

ad 1/10

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DBKE04A-D  
PRODUCT NAME: KE11F SYSTEM EXERCISER OVERLAY  
DATE CREATED: 3-OCT-72  
MAINTAINER: DIAGNOSTIC GROUP  
AUTHOR: KEN CHAPMAN

COPYRIGHT (C) 1972  
DIGITAL EQUIPMENT CORPORATION  
MAYNARD, MASSACHUSETTS 01754

CONTENTS  
-----

1,	ABSTRACT
2,	REQUIREMENTS
2,1	Equipment
2,2	Storage
2,3	Preliminary programs
3,	LOADING PROCEDURE
4,	STARTING PROCEDURE
5,	OPERATING PROCEDURE
6,	ERRORS
7,	RESTRICTIONS
8,	MISCELLANEOUS
9,	PROGRAM DESCRIPTION

## 1: Abstract

This program is an overlay for GTP (MAINDEC-11-DZQGA) or CTP (MAINDEC-11-DZQCA) which tests the KE11F (PDP-11 Floating Instruction Set (FADD, FSUB, FMUL, and FDIV)) using all registers with fixed number patterns. Overflow, underflow, and divide by zero are checked to insure that the error trap works.

## 2: Requirements

## 2:1 Equipment

PDP-11 (KD11A) standard computer with KE11F option and a minimum of 12K of memory

## 2:2 Storage

GTP or CTP = 0 to 37500  
KE11F Overlay = 40000 to 57500

## 2:3 Preliminary Programs

All processor, KE11F, and device diagnostics

## 3: Loading Procedure

Load GTP or CTP into memory with the ABS loader then load the overlay into memory with the ABS loader.

## 4: Starting Procedure

See GTP or CTP

## 5: Operating Procedure

See GTP or CTP

## 6: Errors

See GTP or CTP

## 7: Restrictions

None

8, Miscellaneous  
See GTP or CTP

9, Program Description

This program is an overlay for GTP, CTP, or a similar type program. The user should load GTP or CTP as usual then load the overlay on top of it. The linking of the two programs is done automatically. To start, use the same switch settings as GTP or CTP and the same starting procedure. The subroutine abstract and error reporting is the same as the controlling program (GTP or CTP). The overlay tests all KE11F instructions including error conditions (overflow, underflow, and divide by zero). It uses every register as the stack pointer at least once with each instruction.

MAINDEC=11-DBKE0-A  
DBKE0A,P11

KE11F (PDP-11 FIS) SYSTEM EXERCISER OVERLAY  
TABLE OF CONTENTS

MACY11,620 3-OCT-72 17129

1	EQUALITIES
49	VECTOR AREA, STACKS, ANSWER AREA, AND SETUP ROUTINE
78	FAOD TEST SECTION
1003	FSUB TEST SECTION
1634	FMUL TEST SECTION
2013	FDIV TEST SECTION
2539	TEST OF FIS USING REGISTER 7 (PC)
2736	TEST OF ALL FIS AT ONCE
2795	PUSH AND POP SUBROUTINES

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

104400  
104000  
000004  
177776  
177570  
177570  
000007  
000000  
000001  
000002  
000003  
000004  
000005  
000005  
000006  
000007  
100000  
040000  
020000  
010000  
004000  
002000  
001000  
000400  
000001  
000002  
000004  
000010  
000020  
000040  
000100  
000200  
000400  
001000  
002000  
004000  
010000  
020000  
040000  
100000

TITLE MAINDEC-11-DBKE0-A  
ENABL ABS

KE11F (PDP-11 FIS) SYSTEM EXERCISER OVERLAY

ICOPYRIGHT 1972, DIGITAL EQUIPMENT CORP., MAYNARD, MASS  
IPROGRAM BY KEN CHAPMAN

SCOPE= TRAP  
HLT= EMT  
TYPE= IOT  
PS= 177776  
SWR= 177570  
DISPLAY=SWR  
BELL= 7  
R0= X0  
R1= X1  
R2= X2  
R3= X3  
R4= X4  
R5= X5  
TTV= X5  
SP= X6  
PC= X7  
SW15= 100000  
SW14= 40000  
SW13= 20000  
SW12= 10000  
SW11= 4000  
SW10= 2000  
SW09= 1000  
SW08= 400  
BI0 = 000001  
BI1 = 000002  
BI2 = 000004  
BI3 = 000010  
BI4 = 000020  
BI5 = 000040  
BI6 = 000100  
BI7 = 000200  
BI8 = 000400  
BI9 = 001000  
BI10 = 002000  
BI11 = 004000  
BI12 = 010000  
BI13 = 020000  
BI14 = 040000  
BI15 = 100000

```
48          000020          20  
49 000020 040060          KE11F  
50  
51          040000          40000  
52 040000 000000        ICNTI 0  
53 040002 000500        STACK1 500  
54 040004 000244        FISVEC 244  
55 040006 000246        FISLVL 246  
56  
57 040010 000000        SPSHI 0  
58 040012 000000        SSP1 0  
59 040014 000000        ANS1 0  
60 040016 000000        ANS2 0  
61 040020 000000        ANS3 0  
62 040022 000000        ANS4 0  
63 040024 000000        ANS5 0  
64 040026 000000        ANS6 0  
65 040030 000000 000000 000000 0,0,0,0  
66 040036 000000  
67 040040 000000        STACK0 0  
68 040042 000000        STACK2 0  
69 040044 000000        STACK4 0  
70 040046 000000        STACK6 0  
71 040050 000000 000000 000000 0,0,0,0  
72 040056 000000  
73  
74 040060 010667 177716  KE11Fi MOV SP, STACK
```

ENTER FIS VIA IOT

ITERATION COUNT = LH TEST NO. = RH  
STACK POINTER (%6) INITIAL ADR  
FIS TRAP VECTOR ADDRESS

PROCESSOR STATUS WORD  
STACK POINTER  
FIRST ANSWER (SEE CODE)

NON-%6 STACK BUFFER

NON-%6 STACK NORMAL LIMIT

NON-%6 STACK BUFFER

SAVE ORIGINAL STACK POINTER

```

75
76
77
78 |*****|
79 |TEST 1|      FADD (KE11F FLOATING ADD INSTRUCTION)
80 |      000000,000000 + 000000,000000 = 000000,000000
81 |      PS = 004,      STACK POINTER = R0
82 |*****|
83 040064 012767 000001 177706 TS11  MOV    #1,    ICNT    |KEEP TRACK OF TEST NUMBER
84 040072 004567 013654          JSR    R5,    PUSHR   |PUSH 4 WORDS ONTO R0 STACK, SET PRIORITY
85 040076 000000 000000          ,WORD 000000,000000 |SECOND OPERAND ON TOP
86 040102 000000 000000          ,WORD 000000,000000 |FIRST OPERAND ON BOTTOM
87 040106 000000 000000          ,WORD 000          |PROCESSOR PRIORITY LEVEL
88 040110 054236 000340          ,WORD TRAPER,340    |FIS TRAP VECTOR
89 040114 012700 040040          MOV    #STACK0,R0   |SET UP STACK POINTER
90
91 040120 000240          NOP
92 040122 079000          FADD+  R0          |FLOATING ADD ON THE R0 STACK
93
94 040124 004707 013654          JSR    PC,    POPR    |POP THE ANSWER
95 040130 010007 177656          MOV    R0,    SSP    |SAVE "STACK POINTER"
96 040134 022707 000004 177646          CMP    #004,   SPSW   |CHECK PS (EXCEPT T BIT)
97 040142 001401          BEQ    ,+4         |BRANCH IF OK
98 040144 104000          HLT
99
100 040146 022707 040044 177636          CMP    #STACK4,SSP  |CHECK THE STACK POINTER (R0)
101 040154 001401          BEQ    ,+4         |BRANCH IF OK
102 040156 104000          HLT
103
104 040160 022707 000000 177626          CMP    #000000,ANS1 |CHECK FIRST HALF OF ANSWER
105 040166 001401          BEQ    ,+4         |BRANCH IF OK
106 040170 104000          HLT
107
108 040172 022707 000000 177616          CMP    #000000,ANS2 |CHECK SECOND HALF OF ANSWER
109 040200 001401          BEQ    ,+4         |BRANCH IF OK
110 040202 104000          HLT
111
112 040204 122707 000001 177566 EN011  CMPB   #1,    ICNT    |CHECK THE TEST NUMBER
113 040212 001401          BEQ    ,+4         |BRANCH IF OK
114 040214 104000          HLT
115
116 040216 104400          SCOPE
117
118
119 |*****|
120 |TEST 2|      FADD (KE11F FLOATING ADD INSTRUCTION)
121 |      040200,000000 + 040200,000000 = 040400,000000
122 |      PS = 040,      STACK POINTER = R1
123 |*****|
124
125 040220 012767 000002 177552 TS21  MOV    #2,    ICNT    |KEEP TRACK OF TEST NUMBER
126 040226 004567 013520          JSR    R5,    PUSHR   |PUSH 4 WORDS ONTO R1 STACK, SET PRIORITY
127 040232 040200 000000          ,WORD 040200,000000 |SECOND OPERAND ON TOP
128 040234 040200 000000          ,WORD 040200,000000 |FIRST OPERAND ON BOTTOM

```





183										
184	040436	022767	040044	177346		CMP	#STACK4,SP		ICHECK THE STACK POINTER (R2)	
185	040444	001401				BEG	,+4		IBRANCH IF OK	
186	040446	104000				HLT			ISTACK POINTER (R2) NOT EQUAL TO #STACK4	
187										
188	040450	022767	000000	177336		CMP	#000000,ANS1		ICHECK FIRST HALF OF ANSWER	
189	040456	001401				BEG	,+4		IBRANCH IF OK	
190	040460	104000				HLT			IANS1 NOT EQUAL TO 000000	
191										
192	040462	022767	000000	177326		CMP	#000000,ANS2		ICHECK SECOND HALF OF ANSWER	
193	040470	001401				BEG	,+4		IBRANCH IF OK	
194	040472	104000				HLT			IANS2 NOT EQUAL TO 000000	
195										
196	040474	122767	000003	177276	EN031	CMPB	#3, ICNT		ICHECK THE TEST NUMBER	
197	040502	001401				BEG	,+4		IBRANCH IF OK	
198	040504	104000				HLT			IWRONG TEST! PC MUST HAVE FOULDED UP!	
199										
200	040506	104400							SCOPE	
201										
202										
203										
204										
205										
206										
207										
208										
209	040510	012767	000004	177262	TS141	MOV	#4, ICNT		IKEEP TRACK OF TEST NUMBER	
210	040516	004567	013072			JSR	R5, PUSH5		IPUSH 4 WORDS ONTO STACK, SET PRIORITY	
211	040522	152525	052524			,WORD	152525,052524		ISECOND OPERAND ON TOP	
212	040526	052525	052525			,WORD	052525,052525		IFIRST OPERAND ON BOTTOM	
213	040532	000017				,WORD	017		IPROCESSOR PRIORITY LEVEL	
214	040534	054236	000340			,WORD	TRAPER,340		IFIS TRAP VECTOR	
215										
216	040540	000240				NOP				
217	040542	075006				FA0D+	SP		IFLOATING ADD ON THE STACK	
218										
219	040544	004767	013064			JSR	PC, POPS		IPOP THE ANSWER	
220	040550	026706	177226			CMP	STACK, SP		ICHECK THE STACK POINTER	
221	040554	001404				BEG	TS44		IBRANCH IF OK	
222	040556	016706	177220			MOV	STACK, SP		IRESTORE STACK POINTER	
223	040562	104000				HLT			ISTACK POINTER FOULDED UP	
224	040564	000417				BR	END4		ISRIP REST OF TEST	
225										
226	040566	022767	000000	177214	TS441	CMP	#000, SP5W		ICHECK PS (EXCEPT 7 BIT)	
227	040574	001401				BEG	,+4		IBRANCH IF OK	
228	040576	104000				HLT			IPS NOT EQUAL TO 000	
229										
230	040600	022767	044600	177206		CMP	#044600,ANS1		ICHECK FIRST HALF OF ANSWER	
231	040606	001401				BEG	,+4		IBRANCH IF OK	
232	040610	104000				HLT			IANS1 NOT EQUAL TO 044600	
233										
234	040612	022767	000000	177176		CMP	#000000,ANS2		ICHECK SECOND HALF OF ANSWER	
235	040620	001401				BEG	,+4		IBRANCH IF OK	
236	040622	104000				HLT			IANS2 NOT EQUAL TO 000000	

```

*****
I TEST 41      PADD (KE11F FLOATING ADD INSTRUCTION)
I      052525,052525 + 152525,052524 = 044600,000000
I      PS = 000,      STACK POINTER = SP
*****
  
```

```

237
238 040624 122767 000004 177146 EN041 CMPB #4, ICNT ICHECK THE TEST NUMBER
239 040632 001401 BEQ ,+4 IBRANCH IF OK
240 040634 104000 HLT IWRONG TEST! PC MUST HAVE FOULED UP.
241
242 040636 104400 SCOPE
243
244
245
246 *****
247 ITEST 51 FADD (KE11F FLOATING ADD INSTRUCTION)
248 I 125200,000000 + 029177,177777 = 117200,000000
249 I PS = 110, STACK POINTER = SP
250 *****
251 040640 012767 000005 177132 TS751 MOV #5, ICNT IKEEP TRACK OF TEST NUMBER
252 040646 004567 012722 JSR R5, PUSHS IPUSH 4 WORDS ONTO STACK, SET PRIORITY
253 040652 029177 177777 ,WORD 029177,177777 ISECOND OPERAND ON TOP
254 040656 125200 000000 ,WORD 125200,000000 IFIRST OPERAND ON BOTTOM
255 040662 000107 ,WORD 107 IPROCESSOR PRIORITY LEVEL
256 040664 054236 000340 ,WORD TRAPER,340 IFIS TRAP VECTOR
257
258 040670 000240 NOP
259 040672 075006 FADD+ SP IFLOATING ADD ON THE STACK
260
261 040674 004767 012734 JSR PC, POPS IPOP THE ANSWER
262 040700 026706 177076 CMP STACK, SP ICHECK THE STACK POINTER
263 040704 001401 BEQ TSA5 IBRANCH IF OK
264 040706 016706 177070 MOV STACK, SP IRESTORE STACK POINTER
265 040712 104000 HLT ISTACK POINTER FOULED UP
266 040714 000417 BR EN05 ISKIP REST OF TEST
267
268 040716 022767 000110 177064 TSA51 CMP #110, SPSW ICHECK PS (EXCEPT F BIT)
269 040724 001401 BEQ ,+4 IBRANCH IF OK
270 040726 104000 HLT IPS NOT EQUAL TO 110
271
272 040730 022767 117200 177056 CMP #117200,ANS1 ICHECK FIRST HALF OF ANSWER
273 040736 001401 BEQ ,+4 IBRANCH IF OK
274 040740 104000 HLT IANS1 NOT EQUAL TO 117200
275
276 040742 022767 000000 177046 CMP #000000,ANS2 ICHECK SECOND HALF OF ANSWER
277 040750 001401 BEQ ,+4 IBRANCH IF OK
278 040752 104000 HLT IANS2 NOT EQUAL TO 000000
279
280 040754 122767 000005 177016 EN051 CMPB #5, ICNT ICHECK THE TEST NUMBER
281 040762 001401 BEQ ,+4 IBRANCH IF OK
282 040764 104000 HLT IWRONG TEST! PC MUST HAVE FOULED UP.
283
284 040766 104400 SCOPE
285
286
287
288 *****
289 ITEST 61 FADD (KE11F FLOATING ADD INSTRUCTION)
290 I 135753,024642 + 100125,052525 = 135753,024642
291 I PS = 150, STACK POINTER = R5
    
```

```

291
292
293 040770 012767 000006 177002 TSF61 MOV #6, ICNT ;KEEP TRACK OF TEST NUMBER
294 040776 004567 012750 JSR R5, PUSHR ;PUSH 4 WORDS ONTO R5 STACK, SET PRIORITY
295 041002 100125 052525 ;WORD 100125,052525 ;SECOND OPERAND ON TOP
296 041006 135753 024642 ;WORD 135753,024642 ;FIRST OPERAND ON BOTTOM
297 041012 000147 ;WORD 147 ;PROCESSOR PRIORITY LEVEL
298 041014 054236 000340 ;WORD TRAPER,340 ;FIS TRAP VECTOR
299 041020 012705 040040 MOV #STACK0,R5 ;SET UP STACK POINTER
300
301 041024 000240 NOP
302 041026 075005 FADD+ R5 ;FLOATING ADD ON THE R5 STACK
303
304 041030 004767 012750 JSR PC, POPR ;POP THE ANSWER
305 041034 010567 176752 MOV R5, SSP ;SAVE "STACK POINTER"
306 041040 022767 000150 176742 CMP #150, SPSW ;CHECK PS (EXCEPT 1 BIT)
307 041046 001401 BEQ ,+4 ;BRANCH IF OK
308 041050 104000 HLT ;PS NOT EQUAL TO 150
309
310 041052 022767 040044 176732 CMP #STACK4,SSP ;CHECK THE STACK POINTER (R5)
311 041060 001401 BEQ ,+4 ;BRANCH IF OK
312 041062 104000 HLT ;STACK POINTER (R5) NOT EQUAL TO #STACK4
313
314 041064 022767 135753 176722 CMP #135753,ANS1 ;CHECK FIRST HALF OF ANSWER
315 041072 001401 BEQ ,+4 ;BRANCH IF OK
316 041074 104000 HLT ;ANS1 NOT EQUAL TO 135753
317
318 041076 022767 024642 176712 CMP #024642,ANS2 ;CHECK SECOND HALF OF ANSWER
319 041104 001401 BEQ ,+4 ;BRANCH IF OK
320 041106 104000 HLT ;ANS2 NOT EQUAL TO 024642
321
322 041110 122767 000006 176662 END61 CMPB #6, ICNT ;CHECK THE TEST NUMBER
323 041116 001401 BEQ ,+4 ;BRANCH IF OK
324 041120 104000 HLT ;WRONG TEST! PC MUST HAVE FOULED UP!
325
326 041122 104400 SCOPE
327
328
329
330 ;*****
331 ;TEST 71 FADD (KE11F FLOATING ADD INSTRUCTION)
332 ; 000052,125252 + 001357,024642 = 001357,024642
333 ; PS = 040, STACK POINTER = R1
334 ;*****
335 041124 012767 000007 176646 TSF71 MOV #7, ICNT ;KEEP TRACK OF TEST NUMBER
336 041132 004567 012614 JSR R5, PUSHR ;PUSH 4 WORDS ONTO R1 STACK, SET PRIORITY
337 041136 001357 024642 ;WORD 001357,024642 ;SECOND OPERAND ON TOP
338 041142 000052 125252 ;WORD 000052,125252 ;FIRST OPERAND ON BOTTOM
339 041146 000057 ;WORD 057 ;PROCESSOR PRIORITY LEVEL
340 041150 054236 000340 ;WORD TRAPER,340 ;FIS TRAP VECTOR
341 041154 012701 040040 MOV #STACK0,R1 ;SET UP STACK POINTER
342
343 041160 000240 NOP
344 041162 075001 FADD+ R1 ;FLOATING ADD ON THE R1 STACK
    
```

```

345
346 041164 004767 012614 JSR PC, POBR IPOP THE ANSWER
347 041170 010167 176616 MOV R1, SSP ISAVE "STACK POINTER"
348 041174 022767 000040 176606 CMP #040, SPSW ICHECK PS (EXCEPT T BIT)
349 041202 001401 BEQ ,+4 IBRANCH IF OK
350 041204 104000 HLT IPS NOT EQUAL TO 040
351
352 041206 022767 040044 174576 CMP #STACK4,SSP ICHECK THE STACK POINTER (R1)
353 041214 001401 BEQ ,+4 IBRANCH IF OK
354 041216 104000 HLT ISTACK POINTER (R1) NOT EQUAL TO #STACK4
355
356 041220 022767 001357 174566 CMP #001357,ANS1 ICHECK FIRST HALF OF ANSWER
357 041226 001401 BEQ ,+4 IBRANCH IF OK
358 041230 104000 HLT IANS1 NOT EQUAL TO 001357
359
360 041232 022767 024642 174556 CMP #024642,ANS2 ICHECK SECOND HALF OF ANSWER
361 041240 001401 BEQ ,+4 IBRANCH IF OK
362 041242 104000 HLT IANS2 NOT EQUAL TO 024642
363
364 041244 122767 000007 174526 EN071 CMPB #7, ICNT ICHECK THE TEST NUMBER
365 041252 001401 BEQ ,+4 IBRANCH IF OK
366 041254 104000 HLT IWRONG TEST! PC MUST HAVE FOULED UP.
367
368 041256 104400 SCOPE
369
370
371
372
373
374
375
376
377 041260 012767 000010 174512 TSF101 MOV #10, ICNT IKEEP TRACK OF TEST NUMBER
378 041266 004567 012400 JSR R5, PUSHR IPUSH 4 WORDS ONTO R5 STACK, SET PRIORITY
379 041272 000200 000000 ,WORD 000200,000000 ISECOND OPERAND ON TOP
380 041276 100400 000000 ,WORD 100400,000000 IFIRST OPERAND ON BOTTOM
381 041302 000140 ,WORD 140 IPROCESSOR PRIORITY LEVEL
382 041304 054236 000340 ,WORD TRAPER,340 IFIS TRAP VECTOR
383 041310 012705 040040 MOV #STACK0,R5 ISET UP STACK POINTER
384
385 041314 000240 NOP
386 041316 075005 FADD+ R5 IFLOATING ADD ON THE R5 STACK
387
388 041320 004767 012460 JSR PC, POBR IPOP THE ANSWER
389 041324 010567 174462 MOV R5, SSP ISAVE "STACK POINTER"
390 041330 022767 000150 176452 CMP #150, SPSW ICHECK PS (EXCEPT T BIT)
391 041336 001401 BEQ ,+4 IBRANCH IF OK
392 041340 104000 HLT IPS NOT EQUAL TO 150
393
394 041342 022767 040044 176442 CMP #STACK4,SSP ICHECK THE STACK POINTER (R5)
395 041350 001401 BEQ ,+4 IBRANCH IF OK
396 041352 104000 HLT ISTACK POINTER (R5) NOT EQUAL TO #STACK4
397
398 041354 022767 100200 176432 CMP #100200,ANS1 ICHECK FIRST HALF OF ANSWER

