

DDDDDDDDDDDD	DDDDDDDDDDDD	FFFFFFFFFFFFFF	FFFFFFFFFFFFFF	PPPP	AAAAAAAAAA	AAAAAAAAAA
DDDDDDDDDDDD	DDDDDDDDDDDD	FFFFFFFFFFFFFF	FFFFFFFFFFFFFF	PPPP	AAAAAAAAAA	AAAAAAAAAA
DDDDDDDDDDDD	DDDDDDDDDDDD	FFFFFFFFFFFFFF	FFFFFFFFFFFFFF	PPPPPPPPPPPP	AAAAAAAAAA	AAAAAAAAAA
DDDD	DDDD	FFF	FF-	PPP	PPP	AAA AAA AAA AAA
DD	DD	FFF	FFF	PPP	PPP	AAA AAA AAA AAA
DDDD	DDDD	FFF	FFF	PPP	PPP	AAA AAA AAA AAA
DDDD	DDDD	FFF	FFF	PPP	PPP	AAA AAA AAA AAA
DDDD	DDDD	FFF	FFF	PPP	PPP	AAA AAA AAA AAA
DDDD	DDDD	FFF	FFF	PPP	PPP	AAA AAA AAA AAA
DDDD	DDDD	FFF	FFF	PPP	PPP	AAA AAA AAA AAA
DDDD	DDDD	FFFFFFFFFFFFFF	FFFFFFFFFFFFFF	PPPPPPPPPPPP	AAA	AAA AAA AAA AAA
DDDD	DDDD	FFFFFFFFFFFFFF	FFFFFFFFFFFFFF	PPPPPPPPPPPP	AAA	AAA AAA AAA AAA
DDDD	DDDD	FFFFFFFFFFFFFF	FFFFFFFFFFFFFF	PPPPPPPPPPPP	AAA	AAA AAA AAA AAA
DDDD	DDDD	FFF	FFF	PPP	PPP	AAAAAAAAAAAAAAAAAAAA
DDDD	DDDD	FFF	FFF	PPP	PPP	AAAAAAAAAAAAAAAAAAAA
DDDD	DDDD	FFF	FFF	PPP	PPP	AAAAAAAAAAAAAAAAAAAA
DDDD	DDDD	FFF	FFF	PPP	PPP	AAA AAA AAA AAA
DDDD	DDDD	FFF	FFF	PPP	PPP	AAA AAA AAA AAA
DDDD	DDDD	FFF	FFF	PPP	PPP	AAA AAA AAA AAA
DDDDDDDDDDDD	DDDDDDDDDDDD	FFF	FFF	PPP	PPP	AAA AAA AAA AAA
DDDDDDDDDDDD	DDDDDDDDDDDD	FFF	FFF	PPP	PPP	AAA AAA AAA AAA
DDDDDDDDDDDD	DDDDDDDDDDDD	FFF	FFF	PPP	PPP	AAA AAA AAA AAA

FPU Diag

SSSSSSSSSSSS	EEEEEEEEEEEEEE	QQQQQQQQQQ	
SSSSSSSSSSSS	EEEEEEEEEEEEEE	QQQQQQQQQQ	
SSSSSSSSSSSS	EEEEEEEEEEEEEE	QQQQQQQQQQ	
SSS	EEE	QQQ	QQQ
SSS	EEE	QQQ	QQQ
SSS	EEE	QQQ	QQQ
SSS	EEE	QQQ	QQQ
SSS	EEE	QQQ	QQQ
SSS	EEE	QQQ	QQQ
SSSSSSSSSS	EEEEEEEEEEEEEE	QQQ	QQQ
SSSSSSSSSS	EEEEEEEEEEEEEE	QQQ	QQQ
SSSSSSSSSS	EEEEEEEEEEEEEE	QQQ	QQQ
SSS	EEE	QQQ	QQQ
SSS	EEE	QQQ	QQQ
SSS	EEE	QQQ	QQQ
SSS	EEE	QQQ	QQQ
SSS	EEE	QQQ	QQQ
SSSSSSSSSSSS	EEEEEEEEEEEEEE	QQQQQQ	QQQ
SSSSSSSSSSSS	EEEEEEEEEEEEEE	QQQQQQ	QQQ
SSSSSSSSSSSS	EEEEEEEEEEEEEE	QQQQQQ	QQQ

