ;
5705      ;
5706      ;
5707      ;A TRAP CAN ONLY OCCUR IF ROUNDING CAUSES OVERFLOW
5708      ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
5709      ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
5710      SDFSUB: MOV      (SP)+,R5
5711      027714 012737 027774 000244      MOV      #501,0#FPVEC
5712      027722 012702 000200      MOV      #200,R2
5713      027726 170102      LDFPS      R2
5714      027730 172415      LDD      (R5),ACO
5715      027732 012701 001126      MOV      #RECDST,R1
5716      027736 016502 000014      MOV      14(R5),R2
5717      027742 170102      LDFPS      R2
5718      ;
5719      027744 176011      40$: STCDF      ACO,(R1)
5720      027746 170327      1$: STST      (PC)+
5721      027750 000000      .WORD      0
5722      ;
5723      ;INSTRUCTION DIDNT TRAP
5724      027752 032737 000001 001042      BIT      #1,0#FLAG
5725      027760 001426      BEQ      2$
5726      027762 104000      ERROR
5727      027764 000414      .WORD      414
5728      027766 002013      .WORD      FPPERR
5729      ;
5730      027770 000167 000042      JMP      2$
5731      ;
5732      ;INSTRUCTION TRAPPED
5733      027774 032737 000001 001042      50$: BIT      #1,0#FLAG
5734      030002 001005      BNE      51$
5735      030004 104000      ERROR
5736      030006 000415      .WORD      415
5737      030010 002013      .WORD      FPPERR
5738      ;
5739      030012 000167 000020      JMP      2$
5740      030016 012604      51$: MOV      (SP)+,R4
5741      030020 005726      TST      (SP)+
5742      030022 022704 027746      CMP      #11,R4
5743      030026 001403      BEQ      2$
5744      030030 104000      ERROR
5745      030032 000416      .WORD      416
5746      030034 002013      .WORD      FPPERR
5747      ;
5748      ;
5749      ;
5750      ;COMMON CODE FOR TRAP AND NO TRAP
5751      030036 170203      2$: VERIFY STATUS
5752      030040 016502 000016      STFPS      R3
5753      030044 020203      MOV      16(R5),R2
5754      030046 001403      CMP      R2,R3
5755      030050 104000      BEQ      3$
5756      030052 000417      ERROR
5757      030054 002013      .WORD      417
5758      .WORD      FPPERR

; RETURN ADDRESS TO USE AS POINTER
; REDIRECT TRAP VECTOR
; SET TO DOUBLE MODE FOR LOAD
; LOAD FPS
; LOAD ACO WITH TEST DATA
; POINT TO RESULT AREA
; GET TEST FPS
; LOAD TEST FPS

; *TEST INSTRUCTION
; WAIT FOR POSSIBLE FPA TRAP.
; STORE STATUS HERE.

; VERIFY A NO TRAP CONDITION
; BRANCH IF GOOD
; ALL ERRORS TO TRAP TO EMT VECTOR
; UNIQUE ERROR NUMBER
; ADDRESS OF ERROR MESSAGE
; INSTRUCTION SHOULD HAVE TRAPPED
; REJOIN CODE

; SEE IF EXPECTING A TRAP
; BRANCH IF EXPECTING A TRAP
; ALL ERRORS TO TRAP TO EMT VECTOR
; UNIQUE ERROR NUMBER
; ADDRESS OF ERROR MESSAGE
; INSTRUCTION WASNT SUPPOSE TO TRAP
; REJOIN CODE
; SEE IF PC = INSTRUCTION
; CLEAN UP STACK
; BRANCH IF GOOD COMPARE
; ALL ERRORS TO TRAP TO EMT VECTOR
; UNIQUE ERROR NUMBER
; ADDRESS OF ERROR MESSAGE
; PC WAS INCORRECT

; SAVE FPS
; GET EXPECTED STATUS
; VERIFY STATUS
; BRANCH IF GOOD
; ALL ERRORS TO TRAP TO EMT VECTOR
; UNIQUE ERROR NUMBER
; ADDRESS OF ERROR MESSAGE
; BAD FPS
```



```

5759 030056 010504      3$:  MOV    R5,R4          ;POINT TO EXPECTED DATA
5760 030060 062704 000010      ADD    #10,R4
5761 030064 004767 152040      4$:  JSR    R7,DATVFR          ;VERIFY DATA
5762 030070 005767 150744      TST    COUNT
5763 030074 001403      BEQ     5$
5764 030076 104000      ERROR
5765 030100 000420      .WORD   420
5766 030102 002013      .WORD   FPPERR          ;BAD ACO
5767
5768 030104 005737 001042      5$:  TST    #0FLAG
5769 030110 001002      BNE     7$
5770 030112 000165 000020      JMP    20(R5)
5771
5772 030116 012704 001106      ;VERIFY 7$:  MOV    #RECFEC,R4
5773 030122 170314      STST   (R4)
5774 030124 021427 000010      CMP    (R4),#10
5775 030130 001403      BEQ     8$
5776 030132 104000      ERROR
5777 030134 000421      .WORD   421
5778 030136 002013      .WORD   FPPERR          ;SEE IF NEED TO CHECK FEC
5779
5780 030140 000165 000020      8$:  JMP    20(R5)          ;BRANCH IF NEED TO CHECK
5781
5782 030144      ;HOP17:
5783 030144      MSFDI:
5784
5785      ;*****
5786      ;*TEST 71      TEST STCDF - USING ILLEGAL ACCUMULATOR
5787      ;*****
5788 030144 005267 150634      TST71:
5789 030150 012701 040000      INC     #TESTN          ;INCREMENT TEST NUMBER
5790 030154 170101      MOV     #40000,R1          ;DISABLE INTERRUPTS
5791 030156 176006      LDFPS  R1
5792 030160 170202      STCFD  ACO,AC6
5793 030162 170303      STFPS  R2
5794 030164 022702 140000      STST   R3
5795 030170 001403      CMP     #140000,R2
5796 030172 104000      BEQ     1$
5797 030174 000422      ERROR
5798 030176 002013      .WORD   422
5799
5800 030200 022703 000002      1$:  CMP     #2,R3
5801 030204 001403      BEQ     2$
5802 030206 104000      ERROR
5803 030210 000423      .WORD   423
5804 030212 002013      .WORD   FPPERR          ;FER BIT NOT SET ON ILLEGAL INST.
5805
5806 030214      2$:
5807
5808      ;MCLRD:
5809 030214
5810
5811      ;*****
5812      ;*TEST 72      TEST CLRD
5813      ;*****
5814 030214 005267 150564      TST72:
5815
5816      INC     #TESTN          ;INCREMENT TEST NUMBER

```

5815	030220	012701	001726		MOV	#TAB47,R1		;POINT TO DATA
5816	030224	012704	000200		MOV	#200,R4		;SET FPP STATUS TO DOUBLE
5817	030230	170104			LDFPS	R4		
5818	030232	172411			LDD	(R1),ACO		
5819	030234	012701	001126		MOV	#RECDST,R1		;POINT TO DATA BUFFER
5820	030240	174011			STD	ACO,(R1)		;STORE GARBAGE
5821	030242	170411			CLRD	(R1)		;CLEAR DATA BUFFER
5822	030244	012704	001256		MOV	#TAB6,R4		;VERIFY BUFFER =0
5823	030250	004767	151672		JSR	R7,DATVER		
5824	030254	005767	150560		TST	COUNT		
5825	030260	001403			BEQ	1#		;BRANCH I RECDST = 0
5826	030262	104000			ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
5827	030264	000424			.WORD	424		;UNIQUE ERROR NUMBER
5828	030266	002013			.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
5829								
5830	030270	170202		1#:	STFPS	R2		;RECDST NOT CLEARED
5831	030272	020227	000204		CMP	R2,#204		;SAVE STATUS
5832	030276	001403			BEQ	2#		;VERIFY STATUS
5833	030300	104000			ERROR			;BRANCH IF GOOD
5834	030302	000425			.WORD	425		;ALL ERRORS TO TRAP TO EMT VECTOR
5835	030304	002013			.WORD	FPPERR		;UNIQUE ERROR NUMBER
5836								;ADDRESS OF ERROR MESSAGE
5837	030306			2#:				;BAD STATUS
5838								
5839								
5840	030306							
5841								
5842								
5843								
5844	030306							
5845	030306	005267	150472					
5846	030312	012704	040200					
5847	030316	170104						
5848	030320	170406						
5849	030322	170203						
5850	030324	170305						
5851	030326	022703	140200					
5852	030332	001403						
5853	030334	104000						
5854	030336	000426						
5855	030340	002013						
5856								
5857	030342	022705	000002	1#:	CMP	#2,R5		;ERROR IN FPS
5858	030346	001403			BEQ	2#		;VERIFY FEC =2 OPCODE ERROR
5859	030350	104000			ERROR			;BRANCH IF GOOD
5860	030352	000427			.WORD	427		;ALL ERRORS TO TRAP TO EMT VECTOR
5861	030354	002013			.WORD	FPPERR		;UNIQUE ERROR NUMBER
5862								;ADDRESS OF ERROR MESSAGE
5863	030356			2#:				;BAD FEC
5864								
5865								
5866	030356							
5867								
5868								
5869								
5870	030356							

MCLRI:
 ;*****
 ;*TEST 73 TEST CLRD, ILLEGAL ACCUMULATOR
 ;*****
 TST73:
 INC #TESTN ;INCREMENT TEST NUMBER
 MOV #40200,R4 ;DISABLE INTERRUPTS
 LDFPS R4 ;LOAD STATUS
 CLRD R6 ;*TEST INSTRUCTION WITH ILLEGAL ACC
 STFPS R3 ;SAVE STATUS
 STST R5 ;SAVE FEC
 CMP #140200,R3 ;VERIFY ERROR
 BEQ 1# ;BRANCH IF FER SET
 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
 .WORD 426 ;UNIQUE ERROR NUMBER
 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE

ERROR IN FPS
 VERIFY FEC =2 OPCODE ERROR
 BRANCH IF GOOD
 ALL ERRORS TO TRAP TO EMT VECTOR
 UNIQUE ERROR NUMBER
 ADDRESS OF ERROR MESSAGE
 BAD FEC

MLS1:
 ;*****
 ;*TEST 74 TEST LDFPS, STFPS MODE 1
 ;*****
 TST74:


```

5871 030356 005267 150422 INC $TESTN ;INCREMENT TEST NUMBER
5872 030362 012704 001136 MOV $TSTLOC,R4 ;POINT R4 TO RAM
5873 030366 012714 147757 MOV $147757,(R4) ;SETUP EXPECTED STATUS
5874 030372 012701 001116 MOV $RECST,R1 ;SET BUFFER FOR RECEIVED STATUS
5875 030376 012737 030462 000244 MOV $10$,0$FPVEC ;SETUP TRAP VECTOR
5876 030404 170114 LDFPS (R4); ;*TEST INSTRUCTION
5877 030406 170211 STFPS (R1); ;*TEST INSTRUCTION
5878 030410 020427 001136 CMP R4,$TSTLOC ;VERIFY R4
5879 030414 001403 BEQ 1$ ;BRANCH IF GOOD
5880 030416 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
5881 030420 000430 .WORD 430 ;UNIQUE ERROR NUMBER
5882 030422 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
5883
5884 030424 020127 001116 1$: CMP R1,$RECST ;VERIFY R1
5885 030430 001403 BEQ 2$ ;BRANCH IF GOOD
5886 030432 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
5887 030434 000431 .WORD 431 ;UNIQUE ERROR NUMBER
5888 030436 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
5889
5890 030440 023727 001116 147757 2$: CMP $RECST,$147757 ;BAD R1
5891 030446 001412 BEQ 3$ ;VERIFY STATUS
5892 030450 104000 ERROR ;BRANCH F GOOD
5893 030452 000432 .WORD 432 ;ALL ERRORS TO TRAP TO EMT VECTOR
5894 030454 002013 .WORD FPPERR ;UNIQUE ERROR NUMBER
5895 ;ADDRESS OF ERROR MESSAGE
5896 030456 000167 000012 JMP 3$ ;BAD STATUS\
5897 ;UNEXPECTED TRAP ;GET OVER TRAP
5898 030462 012600 10$: MOV (SP),R0 ;SAVE PC
5899 030464 012605 MOV (SP),R5 ;SAVE PS
5900 030466 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
5901 030470 000433 .WORD 433 ;UNIQUE ERROR NUMBER
5902 030472 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
5903 ;UNEXPECTED TRAP
5904 030474 3$:
5905
5906
5907 030474
5908
5909
5910
5911 030474
5912 030474 005267 150304 TST75:
5913 030500 012704 001136 INC $TESTN ;INCREMENT TEST NUMBER
5914 030504 012714 145557 MOV $TSTLOC,R4 ;POINT R4 TO RAM
5915 030510 012701 001116 MOV $145557,(R4) ;SETUP EXPECTED STATUS
5916 030514 012737 030600 000244 MOV $RECST,R1 ;SET BUFFER FOR RECEIVED STATUS
5917 030522 170124 LDFPS (R4); ;SETUP TRAP VECTOR
5918 030524 170221 STFPS (R1); ;*TEST INSTRUCTION
5919 030526 020427 001140 CMP R4,$TSTLOC*2 ;*TEST INSTRUCTION
5920 030532 001403 BEQ 1$ ;VERIFY R4
5921 030534 104000 ERROR ;BRANCH IF GOOD
5922 030536 000434 .WORD 434 ;ALL ERRORS TO TRAP TO EMT VECTOR
5923 030540 002013 .WORD FPPERR ;UNIQUE ERROR NUMBER
5924 ;ADDRESS OF ERROR MESSAGE
5925 030542 020127 001120 1$: CMP R1,$RECST*2 ;VERIFY R1
5926 030546 001403 BEQ 2$ ;BRANCH IF GOOD

```

```

5927 030550 104000 ERROR
5928 030552 000435 .WORD 435 ;ALL ERRORS TO TRAP TO EMT VECTOR
5929 030554 002013 .WORD FPPERR ;UNIQUE ERROR NUMBER
5930 ;ADDRESS OF ERROR MESSAGE
5931 030556 023727 001116 145557 2: CMP @RECST,@145557 ;BAD R1 ;VERIFY STATUS
5932 030564 001412 BEQ 3: ;BRANCH F GOOD
5933 030566 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
5934 030570 000436 .WORD 436 ;UNIQUE ERROR NUMBER
5935 030572 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
5936 ;BAD STATUS\ ;GET OVER TRAP
5937 030574 000167 000012 JMP 3:
5938 ;UNEXPECTED TRAP
5939 030600 012600 10: MOV (SP)+,R0 ;SAVE PC
5940 030602 012605 MOV (SP)+,R5 ;SAVE PS
5941 030604 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
5942 030606 000437 .WORD 437 ;UNIQUE ERROR NUMBER
5943 030610 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
5944 ;UNEXPECTED TRAP
5945 030612 3:
5946
5947
5948 030612
5949
5950
5951
5952 030612
5953 030612 005267 150166 TST76: INC $TESTN ;INCREMENT TEST NUMBER
5954 030616 012704 001136 MOV @TSTLOC,R4 ;POINT R4 TO RAM
5955 030622 012737 001142 001136 MOV @TSTLOC+4,@TSTLOC ;TSTLOC= DEFERRED ADDRESS
5956 030630 012737 147501 001142 MOV @147501,@TSTLOC+4 ;SETUP EXPECTED STATUS
5957 030636 012701 001146 MOV @TSTLOC+10,R1 ;R1 POINTS TO TSTLOC+10
5958 030642 012737 001116 001146 MOV @RECST,@TSTLOC+10 ;SET DEFERRED BUFFER FOR RECEIVED STATUS
5959 030650 012737 030734 000244 MOV @108,@FPVcC ;SETUP TRAP VECTOR
5960 030656 170134 LDFPS @R4) ;*TEST INSTRUCTION
5961 030660 170231 STFPS @R1) ;*TEST INSTRUCTION
5962 030662 020427 001140 CMP R4,@TSTLOC+2 ;VERIFY R4
5963 030666 001403 BEQ 1: ;BRANCH IF GOOD
5964 030670 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
5965 030672 000440 .WORD 440 ;UNIQUE ERROR NUMBER
5966 030674 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
5967
5968 030676 020127 001150 1: CMP R1,@TSTLOC+12 ;VERIFY R1
5969 030702 001403 BEQ 2: ;BRANCH IF GOOD
5970 030704 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
5971 030706 000441 .WORD 441 ;UNIQUE ERROR NUMBER
5972 030710 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
5973
5974 030712 023727 001116 147501 2: CMP @RECST,@147501 ;BAD R1 ;VERIFY STATUS
5975 030720 001412 BEQ 3: ;BRANCH F GOOD
5976 030722 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
5977 030724 000442 .WORD 442 ;UNIQUE ERROR NUMBER
5978 030726 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
5979 ;BAD STATUS\ ;GET OVER TRAP
5980 030730 000167 000012 JMP 3:
5981 ;UNEXPECTED TRAP
5982 030734 012600 10: MOV (SP)+,R0 ;SAVE PC

```



```

5983 030736 012605      MOV      (SP)+,R5      ;SAVE PS
5984 030740 104000      ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
5985 030742 000443      .WORD    443              ;UNIQUE ERROR NUMBER
5986 030744 002013      .WORD    FPPERR          ;ADDRESS OF ERROR MESSAGE
5987                                     ;UNEXPECTED TRAP
5988 030746      3$:
5989
5990
5991 030746      MLS4:
5992      ;*****
5993      ;*TEST 77      TEST LDFPS, STFPS MODE 4
5994      ;*****
5995      TST77:
5996 030746 005267 150032      INC      $TESTN      ;INCREMENT TEST NUMBER
5997 030752 012704 001140      MOV      @TSTLOC+2,R4    ;POINT R4 TO RAM
5998 030756 012737 147757 001136      MOV      @147757,@TSTLOC  ;TSTLOC= STATUS ADDRESS
5999 030764 012701 001120      MOV      @RECST+2,R1    ;SET BUFFER FOR RECEIVED STATUS
6000 030770 012737 031054 000244      MOV      @10$,@FPVEC    ;SETUP TRAP VECTOR
6001 030776 170144      LDFPS    -(R4)      ;*TEST INSTRUCTION
6002 031000 170241      STFPS    -(R1)      ;*TEST INSTRUCTION
6003 031002 020427 001136      CMP      R4,@TSTLOC    ;VERIFY R4
6004 031006 001403      BEQ      1$          ;BRANCH IF GOOD
6005 031010 104000      ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
6006 031012 000444      .WORD    444              ;UNIQUE ERROR NUMBER
6007 031014 002013      .WORD    FPPERR          ;ADDRESS OF ERROR MESSAGE
6008
6009 031016 020127 001116      1$:      CMP      R1,@RECST    ;VERIFY R1
6010 031022 001403      BEQ      2$          ;BRANCH IF GOOD
6011 031024 104000      ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
6012 031026 000445      .WORD    445              ;UNIQUE ERROR NUMBER
6013 031030 002013      .WORD    FPPERR          ;ADDRESS OF ERROR MESSAGE
6014
6015 031032 023727 001116 147757 2$:      CMP      @@RECST,@147757 ;BAD R1
6016 031040 001412      BEQ      3$          ;VERIFY STATUS
6017 031042 104000      ERROR                      ;BRANCH F GOOD
6018 031044 000446      .WORD    446              ;ALL ERRORS TO TRAP TO EMT VECTOR
6019 031046 002013      .WORD    FPPERR          ;UNIQUE ERROR NUMBER
6020                                     ;ADDRESS OF ERROR MESSAGE
6021 031050 000167 000012      JMP      3$          ;BAD STATUS\
6022                                     ;GET OVER TRAP
6023 031054 012600      ;UNEXPECTED TRAP
6024 031056 012605      10$:      MOV      (SP)+,R0      ;SAVE PC
6025 031060 104000      MOV      (SP)+,R5      ;SAVE PS
6026 031062 000447      ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
6027 031064 002013      .WORD    447              ;UNIQUE ERROR NUMBER
6028                                     ;ADDRESS OF ERROR MESSAGE
6029                                     ;UNEXPECTED TRAP
6030      3$:
6031
6032 031066      MLS5:
6033      ;*****
6034      ;*TEST 100     TEST LDFPS, STFPS MODE 5
6035      ;*****
6036 031066      TST100:
6037 031066 005267 147712      INC      $TESTN      ;INCREMENT TEST NUMBER
6038 031072 012704 001140      MOV      @TSTLOC+2,R4    ;POINT R4 TO RAM

```

