

KD11-K

FLOAT PT UN ADV INST SET
MD-11-DQFPB-B

EP-DQFPB-B-DL-A
COPYRIGHT © 1977
FICHE 2 OF 2

JUN 1977
digital
MADE IN USA

This microfiche card contains a grid of frames. The frames on the left side of the card contain data, while the right side is mostly blank. The data in the frames is organized into columns and rows, with some frames containing headers and footers. The data appears to be a list of items or a table of values.

Frame	Content
1	Header and data
2	Header and data
3	Header and data
4	Header and data
5	Header and data
6	Header and data
7	Header and data
8	Header and data
9	Header and data
10	Header and data
11	Header and data
12	Header and data
13	Header and data
14	Header and data
15	Header and data
16	Header and data
17	Header and data
18	Header and data
19	Header and data
20	Header and data
21	Header and data
22	Header and data
23	Header and data
24	Header and data
25	Header and data
26	Header and data
27	Header and data
28	Header and data
29	Header and data
30	Header and data
31	Header and data
32	Header and data
33	Header and data
34	Header and data
35	Header and data
36	Header and data
37	Header and data
38	Header and data
39	Header and data
40	Header and data
41	Header and data
42	Header and data
43	Header and data
44	Header and data
45	Header and data
46	Header and data
47	Header and data
48	Header and data
49	Header and data
50	Header and data
51	Header and data
52	Header and data
53	Header and data
54	Header and data
55	Header and data
56	Header and data
57	Header and data
58	Header and data
59	Header and data
60	Header and data
61	Header and data
62	Header and data
63	Header and data
64	Header and data
65	Header and data
66	Header and data
67	Header and data
68	Header and data
69	Header and data
70	Header and data
71	Header and data
72	Header and data
73	Header and data
74	Header and data
75	Header and data
76	Header and data
77	Header and data
78	Header and data
79	Header and data
80	Header and data
81	Header and data
82	Header and data
83	Header and data
84	Header and data
85	Header and data
86	Header and data
87	Header and data
88	Header and data
89	Header and data
90	Header and data
91	Header and data
92	Header and data
93	Header and data
94	Header and data
95	Header and data
96	Header and data
97	Header and data
98	Header and data
99	Header and data
100	Header and data

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

B01

EOF1DZTCB0SEQ
PDP10 411

00010000

770608

PDP10 411

BDHDR1DQFPBBSEQ

00010000

770608

I D E N T I F I C A T I O N

PRODUCT CODE: MAINDEC-11-DQFPB-B-D
PRODUCT NAME: PDP-11/6X - FP11-E FLOATING POINT UNIT
 ADVANCED INSTRUCTION TESTS
DATE : MAY, 1977
MAINTAINER: DIAGNOSTIC GROUP
AUTHOR: DON NORTH
REVISED BY: DON 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.

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 (FP11-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

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 0-46346(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.

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=(00000) 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 8000")
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.

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 INDEFINITELY 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 INDEFINITELY, 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 "MHAMI" 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

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 HFP FP11-E OPTION ONLY
SW00=1 --> TEST HFP 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.

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) IF CLEAR AND UNDERFLOW, ANSWER <-- ZERO
9	001000	FIV - FLOATING OVERFLOW INTERRUPT IF SET AND OVERFLOW, SET FER, STORE ANSWER, EXPONENT WRONG BY -400(8) IF CLEAR AND OVERFLOW, ANSWER <-- ZERO
8	000400	FIC - FLOATING INTEGER CONVERSION INTERRUPT IF SET AND "STCFI" ERROR, ANSWER <-- ZERO, SET ERROR IF CLEAR AND "STCFI" ERROR, ANSWER <-- ZERO
7	000200	FD - FLOATING MODE 1=DOUBLE, 64 BIT OPERANDS (4M) 0=SINGLE, 32 BIT OPERANDS (2M)
6	000100	FL - INTEGER MODE 1=LONG, 32 BIT INTEGERS (2M) 0=SHORT, 16 BIT INTEGERS (1M)
5	000040	FT - ROUND/TRUNCATE MODE 1=TRUNCATE RESULTS 0=ROUND RESULTS
4	000020	FMM - PUT FP11-E ONLY IN MAINTENANCE MODE
3:0	000017	FN-FZ-FV-FC - FLOATING CONDITION CODES

FLOATING EXCEPTION CODES (FEC):

OCTAL	ENABLE	FUNCTION
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.

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

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

MODEL	AVERAGE EXECUTION TIME PER PASS	
	SHORTEST PASS	LONGEST PASS
PDP-11/6X MICROCODE	0:01	3:15
PDP-11/6X W/FP11-E	0:01	2:30

TIMES SPECIFIED AS (MINUTES):(SECONDS)

SHORTEST PASS ::= PASS=1, NO ITERATIONS, USING:
SWR=(004003) FOR PDP-11/6X MICROCODE
SWR=(004002) FOR PDP-11/6X W/FP11-E

LONGEST PASS ::= PASS=2, 2000, ITERATIONS/TEST, USING:
SWR=(000003) FOR PDP-11/6X MICROCODE
SWR=(000002) FOR PDP-11/6X W/FP11-E

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 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. THIS IN MANY CASES (THE "ADDF" 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 BASIC ELEMENTS. MORE COMPLEX ELEMENTS ARE TESTED AFTWARDS, 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(B) OF MEMORY. 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(B) (FROM THE SCOPE=IOT EQUATE). DEPENDING UPON THE SWITCH SETTINGS (SEE 5.2), CODE IS PRESENT TO: LOAD THE FP11 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:

- SMXCNT - MAXIMUM NUMBER OF ITERATIONS PER TEST (GENERALLY WILL BE 2000(10))
- STSTNM - A COUNTER INDICATING THE NUMBER (1-377(B)) OF THE TEST CURRENTLY BEING EXECUTED
- SLPADR - CONTAINS THE ADDRESS TO WHICH THE SCOPE ROUTINE 10200 WILL LOOP, IF THE CURRENT TEST IS BEING LOOPED UPON
- SLPERR - 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 SLPADR, ABOVE.

9.3.3 ERROR ROUTINE - SERROR

THE ERROR ROUTINE IS ENTERED WHEN THE TEST CODE HAS DETERMINED THAT AN ERROR HAS OCCURRED AS PART OF A TEST. THROUGH USE OF 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 "ENT" INSTRUCTION IS USED, TRAPPING THROUGH LOCATION 30(8). (NOTE THE EQUATE ERROR N=ENT N). THE LOWER BYTE OF THE ENT 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:

- EREGD 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 - STYPERR

THIS ROUTINE (STYPERR 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 - STYPE

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 - STYPOC

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.

14	OPERATIONAL SWITCH SETTINGS
32	BASIC DEFINITIONS
193	TRAP CATCHER
202	STARTING ADDRESS(ES)
205	ACT11 HOOKS
216	APT PARAMETER BLOCK
239	COMMON TAGS
328	APT MAILBOX-ETABLE
355	ERROR POINTER TABLE
426	PROGRAM DEFINED COMMON TAGS
467	START OF PASS ROUTINE
475	INITIALIZE THE COMMON TAGS
601	T1 TEST OF CHPF INSTR, DATA
620	T2 TEST OF CHPF INSTR, DATA
639	T3 TEST OF CHPF INSTR, DATA
658	T4 TEST OF CHPF INSTR, DATA
677	T5 TEST OF CHPF INSTR, DATA
696	T6 TEST OF CHPF INSTR, DATA
715	T7 TEST OF CHPF INSTR, DATA
734	T10 TEST OF CHPF INSTR, DATA
753	T11 TEST OF CHPF INSTR, DATA
772	T12 TEST OF CHPF INSTR, DATA
791	T13 TEST OF CHPF INSTR, DATA
810	T14 TEST OF CHPF INSTR, DATA
830	T15 TEST OF CHPF INSTR, DATA
851	T16 TEST OF CHPF INSTR, DATA
872	T17 TEST OF CHPF INSTR, DATA
893	T20 TEST OF CHPF INSTR, DATA
914	T21 TEST OF CHPF INSTR, DATA
935	T22 TEST OF CHPF INSTR, DATA
956	T23 TEST OF CHPF INSTR, DATA
977	T24 TEST OF CHPF INSTR, DATA
998	T25 TEST OF CHPF INSTR, DATA
1019	T26 TEST OF CHPF INSTR, DATA
1040	T27 TEST OF CHPF INSTR, DATA
1061	T30 TEST OF CHPF INSTR, DATA
1082	T31 TEST OF CHPF INSTR, DATA
1103	T32 TEST OF CHPF INSTR, DATA
1122	T33 TEST OF CHPF INSTR, DATA
1142	T34 TEST OF CHPF INSTR, DATA
1162	T35 TEST OF CHPF INSTR, DATA
1182	T36 TEST OF CHPF INSTR, DATA
1202	T37 TEST OF CHPF INSTR, DATA
1222	T40 TEST OF CHPF INSTR, DATA
1242	T41 TEST OF CHPF INSTR, DATA
1262	T42 TEST OF CHPF INSTR, DATA
1282	T43 TEST OF CHPF INSTR, DATA
1303	T44 TEST OF CHPF INSTR, DATA
1322	T45 TEST OF CHPF INSTR, DATA
1342	T46 TEST OF CHPF INSTR, DATA
1362	T47 TEST OF CHPF INSTR, DATA
1382	T50 TEST OF CHPF INSTR, DATA
1402	T51 TEST OF CHPF INSTR, DATA
1422	T52 TEST OF CHPF INSTR, DATA
1442	T53 TEST OF CHPF INSTR, DATA
1463	T54 TEST OF CHPF INSTR, DATA

1486	T55	TEST	OF	8000	INSTR.	DATA	SET	11-11
1509	T56	TEST	OF	8000	INSTR.	DATA	SET	11-12
1532	T57	TEST	OF	8000	INSTR.	DATA	SET	11-13
1555	T60	TEST	OF	8000	INSTR.	DATA	SET	11-14
1578	T61	TEST	OF	8000	INSTR.	DATA	SET	11-15
1601	T62	TEST	OF	8000	INSTR.	DATA	SET	11-16
1624	T63	TEST	OF	8000	INSTR.	DATA	SET	11-17
1647	T64	TEST	OF	8000	INSTR.	DATA	SET	11-18
1670	T65	TEST	OF	8000	INSTR.	DATA	SET	11-19
1693	T66	TEST	OF	8000	INSTR.	DATA	SET	11-20
1716	T67	TEST	OF	8000	INSTR.	DATA	SET	11-21
1739	T70	TEST	OF	8000	INSTR.	DATA	SET	11-22
1762	T71	TEST	OF	8000	INSTR.	DATA	SET	11-23
1785	T72	TEST	OF	8000	INSTR.	DATA	SET	11-24
1808	T73	TEST	OF	8000	INSTR.	DATA	SET	11-25
1831	T74	TEST	OF	8000	INSTR.	DATA	SET	11-26
1854	T75	TEST	OF	8000	INSTR.	DATA	SET	11-27
1877	T76	TEST	OF	8000	INSTR.	DATA	SET	11-28
1901	T77	TEST	OF	8000	INSTR.	DATA	SET	11-29
1921	T100	TEST	OF	8000	INSTR.	DATA	SET	11-30
1941	T101	TEST	OF	8000	INSTR.	DATA	SET	11-31
1961	T102	TEST	OF	8000	INSTR.	DATA	SET	11-32
1981	T103	TEST	OF	8000	INSTR.	DATA	SET	11-33
2001	T104	TEST	OF	8000	INSTR.	DATA	SET	11-34
2021	T105	TEST	OF	8000	INSTR.	DATA	SET	11-35
2041	T106	TEST	OF	8000	INSTR.	DATA	SET	11-36
2061	T107	TEST	OF	8000	INSTR.	DATA	SET	11-37
2081	T110	TEST	OF	8000	INSTR.	DATA	SET	11-38
2101	T111	TEST	OF	8000	INSTR.	DATA	SET	11-39
2121	T112	TEST	OF	8000	INSTR.	DATA	SET	11-40
2141	T113	TEST	OF	8000	INSTR.	DATA	SET	11-41
2161	T114	TEST	OF	8000	INSTR.	DATA	SET	11-42
2181	T115	TEST	OF	8000	INSTR.	DATA	SET	11-43
2201	T116	TEST	OF	8000	INSTR.	DATA	SET	11-44
2221	T117	TEST	OF	8000	INSTR.	DATA	SET	11-45
2241	T120	TEST	OF	8000	INSTR.	DATA	SET	11-46
2261	T121	TEST	OF	8000	INSTR.	DATA	SET	11-47
2281	T122	TEST	OF	8000	INSTR.	DATA	SET	11-48
2301	T123	TEST	OF	8000	INSTR.	DATA	SET	11-49
2321	T124	TEST	OF	8000	INSTR.	DATA	SET	11-50
2341	T125	TEST	OF	8000	INSTR.	DATA	SET	11-51
2361	T126	TEST	OF	8000	INSTR.	DATA	SET	11-52
2381	T127	TEST	OF	8000	INSTR.	DATA	SET	11-53
2401	T130	TEST	OF	8000	INSTR.	DATA	SET	11-54
2421	T131	TEST	OF	8000	INSTR.	DATA	SET	11-55
2441	T132	TEST	OF	8000	INSTR.	DATA	SET	11-56
2461	T133	TEST	OF	8000	INSTR.	DATA	SET	11-57
2481	T134	TEST	OF	8000	INSTR.	DATA	SET	11-58
2501	T135	TEST	OF	8000	INSTR.	DATA	SET	11-59
2521	T136	TEST	OF	8000	INSTR.	DATA	SET	11-60
2541	T137	TEST	OF	8000	INSTR.	DATA	SET	11-61
2561	T140	TEST	OF	8000	INSTR.	DATA	SET	11-62
2581	T141	TEST	OF	8000	INSTR.	DATA	SET	11-63
2601	T142	TEST	OF	8000	INSTR.	DATA	SET	11-64
2621	T143	TEST	OF	8000	INSTR.	DATA	SET	11-65
2641	T144	TEST	OF	8000	INSTR.	DATA	SET	11-66

2704	T145	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-13
2724	T146	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-14
2744	T147	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-15
2764	T150	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-16
2784	T151	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-17
2804	T152	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-20
2825	T153	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-20
2848	T154	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-20
2871	T155	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-20
2894	T156	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-20
2917	T157	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-20
2940	T160	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-20
2963	T161	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-20
2986	T162	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-10
3009	T163	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-11
3032	T164	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-11
3055	T165	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-14
3078	T166	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-14
3101	T167	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-15
3124	T170	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-16
3147	T171	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-17
3170	T172	TEST	OF	MULF	INSTR.	DATA	SET	MULF	-20
3193	T173	TEST	OF	DIVF	INSTR.	DATA	SET	DIVF	-1
3213	T174	TEST	OF	DIVF	INSTR.	DATA	SET	DIVF	-2
3233	T175	TEST	OF	DIVF	INSTR.	DATA	SET	DIVF	-3
3253	T176	TEST	OF	DIVF	INSTR.	DATA	SET	DIVF	-4
3273	T177	TEST	OF	DIVF	INSTR.	DATA	SET	DIVF	-5
3293	T200	TEST	OF	DIVF	INSTR.	DATA	SET	DIVF	-6
3313	T201	TEST	OF	DIVF	INSTR.	DATA	SET	DIVF	-7
3333	T202	TEST	OF	DIVF	INSTR.	DATA	SET	DIVF	-10
3353	T203	TEST	OF	DIVF	INSTR.	DATA	SET	DIVF	-11
3373	T204	TEST	OF	DIVF	INSTR.	DATA	SET	DIVF	-12
3393	T205	TEST	OF	DIVF	INSTR.	DATA	SET	DIVF	-13
3413	T206	TEST	OF	DIVF	INSTR.	DATA	SET	DIVF	-14
3433	T207	TEST	OF	DIVF	INSTR.	DATA	SET	DIVF	-15
3453	T210	TEST	OF	DIVF	INSTR.	DATA	SET	DIVF	-16
3473	T211	TEST	OF	DIVF	INSTR.	DATA	SET	DIVF	-17
3493	T212	TEST	OF	DIVF	INSTR.	DATA	SET	DIVF	-20
3513	T213	TEST	OF	DIVF	INSTR.	DATA	SET	DIVF	-21
3534	T214	TEST	OF	DIVD	INSTR.	DATA	SET	DIVD	-1
3557	T215	TEST	OF	DIVD	INSTR.	DATA	SET	DIVD	-2
3580	T216	TEST	OF	DIVD	INSTR.	DATA	SET	DIVD	-3
3603	T217	TEST	OF	DIVD	INSTR.	DATA	SET	DIVD	-4
3626	T220	TEST	OF	DIVD	INSTR.	DATA	SET	DIVD	-5
3649	T221	TEST	OF	DIVD	INSTR.	DATA	SET	DIVD	-6
3672	T222	TEST	OF	DIVD	INSTR.	DATA	SET	DIVD	-7
3695	T223	TEST	OF	DIVD	INSTR.	DATA	SET	DIVD	-10
3718	T224	TEST	OF	DIVD	INSTR.	DATA	SET	DIVD	-11
3741	T225	TEST	OF	DIVD	INSTR.	DATA	SET	DIVD	-12
3764	T226	TEST	OF	DIVD	INSTR.	DATA	SET	DIVD	-13
3787	T227	TEST	OF	DIVD	INSTR.	DATA	SET	DIVD	-14
3810	T230	TEST	OF	DIVD	INSTR.	DATA	SET	DIVD	-15
3833	T231	TEST	OF	DIVD	INSTR.	DATA	SET	DIVD	-16
3856	T232	TEST	OF	DIVD	INSTR.	DATA	SET	DIVD	-17
3879	T233	TEST	OF	DIVD	INSTR.	DATA	SET	DIVD	-20
3902	T234	TEST	OF	MOOF (2 ACC)	INSTR.	DATA	SET	MO2F-1	

3923	T235	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-1
3944	T236	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-2
3965	T237	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-3
3986	T240	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-4
4007	T241	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-5
4028	T242	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-6
4049	T243	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-7
4070	T244	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-8
4091	T245	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-9
4112	T246	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-10
4133	T247	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-11
4154	T250	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-12
4175	T251	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-13
4196	T252	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-14
4217	T253	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-15
4239	T254	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-16
4264	T255	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-17
4289	T256	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-18
4314	T257	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-19
4339	T260	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-20
4364	T261	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-21
4389	T262	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-22
4414	T263	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-23
4439	T264	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-24
4464	T265	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-25
4489	T266	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-26
4514	T267	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-27
4539	T270	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-28
4564	T271	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-29
4589	T272	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-30
4614	T273	TEST OF	MODF(2	ACC)	INSTR,	DATA	SET	MD10-31
4639	T274	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-32
4660	T275	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-33
4681	T276	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-34
4702	T277	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-35
4723	T300	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-36
4744	T301	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-37
4765	T302	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-38
4786	T303	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-39
4807	T304	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-40
4828	T305	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-41
4849	T306	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-42
4870	T307	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-43
4891	T310	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-44
4912	T311	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-45
4933	T312	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-46
4954	T313	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-47
4976	T314	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-48
5001	T315	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-49
5026	T316	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-50
5051	T317	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-51
5076	T320	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-52
5101	T321	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-53
5126	T322	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-54
5151	T323	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-55
5176	T324	TEST OF	MODF(1	ACC)	INSTR,	DATA	SET	MD10-56

5201	T325	TEST OF	MOOD(1	ACC)	INSTR,	DATA	SET	MD1D-12
5226	T326	TEST OF	MOOD(1	ACC)	INSTR,	DATA	SET	MD1D-13
5251	T327	TEST OF	MOOD(1	ACC)	INSTR,	DATA	SET	MD1D-14
5276	T330	TEST OF	MOOD(1	ACC)	INSTR,	DATA	SET	MD1D-15
5301	T331	TEST OF	MOOD(1	ACC)	INSTR,	DATA	SET	MD1D-16
5326	T332	TEST OF	MOOD(1	ACC)	INSTR,	DATA	SET	MD1D-17
5351	T333	TEST OF	MOOD(1	ACC)	INSTR,	DATA	SET	MD1D-20
5376	T334	TEST OF	LDCDF		INSTR,	DATA	SET	MD1D-11
5396	T335	TEST OF	LDCDF		INSTR,	DATA	SET	MD1D-11
5416	T336	TEST OF	LDCDF		INSTR,	DATA	SET	MD1D-11
5436	T337	TEST OF	LDCDF		INSTR,	DATA	SET	MD1D-11
5456	T340	TEST OF	LDCDF		INSTR,	DATA	SET	MD1D-11
5476	T341	TEST OF	LDCDF		INSTR,	DATA	SET	MD1D-11
5496	T342	TEST OF	LDCDF		INSTR,	DATA	SET	MD1D-11
5516	T343	TEST OF	LDCDF		INSTR,	DATA	SET	MD1D-11
5536	T344	TEST OF	LDCDF		INSTR,	DATA	SET	MD1D-11
5556	T345	TEST OF	LDCDF		INSTR,	DATA	SET	MD1D-11
5576	T346	TEST OF	LDCDF		INSTR,	DATA	SET	MD1D-11
5596	T347	TEST OF	LDCDF		INSTR,	DATA	SET	MD1D-11
5616	T350	TEST OF	LDCDF		INSTR,	DATA	SET	MD1D-11
5636	T351	TEST OF	LDCDF		INSTR,	DATA	SET	MD1D-11
5657	T352	TEST OF	LDCFD		INSTR,	DATA	SET	MD1D-11
5677	T353	TEST OF	LDCFD		INSTR,	DATA	SET	MD1D-11
5697	T354	TEST OF	LDCFD		INSTR,	DATA	SET	MD1D-11
5717	T355	TEST OF	LDCFD		INSTR,	DATA	SET	MD1D-11
5737	T356	TEST OF	LDCFD		INSTR,	DATA	SET	MD1D-11
5757	T357	TEST OF	LDCFD		INSTR,	DATA	SET	MD1D-11
5777	T360	TEST OF	LDCFD		INSTR,	DATA	SET	MD1D-11
5797	T361	TEST OF	LDCFD		INSTR,	DATA	SET	MD1D-11
5818	T362	TEST OF	STCDF		INSTR,	DATA	SET	MD1D-11
5838	T363	TEST OF	STCDF		INSTR,	DATA	SET	MD1D-11
5858	T364	TEST OF	STCDF		INSTR,	DATA	SET	MD1D-11
5878	T365	TEST OF	STCDF		INSTR,	DATA	SET	MD1D-11
5898	T366	TEST OF	STCDF		INSTR,	DATA	SET	MD1D-11
5918	T367	TEST OF	STCDF		INSTR,	DATA	SET	MD1D-11
5938	T370	TEST OF	STCDF		INSTR,	DATA	SET	MD1D-11
5958	T371	TEST OF	STCDF		INSTR,	DATA	SET	MD1D-11
5978	T372	TEST OF	STCDF		INSTR,	DATA	SET	MD1D-11
5998	T373	TEST OF	STCDF		INSTR,	DATA	SET	MD1D-11
6018	T374	TEST OF	STCDF		INSTR,	DATA	SET	MD1D-11
6038	T375	TEST OF	STCDF		INSTR,	DATA	SET	MD1D-11
6059	T376	TEST OF	STCFD		INSTR,	DATA	SET	MD1D-11
6079	T377	TEST OF	STCFD		INSTR,	DATA	SET	MD1D-11
6099	T400	TEST OF	STCFD		INSTR,	DATA	SET	MD1D-11
6119	T401	TEST OF	STCFD		INSTR,	DATA	SET	MD1D-11
6139	T402	TEST OF	STCFD		INSTR,	DATA	SET	MD1D-11
6159	T403	TEST OF	STCFD		INSTR,	DATA	SET	MD1D-11
6179	T404	TEST OF	LDCIF		INSTR,	DATA	SET	MD1D-11
6197	T405	TEST OF	LDCIF		INSTR,	DATA	SET	MD1D-11
6215	T406	TEST OF	LDCIF		INSTR,	DATA	SET	MD1D-11
6233	T407	TEST OF	LDCIF		INSTR,	DATA	SET	MD1D-11
6251	T410	TEST OF	LDCIF		INSTR,	DATA	SET	MD1D-11
6270	T411	TEST OF	LDCID		INSTR,	DATA	SET	MD1D-11
6289	T412	TEST OF	LDCID		INSTR,	DATA	SET	MD1D-11
6308	T413	TEST OF	LDCID		INSTR,	DATA	SET	MD1D-11
6327	T414	TEST OF	LDCID		INSTR,	DATA	SET	MD1D-11

6346	T415	TEST OF	LDCID	INSTR,	DATA	SET	LDCID-5
6366	T416	TEST OF	LDCLF	INSTR,	DATA	SET	LDCLF-1
6384	T417	TEST OF	LDCLF	INSTR,	DATA	SET	LDCLF-1
6402	T420	TEST OF	LDCLF	INSTR,	DATA	SET	LDCLF-1
6420	T421	TEST OF	LDCLF	INSTR,	DATA	SET	LDCLF-1
6438	T422	TEST OF	LDCLF	INSTR,	DATA	SET	LDCLF-1
6456	T423	TEST OF	LDCLF	INSTR,	DATA	SET	LDCLF-1
6474	T424	TEST OF	LDCLF	INSTR,	DATA	SET	LDCLF-1
6493	T425	TEST OF	LDCLD	INSTR,	DATA	SET	LDCLD-1
6512	T426	TEST OF	LDCLD	INSTR,	DATA	SET	LDCLD-1
6531	T427	TEST OF	LDCLD	INSTR,	DATA	SET	LDCLD-1
6550	T430	TEST OF	LDCLD	INSTR,	DATA	SET	LDCLD-1
6569	T431	TEST OF	LDCLD	INSTR,	DATA	SET	LDCLD-1
6589	T432	TEST OF	STCFI	INSTR,	DATA	SET	STCFI-1
6608	T433	TEST OF	STCFI	INSTR,	DATA	SET	STCFI-1
6627	T434	TEST OF	STCFI	INSTR,	DATA	SET	STCFI-1
6646	T435	TEST OF	STCFI	INSTR,	DATA	SET	STCFI-1
6665	T436	TEST OF	STCFI	INSTR,	DATA	SET	STCFI-1
6684	T437	TEST OF	STCFI	INSTR,	DATA	SET	STCFI-1
6703	T440	TEST OF	STCFI	INSTR,	DATA	SET	STCFI-1
6723	T441	TEST OF	STCOI	INSTR,	DATA	SET	STCOI-1
6743	T442	TEST OF	STCOI	INSTR,	DATA	SET	STCOI-1
6763	T443	TEST OF	STCOI	INSTR,	DATA	SET	STCOI-1
6783	T444	TEST OF	STCOI	INSTR,	DATA	SET	STCOI-1
6803	T445	TEST OF	STCOI	INSTR,	DATA	SET	STCOI-1
6823	T446	TEST OF	STCOI	INSTR,	DATA	SET	STCOI-1
6843	T447	TEST OF	STCOI	INSTR,	DATA	SET	STCOI-1
6864	T450	TEST OF	STCFL	INSTR,	DATA	SET	STCFL-1
6883	T451	TEST OF	STCFL	INSTR,	DATA	SET	STCFL-1
6902	T452	TEST OF	STCFL	INSTR,	DATA	SET	STCFL-1
6921	T453	TEST OF	STCFL	INSTR,	DATA	SET	STCFL-1
6940	T454	TEST OF	STCFL	INSTR,	DATA	SET	STCFL-1
6959	T455	TEST OF	STCFL	INSTR,	DATA	SET	STCFL-1
6978	T456	TEST OF	STCFL	INSTR,	DATA	SET	STCFL-1
6998	T457	TEST OF	STCOL	INSTR,	DATA	SET	STCOL-1
7018	T460	TEST OF	STCOL	INSTR,	DATA	SET	STCOL-1
7038	T461	TEST OF	STCOL	INSTR,	DATA	SET	STCOL-1
7058	T462	TEST OF	STCOL	INSTR,	DATA	SET	STCOL-1
7078	T463	TEST OF	STCOL	INSTR,	DATA	SET	STCOL-1
7098	T464	TEST OF	STCOL	INSTR,	DATA	SET	STCOL-1
7118	T465	TEST OF	STCOL	INSTR,	DATA	SET	STCOL-1
7139	T466	TEST OF	LDEXP/F	INSTR,	DATA	SET	LDEXP-1
7159	T467	TEST OF	LDEXP/F	INSTR,	DATA	SET	LDEXP-1
7179	T470	TEST OF	LDEXP/F	INSTR,	DATA	SET	LDEXP-1
7199	T471	TEST OF	LDEXP/F	INSTR,	DATA	SET	LDEXP-1
7219	T472	TEST OF	LDEXP/F	INSTR,	DATA	SET	LDEXP-1
7239	T473	TEST OF	LDEXP/F	INSTR,	DATA	SET	LDEXP-1
7259	T474	TEST OF	LDEXP/F	INSTR,	DATA	SET	LDEXP-1
7279	T475	TEST OF	LDEXP/F	INSTR,	DATA	SET	LDEXP-1
7299	T476	TEST OF	LDEXP/F	INSTR,	DATA	SET	LDEXP-1
7319	T477	TEST OF	LDEXP/F	INSTR,	DATA	SET	LDEXP-1
7339	T500	TEST OF	LDEXP/F	INSTR,	DATA	SET	LDEXP-1
7359	T501	TEST OF	LDEXP/F	INSTR,	DATA	SET	LDEXP-1
7379	T502	TEST OF	LDEXP/F	INSTR,	DATA	SET	LDEXP-1
7399	T503	TEST OF	LDEXP/F	INSTR,	DATA	SET	LDEXP-1
7419	T504	TEST OF	LDEXP/F	INSTR,	DATA	SET	LDEXP-1

7439	T505	TEST OF LDEXP/F	INSTR, DATA SET	LEXF-20
7459	T506	TEST OF LDEXP/F	INSTR, DATA SET	LEXF-21
7479	T507	TEST OF LDEXP/F	INSTR, DATA SET	LEXF-22
7499	T510	TEST OF LDEXP/F	INSTR, DATA SET	LEXF-23
7519	T511	TEST OF LDEXP/F	INSTR, DATA SET	LEXF-24
7540	T512	TEST OF LDEXP/D	INSTR, DATA SET	LEXD-1
7562	T513	TEST OF LDEXP/D	INSTR, DATA SET	LEXD-2
7584	T514	TEST OF LDEXP/D	INSTR, DATA SET	LEXD-3
7606	T515	TEST OF LDEXP/D	INSTR, DATA SET	LEXD-4
7628	T516	TEST OF LDEXP/D	INSTR, DATA SET	LEXD-5
7650	T517	TEST OF LDEXP/D	INSTR, DATA SET	LEXD-6
7672	T520	TEST OF LDEXP/D	INSTR, DATA SET	LEXD-7
7694	T521	TEST OF LDEXP/D	INSTR, DATA SET	LEXD-10
7716	T522	TEST OF LDEXP/D	INSTR, DATA SET	LEXD-11
7738	T523	TEST OF LDEXP/D	INSTR, DATA SET	LEXD-12
7760	T524	TEST OF LDEXP/D	INSTR, DATA SET	LEXD-13
7782	T525	TEST OF LDEXP/D	INSTR, DATA SET	LEXD-14
7804	T526	TEST OF LDEXP/D	INSTR, DATA SET	LEXD-15
7826	T527	TEST OF LDEXP/D	INSTR, DATA SET	LEXD-16
7848	T530	TEST OF LDEXP/D	INSTR, DATA SET	LEXD-17
7870	T531	TEST OF LDEXP/D	INSTR, DATA SET	LEXD-20
7892	T532	TEST OF LDEXP/D	INSTR, DATA SET	LEXD-21
7914	T533	TEST OF LDEXP/D	INSTR, DATA SET	LEXD-22
7936	T534	TEST OF LDEXP/D	INSTR, DATA SET	LEXD-23
7958	T535	TEST OF LDEXP/D	INSTR, DATA SET	LEXD-24
7981	T536	TEST OF STEXP/F	INSTR, DATA SET	SEXF-1
7999	T537	TEST OF STEXP/F	INSTR, DATA SET	SEXF-2
8017	T540	TEST OF STEXP/F	INSTR, DATA SET	SEXF-3
8035	T541	TEST OF STEXP/F	INSTR, DATA SET	SEXF-4
8053	T542	TEST OF STEXP/F	INSTR, DATA SET	SEXF-5
8071	T543	TEST OF STEXP/F	INSTR, DATA SET	SEXF-6
8089	T544	TEST OF STEXP/F	INSTR, DATA SET	SEXF-7
8108	T545	TEST OF STEXP/D	INSTR, DATA SET	SEXD-1
8127	T546	TEST OF STEXP/D	INSTR, DATA SET	SEXD-2
8146	T547	TEST OF STEXP/D	INSTR, DATA SET	SEXD-3
8165	T550	TEST OF STEXP/D	INSTR, DATA SET	SEXD-4
8184	T551	TEST OF STEXP/D	INSTR, DATA SET	SEXD-5
8203	T552	TEST OF STEXP/D	INSTR, DATA SET	SEXD-6
8222	T553	TEST OF STEXP/D	INSTR, DATA SET	SEXD-7
8243	SUB PASS END CONTROL			
8284	END OF PASS ROUTINE (MODIFIED SYSMAC)			
8320	SUBR TO TEST THE	CHPF INSTRUCTION		
8364	SUBR TO TEST THE	CHPD INSTRUCTION		
8410	SUBR TO TEST THE	ADDF INSTRUCTION		
8453	SUBR TO TEST THE	ADDD INSTRUCTION		
8499	SUBR TO TEST THE	SUBF INSTRUCTION		
8542	SUBR TO TEST THE	SUBD INSTRUCTION		
8588	SUBR TO TEST THE	MULF INSTRUCTION		
8631	SUBR TO TEST THE	MULD INSTRUCTION		
8677	SUBR TO TEST THE	DIVF INSTRUCTION		
8720	SUBR TO TEST THE	DIVD INSTRUCTION		
8767	SUBR TO TEST THE	MODF INSTRUCTION,	USING 2 ACCUMULATORS	
8823	SUBR TO TEST THE	MOOD INSTRUCTION,	USING 2 ACCUMULATORS	
8884	SUBR TO TEST THE	MODF INSTRUCTION,	USING 1 ACCUMULATOR	
8940	SUBR TO TEST THE	MOOD INSTRUCTION,	USING 1 ACCUMULATOR	
9001	SUBR TO TEST THE	LDCDF INSTRUCTION		

9045	SUBR TO TEST THE LDCFD INSTRUCTION
9091	SUBR TO TEST THE STCDF INSTRUCTION
9134	SUBR TO TEST THE STCFD INSTRUCTION
9169	SUBR TO TEST THE LDCIF INSTRUCTION
9202	SUBR TO TEST THE LDCID INSTRUCTION
9239	SUBR TO TEST THE LDCLF INSTRUCTION
9272	SUBR TO TEST THE LDCLD INSTRUCTION
9306	SUBR TO TEST THE STCFI INSTRUCTION
9356	SUBR TO TEST THE STCDI INSTRUCTION
9406	SUBR TO TEST THE STCFI INSTRUCTION
9458	SUBR TO TEST THE STCDI INSTRUCTION
9507	SUBR TO TEST THE LDEXP INSTRUCTION, F MODE
9550	SUBR TO TEST THE LDEXP INSTRUCTION, D MODE
9596	SUBR TO TEST THE STEXP INSTRUCTION, F MODE
9636	SUBR TO TEST THE STEXP INSTRUCTION, D MODE
9674	FPP UNEXPECTED TRAP CATCHER
9685	SCOPE HANDLER ROUTINE
9749	ERROR HANDLER ROUTINE
9812	ERROR MESSAGE TIMEOUT ROUTINE (MODIFIED SYSMAC)
9878	TYPE ROUTINE
9957	APT COMMUNICATIONS ROUTINE
10014	BINARY TO OCTAL (ASCII) AND TYPE
10091	TRAP DECODER
10114	TRAP TABLE
10128	POWER DOWN AND UP ROUTINES
10175	ERROR MESSAGES, DATA HEADERS, DATA VECTORS, ETC

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

```

.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-DZCAC-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 WFP/WFP ALTERNATELY EACH PASS 1=TEST ONLY UNIT SPECIFIED IN SW<00>
0	000002	0=SELECT WFP, IF SW<01>=1
	000003	1=SELECT WFP, IF SW<01>=1

.SBTTL BASIC DEFINITIONS

```

*INITIAL ADDRESS OF THE STACK POINTER *** 1100 ***
001100 STACK= 1100
.EQUIV EMT,ERROR ;;BASIC DEFINITION OF ERROR CALL
.EQUIV IOT,SCOPE ;;BASIC DEFINITION OF SCOPE CALL

```

*MISCELLANEOUS DEFINITIONS

```

000011 HT= 11 ;;CODE FOR HORIZONTAL TAB
000012 LF= 12 ;;CODE FOR LINE FEED
000015 CR= 15 ;;CODE FOR CARRIAGE RETURN
000200 CRLF= 200 ;;CODE FOR CARRIAGE RETURN-LINE FEED
177776 PS= 177776 ;;PROCESSOR STATUS WORD
177774 .EQUIV PS,PSW
177772 STKLM= 177774 ;;STACK LIMIT REGISTER
177570 PIR= 177772 ;;PROGRAM INTERRUPT REQUEST REGISTER
177570 DSMR= 177570 ;;HARDWARE SWITCH REGISTER
177570 DDISP= 177570 ;;HARDWARE DISPLAY REGISTER

```

*GENERAL PURPOSE REGISTER DEFINITIONS

```

000000 R0= %0 ;;GENERAL REGISTER
000001 R1= %1 ;;GENERAL REGISTER
000002 R2= %2 ;;GENERAL REGISTER
000003 R3= %3 ;;GENERAL REGISTER
000004 R4= %4 ;;GENERAL REGISTER
000005 R5= %5 ;;GENERAL REGISTER

```



```

57 000006 R6= %6 :: GENERAL REGISTER
58 000007 R7= %7 :: GENERAL REGISTER
59 000006 SP= %6 :: STACK POINTER
60 000007 PC= %7 :: PROGRAM COUNTER
61
62 ::#PRIORITY LEVEL DEFINITIONS
63 000000 PR0= 0 :: PRIORITY LEVEL 0
64 000040 PR1= 40 :: PRIORITY LEVEL 1
65 000100 PR2= 100 :: PRIORITY LEVEL 2
66 000140 PR3= 140 :: PRIORITY LEVEL 3
67 000200 PR4= 200 :: PRIORITY LEVEL 4
68 000240 PR5= 240 :: PRIORITY LEVEL 5
69 000300 PR6= 300 :: PRIORITY LEVEL 6
70 000340 PR7= 340 :: PRIORITY LEVEL 7
71
72 ::#"SWITCH REGISTER" SWITCH DEFINITIONS
73 100000 SW15= 100000
74 040000 SW14= 40000
75 020000 SW13= 20000
76 010000 SW12= 10000
77 004000 SW11= 4000
78 002000 SW10= 2000
79 001000 SW09= 1000
80 000400 SW08= 400
81 000200 SW07= 200
82 000100 SW06= 100
83 000040 SW05= 40
84 000020 SW04= 20
85 000010 SW03= 10
86 000004 SW02= 4
87 000002 SW01= 2
88 000001 SW00= 1
89 .EQUIV SW09, SW9
90 .EQUIV SW08, SW8
91 .EQUIV SW07, SW7
92 .EQUIV SW06, SW6
93 .EQUIV SW05, SW5
94 .EQUIV SW04, SW4
95 .EQUIV SW03, SW3
96 .EQUIV SW02, SW2
97 .EQUIV SW01, SW1
98 .EQUIV SW00, SW0
99
100 ::#DATA BIT DEFINITIONS (BIT00 TO BIT15)
101 100000 BIT15= 100000
102 040000 BIT14= 40000
103 020000 BIT13= 20000
104 010000 BIT12= 10000
105 004000 BIT11= 4000
106 002000 BIT10= 2000
107 001000 BIT09= 1000
108 000400 BIT08= 400
109 000200 BIT07= 200
110 000100 BIT06= 100
111 000040 BIT05= 40
112 000020 BIT04= 20

```

113 000010
114 000004
115 000002
116 000001
117
118
119
120
121
122
123
124
125
126
127
128
129 000004
130 000010
131 000014
132 000014
133 000014
134 000020
135 000024
136 000030
137 000034
138 000060
139 000064
140 000240
141
142
143 076600
144
145 000022
146
147 000144
148 000344
149
150
151 000244
152
153
154 000000
155 000001
156 000002
157 000003
158 000004
159 000005
160
161
162 052525
163 052525
164 125252
165 125252
166 007417
167 170360
168 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 ;: T= BIT
TRTVEC= 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


```

169 177777
170 100000
171 077777
172 177777
173 000200
174 100200
175 000177
176 100177
177 040200
178 140200
179 104210
180 000377
181 177400
182
183
184 147757
185 000000
186 000000
187
188
189 177760
190
191
192
193
194 000000
195
196
197
198 000174
199 000174 000000
200 000176 000000
201
202 000200 000137 002400
203
204
205
206
207
208 000204
209 000046
210 000046 033314
211 000052
212 000052 000000
213 000204
214 001000
215
216
217
218
219
220 001000
221 000024
222 000024 000200
223 000044
224 000044 001000

```

```

MI= 177777 ; 1111...11 MINUS ONE, ALL 1'S
MO= 100000 ; 1000...00 MINUS ZERO
LGP= 077777 ; 0111...11 LGST + NUM (1ST WD FLT)
LGN= 177777 ; 1111...11 LGST - NUM (1ST WD FLT)
SNP= 000200 ; +1*2**128, SMLT + NUM (1ST WD FLT)
SMN= 100200 ; -1*2**128, SMLT - NUM (1ST WD FLT)
ZXIMP= 000177 ; ZERO EXP, ALL 1-S MANT (1ST WD FLT)
ZXIMN= 100177 ; ZERO EXP, ALL 1-S MANT (1ST WD FLT)
FIP= 040200 ; +1.0E+0, 1ST WD FLT
FIN= 140200 ; -1.0E+0, 1ST WD FLT
P13Z= 104210 ; 1000100010001000
LB= 000377 ; 0000000011111111 LOWER BYTE
UB= 177400 ; 1111111100000000 UPPER BYTE

```

```

;#FPS BIT PATTERNS
FPS1= 147757 ; ALL BITS ON (READABLE)
FPS0= 000000 ; ALL BITS OFF
NA= 000000 ; FOR FEC, WHEN NOT APPLICABLE

```

```

;#PSW BIT PATTERNS
CCONLY= 177760 ; FOR BIC TO GET CC BITS ONLY

```

.SBTTL TRAP CATCHER

```

.=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

```

```

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

```

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

```

.SBTTL APT PARAMETER BLOCK

```

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

```

225	001000
226	
227	
228	
229	
230	001000
231	001000 000000
232	001002 001324
233	001004 000001
234	001006 000001
235	001010 000000
236	001012 000014
237	

```

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

SAPTHD:
SHIBTS: .WORD 0 ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
SMBADR: .WORD SMBL ;;ADDRESS OF APT MAILBOX (BITS 0-15)
STSTM: .WORD 1 ;;RUN TIM OF LONGEST TEST
SPASTM: .WORD 1 ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
SUNITH: .WORD 0 ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
      .WORD SETEND-SMIBL/2 ;;LENGTH MAILBOX-ETABLE(WORDS)

```


.SBTTL COMMON TAGS

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

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
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

001100 001100
001100 000000
001102 000000
001104 000000
001106 000000
001110 000000
001112 000000
001114 000000
001116 000000
001120 000001
001122 000000
001124 000000
001126 000000
001130 000000
001132 000000
001134 000000
001136 000000
001140 000
001141 000
001142 000000
001144 177570
001146 177570
001150 000000
001152 177560
001154 177562
001156 177564
001160 177566
001162 000
001163 002
001164 012
001165 000
001166 000000
001170 000000
001172 000000
001174 000000
001176 000000
001200 000000
001202 000000
001204 000000
001206 000000
001210 000000
001212 000000
001214 000000
001216 000000
001220 000000
001222 000000

. =1100
SCMTAG: .WORD 0 ; START OF COMMON TAGS
; ;-----START OF CLEAR COMMON TAGS-----
STSTM: .WORD 0 ; CONTAINS THE TEST NUMBER
SERFLG: .WORD 0 ; CONTAINS ERROR FLAG
SICNT: .WORD 0 ; CONTAINS SUBTEST ITERATION COUNT
SLPADR: .WORD 0 ; CONTAINS SCOPE LOOP ADDRESS
SLPEAR: .WORD 0 ; CONTAINS SCOPE RETURN FOR ERRORS
SERTTL: .WORD 0 ; CONTAINS TOTAL ERRORS DETECTED
SITEMB: .WORD 0 ; CONTAINS ITEM CONTROL BYTE
SERMAX: .WORD 1 ; CONTAINS MAX. ERRORS PER TEST
SERRPC: .WORD 0 ; CONTAINS PC OF LAST ERROR INSTRUCTION
SGDADR: .WORD 0 ; CONTAINS ADDRESS OF 'GOOD' DATA
SBDADR: .WORD 0 ; CONTAINS ADDRESS OF 'BAD' DATA
SGDOAT: .WORD 0 ; CONTAINS 'GOOD' DATA
SBDGAT: .WORD 0 ; CONTAINS 'BAD' DATA
; .WORD 0 ; RESERVED—NOT TO BE USED
SAUTOB: .BYTE 0 ; AUTOMATIC MODE INDICATOR
SINTAG: .BYTE 0 ; INTERRUPT MODE INDICATOR
; .WORD 0 ;
; ;-----END OF CLEAR COMMON TAGS-----
SWR: .WORD DSWR ; ADDRESS OF SWITCH REGISTER
DISPLA: .WORD DDISP ; ADDRESS OF DISPLAY REGISTER
SLPTST: .WORD 0 ; CONTAINS TEST NUMBER TO LOOP UPON
STKS: 177560 ; TTY KBD STATUS
STKB: 177562 ; TTY KBD BUFFER
STPS: 177564 ; TTY PRINTER STATUS REG. ADDRESS
STPB: 177566 ; TTY PRINTER BUFFER REG. ADDRESS
SNLL: .BYTE 0 ; CONTAINS NULL CHARACTER FOR FILLS
SFILLS: .BYTE 2 ; CONTAINS # OF FILLER CHARACTERS REQUIRED
SFILLC: .BYTE 12 ; INSERT FILL CHARS. AFTER A "LINE FEED"
STPFLG: .BYTE 0 ; "TERMINAL AVAILABLE" FLAG (BIT<07>=0=YES)
SREGAD: .WORD 0 ; CONTAINS THE ADDRESS FROM WHICH (SREG0) WAS OBTAINED
SREG0: .WORD 0 ; CONTAINS ((SREGAD)+0)
SREG1: .WORD 0 ; CONTAINS ((SREGAD)+2)
SREG2: .WORD 0 ; CONTAINS ((SREGAD)+4)
SREG3: .WORD 0 ; CONTAINS ((SREGAD)+6)
SREG4: .WORD 0 ; CONTAINS ((SREGAD)+10)
SREG5: .WORD 0 ; CONTAINS ((SREGAD)+12)
SREG6: .WORD 0 ; CONTAINS ((SREGAD)+14)
SREG7: .WORD 0 ; CONTAINS ((SREGAD)+16)
SREG10: .WORD 0 ; CONTAINS ((SREGAD)+20)
SREG11: .WORD 0 ; CONTAINS ((SREGAD)+22)
SREG12: .WORD 0 ; CONTAINS ((SREGAD)+24)
SREG13: .WORD 0 ; CONTAINS ((SREGAD)+26)
SREG14: .WORD 0 ; CONTAINS ((SREGAD)+30)
SREG15: .WORD 0 ; CONTAINS ((SREGAD)+32)

E03

FPU ADVANCED INSTR TESTS
DQFPBB.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 9
APT MAILBOX-ETABLE

SEQ 0176

350
351
352 001354
353

.*
.*
\$ETEND:
.MEXIT

BIT 9=FLOATING POINT PROCESSOR
BIT 8=MEMORY MANAGEMENT

ERROR POINTER TABLE

.SBTTL ERROR POINTER TABLE

SERRTB:

;;THIS TABLE CONTAINS THE INFORMATION FOR EACH ERROR THAT CAN OCCUR.
;;THE INFORMATION IS OBTAINED BY USING THE INDEX NUMBER FOUND IN
;;LOCATION SITE#B, THIS NUMBER INDICATES WHICH ITEM IN THE TABLE IS PERTINENT.
;;NOTE1: IF SITE#B IS 0 THE ONLY PERTINENT DATA IS (SERVPC).
;;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 (SERRTB) HAS BEEN MODIFIED,
ELIMINATING UNUSED VALUE FOR DATA FORMAT POINTER.
ERROR TYPING ROUTINE HAS ALSO BEEN MODIFIED
ACCORDINGLY.