```

```

*****
!TEST 101 FADD (KE11F FLOATING ADD INSTRUCTION)
! 100400,000000 + 000200,000000 = 100200,000000
! PS = 150, STACK POINTER = R5
*****

```

```

399 041362 001401          BEQ    ,+4          |BRANCH IF OK
400 041364 104000          HLT                    |ANS1 NOT EQUAL TO 104000
401
402 041366 022767 000000 174422      CMP    #000000,ANS2   |CHECK SECOND HALF OF ANSWER
403 041374 001401          BEQ    ,+4          |BRANCH IF OK
404 041376 104000          HLT                    |ANS2 NOT EQUAL TO 000000
405
406 041400 122767 000010 176372  EN010:  CMPB  #10,    ICNT   |CHECK THE TEST NUMBER
407 041406 001401          BEQ    ,+4          |BRANCH IF OK
408 041410 104000          HLT                    |WRONG TEST! PC MUST HAVE FOULED UP!
409
410 041412 104400          SCOPE
411
412
413
414
415
416
417
418
419 041414 012767 000011 176356  TSF11:  MOV    #11,    ICNT   |KEEP TRACK OF TEST NUMBER
420 041422 004567 012324          JSR    R5,    PUSHR   |PUSH 4 WORDS ONTO R3 STACK, SET PRIORITY
421 041426 000377 177777          ,WORD 000377,177777 |SECOND OPERAND ON TOP
422 041432 100200 000000          ,WORD 100200,000000 |FIRST OPERAND ON BOTTOM
423 041436 000157          ,WORD 157           |PROCESSOR PRIORITY LEVEL
424 041440 041470 000000          ,WORD ISR11, 000     |FIS TRAP VECTOR
425 041444 012703 040040          MOV    #STACK0,R3   |SET UP R3 AS STACK POINTER
426
427 041450 000240          NOP
428 041452 075003          FADD+  R3          |FLOATING ADD ON THE R3 STACK
429
430 041454 004767 012324          RTA11:  JSR    X7,    POPR   |POP THE "ANSWER"
431 041460 010367 176326          MOV    R3,    SSP   |GIVE STACK POINTER (R3)
432 041464 104000          HLT                    |FIS TRAP DIDN'T OCCURE!
433 041466 000454          BR     END11
434
435 041470 004767 012342          ISR11:  JSR    X7,    POPR   |POP ALL DATA OFF THE STACKS
436 041474 010367 176312          MOV    R3,    SSP   |GIVE STACK POINTER (R3)
437 041500 022767 000000 174302      CMP    #000,    SPW   |CHECK PS AFTER FIS TRAP
438 041506 001401          BEQ    ,+4          |BRANCH IF OK
439 041510 104000          HLT                    |PS AFTER FIS TRAP NOT EQUAL TO 000
440
441 041512 022767 040040 176272      CMP    #STACK0,SSP  |CHECK THE STACK POINTER (R3)
442 041520 001401          BEQ    ,+4          |BRANCH IF OK
443 041522 104000          HLT                    |STACK POINTER (R3) NOT EQUAL TO #STACK0
444
445 041524 022767 041454 176262      CMP    #RTA11, ANS1  |CHECK FIS TRAP RETURN ADDRESS
446 041532 001401          BEQ    ,+4          |BRANCH IF OK
447 041534 104000          HLT                    |FIS TRAP AT WRONG ADDRESS
448
449 041536 022767 000152 176292      CMP    #152,    ANS2  |CHECK PS BEFORE FIS TRAP
450 041544 001401          BEQ    ,+4          |BRANCH IF OK
451 041546 104000          HLT                    |PS AT FIS TRAP TIME NOT 152
452

```

453	041550	022767	000377	176242	CMP	#000377,ANS3	ICHECK DATA FROM THE STACK
454	041556	001401			BEQ	:+4	IBRANCH IF OK
455	041560	104000			HLT		IDATA ON STACK (000377) CHANGED
456							
457	041562	022767	177777	176232	CMP	#177777,ANS4	ICHECK DATA FROM STACK
458	041570	001401			BEQ	:+4	IBRANCH IF OK
459	041572	104000			HLT		IDATA ON STACK (177777) CHANGED
460							
461	041574	022767	100200	176222	CMP	#100200,ANS5	ICHECK DATA FROM STACK
462	041602	001401			BEQ	:+4	IBRANCH IF OK
463	041604	104000			HLT		IDATA ON STACK (100200) CHANGED
464							
465	041606	022767	000000	176212	CMP	#000000,ANS6	ICHECK DATA FROM STACK
466	041614	001401			BEQ	:+4	IBRANCH IF OK
467	041616	104000			HLT		IDATA ON STACK (000000) CHANGED
468							
469	041620	122767	000011	174152	ENB111	CMPB #11, ICNT	ICHECK THE TEST NUMBER
470	041626	001401			BEQ	:+4	IBRANCH IF OK
471	041630	104000			HLT		IWRONG TEST! PC MUST HAVE FOULED UP.
472							
473	041632	104400					SCOPE
474							
475							
476							
477							*****
478							ITEST I21 FADD (KE11F FLOATING ADD INSTRUCTION)
479							I 000425,052525 + 100252,125252 = 000200,000000
480							I PS = 000, STACK POINTER = R4
481							*****
482	041634	012767	000012	176136	TSF121	MOV #12, ICNT	KEEP TRACK OF TEST NUMBER
483	041642	004567	012104		JSR	R5, PUSHR	IPUSH 4 WORDS ONTO R4 STACK, SET PRIORITY
484	041646	100252	125252		,WORD	100252,125252	ISECOND OPERAND ON TOP
485	041652	000425	052525		,WORD	000425,052525	IFIRST OPERAND ON BOTTOM
486	041656	000017			,WORD	017	IPROCESSOR PRIORITY LEVEL
487	041660	054236	000340		,WORD	TRAPER,340	IFIS TRAP VECTOR
488	041664	012704	040040		MOV	#STACK0,R4	ISSET UP STACK POINTER
489							
490	041670	000240			NOP		
491	041672	075004			FADD+	R4	IFLOATING ADD ON THE R4 STACK
492							
493	041674	004767	012104		JSR	PC, POBR	IPOP THE ANSWER
494	041700	010467	176106		MOV	R4, SS#	ISAVE "STACK POINTER"
495	041704	022767	000000	176076	CMP	#000, SPSW	ICHECK PS (EXCEPT 7 BIT)
496	041712	001401			BEQ	:+4	IBRANCH IF OK
497	041714	104000			HLT		IPS NOT EQUAL TO 000
498							
499	041716	022767	040044	176066	CMP	#STACK4,SS#	ICHECK THE STACK POINTER (R4)
500	041724	001401			BEQ	:+4	IBRANCH IF OK
501	041726	104000			HLT		ISTACK POINTER (R4) NOT EQUAL TO #STACK4
502							
503	041730	022767	000200	176056	CMP	#000200,ANS1	ICHECK FIRST HALF OF ANSWER
504	041736	001401			BEQ	:+4	IBRANCH IF OK
505	041740	104000			HLT		IANS1 NOT EQUAL TO 000200
506							

```

507 041742 022767 000000 176046      CMP      #000000,ANS2      ICHECK SECOND HALF OF ANSWER
508 041750 001401                      BEQ      ,+4            IBRANCH IF OK
509 041752 104000                      HLT                               IANS2 NOT EQUAL TO 000000
510
511 041754 122767 000012 176016  END121  CMPB     #12,      ICNT      ICHECK THE TEST NUMBER
512 041762 001401                      BEQ      ,+4            IBRANCH IF OK
513 041764 104000                      HLT                               IWRONG TEST! PC MUST HAVE FOULED UP!
514
515 041766 104400                      SCOPE
516
517
518
519
520
521
522
523
524 041770 012767 000013 176002  TSF13I  MOV      #13,      ICNT      IKEEP TRACK OF TEST NUMBER
525 041776 004967 011972                      JSR      R5,      PUSH5     IPUSH 4 WORDS ONTO STACK, SET PRIORITY
526 042002 100377 177777                      ;WORD    100377,177777     ISECOND OPERAND ON TOP
527 042006 000200 000000                      ;WORD    000200,000000     IFIRST OPERAND ON BOTTOM
528 042012 000007                      ;WORD    057              IPROCESSOR PRIORITY LEVEL
529 042014 042040 000140                      ;WORD    ISR13, 140       IFIS TRAP VECTOR
530
531 042020 000240                      NOP
532 042022 079006                      PADD+   SP              IFLOATING ADD ON THE STACK
533
534 042024 004767 011604                      RTA13I  JSR      X7,      POP5      IPOP THE "ANSWER"
535 042030 104000                      HLT                               IFIS TRAP DIDN'T OCCURE!
536 042032 016706 175744                      MOV      STACK, SP      IRESTORE THE STACK POINTER
537 042036 000494                      BR      END13
538
539 042040 004767 011624                      ISA13I  JSR      X7,      POP5     IPOP ALL DATA OFF THE STACK
540 042044 026706 175732                      CMP      STACK, SP      ICHECK THE STACK POINTER
541 042050 001404                      BEQ      ISA13          IBRANCH IF OK
542 042052 016706 175724                      MOV      STACK, SP      IRESTORE THE STACK POINTER
543 042056 104000                      HLT                               ISTACK POINTER FOULED UP
544 042060 000443                      BR      END13          ISKIP REST OF TEST
545
546 042062 022767 000140 175720  ISA13I  CMP      #140,     SP#W     ICHECK PS AFTER FIS TRAP
547 042070 001401                      BEQ      ,+4            IBRANCH IF OK
548 042072 104000                      HLT                               IPS AFTER FIS TRAP NOT EQUAL TO 140
549
550 042074 022767 042024 175712                      CMP      #RTA13, ANS1    ICHECK FIS TRAP RETURN ADDRESS
551 042102 001401                      BEQ      ,+4            IBRANCH IF OK
552 042104 104000                      HLT                               IFIS TRAP AT WRONG ADDRESS
553
554 042106 022767 000052 175702                      CMP      #052,     ANS2    ICHECK PS BEFORE FIS TRAP
555 042114 001401                      BEQ      ,+4            IBRANCH IF OK
556 042116 104000                      HLT                               IPS AT FIS TRAP TIME NOT 052
557
558 042120 022767 100377 175672                      CMP      #100377,ANS3    ICHECK DATA FROM THE STACK
559 042126 001401                      BEQ      ,+4            IBRANCH IF OK
560 042130 104000                      HLT                               IDATA ON STACK (100377) CHANGED

```

```

;*****
;TEST I3I      PADD (KE11F FLOATING ADD INSTRUCTION)
; 000200,000000 + 100377,177777 ==> UNDERFLOW
; PS(ON STACK) = 052;      STACK POINTER = SP
;*****

```



561										
562	042132	022767	177777	175662		CMP	#177777,ANS4			ICHECK DATA FROM STACK
563	042140	001401				BEG	,+4			IBRANCH IF OK
564	042142	104000				HLT				IDATA ON STACK (177777) CHANGED
565										
566	042144	022767	000200	175652		CMP	#000200,ANS5			ICHECK DATA FROM STACK
567	042152	001401				BEG	,+4			IBRANCH IF OK
568	042154	104000				HLT				IDATA ON STACK (000200) CHANGED
569										
570	042156	022767	000000	175642		CMP	#000000,ANS6			ICHECK DATA FROM STACK
571	042164	001401				BEG	,+4			IBRANCH IF OK
572	042166	104000				HLT				IDATA ON STACK (000000) CHANGED
573										
574	042170	122767	000013	175602	END13:	CMPB	#13, ICNT			ICHECK THE TEST NUMBER
575	042176	001401				BEG	,+4			IBRANCH IF OK
576	042200	104000				HLT				!WRONG TEST! PC MUST HAVE FOULED UP!
577										
578	042202	104400								SCOPE
579										
580										
581										
582										*****
583										!TEST 14: FADD (KE11F FLOATING ADD INSTRUCTION)
584										! 100425,052525 + 000252,125252 = 100200,000000
585										! PS = 110, STACK POINTER = SP
586										*****
587	042204	012767	000014	175566	TS#14:	MOV	#14, ICNT			!KEEP TRACK OF TEST NUMBER
588	042212	004567	011356			JSR	R5, PUSH5			!PUSH 4 WORDS ONTO STACK, SET PRIORITY
589	042216	000252	125252			,WORD	000252,125252			!SECOND OPERAND ON TOP
590	042222	100425	052525			,WORD	100425,052525			!FIRST OPERAND ON BOTTOM
591	042226	000107				,WORD	107			!PROCESSOR PRIORITY LEVEL
592	042230	054236	000340			,WORD	TRAPER,340			!FIS TRAP VECTOR
593										
594	042234	000240				NOP				
595	042236	075006				FADD+	SP			!FLOATING ADD ON THE STACK
596										
597	042240	004767	011370			JSR	PC, POP5			!POP THE ANSWER
598	042244	026706	175532			CMP	STACK, SP			ICHECK THE STACK POINTER
599	042250	001404				BEG	TS#14			IBRANCH IF OK
600	042252	016706	175524			MOV	STACK, SP			!RESTORE STACK POINTER
601	042256	104000				HLT				!STACK POINTER FOULED UP
602	042260	000417				BR	END14			!SKIP REST OF TEST
603										
604	042262	022767	000110	175520	TS#14:	CMP	#110, SPSW			ICHECK PS (EXCEPT 7 BIT)
605	042270	001401				BEG	,+4			IBRANCH IF OK
606	042272	104000				HLT				!PS NOT EQUAL TO 110
607										
608	042274	022767	100200	175512		CMP	#100200,ANS1			ICHECK FIRST HALF OF ANSWER
609	042302	001401				BEG	,+4			IBRANCH IF OK
610	042304	104000				HLT				!ANS1 NOT EQUAL TO 100200
611										
612	042306	022767	000000	175502		CMP	#000000,ANS2			ICHECK SECOND HALF OF ANSWER
613	042314	001401				BEG	,+4			IBRANCH IF OK
614	042316	104000				HLT				!ANS2 NOT EQUAL TO 000000

```

615
616 042320 122767 000014 175452 EN014: CMPB #14, ICNT ICHECK THE TEST NUMBER
617 042326 001401 BEQ ,+4 IBRANCH IF OK
618 042330 104000 HLT IWRONG TEST! PC MUST HAVE FOULED UP!
619
620 042332 104400 SCOPE
621
622
623 ;*****
624 ITEST I5) FADD (KE11F FLOATING ADD INSTRUCTION)
625 I 077652,125252 + 077452,125252 = 077777,177777
626 I PS = 040, STACK POINTER = SP
627 ;*****
628
629 042334 012767 000015 175436 TS15: MOV #15, ICNT IKEEP TRACK OF TEST NUMBER
630 042342 004567 011226 JSR R5, PUSH5 IPUSH 4 WORDS ONTO STACK, SET PRIORITY
631 042346 077452 125252 ,WORD 077452,125252 ISECOND OPERAND ON TOP
632 042352 077652 125252 ,WORD 077652,125252 IFIRST OPERAND ON BOTTOM
633 042356 000057 ,WORD 057 IPROCESSOR PRIORITY LEVEL
634 042360 054236 000340 ,WORD TRAPER,340 IFIS TRAP VECTOR
635
636 042364 000240 NOP
637 042366 075006 FADD+ SP IFLOATING ADD ON THE STACK
638
639 042370 004767 011240 JSR PC, POPS IPOP THE ANSWER
640 042374 026706 175402 CMP STACK, SP ICHECK THE STACK POINTER
641 042400 001404 BEQ TSA15 IBRANCH IF OK
642 042402 016706 175374 MOV STACK, SP IRESTORE STACK POINTER
643 042406 104000 HLT ISTACK POINTER FOULED UP
644 042410 000417 BR END15 ISKIP REST OF TEST
645
646 042412 022767 000040 175370 TSA15: CMP #040, SP0W ICHECK PS (EXCEPT T BIT)
647 042420 001401 BEQ ,+4 IBRANCH IF OK
648 042422 104000 HLT IPS NOT EQUAL TO 040
649
650 042424 022767 077777 175362 CMP #077777,ANS1 ICHECK FIRST HALF OF ANSWER
651 042432 001401 BEQ ,+4 IBRANCH IF OK
652 042434 104000 HLT IANS1 NOT EQUAL TO 077777
653
654 042436 022767 177777 175352 CMP #177777,ANS2 ICHECK SECOND HALF OF ANSWER
655 042444 001401 BEQ ,+4 IBRANCH IF OK
656 042446 104000 HLT IANS2 NOT EQUAL TO 177777
657
658 042450 122767 000015 175322 EN015: CMPB #15, ICNT ICHECK THE TEST NUMBER
659 042456 001401 BEQ ,+4 IBRANCH IF OK
660 042460 104000 HLT IWRONG TEST! PC MUST HAVE FOULED UP!
661
662 042462 104400 SCOPE
663
664
665 ;*****
666 ITEST I6) FADD (KE11F FLOATING ADD INSTRUCTION)
667 I 177452,125253 + 177652,125252 ==> OVERFLOW
668 I PS(ON STACK) = 102, STACK POINTER = R1

```

```

669
670
671 042464 012767 000016 175306 TSF16: MOV #16, ICNT ;KEEP TRACK OF TEST NUMBER
672 042472 004567 011254 JSR R5, PUSHR ;PUSH 4 WORDS ONTO R1 STACK, SET PRIORITY
673 042476 177652 125252 ;WORD 177652,125252 ;SECOND OPERAND ON TOP
674 042502 177452 125253 ;WORD 177452,125253 ;FIRST OPERAND ON BOTTOM
675 042506 000105 ;WORD 105 ;PROCESSOR PRIORITY LEVEL
676 042510 042540 000052 ;WORD ISR16, 052 ;FIS TRAP VECTOR
677 042514 012701 040040 MOV #STACK0,R1 ;SET UP R1 AS STACK POINTER
678
679 042520 000240 NOP
680 042522 075001 FADD+ R1 ;FLOATING ADD ON THE R1 STACK
681
682 042524 004767 011254 RTA16: JSR X7, POPR ;POP THE "ANSWER"
683 042530 010167 175256 MOV R1, SSP ;SAVE STACK POINTER (R1)
684 042534 104000 HLT ;FIS TRAP DIDN'T OCCURE!
685 042536 000454 BR END16
686
687 042540 004767 011272 ISR16: JSR X7, POPER ;POP ALL DATA OFF THE STACKS
688 042544 010167 175242 MOV R1, SSP ;SAVE STACK POINTER (R1)
689 042550 022767 000052 175232 CMP #052, SPSW ;CHECK PS AFTER FIS TRAP
690 042556 001401 BEQ ,+4 ;BRANCH IF OK
691 042560 104000 HLT ;PS AFTER FIS TRAP NOT EQUAL TO 052
692
693 042562 022767 040040 175222 CMP #STACK0,SSP ;CHECK THE STACK POINTER (R1)
694 042570 001401 BEQ ,+4 ;BRANCH IF OK
695 042572 104000 HLT ;STACK POINTER (R1) NOT EQUAL TO #STACK0
696
697 042574 022767 042524 175212 CMP #RTA16,ANS1 ;CHECK FIS TRAP RETURN ADDRESS
698 042602 001401 BEQ ,+4 ;BRANCH IF OK
699 042604 104000 HLT ;FIS TRAP AT WRONG ADDRESS
700
701 042606 022767 000102 175202 CMP #102,ANS2 ;CHECK PS BEFORE FIS TRAP
702 042614 001401 BEQ ,+4 ;BRANCH IF OK
703 042616 104000 HLT ;PS AT FIS TRAP TIME NOT 102
704
705 042620 022767 177652 175172 CMP #177652,ANS3 ;CHECK DATA FROM THE STACK
706 042626 001401 BEQ ,+4 ;BRANCH IF OK
707 042630 104000 HLT ;DATA ON STACK (177652) CHANGED
708
709 042632 022767 125252 175162 CMP #125252,ANS4 ;CHECK DATA FROM STACK
710 042640 001401 BEQ ,+4 ;BRANCH IF OK
711 042642 104000 HLT ;DATA ON STACK (125252) CHANGED
712
713 042644 022767 177452 175152 CMP #177452,ANS5 ;CHECK DATA FROM STACK
714 042652 001401 BEQ ,+4 ;BRANCH IF OK
715 042654 104000 HLT ;DATA ON STACK (177452) CHANGED
716
717 042656 022767 125253 175142 CMP #125253,ANS6 ;CHECK DATA FROM STACK
718 042664 001401 BEQ ,+4 ;BRANCH IF OK
719 042666 104000 HLT ;DATA ON STACK (125253) CHANGED
720
721 042670 122767 000016 175102 END16: CMPB #16, ICNT ;CHECK THE TEST NUMBER
722 042676 001401 BEQ ,+4 ;BRANCH IF OK
    
```

```

723 042700 104000          HLT          ;WRONG TEST! PC MUST HAVE FOULED UP;
724
725 042702 104400          SCOPE
726
727
728
729
730
731
732
733
734 042704 012767 000017 175066 TST17: MOV    #17,    ICNT    ;KEEP TRACK OF TEST NUMBER
735 042712 004567 011034      JSR    R5,    PUSHR   ;PUSH 4 WORDS ONTO R4 STACK, SET PRIORITY
736 042716 177652 125252      ;WORD 177652,125252 ;SECOND OPERAND ON TOP
737 042722 177452 125252      ;WORD 177452,125252 ;FIRST OPERAND ON BOTTOM
738 042726 000157              ;WORD 157           ;PROCESSOR PRIORITY LEVEL
739 042730 054236 000340      ;WORD TRAPER,340    ;FIS TRAP VECTOR
740 042734 012704 040040      MOV    #STACK0,R4   ;SET UP STACK POINTER
741
742 042740 000240              NOP
743 042742 075004              FADD+  R4           ;FLOATING ADD ON THE R4 STACK
744
745 042744 004767 011034      JSR    PC,    POBR   ;POP THE ANSWER
746 042750 010467 175036      MOV    R4,    SSP    ;SAVE "STACK POINTER"
747 042754 022767 000150 175026 CMP    #150,   SPSW   ;CHECK PS (EXCEPT 1 BIT)
748 042762 001401              BEQ    ,+4          ;BRANCH IF OK
749 042764 104000              HLT
750
751 042766 022767 040044 175016 CMP    #STACK4,SSP   ;CHECK THE STACK POINTER (R4)
752 042774 001401              BEQ    ,+4          ;BRANCH IF OK
753 042776 104000              HLT
754
755 043000 022767 177777 175006 CMP    #177777,ANS1  ;CHECK FIRST HALF OF ANSWER
756 043006 001401              BEQ    ,+4          ;BRANCH IF OK
757 043010 104000              HLT
758
759 043012 022767 177777 174776 CMP    #177777,ANS2  ;CHECK SECOND HALF OF ANSWER
760 043020 001401              BEQ    ,+4          ;BRANCH IF OK
761 043022 104000              HLT
762
763 043024 122767 000017 174746 EN017: CMPB   #17,    ICNT    ;CHECK THE TEST NUMBER
764 043032 001401              BEQ    ,+4          ;BRANCH IF OK
765 043034 104000              HLT
766
767 043036 104400          SCOPE
768
769
770
771
772
773
774
775
776 043040 012767 000020 174732 TST20: MOV    #20,    ICNT    ;KEEP TRACK OF TEST NUMBER
    
```

```

;*****
;TEST 17: FADD (KE11F FLOATING ADD INSTRUCTION)
; 177452,125252 + 177652,125252 = 177777,177777
; PS = 150, STACK POINTER = R4
;*****
    
```

```

;*****
;TEST 20: FADD (KE11F FLOATING ADD INSTRUCTION)
; 077652,125253 + 077452,125252 ==> OVERFLOW
; PS(ON STACK) = 002, STACK POINTER = SP
;*****
    
```

