

This image displays a grid of 100 small data sheets, arranged in 10 rows and 10 columns. Each sheet contains technical specifications, diagrams, and component information for various parts, likely related to the PDP11/6X system. The sheets are organized into sections, with some larger sheets on the left side and smaller ones on the right. The text on the sheets is dense and includes various alphanumeric codes and technical details.

KD11-K

PDP11/6X FP11E FP ADV.
MD-11-DQFPB-A

EP-DQFPB-A-DL-A

COPYRIGHT © 1977

FICHE 2 OF 2

APR 1977

digital

MADE IN USA

This microfiche grid contains 144 frames arranged in 12 columns and 12 rows. The frames contain various data, including:

- Column 1: Contains text and small diagrams, possibly related to system specifications or user manuals.
- Column 2: Contains vertical bar patterns, likely representing binary data or test results.
- Column 3: Contains vertical bar patterns, similar to column 2.
- Column 4: Contains vertical bar patterns, similar to column 2.
- Column 5: Contains vertical bar patterns, similar to column 2.
- Column 6: Contains vertical bar patterns, similar to column 2.
- Column 7: Contains vertical bar patterns, similar to column 2.
- Column 8: Contains vertical bar patterns, similar to column 2.
- Column 9: Contains vertical bar patterns, similar to column 2.
- Column 10: Contains vertical bar patterns, similar to column 2.
- Column 11: Contains vertical bar patterns, similar to column 2.
- Column 12: Contains vertical bar patterns, similar to column 2.

B01

EOF100FPB880411
DQFPBA.MEM

09-FEB-77 10:03

ADVANCED INSTRUCTION TESTS

MACY PDP28 (9006)

09-FEB-FRIBRIDGE BASE PAGE 1

00010000

770323

000000

.REPT 0

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DQFPB-A-D
 PRODUCT NAME: PDP-11/6X - FP11-E FLOATING POINT UNIT
 ADVANCED INSTRUCTION TESTS
 DATE: MARCH 1977
 MAINTAINER: DIAGNOSTIC GROUP
 AUTHOR: DONALD NORTH

COPYRIGHT (C) 1977
 DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS

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

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

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

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46

47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

CONTENTS

- 1. ABSTRACT
- 2. REQUIREMENTS
 - 2.1 EQUIPMENT
 - 2.2 STORAGE
 - 2.3 PRELIMINARY PROGRAMS
- 3. LOADING PROCEDURE
- 4. STARTING PROCEDURE
 - 4.1 CONTROL SWITCH SETTINGS
 - 4.2 STARTING ADDRESS
 - 4.3 PROGRAM/OPERATOR ACTION
- 5. OPERATING PROCEDURE
 - 5.1 OPERATIONAL SWITCH SETTINGS
 - 5.2 PROGRAM/OPERATOR ACTION
 - 5.3 HOT (FPI1-E) / WARM (PDP-11/6X) SELECTION
- 6. ERRORS
 - 6.1.1 ERROR MESSAGE FORMAT
 - 6.1.2 FLOATING POINT DATA FORMAT
 - 6.2 RECOVERY
 - 6.3 CAUSES
- 7. RESTRICTIONS
 - 7.1 STARTING
 - 7.2 OPERATIONAL
- 8. MISCELLANEOUS
 - 8.1 EXECUTION TIME
 - 8.2 STACK POINTER
 - 8.3 POWER FAIL
- 9. PROGRAM DESCRIPTION
 - 9.1 ORGANIZATION
 - 9.2 TEST DESCRIPTION
 - 9.3 SUBROUTINE ABSTRACTS
- 10. ACT/APT/XXDP

MAINDEC-11-DQFPB-A

PAGE 3

101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156

1. ABSTRACT

THIS PROGRAM EXTENDS THE TESTING OF INSTRUCTION FUNCTIONALITY TO THE REMAINDER OF THE PDP-11/6X FLOATING POINT INSTRUCTION SET NOT COVERED IN THE BASIC INSTRUCTION TESTS. FULL TESTING IN ALL PDP-11/6X FPU MODES OF ALL THE MULTIPLE OPERAND ARITHMETIC, COMPARISON, AND INTEGER TO FLOAT CONVERSION INSTRUCTIONS IS PERFORMED. BOTH "HOT" (FP11-E OPTION) AND "WARM" (PDP-11/6X MICROCODE) FLOATING POINT UNITS CAN BE SELECTED FOR TESTING.

2. REQUIREMENTS

2.1 EQUIPMENT

PDP-11/6X STANDARD COMPUTER WITH MINIMUM 16K OF MEMORY. OPTIONAL FP11-E FLOATING POINT UNIT, IF SELECTED.

2.2 STORAGE

THE PROGRAM USES MEMORY 0-46114(8). THE UPPER 2.0K WORDS ARE RESERVED FOR THE XXDP MONITOR, IF EMPLOYED.

2.3 PRELIMINARY PROGRAMS

THE CPU, CACHE, AND MEMORY TEST PROGRAMS MUST BE RUN FIRST TO VERIFY THE CORRECT OPERATION OF THE BASE MACHINE.

THE PDP-11/6X - FP11-E FLOATING POINT PROCESSOR INSTRUCTION SET TESTS SHOULD THEN BE RUN IN THE FOLLOWING ORDER:

- (1) DQFPA FPU BASIC INSTRUCTION TESTS
- (2) DQFPB FPU ADVANCED INSTRUCTION TESTS
- (3) DQFPC FPU INSTRUCTION EXERCISER
- (4) DQFPD FPU ADD/SUB/MUL/DIV RANDOM EXERCISER

3. LOADING PROCEDURE

USE THE STANDARD PROCEDURE FOR ABSOLUTE TAPES, OR LOAD VIA XXDP MEDIA.

4. STARTING PROCEDURE

4.1 CONTROL SWITCH SETTINGS

SEE SECTION 5.1
SWITCH REGISTER (00000) IS WORST CASE TEST.

MAINDEC-11-DQFPB-A

PAGE 4

157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211

4.2 STARTING ADDRESS

THE PROGRAM MUST ALWAYS BE STARTED AT LOCATION 200(8).

4.3 PROGRAM/OPERATOR ACTION

LOADING VIA ABSOLUTE PAPERTAPE:

- (1) LOAD PROGRAM INTO MEMORY USING ABS LOADER.
- (2) LOAD ADDRESS 200 (8).
- (3) SET SWITCHES (SEE SECTION 5.1)
SR=(000000) IS WORST CASE TEST.
- (4) PRESS CONTROL/START TO BEGIN.
- (5) PROGRAM TYPES IDENTIFICATION HEADER (VERIFY THAT THE
CORRECT PROGRAM HAS BEEN LOADED!), AND EXECUTION BEGINS.

5. OPERATING PROCEDURE

5.1 OPERATIONAL SWITCH SETTINGS

THE DEFINITION OF THE SPECIFIC BITS IN THE SWITCH REGISTER
(EITHER HARDWARE OR SOFTWARE) ARE AS FOLLOWS:

SW15=1	100000	HALT ON ERROR
SW14=1	040000	LOOP ON CURRENTLY EXECUTING TEST
SW13=1	020000	INHIBIT ERROR TYPEOUTS (WHICH IS AN "ERROR MESSAGE" RESULTING FROM AN ERROR DETECTED IN THE HARDWARE)
SW12=1	010000	INHIBIT STATUS TYPEOUTS (WHICH IS A NON-ERROR RELATED INFORMATIVE MESSAGE, SUCH AS "END PASS #XXX")
SW11=1	004000	INHIBIT ITERATIONS PER TEST
SW10	002000	SET=BELL ON ERROR/CLEAR=BELL ON PASS END
SW09=1	001000	LOOP ON ERROR
SW08=1	000400	LOOP ON TEST NUMBER IN "SLPTST" IF SET, THEN THE TEST SPECIFIED BY THE TEST NUMBER CONTAINED IN THE MEMORY WORD "SLPTST" (SEE PROGRAM LISTING) WILL SPECIFY THE DESIRED TEST ON WHICH TO LOOP.
SW01	000002	CLEAR=TEST HOT-FP/WARM-FP ALTERNATELY EACH PASS (IE, PASS#1 HFP, PASS#1 WFP, PASS#2 HFP, PASS#2 WFP, ETC)
SW00	000001	SET=TEST ONLY UNIT SPECIFIED IN SW00 SET=SELECT WARM FP, IF SW01=1 CLEAR=SELECT HOT FP, IF SW01=1

NOTE FOR SW01, SW00 - IF NO HOT FP (FP11-E) IS PRESENT,
THEN WARM FP (PDP-11/6X MICROCODE) IS AUTOMATICALLY
SELECTED.

212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267

5.2 PROGRAM/OPERATOR ACTION

ONCE EXECUTION HAS BEGUN, MINIMAL OPERATOR INTERVENTION IS REQUIRED, UNLESS THE PROGRAM DETECTS AN ERROR IN THE HARDWARE.

IF ALL IS WELL, THE PROGRAM TYPES ITS NAME UPON BEGINNING, AND AT THE START OF EACH PASS, THE CURRENT PASS NUMBER (IN OCTAL) IS ECHOED. NOTE THAT SETTING SW<12>=1 WILL INHIBIT THE TYPEOUT OF THE BEGIN AND END PASS MESSAGES.

IF SW<10>=0, THE CONSOLE BELL WILL BE RUNG AT THE END OF EACH PASS. NOTE THAT ONLY SW<10> AFFECTS THE BELL RINGING AT END OF PASS - SW<12> HAS NO EFFECT ON THIS FUNCTION.

IF AN ERROR OCCURS DURING EXECUTION, MANY VARIATIONS IN ACTION ARE POSSIBLE DEPENDING UPON THE SWITCH SETTINGS.

SW<15>=1 WILL CAUSE THE CPU TO HALT AFTER AN ERROR.
SW<13>=1 WILL ALSO INHIBIT ANY ERROR MESSAGE TYPEOUT THAT WOULD OCCUR AT THIS TIME.

SW<10>=1 WILL CAUSE THE CONSOLE BELL TO BE RUNG ONLY WHEN AN ERROR IS DETECTED (AND NOT AT THE END OF A PASS).

SW<9>=1 CAUSES THE PROGRAM TO LOOP ON THE MOST RECENT ERROR, AS LONG AS IT CONTINUES TO OCCUR.

THERE ARE ALSO SEVERAL OTHER GENERAL USE FUNCTIONS DEFINED BY THE SWITCHES:

SW<11>=1 WILL INHIBIT THE ITERATIONS (=2000(10)) PERFORMED OF EACH TEST ON PASSES 2, 3, 4 THRU THE PROGRAM.

SW<14>=1 CAUSES THE PROGRAM TO LOOP INDEFINATELY ON THE CURRENTLY EXECUTING TEST.

SW<8>=1 CAUSES THE PROGRAM TO CONTINUE EXECUTION AS NORMAL EXCEPT WHEN THE CONTENTS OF MEMORY WORD "SLPTST" MATCHES THE NUMBER OF THE TEST CURRENTLY EXECUTING. AT THIS POINT, THE TEST IS LOOPED ON INDEFINATELY, UNTIL EITHER SW<8>=0 OR "SLPTST" IS CHANGED. NOTE THAT IF "SLPTST" DOES NOT MATCH THE TEST NUMBER OF ANY TEST, THE CONTENTS OF "SLPTST" ARE EFFECTIVELY IGNORED, AND EXECUTION PROCEEDS NORMALLY.

5.3 HOT (FP11-E) / WARM (PDP-11/6X) SELECTION

WHEN THE PROGRAM IS STARTED (AT 200(8)), A MESSAGE IS OPTIONALLY PRINTED INDICATING THE PRESCENCE/ABSCENCE OF AN FP11-E HOT FLOATING POINT UNIT OPTION (BASED UPON WHETHER "WHAMI" BIT<04> IS 1/0 RESPECTIVELY).

IF NO FP11-E HOT FP OPTION IS PRESENT, THE MESSAGE IS TYPED, AND ANY ATTEMPTS TO SELECT IT FOR TESTING VIA SW01 AND SW00 ARE IGNORED. ONLY WARM FP (PDP-11/6X MICROCODE) FLOATING POINT CAN BE TESTED/SELECTED.

IF THE FP11-E IS PRESENT, TEST SELECTION IS AS FOLLOWS:

WHEN SW01=0, THE HOT AND WARM FLOATING POINT UNITS ARE TESTED

268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306

ALTERNATELY EACH PASS - IN THE ORDER (1) HOT, THEN (2) WARM.
NOTE THAT EACH "PASS" NOW CONSISTS OF TWO SEPARATE SUB-PASSES.

WHEN SW01=1, THEN DEDICATED SELECTION OF A PARTICULAR UNIT IS
SPECIFIED IN SW00:

SW00=0 --> TEST WFP FP11-E OPTION ONLY
SW00=1 --> TEST WFP PDP-11/6X MICROCODE ONLY

6. ERRORS

6.1 FORMAT OF MESSAGES

6.1.1 ALL ERROR MESSAGES CONSIST OF THREE LINES OF DATA:

THE FIRST LINE IS A BRIEF MESSAGE WHICH EXPLAINS WHAT ERROR
WAS DETECTED (EG, THE RESULT OF THE "ABSF" INSTRUCTION WAS
BAD).

THE PREFIX "HOT:" OR "WARM:" IS ALSO ATTACHED TO THE MESSAGE
TO INDICATE THE SOURCE OF THE ERROR; THE FP11-E UNIT OR THE
PDP-11/6X RESPECTIVELY.

THE SECOND LINE CONSISTS OF DATA HEADERS TO IDENTIFY THE
VALUES TYPED OUT ON LINE THREE. THESE HEADERS WILL EITHER BE
OF THE FORM "EXPECTED" AND "RECEIVED" DATA, OR WILL BE A
MNEMONIC NAME OF A WORD LOCATION IN MEMORY OR REGISTERS.

THE THIRD LINE DISPLAYS THE CONTENTS OF THE LOCATIONS
SPECIFIED BY LINE TWO AS SIX DIGIT OCTAL NUMBERS. NOTE THAT
ALL DATA DISPLAYED IN ANY MESSAGES ARE OCTAL NUMBERS.

AS EXPLAINED IN SECTION 5.2, SETTING SW<13>=1 WILL SUPPRESS
THE TYPING OF THESE MESSAGES.

307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
329
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361

6.1.2 FLOATING POINT UNIT DATA FORMATS:

FLOATING POINT STATUS WORD (FPS):

BIT#	OCTAL	FUNCTION
15	100000	FER - FLOATING ERROR FLAG SET WHEN EITHER FIUV, FIU, FIV, FIC ENABLED AND APPROPRIATE EXCEPTION OCCURRED.
14	040000	FID - FLOATING DISABLE INTERRUPTS NO FP INTERRUPTS TO VECTOR 244(8) IF SET.
13, 12		NOT USED
11	004000	FIUV - FLOATING UNDEFINED VARIABLE INTERRUPT IF SET, (-0) MEMORY DATA IS ERROR
10	002000	FIU - FLOATING INTR UNDERFLOW IF SET AND UNDERFLOW, SET FER, STORE ANSWER, EXPONENT WRONG BY +400(8)
9	001000	FIV - FLOATING OVERFLOW INTERRUPT IF SET AND OVERFLOW, SET FER, STORE ANSWER, EXPONENT WRONG BY -400(8)
8	000400	FIC - FLOATING INTEGER CONVERSION INTERRUPT IF SET AND "STCFI" ERROR, ANSWER <-- ZERO, SET ERROR
7	000200	FD - FLOATING MODE 1=DOUBLE, 64 BIT OPERANDS (4W) 0=SINGLE, 32 BIT OPERANDS (2W)
6	000100	FL - INTEGER MODE 1=LONG, 32 BIT INTEGERS (2W) 0=SHORT, 16 BIT INTEGERS (1W)
5	000040	FT - ROUND/TRUNCATE MODE 1=TRUNCATE RESULTS 0=ROUND RESULTS
4	000020	FMM - PUT FPII-E ONLY IN MAINTENANCE MODE
3:0	000017	FN-FZ-FV-FC - FLOATING CONDITION CODES

FLOATING EXCEPTION CODES (FEC):

OCTAL	ENABLE	(NOT USED)
00	(NONE)	(NOT USED)
02	(NONE)	FP OPCODE ERROR
04	(NONE)	FP DIVIDE-BY-ZERO ERROR
06	W/FIC	FP INTEGER CONVERSION ERROR
10	W/FIV	FP OVERFLOW ERROR
12	W/FIU	FP UNDERFLOW ERROR
14	W/FIUV	FP UNDEFINED-VARIABLE/(-0) ERROR
16	W/FMM	FP MAINTENANCE TRAP

NOTE - IN "FEC" CODE TYPEOUTS IN ERROR MESSAGES ONLY THE LOW ORDER BYTE IS USED - IGNORE THE PROGRAM FLAG BIT IN THE UPPER BYTE.

362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416

FLOATING POINT DATA:

IN FLOAT MODE (FD=0), IS 2-16. BIT WORDS, 32. BITS
IN DOUBLE MODE (FD=1), IS 4-16. BIT WORDS, 64. BITS

FIRST WORD: (BOTH F, D MODES)

B15=SIGN OF NUMBER (1/-, 0/+)
B14:07=EXPONENT, 8.BITS, FROM -128./+127.
B06:00=FRACTION, 7.BITS

SECOND WORD: (BOTH F, D MODES)

B15:00=FRACTION, 16.BITS

THIRD, FOURTH WORDS: (ONLY D MODE)

B15:00, B15:00=FRACTION, 32. BITS

IN F MODE, THE COMPOSITE 24. BIT FRACTION
IS FORMED BY:

.1#[WORD1-BIT<06:00>]#[WORD2-BIT<15:00>]

IN D MODE, THE COMPOSITE 56. BIT FRACTION
IS FORMED BY:

.1#[WORD1-BIT<06:00>]#[WORD2-BIT<15:00>]
#[WORD3-BIT<15:00>]#[WORD4-BIT<15:00>]

FOR A MORE DETAILED OPERATION/EXPLANATION OF FLOATING POINT
DATA FORMATS AND OPERATIONS, SEE THE PDP-11/6X PROCESSOR
HANDBOOK SECTION ON THE FLOATING POINT INSTRUCTION SET.

6.2 RECOVERY

RECOVERY FROM ERRORS HAS BEEN ATTEMPTED TO BE MADE AS
AUTOMATIC AND EFFORTLESS AS POSSIBLE. HOWEVER, IN MANY CASES,
DUE TO THE NATURE OF THE ERROR, THE PROGRAM MAY NOT EVEN BE
ABLE TO BE RUN (EG. IF THE FLOATING POINT MODULE IS IN A HUNG
STATE, AND CAN NEVER ENTER THE READY STATE TO ACCEPT A NEW FPP
INSTRUCTION). AT THIS POINT, SOLVING THE PROBLEM IS A DIRECT
FUNCTION OF THE OPERATORS' INGENUITY. THIS TEST SERIES HAS
BEEN DESIGNED TO TEST THE FLOATING POINT PROCESSOR SO THAT
THESE TYPES OF FAILURES TO RUN WILL BE MINIMAL. THE TESTS
HAVE BEEN PLACED IN A SPECIFICALLY STRUCTURED SEQUENCE IN THE
PROGRAM TO IMPLEMENT THIS STRATEGY: TESTING THE MOST BASIC
ELEMENTS FIRST, PROCEEDING UPWARD IN COMPLEXITY AFTER
ESTABLISHING THEIR CORRECT OPERATION. THIS IS WHY IT IS
EXTREMELY IMPORTANT THAT THE FLOATING POINT TEST PROGRAMS BE
(1) RUN IN THE PRESCRIBED ORDER, AND (2) ONLY BE STARTED AT
THEIR BEGINNING ADDRESS (USUALLY 200(8)). THE PROGRAM WILL
DISPLAY, AT AN ERROR, THE MOST PERTINENT INFORMATION RELATING
TO THE ERROR, AND A BRIEF EXPLANATION OF THE FAILING FUNCTION.

6.3 CAUSES

417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472

THESE TEST PROGRAMS ARE NOT HARDWARE ORIENTED, AND AS SUCH IT IS NOT POSSIBLE TO CALL OUT PARTICULAR HARDWARE AREAS AND MODULES RELATING TO A GIVEN FUNCTIONAL FAILURE. HARDWARE DIAGNOSIS FOR A PARTICULAR MACHINE MUST BE DONE USING THE APPROPRIATE ENGINEERING ROM FLOWS AND PRINTS, ALONG WITH THE KNOWN FUNCTIONAL ERRORS (AS DETECTED BY THE PROGRAMS). THIS IS THE INTENT UNDER WHICH THESE INSTRUCTION TESTS WERE DESIGNED AND CODED.

7. RESTRICTIONS

7.1 STARTING

THE PROGRAM MUST BE STARTED AT LOCATION 200(B) ALWAYS.

7.2 OPERATIONAL

THERE ARE NO OPERATIONAL RESTRICTIONS.

8. MISCELLANEOUS

8.1 EXECUTION TIME

AVERAGE EXECUTION TIME PER PASS

MODEL	SHORTEST PASS	LONGEST PASS
PDP-11/6X	1 SEC	3 MIN:00 SEC
PDP-11/6X W/FP11-E	1 SEC	X MIN:XX SEC

SEC = SECONDS / MIN = MINUTES

SHORTEST PASS ::= NO ITERATIONS, USING SWR=(004000)

LONGEST PASS ::= 2000(10) ITERATIONS/TEST, USING SWR=(000000)

8.2 STACK POINTER

THE STACK POINTER IS SET TO 1100(B) AT THE START OF EACH PASS. IF ALL IS OPERATING CORRECTLY, IT SHOULD ALSO BE THIS VALUE AT THE START OF EACH TEST, AND AT THE END OF A PASS.

8.3 POWER FAIL

THE TESTS MAY BE POWER FAILED AT ANY TIME. SPURIOUS ERROR

473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528

MESSAGES MAY OCCUR IF THE FAILURE OCCURRED WHILE THE F.P.U. WAS EXECUTING A FUNCTION, AS NONE OF ITS REGISTERS (FPS, FEC, FEA, ACCUMULATORS) ARE SAVED IN THE EVENT OF A POWER FAILURE. HOWEVER, THESE MESSAGES SHOULD ONLY OCCUR ONCE (IF AT ALL) IMMEDIATELY AFTER POWER IS RESTORED. WHEN POWER IS RESTORED, "POWER" IS TYPED ON THE CONSOLE AND EXECUTION CONTINUES WHERE IT WAS INTERRUPTED.

NOTE THAT THE "VOLATILE" SWITCH REGISTER CONTENTS ARE SAVED AND RESTORED FROM THE STACK IN A POWER FAIL SEQUENCE; THEREFORE THE SWITCH REGISTER SETTINGS SHOULD NOT BE LOST OVER A POWER FAIL.

9. PROGRAM DESCRIPTION

9.1 ORGANIZATION

THESE PROGRAMS ARE ORGANIZED AS MUCH AS POSSIBLE IN A STRAIGHTFORWARD, LINEAR MANNER. THE MAIN BODY OF CODE IS STRUCTURED AS FOLLOWS:

- (1) INITIALIZATION ROUTINE
 - SETS UP VECTORS, TYPES HEADER, ETC.
- (2) MAIN BODY OF TESTS
 - INLINE TEST CODE, INLINE TEST CALLS
- (3) END OF PASS ROUTINE
 - END OF PASS PROCESSING
- (4) TEST SUBROUTINES
 - SUBROUTINES CONTAINING COMMON TEST CODE
- (5) OVERHEAD ROUTINES
 - SERVICE SUBROUTINES (TYPEOUT, ETC.)

WHEREVER FEASIBLE, COMMON SECTIONS OF CODE FOR WIDELY USED FUNCTIONS ARE CONDENSED INTO SUBROUTINES TO CONSERVE MEMORY. THIS INCLUDES NOT ONLY STANDARD SERVICE ROUTINES (SUCH AS SCOPE, ERROR, AND ASCII TYPEOUT), BUT ALSO TESTING ROUTINES WHICH PERFORM VERY SIMILAR FUNCTIONS. THUS IN MANY CASES (THE "A0DF" INSTRUCTION TESTING, FOR EXAMPLE) A SINGLE BODY OF CODE (A SUBROUTINE) IS USED TO PERFORM ALL THE FUNCTIONAL TESTS, WITH A VARIABLE PARAMETER LIST PASSED AT EACH CALL CONTAINING THE DATA OPERANDS AND EXPECTED RESULT FOR EACH INDIVIDUAL TEST. THIS CONSTRUCTION FACILITATES THE ADDITION/DELETION OF TESTS (SHOULD THAT EVER BE NECESSARY), AND ALSO GREATLY CONSERVES MEMORY SPACE REQUIREMENTS WHEN A LARGE NUMBER OF CALLS TO A GIVEN BODY OF CODE ARE REQUIRED.

THE INDIVIDUAL TESTS WITHIN EACH PROGRAM HAVE ALSO BEEN SEQUENCED IN A PARTICULAR ORDER TO FACILITATE THE DETECTION AND RESOLUTION OF ERRORS AS QUICKLY AS POSSIBLE. EACH OF THE TESTS BEGINS AS SIMPLY AS POSSIBLE, FIRST TESTING THE MOST

529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584

BASIC ELEMENTS. MORE COMPLEX ELEMENTS ARE TESTED AFTERWARDS, EMPLOYING A PHILOSOPHY THAT THE SIMPLER THE TEST, THE BETTER THE RESOLUTION. ALL FUNCTIONS ARE EVENTUALLY TESTED, BUT HOPEFULLY MOST ERRORS WILL BE CAUGHT AND CORRECTED EARLY. A MUCH MORE DETAILED ANALYSIS OF THE SEQUENCE OF TESTS PERFORMED IS PRESENTED IN SECTION 9.2.

9.2 TEST DESCRIPTION

THIS DIAGNOSTIC IS STRUCTURED TO SEQUENTIALLY PERFORM THE FOLLOWING SERIES OF TESTS ON THE FUNCTIONALITY OF THE FLOATING POINT DUAL OPERAND INSTRUCTIONS:

- (1) 'CMP-' COMPARE, F/D MODES
- (2) 'ADD-' ADD, F/D MODES
- (3) 'SUB-' SUBTRACT, F/D MODES
- (4) 'MUL-' MULTIPLY, F/D MODES
- (5) 'DIV-' DIVIDE, F/D MODES
- (6) 'MOD-' MODULO, F/D MODES, 2 ACCUMULATORS
- (7) 'MOD-' MODULO, F/D MODES, 1 ACCUMULATOR
- (8) 'LDC--' LOAD-CONVERT, F (->) D MODES
- (9) 'STC--' STORE-CONVERT, F (->) D MODES
- (10) 'LDC--' LOAD-CONVERT, I-F/I-D/L-F/L-D MODES
- (11) 'STC--' STORE-CONVERT, F-I/D-I/F-L/D-L MODES
- (12) 'LDEXP' LOAD EXPONENT, F/D MODES
- (13) 'STEXP' STORE EXPONENT, F/D MODES

EACH OF THE ABOVE TESTS IS PERFORMED BY A SUBROUTINE SPECIFIC TO THE INSTRUCTION. AN ARGUMENT LIST IS PASSED CONTAINING THE INITIAL DATA, EXPECTED RESULTS/STATUS/EXCEPTIONS.

EACH OF THE ABOVE INSTRUCTIONS IS TESTED IN (WHEN APPLICABLE) THE FOLLOWING INSTANCES:

- (A) FLOATING(F)/DOUBLE(D) MODES
- (B) INTEGER(I)/LONG(L) MODES
- (C) ROUND(R)/TRUNCATE(T) MODES
- (D) EXCEPTION CONDITIONS:
OVERFLOW, UNDERFLOW, -0, DIVIDE/0, INTEGER-CONVERT
(ENABLED AND DISABLED MODES)

9.3 SUBROUTINE ABSTRACTS

9.3.1 TRAPCATCHER

THE TRAPCATCHER IS A SERIES OF INSTRUCTIONS OCCUPYING THE INTERRUPT VECTOR AREA OF MEMORY. IT CONSISTS OF THE SEQUENCE:

```
.WORD +2 ;PC AFTER TRAP
.WORD 0 ;PS AFTER TRAP
```

PLACED AT EACH VECTOR ADDRESS IN LOCATIONS 4-776(8) OF MEMORY.

585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640

THE FIRST WORD OF EACH PAIR ("PC AFTER TRAP") POINTS TO THE SECOND WORD, WHICH SERVES A DUAL PURPOSE AS (1) THE NEW LOADED PS (ALL ZEROS), AND (2) THE NEXT INSTRUCTION TO EXECUTE (0=HALT).

WHEN THE PROGRAM IS EXECUTING, ANY REQUIRED VECTORS ARE SET UP IN THE VECTOR AREA WITH APPROPRIATE VALUES. THE OTHERS BEING LEFT IN THE "TRAPCATCHER" STATE. THUS, IF AN UNEXPECTED TRAP EVER OCCURS IN THE MACHINE, IT WILL BE CAUGHT, AND THE MACHINE SUBSEQUENTLY HALTED, DISPLAYING THE VECTOR ADDRESS * PLUS FOUR * IN THE ADDRESS LIGHTS.

9.3.2 SCOPE ROUTINE - \$SCOPE

THE SCOPE ROUTINE IS ENTERED FROM THE FIRST INSTRUCTION OF EACH TEST IN THE PROGRAM. (NOTE THAT BY DEFINITION, A "TEST" WILL BE DESIGNATED AS THE SECTION OF CODE BETWEEN TWO "SCOPE" STATEMENTS.) THIS ROUTINE PROVIDES THE OVERHEAD CODE NECESSARY TO IMPLEMENT SEVERAL OF THE SWITCH REGISTER CONTROL OPTIONS. UPON ENTRANCE TO A TEST, THE SCOPE STATEMENT AT THE BEGINNING SETS UP CERTAIN LOCATIONS (SEE BELOW) TO SPECIFY THE CURRENT TEST NUMBER AND LOOPING ADDRESS (FOR ITERATIONS). CONTROL IS THEN PASSED TO THE ACTUAL TEST CODE, PERFORMING THE DESIRED TEST. UPON EXIT, THE SCOPE STATEMENT OF THE NEXT TEST IS ENTERED, WHICH DETERMINES WHETHER TO (1) LOOP BACK TO THE PREVIOUS TEST (EG. FOR ITERATIONS) OR (2) INITIALIZE FOR THE NEXT TEST (AS DESCRIBED EARLIER, ABOVE).

ENTRANCE TO THE SCOPE ROUTINE IS VIA AN "IOT" TRAP CALL THROUGH LOCATION 20(8). (FROM THE SCOPE=IOT EQUATE). DEPENDING UPON THE SWITCH SETTINGS (SEE 5.2) CODE IS PRESENT TO: LOAD THE FPII MICRO BREAK REGISTER, LOOP ON THE CURRENTLY EXECUTING TEST, LOOP ON A SPECIFIC TEST, PERFORM ITERATIONS OF EACH TEST, AND SET UP ADDRESSES FOR POSSIBLE LOOPING ON ERRORS. IMPORTANT VALUES USED IN THIS ROUTINE ARE:

- \$MXCNT - MAXIMUM NUMBER OF ITERATIONS PER TEST (GENERALLY WILL BE 2000(10))
- \$STSTM - A COUNTER INDICATING THE NUMBER (1-377(8)) OF THE TEST CURRENTLY BEING EXECUTED
- \$LPADR - CONTAINS THE ADDRESS TO WHICH THE SCOPE ROUTINE 10200 WILL LOOP, IF THE CURRENT TEST IS BEING LOOPED UPON
- \$LPERR - CONTAINS THE ADDRESS TO WHICH THE ERROR ROUTINE (SEE 9.3.3) WILL LOOP, IF AN ERROR OCCURS AND THE LOOPING ON AN ERROR OPTION IS SPECIFIED IN THE SWITCHES. SET UP BY SCOPE, GENERALLY WILL BE THE SAME AS \$LPADR, ABOVE.

9.3.3 ERROR ROUTINE - \$ERROR

THE ERROR ROUTINE IS ENTERED WHEN THE TEST CODE HAS DETERMINED THAT AN ERROR HAS OCCURRED AS PART OF A TEST. THROUGH USE OF

641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695

THIS ROUTINE, THE TEST HAS A MEANS OF SIGNALING AN ERROR TO THE 10380 OPERATOR/MONITOR; AND IMPLEMENTING THE CONTROL FUNCTIONS FOR HALTING ON ERROR, BELL ON ERROR, AND LOOPING ON ERROR. IN ADDITION, THE ERROR ROUTINE HAS THE PROVISION TO TYPE OUT ON THE OPERATOR'S CONSOLE A MESSAGE BRIEFLY EXPLAINING THE ERROR, AND SOME OF THE MOST PERTINENT DATA VALUES TO HELP DIAGNOSE THE CAUSE (SEE SECTION 6.2).

THE CALLING MECHANISM IS SIMILAR TO THAT EMPLOYED FOR THE SCOPE ROUTINE (VIA A TRAP), EXCEPT IN THIS INSTANCE, THE "EMT" INSTRUCTION IS USED, TRAPPING THROUGH LOCATION 30(8). (NOTE THE EQUATE ERROR N=EMT N). THE LOWER BYTE OF THE EMT INSTRUCTION IS CAPABLE OF TRANSMITTING A NUMBER FROM 0-377(8), WHICH WILL BE TERMED THE "ERROR ITEM NUMBER." THIS NUMBER DETERMINES WHICH ERROR MESSAGE, AND ASSOCIATED DATA VALUES WILL BE TYPED OUT WHEN A PARTICULAR ERROR IS SIGNALLED. IF THIS NUMBER IS ZERO, JUST THE PC OF THE CALLING "ERROR" INSTRUCTION WILL BE TYPED, OTHERWISE, THE NUMBER IS USED AS AN INDEX THROUGH THE ERROR TABLE (SERRTB) TO FIND THE APPROPRIATE VALUES TO TYPE (SEE PROGRAM LISTING FOR FURTHER DETAILS).

IMPORTANT VALUES USED IN THIS ROUTINE ARE:

EREG0 THRU EREG7 - CONTENTS OF GENERAL REGISTERS R0 THRU R7 JUST BEFORE ERROR CALL
SERTTL - CUMULATIVE NUMBER OF ERRORS ENCOUNTERED TO DATE
SERRPC - CONTAINS THE PC OF THE "ERROR" INSTRUCTION JUST EXECUTED
SLPERR - CONTAINS THE ADDRESS WHICH WILL BE LOOPED UPON FOR THE ERROR LOOPING FACILITY

9.3.4 ERROR MESSAGE TYPEOUT ROUTINE - \$TYPERR

THIS ROUTINE (\$TYPERR ENTRY POINT) IS CALLED BY THE ERROR PROCESSING ROUTINE DESCRIBED IN 9.3.3 ABOVE. ITS PURPOSE IS TO IMPLEMENT THE ERROR MESSAGE/DATA VALUE ERROR TYPEOUT FACILITY. THE SUBROUTINE WILL, GIVEN THE INDEXING BYTE FROM THE ERROR CALL INSTRUCTION, PICK UP THE CORRECT ERROR MESSAGE VECTOR FROM SERRTB (ERROR TABLE), AND TYPE OUT THE ERROR MESSAGE, DATA HEADER, AND DATA VALUES ON THE CONSOLE.

9.3.5 TYPE ROUTINE - \$TYPE

THIS ROUTINE IS THE STANDARD SYSTEM TYPEOUT ROUTINE FOR ASCII SINGLE-CHARACTER-PER-BYTE STRINGS. IT IS CALLED THROUGH A TRAP INSTRUCTION WITH THE NEXT WORD CONTAINING THE ADDRESS OF THE FIRST CHARACTER IN THE STRING. TYPING TERMINATES WHEN AN ALL-ZERO BYTE IS FOUND. HORIZONTAL TAB STOPS ARE ALSO AUTOMATICALLY PLACED.

9.3.6 OCTAL NUMBER TYPE ROUTINE - \$TYPOC

MAINDEC-11-DQFPB-A

PAGE 14

696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742

THIS ROUTINE CONVERTS THE TOP NUMBER ON THE STACK TO A 6-DIGIT OCTAL REPRESENTATION, AND TYPES IT ON THE CONSOLE USING THE TYPE ROUTINE STYPE. SEE LISTING FOR OPTIONS AND FURTHER DETAILS.

9.3.7 POWER UP AND DOWN ROUTINES - SPWRUP AND SPWRDN

THESE TWO ROUTINES ARE ENTERED FOR THE POWER UP AND DOWN CONDITIONS, RESPECTIVELY. THE POWER DOWN ROUTINE (SPWRDN) SAVES THE GENERAL REGISTERS AND STACK POINTER. THE POWER UP ROUTINE (SPWRUP) CORRESPONDINGLY RESTORES THE REGISTERS, STACK POINTER, AND TYPES THE MESSAGE "POWER" WHEN POWER IS RESTORED. THE VOLATILE INTERNAL SWITCH REGISTER IS ALSO SAVED/RESTORED BY THIS ROUTINE.

9.3.8 END OF PASS ROUTINE - SEOP

THE END OF PASS ROUTINE COUNTS THE NUMBER OF PASSES PERFORMED, DINGS THE BELL/TYPES A MESSAGE (IF ENABLED), SETS/CLEARs THE T-BIT (IF ENABLED), AND ALSO INTERFACES TO THE MONITOR, IF PRESENT. IT ALSO OPTIONALLY LOOPS FOR A NUMBER OF SUBPASSES BEFORE SIGNALLING AN END OF PASS CONDITION.

10. ACT/APT/XXDP

10.1 ACT COMPATIBILITY

THIS PROGRAM WILL RUN UNDER THE ACT SYSTEM.

10.2 APT COMPATIBILITY

THIS PROGRAM WILL RUN UNDER THE APT SYSTEM MONITOR. ALL NECESSARY SOFTWARE COMMUNICATION HOOKS ARE PRESENT.

10.3 XXDP COMPATIBILITY

FOR XXDP MEDIA COMPATIBILITY, THE TOP 2K WORDS OF THE 16K WORD MINIMUM MEMORY AREA ARE NOT DISTURBED DURING EXECUTION.

.ENDR

743
 744
 745
 746
 747
 748
 749
 750
 751
 752
 753
 754
 755
 756
 757
 758
 759
 760
 761
 762
 763
 764
 765
 766
 767
 768
 769
 770
 771
 772
 773
 774
 775
 776
 777
 778
 779
 780
 781
 782
 783
 784
 785
 786
 787
 788
 789
 790
 791
 792
 793
 794
 795
 796
 797
 798

```

.TITLE FPU ADVANCED INSTR TESTS
*COPYRIGHT (C) 1976
*DIGITAL EQUIPMENT CORP.
*MAYNARD, MASS. 01754
*
*PROGRAM BY DONALD NORTH
*
*THIS PROGRAM WAS ASSEMBLED USING THE PDP-11 MAINDEC SYSMAC
*PACKAGE (MAINDEC-11-DZQAC-C3), JAN 19, 1977.
*

.SBTTL OPERATIONAL SWITCH SETTINGS
*
*      SWITCH  OCTAL          USE
*      -----  -----  -----
*      15      100000      HALT ON ERROR
*      14      040000      LOOP ON CURRENTLY EXECUTING TEST
*      13      020000      INHIBIT ERROR TYPEOUTS
*      12      010000      INHIBIT STATUS TYPEOUTS
*      11      004000      INHIBIT ITERATIONS
*      10      000000      0=BELL ON PASS END
*                       1=BELL ON ERROR
*      9       001000      LOOP ON ERROR
*      8       000400      LOOP ON TEST NUMBER IN "SLPTST"
*      1       000000      0=TEST HFP/WFP ALTERNATELY EACH PASS
*                       1=TEST ONLY UNIT SPECIFIED IN SW<00>
*      0       000002      0=SELECT HFP, IF SW<01>=1
*                       1=SELECT WFP, IF SW<01>=1
*

.SBTTL BASIC DEFINITIONS
*INITIAL ADDRESS OF THE STACK POINTER *** 1100 ***
STACK= 1100
.EQUIV EMT,ERROR      ;;BASIC DEFINITION OF ERROR CALL
.EQUIV IOT,SCOPE      ;;BASIC DEFINITION OF SCOPE CALL

*MISCELLANEOUS DEFINITIONS
HT= 11      ;;CODE FOR HORIZONTAL TAB
LF= 12      ;;CODE FOR LINE FEED
CR= 15      ;;CODE FOR CARRIAGE RETURN
CRLF= 200   ;;CODE FOR CARRIAGE RETURN-LINE FEED
PS= 177776  ;;PROCESSOR STATUS WORD
.EQUIV PS,PSW
STKLMT= 177774 ;;STACK LIMIT REGISTER
PIRQ= 177772  ;;PROGRAM INTERRUPT REQUEST REGISTER
DSWR= 177570  ;;HARDWARE SWITCH REGISTER
DDISP= 177570 ;;HARDWARE DISPLAY REGISTER

*GENERAL PURPOSE REGISTER DEFINITIONS
R0= %0      ;;GENERAL REGISTER
R1= %1      ;;GENERAL REGISTER
R2= %2      ;;GENERAL REGISTER
R3= %3      ;;GENERAL REGISTER
R4= %4      ;;GENERAL REGISTER
R5= %5      ;;GENERAL REGISTER
  
```

001100
 000011
 000012
 000015
 000200
 177776
 177774
 177772
 177570
 177570
 000000
 000001
 000002
 000003
 000004
 000005

799 000006
800 000007
801 000006
802 000007
803
804

R6= %6 ;: GENERAL REGISTER
R7= %7 ;: GENERAL REGISTER
SP= %6 ;: STACK POINTER
PC= %7 ;: PROGRAM COUNTER

805 000000
806 000040
807 000100
808 000140
809 000200
810 000240
811 000300
812 000340
813

: *PRIORITY LEVEL DEFINITIONS
PR0= 0 ;: PRIORITY LEVEL 0
PR1= 40 ;: PRIORITY LEVEL 1
PR2= 100 ;: PRIORITY LEVEL 2
PR3= 140 ;: PRIORITY LEVEL 3
PR4= 200 ;: PRIORITY LEVEL 4
PR5= 240 ;: PRIORITY LEVEL 5
PR6= 300 ;: PRIORITY LEVEL 6
PR7= 340 ;: PRIORITY LEVEL 7

814
815 100000
816 040000
817 020000
818 010000
819 004000
820 002000
821 001000
822 000400
823 000200
824 000100
825 000040
826 000020
827 000010
828 000004
829 000002
830 000001
831

: *"SWITCH REGISTER" SWITCH DEFINITIONS
SW15= 100000
SW14= 40000
SW13= 20000
SW12= 10000
SW11= 4000
SW10= 2000
SW09= 1000
SW08= 400
SW07= 200
SW06= 100
SW05= 40
SW04= 20
SW03= 10
SW02= 4
SW01= 2
SW00= 1
.EQUIV SW09, SW9
.EQUIV SW08, SW8
.EQUIV SW07, SW7
.EQUIV SW06, SW6
.EQUIV SW05, SW5
.EQUIV SW04, SW4
.EQUIV SW03, SW3
.EQUIV SW02, SW2
.EQUIV SW01, SW1
.EQUIV SW00, SW0

841
842
843 100000
844 040000
845 020000
846 010000
847 004000
848 002000
849 001000
850 000400
851 000200
852 000100
853 000040
854 000020

: *DATA BIT DEFINITIONS (BIT00 TO BIT15)
BIT15= 100000
BIT14= 40000
BIT13= 20000
BIT12= 10000
BIT11= 4000
BIT10= 2000
BIT09= 1000
BIT08= 400
BIT07= 200
BIT06= 100
BIT05= 40
BIT04= 20

855 000010
 856 000004
 857 000002
 858 000001
 859
 860
 861
 862
 863
 864
 865
 866
 867
 868
 869
 870
 871 000004
 872 000010
 873 000014
 874 000014
 875 000014
 876 000020
 877 000024
 878 000030
 879 000034
 880 000060
 881 000064
 882 000240
 883
 884
 885 076600
 886
 887 000022
 888
 889 000144
 890 000344
 891
 892
 893 000244
 894
 895
 896 000000
 897 000001
 898 000002
 899 000003
 900 000004
 901 000005
 902
 903
 904 052525
 905 052525
 906 125252
 907 125252
 908 007417
 909 170360
 910 177776

BIT03= 10
 BIT02= 4
 BIT01= 2
 BIT00= 1
 .EQUIV BIT09,BIT9
 .EQUIV BIT08,BIT8
 .EQUIV BIT07,BIT7
 .EQUIV BIT06,BIT6
 .EQUIV BIT05,BIT5
 .EQUIV BIT04,BIT4
 .EQUIV BIT03,BIT3
 .EQUIV BIT02,BIT2
 .EQUIV BIT01,BIT1
 .EQUIV BIT00,BIT0

.*BASIC "CPU" TRAP VECTOR ADDRESSES
 ERRVEC= 4 ; TIME OUT AND OTHER ERRORS
 RESVEC= 10 ; RESERVED AND ILLEGAL INSTRUCTIONS
 TRITVEC= 14 ; "I" BIT
 TRIVEC= 14 ; TRACE TRAP
 BPTVEC= 14 ; BREAKPOINT TRAP (BPT)
 IOTVEC= 20 ; INPUT/OUTPUT TRAP (IOT) **SCOPE**
 PWRVEC= 24 ; POWER FAIL
 EMTVEC= 30 ; EMULATOR TRAP (EMT) **ERROR**
 TRAPVEC= 34 ; "TRAP" TRAP
 TKVEC= 60 ; TTY KEYBOARD VECTOR
 TPVEC= 64 ; TTY PRINTER VECTOR
 PIRQVEC= 240 ; PROGRAM INTERRUPT REQUEST VECTOR

.*MED CODES
 MED= 076600 ; OPCODE
 RWHAMI= 022 ; READ WHAMI
 RFLAG= 144 ; READ FLAGS
 WFLAG= 344 ; WRITE FLAGS

.*FLOATING POINT INTERRUPT VECTOR
 FPPVEC= 244

.*FLOATING POINT REGISTER DEFINITIONS
 AC0= %0
 AC1= %1
 AC2= %2
 AC3= %3
 AC4= %4
 AC5= %5

.*BIT PATTERNS FOR TESTS
 ALTP= 052525 ; 0101...01
 AP= ALTP
 ALTN= 125252 ; 1010...10
 AN= ALTN
 ALT4P= 007417 ; 0000111100001111
 ALT4N= 170360 ; 1111000011110000
 M2= 177776 ; 1111...10 MINUS TWO

911 177777
912 100000
913 077777
914 177777
915 000200
916 100200
917 000177
918 100177
919 040200
920 140200
921 104210
922 000377
923 177400
924
925

M1= 177777
M0= 100000
LGP= 077777
LGN= 177777
SMP= 000200
SMN= 100200
ZXIMP= 000177
ZXIMN= 100177
FIP= 040200
FIN= 140200
P13Z= 104210
LB= 000377
UB= 177400

; 1111...11 MINUS ONE, ALL 1'S
; 1000...00 MINUS ZERO
; 0111...11 LGST + NUM (1ST WD FLT)
; 1111...11 LGST - NUM (1ST WD FLT)
; +1*2**-128, SMLT + NUM (1ST WD FLT)
; -1*2**-128, SMLT - NUM (1ST WD FLT)
; ZERO EXP, ALL 1-S MANT (1ST WD FLT)
; ZERO EXP, ALL 1-S MANT (1ST WD FLT)
; +1.0E+0, 1ST WD FLT
; -1.0E+0, 1ST WD FLT
; 1000100010001000
; 0000000011111111 LOWER BYTE
; 1111111100000000 UPPER BYTE

926 147757
927 000000
928 000000
929
930
931 177760
932
933
934
935

;
#FPS BIT PATTERNS
FPS1= 147757
FPS0= 000000
NA= 000000

; ALL BITS ON (READABLE)
; ALL BITS OFF
; FOR FEC, WHEN NOT APPLICABLE

;
#PSW BIT PATTERNS
CCONLY= 177760

; FOR BIC TO GET CC BITS ONLY

.SBTTL TRAP CATCHER

936 000000
937
938
939
940 000174
941 000174 000000
942 000176 000000
943
944 000200 000137 002162
945
946
947
948
949

.=0
;*ALL UNUSED LOCATIONS FROM 4 - 776 CONTAIN A ".+2,HALT"
;*SEQUENCE TO CATCH ILLEGAL TRAPS AND INTERRUPTS
;*LOCATION 0 CONTAINS 0 TO CATCH IMPROPERLY LOADED VECTORS
.=174
DISPREG: .WORD 0 ;; SOFTWARE DISPLAY REGISTER
SWREG: .WORD 0 ;; SOFTWARE SWITCH REGISTER
.SBTTL STARTING ADDRESS(ES)
JMP @#START ;; JUMP TO STARTING ADDRESS OF PROGRAM

.SBTTL ACT11 HOOKS

950 000204
951 000046
952 000046 033062
953 000052
954 000052 000000
955 000204
956 001000
957
958
959

;; *****
;HOOKS REQUIRED BY ACT11
\$SVPC= . ;SAVE PC
.=46
\$ENDAD ;; 1)SET LOC.46 TO ADDRESS OF \$ENDAD IN .SEOP
.=52
.WORD 0 ;; 2)SET LOC.52 TO ZERO
.= \$SVPC ;; RESTORE PC
.=1000

.SBTTL APT PARAMETER BLOCK

960
961
962 001000
963 000024
964 000024 000200
965 000044
966 000044 001000

;; *****
;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
;; *****
.\$X= . ;; SAVE CURRENT LOCATION
.=24 ;; SET POWER FAIL TO POINT TO START OF PROGRAM
200 ;; FOR APT START UP
.=44 ;; POINT TO APT INDIRECT ADDRESS PNTR.
\$APTHDR ;; POINT TO APT HEADER BLOCK

967	001000	
968		
969		
970		
971		
972	001000	
973	001000	000000
974	001002	001324
975	001004	000001
976	001006	000001
977	001010	000000
978	001012	000014
979		

```

      .=.SX ;;RESET LOCATION COUNTER
;*****
;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
;INTERFACE SPEC.

$APTHD:
$HIBTS: .WORD 0 ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
$MBAOR: .WORD $MAIL ;;ADDRESS OF APT MAILBOX (BITS 0-15)
$STSM: .WORD 1 ;;RUN TIM OF LONGEST TEST
$PASTM: .WORD 1 ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
$UNITM: .WORD 0 ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
      .WORD $ETEND-$MAIL/2 ;;LENGTH MAILBOX-ETABLE(WORDS)

```

.SBTTL COMMON TAGS

*THIS TABLE CONTAINS VARIOUS COMMON STORAGE LOCATIONS
*USED IN THE PROGRAM.

980
981
982
983
984
985
986
987 001100 001100
988 001100 000000
989 001102 000000
990 001104 000000
991 001106 000000
992 001110 000000
993 001112 000000
994 001114 000000
995 001116 000000
996 001120 000000
997 001122 000001
998 001124 000000
999 001126 000000
1000 001130 000000
1001 001132 000000
1002 001134 000000
1003 001136 000000
1004 001140 000000
1005 001142 000
1006 001143 000
1007 001144 000000
1008 001146 177570
1009 001150 177570
1010 001:52 177560
1011 001:54 177562
1012 001156 177564
1013 001160 177566
1014 001162 000
1015 001163 002
1016 001164 012
1017 001165 000
1018 001166 000000
1019
1020 001170 000000
1021 001172 000000
1022 001174 000000
1023 001176 000000
1024 001200 000000
1025 001202 000000
1026 001204 000000
1027 001206 000000
1028 001210 000000
1029 001212 000000
1030 001214 000000
1031 001216 000000
1032 001220 000000
1033 001222 000000
1034 001224 000000
1035 001226 000000

.SMTAG: . =1100
\$STNM: .WORD 0
\$ERFLG: .WORD 0
\$ICNT: .WORD 0
\$LPADR: .WORD 0
\$LPST: .WORD 0
\$LPERR: .WORD 0
\$ERTTL: .WORD 0
\$ITEMB: .WORD 0
\$ERMAX: .WORD 1
\$ERRPC: .WORD 0
\$GDADR: .WORD 0
\$BDADR: .WORD 0
\$GDADR: .WORD 0
\$BDADR: .WORD 0
\$AUTOB: .BYTE 0
\$INTAG: .BYTE 0
\$SWR: .WORD DSWR
DISPLAY: .WORD DDISP
\$TKS: 177560
\$TKB: 177562
\$TPS: 177564
\$TPB: 177566
\$NULL: .BYTE 0
\$FILLS: .BYTE 2
\$FILLC: .BYTE 12
\$TPFLG: .BYTE 0
\$REGAD: .WORD 0
\$REG0: .WORD 0
\$REG1: .WORD 0
\$REG2: .WORD 0
\$REG3: .WORD 0
\$REG4: .WORD 0
\$REG5: .WORD 0
\$REG6: .WORD 0
\$REG7: .WORD 0
\$REG10: .WORD 0
\$REG11: .WORD 0
\$REG12: .WORD 0
\$REG13: .WORD 0
\$REG14: .WORD 0
\$REG15: .WORD 0
\$REG16: .WORD 0
\$REG17: .WORD 0

;; START OF COMMON TAGS
CONTAINS THE TEST NUMBER
CONTAINS ERROR FLAG
CONTAINS SUBTEST ITERATION COUNT
CONTAINS SCOPE LOOP ADDRESS
CONTAINS TEST NUMBER TO LOOP UPON
CONTAINS SCOPE RETURN FOR ERRORS
CONTAINS TOTAL ERRORS DETECTED
CONTAINS ITEM CONTROL BYTE
CONTAINS MAX. ERRORS PER TEST
CONTAINS PC OF LAST ERROR INSTRUCTION
CONTAINS ADDRESS OF 'GOOD' DATA
CONTAINS ADDRESS OF 'BAD' DATA
CONTAINS 'GOOD' DATA
CONTAINS 'BAD' DATA
RESERVED--NOT TO BE USED
AUTOMATIC MODE INDICATOR
INTERRUPT MODE INDICATOR
ADDRESS OF SWITCH REGISTER
ADDRESS OF DISPLAY REGISTER
TTY KBD STATUS
TTY KBD BUFFER
TTY PRINTER STATUS REG. ADDRESS
TTY PRINTER BUFFER REG. ADDRESS
CONTAINS NULL CHARACTER FOR FILLS
CONTAINS # OF FILLER CHARACTERS REQUIRED
INSERT FILL CHARS. AFTER A "LINE FEED"
"TERMINAL AVAILABLE" FLAG (BIT<07>=0=YES)
CONTAINS THE ADDRESS FROM WHICH (\$REG0) WAS OBTAINED
CONTAINS ((\$REGAD)+0)
CONTAINS ((\$REGAD)+2)
CONTAINS ((\$REGAD)+4)
CONTAINS ((\$REGAD)+6)
CONTAINS ((\$REGAD)+10)
CONTAINS ((\$REGAD)+12)
CONTAINS ((\$REGAD)+14)
CONTAINS ((\$REGAD)+16)
CONTAINS ((\$REGAD)+20)
CONTAINS ((\$REGAD)+22)
CONTAINS ((\$REGAD)+24)
CONTAINS ((\$REGAD)+26)
CONTAINS ((\$REGAD)+30)
CONTAINS ((\$REGAD)+32)
CONTAINS ((\$REGAD)+34)
CONTAINS ((\$REGAD)+36)

1036	001230	000000	STMP0:	.WORD	0	:: USER DEFINED
1037	001232	000000	STMP1:	.WORD	0	:: USER DEFINED
1038	001234	000000	STMP2:	.WORD	0	:: USER DEFINED
1039	001236	000000	STMP3:	.WORD	0	:: USER DEFINED
1040	001240	000000	STMP4:	.WORD	0	:: USER DEFINED
1041	001242	000000	STMP5:	.WORD	0	:: USER DEFINED
1042	001244	000000	STMP6:	.WORD	0	:: USER DEFINED
1043	001246	000000	STMP7:	.WORD	0	:: USER DEFINED
1044	001250	000000	STMP10:	.WORD	0	:: USER DEFINED
1045	001252	000000	STMP11:	.WORD	0	:: USER DEFINED
1046	001254	000000	STMP12:	.WORD	0	:: USER DEFINED
1047	001256	000000	STMP13:	.WORD	0	:: USER DEFINED
1048	001260	000000	STMP14:	.WORD	0	:: USER DEFINED
1049	001262	000000	STMP15:	.WORD	0	:: USER DEFINED
1050	001264	000000	STMP16:	.WORD	0	:: USER DEFINED
1051	001266	000000	STMP17:	.WORD	0	:: USER DEFINED
1052	001270	000000	STMP20:	.WORD	0	:: USER DEFINED
1053	001272	000000	STMP21:	.WORD	0	:: USER DEFINED
1054	001274	000000	STMP22:	.WORD	0	:: USER DEFINED
1055	001276	000000	STMP23:	.WORD	0	:: USER DEFINED
1056	001300	000000	STMP24:	.WORD	0	:: USER DEFINED
1057	001302	000000	STMP25:	.WORD	0	:: USER DEFINED
1058	001304	000000	STMP26:	.WORD	0	:: USER DEFINED
1059	001306	000000	STMP27:	.WORD	0	:: USER DEFINED
1060	001310	000000	STIMES:	0		MAX. NUMBER OF ITERATIONS
1061	001312	000000	\$ESCAPE:	0		ESCAPE ON ERROR ADDRESS
1062	001314	177607	\$BELL:	.ASCIZ	<207><377><377>	CODE FOR BELL
1063	001320	077	\$QUES:	.ASCII	/??	QUESTION MARK
1064	001321	015	\$CRLF:	.ASCII	<15>	CARRIAGE RETURN
1065	001322	000012	\$LF:	.ASCIZ	<12>	LINE FEED
1066			:*****			
1067			.SBTTL APT MAILBOX-ETABLE			
1068			:*****			
1069			.EVEN			
1070			:*****			
1071	001324		\$MAIL:			APT MAILBOX
1072	001324	000000	\$MSGTY:	.WORD	AMSGTY	MESSAGE TYPE CODE
1073	001326	000000	\$FATAL:	.WORD	AFATAL	FATAL ERROR NUMBER
1074	001330	000000	\$TESTN:	.WORD	ATESTN	TEST NUMBER
1075	001332	000000	\$PASS:	.WORD	APASS	PASS COUNT
1076	001334	000000	\$DEVCT:	.WORD	ADEVCT	DEVICE COUNT
1077	001336	000000	\$UNIT:	.WORD	ALUNIT	I/O UNIT NUMBER
1078	001340	000000	\$MSGAD:	.WORD	AMSGAD	MESSAGE ADDRESS
1079	001342	000000	\$MSGLG:	.WORD	AMSGLG	MESSAGE LENGTH
1080	001344		\$ETABLE:			APT ENVIRONMENT TABLE
1081	001344	000	\$ENV:	.BYTE	AENV	ENVIRONMENT BYTE
1082	001345	000	\$ENVM:	.BYTE	AENVM	ENVIRONMENT MODE BITS
1083	001346	000000	\$SWREG:	.WORD	ASWREG	APT SWITCH REGISTER
1084	001350	000000	\$USWR:	.WORD	AUSWR	USER SWITCHES
1085	001352	000000	\$CPUOP:	.WORD	ACPUOP	CPU TYPE, OPTIONS
1086			:*			
1087			:*			
1088			:*			
1089			:*			
1090			:*			
1091			:*			

BITS 15-11=CPU TYPE
 11/04=01, 11/05=02, 11/20=03, 11/40=04, 11/45=05
 11/70=06, PDQ=07, Q=10
 BIT 10=REAL TIME CLOCK
 BIT 9=FLOATING POINT PROCESSOR
 BIT 8=MEMORY MANAGEMENT

J02

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 24
APT MAILBOX-ETABLE

1092 001354
1093

SETEND:
.MEXIT

.SBTTL ERROR POINTER TABLE

\$ERRTB:

;*THIS TABLE CONTAINS THE INFORMATION FOR EACH ERROR THAT CAN OCCUR.
;*THE INFORMATION IS OBTAINED BY USING THE INDEX NUMBER FOUND IN
;*LOCATION \$ITEMB. THIS NUMBER INDICATES WHICH ITEM IN THE TABLE IS PERTINENT.
;*NOTE1: IF \$ITEMB IS 0 THE ONLY PERTINENT DATA IS (\$ERRPC).
;*NOTE2: EACH ITEM IN THE TABLE CONTAINS 4 POINTERS EXPLAINED AS FOLLOWS:

;* EM ::POINTS TO THE ERROR MESSAGE
;* DH ::POINTS TO THE DATA HEADER
;* DT ::POINTS TO THE DATA
;* DF ::POINTS TO THE DATA FORMAT

;*NOTE: ERROR VECTOR TABLE (\$ERRTB) HAS BEEN MODIFIED,
ELIMINATING UNUSED VALUE FOR DATA FORMAT POINTER.
ERROR TYPING ROUTINE HAS ALSO BEEN MODIFIED
ACCORDINGLY.

***** VECTORS FOR FPS ERRORS *****

1113	001354	043715	045134	045410	EMV001:	.WORD	EMA,DHA,DTA	LDCIF,STCFI,STEXP/F,TRAP-TSTR
1114	001362	043715	045134	045416	EMV002:	.WORD	EMA,DHA,DTB	CMPF,LDCLF,STCFL
1115	001370	043715	045134	045424	EMV003:	.WORD	EMA,DHA,DTC	LDCID,STCDI,LDEXP/F,STEXP/D
1116	001376	043715	045134	045432	EMV004:	.WORD	EMA,DHA,DTD	ADDF,SUBF,MULF,DIVF,LDCDF,LDCFD
1117								STCDF,LDCLD,STCDL
1118	001404	043715	045134	045440	EMV005:	.WORD	EMA,DHA,DTE	CMPD,MOOF,STCFD
1119	001412	043715	045134	045446	EMV006:	.WORD	EMA,DHA,DTF	LDEXP/D
1120	001420	043715	045134	045454	EMV007:	.WORD	EMA,DHA,DTG	ADD,SUBD,MULD,DIVD
1121	001426	043715	045134	045462	EMV010:	.WORD	EMA,DHA,DTH	MODD

***** VECTORS FOR FEC/FEA ERRORS *****

1123	001434	043741	045150	045470	EMV011:	.WORD	EMB,DHB,DTI	STCFI,TRAP-TSTR
1124	001442	043741	045150	045502	EMV012:	.WORD	EMB,DHB,DTJ	CMPF,STCFL
1125	001450	043741	045150	045514	EMV013:	.WORD	EMB,DHB,DTK	STCDI,LDEXP/F
1126	001456	043741	045150	045526	EMV014:	.WORD	EMB,DHB,DTL	ADDF,SUBF,MULF,DIVF,LDCDF
1127								LDCFD,STCDF,STCDL
1128	001464	043741	045150	045540	EMV015:	.WORD	EMB,DHB,DTM	CMPD,MOOF
1129	001472	043741	045150	045552	EMV016:	.WORD	EMB,DHB,DTN	LDEXP/D
1130	001500	043741	045150	045564	EMV017:	.WORD	EMB,DHB,DTO	ADD,SUBD,MULD,DIVD
1131	001506	043741	045150	045576	EMV020:	.WORD	EMB,DHB,DTP	MODD

***** VECTORS FOR RESULT ERRORS *****

1133	001514	044125	045210	045624	EMV021:	.WORD	EME,DHC,DTS	CMPF
1134	001522	044125	045246	045706	EMV022:	.WORD	EME,DHD,DTX	CMPD
1135	001530	044214	045210	045662	EMV023:	.WORD	EMF,DHC,DTV	ADDF,SUBF
1136	001536	044214	045246	046016	EMV024:	.WORD	EMF,DHD,DTAB	ADD,SUBD
1137	001544	044266	045210	045662	EMV025:	.WORD	EMG,DHC,DTV	MULF,DIVF
1138	001552	044266	045246	046016	EMV026:	.WORD	EMG,DHD,DTAB	MULD,DIVD
1139	001560	044340	045210	045662	EMV027:	.WORD	EMH,DHC,DTV	MODF-FRAC
1140	001566	044421	045210	045674	EMV030:	.WORD	EMI,DHC,DTW	MOOF-INT
1141	001574	044340	045246	046016	EMV031:	.WORD	EMH,DHD,DTAB	MODD-FRAC
1142	001602	044421	045246	046040	EMV032:	.WORD	EMI,DHD,DTAC	MODD-INT
1143	001610	044555	045210	045662	EMV033:	.WORD	EMK,DHC,DTV	LDCDF,STCDF
1144	001616	044477	045246	045752	EMV034:	.WORD	EMJ,DHD,DTZ	LDCFD
1145	001624	044477	045246	045774	EMV035:	.WORD	EMJ,DHD,DTAA	STCFD
1146	001632	044633	045210	045636	EMV036:	.WORD	EML,DHC,DTT	LDCIF
1147	001640	044633	045246	045730	EMV037:	.WORD	EML,DHD,DTY	LDCID
1148	001646	044633	045210	045650	EMV040:	.WORD	EML,DHC,DTU	LDCLF
1149	001654	044633	045246	045752	EMV041:	.WORD	EML,DHD,DTZ	LDCLD

1150	001662	044713	045134	045610	EMV042:	.WORD	EMM,DHA,DTQ	:	STCFI
1151	001670	044713	045134	045616	EMV043:	.WORD	EMM,DHA,DTR	:	STCDI
1152	001676	044713	045210	045650	EMV044:	.WORD	EMM,DHC,DTU	:	STCFL
1153	001704	044713	045210	045662	EMV045:	.WORD	EMM,DHC,DTV	:	STCDL
1154	001712	044773	045210	045650	EMV046:	.WORD	EMM,DHC,DTU	:	LDEXP/F
1155	001720	044773	045246	045774	EMV047:	.WORD	EMM,DHD,DTAA	:	LDEXP/D
1156	001726	045053	045134	045610	EMV050:	.WORD	EMO,DHA,DTQ	:	STEXP/F
1157	001734	045053	045134	045616	EMV051:	.WORD	EMO,DHA,DTR	:	STEXP/D
1158	001742	000000	000000	000000	EMV052:	.WORD	0,0,0	:	(UNUSED)
1159	001750	000000	000000	000000	EMV053:	.WORD	0,0,0	:	(UNUSED)
1160							;*****		VECTORS FOR CC COPY ERRORS *****
1161	001756	044056	045134	046062	EMV054:	.WORD	EMD,DHA,DTAD	:	STCFI,STCDI,STEXP/F,STEXP/D
1162	001764	044056	045134	046070	EMV055:	.WORD	EMD,DHA,DTAE	:	STCFL,STCDL
1163							;*****		VECTOR FOR ILLEGAL TRAP CATCHER ROUTINE *****
1164	001772	043771	045344	046076	EMV056:	.WORD	EMC,DHF,DTAK	:	UNEXPECTED TRAP

```

1165 .SBTTL PROGRAM DEFINED COMMON TAGS
1166 :#VARIABLES
1167 002000 000000 FPS: .WORD 0 ; FPS STORED HERE AFTER STFPS
1168 002002 000000 FEC: .WORD 0 ; FEC STORED HERE AFTER STST
1169 002004 000000 FEA: .WORD 0 ; FEA STORED HERE AFTER STST
1170 002006 000000 FPPOPC: .WORD 0 ; OLD PC SAVED HERE AFTER TRAP AFTER TRAP
1171 002010 000000 FPPOPS: .WORD 0 ; OLD PS SAVED HERE
1172 002012 000000 FPPOSP: .WORD 0 ; SP AFTER TRAP
1173 002014 000000 EXPFEA: .WORD 0 ; EXPECTED FEA
1174
1175 :#REGISTER CONTENTS, AT ERROR, FOR DISPLAY
1176 002016 000000 EREG0: .WORD 0
1177 002020 000007 EREG1: .WORD 0
1178 002022 00000W EREG2: .WORD 0
1179 002024 000000 EREG3: .WORD 0
1180 002026 000000 EREG4: .WORD 0
1181 002030 000000 EREG5: .WORD 0
1182 002032 000000 EREG6: .WORD 0
1183 002034 000000 EREG7: .WORD 0
1184
1185 :#CONSTANTS
1186 002036 052525 177777 125252 PREVAC: .WORD ALTP,M1,ALTN,0 ; PREV CONTENTS OF FLOAT AC
1187 002044 000000
1188
1189 :#MESSAGES FOR BEGIN PROGRAM/START OF PASS/ETC
1190
1191 002046 005015 005012 042115 BGNMES: .ASCII <15><12><12><12>"MD-11-DQFPB-A)...".
1192 002054 030455 026461 050504
1193 002062 050106 026502 037101
1194 002070 027056 056
1195 002073 120 050104 030455 .ASCIZ "PDP-11/6X F.P.U. ADVANCED INSTRUCTION TESTS"<15><12>
1196 002100 027461 054066 043040
1197 002106 050056 052456 020056
1198 002114 042101 040526 041516
1199 002122 042105 044440 051516
1200 002130 051124 041525 044524
1201 002136 047117 052040 051505
1202 002144 051524 05015 000
1203 002151 015 050012 051501 NWPAS1: .ASCIZ <15><12>"PASS #".
1204 002156 020123 000043

```

```

1205
1206
1207
1208
1209
1210
1211
1212
1213 002162
1214
1215
1216 002162 012706 001100
1217 002166 005026
1218 002170 022706 001146
1219 002174 001374
1220 002176 012706 001100
1221
1222 002202 012737 041530 000020
1223 002210 012737 000340 000022
1224 002216 012737 042006 000030
1225 002224 012737 000340 000032
1226 002232 012737 043442 000034
1227 002240 012737 000340 000036
1228 002246 012737 043510 000024
1229 002254 012737 000340 000026
1230 002262 013737 033032 033024
1231 002270 005037 001310
1232 002274 005037 001312
1233 002300 012737 000001 001122
1234 002306 012737 002306 001110
1235 002314 012737 002314 001114
1236
1237
1238 002322 013746 000004
1239 002326 012737 002362 000004
1240 002334 012737 177570 001146
1241 002342 012737 177570 001150
1242 002350 022777 177777 176570
1243 002356 001012
1244
1245 002360 000403
1246 002362 012716 002370 64$:
1247 002366 000002
1248 002370 012737 000176 001146 65$:
1249 002376 012737 000174 001150
1250 002404 012637 000004 66$:
1251
1252 002410 005037 001332
1253 002414 132737 000200 001345
1254 002422 001403
1255 002424 012737 001346 001146
1256 002432
1257
1258
1259 002432 012737 041470 000244
1260 002440 005037 000246

```

```

.SBTTL START OF PASS ROUTINE
.EVEN ; START ON AN EVEN BOUNDARY
;*****
;.ENABL AMA ; ASSEMBLE ALL RELATIVE REFERENCES AS ABSOLUTE
;*****
START:
.SBTTL INITIALIZE THE COMMON TAGS
;;CLEAR THE COMMON TAGS ($CMTAG) AREA
MOV #SCMTAG,R6 ;;FIRST LOCATION TO BE CLEARED
CLR (R6)+ ;;CLEAR MEMORY LOCATION
CMP #SWR,R6 ;;DONE?
BNE -6 ;;LOOP BACK IF NO
MOV #STACK,SP ;;SETUP THE STACK POINTER
;;INITIALIZE A FEW VECTORS
MOV #SCOPE,@IOTVEC ;;IOT VECTOR FOR SCOPE ROUTINE
MOV #340,@IOTVEC+2 ;;LEVEL 7
MOV #ERROR,@EMTVEC ;;EMT VECTOR FOR ERROR ROUTINE
MOV #340,@EMTVEC+2 ;;LEVEL 7
MOV #TRAP,@TRAPVEC ;;TRAP VECTOR FOR TRAP CALLS
MOV #340,@TRAPVEC+2 ;;LEVEL 7
MOV #SPWRON,@PWAVEC ;;POWER FAILURE VECTOR
MOV #340,@PWAVEC+2 ;;LEVEL 7
MOV #ENDCT,@SEOPCT ;;SETUP END-OF-PROGRAM COUNTER
CLR $TIMES ;;INITIALIZE NUMBER OF ITERATIONS
CLR $ESCAPE ;;CLEAR THE ESCAPE ON ERROR ADDRESS
MOV #1,$ERRMAX ;;ALLOW ONE ERROR PER TEST
MOV #,$SLPADR ;;INITIALIZE THE LOOP ADDRESS FOR SCOPE
MOV #,$SLPERR ;;SETUP THE ERROR LOOP ADDRESS
;;SIZE FOR A HARDWARE SWITCH REGISTER. IF NOT FOUND OR IT IS
;;EQUAL TO A "-1" SETUP FOR A SOFTWARE SWITCH REGISTER.
MOV @ERRVEC-(SP) ;;SAVE ERROR VECTOR
MOV #64$,@ERRVEC ;;SET UP ERROR VECTOR
MOV #DSWR,SWR ;;SETUP FOR A HARDWARE SWICH REGISTER
MOV #DDISP,DISPLAY ;;AND A HARDWARE DISPLAY REGISTER
CMP #-1,$SWR ;;TRY TO REFERENCE HARDWARE SWR
BNE 66$ ;;BRANCH IF NO TIMEOUT TRAP OCCURRED
; AND THE HARDWARE SWR IS NOT = -1
BR 65$ ;;BRANCH IF NO TIMEOUT
MOV #65$,(SP) ;;SET UP FOR TRAP RETURN
RTI
MOV #SWREG,SWR ;;POINT TO SOFTWARE SWR
MOV #DISPREG,DISPLAY
MOV (SP)+,@ERRVEC ;;RESTORE ERROR VECTOR
CLR $PASS ;;CLEAR PASS COUNT
BITB #APTSIZE,$ENVM ;;TEST USER SIZE UNDER APT
BEQ 67$ ;;YES,USE NON-APT SWITCH
MOV #SSWREG,SWR ;;NO,USE APT SWITCH REGISTER
67$:
; SET UP FPP UNEXPECTED TRAP CATCHER - - - - -
MOV #FPPILT,@FPPVEC ;;NEW PC AT FPP TRAP
CLR @FPPVEC+2 ;;NEW PS AT FPP TRAP

```

```

1261
1262 002444 104401 002046          TYPE      ,BGNMES          ; ID MESSAGE AT START
1263
1264 ;////////////////////////////////////
1265 ; MESSAGE ON WHETHER OR NOT HFP UNIT IS PRESENT
1266 ;
1267 002450 076600 000022          MED          RWHAMI          ;WHAMI INTO RO
1268 002454 032700 000020          BIT          #BIT04,RO      ;IS THERE A HFP UNIT ?
1269 002460 001403          BEQ          70$           ;NO, BR
1270 002462 104401 002476          TYPE          68$           ;INDICATE FP11-E PRESENT
1271 002466 000453          BR          NEWPAS         ;GO FOR SUBPASS INIT
1272 002470 104401 002536          70$:         TYPE          69$           ;INDICATE NO FP11-E
1273 002474 000450          BR          NEWPAS         ;GO FOR SUBPASS INIT
1274
1275 002476 005015 020052 050106          68$:         .ASCIZ <15><12>*" FP11-E HFP UNIT PRESENT *"<15><12>
1276 002504 030461 042455 044040
1277 002512 050106 052440 044516
1278 002520 020124 051120 051505
1279 002526 047105 020124 006452
1280 002534 000012
1281 002536 005015 020052 047516          69$:         .ASCIZ <15><12>*" NO FP11-E HFP UNIT - ALL TESTS HFP ONLY *"<15><12>
1282 002544 043040 030520 026461
1283 002552 020105 043110 020120
1284 002560 047125 052111 026440
1285 002566 040440 046114 052040
1286 002574 051505 051524 053440
1287 002602 050106 047440 046116
1288 002610 020131 006452 000012
1289
1290 .EVEN
1291 ;////////////////////////////////////
1292
1293 ;*****
1294 ;NEW PASS ENTERS HERE
1295 ;*****
1296
1297
1298 002616 012706 001100          NEWPAS:      MOV          #STACK,SP      ;RESET STACK PTR
1299
1300 002622 032777 010000 176316          BIT          #BIT12,SWR      ;INHIBIT STATUS TYPEOUTS ?
1301 002630 001011          BNE          SUBPAS         ;BR IF YES
1302
1303 002632 104401 002151          TYPE          NEWPAS1        ;"PASS #"
1304 002636 013746 001332          MOV          $PASS,-(SP)      ;PASS COUNT INTO ...
1305 002642 005216          INC          (SP)            ;1-N RANGE
1306 002644 104403          TYPOS        ;TYPE OCTAL
1307 002646 006 000          .BYTE        6,0            ;6 DIGITS, NO LEADING ZEROS
1308 002650 104401 001321          TYPE          , $CRLF        ;END THE LINE
1309
1310
1311 ;*****
1312 ;NEW SUBPASS ENTERS HERE
1313 ;*****
1314
1315 002654 076600 000022          SUBPAS:      MED          RWHAMI          ;GET WHAMI INTO RO
1316 002660 032700 000020          BIT          #BIT04,RO      ;1=HFP PRESENT, 0=NO

```

```

1317 002664 001430      BEQ      20$      ; IF NO HFP, TEST WARM ONLY
1318
1319 002666 076600 000144  MED      ,RFLAG  ; GET FLAGS INTO R0
1320
1321 002672 032777 000002 176246  BIT      #SW01,2SWR ; SW01: 1=HFP OR WFP TEST ONLY
1322 002700 001413      BEQ      1$      ; 0=ALTERNATE HFP/WFP PER PASS
1323
1324 002702 032777 000001 176236  BIT      #SW00,2SWR ; SW00: 1=HFP ONLY
1325 002710 001403      BEQ      2$      ; 0=HFP ONLY
1326 002712 042700 010000      BIC      #BIT12,R0 ; CLEAR HFP ENABLE FLAG<5> FOR WFP
1327 002716 000402      BR       3$      ;
1328 002720 052700 010000      BIS      #BIT12,R0 ; SET HFP ENABLE FLAG<5> FOR HFP
1329 002724 076600 000344      2$: MED      ,WFLAG  ; REWRITE FLAGS
1330
1331 002730 032700 010000      3$: BIT      #BIT12,R0 ; TEST WHO'S ENABLED: HOT, WARM
1332 002734 001404      BEQ      20$     ; SET APPROPRIATE HEADER:
1333
1334 002736 012737 043700 042252 19$: MOV      #ASCHOT,HOTWRM ; "HOT: "
1335 002744 000403      BR       21$     ;
1336 002746 012737 043706 042252 20$: MOV      #ASCWRM,HOTWRM ; "WARM: "
1337 002754 005037 001102      21$: CLR      $STNM  ; ALL DONE, RESET TEST NUMBER COUNTER

```

```

1338
1339
1340
1341
1342
1343 002760 000004
1344 002762 012705 002774
1345 002766 004737 033076
1346
1347 002772 000407
1348
1349 002774
1350 002774 000000 000000
1351 003000 000000 000000
1352 003004 047453 047444
1353 003010 000000
1354
1355
1356
1357
1358
1359
1360
1361 003012 000004
1362 003014 012705 003026
1363 003020 004737 033076
1364
1365 003024 000407
1366
1367 003026
1368 003026 052525 052525
1369 003032 052525 052525
1370 003036 047513 047504
1371 003042 000000
1372
1373
1374
1375
1376
1377
1378
1379 003044 000004
1380 003046 012705 003060
1381 003052 004737 033076
1382
1383 003056 000407
1384
1385 003060
1386 003060 077777 177777
1387 003064 177777 177777
1388 003070 047507 047510
1389 003074 000000
1390
1391
1392
1393

```

```

*****
: *TEST 1 TEST OF CMPF INSTR, DATA SET CMPF-1
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
↑ST1: SCOPE
MOV #CMPF1,R5 ; PTR TO TEST DATA SET
JSR PC,@#CMPFT ; GO TEST

BR TST2 ;;

CMPF1: ; TEST DATA SET CMPF-1:
.WORD 0,0 ; INITIAL AC FLOAT NUMBER
.WORD 0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 047453,047444 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
: *TEST 2 TEST OF CMPF INSTR, DATA SET CMPF-2
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, LONG INTEGER, ROUND MODES
*****
↑ST2: SCOPE
MOV #CMPF2,R5 ; PTR TO TEST DATA SET
JSR PC,@#CMPFT ; GO TEST

BR TST3 ;;

CMPF2: ; TEST DATA SET CMPF-2:
.WORD ALTP,ALTP ; INITIAL AC FLOAT NUMBER
.WORD ALTP,ALTP ; INITIAL MEM FLOAT NUMBER
.WORD 047513,047504 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
: *TEST 3 TEST OF CMPF INSTR, DATA SET CMPF-3
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, LONG INTEGER, ROUND MODES
*****
↑ST3: SCOPE
MOV #CMPF3,R5 ; PTR TO TEST DATA SET
JSR PC,@#CMPFT ; GO TEST

BR TST4 ;;

CMPF3: ; TEST DATA SET CMPF-3:
.WORD LGP,M1 ; INITIAL AC FLOAT NUMBER
.WORD LGN,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 047507,047510 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
: *TEST 4 TEST OF CMPF INSTR, DATA SET CMPF-4