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

373	001354	044147	045366	045642	EMV001:	.WORD	EMA,DHA,DTA	LDClF,STCFI,STEXP/F,TRAP-TSTR
374	001362	044147	045366	045650	EMV002:	.WORD	EMA,DHA,DTB	CHPF,LDCLF,STCFL
375	001370	044147	045366	045656	EMV003:	.WORD	EMA,DHA,DTC	LDCID,STCDI,LDEXP/F,STEXP/D
376	001376	044147	045366	045664	EMV004:	.WORD	EMA,DHA,DTD	ADDF,SUBF,MULF,DIVF,LDCDF,LCCFD
377								STCDF,LDCLD,STCDL
378	001404	044147	045366	045672	EMV005:	.WORD	EMA,DHA,DTE	CMPO,MOOF,STCFD
379	001412	044147	045366	045700	EMV006:	.WORD	EMA,DHA,DTF	LDEXP/D
380	001420	044147	045366	045706	EMV007:	.WORD	EMA,DHA,DTG	ADDO,SUBO,MULD,DIVD
381	001426	044147	045366	045714	EMV010:	.WORD	EMA,DHA,DTH	MOOD

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

383	001434	044173	045402	045722	EMV011:	.WORD	EMB,DHB,DTI	STCFI,TRAP-TSTR
384	001442	044173	045402	045734	EMV012:	.WORD	EMB,DHB,DTJ	CHPF,STCFL
385	001450	044173	045402	045746	EMV013:	.WORD	EMB,DHB,DTK	STCDI,LDEXP/F
386	001456	044173	045402	045760	EMV014:	.WORD	EMB,DHB,DTL	ADDF,SUBF,MULF,DIVF,LCCDF
387								LCCFD,STCDF,STCDL
388	001464	044173	045402	045772	EMV015:	.WORD	EMB,DHB,DTM	CMPO,MOOF
389	001472	044173	045402	046004	EMV016:	.WORD	EMB,DHB,DTN	LDEXP/D
390	001500	044173	045402	046016	EMV017:	.WORD	EMB,DHB,DTO	ADDO,SUBO,MULD,DIVD
391	001506	044173	045402	046030	EMV020:	.WORD	EMB,DHB,DTP	MOOD

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

393	001514	044357	045442	046056	EMV021:	.WORD	EME,DHC,DTS	CHPF
394	001522	044357	045500	046140	EMV022:	.WORD	EME,DHC,DTX	CMPO
395	001530	044446	045442	046114	EMV023:	.WORD	EMF,DHC,DTV	ADDF,SUBF
396	001536	044446	045500	046250	EMV024:	.WORD	EMF,DHC,DTAB	ADDO,SUBO
397	001544	044520	045442	046114	EMV025:	.WORD	EMG,DHC,DTV	MULF,DIVF
398	001552	044520	045500	046250	EMV026:	.WORD	EMG,DHC,DTAB	MULD,DIVD
399	001560	044572	045442	046114	EMV027:	.WORD	EMH,DHC,DTV	MOOF-FRAC
400	001566	044653	045442	046126	EMV030:	.WORD	EMI,DHC,DTW	MOOF-INT
401	001574	044572	045500	046250	EMV031:	.WORD	EMH,DHC,DTAB	MOOD-FRAC
402	001602	044653	045500	046272	EMV032:	.WORD	EMI,DHC,DTAC	MOOD-INT
403	001610	045007	045442	046114	EMV033:	.WORD	EMK,DHC,DTV	LDCIF,STCDF
404	001616	044731	045500	046204	EMV034:	.WORD	EMJ,DHC,DTZ	LDCID
405	001624	044731	045500	046226	EMV035:	.WORD	EMJ,DHC,DTAA	STCFD
406	001632	045065	045442	046070	EMV036:	.WORD	EML,DHC,DTT	LDCIF
407	001640	045065	045500	046162	EMV037:	.WORD	EML,DHC,DTY	LDCID
408	001646	045065	045442	046102	EMV040:	.WORD	EML,DHC,DTU	LDCLF
409	001654	045065	045500	046204	EMV041:	.WORD	EML,DHC,DTZ	LDCLD

410	001662	045145	045366	046042	EMV042: .WORD	EMH,DHA,DTQ	:	STCFI
411	001670	045145	045366	046050	EMV043: .WORD	EMH,DHA,DTR	:	STCDI
412	001676	045145	045442	046102	EMV044: .WORD	EMH,DHC,DTU	:	STCFL
413	001704	045145	045442	046114	EMV045: .WORD	EMH,DHC,DTV	:	STCDL
414	001712	045223	045442	046102	EMV046: .WORD	EMH,DHC,DTU	:	LDEXP/F
415	001720	045223	045500	046226	EMV047: .WORD	EMH,DMD,DTAA	:	LDEXP/D
416	001726	045305	045366	046042	EMV050: .WORD	EMO,DHA,DTQ	:	STEXP/F
417	001734	045305	045366	046050	EMV051: .WORD	EMO,DHA,DTR	:	STEXP/D
418	001742	000000	000000	000000	EMV052: .WORD	0,0,0	:	(UNUSED)
419	001750	000000	000000	000000	EMV053: .WORD	0,0,0	:	(UNUSED)
420						*****		VECTORS FOR CC COPY ERRORS *****
421	001756	044310	045366	046314	EMV054: .WORD	EMD,DHA,DTAD	:	STCFI,STCDI,STEXP/F,STEXP/D
422	001764	044310	045366	046322	EMV055: .WORD	EMD,DHA,DTAE	:	STCFL,STCDL
423						*****		VECTOR FOR ILLEGAL TRAP CATCHER ROUTINE *****
424	001772	044223	045576	046330	EMV056: .WORD	EMC,DHF,DTAK	:	UNEXPECTED TRAP

```

425
426
427 002000 000000
428 002002 000000
429 002004 000000
430 002006 000000
431 002010 000000
432 002012 000000
433 002014 000000
434
435
436 002016 000000
437 002020 000000
438 002022 000000
439 002024 000000
440 002026 000000
441 002030 000000
442 002032 000000
443 002034 000000
444
445
446 002036 052525 177777 125252
447 002044 000000
448
449
450
451 002046 005015 005012 042115
452 002054 030455 026461 050504
453 002062 050106 026502
454 002066 027102
455 002070 027056
456 002072 042120 026520 030461
457 002100 033057 020130 027106
458 002106 027120 027125 040440
459 002114 053104 047101 042503
460 002122 020104 047111 052123
461 002130 052522 052103 047511
462 002136 020116 042524 052123
463 002144 006523 000012
464 002150 005015 040520 051523
465 002156 021440 000

```

.SBTTL PROGRAM DEFINED COMMON TAGS

:#VARIABLES

```

FPS: .WORD 0 ; FPS STORED HERE AFTER STFPS
FEC: .WORD 00 ; FEC STORED HERE AFTER STST
FEA: .WORD 000 ; FEA STORED HERE AFTER STST
FPPOPC: .WORD 0000 ; OLD PC SAVED HERE AFTER TRAP AFTER TRAP
FPPOPS: .WORD 0000 ; OLD PS SAVED HERE
FPPOSP: .WORD 00 ; SP AFTER TRAP
EXPFEA: .WORD 0 ; EXPECTED FEA

```

:#REGISTER CONTENTS, AT ERROR, FOR DISPLAY

```

EREG0: .WORD 0
EREG1: .WORD 00
EREG2: .WORD 000
EREG3: .WORD 0000
EREG4: .WORD 00000
EREG5: .WORD 000000
EREG6: .WORD 0000000
EREG7: .WORD 00000000

```

:#CONSTANTS

```

PREVAC: .WORD ALTP,M1,ALTN,0 ; PREV CONTENTS OF FLOAT AC

```

:#MESSAGES FOR BEGIN PROGRAM/START OF PASS/ETC

```

BGNMES: .ASCII <CR><LF><LF><LF>"MD-11-DQFPB-"
          .ASCII "B."
          .ASCII " "
          .ASCIZ "PDP-11/6X F.P.U. ADVANCED INSTRUCTION TESTS"<CR><LF>
NMPAS1: .ASCIZ <CR><LF>"PASS #"
```



```

466
467
468
469
470
471
472
473 002400
474
475
476 002400 012706 001100
477 002404 005026
478 002406 022706 001144
479 002412 001374
480 002414 012706 001100
481
482 002420 012737 041762 000020
483 002426 012737 000340 000022
484 002434 012737 042240 000030
485 002442 012737 000340 000032
486 002450 012737 043674 000034
487 002456 012737 000340 000036
488 002464 012737 043742 000024
489 002472 012737 000340 000026
490 002500 013737 033264 033256
491 002506 005037 001310
492 002512 005037 001312
493 002516 012737 000001 001120
494 002524 012737 002524 001110
495 002532 012737 002532 001112
496
497
498 002540 013746 000004
499 002544 012737 002600 000004
500 002552 012737 177570 001144
501 002560 012737 177570 001146
502 002566 022777 177777 176350
503 002574 001012
504
505 002576 000403
506 002600 012716 002606 64S:
507 002604 000002
508 002606 012737 000176 001144 65S:
509 002614 012737 000174 001146
510 002622 012637 000004 66S:
511
512 002626 005037 001332
513 002632 132737 000200 001345
514 002640 001403
515 002642 012737 001346 001144
516 002650
517
518
519 002650 012737 041722 000244
520 002656 005037 000246
521

```

```

.SBTTL START OF PASS ROUTINE

;;*****
;.ENABL AMA ; ASSEMBLE ALL RELATIVE REFERENCES AS ABSOLUTE
;;*****

START:
.SBTTL INITIALIZE THE COMMON TAGS
;;CLEAR THE COMMON TAGS ($CHTAG) AREA
MOV $CHTAG,R6 ;:FIRST LOCATION TO BE CLEARED
CLR (R6)+ ;:CLEAR MEMORY LOCATION
CMP $SMR,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 $SEPAR,$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 $SPWRN,$PWAVEC ;:POWER FAILURE VECTOR
MOV $340,$PWAVEC+2 ;:LEVEL 7
MOV $ENDCT,$EOPCT ;:SETUP END-OF-PROGRAM COUNTER
CLR $TIMES ;:INITIALIZE NUMBER OF ITERATIONS
CLR $ESCAPE ;:CLEAR THE ESCAPE ON ERROR ADDRESS
MOV $1,$SERMAX ;:ALLOW ONE ERROR PER TEST
MOV $,$SLPADR ;:INITIALIZE THE LOOP ADDRESS FOR SCOPE
MOV $,$SLPEAR ;: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 $64S,$ERRVEC ;:SET UP ERROR VECTOR
MOV $DSMR,$SMR ;:SETUP FOR A HARDWARE SWICH REGISTER
MOV $DDISP,$DISPLAY ;:AND A HARDWARE DISPLAY REGISTER
CMP $-1,$SMR ;:TRY TO REFERENCE HARDWARE SMR
BNE 66S ;:BRANCH IF NO TIMEOUT TRAP OCCURRED
;:AND THE HARDWARE SMR IS NOT = -1
BR 65S ;:BRANCH IF NO TIMEOUT
MOV $64S,(SP) ;:SET UP FOR TRAP RETURN
RTI
MOV $SWREG,$SMR ;:POINT TO SOFTWARE SMR
MOV $DISPREG,$DISPLAY
MOV (SP)+,$ERRVEC ;:RESTORE ERROR VECTOR
CLR $PASS ;:CLEAR PASS COUNT
BITB $APTSIZE,$ENVM ;:TEST USER SIZE UNDER APT
BEQ 67S ;:YES,USE NON-APT SWITCH
MOV $SSWREG,$SMR ;:NO,USE APT SWITCH REGISTER
67S:
; SET UP FPP UNEXPECTED TRAP CATCHER - - - - -
MOV $FPPILT,$FPPVEC ;:NEW PC AT FPP TRAP
CLR $FPPVEC+2 ;:NEW PS AT FPP TRAP

```

```

522 002662 104401 002046          TYPE      ,BGNMES          ; ID MESSAGE AT START
;////////////////////////////////////
; MESSAGE ON WHETHER OR NOT HFP UNIT IS PRESENT
;
527 002666 076600 000022          MED        ,RWHAMI          ;WHAMI INTO RO
528 002672 032700 000020          BIT        ,BIT04,RO      ; IS THERE A HFP UNIT ?
529 002676 001403          BEQ        ,70S          ; NO, BR
530 002700 104401 002714          TYPE      ,68S          ; INDICATE FP11-E PRESENT
531 002704 000453          BR        ,NWPAS        ; GO FOR SUBPASS INIT
532 002706 104401 002754          70S:     TYPE      ,69S          ; INDICATE NO FP11-E
533 002712 000450          BR        ,NWPAS        ; GO FOR SUBPASS INIT
535 002714 005015 020052 050106 68S:     .ASCIZ  <15><12>* FP11-E HFP UNIT PRESENT * <15><12>
536 002722 030461 042455 044040
537 002730 050106 052440 044516
538 002736 020124 051120 051505
539 002744 047105 020124 006452
540 002752 000012
541 002754 005015 020052 047516 69S:     .ASCIZ  <15><12>* NO FP11-E HFP UNIT - ALL TESTS WFP ONLY * <15><12>
542 002762 043040 030520 026461
543 002770 020105 043110 020120
544 002776 047125 052111 026440
545 003004 040440 046114 052040
546 003012 051505 051524 053440
547 003020 050106 047440 046116
548 003026 020131 006452 000012
;
; .EVEN
;////////////////////////////////////
;*****
;:NEW PASS ENTERS HERE
;*****
558 003034 012706 001100          NWPAS:   MOV        ,STACK,SP          ;RESET STACK PTR
559
560 003040 032777 010000 176076          BIT        ,BIT12,JSWR      ;INHIBIT STATUS TYPEOUTS ?
561 003046 001011          BNE      ,SUBPAS        ;BR IF YES
562
563 003050 104401 002150          TYPE      ,NWPAS1        ;"PASS #"
564 003054 013746 001332          MOV        ,PASS,-(SP)      ;PASS COUNT INTO ...
565 003060 005216          INC        ,(SP)          ; 1-N RANGE
566 003062 104403          TYP0S     ;TYPE OCTAL
567 003064 006 000          BYTE     ,6,0          ; 6 DIGITS, NO LEADING ZEROS
568 003066 104401 001321          TYPE      ,SCLF          ;END THE LINE
569
570
571
572 ;*****
573 ;:NEW SUBPASS ENTERS HERE
574 ;*****
575 003072 012706 001100          SUBPAS:  MOV        ,STACK,SP          ;RESET SP FOR INSURANCE
576
577 003076 076600 000022          MED        ,RWHAMI          ;GET WHAMI INTO RO

```



```

600
601
602
603
604
605 003176 000004
606 003200 012705 003212
607 003204 004737 033330
608
609 003210 000407
610
611 003212
612 003212 000000 000000
613 003216 000000 000000
614 003222 047453 047444
615 003226 000000
616
617
618
619
620
621
622
623 003230 000004
624 003232 012705 003244
625 003236 004737 033330
626
627 003242 000407
628
629 003244
630 003244 052525 052525
631 003250 052525 052525
632 003254 047513 047504
633 003260 000000
634
635
636
637
638
639
640
641 003262 000004
642 003264 012705 003276
643 003270 004737 033330
644
645 003274 000407
646
647 003276
648 003276 077777 177777
649 003302 177777 177777
650 003306 047507 047510
651 003312 000000
652
653
654
655

```

```

*****
;TEST 1 TEST OF CMPF INSTR, DATA SET CMPF-1
; ALL INTERRUPT ENABLES ON
; SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST1: 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
*****
TST2: 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
*****
TST3: 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

```


M03

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 17
T4 TEST OF CMPF INSTR, DATA SET CMPF-4

SEQ 0184

656			
657			
658			
659	003314	000004	
660	003316	012705	003330
661	003322	004737	033330
662			
663	003326	000407	
664			
665	003330		
666	003330	125252	125252
667	003334	125252	125252
668	003340	047453	047444
669	003344	000000	
670			
671			
672			
673			
674			
675			
676			
677	003346	000004	
678	003350	012705	003362
679	003354	004737	033330
680			
681	003360	000407	
682			
683	003362		
684	003362	177777	177777
685	003366	077777	177777
686	003372	047457	047440
687	003376	000000	
688			
689			
690			
691			
692			
693			
694			
695	003400	000004	
696	003402	012705	003414
697	003406	004737	033330
698			
699	003412	000407	
700			
701	003414		
702	003414	037777	177777
703	003420	040000	000000
704	003424	047517	047500
705	003430	000000	
706			
707			
708			
709			
710			
711			

```

; * ALL INTERRUPT ENABLES ON
; * SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
; * *****
TST4: SCOPE
MOV #CMPF4_RS ; 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
; * *****
TST5: SCOPE
MOV #CMPF5_RS ; 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
; * *****
TST6: SCOPE
MOV #CMPF6_RS ; 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

```

```

712
713 003432 000004
714 003434 012705 003446
715 003440 004737 033330
716
717 003444 000407
718
719 003446
720 003446 050000 000001
721 003452 050000 000000
722 003456 047547 047550
723 003462 000000
724
725
726
727
728
729
730
731 003464 000004
732 003466 012705 003500
733 003472 004737 033330
734
735 003476 000407
736
737 003500
738 003500 126000 000000
739 003504 124000 000000
740 003510 047417 047400
741 003514 000000
742
743
744
745
746
747
748
749 003516 000004
750 003520 012705 003532
751 003524 004737 033330
752
753 003530 000407
754
755 003532
756 003532 007417 007417
757 003536 100000 000000
758 003542 047443 147443
759 003546 100014
760
761
762
763
764
765
766
767 003550 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    MD,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

```


768 003552 012705 003564
 769 003556 004737 033330
 770
 771 003562 000407
 772
 773 003564
 774 003564 006177 177777
 775 003570 004177 177777
 776 003574 047507 047510
 777 003600 000000
 778
 779
 780
 781
 782
 783
 784
 785 003602 000004
 786 003604 012705 003616
 787 003610 004737 033330
 788
 789 003614 000407
 790
 791 003616
 792 003616 125252 125252
 793 003622 100177 177777
 794 003626 043557 043540
 795 003632 000000
 796
 797
 798
 799
 800
 801
 802
 803 003634 000004
 804 003636 012705 003650
 805 003642 004737 033330
 806
 807 003646 000407
 808
 809 003650
 810 003650 000377 177777
 811 003654 000377 177776
 812 003660 047407 047410
 813 003664 000000
 814
 815
 816

```

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 )
  
```

```

817 .....
818 : *TEST 15 TEST OF CMPD INSTR, DATA SET CMPD-1
819 : * ALL INTERRUPT ENABLES ON
820 : * LONG FLOAT, LONG INTEGER, ROUND MODES
821 : .....
822 TST15: SCOPE
823 MOV #CMPD1,R5 ; PTR TO TEST DATA SET
824 JSR PC,@CMPDT ; GO TEST
825
826 BR TST16 ;;
827
828 CMPD1: ; TEST DATA SET CMPD-1:
829 .WORD 0,0,0,0 ; INITIAL AC FLOAT NUMBER
830 .WORD 0,0,0,0 ; INITIAL MEM FLOAT NUMBER
831 .WORD 0,0,0,0 ; INITIAL MEM FLOAT NUMBER
832 .WORD 047713,047704 ; FPS: BEFORE, AFTER
833 .WORD NA ; FEC AFTER ( 0 = N/A )
834
835 .....
836 : *TEST 16 TEST OF CMPD INSTR, DATA SET CMPD-2
837 : * ALL INTERRUPT ENABLES ON
838 : * LONG FLOAT, LONG INTEGER, ROUND MODES
839 : * .....
840 TST16: SCOPE
841 MOV #CMPD2,R5 ; PTR TO TEST DATA SET
842 JSR PC,@CMPDT ; GO TEST
843
844 BR TST17 ;;
845
846 CMPD2: ; TEST DATA SET CMPD-2:
847 .WORD LGN,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
848 .WORD LGP,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
849 .WORD 047717,047700 ; FPS: BEFORE, AFTER
850 .WORD NA ; FEC AFTER ( 0 = N/A )
851
852 .....
853 : *TEST 17 TEST OF CMPD INSTR, DATA SET CMPD-3
854 : * ALL INTERRUPT ENABLES ON
855 : * LONG FLOAT, LONG INTEGER, ROUND MODES
856 : * .....
857 TST17: SCOPE
858 MOV #CMPD3,R5 ; PTR TO TEST DATA SET
859 JSR PC,@CMPDT ; GO TEST
860
861 BR TST20 ;;
862
863 CMPD3: ; TEST DATA SET CMPD-3:
864 .WORD ALT4N,ALT4N,ALT4N,ALT4N ; INITIAL AC FLOAT NUMBER
865 .WORD ALT4N,ALT4N,ALT4N,ALT4N ; INITIAL MEM FLOAT NUMBER
866
867
868
869
870
871
872

```


873 004026 047713 047704
874 004032 000000

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

875
876
877
878
879
880

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

881
882 004034 000004
883 004036 012705 004050
884 004042 004737 033476

↑ST20: SCOPE
MOV #CMPD4,R5 ; PTR TO TEST DATA SET
JSR PC,@#CMPDT ; GO TEST
BR TST21 ;

885 004046 000413
887

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

888 004050
889 004050 077777 177777 177777
890 004056 177777
891 004060 177777 177777 177777
892 004066 177777
893 004070 047647 047650
894 004074 000000

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

895
896
897
898
899
900
901

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

902 004076 000004
903 004100 012705 004112
904 004104 004737 033476

↑ST21: SCOPE
MOV #CMPD5,R5 ; PTR TO TEST DATA SET
JSR PC,@#CMPDT ; GO TEST
BR TST22 ;

905 004110 000413
907

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

908 004112
909 004112 007417 007417 007417
910 004120 007417
911 004122 007417 007417 007417
912 004130 007417
913 004132 047653 047644
914 004136 000000

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

915
916
917
918
919
920
921

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

922 004140 000004
923 004142 012705 004154
924 004146 004737 033476

↑ST22: SCOPE
MOV #CMPD6,R5 ; PTR TO TEST DATA SET
JSR PC,@#CMPDT ; GO TEST
BR TST23 ;

925 004152 000413
927
928 004154

CMPD6: ; TEST DATA SET CMPD-6:

E04

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 22
T22 TEST OF CMPD INSTR, DATA SET CMPD-6

SEQ 0189

929	004154	125252	125252	125252	.WORD	ALTN,ALTN,ALTN,ALTN	; INITIAL AC FLOAT NUMBER
930	004162	177777	177777	177777	.WORD	ZX1MN,M1,M1,M1	; INITIAL MEM FLOAT NUMBER
931	004164	100177	177777	177777	.WORD		
932	004172	177777			.WORD		
933	004174	047703	147703		.WORD	047703,147703	; FPS: BEFORE, AFTER
934	004200	100014			.WORD	100014	; FEC AFTER (0 = N/A)

```

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

```

941					TST23:	SCOPE	
942	004202	000004			MOV	#CMPD7,R5	; PTR TO TEST DATA SET
943	004204	012705	004216		JSR	PC,@CMPDT	; GO TEST
944	004210	004737	033476				
945					BR	TST24	::

946	004214	000413					
947					CMPD7:	; TEST DATA SET CMPD-7:	
948	004216				.WORD	002177,M1,M1,M1	; INITIAL AC FLOAT NUMBER
949	004216	002177	177777	177777	.WORD	005177,M1,M1,M1	; INITIAL MEM FLOAT NUMBER
950	004224	177777			.WORD	047657,047640	; FPS: BEFORE, AFTER
951	004226	005177	177777	177777	.WORD	NA	; FEC AFTER (0 = N/A)
952	004234	177777					
953	004236	047657	047640				
954	004242	000000					

```

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

```

955	004244	000004			TST24:	SCOPE	
956	004246	012705	004260		MOV	#CMPD10,R5	; PTR TO TEST DATA SET
957	004252	004737	033476		JSR	PC,@CMPDT	; GO TEST
958					BR	TST25	::

959	004260				CMPD10:	; TEST DATA SET CMPD-10:	
960	004260	030000	000000	000000	.WORD	030000,000000,000000,000000	; INITIAL AC FLOAT NUMBER
961	004266	000000			.WORD	027777,M1,M1,M1	; INITIAL MEM FLOAT NUMBER
962	004270	027777	177777	177777	.WORD	047647,047650	; FPS: BEFORE, AFTER
963	004276	177777			.WORD	NA	; FEC AFTER (0 = N/A)
964	004300	047647	047650				
965	004304	000000					

```

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

```

966	004306	000004			TST25:	SCOPE	
967	004310	012705	004322		MOV	#CMPD11,R5	; PTR TO TEST DATA SET
968	004314	004737	033476		JSR	PC,@CMPDT	; GO TEST

F04

FPU ADVANCED INSTR TESTS
DQFPBB.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 23
T25 TEST OF CMPD INSTR, DATA SET CMPD-11

SEQ 0190

```

985
986 004320 000413 BR TST26 ;;
987
988 004322 CMPD11: ; TEST DATA SET CMPD-11:
989 004322 102000 000000 000000 .WORD 102000,000000,000000,000000 ; INITIAL AC FLOAT NUMBER
990 004330 000000 .WORD 102000,000000,000000,000001 ; INITIAL MEM FLOAT NUMBER
991 004332 102000 000000 000000 .WORD 102000,000000,000000,000001 ; INITIAL MEM FLOAT NUMBER
992 004340 000001 .WORD 047607,047610 ; FPS: BEFORE, AFTER
993 004342 047607 047610 .WORD NA ; FEC AFTER ( 0 = N/A )
994 004346 000000

```

```

995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040

```

```

*****
*TEST 26 TEST OF CMPD INSTR, DATA SET CMPD-12
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

TST26: SCOPE
MOV #CMPD12,RS ; PTR TO TEST DATA SET
JSR PC,@#CMPDT ; GO TEST

```

```

1006 004362 000413 BR TST27 ;;
1008 004364 CMPD12: ; TEST DATA SET CMPD-12:
1009 004364 002000 000000 000000 .WORD 002000,000000,000000,000000 ; INITIAL AC FLOAT NUMBER
1010 004372 000000 .WORD 014000,000000,000000,000000 ; INITIAL MEM FLOAT NUMBER
1011 004374 014000 000000 000000 .WORD 047757,047740 ; FPS: BEFORE, AFTER
1012 004402 000000 .WORD NA ; FEC AFTER ( 0 = N/A )
1013 004404 047757 047740
1014 004410 000000

```

```

*****
*TEST 27 TEST OF CMPD INSTR, DATA SET CMPD-13
* -0 INTERRUPT ENABLE OFF, ALL OTHERS ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

TST27: SCOPE
MOV #CMPD13,RS ; PTR TO TEST DATA SET
JSR PC,@#CMPDT ; GO TEST

```

```

1026 004424 000413 BR TST30 ;;
1028 004426 CMPD13: ; TEST DATA SET CMPD-13:
1029 004426 000000 000000 000000 .WORD 0,0,0,0 ; INITIAL AC FLOAT NUMBER
1030 004434 000000 .WORD ZX1MN,M1,0,M1 ; INITIAL MEM FLOAT NUMBER
1031 004436 100177 177777 000000 .WORD 043713,043704 ; FPS: BEFORE, AFTER
1032 004444 177777 .WORD NA ; FEC AFTER ( 0 = N/A )
1033 004446 043713 043704
1034 004452 000000

```

```

*****
*TEST 30 TEST OF CMPD INSTR, DATA SET CMPD-14
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

1041
1042 004454 000004
1043 004456 012705 004470
1044 004462 004737 033476
1045
1046 004466 000413
1047
1048 004470
1049 004470 100777 000000 177777
1050 004476 000001
1051 004500 100777 000000 177777
1052 004506 000000
1053 004510 047657 047640
1054 004514 000000
1055
1056

```

```

*****
TST30: SCOPE
MOV      #CMPD14,RS      ; 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 )

```


H04

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 25
T31 TEST OF ADDF INSTR, DATA SET ADDF-1

SEQ 0192

1057			
1058			
1059			
1060			
1061			
1062	004516	000004	
1063	004520	012705	004532
1064	004524	004737	033664
1065			
1066	004530	000411	
1067			
1068	004532		
1069	004532	000177	177777
1070	004536	000177	177777
1071	004542	000000	000000
1072	004546	047453	047444
1073	004552	000000	
1074			
1075			
1076			
1077			
1078			
1079			
1080			
1081	004554	000004	
1082	004556	012705	004570
1083	004562	004737	033664
1084			
1085	004566	000411	
1086			
1087	004570		
1088	004570	000000	000000
1089	004574	125252	125252
1090	004600	125252	125252
1091	004604	047407	047410
1092	004610	000000	
1093			
1094			
1095			
1096			
1097			
1098			
1099			
1100	004612	000004	
1101	004614	012705	004626
1102	004620	004737	033664
1103			
1104	004624	000411	
1105			
1106	004626		
1107	004626	052525	052525
1108	004632	000000	000000
1109	004636	052525	052525
1110	004642	047557	047540
1111	004646	000000	
1112			

```

*****
:TEST 31      TEST OF ADDF INSTR, DATA SET ADDF-1
:*           ALL INTERRUPT ENABLES ON
:*           SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

TST31: SCOPE
        MOV      #ADDF1,RS      ; PTR TO TEST DATA SET
        JSR     PC,#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,RS      ; PTR TO TEST DATA SET
        JSR     PC,#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,RS      ; PTR TO TEST DATA SET
        JSR     PC,#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 )

```

```

1113
1114
1115
1116
1117
1118
1119 004650 000004
1120 004652 012705 004664
1121 004656 004737 033664
1122
1123 004662 000411
1124
1125 004664
1126 004664 077777 177777
1127 004670 177777 177777
1128 004674 000000 000000
1129 004700 047513 047504
1130 004704 000000
1131
1132
1133
1134
1135
1136
1137
1138 004706 000004
1139 004710 012705 004722
1140 004714 004737 033664
1141
1142 004720 000411
1143
1144 004722
1145 004722 042000 000000
1146 004726 050177 177777
1147 004732 050200 000000
1148 004736 047417 047400
1149 004742 000000
1150
1151
1152
1153
1154
1155
1156
1157 004744 000004
1158 004746 012705 004760
1159 004752 004737 033664
1160
1161 004756 000411
1162
1163 004760
1164 004760 042000 000000
1165 004764 050177 177777
1166 004770 050177 177777
1167 004774 047457 047440
1168 005000 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 )
    
```



```

1169
1170
1171
1172
1173
1174
1175
1176 005002 000004
1177 005004 012705 005016
1178 005010 004737 033664
1179
1180 005014 000411
1181
1182 005016
1183 005016 141777 177777
1184 005022 150177 177777
1185 005026 150177 177777
1186 005032 047507 047510
1187 005036 000000
1188
1189
1190
1191
1192
1193
1194
1195 005040 000004
1196 005042 012705 005054
1197 005046 004737 033664
1198
1199 005052 000411
1200
1201 005054
1202 005054 141777 177777
1203 005060 150177 177777
1204 005064 150177 177777
1205 005070 047547 047550
1206 005074 000000
1207
1208
1209
1210
1211
1212
1213
1214 005076 000004
1215 005100 012705 005112
1216 005104 004737 033664
1217
1218 005110 000411
1219
1220 005112
1221 005112 040177 177777
1222 005116 032200 000000
1223 005122 040200 000000
1224 005126 047457 047440

```

```

*****
:TEST 37 TEST OF ADDF INSTR, DATA SET ADDF-7
:*
:* ALL INTERRUPT ENABLES ON
:* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****
TST37: SCOPE
MOV #ADDF7,RS ; PTR TO TEST DATA SET
JSR PC,3#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
*****
TST40: SCOPE
MOV #ADDF10,RS ; PTR TO TEST DATA SET
JSR PC,3#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
*****
TST41: SCOPE
MOV #ADDF11,RS ; PTR TO TEST DATA SET
JSR PC,3#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

```

K04

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 28
T41 TEST OF ADDF INSTR, DATA SET ADDF-11

SEQ 0195

1225	005132	000000	
1226			
1227			
1228			
1229			
1230			
1231			
1232			
1233	005134	000004	
1234	005136	012705	005150
1235	005142	004737	033664
1236			
1237	005146	000411	
1238			
1239	005150		
1240	005150	140252	125252
1241	005154	140052	125252
1242	005160	140377	177777
1243	005164	047407	047410
1244	005170	000000	
1245			
1246			
1247			
1248			
1249			
1250			
1251			
1252	005172	000004	
1253	005174	012705	005206
1254	005200	004737	033664
1255			
1256	005204	000411	
1257			
1258	005206		
1259	005206	040010	104210
1260	005212	040010	104210
1261	005216	040210	104210
1262	005222	047557	047540
1263	005226	000000	
1264			
1265			
1266			
1267			
1268			
1269			
1270			
1271	005230	000004	
1272	005232	012705	005244
1273	005236	004737	033664
1274			
1275	005242	000411	
1276			
1277	005244		
1278	005244	174177	177777
1279	005250	074177	177776
1280	005254	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,R5 ; 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,R5 ; 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,R5 ; 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

```



```

1281 005260 047507 047510
1282 005264 000000
1283
1284
1285
1286
1287
1288
1289
1290 005266 000004
1291 005270 012705 005302
1292 005274 004737 033664
1293
1294 005300 000411
1295
1296 005302
1297 005302 142200 000000
1298 005306 050177 177777
1299 005312 050177 177776
1300 005316 047417 047400
1301 005322 000000
1302
1303
1304
1305
1306
1307
1308
1309 005324 000004
1310 005326 012705 005340
1311 005332 004737 033664
1312
1313 005336 000411
1314
1315 005340
1316 005340 077777 177777
1317 005344 077777 177777
1318 005350 000177 177777
1319 005354 047451 147446
1320 005360 100010
1321
1322
1323
1324
1325
1326
1327
1328 005362 000004
1329 005364 012705 005376
1330 005370 004737 033664
1331
1332 005374 000411
1333
1334 005376
1335 005376 104000 000000
1336 005402 004000 000001

```

```

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

```

```

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

```

```

TST45: SCOPE
MOV #ADDF15,R5 ; PTR TO TEST DATA SET
JSR PC,3#ADDF15 ; GO TEST
BR TST46 ;;

```

```

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 )

```

```

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

```

```

TST46: SCOPE
MOV #ADDF16,R5 ; PTR TO TEST DATA SET
JSR PC,3#ADDF16 ; GO TEST
BR TST47 ;;

```

```

ADDF16: ; TEST DATA SET ADDF-16:
.WORD LGP,M1 ; INITIAL AC FLOAT NUMBER
.WORD LGP,M1 ; INITIAL MEM FLOAT NUMBER
.WORD ZX1#M1 ; EXPECTED FLOAT RESULT
.WORD 047451,147446 ; FPS: BEFORE, AFTER
.WORD 100010 ; FEC AFTER ( 0 = N/A )

```

```

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

```

```

TST47: SCOPE
MOV #ADDF17,R5 ; PTR TO TEST DATA SET
JSR PC,3#ADDF17 ; GO TEST
BR TST50 ;;

```

```

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

```

M04

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 30
T47 TEST OF ADDF INSTR, DATA SET ADDF-17

SEQ 0197

1337 005406 076200 000000
1338 005412 047517 147500
1339 005416 100012
1340
1341
1342
1343
1344
1345
1346
1347 005420 000004
1348 005422 012705 005434
1349 005426 004737 033664
1350
1351 005432 000411
1352
1353 005434
1354 005434 177777 177777
1355 005440 100000 000000
1356 005444 177777 177777
1357 005450 047543 147543
1358 005454 100014
1359
1360
1361
1362
1363
1364
1365
1366 005456 000004
1367 005460 012705 005472
1368 005464 004737 033664
1369
1370 005470 000411
1371
1372 005472
1373 005472 177777 177777
1374 005476 177777 177777
1375 005502 000000 000000
1376 005506 046511 046506
1377 005512 000000
1378
1379
1380
1381
1382
1383
1384
1385 005514 000004
1386 005516 012705 005530
1387 005522 004737 033664
1388
1389 005526 000411
1390
1391 005530
1392 005530 052525 052525

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

*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,#ADDF2 ; GO TEST

BR TST51 ;;

ADDF20: ; TEST DATA SET ADDF-20:
.WORD LGN,M1 ; INITIAL AC FLOAT NUMBER
.WORD M0,0 ; 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,#ADDF2 ; 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,#ADDF2 ; GO TEST

BR TST53 ;;

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


```

1393 005534 100177 177777
1394 005540 052525 052525
1395 005544 043717 043700
1396 005550 000000
1397
1398
1399
1400
1401
1402
1403
1404 005552 000004
1405 005554 012705 005566
1406 005560 004737 033664
1407
1408 005564 000411
1409
1410 005566
1411 005566 004000 000001
1412 005572 104000 000000
1413 005576 000000 000000
1414 005602 045413 045404
1415 005606 000000
1416
1417
1418

```

```

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

```

```

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

```

```

TST53: SCOPE
MOV #ADDF23,R5 ; PTR TO TEST DATA SET
JSR PC,#ADDF2 ; 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 )

```

```

1419
1420
1421
1422
1423
1424 005610 000004
1425 005612 012705 005624
1426 005616 004737 034034
1427
1428 005622 000417
1429
1430 005624
1431 005624 000177 177777 177777
1432 005632 177777
1433 005634 000000 000000 000000
1434 005642 000000
1435 005644 000000 000000 000000
1436 005652 000000
1437 005654 047713 047704
1438 005660 000000
1439
1440
1441
1442
1443
1444
1445
1446 005662 000004
1447 005664 012705 005676
1448 005670 004737 034034
1449
1450 005674 000417
1451
1452 005676
1453 005676 125252 125252 125252
1454 005704 125252
1455 005706 000000 000000 000000
1456 005714 000000
1457 005716 125252 125252 125252
1458 005724 125252
1459 005726 047747 047750
1460 005732 000000
1461
1462
1463
1464
1465
1466
1467
1468 005734 000004
1469 005736 012705 005750
1470 005742 004737 034034
1471
1472 005746 000417
1473
1474 005750

```

```

*****
:TEST 54 TEST OF ADDO INSTR, DATA SET ADDO-1
:* ALL INTERRUPT ENABLES ON
:* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
TST54: SCOPE
MOV #ADD01,RS ; PTR TO TEST DATA SET
JSR PC,2#ADD0T ; GO TEST
BR TST55 ;;
ADD01: ; TEST DATA SET ADDO-1:
.WORD ZX1MP,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
*****
TST55: SCOPE
MOV #ADD02,RS ; PTR TO TEST DATA SET
JSR PC,2#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
*****
TST56: SCOPE
MOV #ADD03,RS ; PTR TO TEST DATA SET
JSR PC,2#ADD0T ; GO TEST
BR TST57 ;;
ADD03: ; TEST DATA SET ADDO-3:

```



```

1475 005750 000177 177777 177777 .WORD ZXIMP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
1476 005756 177777 .WORD ALTP,ALTP,ALTP,ALTP ; INITIAL MEM FLOAT NUMBER
1477 005760 052525 052525 052525 .WORD ALTP,ALTP,ALTP,ALTP ; EXPECTED FLOAT RESULT
1478 005766 052525 .WORD ALTP,ALTP,ALTP,ALTP ; EXPECTED FLOAT RESULT
1479 005770 052525 052525 052525 .WORD 047617,047600 ; FPS: BEFORE, AFTER
1480 005776 052525 .WORD NA ; FEC AFTER ( 0 = N/A )
1481 006000 047617 047600
1482 006004 000000
1483
1484
1485
1486
1487
1488
1489

```

```

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

```

```

1490 006006 000004
1491 006010 012705 006022
1492 006014 004737 034034
1493
1494 006020 000417
1495

```

```

†T57: SCOPE
MOV #A0004,RS ; PTR TO TEST DATA SET
JSR PC,2#A000T ; GO TEST
BR TST60 ;;

```

```

1496 006022
1497 006022 177777 177777 177777
1498 006030 177777
1499 006032 077777 177777 177777
1500 006040 177777
1501 006042 000000 000000 000000
1502 006050 000000
1503 006052 047653 047644
1504 006056 000000
1505
1506
1507

```

```

ADD04: ; TEST DATA SET ADD-4:
.WORD LGN,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 047653,047644 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

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

```

```

1508
1509
1510
1511
1512 006060 000004
1513 006062 012705 006074
1514 006066 004737 034034
1515
1516 006072 000417
1517

```

```

†T60: SCOPE
MOV #A0005,RS ; PTR TO TEST DATA SET
JSR PC,2#A000T ; GO TEST
BR TST61 ;;

```

```

1518 006074
1519 006074 166177 177777 177777
1520 006102 177777
1521 006104 150000 000000 000000
1522 006112 000000
1523 006114 166200 000000 000000
1524 006122 000000
1525 006124 047607 047610
1526 006130 000000
1527
1528
1529
1530

```

```

ADD05: ; TEST DATA SET ADD-5:
.WORD 166177,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD 150000,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 166200,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 047607,047610 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

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

```

```

1531
1532
1533
1534 006132 000004
1535 006134 012705 006146
1536 006140 004737 034034
1537
1538 006144 000417
1539
1540 006146
1541 006146 166177 177777 177777
1542 006154 177777
1543 006156 150000 000000 000000
1544 006164 000000
1545 006166 166177 177777 177777
1546 006174 177777
1547 006176 047647 047650
1548 006202 000000
1549
1550
1551
1552
1553
1554
1555
1556 006204 000004
1557 006206 012705 006220
1558 006212 004737 034034
1559
1560 006216 000417
1561
1562 006220
1563 006220 066177 177777 177777
1564 006226 177777
1565 006230 047777 177777 177777
1566 006236 177777
1567 006240 066177 177777 177777
1568 006246 177777
1569 006250 047717 047700
1570 006254 000000
1571
1572
1573
1574
1575
1576
1577
1578 006256 000004
1579 006260 012705 006272
1580 006264 004737 034034
1581
1582 006270 000417
1583
1584 006272
1585 006272 066177 177777 177777
1586 006300 177777

```

```

*****
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST61: SCOPE
MOV #ADD06,RS ; 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,RS ; 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,RS ; 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

```


E05

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 35
T63 TEST OF ADDO INSTR, DATA SET ADDO-10

SEQ 0202

1587	006302	047777	177777	177777	.WORD	047777,M1,M1,M1	; INITIAL MEM FLOAT NUMBER
1588	006310	177777					
1589	006312	066177	177777	177777	.WORD	066177,M1,M1,M1	; EXPECTED FLOAT RESULT
1590	006320	177777					
1591	006322	047757	047740		.WORD	047757,047740	; FPS: BEFORE, AFTER
1592	006326	000000			.WORD	NA	; FEC AFTER (0 = N/A)

1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642

```

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

```

```

TST64:  SCOPE
        MOV     @ADD11,R5      ; PTR TO TEST DATA SET
        JSR    PC,@ADD0T     ; GO TEST
        BR     TST65        ;;

```

```

ADD11:  ; TEST DATA SET ADDO-11:
        .WORD  004010,P132,P132,P132 ; INITIAL AC FLOAT NUMBER
        .WORD  004010,P132,P132,P132 ; INITIAL MEM FLOAT NUMBER
        .WORD  004210,P132,P132,P132 ; EXPECTED FLOAT RESULT
        .WORD  047617,047600        ; FPS: BEFORE, AFTER
        .WORD  NA                    ; FEC AFTER ( 0 = N/A )

```

```

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

```

```

TST65:  SCOPE
        MOV     @ADD12,R5      ; PTR TO TEST DATA SET
        JSR    PC,@ADD0T     ; GO TEST
        BR     TST66        ;;

```

```

ADD12:  ; 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

```

F05

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 36
T66 TEST OF ADD INSTR, DATA SET ADD-13

SEQ 0203

```

1643
1644 006454 000004
1645 006456 012705 006470
1646 006462 004737 034034
1647
1648 006466 000417
1649
1650 006470
1651 006470 042252 125252 125252
1652 006476 125252
1653 006500 042052 125252 125252
1654 006506 125252
1655 006510 042377 177777 177777
1656 006516 177777
1657 006520 047717 047700
1658 006524 000000
1659
1660
1661
1662
1663
1664
1665
1666 006526 000004
1667 006530 012705 006542
1668 006534 004737 034034
1669
1670 006540 000417
1671
1672 006542
1673 006542 074177 177777 177777
1674 006550 177777
1675 006552 174177 177777 177777
1676 006560 177776
1677 006562 056200 000000 000000
1678 006570 000000
1679 006572 047617 047600
1680 006576 000000
1681
1682
1683
1684
1685
1686
1687
1688 006600 000004
1689 006602 012705 006614
1690 006606 004737 034034
1691
1692 006612 000417
1693
1694 006614
1695 006614 132200 000000 000000
1696 006622 000000
1697 006624 050177 177777 177777
1698 006632 177777

```

```

*****
TST66: SCOPE
MOV      #ADD013_RS      ; PTR TO TEST DATA SET
JSR      PC, @ADD0T     ; GO TEST

BR       TST67          ;;

ADD013: ; TEST DATA SET ADD-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 ADD INSTR, DATA SET ADD-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 ADD-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 ADD INSTR, DATA SET ADD-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 ADD-15:
.WORD    132200,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD    050177,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER

```


1699 006634 050177 177777 177777
1700 006642 177776
1701 006644 047717 047700
1702 006650 000000

.WORD 050177,M1,M1,M2 ; EXPECTED FLOAT RESULT
.WORD 047717,047700 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

1703
1704
1705

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

1706
1707
1708
1709

TST71: SCOPE
MOV #ADD16,R5 ; PTR TO TEST DATA SET
JSR PC,@ADD0T ; GO TEST
BR TST72 ;;

1710 006652 000004
1711 006654 012705 006666
1712 006660 004737 034034

ADD16: ; TEST DATA SET ADD-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)

1713
1714 006664 000417
1715

1716 006666
1717 006666 077777 177777 177777
1718 006674 177777
1719 006676 100177 177777 177777
1720 006704 177777
1721 006706 077777 177777 177777
1722 006714 177777
1723 006716 047603 147603
1724 006722 100014

1725
1726
1727

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

1728
1729
1730
1731

TST72: SCOPE
MOV #ADD17,R5 ; PTR TO TEST DATA SET
JSR PC,@ADD0T ; GO TEST
BR TST73 ;;

1732 006724 000004
1733 006726 012705 006740
1734 006732 004737 034034

ADD17: ; TEST DATA SET ADD-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)

1735
1736 006736 000417
1737

1738 006740
1739 006740 102000 000000 000000
1740 006746 000001
1741 006750 002000 000000 000000
1742 006756 000000
1743 006760 164200 000000 000000
1744 006766 000000
1745 006770 047647 147650
1746 006774 100012

1747
1748
1749

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

1750
1751
1752
1753

TST73: SCOPE

1754 006776 000004

H05

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 38
T73 TEST OF ADDO INSTR, DATA SET ADDO-20

SEQ 0205

```

1755 007000 012705 007012      MOV      #ADD020,R5      ; PTR TO TEST DATA SET
1756 007004 004737 034034      JSR      PC,@#ADD0T     ; GO TEST
1757
1758 007010 000417              BR       TST74          ;;
1759
1760 007012              ADD020: ; TEST DATA SET ADDO-20:
1761 007012 177777 177777 177777      .WORD   LGN,M1,M1,M1   ; INITIAL AC FLOAT NUMBER
1762 007020 177777              .WORD   LGN,M1,M1,M1   ; INITIAL MEM FLOAT NUMBER
1763 007022 177777 177777 177777      .WORD   LGN,M1,M1,M1   ; INITIAL MEM FLOAT NUMBER
1764 007030 177777              .WORD   ZX1MN,M1,M1,M1 ; EXPECTED FLOAT RESULT
1765 007032 100177 177777 177777      .WORD   047701,147716 ; FPS: BEFORE, AFTER
1766 007040 177777              .WORD   100010         ; FEC AFTER ( 0 = N/A )
1767 007042 047701 147716
1768 007046 100010
1769
1770
1771

```

```

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

```

```

1772
1773
1774
1775
1776 007050 000004              TST74: SCOPE
1777 007052 012705 007064      MOV      #ADD021,R5      ; PTR TO TEST DATA SET
1778 007056 004737 034034      JSR      PC,@#ADD0T     ; GO TEST
1779
1780 007062 000417              BR       TST75          ;;
1781
1782 007064              ADD021: ; TEST DATA SET ADDO-21:
1783 007064 002000 000000 000000      .WORD   002000,0,0,0   ; INITIAL AC FLOAT NUMBER
1784 007072 000000              .WORD   102000,0,0,2   ; INITIAL MEM FLOAT NUMBER
1785 007074 102000 000000 000000      .WORD   0,0,0,0        ; EXPECTED FLOAT RESULT
1786 007102 000002              .WORD   045713,045704 ; FPS: BEFORE, AFTER
1787 007104 000000 000000 000000      .WORD   NA              ; FEC AFTER ( 0 = N/A )
1788 007112 000000
1789 007114 045713 045704
1790 007120 000000
1791
1792
1793

```

```

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

```

```

1794
1795
1796
1797
1798 007122 000004              TST75: SCOPE
1799 007124 012705 007136      MOV      #ADD022,R5      ; PTR TO TEST DATA SET
1800 007130 004737 034034      JSR      PC,@#ADD0T     ; GO TEST
1801
1802 007134 000417              BR       TST76          ;;
1803
1804 007136              ADD022: ; TEST DATA SET ADDO-22:
1805 007136 077777 000000 177777      .WORD   LGP,0,M1,0     ; INITIAL AC FLOAT NUMBER
1806 007144 000000              .WORD   M0,0,0,0       ; INITIAL MEM FLOAT NUMBER
1807 007146 100000 000000 000000      .WORD   LGP,0,M1,0     ; EXPECTED FLOAT RESULT
1808 007154 000000
1809 007156 077777 000000 177777
1810 007164 000000

```


FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 39
T75 TEST OF ADDO INSTR, DATA SET ADDO-22

SEQ 0206

1811	007166	043757	043740
1812	007172	000000	
1813			
1814			
1815			
1816			
1817			
1818			
1819			
1820	007174	000004	
1821	007176	012705	007210
1822	007202	004737	034034
1823			
1824	007206	000417	
1825			
1826	007210		
1827	007210	077777	177777 177777
1828	007216	177777	
1829	007220	077777	177777 177777
1830	007226	177777	
1831	007230	000000	000000 000000
1832	007236	000000	
1833	007240	046611	046606
1834	007244	000000	
1835			
1836			
1837			

.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 )

```

```

1838
1839
1840
1841
1842
1843 007246 000004
1844 007250 012705 007262
1845 007254 004737 034224
1846
1847 007260 000411
1848
1849 007262
1850 007262 000000 000000
1851 007266 000000 000000
1852 007272 000000 000000
1853 007276 047413 047404
1854 007302 000000
1855
1856
1857
1858
1859
1860
1861
1862 007304 000004
1863 007306 012705 007320
1864 007312 004737 034224
1865
1866 007316 000411
1867
1868 007320
1869 007320 000177 177777
1870 007324 000177 125252
1871 007330 000000 000000
1872 007334 047453 047444
1873 007340 000000
1874
1875
1876
1877
1878
1879
1880
1881 007342 000004
1882 007344 012705 007356
1883 007350 004737 034224
1884
1885 007354 000411
1886
1887 007356
1888 007356 000177 052525
1889 007362 100345 123456
1890 007366 000345 123456
1891 007372 047517 047500
1892 007376 000000
1893
    
```

```

*****
*TEST 77 TEST OF SUBF INSTR, DATA SET SUBF-1
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****
    
```

```

TST77: SCOPE
MOV #SUBF1,RS ; 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
*****
    
```

```

TST100: SCOPE
MOV #SUBF2,RS ; 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
*****
    
```

```

TST101: SCOPE
MOV #SUBF3,RS ; 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 )
    
```


K05

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 41
T101 TEST OF SUBF INSTR, DATA SET SUBF-3

SEQ 0208

1894			
1895			
1896			
1897			
1898			
1899			
1900	007400	000004	
1901	007402	012705	007414
1902	007406	004737	034224
1903			
1904	007412	000411	
1905			
1906	007414		
1907	007414	040200	000000
1908	007420	040200	000000
1909	007424	000000	000000
1910	007430	047553	047544
1911	007434	000000	
1912			
1913			
1914			
1915			
1916			
1917			
1918			
1919	007436	000004	
1920	007440	012705	007452
1921	007444	004737	034224
1922			
1923	007450	000411	
1924			
1925	007452		
1926	007452	140200	000000
1927	007456	140200	000000
1928	007462	000000	000000
1929	007466	047413	047404
1930	007472	000000	
1931			
1932			
1933			
1934			
1935			
1936			
1937			
1938	007474	000004	
1939	007476	012705	007510
1940	007502	004737	034224
1941			
1942	007506	000411	
1943			
1944	007510		
1945	007510	150365	052525
1946	007514	047252	125252
1947	007520	150377	177777
1948	007524	047447	047450
1949	007530	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,RS ; 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,RS ; 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,RS ; 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 )

```

L05

FPU ADVANCED INSTR TESTS
DQFPBB.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 42
T104 TEST OF SUBF INSTR, DATA SET SUBF-6

SEQ 0209

1950			
1951			
1952			
1953			
1954			
1955			
1956			
1957	007532	000004	
1958	007534	012705	007546
1959	007540	004737	034224
1960			
1961	007544	000411	
1962			
1963	007546		
1964	007546	050365	052525
1965	007552	147252	125252
1966	007556	050400	000000
1967	007562	047517	047500
1968	007566	000000	
1969			
1970			
1971			
1972			
1973			
1974			
1975			
1976	007570	000004	
1977	007572	012705	007604
1978	007576	004737	034224
1979			
1980	007602	000411	
1981			
1982	007604		
1983	007604	077777	177777
1984	007610	100177	177777
1985	007614	077777	177777
1986	007620	047555	147555
1987	007624	100014	
1988			
1989			
1990			
1991			
1992			
1993			
1994			
1995	007626	000004	
1996	007630	012705	007642
1997	007634	004737	034224
1998			
1999	007640	000411	
2000			
2001	007642		
2002	007642	077777	177777
2003	007646	100177	177777
2004	007652	077777	177777
2005	007656	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   ZX1AN,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   ZX1AN,M1      ; INITIAL MEM FLOAT NUMBER
.WORD   LGP,M1         ; EXPECTED FLOAT RESULT
.WORD   043457,043440 ; FPS: BEFORE, AFTER

```


M05

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 43
T107 TEST OF SUBF INSTR, DATA SET SUBF-11

SEQ 0210

2006	007662	000000	
2007			
2008			
2009			
2010			
2011			
2012			
2013			
2014	007664	000004	
2015	007666	012705	007700
2016	007672	004737	034224
2017			
2018	007676	000411	
2019			
2020	007700		
2021	007700	177777	177777
2022	007704	071600	000000
2023	007710	177777	177777
2024	007714	047447	047450
2025	007720	000000	
2026			
2027			
2028			
2029			
2030			
2031			
2032			
2033	007722	000004	
2034	007724	012705	007736
2035	007730	004737	034224
2036			
2037	007734	000411	
2038			
2039	007736		
2040	007736	177777	177777
2041	007742	071600	000000
2042	007746	100000	000000
2043	007752	047501	147516
2044	007756	100010	
2045			
2046			
2047			
2048			
2049			
2050			
2051			
2052	007760	000004	
2053	007762	012705	007774
2054	007766	004737	034224
2055			
2056	007772	000411	
2057			
2058	007774		
2059	007774	177777	177777
2060	010000	071600	000000
2061	010004	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

```

2062 010010 046511 046506
 2063 010014 000000
 2064
 2065
 2066
 2067
 2068
 2069
 2070
 2071 010016 000004
 2072 010020 012705 010032
 2073 010024 004737 034224
 2074
 2075 010030 000411
 2076
 2077 010032
 2078 010032 004200 000000
 2079 010036 004200 000001
 2080 010042 176400 000000
 2081 010046 047447 147450
 2082 010052 100012
 2083
 2084
 2085
 2086
 2087
 2088
 2089
 2090 010054 000004
 2091 010056 012705 010070
 2092 010062 004737 034224
 2093
 2094 010066 000411
 2095
 2096 010070
 2097 010070 004200 000000
 2098 010074 004200 000001
 2099 010100 000000 000000
 2100 010104 045453 045444
 2101 010110 000000
 2102
 2103
 2104

.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
*****

```

```

TST113: 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
*****

```

```

TST114: 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 )

```


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
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160

010112 000004
010114 012705 010126
010120 004737 034374

010124 000417

010126 000000 000000 000000
010134 000000
010136 000000 000000 000000
010144 000000
010146 000000 000000 000000
010154 000000
010156 047753 047744
010162 000000

```
*****  
: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 )
```

010164 000004
010166 012705 010200
010172 004737 034374

010176 000417

010200 000177 052525 052525
010206 052525
010210 000177 177777 177777
010216 177777
010220 000000 000000 000000
010226 000000
010230 047713 047704
010234 000000

```
*****  
: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 )
```

010236 000004
010240 012705 010252
010244 004737 034374

010250 000417

010252

```
*****  
: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:
```

2161	010252	000000	000000	000000	.WORD	0,0,0,0	; INITIAL AC FLOAT NUMBER
2162	010260	000000					
2163	010262	012345	177777	125252	.WORD	012345,M1,ALTN,M0	; INITIAL MEM FLOAT NUMBER
2164	010270	100000					
2165	010272	112345	177777	125252	.WORD	112345,M1,ALTN,M0	; EXPECTED FLOAT RESULT
2166	010300	100000					
2167	010302	047647	047650		.WORD	047647,047650	; FPS: BEFORE, AFTER
2168	010306	000000			.WORD	NA	; FEC AFTER (0 = N/A)

```

2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216

```

```

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

```

```

TST120: SCOPE
MOV #SUBD4,RS ; PTR TO TEST DATA SET
JSR PC,2#SUBDT ; GO TEST
BR TST121 ;;

```

```

SUBD4: ; TEST DATA SET SUBD-4:
.WORD FIN,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD FIN,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 121 TEST OF SUBD INSTR, DATA SET SUBD-5
;
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

TST121: SCOPE
MOV #SUBD5,RS ; PTR TO TEST DATA SET
JSR PC,2#SUBDT ; GO TEST
BR TST122 ;;

```

```

SUBD5: ; TEST DATA SET SUBD-5:
.WORD F1P,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD F1P,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 047613,047604 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

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

```



```

2217
2218
2219
2220 010434 000004
2221 010436 012705 010450
2222 010442 004737 034374
2223
2224 010446 000417
2225
2226 010450
2227 010450 037252 125252 125252
2228 010456 125252
2229 010460 140365 052525 052525
2230 010466 052525
2231 010470 040377 177777 177777
2232 010476 177777
2233 010500 047757 047740
2234 010504 000000
2235
2236
2237
2238
2239
2240
2241
2242 010506 000004
2243 010510 012705 010522
2244 010514 004737 034374
2245
2246 010520 000417
2247
2248 010522
2249 010522 137252 125252 125252
2250 010530 125252
2251 010532 040365 052525 052525
2252 010540 052525
2253 010542 140400 000000 000000
2254 010550 000000
2255 010552 047607 047610
2256 010556 000000
2257
2258
2259
2260
2261
2262
2263
2264 010560 000004
2265 010562 012705 010574
2266 010566 004737 034374
2267
2268 010572 000417
2269
2270 010574
2271 010574 177777 177777 000000
2272 010602 177777

```

```

*****
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****
TST122: SCOPE
MOV #SUBD6,RS ; 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,RS ; 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,RS ; 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

```

2273	010604	100000	000000	177777	.WORD	MO,0,M1,0	; INITIAL MEM FLOAT NUMBER
2274	010612	000000					
2275	010614	177777	177777	000000	.WORD	LGN,M1,0,M1	; EXPECTED FLOAT RESULT
2276	010622	177777					
2277	010624	047603	147603		.WORD	047603,147603	; FPS: BEFORE, AFTER
2278	010630	100014			.WORD	100014	; FEC AFTER (0 = N/A)

2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328

```

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

```

```

TST125: SCOPE
MOV #SUBD11,RS ; PTR TO TEST DATA SET
JSR PC,2#SUBDT ; GO TEST
BR TST126 ;;

SUBD11: ; TEST DATA SET SUBD-11:
.WORD LGN,M1,0,M1 ; INITIAL AC FLOAT NUMBER
.WORD MO,0,M1,0 ; INITIAL MEM FLOAT NUMBER
.WORD LGN,M1,0,M1 ; EXPECTED FLOAT RESULT
.WORD 043707,043710 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

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

```

```

TST126: SCOPE
MOV #SUBD12,RS ; PTR TO TEST DATA SET
JSR PC,2#SUBDT ; GO TEST
BR TST127 ;;

SUBD12: ; TEST DATA SET SUBD-12:
.WORD 104200,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD 104200,0,0,1 ; INITIAL MEM FLOAT NUMBER
.WORD 066400,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 047717,147700 ; FPS: BEFORE, AFTER
.WORD 100012 ; FEC AFTER ( 0 = N/A )

```

```

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

```



```

2329
2330 010756 000004
2331 010760 012705 010772
2332 010764 004737 034374
2333
2334 010770 000417
2335
2336 010772
2337 010772 104200 000000 000000
2338 011000 000001
2339 011002 104200 000000 000000
2340 011010 000000
2341 011012 000000 000000 000000
2342 011020 000000
2343 011022 045713 045704
2344 011026 000000
2345
2346
2347
2348
2349
2350
2351
2352 011030 000004
2353 011032 012705 011044
2354 011036 004737 034374
2355
2356 011042 000417
2357
2358 011044
2359 011044 077777 177777 177777
2360 011052 177777
2361 011054 161600 000000 000000
2362 011062 000000
2363 011064 077777 177777 177777
2364 011072 177777
2365 011074 047757 047740
2366 011100 000000
2367
2368
2369
2370
2371
2372
2373
2374 011102 000004
2375 011104 012705 011116
2376 011110 004737 034374
2377
2378 011114 000417
2379
2380 011116
2381 011116 077777 177777 177777
2382 011124 177777
2383 011126 161600 000000 000000
2384 011134 000000

```

```

*****
TST127: 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
*****
TST130: 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
*****
TST131: 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

```

2385	011136	000000	000000	000000	.WORD	0,0,0,0 ; EXPECTED FLOAT RESULT
2386	011144	000000				
2387	011146	047611	147606		.WORD	047611,147606 ; FPS: BEFORE, AFTER
2388	011152	100010			.WORD	100010 ; FEC AFTER (0 = N/A)

2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412

```

*****
: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,RS      ; 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 )