LPTSPB Version 6(100344) Running on LPT040  
 \*START\* User WILLIAMS,LOR [400,2003] Job DFFPAA Seq. 361 Date 21-Dec-76 07:18:37 Monitor IPC-D 602 [6B0] \*START\*  
 Request created: 20-Dec-76 20:07:46 /TO:ML21-4;WILLIA -- distribution to ML21-4, slot 153  
 File: DSK20:DFFPAA.SEQ[400,2003] Created: 17-Nov-76 21:02:00 <157> Printed: 21-Dec-76 09:07:36  
 QUEUE Switches: /PRINT:ARROW /FILE:ASCII /COPIES:5 /SPACING:1 /LIMIT:566 /FORMS:NORMAL  
 File will be RENAMED to <057> protection

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

.REM 6

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DFEPA-A-D  
PRODUCT NAME: PDP-11/34 FPP DIAGNOSTIC PART 1  
DATE: DECEMBER 1976  
AUTHOR: ANTHONY VEZZA

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

THE SOFTWARE DESCRIBED IN THIS DOCUMENT 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.

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

COPYRIGHT (C) 1976 BY DIGITAL EQUIPMENT CORPORATION

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  
101  
102  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112

CONTENTS

- 1. ABSTRACT
- 2. REQUIREMENTS
  - 2.1 EQUIPMENT
  - 2.2 STOPAGE
  - 2.3 PRELIMINARY PROGRAMS
- 3. LOADING PROCEDURE
- 4. STARTING PROCEDURE
  - 4.1 CONTROL SWITCH SETTINGS
  - 4.2 STARTING ADDRESS
  - 4.3 PROGRAM AND OPERATOR INTERACTION
- 5. OPERATING PROCEDURE
  - 5.1 OPERATIONAL SWITCH SETTINGS
  - 5.3 OPERATOR ACTION
- 6. ERRORS
  - 6.1 SUMMARY
  - 6.2 ERROR RECOVERY
- 7. RESTRICTIONS
  - 7.1 STARTING RESTRICTIONS
  - 7.2 OPERATING RESTRICTIONS
- 8. MISCELLANEOUS
  - 8.1 EXECUTION TIMES
  - 8.2 STACK POINTER
  - 8.3 PASS COUNT
  - 8.4 T-BIT TRAPPING
  - 8.5 SOFTWARE SWITCH REGISTER
  - 8.6 INTERRUPTS TEST
  - 8.7 ACT, APT AND XXDP COMPATIBILITY
- 9. PROGRAM DESCRIPTION
  - 9.1 DEEPA
- 10. LISTING
  - 10.1 DEEPA

113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161  
162  
163  
164  
165  
166  
167  
168

-----

THE THREE PROGRAMS:

DFPBA DFFPB DFFPC

ARE DESIGN TO DETECT AND REPORT LOGIC FAULTS IN THE PDP 11/34 FPP-A FLOATING POINT PROCESSOR. THE DESIGN IS AN ATTEMPT TO REACH ALL ROM STATES, TAKE ALL BRANCH MICRO TESTS (BUT'S) AND VERIFY ALL THE LOGIC. THEY CONSIST OF 155 (OCT) INDIVIDUAL TESTS SEQUENCED TO DETECT AND ATTEMPT TO IDENTIFY FAULTS WITH A MINIMUM HARDWARE OR SOFTWARE LEVEL. THE TESTS ARE PARTIONED INTO THREE STAND-ALONE PROGRAMS DESCRIBED BELOW.

NOTE THAT ERROR REPORTS IN THESE PROGRAMS ARE BASED UPON THE KNOWLEDGE THAT ALL PREVIOUS TESTS HAVE BEEN RUN AND IN MOST CASE THAT THERE IS ONLY A SINGLE POINT FAULT IN THE FPP-A. IF THE PROGRAMS OR TESTS ARE NOT RUN IN ORDER THEN ERROR MESSAGES MAY NOT BE ACCURATE.