```

E03

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 32
T4 TEST OF CMPF INSTR, DATA SET CMPF-4

1394
1395
1396
1397 003076 000004
1398 003100 012705 003112
1399 003104 004737 033076
1400
1401 003110 000407
1402
1403 003112
1404 003112 125252 125252
1405 003116 125252 125252
1406 003122 047453 047444
1407 003126 000000
1408
1409
1410
1411
1412
1413
1414
1415 003130 000004
1416 003132 012705 003144
1417 003136 004737 033076
1418
1419 003142 000407
1420
1421 003144
1422 003144 177777 177777
1423 003150 077777 177777
1424 003154 047457 047440
1425 003160 000000
1426
1427
1428
1429
1430
1431
1432
1433 003162 000004
1434 003164 012705 003176
1435 003170 004737 033076
1436
1437 003174 000407
1438
1439 003176
1440 003176 037777 177777
1441 003202 040000 000000
1442 003206 047517 047500
1443 003212 000000
1444
1445
1446
1447
1448
1449

```

: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
: *****
↑ST4: SCOPE
      MOV #CMPF4,R5 ; PTR TO TEST DATA SET
      JSR PC,@#CMPFT ; GO TEST

      BR TST5 ;;

CMPF4: ; TEST DATA SET CMPF-4:
      .WORD ALTN,ALTN ; INITIAL AC FLOAT NUMBER
      .WORD ALTN,ALTN ; INITIAL MEM FLOAT NUMBER
      .WORD 047453,047444 ; FPS: BEFORE, AFTER
      .WORD NA ; FEC AFTER ( 0 = N/A )

```

```

: *****
: *TEST 5 TEST OF CMPF INSTR, DATA SET CMPF-5
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
: *****
↑ST5: SCOPE
      MOV #CMPF5,R5 ; PTR TO TEST DATA SET
      JSR PC,@#CMPFT ; GO TEST

      BR TST6 ;;

CMPF5: ; TEST DATA SET CMPF-5:
      .WORD LGN,M1 ; INITIAL AC FLOAT NUMBER
      .WORD LGP,M1 ; INITIAL MEM FLOAT NUMBER
      .WORD 047457,047440 ; FPS: BEFORE, AFTER
      .WORD NA ; FEC AFTER ( 0 = N/A )

```

```

: *****
: *TEST 6 TEST OF CMPF INSTR, DATA SET CMPF-6
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, LONG INTEGER, ROUND MODES
: *****
↑ST6: SCOPE
      MOV #CMPF6,R5 ; PTR TO TEST DATA SET
      JSR PC,@#CMPFT ; GO TEST

      BR TST7 ;;

CMPF6: ; TEST DATA SET CMPF-6:
      .WORD 037777,M1 ; INITIAL AC FLOAT NUMBER
      .WORD 040000,000000 ; INITIAL MEM FLOAT NUMBER
      .WORD 047517,047500 ; FPS: BEFORE, AFTER
      .WORD NA ; FEC AFTER ( 0 = N/A )

```

```

: *****
: *TEST 7 TEST OF CMPF INSTR, DATA SET CMPF-7
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, LONG INTEGER, TRUNCATE MODES

```


F03

FPU ADVANCED INSTR TESTS
DQFP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 33
T7 TEST OF CMPF INSTR, DATA SET CMPF-7

```

1450
1451 003214 000004
1452 003216 012705 003230
1453 003222 004737 033076
1454
1455 003226 000407
1456
1457 003230
1458 003230 050000 000001
1459 003234 050000 000000
1460 003240 047547 047550
1461 003244 000000
1462
1463
1464
1465
1466
1467
1468
1469 003246 000004
1470 003250 012705 003262
1471 003254 004737 033076
1472
1473 003260 000407
1474
1475 003262
1476 003262 126000 000000
1477 003266 124000 000000
1478 003272 047417 047400
1479 003276 000000
1480
1481
1482
1483
1484
1485
1486
1487 003300 000004
1488 003302 012705 003314
1489 003306 004737 033076
1490
1491 003312 000407
1492
1493 003314
1494 003314 007417 007417
1495 003320 100000 000000
1496 003324 047443 147443
1497 003330 100014
1498
1499
1500
1501
1502
1503
1504
1505 003332 000004

```

```

*****
TST7: SCOPE
      MOV      #CMPF7,R5      ; PTR TO TEST DATA SET
      JSR      PC,@#CMPFT    ; GO TEST
      BR       TST10        ;;

CMPF7: ; TEST DATA SET CMPF-7:
      .WORD   050000,000001  ; INITIAL AC FLOAT NUMBER
      .WORD   050000,000000  ; INITIAL MEM FLOAT NUMBER
      .WORD   047547,047550  ; FPS: BEFORE, AFTER
      .WORD   NA              ; FEC AFTER ( 0 = N/A )

*****
*TEST 10      TEST OF CMPF INSTR, DATA SET CMPF-10
*             ALL INTERRUPT ENABLES ON
*             SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****
TST10: SCOPE
      MOV      #CMPF10,R5    ; PTR TO TEST DATA SET
      JSR      PC,@#CMPFT   ; GO TEST
      BR       TST11        ;;

CMPF10: ; TEST DATA SET CMPF-10:
      .WORD   126000,000000  ; INITIAL AC FLOAT NUMBER
      .WORD   124000,000000  ; INITIAL MEM FLOAT NUMBER
      .WORD   047417,047400  ; FPS: BEFORE, AFTER
      .WORD   NA              ; FEC AFTER ( 0 = N/A )

*****
*TEST 11      TEST OF CMPF INSTR, DATA SET CMPF-11
*             ALL INTERRUPT ENABLES ON
*             SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST11: SCOPE
      MOV      #CMPF11,R5    ; PTR TO TEST DATA SET
      JSR      PC,@#CMPFT   ; GO TEST
      BR       TST12        ;;

CMPF11: ; TEST DATA SET CMPF-11:
      .WORD   ALT4P,ALT4P    ; INITIAL AC FLOAT NUMBER
      .WORD   MO,0           ; INITIAL MEM FLOAT NUMBER
      .WORD   047443,147443  ; FPS: BEFORE, AFTER
      .WORD   100014        ; FEC AFTER ( 0 = N/A )

*****
*TEST 12      TEST OF CMPF INSTR, DATA SET CMPF-12
*             ALL INTERRUPT ENABLES ON
*             SHORT FLOAT, LONG INTEGER, ROUND MODES
*****
TST12: SCOPE

```

G03

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 34
T12 TEST OF CMPF INSTR, DATA SET CMPF-12

1506 003334 012705 003346
1507 003340 004737 033076
1508
1509 003344 000407
1510
1511 003346
1512 003346 006177 177777
1513 003352 004177 177777
1514 003356 047507 047510
1515 003362 000000
1516
1517
1518
1519
1520
1521
1522
1523 003364 000004
1524 003366 012705 003400
1525 003372 004737 033076
1526
1527 003376 000407
1528
1529 003400
1530 003400 125252 125252
1531 003404 100177 177777
1532 003410 043557 043540
1533 003414 000000
1534
1535
1536
1537
1538
1539
1540
1541 003416 000004
1542 003420 012705 003432
1543 003424 004737 033076
1544
1545 003430 000407
1546
1547 003432
1548 003432 000377 177777
1549 003436 000377 177776
1550 003442 047407 047410
1551 003446 000000
1552
1553
1554

```

MOV      #CMPF12,R5      ; PTR TO TEST DATA SET
JSR      PC,@#CMPFT      ; GO TEST

BR       TST13           ;;

CMPF12:  ; TEST DATA SET CMPF-12:
        .WORD 006177,M1   ; INITIAL AC FLOAT NUMBER
        .WORD 004177,M1   ; INITIAL MEM FLOAT NUMBER
        .WORD 047507,047510 ; FPS: BEFORE, AFTER
        .WORD NA          ; FEC AFTER ( 0 = N/A )

:*****
: *TEST 13      TEST OF CMPF INSTR, DATA SET CMPF-13
: *            -0 INTERRUPT ENABLE OFF, ALL OTHERS ON
: *            SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
:*****
↑TST13:  SCOPE
        MOV      #CMPF13,R5      ; PTR TO TEST DATA SET
        JSR      PC,@#CMPFT      ; GO TEST

BR       TST14           ;;

CMPF13:  ; TEST DATA SET CMPF-13:
        .WORD ALTN,ALTN   ; INITIAL AC FLOAT NUMBER
        .WORD ZX1MN,M1    ; INITIAL MEM FLOAT NUMBER
        .WORD 043557,043540 ; FPS: BEFORE, AFTER
        .WORD NA          ; FEC AFTER ( 0 = N/A )

:*****
: *TEST 14      TEST OF CMPF INSTR, DATA SET CMPF-14
: *            ALL INTERRUPT ENABLES ON
: *            SHORT FLOAT, SHORT INTEGER, ROUND MODES
:*****
↑TST14:  SCOPE
        MOV      #CMPF14,R5      ; PTR TO TEST DATA SET
        JSR      PC,@#CMPFT      ; GO TEST

BR       TST15           ;;

CMPF14:  ; TEST DATA SET CMPF-14:
        .WORD 000377,M1   ; INITIAL AC FLOAT NUMBER
        .WORD 000377,M2   ; INITIAL MEM FLOAT NUMBER
        .WORD 047407,047410 ; FPS: BEFORE, AFTER
        .WORD NA          ; FEC AFTER ( 0 = N/A )
    
```

H03

FPU ADVANCED INSTR TESTS
DQFP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 35
T15 TEST OF CMPD INSTR, DATA SET CMPD-1

```

1555
1556
1557
1558
1559
1560 003450 000004
1561 003452 012705 003464
1562 003456 004737 033244
1563
1564 003462 000413
1565
1566 003464
1567 003464 000000 000000 000000
1568 003472 000000
1569 003474 000000 000000 000000
1570 003502 000000
1571 003504 047713 047704
1572 003510 000000
1573
1574
1575
1576
1577
1578
1579
1580 003512 000004
1581 003514 012705 003526
1582 003520 004737 033244
1583
1584 003524 000413
1585
1586 003526
1587 003526 177777 177777 177777
1588 003534 177777
1589 003536 077777 177777 177777
1590 003544 177777
1591 003546 047717 047700
1592 003552 000000
1593
1594
1595
1596
1597
1598
1599
1600 003554 000004
1601 003556 012705 003570
1602 003562 004737 033244
1603
1604 003566 000413
1605
1606 003570
1607 003570 170360 170360 170360
1608 003576 170360
1609 003600 170360 170360 170360
1610 003606 170360

```

```

*****
*TEST 15 TEST OF CMPD INSTR, DATA SET CMPD-1
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
TST15: SCOPE
MOV #CMPD1,R5 ; PTR TO TEST DATA SET
JSR PC,#CMPD1 ; GO TEST
BR TST16 ;;

CMPD1: ; TEST DATA SET CMPD-1:
.WORD 0,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD 0,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 047713,047704 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
*TEST 16 TEST OF CMPD INSTR, DATA SET CMPD-2
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
TST16: SCOPE
MOV #CMPD2,R5 ; PTR TO TEST DATA SET
JSR PC,#CMPD2 ; GO TEST
BR TST17 ;;

CMPD2: ; TEST DATA SET CMPD-2:
.WORD LGN,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD LGP,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 047717,047700 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
*TEST 17 TEST OF CMPD INSTR, DATA SET CMPD-3
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
TST17: SCOPE
MOV #CMPD3,R5 ; PTR TO TEST DATA SET
JSR PC,#CMPD3 ; GO TEST
BR TST20 ;;

CMPD3: ; TEST DATA SET CMPD-3:
.WORD ALT4N,ALT4N,ALT4N,ALT4N ; INITIAL AC FLOAT NUMBER
.WORD ALT4N,ALT4N,ALT4N,ALT4N ; INITIAL MEM FLOAT NUMBER

```

1611 003610 047713 047704
1612 003614 000000

.WORD 047713,047704 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER (0 = N/A)

1613
1614
1615
1616
1617
1618
1619

*TEST 20 TEST OF CMPD INSTR, DATA SET CMPD-4
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES

1620 003616 000004
1621 003620 012705 003632
1622 003624 004737 033244

TST20: SCOPE
MOV #CMPD4,R5 ; PTR TO TEST DATA SET
JSR PC,#CMPDT ; GO TEST

1623
1624 003630 000413
1625

BR TST21 ; ;

1626 003632
1627 003632 077777 177777 177777

CMPD4: ; TEST DATA SET CMPD-4:
.WORD LGP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER

1628 003640 177777
1629 003642 177777 177777 177777

.WORD LGN,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER

1630 003650 177777
1631 003652 047647 047650
1632 003656 000000

.WORD 047647,047650 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER (0 = N/A)

1633
1634
1635

*TEST 21 TEST OF CMPD INSTR, DATA SET CMPD-5
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES

1636
1637
1638
1639

1640 003660 000004
1641 003662 012705 003674
1642 003666 004737 033244

TST21: SCOPE
MOV #CMPD5,R5 ; PTR TO TEST DATA SET
JSR PC,#CMPDT ; GO TEST

1643
1644 003672 000413
1645

BR TST22 ; ;

1646 003674
1647 003674 007417 007417 007417

CMPD5: ; TEST DATA SET CMPD-5:
.WORD ALT4P,ALT4P,ALT4P,ALT4P ; INITIAL AC FLOAT NUMBER

1648 003702 007417
1649 003704 007417 007417 007417

.WORD ALT4P,ALT4P,ALT4P,ALT4P ; INITIAL MEM FLOAT NUMBER

1650 003712 007417
1651 003714 047653 047644
1652 003720 000000

.WORD 047653,047644 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER (0 = N/A)

1653
1654
1655

*TEST 22 TEST OF CMPD INSTR, DATA SET CMPD-6
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, ROUND MODES

1656
1657
1658
1659

1660 003722 000004
1661 003724 012705 003736
1662 003730 004737 033244

TST22: SCOPE
MOV #CMPD6,R5 ; PTR TO TEST DATA SET
JSR PC,#CMPDT ; GO TEST

1663
1664 003734 000413
1665

BR TST23 ; ;

1666 003736

CMPD6: ; TEST DATA SET CMPD-6:

J03

FPU ADVANCED INSTR TESTS
DQFP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 37
T22 TEST OF CMPD INSTR, DATA SET CMPD-6

1667	003736	125252	125252	125252	.WORD	ALTN,ALTN,ALTN,ALTN	; INITIAL AC FLOAT NUMBER
1668	003744	125252					
1669	003746	100177	177777	177777	.WORD	ZX1MN,M1,M1,M1	; INITIAL MEM FLOAT NUMBER
1670	003754	177777					
1671	003756	047703	147703		.WORD	047703,147703	; FPS: BEFORE, AFTER
1672	003762	100014			.WORD	100014	; FEC AFTER (0 = N/A)
1673							
1674							

```

*****
:TEST 23      TEST OF CMPD INSTR, DATA SET CMPD-7
:*           ALL INTERRUPT ENABLES ON
:*           LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

1680	003764	000004			TST23: SCOPE		
1681	003766	012705	004000		MOV	#CMPD7,R5	; PTR TO TEST DATA SET
1682	003772	004737	033244		JSR	PC,#CMPD7	; GO TEST
1683							
1684	003776	000413			BR	TST24	::
1685							

1686	004000				CMPD7: ; TEST DATA SET CMPD-7:		
1687	004000	002177	177777	177777	.WORD	002177,M1,M1,M1	; INITIAL AC FLOAT NUMBER
1688	004006	177777					
1689	004010	005177	177777	177777	.WORD	005177,M1,M1,M1	; INITIAL MEM FLOAT NUMBER
1690	004016	177777					
1691	004020	047657	047640		.WORD	047657,047640	; FPS: BEFORE, AFTER
1692	004024	000000			.WORD	NA	; FEC AFTER (0 = N/A)
1693							
1694							

```

*****
:TEST 24      TEST OF CMPD INSTR, DATA SET CMPD-10
:*           ALL INTERRUPT ENABLES ON
:*           LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

1695					TST24: SCOPE		
1700	004026	000004			MOV	#CMPD10,R5	; PTR TO TEST DATA SET
1701	004030	012705	004042		JSR	PC,#CMPD10	; GO TEST
1702	004034	004737	033244				
1703							
1704	004040	000413			BR	TST25	::
1705							

1706	004042				CMPD10: ; TEST DATA SET CMPD-10:		
1707	004042	030000	000000	000000	.WORD	030000,000000,000000,000000	; INITIAL AC FLOAT NUMBER
1708	004050	000000					
1709	004052	027777	177777	177777	.WORD	027777,M1,M1,M1	; INITIAL MEM FLOAT NUMBER
1710	004060	177777					
1711	004062	047647	047650		.WORD	047647,047650	; FPS: BEFORE, AFTER
1712	004066	000000			.WORD	NA	; FEC AFTER (0 = N/A)
1713							
1714							

```

*****
:TEST 25      TEST OF CMPD INSTR, DATA SET CMPD-11
:*           ALL INTERRUPT ENABLES ON
:*           LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

1715					TST25: SCOPE		
1720	004070	000004			MOV	#CMPD11,R5	; PTR TO TEST DATA SET
1721	004072	012705	004104		JSR	PC,#CMPD11	; GO TEST
1722	004076	004737	033244				

K03

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 38
T25 TEST OF CMPD INSTR, DATA SET CMPD-11

```

1723
1724 004102 000413 BR TST26 ;;
1725
1726 004104 CMPD11: ; TEST DATA SET CMPD-11:
1727 004104 102000 000000 000000 .WORD 102000,000000,000000,000000 ; INITIAL AC FLOAT NUMBER
1728 004112 000000
1729 004114 102000 000000 000000 .WORD 102000,000000,000000,000001 ; INITIAL MEM FLOAT NUMBER
1730 004122 000001
1731 004124 047607 047610 .WORD 047607,047610 ; FPS: BEFORE, AFTER
1732 004130 000000 .WORD NA ; FEC AFTER ( 0 = N/A )

```

```

1733
1734
1735
1736 ;*****
1737 ;*TEST 26 TEST OF CMPD INSTR, DATA SET CMPD-12
1738 ;* ALL INTERRUPT ENABLES ON
1739 ;* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
1740 ;*****

```

```

1740 004132 000004 TST26: SCOPE
1741 004134 012705 004146 MOV #CMPD12,R5 ; PTR TO TEST DATA SET
1742 004140 004737 033244 JSR PC,#CMPD1 ; GO TEST
1743
1744 004144 000413 BR TST27 ;;

```

```

1745
1746 004146 CMPD12: ; TEST DATA SET CMPD-12:
1747 004146 002000 000000 000000 .WORD 002000,000000,000000,000000 ; INITIAL AC FLOAT NUMBER
1748 004154 000000
1749 004156 014000 000000 000000 .WORD 014000,000000,000000,000000 ; INITIAL MEM FLOAT NUMBER
1750 004164 000000
1751 004166 047757 047740 .WORD 047757,047740 ; FPS: BEFORE, AFTER
1752 004172 000000 .WORD NA ; FEC AFTER ( 0 = N/A )

```

```

1753
1754
1755 ;*****
1756 ;*TEST 27 TEST OF CMPD INSTR, DATA SET CMPD-13
1757 ;* -0 INTERRUPT ENABLE OFF, ALL OTHERS ON
1758 ;* LONG FLOAT, LONG INTEGER, ROUND MODES
1759 ;*****

```

```

1760 004174 000004 TST27: SCOPE
1761 004176 012705 004210 MOV #CMPD13,R5 ; PTR TO TEST DATA SET
1762 004202 004737 033244 JSR PC,#CMPD1 ; GO TEST
1763
1764 004206 000413 BR TST30 ;;

```

```

1765
1766 004210 CMPD13: ; TEST DATA SET CMPD-13:
1767 004210 000000 000000 000000 .WORD 0,0,0,0 ; INITIAL AC FLOAT NUMBER
1768 004216 000000
1769 004220 100177 177777 000000 .WORD ZX1MN,M1,0,M1 ; INITIAL MEM FLOAT NUMBER
1770 004226 177777
1771 004230 043713 043704 .WORD 043713,043704 ; FPS: BEFORE, AFTER
1772 004234 000000 .WORD NA ; FEC AFTER ( 0 = N/A )

```

```

1773
1774
1775 ;*****
1776 ;*TEST 30 TEST OF CMPD INSTR, DATA SET CMPD-14
1777 ;* ALL INTERRUPT ENABLES ON
1778 ;* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES

```

1779
 1780 004236 000004
 1781 004240 012705 004252
 1782 004244 004737 033244
 1783
 1784 004250 000413
 1785
 1786 004252
 1787 004252 100777 000000 177777
 1788 004260 000001
 1789 004262 100777 000000 177777
 1790 004270 000000
 1791 004272 047657 047640
 1792 004276 000000
 1793
 1794

```

:*****
↑T30: SCOPE
      MOV      #CMPD14,R5      ; PTR TO TEST DATA SET
      JSR      PC,@#CMPDT     ; GO TEST
      BR       TST31          ;;
CMPD14: ; TEST DATA SET CMPD-14:
        .WORD  100777,000000,M1,000001 ; INITIAL AC FLOAT NUMBER
        .WORD  100777,000000,M1,000000 ; INITIAL MEM FLOAT NUMBER
        .WORD  047657,047640          ; FPS: BEFORE, AFTER
        .WORD  NA                      ; FEC AFTER ( 0 = N/A )
  
```

M03

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 40
T31 TEST OF ADDF INSTR, DATA SET ADDF-1

1795			
1796			
1797			
1798			
1799			
1800	004300	000004	
1801	004302	012705	004314
1802	004306	004737	033432
1803			
1804	004312	000411	
1805			
1806	004314		
1807	004314	000177	177777
1808	004320	000177	177777
1809	004324	000000	000000
1810	004330	047453	047444
1811	004334	000000	
1812			
1813			
1814			
1815			
1816			
1817			
1818			
1819	004336	000004	
1820	004340	012705	004352
1821	004344	004737	033432
1822			
1823	004350	000411	
1824			
1825	004352		
1826	004352	000000	000000
1827	004356	125252	125252
1828	004362	125252	125252
1829	004366	047407	047410
1830	004372	000000	
1831			
1832			
1833			
1834			
1835			
1836			
1837			
1838	004374	000004	
1839	004376	012705	004410
1840	004402	004737	033432
1841			
1842	004406	000411	
1843			
1844	004410		
1845	004410	052525	052525
1846	004414	000000	000000
1847	004420	052525	052525
1848	004424	047557	047540
1849	004430	000000	
1850			

```

*****
*TEST 31      TEST OF ADDF INSTR, DATA SET ADDF-1
*
*            ALL INTERRUPT ENABLES ON
*            SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST31:  SCOPE
        MOV      #ADDF1,R5      ; PTR TO TEST DATA SET
        JSR      PC,2#ADDF1     ; GO TEST

        BR      TST32          ;;

ADDF1:  ; TEST DATA SET ADDF-1:
        .WORD   ZXIMP,M1        ; INITIAL AC FLOAT NUMBER
        .WORD   ZXIMP,M1        ; INITIAL MEM FLOAT NUMBER
        .WORD   0,0             ; EXPECTED FLOAT RESULT
        .WORD   047453,047444   ; FPS: BEFORE, AFTER
        .WORD   NA              ; FEC AFTER ( 0 = N/A )

*****
*TEST 32      TEST OF ADDF INSTR, DATA SET ADDF-2
*
*            ALL INTERRUPT ENABLES ON
*            SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****
TST32:  SCOPE
        MOV      #ADDF2,R5      ; PTR TO TEST DATA SET
        JSR      PC,2#ADDF2     ; GO TEST

        BR      TST33          ;;

ADDF2:  ; TEST DATA SET ADDF-2:
        .WORD   0,0             ; INITIAL AC FLOAT NUMBER
        .WORD   ALTN,ALTN       ; INITIAL MEM FLOAT NUMBER
        .WORD   ALTN,ALTN       ; EXPECTED FLOAT RESULT
        .WORD   047407,047410   ; FPS: BEFORE, AFTER
        .WORD   NA              ; FEC AFTER ( 0 = N/A )

*****
*TEST 33      TEST OF ADDF INSTR, DATA SET ADDF-3
*
*            ALL INTERRUPT ENABLES ON
*            SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****
TST33:  SCOPE
        MOV      #ADDF3,R5      ; PTR TO TEST DATA SET
        JSR      PC,2#ADDF3     ; GO TEST

        BR      TST34          ;;

ADDF3:  ; TEST DATA SET ADDF-3:
        .WORD   ALTP,ALTP       ; INITIAL AC FLOAT NUMBER
        .WORD   0,0             ; INITIAL MEM FLOAT NUMBER
        .WORD   ALTP,ALTP       ; EXPECTED FLOAT RESULT
        .WORD   047557,047540   ; FPS: BEFORE, AFTER
        .WORD   NA              ; FEC AFTER ( 0 = N/A )

```



```

1851
1852
1853
1854
1855
1856
1857 004432 000004
1858 004434 012705 004446
1859 004440 004737 033432
1860
1861 004444 000411
1862
1863 004446
1864 004446 077777 177777
1865 004452 177777 177777
1866 004456 000000 000000
1867 004462 047513 047504
1868 004466 000000
1869
1870
1871
1872
1873
1874
1875
1876 004470 000004
1877 004472 012705 004504
1878 004476 004737 033432
1879
1880 004502 000411
1881
1882 004504
1883 004504 042000 000000
1884 004510 050177 177777
1885 004514 050200 000000
1886 004520 047417 047400
1887 004524 000000
1888
1889
1890
1891
1892
1893
1894
1895 004526 000004
1896 004530 012705 004542
1897 004534 004737 033432
1898
1899 004540 000411
1900
1901 004542
1902 004542 042000 000300
1903 004546 050177 177777
1904 004552 050177 177777
1905 004556 047457 047440
1906 004562 000000

```

```

*****
*TEST 34 TEST OF ADDF INSTR, DATA SET ADDF-4
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

TST34: SCOPE
MOV #ADDF4,RS ; PTR TO TEST DATA SET
JSR PC,#ADDF4 ; GO TEST
BR TST35 ;;

```

```

ADDF4: ; TEST DATA SET ADDF-4:
.WORD LGP,M1 ; INITIAL AC FLOAT NUMBER
.WORD LGN,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD 047513,047504 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 35 TEST OF ADDF INSTR, DATA SET ADDF-5
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

TST35: SCOPE
MOV #ADDF5,RS ; PTR TO TEST DATA SET
JSR PC,#ADDF5 ; GO TEST
BR TST36 ;;

```

```

ADDF5: ; TEST DATA SET ADDF-5:
.WORD 042000,000000 ; INITIAL AC FLOAT NUMBER
.WORD 050177,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 050200,000000 ; EXPECTED FLOAT RESULT
.WORD 047417,047400 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 36 TEST OF ADDF INSTR, DATA SET ADDF-6
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

TST36: SCOPE
MOV #ADDF6,RS ; PTR TO TEST DATA SET
JSR PC,#ADDF6 ; GO TEST
BR TST37 ;;

```

```

ADDF6: ; TEST DATA SET ADDF-6:
.WORD 042000,000000 ; INITIAL AC FLOAT NUMBER
.WORD 050177,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 050177,M1 ; EXPECTED FLOAT RESULT
.WORD 047457,047440 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

1907
1908
1909
1910
1911
1912
1913
1914 004564 000004
1915 004566 012705 004600
1916 004572 004737 033432
1917
1918 004576 000411
1919
1920 004600
1921 004600 141777 177777
1922 004604 150177 177777
1923 004610 150177 177777
1924 004614 047507 047510
1925 004620 000000
1926
1927
1928
1929
1930
1931
1932
1933 004622 000004
1934 004624 012705 004636
1935 004630 004737 033432
1936
1937 004634 000411
1938
1939 004636
1940 004636 141777 177777
1941 004642 150177 177777
1942 004646 150177 177777
1943 004652 047547 047550
1944 004656 000000
1945
1946
1947
1948
1949
1950
1951
1952 004660 000004
1953 004662 012705 004674
1954 004666 004737 033432
1955
1956 004672 000411
1957
1958 004674
1959 004674 040177 177777
1960 004700 032200 000000
1961 004704 040200 000000
1962 004710 047457 047440

```

```

*****
*TEST 37 TEST OF ADDF INSTR, DATA SET ADDF-7
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

†ST37: SCOPE
MOV #ADDF7,R5 ; PTR TO TEST DATA SET
JSR PC,@#ADDF7 ; GO TEST
BR TST40 ;;

```

```

ADDF7: ; TEST DATA SET ADDF-7:
.WORD 141777,M1 ; INITIAL AC FLOAT NUMBER
.WORD 150177,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 150177,M1 ; EXPECTED FLOAT RESULT
.WORD 047507,047510 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 40 TEST OF ADDF INSTR, DATA SET ADDF-10
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

†ST40: SCOPE
MOV #ADDF10,R5 ; PTR TO TEST DATA SET
JSR PC,@#ADDF10 ; GO TEST
BR TST41 ;;

```

```

ADDF10: ; TEST DATA SET ADDF-10:
.WORD 141777,M1 ; INITIAL AC FLOAT NUMBER
.WORD 150177,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 150177,M1 ; EXPECTED FLOAT RESULT
.WORD 047547,047550 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 41 TEST OF ADDF INSTR, DATA SET ADDF-11
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

†ST41: SCOPE
MOV #ADDF11,R5 ; PTR TO TEST DATA SET
JSR PC,@#ADDF11 ; GO TEST
BR TST42 ;;

```

```

ADDF11: ; TEST DATA SET ADDF-11:
.WORD 040177,M1 ; INITIAL AC FLOAT NUMBER
.WORD 032200,000000 ; INITIAL MEM FLOAT NUMBER
.WORD 040200,000000 ; EXPECTED FLOAT RESULT
.WORD 047457,047440 ; FPS: BEFORE, AFTER

```

1963	004714	000000	
1964			
1965			
1966			
1967			
1968			
1969			
1970			
1971	004716	000004	
1972	004720	012705	004732
1973	004724	004737	033432
1974			
1975	004730	000411	
1976			
1977	004732		
1978	004732	140252	125252
1979	004736	140052	125252
1980	004742	140377	177777
1981	004746	047407	047410
1982	004752	000000	
1983			
1984			
1985			
1986			
1987			
1988			
1989			
1990	004754	000004	
1991	004756	012705	004770
1992	004762	004737	033432
1993			
1994	004766	000411	
1995			
1996	004770		
1997	004770	040010	104210
1998	004774	040010	104210
1999	005000	040210	104210
2000	005004	047557	047540
2001	005010	000000	
2002			
2003			
2004			
2005			
2006			
2007			
2008			
2009	005012	000004	
2010	005014	012705	005026
2011	005020	004737	033432
2012			
2013	005024	000411	
2014			
2015	005026		
2016	005026	174177	177777
2017	005032	074177	177776
2018	005036	166200	000000

```

.WORD NA ; FEC AFTER ( 0 = N/A )

*****
*TEST 42 TEST OF ADDF INSTR, DATA SET ADDF-12
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****
TST42: SCOPE
MOV #ADDF12,RS ; PTR TO TEST DATA SET
JSR PC,@ADDF12 ; GO TEST
BR TST43 ;;

ADDF12: ; TEST DATA SET ADDF-12:
.WORD 140252,125252 ; INITIAL AC FLOAT NUMBER
.WORD 140052,125252 ; INITIAL MEM FLOAT NUMBER
.WORD 140377,M1 ; EXPECTED FLOAT RESULT
.WORD 047407,047410 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
*TEST 43 TEST OF ADDF INSTR, DATA SET ADDF-13
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****
TST43: SCOPE
MOV #ADDF13,RS ; PTR TO TEST DATA SET
JSR PC,@ADDF13 ; GO TEST
BR TST44 ;;

ADDF13: ; TEST DATA SET ADDF-13:
.WORD 040010,104210 ; INITIAL AC FLOAT NUMBER
.WORD 040010,104210 ; INITIAL MEM FLOAT NUMBER
.WORD 040210,104210 ; EXPECTED FLOAT RESULT
.WORD 047557,047540 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
*TEST 44 TEST OF ADDF INSTR, DATA SET ADDF-14
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****
TST44: SCOPE
MOV #ADDF14,RS ; PTR TO TEST DATA SET
JSR PC,@ADDF14 ; GO TEST
BR TST45 ;;

ADDF14: ; TEST DATA SET ADDF-14:
.WORD 174177,M1 ; INITIAL AC FLOAT NUMBER
.WORD 074177,M2 ; INITIAL MEM FLOAT NUMBER
.WORD 166200,000000 ; EXPECTED FLOAT RESULT

```

2019 005042 047507 047510
2020 005046 000000

.WORD 047507,047510 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

2021
2022
2023
2024
2025
2026
2027

*TEST 45 TEST OF ADDF INSTR, DATA SET ADDF-15
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, ROUND MODES

2028 005050 000004
2029 005052 012705 005064
2030 005056 004737 033432

TST45: SCOPE
MOV #ADDF15,R5 ; PTR TO TEST DATA SET
JSR PC,#ADDF15 ; GO TEST

2031
2032 005062 000411

BR TST46 ; ;

2033
2034 005064
2035 005064 142200 000000
2036 005070 050177 177777
2037 005074 050177 177776
2038 005100 047417 047400
2039 005104 000000

ADDF15: ; TEST DATA SET ADDF-15:
.WORD 142200,000000 ; INITIAL AC FLOAT NUMBER
.WORD 050177,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 050177,M2 ; EXPECTED FLOAT RESULT
.WORD 047417,047400 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

2040
2041
2042
2043
2044
2045
2046

*TEST 46 TEST OF ADDF INSTR, DATA SET ADDF-16
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES

2047 005106 000004
2048 005110 012705 005122
2049 005114 004737 033432

TST46: SCOPE
MOV #ADDF16,R5 ; PTR TO TEST DATA SET
JSR PC,#ADDF16 ; GO TEST

2050
2051 005120 000411

BR TST47 ; ;

2052
2053 005122
2054 005122 077777 177777
2055 005126 077777 177777
2056 005132 000177 177777
2057 005136 047451 147446
2058 005142 100010

ADDF16: ; TEST DATA SET ADDF-16:
.WORD LGP,M1 ; INITIAL AC FLOAT NUMBER
.WORD LGP,M1 ; INITIAL MEM FLOAT NUMBER
.WORD ZXIMP,M1 ; EXPECTED FLOAT RESULT
.WORD 047451,147446 ; FPS: BEFORE, AFTER
.WORD 100010 ; FEC AFTER (0 = N/A)

2059
2060
2061
2062
2063
2064
2065

*TEST 47 TEST OF ADDF INSTR, DATA SET ADDF-17
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES

2066 005144 000004
2067 005146 012705 005160
2068 005152 004737 033432

TST47: SCOPE
MOV #ADDF17,R5 ; PTR TO TEST DATA SET
JSR PC,#ADDF17 ; GO TEST

2069
2070 005156 000411

BR TST50 ; ;

2071
2072 005160
2073 005160 104000 000000
2074 005164 004000 000001

ADDF17: ; TEST DATA SET ADDF-17:
.WORD 104000,0 ; INITIAL AC FLOAT NUMBER
.WORD 004000,1 ; INITIAL MEM FLOAT NUMBER

E04

FPU ADVANCED INSTR TESTS
D0FPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 45
T47 TEST OF ADDF INSTR, DATA SET ADDF-17

2075 005170 076200 000000
2076 005174 047517 147500
2077 005200 100012

.WORD 076200,0 ; EXPECTED FLOAT RESULT
.WORD 047517,147500 ; FPS: BEFORE AFTER
.WORD 100012 ; FEC AFTER (0 = N/A)

2078
2079

2080
2081

2082
2083

2084
2085

2086
2087

2088
2089

2090
2091

2092
2093

2094
2095

2096
2097

2098
2099

2100
2101

2102
2103

2104
2105

2106
2107

2108
2109

2110
2111

2112
2113

2114
2115

2116
2117

2118
2119

2120
2121

2122
2123

2124
2125

2126
2127

2128
2129

2130

*TEST 50 TEST OF ADDF INSTR, DATA SET ADDF-20
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES

TST50: SCOPE
MOV #ADDF20,R5 ; PTR TO TEST DATA SET
JSR PC,#ADDF20 ; GO TEST
BR TST51 ;

ADDF20: ; TEST DATA SET ADDF-20:
.WORD LGN,M1 ; INITIAL AC FLOAT NUMBER
.WORD MO,0 ; INITIAL MEM FLOAT NUMBER
.WORD LGN,M1 ; EXPECTED FLOAT RESULT
.WORD 047543,147543 ; FPS: BEFORE AFTER
.WORD 100014 ; FEC AFTER (0 = N/A)

*TEST 51 TEST OF ADDF INSTR, DATA SET ADDF-21
* OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES

TST51: SCOPE
MOV #ADDF21,R5 ; PTR TO TEST DATA SET
JSR PC,#ADDF21 ; GO TEST
BR TST52 ;

ADDF21: ; TEST DATA SET ADDF-21:
.WORD LGN,M1 ; INITIAL AC FLOAT NUMBER
.WORD LGN,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD 046511,046506 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER (0 = N/A)

*TEST 52 TEST OF ADDF INSTR, DATA SET ADDF-22
* -0 INTERRUPT ENABLE OFF, ALL OTHERS ON
* LONG FLOAT, LONG INTEGER, ROUND MODES

TST52: SCOPE
MOV #ADDF22,R5 ; PTR TO TEST DATA SET
JSR PC,#ADDF22 ; GO TEST
BR TST53 ;

ADDF22: ; TEST DATA SET ADDF-22:
.WORD ALTP,ALTP ; INITIAL AC FLOAT NUMBER

F04

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 46
T52 TEST OF ADDF INSTR, DATA SET ADDF-22

2131 005316 100177 177777
2132 005322 052525 052525
2133 005326 043717 043700
2134 005332 000000

.WORD ZX1MN,M1 ; INITIAL MEM FLOAT NUMBER
.WORD ALTP,ALTP ; EXPECTED FLOAT RESULT
.WORD 043717,043700 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156

005334 000004
005336 012705 005350
005342 004737 033432
005346 000411
005350
005350 004000 000001
005354 104000 000000
005360 000000 000000
005364 045413 045404
005370 000000

*TEST 53 TEST OF ADDF INSTR, DATA SET ADDF-23
* UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
* SHORT FLOAT, SHORT INTEGER, ROUND MODES

↑ST53: SCOPE
MOV #ADDF23,R5 ; PTR TO TEST DATA SET
JSR PC,#ADDF1 ; GO TEST
BR TST54 ;;

ADDF23: ; TEST DATA SET ADDF-23:
.WORD 004000,000001 ; INITIAL AC FLOAT NUMBER
.WORD 104000,000000 ; INITIAL MEM FLOAT NUMBER
.WORD 000000,000000 ; EXPECTED FLOAT RESULT
.WORD 045413,045404 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

```

2157
2158
2159
2160
2161
2162 005372 000004
2163 005374 012705 005406
2164 005400 004737 033602
2165
2166 005404 000417
2167
2168 005406
2169 005406 000177 177777 177777
2170 005414 177777
2171 005416 000000 000000 000000
2172 005424 000000
2173 005426 000000 000000 000000
2174 005434 000000
2175 005436 047713 047704
2176 005442 000000
2177
2178
2179
2180
2181
2182
2183
2184 005444 000004
2185 005446 012705 005460
2186 005452 004737 033602
2187
2188 005456 000417
2189
2190 005460
2191 005460 125252 125252 125252
2192 005466 125252
2193 005470 000000 000000 000000
2194 005476 000000
2195 005500 125252 125252 125252
2196 005506 125252
2197 005510 047747 047750
2198 005514 000000
2199
2200
2201
2202
2203
2204
2205
2206 005516 000004
2207 005520 012705 005532
2208 005524 004737 033602
2209
2210 005530 000417
2211
2212 005532

```

```

*****
*TEST 54 TEST OF ADDO INSTR, DATA SET ADDO-1
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
↑ST54: SCOPE
MOV #ADD01,R5 ; PTR TO TEST DATA SET
JSR PC,@#ADD0T ; GO TEST
BR TST55 ;;

ADD01: ; TEST DATA SET ADDO-1:
.WORD ZXIMP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD 0,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 047713,047704 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
*TEST 55 TEST OF ADDO INSTR, DATA SET ADDO-2
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****
↑ST55: SCOPE
MOV #ADD02,R5 ; PTR TO TEST DATA SET
JSR PC,@#ADD0T ; GO TEST
BR TST56 ;;

ADD02: ; TEST DATA SET ADDO-2:
.WORD ALTN,ALTN,ALTN,ALTN ; INITIAL AC FLOAT NUMBER
.WORD 0,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD ALTN,ALTN,ALTN,ALTN ; EXPECTED FLOAT RESULT
.WORD 047747,047750 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
*TEST 56 TEST OF ADDO INSTR, DATA SET ADDO-3
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****
↑ST56: SCOPE
MOV #ADD03,R5 ; PTR TO TEST DATA SET
JSR PC,@#ADD0T ; GO TEST
BR TST57 ;;

ADD03: ; TEST DATA SET ADDO-3:

```

H04

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 48
T56 TEST OF ADDO INSTR, DATA SET ADDO-3

2213	005532	000177	177777	177777	.WORD	ZXIMP,M1,M1,M1	; INITIAL AC FLOAT NUMBER
2214	005540	177777					
2215	005542	052525	052525	052525	.WORD	ALTP,ALTP,ALTP,ALTP	; INITIAL MEM FLOAT NUMBER
2216	005550	052525					
2217	005552	052525	052525	052525	.WORD	ALTP,ALTP,ALTP,ALTP	; EXPECTED FLOAT RESULT
2218	005560	052525					
2219	005562	047617	047600		.WORD	047617,047600	; FPS: BEFORE, AFTER
2220	005566	000000			.WORD	NA	; FEC AFTER (0 = N/A)

2221
2222
2223

```

*****
; *TEST 57 TEST OF ADDO INSTR, DATA SET ADDO-4
; *
; * ALL INTERRUPT ENABLES ON
; * LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

2228	005570	000004			TST57:	SCOPE	
2229	005572	012705	005604		MOV	#ADD04,RS	; PTR TO TEST DATA SET
2230	005576	004737	033602		JSR	PC,#ADD0T	; GO TEST
2231							
2232	005602	000417			BR	TST60	::

2233							
2234	005604				ADD04:	; TEST DATA SET ADDO-4:	
2235	005604	177777	177777	177777	.WORD	LGN,M1,M1,M1	; INITIAL AC FLOAT NUMBER
2236	005612	177777					
2237	005614	077777	177777	177777	.WORD	LGP,M1,M1,M1	; INITIAL MEM FLOAT NUMBER
2238	005622	177777					
2239	005624	000000	000000	000000	.WORD	0,0,0,0	; EXPECTED FLOAT RESULT
2240	005632	000000					
2241	005634	047653	047644		.WORD	047653,047644	; FPS: BEFORE, AFTER
2242	005640	000000			.WORD	NA	; FEC AFTER (0 = N/A)

2243
2244
2245

```

*****
; *TEST 60 TEST OF ADDO INSTR, DATA SET ADDO-5
; *
; * ALL INTERRUPT ENABLES ON
; * LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

2250	005642	000004			TST60:	SCOPE	
2251	005644	012705	005656		MOV	#ADD05,RS	; PTR TO TEST DATA SET
2252	005650	004737	033602		JSR	PC,#ADD0T	; GO TEST
2253							
2254	005654	000417			BR	TST61	::

2255							
2256	005656				ADD05:	; TEST DATA SET ADDO-5:	
2257	005656	166177	177777	177777	.WORD	166177,M1,M1,M1	; INITIAL AC FLOAT NUMBER
2258	005664	177777					
2259	005666	150000	000000	000000	.WORD	150000,0,0,0	; INITIAL MEM FLOAT NUMBER
2260	005674	000000					
2261	005676	166200	000000	000000	.WORD	166200,0,0,0	; EXPECTED FLOAT RESULT
2262	005704	000000					
2263	005706	047607	047610		.WORD	047607,047610	; FPS: BEFORE, AFTER
2264	005712	000000			.WORD	NA	; FEC AFTER (0 = N/A)

2265
2266
2267

```

*****
; *TEST 61 TEST OF ADDO INSTR, DATA SET ADDO-6

```

2268


```

2269
2270
2271
2272 005714 000004
2273 005716 012705 005730
2274 005722 004737 033602
2275
2276 005726 000417
2277
2278 005730
2279 005730 166177 177777 177777
2280 005736 177777
2281 005740 150000 000000 000000
2282 005746 000000
2283 005750 166177 177777 177777
2284 005756 177777
2285 005760 047647 047650
2286 005764 000000
2287
2288
2289
2290
2291
2292
2293
2294 005766 000004
2295 005770 012705 006002
2296 005774 004737 033602
2297
2298 006000 000417
2299
2300 006002
2301 006002 066177 177777 177777
2302 006010 177777
2303 006012 047777 177777 177777
2304 006020 177777
2305 006022 066177 177777 177777
2306 006030 177777
2307 006032 047717 047700
2308 006036 000000
2309
2310
2311
2312
2313
2314
2315
2316 006040 000004
2317 006042 012705 006054
2318 006046 004737 033602
2319
2320 006052 000417
2321
2322 006054
2323 006054 066177 177777 177777
2324 006062 177777

```

```

;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST61: SCOPE
MOV #ADD06,R5 ; PTR TO TEST DATA SET
JSR PC,@#ADD0T ; GO TEST
BR TST62 ;;

ADD06: ; TEST DATA SET ADDO-6:
.WORD 166177,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD 150000,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 166177,M1,M1,M1 ; EXPECTED FLOAT RESULT
.WORD 047647,047650 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
*TEST 62 TEST OF ADDO INSTR, DATA SET ADDO-7
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
TST62: SCOPE
MOV #ADD07,R5 ; PTR TO TEST DATA SET
JSR PC,@#ADD0T ; GO TEST
BR TST63 ;;

ADD07: ; TEST DATA SET ADDO-7:
.WORD 066177,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD 047777,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 066177,M1,M1,M1 ; EXPECTED FLOAT RESULT
.WORD 047717,047700 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
*TEST 63 TEST OF ADDO INSTR, DATA SET ADDO-10
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****
TST63: SCOPE
MOV #ADD10,R5 ; PTR TO TEST DATA SET
JSR PC,@#ADD0T ; GO TEST
BR TST64 ;;

ADD10: ; TEST DATA SET ADDO-10:
.WORD 066177,M1,M1,M1 ; INITIAL AC FLOAT NUMBER

```

J04

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 50
T63 TEST OF ADDO INSTR, DATA SET ADDO-10

2325	006064	047777	177777	177777	.WORD	047777,M1,M1,M1	; INITIAL MEM FLOAT NUMBER
2326	006072	177777					
2327	006074	066177	177777	177777	.WORD	066177,M1,M1,M1	; EXPECTED FLOAT RESULT
2328	006102	177777					
2329	006104	047757	047740		.WORD	047757,047740	; FPS: BEFORE AFTER
2330	006110	000000			.WORD	NA	; FEC AFTER (0 = N/A)

```

2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359

```

```

*****
*TEST 64 TEST OF ADDO INSTR, DATA SET ADDO-11
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

†T64: SCOPE
      MOV #ADD011,R5 ; PTR TO TEST DATA SET
      JSR PC,@#ADD0T ; GO TEST
      BR TST65 ;;

```

```

ADD011: ; TEST DATA SET ADDO-11:
        .WORD 004010,P13Z,P13Z,P13Z ; INITIAL AC FLOAT NUMBER
        .WORD 004010,P13Z,P13Z,P13Z ; INITIAL MEM FLOAT NUMBER
        .WORD 004210,P13Z,P13Z,P13Z ; EXPECTED FLOAT RESULT
        .WORD 047617,047600 ; FPS: BEFORE AFTER
        .WORD NA ; FEC AFTER ( 0 = N/A )

```

```

2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380

```

```

*****
*TEST 65 TEST OF ADDO INSTR, DATA SET ADDO-12
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

†T65: SCOPE
      MOV #ADD012,R5 ; PTR TO TEST DATA SET
      JSR PC,@#ADD0T ; GO TEST
      BR TST66 ;;

```

```

ADD012: ; TEST DATA SET ADDO-12:
        .WORD 122200,0,0,0 ; INITIAL AC FLOAT NUMBER
        .WORD 140177,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
        .WORD 140200,0,0,0 ; EXPECTED FLOAT RESULT
        .WORD 047747,047750 ; FPS: BEFORE AFTER
        .WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 66 TEST OF ADDO INSTR, DATA SET ADDO-13
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****

```

K04

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 51
T66 TEST OF ADDO INSTR, DATA SET ADDO-13

```

2381
2382 006236 000004
2383 006240 012705 006252
2384 006244 004737 033602
2385
2386 006250 000417
2387
2388 006252
2389 006252 042252 125252 125252
2390 006260 125252
2391 006262 042052 125252 125252
2392 006270 125252
2393 006272 042377 177777 177777
2394 006300 177777
2395 006302 047717 047700
2396 006306 000000
2397
2398
2399
2400
2401
2402
2403
2404 006310 000004
2405 006312 012705 006324
2406 006316 004737 033602
2407
2408 006322 000417
2409
2410 006324
2411 006324 074177 177777 177777
2412 006332 177777
2413 006334 174177 177777 177777
2414 006342 177776
2415 006344 056200 000000 000000
2416 006352 000000
2417 006354 047617 047600
2418 006360 000000
2419
2420
2421
2422
2423
2424
2425
2426 006362 000004
2427 006364 012705 006376
2428 006370 004737 033602
2429
2430 006374 000417
2431
2432 006376
2433 006376 132200 000000 000000
2434 006404 000000
2435 006406 050177 177777 177777
2436 006414 177777

```

```

*****
TST66: SCOPE
MOV #ADD013,RS ; PTR TO TEST DATA SET
JSR PC,@ADD0T ; GO TEST
BR TST67 ;;
ADD013: ; TEST DATA SET ADDO-13:
.WORD 042252,ALTN,ALTN,ALTN ; INITIAL AC FLOAT NUMBER
.WORD 042052,ALTN,ALTN,ALTN ; INITIAL MEM FLOAT NUMBER
.WORD 042377,M1,M1,M1 ; EXPECTED FLOAT RESULT
.WORD 047717,047700 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
*****
*TEST 67 TEST OF ADDO INSTR, DATA SET ADDO-14
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****
TST67: SCOPE
MOV #ADD014,RS ; PTR TO TEST DATA SET
JSR PC,@ADD0T ; GO TEST
BR TST70 ;;
ADD014: ; TEST DATA SET ADDO-14:
.WORD 074177,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD 174177,M1,M1,M2 ; INITIAL MEM FLOAT NUMBER
.WORD 056200,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 047617,047600 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
*****
*TEST 70 TEST OF ADDO INSTR, DATA SET ADDO-15
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
TST70: SCOPE
MOV #ADD015,RS ; PTR TO TEST DATA SET
JSR PC,@ADD0T ; GO TEST
BR TST71 ;;
ADD015: ; TEST DATA SET ADDO-15:
.WORD 132200,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD 050177,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER

```

2437	006416	050177	177777	177777	.WORD	050177,M1,M1,M2 ; EXPECTED FLOAT RESULT
2438	006424	177776				
2439	006426	047717	047700		.WORD	047717,047700 ; FPS: BEFORE, AFTER
2440	006432	000000			.WORD	NA ; FEC AFTER (0 = N/A)

```

*****
*TEST 71      TEST OF ADDO INSTR, DATA SET ADDO-16
*            ALL INTERRUPT ENABLES ON
*            LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

2447	006434	000004			†ST71: SCOPE	
2449	006436	012705	006450		MOV	#ADD016,R5 ; PTR TO TEST DATA SET
2450	006442	004737	033602		JSR	PC,#ADD01 ; GO TEST
2448					BR	TST72 ; ;

```

ADD016: ; TEST DATA SET ADDO-16:
.WORD   LGP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD   ZX1MN,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD   LGP,M1,M1,M1 ; EXPECTED FLOAT RESULT
.WORD   047603,147603 ; FPS: BEFORE, AFTER
.WORD   100014 ; FEC AFTER ( 0 = N/A )

```

2451	006446	000417				
2452						
2453						
2454	006450					
2455	006450	077777	177777	177777		
2456	006456	177777				
2457	006460	100177	177777	177777		
2458	006466	177777				
2459	006470	077777	177777	177777		
2460	006476	177777				
2461	006500	047603	147603			
2462	006504	100014				

```

*****
*TEST 72      TEST OF ADDO INSTR, DATA SET ADDO-17
*            ALL INTERRUPT ENABLES ON
*            LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

2469	006506	000004			†ST72: SCOPE	
2471	006510	012705	006522		MOV	#ADD017,R5 ; PTR TO TEST DATA SET
2472	006514	004737	033602		JSR	PC,#ADD01 ; GO TEST
2470					BR	TST73 ; ;

```

ADD017: ; TEST DATA SET ADDO-17:
.WORD   102000,0,0,1 ; INITIAL AC FLOAT NUMBER
.WORD   002000,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD   164200,0,0,0 ; EXPECTED FLOAT RESULT
.WORD   047647,147650 ; FPS: BEFORE, AFTER
.WORD   100012 ; FEC AFTER ( 0 = N/A )

```

2473						
2474	006520	000417				
2475						
2476	006522					
2477	006522	102000	000000	000000		
2478	006530	000001				
2479	006532	002000	000000	000000		
2480	006540	000000				
2481	006542	164200	000000	000000		
2482	006550	000000				
2483	006552	047647	147650			
2484	006556	100012				

```

*****
*TEST 73      TEST OF ADDO INSTR, DATA SET ADDO-20
*            ALL INTERRUPT ENABLES ON
*            LONG FLOAT, LONG INTEGER, ROUND MODES
*****

```

2485						
2486						
2487						
2488						
2489						
2490						
2491						
2492	006560	000004			†ST73: SCOPE	

M04

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 53
T73 TEST OF ADDO INSTR, DATA SET ADDO-20

2493	006562	012705	006574	MOV	#ADD020,R5	; PTR TO TEST DATA SET
2494	006566	004737	033602	JSR	PC,2#ADD0T	; GO TEST
2495						
2496	006572	000417		BR	TST74	::
2497						
2498	006574			ADD020:	; TEST DATA SET ADDO-20:	
2499	006574	177777	177777	177777	.WORD LGN,M1,M1,M1	; INITIAL AC FLOAT NUMBER
2500	006602	177777				
2501	006604	177777	177777	177777	.WORD LGN,M1,M1,M1	; INITIAL MEM FLOAT NUMBER
2502	006612	177777				
2503	006614	100177	177777	177777	.WORD ZX1MN,M1,M1,M1	; EXPECTED FLOAT RESULT
2504	006622	177777				
2505	006624	047701	147716		.WORD 047701,147716	; FPS: BEFORE, AFTER
2506	006630	100010			.WORD 100010	; FEC AFTER (0 = N/A)
2507						
2508						

```

:*****
:TEST 74 TEST OF ADDO INSTR, DATA SET ADDO-21
:* UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
:* LONG FLOAT, LONG INTEGER, ROUND MODES
:*****

```

2514	006632	000004		TST74:	SCOPE	
2515	006634	012705	006646	MOV	#ADD021,R5	; PTR TO TEST DATA SET
2516	006640	004737	033602	JSR	PC,2#ADD0T	; GO TEST
2517						
2518	006644	000417		BR	TST75	::
2519						
2520	006646			ADD021:	; TEST DATA SET ADDO-21:	
2521	006646	002000	000000	000000	.WORD 002000,0,0,0	; INITIAL AC FLOAT NUMBER
2522	006654	000000				
2523	006656	102000	000000	000000	.WORD 102000,0,0,2	; INITIAL MEM FLOAT NUMBER
2524	006664	000002				
2525	006666	000000	000000	000000	.WORD 0,0,0,0	; EXPECTED FLOAT RESULT
2526	006674	000000				
2527	006676	045713	045704		.WORD 045713,045704	; FPS: BEFORE, AFTER
2528	006702	000000			.WORD NA	; FEC AFTER (0 = N/A)
2529						

```

:*****
:TEST 75 TEST OF ADDO INSTR, DATA SET ADDO-22
:* -O INTERRUPT ENABLE OFF, ALL OTHERS ON
:* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
:*****

```

2536	006704	000004		TST75:	SCOPE	
2537	006706	012705	006720	MOV	#ADD022,R5	; PTR TO TEST DATA SET
2538	006712	004737	033602	JSR	PC,2#ADD0T	; GO TEST
2539						
2540	006716	000417		BR	TST76	::
2541						
2542	006720			ADD022:	; TEST DATA SET ADDO-22:	
2543	006720	077777	000000	177777	.WORD LGP,0,M1,0	; INITIAL AC FLOAT NUMBER
2544	006726	000000				
2545	006730	100000	000000	000000	.WORD M0,0,0,0	; INITIAL MEM FLOAT NUMBER
2546	006736	000000				
2547	006740	077777	000000	177777	.WORD LGP,0,M1,0	; EXPECTED FLOAT RESULT
2548	006746	000000				

FPU ADVANCED INSTR TESTS
D0FPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 54
T75 TEST OF ADDO INSTR, DATA SET ADDO-22

2549	006750	043757	043740
2550	006754	000000	
2551			
2552			
2553			
2554			
2555			
2556			
2557			
2558	006756	000004	
2559	006760	012705	006772
2560	006764	004737	033602
2561			
2562	006770	000417	
2563			
2564	006772		
2565	006772	077777	177777 177777
2566	007000	177777	
2567	007002	077777	177777 177777
2568	007010	177777	
2569	007012	000000	000000 000000
2570	007020	000000	
2571	007022	046611	046606
2572	007026	000000	
2573			
2574			
2575			

.WORD 043757,043740 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

```

*****
; TEST 76 TEST OF ADDO INSTR, DATA SET ADDO-23
; * OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
; * LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

†ST76: SCOPE
MOV #ADD023,R5 ; PTR TO TEST DATA SET
JSR PC,@#ADD0T ; GO TEST
BR TST77 ;;

```

```

ADD023: ; TEST DATA SET ADDO-23:
.WORD LGP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD LGP,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 046611,046606 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

2576
2577
2578
2579
2580
2581 007030 000004
2582 007032 012705 007044
2583 007036 004737 033772
2584
2585 007042 000411
2586
2587 007044
2588 007044 000000 000000
2589 007050 000000 000000
2590 007054 000000 000000
2591 007060 047413 047404
2592 007064 000000
2593
2594
2595
2596
2597
2598
2599
2600 007066 000004
2601 007070 012705 007102
2602 007074 004737 033772
2603
2604 007100 000411
2605
2606 007102
2607 007102 000177 177777
2608 007106 000177 125252
2609 007112 000000 000000
2610 007116 047453 047444
2611 007122 000000
2612
2613
2614
2615
2616
2617
2618
2619 007124 000004
2620 007126 012705 007140
2621 007132 004737 033772
2622
2623 007136 000411
2624
2625 007140
2626 007140 000177 052525
2627 007144 100345 123456
2628 007150 000345 123456
2629 007154 047517 047500
2630 007160 000000
2631

```

```

*****
*TEST 77      TEST OF SUBF INSTR, DATA SET SUBF-1
*
*            ALL INTERRUPT ENABLES ON
*            SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****
↑ST77: SCOPE
      MOV      #SUBF1,R5      ; PTR TO TEST DATA SET
      JSR      PC,@#SUBFT    ; GO TEST

      BR      TST100        ;;

SUBF1: ; TEST DATA SET SUBF-1:
      .WORD   0,0            ; INITIAL AC FLOAT NUMBER
      .WORD   0,0            ; INITIAL MEM FLOAT NUMBER
      .WORD   0,0            ; EXPECTED FLOAT RESULT
      .WORD   047413,047404 ; FPS: BEFORE, AFTER
      .WORD   NA             ; FEC AFTER ( 0 = N/A )

*****
*TEST 100     TEST OF SUBF INSTR, DATA SET SUBF-2
*
*            ALL INTERRUPT ENABLES ON
*            SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
↑ST100: SCOPE
      MOV      #SUBF2,R5     ; PTR TO TEST DATA SET
      JSR      PC,@#SUBFT   ; GO TEST

      BR      TST101        ;;

SUBF2: ; TEST DATA SET SUBF-2:
      .WORD   ZXIMP,M1      ; INITIAL AC FLOAT NUMBER
      .WORD   ZXIMP,ALTN    ; INITIAL MEM FLOAT NUMBER
      .WORD   0,0            ; EXPECTED FLOAT RESULT
      .WORD   047453,047444 ; FPS: BEFORE, AFTER
      .WORD   NA             ; FEC AFTER ( 0 = N/A )

*****
*TEST 101     TEST OF SUBF INSTR, DATA SET SUBF-3
*
*            ALL INTERRUPT ENABLES ON
*            SHORT FLOAT, LONG INTEGER, ROUND MODES
*****
↑ST101: SCOPE
      MOV      #SUBF3,R5     ; PTR TO TEST DATA SET
      JSR      PC,@#SUBFT   ; GO TEST

      BR      TST102        ;;

SUBF3: ; TEST DATA SET SUBF-3:
      .WORD   ZXIMP,ALTP    ; INITIAL AC FLOAT NUMBER
      .WORD   100345,123456 ; INITIAL MEM FLOAT NUMBER
      .WORD   000345,123456 ; EXPECTED FLOAT RESULT
      .WORD   047517,047500 ; FPS: BEFORE, AFTER
      .WORD   NA             ; FEC AFTER ( 0 = N/A )

```

2632			
2633			
2634			
2635			
2636			
2637			
2638	007162	000004	
2639	007164	012705	007176
2640	007170	004737	033772
2641			
2642	007174	000411	
2643			
2644	007176		
2645	007176	040200	000000
2646	007202	040200	000000
2647	007206	000000	000000
2648	007212	047553	047544
2649	007216	000000	
2650			
2651			
2652			
2653			
2654			
2655			
2656			
2657	007220	000004	
2658	007222	012705	007234
2659	007226	004737	033772
2660			
2661	007232	000411	
2662			
2663	007234		
2664	007234	140200	000000
2665	007240	140200	000000
2666	007244	000000	000000
2667	007250	047413	047404
2668	007254	000000	
2669			
2670			
2671			
2672			
2673			
2674			
2675			
2676	007256	000004	
2677	007260	012705	007272
2678	007264	004737	033772
2679			
2680	007270	000411	
2681			
2682	007272		
2683	007272	150365	052525
2684	007276	047252	125252
2685	007302	150377	177777
2686	007306	047447	047450
2687	007312	000000	

```

*****
:TEST 102      TEST OF SUBF INSTR, DATA SET SUBF-4
:*            ALL INTERRUPT ENABLES ON
:*            SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

TST102: SCOPE
MOV      #SUBF4,R5      ; PTR TO TEST DATA SET
JSR      PC,@#SUBFT    ; GO TEST
BR       TST103        ;;

```

```

SUBF4: ; TEST DATA SET SUBF-4:
.WORD   FIP,0          ; INITIAL AC FLOAT NUMBER
.WORD   FIP,0          ; INITIAL MEM FLOAT NUMBER
.WORD   0,0           ; EXPECTED FLOAT RESULT
.WORD   047553,047544 ; FPS: BEFORE, AFTER
.WORD   NA             ; FEC AFTER ( 0 = N/A )

```

```

*****
:TEST 103      TEST OF SUBF INSTR, DATA SET SUBF-5
:*            ALL INTERRUPT ENABLES ON
:*            SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

TST103: SCOPE
MOV      #SUBF5,R5      ; PTR TO TEST DATA SET
JSR      PC,@#SUBFT    ; GO TEST
BR       TST104        ;;

```

```

SUBF5: ; TEST DATA SET SUBF-5:
.WORD   FIN,0          ; INITIAL AC FLOAT NUMBER
.WORD   FIN,0          ; INITIAL MEM FLOAT NUMBER
.WORD   0,0           ; EXPECTED FLOAT RESULT
.WORD   047413,047404 ; FPS: BEFORE, AFTER
.WORD   NA             ; FEC AFTER ( 0 = N/A )

```

```

*****
:TEST 104      TEST OF SUBF INSTR, DATA SET SUBF-6
:*            ALL INTERRUPT ENABLES ON
:*            SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

TST104: SCOPE
MOV      #SUBF6,R5      ; PTR TO TEST DATA SET
JSR      PC,@#SUBFT    ; GO TEST
BR       TST105        ;;

```

```

SUBF6: ; TEST DATA SET SUBF-6:
.WORD   150365,ALTP    ; INITIAL AC FLOAT NUMBER
.WORD   047252,ALTN    ; INITIAL MEM FLOAT NUMBER
.WORD   150377,M1      ; EXPECTED FLOAT RESULT
.WORD   047447,047450 ; FPS: BEFORE, AFTER
.WORD   NA             ; FEC AFTER ( 0 = N/A )

```



```

2688
2689
2690
2691
2692
2693
2694
2695 007314 000004
2696 007316 012705 007330
2697 007322 004737 033772
2698
2699 007326 000411
2700
2701 007330
2702 007330 050365 052525
2703 007334 147252 125252
2704 007340 050400 000000
2705 007344 047517 047500
2706 007350 000000
2707
2708
2709
2710
2711
2712
2713
2714 007352 000004
2715 007354 012705 007366
2716 007360 004737 033772
2717
2718 007364 000411
2719
2720 007366
2721 007366 077777 177777
2722 007372 100177 177777
2723 007376 077777 177777
2724 007402 047555 147555
2725 007406 100014
2726
2727
2728
2729
2730
2731
2732
2733 007410 000004
2734 007412 012705 007424
2735 007416 004737 033772
2736
2737 007422 000411
2738
2739 007424
2740 007424 077777 177777
2741 007430 100177 177777
2742 007434 077777 177777
2743 007440 043457 043440

```

```

*****
*TEST 105 TEST OF SUBF INSTR, DATA SET SUBF-7
*
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

TST105: SCOPE
MOV #SUBF7,R5 ; PTR TO TEST DATA SET
JSR PC,@SUBFT ; GO TEST
BR TST106 ;;

```

```

SUBF7: ; TEST DATA SET SUBF-7:
.WORD 050365,ALTP ; INITIAL AC FLOAT NUMBER
.WORD 147252,ALTN ; INITIAL MEM FLOAT NUMBER
.WORD 050400,0 ; EXPECTED FLOAT RESULT
.WORD 047517,047500 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 106 TEST OF SUBF INSTR, DATA SET SUBF-10
*
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

TST106: SCOPE
MOV #SUBF10,R5 ; PTR TO TEST DATA SET
JSR PC,@SUBFT ; GO TEST
BR TST107 ;;

```

```

SUBF10: ; TEST DATA SET SUBF-10:
.WORD LGP,M1 ; INITIAL AC FLOAT NUMBER
.WORD ZX1MN,M1 ; INITIAL MEM FLOAT NUMBER
.WORD LGP,M1 ; EXPECTED FLOAT RESULT
.WORD 047555,147555 ; FPS: BEFORE, AFTER
.WORD 100014 ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 107 TEST OF SUBF INSTR, DATA SET SUBF-11
*
* -0 INTERRUPT ENABLE OFF, ALL OTHERS ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

TST107: SCOPE
MOV #SUBF11,R5 ; PTR TO TEST DATA SET
JSR PC,@SUBFT ; GO TEST
BR TST110 ;;

```

```

SUBF11: ; TEST DATA SET SUBF-11:
.WORD LGP,M1 ; INITIAL AC FLOAT NUMBER
.WORD ZX1MN,M1 ; INITIAL MEM FLOAT NUMBER
.WORD LGP,M1 ; EXPECTED FLOAT RESULT
.WORD 043457,043440 ; FPS: BEFORE, AFTER

```

2744	007444	000000	
2745			
2746			
2747			
2748			
2749			
2750			
2751			
2752	007446	000004	
2753	007450	012705	007462
2754	007454	004737	033772
2755			
2756	007460	000411	
2757			
2758	007462		
2759	007462	177777	177777
2760	007466	071600	000000
2761	007472	177777	177777
2762	007476	047447	047450
2763	007502	000000	
2764			
2765			
2766			
2767			
2768			
2769			
2770			
2771	007504	000004	
2772	007506	012705	007520
2773	007512	004737	033772
2774			
2775	007516	000411	
2776			
2777	007520		
2778	007520	177777	177777
2779	007524	071600	000000
2780	007530	100000	000000
2781	007534	047501	147516
2782	007540	100010	
2783			
2784			
2785			
2786			
2787			
2788			
2789			
2790	007542	000004	
2791	007544	012705	007556
2792	007550	004737	033772
2793			
2794	007554	000411	
2795			
2796	007556		
2797	007556	177777	177777
2798	007562	071600	000000
2799	007566	000000	000000

```

.WORD NA ; FEC AFTER ( 0 = N/A )

*****
*TEST 110 TEST OF SUBF INSTR, DATA SET SUBF-12
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST110: SCOPE
MOV #SUBF12,R5 ; PTR TO TEST DATA SET
JSR PC,#SUBFT ; GO TEST
BR TST111 ;;

SUBF12: ; TEST DATA SET SUBF-12:
.WORD LGN,M1 ; INITIAL AC FLOAT NUMBER
.WORD 071600,0 ; INITIAL MEM FLOAT NUMBER
.WORD LGN,M1 ; EXPECTED FLOAT RESULT
.WORD 047447,047450 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
*TEST 111 TEST OF SUBF INSTR, DATA SET SUBF-13
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****
TST111: SCOPE
MOV #SUBF13,R5 ; PTR TO TEST DATA SET
JSR PC,#SUBFT ; GO TEST
BR TST112 ;;

SUBF13: ; TEST DATA SET SUBF-13:
.WORD LGN,M1 ; INITIAL AC FLOAT NUMBER
.WORD 071600,0 ; INITIAL MEM FLOAT NUMBER
.WORD M0,0 ; EXPECTED FLOAT RESULT
.WORD 047501,147516 ; FPS: BEFORE AFTER
.WORD 100010 ; FEC AFTER ( 0 = N/A )

*****
*TEST 112 TEST OF SUBF INSTR, DATA SET SUBF-14
* OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****
TST112: SCOPE
MOV #SUBF14,R5 ; PTR TO TEST DATA SET
JSR PC,#SUBFT ; GO TEST
BR TST113 ;;

SUBF14: ; TEST DATA SET SUBF-14:
.WORD LGN,M1 ; INITIAL AC FLOAT NUMBER
.WORD 071600,0 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT

```

F05

FPU ADVANCED INSTR TESTS
D0FPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 59
T112 TEST OF SUBF INSTR, DATA SET SUBF-14

2800 007572 046511 046506
2801 007576 000000
2802
2803
2804
2805
2806
2807
2808
2809 007600 000004
2810 007602 012705 007614
2811 007606 004737 033772
2812
2813 007612 000411
2814
2815 007614
2816 007614 004200 000000
2817 007620 004200 000001
2818 007624 176400 000000
2819 007630 047447 147450
2820 007634 100012
2821
2822
2823
2824
2825
2826
2827
2828 007636 000004
2829 007640 012705 007652
2830 007644 004737 033772
2831
2832 007650 000411
2833
2834 007652
2835 007652 004200 000000
2836 007656 004200 000001
2837 007662 000000 000000
2838 007666 045453 045444
2839 007672 000000
2840
2841
2842

.WORD 046511,046506 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

*TEST 113 TEST OF SUBF INSTR, DATA SET SUBF-15
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES

↑ST113: SCOPE
MOV #SUBF15,R5 ; PTR TO TEST DATA SET
JSR PC,@#SUBFT ; GO TEST
BR TST114 ;;

SUBF15: ; TEST DATA SET SUBF-15:
.WORD 004200,0 ; INITIAL AC FLOAT NUMBER
.WORD 004200,1 ; INITIAL MEM FLOAT NUMBER
.WORD 176400,0 ; EXPECTED FLOAT RESULT
.WORD 047447,147450 ; FPS: BEFORE, AFTER
.WORD 100012 ; FEC AFTER (0 = N/A)

*TEST 114 TEST OF SUBF INSTR, DATA SET SUBF-16
* UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES

↑ST114: SCOPE
MOV #SUBF16,R5 ; PTR TO TEST DATA SET
JSR PC,@#SUBFT ; GO TEST
BR TST115 ;;

SUBF16: ; TEST DATA SET SUBF-16:
.WORD 004200,0 ; INITIAL AC FLOAT NUMBER
.WORD 004200,1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD 045453,045444 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

```

2843
2844
2845
2846
2847
2848 007674 000004
2849 007676 012705 007710
2850 007702 004737 034142
2851
2852 007706 000417
2853
2854 007710
2855 007710 000000 000000 000000
2856 007716 000000
2857 007720 000000 000000 000000
2858 007726 000000
2859 007730 000000 000000 000000
2860 007736 000000
2861 007740 047753 047744
2862 007744 000000
2863
2864
2865
2866
2867
2868
2869
2870 007746 000004
2871 007750 012705 007762
2872 007754 004737 034142
2873
2874 007760 000417
2875
2876 007762
2877 007762 000177 052525 052525
2878 007770 052525
2879 007772 000177 177777 177777
2880 010000 177777
2881 010002 000000 000000 000000
2882 010010 000000
2883 010012 047713 047704
2884 010016 000000
2885
2886
2887
2888
2889
2890
2891
2892 010020 000004
2893 010022 012705 010034
2894 010026 004737 034142
2895
2896 010032 000417
2897
2898 010034

```

```

*****
*TEST 115 TEST OF SUBD INSTR, DATA SET SUBD-1
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****
TST115: SCOPE
MOV #SUBD1,R5 ; PTR TO TEST DATA SET
JSR PC,@#SUBDT ; GO TEST
BR TST116 ;;

SUBD1: ; TEST DATA SET SUBD-1:
.WORD 0,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD 0,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 047753,047744 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
*TEST 116 TEST OF SUBD INSTR, DATA SET SUBD-2
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
TST116: SCOPE
MOV #SUBD2,R5 ; PTR TO TEST DATA SET
JSR PC,@#SUBDT ; GO TEST
BR TST117 ;;

SUBD2: ; TEST DATA SET SUBD-2:
.WORD ZXIMP,ALTP,ALTP,ALTP ; INITIAL AC FLOAT NUMBER
.WORD ZXIMP,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 047713,047704 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
*TEST 117 TEST OF SUBD INSTR, DATA SET SUBD-3
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST117: SCOPE
MOV #SUBD3,R5 ; PTR TO TEST DATA SET
JSR PC,@#SUBDT ; GO TEST
BR TST120 ;;

SUBD3: ; TEST DATA SET SUBD-3:

```

H05

FPU ADVANCED INSTR TESTS
DQFP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 61
T117 TEST OF SUBD INSTR, DATA SET SUBD-3

2899	010034	000000	000000	000000	.WORD	0,0,0,0 ; INITIAL AC FLOAT NUMBER
2900	010042	000000				
2901	010044	012345	177777	125252	.WORD	012345,M1,ALTN,MO ; INITIAL MEM FLOAT NUMBER
2902	010052	100000				
2903	010054	112345	177777	125252	.WORD	112345,M1,ALTN,MO ; EXPECTED FLOAT RESULT
2904	010062	100000				
2905	010064	047647	047650		.WORD	047647,047650 ; FPS: BEFORE, AFTER
2906	010070	000000			.WORD	NA ; FEC AFTER (0 = N/A)

```

*****
;TEST 120      TEST OF SUBD INSTR, DATA SET SUBD-4
;
;              ALL INTERRUPT ENABLES ON
;              LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

2910					TST120: SCOPE	
2911					MOV	#SUBD4,R5 ; PTR TO TEST DATA SET
2912					JSR	PC,@#SUBDT ; GO TEST
2913						
2914	010072	000004			BR	TST121 ;;
2915	010074	012705	010106			
2916	010100	004737	034142			
2917						
2918	010104	000417				
2919						
2920	010106				SUBD4: ; TEST DATA SET SUBD-4:	
2921	010106	140200	000000	000000	.WORD	F1N,0,0,0 ; INITIAL AC FLOAT NUMBER
2922	010114	000000				
2923	010116	140200	000000	000000	.WORD	F1N,0,0,0 ; INITIAL MEM FLOAT NUMBER
2924	010124	000000				
2925	010126	000000	000000	000000	.WORD	0,0,0,0 ; EXPECTED FLOAT RESULT
2926	010134	000000				
2927	010136	047753	047744		.WORD	047753,047744 ; FPS: BEFORE, AFTER
2928	010142	000000			.WORD	NA ; FEC AFTER (0 = N/A)

```

*****
;TEST 121      TEST OF SUBD INSTR, DATA SET SUBD-5
;
;              ALL INTERRUPT ENABLES ON
;              LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

2931					TST121: SCOPE	
2932					MOV	#SUBD5,R5 ; PTR TO TEST DATA SET
2933					JSR	PC,@#SUBDT ; GO TEST
2934						
2935						
2936	010144	000004			BR	TST122 ;;
2937	010146	012705	010160			
2938	010152	004737	034142			
2939						
2940	010156	000417				
2941						
2942	010160				SUBD5: ; TEST DATA SET SUBD-5:	
2943	010160	040200	000000	000000	.WORD	F1P,0,0,0 ; INITIAL AC FLOAT NUMBER
2944	010166	000000				
2945	010170	040200	000000	000000	.WORD	F1P,0,0,0 ; INITIAL MEM FLOAT NUMBER
2946	010176	000000				
2947	010200	000000	000000	000000	.WORD	0,0,0,0 ; EXPECTED FLOAT RESULT
2948	010206	000000				
2949	010210	047613	047604		.WORD	047613,047604 ; FPS: BEFORE, AFTER
2950	010214	000000			.WORD	NA ; FEC AFTER (0 = N/A)

```

*****
;TEST 122      TEST OF SUBD INSTR, DATA SET SUBD-6

```

2951
2952
2953
2954

```

2955
2956
2957
2958 010216 000004
2959 010220 012705 010232
2960 010224 004737 034142
2961
2962 010230 000417
2963
2964 010232
2965 010232 037252 125252 125252
2966 010240 125252
2967 010242 140365 052525 052525
2968 010250 052525
2969 010252 040377 177777 177777
2970 010260 177777
2971 010262 047757 047740
2972 010266 000000
2973
2974
2975
2976
2977
2978
2979
2980 010270 000004
2981 010272 012705 010304
2982 010276 004737 034142
2983
2984 010302 000417
2985
2986 010304
2987 010304 137252 125252 125252
2988 010312 125252
2989 010314 040365 052525 052525
2990 010322 052525
2991 010324 140400 000000 000000
2992 010332 000000
2993 010334 047607 047610
2994 010340 000000
2995
2996
2997
2998
2999
3000
3001
3002 010342 000004
3003 010344 012705 010356
3004 010350 004737 034142
3005
3006 010354 000417
3007
3008 010356
3009 010356 177777 177777 000000
3010 010364 177777

```

```

;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
;*****
↑TST122: SCOPE
MOV #SUBD6,R5 ; PTR TO TEST DATA SET
JSR PC,@#SUBDT ; GO TEST
BR TST123 ;;

SUBD6: ; TEST DATA SET SUBD-6:
.WORD 037252,ALTN,ALTN,ALTN ; INITIAL AC FLOAT NUMBER
.WORD 140365,ALTP,ALTP,ALTP ; INITIAL MEM FLOAT NUMBER
.WORD 040377,M1,M1,M1 ; EXPECTED FLOAT RESULT
.WORD 047757,047740 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

;*****
;TEST 123 TEST OF SUBD INSTR, DATA SET SUBD-7
;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, SHORT INTEGER, ROUND MODES
;*****
↑TST123: SCOPE
MOV #SUBD7,R5 ; PTR TO TEST DATA SET
JSR PC,@#SUBDT ; GO TEST
BR TST124 ;;