```

2396	011154	000004		
2397	011156	012705	011170	
2398	011162	004737	034374	
2400	011166	000417		
2402	011170			
2403	011170	077777	177777	177777
2404	011176	177777		
2405	011200	161600	000000	000000
2406	011206	000000		
2407	011210	000000	000000	000000
2408	011216	000000		
2409	011220	046611	046606	
2410	011224	000000		


```

2413
2414
2415
2416
2417
2418 011226 000004
2419 011230 012705 011242
2420 011234 004737 034564
2421
2422 011240 000411
2423
2424 011242
2425 011242 000000 000000
2426 011246 177777 177777
2427 011252 000000 000000
2428 011256 047413 047404
2429 011262 000000
2430
2431
2432
2433
2434
2435
2436
2437 011264 000004
2438 011266 012705 011300
2439 011272 004737 034564
2440
2441 011276 000411
2442
2443 011300
2444 011300 077777 177777
2445 011304 000177 177777
2446 011310 000000 000000
2447 011314 047503 047504
2448 011320 000000
2449
2450
2451
2452
2453
2454
2455
2456 011322 000004
2457 011324 012705 011336
2458 011330 004737 034564
2459
2460 011334 000411
2461
2462 011336
2463 011336 123652 125252
2464 011342 017500 000000
2465 011346 103177 177777
2466 011352 047447 047450
2467 011356 000000
2468

```

```

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

```

```

TST133: 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
*****

```

```

TST134: 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 ZX1MP,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
*****

```

```

TST135: 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 )

```


011512
011514
011520
011524
011526
011526
011532
011536
011542
011546
011550
011552
011556
011562
011564
011564
011570
011574
011600
011604
011606
011610
011614
011620
011622
011622
011626
011632
011636

000004
012705 011526
004737 034564

000411

000004
140277 000000
060000 000001
160077 000001
047407 047410
000000

000004
012705 011564
004737 034564

000411

000004
060000 000001
040277 000000
060077 000001
047457 047440
000000

000004
012705 011622
004737 034564

000411

000004
140300 000000
160000 000001
060100 000002
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
```

```

011642 000000
011644 000004
011646 012705 011660
011652 004737 034564
011656 000411
011660
011660 060000 000001
011664 140300 000000
011670 160100 000001
011674 047547 047550
011700 000000
011702 000004
011704 012705 011716
011710 004737 034564
011714 000411
011716
011716 002177 177777
011722 002177 177777
011726 044177 177776
011732 047513 147500
011736 100012
011740 000004
011742 012705 011754
011746 004737 034564
011752 000411
011754
011754 170000 000000
011760 050200 000000
011764 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,R5 ; 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,R5 ; 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,R5 ; 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

```

2-20

2637 011770 047441 147456
2638 011774 100010

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

2640
2641
2642
2643
2644
2645

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

2646 011776 000004
2647 012000 012705 012012
2648 012004 004737 034564
2649
2650 012010 000411

TST147: SCOPE
MOV #MULF15,R5 ; PTR TO TEST DATA SET
JSR PC,#MULFT ; GO TEST
BR TST150 ;;

2651
2652 012012
2653 012012 177777 177777
2654 012016 100177 177777
2655 012022 177777 177777
2656 012026 047447 147447
2657 012032 100014

MULF15: ; TEST DATA SET MULF-15:
.WORD LGN,M1 ; INITIAL AC FLOAT NUMBER
.WORD ZX1/M,M1 ; INITIAL MEM FLOAT NUMBER
.WORD LGN,M1 ; EXPECTED FLOAT RESULT
.WORD 047447,147447 ; FPS: BEFORE, AFTER
.WORD 100014 ; FEC AFTER (0 = N/A)

2658
2659
2660
2661
2662
2663
2664

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

2665 012034 000004
2666 012036 012705 012050
2667 012042 004737 034564
2668
2669 012046 000411

TST150: SCOPE
MOV #MULF16,R5 ; PTR TO TEST DATA SET
JSR PC,#MULFT ; GO TEST
BR TST151 ;;

2670
2671 012050
2672 012050 050377 000000
2673 012054 070000 177777
2674 012060 000000 000000
2675 012064 046411 046406
2676 012070 000000

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)

2677
2678
2679
2680
2681
2682
2683

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

2684 012072 000004
2685 012074 012705 012106
2686 012100 004737 034564
2687
2688 012104 000411

TST151: SCOPE
MOV #MULF17,R5 ; PTR TO TEST DATA SET
JSR PC,#MULFT ; GO TEST
BR TST152 ;;

2689 012106
2690 012106 002177 177777
2691 012106 002177 177777
2692 012112 002177 177777

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

M06

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 56
T151 TEST OF MULF INSTR, DATA SET MULF-17

SEQ 0223

2693	012116	000000	000000
2694	012122	045553	045544
2695	012126	000000	
2696			
2697			
2698			
2699			
2700			
2701			
2702			
2703	012130	000004	
2704	012132	012705	012144
2705	012136	004737	034564
2706			
2707	012142	000411	
2708			
2709	012144		
2710	012144	052525	052525
2711	012150	100000	177777
2712	012154	000000	000000
2713	012160	043513	043504
2714	012164	000000	
2715			
2716			
2717			

```

: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 M0, M1 ; INITIAL MEM FLOAT NUMBER
:WORD 0, 0 ; EXPECTED FLOAT RESULT
:WORD 043513, 043504 ; FPS: BEFORE, AFTER
:WORD NA ; FEC AFTER ( 0 = N/A )

```



```

2718
2719
2720
2721
2722
2723 012166 000004
2724 012170 012705 012202
2725 012174 004737 034734
2726
2727 012200 000417
2728
2729 012202
2730 012202 077777 177777 177777
2731 012210 177777
2732 012212 000000 000000 000000
2733 012220 000000
2734 012222 000000 000000 000000
2735 012230 000000
2736 012232 047713 047704
2737 012236 000000
2738
2739
2740
2741
2742
2743
2744
2745 012240 000004
2746 012242 012705 012254
2747 012246 004737 034734
2748
2749 012252 000417
2750
2751 012254
2752 012254 000177 177777 177777
2753 012262 177777
2754 012264 177777 177777 177777
2755 012272 177777
2756 012274 000000 000000 000000
2757 012302 000000
2758 012304 047603 047604
2759 012310 000000
2760
2761
2762
2763
2764
2765
2766
2767 012312 000004
2768 012314 012705 012326
2769 012320 004737 034734
2770
2771 012324 000417
2772
2773 012326

```

```

*****
*TEST 153 TEST OF MULD INSTR, DATA SET MULD-1
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
TST153: SCOPE
MOV #MULD1,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULD1 ; 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
*****
TST154: SCOPE
MOV #MULD2,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULD2 ; 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
*****
TST155: SCOPE
MOV #MULD3,R5 ; PTR TO TEST DATA SET
JSR PC,@#MULD3 ; GO TEST
BR TST156 ;;

MULD3: ; TEST DATA SET MULD-3:

```

2774	012326	023652	125252	125252	.WORD	023652,ALTN,ALTN,ALTN	; INITIAL AC FLOAT NUMBER
2775	012334	125252					
2776	012336	017500	000000	000000	.WORD	017500,0,0,0	; INITIAL MEM FLOAT NUMBER
2777	012344	000000					
2778	012346	003177	177777	177777	.WORD	003177,M1,M1,M1	; EXPECTED FLOAT RESULT
2779	012354	177777					
2780	012356	047757	047740		.WORD	047757,047740	; FPS: BEFORE, AFTER
2781	012362	000000			.WORD	NA	; FEC AFTER (0 = N/A)

```

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

```

2789	012364	000004			TST156: SCOPE		
2790	012366	012705	012400		MOV	#MULD4,R5	; PTR TO TEST DATA SET
2791	012372	004737	034734		JSR	PC,3#MULD4	; GO TEST
2792							
2793	012376	000417			BR	TST157	::

```

MULD4: ; TEST DATA SET MULD-4:
;WORD 117500,0,0,0 ; INITIAL AC FLOAT NUMBER
;WORD 123652,ALTN,ALTN,ALTN ; INITIAL MEM FLOAT NUMBER
;WORD 003177,M1,M1,M1 ; EXPECTED FLOAT RESULT
;WORD 047617,047600 ; FPS: BEFORE, AFTER
;WORD NA ; FEC AFTER ( 0 = N/A )

```

2795	012400						
2796	012400	117500	000000	000000			
2797	012406	000000					
2798	012410	123652	125252	125252	.WORD	123652,ALTN,ALTN,ALTN	; INITIAL MEM FLOAT NUMBER
2799	012416	125252					
2800	012420	003177	177777	177777	.WORD	003177,M1,M1,M1	; EXPECTED FLOAT RESULT
2801	012426	177777					
2802	012430	047617	047600		.WORD	047617,047600	; FPS: BEFORE, AFTER
2803	012434	000000			.WORD	NA	; FEC AFTER (0 = N/A)

```

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

```

2811	012436	000004			TST157: SCOPE		
2812	012440	012705	012452		MOV	#MULD5,R5	; PTR TO TEST DATA SET
2813	012444	004737	034734		JSR	PC,3#MULD5	; GO TEST
2814							
2815	012450	000417			BR	TST160	::

```

MULD5: ; TEST DATA SET MULD-5:
;WORD 165400,0,0,1 ; INITIAL AC FLOAT NUMBER
;WORD 037577,M1,M1,M2 ; INITIAL MEM FLOAT NUMBER
;WORD 164777,M1,M1,M1 ; EXPECTED FLOAT RESULT
;WORD 047747,047750 ; FPS: BEFORE, AFTER
;WORD NA ; FEC AFTER ( 0 = N/A )

```

2817	012452						
2818	012452	165400	000000	000000			
2819	012460	000001					
2820	012462	037577	177777	177777	.WORD	037577,M1,M1,M2	; INITIAL MEM FLOAT NUMBER
2821	012470	177776					
2822	012472	164777	177777	177777	.WORD	164777,M1,M1,M1	; EXPECTED FLOAT RESULT
2823	012500	177777					
2824	012502	047747	047750		.WORD	047747,047750	; FPS: BEFORE, AFTER
2825	012506	000000			.WORD	NA	; FEC AFTER (0 = N/A)

```

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

```

2826
2827
2828
2829


```

2830
2831
2832
2833 012510 000004
2834 012512 012705 012524
2835 012516 004737 034734
2836
2837 012522 000417
2838
2839 012524
2840 012524 165400 000000 000000
2841 012532 000001
2842 012534 037577 177777 177777
2843 012542 177776
2844 012544 165000 000000 000000
2845 012552 000000
2846 012554 047707 047710
2847 012560 000000

```

```

*****
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
TST160: SCOPE
MOV #MULD6,RS ; 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 )

```

```

2848
2849
2850
2851
2852
2853
2854
2855 012562 000004
2856 012564 012705 012576
2857 012570 004737 034734
2858
2859 012574 000417
2860
2861 012576
2862 012576 040277 000000 000000
2863 012604 000000
2864 012606 034200 000000 000000
2865 012614 000001
2866 012616 034277 000000 000000
2867 012624 000001
2868 012626 047657 047640
2869 012632 000000

```

```

*****
* TEST 161 TEST OF MULD INSTR, DATA SET MULD-7
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST161: SCOPE
MOV #MULD7,RS ; 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 )

```

```

2870
2871
2872
2873
2874
2875
2876
2877 012634 000004
2878 012636 012705 012650
2879 012642 004737 034734
2880
2881 012646 000417
2882
2883 012650
2884 012650 140277 000000 000000
2885 012656 000000

```

```

*****
* TEST 162 TEST OF MULD INSTR, DATA SET MULD-10
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****
TST162: SCOPE
MOV #MULD10,RS ; 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

```

2886	012660	034200	000000	000000	.WORD	034200,0,0,1	; INITIAL MEM FLOAT NUMBER
2887	012666	000001					
2888	012670	134277	000000	000000	.WORD	134277,0,0,1	; EXPECTED FLOAT RESULT
2889	012676	000001					
2890	012700	047607	047610		.WORD	047607,047610	; FPS: BEFORE, AFTER
2891	012704	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
: *
*****

```

2899	012706	000004			TST163: SCOPE		
2900	012710	012705	012722		MOV	#MULD11,RS	; PTR TO TEST DATA SET
2901	012714	004737	034734		JSR	PC,#MULDT	; GO TEST
2902							
2903	012720	000417			BR	TST164	::

2905	012722				MULD11: ; TEST DATA SET MULD-11:		
2906	012722	040300	000000	000000	.WORD	040300,0,0,0	; INITIAL AC FLOAT NUMBER
2907	012730	000000					
2908	012732	134200	000000	000000	.WORD	134200,0,0,1	; INITIAL MEM FLOAT NUMBER
2909	012740	000001					
2910	012742	134300	000000	000000	.WORD	134300,0,0,1	; EXPECTED FLOAT RESULT
2911	012750	000001					
2912	012752	047747	047750		.WORD	047747,047750	; FPS: BEFORE, AFTER
2913	012756	000000			.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
: *
*****

```

2921	012760	000004			TST164: SCOPE		
2922	012762	012705	012774		MOV	#MULD12,RS	; PTR TO TEST DATA SET
2923	012766	004737	034734		JSR	PC,#MULDT	; GO TEST
2924							
2925	012772	000417			BR	TST165	::

2927	012774				MULD12: ; TEST DATA SET MULD-12:		
2928	012774	140300	000000	000000	.WORD	140300,0,0,0	; INITIAL AC FLOAT NUMBER
2929	013002	000000					
2930	013004	134200	000000	000000	.WORD	134200,0,0,1	; INITIAL MEM FLOAT NUMBER
2931	013012	000001					
2932	013014	034300	000000	000000	.WORD	034300,0,0,2	; EXPECTED FLOAT RESULT
2933	013022	000002					
2934	013024	047717	047700		.WORD	047717,047700	; FPS: BEFORE, AFTER
2935	013030	000000			.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
: *
*****

```

2941


```

2943
2944 013032 000004
2945 013034 012705 013046
2946 013040 004737 034734
2947
2948 013044 000417
2949
2950 013046
2951 013046 177777 177777 177777
2952 013054 177777
2953 013056 177777 177777 177777
2954 013064 177777
2955 013066 037577 177777 177777
2956 013074 177776
2957 013076 047655 147642
2958 013102 100010
2959
2960
2961
2962
2963
2964
2965
2966 013104 000004
2967 013106 012705 013120
2968 013112 004737 034734
2969
2970 013116 000417
2971
2972 013120
2973 013120 077777 177777 177777
2974 013126 177777
2975 013130 077777 177777 177777
2976 013136 177777
2977 013140 000000 000000 000000
2978 013146 000000
2979 013150 046751 046746
2980 013154 000000
2981
2982
2983
2984
2985
2986
2987 013156 000004
2988 013160 012705 013172
2989 013164 004737 034734
2990
2991 013170 000417
2992
2993 013172
2994 013172 003177 177777 177777
2995 013200 177777
2996 013202 101177 177777 177777
2997 013210 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

```

F07

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 62
T167 TEST OF MULD INSTR, DATA SET MULD-15

SEQ 0229

2998	013212	144177	177777	177777	.WORD	144177,M1,M1,M2	; EXPECTED FLOAT RESULT
2999	013220	177776					
3000	013222	047647	147650		.WORD	047647,147650	; FPS: BEFORE, AFTER
3001	013226	100012			.WORD	100012	; FEC AFTER (0 = N/A)

```

3002
3003
3004
3005
3006
3007
3008
*****
:TEST 170 TEST OF MULD INSTR, DATA SET MULD-16
:
: UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
: LONG FLOAT, LONG INTEGER, ROUND MODES
:
*****

```

```

3009 013230 000004
3010 013232 012705 013244
3011 013236 004737 034734
3012
3013 013242 000417
3014
TST170: SCOPE
MOV #MULD16,R5 ; PTR TO TEST DATA SET
JSR PC,#MULDT ; GO TEST
BR TST171 ;;

```

```

3015 013244
MULD16: ; TEST DATA SET MULD-16:
3016 013244 103177 177777 177777 .WORD 103177,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
3017 013252 177777
3018 013254 001177 177777 177777 .WORD 001177,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
3019 013262 177777
3020 013264 000000 000000 000000 .WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
3021 013272 000000
3022 013274 045713 045704 .WORD 045713,045704 ; FPS: BEFORE, AFTER
3023 013300 000000 .WORD NA ; FEC AFTER ( 0 = N/A )
3024
3025

```

```

3026
3027
3028
3029
3030
*****
:TEST 171 TEST OF MULD INSTR, DATA SET MULD-17
:
: ALL INTERRUPT ENABLES ON
: LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
:
*****

```

```

3031 013302 000004
3032 013304 012705 013316
3033 013310 004737 034734
3034
3035 013314 000417
3036
TST171: SCOPE
MOV #MULD17,R5 ; PTR TO TEST DATA SET
JSR PC,#MULDT ; GO TEST
BR TST172 ;;

```

```

3037 013316
MULD17: ; TEST DATA SET MULD-17:
3038 013316 052525 052525 052525 .WORD ALTP,ALTP,ALTP,ALTP ; INITIAL AC FLOAT NUMBER
3039 013324 052525
3040 013326 100177 177777 177777 .WORD ZX1MN,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
3041 013334 177777
3042 013336 052525 052525 052525 .WORD ALTP,ALTP,ALTP,ALTP ; EXPECTED FLOAT RESULT
3043 013344 052525
3044 013346 047657 147657 .WORD 047657,147657 ; FPS: BEFORE, AFTER
3045 013352 100014 .WORD 100014 ; FEC AFTER ( 0 = N/A )
3046
3047

```

```

3048
3049
3050
3051
3052
*****
:TEST 172 TEST OF MULD INSTR, DATA SET MULD-20
:
: -0 INTERRUPT ENABLE OFF, ALL OTHERS ON
: LONG FLOAT, LONG INTEGER, TRUNCATE MODES
:
*****

```

```

3053 013354 000004
TST172: SCOPE

```


G07

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 63
T172 TEST OF MULD INSTR, DATA SET MULD-20

SEQ 0230

3054	013356	012705	013370		MOV	#MULD20,R5	:	PTR TO TEST DATA SET
3055	013362	004737	034734		JSR	PC,#MULDT	:	GO TEST
3056								
3057	013366	000417			BR	TST173	::	
3058								
3059	013370				MULD20:	:	TEST DATA SET MULD-20:	
3060	013370	125252	125252	125252	.WORD	ALTN,ALTN,ALTN,ALTN	:	INITIAL AC FLOAT NUMBER
3061	013376	125252						
3062	013400	100000	177777	052525	.WORD	M0,M1,ALTP,ALTN	:	INITIAL MEM FLOAT NUMBER
3063	013406	125252						
3064	013410	000000	000000	000000	.WORD	0,0,0,0	:	EXPECTED FLOAT RESULT
3065	013416	000000						
3066	013420	043753	043744		.WORD	043753,043744	:	FPS: BEFORE, AFTER
3067	013424	000000			.WORD	NA	:	FEC AFTER (0 = N/A)
3068								
3069								

H07

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 64
T173 TEST OF DIVF INSTR, DATA SET DIVF-1

SEQ 0231

3070			
3071			
3072			
3073			
3074			
3075	013426	000004	
3076	013430	012705	013442
3077	013434	004737	035124
3078			
3079	013440	000411	
3080			
3081	013442		
3082	013442	103177	177777
3083	013446	023652	125252
3084	013452	117500	000000
3085	013456	047447	047450
3086	013462	000000	
3087			
3088			
3089			
3090			
3091			
3092			
3093			
3094	013464	000004	
3095	013466	012705	013500
3096	013472	004737	035124
3097			
3098	013476	000411	
3099			
3100	013500		
3101	013500	052525	052525
3102	013504	000000	000000
3103	013510	052525	052525
3104	013514	047517	147517
3105	013520	100004	
3106			
3107			
3108			
3109			
3110			
3111			
3112			
3113	013522	000004	
3114	013524	012705	013536
3115	013530	004737	035124
3116			
3117	013534	000411	
3118			
3119	013536		
3120	013536	140400	000000
3121	013542	040500	000000
3122	013546	140052	125252
3123	013552	047447	047450
3124	013556	000000	
3125			

```
*****
: *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 )
```



```

3126
3127
3128
3129
3130
3131
3132 013560 000004
3133 013562 012705 013574
3134 013566 004737 035124
3135
3136 013572 000411
3137
3138 013574
3139 013574 040400 000000
3140 013600 140500 000000
3141 013604 140052 125253
3142 013610 047507 047510
3143 013614 000000
3144
3145
3146
3147
3148
3149
3150
3151 013616 000004
3152 013620 012705 013632
3153 013624 004737 035124
3154
3155 013630 000411
3156
3157 013632
3158 013632 007417 007417
3159 013636 007417 007417
3160 013642 040200 000000
3161 013646 047417 047400
3162 013652 000000
3163
3164
3165
3166
3167
3168
3169
3170 013654 000004
3171 013656 012705 013670
3172 013662 004737 035124
3173
3174 013666 000411
3175
3176 013670
3177 013670 160400 000000
3178 013674 154000 000000
3179 013700 044600 000000
3180 013704 047557 047540
3181 013710 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 )

```

```

3182
3183
3184
3185
3186
3187
3188
3189 013712 000004
3190 013714 012705 013726
3191 013720 004737 035124
3192
3193 013724 000411
3194
3195 013726
3196 013726 000177 177777
3197 013732 177777 177777
3198 013736 000000 000000
3199 013742 047453 047444
3200 013746 000000
3201
3202
3203
3204
3205
3206
3207
3208 013750 000004
3209 013752 012705 013764
3210 013756 004737 035124
3211
3212 013762 000411
3213
3214 013764
3215 013764 160077 000000
3216 013770 140277 000000
3217 013774 060000 000000
3218 014000 047517 047500
3219 014004 000000
3220
3221
3222
3223
3224
3225
3226
3227 014006 000004
3228 014010 012705 014022
3229 014014 004737 035124
3230
3231 014020 000411
3232
3233 014022
3234 014022 160077 000000
3235 014026 040277 000000
3236 014032 160000 000000
3237 014036 047447 047450

```

```

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

```

```

†TST201: SCOPE
MOV #DIVF7,R5 ; PTR TO TEST DATA SET
JSR PC,3#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
*****

```

```

†TST202: SCOPE
MOV #DIVF10,R5 ; PTR TO TEST DATA SET
JSR PC,3#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
*****

```

```

†TST203: SCOPE
MOV #DIVF11,R5 ; PTR TO TEST DATA SET
JSR PC,3#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

```


K07

FPU ADVANCED INSTR TESTS
DQFPBB.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 67
T203 TEST OF DIVF INSTR, DATA SET DIVF-11

SEQ 0234

3238	014042	000000	
3239			
3240			
3241			
3242			
3243			
3244			
3245			
3246	014044	000004	
3247	014046	012705	014060
3248	014052	004737	035124
3249			
3250	014056	000411	
3251			
3252	014060		
3253	014060	060100	000001
3254	014064	040300	000000
3255	014070	060000	000001
3256	014074	047517	047500
3257	014100	000000	
3258			
3259			
3260			
3261			
3262			
3263			
3264			
3265	014102	000004	
3266	014104	012705	014116
3267	014110	004737	035124
3268			
3269	014114	000411	
3270			
3271	014116		
3272	014116	060100	000001
3273	014122	140300	000000
3274	014126	160000	000000
3275	014132	047447	047450
3276	014136	000000	
3277			
3278			
3279			
3280			
3281			
3282			
3283			
3284	014140	000004	
3285	014142	012705	014154
3286	014146	004737	035124
3287			
3288	014152	000411	
3289			
3290	014154		
3291	014154	000177	177777
3292	014160	100177	177777
3293	014164	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

```

3294 014170 047543 147543
3295 014174 100014

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

3296
3297
3298
3299
3300
3301
3302

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

3303 014176 000004
3304 014200 012705 014212
3305 014204 004737 035124
3306
3307 014210 000411
3308

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

BR TST210 ;;

3309 014212
3310 014212 000177 177777
3311 014216 100177 177777
3312 014222 000177 177777
3313 014226 043413 143413
3314 014232 100004
3315

DIVF15: ; TEST DATA SET DIVF-15:
.WORD ZX1MP,M1 ; INITIAL AC FLOAT NUMBER
.WORD ZX1MN,M1 ; INITIAL MEM FLOAT NUMBER
.WORD ZX1MP,M1 ; EXPECTED FLOAT RESULT
.WORD 043413,143413 ; FPS: BEFORE, AFTER
.WORD 100004 ; FEC AFTER (0 = N/A)

3316
3317
3318
3319
3320
3321

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

3322 014234 000004
3323 014236 012705 014250
3324 014242 004737 035124
3325
3326 014246 000411
3327

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

BR TST211 ;;

3328 014250
3329 014250 077777 052525
3330 014254 003777 170360
3331 014260 034177 062134
3332 014264 047515 147502
3333 014270 100010
3334

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)

3335
3336
3337
3338
3339
3340

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

3341 014272 000004
3342 014274 012705 014306
3343 014300 004737 035124
3344
3345 014304 000411
3346

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

BR TST212 ;;

3347 014306
3348 014306 177777 052525
3349 014312 103777 170360

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

3350 014316 000000 000000
3351 014322 046411 046406
3352 014326 000000

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

3353
3354
3355
3356
3357
3358
3359

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

3360 014330 000004
3361 014332 012705 014344
3362 014336 004737 035124

TST212: SCOPE
MOV #DIVF20,R5 ; PTR TO TEST DATA SET
JSR PC,#DIVFT ; GO TEST
BR TST213 ;

3363
3364 014342 000411

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)

3365
3366 014344
3367 014344 100200 000000
3368 014350 077777 177777
3369 014354 140400 000000
3370 014360 047547 147550
3371 014364 100012

3372
3373
3374

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

3375
3376
3377
3378

TST213: SCOPE
MOV #DIVF21,R5 ; PTR TO TEST DATA SET
JSR PC,#DIVFT ; GO TEST
BR TST214 ;

3379 014366 000004
3380 014370 012705 014402
3381 014374 004737 035124

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)

3382
3383 014400 000411
3384

3385 014402
3386 014402 000200 000000
3387 014406 177777 177777
3388 014412 000000 000000
3389 014416 045453 045444
3390 014422 000000

3391
3392
3393

```

3394
3395
3396
3397
3398
3399 014424 000004
3400 014426 012705 014440
3401 014432 004737 035274
3402
3403 014436 000417
3404
3405 014440
3406 014440 000177 177777 177777
3407 014446 177777
3408 014450 077777 177777 177777
3409 014456 177777
3410 014460 000000 000000 000000
3411 014466 000000
3412 014470 047613 047604
3413 014474 000000
3414
3415
3416
3417
3418
3419
3420
3421 014476 000004
3422 014500 012705 014512
3423 014504 004737 035274
3424
3425 014510 000417
3426
3427 014512
3428 014512 034277 000000 000000
3429 014520 000000
3430 014522 040277 000000 000000
3431 014530 000000
3432 014532 034200 000000 000000
3433 014540 000000
3434 014542 047717 047700
3435 014546 000000
3436
3437
3438
3439
3440
3441
3442
3443 014550 000004
3444 014552 012705 014564
3445 014556 004737 035274
3446
3447 014562 000417
3448
3449 014564

```

```

*****
:TEST 214 TEST OF DIVD INSTR, DATA SET DIVD-1
:* ALL INTERRUPT ENABLES ON
:* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****
TST214: SCOPE
MOV #DIVD1,R5 ; PTR TO TEST DATA SET
JSR PC,@DIVDT ; GO TEST
BR TST215 ;;
DIVD1: ; TEST DATA SET DIVD-1:
.WORD ZXIMP,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 047613,047604 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
:TEST 215 TEST OF DIVD INSTR, DATA SET DIVD-2
:* ALL INTERRUPT ENABLES ON
:* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
TST215: SCOPE
MOV #DIVD2,R5 ; PTR TO TEST DATA SET
JSR PC,@DIVDT ; GO TEST
BR TST216 ;;
DIVD2: ; TEST DATA SET DIVD-2:
.WORD 034277,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD 040277,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 034200,0,0,0 ; EXPECTED FLOAT RESULT
.WORD 047717,047700 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
:TEST 216 TEST OF DIVD INSTR, DATA SET DIVD-3
:* ALL INTERRUPT ENABLES ON
:* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST216: SCOPE
MOV #DIVD3,R5 ; PTR TO TEST DATA SET
JSR PC,@DIVDT ; GO TEST
BR TST217 ;;
DIVD3: ; TEST DATA SET DIVD-3:

```


3450	014564	134277	000000	000000	.WORD	134277,0,0,0	; INITIAL AC FLOAT NUMBER
3451	014572	000000					
3452	014574	040277	000000	000000	.WORD	040277,0,0,0	; INITIAL MEM FLOAT NUMBER
3453	014602	000000					
3454	014604	134200	000000	000000	.WORD	134200,0,0,0	; EXPECTED FLOAT RESULT
3455	014612	000000					
3456	014614	047647	047650		.WORD	047647,047650	; FPS: BEFORE, AFTER
3457	014620	000000			.WORD	NA	; FEC AFTER (0 = N/A)

```

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

```

3465	014622	000004			TST217: SCOPE		
3466	014624	012705	014636		MOV	#DIVD4,RS	; PTR TO TEST DATA SET
3467	014630	004737	035274		JSR	PC,#DIVDT	; GO TEST
3468							
3469	014634	000417			BR	TST220	::

3471	014636				DIVD4: ; TEST DATA SET DIVD-4:		
3472	014636	134300	000000	000000	.WORD	134300,0,0,1	; INITIAL AC FLOAT NUMBER
3473	014644	000001					
3474	014646	140300	000000	000000	.WORD	140300,0,0,0	; INITIAL MEM FLOAT NUMBER
3475	014654	000000					
3476	014656	034200	000000	000000	.WORD	034200,0,0,0	; EXPECTED FLOAT RESULT
3477	014664	000000					
3478	014666	047757	047740		.WORD	047757,047740	; FPS: BEFORE, AFTER
3479	014672	000000			.WORD	NA	; FEC AFTER (0 = N/A)

```

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

```

3487	014674	000004			TST220: SCOPE		
3488	014676	012705	014710		MOV	#DIVD5,RS	; PTR TO TEST DATA SET
3489	014702	004737	035274		JSR	PC,#DIVDT	; GO TEST
3490							
3491	014706	000417			BR	TST221	::

3493	014710				DIVD5: ; TEST DATA SET DIVD-5:		
3494	014710	034300	000000	000000	.WORD	034300,0,0,1	; INITIAL AC FLOAT NUMBER
3495	014716	000001					
3496	014720	140300	000000	000000	.WORD	140300,0,0,0	; INITIAL MEM FLOAT NUMBER
3497	014726	000000					
3498	014730	134200	000000	000000	.WORD	134200,0,0,1	; EXPECTED FLOAT RESULT
3499	014736	000001					
3500	014740	047607	047610		.WORD	047607,047610	; FPS: BEFORE, AFTER
3501	014744	000000			.WORD	NA	; FEC AFTER (0 = N/A)

```

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

```

3502
3503
3504
3505

3506
3507
3508
3509
3510
3511
3512
3513
3514
3515
3516
3517
3518
3519
3520
3521
3522
3523
3524
3525
3526
3527
3528
3529
3530
3531
3532
3533
3534
3535
3536
3537
3538
3539
3540
3541
3542
3543
3544
3545
3546
3547
3548
3549
3550
3551
3552
3553
3554
3555
3556
3557
3558
3559
3560
3561

014746 000004
014750 012705 014762
014754 004737 035274

014760 000417

014762
014762 100400 000000 000000
014770 000000
014772 000500 000000 000000
015000 000000
015002 140052 125252 125252
015010 125252
015012 047647 047650
015016 000000

015020 000004
015022 012705 015034
015026 004737 035274

015032 000417

015034
015034 100400 000000 000000
015042 000000
015044 000500 000000 000000
015052 000000
015054 140052 125252 125252
015062 125252
015064 047607 047610
015070 000000

015072 000004
015074 012705 015106
015100 004737 035274

015104 000417

015106
015106 170360 170360 170360
015114 170360

```

: * ALL INTERRUPT ENABLES ON
: * LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
: *
: *****
TST221: 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
: *
: *****
TST222: 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
: *
: *****
TST223: SCOPE
MOV #DIVD10,R5 ; PTR TO TEST DATA SET
JSR PC,@DIVDT ; GO TEST

BR TST224 ;;

DIVD10: ; TEST DATA SET DIVD-10:
.WORD ALT4N,ALT4N,ALT4N,ALT4N ; INITIAL AC FLOAT NUMBER

```


3562	015116	170360	170360	170360	.WORD	ALT4N,ALT4N,ALT4N,ALT4N ; INITIAL MEM FLOAT NUMBER
3563	015124	170360				
3564	015126	040200	000000	000000	.WORD	F1P,0,0,0 ; EXPECTED FLOAT RESULT
3565	015134	000000				
3566	015136	047717	047700		.WORD	047717,047700 ; FPS: BEFORE AFTER
3567	015142	000000			.WORD	NA ; FEC AFTER (0 = N/A)

```

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

```

3575	015144	000004			TST224: SCOPE	
3576	015146	012705	015160		MOV	#DIVD11,RS ; PTR TO TEST DATA SET
3577	015152	004737	035274		JSR	PC,#DIVDT ; GO TEST
3578						
3579	015156	000417			BR	TST225 ;;

3581	015160				DIVD11: ; TEST DATA SET DIVD-11:	
3582	015160	070200	000000	000000	.WORD	070200,0,0,0 ; INITIAL AC FLOAT NUMBER
3583	015166	000000				
3584	015170	050400	000000	000000	.WORD	050400,0,0,0 ; INITIAL MEM FLOAT NUMBER
3585	015176	000000				
3586	015200	060000	000000	000000	.WORD	060000,0,0,0 ; EXPECTED FLOAT RESULT
3587	015206	000000				
3588	015210	047657	047640		.WORD	047657,047640 ; FPS: BEFORE AFTER
3589	015214	000000			.WORD	NA ; FEC AFTER (0 = N/A)

```

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

```

3597	015216	000004			TST225: SCOPE	
3598	015220	012705	015232		MOV	#DIVD12,RS ; PTR TO TEST DATA SET
3599	015224	004737	035274		JSR	PC,#DIVDT ; GO TEST
3600						
3601	015230	000417			BR	TST226 ;;

3603	015232				DIVD12: ; TEST DATA SET DIVD-12:	
3604	015232	125252	125252	125252	.WORD	ALTN,ALTN,ALTN,ALTN ; INITIAL AC FLOAT NUMBER
3605	015240	125252				
3606	015242	000000	000000	000000	.WORD	0,0,0,0 ; INITIAL MEM FLOAT NUMBER
3607	015250	000000				
3608	015252	125252	125252	125252	.WORD	ALTN,ALTN,ALTN,ALTN ; EXPECTED FLOAT RESULT
3609	015260	125252				
3610	015262	047707	147707		.WORD	047707,147707 ; FPS: BEFORE AFTER
3611	015266	100004			.WORD	100004 ; FEC AFTER (0 = N/A)

```

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

```

3617

```

3618
3619 015270 000004
3620 015272 012705 015304
3621 015276 004737 035274
3622
3623 015302 000417
3624
3625 015304
3626 015304 000177 177777 177777
3627 015312 177777
3628 015314 100177 177777 177777
3629 015322 177777
3630 015324 000177 177777 177777
3631 015332 177777
3632 015334 047643 147643
3633 015340 100014
3634
3635
3636
3637
3638
3639
3640
3641 015342 000004
3642 015344 012705 015356
3643 015350 004737 035274
3644
3645 015354 000417
3646
3647 015356
3648 015356 000177 177777 177777
3649 015364 177777
3650 015366 100177 177777 177777
3651 015374 177777
3652 015376 000177 177777 177777
3653 015404 177777
3654 015406 043643 143643
3655 015412 100004
3656
3657
3658
3659
3660
3661
3662
3663 015414 000004
3664 015416 012705 015430
3665 015422 004737 035274
3666
3667 015426 000417
3668
3669 015430
3670 015430 052525 052525 052525
3671 015436 052525
3672 015440 000200 000000 000000
3673 015446 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 ZX1MP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD ZX1MN,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD ZX1MP,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 ZX1MP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD ZX1MN,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD ZX1MP,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

```


F08

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 75
T230 TEST OF DIVD INSTR, DATA SET DIVD-15

SEQ 0242

3674	015450	012525	052525	052525	.WORD	012525,ALTP,ALTP,ALTP	; EXPECTED FLOAT RESULT
3675	015456	052525					
3676	015460	047615	147602		.WORD	047615,147602	; FPS: BEFORE, AFTER
3677	015464	100010			.WORD	100010	; FEC AFTER (0 = N/A)

```

3678
3679
3680
3681
3682
3683
3684
3685
3686
3687
3688
3689
3690
3691
3692
3693
3694
3695
3696
3697
3698
3699
3700
3701
3702
3703
3704
3705
3706

```

```

*****
: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,R5 ; 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 )

```

```

3707
3708
3709
3710
3711
3712
3713
3714
3715
3716
3717
3718
3719
3720
3721
3722
3723
3724
3725
3726
3727
3728
3729

```

```

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

```

```

TST232: SCOPE
MOV #DIVD17,R5 ; 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 )

```

```

3729
3728
3727
3726
3725
3724
3723
3722
3721
3720
3719
3718
3717
3716
3715
3714
3713
3712
3711
3710
3709
3708
3707
3706
3705
3704
3703
3702
3701
3700

```

```

*****
: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
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 76
T233 TEST OF DIVD INSTR, DATA SET DIVD-20

SEQ 0243

3730	015614	012705	015626
3731	015620	004737	035274
3732			
3733	015624	000417	
3734			
3735	015626		
3736	015626	100200	177777 125252
3737	015634	000000	
3738	015636	177777	177777 177777
3739	015644	177777	
3740	015646	000000	000000 000000
3741	015654	000000	
3742	015656	045713	045704
3743	015662	000000	
3744			
3745			

MOV #DIVD20,R5 ; PTR TO TEST DATA SET
JSR PC,#DIVDT ; 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)

H08

FPU ADVANCED INSTR TESTS
DQFPBB.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 77
T234 TEST OF MODF(2 ACC) INSTR, DATA SET MD2F-1

SEQ 0244

3746			
3747			
3748			
3749			
3750			
3751	015664	000004	
3752	015666	012705	015700
3753	015672	004737	035464
3754			
3755	015676	000413	
3756			
3757	015700		
3758	015700	000000	000000
3759	015704	000000	000000
3760	015710	000000	000000
3761	015714	000000	000000
3762	015720	047513	047504
3763	015724	000000	
3764			
3765			
3766			
3767			
3768			
3769			
3770			
3771	015726	000004	
3772	015730	012705	015742
3773	015734	004737	035464
3774			
3775	015740	000413	
3776			
3777	015742		
3778	015742	000177	177777
3779	015746	077777	177777
3780	015752	000000	000000
3781	015756	000000	000000
3782	015762	047553	047544
3783	015766	000000	
3784			
3785			
3786			
3787			
3788			
3789			
3790			
3791	015770	000004	
3792	015772	012705	016004
3793	015776	004737	035464
3794			
3795	016002	000413	
3796			
3797	016004		
3798	016004	177777	177777
3799	016010	100177	177777
3800	016014	000000	000000
3801	016020	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,#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,#MD2FT ; GO TEST
BR TST236 ;;

```

```

MD2F2: ; TEST DATA SET MD2F-2:
.WORD ZX1MP,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,#MD2FT ; GO TEST
BR TST237 ;;

```

```

MD2F3: ; TEST DATA SET MD2F-3:
.WORD LGN,M1 ; INITIAL AC FLOAT NUMBER
.WORD ZX1MN,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD 0,0 ; EXPECTED INTEGER-PART FLOAT RESULT

```

3802 016024 043413 043404
3803 016030 000000

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

3804
3805
3806
3807
3808
3809
3810

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

3811 016032 000004
3812 016034 012705 016046
3813 016040 004737 035464

TST237: SCOPE
MOV #MD2F4,R5 ; PTR TO TEST DATA SET
JSR PC, @MD2FT ; GO TEST

3814
3815 016044 000413

BR TST240 ;

3816
3817 016046
3818 016046 177777 177777
3819 016052 100177 177777
3820 016056 177777 177777
3821 016062 052525 177777
3822 016066 047447 147447
3823 016072 100014

MD2F4: ; TEST DATA SET MD2F-4:
:WORD LGN,M1 ; INITIAL AC FLOAT NUMBER
:WORD ZX1FN,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)

3824
3825
3826
3827
3828
3829
3830

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

3831 016074 000004
3832 016076 012705 016110
3833 016102 004737 035464

TST240: SCOPE
MOV #MD2F5,R5 ; PTR TO TEST DATA SET
JSR PC, @MD2FT ; GO TEST

3834
3835 016106 000413

BR TST241 ;

3836
3837 016110
3838 016110 042177 000000
3839 016114 140200 000000
3840 016120 000000 000000
3841 016124 142177 000000
3842 016130 047553 047544
3843 016134 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)

3844
3845
3846
3847
3848
3849
3850

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

3851 016136 000004
3852 016140 012705 016152
3853 016144 004737 035464

TST241: SCOPE
MOV #MD2F6,R5 ; PTR TO TEST DATA SET
JSR PC, @MD2FT ; GO TEST

3854
3855 016150 000413

BR TST242 ;

3856
3857 016152

MD2F6: ; TEST DATA SET MD2F-6:

J08

FPU ADVANCED INSTR TESTS
DGFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 79
T241 TEST OF MODF(2 ACC) INSTR, DATA SET MD2F-6

SEQ 0246

3858	016152	040200	000000
3859	016156	140177	177777
3860	016162	140177	177777
3861	016166	000000	000000
3862	016172	047507	047510
3863	016176	000000	

```
.WORD F1P,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 )
```

3864			
3865			
3866			
3867			
3868			
3869			
3870			

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

3871	016200	000004	
3872	016202	012705	016214
3873	016206	004737	035464
3874			
3875	016212	000413	
3876			

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

3877	016214		
3878	016214	142176	077600
3879	016220	140200	000000
3880	016224	037777	000000
3881	016230	042176	000000
3882	016234	047457	047440
3883	016240	000000	

```
MD2F7: ; TEST DATA SET MD2F-7:
.WORD 142176,077600 ; INITIAL AC FLOAT NUMBER
.WORD F1N,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 )
```

3884			
3885			
3886			
3887			
3888			
3889			
3890			

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

3891	016242	000004	
3892	016244	012705	016256
3893	016250	004737	035464
3894			
3895	016254	000413	
3896			

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

3897	016256		
3898	016256	042177	100000
3899	016262	040200	000000
3900	016266	040000	000000
3901	016272	042177	000000
3902	016276	047417	047400
3903	016302	000000	

```
MD2F10: ; TEST DATA SET MD2F-10:
.WORD 042177,M0 ; INITIAL AC FLOAT NUMBER
.WORD F1P,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 )
```

3904			
3905			
3906			
3907			
3908			
3909			
3910			

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

3911	016304	000004	
3912	016306	012705	016320
3913	016312	004737	035464

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

```

3914
3915 016316 000413
3916
3917 016320
3918 016320 140200 000000
3919 016324 040377 177777
3920 016330 140177 177776
3921 016334 140200 000000
3922 016340 047547 047550
3923 016344 000000

```

```

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 )

```

```

3924
3925
3926
3927
3928
3929
3930

```

```

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

```

```

3931 016346 000004
3932 016350 012705 016362
3933 016354 004737 035464

```

```

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

```

```

3934
3935 016360 000413

```

```

BR TST246 ;;

```

```

3936
3937 016362
3938 016362 060452 125252
3939 016366 021700 000000
3940 016372 040177 177400
3941 016376 042177 000000
3942 016402 047517 047500
3943 016406 000000

```

```

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 )

```

```

3944
3945
3946
3947
3948
3949
3950

```

```

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

```

```

3951 016410 000004
3952 016412 012705 016424
3953 016416 004737 035464

```

```

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

```

```

3954
3955 016422 000413

```

```

BR TST247 ;;

```

```

3956
3957 016424
3958 016424 041000 000001
3959 016430 141377 177776
3960 016434 140177 177777
3961 016440 142177 000000
3962 016444 047547 047550
3963 016450 000000

```

```

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 )

```

```

3964
3965
3966
3967
3968
3969

```

```

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

```



```

3970
3971 016452 000004
3972 016454 012705 016466
3973 016460 004737 035464
3974
3975 016464 000413
3976
3977 016466
3978 016466 041000 000001
3979 016472 141377 177776
3980 016476 140200 000000
3981 016502 142177 000000
3982 016506 047507 047510
3983 016512 000000
3984
3985
3986
3987
3988
3989
3990
3991 016514 000004
3992 016516 012705 016530
3993 016522 004737 035464
3994
3995 016526 000413
3996
3997 016530
3998 016530 077600 000000
3999 016534 044452 125252
4000 016540 000000 000000
4001 016544 004052 125252
4002 016550 047411 147406
4003 016554 100010
4004
4005
4006
4007
4008
4009
4010
4011 016556 000004
4012 016560 012705 016572
4013 016564 004737 035464
4014
4015 016570 000413
4016
4017 016572
4018 016572 077600 000000
4019 016576 044452 125252
4020 016602 000000 000000
4021 016606 000000 000000
4022 016612 046411 046406
4023 016616 000000
4024
4025

```

```

*****
†ST247: SCOPE
MOV      #MD2F14,R5      ; PTR TO TEST DATA SET
JSR      PC,2#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
*****

```

```

†ST250: SCOPE
MOV      #MD2F15,R5      ; PTR TO TEST DATA SET
JSR      PC,2#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
*****

```

```

†ST251: SCOPE
MOV      #MD2F16,R5      ; PTR TO TEST DATA SET
JSR      PC,2#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 )

```

M08

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 82
T252 TEST OF MODF(2 ACC) INSTR, DATA SET MD2F-17

SEG 0249

4026			
4027			
4028			
4029			
4030			
4031	016620	000004	
4032	016622	012705	016634
4033	016626	004737	035464
4034			
4035	016632	000413	
4036			
4037	016634		
4038	016634	001577	177777
4039	016640	101000	000000
4040	016644	142377	177777
4041	016650	000000	000000
4042	016654	047547	147550
4043	016660	100012	
4044			
4045			
4046			
4047			
4048			
4049			
4050			
4051	016662	000004	
4052	016664	012705	016676
4053	016670	004737	035464
4054			
4055	016674	000413	
4056			
4057	016676		
4058	016676	001577	177777
4059	016702	101000	000000
4060	016706	000000	000000
4061	016712	000000	000000
4062	016716	045553	045544
4063	016722	000000	
4064			
4065			
4066			

```

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

```

```

TST252: 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
*****

```

```

TST253: 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 )

```



```

4067
4068
4069
4070
4071
4072 016724 000004
4073 016726 012705 016740
4074 016732 004737 035666
4075
4076 016736 000423
4077
4078 016740
4079 016740 000000 000000 000000
4080 016746 000000
4081 016750 000000 000000 000000
4082 016756 000000
4083 016760 000000 000000 000000
4084 016766 000000
4085 016770 000000 000000 000000
4086 016776 000000
4087 017000 047653 047644
4088 017004 000000
4089
4090
4091
4092
4093
4094
4095
4096 017006 000004
4097 017010 012705 017022
4098 017014 004737 035666
4099
4100 017020 000423
4101
4102 017022
4103 017022 000177 177777 177777
4104 017030 177777
4105 017032 177777 177777 177777
4106 017040 177777
4107 017042 000000 000000 000000
4108 017050 000000
4109 017052 000000 000000 000000
4110 017060 000000
4111 017062 047713 047704
4112 017066 000000
4113
4114
4115
4116
4117
4118
4119
4120 017070 000004
4121 017072 012705 017104
4122 017076 004737 035666

```

```

*****
: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

```

```

4123
4124 017102 000423 BR TST257 ;;
4125
4126 017104 MD2D3: ; TEST DATA SET MD2D-3:
4127 017104 077777 177777 177777 .WORD LGP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
4128 017112 177777
4129 017114 100177 177777 177777 .WORD ZX1MN,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
4130 017122 177777
4131 017124 000000 000000 000000 .WORD 0,0,0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
4132 017132 000000
4133 017134 000000 000000 000000 .WORD 0,0,0,0 ; EXPECTED INTEGER-PART FLOAT RESULT
4134 017142 000000
4135 017144 043653 043644 .WORD 043653,043644 ; FPS: BEFORE, AFTER
4136 017150 000000 .WORD NA ; FEC AFTER ( 0 = N/A )
4137
4138
4139
4140 :*****
4141 :*TEST 257 TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-4
4142 :* ALL INTERRUPT ENABLES ON
4143 :* LONG FLOAT, LONG INTEGER, ROUND MODES
4144 :*****
4144 017152 000004 TST257: SCOPE
4145 017154 012705 017166 MOV #MD2D4,R5 ; PTR TO TEST DATA SET
4146 017160 004737 035666 JSR PC,#MD2DT ; GO TEST
4147
4148 017164 000423 BR TST260 ;;
4149
4150 MD2D4: ; TEST DATA SET MD2D-4:
4151 017166 077777 177777 177777 .WORD LGP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
4152 017174 177777
4153 017176 100177 177777 177777 .WORD ZX1MN,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
4154 017204 177777
4155 017206 077777 177777 177777 .WORD LGP,M1,M1,M1 ; EXPECTED FRACTION-PART FLOAT RESULT
4156 017214 177777
4157 017216 052525 177777 125252 .WORD ALTP,M1,ALTN,0 ; EXPECTED INTEGER-PART FLOAT RESULT
4158 017224 000000
4159 017226 047713 147713 .WORD 047713,147713 ; FPS: BEFORE, AFTER
4160 017232 100014 .WORD 100014 ; FEC AFTER ( 0 = N/A )
4161
4162
4163 :*****
4164 :*TEST 260 TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-5
4165 :* ALL INTERRUPT ENABLES ON
4166 :* LONG FLOAT, SHORT INTEGER, ROUND MODES
4167 :*****
4168 017234 000004 TST260: SCOPE
4169 017236 012705 017250 MOV #MD2D5,R5 ; PTR TO TEST DATA SET
4170 017242 004737 035666 JSR PC,#MD2DT ; GO TEST
4171
4172 017246 000423 BR TST261 ;;
4173
4174 MD2D5: ; TEST DATA SET MD2D-5:
4175 017250 042177 000000 000000 .WORD 042177,0,0,0 ; INITIAL AC FLOAT NUMBER
4176 017256 000000
4177 017260 040200 000000 000000 .WORD F1P,0,0,0 ; INITIAL MEM FLOAT NUMBER
4178 017266 000000

```


4179	017270	000000	000000	000000	.WORD	0,0,0,0	; EXPECTED FRACTION-PART FLOAT RESULT
4180	017276	000000					
4181	017300	042177	000000	000000	.WORD	042177,0,0,0	; EXPECTED INTEGER-PART FLOAT RESULT
4182	017306	000000					
4183	017310	047613	047604		.WORD	047613,047604	; FPS: BEFORE, AFTER
4184	017314	000000			.WORD	NA	; FEC AFTER (0 = N/A)

4185
4186
4187
4188
4189
4190
4191
4192
4193
4194
4195
4196
4197
4198
4199
4200
4201
4202
4203
4204
4205
4206
4207
4208
4209

```

*****
:TEST 261      TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-6
:             ALL INTERRUPT ENABLES ON
:             LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

TST261: SCOPE
        MOV     #MD2D6,R5      ; PTR TO TEST DATA SET
        JSR    PC,@#MD2DT     ; GO TEST
        BR     TST262        ;;

MD2D6:  ; TEST DATA SET MD2D-6:
        .WORD  FIN,0,0,0      ; INITIAL AC FLOAT NUMBER
        .WORD  040177,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
        .WORD  140177,M1,M1,M1 ; EXPECTED FRACTION-PART FLOAT RESULT
        .WORD  0,0,0,0       ; EXPECTED INTEGER-PART FLOAT RESULT
        .WORD  047747,047750 ; FPS: BEFORE, AFTER
        .WORD  NA            ; FEC AFTER ( 0 = N/A )

```

4210
4211
4212
4213
4214
4215
4216
4217
4218
4219
4220
4221
4222
4223
4224
4225
4226
4227
4228
4229
4230
4231
4232
4233
4234

```

*****
:TEST 262      TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-7
:             ALL INTERRUPT ENABLES ON
:             LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

TST262: SCOPE
        MOV     #MD2D7,R5      ; PTR TO TEST DATA SET
        JSR    PC,@#MD2DT     ; GO TEST
        BR     TST263        ;;

MD2D7:  ; TEST DATA SET MD2D-7:
        .WORD  042176,077600,0,0 ; INITIAL AC FLOAT NUMBER
        .WORD  FIN,0,0,0      ; INITIAL MEM FLOAT NUMBER
        .WORD  137777,0,0,0   ; EXPECTED FRACTION-PART FLOAT RESULT
        .WORD  142176,0,0,0   ; EXPECTED INTEGER-PART FLOAT RESULT
        .WORD  047607,047610 ; FPS: BEFORE, AFTER
        .WORD  NA            ; FEC AFTER ( 0 = N/A )

```

```

4235
4236
4237
4238
4239
4240 017462 000004
4241 017464 012705 017476
4242 017470 004737 035666
4243
4244 017474 000423
4245
4246 017476
4247 017476 142177 100000 000000
4248 017504 000000
4249 017506 040200 000000 000000
4250 017514 000000
4251 017516 140000 000000 000000
4252 017524 000000
4253 017526 142177 000000 000000
4254 017534 000000
4255 017536 047747 047750
4256 017542 000000
4257
4258
4259
4260
4261
4262
4263
4264 017544 000004
4265 017546 012705 017560
4266 017552 004737 035666
4267
4268 017556 000423
4269
4270 017560
4271 017560 140200 000000 000000
4272 017566 000000
4273 017570 140377 177777 177777
4274 017576 177777
4275 017600 040177 177777 177777
4276 017606 177776
4277 017610 040200 000000 000000
4278 017616 000000
4279 017620 047617 047600
4280 017624 000000
4281
4282
4283
4284
4285
4286
4287
4288 017626 000004
4289 017630 012705 017642
4290 017634 004737 035666

```

```

*****
*TEST 263 TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-10
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****
TST263: SCOPE
MOV #MD2D10,R5 ; PTR TO TEST DATA SET
JSR PC,#MD2DT ; GO TEST

BR TST264 ;;

MD2D10: ; TEST DATA SET MD2D-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 142177,0,0,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047747,047750 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 264 TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-11
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****
TST264: SCOPE
MOV #MD2D11,R5 ; PTR TO TEST DATA SET
JSR PC,#MD2DT ; GO TEST

BR TST265 ;;

MD2D11: ; TEST DATA SET MD2D-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 F1P,0,0,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047617,047600 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
*TEST 265 TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-12
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****
TST265: SCOPE
MOV #MD2D12,R5 ; PTR TO TEST DATA SET
JSR PC,#MD2DT ; GO TEST

```



```

4291
4292 017640 000423 BR TST266 ;;
4293
4294 017642 MD2D12: ; TEST DATA SET MD2D-12:
4295 017642 167452 125252 125252 .WORD 167452,AN,AN,AN ; INITIAL AC FLOAT NUMBER
4296 017650 125252 .WORD 112700,0,0,0 ; INITIAL MEM FLOAT NUMBER
4297 017652 112700 000000 000000 .WORD 040177,M1,M1,UB ; EXPECTED FRACTION-PART FLOAT RESULT
4298 017660 000000 .WORD 042177,0,0,0 ; EXPECTED INTEGER-PART FLOAT RESULT
4299 017662 040177 177777 177777 .WORD 047757,047740 ; FPS: BEFORE, AFTER
4300 017670 177400 .WORD NA ; FEC AFTER ( 0 = N/A )
4301 017672 042177 000000 000000
4302 017700 000000
4303 017702 047757 047740
4304 017706 000000
4305
4306
4307

```

```

*****
;TEST 266 TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-13
;
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