A. DFFPA

DFFPA TESTS:

LDFPS  
STEPS  
CFCC  
SETF, SETD, SETI AND SETL  
STST  
LDF AND LDD (ALL SOURCE MODES)  
STD (MODE 0 AND 1)  
ADDF, ADDD AND SUBD (MOST CONDITIONS)

B. DFFPB

DFFPB TESTS:

ADDF, ADDD AND SUBD (ALL CONDITIONS NOT TESTED IN DFFPA)  
CMPD AND CMRF  
DIVD AND DIVF  
MULD AND MULF  
MODD AND MODF

C. DFFPC

DFFPC TESTS:

STF AND STD (ALL MODES)  
STCFD AND STCDF  
CLR D AND CLRF  
NEG F AND NEG D

169 ABSF AND ABSD  
170 TSTF AND TSTD  
171 NEGF, ARSF AND TSTF (ALL SOURCE MODES)  
172 NEGF, ARSF AND TSTF (ALL SOURCE MODES)  
173 LDFPS (ALL SOURCE MODES)  
174 LDCIF AND LDCIF  
175 LDCID AND LDCID  
176 LDEXP  
177 STEPS (ALL DESTINATION MODES)  
178 STCFI AND STCFI  
179 STCDL AND STCDI  
180 STEXP  
181 STST

2. REQUIREMENTS  
-----

2.1 EQUIPMENT

A PDP 11/34 (WITH OR WITHOUT CONSOLE), LA30 (OR EQUIVALENT) AND AN FP11-A FLOATING POINT PROCESSOR. NOTE THAT A SPECIAL INTERRUPTS TEST MODULE IS BEING DESIGNED FOR USE IN THE MANUFACTURING ENVIRONMENT. WHEN THIS DEVICE IS PRESENT THE PROGRAM DFFPR WILL MAKE USE OF IT TO TEST THE FPP INTERRUPT ON BUS REQUEST FUNCTIONS.

2.2 STORAGE

ALL THREE PROGRAM REQUIRE A MEMORY SYSTEM OF AT LEAST 16K TO LOAD AND RUN.

2.3 PRELIMINARY PROGRAMS

THESE THREE DIAGNOSTICS WILL ASSUME THAT THE PDP 11/34 CENTRAL PROCESSOR IS FAULTLESS, THEREFORE WHEN IN DOUBT RUN THE PDP 11/34 PROCESSOR DIAGNOSTICS BEFORE THESE FP11-A DIAGNOSTICS.

3. LOADING PROCEDURE  
-----

THE PROGRAMS WILL BE SUPPLIED ON THE 11/34 DIAGNOSTIC MEDIA. REFER TO THE XXOP OPERATING MANUAL FOR FURTHER INFORMATION.

4. STARTING PROCEDURE  
-----

4.1 CONTROL SWITCH SETTINGS

SEE SECTION 5.1

4.2 PROGRAM AND OPERATOR ACTION

- 225
- 226 1. LOAD PROGRAM INTO MEMORY
- 227 2. LOAD ADDRESS 200
- 228 3. SET CONSOLE SWITCHES (IF CONSOLE IS PRESENT)
- 229 4. PRESS START
- 230 ON FIRST PASS THE PROGRAM
- 231 WILL IDENTIFY ITSELF. NOTE THAT IF THERE IS
- 232 NO PHYSICAL CONSOLE THE PROGRAM WILL REQUEST
- 233 THE OPERATOR FOR INITIAL VALUE FOR THE
- 234 SOFTWARE SWITCH REGISTER (SEE SECTION 8.5).
- 235 IF RUNNING UNDER ACT, APT OR CHAIN THIS DOES
- 236 NOT APPLY.
- 237 5. THE PROGRAM WILL LOOP AND AN END OF PASS AND
- 238 ERROR SUMMARY WILL BE TYPED AT THE END OF
- 239 EVERY PASS.
- 240

5. OPERATING PROCEDURE

5.1 OPERATIONAL SWITCH SETTINGS

THE SWITCH SETTING ARE:

	OCTAL	
SW<15>=1	100000	HALT ON ERROR
SW<14>=1	40000	LOOP ON CURRENT TEST
SW<13>=1	20000	INHIBIT ERROR TYPE OUTS
SW<12>=1	10000	INHIBIT T-BIT TRAPPING
SW<11>=1	4000	INHIBIT ITERATIONS
SW<10>=1	2000	RING TTY BELL ON ERROR
SW<9>=1	1000	LOOP ON ERROR
SW<8>=1	400	LOOP ON TEST SPECIFIED IN SW<6>
		THROUGH SW<0>
SW<7>=1	200	PRINT ERROR SUMMARY EVEN IF
		SW<13>=1, THIS APPLIES ONLY TO
		PROGRAM DFEPA.
SW<7>=1	200	DESELECT CORRECT INTERRUPT TEST IN
		PROGRAM DFEPA. NOTE THAT THIS TEST
		WILL AUTOMATICALLY BE DESELECTED BY
		THE ABSENCE OF THE SPECIAL TEST
		EQUIPMENT DESIGNED TO CONDUCT THIS
		TEST. IF THIS EQUIPMENT IS NOT
		INSTALLED THERE IS NO NEED TO
		DESELECT THIS TEST. THIS APPLIES
		ONLY TO PROGRAM DFEPA!

6. ERRORS

6.1 SUMMARIES

IN PROGRAM DFEPA TESTS 1 AND 11 HAVE A SPECIAL ERROR SUMMARY FEATURE. THESE TWO TEST RUN MANY TEST PATTERNS THROUGH THE LOGIC. AFTER AN ERROR IS ENCOUNTERED, ONLY THE FIRST FIVE ERRORS ARE REPORTED

281 (TYPED ON THE TTY). EVERY ERROR THOUGH IS LOGGED  
282 AND AN ERROR SUMMARY IS PRINTED WHEN THE TEST IS  
283 COMPLETE. NOTE THAT IF SW<13>=1 THIS SUMMARY WILL  
284 NOT BE TYPED UNLESS SW<7>=1. IN OTHER WORDS TO GET  
285 JUST AN ERROR SUMMARY FROM EITHER OF THESE TWO TESTS  
286 1 AND 11 IN PROGRAM DFEPA BOTH SWITCHES 13 AND 7  
287 MUST = 1.

288  
289 6.2 ERROR RECOVERY

290 SW<15:9>=0... MOST ERRORS WILL CAUSE EXECUTION TO  
291 GO TO THE START OF THE NEXT TEST  
292 AFTER THE MESSAGE IS TYPED. A FEW  
293 TESTS ARE IN SECTIONS. IN THESE  
294 TESTS AN ERROR WILL CAUSE EXECUTION  
295 TO GO TO THE NEXT SECTION AFTER THE  
296 MESSAGE IS TYPED.

297  
298  
299 SW<15>=1... THE PROGRAM WILL HALT AFTER TYPING  
300 THE ERROR MESSAGE. PRESSING THE  
301 CONSOLE CONTINUE WILL CAUSE THE  
302 PROGRAM TO CONTINUE AS IF SW<15>=0.

303  
304 7. RESTRICTIONS  
305 -----

306  
307 NONE

308  
309  
310 8. MISCELLANEOUS  
311 -----

312  
313 8.1 EXECUTION TIMES

314 LESS THAN 10 SECONDS FOR EACH PROGRAM ON ANY PASS.

315  
316  
317 8.2 STACK POINTER

318 THE STACK POINTER IS INITIALIZED TO 1100 IN EACH OF  
319 THE THREE PROGRAMS.

320  
321  
322 8.3 PASS COUNT

323 THE PROGRAM MAKES ONE PASS FOR EACH END OF PASS  
324 MESSAGE TYPED. THE END OF PASS MESSAGE DESCRIBES  
325 THE TOTAL NUMBER OF PASSES COMPLETED AND THE TOTAL  
326 NUMBER OF ERRORS SINCE THE LAST END OF PASS MESSAGE.

327  
328  
329 8.4 T-BIT TRAPPING

330 IF SW<12>=0 EACH PROGRAM WILL RUN WITH TRACE TRAPS  
331 ON EVERY OTHER PASS. FIRST PASS WILL NOT ENABLE  
332 TRACE TRAPS. NOTE SW<12>=1 DISABLES T-BIT TRAPS.