SUBD7: ; TEST DATA SET SUBD-7:
.WORD 137252,ALTN,ALTN,ALTN ; INITIAL AC FLOAT NUMBER
.WORD 040365,ALTP,ALTP,ALTP ; INITIAL MEM FLOAT NUMBER
.WORD 140400,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 047607,047610 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

;*****
;TEST 124 TEST OF SUBD INSTR, DATA SET SUBD-10
;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, SHORT INTEGER, ROUND MODES
;*****
↑TST124: SCOPE
MOV #SUBD10,R5 ; PTR TO TEST DATA SET
JSR PC,@#SUBDT ; GO TEST
BR TST125 ;;

SUBD10: ; TEST DATA SET SUBD-10:
.WORD LGN,M1,0,M1 ; INITIAL AC FLOAT NUMBER

```

J05

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 63
T124 TEST OF SUBD INSTR, DATA SET SUBD-10

3011	010366	100000	000000	177777	.WORD	MO,0,M1,0	; INITIAL MEM FLOAT NUMBER
3012	010374	000000					
3013	010376	177777	177777	000000	.WORD	LGN,M1,0,M1	; EXPECTED FLOAT RESULT
3014	010404	177777					
3015	010406	047603	147603		.WORD	047603,147603	; FPS: BEFORE, AFTER
3016	010412	100014			.WORD	100014	; FEC AFTER (0 = N/A)

3017
3018
3019

```

:*****
:TEST 125 TEST OF SUBD INSTR, DATA SET SUBD-11
:*
:* -0 INTERRUPT ENABLE OFF, ALL OTHERS ON
:* LONG FLOAT, LONG INTEGER, ROUND MODES
:*****

```

3020
3021
3022
3023

3024	010414	000004			TST125: SCOPE		
3025	010416	012705	010430		MOV	#SUBD11,R5	; PTR TO TEST DATA SET
3026	010422	004737	034142		JSR	PC,#SUBD11	; GO TEST
3027							
3028	010426	000417			BR	TST126	::

3029
3030
3031

```

SUBD11: ; TEST DATA SET SUBD-11:
:WORD LGN,M1,0,M1 ; INITIAL AC FLOAT NUMBER

```

3031	010430	177777	177777	000000	.WORD	LGN,M1,0,M1	; INITIAL AC FLOAT NUMBER
3032	010436	177777					
3033	010440	100000	000000	177777	.WORD	MO,0,M1,0	; INITIAL MEM FLOAT NUMBER
3034	010446	000000					
3035	010450	177777	177777	000000	.WORD	LGN,M1,0,M1	; EXPECTED FLOAT RESULT
3036	010456	177777					
3037	010460	043707	043710		.WORD	043707,043710	; FPS: BEFORE, AFTER
3038	010464	000000			.WORD	NA	; FEC AFTER (0 = N/A)

3039
3040
3041

```

:*****
:TEST 126 TEST OF SUBD INSTR, DATA SET SUBD-12
:*
:* ALL INTERRUPT ENABLES ON
:* LONG FLOAT, LONG INTEGER, ROUND MODES
:*****

```

3042
3043
3044
3045

3046	010466	000004			TST126: SCOPE		
3047	010470	012705	010502		MOV	#SUBD12,R5	; PTR TO TEST DATA SET
3048	010474	004737	034142		JSR	PC,#SUBD12	; GO TEST
3049							
3050	010500	000417			BR	TST127	::

3051
3052
3053

```

SUBD12: ; TEST DATA SET SUBD-12:
:WORD 104200,0,0,0 ; INITIAL AC FLOAT NUMBER

```

3053	010502	104200	000000	000000	.WORD	104200,0,0,0	; INITIAL AC FLOAT NUMBER
3054	010510	000000					
3055	010512	104200	000000	000000	.WORD	104200,0,0,1	; INITIAL MEM FLOAT NUMBER
3056	010520	000001					
3057	010522	066400	000000	000000	.WORD	066400,0,0,0	; EXPECTED FLOAT RESULT
3058	010530	000000					
3059	010532	047717	147700		.WORD	047717,147700	; FPS: BEFORE, AFTER
3060	010536	100012			.WORD	100012	; FEC AFTER (0 = N/A)

3061
3062
3063

```

:*****
:TEST 127 TEST OF SUBD INSTR, DATA SET SUBD-13
:*
:* UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
:* LONG FLOAT, LONG INTEGER, ROUND MODES

```

3064
3065
3066

```

3067
3068 010540 000004
3069 010542 012705 010554
3070 010546 004737 034142
3071
3072 010552 000417
3073
3074 010554
3075 010554 104200 000000 000000
3076 010562 000001
3077 010564 104200 000000 000000
3078 010572 000000
3079 010574 000000 000000 000000
3080 010602 000000
3081 010604 045713 045704
3082 010610 000000
3083
3084
3085
3086
3087
3088
3089
3090 010612 000004
3091 010614 012705 010626
3092 010620 004737 034142
3093
3094 010624 000417
3095
3096 010626
3097 010626 077777 177777 177777
3098 010634 177777
3099 010636 161600 000000 000000
3100 010644 000000
3101 010646 077777 177777 177777
3102 010654 177777
3103 010656 047757 047740
3104 010662 000000
3105
3106
3107
3108
3109
3110
3111
3112 010664 000004
3113 010666 012705 010700
3114 010672 004737 034142
3115
3116 010676 000417
3117
3118 010700
3119 010700 077777 177777 177777
3120 010706 177777
3121 010710 161600 000000 000000
3122 010716 000000

```

```

*****
↑ST127: SCOPE
MOV #SUBD13,R5 ; PTR TO TEST DATA SET
JSR PC,@SUBDT ; GO TEST
BR TST130 ;;
SUBD13: ; TEST DATA SET SUBD-13:
.WORD 104200,0,0,1 ; INITIAL AC FLOAT NUMBER
.WORD 104200,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 045713,045704 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
*****
↑TEST 130 TEST OF SUBD INSTR, DATA SET SUBD-14
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****
↑ST130: SCOPE
MOV #SUBD14,R5 ; PTR TO TEST DATA SET
JSR PC,@SUBDT ; GO TEST
BR TST131 ;;
SUBD14: ; TEST DATA SET SUBD-14:
.WORD LGP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD 161600,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD LGP,M1,M1,M1 ; EXPECTED FLOAT RESULT
.WORD 047757,047740 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
*****
↑TEST 131 TEST OF SUBD INSTR, DATA SET SUBD-15
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****
↑ST131: SCOPE
MOV #SUBD15,R5 ; PTR TO TEST DATA SET
JSR PC,@SUBDT ; GO TEST
BR TST132 ;;
SUBD15: ; TEST DATA SET SUBD-15:
.WORD LGP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD 161600,0,0,0 ; INITIAL MEM FLOAT NUMBER

```


FPU ADVANCED INSTR TESTS
D0FP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 65
T131 TEST OF SUBD INSTR, DATA SET SUBD-15

3123	010720	000000	000000	000000	.WORD	0,0,0,0	; EXPECTED FLOAT RESULT
3124	010726	000000					
3125	010730	047611	147606		.WORD	047611,147606	; FPS: BEFORE, AFTER
3126	010734	100010			.WORD	100010	; FEC AFTER (0 = N/A)

3127
3128
3129
3130
3131
3132
3133
3134
3135
3136
3137
3138
3139
3140
3141
3142
3143
3144
3145
3146
3147
3148
3149
3150

```

*****
*TEST 132      TEST OF SUBD INSTR, DATA SET SUBD-16
*              OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
*              LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

TST132: SCOPE
        MOV     #SUBD16,R5      ; PTR TO TEST DATA SET
        JSR    PC,@#SUBDT     ; GO TEST
        BR     TST133         ;;

```

```

SUBD16: ; TEST DATA SET SUBD-16:
        .WORD  LGP,M1,M1,M1    ; INITIAL AC FLOAT NUMBER
        .WORD  161600,0,0,0   ; INITIAL MEM FLOAT NUMBER
        .WORD  0,0,0,0        ; EXPECTED FLOAT RESULT
        .WORD  046611,046606  ; FPS: BEFORE, AFTER
        .WORD  NA              ; FEC AFTER ( 0 = N/A )

```

3134	010736	000004		
3135	010740	012705	010752	
3136	010744	004737	034142	
3138	010750	000417		
3140	010752			
3141	010752	077777	177777	177777
3142	010760	177777		
3143	010762	161600	000000	000000
3144	010770	000000		
3145	010772	000000	000000	000000
3146	011000	000000		
3147	011002	046611	046606	
3148	011006	000000		

M05

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 66
T133 TEST OF MULF INSTR, DATA SET MULF-1

3151			
3152			
3153			
3154			
3155			
3156	011010	000004	
3157	011012	012705	011024
3158	011016	004737	034332
3159			
3160	011022	000411	
3161			
3162	011024		
3163	011024	000000	000000
3164	011030	177777	177777
3165	011034	000000	000000
3166	011040	047413	047404
3167	011044	000000	
3168			
3169			
3170			
3171			
3172			
3173			
3174			
3175	011046	000004	
3176	011050	012705	011062
3177	011054	004737	034332
3178			
3179	011060	000411	
3180			
3181	011062		
3182	011062	077777	177777
3183	011066	000177	177777
3184	011072	000000	000000
3185	011076	047503	047504
3186	011102	000000	
3187			
3188			
3189			
3190			
3191			
3192			
3193			
3194	011104	000004	
3195	011106	012705	011120
3196	011112	004737	034332
3197			
3198	011116	000411	
3199			
3200	011120		
3201	011120	123652	125252
3202	011124	017500	000000
3203	011.30	103177	177777
3204	011134	047447	047450
3205	011140	000000	
3206			

```

*****
*TEST 133 TEST OF MULF INSTR, DATA SET MULF-1
*
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

†T133: SCOPE
MOV #MULF1,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULFT ; GO TEST
BR TST134 ;;

```

```

MULF1: ; TEST DATA SET MULF-1:
.WORD 0,0 ; INITIAL AC FLOAT NUMBER
.WORD LGN,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD 047413,047404 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 134 TEST OF MULF INSTR, DATA SET MULF-2
*
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

†T134: SCOPE
MOV #MULF2,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULFT ; GO TEST
BR TST135 ;;

```

```

MULF2: ; TEST DATA SET MULF-2:
.WORD LGP,M1 ; INITIAL AC FLOAT NUMBER
.WORD ZXIMP,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD 047503,047504 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 135 TEST OF MULF INSTR, DATA SET MULF-3
*
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

†T135: SCOPE
MOV #MULF3,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULFT ; GO TEST
BR TST136 ;;

```

```

MULF3: ; TEST DATA SET MULF-3:
.WORD 123652,ALTN ; INITIAL AC FLOAT NUMBER
.WORD 017500,000000 ; INITIAL MEM FLOAT NUMBER
.WORD 103177,M1 ; EXPECTED FLOAT RESULT
.WORD 047447,047450 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

N05

FPU ADVANCED INSTR TESTS
 DAFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 67
 T135 TEST OF MULF INSTR, DATA SET MULF-3

```

3207
3208
3209
3210
3211
3212
3213 011142 000004
3214 011144 012705 011156
3215 011150 004737 034332
3216
3217 011154 000411
3218
3219 011156
3220 011156 017500 000000
3221 011162 023652 125252
3222 011166 003177 177777
3223 011172 047417 047400
3224 011176 000000
3225
3226
3227
3228
3229
3230
3231
3232 011200 000004
3233 011202 012705 011214
3234 011206 004737 034332
3235
3236 011212 000411
3237
3238 011214
3239 011214 036400 000001
3240 011220 106777 177776
3241 011224 105177 177777
3242 011230 047547 047550
3243 011234 000000
3244
3245
3246
3247
3248
3249
3250
3251 011236 000004
3252 011240 012705 011252
3253 011244 004737 034332
3254
3255 011250 000411
3256
3257 011252
3258 011252 036400 000001
3259 011256 106777 177776
3260 011262 105200 000000
3261 011266 047407 047410
3262 011272 000000
  
```

```

*****
;TEST 136 TEST OF MULF INSTR, DATA SET MULF-4
;*
;* ALL INTERRUPT ENABLES ON
;* SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****
TST136: SCOPE
MOV #MULF4,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULFT ; GO TEST
BR TST137 ;;
  
```

```

MULF4: ; TEST DATA SET MULF-4:
.WORD 017500,000000 ; INITIAL AC FLOAT NUMBER
.WORD 023652,ALTN ; INITIAL MEM FLOAT NUMBER
.WORD 003177,M1 ; EXPECTED FLOAT RESULT
.WORD 047417,047400 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
  
```

```

*****
;TEST 137 TEST OF MULF INSTR, DATA SET MULF-5
;*
;* ALL INTERRUPT ENABLES ON
;* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****
TST137: SCOPE
MOV #MULF5,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULFT ; GO TEST
BR TST140 ;;
  
```

```

MULF5: ; TEST DATA SET MULF-5:
.WORD 036400,000001 ; INITIAL AC FLOAT NUMBER
.WORD 106777,M2 ; INITIAL MEM FLOAT NUMBER
.WORD 105177,M1 ; EXPECTED FLOAT RESULT
.WORD 047547,047550 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
  
```

```

*****
;TEST 140 TEST OF MULF INSTR, DATA SET MULF-6
;*
;* ALL INTERRUPT ENABLES ON
;* SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****
TST140: SCOPE
MOV #MULF6,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULFT ; GO TEST
BR TST141 ;;
  
```

```

MULF6: ; TEST DATA SET MULF-6:
.WORD 036400,000001 ; INITIAL AC FLOAT NUMBER
.WORD 106777,M2 ; INITIAL MEM FLOAT NUMBER
.WORD 105200,000000 ; EXPECTED FLOAT RESULT
.WORD 047407,047410 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
  
```

```

3263
3264
3265
3266
3267
3268
3269
3270 011274 000004
3271 011276 012705 011310
3272 011302 004737 034332
3273
3274 011306 000411
3275
3276 011310
3277 011310 140277 000000
3278 011314 060000 000001
3279 011320 160077 000001
3280 011324 047407 047410
3281 011330 000000
3282
3283
3284
3285
3286
3287
3288
3289 011332 000004
3290 011334 012705 011346
3291 011340 004737 034332
3292
3293 011344 000411
3294
3295 011346
3296 011346 060000 000001
3297 011352 040277 000000
3298 011356 060077 000001
3299 011362 047457 047440
3300 011366 000000
3301
3302
3303
3304
3305
3306
3307
3308 011370 000004
3309 011372 012705 011404
3310 011376 004737 034332
3311
3312 011402 000411
3313
3314 011404
3315 011404 140300 000000
3316 011410 160000 000001
3317 011414 060100 000002
3318 011420 047517 047500

```

```

*****
: TEST 141 TEST OF MULF INSTR, DATA SET MULF-7
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

TST141: SCOPE
MOV #MULF7,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULFT ; GO TEST
BR TST142 ;;

```

```

MULF7: ; TEST DATA SET MULF-7:
.WORD 140277,000000 ; INITIAL AC FLOAT NUMBER
.WORD 060000,000001 ; INITIAL MEM FLOAT NUMBER
.WORD 160077,000001 ; EXPECTED FLOAT RESULT
.WORD 047407,047410 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
: TEST 142 TEST OF MULF INSTR, DATA SET MULF-10
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

TST142: SCOPE
MOV #MULF10,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULFT ; GO TEST
BR TST143 ;;

```

```

MULF10: ; TEST DATA SET MULF-10:
.WORD 060000,000001 ; INITIAL AC FLOAT NUMBER
.WORD 040277,000000 ; INITIAL MEM FLOAT NUMBER
.WORD 060077,000001 ; EXPECTED FLOAT RESULT
.WORD 047457,047440 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
: TEST 143 TEST OF MULF INSTR, DATA SET MULF-11
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

TST143: SCOPE
MOV #MULF11,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULFT ; GO TEST
BR TST144 ;;

```

```

MULF11: ; TEST DATA SET MULF-11:
.WORD 140300,000000 ; INITIAL AC FLOAT NUMBER
.WORD 160000,000001 ; INITIAL MEM FLOAT NUMBER
.WORD 060100,000002 ; EXPECTED FLOAT RESULT
.WORD 047517,047500 ; FPS: BEFORE, AFTER

```

```

3319 011424 000000
3320
3321
3322
3323
3324
3325
3326
3327 011426 000004
3328 011430 012705 011442
3329 011434 004737 034332
3330
3331 011440 000411
3332
3333 011442
3334 011442 060000 000001
3335 011446 140300 000000
3336 011452 160100 000001
3337 011456 047547 047550
3338 011462 000000
3339
3340
3341
3342
3343
3344
3345
3346 011464 000004
3347 011466 012705 011500
3348 011472 004737 034332
3349
3350 011476 000411
3351
3352 011500
3353 011500 002177 177777
3354 011504 002177 177777
3355 011510 044177 177776
3356 011514 047513 147500
3357 011520 100012
3358
3359
3360
3361
3362
3363
3364
3365 011522 000004
3366 011524 012705 011536
3367 011530 004737 034332
3368
3369 011534 000411
3370
3371 011536
3372 011536 170000 000000
3373 011542 050200 000000
3374 011546 100000 000000

```

```

.WORD NA ; FEC AFTER ( 0 = N/A )

:*****
:TEST 144 TEST OF MULF INSTR, DATA SET MULF-12
:* ALL INTERRUPT ENABLES ON
:* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
:*****
TST144: SCOPE
MOV #MULF12,RS ; PTR TO TEST DATA SET
JSR PC,@#MULFT ; GO TEST

BR TST145 ;;

MULF12: ; TEST DATA SET MULF-12:
.WORD 060000,000001 ; INITIAL AC FLOAT NUMBER
.WORD 140300,000000 ; INITIAL MEM FLOAT NUMBER
.WORD 160100,000001 ; EXPECTED FLOAT RESULT
.WORD 047547,047550 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

:*****
:TEST 145 TEST OF MULF INSTR, DATA SET MULF-13
:* ALL INTERRUPT ENABLES ON
:* SHORT FLOAT, LONG INTEGER, ROUND MODES
:*****
TST145: SCOPE
MOV #MULF13,RS ; PTR TO TEST DATA SET
JSR PC,@#MULFT ; GO TEST

BR TST146 ;;

MULF13: ; TEST DATA SET MULF-13:
.WORD 002177,M1 ; INITIAL AC FLOAT NUMBER
.WORD 002177,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 044177,M2 ; EXPECTED FLOAT RESULT
.WORD 047513,147500 ; FPS: BEFORE, AFTER
.WORD 100012 ; FEC AFTER ( 0 = N/A )

:*****
:TEST 146 TEST OF MULF INSTR, DATA SET MULF-14
:* ALL INTERRUPT ENABLES ON
:* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
:*****
TST146: SCOPE
MOV #MULF14,RS ; PTR TO TEST DATA SET
JSR PC,@#MULFT ; GO TEST

BR TST147 ;;

MULF14: ; TEST DATA SET MULF-14:
.WORD 170000,000000 ; INITIAL AC FLOAT NUMBER
.WORD 050200,000000 ; INITIAL MEM FLOAT NUMBER
.WORD M0,0 ; EXPECTED FLOAT RESULT

```

3375 011552 047441 147456
3376 011556 100010

.WORD 047441,147456 ; FPS: BEFORE, AFTER
.WORD 100010 ; FEC AFTER (0 = N/A)

3377
3378
3379
3380
3381
3382
3383

*TEST 147 TEST OF MULF INSTR, DATA SET MULF-15
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES

3384 011560 000004
3385 011562 012705 011574
3386 011566 004737 034332

†ST147: SCOPE
MOV #MULF15,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULFT ; GO TEST

3387
3388 011572 000411

BR TST150 ; ;

3389
3390 011574
3391 011574 177777 177777
3392 011600 100177 177777
3393 011604 177777 177777
3394 011610 047447 147447
3395 011614 100014

MULF15: ; TEST DATA SET MULF-15:
.WORD LGN,M1 ; INITIAL AC FLOAT NUMBER
.WORD ZX1FN,M1 ; INITIAL MEM FLOAT NUMBER
.WORD LGN,M1 ; EXPECTED FLOAT RESULT
.WORD 047447,147447 ; FPS: BEFORE, AFTER
.WORD 100014 ; FEC AFTER (0 = N/A)

3396
3397
3398
3399

*TEST 150 TEST OF MULF INSTR, DATA SET MULF-16
* OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
* SHORT FLOAT, SHORT INTEGER, ROUND MODES

3400
3401
3402

†ST150: SCOPE
MOV #MULF16,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULFT ; GO TEST

3403 011616 000004
3404 011620 012705 011632
3405 011624 004737 034332

BR TST151 ; ;

3406
3407 011630 000411

MULF16: ; TEST DATA SET MULF-16:
.WORD 050377,000000 ; INITIAL AC FLOAT NUMBER
.WORD 070000,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD 046411,046406 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

3408
3409 011632
3410 011632 050377 000000
3411 011636 070000 177777
3412 011642 000000 000000
3413 011646 046411 046406
3414 011652 000000

3415
3416
3417

*TEST 151 TEST OF MULF INSTR, DATA SET MULF-17
* UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES

3418
3419
3420
3421

†ST151: SCOPE
MOV #MULF17,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULFT ; GO TEST

3422 011654 000004
3423 011656 012705 011670
3424 011662 004737 034332

BR TST152 ; ;

3425
3426 011666 000411

MULF17: ; TEST DATA SET MULF-17:
.WORD 002177,M1 ; INITIAL AC FLOAT NUMBER
.WORD 002177,M1 ; INITIAL MEM FLOAT NUMBER

3427
3428 011670
3429 011670 002177 177777
3430 011674 002177 177777

FPU ADVANCED INSTR TESTS
DQFP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 71
T151 TEST OF MULF INSTR, DATA SET MULF-17

3431	011700	000000	000000
3432	011704	045553	045544
3433	011710	000000	
3434			
3435			
3436			
3437			
3438			
3439			
3440			
3441	011712	000004	
3442	011714	012705	011726
3443	011720	004737	034332
3444			
3445	011724	000411	
3446			
3447	011726		
3448	011726	052525	052525
3449	011732	100000	177777
3450	011736	000000	000000
3451	011742	043513	043504
3452	011746	000000	
3453			
3454			
3455			

```
.WORD 0.0 ; EXPECTED FLOAT RESULT
.WORD 045553,045544 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
```

```
*****
*TEST 152 TEST OF MULF INSTR, DATA SET MULF-20
* -0 INTERRUPT ENABLE OFF, ALL OTHERS ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****
```

```
TST152: SCOPE
MOV #MULF20,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULFT ; GO TEST
BR TST153 ;;
```

```
MULF20: ; TEST DATA SET MULF-20:
.WORD ALTP,ALTP ; INITIAL AC FLOAT NUMBER
.WORD MO,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0.0 ; EXPECTED FLOAT RESULT
.WORD 043513,043504 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
```

F06

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 72
T153 TEST OF MULD INSTR, DATA SET MULD-1

3456
3457
3458
3459
3460
3461 011750 000004
3462 011752 012705 011764
3463 011756 004737 034502
3464
3465 011762 000417
3466
3467 011764
3468 011764 077777 177777 177777
3469 011772 177777
3470 011774 000000 000000 000000
3471 012002 000000
3472 012004 000000 000000 000000
3473 012012 000000
3474 012014 047713 047704
3475 012020 000000
3476
3477
3478
3479
3480
3481
3482
3483 012022 000004
3484 012024 012705 012036
3485 012030 004737 034502
3486
3487 012034 000417
3488
3489 012036
3490 012036 000177 177777 177777
3491 012044 177777
3492 012046 177777 177777 177777
3493 012054 177777
3494 012056 000000 000000 000000
3495 012064 000000
3496 012066 047603 047604
3497 012072 000000
3498
3499
3500
3501
3502
3503
3504
3505 012074 000004
3506 012076 012705 012110
3507 012102 004737 034502
3508
3509 012106 000417
3510
3511 012110

```

*****
*TEST 153      TEST OF MULD INSTR, DATA SET MULD-1
*              ALL INTERRUPT ENABLES ON
*              LONG FLOAT, LONG INTEGER, ROUND MODES
*****
↑ST153: SCOPE
MOV      #MULD1,R5      ; PTR TO TEST DATA SET
JSR      PC,@#MULDT    ; GO TEST

BR       TST154        ;;

MULD1:  ; TEST DATA SET MULD-1:
.WORD   LGP,M1,M1,M1  ; INITIAL AC FLOAT NUMBER
.WORD   0,0,0,0      ; INITIAL MEM FLOAT NUMBER
.WORD   0,0,0,0      ; EXPECTED FLOAT RESULT
.WORD   047713,047704 ; FPS: BEFORE, AFTER
.WORD   NA            ; FEC AFTER ( 0 = N/A )

*****
*TEST 154      TEST OF MULD INSTR, DATA SET MULD-2
*              ALL INTERRUPT ENABLES ON
*              LONG FLOAT, SHORT INTEGER, ROUND MODES
*****
↑ST154: SCOPE
MOV      #MULD2,R5      ; PTR TO TEST DATA SET
JSR      PC,@#MULDT    ; GO TEST

BR       TST155        ;;

MULD2:  ; TEST DATA SET MULD-2:
.WORD   ZXIMP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD   LGN,M1,M1,M1  ; INITIAL MEM FLOAT NUMBER
.WORD   0,0,0,0      ; EXPECTED FLOAT RESULT
.WORD   047603,047604 ; FPS: BEFORE, AFTER
.WORD   NA            ; FEC AFTER ( 0 = N/A )

*****
*TEST 155      TEST OF MULD INSTR, DATA SET MULD-3
*              ALL INTERRUPT ENABLES ON
*              LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****
↑ST155: SCOPE
MOV      #MULD3,R5      ; PTR TO TEST DATA SET
JSR      PC,@#MULDT    ; GO TEST

BR       TST156        ;;

MULD3:  ; TEST DATA SET MULD-3:

```


FPU ADVANCED INSTR TESTS
D0FPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 73
T155 TEST OF MULD INSTR, DATA SET MULD-3

3512	012110	023652	125252	125252	.WORD	023652,ALTN,ALTN,ALTN	; INITIAL AC FLOAT NUMBER
3513	012116	125252					
3514	012120	017500	000000	000000	.WORD	017500,0,0,0	; INITIAL MEM FLOAT NUMBER
3515	012126	000000					
3516	012130	003177	177777	177777	.WORD	003177,M1,M1,M1	; EXPECTED FLOAT RESULT
3517	012136	177777					
3518	012140	047757	047740		.WORD	047757,047740	; FPS: BEFORE, AFTER
3519	012144	000000			.WORD	NA	; FEC AFTER (0 = N/A)

3520							
3521							
3522							

```

*****
; *TEST 156      TEST OF MULD INSTR, DATA SET MULD-4
; *              ALL INTERRUPT ENABLES ON
; *              LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

3527	012146	000004			↑TST156: SCOPE		
3528	012150	012705	012162		MOV	#MULD4,R5	; PTR TO TEST DATA SET
3529	012154	004737	034502		JSR	PC,2#MULDT	; GO TEST
3530							
3531	012160	000417			BR	TST157	::

3532							
3533	012162				MULD4: ; TEST DATA SET MULD-4:		
3534	012162	117500	000000	000000	.WORD	117500,0,0,0	; INITIAL AC FLOAT NUMBER
3535	012170	000000					
3536	012172	123652	125252	125252	.WORD	123652,ALTN,ALTN,ALTN	; INITIAL MEM FLOAT NUMBER
3537	012200	125252					
3538	012202	003177	177777	177777	.WORD	003177,M1,M1,M1	; EXPECTED FLOAT RESULT
3539	012210	177777					
3540	012212	047617	047600		.WORD	047617,047600	; FPS: BEFORE, AFTER
3541	012216	000000			.WORD	NA	; FEC AFTER (0 = N/A)

3542							
3543							
3544							

```

*****
; *TEST 157      TEST OF MULD INSTR, DATA SET MULD-5
; *              ALL INTERRUPT ENABLES ON
; *              LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

3549	012220	000004			↑TST157: SCOPE		
3550	012222	012705	012234		MOV	#MULD5,R5	; PTR TO TEST DATA SET
3551	012226	004737	034502		JSR	PC,2#MULDT	; GO TEST
3552							
3553	012232	000417			BR	TST160	::

3554							
3555	012234				MULD5: ; TEST DATA SET MULD-5:		
3556	012234	165400	000000	000000	.WORD	165400,0,0,1	; INITIAL AC FLOAT NUMBER
3557	012242	000001					
3558	012244	037577	177777	177777	.WORD	037577,M1,M1,M2	; INITIAL MEM FLOAT NUMBER
3559	012252	177776					
3560	012254	164777	177777	177777	.WORD	164777,M1,M1,M1	; EXPECTED FLOAT RESULT
3561	012262	177777					
3562	012264	047747	047750		.WORD	047747,047750	; FPS: BEFORE, AFTER
3563	012270	000000			.WORD	NA	; FEC AFTER (0 = N/A)

3564							
3565							
3566							

```

*****
; *TEST 160      TEST OF MULD INSTR, DATA SET MULD-6

```

3567							
------	--	--	--	--	--	--	--

H06

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 74
T160 TEST OF MULD INSTR, DATA SET MULD-6

```

3568
3569
3570
3571 012272 000004
3572 012274 012705 012306
3573 012300 004737 034502
3574
3575 012304 000417
3576
3577 012306
3578 012306 165400 000000 000000
3579 012314 000001
3580 012316 037577 177777 177777
3581 012324 177776
3582 012326 165000 000000 000000
3583 012334 000000
3584 012336 047707 047710
3585 012342 000000
3586
3587
3588
3589
3590
3591
3592
3593 012344 000004
3594 012346 012705 012360
3595 012352 004737 034502
3596
3597 012356 000417
3598
3599 012360
3600 012360 040277 000000 000000
3601 012366 000000
3602 012370 034200 000000 000000
3603 012376 000001
3604 012400 034277 000000 000000
3605 012406 000001
3606 012410 047657 047640
3607 012414 000000
3608
3609
3610
3611
3612
3613
3614
3615 012416 000004
3616 012420 012705 012432
3617 012424 004737 034502
3618
3619 012430 000417
3620
3621 012432
3622 012432 140277 000000 000000
3623 012440 000000

```

```

;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, LONG INTEGER, ROUND MODES
;*****
↑ST160: SCOPE
MOV #MULD6,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULDT ; GO TEST
BR TST161 ;;

MULD6: ; TEST DATA SET MULD-6:
.WORD 165400,0,0,1 ; INITIAL AC FLOAT NUMBER
.WORD 037577,M1,M1,M2 ; INITIAL MEM FLOAT NUMBER
.WORD 165000,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 047707,047710 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

;*****
; TEST 161 TEST OF MULD INSTR, DATA SET MULD-7
;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
;*****
↑ST161: SCOPE
MOV #MULD7,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULDT ; GO TEST
BR TST162 ;;

MULD7: ; TEST DATA SET MULD-7:
.WORD 040277,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD 034200,0,0,1 ; INITIAL MEM FLOAT NUMBER
.WORD 034277,0,0,1 ; EXPECTED FLOAT RESULT
.WORD 047657,047640 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

;*****
; TEST 162 TEST OF MULD INSTR, DATA SET MULD-10
;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, SHORT INTEGER, ROUND MODES
;*****
↑ST162: SCOPE
MOV #MULD10,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULDT ; GO TEST
BR TST163 ;;

MULD10: ; TEST DATA SET MULD-10:
.WORD 140277,0,0,0 ; INITIAL AC FLOAT NUMBER

```

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 75
T162 TEST OF MULD INSTR, DATA SET MULD-10

3624	012442	034200	000000	000000	.WORD	034200,0,0,1	; INITIAL MEM FLOAT NUMBER
3625	012450	000001					
3626	012452	134277	000000	000000	.WORD	134277,0,0,1	; EXPECTED FLOAT RESULT
3627	012460	000001					
3628	012462	047607	047610		.WORD	047607,047610	; FPS: BEFORE AFTER
3629	012466	000000			.WORD	NA	; FEC AFTER (0 = N/A)

```

*****
*TEST 163      TEST OF MULD INSTR, DATA SET MULD-11
*
*              ALL INTERRUPT ENABLES ON
*              LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

3637	012470	000004			TST163: SCOPE		
3638	012472	012705	012504		MOV	#MULD11,R5	; PTR TO TEST DATA SET
3639	012476	004737	034502		JSR	PC,#MULDT	; GO TEST
3640							
3641	012502	000417			BR	TST164	::

```

MULD11: ; TEST DATA SET MULD-11:
.WORD 040300,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD 134200,0,0,1 ; INITIAL MEM FLOAT NUMBER
.WORD 134300,0,0,1 ; EXPECTED FLOAT RESULT
.WORD 047747,047750 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 164      TEST OF MULD INSTR, DATA SET MULD-12
*
*              ALL INTERRUPT ENABLES ON
*              LONG FLOAT, LONG INTEGER, ROUND MODES
*****

```

3659	012542	000004			TST164: SCOPE		
3660	012544	012705	012556		MOV	#MULD12,R5	; PTR TO TEST DATA SET
3661	012550	004737	034502		JSR	PC,#MULDT	; GO TEST
3662							
3663	012554	000417			BR	TST165	::

```

MULD12: ; TEST DATA SET MULD-12:
.WORD 140300,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD 134200,0,0,1 ; INITIAL MEM FLOAT NUMBER
.WORD 034300,0,0,2 ; EXPECTED FLOAT RESULT
.WORD 047717,047700 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 165      TEST OF MULD INSTR, DATA SET MULD-13
*
*              ALL INTERRUPT ENABLES ON
*              LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

3679

J06

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 76
T165 TEST OF MULD INSTR, DATA SET MULD-13

```

3680
3681 012614 000004
3682 012616 012705 012630
3683 012622 004737 034502
3684
3685 012626 000417
3686
3687 012630
3688 012630 177777 177777 177777
3689 012636 177777
3690 012640 177777 177777 177777
3691 012646 177777
3692 012650 037577 177777 177777
3693 012656 177776
3694 012660 047655 147642
3695 012664 100010
3696
3697
3698
3699
3700
3701
3702
3703 012666 000004
3704 012670 012705 012702
3705 012674 004737 034502
3706
3707 012700 000417
3708
3709 012702
3710 012702 077777 177777 177777
3711 012710 177777
3712 012712 077777 177777 177777
3713 012720 177777
3714 012722 000000 000000 000000
3715 012730 000000
3716 012732 046751 046746
3717 012736 000000
3718
3719
3720
3721
3722
3723
3724
3725 012740 000004
3726 012742 012705 012754
3727 012746 004737 034502
3728
3729 012752 000417
3730
3731 012754
3732 012754 003177 177777 177777
3733 012762 177777
3734 012764 101177 177777 177777
3735 012772 177777

```

```

*****
TST165: SCOPE
MOV #MULD13,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULDT ; GO TEST
BR TST166 ;;

MULD13: ; TEST DATA SET MULD-13:
.WORD LGN,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD LGN,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 037577,M1,M1,M2 ; EXPECTED FLOAT RESULT
.WORD 047655,147642 ; FPS: BEFORE, AFTER
.WORD 100010 ; FEC AFTER ( 0 = N/A )

*****
*TEST 166 TEST OF MULD INSTR, DATA SET MULD-14
* OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****
TST166: SCOPE
MOV #MULD14,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULDT ; GO TEST
BR TST167 ;;

MULD14: ; TEST DATA SET MULD-14:
.WORD LGP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD LGP,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 046751,046746 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
*TEST 167 TEST OF MULD INSTR, DATA SET MULD-15
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST167: SCOPE
MOV #MULD15,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULDT ; GO TEST
BR TST170 ;;

MULD15: ; TEST DATA SET MULD-15:
.WORD 003177,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD 101177,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER

```

K06

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 77
T167 TEST OF MULD INSTR, DATA SET MULD-15

3736 012774 144177 177777 177777 .WORD 144177,M1,M1,M2 ; EXPECTED FLOAT RESULT
3737 013002 177776
3738 013004 047647 147650 .WORD 047647,147650 ; FPS: BEFORE AFTER
3739 013010 100012 .WORD 100012 ; FEC AFTER (0 = N/A)

3740
3741
3742
3743
3744
3745
3746
3747 013012 000004
3748 013014 012705 013026
3749 013020 004737 034502
3750
3751 013024 000417
3752
3753 013026
3754 013026 103177 177777 177777
3755 013034 177777
3756 013036 001177 177777 177777
3757 013044 177777
3758 013046 000000 000000 000000
3759 013054 000000
3760 013056 045713 045704 .WORD 045713,045704 ; FPS: BEFORE AFTER
3761 013062 000000 .WORD NA ; FEC AFTER (0 = N/A)

```
*****
*TEST 170 TEST OF MULD INSTR, DATA SET MULD-16
* UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
```

```
TST170: SCOPE
MOV #MULD16,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULDT ; GO TEST
BR TST171 ;;
```

```
MULD16: ; TEST DATA SET MULD-16:
.WORD 103177,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD 001177,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 045713,045704 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
```

3762
3763
3764
3765
3766
3767
3768
3769 013064 000004
3770 013066 012705 013100
3771 013072 004737 034502
3772
3773 013076 000417
3774
3775 013100
3776 013100 052525 052525 052525
3777 013106 052525
3778 013110 100177 177777 177777
3779 013116 177777
3780 013120 052525 052525 052525
3781 013126 052525
3782 013130 047657 147657 .WORD 047657,147657 ; FPS: BEFORE AFTER
3783 013134 100014 .WORD 100014 ; FEC AFTER (0 = N/A)

```
*****
*TEST 171 TEST OF MULD INSTR, DATA SET MULD-17
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
```

```
TST171: SCOPE
MOV #MULD17,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULDT ; GO TEST
BR TST172 ;;
```

```
MULD17: ; TEST DATA SET MULD-17:
.WORD ALTP,ALTP,ALTP,ALTP ; INITIAL AC FLOAT NUMBER
.WORD ZX1MN,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD ALTP,ALTP,ALTP,ALTP ; EXPECTED FLOAT RESULT
.WORD 047657,147657 ; FPS: BEFORE AFTER
.WORD 100014 ; FEC AFTER ( 0 = N/A )
```

3784
3785
3786
3787
3788
3789
3790
3791 013136 000004

```
*****
*TEST 172 TEST OF MULD INSTR, DATA SET MULD-20
* -0 INTERRUPT ENABLE OFF, ALL OTHERS ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****
```

```
TST172: SCOPE
```

FPU ADVANCED INSTR TESTS
D0FPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 78
T172 TEST OF MULD INSTR, DATA SET MULD-20

3792	013140	012705	013152
3793	013144	004737	034502
3794			
3795	013150	000417	
3796			
3797	013152		
3798	013152	125252	125252 125252
3799	013160	125252	
3800	013162	100000	177777 052525
3801	013170	125252	
3802	013172	000000	000000 000000
3803	013200	000000	
3804	013202	043753	043744
3805	013206	000000	
3806			
3807			

```

MOV #MULD20,R5 ; PTR TO TEST DATA SET
JSR PC,@MULDT ; GO TEST
BR TS^173 ;;

```

```

MULD20: ; TEST DATA SET MULD-20:
.WORD ALTN,ALTN,ALTN,ALTN ; INITIAL AC FLOAT NUMBER
.WORD M0,M1,ALTP,ALTN ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 043753,043744 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

M06

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 79
T173 TEST OF DIVF INSTR, DATA SET DIVF-1

```

3808
3809
3810
3811
3812
3813 013210 000004
3814 013212 012705 013224
3815 013216 004737 034672
3816
3817 013222 000411
3818
3819 013224
3820 013224 103177 177777
3821 013230 023652 125252
3822 013234 117500 000000
3823 013240 047447 047450
3824 013244 000000
3825
3826
3827
3828
3829
3830
3831
3832 013246 000004
3833 013250 012705 013262
3834 013254 004737 034672
3835
3836 013260 000411
3837
3838 013262
3839 013262 052525 052525
3840 013266 000000 000000
3841 013272 052525 052525
3842 013276 047517 147517
3843 013302 100004
3844
3845
3846
3847
3848
3849
3850
3851 013304 000004
3852 013306 012705 013320
3853 013312 004737 034672
3854
3855 013316 000411
3856
3857 013320
3858 013320 140400 000000
3859 013324 040500 000000
3860 013330 140052 125252
3861 013334 047447 047450
3862 013340 000000
3863

```

```

*****
*TEST 173 TEST OF DIVF INSTR, DATA SET DIVF-1
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

TST173: SCOPE
MOV #DIVF1,R5 ; PTR TO TEST DATA SET
JSR PC,#DIVFT ; GO TEST
BR TST174 ;;

```

```

DIVF1: ; TEST DATA SET DIVF-1:
.WORD 103177,M1 ; INITIAL AC FLOAT NUMBER
.WORD 023652,ALTN ; INITIAL MEM FLOAT NUMBER
.WORD 117500,0 ; EXPECTED FLOAT RESULT
.WORD 047447,047450 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 174 TEST OF DIVF INSTR, DATA SET DIVF-2
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

TST174: SCOPE
MOV #DIVF2,R5 ; PTR TO TEST DATA SET
JSR PC,#DIVFT ; GO TEST
BR TST175 ;;

```

```

DIVF2: ; TEST DATA SET DIVF-2:
.WORD ALTP,ALTP ; INITIAL AC FLOAT NUMBER
.WORD 0,0 ; INITIAL MEM FLOAT NUMBER
.WORD ALTP,ALTP ; EXPECTED FLOAT RESULT
.WORD 047517,147517 ; FPS: BEFORE, AFTER
.WORD 100004 ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 175 TEST OF DIVF INSTR, DATA SET DIVF-3
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

TST175: SCOPE
MOV #DIVF3,R5 ; PTR TO TEST DATA SET
JSR PC,#DIVFT ; GO TEST
BR TST176 ;;

```

```

DIVF3: ; TEST DATA SET DIVF-3:
.WORD 140400,000000 ; INITIAL AC FLOAT NUMBER
.WORD 040500,000000 ; INITIAL MEM FLOAT NUMBER
.WORD 140052,ALTN ; EXPECTED FLOAT RESULT
.WORD 047447,047450 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

N06

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 80
T175 TEST OF DIVF INSTR, DATA SET DIVF-3

```

3864
3865
3866
3867
3868
3869
3870 013342 000004
3871 013344 012705 013356
3872 013350 004737 034672
3873
3874 013354 000411
3875
3876 013356
3877 013356 040400 000000
3878 013362 140500 000000
3879 013366 140052 125253
3880 013372 047507 047510
3881 013376 000000
3882
3883
3884
3885
3886
3887
3888
3889 013400 000004
3890 013402 012705 013414
3891 013406 004737 034672
3892
3893 013412 000411
3894
3895 013414
3896 013414 007417 007417
3897 013420 007417 007417
3898 013424 040200 000000
3899 013430 047417 047400
3900 013434 000000
3901
3902
3903
3904
3905
3906
3907
3908 013436 000004
3909 013440 012705 013452
3910 013444 004737 034672
3911
3912 013450 000411
3913
3914 013452
3915 013452 160400 000000
3916 013456 154000 000000
3917 013462 044600 000000
3918 013466 047557 047540
3919 013472 000000

```

```

*****
*TEST 176 TEST OF DIVF INSTR, DATA SET DIVF-4
*
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

TST176: SCOPE
MOV #DIVF4,R5 ; PTR TO TEST DATA SET
JSR PC,@DIVFT ; GO TEST
BR TST177 ;;

```

```

DIVF4: ; TEST DATA SET DIVF-4:
.WORD 040400,000000 ; INITIAL AC FLOAT NUMBER
.WORD 140500,000000 ; INITIAL MEM FLOAT NUMBER
.WORD 140052,ALTN+1 ; EXPECTED FLOAT RESULT
.WORD 047507,047510 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 177 TEST OF DIVF INSTR, DATA SET DIVF-5
*
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

TST177: SCOPE
MOV #DIVF5,R5 ; PTR TO TEST DATA SET
JSR PC,@DIVFT ; GO TEST
BR TST200 ;;

```

```

DIVF5: ; TEST DATA SET DIVF-5:
.WORD ALT4P,ALT4P ; INITIAL AC FLOAT NUMBER
.WORD ALT4P,ALT4P ; INITIAL MEM FLOAT NUMBER
.WORD FIP 0 ; EXPECTED FLOAT RESULT
.WORD 047417,047400 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 200 TEST OF DIVF INSTR, DATA SET DIVF-6
*
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

TST200: SCOPE
MOV #DIVF6,R5 ; PTR TO TEST DATA SET
JSR PC,@DIVFT ; GO TEST
BR TST201 ;;

```

```

DIVF6: ; TEST DATA SET DIVF-6:
.WORD 160400,000000 ; INITIAL AC FLOAT NUMBER
.WORD 154000,000000 ; INITIAL MEM FLOAT NUMBER
.WORD 044600,000000 ; EXPECTED FLOAT RESULT
.WORD 047557,047540 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```



```

3920
3921
3922
3923
3924
3925
3926
3927 013474 000004
3928 013476 012705 013510
3929 013502 004737 034672
3930
3931 013506 000411
3932
3933 013510
3934 013510 000177 177777
3935 013514 177777 177777
3936 013520 000000 000000
3937 013524 047453 047444
3938 013530 000000
3939
3940
3941
3942
3943
3944
3945
3946 013532 000004
3947 013534 012705 013546
3948 013540 004737 034672
3949
3950 013544 000411
3951
3952 013546
3953 013546 160077 000000
3954 013552 140277 000000
3955 013556 060000 000000
3956 013562 047517 047500
3957 013566 000000
3958
3959
3960
3961
3962
3963
3964
3965 013570 000004
3966 013572 012705 013604
3967 013576 004737 034672
3968
3969 013602 000411
3970
3971 013604
3972 013604 160077 000000
3973 013610 040277 000000
3974 013614 160000 000000
3975 013620 047447 047450

```

```

*****
:TEST 201 TEST OF DIVF INSTR, DATA SET DIVF-7
:*
:* ALL INTERRUPT ENABLES ON
:* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

†ST201: SCOPE
MOV #DIVF7,R5 ; PTR TO TEST DATA SET
JSR PC,@DIVFT ; GO TEST
BR TST202 ;;

```

```

DIVF7: ; TEST DATA SET DIVF-7:
.WORD ZXIMP,M1 ; INITIAL AC FLOAT NUMBER
.WORD LGN,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD 047453,047444 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
:TEST 202 TEST OF DIVF INSTR, DATA SET DIVF-10
:*
:* ALL INTERRUPT ENABLES ON
:* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

†ST202: SCOPE
MOV #DIVF10,R5 ; PTR TO TEST DATA SET
JSR PC,@DIVFT ; GO TEST
BR TST203 ;;

```

```

DIVF10: ; TEST DATA SET DIVF-10:
.WORD 160077,000000 ; INITIAL AC FLOAT NUMBER
.WORD 140277,000000 ; INITIAL MEM FLOAT NUMBER
.WORD 060000,000000 ; EXPECTED FLOAT RESULT
.WORD 047517,047500 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
:TEST 203 TEST OF DIVF INSTR, DATA SET DIVF-11
:*
:* ALL INTERRUPT ENABLES ON
:* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

†ST203: SCOPE
MOV #DIVF11,R5 ; PTR TO TEST DATA SET
JSR PC,@DIVFT ; GO TEST
BR TST204 ;;

```

```

DIVF11: ; TEST DATA SET DIVF-11:
.WORD 160077,000000 ; INITIAL AC FLOAT NUMBER
.WORD 040277,000000 ; INITIAL MEM FLOAT NUMBER
.WORD 160000,000000 ; EXPECTED FLOAT RESULT
.WORD 047447,047450 ; FPS: BEFORE, AFTER

```

```

3976 013624 000000
3977
3978
3979
3980
3981
3982
3983
3984 013626 000004
3985 013630 012705 013642
3986 013634 004737 034672
3987
3988 013640 000411
3989
3990 013642
3991 013642 060100 000001
3992 013646 040300 000000
3993 013652 060000 000001
3994 013656 047517 047500
3995 013662 000000
3996
3997
3998
3999
4000
4001
4002
4003 013664 000004
4004 013666 012705 013700
4005 013672 004737 034672
4006
4007 013676 000411
4008
4009 013700
4010 013700 060100 000001
4011 013704 140300 000000
4012 013710 160000 000000
4013 013714 047447 047450
4014 013720 000000
4015
4016
4017
4018
4019
4020
4021
4022 013722 000004
4023 013724 012705 013736
4024 013730 004737 034672
4025
4026 013734 000411
4027
4028 013736
4029 013736 000177 177777
4030 013742 100177 177777
4031 013746 000177 177777

```

```

.WORD NA ; FEC AFTER ( 0 = N/A )

;*****
;TEST 204 TEST OF DIVF INSTR, DATA SET DIVF-12
;* ALL INTERRUPT ENABLES ON
;* SHORT FLOAT, LONG INTEGER, ROUND MODES
;*****
TST204: SCOPE
MOV #DIVF12,R5 ; PTR TO TEST DATA SET
JSR PC,@#DIVFT ; GO TEST

BR TST205 ;;

DIVF12: ; TEST DATA SET DIVF-12:
.WORD 060100,000001 ; INITIAL AC FLOAT NUMBER
.WORD 040300,000000 ; INITIAL MEM FLOAT NUMBER
.WORD 060000,000001 ; EXPECTED FLOAT RESULT
.WORD 047517,047500 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

;*****
;TEST 205 TEST OF DIVF INSTR, DATA SET DIVF-13
;* ALL INTERRUPT ENABLES ON
;* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
;*****
TST205: SCOPE
MOV #DIVF13,R5 ; PTR TO TEST DATA SET
JSR PC,@#DIVFT ; GO TEST

BR TST206 ;;

DIVF13: ; TEST DATA SET DIVF-13:
.WORD 060100,000001 ; INITIAL AC FLOAT NUMBER
.WORD 140300,000000 ; INITIAL MEM FLOAT NUMBER
.WORD 160000,000000 ; EXPECTED FLOAT RESULT
.WORD 047447,047450 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

;*****
;TEST 206 TEST OF DIVF INSTR, DATA SET DIVF-14
;* ALL INTERRUPT ENABLES ON
;* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
;*****
TST206: SCOPE
MOV #DIVF14,R5 ; PTR TO TEST DATA SET
JSR PC,@#DIVFT ; GO TEST

BR TST207 ;;

DIVF14: ; TEST DATA SET DIVF-14:
.WORD ZXIMP,M1 ; INITIAL AC FLOAT NUMBER
.WORD ZXIMN,M1 ; INITIAL MEM FLOAT NUMBER
.WORD ZXIMP,M1 ; EXPECTED FLOAT RESULT

```

4032 013752 047543 147543
4033 013756 100014

.WORD 047543,147543 ; FPS: BEFORE, AFTER
.WORD 100014 ; FEC AFTER (0 = N/A)

4034
4035
4036
4037
4038
4039
4040

*TEST 207 TEST OF DIVF INSTR, DATA SET DIVF-15
* -0 INTERRUPT ENABLE OFF, ALL OTHERS ON
* SHORT FLOAT, SHORT INTEGER, ROUND MODES

4041 013760 000004
4042 013762 012705 013774
4043 013766 004737 034672

TST207: SCOPE
MOV #DIVF15,R5 ; PTR TO TEST DATA SET
JSR PC,#DIVFT ; GO TEST

4044
4045 013772 000411

BR TST210 ; ;

4046
4047 013774
4048 013774 000177 177777
4049 014000 100177 177777
4050 014004 000177 177777
4051 014010 043413 143413
4052 014014 100004

DIVF15: ; TEST DATA SET DIVF-15:
.WORD ZXIMP,M1 ; INITIAL AC FLOAT NUMBER
.WORD ZXIMN,M1 ; INITIAL MEM FLOAT NUMBER
.WORD ZXIMP,M1 ; EXPECTED FLOAT RESULT
.WORD 043413,143413 ; FPS: BEFORE, AFTER
.WORD 100004 ; FEC AFTER (0 = N/A)

4053
4054
4055

4056
4057
4058
4059

*TEST 210 TEST OF DIVF INSTR, DATA SET DIVF-16
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES

4060 014016 000004
4061 014020 012705 014032
4062 014024 004737 034672

TST210: SCOPE
MOV #DIVF16,R5 ; PTR TO TEST DATA SET
JSR PC,#DIVFT ; GO TEST

4063
4064 014030 000411

BR TST211 ; ;

4065
4066 014032
4067 014032 077777 052525
4068 014036 003777 170360
4069 014042 034177 062134
4070 014046 047515 147502
4071 014052 100010

DIVF16: ; TEST DATA SET DIVF-16:
.WORD LGP,ALTP ; INITIAL AC FLOAT NUMBER
.WORD 003777,ALT4N ; INITIAL MEM FLOAT NUMBER
.WORD 034177,062134 ; EXPECTED FLOAT RESULT
.WORD 047515,147502 ; FPS: BEFORE, AFTER
.WORD 100010 ; FEC AFTER (0 = N/A)

4072
4073
4074

4075
4076
4077

*TEST 211 TEST OF DIVF INSTR, DATA SET DIVF-17
* OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
* SHORT FLOAT, SHORT INTEGER, ROUND MODES

4078
4079 014054 000004
4080 014056 012705 014070
4081 014062 004737 034672

TST211: SCOPE
MOV #DIVF17,R5 ; PTR TO TEST DATA SET
JSR PC,#DIVFT ; GO TEST

4082
4083 014066 000411

BR TST212 ; ;

4084
4085 014070

4086 014070 177777 052525
4087 014074 103777 170360

DIVF17: ; TEST DATA SET DIVF-17:
.WORD LGN,ALTP ; INITIAL AC FLOAT NUMBER
.WORD 103777,ALT4N ; INITIAL MEM FLOAT NUMBER

4088 014100 000000 000000
4089 014104 046411 046406
4090 014110 000000

.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD 046411,046406 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER (0 = N/A)

4091
4092
4093
4094
4095
4096
4097

*TEST 212 TEST OF DIVF INSTR, DATA SET DIVF-20
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES

4098 014112 000004
4099 014114 012705 014126
4100 014120 004737 034672

TST212: SCOPE
MOV #DIVF20,R5 ; PTR TO TEST DATA SET
JSR PC,#DIVFT ; GO TEST

4101
4102 014124 000411
4103

BR TST213 ; ;

4104 014126
4105 014126 100200 000000
4106 014132 077777 177777
4107 014136 140400 000000
4108 014142 047547 147550
4109 014146 100012

DIVF20: ; TEST DATA SET DIVF-20:
.WORD SMN,0 ; INITIAL AC FLOAT NUMBER
.WORD LGP,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 140400,000000 ; EXPECTED FLOAT RESULT
.WORD 047547,147550 ; FPS: BEFORE AFTER
.WORD 100012 ; FEC AFTER (0 = N/A)

4110
4111
4112
4113
4114
4115
4116

*TEST 213 TEST OF DIVF INSTR, DATA SET DIVF-21
* UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES

4117 014150 000004
4118 014152 012705 014164
4119 014156 004737 034672

TST213: SCOPE
MOV #DIVF21,R5 ; PTR TO TEST DATA SET
JSR PC,#DIVFT ; GO TEST

4120
4121 014162 000411
4122

BR TST214 ; ;

4123 014164
4124 014164 000200 000000
4125 014170 177777 177777
4126 014174 000000 000000
4127 014200 045453 045444
4128 014204 000000

DIVF21: ; TEST DATA SET DIVF-21:
.WORD SMP,0 ; INITIAL AC FLOAT NUMBER
.WORD LGN,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD 045453,045444 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER (0 = N/A)

4129
4130
4131

```

4132 .....
4133 : *TEST 214 TEST OF DIVD INSTR, DATA SET DIVD-1
4134 : * ALL INTERRUPT ENABLES ON
4135 : * LONG FLOAT, SHORT INTEGER, ROUND MODES
4136 : .....
4137 014206 000004
4138 014210 012705 014222
4139 014214 004737 035042
4140
4141 014220 000417 BR TST215 ;;
4142
4143 014222 DIVD1: ; TEST DATA SET DIVD-1:
4144 014222 000177 177777 177777 .WORD ZXIMP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
4145 014230 177777
4146 014232 077777 177777 177777 .WORD LGP,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
4147 014240 177777
4148 014242 000000 000000 000000 .WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
4149 014250 000000
4150 014252 047613 047604 .WORD 047613,047604 ; FPS: BEFORE, AFTER
4151 014256 000000 .WORD NA ; FEC AFTER ( 0 = N/A )
4152
4153
4154 .....
4155 : *TEST 215 TEST OF DIVD INSTR, DATA SET DIVD-2
4156 : * ALL INTERRUPT ENABLES ON
4157 : * LONG FLOAT, LONG INTEGER, ROUND MODES
4158 : .....
4159 014260 000004
4160 014262 012705 014274
4161 014266 004737 035042
4162
4163 014272 000417 BR TST216 ;;
4164
4165 014274 DIVD2: ; TEST DATA SET DIVD-2:
4166 014274 034277 000000 000000 .WORD 034277,0,0,0 ; INITIAL AC FLOAT NUMBER
4167 014302 000000
4168 014304 040277 000000 000000 .WORD 040277,0,0,0 ; INITIAL MEM FLOAT NUMBER
4169 014312 000000
4170 014314 034200 000000 000000 .WORD 034200,0,0,0 ; EXPECTED FLOAT RESULT
4171 014322 000000
4172 014324 047717 047700 .WORD 047717,047700 ; FPS: BEFORE, AFTER
4173 014330 000000 .WORD NA ; FEC AFTER ( 0 = N/A )
4174
4175
4176 .....
4177 : *TEST 216 TEST OF DIVD INSTR, DATA SET DIVD-3
4178 : * ALL INTERRUPT ENABLES ON
4179 : * LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
4180 : .....
4181 014332 000004
4182 014334 012705 014346
4183 014340 004737 035042
4184
4185 014344 000417 BR TST217 ;;
4186
4187 014346 DIVD3: ; TEST DATA SET DIVD-3:

```

FPU ADVANCED INSTR TESTS
DOFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 86
T216 TEST OF DIVD INSTR, DATA SET DIVD-3

```

4188 014346 134277 000000 000000 .WORD 134277,0,0,0 ; INITIAL AC FLOAT NUMBER
4189 014354 000000 .WORD 040277,0,0,0 ; INITIAL MEM FLOAT NUMBER
4190 014356 040277 000000 000000 .WORD 134200,0,0,0 ; EXPECTED FLOAT RESULT
4191 014364 000000 .WORD 047647,047650 ; FPS: BEFORE, AFTER
4192 014366 134200 000000 000000 .WORD NA ; FEC AFTER ( 0 = N/A )
4193 014374 000000
4194 014376 047647 047650
4195 014402 000000
4196
4197
4198
4199
4200
4201
4202

```

```

*****
;TEST 217 TEST OF DIVD INSTR, DATA SET DIVD-4
;
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

4203 014404 000004 TST217: SCOPE
4204 014406 012705 014420 MOV #DIVD4,R5 ; PTR TO TEST DATA SET
4205 014412 004737 035042 JSR PC,#DIVDT ; GO TEST
4206
4207 014416 000417 BR TST220 ;;
4208

```

```

4209 014420 DIVD4: ; TEST DATA SET DIVD-4:
4210 014420 134300 000000 000000 .WORD 134300,0,0,1 ; INITIAL AC FLOAT NUMBER
4211 014426 000001 .WORD 140300,0,0,0 ; INITIAL MEM FLOAT NUMBER
4212 014430 140300 000000 000000 .WORD 034200,0,0,0 ; EXPECTED FLOAT RESULT
4213 014436 000000 .WORD 047757,047740 ; FPS: BEFORE, AFTER
4214 014440 034200 000000 000000 .WORD NA ; FEC AFTER ( 0 = N/A )
4215 014446 000000
4216 014450 047757 047740
4217 014454 000000
4218
4219
4220

```

```

*****
;TEST 220 TEST OF DIVD INSTR, DATA SET DIVD-5
;
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

4225 014456 000004 TST220: SCOPE
4226 014460 012705 014472 MOV #DIVD5,R5 ; PTR TO TEST DATA SET
4227 014464 004737 035042 JSR PC,#DIVDT ; GO TEST
4228
4229 014470 000417 BR TST221 ;;
4230

```

```

4231 014472 DIVD5: ; TEST DATA SET DIVD-5:
4232 014472 034300 000000 000000 .WORD 034300,0,0,1 ; INITIAL AC FLOAT NUMBER
4233 014500 000001 .WORD 140300,0,0,0 ; INITIAL MEM FLOAT NUMBER
4234 014502 140300 000000 000000 .WORD 134200,0,0,1 ; EXPECTED FLOAT RESULT
4235 014510 000000 .WORD 047607,047610 ; FPS: BEFORE, AFTER
4236 014512 134200 000000 000000 .WORD NA ; FEC AFTER ( 0 = N/A )
4237 014520 000001
4238 014522 047607 047610
4239 014526 000000
4240
4241
4242
4243

```

```

*****
;TEST 221 TEST OF DIVD INSTR, DATA SET DIVD-6

```

```

4244
4245
4246
4247 014530 000004
4248 014532 012705 014544
4249 014536 004737 035042
4250
4251 014542 000417
4252
4253 014544
4254 014544 100400 000000 000000
4255 014552 000000
4256 014554 000500 000000 000000
4257 014562 000000
4258 014564 140052 125252 125252
4259 014572 125252
4260 014574 047647 047650
4261 014600 000000
4262
4263
4264
4265
4266
4267
4268
4269 014602 000004
4270 014604 012705 014616
4271 014610 004737 035042
4272
4273 014614 000417
4274
4275 014616
4276 014616 100400 000000 000000
4277 014624 000000
4278 014626 000500 000000 000000
4279 014634 000000
4280 014636 140052 125252 125252
4281 014644 125253
4282 014646 047607 047610
4283 014652 000000
4284
4285
4286
4287
4288
4289
4290
4291 014654 000004
4292 014656 012705 014670
4293 014662 004737 035042
4294
4295 014666 000417
4296
4297 014670
4298 014670 170360 170360 170360
4299 014676 170360

```

```

;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
;*****
↑ST221: SCOPE
MOV #DIVD6,R5 ; PTR TO TEST DATA SET
JSR PC,@#DIVDT ; GO TEST
BR TST222 ;;
DIVD6: ; TEST DATA SET DIVD-6:
.WORD 100400,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD 000500,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 140052,ALTN,ALTN,ALTN ; EXPECTED FLOAT RESULT
.WORD 047647,047650 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
;*****
;*TEST 222 TEST OF DIVD INSTR, DATA SET DIVD-7
;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, SHORT INTEGER, ROUND MODES
;*****
↑ST222: SCOPE
MOV #DIVD7,R5 ; PTR TO TEST DATA SET
JSR PC,@#DIVDT ; GO TEST
BR TST223 ;;
DIVD7: ; TEST DATA SET DIVD-7:
.WORD 100400,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD 000500,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 140052,ALTN,ALTN,ALTN+1 ; EXPECTED FLOAT RESULT
.WORD 047607,047610 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
;*****
;*TEST 223 TEST OF DIVD INSTR, DATA SET DIVD-10
;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, LONG INTEGER, ROUND MODES
;*****
↑ST223: SCOPE
MOV #DIVD10,R5 ; PTR TO TEST DATA SET
JSR PC,@#DIVDT ; GO TEST
BR TST224 ;;
DIVD10: ; TEST DATA SET DIVD-10:
.WORD ALTN,ALTN,ALTN,ALTN ; INITIAL AC FLOAT NUMBER

```

FPU ADVANCED INSTR TESTS
D&FPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 88
T223 TEST OF DIVD INSTR, DATA SET DIVD-10

4300	014700	170360	170360	170360	.WORD	ALT4N,ALT4N,ALT4N,ALT4N ;	INITIAL MEM FLOAT NUMBER
4301	014706	170360					
4302	014710	040200	000000	000000	.WORD	F1P,0,0,0	; EXPECTED FLOAT RESULT
4303	014716	000000					
4304	014720	047717	047700		.WORD	047717,047700	; FPS: BEFORE, AFTER
4305	014724	000000			.WORD	NA	; FEC AFTER (0 = N/A)

4306
4307

```

*****
: *TEST 224 TEST OF DIVD INSTR, DATA SET DIVD-11
: *
: * ALL INTERRUPT ENABLES ON
: * LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
: *
*****

```

4313	014726	000004			TST224: SCOPE		
4314	014730	012705	014742		MOV	#DIVD11,R5	; PTR TO TEST DATA SET
4315	014734	004737	035042		JSR	PC,#DIVDT	; GO TEST
4316							
4317	014740	000417			BR	TST225	::

4318					DIVD11: ; TEST DATA SET DIVD-11:		
4319	014742						
4320	014742	070200	000000	000000	.WORD	070200,0,0,0	; INITIAL AC FLOAT NUMBER
4321	014750	000000					
4322	014752	050400	000000	000000	.WORD	050400,0,0,0	; INITIAL MEM FLOAT NUMBER
4323	014760	000000					
4324	014762	060000	000000	000000	.WORD	060000,0,0,0	; EXPECTED FLOAT RESULT
4325	014770	000000					
4326	014772	047657	047640		.WORD	047657,047640	; FPS: BEFORE, AFTER
4327	014776	000000			.WORD	NA	; FEC AFTER (0 = N/A)

4328
4329

```

*****
: *TEST 225 TEST OF DIVD INSTR, DATA SET DIVD-12
: *
: * ALL INTERRUPT ENABLES ON
: * LONG FLOAT, LONG INTEGER, ROUND MODES
: *
*****

```

4335	015000	000004			TST225: SCOPE		
4336	015002	012705	015014		MOV	#DIVD12,R5	; PTR TO TEST DATA SET
4337	015006	004737	035042		JSR	PC,#DIVDT	; GO TEST
4338							
4339	015012	000417			BR	TST226	::

4340					DIVD12: ; TEST DATA SET DIVD-12:		
4341	015014						
4342	015014	125252	125252	125252	.WORD	ALTN,ALTN,ALTN,ALTN	; INITIAL AC FLOAT NUMBER
4343	015022	125252					
4344	015024	000000	000000	000000	.WORD	0,0,0,0	; INITIAL MEM FLOAT NUMBER
4345	015032	000000					
4346	015034	125252	125252	125252	.WORD	ALTN,ALTN,ALTN,ALTN	; EXPECTED FLOAT RESULT
4347	015042	125252					
4348	015044	047707	147707		.WORD	047707,147707	; FPS: BEFORE, AFTER
4349	015050	100004			.WORD	100004	; FEC AFTER (0 = N/A)

4350
4351

```

*****
: *TEST 226 TEST OF DIVD INSTR, DATA SET DIVD-13
: *
: * ALL INTERRUPT ENABLES ON
: * LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
: *
*****

```

4352
4353
4354
4355


```

4356
4357 015052 000004
4358 015054 012705 015066
4359 015060 004737 035042
4360
4361 015064 000417
4362
4363 015066
4364 015066 000177 177777 177777
4365 015074 177777
4366 015076 100177 177777 177777
4367 015104 177777
4368 015106 000177 177777 177777
4369 015114 177777
4370 015116 047643 147643
4371 015122 100014
4372
4373
4374
4375
4376
4377
4378
4379 015124 000004
4380 015126 012705 015140
4381 015132 004737 035042
4382
4383 015136 000417
4384
4385 015140
4386 015140 000177 177777 177777
4387 015146 177777
4388 015150 100177 177777 177777
4389 015156 177777
4390 015160 000177 177777 177777
4391 015166 177777
4392 015170 043643 143643
4393 015174 100004
4394
4395
4396
4397
4398
4399
4400
4401 015176 000004
4402 015200 012705 015212
4403 015204 004737 035042
4404
4405 015210 000417
4406
4407 015212
4408 015212 052525 052525 052525
4409 015220 052525
4410 015222 000200 000000 000000
4411 015230 000000

```

```

*****
TST226: SCOPE
MOV      #DIVD13,R5      ; PTR TO TEST DATA SET
JSR      PC,@#DIVDT     ; GO TEST

BR       TST227         ;;

DIVD13: ; TEST DATA SET DIVD-13:
.WORD    ZXIMP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD    ZX1MN,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD    ZXIMP,M1,M1,M1 ; EXPECTED FLOAT RESULT
.WORD    047643,147643  ; FPS: BEFORE, AFTER
.WORD    100014         ; FEC AFTER ( 0 = N/A )

*****
*TEST 227 TEST OF DIVD INSTR, DATA SET DIVD-14
* -0 INTERRUPT ENABLE OFF, ALL OTHERS ON
* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST227: SCOPE
MOV      #DIVD14,R5     ; PTR TO TEST DATA SET
JSR      PC,@#DIVDT    ; GO TEST

BR       TST230         ;;

DIVD14: ; TEST DATA SET DIVD-14:
.WORD    ZXIMP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD    ZX1MN,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD    ZXIMP,M1,M1,M1 ; EXPECTED FLOAT RESULT
.WORD    043643,143643  ; FPS: BEFORE, AFTER
.WORD    100004         ; FEC AFTER ( 0 = N/A )

*****
*TEST 230 TEST OF DIVD INSTR, DATA SET DIVD-15
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****
TST230: SCOPE
MOV      #DIVD15,R5     ; PTR TO TEST DATA SET
JSR      PC,@#DIVDT    ; GO TEST

BR       TST231         ;;

DIVD15: ; TEST DATA SET DIVD-15:
.WORD    ALTP,ALTP,ALTP,ALTP ; INITIAL AC FLOAT NUMBER
.WORD    SMP,0,0,0        ; INITIAL MEM FLOAT NUMBER

```

K07

FPU ADVANCED INSTR TESTS
DQFP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 90
T230 TEST OF DIVD INSTR, DATA SET DIVD-15

4412	015232	012525	052525	052525	.WORD	012525,ALTP,ALTP,ALTP	; EXPECTED FLOAT RESULT
4413	015240	052525					
4414	015242	047615	147602		.WORD	047615,147602	; FPS: BEFORE, AFTER
4415	015246	100010			.WORD	100010	; FEC AFTER (0 = N/A)

```

4416
4417
4418
4419
4420
4421
4422
4423
4424
4425
4426
4427
4428
4429
4430
4431
4432
4433
4434
4435
4436
4437
4438
4439
4440
4441
4442
4443
4444
4445
4446
4447
4448
4449
4450
4451
4452
4453
4454
4455
4456
4457
4458
4459
4460
4461
4462
4463
4464
4465
4466
4467

```

```

*****
*TEST 231 TEST OF DIVD INSTR, DATA SET DIVD-16
* OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

TST231: SCOPE
MOV #DIVD16,RS ; PTR TO TEST DATA SET
JSR PC,@#DIVDT ; GO TEST
BR TST232 ;;

```

```

DIVD16: ; TEST DATA SET DIVD-16:
.WORD ALTP,ALTP,ALTP,ALTP ; INITIAL AC FLOAT NUMBER
.WORD SMN,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 046611,046606 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 232 TEST OF DIVD INSTR, DATA SET DIVD-17
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

TST232: SCOPE
MOV #DIVD17,RS ; PTR TO TEST DATA SET
JSR PC,@#DIVDT ; GO TEST
BR TST233 ;;

```

```

DIVD17: ; TEST DATA SET DIVD-17:
.WORD SMN,M1,ALTN,0 ; INITIAL AC FLOAT NUMBER
.WORD LGP,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 140400,M1,ALTN,1 ; EXPECTED FLOAT RESULT
.WORD 047707,147710 ; FPS: BEFORE, AFTER
.WORD 100012 ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 233 TEST OF DIVD INSTR, DATA SET DIVD-20
* UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

TST233: SCOPE

```

FPU ADVANCED INSTR TESTS
D9FP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 91
T233 TEST OF DIVD INSTR, DATA SET DIVD-20

4468	015376	012705	015410
4469	015402	004737	035042
4470			
4471	015406	000417	
4472			
4473	015410		
4474	015410	100200	177777 125252
4475	015416	000000	
4476	015420	177777	177777 177777
4477	015426	177777	
4478	015430	000000	000000 000000
4479	015436	000000	
4480	015440	045713	045704
4481	015444	000000	
4482			
4483			

```

MOV      #DIVD20,R5      ; PTR TO TEST DATA SET
JSR      PC,@#DIVDVT     ; GO TEST
BR       TST234          ;;

DIVD20: ; TEST DATA SET DIVD-20:
.WORD    SMN,M1,ALTN,0   ; INITIAL AC FLOAT NUMBER
.WORD    LGN,M1,M1,M1   ; INITIAL MEM FLOAT NUMBER
.WORD    0,0,0,0        ; EXPECTED FLOAT RESULT
.WORD    045713,045704  ; FPS: BEFORE, AFTER
.WORD    NA              ; FEC AFTER ( 0 = N/A )

```

M07

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 92
T234 TEST OF MODF(2 ACC) INSTR, DATA SET MD2F-1

4484			
4485			
4486			
4487			
4488			
4489	015446	000004	
4490	015450	012705	015462
4491	015454	004737	035232
4492			
4493	015460	000413	
4494			
4495	015462		
4496	015462	000000	000000
4497	015466	000000	000000
4498	015472	000000	000000
4499	015476	000000	000000
4500	015502	047513	047504
4501	015506	000000	
4502			
4503			
4504			
4505			
4506			
4507			
4508			
4509	015510	000004	
4510	015512	012705	015524
4511	015516	004737	035232
4512			
4513	015522	000413	
4514			
4515	015524		
4516	015524	000177	177777
4517	015530	077777	177777
4518	015534	000000	000000
4519	015540	000000	000000
4520	015544	047553	047544
4521	015550	000000	
4522			
4523			
4524			
4525			
4526			
4527			
4528			
4529	015552	000004	
4530	015554	012705	015566
4531	015560	004737	035232
4532			
4533	015564	000413	
4534			
4535	015566		
4536	015566	177777	177777
4537	015572	100177	177777
4538	015576	000000	000000
4539	015602	000000	000000

```

*****
*TEST 234      TEST OF MODF(2 ACC) INSTR, DATA SET MD2F-1
*              ALL INTERRUPT ENABLES ON
*              SHORT FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

TST234: SCOPE
MOV      #MD2F1,R5      ; PTR TO TEST DATA SET
JSR      PC,2#MD2FT    ; GO TEST
BR       TST235        ;;

```

```

MD2F1: ; TEST DATA SET MD2F-1:
.WORD   0,0            ; INITIAL AC FLOAT NUMBER
.WORD   0,0            ; INITIAL MEM FLOAT NUMBER
.WORD   0,0            ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD   0,0            ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD   047513,047504 ; FPS: BEFORE, AFTER
.WORD   NA             ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 235      TEST OF MODF(2 ACC) INSTR, DATA SET MD2F-2
*              ALL INTERRUPT ENABLES ON
*              SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

TST235: SCOPE
MOV      #MD2F2,R5      ; PTR TO TEST DATA SET
JSR      PC,2#MD2FT    ; GO TEST
BR       TST236        ;;

```

```

MD2F2: ; TEST DATA SET MD2F-2:
.WORD   ZXIMP,M1       ; INITIAL AC FLOAT NUMBER
.WORD   LGP,M1         ; INITIAL MEM FLOAT NUMBER
.WORD   0,0            ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD   0,0            ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD   047553,047544 ; FPS: BEFORE, AFTER
.WORD   NA             ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 236      TEST OF MODF(2 ACC) INSTR, DATA SET MD2F-3
*              -0 INTERRUPT ENABLE OFF, ALL OTHERS ON
*              SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

TST236: SCOPE
MOV      #MD2F3,R5      ; PTR TO TEST DATA SET
JSR      PC,2#MD2FT    ; GO TEST
BR       TST237        ;;

```

```

MD2F3: ; TEST DATA SET MD2F-3:
.WORD   LGN,M1         ; INITIAL AC FLOAT NUMBER
.WORD   ZXIMN,M1       ; INITIAL MEM FLOAT NUMBER
.WORD   0,0            ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD   0,0            ; EXPECTED INTEGER-PART FLOAT RESULT

```

4540 015606 043413 043404
4541 015612 000000

.WORD 043413,043404 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

4542
4543
4544
4545
4546
4547
4548

; *TEST 237 TEST OF MODF(2 ACC) INSTR, DATA SET MD2F-4
; * ALL INTERRUPT ENABLES ON
; * SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES

4549 015614 000004
4550 015616 012705 015630
4551 015622 004737 035232

†ST237: SCOPE
MOV #MD2F4,R5 ; PTR TO TEST DATA SET
JSR PC,2#MD2FT ; GO TEST

4552
4553 015626 000413

BR TST240 ; ;

4554
4555 015630
4556 015630 177777 177777
4557 015634 100177 177777
4558 015640 177777 177777
4559 015644 052525 177777
4560 015650 047447 147447
4561 015654 100014

MD2F4: ; TEST DATA SET MD2F-4:
.WORD LGN,M1 ; INITIAL AC FLOAT NUMBER
.WORD ZX1#N,M1 ; INITIAL MEM FLOAT NUMBER
.WORD LGN,M1 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD ALTP,M1 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047447,147447 ; FPS: BEFORE, AFTER
.WORD 100014 ; FEC AFTER (0 = N/A)

4562
4563
4564
4565
4566
4567
4568

; *TEST 240 TEST OF MODF(2 ACC) INSTR, DATA SET MD2F-5
; * ALL INTERRUPT ENABLES ON
; * SHORT FLOAT, LONG INTEGER, TRUNCATE MODES

4569 015656 000004
4570 015660 012705 015672
4571 015664 004737 035232

†ST240: SCOPE
MOV #MD2F5,R5 ; PTR TO TEST DATA SET
JSR PC,2#MD2FT ; GO TEST

4572
4573 015670 000413

BR TST241 ; ;

4574
4575 015672
4576 015672 042177 000000
4577 015676 140200 000000
4578 015702 000000 000000
4579 015706 142177 000000
4580 015712 047553 047544
4581 015716 000000

MD2F5: ; TEST DATA SET MD2F-5:
.WORD 042177,0 ; INITIAL AC FLOAT NUMBER
.WORD FIN,0 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD 142177,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047553,047544 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

4582
4583
4584
4585
4586
4587
4588

; *TEST 241 TEST OF MODF(2 ACC) INSTR, DATA SET MD2F-6
; * ALL INTERRUPT ENABLES ON
; * SHORT FLOAT, LONG INTEGER, ROUND MODES

4589 015720 000004
4590 015722 012705 015734
4591 015726 004737 035232

†ST241: SCOPE
MOV #MD2F6,R5 ; PTR TO TEST DATA SET
JSR PC,2#MD2FT ; GO TEST

4592
4593 015732 000413

BR TST242 ; ;

4594
4595 015734

MD2F6: ; TEST DATA SET MD2F-6:

4596	015734	040200	000000
4597	015740	140177	177777
4598	015744	140177	177777
4599	015750	000000	000000
4600	015754	047507	047510
4601	015760	000000	

```
.WORD FIP,0 ; INITIAL AC FLOAT NUMBER
.WORD 140177,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 140177,M1 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD 0,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047507,047510 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
```

4602
4603
4604

```
*****
*TEST 242 TEST OF MODF(2 ACC) INSTR, DATA SET MD2F-7
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
```

4609	015762	000004	
4610	015764	012705	015776
4611	015770	004737	035232

```
TST242: SCOPE
MOV #MD2F7,R5 ; PTR TO TEST DATA SET
JSR PC,2#MD2FT ; GO TEST
BR TST243 ;;
```

4613	015774	000413	
4615	015776		
4616	015776	142176	077600
4617	016002	140200	000000
4618	016006	037777	000000
4619	016012	042176	000000
4620	016016	047457	047440
4621	016022	000000	

```
MD2F7: ; TEST DATA SET MD2F-7:
.WORD 142176,077600 ; INITIAL AC FLOAT NUMBER
.WORD FIP,0 ; INITIAL MEM FLOAT NUMBER
.WORD 037777,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD 042176,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047457,047440 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
```

4622
4623
4624

```
*****
*TEST 243 TEST OF MODF(2 ACC) INSTR, DATA SET MD2F-10
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****
```

4629	016024	000004	
4630	016026	012705	016040
4631	016032	004737	035232