4308
4309
4310
4311
4312 017710 000004 TST266: SCOPE
4313 017712 012705 017724 MOV #MD2D13,R5 ; PTR TO TEST DATA SET
4314 017716 004737 035666 JSR PC,#MD2DT ; GO TEST
4315
4316 017722 000423 BR TST267 ;;
4317
4318 MD2D13: ; TEST DATA SET MD2D-13:
4319 017724 041000 000000 000000 .WORD 041000,0,0,1 ; INITIAL AC FLOAT NUMBER
4320 017732 000001 .WORD 037577,M1,M1,M2 ; INITIAL MEM FLOAT NUMBER
4321 017734 037577 177777 177777 .WORD 040177,M1,M1,M1 ; EXPECTED FRACTION-PART FLOAT RESULT
4322 017742 177776 .WORD 040200,0,0,0 ; EXPECTED INTEGER-PART FLOAT RESULT
4323 017744 040177 177777 177777 .WORD 047657,047640 ; FPS: BEFORE, AFTER
4324 017752 177777 .WORD NA ; FEC AFTER ( 0 = N/A )
4325 017754 040200 000000 000000
4326 017762 000000
4327 017764 047657 047640
4328 017770 000000
4329
4330

```

```

*****
;TEST 267 TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-14
;
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

4331
4332
4333
4334
4335
4336 017772 000004 TST267: SCOPE
4337 017774 012705 020006 MOV #MD2D14,R5 ; PTR TO TEST DATA SET
4338 020000 004737 035666 JSR PC,#MD2DT ; GO TEST
4339
4340 020004 000423 BR TST270 ;;
4341
4342 MD2D14: ; TEST DATA SET MD2D-14:
4343 020006 041000 000000 000000 .WORD 041000,0,0,1 ; INITIAL AC FLOAT NUMBER
4344 020014 000001 .WORD 037577,M1,M1,M2 ; INITIAL MEM FLOAT NUMBER
4345 020016 037577 177777 177777
4346 020024 177776

```

4347	020026	040200	000000	000000	.WORD	040200,0,0,0	; EXPECTED FRACTION-PART FLOAT RESULT
4348	020034	000000					
4349	020036	040200	000000	000000	.WORD	040200,0,0,0	; EXPECTED INTEGER-PART FLOAT RESULT
4350	020044	000000					
4351	020046	047617	047600		.WORD	047617,047600	; FPS: BEFORE AFTER
4352	020052	000000			.WORD	NA	; FEC AFTER (0 = N/A)

4353
4354

```

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

```

4360	020054	000004					
4361	020056	012705	020070				
4362	020062	004737	035666				

```

TST270: SCOPE
MOV #MD2D15,R5 ; PTR TO TEST DATA SET
JSR PC,#MD2DT ; GO TEST

```

4363							
4364	020066	000423			BR	TST271	::

```

MD2D15: ; 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 )

```

4365							
4366	020070						
4367	020070	140452	125252	125252			
4368	020076	125252					
4369	020100	077600	000000	000000			
4370	020106	000000					
4371	020110	000000	000000	000000			
4372	020116	000000					
4373	020120	100052	125252	125252			
4374	020126	125252					
4375	020130	047651	147646				
4376	020134	100010					

4377
4378

```

*****
: 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
*****

```

4379							
4380							
4381							
4382							
4383							

```

TST271: SCOPE
MOV #MD2D16,R5 ; PTR TO TEST DATA SET
JSR PC,#MD2DT ; GO TEST

```

4384	020136	000004					
4385	020140	012705	020152				
4386	020144	004737	035666				

4387							
4388	020150	000423			BR	TST272	::

```

MD2D16: ; 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 )

```

4389							
4390	020152						
4391	020152	140452	125252	125252			
4392	020160	125252					
4393	020162	077600	000000	000000			
4394	020170	000000					
4395	020172	000000	000000	000000			
4396	020200	000000					
4397	020202	000000	000000	000000			
4398	020210	000000					
4399	020212	046751	046746				
4400	020216	000000					

4401
4402


```

4403
4404
4405
4406
4407
4408 020220 000004
4409 020222 012705 020234
4410 020226 004737 035666
4411
4412 020232 000423
4413
4414 020234
4415 020234 101577 177777 177777
4416 020242 177777
4417 020244 101000 000000 000000
4418 020252 000000
4419 020254 042377 177777 177777
4420 020262 177777
4421 020264 000000 000000 000000
4422 020272 000000
4423 020274 047617 147600
4424 020300 100012
4425
4426
4427
4428
4429
4430
4431
4432 020302 000004
4433 020304 012705 020316
4434 020310 004737 035666
4435
4436 020314 000423
4437
4438 020316
4439 020316 101577 177777 177777
4440 020324 177777
4441 020326 101000 000000 000000
4442 020334 000000
4443 020336 000000 000000 000000
4444 020344 000000
4445 020346 000000 000000 000000
4446 020354 000000
4447 020356 045713 045704
4448 020362 000000
4449
4450

```

```

*****
*TEST 272 TEST OF MODD(2 ACC) INSTR, DATA SET MD2D-17
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****
TST272: SCOPE
MOV #MD2D17,R5 ; PTR TO TEST DATA SET
JSR PC,#MD2DT ; GO TEST
BR TST273 ;;

MD2D17: ; TEST DATA SET MD2D-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 MODD(2 ACC) INSTR, DATA SET MD2D-20
* UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
TST273: SCOPE
MOV #MD2D20,R5 ; PTR TO TEST DATA SET
JSR PC,#MD2DT ; GO TEST
BR TST274 ;;

MD2D20: ; TEST DATA SET MD2D-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 )

```

```

4451
4452
4453
4454
4455
4456 020364 000004
4457 020366 012705 020400
4458 020372 004737 036130
4459
4460 020376 000413
4461
4462 020400
4463 020400 000000 000000
4464 020404 000000 000000
4465 020410 000000 000000
4466 020414 052525 177777
4467 020420 047513 047504
4468 020424 000000
4469
4470
4471
4472
4473
4474
4475
4476 020426 000004
4477 020430 012705 020442
4478 020434 004737 036130
4479
4480 020440 000413
4481
4482 020442
4483 020442 000177 177777
4484 020446 077777 177777
4485 020452 000000 000000
4486 020456 052525 177777
4487 020462 047553 047544
4488 020466 000000
4489
4490
4491
4492
4493
4494
4495
4496 020470 000004
4497 020472 012705 020504
4498 020476 004737 036130
4499
4500 020502 000413
4501
4502 020504
4503 020504 177777 177777
4504 020510 100177 177777
4505 020514 000000 000000
4506 020520 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,3#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,3#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,3#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

```


4507 020524 043413 043404
4508 020530 000000

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

4509
4510
4511
4512
4513
4514
4515

*TEST 277 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-4
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES

4516 020532 000004
4517 020534 012705 020546
4518 020540 004737 036130

TST277: SCOPE
MOV #MDIF4,R5 ; PTR TO TEST DATA SET
JSR PC,@MDIFT ; GO TEST

4519
4520 020544 000413

BR TST300 ; ;

4521
4522 020546
4523 020546 177777 177777
4524 020552 100177 177777
4525 020556 177777 177777
4526 020562 052525 177777
4527 020566 047447 147447
4528 020572 100014

MDIF4: ; TEST DATA SET MDIF-4:
.WORD LGN,M1 ; INITIAL AC FLOAT NUMBER
.WORD ZX1FN,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)

4529
4530
4531
4532
4533
4534
4535

*TEST 300 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-5
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES

4536 020574 000004
4537 020576 012705 020610
4538 020602 004737 036130

TST300: SCOPE
MOV #MDIF5,R5 ; PTR TO TEST DATA SET
JSR PC,@MDIFT ; GO TEST

4539
4540 020606 000413

BR TST301 ; ;

4541
4542 020610
4543 020610 042177 000000
4544 020614 140200 000000
4545 020620 000000 000000
4546 020624 052525 177777
4547 020630 047553 047544
4548 020634 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)

4549
4550
4551
4552
4553
4554
4555

*TEST 301 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-6
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES

4556 020636 000004
4557 020640 012705 020652
4558 020644 004737 036130

TST301: SCOPE
MOV #MDIF6,R5 ; PTR TO TEST DATA SET
JSR PC,@MDIFT ; GO TEST

4559
4560 020650 000413

BR TST302 ; ;

4561
4562 020652

MDIF6: ; TEST DATA SET MDIF-6:

4563	020652	040200	000000
4564	020656	140177	177777
4565	020662	140177	177777
4566	020666	052525	177777
4567	020672	047507	047510
4568	020676	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 )
```

4569			
4570			
4571			
4572			
4573			
4574			
4575			

```
*****
:TEST 302 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-7
:*
:* ALL INTERRUPT ENABLES ON
:* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
```

4576	020700	000004	
4577	020702	012705	020714
4578	020706	004737	036130
4579			
4580	020712	000413	
4581			

```
TST302: SCOPE
MOV #MDIF7,R5 ; PTR TO TEST DATA SET
JSR PC,@MDIFT ; GO TEST
BR TST303 ;;
```

4582	020714		
4583	020714	142176	077600
4584	020720	140200	000000
4585	020724	037777	000000
4586	020730	052525	177777
4587	020734	047457	047440
4588	020740	000000	
4589			
4590			

```
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 )
```

4591			
4592			
4593			
4594			
4595			

```
*****
:TEST 303 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-10
:*
:* ALL INTERRUPT ENABLES ON
:* SHORT FLOAT, SHORT INTEGER, ROUND MODES
*****
```

4596	020742	000004	
4597	020744	012705	020756
4598	020750	004737	036130
4599			
4600	020754	000413	
4601			

```
TST303: SCOPE
MOV #MDIF10,R5 ; PTR TO TEST DATA SET
JSR PC,@MDIFT ; GO TEST
BR TST304 ;;
```

4602	020756		
4603	020756	042177	100000
4604	020762	040200	000000
4605	020766	040000	000000
4606	020772	052525	177777
4607	020776	047417	047400
4608	021002	000000	
4609			
4610			

```
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 )
```

4611			
4612			
4613			
4614			
4615			

```
*****
:TEST 304 TEST OF MODF(1 ACC) INSTR, DATA SET MDIF-11
:*
:* ALL INTERRUPT ENABLES ON
:* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****
```

4616	021004	000004	
4617	021006	012705	021020
4618	021012	004737	036130

```
TST304: SCOPE
MOV #MDIF11,R5 ; PTR TO TEST DATA SET
JSR PC,@MDIFT ; GO TEST
```


4619
 4620 021016 000413
 4621
 4622 021020
 4623 021020 140200 000000
 4624 021024 040377 177777
 4625 021030 140177 177776
 4626 021034 052525 177777
 4627 021040 047547 047550
 4628 021044 000000
 4629
 4630
 4631
 4632
 4633
 4634
 4635
 4636 021046 000004
 4637 021050 012705 021062
 4638 021054 004737 036130
 4639
 4640 021060 000413
 4641
 4642 021062
 4643 021062 060452 125252
 4644 021066 021700 000000
 4645 021072 040177 177400
 4646 021076 052525 177777
 4647 021102 047517 047500
 4648 021106 000000
 4649
 4650
 4651
 4652
 4653
 4654
 4655
 4656 021110 000004
 4657 021112 012705 021124
 4658 021116 004737 036130
 4659
 4660 021122 000413
 4661
 4662 021124
 4663 021124 041000 000001
 4664 021130 141377 177776
 4665 021134 140177 177777
 4666 021140 052525 177777
 4667 021144 047547 047550
 4668 021150 000000
 4669
 4670
 4671
 4672
 4673
 4674

```

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
  
```

```

4675
4676 021152 000004
4677 021154 012705 021166
4678 021160 004737 036130
4679
4680 021164 000413
4681
4682 021166
4683 021166 041000 000001
4684 021172 141377 177776
4685 021176 140200 000000
4686 021202 052525 177777
4687 021206 047507 047510
4688 021212 000000
4689
4690
4691
4692
4693
4694
4695
4696 021214 000004
4697 021216 012705 021230
4698 021222 004737 036130
4699
4700 021226 000413
4701
4702 021230
4703 021230 077600 000000
4704 021234 040452 125252
4705 021240 000000 000000
4706 021244 052525 177777
4707 021250 047411 147406
4708 021254 100010
4709
4710
4711
4712
4713
4714
4715
4716 021256 000004
4717 021260 012705 021272
4718 021264 004737 036130
4719
4720 021270 000413
4721
4722 021272
4723 021272 077600 000000
4724 021276 040452 125252
4725 021302 000000 000000
4726 021306 052525 177777
4727 021312 046411 046406
4728 021316 000000
4729
4730

```

```

*****
TST307: SCOPE
MOV      #MDIF14,R5      ; PTR TO TEST DATA SET
JSR      PC,@#MDIFT     ; 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
*****

```

```

TST310: SCOPE
MOV      #MDIF15,R5      ; PTR TO TEST DATA SET
JSR      PC,@#MDIFT     ; 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
*****

```

```

TST311: SCOPE
MOV      #MDIF16,R5      ; PTR TO TEST DATA SET
JSR      PC,@#MDIFT     ; 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 )

```


M09

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 95
T312 TEST OF MODF(1 ACC) INSTR, DATA SET MD1F-17

SEQ 0262

4731			
4732			
4733			
4734			
4735			
4736	021320	000004	
4737	021322	012705	021334
4738	021326	004737	036130
4739			
4740	021332	000413	
4741			
4742	021334		
4743	021334	001577	177777
4744	021340	101000	000000
4745	021344	142377	177777
4746	021350	052525	177777
4747	021354	047547	147550
4748	021360	100012	
4749			
4750			
4751			
4752			
4753			
4754			
4755			
4756	021362	000004	
4757	021364	012705	021376
4758	021370	004737	036130
4759			
4760	021374	000413	
4761			
4762	021376		
4763	021376	001577	177777
4764	021402	101000	000000
4765	021406	000000	000000
4766	021412	052525	177777
4767	021416	045553	045544
4768	021422	000000	
4769			
4770			
4771			

```

*****
:TEST 312 TEST OF MODF(1 ACC) INSTR, DATA SET MD1F-17
:*
:* ALL INTERRUPT ENABLES ON
:* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

TST312: SCOPE
MOV #MD1F17,R5 ; PTR TO TEST DATA SET
JSR PC,#MD1FT ; GO TEST
BR TST313 ;;

```

```

MD1F17: ; TEST DATA SET MD1F-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 MD1F-20
:*
:* UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
:* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

TST313: SCOPE
MOV #MD1F20,R5 ; PTR TO TEST DATA SET
JSR PC,#MD1FT ; GO TEST
BR TST314 ;;

```

```

MD1F20: ; TEST DATA SET MD1F-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 )

```

```

4772
4773
4774
4775
4776
4777 021424 000004
4778 021426 012705 021440
4779 021432 004737 036332
4780
4781 021436 000423
4782
4783 021440
4784 021440 000000 000000 000000
4785 021446 000000
4786 021450 000000 000000 000000
4787 021456 000000
4788 021460 000000 000000 000000
4789 021466 000000
4790 021470 052525 177777 125252
4791 021476 000000
4792 021500 047653 047644
4793 021504 000000
4794
4795
4796
4797
4798
4799
4800
4801 021506 000004
4802 021510 012705 021522
4803 021514 004737 036332
4804
4805 021520 000423
4806
4807 021522
4808 021522 000177 177777 177777
4809 021530 177777
4810 021532 177777 177777 177777
4811 021540 177777
4812 021542 000000 000000 000000
4813 021550 000000
4814 021552 052525 177777 125252
4815 021560 000000
4816 021562 047713 047704
4817 021566 000000
4818
4819
4820
4821
4822
4823
4824
4825 021570 000004
4826 021572 012705 021604
4827 021576 004737 036332

```

```

*****
*TEST 314 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-1
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST314: 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
*****
TST315: 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
*****
TST316: SCOPE
MOV #MD1D3,R5 ; PTR TO TEST DATA SET
JSR PC,@MD1DT ; GO TEST

```


B10

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 97
T316 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-3

SEQ 0264

```

4828
4829 021602 000423 BR TST317 ;;
4830
4831 021604 MD1D3: ; TEST DATA SET MD1D-3:
4832 021604 077777 177777 177777 .WORD LGP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
4833 021612 177777
4834 021614 100177 177777 177777 .WORD ZX1MN,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
4835 021622 177777
4836 021624 000000 000000 000000 .WORD 0,0,0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
4837 021632 000000
4838 021634 052525 177777 125252 .WORD AP,M1,AN,0 ; EXPECTED INTEGER-PART FLOAT RESULT
4839 021642 000000
4840 021644 043653 043644 .WORD 043653,043644 ; FPS: BEFORE, AFTER
4841 021650 000000 .WORD NA ; FEC AFTER ( 0 = N/A )
4842
4843
4844
4845
4846
4847
4848

```

```

*****
;TEST 317 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-4
;
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

4849 021652 000004
4850 021654 012705 021666
4851 021660 004737 036332
4852
4853 021664 000423 BR TST320 ;;
4854
4855 MD1D4: ; TEST DATA SET MD1D-4:
4856 021666 077777 177777 177777 .WORD LGP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
4857 021674 177777
4858 021676 100177 177777 177777 .WORD ZX1MN,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
4859 021704 177777
4860 021706 077777 177777 177777 .WORD LGP,M1,M1,M1 ; EXPECTED FRACTION-PART FLOAT RESULT
4861 021714 177777
4862 021716 052525 177777 125252 .WORD AP,M1,AN,0 ; EXPECTED INTEGER-PART FLOAT RESULT
4863 021724 000000
4864 021726 047713 147713 .WORD 047713,147713 ; FPS: BEFORE, AFTER
4865 021732 100014 .WORD 100014 ; FEC AFTER ( 0 = N/A )
4866
4867
4868

```

```

*****
;TEST 320 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-5
;
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

4869
4870
4871
4872
4873 021734 000004
4874 021736 012705 021750
4875 021742 004737 036332
4876
4877 021746 000423 BR TST321 ;;
4878
4879 MD1D5: ; TEST DATA SET MD1D-5:
4880 021750 042177 000000 000000 .WORD 042177,0,0,0 ; INITIAL AC FLOAT NUMBER
4881 021756 000000
4882 021760 040200 000000 000000 .WORD F1P,0,0,0 ; INITIAL MEM FLOAT NUMBER
4883 021766 000000

```

C10

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 98
T320 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-5

SEQ 0265

4884	021770	000000	000000	000000	.WORD	0,0,0,0	; EXPECTED FRACTION-PART FLOAT RESULT
4885	021776	000000					
4886	022000	052525	177777	125252	.WORD	AP,M1,AN,0	; EXPECTED INTEGER-PART FLOAT RESULT
4887	022006	000000					
4888	022010	047613	047604		.WORD	047613,047604	; FPS: BEFORE, AFTER
4889	022014	000000			.WORD	NA	; FEC AFTER (0 = N/A)

```

4890
4891
4892
4893
4894
4895
4896
4897
4898
4899
4900
4901
4902
4903
4904
4905
4906
4907
4908
4909
4910
4911
4912
4913
4914
4915
4916
4917
4918
4919
4920

```

```

*****
*TEST 321 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-6
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

TST321: SCOPE
MOV #MD1D6,R5 ; PTR TO TEST DATA SET
JSR PC,@MD1DT ; GO TEST
BR TST322 ;;

```

```

MD1D6: ; TEST DATA SET MD1D-6:
.WORD FIN,0,0,0 ; INITIAL AC FLOAT NUMBER
.WORD 040177,M1,M1,M1 ; INITIAL MEM FLOAT NUMBER
.WORD 140177,M1,M1,M1 ; 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 )

```

4897	022016	000004		
4898	022020	012705	022032	
4899	022024	004737	036332	
4901	022030	000423		
4903	022032			
4904	022032	140200	000000	000000
4905	022040	000000		
4906	022042	040177	177777	177777
4907	022050	177777		
4908	022052	140177	177777	177777
4909	022060	177777		
4910	022062	052525	177777	125252
4911	022070	000000		
4912	022072	047747	047750	
4913	022076	000000		

```

*****
*TEST 322 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-7
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

TST322: SCOPE
MOV #MD1D7,R5 ; PTR TO TEST DATA SET
JSR PC,@MD1DT ; GO TEST
BR TST323 ;;

```

```

MD1D7: ; TEST DATA SET MD1D-7:
.WORD 042176,077600,0,0 ; INITIAL AC FLOAT NUMBER
.WORD FIN,0,0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 137777,0,0,0 ; EXPECTED FRACTION-PART FLOAT RESULT
.WORD AP,M1,AN,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047607,047610 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

4921	022100	000004		
4922	022102	012705	022114	
4923	022106	004737	036332	
4925	022112	000423		
4927	022114			
4928	022114	042176	077600	000000
4929	022122	000000		
4930	022124	140200	000000	000000
4931	022132	000000		
4932	022134	137777	000000	000000
4933	022142	000000		
4934	022144	052525	177777	125252
4935	022152	000000		
4936	022154	047607	047610	
4937	022160	000000		
4938				
4939				


```

4940
4941
4942
4943
4944
4945 022162 000004
4946 022164 012705 022176
4947 022170 004737 036332
4948
4949 022174 000423
4950
4951 022176
4952 022176 142177 100000 000000
4953 022204 000000
4954 022206 040200 000000 000000
4955 022214 000000
4956 022216 140000 000000 000000
4957 022224 000000
4958 022226 052525 177777 125252
4959 022234 000000
4960 022236 047747 047750
4961 022242 000000

```

```

*****
*TEST 323 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-10
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****
TST323: SCOPE
MOV #MD1D10,R5 ; PTR TO TEST DATA SET
JSR PC,@MD1DT ; GO TEST
BR TST324 ;;
MD1D10: ; TEST DATA SET MD1D-10:
.WORD 142177,MD,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 AP,M1,AN,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047747,047750 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

4962
4963
4964
4965
4966
4967
4968
4969 022244 000004
4970 022246 012705 022260
4971 022252 004737 036332
4972
4973 022256 000423
4974
4975 022260
4976 022260 140200 000000 000000
4977 022266 000000
4978 022270 140377 177777 177777
4979 022276 177777
4980 022300 040177 177777 177777
4981 022306 177776
4982 022310 052525 177777 125252
4983 022316 000000
4984 022320 047617 047600
4985 022324 000000

```

```

*****
*TEST 324 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-11
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****
TST324: SCOPE
MOV #MD1D11,R5 ; PTR TO TEST DATA SET
JSR PC,@MD1DT ; GO TEST
BR TST325 ;;
MD1D11: ; TEST DATA SET MD1D-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 AP,M1,AN,0 ; EXPECTED INTEGER-PART FLOAT RESULT
.WORD 047617,047600 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

4986
4987
4988
4989
4990
4991
4992
4993 022326 000004
4994 022330 012705 022342
4995 022334 004737 036332

```

```

*****
*TEST 325 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-12
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****
TST325: SCOPE
MOV #MD1D12,R5 ; PTR TO TEST DATA SET
JSR PC,@MD1DT ; GO TEST

```

E10

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 100
T325 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-12

SEQ 0267

```

4996
4997 022340 000423 BR TST326 ;;
4998
4999 022342 MD1D12: ; TEST DATA SET MD1D-12:
5000 022342 167452 125252 125252 .WORD 167452,AN,AN,AN ; INITIAL AC FLOAT NUMBER
5001 022350 125252 .WORD 112700,0,0,0 ; INITIAL MEM FLOAT NUMBER
5002 022352 112700 000000 000000 .WORD 040177,M1,M1,UB ; EXPECTED FRACTION-PART FLOAT RESULT
5003 022360 000000 .WORD AP,M1,AN,0 ; EXPECTED INTEGER-PART FLOAT RESULT
5004 022362 040177 177777 177777 .WORD 047757,047740 ; FPS: BEFORE, AFTER
5005 022370 177400 .WORD NA ; FEC AFTER ( 0 = N/A )
5006 022372 052525 177777 125252
5007 022400 000000
5008 022402 047757 047740
5009 022406 000000
5010
5011
5012
5013
5014
5015
5016

```

```

*****
; TEST 326 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-13
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

5017 022410 000004 †TST326: SCOPE
5018 022412 012705 022424 MOV #MD1D13,R5 ; PTR TO TEST DATA SET
5019 022416 004737 036332 JSR PC,@MD1DT ; GO TEST
5020
5021 022422 000423 BR TST327 ;;
5022
5023 022424 MD1D13: ; TEST DATA SET MD1D-13:
5024 022424 041000 000000 000000 .WORD 041000,0,0,1 ; INITIAL AC FLOAT NUMBER
5025 022432 000001 .WORD 037577,M1,M1,M2 ; INITIAL MEM FLOAT NUMBER
5026 022434 037577 177777 177777 .WORD 040177,M1,M1,M1 ; EXPECTED FRACTION-PART FLOAT RESULT
5027 022442 177776 .WORD AP,M1,AN,0 ; EXPECTED INTEGER-PART FLOAT RESULT
5028 022444 040177 177777 177777 .WORD 047657,047640 ; FPS: BEFORE, AFTER
5029 022452 177777 .WORD NA ; FEC AFTER ( 0 = N/A )
5030 022454 052525 177777 125252
5031 022462 000000
5032 022464 047657 047640
5033 022470 000000
5034
5035
5036
5037
5038
5039
5040

```

```

*****
; TEST 327 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-14
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

5041 022472 000004 †TST327: SCOPE
5042 022474 012705 022506 MOV #MD1D14,R5 ; PTR TO TEST DATA SET
5043 022500 004737 036332 JSR PC,@MD1DT ; GO TEST
5044
5045 022504 000423 BR TST330 ;;
5046
5047 022506 MD1D14: ; TEST DATA SET MD1D-14:
5048 022506 041000 000000 000000 .WORD 041000,0,0,1 ; INITIAL AC FLOAT NUMBER
5049 022514 000001 .WORD 037577,M1,M1,M2 ; INITIAL MEM FLOAT NUMBER
5050 022516 037577 177777 177777
5051 022524 177776

```


F10

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 101
T327 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-14

SEQ 0268

5052	022526	040200	000000	000000	.WORD	040200,0,0,0	; EXPECTED FRACTION-PART FLOAT RESULT
5053	022534	000000					
5054	022536	052525	177777	125252	.WORD	AP,M1,AN,0	; EXPECTED INTEGER-PART FLOAT RESULT
5055	022544	000000					
5056	022546	047617	047600		.WORD	047617,047600	; FPS: BEFORE, AFTER
5057	022552	000000			.WORD	NA	; FEC AFTER (0 = N/A)

```

5058
5059
5060
5061
5062
5063
5064
5065
5066
5067
5068
5069
5070
5071
5072
5073
5074
5075
5076
5077
5078
5079
5080
5081
5082
5083
5084
5085
5086
5087
5088

```

```

*****
*TEST 330 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-15
*
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

TST330: SCOPE
MOV #MD1D15,R5 ; PTR TO TEST DATA SET
JSR PC,@MD1DT ; GO TEST
BR TST331 ;;

```

```

MD1D15: ; TEST DATA SET MD1D-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 )

```

5065	022554	000004					
5066	022556	012705	022570				
5067	022562	004737	036332				
5069	022566	000423					
5071	022570						
5072	022570	142452	125252	125252			
5073	022576	125252					
5074	022600	077600	000000	000000			
5075	022606	000000					
5076	022610	000000	000000	000000			
5077	022616	000000					
5078	022620	052525	177777	125252			
5079	022626	000000					
5080	022630	047651	147646				
5081	022634	100010					

```

5082
5083
5084
5085
5086
5087
5088
5089
5090
5091
5092
5093
5094
5095
5096
5097
5098
5099
5100
5101
5102
5103
5104
5105
5106
5107

```

```

*****
*TEST 331 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-16
*
* OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

TST331: SCOPE
MOV #MD1D16,R5 ; PTR TO TEST DATA SET
JSR PC,@MD1DT ; GO TEST
BR TST332 ;;

```

```

MD1D16: ; TEST DATA SET MD1D-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 )

```

5089	022636	000004					
5090	022640	012705	022652				
5091	022644	004737	036332				
5093	022650	000423					
5095	022652						
5096	022652	142452	125252	125252			
5097	022660	125252					
5098	022662	077600	000000	000000			
5099	022670	000000					
5100	022672	000000	000000	000000			
5101	022700	000000					
5102	022702	052525	177777	125252			
5103	022710	000000					
5104	022712	046751	046746				
5105	022716	000000					

G10

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 102
T332 TEST OF MODD(1 ACC) INSTR, DATA SET MD1D-17

SEQ 0269

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
5141
5142
5143
5144
5145
5146
5147
5148
5149
5150
5151
5152
5153
5154
5155

```
*****
: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 )
```


H10

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 103
T334 TEST OF LCDF INSTR, DATA SET LCDF-1

SEQ 0270

```
5156  
5157  
5158  
5159  
5160  
5161 023064 000004  
5162 023066 012705 023100  
5163 023072 004737 036574  
5164  
5165 023076 000411  
5166  
5167 023100  
5168 023100 000000 000000 000000  
5169 023106 000000  
5170 023110 000000 000000  
5171 023114 047413 047404  
5172 023120 000000  
5173  
5174  
5175  
5176  
5177  
5178  
5179  
5180 023122 000004  
5181 023124 012705 023136  
5182 023130 004737 036574  
5183  
5184 023134 000411  
5185  
5186 023136  
5187 023136 100177 177777 177777  
5188 023144 177777  
5189 023146 052525 177777  
5190 023152 047503 147514  
5191 023156 100014  
5192  
5193  
5194  
5195  
5196  
5197  
5198  
5199 023160 000004  
5200 023162 012705 023174  
5201 023166 004737 036574  
5202  
5203 023172 000411  
5204  
5205 023174  
5206 023174 000177 177777 177777  
5207 023202 177777  
5208 023204 000000 000000  
5209 023210 047453 047444  
5210 023214 000000  
5211
```

```
*****  
:TEST 334 TEST OF LCDF INSTR, DATA SET LCDF-1  
:* ALL INTERRUPT ENABLES ON  
:* SHORT FLOAT, SHORT INTEGER, ROUND MODES  
*****  
TST334: SCOPE  
MOV #LCDF1,R5 ; PTR TO TEST DATA SET  
JSR PC,#LCDF1 ; GO TEST  
BR TST335 ;;  
LCDF1: ; TEST DATA SET LCDF-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 LCDF INSTR, DATA SET LCDF-2  
:* ALL INTERRUPT ENABLES ON  
:* SHORT FLOAT, LONG INTEGER, ROUND MODES  
*****  
TST335: SCOPE  
MOV #LCDF2,R5 ; PTR TO TEST DATA SET  
JSR PC,#LCDF2 ; GO TEST  
BR TST336 ;;  
LCDF2: ; TEST DATA SET LCDF-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 LCDF INSTR, DATA SET LCDF-3  
:* ALL INTERRUPT ENABLES ON  
:* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES  
*****  
TST336: SCOPE  
MOV #LCDF3,R5 ; PTR TO TEST DATA SET  
JSR PC,#LCDF3 ; GO TEST  
BR TST337 ;;  
LCDF3: ; TEST DATA SET LCDF-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 )
```

5212
5213
5214
5215
5216
5217
5218
5219
5220
5221
5222
5223
5224
5225
5226
5227
5228
5229
5230
5231
5232
5233
5234
5235
5236
5237
5238
5239
5240
5241
5242
5243
5244
5245
5246
5247
5248
5249
5250
5251
5252
5253
5254
5255
5256
5257
5258
5259
5260
5261
5262
5263
5264
5265
5266
5267

023216 000004
023220 012705 023232
023224 004737 036574

023230 000411

023232 040200 000000 000000
023240 000000
023242 040200 000000
023246 047557 047540
023252 000000

023254 000004
023256 012705 023270
023262 004737 036574

023266 000411

023270 140200 000000 100000
023276 000000
023300 140200 000001
023304 047407 047410
023310 000000

023312 000004
023314 012705 023326
023320 004737 036574

023324 000411

023326 140200 000000 100000
023334 000000
023336 140200 000000
023342 047447 047450
023346 000000

```
*****  
*TEST 337 TEST OF LCDF INSTR, DATA SET LCDF-4  
* ALL INTERRUPT ENABLES ON  
* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES  
*****  
TST337: 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  
*****  
TST340: 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  
*****  
TST341: 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 )
```


5268
5269
5270
5271
5272
5273
5274
5275 023350 000004
5276 023352 012705 023364
5277 023356 004737 036574
5278
5279 023362 000411
5280
5281 023364
5282 023364 077777 177777 177777
5283 023372 177777
5284 023374 000000 000000
5285 023400 047511 147506
5286 023404 100010
5287
5288
5289
5290
5291
5292
5293
5294 023406 000004
5295 023410 012705 023422
5296 023414 004737 036574
5297
5298 023420 000411
5299
5300 023422
5301 023422 077777 177777 177777
5302 023430 177777
5303 023432 077777 177777
5304 023436 047557 047540
5305 023442 000000
5306
5307
5308
5309
5310
5311
5312
5313 023444 000004
5314 023446 012705 023460
5315 023452 004737 036574
5316
5317 023456 000411
5318
5319 023460
5320 023460 121177 177777 100000
5321 023466 000000
5322 023470 121200 000000
5323 023474 047407 047410

```

*****
*TEST 342 TEST OF LCDF INSTR, DATA SET LCDF-7
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

TST342: 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
*****

```

```

TST343: 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
*****

```

```

TST344: 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

```

K10

FPU ADVANCED INSTR TESTS
DGFPBB.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 106
T344 TEST OF LDCDF INSTR, DATA SET LDCDF-11

SEQ 0273

5324 023500 000000

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

:TEST 345 TEST OF LDCDF INSTR, DATA SET LDCDF-12
: ALL INTERRUPT ENABLES ON
: SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
:*****

5332 023502 000004
5333 023504 012705 023516
5334 023510 004737 036574

TST345: SCOPE
MOV #LDCDF12,RS ; PTR TO TEST DATA SET
JSR PC,2#LDCDF ; GO TEST

5336 023514 000411

BR TST346 ;;

5338 023516 121177 177777 100000

LDCDF12: ; TEST DATA SET LDCDF-12:
.WORD 121177,M1,MD,0 ; INITIAL MEM FLOAT NUMBER

5340 023524 000000
5341 023526 121177 177777
5342 023532 047447 047450
5343 023536 000000

.WORD 121177,M1 ; EXPECTED FLOAT RESULT
.WORD 047447,047450 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

:TEST 346 TEST OF LDCDF INSTR, DATA SET LDCDF-13
: ALL INTERRUPT ENABLES ON
: SHORT FLOAT, LONG INTEGER, ROUND MODES
:*****

5351 023540 000004
5352 023542 012705 023554
5353 023546 004737 036574

TST346: SCOPE
MOV #LDCDF13,RS ; PTR TO TEST DATA SET
JSR PC,2#LDCDF ; GO TEST

5355 023552 000411

BR TST347 ;;

5357 023554 040200 000000 077777

LDCDF13: ; TEST DATA SET LDCDF-13:
.WORD F1P,0,LGP,M1 ; INITIAL MEM FLOAT NUMBER

5359 023562 177777
5360 023564 040200 000000
5361 023570 047517 047500
5362 023574 000000

.WORD F1P,0 ; EXPECTED FLOAT RESULT
.WORD 047517,047500 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

:TEST 347 TEST OF LDCDF INSTR, DATA SET LDCDF-14
: ALL INTERRUPT ENABLES ON
: SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
:*****

5370 023576 000004
5371 023600 012705 023612
5372 023604 004737 036574

TST347: SCOPE
MOV #LDCDF14,RS ; PTR TO TEST DATA SET
JSR PC,2#LDCDF ; GO TEST

5374 023610 000411

BR TST350 ;;

5376 023612 040200 000000 077777

LDCDF14: ; TEST DATA SET LDCDF-14:
.WORD F1P,0,LGP,M1 ; INITIAL MEM FLOAT NUMBER

5377 023620 177777
5378 023622 040200 000000

.WORD F1P,0 ; EXPECTED FLOAT RESULT

L10

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 107
T347 TEST OF LCDF INSTR, DATA SET LCDF-14

SEQ 0274

5380 023626 047557 047540
5381 023632 000000

.WORD 047557,047540 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER (0 = N/A)

5382
5383
5384
5385
5386
5387
5388
5389

:TEST 350 TEST OF LCDF INSTR, DATA SET LCDF-15
: * OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
: * SHORT FLOAT, LONG INTEGER, ROUND MODES
:*****

5390 023634 000004
5391 023636 012705 023650
5392 023642 004737 036574

TST350: SCOPE
MOV #LCDF15,R5 ; PTR TO TEST DATA SET
JSR PC,#LCDFT ; GO TEST
BR TST351 ;;

5393 023646 000411
5394
5395 023650
5396 023650 177777 177777 100000

LCDF15: ; TEST DATA SET LCDF-15:
.WORD LGN,M1,M0,0 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD 046511,046506 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER (0 = N/A)

5397 023656 000000
5398 023660 000000 000000
5399 023664 046511 046506
5400 023670 000000

5401
5402
5403
5404
5405
5406
5407

:TEST 351 TEST OF LCDF INSTR, DATA SET LCDF-16
: * -0 INTERRUPT ENABLE OFF, ALL OTHERS ON
: * SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
:*****

5408 023672 000004
5409 023674 012705 023706
5410 023700 004737 036574

TST351: SCOPE
MOV #LCDF16,R5 ; PTR TO TEST DATA SET
JSR PC,#LCDFT ; GO TEST
BR TST352 ;;

5411 023704 000411
5412
5413 023706
5414 023706 100000 177777 177777

LCDF16: ; TEST DATA SET LCDF-16:
.WORD M0,M1,M1,0 ; INITIAL MEM FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT
.WORD 043453,043444 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER (0 = N/A)

5415 023714 000000
5416 023716 000000 000000
5417 023722 043453 043444
5418 023726 000000

5419
5420
5421
5422

M10

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 108
T352 TEST OF LDCFD INSTR, DATA SET LCFD-1

SEQ 0275

5423				
5424				
5425				
5426				
5427				
5428	023730	000004		
5429	023732	012705	023744	
5430	023736	004737	036744	
5431				
5432	023742	000411		
5433				
5434	023744			
5435	023744	100000	000000	
5436	023750	052525	177777	125252
5437	023756	000000		
5438	023760	047643	147654	
5439	023764	100014		
5440				
5441				
5442				
5443				
5444				
5445				
5446				
5447	023766	000004		
5448	023770	012705	024002	
5449	023774	004737	036744	
5450				
5451	024000	000411		
5452				
5453	024002			
5454	024002	125252	125252	
5455	024006	125252	125252	000000
5456	024014	000000		
5457	024016	047607	047610	
5458	024022	000000		
5459				
5460				
5461				
5462				
5463				
5464				
5465				
5466	024024	000004		
5467	024026	012705	024040	
5468	024032	004737	036744	
5469				
5470	024036	000411		
5471				
5472	024040			
5473	024040	000000	000000	
5474	024044	000000	000000	000000
5475	024052	000000		
5476	024054	047753	047744	
5477	024060	000000		
5478				

```

*****
:TEST 352 TEST OF LDCFD INSTR, DATA SET LCFD-1
:* ALL INTERRUPT ENABLES ON
:* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

†ST352: SCOPE
MOV #LCFD1,R5 ; PTR TO TEST DATA SET
JSR PC,#LCFDT ; GO TEST
BR TST353 ;;

```

```

LCFD1: ; TEST DATA SET LCFD-1:
.WORD 0,0 ; INITIAL MEM FLOAT NUMBER
.WORD ALTP,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
*****

```

```

†ST353: SCOPE
MOV #LCFD2,R5 ; PTR TO TEST DATA SET
JSR PC,#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
*****

```

```

†ST354: SCOPE
MOV #LCFD3,R5 ; PTR TO TEST DATA SET
JSR PC,#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 )

```



```

5479
5480
5481
5482
5483
5484
5485 024062 000004
5486 024064 012705 024076
5487 024070 004737 036744
5488
5489 024074 000411
5490
5491 024076
5492 024076 077777 177777
5493 024102 077777 177777 000000
5494 024110 000000
5495 024112 047717 047700
5496 024116 000000
5497
5498
5499
5500
5501
5502
5503
5504 024120 000004
5505 024122 012705 024134
5506 024126 004737 036744
5507
5508 024132 000411
5509
5510 024134
5511 024134 000177 177777
5512 024140 000000 000000 000000
5513 024146 000000
5514 024150 047653 047644
5515 024154 000000
5516
5517
5518
5519
5520
5521
5522
5523 024156 000004
5524 024160 012705 024172
5525 024164 004737 036744
5526
5527 024170 000411
5528
5529 024172
5530 024172 177777 177777
5531 024176 177777 177777 000000
5532 024204 000000
5533 024206 047607 047610
5534 024212 000000

```

```

*****
*TEST 355 TEST OF LDCFD INSTR, DATA SET LCFD-4
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

TST355: SCOPE
MOV #LCFD4,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCFDT ; GO TEST
BR TST356 ;;

```

```

LCFD4: ; TEST DATA SET LCFD-4:
.WORD LCP,M1 ; INITIAL MEM FLOAT NUMBER
.WORD LCP,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
*****

```

```

TST356: 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
*****

```

```

TST357: 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 )

```

5535
5536
5537
5538
5539
5540
5541
5542
5543
5544
5545
5546
5547
5548
5549
5550
5551
5552
5553
5554
5555
5556
5557
5558
5559
5560
5561
5562
5563
5564
5565
5566
5567
5568
5569
5570
5571
5572
5573
5574
5575

```
*****  
*TEST 360 TEST OF LCFD INSTR, DATA SET LCFD-7  
* -0 INTERRUPT ENABLE OFF, ALL OTHERS ON  
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES  
*****
```

024214 000004
024216 012705 024230
024222 004737 036744

```
†ST360: SCOPE  
MOV #LCFD7_RS ; PTR TO TEST DATA SET  
JSR PC,#LCFDT ; GO TEST  
  
BR TST361 ;;
```

024226 000411

```
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 )
```

024230 100177 177777
024230 000000 000000 000000
024234 000000
024242 000000
024244 043753 043744
024250 000000

```
*****  
*TEST 361 TEST OF LCFD INSTR, DATA SET LCFD-10  
* ALL INTERRUPT ENABLES ON  
* LONG FLOAT, LONG INTEGER, ROUND MODES  
*****
```

024252 000004
024254 012705 024266
024260 004737 036744

```
†ST361: SCOPE  
MOV #LCFD10_RS ; PTR TO TEST DATA SET  
JSR PC,#LCFDT ; GO TEST  
  
BR TST362 ;;
```

024264 000411

```
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 )
```

024266 007417 007417
024266 007417 007417 000000
024272 007417
024300 000000
024302 047717 047700
024306 000000


```

5576 .....
5577 : *TEST 362 TEST OF STCDF INSTR, DATA SET SCDF-1
5578 : * ALL INTERRUPT ENABLES ON
5579 : * LONG FLOAT, LONG INTEGER, TRUNCATE MODES
5580 .....
5581 TST362: SCOPE
5582 MOV #SCDF1,RS ; PTR TO TEST DATA SET
5583 JSR PC,@SCDFT ; GO TEST
5584
5585 BR TST363 ;;
5586
5587 SCDF1: ; TEST DATA SET SCDF-1:
5588 .WORD 0,0,0,0 ; INITIAL AC FLOAT NUMBER
5589
5590 .WORD 0,0 ; EXPECTED FLOAT RESULT
5591 .WORD 047753,047744 ; FPS: BEFORE, AFTER
5592 .WORD NA ; FEC AFTER ( 0 = N/A )
5593
5594 .....
5595 : *TEST 363 TEST OF STCDF INSTR, DATA SET SCDF-2
5596 : * ALL INTERRUPT ENABLES ON
5597 : * LONG FLOAT, LONG INTEGER, ROUND MODES
5598 .....
5599 TST363: SCOPE
5600 MOV #SCDF2,RS ; PTR TO TEST DATA SET
5601 JSR PC,@SCDFT ; GO TEST
5602
5603 BR TST364 ;;
5604
5605 SCDF2: ; TEST DATA SET SCDF-2:
5606 .WORD FIN,0,MO,0 ; INITIAL AC FLOAT NUMBER
5607
5608 .WORD FIN,1 ; EXPECTED FLOAT RESULT
5609 .WORD 047707,047710 ; FPS: BEFORE, AFTER
5610 .WORD NA ; FEC AFTER ( 0 = N/A )
5611
5612 .....
5613 : *TEST 364 TEST OF STCDF INSTR, DATA SET SCDF-3
5614 : * ALL INTERRUPT ENABLES ON
5615 : * LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
5616 .....
5617 TST364: SCOPE
5618 MOV #SCDF3,RS ; PTR TO TEST DATA SET
5619 JSR PC,@SCDFT ; GO TEST
5620
5621 BR TST365 ;;
5622
5623 SCDF3: ; TEST DATA SET SCDF-3:
5624 .WORD FIP,0,MO,0 ; INITIAL AC FLOAT NUMBER
5625
5626 .WORD FIP,0 ; EXPECTED FLOAT RESULT
5627 .WORD 047657,047640 ; FPS: BEFORE, AFTER
5628 .WORD NA ; FEC AFTER ( 0 = N/A )
5629
5630
5631

```

```

5632
5633
5634
5635
5636
5637
5638 024442 000004
5639 024444 012705 024456
5640 024450 004737 037134
5641
5642 024454 000411
5643
5644 024456
5645 024456 000177 177777 177777
5646 024464 177777
5647 024466 000000 000000
5648 024472 047613 047604
5649 024476 000000
5650
5651
5652
5653
5654
5655
5656
5657 024500 000004
5658 024502 012705 024514
5659 024506 004737 037134
5660
5661 024512 000411
5662
5663 024514
5664 024514 040200 000000 100000
5665 024522 000000
5666 024524 040200 000001
5667 024530 047717 047700
5668 024534 000000
5669
5670
5671
5672
5673
5674
5675
5676 024536 000004
5677 024540 012705 024552
5678 024544 004737 037134
5679
5680 024550 000411
5681
5682 024552
5683 024552 177777 177777 177777
5684 024560 177777
5685 024562 177777 177777
5686 024566 047747 047750
5687 024572 000000

```

```

*****
:TEST 365 TEST OF STCDF INSTR, DATA SET SCDF-4
: ALL INTERRUPT ENABLES ON
: LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

TST365: 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
*****

```

```

TST366: SCOPE
MOV #SCDF5,R5 ; PTR TO TEST DATA SET
JSR PC,@SCDFT ; GO TEST
BR TST367 ;;
SCDF5: ; TEST DATA SET SCDF-5:
.WORD F1P,0,MO,0 ; INITIAL AC FLOAT NUMBER
.WORD F1P,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
*****

```

```

TST367: 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 )

```



```

5688
5689
5690
5691
5692
5693
5694
5695 024574 000004
5696 024576 012705 024610
5697 024602 004737 037134
5698
5699 024606 000411
5700
5701 024610
5702 024610 040200 000000 077777
5703 024616 177777
5704 024620 040200 000000
5705 024624 047617 047600
5706 024630 000000
5707
5708
5709
5710
5711
5712
5713
5714 024632 000004
5715 024634 012705 024646
5716 024640 004737 037134
5717
5718 024644 000411
5719
5720 024646
5721 024646 177777 177777 177777
5722 024654 177777
5723 024656 100000 000000
5724 024662 047601 147616
5725 024666 100010
5726
5727
5728
5729
5730
5731
5732
5733 024670 000004
5734 024672 012705 024704
5735 024676 004737 037134
5736
5737 024702 000411
5738
5739 024704
5740 024704 040200 000000 077777
5741 024712 177777
5742 024714 040200 000000
5743 024720 047757 047740

```

```

*****
:TEST 370 TEST OF STCDF INSTR, DATA SET SCDF-7
:*
:* ALL INTERRUPT ENABLES ON
:* LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

TST370: SCOPE
MOV #SCDF7_RS ; PTR TO TEST DATA SET
JSR PC,@#SCDFT ; GO TEST
BR TST371 ;;

SCDF7: ; TEST DATA SET SCDF-7:
.WORD F1P,0,LGP,M1 ; INITIAL AC FLOAT NUMBER
.WORD F1P,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
*****

```

```

TST371: SCOPE
MOV #SCDF10_RS ; PTR TO TEST DATA SET
JSR PC,@#SCDFT ; GO TEST
BR TST372 ;;

SCDF10: ; TEST DATA SET SCDF-10:
.WORD LGN,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD M0,0 ; 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
*****

```

```

TST372: SCOPE
MOV #SCDF11_RS ; PTR TO TEST DATA SET
JSR PC,@#SCDFT ; GO TEST
BR TST373 ;;

SCDF11: ; TEST DATA SET SCDF-11:
.WORD F1P,0,LGP,M1 ; INITIAL AC FLOAT NUMBER
.WORD F1P,0 ; EXPECTED FLOAT RESULT
.WORD 047757,047740 ; FPS: BEFORE, AFTER

```

F11

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 114
T372 TEST OF STCDF INSTR, DATA SET SCDF-11

SEQ 0281

5744 024724 000000

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

5745

5746

5747

5748

5749

5750

5751

5752 024726 000004

5753 024730 012705 024742

5754 024734 004737 037134

5755

5756 024740 000411

5757

5758 024742

5759 024742 101777 177777 100000

5760 024750 000000

5761 024752 102000 000000

5762 024756 047707 047710

5763 024762 000000

5764

5765

5766

5767

5768 024764 000004

5769 024766 012705 025000

5770 024772 004737 037134

5771 024776 000411

5772

5773 025000

5774 025000 101777 177777 100000

5775 025006 000000

5776 025010 101777 177777

5777 025014 047647 047650

5778 025020 000000

5779

5780

5781

5782

5783 025022 000004

5784 025024 012705 025036

5785 025030 004737 037134

5786 025034 000411

5787

5788 025036

5789 025036 077777 177777 100000

5790 025044 000000

5791 025046 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_RS ; PTR TO TEST DATA SET
JSR PC, @SCDF12 ; GO TEST
BR TST374 ;;

SCDF12: ; TEST DATA SET SCDF-12:
.WORD 101777, M1, M0, 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_RS ; PTR TO TEST DATA SET
JSR PC, @SCDF13 ; GO TEST
BR TST375 ;;

SCDF13: ; TEST DATA SET SCDF-13:
.WORD 101777, M1, M0, 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_RS ; PTR TO TEST DATA SET
JSR PC, @SCDF14 ; GO TEST
BR TST376 ;;

SCDF14: ; TEST DATA SET SCDF-14:
.WORD LGP, M1, M0, 0 ; INITIAL AC FLOAT NUMBER
.WORD 0, 0 ; EXPECTED FLOAT RESULT

G11

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 115
T375 TEST OF STCDF INSTR, DATA SET SCDF-14

SEQ 0282

5800 025052 046611 046606
5801 025056 000000
5802
5803
5804

.WORD 046611,046606 ; FPS: BEFORE AFTER
.WORD NA ; FEC AFTER (0 = N/A)

```

5805 .....
5806 : *TEST 376 TEST OF STCFD INSTR, DATA SET SCFD-1
5807 : * ALL INTERRUPT ENABLES ON
5808 : * SHORT FLOAT, SHORT INTEGER, ROUND MODES
5809 : .....
5810 TST376: SCOPE
5811 MOV #SCFD1,R5 ; PTR TO TEST DATA SET
5812 JSR PC,@#SCFDT ; GO TEST
5813
5814 BR TST377 ;;
5815
5816 SCFD1: ; TEST DATA SET SCFD-1:
5817 .WORD ALTP,ALTP,ALTP,ALTP ; INITIAL AC FLOAT NUMBER
5818
5819 .WORD ALTP,ALTP,0,0 ; EXPECTED FLOAT RESULT
5820
5821 .WORD 047417,047400 ; FPS: BEFORE, AFTER
5822
5823 .....
5824 : *TEST 377 TEST OF STCFD INSTR, DATA SET SCFD-2
5825 : * ALL INTERRUPT ENABLES ON
5826 : * SHORT FLOAT, LONG INTEGER, ROUND MODES
5827 : .....
5828 TST377: SCOPE
5829 MOV #SCFD2,R5 ; PTR TO TEST DATA SET
5830 JSR PC,@#SCFDT ; GO TEST
5831
5832 BR TST400 ;;
5833
5834 SCFD2: ; TEST DATA SET SCFD-2:
5835 .WORD ZXIMP,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
5836
5837 .WORD 0,0,0,0 ; EXPECTED FLOAT RESULT
5838
5839 .WORD 047513,047504 ; FPS: BEFORE, AFTER
5840
5841 .....
5842 : *TEST 400 TEST OF STCFD INSTR, DATA SET SCFD-3
5843 : * ALL INTERRUPT ENABLES ON
5844 : * SHORT FLOAT, SHORT INTEGER, ROUND MODES
5845 : .....
5846 TST400: SCOPE
5847 MOV #SCFD3,R5 ; PTR TO TEST DATA SET
5848 JSR PC,@#SCFDT ; GO TEST
5849
5850 BR TST401 ;;
5851
5852 SCFD3: ; TEST DATA SET SCFD-3:
5853 .WORD LGN,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
5854
5855 .WORD LGN,M1,0,0 ; EXPECTED FLOAT RESULT
5856
5857 .WORD 047407,047410 ; FPS: BEFORE, AFTER
5858
5859
5860

```



```

5861
5862
5863
5864
5865
5866
5867 025220 000004
5868 025222 012705 025234
5869 025226 004737 037300
5870
5871 025232 000412
5872
5873 025234
5874 025234 170360 170360 170360
5875 025242 170360
5876 025244 170360 170360 000000
5877 025252 000000
5878 025254 047547 047550
5879
5880
5881
5882
5883
5884
5885
5886 025260 000004
5887 025262 012705 025274
5888 025266 004737 037300
5889
5890 025272 000412
5891
5892 025274
5893 025274 000000 000000 000000
5894 025302 000000
5895 025304 000000 000000 000000
5896 025312 000000
5897 025314 047453 047444
5898
5899
5900
5901
5902
5903
5904
5905 025320 000004
5906 025322 012705 025334
5907 025326 004737 037300
5908
5909 025332 000412
5910
5911 025334
5912 025334 077777 000000 177777
5913 025342 177777
5914 025344 077777 000000 000000
5915 025352 000000
5916 025354 047517 047500

*****
;TEST 401 TEST OF STCFD INSTR, DATA SET SCFD-4
;*
;* ALL INTERRUPT ENABLES ON
;* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
*****
TST401: SCOPE
MOV #SCFD4,RS ; PTR TO TEST DATA SET
JSR PC,@#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
*****
TST402: SCOPE
MOV #SCFD5,RS ; PTR TO TEST DATA SET
JSR PC,@#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
*****
TST403: SCOPE
MOV #SCFD6,RS ; PTR TO TEST DATA SET
JSR PC,@#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

```

J11

FPU ADVANCED INSTR TESTS
DQFPBB.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 118
T403 TEST OF STCFD INSTR, DATA SET SCFD-6

SEQ 0285

5917
5918


```

5919
5920
5921
5922
5923
5924 025360 000004
5925 025362 012705 025374
5926 025366 004737 037422
5927
5928 025372 000405
5929
5930 025374
5931 025374 100000
5932 025376 144000 000000
5933 025402 047407 047410
5934
5935
5936
5937
5938
5939
5940
5941 025406 000004
5942 025410 012705 025422
5943 025414 004737 037422
5944
5945 025420 000405
5946
5947 025422
5948 025422 007417
5949 025424 043160 170000
5950 025430 047457 047440
5951
5952
5953
5954
5955
5956
5957
5958 025434 000004
5959 025436 012705 025450
5960 025442 004737 037422
5961
5962 025446 000405
5963
5964 025450
5965 025450 000000
5966 025452 000000 000000
5967 025456 047413 047404
5968
5969
5970
5971
5972
5973
5974

```