```

6039 031076 012737 001142 001136      MOV    #TSTLOC+4,#TSTLOC      ;TSTLOC= DEFERRED ADDRESS
6040 031104 012737 147501 001142      MOV    #147501,#TSTLOC+4    ;SETUP EXPECTED STATUS
6041 031112 012701 001150              MOV    #TSTLOC+12,R1      ;R1 POINTS TO 412
6042 031116 012737 001116 001146      MOV    #RECST,#TSTLOC+10    ;SET DEFERRED BUFFER FOR RECEIVED STATUS
6043 031124 012737 031210 000244      MOV    #10#,#FPVEC        ;SETUP TRAP VECTOR
6044 031132 170154              LDFPS  4(R4)          ;*TEST INSTRUCTION
6045 031134 170251              STFPS  -100(R1)         ;*TEST INSTRUCTION
6046 031136 020427 001136              CMP    R4,#TSTLOC        ;VERIFY R4
6047 031142 001403              BEQ    1#              ;BRANCH IF GOOD
6048 031144 104000              ERROR             ;ALL ERRORS TO TRAP TO EMT VECTOR
6049 031146 000450              .WORD  450              ;UNIQUE ERROR NUMBER
6050 031150 002013              .WORD  FPPERR          ;ADDRESS OF ERROR MESSAGE
6051
6052 031152 020127 001146              1# :    CMP    R1,#TSTLOC+10      ;VERIFY R1
6053 031156 001403              BEQ    2#              ;BRANCH IF GOOD
6054 031160 104000              ERROR             ;ALL ERRORS TO TRAP TO EMT VECTOR
6055 031162 000451              .WORD  451              ;UNIQUE ERROR NUMBER
6056 031164 002013              .WORD  FPPERR          ;ADDRESS OF ERROR MESSAGE
6057
6058 031166 023727 001116 147501 2# :    CMP    #RECST,#147501    ;BAD R1
6059 031174 001412              BEQ    3#              ;VERIFY STATUS
6060 031176 104000              ERROR             ;BRANCH F GOOD
6061 031200 000452              .WORD  452              ;ALL ERRORS TO TRAP TO EMT VECTOR
6062 031202 002013              .WORD  FPPERR          ;UNIQUE ERROR NUMBER
6063                                ;ADDRESS OF ERROR MESSAGE
6064 031204 000167 000012              JMP    3#              ;BAD STATUS\
6065                                ;GET OVER TRAP
6066                                ;UNEXPECTED TRAP
6067 031210 012600              10# :    MOV    (SP)+,R0          ;SAVE PC
6068 031212 012605              MOV    (SP)+,R5          ;SAVE PS
6069 031214 104000              ERROR             ;ALL ERRORS TO TRAP TO EMT VECTOR
6070 031216 000453              .WORD  453              ;UNIQUE ERROR NUMBER
6071 031220 002013              .WORD  FPPERR          ;ADDRESS OF ERROR MESSAGE
6072                                ;UNEXPECTED TRAP
6073                                3# :
6074
6075 031222              MLS6:
6076              ;*****
6077              ;*TEST 101      TEST LDFPS, STFPS MODE 6
6078              ;*****
6079              TST101:
6080 031222 005267 147556              INC    #TESTN            ;INCREMENT TEST NUMBER
6081 031226 012704 001136              MOV    #TSTLOC,R4        ;POINT R4 TO RAM
6082 031232 012737 140001 001142      MOV    #140001,#TSTLOC+4    ;SETUP EXPECTED STATUS
6083 031240 012701 001246              MOV    #TSTLOC+110,R1      ;R1 WILL POINT TO TESTLOC+10
6084 031244 012737 031334 000244      MOV    #10#,#FPVEC        ;SETUP TRAP VECTOR
6085 031252 170164 000004              LDFPS  4(R4)          ;*TEST INSTRUCTION
6086 031256 170261 177700              STFPS  -100(R1)         ;*TEST INSTRUCTION
6087 031262 020427 001136              CMP    R4,#TSTLOC        ;VERIFY R4
6088 031266 001403              BEQ    1#              ;BRANCH IF GOOD
6089 031270 104000              ERROR             ;ALL ERRORS TO TRAP TO EMT VECTOR
6090 031272 000454              .WORD  454              ;UNIQUE ERROR NUMBER
6091 031274 002013              .WORD  FPPERR          ;ADDRESS OF ERROR MESSAGE
6092
6093 031276 020127 001246              1# :    CMP    R1,#TSTLOC+110    ;VERIFY R1
6094 031302 001403              BEQ    2#              ;BRANCH IF GOOD
```


M9

```

6095 031304 104000 ERROR .WORD 455 ;ALL ERRORS TO TRAP TO EMT VECTOR
6096 031306 000455 .WORD FPPERR ;UNIQUE ERROR NUMBER
6097 031310 002013 ;ADDRESS OF ERROR MESSAGE
6098
6099 031312 023727 001146 140001 2: CMP @TSTLOC+10,#140001 ;BAD R1
6100 031320 001412 BEQ 3: ;VERIFY STATUS
6101 031322 104000 ERROR ;BRANCH F GOOD
6102 031324 000456 .WORD 456 ;ALL ERRORS TO TRAP TO EMT VECTOR
6103 031326 002013 .WORD FPPERR ;UNIQUE ERROR NUMBER
6104 ;ADDRESS OF ERROR MESSAGE
6105 031330 000167 000012 JMP 3: ;BAD STATUS\
6106 ;UNEXPECTED TRAP ;GET OVER TRAP
6107 031334 012600 10: MOV (SP)+,R0 ;SAVE PC
6108 031336 012605 MOV (SP)+,R5 ;SAVE PS
6109 031340 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
6110 031342 000457 .WORD 457 ;UNIQUE ERROR NUMBER
6111 031344 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
6112 ;UNEXPECTED TRAP
6113 031346 3:
6114
6115
6116 031346
6117
6118
6119
6120 031346
6121 031346 005267 147432
6122 031352 012704 001236
6123 031356 012737 001142 001136
6124 031364 012737 145501 001142
6125 031372 012701 001046
6126 031376 012737 001146 001140
6127 031404 012737 031474 000244
6128 031412 170174 177700
6129 031416 170271 000072
6130 031422 020427 001236
6131 031426 001403
6132 031430 104000
6133 031432 000460
6134 031434 002013
6135
6136 031436 020127 001046 1: CMP R1,@TSTLOC-70 ;VERIFY R1
6137 031442 001403 BEQ 2: ;BRANCH IF GOOD
6138 031444 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
6139 031446 000461 .WORD 461 ;UNIQUE ERROR NUMBER
6140 031450 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
6141
6142 031452 023727 001146 145501 2: CMP @TSTLOC+10,#145501 ;BAD R1
6143 031460 001412 BEQ 3: ;VERIFY STATUS
6144 031462 104000 ERROR ;BRANCH F GOOD
6145 031464 000462 .WORD 462 ;ALL ERRORS TO TRAP TO EMT VECTOR
6146 031466 002013 .WORD FPPERR ;UNIQUE ERROR NUMBER
6147 ;ADDRESS OF ERROR MESSAGE
6148 031470 000167 000012 JMP 3: ;BAD STATUS\
6149 ;UNEXPECTED TRAP ;GET OVER TRAP
6150 031474 012600 10: MOV (SP)+,R0 ;SAVE PC
    
```

MLS7:

```

;*****
;TEST 102 TEST LDFPS, STFPS MODE 7
;*****
TST102:
    
```

```

INC #TESTN ;INCREMENT TEST NUMBER
MOV @TSTLOC+100,R4 ;POINT R4 TO RAM
MOV @TSTLOC+4,@TSTLOC ;TSTLOC= DEFERRED ADDRESS
MOV #145501,@TSTLOC+4 ;SETUP EXPECTED STATUS
MOV @TSTLOC-70,R1 ;R1 POINTS TO TSTLOC+10
MOV @TSTLOC+10,@TSTLOC+2 ;
MOV #10,@FPVEC ;SETUP TRAP VECTOR
LDFPS @-100(R4) ;TEST INSTRUCTION
STFPS @72(R1) ;TEST INSTRUCTION
CMP R4,@TSTLOC+100 ;VERIFY R4
BEQ 1: ;BRANCH IF GOOD
ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 460 ;UNIQUE ERROR NUMBER
.WORD FPPERR ;ADDRESS OF ERROR MESSAGE
;
1: CMP R1,@TSTLOC-70 ;VERIFY R1
BEQ 2: ;BRANCH IF GOOD
ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 461 ;UNIQUE ERROR NUMBER
.WORD FPPERR ;ADDRESS OF ERROR MESSAGE
;BAD R1
2: CMP @TSTLOC+10,#145501 ;VERIFY STATUS
BEQ 3: ;BRANCH F GOOD
ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
.WORD 462 ;UNIQUE ERROR NUMBER
.WORD FPPERR ;ADDRESS OF ERROR MESSAGE
;BAD STATUS\
3: JMP 3: ;GET OVER TRAP
;UNEXPECTED TRAP
10: MOV (SP)+,R0 ;SAVE PC
    
```

```

6151 031476 012605      MOV      (SP)+,R5      ;SAVE PS
6152 031500 104000      ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
6153 031502 000463      .WORD    463              ;UNIQUE ERROR NUMBER
6154 031504 002013      .WORD    FPPERR           ;ADDRESS OF ERROR MESSAGE
6155                                     ;UNEXPECTED TRAP
6156 031506      3$:
6157
6158
6159 031506      MLDC2:
6160      ;*****
6161      ;*TEST 103      TEST LDCLD MODE 27
6162      ;*****
6163      TST103:
6164 031506 005267 147272      INC      $TESTN      ;INCREMENT TEST NUMBER
6165 031512 005001      CLR      R1              ;INIT R1
6166 031514 012704 007700      MOV      #7700,R4      ;FPS=DOUBLE, LONG
6167 031520 170104      LDFPS    R4
6168 031522 012737 031562 000244      MOV      #10$,#FPVEC      ;SETUP WILD TRAP
6169 031530 177027      LDCLD    (R7)+,AC0      ;*TEST INSTRUCTION
6170 031532 005201      INC      R1
6171 031534 005201      INC      R1
6172 031536 005201      INC      R1
6173 031540 005201      INC      R1
6174 031542 020127 000003      CMP      R1,#3      ;VERIFY
6175 031546 001412      BEQ      1$              ;BRANCH IF GOOD
6176 031550 104000      ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
6177 031552 000464      .WORD    464              ;UNIQUE ERROR NUMBER
6178 031554 002013      .WORD    FPPERR           ;ADDRESS OF ERROR MESSAGE
6179                                     ;INSTRUCTION FAILED
6180 031556 000167 000012      JMP      1$              ;JUMP OVER WILD TRAP
6181 031562 012600      10$:      MOV      (SP)+,R0      ;SAVE PC
6182 031564 012605      MOV      (SP)+,R5      ;SAVE PS
6183 031566 104000      ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
6184 031570 000465      .WORD    465              ;UNIQUE ERROR NUMBER
6185 031572 002013      .WORD    FPPERR           ;ADDRESS OF ERROR MESSAGE
6186                                     ;WILD TRAP ON INSTRUCTION
6187 031574 012704 001266      MOV      #TAB6A,R4      ;POINT TO EXPECTED DATA
6188 031600 012701 001126      MOV      #RECDST,R1      ;POINT TO DATA BUFFER
6189 031604 174011      STD      AC0,(R1)      ;VERIFY DATA
6190 031606 004767 150334      JSR      R7,DATVER
6191 031612 005767 147222      TST      COUNT
6192 031616 001403      BEQ      2$              ;BRANCH IF GOOD DATA
6193 031620 104000      ERROR                      ;ALL ERRORS TO TRAP TO EMT VECTOR
6194 031622 000466      .WORD    466              ;UNIQUE ERROR NUMBER
6195 031624 002013      .WORD    FPPERR           ;ADDRESS OF ERROR MESSAGE
6196                                     ;BAD DATA
6197 031626      2$:
6198
6199
6200
6201 031626      MLCF:
6202      ;*****
6203      ;*TEST 104      TEST LDCIF, LDCLF
6204      ;*****
6205 031626 005267 147152      TST104:      INC      $TESTN      ;INCREMENT TEST NUMBER
6206

```


6207				:1/INT=0			
6208	031632	004767	000500	JSR	R7,LCFSUB		:DO TEST
6209	031636	000000	000100	.WORD	0,0	:FSRC	
6210	031642	000000	000000	.WORD	0,0	:RESULT	
6211	031646	000000		.WORD	0	:TEST FPS	
6212	031650	000004		.WORD	4	:RESULT FPS	
6213				:2/INT=0,-1			
6214	031652	004767	000460	JSR	R7,LCFSUB		:DO TEST
6215	031656	000000	177777	.WORD	0,-1	:FSRC	
6216	031662	000000	000000	.WORD	0,0	:RESULT	
6217	031666	007440		.WORD	7440	:TEST FPS	
6218	031670	007444		.WORD	7444	:RESULT FPS	
6219				:3/LONG=0			
6220	031672	004767	000440	JSR	R7,LCFSUB		:DO TEST
6221	031676	000000	000000	.WORD	0,0	:FSRC	
6222	031702	000000	000000	.WORD	0,0	:RESULT	
6223	031706	000100		.WORD	100	:TEST FPS	
6224	031710	000104		.WORD	104	:RESULT FPS	
6225				:4/INT=40000			
6226	031712	004767	000420	JSR	R7,LCFSUB		:DO TEST
6227	031716	040000	000000	.WORD	40000,0	:FSRC	
6228	031722	043600	000000	.WORD	43600,0	:RESULT	
6229	031726	000017		.WORD	17	:TEST FPS	
6230	031730	000000		.WORD	0	:RESULT FPS	
6231				:5/LONG=1			
6232	031732	004767	000400	JSR	R7,LCFSUB		:DO TEST
6233	031736	000000	000001	.WORD	0,1	:FSRC	
6234	031742	040200	000000	.WORD	40200,0	:RESULT	
6235	031746	000117		.WORD	117	:TEST FPS	
6236	031750	000100		.WORD	100	:RESULT FPS	
6237				:6/INT=PATTERN			
6238	031752	004767	000360	JSR	R7,LCFSUB		:DO TEST
6239	031756	000252	025252	.WORD	252,25252	:FSRC	
6240	031762	042052	000000	.WORD	42052,0	:RESULT	
6241	031766	000000		.WORD	0	:TEST FPS	
6242	031770	000000		.WORD	0	:RESULT FPS	
6243				:7/INT=-40000			
6244	031772	004767	000340	JSR	R7,LCFSUB		:DO TEST
6245	031776	140000	000000	.WORD	-40000,0	:FSRC	
6246	032002	143600	000000	.WORD	143600,0	:RESULT	
6247	032006	000007		.WORD	7	:TEST FPS	
6248	032010	000010		.WORD	10	:RESULT FPS	
6249				:8/INT=-1			
6250	032012	004767	000320	JSR	R7,LCFSUB		:DO TEST
6251	032016	177777	000000	.WORD	-1,0	:FSRC	
6252	032022	140200	000000	.WORD	140200,0	:RESULT	
6253	032026	000007		.WORD	7	:TEST FPS	
6254	032030	000010		.WORD	10	:RESULT FPS	
6255				:9/INT=PATTERN			
6256	032032	004767	000300	JSR	R7,LCFSUB		:DO TEST
6257	032036	125252	125252	.WORD	125252,125252	:FSRC	
6258	032042	143652	126000	.WORD	143652,126000	:RESULT	
6259	032046	000007		.WORD	7	:TEST FPS	
6260	032050	000010		.WORD	10	:RESULT FPS	
6261				:10/LONG=40000			
6262	032052	004767	000260	JSR	R7,LCFSUB		:DO TEST

6263	032056	040000	000000	.WORD	40000,0	FSRC	
6264	032062	047600	000000	.WORD	47600,0	RESULT	
6265	032066	000117		.WORD	117	TEST FPS	
6266	032070	000100		.WORD	100	RESULT FPS	
6267				;11/LONG=1			
6268	032072	004767	000240	JSR	R7,LCFSUB	DO TEST	
6269	032076	000000	000001	.WORD	0,1	FSRC	
6270	032102	040200	000000	.WORD	40200,0	RESULT	
6271	032106	007557		.WORD	7557	TEST FPS	
6272	032110	007540		.WORD	7540	RESULT FPS	
6273				;12/LONG=PATTERN			
6274	032112	004767	000220	JSR	R7,LCFSUB	DO TEST	
6275	032116	000000	000252	.WORD	0,252	FSRC	
6276	032122	042052	000000	.WORD	42052,0	RESULT	
6277	032126	007557		.WORD	7557	TEST FPS	
6278	032130	007540		.WORD	7540	RESULT FPS	
6279				;13/LONG = -40000			
6280	032132	004767	000200	JSR	R7,LCFSUB	DO TEST	
6281	032136	140000	000000	.WORD	-40000,0	FSRC	
6282	032142	147600	000000	.WORD	147600,0	RESULT	
6283	032146	000107		.WORD	107	TEST FPS	
6284	032150	000110		.WORD	110	RESULT FPS	
6285				;14/LONG=-1			
6286	032152	004767	000160	JSR	R7,LCFSUB	DO TEST	
6287	032156	177777	177777	.WORD	-1,-1	FSRC	
6288	032162	140200	000000	.WORD	140200,0	RESULT	
6289	032166	007500		.WORD	7500	TEST FPS	
6290	032170	007510		.WORD	7510	RESULT FPS	
6291				;15/LONG=PATTERN			
6292	032172	004767	000140	JSR	R7,LCFSUB	DO TEST	
6293	032176	125252	125252	.WORD	125252,125252	FSRC	
6294	032202	147652	125253	.WORD	147652,125253	RESULT	
6295	032206	000105		.WORD	105	TEST FPS	
6296	032210	000110		.WORD	110	RESULT FPS	
6297				;16/LONG=77777,177500			
6298	032212	004767	000120	JSR	R7,LCFSUB	DO TEST	
6299	032216	077777	177500	.WORD	77777,177500	FSRC	
6300	032222	047777	177777	.WORD	47777,177777	RESULT	
6301	032226	000117		.WORD	117	TEST FPS	
6302	032230	000100		.WORD	100	RESULT FPS	
6303				;17/LONG=40000,100			
6304	032232	004767	000100	JSR	R7,LCFSUB	DO TEST	
6305	032236	040000	000100	.WORD	40000,100	FSRC	
6306	032242	047600	000001	.WORD	47600,1	RESULT	
6307	032246	007502		.WORD	7502	TEST FPS	
6308	032250	007500		.WORD	7500	RESULT FPS	
6309				;18/LONG=40000,100 - TRUNCATE			
6310	032252	004767	000060	JSR	R7,LCFSUB	DO TEST	
6311	032256	040000	000100	.WORD	40000,100	FSRC	
6312	032262	047600	000000	.WORD	47600,0	RESULT	
6313	032266	007557		.WORD	7557	TEST FPS	
6314	032270	007540		.WORD	7540	RESULT FPS	
6315				;19/INT= MOST NEGATIVE			
6316	032272	004767	000040	JSR	R7,LCFSUB	DO TEST	
6317	032276	100000	000000	.WORD	100000,0	FSRC	
6318	032302	144000	000000	.WORD	144000,0	RESULT	