```
TST243: SCOPE
MOV #MD2F10,R5 ; PTR TO TEST DATA SET
JSR PC,2#MD2FT ; GO TEST
BR TST244 ;;
```

4633	016036	000413	
4635	016040		
4636	016040	042177	100000
4637	016044	040200	000000
4638	016050	040000	000000
4639	016054	042177	000000
4640	016060	047417	047400
4641	016064	000000	

```
MD2F10: ; TEST DATA SET MD2F-10:
.WORD 042177,M0 ; INITIAL AC FLOAT NUMBER
.WORD FIP,0 ; INITIAL MEM FLOAT NUMBER
.WORD 040000,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD 042177,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047417,047400 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
```

4642
4643
4644

```
*****
*TEST 244 TEST OF MODF(2 ACC) INSTR, DATA SET MD2F-11
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****
```

4649	016066	000004	
4650	016070	012705	016102
4651	016074	004737	035232

```
TST244: SCOPE
MOV #MD2F11,R5 ; PTR TO TEST DATA SET
JSR PC,2#MD2FT ; GO TEST
```

4652
4653 016100 000413
4654
4655 016102
4656 016102 140200 000000
4657 016106 040377 177777
4658 016112 140177 177776
4659 016116 140200 000000
4660 016122 047547 047550
4661 016126 000000
4662
4663
4664
4665
4666
4667
4668
4669 016130 000004
4670 016132 012705 016144
4671 016136 004737 035232
4672
4673 016142 000413
4674
4675 016144
4676 016144 060452 125252
4677 016150 021700 000000
4678 016154 040177 177400
4679 016160 042177 000000
4680 016164 047517 047500
4681 016170 000000
4682
4683
4684
4685
4686
4687
4688
4689 016172 000004
4690 016174 012705 016206
4691 016200 004737 035232
4692
4693 016204 000413
4694
4695 016206
4696 016206 041000 000001
4697 016212 141377 177776
4698 016216 140177 177777
4699 016222 142177 000000
4700 016226 047547 047550
4701 016232 000000
4702
4703
4704
4705
4706
4707

```

BR      TST245      ;;

MD2F11: ; TEST DATA SET MD2F-11:
        .WORD  FIN 0      ; INITIAL AC FLOAT NUMBER
        .WORD  040377,M1  ; INITIAL MEM FLOAT NUMBER
        .WORD  140177,M2  ; EXPECTED FRACTION-PART FLOAT RESULT
        .WORD  FIN 0      ; EXPECTED INTEGER-PART FLOAT RESULT
        .WORD  047547,047550 ; FPS: BEFORE, AFTER
        .WORD  NA          ; FEC AFTER ( 0 = N/A )

```

```

*****
; TEST 245      TEST OF MODF(2 ACC) INSTR, DATA SET MD2F-12
;
; ALL INTERRUPT ENABLES ON
; SHORT FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

†TST245: SCOPE
        MOV     #MD2F12,R5    ; PTR TO TEST DATA SET
        JSR    PC,@#MD2FT    ; GO TEST

```

```
BR      TST246      ;;
```

```

MD2F12: ; TEST DATA SET MD2F-12:
        .WORD  060452,ALTN  ; INITIAL AC FLOAT NUMBER
        .WORD  021700,0     ; INITIAL MEM FLOAT NUMBER
        .WORD  040177,UB    ; EXPECTED FRACTION-PART FLOAT RESULT
        .WORD  042177,0     ; EXPECTED INTEGER-PART FLOAT RESULT
        .WORD  047517,047500 ; FPS: BEFORE, AFTER
        .WORD  NA          ; FEC AFTER ( 0 = N/A )

```

```

*****
; TEST 246      TEST OF MODF(2 ACC) INSTR, DATA SET MD2F-13
;
; ALL INTERRUPT ENABLES ON
; SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

†TST246: SCOPE
        MOV     #MD2F13,R5    ; PTR TO TEST DATA SET
        JSR    PC,@#MD2FT    ; GO TEST

```

```
BR      TST247      ;;
```

```

MD2F13: ; TEST DATA SET MD2F-13:
        .WORD  041000,000001 ; INITIAL AC FLOAT NUMBER
        .WORD  141377,M2     ; INITIAL MEM FLOAT NUMBER
        .WORD  140177,M1     ; EXPECTED FRACTION-PART FLOAT RESULT
        .WORD  142177,000000 ; EXPECTED INTEGER-PART FLOAT RESULT
        .WORD  047547,047550 ; FPS: BEFORE, AFTER
        .WORD  NA          ; FEC AFTER ( 0 = N/A )

```

```

*****
; TEST 247      TEST OF MODF(2 ACC) INSTR, DATA SET MD2F-14
;
; ALL INTERRUPT ENABLES ON
; SHORT FLOAT, LONG INTEGER, ROUND MODES

```

```

4708
4709 016234 000004
4710 016236 012705 016250
4711 016242 004737 035232
4712
4713 016246 000413
4714
4715 016250
4716 016250 041000 000001
4717 016254 141377 177776
4718 016260 140200 000000
4719 016264 142177 000000
4720 016270 047507 047510
4721 016274 000000
4722
4723
4724
4725
4726
4727
4728
4729 016276 000004
4730 016300 012705 016312
4731 016304 004737 035232
4732
4733 016310 000413
4734
4735 016312
4736 016312 077600 000000
4737 016316 044452 125252
4738 016322 000000 000000
4739 016326 004052 125252
4740 016332 047411 147406
4741 016336 100010
4742
4743
4744
4745
4746
4747
4748
4749 016340 000004
4750 016342 012705 016354
4751 016346 004737 035232
4752
4753 016352 000413
4754
4755 016354
4756 016354 077600 000000
4757 016360 044452 125252
4758 016364 000000 000000
4759 016370 000000 000000
4760 016374 046411 046406
4761 016400 000000
4762
4763

```

```

*****
↑TST247: SCOPE
MOV #MD2F14,R5 ; PTR TO TEST DATA SET
JSR PC,@MD2FT ; GO TEST
BR TST250 ;;
MD2F14: ; TEST DATA SET MD2F-14:
.WORD 041000,000001 ; INITIAL AC FLOAT NUMBER
.WORD 141377,M2 ; INITIAL MEM FLOAT NUMBER
.WORD 140200,000000 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD 142177,000000 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047507,047510 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 250 TEST OF MODF(2 ACC) INSTR, DATA SET MD2F-15
*
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

↑TST250: SCOPE
MOV #MD2F15,R5 ; PTR TO TEST DATA SET
JSR PC,@MD2FT ; GO TEST
BR TST251 ;;
MD2F15: ; TEST DATA SET MD2F-15:
.WORD 077600,0 ; INITIAL AC FLOAT NUMBER
.WORD 044452,ALTN ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD 004052,ALTN ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047411,147406 ; FPS: BEFORE, AFTER
.WORD 100010 ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 251 TEST OF MODF(2 ACC) INSTR, DATA SET MD2F-16
*
* OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
* SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

↑TST251: SCOPE
MOV #MD2F16,R5 ; PTR TO TEST DATA SET
JSR PC,@MD2FT ; GO TEST
BR TST252 ;;
MD2F16: ; TEST DATA SET MD2F-16:
.WORD 077600,0 ; INITIAL AC FLOAT NUMBER
.WORD 044452,ALTN ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD 0,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 046411,046406 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```


4764
 4765
 4766
 4767
 4768
 4769 016402 000004
 4770 016404 012705 016416
 4771 016410 004737 035232
 4772
 4773 016414 000413
 4774
 4775 016416
 4776 016416 001577 177777
 4777 016422 101000 000000
 4778 016426 142377 177777
 4779 016432 000000 000000
 4780 016436 047547 147550
 4781 016442 100012
 4782
 4783
 4784
 4785
 4786
 4787
 4788
 4789 016444 000004
 4790 016446 012705 016460
 4791 016452 004737 035232
 4792
 4793 016456 000413
 4794
 4795 016460
 4796 016460 001577 177777
 4797 016464 101000 000000
 4798 016470 000000 000000
 4799 016474 000000 000000
 4800 016500 045553 045544
 4801 016504 000000
 4802
 4803
 4804

```

*****
: *TEST 252      TEST OF MODF(2 ACC) INSTR, DATA SET MD2F-17
: *              ALL INTERRUPT ENABLES ON
: *              SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

↑ST252: SCOPE
      MOV      #MD2F17,R5      ; PTR TO TEST DATA SET
      JSR      PC,@#MD2FT     ; GO TEST
      BR       TST253        ;;

```

```

MD2F17: ; TEST DATA SET MD2F-17:
      .WORD    001577,M1      ; INITIAL AC FLOAT NUMBER
      .WORD    101000,0      ; INITIAL MEM FLOAT NUMBER
      .WORD    142377,M1      ; EXPECTED FRACTION-PART FLOAT RESULT
      .WORD    0,0           ; EXPECTED INTEGER-PART FLOAT RESULT
      .WORD    047547,147550 ; FPS: BEFORE, AFTER
      .WORD    100012        ; FEC AFTER ( 0 = N/A )

```

```

*****
: *TEST 253      TEST OF MODF(2 ACC) INSTR, DATA SET MD2F-20
: *              UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
: *              SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

↑ST253: SCOPE
      MOV      #MD2F20,R5     ; PTR TO TEST DATA SET
      JSR      PC,@#MD2FT     ; GO TEST
      BR       TST254        ;;

```

```

MD2F20: ; TEST DATA SET MD2F-20:
      .WORD    001577,M1      ; INITIAL AC FLOAT NUMBER
      .WORD    101000,0      ; INITIAL MEM FLOAT NUMBER
      .WORD    0,0           ; EXPECTED FRACTION-PART FLOAT RESULT
      .WORD    0,0           ; EXPECTED INTEGER-PART FLOAT RESULT
      .WORD    045553,045544 ; FPS: BEFORE, AFTER
      .WORD    NA            ; FEC AFTER ( 0 = N/A )

```

F08

FPU ADVANCED INSTR TESTS
D0FPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 98
T254 TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-1

```

4805
4806
4807
4808
4809
4810 016506 000004
4811 016510 012705 016522
4812 016514 004737 035434
4813
4814 016520 000423
4815
4816 016522
4817 016522 000000 000000 000000
4818 016530 000000
4819 016532 000000 000000 000000
4820 016540 000000
4821 016542 000000 000000 000000
4822 016550 000000
4823 016552 000000 000000 000000
4824 016560 000000
4825 016562 047653 047644
4826 016566 000000
4827
4828
4829
4830
4831
4832
4833
4834 016570 000004
4835 016572 012705 016604
4836 016576 004737 035434
4837
4838 016602 000423
4839
4840 016604
4841 016604 000177 177777 177777
4842 016612 177777
4843 016614 177777 177777 177777
4844 016622 177777
4845 016624 000000 000000 000000
4846 016632 000000
4847 016634 000000 000000 000000
4848 016642 000000
4849 016644 047713 047704
4850 016650 000000
4851
4852
4853
4854
4855
4856
4857
4858 016652 000004
4859 016654 012705 016666
4860 016660 004737 035434

```

```

*****
; TEST 254 TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-1
; *
; * ALL INTERRUPT ENABLES ON
; * LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
; *
*****
†ST254: SCOPE
MOV #MD2D1,R5 ; PTR TO TEST DATA SET
JSR PC,@MD2DT ; GO TEST
BR TST255 ;;

MD2D1: ; TEST DATA SET MD2D-1:
.WORD 0,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD 0,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD 0,0,0,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047653,047644 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
; TEST 255 TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-2
; *
; * ALL INTERRUPT ENABLES ON
; * LONG FLOAT, LONG INTEGER, ROUND MODES
; *
*****
†ST255: SCOPE
MOV #MD2D2,R5 ; PTR TO TEST DATA SET
JSR PC,@MD2DT ; GO TEST
BR TST256 ;;

MD2D2: ; TEST DATA SET MD2D-2:
.WORD ZXIMP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD LGN,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD 0,0,0,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047713,047704 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
; TEST 256 TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-3
; *
; * -0 INTERRUPT ENABLE OFF, ALL OTHERS ON
; * LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
; *
*****
†ST256: SCOPE
MOV #MD2D3,R5 ; PTR TO TEST DATA SET
JSR PC,@MD2DT ; GO TEST

```

```

4861
4862 016664 000423          BR      TST257          ;;
4863
4864 016666          MD203: ; TEST DATA SET MD20-3:
4865 016666 077777 177777 177777 .WORD  LGP,M1,M1,M1  ; INITIAL AC FLOAT NUMBER
4866 016674 177777
4867 016676 100177 177777 177777 .WORD  ZX1MN,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
4868 016704 177777
4869 016706 000000 000000 000000 .WORD  0,0,0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
4870 016714 000000
4871 016716 000000 000000 000000 .WORD  0,0,0,0 ; EXPECTED INTEGER-PART FLOAT RESULT
4872 016724 000000
4873 016726 043653 043644 .WORD  043653,043644 ; FPS: BEFORE, AFTER
4874 016732 000000 .WORD  NA ; FEC AFTER ( 0 = N/A )
4875
4876
4877
4878 ;*****
4879 ;*TEST 257 TEST OF MODD(2 ACC) INSTR, DATA SET MD20-4
4880 ;*
4881 ;* ALL INTERRUPT ENABLES ON
4882 ;* LONG FLOAT, LONG INTEGER, ROUND MODES
4883 ;*****
4882 016734 000004          †T257: SCOPE
4883 016736 012705 016750      MOV      #MD204,R5 ; PTR TO TEST DATA SET
4884 016742 004737 035434      JSR      PC,#MD20T ; GO TEST
4885
4886 016746 000423          BR      TST260          ;;
4887
4888 016750          MD204: ; TEST DATA SET MD20-4:
4889 016750 077777 177777 177777 .WORD  LGP,M1,M1,M1  ; INITIAL AC FLOAT NUMBER
4890 016756 177777
4891 016760 100177 177777 177777 .WORD  ZX1MN,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
4892 016766 177777
4893 016770 077777 177777 177777 .WORD  LGP,M1,M1,M1  ; EXPECTED FRACTION-PART FLOAT RESULT
4894 016776 177777
4895 017000 052525 177777 125252 .WORD  ALTP,M1,ALTN,0 ; EXPECTED INTEGER-PART FLOAT RESULT
4896 017006 000000
4897 017010 047713 147713 .WORD  047713,147713 ; FPS: BEFORE, AFTER
4898 017014 100014 .WORD  100014 ; FEC AFTER ( 0 = N/A )
4899
4900
4901 ;*****
4902 ;*TEST 260 TEST OF MODD(2 ACC) INSTR, DATA SET MD20-5
4903 ;*
4904 ;* ALL INTERRUPT ENABLES ON
4905 ;* LONG FLOAT, SHORT INTEGER, ROUND MODES
4906 ;*****
4906 017016 000004          †T260: SCOPE
4907 017020 012705 017032      MOV      #MD205,R5 ; PTR TO TEST DATA SET
4908 017024 004737 035434      JSR      PC,#MD20T ; GO TEST
4909
4910 017030 000423          BR      TST261          ;;
4911
4912 017032          MD205: ; TEST DATA SET MD20-5:
4913 017032 042177 000000 000000 .WORD  042177,0,0,0  ; INITIAL AC FLOAT NUMBER
4914 017040 000000
4915 017042 040200 000000 000000 .WORD  F1P,0,0,0 ; INITIAL MEM FLOAT NUMBER
4916 017050 000000

```

H08

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 100
T260 TEST CF MODD(2 ACC) INSTR, DATA SET MD2D-5

4917	017052	000000	000000	000000	.WORD	0,0,0,0	; EXPECTED FRACTION-PART FLOAT RESULT
4918	017060	000000					
4919	017062	042177	000000	000000	.WORD	042177,0,0,0	; EXPECTED INTEGER-PART FLOAT RESULT
4920	017070	000000					
4921	017072	047613	047604		.WORD	047613,047604	; FPS: BEFORE AFTER
4922	017076	000000			.WORD	NA	; FEC AFTER (0 = N/A)

```

4923
4924
4925
4926
4927
4928
4929
*****
; TEST 261 TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-6
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

4930	017100	000004			TST261: SCOPE		
4931	017102	012705	017114		MOV	#MD206,R5	; PTR TO TEST DATA SET
4932	017106	004737	035434		JSR	PC,#MD20T	; GO TEST
4933							
4934	017112	000423			BR	TST262	::
4935							
4936	017114				MD206: ; TEST DATA SET MD2D-6:		
4937	017114	140200	000000	000000	.WORD	F1N,0,0,0	; INITIAL AC FLOAT NUMBER
4938	017122	000000					
4939	017124	040177	177777	177777	.WORD	040177,M1,M1,M1	; INITIAL MEM FLOAT NUMBER
4940	017132	177777					
4941	017134	140177	177777	177777	.WORD	140177,M1,M1,M1	; EXPECTED FRACTION-PART FLOAT RESULT
4942	017142	177777					
4943	017144	000000	000000	000000	.WORD	0,0,0,0	; EXPECTED INTEGER-PART FLOAT RESULT
4944	017152	000000					
4945	017154	047747	047750		.WORD	047747,047750	; FPS: BEFORE AFTER
4946	017160	000000			.WORD	NA	; FEC AFTER (0 = N/A)

```

4947
4948
4949
4950
4951
4952
4953
*****
; TEST 262 TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-7
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

4954	017162	000004			TST262: SCOPE		
4955	017164	012705	017176		MOV	#MD207,R5	; PTR TO TEST DATA SET
4956	017170	004737	035434		JSR	PC,#MD20T	; GO TEST
4957							
4958	017174	000423			BR	TST263	::
4959							
4960	017176				MD207: ; TEST DATA SET MD2D-7:		
4961	017176	042176	077600	000000	.WORD	042176,077600,0,0	; INITIAL AC FLOAT NUMBER
4962	017204	000000					
4963	017206	140200	000000	000000	.WORD	F1N,0,0,0	; INITIAL MEM FLOAT NUMBER
4964	017214	000000					
4965	017216	137777	000000	000000	.WORD	137777,0,0,0	; EXPECTED FRACTION-PART FLOAT RESULT
4966	017224	000000					
4967	017226	142176	000000	000000	.WORD	142176,0,0,0	; EXPECTED INTEGER-PART FLOAT RESULT
4968	017234	000000					
4969	017236	047607	047610		.WORD	047607,047610	; FPS: BEFORE AFTER
4970	017242	000000			.WORD	NA	; FEC AFTER (0 = N/A)

4971
4972

```

4973
4974
4975
4976
4977
4978 017244 000004
4979 017246 012705 017260
4980 017252 004737 035434
4981
4982 017256 000423
4983
4984 017260
4985 017260 142177 100000 000000
4986 017266 000000
4987 017270 040200 000000 000000
4988 017276 000000
4989 017300 140000 000000 000000
4990 017306 000000
4991 017310 142177 000000 000000
4992 017316 000000
4993 017320 047747 047750
4994 017324 000000
4995
4996
4997
4998
4999
5000
5001
5002 017326 000004
5003 017330 012705 017342
5004 017334 004737 035434
5005
5006 017340 000423
5007
5008 017342
5009 017342 140200 000000 000000
5010 017350 000000
5011 017352 140377 177777 177777
5012 017360 177777
5013 017362 040177 177777 177777
5014 017370 177776
5015 017372 040200 000000 000000
5016 017400 000000
5017 017402 047617 047600
5018 017406 000000
5019
5020
5021
5022
5023
5024
5025
5026 017410 000004
5027 017412 012705 017424
5028 017416 004737 035434

```

```

*****
*TEST 263 TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-10
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****
†ST263: SCOPE
MOV #MD2D10,R5 ; PTR TO TEST DATA SET
JSR PC, @MD2DT ; GO TEST

BR TST264 ;;

MD2D10: ; TEST DATA SET MD2D-10:
.WORD 142177,MO,0,0 ; INITIAL AC FLOAT NUMBER
.WORD FIP,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 140000,0,0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD 142177,0,0,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047747,047750 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

4997
4998
4999
5000
5001
5002 017326 000004
5003 017330 012705 017342
5004 017334 004737 035434
5005
5006 017340 000423
5007
5008 017342
5009 017342 140200 000000 000000
5010 017350 000000
5011 017352 140377 177777 177777
5012 017360 177777
5013 017362 040177 177777 177777
5014 017370 177776
5015 017372 040200 000000 000000
5016 017400 000000
5017 017402 047617 047600
5018 017406 000000
5019
5020
5021
5022
5023
5024
5025
5026 017410 000004
5027 017412 012705 017424
5028 017416 004737 035434

```

```

*****
*TEST 264 TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-11
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****
†ST264: SCOPE
MOV #MD2D11,R5 ; PTR TO TEST DATA SET
JSR PC, @MD2DT ; GO TEST

BR TST265 ;;

MD2D11: ; TEST DATA SET MD2D-11:
.WORD FIN,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD 140377,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 040177,M1,M1,M2 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD FIP,0,0,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047617,047600 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

5021
5022
5023
5024
5025
5026 017410 000004
5027 017412 012705 017424
5028 017416 004737 035434

```

```

*****
*TEST 265 TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-12
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****
†ST265: SCOPE
MOV #MD2D12,R5 ; PTR TO TEST DATA SET
JSR PC, @MD2DT ; GO TEST

```

J08

FPU ADVANCED INSTR TESTS
 DDFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 102
 T265 TEST OF MODD(2 ACC) INSTR, DATA SET MD20-12

```

5029
5030 017422 000423          BR      TST266          ;;
5031
5032 017424          MD2012: ; TEST DATA SET MD20-12:
5033 017424 167452 125252 125252 .WORD 167452,AN,AN,AN ; INITIAL AC FLOAT NUMBER
5034 017432 125252
5035 017434 112700 000000 000000 .WORD 112700,0,0,0 ; INITIAL MEM FLOAT NUMBER
5036 017442 000000
5037 017444 040177 177777 177777 .WORD 040177,M1,M1,UB ; EXPECTED FRACTION-PART FLOAT RESULT
5038 017452 177400
5039 017454 042177 000000 000000 .WORD 042177,0,0,0 ; EXPECTED INTEGER-PART FLOAT RESULT
5040 017462 000000
5041 017464 047757 047740 .WORD 047757,047740 ; FPS: BEFORE, AFTER
5042 017470 000000 .WORD NA ; FEC AFTER ( 0 = N/A )
5043
5044
5045
5046
5047
5048
5049

```

```

*****
; *TEST 266 TEST OF MODD(2 ACC) INSTR, DATA SET MD20-13
; * ALL INTERRUPT ENABLES ON
; * LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

5050 017472 000004          †ST266: SCOPE
5051 017474 012705 017506      MOV      #MD2013,R5 ; PTR TO TEST DATA SET
5052 017500 004737 035434      JSR      PC,2#MD2DT ; GO TEST
5053
5054 017504 000423          BR      TST267          ;;
5055

```

```

5056 017506          MD2013: ; TEST DATA SET MD20-13:
5057 017506 041000 000000 000000 .WORD 041000,0,0,1 ; INITIAL AC FLOAT NUMBER
5058 017514 000001
5059 017516 037577 177777 177777 .WORD 037577,M1,M1,M2 ; INITIAL MEM FLOAT NUMBER
5060 017524 177776
5061 017526 040177 177777 177777 .WORD 040177,M1,M1,M1 ; EXPECTED FRACTION-PART FLOAT RESULT
5062 017534 177777
5063 017536 040200 000000 000000 .WORD 040200,0,0,0 ; EXPECTED INTEGER-PART FLOAT RESULT
5064 017544 000000
5065 017546 047657 047640 .WORD 047657,047640 ; FPS: BEFORE, AFTER
5066 017552 000000 .WORD NA ; FEC AFTER ( 0 = N/A )
5067
5068
5069

```

```

*****
; *TEST 267 TEST OF MODD(2 ACC) INSTR, DATA SET MD20-14
; * ALL INTERRUPT ENABLES ON
; * LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

5070
5071
5072
5073
5074 017554 000004          †ST267: SCOPE
5075 017556 012705 017570      MOV      #MD2014,R5 ; PTR TO TEST DATA SET
5076 017562 004737 035434      JSR      PC,2#MD2DT ; GO TEST
5077
5078 017566 000423          BR      TST270          ;;
5079

```

```

5080 017570          MD2014: ; TEST DATA SET MD20-14:
5081 017570 041000 000000 000000 .WORD 041000,0,0,1 ; INITIAL AC FLOAT NUMBER
5082 017576 000001
5083 017600 037577 177777 177777 .WORD 037577,M1,M1,M2 ; INITIAL MEM FLOAT NUMBER
5084 017606 177776

```

K08

FPU ADVANCED INSTR TESTS
D9FP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 103
T267 TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-14

5085	017610	040200	000000	000000	.WORD	040200,0,0,0	; EXPECTED FRACTION-PART FLOAT RESULT
5086	017616	000000					
5087	017620	040200	000000	000000	.WORD	040200,0,0,0	; EXPECTED INTEGER-PART FLOAT RESULT
5088	017626	000000					
5089	017630	047617	047600		.WORD	047617,047600	; FPS: BEFORE, AFTER
5090	017634	000000			.WORD	NA	; FEC AFTER (0 = N/A)

```

5091
5092
5093
5094
5095
5096
5097
5098
5099
5100
5101
5102
5103
5104
5105
5106
5107
5108
5109
5110
5111
5112
5113
5114
5115
5116
5117
5118
5119
5120
5121
5122
5123
5124
5125
5126
5127
5128
5129
5130
5131
5132
5133
5134
5135
5136
5137
5138
5139
5140

```

```

*****
; *TEST 270 TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-15
; *
; * ALL INTERRUPT ENABLES ON
; * LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
; *
*****

```

```

†TST270: SCOPE
MOV #MD2015,R5 ; PTR TO TEST DATA SET
JSR PC,@MD2DT ; GO TEST
BR TST271 ;;

```

```

MD2015: ; TEST DATA SET MD2D-15:
.WORD 140452,AN,AN,AN ; INITIAL AC FLOAT NUMBER
.WORD 077600,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD 100052,AN,AN,AN ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047651,147646 ; FPS: BEFORE, AFTER
.WORD 100010 ; FEC AFTER ( 0 = N/A )

```

```

*****
; *TEST 271 TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-16
; *
; * OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
; * LONG FLOAT, LONG INTEGER, TRUNCATE MODES
; *
*****

```

```

†TST271: SCOPE
MOV #MD2016,R5 ; PTR TO TEST DATA SET
JSR PC,@MD2DT ; GO TEST
BR TST272 ;;

```

```

MD2016: ; TEST DATA SET MD2D-16:
.WORD 140452,AN,AN,AN ; INITIAL AC FLOAT NUMBER
.WORD 077600,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD 0,0,0,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 046751,046746 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

FPU ADVANCED INSTR TESTS
D9FP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 104
T272 TEST OF MOD0(2 ACC) INSTR, DATA SET MD20-17

S141
 S142
 S143
 S144
 S145
 S146 020002 000004
 S147 020004 012705 020016
 S148 020010 004737 035434
 S149
 S150 020014 000423
 S151
 S152 020016
 S153 020016 101577 177777 177777
 S154 020024 177777
 S155 020026 101000 000000 000000
 S156 020034 000000
 S157 020036 042377 177777 177777
 S158 020044 177777
 S159 020046 000000 000000 000000
 S160 020054 000000
 S161 020056 047617 147600
 S162 020062 100012
 S163
 S164
 S165
 S166
 S167
 S168
 S169
 S170 020064 000004
 S171 020066 012705 020100
 S172 020072 004737 035434
 S173
 S174 020076 000423
 S175
 S176 020100
 S177 020100 101577 177777 177777
 S178 020106 177777
 S179 020110 101000 000000 000000
 S180 020116 000000
 S181 020120 000000 000000 000000
 S182 020126 000000
 S183 020130 000000 000000 000000
 S184 020136 000000
 S185 020140 045713 045704
 S186 020144 000000
 S187
 S188

```

*****
*TEST 272      TEST OF MOD0(2 ACC) INSTR, DATA SET MD20-17
*              ALL INTERRUPT ENABLES ON
*              LONG FLOAT, SHORT INTEGER, ROUND MODES
*****
TST272: SCOPE
MOV      #MD2017,R5      ; PTR TO TEST DATA SET
JSR      PC,#MD20T      ; GO TEST
BR       TST273        ;;

MD2017: ; TEST DATA SET MD20-17:
.WORD   101577,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD   101000,0,0,0    ; INITIAL MEM FLOAT NUMBER
.WORD   042377,M1,M1,M1 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD   0,0,0,0        ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD   047617,147600  ; FPS: BEFORE AFTER
.WORD   100012         ; FEC AFTER ( 0 = N/A )

*****
*TEST 273      TEST OF MOD0(2 ACC) INSTR, DATA SET MD20-20
*              UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
*              LONG FLOAT, LONG INTEGER, ROUND MODES
*****
TST273: SCOPE
MOV      #MD2020,R5     ; PTR TO TEST DATA SET
JSR      PC,#MD20T     ; GO TEST
BR       TST274        ;;

MD2020: ; TEST DATA SET MD20-20:
.WORD   101577,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD   101000,0,0,0    ; INITIAL MEM FLOAT NUMBER
.WORD   0,0,0,0        ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD   0,0,0,0        ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD   045713,045704  ; FPS: BEFORE AFTER
.WORD   NA              ; FEC AFTER ( 0 = N/A )
  
```


M08

FPU ADVANCED INSTR TESTS
D9FP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 105
T274 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-1

5189			
5190			
5191			
5192			
5193			
5194	020146	000004	
5195	020150	012705	020162
5196	020154	004737	035676
5197			
5198	020160	000413	
5199			
5200	020162		
5201	020162	000000	000000
5202	020166	000000	000000
5203	020172	000000	000000
5204	020176	052525	177777
5205	020202	047513	047504
5206	020206	000000	
5207			
5208			
5209			
5210			
5211			
5212			
5213			
5214	020210	000004	
5215	020212	012705	020224
5216	020216	004737	035676
5217			
5218	020222	000413	
5219			
5220	020224		
5221	020224	000177	177777
5222	020230	077777	177777
5223	020234	000000	000000
5224	020240	052525	177777
5225	020244	047553	047544
5226	020250	000000	
5227			
5228			
5229			
5230			
5231			
5232			
5233			
5234	020252	000004	
5235	020254	012705	020266
5236	020260	004737	035676
5237			
5238	020264	000413	
5239			
5240	020266		
5241	020266	177777	177777
5242	020272	100177	177777
5243	020276	000000	000000
5244	020302	052525	177777

```

*****
; TEST 274 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-1
; * ALL INTERRUPT ENABLES ON
; * SHORT FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

↑TST274: SCOPE
MOV #MDIF1,R5 ; PTR TO TEST DATA SET
JSR PC,#MDIFT ; GO TEST
BR TST275 ;;

```

```

MDIF1: ; TEST DATA SET MDIF-1:
.WORD 0,0 ; INITIAL AC FLOAT NUMBER
.WORD 0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047513,047504 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
; TEST 275 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-2
; * ALL INTERRUPT ENABLES ON
; * SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

↑TST275: SCOPE
MOV #MDIF2,R5 ; PTR TO TEST DATA SET
JSR PC,#MDIFT ; GO TEST
BR TST276 ;;

```

```

MDIF2: ; TEST DATA SET MDIF-2:
.WORD ZXIMP,M1 ; INITIAL AC FLOAT NUMBER
.WORD LGP,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047553,047544 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
; TEST 276 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-3
; * -0 INTERRUPT ENABLE OFF, ALL OTHERS ON
; * SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

↑TST276: SCOPE
MOV #MDIF3,R5 ; PTR TO TEST DATA SET
JSR PC,#MDIFT ; GO TEST
BR TST277 ;;

```

```

MDIF3: ; TEST DATA SET MDIF-3:
.WORD LGN,M1 ; INITIAL AC FLOAT NUMBER
.WORD ZXIMP,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1 ; EXPECTED INTEGER-PART FLOAT RESULT

```

5245 020306 043413 043404
5246 020312 000000

.WORD 043413,043404 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

5247
5248
5249
5250
5251
5252

*TEST 277 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-4
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES

5253
5254 020314 000004
5255 020316 012705 020330
5256 020322 004737 035676

TST277: SCOPE
MOV #MDIF4,R5 ; PTR TO TEST DATA SET
JSR PC,@MDIFT ; GO TEST

BR TST300 ;

5257
5258 020326 000413
5259
5260 020330
5261 020330 177777 177777
5262 020334 100177 177777
5263 020340 177777 177777
5264 020344 052525 177777
5265 020350 047447 147447
5266 020354 100014

MDIF4: ; TEST DATA SET MDIF-4:
.WORD LGN,M1 ; INITIAL AC FLOAT NUMBER
.WORD ZXIMN,M1 ; INITIAL MEM FLOAT NUMBER
.WORD LGN,M1 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047447,147447 ; FPS: BEFORE, AFTER
.WORD 100014 ; FEC AFTER (0 = N/A)

5267
5268
5269
5270
5271
5272

*TEST 300 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-5
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES

5273
5274 020356 000004
5275 020360 012705 020372
5276 020364 004737 035676

TST300: SCOPE
MOV #MDIF5,R5 ; PTR TO TEST DATA SET
JSR PC,@MDIFT ; GO TEST

BR TST301 ;

5277
5278 020370 000413
5279
5280 020372
5281 020372 042177 000000
5282 020376 140200 000000
5283 020402 000000 000000
5284 020406 052525 177777
5285 020412 047553 047544
5286 020416 000000

MDIF5: ; TEST DATA SET MDIF-5:
.WORD 042177,0 ; INITIAL AC FLOAT NUMBER
.WORD FIN,0 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047553,047544 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

5287
5288
5289
5290
5291
5292

*TEST 301 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-6
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES

5293
5294 020420 000004
5295 020422 012705 020434
5296 020426 004737 035676

TST301: SCOPE
MOV #MDIF6,R5 ; PTR TO TEST DATA SET
JSR PC,@MDIFT ; GO TEST

BR TST302 ;

5297
5298 020432 000413
5299
5300 020434

MDIF6: ; TEST DATA SET MDIF-6:

5301	020434	040200	000000
5302	020440	140177	177777
5303	020444	140177	177777
5304	020450	052525	177777
5305	020454	047507	047510
5306	020460	000000	

```
.WORD FIP 0 ; INITIAL AC FLOAT NUMBER
.WORD 140177,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 140177,M1 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047507,047510 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
```

5307			
5308			
5309			
5310			
5311			
5312			
5313			

```
*****
*TEST 302 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-7
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
```

5314	020462	000004	
5315	020464	012705	020476
5316	020470	004737	035676
5317			
5318	020474	000413	
5319			

```
↑ST302: SCOPE
MOV #MDIF7,R5 ; PTR TO TEST DATA SET
JSR PC,#MDIF7 ; GO TEST
BR TST303 ;;
```

5320	020476		
5321	020476	142176	077600
5322	020502	140200	000000
5323	020506	037777	000000
5324	020512	052525	177777
5325	020516	047457	047440
5326	020522	000000	

```
MDIF7: ; TEST DATA SET MDIF-7:
.WORD 142176,077600 ; INITIAL AC FLOAT NUMBER
.WORD FIP,0 ; INITIAL MEM FLOAT NUMBER
.WORD 037777,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047457,047440 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
```

5327			
5328			
5329			
5330			
5331			
5332			
5333			

```
*****
*TEST 303 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-10
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****
```

5334	020524	000004	
5335	020526	012705	020540
5336	020532	004737	035676
5337			
5338	020536	000413	
5339			

```
↑ST303: SCOPE
MOV #MDIF10,R5 ; PTR TO TEST DATA SET
JSR PC,#MDIF10 ; GO TEST
BR TST304 ;;
```

5340	020540		
5341	020540	042177	100000
5342	020544	040200	000000
5343	020550	040000	000000
5344	020554	052525	177777
5345	020560	047417	047400
5346	020564	000000	

```
MDIF10: ; TEST DATA SET MDIF-10:
.WORD 042177,M0 ; INITIAL AC FLOAT NUMBER
.WORD FIP,0 ; INITIAL MEM FLOAT NUMBER
.WORD 040000,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047417,047400 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
```

5347			
5348			
5349			
5350			
5351			
5352			
5353			

```
*****
*TEST 304 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-11
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****
```

5354	020566	000004	
5355	020570	012705	020602
5356	020574	004737	035676

```
↑ST304: SCOPE
MOV #MDIF11,R5 ; PTR TO TEST DATA SET
JSR PC,#MDIF11 ; GO TEST
```

5357
5358 020600 000413
5359
5360 020602
5361 020602 140200 000000
5362 020606 040377 177777
5363 020612 140177 177776
5364 020616 052525 177777
5365 020622 047547 047550
5366 020626 000000
5367
5368
5369
5370
5371
5372
5373
5374 020630 000004
5375 020632 012705 020644
5376 020636 004737 035676
5377
5378 020642 000413
5379
5380 020644
5381 020644 060452 125252
5382 020650 021700 000000
5383 020654 040177 177400
5384 020660 052525 177777
5385 020664 047517 04.500
5386 020670 000000
5387
5388
5389
5390
5391
5392
5393
5394 020672 000004
5395 020674 012705 020706
5396 020700 004737 035676
5397
5398 020704 000413
5399
5400 020706
5401 020706 041000 000001
5402 020712 141377 177776
5403 020716 140177 177777
5404 020722 052525 177777
5405 020726 047547 047550
5406 020732 000000
5407
5408
5409
5410
5411
5412

```
BR TST305 ;;
MDIF11: ; TEST DATA SET MDIF-11:
.WORD FIN,0 ; INITIAL AC FLOAT NUMBER
.WORD 040377,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 140177,M2 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047547,047550 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
```

```
*****
*TEST 305 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-12
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****
```

```
↑TST305: SCOPE
MOV #MDIF12,R5 ; PTR TO TEST DATA SET
JSR PC,#MDIFT ; GO TEST
BR TST306 ;;
```

```
MDIF12: ; TEST DATA SET MDIF-12:
.WORD 060452,ALTN ; INITIAL AC FLOAT NUMBER
.WORD 021700,0 ; INITIAL MEM FLOAT NUMBER
.WORD 040177,UB ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047517,047500 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
```

```
*****
*TEST 306 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-13
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****
```

```
↑TST306: SCOPE
MOV #MDIF13,R5 ; PTR TO TEST DATA SET
JSR PC,#MDIFT ; GO TEST
BR TST307 ;;
```

```
MDIF13: ; TEST DATA SET MDIF-13:
.WORD 041000,000001 ; INITIAL AC FLOAT NUMBER
.WORD 141377,M2 ; INITIAL MEM FLOAT NUMBER
.WORD 140177,M1 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047547,047550 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
```

```
*****
*TEST 307 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-14
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****
```

```

5413
5414 020734 000004
5415 020736 012705 020750
5416 020742 004737 035676
5417
5418 020746 000413
5419
5420 020750
5421 020750 041000 000001
5422 020754 141377 177776
5423 020760 140200 000000
5424 020764 052525 177777
5425 020770 047507 047510
5426 020774 000000
5427
5428
5429
5430
5431
5432
5433
5434 020776 000004
5435 021000 012705 021012
5436 021004 004737 035676
5437
5438 021010 000413
5439
5440 021012
5441 021012 077600 000000
5442 021016 040452 125252
5443 021022 000000 000000
5444 021026 052525 177777
5445 021032 047411 147406
5446 021036 100010
5447
5448
5449
5450
5451
5452
5453
5454 021040 000004
5455 021042 012705 021054
5456 021046 004737 035676
5457
5458 021052 000413
5459
5460 021054
5461 021054 077600 000000
5462 021060 040452 125252
5463 021064 000000 000000
5464 021070 052525 177777
5465 021074 046411 046406
5466 021100 000000
5467
5468

```

```

*****
↑ST307: SCOPE
MOV #MDIF14,R5 ; PTR TO TEST DATA SET
JSR PC,2#MD1FT ; GO TEST
BR TST310 ;;

```

```

MDIF14: ; TEST DATA SET MDIF-14:
.WORD 041000,000001 ; INITIAL AC FLOAT NUMBER
.WORD 141377,M2 ; INITIAL MEM FLOAT NUMBER
.WORD 140200,000000 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047507,047510 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 310 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-15
*
* ALL INTERRUPT ENABLES ON
*
* SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

↑ST310: SCOPE
MOV #MDIF15,R5 ; PTR TO TEST DATA SET
JSR PC,2#MD1FT ; GO TEST
BR TST311 ;;

```

```

MDIF15: ; TEST DATA SET MDIF-15:
.WORD 077600,0 ; INITIAL AC FLOAT NUMBER
.WORD 040452,ALTN ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047411,147406 ; FPS: BEFORE, AFTER
.WORD 100010 ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 311 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-16
*
* OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
*
* SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

↑ST311: SCOPE
MOV #MDIF16,R5 ; PTR TO TEST DATA SET
JSR PC,2#MD1FT ; GO TEST
BR TST312 ;;

```

```

MDIF16: ; TEST DATA SET MDIF-16:
.WORD 077600,0 ; INITIAL AC FLOAT NUMBER
.WORD 040452,ALTN ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 046411,046406 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

5469
5470
5471
5472
5473
5474 021102 000004
5475 021104 012705 021116
5476 021110 004737 035676
5477
5478 021114 000413
5479
5480 021116
5481 021116 001577 177777
5482 021122 101000 000000
5483 021126 142377 177777
5484 021132 052525 177777
5485 021136 047547 147550
5486 021142 100012
5487
5488
5489
5490
5491
5492
5493
5494 021144 000004
5495 021146 012705 021160
5496 021152 004737 035676
5497
5498 021156 000413
5499
5500 021160
5501 021160 001577 177777
5502 021164 101000 000000
5503 021170 000000 000000
5504 021174 052525 177777
5505 021200 045553 045544
5506 021204 000000
5507
5508
5509

```

*****
*TEST 312      TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-17
*              ALL INTERRUPT ENABLES ON
*              SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

†ST312: SCOPE
MOV      #MDIF17,R5      ; PTR TO TEST DATA SET
JSR      PC,#MD1FT      ; GO TEST
BR       TST313         ;;

```

```

MDIF17: ; TEST DATA SET MDIF-17:
.WORD   001577,M1      ; INITIAL AC FLOAT NUMBER
.WORD   101000,0       ; INITIAL MEM FLOAT NUMBER
.WORD   142377,M1      ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD   AP,M1          ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD   047547,147550 ; FPS: BEFORE AFTER
.WORD   100012         ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 313      TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-20
*              UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
*              SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

†ST313: SCOPE
MOV      #MDIF20,R5     ; PTR TO TEST DATA SET
JSR      PC,#MD1FT     ; GO TEST
BR       TST314       ;;

```

```

MDIF20: ; TEST DATA SET MDIF-20:
.WORD   001577,M1      ; INITIAL AC FLOAT NUMBER
.WORD   101000,0       ; INITIAL MEM FLOAT NUMBER
.WORD   0,0           ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD   AP,M1          ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD   045553,045544 ; FPS: BEFORE AFTER
.WORD   NA             ; FEC AFTER ( 0 = N/A )

```

```

5510 ;*****
5511 ;*TEST 314 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-1
5512 ;*
5513 ;* ALL INTERRUPT ENABLES ON
5514 ;* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
5515 ;*****
5515 021206 000004
5516 021210 012705 021222
5517 021214 004737 036100
5518
5519 021220 000423
5520
5521 021222
5522 021222 000000 000000 000000
5523 021230 000000
5524 021232 000000 000000 000000
5525 021240 000000
5526 021242 000000 000000 000000
5527 021250 000000
5528 021252 052525 177777 125252
5529 021260 000000
5530 021262 047653 047644
5531 021266 000000
5532
5533
5534 ;*****
5535 ;*TEST 315 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-2
5536 ;*
5537 ;* ALL INTERRUPT ENABLES ON
5538 ;* LONG FLOAT, LONG INTEGER, ROUND MODES
5539 ;*****
5539 021270 000004
5540 021272 012705 021304
5541 021276 004737 036100
5542
5543 021302 000423
5544
5545 021304
5546 021304 000177 177777 177777
5547 021312 177777
5548 021314 177777 177777 177777
5549 021322 177777
5550 021324 000000 000000 000000
5551 021332 000000
5552 021334 052525 177777 125252
5553 021342 000000
5554 021344 047713 047704
5555 021350 000000
5556
5557
5558 ;*****
5559 ;*TEST 316 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-3
5560 ;*
5561 ;* -0 INTERRUPT ENABLE OFF, ALL OTHERS ON
5562 ;* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
5563 ;*****
5563 021352 000004
5564 021354 012705 021366
5565 021360 004737 036100

```

```

;*****
;*TEST 314 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-1
;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
;*****
†T314: SCOPE
MOV #MD1D1,R5 ; PTR TO TEST DATA SET
JSR PC,@MD1DT ; GO TEST

BR TST315 ;;

MD1D1: ; TEST DATA SET MD1D-1:
.WORD 0,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD 0,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1,AN,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047653,047644 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

;*****
;*TEST 315 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-2
;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, LONG INTEGER, ROUND MODES
;*****
†T315: SCOPE
MOV #MD1D2,R5 ; PTR TO TEST DATA SET
JSR PC,@MD1DT ; GO TEST

BR TST316 ;;

MD1D2: ; TEST DATA SET MD1D-2:
.WORD ZXIMP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD LGN,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1,AN,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047713,047704 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

;*****
;*TEST 316 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-3
;*
;* -0 INTERRUPT ENABLE OFF, ALL OTHERS ON
;* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
;*****
†T316: SCOPE
MOV #MD1D3,R5 ; PTR TO TEST DATA SET
JSR PC,@MD1DT ; GO TEST

```

```

5566
5567 021364 000423          BR      TST317          ;;
5568
5569 021366          MD103: ; TEST DATA SET MD10-3:
5570 021366 077777 177777 177777 .WORD  LGP,M1,M1,M1  ; INITIAL AC FLOAT NUMBER
5571 021374 177777          .WORD  ZX1MN,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
5572 021376 100177 177777 177777 .WORD  0,0,0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
5573 021404 177777          .WORD  AP,M1,AN,0    ; EXPECTED INTEGER-PART FLOAT RESULT
5574 021406 000000 000000 000000 .WORD  043653,043644 ; FPS: BEFORE AFTER
5575 021414 000000          .WORD  NA           ; FEC AFTER ( 0 = N/A )
5576 021416 052525 177777 125252
5577 021424 000000
5578 021426 043653 043644
5579 021432 000000
5580
5581

```

```

*****
; TEST 317 TEST OF MODD(1 ACC) INSTR, DATA SET MD10-4
; * ALL INTERRUPT ENABLES ON
; * LONG FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

5582
5583
5584
5585
5586
5587 021434 000004          †ST317: SCOPE
5588 021436 012705 021450      MOV     #MD104,R5      ; PTR TO TEST DATA SET
5589 021442 004737 036100      JSR     PC,#MD10T     ; GO TEST
5590
5591 021446 000423          BR      TST320          ;;
5592
5593 021450          MD104: ; TEST DATA SET MD10-4:
5594 021450 077777 177777 177777 .WORD  LGP,M1,M1,M1  ; INITIAL AC FLOAT NUMBER
5595 021456 177777          .WORD  ZX1MN,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
5596 021460 100177 177777 177777 .WORD  LGP,M1,M1,M1  ; EXPECTED FRACTION-PART FLOAT RESULT
5597 021466 177777          .WORD  AP,M1,AN,0    ; EXPECTED INTEGER-PART FLOAT RESULT
5598 021470 077777 177777 177777 .WORD  047713,147713 ; FPS: BEFORE AFTER
5599 021476 177777          .WORD  100014       ; FEC AFTER ( 0 = N/A )
5600 021500 052525 177777 125252
5601 021506 000000
5602 021510 047713 147713
5603 021514 100014
5604
5605

```

```

*****
; TEST 320 TEST OF MODD(1 ACC) INSTR, DATA SET MD10-5
; * ALL INTERRUPT ENABLES ON
; * LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

5606
5607
5608
5609
5610
5611 021516 000004          †ST320: SCOPE
5612 021520 012705 021532      MOV     #MD105,R5      ; PTR TO TEST DATA SET
5613 021524 004737 036100      JSR     PC,#MD10T     ; GO TEST
5614
5615 021530 000423          BR      TST321          ;;
5616
5617 021532          MD105: ; TEST DATA SET MD10-5:
5618 021532 042177 000000 000000 .WORD  042177,0,0,0  ; INITIAL AC FLOAT NUMBER
5619 021540 000000          .WORD  F1P,0,0,0    ; INITIAL MEM FLOAT NUMBER
5620 021542 040200 000000 000000
5621 021550 000000

```


FPU ADVANCED INSTR TESTS
DQFP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 113
T320 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-5

5622	021552	000000	000000	000000	.WORD	0,0,0,0	; EXPECTED FRACTION-PART FLOAT RESULT
5623	021560	000000					
5624	021562	052525	177777	125252	.WORD	AP,M1,AN,0	; EXPECTED INTEGER-PART FLOAT RESULT
5625	021570	000000					
5626	021572	047613	047604		.WORD	047613,047604	; FPS: BEFORE, AFTER
5627	021576	000000			.WORD	NA	; FEC AFTER (0 = N/A)

5628							
5629							
5630							

```

*****
;TEST 321 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-6
;
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

†ST321: SCOPE
MOV #MD1D6,R5 ; PTR TO TEST DATA SET
JSR PC,@#MD1DT ; GO TEST
BR TST322 ;;

```

5635	021600	000004			MD1D6: ; TEST DATA SET MD1D-6:		
5636	021602	012705	021614		.WORD	FIN,0,0,0	; INITIAL AC FLOAT NUMBER
5637	021606	004737	036100		.WORD	040177,M1,M1,M1	; INITIAL MEM FLOAT NUMBER
5638							
5639	021612	000423			.WORD	140177,M1,M1,M1	; EXPECTED FRACTION-PART FLOAT RESULT
5640					.WORD	AP,M1,AN,0	; EXPECTED INTEGER-PART FLOAT RESULT
5641	021614				.WORD	047747,047750	; FPS: BEFORE, AFTER
5642	021614	140200	000000	000000	.WORD	NA	; FEC AFTER (0 = N/A)
5643	021622	000000					
5644	021624	040177	177777	177777			
5645	021632	177777					
5646	021634	140177	177777	177777			
5647	021642	177777					
5648	021644	052525	177777	125252			
5649	021652	000000					
5650	021654	047747	047750				
5651	021660	000000					

5652							
5653							
5654							

```

*****
;TEST 322 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-7
;
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

†ST322: SCOPE
MOV #MD1D7,R5 ; PTR TO TEST DATA SET
JSR PC,@#MD1DT ; GO TEST
BR TST323 ;;

```

5659	021662	000004			MD1D7: ; TEST DATA SET MD1D-7:		
5660	021664	012705	021676		.WORD	042176,077600,0,0	; INITIAL AC FLOAT NUMBER
5661	021670	004737	036100		.WORD	FIN,0,0,0	; INITIAL MEM FLOAT NUMBER
5662					.WORD	137777,0,0,0	; EXPECTED FRACTION-PART FLOAT RESULT
5663	021674	000423			.WORD	AP,M1,AN,0	; EXPECTED INTEGER-PART FLOAT RESULT
5664					.WORD	047607,047610	; FPS: BEFORE, AFTER
5665	021676	042176	077600	000000	.WORD	NA	; FEC AFTER (0 = N/A)
5666	021704	000000					
5667	021706	140200	000000	000000			
5668	021714	000000					
5669	021716	137777	000000	000000			
5670	021724	000000					
5671	021726	052525	177777	125252			
5672	021734	000000					
5673	021736	047607	047610				
5674	021742	000000					

5675							
5676							
5677							

FPU ADVANCED INSTR TESTS
DQFP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 114
T323 TEST OF MODD(1 ACC) INSTR, DATA SET MD10-10

5678				
5679				
5680				
5681				
5682				
5683	021744	000004		
5684	021746	012705	021760	
5685	021752	004737	036100	
5686				
5687	021756	000423		
5688				
5689	021760			
5690	021760	142177	100000	000000
5691	021766	000000		
5692	021770	040200	000000	000000
5693	021776	000000		
5694	022000	140000	000000	000000
5695	022006	000000		
5696	022010	052525	177777	125252
5697	022016	000000		
5698	022020	047747	047750	
5699	022024	000000		

```

*****
: TEST 323      TEST OF MODD(1 ACC) INSTR, DATA SET MD10-10
: *
: *            ALL INTERRUPT ENABLES ON
: *            LONG FLOAT, LONG INTEGER, TRUNCATE MODES
: *****

```

```

†ST323: SCOPE
      MOV      #MD1010,R5      ; PTR TO TEST DATA SET
      JSR      PC,@#MD10T     ; GO TEST
      BR       TST324        ;;

```

```

MD1010: ; TEST DATA SET MD10-10:
      .WORD   142177,MD,0,0   ; INITIAL AC FLOAT NUMBER
      .WORD   F1P,0,0,0      ; INITIAL MEM FLOAT NUMBER
      .WORD   140000,0,0,0   ; EXPECTED FRACTION-PART FLOAT RESULT
      .WORD   AP,M1,AN,0     ; EXPECTED INTEGER-PART FLOAT RESULT
      .WORD   047747,047750  ; FPS: BEFORE AFTER
      .WORD   NA              ; FEC AFTER ( 0 = N/A )

```

5700				
5701				
5702				
5703				
5704				
5705				
5706				
5707	022026	000004		
5708	022030	012705	022042	
5709	022034	004737	036100	
5710				
5711	022040	000423		
5712				
5713	022042			
5714	022042	140200	000000	000000
5715	022050	000000		
5716	022052	140377	177777	177777
5717	022060	177777		
5718	022062	040177	177777	177777
5719	022070	177776		
5720	022072	052525	177777	125252
5721	022100	000000		
5722	022102	047617	047600	
5723	022106	000000		

```

*****
: TEST 324      TEST OF MODD(1 ACC) INSTR, DATA SET MD10-11
: *
: *            ALL INTERRUPT ENABLES ON
: *            LONG FLOAT, SHORT INTEGER, ROUND MODES
: *****

```

```

†ST324: SCOPE
      MOV      #MD1011,R5     ; PTR TO TEST DATA SET
      JSR      PC,@#MD10T     ; GO TEST
      BR       TST325        ;;

```

```

MD1011: ; TEST DATA SET MD10-11:
      .WORD   F1N,0,0,0      ; INITIAL AC FLOAT NUMBER
      .WORD   140377,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
      .WORD   040177,M1,M1,M2 ; EXPECTED FRACTION-PART FLOAT RESULT
      .WORD   AP,M1,AN,0     ; EXPECTED INTEGER-PART FLOAT RESULT
      .WORD   047617,047600  ; FPS: BEFORE AFTER
      .WORD   NA              ; FEC AFTER ( 0 = N/A )

```

5724				
5725				
5726				
5727				
5728				
5729				
5730				
5731	022110	000004		
5732	022112	012705	022124	
5733	022116	004737	036100	

```

*****
: TEST 325      TEST OF MODD(1 ACC) INSTR, DATA SET MD10-12
: *
: *            ALL INTERRUPT ENABLES ON
: *            LONG FLOAT, LONG INTEGER, TRUNCATE MODES
: *****

```

```

†ST325: SCOPE
      MOV      #MD1012,R5     ; PTR TO TEST DATA SET
      JSR      PC,@#MD10T     ; GO TEST

```

```

5734
5735 022122 000423          BR      TST326          ;;
5736
5737 022124          MD1D12: ; TEST DATA SET MD1D-12:
5738 022124 167452 125252 125252 .WORD 167452,AN,AN,AN ; INITIAL AC FLOAT NUMBER
5739 022132 125252
5740 022134 112700 000000 000000 .WORD 112700,0,0,0 ; INITIAL MEM FLOAT NUMBER
5741 022142 000000
5742 022144 040177 177777 177777 .WORD 040177,M1,M1,UB ; EXPECTED FRACTION-PART FLOAT RESULT
5743 022152 177400
5744 022154 052525 177777 125252 .WORD AP,M1,AN,0 ; EXPECTED INTEGER-PART FLOAT RESULT
5745 022162 000000
5746 022164 047757 047740 .WORD 047757,047740 ; FPS: BEFORE, AFTER
5747 022170 000000 .WORD NA ; FEC AFTER ( 0 = N/A )
5748
5749

```

```

*****
*TEST 326 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-13
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

5750
5751
5752
5753
5754
5755 022172 000004
5756 022174 012705 022206
5757 022200 004737 036100
5758
5759 022204 000423          BR      TST327          ;;
5760

```

```

†TST326: SCOPE
MOV      #MD1D13,R5 ; PTR TO TEST DATA SET
JSR      PC,#MD1DT ; GO TEST

```

```

5761 022206          MD1D13: ; TEST DATA SET MD1D-13:
5762 022206 041000 000000 000000 .WORD 041000,0,0,1 ; INITIAL AC FLOAT NUMBER
5763 022214 000001
5764 022216 037577 177777 177777 .WORD 037577,M1,M1,M2 ; INITIAL MEM FLOAT NUMBER
5765 022224 177776
5766 022226 040177 177777 177777 .WORD 040177,M1,M1,M1 ; EXPECTED FRACTION-PART FLOAT RESULT
5767 022234 177777
5768 022236 052525 177777 125252 .WORD AP,M1,AN,0 ; EXPECTED INTEGER-PART FLOAT RESULT
5769 022244 000000
5770 022246 047657 047640 .WORD 047657,047640 ; FPS: BEFORE, AFTER
5771 022252 000000 .WORD NA ; FEC AFTER ( 0 = N/A )
5772
5773

```

```

*****
*TEST 327 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-14
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

5774
5775
5776
5777
5778
5779 022254 000004
5780 022256 012705 022270
5781 022262 004737 036100
5782
5783 022266 000423          BR      TST330          ;;
5784

```

```

†TST327: SCOPE
MOV      #MD1D14,R5 ; PTR TO TEST DATA SET
JSR      PC,#MD1DT ; GO TEST

```

```

5785 022270          MD1D14: ; TEST DATA SET MD1D-14:
5786 022270 041000 000000 000000 .WORD 041000,0,0,1 ; INITIAL AC FLOAT NUMBER
5787 022276 000001
5788 022300 037577 177777 177777 .WORD 037577,M1,M1,M2 ; INITIAL MEM FLOAT NUMBER
5789 02.306 177776

```

FPU ADVANCED INSTR TESTS
DFFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 116
T327 TEST OF MODD(1 ACC) INSTR, DATA SET MD10-14

5790	022310	040200	000000	000000	.WORD	040200,0,0,0	; EXPECTED FRACTION-PART FLOAT RESULT
5791	022316	000000					
5792	022320	052525	177777	125252	.WORD	AP,M1,AN,0	; EXPECTED INTEGER-PART FLOAT RESULT
5793	022326	000000					
5794	022330	047617	047600		.WORD	047617,047600	; FPS: BEFORE, AFTER
5795	022334	000000			.WORD	NA	; FEC AFTER (0 = N/A)

```

5796
5797
5798
5799
5800
5801
5802
5803
5804
5805
5806
5807
5808
5809
5810
5811
5812
5813
5814
5815
5816
5817
5818
5819
5820
5821
5822
5823
5824
5825
5826
5827
5828
5829
5830
5831
5832
5833
5834
5835
5836
5837
5838
5839
5840
5841
5842
5843
5844
5845

```

```

*****
*TEST 330 TEST OF MODD(1 ACC) INSTR, DATA SET MD10-15
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST330: SCOPE
MOV #MD1015,R5 ; PTR TO TEST DATA SET
JSR PC,#MD10T ; GO TEST
BR TST331 ;;

```

```

MD1015: ; TEST DATA SET MD10-15:
.WORD 142452,ALTN,ALTN,ALTN ; INITIAL AC FLOAT NUMBER
.WORD 077600,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1,AN,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047651,147646 ; FPS: BEFORE, AFTER
.WORD 100010 ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 331 TEST OF MODD(1 ACC) INSTR, DATA SET MD10-16
*
* OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****
TST331: SCOPE
MOV #MD1016,R5 ; PTR TO TEST DATA SET
JSR PC,#MD10T ; GO TEST
BR TST332 ;;

```

```

MD1016: ; TEST DATA SET MD10-16:
.WORD 142452,ALTN,ALTN,ALTN ; INITIAL AC FLOAT NUMBER
.WORD 077600,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1,AN,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 046751,046746 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

5846
5847
5848
5849
5850
5851 022502 000004
5852 022504 012705 022516
5853 022510 004737 036100
5854
5855 022514 000423
5856
5857 022516
5858 022516 101577 177777 177777
5859 022524 177777
5860 022526 101000 000000 000000
5861 022534 000000
5862 022536 042377 177777 177777
5863 022544 177777
5864 022546 052525 177777 125252
5865 022554 000000
5866 022556 047617 147600
5867 022562 100012
5868
5869
5870
5871
5872
5873
5874
5875 022564 000004
5876 022566 012705 022600
5877 022572 004737 036100
5878
5879 022576 000423
5880
5881 022600
5882 022600 101577 177777 177777
5883 022606 177777
5884 022610 101000 000000 000000
5885 022616 000000
5886 022620 000000 000000 000000
5887 022626 000000
5888 022630 052525 177777 125252
5889 022636 000000
5890 022640 045713 045704
5891 022644 000000
5892
5893

```

```

*****
*TEST 332 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-17
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****
TST332: SCOPE
MOV #MD1D17,R5 ; PTR TO TEST DATA SET
JSR PC,#MD1DT ; GO TEST
BR TST333 ;;

MD1D17: ; TEST DATA SET MD1D-17:
.WORD 101577,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD 101000,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 042377,M1,M1,M1 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1,AN,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047617,147600 ; FPS: BEFORE AFTER
.WORD 100012 ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 333 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-20
* UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
TST333: SCOPE
MOV #MD1D20,R5 ; PTR TO TEST DATA SET
JSR PC,#MD1DT ; GO TEST
BR TST334 ;;

MD1D20: ; TEST DATA SET MD1D-20:
.WORD 101577,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD 101000,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1,AN,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 045713,045704 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

5894
5895
5896
5897
5898
5899 022646 000004
5900 022650 012705 0226b2
5901 022654 004737 036342
5902
5903 022660 000411
5904
5905 022662
5906 022662 000000 000000 000000
5907 022670 000000
5908 022672 000000 000000
5909 022676 047413 047404
5910 022702 000000
5911
5912
5913
5914
5915
5916
5917
5918 022704 000004
5919 022706 012705 022720
5920 022712 004737 036342
5921
5922 022716 000411
5923
5924 022720
5925 022720 100177 177777 177777
5926 022726 177777
5927 022730 052525 177777
5928 022734 047503 147514
5929 022740 100014
5930
5931
5932
5933
5934
5935
5936
5937 022742 000004
5938 022744 012705 022756
5939 022750 004737 036342
5940
5941 022754 000411
5942
5943 022756
5944 022756 000177 177777 177777
5945 022764 177777
5946 022766 000000 000000
5947 022772 047453 047444
5948 022776 000000
5949

```

```

*****
: TEST 334 TEST OF LCDCF INSTR, DATA SET LCDCF-1
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, SHORT INTEGER, ROUND MODES
: *****
†ST334: SCOPE
MOV #LCDCF1,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCDCF1 ; GO TEST
BR TST335 ;;

LCDCF1: ; TEST DATA SET LCDCF-1:
.WORD 0,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD 047413,047404 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
: TEST 335 TEST OF LCDCF INSTR, DATA SET LCDCF-2
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, LONG INTEGER, ROUND MODES
: *****
†ST335: SCOPE
MOV #LCDCF2,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCDCF2 ; GO TEST
BR TST336 ;;

LCDCF2: ; TEST DATA SET LCDCF-2:
.WORD ZX1MN,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD ALTP,M1 ; EXPECTED FLOAT RESULT
.WORD 047503,147514 ; FPS: BEFORE, AFTER
.WORD 100014 ; FEC AFTER ( 0 = N/A )

*****
: TEST 336 TEST OF LCDCF INSTR, DATA SET LCDCF-3
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
: *****
†ST336: SCOPE
MOV #LCDCF3,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCDCF3 ; GO TEST
BR TST337 ;;

LCDCF3: ; TEST DATA SET LCDCF-3:
.WORD ZX1MP,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD 047453,047444 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

5950
5951
5952
5953
5954
5955
5956 023000 000004
5957 023002 012705 023014
5958 023006 004737 036342
5959
5960 023012 000411
5961
5962 023014
5963 023014 040200 000000 000000
5964 023022 000000
5965 023024 040200 000000
5966 023030 047557 047540
5967 023034 000000
5968
5969
5970
5971
5972
5973
5974
5975 023036 000004
5976 023040 012705 023052
5977 023044 004737 036342
5978
5979 023050 000411
5980
5981 023052
5982 023052 140200 000000 100000
5983 023060 000000
5984 023062 140200 000001
5985 023066 047407 047410
5986 023072 000000
5987
5988
5989
5990
5991
5992
5993
5994 023074 000004
5995 023076 012705 023110
5996 023102 004737 036342
5997
5998 023106 000411
5999
6000 023110
6001 023110 140200 000000 100000
6002 023116 000000
6003 023120 140200 000000
6004 023124 047447 047450
6005 023130 000000

```

```

*****
*TEST 337 TEST OF LCDF INSTR, DATA SET LCDF-4
*
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****
†T337: SCOPE
MOV #LCDF4,R5 ; PTR TO TEST DATA SET
JSR PC,#LCDF4 ; GO TEST
BR TST340 ;;
LCDF4: ; TEST DATA SET LCDF-4:
.WORD F1P,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD F1P,0 ; EXPECTED FLOAT RESULT
.WORD 047557,047540 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
*****
*TEST 340 TEST OF LCDF INSTR, DATA SET LCDF-5
*
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****
†T340: SCOPE
MOV #LCDF5,R5 ; PTR TO TEST DATA SET
JSR PC,#LCDF5 ; GO TEST
BR TST341 ;;
LCDF5: ; TEST DATA SET LCDF-5:
.WORD F1N,0,M0,0 ; INITIAL MEM FLOAT NUMBER
.WORD F1N,1 ; EXPECTED FLOAT RESULT
.WORD 047407,047410 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
*****
*TEST 341 TEST OF LCDF INSTR, DATA SET LCDF-6
*
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
†T341: SCOPE
MOV #LCDF6,R5 ; PTR TO TEST DATA SET
JSR PC,#LCDF6 ; GO TEST
BR TST342 ;;
LCDF6: ; TEST DATA SET LCDF-6:
.WORD F1N,0,M0,0 ; INITIAL MEM FLOAT NUMBER
.WORD F1N,0 ; EXPECTED FLOAT RESULT
.WORD 047447,047450 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

B10

FPU ADVANCED INSTR TESTS
DAPPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 120
T341 TEST OF LCDF INSTR, DATA SET LCDF-6

6006
6007
6008
6009
6010
6011
6012
6013 023132 000004
6014 023134 012705 023146
6015 023140 004737 036342
6016
6017 023144 000411
6018
6019 023146
6020 023146 077777 177777 177777
6021 023154 177777
6022 023156 000000 000000
6023 023162 047511 147506
6024 023166 100010
6025
6026
6027
6028
6029
6030
6031
6032 023170 000004
6033 023172 012705 023204
6034 023176 004737 036342
6035
6036 023202 000411
6037
6038 023204
6039 023204 077777 177777 177777
6040 023212 177777
6041 023214 077777 177777
6042 023220 047557 047540
6043 023224 000000
6044
6045
6046
6047
6048
6049
6050
6051 023226 000004
6052 023230 012705 023242
6053 023234 004737 036342
6054
6055 023240 000411
6056
6057 023242
6058 023242 121177 177777 100000
6059 023250 000000
6060 023252 121200 000000
6061 023256 047407 047410

```

*****
*TEST 342 TEST OF LCDF INSTR, DATA SET LCDF-7
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

†T342: SCOPE
MOV #LCDF7,R5 ; PTR TO TEST DATA SET
JSR PC,#LCDF7 ; GO TEST
BR TST343 ;;

```

```

LCDF7: ; TEST DATA SET LCDF-7:
.WORD LGP,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD 047511,147506 ; FPS: BEFORE, AFTER
.WORD 100010 ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 343 TEST OF LCDF INSTR, DATA SET LCDF-10
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

†T343: SCOPE
MOV #LCDF10,R5 ; PTR TO TEST DATA SET
JSR PC,#LCDF10 ; GO TEST
BR TST344 ;;

```

```

LCDF10: ; TEST DATA SET LCDF-10:
.WORD LGP,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD LGP,M1 ; EXPECTED FLOAT RESULT
.WORD 047557,047540 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 344 TEST OF LCDF INSTR, DATA SET LCDF-11
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

†T344: SCOPE
MOV #LCDF11,R5 ; PTR TO TEST DATA SET
JSR PC,#LCDF11 ; GO TEST
BR TST345 ;;

```

```

LCDF11: ; TEST DATA SET LCDF-11:
.WORD 121177,M1,M0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 121200,0 ; EXPECTED FLOAT RESULT
.WORD 047407,047410 ; FPS: BEFORE, AFTER

```


6062 023262 000000
6063
6064
6065
6066
6067
6068
6069
6070 023264 000004
6071 023266 012705 023300
6072 023272 004737 036342
6073
6074 023276 000411
6075
6076 023300
6077 023300 121177 177777 100000
6078 023306 000000
6079 023310 121177 177777
6080 023314 047447 047450
6081 023320 000000
6082
6083
6084
6085
6086
6087
6088
6089 023322 000004
6090 023324 012705 023336
6091 023330 004737 036342
6092
6093 023334 000411
6094
6095 023336
6096 023336 040200 000000 077777
6097 023344 177777
6098 023346 040200 000000
6099 023352 047517 047500
6100 023356 000000
6101
6102
6103
6104
6105
6106
6107
6108 023360 000004
6109 023362 012705 023374
6110 023366 004737 036342
6111
6112 023372 000411
6113
6114 023374
6115 023374 040200 000000 077777
6116 023402 177777
6117 023404 040200 000000

.WORD NA ; FEC AFTER (0 = N/A)

```

*****
; *TEST 345 TEST OF LCDF INSTR, DATA SET LCDF-12
; * ALL INTERRUPT ENABLES ON
; * SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

†ST345: SCOPE
MOV #LCDF12,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCDF12 ; GO TEST
BR TST346 ;;

```

```

LCDF12: ; TEST DATA SET LCDF-12:
.WORD 121177,M1,MO,0 ; INITIAL MEM FLOAT NUMBER
.WORD 121177,M1 ; EXPECTED FLOAT RESULT
.WORD 047447,047450 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
; *TEST 346 TEST OF LCDF INSTR, DATA SET LCDF-13
; * ALL INTERRUPT ENABLES ON
; * SHORT FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

†ST346: SCOPE
MOV #LCDF13,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCDF13 ; GO TEST
BR TST347 ;;

```

```

LCDF13: ; TEST DATA SET LCDF-13:
.WORD F1P,0,LGP,M1 ; INITIAL MEM FLOAT NUMBER
.WORD F1P,0 ; EXPECTED FLOAT RESULT
.WORD 047517,047500 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
; *TEST 347 TEST OF LCDF INSTR, DATA SET LCDF-14
; * ALL INTERRUPT ENABLES ON
; * SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

†ST347: SCOPE
MOV #LCDF14,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCDF14 ; GO TEST
BR TST350 ;;

```

```

LCDF14: ; TEST DATA SET LCDF-14:
.WORD F1P,0,LGP,M1 ; INITIAL MEM FLOAT NUMBER
.WORD F1P,0 ; EXPECTED FLOAT RESULT

```

6118 023410 047557 047540
6119 023414 000000

.WORD 047557,047540 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

6120
6121
6122
6123
6124
6125
6126

: *TEST 350 TEST OF LCDF INSTR, DATA SET LCDF-15
: * OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
: * SHORT FLOAT, LONG INTEGER, ROUND MODES
: *****

6127 023416 000004
6128 023420 012705 023432
6129 023424 004737 036342

↑TST350: SCOPE
MOV #LCDF15,R5 ; PTR TO TEST DATA SET
JSR PC,2#LCDF15 ; GO TEST

6130
6131 023430 000411

BR TST351 ;;

6132
6133 023432
6134 023432 177777 177777 100000

LCDF15: ; TEST DATA SET LCDF-15:
.WORD LGN,M1,M0,0 ; INITIAL MEM FLOAT NUMBER

6135 023440 000000
6136 023442 000000 000000
6137 023446 046511 046506
6138 023452 000000

.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD 046511,046506 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

6139
6140
6141

: *TEST 351 TEST OF LCDF INSTR, DATA SET LCDF-16
: * -O INTERRUPT ENABLE OFF, ALL OTHERS ON
: * SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
: *****

6142
6143
6144
6145
6146 023454 000004
6147 023456 012705 023470
6148 023462 004737 036342

↑TST351: SCOPE
MOV #LCDF16,R5 ; PTR TO TEST DATA SET
JSR PC,2#LCDF16 ; GO TEST

6149
6150 023466 000411

BR TST352 ;;

6151
6152 023470
6153 023470 100000 177777 177777

LCDF16: ; TEST DATA SET LCDF-16:
.WORD M0,M1,M1,0 ; INITIAL MEM FLOAT NUMBER

6154 023476 000000
6155 023500 000000 000000
6156 023504 043453 043444
6157 023510 000000

.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD 043453,043444 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

6158
6159
6160

E10

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 123
T352 TEST OF LDCFD INSTR, DATA SET LCFD-1

6161				
6162				
6163				
6164				
6165				
6166	023512	000004		
6167	023514	012705	023526	
6168	023520	004737	036512	
6169				
6170	023524	000411		
6171				
6172	023526			
6173	023526	100000	000000	
6174	023532	052525	177777	125252
6175	023540	000000		
6176	023542	047643	147654	
6177	023546	100014		
6178				
6179				
6180				
6181				
6182				
6183				
6184				
6185	023550	000004		
6186	023552	012705	023564	
6187	023556	004737	036512	
6188				
6189	023562	000411		
6190				
6191	023564			
6192	023564	125252	125252	
6193	023570	125252	125252	000000
6194	023576	000000		
6195	023600	047607	047610	
6196	023604	000000		
6197				
6198				
6199				
6200				
6201				
6202				
6203				
6204	023606	000004		
6205	023610	012705	023622	
6206	023614	004737	036512	
6207				
6208	023620	000411		
6209				
6210	023622			
6211	023622	000000	000000	
6212	023626	000000	000000	000000
6213	023634	000000		
6214	023636	047753	047744	
6215	023642	000000		
6216				

```

*****
*TEST 352 TEST OF LDCFD INSTR, DATA SET LCFD-1
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

†T352: SCOPE
MOV #LCFD1,R5 ; PTR TO TEST DATA SET
JSR PC,2#LCFDT ; GO TEST
BR TST353 ;;

```

```

LCFD1: ; TEST DATA SET LCFD-1:
.WORD MO,0 ; INITIAL MEM FLOAT NUMBER
.WORD ALTN,M1,ALTN,0 ; EXPECTED FLOAT RESULT
.WORD 047643,147654 ; FPS: BEFORE AFTER
.WORD 100014 ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 353 TEST OF LDCFD INSTR, DATA SET LCFD-2
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

†T353: SCOPE
MOV #LCFD2,R5 ; PTR TO TEST DATA SET
JSR PC,2#LCFDT ; GO TEST
BR TST354 ;;

```

```

LCFD2: ; TEST DATA SET LCFD-2:
.WORD ALTN,ALTN ; INITIAL MEM FLOAT NUMBER
.WORD ALTN,ALTN,0,0 ; EXPECTED FLOAT RESULT
.WORD 047607,047610 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 354 TEST OF LDCFD INSTR, DATA SET LCFD-3
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

†T354: SCOPE
MOV #LCFD3,R5 ; PTR TO TEST DATA SET
JSR PC,2#LCFDT ; GO TEST
BR TST355 ;;

```

```

LCFD3: ; TEST DATA SET LCFD-3:
.WORD 0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 047753,047744 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

F10

FPU ADVANCED INSTR TESTS
DOFP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 124
T354 TEST OF LDCFD INSTR, DATA SET LCFD-3

6217				
6218				
6219				
6220				
6221				
6222				
6223	023644	000004		
6224	023646	012705	023660	
6225	023652	004737	036512	
6226				
6227	023656	000411		
6228				
6229	023660			
6230	023660	077777	177777	
6231	023664	077777	177777	000000
6232	023672	000000		
6233	023674	047717	047700	
6234	023700	000000		
6235				
6236				
6237				
6238				
6239				
6240				
6241				
6242	023702	000004		
6243	023704	012705	023716	
6244	023710	004737	036512	
6245				
6246	023714	000411		
6247				
6248	023716			
6249	023716	000177	177777	
6250	023722	000000	000000	000000
6251	023730	000000		
6252	023732	047653	047644	
6253	023736	000000		
6254				
6255				
6256				
6257				
6258				
6259				
6260				
6261	023740	000004		
6262	023742	012705	023754	
6263	023746	004737	036512	
6264				
6265	023752	000411		
6266				
6267	023754			
6268	023754	177777	177777	
6269	023760	177777	177777	000000
6270	023766	000000		
6271	023770	047607	047610	
6272	023774	000000		

```

*****
;TEST 355 TEST OF LDCFD INSTR, DATA SET LCFD-4
;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
↑ST355: SCOPE
MOV #LCFD4,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCFDT ; GO TEST

BR TST356 ;;

```

```

LCFD4: ; TEST DATA SET LCFD-4:
.WORD LGP,M1 ; INITIAL MEM FLOAT NUMBER
.WORD LGP,M1,0,0 ; EXPECTED FLOAT RESULT

.WORD 047717,047700 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
;TEST 356 TEST OF LDCFD INSTR, DATA SET LCFD-5
;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
↑ST356: SCOPE
MOV #LCFD5,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCFDT ; GO TEST

BR TST357 ;;

```

```

LCFD5: ; TEST DATA SET LCFD-5:
.WORD ZXIMP,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FLOAT RESULT

.WORD 047653,047644 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
;TEST 357 TEST OF LDCFD INSTR, DATA SET LCFD-6
;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****
↑ST357: SCOPE
MOV #LCFD6,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCFDT ; GO TEST

BR TST360 ;;

```

```

LCFD6: ; TEST DATA SET LCFD-6:
.WORD LGN,M1 ; INITIAL MEM FLOAT NUMBER
.WORD LGN,M1,0,0 ; EXPECTED FLOAT RESULT

.WORD 047607,047610 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

G10

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 125
T357 TEST OF LDCFD INSTR, DATA SET LCFD-6

6273				
6274				
6275				
6276				
6277				
6278				
6279				
6280	023776	000004		
6281	024000	012705	024012	
6282	024004	004737	036512	
6283				
6284	024010	000411		
6285				
6286	024012			
6287	024012	100177	177777	
6288	024016	000000	000000	000000
6289	024024	000000		
6290	024026	043753	043744	
6291	024032	000000		
6292				
6293				
6294				
6295				
6296				
6297				
6298				
6299	024034	000004		
6300	024036	012705	024050	
6301	024042	004737	036512	
6302				
6303	024046	000411		
6304				
6305	024050			
6306	024050	007417	007417	
6307	024054	007417	007417	000000
6308	024062	000000		
6309	024064	047717	047700	
6310	024070	000000		
6311				
6312				
6313				

```

*****
;TEST 360 TEST OF LDCFD INSTR, DATA SET LCFD-7
;
; -0 INTERRUPT ENABLE OFF, ALL OTHERS ON
; LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

†ST360: SCOPE
MOV #LCFD7,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCFDT ; GO TEST
BR TST361 ;;

LCFD7: ; TEST DATA SET LCFD-7:
.WORD ZX1MN,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 043753,043744 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
;TEST 361 TEST OF LDCFD INSTR, DATA SET LCFD-10
;
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

†ST361: SCOPE
MOV #LCFD10,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCFDT ; GO TEST
BR TST362 ;;

LCFD10: ; TEST DATA SET LCFD-10:
.WORD ALT4P,ALT4P ; INITIAL MEM FLOAT NUMBER
.WORD ALT4P,ALT4P,0,0 ; EXPECTED FLOAT RESULT
.WORD 047717,047700 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