```

*****
*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
*****

```

5975 025462 000004
5976 025464 012705 025476
5977 025470 004737 037422
5978
5979 025474 000405
5980
5981 025476
5982 025476 170360
5983 025500 143161 000000
5984 025504 047447 047450
5985
5986
5987
5988
5989
5990
5991
5992 025510 000004
5993 025512 012705 025524
5994 025516 004737 037422
5995
5996 025522 000405
5997
5998 025524
5999 025524 077777
6000 025526 043777 177000
6001 025532 047417 047400
6002
6003
6004

TST407: SCOPE
MOV #LCIF4,RS ; PTR TO TEST DATA SET
JSR PC,2#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 ENABLES ON
* SHORT FLOAT, SHORT INTEGER, ROUND MODES

TST410: SCOPE
MOV #LCIF5,RS ; PTR TO TEST DATA SET
JSR PC,2#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

M11

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 121
T411 TEST OF LDCID INSTR, DATA SET LCID-1

SEQ 0288

```

6005
6006
6007
6008
6009
6010 025536 000004
6011 025540 012705 025552
6012 025544 004737 037522
6013
6014 025550 000407
6015
6016 025552
6017 025552 107070
6018 025554 143743 110000 000000
6019 025562 000000
6020 025564 047600 047610
6021
6022
6023
6024
6025
6026
6027
6028 025570 000004
6029 025572 012705 025604
6030 025576 004737 037522
6031
6032 025602 000407
6033
6034 025604
6035 025604 000000
6036 025606 000000 000000 000000
6037 025614 000000
6038 025616 047653 047644
6039
6040
6041
6042
6043
6044
6045
6046 025622 000004
6047 025624 012705 025636
6048 025630 004737 037522
6049
6050 025634 000407
6051
6052 025636
6053 025636 077777
6054 025640 043777 177000 000000
6055 025646 000000
6056 025650 047657 047640
6057
6058
6059
6060

```

```

*****
;TEST 411 TEST OF LDCID INSTR, DATA SET LCID-1
;
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, SHORT INTEGER, ROUND MODES
;
*****
TST411: SCOPE
MOV #LCID1,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCIDT ; GO TEST

BR TST412 ;;

LCID1: ; TEST DATA SET LCID-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 LCID-2
;
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
;
*****
TST412: SCOPE
MOV #LCID2,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCIDT ; GO TEST

BR TST413 ;;

LCID2: ; TEST DATA SET LCID-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 LCID-3
;
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
;
*****
TST413: SCOPE
MOV #LCID3,R5 ; PTR TO TEST DATA SET
JSR PC,@#LCIDT ; GO TEST

BR TST414 ;;

LCID3: ; TEST DATA SET LCID-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 LCID-4

```

6061
6062
6063
6064 025654 000004
6065 025656 012705 025670
6066 025662 004737 037522
6067
6068 025666 000407
6069
6070 025670
6071 025670 070707
6072 025672 043743 107000 000000
6073 025700 000000
6074 025702 047617 047600
6075
6076
6077
6078
6079
6080
6081
6082 025706 000004
6083 025710 012705 025722
6084 025714 004737 037522
6085
6086 025720 000407
6087
6088 025722
6089 025722 100000
6090 025724 144000 000000 000000
6091 025732 000000
6092 025734 047647 047650
6093
6094
6095

```

;*          ALL INTERRUPT ENABLES ON
;*          LONG FLOAT, SHORT INTEGER, ROUND MODES
;*****
TST414: SCOPE
MOV      #LCID4,R5      ; PTR TO TEST DATA SET
JSR      PC,@#LCIDT    ; GO TEST
BR       TST415        ;;

LCID4:   ; TEST DATA SET LCID-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 LCID-5
;*          ALL INTERRUPT ENABLES ON
;*          LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
;*****
TST415: SCOPE
MOV      #LCID5,R5      ; PTR TO TEST DATA SET
JSR      PC,@#LCIDT    ; GO TEST
BR       TST416        ;;

LCID5:   ; TEST DATA SET LCID-5:
.WORD   100000        ; INITIAL INTEGER VALUE
.WORD   144000,000000,0,0 ; EXPECTED FLOAT RESULT
.WORD   047647,047650 ; FPS: BEFORE, AFTER

```



```

6096
6097
6098
6099
6100
6101 025740 000004
6102 025742 012705 025754
6103 025746 004737 037642
6104
6105 025752 000406
6106
6107 025754
6108 025754 077777 177777
6109 025760 050000 000000
6110 025764 047517 047500
6111
6112
6113
6114
6115
6116
6117
6118 025770 000004
6119 025772 012705 026004
6120 025776 004737 037642
6121
6122 026002 000406
6123
6124 026004
6125 026004 077777 177777
6126 026010 047777 177777
6127 026014 047557 047540
6128
6129
6130
6131
6132
6133
6134
6135 026020 000004
6136 026022 012705 026034
6137 026026 004737 037642
6138
6139 026032 000406
6140
6141 026034
6142 026034 170360 170360
6143 026040 147160 170361
6144 026044 047507 047510
6145
6146
6147
6148
6149
6150
6151

```

```

*****
: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,2#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,2#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,2#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
*****

```

```

6152 026050 000004
6153 026052 012705 026064
6154 026056 004737 037642
6155
6156 026062 000406
6157
6158 026064
6159 026064 000000 000000
6160 026070 000000 000000
6161 026074 047513 047504
6162
6163
6164
6165
6166
6167
6168
6169 026100 000004
6170 026102 012705 026114
6171 026106 004737 037642
6172
6173 026112 000406
6174
6175 026114
6176 026114 077777 177677
6177 026120 047777 177777
6178 026124 047517 047500
6179
6180
6181
6182
6183
6184
6185
6186 026130 000004
6187 026132 012705 026144
6188 026136 004737 037642
6189
6190 026142 000406
6191
6192 026144
6193 026144 100000 000000
6194 026150 150000 000000
6195 026154 047547 047550
6196
6197
6198
6199
6200
6201
6202
6203 026160 000004
6204 026162 012705 026174
6205 026166 004737 037642
6206
6207 026172 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
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 125
T424 TEST OF LDCLF INSTR, DATA SET LCLF-7

SEQ 0292

6208			
6209	026174		
6210	026174	043434	070707
6211	026200	047616	034343
6212	026204	047557	047540
6213			
6214			
6215			

```

LCLF7: ; TEST DATA SET LCLF-7:
        .WORD 043434,070707 ; INITIAL INTEGER VALUE
        .WORD 047616,034343 ; EXPECTED FLOAT RESULT
        .WORD 047557,047540 ; FPS: BEFORE, AFTER
  
```

6216				
6217				
6218				
6219				
6220				
6221	026210	000004		
6222	026212	012705	026224	
6223	026216	004737	037742	
6224				
6225	026222	000410		
6226				
6227	026224			
6228	026224	007417	007417	
6229	026230	047160	170360	170000
6230	026236	000000		
6231	026240	047717	047700	
6232				
6233				
6234				
6235				
6236				
6237				
6238				
6239	026244	000004		
6240	026246	012705	026260	
6241	026252	004737	037742	
6242				
6243	026256	000410		
6244				
6245	026260			
6246	026260	100000	000000	
6247	026264	150000	000000	000000
6248	026272	000000		
6249	026274	047747	047750	
6250				
6251				
6252				
6253				
6254				
6255				
6256				
6257	026300	000004		
6258	026302	012705	026314	
6259	026306	004737	037742	
6260				
6261	026312	000410		
6262				
6263	026314			
6264	026314	077777	177777	
6265	026320	047777	177777	177000
6266	026326	000000		
6267	026330	047757	047740	
6268				
6269				
6270				
6271				

```

*****
; *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

```



```

6272
6273
6274
6275 026334 000004
6276 026336 012705 026350
6277 026342 004737 037742
6278
6279 026346 000410
6280
6281 026350
6282 026350 107070 161616
6283 026354 147743 107070 162000
6284 026362 000000
6285 026364 047700 047710
6286
6287
6288
6289
6290
6291
6292
6293 026370 000004
6294 026372 012705 026404
6295 026376 004737 037742
6296
6297 026402 000410
6298
6299 026404
6300 026404 000000 000000
6301 026410 000000 000000 000000
6302 026416 000000
6303 026420 047753 047744
6304
6305
6306

```

```

;*
;* 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

```

```

6307
6308
6309
6310
6311
6312 026424 000004
6313 026426 012705 026440
6314 026432 004737 040062
6315
6316 026436 000406
6317
6318 026440
6319 026440 000000 000000
6320 026444 000000
6321 026446 047453 047444
6322 026452 000000
6323
6324
6325
6326
6327
6328
6329
6330 026454 000004
6331 026456 012705 026470
6332 026462 004737 040062
6333
6334 026466 000406
6335
6336 026470
6337 026470 041532 000000
6338 026474 000066
6339 026476 047457 047440
6340 026502 000000
6341
6342
6343
6344
6345
6346
6347
6348 026504 000004
6349 026506 012705 026520
6350 026512 004737 040062
6351
6352 026516 000406
6353
6354 026520
6355 026520 052525 052525
6356 026524 000000
6357 026526 047452 147445
6358 026532 100006
6359
6360
6361
6362

```

```

*****
;TEST 432 TEST OF STCFI INSTR, DATA SET SCFI-1
;*
;* ALL INTERRUPT ENABLES ON
;* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST432: 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
*****
TST433: 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
*****
TST434: 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

```


H12

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 129
T435 TEST OF STCFI INSTR, DATA SET SCFI-4

SEQ 0296

6363			
6364			
6365			
6366	026534	000004	
6367	026536	012705	026550
6368	026542	004737	040062
6369			
6370	026546	000406	
6371			
6372	026550		
6373	026550	141531	177777
6374	026554	177712	
6375	026556	047407	047410
6376	026562	000000	
6377			
6378			
6379			
6380			
6381			
6382			
6383			
6384	026564	000004	
6385	026566	012705	026600
6386	026572	004737	040062
6387			
6388	026576	000406	
6389			
6390	026600		
6391	026600	041532	000000
6392	026604	000066	
6393	026606	047417	047400
6394	026612	000000	
6395			
6396			
6397			
6398			
6399			
6400			
6401			
6402	026614	000004	
6403	026616	012705	026630
6404	026622	004737	040062
6405			
6406	026626	000406	
6407			
6408	026630		
6409	026630	172011	123456
6410	026634	000000	
6411	026636	047052	047045
6412	026642	000000	
6413			
6414			
6415			
6416			
6417			
6418			

```

;*
;* 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
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 130
T440 TEST OF STCFI INSTR, DATA SET SCFI-7

SEQ 0297

6419			
6420	026644	000004	
6421	026646	012705	026660
6422	026652	004737	040062
6423			
6424	026656	000406	
6425			
6426	026660		
6427	026660	000000	177777
6428	026664	000000	
6429	026666	047413	047404
6430	026672	000000	
6431			
6432			
6433			

```

:*****
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 )

```



```

6434
6435
6436
6437
6438
6439 026674 000004
6440 026676 012705 026710
6441 026702 004737 040260
6442
6443 026706 000410
6444
6445 026710
6446 026710 044000 000000 000000
6447 026716 000000
6448 026720 000000
6449 026722 047652 147645
6450 026726 100006
6451
6452
6453
6454
6455
6456
6457
6458 026730 000004
6459 026732 012705 026744
6460 026736 004737 040260
6461
6462 026742 000410
6463
6464 026744
6465 026744 043777 177377 177777
6466 026752 177777
6467 026754 077777
6468 026756 047617 047600
6469 026762 000000
6470
6471
6472
6473
6474
6475
6476
6477 026764 000004
6478 026766 012705 027000
6479 026772 004737 040260
6480
6481 026776 000410
6482
6483 027000
6484 027000 000000 000000 000000
6485 027006 000000
6486 027010 000000
6487 027012 047613 047604
6488 027016 000000
6489

```

```

*****
:TEST 441 TEST OF STCDI INSTR, DATA SET SCDI-1
:*
:* ALL INTERRUPT ENABLES ON
:* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST441: SCOPE
MOV #SCDI1,R5 ; 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,R5 ; 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,R5 ; 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 )

```

```

6490
6491
6492
6493
6494
6495
6496 027020 000004
6497 027022 012705 027034
6498 027026 004737 040260
6499
6500 027032 000410
6501
6502 027034
6503 027034 143161 007777 177777
6504 027042 177777
6505 027044 170360
6506 027046 047607 047610
6507 027052 000000
6508
6509
6510
6511
6512
6513
6514
6515 027054 000004
6516 027056 012705 027070
6517 027062 004737 040260
6518
6519 027066 000410
6520
6521 027070
6522 027070 143777 177777 177777
6523 027076 177777
6524 027100 100001
6525 027102 047647 047650
6526 027106 000000
6527
6528
6529
6530
6531
6532
6533
6534 027110 000004
6535 027112 012705 027124
6536 027116 004737 040260
6537
6538 027122 000410
6539
6540 027124
6541 027124 152525 052525 177777
6542 027132 000000
6543 027134 000000
6544 027136 047212 047205
6545 027142 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 )

```


6546
6547
6548
6549
6550
6551
6552
6553
6554
6555
6556
6557
6558
6559
6560
6561
6562
6563
6564
6565
6566
6567

```

*****
*TEST 447      TEST OF STCDI INSTR, DATA SET SCDI-7
*              ALL INTERRUPT ENABLES ON
*              LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST447: SCOPE
MOV      #SCDI7_RS      ; PTR TO TEST DATA SET
JSR      PC,#SCDI7      ; 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 )

```

```

027144 000004
027146 012705 027160
027152 004737 040260
027156 000410
027160
027160 140377 177777 177777
027166 052525
027170 177777
027172 047647 047650
027176 000000

```

```

6568
6569
6570
6571
6572
6573 027200 000004
6574 027202 012705 027214
6575 027206 004737 040456
6576
6577 027212 000407
6578
6579 027214
6580 027214 047777 177777
6581 027220 077777 177600
6582 027224 047517 047500
6583 027230 000000
6584
6585
6586
6587
6588
6589
6590
6591 027232 000004
6592 027234 012705 027246
6593 027240 004737 040456
6594
6595 027244 000407
6596
6597 027246
6598 027246 150000 000001
6599 027252 000000 000000
6600 027256 047512 147505
6601 027262 100006
6602
6603
6604
6605
6606
6607
6608
6609 027264 000004
6610 027266 012705 027300
6611 027272 004737 040456
6612
6613 027276 000407
6614
6615 027300
6616 027300 037777 177777
6617 027304 000000 000000
6618 027310 047553 047544
6619 027314 000000
6620
6621
6622
6623

```

```

*****
; *TEST 450 TEST OF STCFL INSTR, DATA SET SCFL-1
; * ALL INTERRUPT ENABLES ON
; * SHORT FLOAT, LONG INTEGER, ROUND MODES
; *
*****
TST450: 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
; *
*****
TST451: 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
; *
*****
TST452: 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

```



```

6624
6625
6626
6627 027316 000004
6628 027320 012705 027332
6629 027324 004737 040456
6630
6631 027330 000407
6632
6633 027332
6634 027332 000000 000000
6635 027336 000000 000000
6636 027342 047553 047544
6637 027346 000000
6638
6639
6640
6641
6642
6643
6644
6645 027350 000004
6646 027352 012705 027364
6647 027356 004737 040456
6648
6649 027362 000407
6650
6651 027364
6652 027364 147777 177777
6653 027370 100000 000200
6654 027374 047507 047510
6655 027400 000000
6656
6657
6658
6659
6660
6661
6662
6663 027402 000004
6664 027404 012705 027416
6665 027410 004737 040456
6666
6667 027414 000407
6668
6669 027416
6670 027416 040577 177777
6671 027422 000000 000003
6672 027426 047517 047500
6673 027432 000000
6674
6675
6676
6677
6678
6679

```

```

;*
;* 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

```

```

6680
6681 027434 000004
6682 027436 012705 027450
6683 027442 004737 040456
6684
6685 027446 000407
6686
6687 027450
6688 027450 066666 123456
6689 027454 000000 000000
6690 027460 047152 047145
6691 027464 000000
6692
6693
6694

```

```

*****
†T456: SCOPE
      MOV      #SCFL7,R5      ; PTR TO TEST DATA SET
      JSR      PC,2#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 )

```



```

6695
6696
6697
6698
6699
6700 027466 000004
6701 027470 012705 027502
6702 027474 004737 040664
6703
6704 027500 000411
6705
6706 027502
6707 027502 050000 177000 177000
6708 027510 177000
6709 027512 000000 000000
6710 027516 047712 147705
6711 027522 100006
6712
6713
6714
6715
6716
6717
6718
6719 027524 000004
6720 027526 012705 027540
6721 027532 004737 040664
6722
6723 027536 000411
6724
6725 027540
6726 027540 047777 177777 177377
6727 027546 177777
6728 027550 077777 177777
6729 027554 047717 047700
6730 027560 000000
6731
6732
6733
6734
6735
6736
6737
6738 027562 000004
6739 027564 012705 027576
6740 027570 004737 040664
6741
6742 027574 000411
6743
6744 027576
6745 027576 137777 125252 177777
6746 027604 177777
6747 027606 000000 000000
6748 027612 047713 047704
6749 027616 000000
6750

```

```

*****
:TEST 457 TEST OF STCDL INSTR, DATA SET SCDL-1
:* ALL INTERRUPT ENABLES ON
:* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
TST457: 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
*****
TST460: 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
*****
TST461: 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 )

```

```

6751
6752
6753
6754
6755
6756
6757 027620 000004
6758 027622 012705 027634
6759 027626 004737 040664
6760
6761 027632 000411
6762
6763 027634
6764 027634 147777 177777 177777
6765 027642 177777
6766 027644 100000 000001
6767 027650 047707 047710
6768 027654 000000
6769
6770
6771
6772
6773
6774
6775
6776 027656 000004
6777 027660 012705 027672
6778 027664 004737 040664
6779
6780 027670 000411
6781
6782 027672
6783 027672 047160 170360 177777
6784 027700 177777
6785 027702 007417 007417
6786 027706 047757 047740
6787 027712 000000
6788
6789
6790
6791
6792
6793
6794
6795 027714 000004
6796 027716 012705 027730
6797 027722 004737 040664
6798
6799 027726 000411
6800
6801 027730
6802 027730 000177 177777 125252
6803 027736 101010
6804 027740 000000 000000
6805 027744 047713 047704
6806 027750 000000

```

```

*****
:TEST 462 TEST OF STCOL INSTR, DATA SET SCOL-4
:*
:* ALL INTERRUPT ENABLES ON
:* LONG FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

TST462: SCOPE
MOV #SCDL4,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCDLT ; GO TEST
BR TST463 ;;

```

```

SCDL4: ; TEST DATA SET SCOL-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 STCOL INSTR, DATA SET SCOL-5
:*
:* ALL INTERRUPT ENABLES ON
:* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

TST463: SCOPE
MOV #SCDL5,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCDLT ; GO TEST
BR TST464 ;;

```

```

SCDL5: ; TEST DATA SET SCOL-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 STCOL INSTR, DATA SET SCOL-6
:*
:* ALL INTERRUPT ENABLES ON
:* LONG FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

TST464: SCOPE
MOV #SCDL6,R5 ; PTR TO TEST DATA SET
JSR PC,@#SCDLT ; GO TEST
BR TST465 ;;

```

```

SCDL6: ; TEST DATA SET SCOL-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 )

```


6807
6808
6809
6810
6811
6812
6813
6814
6815
6816
6817
6818
6819
6820
6821
6822
6823
6824
6825
6826
6827
6828

027752 000004
027754 012705 027766
027760 004737 040664
027764 000411
027766
027766 062141 125252 052525
027774 125252
027776 000000 000000
030002 047312 047305
030006 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  
:*****  
TST465: SCOPE  
MOV #SCDL7_RS ; 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 )
```

```

6829
6830
6831
6832
6833
6834 030010 000004
6835 030012 012705 030024
6836 030016 004737 041072
6837
6838 030022 000410
6839
6840 030024
6841 030024 020177 177777
6842 030030 000377 177777
6843 030034 000201
6844 030036 047555 147542
6845 030042 100010
6846
6847
6848
6849
6850
6851
6852
6853 030044 000004
6854 030046 012705 030060
6855 030052 004737 041072
6856
6857 030056 000410
6858
6859 030060
6860 030060 120000 000000
6861 030064 100000 000000
6862 030070 000200
6863 030072 047501 147516
6864 030076 100010
6865
6866
6867
6868
6869
6870
6871
6872 030100 000004
6873 030102 012705 030114
6874 030106 004737 041072
6875
6876 030112 000410
6877
6878 030114
6879 030114 020125 052525
6880 030120 077725 052525
6881 030124 000177
6882 030126 047457 047440
6883 030132 000000
6884

```

```

*****
: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,3#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,3#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,3#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 )

```



```

6885
6886
6887
6888
6889
6890
6891 030134 000004
6892 030136 012705 030150
6893 030142 004737 041072
6894
6895 030146 000410
6896
6897 030150
6898 030150 120052 125252
6899 030154 160052 125252
6900 030160 000100
6901 030162 047407 047410
6902 030166 000000
6903
6904
6905
6906
6907
6908
6909
6910 030170 000004
6911 030172 012705 030204
6912 030176 004737 041072
6913
6914 030202 000410
6915
6916 030204
6917 030204 020017 007417
6918 030210 040217 007417
6919 030214 000001
6920 030216 047557 047540
6921 030222 000000
6922
6923
6924
6925
6926
6927
6928
6929 030224 000004
6930 030226 012705 030240
6931 030232 004737 041072
6932
6933 030236 000410
6934
6935 030240
6936 030240 120160 170360
6937 030244 140160 170360
6938 030250 000000
6939 030252 047507 047510
6940 030256 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,ALT4P ; INITIAL AC FLOAT NUMBER
.WORD 040217,ALT4P ; 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,ALT4N ; INITIAL AC FLOAT NUMBER
.WORD 140160,ALT4N ; EXPECTED FLOAT RESULT
.WORD 0 ; EXPONENT TO BE LOADED
.WORD 047507,047510 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

H13

FPU ADVANCED INSTR TESTS
DQFPBB.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 142
T473 TEST OF LDEXP/F INSTR, DATA SET LEXF-6

SEQ 0309

6941
6942
6943
6944
6945
6946
6947
6948 030260 000004
6949 030262 012705 030274
6950 030266 004737 041072
6951
6952 030272 000410
6953
6954 030274
6955 030274 020177 177777
6956 030300 037777 177777
6957 030304 177777
6958 030306 047457 047440
6959 030312 000000
6960
6961
6962
6963
6964
6965
6966
6967 030314 000004
6968 030316 012705 030330
6969 030322 004737 041072
6970
6971 030326 000410
6972
6973 030330
6974 030330 120000 000000
6975 030334 120000 000000
6976 030340 177700
6977 030342 047407 047410
6978 030346 000000
6979
6980
6981
6982
6983
6984
6985
6986 030350 000004
6987 030352 012705 030364
6988 030356 004737 041072
6989
6990 030362 000410
6991
6992 030364
6993 030364 020125 052525
6994 030370 000325 052525
6995 030374 177601
6996 030376 047557 047540

```
*****  
:TEST 474 TEST OF LDEXP/F INSTR, DATA SET LEXF-7  
:* ALL INTERRUPT ENABLES ON  
:* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES  
*****
```

```
TST474: SCOPE  
MOV #LEXF7,R5 ; PTR TO TEST DATA SET  
JSR PC,@#LEXFT ; 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  
*****
```

```
TST475: SCOPE  
MOV #LEXF10,R5 ; PTR TO TEST DATA SET  
JSR PC,@#LEXFT ; 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  
*****
```

```
TST476: SCOPE  
MOV #LEXF11,R5 ; PTR TO TEST DATA SET  
JSR PC,@#LEXFT ; 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
```



```

6997 030402 000000
6998
6999
7000
7001
7002
7003
7004
7005 030404 000004
7006 030406 012705 030420
7007 030412 004737 041072
7008
7009 030416 000410
7010
7011 030420
7012 030420 120052 125252
7013 030424 100052 125252
7014 030430 177600
7015 030432 047503 147514
7016 030436 100012
7017
7018
7019
7020
7021
7022
7023
7024 030440 000004
7025 030442 012705 030454
7026 030446 004737 041072
7027
7028 030452 000410
7029
7030 030454
7031 030454 020017 007417
7032 030460 077617 007417
7033 030464 177577
7034 030466 047457 147440
7035 030472 100012
7036
7037
7038
7039
7040
7041
7042
7043 030474 000004
7044 030476 012705 030510
7045 030502 004737 041072
7046
7047 030506 000410
7048
7049 030510
7050 030510 120160 170360
7051 030514 177560 170360
7052 030520 177576

```

```

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

*****
*TEST 477 TEST OF LDEXP/F INSTR, DATA SET LEXF-12
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****
TST477: SCOPE
MOV #LEXF12,R5 ; PTR TO TEST DATA SET
JSR PC,#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 )

*****
*TEST 500 TEST OF LDEXP/F INSTR, DATA SET LEXF-13
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST500: SCOPE
MOV #LEXF13,R5 ; PTR TO TEST DATA SET
JSR PC,#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 )

*****
*TEST 501 TEST OF LDEXP/F INSTR, DATA SET LEXF-14
* ALL INTERRUPT ENABLES ON
* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****
TST501: SCOPE
MOV #LEXF14,R5 ; PTR TO TEST DATA SET
JSR PC,#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

```

7053 030522 047507 147510
7054 030526 100012

.WORD 047507,147510 ; FPS: BEFORE, AFTER
.WORD 100012 ; FEC AFTER (0 = N/A)

7055
7056
7057
7058
7059
7060

:TEST 502 TEST OF LDEXP/F INSTR, DATA SET LEXF-15
:* ALL INTERRUPT ENABLES ON
:* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
:*****

7061
7062 030530 000004
7063 030532 012705 030544
7064 030536 004737 041072
7065
7066 030542 000410
7067

TST502: SCOPE
MOV #LEXF15,R5 ; PTR TO TEST DATA SET
JSR PC,@#LEXFT ; GO TEST
BR TST503 ;;

7068 030544
7069 030544 020177 177777
7070 030550 077377 177777
7071 030554 177575
7072 030556 047457 147440
7073 030562 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)

7074
7075
7076
7077
7078
7079

:TEST 503 TEST OF LDEXP/F INSTR, DATA SET LEXF-16
:* ALL INTERRUPT ENABLES ON
:* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
:*****

7080
7081 030564 000004
7082 030566 012705 030600
7083 030572 004737 041072
7084
7085 030576 000410
7086

TST503: SCOPE
MOV #LEXF16,R5 ; PTR TO TEST DATA SET
JSR PC,@#LEXFT ; GO TEST
BR TST504 ;;

7087 030600
7088 030600 142000 000000
7089 030604 140000 000000
7090 030610 000000
7091 030612 047547 047550
7092 030616 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)

7093
7094
7095
7096
7097
7098
7099

:TEST 504 TEST OF LDEXP/F INSTR, DATA SET LEXF-17
:* OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
:* SHORT FLOAT, LONG INTEGER, TRUNCATE MODES
:*****

7100 030620 000004
7101 030622 012705 030634
7102 030626 004737 041072
7103
7104 030632 000410
7105

TST504: SCOPE
MOV #LEXF17,R5 ; PTR TO TEST DATA SET
JSR PC,@#LEXFT ; GO TEST
BR TST505 ;;

7106 030634
7107 030634 020177 177777
7108 030640 000000 000000

LEXF17: ; TEST DATA SET LEXF-17:
.WORD 020177,M1 ; INITIAL AC FLOAT NUMBER
.WORD 0,0 ; EXPECTED FLOAT RESULT

7109 030644 000201
7110 030646 046551 046546
7111 030652 000000

.WORD 201 ; EXPONENT TO BE LOADED
.WORD 046551,046546 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER (0 = N/A)

7112
7113
7114
7115
7116
7117
7118

:TEST 505 TEST OF LDEXP/F INSTR, DATA SET LEXF-20
: * OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
: * SHORT FLOAT, LONG INTEGER, ROUND MODES
:*****

7119 030654 000004
7120 030656 012705 030670
7121 030662 004737 041072

TST505: SCOPE
MOV #LEXF20,R5 ; PTR TO TEST DATA SET
JSR PC,@#LEXF ; GO TEST

7122
7123 030666 000410
7124

BR TST506 ; ;

7125 030670
7126 030670 120000 000000
7127 030674 000000 000000
7128 030700 000200
7129 030702 046511 046506
7130 030706 000000

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)

7131
7132
7133
7134
7135
7136
7137

:TEST 506 TEST OF LDEXP/F INSTR, DATA SET LEXF-21
: * UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
: * SHORT FLOAT, LONG INTEGER, ROUND MODES
:*****

7138 030710 000004
7139 030712 012705 030724
7140 030716 004737 041072

TST506: SCOPE
MOV #LEXF21,R5 ; PTR TO TEST DATA SET
JSR PC,@#LEXF ; GO TEST

7141
7142 030722 000410
7143

BR TST507 ; ;

7144 030724
7145 030724 120052 125252
7146 030730 000000 000000
7147 030734 177600
7148 030736 045513 045504
7149 030742 000000

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)

7150
7151
7152
7153
7154
7155
7156

:TEST 507 TEST OF LDEXP/F INSTR, DATA SET LEXF-22
: * UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
: * SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
:*****

7157 030744 000004
7158 030746 012705 030760
7159 030752 004737 041072

TST507: SCOPE
MOV #LEXF22,R5 ; PTR TO TEST DATA SET
JSR PC,@#LEXF ; GO TEST

7160
7161 030756 000410
7162

BR TST510 ; ;

7163 030760
7164 030760 020017 007417

LEXF22: ; TEST DATA SET LEXF-22:
.WORD 020017,ALT4P ; INITIAL AC FLOAT NUMBER

```

7165 030764 000000 000000
7166 030770 177577
7167 030772 045453 045444
7168 030776 000000
7169
7170
7171
7172
7173
7174
7175
7176 031000 000004
7177 031002 012705 031014
7178 031006 004737 041072
7179
7180 031012 000410
7181
7182 031014
7183 031014 120160 170360
7184 031020 000000 000000
7185 031024 177576
7186 031026 045513 045504
7187 031032 000000
7188
7189
7190
7191
7192
7193
7194
7195 031034 000004
7196 031036 012705 031050
7197 031042 004737 041072
7198
7199 031046 000410
7200
7201 031050
7202 031050 020177 177777
7203 031054 000000 000000
7204 031060 177575
7205 031062 045453 045444
7206 031066 000000
7207
7208
7209

```

```

.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 )

*****
:TEST 510 TEST OF LDEXP/F INSTR, DATA SET LEXF-23
:* UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
:* SHORT FLOAT, LONG INTEGER, ROUND MODES
*****
TST510: SCOPE
MOV #LEXF23,R5 ; PTR TO TEST DATA SET
JSR PC,@#LEXFT ; GO TEST
BR TST511 ;;

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 )

*****
:TEST 511 TEST OF LDEXP/F INSTR, DATA SET LEXF-24
:* UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
:* SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST511: SCOPE
MOV #LEXF24,R5 ; PTR TO TEST DATA SET
JSR PC,@#LEXFT ; GO TEST
BR TST512 ;;

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 )

```


M13

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(10.5) 25-APR-77 09:19 PAGE 147
T512 TEST OF LDEXP/D INSTR, DATA SET LEXD-1

SEQ 0314

```

7210
7211
7212
7213
7214
7215 031070 000004
7216 031072 012705 031104
7217 031076 004737 041242
7218
7219 031102 000414
7220
7221 031104
7222 031104 152325 052525 052525
7223 031112 052525
7224 031114 100325 052525 052525
7225 031122 052525
7226 031124 000201
7227 031126 047645 147652
7228 031132 100010
7229
7230
7231
7232
7233
7234
7235
7236 031134 000004
7237 031136 012705 031150
7238 031142 004737 041242
7239
7240 031146 000414
7241
7242 031150
7243 031150 052377 177777 177777
7244 031156 177777
7245 031160 000177 177777 177777
7246 031166 177777
7247 031170 000200
7248 031172 047711 147706
7249 031176 100010
7250
7251
7252
7253
7254
7255
7256
7257 031200 000004
7258 031202 012705 031214
7259 031206 004737 041242
7260
7261 031212 000414
7262
7263 031214
7264 031214 152360 170360 170360
7265 031222 170360

```

```

*****
:TEST 512 TEST OF LDEXP/D INSTR, DATA SET LEXD-1
:*
:* ALL INTERRUPT ENABLES ON
:* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****
TST512: SCOPE
MOV #LEXD1,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXDT ; GO TEST
BR TST513 ;;
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
*****
TST513: SCOPE
MOV #LEXD2,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXDT ; GO TEST
BR TST514 ;;
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
*****
TST514: SCOPE
MOV #LEXD3,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXDT ; GO TEST
BR TST515 ;;
LEXD3: ; TEST DATA SET LEXD-3:
.WORD 152360,ALT4N,ALT4N,ALT4N ; INITIAL AC FLOAT NUMBER

```

7266	031224	177760	170360	170360	.WORD	177760,ALT4N,ALT4N,ALT4N	; EXPECTED FLOAT RESULT
7267	031232	170360					
7268	031234	000177			.WORD	177	; EXPONENT TO BE LOADED
7269	031236	047607	047610		.WORD	047607,047610	; FPS: BEFORE, AFTER
7270	031242	000000			.WORD	NA	; FEC AFTER (0 = N/A)

```

7271
7272
7273
7274
7275
7276
7277
7278
7279
7280
7281
7282
7283
7284
7285
7286
7287
7288
7289
7290
7291
7292
7293
7294
7295
7296
7297
7298

```

```

*****
*TEST 515 TEST OF LDEXP/D INSTR, DATA SET LEXD-4
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

TST515: 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 )

```

7278	031244	000004		
7279	031246	012705	031260	
7280	031252	004737	041242	
7282	031256	000414		
7284	031260			
7285	031260	052200	000000	000000
7286	031266	000000		
7287	031270	060000	000000	000000
7288	031276	000000		
7289	031300	000100		
7290	031302	047757	047740	
7291	031306	000000		

```

*****
*TEST 516 TEST OF LDEXP/D INSTR, DATA SET LEXD-5
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

TST516: 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 )

```

7299	031310	000004		
7300	031312	012705	031324	
7301	031316	004737	041242	
7302				
7303	031322	000414		
7304				
7305	031324			
7306	031324	152252	125252	125252
7307	031332	125252		
7308	031334	140252	125252	125252
7309	031342	125252		
7310	031344	000001		
7311	031346	047647	047650	
7312	031352	000000		

```

7313
7314
7315
7316
7317
7318
7319

```

```

*****
*TEST 517 TEST OF LDEXP/D INSTR, DATA SET LEXD-6
* ALL INTERRUPT ENABLES ON
* LONG FLOAT, LONG INTEGER, ROUND MODES
*****

```

```

TST517: SCOPE
MOV #LEXD6,R5 ; PTR TO TEST DATA SET

```

7320	031354	000004		
7321	031356	012705	031370	

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 149
TS17 TEST OF LDEXP/D INSTR, DATA SET LEXD-6

SEQ 0316

7322 031362 004737 041242
7323
7324 031366 000414
7325
7326 031370
7327 031370 052217 007417 007417
7328 031376 007417
7329 031400 040017 007417 007417
7330 031406 007417
7331 031410 000000
7332 031412 047717 047700
7333 031416 000000
7334
7335
7336
7337
7338
7339
7340
7341 031420 000004
7342 031422 012705 031434
7343 031426 004737 041242
7344
7345 031432 000414
7346
7347 031434
7348 031434 152325 052525 052525
7349 031442 052525
7350 031444 137725 052525 052525
7351 031452 052525
7352 031454 177777
7353 031456 047607 047610
7354 031462 000000
7355
7356
7357
7358
7359
7360
7361
7362 031464 000004
7363 031466 012705 031500
7364 031472 004737 041242
7365
7366 031476 000414
7367
7368 031500
7369 031500 052377 177777 177777
7370 031506 177777
7371 031510 020177 177777 177777
7372 031516 177777
7373 031520 177700
7374 031522 047757 047740
7375 031526 000000
7376
7377

```

JSR PC,3#LEXDT ; GO TEST
BR TST520 ;;
LEXD6: ; TEST DATA SET LEXD-6:
.WORD 052217,ALT4P,ALT4P,ALT4P ; INITIAL AC FLOAT NUMBER
.WORD 040017,ALT4P,ALT4P,ALT4P ; EXPECTED FLOAT RESULT
.WORD 0 ; EXPONENT TO BE LOADED
.WORD 047717,047700 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
;TEST 520 TEST OF LDEXP/D INSTR, DATA SET LEXD-7
;
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, SHORT INTEGER, ROUND MODES
*****

```

```

↑TST520: SCOPE
MOV #LEXD7,R5 ; PTR TO TEST DATA SET
JSR PC,3#LEXDT ; GO TEST
BR TST521 ;;

```

```

LEXD7: ; TEST DATA SET LEXD-7:
.WORD 152325,ALTP,ALTP,ALTP ; INITIAL AC FLOAT NUMBER
.WORD 137725,ALTP,ALTP,ALTP ; EXPECTED FLOAT RESULT
.WORD -1 ; EXPONENT TO BE LOADED
.WORD 047607,047610 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

*****
;TEST 521 TEST OF LDEXP/D INSTR, DATA SET LEXD-10
;
; ALL INTERRUPT ENABLES ON
; LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

↑TST521: SCOPE
MOV #LEXD10,R5 ; PTR TO TEST DATA SET
JSR PC,3#LEXDT ; GO TEST
BR TST522 ;;

```

```

LEXD10: ; TEST DATA SET LEXD-10:
.WORD 052377,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD 020177,M1,M1,M1 ; EXPECTED FLOAT RESULT
.WORD -100 ; EXPONENT TO BE LOADED
.WORD 047757,047740 ; FPS: BEFORE, AFTER
.WORD NA ; FEC AFTER ( 0 = N/A )

```

```

7378
7379
7380
7381
7382
7383 031530 000004
7384 031532 012705 031544
7385 031536 004737 041242
7386
7387 031542 000414
7388
7389 031544
7390 031544 152360 170360 170360
7391 031552 170360
7392 031554 100360 170360 170360
7393 031562 170360
7394 031564 177601
7395 031566 047647 047650
7396 031572 000000
7397
7398
7399
7400
7401
7402
7403
7404 031574 000004
7405 031576 012705 031610
7406 031602 004737 041242
7407
7408 031606 000414
7409
7410 031610
7411 031610 052200 000000 000000
7412 031616 000000
7413 031620 000000 000000 000000
7414 031626 000000
7415 031630 177600
7416 031632 047713 147704
7417 031636 100012
7418
7419
7420
7421
7422
7423
7424
7425 031640 000004
7426 031642 012705 031654
7427 031646 004737 041242
7428
7429 031652 000414
7430
7431 031654
7432 031654 152252 125252 125252
7433 031662 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,ALTN,ALTN,ALTN ; INITIAL AC FLOAT NUMBER

```



```

7434 031664 177652 125252 125252 .WORD 177652,ALTN,ALTN,ALTN ; EXPECTED FLOAT RESULT
7435 031672 125252
7436 031674 177577 .WORD -201 ; EXPONENT TO BE LOADED
7437 031676 047607 147610 .WORD 047607,147610 ; FPS: BEFORE, AFTER
7438 031702 100012 .WORD 100012 ; FEC AFTER ( 0 = N/A )
7439
7440
7441
7442
7443
7444
7445

```

```

*****
:TEST 525 TEST OF LDEXP/D INSTR, DATA SET LEXD-14
:* ALL INTERRUPT ENABLES ON
:* LONG FLOAT, LONG INTEGER, TRUNCATE MODES
*****

```

```

7446 031704 000004
7447 031706 012705 031720
7448 031712 004737 041242
7449
7450 031716 000414
7451
7452
7453
7454
7455
7456
7457
7458
7459
7460
7461
7462
7463
7464
7465
7466

```

```

TST525: SCOPE
MOV #LEXD14,R5 ; PTR TO TEST DATA SET
JSR PC,@#LEXDT ; GO TEST
BR TST526 ;;

LEXD14: ; TEST DATA SET LEXD-14:
.WORD 052217,ALT4P,ALT4P,ALT4P ; INITIAL AC FLOAT NUMBER
.WORD 077417,ALT4P,ALT4P,ALT4P ; EXPECTED FLOAT RESULT
.WORD -202 ; EXPONENT TO BE LOADED
.WORD 047757,147740 ; FPS: BEFORE, AFTER
.WORD 100012 ; FEC AFTER ( 0 = N/A )

```

```

7467 031750 000004
7468 031752 012705 031764
7469 031756 004737 041242
7470
7471 031762 000414
7472
7473
7474
7475
7476
7477
7478
7479
7480
7481
7482
7483
7484
7485
7486
7487
7488
7489

```

```

*****
:TEST 526 TEST OF LDEXP/D INSTR, DATA SET LEXD-15
:* ALL INTERRUPT ENABLES ON
:* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
*****

```

```

TST526: SCOPE
MOV #LEXD15,R5 ; PTR TO TEST DATA SET
JSR PC,@#LEXDT ; GO TEST
BR TST527 ;;

LEXD15: ; TEST DATA SET LEXD-15:
.WORD 152325,ALTP,ALTP,ALTP ; INITIAL AC FLOAT NUMBER
.WORD 177325,ALTP,ALTP,ALTP ; EXPECTED FLOAT RESULT
.WORD -203 ; EXPONENT TO BE LOADED
.WORD 047647,147650 ; FPS: BEFORE, AFTER
.WORD 100012 ; FEC AFTER ( 0 = N/A )

```

```

7488 032014 000004
7489 032016 012705 032030

```

```

*****
:TEST 527 TEST OF LDEXP/D INSTR, DATA SET LEXD-16
:* ALL INTERRUPT ENABLES ON
:* LONG FLOAT, LONG INTEGER, ROUND MODES
*****
TST527: SCOPE
MOV #LEXD16,R5 ; PTR TO TEST DATA SET

```

7490 032022 004737 041242
 7491
 7492 032026 000414
 7493
 7494 032030
 7495 032030 177600 000000 000000
 7496 032036 000000
 7497 032040 140000 000000 000000
 7498 032046 000000
 7499 032050 000000
 7500 032052 047707 047710
 7501 032056 000000

JSR PC,3#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)

7502
 7503
 7504
 7505
 7506
 7507
 7508
 7509 032060 000004
 7510 032062 012705 032074
 7511 032066 004737 041242
 7512
 7513 032072 000414
 7514
 7515 032074
 7516 032074 152325 052525 052525
 7517 032102 052525
 7518 032104 000000 000000 000000
 7519 032112 000000
 7520 032114 000201
 7521 032116 046651 046646
 7522 032122 000000
 7523
 7524
 7525
 7526
 7527
 7528
 7529
 7530 032124 000004
 7531 032126 012705 032140
 7532 032132 004737 041242
 7533
 7534 032136 000414
 7535
 7536 032140
 7537 032140 052377 177777 177777
 7538 032146 177777
 7539 032150 000000 000000 000000
 7540 032156 000000
 7541 032160 000200
 7542 032162 046711 046706
 7543 032166 000000
 7544
 7545

 : TEST 530 TEST OF LDEXP/D INSTR, DATA SET LEXD-17
 : * OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
 : * LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
 : *****
 †TST530: SCOPE
 MOV #LEXD17,R5 ; PTR TO TEST DATA SET
 JSR PC,3#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)

7526
 7527
 7528
 7529
 7530 032124 000004
 7531 032126 012705 032140
 7532 032132 004737 041242
 7533
 7534 032136 000414
 7535
 7536 032140
 7537 032140 052377 177777 177777
 7538 032146 177777
 7539 032150 000000 000000 000000
 7540 032156 000000
 7541 032160 000200
 7542 032162 046711 046706
 7543 032166 000000
 7544
 7545

 : TEST 531 TEST OF LDEXP/D INSTR, DATA SET LEXD-20
 : * OVERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
 : * LONG FLOAT, LONG INTEGER, ROUND MODES
 : *****
 †TST531: SCOPE
 MOV #LEXD20,R5 ; PTR TO TEST DATA SET
 JSR PC,3#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)


```

7546
7547
7548
7549
7550
7551 032170 000004
7552 032172 012705 032204
7553 032176 004737 041242
7554
7555 032202 000414
7556
7557 032204
7558 032204 052201 000002 000003
7559 032212 000000
7560 032214 000000 000000 000000
7561 032222 000000
7562 032224 177600
7563 032226 045713 045704
7564 032232 000000
7565
7566
7567
7568
7569
7570
7571
7572 032234 000004
7573 032236 012705 032250
7574 032242 004737 041242
7575
7576 032246 000414
7577
7578 032250
7579 032250 152252 125252 125252
7580 032256 125252
7581 032260 000000 000000 000000
7582 032266 000000
7583 032270 177577
7584 032272 045613 045604
7585 032276 000000
7586
7587
7588
7589
7590
7591
7592
7593 032300 000004
7594 032302 012705 032314
7595 032306 004737 041242
7596
7597 032312 000414
7598
7599 032314
7600 032314 052217 007417 007417
7601 032322 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

```

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 154
T534 TEST OF LDEXP/D INSTR, DATA SET LEXD-23

SEQ 0321

7602 032324 000000 000000 000000 .WORD 0,0,0,0 ; EXPECTED FLOAT RESULT

7603 032332 000000

7604 032334 177576

7605 032336 045753 045744

7606 032342 000000

7607

7608

7609

7610

7611

7612

7613

7614 032344 000004

7615 032346 012705 032360

7616 032352 004737 041242

7617

7618 032356 000414

7619

7620 032360

7621 032360 152325 052525 052525

7622 032366 052525

7623 032370 000000 000000 000000

7624 032376 000000

7625 032400 177575

7626 032402 045653 045644

7627 032406 000000

7628

7629

7630

```
*****
: TEST 535 TEST OF LDEXP/D INSTR, DATA SET LEXD-24
: * UNDERFLOW INTERRUPT ENABLE OFF, ALL OTHERS ON
: * LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
: *****
```

```
TST535: SCOPE
MOV #LEXD24,R5 ; PTR TO TEST DATA SET
JSR PC,#LEXDT ; GO TEST
BR TST536 ;;
```

```
LEXD24: ; TEST DATA SET LEXD-24:
.WORD 152325,ALTP,ALTP,ALTP ; INITIAL AC FLOAT NUMBER
```

.WORD 0,0,0,0 ; EXPECTED FLOAT RESULT

.WORD -203 ; EXPONENT TO BE LOADED

.WORD 045653,045644 ; FPS: BEFORE, AFTER

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


```

7631
7632
7633
7634
7635
7636 032410 000004
7637 032412 012705 032424
7638 032416 004737 041432
7639
7640 032422 000405
7641
7642 032424
7643 032424 177777 177777
7644 032430 000177
7645 032432 047457 047440
7646
7647
7648
7649
7650
7651
7652
7653 032436 000004
7654 032440 012705 032452
7655 032444 004737 041432
7656
7657 032450 000405
7658
7659 032452
7660 032452 060052 125252
7661 032456 000100
7662 032460 047517 047500
7663
7664
7665
7666
7667
7668
7669
7670 032464 000004
7671 032466 012705 032500
7672 032472 004737 041432
7673
7674 032476 000405
7675
7676 032500
7677 032500 140270 107070
7678 032504 000001
7679 032506 047557 047540
7680
7681
7682
7683
7684
7685
7686

```

```

*****
: *TEST 536 TEST OF STEXP/F INSTR, DATA SET SEXF-1
: * ALL INTERRUPT ENABLES ON
: * SHORT FLOAT, SHORT INTEGER, TRUNCATE MODES
: *****
TST536: SCOPE
MOV #SEXF1,RS ; PTR TO TEST DATA SET
JSR PC,#SEXF1 ; GO TEST
BR TST537 ;;

```

```

SEXF1: ; TEST DATA SET SEXF-1:
.WORD MI,MI ; 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
: *****
TST537: SCOPE
MOV #SEXF2,RS ; PTR TO TEST DATA SET
JSR PC,#SEXF2 ; 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
: *****
TST540: SCOPE
MOV #SEXF3,RS ; PTR TO TEST DATA SET
JSR PC,#SEXF3 ; 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
: *****

```

FPU ADVANCED INSTR TESTS
DQFPBB.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 156
T541 TEST OF STEXP/F INSTR, DATA SET SEXF-4

SEQ 0323

7687	032512	000004	
7688	032514	012705	032526
7689	032520	004737	041432
7690			
7691	032524	000405	
7692			
7693	032526		
7694	032526	040125	007417
7695	032532	000000	
7696	032534	047413	047404
7697			
7698			
7699			
7700			
7701			
7702			
7703			
7704	032540	000004	
7705	032542	012705	032554
7706	032546	004737	041432
7707			
7708	032552	000405	
7709			
7710	032554		
7711	032554	137760	170360
7712	032560	177777	
7713	032562	047407	047410
7714			
7715			
7716			
7717			
7718			
7719			
7720			
7721	032566	000004	
7722	032570	012705	032602
7723	032574	004737	041432
7724			
7725	032600	000405	
7726			
7727	032602		
7728	032602	100307	070707
7729	032606	177601	
7730	032610	047507	047510
7731			
7732			
7733			
7734			
7735			
7736			
7737			
7738	032614	000004	
7739	032616	012705	032630
7740	032622	004737	041432
7741			
7742	032626	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 ;;

```


J14

FPU ADVANCED INSTR TESTS
DQFPBB.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 157
T544 TEST OF STEXP/F INSTR, DATA SET SEXF-7

SEQ 0324

7743
7744 032630
7745 032630 000000 000000
7746 032634 177600
7747 032636 047447 047450
7748
7749
7750

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

```

7751 .....
7752 ;*TEST 545 TEST OF STEXP/D INSTR, DATA SET SEXD-1
7753 ;*
7754 ;* ALL INTERRUPT ENABLES ON
7755 ;* LONG FLOAT, LONG INTEGER, ROUND MODES
7756 ;*.....
7756 032642 000004
7757 032644 012705 032656
7758 032650 004737 041566
7759
7760 032654 000407
7761
7762 032656
7763 032656 077600 000000 000000 SEXD1: ; TEST DATA SET SEXD-1:
7764 032664 000000 .WORD 077600,0,0,0 ; INITIAL AC FLOAT NUMBER
7765 032666 000177 .WORD 177 ; EXPONENT EXPECTED TO BE STORED
7766 032670 047717 047700 .WORD 047717,047700 ; FPS: BEFORE, AFTER
7767
7768
7769 .....
7770 ;*TEST 546 TEST OF STEXP/D INSTR, DATA SET SEXD-2
7771 ;*
7772 ;* ALL INTERRUPT ENABLES ON
7773 ;* LONG FLOAT, SHORT INTEGER, TRUNCATE MODES
7774 ;*.....
7774 032674 000004
7775 032676 012705 032710
7776 032702 004737 041566
7777
7778 032706 000407
7779
7780 032710
7781 032710 040360 170360 170360 SEXD2: ; TEST DATA SET SEXD-2:
7782 032716 170360 .WORD 040360,ALT4N,ALT4N,ALT4N ; INITIAL AC FLOAT NUMBER
7783 032720 000001 .WORD 1 ; EXPONENT EXPECTED TO BE STORED
7784 032722 047657 047640 .WORD 047657,047640 ; FPS: BEFORE, AFTER
7785
7786
7787 .....
7788 ;*TEST 547 TEST OF STEXP/D INSTR, DATA SET SEXD-3
7789 ;*
7790 ;* ALL INTERRUPT ENABLES ON
7791 ;* LONG FLOAT, SHORT INTEGER, ROUND MODES
7792 ;*.....
7792 032726 000004
7793 032730 012705 032742
7794 032734 004737 041566
7795
7796 032740 000407
7797
7798 032742
7799 032742 140107 070707 070707 SEXD3: ; TEST DATA SET SEXD-3:
7800 032750 070707 .WORD 140107,070707,070707,070707 ; INITIAL AC FLOAT NUMBER
7801 032752 000000 .WORD 0 ; EXPONENT EXPECTED TO BE STORED
7802 032754 047613 047604 .WORD 047613,047604 ; FPS: BEFORE, AFTER
7803
7804
7805 .....
7806 ;*TEST 550 TEST OF STEXP/D INSTR, DATA SET SEXD-4

```


L14

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 159
T550 TEST OF STEXP/D INSTR, DATA SET SEXD-4

SEQ 0326

7907
7908
7909
7910 032760 000004
7911 032762 012705 032774
7912 032766 004737 041566
7913
7914 032772 000407
7915
7916 032774
7917 032774 037652 125252 125252
7918 033002 125252
7919 033004 177777
7920 033006 047707 047710
7921
7922
7923
7924
7925
7926
7927
7928 033012 000004
7929 033014 012705 033026
7930 033020 004737 041566
7931
7932 033024 000407
7933
7934 033026
7935 033026 120070 107070 107070
7936 033034 107070
7937 033036 177700
7938 033040 047747 047750
7939
7940
7941
7942
7943
7944
7945
7946 033044 000004
7947 033046 012705 033060
7948 033052 004737 041566
7949
7950 033056 000407
7951
7952 033060
7953 033060 000217 007417 007417
7954 033066 007417
7955 033070 177601
7956 033072 047647 047650
7957
7958
7959
7960
7961
7962

```

;*          ALL INTERRUPT ENABLES ON
;*          LONG FLOAT, LONG INTEGER, ROUND MODES
;*****
TST550: SCOPE
MOV      #SEXD4,RS      ; 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
;*****
TST551: SCOPE
MOV      #SEXD5,RS      ; 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
;*****
TST552: SCOPE
MOV      #SEXD6,RS      ; 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
;*****

```

M14

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 160
T553 TEST OF STEXP/D INSTR, DATA SET SEXD-7

SEQ 0327

7863				
7864	033076	000004		
7865	033100	012705	033112	
7866	033104	004737	041566	
7867				
7868	033110	000407		
7869				
7870	033112			
7871	033112	000177	177777	177777
7872	033120	177777		
7873	033122	177600		
7874	033124	047607	047610	
7875				
7876				
7877				

```

*****
↑T553: SCOPE
MOV #SEXD7,R5 ; PTR TO TEST DATA SET
JSR PC,@#SEXDT ; GO TEST
BR T554 ;;
SEXD7: ; TEST DATA SET SEXD-7:
.WORD 000177,M1,M1,M1 ; INITIAL AC FLOAT NUMBER
.WORD -200 ; EXPONENT EXPECTED TO BE STORED
.WORD 047607,047610 ; FPS: BEFORE, AFTER

```



```

7878
7879
7880
7881 033130
7882 033130 000004
7883 033132 005037 001310
7884 033136 005037 001104
7885 033142 005037 001102
7886
7887
7888
7889
7890
7891
7892
7893
7894
7895
7896
7897 033146 076600 000022
7898 033152 032700 000020
7899 033156 001423
7900
7901 033160 032777 000002 145756
7902 033166 001017
7903
7904 033170 012701 010000
7905 033174 076600 000144
7906 033200 030100
7907 033202 001402
7908 033204 040100
7909 033206 000401
7910 033210 050100
7911 033212 076600 000344
7912
7913 033216 030100
7914 033220 001002
7915 033222 000137 003072
7916
7917
7918
7919
7920
7921
7922
7923
7924
7925
7926
7927
7928 033226
7929 033226 005037 001104
7930 033232 005037 001102
7931 033236 005037 001310
7932 033242 005237 001332
7933 033246 042737 100000 001332

```

```

;*****
.SBTTL SUB PASS END CONTROL

TST554: .FORCE LAST TEST NUMBER
        SCOPE          ;CHECK FOR TEST ITERATIONS HERE
        CLR            STIMES ;DONT ITERATE THIS "TEST"
        CLR            SERFLG ;NO ERRORS HERE
        CLR            STSTNM ;ZAP TEST ## WHEN DONE WITH A PASS

;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            SEOP ;EXIT IF NONE

        BIT            #SW01,SWR ;1=HFP OR WFP TEST ONLY
        BNE            SEOP ;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            SEOP ;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

SEOP:   CLR            SERFLG ;ZERO ERROR COUNT
        CLR            STSTNM ;ZERO TEST NUMBER
        CLR            STIMES ;ZERO NUMBER OF ITERATIONS
        INC            $PASS ;INCREMENT PASS COUNT
        BIC            #100000,$PASS ; BUT NEVER LET IN GO NEGATIVE