333  
334  
335 8.5 SOFTWARE SWITCH REGISTER  
336

337 EACH OF THE THREE PROGRAMS WILL RUN WITH OR WITHOUT  
338 A CONSOLE SWITCH REGISTER. IF A PHYSICAL CONSOLE  
339 SWITCH REGISTER IS PRESENT ON THE SYSTEM, THEN THESE  
340 PROGRAMS WILL GO AHEAD AND USE IT FOR THE SWITCH  
341 FUNCTIONS DESCRIBED IN 5.1 ABOVE. IF HOWEVER THERE  
342 IS NO CONSOLE SWITCH REGISTER ON THE SYSTEM A  
343 SOFTWARE SWITCH REGISTER WILL BE USED. THIS  
344 SOFTWARE SWITCH REGISTER CAN BE EXAMINED OR MODIFIED  
345 AT ANY TIME BY THE USER IF HE TYPES CONTROL G WHILE  
346 THE PROGRAM IS RUNNING. THIS CONTROL G WILL CAUSE  
347 THE CONTENTS OF THE SOFTWARE SWITCH REGISTER TO BE  
348 TYPED ON THE TTY AND ASK THE USER FOR A NEW VALUE.  
349 WHEN THE USER TYPES A VALUE AND CARRIAGE RETURN THEN  
350 THE PROGRAM WILL RESUME TESTING AT THE SAME POINT AT  
351 WHICH IT LEFT OFF WHEN THE USER TYPED CONTROL G.  
352 NOTE THAT WHEN NOT RUNNING UNDER ACT, APT OR CHAIN  
353 THE USER WILL BE ASKED FOR A SOFTWARE SWITCH  
354 REGISTER VALUE AFTER LOADING ADDRESS 200 AND  
355 STARTING THE PROGRAM THE FIRST TIME THE PROGRAM IS  
356 RUN AFTER LOADING (ONLY IF NO CONSOLE SWITCH  
357 REGISTER IS ON THE SYSTEM).

#### 8.6 INTERRUPTS TEST

361 IN PROGRAM DFEPA THERE IS A SPECIAL TEST FOR  
362 CHECKING THE CORRECT FLOWS OF THE FPP. THIS TEST  
363 CAN BE RUN ONLY IF A SPECIAL TEST MODULE IS IN THE  
364 SYSTEM. THIS MODULE WILL PROBABLY ONLY BE USED IN  
365 MANUFACTURING. IF THIS MODULE IS NOT IN THE SYSTEM  
366 THIS TEST WILL AUTOMATICALLY BE DESELECTED. IF THIS  
367 TEST MODULE IS ON THE SYSTEM AND SW<7>=0 THIS TEST  
368 WILL BE RUN. IF SW<7>=1 THIS TEST WILL BE  
369 DESELECTED.

#### 8.7 ACT, APT AND XDP COMPATIBILITY

373 THESE PROGRAMS ARE FULLY COMPATIBLE WITH:  
374 APT  
375 ACT  
376 XDP MONITOR AND CHAIN PROGRAMS.  
377  
378  
379  
380  
381

#### 9. PROGRAM DESCRIPTION

382  
383  
384  
385  
386  
387  
388  
389  
390  
391  
392



393  
394  
395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448

TEST 1 LDFPS, STEPS AND DATA PATHS TEST  
-----

THIS IS A TEST OF THE LDFPS (LOAD FLOATING POINT STATUS) AND STEPS (STORE FLOATING POINT STATUS) INSTRUCTIONS. A COUNT PATTERN IS GENERATED AND RUN THROUGH THE FLOATING POINT STATUS REGISTER. THIS WILL TEST THE 16-BIT TRI STATE BUS WHICH CONNECTS THE CPU WITH THE FPP AND ALSO RUNS INTERNALLY WITHIN THE FPP. ONLY DM0 AND SM0 ARE USED. NOTE THAT A MASK MUST BE USED BECAUSE SOME OF THE FPS BITS CANNOT BE SET.