H10

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 126
T362 TEST OF STCDF INSTR, DATA SET SCDF-1

```
6314
6315
6316
6317
6318
6319 024072 000004
6320 024074 012705 024106
6321 024100 004737 036702
6322
6323 024104 000411
6324
6325 024106
6326 024106 000000 000000 000000
6327 024114 000000
6328 024116 000000 000000
6329 024122 047753 047744
6330 024126 000000
6331
6332
6333
6334
6335
6336
6337
6338 024130 000004
6339 024132 012705 024144
6340 024136 004737 036702
6341
6342 024142 000411
6343
6344 024144
6345 024144 140200 000000 100000
6346 024152 000000
6347 024154 140200 000001
6348 024160 047707 047710
6349 024164 000000
6350
6351
6352
6353
6354
6355
6356
6357 024166 000004
6358 024170 012705 024202
6359 024174 004737 036702
6360
6361 024200 000411
6362
6363 024202
6364 024202 040200 000000 100000
6365 024210 000000
6366 024212 040200 000000
6367 024216 047657 047640
6368 024222 000000
6369
```

```
*****
*TEST 362 TEST OF STCDF INSTR, DATA SET SCDF-1
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****
†ST362: SCOPE
MOV #SCDF1,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCDFT ; GO TEST
BR TST363 ;;
SCDF1: ; TEST DATA SET SCDF-1:
.WORD 0,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD 047753,047744 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
*****
*TEST 363 TEST OF STCDF INSTR, DATA SET SCDF-2
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
†ST363: SCOPE
MOV #SCDF2,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCDFT ; GO TEST
BR TST364 ;;
SCDF2: ; TEST DATA SET SCDF-2:
.WORD FIN,0,MO,0 ; INITIAL AC FLOAT NUMBER
.WORD FIN,1 ; EXPECTED FLOAT RESULT
.WORD 047707,047710 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
*****
*TEST 364 TEST OF STCDF INSTR, DATA SET SCDF-3
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
†ST364: SCOPE
MOV #SCDF3,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCDFT ; GO TEST
BR TST365 ;;
SCDF3: ; TEST DATA SET SCDF-3:
.WORD FIP,0,MO,0 ; INITIAL AC FLOAT NUMBER
.WORD FIP,0 ; EXPECTED FLOAT RESULT
.WORD 047657,047640 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
```

```

6370
6371
6372
6373
6374
6375
6376 024224 000004
6377 024226 012705 024240
6378 024232 004737 036702
6379
6380 024236 000411
6381
6382 024240
6383 024240 000177 177777 177777
6384 024246 177777
6385 024250 000000 000000
6386 024254 047613 047604
6387 024260 000000
6388
6389
6390
6391
6392
6393
6394
6395 024262 000004
6396 024264 012705 024276
6397 024270 004737 036702
6398
6399 024274 000411
6400
6401 024276
6402 024276 040200 000000 100000
6403 024304 000000
6404 024306 040200 000001
6405 024312 047717 047700
6406 024316 000000
6407
6408
6409
6410
6411
6412
6413
6414 024320 000004
6415 024322 012705 024334
6416 024326 004737 036702
6417
6418 024332 000411
6419
6420 024334
6421 024334 177777 177777 177777
6422 024342 177777
6423 024344 177777 177777
6424 024350 047747 047750
6425 024354 000000

```

```

*****
*TEST 365 TEST OF STCDF INSTR, DATA SET SCDF-4
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

†ST365: SCOPE
MOV #SCDF4,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCDFT ; GO TEST
BR TST366 ;;

```

```

SCDF4: ; TEST DATA SET SCDF-4:
.WORD ZXIMP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD 047613,047604 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 366 TEST OF STCDF INSTR, DATA SET SCDF-5
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

†ST366: SCOPE
MOV #SCDF5,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCDFT ; GO TEST
BR TST367 ;;

```

```

SCDF5: ; TEST DATA SET SCDF-5:
.WORD FIP,0,MO,0 ; INITIAL AC FLOAT NUMBER
.WORD FIP,1 ; EXPECTED FLOAT RESULT
.WORD 047717,047700 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 367 TEST OF STCDF INSTR, DATA SET SCDF-6
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

†ST367: SCOPE
MOV #SCDF6,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCDFT ; GO TEST
BR TST370 ;;

```

```

SCDF6: ; TEST DATA SET SCDF-6:
.WORD LGN,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD LGN,M1 ; EXPECTED FLOAT RESULT
.WORD 047747,047750 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

J10

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 128
T367 TEST OF STCDF INSTR, DATA SET SCDF-6

6426
6427
6428
6429
6430
6431
6432
6433 024356 000004
6434 024360 012705 024372
6435 024364 004737 036702
6436
6437 024370 000411
6438
6439 024372
6440 024372 040200 000000 077777
6441 024400 177777
6442 024402 040200 000000
6443 024406 047617 047600
6444 024412 000000
6445
6446
6447
6448
6449
6450
6451
6452 024414 000004
6453 024416 012705 024430
6454 024422 004737 036702
6455
6456 024426 000411
6457
6458 024430
6459 024430 177777 177777 177777
6460 024436 177777
6461 024440 100000 000000
6462 024444 047601 147616
6463 024450 100010
6464
6465
6466
6467
6468
6469
6470
6471 024452 000004
6472 024454 012705 024466
6473 024460 004737 036702
6474
6475 024464 000411
6476
6477 024466
6478 024466 040200 000000 077777
6479 024474 177777
6480 024476 040200 000000
6481 024502 047757 047740

```

*****
;TEST 370 TEST OF STCDF INSTR, DATA SET SCDF-7
;
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, SHORT INTEGER, ROUND MODES
*****
†ST370: SCOPE
MOV #SCDF7_RS ; PTR TO TEST DATA SET
JSR PC,@#SCDF7 ; GO TEST
BR TST371 ;;

SCDF7: ; TEST DATA SET SCDF-7:
.WORD FIP,0,LGP,M1 ; INITIAL AC FLOAT NUMBER
.WORD FIP,0 ; EXPECTED FLOAT RESULT
.WORD 047617,047600 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
;TEST 371 TEST OF STCDF INSTR, DATA SET SCDF-10
;
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, SHORT INTEGER, ROUND MODES
*****
†ST371: SCOPE
MOV #SCDF10_RS ; PTR TO TEST DATA SET
JSR PC,@#SCDF10 ; GO TEST
BR TST372 ;;

SCDF10: ; TEST DATA SET SCDF-10:
.WORD LGN,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD MO,0 ; EXPECTED FLOAT RESULT
.WORD 047601,147616 ; FPS: BEFORE, AFTER
.WORD 100010 ; FEC AFTER ( 0 = N/A )

*****
;TEST 372 TEST OF STCDF INSTR, DATA SET SCDF-11
;
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****
†ST372: SCOPE
MOV #SCDF11_RS ; PTR TO TEST DATA SET
JSR PC,@#SCDF11 ; GO TEST
BR TST373 ;;

SCDF11: ; TEST DATA SET SCDF-11:
.WORD FIP,0,LGP,M1 ; INITIAL AC FLOAT NUMBER
.WORD FIP,0 ; EXPECTED FLOAT RESULT
.WORD 047757,047740 ; FPS: BEFORE, AFTER

```


K10

FPU ADVANCED INSTR TESTS
DQFP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 129
T372 TEST OF STCDF INSTR, DATA SET SCDF-11

6482 024506 000000 .WORD NA ; FEC AFTER (0 = N/A)

6483
6484
6485
6486
6487
6488
6489
6490 024510 000004
6491 024512 012705 024524
6492 024516 004737 036702
6493
6494 024522 000411
6495
6496 024524
6497 024524 101777 177777 100000
6498 024532 000000
6499 024534 102000 000000
6500 024540 047707 047710
6501 024544 000000
6502
6503
6504
6505
6506
6507
6508
6509 024546 000004
6510 024550 012705 024562
6511 024554 004737 036702
6512
6513 024560 000411
6514
6515 024562
6516 024562 101777 177777 100000
6517 024570 000000
6518 024572 101777 177777
6519 024576 047647 047650
6520 024602 000000
6521
6522
6523
6524
6525
6526
6527
6528 024604 000004
6529 024606 012705 024620
6530 024612 004737 036702
6531
6532 024616 000411
6533
6534 024620
6535 024620 077777 177777 100000
6536 024626 000000
6537 024630 000000 000000

```
*****  
: TEST 373 TEST OF STCDF INSTR, DATA SET SCDF-12  
: * ALL INTERRUPT ENABLES ON  
: * LONG FLOAT, LONG INTEGER, ROUND MODES  
: *****
```

```
TST373: SCOPE  
MOV #SCDF12,R5 ; PTR TO TEST DATA SET  
JSR PC,@#SCDFT ; GO TEST  
BR TST374 ;;
```

```
SCDF12: ; TEST DATA SET SCDF-12:  
.WORD 101777,M1,MO,0 ; INITIAL AC FLOAT NUMBER  
.WORD 102000,0 ; EXPECTED FLOAT RESULT  
.WORD 047707,047710 ; FPS: BEFORE, AFTER  
.WORD NA ; FEC AFTER ( 0 = N/A )
```

```
*****  
: TEST 374 TEST OF STCDF INSTR, DATA SET SCDF-13  
: * ALL INTERRUPT ENABLES ON  
: * LONG FLOAT, SHORT INTEGER, TRUNCATE MODES  
: *****
```

```
TST374: SCOPE  
MOV #SCDF13,R5 ; PTR TO TEST DATA SET  
JSR PC,@#SCDFT ; GO TEST  
BR TST375 ;;
```

```
SCDF13: ; TEST DATA SET SCDF-13:  
.WORD 101777,M1,MO,0 ; INITIAL AC FLOAT NUMBER  
.WORD 101777,M1 ; EXPECTED FLOAT RESULT  
.WORD 047647,047650 ; FPS: BEFORE, AFTER  
.WORD NA ; FEC AFTER ( 0 = N/A )
```

```
*****  
: TEST 375 TEST OF STCDF INSTR, DATA SET SCDF-14  
: * OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON  
: * LONG FLOAT, SHORT INTEGER, ROUND MODES  
: *****
```

```
TST375: SCOPE  
MOV #SCDF14,R5 ; PTR TO TEST DATA SET  
JSR PC,@#SCDFT ; GO TEST  
BR TST376 ;;
```

```
SCDF14: ; TEST DATA SET SCDF-14:  
.WORD LGP,M1,MO,0 ; INITIAL AC FLOAT NUMBER  
.WORD 0,0 ; EXPECTED FLOAT RESULT
```

L10

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 130
T375 TEST OF STCDF INSTR, DATA SET SCDF-14

6538 024634 046611 046606
6539 024640 000000
6540
6541
6542

.WORD 046611,046606 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

M10

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 131
T376 TEST OF STCFD INSTR, DATA SET SCFD-1

```

6543 .....
6544 *TEST 376 TEST OF STCFD INSTR, DATA SET SCFD-1
6545 * ALL INTERRUPT ENABLES ON
6546 * SHORT FLOAT, SHORT INTEGER, ROUND MODES
6547 .....
6548 024642 000004
6549 024644 012705 024656
6550 024650 004737 037046
6551
6552 024654 000412
6553
6554 024656
6555 024656 052525 052525 052525 SCFD1: ; TEST DATA SET SCFD-1:
6556 024664 052525 .WORD ALTP,ALTP,ALTP,ALTP ; INITIAL AC FLOAT NUMBER
6557 024666 052525 052525 000000 .WORD ALTP,ALTP,0,0 ; EXPECTED FLOAT RESULT
6558 024674 000000
6559 024676 047417 047400 .WORD 047417,047400 ; FPS: BEFORE, AFTER
6560
6561
6562 .....
6563 *TEST 377 TEST OF STCFD INSTR, DATA SET SCFD-2
6564 * ALL INTERRUPT ENABLES ON
6565 * SHORT FLOAT, LONG INTEGER, ROUND MODES
6566 .....
6567 024702 000004
6568 024704 012705 024716
6569 024710 004737 037046
6570
6571 024714 000412
6572
6573 024716
6574 024716 000177 177777 177777 SCFD2: ; TEST DATA SET SCFD-2:
6575 024724 177777 .WORD ZXIMP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
6576 024726 000000 000000 000000 .WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
6577 024734 000000
6578 024736 047513 047504 .WORD 047513,047504 ; FPS: BEFORE, AFTER
6579
6580
6581 .....
6582 *TEST 400 TEST OF STCFD INSTR, DATA SET SCFD-3
6583 * ALL INTERRUPT ENABLES ON
6584 * SHORT FLOAT, SHORT INTEGER, ROUND MODES
6585 .....
6586 024742 000004
6587 024744 012705 024756
6588 024750 004737 037046
6589
6590 024754 000412
6591
6592 024756
6593 024756 177777 177777 177777 SCFD3: ; TEST DATA SET SCFD-3:
6594 024764 177777 .WORD LGN,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
6595 024766 177777 177777 000000 .WORD LGN,M1,0,0 ; EXPECTED FLOAT RESULT
6596 024774 000000
6597 024776 047407 047410 .WORD 047407,047410 ; FPS: BEFORE, AFTER
6598

```

```

6599
6600
6601
6602
6603
6604
6605 025002 000004
6606 025004 012705 025016
6607 025010 004737 037046
6608
6609 025014 000412
6610
6611 025016
6612 025016 170360 170360 170360
6613 025024 170360
6614 025026 170360 170360 000000
6615 025034 000000
6616 025036 047547 047550
6617
6618
6619
6620
6621
6622
6623
6624 025042 000004
6625 025044 012705 025056
6626 025050 004737 037046
6627
6628 025054 000412
6629
6630 025056
6631 025056 000000 000000 000000
6632 025064 000000
6633 025066 000000 000000 000000
6634 025074 000000
6635 025076 047453 047444
6636
6637
6638
6639
6640
6641
6642
6643 025102 000004
6644 025104 012705 025116
6645 025110 004737 037046
6646
6647 025114 000412
6648
6649 025116
6650 025116 077777 000000 177777
6651 025124 177777
6652 025126 077777 000000 000000
6653 025134 000000
6654 025136 047517 047500

```

```

*****
: *TEST 401 TEST OF STCFD INSTR, DATA SET SCFD-4
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
: *
*****

```

```

†ST401: SCOPE
MOV #SCFD4,R5 ; PTR TO TEST DATA SET
JSR PC,2#SCFDT ; GO TEST
BR TST402 ;;

SCFD4: ; TEST DATA SET SCFD-4:
.WORD ALT4N,ALT4N,ALT4N,ALT4N ; INITIAL AC FLOAT NUMBER
.WORD ALT4N,ALT4N,0,0 ; EXPECTED FLOAT RESULT
.WORD 047547,047550 ; FPS: BEFORE, AFTER

```

```

*****
: *TEST 402 TEST OF STCFD INSTR, DATA SET SCFD-5
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
: *
*****

```

```

†ST402: SCOPE
MOV #SCFD5,R5 ; PTR TO TEST DATA SET
JSR PC,2#SCFDT ; GO TEST
BR TST403 ;;

SCFD5: ; TEST DATA SET SCFD-5:
.WORD 0,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 047453,047444 ; FPS: BEFORE, AFTER

```

```

*****
: *TEST 403 TEST OF STCFD INSTR, DATA SET SCFD-6
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, LONG INTEGER, ROUND MODES
: *
*****

```

```

†ST403: SCOPE
MOV #SCFD6,R5 ; PTR TO TEST DATA SET
JSR PC,2#SCFDT ; GO TEST
BR TST404 ;;

SCFD6: ; TEST DATA SET SCFD-6:
.WORD LGP,0,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD LGP,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 047517,047500 ; FPS: BEFORE, AFTER

```

B11

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 133
T403 TEST OF STCFD INSTR, DATA SET SCFD-6

6655
6656

6657			
6658			
6659			
6660			
6661			
6662	025142	000004	
6663	025144	012705	025156
6664	025150	004737	037170
6665			
6666	025154	000405	
6667			
6668	025156		
6669	025156	100000	
6670	025160	144000	000000
6671	025164	047407	047410
6672			
6673			
6674			
6675			
6676			
6677			
6678			
6679	025170	000004	
6680	025172	012705	025204
6681	025176	004737	037170
6682			
6683	025202	000405	
6684			
6685	025204		
6686	025204	007417	
6687	025206	043160	170000
6688	025212	047457	047440
6689			
6690			
6691			
6692			
6693			
6694			
6695			
6696	025216	000004	
6697	025220	012705	025232
6698	025224	004737	037170
6699			
6700	025230	000405	
6701			
6702	025232		
6703	025232	000000	
6704	025234	000000	000000
6705	025240	047413	047404
6706			
6707			
6708			
6709			
6710			
6711			
6712			

```

*****
;TEST 404 TEST OF LDCIF INSTR, DATA SET LCIF-1
;
; ALL INTERRUPT ENABLES ON
; SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****
TST404: SCOPE
MOV #LCIF1,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCIFT ; GO TEST
BR TST405 ;;

LCIF1: ; TEST DATA SET LCIF-1:
.WORD 100000 ; INITIAL INTEGER VALUE
.WORD 144000,000000 ; EXPECTED FLOAT RESULT
.WORD 047407,047410 ; FPS: BEFORE, AFTER

*****
;TEST 405 TEST OF LDCIF INSTR, DATA SET LCIF-2
;
; ALL INTERRUPT ENABLES ON
; SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST405: SCOPE
MOV #LCIF2,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCIFT ; GO TEST
BR TST406 ;;

LCIF2: ; TEST DATA SET LCIF-2:
.WORD 007417 ; INITIAL INTEGER VALUE
.WORD 043160,170000 ; EXPECTED FLOAT RESULT
.WORD 047457,047440 ; FPS: BEFORE, AFTER

*****
;TEST 406 TEST OF LDCIF INSTR, DATA SET LCIF-3
;
; ALL INTERRUPT ENABLES ON
; SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****
TST406: SCOPE
MOV #LCIF3,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCIFT ; GO TEST
BR TST407 ;;

LCIF3: ; TEST DATA SET LCIF-3:
.WORD 000000 ; INITIAL INTEGER VALUE
.WORD 000000,000000 ; EXPECTED FLOAT RESULT
.WORD 047413,047404 ; FPS: BEFORE, AFTER

*****
;TEST 407 TEST OF LDCIF INSTR, DATA SET LCIF-4
;
; ALL INTERRUPT ENABLES ON
; SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

FPU ADVANCED INSTR TESTS
D9FPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 135
T407 TEST OF LDCIF INSTR, DATA SET LCIF-4

6713	025244	000004	
6714	025246	012705	025260
6715	025252	004737	037170
6716			
6717	025256	000405	
6718			
6719	025260		
6720	025260	170360	
6721	025262	143161	000000
6722	025266	047447	047450
6723			
6724			
6725			
6726			
6727			
6728			
6729			
6730	025272	000004	
6731	025274	012705	025306
6732	025300	004737	037170
6733			
6734	025304	000405	
6735			
6736	025306		
6737	025306	077777	
6738	025310	043777	177000
6739	025314	047417	047400
6740			
6741			
6742			

```
TST407: SCOPE
MOV      #LCIF4,R5      ; PTR TO TEST DATA SET
JSR      PC,#LCIFT     ; GO TEST
BR       TST410        ;;
```

```
LCIF4: ; TEST DATA SET LCIF-4:
.WORD   170360 ; INITIAL INTEGER VALUE
.WORD   143161,000000 ; EXPECTED FLOAT RESULT
.WORD   047447,047450 ; FPS: BEFORE, AFTER
```

```
*****
*TEST 410 TEST OF LDCIF INSTR, DATA SET LCIF-5
* ALL INTERRUPT ENABLS ON
* SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****
```

```
TST410: SCOPE
MOV      #LCIF5,R5      ; PTR TO TEST DATA SET
JSR      PC,#LCIFT     ; GO TEST
BR       TST411        ;;
```

```
LCIF5: ; TEST DATA SET LCIF-5:
.WORD   077777 ; INITIAL INTEGER VALUE
.WORD   043777,177000 ; EXPECTED FLOAT RESULT
.WORD   047417,047400 ; FPS: BEFORE, AFTER
```

E11

FPU ADVANCED INSTR TESTS
DOFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 136
T411 TEST OF LDCID INSTR, DATA SET LDCID-1

6743				
6744				
6745				
6746				
6747				
6748	025320	000004		
6749	025322	012705	025334	
6750	025326	004737	037270	
6751				
6752	025332	000407		
6753				
6754	025334			
6755	025334	107070		
6756	025336	143743	110000	000000
6757	025344	000000		
6758	025346	047600	047610	
6759				
6760				
6761				
6762				
6763				
6764				
6765				
6766	025352	000004		
6767	025354	012705	025366	
6768	025360	004737	037270	
6769				
6770	025364	000407		
6771				
6772	025366			
6773	025366	000000		
6774	025370	000000	000000	000000
6775	025376	000000		
6776	025400	047653	047644	
6777				
6778				
6779				
6780				
6781				
6782				
6783				
6784	025404	000004		
6785	025406	012705	025420	
6786	025412	004737	037270	
6787				
6788	025416	000407		
6789				
6790	025420			
6791	025420	077777		
6792	025422	043777	177000	000000
6793	025430	000000		
6794	025432	047657	047640	
6795				
6796				
6797				
6798				

```

*****
;TEST 411      TEST OF LDCID INSTR, DATA SET LDCID-1
;
;      ALL INTERRUPT ENABLES ON
;      LONG FLOAT, SHORT INTEGER, ROUND MODES
*****
TST411: SCOPE
        MOV     #LDCID1,R5      ; PTR TO TEST DATA SET
        JSR    PC,@#LDCIDT     ; GO TEST
        BR     TST412         ;;

LDCID1: ; TEST DATA SET LDCID-1:
        .WORD  107070          ; INITIAL INTEGER VALUE
        .WORD  143743,110000,0,0 ; EXPECTED FLOAT RESULT
        .WORD  047600,047610  ; FPS: BEFORE, AFTER

*****
;TEST 412      TEST OF LDCID INSTR, DATA SET LDCID-2
;
;      ALL INTERRUPT ENABLES ON
;      LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST412: SCOPE
        MOV     #LDCID2,R5      ; PTR TO TEST DATA SET
        JSR    PC,@#LDCIDT     ; GO TEST
        BR     TST413         ;;

LDCID2: ; TEST DATA SET LDCID-2:
        .WORD  000000          ; INITIAL INTEGER VALUE
        .WORD  000000,000000,0,0 ; EXPECTED FLOAT RESULT
        .WORD  047653,047644  ; FPS: BEFORE, AFTER

*****
;TEST 413      TEST OF LDCID INSTR, DATA SET LDCID-3
;
;      ALL INTERRUPT ENABLES ON
;      LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST413: SCOPE
        MOV     #LDCID3,R5      ; PTR TO TEST DATA SET
        JSR    PC,@#LDCIDT     ; GO TEST
        BR     TST414         ;;

LDCID3: ; TEST DATA SET LDCID-3:
        .WORD  077777          ; INITIAL INTEGER VALUE
        .WORD  043777,177000,0,0 ; EXPECTED FLOAT RESULT
        .WORD  047657,047640  ; FPS: BEFORE, AFTER

*****
;TEST 414      TEST OF LDCID INSTR, DATA SET LDCID-4

```


FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 137
T414 TEST OF LDCID INSTR, DATA SET LDCID-4

6799				
6800				
6801				
6802	025436	000004		
6803	025440	012705	025452	
6804	025444	004737	037270	
6805				
6806	025450	000407		
6807				
6808	025452			
6809	025452	070707		
6810	025454	043743	107000	000000
6811	025462	000000		
6812	025464	047617	047600	
6813				
6814				
6815				
6816				
6817				
6818				
6819				
6820	025470	000004		
6821	025472	012705	025504	
6822	025476	004737	037270	
6823				
6824	025502	000407		
6825				
6826	025504			
6827	025504	100000		
6828	025506	144000	000000	000000
6829	025514	000000		
6830	025516	047647	047650	
6831				
6832				
6833				

```

;*          ALL INTERRUPT ENABLES ON
;*          LONG FLOAT, SHORT INTEGER, ROUND MODES
;*****
↑ST414: SCOPE
MOV      #LDCID4,R5      ; PTR TO TEST DATA SET
JSR      PC,@#LDCIDT    ; GO TEST
BR       TST415         ;;

LDCID4: ; TEST DATA SET LDCID-4:
.WORD   070707          ; INITIAL INTEGER VALUE
.WORD   043743,107000,0,0 ; EXPECTED FLOAT RESULT
.WORD   047617,047600  ; FPS: BEFORE, AFTER

```

```

;*****
;TEST 415  TEST OF LDCID INSTR, DATA SET LDCID-5
;*          ALL INTERRUPT ENABLES ON
;*          LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
;*****
↑ST415: SCOPE
MOV      #LDCID5,R5      ; PTR TO TEST DATA SET
JSR      PC,@#LDCIDT    ; GO TEST
BR       TST416         ;;

LDCID5: ; TEST DATA SET LDCID-5:
.WORD   100000          ; INITIAL INTEGER VALUE
.WORD   144000,000000,0,0 ; EXPECTED FLOAT RESULT
.WORD   047647,047650  ; FPS: BEFORE, AFTER

```

G11

FPU ADVANCED INSTR TESTS
DOFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 138
T416 TEST OF LDCLF INSTR, DATA SET LCLF-1

6834
6835
6836
6837
6838
6839 025522 000004
6840 025524 012705 025536
6841 025530 004737 037410
6842
6843 025534 000406
6844
6845 025536
6846 025536 077777 177777
6847 025542 050000 000000
6848 025546 047517 047500
6849
6850
6851
6852
6853
6854
6855
6856 025552 000004
6857 025554 012705 025566
6858 025560 004737 037410
6859
6860 025564 000406
6861
6862 025566
6863 025566 077777 177777
6864 025572 047777 177777
6865 025576 047557 047540
6866
6867
6868
6869
6870
6871
6872
6873 025602 000004
6874 025604 012705 025616
6875 025610 004737 037410
6876
6877 025614 000406
6878
6879 025616
6880 025616 170360 170360
6881 025622 147160 170361
6882 025626 047507 047510
6883
6884
6885
6886
6887
6888
6889

```
*****
*TEST 416 TEST OF LDCLF INSTR, DATA SET LCLF-1
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****
↑TST416: SCOPE
MOV #LCLF1,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCLFT ; GO TEST
BR TST417 ;;

LCLF1: ; TEST DATA SET LCLF-1:
.WORD 077777,M1 ; INITIAL INTEGER VALUE
.WORD 050000,000000 ; EXPECTED FLOAT RESULT
.WORD 047517,047500 ; FPS: BEFORE, AFTER

*****
*TEST 417 TEST OF LDCLF INSTR, DATA SET LCLF-2
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****
↑TST417: SCOPE
MOV #LCLF2,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCLFT ; GO TEST
BR TST420 ;;

LCLF2: ; TEST DATA SET LCLF-2:
.WORD 077777,M1 ; INITIAL INTEGER VALUE
.WORD 047777,M1 ; EXPECTED FLOAT RESULT
.WORD 047557,047540 ; FPS: BEFORE, AFTER

*****
*TEST 420 TEST OF LDCLF INSTR, DATA SET LCLF-3
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****
↑TST420: SCOPE
MOV #LCLF3,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCLFT ; GO TEST
BR TST421 ;;

LCLF3: ; TEST DATA SET LCLF-3:
.WORD 170360,170360 ; INITIAL INTEGER VALUE
.WORD 147160,170361 ; EXPECTED FLOAT RESULT
.WORD 047507,047510 ; FPS: BEFORE, AFTER

*****
*TEST 421 TEST OF LDCLF INSTR, DATA SET LCLF-4
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****
```

H11

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 139
T421 TEST OF LDCLF INSTR, DATA SET LCLF-4

6890	025632	000004	
6891	025634	012705	025646
6892	025640	004737	037410
6893			
6894	025644	000406	
6895			
6896	025646		
6897	025646	000000	000000
6898	025652	000000	000000
6899	025656	047513	047504
6900			
6901			
6902			
6903			
6904			
6905			
6906			
6907	025662	000004	
6908	025664	012705	025676
6909	025670	004737	037410
6910			
6911	025674	000406	
6912			
6913	025676		
6914	025676	077777	177677
6915	025702	047777	177777
6916	025706	047517	047500
6917			
6918			
6919			
6920			
6921			
6922			
6923			
6924	025712	000004	
6925	025714	012705	025726
6926	025720	004737	037410
6927			
6928	025724	000406	
6929			
6930	025726		
6931	025726	100000	000000
6932	025732	150000	000000
6933	025736	047547	047550
6934			
6935			
6936			
6937			
6938			
6939			
6940			
6941	025742	000004	
6942	025744	012705	025756
6943	025750	004737	037410
6944			
6945	025754	000406	

```
TST421: SCOPE
MOV    #LCLF4,R5      ; PTR TO TEST DATA SET
JSR    PC,@#LCLFT    ; GO TEST

BR     TST422        ;;
```

```
LCLF4: ; TEST DATA SET LCLF-4:
.WORD  000000,000000 ; INITIAL INTEGER VALUE
.WORD  000000,000000 ; EXPECTED FLOAT RESULT
.WORD  047513,047504 ; FPS: BEFORE, AFTER
```

```
*****
;TEST 422 TEST OF LDCLF INSTR, DATA SET LCLF-5
;
; ALL INTERRUPT ENABLES ON
; SHORT FLOAT, LONG INTEGER, ROUND MODES
*****
```

```
TST422: SCOPE
MOV    #LCLF5,R5      ; PTR TO TEST DATA SET
JSR    PC,@#LCLFT    ; GO TEST

BR     TST423        ;;
```

```
LCLF5: ; TEST DATA SET LCLF-5:
.WORD  077777,177677 ; INITIAL INTEGER VALUE
.WORD  047777,M1     ; EXPECTED FLOAT RESULT
.WORD  047517,047500 ; FPS: BEFORE, AFTER
```

```
*****
;TEST 423 TEST OF LDCLF INSTR, DATA SET LCLF-6
;
; ALL INTERRUPT ENABLES ON
; SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****
```

```
TST423: SCOPE
MOV    #LCLF6,R5      ; PTR TO TEST DATA SET
JSR    PC,@#LCLFT    ; GO TEST

BR     TST424        ;;
```

```
LCLF6: ; TEST DATA SET LCLF-6:
.WORD  100000,000000 ; INITIAL INTEGER VALUE
.WORD  150000,000000 ; EXPECTED FLOAT RESULT
.WORD  047547,047550 ; FPS: BEFORE, AFTER
```

```
*****
;TEST 424 TEST OF LDCLF INSTR, DATA SET LCLF-7
;
; ALL INTERRUPT ENABLES ON
; SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****
```

```
TST424: SCOPE
MOV    #LCLF7,R5      ; PTR TO TEST DATA SET
JSR    PC,@#LCLFT    ; GO TEST

BR     TST425        ;;
```

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 140
T424 TEST OF LDCLF INSTR, DATA SET LCLF-7

6946			
6947	025756		
6948	025756	043434	070707
6949	025762	047616	034343
6950	025766	047557	047540
6951			
6952			
6953			

LCLF7: ; TEST DATA SET LCLF-7:
 .WORD 043434,070707 ; INITIAL INTEGER VALUE
 .WORD 047616,034343 ; EXPECTED FLOAT RESULT
 .WORD 047557,047540 ; FPS: BEFORE, AFTER

```

6954
6955
6956
6957
6958
6959 025772 000004
6960 025774 012705 026006
6961 026000 004737 037510
6962
6963 026004 000410
6964
6965 026006
6966 026006 007417 007417
6967 026012 047160 170360 170000
6968 026020 000000
6969 026022 047717 047700
6970
6971
6972
6973
6974
6975
6976
6977 026026 000004
6978 026030 012705 026042
6979 026034 004737 037510
6980
6981 026040 000410
6982
6983 026042
6984 026042 100000 000000
6985 026046 150000 000000 000000
6986 026054 000000
6987 026056 047747 047750
6988
6989
6990
6991
6992
6993
6994
6995 026062 000004
6996 026064 012705 026076
6997 026070 004737 037510
6998
6999 026074 000410
7000
7001 026076
7002 026076 077777 177777
7003 026102 047777 177777 177000
7004 026110 000000
7005 026112 047757 047740
7006
7007
7008
7009

```

```

*****
; *TEST 425 TEST OF LDCLD INSTR, DATA SET LCLD-1
; * ALL INTERRUPT ENABLES ON
; * LONG FLOAT, LONG INTEGER, ROUND MODES
*****
TST425: SCOPE
MOV #LCLD1,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCLDT ; GO TEST

BR TST426 ;;

LCLD1: ; TEST DATA SET LCLD-1:
.WORD 007417,007417 ; INITIAL INTEGER VALUE
.WORD 047160,170360,170000,0 ; EXPECTED FLOAT RESULT
.WORD 047717,047700 ; FPS: BEFORE, AFTER

*****
; *TEST 426 TEST OF LDCLD INSTR, DATA SET LCLD-2
; * ALL INTERRUPT ENABLES ON
; * LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****
TST426: SCOPE
MOV #LCLD2,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCLDT ; GO TEST

BR TST427 ;;

LCLD2: ; TEST DATA SET LCLD-2:
.WORD 100000,000000 ; INITIAL INTEGER VALUE
.WORD 150000,000000,000000,0 ; EXPECTED FLOAT RESULT
.WORD 047747,047750 ; FPS: BEFORE, AFTER

*****
; *TEST 427 TEST OF LDCLD INSTR, DATA SET LCLD-3
; * ALL INTERRUPT ENABLES ON
; * LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****
TST427: SCOPE
MOV #LCLD3,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCLDT ; GO TEST

BR TST430 ;;

LCLD3: ; TEST DATA SET LCLD-3:
.WORD 077777,M1 ; INITIAL INTEGER VALUE
.WORD 047777,M1,177000,0 ; EXPECTED FLOAT RESULT
.WORD 047757,047740 ; FPS: BEFORE, AFTER

*****
; *TEST 430 TEST OF LDCLD INSTR, DATA SET LCLD-4

```

K11

FPU ADVANCED INSTR TESTS
DFFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 142
T430 TEST OF LCLD INSTR, DATA SET LCLD-4

7010				
7011				
7012				
7013	026116	000004		
7014	026120	012705	026132	
7015	026124	004737	037510	
7016				
7017	026130	000410		
7018				
7019	026132			
7020	026132	107070	161616	
7021	026136	147743	107070	162000
7022	026144	000000		
7023	026146	047700	047710	
7024				
7025				
7026				
7027				
7028				
7029				
7030				
7031	026152	000004		
7032	026154	012705	026166	
7033	026160	004737	037510	
7034				
7035	026164	000410		
7036				
7037	026166			
7038	026166	000000	000000	
7039	026172	000000	000000	000000
7040	026200	000000		
7041	026202	047753	047744	
7042				
7043				
7044				

```

;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, LONG INTEGER, ROUND MODES
;*****
TST430: SCOPE
MOV #LCLD4,RS ; PTR TO TEST DATA SET
JSR PC,#LCLDT ; GO TEST
BR TST431 ;;

LCLD4: ; TEST DATA SET LCLD-4:
.WORD 107070,161616 ; INITIAL INTEGER VALUE
.WORD 147743,107070,162000,0 ; EXPECTED FLOAT RESULT
.WORD 047700,047710 ; FPS: BEFORE, AFTER

```

```

;*****
;TEST 431 TEST OF LCLD INSTR, DATA SET LCLD-5
;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
;*****
TST431: SCOPE
MOV #LCLD5,RS ; PTR TO TEST DATA SET
JSR PC,#LCLDT ; GO TEST
BR TST432 ;;

LCLD5: ; TEST DATA SET LCLD-5:
.WORD 000000,000000 ; INITIAL INTEGER VALUE
.WORD 000000,000000,000000,0 ; EXPECTED FLOAT RESULT
.WORD 047753,047744 ; FPS: BEFORE, AFTER

```

```

7045
7046
7047
7048
7049
7050 026206 000004
7051 026210 012705 026222
7052 026214 004737 037630
7053
7054 026220 000406
7055
7056 026222
7057 026222 000000 000000
7058 026226 000000
7059 026230 047453 047444
7060 026234 000000
7061
7062
7063
7064
7065
7066
7067
7068 026236 000004
7069 026240 012705 026252
7070 026244 004737 037630
7071
7072 026250 000406
7073
7074 026252
7075 026252 041532 000000
7076 026256 000066
7077 026260 047457 047440
7078 026264 000000
7079
7080
7081
7082
7083
7084
7085
7086 026266 000004
7087 026270 012705 026302
7088 026274 004737 037630
7089
7090 026300 000406
7091
7092 026302
7093 026302 052525 052525
7094 026306 000000
7095 026310 047452 147445
7096 026314 100006
7097
7098
7099
7100

```

```

*****
; *TEST 432 TEST OF STCFI INSTR, DATA SET SCFI-1
; * ALL INTERRUPT ENABLES ON
; * SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
†ST432: SCOPE
MOV #SCFI1,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCFIT ; GO TEST
BR TST433 ;;

SCFI1: ; TEST DATA SET SCFI-1:
.WORD 000000,000000 ; INITIAL FLOAT VALUE
.WORD 000000 ; EXPECTED INTEGER RESULT
.WORD 047453,047444 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
; *TEST 433 TEST OF STCFI INSTR, DATA SET SCFI-2
; * ALL INTERRUPT ENABLES ON
; * SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
†ST433: SCOPE
MOV #SCFI2,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCFIT ; GO TEST
BR TST434 ;;

SCFI2: ; TEST DATA SET SCFI-2:
.WORD 041532,000000 ; INITIAL FLOAT VALUE
.WORD 000066 ; EXPECTED INTEGER RESULT
.WORD 047457,047440 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
; *TEST 434 TEST OF STCFI INSTR, DATA SET SCFI-3
; * ALL INTERRUPT ENABLES ON
; * SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
†ST434: SCOPE
MOV #SCFI3,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCFIT ; GO TEST
BR TST435 ;;

SCFI3: ; TEST DATA SET SCFI-3:
.WORD 052525,052525 ; INITIAL FLOAT VALUE
.WORD 000000 ; EXPECTED INTEGER RESULT
.WORD 047452,147445 ; FPS: BEFORE AFTER
.WORD 100006 ; FEC AFTER ( 0 = N/A )

*****
; *TEST 435 TEST OF STCFI INSTR, DATA SET SCFI-4

```

M11

FPU ADVANCED INSTR TESTS
DOFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 144
T435 TEST OF STCFI INSTR, DATA SET SCFI-4

7101
7102
7103
7104 026316 000004
7105 026320 012705 026332
7106 026324 004737 037630
7107
7108 026330 000406
7109
7110 026332
7111 026332 141531 177777
7112 026336 177712
7113 026340 047407 047410
7114 026344 000000
7115
7116
7117
7118
7119
7120
7121
7122 026346 000004
7123 026350 012705 026362
7124 026354 004737 037630
7125
7126 026360 000406
7127
7128 026362
7129 026362 041532 000000
7130 026366 000066
7131 026370 047417 047400
7132 026374 000000
7133
7134
7135
7136
7137
7138
7139
7140 026376 000004
7141 026400 012705 026412
7142 026404 004737 037630
7143
7144 026410 000406
7145
7146 026412
7147 026412 172011 123456
7148 026416 000000
7149 026420 047052 047045
7150 026424 000000
7151
7152
7153
7154
7155
7156

```

;*
;* ALL INTERRUPT ENABLES ON
;* SHORT FLOAT, SHORT INTEGER, ROUND MODES
;*****
TST435: SCOPE
MOV #SCFI4,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCFIT ; GO TEST
BR TST436 ;;

SCFI4: ; TEST DATA SET SCFI-4:
.WORD 141531,M1 ; INITIAL FLOAT VALUE
.WORD 177712 ; EXPECTED INTEGER RESULT
.WORD 047407,047410 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

;*****
;TEST 436 TEST OF STCFI INSTR, DATA SET SCFI-5
;*
;* ALL INTERRUPT ENABLES ON
;* SHORT FLOAT, SHORT INTEGER, ROUND MODES
;*****
TST436: SCOPE
MOV #SCFI5,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCFIT ; GO TEST
BR TST437 ;;

SCFI5: ; TEST DATA SET SCFI-5:
.WORD 041532,000000 ; INITIAL FLOAT VALUE
.WORD 000066 ; EXPECTED INTEGER RESULT
.WORD 047417,047400 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

;*****
;TEST 437 TEST OF STCFI INSTR, DATA SET SCFI-6
;*
;* INTEGER CONVERSION INTERRUPT ENABLE OFF, ALL OTHERS ON
;* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
;*****
TST437: SCOPE
MOV #SCFI6,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCFIT ; GO TEST
BR TST440 ;;

SCFI6: ; TEST DATA SET SCFI-6:
.WORD 172011,123456 ; INITIAL FLOAT VALUE
.WORD 000000 ; EXPECTED INTEGER RESULT
.WORD 047052,047045 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

;*****
;TEST 440 TEST OF STCFI INSTR, DATA SET SCFI-7
;*
;* ALL INTERRUPT ENABLES ON
;* SHORT FLOAT, SHORT INTEGER, ROUND MODES
;*****
```


FPU ADVANCED INSTR TESTS
DQFP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 145
T440 TEST OF STCFI INSTR, DATA SET SCFI-7

7157			
7158	026426	000004	
7159	026430	012705	026442
7160	026434	004737	037630
7161			
7162	026440	000406	
7163			
7164	026442		
7165	026442	000000	177777
7166	026446	000000	
7167	026450	047413	047404
7168	026454	000000	
7169			
7170			
7171			

```

*****
TST440: SCOPE
MOV #SCFI7,R5 ; PTR TO TEST DATA SET
JSR PC,#SCFIT ; GO TEST

BR TST441 ;;

SCFI7: ; TEST DATA SET SCFI-7:
.WORD 000000,M1 ; INITIAL FLOAT VALUE
.WORD 000000 ; EXPECTED INTEGER RESULT
.WORD 047413,047404 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

7172
7173
7174
7175
7176
7177 026456 000004
7178 026460 012705 026472
7179 026464 004737 040026
7180
7181 026470 000410
7182
7183 026472
7184 026472 044000 000000 000000
7185 026500 000000
7186 026502 000000
7187 026504 047652 147645
7188 026510 100006
7189
7190
7191
7192
7193
7194
7195
7196 026512 000004
7197 026514 012705 026526
7198 026520 004737 040026
7199
7200 026524 000410
7201
7202 026526
7203 026526 043777 177377 177777
7204 026534 177777
7205 026536 077777
7206 026540 047617 047600
7207 026544 000000
7208
7209
7210
7211
7212
7213
7214
7215 026546 000004
7216 026550 012705 026562
7217 026554 004737 040026
7218
7219 026560 000410
7220
7221 026562
7222 026562 000000 000000 000000
7223 026570 000000
7224 026572 000000
7225 026574 047613 047604
7226 026600 000000
7227

```

```

*****
*TEST 441 TEST OF STCDI INSTR, DATA SET SCDI-1
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST441: SCOPE
MOV #SCDI1,RS ; PTR TO TEST DATA SET
JSR PC,@#SCDIT ; GO TEST
BR TST442 ;;

SCDI1: ; TEST DATA SET SCDI-1:
.WORD 044000,000000,000000,000000 ; INITIAL FLOAT VALUE
.WORD 000000 ; EXPECTED INTEGER RESULT
.WORD 047652,147645 ; FPS: BEFORE AFTER
.WORD 100006 ; FEC AFTER ( 0 = N/A )

*****
*TEST 442 TEST OF STCDI INSTR, DATA SET SCDI-2
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****
TST442: SCOPE
MOV #SCDI2,RS ; PTR TO TEST DATA SET
JSR PC,@#SCDIT ; GO TEST
BR TST443 ;;

SCDI2: ; TEST DATA SET SCDI-2:
.WORD 043777,177377,M1,M1 ; INITIAL FLOAT VALUE
.WORD 077777 ; EXPECTED INTEGER RESULT
.WORD 047617,047600 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
*TEST 443 TEST OF STCDI INSTR, DATA SET SCDI-3
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****
TST443: SCOPE
MOV #SCDI3,RS ; PTR TO TEST DATA SET
JSR PC,@#SCDIT ; GO TEST
BR TST444 ;;

SCDI3: ; TEST DATA SET SCDI-3:
.WORD 000000,000000,000000,000000 ; INITIAL FLOAT VALUE
.WORD 000000 ; EXPECTED INTEGER RESULT
.WORD 047613,047604 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

7228
7229
7230
7231
7232
7233
7234 026602 000004
7235 026604 012705 026616
7236 026610 004737 040026
7237
7238 026614 000410
7239
7240 026616
7241 026616 143161 007777 177777
7242 026624 177777
7243 026626 170360
7244 026630 047607 047610
7245 026634 000000
7246
7247
7248
7249
7250
7251
7252
7253 026636 000004
7254 026640 012705 026652
7255 026644 004737 040026
7256
7257 026650 000410
7258
7259 026652
7260 026652 143777 177777 177777
7261 026660 177777
7262 026662 100001
7263 026664 047647 047650
7264 026670 000000
7265
7266
7267
7268
7269
7270
7271
7272 026672 000004
7273 026674 012705 026706
7274 026700 004737 040026
7275
7276 026704 000410
7277
7278 026706
7279 026706 152525 052525 177777
7280 026714 000000
7281 026716 000000
7282 026720 047212 047205
7283 026724 000000

```

```

*****
;TEST 444 TEST OF STCDI INSTR, DATA SET SCDI-4
;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

†ST444: SCOPE
MOV #SCDI4,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCDIT ; GO TEST
BR TST445 ;;
SCDI4: ; TEST DATA SET SCDI-4:
.WORD 143161,007777,M1,M1 ; INITIAL FLOAT VALUE
.WORD 170360 ; EXPECTED INTEGER RESULT
.WORD 047607,047610 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
;TEST 445 TEST OF STCDI INSTR, DATA SET SCDI-5
;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

†ST445: SCOPE
MOV #SCDI5,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCDIT ; GO TEST
BR TST446 ;;
SCDI5: ; TEST DATA SET SCDI-5:
.WORD 143777,M1,M1,M1 ; INITIAL FLOAT VALUE
.WORD 100001 ; EXPECTED INTEGER RESULT
.WORD 047647,047650 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
;TEST 446 TEST OF STCDI INSTR, DATA SET SCDI-6
;*
;* INTEGER CONVERSION INTERRUPT ENABLE OFF, ALL OTHERS ON
;* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

†ST446: SCOPE
MOV #SCDI6,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCDIT ; GO TEST
BR TST447 ;;
SCDI6: ; TEST DATA SET SCDI-6:
.WORD 152525,052525,M1,000000 ; INITIAL FLOAT VALUE
.WORD 000000 ; EXPECTED INTEGER RESULT
.WORD 047212,047205 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

7284
7285
7286
7287
7288
7289
7290
7291 026726 000004
7292 026730 012705 026742
7293 026734 004737 040026
7294
7295 026740 000410
7296
7297 026742
7298 026742 140377 177777 177777
7299 026750 052525
7300 026752 177777
7301 026754 047647 047650
7302 026760 000000
7303
7304
7305

```

```

*****
: *TEST 447      TEST OF STCDI INSTR, DATA SET SCDI-7
: *              ALL INTERRUPT ENABLES ON
: *              LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
↑ST447: SCOPE
      MOV      #SCDI7,R5      ; PTR TO TEST DATA SET
      JSR      PC,@#SCDIT    ; GO TEST

      BR      TST450        ;;

SCDI7: ; TEST DATA SET SCDI-7:
      .WORD   140377,M1,M1,052525 ; INITIAL FLOAT VALUE

      .WORD   M1              ; EXPECTED INTEGER RESULT
      .WORD   047647,047650   ; FPS: BEFORE, AFTER
      .WORD   NA              ; FEC AFTER ( 0 = N/A )

```

```

7306
7307
7308
7309
7310
7311 026762 000004
7312 026764 012705 026776
7313 026770 004737 040224
7314
7315 026774 000407
7316
7317 026776
7318 026776 047777 177777
7319 027002 077777 177600
7320 027006 047517 047500
7321 027012 000000
7322
7323
7324
7325
7326
7327
7328
7329 027014 000004
7330 027016 012705 027030
7331 027022 004737 040224
7332
7333 027026 000407
7334
7335 027030
7336 027030 150000 000001
7337 027034 000000 000000
7338 027040 047512 147505
7339 027044 100006
7340
7341
7342
7343
7344
7345
7346
7347 027046 000004
7348 027050 012705 027062
7349 027054 004737 040224
7350
7351 027060 000407
7352
7353 027062
7354 027062 037777 177777
7355 027066 000000 000000
7356 027072 047553 047544
7357 027076 000000
7358
7359
7360
7361

```

```

*****
; *TEST 450 TEST OF STCFL INSTR, DATA SET SCFL-1
; * ALL INTERRUPT ENABLES ON
; * SHORT FLOAT, LONG INTEGER, ROUND MODES
*****
†ST450: SCOPE
MOV #SCFL1,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCFLT ; GO TEST
BR TST451 ;;

SCFL1: ; TEST DATA SET SCFL-1:
.WORD 047777,M1 ; INITIAL FLOAT VALUE
.WORD 077777,177600 ; EXPECTED INTEGER RESULT
.WORD 047517,047500 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
; *TEST 451 TEST OF STCFL INSTR, DATA SET SCFL-2
; * ALL INTERRUPT ENABLES ON
; * SHORT FLOAT, LONG INTEGER, ROUND MODES
*****
†ST451: SCOPE
MOV #SCFL2,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCFLT ; GO TEST
BR TST452 ;;

SCFL2: ; TEST DATA SET SCFL-2:
.WORD 150000,000001 ; INITIAL FLOAT VALUE
.WORD 000000,000000 ; EXPECTED INTEGER RESULT
.WORD 047512,147505 ; FPS: BEFORE, AFTER
.WORD 100006 ; FEC AFTER ( 0 = N/A )

*****
; *TEST 452 TEST OF STCFL INSTR, DATA SET SCFL-3
; * ALL INTERRUPT ENABLES ON
; * SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****
†ST452: SCOPE
MOV #SCFL3,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCFLT ; GO TEST
BR TST453 ;;

SCFL3: ; TEST DATA SET SCFL-3:
.WORD 037777,M1 ; INITIAL FLOAT VALUE
.WORD 000000,000000 ; EXPECTED INTEGER RESULT
.WORD 047553,047544 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
; *TEST 453 TEST OF STCFL INSTR, DATA SET SCFL-4

```

```

7362
7363
7364
7365 027100 000004
7366 027102 012705 027114
7367 027106 004737 040224
7368
7369 027112 000407
7370
7371 027114
7372 027114 000000 000000
7373 027120 000000 000000
7374 027124 047553 047544
7375 027130 000000
7376
7377
7378
7379
7380
7381
7382
7383 027132 000004
7384 027134 012705 027146
7385 027140 004737 040224
7386
7387 027144 000407
7388
7389 027146
7390 027146 147777 177777
7391 027152 100000 000200
7392 027156 047507 047510
7393 027162 000000
7394
7395
7396
7397
7398
7399
7400
7401 027164 000004
7402 027166 012705 027200
7403 027172 004737 040224
7404
7405 027176 000407
7406
7407 027200
7408 027200 040577 177777
7409 027204 000000 000003
7410 027210 047517 047500
7411 027214 000000
7412
7413
7414
7415
7416
7417

```

```

;*
;* ALL INTERRUPT ENABLES ON
;* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
;*****
TST453: SCOPE
MOV #SCFL4,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCFLT ; GO TEST
BR TST454 ;;

SCFL4: ; TEST DATA SET SCFL-4:
.WORD 000000,000000 ; INITIAL FLOAT VALUE
.WORD 000000,000000 ; EXPECTED INTEGER RESULT
.WORD 047553,047544 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

;*****
;*TEST 454 TEST OF STCFL INSTR, DATA SET SCFL-5
;*
;* ALL INTERRUPT ENABLES ON
;* SHORT FLOAT, LONG INTEGER, ROUND MODES
;*****
TST454: SCOPE
MOV #SCFL5,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCFLT ; GO TEST
BR TST455 ;;

SCFL5: ; TEST DATA SET SCFL-5:
.WORD 147777,M1 ; INITIAL FLOAT VALUE
.WORD 100000,000200 ; EXPECTED INTEGER RESULT
.WORD 047507,047510 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

;*****
;*TEST 455 TEST OF STCFL INSTR, DATA SET SCFL-6
;*
;* ALL INTERRUPT ENABLES ON
;* SHORT FLOAT, LONG INTEGER, ROUND MODES
;*****
TST455: SCOPE
MOV #SCFL6,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCFLT ; GO TEST
BR TST456 ;;

SCFL6: ; TEST DATA SET SCFL-6:
.WORD 040577,M1 ; INITIAL FLOAT VALUE
.WORD 000000,000003 ; EXPECTED INTEGER RESULT
.WORD 047517,047500 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

;*****
;*TEST 456 TEST OF STCFL INSTR, DATA SET SCFL-7
;*
;* INTEGER CONVERSION INTERRUPT ENABLE OFF, ALL OTHERS ON
;* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES

```

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 151
T456 TEST OF STCFL INSTR, DATA SET SCFL-7

7418			
7419	027216	000004	
7420	027220	012705	027232
7421	027224	004737	040224
7422			
7423	027230	000407	
7424			
7425	027232		
7426	027232	066666	123456
7427	027236	000000	000000
7428	027242	047152	047145
7429	027246	000000	
7430			
7431			
7432			

```

*****
↑ST456: SCOPE
      MOV      #SCFL7,R5      ; PTR TO TEST DATA SET
      JSR      PC,@#SCFLT    ; GO TEST
      BR       TST457        ;;
SCFL7: ; TEST DATA SET SCFL-7:
      .WORD   066666,123456  ; INITIAL FLOAT VALUE
      .WORD   000000,000000  ; EXPECTED INTEGER RESULT
      .WORD   047152,047145  ; FPS: BEFORE, AFTER
      .WORD   NA              ; FEC AFTER ( 0 = N/A )

```

```

7433
7434
7435
7436
7437
7438 027250 000004
7439 027252 012705 027264
7440 027256 004737 040432
7441
7442 027262 000411
7443
7444 027264
7445 027264 050000 177000 177000
7446 027272 177000
7447 027274 000000 000000
7448 027300 047712 147705
7449 027304 100006
7450
7451
7452
7453
7454
7455
7456
7457 027306 000004
7458 027310 012705 027322
7459 027314 004737 040432
7460
7461 027320 000411
7462
7463 027322
7464 027322 047777 177777 177377
7465 027330 177777
7466 027332 077777 177777
7467 027336 047717 047700
7468 027342 000000
7469
7470
7471
7472
7473
7474
7475
7476 027344 000004
7477 027346 012705 027360
7478 027352 004737 040432
7479
7480 027356 000411
7481
7482 027360
7483 027360 137777 125252 177777
7484 027366 177777
7485 027370 000000 000000
7486 027374 047713 047704
7487 027400 000000
7488

```

```

*****
*TEST 457 TEST OF STCDL INSTR, DATA SET SCDL-1
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
↑ST457: SCOPE
MOV #SCDL1,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCDLT ; GO TEST
BR TST460 ;;
SCDL1: ; TEST DATA SET SCDL-1:
.WORD 050000,177000,177000,177000 ; INITIAL FLOAT VALUE
.WORD 000000,000000 ; EXPECTED INTEGER RESULT
.WORD 047712,147705 ; FPS: BEFORE, AFTER
.WORD 100006 ; FEC AFTER ( 0 = N/A )

*****
*TEST 460 TEST OF STCDL INSTR, DATA SET SCDL-2
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
↑ST460: SCOPE
MOV #SCDL2,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCDLT ; GO TEST
BR TST461 ;;
SCDL2: ; TEST DATA SET SCDL-2:
.WORD 047777,M1,177377,M1 ; INITIAL FLOAT VALUE
.WORD 077777,M1 ; EXPECTED INTEGER RESULT
.WORD 047717,047700 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

*****
*TEST 461 TEST OF STCDL INSTR, DATA SET SCDL-3
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
↑ST461: SCOPE
MOV #SCDL3,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCDLT ; GO TEST
BR TST462 ;;
SCDL3: ; TEST DATA SET SCDL-3:
.WORD 137777,125252,M1,M1 ; INITIAL FLOAT VALUE
.WORD 000000,000000 ; EXPECTED INTEGER RESULT
.WORD 047713,047704 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```



```

7489
7490
7491
7492
7493
7494
7495 027402 000004
7496 027404 012705 027416
7497 027410 004737 040432
7498
7499 027414 000411
7500
7501 027416
7502 027416 147777 177777 177777
7503 027424 177777
7504 027426 100000 000001
7505 027432 047707 047710
7506 027436 000000
7507
7508
7509
7510
7511
7512
7513
7514 027440 000004
7515 027442 012705 027454
7516 027446 004737 040432
7517
7518 027452 000411
7519
7520 027454
7521 027454 047160 170360 177777
7522 027462 177777
7523 027464 007417 007417
7524 027470 047757 047740
7525 027474 000000
7526
7527
7528
7529
7530
7531
7532
7533 027476 000004
7534 027500 012705 027512
7535 027504 004737 040432
7536
7537 027510 000411
7538
7539 027512
7540 027512 000177 177777 125252
7541 027520 101010
7542 027522 000000 000000
7543 027526 047713 047704
7544 027532 000000

```

```

*****
*TEST 462 TEST OF STCDL INSTR, DATA SET SCDL-4
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

↑ST462: SCOPE
MOV #SCDL4,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCDLT ; GO TEST
BR TST463 ;;

```

```

SCDL4: ; TEST DATA SET SCDL-4:
.WORD 147777,M1,M1,M1 ; INITIAL FLOAT VALUE
.WORD 100000,000001 ; EXPECTED INTEGER RESULT
.WORD 047707,047710 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 463 TEST OF STCDL INSTR, DATA SET SCDL-5
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

↑ST463: SCOPE
MOV #SCDL5,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCDLT ; GO TEST
BR TST464 ;;

```

```

SCDL5: ; TEST DATA SET SCDL-5:
.WORD 047160,170360,M1,M1 ; INITIAL FLOAT VALUE
.WORD 007417,007417 ; EXPECTED INTEGER RESULT
.WORD 047757,047740 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 464 TEST OF STCDL INSTR, DATA SET SCDL-6
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

↑ST464: SCOPE
MOV #SCDL6,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCDLT ; GO TEST
BR TST465 ;;

```

```

SCDL6: ; TEST DATA SET SCDL-6:
.WORD 000177,M1,125252,101010 ; INITIAL FLOAT VALUE
.WORD 000000,000000 ; EXPECTED INTEGER RESULT
.WORD 047713,047704 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

7545
7546
7547
7548
7549
7550
7551
7552
7553
7554
7555
7556
7557
7558
7559
7560
7561
7562
7563
7564
7565
7565

027534 000004
027536 012705 027550
027542 004737 040432
027546 000411
027550
027550 062141 125252 052525
027556 125252
027560 000000 000000
027564 047312 047305
027570 000000

```
*****  
*TEST 465 TEST OF STCDL INSTR, DATA SET SCDL-7  
* INTEGER CONVERSION INTERRUPT ENABLE OFF, ALL OTHERS ON  
* LONG FLOAT, LONG INTEGER, ROUND MODES  
*****  
↑ST465: SCOPE  
MOV #SCDL7,R5 ; PTR TO TEST DATA SET  
JSR PC,@#SCDLT ; GO TEST  
BR TST466 ;;  
SCDL7: ; TEST DATA SET SCDL-7:  
.WORD 062141,125252,052525,125252 ; INITIAL FLOAT VALUE  
.WORD 000000,000000 ; EXPECTED INTEGER RESULT  
.WORD 047312,047305 ; FPS: BEFORE AFTER  
.WORD NA ; FEC AFTER ( 0 = N/A )
```



```

7567
7568
7569
7570
7571
7572 027572 000004
7573 027574 012705 027606
7574 027600 004737 040640
7575
7576 027604 000410
7577
7578 027606
7579 027606 020177 177777
7580 027612 000377 177777
7581 027616 000201
7582 027620 047555 147542
7583 027624 100010
7584
7585
7586
7587
7588
7589
7590
7591 027626 000004
7592 027630 012705 027642
7593 027634 004737 040640
7594
7595 027640 000410
7596
7597 027642
7598 027642 120000 000000
7599 027646 100000 000000
7600 027652 000200
7601 027654 047501 147516
7602 027660 100010
7603
7604
7605
7606
7607
7608
7609
7610 027662 000004
7611 027664 012705 027676
7612 027670 004737 040640
7613
7614 027674 000410
7615
7616 027676
7617 027676 020125 052525
7618 027702 077725 052525
7619 027706 000177
7620 027710 047457 047440
7621 027714 000000
7622

```

```

*****
: TEST 466 TEST OF LDEXP/F INSTR, DATA SET LEXF-1
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
: *****
TST466: SCOPE
MOV #LEXF1,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXFT ; GO TEST
BR TST467 ;;

LEXF1: ; TEST DATA SET LEXF-1:
.WORD 020177,M1 ; INITIAL AC FLOAT NUMBER
.WORD 000377,M1 ; EXPECTED FLOAT RESULT
.WORD 201 ; EXPONENT TO BE LOADED
.WORD 047555,147542 ; FPS: BEFORE, AFTER
.WORD 100010 ; FEC AFTER ( 0 = N/A )

*****
: TEST 467 TEST OF LDEXP/F INSTR, DATA SET LEXF-2
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, LONG INTEGER, ROUND MODES
: *****
TST467: SCOPE
MOV #LEXF2,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXFT ; GO TEST
BR TST470 ;;

LEXF2: ; TEST DATA SET LEXF-2:
.WORD 120000,0 ; INITIAL AC FLOAT NUMBER
.WORD 100000,0 ; EXPECTED FLOAT RESULT
.WORD 200 ; EXPONENT TO BE LOADED
.WORD 047501,147516 ; FPS: BEFORE, AFTER
.WORD 100010 ; FEC AFTER ( 0 = N/A )

*****
: TEST 470 TEST OF LDEXP/F INSTR, DATA SET LEXF-3
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
: *****
TST470: SCOPE
MOV #LEXF3,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXFT ; GO TEST
BR TST471 ;;

LEXF3: ; TEST DATA SET LEXF-3:
.WORD 020125,ALTP ; INITIAL AC FLOAT NUMBER
.WORD 077725,ALTP ; EXPECTED FLOAT RESULT
.WORD 177 ; EXPONENT TO BE LOADED
.WORD 047457,047440 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

7623
7624
7625
7626
7627
7628
7629 027716 000004
7630 027720 012705 027732
7631 027724 004737 040640
7632
7633 027730 000410
7634
7635 027732
7636 027732 120052 125252
7637 027736 160052 125252
7638 027742 000100
7639 027744 047407 047410
7640 027750 000000
7641
7642
7643
7644
7645
7646
7647
7648 027752 000004
7649 027754 012705 027766
7650 027760 004737 040640
7651
7652 027764 000410
7653
7654 027766
7655 027766 020017 007417
7656 027772 040217 007417
7657 027776 000001
7658 030000 047557 047540
7659 030004 000000
7660
7661
7662
7663
7664
7665
7666
7667 030006 000004
7668 030010 012705 030022
7669 030014 004737 040640
7670
7671 030020 000410
7672
7673 030022
7674 030022 120160 170360
7675 030026 140160 170360
7676 030032 000000
7677 030034 047507 047510
7678 030040 000000

```

```

*****
: TEST 471 TEST OF LDEXP/F INSTR, DATA SET LEXF-4
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, SHORT INTEGER, ROUND MODES
: *****

```

```

TST471: SCOPE
MOV #LEXF4,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXFT ; GO TEST
BR TST472 ;;

```

```

LEXF4: ; TEST DATA SET LEXF-4:
.WORD 120052,ALTN ; INITIAL AC FLOAT NUMBER
.WORD 160052,ALTN ; EXPECTED FLOAT RESULT
.WORD 100 ; EXPONENT TO BE LOADED
.WORD 047407,047410 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
: TEST 472 TEST OF LDEXP/F INSTR, DATA SET LEXF-5
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
: *****

```

```

TST472: SCOPE
MOV #LEXF5,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXFT ; GO TEST
BR TST473 ;;

```

```

LEXF5: ; TEST DATA SET LEXF-5:
.WORD 020017,ALTN ; INITIAL AC FLOAT NUMBER
.WORD 040217,ALTN ; EXPECTED FLOAT RESULT
.WORD 1 ; EXPONENT TO BE LOADED
.WORD 047557,047540 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
: TEST 473 TEST OF LDEXP/F INSTR, DATA SET LEXF-6
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, LONG INTEGER, ROUND MODES
: *****

```

```

TST473: SCOPE
MOV #LEXF6,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXFT ; GO TEST
BR TST474 ;;

```

```

LEXF6: ; TEST DATA SET LEXF-6:
.WORD 120160,ALTN ; INITIAL AC FLOAT NUMBER
.WORD 140160,ALTN ; EXPECTED FLOAT RESULT
.WORD 0 ; EXPONENT TO BE LOADED
.WORD 047507,047510 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

7679
7680
7681
7682
7683
7684
7685
7686 030042 000004
7687 030044 012705 030056
7688 030050 004737 040640
7689
7690 030054 000410
7691
7692 030056
7693 030056 020177 177777
7694 030062 037777 177777
7695 030066 177777
7696 030070 047457 047440
7697 030074 000000
7698
7699
7700
7701
7702
7703
7704
7705 030076 000004
7706 030100 012705 030112
7707 030104 004737 040640
7708
7709 030110 000410
7710
7711 030112
7712 030112 120000 000000
7713 030116 120000 000000
7714 030122 177700
7715 030124 047407 047410
7716 030130 000000
7717
7718
7719
7720
7721
7722
7723
7724 030132 000004
7725 030134 012705 030146
7726 030140 004737 040640
7727
7728 030144 000410
7729
7730 030146
7731 030146 020125 052525
7732 030152 000325 052525
7733 030156 177601
7734 030160 047557 047540

```

```

*****
*TEST 474 TEST OF LDEXP/F INSTR, DATA SET LEXF-7
*
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

†ST474: SCOPE
MOV #LEXF7,R5 ; PTR TO TEST DATA SET
JSR PC, @LEXF7 ; GO TEST
BR TST475 ;;

```

```

LEXF7: ; TEST DATA SET LEXF-7:
.WORD 020177,M1 ; INITIAL AC FLOAT NUMBER
.WORD 037777,M1 ; EXPECTED FLOAT RESULT
.WORD -1 ; EXPONENT TO BE LOADED
.WORD 047457,047440 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 475 TEST OF LDEXP/F INSTR, DATA SET LEXF-10
*
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

†ST475: SCOPE
MOV #LEXF10,R5 ; PTR TO TEST DATA SET
JSR PC, @LEXF10 ; GO TEST
BR TST476 ;;

```

```

LEXF10: ; TEST DATA SET LEXF-10:
.WORD 120000,0 ; INITIAL AC FLOAT NUMBER
.WORD 120000,0 ; EXPECTED FLOAT RESULT
.WORD -100 ; EXPONENT TO BE LOADED
.WORD 047407,047410 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 476 TEST OF LDEXP/F INSTR, DATA SET LEXF-11
*
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

†ST476: SCOPE
MOV #LEXF11,R5 ; PTR TO TEST DATA SET
JSR PC, @LEXF11 ; GO TEST
BR TST477 ;;

```

```

LEXF11: ; TEST DATA SET LEXF-11:
.WORD 020125,ALTP ; INITIAL AC FLOAT NUMBER
.WORD 000325,ALTP ; EXPECTED FLOAT RESULT
.WORD -177 ; EXPONENT TO BE LOADED
.WORD 047557,047540 ; FPS: BEFORE, AFTER

```

7735 030164 000000 .WORD NA ; FEC AFTER (0 = N/A)

7736
7737
7738 ; *****
7739 ; *TEST 477 TEST OF LDEXP/F INSTR, DATA SET LEXF-12
7740 ; * ALL INTERRUPT ENABLES ON
7741 ; * SHORT FLOAT, LONG INTEGER, ROUND MODES
7742 ; *****

7743 030166 000004
7744 030170 012705 030202
7745 030174 004737 040640
7746
7747 030200 000410
7748
7749
7750 030202 120052 125252
7751 030206 100052 125252
7752 030212 177600
7753 030214 047503 147514
7754 03022J 100012

↑ST477: SCOPE
MOV #LEXF12,R5 ; PTR TO TEST DATA SET
JSR PC,2#LEXFT ; GO TEST
BR TST500 ; ;

LEXF12: ; TEST DATA SET LEXF-12:
.WORD 120052,ALTN ; INITIAL AC FLOAT NUMBER
.WORD 100052,ALTN ; EXPECTED FLOAT RESULT
.WORD -200 ; EXPONENT TO BE LOADED
.WORD 047503,147514 ; FPS: BEFORE, AFTER
.WORD 100012 ; FEC AFTER (0 = N/A)

7755
7756
7757 ; *****
7758 ; *TEST 500 TEST OF LDEXP/F INSTR, DATA SET LEXF-13
7759 ; * ALL INTERRUPT ENABLES ON
7760 ; * SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
7761 ; *****

7762 030222 000004
7763 030224 012705 030236
7764 030230 004737 040640
7765
7766 030234 000410
7767
7768
7769 030236 020017 007417
7770 030242 077617 007417
7771 030246 177577
7772 030250 047457 147440
7773 030254 100012

↑ST500: SCOPE
MOV #LEXF13,R5 ; PTR TO TEST DATA SET
JSR PC,2#LEXFT ; GO TEST
BR TST501 ; ;

LEXF13: ; TEST DATA SET LEXF-13:
.WORD 020017,ALT4P ; INITIAL AC FLOAT NUMBER
.WORD 077617,ALT4P ; EXPECTED FLOAT RESULT
.WORD -201 ; EXPONENT TO BE LOADED
.WORD 047457,147440 ; FPS: BEFORE, AFTER
.WORD 100012 ; FEC AFTER (0 = N/A)

7774
7775
7776 ; *****
7777 ; *TEST 501 TEST OF LDEXP/F INSTR, DATA SET LEXF-14
7778 ; * ALL INTERRUPT ENABLES ON
7779 ; * SHORT FLOAT, LONG INTEGER, ROUND MODES
7780 ; *****

7781 030256 000004
7782 030260 012705 030272
7783 030264 004737 040640
7784
7785 030270 000410
7786
7787
7788 030272 120160 170360
7789 030276 177560 170360
7790 030302 177576

↑ST501: SCOPE
MOV #LEXF14,R5 ; PTR TO TEST DATA SET
JSR PC,2#LEXFT ; GO TEST
BR TST502 ; ;

LEXF14: ; TEST DATA SET LEXF-14:
.WORD 120160,ALT4N ; INITIAL AC FLOAT NUMBER
.WORD 177560,ALT4N ; EXPECTED FLOAT RESULT
.WORD -202 ; EXPONENT TO BE LOADED

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 159
T501 TEST OF LDEXP/F INSTR, DATA SET LEXF-14

7791 030304 047507 147510
7792 030310 100012

.WORD 047507,147510 ; FPS: BEFORE, AFTER
.WORD 100012 ; FEC AFTER (0 = N/A)

7793
7794
7795
7796
7797
7798
7799

*TEST 502 TEST OF LDEXP/F INSTR, DATA SET LEXF-15
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES

7800 030312 000004
7801 030314 012705 030326
7802 030320 004737 040640
7803
7804 030324 000410
7805

TST502: SCOPE
MOV #LEXF15,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXF15 ; GO TEST

BR TST503 ;;

7806 030326
7807 030326 020177 177777
7808 030332 077377 177777
7809 030336 177575
7810 030340 047457 147440
7811 030344 100012

LEXF15: ; TEST DATA SET LEXF-15:
.WORD 020177,M1 ; INITIAL AC FLOAT NUMBER
.WORD 077377,M1 ; EXPECTED FLOAT RESULT
.WORD -203 ; EXPONENT TO BE LOADED
.WORD 047457,147440 ; FPS: BEFORE, AFTER
.WORD 100012 ; FEC AFTER (0 = N/A)

7812
7813
7814
7815
7816
7817
7818

*TEST 503 TEST OF LDEXP/F INSTR, DATA SET LEXF-16
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES

7819 030346 000004
7820 030350 012705 030362
7821 030354 004737 040640
7822
7823 030360 000410
7824

TST503: SCOPE
MOV #LEXF16,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXF16 ; GO TEST

BR TST504 ;;

7825 030362
7826 030362 142000 000000
7827 030366 140000 000000
7828 030372 000000
7829 030374 047547 047550
7830 030400 000000

LEXF16: ; TEST DATA SET LEXF-16:
.WORD 142000,0 ; INITIAL AC FLOAT NUMBER
.WORD 140000,0 ; EXPECTED FLOAT RESULT
.WORD 0 ; EXPONENT TO BE LOADED
.WORD 047547,047550 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

7831
7832
7833
7834
7835
7836
7837

*TEST 504 TEST OF LDEXP/F INSTR, DATA SET LEXF-17
* OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES

7838 030402 000004
7839 030404 012705 030416
7840 030410 004737 040640
7841
7842 030414 000410
7843

TST504: SCOPE
MOV #LEXF17,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXF17 ; GO TEST

BR TST505 ;;

7844 030416
7845 030416 020177 177777
7846 030422 000000 000000

LEXF17: ; TEST DATA SET LEXF-17:
.WORD 020177,M1 ; INITIAL AC FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT

FPU ADVANCED INSTR TESTS
 DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 160
 T504 TEST OF LDEXP/F INSTR, DATA SET LEXF-17

7847 030426 000201
 7848 030430 046551 046546
 7849 030434 000000

.WORD 201 ; EXPONENT TO BE LOADED
 .WORD 046551,046546 ; FPS: BEFORE, AFTER
 .WORD NA ; FEC AFTER (0 = N/A)

7850
 7851
 7852
 7853
 7854
 7855
 7856
 7857 030436 000004
 7858 030440 012705 030452
 7859 030444 004737 040640
 7860
 7861 030450 000410
 7862
 7863 030452
 7864 030452 120000 000000
 7865 030456 000000 000000
 7866 030462 000200
 7867 030464 046511 046506
 7868 030470 000000
 7869
 7870
 7871
 7872
 7873
 7874
 7875
 7876 030472 000004
 7877 030474 012705 030506
 7878 030500 004737 040640
 7879
 7880 030504 000410
 7881
 7882 030506
 7883 030506 120052 125252
 7884 030512 000000 000000
 7885 030516 177600
 7886 030520 045513 045504
 7887 030524 000000
 7888
 7889
 7890
 7891
 7892
 7893
 7894
 7895 030526 000004
 7896 030530 012705 030542
 7897 030534 004737 040640
 7898
 7899 030540 000410
 7900
 7901 030542
 7902 030542 020017 007417

```

*****
; TEST 505 TEST OF LDEXP/F INSTR, DATA SET LEXF-20
; * OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
; * SHORT FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

†T505: SCOPE
MOV #LEXF20,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXFT ; GO TEST

BR T506 ;;

```

```

LEXF20: ; TEST DATA SET LEXF-20:
.WORD 120000,0 ; INITIAL AC FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD 200 ; EXPONENT TO BE LOADED
.WORD 046511,046506 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
; TEST 506 TEST OF LDEXP/F INSTR, DATA SET LEXF-21
; * UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
; * SHORT FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

†T506: SCOPE
MOV #LEXF21,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXFT ; GO TEST

BR T507 ;;

```

```

LEXF21: ; TEST DATA SET LEXF-21:
.WORD 120052,ALTN ; INITIAL AC FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD -200 ; EXPONENT TO BE LOADED
.WORD 045513,045504 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
; TEST 507 TEST OF LDEXP/F INSTR, DATA SET LEXF-22
; * UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
; * SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

†T507: SCOPE
MOV #LEXF22,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXFT ; GO TEST

BR T510 ;;

```

```

LEXF22: ; TEST DATA SET LEXF-22:
.WORD 020017,ALT4P ; INITIAL AC FLOAT NUMBER

```


FPU ADVANCED INSTR TESTS
DQFP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 161
T507 TEST OF LDEXP/F INSTR, DATA SET LEXF-22

7903 030546 000000 000000
7904 030552 177577
7905 030554 045453 045444
7906 030560 000000

.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD -201 ; EXPONENT TO BE LOADED
.WORD 045453,045444 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER (0 = N/A)

7907
7908
7909
7910
7911
7912
7913

*TEST 510 TEST OF LDEXP/F INSTR, DATA SET LEXF-23
* UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES

7914 030562 000004
7915 030564 012705 030576
7916 030570 004737 040640
7917
7918 030574 000410
7919

TST510: SCOPE
MOV #LEXF23,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXF24 ; GO TEST
BR TST511 ;;

7920 030576
7921 030576 120160 170360
7922 030602 000000 000000
7923 030606 177576
7924 030610 045513 045504
7925 030614 000000

LEXF23: ; TEST DATA SET LEXF-23:
.WORD 120160,ALT4N ; INITIAL AC FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD -202 ; EXPONENT TO BE LOADED
.WORD 045513,045504 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER (0 = N/A)

7926
7927
7928
7929
7930
7931
7932

*TEST 511 TEST OF LDEXP/F INSTR, DATA SET LEXF-24
* UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES

7933 030616 000004
7934 030620 012705 030632
7935 030624 004737 040640
7936
7937 030630 000410
7938

TST511: SCOPE
MOV #LEXF24,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXF24 ; GO TEST
BR TST512 ;;

7939 030632
7940 030632 020177 177777
7941 030636 000000 000000
7942 030642 177575
7943 030644 045453 045444
7944 030650 000000

LEXF24: ; TEST DATA SET LEXF-24:
.WORD 020177,M1 ; INITIAL AC FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD -203 ; EXPONENT TO BE LOADED
.WORD 045453,045444 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER (0 = N/A)

7945
7946
7947

E13

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 162
T512 TEST OF LDEXP/D INSTR, DATA SET LEXD-1

```

7948
7949
7950
7951
7952
7953 030652 000004
7954 030654 012705 030666
7955 030660 004737 041010
7956
7957 030664 000414
7958
7959 030666
7960 030666 152325 052525 052525
7961 030674 052525
7962 030676 100325 052525 052525
7963 030704 052525
7964 030706 000201
7965 030710 047645 147652
7966 030714 100010
7967
7968
7969
7970
7971
7972
7973
7974 030716 000004
7975 030720 012705 030732
7976 030724 004737 041010
7977
7978 030730 000414
7979
7980 030732
7981 030732 052377 177777 177777
7982 030740 177777
7983 030742 000177 177777 177777
7984 030750 177777
7985 030752 000200
7986 030754 047711 147706
7987 030760 100010
7988
7989
7990
7991
7992
7993
7994
7995 030762 000004
7996 030764 012705 030776
7997 030770 004737 041010
7998
7999 030774 000414
8000
8001 030776
8002 030776 152360 170360 170360
8003 031004 170360

```

```

*****
: *TEST 512 TEST OF LDEXP/D INSTR, DATA SET LEXD-1
: * ALL INTERRUPT ENABLES ON
: * LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
: *****
↑T512: SCOPE
MOV #LEXD1,R5 ; PTR TO TEST DATA SET
JSR PC,2#LEXDT ; GO TEST
BR T513 ;;
LEXD1: ; TEST DATA SET LEXD-1:
.WORD 152325,ALTP,ALTP,ALTP ; INITIAL AC FLOAT NUMBER
.WORD 100325,ALTP,ALTP,ALTP ; EXPECTED FLOAT RESULT
.WORD 201 ; EXPONENT TO BE LOADED
.WORD 047645,147652 ; FPS: BEFORE, AFTER
.WORD 100010 ; FEC AFTER ( 0 = N/A )

```

```

*****
: *TEST 513 TEST OF LDEXP/D INSTR, DATA SET LEXD-2
: * ALL INTERRUPT ENABLES ON
: * LONG FLOAT, LONG INTEGER, ROUND MODES
: *****
↑T513: SCOPE
MOV #LEXD2,R5 ; PTR TO TEST DATA SET
JSR PC,2#LEXDT ; GO TEST
BR T514 ;;
LEXD2: ; TEST DATA SET LEXD-2:
.WORD 052377,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD 000177,M1,M1,M1 ; EXPECTED FLOAT RESULT
.WORD 200 ; EXPONENT TO BE LOADED
.WORD 047711,147706 ; FPS: BEFORE, AFTER
.WORD 100010 ; FEC AFTER ( 0 = N/A )

```

```

*****
: *TEST 514 TEST OF LDEXP/D INSTR, DATA SET LEXD-3
: * ALL INTERRUPT ENABLES ON
: * LONG FLOAT, SHORT INTEGER, ROUND MODES
: *****
↑T514: SCOPE
MOV #LEXD3,R5 ; PTR TO TEST DATA SET
JSR PC,2#LEXDT ; GO TEST
BR T515 ;;
LEXD3: ; TEST DATA SET LEXD-3:
.WORD 152360,ALT4N,ALT4N,ALT4N ; INITIAL AC FLOAT NUMBER