```

6319 032306 000007 .WORD 7 ; TEST FPS
6320 032310 000010 .WORD 10 ; RESULT FPS
6321 ;20/LONG= MOST NEGATIVE
6322 032312 004767 000020 JSR R7,LCFSUB ; DO TEST
6323 032316 100000 000000 .WORD 100000,0 ; FSRC
6324 032322 150000 000000 .WORD 150000,0 ; RESULT
6325 032326 000107 .WORD 107 ; TEST FPS
6326 032330 000110 .WORD 110 ; RESULT FPS
6327 ;
6328 ;
6329 032332 000167 000126 JMP HOP18 ; GET OVER SUBROUTINE
6330 ;
6331 ;
6332 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
6333 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
6334 ;LDCIF, LDCLF
6335 ;
6336 ;
6337 ; FSRC
6338 ; RESULT
6339 ; FPS BEFORE EXECUTION
6340 ; FPS AFTER EXECUTION
6341 ;
6342 ; NO TRAP CAN OCCUR
6343 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
6344 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
6345 032336 012602 LCFSUB: MOV (SP)+,R2 ; RETURN ADDRESS TO USE AS POINTER
6346 032340 012737 032446 000244 MOV #501,B#FVEC ; REDIRECT TRAP VECTOR
6347 032346 012701 001126 MOV #RECDST,R1 ; POINT TO RESULT AREA
6348 032352 016200 000010 MOV 10(R2),R0 ; GET TEST FPS
6349 032356 170100 LDFPS R0 ; LOAD TEST FPS
6350 032360 010204 MOV R2,R4 ; POINT TO TEST DATA
6351 ;
6352 032362 177014 401: LDCIF (R4),AC0 ; *TEST INSTRUCTION (ACCORDING TO MODE)
6353 ;
6354 ; VERIFY STATUS
6355 032364 170203 21: STFPS R3 ; SAVE FPS
6356 032366 012700 000200 MOV #200,R0 ; SET FPP STATUS TO DOUBLE
6357 032372 170100 LDFPS R0 ;
6358 032374 174011 STD AC0,(R1) ; SAVE TEST RESULT INTO RECDST
6359 032376 016200 000012 MOV 12(R2),R0 ; GET EXPECTED STATUS
6360 032402 020003 CMP R0,R3 ; VERIFY STATUS
6361 032404 001403 BEQ 31 ; BRANCH IF GOOD
6362 032406 104000 ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
6363 032410 000467 .WORD 467 ; UNIQUE ERROR NUMBER
6364 032412 002013 .WORD FPPERR ; ADDRESS OF ERROR MESSAGE
6365 ;
6366 032414 010204 31: MOV R2,R4 ; BAD FPS
6367 032416 062704 000004 ADD #4,R4 ; POINT TO EXPECTED DATA
6368 032422 004767 147502 41: JSR R7,DATVFR ; VERIFY DATA
6369 032426 005767 146406 TST COUNT ;
6370 032432 001403 BEQ 51 ; BRANCH IF GOOD
6371 032434 104000 ERROR ; ALL ERRORS TO TRAP TO EMT VECTOR
6372 032436 000470 .WORD 470 ; UNIQUE ERROR NUMBER
6373 032440 002013 .WORD FPPERR ; ADDRESS OF ERROR MESSAGE
6374 ; BAD AC0

```

```

6375 032442 000162 000014 54: JMP 14(R2) ;RETURN FROM TEST
6376
6377 ;INSTRUCTION TRAPPED
6378 032446 012600 504: MOV (SP),R0 ;SAVE PC
6379 032450 012605 MOV (SP),R5 ;SAVE PS
6380 032452 104000 ;ALL ERRORS TO TRAP TO EMT VECTOR
6381 032454 000471 ;UNIQUE ERROR NUMBER
6382 032456 002013 .WORD 471 ;ADDRESS OF ERROR MESSAGE
6383 ;INSTRUCTION WASNT SUPPOSE TO TRAP
6384 032460 000167 177756 JMP 54 ;CONTINUE
6385 032464
6386
6387 032464
6388
6389
6390
6391 032464
6392 032464 005267 146314
6393
6394 032470 004767 000264
6395 032474 000000 000000 ;1/LONG=0 INC #TESTN ;INCREMENT TEST NUMBER
6396 032500 000000 000000 000000 JSR R7,LCDSUB ;DO TEST
6397 032506 000000 .WORD 0,0 .FSRC
6398 032510 007313 .WORD 0,0,0,0 ;RESULT
6399 032512 007304 .WORD 7313 ;TEST FPS
6400 .WORD 7304 ;RESULT FPS
6401 032514 004767 000240 ;2/INT=0 JSR R7,LCDSUB ;DO TEST
6402 032520 000000 000001 .WORD 0,1 .FSRC
6403 032524 040200 000000 000000 .WORD 40200,0,0,0 ;RESULT
6404 032532 000000 .WORD 7757 ;TEST FPS
6405 032534 007757 .WORD 7740 ;RESULT FPS
6406 032536 007740
6407
6408 032540 004767 000214 ;3/INT=40000 JSR R7,LCDSUB ;DO TEST
6409 032544 040000 177777 .WORD 40000,-1 .FSRC
6410 032550 043600 000000 000000 .WORD 43600,0,0,0 ;RESULT
6411 032556 000000 .WORD 7617 ;TEST FPS
6412 032560 007617 .WORD 7600 ;RESULT FPS
6413 032562 007600
6414
6415 032564 004767 000170 ;4/INT=-40000 JSR R7,LCDSUB ;DO TEST
6416 032570 140000 177777 .WORD -40000,-1 .FSRC
6417 032574 143600 000000 000000 .WORD 143600,0,0,0 ;RESULT
6418 032602 000000 .WORD 7600 ;TEST FPS
6419 032604 007600 .WORD 7610 ;RESULT FPS
6420 032606 007610
6421
6422 032610 004767 000144 ;5/LONG=40000 JSR R7,LCDSUB ;DO TEST
6423 032614 040000 000000 .WORD 40000,0 .FSRC
6424 032620 047600 000000 000000 .WORD 47600,0,0,0 ;RESULT
6425 032626 000000 .WORD 7757 ;TEST FPS
6426 032630 007757 .WORD 7740 ;RESULT FPS
6427 032632 007740
6428
6429 032634 004767 000120 ;6/LONG=1 JSR R7,LCDSUB ;DO TEST
6430 032640 000000 000001 .WORD 0,1 .FSRC

```



```

6431 032644 040200 000000 000000 .WORD 40200,0,0,0 ;RESULT
6432 032652 000000
6433 032654 000300 .WORD 300 ; TEST FPS
6434 032656 000300 .WORD 300 ;RESULT FPS
6435 ;7/LONG=-2
6436 032660 004767 000074 JSR R7,LCDSUB ;DO TEST
6437 032664 177777 177776 .WORD -1,-2 ;FSRC
6438 032670 140400 000000 000000 .WORD 140400,0,0,0 ;RESULT
6439 032676 000000
6440 032700 007300 .WORD 7300 ; TEST FPS
6441 032702 007310 .WORD 7310 ;RESULT FPS
6442 ;8/INT=PATTERN
6443 032704 004767 000050 JSR R7,LCDSUB ;DO TEST
6444 032710 123456 176543 .WORD 123456,176543 ;FSRC
6445 032714 143661 122000 000000 .WORD 143661,122000,0,0 ;RESULT
6446 032722 000000
6447 032724 000200 .WORD 200 ; TEST FPS
6448 032726 000210 .WORD 210 ;RESULT FPS
6449 ;9/LONG=PATTERN
6450 032730 004767 000024 JSR R7,LCDSUB ;DO TEST
6451 032734 125252 125252 .WORD 125252,125252 ;FSRC
6452 032740 147652 125252 126000 .WORD 147652,125252,126000,0 ;RESULT
6453 032746 000000
6454 032750 000300 .WORD 300 ; TEST FPS
6455 032752 000310 .WORD 310 ;RESULT FPS
6456 ;
6457 ;
6458 ;
6459 032754 000167 000126 JMP HOP19 ;GET OVER SUBROUTINE
6460 ;
6461 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
6462 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
6463 ;LDCID, LDCLD
6464 ;
6465 ;
6466 ; FSRC
6467 ; RESULT
6468 ; FPS BEFORE EXECUTION
6469 ; FPS AFTER EXECUTION
6470 ;
6471 ;NO TRAP CAN OCCUR
6472 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
6473 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
6474 ;
6475 032760 012602 LCDSUB: MOV (SP)+,R2 ; RETURN ADDRESS TO USE AS POINTER
6476 032762 012737 033070 000244 MOV #501,0#FPVEC ;REDIRECT TRAP VECTOR
6477 032770 012701 001126 MOV #RECDST,R1 ;POINT TO RESULT AREA
6478 032774 016200 000014 MOV 14(R2),R0 ;GET TEST FPS
6479 033000 170100 LDFPS R0 ;LOAD TEST FPS
6480 033002 010204 MOV R2,R4 ;POINT TO TEST DATA
6481 033004 177014 ;
6482 ;406: LDCID (R4),AC0 ;*TEST INSTRUCTION (ACCORDING TO MODE)
6483 ;
6484 033006 170203 ;VERIFY STATUS
6485 033010 012700 000200 STFPS R3 ;SAVE FPS
6486 033014 170100 MOV #200,R0 ;SET FPP STATUS TO DOUBLE
LDFPS R0 ;

```

6487	033016	174011		STD	ACO.(R1)		;SAVE TEST RESULT INTO RECDST
6488	033020	016200	000016	MOV	16(R2),R0		;GET EXPECTED STATUS
6489	033024	020003		CMP	R0,R3		;VERIFY STATUS
6490	033026	001403		BEQ	3:		;BRANCH IF GOOD
6491	033030	104000		ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
6492	033032	000472		.WORD	472		;UNIQUE ERROR NUMBER
6493	033034	002013		.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
6494							
6495	033036	010204		3: MOV	R2,R4		;BAD FPS
6496	033040	062704	000004	ADD	44,R4		;POINT TO EXPECTED DATA
6497	033044	004767	147076	4: JSR	R7,DATVER		
6498	033050	005767	145764	TST	COUNT		;VERIFY DATA
6499	033054	001403		BEQ	5:		;BRANCH IF GOOD
6500	033056	104000		ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
6501	033060	000473		.WORD	473		;UNIQUE ERROR NUMBER
6502	033062	002013		.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
6503							
6504	033064	000162	000020	5: JMP	20(R2)		;BAD ACO
6505							;RETURN FROM TEST
6506							
6507	033070	012600					
6508	033072	012605		50: MOV	(SP)+,R0		;SAVE PC
6509	033074	104000		MOV	(SP)+,R5		;SAVE PS
6510	033076	000474		ERROR			;ALL ERRORS TO TRAP TO EMT VECTOR
6511	033100	002013		.WORD	474		;UNIQUE ERROR NUMBER
6512				.WORD	FPPERR		;ADDRESS OF ERROR MESSAGE
6513	033102	000167	177756	JMP	5:		;INSTRUCTION WASNT SUPPOSE TO TRAP
6514							;CONTINUE
6515	033106						
6516							
6517	033106						
6518							
6519							
6520							
6521	033106						
6522	033106	005267	145672				
6523							
6524	033112	005037	001042				
6525	033116	004767	001140				
6526	033122	123456	067012 025252				
6527	033130	171717					
6528	033132	000010					
6529	033134	142056	067012 025252				
6530	033142	171717					
6531	033144	007757					
6532	033146	007750					
6533							
6534	033150	005037	001042				
6535	033154	004767	001102				
6536	033160	023456	070123 100000				
6537	033166	000001					
6538	033170	000177					
6539	033172	077656	070123 100000				
6540	033200	000001					
6541	033202	007700					
6542	033204	007700					


```

6543      ;3/EXP=56
6544      CLR      @FLAG      ;NO INTERRUPTS
6545      JSR      R7,LXPSUB    ;DO TEST
6546      .WORD    55555,44444,33333,22222 ;ACO
6547      .WORD    56          ;EXP
6548      .WORD    53555,44444,33333,22222 ;RESULT
6549      .WORD    7757        ; TEST FPS
6550      .WORD    7740        ;RESULT FPS
6551      ;4/EXP=-151, ACO=UV
6552      CLR      @FLAG      ;NO INTERRUPTS
6553      JSR      R7,LXPSUB    ;DO TEST
6554      .WORD    100077,-1,-1,-2 ;ACO
6555      .WORD    -155        ;EXP
6556      .WORD    104677,-1,-1,-2 ;RESULT
6557      .WORD    7757        ; TEST FPS
6558      .WORD    7750        ;RESULT FPS
6559      ;5/EXP=-177
6560      CLR      @FLAG      ;NO INTERRUPTS
6561      JSR      R7,LXPSUB    ;DO TEST
6562      .WORD    177,-1,-1,-2 ;ACO
6563      .WORD    -177        ;EXP
6564      .WORD    377,-1,-1,-2 ;RESULT
6565      .WORD    7700        ; TEST FPS
6566      .WORD    7700        ;RESULT FPS
6567      ;6/EXP=-200, UNDERFLOW
6568      MOV      @1,@FLAG    ; INTERRUPTS
6569      JSR      R7,LXPSUB    ;DO TEST
6570      .WORD    30131,32334,35363,73031 ;ACO
6571      .WORD    -200        ;EXP
6572      .WORD    131,32334,35363,73031 ;RESULT
6573      .WORD    7740        ; TEST FPS
6574      .WORD    107744      ;RESULT FPS
6575      .WORD    12          ;FEC
6576      ;7/EXP=LARGEST NEGATIVE
6577      MOV      @1,@FLAG    ;EXPECT INTERRUPTS
6578      JSR      R7,LXPSUB    ;DO TEST
6579      .WORD    0,123,456,1 ;ACO
6580      .WORD    100000      ;EXP
6581      .WORD    40000,123,456,1 ;RESULT
6582      .WORD    2200        ; TEST FPS
6583      .WORD    102200      ;RESULT FPS
6584      .WORD    12          ;FEC
6585      ;8/EXP=-200, NEG. ACO
6586      MOV      @1,@FLAG    ; INTERRUPTS
6587      JSR      R7,LXPSUB    ;DO TEST
6588      .WORD    111111,100000,100000,-1 ;ACO
6589
6590
6591
6592
6593
6594
6595
6596
6597
6598
  
```

```

6599 033464 177777
6600 033466 177600
6601 033470 100111 100000 100000 .WORD -200 ;EXP
6602 033476 177777 .WORD 100111,100000,100000,-1 ;RESULT
6603 033500 002217 .WORD 2217 ; TEST FPS
6604 033502 102214 .WORD 102214 ;RESULT FPS
6605 033504 000012 .WORD 12 ;FEC
6606 ;9/EXP=-1743, FIU=0
6607 033506 012737 000002 001042 MOV #2,B#FLAG ;NO INTERRUPTS
6608 033514 004767 000542 JSR R7,LXPSUB ;DO TEST
6609 033520 123456 012346 012346 .WORD 123456,12346,12346,123 ;ACO
6610 033526 000123
6611 033530 176035 .WORD -1743 ;EXP
6612 033532 000000 000000 000000 .WORD 0,0,0,0 ;RESULT
6613 033540 000000
6614 033542 005700 .WORD 5700 ; TEST FPS
6615 033544 005704 .WORD 5704 ;RESULT FPS
6616 033546 000012 .WORD 12 ;FEC
6617 ;10/EXP=-16616, FID=1
6618 033550 012737 000002 001042 MOV #2,B#FLAG ;NO INTERRUPTS
6619 033556 004767 000500 JSR R7,LXPSUB ;DO TEST
6620 033562 000377 123456 065432 .WORD 377,123456,65432,1 ;ACO
6621 033570 000001
6622 033572 161162 .WORD -16616 ;EXP
6623 033574 074577 123456 065432 .WORD 74577,123456,65432,1 ;RESULT
6624 033602 000001
6625 033604 047700 .WORD 47700 ; TEST FPS
6626 033606 147700 .WORD 147700 ;RESULT FPS
6627 033610 000012 .WORD 12 ;FEC
6628 ;11/EXP=177, ACO=UNDEFINED VARIABLE
6629 033612 005037 001042 CLR B#FLAG ;NO INTERRUPTS
6630 033616 004767 000440 JSR R7,LXPSUB ;DO TEST
6631 033622 100177 177777 177777 .WORD 100177,-1,-1,-1 ;ACO
6632 033630 177777
6633 033632 000177 .WORD 177 ;EXP
6634 033634 177777 177777 177777 .WORD -1,-1,-1,-1 ;RESULT
6635 033642 177777
6636 033644 007700 .WORD 7700 ; TEST FPS
6637 033646 007710 .WORD 7710 ;RESULT FPS
6638 ;12/EXP=150 ACO=POS
6639 033650 005037 001042 CLR B#FLAG ;NO INTERRUPT
6640 033654 004767 000402 JSR R7,LXPSUB ;DO TEST
6641 033660 000200 000100 000200 .WORD 200,100,200,300 ;ACO
6642 033666 000300
6643 033670 000150 .WORD 150 ;EXP
6644 033672 072000 000100 000200 .WORD 72000,100,200,300 ;RESULT
6645 033700 000300
6646 033702 007717 .WORD 7717 ; TEST FPS
6647 033704 007700 .WORD 7700 ;RESULT FPS
6648 ;13/EXP=200, ACO=NEG
6649 033706 012737 000001 001042 MOV #1,B#FLAG
6650 033714 004767 000342 JSR R7,LXPSUB ;DO TEST
6651 033720 177777 177777 177777 .WORD -1,-1,-1,-1 ;ACO
6652 033726 177777
6653 033730 000200 .WORD 200 ;EXP
6654 033732 100177 177777 177777 .WORD 100177,-1,-1,-1 ;RESULT
  
```



```

6655 033740 177777
6656 033742 007705
6657 033744 107716
6658 033746 000010
6659
6660 033750 012737 000002 001042 :14/EXP=400, FID
6661 033756 004767 000300 MOV #2,0#FLAG ; INTERRUPT
6662 033762 000555 177777 177776 JSR R7,LXPSUB ; DO TEST
6663 033770 177775 .WORD 555,-1,-2,-3 ; ACO
6664 033772 000400 .WORD 400 ; EXP
6665 033774 040155 177777 177776 .WORD 40155,-1,-2,-3 ; RESULT
6666 034002 177775
6667 034004 047700 .WORD 47700 ; TEST FPS
6668 034006 147702 .WORD 147702 ; RESULT FPS
6669 034010 000010 .WORD 10 ; FEC
6670
6671 034012 012737 000000 001042 :15/EXP=11011 FIU=0
6672 034020 004767 000236 MOV #0,0#FLAG ; NO INTERRUPT
6673 034024 177773 177777 177776 JSR R7,LXPSUB ; DO TEST
6674 034032 177775 .WORD 177773,-1,-2,-3 ; ACO
6675 034034 011011 .WORD 11011 ; EXP
6676 034036 000000 000000 000000 .WORD 0,0,0,0 ; RESULT
6677 034044 000000
6678 034046 006700 .WORD 6700 ; TEST FPS
6679 034050 006706 .WORD 6706 ; RESULT FPS
6680
6681 034052 012737 000001 001042 :16/EXP=LARGEST POSITIVE
6682 034060 004767 000176 MOV #1,0#FLAG ; INTERRUPT
6683 034064 123456 000100 000100 JSR R7,LXPSUB ; DO TEST
6684 034072 000200 .WORD 123456,100,100,200 ; ACO
6685 034074 077777 .WORD 77777 ; EXP
6686 034076 137656 000100 000100 .WORD 137656,100,100,200 ; RESULT
6687 034104 000200
6688 034106 007740 .WORD 7740 ; TEST FPS
6689 034110 107752 .WORD 107752 ; RESULT FPS
6690 034112 000010 .WORD 10 ; FEC
6691
6692 034114 005037 001042 :17/FLOATING
6693 034120 004767 000136 CLR 0#FLAG ; NO INTERRUPT
6694 034124 123456 023465 000555 JSR R7,LXPSUB ; DO TEST
6695 034132 000444 .WORD 123456,23465,555,444 ; ACO
6696 034134 000050 .WORD 50 ; EXP
6697 034136 152056 023465 000555 .WORD 152056,23465,555,444 ; RESULT
6698 034144 000444
6699 034146 007500 .WORD 7500 ; TEST FPS
6700 034150 007510 .WORD 7510 ; RESULT FPS
6701
6702 034152 012737 000001 001042 :18/FLOATING UNDERFLOW
6703 034160 004767 000076 MOV #1,0#FLAG ; INTERRUPT
6704 034164 000333 000444 000555 JSR R7,LXPSUB ; DO TEST
6705 034172 000666 .WORD 333,444,555,666 ; ACO
6706 034174 177600 .WORD -200 ; EXP
6707 034176 000133 000444 000555 .WORD 133,444,555,666 ; RESULT
6708 034204 000666
6709 034206 007500 .WORD 7500 ; TEST FPS
6710 034210 107504 .WORD 107504 ; RESULT FPS

```