777	043046	004567	010522		JSR	R5,	PUSHS		IPUSH 4 WORDS ONTO STACK, SET PRIORITY
778	043052	077452	125252		,WORD	077452,125252			ISECOND OPERAND ON TOP
779	043056	077652	125253		,WORD	077652,125253			IFIRST OPERAND ON BOTTOM
780	043062	000003			,WORD	003			IPROCESSOR PRIORITY LEVEL
781	043064	043110	000144		,WORD	ISR20, 144			IFIS TRAP VECTOR
782									
783	043070	000240			NOP				
784	043072	075006			FADD+	SP			IFLOATING ADD ON THE STACK
785									
786	043074	004767	010534		RTA20:	JSR	X7,	POPS	IPOP THE "ANSWER"
787	043100	104000			HLT				IFIS TRAP DIDN'T OCCURE!
788	043102	016706	174674		MOV	STACK, SP			IRESTORE THE STACK POINTER
789	043106	000454			BR	END20			
790									
791	043110	004767	010554		ISR20:	JSR	X7,	POPS	IPOP ALL DATA OFF THE STACK
792	043114	026706	174662		CMP	STACK, SP			ICHECK THE STACK POINTER
793	043120	001404			BEG	ISA20			IBRANCH IF OK
794	043122	016706	174654		MOV	STACK, SP			IRESTORE THE STACK POINTER
795	043126	104000			HLT				ISTACK POINTER FOULED UP
796	043130	000443			BR	END20			ISRIP REST OF TEST
797									
798	043132	022767	000144	174650	ISA20:	CMP	#144,	SP0W	ICHECK PS AFTER FIS TRAP
799	043140	001401			BEG	,+4			IBRANCH IF OK
800	043142	104000			HLT				IPS AFTER FIS TRAP NOT EQUAL TO 144
801									
802	043144	022767	043074	174642	CMP	#RTA20, AN91			ICHECK FIS TRAP RETURN ADDRESS
803	043152	001401			BEG	,+4			IBRANCH IF OK
804	043154	104000			HLT				IFIS TRAP AT WRONG ADDRESS
805									
806	043156	022767	000002	174632	CMP	#002, AN92			ICHECK PS BEFORE FIS TRAP
807	043164	001401			BEG	,+4			IBRANCH IF OK
808	043166	104000			HLT				IPS AT FIS TRAP TIME NOT 002
809									
810	043170	022767	077452	174622	CMP	#077452, AN93			ICHECK DATA FROM THE STACK
811	043176	001401			BEG	,+4			IBRANCH IF OK
812	043200	104000			HLT				IDATA ON STACK (077452) CHANGED
813									
814	043202	022767	125252	174612	CMP	#125252, AN94			ICHECK DATA FROM STACK
815	043210	001401			BEG	,+4			IBRANCH IF OK
816	043212	104000			HLT				IDATA ON STACK (125252) CHANGED
817									
818	043214	022767	077652	174602	CMP	#077652, AN95			ICHECK DATA FROM STACK
819	043222	001401			BEG	,+4			IBRANCH IF OK
820	043224	104000			HLT				IDATA ON STACK (077652) CHANGED
821									
822	043226	022767	125253	174572	CMP	#125253, AN96			ICHECK DATA FROM STACK
823	043234	001401			BEG	,+4			IBRANCH IF OK
824	043236	104000			HLT				IDATA ON STACK (125253) CHANGED
825									
826	043240	122767	000020	174532	END20:	CMPB	#20,	ICNT	ICHECK THE TEST NUMBER
827	043246	001401			BEG	,+4			IBRANCH IF OK
828	043250	104000			HLT				IWRONG TEST! PC MUST HAVE FOULED UP!
829									
830	043252	104400			SCOPE				

831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884

043254 012767 000021 174516  
043262 004567 010464  
043266 043125 052525  
043272 035152 125252  
043276 000147  
043300 054236 000340  
043304 012703 040040  
043310 000240  
043312 075003  
043314 004767 010464  
043320 010367 174466  
043324 022767 000140 174436  
043332 001401  
043334 104000  
043336 022767 040044 174446  
043344 001401  
043346 104000  
043350 022767 043125 174436  
043356 001401  
043360 104000  
043362 022767 052526 174426  
043370 001401  
043372 104000  
043374 122767 000021 174376  
043402 001401  
043404 104000  
043406 104400

```
*****  
ITEST 21 FADD (KE11F FLOATING ADD INSTRUCTION)  
| 035152,125252 + 043125,052525 = 043125,052526  
| PS = 140, STACK POINTER = R3  
*****
```

```
TST21: MOV #21, ICNT ;KEEP TRACK OF TEST NUMBER  
JSR R5, PUSHR ;PUSH 4 WORDS ONTO R3 STACK, SET PRIORITY  
;WORD 043125,052525 ;SECOND OPERAND ON TOP  
;WORD 035152,125252 ;FIRST OPERAND ON BOTTOM  
;WORD 147 ;PROCESSOR PRIORITY LEVEL  
;WORD TRAPER,340 ;FIS TRAP VECTOR  
MOV #STACK0,R3 ;SET UP STACK POINTER  
  
NOP  
FADD+ R3 ;FLOATING ADD ON THE R3 STACK  
  
JSR PC, POPR ;POP THE ANSWER  
MOV R3, SS# ;SAVE "STACK POINTER"  
CMP #140, SPSW ;CHECK PS (EXCEPT T BIT)  
BEQ ,+4 ;BRANCH IF OK  
HLT ;PS NOT EQUAL TO 140  
  
CMP #STACK4,SS# ;CHECK THE STACK POINTER (R3)  
BEQ ,+4 ;BRANCH IF OK  
HLT ;STACK POINTER (R3) NOT EQUAL TO #STACK4  
  
CMP #043125,ANS1 ;CHECK FIRST HALF OF ANSWER  
BEQ ,+4 ;BRANCH IF OK  
HLT ;ANS1 NOT EQUAL TO 043125  
  
CMP #052526,ANS2 ;CHECK SECOND HALF OF ANSWER  
BEQ ,+4 ;BRANCH IF OK  
HLT ;ANS2 NOT EQUAL TO 052526  
  
EN021: CMPB #21, ICNT ;CHECK THE TEST NUMBER  
BEQ ,+4 ;BRANCH IF OK  
HLT ;WRONG TEST! PC MUST HAVE FOULED UP!
```

SCOPE

```
*****  
ITEST 22 FADD (KE11F FLOATING ADD INSTRUCTION)  
| 143325,052525 + 135152,125252 = 143325,052525  
| PS = 050, STACK POINTER = R0  
*****
```

```
TST22: MOV #22, ICNT ;KEEP TRACK OF TEST NUMBER  
JSR R5, PUSHR ;PUSH 4 WORDS ONTO R0 STACK, SET PRIORITY  
;WORD 135152,125252 ;SECOND OPERAND ON TOP  
;WORD 143325,052525 ;FIRST OPERAND ON BOTTOM
```

885	043432	000043				,WORD	043		PROCESSOR PRIORITY LEVEL
886	043434	054236	000340			,WORD	TRAPER,340		FIS TRAP VECTOR
887	043440	012700	040040			MOV	#STACK0,R0		ISSET UP STACK POINTER
888									
889	043444	000240				NOP			
890	043446	075000				FADD+	R0		IFLOATING ADD ON THE R0 STACK
891									
892	043450	004767	010330			JSR	PC, POPR		IPOP THE ANSWER
893	043454	010067	174332			MOV	R0, SSP		ISAVE "STACK POINTER"
894	043460	022767	000050	174322		CMP	#050, SPSW		ICHECK PS (EXCEPT T BIT)
895	043466	001401				BEQ	,+4		IBRANCH IF OK
896	043470	104000				HLT			IPS NOT EQUAL TO 050
897									
898	043472	022767	040044	174312		CMP	#STACK4,SS#		ICHECK THE STACK POINTER (R0)
899	043500	001401				BEQ	,+4		IBRANCH IF OK
900	043502	104000				HLT			ISTACK POINTER (R0) NOT EQUAL TO #STACK4
901									
902	043504	022767	143325	174302		CMP	#143325,ANS1		ICHECK FIRST HALF OF ANSWER
903	043512	001401				BEQ	,+4		IBRANCH IF OK
904	043514	104000				HLT			IANS1 NOT EQUAL TO 143325
905									
906	043516	022767	052525	174272		CMP	#052525,ANS2		ICHECK SECOND HALF OF ANSWER
907	043524	001401				BEQ	,+4		IBRANCH IF OK
908	043526	104000				HLT			IANS2 NOT EQUAL TO 052525
909									
910	043530	122767	000022	174242	END221	CMPB	#22, ICNT		ICHECK THE TEST NUMBER
911	043536	001401				BEQ	,+4		IBRANCH IF OK
912	043540	104000				HLT			IFRONG TEST! PC MUST HAVE FOULED UP!
913									
914	043542	104400				SCOPE			
915									
916									
917									
918									
919									
920									
921									
922									
923	043544	012767	000023	174226	TS#231	MOV	#23, ICNT		IKEEP TRACK OF TEST NUMBER
924	043552	004567	010174			JSR	R5, PUSHR		IPUSH 4 WORDS ONTO R5 STACK, SET PRIORITY
925	043556	143325	052525			,WORD	143325,052525		ISECOND OPERAND ON TOP
926	043562	135152	125252			,WORD	135152,125252		IFIRST OPERAND ON BOTTOM
927	043566	000157				,WORD	157		IPROCESSOR PRIORITY LEVEL
928	043570	054236	000340			,WORD	TRAPER,340		IFIS TRAP VECTOR
929	043574	012700	040040			MOV	#STACK0,R5		ISSET UP STACK POINTER
930									
931	043600	000240				NOP			
932	043602	075000				FADD+	R5		IFLOATING ADD ON THE R5 STACK
933									
934	043604	004767	010174			JSR	PC, POPR		IPOP THE ANSWER
935	043610	010567	174176			MOV	R5, SSP		ISAVE "STACK POINTER"
936	043614	022767	000150	174166		CMP	#150, SPSW		ICHECK PS (EXCEPT T BIT)
937	043622	001401				BEQ	,+4		IBRANCH IF OK
938	043624	104000				HLT			IPS NOT EQUAL TO 150

```

|*****
|TEST 231      FADD (KE11F FLOATING ADD INSTRUCTION)
|      135152,125252 + 143325,052525 = 143325,052525
|      PS = 150,      STACK POINTER = R5
|*****

```

```

939
940 043626 022767 040044 174156      CMP      #STACK4,SSP      ;CHECK THE STACK POINTER (R5)
941 043634 001401                      BEQ      ,+4              ;BRANCH IF OK
942 043636 104000                      HLT
943
944 043640 022767 143325 174146      CMP      #143325,ANS1    ;CHECK FIRST HALF OF ANSWER
945 043646 001401                      BEQ      ,+4              ;BRANCH IF OK
946 043650 104000                      HLT
947
948 043652 022767 052525 174136      CMP      #052525,ANS2    ;CHECK SECOND HALF OF ANSWER
949 043660 001401                      BEQ      ,+4              ;BRANCH IF OK
950 043662 104000                      HLT
951
952 043664 122767 000023 174106  EN023i  CMPB     #23,    ICNT     ;CHECK THE TEST NUMBER
953 043672 001401                      BEQ      ,+4              ;BRANCH IF OK
954 043674 104000                      HLT
955
956 043676 104400                      SCOPE
957
958
959
960
961
962
963
964
965 043700 012767 000024 174072  TS#24i  MOV      #24,    ICNT     ;KEEP TRACK OF TEST NUMBER
966 043706 004567 010040                      JSR      R5,    PUSHR    ;PUSH 4 WORDS ONTO R2 STACK, SET PRIORITY
967 043712 035152 125252                      ;WORD    035152,125252  ;SECOND OPERAND ON TOP
968 043716 043125 052525                      ;WORD    043125,052525  ;FIRST OPERAND ON BOTTOM
969 043722 000040                      ;WORD    040              ;PROCESSOR PRIORITY LEVEL
970 043724 054236 000340                      ;WORD    TRAPER,340      ;FIS TRAP VECTOR
971 043730 012702 040040                      MOV      #STACK0,R2     ;SET UP STACK POINTER
972
973 043734 000240                      NOP
974 043736 075002                      FADD+   R2              ;FLOATING ADD ON THE R2 STACK
975
976 043740 004767 010040                      JSR      PC,    POPR     ;POP THE ANSWER
977 043744 010267 174042                      MOV      R2,    SSP     ;SAVE "STACK POINTER"
978 043750 022767 000040 174032      CMP      #040,    SP5W   ;CHECK PS (EXCEPT 7 BIT)
979 043756 001401                      BEQ      ,+4              ;BRANCH IF OK
980 043760 104000                      HLT
981
982 043762 022767 040044 174022      CMP      #STACK4,SSP    ;CHECK THE STACK POINTER (R2)
983 043770 001401                      BEQ      ,+4              ;BRANCH IF OK
984 043772 104000                      HLT
985
986 043774 022767 043125 174012      CMP      #043125,ANS1    ;CHECK FIRST HALF OF ANSWER
987 044002 001401                      BEQ      ,+4              ;BRANCH IF OK
988 044004 104000                      HLT
989
990 044006 022767 052526 174002      CMP      #052526,ANS2    ;CHECK SECOND HALF OF ANSWER
991 044014 001401                      BEQ      ,+4              ;BRANCH IF OK
992 044016 104000                      HLT
    
```

```

;*****
;TEST 24i      FADD (KE11F FLOATING ADD INSTRUCTION)
;      043125,052525 + 035152,125252 = 043125,052526
;      PS = 040,      STACK POINTER = R2
;*****
    
```



```

993
994 044020 122767 000024 173752 END24: CMPB #24, ICNT ICHECK THE TEST NUMBER
995 044026 001401 BEQ ,+4 IBRANCH IF OK
996 044030 104000 HLT IWRONG TEST! PC MUST HAVE FOULED UP!
997
998 044032 104400 SCOPE
999
1000
1001
1002 |*****
1003 |TEST 25: FSUB (KE11F FLOATING SUBTRACT INSTRUCTION)
1004 | 040200,000000 - 140200,000000 = 040400,000000
1005 | PS = 040, STACK POINTER = R1
1006 |*****
1007 044034 012767 000025 173736 TS25: MOV #25, ICNT IKEEP TRACK OF TEST NUMBER
1008 044042 004567 007704 JSR R5, PUSHR IPUSH 4 WORDS ONTO R1 STACK, SET PRIORITY
1009 044046 140200 000000 ;WORD 140200,000000 ISECOND OPERAND ON TOP
1010 044052 040200 000000 ;WORD 040200,000000 IFIRST OPERAND ON BOTTOM
1011 044056 000040 ;WORD 040 IPROCESSOR PRIORITY LEVEL
1012 044060 054236 000340 ;WORD TRAPER, 340 IFIS TRAP VECTOR
1013 044064 012701 040040 MOV #STACK0,R1 ISET UP STACK POINTER
1014
1015 044070 000240 NOP
1016 044072 075011 FSUB+ R1 IFLOATING SUBTRACT ON THE R1 STACK
1017
1018 044074 004767 007704 JSR PC, POPR IPOP THE ANSWER
1019 044100 010167 173706 MOV R1, SSP ISAVE "STACK POINTER"
1020 044104 022767 000040 173676 CMP #040, SPSW ICHECK PS (EXCEPT 1 BIT)
1021 044112 001401 BEQ ,+4 IBRANCH IF OK
1022 044114 104000 HLT IPS NOT EQUAL TO 040
1023
1024 044116 022767 040044 173666 CMP #STACK4,SSP ICHECK THE STACK POINTER (R1)
1025 044124 001401 BEQ ,+4 IBRANCH IF OK
1026 044126 104000 HLT ISTACK POINTER (R1) NOT EQUAL TO #STACK4
1027
1028 044130 022767 040400 173656 CMP #040400,ANS1 ICHECK FIRST HALF OF ANSWER
1029 044136 001401 BEQ ,+4 IBRANCH IF OK
1030 044140 104000 HLT IANS1 NOT EQUAL TO 040400
1031
1032 044142 022767 000000 173646 CMP #000000,ANS2 ICHECK SECOND HALF OF ANSWER
1033 044150 001401 BEQ ,+4 IBRANCH IF OK
1034 044152 104000 HLT IANS2 NOT EQUAL TO 000000
1035
1036 044154 122767 000025 173616 END25: CMPB #25, ICNT ICHECK THE TEST NUMBER
1037 044162 001401 BEQ ,+4 IBRANCH IF OK
1038 044164 104000 HLT IWRONG TEST! PC MUST HAVE FOULED UP!
1039
1040 SCOPE
1041
1042
1043 |*****
1044 |TEST 26: FSUB (KE11F FLOATING SUBTRACT INSTRUCTION)
1045 | 177777,177777 - 177777,177777 = 000000,000000
1046 | PS = 104, STACK POINTER = R2
    
```

```

1047
1048
1049 044170 012767 000026 173602 TS#261 MOV #26, ICNT IKEEP TRACK OF TEST NUMBER
1050 044176 004507 007590 JSR R5, PUSHR IPUSH 4 WORDS ONTO R2 STACK, SET PRIORITY
1051 044202 177777 177777 ;WORD 177777,177777 ISECOND OPERAND ON TOP
1052 044206 177777 177777 ;WORD 177777,177777 IFIRST OPERAND ON BOTTOM
1053 044212 000100 ;WORD 100 IPOCESSOR PRIORITY LEVEL
1054 044214 054236 000340 ;WORD TRAPER, 340 IFIS TRAP VECTOR
1055 044220 012702 040040 MOV #STACK0,R2 ISET UP STACK POINTER
1056
1057 044224 000240 NOP
1058 044226 079012 FSUB+ R2 IFLOATING SUBTRACT ON THE R2 STACK
1059
1060 044230 004707 007550 JSR PC, POPR IPOP THE ANSWER
1061 044234 010267 173552 MOV R2, SS# ISIVE "STACK POINTER"
1062 044240 022707 000104 173542 CMP #104, SP$W ICHECK PS (EXCEPT T BIT)
1063 044246 001401 BEQ ,+4 IBRANCH IF OK
1064 044250 104000 HLT IPS NOT EQUAL TO 104
1065
1066 044252 022707 040044 173532 CMP #STACK4,SS# ICHECK THE STACK POINTER (R2)
1067 044260 001401 BEQ ,+4 IBRANCH IF OK
1068 044262 104000 HLT ISTACK POINTER (R2) NOT EQUAL TO #STACK4
1069
1070 044264 022707 000000 173522 CMP #000000,ANS1 ICHECK FIRST HALF OF ANSWER
1071 044272 001401 BEQ ,+4 IBRANCH IF OK
1072 044274 104000 HLT IANS1 NOT EQUAL TO 000000
1073
1074 044276 022707 000000 173512 CMP #000000,ANS2 ICHECK SECOND HALF OF ANSWER
1075 044304 001401 BEQ ,+4 IBRANCH IF OK
1076 044306 104000 HLT IANS2 NOT EQUAL TO 000000
1077
1078 044310 122767 000026 173462 END#261 CMPB #26, ICNT ICHECK THE TEST NUMBER
1079 044316 001401 BEQ ,+4 IBRANCH IF OK
1080 044320 104000 HLT IWRONG TEST! PC MUST HAVE FOULED UP!
1081
1082 044322 104400 SCOPE
1083
1084
1085
1086 |*****
1087 |TEST 271 FSUB (KE11F FLOATING SUBTRACT INSTRUCTION)
1088 | 125252,125252 * 125252,125253 * 017400,000000
1089 | PS = 040, STACK POINTER = R0
1090 |*****
1091 044324 012767 000027 173446 TS#271 MOV #27, ICNT IKEEP TRACK OF TEST NUMBER
1092 044332 004507 007414 JSR R5, PUSHR IPUSH 4 WORDS ONTO R0 STACK, SET PRIORITY
1093 044336 125252 125253 ;WORD 125252,125253 ISECOND OPERAND ON TOP
1094 044342 125252 125252 ;WORD 125252,125252 IFIRST OPERAND ON BOTTOM
1095 044346 000047 ;WORD 047 IPOCESSOR PRIORITY LEVEL
1096 044350 054236 000340 ;WORD TRAPER, 340 IFIS TRAP VECTOR
1097 044354 012700 040040 MOV #STACK0,R0 ISET UP STACK POINTER
1098
1099 044360 000240 NOP
1100 044362 079010 FSUB+ R0 IFLOATING SUBTRACT ON THE R0 STACK