```

F13

FPU ADVANCED INSTR TESTS
DQFP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 163
T514 TEST OF LDEXP/D INSTR, DATA SET LEXD-3

8004	031006	177760	170360	170360	.WORD	177760,ALT4N,ALT4N,ALT4N	; EXPECTED FLOAT RESULT
8005	031014	170360					
8006	031016	000177			.WORD	177	; EXPONENT TO BE LOADED
8007	031020	047607	047610		.WORD	047607,047610	; FPS: BEFORE, AFTER
8008	031024	000000			.WORD	NA	; FEC AFTER (0 = N/A)

```

8009
8010
8011
8012
8013
8014
8015
*****
; TEST 515 TEST OF LDEXP/D INSTR, DATA SET LEXD-4
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

†ST515: SCOPE
MOV #LEXD4,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXDT ; GO TEST
BR TST516 ;;

```

```

LEXD4: ; TEST DATA SET LEXD-4:
.WORD 052200,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD 060000,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 100 ; EXPONENT TO BE LOADED
.WORD 047757,047740 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

8030
8031
8032
8033
8034
8035
8036
*****
; TEST 516 TEST OF LDEXP/D INSTR, DATA SET LEXD-5
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

†ST516: SCOPE
MOV #LEXD5,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXDT ; GO TEST
BR TST517 ;;

```

```

LEXD5: ; TEST DATA SET LEXD-5:
.WORD 152252,ALTN,ALTN,ALTN ; INITIAL AC FLOAT NUMBER
.WORD 140252,ALTN,ALTN,ALTN ; EXPECTED FLOAT RESULT
.WORD 1 ; EXPONENT TO BE LOADED
.WORD 047647,047650 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

8051
8052
8053
8054
8055
8056
8057
*****
; TEST 517 TEST OF LDEXP/D INSTR, DATA SET LEXD-6
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

†ST517: SCOPE
MOV #LEXD6,R5 ; PTR TO TEST DATA SET

```

8058	031136	000004					
8059	031140	012705	031152				

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 164
TS17 TEST OF LDEXP/D INSTR, DATA SET LEXD-6

```

8060 031144 004737 041010 JSR PC,2#LEXDT ; GO TEST
8061
8062 031150 000414 BR TST520 ;;
8063
8064 031152 LEXD6: ; TEST DATA SET LEXD-6:
8065 031152 052217 007417 007417 .WORD 052217,ALT4P,ALT4P,ALT4P ; INITIAL AC FLOAT NUMBER
8066 031160 007417
8067 031162 040017 007417 007417 .WORD 040017,ALT4P,ALT4P,ALT4P ; EXPECTED FLOAT RESULT
8068 031170 007417
8069 031172 000000 .WORD 0 ; EXPONENT TO BE LOADED
8070 031174 047717 047700 .WORD 047717,047700 ; FPS: BEFORE, AFTER
8071 031200 000000 .WORD NA ; FEC AFTER ( 0 = N/A )
8072
8073
8074
8075
8076
8077
8078

```

```

*****
*TEST 520 TEST OF LDEXP/D INSTR, DATA SET LEXD-7
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

8079 031202 000004 TST520: SCOPE
8080 031204 012705 031216 MOV #LEXD7,R5 ; PTR TO TEST DATA SET
8081 031210 004737 041010 JSR PC,2#LEXDT ; GO TEST
8082
8083 031214 000414 BR TST521 ;;
8084
8085 031216 LEXD7: ; TEST DATA SET LEXD-7:
8086 031216 152325 052525 052525 .WORD 152325,ALTP,ALTP,ALTP ; INITIAL AC FLOAT NUMBER
8087 031224 052525
8088 031226 137725 052525 052525 .WORD 137725,ALTP,ALTP,ALTP ; EXPECTED FLOAT RESULT
8089 031234 052525
8090 031236 177777 .WORD -1 ; EXPONENT TO BE LOADED
8091 031240 047607 047610 .WORD 047607,047610 ; FPS: BEFORE, AFTER
8092 031244 000000 .WORD NA ; FEC AFTER ( 0 = N/A )
8093
8094
8095
8096
8097
8098
8099

```

```

*****
*TEST 521 TEST OF LDEXP/D INSTR, DATA SET LEXD-10
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

8100 031246 000004 TST521: SCOPE
8101 031250 012705 031262 MOV #LEXD10,R5 ; PTR TO TEST DATA SET
8102 031254 004737 041010 JSR PC,2#LEXDT ; GO TEST
8103
8104 031260 000414 BR TST522 ;;
8105
8106 031262 LEXD10: ; TEST DATA SET LEXD-10:
8107 031262 052377 177777 177777 .WORD 052377,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
8108 031270 177777
8109 031272 020177 177777 177777 .WORD 020177,M1,M1,M1 ; EXPECTED FLOAT RESULT
8110 031300 177777
8111 031302 177700 .WORD -100 ; EXPONENT TO BE LOADED
8112 031304 047757 047740 .WORD 047757,047740 ; FPS: BEFORE, AFTER
8113 031310 000000 .WORD NA ; FEC AFTER ( 0 = N/A )
8114
8115

```

```

8116
8117
8118
8119
8120
8121 031312 000004
8122 031314 012705 031326
8123 031320 004737 041010
8124
8125 031324 000414
8126
8127 031326
8128 031326 152360 170360 170360
8129 031334 170360
8130 031336 100360 170360 170360
8131 031344 170360
8132 031346 177601
8133 031350 047647 047650
8134 031354 000000
8135
8136
8137
8138
8139
8140
8141
8142 031356 000004
8143 031360 012705 031372
8144 031364 004737 041010
8145
8146 031370 000414
8147
8148 031372
8149 031372 052200 000000 000000
8150 031400 000000
8151 031402 000000 000000 000000
8152 031410 000000
8153 031412 177600
8154 031414 047713 147704
8155 031420 100012
8156
8157
8158
8159
8160
8161
8162
8163 031422 000004
8164 031424 012705 031436
8165 031430 004737 041010
8166
8167 031434 000414
8168
8169 031436
8170 031436 152252 125252 125252
8171 031444 125252

```

```

*****
: TEST 522 TEST OF LDEXP/D INSTR, DATA SET LEXD-11
: * ALL INTERRUPT ENABLES ON
: * LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
: *****
TST522: SCOPE
MOV #LEXD11,R5 ; PTR TO TEST DATA SET
JSR PC,@#LEXDT ; GO TEST
BR TST523 ;;

LEXD11: ; TEST DATA SET LEXD-11:
.WORD 152360,ALT4N,ALT4N,ALT4N ; INITIAL AC FLOAT NUMBER
.WORD 100360,ALT4N,ALT4N,ALT4N ; EXPECTED FLOAT RESULT
.WORD -177 ; EXPONENT TO BE LOADED
.WORD 047647,047650 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
: TEST 523 TEST OF LDEXP/D INSTR, DATA SET LEXD-12
: * ALL INTERRUPT ENABLES ON
: * LONG FLOAT, LONG INTEGER, ROUND MODES
: *****
TST523: SCOPE
MOV #LEXD12,R5 ; PTR TO TEST DATA SET
JSR PC,@#LEXDT ; GO TEST
BR TST524 ;;

LEXD12: ; TEST DATA SET LEXD-12:
.WORD 052200,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
.WORD -200 ; EXPONENT TO BE LOADED
.WORD 047713,147704 ; FPS: BEFORE, AFTER
.WORD 100012 ; FEC AFTER ( 0 = N/A )

```

```

*****
: TEST 524 TEST OF LDEXP/D INSTR, DATA SET LEXD-13
: * ALL INTERRUPT ENABLES ON
: * LONG FLOAT, SHORT INTEGER, ROUND MODES
: *****
TST524: SCOPE
MOV #LEXD13,R5 ; PTR TO TEST DATA SET
JSR PC,@#LEXDT ; GO TEST
BR TST525 ;;

LEXD13: ; TEST DATA SET LEXD-13:
.WORD 152252,ALT4N,ALT4N,ALT4N ; INITIAL AC FLOAT NUMBER

```

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 166
T524 TEST OF LDEXP/D INSTR, DATA SET LEXD-13

8172	031446	177652	125252	125252	.WORD	177652,ALTN,ALTN,ALTN	; EXPECTED FLOAT RESULT
8173	031454	125252					
8174	031456	177577			.WORD	-201	; EXPONENT TO BE LOADED
8175	031460	047607	147610		.WORD	047607,147610	; FPS: BEFORE, AFTER
8176	031464	100012			.WORD	100012	; FEC AFTER (0 = N/A)

8177
8178
8179

```

*****
: TEST 525 TEST OF LDEXP/D INSTR, DATA SET LEXD-14
: *
: * ALL INTERRUPT ENABLES ON
: * LONG FLOAT, LONG INTEGER, TRUNCATE MODES
: *
*****

```

```

↑ST525: SCOPE
MOV #LEXD14,R5 ; PTR TO TEST DATA SET
JSR PC,@#LEXDT ; GO TEST
BR TST526 ;;

```

8184	031466	000004			LEXD14: ; TEST DATA SET LEXD-14:		
8185	031470	012705	031502		.WORD	052217,ALTP,ALTP,ALTP	; INITIAL AC FLOAT NUMBER
8186	031474	004737	041010				
8187							
8188	031500	000414					
8189							
8190	031502						
8191	031502	052217	007417	007417	.WORD	077417,ALTP,ALTP,ALTP	; EXPECTED FLOAT RESULT
8192	031510	007417					
8193	031512	077417	007417	007417	.WORD	-202	; EXPONENT TO BE LOADED
8194	031520	007417			.WORD	047757,147740	; FPS: BEFORE, AFTER
8195	031522	177576			.WORD	100012	; FEC AFTER (0 = N/A)
8196	031524	047757	147740				
8197	031530	100012					

8198
8199
8200

```

*****
: TEST 526 TEST OF LDEXP/D INSTR, DATA SET LEXD-15
: *
: * ALL INTERRUPT ENABLES ON
: * LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
: *
*****

```

```

↑ST526: SCOPE
MOV #LEXD15,R5 ; PTR TO TEST DATA SET
JSR PC,@#LEXDT ; GO TEST
BR TST527 ;;

```

8201					LEXD15: ; TEST DATA SET LEXD-15:		
8202					.WORD	152325,ALTP,ALTP,ALTP	; INITIAL AC FLOAT NUMBER
8203							
8204							
8205	031532	000004					
8206	031534	012705	031546				
8207	031540	004737	041010				
8208							
8209	031544	000414					
8210							
8211	031546						
8212	031546	152325	052525	052525	.WORD	177325,ALTP,ALTP,ALTP	; EXPECTED FLOAT RESULT
8213	031554	052525					
8214	031556	177325	052525	052525	.WORD	-203	; EXPONENT TO BE LOADED
8215	031564	052525			.WORD	047647,147650	; FPS: BEFORE, AFTER
8216	031566	177575			.WORD	100012	; FEC AFTER (0 = N/A)
8217	031570	047647	147650				
8218	031574	100012					

8219
8220

```

*****
: TEST 527 TEST OF LDEXP/D INSTR, DATA SET LEXD-16
: *
: * ALL INTERRUPT ENABLES ON
: * LONG FLOAT, LONG INTEGER, ROUND MODES
: *
*****

```

```

↑ST527: SCOPE
MOV #LEXD16,R5 ; PTR TO TEST DATA SET

```

8221
8222
8223
8224
8225
8226 031576 000004
8227 031600 012705 031612

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 167
T527 TEST OF LDEXP/D INSTR, DATA SET LEXD-16

8228 031604 004737 041010
8229
8230 031610 000414
8231
8232 031612
8233 031612 177600 000000 000000
8234 031620 000000
8235 031622 140000 000000 000000
8236 031630 000000
8237 031632 000000
8238 031634 047707 047710
8239 031640 000000
8240
8241
8242
8243
8244
8245
8246

JSR PC, @#LEXDT ; GO TEST
BR TST530 ; ;
LEXD16: ; TEST DATA SET LEXD-16:
.WORD 177600,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD 140000,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 0 ; EXPONENT TO BE LOADED
.WORD 047707,047710 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

: *TEST 530 TEST OF LDEXP/D INSTR, DATA SET LEXD-17
: * OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
: * LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
: *****

8247 031642 000004
8248 031644 012705 031656
8249 031650 004737 041010
8250
8251 031654 000414
8252
8253 031656
8254 031656 152325 052525 052525
8255 031664 052525
8256 031666 000000 000000 000000
8257 031674 000000
8258 031676 000201
8259 031700 046651 046646
8260 031704 000000
8261
8262
8263
8264
8265
8266
8267

†T530: SCOPE
MOV #LEXD17,R5 ; PTR TO TEST DATA SET
JSR PC, @#LEXDT ; GO TEST
BR TST531 ; ;
LEXD17: ; TEST DATA SET LEXD-17:
.WORD 152325,ALTP,ALTP,ALTP ; INITIAL AC FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 201 ; EXPONENT TO BE LOADED
.WORD 046651,046646 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

: *TEST 531 TEST OF LDEXP/D INSTR, DATA SET LEXD-20
: * OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
: * LONG FLOAT, LONG INTEGER, ROUND MODES
: *****

8268 031706 000004
8269 031710 012705 031722
8270 031714 004737 041010
8271
8272 031720 000414
8273
8274 031722
8275 031722 052377 177777 177777
8276 031730 177777
8277 031732 000000 000000 000000
8278 031740 000000
8279 031742 000200
8280 031744 046711 046706
8281 031750 000000
8282
8283

†T531: SCOPE
MOV #LEXD20,R5 ; PTR TO TEST DATA SET
JSR PC, @#LEXDT ; GO TEST
BR TST532 ; ;
LEXD20: ; TEST DATA SET LEXD-20:
.WORD 052377,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 200 ; EXPONENT TO BE LOADED
.WORD 046711,046706 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

```

8284
8285
8286
8287
8288
8289 031752 000004
8290 031754 012705 031766
8291 031760 004737 041010
8292
8293 031764 000414
8294
8295 031766
8296 031766 052201 000002 000003
8297 031774 000000
8298 031776 000000 000000 000000
8299 032004 000000
8300 032006 177600
8301 032010 045713 045704
8302 032014 000000
8303
8304
8305
8306
8307
8308
8309
8310 032016 000004
8311 032020 012705 032032
8312 032024 004737 041010
8313
8314 032030 000414
8315
8316 032032
8317 032032 152252 125252 125252
8318 032040 125252
8319 032042 000000 000000 000000
8320 032050 000000
8321 032052 177577
8322 032054 045613 045604
8323 032060 000000
8324
8325
8326
8327
8328
8329
8330
8331 032062 000004
8332 032064 012705 032076
8333 032070 004737 041010
8334
8335 032074 000414
8336
8337 032076
8338 032076 052217 007417 007417
8339 032104 007417

```

```

*****
: TEST 532 TEST OF LDEXP/D INSTR, DATA SET LEXD-21
: * UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
: * LONG FLOAT, LONG INTEGER, ROUND MODES
: *****
TST532: SCOPE
MOV #LEXD21,R5 ; PTR TO TEST DATA SET
JSR PC,@#LEXDT ; GO TEST
BR TST533 ;;
LEXD21: ; TEST DATA SET LEXD-21:
.WORD 052201,2,3,0 ; INITIAL AC FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
.WORD -200 ; EXPONENT TO BE LOADED
.WORD 045713,045704 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
*****
: TEST 533 TEST OF LDEXP/D INSTR, DATA SET LEXD-22
: * UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
: * LONG FLOAT, SHORT INTEGER, ROUND MODES
: *****
TST533: SCOPE
MOV #LEXD22,R5 ; PTR TO TEST DATA SET
JSR PC,@#LEXDT ; GO TEST
BR TST534 ;;
LEXD22: ; TEST DATA SET LEXD-22:
.WORD 152252,ALTN,ALTN,ALTN ; INITIAL AC FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
.WORD -201 ; EXPONENT TO BE LOADED
.WORD 045613,045604 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )
*****
: TEST 534 TEST OF LDEXP/D INSTR, DATA SET LEXD-23
: * UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
: * LONG FLOAT, LONG INTEGER, TRUNCATE MODES
: *****
TST534: SCOPE
MOV #LEXD23,R5 ; PTR TO TEST DATA SET
JSR PC,@#LEXDT ; GO TEST
BR TST535 ;;
LEXD23: ; TEST DATA SET LEXD-23:
.WORD 052217,ALT4P,ALT4P,ALT4P ; INITIAL AC FLOAT NUMBER

```


L13

FPU ADVANCED INSTR TESTS
D9FP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 169
T534 TEST OF LDEXP/D INSTR, DATA SET LEXD-23

8340	032106	000000	000000	000000	.WORD	0,0,0,0	; EXPECTED FLOAT RESULT
8341	032114	000000					
8342	032116	177576			.WORD	-202	; EXPONENT TO BE LOADED
8343	032120	045753	045744		.WORD	045753,045744	; FPS: BEFORE, AFTER
8344	032124	000000			.WORD	NA	; FEC AFTER (0 = N/A)

8345
8346
8347

```

:*****
:TEST 535 TEST OF LDEXP/D INSTR, DATA SET LEXD-24
:* UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
:* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
:*****

```

8348							
8349							
8350							
8351							
8352	032126	000004					
8353	032130	012705	032142				
8354	032134	004737	041010				

```

TST535: SCOPE
MOV #LEXD24,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXDT ; GO TEST
BR TST536 ;;

```

8355
8356
8357

```

LEXD24: ; TEST DATA SET LEXD-24:
.WORD 152325,ALTP,ALTP,ALTP ; INITIAL AC FLOAT NUMBER

```

8358	032142						
8359	032142	152325	052525	052525			
8360	032150	052525					
8361	032152	000000	000000	000000	.WORD	0,0,0,0	; EXPECTED FLOAT RESULT
8362	032160	000000					
8363	032162	177575			.WORD	-203	; EXPONENT TO BE LOADED
8364	032164	045653	045644		.WORD	045653,045644	; FPS: BEFORE, AFTER
8365	032170	000000			.WORD	NA	; FEC AFTER (0 = N/A)

8366
8367
8368

M13

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 170
T536 TEST OF STEXP/F INSTR, DATA SET SEXF-1

8369
8370
8371
8372
8373
8374 032172 000004
8375 032174 012705 032206
8376 032200 004737 041200
8377
8378 032204 000405
8379
8380 032206
8381 032206 177777 177777
8382 032212 000177
8383 032214 047457 047440
8384
8385
8386
8387
8388
8389
8390
8391 032220 000004
8392 032222 012705 032234
8393 032226 004737 041200
8394
8395 032232 000405
8396
8397 032234
8398 032234 060052 125252
8399 032240 000100
8400 032242 047517 047500
8401
8402
8403
8404
8405
8406
8407
8408 032246 000004
8409 032250 012705 032262
8410 032254 004737 041200
8411
8412 032260 000405
8413
8414 032262
8415 032262 140270 107070
8416 032266 000001
8417 032270 047557 047540
8418
8419
8420
8421
8422
8423
8424

```
*****
; *TEST 536 TEST OF STEXP/F INSTR, DATA SET SEXF-1
; * ALL INTERRUPT ENABLES ON
; * SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
; *****
†T536: SCOPE
MOV #SEXF1,R5 ; PTR TO TEST DATA SET
JSR PC,@#SEXFT ; GO TEST

BR TST537 ;;

SEXF1: ; TEST DATA SET SEXF-1:
.WORD M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD 177 ; EXPONENT EXPECTED TO BE STORED
.WORD 047457,047440 ; FPS: BEFORE, AFTER

*****
; *TEST 537 TEST OF STEXP/F INSTR, DATA SET SEXF-2
; * ALL INTERRUPT ENABLES ON
; * SHORT FLOAT, LONG INTEGER, ROUND MODES
; *****
†T537: SCOPE
MOV #SEXF2,R5 ; PTR TO TEST DATA SET
JSR PC,@#SEXFT ; GO TEST

BR TST540 ;;

SEXF2: ; TEST DATA SET SEXF-2:
.WORD 060052,ALTN ; INITIAL AC FLOAT NUMBER
.WORD 100 ; EXPONENT EXPECTED TO BE STORED
.WORD 047517,047500 ; FPS: BEFORE, AFTER

*****
; *TEST 540 TEST OF STEXP/F INSTR, DATA SET SEXF-3
; * ALL INTERRUPT ENABLES ON
; * SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
; *****
†T540: SCOPE
MOV #SEXF3,R5 ; PTR TO TEST DATA SET
JSR PC,@#SEXFT ; GO TEST

BR TST541 ;;

SEXF3: ; TEST DATA SET SEXF-3:
.WORD 140270,107070 ; INITIAL AC FLOAT NUMBER
.WORD 1 ; EXPONENT EXPECTED TO BE STORED
.WORD 047557,047540 ; FPS: BEFORE, AFTER

*****
; *TEST 541 TEST OF STEXP/F INSTR, DATA SET SEXF-4
; * ALL INTERRUPT ENABLES ON
; * SHORT FLOAT, SHORT INTEGER, ROUND MODES
; *****
```

N13

FPU ADVANCED INSTR TESTS
D0FP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 171
T541 TEST OF STEXP/F INSTR, DATA SET SEXF-4

0425 032274 000004
0426 032276 012705 032310
0427 032302 004737 041200
0428
0429 032306 000405
0430
0431 032310
0432 032310 040125 007417
0433 032314 000000
0434 032316 047413 047404
0435
0436
0437
0438
0439
0440
0441
0442 032322 000004
0443 032324 012705 032336
0444 032330 004737 041200
0445
0446 032334 000405
0447
0448 032336
0449 032336 137760 170360
0450 032342 177777
0451 032344 047407 047410
0452
0453
0454
0455
0456
0457
0458
0459 032350 000004
0460 032352 012705 032364
0461 032356 004737 041200
0462
0463 032362 000405
0464
0465 032364
0466 032364 100307 070707
0467 032370 177601
0468 032372 047507 047510
0469
0470
0471
0472
0473
0474
0475
0476 032376 000004
0477 032400 012705 032412
0478 032404 004737 041200
0479
0480 032410 000405

TST541: SCOPE
MOV #SEXF4,R5 ; PTR TO TEST DATA SET
JSR PC,@#SEXFT ; GO TEST

BR TST542 ;;

SEXF4: ; TEST DATA SET SEXF-4:
.WORD 040125,ALT4P ; INITIAL AC FLOAT NUMBER
.WORD 0 ; EXPONENT EXPECTED TO BE STORED
.WORD 047413,047404 ; FPS: BEFORE, AFTER

*TEST 542 TEST OF STEXP/F INSTR, DATA SET SEXF-5
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, ROUND MODES

TST542: SCOPE
MOV #SEXF5,R5 ; PTR TO TEST DATA SET
JSR PC,@#SEXFT ; GO TEST

BR TST543 ;;

SEXF5: ; TEST DATA SET SEXF-5:
.WORD 137760,ALT4N ; INITIAL AC FLOAT NUMBER
.WORD -1 ; EXPONENT EXPECTED TO BE STORED
.WORD 047407,047410 ; FPS: BEFORE, AFTER

*TEST 543 TEST OF STEXP/F INSTR, DATA SET SEXF-6
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES

TST543: SCOPE
MOV #SEXF6,R5 ; PTR TO TEST DATA SET
JSR PC,@#SEXFT ; GO TEST

BR TST544 ;;

SEXF6: ; TEST DATA SET SEXF-6:
.WORD 100307,070707 ; INITIAL AC FLOAT NUMBER
.WORD -177 ; EXPONENT EXPECTED TO BE STORED
.WORD 047507,047510 ; FPS: BEFORE, AFTER

*TEST 544 TEST OF STEXP/F INSTR, DATA SET SEXF-7
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES

TST544: SCOPE
MOV #SEXF7,R5 ; PTR TO TEST DATA SET
JSR PC,@#SEXFT ; GO TEST

BR TST545 ;;

B14

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 172
T544 TEST OF STEXP/F INSTR, DATA SET SEXF-7

8481
8482 032412
8483 032412 000000 000000
8484 032416 177600
8485 032420 047447 047450
8486
8487
8488

SEXF7: ; TEST DATA SET SEXF-7:
:WORD 0,0 ; INITIAL AC FLOAT NUMBER
:WORD -200 ; EXPONENT EXPECTED TO BE STORED
:WORD 047447,047450 ; FPS: BEFORE, AFTER

```

8489
8490
8491
8492
8493
8494 032424 000004
8495 032426 012705 032440
8496 032432 004737 041334
8497
8498 032436 000407
8499
8500 032440
8501 032440 077600 000000 000000
8502 032446 000000
8503 032450 000177
8504 032452 047717 047700
8505
8506
8507
8508
8509
8510
8511
8512 032456 000004
8513 032460 012705 032472
8514 032464 004737 041334
8515
8516 032470 000407
8517
8518 032472
8519 032472 040360 170360 170360
8520 032500 170360
8521 032502 000001
8522 032504 047657 047640
8523
8524
8525
8526
8527
8528
8529
8530 032510 000004
8531 032512 012705 032524
8532 032516 004737 041334
8533
8534 032522 000407
8535
8536 032524
8537 032524 140107 070707 070707
8538 032532 070707
8539 032534 000000
8540 032536 047613 047604
8541
8542
8543
8544

```

```

*****
; *TEST 545 TEST OF STEXP/D INSTR, DATA SET SEXD-1
; *
; * ALL INTERRUPT ENABLES ON
; * LONG FLOAT, LONG INTEGER, ROUND MODES
*****
↑ST545: SCOPE
MOV #SEXD1,R5 ; PTR TO TEST DATA SET
JSR PC,@#SEXDT ; GO TEST
BR TST546 ;;
SEXD1: ; TEST DATA SET SEXD-1:
.WORD 077600,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD 177 ; EXPONENT EXPECTED TO BE STORED
.WORD 047717,047700 ; FPS: BEFORE, AFTER

*****
; *TEST 546 TEST OF STEXP/D INSTR, DATA SET SEXD-2
; *
; * ALL INTERRUPT ENABLES ON
; * LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
↑ST546: SCOPE
MOV #SEXD2,R5 ; PTR TO TEST DATA SET
JSR PC,@#SEXDT ; GO TEST
BR TST547 ;;
SEXD2: ; TEST DATA SET SEXD-2:
.WORD 040360,ALTN,ALTN,ALTN ; INITIAL AC FLOAT NUMBER
.WORD 1 ; EXPONENT EXPECTED TO BE STORED
.WORD 047657,047640 ; FPS: BEFORE, AFTER

*****
; *TEST 547 TEST OF STEXP/D INSTR, DATA SET SEXD-3
; *
; * ALL INTERRUPT ENABLES ON
; * LONG FLOAT, SHORT INTEGER, ROUND MODES
*****
↑ST547: SCOPE
MOV #SEXD3,R5 ; PTR TO TEST DATA SET
JSR PC,@#SEXDT ; GO TEST
BR TST550 ;;
SEXD3: ; TEST DATA SET SEXD-3:
.WORD 140107,070707,070707,070707 ; INITIAL AC FLOAT NUMBER
.WORD 0 ; EXPONENT EXPECTED TO BE STORED
.WORD 047613,047604 ; FPS: BEFORE, AFTER

*****
; *TEST 550 TEST OF STEXP/D INSTR, DATA SET SEXD-4

```

8545
8546
8547
8548
8549
8550
8551
8552
8553
8554
8555
8556
8557
8558
8559
8560
8561
8562
8563
8564
8565
8566
8567
8568
8569
8570
8571
8572
8573
8574
8575
8576
8577
8578
8579
8580
8581
8582
8583
8584
8585
8586
8587
8588
8589
8590
8591
8592
8593
8594
8595
8596
8597
8598
8599
8600

032542 000004
032544 012705 032556
032550 004737 041334
032554 000407
032556 037652 125252 125252
032564 125252
032566 177777
032570 047707 047710

032574 000004
032576 012705 032610
032602 004737 041334
032606 000407
032610 120070 107070 107070
032616 107070
032620 177700
032622 047747 047750

032626 000004
032630 012705 032642
032634 004737 041334
032640 000407
032642 000217 007417 007417
032650 007417
032652 177601
032654 047647 047650

```

;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, LONG INTEGER, ROUND MODES
;*****
↑T550: SCOPE
MOV #SEXD4,R5 ; PTR TO TEST DATA SET
JSR PC,@#SEXDT ; GO TEST
BR TST551 ;;

SEXD4: ; TEST DATA SET SEXD-4:
.WORD 037652,ALTN,ALTN,ALTN ; INITIAL AC FLOAT NUMBER
.WORD -1 ; EXPONENT EXPECTED TO BE STORED
.WORD 047707,047710 ; FPS: BEFORE, AFTER

;*****
; TEST 551 TEST OF STEXP/D INSTR, DATA SET SEXD-5
;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
;*****
↑T551: SCOPE
MOV #SEXD5,R5 ; PTR TO TEST DATA SET
JSR PC,@#SEXDT ; GO TEST
BR TST552 ;;

SEXD5: ; TEST DATA SET SEXD-5:
.WORD 120070,107070,107070,107070 ; INITIAL AC FLOAT NUMBER
.WORD -100 ; EXPONENT EXPECTED TO BE STORED
.WORD 047747,047750 ; FPS: BEFORE, AFTER

;*****
; TEST 552 TEST OF STEXP/D INSTR, DATA SET SEXD-6
;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
;*****
↑T552: SCOPE
MOV #SEXD6,R5 ; PTR TO TEST DATA SET
JSR PC,@#SEXDT ; GO TEST
BR TST553 ;;

SEXD6: ; TEST DATA SET SEXD-6:
.WORD 000217,ALT4P,ALT4P,ALT4P ; INITIAL AC FLOAT NUMBER
.WORD -177 ; EXPONENT EXPECTED TO BE STORED
.WORD 047647,047650 ; FPS: BEFORE, AFTER

;*****
; TEST 553 TEST OF STEXP/D INSTR, DATA SET SEXD-7
;*
;* ALL INTERRUPT ENABLES ON
;* LONG FLOAT, SHORT INTEGER, ROUND MODES

```

E14

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 175
T553 TEST OF STEXP/D INSTR, DATA SET SEXD-7

8601				
8602	032660	000004		
8603	032662	012705	032674	
8604	032666	004737	041334	
8605				
8606	032672	000407		
8607				
8608	032674			
8609	032674	000177	177777	177777
8610	032702	177777		
8611	032704	177600		
8612	032706	047607	047610	
8613				
8614				
8615				

```

:*****
↑T553: SCOPE
      MOV    #SEXD7,R5      ; PTR TO TEST DATA SET
      JSR    PC,@#SEXDT    ; GO TEST
      BR     TST554        ;;
SEXD7: ; TEST DATA SET SEXD-7:
      .WORD 000177,MI,MI,MI ; INITIAL AC FLOAT NUMBER
      .WORD -200             ; EXPONENT EXPECTED TO BE STORED
      .WORD 047607,047610   ; FPS: BEFORE, AFTER

```

```

8616
8617
8618
8619 032712
8620 032712 000004
8621
8622
8623
8624
8625
8626
8627
8628
8629
8630
8631
8632 032714 076600 000022
8633 032720 032700 000020
8634 032724 001423
8635
8636 032726 032777 000002 146212
8637 032734 001017
8638
8639 032736 012701 010000
8640 032742 076600 000144
8641 032746 030100
8642 032750 001402
8643 032752 040100
8644 032754 000401
8645 032756 050100
8646 032760 076600 000344
8647
8648 032764 030100
8649 032766 001002
8650 032770 000137 002654
8651
8652
8653
8654
8655
8656
8657
8658
8659
8660
8661
8662
8663 032774
8664 032774 005037 001104
8665 033000 005037 001102
8666 033004 005037 001310
8667 033010 005237 001332
8668 033014 042737 100000 001332
8669 033022 005327
8670 033024 000001
8671 033026 003021

```

```

;*****
.SBTTL SUB PASS END CONTROL

TST554: ;FORCE LAST TEST NUMBER
SCOPE ;CHECK FOR TEST ITERATIONS HERE

;IF TEST ONLY EITHER HFP OR WFP. ENTER "EOP" ROUTINE DIRECTLY

; IF IN ALTERNATE HFP/WFP MODE,
; COMPLEMENT FLAG<5> HFP ENABLE BIT,
; ENTER EOP ROUTINE ONLY IF ABOUT TO TEST HFP NEXT,
; TESTING SEQUENCE IS: PASS#1 HFP SUB-PASS
; PASS#1 WFP SUB-PASS
; PASS#2 HFP SUB-PASS
; ...

MED RWHAMI ;GET WHAMI INTO RO
BIT #BIT04,RO ;1=HFP PRESENT, 0=NONE
BEQ $EOP ;EXIT IF NONE

BIT #SW01,$SWR ;1=HFP OR WFP TEST ONLY
BNE $EOP ;0=ALTERNATE HFP AND WFP TESTS

MOV #BIT12,R1 ;HFP PRESENT, AND IN ALTERNATE MODE;
MED RFLAG ;SO READ FLAGS
BIT R1,RO ;COMPLEMENT FLAG<5>=BIT12=HFP ENABLE FLAG
BEQ 1$
BIC R1,RO ;CLEAR BIT 12
BR 2$
1$: BIS R1,RO ;SET BIT 12
2$: MED ,WFLAG ;REWRITE FLAGS

BIT R1,RO ;HFP OR WFP NEXT ?
BNE $EOP ;IF HFP AGAIN, START NEW PASS
JMP @SUBPAS ;IF WFP, NEXT SUBPASS

```

```

;*****
.SBTTL END OF PASS ROUTINE (MODIFIED SYSMAC)

;*INCREMENT THE PASS NUMBER ($PASS)
;*IF SW<10>=0, DING BELL ON PASS END
;*IF THERE'S A MONITOR, GO TO IT
;* ELSE JUMP TO NEWPAS

$EOP:
CLR $ERFLG ;ZERO ERROR COUNT
CLR $STNM ;ZERO TEST NUMBER
CLR $TIMES ;ZERO NUMBER OF ITERATION.
INC $PASS ;INCREMENT PASS COUNT,
; BUT NEVER LET IN GO NEGATI.
BIC #100000,$PASS ;PASS LOOP ?
DEC (PC)+
$EOPCT: .WORD 1
BGT $DOAGN ;YES

```


FPU ADVANCED INSTR TESTS
DQFP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 177
END OF PASS ROUTINE (MODIFIED SYSMAC)

8672	033030	012737			MOV	(PC)+,2(PC)+	:RESTORE COUNTER
8673	033032	000001			SENDCT: .WORD	1	:
8674	033034	033024			\$EOPCT		:
8675	033036	032777	002000	146102	BIT	#SW10,2SWR	:BELL ON PASS END ?
8676	033044	001002			BNE	\$GET42	:NO
8677	033046	104401	001314		TYPE	,\$BELL	:YES
8678							
8679	033052	013700	000042		\$GET42: MOV	2#42,RO	:GET MONITOR ADDRESS
8680	033056	001405			BEQ	\$DOAGN	:NO MONITOR
8681	033060	000005			RESET		:CLEAR WORLD
8682							
8683	033062	004710			SENDAD: JSR	PC,(RO)	:GO TO MONITOR
8684	033064	000240			NOP		:
8685	033066	000240			NOP		:RESERVED FOR ACT11
8686	033070	000240			NOP		:
8687							
8688	033072	000137	002616		\$DOAGN: JMP	2#NEWPAS	:RETURN
8689							
8690							

```

.SBTTL SUBR TO TEST THE CMPF INSTRUCTION
CMPFT:
MOV #7,R0 ; LOAD $TMP0-6
MOV R5,R1 ; WITH TEST DATA SETS
MOV #TMP0,R2 ; FOR DISPLAY LATER
MOV (R1)+,(R2)+ ;
SOB RO,-2 ;
MOV #CMPFL,$LPERR ; ERROR LOOPING ADDRESS
CMPFL: SETF ; F MODE
LDF (R5),AC3 ; INITIAL AC FLOAT NUMBER
LDFPS 10(R5) ; INITIAL FPS
CMPFI: CMPF 4(R5),AC3 ; (MEM)-(AC3)
STFPS FPS ; STORE FPS AFTER
STST FEC ; STORE FEC/FEA AFTER
STF AC3,$REG0 ; STORE AC NUMBER
8711 033152 023765 002000 000012 CMP FPS,12(R5) ; CHECK FPS
8712 033160 001401 65$ ; FPS IS OK
8713 033162 104002 2 ; FPS BAD
8714 033164 005765 000014 65$: TST 14(R5) ; DOES FEC/FEA APPLY?
8715 033170 100014 BPL 66$ ; NO - SKIP TEST
8716 033172 012737 033132 002014 MOV #CMPFI,EXPFEA ; GET EXPECTED FEA
8717 033200 123765 002002 000014 CMPB FEC,14(R5) ; COMPARE FEC-S
8718 033206 001004 BNE 64$ ; NOT EQUAL
8719 033210 023737 002004 002014 CMP FEA,EXPFEA ; COMPARE FEA-S
8720 033216 001401 BEQ 66$ ; FEC, FEA OK
8721 033220 104012 64$: ERROR 12 ; FEC OR FEA ARE BAD
8722 033222 66$:
8724 033222 023715 001170 CMP $REG0,(R5) ; 1ST WORD OF RESULT CHECK?
8725 033226 001004 BNE 67$ ; NO
8726 033230 023765 001172 000002 CMP $REG1,2(R5) ; 2ND WORD OF RESULT CHECK?
8727 033236 001401 BEQ 68$ ; ALL WORDS OK
8728 033240 104021 67$: ERROR 21 ; NUMBERS NOT EQUAL
8729 033242 68$:
8731 033242 000207 RTS PC ; RETURN TO TEST CALLER
8733 ;:*****
.SBTTL SUBR TO TEST THE CMPD INSTRUCTION
CMPDT:
MOV #13,R0 ; LOAD $TMP0-12
MOV R5,R1 ; WITH TEST DATA SETS
MOV #TMP0,R2 ; FOR DISPLAY LATER
MOV (R1)+,(R2)+ ;
SOB RO,-2 ;
MOV #CMPDL,$LPERR ; ERROR LOOPING ADDRESS
CMPDL: SETD ; D MODE
LDD (R5),AC2 ; INITIAL AC FLOAT NUMBER

```

FPU ADVANCED INSTR TESTS
 DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 179
 SUBR TO TEST THE CMPD INSTRUCTION

8747	033274	170165	000020		LDFPS	20(R5)		; INITIAL FPS
8748								
8749	033300	173665	000010		CMPDI:	CMPD	10(R5),AC2	; (MEM)-(AC2)
8750								
8751	033304	170237	002000		STEPS	FPS		; STORE FPS AFTER
8752	033310	170337	002002		STST	FEC		; STORE FEC/FEA AFTER
8753	033314	174237	001170		STD	AC2,\$REGO		; STORE AC AFTER
8754								
8755	033320	023765	002000	000022	CMP	FPS,22(R5)		; CHECK FPS
8756	033326	001401			BEQ	65\$; FPS IS OK
8757	033330	104005			ERROR	5		; FPS BAD
8758	033332	005765	000024	65\$:	TST	24(R5)		; DOES FEC/FEA APPLY?
8759	033336	100014			BPL	66\$; NO - SKIP TEST
8760	033340	012737	033300	002014	MOV	#CMPDI,EXPFEA		; GET EXPECTED FEA
8761	033346	123765	002002	000024	CMPB	FEC,24(R5)		; COMPARE FEC-S
8762	033354	001004			BNE	64\$; NOT EQUAL
8763	033356	023737	002004	002014	CMP	FEA,EXPFEA		; COMPARE FEA-S
8764	033364	001401			BEQ	66\$; FEC, FEA OK
8765	033366	104015		64\$:	ERROR	15		; FEC OR FEA ARE BAD
8766	033370			66\$:				
8767								
8768	033370	023715	001170		CMP	\$REG0,(R5)		; 1ST WORD OF RESULT CHECK?
8769	033374	001014			BNE	67\$; NO
8770	033376	023765	001172	000002	CMP	\$REG1,2(R5)		; 2ND WORD OF RESULT CHECK?
8771	033404	001010			BNE	67\$; NO
8772	033406	023765	001174	000004	CMP	\$REG2,4(R5)		; 3RD WORD OF RESULT CHECK?
8773	033414	001004			BNE	67\$; NO
8774	033416	023765	001176	000006	CMP	\$REG3,6(R5)		; 4TH WORD OF RESULT CHECK?
8775	033424	001401			BEQ	68\$; ALL WORDS OK
8776	033426	104022		67\$:	ERROR	22		; NUMBERS NOT EQUAL
8777	033430			68\$:				
8778								
8779	033430	000207			RTS	PC		; RETURN TO TEST CALLER
8780								

```

      .SBTTL SUBR TO TEST THE ADDF INSTRUCTION
8781
8782
8783 033432
8784 033432 012700 000011
8785 033436 010501
8786 033440 012702 001230
8787 033444 012122
8788 033446 077002
8789 033450 012737 033456 001114
8790
8791 033456 170001
8792 033460 172515
8793 033462 170165 000014
8794
8795 033466 172165 000004
8796
8797 033472 172237 002000
8798 033476 170337 002002
8799 033502 174137 001170
8800
8801 033506 023765 002000 000016
8802 033514 001401
8803 033516 104004
8804 033520 005765 000020 65$:
8805 033524 100014
8806 033526 012737 033466 002014
8807 033534 123765 002002 000020
8808 033542 001004
8809 033544 023737 002004 002014
8810 033552 001401
8811 033554 104014 64$:
8812 033556 66$:
8813
8814 033556 023765 001170 000010
8815 033564 001004
8816 033566 023765 001172 000012
8817 033574 001401
8818 033576 104023 67$:
8819 033600 68$:
8820
8821 033600 000207
8822
8823
8824
8825
8826 033602
8827 033602 012700 000017
8828 033606 010501
8829 033610 012702 001230
8830 033614 012122
8831 033616 077002
8832 033620 012737 033626 001114
8833
8834 033626 170011
8835 033630 172415
8836 033632 170165 000030

      ADDFT:
      MOV      #11,R0      ; LOAD $TMP0-10
      MOV      R5,R1      ; WITH TEST DATA SETS
      MOV      #TMP0,R2   ; FOR DISPLAY LATER
      MOV      (R1)+(R2)+
      SOB      R0,-2
      MOV      #ADDFL,$LPERR ; ERROR LOOPING ADDRESS

      ADDFL: SETF      ; F MODE
      LDF      (R5),AC1   ; INITIAL AC FLOAT NUMBER
      LDFPS   14(R5)     ; INITIAL FPS

      ADDFI: ADDF      4(R5),AC1 ; (AC1)+(MEM)->AC1

      STFPS   FPS        ; STORE FPS AFTER
      STST   FEC        ; STORE FEC/FEA AFTER
      STF    AC1,$REG0   ; RESULT OF ADDF

      CMP     FPS,16(R5) ; CHECK FPS
      BEQ    65$        ; FPS IS OK
      ERROR  4          ; FPS BAD
      TST    20(R5)     ; DOES FEC/FEA APPLY?
      BPL    66$        ; NO - SKIP TEST
      MOV    #ADDFI,EXPFEA ; GET EXPECTED FEA
      CMPB  FEC,20(R5)  ; COMPARE FEC-S
      BNE   64$        ; NOT EQUAL
      CMP   FEA,EXPFEA ; COMPARE FEA-S
      BEQ   66$        ; FEC, FEA OK
      ERROR 14         ; FEC OR FEA ARE BAD

      CMP     $REG0,10(R5) ; 1ST WORD OF RESULT CHECK?
      BNE    67$        ; NO
      CMP     $REG1,12(R5) ; 2ND WORD OF RESULT CHECK?
      BEQ    68$        ; ALL WORDS OK
      ERROR  23        ; NUMBERS NOT EQUAL

      RTS     PC        ; RETURN TO TEST CALLER

;*****
.SBTTL SUBR TO TEST THE ADDD INSTRUCTION
      ADDDT:
      MOV      #17,R0      ; LOAD $TMP0-16
      MOV      R5,R1      ; WITH TEST DATA SETS
      MOV      #TMP0,R2   ; FOR DISPLAY LATER
      MOV      (R1)+(R2)+
      SOB      R0,-2
      MOV      #ADDDL,$LPERR ; ERROR LOOPING ADDRESS

      ADDDL: SETD      ; D MODE
      LDD      (R5),ACC   ; INITIAL AC FLOAT NUMBER
      LDFPS   30(R5)     ; INITIAL FPS

```



```

8870      .SBTTL SUBR TO TEST THE SUBF INSTRUCTION
8871
8872      SUBFT:
8873      033772 012700 000011      MOV      #11,R0      ; LOAD $TMP0-10
8874      033776 010501              MOV      R5,R1      ; WITH TEST DATA SETS
8875      034000 012702 001230      MOV      #TMP0,R2    ; FOR DISPLAY LATER
8876      034004 012122              MOV      (R1)+,(R2)+
8877      034006 077002              SOB      RO,-2
8878      034010 012737 034016 001114  MOV      #SUBFL,$LPERR ; ERROR LOOPING ADDRESS
8879
8880      SUBFL: SETF              ; F MODE
8881      034020 172415              LDF      (R5),AC0    ; INITIAL AC FLOAT NUMBER
8882      034022 170165 000014      LDFPS   14(R5)      ; INITIAL FPS
8883
8884      SUBFI: SUBF      4(R5),AC0 ; (AC0)-(MEM)->AC0
8885
8886      034032 170237 002000      STFPS   FPS         ; STORE FPS AFTER
8887      034036 170337 002002      STST    FEC         ; STORE FEC/FEA AFTER
8888      034042 174037 001170      STF     AC0,$REG0   ; RESULT OF SUBF
8889
8890      034046 023765 002000 000016  CMP      FPS,16(R5) ; CHECK FPS
8891      034054 001401              BEQ     65$         ; FPS IS OK
8892      034056 104004              ERROR   4          ; FPS BAD
8893      034060 005765 000020      65$:   TST      20(R5) ; DOES FEC/FEA APPLY?
8894      034064 100014              BPL     66$         ; NO - SKIP TEST
8895      034066 012737 034026 002014  MOV      #SUBFI,EXPFEA ; GET EXPECTED FEA
8896      034074 123765 002002 000020  CMPB    FEC,20(R5) ; COMPARE FEC-S
8897      034102 001004              BNE     64$         ; NOT EQUAL
8898      034104 023737 002004 002014  CMP      FEA,EXPFEA ; COMPARE FEA-S
8899      034112 001401              BEQ     66$         ; FEC, FEA OK
8900      034114 104014              64$:   ERROR   14   ; FEC OR FEA ARE BAD
8901      034116
8902
8903      034116 023765 001170 000010  CMP      $REG0,10(R5) ; 1ST WORD OF RESULT CHECK?
8904      034124 001004              BNE     67$         ; NO
8905      034126 023765 001172 000012  CMP      $REG1,12(R5) ; 2ND WORD OF RESULT CHECK?
8906      034134 001401              BEQ     68$         ; ALL WORDS OK
8907      034136 104023              67$:   ERROR   23   ; NUMBERS NOT EQUAL
8908      034140
8909      68$:
8910      034140 000207      RTS     PC          ; RETURN TO TEST CALLER
8911
8912      ;:*****
8913      .SBTTL SUBR TO TEST THE SUBD INSTRUCTION
8914
8915      SUBDT:
8916      034142 012700 000017      MOV      #17,R0      ; LOAD $TMP0-16
8917      034146 010501              MOV      R5,R1      ; WITH TEST DATA SETS
8918      034150 012702 001230      MOV      #TMP0,R2    ; FOR DISPLAY LATER
8919      034154 012122              MOV      (R1)+,(R2)+
8920      034156 077002              SOB      RO,-2
8921      034160 012737 034166 001114  MOV      #SUBDL,$LPERR ; ERROR LOOPING ADDRESS
8922
8923      SUBDL: SETD              ; D MODE
8924      034170 172715              LDD     (R5),AC3    ; INITIAL AC FLOAT NUMBER
8925      034172 170165 000030      LDFPS   30(R5)      ; INITIAL FPS

```



```

.SBTTL SUBR TO TEST THE MULF INSTRUCTION
MULFT:
MOV #11,R0 ; LOAD $TMP0-10
MOV R5,R1 ; WITH TEST DATA SETS
MOV #TMP0,R2 ; FOR DISPLAY LATER
MOV (R1)+(R2)+ ;
SOB R0,-2 ;
MOV #MULFL,$LPERR ; ERROR LOOPING ADDRESS
MULFL: SETF ; F MODE
LDF (R5),AC3 ; INITIAL AC FLOAT NUMBER
LDFPS 14(R5) ; INITIAL FPS
MULFI: MULF 4(R5),AC3 ; (AC3)*(MEM)->AC3
STFPS FPS ; STORE FPS AFTER
STST FEC ; STORE FEC/FEA AFTER
STF AC3,$REG0 ; RESULT OF MULF
8979 034406 023765 002000 000016 CMP FPS,16(R5) ; CHECK FPS
8980 034414 001401 BEQ 65$ ; FPS IS OK
8981 034416 104004 ERROR 4 ; FPS BAD
8982 034420 005765 000020 65$: TST 20(R5) ; DOES FEC/FEA APPLY?
8983 034424 100014 BPL 66$ ; NO - SKIP TEST
8984 034426 012737 034366 002014 MOV #MULFI,EXPFEA ; GET EXPECTED FEA
8985 034434 123765 002002 000020 CMPB FEC,20(R5) ; COMPARE FEC-S
8986 034442 001004 BNE 64$ ; NOT EQUAL
8987 034444 023737 002004 002014 CMP FEA,EXPFEA ; COMPARE FEA-S
8988 034452 001401 BEQ 66$ ; FEC, FEA OK
8989 034454 104014 64$: ERROR 14 ; FEC OR FEA ARE BAD
8990 034456 66$:
8992 034456 023765 001170 000010 CMP $REG0,10(R5) ; 1ST WORD OF RESULT CHECK?
8993 034464 001004 BNE 67$ ; NO
8994 034466 023765 001172 000012 CMP $REG1,12(R5) ; 2ND WORD OF RESULT CHECK?
8995 034474 001401 BEQ 68$ ; ALL WORDS OK
8996 034476 104025 67$: ERROR 25 ; NUMBERS NOT EQUAL
8997 034500 68$:
8999 034500 000207 RTS PC ; RETURN TO TEST CALLER
9000
9001 ;:*****
9002 .SBTTL SUBR TO TEST THE MULD INSTRUCTION
9003
9004 034502 MULDT:
9005 034502 012700 000017 MOV #17,R0 ; LOAD $TMP0-16
9006 034506 010501 MOV R5,R1 ; WITH TEST DATA SETS
9007 034510 012702 001230 MOV #TMP0,R2 ; FOR DISPLAY LATER
9008 034514 012122 MOV (R1)+(R2)+ ;
9009 034516 077002 SOB R0,-2 ;
9010 034520 012737 034526 001114 MOV #MULD, $LPERR ; ERROR LOOPING ADDRESS
9011
9012 034526 170011 MULDL: SETD ; D MODE
9013 034530 172615 LDD (R5),AC2 ; INITIAL AC FLOAT NUMBER
9014 034532 170165 000030 LDFPS 30(R5) ; INITIAL FPS

```



```

.SBTTL SUBR TO TEST THE DIVF INSTRUCTION
DIVFT:
9048
9049
9050 034672 012700 000011      MOV      #11,R0      ; LOAD $TMPD-10
9051 034672 010501              MOV      R5,R1      ; WITH TEST DATA SETS
9052 034676 012702 001230      MOV      #TMPD,R2   ; FOR DISPLAY LATER
9053 034700 012122              MOV      (R1)+(R2)+
9054 034704 077002              SOB      R0,-2
9055 034706 012737 034716 001114  MOV      #DIVFL,$LPERR ; ERROR LOOPING ADDRESS
9056 034710
9057
9058 034716 170001      DIVFL:  SETF      ; F MODE
9059 034720 172615      LDF      (R5),AC2   ; INITIAL AC FLOAT NUMBER
9060 034722 170165 000014      LDFPS   14(R5)     ; INITIAL FPS
9061
9062 034726 174665 000004      DIVFI:  DIVF      4(R5),AC2 ; (AC2)/(MEM)->AC2
9063
9064 034732 170237 002000      STFPS   FPS        ; STORE FPS AFTER
9065 034736 170337 002002      STST    FEC        ; STORE FEC/FEA AFTER
9066 034742 174237 001170      STF     AC2,$REGO   ; RESULT OF DIVF
9067
9068 034746 023765 002000 000016      CMP     FPS,16(R5)  ; CHECK FPS
9069 034754 001401              BEQ     65$         ; FPS IS OK
9070 034756 104004              ERROR   4          ; FPS BAD
9071 034760 005765 000020      65$:   TST     20(R5)  ; DOES FEC/FEA APPLY?
9072 034764 100014              BPL     66$         ; NO - SKIP TEST
9073 034766 012737 034726 002014      MOV     #DIVFI,EXPFEA ; GET EXPECTED FEA
9074 034774 123765 002002 000020      CMPB   FEC,20(R5)  ; COMPARE FEC-S
9075 035002 001004              BNE     64$         ; NOT EQUAL
9076 035004 023737 002004 002014      CMP     FEA,EXPFEA ; COMPARE FEA-S
9077 035012 001401              BEQ     66$         ; FEC, FEA OK
9078 035014 104014              64$:   ERROR   14   ; FEC OR FEA ARE BAD
9079 035016
9080
9081 035016 023765 001170 000010      CMP     $REGO,10(R5) ; 1ST WORD OF RESULT CHECK?
9082 035024 001004              BNE     67$         ; NO
9083 035026 023765 001172 000012      CMP     $REG1,12(R5) ; 2ND WORD OF RESULT CHECK?
9084 035034 001401              BEQ     68$         ; ALL WORDS OK
9085 035036 104025              67$:   ERROR   25   ; NUMBERS NOT EQUAL
9086 035040
9087
9088 035040 000207      RTS     PC          ; RETURN TO TEST CALLER
9089
9090
9091
9092
9093
9094 035042 012700 000017      .: *****
.SBTTL SUBR TO TEST THE DIVD INSTRUCTION
DIVDT:
9095 035042 010501              MOV      #17,R0      ; LOAD $TMPD-16
9096 035046 012702 001230      MOV      R5,R1      ; WITH TEST DATA SETS
9097 035050 012122              MOV      #TMPD,R2   ; FOR DISPLAY LATER
9098 035054 012122              MOV      (R1)+(R2)+
9099 035056 077002              SOB      R0,-2
9100 035060 012737 035066 001114  MOV      #DIVDL,$LPERR ; ERROR LOOPING ADDRESS
9101
9102
9103 035066 170011      DIVDL:  SETD      ; D MODE
          LDD      (R5),AC1 ; INITIAL AC FLOAT NUMBER
          LDFPS   30(R5)  ; INITIAL FPS

```


E15

FPU ADVANCED INSTR TESTS
DOFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 188
SUBR TO TEST THE MODF INSTRUCTION, USING 2 ACCUMULATORS

```

.SBTTL SUBR TO TEST THE MODF INSTRUCTION, USING 2 ACCUMULATORS
MD2FT:
9138
9139
9140 035232
9141 035232 012700 000013
9142 035236 010501
9143 035240 012702 001230
9144 035244 012112
9145 035246 077002
9146 035250 012737 035256 001114
9147
9148 035256 170001 MD2FL: SETF ; F MODE
9149 035260 172615 LDF (R5),AC2 ; INITIAL AC FLOAT NUMBER
9150 035262 172737 002036 LDF PREVAC,AC3 ; FOR FEC-14 TEST
9151 035266 170165 000020 LDFPS 20(R5) ; INITIAL FPS
9152
9153 035272 171665 000004 MD2FI: MODF 4(R5),AC2 ; FRAC((AC2)*(MEM))->AC2
9154 ; INT((AC2)*(MEM))->AC3
9155
9156 035276 170237 002000 STFPS FPS ; STORE FPS AFTER
9157 035302 170337 002002 STST FEC ; STORE FEC/FEA AFTER
9158 035306 174237 001170 STF AC2,$REG0 ; STORE FRAC PART
9159 035312 174337 001174 STF AC3,$REG2 ; STORE INT PART
9160
9161 035316 023765 002000 000022 CMP FPS,22(R5) ; CHECK FPS
9162 035324 001401 BEQ 65$ ; FPS IS OK
9163 035326 104005 ERROR 5 ; FPS BAD
9164 035330 005765 000024 65$: TST 24(R5) ; DOES FEC/FEA APPLY?
9165 035334 100014 BPL 66$ ; NO - SKIP TEST
9166 035336 012737 035272 002014 MOV #MD2FI,EXPFEA ; GET EXPECTED FEA
9167 035344 123765 002002 000024 CMPB FEC,24(R5) ; COMPARE FEC-S
9168 035352 001004 BNE 64$ ; NOT EQUAL
9169 035354 023737 002004 002014 CMP FEA,EXPFEA ; COMPARE FEA-S
9170 035362 001401 BEQ 66$ ; FEC, FEA OK
9171 035364 104015 64$: ERROR 15 ; FEC OR FEA ARE BAD
9172 035366
9173
9174 ; CHECK FRACTION PART
9175 035366 023765 001170 000010 CMP $REG0,10(R5) ; 1ST WORD OF RESULT CHECK?
9176 035374 001004 BNE 67$ ; NO
9177 035376 023765 001172 000012 CMP $REG1,12(R5) ; 2ND WORD OF RESULT CHECK?
9178 035404 001401 BEQ 68$ ; ALL WORDS OK
9179 035406 104027 67$: ERROR 27 ; NUMBERS NOT EQUAL
9180 035410 68$:
9181
9182 ; CHECK INTEGER PART
9183 035410 023765 001174 000014 CMP $REG2,14(R5) ; 1ST WORD OF RESULT CHECK?
9184 035416 001004 BNE 69$ ; NO
9185 035420 023765 001176 000016 CMP $REG3,16(R5) ; 2ND WORD OF RESULT CHECK?
9186 035426 001401 BEQ 70$ ; ALL WORDS OK
9187 035430 104030 69$: ERROR 30 ; NUMBERS NOT EQUAL
9188 035432 70$:
9189
9190 035432 000207 RTS PC ; RETURN TO TEST CALLER
9191
9192
9193 ;*****

```

FPU ADVANCED INSTR TESTS
 DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 189
 SUBR TO TEST THE MODD INSTRUCTION, USING 2 ACCUMULATORS

.SBTTL SUBR TO TEST THE MODD INSTRUCTION, USING 2 ACCUMULATORS

9194									
9195									
9196	035434								
9197	035434	012700	000023						
9198	035440	010501							
9199	035442	012702	001230						
9200	035446	012122							
9201	035450	077002							
9202	035452	012737	035460	001114					
9203									
9204	035460	170011							
9205	035462	172415							
9206	035464	172537	002036						
9207	035470	170165	000040						
9208									
9209	035474	171465	000010						
9210									
9211									
9212	035500	170237	002000						
9213	035504	170337	002002						
9214	035510	174037	001170						
9215	035514	174137	001200						
9216									
9217	035520	023765	002000	000042					
9218	035526	001401							
9219	035530	104010							
9220	035532	005765	000044						
9221	035536	100014							
9222	035540	012737	035474	002014					
9223	035546	123765	002002	000044					
9224	035554	001004							
9225	035556	023737	002004	002014					
9226	035564	001401							
9227	035566	104020							
9228	035570								
9229									
9230									
9231	035570	023765	001170	000020					
9232	035576	001014							
9233	035600	023765	001172	000022					
9234	035606	001010							
9235	035610	023765	001174	000024					
9236	035616	001004							
9237	035620	023765	001176	000026					
9238	035626	001401							
9239	035630	104031							
9240	035632								
9241									
9242									
9243	035632	023765	001200	000030					
9244	035640	001014							
9245	035642	023765	001202	000032					
9246	035650	001010							
9247	035652	023765	001204	000034					
9248	035660	001004							
9249	035662	023765	001206	000036					

```

MD2DT:  MOV      #23,R0          ; LOAD $TMP0-22
        MOV      R5,R1          ; WITH TEST DATA SETS
        MOV      #TMP0,R2       ; FOR DISPLAY LATER
        MOV      (R1)+(R2)+     ;
        SOB      R0,-2          ;
        MOV      #MD2DL,$LPERR  ; ERROR LOOPING ADDRESS

MD2DL:  SETD     D MODE          ; D MODE
        LDC     (R5),AC0        ; INITIAL AC FLOAT NUMBER
        LDD     PREVAC,AC1     ; FOR FEC-14 TEST
        LDFPS   40(R5)         ; INITIAL FPS

MD2DI:  MODD     10(R5),AC0     ; FRAC((AC0)*(MEM))->AC0
        ; INT((AC0)*(MEM))->AC1

        STFPS   FPS            ; STORE FPS AFTER
        STST    FEC            ; STORE FEC/FEA AFTER
        STD     AC0,$REG0       ; STORE FRAC PART
        STD     AC1,$REG4       ; STORE INT PART

        CMP     FPS,42(R5)     ; CHECK FPS
        BEQ     65$            ; FPS IS OK
        ERROR   10            ; FPS BAD

65$:    TST     44(R5)         ; DOES FEC/FEA APPLY?
        BPL     66$            ; NO - SKIP TEST
        MOV     #MD2DI,EXPFEA  ; GET EXPECTED FEA
        CMPB    FEC,44(R5)     ; COMPARE FEC-S
        BNE     64$            ; NOT EQUAL
        CMP     FEA,EXPFEA     ; COMPARE FEA-S
        BEQ     66$            ; FEC, FEA OK
        ERROR   20            ; FEC OR FEA ARE BAD

64$:    ; CHECK FRACTION PART OF RESULT
        CMP     $REG0,20(R5)   ; 1ST WORD OF RESULT CHECK?
        BNE     67$            ; NO
        CMP     $REG1,22(R5)   ; 2ND WORD OF RESULT CHECK?
        BNE     67$            ; NO
        CMP     $REG2,24(R5)   ; 3RD WORD OF RESULT CHECK?
        BNE     67$            ; NO
        CMP     $REG3,26(R5)   ; 4TH WORD OF RESULT CHECK?
        BEQ     68$            ; ALL WORDS OK
        ERROR   31            ; NUMBERS NOT EQUAL

67$:    ; CHECK INTEGER PART
        CMP     $REG4,30(R5)   ; 1ST WORD OF RESULT CHECK?
        BNE     69$            ; NO
        CMP     $REG5,32(R5)   ; 2ND WORD OF RESULT CHECK?
        BNE     69$            ; NO
        CMP     $REG6,34(R5)   ; 3RD WORD OF RESULT CHECK?
        BNE     69$            ; NO
        CMP     $REG7,36(R5)   ; 4TH WORD OF RESULT CHECK?

```

G15

FPU ADVANCED INSTR TESTS
DOFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 190
SUBR TO TEST THE MODD INSTRUCTION, USING 2 ACCUMULATORS

9250 035670 001401
9251 035672 104032
9252 035674
9253
9254 035674 000207

69\$: BEQ 70\$; ALL WORDS OK
70\$: ERROR 32 ; NUMBERS NOT EQUAL
RTS PC ; RETURN TO TEST CALLER

H15

FPU ADVANCED INSTR TESTS
DOFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 191
SUBR TO TEST THE MODF INSTRUCTION, USING 1 ACCUMULATOR

```

9255      .SBTTL  SUBR TO TEST THE MODF INSTRUCTION, USING 1 ACCUMULATOR
9256
9257      035676      MD1FT:
9258      035676      012700      000013      MOV      #13,R0      ; LOAD $TMP0-12
9259      035702      010501      MOV      R5,R1      ; WITH TEST DATA SETS
9260      035704      012702      001230      MOV      #TMP0,R2      ; FOR DISPLAY LATER
9261      035710      012122      MOV      (R1)+,(R2)+
9262      035712      077002      SOB      R0,-2
9263      035714      012737      035722      001114      MOV      #MD1FL,$LPERR ; ERROR LOOPING ADDRESS
9264
9265      035722      170001      MD1FL: SETF      ; F MODE
9266      035724      172715      LDF      (R5),AC3      ; INITIAL AC FLOAT NUMBER
9267      035726      172637      002036      LDF      PREVAC,AC2    ; AC2 SHOULD NOT CHANGE
9268      035732      170165      000020      LDFPS    20(R5)      ; INITIAL FPS
9269
9270      035736      171765      000004      MD1FI: MODF     4(R5),AC3 ; FRAC[(AC3)*(MEM)]->AC3
9271                                          ; INT[(AC3)*(MEM)]->LOST
9272
9273      035742      170237      002000      STFPS    FPS          ; STORE FPS AFTER
9274      035746      170337      002002      STST     FEC          ; STORE FEC/FEA AFTER
9275      035752      174337      001170      STF      AC3,$REGD    ; STORE FRAC PART
9276      035756      174237      001174      STF      AC2,$REG2    ; STORE UNCHANGED AC3
9277
9278      035762      023765      002000      000022      CMP      FPS,22(R5)   ; CHECK FPS
9279      035770      001401      BEQ      65$         ; FPS IS OK
9280      035772      104005      ERROR    5           ; FPS BAD
9281      035774      005765      000024      65$:  TST      24(R5)   ; DOES FEC/FEA APPLY?
9282      036000      100014      BPL      66$         ; NO - SKIP TEST
9283      036002      012737      035736      002014      MOV      #MD1FI,EXPFEA ; GET EXPECTED FEA
9284      036010      123765      002002      000024      CMPB     FEC,24(R5)   ; COMPARE FEC-S
9285      036016      001004      BNE      64$         ; NOT EQUAL
9286      036020      023737      002004      002014      CMP      FEA,EXPFEA   ; COMPARE FEA-S
9287      036026      001401      BEQ      66$         ; FEC, FEA OK
9288      036030      104015      64$:  ERROR    15     ; FEC OR FEA ARE BAD
9289      036032      66$:
9290
9291      ; CHECK FRACTION PART
9292      036032      023765      001170      000010      CMP      $REGD,10(R5) ; 1ST WORD OF RESULT CHECK?
9293      036040      001004      BNE      67$         ; NO
9294      036042      023765      001172      000012      CMP      $REG1,12(R5) ; 2ND WORD OF RESULT CHECK?
9295      036050      001401      BEQ      68$         ; ALL WORDS OK
9296      036052      104027      67$:  ERROR    27     ; NUMBERS NOT EQUAL
9297      036054      68$:
9298
9299      ; CHECK UNCHANGED PART
9300      036054      023765      001174      000014      CMP      $REG2,14(R5) ; 1ST WORD OF RESULT CHECK?
9301      036062      001004      BNE      69$         ; NO
9302      036064      023765      001176      000016      CMP      $REG3,16(R5) ; 2ND WORD OF RESULT CHECK?
9303      036072      001401      BEQ      70$         ; ALL WORDS OK
9304      036074      104030      69$:  ERROR    30     ; NUMBERS NOT EQUAL
9305      036076      70$:
9306
9307      036076      000207      RTS      PC          ; RETURN TO TEST CALLER
9308
9309
9310

```

;;*****

```

.SBTTL SUBR TO TEST THE MODD INSTRUCTION, USING 1 ACCUMULATOR

9311
9312
9313 036100
9314 036100 012700 000023
9315 036104 010501
9316 036106 012702 001230
9317 036112 012122
9318 036114 077002
9319 036116 012737 036124 001114
9320
9321 036124 170011
9322 036126 172515
9323 036130 172437 002036
9324 036134 170165 000040
9325
9326 036140 171565 000010
9327
9328
9329 036144 170237 002000
9330 036150 170337 002002
9331 036154 174137 001170
9332 036160 174037 001200
9333
9334 036164 023765 002000 000042
9335 036172 001401
9336 036174 104010
9337 036176 005765 000044
9338 036202 100014
9339 036204 012737 036140 002014
9340 036212 123765 002002 000044
9341 036220 001004
9342 036222 023737 002004 002014
9343 036230 001401
9344 036232 104020
9345 036234
9346
9347
9348 036234 023765 001170 000020
9349 036242 001014
9350 036244 023765 001172 000022
9351 036252 001010
9352 036254 023765 001174 000024
9353 036262 001004
9354 036264 023765 001176 000026
9355 036272 001401
9356 036274 104031
9357 036276
9358
9359
9360 036276 023765 001200 000030
9361 036304 001014
9362 036306 023765 001202 000032
9363 036314 001010
9364 036316 023765 001204 000034
9365 036324 001004
9366 036326 023765 001206 000036

MD1DT:
MOV #23,R0 ; LOAD $TMP0-22
MOV R5,R1 ; WITH TEST DATA SETS
MOV #TMP0,R2 ; FOR DISPLAY LATER
MOV (R1)+,(R2)+
SOB RO,-2
MOV #MD1DL,$LPERR ; ERROR LOOPING ADDRESS

MD1DL:
SETD ; D MODE
LDC (R5),AC1 ; INITIAL AC FLOAT NUMBER
LDD PREVAC,ACD ; ACD SHOULD NOT CHANGE
LDFPS 40(R5) ; INITIAL FPS

MD1DI:
MODD 10(R5),AC1 ; FRAC[(AC1)*(MEM)]->AC1
; INT[(AC1)*(MEM)]->LOST

STFPS FPS ; STORE FPS AFTER
STST FEC ; STORE FEC/FEA AFTER
STD AC1,$REG0 ; STORE FRAC PART
STD ACD,$REG4 ; STORE UNCHANGED ACD

CMP FPS,42(R5) ; CHECK FPS
BEQ 65$ ; FPS IS OK
ERROR 10 ; FPS BAD
TST 44(R5) ; DOES FEC/FEA APPLY?
BPL 66$ ; NO - SKIP TEST
MOV #MD1DI,EXPFEA ; GET EXPECTED FEA
CMPB FEC,44(R5) ; COMPARE FEC-S
BNE 64$ ; NOT EQUAL
CMP FEA,EXPFEA ; COMPARE FEA-S
BEQ 66$ ; FEC, FEA OK
ERROR 20 ; FEC OR FEA ARE BAD

; CHECK FRACTION PART OF RESULT
CMP $REG0,20(R5) ; 1ST WORD OF RESULT CHECK?
BNE 67$ ; NO
CMP $REG1,22(R5) ; 2ND WORD OF RESULT CHECK?
BNE 67$ ; NO
CMP $REG2,24(R5) ; 3RD WORD OF RESULT CHECK?
BNE 67$ ; NO
CMP $REG3,26(R5) ; 4TH WORD OF RESULT CHECK?
BEQ 68$ ; ALL WORDS OK
ERROR 31 ; NUMBERS NOT EQUAL

; CHECK UNCHANGED PART
CMP $REG4,30(R5) ; 1ST WORD OF RESULT CHECK?
BNE 69$ ; NO
CMP $REG5,32(R5) ; 2ND WORD OF RESULT CHECK?
BNE 69$ ; NO
CMP $REG6,34(R5) ; 3RD WORD OF RESULT CHECK?
BNE 69$ ; NO
CMP $REG7,36(R5) ; 4TH WORD OF RESULT CHECK?

```


J15

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 193
SUBR TO TEST THE MODD INSTRUCTION, USING 1 ACCUMULATOR

9367 036334 001401
9368 036336 104032
9369 036340
9370
9371 036340 000207

69\$: BEQ 70\$; ALL WORDS OK
70\$: ERROR 32 ; NUMBERS NOT EQUAL
RTS PC ; RETURN TO TEST CALLER

```

.SBTTL SUBR TO TEST THE LDCDF INSTRUCTION

LCDFT:
9372          036342      012700  000011      MOV      #11,R0      ; LOAD $TMPD-10
9373          036342      010501      MOV      R5,R1      ; WITH TEST DATA SETS
9374          036346      012702  001230      MOV      #TMPD,R2   ; FOR DISPLAY LATER
9375          036346      012122      MOV      (R1)+,(R2)+
9376          036354      077002      SOB      R0,-2
9377          036356      012737  036366  001114      MOV      #LCDFL,$LPERR ; ERROR LOOPING ADDRESS
9378          036366      170011      LCDFL:  SETD      ; D MODE
9379          036370      172537  002036      LDF      PREVAC,AC1 ; PREV CONTENTS TO ACC; FOR FEC-14 TEST
9380          036374      170165  000014      LDFPS   14(R5)      ; INITIAL FPS
9381          036400      177515      LCDFI:  LDCDF   (R5),AC1 ; DTOF[(MEM)]->AC1
9382          036402      170237  002000      STFPS   FPS          ; STORE FPS AFTER
9383          036406      170337  002002      STST    FEC          ; STORE FEC/FEA AFTER
9384          036412      174137  001170      STF     AC1,$REGO    ; STORE RESULT
9385          036416      023765  002000  000016      CMP     FPS,16(R5)   ; CHECK FPS
9386          036424      001401      BEQ     65$          ; FPS IS OK
9387          036426      104004      ERROR   4            ; FPS BAD
9388          036430      005765  000020  65$:  TST     20(R5)     ; DOES FEC/FEA APPLY?
9389          036434      100014      BPL     66$          ; NO - SKIP TEST
9390          036436      012737  036400  002014      MOV     #LCDFI,EXPFEH ; GET EXPECTED FEA
9391          036444      123765  002002  000020      CMPB   FEC,20(R5)   ; COMPARE FEC-S
9392          036452      001004      BNE     64$          ; NOT EQUAL
9393          036454      023737  002004  002014      CMP     FEA,EXPFEA  ; COMPARE FEA-S
9394          036462      001401      BEQ     66$          ; FEC, FEA OK
9395          036464      104014      ERROR   14          ; FEC OR FEA ARE BAD
9396          036466      023765  001170  000010      CMP     $REGO,10(R5) ; 1ST WORD OF RESULT CHECK?
9397          036474      001004      BNE     67$          ; NO
9398          036476      023765  001172  000012      CMP     $REG1,12(R5) ; 2ND WORD OF RESULT CHECK?
9399          036504      001401      BEQ     68$          ; ALL WORDS OK
9400          036506      104033  67$:  ERROR   33          ; NUMBERS NOT EQUAL
9401          036510      000207  68$:
9402          036510      000207      RTS     PC           ; RETURN TO TEST CALLER
9403          036512
9404          036512
9405          036512
9406          036512
9407          036512
9408          036512
9409          036512
9410          036512
9411          036512
9412          036512
9413          036512
9414          036512
9415          036512
9416          036512
9417          036512
9418          036512
9419          036512      012700  000011      LCFDT:  MOV      #11,R0      ; LOAD $TMPD-10
9420          036516      010501      MOV      R5,R1      ; WITH TEST DATA SETS
9421          036520      012702  001230      MOV      #TMPD,R2   ; FOR DISPLAY LATER
9422          036524      012122      MOV      (R1)+,(R2)+
9423          036526      077002      SOB      R0,-2
9424          036530      012737  036536  001114      MOV      #LCDFL,$LPERR ; ERROR LOOPING ADDRESS
9425          036536      170011      LCFDL:  SETD      ; D MODE
9426          036540      172637  002036      LDD     PREVAC,AC2 ; PREV CONTENTS TO ACC; FOR FEC-14 TEST

```

.SBTTL SUBR TO TEST THE LDCFD INSTRUCTION

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 195
SUBR TO TEST THE LDCFD INSTRUCTION

9428	036544	170165	000014		LDFPS	14(R5)		; INITIAL FPS
9429								
9430	036550	177615			LCFDI:	LDCFD	(R5),AC2	; FTOD((MEM))->AC2
9431								
9432	036552	170237	002000		STFPS	FPS		; STORE FPS AFTER
9433	036556	170337	002002		STST	FEC		; STORE FEC/FEA AFTER
9434	036562	174237	001170		STD	AC2,\$REGO		; STORE RESULT
9435								
9436	036566	023765	002000	000016	CMP	FPS,16(R5)		; CHECK FPS
9437	036574	001401			BEQ	65\$; FPS IS OK
9438	036576	104004			ERROR	4		; FPS BAD
9439	036600	005765	000020		65\$: TST	20(R5)		; DOES FEC/FEA APPLY?
9440	036604	100014			BPL	66\$; NO - SKIP TEST
9441	036606	012737	036550	002014	MOV	8LCFDI,EXPFEA		; GET EXPECTED FEA
9442	036614	123765	002002	000020	CMPB	FEC,20(R5)		; COMPARE FEC-S
9443	036622	001004			BNE	64\$; NOT EQUAL
9444	036624	023737	002004	002014	CMP	FEA,EXPFEA		; COMPARE FEA-S
9445	036632	001401			BEQ	66\$; FEC, FEA OK
9446	036634	104014			64\$: ERROR	14		; FEC OR FEA ARE BAD
9447	036636				66\$:			
9448								
9449	036636	023765	001170	000004	CMP	\$REG0,4(R5)		; 1ST WORD OF RESULT CHECK?
9450	036644	001014			BNE	67\$; NO
9451	036646	023765	001172	000006	CMP	\$REG1,6(R5)		; 2ND WORD OF RESULT CHECK?
9452	036654	001010			BNE	67\$; NO
9453	036656	023765	001174	000010	CMP	\$REG2,10(R5)		; 3RD WORD OF RESULT CHECK?
9454	036664	001004			BNE	67\$; NO
9455	036666	023765	001176	000012	CMP	\$REG3,12(R5)		; 4TH WORD OF RESULT CHECK?
9456	036674	001401			BEQ	68\$; ALL WORDS OK
9457	036676	104034			67\$: ERROR	34		; NUMBERS NOT EQUAL
9458	036700				68\$:			
9459								
9460	036700	000207			RTS	PC		; RETURN TO TEST CALLER
9461								

M15

FPU ADVANCED INSTR TESTS
 DDFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 196
 SUBR TO TEST THE STCDF INSTRUCTION

```

9462          .SBTTL  SUBR TO TEST THE STCDF INSTRUCTION
9463
9464 036702          SCDFI:  MOV      #11,R0          ; LOAD $TMPD-10
9465 036702 012700 000011      MOV      R5,R1          ; WITH TEST DATA SETS
9466 036706 010501          MOV      #TMPD,R2          ; FOR DISPLAY LATER
9467 036710 012702 001230      MOV      (R1)+(R2)+
9468 036714 012122          SOB      R0,-2
9469 036716 077002          MOV      #SCDFL,$LPERR ; ERROR LOOPING ADDRESS
9470 036720 012737 036726 001114
9471
9472 036726 170011          SCDFL:  SETD           ; D MODE
9473 036730 172715          LDD      (R5),AC3      ; INITIAL AC FLOAT NUMBER
9474 036732 170165 000014      LDFPS   14(R5)        ; INITIAL FPS
9475
9476 036736 176337 001170          SCDFI:  STCDF      AC3,$REGO ; DTOF[(AC3)]->MEM
9477
9478 036742 170237 002000          STFPS   FPS          ; STORE FPS AFTER
9479 036746 170337 002002          STST    FEC          ; STORE FEC/FEA AFTER
9480
9481 036752 023765 002000 000016      CMP      FPS,16(R5)   ; CHECK FPS
9482 036760 001401          BEQ      65$         ; FPS IS OK
9483 036762 104004          ERROR   4           ; FPS BAD
9484 036764 005765 000020          65$:  TST      20(R5)   ; DOES FEC/FEA APPLY?
9485 036770 100014          BPL      66$         ; NO - SKIP TEST
9486 036772 012737 036736 002014      MOV      #SCDFI,EXPFEA ; GET EXPECTED FEA
9487 037000 123765 002002 000020      CMPB    FEC,20(R5)   ; COMPARE FEC-S
9488 037006 001004          BNE      64$         ; NOT EQUAL
9489 037010 023737 002004 002014      CMP      FEA,EXPFEA  ; COMPARE FEA-S
9490 037016 001401          BEQ      66$         ; FEC, FEA OK
9491 037020 104014          64$:  ERROR   14     ; FEC OR FEA ARE BAD
9492 037022          66$:
9493
9494 037022 023765 001170 000010      CMP      $REGO,10(R5) ; 1ST WORD OF RESULT CHECK?
9495 037030 001004          BNE      67$         ; NO
9496 037032 023765 001172 000012      CMP      $REG1,12(R5) ; 2ND WORD OF RESULT CHECK?
9497 037040 001401          BEQ      68$         ; ALL WORDS OK
9498 037042 104033          67$:  ERROR   33     ; NUMBERS NOT EQUAL
9499 037044          68$:
9500
9501 037044 000207          RTS      PC          ; RETURN TO TEST CALLER
9502
9503
9504          ;*****
9505          .SBTTL  SUBR TO TEST THE STCDF INSTRUCTION
9506
9507 037046          SCFDI:  MOV      #12,R0          ; LOAD $TMPD-11
9508 037046 012700 000012      MOV      R5,R1          ; WITH TEST DATA SETS
9509 037052 010501          MOV      #TMPD,R2          ; FOR DISPLAY LATER
9510 037054 012702 001230      MOV      (R1)+(R2)+
9511 037060 012122          SOB      R0,-2
9512 037062 077002          MOV      #SCFDL,$LPERR ; ERROR LOOPING ADDRESS
9513 037064 012737 037072 001114
9514
9515 037072 170011          SCFDL:  SETD           ; USE D MODE
9516 037074 172415          LDD      (R5),AC0    ; INITIAL F FLOAT NUM, FOLLOW W/JUNK
9517 037076 170165 000020      LDFPS   20(R5)        ; INITIAL FPS
  
```

N15

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 197
SUBR TO TEST THE STCFD INSTRUCTION

9518										
9519	037102	176037	001170		STCFD	ACO,\$REG0			; FTOD((ACO))->MEM	
9520										
9521	037106	170237	002000		STFPS	FPS			; STORE FPS AFTER	
9522										
9523	037112	023765	002000	000022	CMP	FPS,22(R5)			; CHECK FPS OK	
9524	037120	001401			BEQ	64\$; OK BRANCH	
9525	037122	104005			ERROR	5			; FPS BAD	
9526	037124							64\$:		
9527										
9528	037124	023765	001170	000010	CMP	\$REG0,10(R5)			; 1ST WORD OF RESULT CHECK?	
9529	037132	001014			BNE	65\$; NO	
9530	037134	023765	001172	000012	CMP	\$REG1,12(R5)			; 2ND WORD OF RESULT CHECK?	
9531	037142	001010			BNE	65\$; NO	
9532	037144	023765	001174	000014	CMP	\$REG2,14(R5)			; 3RD WORD OF RESULT CHECK?	
9533	037152	001004			BNE	65\$; NO	
9534	037154	023765	001176	000016	CMP	\$REG3,16(R5)			; 4TH WORD OF RESULT CHECK?	
9535	037162	001401			BEQ	66\$; ALL WORDS OK	
9536	037164	104035			ERROR	35			; NUMBERS NOT EQUAL	
9537	037166							66\$:		
9538										
9539	037166	000207			RTS	PC			; RETURN TO TEST CALLER	

```

9540          .SBTTL SUBR TO TEST THE LDCIF INSTRUCTION
9541
9542 037170          LCIFT:
9543 037170 012700 000005      MOV     #5,R0          ; LOAD $TMPO-4
9544 037174 010501          MOV     R5,R1          ; WITH TEST DATA SETS
9545 037176 012702 001230      MOV     #STMPO,R2      ; FOR DISPLAY LATER
9546 037202 012122          MOV     (R1)+,(R2)+
9547 037204 077002          SOB     RO,-2
9548 037206 012737 037214 001114  MOV     #LCIFL,$LPERR ; ERROR LOOPING ADDRESS
9549
9550 037214 170165 000006      LCIFL: LDFPS 6(R5)      ; INITIAL FPS
9551
9552 037220 177215          LDCIF  (R5),AC2      ; F[(MEM)]->AC2
9553
9554 037222 170237 002000      STFPS  FPS          ; STORE FPS AFTER
9555 037226 174237 001170      STF     AC2,$REGO     ; STORE RESULT
9556
9557 037232 023765 002000 000010  CMP     FPS,10(R5)    ; CHECK FPS OK
9558 037240 001401          BEQ     64$          ; OK, BRANCH
9559 037242 104001          ERROR  1           ; FPS BAD
9560 037244          64$:
9561
9562 037244 023765 001170 000002  CMP     $REGO,2(R5)   ; 1ST WORD OF RESULT CHECK?
9563 037252 001004          BNE     65$          ; NO
9564 037254 023765 001172 000004  CMP     $REG1,4(R5)   ; 2ND WORD OF RESULT CHECK?
9565 037262 001401          BEQ     66$          ; ALL WORDS OK
9566 037264 104036          65$: ERROR 36      ; NUMBERS NOT EQUAL
9567 037266          66$:
9568
9569 037266 000207          RTS     PC           ; RETURN TO TEST CALLER
9570
9571
9572          ;:*****
9573          .SBTTL SUBR TO TEST THE LDCID INSTRUCTION
9574
9575 037270          LCIDT:
9576 037270 012700 000007      MOV     #7,R0          ; LOAD $TMPO-6
9577 037274 010501          MOV     R5,R1          ; WITH TEST DATA SETS
9578 037276 012702 001230      MOV     #STMPO,R2      ; FOR DISPLAY LATER
9579 037302 012122          MOV     (R1)+,(R2)+
9580 037304 077002          SOB     RO,-2
9581 037306 012737 037314 001114  MOV     #LCIDL,$LPERR ; ERROR LOOPING ADDRESS
9582
9583 037314 170165 000012      LCIDL: LDFPS 12(R5)   ; INITIAL FPS
9584
9585 037320 177315          LDCID  (R5),AC3      ; D[(MEM)]->AC3
9586
9587 037322 170237 002000      STFPS  FPS          ; STORE FPS AFTER
9588 037326 174337 001170      STD     AC3,$REGO     ; STORE RESULT
9589
9590 037332 023765 002000 000014  CMP     FPS,14(R5)    ; CHECK FPS OK
9591 037340 001401          BEQ     64$          ; OK, BRANCH
9592 037342 104003          ERROR  3           ; FPS BAD
9593 037344          64$:
9594
9595 037344 023765 001170 000002  CMP     $REGO,2(R5)   ; 1ST WORD OF RESULT CHECK?