```

7934	033254	005327				DEC	(PC)+	:PASS LOOP ?
7935	033256	000001				SEOPCT: .WORD	1	:YES
7936	033260	003021				BGT	SDOAGN	:RESTORE COUNTER
7937	033262	012737				MOV	(PC)+,2(PC)+	
7938	033264	000001				SENDCT: .WORD	1	
7939	033266	033256				SEOPCT		
7940	033270	032777	002000	145646		BIT	#SW10,2SWR	:BELL ON PASS END ?
7941	033276	001002				BNE	SGET42	:NO
7942	033300	104401	001314			TYPE	,SBELL	:YES
7943								
7944	033304	013700	000042			SGET42: MOV	2#42,RO	:GET MONITOR ADDRESS
7945	033310	001405				BEQ	SDOAGN	:NO MONITOR
7946	033312	000005				RESET		:CLEAR WORLD
7947								
7948	033314	004710				SENDAD: JSR	PC,(RO)	:GO TO MONITOR
7949	033316	000240				NOP		:RESERVED FOR ACT11
7950	033320	000240				NOP		
7951	033322	000240				NOP		
7952								
7953	033324	000137	003034			SDOAGN: JMP	2#NEWPAS	:RETURN
7954								
7955								


```

7956          .SBTTL  SUBR TO TEST THE CMPF INSTRUCTION
7957
7958 033330    CMPFT:  MOV      #7,R0          ; LOAD STMPD-6
7959 033330    012700 000007      MOV      R5,R1          ; WITH TEST DATA SETS
7960 033334    010501          MOV      #STMPD,R2       ; FOR DISPLAY LATER
7961 033336    012702 001230      MOV      (R1)+(R2)+     ;
7962 033342    012122          SOB      R0,-2          ;
7963 033344    077002          MOV      #CMPFL,SLPERR ; ERROR LOOPING ADDRESS
7964 033346    012737 033354 001112
7965
7966 033354    CMPFL:  SETF      ; F MODE
7967 033356    170001          LDF      (R5),AC3      ; INITIAL AC FLOAT NUMBER
7968 033360    172715          LDFPS   10(R5)        ; INITIAL FPS
7969
7970 033364    173765 000004      CMPFI:  CMPF      4(R5),AC3 ; (MEM)-(AC3)
7971
7972 033370    170237 002000      STFPS   FPS          ; STORE FPS AFTER
7973 033374    170337 002002      STST    FEC          ; STORE FEC/FEA AFTER
7974 033400    174337 001170      STF     AC3,$REGO    ; STORE AC NUMBER
7975
7976 033404    023765 002000 000012  CMP      FPS,12(R5)   ; CHECK FPS
7977 033412    001401          BEQ     65$          ; FPS IS OK
7978 033414    104002          ERROR   2           ; FPS BAD
7979 033416    005765 000014      65$:  TST      14(R5)   ; DOES FEC/FEA APPLY?
7980 033422    100014          BPL     66$          ; NO - SKIP TEST
7981 033424    012737 033364 002014  MOV      #CMPFI,EXPFEA ; GET EXPECTED FEA
7982 033432    123765 002002 000014  CMPB    FEC,14(R5)   ; COMPARE FEC-S
7983 033440    001004          BNE     64$          ; NOT EQUAL
7984 033442    023737 002004 002014  CMP     FEA,EXPFEA   ; COMPARE FEA-S
7985 033450    001401          BEQ     66$          ; FEC, FEA OK
7986 033452    104012          64$:  ERROR   12     ; FEC OR FEA ARE BAD
7987 033454
7988
7989 033454    023715 001170      CMP     $REGO,(R5)   ; 1ST WORD OF RESULT CHECK?
7990 033460    001004          BNE     67$          ; NO
7991 033462    023765 001172 000002  CMP     $REG1,2(R5)  ; 2ND WORD OF RESULT CHECK?
7992 033470    001401          BEQ     68$          ; ALL WORDS OK
7993 033472    104021          67$:  ERROR   21     ; NUMBERS NOT EQUAL
7994 033474          68$:
7995
7996 033474    000207          RTS     PC          ; RETURN TO TEST CALLER
7997
7998
7999

```

```

;*****
.SBTTL  SUBR TO TEST THE CMPD INSTRUCTION

```

```

8000
8001
8002 033476    CMPDT:  MOV      #13,R0         ; LOAD STMPD-12
8003 033476    012700 000013      MOV      R5,R1         ; WITH TEST DATA SETS
8004 033502    010501          MOV      #STMPD,R2     ; FOR DISPLAY LATER
8005 033504    012702 001230      MOV      (R1)+(R2)+     ;
8006 033510    012122          SOB      R0,-2         ;
8007 033512    077002          MOV      #CMPDL,SLPERR ; ERROR LOOPING ADDRESS
8008 033514    012737 033522 001112
8009
8010 033522    CMPDL:  SETD      ; D MODE
8011 033524    170011          LOD     (R5),AC2     ; INITIAL AC FLOAT NUMBER
8011 033524    172615

```

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 164
SUBR TO TEST THE CMPD INSTRUCTION

SEQ 0331

8012	033524	170165	000020		LDFPS	20(R5)		; INITIAL FPS
8013								
8014	033532	173665	000010		CMPDI: CMPD	10(R5),AC2		; (MEM)-(AC2)
8015								
8016	033536	170237	002000		STFPS	FPS		; STORE FPS AFTER
8017	033542	170337	002002		STST	FEC		; STORE FEC/FEA AFTER
8018	033546	174237	001170		STD	AC2,\$REGO		; STORE AC AFTER
8019								
8020	033552	023765	002000	000022	CMP	FPS,22(R5)		; CHECK FPS
8021	033560	001401			BEQ	65\$; FPS IS OK
8022	033562	104005			ERROR	5		; FPS BAD
8023	033564	005765	000024	65\$:	TST	24(R5)		; DOES FEC/FEA APPLY?
8024	033570	100014			BPL	66\$; NO - SKIP TEST
8025	033572	012737	033532	002014	MOV	#CMPDI,EXPFEA		; GET EXPECTED FEA
8026	033600	123765	002002	000024	CMPB	FEC,24(R5)		; COMPARE FEC-S
8027	033606	001004			BNE	64\$; NOT EQUAL
8028	033610	023737	002004	002014	CMP	FEA,EXPFEA		; COMPARE FEA-S
8029	033616	001401			BEQ	66\$; FEC, FEA OK
8030	033620	104015		64\$:	ERROR	15		; FEC OR FEA ARE BAD
8031	033622			66\$:				
8032								
8033	033622	023715	001170		CMP	\$REGO,(R5)		; 1ST WORD OF RESULT CHECK?
8034	033626	001014			BNE	67\$; NO
8035	033630	023765	001172	000002	CMP	\$REG1,2(R5)		; 2ND WORD OF RESULT CHECK?
8036	033636	001010			BNE	67\$; NO
8037	033640	023765	001174	000004	CMP	\$REG2,4(R5)		; 3RD WORD OF RESULT CHECK?
8038	033646	001004			BNE	67\$; NO
8039	033650	023765	001176	000006	CMP	\$REG3,6(R5)		; 4TH WORD OF RESULT CHECK?
8040	033656	001401			BEQ	68\$; ALL WORDS OK
8041	033660	104022		67\$:	ERROR	22		; NUMBERS NOT EQUAL
8042	033662			68\$:				
8043								
8044	033662	000207			RTS	PC		; RETURN TO TEST CALLER
8045								


```

8046      .SBTTL SUBR TO TEST THE ADDF INSTRUCTION
8047
8048 033664      ADDFT:  MOV      #11,R0      ; LOAD STMP0-10
8049 033664 012700 000011      MOV      R5,R1      ; WITH TEST DATA SETS
8050 033670 010501      MOV      #STMP0,R2      ; FOR DISPLAY LATER
8051 033672 012702 001230      MOV      (R1)+,(R2)+
8052 033676 012122      SOB      R0,-2
8053 033700 077002      MOV      #ADDFL,SLPERR ; ERROR LOOPING ADDRESS
8054 033702 012737 033710 001112
8055
8056 033710 170001      ADDFL:  SETF      ; F MODE
8057 033712 172515      LDF      (R5),AC1    ; INITIAL AC FLOAT NUMBER
8058 033714 170165 000014      LDFPS   14(R5)      ; INITIAL FPS
8059
8060 033720 172165 000004      ADDFI:  ADDF      4(R5),AC1 ; (AC1)+(MEM)->AC1
8061
8062 033724 170237 002000      STFPS   FPS         ; STORE FPS AFTER
8063 033730 170337 002002      STST    FEC         ; STORE FEC/FEA AFTER
8064 033734 174137 001170      STF     AC1,$REG0   ; RESULT OF ADDF
8065
8066 033740 023765 002000 000016      CMP     FPS,16(R5)  ; CHECK FPS
8067 033746 001401      BEQ     65$         ; FPS IS OK
8068 033750 104004      ERROR   4          ; FPS BAD
8069 033752 005765 000020      65$:   TST     20(R5) ; DOES FEC/FEA APPLY?
8070 033756 100014      BPL     66$         ; NO - SKIP TEST
8071 033760 012737 033720 002014      MOV     #ADDFI,EXPFEA ; GET EXPECTED FEA
8072 033766 123765 002002 000020      CMPB   FEC,20(R5)  ; COMPARE FEC-S
8073 033774 001004      BNE     64$         ; NOT EQUAL
8074 033776 023737 002004 002014      CMP     FEA,EXPFEA ; COMPARE FEA-S
8075 034004 001401      BEQ     66$         ; FEC, FEA OK
8076 034006 104014      64$:   ERROR   14    ; FEC OR FEA ARE BAD
8077 034010      66$:
8078
8079 034010 023765 001170 000010      CMP     $REG0,10(R5) ; 1ST WORD OF RESULT CHECK?
8080 034016 001004      BNE     67$         ; NO
8081 034020 023765 001172 000012      CMP     $REG1,12(R5) ; 2ND WORD OF RESULT CHECK?
8082 034026 001401      BEQ     68$         ; ALL WORDS OK
8083 034030 104023      67$:   ERROR   23    ; NUMBERS NOT EQUAL
8084 034032      68$:
8085
8086 034032 000207      RTS     PC          ; RETURN TO TEST CALLER
8087
8088 ;:*****
8089 .SBTTL SUBR TO TEST THE ADDD INSTRUCTION
8090
8091 034034      ADDDT:  MOV      #17,R0      ; LOAD STMP0-16
8092 034034 012700 000017      MOV      R5,R1      ; WITH TEST DATA SETS
8093 034040 010501      MOV      #STMP0,R2      ; FOR DISPLAY LATER
8094 034042 012702 001230      MOV      (R1)+,(R2)+
8095 034046 012122      SOB      R0,-2
8096 034050 077002      MOV      #ADDDL,SLPERR ; ERROR LOOPING ADDRESS
8097 034052 012737 034060 001112
8098
8099 034060 170011      ADDDL:  SETD      ; D MODE
8100 034062 172415      LDD     (R5),AC0    ; INITIAL AC FLOAT NUMBER
8101 034064 170165 000030      LDFPS   30(R5)      ; INITIAL FPS

```



```

8135      .SBTTL SUBR TO TEST THE SUBF INSTRUCTION
8136
8137 034224 SUBFT:
8138 034224 012700 000011      MOV      #11,R0      ; LOAD STMPO-10
8139 034230 010501      MOV      R5,R1      ; WITH TEST DATA SETS
8140 034232 012702 001230      MOV      #STMPO,R2  ; FOR DISPLAY LATER
8141 034236 012122      MOV      (R1)+,(R2)+
8142 034240 077002      SOB      R0,-2
8143 034242 012737 034250 001112      MOV      #SUBFL,SLPERR ; ERROR LOOPING ADDRESS
8144
8145 034250 170001      SUBFL: SETF      ; F MODE
8146 034252 172415      LDF      (R5),AC0   ; INITIAL AC FLOAT NUMBER
8147 034254 170165 000014      LDFPS   14(R5)     ; INITIAL FPS
8148
8149 034260 173065 000004      SUBFI: SUBF      4(R5),AC0 ; (AC0)-(MEM)->AC0
8150
8151 034264 170237 002000      STFPS   FPS        ; STORE FPS AFTER
8152 034270 170337 002002      STST    FEC        ; STORE FEC/FEA AFTER
8153 034274 174037 001170      STF     AC0,$REGO  ; RESULT OF SUBF
8154
8155 034300 023765 002000 000016      CMP     FPS,16(R5)  ; CHECK FPS
8156 034306 001401      BEQ     65$        ; FPS IS OK
8157 034310 104004      ERROR  4          ; FPS BAD
8158 034312 005765 000020      65$:  TST     20(R5)  ; DOES FEC/FEA APPLY?
8159 034316 100014      BPL     66$        ; NO - SKIP TEST
8160 034320 012737 034260 002014      MOV     #SUBFI,EXPFEA ; GET EXPECTED FEA
8161 034326 123765 002002 000020      CMPB   FEC,20(R5)  ; COMPARE FEC-S
8162 034334 001004      BNE     64$        ; NOT EQUAL
8163 034336 023737 002004 002014      CMP     FEA,EXPFEA ; COMPARE FEA-S
8164 034344 001401      BEQ     66$        ; FEC, FEA OK
8165 034346 104014      64$:  ERROR  14     ; FEC OR FEA ARE BAD
8166 034350
8167
8168 034350 023765 001170 000010      CMP     $REGO,10(R5) ; 1ST WORD OF RESULT CHECK?
8169 034356 001004      BNE     67$        ; NO
8170 034360 023765 001172 000012      CMP     $REG1,12(R5) ; 2ND WORD OF RESULT CHECK?
8171 034366 001401      BEQ     68$        ; ALL WORDS OK
8172 034370 104023      67$:  ERROR  23     ; NUMBERS NOT EQUAL
8173 034372
8174
8175 034372 000207      RTS     PC          ; RETURN TO TEST CALLER
8176
8177 ;:*****
8178 .SBTTL SUBR TO TEST THE SUBD INSTRUCTION
8179
8180 034374 SUBDT:
8181 034374 012700 000017      MOV      #17,R0      ; LOAD STMPO-16
8182 034400 010501      MOV      R5,R1      ; WITH TEST DATA SETS
8183 034402 012702 001230      MOV      #STMPO,R2  ; FOR DISPLAY LATER
8184 034406 012122      MOV      (R1)+,(R2)+
8185 034410 077002      SOB      R0,-2
8186 034412 012737 034420 001112      MOV      #SUBDL,SLPERR ; ERROR LOOPING ADDRESS
8187
8188 034420 170011      SUBDL: SETD     ; D MODE
8189 034422 172715      LDD     (R5),AC3   ; INITIAL AC FLOAT NUMBER
8190 034424 170165 000030      LDFPS   30(R5)     ; INITIAL FPS

```



```

8224          .SBTTL  SUBR TO TEST THE MULF INSTRUCTION
8225
8226          MULFT:
8227          MOV      #11,R0          ; LOAD $TMP0-10
8228          MOV      R5,R1          ; WITH TEST DATA SETS
8229          MOV      #TMP0,R2       ; FOR DISPLAY LATER
8230          MOV      (R1)+(R2)+
8231          SOB      R0,-2
8232          MOV      #MULFL,$LPERR ; ERROR LOOPING ADDRESS
8233
8234          MULFL:  SETF          ; F MODE
8235          LDF      (R5),AC3       ; INITIAL AC FLOAT NUMBER
8236          LDFPS   14(R5)         ; INITIAL FPS
8237
8238          MULFI:  MULF      4(R5),AC3 ; (AC3)*(MEM)->AC3
8239
8240          STFPS   FPS           ; STORE FPS AFTER
8241          STST    FEC           ; STORE FEC/FEA AFTER
8242          STF     AC3,$REG0      ; RESULT OF MULF
8243
8244          CMP     FPS,16(R5)     ; CHECK FPS
8245          BEQ    65$            ; FPS IS OK
8246          ERROR  4              ; FPS BAD
8247          TST    20(R5)         ; DOES FEC/FEA APPLY?
8248          BPL    66$            ; NO - SKIP TEST
8249          MOV    #MULFI,EXPFEA ; GET EXPECTED FEA
8250          CMPB   FEC,20(R5)     ; COMPARE FEC-S
8251          BNE    64$            ; NOT EQUAL
8252          CMP    FEA,EXPFEA     ; COMPARE FEA-S
8253          BEQ    66$            ; FEC, FEA OK
8254          ERROR  14            ; FEC OR FEA ARE BAD
8255          64$:
8256          66$:
8257          CMP    $REG0,10(R5)   ; 1ST WORD OF RESULT CHECK?
8258          BNE    67$            ; NO
8259          CMP    $REG1,12(R5)   ; 2ND WORD OF RESULT CHECK?
8260          BEQ    68$            ; ALL WORDS OK
8261          ERROR  25            ; NUMBERS NOT EQUAL
8262          67$:
8263          68$:
8264          RTS     PC           ; RETURN TO TEST CALLER
8265
8266          ;*****
8267          .SBTTL  SUBR TO TEST THE MULD INSTRUCTION
8268
8269          MULDT:
8270          MOV      #17,R0          ; LOAD $TMP0-16
8271          MOV      R5,R1          ; WITH TEST DATA SETS
8272          MOV      #TMP0,R2       ; FOR DISPLAY LATER
8273          MOV      (R1)+(R2)+
8274          SOB      R0,-2
8275          MOV      #MULD, $LPERR ; ERROR LOOPING ADDRESS
8276
8277          MULD:   SETD          ; D MODE
8278          LDD      (R5),AC2       ; INITIAL AC FLOAT NUMBER
8279          LDFPS   30(R5)         ; INITIAL FPS

```



```

8313          .SBTTL SUBR TO TEST THE DIVF INSTRUCTION
8314
8315 035124          DIVFT:
8316 035124 012700 000011      MOV    #11,R0          ; LOAD STMP0-10
8317 035130 010501          MOV    R5,R1          ; WITH TEST DATA SETS
8318 035132 012702 001230      MOV    #STMP0,R2      ; FOR DISPLAY LATER
8319 035136 012122          MOV    (R1)+,(R2)+
8320 035140 077002          SOB    RO,-2
8321 035142 012737 035150 001112  MOV    #DIVFL,$LPERR ; ERROR LOOPING ADDRESS
8322
8323 035150 170001          DIVFL: SETF          ; F MODE
8324 035152 172615          LDF    (R5),AC2      ; INITIAL AC FLOAT NUMBER
8325 035154 170165 000014      LDFPS 14(R5)        ; INITIAL FPS
8326
8327 035160 174665 000004      DIVFI: DIVF         4(R5),AC2 ; (AC2)/(MEM)->AC2
8328
8329 035164 170237 002000          STFPS  FPS          ; STORE FPS AFTER
8330 035170 170337 002002          STST  FEC          ; STORE FEC/FEA AFTER
8331 035174 174237 001170          STF    AC2,$REGO    ; RESULT OF DIVF
8332
8333 035200 023765 002000 000016      CMP    FPS,16(R5)   ; CHECK FPS
8334 035206 001401          BEQ    65$          ; FPS IS OK
8335 035210 104004          ERROR 4           ; FPS BAD
8336 035212 005765 000020          65$: TST    20(R5)      ; DOES FEC/FEA APPLY?
8337 035216 100014          BPL    66$          ; NO - SKIP TEST
8338 035220 012737 035160 002014      MOV    #DIVFI,EXPFEA ; GET EXPECTED FEA
8339 035226 123765 002002 000020      CMPB  FEC,20(R5)   ; COMPARE FEC-S
8340 035234 001004          BNE    64$          ; NOT EQUAL
8341 035236 023737 002004 002014      CMP    FEA,EXPFEA  ; COMPARE FEA-S
8342 035244 001401          BEQ    66$          ; FEC, FEA OK
8343 035246 104014          64$: ERROR 14      ; FEC OR FEA ARE BAD
8344 035250          66$:
8345
8346 035250 023765 001170 000010      CMP    $REGO,10(R5) ; 1ST WORD OF RESULT CHECK?
8347 035256 001004          BNE    67$          ; NO
8348 035260 023765 001172 000012      CMP    $REG1,12(R5) ; 2ND WORD OF RESULT CHECK?
8349 035266 001401          BEQ    68$          ; ALL WORDS OK
8350 035270 104025          67$: ERROR 25      ; NUMBERS NOT EQUAL
8351 035272          68$:
8352
8353 035272 000207          RTS    PC          ; RETURN TO TEST CALLER
8354
8355 ;:*****
8356 ;.SBTTL SUBR TO TEST THE DIVD INSTRUCTION
8357
8358 035274          DIVDT:
8359 035274 012700 000017      MOV    #17,R0          ; LOAD STMP0-16
8360 035300 010501          MOV    R5,R1          ; WITH TEST DATA SETS
8361 035302 012702 001230      MOV    #STMP0,R2      ; FOR DISPLAY LATER
8362 035306 012122          MOV    (R1)+,(R2)+
8363 035310 077002          SOB    RO,-2
8364 035312 012737 035320 001112  MOV    #DIVDL,$LPERR ; ERROR LOOPING ADDRESS
8365
8366 035320 170011          DIVDL: SETD         ; D MODE
8367 035322 172515          LDD    (R5),AC1    ; INITIAL AC FLOAT NUMBER
8368 035324 170165 000030      LDFPS 30(R5)        ; INITIAL FPS

```


M15

FPU ADVANCED INSTR TESTS
 DGFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 173
 SUBR TO TEST THE MODF INSTRUCTION, USING 2 ACCUMULATORS

SEQ 0340

```

.SBTTL SUBR TO TEST THE MODF INSTRUCTION, USING 2 ACCUMULATORS

MD2FT:
8403
8404
8405 035464
8406 035464 012700 000013
8407 035470 010501
8408 035472 012702 001230
8409 035476 012122
8410 035500 077002
8411 035502 012737 035510 001112
8412
8413 035510 170001
8414 035512 172615
8415 035514 172737 002036
8416 035520 170165 000020
8417
8418 035524 171665 000004
8419
8420
8421 035530 170237 002000
8422 035534 170337 002002
8423 035540 174237 001170
8424 035544 174337 001174
8425
8426 035550 023765 002000 000022
8427 035556 001401
8428 035560 104005
8429 035562 005765 000024
8430 035566 100014
8431 035570 012737 035524 002014
8432 035576 123765 002002 000024
8433 035604 001004
8434 035606 023737 002004 002014
8435 035614 001401
8436 035616 104015
8437 035620
8438
8439
8440 035620 023765 001170 000010
8441 035626 001004
8442 035630 023765 001172 000012
8443 035636 001401
8444 035640 104027
8445 035642
8446
8447
8448 035642 023765 001174 000014
8449 035650 001004
8450 035652 023765 001176 000016
8451 035660 001401
8452 035662 104030
8453 035664
8454
8455 035664 000207
8456
8457
8458

MOV #13,R0 ; LOAD $TMP0-12
MOV R5,R1 ; WITH TEST DATA SETS
MOV #TMP0,R2 ; FOR DISPLAY LATER
MOV (R1)+(R2)+
SOB R0,-2
MOV #MD2FL,$LPERR ; ERROR LOOPING ADDRESS

MD2FL: SETF ; F MODE
LDF (R5),AC2 ; INITIAL AC FLOAT NUMBER
LDF PREVAC,AC3 ; FOR FEC-14 TEST
LDFPS 20(R5) ; INITIAL FPS

MD2FI: MODF 4(R5),AC2 ; FRAC[(AC2)*(MEM)]->AC2
; INT[(AC2)*(MEM)]->AC3

STFPS FPS ; STORE FPS AFTER
STST FEC ; STORE FEC/FEA AFTER
STF AC2,$REG0 ; STORE FRAC PART
STF AC3,$REG2 ; STORE INT PART

CMP FPS,22(R5) ; CHECK FPS
BEQ 65$ ; FPS IS OK
ERROR 5 ; FPS BAD
TST 24(R5) ; DOES FEC/FEA APPLY?
BPL 66$ ; NO - SKIP TEST
MOV #MD2FI,EXPFEA ; GET EXPECTED FEA
CMPB FEC,24(R5) ; COMPARE FEC-S
BNE 64$ ; NOT EQUAL
CMP FEA,EXPFEA ; COMPARE FEA-S
BEQ 66$ ; FEC, FEA OK
ERROR 15 ; FEC OR FEA ARE BAD

; CHECK FRACTION PART
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 27 ; NUMBERS NOT EQUAL

; CHECK INTEGER PART
CMP $REG2,14(R5) ; 1ST WORD OF RESULT CHECK?
BNE 69$ ; NO
CMP $REG3,16(R5) ; 2ND WORD OF RESULT CHECK?
BEQ 70$ ; ALL WORDS OK
ERROR 30 ; NUMBERS NOT EQUAL

RTS PC ; RETURN TO TEST CALLER

```

;;*****

```

.SBTTL SUBR TO TEST THE MODD INSTRUCTION, USING 2 ACCUMULATORS
MD2DT:
8459
8460
8461 035666
8462 035666 012700 000023
8463 035672 010501
8464 035674 012702 001230
8465 035700 012122
8466 035702 077002
8467 035704 012737 035712 001112
8468
8469 035712 170011
8470 035714 172415
8471 035716 172537 002036
8472 035722 170165 000040
8473
8474 035726 171465 000010
8475
8476
8477 035732 170237 002000
8478 035736 170337 002002
8479 035740 174037 001170
8480 035746 174137 001200
8481
8482 035752 023765 002000 000042
8483 035760 001401
8484 035762 104010
8485 035764 005765 000044
8486 035770 100014
8487 035772 012737 035726 002014
8488 036000 123765 002002 000044
8489 036006 001004
8490 036010 023737 002004 002014
8491 036016 001401
8492 036020 104020
8493 036022
8494
8495
8496 036022 023765 001170 000020
8497 036030 001014
8498 036032 023765 001172 000022
8499 036040 001010
8500 036042 023765 001174 000024
8501 036050 001004
8502 036052 023765 001176 000026
8503 036060 001401
8504 036062 104031
8505 036064
8506
8507
8508 036064 023765 001200 000030
8509 036072 001014
8510 036074 023765 001202 000032
8511 036102 001010
8512 036104 023765 001204 000034
8513 036112 001004
8514 036114 023765 001206 000036

MOV #23,R0 ; LOAD $TMP0-22
MOV R5,R1 ; WITH TEST DATA SETS
MOV #TMP0,R2 ; FOR DISPLAY LATER
MOV (R1)+,(R2)+
SOB RO,-2
MOV #MD2DL,$LPERR ; ERROR LOOPING ADDRESS

MD2DL: SETD ; D MODE
LD0 (R5),AC0 ; INITIAL AC FLOAT NUMBER
LD0 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
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$:
65$:
66$:
67$:
68$:

; 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 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?

```


B16

FPU ADVANCED INSTR TESTS
DQFPBB.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 175
SUBR TO TEST THE MODD INSTRUCTION, USING 2 ACCUMULATORS

SEQ 0342

8515 036122 001401
8516 036124 104032
8517 036126
8518
8519 036126 000207

69S: BEQ 70S
70S: ERROR 32
RTS PC

; ALL WORDS OK
; NUMBERS NOT EQUAL
; RETURN TO TEST CALLER

```

8520 .SBTTL SUBR TO TEST THE MODF INSTRUCTION, USING 1 ACCUMULATOR
8521
8522 036130 MD1FT:
8523 036130 012700 000013 MOV #13,RO ; LOAD STMP0-12
8524 036134 010501 MOV R5,R1 ; WITH TEST DATA SETS
8525 036136 012702 001230 MOV #STMP0,R2 ; FOR DISPLAY LATER
8526 036142 012122 MOV (R1)+,(R2)+
8527 036144 077002 SOB RO,-2
8528 036146 012737 036154 001112 MOV #MD1FL,$LPERR ; ERROR LOOPING ADDRESS
8529
8530 036154 170001 MD1FL: SETF ; F MODE
8531 036156 172715 LDF (R5),AC3 ; INITIAL AC FLOAT NUMBER
8532 036160 172637 002036 LDF PREVAC,AC2 ; AC2 SHOULD NOT CHANGE
8533 036164 170165 000020 LDFPS 20(R5) ; INITIAL FPS
8534
8535 036170 171765 000004 MD1FI: MODF 4(R5),AC3 ; FRAC[(AC3)*(MEM)]->AC3
8536 ; INT[(AC3)*(MEM)]->LOST
8537
8538 036174 170237 002000 STFPS FPS ; STORE FPS AFTER
8539 036200 170337 002002 STST FEC ; STORE FEC/FEA AFTER
8540 036204 174337 001170 STF AC3,$REG0 ; STORE FRAC PART
8541 036210 174237 001174 STF AC2,$REG2 ; STORE UNCHANGED AC3
8542
8543 036214 023765 002000 000022 CMP FPS,22(R5) ; CHECK FPS
8544 036222 001401 BEQ 65$ ; FPS IS OK
8545 036224 104005 ERROR 5 ; FPS BAD
8546 036226 005765 000024 65$: TST 24(R5) ; DOES FEC/FEA APPLY?
8547 036232 100014 BPL 66$ ; NO - SKIP TEST
8548 036234 012737 036170 002014 MOV #MD1FI,EXPFEA ; GET EXPECTED FEA
8549 036242 123765 002002 000024 CMPB FEC,24(R5) ; COMPARE FEC-S
8550 036250 001004 BNE 64$ ; NOT EQUAL
8551 036252 023737 002004 002014 CMP FEA,EXPFEA ; COMPARE FEA-S
8552 036260 001401 BEQ 66$ ; FEC, FEA OK
8553 036262 104015 64$: ERROR 15 ; FEC OR FEA ARE BAD
8554 036264 66$:
8555
8556 ; CHECK FRACTION PART
8557 036264 023765 001170 000010 CMP $REG0,10(R5) ; 1ST WORD OF RESULT CHECK?
8558 036272 001004 BNE 67$ ; NO
8559 036274 023765 001172 000012 CMP $REG1,12(R5) ; 2ND WORD OF RESULT CHECK?
8560 036302 001401 BEQ 68$ ; ALL WORDS OK
8561 036304 104027 67$: ERROR 27 ; NUMBERS NOT EQUAL
8562 036306 68$:
8563
8564 ; CHECK UNCHANGED PART
8565 036306 023765 001174 000014 CMP $REG2,14(R5) ; 1ST WORD OF RESULT CHECK?
8566 036314 001004 BNE 69$ ; NO
8567 036316 023765 001176 000016 CMP $REG3,16(R5) ; 2ND WORD OF RESULT CHECK?
8568 036324 001401 BEQ 70$ ; ALL WORDS OK
8569 036326 104030 69$: ERROR 30 ; NUMBERS NOT EQUAL
8570 036330 70$:
8571
8572 036330 000207 RTS PC ; RETURN TO TEST CALLER
8573
8574
8575

```

;;*****


```

      .SBTTL SUBR TO TEST THE MODD INSTRUCTION, USING 1 ACCUMULATOR
      MD1DT:
8576      MOV      #23,RO      ; LOAD STMP0-22
8577      MOV      R5,A1      ; WITH TEST DATA SETS
8578      MOV      #STMP0,R2  ; FOR DISPLAY LATER
8579      MOV      (R1)+(R2)+
8580      SOB      RO,-2      ;
8581      MOV      #MDIDL,SLPERR ; ERROR LOOPING ADDRESS
8582      MDIDL: SETD      ; D MODE
8583      LDD      (R5),AC1   ; INITIAL AC FLOAT NUMBER
8584      LDD      PREVAC,ACO ; ACO SHOULD NOT CHANGE
8585      LDFPS   40(R5)      ; INITIAL FPS
8586      MDIDI: MODD      10(R5),AC1 ; FRAC[(AC1)*(MEM)]->AC1
8587      ; INT[(AC1)*(MEM)]->LOST
8588      STFPS   FPS        ; STORE FPS AFTER
8589      STST   FEC        ; STORE FEC/FEA AFTER
8590      STD    AC1,$REG0   ; STORE FRAC PART
8591      STD    ACO,$REG4  ; STORE UNCHANGED ACO
8592      CMP    FPS,42(R5)  ; CHECK FPS
8593      BEQ    65$        ; FPS IS OK
8594      ERROR 10         ; FPS BAD
8595      TST   44(R5)      ; DOES FEC/FEA APPLY?
8596      BPL   66$        ; NO - SKIP TEST
8597      MOV   #MDIDI,EXPFEA ; GET EXPECTED FEA
8598      CMPB  FEC,44(R5)  ; COMPARE FEC-S
8599      BNE   64$        ; NOT EQUAL
8600      CMP   FEA,EXPFEA ; COMPARE FEA-S
8601      BEQ   66$        ; FEC, FEA OK
8602      ERROR 20         ; FEC OR FEA ARE BAD
8603      64$:
8604      65$:
8605      ; CHECK FRACTION PART OF RESULT
8606      CMP   $REG0,20(R5) ; 1ST WORD OF RESULT CHECK?
8607      BNE   67$        ; NO
8608      CMP   $REG1,22(R5) ; 2ND WORD OF RESULT CHECK?
8609      BNE   67$        ; NO
8610      CMP   $REG2,24(R5) ; 3RD WORD OF RESULT CHECK?
8611      BNE   67$        ; NO
8612      CMP   $REG3,26(R5) ; 4TH WORD OF RESULT CHECK?
8613      BEQ   68$        ; ALL WORDS OK
8614      ERROR 31         ; NUMBERS NOT EQUAL
8615      67$:
8616      68$:
8617      ; CHECK UNCHANGED PART
8618      CMP   $REG4,30(R5) ; 1ST WORD OF RESULT CHECK?
8619      BNE   69$        ; NO
8620      CMP   $REG5,32(R5) ; 2ND WORD OF RESULT CHECK?
8621      BNE   69$        ; NO
8622      CMP   $REG6,34(R5) ; 3RD WORD OF RESULT CHECK?
8623      BNE   69$        ; NO
8624      CMP   $REG7,36(R5) ; 4TH WORD OF RESULT CHECK?
8625      BNE   69$
8626      69$:
8627      69$:
8628      69$:
8629      69$:
8630      69$:
8631      69$:

```

E16

FPU ADVANCED INSTR TESTS
DQFPBB.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 178
SUBR TO TEST THE MODD INSTRUCTION, USING 1 ACCUMULATOR

SEQ 0345

8632 036566 001401
8633 036570 104032
8634 036572
8635
8636 036572 000207

69\$: BEQ 70\$
70\$: ERROR 32

RTS PC

; ALL WORDS OK
; NUMBERS NOT EQUAL

; RETURN TO TEST CALLER

F16

FPU ADVANCED INSTR TESTS
DQFPBB.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 179
SUBR TO TEST THE LDCDF INSTRUCTION

SEQ 0346

```

8637      .SBTTL SUBR TO TEST THE LDCDF INSTRUCTION
8638
8639 036574 LCFDT: MOV #11,R0 ; LOAD STMP0-10
8640 036574 012700 000011 MOV R5,R1 ; WITH TEST DATA SETS
8641 036600 010501 MOV #STMP0,R2 ; FOR DISPLAY LATER
8642 036602 012702 001230 MOV (R1)+,(R2)+ ;
8643 036606 012122 SOB RO,-2 ;
8644 036610 077002 MOV #LCDFL,$LPERR ; ERROR LOOPING ADDRESS
8645 036612 012737 036620 001112
8646
8647 036620 170011 LCFDL: SETD ; D MODE
8648 036622 172537 002036 LDF PREVAC,AC1 ; PREV CONTENTS TO ACC; FOR FEC-14 TEST
8649 036626 170165 000014 LDFPS 14(R5) ; INITIAL FPS
8650
8651 036632 177515 LCFDI: LDCDF (R5),AC1 ; DTOF[(MEM)]->AC1
8652
8653 036634 170237 002000 STFPS FPS ; STORE FPS AFTER
8654 036640 170337 002002 STST FEC ; STORE FEC/FEA AFTER
8655 036644 174137 001170 STF AC1,$REG0 ; STORE RESULT
8656
8657 036650 023765 002000 000016 CMP FPS,16(R5) ; CHECK FPS
8658 036656 001401 BEQ 65$ ; FPS IS OK
8659 036660 104004 ERROR 4 ; FPS BAD
8660 036662 005765 000020 65$: TST 20(R5) ; DOES FEC/FEA APPLY?
8661 036666 100014 BPL 66$ ; NO - SKIP TEST
8662 036670 012737 036632 002014 MOV #LCDFI,EXPFEA ; GET EXPECTED FEA
8663 036676 123765 002002 000020 CMPB FEC,20(R5) ; COMPARE FEC-S
8664 036704 001004 BNE 64$ ; NOT EQUAL
8665 036706 023737 002004 002014 CMP FEA,EXPFEA ; COMPARE FEA-S
8666 036714 001401 BEQ 66$ ; FEC, FEA OK
8667 036716 104014 64$: ERROR 14 ; FEC OR FEA ARE BAD
8668 036720 66$:
8669
8670 036720 023765 001170 000010 CMP $REG0,10(R5) ; 1ST WORD OF RESULT CHECK?
8671 036726 001004 BNE 67$ ; NO
8672 036730 023765 001172 000012 CMP $REG1,12(R5) ; 2ND WORD OF RESULT CHECK?
8673 036736 001401 BEQ 68$ ; ALL WORDS OK
8674 036740 104033 67$: ERROR 33 ; NUMBERS NOT EQUAL
8675 036742 68$:
8676
8677 036742 000207 RTS PC ; RETURN TO TEST CALLER
8678
8679
8680

```

```

8681 .SBTTL SUBR TO TEST THE LDCDF INSTRUCTION
8682
8683 036744 LCFDT: MOV #11,R0 ; LOAD STMP0-10
8684 036744 012700 000011 MOV R5,R1 ; WITH TEST DATA SETS
8685 036750 010501 MOV #STMP0,R2 ; FOR DISPLAY LATER
8686 036752 012702 001230 MOV (R1)+,(R2)+ ;
8687 036756 012122 SOB RO,-2 ;
8688 036760 077002 MOV #LCFDL,$LPERR ; ERROR LOOPING ADDRESS
8689 036762 012737 036770 001112
8690
8691 036770 170011 LCFDL: SETD ; D MODE
8692 036772 172637 002036 LDD PREVAC,AC2 ; PREV CONTENTS TO ACC; FOR FEC-14 TEST

```



```

8727      .SBTTL  SUBR TO TEST THE STCDF INSTRUCTION
8728
8729      SCDFI:
8730      MOV      #11,R0      ; LOAD STMPO-10
8731      MOV      R5,R1      ; WITH TEST DATA SETS
8732      MOV      #STMPO,R2   ; FOR DISPLAY LATER
8733      MOV      (R1)+,(R2)+
8734      SOB      R0,-2
8735      MOV      #SCDFL,$LPERR ; ERROR LOOPING ADDRESS
8736
8737      SCDFL:  SETD      ; D MODE
8738      LDD      (R5),AC3    ; INITIAL AC FLOAT NUMBER
8739      LDFPS   14(R5)     ; INITIAL FPS
8740
8741      SCDFI:  STCDF      AC3,$REG0 ; DTOF[(AC3)]->MEM
8742
8743      STFPS   FPS        ; STORE FPS AFTER
8744      STST    FEC        ; STORE FEC/FEA AFTER
8745
8746      CMP      FPS,16(R5)   ; CHECK FPS
8747      BEQ      65$         ; FPS IS OK
8748      ERROR   4           ; FPS BAD
8749      TST     20(R5)       ; DOES FEC/FEA APPLY?
8750      BPL     66$         ; NO - SKIP TEST
8751      MOV     #SCDFI,EXPFEA ; GET EXPECTED FEA
8752      CMPB   FEC,20(R5)   ; COMPARE FEC-S
8753      BNE     64$         ; NOT EQUAL
8754      CMP     FEA,EXPFEA  ; COMPARE FEA-S
8755      BEQ     66$         ; FEC, FEA OK
8756      ERROR   14         ; FEC OR FEA ARE BAD
8757      64$:
8758      66$:
8759      CMP     $REG0,10(R5) ; 1ST WORD OF RESULT CHECK?
8760      BNE     67$         ; NO
8761      CMP     $REG1,12(R5) ; 2ND WORD OF RESULT CHECK?
8762      BEQ     68$         ; ALL WORDS OK
8763      ERROR   33         ; NUMBERS NOT EQUAL
8764      67$:
8765      68$:
8766      RTS     PC          ; RETURN TO TEST CALLER
8767
8768
8769      ;*****
8770      .SBTTL  SUBR TO TEST THE STCFD INSTRUCTION
8771
8772      SCFDI:
8773      MOV      #12,R0      ; LOAD STMPO-11
8774      MOV      R5,R1      ; WITH TEST DATA SETS
8775      MOV      #STMPO,R2   ; FOR DISPLAY LATER
8776      MOV      (R1)+,(R2)+
8777      SOB      R0,-2
8778      MOV      #SCFDL,$LPERR ; ERROR LOOPING ADDRESS
8779
8780      SCFDL:  SETD      ; USE D MODE
8781      LDD      (R5),AC0    ; INITIAL F FLOAT NUM, FOLLOW W/JUNK
8782      LDFPS   20(R5)     ; INITIAL FPS

```

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 182
SUBR TO TEST THE STCFD INSTRUCTION

SEQ 0349

8783										
8784	037334	176037	001170			STCFD	AC0,\$REG0			; FTOD[(AC0)]->MEM
8785										
8786	037340	170237	002000			STFPS	FPS			; STORE FPS AFTER
8787										
8788	037344	023765	002000	000022		CMP	FPS,22(R5)			; CHECK FPS OK
8789	037352	001401				BEQ	64\$; OK, BRANCH
8790	037354	104005				ERROR	5			; FPS BAD
8791	037356				64\$:					
8792										
8793	037356	023765	001170	000010		CMP	\$REG0,10(R5)			; 1ST WORD OF RESULT CHECK?
8794	037364	001014				BNE	65\$; NO
8795	037366	023765	001172	000012		CMP	\$REG1,12(R5)			; 2ND WORD OF RESULT CHECK?
8796	037374	001010				BNE	65\$; NO
8797	037376	023765	001174	000014		CMP	\$REG2,14(R5)			; 3RD WORD OF RESULT CHECK?
8798	037404	001004				BNE	65\$; NO
8799	037406	023765	001176	000016		CMP	\$REG3,16(R5)			; 4TH WORD OF RESULT CHECK?
8800	037414	001401				BEQ	66\$; ALL WORDS OK
8801	037416	104035			65\$:	ERROR	35			; NUMBERS NOT EQUAL
8802	037420				66\$:					
8803										
8804	037420	000207				RTS	PC			; RETURN TO TEST CALLER


```

8805      .SBTTL  SUBR TO TEST THE LDCIF INSTRUCTION
8806
8807 037422          LCIFT:
8808 037422 012700 000005      MOV    #5,R0          ; LOAD STMP0-4
8809 037426 010501          MOV    R5,R1          ; WITH TEST DATA SETS
8810 037430 012702 001230      MOV    #STMP0,R2      ; FOR DISPLAY LATER
8811 037434 012122          MOV    (R1)+,(R2)+
8812 037436 077002          SOB    RO,-2
8813 037440 012737 037446 001112  MOV    #LCIFL,$LPERR ; ERROR LOOPING ADDRESS
8814
8815 037446 170165 000006      LCIFL: LDFPS 6(R5)      ; INITIAL FPS
8816
8817 037452 177215          LDCIF (R5),AC2      ; FI((MEM))->AC2
8818
8819 037454 170237 002000      STFPS FPS          ; STORE FPS AFTER
8820 037460 174237 001170      STF   AC2,$REG0    ; STORE RESULT
8821
8822 037464 023765 002000 000010  CMP    FPS,10(R5)   ; CHECK FPS OK
8823 037472 001401          BEQ   64$          ; OK, BRANCH
8824 037474 104001          ERROR 1          ; FPS BAD
8825 037476          64$:
8826
8827 037476 023765 001170 000002  CMP    $REG0,2(R5)  ; 1ST WORD OF RESULT CHECK?
8828 037504 001004          BNE   65$          ; NO
8829 037506 023765 001172 000004  CMP    $REG1,4(R5)  ; 2ND WORD OF RESULT CHECK?
8830 037514 001401          BEQ   66$          ; ALL WORDS OK
8831 037516 104036          65$: ERROR 36    ; NUMBERS NOT EQUAL
8832 037520          66$:
8833
8834 037520 000207          RTS    PC          ; RETURN TO TEST CALLER
8835
8836
8837
8838      ;:*****
8839      .SBTTL  SUBR TO TEST THE LDCID INSTRUCTION
8840
8841 037522          LCIDT:
8842 037522 012700 000007      MOV    #7,R0          ; LOAD STMP0-6
8843 037526 010501          MOV    R5,R1          ; WITH TEST DATA SETS
8844 037530 012702 001230      MOV    #STMP0,R2      ; FOR DISPLAY LATER
8845 037534 012122          MOV    (R1)+,(R2)+
8846 037536 077002          SOB    RO,-2
8847 037540 012737 037546 001112  MOV    #LCIDL,$LPERR ; ERROR LOOPING ADDRESS
8848
8849 037546 170165 000012      LCIDL: LDFPS 12(R5)   ; INITIAL FPS
8850
8851 037552 177315          LDCID (R5),AC3      ; D((MEM))->AC3
8852
8853 037554 170237 002000      STFPS FPS          ; STORE FPS AFTER
8854 037560 174337 001170      STD   AC3,$REG0    ; STORE RESULT
8855
8856 037564 023765 002000 000014  CMP    FPS,14(R5)   ; CHECK FPS OK
8857 037572 001401          BEQ   64$          ; OK, BRANCH
8858 037574 104003          ERROR 3          ; FPS BAD
8859 037576          64$:
8860 037576 023765 001170 000002  CMP    $REG0,2(R5)  ; 1ST WORD OF RESULT CHECK?

```

K16

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 184
SUBR TO TEST THE LDCID INSTRUCTION

SEQ 0351

```

8861 037604 001014      BNE 65$      ; NO
8862 037606 023765 001172 000004  CMP $REG1,4(R5) ; 2ND WORD OF RESULT CHECK?
8863 037614 001010      BNE 65$      ; NO
8864 037616 023765 001174 000006  CMP $REG2,6(R5) ; 3RD WORD OF RESULT CHECK?
8865 037624 001004      BNE 65$      ; NO
8866 037626 023765 001176 000010  CMP $REG3,10(R5) ; 4TH WORD OF RESULT CHECK?
8867 037634 001401      BEQ 66$      ; ALL WORDS OK
8868 037636 104037      65$: ERROR 37 ; NUMBERS NOT EQUAL
8869 037640      66$:
8870
8871 037640 000207      RTS PC      ; RETURN TO TEST CALLER
8872
8873
8874
8875

```

```

;*****
;SBTTL SUBR TO TEST THE LDCLF INSTRUCTION

```

```

8876
8877 037642      LCLFT:
8878 037642 012700 000006  MOV #6,R0    ; LOAD $TMP0-5
8879 037646 010501      MOV R5,R1    ; WITH TEST DATA SETS
8880 037650 012702 001230  MOV $TMP0,R2 ; FOR DISPLAY LATER
8881 037654 012122      MOV (R1)+,(R2)+
8882 037656 077002      SOB R0,-2
8883 037660 012737 037666 001112  MOV #LCLFL,$LPERR ; ERROR LOOPING ADDRESS
8884
8885 037666 170165 000010  LCLFL: LDFPS 10(R5) ; INITIAL FPS
8886
8887 037672 177015      LDCLF (R5),AC0 ; F[(MEM)(MEM)]->AC0
8888
8889 037674 170237 002000  STFPS FPS    ; STORE FPS AFTER
8890 037700 174037 001170  STF AC0,$REG0 ; STORE RESULT
8891
8892 037704 023765 002000 000012  CMP FPS,12(R5) ; CHECK FPS OK
8893 037712 001401      BEQ 64$      ; OK BRANCH
8894 037714 104002      64$: ERROR 2 ; FPS BAD
8895 037716
8896
8897 037716 023765 001170 000004  CMP $REG0,4(R5) ; 1ST WORD OF RESULT CHECK?
8898 037724 001004      BNE 65$      ; NO
8899 037726 023765 001172 000006  CMP $REG1,6(R5) ; 2ND WORD OF RESULT CHECK?
8900 037734 001401      BEQ 66$      ; ALL WORDS OK
8901 037736 104040      65$: ERROR 40 ; NUMBERS NOT EQUAL
8902 037740      66$:
8903
8904 037740 000207      RTS PC      ; RETURN TO TEST CALLER
8905
8906
8907

```

```

;*****
;SBTTL SUBR TO TEST THE LCLDL INSTRUCTION

```

```

8908
8909
8910 037742      LCLDT:
8911 037742 012700 000010  MOV #10,R0   ; LOAD $TMP0-7
8912 037746 010501      MOV R5,R1    ; WITH TEST DATA SETS
8913 037750 012702 001230  MOV $TMP0,R2 ; FOR DISPLAY LATER
8914 037754 012122      MOV (R1)+,(R2)+
8915 037756 077002      SOB R0,-2
8916 037760 012737 037766 001112  MOV #LCLDL,$LPERR ; ERROR LOOPING ADDRESS

```


FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 185
SUBR TO TEST THE LDCLD INSTRUCTION

SEQ 0352

8917										
8918	037766	170165	000014		LCLDL:	LDFPS	14(R5)		:	INITIAL FPS
8919										
8920	037772	177115				LDCLD	(R5),AC1		:	D[(MEM)(MEM)]->AC1
8921										
8922	037774	170237	002000			STFPS	FPS		:	STORE FPS AFTER
8923	040000	174137	001170			STD	AC1,\$REG0		:	STORE RESULT
8924										
8925	040004	023765	002000	000016		CMP	FPS,16(R5)		:	CHECK FPS OK
8926	040012	001401				BEQ	64\$:	OK BRANCH
8927	040014	104004				ERROR	4		:	FPS BAD
8928	040016				64\$:					
8929										
8930	040016	023765	001170	000004		CMP	\$REG0,4(R5)		:	1ST WORD OF RESULT CHECK?
8931	040024	001014				BNE	65\$:	NO
8932	040026	023765	001172	000006		CMP	\$REG1,6(R5)		:	2ND WORD OF RESULT CHECK?
8933	040034	001010				BNE	65\$:	NO
8934	040036	023765	001174	000010		CMP	\$REG2,10(R5)		:	3RD WORD OF RESULT CHECK?
8935	040044	001004				BNE	65\$:	NO
8936	040046	023765	001176	000012		CMP	\$REG3,12(R5)		:	4TH WORD OF RESULT CHECK?
8937	040054	001401				BEQ	66\$:	ALL WORDS OK
8938	040056	104041			65\$:	ERROR	41		:	NUMBERS NOT EQUAL
8939	040060				66\$:					
8940										
8941	040060	000207				RTS	PC		:	RETURN TO TEST CALLER

```

      .SBTTL SUBR TO TEST THE STCFI INSTRUCTION
SCFIT:
8942      040062      012700      000006      MOV      #6,R0      ; LOAD STMPO-5
8943      040062      010501      000006      MOV      R5,R1      ; WITH TEST DATA SETS
8944      040062      012700      000006      MOV      #STMPO,R2      ; FOR DISPLAY LATER
8945      040062      012700      000006      MOV      (R1)+,(R2)+
8946      040066      010501      000006
8947      040070      012702      001230      SOB      R0,-2
8948      040074      012122      000006
8949      040076      077002      000006
8950      040100      012737      040106      001112      MOV      #SCFIL,$LPERR ; ERROR LOOPING ADDRESS
8951      040106      170001      000006
8952      040110      172715      000006      SCFIL: SETF      ; F MODE
8953      040112      170165      000006      LDF      (R5),AC3    ; INITIAL AC FLOAT NUMBER
8954      040116      175737      001170      LDFPS   6(R5)       ; INITIAL FPS
8955      040122      013737      177776      001172      SCFII: STCFI     AC3,$REG0 ; I((AC3))->MEM
8956      040130      170237      002000
8957      040134      170337      002002      MOV      2#PS,$REG1  ; SAVE CC-S
8958      040140      023765      002000      000010      STFPS   FPS         ; STORE FPS AFTER
8959      040146      001401      104001      000010      STST    FEC         ; STORE FEC/FEA AFTER
8960      040150      104001      000012      CMP      FPS,10(R5)  ; CHECK FPS
8961      040152      005765      000012      BEQ     65$         ; FPS IS OK
8962      040156      100014      000012      ERROR   1          ; FPS BAD
8963      040160      012737      040116      002014      TST     12(R5)      ; DOES FEC/FEA APPLY?
8964      040166      123765      002002      000012      BPL     66$         ; NO - SKIP TEST
8965      040174      001004      002004      002014      MOV     #SCFII,EXPFEA ; GET EXPECTED FEA
8966      040176      023737      002004      002014      CMPB   FEC,12(R5)  ; COMPARE FEC-S
8967      040204      001401      104011      000011      BNE     64$         ; NOT EQUAL
8968      040206      104011      000011      CMP     FEA,EXPFEA  ; COMPARE FEA-S
8969      040210      000207      000011      BEQ     66$         ; FEC, FEA OK
8970      040216      042737      177760      001172      ERROR   11        ; FEC OR FEA ARE BAD
8971      040224      042737      177760      001174      MOV     FPS,$REG2   ; GET FPS, PS CC BITS ONLY
8972      040232      023737      001172      001174      BIC     #CCONLY,$REG1 ;
8973      040240      001401      104054      000054      BIC     #CCONLY,$REG2 ;
8974      040242      104054      000054      CMP     $REG1,$REG2 ; CC-S COPIED?
8975      040244      023765      001170      000004      BEQ     67$         ; NOT EQUAL, SIGNAL ERROR
8976      040246      001401      104054      000054      ERROR   54        ;
8977      040252      001401      104054      000054      CMP     $REG0,4(R5) ; INTEGER RESULT CHECK?
8978      040254      104042      000042      BEQ     68$         ; NOT EQUAL, SIGNAL ERROR
8979      040256      000207      000007      ERROR   42        ;
8980      040260      000207      000007      RTS     PC          ; RETURN TO TEST CALLER
8981      040266      012702      001230
8982      040266      012702      001230
8983      040266      012702      001230
8984      040266      012702      001230
8985      040266      012702      001230
8986      040266      012702      001230
8987      040266      012702      001230
8988      040266      012702      001230
8989      040266      012702      001230
8990      040266      012702      001230
8991      040266      012702      001230
8992      040266      012702      001230
8993      040266      012702      001230
8994      040266      012702      001230
8995      040266      012702      001230
8996      040266      012702      001230
8997      040266      012702      001230
;*****
.SBTTL SUBR TO TEST THE STCFI INSTRUCTION
SCDIT:
      MOV      #10,R0      ; LOAD STMPO-7
      MOV      R5,R1      ; WITH TEST DATA SETS
      MOV      #STMPO,R2  ; FOR DISPLAY LATER