```

6711 034212 000012 .WORD 12 ;FEC
6712 ;19/FLOATING OVERFLOW
6713 034214 012737 000001 001042 MOV #1,0#FLAG ;INTERRUPT
6714 034222 004767 000034 JSR R7,LXPSUB ;DO TEST
6715 034226 012346 000123 000345 .WORD 12346,123,345,456 ;ACO
6716 034234 000456
6717 034236 000400 .WORD 400 ;EXP
6718 034240 040146 000123 000345 .WORD 40146,123,345,456 ;RESULT
6719 034246 000456
6720 034250 007400 .WORD 7400 ;TEST FPS
6721 034252 107402 .WORD 107402 ;RESULT FPS
6722 034254 000010 .WORD 10 ;FEC
6723 ;
6724 ;
6725 034256 000167 000250 JMP HOP20 ;GET OVER SUBROUTINE
6726 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
6727 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
6728 ;LDEXP
6729 ;
6730 ; ACO
6731 ; EXPONENT
6732 ; RESULT
6733 ; FPS BEFORE EXECUTION
6734 ; FPS AFTER EXECUTION
6735 ; (FEC)
6736 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
6737 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
6738 ;
6739 034262 012602 LXPSUB: MOV (SP)+,R2 ; RETURN ADDRESS TO USE AS POINTER
6740 034264 012737 034352 000244 MOV #50,0#FPVEC ;REDIRECT TRAP VECTOR
6741 034272 012701 001126 MOV #RECDST,R1 ;POINT TO RESULT AREA
6742 034276 012700 000200 MOV #200,R0 ;SET FPS TO DOUBLE
6743 034302 170100 LDFPS R0 ;
6744 034304 010204 MOV R2,R4 ;POINT TO ACO DATA
6745 034306 172414 LDD (R4),ACO ;LOAD ACO
6746 034310 016200 000022 MOV 22(R2),R0 ;GET TEST FPS
6747 034314 170100 LDFPS R0 ;LOAD TEST FPS
6748 034316 016204 000010 MOV 10(R2),R4 ;POINT TO TEST DATA
6749 ;
6750 034322 176404 40$: LDEXP R4,ACO ;*TEST INSTRUCTION (ACCORDING TO MODE)
6751 034324 170327 1$: STST (PC)+ ;WAIT FOR POSSIBLE FPA TRAP.
6752 034326 000000 .WORD 0 ;STORE STATUS HERE
6753 ;
6754 ;
6755 ;INSTRUCTION DIDNT TRAP
6756 034330 032737 000001 001042 BIT #1,0#FLAG ;VERIFY A NO TRAP CONDITION
6757 034336 001426 BEQ 2$ ;BRANCH IF GOOD
6758 034340 104000 ERROR ;ALL ERRORS TO TRAP TO EMT VECTOR
6759 034342 000475 .WORD 475 ;UNIQUE ERROR NUMBER
6760 034344 002013 .WORD FPPERR ;ADDRESS OF ERROR MESSAGE
6761 ;INSTRUCTION SHOULD HAVE TRAPPED
6762 034346 000167 000042 JMP 2$ ;REJOIN CODE
6763 ;
6764 ;INSTRUCTION TRAPPED
6765 034352 032737 000001 001042 50$: BIT #1,0#FLAG ;SEE IF EXPECTING A TRAP
6766 034360 001005 BNE 51$ ;BRANCH IF EXPECTING A TRAP

```



```

6767 034362 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
6768 034364 000476          .WORD          476          ;UNIQUE ERROR NUMBER
6769 034366 002013          .WORD          FPPERR        ;ADDRESS OF ERROR MESSAGE
6770                                     ;INSTRUCTION WASNT SUPPOSE TO TRAP
6771 034370 000167 000020    51$: JMP          2$          ;REJOIN CODE
6772 034374 012604          MOV          (SP)+,R4        ;SEE IF PC = INSTRUCTION
6773 034376 005726          TST          (SP)+          ;CLEAN UP STACK
6774 034400 022704 034324    CMP          #1$,R4        ;
6775 034404 001403          BEQ          2$          ;BRANCH IF GOOD COMPARE
6776 034406 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
6777 034410 000477          .WORD          477          ;UNIQUE ERROR NUMBER
6778 034412 002013          .WORD          FPPERR        ;ADDRESS OF ERROR MESSAGE
6779                                     ;PC WAS INCORRECT
6780
6781 ;COMMON CODE FOR TRAP AND NO TRAP
6782 ;VERIFY STATUS
6783 034414 170203          2$: STFPS          R3          ;SAVE FPS
6784 034416 012700 000200    MOV          #200,R0        ;SETUP FPS
6785 034422 170100          LDFPS          R0          ;FPS=200
6786 034424 174011          STD          ACO,(R1)        ;GET RESULT
6787 034426 016200 000024    MOV          24(R2),R0      ;GET EXPECTED STATUS
6788 034432 020003          CMP          R0,R3          ;VERIFY STATUS
6789 034434 001403          BEQ          3$          ;BRANCH IF GOOD
6790 034436 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
6791 034440 000500          .WORD          500          ;UNIQUE ERROR NUMBER
6792 034442 002013          .WORD          FPPERR        ;ADDRESS OF ERROR MESSAGE
6793                                     ;BAD FPS
6794 034444 010204          3$: MOV          R2,R4        ;POINT TO EXPECTED DATA
6795 034446 062704 000012    ADD          #12,R4        ;
6796 034452 004767 145470    4$: JSR          R7,DATVER  ;VERIFY DATA
6797 034456 005767 144356    TST          COUNT        ;
6798 034462 001403          BEQ          5$          ;BRANCH IF GOOD
6799 034464 104000          ERROR          ;ALL ERRORS TO TRAP TO EMT VECTOR
6800 034466 000501          .WORD          501          ;UNIQUE ERROR NUMBER
6801 034470 002013          .WORD          FPPERR        ;ADDRESS OF ERROR MESSAGE
6802                                     ;BAD ACO
6803 034472 005737 001042    5$: TST          #0FLAG      ;SEE IF NEED TO CHECK FEC
6804 034476 001002          BNE          7$          ;BRANCH IF NEED TO CHECK
6805 034500 000162 000026    JMP          26(R2)        ;RETURN FROM TEST
6806                                     ;
6807 034504 012704 001106    ;VERIFY FEC          ;POINT TO FEC AREA
6808 034510 170314          7$: MOV          #RECFEC,R4  ;SAVE FEC
6809 034512 021462 000026    STST          (R4)        ;VERIFY FEC FOR OVERFLOW
6810 034516 001403          CMP          (R4),26(R2)    ;BRANCH IF GOOD
6811 034520 104000          BEQ          8$          ;ALL ERRORS TO TRAP TO EMT VECTOR
6812 034522 000502          ERROR          ;UNIQUE ERROR NUMBER
6813 034524 002013          .WORD          502          ;ADDRESS OF ERROR MESSAGE
6814                                     ;BAD FEC
6815 034526 000162 000030    8$: JMP          30(R2)    ;RETURN FROM TEST
6816                                     ;
6817 034532          ;HOP20:
6818
6819 034532          MSCD:
6820          ;*****
6821          ;*TEST 107          TEST STCDI, STCDL
6822          ;*****

```

6823	034532			TST107:		
6824	034532	005267	144246	INC	\$TESTN	;INCREMENT TEST NUMBER
6825				;1/ACO=0, INT		
6826	034536	005037	001042	CLR	\$FLAG	;NO INTERRUPTS
6827	034542	004767	000610	JSR	R7,SCDSUB	;DO TEST
6828	034546	000177	000000	.WORD	0177,0,0,0	;ACO
6829	034554	000000				
6830	034556	000000	177777	.WORD	0,-1	;RESULT
6831	034562	007640		.WORD	7640	;TEST FPS
6832	034564	007644		.WORD	7644	;RESULT FPS
6833				;2/ACO=-0, LONG		
6834	034566	005037	001042	CLR	\$FLAG	;INTERRUPT
6835	034572	004767	000560	JSR	R7,SCDSUB	;DO TEST
6836	034576	100177	177777	.WORD	100177,-1,-1,-1	;ACO
6837	034604	177777				
6838	034606	000000	000000	.WORD	0,0	;RESULT
6839	034612	007700		.WORD	7700	;TEST FPS
6840	034614	007704		.WORD	7704	;RESULT FPS
6841				;3/EXP=100, LONG		
6842	034616	005037	001042	CLR	\$FLAG	;NO INTERRUPT
6843	034622	004767	000530	JSR	R7,SCDSUB	;DO TEST
6844	034626	020000	000000	.WORD	20000,0,0,0	;ACO
6845	034634	000000				
6846	034636	000000	000000	.WORD	0,0	;RESULT
6847	034642	000300		.WORD	300	;TEST FPS
6848	034644	000304		.WORD	304	;RESULT FPS
6849				;4/EXP=200, BAISED 0, INT, ROUND		
6850	034646	005037	001042	CLR	\$FLAG	;INTERRUPT
6851	034652	004767	000500	JSR	R7,SCDSUB	;DO TEST
6852	034656	140177	177777	.WORD	140177,177777,1,1	;ACO
6853	034664	000001				
6854	034666	000000	000000	.WORD	0,0	;RESULT
6855	034672	007700		.WORD	7700	;TEST FPS
6856	034674	007704		.WORD	7704	;RESULT FPS
6857				;5/LONG		
6858	034676	005037	001042	CLR	\$FLAG	;INTERRUPT
6859	034702	004767	000450	JSR	R7,SCDSUB	;DO TEST
6860	034706	047667	075757	.WORD	47667,75757,157737,167773	;ACO
6861	034714	167773				
6862	034716	055675	173757	.WORD	55675,173757	;RESULT
6863	034722	007717		.WORD	7717	;TEST FPS
6864	034724	007700		.WORD	7700	;RESULT FPS
6865				;6/LONG, EXP=2**32		
6866	034726	005037	001042	CLR	\$FLAG	;NO INTERRUPT
6867	034732	004767	000420	JSR	R7,SCDSUB	;DO TEST
6868	034736	046400	000000	.WORD	46400,0,0,0	;ACO
6869	034744	000000				
6870	034746	001000	000000	.WORD	1000,0	;RESULT
6871	034752	007700		.WORD	7700	;TEST FPS
6872	034754	007700		.WORD	7700	;RESULT FPS
6873				;7/LONG, EXP>2**32		
6874	034756	012737	000001	MOV	#1,\$FLAG	;INTERRUPT
6875	034764	004767	000366	JSR	R7,SCDSUB	;DO TEST
6876	034770	077607	000000	.WORD	77607,0,0,0	;ACO
6877	034776	000000				
6878	035000	000000	000000	.WORD	0,0	;RESULT

6879	035004	007700			.WORD 7700	; TEST FPS
6880	035006	107705			.WORD 107705	; RESULT FPS
6881					;8/INT, EXP=2**15	
6882	035010	005037	001042		CLR @FLAG	; NO INTERRUPTS
6883	035014	004767	000336		JSR R7,SCDSUB	; DO TEST
6884	035020	043200	000000	000000	.WORD 43200,0,0,0	; ACO
6885	035026	000000				
6886	035030	010000	177777		.WORD 10000,-1	; RESULT
6887	035034	007600			.WORD 7600	; TEST FPS
6888	035036	007600			.WORD 7600	; RESULT FPS
6889					;9/INT, EXP>2**15	
6890	035040	012737	000001	001042	MOV @1,@FLAG	; INTERRUPT
6891	035046	004767	000304		JSR R7,SCDSUB	; DO TEST
6892	035052	077777	177777	177777	.WORD 77777,-1,-1,-1	; ACO
6893	035060	177777				
6894	035062	000000	177777		.WORD 0,-1	; RESULT
6895	035066	007600			.WORD 7600	; TEST FPS
6896	035070	107605			.WORD 107605	; RESULT FPS
6897					;10/INT, EXP>2**15, FID	
6898	035072	012737	000000	001042	MOV @0,@FLAG	; NO INTERRUPT
6899	035100	004767	000252		JSR R7,SCDSUB	; DO TEST
6900	035104	043300	000000	000000	.WORD 43300,0,0,0	; ACO
6901	035112	000000				
6902	035114	000000	014000		.WORD 0,14000	; RESULT
6903	035120	047700			.WORD 47700	; TEST FPS
6904	035122	047700			.WORD 47700	; RESULT FPS
6905					;11/INT, EXP>2**15, FIC=0	
6906	035124	012737	000000	001042	MOV @0,@FLAG	; NO INTERRUPT
6907	035132	004767	000220		JSR R7,SCDSUB	; DO TEST
6908	035136	143300	177777	177777	.WORD 143300,-1,-1,-1	; ACO
6909	035144	177777				
6910	035146	177777	163741		.WORD -1,163741	; RESULT
6911	035152	007300			.WORD 7300	; TEST FPS
6912	035154	007310			.WORD 7310	; RESULT FPS
6913					;12/LONG, EXP>2**32, FID	
6914	035156	012737	000002	001042	MOV @2,@FLAG	; INTERRUPT
6915	035164	004767	000166		JSR R7,SCDSUB	; DO TEST
6916	035170	050100	000000	000000	.WORD 50100,0,0,0	; ACO
6917	035176	000000				
6918	035200	000000	000000		.WORD 0,0	; RESULT
6919	035204	047700			.WORD 47700	; TEST FPS
6920	035206	147705			.WORD 147705	; RESULT FPS
6921					;13/LONG, EXP>2**32, FIC=0	
6922	035210	012737	000000	001042	MOV @0,@FLAG	; NO INTERRUPT
6923	035216	004767	000134		JSR R7,SCDSUB	; DO TEST
6924	035222	050377	177777	177777	.WORD 50377,-1,-1,-1	; ACO
6925	035230	177777				
6926	035232	000000	000000		.WORD 0,0	; RESULT
6927	035236	007300			.WORD 7300	; TEST FPS
6928	035240	007305			.WORD 7305	; RESULT FPS
6929					;14/LONG, EXP<0	
6930	035242	005037	001042		CLR @FLAG	; NO INTERRUPTS
6931	035246	004767	000104		JSR R7,SCDSUB	; DO TEST
6932	035252	100200	177777	177777	.WORD 100200,-1,-1,-1	; ACO
6933	035260	177777				
6934	035262	000000	000000		.WORD 0,0	; RESULT

```

6935 035266 007757 .WORD 7757 ; TEST FPS
6936 035270 007744 .WORD 7744 ; RESULT FPS
6937 ;15/INT, EXP<0
6938 035272 005037 001042 CLR 00FLAG ; NO INTERRUPTS
6939 035276 004767 000054 JSR R7,SCDSUB ; DO TEST
6940 035302 037700 177777 177777 .WORD 37700,-1,-1,-2 ; ACO
6941 035310 177776
6942 035312 000000 177777 .WORD 0,-1 ; RESULT
6943 035316 007600 .WORD 7600 ; TEST FPS
6944 035320 007604 .WORD 7604 ; RESULT FPS
6945 ;16/INT, EXP-10
6946 035322 005037 001042 CLR 00FLAG ; NO INTERRUPTS
6947 035326 004767 000024 JSR R7,SCDSUB ; DO TEST
6948 035332 004377 177777 177777 .WORD 4377,-1,-1,-1 ; ACO
6949 035340 177777
6950 035342 000000 177777 .WORD 0,-1 ; RESULT
6951 035346 007600 .WORD 7600 ; TEST FPS
6952 035350 007604 .WORD 7604 ; RESULT FPS
6953 ;
6954 ;
6955 035352 000167 000244 JMP HOP21 ; GET OVER SUBROUTINE
6956 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
6957 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
6958 ;STCDI, STCDL, STCFI, STCFL
6959 ;
6960 ; ACO
6961 ; RESULT
6962 ; FPS BEFORE EXECUTION
6963 ; FPS AFTER EXECUTION
6964 ; (FEC)
6965 ;
6966 ; TRAP ON CONVERSION FAILURE
6967 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
6968 ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
6969 ;
6970 035356 012602 SCDSUB: MOV (SP),R2 ; RETURN ADDRESS TO USE AS POINTER
6971 035360 012737 035452 000244 MOV 0501,00FPVEC ; REDIRECT TRAP VECTOR
6972 035366 012701 001130 MOV 0RECDST.2,R1 ; POINT TO RESULT AREA
6973 035372 012711 177777 MOV 0-1,(R1) ; PRELOAD RECEIVE DATA BUFFER
6974 035376 012741 177777 MOV 0-1,-(R1) ;
6975 035402 012700 000200 MOV 0200,R0 ; SET FPS TO DOUBLE
6976 035406 170100 LDFPS R0 ;
6977 035410 010204 MOV R2,R4 ; POINT TO ACO DATA
6978 035412 172414 LDD (R4),ACO ; LOAD ACO
6979 035414 016200 000014 MOV 14(R2),R0 ; GET TEST FPS
6980 035420 170100 LDFPS R0 ; LOAD TEST FPS
6981 ;
6982 035422 175411 404: STCDI ACO,(R1) ; TEST INSTRUCTION (ACCORDING TO MODE)
6983 035424 170327 14: STST (PC), ; WAIT FOR POSSIBLE FPA TRAP.
6984 035426 000000 .WORD 0 ; STORE STATUS HERE.
6985 ;
6986 ;
6987 ; INSTRUCTION DIDNT TRAP
6988 035430 032737 000001 001042 BIT 01,00FLAG ; VERIFY A NO TRAP CONDITION
6989 035436 001426 BEQ 24 ; BRANCH IF GOOD
6990 035440 104000 ERROR ; ALL ERRORS TO TRAP TO ENT VECTOR

```



```
6991 035442 000503 .WORD 503
6992 035444 002013 .WORD FPPERR
6993
6994 035446 000167 000042 JMP 24
6995
6996 ; INSTRUCTION TRAPPED
6997 035452 032737 000001 001042 504: BIT 01,00FLAG
6998 035460 001005 BNE 514
6999 035462 104000 ERROR
7000 035464 000504 .WORD 504
7001 035466 002013 .WORD FPPERR
7002
7003 035470 000167 000020 JMP 24
7004 035474 012604 514: MOV (SP)+,R4
7005 035476 005726 TST (SP)+
7006 035500 022704 035424 CMP 014,R4
7007 035504 001403 BEQ 24
7008 035506 104000 ERROR
7009 035510 000505 .WORD 505
7010 035512 002013 .WORD FPPERR
7011
7012
7013 ; COMMON CODE FOR TRAP AND NO TRAP
7014 ; VERIFY STATUS
7015 035514 170203 24: STFPS R3
7016 035516 016200 000016 MOV 16(R2),R0
7017 035522 020003 CMP R0,R3
7018 035524 001403 BEQ 34
7019 035526 104000 ERROR
7020 035530 000506 .WORD 506
7021 035532 002013 .WORD FPPERR
7022
7023 035534 010204 34: MOV R2,R4
7024 035536 062704 000010 ADD 010,R4
7025 035542 004767 144362 44: JSR R7,DATVFR
7026 035546 005767 143266 TST COUNT
7027 035552 001403 BEQ 54
7028 035554 104000 ERROR
7029 035556 000507 .WORD 507
7030 035560 002013 .WORD FPPERR
7031
7032 035562 005737 001042 54: TST 00FLAG
7033 035566 001002 BNE 74
7034 035570 000162 000020 JMP 20(R2)
7035
7036 035574 012704 001106 ; VERIFY FEC
7037 035600 170314 74: MOV 00RECFEC,R4
7038 035602 021427 000006 STS (R4)
7039 035606 001403 CMP (R4),06
7040 035610 104000 BEQ 84
7041 035612 000510 ERROR
7042 035614 002013 .WORD 510
7043 .WORD FPPERR
7044 035616 000162 000020 84: JMP 20(R2)
7045
7046 035622 ; HOP21:
; INSTRUCTION SHOULD HAVE TRAPPED
; REJOIN CODE
; SEE IF EXPECTING A TRAP
; BRANCH IF EXPECTING A TRAP
; ALL ERRORS TO TRAP TO EMT VECTOR
; UNIQUE ERROR NUMBER
; ADDRESS OF ERROR MESSAGE
; INSTRUCTION WASNT SUPPOSE TO TRAP
; REJOIN CODE
; SEE IF PC = INSTRUCTION
; CLEAN UP STACK
; BRANCH IF GOOD COMPARE
; ALL ERRORS TO TRAP TO EMT VECTOR
; UNIQUE ERROR NUMBER
; ADDRESS OF ERROR MESSAGE
; PC WAS INCORRECT
; SAVE FPS
; GET EXPECTED STATUS
; VERIFY STATUS
; BRANCH IF GOOD
; ALL ERRORS TO TRAP TO EMT VECTOR
; UNIQUE ERROR NUMBER
; ADDRESS OF ERROR MESSAGE
; BAD FPS
; POINT TO EXPECTED DATA
; VERIFY DATA
; BRANCH IF GOOD
; ALL ERRORS TO TRAP TO EMT VECTOR
; UNIQUE ERROR NUMBER
; ADDRESS OF ERROR MESSAGE
; BAD ACO
; SEE IF NEED TO CHECK FEC
; BRANCH IF NEED TO CHECK
; RETURN FROM TEST
; POINT TO FEC AREA
; SAVE FEC
; VERIFY FEC FOR OVERFLOW
; BRANCH IF GOOD
; ALL ERRORS TO TRAP TO EMT VECTOR
; UNIQUE ERROR NUMBER
; ADDRESS OF ERROR MESSAGE
; BAD FEC
; RETURN FROM TEST
```