```

```

9596 037352 001014          BNE      65$          ; NO
9597 037354 023765 001172 000004    CMP      $REG1,4(R5) ; 2ND WORD OF RESULT CHECK?
9598 037362 001010          BNE      65$          ; NO
9599 037364 023765 001174 000006    CMP      $REG2,6(R5) ; 3RD WORD OF RESULT CHECK?
9600 037372 001004          BNE      65$          ; NO
9601 037374 023765 001176 000010    CMP      $REG3,10(R5); 4TH WORD OF RESULT CHECK?
9602 037402 001401          BEQ      66$          ; ALL WORDS OK
9603 037404 104037          65$:    ERROR      37    ; NUMBERS NOT EQUAL
9604 037406          66$:
9605
9606 037406 000207          RTS      PC          ; RETURN TO TEST CALLER
9607
9608
9609

```

:SBTTL SUBR TO TEST THE LDCLF INSTRUCTION

```

9610
9611
9612 037410          LCLFT:
9613 037410 012700 000006    MOV      #6,R0      ; LOAD $TMPO-5
9614 037414 010501          MOV      R5,R1      ; WITH TEST DATA SETS
9615 037416 012702 001230    MOV      #TMPO,R2   ; FOR DISPLAY LATER
9616 037422 012122          MOV      (R1)+,(R2)+
9617 037424 077002          SOB      R0,-2
9618 037426 012737 037434 001114    MOV      #LCLFL,$LPERR ; ERROR LOOPING ADDRESS
9619
9620 037434 170165 000010    LCLFL: LDFPS      10(R5) ; INITIAL FPS
9621
9622 037440 177015          LDCLF   (R5),ACD   ; F[(MEM)(MEM)]->ACD
9623
9624 037442 170237 002000    STFPS   FPS        ; STORE FPS AFTER
9625 037446 174037 001170    STF     ACD,$REG0  ; STORE RESULT
9626
9627 037452 023765 002000 000012    CMP     FPS,12(R5) ; CHECK FPS OK
9628 037460 001401          BEQ     64$        ; OK BRANCH
9629 037462 104002          ERROR  2          ; FPS BAD
9630 037464          64$:
9631
9632 037464 023765 001170 000004    CMP     $REG0,4(R5) ; 1ST WORD OF RESULT CHECK?
9633 037472 001004          BNE     65$        ; NO
9634 037474 023765 001172 000006    CMP     $REG1,6(R5) ; 2ND WORD OF RESULT CHECK?
9635 037502 001401          BEQ     66$        ; ALL WORDS OK
9636 037504 104040          65$:    ERROR      40    ; NUMBERS NOT EQUAL
9637 037506          66$:
9638
9639 037506 000207          RTS      PC          ; RETURN TO TEST CALLER
9640
9641
9642
9643
9644

```

:SBTTL SUBR TO TEST THE LCLD INSTRUCTION

```

9645 037510          LCLDT:
9646 037510 012700 000010    MOV      #10,R0     ; LOAD $TMPO-7
9647 037514 010501          MOV      R5,R1     ; WITH TEST DATA SETS
9648 037516 012702 001230    MOV      #TMPO,R2   ; FOR DISPLAY LATER
9649 037522 012122          MOV      (R1)+,(R2)+
9650 037524 077002          SOB      R0,-2
9651 037526 012737 037534 001114    MOV      #LCLDL,$LPERR ; ERROR LOOPING ADDRESS

```

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 200
SUBR TO TEST THE LDCLD INSTRUCTION

9652										
9653	037534	170165	000014		LCLDL:	LDFPS	14(R5)			; INITIAL FPS
9654										
9655	037540	177115				LDCLD	(R5),AC1			; D[(MEM)(MEM)]->AC1
9656										
9657	037542	170237	002000			STFPS	FPS			; STORE FPS AFTER
9658	037546	174137	001170			STD	AC1,\$REGO			; STORE RESULT
9659										
9660	037552	023765	002000	000016		CMP	FPS,16(R5)			; CHECK FPS OK
9661	037560	001401				BEQ	64\$; OK, BRANCH
9662	037562	104004				ERROR	4			; FPS BAD
9663	037564				64\$:					
9664										
9665	037564	023765	001170	000004		CMP	\$REGO,4(R5)			; 1ST WORD OF RESULT CHECK?
9666	037572	001014				BNE	65\$; NO
9667	037574	023765	001172	000006		CMP	\$REG1,6(R5)			; 2ND WORD OF RESULT CHECK?
9668	037602	001010				BNE	65\$; NO
9669	037604	023765	001174	000010		CMP	\$REG2,10(R5)			; 3RD WORD OF RESULT CHECK?
9670	037612	001004				BNE	65\$; NO
9671	037614	023765	001176	000012		CMP	\$REG3,12(R5)			; 4TH WORD OF RESULT CHECK?
9672	037622	001401				BEQ	66\$; ALL WORDS OK
9673	037624	104041			65\$:	ERROR	41			; NUMBERS NOT EQUAL
9674	037626				66\$:					
9675										
9676	037626	000207				RTS	PC			; RETURN TO TEST CALLER

E16

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 201
SUBR TO TEST THE STCFI INSTRUCTION

```

9677          .SBTTL  SUBR TO TEST THE STCFI INSTRUCTION
9678
9679 037630          SCFIT:
9680 037630 012700 000006      MOV    #6,R0          ; LOAD $TMP0-5
9681 037634 010501          MOV    R5,R1          ; WITH TEST DATA SETS
9682 037636 012702 001230      MOV    #TMP0,R2        ; FOR DISPLAY LATER
9683 037642 012122          MOV    (R1)+(R2)+
9684 037644 077002          SOB    R0,-2
9685 037646 012737 037654 001114  MOV    #SCFIL,$LPERR ; ERROR LOOPING ADDRESS
9686
9687 037654 170001          SCFIL: SETF          ; F MODE
9688 037656 172715          LDF    (R5),AC3      ; INITIAL AC FLOAT NUMBER
9689 037660 170165 000006      LDFPS 6(R5)         ; INITIAL FPS
9690
9691 037664 175737 001170          SCFII: STCFI AC3,$REG0 ; I((AC3))->MEM
9692
9693 037670 013737 177776 001172      MOV    #PS,$REG1    ; SAVE CC-S
9694 037676 170237 002000          STFPS FPS          ; STORE FPS AFTER
9695 037702 170337 002002          STST  FEC          ; STORE FEC/FEA AFTER
9696
9697 037706 023765 002000 000010      CMP    FPS,10(R5)   ; CHECK FPS
9698 037714 001401          BEQ    65$         ; FPS IS OK
9699 037716 104001          ERROR 1          ; FPS BAD
9700 037720 005765 000012          65$: TST    12(R5)      ; DOES FEC/FEA APPLY?
9701 037724 100014          BPL    66$         ; NO - SKIP TEST
9702 037726 012737 037664 002014      MOV    #SCFII,EXPFEA ; GET EXPECTED FEA
9703 037734 123765 002002 000012      CMPB  FEC,12(R5)   ; COMPARE FEC-S
9704 037742 001004          BNE    64$         ; NOT EQUAL
9705 037744 023737 002004 002014      CMP    FEA,EXPFEA  ; COMPARE FEA-S
9706 037752 001401          BEQ    66$         ; FEC, FEA OK
9707 037754 104011          64$: ERROR 11     ; FEC OR FEA ARE BAD
9708 037756          66$:
9709
9710 037756 013737 002000 001174      MOV    FPS,$REG2   ; GET FPS, PS CC BITS ONLY
9711 037764 042737 177760 001172      BIC    #CCONLY,$REG1 ;
9712 037772 042737 177760 001174      BIC    #CCONLY,$REG2 ;
9713 040000 023737 001172 001174      CMP    $REG1,$REG2 ; CC-S COPIED?
9714 040006 001401          BEQ    67$         ;
9715 040010 104054          ERROR 54         ; NOT EQUAL, SIGNAL ERROR
9716 040012          67$:
9717
9718 040012 023765 001170 000004      CMP    $REG0,4(R5) ; INTEGER RESULT CHECK?
9719 040020 001401          BEQ    68$         ;
9720 040022 104042          ERROR 42         ; NOT EQUAL, SIGNAL ERROR
9721 040024          68$:
9722
9723 040024 000207          RTS    PC          ; RETURN TO TEST CALLER
9724
9725
9726 ;:*****
9727 .SBTTL  SUBR TO TEST THE STCFI INSTRUCTION
9728
9729          SCFIT:
9730 040026 012700 000010      MOV    #10,R0       ; LOAD $TMP0-7
9731 040032 010501          MOV    R5,R1       ; WITH TEST DATA SETS
9732 040034 012702 001230      MOV    #TMP0,R2    ; FOR DISPLAY LATER

```

F16

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 202
SUBR TO TEST THE STCDI INSTRUCTION

```

9733 040040 012122          MOV      (R1)+,(R2)+      ;
9734 040042 077002          SOB      RO,-2           ;
9735 040044 012737 040052 001114  MOV      #SCDIL,$LPERR   ; ERROR LOOPING ADDRESS
9736
9737 040052 170011          SCDIL:  SETD           ; D MODE
9738 040054 172415          LDD      (R5),AC0       ; INITIAL AC FLOAT NUMBER
9739 040056 170165 000012  LDFPS   12(R5)         ; INITIAL FPS
9740
9741 040062 175437 001170  SCDII:  STCDI        AC0,$REG0 ; I[(AC0)]->MEM
9742
9743 040066 013737 177776 001172  MOV      #FPS,$REG1     ; SAVE CC-S
9744 040074 170237 002000  STFPS   FPS            ; STORE FPS AFTER
9745 040100 170337 002002  STST    FEC            ; STORE FEC/FEA AFTER
9746
9747 040104 023765 002000 000014  CMP      FPS,14(R5)     ; CHECK FPS
9748 040112 001401          BEQ      65$           ; FPS IS OK
9749 040114 104003          ERROR   3             ; FPS BAD
9750 040116 005765 000016 65$:    TST      16(R5)     ; DOES FEC/FEA APPLY?
9751 040122 100014          BPL      66$           ; NO - SKIP TEST
9752 040124 012737 040062 002014  MOV      #SCDII,EXPFEA ; GET EXPECTED FEA
9753 040132 123765 002002 000016  CMPB    FEC,16(R5)     ; COMPARE FEC-S
9754 040140 001004          BNE      64$           ; NOT EQUAL
9755 040142 023737 002004 002014  CMP      FEA,EXPFEA    ; COMPARE FEA-S
9756 040150 001401          BEQ      66$           ; FEC, FEA OK
9757 040152 104013          64$:    ERROR   13      ; FEC OR FEA ARE BAD
9758 040154          66$:
9759
9760 040154 013737 002000 001174  MOV      FPS,$REG2     ; GET FPS, PS CC BITS ONLY
9761 040162 042737 177760 001172  BIC      #CCONLY,$REG1 ;
9762 040170 042737 177760 001174  BIC      #CCONLY,$REG2 ;
9763 040176 023737 001172 001174  CMP      $REG1,$REG2   ; CC-S COPIED?
9764 040204 001401          BEQ      67$           ;
9765 040206 104054          ERROR   54            ; NOT EQUAL, SIGNAL ERROR
9766 040210          67$:
9767
9768 040210 023765 001170 000010  CMP      $REG0,10(R5)  ; INTEGER RESULT CHECK?
9769 040216 001401          BEQ      68$           ;
9770 040220 104043          ERROR   43            ; NOT EQUAL, SIGNAL ERROR
9771 040222          68$:
9772
9773 040222 000207          RTS      PC            ; RETURN TO TEST CALLER
9774
9775
9776
9777          ;:*****
          ;SBTTL SUBR TO TEST THE STCFL INSTRUCTION
9778
9779          SCFLT:
9780 040224 012700 000007          MOV      #7,RO         ; LOAD $TMP0-6
9781 040230 010501          MOV      R5,R1        ; WITH TEST DATA SETS
9782 040232 012702 001230          MOV      #TMP0,R2     ; FOR DISPLAY LATER
9783 040236 012122          MOV      (R1)+,(R2)+ ;
9784 040240 077002          SOB      RO,-2         ;
9785 040242 012737 040250 001114  MOV      #SCFLL,$LPERR ; ERROR LOOPING ADDRESS
9786
9787          SCFLL:  SETF           ; F MODE
9788 040250 170001          LDF      (R5),AC1     ; INITIAL AC FLOAT NUMBER
9789 040252 172515

```

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 203
SUBR TO TEST THE STCFL INSTRUCTION

```

9789 040254 170165 000010          LDFPS 10(R5)          ; INITIAL FPS
9790
9791 040260 175537 001170          SCFLI: STCFL AC1,$REG0 ; L((AC1))->MEM
9792
9793 040264 013737 177776 001174          MOV 2#PS,$REG2      ; SAVE CC-S
9794 040272 170237 002000          STFPS FPS           ; STORE FPS AFTER
9795 040276 170337 002002          STST FEC           ; STORE FEC/FEA AFTER
9796
9797 040302 023765 002000 000012          CMP  FPS,12(R5)     ; CHECK FPS
9798 040310 001401 65$          BEQ 65$            ; FPS IS OK
9799 040312 104002          ERROR 2           ; FPS BAD
9800 040314 005765 000014          65$: TST 14(R5)     ; DOES FEC/FEA APPLY?
9801 040320 100014          BPL 66$          ; NO - SKIP TEST
9802 040322 012737 040260 002014          MOV #SCFLI,EXPFEA  ; GET EXPECTED FEA
9803 040330 123765 002002 000014          CMPB FEC,14(R5)   ; COMPARE FEC-S
9804 040336 001004          BNE 64$          ; NOT EQUAL
9805 040340 023737 002004 002014          CMP  FEA,EXPFEA   ; COMPARE FEA-S
9806 040346 001401 66$          BEQ 66$          ; FEC, FEA OK
9807 040350 104012          64$: ERROR 12     ; FEC OR FEA ARE BAD
9808 040352          66$:
9809
9810 040352 013737 002000 001176          MOV  FPS,$REG3     ; GET FPS, PS CC BITS ONLY
9811 040360 042737 177760 001174          BIC #CCONLY,$REG2 ;
9812 040366 042737 177760 001176          BIC #CCONLY,$REG3 ;
9813 040374 023737 001174 001176          CMP  $REG2,$REG3  ; CC-S COPIED?
9814 040402 001401 67$          BEQ 67$          ;
9815 040404 104055          ERROR 55        ; NOT EQUAL, SIGNAL ERROR
9816 040406          67$:
9817
9818 040406 023765 001170 000004          CMP  $REG0,4(R5)  ; 1ST WORD OF RESULT CHECK?
9819 040414 001004 68$          BNE 68$          ; NO
9820 040416 023765 001172 000006          CMP  $REG1,6(R5)  ; 2ND WORD OF RESULT CHECK?
9821 040424 001401 69$          BEQ 69$          ; ALL WORDS OK
9822 040426 104044          68$: ERROR 44   ; NUMBERS NOT EQUAL
9823 040430          69$:
9824
9825 040430 000207          RTS  PC          ; RETURN TO TEST CALLER
9826
9827
9828
9829
9830

```

::*****

.SBTTL SUBR TO TEST THE STCDL INSTRUCTION

```

9831 040432          SCDLT:
9832 040432 012700 000011          MOV  #11,R0       ; LOAD $TMP0-10
9833 040436 010501          MOV  R5,R1       ; WITH TEST DATA SETS
9834 040440 012702 001230          MOV  #TMP0,R2    ; FOR DISPLAY LATER
9835 040444 012122          MOV  (R1)+,(R2)+ ;
9836 040446 077002          SOB  R0,-2       ;
9837 040450 012737 040456 001114          MOV  #SCDLL,$LPERR ; ERROR LOOPING ADDRESS
9838
9839 040456 170011          SCDLL: SETD      ; D MODE
9840 040460 172615          LDD  (R5),AC2    ; INITIAL AC FLOAT NUMBER
9841 040462 170165 000014          LDFPS 14(R5)    ; INITIAL FPS
9842
9843 040466 175637 001170          SCDLI: STCDL AC2,$REG0 ; L((AC2))->MEM, MEM
9844

```

H16

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 204
SUBR TO TEST THE STCDL INSTRUCTION

9845	040472	013737	177776	001174	MOV	2#PS, \$REG2	:	SAVE CC-S
9846	040500	170237	002000		STFPS	FPS	:	STORE FPS AFTER
9847	040504	170337	002002		STST	FEC	:	STORE FEC/FEA AFTER
9848								
9849	040510	023765	002000	000016	CMP	FPS, 16(R5)	:	CHECK FPS
9850	040516	001401			BEQ	65\$:	FPS IS OK
9851	040520	104004			ERROR	4	:	FPS BAD
9852	040522	005765	000020		TST	20(R5)	:	DOES FEC/FEA APPLY?
9853	040526	100014			BPL	66\$:	NO - SKIP TEST
9854	040530	012737	040466	002014	MOV	#SCDLI, EXPFEA	:	GET EXPECTED FEA
9855	040536	123765	002002	000020	CMPB	FEC, 20(R5)	:	COMPARE FEC-S
9856	040544	001004			BNE	64\$:	NOT EQUAL
9857	040546	023737	002004	002014	CMP	FEA, EXPFEA	:	COMPARE FEA-S
9858	040554	001401			BEQ	66\$:	FEC, FEA OK
9859	040556	104014			ERROR	14	:	FEC OR FEA ARE BAD
9860	040560							
9861								
9862	040560	013737	002000	001176	MOV	FPS, \$REG3	:	GET FPS, PS CC BITS ONLY
9863	040566	042737	177760	001174	BIC	#CCONLY, \$REG2	:	
9864	040574	042737	177760	001176	BIC	#CCONLY, \$REG3	:	
9865	040602	023737	001174	001176	CMP	\$REG2, \$REG3	:	CC-S COPIED?
9866	040610	001401			BEQ	67\$:	
9867	040612	104055			ERROR	55	:	NOT EQUAL, SIGNAL ERROR
9868	040614							
9869								
9870	040614	023765	001170	000010	CMP	\$REG0, 10(R5)	:	1ST WORD OF RESULT CHECK?
9871	040622	001004			BNE	68\$:	NO
9872	040624	023765	001172	000012	CMP	\$REG1, 12(R5)	:	2ND WORD OF RESULT CHECK?
9873	040632	001401			BEQ	69\$:	ALL WORDS OK
9874	040634	104045			ERROR	45	:	NUMBERS NOT EQUAL
9875	040636							
9876								
9877	040635	000207			RTS	PC	:	RETURN TO TEST CALLER

FPU ADVANCED INSTR TESTS
 DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 205
 SUBR TO TEST THE LDEXP INSTRUCTION, F MODE

```

9878          .SBTTL  SUBR TO TEST THE LDEXP INSTRUCTION, F MODE
9879
9880 040640          LEXFT:
9881 040640 012700 000010      MOV    #10,R0          ; LOAD $TMP0-7
9882 040644 010501          MOV    R5,R1          ; WITH TEST DATA SETS
9883 040646 012702 001230      MOV    #TMP0,R2        ; FOR DISPLAY LATER
9884 040652 012122          MOV    (R1)+,(R2)+
9885 040654 077002          SOB    R0,-2
9886 040656 012737 040664 001114  MOV    #LEXFL,$LPERR ; ERROR LOOPING ADDRESS
9887
9888 040664 170001          LEXFL: SETF          ; F MODE
9889 040666 172515          LDF    (R5),AC1      ; INITIAL FLOAT NUMBER
9890 040670 170165 000012      LDFPS 12(R5)        ; INITIAL FPS
9891
9892 040674 176565 000010      LEXFI: LDEXP 10(R5),AC1 ; EXP: MEM -> AC1
9893
9894 040700 170237 002000      STFPS FPS          ; STORE FPS AFTER
9895 040704 170337 002002      STST  FEC          ; STORE FEC/FEA AFTER
9896
9897 040710 023765 002000 000014  CMP    FPS,14(R5)   ; CHECK FPS
9898 040716 001401          BEQ    65$          ; FPS IS OK
9899 040720 104003          ERROR 3           ; FPS BAD
9900 040722 005765 000016      65$:  TST    16(R5)   ; DOES FEC/FEA APPLY?
9901 040726 100014          BPL    66$          ; NO - SKIP TEST
9902 040730 012737 040674 002014  MOV    #LEXFI,EXPFEA ; GET EXPECTED FEA
9903 040736 123735 002002 000016  CMPB  FEC,16(R5)   ; COMPARE FEC-S
9904 040744 001004          BNE    64$          ; NOT EQUAL
9905 040746 023737 002004 002014  CMP    FEA,EXPFEA  ; COMPARE FEA-S
9906 040754 001401          BEQ    66$          ; FEC, FEA OK
9907 040756 104013          64$:  ERROR 13      ; FEC OR FEA ARE BAD
9908 040760          66$:
9909
9910 040760 174137 001170          STF    AC1,$REG0    ; STORE RESULTANT FLOAT NUMBER
9911 040764 023765 001170 000004  CMP    $REG0,4(R5) ; 1ST WORD OF RESULT CHECK?
9912 040772 001004          BNE    67$          ; NO
9913 040774 023765 001172 000006  CMP    $REG1,6(R5) ; 2ND WORD OF RESULT CHECK?
9914 041002 001401          BEQ    68$          ; ALL WORDS OK
9915 041004 104046          67$:  ERROR 46      ; NUMBERS NOT EQUAL
9916 041006          68$:
9917
9918 041006 000207          RTS    PC          ; RETURN TO TEST CALLER
9919
9920          ;:*****
9921          ;.SBTTL  SUBR TO TEST THE LDEXP INSTRUCTION, D MODE
9922
9923 041010          LEXDT:
9924 041010 012700 000014      MOV    #14,R0          ; LOAD $TMP0-13
9925 041014 010501          MOV    R5,R1          ; WITH TEST DATA SETS
9926 041016 012702 001230      MOV    #TMP0,R2        ; FOR DISPLAY LATER
9927 041022 012122          MOV    (R1)+,(R2)+
9928 041024 077002          SOB    R0,-2
9929 041026 012737 041034 001114  MOV    #LEXDL,$LPERR ; ERROR LOOPING ADDRESS
9930
9931 041034 170011          LEXDL: SETD         ; D MODE
9932 041036 172415          LDD    (R5),AC0    ; INITIAL FLOAT NUMBER
9933 041040 170165 000022      LDFPS 22(R5)        ; INITIAL FPS
  
```


K16

FPU ADVANCED INSTR TESTS
 DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 207
 SUBR TO TEST THE STEXP INSTRUCTION, F MODE

```

9967          .SBTTL  SUBR TO TEST THE STEXP INSTRUCTION, F MODE
9968
9969 041200          SEXFT:
9970 041200 012700 000005      MOV    #5,R0          ; LOAD $TMPD-4
9971 041204 010501          MOV    R5,R1          ; WITH TEST DATA SETS
9972 041206 012702 001230      MOV    #TMPD,R2       ; FOR DISPLAY LATER
9973 041212 012122          MOV    (R1)+(R2)+    ;
9974 041214 077002          SOB    R0,-2         ;
9975 041216 012737 041224 001114  MOV    #SEXFL,$LPERR ; ERROR LOOPING ADDRESS
9976
9977 041224 170001          SEXFL: SETF          ; F MODE
9978 041226 172615          LDF    (R5),AC2     ; INITIAL FLOAT NUMBER
9979 041230 170165 000006      LDFPS 6(R5)        ; INITIAL FPS
9980
9981 041234 175237 001170          SEXFI: STEXP AC2,$REGD ; EXP: AC2 -> MEM
9982
9983 041240 013737 177776 001172      MOV    #PS,$REG1    ; GET PS RIGHT AWAY, FOR CC BITS
9984 041246 170237 002000          STFPS FPS         ; STORE FPS AFTER
9985
9986 041252 023765 002000 000010      CMP    FPS,10(R5)   ; CHECK FPS OK
9987 041260 001401          BEQ    64$         ; OK, BRANCH
9988 041262 104001          ERROR 1           ; FPS BAD
9989 041264          64$:
9990
9991 041264 013737 002000 001174      MOV    FPS,$REG2    ; GET FPS, PS CC BITS ONLY
9992 041272 042737 177760 001172      BIC    #CCONLY,$REG1 ;
9993 041300 042737 177760 001174      BIC    #CCONLY,$REG2 ;
9994 041306 023737 001172 001174      CMP    $REG1,$REG2 ; CC-S COPIED?
9995 041314 001401          BEQ    65$         ;
9996 041316 104054          ERROR 54         ; NOT EQUAL, SIGNAL ERROR
9997 041320          65$:
9998
9999 041320 023765 001170 000004      CMP    $REGD,4(R5) ; EXP CHECK?
10000 041326 001401          BEQ    66$         ;
10001 041330 104050          ERROR 50         ; NOT EQUAL, SIGNAL ERROR
10002 041332          66$:
10003
10004 041332 000207          RTS    PC          ; RETURN TO TEST CALLER
10005
10006          ;*****
10007          .SBTTL  SUBR TO TEST THE STEXP INSTRUCTION, D MODE
10008
10009 041334          SEXDT:
10010 041334 012700 000007      MOV    #7,R0          ; LOAD $TMPD-6
10011 041340 010501          MOV    R5,R1          ; WITH TEST DATA SETS
10012 041342 012702 001230      MOV    #TMPD,R2       ; FOR DISPLAY LATER
10013 041346 012122          MOV    (R1)+(R2)+    ;
10014 041350 077002          SOB    R0,-2         ;
10015 041352 012737 041360 001114  MOV    #SEXDL,$LPERR ; ERROR LOOPING ADDRESS
10016
10017 041360 170011          SEXDL: SETD         ; D MODE
10018 041362 172715          LDD    (R5),AC3     ; INITIAL FLOAT NUMBER
10019 041364 170165 000012      LDFPS 12(R5)       ; INITIAL FPS
10020
10021 041370 175337 001170          SEXDI: STEXP AC3,$REGD ; EXP: AC3 -> MEM
10022

```

L16

FPU ADVANCED INSTR TESTS
 DQFPBR.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 208
 SUBR TO TEST THE STEXP INSTRUCTION, D MODE

10023	041374	013737	177776	001172	MOV	2#PS, \$REG1	; GET PS RIGHT AWAY
10024	041402	170237	002000		STFPS	FPS	; STORE FPS AFTER
10025							
10026	041406	023765	002000	000014	CMP	FPS, 14(R5)	; CHECK FPS OK
10027	041414	001401			BEQ	64\$; OK BRANCH
10028	041416	104003			ERROR	3	; FPS BAD
10029	041420			64\$:			
10030							
10031	041420	013737	002000	001174	MOV	FPS, \$REG2	; GET FPS, PS CC BITS ONLY
10032	041426	042737	177760	001172	BIC	#CCONLY, \$REG1	
10033	041434	042737	177760	001174	BIC	#CCONLY, \$REG2	
10034	041442	023737	001172	001174	CMP	\$REG1, \$REG2	; CC-S COPIED?
10035	041450	001401			BEQ	65\$	
10036	041452	104054			ERROR	54	; NOT EQUAL, SIGNAL ERROR
10037	041454			65\$:			
10038							
10039	041454	023765	001170	000010	CMP	\$REG0, 10(R5)	; EXP CHECK?
10040	041462	001401			BEQ	66\$	
10041	041464	104051			ERROR	51	; NOT EQUAL, SIGNAL ERROR
10042	041466			66\$:			
10043							
10044	041466	000207			RTS	PC	; RETURN TO TEST CALLER

M16

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 209
FPP UNEXPECTED TRAP CATCHER

10045			
10046			
10047	041470	010637	002012
10048	041474	012637	002006
10049	041500	012637	002010
10050	041504	170237	002000
10051	041510	170337	002002
10052	041514	104056	
10053	041516	013746	002010
10054	041522	013746	002006
10055	041526	000002	

.SBTTL FPP UNEXPECTED TRAP CATCHER

FPPILT:	MOV	SP, FPPOSP	:	SP AFTER TRAP
	MOV	(SP)+, FPPOPC	:	POP OLD PC FOR DISPLAY
	MOV	(SP)+, FPPOPS	:	POP OLD PS FOR DISPLAY
	STFPS	FPS	:	GET FPS
	STST	FEC	:	GET FEC/FEA
	ERROR	56	:	SIGNAL UNEXPECTED FPP TRAP
	MOV	FPPOPS, -(SP)	:	PUSH PS
	MOV	FPPOPC, -(SP)	:	PUSH PC
	RTI		:	CONTINUE, RECOVER AT LAST TRAP ONLY

10056
10057
10058
10059
10060
10061
10062
10063
10064
10065
10066
10067
10068
10069
10070
10071
10072
10073
10074
10075
10076
10077
10078
10079
10080
10081
10082
10083
10084
10085
10086
10087
10088
10089
10090
10091
10092
10093
10094
10095
10096
10097
10098
10099
10100
10101
10102
10103
10104
10105
10106
10107
10108
10109
10110
10111

041530
041530
041530 032777 040000 137410
041536 001114
041540 000416
041542 013746 000004
041546 012737 041566 000004
041554 005737 177060
041560 012637 000004
041564 000463
041566 022626
041570 012637 000004
041574 000423
041576 032777 000400 137342
041604 001404
041606 023737 001112 001102
041614 001465
041616 005737 001104
041622 001421
041624 023737 001122 001104
041632 101015
041634 032777 001000 137304
041642 001404
041644 013737 001114 001110
041652 000446
041654 005037 001104
041660 005037 001310
041664 000415
041666 032777 004000 137252
041674 001011
041676 005737 001332
041702 001406
041704 005237 001106
041710 023737 001310 001106
041716 002024
041720 012737 000001 001106
041726 013737 042004 001310
041734 005237 001102
041740 013737 001102 001330
041746 011637 001110

```
.SBTTL SCOPE HANDLER ROUTINE
;*****
;THIS ROUTINE CONTROLS THE LOOPING OF SUBTESTS. IT WILL INCREMENT
;AND LOAD THE TEST NUMBER(STSTNM) INTO THE DISPLAY REG.(DISPLAY<15:0>)
;THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
;SW14=1 LOOP ON TEST
;SW11=1 INHIBIT ITERATIONS
;SW09=1 LOOP ON ERROR
;SW08=1 LOOP ON TEST IN "SLPTST"
;CALL
;* SCOPE ;;SCOPE=IOT

$SCOPE:
64$:
1$: BIT #BIT14,2SWR ;:LOOP ON PRESENT TEST?
BNE $OVER ;:YES IF SW14=1
;*****START OF CODE FOR THE XOR TESTER*****
$XTSTR: BR 6$ ;:IF RUNNING ON THE "XOR" TESTER CHANGE
;THIS INSTRUCTION TO A "NOP" (NOP=240)
MOV 2$ERRVEC, -(SP) ;:SAVE THE CONTENTS OF THE ERROR VECTOR
MOV 2$5,2$ERRVEC ;:SET FOR TIMEOUT
TST 2$177060 ;:TIME OUT ON XOR?
MOV (SP)+, 2$ERRVEC ;:RESTORE THE ERROR VECTOR
BR $SVLAD ;:GO TO THE NEXT TEST
5$: CMP (3P)+, (3P)+ ;:CLEAR THE STACK AFTER A TIME OUT
MOV (3P)+, 2$ERRVEC ;:RESTORE THE ERROR VECTOR
BR 7$ ;:LOOP ON THE PRESENT TEST
6$;*****END OF CODE FOR THE XOR TESTER*****
BIT #BIT08,2SWR ;:LOOP ON SPEC. TEST?
BEQ 2$ ;:BR IF NO
CMP SLPTST, STSTNM ;:ON THE RIGHT TEST?
BEQ $OVER ;:BR IF YES
2$: TST SERFLG ;:HAS AN ERROR OCCURRED?
BEQ 3$ ;:BR IF NO
CMP SERMAX, SERFLG ;:MAX. ERRORS FOR THIS TEST OCCURRED?
BHI 3$ ;:BR IF NO
BIT #BIT09,2SWR ;:LOOP ON ERROR?
BEQ 4$ ;:BR IF NO
7$: MOV $LPERR, $LPADR ;:SET LOOP ADDRESS TO LAST SCOPE
BR $OVER
4$: CLR SERFLG ;:ZERO THE ERROR FLAG
CLR $TIMES ;:CLEAR THE NUMBER OF ITERATIONS TO MAKE
BR 1$ ;:ESCAPE TO THE NEXT TEST
3$: BIT #BIT11,2SWR ;:INHIBIT ITERATIONS?
BNE 1$ ;:BR IF YES
TST $PASS ;:IF FIRST PASS OF PROGRAM
BEQ 1$ ;:INHIBIT ITERATIONS
INC $ICNT ;:INCREMENT ITERATION COUNT
CMP $TIMES, $ICNT ;:CHECK THE NUMBER OF ITERATIONS MADE
BGE $OVER ;:BR IF MORE ITERATION REQUIRED
1$: MOV #1, $ICNT ;:REINITIALIZE THE ITERATION COUNTER
MOV $MXCNT, $TIMES ;:SET NUMBER OF ITERATIONS TO DO
$SVLAD: INC STSTNM ;:COUNT TEST NUMBERS
MOV STSTNM, $TESTN ;:SET TEST NUMBER IN APT MAILBOX
MOV (SP), $LPADR ;:SAVE SCOPE LOOP ADDRESS
```

C01

FPU ADVANCED INSTR TESTS
D3FPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 211
SCOPE HANDLER ROUTINE

10112	041752	011637	001114		MOV	(SP), \$LPERR	:: SAVE ERROR LOOP ADDRESS
10113	041756	005037	001312		CLR	\$ESCAPE	:: CLEAR THE ESCAPE FROM ERROR ADDRESS
10114	041762	012737	000001	001122	MOV	#1, \$SERMAX	:: ONLY ALLOW ONE(1) ERROR ON NEXT TEST
10115	041770	013777	001102	137152	\$OVER: MOV	\$TSTNM, @DISPLAY	:: DISPLAY TEST NUMBER
10116	041776	013716	001110		MOV	\$LPADR, (SP)	:: FUDGE RETURN ADDRESS
10117	042002	000002			RTI		:: FIXES PS
10118	042004	003720			\$MXCNT: 2000.		:: MAX. NUMBER OF ITERATIONS

```

10119
10120
10121
10122
10123
10124
10125
10126
10127
10128
10129
10130
10131
10132
10133 042006
10134 042006 010037 002016
10135 042012 010137 002020
10136 042016 010237 002022
10137 042022 010337 002024
10138 042026 010437 002026
10139 042032 010537 002030
10140 042036 010637 002032
10141 042042 062737 000004 002032
10142 042050 011637 002034
10143 042054 005237 001104
10144 042060 001775
10145 042062 013777 001102 137060
10146 042070 032777 002000 137050
10147 042076 001402
10148 042100 104401 001314
10149 042104 005237 001116
10150 042110 011637 001124
10151 042114 162737 000002 001124
10152 042122 117737 136776 001120
10153 042130 032777 020000 137010
10154 042136 001004
10155 042140 004737 042250
10156 042144 104401 001321
10157 042150
10158 042150 122737 000001 001344
10159 042156 001007
10160 042160 113737 001120 042172
10161 042166 004737 042764
10162 042172 000
10163 042173 000
10164 042174 000777
10165 042176 005777 136744
10166 042202 100001
10167 042204 000000
10168 042206 032777 001000 136732
10169 042214 001402
10170 042216 013716 001114
10171 042222 005737 001312
10172 042226 001402
10173 042230 013716 001312
10174 042234

```

```

.SBTTL ERROR HANDLER ROUTINE

*****
*THIS ROUTINE WILL INCREMENT THE ERROR FLAG AND THE ERROR COUNT,
*SAVE THE ERROR ITEM NUMBER AND THE ADDRESS OF THE ERROR CALL
*AND GO TO STYPERR ON ERROR
*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
*SW15=1 HALT ON ERROR
*SW13=1 INHIBIT ERROR TYPEOUTS
*SW10=1 BELL ON ERROR
*SW09=1 LOOP ON ERROR
*CALL
* ERROR N ;;ERROR=EMT AND N=ERROR ITEM NUMBER

$ERROR:
MOV R0, EREG0 ; DISPLAY R0
MOV R1, EREG1 ; R1
MOV R2, EREG2 ; R2
MOV R3, EREG3 ; R3
MOV R4, EREG4 ; R4
MOV R5, EREG5 ; R5
MOV R6, EREG6 ; GET R6(SP) BEFORE TRAP
ADD #4, EREG6
MOV (SP), EREG7 ; PC -> ERROR CALL INSTR
INC $ERFLG ; SET THE ERROR FLAG
BEQ 7$ ; DON'T LET THE FLAG GO TO ZERO
MOV $STSTM, $DISPLAY ; DISPLAY TEST NUMBER
BIT #BIT10, $SWR ; BELL ON ERROR?
BEQ 1$ ; NO - SKIP
TYPE $BELL ; RING BELL
INC $ERTTL ; COUNT THE NUMBER OF ERRORS
MOV (SP), $ERRPC ; GET ADDRESS OF ERROR INSTRUCTION
SUB #2, $ERRPC
MOVB $ERRPC, $ITEMB ; STRIP AND SAVE THE ERROR ITEM CODE
BIT #BIT13, $SWR ; SKIP TYPEOUT IF SET
BNE 20$ ; SKIP TYPEOUTS
JSP PC, $STYPERR ; GO TO USER ERROR ROUTINE
TYPE $CRLF

20$:
CMPB #APTENV, $ENV ; RUNNING IN APT MODE
BNE 2$ ; NO SKIP APT ERROR REPORT
MOVB $ITEMB, 21$ ; SET ITEM NUMBER AS ERROR NUMBER
JSR PC, $ATY4 ; REPORT FATAL ERROR TO APT

21$:
.BYTE 0
.BYTE 0

22$:
BR 22$ ; APT ERROR LOOP
TST $SWR ; HALT ON ERROR
BPL 3$ ; SKIP IF CONTINUE
HALT ; HALT ON ERROR!

3$:
BIT #BIT09, $SWR ; LOOP ON ERROR SWITCH SET?
BEQ 4$ ; BR IF NO
MOV $LPERR, (SP) ; FUDGE RETURN FOR LOOPING
TST $ESCAPE ; CHECK FOR AN ESCAPE ADDRESS
BEQ 5$ ; BR IF NONE
MOV $ESCAPE, (SP) ; FUDGE RETURN ADDRESS FOR ESCAPE

4$:
5$:

```

E01

FPU ADVANCED INSTR TESTS
DQFP8A.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 213
ERROR HANDLER ROUTINE

10175	042234	022737	033062	000042	CMP	#SENDAD, J#42	:::ACT-11 AUTO-ACCEPT?
10176	042242	001001			BNE	6S	:::BRANCH IF NO
10177	042244	000000			HALT		:::YES
10178	042246			6S:			
10179	042246	000002		64S:	RTI		;RETURN

```

10180
10181
10182
10183
10184
10185
10186
10187
10188
10189
10190
10191
10192
10193
10194 042250
10195 042250 104401
10196 042252 001321
10197 042254 010046
10198 042256 010146
10199 042260 005000
10200 042262 153700 001120
10201 042266 001004
10202
10203 042270 013746 001124
10204 042274 104402
10205 042276 000452
10206 042300 005300
10207 042302 006300
10208 042304 010001
10209 042306 006300
10210 042310 060100
10211 042312 062700 001354
10212 042316 012037 042326
10213 042322 001404
10214 042324 104401
10215 042326 000000
10216 042330 104401 001321
10217 042334 104401 042444
10218 042340 012037 042350
10219 042344 001402
10220 042346 104401
10221 042350 000000
10222 042352 104401 001321
10223 042356 017746 000054
10224 042362 104402
10225 042364 104401 042442
10226 042370 017746 000044
10227 042374 104402
10228 042376 104401 042442
10229 042402 011000
10230 042404 001407
10231 042406 013046
10232 042410 104402
10233 042412 005710
10234 042414 001403
10235 042416 104401 042442
    
```

```

;*****
    
```

```

.SBTTL ERROR MESSAGE TYPEOUT ROUTINE (MODIFIED SYSMAC)
    
```

```

; *THIS ROUTINE USES THE "ITEM CONTROL BYTE" ($ITEMB) TO DETERMINE WHICH
; *ERROR IS TO BE REPORTED. IT THEN OBTAINS, FROM THE "ERROR TABLE",
; *($ERRTB) THE ERROR MESSAGE, DATA HEADER, AND DATA VALUES TO PRINT.
; *THIS ROUTINE IS IDENTICAL TO THE SYSMAC ROUTINE $ERRTYP, EXCEPT THIS
; *ROUTINE PUTS A <HT> BETWEEN OCTAL TYPED DATA VALUES, SO THAT EACH
; *VALUE STARTS AT A HORIZONTAL TAB STOP. ALSO, THE DATA FORMAT
; *POINTER HAS BEEN ELIMINATED FROM THE ERROR VECTOR. THIS ROUTINE
; *ALSO ALWAYS PRINTS $TESTN AND $ERRPC AS THE FIRST TWO DATA ELEMENTS
; *(WITH APPROPRIATE HEADERS).
    
```

```

$TYPERR:
    
```

```

HOTWRM: .WORD $SCRLF ; TYPE "HOT" OR "WARM"
        MOV RO,-(SP) ; PTR TO MESSAGE
        MOV R1,-(SP) ; SAVE R0
        CLR RO ; SAVE R1
        BISB @($ITEMB,RO) ; PICKUP ITEM INDEX
        BNE IS ; IF ITEM NUMBER FROM ERROR 0,
        ; JUST TYPE PC OF ERROR
        MOV $ERRPC,-(SP) ; GET ERROR PC FOR TYPEOUT
        TYPCC ; TYPE OCTAL, ALL DIGITS
        BR 7$ ; EXIT
IS: DEC RO ; ADJUST ERROR # FOR TABLE INDEX
    ASL RO ; OF 6 BYTES/ENTRY
    MOV RO,R1
    ASL RO
    ADD R1,RO
    ADD @($ERRTB,RO) ; FORM TABLE PTR
    MOV (RO)+,2$ ; PICKUP "ERROR MESSAGE" PTR
    BEQ 3$ ; SKIP TYPEOUT IF NULL
    TYPE "ERROR MESSAGE" ; TYPE "ERROR MESSAGE"
    .WORD 0 ; "ERROR MESSAGE" PTR HERE
    TYPE $SCRLF ; CR & LF
    TYPE 11$ ; "TEST # ERR PC" HEADER
    MOV (RO)+,4$ ; PICKUP "DATA HEADER" PTR
    BEQ 5$ ; SKIP TYPEOUT IF NULL
    TYPE "DATA HEADER" ; TYPE "DATA HEADER"
    .WORD 0 ; "DATA HEADER" PTR HERE
    TYPE $SCRLF ; CR & LF
    MOV @8$,-(SP) ; ($TESTN)
    TYPCC ; OCTAL W/ LEADING ZEROS
    TYPE 10$ ; <HT>
    MOV @9$,-(SP) ; ($ERRPC)
    TYPCC ; OCTAL W/ LEADING ZEROS
    TYPE 10$ ; <HT>
    MOV (RO),RO ; PICKUP "DATA TABLE" PTR
    BEQ 7$ ; EXIT IF NULL
    MOV @2(RO)+,-(SP) ; SAVE ... FOR TYPEOUT
    TYPCC ; TYPE OCTAL, ALL DIGITS
    TST (RO) ; ANOTHER NUMBER ?
    BEQ 7$ ; NO - EXIT
    TYPE ,10$ ; TAB BETWEEN ELEMENTS
    
```

GO1

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 215
ERROR MESSAGE TYPEOUT ROUTINE (MODIFIED SYSMAC)

10236	042422	000771		
10237	042424	012601		
10238	042426	012600		
10239	042430	104401	001321	
10240	042434	000207		
10241	042436	001330		
10242	042440	001124		
10243	042442	000011		
10244	042444	042524	052123	021440
10245	042452	042411	051122	050040
10246	042460	004503	000	
10247		042464		

```
7S: BR 6S ; LOOP ON DATA TABLE VECTOR
MOV (SP)+,R1 ; RESTORE R1
MOV (SP)+,R0 ; RESTORE R0
TYPE ,SCLF ; CR & LF
RTS PC ; RETURN
8S: .WORD $TESTN
9S: .WORD $ERRPC
10S: .ASCIZ <11> ; <HT>
11S: .ASCIZ *TEST # ERR PC
.EVEN
```

```

10248
10249
10250
10251
10252
10253
10254
10255
10256
10257
10258
10259
10260
10261
10262
10263
10264
10265 042464 105737 001165
10266 042470 100002
10267 042472 000000
10268 042474 000430
10269 042476 010046
10270 042500 017600 000002
10271 042504 122737 000001 001344
10272 042512 001011
10273 042514 132737 000100 001345
10274 042522 001405
10275 042524 010037 042534
10276 042530 004737 042754
10277 042534 000000
10278 042536 132737 000040 001345
10279 042544 001003
10280 042546 112046
10281 042550 001005
10282 042552 005726
10283 042554 012600
10284 042556 062716 000002
10285 042562 000002
10286 042564 122716 000011
10287 042570 001430
10288 042572 122716 000200
10289 042576 001006
10290 042600 005726
10291 042602 104401
10292 042604 001321
10293 042606 105037 042742
10294 042612 000755
10295 042614 004737 042676
10296 042620 123726 001164
10297 042624 001350
10298 042626 013746 001162
10299
10300 042632 105366 000001
10301 042636 002770
10302 042640 004737 042676
10303 042644 105337 042742

```

.SBTTL TYPE ROUTINE

```

*****
*ROUTINE TO TYPE ASCIZ MESSAGE. MESSAGE MUST TERMINATE WITH A 0 BYTE.
*THE ROUTINE WILL INSERT A NUMBER OF NULL CHARACTERS AFTER A LINE FEED.
*NOTE1: $NULL CONTAINS THE CHARACTER TO BE USED AS THE FILLER CHARACTER.
*NOTE2: $FILLS CONTAINS THE NUMBER OF FILLER CHARACTERS REQUIRED.
*NOTE3: $FILLC CONTAINS THE CHARACTER TO FILL AFTER.
*
*CALL:
*1) USING A TRAP INSTRUCTION
* TYPE ,MESADR ;;MESADR IS FIRST ADDRESS OF AN ASCIZ STRING
*OR
* TYPE
* MESADR
*
$TYPE: TSTB $TFPLG ;; IS THERE A TERMINAL?
      BPL 1$ ;; BR IF YES
      HALT ;; HALT HERE IF NO TERMINAL
      BR 3$ ;; LEAVE
1$: MOV RO,-(SP) ;; SAVE RO
   MOV 22(SP),RO ;; GET ADDRESS OF ASCIZ STRING
   CMPB #APTENV,$ENV ;; RUNNING IN APT MODE
   BNE 62$ ;; NO, GO CHECK FOR APT CONSOLE
   BITB #APTSPool,$ENVM ;; SPOOL MESSAGE TO APT
   BEQ 62$ ;; NO, GO CHECK FOR CONSOLE
   MOV RO,61$ ;; SETUP MESSAGE ADDRESS FOR APT
   JSR PC,$ATY3 ;; SPOOL MESSAGE TO APT
   .WORD 0 ;; MESSAGE ADDRESS
62$: BITB #APTCsup,$ENVM ;; APT CONSOLE SUPPRESSED
   BNE 60$ ;; YES, SKIP TYPE OUT
2$: MOV (RO)+,-(SP) ;; PUSH CHARACTER TO BE TYPED ONTO STACK
   BNE 4$ ;; BR IF IT ISN'T THE TERMINATOR
   TST (SP)+ ;; IF TERMINATOR POP IT OFF THE STACK
60$: MOV (SP)+,RO ;; RESTORE RO
3$: ADD #2,(SP) ;; ADJUST RETURN PC
   RTI ;; RETURN
4$: CMPB #HT,(SP) ;; BRANCH IF <HT>
   BEQ 8$ ;;
   CMPB #CRLF,(SP) ;; BRANCH IF NOT <CRLF>
   BNE 5$ ;;
   TST (SP)+ ;; POP <CR><LF> EQUIV
   TYPE ;; TYPE A CR AND LF
   $CRLF
   CLRB $CHARCNT ;; CLEAR CHARACTER COUNT
   BR 2$ ;; GET NEXT CHARACTER
5$: JSR PC,$TYPEC ;; GO TYPE THIS CHARACTER
6$: CMPB $FILLC,(SP)+ ;; IS IT TIME FOR FILLER CHARS.?
   BNE 2$ ;; IF NO GO GET NEXT CHAR.
   MOV $NULL,-(SP) ;; GET # OF FILLER CHARS. NEEDED
   ;; AND THE NULL CHAR.
7$: DECB 1(SP) ;; DOES A NULL NEED TO BE TYPED?
   BLT 6$ ;; BR IF NO--GO POP THE NULL OFF OF STACK
   JSR PC,$TYPEC ;; GO TYPE A NULL
   DECB $CHARCNT ;; DO NOT COUNT AS A COUNT

```



```

10304 042650 000770          BR      75          ;;LOOP
10305
10306          ;HORIZONTAL TAB PROCESSOR
10307
10308 042652 112716 000040    8$:   MOVB   #' (SP)          ;; REPLACE TAB WITH SPACE
10309 042656 004737 042676    9$:   JSR    PC,$TYPEC          ;; TYPE A SPACE
10310 042662 132737 000007 042742  BITB   #',$CHARCNT          ;; BRANCH IF NOT AT
10311 042670 001372          BNE    9$          ;; TAB STOP
10312 042672 005726          TST    (SP)+          ;; POP SPACE OFF STACK
10313 042674 000724          BR     2$          ;; GET NEXT CHARACTER
10314 042676 105777 136254    $TYPEC: TSTB  2$STPS          ;; WAIT UNTIL PRINTER IS READY
10315 042702 100375          BPL    $TYPEC
10316 042704 116677 000002 136246  MOVB   2(SP),2$TPB          ;; LOAD CHAR TO BE TYPED INTO DATA REG.
10317 042712 122766 000015 000002  CMPB   #CR,2(SP)          ;; IS CHARACTER A CARRIAGE RETURN?
10318 042720 001003          BNE    1$          ;; BRANCH IF NO
10319 042722 105037 042742          CLRB   $CHARCNT          ;; YES--CLEAR CHARACTER COUNT
10320 042726 000406          BR     $TYPEX          ;; EXIT
10321 042730 122766 000012 000002  1$:   CMPB   #LF,2(SP)          ;; IS CHARACTER A LINE FEED?
10322 042736 001402          BEQ    $TYPEX          ;; BRANCH IF YES
10323 042740 105227          INCB   (PC)+          ;; COUNT THE CHARACTER
10324 042742 000000          $CHARCNT: .WORD 0          ;; CHARACTER COUNT STORAGE
10325 042744 000207          $TYPEX: RTS   PC
10326
  
```

.SBTTL APT COMMUNICATIONS ROUTINE

```

10327
10328
10329
10330 042746 112737 000001 043212 $ATY1: MOVB #1, $FFLG ;; TO REPORT FATAL ERROR
10331 042754 112737 000001 043210 $ATY3: MOVB #1, $MFLG ;; TO TYPE A MESSAGE
10332 042762 000403 BR $ATYC
10333 042764 112737 000001 043212 $ATY4: MOVB #1, $FFLG ;; TO ONLY REPORT FATAL ERROR
10334 042772 $ATYC:
10335 042772 010046 MOV RO, -(SP) ;; PUSH RO ON STACK
10336 042774 010146 MOV RI, -(SP) ;; PUSH RI ON STACK
10337 042776 105737 043210 TSTB $MFLG ;; SHOULD TYPE A MESSAGE?
10338 043002 001450 BEQ $S IF NOT: BR
10339 043004 122737 000001 001344 CMPB $APTENV, $ENV ;; OPERATING UNDER APT?
10340 043012 001031 BNE $S IF NOT: BR
10341 043014 132737 000100 001345 BITB $APTPOOL, $ENVM ;; SHOULD SPOOL MESSAGES?
10342 043022 001425 BEQ $S IF NOT: BR
10343 043024 017600 000004 MOV #4(SP), RO ;; GET MESSAGE ADDR.
10344 043030 062766 000002 000004 ADD #2, 4(SP) ;; BUMP RETURN ADDR.
10345 043036 005737 001324 1$: TST $MSGTYPE ;; SEE IF DONE W/ LAST XMISSION?
10346 043042 001375 BNE $S IF NOT: WAIT
10347 043044 010037 001340 MOV RO, $MSGAD ;; PUT ADDR IN MAILBOX
10348 043050 105720 2$: TSTB (RO)+ ;; FIND END OF MESSAGE
10349 043052 001376 BNE $S
10350 043054 163700 001340 SUB $MSGAD, RO ;; SUB START OF MESSAGE
10351 043060 006200 ASR RO ;; GET MESSAGE LNTH IN WORDS
10352 043062 010037 001342 MOV RO, $MSG LGT ;; PUT LENGTH IN MAILBOX
10353 043066 012737 000004 001324 MOV #4, $MSGTYPE ;; TELL APT TO TAKE MSG.
10354 043074 000413 BR $S
10355 043076 017637 000004 043122 3$: MOV #4(SP), 4$ ;; PUT MSG ADDR IN JSR LINKAGE
10356 043104 062766 000002 000004 ADD #2, 4(SP) ;; BUMP RETURN ADDRESS
10357 043112 013746 177776 MOV 177776, -(SP) ;; PUSH 177776 ON STACK
10358 043116 004737 042464 JSR PC, $TYPE ;; CALL TYPE MACRO
10359 043122 000000 4$: .WORD 0
10360 043124 5$:
10361 043124 105737 043212 10$: TSTB $FFLG ;; SHOULD REPORT FATAL ERROR?
10362 043130 001416 BEQ $S IF NOT: BR
10363 043132 005737 001344 TST $ENV ;; RUNNING UNDER APT?
10364 043136 001413 BEQ $S IF NOT: BR
10365 043140 005737 001324 11$: TST $MSGTYPE ;; FINISHED LAST MESSAGE?
10366 043144 001375 BNE $S IF NOT: WAIT
10367 043146 017637 000004 001326 MOV #4(SP), $FATAL ;; GET ERROR #
10368 043154 062766 000002 000004 ADD #2, 4(SP) ;; BUMP RETURN ADDR.
10369 043162 005237 001324 INC $MSGTYPE ;; TELL APT TO TAKE ERROR
10370 043166 105037 043212 12$: CLRB $FFLG ;; CLEAR FATAL FLAG
10371 043172 105037 043211 CLRB $LFLG ;; CLEAR LOG FLAG
10372 043176 105037 043210 CLRB $MFLG ;; CLEAR MESSAGE FLAG
10373 043202 012601 MOV (SP)+, R1 ;; POP STACK INTO R1
10374 043204 012600 MOV (SP)+, R0 ;; POP STACK INTO R0
10375 043206 000207 RTS PC ;; RETURN
10376 043210 000 $MFLG: .BYTE 0 ;; MESSG. FLAG
10377 043211 000 $LFLG: .BYTE 0 ;; LOG FLAG
10378 043212 000 $FFLG: .BYTE 0 ;; FATAL FLAG
10379 043214 .EVEN
10380 000200 APTSIZE=200
10381 000001 APTENV=001
10382 000100 APTPOOL=100

```

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

10383

000040

MACY11 27(1006) 09-FEB-77 10:05 PAGE 219
APT COMMUNICATIONS ROUTINE

K01

APTCSUP=040

.SBTTL BINARY TO OCTAL (ASCII) AND TYPE

10384
10385
10386
10387
10388
10389
10390
10391
10392
10393
10394
10395
10396
10397
10398
10399
10400
10401
10402
10403
10404
10405
10406
10407
10408
10409
10410
10411
10412
10413
10414
10415
10416
10417
10418
10419
10420
10421
10422
10423
10424
10425
10426
10427
10428
10429
10430
10431
10432
10433
10434
10435
10436
10437
10438
10439

043214 017646 000000
043220 116637 000001
043226 112637 043441
043232 062716 000002
043236 000406
043240 112737 000001
043246 112737 000006
043254 112737 000005
043262 010346
043264 010446
043266 010546
043270 113704 043441
043274 005404
043276 062704 000006
043302 110437 043440
043306 113704 043437
043312 016605 000012
043316 005003
043320 006105
043322 000404
043324 006105
043326 006105
043330 006105
043332 010503
043334 006103
043336 105337 043440
043342 100016
043344 042703 177770
043350 001002
043352 005704
043354 001403

043437

043437

043441

043436

043441

```

*****
*THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 6-DIGIT
*OCTAL (ASCII) NUMBER AND TYPE IT.
*$TYPOS---ENTER HERE TO SETUP SUPPRESS ZEROS AND NUMBER OF DIGITS TO TYPE
*CALL:
*      MOV      NUM,-(SP)      ;;NUMBER TO BE TYPED
*      TYPOS    ;;CALL FOR TYPEOUT
*      .BYTE   N              ;;N=1 TO 6 FOR NUMBER OF DIGITS TO TYPE
*      .BYTE   M              ;;M=1 OR 0
*                               ;;1=TYPE LEADING ZEROS
*                               ;;0=SUPPRESS LEADING ZEROS
*$STYPON----ENTER HERE TO TYPE OUT WITH THE SAME PARAMETERS AS THE LAST
*$TYPOS OR $TYPOC
*CALL:
*      MOV      NUM,-(SP)      ;;NUMBER TO BE TYPED
*      TYPON    ;;CALL FOR TYPEOUT
*$STYPOC---ENTER HERE FOR TYPEOUT OF A 16 BIT NUMBER
*CALL:
*      MOV      NUM,-(SP)      ;;NUMBER TO BE TYPED
*      TYPOC    ;;CALL FOR TYPEOUT
*$TYPOS: MOV      2(SP),-(SP)    ;; PICKUP THE MODE
        MOV      1(SP),SOFILL    ;; LOAD ZERO FILL SWITCH
        MOV      (SP)+,SOMODE+1  ;; NUMBER OF DIGITS TO TYPE
        ADD      #2,(SP)        ;; ADJUST RETURN ADDRESS
        BR       $TYPON
*$TYPOC: MOV      #1,SOFILL      ;; SET THE ZERO FILL SWITCH
        MOV      #6,SOMODE+1     ;; SET FOR SIX(6) DIGITS
*$TYPON: MOV      #5,SOCNT      ;; SET THE ITERATION COUNT
        MOV      R3,-(SP)        ;; SAVE R3
        MOV      R4,-(SP)        ;; SAVE R4
        MOV      R5,-(SP)        ;; SAVE R5
        MOV      SOMODE+1,R4     ;; GET THE NUMBER OF DIGITS TO TYPE
        NEG      R4
        ADD      #6,R4          ;; SUBTRACT IT FOR MAX. ALLOWED
        MOV      R4,SOMODE      ;; SAVE IT FOR USE
        MOV      SOFILL,R4      ;; GET THE ZERO FILL SWITCH
        MOV      12(SP),R5      ;; PICKUP THE INPUT NUMBER
        CLR      R3            ;; CLEAR THE OUTPUT WORD
        ROL     R5             ;; ROTATE MSB INTO "C"
        BR      3$            ;; GO DO MSB
        ROL     R5             ;; FORM THIS DIGIT
        ROL     R5
        ROL     R5
        MOV     R5,R3
        ROL     R3            ;; GET LSB OF THIS DIGIT
        DECB   SOMODE         ;; TYPE THIS DIGIT?
        BPL    7$            ;; BR IF NO
        BIC    #177770,R3     ;; GET RID OF JUNK
        BNE   4$            ;; TEST FOR 0
        TST   R4             ;; SUPPRESS THIS 0?
        BEQ   5$            ;; BR IF YES

```

TRAP DECODER

.SBTTL TRAP DECODER

; THIS ROUTINE WILL PICKUP THE LOWER BYTE OF THE "TRAP" INSTRUCTION
; AND USE IT TO INDEX THROUGH THE TRAP TABLE FOR THE STARTING ADDRESS
; OF THE DESIRED ROUTINE. THEN USING THE ADDRESS OBTAINED IT WILL
; GO TO THAT ROUTINE.

10461
10462
10463
10464
10465
10466
10467
10468
10469 043442 010046
10470 043444 016600 000002
10471 043450 005740
10472 043452 111000
10473 043454 006300
10474 043456 016000 043476
10475 04346 000200
10476
10477
10478
10479
10480 043464 011646
10481 043466 016666 000004 000002
10482 043474 000002
10483
10484
10485
10486
10487
10488
10489
10490
10491 043476 043464
10492 043500 042464
10493 043502 043240
10494 043504 043214
10495 043506 043254
10496
10497

```
$TRAP:  MOV  RO, -(SP)          ;; SAVE RO
        MOV  2(SP), RO        ;; GET TRAP ADDRESS
        TST  -(RO)           ;; BACKUP BY 2
        MOVB (RO), RO        ;; GET RIGHT BYTE OF TRAP
        RSL  RO              ;; POSITION FOR INDEXING
        MOV  $TRPAD(RO), RO  ;; INDEX TO TABLE
        RTS  RO              ;; GO TO ROUTINE
```

;; THIS IS USE TO HANDLE THE "GETPRI" MACRO

```
$TRAP2: MOV  (SP), -(SP)      ;; MOVE THE PC DOWN
        MOV  4(SP), 2(SP)    ;; MOVE THE PSW DOWN
        FTI                    ;; RESTORE THE PSW
```

.SBTTL TRAP TABLE

; THIS TABLE CONTAINS THE STARTING ADDRESSES OF THE ROUTINES CALLED
; BY THE "TRAP" INSTRUCTION.

```
ROUTINE
-----
$TRPAD:  .WORD  $TRAP2
        $TYPE  ;; CALL=TYPE      TRAP+1(104401)  TTY TYPEOUT ROUTINE
        $TYPOC ;; CALL=TYPOC    TRAP+2(104402)  TYPE OCTAL NUMBER (WITH LEADING ZEROS)
        $TYPOS ;; CALL=TYPOS   TRAP+3(104403)  TYPE OCTAL NUMBER (NO LEADING ZEROS)
        $TYPON ;; CALL=TYPON   TRAP+4(104404)  TYPE OCTAL NUMBER (AS PER LAST CALL)
```

.SBTTL POWER DOWN AND UP ROUTINES

```

10498
10499
10500
10501
10502 043510 012737 043662 000024
10503 043516 012737 000340 000026
10504 043524 010046
10505 043526 010146
10506 043530 010246
10507 043532 010346
10508 043534 010446
10509 043536 010546
10510 043540 017746 135402
10511 043544 010637 043666
10512 043550 012737 043562 000024
10513 043556 000000
10514 043560 000776
10515
10516
10517
10518 043562 012737 043662 000024
10519 043570 013706 043666
10520 043574 005037 043666
10521 043600 005237 043666
10522 043604 001375
10523 043606 011600
10524 043610 076600 000226
10525 043614 012677 135326
10526 043620 012605
10527 043622 012604
10528 043624 012603
10529 043626 012602
10530 043630 012601
10531 043632 012600
10532 043634 012737 043510 000024
10533 043642 012737 000340 000026
10534 043650 104401
10535 043652 043670 SPWRMG: .WORD $POWER
10536 043654 012716 MOV (PC)+, (SP)
10537 043656 002162 SPWRAD: .WORD START
10538 043660 000002 RTI
10539 043662 000000 $ILLUP: HALT
10540 043664 000776 BR .-2
10541 043666 000000 $$SAVR6: 0
10542 043670 005015 $POWER: .ASCIZ <15><12>"POWER"
10543 043676 000122
10544 .EVEN

:*****
:POWER DOWN ROUTINE
$PWRDN: MOV $ILLUP, 2#$PWRVEC ;; SET FOR FAST UP
MOV #340, 2#$PWRVEC+2 ;; PRIO:7
MOV RO, -(SP) ;; PUSH RO ON STACK
MOV R1, -(SP) ;; PUSH R1 ON STACK
MOV R2, -(SP) ;; PUSH R2 ON STACK
MOV R3, -(SP) ;; PUSH R3 ON STACK
MOV R4, -(SP) ;; PUSH R4 ON STACK
MOV R5, -(SP) ;; PUSH R5 ON STACK
MOV @SWR, -(SP) ;; PUSH @SWR ON STACK
MOV SP, $$SAVR6 ;; SAVE SP
MOV $PWRUP, 2#$PWRVEC ;; SET UP VECTOR
HALT
BR .-2 ;; HANG UP

:*****
:POWER UP ROUTINE
$PWRUP: MOV $ILLUP, 2#$PWRVEC ;; SET FOR FAST DOWN
MOV $$SAVR6, SP ;; GET SP
CLR $SAVR6 ;; WAIT LOOP FOR THE TTY
1$: INC $SAVR6 ;; WAIT FOR THE INC
BNE 1$ OF WORD
MOV (SP), RO ;; GET SAVED SWR OFF STACK
MED #226, RO ;; RESTORE SWR CONTENTS
MOV (SP)+, @SWR ;; POP STACK INTO @SWR
MOV (SP)+, R5 ;; POP STACK INTO R5
MOV (SP)+, R4 ;; POP STACK INTO R4
MOV (SP)+, R3 ;; POP STACK INTO R3
MOV (SP)+, R2 ;; POP STACK INTO R2
MOV (SP)+, R1 ;; POP STACK INTO R1
MOV (SP)+, RO ;; POP STACK INTO RO
MOV $PWRDN, 2#$PWRVEC ;; SET UP THE POWER DOWN VECTOR
MOV #340, 2#$PWRVEC+2 ;; PRIO:7
TYPE REPORT THE POWER FAILURE
$POWER POWER FAIL MESSAGE POINTER
(PC)+, (SP) RESTART AT START
START RESTART ADDRESS

: THE POWER UP SEQUENCE WAS STARTED
: BEFORE THE POWER DOWN WAS COMPLETE
: PUT THE SP HERE

```

FPU ADVANCED INSTR TESTS
D0FPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 224
ERROR MESSAGES, DATA HEADERS, DATA VECTORS, ETC

10545				
10546				
10547				
10548	043700	047510	035124	000040
10549	043706	040527	046522	020072
10550	043714	000		
10551				
10552				
10553	043715	122	041505	044505
10554	043722	042526	020104	050106
10555	043730	020123	051511	041040
10556	043736	042101	000	
10557	043741	122	041505	044505
10558	043746	042526	020104	042506
10559	043754	027503	042506	020101
10560	043762	051511	041040	042101
10561	043770	000		
10562	043771	125	042516	050130
10563	043776	041505	042524	020104
10564	044004	046106	040517	044524
10565	044012	043516	050040	044517
10566	044020	052116	052040	040522
10567	044026	026120	044440	047107
10568	044034	051117	042105	023040
10569	044042	041440	047117	044524
10570	044050	052516	047111	000107
10571	044056	050103	020125	051520
10572	044064	041440	047117	044504
10573	044072	044524	047117	041440
10574	044100	042117	051505	051440
10575	044106	052105	044440	041516
10576	044114	051117	042522	052103
10577	044122	054514	000	
10578	044125	103	050115	043050
10579	044132	042057	020051	050117
10580	044140	051105	052101	047511
10581	044146	020116	020055	042522
10582	044154	044507	052123	051105
10583	044162	046440	042117	043111
10584	044170	042511	020104	043101
10585	044176	042524	027122	054105
10586	044204	041505	052135	047511
10587	044212	000116		
10588	044214	042101	027504	052523
10589	044222	024102	027506	024504
10590	044230	047440	042520	040522
10591	044236	044524	047117	026440
10592	044244	051040	051505	046125
10593	044252	020124	047111	047503
10594	044260	051122	041505	000124
10595	044266	052515	027514	044504
10596	044274	024126	027506	024504
10597	044302	047440	042520	040522
10598	044310	044524	047117	026440
10599	044316	051040	051505	046125
10600	044324	020124	047111	047503

.SBTTL ERROR MESSAGES, DATA HEADERS, DATA VECTORS, ETC

: MESSAGE PREFIXES

ASCHOT: .ASCIZ "HOT: "

ASCWARM: .ASCIZ "WARM: "

: ERROR MESSAGES HERE

EMA: .ASCIZ "RECEIVED FPS IS BAD"

EMB: .ASCIZ "RECEIVED FEC/FEA IS BAD"

EMC: .ASCIZ "UNEXPECTED FLOATING POINT TRAP, IGNORED & CONTINUING"

EMD: .ASCIZ "CPU PS CONDITION CODES SET INCORRECTLY"

EME: .ASCIZ "CMP(F/D) OPERATION - REGISTER MODIFIED AFTER EXECUTION"

EMF: .ASCIZ "ADD/SUB(F/D) OPERATION - RESULT INCORRECT"

EMG: .ASCIZ "MUL/DIV(F/D) OPERATION - RESULT INCORRECT"

FPU ADVANCED INSTR TESTS
 DAFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 225
 ERROR MESSAGES, DATA HEADERS, DATA VECTORS, ETC

10601	044332	051122	041505	000124	
10602	044340	047515	024104	027506	EMH: .ASCIZ "MOD(F/D) OPERATION - FRACTIONAL RESULT INCORRECT"
10603	044346	024504	047440	042520	
10604	044354	040522	044524	047117	
10605	044362	026440	043040	040522	
10606	044370	052103	047511	040516	
10607	044376	020114	042522	052523	
10608	044404	052114	044440	041516	
10609	044412	051117	042522	052103	
10610	044420	000			
10611	044421	115	042117	043050	EMI: .ASCIZ "MOD(F/D) OPERATION - INTEGER RESULT INCORRECT"
10612	044426	042057	020051	050117	
10613	044434	051105	052101	047511	
10614	044442	020116	020055	047111	
10615	044450	042524	042507	020122	
10616	044456	042522	052523	052114	
10617	044464	044440	041516	051117	
10618	044472	042522	052103	000	
10619	044477	106	047514	052101	EMJ: .ASCIZ "FLOAT-TO-DOUBLE CONVERSION - RESULT INCORRECT"
10620	044504	052055	026517	047504	
10621	044512	041125	042514	011440	
10622	044520	047117	042526	051522	
10623	044526	047511	020116	020055	
10624	044534	042522	052523	052114	
10625	044542	044440	041516	051117	
10626	044550	042522	052103	000	
10627	044555	104	052517	046102	EMK: .ASCIZ "DOUBLE-TO-FLOAT CONVERSION - RESULT INCORRECT"
10628	044562	026505	047524	043055	
10629	044570	047514	052101	041440	
10630	044576	047117	042526	051522	
10631	044604	047511	020116	020055	
10632	044612	042522	052523	052114	
10633	044620	044440	041516	051117	
10634	044626	042522	052103	000	
10635	044633	106	054111	042105	EML: .ASCIZ "FIXED-TO-FLOATING CONVERSION - RESULT INCORRECT"
10636	044640	052055	026517	046106	
10637	044646	040517	044524	043516	
10638	044654	041440	047117	042526	
10639	044662	051522	047511	020116	
10640	044670	020055	042522	052523	
10641	044676	052114	044440	041516	
10642	044704	051117	042522	052103	
10643	044712	000			
10644	044713	106	047514	052101	EMM: .ASCIZ "FLOATING-TO-FIXED CONVERSION - RESULT INCORRECT"
10645	044720	047111	026507	047524	
10646	044726	043055	054111	042105	
10647	044734	041440	047117	042526	
10648	044742	051522	047511	020116	
10649	044750	020055	042522	052523	
10650	044756	052114	044440	041516	
10651	044764	051117	042522	052103	
10652	044772	000			
10653	044773	114	040517	020104	EMN: .ASCIZ "LOAD EXPONENT(F/D) OPERATION - RESULT INCORRECT"
10654	045000	054105	047520	042516	
10655	045006	052116	043050	042057	
10656	045014	020051	050117	051105	

E02

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 226
ERROR MESSAGES, DATA HEADERS, DATA VECTORS, ETC

10657	045022	052101	047511	020116
10658	045030	020055	042522	052523
10659	045036	052114	044440	041516
10660	045044	051117	042522	052103
10661	045052	000		
10662	045053	123	047524	042522
10663	045060	042440	050130	047117
10664	045066	047105	024124	027506
10665	045074	024504	047440	042520
10666	045102	040522	044524	047117
10667	045110	026440	051040	051505
10668	045116	046125	020124	047111
10669	045124	047503	051122	041505
10670	045132	000124		

EMO: .ASCIZ "STORE EXPONENT(F/D) OPERATION - RESULT INCORRECT"

F02

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 227
ERROR MESSAGES, DATA HEADERS, DATA VECTORS, ETC

10671				
10672	045134	054105	023520	004504
10673	045142	041522	023526	000104
10674	045150	054105	023520	026504
10675	045156	042506	026503	041522
10676	045164	023526	004504	054105
10677	045172	023520	026504	042506
10678	045200	026501	041522	023526
10679	045206	000104		
10680	045210	026455	042455	050130
10681	045216	041505	042524	026504
10682	045224	026455	026411	026455
10683	045232	042522	042503	053111
10684	045240	042105	026455	000055
10685	045246	026455	026455	026455
10686	045254	026455	026455	042455
10687	045262	050130	041505	042524
10688	045270	026504	026455	026455
10689	045276	026455	026455	026455
10690	045304	026411	026455	026455
10691	045312	026455	026455	026455
10692	045320	042522	042503	053111
10693	045326	042105	026455	026455
10694	045334	026455	026455	026455
10695	045342	000055		
10696	045344	046117	020104	041520
10697	045352	047411	042114	050040
10698	045360	004523	042516	020127
10699	045366	050123	020011	050106
10700	045374	004523	043040	041505
10701	045402	020011	042506	000101

```

: DATA HEADERS HERE
DHA: .ASCIZ "EXP'D RCV'D"
DHB: .ASCIZ "EXP'D-FEC-RCV'D          EXP'D-FEA-RCV'D"
DHC: .ASCIZ "----EXPECTED---- ---RECEIVED----"
DHD: .ASCIZ "-----EXPECTED----- -----RECEIVED-----"
DHF: .ASCIZ "OLD PC OLD PS  NEW SP  FPS    FEC    FEA"