```



```

8998 040272 012122      MOV      (R1)+,(R2)+      ;
8999 040274 077002      SOB      RO,-2           ;
9000 040276 012737 040304 001112      MOV      #SCDIL,SLPERR   ; ERROR LOOPING ADDRESS
9001
9002 040304 170011      SC0IL:  SETD             ; D MODE
9003 040306 172415      LDD      (R5),AC0        ; INITIAL AC FLOAT NUMBER
9004 040310 170165 000012      LDFPS   12(R5)          ; INITIAL FPS
9005
9006 040314 175437 001170      SC0II:  STCDI          ACO,$REG0 ; I((ACO))->MEM
9007
9008 040320 013737 177776 001172      MOV      2#PS,$REG1     ; SAVE CC-S
9009 040326 170237 002000      STFPS   FPS             ; STORE FPS AFTER
9010 040332 170337 002002      STST    FEC             ; STORE FEC/FEA AFTER
9011
9012 040336 023765 002000 000014      CMP      FPS,14(R5)     ; CHECK FPS
9013 040344 001401      BEQ     65$             ; FPS IS OK
9014 040346 104003      ERROR   3               ; FPS BAD
9015 040350 005765 000016      65$:   TST      16(R5)   ; DOES FEC/FEA APPLY?
9016 040354 100014      BPL     66$             ; NO - SKIP TEST
9017 040356 012737 040314 002014      MOV      #SCDII,EXPFEA ; GET EXPECTED FEA
9018 040364 123765 002002 000016      CMPB    FEC,16(R5)     ; COMPARE FEC-S
9019 040372 001004      BNE     64$             ; NOT EQUAL
9020 040374 023737 002004 002014      CMP      FEA,EXPFEA    ; COMPARE FEA-S
9021 040402 001401      BEQ     66$             ; FEC, FEA OK
9022 040404 104013      64$:   ERROR   13       ; FEC OR FEA ARE BAD
9023 040406
9024
9025 040406 013737 002000 001174      MOV      FPS,$REG2     ; GET FPS, PS CC BITS ONLY
9026 040414 042737 177760 001172      BIC     #CCONLY,$REG1  ;
9027 040422 042737 177760 001174      BIC     #CCONLY,$REG2  ;
9028 040430 023737 001172 001174      CMP     $REG1,$REG2    ; CC-S COPIED?
9029 040436 001401      BEQ     67$             ;
9030 040440 104054      ERROR   54             ; NOT EQUAL, SIGNAL ERROR
9031 040442      67$:
9032
9033 040442 023765 001170 000010      CMP     $REG0,10(R5)   ; INTEGER RESULT CHECK?
9034 040450 001401      BEQ     68$             ;
9035 040452 104043      ERROR   43             ; NOT EQUAL, SIGNAL ERROR
9036 040454      68$:
9037
9038 040454 000207      RTS     PC              ; RETURN TO TEST CALLER
9039
9040
9041 ;:*****
9042 .SBTTL SUBR TO TEST THE STCFL INSTRUCTION
9043
9044 SCFLT:
9045 040456 012700 000007      MOV     #7,RO           ; LOAD STMP0-6
9046 040462 010501      MOV     R5,R1           ; WITH TEST DATA SETS
9047 040464 012702 001230      MOV     #STMP0,R2       ; FOR DISPLAY LATER
9048 040470 012122      MOV     (R1)+,(R2)+    ;
9049 040472 077002      SOB     RO,-2           ;
9050 040474 012737 040502 001112      MOV     #SCFLL,SLPERR  ; ERROR LOOPING ADDRESS
9051
9052 SCFLL: SETF             ; F MODE
9053 040504 172515      LDF     (R5),AC1        ; INITIAL AC FLOAT NUMBER

```

```

9054 040506 170165 000010          LDFPS 10(R5)          ; INITIAL FPS
9055
9056 040512 175537 001170          SCFLI: STCFL AC1,$REG0 ; L[(AC1)]->MEM
9057
9058 040516 013737 177776 001174    MOV 2#PS,$REG2        ; SAVE CC-S
9059 040524 170237 002000          STFPS FPS              ; STORE FPS AFTER
9060 040530 170337 002002          STST FEC              ; STORE FEC/FEA AFTER
9061
9062 040534 023765 002000 000012    CMP FPS,12(R5)        ; CHECK FPS
9063 040542 001401                    BEQ 65$                ; FPS IS OK
9064 040544 104002                    ERROR 2                ; FPS BAD
9065 040546 005765 000014          65$: TST 14(R5)        ; DOES FEC/FEA APPLY?
9066 040552 100014                    BPL 66$                ; NO - SKIP TEST
9067 040554 012737 040512 002014    MOV 8SCFLI,EXPFEA     ; GET EXPECTED FEA
9068 040562 123765 002002 000014    CMPB FEC,14(R5)       ; COMPARE FEC-S
9069 040570 001004                    BNE 64$                ; NOT EQUAL
9070 040572 023737 002004 002014    CMP FEA,EXPFEA        ; COMPARE FEA-S
9071 040600 001401                    BEQ 66$                ; FEC, FEA OK
9072 040602 104012          64$: ERROR 12         ; FEC OR FEA ARE BAD
9073 040604          66$:
9074
9075 040604 013737 002000 001176    MOV FPS,$REG3         ; GET FPS, PS CC BITS ONLY
9076 040612 042737 177760 001174    BIC 8CCONLY,$REG2    ;
9077 040620 042737 177760 001176    BIC 8CCONLY,$REG3    ;
9078 040626 023737 001174 001176    CMP $REG2,$REG3      ; CC-S COPIED?
9079 040634 001401                    BEQ 67$                ;
9080 040636 104055          67$: ERROR 55         ; NOT EQUAL, SIGNAL ERROR
9081 040640
9082
9083 040640 023765 001170 000004    CMP $REG0,4(R5)      ; 1ST WORD OF RESULT CHECK?
9084 040646 001004                    BNE 68$                ; NO
9085 040650 023765 001172 000006    CMP $REG1,6(R5)     ; 2ND WORD OF RESULT CHECK?
9086 040656 001401                    BEQ 69$                ; ALL WORDS OK
9087 040660 104044          68$: ERROR 44         ; NUMBERS NOT EQUAL
9088 040662          69$:
9089
9090 040662 000207          RTS PC                ; RETURN TO TEST CALLER
9091
9092
9093
9094          ;:*****
          .SBTTL SUBR TO TEST THE STCDL INSTRUCTION
9095
9096          SCDLT:
9097 040664 012700 000011          MOV #11,R0            ; LOAD STMP0-10
9098 040670 010501                    MOV R5,R1             ; WITH TEST DATA SETS
9099 040672 012702 001230          MOV #STMP0,R2        ; FOR DISPLAY LATER
9100 040676 012122                    MOV (R1)+,(R2)+
9101 040700 077002                    SOB R0,-2
9102 040702 012737 040710 001112    MOV #SCDLL,$LPERR    ; ERROR LOOPING ADDRESS
9103
9104          SCDLL: SETD
9105 040710 170011                    LDD (R5),AC2          ; D MODE
9106 040712 172615                    LDFPS 14(R5)         ; INITIAL AC FLOAT NUMBER
9107 040714 170165 000014          ; INITIAL FPS
9108 040720 175637 001170          SCCLI: STCDL AC2,$REG0 ; L[(AC2)]->MEM, MEM
9109

```


9110	040724	013737	177776	001174	MOV	2#PS, SREG2	:	SAVE CC-S
9111	040732	170237	002000		STEPS	FPS	:	STORE FPS AFTER
9112	040736	170337	002002		STST	FEC	:	STORE FEC/FEA AFTER
9113								
9114	040742	023765	002000	000016	CMP	FPS, 16(RS)	:	CHECK FPS
9115	040750	001401			BEQ	65\$:	FPS IS OK
9116	040752	104004			ERROR	4	:	FPS BAD
9117	040754	005765	000020		65\$: TST	20(RS)	:	DOES FEC/FEA APPLY?
9118	040760	100014			BPL	66\$:	NO - SKIP TEST
9119	040762	012737	040720	002014	MOV	#SCOLI, EXPFEA	:	GET EXPECTED FEA
9120	040770	123765	002002	000020	CMPB	FEC, 20(RS)	:	COMPARE FEC-S
9121	040776	001004			BNE	64\$:	NOT EQUAL
9122	041000	023737	002004	002014	CMP	FEA, EXPFEA	:	COMPARE FEA-S
9123	041006	001401			BEQ	66\$:	FEC, FEA OK
9124	041010	104014			64\$: ERROR	14	:	FEC OR FEA ARE BAD
9125	041012				66\$:			
9126								
9127	041012	013737	002000	001176	MOV	FPS, SREG3	:	GET FPS, PS CC BITS ONLY
9128	041020	042737	177760	001174	BIC	#CCONLY, SREG2	:	
9129	041026	042737	177760	001176	BIC	#CCONLY, SREG3	:	
9130	041034	023737	001174	001176	CMP	SREG2, SREG3	:	CC-S COPIED?
9131	041042	001401			BEQ	67\$:	
9132	041044	104055			ERROR	55	:	NOT EQUAL, SIGNAL ERROR
9133	041046				67\$:			
9134								
9135	041046	023765	001170	000010	CMP	SREG0, 10(RS)	:	1ST WORD OF RESULT CHECK?
9136	041054	001004			BNE	68\$:	NO
9137	041056	023765	001172	000012	CMP	SREG1, 12(RS)	:	2ND WORD OF RESULT CHECK?
9138	041064	001401			BEQ	69\$:	ALL WORDS OK
9139	041066	104045			68\$: ERROR	45	:	NUMBERS NOT EQUAL
9140	041070				69\$:			
9141								
9142	041070	000207			RTS	PC	:	RETURN TO TEST CALLER

E01

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 190
SUBR TO TEST THE LDEXP INSTRUCTION, F MODE

SEQ 0357

```

9143      .SBTTL  SUBR TO TEST THE LDEXP INSTRUCTION, F MODE
9144
9145      LEXFT:  MOV      #10,RO      ; LOAD STMP0-7
9146      041072 012700 000010      MOV      R5,R1      ; WITH TEST DATA SETS
9147      041076 010501              MOV      #STMP0,R2   ; FOR DISPLAY LATER
9148      041100 012702 001230      MOV      (R1)+(R2)+
9149      041104 012122              SOB      RO,-2
9150      041106 077002              MOV      #LEXFL,SLPERR ; ERROR LOOPING ADDRESS
9151      041110 012737 041116 001112
9152
9153      LEXFL:  SETF      ; F MODE
9154      041120 170001              LDF      (R5),AC1   ; INITIAL FLOAT NUMBER
9155      041122 172515              LDFPS   12(R5)     ; INITIAL FPS
9156
9157      LEXFI:  LDEXP   10(R5),AC1 ; EXP: MEM -> AC1
9158
9159      STFPS   FPS      ; STORE FPS AFTER
9160      041132 170237 002000      STST    FEC        ; STORE FEC/FEA AFTER
9161      041136 170337 002002
9162      041142 023765 002000 000014      CMP     FPS,14(R5) ; CHECK FPS
9163      041150 001401              BEQ     65$        ; FPS IS OK
9164      041152 104003              ERROR   3          ; FPS BAD
9165      041154 005765 000016      65$:   TST     16(R5) ; DOES FEC/FEA APPLY?
9166      041160 100014              BPL     66$        ; NO - SKIP TEST
9167      041162 012737 041126 002014      MOV     #LEXFI,EXPFEA ; GET EXPECTED FEA
9168      041170 123765 002002 000016      CMPB   FEC,16(R5) ; COMPARE FEC-S
9169      041176 001004              BNE     64$        ; NOT EQUAL
9170      041200 023737 002004 002014      CMP     FEA,EXPFEA ; COMPARE FEA-S
9171      041206 001401              BEQ     66$        ; FEC, FEA OK
9172      041210 104013              64$:   ERROR   13   ; FEC OR FEA ARE BAD
9173      041212
9174
9175      STF     AC1,SREG0 ; STORE RESULTANT FLOAT NUMBER
9176      041216 023765 001170 000004      CMP     SREG0,4(R5) ; 1ST WORD OF RESULT CHECK?
9177      041224 001004              BNE     67$        ; NO
9178      041226 023765 001172 000006      CMP     SREG1,6(R5) ; 2ND WORD OF RESULT CHECK?
9179      041234 001401              BEQ     68$        ; ALL WORDS OK
9180      041236 104046              67$:   ERROR   46   ; NUMBERS NOT EQUAL
9181      041240
9182
9183      RTS     PC      ; RETURN TO TEST CALLER
9184
9185      ;*****
9186      .SBTTL  SUBR TO TEST THE LDEXP INSTRUCTION, D MODE
9187
9188      LEXDT:  MOV      #14,RO      ; LOAD STMP0-13
9189      041242 012700 000014      MOV      R5,R1      ; WITH TEST DATA SETS
9190      041246 010501              MOV      #STMP0,R2   ; FOR DISPLAY LATER
9191      041250 012702 001230      MOV      (R1)+(R2)+
9192      041254 012122              SOB      RO,-2
9193      041256 077002              MOV      #LEXDL,SLPERR ; ERROR LOOPING ADDRESS
9194      041260 012737 041266 001112
9195
9196      LEXDL:  SETD      ; D MODE
9197      041270 170011              LDD     (R5),AC0   ; INITIAL FLOAT NUMBER
9198      041272 172415              LDFPS   22(R5)     ; INITIAL FPS

```


G01

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 192
SUBR TO TEST THE STXP INSTRUCTION, F MODE

SEQ 0359

```

9232          .SBTTL  SUBR TO TEST THE STXP INSTRUCTION, F MODE
9233
9234 041432          SEXFT:
9235 041432 012700 000005      MOV      #5,R0          ; LOAD STMP0-4
9236 041436 010501          MOV      R5,R1          ; WITH TEST DATA SETS
9237 041440 012702 001230      MOV      #STMP0,R2       ; FOR DISPLAY LATER
9238 041444 012122          MOV      (R1)+,(R2)+
9239 041446 077002          SOB      R0,-2
9240 041450 012737 041456 001112  MOV      #SEXFL,SLPERR ; ERROR LOOPING ADDRESS
9241
9242 041456 170001          SEXFL: SETF          ; F MODE
9243 041460 172615          LDF      (R5),AC2      ; INITIAL FLOAT NUMBER
9244 041462 170165 000006      LDFPS   6(R5)         ; INITIAL FPS
9245
9246 041466 175237 001170          SEXFI: STXP      AC2,$REG0 ; EXP: AC2 -> MEM
9247
9248 041472 013737 177776 001172      MOV      2#PS,$REG1    ; GET PS RIGHT AWAY, FOR CC BITS
9249 041500 170237 002000          STFPS   FPS          ; STORE FPS AFTER
9250
9251 041504 023765 002000 000010      CMP      FPS,10(R5)    ; CHECK FPS OK
9252 041512 001401          BEQ          64$      ; OK, BRANCH
9253 041514 104001          ERROR     1         ; FPS BAD
9254 041516
9255 64$:
9256 041516 013737 002000 001174      MOV      FPS,$REG2     ; GET FPS, PS CC BITS ONLY
9257 041524 042737 177760 001172      BIC      #CCONLY,$REG1 ;
9258 041532 042737 177760 001174      BIC      #CCONLY,$REG2 ;
9259 041540 023737 001172 001174      CMP      $REG1,$REG2  ; CC-S COPIED?
9260 041546 001401          BEQ          65$      ;
9261 041550 104054          ERROR     54        ; NOT EQUAL, SIGNAL ERROR
9262 041552
9263 65$:
9264 041552 023765 001170 000004      CMP      $REG0,4(R5)  ; EXP CHECK?
9265 041560 001401          BEQ          66$      ;
9266 041562 104050          ERROR     50        ; NOT EQUAL, SIGNAL ERROR
9267 041564
9268 66$:
9269 041564 000207          RTS      PC          ; RETURN TO TEST CALLER
9270
9271 ;:*****
9272          .SBTTL  SUBR TO TEST THE STXP INSTRUCTION, D MODE
9273
9274 041566          SEXDT:
9275 041566 012700 000007      MOV      #7,R0          ; LOAD STMP0-6
9276 041572 010501          MOV      R5,R1          ; WITH TEST DATA SETS
9277 041574 012702 001230      MOV      #STMP0,R2       ; FOR DISPLAY LATER
9278 041600 012122          MOV      (R1)+,(R2)+
9279 041602 077002          SOB      R0,-2
9280 041604 012737 041612 001112  MOV      #SEXDL,SLPERR ; ERROR LOOPING ADDRESS
9281
9282 041612 170011          SEXDL: SETD         ; D MODE
9283 041614 172715          LDD      (R5),AC3      ; INITIAL FLOAT NUMBER
9284 041616 170165 000012      LDFPS   12(R5)        ; INITIAL FPS
9285
9286 041622 175337 001170          SEXDI: STXP      AC3,$REG0 ; EXP: AC3 -> MEM
9287

```


H01

FPU ADVANCED INSTR TESTS
DQFPBB.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 193
SUBR TO TEST THE STXP INSTRUCTION, D MODE

SEQ 0360

9288	041626	013737	177776	001172	MOV	FPS, SREG1	:	GET PS RIGHT AWAY
9289	041634	170237	002000		STFPS	FPS	:	STORE FPS AFTER
9290								
9291	041640	023765	002000	000014	CMP	FPS, 14(R5)	:	CHECK FPS OK
9292	041646	001401			BEQ	64\$:	OK BRANCH
9293	041650	104003			ERROR	3	:	FPS BAD
9294	041652							
9295								
9296	041652	013737	002000	001174	MOV	FPS, SREG2	:	GET FPS, PS CC BITS ONLY
9297	041660	042737	177760	001172	BIC	CCONLY, SREG1	:	
9298	041666	042737	177760	001174	BIC	CCONLY, SREG2	:	
9299	041674	023737	001172	001174	CMP	SREG1, SREG2	:	CC-S COPIED?
9300	041702	001401			BEQ	65\$:	
9301	041704	104054			ERROR	54	:	NOT EQUAL, SIGNAL ERROR
9302	041706							
9303								
9304	041706	023765	001170	000010	CMP	SREG0, 10(R5)	:	EXP CHECK?
9305	041714	001401			BEQ	66\$:	
9306	041716	104051			ERROR	51	:	NOT EQUAL, SIGNAL ERROR
9307	041720							
9308								
9309	041720	000207			RTS	PC	:	RETURN TO TEST CALLER

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 194
FPP UNEXPECTED TRAP CATCHER

SEQ 0361

9310			
9311			
9312	041722	010637	002012
9313	041726	012637	002006
9314	041732	012637	002010
9315	041736	170237	002000
9316	041742	170337	002002
9317	041746	104056	
9318	041750	013746	002010
9319	041754	013746	002006
9320	041760	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


```

9321      .SBTTL  SCOPE HANDLER ROUTINE
9322
9323      ;*****
9324      ;THIS ROUTINE CONTROLS THE LOOPING OF SUBTESTS. IT WILL INCREMENT
9325      ;AND LOAD THE TEST NUMBER($STNM) INTO THE DISPLAY REG.(DISPLAY<15:0>)
9326      ;THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
9327      ;$SW14=1      LOOP ON TEST
9328      ;$SW11=1      INHIBIT ITERATIONS
9329      ;$SW09=1      LOOP ON ERROR
9330      ;$SW08=1      LOOP ON TEST IN "$LPTST"
9331      ;CALL
9332      ;*      SCOPE      ;;SCOPE=IOT
9333
9334      $SCOPE:
9335      64$:
9336      041762 032777 040000 137154 1$:  BIT  #BIT14,$SWR      ;; LOOP ON PRESENT TEST?
9337      041770 001114      BNE  $OVER      ;; YES IF SW14=1
9338      ;*****START OF CODE FOR THE XOR TESTER*****
9339      041772 000416      $XTSTR: BR  6$      ;; IF RUNNING ON THE "XOR" TESTER CHANGE
9340      ;THIS INSTRUCTION TO A "NOP" (NOP=240)
9341      041774 013746 000004      MOV  2$ERRVEC,-($P)      ;; SAVE THE CONTENTS OF THE ERROR VECTOR
9342      042000 012737 042020 000004      MOV  85$,$ERRVEC      ;; SET FOR TIMEOUT
9343      042006 005737 177060      TST  2$177060      ;; TIME OUT ON XOR?
9344      042012 012637 000004      MOV  ($P)+,$ERRVEC      ;; RESTORE THE ERROR VECTOR
9345      042016 000463      BR   $SVLAD      ;; GO TO THE NEXT TEST
9346      042020 022626      5$:  CMP  ($P)+,$($P)+      ;; CLEAR THE STACK AFTER A TIME OUT
9347      042022 012637 000004      MOV  ($P)+,$ERRVEC      ;; RESTORE THE ERROR VECTOR
9348      042026 000423      BR   7$      ;; LOOP ON THE PRESENT TEST
9349      042030      6$: ;*****END OF CODE FOR THE XOR TESTER*****
9350      042030 032777 000400 137106      BIT  #BIT08,$SWR      ;; LOOP ON SPEC. TEST?
9351      042036 001404      BEQ  2$      ;; BR IF NO
9352      042040 023737 001150 001102      CMP  $LPTST,$STNM      ;; ON THE RIGHT TEST?
9353      042046 001465      BEQ  $OVER      ;; BR IF YES
9354      042050 005737 001104      2$:  TST  $ERFLG      ;; HAS AN ERROR OCCURRED?
9355      042054 001421      BEQ  3$      ;; BR IF NO
9356      042056 023737 001120 001104      CMP  $ERMAX,$ERFLG      ;; MAX. ERRORS FOR THIS TEST OCCURRED?
9357      042064 101015      BHI  3$      ;; BR IF NO
9358      042066 032777 001000 137050      BIT  #BIT09,$SWR      ;; LOOP ON ERROR?
9359      042074 001404      BEQ  4$      ;; BR IF NO
9360      042076 013737 001112 001110 7$:  MOV  $LPERR,$LPADR      ;; SET LOOP ADDRESS TO LAST SCOPE
9361      042104 000446      BR   $OVER
9362      042106 005037 001104      4$:  CLR  $ERFLG      ;; ZERO THE ERROR FLAG
9363      042112 005037 001310      CLR  $STIMES      ;; CLEAR THE NUMBER OF ITERATIONS TO MAKE
9364      042116 000415      BR   1$      ;; ESCAPE TO THE NEXT TEST
9365      042120 032777 004000 137016 3$:  BIT  #BIT11,$SWR      ;; INHIBIT ITERATIONS?
9366      042126 001011      BNE  1$      ;; BR IF YES
9367      042130 005737 001332      TST  $PASS      ;; IF FIRST PASS OF PROGRAM
9368      042134 001406      BEQ  1$      ;; INHIBIT ITERATIONS
9369      042136 005237 001106      INC  $ICNT      ;; INCREMENT ITERATION COUNT
9370      042142 023737 001310 001106      CMP  $STIMES,$ICNT      ;; CHECK THE NUMBER OF ITERATIONS MADE
9371      042150 002024      BGE  $OVER      ;; BR IF MORE ITERATION REQUIRED
9372      042152 012737 000001 001106 1$:  MOV  81,$ICNT      ;; REINITIALIZE THE ITERATION COUNTER
9373      042160 013737 042236 001310      MOV  $MXCNT,$STIMES      ;; SET NUMBER OF ITERATIONS TO DO
9374      042166 005237 001102      $SVLAD: INC  $STNM      ;; COUNT TEST NUMBERS
9375      042172 013737 001102 001330      MOV  $STNM,$STSTN      ;; SET TEST NUMBER IN APT MAILBOX
9376      042200 011637 001110      MOV  ($P),$LPADR      ;; SAVE SCOPE LOOP ADDRESS

```

K01

FPU ADVANCED INSTR TESTS
DQFPBB.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 196
SCOPE HANDLER ROUTINE

SEQ 0363

9377	042204	011637	001112
9378	042210	005037	001312
9379	042214	012737	000001
9380	042222	013777	001102
9381	042230	013716	001110
9382	042234	000002	
9383	042236	003720	

001120
136716

SOVER:

SMXCNT: 2000.

```

MOV (SP), SLPERR
CLR SESCPE
MOV #1, SERMAX
MOV STSTN, DISPLAY
MOV SLPADR, (SP)
RTI

```

```

:: SAVE ERROR LOOP ADDRESS
:: CLEAR THE ESCAPE FROM ERROR ADDRESS
:: ONLY ALLOW ONE(1) ERROR ON NEXT TEST
:: DISPLAY TEST NUMBER
:: FUDGE RETURN ADDRESS
:: FIXES PS
:: MAX. NUMBER OF ITERATIONS

```



```

9384
9385
9386
9387
9388
9389
9390
9391
9392
9393
9394
9395
9396
9397
9398 042240
9399 042240 010037 002016
9400 042244 010137 002020
9401 042250 010237 002022
9402 042254 010337 002024
9403 042260 010437 002026
9404 042264 010537 002030
9405 042270 010637 002032
9406 042274 062737 000004 002032
9407 042302 011637 002034
9408 042306 005237 001104
9409 042312 001775
9410 042314 013777 001102 136624
9411 042322 032777 002000 136614
9412 042330 001402
9413 042332 104401 001314
9414 042336 005237 001114
9415 042342 011637 001122
9416 042346 162737 000002 001122
9417 042354 117737 136542 001116
9418 042362 032777 020000 136554
9419 042370 001004
9420 042372 004737 042502
9421 042376 104401 001321
9422 042402
9423 042402 122737 000001 001344
9424 042410 001007
9425 042412 113737 001116 042424
9426 042420 004737 043216
9427 042424 000
9428 042425 000
9429 042426 000777
9430 042430 005777 136510
9431 042434 100001
9432 042436 000000
9433 042440 032777 001000 136476
9434 042446 001402
9435 042450 013716 001112
9436 042454 005737 001312
9437 042460 001402
9438 042462 013716 001312
9439 042466

```

.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

```

SERROR:

```

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 SERFLG        SET THE ERROR FLAG
BEQ 7$            DON'T LET THE FLAG GO TO ZERO
MOV $STNM,$DISP  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
JSR PC,$TYPERR  ;;GO TO USER ERROR ROUTINE
TYPE $SCRLF

20$: CMPB #APTENV,$ENV ;;RUNNING IN APT MODE
BNE 2$           NO SKIP APT ERROR REPORT
MOVB $ITEMB,$I1$ ;;SET ITEM NUMBER AS ERROR NUMBER
JSR PC,$SATY4   ;;REPORT FATAL ERROR TO APT

21$: .BYTE 0
      .BYTE 0

22$: BR 22$      ;;APT ERROR LOOP
2$: 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 $ESCAPE,(SP) ;;FUDGE RETURN ADDRESS FOR ESCAPE
5$:

```

MO1

FPU ADVANCED INSTR TESTS
DQFPBB.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 198
ERROR HANDLER ROUTINE

SEQ 0365

9440	042466	022737	033314	000042	CMP	SENDAD, 2#42	:::ACT-11 AUTO-ACCEPT?
9441	042474	001001			BNE	65	:::BRANCH IF NO
9442	042476	000000			HALT		:::YES
9443	042500			65:			
9444	042500	000002		645:	RTI		;RETURN


```

9445
9446
9447
9448
9449
9450
9451
9452
9453
9454
9455
9456
9457
9458
9459 042502
9460 042502 104401
9461 042504 001321
9462 042506 010046
9463 042510 010146
9464 042512 005000
9465 042514 153700 001116
9466 042520 001004
9467
9468 042522 013746 001122
9469 042526 104402
9470 042530 000452
9471 042532 005300
9472 042534 006300
9473 042536 010001
9474 042540 006300
9475 042542 060100
9476 042544 062700 001354
9477 042550 012037 042560
9478 042554 001404
9479 042556 104401
9480 042560 000000
9481 042562 104401 001321
9482 042566 104401 042676
9483 042572 012037 042602
9484 042576 001402
9485 042600 104401
9486 042602 000000
9487 042604 104401 001321
9488 042610 017746 000054
9489 042614 104402
9490 042616 104401 042674
9491 042622 017746 000044
9492 042626 104402
9493 042630 104401 042674
9494 042634 011000
9495 042636 001407
9496 042640 013046
9497 042642 104402
9498 042644 005710
9499 042646 001403
9500 042650 104401 042674

```

;;*****

.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",
: *(SERRTB) THE ERROR MESSAGE, DATA HEADER, AND DATA VALUES TO PRINT.
: *THIS ROUTINE IS IDENTICAL TO THE SYSMAC ROUTINE SERRTYP, 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 SERRPC AS THE FIRST TWO DATA ELEMENTS
: *(WITH APPROPRIATE HEADERS).

```

STYPERR:

```

HOTWRM: TYPE .WORD SCRLF
MOV RO,-(SP)
MOV RI,-(SP)
CLR RO
BISB @($ITEMB,RO)
BNE IS
IF ITEM NUMBER FROM ERROR 0,
JUST TYPE PC OF ERROR
GET ERROR PC FOR TYPEOUT
TYPE OCTAL, ALL DIGITS
EXIT
ADJUST ERROR # FOR TABLE INDEX
OF 6 BYTES/ENTRY

FORM TABLE PTR
PICKUP "ERROR MESSAGE" PTR
SKIP TYPEOUT IF NULL
TYPE "ERROR MESSAGE"
"ERROR MESSAGE" PTR HERE
CR & LF
"TEST # ERR PC" HEADER
PICKUP "DATA HEADER" PTR
SKIP TYPEOUT IF NULL
TYPE "DATA HEADER"
"DATA HEADER" PTR HERE
CR & LF
($TESTN)
OCTAL W/ LEADING ZEROS
<HT>
(SERRPC)
OCTAL W/ LEADING ZEROS
<HT>
PICKUP "DATA TABLE" PTR
EXIT IF NULL
SAVE ... FOR TYPEOUT
TYPE OCTAL, ALL DIGITS
ANOTHER NUMBER ?
NO - EXIT
TAB BETWEEN ELEMENTS

IS: DEC RO
ASL RO
MOV RO,RI
ASL RO
ADD RI,RO
ADD @SERRTB,RO
MOV (RO)+,2$
BEQ 3$

2$: .WORD 0
TYPE ,SCRLF

3$: TYPE ,11$
MOV (RO)+,4$
BEQ 5$

4$: .WORD 0
5$: TYPE ,SCRLF
MOV @8$,-(SP)
TYPOC
TYPE ,10$
MOV (RO),RO
BEQ 7$

6$: MOV @2(RO)+,-(SP)
TYPOC
TST (RO)
BEQ 7$
TYPE ,10$

```



```

9569 043102 000770          BR      7S          ;;LOOP
9570
9571          ;HORIZONTAL TAB PROCESSOR
9572
9573 043104 112716 000040      BS:     MOVB    #' (SP)          ;;REPLACE TAB WITH SPACE
9574 043110 004737 043130      9S:     JSR     PC,STYPEC          ;;TYPE A SPACE
9575 043114 132737 000007 043174  BITB    #' ,SCHARCNT          ;;BRANCH IF NOT AT
9576 043122 001372          BNE     9S          ;;TAB STOP
9577 043124 005726          TST     (SP)+          ;;POP SPACE OFF STACK
9578 043126 000724          BR      2S          ;;GET NEXT CHARACTER
9579 043130 105777 136022      STYPEC: TSTB   STPS          ;;WAIT UNTIL PRINTER IS READY
9580 043134 100375          BPL     STYPEC
9581 043136 116677 000002 136014  MOVB    2(SP),STPB          ;;LOAD CHAR TO BE TYPED INTO DATA REG.
9582 043144 122766 000015 000002  CMPB    #CR,2(SP)          ;;IS CHARACTER A CARRIAGE RETURN?
9583 043152 001003          BNE     IS          ;;BRANCH IF NO
9584 043154 105037 043174          CLRB   SCHARCNT          ;;YES--CLEAR CHARACTER COUNT
9585 043160 000406          BR      STYPEX          ;;EXIT
9586 043162 122766 000012 000002  IS:     CMPB    #LF,2(SP)          ;;IS CHARACTER A LINE FEED?
9587 043170 001402          BEQ    STYPEX          ;;BRANCH IF YES
9588 043172 105227          INCB   (PC)+          ;;COUNT THE CHARACTER
9589 043174 000000      SCHARCNT: .WORD 0          ;;CHARACTER COUNT STORAGE
9590 043176 000207      STYPEX: RTS    PC
9591

```


.SBTTL APT COMMUNICATIONS ROUTINE

```

9592
9593
9594
9595 043200 112737 000001 043444 SATY1: MOV 81,SFFLG ;;TO REPORT FATAL ERROR
9596 043206 112737 000001 043442 SATY3: MOV 81,SMFLG ;;TO TYPE A MESSAGE
9597 043214 000403 BR SATYC
9598 043216 112737 000001 043444 SATY4: MOV 81,SFFLG ;;TO ONLY REPORT FATAL ERROR
9599 043224 SATYC:
9600 043224 010046 MOV R0,-(SP) ;;PUSH R0 ON STACK
9601 043226 010146 MOV R1,-(SP) ;;PUSH R1 ON STACK
9602 043230 105737 043442 TST SMFLG ;;SHOULD TYPE A MESSAGE?
9603 043234 001450 BEQ 55 ;;IF NOT: BR
9604 043236 122737 000001 001344 CMPB #APTENV,SENV ;;OPERATING UNDER APT?
9605 043244 001031 BNE 35 ;;IF NOT: BR
9606 043246 132737 000100 001345 BITB #APTPOOL,SENV ;;SHOULD SPOOL MESSAGES?
9607 043254 001425 BEQ 35 ;;IF NOT: BR
9608 043256 017600 000004 MOV #4(SP),R0 ;;GET MESSAGE ADDR.
9609 043262 062766 000002 000004 ADD #2,4(SP) ;;BUMP RETURN ADDR.
9610 043270 005737 001324 15: TST SMSGTYPE ;;SEE IF DONE W/ LAST XMISSION?
9611 043274 001375 BNE 15 ;;IF NOT: WAIT
9612 043276 010037 001340 MOV R0,SMSGAD ;;PUT ADDR IN MAILBOX
9613 043302 105720 25: TSTB (R0)+ ;;FIND END OF MESSAGE
9614 043304 001376 BNE 25
9615 043306 163700 001340 SUB SMSGAD,R0 ;;SUB START OF MESSAGE
9616 043312 006200 ASR R0 ;;GET MESSAGE LNTH IN WORDS
9617 043314 010037 001342 MOV R0,SMSG LGT ;;PUT LENGTH IN MAILBOX
9618 043320 012737 000004 001324 MOV #4,SMSGTYPE ;;TELL APT TO TAKE MSG.
9619 043326 000413 BR 55
9620 043330 017637 000004 043354 35: MOV #4(SP),45 ;;PUT MSG ADDR IN JSR LINKAGE
9621 043336 062766 000002 000004 ADD #2,4(SP) ;;BUMP RETURN ADDRESS
9622 043344 013746 177776 MOV 177776,-(SP) ;;PUSH 177776 ON STACK
9623 043350 004737 042716 JSR PC,STYPE ;;CALL TYPE MACRO
9624 043354 000000 45: .WORD 0
9625 043356 55:
9626 043356 105737 043444 105: TSTB SFFLG ;;SHOULD REPORT FATAL ERROR?
9627 043362 001416 BEQ 125 ;;IF NOT: BR
9628 043364 005737 001344 TST SENV ;;RUNNING UNDER APT?
9629 043370 001413 BEQ 125 ;;IF NOT: BR
9630 043372 005737 001324 115: TST SMSGTYPE ;;FINISHED LAST MESSAGE?
9631 043376 001375 BNE 115 ;;IF NOT: WAIT
9632 043400 017637 000004 001326 MOV #4(SP),SFATAL ;;GET ERROR #
9633 043406 062766 000002 000004 ADD #2,4(SP) ;;BUMP RETURN ADDR.
9634 043414 005237 001324 INC SMSGTYPE ;;TELL APT TO TAKE ERROR
9635 043420 105037 043444 125: CLRB SFFLG ;;CLEAR FATAL FLAG
9636 043424 105037 043443 CLRB SLFLG ;;CLEAR LOG FLAG
9637 043430 105037 043442 CLRB SMFLG ;;CLEAR MESSAGE FLAG
9638 043434 012601 MOV (SP)+,R1 ;;POP STACK INTO R1
9639 043436 012600 MOV (SP)+,R0 ;;POP STACK INTO R0
9640 043440 000207 RTS PC ;;RETURN
9641 043442 000 SMFLG: .BYTE 0 ;;MESSG. FLAG
9642 043443 000 SLFLG: .BYTE 0 ;;LOG FLAG
9643 043444 000 SFFLG: .BYTE 0 ;;FATAL FLAG
9644 043446 .EVEN
9645 000200 APTSIZE=200
9646 000001 APTENV=001
9647 000100 APTPOOL=100

```

F02

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 204
APT COMMUNICATIONS ROUTINE

SEQ 0371

9648

000040

APTCSUP=040

H02

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 206
BINARY TO OCTAL (ASCII) AND TYPE

SEQ 0373

9705	043610	005204							
9706	043612	052703	000060	4S:	INC	R4	;;	DON'T SUPPRESS ANYMORE 0'S	
9707	043616	052703	000040		BIS	8'0,R3	;;	MAKE THIS DIGIT ASCII	
9708	043622	110337	043666	5S:	BIS	8' R3	;;	MAKE ASCII IF NOT ALREADY	
9709	043626	104401	043666		MOVB	R3 8S	;;	SAVE FOR TYPING	
9710	043632	105337	043670		TYPE	8S	;;	GO TYPE THIS DIGIT	
9711	043636	003347		7S:	DECB	\$OCNT	;;	COUNT BY 1	
9712	043640	002402			BGT	2S	;;	BR IF MORE TO DO	
9713	043642	005204			BLT	6S	;;	BR IF DONE	
9714	043644	000744			INC	R4	;;	INSURE LAST DIGIT ISN'T A BLANK	
9715	043646	012605			BR	2S	;;	GO DO THE LAST DIGIT	
9716	043650	012604		6S:	MOV	(SP)+,R5	;;	RESTORE R5	
9717	043652	012603			MOV	(SP)+,R4	;;	RESTORE R4	
9718	043654	016666	000002 000004		MOV	(SP)+,R3	;;	RESTORE R3	
9719	043662	012616			MOV	2(SP),4(SP)	;;	SET THE STACK FOR RETURNING	
9720	043664	000002			MOV	(SP)+,(SP)	;;	RETURN	
9721	043666	000			RTI		;;	STORAGE FOR ASCII DIGIT	
9722	043667	000		8S:	.BYTE	0	;;	TERMINATOR FOR TYPE ROUTINE	
9723	043670	000			.BYTE	00	;;	OCTAL DIGIT COUNTER	
9724	043671	000		\$OCNT:	.BYTE	00	;;	ZERO FILL SWITCH	
9725	043672	000000		\$OFILL:	.BYTE	00	;;	NUMBER OF DIGITS TO TYPE	
				\$OMODE:	.WORD	0	;;		


```

9726
9727
9728
9729
9730
9731
9732
9733
9734 043674 010046
9735 043676 016600 000002
9736 043702 005740
9737 043704 111000
9738 043706 006300
9739 043710 016000 043730
9740 043714 000200
9741
9742
9743
9744
9745 043716 011646
9746 043720 016666 000004 000002
9747 043726 000002
9748
9749
9750
9751
9752
9753
9754
9755
9756 043730 043716
9757 043732 042716
9758 043734 043472
9759 043736 043446
9760 043740 043506
9761
9762

```

.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.

```

```

STRAP:  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
        ASL    RO                ;;POSITION FOR INDEXING
        MOV    STRPAD(RO), RO    ;;INDEX TO TABLE
        RTS    RO                ;;GO TO ROUTINE

```

;;THIS IS USE TO HANDLE THE "GETPRI" MACRO

```

STRAP2: MOV    (SP), -(SP)       ;;MOVE THE PC DOWN
        MOV    4(SP), 2(SP)     ;;MOVE THE PSW DOWN
        RTI                      ;;RESTORE THE PSW

```

.SBTTL TRAP TABLE

```

;THIS TABLE CONTAINS THE STARTING ADDRESSES OF THE ROUTINES CALLED
;BY THE "TRAP" INSTRUCTION.

```

ROUTINE

```

STRPAD: .WORD  STRAP2           TRAP+1(104401)  TTY TYPEOUT ROUTINE
        $TYPE  ;;CALL=TYPE
        $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

```

9763
9764
9765
9766
9767 043742 012737 044114 000024
9768 043750 012737 000340 000026
9769 043756 010046
9770 043760 010146
9771 043762 010246
9772 043764 010346
9773 043766 010446
9774 043770 010546
9775 043772 017746 135146
9776 043776 010637 044120
9777 044002 012737 044014 000024
9778 044010 000000
9779 044012 000776
9780
9781
9782
9783 044014 012737 044114 000024
9784 044022 013706 044120
9785 044026 005037 044120
9786 044032 005237 044120
9787 044036 001375
9788 044040 011600
9789 044042 076600 000226
9790 044046 012677 135072
9791 044052 012605
9792 044054 012604
9793 044056 012603
9794 044060 012602
9795 044062 012601
9796 044064 012600
9797 044066 012737 043742 000024
9798 044074 012737 000340 000026
9799 044102 104401
9800 044104 044122
9801 044106 012716
9802 044110 002400
9803 044112 000002
9804 044114 000000
9805 044116 000776
9806 044120 000000
9807 044122 005015 047520 042527
9808 044130 000122
9809

:*****
:POWER DOWN ROUTINE
SPWRDN: MOV $SILLUP,2#PWRVEC ;;SET FOR FAST UP
MOV $340,2#PWRVEC+2 ;;PRIO:7
MOV RO,-(SP) ;;PUSH RO ON STACK
MOV R1,-(SP) ;;PUSH R1 ON STACK
MOV R2,-(SP) ;;PUSH R2 ON STACK
MOV R3,-(SP) ;;PUSH R3 ON STACK
MOV R4,-(SP) ;;PUSH R4 ON STACK
MOV R5,-(SP) ;;PUSH R5 ON STACK
MOV 2SMR,-(SP) ;;PUSH 2SMR ON STACK
MOV SP,$SAVR6 ;;SAVE SP
MOV $SPWRUP,2#PWRVEC ;;SET UP VECTOR
HALT
BR .-2 ;;HANG UP

:*****
:POWER UP ROUTINE
SPWRUP: MOV $SILLUP,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 SMR OFF STACK
MED 226 ;;RESTORE SMR CONTENTS
MOV (SP)+,2SMR ;;POP STACK INTO 2SMR
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 $SPWRDN,2#PWRVEC ;;SET UP THE POWER DOWN VECTOR
MOV $340,2#PWRVEC+2 ;;PRIO:7
TYPE $POWER ;;REPORT THE POWER FAILURE
SPWRMG: .WORD $POWER ;;POWER FAIL MESSAGE POINTER
MOV (PC)+,(SP) ;;RESTART AT START
SPWRAD: .WORD START ;;RESTART ADDRESS
RTI
SILLUP: HALT
BR .-2
$SAVR6: 0
$POWER: .ASCIZ <15><12>"POWER"
.EVEN

```


.SBTTL ERROR MESSAGES, DATA HEADERS, DATA VECTORS, ETC

: MESSAGE PREFIXES

9810				
9811				
9812				
9813	044132	047510	035124	000040
9814	044140	040527	046522	020072
9815	044146	000		
9816				
9817				
9818	044147	122	041505	044505
9819	044154	042524	020104	050106
9820	044162	020123	051511	041040
9821	044170	042101	000	
9822	044173	122	041505	044505
9823	044200	042524	020104	042506
9824	044206	027503	042506	020101
9825	044214	051511	041040	042101
9826	044222	000		
9827	044223	122	042516	050130
9828	044230	041505	042524	020104
9829	044236	046106	040517	044524
9830	044244	043516	050040	044517
9831	044252	052116	052040	040522
9832	044260	026120	044440	047107
9833	044266	051117	042105	023040
9834	044274	041440	047117	044524
9835	044302	052516	047111	000107
9836	044310	050103	020125	051520
9837	044316	041440	047117	044504
9838	044324	044524	047117	041440
9839	044332	042117	051505	051440
9840	044340	052105	044440	041516
9841	044346	051117	042522	052103
9842	044354	054514	000	
9843	044357	103	050115	043050
9844	044364	042057	020051	050117
9845	044372	051105	052101	047511
9846	044400	020116	020055	042522
9847	044406	044507	052123	051105
9848	044414	046440	042117	043111
9849	044422	042511	020104	043101
9850	044430	042524	020122	054105
9851	044436	041505	052125	047511
9852	044444	000116		
9853	044446	042101	027504	052523
9854	044454	024102	027506	024504
9855	044462	047440	042520	040522
9856	044470	044524	047117	026440
9857	044476	051040	051505	046125
9858	044504	020124	047111	047503
9859	044512	051122	041505	000124
9860	044520	052515	027514	044504
9861	044526	024126	027506	024504
9862	044534	047440	042520	040522
9863	044542	044524	047117	026440
9864	044550	051040	051505	046125
9865	044556	020124	047111	047503

EMA: : ERROR MESSAGES HERE
.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"

9866	044564	051122	041505	000124	
9867	044572	047516	024104	027506	EMH: .ASCIZ "MOD(F/D) OPERATION - FRACTIONAL RESULT INCORRECT"
9868	044600	024504	047440	042520	
9869	044606	040522	044524	047117	
9870	044614	026440	043040	040522	
9871	044622	052103	047511	040516	
9872	044630	020114	042522	052523	
9873	044636	052114	044440	041516	
9874	044644	051117	042522	052103	
9875	044652	000			
9876	044653	115	042117	043050	EMI: .ASCIZ "MOD(F/D) OPERATION - INTEGER RESULT INCORRECT"
9877	044660	042057	020051	050117	
9878	044666	051105	052101	047511	
9879	044674	020116	020055	047111	
9880	044702	042524	042507	020122	
9881	044710	042522	052523	052114	
9882	044716	044440	041516	051117	
9883	044724	042522	052103	000	
9884	044731	106	047514	052101	EMJ: .ASCIZ "FLOAT-TO-DOUBLE CONVERSION - RESULT INCORRECT"
9885	044736	052055	026517	047504	
9886	044744	041125	042514	041440	
9887	044752	047117	042526	051522	
9888	044760	047511	020116	020055	
9889	044766	042522	052523	052114	
9890	044774	044440	041516	051117	
9891	045002	042522	052103	000	
9892	045007	104	052517	046102	EMK: .ASCIZ "DOUBLE-TO-FLOAT CONVERSION - RESULT INCORRECT"
9893	045014	026505	047524	043055	
9894	045022	047514	052101	041440	
9895	045030	047117	042526	051522	
9896	045036	047511	020116	020055	
9897	045044	042522	052523	052114	
9898	045052	044440	041516	051117	
9899	045060	042522	052103	000	
9900	045065	106	054111	042105	EML: .ASCIZ "FIXED-TO-FLOATING CONVERSION - RESULT INCORRECT"
9901	045072	052055	026517	046106	
9902	045100	040517	044524	043516	
9903	045106	041440	047117	042526	
9904	045114	051522	047511	020116	
9905	045122	020055	042522	052523	
9906	045130	052114	044440	041516	
9907	045136	051117	042522	052103	
9908	045144	000			
9909	045145	106	047514	052101	EMM: .ASCIZ "FLOATING-TO-FIXED CONVERSION - RESULT INCORRECT"
9910	045152	047111	026507	047524	
9911	045160	043055	054111	042105	
9912	045166	041440	047117	042526	
9913	045174	051522	047511	020116	
9914	045202	020055	042522	052523	
9915	045210	052114	044440	041516	
9916	045216	051117	042522	052103	
9917	045224	000			
9918	045225	114	040517	020104	EMN: .ASCIZ "LOAD EXPONENT(F/D) OPERATION - RESULT INCORRECT"
9919	045232	054105	047520	042516	
9920	045240	052116	043050	042057	
9921	045246	020051	050117	051105	

9922	045254	052101	047511	020116
9923	045262	020055	042522	052523
9924	045270	052114	044440	041516
9925	045276	051117	042522	052103
9926	045304	000		
9927	045305	123	047524	042522
9928	045312	042440	050130	047117
9929	045320	047105	024124	027506
9930	045326	024504	047440	042520
9931	045334	040522	044524	047117
9932	045342	026440	051040	051505
9933	045350	046125	020124	047111
9934	045356	047503	051122	041505
9935	045364	000124		

EMO: .ASCIZ "STORE EXPONENT(F/D) OPERATION - RESULT INCORRECT"

9936
9937 045366 054105 023520 004504
9938 045374 041522 023526 000104
9939 045402 054105 023520 026504
9940 045410 042506 026503 041522
9941 045416 023526 004504 054105
9942 045424 023520 026504 042506
9943 045432 026501 041522 023526
9944 045440 000104
9945 045448 026455 042455 050130
9946 045456 041506 042524 026504
9947 045464 026455 026411 026455
9948 045472 042503 042503 053111
9949 045472 042106 026455 000055
9950 045500 026455 026455 026455
9951 045506 026455 026455 042455
9952 045514 050130 041506 042524
9953 045522 026455 026455 026455
9954 045530 026455 026455 026455
9955 045536 026411 026455 026455
9956 045544 026455 026455 026455
9957 045552 042522 042503 053111
9958 045560 042106 026455 026455
9959 045566 026455 026455 026455
9960 045574 000055
9961 045576 046117 020104 041520
9962 045604 047411 042114 050040
9963 045612 004523 042516 020127
9964 045620 050123 020011 050106
9965 045626 004523 043040 041505
9966 045634 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"

FPU ADVANCED INSTR TESTS
DQFPBB.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 214
ERROR MESSAGES, DATA HEADERS, DATA VECTORS, ETC

SEQ 0381

10023	046260	001170	001172	001174		.WORD	SREG0,SREG1,SREG2,SREG3,0
10024	046266	001176	000000				
10025	046272	001260	001262	001264	DTAC:	.WORD	STMP14,STMP15,STMP16,STMP17
10026	046300	001266					
10027	046302	001200	001202	001204		.WORD	SREG4,SREG5,SREG6,SREG7,0
10028	046310	001206	000000				
10029	046314	001174	001172	000000	DTAD:	.WORD	SREG2,SREG1,0
10030	046322	001176	001174	000000	DTAE:	.WORD	SREG3,SREG2,0
10031	046330	002006	002010	002012	DTAK:	.WORD	FPPOPC,FPPOPS,FPPOSP,FPS,FEC,FEA,0
10032	046336	002000	002002	002004			
10033	046344	000000					
10034							
10035							
10036							
10037	000001						

; THE END
.END

TST102	007400	1885	1900#
TST103	007436	1904	1919#
TST104	007474	1923	1938#
TST105	007532	1942	1957#
TST106	007570	1961	1976#
TST107	007626	1980	1995#
TST11	003516	735	749#
TST110	007664	1999	2014#
TST111	007722	2018	2033#
TST112	007760	2037	2052#
TST113	010016	2056	2071#
TST114	010054	2075	2090#
TST115	010112	2094	2110#
TST116	010164	2114	2132#
TST117	010236	2136	2154#
TST12	003550	753	767#
TST120	010310	2158	2176#
TST121	010362	2180	2198#
TST122	010434	2202	2220#
TST123	010506	2224	2242#
TST124	010560	2246	2264#
TST125	010632	2268	2286#
TST126	010704	2290	2308#
TST127	010756	2312	2330#
TST13	003602	771	785#
TST130	011030	2334	2352#
TST131	011102	2356	2374#
TST132	011154	2378	2396#
TST133	011226	2400	2418#
TST134	011264	2422	2437#
TST135	011322	2441	2456#
TST136	011360	2460	2475#
TST137	011416	2479	2494#
TST14	003634	789	803#
TST140	011454	2498	2513#
TST141	011512	2517	2532#
TST142	011550	2536	2551#
TST143	011606	2555	2570#
TST144	011644	2574	2589#
TST145	011702	2593	2608#
TST146	011740	2612	2627#
TST147	011776	2631	2646#
TST15	003666	807	822#
TST150	012034	2650	2665#
TST151	012072	2669	2684#
TST152	012130	2688	2703#
TST153	012166	2707	2723#
TST154	012240	2727	2745#
TST155	012312	2749	2767#
TST156	012364	2771	2789#
TST157	012436	2793	2811#
TST16	003730	826	842#
TST160	012510	2815	2833#
TST161	012562	2837	2855#
TST162	012634	2859	2877#
TST163	012706	2881	2899#

TST164	012760	2903	2921
TST165	013032	2925	2943
TST166	013104	2947	2965
TST167	013156	2969	2987
TST17	003772	846	862
TST170	013230	2991	3009
TST171	013302	3013	3031
TST172	013354	3035	3053
TST173	013426	3057	3075
TST174	013464	3079	3094
TST175	013522	3098	3113
TST176	013560	3117	3132
TST177	013616	3136	3151
TST2	003230	609	623
TST20	004034	866	882
TST200	013654	3155	3170
TST201	013712	3174	3189
TST202	013750	3193	3208
TST203	014006	3212	3227
TST204	014044	3231	3246
TST205	014102	3250	3265
TST206	014140	3269	3284
TST207	014176	3288	3303
TST21	004076	886	902
TST210	014234	3307	3322
TST211	014272	3326	3341
TST212	014330	3345	3360
TST213	014366	3364	3379
TST214	014424	3383	3399
TST215	014476	3403	3421
TST216	014550	3425	3443
TST217	014622	3447	3465
TST22	004140	906	922
TST220	014674	3469	3487
TST221	014746	3491	3509
TST222	015020	3513	3531
TST223	015072	3535	3553
TST224	015144	3557	3575
TST225	015216	3579	3597
TST226	015270	3601	3619
TST227	015342	3623	3641
TST23	004202	926	942
TST230	015414	3645	3663
TST231	015466	3667	3685
TST232	015540	3689	3707
TST233	015612	3711	3729
TST234	015664	3733	3751
TST235	015726	3755	3771
TST236	015770	3775	3791
TST237	016032	3795	3811
TST24	004244	946	962
TST240	016074	3815	3831
TST241	016136	3835	3851
TST242	016200	3855	3871
TST243	016242	3875	3891
TST244	016304	3895	3911

H04

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 233
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0399

TST245	016346	3915	39318
TST246	016410	3935	39518
TST247	016452	3955	39718
TST25	004306	966	9828
TST250	016514	3975	39918
TST251	016556	3995	40118
TST252	016620	4015	40318
TST253	016662	4035	40518
TST254	016724	4055	40728
TST255	017006	4076	40968
TST256	017070	4100	41208
TST257	017152	4124	41448
TST26	004350	986	10028
TST260	017234	4148	41688
TST261	017316	4172	41928
TST262	017400	4196	42168
TST263	017462	4220	42408
TST264	017544	4244	42648
TST265	017626	4268	42888
TST266	017710	4292	43128
TST267	017772	4316	43368
TST27	004412	1006	10228
TST270	020054	4340	43608
TST271	020136	4364	43848
TST272	020220	4388	44088
TST273	020302	4412	44328
TST274	020364	4436	44568
TST275	020426	4460	44768
TST276	020470	4480	44968
TST277	020532	4500	45168
TST3	003262	627	6418
TST30	004454	1026	10428
TST300	020574	4520	45368
TST301	020636	4540	45568
TST302	020700	4560	45768
TST303	020742	4580	45968
TST304	021004	4600	46168
TST305	021046	4620	46368
TST306	021110	4640	46568
TST307	021152	4660	46768
TST31	004516	1046	10628
TST310	021214	4680	46968
TST311	021256	4700	47168
TST312	021320	4720	47368
TST313	021362	4740	47568
TST314	021424	4760	47778
TST315	021506	4781	48018
TST316	021570	4805	48258
TST317	021652	4829	48498
TST32	004554	1066	10818
TST320	021734	4853	48738
TST321	022016	4877	48978
TST322	022100	4901	49218
TST323	022162	4925	49458
TST324	022244	4949	49698
TST325	022326	4973	49938