```

7047
7048 035622
7049
7050
7051
7052 035622
7053 035622 005267 143156
7054
7055 035626 005037 001042
7056 035632 004767 177520
7057 035636 044541 052525 177777
7058 035644 177777
7059 035646 000003 102525
7060 035652 007517
7061 035654 007500
7062
7063 035656 005037 001042
7064 035662 004767 177470
7065 035666 002300 177777 177777
7066 035674 177777
7067 035676 000000 177777
7068 035702 007400
7069 035704 007404
7070
7071 035706 012737 000001 001042
7072 035714 004767 177436
7073 035720 070000 177777 177777
7074 035726 177777
7075 035730 000000 000000
7076 035734 007540
7077 035736 107545
7078
7079 035740 005037 001042
7080 035744 004767 177406
7081 035750 052000 000000 177777
7082 035756 177777
7083 035760 000000 177777
7084 035764 047000
7085 035766 047005
7086
7087
7088
7089 J35770
7090
7091
7092
7093 035770
7094 035770 005267 143010
7095
7096 035774 004767 000154
7097 036000 020000 000000 000000
7098 036006 000000
7099 036010 177700
7100 036012 007740
7101 036014 007750
7102

; MSCF:
;*****
; TEST 110 TEST STCFI, STCFL
;*****
TST110:
;1/LONG INC #TESTN ;INCREMENT TEST NUMBER
EXP =30
CLR #FLAG ;NO INTERRUPTS
JSR R7,SCDSUB ;DO TEST
;WORD 44541,52525,-1,-1 ;ACO

;WORD 3,102525 ;RESULT
;WORD 7517 ;TEST FPS
;WORD 7500 ;RESULT FPS

;2/INT, EXP<0
CLR #FLAG ;NO INTERRUPTS
JSR R7,SCDSUB ;DO TEST
;WORD 2300,-1,-1,-1 ;ACO

;WORD 0,-1 ;RESULT
;WORD 7400 ;TEST FPS
;WORD 7404 ;RESULT FPS

;3/LONG, EXP>=2*32
MOV #1,#FLAG ;INTERRUPT
JSR R7,SCDSUB ;DO TEST
;WORD 70000,-1,-1,-1 ;ACO

;WORD 0,0 ;RESULT
;WORD 7540 ;TEST FPS
;WORD 107545 ;RESULT FPS

;4/INT,EXP=5, FIC=0, FID=1
CLR #FLAG ;NO INTERRUPTS
JSR R7,SCDSUB ;DO TEST
;WORD 52000,0,-1,-1 ;ACO

;WORD 0,-1 ;RESULT
;WORD 47000 ;TEST FPS
;WORD 47005 ;RESULT FPS

;

; MSXP:
;*****
; TEST 111 TEST STEXP
;*****
TST111:
;1/EXP=100 INC #TESTN ;INCREMENT TEST NUMBER
JSR R7, SXPSUB ;DO TEST
;WORD 20000,0,0,0 ;ACO

;WORD -100 ;RESULT
;WORD 7740 ;TEST FPS
;WORD 7750 ;RESULT FPS

;2/EXP=201 FLOAT, NEG

```



```

7103 036016 004767 000132      JSR      R7,SXPSUB      ;DO TEST
7104 036022 140377 177777      .WORD    140377,-1,-1,0      ;ACO
7105 036030 000000
7106 036032 000001      .WORD    1      ;RESULT
7107 036034 007500      .WORD    7500      ;TEST FPS
7108 036036 007500      .WORD    7500      ;RESULT FPS
7109      ;3/EXP=-177
7110 036040 004767 000110      JSR      R7,SXPSUB      ;DO TEST
7111 036044 000177 177777      .WORD    177,-1,-1,-1      ;ACO
7112 036052 177777
7113 036054 177600      .WORD    177600      ;RESULT
7114 036056 007700      .WORD    7700      ;TEST FPS
7115 036060 007710      .WORD    7710      ;RESULT FPS
7116      ;4/EXP=-100
7117 036062 004767 000065      JSR      R7,SXPSUB      ;DO TEST
7118 036066 020000 000000 177777 .WORD    20000,0,-1,-1      ;ACO
7119 036074 177777
7120 036076 177700      .WORD    -100      ;RESULT
7121 036100 040200      .WORD    40200      ;TEST FPS
7122 036102 040210      .WORD    40210      ;RESULT FPS
7123      ;5/EXP=200
7124 036104 004767 000044      JSR      R7,SXPSUB      ;DO TEST
7125 036110 040000 000000 000000 .WORD    40000,0,0,0      ;ACO
7126 036116 000000
7127 036120 000000      .WORD    0      ;RESULT
7128 036122 007700      .WORD    7700      ;TEST FPS
7129 036124 007704      .WORD    7704      ;RESULT FPS
7130      ;6/EXP=0
7131 036126 004767 000022      JSR      R7,SXPSUB      ;DO TEST
7132 036132 000177 177777 177777 .WORD    177,-1,-1,-1      ;ACO
7133 036140 177777
7134 036142 177600      .WORD    177600      ;RESULT
7135 036144 000000      .WORD    0      ;TEST FPS
7136 036146 000010      .WORD    10      ;RESULT FPS
7137
7138      ;
7139 036150 000167 000120      JMP      HOP22      ;GET OVER SUBROUTINE
7140      ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
7141      ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
7142      ;STEXP
7143      ;
7144      ;      ACO
7145      ;      EXPONENT RESULT
7146      ;      FPS BEFORE EXECUTION
7147      ;      FPS AFTER EXECUTION
7148      ;
7149      ;NO TRAPS CAN OCCUR
7150      ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
7151      ;XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
7152      ;
7152 036154 012602      SXPSUB: MOV      (SP),R2      ; RETURN ADDRESS TO USE AS POINTER
7153 036156 012737 036256 000244   MOV      #504,R0FPVEC      ;REDIRECT TRAP VECTOR
7154 036164 012701 001126           MOV      @RECDST,R1      ;POINT TO RESULT AREA
7155 036170 012700 000200           MOV      #200,R0      ;SET FPS TO DOUBLE
7156 036174 170100           LDFPS   R0      ;
7157 036176 010204           MOV      R2,R4      ;POINT TO ACO DATA
7158 036200 172414           LDD      (R4),ACO      ;LOAD ACO
  
```

Address	Hex	Dec	Label	Op	Op2	Op3	Op4	Op5	Op6	Op7	Op8	Op9	Op10	Op11	Op12	Op13	Op14	Op15	Op16	Op17	Op18	Op19	Op20	Op21	Op22	Op23	Op24	Op25	Op26	Op27	Op28	Op29	Op30	Op31	Op32	Op33	Op34	Op35	Op36	Op37	Op38	Op39	Op40	Op41	Op42	Op43	Op44	Op45	Op46	Op47	Op48	Op49	Op50	Op51	Op52	Op53	Op54	Op55	Op56	Op57	Op58	Op59	Op60	Op61	Op62	Op63	Op64	Op65	Op66	Op67	Op68	Op69	Op70	Op71	Op72	Op73	Op74	Op75	Op76	Op77	Op78	Op79	Op80	Op81	Op82	Op83	Op84	Op85	Op86	Op87	Op88	Op89	Op90	Op91	Op92	Op93	Op94	Op95	Op96	Op97	Op98	Op99	Op100	Op101	Op102	Op103	Op104	Op105	Op106	Op107	Op108	Op109	Op110	Op111	Op112	Op113	Op114	Op115	Op116	Op117	Op118	Op119	Op120	Op121	Op122	Op123	Op124	Op125	Op126	Op127	Op128	Op129	Op130	Op131	Op132	Op133	Op134	Op135	Op136	Op137	Op138	Op139	Op140	Op141	Op142	Op143	Op144	Op145	Op146	Op147	Op148	Op149	Op150	Op151	Op152	Op153	Op154	Op155	Op156	Op157	Op158	Op159	Op160	Op161	Op162	Op163	Op164	Op165	Op166	Op167	Op168	Op169	Op170	Op171	Op172	Op173	Op174	Op175	Op176	Op177	Op178	Op179	Op180	Op181	Op182	Op183	Op184	Op185	Op186	Op187	Op188	Op189	Op190	Op191	Op192	Op193	Op194	Op195	Op196	Op197	Op198	Op199	Op200	Op201	Op202	Op203	Op204	Op205	Op206	Op207	Op208	Op209	Op210	Op211	Op212	Op213	Op214	Op215	Op216	Op217	Op218	Op219	Op220	Op221	Op222	Op223	Op224	Op225	Op226	Op227	Op228	Op229	Op230	Op231	Op232	Op233	Op234	Op235	Op236	Op237	Op238	Op239	Op240	Op241	Op242	Op243	Op244	Op245	Op246	Op247	Op248	Op249	Op250	Op251	Op252	Op253	Op254	Op255	Op256	Op257	Op258	Op259	Op260	Op261	Op262	Op263	Op264	Op265	Op266	Op267	Op268	Op269	Op270	Op271	Op272	Op273	Op274	Op275	Op276	Op277	Op278	Op279	Op280	Op281	Op282	Op283	Op284	Op285	Op286	Op287	Op288	Op289	Op290	Op291	Op292	Op293	Op294	Op295	Op296	Op297	Op298	Op299	Op300	Op301	Op302	Op303	Op304	Op305	Op306	Op307	Op308	Op309	Op310	Op311	Op312	Op313	Op314	Op315	Op316	Op317	Op318	Op319	Op320	Op321	Op322	Op323	Op324	Op325	Op326	Op327	Op328	Op329	Op330	Op331	Op332	Op333	Op334	Op335	Op336	Op337	Op338	Op339	Op340	Op341	Op342	Op343	Op344	Op345	Op346	Op347	Op348	Op349	Op350	Op351	Op352	Op353	Op354	Op355	Op356	Op357	Op358	Op359	Op360	Op361	Op362	Op363	Op364	Op365	Op366	Op367	Op368	Op369	Op370	Op371	Op372	Op373	Op374	Op375	Op376	Op377	Op378	Op379	Op380	Op381	Op382	Op383	Op384	Op385	Op386	Op387	Op388	Op389	Op390	Op391	Op392	Op393	Op394	Op395	Op396	Op397	Op398	Op399	Op400	Op401	Op402	Op403	Op404	Op405	Op406	Op407	Op408	Op409	Op410	Op411	Op412	Op413	Op414	Op415	Op416	Op41
---------	-----	-----	-------	----	-----	-----	-----	-----	-----	-----	-----	-----	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	------


```
7195 .MCALL IDMSG,ENDPAS
7196 .SBTTL END OF PASS ROUTINE
7197
7198 ;*****
7199 ;*INCREMENT THE PASS NUMBER ($PASS)
7200 ;*INDICATE END-OF-PROGRAM AFTER 1 PASSES THRU THE PROGRAM
7201 ;*IF THERES A MONITOR GO TO IT
7202 ;*IF THERE ISN'T JUMP TO RESTART
7203
7204 036274 $EOP:
7205 036274 005767 142506 TST $PASS ;ONLY TYPE MESSAGE AT END OF FIRST PASS
7206 036300 001002 BNE SKIPID ;IF >0 THEN SKIP THE ID MESSAGE
7207 036302 104401 036402 TYPE .MSG1 ;ELSE TYPE THE ID MESSAGE
7208 036306
7209 036306 005267 142474 SKIPID: INC $PASS ;;INCREMENT THE PASS NUMBER
7210 036312 042767 100000 142466 BIC #100000,$PASS ;;DON'T ALLOW A NEG. NUMBER
7211 036320 005327 DEC (PC). ;;LOOP?
7212 036322 000001 $EOPCT: .WORD 1
7213 036324 003022 BGT $DOAGN ;;YES
7214 036326 012737 MOV (PC)+,$(PC). ;;RESTORE COUNTER
7215 036330 000001 $ENDCT: .WORD 1
7216 036332 036322 $EOPCT
7217 036334 104401 036456 TYPE .MSG2
7218 036340 016746 142442 MOV $PASS,-(SP) ;;SAVE $PASS FOR TYPEOUT
7219 036344 104405 TYPDS ;;GO TYPE--DECIMAL ASCII WITH SIGN
7220 036346 104401 036376 TYPE .$ENULL
7221 036352 013700 000042 $GET42: MOV $#42,R0 ;;GET MONITOR ADDRESS
7222 036356 001405 BEQ $DOAGN ;;BRANCH IF NO MONITOR
7223 036360 000005 RESET ;;CLEAR THE WORLD
7224 036362 004710 $ENDAD: JSR PC,(R0) ;;GO TO MONITOR
7225 036364 000240 NOP ;;SAVE ROOM
7226 036366 000240 NOP ;;FOR
7227 036370 000240 NOP ;;ACT11
7228 036372
7229 036372 000137 $DOAGN: JMP $(PC). ;;RETURN
7230 036374 002410 $RTNAD: .WORD RESTART
7231 036376 377 377 000 $ENULL: .BYTE -1,-1,0 ;;NULL CHARACTER STRING
7232 036402
7233 036402 005015 055103 042113 MSG1: .EVEN
7234 036410 026514 026502 020060 .ASCIZ <CR><LF>/CZKDL-B-0 KDJ11 FLOATING POINT DIAGNOSTIC/
7235 036416 042113 030512 020061
7236 036424 046106 040517 044524
7237 036432 043516 050040 044517
7238 036440 052116 042040 040511
7239 036446 047107 051517 044524
7240 036454 000103
7241 036456 005015 055103 042113 MSG2: .ASCIZ <CR><LF>/CZKDLB END PASS #/
7242 036464 041114 042440 042116
7243 036472 050040 051501 020123
7244 036500 000043
7245
7246 .EVEN
7247 .SBTTL TYPE ROUTINE
7248
7249 ;*****
7250 ;*ROUTINE TO TYPE ASCIZ MESSAGE. MESSAGE MUST TERMINATE WITH A 0 BYTE.
;*THE ROUTINE WILL INSERT A NUMBER OF NULL CHARACTERS AFTER A LINE FEED.
```

```
7251 ;*NOTE1: $NULL CONTAINS THE CHARACTER TO BE USED AS THE FILLER CHARACTER.
7252 ;*NOTE2: $FILLS CONTAINS THE NUMBER OF FILLER CHARACTERS REQUIRED.
7253 ;*NOTE3: $FILLC CONTAINS THE CHARACTER TO FILL AFTER.
7254 ;*
7255 ;*CALL:
7256 ;*1) USING A TRAP INSTRUCTION
7257 ;* TYPE ,MESADR ;:MESADR IS FIRST ADDRESS OF AN ASCIZ STRING
7258 ;*OR
7259 ;* TYPE
7260 ;* MESADR
7261 ;*
7262
7263 036502 105767 000343 $TYPE: TSTB $TPFLG ;:IS THERE A TERMINAL?
7264 036506 100002 BPL 1$ ;:BR IF YES
7265 036510 000000 HALT ;:HALT HERE IF NO TERMINAL
7266 036512 000430 BR 3$ ;:LEAVE
7267 036514 010046 1$: MOV RO,-(SP) ;:SAVE RO
7268 036516 017600 000002 MOV @2(SP),RO ;:GET ADDRESS OF ASCIZ STRING
7269 036522 122767 000001 142270 CMPB @APTENV,$ENV ;:RUNNING IN APT MODE
7270 036530 001011 BNE 62$ ;:NO,GO CHECK FOR APT CONSOLE
7271 036532 132767 000100 142261 BITB @APTSPool,$ENVM ;:SPOOL MESSAGE TO APT
7272 036540 001405 BEQ 62$ ;:NO,GO CHECK FOR CONSOLE
7273 036542 010067 000004 MOV RO,61$ ;:SETUP MESSAGE ADDRESS FOR APT
7274 036546 004767 001034 JSR PC,$ATY3 ;:SPOOL MESSAGE TO APT
7275 036552 000000 61$: .WORD 0 ;:MESSAGE ADDRESS
7276 036554 132767 000040 142237 62$: BITB @APTCSUP,$ENVM ;:APT CONSOLE SUPPRESSED
7277 036562 001003 BNE 60$ ;:YES,SKIP TYPE OUT
7278 036564 112046 2$: MOVB (RO),-(SP) ;:PUSH CHARACTER TO BE TYPED ONTO STACK
7279 036566 001005 BNE 4$ ;:BR IF IT ISN'T THE TERMINATOR
7280 036570 005726 TST (SP), ;:IF TERMINATOR POP IT OFF THE STACK
7281 036572 012600 60$: MOV (SP),RO ;:RESTORE RO
7282 036574 062716 000002 3$: ADD @2,(SP) ;:ADJUST RETURN PC
7283 036600 000002 RTI ;:RETURN
7284 036602 122716 000011 4$: CMPB @HT,(SP) ;:BRANCH IF <HT>
7285 036606 001430 BEQ 8$ ;:BRANCH IF NOT <CRLF>
7286 036610 122716 000200 CMPB @CRLF,(SP) ;:BRANCH IF NOT <CRLF>
7287 036614 001006 BNE 5$ ;:POP <CR><LF> EQUIV
7288 036616 005726 TST (SP), ;:TYPE A CR AND LF
7289 036620 104401 TYPE ;:
7290 036622 002077 $CRLF ;:
7291 036624 105067 000202 CLRB $CHARCNT ;:CLEAR CHARACTER COUNT
7292 036630 000755 BR 2$ ;:GET NEXT CHARACTER
7293 036632 004767 000056 5$: JSR PC,$TYPEC ;:GO TYPE THIS CHARACTER
7294 036636 126726 000206 6$: CMPB $FILLC,(SP), ;:IS IT TIME FOR FILLER CHARS.?
7295 036642 001350 BNE 2$ ;:IF NO GO GET NEXT CHAR.
7296 036644 016746 000176 MOV $NULL,-(SP) ;:GET # OF FILLER CHARS. NEEDED
7297 ;:AND THE NULL CHAR.
7298 036650 105366 000001 7$: DECB 1(SP) ;:DOES A NULL NEED TO BE TYPED?
7299 036654 002770 BLT 6$ ;:BR IF NO--GO POP THE NULL OFF OF STACK
7300 036656 004767 000032 JSR PC,$TYPEC ;:GO TYPE A NULL
7301 036662 105367 000144 DECB $CHARCNT ;:DO NOT COUNT AS A COUNT
7302 036666 000770 BR 7$ ;:LOOP
7303
7304 ;HORIZONTAL TAB PROCESSOR
7305
7306 036670 112716 000040 8$: MOVB 0' ,(SP) ;:REPLACE TAB WITH SPACE
```