```

1101									
1102	044364	004767	007414		JSR	PC,	POPR	IPOP THE ANSWER	
1103	044370	010067	173416		MOV	R0,	ESP	ISAVE "STACK POINTER"	
1104	044374	022767	000040	173406	CMP	#040,	SPSW	ICHECK PS (EXCEPT T BIT)	
1105	044402	001401			BEG	,+4		IBRANCH IF OK	
1106	044404	104000			HLT			IPS NOT EQUAL TO 040	
1107									
1108	044406	022767	040044	173376	CMP	#STACK4,SSP		ICHECK THE STACK POINTER (R0)	
1109	044414	001401			BEG	,+4		IBRANCH IF OK	
1110	044416	104000			HLT			ISTACK POINTER (R0) NOT EQUAL TO #STACK4	
1111									
1112	044420	022767	017400	173366	CMP	#017400,ANS1		ICHECK FIRST HALF OF ANSWER	
1113	044426	001401			BEG	,+4		IBRANCH IF OK	
1114	044430	104000			HLT			IANS1 NOT EQUAL TO 017400	
1115									
1116	044432	022767	000000	173356	CMP	#000000,ANS2		ICHECK SECOND HALF OF ANSWER	
1117	044440	001401			BEG	,+4		IBRANCH IF OK	
1118	044442	104000			HLT			IANS2 NOT EQUAL TO 000000	
1119									
1120	044444	122767	000027	173326	EN027:	CMPB	#27,	ICNT	ICHECK THE TEST NUMBER
1121	044452	001401			BEG	,+4		IBRANCH IF OK	
1122	044454	104000			HLT			IWRONG TEST! PC MUST HAVE FOULED UP.	
1123									
1124	044456	104400						SCOPE	
1125									
1126									
1127									
1128									
1129									
1130									
1131									
1132									
1133	044460	012767	000030	173312	TSY30:	MOV	#30,	ICNT	KEEP TRACK OF TEST NUMBER
1134	044466	004507	007102		JSR	R5,	PUSHS		IPUSH 4 WORDS ONTO STACK, SET PRIORITY
1135	044472	100177	177777		,WORD	100177,177777			ISECOND OPERAND ON TOP
1136	044476	002460	123456		,WORD	002460,123456			IFIRST OPERAND ON BOTTOM
1137	044502	000015			,WORD	015			IPROCESSOR PRIORITY LEVEL
1138	044504	054236	000340		,WORD	TRAPER, 340			IFIS TRAP VECTOR
1139									
1140	044510	000240			NOP				
1141	044512	075016			FSUB+	SP			IFLOATING SUBTRACT ON THE STACK
1142									
1143	044514	004767	007114		JSR	PC,	POPS		IPOP THE ANSWER
1144	044520	026706	173256		CMP	STACK,	SP		ICHECK THE STACK POINTER
1145	044524	001401			BEG	TSY30			IBRANCH IF OK
1146	044526	016706	173250		MOV	STACK,	SP		IRESTORE STACK POINTER
1147	044532	104000			HLT				ISTACK POINTER FOULED UP
1148	044534	000417			BR	END30			ISKIP REST OF TEST
1149									
1150	044536	022767	000000	173244	TSY30:	CMP	#000,	SPSW	ICHECK PS (EXCEPT T BIT)
1151	044544	001401			BEG	,+4			IBRANCH IF OK
1152	044546	104000			HLT				IPS NOT EQUAL TO 000
1153									
1154	044550	022767	002460	173236	CMP	#002460,ANS1			ICHECK FIRST HALF OF ANSWER

```

|*****|
|TEST 30|          FSUB (KE11F FLOATING SUBTRACT INSTRUCTION)
|          002460,123456 = 100177,177777 = 002460,123456
|          PS = 000,          STACK POINTER = SP
|*****|

```



1209  
1210  
1211  
1212  
1213  
1214  
1215  
1216  
1217  
1218  
1219  
1220  
1221  
1222  
1223  
1224  
1225  
1226  
1227  
1228  
1229  
1230  
1231  
1232  
1233  
1234  
1235  
1236  
1237  
1238  
1239  
1240  
1241  
1242  
1243  
1244  
1245  
1246  
1247  
1248  
1249  
1250  
1251  
1252  
1253  
1254  
1255  
1256  
1257  
1258  
1259  
1260  
1261  
1262

```

*****
)TEST 321      FSUB (KE11F FLOATING SUBTRACT INSTRUCTION)
)      100400,000000 = 100200,000000 = 100200,000000
)      PS = 150,      STACK POINTER = R5
)*****
    
```

```

1217 044744 012767 000032 173026 TSF321 MOV #32, ICNT IKEEP TRACK OF TEST NUMBER
1218 044752 004567 006774 JSR R5, PUSHR IPUSH 4 WORDS ONTO R5 STACK, SET PRIORITY
1219 044756 100200 000000 ;WORD 100200,000000 ISECOND OPERAND ON TOP
1220 044762 100400 000000 ;WORD 100400,000000 IFIRST OPERAND ON BOTTOM
1221 044766 000140 ;WORD 140 IPROCESSOR PRIORITY LEVEL
1222 044770 054236 000340 ;WORD TRAPER, 340 IFIS TRAP VECTOR
1223 044774 012705 040040 MOV #STACK0,R5 ISET UP STACK POINTER
1224
1225 045000 000240 NOP
1226 045002 075015 FSUB+ R5 IFLOATING SUBTRACT ON THE R5 STACK
1227
1228 045004 004767 006774 JSR PC, POPR IPOP THE ANSWER
1229 045010 010567 172776 MOV R5, SSP ISAVE "STACK POINTER"
1230 045014 022767 000150 172766 CMP #150, SPSW ICHECK PS (EXCEPT 1 BIT)
1231 045022 001401 BEQ ,+4 IBRANCH IF OK
1232 045024 104000 HLT IPS NOT EQUAL TO 150
1233
1234 045026 022767 040044 172756 CMP #STACK4,SSP ICHECK THE STACK POINTER (R5)
1235 045034 001401 BEQ ,+4 IBRANCH IF OK
1236 045036 104000 HLT ISTACK POINTER (R5) NOT EQUAL TO #STACK4
1237
1238 045040 022767 100200 172746 CMP #100200,ANS1 ICHECK FIRST HALF OF ANSWER
1239 045046 001401 BEQ ,+4 IBRANCH IF OK
1240 045050 104000 HLT IANS1 NOT EQUAL TO 100200
1241
1242 045052 022767 000000 172736 CMP #000000,ANS2 ICHECK SECOND HALF OF ANSWER
1243 045060 001401 BEQ ,+4 IBRANCH IF OK
1244 045062 104000 HLT IANS2 NOT EQUAL TO 000000
1245
1246 045064 122767 000032 172706 EN0321 CMPB #32, ICNT ICHECK THE TEST NUMBER
1247 045072 001401 BEQ ,+4 IBRANCH IF OK
1248 045074 104000 HLT IWRONG TEST! PC MUST HAVE FOULED UP!
1249
1250 045076 104400 SCOPE
    
```

```

*****
)TEST 331      FSUB (KE11F FLOATING SUBTRACT INSTRUCTION)
)      000425,052525 = 000252,125252 = 000200,000000
)      PS = 000,      STACK POINTER = R4
)*****
    
```

```

1259 045100 012767 000033 172672 TSF331 MOV #33, ICNT IKEEP TRACK OF TEST NUMBER
1260 045106 004567 006640 JSR R5, PUSHR IPUSH 4 WORDS ONTO R4 STACK, SET PRIORITY
1261 045112 000252 125252 ;WORD 000252,125252 ISECOND OPERAND ON TOP
1262 045116 000425 052525 ;WORD 000425,052525 IFIRST OPERAND ON BOTTOM
    
```



1317	045310	026706	172466			CMP	STACK, SP		ICHECK THE STACK POINTER
1318	045314	001404				BEG	ISA34		IBRANCH IF OK
1319	045316	016706	172460			MOV	STACK, SP		IRESTORE THE STACK POINTER
1320	045322	104000				HLT			ISTACK POINTER FOULED UP
1321	045324	000443				BR	END34		ISKIP REST OF TEST
1322									
1323	045326	022767	000140	172454	ISA34	CMP	#140, SPSW		ICHECK PS AFTER FIS TRAP
1324	045334	001401				BEG	,+4		IBRANCH IF OK
1325	045336	104000				HLT			IPS AFTER FIS TRAP NOT EQUAL TO 140
1326									
1327	045340	022767	045270	172446		CMP	WRTA34, ANS1		ICHECK FIS TRAP RETURN ADDRESS
1328	045346	001401				BEG	,+4		IBRANCH IF OK
1329	045350	104000				HLT			IFIS TRAP AT WRONG ADDRESS
1330									
1331	045352	022767	000052	172436		CMP	#052, ANS2		ICHECK PS BEFORE FIS TRAP
1332	045360	001401				BEG	,+4		IBRANCH IF OK
1333	045362	104000				HLT			IPS AT FIS TRAP TIME NOT 052
1334									
1335	045364	022767	000252	172426		CMP	#000252, ANS3		ICHECK DATA FROM THE STACK
1336	045372	001401				BEG	,+4		IBRANCH IF OK
1337	045374	104000				HLT			IDATA ON STACK (000252) CHANGED
1338									
1339	045376	022767	125253	172416		CMP	#125253, ANS4		ICHECK DATA FROM STACK
1340	045404	001401				BEG	,+4		IBRANCH IF OK
1341	045406	104000				HLT			IDATA ON STACK (125253) CHANGED
1342									
1343	045410	022767	000425	172406		CMP	#000425, ANS5		ICHECK DATA FROM STACK
1344	045416	001401				BEG	,+4		IBRANCH IF OK
1345	045420	104000				HLT			IDATA ON STACK (000425) CHANGED
1346									
1347	045422	022767	052525	172396		CMP	#052525, ANS6		ICHECK DATA FROM STACK
1348	045430	001401				BEG	,+4		IBRANCH IF OK
1349	045432	104000				HLT			IDATA ON STACK (052525) CHANGED
1350									
1351	045434	122767	000034	172336	END34	CMPB	#34, ICNT		ICHECK THE TEST NUMBER
1352	045442	001401				BEG	,+4		IBRANCH IF OK
1353	045444	104000				HLT			WRONG TEST! PC MUST HAVE FOULED UP!
1354									
1355	045446	104400					SCOPE		
1356									
1357									
1358									
1359									
1360									
1361									
1362									
1363									
1364	045450	012767	000035	172322	TS#35	MOV	#35, ICNT		ICHECK TRACK OF TEST NUMBER
1365	045456	004567	000112			JSR	R5, PUSH5		IPUSH 4 WORDS ONTO STACK, SET PRIORITY
1366	045462	077652	125252			,WORD	077652,125252		ISECOND OPERAND ON TOP
1367	045466	177452	125252			,WORD	177452,125252		IFIRST OPERAND ON BOTTOM
1368	045472	000157				,WORD	157		IPROCESSOR PRIORITY LEVEL
1369	045474	054236	000340			,WORD	TRAPER, 340		IFIS TRAP VECTOR
1370									

```

*****
|TEST 35:          FSUB (KE11F FLOATING SUBTRACT INSTRUCTION)
|          177452,125252 = 077652,125252 = 177777,177777
|          PS = 150,          STACK POINTER = SP
*****
    
```

1371	045500	000240				NOP			
1372	045502	075016				PSUB+	SP		IFLOATING SUBTRACT ON THE STACK
1373									
1374	045504	004767	000124			JSR	PC,	POPS	IFOP THE ANSWER
1375	045510	026706	172266			CMP	STACK,	SP	ICHECK THE STACK POINTER
1376	045514	001404				BEG	TSI35		IBRANCH IF OK
1377	045516	016706	172260			MOV	STACK,	SP	IRESTORE STACK POINTER
1378	045522	104000				HLT			ISTACK POINTER FOULED UP
1379	045524	000417				BR	END35		ISRIP REST OF TEST
1380									
1381	045526	022767	000150	172254	TSI35	CMP	#150,	SP0W	ICHECK PS (EXCEPT T BIT)
1382	045534	001401				BEG	:+4		IBRANCH IF OK
1383	045536	104000				HLT			IPS NOT EQUAL TO 150
1384									
1385	045540	022767	177777	172246		CMP	#177777,ANS1		ICHECK FIRST HALF OF ANSWER
1386	045546	001401				BEG	:+4		IBRANCH IF OK
1387	045550	104000				HLT			IANS1 NOT EQUAL TO 177777
1388									
1389	045552	022767	177777	172236		CMP	#177777,ANS2		ICHECK SECOND HALF OF ANSWER
1390	045560	001401				BEG	:+4		IBRANCH IF OK
1391	045562	104000				HLT			IANS2 NOT EQUAL TO 177777
1392									
1393	045564	122767	000035	172206	END35	CMPB	#35,	ICNT	ICHECK THE TEST NUMBER
1394	045572	001401				BEG	:+4		IBRANCH IF OK
1395	045574	104000				HLT			IWRONG TEST! PC MUST HAVE FOULED UP!
1396									
1397	045576	104400							SCOPE
1398									
1399									
1400									
1401									
1402									
1403									
1404									
1405									
1406	045600	012767	000036	172172	TSI36	MOV	#36,	ICNT	ICHECK TRACK OF TEST NUMBER
1407	045606	004567	000140			JSR	R3,	PUSHR	IPUSH 4 WORDS ONTO R3 STACK, SET PRIORITY
1408	045612	177492	125252			,WORD	177492,125252		ISECOND OPERAND ON TOP
1409	045616	077652	125253			,WORD	077652,125253		IFIRST OPERAND ON BOTTOM
1410	045622	000015				,WORD	015		IPROCESSOR PRIORITY LEVEL
1411	045624	049654	000144			,WORD	ISR36, 144		IFIS TRAP VECTOR
1412	045630	012703	040040			MOV	#STACK0,R3		ISET UP R3 AS STACK POINTER
1413									
1414	045634	000240				NOP			
1415	045636	075013				PSUB+	R3		IFLOATING SUBTRACT ON THE R3 STACK
1416									
1417	045640	004767	000140			JSR	X7,	POPR	IFOP THE "ANSWER"
1418	045644	010367	172142			MOV	R3,	SSP	ISAVE STACK POINTER (R3)
1419	045650	104000				HLT			IFIS TRAP DIDN'T OCCURE!
1420	045652	000454				BR	END36		
1421									
1422	045654	004767	000156			JSR	X7,	POPR	IFOP ALL DATA OFF THE STACKS
1423	045660	010367	172126			MOV	R3,	SSP	ISAVE STACK POINTER (R3)
1424	045664	022767	000144	172116		CMP	#144,	SP0W	ICHECK PS AFTER FIS TRAP

```

*****
I TEST 36: PSUB (KE11F FLOATING SUBTRACT INSTRUCTION)
I 077652,125253 = 177452,125252 ==> OVERFLOW
I PS(ON STACK) = 002, STACK POINTER = R3
*****
    
```





```

1462
1463
1464
1465
1466
1467
1468
1469 040020 012767 000037 171752 TS371 MOV #37, ICNT ;KEEP TRACK OF TEST NUMBER
1470 040026 004567 005720 JSR R5, PUSHR ;PUSH 4 WORDS ONTO R3 STACK, SET PRIORITY
1471 040032 043125 052525 ;WORD 043125,052525 ;SECOND OPERAND ON TOP
1472 040036 035152 125252 ;WORD 035152,125252 ;FIRST OPERAND ON BOTTOM
1473 040042 000147 ;WORD 147 ;PROCESSOR PRIORITY LEVEL
1474 040044 054236 000340 ;WORD TRAPER, 340 ;FIS TRAP VECTOR
1475 040050 012703 040040 MOV #STACK0,R3 ;SET UP STACK POINTER
1476
1477 040054 000240 NOP
1478 040056 075013 FSUB+ R3 ;FLOATING SUBTRACT ON THE R3 STACK
1479
1480 040060 004767 005720 JSR PC, POBR ;POP THE ANSWER
1481 040064 010367 171722 MOV R3, SSP ;SAVE "STACK POINTER"
1482 040070 022767 000150 171712 CMP #150, SPSW ;CHECK PS (EXCEPT 7 BIT)
1483 040076 001401 BEQ .+4 ;BRANCH IF OK
1484 040100 104000 HLT ;PS NOT EQUAL TO 150
1485
1486 040102 022767 040044 171702 CMP #STACK4,SSP ;CHECK THE STACK POINTER (R3)
1487 040110 001401 BEQ .+4 ;BRANCH IF OK
1488 040112 104000 HLT ;STACK POINTER (R3) NOT EQUAL TO #STACK4
1489
1490 040114 022767 143125 171672 CMP #143125,ANS1 ;CHECK FIRST HALF OF ANSWER
1491 040122 001401 BEQ .+4 ;BRANCH IF OK
1492 040124 104000 HLT ;ANS1 NOT EQUAL TO 143125
1493
1494 040126 022767 052524 171662 CMP #052524,ANS2 ;CHECK SECOND HALF OF ANSWER
1495 040134 001401 BEQ .+4 ;BRANCH IF OK
1496 040136 104000 HLT ;ANS2 NOT EQUAL TO 052524
1497
1498 040140 122767 000037 171632 EN0371 CMPB #37, ICNT ;CHECK THE TEST NUMBER
1499 040146 001401 BEQ .+4 ;BRANCH IF OK
1500 040150 104000 HLT ;WRONG TEST! PC MUST HAVE FOULED UP!
1501
1502 040152 104400 SCOPE
1503
1504
1505
1506
1507
1508
1509
1510
1511 040154 012767 000040 171616 TS3401 MOV #40, ICNT ;KEEP TRACK OF TEST NUMBER
1512 040162 004567 005364 JSR R5, PUSHR ;PUSH 4 WORDS ONTO R0 STACK, SET PRIORITY
1513 040166 130192 129252 ;WORD 130192,129252 ;SECOND OPERAND ON TOP
1514 040172 143325 052525 ;WORD 143325,052525 ;FIRST OPERAND ON BOTTOM
1515 040176 000043 ;WORD 043 ;PROCESSOR PRIORITY LEVEL

```

1516	046200	054236	000340			WORD	TRAPER, 340	IF IS TRAP VECTOR
1517	046204	012700	040040			MOV	#STACK0,R0	IF SET UP STACK POINTER
1518								
1519	046210	000240				NOP		
1520	046212	075010				FSUB+	R0	IF FLOATING SUBTRACT ON THE R0 STACK
1521								
1522	046214	004767	005564			JSR	PC, POPR	IF POP THE ANSWER
1523	046220	010067	171566			MOV	R0, SSP	IF SAVE "STACK POINTER"
1524	046224	022767	000050	171556		CMP	#050, SPSW	IF CHECK PS (EXCEPT T BIT)
1525	046232	001401				BEQ	,+4	IF BRANCH IF OK
1526	046234	104000				HLT		IF PS NOT EQUAL TO 050
1527								
1528	046236	022767	040044	171546		CMP	#STACK4,SSP	IF CHECK THE STACK POINTER (R0)
1529	046244	001401				BEQ	,+4	IF BRANCH IF OK
1530	046246	104000				HLT		IF STACK POINTER (R0) NOT EQUAL TO #STACK4
1531								
1532	046250	022767	143325	171536		CMP	#143325,ANS1	IF CHECK FIRST HALF OF ANSWER
1533	046256	001401				BEQ	,+4	IF BRANCH IF OK
1534	046260	104000				HLT		IF ANS1 NOT EQUAL TO 143325
1535								
1536	046262	022767	052525	171526		CMP	#052525,ANS2	IF CHECK SECOND HALF OF ANSWER
1537	046270	001401				BEQ	,+4	IF BRANCH IF OK
1538	046272	104000				HLT		IF ANS2 NOT EQUAL TO 052525
1539								
1540	046274	122767	000040	171476	END401	CMPB	#40, ICNT	IF CHECK THE TEST NUMBER
1541	046302	001401				BEQ	,+4	IF BRANCH IF OK
1542	046304	104000				HLT		IF WRONG TEST! PC MUST HAVE FOULED UP!
1543								
1544	046306	104400					SCOPE	
1545								
1546								
1547								
1548								
1549								
1550								
1551								
1552								
1553	046310	012767	000041	171462	TS7411	MOV	#41, ICNT	IF KEEP TRACK OF TEST NUMBER
1554	046316	004567	005430			JSR	R5, PUSHR	IF PUSH 4 WORDS ONTO R5 STACK, SET PRIORITY
1555	046322	143325	052525			WORD	143325,052525	IF SECOND OPERAND ON TOP
1556	046326	135152	125252			WORD	135152,125252	IF FIRST OPERAND ON BOTTOM
1557	046332	000197				WORD	197	IF PROCESSOR PRIORITY LEVEL
1558	046334	054236	000340			WORD	TRAPER, 340	IF IS TRAP VECTOR
1559	046340	012700	040040			MOV	#STACK0,R5	IF SET UP STACK POINTER
1560								
1561	046344	000240				NOP		
1562	046346	075015				FSUB+	R5	IF FLOATING SUBTRACT ON THE R5 STACK
1563								
1564	046350	004767	005430			JSR	PC, POPR	IF POP THE ANSWER
1565	046354	010567	171432			MOV	R5, SSP	IF SAVE "STACK POINTER"
1566	046360	022767	000140	171422		CMP	#140, SPSW	IF CHECK PS (EXCEPT T BIT)
1567	046366	001401				BEQ	,+4	IF BRANCH IF OK
1568	046370	104000				HLT		IF PS NOT EQUAL TO 140
1569								

```

*****
TEST 411 FSUB (KE11F FLOATING SUBTRACT INSTRUCTION)
| 135152,125252 * 143325,052525 = 043325,052525
| PS = 140, STACK POINTER = R5
*****