```

					; DATA ADDRESS VECTOR	
10702						.EVEN
10703						.WORD
10704	045410	001240	002000	000000	DTA:	.WORD STMP4,FPS,0
10705	045416	001242	002000	000000	DTB:	.WORD STMP5,FPS,0
10706	045424	001244	002000	000000	DTC:	.WORD STMP6,FPS,0
10707	045432	001246	002000	000000	DTD:	.WORD STMP7,FPS,0
10708	045440	001252	002000	000000	DTE:	.WORD STMP11,FPS,0
10709	045446	001254	002000	000000	DTF:	.WORD STMP12,FPS,0
10710	045454	001262	002000	000000	DTG:	.WORD STMP15,FPS,0
10711	045462	001272	002000	000000	DTH:	.WORD STMP21,FPS,0
10712	045470	001242	002002	002014	DTI:	.WORD STMP5,FEC,EXPFEA,FEA,0
10713	045476	002004	000000			
10714	045502	001244	002002	002014	DTJ:	.WORD STMP6,FEC,EXPFEA,FEA,0
10715	045510	002004	000000			
10716	045514	001246	002002	002014	DTK:	.WORD STMP7,FEC,EXPFEA,FEA,0
10717	045522	002004	000000			
10718	045526	001250	002002	002014	DTL:	.WORD STMP10,FEC,EXPFEA,FEA,0
10719	045534	002004	000000			
10720	045540	001254	002002	002014	DTM:	.WORD STMP12,FEC,EXPFEA,FEA,0
10721	045546	002004	000000			
10722	045552	001256	002002	002014	DTN:	.WORD STMP13,FEC,EXPFEA,FEA,0
10723	045560	002004	000000			
10724	045564	001264	002002	002014	DTO:	.WORD STMP16,FEC,EXPFEA,FEA,0
10725	045572	002004	000000			
10726	045576	001274	002002	002014	DTP:	.WORD STMP22,FEC,EXPFEA,FEA,0
10727	045604	002004	000000			
10728	045610	001234	001170	000000	DTQ:	.WORD STMP2,\$REG0,0
10729	045616	001240	001170	000000	DTR:	.WORD STMP4,\$REG0,0
10730	045624	001230	001232	001170	DTS:	.WORD STMP0,\$TMP1,\$REG0,\$REG1,0
10731	045632	001172	000000			
10732	045636	001232	001234	001170	DTT:	.WORD STMP1,\$TMP2,\$REG0,\$REG1,0
10733	045644	001172	000000			
10734	045650	001234	001236	001170	DTU:	.WORD STMP2,\$TMP3,\$REG0,\$REG1,0
10735	045656	001172	000000			
10736	045662	001240	001242	001170	DTV:	.WORD STMP4,\$TMP5,\$REG0,\$REG1,0
10737	045670	001172	000000			
10738	045674	001244	001246	001174	DTW:	.WORD STMP6,\$TMP7,\$REG2,\$REG3,0
10739	045702	001176	000000			
10740	045706	001230	001232	001234	DTX:	.WORD STMP0,\$TMP1,\$TMP2,\$TMP3
10741	045714	001236				
10742	045716	001170	001172	001174		.WORD \$REG0,\$REG1,\$REG2,\$REG3,0
10743	045724	001176	000000			
10744	045730	001232	001234	001236	DTY:	.WORD STMP1,\$TMP2,\$TMP3,\$TMP4
10745	045736	001240				
10746	045740	001170	001172	001174		.WORD \$REG0,\$REG1,\$REG2,\$REG3,0
10747	045746	001176	000000			
10748	045752	001234	001236	001240	DTZ:	.WORD STMP2,\$TMP3,\$TMP4,\$TMP5
10749	045760	001242				
10750	045762	001170	001172	001174		.WORD \$REG0,\$REG1,\$REG2,\$REG3,0
10751	045770	001176	000000			
10752	045774	001240	001242	001244	DTAA:	.WORD STMP4,\$TMP5,\$TMP6,\$TMP7
10753	046002	001246				
10754	046004	001170	001172	001174		.WORD \$REG0,\$REG1,\$REG2,\$REG3,0
10755	046012	001176	000000			
10756	046016	001250	001252	001254	DTAB:	.WORD STMP10,\$TMP11,\$TMP12,\$TMP13
10757	046024	001256				

H02

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 229
ERROR MESSAGES, DATA HEADERS, DATA VECTORS, ETC

10758	046026	001170	001172	001174		.WORD	\$REG0,\$REG1,\$REG2,\$REG3,0
10759	046034	001176	000000				
10760	046040	001260	001262	001264	DTAC:	.WORD	\$TMP14,\$TMP15,\$TMP16,\$TMP17
10761	046046	001266					
10762	046050	001200	001202	001204		.WORD	\$REG4,\$REG5,\$REG6,\$REG7,0
10763	046056	001206	000000				
10764	046062	001174	001172	000000	DTAD:	.WORD	\$REG2,\$REG1,0
10765	046070	001176	001174	000000	DTAE:	.WORD	\$REG3,\$REG2,0
10766	046076	002006	002010	002012	DTAK:	.WORD	FPPOPC,FPPOPS,FPPOSP,FPS,FEC,FEA,0
10767	046104	002000	002002	002004			
10768	046112	000000					
10769							
10770							
10771							
10772		000001					

; THE END
.END

DIVF3	013320	3852	3857#						
DIVF4	013356	3871	3876#						
DIVF5	013414	3890	3895#						
DIVF6	013452	3909	3914#						
DIVF7	013510	3928	3933#						
DSMR =	177570	789#	1008	1240					
DTA	045410	1113	10704#						
DTAA	045774	1145	1155	10752#					
DTAB	046016	1136	1138	1141	10756#				
DTAC	046040	1142	10760#						
DTAD	046062	1161	10764#						
DTAE	046070	1162	10765#						
DTAK	046076	1164	10766#						
DTB	045416	1114	10705#						
DTC	045424	1115	10706#						
DTD	045432	1116	10707#						
DTE	045440	1118	10708#						
DTF	045446	1119	10709#						
DTG	045454	1120	10710#						
DTH	045462	1121	10711#						
DTI	045470	1123	10712#						
DTJ	045502	1124	10714#						
DTK	045514	1125	10716#						
DTL	045526	1126	10718#						
DTM	045540	1128	10720#						
DTN	045552	1129	10722#						
DTO	045564	1130	10724#						
DTP	045576	1131	10726#						
DTQ	045610	1150	1156	10728#					
DTR	045616	1151	1157	10729#					
DTS	045624	1133	10730#						
DTT	045636	1146	10732#						
DTU	045650	1148	1152	1154	10734#				
DTV	045662	1135	1137	1139	1143	1153	10736#		
DTW	045674	1140	10738#						
DTX	045706	1134	10740#						
DTY	045730	1147	10744#						
DTZ	045752	1144	1149	10748#					
EMA	043715	1113	1114	1115	1116	1118	1119	1120	1121
EMB	043741	1123	1124	1125	1126	1128	1129	1130	1131
EMC	043771	1164	10562#						10553#
EMD	044056	1161	1162	10571#					10557#
EME	044125	1133	1134	10578#					
EMF	044214	1135	1136	10588#					
EMG	044266	1137	1138	10595#					
EMH	044340	1139	1141	10602#					
EMI	044421	1140	1142	10611#					
EMJ	044477	1144	1145	10619#					
EMK	044555	1143	10627#						
EML	044633	1146	1147	1148	1149	10635#			
EMM	044713	1150	1151	1152	1153	10644#			
EMN	044773	1154	1155	10653#					
EMO	045053	1156	1157	10662#					
EMTVEC=	000030	878#	1224#	1225#					
EMV001	001354	1113#							
EMV002	001362	1114#							

FPU ADVANCED INSTR TESTS
DOFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 236
CROSS REFERENCE TABLE -- USER SYMBOLS

EMV003	001370	1115#																			
EMV004	001376	1116#																			
EMV005	001404	1118#																			
EMV006	001412	1119#																			
EMV007	001420	1120#																			
EMV010	001426	1121#																			
EMV011	001434	1123#																			
EMV012	001442	1124#																			
EMV013	001450	1125#																			
EMV014	001456	1126#																			
EMV015	001464	1128#																			
EMV016	001472	1129#																			
EMV017	001500	1130#																			
EMV020	001506	1131#																			
EMV021	001514	1133#																			
EMV022	001522	1134#																			
EMV023	001530	1135#																			
EMV024	001536	1136#																			
EMV025	001544	1137#																			
EMV026	001552	1138#																			
EMV027	001560	1139#																			
EMV030	001566	1140#																			
EMV031	001574	1141#																			
EMV032	001602	1142#																			
EMV033	001610	1143#																			
EMV034	001616	1144#																			
EMV035	001624	1145#																			
EMV036	001632	1146#																			
EMV037	001640	1147#																			
EMV040	001646	1148#																			
EMV041	001654	1149#																			
EMV042	001662	1150#																			
EMV043	001670	1151#																			
EMV044	001676	1152#																			
EMV045	001704	1153#																			
EMV046	001712	1154#																			
EMV047	001720	1155#																			
EMV050	001726	1156#																			
EMV051	001734	1157#																			
EMV052	001742	1158#																			
EMV053	001750	1159#																			
EMV054	001756	1161#																			
EMV055	001764	1162#																			
EMV056	001772	1164#																			
EREG0	002016	1176#	10134*																		
EREG1	002020	1177#	10135*																		
EREG2	002022	1178#	10136*																		
EREG3	002024	1179#	10137*																		
EREG4	002026	1180#	10138*																		
EREG5	002030	1181#	10139*																		
EREG6	002032	1182#	10140*	10141*																	
EREG7	002034	1183#	10142*																		
ERRVEC=	000004	871#	1239#	1250#	10076	10077#	10079#	10082#													
EXPF EA	002014	1173#	8716#	8719	8760#	8763	8806#	8809	8849#	8852	8895*	8898	8938*	8941							
		8984#	8987	9027#	9030	9073#	9076	9116#	9119	9166*	9169	9222*	9225	9283#							
		9286	9339#	9342	9397#	9400	9441*	9444	9486#	9489	9702*	9705	9752*	9755							

F03

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 241
CROSS REFERENCE TABLE -- USER SYMBOLS

MD2F3	015566	4530	4535#																	
MD2F4	015630	4550	4555#																	
MD2F5	015672	4570	4575#																	
MD2F6	015734	4590	4595#																	
MD2F7	015776	4610	4615#																	
MED =	076600	885#	1267	1315	1319	1329	8632	8640	8646	10524										
MULD1	034536	9016#	9027																	
MULD1	034526	9010	9012#																	
MULD1	034502	3463	3485	3507	3529	3551	3573	3595	3617	3639	3661	3683	3705	3727						
		3749	3771	3793	9004#															
MULD1	011764	3462	3467#																	
MULD10	012432	3616	3621#																	
MULD11	012504	3638	3643#																	
MULD12	012556	3660	3665#																	
MULD13	012630	3682	3687#																	
MULD14	012702	3704	3709#																	
MULD15	012754	3726	3731#																	
MULD16	013026	3748	3753#																	
MULD17	013100	3770	3775#																	
MULD2	012036	3484	3489#																	
MULD20	013152	3792	3797#																	
MULD3	012110	3506	3511#																	
MULD4	012162	3528	3533#																	
MULD5	012234	3550	3555#																	
MULD6	012306	3572	3577#																	
MULD7	012360	3594	3599#																	
MULF1	034366	8973#	8984																	
MULFL	034356	8967	8969#																	
MULFT	034332	3158	3177	3196	3215	3234	3253	3272	3291	3310	3329	3348	3367	3386						
		3405	3424	3443	8961#															
MULF1	011024	3157	3162#																	
MULF10	011346	3290	3295#																	
MULF11	011404	3309	3314#																	
MULF12	011442	3328	3333#																	
MULF13	011500	3347	3352#																	
MULF14	011536	3366	3371#																	
MULF15	011574	3385	3390#																	
MULF16	011632	3404	3409#																	
MULF17	011670	3423	3428#																	
MULF2	011062	3176	3181#																	
MULF20	011726	3442	3447#																	
MULF3	011120	3195	3200#																	
MULF4	011156	3214	3219#																	
MULF5	011214	3233	3238#																	
MULF6	011252	3252	3257#																	
MULF7	011310	3271	3276#																	
MO =	100000	912#	1495	2093	2545	2780	2901	2903	3011	3033	3374	3449	3800	4636						
		4985	5341	5690	5982	6001	6058	6077	6134	6153	6173	6345	6364	6402						
		6461	6497	6516	6535															
M1 =	177777	911#	1186	1386	1387	1422	1423	1440	1512	1513	1531	1548	1587	1589						
		1627	1629	1669	1687	1689	1709	1769	1787	1789	1807	1808	1864	1865						
		1884	1903	1904	1921	1922	1923	1940	1941	1942	1959	1980	2016	2036						
		2054	2055	2056	2092	2094	2111	2112	2131	2169	2213	2235	2237	2257						
		2279	2283	2301	2303	2305	2323	2325	2327	2369	2393	2411	2413	2435						
		2437	2455	2457	2459	2499	2501	2503	2543	2547	2565	2567	2607	2685						
		2721	2722	2723	2740	2741	2742	2759	2761	2778	2797	2879	2901	2903						

2969	3009	3011	3013	3031	3033	3035	3097	3101	3119	3141	3164	3182
3183	3203	3222	3241	3353	3354	3391	3392	3393	3411	3429	3430	3449
3468	3490	3492	3516	3538	3558	3560	3580	3688	3690	3692	3710	3712
3732	3734	3736	3754	3756	3778	3800	3820	3934	3935	4029	4030	4031
4048	4049	4050	4106	4125	4144	4146	4364	4366	4368	4386	4388	4390
4452	4454	4456	4474	4476	4516	4517	4536	4537	4556	4557	4558	4559
4597	4598	4657	4698	4776	4778	4796	4841	4843	4865	4867	4889	4891
4893	4895	4939	4941	5011	5013	5037	5059	5061	5083	5153	5157	5177
5204	5221	5222	5224	5241	5242	5244	5261	5262	5263	5264	5284	5302
5303	5304	5324	5344	5362	5364	5384	5403	5404	5424	5444	5464	5481
5483	5484	5501	5504	5528	5546	5548	5552	5570	5572	5576	5594	5596
5598	5600	5624	5644	5646	5648	5672	5696	5716	5718	5720	5742	5744
5764	5766	5768	5788	5792	5816	5840	5858	5862	5864	5882	5888	5925
5927	5944	6020	6039	6041	6058	6077	6079	6096	6115	6134	6153	6174
6230	6231	6249	6268	6269	6287	6383	6421	6423	6440	6459	6478	6497
6516	6518	6535	6574	6593	6595	6650	6846	6863	6864	6915	7002	7003
7111	7165	7203	7241	7260	7279	7298	7300	7318	7354	7390	7408	7464
7466	7483	7502	7521	7540	7579	7580	7693	7694	7807	7808	7845	7940
7981	7983	8107	8109	8275	8381	8609						
910#	1549	2017	2037	2413	2437	3240	3259	3355	3558	3580	3692	3736
4658	4697	4717	5013	5059	5083	5363	5402	5422	5718	5764	5788	
928#	1353	1371	1389	1407	1425	1443	1461	1479	1515	1533	1551	1572
1592	1612	1632	1652	1692	1712	1732	1752	1772	1792	1811	1830	1849
1868	1887	1906	1925	1944	1963	1982	2001	2020	2039	2115	2134	2153
2176	2198	2220	2242	2264	2286	2308	2330	2352	2374	2396	2418	2440
2528	2550	2572	2592	2611	2630	2649	2668	2687	2706	2744	2763	2801
2839	2862	2884	2906	2928	2950	2972	2994	3038	3082	3104	3148	3167
3186	3205	3224	3243	3262	3281	3300	3319	3338	3414	3433	3452	3475
3497	3519	3541	3563	3585	3607	3629	3651	3673	3717	3761	3805	3824
3862	3881	3900	3919	3938	3957	3976	3995	4014	4090	4128	4151	4173
4195	4217	4239	4261	4283	4305	4327	4437	4481	4501	4521	4541	4581
4601	4621	4641	4661	4681	4701	4721	4761	4801	4826	4850	4874	4922
4946	4970	4994	5018	5042	5066	5090	5138	5186	5206	5226	5246	5286
5306	5326	5346	5366	5386	5406	5426	5466	5506	5531	5555	5579	5627
5651	5675	5699	5723	5747	5771	5795	5843	5891	5910	5948	5967	5986
6005	6043	6062	6081	6100	6119	6138	6157	6196	6215	6234	6253	6272
6291	6310	6330	6349	6368	6387	6406	6425	6444	6482	6501	6520	6539
7060	7078	7114	7132	7150	7168	7207	7226	7245	7264	7283	7302	7321
7357	7375	7393	7411	7429	7468	7487	7506	7525	7544	7563	7621	7640
7659	7678	7697	7716	7735	7830	7849	7868	7887	7906	7925	7944	8008
8029	8050	8071	8092	8113	8134	8239	8260	8281	8302	8323	8344	8365
1271	1273	1298#	8688									
1203#	1303											
788#												
882#												
1186#	9150	9206	9267	9323	9383	9427						
805#												
806#												
807#												
808#												
809#												
810#												
811#												
812#												
785#	786	9693	9743	9793	9845	9983	10023					
786#												

ME = 177776

MA = 000000

NEWPAS 002616
 NWPAS1 002151
 PIRQ = 177772
 PIRQVE = 000240
 PREVAC 002036
 PRO = 000000
 PR1 = 000040
 PR2 = 000100
 PR3 = 000140
 PR4 = 000200
 PR5 = 000240
 PR6 = 000300
 PR7 = 000340
 PS = 177776
 PSM = 177776

K03

FPU ADVANCED INSTR TESTS
DOPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 246
CROSS REFERENCE TABLE -- USER SYMBOLS

TST102	007162	2623	2638#
TST103	007220	2642	2657#
TST104	007256	2661	2676#
TST105	007314	2680	2695#
TST106	007352	2699	2714#
TST107	007410	2718	2733#
TST11	003300	1473	1487#
TST110	007446	2737	2752#
TST111	007504	2756	2771#
TST112	007542	2775	2790#
TST113	007600	2794	2809#
TST114	007636	2813	2828#
TST115	007674	2832	2848#
TST116	007746	2852	2870#
TST117	010020	2874	2892#
TST12	003332	1491	1505#
TST120	010072	2896	2914#
TST121	010144	2918	2936#
TST122	010216	2940	2958#
TST123	010270	2962	2980#
TST124	010342	2984	3002#
TST125	010414	3006	3024#
TST126	010466	3028	3046#
TST127	010540	3050	3068#
TST13	003364	1509	1523#
TST130	010612	3072	3090#
TST131	010664	3094	3112#
TST132	010736	3116	3134#
TST133	011010	3138	3156#
TST134	011046	3160	3175#
TST135	011104	3179	3194#
TST136	011142	3198	3213#
TST137	011200	3217	3232#
TST14	003416	1527	1541#
TST140	011236	3236	3251#
TST141	011274	3255	3270#
TST142	011332	3274	3289#
TST143	011370	3293	3308#
TST144	011426	3312	3327#
TST145	011464	3331	3346#
TST146	011522	3350	3365#
TST147	011560	3369	3384#
TST15	003450	1545	1560#
TST150	011616	3388	3403#
TST151	011654	3407	3422#
TST152	011712	3426	3441#
TST153	011750	3445	3461#
TST154	012022	3465	3483#
TST155	012074	3487	3505#
TST156	012146	3509	3527#
TST157	012220	3531	3549#
TST16	003512	1564	1580#
TST160	012272	3553	3571#
TST161	012344	3575	3593#
TST162	012416	3597	3615#
TST163	012470	3619	3637#

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 247
CROSS REFERENCE TABLE -- USER SYMBOLS

TST164	012542	3641	3659#
TST165	012614	3663	3681#
TST166	012666	3685	3703#
TST167	012740	3707	3725#
TST17	003554	1584	1600#
TST170	013012	3729	3747#
TST171	013064	3751	3769#
TST172	013136	3773	3791#
TST173	013210	3795	3813#
TST174	013246	3817	3832#
TST175	013304	3836	3851#
TST176	013342	3855	3870#
TST177	013400	3874	3889#
TST2	003012	1347	1361#
TST20	003616	1604	1620#
TST200	013436	3893	3908#
TST201	013474	3912	3927#
TST202	013532	3931	3946#
TST203	013570	3950	3965#
TST204	013626	3969	3984#
TST205	013664	3988	4003#
TST206	013722	4007	4022#
TST207	013760	4026	4041#
TST21	003660	1624	1640#
TST210	014016	4045	4060#
TST211	014054	4064	4079#
TST212	014112	4083	4098#
TST213	014150	4102	4117#
TST214	014206	4121	4137#
TST215	014260	4141	4159#
TST216	014332	4163	4181#
TST217	014404	4185	4203#
TST22	003722	1644	1660#
TST220	014456	4207	4225#
TST221	014530	4229	4247#
TST222	014602	4251	4269#
TST223	014654	4273	4291#
TST224	014726	4295	4313#
TST225	015000	4317	4335#
TST226	015052	4339	4357#
TST227	015124	4361	4379#
TST23	003764	1664	1680#
TST230	015176	4383	4401#
TST231	015250	4405	4423#
TST232	015322	4427	4445#
TST233	015374	4449	4467#
TST234	015446	4471	4489#
TST235	015510	4493	4509#
TST236	015552	4513	4529#
TST237	015614	4533	4549#
TST24	004026	1684	1700#
TST240	015656	4553	4569#
TST241	015720	4573	4589#
TST242	015762	4593	4609#
TST243	016024	4613	4629#
TST244	016066	4633	4649#

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 248
CROSS REFERENCE TABLE -- USER SYMBOLS

TST245	016130	4653	4669#
TST246	016172	4673	4689#
TST247	016234	4693	4709#
TST25	004070	1704	1720#
TST250	016276	4713	4729#
TST251	016340	4733	4749#
TST252	016402	4753	4769#
TST253	016444	4773	4789#
TST254	016506	4793	4810#
TST255	016570	4814	4834#
TST256	016652	4838	4858#
TST257	016734	4862	4882#
TST26	004132	1724	1740#
TST260	017016	4886	4906#
TST261	017100	4910	4930#
TST262	017162	4934	4954#
TST263	017244	4958	4978#
TST264	017326	4982	5002#
TST265	017410	5006	5026#
TST266	017472	5030	5050#
TST267	017554	5054	5074#
TST27	004174	1744	1760#
TST270	017636	5078	5098#
TST271	017720	5102	5122#
TST272	020002	5126	5146#
TST273	020064	5150	5170#
TST274	020146	5174	5194#
TST275	020210	5198	5214#
TST276	020252	5218	5234#
TST277	020314	5238	5254#
TST3	003044	1365	1379#
TST30	004236	1764	1780#
TST300	020356	5258	5274#
TST301	020420	5278	5294#
TST302	020462	5298	5314#
TST303	020524	5318	5334#
TST304	020566	5338	5354#
TST305	020630	5358	5374#
TST306	020672	5378	5394#
TST307	020734	5398	5414#
TST31	004300	1784	1800#
TST310	020776	5418	5434#
TST311	021040	5438	5454#
TST312	021102	5458	5474#
TST313	021144	5478	5494#
TST314	021206	5498	5515#
TST315	021270	5519	5539#
TST316	021352	5543	5563#
TST317	021434	5567	5587#
TST32	004336	1804	1819#
TST320	021516	5591	5611#
TST321	021600	5615	5635#
TST322	021662	5639	5659#
TST323	021744	5663	5683#
TST324	022026	5687	5707#
TST325	022110	5711	5731#

TST326	022172	5735	5755#
TST327	022254	5759	5779#
TST33	004374	1823	1838#
TST330	022336	5783	5803#
TST331	022420	5807	5827#
TST332	022502	5831	5851#
TST333	022564	5855	5875#
TST334	022646	5879	5899#
TST335	022704	5903	5918#
TST336	022742	5922	5937#
TST337	023000	5941	5956#
TST34	004432	1842	1857#
TST340	023036	5960	5975#
TST341	023074	5979	5994#
TST342	023132	5998	6013#
TST343	023170	6017	6032#
TST344	023226	6036	6051#
TST345	023264	6055	6070#
TST346	023322	6074	6089#
TST347	023360	6093	6108#
TST35	004470	1861	1876#
TST350	023416	6112	6127#
TST351	023454	6131	6146#
TST352	023512	6150	6166#
TST353	023550	6170	6185#
TST354	023606	6189	6204#
TST355	023644	6208	6223#
TST356	023702	6227	6242#
TST357	023740	6246	6261#
TST36	004526	1880	1895#
TST360	023776	6265	6280#
TST361	024034	6284	6299#
TST362	024072	6303	6319#
TST363	024130	6323	6338#
TST364	024166	6342	6357#
TST365	024224	6361	6376#
TST366	024262	6380	6395#
TST367	024320	6399	6414#
TST37	004564	1899	1914#
TST370	024356	6418	6433#
TST371	024414	6437	6452#
TST372	024452	6456	6471#
TST373	024510	6475	6490#
TST374	024546	6494	6509#
TST375	024604	6513	6528#
TST376	024642	6532	6548#
TST377	024702	6552	6567#
TST4	003076	1383	1397#
TST40	004622	1918	1933#
TST400	024742	6571	6586#
TST401	025002	6590	6605#
TST402	025042	6609	6624#
TST403	025102	6628	6643#
TST404	025142	6647	6662#
TST405	025170	6666	6679#
TST406	025216	6683	6696#

TST407	025244	6700	6713#
TST41	004660	1937	1952#
TST410	025272	6717	6730#
TST411	025320	6734	6748#
TST412	025352	6752	6766#
TST413	025404	6770	6784#
TST414	025436	6788	6802#
TST415	025470	6806	6820#
TST416	025522	6824	6839#
TST417	025552	6843	6856#
TST42	004716	1956	1971#
TST420	025602	6860	6873#
TST421	025632	6877	6890#
TST422	025662	6894	6907#
TST423	025712	6911	6924#
TST424	025742	6928	6941#
TST425	025772	6945	6959#
TST426	026026	6963	6977#
TST427	026062	6981	6995#
TST43	004754	1975	1990#
TST430	026116	6999	7013#
TST431	026152	7017	7031#
TST432	026206	7035	7050#
TST433	026236	7054	7068#
TST434	026266	7072	7086#
TST435	026316	7090	7104#
TST436	026346	7108	7122#
TST437	026376	7126	7140#
TST44	005012	1994	2009#
TST440	026426	7144	7158#
TST441	026456	7162	7177#
TST442	026512	7181	7196#
TST443	026546	7200	7215#
TST444	026602	7219	7234#
TST445	026636	7238	7253#
TST446	026672	7257	7272#
TST447	026726	7276	7291#
TST45	005050	2013	2028#
TST450	026762	7295	7311#
TST451	027014	7315	7329#
TST452	027046	7333	7347#
TST453	027100	7351	7365#
TST454	027132	7369	7383#
TST455	027164	7387	7401#
TST456	027216	7405	7419#
TST457	027250	7423	7438#
TST46	005106	2032	2047#
TST460	027306	7442	7457#
TST461	027344	7461	7476#
TST462	027402	7480	7495#
TST463	027440	7499	7514#
TST464	027476	7518	7533#
TST465	027534	7537	7552#
TST466	027572	7556	7572#
TST467	027626	7576	7591#
TST47	005144	2051	2066#

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 251
CROSS REFERENCE TABLE -- USER SYMBOLS

TST470	027662	7595	7610#
TST471	027716	7614	7629#
TST472	027752	7633	7648#
TST473	030006	7652	7667#
TST474	030042	7671	7686#
TST475	030076	7690	7705#
TST476	030132	7709	7724#
TST477	030166	7728	7743#
TST5	003130	1401	1415#
TST50	005202	2070	2085#
TST500	030222	7747	7762#
TST501	030256	7766	7781#
TST502	030312	7785	7800#
TST503	030346	7804	7819#
TST504	030402	7823	7838#
TST505	030436	7842	7857#
TST506	030472	7861	7876#
TST507	030526	7880	7895#
TST51	005240	2089	2104#
TST510	030562	7899	7914#
TST511	030616	7918	7933#
TST512	030652	7937	7953#
TST513	030716	7957	7974#
TST514	030762	7978	7995#
TST515	031026	7999	8016#
TST516	031072	8020	8037#
TST517	031136	8041	8058#
TST52	005276	2108	2123#
TST520	031202	8062	8079#
TST521	031246	8083	8100#
TST522	031312	8104	8121#
TST523	031356	8125	8142#
TST524	031422	8146	8163#
TST525	031466	8167	8184#
TST526	031532	8188	8205#
TST527	031576	8209	8226#
TST53	005334	2127	2142#
TST530	031642	8230	8247#
TST531	031706	8251	8268#
TST532	031752	8272	8289#
TST533	032016	8293	8310#
TST534	032062	8314	8331#
TST535	032126	8335	8352#
TST536	032172	8356	8374#
TST537	032220	8378	8391#
TST54	005372	2146	2162#
TST540	032246	8395	8408#
TST541	032274	8412	8425#
TST542	032322	8429	8442#
TST543	032350	8446	8459#
TST544	032376	8463	8476#
TST545	032424	8480	8494#
TST546	032456	8498	8512#
TST547	032510	8516	8530#
TST55	005444	2166	2184#
TST550	032542	8534	8548#

FPU ADVANCED INSTR TESTS
DFFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 252
CROSS REFERENCE TABLE -- USER SYMBOLS

TST551	032574	8552	8566#																		
TST552	032626	8570	8584#																		
TST553	032660	8588	8602#																		
TST554	032712	8606	8619#																		
TST56	005516	2188	2206#																		
TST57	005570	2210	2228#																		
TST6	003162	1419	1433#																		
TST60	005642	2232	2250#																		
TST61	005714	2254	2272#																		
TST62	005766	2276	2294#																		
TST63	006040	2298	2316#																		
TST64	006112	2320	2338#																		
TST65	006164	2342	2360#																		
TST66	006236	2364	2382#																		
TST67	006310	2386	2404#																		
TST7	003214	1437	1451#																		
TST70	006362	2408	2426#																		
TST71	006434	2430	2448#																		
TST72	006506	2452	2470#																		
TST73	006560	2474	2492#																		
TST74	006632	2496	2514#																		
TST75	006704	2518	2536#																		
TST76	006756	2540	2558#																		
TST77	007030	2562	2581#																		
TYPE =	104401	1262	1270	1272	1303	1308	8677	10148	10156	10195	10214	10216	10217	10220							
		10222	10225	10228	10235	10239	10291	10444	10492#	10534											
TYP0C =	104402	10204	10224	10227	10232	10493#															
TYP0N =	104404	10495#																			
TYP0S =	104403	1306	10494#																		
UB =	177400	923#	4678	5037	5383	5742															
WFLAG =	000344	890#	1329	8646																	
ZX1MN =	100177	918#	1531	1669	1769	2131	2457	2503	2722	2741	3392	3778	4030	4049							
		4366	4388	4537	4557	4867	4891	5242	5262	5572	5596	5925	6287								
ZX1MP =	000177	917#	1807	1808	2056	2169	2213	2607	2608	2626	2877	2879	3183	3490							
		3934	4029	4031	4048	4050	4144	4364	4368	4386	4390	4516	4841	5221							
		5546	5944	6249	6383	6574															
		966	972#																		
\$APTHD	001000	10361	10376																		
\$ASTAT=	***** U	10332	10334#																		
\$ATYC	042772	10330#																			
\$ATY1	042746	10276	10331#																		
\$ATY3	042754	10161	10333#																		
\$ATY4	042764	1005#																			
\$AUT08	001142	1000#																			
\$B0ADR	001130	1002#																			
\$B00AT	001134	1062#	8677	10148	10180																
\$BELL	001314	10293#	10303#	10310	10319*	10324#															
\$CHARC	042742	10498																			
\$CKSWR=	***** U	987#	1215	1216	1224	1230	1231	1232													
\$CHTAG	001100	1020#	1021#	1022#	1023#	1024#	1025#	1026#	1027#	1028#	1029#	1030#	1031#	1032#							
\$CM1 =	000020	1033#	1034#	1035#	1036#																
\$CM2 =	000040	1020#	1021#	1022#	1023#	1024#	1025#	1026#	1027#	1028#	1029#	1030#	1031#	1032#							
		1033#	1034#	1035#	1036#																
\$CM3 =	000020	1018#	1020																		
\$CM4 =	000030	1036#	1037#	1038#	1039#	1040#	1041#	1042#	1043#	1044#	1045#	1046#	1047#	1048#							
		1049#	1050#	1051#	1052#	1053#	1054#	1055#	1056#	1057#	1058#	1059#	1060#								

F04

FPU ADVANCED INSTR TESTS
 DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 254
 CROSS REFERENCE TABLE -- USER SYMBOLS

2355#	2357	2377#	2379	2399#	2401	2421#	2423	2443#	2445	2465#	2467	2487#
2489	2509#	2511	2531#	2533	2553#	2555	2576#	2578	2595#	2597	2614#	2616
2633#	2635	2652#	2654	2671#	2673	2690#	2692	2709#	2711	2728#	2730	2747#
2749	2766#	2768	2785#	2787	2804#	2806	2823#	2825	2843#	2845	2865#	2867
2887#	2889	2909#	2911	2931#	2933	2953#	2955	2975#	2977	2997#	2999	3019#
3021	3041#	3043	3063#	3065	3085#	3087	3107#	3109	3129#	3131	3151#	3153
3170#	3172	3189#	3191	3208#	3210	3227#	3229	3246#	3248	3265#	3267	3284#
3286	3303#	3305	3322#	3324	3341#	3343	3360#	3362	3379#	3381	3398#	3400
3417#	3419	3436#	3438	3456#	3458	3478#	3480	3500#	3502	3522#	3524	3544#
3546	3566#	3568	3588#	3590	3610#	3612	3632#	3634	3654#	3656	3676#	3678
3698#	3700	3720#	3722	3742#	3744	3764#	3766	3786#	3788	3808#	3810	3827#
3829	3846#	3848	3865#	3867	3884#	3886	3903#	3905	3922#	3924	3941#	3943
3960#	3962	3979#	3981	3998#	4000	4017#	4019	4036#	4038	4055#	4057	4074#
4076	4093#	4095	4112#	4114	4132#	4134	4154#	4156	4176#	4178	4198#	4200
4220#	4222	4242#	4244	4264#	4266	4286#	4288	4308#	4310	4330#	4332	4352#
4354	4374#	4376	4395#	4398	4418#	4420	4440#	4442	4462#	4464	4484#	4486
4504#	4506	4524#	4526	4544#	4546	4564#	4566	4584#	4586	4604#	4606	4624#
4626	4644#	4646	4664#	4666	4684#	4686	4704#	4706	4724#	4726	4744#	4746
4764#	4766	4784#	4786	4805#	4807	4829#	4831	4853#	4855	4877#	4879	4901#
4903	4925#	4927	4949#	4951	4973#	4975	4997#	4999	5021#	5023	5045#	5047
5069#	5071	5093#	5095	5117#	5119	5141#	5143	5165#	5167	5189#	5191	5209#
5211	5229#	5231	5249#	5251	5269#	5271	5289#	5291	5309#	5311	5329#	5331
5349#	5351	5369#	5371	5389#	5391	5409#	5411	5429#	5431	5449#	5451	5469#
5471	5489#	5491	5510#	5512	5534#	5536	5558#	5560	5582#	5584	5606#	5608
5630#	5632	5654#	5656	5678#	5680	5702#	5704	5726#	5728	5750#	5752	5774#
5776	5798#	5800	5822#	5824	5846#	5848	5870#	5872	5894#	5896	5913#	5915
5932#	5934	5951#	5953	5970#	5972	5989#	5991	6008#	6010	6027#	6029	6046#
6048	6065#	6067	6084#	6086	6103#	6105	6122#	6124	6141#	6143	6161#	6163
6180#	6182	6199#	6201	6218#	6220	6237#	6239	6256#	6258	6275#	6277	6294#
6296	6314#	6316	6333#	6335	6352#	6354	6371#	6373	6390#	6392	6409#	6411
6428#	6430	6447#	6449	6466#	6468	6485#	6487	6504#	6506	6523#	6525	6543#
6545	6562#	6564	6581#	6583	6600#	6602	6619#	6621	6638#	6640	6657#	6659
6674#	6676	6691#	6693	6708#	6710	6725#	6727	6743#	6745	6761#	6763	6779#
6781	6797#	6799	6815#	6817	6834#	6836	6851#	6853	6868#	6870	6885#	6887
6902#	6904	6919#	6921	6936#	6938	6954#	6956	6972#	6974	6990#	6992	7008#
7010	7026#	7028	7045#	7047	7063#	7065	7081#	7083	7099#	7101	7117#	7119
7135#	7137	7153#	7155	7172#	7174	7191#	7193	7210#	7212	7229#	7231	7248#
7250	7267#	7269	7286#	7288	7306#	7308	7324#	7326	7342#	7344	7360#	7362
7378#	7380	7396#	7398	7414#	7416	7433#	7435	7452#	7454	7471#	7473	7490#
7492	7509#	7511	7528#	7530	7547#	7549	7567#	7569	7586#	7588	7605#	7607
7624#	7626	7643#	7645	7662#	7664	7681#	7683	7700#	7702	7719#	7721	7738#
7740	7757#	7759	7776#	7778	7795#	7797	7814#	7816	7833#	7835	7852#	7854
7871#	7873	7890#	7892	7909#	7911	7928#	7930	7948#	7950	7969#	7971	7990#
7992	8011#	8013	8032#	8034	8053#	8055	8074#	8076	8095#	8097	8116#	8118
8137#	8139	8158#	8160	8179#	8181	8200#	8202	8221#	8223	8242#	8244	8263#
8265	8284#	8286	8305#	8307	8326#	8328	8347#	8349	8369#	8371	8386#	8388
8403#	8405	8420#	8422	8437#	8439	8454#	8456	8471#	8473	8489#	8491	8507#
8509	8525#	8527	8543#	8545	8561#	8563	8579#	8581	8597#	8599		

SOCNT 043436
 SOMODE 043440
 SOVER 041770
 SPASS 001332
 SPASTM 001006
 SPOWER 043670
 SPWRAD 043656
 SPWRON 043510

10416# 10445# 10458#
 10411# 10415# 10420 10423# 10434# 10460#
 10072 10088 10096 10106 10115#
 1075# 1252# 1304 8667# 8668# 10102 10119
 976#
 10535 10542#
 10537#
 1228 10502# 10532

		3792	3814	3833	3852	3871	3890	3909	3928	3947	3966	3985	4004	4023
		4042	4061	4080	4099	4118	4138	4160	4182	4204	4226	4248	4270	4292
		4314	4336	4358	4380	4402	4424	4446	4468	4490	4510	4530	4550	4570
		4590	4610	4630	4650	4670	4690	4710	4730	4750	4770	4790	4811	4835
		4859	4883	4907	4931	4955	4979	5003	5027	5051	5075	5099	5123	5147
		5171	5195	5215	5235	5255	5275	5295	5315	5335	5355	5375	5395	5415
		5435	5455	5475	5495	5516	5540	5564	5588	5612	5636	5660	5684	5708
		5732	5756	5780	5804	5828	5852	5876	5900	5919	5938	5957	5976	5995
		6014	6033	6052	6071	6090	6109	6128	6147	6167	6186	6205	6224	6243
		6262	6281	6300	6320	6339	6358	6377	6396	6415	6434	6453	6472	6491
		6510	6529	6549	6568	6587	6606	6625	6644	6663	6680	6697	6714	6731
		6749	6767	6785	6803	6821	6840	6857	6874	6891	6908	6925	6942	6960
		6978	6996	7014	7032	7051	7069	7087	7105	7123	7141	7159	7178	7197
		7216	7235	7254	7273	7292	7312	7330	7348	7366	7384	7402	7420	7439
		7458	7477	7496	7515	7534	7553	7573	7592	7611	7630	7649	7668	7687
		7706	7725	7744	7763	7782	7801	7820	7839	7858	7877	7896	7915	7934
		7954	7975	7996	8017	8038	8059	8080	8101	8122	8143	8164	8185	8206
		8227	8248	8269	8290	8311	8332	8353	8375	8392	8409	8426	8443	8460
		8477	8495	8513	8531	8549	8567	8585	8603	10061	10062	10063	10064	10065
		10071	10083	10085	10086	10089	10090	10091	10098	10099	10100	10112	10115	10118
		10125	10126	10127	10128	10129	10146	10153	10165	10168	10180	10538		
SSWREG	001346	1083#	1255											
SSWRMK=	000000	10065	10087											
STESTN	001330	1074#	10110*	10241										
STIMES	001310	1060#	1231*	8666*	10098*	10105	10108*	10118						
STKB	001154	1011#												
STKS	001152	1010#												
STMP0	001230	1036#	8696	8740	8786	8829	8875	8918	8964	9007	9053	9096	9143	9199
		9260	9316	9377	9421	9467	9510	9545	9578	9615	9648	9682	9732	9782
		9834	9883	9926	9972	10012	10730	10740						
STMP1	001232	1037#	10730	10732	10740	10744								
STMP10	001250	1044#	10718	10756										
STMP11	001252	1045#	10708	10756										
STMP12	001254	1046#	10709	10720	10756									
STMP13	001256	1047#	10722	10756										
STMP14	001260	1048#	10760											
STMP15	001262	1049#	10710	10760										
STMP16	001264	1050#	10724	10760										
STMP17	001266	1051#	10760											
STMP2	001234	1038#	10728	10732	10734	10740	10744	10748						
STMP20	001270	1052#												
STMP21	001272	1053#	10711											
STMP22	001274	1054#	10726											
STMP23	001276	1055#												
STMP24	001300	1056#												
STMP25	001302	1057#												
STMP26	001304	1058#												
STMP27	001306	1059#												
STMP3	001236	1039#	10734	10740	10744	10748								
STMP4	001240	1040#	10704	10729	10736	10744	10748	10752						
STMP5	001242	1041#	10705	10712	10736	10748	10752							
STMP6	001244	1042#	10706	10714	10738	10752								
STMP7	001246	1043#	10707	10716	10738	10752								
STN =	000554	743#	754	1338	1344#	1347	1356	1362#	1365	1374	1380#	1383	1392	1399#
		1401	1410	1416#	1419	1428	1434#	1437	1446	1452#	1455	1464	1470#	1473
		1482	1488#	1491	1500	1506#	1509	1518	1524#	1527	1536	1542#	1545	1555

1561#	1564	1575	1581#	1584	1595	1601#	1604	1615	1621#	1624	1635	1641#
1644	1655	1661#	1664	1675	1681#	1684	1695	1701#	1704	1715	1721#	1724
1735	1741#	1744	1755	1761#	1764	1775	1781#	1784	1795	1801#	1804	1814
1820#	1823	1833	1839#	1842	1852	1858#	1861	1871	1877#	1880	1890	1896#
1899	1909	1915#	1918	1928	1934#	1937	1947	1953#	1956	1966	1972#	1975
1985	1991#	1994	2004	2010#	2013	2023	2029#	2032	2042	2048#	2051	2051
2067#	2070	2080	2086#	2089	2099	2105#	2108	2118	2124#	2127	2137	2143#
2146	2157	2163#	2166	2179	2185#	2188	2201	2207#	2210	2223	2229#	2232
2245	2251#	2254	2267	2273#	2276	2289	2295#	2298	2311	2317#	2320	2333
2339#	2342	2355	2361#	2364	2377	2383#	2386	2399	2405#	2408	2421	2427#
2430	2443	2449#	2452	2465	2471#	2474	2487	2493#	2496	2509	2515#	2518
2531	2537#	2540	2552	2559#	2562	2576	2582#	2585	2595	2601#	2604	2614
2620#	2623	2633	2639#	2642	2652	2658#	2661	2671	2677#	2680	2690	2696#
2699	2709	2715#	2718	2728	2734#	2737	2747	2753#	2756	2766	2772#	2775
2785	2791#	2794	2804	2810#	2813	2823	2829#	2832	2843	2849#	2852	2865
2871#	2874	2887	2893#	2896	2909	2915#	2918	2931	2937#	2940	2953	2959#
2962	2975	2981#	2984	2997	3003#	3006	3019	3025#	3028	3041	3047#	3050
3063	3069#	3072	3085	3091#	3094	3107	3113#	3116	3129	3135#	3138	3151
3157#	3160	3170	3176#	3179	3189	3195#	3198	3208	3214#	3217	3227	3233#
3236	3246	3252#	3255	3265	3271#	3274	3284	3290#	3293	3303	3309#	3312
3322	3328#	3331	3341	3347#	3350	3360	3366#	3369	3379	3385#	3388	3398
3404#	3407	3417	3423#	3426	3436	3442#	3445	3456	3462#	3465	3478	3484#
3487	3500	3506#	3509	3522	3528#	3531	3544	3550#	3553	3566	3572#	3575
3588	3594#	3597	3610	3616#	3619	3632	3638#	3641	3654	3660#	3663	3676
3682#	3685	3698	3704#	3707	3720	3726#	3729	3742	3748#	3751	3764	3770#
3773	3786	3792#	3795	3808	3814#	3817	3827	3833#	3836	3846	3852#	3855
3865	3871#	3874	3884	3890#	3893	3903	3909#	3912	3922	3928#	3931	3941
3947#	3950	3960	3966#	3969	3979	3985#	3988	3998	4004#	4007	4017	4023#
4026	4036	4042#	4045	4055	4061#	4064	4074	4080#	4083	4093	4099#	4102
4112	4118#	4121	4132	4138#	4141	4154	4160#	4163	4176	4182#	4185	4198
4204#	4207	4220	4226#	4229	4242	4248#	4251	4264	4270#	4273	4286	4292#
4295	4308	4314#	4317	4330	4336#	4339	4352	4358#	4361	4374	4380#	4383
4396	4402#	4405	4418	4424#	4427	4440	4446#	4449	4462	4468#	4471	4484
4490#	4493	4504	4510#	4513	4524	4530#	4533	4544	4550#	4553	4564	4570#
4573	4584	4590#	4593	4604	4610#	4613	4624	4630#	4633	4644	4650#	4653
4664	4670#	4673	4684	4690#	4693	4704	4710#	4713	4724	4730#	4733	4744
4750#	4753	4764	4770#	4773	4784	4790#	4793	4805	4811#	4814	4829	4835#
4838	4853	4859#	4862	4877	4883#	4886	4901	4907#	4910	4925	4931#	4934
4949	4955#	4958	4973	4979#	4982	4997	5003#	5006	5021	5027#	5030	5045
5051#	5054	5069	5075#	5078	5093	5099#	5102	5117	5123#	5126	5141	5147#
5150	5165	5171#	5174	5189	5195#	5198	5209	5215#	5218	5229	5235#	5238
5249	5255#	5258	5269	5275#	5278	5289	5295#	5298	5309	5315#	5318	5329
5335#	5338	5349	5355#	5358	5369	5375#	5378	5389	5395#	5398	5409	5415#
5418	5429	5435#	5438	5449	5455#	5458	5469	5475#	5478	5489	5495#	5498
5510	5516#	5519	5534	5540#	5543	5558	5564#	5567	5582	5588#	5591	5606
5612#	5615	5630	5636#	5639	5654	5660#	5663	5678	5684#	5687	5702	5708#
5711	5726	5732#	5735	5750	5756#	5759	5774	5780#	5783	5798	5804#	5807
5822	5828#	5831	5846	5852#	5855	5870	5876#	5879	5894	5900#	5903	5913
5919#	5922	5932	5938#	5941	5951	5957#	5960	5970	5976#	5979	5989	5995#
5998	6008	6014#	6017	6027	6033#	6036	6046	6052#	6055	6065	6071#	6074
6084	6090#	6093	6103	6109#	6112	6122	6128#	6131	6141	6147#	6150	6161
6167#	6170	6180	6186#	6189	6199	6205#	6208	6218	6224#	6227	6237	6243#
6246	6256	6262#	6265	6275	6281#	6284	6294	6300#	6303	6314	6320#	6323
6333	6339#	6342	6352	6358#	6361	6371	6377#	6380	6390	6396#	6399	6409
6415#	6418	6428	6434#	6437	6447	6453#	6456	6466	6472#	6475	6485	6491#
6494	6504	6510#	6513	6523	6529#	6532	6543	6549#	6552	6562	6568#	6571

COMP4	743#														
COMP40	743#														
COMP41	743#	7137													
COMP42	743#														
COMP43	743#	7416													
COMP44	743#	7269													
COMP45	743#														
COMP46	743#	7549													
COMP47	743#														
ENDCOM	883#														
ERRCMP	743#	9713	9718	9763	9768	9813	9865	9994	9999	10034	10039				
ERRLUR	743#	10179													
ERROR	777#	8713	8721	8728	8757	8765	8776	8803	8811	8818	8846	8854	8865	8892	8900
	8907	8935	8943	8954	8981	8989	8996	9024	9032	9043	9070	9078	9085	9113	9121
	9132	9163	9171	9179	9187	9219	9227	9239	9251	9280	9288	9296	9304	9336	9344
	9356	9368	9394	9402	9409	9438	9446	9457	9483	9491	9498	9525	9536	9559	9566
	9592	9603	9629	9636	9662	9673	9699	9707	9715	9720	9749	9757	9765	9770	9799
	9807	9815	9822	9851	9859	9867	9874	9899	9907	9915	9942	9950	9962	9988	9996
	10001	10028	10036	10041	10052										
ESCAPE	883#														
FCOM0	743#	1340	1358	1376	1394	1412	1430	1448	1466	1484	1502	1538	1557	1577	1597
	1617	1637	1657	1677	1697	1717	1737	1777	1797	1816	1835	1854	1873	1892	1911
	1930	1949	1968	1987	2006	2025	2044	2063	2082	2159	2181	2203	2225	2247	2269
	2291	2313	2335	2357	2379	2401	2423	2445	2467	2489	2578	2597	2616	2635	2654
	2673	2692	2711	2749	2768	2806	2845	2867	2889	2911	2933	2955	2977	2999	3043
	3087	3109	3153	3172	3191	3210	3229	3248	3267	3286	3305	3324	3343	3362	3381
	3458	3480	3502	3524	3546	3568	3590	3612	3634	3656	3673	3722	3766	3810	3829
	3848	3867	3886	3905	3924	3943	3962	3981	4000	4019	4057	4095	4134	4156	4178
	4200	4222	4244	4266	4288	4310	4332	4354	4398	4442	4486	4506	4546	4566	4586
	4606	4626	4646	4666	4686	4706	4726	4766	4807	4831	4879	4903	4927	4951	4975
	4999	5023	5047	5071	5095	5143	5191	5211	5251	5271	5291	5311	5331	5351	5371
	5391	5411	5431	5471	5512	5536	5584	5608	5632	5656	5680	5704	5728	5752	5776
	5800	5848	5896	5915	5934	5953	5972	5991	6010	6029	6048	6067	6086	6105	6163
	6182	6201	6220	6239	6258	6296	6316	6335	6354	6373	6392	6411	6430	6449	6468
	6487	6506	6545	6564	6583	6602	6621	6640	6659	6676	6693	6710	6727	6745	6763
	6781	6799	6817	6836	6853	6870	6887	6904	6921	6938	6956	6974	6992	7010	7028
	7047	7065	7083	7101	7119	7155	7174	7193	7212	7231	7250	7288	7308	7326	7344
	7362	7380	7398	7435	7454	7473	7492	7511	7530	7569	7588	7607	7626	7645	7664
	7683	7702	7721	7740	7759	7778	7797	7816	7950	7971	7992	8013	8034	8055	8076
	8097	8118	8139	8160	8181	8202	8223	8371	8388	8405	8422	8439	8456	8473	8491
	8509	8527	8545	8563	8581	8599									
FCOM1	743#	1520	1757	2120	2533	2730	3021	3438	3788	4038	4376	4526	4855	5231	5560
	6143	6277													
FCOM2	743#	2139	2511	2825	3065	3419	3744	4114	4464	4786	5167	5491	5872	7873	7892
	7911	7930	8286	8307	8328	8349									
FCOM3	743#	2101	2555	2787	3131	3400	3700	4076	4420	4746	5119	5451	5824	6124	6525
	7835	7854	8244	8265											
FCOM4	743#	7137	7269	7416	7549										
FPRGTO	743#														
FPRGT1	743#														
FPSFEC	743#	8711	8755	8801	8844	8890	8933	8979	9022	9068	9111	9161	9217	9278	9334
	9392	9436	9481	9697	9747	9797	9849	9897	9940						
FPSTST	743#	9523	9557	9590	9627	9660	9986	10026							
GENCOM	743#														
GENTS1	743#	1338	1356	1374	1392	1410	1428	1446	1464	1482	1500	1518	1536	1555	1575
	1595	1615	1635	1655	1675	1695	1715	1735	1755	1775	1795	1814	1833	1852	1871

	1890	1909	1928	1947	1966	1985	2004	2023	2042	2061	2080	2099	2118	2137	2157
	2179	2201	2223	2245	2267	2289	2311	2333	2355	2377	2399	2421	2443	2465	2487
	2509	2531	2553	2576	2595	2614	2633	2652	2671	2690	2709	2728	2747	2766	2785
	2804	2823	2843	2865	2887	2909	2931	2953	2975	2997	3019	3041	3063	3085	3107
	3129	3151	3170	3189	3208	3227	3246	3265	3284	3303	3322	3341	3360	3379	3398
	3417	3436	3456	3478	3500	3522	3544	3566	3588	3610	3632	3654	3676	3698	3720
	3742	3764	3786	3808	3827	3846	3865	3884	3903	3922	3941	3960	3979	3998	4017
	4036	4055	4074	4093	4112	4132	4154	4176	4198	4220	4242	4264	4286	4308	4330
	4352	4374	4396	4418	4440	4462	4484	4504	4524	4544	4564	4584	4604	4624	4644
	4664	4684	4704	4724	4744	4764	4784	4805	4829	4853	4877	4901	4925	4949	4973
	4997	5021	5045	5069	5093	5117	5141	5165	5189	5209	5229	5249	5269	5289	5309
	5329	5349	5369	5389	5409	5429	5449	5469	5489	5510	5534	5558	5582	5606	5630
	5654	5678	5702	5726	5750	5774	5798	5822	5846	5870	5894	5913	5932	5951	5970
	5989	6008	6027	6046	6065	6084	6103	6122	6141	6161	6180	6199	6218	6237	6256
	6275	6294	6314	6333	6352	6371	6390	6409	6428	6447	6466	6485	6504	6523	6543
GENTS2	6562	6581	6600	6619	6638										
	743#	6657	6674	6691	6708	6725	6743	6761	6779	6797	6815	6834	6851	6868	6885
	6902	6919	6936	6954	6972	6990	7008	7026	7045	7063	7081	7099	7117	7135	7153
	7172	7191	7210	7229	7248	7267	7286	7306	7324	7342	7360	7378	7396	7414	7433
	7452	7471	7490	7509	7528	7547									
GENTS3	743#	7567	7586	7605	7624	7643	7662	7681	7700	7719	7738	7757	7776	7795	7814
	7833	7852	7871	7890	7909	7928	7948	7969	7990	8011	8032	8053	8074	8095	8116
	8137	8158	8179	8200	8221	8242	8263	8284	8305	8326	8347	8369	8386	8403	8420
	8437	8454	8471	8489	8507	8525	8543	8561	8579	8597					
GENTS4	743#														
GETPRI	883#														
GETSMR	883#														
GTSTD	743#	8824	8913	9002	9091										
GTSTF	743#	8781	8870	8959	9048										
HTSTD	743#														
HTSTF	743#														
MOVDIS	743#	8693	8737	8783	8826	8872	8915	8961	9004	9050	9093	9140	9196	9257	9313
	9374	9418	9464	9507	9542	9575	9612	9645	9679	9729	9779	9831	9880	9923	9969
	10009														
MOVDI1	8694#	8738#	8784#	8827#	8873#	8916#	8962#	9005#	9051#	9094#	9141#	9197#	9258#	9314#	9375#
	9419#	9465#	9508#	9543#	9576#	9613#	9646#	9680#	9730#	9780#	9832#	9881#	9924#	9970#	10010#
MOVDI2	8694#	8738#	8784#	8827#	8873#	8916#	8962#	9005#	9051#	9094#	9141#	9197#	9258#	9314#	9375#
	9419#	9465#	9508#	9543#	9576#	9613#	9646#	9680#	9730#	9780#	9832#	9881#	9924#	9970#	10010#
MULT	883#														
NEWTST	883#	1338	1356	1374	1392	1410	1428	1446	1464	1482	1500	1518	1536	1555	1575
	1595	1615	1635	1655	1675	1695	1715	1735	1755	1775	1795	1814	1833	1852	1871
	1890	1909	1928	1947	1966	1985	2004	2023	2042	2061	2080	2099	2118	2137	2157
	2179	2201	2223	2245	2267	2289	2311	2333	2355	2377	2399	2421	2443	2465	2487
	2509	2531	2553	2576	2595	2614	2633	2652	2671	2690	2709	2728	2747	2766	2785
	2804	2823	2843	2865	2887	2909	2931	2953	2975	2997	3019	3041	3063	3085	3107
	3129	3151	3170	3189	3208	3227	3246	3265	3284	3303	3322	3341	3360	3379	3398
	3417	3436	3456	3478	3500	3522	3544	3566	3588	3610	3632	3654	3676	3698	3720
	3742	3764	3786	3808	3827	3846	3865	3884	3903	3922	3941	3960	3979	3998	4017
	4036	4055	4074	4093	4112	4132	4154	4176	4198	4220	4242	4264	4286	4308	4330
	4352	4374	4396	4418	4440	4462	4484	4504	4524	4544	4564	4584	4604	4624	4644
	4664	4684	4704	4724	4744	4764	4784	4805	4829	4853	4877	4901	4925	4949	4973
	4997	5021	5045	5069	5093	5117	5141	5165	5189	5209	5229	5249	5269	5289	5309
	5329	5349	5369	5389	5409	5429	5449	5469	5489	5510	5534	5558	5582	5606	5630
	5654	5678	5702	5726	5750	5774	5798	5822	5846	5870	5894	5913	5932	5951	5970
	5989	6008	6027	6046	6065	6084	6103	6122	6141	6161	6180	6199	6218	6237	6256
	6275	6294	6314	6333	6352	6371	6390	6409	6428	6447	6466	6485	6504	6523	6543

	6562	6581	6600	6619	6638	6657	6674	6691	6708	6725	6743	6761	6779	6797	6815
	6834	6851	6868	6885	6902	6919	6936	6954	6972	6990	7008	7026	7045	7063	7081
	7099	7117	7135	7153	7172	7191	7210	7229	7248	7267	7286	7306	7324	7342	7360
	7378	7396	7414	7433	7452	7471	7490	7509	7528	7547	7567	7586	7605	7624	7643
	7662	7681	7700	7719	7738	7757	7776	7795	7814	7833	7852	7871	7890	7909	7928
	7948	7969	7990	8011	8032	8053	8074	8095	8116	8137	8158	8179	8200	8221	8242
	8263	8284	8305	8326	8347	8369	8386	8403	8420	8437	8454	8471	8489	8507	8525
	8543	8561	8579	8597											
POP	883#	10373	10374	10525	10526										
PUSH	883#	10334	10336	10357	10504	10510									
REPORT	883#														
SBTST1	743#	1338	1356	1374	1392	1410	1428	1446	1464	1482	1500	1518	1536	1555	1575
	1595	1615	1635	1655	1675	1695	1715	1735	1755	1775	1795	1814	1833	1852	1871
	1890	1909	1928	1947	1966	1985	2004	2023	2042	2061	2080	2099	2118	2137	2157
	2179	2201	2223	2245	2267	2289	2311	2333	2355	2377	2399	2421	2443	2465	2487
	2509	2531	2553	2576	2595	2614	2633	2652	2671	2690	2709	2728	2747	2766	2785
	2804	2823	2843	2865	2887	2909	2931	2953	2975	2997	3019	3041	3063	3085	3107
	3129	3151	3170	3189	3208	3227	3246	3265	3284	3303	3322	3341	3360	3379	3398
	3417	3436	3456	3478	3500	3522	3544	3566	3588	3610	3632	3654	3676	3698	3720
	3742	3764	3786	3808	3827	3846	3865	3884	3903	3922	3941	3960	3979	3998	4017
	4036	4055	4074	4093	4112	4132	4154	4176	4198	4220	4242	4264	4286	4308	4330
	4352	4374	4396	4418	4440	4462	4484	4504	4524	4544	4564	4584	4604	4624	4644
	4664	4684	4704	4724	4744	4764	4784	4805	4829	4853	4877	4901	4925	4949	4973
	4997	5021	5045	5069	5093	5117	5141	5165	5189	5209	5229	5249	5269	5289	5309
	5329	5349	5369	5389	5409	5429	5449	5469	5489	5510	5534	5558	5582	5606	5630
	5654	5678	5702	5726	5750	5774	5798	5822	5846	5870	5894	5913	5932	5951	5970
	5989	6008	6027	6046	6065	6084	6102	6122	6141	6161	6180	6199	6218	6237	6256
	6275	6294	6314	6333	6352	6371	6390	6409	6428	6447	6466	6485	6504	6523	6543
	6562	6581	6600	6619	6638	6657	6674	6691	6708	6725	6743	6761	6779	6797	6815
	6834	6851	6868	6885	6902	6919	6936	6954	6972	6990	7008	7026	7045	7063	7081
	7099	7117	7135	7153	7172	7191	7210	7229	7248	7267	7286	7306	7324	7342	7360
	7378	7396	7414	7433	7452	7471	7490	7509	7528	7547	7567	7586	7605	7624	7643
	7662	7681	7700	7719	7738	7757	7776	7795	7814	7833	7852	7871	7890	7909	7928
	7948	7969	7990	8011	8032	8053	8074	8095	8116	8137	8158	8179	8200	8221	8242
	8263	8284	8305	8326	8347	8369	8386	8403	8420	8437	8454	8471	8489	8507	8525
	8543	8561	8579	8597											
SBTST2	743#														
SCOM0	743#	1467	1539	1817	1874	1969	2026	2140	2579	2655	3154	3211	3249	3268	3401
	3887	4039	4077	4527	4627	4727	4747	5232	5332	5432	5452	5897	5973	6049	6546
SCOM1	6584	6660	6694	6728	7102	7120	7156	7627	7703	8423	8440				
	743#	1341	1395	1413	1485	1798	1893	1950	2045	2598	2674	2731	2750	2807	2826
	3192	3287	3363	3382	3811	3849	3925	3963	4001	4115	4547	4607	5252	5312	5935
	5992	6068	6144	6622	6677	6711	7048	7066	7084	7138	7608	7684	7760	7798	7893
	7931	8372	8474												
SCOM2	743#	1359	1377	1431	1503	1855	1912	2007	2064	2102	2617	2693	2769	2788	3173
	3306	3344	3439	3830	3868	3944	3982	4058	4487	4587	4667	4707	5192	5292	5372
	5412	5916	6011	6087	6125	6565	6641	6837	6871	6888	6905	7309	7327	7381	7399
	7589	7665	7741	7779	7855	7874	7912	8389	8457						
SCOM3	743#	1449	1521	1836	1931	1988	2083	2636	2712	3230	3325	3420	3906	4020	4096
	4507	4567	4647	4687	4767	4787	5212	5272	5352	5392	5472	5492	5954	6030	6106
	6603	6854	6922	6939	7345	7363	7417	7570	7646	7722	7817	7836	8406		
SCOM4	743#	1718	2204	2248	2336	2402	2446	2556	2934	2978	3000	3110	3132	3481	3525
	3613	4135	4223	4267	4399	4421	4904	4952	5000	5072	5144	5609	5657	5705	5777
	5849	6183	6259	6374	6431	6450	6526	6746	6800	7194	7213	7232	7270	7993	8077
	8161	8308	8528	8600											
SCOM5	743#	1618	1638	1678	1698	1778	2226	2270	2468	2890	3591	3679	3723	3767	4179

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 264
CROSS REFERENCE TABLE -- MACRO NAMES

SCOM6	4245 6507 743	4311 6764 1558	4355 6782 1578	4377 6818 1598	4808 7175 1658	4856 7251 1758	5048 7289 2121	5096 7951 2160	5513 8035 2292	5561 8119 2380	5753 8203 2424	5801 8245 2490	6164 8350 2512	6240 8510 2868	6355 8582 3022
SCOM7	3044 5585 7972 743	3066 5873 8056 1738	3459 6221 8140 2182	3569 6297 8224 2314	3657 6336 8266 2358	3745 6393 8287 2534	4157 6488 8492 2846	4289 6957 8546 2912	4333 7011 8564 2956	4443 7436 8564 3088	4465 7455 8564 3503	4832 7474 8564 3547	4880 7493 8564 3635	5168 7531 8564 3701	5537 7550 8564 3789
SCOPE	6993 778	7029 1343	7512 1361	8014 1379	8098 1397	8182 1415	8329 1433	8564 1451	1469 1760	1487 1780	1505 1800	1523 1819	1541 1838	1560 1857	1580 1876
SCPLUR	1600 1895 2184 2514 2809 3134 3422 3747 4041 4357 4669 5002 5334 5659 5994 6280 6567 6839 7104 7383 7667 7953 8268 8548	1620 1914 2206 2536 2828 3156 3441 3769 4060 4379 4689 5026 5354 5683 6013 6299 6586 6856 7122 7401 7686 7974 8289 8566	1640 1933 2228 2558 2848 3175 3461 3791 4079 4401 4709 5050 5374 5707 6032 6319 6605 6873 7140 7419 7705 7995 8310 8584	1660 1952 2250 2581 2870 3194 3483 3813 4098 4423 4729 5074 5394 5731 6051 6338 6624 6890 7158 7438 7724 8016 8331 8602	1680 1971 2272 2600 2892 3213 3505 3832 4117 4445 4749 5098 5414 5755 6070 6357 6643 6907 7177 7457 7743 8037 8352 8620	1700 1990 2294 2619 2914 3232 3527 3851 4137 4467 4769 5122 5434 5779 6089 6376 6662 6924 7196 7476 7762 8058 8374 8620	1720 2009 2316 2638 2936 3251 3549 3870 4159 4489 4789 5146 5454 5803 6108 6395 6679 6941 7215 7495 7781 8079 8391 8620	1740 2028 2338 2657 2958 3270 3571 3889 4181 4509 4810 5170 5474 5827 6127 6414 6696 6959 7234 7514 7800 8100 8408 8620	1760 2047 2360 2676 2980 3289 3593 3908 4203 4529 4834 5194 5494 5851 6146 6433 6713 6977 7253 7533 7819 8121 8425 8620	1780 2066 2382 2695 2980 3308 3615 3927 4203 4549 4858 5214 5515 5875 6166 6452 6730 6995 7272 7552 7838 8142 8442 8620	1800 2085 2404 2714 3002 3327 3637 3946 4225 4569 4882 5234 5539 5899 6185 6471 6748 7013 7291 7572 7857 8163 8459 8620	1819 2104 2426 2733 3046 3346 3659 3965 4247 4589 4906 5254 5563 5918 6204 6490 6766 7031 7311 7591 7876 8184 8476 8620	1838 2123 2448 2752 3068 3365 3681 3984 4269 4589 4930 5294 5587 5937 6223 6509 6784 7050 7329 7610 7895 8205 8494 8620	1857 2142 2470 2771 3090 3384 3703 4003 4291 4609 4954 5294 5611 5956 6242 6528 6802 7068 7347 7629 7914 8226 8512 8620	1876 2162 2492 2790 3112 3403 3725 4022 4335 4649 4978 5314 5635 5975 6261 6548 6820 7086 7365 7648 7933 8247 8530 8620
SEADAT	743	10070 1352	1370 1651	1388 1671	1406 1691	1424 1711	1442 1731	1460 1751	1478 1771	1496 1791	1514 1810	1532 1829	1550 1848	1571 1867	1591 1886
	1611 1905 2197 2527 2819 3147 3432 3760 4051 4370 4680 5017 5345 5674 6004 6290 6578 6848 7113 7392	1631 1924 2219 2549 2838 3166 3451 3782 4070 4392 4700 5041 5365 5698 6023 6309 6597 6865 7131 7410	1651 1943 2241 2571 2861 3185 3474 3804 4089 4414 4720 5065 5385 5722 6042 6329 6616 6882 7149 7428	1671 1962 2263 2591 2883 3204 3496 3823 4108 4436 4740 5089 5405 5746 6061 6348 6635 6899 7167 7448	1691 1981 2285 2610 2905 3223 3518 3842 4127 4458 4760 5113 5425 5770 6080 6367 6654 6916 7187 7467	1711 2000 2307 2629 2927 3242 3540 3861 4150 4480 4780 5137 5445 5794 6099 6386 6671 6933 7206 7486	1731 2019 2329 2648 2949 3261 3562 3880 4172 4500 4800 5161 5465 5818 6118 6405 6688 6950 7225 7505	1751 2038 2351 2667 2971 3280 3584 3899 4194 4520 4825 5185 5485 5842 6137 6424 6705 6969 7244 7524	1771 2057 2373 2686 2993 3299 3606 3918 4216 4540 4849 5205 5505 5866 6156 6443 6722 6987 7263 7543	1791 2076 2395 2705 2993 3318 3628 3937 4238 4560 4873 5225 5530 5890 6176 6462 6739 7005 7282 7562	1810 2095 2417 2724 3037 3337 3650 3956 4260 4580 4897 5245 5554 5909 6195 6481 6758 7023 7301 7582	1829 2114 2439 2743 3059 3356 3672 3975 4282 4600 4921 5265 5578 5928 6214 6500 6776 7041 7320 7601	1848 2133 2461 2762 3081 3375 3694 3994 4304 4620 4945 5285 5602 5947 6233 6519 6794 7059 7338 7620	1867 2152 2483 2781 3103 3394 3716 4013 4326 4640 4969 5305 5626 5966 6252 6538 6812 7077 7356 7639	1886 2175 2505 2800 3125 3413 3738 4032 4348 4660 4993 5325 5650 5985 6271 6559 6830 7095 7374 7658

	7677	7696	7715	7734	7753	7772	7791	7810	7829	7848	7867	7886	7905	7924	7943
	7965	7986	8007	8028	8049	8070	8091	8112	8133	8154	8175	8196	8217	8238	8259
	8280	8301	8322	8343	8364	8383	8400	8417	8434	8451	8468	8485	8504	8522	8540
	8558	8576	8594	8612											
SETPRI	883#														
SETREG	743#	10134													
SETTRA	10484#	10493	10494	10495											
SETUP	743#	883#	1213												
SKIP	883#	1347	1365	1383	1401	1419	1437	1455	1473	1491	1509	1527	1545	1564	1584
	1604	1624	1644	1664	1684	1704	1724	1744	1764	1784	1804	1823	1842	1861	1880
	1899	1918	1937	1956	1975	1994	2013	2032	2051	2070	2089	2108	2127	2146	2166
	2188	2210	2232	2254	2276	2298	2320	2342	2364	2386	2408	2430	2452	2474	2496
	2518	2540	2562	2585	2604	2623	2642	2661	2680	2699	2718	2737	2756	2775	2794
	2813	2832	2852	2874	2896	2918	2940	2962	2984	3006	3028	3050	3072	3094	3116
	3138	3160	3179	3198	3217	3236	3255	3274	3293	3312	3331	3350	3369	3388	3407
	3426	3445	3465	3487	3509	3531	3553	3575	3597	3619	3641	3663	3685	3707	3729
	3751	3773	3795	3817	3836	3855	3874	3893	3912	3931	3950	3969	3988	4007	4026
	4045	4064	4083	4102	4121	4141	4163	4185	4207	4229	4251	4273	4295	4317	4339
	4361	4383	4405	4427	4449	4471	4493	4513	4533	4553	4573	4593	4613	4633	4653
	4673	4693	4713	4733	4753	4773	4793	4814	4838	4862	4886	4910	4934	4958	4982
	5006	5030	5054	5078	5102	5126	5150	5174	5198	5218	5238	5258	5278	5298	5318
	5338	5358	5378	5398	5418	5438	5458	5478	5498	5519	5543	5567	5591	5615	5639
	5663	5687	5711	5735	5759	5783	5807	5831	5855	5879	5903	5922	5941	5960	5979
	5998	6017	6036	6055	6074	6093	6112	6131	6150	6170	6189	6208	6227	6246	6265
	6284	6303	6323	6342	6361	6380	6399	6418	6437	6456	6475	6494	6513	6532	6552
	6571	6590	6609	6628	6647	6666	6683	6700	6717	6734	6752	6770	6788	6806	6824
	6843	6860	6877	6894	6911	6928	6945	6963	6981	6999	7017	7035	7054	7072	7090
	7108	7126	7144	7162	7181	7200	7219	7238	7257	7276	7295	7315	7333	7351	7369
	7387	7405	7423	7442	7461	7480	7499	7518	7537	7556	7576	7595	7614	7633	7652
	7671	7690	7709	7728	7747	7766	7785	7804	7823	7842	7861	7880	7899	7918	7937
	7957	7978	7999	8020	8041	8062	8083	8104	8125	8146	8167	8188	8209	8230	8251
	8272	8293	8314	8335	8356	8378	8395	8412	8429	8446	8463	8480	8498	8516	8534
	8552	8570	8588	8606											
SLASH	883#														
SPACE	883#														
STARS	883#														
	948	959	961	968	982	1066	1069	1209	1211	1294	1296	1311	1313	1338	
	1342	1356	1360	1374	1378	1392	1396	1410	1414	1428	1432	1446	1450	1464	1468
	1482	1486	1500	1504	1518	1522	1536	1540	1555	1559	1575	1579	1595	1599	1615
	1619	1635	1639	1655	1659	1675	1679	1695	1699	1715	1719	1735	1739	1755	1759
	1775	1779	1795	1799	1814	1818	1833	1837	1852	1856	1871	1875	1890	1894	1909
	1913	1928	1932	1947	1951	1966	1970	1985	1989	2004	2008	2023	2027	2042	2046
	2061	2065	2080	2084	2099	2103	2118	2122	2137	2141	2157	2161	2179	2183	2201
	2205	2223	2227	2245	2249	2267	2271	2289	2293	2311	2315	2333	2337	2355	2359
	2377	2381	2399	2403	2421	2425	2443	2447	2465	2469	2487	2491	2509	2513	2531
	2535	2553	2557	2576	2580	2595	2599	2614	2618	2633	2637	2652	2656	2671	2675
	2690	2694	2709	2713	2728	2732	2747	2751	2766	2770	2785	2789	2804	2808	2823
	2827	2843	2847	2865	2869	2887	2891	2909	2913	2931	2935	2953	2957	2975	2979
	2997	3001	3019	3023	3041	3045	3063	3067	3085	3089	3107	3111	3129	3133	3151
	3155	3170	3174	3189	3193	3208	3212	3227	3231	3246	3250	3265	3269	3284	3288
	3303	3307	3322	3326	3341	3345	3360	3364	3379	3383	3398	3402	3417	3421	3436
	3440	3456	3460	3478	3482	3500	3504	3522	3526	3544	3548	3566	3570	3588	3592
	3610	3614	3632	3636	3654	3658	3676	3680	3698	3702	3720	3724	3742	3746	3764
	3768	3786	3790	3808	3812	3827	3831	3846	3850	3865	3869	3884	3888	3903	3907
	3922	3926	3941	3945	3960	3964	3979	3983	3998	4002	4017	4021	4036	4040	4055
	4059	4074	4078	4093	4097	4112	4116	4132	4136	4154	4158	4176	4180	4198	4202
	4220	4224	4242	4246	4264	4268	4286	4290	4308	4312	4330	4334	4352	4356	4374

4378	4396	4400	4418	4422	4440	4444	4462	4466	4484	4488	4504	4508	4524	4528	
4544	4548	4564	4568	4584	4588	4604	4608	4624	4628	4644	4648	4664	4668	4684	
4688	4704	4708	4724	4728	4744	4748	4764	4768	4784	4788	4805	4809	4829	4833	
4853	4857	4877	4881	4901	4905	4925	4929	4949	4953	4973	4977	4997	5001	5021	
5025	5045	5049	5069	5073	5093	5097	5117	5121	5141	5145	5165	5169	5189	5193	
5209	5213	5229	5233	5249	5253	5269	5273	5289	5293	5309	5313	5329	5333	5349	
5353	5369	5373	5389	5393	5409	5413	5429	5433	5449	5453	5469	5473	5489	5493	
5510	5514	5534	5538	5558	5562	5582	5586	5606	5610	5630	5634	5654	5658	5678	
5682	5702	5706	5726	5730	5750	5754	5774	5778	5798	5802	5822	5826	5846	5850	
5870	5874	5894	5898	5913	5917	5932	5936	5951	5955	5970	5974	5989	5993	6008	
6012	6027	6031	6046	6050	6065	6069	6084	6088	6103	6107	6122	6126	6141	6145	
6161	6165	6180	6184	6199	6203	6218	6222	6237	6241	6256	6260	6275	6279	6294	
6298	6314	6318	6333	6337	6352	6356	6371	6375	6390	6394	6409	6413	6428	6432	
6447	6451	6466	6470	6485	6489	6504	6508	6523	6527	6543	6547	6562	6566	6581	
6585	6600	6604	6619	6623	6638	6642	6657	6661	6674	6678	6691	6695	6708	6712	
6725	6729	6743	6747	6761	6765	6779	6783	6797	6801	6815	6819	6834	6838	6851	
6855	6868	6872	6885	6889	6902	6906	6919	6923	6936	6940	6954	6958	6972	6976	
6990	6994	7008	7012	7026	7030	7045	7049	7063	7067	7081	7085	7099	7103	7117	
7121	7135	7139	7153	7157	7172	7176	7191	7195	7210	7214	7229	7233	7248	7252	
7267	7271	7286	7290	7306	7310	7324	7328	7342	7346	7360	7364	7378	7382	7396	
7400	7414	7418	7433	7437	7452	7456	7471	7475	7490	7494	7509	7513	7528	7532	
7547	7551	7567	7571	7586	7590	7605	7609	7624	7628	7643	7647	7662	7666	7681	
7685	7700	7704	7719	7723	7738	7742	7757	7761	7776	7780	7795	7799	7814	7818	
7833	7837	7852	7856	7871	7875	7890	7894	7909	7913	7928	7932	7948	7952	7969	
7973	7990	7994	8011	8015	8032	8036	8053	8057	8074	8078	8095	8099	8116	8120	
8137	8141	8158	8162	8179	8183	8200	8204	8221	8225	8242	8246	8263	8267	8284	
8288	8305	8309	8326	8330	8347	8351	8369	8373	8386	8390	8403	8407	8420	8424	
8437	8441	8454	8458	8471	8475	8489	8493	8507	8511	8525	8529	8543	8547	8561	
8565	8579	8583	8597	8601	8616	8653	8734	8823	8912	9001	9090	9193	9310	9415	
9504	9572	9609	9642	9726	9776	9828	9920	10006	10058	10121	10180	10250	10329	10386	
10463	10500	10516													
STATUS	743#														
SWRSU	883#	1236#													
TAD001	743#														
TAD002	743#														
TAD0F1	743#														
TAD0F2	743#														
TAD0R1	743#														
TAD0R2	743#														
TRMTRP	10484#														
TYPBIN	883#														
TYPDEC	883#														
TYPNAM	883#														
TYPNUM	883#														
TYPOCS	883#														
TYPOCT	883#														
TYPTXT	883#														
UPCODE	743#	10523													
SSCHRE	980#	1020	1021	1022	1023	1024	1025	1026	1027	1028	1029	1030	1031	1032	1033
	1034	1035													
SSCHTM	980#	1036	1037	1038	1039	1040	1041	1042	1043	1044	1045	1046	1047	1048	1049
	1050	1051	1052	1053	1054	1055	1056	1057	1058	1059					
SSESCA	883#														
SSNEWT	883#	1338	1356	1374	1392	1410	1428	1446	1464	1482	1500	1518	1536	1555	1575
	1595	1615	1635	1655	1675	1695	1715	1735	1755	1775	1795	1814	1833	1852	1871
	1890	1909	1928	1947	1966	1985	2004	2023	2042	2061	2080	2099	2118	2137	2157

FPU ADVANCED INSTR TESTS
DQFPBA.P11 09-FEB-77 10:02

MACY11 27(1006) 09-FEB-77 10:05 PAGE 268
CROSS REFERENCE TABLE -- MACRO NAMES

.SAPTH	743#	957
.SAPTY	743#	10327
.SCATC	743#	934
.SCHTA	743#	980
.SEOP	743#	8653
.SERRO	743#	10119
.SPOWE	743#	10498
.SSCOP	743#	10056
.STRAP	743#	10461
.STYER	743#	10180
.STYPE	743#	10248
.STYPO	743#	10384

. ABS. 046114 000

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

DSKZ:DQFPBA,DSK 2:DQFPBA.SEO/SOL/CRF=DQFPBA.MEM,DQFPBA.MAC,DQFPBA.P11
RUN-TIME: 58 61 5 SECONDS
RUN-TIME RATIO: 933/126=7.4
CORE USED: 33K (66 PAGES)