```
7307 036674 004767 000014 9$: JSR PC,$TYPEC ;;TYPE A SPACE
7308 036700 132767 000007 000124 BITB 47,$CHARCNT ;;BRANCH IF NOT AT
7309 036706 001372 BNE 9$ ;;TAB STOP
7310 036710 005726 TST (SP), 2$ ;;POP SPACE OFF STACK
7311 036712 000724 BR 2$ ;;GET NEXT CHARACTER
7312 036714 $TYPEC:
7313 036714 105777 000116 TSTB 8$TKS ;;CHAR IN KYBD BUFFER? ;MJD001
7314 036720 100022 BPL 10$ ;;BR IF NOT ;MJD001
7315 036722 017746 000112 MOV 8$TKB,-(SP) ;;GET CHAR ;MJD001
7316 036726 042716 177600 BIC 8$177600,(SP) ;;STRIP EXTRANEIOUS BITS ;MJD001
7317 036732 122716 000023 CMPB 8$XOFF,(SP) ;;WAS CHAR XOFF ;MJD001
7318 036736 001012 BNE 102$ ;;BR IF NOT ;MJD001
7319 036740 101$:
7320 036740 105777 000072 TSTB 8$TKS ;;WAIT FOR CHAR ;MJD001
7321 036744 100375 BPL 101$ ;MJD001
7322 036746 117716 000066 MOV 8$TKB,(SP) ;;GET CHAR ;MJD001
7323 036752 042716 177600 BIC 8$177600,(SP) ;;STRIP IT ;MJD001
7324 036756 122716 000021 CMPB 8$XON,(SP) ;;WAS IT XON? ;MJD001
7325 036762 001366 BNE 101$ ;;BR IF NOT ;MJD001
7326 036764 102$:
7327 036764 005726 TST (SP), 10$ ;;FIX STACK ;MJD001
7328 036766 10$:
7329 036766 105777 000050 TSTB 8$TPS ;;WAIT UNTIL PRINTER IS READY ;MJD001
7330 036772 100375 BPL 10$ ;MJD001
7331 036774 116677 000002 000042 MOV 2(SP),8$TPB ;;LOAD CHAR TO BE TYPED INTO DATA REG. ;MJD001
7332 037002 122766 000015 000002 CMPB 8$CR,2(SP) ;;IS CHARACTER A CARRIAGE RETURN?
7333 037010 001003 BNE 1$ ;;BRANCH IF NO
7334 037012 105067 000014 CLRB $CHARCNT ;;YES--CLEAR CHARACTER COUNT
7335 037016 000406 BR $TYPEX ;;EXIT
7336 037020 122766 000012 000002 1$: CMPB 8$LF,2(SP) ;;IS CHARACTER A LINE FEED?
7337 037026 001402 BZQ $TYPEX ;;BRANCH IF YES
7338 037030 105227 INCB (PC), 0 ;;COUNT THE CHARACTER
7339 037032 000000 $CHARCNT: .WORD 0 ;;CHARACTER COUNT STORAGE
7340 037034 000207 $TYPEX: RTS PC
7341
7342 037036 177560 $TKS: .WORD 177560 ;;TTY KDB STATUS ;MJD001
7343 037040 177562 $TKB: .WORD 177562 ;;TTY KDB BUFFER ;MJD001
7344 037042 177564 $TPS: .WORD 177564 ;;TTY PRINTER STATUS REG. ADDRESS
7345 037044 177566 $TPB: .WORD 177566 ;;TTY PRINTER BUFFER REG. ADDRESS
7346 037046 000 $NULL: .BYTE 0 ;;CONTAINS NULL CHARACTER FOR FILLS
7347 037047 002 $FILLS: .BYTE 2 ;;CONTAINS # OF FILLER CHARACTERS REQUIRED
7348 037050 012 $FILLC: .BYTE 12 ;;INSERT FILL CHARS. AFTER A "LINE FEED"
7349 037051 000 $TPFLG: .BYTE 0 ;;"TERMINAL AVAILABLE" FLAG (BIT<07>-0=YES)
7350 037052 077 $QUES: .ASCII "?" ;;QUESTION MARK
7351 037053 012 000 $LF: .ASCIIZ <12> ;;LINEFEED
7352 037056 .EVEN
7353 .SBTTL CONVERT BINARY TO DECIMAL AND TYPE ROUTINE
7354
7355 ;*****
7356 ;*THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 5-DIGIT
7357 ;*SIGNED DECIMAL (ASCII) NUMBER AND TYPE IT. DEPENDING ON WHETHER THE
7358 ;*NUMBER IS POSITIVE OR NEGATIVE A SPACE OR A MINUS SIGN WILL BE TYPED
7359 ;*BEFORE THE FIRST DIGIT OF THE NUMBER. LEADING ZEROS WILL ALWAYS BE
7360 ;*REPLACED WITH SPACES.
7361 ;*CALL:
7362 ;* MOV NUM,-(SP) ;;PUT THE BINARY NUMBER ON THE STACK
```

```
7363                                     ;*      TYPDS                                     ;:GO TO THE ROUTINE
7364                                     $TYPDS:
7365 037056                                MOV      R0,-(SP)                                ;:PUSH R0 ON STACK
7366 037056 010046                        MOV      R1,-(SP)                                ;:PUSH R1 ON STACK
7367 037060 010146                        MOV      R2,-(SP)                                ;:PUSH R2 ON STACK
7368 037062 010246                        MOV      R3,-(SP)                                ;:PUSH R3 ON STACK
7369 037064 010346                        MOV      R5,-(SP)                                ;:PUSH R5 ON STACK
7370 037066 010546                        MOV      #20200,-(SP)                        ;:SET BLANK SWITCH AND SIGN
7371 037070 012746 020200                MOV      20(SP),R5                        ;:GET THE INPUT NUMBER
7372 037074 016605 000020                BPL      1$                                ;:BR IF INPUT IS POS.
7373 037100 100004                        NEG      R5                                ;:MAKE THE BINARY NUMBER POS.
7374 037102 005405                        MOV      #'-,1(SP)                        ;:MAKE THE ASCII NUMBER NEG.
7375 037104 112766 000055 000001        1$:      CLRB  R0                                ;:ZERO THE CONSTANTS INDEX
7376 037112 005000                        MOV      #DBLK,R3                        ;:SETUP THE OUTPUT POINTER
7377 037114 012703 037272                MOVB    #' ,(R3)                        ;:SET THE FIRST CHARACTER TO A BLANK
7378 037120 112723 000040                2$:      CLR  R2                                ;:CLEAR THE BCD NUMBER
7379 037124 005002                        MOV      $DTBL(R0),R1                    ;:GET THE CONSTANT
7380 037126 016001 037262                3$:      SUB  R1,R5                        ;:FORM THIS BCD DIGIT
7381 037132 160105                        BLT      4$                                ;:BR IF DONE
7382 037134 002402                        INC      R2                                ;:INCREASE THE BCD DIGIT BY 1
7383 037136 005202                        BR       3$
7384 037140 000774                        4$:      ADD  R1,R5
7385 037142 060105                        TST      R2                                ;:ADD BACK THE CONSTANT
7386 037144 005702                        BNE      5$                                ;:CHECK IF BCD DIGIT=0
7387 037146 001002                        TSTB    (SP)                        ;:FALL THROUGH IF 0
7388 037150 105716                        BMI      7$                                ;:STILL DOING LEADING 0'S?
7389 037152 100407                        5$:      ASLB  (SP)                        ;:BR IF YES
7390 037154 106316                        BCC      6$                                ;:MSD?
7391 037156 103003                        MOVB    1(SP),-1(R3)                    ;:BR IF NO
7392 037160 116663 000001 177777        6$:      BIS  #'0,R2                        ;:YES--SET THE SIGN
7393 037166 052702 000060                7$:      BIS  #' ,R2                        ;:MAKE THE BCD DIGIT ASCII
7394 037172 052702 000040                MOVB    R2,(R3)                        ;:MAKE IT A SPACE IF NOT ALREADY A DIGIT
7395 037176 110223                        TST      (R0)                        ;:PUT THIS CHARACTER IN THE OUTPUT BUFFER
7396 037200 005720                        CMP      R0,#10                        ;:JUST INCREMENTING
7397 037202 020027 000010                BLT      2$                                ;:CHECK THE TABLE INDEX
7398 037206 002746                        BGT      8$                                ;:GO DO THE NEXT DIGIT
7399 037210 003002                        MOV      R5,R2                                ;:GO TO EXIT
7400 037212 010502                        BR       6$                                ;:GET THE LSD
7401 037214 000764                        TSTB    (SP)                        ;:GO CHANGE TO ASCII
7402 037216 105726                        8$:      BPL  9$                                ;:WAS THE LSD THE FIRST NON-ZERO?
7403 037220 100003                        MOVB    -1(SP),-2(R3)                    ;:BR IF NO
7404 037222 116663 177777 177776        9$:      CLRB  (R3)                        ;:YES--SET THE SIGN FOR TYPING
7405 037230 105013                        MOV      (SP),R5                        ;:SET THE TERMINATOR
7406 037232 012605                        MOV      (SP),R3                        ;:POP STACK INTO R5
7407 037234 012603                        MOV      (SP),R2                        ;:POP STACK INTO R3
7408 037236 012602                        MOV      (SP),R1                        ;:POP STACK INTO R2
7409 037240 012601                        MOV      (SP),R0                        ;:POP STACK INTO R1
7410 037242 012600                        TYPE    ,DBLK                        ;:POP STACK INTO R0
7411 037244 104401 037272                MOV      2(SP),4(SP)                    ;:NOW TYPE THE NUMBER
7412 037250 016666 000002 000004        MOV      (SP),R0                        ;:ADJUST THE STACK
7413 037256 012616                        RTI                                     ;:RETURN TO USER
7414 037260 000002                        $DTBL: 10000.
7415 037262 023420                        1000.
7416 037264 001750                        100.
7417 037266 000144                        10.
7418 037270 000012                        10.
```



```

7419 037272 000004 $DBLK: .BLKW 4
7420 .SBTTL BINARY TO OCTAL (ASCII) AND TYPE
7421
7422
7423
7424
7425
7426
7427
7428
7429
7430
7431
7432
7433
7434
7435
7436
7437
7438
7439
7440
7441
7442
7443
7444
7445 037302 017646 000000
7446 037306 116667 000001 000211
7447 037314 112667 000207
7448 037320 062716 000002
7449 037324 000406
7450 037326 112767 000001 000171
7451 037334 112767 000006 000165
7452 037342 112767 000005 000154
7453 037350 010346
7454 037352 010446
7455 037354 010546
7456 037356 116704 000145
7457 037362 005404
7458 037364 062704 000006
7459 037370 110467 000132
7460 037374 116704 000125
7461 037400 016605 000012
7462 037404 005003
7463 037406 006105
7464 037410 000404
7465 037412 006105
7466 037414 006105
7467 037416 006105
7468 037420 010503
7469 037422 006103
7470 037424 105367 000076
7471 037430 100016
7472 037432 042703 177770
7473 037436 001002
7474 037440 005704

;*****
;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    $OFILL, R4       ;;GET THE ZERO FILL SWITCH
        MOV     12 (SP), R5      ;;PICKUP THE INPUT NUMBER
        CLR     R3              ;;CLEAR THE OUTPUT WORD
1$:     ROL     R5               ;;ROTATE MSB INTO "C"
        BR      3$             ;;GO DO MSB
2$:     ROL     R5               ;;FORM THIS DIGIT
        ROL     R5
        ROL     R5
        MOV     R5, R3
3$:     ROL     R3               ;;GET LSR 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?

```

```
7475 037442 001403      BEQ      5$      ;;BR IF YES
7476 037444 005204      4$: INC      R4      ;;DON'T SUPPRESS ANYMORE O'S
7477 037446 052703 000060      BIS      #'0,R3  ;;MAKE THIS DIGIT ASCII
7478 037452 052703 000040      5$: BIS      #' ,R3  ;;MAKE ASCII IF NOT ALREADY
7479 037456 110367 000040      MOV      R3,8$  ;;SAVE FOR TYPING
7480 037462 104401 037522      TYPE     ,8$  ;;GO TYPE THIS DIGIT
7481 037466 105367 000032      7$: DECB     $OCNT  ;;COUNT BY 1
7482 037472 003347      BGT      2$      ;;BR IF MORE TO DO
7483 037474 002402      BLT      6$      ;;BR IF DONE
7484 037476 005204      INC      R4      ;;INSURE LAST DIGIT ISN'T A BLANK
7485 037500 000744      BR       2$      ;;GO DO THE LAST DIGIT
7486 037502 012605      6$: MOV      (SP)+,R5  ;;RESTORE R5
7487 037504 012604      MOV      (SP)+,R4  ;;RESTORE R4
7488 037506 012603      MOV      (SP)+,R3  ;;RESTORE R3
7489 037510 016666 000002 000004      MOV      2(SP),4(SP)  ;;SET THE STACK FOR RETURNING
7490 037516 012616      MOV      (SP)+,(SP)
7491 037520 000002      RTI      ;;RETURN
7492 037522 000      8$: .BYTE     0      ;;STORAGE FOR ASCII DIGIT
7493 037523 000      .BYTE     0      ;;TERMINATOR FOR TYPE ROUTINE
7494 037524 000      $OCNT: .BYTE     0      ;;OCTAL DIGIT COUNTER
7495 037525 000      $OFILL: .BYTE     0      ;;ZERO FILL SWITCH
7496 037526 000000      $OMODE: .WORD     0      ;;NUMBER OF DIGITS TO TYPE
7497      .SBTTL TRAP DECODER
7498
7499      ;;*****
7500      ;*THIS ROUTINE WILL PICKUP THE LOWER BYTE OF THE "TRAP" INSTRUCTION
7501      ;*AND USE IT TO INDEX THROUGH THE TRAP TABLE FOR THE STARTING ADDRESS
7502      ;*OF THE DESIRED ROUTINE. THEN USING THE ADDRESS OBTAINED IT WILL
7503      ;*GO TO THAT ROUTINE.
7504
7505 037530 010046      $TRAP: MOV      RO,-(SP)  ;;SAVE RO
7506 037532 016600 000002      MOV      2(SP),RO  ;;GET TRAP ADDRESS
7507 037536 005740      TST      -(RO)  ;;BACKUP BY 2
7508 037540 111000      MOV      (RO),RO  ;;GET RIGHT BYTE OF TRAP
7509 037542 006300      ASL      RO  ;;POSITION FOR INDEXING
7510 037544 016000 037564      MOV      $TRPAD(RO),RO  ;;INDEX TO TABLE
7511 037550 000200      RTS      RO  ;;GO TO ROUTINE
7512
7513
7514      ;;THIS IS USE TO HANDLE THE "GETPRI" MACRO
7515
7516 037552 011646      $TRAP2: MOV      (SP),-(SP)  ;;MOVE THE PC DOWN
7517 037554 016666 000004 000002      MOV      4(SP),2(SP)  ;;MOVE THE PSW DOWN
7518 037562 000002      RTI      ;;RESTORE THE PSW
7519
7520      .MACRO SETTRAP A,B,MSG
7521      ;;SET A,B,\<TRAP>,$TRP,<MSG>
7522
7523      .NLIST
7524      $TRP=$TRP+1
7525      .LIST
7526      .ENDM SETTRAP
7527      .MACRO $$SET A,B,C,D,COMNT
7528      .IF EQ $TRP-1
7529      .SBTTL TRAP TABLE
7530
7531      ;*THIS TABLE CONTAINS THE STARTING ADDRESSES OF THE ROUTINES CALLED
```



```
7531      ;*BY THE "TRAP" INSTRUCTION.
7532
7533      :      ROUTINE
7534      :      -----
7535      $TRPAD: .WORD      $TRAP2
7536      .ENDC
7537      .IIF NDF GNS,.NLIST
7538      A=      C
7539      .IIF NDF GNS,.LIST
7540      B      ;:CALL=A      TRAP+D(C)      COMNT
7541      .ENDM      $$SET
7542      .MACRO      TRMTRP
7543      $TERM=.-$TRPAD
7544      .ENDM      TRMTRP
7545      .SBTTL      TRAP TABLE
7546
7547      ;*THIS TABLE CONTAINS THE STARTING ADDRESSES OF THE ROUTINES CALLED
7548      ;*BY THE "TRAP" INSTRUCTION.
7549
7550      :      ROUTINE
7551      :      -----
7552      $TRPAD: .WORD      $TRAP2
7553      $TYPE      ;:CALL=TYPE      TRAP+1(104401)      TTY TYPEOUT ROUTINE
7554      $TYPOC      ;:CALL=TYPOC      TRAP+2(104402)      TYPE OCTAL NUMBER (WITH LEADING ZEROS)
7555      $TYPOS      ;:CALL=TYPOS      TRAP+3(104403)      TYPE OCTAL NUMBER (NO LEADING ZEROS)
7556      $TYPON      ;:CALL=TYPON      TRAP+4(104404)      TYPE OCTAL NUMBER (AS PER LAST CALL)
7557      $TYPDS      ;:CALL=TYPDS      TRAP+5(104405)      TYPE DECIMAL NUMBER (WITH SIGN)
7558
7559
7560      .SBTTL      APT COMMUNICATIONS ROUTINE
7561
7562      ;:*****
7563      037600      112767      000001      000236      $ATY1:      MOVB      #1,$FFLG      ;:TO REPORT FATAL ERROR
7564      037606      112767      000001      000226      $ATY3:      MOVB      #1,$MFLG      ;:TO TYPE A MESSAGE
7565      037614      000403
7566      037616      112767      000001      000220      $ATY4:      MOVB      #1,$FFLG      ;:TO ONLY REPORT FATAL ERROR
7567      037624
7568      037624      010046
7569      037626      010146
7570      037630      105767      000206
7571      037634      001450
7572      037636      122767      000001      141154
7573      037644      001031
7574      037646      132767      000100      141145
7575      037654      001425
7576      037656      017600      000004
7577      037662      062766      000002      000004
7578      037670      005767      141104      1$:      TST      $MSGTYPE      ;:SEE IF DONE W/ LAST XMISSION?
7579      037674      001375
7580      037676      010067      141112
7581      037702      105720
7582      037704      001376      2$:      TSTB      (R0),      ;:FIND END OF MESSAGE
7583      037706      166700      141102
7584      037712      006200
7585      037714      010067      141076
7586      037720      012767      000004      141052      MOV      #4,$MSGTYPE      ;:TELL APT TO TAKE MSG.
```

```

7587 037726 000413 BR 5$
7588 037730 017667 000004 000016 3$: MOV @4(SP),4$ ;;PUT MSG ADDR IN JSR LINKAGE
7589 037736 062766 000002 000004 ADD @2,4(SP) ;;BUMP RETURN ADDRESS
7590 037744 016746 140026 MOV 177776,-(SP) ;;PUSH 177776 ON STACK
7591 037750 004767 176526 JSR PC,$TYPE ;;CALL TYPE MACRO
7592 037754 000000 4$: .WORD 0
7593 037756 5$:
7594 037756 105767 000062 10$: TSTB $FFLG ;;SHOULD REPORT FATAL ERROR?
7595 037762 001416 BEQ 12$ ;;IF NOT: BR
7596 037764 005767 141030 TST $ENV ;;RUNNING UNDER APT?
7597 037770 001413 BEQ 12$ ;;IF NOT: BR
7598 037772 005767 141002 11$: TST $MSGTYPE ;;FINISHED LAST MESSAGE?
7599 037776 001375 BNE 11$ ;;IF NOT: WAIT
7600 040000 017667 000004 140774 MOV @4(SP),$FATAL ;;GET ERROR #
7601 040006 062766 000002 000004 ADD @2,4(SP) ;;BUMP RETURN ADDR.
7602 040014 005267 140760 INC $MSGTYPE ;;TELL APT TO TAKE ERROR
7603 040020 105067 000020 12$: CLRB $FFLG ;;CLEAR FATAL FLAG
7604 040024 105067 000013 CLRB $LFLG ;;CLEAR LOG FLAG
7605 040030 105067 000006 CLRB $MFLG ;;CLEAR MESSAGE FLAG
7606 040034 012601 MOV (SP)+,R1 ;;POP STACK INTO R1
7607 040036 012600 MOV (SP)+,R0 ;;POP STACK INTO R0
7608 040040 000207 RTS PC ;;RETURN
7609 040042 000 $MFLG: .BYTE 0 ;;MESSG. FLAG
7610 040043 000 $LFLG: .BYTE 0 ;;LOG FLAG
7611 040044 000 $FFLG: .BYTE 0 ;;FATAL FLAG
7612 040046 .EVEN
7613 000200 APTSIZE=200
7614 000001 APTENV=001
7615 000100 APTSPOOL=100
7616 000040 APTCSUP=040
7617 *****
7618 ;THIS ROUTINE WILL INCREMENT THE ERROR COUNT AND THEN PASS THE UNIQUE
7619 ;ERROR NUMBER TO THE APT ERROR ROUTINE TO BE REPORTED TO THE APT SYSTEM.
7620
7621 040046 005267 141000 $ERROR: INC $ERFLG ;;INCREMENT ERROR FLAG
7622 040052 001775 BEQ $ERROR ;;DON'T LET IT GO TO ZERO
7623 040054 005267 140764 INC ERRCNT ;;INCREMENT THE ERROR COUNT
7624 040060 021627 001002 CMP (SP), #1002 ;;IS ERROR FROM VECTOR AREA
7625 040064 101010 BHI 1$ ;;IF YES THEN
7626 040066 012767 007777 000106 MOV @7777,3$ ;;REPORT AN UNEXPECTED TRAP
7627 040074 012637 001062 MOV (SP)+,@$SAVSP1 ;;SAVE UNEXPECTED TRAP DATA
7628 040100 012637 001064 MOV (SP)+,@$SAVSP2 ;;AND RESTORE SP
7629 040104 000430 BR 2$ ;;ELSE
7630 040106 017667 000000 000066 1$: MOV @4(SP),3$ ;;REPORT UNIQUE ERROR NUMBER TO APT
7631 040114 011667 000072 MOV (SP),101$ ;;SAVE ERROR PC
7632 040120 062716 000002 ADD @2,(SP) ;;GET OVER UNIQUE ERROR NUMBER FOR RETURN
7633 040124 017637 000000 040134 100$: MOV @4(SP),@102$
7634 040132 104401 TYPE ;;TYPE ERROR MESSAGE
7635 040134 000000 102$: .WORD 0
7636 040136 062716 000002 ADD @2,(SP) ;;GET OVER ERROR MESSAGE
7637 040142 104401 002046 TYPE ,ERR1
7638 040146 016746 000030 MOV 3$,-(SP) ;;PUSH UNIQUE ERROR NUMBER ON THE STACK
7639 040152 104402 TYPEOC ;;TYPE OCTAL ERROR NUMBER
7640 040154 104401 002062 TYPE ,ERR2
7641 040160 016746 000026 MOV 101$,-(SP) ;;PUSH ERROR PC ON THE STACK
7642 040164 104402 TYPEOC ;;TYPE THE ERROR PC

```


7643	040166	122767	000001	140624	2:	CMPB	@APTENV,\$ENV
7644	040174	001004				BNE	5:
7645	040176	004767	177414			JSR	PC,\$ATY4
7646	040202	000000			3:	.WORD	0
7647	040204	000777			4:	BR	4:
7648	040206	000000			5:	HALT	
7649	040210	000002				RTI	
7650	040212	000000			101:	.WORD	0
7651	040214				\$PATCH:		
7652	040214	000010				.BLKW	10
7653		000001			.END		