```

1570	046372	022767	040044	171412	CMP	#STACK4,SSP	ICHECK THE STACK POINTER (R5)
1571	046400	001401			BEQ	,+4	IBRANCH IF OK
1572	046402	104000			HLT		ISTACK POINTER (R5) NOT EQUAL TO #STACK4
1573							
1574	046404	022767	043325	171402	CMP	#043325,ANS1	ICHECK FIRST HALF OF ANSWER
1575	046412	001401			BEQ	,+4	IBRANCH IF OK
1576	046414	104000			HLT		IANS1 NOT EQUAL TO 043325
1577							
1578	046416	022767	052525	171372	CMP	#052525,ANS2	ICHECK SECOND HALF OF ANSWER
1579	046424	001401			BEQ	,+4	IBRANCH IF OK
1580	046426	104000			HLT		IANS2 NOT EQUAL TO 052525
1581							
1582	046430	122767	000041	171342	EN041	#41, ICNT	ICHECK THE TEST NUMBER
1583	046436	001401			BEQ	,+4	IBRANCH IF OK
1584	046440	104000			HLT		IWRONG TEST! PC MUST HAVE FOULED UP!
1585							
1586	046442	104400			SCOPE		
1587							
1588							
1589							
1590							
1591							
1592							
1593							
1594							
1595	046444	012767	000042	171326	TS#42:	MOV #42, ICNT	ICHECK THE TEST NUMBER
1596	046452	003567	005274		JSR	R5, PUSHR	IPUSH 2 WORDS ONTO R2 STACK, SET PRIORITY
1597	046456	035152	125252		,WORD	035152,125252	ISECOND OPERAND ON TOP
1598	046462	043125	052525		,WORD	043125,052525	IFIRST OPERAND ON BOTTOM
1599	046466	000040			,WORD	040	IPROCESSOR PRIORITY LEVEL
1600	046470	054236	000340		,WORD	TRAPER, 340	IFIS TRAP VECTOR
1601	046474	012702	040040		MOV	#STACK0,R2	ISSET UP STACK POINTER
1602							
1603	046500	000240			NOP		
1604	046502	075012			PSUB+	R2	IFLOATING SUBTRACT ON THE R2 STACK
1605							
1606	046504	004767	005274		JSR	PC, POPR	IPOP THE ANSWER
1607	046510	010267	171276		MOV	R2, SSP	ISAVE "STACK POINTER"
1608	046514	022767	000040	171266	CMP	#040, SPSW	ICHECK PS (EXCEPT 7 BIT)
1609	046522	001401			BEQ	,+4	IBRANCH IF OK
1610	046524	104000			HLT		IPB NOT EQUAL TO 040
1611							
1612	046526	022767	040044	171256	CMP	#STACK4,SSP	ICHECK THE STACK POINTER (R2)
1613	046534	001401			BEQ	,+4	IBRANCH IF OK
1614	046536	104000			HLT		ISTACK POINTER (R2) NOT EQUAL TO #STACK4
1615							
1616	046540	022767	043125	171246	CMP	#043125,ANS1	ICHECK FIRST HALF OF ANSWER
1617	046546	001401			BEQ	,+4	IBRANCH IF OK
1618	046550	104000			HLT		IANS1 NOT EQUAL TO 043125
1619							
1620	046552	022767	052524	171236	CMP	#052524,ANS2	ICHECK SECOND HALF OF ANSWER
1621	046560	001401			BEQ	,+4	IBRANCH IF OK
1622	046562	104000			HLT		IANS2 NOT EQUAL TO 052524
1623							

```

*****
I TEST 42: PSUB (KE11F FLOATING SUBTRACT INSTRUCTION)
I   043125,052525 * 035152,125252 = 043125,052524
I   PS = 040, STACK POINTER = R2
*****

```

```

1624 046564 122767 000042 171206 EN042I CMPB #42, ICNT ICHECK THE TEST NUMBER
1625 046572 001401 BEQ .+4 IBRANCH IF OK
1626 046574 104000 HLT IWRONG TEST! PC MUST HAVE FOULED UP.
1627
1628 046576 104400 SCOPE
1629
1630
1631
1632
1633 I*****
1633 ITEST 43I PMUL (KE11F FLOATING MULTIPLY INSTRUCTION)
1634 I 000000,000000 * 000000,000000 = 000000,000000
1634 I PS = 104, STACK POINTER = R4
1635 I*****
1636
1637 046600 012767 000043 171172 TSF43I MOV #43, ICNT IKEEP TRACK OF TEST NUMBER
1638 046606 004567 005140 JSR R5, PUSHR IPUSH 4 WORDS ONTO R4 STACK, SET PRIORITY
1639 046612 000000 000000 ,WORD 000000,000000 ISECOND OPERAND ON TOP
1640 046616 000000 000000 ,WORD 000000,000000 IFIRST OPERAND ON BOTTOM
1641 046622 000111 ,WORD 111 IPROCESSOR PRIORITY LEVEL
1642 046624 004236 000340 ,WORD TRAPER, 340 IFIS TRAP VECTOR
1643 046630 012704 040040 MOV #STACK0,R4 ISET UP STACK POINTER
1644
1645 046634 000240 NOP
1646 046636 075024 FMUL+ R4 IFLOATING MULTIPLY ON THE R4 STACK
1647
1648 046640 004767 005140 JSR PC, POPR IPDP THE ANSWER
1649 046644 010467 171142 MOV R4, SSF ISAVE "STACK POINTER"
1650 046650 022767 000104 171132 CMP #104, SPSW ICHECK PS (EXCEPT 7 BIT)
1651 046656 001401 BEQ .+4 IBRANCH IF OK
1652 046660 104000 HLT IPS NOT EQUAL TO 104
1653
1654 046662 022767 040044 171122 CMP #STACK4,SSF ICHECK THE STACK POINTER (R4)
1655 046670 001401 BEQ .+4 IBRANCH IF OK
1656 046672 104000 HLT ISTACK POINTER (R4) NOT EQUAL TO #STACK4
1657
1658 046674 022767 000000 171112 CMP #000000,ANS1 ICHECK FIRST HALF OF ANSWER
1659 046702 001401 BEQ .+4 IBRANCH IF OK
1660 046704 104000 HLT IANS1 NOT EQUAL TO 000000
1661
1662 046706 022767 000000 171102 CMP #000000,ANS2 ICHECK SECOND HALF OF ANSWER
1663 046714 001401 BEQ .+4 IBRANCH IF OK
1664 046716 104000 HLT IANS2 NOT EQUAL TO 000000
1665
1666 046720 122767 000043 171052 EN043I CMPB #43, ICNT ICHECK THE TEST NUMBER
1667 046726 001401 BEQ .+4 IBRANCH IF OK
1668 046730 104000 HLT IWRONG TEST! PC MUST HAVE FOULED UP.
1669
1670 046732 104400 SCOPE
1671
1672
1673 I*****
1674 ITEST 44I PMUL (KE11F FLOATING MULTIPLY INSTRUCTION)
1675 I 140200,000000 * 052345,123456 = 152345,123456
1676 I PS = 150, STACK POINTER = R2
1677 I*****

```

1678										
1679	046734	012767	000044	171036	TSF44I	MOV	#44,	ICNT		KEEP TRACK OF TEST NUMBER
1680	046742	004567	005004			JSR	R5,	PUSHR		PUSH 4 WORDS ONTO R2 STACK, SET PRIORITY
1681	046746	052345	123456			,WORD	052345,123456			SECOND OPERAND ON TOP
1682	046752	140200	000000			,WORD	140200,000000			FIRST OPERAND ON BOTTOM
1683	046756	000143				,WORD	143			PROCESSOR PRIORITY LEVEL
1684	046760	054236	000340			,WORD	TRAPER, 340			FIS TRAP VECTOR
1685	046764	012702	040040			MOV	#STACK0,R2			SET UP STACK POINTER
1686										
1687	046770	000240				NOP				
1688	046772	075022				FMUL+	R2			FLOATING MULTIPLY ON THE R2 STACK
1689										
1690	046774	004767	005004			JSR	PC,	POPR		POP THE ANSWER
1691	047000	010267	171006			MOV	R2,	SSP		SAVE "STACK POINTER"
1692	047004	022767	000150	170776		CMP	#150,	SPSW		CHECK PS (EXCEPT 7 BIT)
1693	047012	001401				BEG	,+4			BRANCH IF OK
1694	047014	104000				HLT				PS NOT EQUAL TO 150
1695										
1696	047016	022767	040044	170766		CMP	#STACK4,SSP			CHECK THE STACK POINTER (R2)
1697	047024	001401				BEG	,+4			BRANCH IF OK
1698	047026	104000				HLT				STACK POINTER (R2) NOT EQUAL TO #STACK4
1699										
1700	047030	022767	152345	170756		CMP	#152345,ANS1			CHECK FIRST HALF OF ANSWER
1701	047036	001401				BEG	,+4			BRANCH IF OK
1702	047040	104000				HLT				ANS1 NOT EQUAL TO 152345
1703										
1704	047042	022767	123456	170746		CMP	#123456,ANS2			CHECK SECOND HALF OF ANSWER
1705	047050	001401				BEG	,+4			BRANCH IF OK
1706	047052	104000				HLT				ANS2 NOT EQUAL TO 123456
1707										
1708	047054	122767	000044	170716	END44I	CMPB	#44,	ICNT		CHECK THE TEST NUMBER
1709	047062	001401				BEG	,+4			BRANCH IF OK
1710	047064	104000				HLT				WRONG TEST! PC MUST HAVE FOULED UP!
1711										
1712	047066	104400								SCOPE
1713										
1714										
1715										
1716										
1717										
1718										
1719										
1720										
1721	047070	012767	000045	170702	TSF45I	MOV	#45,	ICNT		KEEP TRACK OF TEST NUMBER
1722	047076	004567	004650			JSR	R5,	PUSHR		PUSH 4 WORDS ONTO R5 STACK, SET PRIORITY
1723	047102	139753	024642			,WORD	139753,024642			SECOND OPERAND ON TOP
1724	047106	100100	058525			,WORD	100125,052525			FIRST OPERAND ON BOTTOM
1725	047112	000117				,WORD	117			PROCESSOR PRIORITY LEVEL
1726	047114	054236	000340			,WORD	TRAPER, 340			FIS TRAP VECTOR
1727	047120	012702	040040			MOV	#STACK0,R5			SET UP STACK POINTER
1728										
1729	047124	000240				NOP				
1730	047126	075025				FMUL+	R5			FLOATING MULTIPLY ON THE R5 STACK
1731										

```

|*****
|TEST 45)      FMUL (KE11F FLOATING MULTIPLY INSTRUCTION)
|      100125,052525 * 139753,024642 = 000000,000000
|      PS = 104,      STACK POINTER = R5
|*****

```

1732	047130	004767	004650		JSR	PC,	POPR	IPOP THE ANSWER
1733	047134	010567	170652		MOV	R5,	SSP	ISAVE "STACK POINTER"
1734	047140	022767	000104	170642	CMP	#104,	SPSW	ICHECK PS (EXCEPT T BIT)
1735	047146	001401			BEG	,+4		IBRANCH IF OK
1736	047150	104000			HLT			IPS NOT EQUAL TO 104
1737								
1738	047152	022767	040044	170632	CMP	#STACK4,SSP		ICHECK THE STACK POINTER (R5)
1739	047160	001401			BEG	,+4		IBRANCH IF OK
1740	047162	104000			HLT			ISSTACK POINTER (R5) NOT EQUAL TO #STACK4
1741								
1742	047164	022767	000000	170622	CMP	#000000,ANS1		ICHECK FIRST HALF OF ANSWER
1743	047172	001401			BEG	,+4		IBRANCH IF OK
1744	047174	104000			HLT			IANS1 NOT EQUAL TO 000000
1745								
1746	047176	022767	000000	170612	CMP	#000000,ANS2		ICHECK SECOND HALF OF ANSWER
1747	047204	001401			BEG	,+4		IBRANCH IF OK
1748	047206	104000			HLT			IANS2 NOT EQUAL TO 000000
1749								
1750	047210	122767	000045	170562	EN045I	CMPB	#45,	ICNT
1751	047216	001401			BEG	,+4		ICHECK THE TEST NUMBER
1752	047220	104000			HLT			IBRANCH IF OK
1753								IWRONG TEST! PC MUST HAVE FOULED UP!
1754	047222	104400						
1755								
1756								
1757								
1758								
1759								
1760								
1761								
1762								
1763	047224	012767	000046	170546	TS#46I	MOV	#46,	ICNT
1764	047232	004567	004514		JSR	R5,	PUSHR	KEEP TRACK OF TEST NUMBER
1765	047236	000052	125252		,WORD	000052,125252		IPUSH 4 WORDS ONTO R3 STACK, SET PRIORITY
1766	047242	161616	161616		,WORD	161616,161616		ISECOND OPERAND ON TOP
1767	047246	000017			,WORD	017		IFIRST OPERAND ON BOTTOM
1768	047250	054236	000340		,WORD	TRAPER, 340		IPROCESSOR PRIORITY LEVEL
1769	047254	012703	040040		MOV	#STACK0,R3		IFIS TRAP VECTOR
1770								ISET UP STACK POINTER
1771	047260	000240			NOP			
1772	047262	070023			FMUL+	R3		IFLOATING MULTIPLY ON THE R3 STACK
1773								
1774	047264	004767	004514		JSR	PC,	POPR	IPOP THE ANSWER
1775	047270	010367	170516		MOV	R3,	SSP	ISAVE "STACK POINTER"
1776	047274	022767	000004	170506	CMP	#004,	SPSW	ICHECK PS (EXCEPT T BIT)
1777	047302	001401			BEG	,+4		IBRANCH IF OK
1778	047304	104000			HLT			IPS NOT EQUAL TO 004
1779								
1780	047306	022767	040044	170476	CMP	#STACK4,SSP		ICHECK THE STACK POINTER (R3)
1781	047314	001401			BEG	,+4		IBRANCH IF OK
1782	047316	104000			HLT			ISSTACK POINTER (R3) NOT EQUAL TO #STACK4
1783								
1784	047320	022767	000000	170466	CMP	#000000,ANS1		ICHECK FIRST HALF OF ANSWER
1785	047326	001401			BEG	,+4		IBRANCH IF OK

```

*****
TEST 46)          FMUL (KE11F FLOATING MULTIPLY INSTRUCTION)
:          161616,161616 * 000052,125252 = 000000,000000
:          PS = 004,          STACK POINTER = R3
*****

```

1786	047330	104000			HLT			ANS1 NOT EQUAL TO 000000
1787								
1788	047332	022767	000000	170496	CMP	#000000,ANS2		ICHECK SECOND HALF OF ANSWER
1789	047340	001401			BEG	,+4		IBRANCH IF OK
1790	047342	104000			HLT			ANS2 NOT EQUAL TO 000000
1791								
1792	047344	122767	000046	170426	ENB46:	CMPB	#46, ICNT	ICHECK THE TEST NUMBER
1793	047352	001401			BEG	,+4		IBRANCH IF OK
1794	047354	104000			HLT			WRONG TEST! PC MUST HAVE FOULED UP!
1795								
1796	047356	104400			SCOPE			
1797								
1798								
1799								
1800								
1801								
1802								
1803								
1804								
1805	047360	012767	000047	170412	TSF47:	MOV	#47, ICNT	ICHECK TRACK OF TEST NUMBER
1806	047366	004567	004202		JSR	R5, PUSH5		IPUSH 5 WORDS ONTO STACK, SET PRIORITY
1807	047372	041500	000000		,WORD	041500,000000		ISECOND OPERAND ON TOP
1808	047376	176452	125252		,WORD	176452,125252		IFIRST OPERAND ON BOTTOM
1809	047402	000157			,WORD	157		IPROCESSOR PRIORITY LEVEL
1810	047404	054236	000340		,WORD	TRAPER, 340		IFIS TRAP VECTOR
1811								
1812	047410	000240			NOP			
1813	047412	075026			FMUL+	SP		IFLOATING MULTIPLY ON THE STACK
1814								
1815	047414	004767	004214		JSR	PC, PONS		IPOP THE ANSWER
1816	047420	026706	170356		CMP	STACK, SP		ICHECK THE STACK POINTER
1817	047424	001404			BEG	TS47		IBRANCH IF OK
1818	047426	016706	170350		MOV	STACK, SP		IRESTORE STACK POINTER
1819	047432	104000			HLT			ISTACK POINTER FOULED UP
1820	047434	000417			BR	END47		ISRIP REST OF TEST
1821								
1822	047436	022767	000150	170344	TSI47:	CMP	#150, SPSW	ICHECK PS (EXCEPT 7 01F)
1823	047444	001401			BEG	,+4		IBRANCH IF OK
1824	047446	104000			HLT			IPS NOT EQUAL TO 150
1825								
1826	047450	022767	177777	170336	CMP	#177777,ANS1		ICHECK FIRST HALF OF ANSWER
1827	047456	001401			BEG	,+4		IBRANCH IF OK
1828	047460	104000			HLT			ANS1 NOT EQUAL TO 177777
1829								
1830	047462	022767	177777	170326	CMP	#177777,ANS2		ICHECK SECOND HALF OF ANSWER
1831	047470	001401			BEG	,+4		IBRANCH IF OK
1832	047472	104000			HLT			ANS2 NOT EQUAL TO 177777
1833								
1834	047474	122767	000047	170276	ENB49:	CMPB	#47, ICNT	ICHECK THE TEST NUMBER
1835	047502	001401			BEG	,+4		IBRANCH IF OK
1836	047504	104000			HLT			WRONG TEST! PC MUST HAVE FOULED UP!
1837								
1838	047506	104400			SCOPE			
1839								

```

*****
|TEST 47: FMUL (KE11F FLOATING MULTIPLY INSTRUCTION)
| 176452,125252 * 041500,000000 = 177777,177777
| PS = 150, STACK POINTER = SP
*****
    
```



```

1840
1841
1842 |*****
1843 |TEST 501 FMUL (KE11F FLOATING MULTIPLY INSTRUCTION)
1844 | 076452,125252 * 041500,000001 ==> OVERFLOW
1845 | PS(ON STACK) = 102; STACK POINTER = SP
1846 |*****
1847 047510 012767 000050 170262 TS1501 MOV #50, ICNT |KEEP TRACK OF TEST NUMBER
1848 047516 004567 004052 JSR R5, PUSHS |PUSH 4 WORDS ONTO STACK, SET PRIORITY
1849 047522 041500 000001 ,WORD 041500,000001 |SECOND OPERAND ON TOP
1850 047526 076452 125252 ,WORD 076452,125252 |FIRST OPERAND ON BOTTOM
1851 047532 000105 ,WORD 105 |PROCESSOR PRIORITY LEVEL
1852 047534 047560 000157 ,WORD ISR50, 157 |FIS TRAP VECTOR
1853
1854 047540 000240 NOP
1855 047542 075026 FMUL+ SP |FLOATING MULTIPLY ON THE STACK
1856
1857 047544 004767 004064 RTA501 JSR X7, POP5 |POP THE "ANSWER"
1858 047550 104000 HLT |FIS TRAP DIDN'T OCCURE!
1859 047552 016706 170224 MOV STACK, SP |RESTORE THE STACK POINTER
1860 047556 000454 BR ENDS0
1861
1862 047560 004767 004104 ISA501 JSR X7, POP5 |POP ALL DATA OFF THE STACK
1863 047564 026706 170212 CMP STACK, SP |CHECK THE STACK POINTER
1864 047570 001404 BEQ ISA50 |BRANCH IF OK
1865 047572 016706 170204 MOV STACK, SP |RESTORE THE STACK POINTER
1866 047576 104000 HLT |STACK POINTER FOULED UP
1867 047600 000443 BR ENDS0 |SKIP REST OF TEST
1868
1869 047602 022767 000157 170200 ISA501 CMP #157, SPSW |CHECK PS AFTER FIS TRAP
1870 047610 001404 BEQ ,+4 |BRANCH IF OK
1871 047612 104000 HLT |PS AFTER FIS TRAP NOT EQUAL TO 157
1872
1873 047614 022767 047544 170172 CMP #RTA50, AN51 |CHECK FIS TRAP RETURN ADDRESS
1874 047622 001404 BEQ ,+4 |BRANCH IF OK
1875 047624 104000 HLT |FIS TRAP AT WRONG ADDRESS
1876
1877 047626 022767 000102 170162 CMP #102, AN52 |CHECK PS BEFORE FIS TRAP
1878 047634 001404 BEQ ,+4 |BRANCH IF OK
1879 047636 104000 HLT |PS AT FIS TRAP TIME NOT 102
1880
1881 047640 022767 041500 170152 CMP #041500,AN53 |CHECK DATA FROM THE STACK
1882 047646 001404 BEQ ,+4 |BRANCH IF OK
1883 047650 104000 HLT |DATA ON STACK (041500) CHANGED
1884
1885 047652 022767 000001 170142 CMP #000001,AN54 |CHECK DATA FROM STACK
1886 047660 001404 BEQ ,+4 |BRANCH IF OK
1887 047662 104000 HLT |DATA ON STACK (000001) CHANGED
1888
1889 047664 022767 076452 170132 CMP #076452,AN55 |CHECK DATA FROM STACK
1890 047672 001404 BEQ ,+4 |BRANCH IF OK
1891 047674 104000 HLT |DATA ON STACK (076452) CHANGED
1892
1893 047676 022767 125252 170122 CMP #125252,AN56 |CHECK DATA FROM STACK