TST326	022410	4997	5017#
TST327	022472	5021	5041#
TST33	004612	1085	1100#
TST330	022554	5045	5065#
TST331	022536	5069	5089#
TST332	022720	5093	5113#
TST333	023002	5117	5137#
TST334	023064	5141	5161#
TST335	023122	5165	5180#
TST336	023160	5184	5199#
TST337	023216	5203	5218#
TST34	004650	1104	1119#
TST340	023254	5222	5237#
TST341	023312	5241	5256#
TST342	023350	5260	5275#
TST343	023406	5279	5294#
TST344	023444	5298	5313#
TST345	023502	5317	5332#
TST346	023540	5336	5351#
TST347	023576	5355	5370#
TST35	004706	1123	1138#
TST350	023634	5374	5389#
TST351	023672	5393	5408#
TST352	023730	5412	5428#
TST353	023766	5432	5447#
TST354	024024	5451	5466#
TST355	024062	5470	5485#
TST356	024120	5489	5504#
TST357	024156	5508	5523#
TST36	004744	1142	1157#
TST360	024214	5527	5542#
TST361	024252	5546	5561#
TST362	024310	5565	5581#
TST363	024346	5585	5600#
TST364	024404	5604	5619#
TST365	024442	5623	5638#
TST366	024500	5642	5657#
TST367	024536	5661	5676#
TST37	005002	1161	1176#
TST370	024574	5680	5695#
TST371	024632	5699	5714#
TST372	024670	5718	5733#
TST373	024726	5737	5752#
TST374	024764	5756	5771#
TST375	025022	5775	5790#
TST376	025060	5794	5810#
TST377	025120	5814	5829#
TST4	003314	645	659#
TST40	005040	1180	1195#
TST400	025160	5833	5848#
TST401	025220	5852	5867#
TST402	025260	5871	5886#
TST403	025320	5890	5905#
TST404	025360	5909	5924#
TST405	025406	5928	5941#
TST406	025434	5945	5958#

TST407	025462	5962	5975
TST41	005076	1199	1214
TST410	025510	5979	5992
TST411	025536	5996	6010
TST412	025570	6014	6028
TST413	025622	6032	6046
TST414	025654	6050	6064
TST415	025706	6068	6082
TST416	025740	6086	6101
TST417	025770	6105	6118
TST42	005134	1218	1233
TST420	026020	6122	6135
TST421	026050	6139	6152
TST422	026100	6156	6169
TST423	026130	6173	6186
TST424	026160	6190	6203
TST425	026210	6207	6221
TST426	026244	6225	6239
TST427	026300	6243	6257
TST43	005172	1237	1252
TST430	026334	6261	6275
TST431	026370	6279	6293
TST432	026424	6297	6312
TST433	026454	6316	6330
TST434	026504	6334	6348
TST435	026534	6352	6366
TST436	026564	6370	6384
TST437	026614	6388	6402
TST44	005230	1256	1271
TST440	026644	6406	6420
TST441	026674	6424	6439
TST442	026730	6443	6458
TST443	026764	6462	6477
TST444	027020	6481	6496
TST445	027054	6500	6515
TST446	027110	6519	6534
TST447	027144	6538	6553
TST45	005266	1275	1290
TST450	027200	6557	6573
TST451	027232	6577	6591
TST452	027264	6595	6609
TST453	027316	6613	6627
TST454	027350	6631	6645
TST455	027402	6649	6663
TST456	027434	6667	6681
TST457	027466	6685	6700
TST46	005324	1294	1309
TST460	027524	6704	6719
TST461	027562	6723	6738
TST462	027620	6742	6757
TST463	027656	6761	6776
TST464	027714	6780	6795
TST465	027752	6799	6814
TST466	030010	6818	6834
TST467	030044	6838	6853
TST47	005362	1313	1328

K04

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 236
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0402

TST470	030100	6857	6872
TST471	030134	6876	6891
TST472	030170	6895	6910
TST473	030224	6914	6929
TST474	030260	6933	6948
TST475	030314	6952	6967
TST476	030350	6971	6986
TST477	030404	6990	7005
TST5	003346	663	677
TST50	005420	1332	1347
TST500	030440	7009	7024
TST501	030474	7028	7043
TST502	030530	7047	7062
TST503	030564	7066	7081
TST504	030620	7085	7100
TST505	030654	7104	7119
TST506	030710	7123	7138
TST507	030744	7142	7157
TST51	005456	1351	1366
TST510	031000	7161	7176
TST511	031034	7180	7195
TST512	031070	7199	7215
TST513	031134	7219	7236
TST514	031200	7240	7257
TST515	031244	7261	7278
TST516	031310	7282	7299
TST517	031354	7303	7320
TST52	005514	1370	1385
TST520	031420	7324	7341
TST521	031464	7345	7362
TST522	031530	7366	7383
TST523	031574	7387	7404
TST524	031640	7408	7425
TST525	031704	7429	7446
TST526	031750	7450	7467
TST527	032014	7471	7488
TST53	005552	1389	1404
TST530	032060	7492	7509
TST531	032124	7513	7530
TST532	032170	7534	7551
TST533	032234	7555	7572
TST534	032300	7576	7593
TST535	032344	7597	7614
TST536	032410	7618	7636
TST537	032436	7640	7653
TST54	005610	1408	1424
TST540	032464	7657	7670
TST541	032512	7674	7687
TST542	032540	7691	7704
TST543	032566	7708	7721
TST544	032614	7725	7738
TST545	032642	7742	7756
TST546	032674	7760	7774
TST547	032726	7778	7792
TST55	005662	1428	1446
TST550	032760	7796	7810

1617#	1619	1639#	1641	1661#	1663	1683#	1685	1705#	1707	1727#	1729	1749#
1751	1771#	1773	1793#	1795	1815#	1817	1838#	1840	1857#	1859	1876#	1878
1895#	1897	1914#	1916	1933#	1935	1952#	1954	1971#	1973	1990#	1992	2009#
2011	2028#	2030	2047#	2049	2066#	2068	2085#	2087	2105#	2107	2127#	2129
2149#	2151	2171#	2173	2193#	2195	2215#	2217	2237#	2239	2259#	2261	2281#
2283	2303#	2305	2325#	2327	2347#	2349	2369#	2371	2391#	2393	2413#	2415
2432#	2434	2451#	2453	2470#	2472	2489#	2491	2508#	2510	2527#	2529	2546#
2548	2565#	2567	2584#	2586	2603#	2605	2622#	2624	2641#	2643	2660#	2662
2679#	2681	2698#	2700	2718#	2720	2740#	2742	2762#	2764	2784#	2786	2806#
2808	2828#	2830	2850#	2852	2872#	2874	2894#	2896	2916#	2918	2938#	2940
2960#	2962	2982#	2984	3004#	3006	3026#	3028	3048#	3050	3070#	3072	3089#
3091	3108#	3110	3127#	3129	3146#	3148	3165#	3167	3184#	3186	3203#	3205
3222#	3224	3241#	3243	3260#	3262	3279#	3281	3298#	3300	3317#	3319	3336#
3338	3355#	3357	3374#	3376	3394#	3396	3416#	3418	3438#	3440	3460#	3462
3482#	3484	3504#	3506	3526#	3528	3548#	3550	3570#	3572	3592#	3594	3614#
3616	3636#	3638	3658#	3660	3680#	3682	3702#	3704	3724#	3726	3746#	3748
3766#	3768	3786#	3788	3806#	3808	3826#	3828	3846#	3848	3866#	3868	3886#
3888	3906#	3908	3926#	3928	3946#	3948	3966#	3968	3986#	3988	4006#	4008
4026#	4028	4046#	4048	4067#	4069	4091#	4093	4115#	4117	4139#	4141	4163#
4165	4187#	4189	4211#	4213	4235#	4237	4259#	4261	4283#	4285	4307#	4309
4331#	4333	4355#	4357	4379#	4381	4403#	4405	4427#	4429	4451#	4453	4471#
4473	4491#	4493	4511#	4513	4531#	4533	4551#	4553	4571#	4573	4591#	4593
4611#	4613	4631#	4633	4651#	4653	4671#	4673	4691#	4693	4711#	4713	4731#
4733	4751#	4753	4772#	4774	4796#	4798	4820#	4822	4844#	4846	4868#	4870
4892#	4894	4916#	4918	4940#	4942	4964#	4966	4988#	4990	5012#	5014	5036#
5038	5060#	5062	5084#	5086	5108#	5110	5132#	5134	5156#	5158	5175#	5177
5194#	5196	5213#	5215	5232#	5234	5251#	5253	5270#	5272	5289#	5291	5308#
5310	5327#	5329	5346#	5348	5365#	5367	5384#	5386	5403#	5405	5423#	5425
5442#	5444	5461#	5463	5480#	5482	5499#	5501	5518#	5520	5537#	5539	5556#
5558	5576#	5578	5595#	5597	5614#	5616	5633#	5635	5652#	5654	5671#	5673
5690#	5692	5709#	5711	5728#	5730	5747#	5749	5766#	5768	5785#	5787	5805#
5807	5824#	5826	5843#	5845	5862#	5864	5881#	5883	5900#	5902	5919#	5921
5936#	5938	5953#	5955	5970#	5972	5987#	5989	6005#	6007	6023#	6025	6041#
6043	6059#	6061	6077#	6079	6096#	6098	6113#	6115	6130#	6132	6147#	6149
6164#	6166	6181#	6183	6198#	6200	6216#	6218	6234#	6236	6252#	6254	6270#
6272	6288#	6290	6307#	6309	6325#	6327	6343#	6345	6361#	6363	6379#	6381
6397#	6399	6415#	6417	6434#	6436	6453#	6455	6472#	6474	6491#	6493	6510#
6512	6529#	6531	6548#	6550	6568#	6570	6586#	6588	6604#	6606	6622#	6624
6640#	6642	6658#	6660	6676#	6678	6695#	6697	6714#	6716	6733#	6735	6752#
6754	6771#	6773	6790#	6792	6809#	6811	6829#	6831	6848#	6850	6867#	6869
6886#	6888	6905#	6907	6924#	6926	6943#	6945	6962#	6964	6981#	6983	7000#
7002	7019#	7021	7038#	7040	7057#	7059	7076#	7078	7095#	7097	7114#	7116
7133#	7135	7152#	7154	7171#	7173	7190#	7192	7210#	7212	7231#	7233	7252#
7254	7273#	7275	7294#	7296	7315#	7317	7336#	7338	7357#	7359	7378#	7380
7399#	7401	7420#	7422	7441#	7443	7462#	7464	7483#	7485	7504#	7506	7525#
7527	7546#	7548	7567#	7569	7588#	7590	7609#	7611	7631#	7633	7648#	7650
7665#	7667	7682#	7684	7699#	7701	7716#	7718	7733#	7735	7751#	7753	7769#
7771	7787#	7789	7805#	7807	7823#	7825	7841#	7843	7859#	7861		
9681#	9710#	9723#		9688#	9699#	9725#						
9676#	9680#	9685	9688#	9699#	9725#							
9337	9353	9361	9371	9380#	9367	9384						
335#	512#	564	7932#	7933#								
234#												
9800	9807#											
9802#												
488	9767#	9797										

SOCNT 043670
 SOMODE 043672
 SOVER 042222
 SPASS 001332
 SPASTH 001006
 SPOWER 044122
 SPWRAD 044110
 SPWRON 043742

		3054	3076	3095	3114	3133	3152	3171	3190	3209	3228	3247	3266	3285
		3304	3323	3342	3361	3380	3400	3422	3444	3466	3488	3510	3532	3554
		3576	3598	3620	3642	3664	3686	3708	3730	3752	3772	3792	3812	3832
		3852	3872	3892	3912	3932	3952	3972	3992	4012	4032	4052	4073	4097
		4121	4145	4169	4193	4217	4241	4265	4289	4313	4337	4361	4385	4409
		4433	4457	4477	4497	4517	4537	4557	4577	4597	4617	4637	4657	4677
		4697	4717	4737	4757	4778	4802	4826	4850	4874	4898	4922	4946	4970
		4994	5018	5042	5066	5090	5114	5138	5162	5181	5200	5219	5238	5257
		5276	5295	5314	5333	5352	5371	5390	5409	5429	5448	5467	5486	5505
		5524	5543	5562	5582	5601	5620	5639	5658	5677	5696	5715	5734	5753
		5772	5791	5811	5830	5849	5868	5887	5906	5925	5942	5959	5976	5993
		6011	6029	6047	6065	6083	6102	6119	6136	6153	6170	6187	6204	6222
		6240	6258	6276	6294	6313	6331	6349	6367	6385	6403	6421	6440	6459
		6478	6497	6516	6535	6554	6574	6592	6610	6628	6646	6664	6682	6701
		6720	6739	6758	6777	6796	6815	6835	6854	6873	6892	6911	6930	6949
		6968	6987	7006	7025	7044	7063	7082	7101	7120	7139	7158	7177	7196
		7216	7237	7258	7279	7300	7321	7342	7363	7384	7405	7426	7447	7468
		7489	7510	7531	7552	7573	7594	7615	7637	7654	7671	7688	7705	7722
		7739	7757	7775	7793	7811	7829	7847	7865	9326	9327	9328	9329	9330
		9336	9348	9350	9351	9354	9355	9356	9363	9364	9365	9377	9380	9383
		9390	9391	9392	9393	9394	9411	9418	9430	9433	9445	9803		
SSWREG	001346	343#	515											
SSWRNK=	000000	9330	9352											
STESTN	001330	334#	9375*	9506										
STIMES	001310	320#	491*	7883*	7931*	9363*	9370	9373*	9383					
STKB	001154	271#												
STKS	001152	270#												
STNPO	001230	296#	7961	8005	8051	8094	8140	8183	8229	8272	8318	8361	8408	8464
		8525	8581	8642	8686	8732	8775	8810	8843	8880	8913	8947	8997	9047
		9099	9148	9191	9237	9277	9995	10005						
STMP1	001232	297#	9995	9997	10005	10009								
STMP10	001250	304#	9983	10021										
STMP11	001252	305#	9973	10021										
STMP12	001254	306#	9974	9985	10021									
STMP13	001256	307#	9987	10021										
STMP14	001260	308#	10025											
STMP15	001262	309#	9975	10025										
STMP16	001264	310#	9989	10025										
STMP17	001266	311#	10025											
STMP2	001234	298#	9993	9997	9999	10005	10009	10013						
STMP20	001270	312#												
STMP21	001272	313#	9976											
STMP22	001274	314#	9991											
STMP23	001276	315#												
STMP24	001300	316#												
STMP25	001302	317#												
STMP26	001304	318#												
STMP27	001306	319#												
STMP3	001236	299#	9999	10005	10009	10013								
STMP4	001240	300#	9969	9994	10001	10009	10013	10017						
STMP5	001242	301#	9970	9977	10001	10013	10017							
STMP6	001244	302#	9971	9979	10003	10017								
STMP7	001246	303#	9972	9981	10003	10017								
STN =	000554	1#	12	600	606#	609	618	624#	627	636	642#	645	654	660#
		663	672	678#	681	690	696#	699	708	714#	717	726	732#	735
		744	750#	753	762	768#	771	780	786#	789	798	804#	807	817

823	826	837	843	846	857	863	866	877	883	886	897	903
906	917	923	926	937	943	946	957	963	966	977	983	986
997	1003	1006	1017	1023	1026	1037	1043	1046	1057	1063	1066	1076
1082	1085	1095	1101	1104	1114	1120	1123	1133	1139	1142	1152	1158
1161	1171	1177	1180	1190	1196	1199	1209	1215	1218	1228	1234	1237
1247	1253	1256	1266	1272	1275	1285	1291	1294	1304	1310	1313	1323
1329	1332	1342	1348	1351	1361	1367	1370	1380	1386	1389	1399	1405
1408	1419	1425	1428	1441	1447	1450	1463	1469	1472	1485	1491	1494
1507	1513	1516	1529	1535	1538	1551	1557	1560	1573	1579	1582	1595
1601	1604	1617	1623	1626	1639	1645	1648	1661	1667	1670	1683	1689
1692	1705	1711	1714	1727	1733	1736	1749	1755	1758	1771	1777	1780
1793	1799	1802	1815	1821	1824	1838	1844	1847	1857	1863	1866	1876
1882	1885	1895	1901	1904	1914	1920	1923	1933	1939	1942	1952	1958
1961	1971	1977	1980	1990	1996	1999	2009	2015	2018	2028	2034	2037
2047	2053	2056	2066	2072	2075	2085	2091	2094	2105	2111	2114	2127
2133	2136	2149	2155	2158	2171	2177	2180	2193	2199	2202	2215	2221
2224	2237	2243	2246	2259	2265	2268	2281	2287	2290	2303	2309	2312
2325	2331	2334	2347	2353	2356	2369	2375	2378	2391	2397	2400	2413
2419	2422	2432	2438	2441	2451	2457	2460	2470	2476	2479	2489	2495
2498	2508	2514	2517	2527	2533	2536	2546	2552	2555	2565	2571	2574
2584	2590	2593	2603	2609	2612	2622	2628	2631	2641	2647	2650	2660
2666	2669	2679	2685	2688	2698	2704	2707	2718	2724	2727	2740	2746
2749	2762	2768	2771	2784	2790	2793	2806	2812	2815	2828	2834	2837
2850	2856	2859	2872	2878	2881	2894	2900	2903	2916	2922	2925	2938
2944	2947	2960	2966	2969	2982	2988	2991	3004	3010	3013	3026	3032
3035	3048	3054	3057	3070	3076	3079	3089	3095	3098	3108	3114	3117
3127	3133	3136	3146	3152	3155	3165	3171	3174	3184	3190	3193	3203
3209	3212	3222	3228	3231	3241	3247	3250	3260	3266	3269	3279	3285
3288	3298	3304	3307	3317	3323	3326	3336	3342	3345	3355	3361	3364
3374	3380	3383	3394	3400	3403	3416	3422	3425	3438	3444	3447	3460
3466	3469	3482	3488	3491	3504	3510	3513	3526	3532	3535	3548	3554
3557	3570	3576	3579	3592	3598	3601	3614	3620	3623	3636	3642	3645
3658	3664	3667	3680	3686	3689	3702	3708	3711	3724	3730	3733	3746
3752	3755	3766	3772	3775	3786	3792	3795	3806	3812	3815	3826	3832
3835	3846	3852	3855	3866	3872	3875	3886	3892	3895	3906	3912	3915
3926	3932	3935	3946	3952	3955	3966	3972	3975	3986	3992	3995	4006
4012	4015	4026	4032	4035	4046	4052	4055	4067	4073	4076	4091	4097
4100	4115	4121	4124	4139	4145	4148	4163	4169	4172	4187	4193	4196
4211	4217	4220	4235	4241	4244	4259	4265	4268	4283	4289	4292	4307
4313	4316	4331	4337	4340	4355	4361	4364	4379	4385	4388	4403	4409
4412	4427	4433	4436	4451	4457	4460	4471	4477	4480	4491	4497	4500
4511	4517	4520	4531	4537	4540	4551	4557	4560	4571	4577	4580	4591
4597	4600	4611	4617	4620	4631	4637	4640	4651	4657	4660	4671	4677
4680	4691	4697	4700	4711	4717	4720	4731	4737	4740	4751	4757	4760
4772	4778	4781	4796	4802	4805	4820	4826	4829	4844	4850	4853	4868
4874	4877	4892	4898	4901	4916	4922	4925	4940	4946	4949	4964	4970
4973	4988	4994	4997	5012	5018	5021	5036	5042	5045	5060	5066	5069
5084	5090	5093	5108	5114	5117	5132	5138	5141	5156	5162	5165	5175
5181	5184	5194	5200	5203	5213	5219	5222	5232	5238	5241	5251	5257
5260	5270	5276	5279	5289	5295	5298	5308	5314	5317	5327	5333	5336
5346	5352	5355	5365	5371	5374	5384	5390	5393	5403	5409	5412	5423
5429	5432	5442	5448	5451	5461	5467	5470	5480	5486	5489	5499	5505
5508	5518	5524	5527	5537	5543	5546	5556	5562	5565	5576	5582	5585
5595	5601	5604	5614	5620	5623	5633	5639	5642	5652	5658	5661	5671
5677	5680	5690	5696	5699	5709	5715	5718	5728	5734	5737	5747	5753
5756	5766	5772	5775	5785	5791	5794	5805	5811	5814	5824	5830	5833

5843	5849#	5852	5862	5868#	5871	5881	5887#	5890	5900	5906#	5909	5919
5925#	5928	5936	5942#	5945	5953	5959#	5962	5970	5976#	5979	5987	5993#
5996	6005	6011#	6014	6023	6029#	6032	6041	6047#	6050	6059	6065#	6068
6077	6083#	6086	6096	6102#	6105	6113	6119#	6122	6130	6136#	6139	6147
6153#	6156	6164	6170#	6173	6181	6187#	6190	6198	6204#	6207	6216	6222#
6225	6234	6240#	6243	6252	6258#	6261	6270	6276#	6279	6288	6294#	6297
6307	6313#	6316	6325	6331#	6334	6343	6349#	6352	6361	6367#	6370	6379
6385#	6388	6397	6403#	6406	6415	6421#	6424	6434	6440#	6443	6453	6459#
6462	6472	6478#	6481	6491	6497#	6500	6510	6516#	6519	6529	6535#	6538
6548	6554#	6557	6568	6574#	6577	6586	6592#	6595	6604	6610#	6613	6622
6628#	6631	6640	6646#	6649	6658	6664#	6667	6676	6682#	6685	6695	6701#
6704	6714	6720#	6723	6733	6739#	6742	6752	6758#	6761	6771	6777#	6780
6790	6796#	6799	6809	6815#	6818	6829	6835#	6838	6848	6854#	6857	6867
6873#	6876	6886	6892#	6895	6905	6911#	6914	6924	6930#	6933	6943	6949#
6952	6962	6968#	6971	6981	6987#	6990	7000	7006#	7009	7019	7025#	7028
7038	7044#	7047	7057	7063#	7066	7076	7082#	7085	7095	7101#	7104	7114
7120#	7123	7133	7139#	7142	7152	7158#	7161	7171	7177#	7180	7190	7196#
7199	7210	7216#	7219	7231	7237#	7240	7252	7258#	7261	7273	7279#	7282
7294	7300#	7303	7315	7321#	7324	7336	7342#	7345	7357	7363#	7366	7378
7384#	7387	7399	7405#	7408	7420	7426#	7429	7441	7447#	7450	7462	7468#
7471	7483	7489#	7492	7504	7510#	7513	7525	7531#	7534	7546	7552#	7555
7567	7573#	7576	7588	7594#	7597	7609	7615#	7618	7631	7637#	7640	7648
7654#	7657	7665	7671#	7674	7682	7688#	7691	7699	7705#	7708	7716	7722#
7725	7733	7739#	7742	7751	7757#	7760	7769	7775#	7778	7787	7793#	7796
7805	7811#	7814	7823	7829#	7832	7841	7847#	7850	7859	7865#	7868	7878
273#	9581#	9592										
277#	9530	9592										
272#	9579	9592										
486	9734#											
9745#	9756											
9749#	9758#	9759#	9760#	9761#								
9739	9756#											
233#												
248#	7885#	7930#	9352	9374#	9375	9380	9384	9410	9445			
9761												
9761												
9530#	9623	9749	9757									
9560	9567	9574	9579#	9580								
9420	9459#											
9585	9587	9590#										
9679#	9758											
9678	9681#	9760										
9674#	9759											
337#												
235#												
344#												
9339#												
9675#	9679#	9689	9724#									
9336	9420											
194#	198#	208	209#	211#	213#	214#	220	221#	223#	225#	244#	326
468#	479	494	495	7963	8007	8053	8096	8142	8185	8231	8274	8320
8363	8410	8466	8527	8583	8644	8688	8734	8777	8812	8845	8882	8915
8949	8999	9049	9101	9150	9193	9239	9279	9383	9384	9445	9512#	9592
9644#	9779	9805										
9596	9599											
220#	225											

STPB 001160
 STPFLG 001165
 STPS 001156
 STRAP 043674
 STRAP2 043716
 STRP = 000005
 STRPAD 043730
 STSTM 001004
 STSTM 001102
 STYPB= ##### U
 STYPS= ##### U
 STYPE 042716
 STYPEC 043130
 STYPER 042502
 STYPEX 043176
 STYPOC 043472
 STYPON 043506
 STYPOS 043446
 SUNIT 001336
 SUNITM 001010
 SUSWR 001350
 EXTSTR 041772
 SOFILL 043671
 S4OCAT= ##### U
 = 046346
 .SASTA= ##### U
 .SX = 001000

COMP4	18														
COMP40	18														
COMP41	18	6399													
COMP42	18														
COMP43	18	6678													
COMP44	18	6531													
COMP45	18														
COMP46	18	6811													
COMP47	18														
ENDCOM	1418														
ERRCMP	18	8978	8983	9028	9033	9078	9130	9259	9264	9299	9304				
ERRLLR	18	9444													
ERROR	358	7978	7986	7993	8022	8030	8041	8068	8076	8083	8111	8119	8130	8157	8165
	8172	8200	8208	8219	8246	8254	8261	8289	8297	8308	8335	8343	8350	8378	8386
	8397	8428	8436	8444	8452	8484	8492	8504	8516	8545	8553	8561	8569	8601	8609
	8621	8633	8659	8667	8674	8703	8711	8722	8748	8756	8763	8790	8801	8824	8831
	8857	8868	8894	8901	8927	8938	8964	8972	8980	8985	9014	9022	9030	9035	9064
	9072	9080	9087	9116	9124	9132	9139	9164	9172	9180	9207	9215	9227	9253	9261
	9266	9293	9301	9306	9317										
ESCAPE	1418														
FCOM0	18	602	620	638	656	674	692	710	728	746	764	800	819	839	859
	879	899	919	939	959	979	999	1039	1059	1078	1097	1116	1135	1154	1173
	1192	1211	1230	1249	1268	1287	1306	1325	1344	1421	1443	1465	1487	1509	1531
	1553	1575	1597	1619	1641	1663	1685	1707	1729	1751	1840	1859	1878	1897	1916
	1935	1954	1973	2011	2030	2068	2107	2129	2151	2173	2195	2217	2239	2261	2305
	2349	2371	2415	2434	2453	2472	2491	2510	2529	2548	2567	2586	2605	2624	2643
	2720	2742	2764	2786	2808	2830	2852	2874	2896	2918	2940	2984	3028	3072	3091
	3110	3129	3148	3167	3186	3205	3224	3243	3262	3281	3319	3357	3396	3418	3440
	3462	3484	3506	3528	3550	3572	3594	3616	3660	3704	3748	3768	3808	3828	3848
	3868	3888	3908	3928	3948	3968	3988	4028	4069	4093	4141	4165	4189	4213	4237
	4261	4285	4309	4333	4357	4405	4453	4473	4513	4533	4553	4573	4593	4613	4633
	4653	4673	4693	4733	4774	4798	4846	4870	4894	4918	4942	4966	4990	5014	5038
	5062	5110	5158	5177	5196	5215	5234	5253	5272	5291	5310	5329	5348	5367	5425
	5444	5463	5482	5501	5520	5558	5578	5597	5616	5635	5654	5673	5692	5711	5730
	5749	5768	5807	5826	5845	5864	5883	5902	5921	5938	5955	5972	5989	6007	6025
	6043	6061	6079	6098	6115	6132	6149	6166	6183	6200	6218	6236	6254	6272	6290
	6309	6327	6345	6363	6381	6417	6436	6455	6474	6493	6512	6550	6570	6588	6606
	6624	6642	6660	6697	6716	6735	6754	6773	6792	6831	6850	6869	6888	6907	6926
	6945	6964	6983	7002	7021	7040	7059	7078	7212	7233	7254	7275	7296	7317	7338
	7359	7380	7401	7422	7443	7464	7485	7633	7650	7667	7684	7701	7718	7735	7753
	7771	7789	7807	7825	7843	7861									
FCOM1	18	782	1019	1382	1795	1992	2283	2700	3050	3300	3638	3788	4117	4493	4822
	5405	5539													
FCOM2	18	1401	1773	2087	2327	2681	3006	3376	3726	4048	4429	4753	5134	7135	7154
	7173	7192	7548	7569	7590	7611									
FCOM3	18	1363	1817	2049	2393	2662	2962	3338	3682	4008	4381	4713	5086	5386	5787
	7097	7116	7506	7527											
FCOM4	18	6399	6531	6678	6811										
FPRGTO	18														
FPRGT1	18														
FPSFEC	18	7976	8020	8066	8109	8155	8198	8244	8287	8333	8376	8426	8482	8543	8599
	8657	8701	8746	8962	9012	9062	9114	9162	9205						
FPSTST	18	8788	8822	8855	8892	8925	9251	9291							
GENCOM	18														
GENTS1	18	600	618	636	654	672	690	708	726	744	762	780	798	817	837
	857	877	897	917	937	957	977	997	1017	1037	1057	1076	1095	1114	1133

H05

FPU ADVANCED INSTR TESTS
DQFP88.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 247
CROSS REFERENCE TABLE -- MACRO NAMES

SEQ 0412

	1152	1171	1190	1209	1228	1247	1266	1285	1304	1323	1342	1361	1380	1399	1419
	1441	1463	1485	1507	1529	1551	1573	1595	1617	1639	1661	1683	1705	1727	1749
	1771	1793	1815	1838	1857	1876	1895	1914	1933	1952	1971	1990	2009	2028	2047
	2066	2085	2105	2127	2149	2171	2193	2215	2237	2259	2281	2303	2325	2347	2369
	2391	2413	2432	2451	2470	2489	2508	2527	2546	2565	2584	2603	2622	2641	2660
	2679	2698	2718	2740	2762	2784	2806	2828	2850	2872	2894	2916	2938	2960	2982
	3004	3026	3048	3070	3089	3108	3127	3146	3165	3184	3203	3222	3241	3260	3279
	3298	3317	3336	3355	3374	3394	3416	3438	3460	3482	3504	3526	3548	3570	3592
	3614	3636	3658	3680	3702	3724	3746	3766	3786	3806	3826	3846	3866	3886	3906
	3926	3946	3966	3986	4006	4026	4046	4067	4091	4115	4139	4163	4187	4211	4235
	4259	4283	4307	4331	4355	4379	4403	4427	4451	4471	4491	4511	4531	4551	4571
	4591	4611	4631	4651	4671	4691	4711	4731	4751	4772	4796	4820	4844	4868	4892
	4916	4940	4964	4988	5012	5036	5060	5084	5108	5132	5156	5175	5194	5213	5232
	5251	5270	5289	5308	5327	5346	5365	5384	5403	5423	5442	5461	5480	5499	5518
	5537	5556	5576	5595	5614	5633	5652	5671	5690	5709	5728	5747	5766	5785	5805
GENTS2	18	5919	5936	5953	5970	5987	6005	6023	6041	6059	6077	6096	6113	6130	6147
	6164	6181	6198	6216	6234	6252	6270	6288	6307	6325	6343	6361	6379	6397	6415
	6434	6453	6472	6491	6510	6529	6548	6568	6586	6604	6622	6640	6658	6676	6695
GENTS3	18	6829	6848	6867	6886	6905	6924	6943	6962	6981	7000	7019	7038	7057	7076
	7095	7114	7133	7152	7171	7190	7210	7231	7252	7273	7294	7315	7336	7357	7378
	7399	7420	7441	7462	7483	7504	7525	7546	7567	7588	7609	7631	7648	7665	7682
	7699	7716	7733	7751	7769	7787	7805	7823	7841	7859					
GENTS4	18														
GETPRI	1418														
GETSMR	1418														
GTSTD	18	8089	8178	8267	8356										
GTSTF	18	8046	8135	8224	8313										
HTSTD	18														
HTSTF	18														
NOVDIS	18	7958	8002	8048	8091	8137	8180	8226	8269	8315	8358	8405	8461	8522	8578
	8639	8683	8729	8772	8807	8840	8877	8910	8944	8994	9044	9096	9145	9188	9234
	9274														
NOVDI1	79598	80038	80498	80928	81388	81818	82278	82708	83168	83598	84068	84628	85238	85798	86408
	86848	87308	87738	88088	88418	88788	89118	89458	89958	90458	90978	91468	91898	92358	92758
NOVDI2	79598	80038	80498	80928	81388	81818	82278	82708	83168	83598	84068	84628	85238	85798	86408
	86848	87308	87738	88088	88418	88788	89118	89458	89958	90458	90978	91468	91898	92358	92758
MULT	1418														
NEWTST	1418	600	618	636	654	672	690	708	726	744	762	780	798	817	837
	857	877	897	917	937	957	977	997	1017	1037	1057	1076	1095	1114	1133
	1152	1171	1190	1209	1228	1247	1266	1285	1304	1323	1342	1361	1380	1399	1419
	1441	1463	1485	1507	1529	1551	1573	1595	1617	1639	1661	1683	1705	1727	1749
	1771	1793	1815	1838	1857	1876	1895	1914	1933	1952	1971	1990	2009	2028	2047
	2066	2085	2105	2127	2149	2171	2193	2215	2237	2259	2281	2303	2325	2347	2369
	2391	2413	2432	2451	2470	2489	2508	2527	2546	2565	2584	2603	2622	2641	2660
	2679	2698	2718	2740	2762	2784	2806	2828	2850	2872	2894	2916	2938	2960	2982
	3004	3026	3048	3070	3089	3108	3127	3146	3165	3184	3203	3222	3241	3260	3279
	3298	3317	3336	3355	3374	3394	3416	3438	3460	3482	3504	3526	3548	3570	3592
	3614	3636	3658	3680	3702	3724	3746	3766	3786	3806	3826	3846	3866	3886	3906
	3926	3946	3966	3986	4006	4026	4046	4067	4091	4115	4139	4163	4187	4211	4235
	4259	4283	4307	4331	4355	4379	4403	4427	4451	4471	4491	4511	4531	4551	4571
	4591	4611	4631	4651	4671	4691	4711	4731	4751	4772	4796	4820	4844	4868	4892
	4916	4940	4964	4988	5012	5036	5060	5084	5108	5132	5156	5175	5194	5213	5232
	5251	5270	5289	5308	5327	5346	5365	5384	5403	5423	5442	5461	5480	5499	5518
	5537	5556	5576	5595	5614	5633	5652	5671	5690	5709	5728	5747	5766	5785	5805

	5824	5843	5862	5881	5900	5919	5936	5953	5970	5987	6005	6023	6041	6059	6077
	6096	6113	6130	6147	6164	6181	6198	6216	6234	6252	6270	6288	6307	6325	6343
	6361	6379	6397	6415	6434	6453	6472	6491	6510	6529	6548	6568	6586	6604	6622
	6640	6658	6676	6695	6714	6733	6752	6771	6790	6809	6829	6848	6867	6886	6905
	6924	6943	6962	6981	7000	7019	7038	7057	7076	7095	7114	7133	7152	7171	7190
	7210	7231	7252	7273	7294	7315	7336	7357	7378	7399	7420	7441	7462	7483	7504
	7525	7546	7567	7588	7609	7631	7648	7665	7682	7699	7716	7733	7751	7769	7787
	7805	7823	7841	7859											
POP	1418	9638	9639	9790	9791										
PUSH	1418	9599	9601	9622	9769	9775									
REPORT	1418														
SBTST1	18	600	618	636	654	672	690	708	726	744	762	780	798	817	837
	857	877	897	917	937	957	977	997	1017	1037	1057	1076	1095	1114	1133
	1152	1171	1190	1209	1228	1247	1266	1285	1304	1323	1342	1361	1380	1399	1419
	1441	1463	1485	1507	1529	1551	1573	1595	1617	1639	1661	1683	1705	1727	1749
	1771	1793	1815	1838	1857	1876	1895	1914	1933	1952	1971	1990	2009	2028	2047
	2066	2085	2105	2127	2149	2171	2193	2215	2237	2259	2281	2303	2325	2347	2369
	2391	2413	2432	2451	2470	2489	2508	2527	2546	2565	2584	2603	2622	2641	2660
	2679	2698	2718	2740	2762	2784	2806	2828	2850	2872	2894	2916	2938	2960	2982
	3004	3026	3048	3070	3089	3108	3127	3146	3165	3184	3203	3222	3241	3260	3279
	3298	3317	3336	3355	3374	3394	3416	3438	3460	3482	3504	3526	3548	3570	3592
	3614	3636	3658	3680	3702	3724	3746	3766	3786	3806	3826	3846	3866	3886	3906
	3926	3946	3966	3986	4006	4026	4046	4067	4091	4115	4139	4163	4187	4211	4235
	4259	4283	4307	4331	4355	4379	4403	4427	4451	4471	4491	4511	4531	4551	4571
	4591	4611	4631	4651	4671	4691	4711	4731	4751	4772	4796	4820	4844	4868	4892
	4916	4940	4964	4988	5012	5036	5060	5084	5108	5132	5156	5175	5194	5213	5232
	5251	5270	5289	5308	5327	5346	5365	5384	5403	5423	5442	5461	5480	5499	5518
	5537	5556	5576	5595	5614	5633	5652	5671	5690	5709	5728	5747	5766	5785	5805
	5824	5843	5862	5881	5900	5919	5936	5953	5970	5987	6005	6023	6041	6059	6077
	6096	6113	6130	6147	6164	6181	6198	6216	6234	6252	6270	6288	6307	6325	6343
	6361	6379	6397	6415	6434	6453	6472	6491	6510	6529	6548	6568	6586	6604	6622
	6640	6658	6676	6695	6714	6733	6752	6771	6790	6809	6829	6848	6867	6886	6905
	6924	6943	6962	6981	7000	7019	7038	7057	7076	7095	7114	7133	7152	7171	7190
	7210	7231	7252	7273	7294	7315	7336	7357	7378	7399	7420	7441	7462	7483	7504
	7525	7546	7567	7588	7609	7631	7648	7665	7682	7699	7716	7733	7751	7769	7787
	7805	7823	7841	7859											
SBTST2	18														
SCOM0	18	729	801	1079	1136	1231	1288	1402	1841	1917	2416	2473	2511	2530	2663
	3149	3301	3339	3789	3889	3989	4009	4494	4594	4694	4714	5159	5235	5311	5808
SCOM1	18	5846	5922	5956	5990	6364	6382	6418	6889	6965	7702	7702	7702	7702	7702
	2454	603	657	675	747	1060	1155	1212	1307	1860	1936	1993	2012	2069	2088
	5254	2549	2625	2644	3073	3111	3187	3225	3263	3377	3809	3869	4514	4574	5197
	7193	5330	5406	5884	5939	5973	6310	6328	6346	6400	6870	6946	7022	7060	7155
SCOM2	18	7634	7736												
	2568	621	639	693	765	1117	1174	1269	1326	1364	1879	1955	2031	2050	2435
	4674	2606	2701	3092	3130	3206	3244	3320	3749	3849	3929	3969	4454	4554	4634
	6851	5178	5273	5349	5387	5827	5903	6099	6133	6150	6167	6571	6589	6643	6661
SCOM3	18	6927	7003	7041	7117	7136	7174	7651	7719						
	3769	711	783	1098	1193	1250	1345	1898	1974	2492	2587	2682	3168	3282	3358
	5865	3829	3909	3949	4029	4049	4474	4534	4614	4654	4734	4754	5216	5292	5368
SCOM4	18	6116	6184	6201	6607	6625	6679	6832	6908	6984	7079	7098	7668	7668	7668
	2875	980	1466	1510	1598	1664	1708	1818	2196	2240	2262	2372	2394	2743	2787
	5111	3397	3485	3529	3661	3683	4166	4214	4262	4334	4406	4871	4919	4967	5039
	7423	5445	5521	5636	5693	5712	5788	6008	6062	6456	6475	6494	6532	7255	7339
SCOM5	18	7570	7790	7862											
	880	900	940	960	1040	1488	1532	1730	2152	2853	2941	2985	3029	3441	

SC0M6	3507 5769 18	3573 6026 820	3617 6044 840	3639 6080 860	4070 6437 920	4118 6513 1020	4310 6551 1383	4358 7213 1422	4775 7297 1554	4823 7381 1642	5015 7465 1686	5063 7507 1752	5426 7612 1774	5502 7772 2130	5617 7844 2284
SC0M7	2306 4847 7234 18	2328 5135 7318	2721 5483 7402	2831 5559 7486	2919 5598 7528	3007 5655 7549	3419 5750 7754	3551 6219 7808	3595 6273 7808	2218 5087 2350	2765 5540 2809	2809 5579 2897	2897 5674 2963	2963 5731 3051	3051 6237 6812
SCOPE	3463 6255 368	4190 6291 605	4238 6774 623	4286 7276 641	4382 7360 659	4895 7444 677	4943 7591 695	4991 7826 713	5087 7826 731	5464 7826 749	5540 7826 767	5579 7826 785	5674 7826 803	5731 7826 822	6237 7826 842
SCPLUR	862 1157 1446 1776 2071 2396 2684 3009 3303 3619 3931 4264 4596 4921 5256 5542 5829 6101 6366 6645 6929 7215 7530 7810	882 1176 1468 1798 2090 2418 2703 3031 3322 3641 3951 4288 4616 4945 5275 5561 5848 6118 6384 6663 6948 7236 7551 7828	902 1195 1490 1820 2110 2437 2723 3053 3341 3663 3971 4312 4636 4969 5294 5581 5867 6135 6402 6681 6967 7257 7572 7846	922 1214 1512 1843 2132 2456 2745 3075 3360 3685 3991 4336 4656 4993 5313 5600 5886 6152 6420 6700 6986 7278 7593 7864	942 1233 1534 1862 2154 2475 2767 3094 3379 3707 4011 4360 4676 5017 5332 5619 5905 6169 6439 6719 7005 7299 7614 7882	962 1252 1556 1881 2176 2494 2789 3113 3399 3729 4031 4384 4696 5041 5351 5638 5924 6186 6458 6738 7024 7320 7636 7902	982 1271 1578 1900 2198 2513 2811 3132 3421 3751 4051 4408 4716 5065 5370 5657 5941 6203 6477 6757 7043 7341 7653 7919	1002 1290 1578 1900 2220 2532 2833 3151 3443 3771 4072 4432 4736 5089 5389 5676 5958 6221 6496 6776 7062 7362 7670 7936	1022 1309 1622 1938 2242 2551 2855 3170 3465 3791 4096 4456 4756 5113 5408 5695 5975 6239 6496 6776 7062 7362 7687 7953	1042 1328 1644 1957 2264 2570 2877 3189 3487 3811 4120 4476 4777 5137 5428 5714 5992 6257 6534 6814 7100 7404 7704 7968	1062 1347 1666 1976 2286 2589 2899 3208 3509 3831 4144 4496 4801 5161 5447 5733 6010 6275 6553 6834 7119 7425 7721 7985	1081 1366 1688 1995 2308 2608 2921 3227 3531 3851 4168 4516 4825 5180 5466 5752 6028 6293 6573 6853 7138 7446 7738 7999	1100 1385 1710 2014 2330 2627 2943 3246 3553 3871 4192 4536 4849 5214 5485 5771 6046 6312 6591 6872 7157 7467 7756 8017	1119 1404 1732 2033 2352 2646 2965 3265 3575 3891 4216 4556 4873 5248 5504 5790 6064 6330 6609 6891 7176 7488 7774 8035	1138 1424 1754 2052 2374 2665 2987 3284 3597 3911 4240 4576 4897 5277 5523 5810 6082 6348 6627 6910 7195 7509 7792 8053
SEADAT	873 1167 1459 1789 2081 2409 2694 3022 3313 3632 3942 4279 4607 4936 5266 5552 5840 6110 6375 6654	893 1186 1481 1811 2100 2428 2713 3044 3332 3654 3962 4303 4627 4960 5285 5571 5859 6127 6393 6672	913 1205 1503 1833 2123 2447 2736 3066 3351 3676 3982 4327 4647 4984 5304 5591 5878 6144 6411 6690	933 1224 1525 1853 2145 2467 2758 3085 3370 3698 4002 4351 4687 5008 5323 5610 5897 6161 6429 6710	953 1243 1547 1872 2167 2485 2780 3104 3389 3720 4022 4375 4687 5032 5342 5629 5916 6178 6449 6729	973 1262 1569 1891 2189 2504 2802 3123 3412 3742 4042 4399 4707 5056 5361 5648 5933 6195 6468 6748	993 1281 1591 1910 2211 2523 2824 3142 3434 3762 4062 4423 4727 5080 5380 5667 5950 6212 6487 6767	1013 1300 1613 1929 2233 2542 2846 3161 3456 3782 4087 4447 4747 5104 5399 5686 5967 6231 6506 6786	1033 1319 1635 1948 2255 2561 2868 3180 3478 3802 4111 4467 4767 5128 5418 5696 5974 6249 6525 6805	1053 1338 1657 1967 2277 2580 2890 3199 3500 3822 4135 4487 4792 5152 5438 5705 5984 6267 6544 6824	1072 1357 1679 1986 2299 2599 2912 3218 3522 3842 4159 4507 4816 5171 5457 5743 6001 6267 6544 6824	1091 1376 1701 1986 2321 2618 2934 3237 3544 3862 4183 4527 4840 5190 5474 5762 6020 6285 6563 6844	1110 1395 1723 2005 2343 2637 2956 3257 3566 3882 4183 4547 4864 5209 5495 5781 6038 6303 6582 6863	1129 1414 1745 2043 2365 2656 2978 3279 3588 3902 4231 4567 4888 5228 5514 5800 6074 6339 6618 6901	1148 1437 1767 2062 2387 2675 3000 3294 3610 3922 4255 4587 4912 5247 5533 5821 6092 6357 6636 6920

K05

FPU ADVANCED INSTR TESTS
DQFPBB.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 250
CROSS REFERENCE TABLE -- MACRO NAMES

SEQ 0415

	6939	6958	6977	6996	7015	7034	7053	7072	7091	7110	7129	7148	7167	7186	7205
	7227	7248	7269	7290	7311	7332	7353	7374	7395	7416	7437	7458	7479	7500	7521
	7542	7563	7584	7605	7626	7645	7662	7679	7696	7713	7730	7747	7766	7784	7802
	7820	7838	7856	7874											
SETPRI	1418														
SETREG	18	9399													
SETTRA	97498	9758	9759	9760											
SETUP	18	1418	473												
SKIP	1418	609	627	645	663	681	699	717	735	753	771	789	807	826	846
	866	886	906	926	946	966	986	1006	1026	1046	1066	1085	1104	1123	1142
	1161	1180	1199	1218	1237	1256	1275	1294	1313	1332	1351	1370	1389	1408	1428
	1450	1472	1494	1516	1538	1560	1582	1604	1626	1648	1670	1692	1714	1736	1758
	1780	1802	1824	1847	1866	1885	1904	1923	1942	1961	1980	1999	2018	2037	2056
	2075	2094	2114	2136	2158	2180	2202	2224	2246	2268	2290	2312	2334	2356	2378
	2400	2422	2441	2460	2479	2498	2517	2536	2555	2574	2593	2612	2631	2650	2669
	2688	2707	2727	2749	2771	2793	2815	2837	2859	2881	2903	2925	2947	2969	2991
	3013	3035	3057	3079	3098	3117	3136	3155	3174	3193	3212	3231	3250	3269	3288
	3307	3326	3345	3364	3383	3403	3425	3447	3469	3491	3513	3535	3557	3579	3601
	3623	3645	3667	3689	3711	3733	3755	3775	3795	3815	3835	3855	3875	3895	3915
	3935	3955	3975	3995	4015	4035	4055	4076	4100	4124	4148	4172	4196	4220	4244
	4268	4292	4316	4340	4364	4388	4412	4436	4460	4480	4500	4520	4540	4560	4580
	4600	4620	4640	4660	4680	4700	4720	4740	4760	4781	4805	4829	4853	4877	4901
	4925	4949	4973	4997	5021	5045	5069	5093	5117	5141	5165	5189	5203	5222	5241
	5260	5279	5298	5317	5336	5355	5374	5393	5412	5432	5451	5470	5489	5508	5527
	5546	5565	5585	5604	5623	5642	5661	5680	5699	5718	5737	5756	5775	5794	5814
	5833	5852	5871	5890	5909	5928	5945	5962	5979	5996	6014	6032	6050	6068	6086
	6105	6122	6139	6156	6173	6190	6207	6225	6243	6261	6279	6297	6316	6334	6352
	6370	6388	6406	6424	6443	6462	6481	6500	6519	6538	6557	6577	6595	6613	6631
	6649	6667	6685	6704	6723	6742	6761	6780	6799	6818	6838	6857	6876	6895	6914
	6933	6952	6971	6990	7009	7028	7047	7066	7085	7104	7123	7142	7161	7180	7199
	7219	7240	7261	7282	7303	7324	7345	7366	7387	7408	7429	7450	7471	7492	7513
	7534	7555	7576	7597	7618	7640	7657	7674	7691	7708	7725	7742	7760	7778	7796
	7814	7832	7850	7868											
SLASH	1418														
SPACE	1418														
STARS	1418														
	604	206	217	219	226	240	326	329	469	471	554	556	571	573	600
	618	618	622	636	640	654	658	672	676	690	694	708	712	726	730
	744	748	762	766	780	784	798	802	817	821	837	841	857	861	877
	881	897	901	917	921	937	941	957	961	977	981	997	1001	1017	1021
	1037	1041	1057	1061	1076	1080	1095	1099	1114	1118	1133	1137	1152	1156	1171
	1175	1190	1194	1209	1213	1228	1232	1247	1251	1266	1270	1285	1289	1304	1308
	1323	1327	1342	1346	1361	1365	1380	1384	1399	1403	1419	1423	1441	1445	1463
	1467	1485	1489	1507	1511	1529	1533	1551	1555	1573	1577	1595	1599	1617	1621
	1639	1643	1661	1665	1683	1687	1705	1709	1727	1731	1749	1753	1771	1775	1793
	1797	1815	1819	1838	1842	1857	1861	1876	1880	1895	1899	1914	1918	1933	1937
	1952	1956	1971	1975	1990	1994	2009	2013	2028	2032	2047	2051	2066	2070	2085
	2089	2105	2109	2127	2131	2149	2153	2171	2175	2193	2197	2215	2219	2237	2241
	2259	2263	2281	2285	2303	2307	2325	2329	2347	2351	2369	2373	2391	2395	2413
	2417	2432	2436	2451	2455	2470	2474	2489	2493	2508	2512	2527	2531	2546	2550
	2565	2569	2584	2588	2603	2607	2622	2626	2641	2645	2660	2664	2679	2683	2698
	2702	2718	2722	2740	2744	2762	2766	2784	2788	2806	2810	2828	2832	2850	2854
	2872	2876	2894	2898	2916	2920	2938	2942	2960	2964	2982	2986	3004	3008	3026
	3030	3048	3052	3070	3074	3089	3093	3108	3112	3127	3131	3146	3150	3165	3169
	3184	3188	3203	3207	3222	3226	3241	3245	3260	3264	3279	3283	3298	3302	3317
	3321	3336	3340	3355	3359	3374	3378	3394	3398	3416	3420	3438	3442	3460	3464
	3482	3486	3504	3508	3526	3530	3548	3552	3570	3574	3592	3596	3614	3618	3636

3640	3658	3662	3680	3684	3702	3706	3724	3728	3746	3750	3766	3770	3786	3790
3806	3810	3826	3830	3846	3850	3866	3870	3886	3890	3906	3910	3926	3930	3946
3950	3966	3970	3986	3990	4006	4010	4026	4030	4046	4050	4067	4071	4091	4095
4115	4119	4139	4143	4163	4167	4187	4191	4211	4215	4235	4239	4259	4263	4283
4287	4307	4311	4331	4335	4355	4359	4379	4383	4403	4407	4427	4431	4451	4455
4471	4475	4491	4495	4511	4515	4531	4535	4551	4555	4571	4575	4591	4595	4611
4615	4631	4635	4651	4655	4671	4675	4691	4695	4711	4715	4731	4735	4751	4755
4772	4776	4796	4800	4820	4824	4844	4848	4868	4872	4892	4896	4916	4920	4940
4944	4964	4968	4988	4992	5012	5016	5036	5040	5060	5064	5084	5088	5108	5112
5132	5136	5156	5160	5175	5179	5194	5198	5213	5217	5232	5236	5251	5255	5270
5274	5289	5293	5308	5312	5327	5331	5346	5350	5365	5369	5384	5388	5403	5407
5423	5427	5442	5446	5461	5465	5480	5494	5499	5503	5518	5522	5537	5541	5556
5560	5576	5580	5595	5599	5614	5618	5633	5637	5652	5656	5671	5675	5690	5694
5709	5713	5728	5732	5747	5751	5766	5770	5785	5789	5805	5809	5824	5828	5843
5847	5862	5866	5881	5885	5900	5904	5919	5923	5936	5940	5953	5957	5970	5974
5987	5991	6005	6009	6023	6027	6041	6045	6059	6063	6077	6081	6096	6100	6113
6117	6130	6134	6147	6151	6164	6168	6181	6185	6198	6202	6216	6220	6234	6238
6252	6256	6270	6274	6288	6292	6307	6311	6325	6329	6343	6347	6361	6365	6379
6383	6397	6401	6415	6419	6434	6438	6453	6457	6472	6476	6491	6495	6510	6514
6529	6533	6548	6552	6568	6572	6586	6590	6604	6608	6622	6626	6640	6644	6658
6662	6676	6680	6695	6699	6714	6718	6733	6737	6752	6756	6771	6775	6790	6794
6809	6813	6829	6833	6848	6852	6867	6871	6886	6890	6905	6909	6924	6928	6943
6947	6962	6966	6981	6985	7000	7004	7019	7023	7038	7042	7057	7061	7076	7080
7095	7099	7114	7118	7133	7137	7152	7156	7171	7175	7190	7194	7210	7214	7231
7235	7252	7256	7273	7277	7294	7298	7315	7319	7336	7340	7357	7361	7378	7382
7399	7403	7420	7424	7441	7445	7462	7466	7483	7487	7504	7508	7525	7529	7546
7550	7567	7571	7588	7592	7609	7613	7631	7635	7648	7652	7665	7669	7682	7686
7699	7703	7716	7720	7733	7737	7751	7755	7769	7773	7787	7791	7805	7809	7823
7827	7841	7845	7859	7863	7878	7918	7999	8088	8177	8266	8355	8458	8575	8680
8769	8837	8874	8907	8991	9041	9093	9185	9271	9323	9386	9445	9515	9594	9651
9728	9765	9781												

STATUS	18														
SMRSU	1418	4968													
TAD001	18														
TAD002	18														
TADDF1	18														
TADDF2	18														
TADDR1	18														
TADDR2	18														
TRNTRP	97498														
TYPBIN	1418														
TYPDEC	1418														
TYPNAM	1418														
TYPNUM	1418														
TYPOCS	1418														
TYPOCT	1418														
TYPTXT	1418														
UPCODE	18	9788													
SSCHRE	2388	280	281	282	283	284	285	286	287	288	289	290	291	292	293
	294	295													
SSCHTH	2388	296	297	298	299	300	301	302	303	304	305	306	307	308	309
	310	311	312	313	314	315	316	317	318	319					
SSESCA	1418														
SSNEWT	1418	600	618	636	654	672	690	708	726	744	762	780	798	817	837
	857	877	897	917	937	957	977	997	1017	1037	1057	1076	1095	1114	1133
	1152	1171	1190	1209	1228	1247	1266	1285	1304	1323	1342	1361	1380	1399	1419

FPU ADVANCED INSTR TESTS
DQFPBB.P11 19-APR-77 13:37

MACY11 27(1006) 25-APR-77 09:19 PAGE 253
CROSS REFERENCE TABLE -- MACRO NAMES

SEQ 0418

.SAPTH	18	215
.SAPTY	18	9592
.SCATC	18	192
.SCHTA	18	238
.SEOP	18	7918
.SERRO	18	9384
.SPOWE	18	9763
.SSCOP	18	9321
.STRAP	18	9726
.STYER	18	9445
.STYPE	18	9513
.STYPO	18	9649

. ABS. 046346 000

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

DSKZ:DQFPBB,DSKZ:DQFPBB/CRF/SOL/P/DOC/CPU:70/EX/EN:WRP=DQFP.MAC,DQFPBB.P11
RUN-TIME: 58 60 5 SECONDS
RUN-TIME RATIO: 735/123=5.9
CORE USED: 26K (52 PAGES)

DOCUMENT PAGES: 258
WRAP-AROUND: 0%

USER SYMBOLS: 1265
MACRO NAMES: 143
UNDF SYMBOLS: 14
DISK BLOCKS READ: 1159
DISK BLKS WRITTEN: 1261
KILO CORE SECONDS: 4923

EOF1DQFPBBSEQ

00010000

770608

PDP10 411

8