```

;CHECK TO MAKE SURE WE'RE IN APT MODE
;IF YES THEN
;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	547					
ACDW1 = 000000	547					
ACDW2 = 000000	547					
ACPUOP = 000000	547	562				
ADD1 = 003556	1083	1091	1099	1107	1116	11320
ADDW0 = 000000	547					
ADDW1 = 000000	547					
ADDW10 = 000000	547					
ADDW11 = 000000	547					
ADDW12 = 000000	547					
ADDW13 = 000000	547					
ADDW14 = 000000	547					
ADDW15 = 000000	547					
ADDW2 = 000000	547					
ADDW3 = 000000	547					
ADDW4 = 000000	547					
ADDW5 = 000000	547					
ADDW6 = 000000	547					
ADDW7 = 000000	547					
ADDW8 = 000000	547					
ADDW9 = 000000	547					
ADEVCT = 000000	547	553				
ADEVN = 000000	547					
AENV = 000000	547	558				
AENVN = 000000	547	559				
AFATAL = 000000	547	550				
ALLCTR = 001056	5890					
AMADR1 = 000000	547					
AMADR2 = 000000	547					
AMADR3 = 000000	547					
AMADR4 = 000000	547					
AMAMS1 = 000000	547					
AMAMS2 = 000000	547					
AMAMS3 = 000000	547					
AMAMS4 = 000000	547					
AMSGAD = 000000	547	555				
AMSGLG = 000000	547	556				
AMSGTY = 000000	547	549				
AMTYP1 = 000000	547					
AMTYP2 = 000000	547					
AMTYP3 = 000000	547					
AMTYP4 = 000000	547					
APASS = 000000	547	552				
APRIOR = 000000	547					
APTCU = 000040	7276	76160				
APTENV = 000001	7269	7572	76140	7643		
APTSIZ = 000200	867	76130				
APTSPO = 000100	7271	7574	76150			
ASWREG = 000000	547	560				
ATESTN = 000000	547	551				
AUNIT = 000000	547	554				
AUSWR = 002000	4660	547	561			
AVECT1 = 000000	547					
AVECT2 = 000000	547					
BEVENT = 177546	4580					
BFA = 003610	1077	1085	1093	1101	1110	1119 11380

[illegible]

1497	1503											
1569	1574											
1498	1503											
1568	1574											
5809												
5840												
4039												
4047	4056	4065	4073	4095								
581	817	822	823	2919	2964	3004	3040	3070	3106	3144	3175	3194
3211	3229	3263	3282	3313	3332	3357	3371	3403	3415	3431	3444	3476
3512	3550	3575	3597	3619	3651	3670	3697	3711	3742	3780	3805	3843
3868	3906	3938	3968	4006	4030	4122	4361	4560	4779	4987	5206	5217
5506	5517	5622	5762	5824	6191	6369	6498	6797	7026			
455												
1	166	509	603	756	877	7233						
194	752	756	761	763	766	7233	7241	7332	7344			
195	7286	7344										
1	874											
820	2918	2963	3003	3039	3069	3105	3143	3174	3193	3210	3228	3262
3281	3312	3331	3356	3370	3402	3414	3430	3443	3475	3511	3549	3574
3596	3618	3650	3669	3696	3710	3741	3779	3804	3842	3867	3905	3937
3967	4005	4029	4121	4559	4986	5505	5516	5621	5823	6190	6497	6796
815	4360	4778	5205	5216	5761	6368	7025					
818	823	826										
588												
201	585	848										
585	848	856										
495	856											
4381												
4133												
200	584	847										
4389	4401	4413	4424	4435	4446	4457	4468	4480	4506			
4141	4151	4160	4168	4177	4185	4193	4201	4209	4217	4225	4233	4241
4250	4258	4267	4276	4285	4307							
1275	1291	1296										
1278	1279											
1280	1304											
1288	1325	1331										
1289	1292											
1293	1295											
1294	1297											
289	836	837										
794												
583	7623											
793												
752												
465	785	788	941	945	956	959	965	968	974	977	983	986
993	996	1003	1006	1079	1082	1087	1090	1095	1098	1103	1106	1112
1115	1121	1124	1175	1178	1208	1211	1228	1231	1236	1239	1246	1249
125												

1943#	1947	1950#	1956	1960#	1985	1988#	1993	1996#	2001	2004#	2008	2011#
2017	2021#	2048	2051#	2055	2058#	2062	2065#	2069	2072#	2076	2079#	2085
2089#	2115	2118#	2123	2126#	2130	2133#	2139	2143#	2169	2172#	2176	2179#
2185	2189#	2216	2219#	2223	2226#	2229	2232#	2236	2239#	2245	2249#	2273
2276#	2281	2284#	2289	2292#	2296	2299#	2316	2319#	2327	2330#	2338	2341#
2351	2354#	2362	2365#	2372	2375#	2384	2387#	2437	2440#	2445	2448#	2454
2457#	2463	2466#	2472	2475#	2481	2484#	2493	2496#	2545	2548#	2553	2556#
2562	2565#	2571	2574#	2580	2583#	2589	2592#	2600	2603#	2650	2653#	2657
2660#	2664	2667#	2674	2678#	2705	2708#	2716	2719#	2727	2730#	2738	2741#
2792	2795#	2803	2806#	2814	2817#	2825	2828#	2871	2874#	2878	2881#	2884
2887#	2906	2909#	2912	2915#	2922	2925#	2945	2948#	2951	2954#	2957	2960#
2966	2969#	2992	2995#	2998	3001#	3006	3009#	3028	3031#	3034	3037#	3042
3045#	3063	3066#	3072	3075#	3094	3097#	3100	3103#	3108	3111#	3131	3134#
3139	3142#	3146	3149#	3168	3171#	3177	3180#	3189	3192#	3196	3199#	3205
3208#	3213	3216#	3224	3227#	3231	3234#	3257	3260#	3265	3268#	3276	3279#
3284	3287#	3308	3311#	3315	3318#	3326	3329#	3334	3337#	3359	3362#	3373
3376#	3397	3400#	3405	3408#	3417	3420#	3433	3436#	3446	3449#	3470	3473#
3478	3481#	3489	3492#	3498	3501#	3505	3508#	3514	3517#	3527	3530#	3536
3539#	3543	3546#	3552	3555#	3569	3572#	3577	3580#	3591	3594#	3599	3602#
3613	3616#	3621	3624#	3645	3648#	3653	3656#	3664	3667#	3672	3675#	3699
3702#	3713	3716#	3736	3739#	3744	3747#	3757	3760#	3766	3769#	3773	3776#
3782	3785#	3799	3802#	3807	3810#	3820	3823#	3829	3832#	3836	3839#	3845
3848#	3862	3865#	3870	3873#	3883	3886#	3894	3897#	3900	3903#	3908	3911#
3932	3935#	3940	3943#	3953	3956#	3962	3965#	3970	3973#	3984	3987#	4000
4003#	4008	4011#	4024	4027#	4032	4035#	4112	4115#	4124	4127#	4323	4326#
4332	4335#	4341	4344#	4354	4357#	4363	4366#	4374	4377#	4522	4525#	4531
4534#	4540	4543#	4553	4556#	4562	4565#	4573	4576#	4741	4744#	4750	4753#
4759	4762#	4772	4775#	4781	4784#	4793	4796#	4949	4952#	4958	4961#	4967
4970#	4980	4983#	4989	4992#	5001	5004#	5167	5170#	5176	5179#	5185	5188#
5199	5202#	5208	5211#	5219	5222#	5230	5233#	5467	5470#	5476	5479#	5485
5488#	5499	5502#	5508	5511#	5519	5522#	5530	5533#	5615	5618#	5624	5627#
5630	5634#	5726	5729#	5735	5738#	5744	5747#	5755	5758#	5764	5767#	5776
5779#	5796	5799#	5802	5805#	5826	5829#	5833	5836#	5853	5856#	5859	5862#
5880	5883#	5886	5889#	5892	5895#	5900	5903#	5921	5924#	5927	5930#	5933
5936#	5941	5944#	5964	5967#	5970	5973#	5976	5979#	5984	5987#	6005	6008#
6011	6014#	6017	6020#	6025	6028#	6048	6051#	6054	6057#	6060	6063#	6068
6071#	6089	6092#	6095	6098#	6101	6104#	6109	6112#	6132	6135#	6138	6141#
6144	6147#	6152	6155#	6176	6179#	6183	6186#	6193	6196#	6362	6365#	6371
6374#	6380	6383#	6491	6494#	6500	6503#	6509	6512#	6758	6761#	6767	6770#
6776	6779#	6790	6793#	6799	6802#	6811	6814#	6990	6993#	6999	7002#	7008
7011#	7019	7022#	7028	7031#	7040	7043#	7169	7172#	7176	7179#	7185	7188#

ERRTN = 000000
ERRVEC= 000004
ERR1 002046
ERR2 002062
EXPDAT 001034
FACU = ***** U
FIN1 003742
FIN10 005450
FIN11 005560
FIN13 006004
FIN14 006160
FIN15 006400
FIN16 006572
FIN17 007006
FIN2 004044

463#
282# 845
761# 7637
763# 7640
579#
1040
1169 1180#
1565 1596#
1630 1636#
1706 1715#
1759 1768#
1827 1836#
1887 1896#
1954 1963#
1199 1213#

846# 857#

2015	20240
2083	20920
2137	21460
2183	21910
2243	22520
2379	24150
2487	25230
2594	26300
2672	26810
2732	27680
2819	28550
1302	13460
1378	13860
1437	14540
1494	15250
5820	40460

FPPERR 002013

FPPTST = ***** U
FPTS1 = ***** U
FPTS10 = ***** U

582*	4046*	4055*	4064*	4116	4140*	4150*	4159*	4167*	4176*	4184*	4192*	4200*
4208*	4216*	4224*	4232*	4240*	4249*	4257*	4266*	4275*	4284*	4321	4330	4367
4388*	4400*	4412*	4423*	4434*	4445*	4456*	4467*	4479*	4520	4529	4566	4587*
4595*	4603*	4611*	4619*	4627*	4635*	4643*	4651*	4659*	4667*	4675*	4684*	4693*
4702*	4739	4748	4785	4806*	4817*	4828*	4839*	4850*	4861*	4872*	4883*	4895*
4906*	4947	4956	4993	5014*	5023*	5032*	5041*	5050*	5060*	5069*	5078*	5087*
5096*	5105*	5114*	5123*	5165	5174	5223	5247*	5260*	5273*	5286*	5299*	5312
5325*	5338*	5351*	5364*	5378*	5391*	5405*	5419*	5465	5474	5523	5646*	5654*
5662*	5670*	5678*	5686*	5724	5733	5768	6524*	6534*	6544*	6554*	6564*	6574*
6585*	6596*	6607*	6618*	6629*	6639*	6649*	6660*	6671*	6681*	6692*	6702*	6713*
6756	6765	6803	6826*	6834*	6842*	6850*	6858*	6866*	6874*	6882*	6890*	6898*
6906*	6914*	6922*	6930*	6938*	6946*	6988	6997	7032	7055*	7063*	7071*	7079*
756*	787	944	958	967	976	985	995	1005	1081	1089	1097	1105
1114	1123	1177	1210	1230	1238	1248	1256	1286	1308	1316	1329	1336
1343	1366	1373	1383	1406	1414	1423	1431	1451	1501	1509	1572	1580
1618	1625	1654	1660	1687	1694	1701	1711	1739	1747	1754	1764	1793
1800	1807	1815	1822	1832	1860	1867	1875	1882	1892	1921	1927	1934
1942	1949	1959	1987	1995	2003	2010	2020	2050	2057	2064	2071	2078
2088	2117	2125	2132	2142	2171	2178	2188	2218	2225	2231	2238	2248
2275	2283	2291	2298	2318	2329	2340	2353	2364	2374	2386	2439	2447
2456	2465	2474	2483	2495	2547	2555	2564	2573	2582	2591	2602	2652
2659	2666	2677	2707	2718	2729	2740	2794	2805	2816	2827	2873	2880
2886	2908	2914	2924	2947	2953	2959	2968	2994	3000	3008	3030	3036
3044	3065	3074	3096	3102	3110	3133	3141	3148	3170	3179	3191	3198
3207	3215	3226	3233	3259	3267	3278	3286	3310	3317	3328	3336	3361
3375	3399	3407	3419	3435	3448	3472	3480	3491	3500	3507	3516	3529
3538	3545	3554	3571	3579	3593	3601	3615	3623	3647	3655	3666	3674
3701	3715	3738	3746	3759	3768	3775	3784	3801	3809	3822	3831	3838
3847	3864	3872	3885	3896	3902	3910	3934	3942	3955	3964	3972	3986
4002	4010	4026	4034	4114	4126	4325	4334	4343	4356	4365	4376	4524
4533	4542	4555	4564	4575	4743	4752	4761	4774	4783	4795	4951	4960
4969	4982	4991	5003	5169	5178	5187	5201	5210	5221	5232	5469	5478
5487	5501	5510	5521	5532	5617	5626	5633	5728	5737	5746	5757	5766
5778	5798	5804	5828	5835	5855	5861	5882	5888	5894	5902	5923	

[illegible]

[illegible]

[illegible]

MMULF	022242	4580#												
MMUTST=	000001	1#	166	509	595	752	756	761	782	874	877	7233	7497	
MMVEC =	000250	298#												
MNGOP	015104	3454#												
MNNRM1	013700	3152#												
MNNRM2	014164	3238#												
MNNRM3	014342	3291#												
MNNRM4	014510	3341#												
MNRM	016066	3679#												
MODD =	***** U	5240												
MODF =	***** U	5007												
MODGAR	025426	5154	5236#	5454										
MSB	015720	3629#												
MSCD	034532	6819#												
MSCF	035622	7048#												
MSDF	027456	5639#												
MSER =	177744	453#												
MSFD	027136	5537#												
MSFDI	030144	5783#												
MSG1	036402	7207	7233#											
MSG2	036456	7217	7241#											
MSXP	035770	7089#												
MULD =	***** U	4799												
MULF =	***** U	4580												
MUVAD	016212	3719#												
MXDF1	014634	3380#												
MXVDEL=	000001	1#												
NGOP =	***** U	3454												
NNRM1 =	***** U	3152												
NNRM2 =	***** U	3238												
NNRM3 =	***** U	3291												
NNRM4 =	***** U	3341												
NRM =	***** U	3679												
PARRAM=	000001	1#												
PIRQ =	177772	199#												
PIRQVE=	000240	293#												
PRO =	000000	216#												
PR1 =	000040	217#												
PR2 =	000100	218#												
PR3 =	000140	219#												
PR4 =	000200	220#												
PR5 =	000240	221#												
PR6 =	000300	222#												
PR7 =	000340	223#												
PS =	177776	196#	197											
PSW =	177776	197#												
PWRVEC=	000024	288#												
QBUSEX=	000001	1#	602											
RBUF =	177562	460#												
RCSR =	177560	459#												
RECDAT	001036	580#												
RECDST	001126	600#	2897	2934	2962	2978	301P	3054	3034	3120	3159*	3160*	3172	3244
		3297	3351	3389	3466	3494	35	3565	3587	3609	3638	3693	3708	3732
		3762	3795	3825	3858	3888	3928	3958	3996	4020	4108	4350	4549	4768
		4976	5194	5494	5603	5715	581G	6138	6347	6476	6741	6972	7154	7174
RECFEC	001106	598#	2865	5772	6807	7036								

[illegible]

STACK = 001000	187#	834							
START 002200	492	498	831#						
STBOT = 001000	470#								
STKLMT = 177774	198#								
SUBT 003524	1064	1067	1069	1071	1073	1127#			
SWR 001046	584#	847*	849	855*	869*				
SWREG 000176	496#	855							
SW0 = 000001	251#								
SW00 = 000001	241#	251							
SW01 = 000002	240#	250							
SW02 = 000004	239#	249							
SW03 = 000010	238#	248							
SW04 = 000020	237#	247							
SW05 = 000040	236#	246							
SW06 = 000100	235#	245							
SW07 = 000200	234#	244							
SW08 = 000400	233#	243							
SW09 = 001000	232#	242							
SW1 = 000002	250#								
SW10 = 002000	231#								
SW11 = 004000	230#								
SW12 = 010000	229#								
SW13 = 020000	228#								
SW14 = 040000	227#								
SW15 = 100000	226#								
SW2 = 000004	249#								
SW3 = 000010	248#								
SW4 = 000020	247#								
SW5 = 000040	246#								
SW6 = 000100	245#								
SW7 = 000200	244#								
SW8 = 000400	243#								
SW9 = 001000	242#								
SXP = ***** U	7089								
SXPSUB 036154	7096	7103	7110	7117	7124	7131	7152#		
TAB1 001176	610#	2898	2910	2916	3020	3038	3086	3104	3299
TAB10 001326	652#	3390	3429						
TAB11 001336	656#	3388							
TAB11A 001346	660#	3401							
TAB12 001356	662#	3409							
TAB13 001366	664#								
TAB13B 001376	668#	3412							
TAB14 001406	670#	3422	3657						
TAB15 001416	674#	3423							
TAB16 001426	678#	3437	3948						
TAB17 001436	682#	3442							
TAB18 001446	685#	3330							
TAB2 001206	614#	2936	2961	3055	3246				
TAB21 001456	689#	3462	3483	3520	3637				
TAB22 001466	691#	3474							
TAB23 001476	693#	3484	3519						
TAB24 001506	695#	3510	3548						
TAB25 001516	697#	3558	3582						
TAB26 001526	699#	3557	3583						
TAB27 001536	701#	3573	3595						
TAB28 001546	703#	3605							

[illegible]

[illegible]

TST21	006572	19020		
TST22	007006	19690		
TST23	007206	20300		
TST24	007430	20970		
TST25	007616	21520		
TST26	007754	21970		
TST27	010156	22580		
TST3	003650	11560		
TST30	010320	23060		
TST31	010720	24210		
TST32	011264	25290		
TST33	011632	26360		
TST34	011774	26870		
TST35	012270	27740		
TST36	012566	28610		
TST37	012656	28950		
TST4	003742	11860		
TST40	012764	29320		
TST41	013124	29760		
TST42	013250	30160		
TST43	013360	30520		
TST44	013460	30820		
TST45	013566	31180		
TST46	013700	31560		
TST47	014164	32420		
TST5	004044	12190		
TST50	014342	32950		
TST51	014510	33450		
TST52	014634	33840		
TST53	015104	34580		
TST54	015720	36330		
TST55	016066	36830		
TST56	016212	37230		
TST57	017120	39190		
TST6	004164	12650		
TST60	017542	40430		
TST61	020056	41370		
TST62	021244	43850		
TST63	022242	45840		
TST64	023304	48030		
TST65	024336	50110		
TST66	025436	52440		
TST67	027136	55410		
TST7	004444	13520		
TST70	027456	56430		
TST71	030144	57870		
TST72	030214	58130		
TST73	030306	58440		
TST74	030356	58700		
TST75	030474	59110		
TST76	030612	59520		
TST77	030746	59950		
TS10D1	005420	1538	1550	15840
TS10D2	005430	1540	1547	15880
TS10D4	005440	1558	15920	
TS11D1	005550	1607	16320	