```

```

1894 047704 001401          BEQ      ,+4          |BRANCH IF OK
1895 047706 104000          HLT
1896
1897 047710 122767 000050 170062 EN0501 CMPB    #50, ICNT    |CHECK THE TEST NUMBER
1898 047716 001401          BEQ      ,+4          |BRANCH IF OK
1899 047720 104000          HLT          |WRONG TEST! PC MUST HAVE FOULED UP!
1900
1901 047722 104400          SCOPE
1902
1903
1904
1905 |*****
1905 |TEST $11 FMUL (KE11F FLOATING MULTIPLY INSTRUCTION)
1906 |      124252,125252 * 114100,000001 = 000200,000000
1907 |      PS = 000,          STACK POINTER = R1
1908 |*****
1909
1910 047724 012767 000051 170046 TS9511 MOV     #51, ICNT    |KEEP TRACK OF TEST NUMBER
1911 047732 004567 004014          JSR     R5,  PUSHR   |PUSH 4 WORDS ONTO R1 STACK, SET PRIORITY
1912 047736 114100 000001          ,WORD  114100,000001 |SECOND OPERAND ON TOP
1913 047742 124252 125252          ,WORD  124252,125252 |FIRST OPERAND ON BOTTOM
1914 047746 000000          ,WORD  000          |PROCESSOR PRIORITY LEVEL
1915 047750 054236 000340          ,WORD  TRAPER, 340   |FIS TRAP VECTOR
1916 047754 012767 040040          MOV     #STACK0,R1   |SET UP STACK POINTER
1917
1918 047760 000240          NOP
1919 047762 079021          FMUL+  R1          |FLOATING MULTIPLY ON THE R1 STACK
1920
1921 047764 004767 004014          JSR     PC,  POPR    |POP THE ANSWER
1922 047770 010167 170016          MOV     R1,   SSP    |SAVE "STACK POINTER"
1923 047774 022767 000000 170006          CMP     #000,  SPSW   |CHECK PS (EXCEPT 7 BIT)
1924 050002 001401          BEQ     ,+4          |BRANCH IF OK
1925 050004 104000          HLT          |PS NOT EQUAL TO 000
1926
1927 050006 022767 040044 167776          CMP     #STACK4,SSP  |CHECK THE STACK POINTER (R1)
1928 050014 001401          BEQ     ,+4          |BRANCH IF OK
1929 050016 104000          HLT          |STACK POINTER (R1) NOT EQUAL TO #STACK4
1930
1931 050020 022767 000200 167766          CMP     #000200,ANS1 |CHECK FIRST HALF OF ANSWER
1932 050026 001401          BEQ     ,+4          |BRANCH IF OK
1933 050030 104000          HLT          |ANS1 NOT EQUAL TO 000200
1934
1935 050032 022767 000000 167756          CMP     #000000,ANS2 |CHECK SECOND HALF OF ANSWER
1936 050040 001401          BEQ     ,+4          |BRANCH IF OK
1937 050042 104000          HLT          |ANS2 NOT EQUAL TO 000000
1938
1939 050044 122767 000051 167726 EN0511 CMPB    #51, ICNT    |CHECK THE TEST NUMBER
1940 050052 001401          BEQ     ,+4          |BRANCH IF OK
1941 050054 104000          HLT          |WRONG TEST! PC MUST HAVE FOULED UP!
1942
1943 050056 104400          SCOPE
1944
1945
1946
1947 |*****
1947 |TEST $21 FMUL (KE11F FLOATING MULTIPLY INSTRUCTION)
    
```

```
1948 | 024252,125252 * 114100,000000 => UNDERFLOW  
1949 | PS(ON STACK) = 112; STACK POINTER = R0  
1950 | *****  
1951 |  
1952 | 050060 012767 000052 167712 TS1521 MOV #52, ICNT | IKEEP TRACK OF TEST NUMBER  
1953 | 050066 004567 003660 JSR R5, PUSHR | IPUSH 4 WORDS ONTO R0 STACK, SET PRIORITY  
1954 | 050072 114100 000000 ;WORD 114100,000000 | ISECOND OPERAND ON TOP  
1955 | 050076 024252 125252 ;WORD 024252,125252 | IFIRST OPERAND ON BOTTOM  
1956 | 050102 000105 ;WORD 105 | IPROCESSOR PRIORITY LEVEL  
1957 | 050104 050134 000057 ;WORD ISR52, 057 | IFIS TRAP VECTOR  
1958 | 050110 012700 040040 MOV #STACK0,R0 | ISET UP R0 AS STACK POINTER  
1959 |  
1960 | 050114 000240 NOP |  
1961 | 050116 079020 FMUL+ R0 | IFLOATING MULTIPLY ON THE R0 STACK  
1962 |  
1963 | 050120 004767 003660 RTA521 JSR X7, POPR | IPOP THE "ANSWER"  
1964 | 050124 010067 167662 MOV R0, SS# | ISAVE STACK POINTER (R0)  
1965 | 050130 104000 HLT | IFIS TRAP DIDN'T OCCURE!  
1966 | 050132 000454 BR END52 |  
1967 |  
1968 | 050134 004767 003676 ISR521 JSR X7, POWER | IPOP ALL DATA OFF THE STACKS  
1969 | 050140 010067 167646 MOV R0, SS# | ISAVE STACK POINTER (R0)  
1970 | 050144 022767 000057 167636 CMP #057, SP#W | ICHECK PS AFTER FIS TRAP  
1971 | 050152 001401 BEQ +4 | IBRANCH IF OK  
1972 | 050154 104000 HLT | IPS AFTER FIS TRAP NOT EQUAL TO 057  
1973 |  
1974 | 050156 022767 040040 167626 CMP #STACK0,SS# | ICHECK THE STACK POINTER (R0)  
1975 | 050164 001401 BEQ +4 | IBRANCH IF OK  
1976 | 050166 104000 HLT | ISTACK POINTER (R0) NOT EQUAL TO #STACK0  
1977 |  
1978 | 050170 022767 050120 167616 CMP #RTA52, AN#1 | ICHECK FIS TRAP RETURN ADDRESS  
1979 | 050176 001401 BEQ +4 | IBRANCH IF OK  
1980 | 050200 104000 HLT | IFIS TRAP AT WRONG ADDRESS  
1981 |  
1982 | 050202 022767 000112 167606 CMP #112, AN#2 | ICHECK PS BEFORE FIS TRAP  
1983 | 050210 001401 BEQ +4 | IBRANCH IF OK  
1984 | 050212 104000 HLT | IPS AT FIS TRAP TIME NOT 112  
1985 |  
1986 | 050214 022767 114100 167576 CMP #114100,AN#3 | ICHECK DATA FROM THE STACK  
1987 | 050222 001401 BEQ +4 | IBRANCH IF OK  
1988 | 050224 104000 HLT | IDATA ON STACK (114100) CHANGED  
1989 |  
1990 | 050226 022767 000000 167566 CMP #000000,AN#4 | ICHECK DATA FROM STACK  
1991 | 050234 001401 BEQ +4 | IBRANCH IF OK  
1992 | 050236 104000 HLT | IDATA ON STACK (000000) CHANGED  
1993 |  
1994 | 050240 022767 024252 167556 CMP #024252,AN#5 | ICHECK DATA FROM STACK  
1995 | 050246 001401 BEQ +4 | IBRANCH IF OK  
1996 | 050250 104000 HLT | IDATA ON STACK (024252) CHANGED  
1997 |  
1998 | 050252 022767 125252 167546 CMP #125252,AN#6 | ICHECK DATA FROM STACK  
1999 | 050260 001401 BEQ +4 | IBRANCH IF OK  
2000 | 050262 104000 HLT | IDATA ON STACK (125252) CHANGED  
2001 |
```

```

2002 050264 122767 000052 167506 EN0521 CMPB #52, ICNT ICHECK THE TEST NUMBER
2003 050272 001401 BEQ ,+4 IBRANCH IF OK
2004 050274 104000 HLT IWRONG TEST! PC MUST HAVE FOULED UP!
2005
2006 050276 104400 SCOPE
2007
2008
2009
2010 ;*****
2011 ;TEST 93: PDIV (KE11F FLOATING DIVIDE INSTRUCTION)
2012 ; 000000,000000 / 140670,123456 = 000000,000000
2013 ; PS = 104, STACK POINTER = R3
2014 ;*****
2015 050300 012767 000053 167472 TS953: MOV #53, ICNT IKEEP TRACK OF TEST NUMBER
2016 050306 004567 003440 JSR R5, PUSHR IPUSH 4 WORDS ONTO R5 STACK, SET PRIORITY
2017 050312 140670 123456 ,WORD 140670,123456 ISECOND OPERAND ON TOP
2018 050316 000000 000000 ,WORD 000000,000000 IFIRST OPERAND ON BOTTOM
2019 050322 000105 ,WORD 105 IPROCESSOR PRIORITY LEVEL
2020 050324 054236 000340 ,WORD TRAPER, 340 IFIS TRAP VECTOR
2021 050330 012703 040040 MOV #STACK0,R3 ISET UP STACK POINTER
2022
2023 050334 000240 NOP
2024 050336 079033 PDIV+ R3 IFLOATING DIVIDE ON THE R5 STACK
2025
2026 050340 004767 003440 JSR PC, POPR IPOP THE ANSWER
2027 050344 010367 167442 MOV R3, SSP ISAVE "STACK POINTER"
2028 050350 022767 000104 167432 CMP #104, SPSW ICHECK PS (EXCEPT 7 BIT)
2029 050356 001401 BEQ ,+4 IBRANCH IF OK
2030 050360 104000 HLT IPS NOT EQUAL TO 104
2031
2032 050362 022767 040044 167422 CMP #STACK4,SSP ICHECK THE STACK POINTER (R3)
2033 050370 001401 BEQ ,+4 IBRANCH IF OK
2034 050372 104000 HLT ISTACK POINTER (R3) NOT EQUAL TO #STACK4
2035
2036 050374 022767 000000 167412 CMP #000000,ANS1 ICHECK FIRST HALF OF ANSWER
2037 050402 001401 BEQ ,+4 IBRANCH IF OK
2038 050404 104000 HLT IANS1 NOT EQUAL TO 000000
2039
2040 050406 022767 000000 167402 CMP #000000,ANS2 ICHECK SECOND HALF OF ANSWER
2041 050414 001401 BEQ ,+4 IBRANCH IF OK
2042 050416 104000 HLT IANS2 NOT EQUAL TO 000000
2043
2044 050420 122767 000053 167392 EN053: CMPB #53, ICNT ICHECK THE TEST NUMBER
2045 050426 001401 BEQ ,+4 IBRANCH IF OK
2046 050430 104000 HLT IWRONG TEST! PC MUST HAVE FOULED UP!
2047
2048 050432 104400 SCOPE
2049
2050
2051 ;*****
2052 ;TEST 94: PDIV (KE11F FLOATING DIVIDE INSTRUCTION)
2053 ; 050525,050529 / 100120,125552 ==> DIVIDE BY ZERO
2054 ; PS(ON STACK) = 053, STACK POINTER = R5
2055 ;*****

```

2056										
2057	050434	012767	000054	167336	TST54	MOV	#54,	ICNT		KEEP TRACK OF TEST NUMBER
2058	050442	004567	003304			JSR	R5,	PUSHR		PUSH 4 WORDS ONTO R5 STACK, SET PRIORITY
2059	050446	100125	125252			,WORD	100125,125252			SECOND OPERAND ON TOP
2060	050452	052525	052525			,WORD	052525,052525			FIRST OPERAND ON BOTTOM
2061	050456	000047				,WORD	047			PROCESSOR PRIORITY LEVEL
2062	050460	050510	000113			,WORD	ISR54, 113			FIS TRAP VECTOR
2063	050464	012705	040040			MOV	#STACK0,R5			SET UP R5 AS STACK POINTER
2064										
2065	050470	000240				NOF				
2066	050472	075035				PDIV+	R5			FLOATING DIVIDE ON THE R5 STACK
2067										
2068	050474	004767	003304		RTA54	JSR	X7,	POPR		POP THE "ANSWER"
2069	050500	010567	167306			MOV	R5,	SSP		SAVE STACK POINTER (R5)
2070	050504	104000				HLT				FIS TRAP DIDN'T OCCURE!
2071	050506	000454				BR	END54			
2072										
2073	050510	004767	003322		ISR54	JSR	X7,	POPR		POP ALL DATA OFF THE STACKS
2074	050514	010567	167272			MOV	R5,	SSP		SAVE STACK POINTER (R5)
2075	050520	022767	000113	167262		CMP	#113,	SPSW		CHECK PS AFTER FIS TRAP
2076	050526	001401				BEG	,+4			BRANCH IF OK
2077	050530	104000				HLT				PS AFTER FIS TRAP NOT EQUAL TO 113
2078										
2079	050532	022767	040040	167252		CMP	#STACK0,SSP			CHECK THE STACK POINTER (R5)
2080	050540	001401				BEG	,+4			BRANCH IF OK
2081	050542	104000				HLT				STACK POINTER (R5) NOT EQUAL TO #STACK0
2082										
2083	050544	022767	050474	167242		CMP	#RTA54,ANS1			CHECK FIS TRAP RETURN ADDRESS
2084	050552	001401				BEG	,+4			BRANCH IF OK
2085	050554	104000				HLT				FIS TRAP AT WRONG ADDRESS
2086										
2087	050556	022767	000053	167232		CMP	#053,ANS2			CHECK PS BEFORE FIS TRAP
2088	050564	001401				BEG	,+4			BRANCH IF OK
2089	050566	104000				HLT				PS AT FIS TRAP TIME NOT 053
2090										
2091	050570	022767	100125	167222		CMP	#100125,ANS3			CHECK DATA FROM THE STACK
2092	050576	001401				BEG	,+4			BRANCH IF OK
2093	050600	104000				HLT				DATA ON STACK (100125) CHANGED
2094										
2095	050602	022767	125252	167212		CMP	#125252,ANS4			CHECK DATA FROM STACK
2096	050610	001401				BEG	,+4			BRANCH IF OK
2097	050612	104000				HLT				DATA ON STACK (125252) CHANGED
2098										
2099	050614	022767	052525	167202		CMP	#052525,ANS5			CHECK DATA FROM STACK
2100	050622	001401				BEG	,+4			BRANCH IF OK
2101	050624	104000				HLT				DATA ON STACK (052525) CHANGED
2102										
2103	050626	022767	052525	167172		CMP	#052525,ANS6			CHECK DATA FROM STACK
2104	050634	001401				BEG	,+4			BRANCH IF OK
2105	050636	104000				HLT				DATA ON STACK (052525) CHANGED
2106										
2107	050640	122767	000054	167132	END54	CMPB	#54,	ICNT		CHECK THE TEST NUMBER
2108	050646	001401				BEG	,+4			BRANCH IF OK
2109	050650	104000				HLT				WRONG TEST! PC MUST HAVE FOULED UP!

```

2110
2111 050652 104400          SCOPE
2112
2113
2114
2115 |*****
|TEST 55)      FDIV (KE11P FLOATING DIVIDE INSTRUCTION)
2116 |      125252,125252 / 140200,000000 = 025252,125252
2117 |      PS = 000,      STACK POINTER = SP
2118 |*****
2119
2120 050654 012767 000055 167116 TS#55) MOV      #55,      ICNT      |KEEP TRACK OF TEST NUMBER
2121 050662 004567 002706          JBR      R5,      PUSH5     |PUSH 4 WORDS ONTO STACK, SET PRIORITY
2122 050666 140200 000000          ,WORD    140200,000000     |SECOND OPERAND ON TOP
2123 050672 125252 125252          ,WORD    125252,125252     |FIRST OPERAND ON BOTTOM
2124 050676 000017          ,WORD    017              |PROCESSOR PRIORITY LEVEL
2125 050700 054236 000340          ,WORD    TRAPER, 340      |PIS TRAP VECTOR
2126
2127 050704 000240          NOP
2128 050706 075036          FDIV+   SP              |FLOATING DIVIDE ON THE STACK
2129
2130 050710 004767 002720          JBR      PC,      POFS     |POP THE ANSWER
2131 050714 026706 167062          CMP      STACK,   SP      |CHECK THE STACK POINTER
2132 050720 001404          BEQ      TS#55         |BRANCH IF OK
2133 050722 016706 167054          MOV      STACK,   SP      |RESTORE STACK POINTER
2134 050726 104000          HLT
2135 050730 000417          BR       END55         |STRIP REST OF TEST
2136
2137 050732 022767 000000 167050 TS#55) CMP      #000,    SP#W     |CHECK PS (EXCEPT # BIT)
2138 050740 001401          BEQ      ,+4          |BRANCH IF OK
2139 050742 104000          HLT                    |PS NOT EQUAL TO 000
2140
2141 050744 022767 025252 167042          CMP      #025252,ANS1    |CHECK FIRST HALF OF ANSWER
2142 050752 001401          BEQ      ,+4          |BRANCH IF OK
2143 050754 104000          HLT                    |ANS1 NOT EQUAL TO 025252
2144
2145 050756 022767 125252 167032          CMP      #125252,ANS2    |CHECK SECOND HALF OF ANSWER
2146 050764 001401          BEQ      ,+4          |BRANCH IF OK
2147 050766 104000          HLT                    |ANS2 NOT EQUAL TO 125252
2148
2149 050770 122767 000055 167002 EN#55) CMPB     #55,      ICNT      |CHECK THE TEST NUMBER
2150 050776 001401          BEQ      ,+4          |BRANCH IF OK
2151 051000 104000          HLT                    |WRONG TEST! PC MUST HAVE FOULED UP.
2152
2153 051002 104400          SCOPE
2154
2155
2156 |*****
2157 |TEST 56)      FDIV (KE11P FLOATING DIVIDE INSTRUCTION)
2158 |      100002,092929 / 000006,123456 ==> DIVIDE BY ZERO
2159 |      PS(ON STACK) = 153,      STACK POINTER = SP
2160 |*****
2161
2162 051004 012767 000056 166766 TS#56) MOV      #56,      ICNT      |KEEP TRACK OF TEST NUMBER
2163 051012 004567 002556          JBR      R5,      PUSH5     |PUSH 4 WORDS ONTO STACK, SET PRIORITY

```



```

2218
2219
2220 |*****
2221 |TEST 571      FDIV (KE11F FLOATING DIVIDE INSTRUCTION)
2222 |      167452,125251 / 127652,125252 = 077777,177776
2223 |      PS = 100,      STACK POINTER = R0
2224 |*****
2225 051220 012767 000057 166552 TS957i MOV #57, ICNT |KEEP TRACK OF TEST NUMBER
2226 051226 004567 002520 JSR R3, PUSHR |PUSH 4 WORDS ONTO R0 STACK, SET PRIORITY
2227 051232 127652 125252 ,WORD 127652,125252 |SECOND OPERAND ON TOP
2228 051236 167452 125251 ,WORD 167452,125251 |FIRST OPERAND ON BOTTOM
2229 051242 000111 ,WORD 111 |PROCESSOR PRIORITY LEVEL
2230 051244 054236 000340 ,WORD TRAPER, 340 |FIS TRAP VECTOR
2231 051250 012700 040040 MOV #STACK0,R0 |SET UP STACK POINTER
2232
2233 051254 000240 NOP
2234 051256 075030 FDIV+ R0 |FLOATING DIVIDE ON THE R0 STACK
2235
2236 051260 004767 002520 JSR PC, POPR |POP THE ANSWER
2237 051264 010007 164522 MOV R0, SSP |SAVE "STACK POINTER"
2238 051270 022767 000100 166512 CMP #100, SP0W |CHECK PS (EXCEPT 1 BIT)
2239 051276 001401 BEQ ,+4 |BRANCH IF OK
2240 051300 104000 HLT |PS NOT EQUAL TO 100
2241
2242 051302 022767 040044 166502 CMP #STACK4,SSP |CHECK THE STACK POINTER (R0)
2243 051310 001401 BEQ ,+4 |BRANCH IF OK
2244 051312 104000 HLT |STACK POINTER (R0) NOT EQUAL TO #STACK4
2245
2246 051314 022767 077777 166472 CMP #077777,ANS1 |CHECK FIRST HALF OF ANSWER
2247 051322 001401 BEQ ,+4 |BRANCH IF OK
2248 051324 104000 HLT |ANS1 NOT EQUAL TO 077777
2249
2250 051326 022767 177776 166462 CMP #177776,ANS2 |CHECK SECOND HALF OF ANSWER
2251 051334 001401 BEQ ,+4 |BRANCH IF OK
2252 051336 104000 HLT |ANS2 NOT EQUAL TO 177776
2253
2254 051340 122767 000057 166432 END57i CMPB #57, ICNT |CHECK THE TEST NUMBER
2255 051346 001401 BEQ ,+4 |BRANCH IF OK
2256 051350 104000 HLT |WRONG TEST! PC MUST HAVE FOULED UP!
2257
2258 051352 104400 SCOPE
2259
2260
2261 |*****
2262 |TEST 601      FDIV (KE11F FLOATING DIVIDE INSTRUCTION)
2263 |      067452,125252 / 127652,125252 => OVERFLOW
2264 |      PS(ON STACK) = 042,      STACK POINTER = R4
2265 |*****
2266
2267 051354 012767 000060 166416 TS960i MOV #60, ICNT |KEEP TRACK OF TEST NUMBER
2268 051362 004567 002364 JSR R3, PUSHR |PUSH 4 WORDS ONTO R4 STACK, SET PRIORITY
2269 051366 127652 125252 ,WORD 127652,125252 |SECOND OPERAND ON TOP
2270 051372 067452 125252 ,WORD 067452,125252 |FIRST OPERAND ON BOTTOM
2271 051376 000042 ,WORD 042 |PROCESSOR PRIORITY LEVEL

```



2272	051400	051430	000157		WORD	ISR60,	157		IF IS TRAP VECTOR
2273	051404	012704	040040		MOV	#STACK0,R4			IF SET UP R4 AS STACK POINTER
2274									
2275	051410	000240			NOP				
2276	051412	075034			FDIV+	R4			IF FLOATING DIVIDE ON THE R4 STACK
2277									
2278	051414	004767	002364	RTA60I	JSR	X7,	POPR		IF POP THE "ANSWER"
2279	051420	010467	166366		MOV	R4,	SSP		IF SAVE STACK POINTER (R4)
2280	051424	104000			HLT				IF IS TRAP DIDN'T OCCURE!
2281	051426	000454			OR	END60			
2282									
2283	051430	004767	002402	ISR60I	JSR	X7,	POPER		IF POP ALL DATA OFF THE STACKS
2284	051434	010467	166352		MOV	R4,	SSP		IF SAVE STACK POINTER (R4)
2285	051440	022767	000157	166342	CMR	#157,	SPSW		IF CHECK PS AFTER FIS TRAP
2286	051446	001401			REQ	,+4			IF BRANCH IF OK
2287	051450	104000			HLT				IF PS AFTER FIS TRAP NOT EQUAL TO 157
2288									
2289	051452	022767	040040	166332	CMR	#STACK0,SSP			IF CHECK THE STACK POINTER (R4)
2290	051460	001401			REQ	,+4			IF BRANCH IF OK
2291	051462	104000			HLT				IF STACK POINTER (R4) NOT EQUAL TO #STACK0
2292									
2293	051464	022767	051414	166322	CMR	#RTA60,ANS1			IF CHECK FIS TRAP RETURN ADDRESS
2294	051472	001401			REQ	,+4			IF BRANCH IF OK
2295	051474	104000			HLT				IF IS TRAP AT WRONG ADDRESS
2296									
2297	051476	022767	000042	166312	CMR	#042,ANS2			IF CHECK PS BEFORE FIS TRAP
2298	051504	001401			REQ	,+4			IF BRANCH IF OK
2299	051506	104000			HLT				IF PS AT FIS TRAP TIME NOT 042
2300									
2301	051510	022767	127652	166302	CMR	#127652,ANS3			IF CHECK DATA FROM THE STACK
2302	051516	001401			REQ	,+4			IF BRANCH IF OK
2303	051520	104000			HLT				IF DATA ON STACK (127652) CHANGED
2304									
2305	051522	022767	125252	166272	CMR	#125252,ANS4			IF CHECK DATA FROM STACK
2306	051530	001401			REQ	,+4			IF BRANCH IF OK
2307	051532	104000			HLT				IF DATA ON STACK (125252) CHANGED
2308									
2309	051534	022767	067452	166262	CMR	#067452,ANS5			IF CHECK DATA FROM STACK
2310	051542	001401			REQ	,+4			IF BRANCH IF OK
2311	051544	104000			HLT				IF DATA ON STACK (067452) CHANGED
2312									
2313	051546	022767	125252	166252	CMR	#125252,ANS6			IF CHECK DATA FROM STACK
2314	051554	001401			REQ	,+4			IF BRANCH IF OK
2315	051556	104000			HLT				IF DATA ON STACK (125252) CHANGED
2316									
2317	051560	122767	000060	166212	END60I	CMR	#60,ICNT		IF CHECK THE TEST NUMBER
2318	051566	001401			REQ	,+4			IF BRANCH IF OK
2319	051570	104000			HLT				IF WRONG TEST! PC MUST HAVE FOULED UP!
2320									
2321	051572	104400			SCOPE				
2322									
2323									
2324									
2325									

\*\*\*\*\*  
 TEST 611 FDIV (KE11F FLOATING DIVIDE INSTRUCTION)

```

2326 | 154200,000000 / 014377,177777 ==> OVERFLOW
2327 | PS(ON STACK) = 042, STACK POINTER = SP
2328 | *****
2329 |
2330 051574 012767 000061 166176 TST61: MOV #61, ICNT ;KEEP TRACK OF TEST NUMBER
2331 051602 004567 001766 JSR R5, PUSH5 ;PUSH 5 WORDS ONTO STACK, SET PRIORITY
2332 051606 014377 177777 ;WORD 014377,177777 ;SECOND OPERAND ON TOP
2333 051612 154200 000000 ;WORD 154200,000000 ;FIRST OPERAND ON BOTTOM
2334 051616 000046 ;WORD 046 ;PROCESSOR PRIORITY LEVEL
2335 051620 051644 000103 ;WORD ISR61, 103 ;FIS TRAP VECTOR
2336 |
2337 051624 000240 NOP
2338 051626 075036 FDIV+ SP ;FLOATING DIVIDE ON THE STACK
2339 |
2340 051630 004767 002000 RTA61: JSR X7, POPS ;POP THE "ANSWER"
2341 051634 104000 HLT ;FIS TRAP DIDN'T OCCURE!
2342 051636 016706 166140 MOV STACK, SP ;RESTORE THE STACK POINTER
2343 051642 000404 BR END61
2344 |
2345 051644 004767 002020 ISR61: JSR X7, POPES ;POP ALL DATA OFF THE STACK
2346 051650 026706 166126 CMP STACK, SP ;CHECK THE STACK POINTER
2347 051654 001404 BEQ ISA61 ;BRANCH IF OK
2348 051656 016706 166120 MOV STACK, SP ;RESTORE THE STACK POINTER
2349 051662 104000 HLT ;STACK POINTER FOULED UP
2350 051664 000443 BR END61 ;SRIP REST OF TEST
2351 |
2352 051666 022767 000103 166114 ISA61: CMP #103, SP0W ;CHECK PS AFTER FIS TRAP
2353 051674 001404 BEQ ;+4 ;BRANCH IF OK
2354 051676 104000 HLT ;PS AFTER FIS TRAP NOT EQUAL TO 103
2355 |
2356 051700 022767 051630 166106 CMP #RTA61, ANS1 ;CHECK FIS TRAP RETURN ADDRESS
2357 051706 001404 BEQ ;+4 ;BRANCH IF OK
2358 051710 104000 HLT ;FIS TRAP AT WRONG ADDRESS
2359 |
2360 051712 022767 000042 166076 CMP #042, ANS2 ;CHECK PS BEFORE FIS TRAP
2361 051720 001404 BEQ ;+4 ;BRANCH IF OK
2362 051722 104000 HLT ;PS AT FIS TRAP TIME NOT 042
2363 |
2364 051724 022767 014377 166066 CMP #014377, ANS3 ;CHECK DATA FROM THE STACK
2365 051732 001404 BEQ ;+4 ;BRANCH IF OK
2366 051734 104000 HLT ;DATA ON STACK (014377) CHANGED
2367 |
2368 051736 022767 177777 166056 CMP #177777, ANS4 ;CHECK DATA FROM STACK
2369 051744 001404 BEQ ;+4 ;BRANCH IF OK
2370 051746 104000 HLT ;DATA ON STACK (177777) CHANGED
2371 |
2372 051750 022767 154200 166046 CMP #154200, ANS5 ;CHECK DATA FROM STACK
2373 051756 001404 BEQ ;+4 ;BRANCH IF OK
2374 051760 104000 HLT ;DATA ON STACK (154200) CHANGED
2375 |
2376 051762 022767 000000 166036 CMP #000000, ANS6 ;CHECK DATA FROM STACK
2377 051770 001404 BEQ ;+4 ;BRANCH IF OK
2378 051772 104000 HLT ;DATA ON STACK (000000) CHANGED
2379 |

```

```

2380 051774 122767 000061 165776 EN0611 CMPB #61, ICNT ICHECK THE TEST NUMBER
2381 052002 001401 BEQ ,+4 IBRANCH IF OK
2382 052004 104000 HLT IWRONG TEST! PC MUST HAVE FOULED UP.
2383
2384 052006 104400 SCOPE
2385
2386
2387
2388
2389
2390
2391
2392
2393 052010 012767 000062 165762 TS7621 MOV #62, ICNT IKEEP TRACK OF TEST NUMBER
2394 052016 004567 001552 JSR R3, PUSH5 IPUSH 4 WORDS ONTO STACK, SET PRIORITY
2395 052022 027652 125253 ;WORD 027652,125253 ISECOND OPERAND ON TOP
2396 052026 167452 125252 ;WORD 167452,125252 IFIRST OPERAND ON BOTTOM
2397 052032 000100 ;WORD 100 IPROCESSOR PRIORITY LEVEL
2398 052034 054236 000340 ;WORD TRAPER, 340 IFIS TRAP VECTOR
2399
2400 052040 000240 NOP
2401 052042 075036 FDIV+ SP IFLOATING DIVIDE ON THE STACK
2402
2403 052044 004767 001564 JSR PC, POP5 IPOP THE ANSWER
2404 052050 026706 165726 CMP STACK, SP ICHECK THE STACK POINTER
2405 052054 001404 BEQ TSA62 IBRANCH IF OK
2406 052056 016706 165720 MOV STACK, SP IRESTORE STACK POINTER
2407 052062 104000 HLT ISTACK POINTER FOULED UP
2408 052064 000417 BR END62 ISRIP REST OF TEST
2409
2410 052066 022767 000110 165714 TSA621 CMP #110, SPSW ICHECK PS (EXCEPT T BIT)
2411 052074 001401 BEQ ,+4 IBRANCH IF OK
2412 052076 104000 HLT IPB NOT EQUAL TO 110
2413
2414 052100 022767 177777 165706 CMP #177777,ANS1 ICHECK FIRST HALF OF ANSWER
2415 052106 001401 BEQ ,+4 IBRANCH IF OK
2416 052110 104000 HLT IANS1 NOT EQUAL TO 177777
2417
2418 052112 022767 177777 165676 CMP #177777,ANS2 ICHECK SECOND HALF OF ANSWER
2419 052120 001401 BEQ ,+4 IBRANCH IF OK
2420 052122 104000 HLT IANS2 NOT EQUAL TO 177777
2421
2422 052124 122767 000062 165646 EN0621 CMPB #62, ICNT ICHECK THE TEST NUMBER
2423 052132 001401 BEQ ,+4 IBRANCH IF OK
2424 052134 104000 HLT IWRONG TEST! PC MUST HAVE FOULED UP.
2425
2426 052136 104400 SCOPE
2427
2428
2429
2430
2431
2432
2433

```

```

|*****
|TEST 621 FDIV (KE11F FLOATING DIVIDE INSTRUCTION)
| 167452,125252 / 027652,125253 = 177777,177777
| PS = 110, STACK POINTER = SP
|*****

```

```

|*****
|TEST 631 FDIV (KE11F FLOATING DIVIDE INSTRUCTION)
| 125252,125252 / 069252,125252 = 100200,000000
| PS = 010, STACK POINTER = R2
|*****

```



2488	052334	004767	001444		RTA64	JSR	X7,	POPR	IPOP THE "ANSWER"
2489	052340	010167	165446			MOV	R1,	SSP	ISAVE STACK POINTER (R1)
2490	052344	104000				HLT			IFIS TRAP DIDN'T OCCURE!
2491	052346	000454				BR	END64		
2492									
2493	052350	004767	001462		ISR64	JSR	X7,	POPER	IPOP ALL DATA OFF THE STACKS
2494	052354	010167	165432			MOV	R1,	SSP	ISAVE STACK POINTER (R1)
2495	052360	022767	000100	165422		CMP	#100,	SPSW	ICHECK PS AFTER FIS TRAP
2496	052366	001401				BEG	:+4		IBRANCH IF OK
2497	052370	104000				HLT			IPS AFTER FIS TRAP NOT EQUAL TO 100
2498									
2499	052372	022767	040040	165412		CMP	#STACK0,SSP		ICHECK THE STACK POINTER (R1)
2500	052400	001401				BEG	:+4		IBRANCH IF OK
2501	052402	104000				HLT			ISTACK POINTER (R1) NOT EQUAL TO #STACK0
2502									
2503	052404	022767	052334	165402		CMP	#RTA64,ANS1		ICHECK FIS TRAP RETURN ADDRESS
2504	052412	001401				BEG	:+4		IBRANCH IF OK
2505	052414	104000				HLT			IFIS TRAP AT WRONG ADDRESS
2506									
2507	052416	022767	000012	165372		CMP	#012,ANS2		ICHECK PS BEFORE FIS TRAP
2508	052424	001401				BEG	:+4		IBRANCH IF OK
2509	052426	104000				HLT			IPS AT FIS TRAP TIME NOT 012
2510									
2511	052430	022767	065252	165362		CMP	#065252,ANS3		ICHECK DATA FROM THE STACK
2512	052436	001401				BEG	:+4		IBRANCH IF OK
2513	052440	104000				HLT			DATA ON STACK (065252) CHANGED
2514									
2515	052442	022767	125252	165352		CMP	#125252,ANS4		ICHECK DATA FROM STACK
2516	052450	001401				BEG	:+4		IBRANCH IF OK
2517	052452	104000				HLT			DATA ON STACK (125252) CHANGED
2518									
2519	052454	022767	025252	165342		CMP	#025252,ANS5		ICHECK DATA FROM STACK
2520	052462	001401				BEG	:+4		IBRANCH IF OK
2521	052464	104000				HLT			DATA ON STACK (025252) CHANGED
2522									
2523	052466	022767	125251	165332		CMP	#125251,ANS6		ICHECK DATA FROM STACK
2524	052474	001401				BEG	:+4		IBRANCH IF OK
2525	052476	104000				HLT			DATA ON STACK (125251) CHANGED
2526									
2527	052500	122767	000064	165272	END64	CMPB	#64,	ICNT	ICHECK THE TEST NUMBER
2528	052506	001401				BEG	:+4		IBRANCH IF OK
2529	052510	104000				HLT			WRONG TEST! PC MUST HAVE FOULED UP!
2530									
2531	052512	104400							SCOPE
2532									
2533									
2534									
2535									
2536									
2537									
2538									
2539									
2540	052514	012767	000065	165256	TS765	MOV	#65,	ICNT	KEEP TRACK OF TEST NUMBER
2541	052522	004567	001402			JSR	R5,	PUSH7	IPUSH 4 WORDS ONTO STACK, SET PRIORITY

```

|*****
|TEST 651      FADD (KE11F FLOATING ADD INSTRUCTION)
|      #04000,105004 + 104000,104000 = 000401,000000
|      PS = 140,      STACK POINTER = PC
|*****

```

2542	052526	052552				,WORD	STK65			ITOP OF STACK
2543	052530	104000	104000			,WORD	104000,104000			ISCOND OPERAND ON TOP
2544	052534	004000	105004			,WORD	004000,105004			IFIRST OPERAND ON BOTTOM
2545	052540	000144				,WORD	144			IPROCESSOR PRIORITY LEVEL
2546	052542	054236	000340			,WORD	TRAPER,340			IFIS TRAP VECTOR
2547										
2548	052546	000240				NOP				
2549	052550	075007				FADD+	PC			IFLOATING ADD ON FOLLOWING 4 WORDS
2550	052552	104000			STR65:	104000				ISHOULD CONTAIN 104000
2551	052554	104000				104000				ISHOULD CONTAIN 104000
2552	052556	004000				004000				IBEFORE FADD, 004000; AFTER, 000400
2553	052560	105004				105004				IBEFORE FADD, 105004; AFTER, 000000
2554										
2555	052562	004767	001400			JSR	PC, P0#7			IPHP THE ANSWER
2556	052566	022767	000140	165214		CMP	#140, SP#W			ICHECK PS (EXCEPT 7 BIT)
2557	052574	001401				REQ	,+4			IBRANCH IF OK
2558	052576	104000				HLT				IF# NOT EQUAL TO 140
2559										
2560	052600	022767	104000	165206		CMP	#104000,ANS1			ICHECK FIRST HALF OF INPUT DATA (STK65)
2561	052606	001401				REQ	,+4			IBRANCH IF OK
2562	052610	104000				HLT				IANS1 NOT EQUAL TO 104000
2563										
2564	052612	022767	104000	165176		CMP	#104000,ANS2			ICHECK SECOND HALF OF INPUT DATA (STK65+2)
2565	052620	001401				REQ	,+4			IBRANCH IF OK
2566	052622	104000				HLT				IANS2 NOT EQUAL TO 104000
2567										
2568	052624	022767	000401	165166		CMP	#000401,ANS3			ICHECK FIRST HALF OF ANSWER
2569	052632	001401				REQ	,+4			IBRANCH IF OK
2570	052634	104000				HLT				IANS3 NOT EQUAL TO 000401
2571										
2572	052636	022767	000000	165196		CMP	#000000,ANS4			ICHECK SECOND HALF OF ANSWER
2573	052644	001401				REQ	,+4			IBRANCH IF OK
2574	052646	104000				HLT				IANS4 NOT EQUAL TO 000000
2575										
2576	052650	122767	000065	165122	EN065:	CMPB	#65, ICNT			ICHECK THE TEST NUMBER
2577	052656	001401				REQ	,+4			IBRANCH IF OK
2578	052660	104000				HLT				IWRONG TEST! PC MUST HAVE FOULED UP!
2579										
2580	052662	104400					SCOPE			
2581										
2582										
2583										
2584										
2585										
2586										
2587										
2588										
2589	052664	012767	000066	165186	TS#66:	MOV	#66, ICNT			KEEP TRACK OF TEST NUMBER
2590	052672	004567	001232			JSR	R5, PUSH#			IPUSH 4 WORDS ONTO STACK, SET PRIORITY
2591	052676	052722				,WORD	STK66			ITOP OF STACK
2592	052700	104000	104000			,WORD	104000,104000			ISCOND OPERAND ON TOP
2593	052704	104000	105004			,WORD	104000,105004			IFIRST OPERAND ON BOTTOM
2594	052710	000092				,WORD	#52			IPROCESSOR PRIORITY LEVEL
2595	052712	054236	000340			,WORD	TRAPER,340			IFIS TRAP VECTOR

```

|*****|
|TEST 66|       FSUB (KE11F FLOATING SUBTRACT INSTRUCTION)
|      |       104000,105004 = 104000,104000 + 100401,000000
|      |       PS = 030,         STACK POINTER = PC
|*****|

```

2596										
2597	052716	000240								
2598	052720	075017								
2599	052722	104000								
2600	052724	104000			STK661	104000				
2601	052726	104000				104000				
2602	052730	105004				105004				
2603										
2604	052732	004767	001230							
2605	052736	022767	000050	165044						
2606	052744	001401								
2607	052746	104000								
2608										
2609	052750	022767	104000	165036						
2610	052756	001401								
2611	052760	104000								
2612										
2613	052762	022767	104000	165026						
2614	052770	001401								
2615	052772	104000								
2616										
2617	052774	022767	100401	165016						
2618	053002	001401								
2619	053004	104000								
2620										
2621	053006	022767	000000	165006						
2622	053014	001401								
2623	053016	104000								
2624										
2625	053020	122767	000066	164732	EN0661					
2626	053026	001401								
2627	053030	104000								
2628										
2629	053032	104400								
2630										
2631										
2632										
2633										
2634										
2635										
2636										
2637										
2638	053034	012767	000067	164736	TS1671					
2639	053042	004567	001062							
2640	053046	053072								
2641	053050	104000	104000							
2642	053054	134600	073601							
2643	053060	000046								
2644	053062	054236	000340							
2645										
2646	053066	000240								
2647	053070	079027								
2648	053072	104000								
2649	053074	104000								

  

```

*****
TEST 671      FMUL (KE11F FLOATING MULTIPLY INSTRUCTION)
|          134600,073601 * 104000,104000 = 000401,000000
|          PS = 040,      STACK POINTER = PC
*****
    
```

2650	053076	134600			134600					IBEFORE FMUL, 134600; AFTER, 000401
2651	053100	073601			073601					IBEFORE FMUL, 073601; AFTER, 000000
2652										
2653	053102	004767	001000			JSR	PC,	POP7		IPOP THE ANSWER
2654	053106	022767	000040	164674		CMP	#040,	SPSW		ICHECK PS (EXCEPT T BIT)
2655	053114	001401				REQ	:+4			IBRANCH IF OK
2656	053116	104000				HLT				IPS NOT EQUAL TO 040
2657										
2658	053120	022767	104000	164666		CMP	#104000,ANS1			ICHECK FIRST HALF OF INPUT DATA (STK67)
2659	053126	001401				REQ	:+4			IBRANCH IF OK
2660	053130	104000				HLT				IANS1 NOT EQUAL TO 104000
2661										
2662	053132	022767	104000	164656		CMP	#104000,ANS2			ICHECK SECOND HALF OF INPUT DATA (STK67+2)
2663	053140	001401				REQ	:+4			IBRANCH IF OK
2664	053142	104000				HLT				IANS2 NOT EQUAL TO 104000
2665										
2666	053144	022767	000401	164646		CMP	#000401,ANS3			ICHECK FIRST HALF OF ANSWER
2667	053152	001401				REQ	:+4			IBRANCH IF OK
2668	053154	104000				HLT				IANS3 NOT EQUAL TO 000401
2669										
2670	053156	022767	000000	164636		CMP	#000000,ANS4			ICHECK SECOND HALF OF ANSWER
2671	053164	001401				REQ	:+4			IBRANCH IF OK
2672	053166	104000				HLT				IANS4 NOT EQUAL TO 000000
2673										
2674	053170	122767	000067	164602	END67:	CMPB	#67,	ICNT		ICHECK THE TEST NUMBER
2675	053176	001401				REQ	:+4			IBRANCH IF OK
2676	053200	104000				HLT				IFRONG TEST! PC MUST HAVE POULLED UP!
2677										
2678	053202	104400								SCOPE
2679										
2680										
2681										
2682										
2683										
2684										
2685										
2686										
2687	053204	012767	000070	164566	TSF70:	MOV	#70,	ICNT		IKEEP TRACK OF TEST NUMBER
2688	053212	004567	000712			JSR	R5,	PUSH7		IPUSH 4 WORDS ONTO STACK, SET PRIORITY
2689	053216	053242				WORD	STK70			ITOP OF STACK
2690	053220	104000	104000			WORD	104000,104000			ISECOND OPERAND ON TOP
2691	053224	102500	146000			WORD	102500,146000			IFIRST OPERAND ON BOTTOM
2692	053230	000157				WORD	157			IPROCESSOR PRIORITY LEVEL
2693	053232	054236	000340			WORD	TRAPER,340			IFIS TRAP VECTOR
2694										
2695	053236	000240				NOP				
2696	053240	075037				FDIV+	PC			IFLOATING DIVIDE ON FOLLOWING 4 WORDS
2697	053242	104000				STK70:	104000			ISHOULD CONTAIN 104000
2698	053244	104000					104000			ISHOULD CONTAIN 104000
2699	053246	102500					102500			IBEFORE FDIV, 102500; AFTER, 036700
2700	053250	146000					146000			IBEFORE FDIV, 146000; AFTER, 000000
2701										
2702	053252	004767	000710			JSR	PC,	POP7		IPOP THE ANSWER
2703	053256	022767	000140	164524		CMP	#140,	SPSW		ICHECK PS (EXCEPT T BIT)

```

*****
ITEST 70:      FDIV (KE11F FLOATING DIVIDE INSTRUCTION)
I      102500,146000 / 104000,104000 = 036700,000000
I      PS = 140,      STACK POINTER = PC
*****