TS26D0	010640	2311	2319	2346	2354	2391#
TS26D1	010650	2322	2330	2357	2365	2395#
TS26D2	010660	2333	2341	2368	2399#	
TS26D3	010670	2320	2355	2403#		
TS26D4	010700	2331	2366	2407#		
TS26D5	010710	2342	2411#			
TS27D0	011204	2428	2499#			
TS27D1	011214	2503#				
TS27D2	011224	2507#				
TS27D3	011234	2440	2511#			
TS27D4	011244	2458	2515#			
TS27D5	011254	2476	2519#			
TS30D0	011552	2536	2606#			
TS30D1	011562	2610#				
TS30D2	011572	2614#				
TS30D3	011602	2548	2618#			
TS30D4	011612	2566	2622#			
TS30D5	011622	2584	2626#			
TS32D0	012210	2694	2744#			
TS32D1	012220	2748#				
TS32D3	012240	2700	2756#			
TS32D4	012250	2711	2760#			
TS32D5	012260	2722	2764#			
TS33D0	012506	2781	2831#			
TS33D1	012516	2835#				
TS33D2	012526	2839#				
TS33D3	012536	2787	2843#			
TS33D4	012546	2798	2847#			
TS33D5	012556	2809	2851#			
TS6DA	004730	1408	1425	1439#		
TS6DAT	004740	1400	1402	1443#		
TS7DA1	005164	1467	1479	1513#		
TS7DA2	005174	1469	1476	1517#		
TS7DA4	005204	1487	1521#			
TYPDS =	104405	7219	7557#			
TYPE =	104401	7207	7217	7220	7289	7411
TYPOC =	104402	7554#	7639	7642	7480	7553#
TYPON =	104404	7556#			7634	7637
TYPOS =	104403	7555#				7640
UDPAR0=	177660	342#				
UDPAR1=	177662	343#				
UDPAR2=	177664	344#				
UDPAR3=	177666	345#				
UDPAR4=	177670	346#				
UDPAR5=	177672	347#				
UDPAR6=	177674	348#				
UDPAR7=	177676	349#				
UDPDR0=	177620	320#				
UDPDR1=	177622	321#				
UDPDR2=	177624	322#				
UDPDR3=	177626	323#				
UDPDR4=	177630	324#				
UDPDR5=	177632	325#				
UDPDR6=	177634	326#				
UDPDR7=	177636	327#				
UIPAR0=	177640	331#				

UIPAR1=	177642	3320				
UIPAR2=	177644	3330				
UIPAR3=	177646	3340				
UIPAR4=	177650	3350				
UIPAR5=	177652	3360				
UIPAR6=	177654	3370				
UIPAR7=	177656	3380				
UIPDR0=	177600	3090				
UIPDR1=	177602	3100				
UIPDR2=	177604	3110				
UIPDR3=	177606	3120				
UIPDR4=	177610	3130				
UIPDR5=	177612	3140				
UIPDR6=	177614	3150				
UIPDR7=	177616	3160				
UVAD =	***** U	3719				
WLDTRP	002106	7830	3789	3852		
XBUF =	177566	4620				
XCSR =	177564	4610				
XDF1 =	***** U	3380				
\$APTHD	000204	518	5240			
\$ASTAT=	***** U	7594	7609			
\$ATYC	037624	7565	75670			
\$ATY1	037600	75630				
\$ATY3	037606	7274	75640			
\$ATY4	037616	75660	7645			
\$CHARC	037032	7291*	7301*	7308	7334*	73390
\$CKSWR=	***** U	7560				
\$CMTAG=	***** U	834	840	842		
\$CPUOP	001026	5620				
\$CRLF	002077	7660	7290	7351		
\$DBLK	037272	7377	7411	74190		
\$DEVCT	001010	5530				
\$DOAGN	036372	7213	7222	72280		
\$DTBL	037262	7380	74150			
\$ENDAD	036362	505	72240			
\$ENDCT	036330	841	72150			
\$ENULL	036376	7220	72310			
\$ENV	001020	5580	7269	7572	7596	7643
\$ENVM	001021	5590	867	7271	7276	7574
\$EOP	036274	72040				
\$EOPCT	036322	841*	72120	7216		
\$ERFLG	001052	5860	842*	7621*		
\$ERROR	040046	836	871	76210	7622	
\$ETABL	001020	5570				
\$ETEND	001030	530	5690			
\$FATAL	001002	5500	7600*			
\$FFLG	040044	7563*	7566*	7594	7603*	76110
\$FILLC	037050	7294	73480			
\$FILLS	037047	73470				
\$GET42	036352	72210				
\$GTSWR=	***** U	7559				
\$HD =	000003	175	176			
\$HIBTS	000204	5250				
\$LF	037053	73510				
\$LFLG	040043	7604*	76100			

\$MAIL	001000	526	530	548#	859	7269								
\$MBADR	000206	526#												
\$MFLG	040042	7564#	7570	7605#	7609#									
\$MSGAD	001014	555#	7580#	7583										
\$MSGLG	001016	556#	7585-											
\$MSGTY	001000	549#	7578	7586#	7598	7602#								
\$NULL	037046	7296	7346#											
\$NWTST =	000001	884#	886	1041#	1043	1153#	1183#	1216#	1262#	1349#	1389#	1457#	1528#	1599#
		1639#	1666#	1718#	1771#	1839#	1899#	1966#	2027#	2094#	2149#	2194#	2255#	2303#
		2418#	2526#	2633#	2684#	2771#	2858#	2892#	2929#	2973#	3013#	3049#	3079#	3115#
		3153#	3239#	3292#	3342#	3381#	3455#	3630#	3680#	3720#	3916#	4040#	4134#	4382#
		4581#	4800#	5008#	5241#	5538#	5640#	5784#	5810#	5841#	5867#	5908#	5949#	5992#
		6033#	6076#	6117#	6160#	6202#	6388#	6518#	6820#	7049#	7090#			
		7452#	7481#	7494#										
		7447#	7451#	7456	7459#	7470#	7496#							
		552#	840#	866#	7205	7209#	7210#	7218	7231					
		528#												
\$OCNT	037524	7651#												
\$OMODE	037526	7350#												
\$PASS	001006	7560												
\$PASTM	000212	7560												
\$PATCH	040214	7560												
\$QUES	037052	7560												
\$RDCHR =	*****	7560												
\$RDDEC =	*****	7560												
\$RDLIN =	*****	7560												
\$RDOCT =	*****	7560												
\$RTNAD	036374	7230#												
\$R2A =	*****	7560												
\$SAVRE =	*****	7560												
\$SETUP =	000126	471#	835	836	838	840	842	843	7209					
\$STUP =	177777	471#												
\$SVPC =	000204	503#	508											
\$SWR =	160000	175	176#	843	895	1050	1158	1188	1221	1267	1354	1394	1462	1533
		1604	1644	1671	1723	1776	1844	1904	1971	2032	2099	2154	2199	2260
		2308	2423	2531	2638	2689	2776	2863	2897	2934	2978	3018	3054	3084
		3120	3158	3244	3297	3347	3386	3460	3635	3685	3725	3921	4045	4139
		4387	4586	4805	5013	5246	5543	5645	5789	5815	5846	5872	5913	5954
		5997	6038	6081	6122	6165	6207	6393	6523	6825	7054	7095	7201	7209
		7223	7229	7231										
\$SWREG	001022	560#	869											
\$TESTN	001004	551#	874#	894#	1049#	1157#	1187#	1220#	1266#	1353#	1393#	1461#	1532#	1603#
		1643#	1670#	1722#	1775#	1843#	1903#	1970#	2031#	2098#	2153#	2198#	2259#	2307#
		2422#	2530#	2637#	2688#	2775#	2862#	2896#	2933#	2977#	3017#	3053#	3083#	3119#
		3157#	3243#	3296#	3346#	3385#	3459#	3634#	3684#	3724#	3920#	4044#	4138#	4386#
		4585#	4804#	5012#	5245#	5542#	5644#	5788#	5814#	5845#	5871#	5912#	5953#	5996#
		6037#	6080#	6121#	6164#	6206#	6392#	6522#	6824#	7053#	7094#			
\$TKB	037040	7315	7322	7343#										
\$TKS	037036	7313	7320	7342#										
\$TN =	000112	175#	884	895#	1041	1050#	1153	1158#	1183	1188#	1216	1221#	1262	1267#
		1349	1354#	1389	1394#	1457	1462#	1528	1533#	1599	1604#	163#	1644#	1666
		1671#	1718	1723#	1771	1776#	1839	1844#	1899	1904#	1966	1971#	2027	2032#
		2094	2099#	2149	2154#	2194	2199#	2255	2260#	2303	2308#	2418	2423#	2526
		2531#	2633	2638#	2684	2689#	2771	2776#	2858	2863#	2892	2897#	2929	2934#
		2973	2978#	3013	3018#	3049	3054#	3079	3084#	3115	3120#	3153	3158#	3239
		3244#	3292	3297#	3342	3347#	3381	3386#	3455	3460#	3630	3635#	3680	3685#
		3720	3725#	3916	3921#	4040	4045#	4134	4139#	4382	4387#	4581	4586#	4800
		4805#	5008	5013#	5241	5246#	5538	5543#	5640	5645#	5784	5789#	5810	5815#
		5841	5846#	5867	5872#	5908	5913#	5949	5954#	5992	5997#	6033	6038#	6076
		6081#	6117	6122#	6160	6165#	6202	6207#	6388	6393#	6518	6523#	6820	6825#

[illegible]

BGNMOD	160#	876													
BGNSUB	161#														
BGNTST	160#	893	1048	1156	1186	1219	1265	1352	1392	1460	1531	1602	1642	1669	1721
	1774	1842	1902	1969	2030	2097	2152	2197	2258	2306	2421	2529	2636	2687	2774
	2861	2895	2932	2976	3016	3052	3082	3118	3156	3242	3295	3345	3384	3458	3633
	3683	3723	3919	4043	4137	4385	4584	4803	5011	5244	5541	5643	5787	5813	5844
	5870	5911	5952	5995	6036	6079	6120	6163	6205	6391	6521	6823	7052	7093	
CKLOOP	160#														
COMMEN	294#														
DEFPRG	1#														
ENDCOM	294#														
ENDMOD	160#	7194													
ENDPAS	7195#	7217													
ENDSUB	161#														
ENDTST	160#	1038	1150	1167	1197	1258	1300	1376	1435	1492	1563	1627	1662	1704	1757
	1825	1885	1952	2013	2081	2135	2181	2241	2299	2376	2484	2592	2670	2730	2817
	2889	2926	2971	3011	3047	3077	3113	3151	3236	3289	3339	3378	3452	3626	3677
	3718	3913	4037	4131	4293	4491	4711	4918	5058	5433	5584	5694	5807	5838	5864
	5905	5946	5989	6030	6073	6114	6157	6199	6328	6457	6724	6954	7087	7138	
ERRDEF	159#														
ERRDF	160#	785	941	956	965	974	983	993	1003	1079	1087	1095	1103	1112	1121
	1175	1208	1228	1236	1246	1254	1284	1306	1314	1326	1333	1340	1364	1371	1380
	1404	1412	1421	1429	1448	1499	1506	1570	1577	1616	1623	1652	1658	1685	1692
	1699	1708	1737	1745	1752	1761	1791	1798	1805	1813	1820	1829	1858	1865	1873
	1880	1889	1919	1925	1932	1940	1947	1956	1985	1993	2001	2008	2017	2048	2055
	2062	2069	2076	2085	2115	2123	2130	2139	2169	2176	2185	2216	2223	2229	2236
	2245	2273	2281	2289	2296	2316	2327	2338	2351	2362	2372	2384	2437	2445	2454
	2463	2472	2481	2493	2545	2553	2562	2571	2580	2589	2600	2650	2657	2664	2674
	2705	2716	2727	2738	2792	2803	2814	2825	2871	2878	2884	2906	2912	2922	2945
	2951	2957	2966	2992	2998	3006	3028	3034	3042	3063	3072	3094	3100	3108	3131
	3139	3146	3168	3177	3189	3196	3205	3213	3224	3231	3257	3265	3276	3284	3308
	3315	3326	3334	3359	3373	3397	3405	3417	3433	3446	3470	3478	3489	3498	3505
	3514	3527	3536	3543	3552	3569	3577	3591	3599	3613	3621	3645	3653	3664	3672
	3699	3713	3736	3744	3757	3766	3773	3782	3799	3807	3820	3829	3836	3845	3862
	3870	3883	3894	3900	3908	3932	3940	3953	3962	3970	3984	4000	4008	4024	4032
	4112	4124	4323	4332	4341	4354	4363	4374	4522	4531	4540	4553	4562	4573	4741
	4750	4759	4772	4781	4793	4949	4958	4967	4980	4989	5001	5167	5176	5185	5199
	5208	5219	5230	5467	5476	5485	5499	5508	5519	5530	5615	5624	5630	5726	5735
	5744	5755	5764	5776	5796	5802	5826	5833	5853	5859	5880	5886	5892	5900	5921
	5927	5933	5941	5964	5970	5976	5984	6005	6011	6017	6025	6048	6054	6060	6068
	6089	6095	6101	6109	6132	6138	6144	6152	6176	6183	6193	6362	6371	6380	6491
	6500	6509	6758	6767	6776	6790	6799	6811	6990	6999	7008	7019	7028	7040	7169
	7176	7185													
ERROR	188#	785	942	956	965	974	983	993	1003	1079	1087	1095	1103	1112	1121
	1175	1208	1228	1236	1246	1254	1284	1306	1314	1327	1334	1341	1364	1371	1381
	1404	1412	1421	1429	1449	1499	1507	1570	1578	1616	1623	1652	1658	1685	1692
	1699	1709	1737	1745	1752	1762	1791	1798	1805	1813	1820	1830	1858	1865	1873
	1880	1890	1919	1925	1932	1940	1947	1957	1985	1993	2001	2008	2018	2048	2055
	2062	2069	2076	2086	2115	2123	2130	2140	2169	2176	2186	2216	2223	2229	2236
	2246	2273	2281	2289	2296	2316	2327	2338	2351	2362	2372	2384	2437	2445	2454
	2463	2472	2481	2493	2545	2553	2562	2571	2580	2589	2600	2650	2657	2664	2675
	2705	2716	2727	2738	2792	2803	2814	2825	2871	2878	2884	2906	2912	2922	2945
	2951	2957	2966	2992	2998	3006	3028	3034	3042	3063	3072	3094	3100	3108	3131
	3139	3146	3168	3177	3189	3196	3205	3213	3224	3231	3257	3265	3276	3284	3308
	3315	3326	3334	3359	3373	3397	3405	3417	3433	3446	3470	3478	3489	3498	3505
	3514	3527	3536	3543	3552	3569	3577	3591	3599	3613	3621	3645	3653	3664	3672

	3699	3713	3736	3744	3757	3766	3773	3782	3799	3807	3820	3829	3836	3845	3862
	3870	3883	3894	3900	3908	3932	3940	3953	3962	3970	3984	4000	4008	4024	4032
	4112	4124	4323	4332	4341	4354	4363	4374	4522	4531	4540	4553	4562	4573	4741
	4750	4759	4772	4781	4793	4949	4958	4967	4980	4989	5001	5167	5176	5185	5199
	5208	5219	5230	5467	5476	5485	5499	5508	5519	5530	5615	5624	5631	5726	5735
	5744	5755	5764	5776	5796	5802	5826	5833	5853	5859	5880	5886	5892	5900	5921
	5927	5933	5941	5964	5970	5976	5984	6005	6011	6017	6025	6048	6054	6060	6068
	6089	6095	6101	6109	6132	6138	6144	6152	6176	6183	6193	6362	6371	6380	6491
	6500	6509	6758	6767	6776	6790	6799	6811	6990	6999	7008	7019	7028	7040	7169
	7176	7185													
ESCAPE	294#														
FPP1MS	883#	886													
FPP2MS	1040#	1043													
FRONT	1#														
FRONT1	1#	156#													
GETPRI	294#														
GETSWR	294#														
IDMSG	7195#	7205													
MULT	294#														
NEWTST	159#	294#	884	1041	1152	1182	1215	1261	1348	1388	1456	1527	1598	1638	1665
	1717	1770	1838	1898	1965	2026	2093	2148	2193	2254	2302	2417	2525	2632	2683
	2770	2858	2892	2929	2973	3013	3049	3079	3115	3153	3239	3292	3342	3381	3455
	3630	3680	3720	3916	4040	4134	4382	4581	4800	5008	5241	5538	5640	5784	5810
	5841	5867	5908	5949	5992	6033	6076	6117	6160	6202	6388	6518	6820	7049	7090
POP	294#	7406	7606	7607											
PUSH	294#	7365	7567	7569	7590										
REPORT	294#														
SCOPE	189#														
SETPRI	294#														
SETTRA	7520#	7545	7554	7555	7556	7557									
SETUP	159#	294#	833												
SKIP	294#														
SLASH	294#														
SPACE	294#														
STARS	294#	501	511	513	520	546	884	886	892	1041	1043	1047	1153	1155	1183
	1185	1216	1218	1262	1264	1349	1351	1389	1391	1457	1459	1528	1530	1599	1601
	1639	1641	1666	1668	1718	1720	1771	1773	1839	1841	1899	1901	1966	1968	2027
	2029	2094	2096	2149	2151	2194	2196	2255	2257	2303	2305	2418	2420	2526	2528
	2633	2635	2684	2686	2771	2773	2858	2860	2892	2894	2929	2931	2973	2975	3013
	3015	3049	3051	3079	3081	3115	3117	3153	3155	3239	3241	3292	3294	3342	3344
	3381	3383	3455	3457	3630	3632	3680	3682	3720	3722	3916	3918	4040	4042	4134
	4136	4382	4384	4581	4583	4800	4802	5008	5010	5241	5243	5538	5540	5640	5642
	5784	5786	5810	5812	5841	5843	5867	5869	5908	5910	5949	5951	5992	5994	6033
	6035	6076	6078	6117	6119	6160	6162	6202	6204	6388	6390	6518	6520	6820	6822
	7049	7051	7090	7092	7198	7248	7355	7422	7499	7562	7617				
SWRSU	294#	843#													
TAIL	1#	7195													
TRMTRP	7542#														
TYPBIN	294#														
TYPDEC	294#	7218													
TYPNAM	294#														
TYPNUM	294#														
TYPOCS	294#														
TYPOCT	294#														
TYPTXT	294#														
\$\$ESCA	294#														

\$\$NEWT	294#	884	1041	1153	1183	1216	1262	1349	1389	1457	1528	1599	1639	1666	1718
	1771	1839	1899	1966	2027	2094	2149	2194	2255	2303	2418	2526	2633	2684	2771
	2858	2892	2929	2973	3013	3049	3079	3115	3153	3239	3292	3342	3381	3455	3630
	3680	3720	3916	4040	4134	4382	4581	4800	5008	5241	5538	5640	5784	5810	5841
	5867	5908	5949	5992	6033	6076	6117	6160	6202	6388	6518	6820	7049	7090	
\$\$SET	7526#	7545	7554	7555	7556	7557									
\$\$SETM	859#	866													
\$\$SKIP	294#														
.EQUAT	159#	184													
.HEADE	161#	166													
.KT11	159#	294													
.SETUP	161#	471													
.\$ACT1	161#	499													
.\$APT8	159#	544													
.\$APTH	161#	509													
.\$APTY	162#	7560													
.\$EOP	159#	7196													
.\$ERRO	162#														
.\$READ	162#														
.\$TRAP	161#	7497													
.\$TYPD	160#	7353													
.\$TYPE	160#	7246													
.\$TYPO	162#	7420													
.\$40CA	159#	472													

. ABS. 040234 000

ERRORS DETECTED: 0

CZKDLB/EN:ABS.CZKDLB.SEQ/CRF/DOC/SOL/NL:TOC=SYSMAC.SML/ML,CZKDLB.MAC/ML,KDJ11A.MAC

RUN-TIME: 249 141 10 SECONDS

RUN-TIME RATIO: 539/400=1.3

CORE USED: 52K (103 PAGES)

DOCUMENT PAGES: 165