```



2704	053264	001401			BEQ	,+4		IBRANCH IF OK
2705	053266	104000			HLT			IJS NOT EQUAL TO 140
2706								
2707	053270	022767	104000	164516	CMP	#104000,ANS1		ICHECK FIRST HALF OF INPUT DATA (STK70)
2708	053276	001401			BEQ	,+4		IBRANCH IF OK
2709	053300	104000			HLT			IANS1 NOT EQUAL TO 104000
2710								
2711	053302	022767	104000	164506	CMP	#104000,ANS2		ICHECK SECOND HALF OF INPUT DATA (STK70+2)
2712	053310	001401			BEQ	,+4		IBRANCH IF OK
2713	053312	104000			HLT			IANS2 NOT EQUAL TO 104000
2714								
2715	053314	022767	036700	164476	CMP	#036700,ANS3		ICHECK FIRST HALF OF ANSWER
2716	053322	001401			BEQ	,+4		IBRANCH IF OK
2717	053324	104000			HLT			IANS3 NOT EQUAL TO 036700
2718								
2719	053326	022767	000000	164466	CMP	#000000,ANS4		ICHECK SECOND HALF OF ANSWER
2720	053334	001401			BEQ	,+4		IBRANCH IF OK
2721	053336	104000			HLT			IANS4 NOT EQUAL TO 000000
2722								
2723	053340	122767	000070	164432	END70:	CMPB	#70, ICNT	ICHECK THE TEST NUMBER
2724	053346	001401			BEQ	,+4		IBRANCH IF OK
2725	053350	104000			HLT			IWRONG TEST; PC MUST HAVE FOULED UP.
2726								
2727	053352	104400			SCOPE			
2728								
2729								
2730								
2731								
2732								
2733								
2734								
2735								
2736								

```

*****
|TEST 711 TEST ALL INSTRUCTION TOGETHER
|                                032107,065432,045670,123456
| 134343,107070 + ----- = 137201,119230
|                                (135252,125252,040616,016161)
| PS=150, STACK POINTER=R4
*****

```

2737								
2738	053354	012737	000071	040000	TS71:	MOV	#71, #ICNT	ISSET UP SUBTEST NUMBER
2739	053362	012704	040050			MOV	#STACK0,R4	ISSET STACK POINTER
2740	053366	012744	107070			MOV	#107070,=(R4)	ILDA0 DATA ONTO STACK
2741	053372	012744	134343			MOV	#134343,=(R4)	
2742	053376	012744	065432			MOV	#065432,=(R4)	
2743	053402	012744	032107			MOV	#032107,=(R4)	
2744	053406	012744	123456			MOV	#123456,=(R4)	
2745	053412	012744	045670			MOV	#045670,=(R4)	
2746	053416	012744	125252			MOV	#125252,=(R4)	
2747	053422	012744	135252			MOV	#135252,=(R4)	
2748	053426	012744	016161			MOV	#016161,=(R4)	
2749	053432	012744	040616			MOV	#040616,=(R4)	
2750	053436	012737	000144	177776		MOV	#144, #MS	ISSET PROCESSOR STATUS
2751								
2752	053444	000240				NOP		
2753	053446	075014				FSUB+	R4	1135252,125252,040616,016161,140616,017434
2754	053450	075034				FDIV+	R4	1045670,123456,140616,017434,145246,047065
2755	053452	075024				FMUL+	R4	1032107,065432,145246,047065,137201,106137
2756	053454	075004				FADD+	R4	1134343,107070,137201,106137,137201,119230
2757								

2758	053456	013767	177776	164324	MOV	#PS,	SPSW	ISAVE FINAL PS
2759	053464	042767	000020	164316	BIC	#20,	SPSW	ICLR T=BIT
2760	053472	012467	164316		MOV	(R4)+,	ANS1	ISAVE FIRST HALF OF ANSWER
2761	053476	012467	164314		MOV	(R4)+,	ANS2	ISAVE SECOND HALF OF ANSWER
2762	053502	010467	164304		MOV	R4,	SSP	ISAVE STACK POINTER
2763	053506	022767	000150	164274	CMP	#150,	SPSW	ICHECK PS (EXCEPT T BIT)
2764	053514	001401			REQ	.*4		IBRANCH IF OK
2765	053516	104000			HLT			IPS NOT EQUAL TO 150
2766								
2767	053520	022767	040050	164264	CMP	#STACK0,SSP		ICHECK THE STACK POINTER (R4)
2768	053526	001401			REQ	.*4		IBRANCH IF OK
2769	053530	104000			HLT			ISTACK POINTER (R4) NOT EQUAL TO 674
2770								
2771	053532	022767	137201	164254	CMP	#137201,ANS1		ICHECK FIRST HALF OF ANSWER
2772	053540	001401			REQ	.*4		IBRANCH IF OK
2773	053542	104000			HLT			IANS1 NOT EQUAL TO 137201
2774								
2775	053544	022767	115230	164244	CMP	#115230,ANS2		ICHECK SECOND HALF OF ANSWER
2776	053552	001401			REQ	.*4		IBRANCH IF OK
2777	053554	104000			HLT			IANS2 NOT EQUAL TO 115230
2778								
2779	053556	122767	000071	164214	ENB71:	CMPB	#71,	ICNT
2780	053564	001401			REQ	.*4		ICHECK THE TEST NUMBER
2781	053566	104000			HLT			IBRANCH IF OK
2782								IWRONG TEST! PC MUST HAVE FOULED UP!
2783	053570	104400			SCOPE			
2784								
2785								
2786	053572	000002			DONE1	RTI		IRETURN TO GTP

```

2787
2788
2789
2790 053574 005726
2791 053576 062705 000010
2792 053602 014546
2793 053604 014546
2794 053606 014546
2795 053610 014546
2796 053612 062705 000010
2797 053616 012537 177776
2798 053622 012577 164156
2799 053626 012577 164154
2800 053632 000115
2801
2802
2803
2804
2805
2806 053634 013767 177776 164146
2807 053642 042767 000020 164140
2808 053650 012604
2809 053652 012667 164136
2810 053656 012667 164134
2811 053662 010667 164124
2812 053666 000114
2813
2814
2815
2816
2817
2818
2819
2820 053670 013767 177776 164112
2821 053676 012604
2822 053700 012667 164110
2823 053704 011667 164106
2824 053710 042767 000020 164100
2825 053716 012746 053724
2826 053722 000002
2827 053724 012667 164070
2828 053730 012667 164066
2829 053734 012667 164064
2830 053740 012667 164062
2831 053744 010667 164042
2832 053750 000114
2833
2834
2835
2836 053752 012704 040040
2837 053756 012524
2838 053760 012524
2839 053762 012524
2840 053764 012524

```

ISUBROUTINE TO PUSH 4 WORDS ONTO THE STACK

```

PUSHSI TST (SP)+ IPOP STACK BY 1
        ADD #10, R5 IPOINT TO END OF DATA
        MOV -(R5), -(SP) IPUSH DATA ONTO THE STACK
        MOV -(R5), -(SP) IPUSH DATA ONTO THE STACK
        MOV -(R5), -(SP) IPUSH DATA ONTO THE STACK
        MOV -(R5), -(SP) IPUSH DATA ONTO THE STACK
        ADD #10, R5 IPOINT TO END OF DATA
        MOV (R5)+, #0 ISET THE PROCESSOR STATUS
        MOV (R5)+, #0 ISET UP FIS ERROR TRAP VECTOR
        MOV (R5)+, #0 ITRAP STATUS
        JMP (R5) IRETURN

```

ISUBROUTINE TO POP 2 WORDS OFF THE STACK  
ALSO SAVES THE PROCESSOR STATUS WORD (EXCEPT T BIT)

```

POPSI MOV #0, SPW ISAVE PROCESSOR STATUS WORD
       BIC #20, SPW ICLEAR T-BIT
       MOV (SP)+, R4 ISAVE RTS ADDRESS
       MOV (SP)+, ANS1 ISAVE THE ANSWER
       MOV (SP)+, ANS2
       MOV SP, SSP ISAVE THE STACK POINTER
       JMP (R4) IRETURN

```

ISUBROUTINE TO POP 6 WORDS OFF THE STACK.  
THE FIRST TWO WERE PUT ON BY THE ERROR TRAP.  
THE LAST FOUR WERE THE ORIGINAL INPUT DATA.  
ALSO SAVES THE PS AND STACK POINTER.

```

POPSI MOV #0, SPW ISAVE PROCESSOR STATUS WORD
       MOV (SP)+, R4 ISAVE RTS ADDRESS
       MOV (SP)+, ANS1 ISAVE RTI ADDRESS
       MOV (SP)+, ANS2 ISAVE RTI STATUS
       BIC #20, ANS2 ICLEAR THE T-BIT
       MOV #15, -(SP)
       RTI IRESTORE THE PROCESSOR STATUS
       MOV (SP)+, ANS3 ISAVE DATA
       MOV (SP)+, ANS4
       MOV (SP)+, ANS5
       MOV (SP)+, ANS6
       MOV SP, SSP ISAVE SP
       JMP (R4) IRTS

```

ISUBROUTINE TO PUSH 4 WORDS ONTO THE STACK

```

PUSHRI MOV #STACK0, R4 ISET R4 TO STACK
        MOV (R5)+, (R4)+ IPUT DATA ON STACK
        MOV (R5)+, (R4)+
        MOV (R5)+, (R4)+
        MOV (R5)+, (R4)+

```

```

2841 053766 012537 177776      MOV      (R5)+, 0#PS      ISET THE PROCESSOR STATUS
2842 053772 012577 164006      MOV      (R5)+, 0FISVEC  ISET UP FIS ERROR TRAP VECTOR
2843 053776 012577 164004      MOV      (R5)+, 0FISLVL  ITRAP STATUS
2844 054002 000205      RTS      R5              IRETURN
2845
2846

```

ISUBROUTINE TO POP 2 WORDS OFF THE STACK  
 ALSO SAVES THE PROCESSOR STATUS WORD (EXCEPT T BIT)

```

2847
2848
2849
2850 054004 013767 177776 163776  POPR1  MOV      0#PS,  SP$W      ISAVE PROCESSOR STATUS WORD
2851 054012 042767 000020 163770  BIC      #20,  SP$W      ICLEAR T-BIT
2852 054020 016767 164020 163766  MOV      STACK4, AN$1    ISAVE THE ANSWER
2853 054026 016767 164014 163762  MOV      STACK6, AN$2    I
2854 054034 000207      RTS      X7
2855
2856

```

ISUBROUTINE TO POP 6 WORDS OFF THE STACKS,  
 THE TWO OFF THE R6 STACK WERE PUT ON BY THE ERROR TRAP,  
 THE FOUR OFF THE SOFTWARE STACK WERE THE ORIGINAL INPUT DATA,  
 ALSO SAVES THE PS AND STACK POINTER AFTER THE FIS TRAP.

```

2857
2858
2859
2860
2861
2862 054036 013767 177776 163744  POPER1 MOV      0#PS,  SP$W      ISAVE PROCESSOR STATUS WORD
2863 054044 012667 000056      MOV      (SP)+,  SAVR$S   ISAVE RTS ADDRESS
2864 054050 012667 163740      MOV      (SP)+,  AN$1     ISAVE RTI ADDRESS
2865 054054 011667 163736      MOV      (SP),   AN$2     ISAVE RTI STATUS
2866 054060 042767 000020 163730  BIC      #20,  AN$2     ICLEAR THE T-BIT
2867 054066 012746 054074      MOV      #1$,   -(SP)
2868 054072 000002      RTI
2869 054074 016767 163740 163716  1SI   MOV      STACK0, AN$3    ISAVE DATA
2870 054102 016767 163734 163712  MOV      STACK2, AN$4    I
2871 054110 016767 163730 163706  MOV      STACK4, AN$5    I
2872 054116 016767 163724 163702  MOV      STACK6, AN$6    I
2873 054124 000137      JMP      0(X7)+         ISIMULATED RTS
2874 054126 000000      SAVR$S 0              IRTS ADDRESS SAVE
2875
2876

```

ISUBROUTINE TO PUSH 4 WORDS ONTO THE PC STACK

```

2877
2878 054130 012504      PUSH$1 MOV      (R5)+,  R4      ISET R4 TO STACK
2879 054132 012524      MOV      (R5)+,  (R2)+    IPUT DATA ON STACK
2880 054134 012524      MOV      (R5)+,  (R2)+    I
2881 054136 012524      MOV      (R5)+,  (R2)+    I
2882 054140 012524      MOV      (R5)+,  (R2)+    I
2883 054142 042737 177757 177776  BIC      #17757, 0#PS    ICLEAR STATUS EXCEPT T-BIT
2884 054150 052537 177776      BIS      (R5)+,  0#PS    ISET THE PROCESSOR STATUS
2885 054154 012577 163624      MOV      (R5)+,  0FISVEC  ISET UP FIS ERROR TRAP VECTOR
2886 054160 012577 163622      MOV      (R5)+,  0FISLVL  ITRAP STATUS
2887 054164 000205      RTS      R5              IRETURN
2888
2889

```

ISUBROUTINE TO POP 4 WORDS OFF THE PC "STACK"  
 ALSO SAVES THE PROCESSOR STATUS WORD (EXCEPT T BIT)

```

2890
2891
2892 054166 013767 177776 163614  POP$1  MOV      0#PS,  SP$W      ISAVE PROCESSOR STATUS WORD
2893 054174 042767 000020 163606  BIC      #20,  SP$W      ICLEAR T-BIT
2894 054202 011600      MOV      (SP),  R0      IGET RETURN ADDRESS

```

2895	054204	162700	000014	SUB	#14,	R0	POINT TO TOP OF "PC STACK"
2896	054210	012067	163600	MOV	(R0)+,	ANS1	SAVE 1ST HALF INPUT DATA
2897	054214	012067	163576	MOV	(R0)+,	ANS2	SAVE 2ND HALF INPUT DATA
2898	054220	010067	163566	MOV	R0,	*SP	SAVE ASSUMED END PC "STACK POINTER"
2899	054224	012067	163570	MOV	(R0)+,	ANS3	SAVE 1ST HALF OF ANSWER
2900	054230	012067	163566	MOV	(R0)+,	ANS4	SAVE 2ND HALF OF ANSWER
2901	054234	000207		RTS		X7	
2902							
2903				ERRONIOUS TRAP SERVICE ROUTINE			
2904							
2905	054236	104000		TRAPERI	HLT		IF IS SHOULDN'T HAVE TRAPED
2906	054240	000002			RTI		
2907							
2908		000001		END			

ANS1	040014	ANS2	040016	ANS3	040020	ANS4	040022
ANS5	040024	ANS6	040026	BELL	000007	RI90	000001
BIT1	000002	BIT10	002000	BIT11	004000	RI91	010000
BIT13	020000	BIT14	040000	BIT15	100000	RI92	000004
BIT3	000010	BIT4	000020	BIT5	000040	RI96	000100
BIT7	000200	BIT8	000400	BIT9	001000	DISPLA	177570
DONE	053572	END1	040204	END10	041400	FN011	041620
END12	041754	END13	042170	END14	042320	FN015	042450
END16	042670	END17	043024	END2	040340	FN020	043240
END21	043374	END22	043930	END23	043664	FN024	044020
END25	044154	END26	044310	END27	044444	FN03	040474
END30	044574	END31	044730	END32	045064	FN033	045220
END34	045434	END35	045964	END36	046004	FN037	046140
END4	040624	END40	046274	END41	046430	FN042	046564
END43	046720	END44	047054	END45	047210	FN046	047344
END47	047474	END5	040734	END50	047710	FN051	050044
END52	050264	END53	050420	END54	050640	FN055	050770
END56	051204	END57	051340	END6	041110	FN060	051560
END61	051774	END62	052124	END63	052260	FN064	052500
END65	052650	END66	053020	END67	053170	FN07	041244
END70	053340	END71	053956	FADD	075000	FDIV	075030
FISLVL	040000	FIBVEC	040004	FMUL	075020	FSUB	075010
HLT	104000	ICNT	040000	ISA13	042062	ISA20	043132
ISA34	045326	ISA50	047602	ISA56	051076	ISA61	051666
ISR11	041470	ISR13	042040	ISR16	042540	ISA20	043110
ISR34	045304	ISR36	045654	ISR50	047560	ISA52	050134
ISR54	050510	ISR56	051054	ISR60	051430	ISA61	051644
ISR64	052350	KE11F	040060	N	000072	PC	X000007
POPER	054036	POPER	053670	POPR	054004	POPS	053634
POP7	054166	PS	177776	PUSHR	053752	PUSHS	053574
PUSH7	054130	RTA11	041454	RTA13	042024	RTA16	042524
RTA20	043074	RTA34	045270	RTA36	045640	RTA50	047544
RTA52	050120	RTA54	050474	RTA56	051040	RTA60	051414
RTA61	051630	RTA64	052334	R0	X000000	RI	X000001
R2	X000002	R3	X000003	R4	X000004	RS	X000005
SAVRTS	054126	SCOPE	104400	SP	X000006	STACK	040002
STACK0	040040	STACK2	040042	STACK4	040044	STACK6	040046
STACK8	040050	STK65	052952	STK66	052722	STR67	053072
STK70	053242	SHR	177970	SW00	000400	SW09	001000
SW10	002000	SHL	004000	SW12	010000	SW13	020000
SW14	040000	SHIS	100000	TRAPER	054236	TSI14	042262
TSA15	042412	TSA30	044936	TSA35	045926	TSI4	040966
TSA47	047436	TSA5	040710	TSA55	050732	TSI62	052066
TS91	040064	TS910	041260	TS911	041410	TS912	041634
TS913	041770	TS914	042304	TS919	042334	TS91A	042464
TS917	042704	TS92	040220	TS920	043040	TS921	043254
TS922	043410	TS923	043944	TS924	043700	TS925	044034
TS926	044170	TS927	044324	TS93	040354	TS930	044460
TS931	044610	TS932	044744	TS933	045100	TS934	045234
TS935	045450	TS936	045600	TS937	046020	TS94	040910
TS940	046154	TS941	046310	TS942	046444	TS943	046600
TS944	046734	TS945	047070	TS946	047224	TS947	047360
TS95	040040	TS950	047510	TS951	047724	TS952	050060
TS953	050300	TS954	050434	TS955	050654	TS956	051004

MAINDEC=11-DBKED=A  
DBKED;P11

KEIIF (PDP-11 PIS) SYSTEM EXERCISER OVERLAY  
SYMBOL TABLE

MACH11.620 3-OCT-72 17129 PAGE 58

TST57 051220  
TST62 052010  
TST66 052064  
TST71 053354  
SSP 040012

TST6 040770  
TST63 052140  
TST67 053034  
TTY =X000005  
;BIT = 177777

TST60 051354  
TST64 052274  
TST7 041124  
TYPE = 000004  
= 054242

TST61 051574  
TST65 052514  
TST70 053204  
SPSW 040010

ERRORS DETECTED: 0





**digital**
 133MCN  
 MAINDEC CHANGE  
 NOTICE

CHANGE NO.

11-DBKEA-A -1

Sheet 1 of 1

<b>AUTHOR</b> K. Chapman <b>DATE</b> <b>EXT.</b> 9/29/72    3123	<b>PROGRAM DATE</b> AUGUST 72	<b>PRODUCT LINE</b> PDP-11/40 V20 06251	<b>MAINDEC NUMBER</b> 11-DBKEA-A									
<b>PROGRAM NAME</b> KellF Instruction Tests		<b>DEVICE</b> KellF Floating Point										
<b>ITEM</b> 0.	The KellF instruction tests program is available from the program library.											
1. 11/29/72	<p><b>PROBLEM:</b> The instruction at location 1072 (MOV 1\$,@#4) should be:                    JMP @#1\$</p> <p><b>CORRECTION:</b> Change the following locations as indicated:</p> <table border="0" data-bbox="324 829 1023 1008"> <thead> <tr> <th style="text-align: left;"><u>Location</u></th> <th style="text-align: left;"><u>Old</u></th> <th style="text-align: left;"><u>New</u></th> </tr> </thead> <tbody> <tr> <td>1072</td> <td>016737</td> <td>000137</td> </tr> <tr> <td>1074</td> <td>000006</td> <td>001104</td> </tr> </tbody> </table>			<u>Location</u>	<u>Old</u>	<u>New</u>	1072	016737	000137	1074	000006	001104
<u>Location</u>	<u>Old</u>	<u>New</u>										
1072	016737	000137										
1074	000006	001104										