ONLY THE FIRST FIVE ERRORS WILL BE REPORTED INDIVIDUALLY. THIS IS TO PREVENT LOCKING OUT THE COMPLETION OF THE TEST BECAUSE OF VIRTUALLY ENDLESS NUMBER OF ERRORS. ONLY FIVE INDIVIDUAL ERRORS WILL BE REPORTED THEN THE TEST WILL BE COMPLETED AND AN ERROR SUMMARY GIVEN (SEE NOTE BELOW).

NOTE THAT THIS TEST KEEPS A DYNAMIC RECORD OF THE LOGICAL 'AND' AND 'OR' OF THE FAILING DATA PATTERNS. THESE CAN BE VERY USEFUL IN DETERMINING STUCK BITS. IF THE USER HAS THE INHIBIT ERROR TYPE OUT SWITCH (SWR13) OFF, THEN THE USER WILL RECIEVE EACH INDIVIDUAL ERROR MESSAGE PLUS AN ERROR SUMMARY AT THE END OF THE TEST. INHIBITING ERROR PRINT OUT WILL INHIBIT ERROR SUMMARY PRINT OUT, EXCEPT IN THE CASE DESCRIBED BELOW. TO GET JUST THE ERROR SUMMARY WITH NO INDIVIDUAL ERROR REPORTS, SET SWITCH REGISTER BIT13 AND SWITCH REGISTER BIT7 BOTH ON.

TEST 2 CECC TEST  
-----

THIS IS A TEST OF THE COPY CONDITION CODES INSTRUCTION, CECC.

TEST 3 SETF, SETD, SETI AND SETL TEST  
-----

THIS IS A TEST OF THE SETF, SETD, SETI AND SETL INSTRUCTIONS. EACH INSTRUCTION IS EXECUTED WITH THE FPS CONTAINING ALL ONES AND ALSO WITH THE FPS CLEAR. THE RESULT OF EACH SITUATION IS CHECKED.

TEST 4 ILLEGAL FPP OP CODES AND STST TEST  
-----

THIS IS A TEST OF THE FPP OPERATION CODES:

449 170004  
450 :  
451 170010  
452 170013  
453 170014  
454 :  
455 170077  
456

457 THESE ARE ILLEGAL INSTRUCTIONS AND WITH INTERRUPTS  
458 ENABLED SHOULD CAUSE A TRAP TO 244. ALSO TESTED  
459 HERE IS THE INSTRUCTION: STST R1, WHICH SHOULD PUT  
460 THE FPC CODE 2 IN R1, AFTER ANY OF THE ABOVE OP  
461 CODES IS EXECUTED.  
462

463 TEST 5 FID, INTERRUPT DISABLE, BIT TEST  
464 -----

465  
466 THIS IS A TEST OF FPC BIT 14 (FID) OR FLOATING  
467 INTERRUPT DISABLE. AN ILLEGAL INSTRUCTION IS  
468 EXECUTED WITH FID=1. NO INTERRUPT SHOULD OCCUR.  
469

470 TEST 6 LDD AND STD, WITH SRC AND DST MODE 1, TEST  
471 -----

472  
473 THIS IS A TEST OF BOTH THE INSTRUCTION:  
474 LDD (R0),AC0  
475 AND THE INSTRUCTION:  
476 STD AC0,(R0) MOST OF THE  
477 FAILURES ARE ISOLATED TO THE SRC OR DST FLOWS. NOTE  
478 THAT THE INTEGRITY OF AC0 HAS NOT BEEN ASSURED.  
479 THIS MEANS THAT IN SOME CASES IT WILL BE IMPOSSIBLE  
480 TO ISOLATE CERTAIN DATA PATTERN FAILURES TO EITHER  
481 THE FLOWS OR THIS ACCUMULATOR.  
482

483 TEST 7 FSRC MODE 0 TEST  
484 -----

485  
486 THIS IS A TEST OF FSRC MODE ZERO USING THE LDD AND  
487 LDF INSTRUCTIONS.  
488

489  
490 TEST 10 FDST MODE 0 TEST  
491 -----

492  
493 THIS IS A TEST OF THE STORE INSTRUCTIONS, STD AND  
494 STF, WITH FDST MODE 0.  
495

496 TEST 11 ACCUMULATORS DATA PATTERNS TEST  
497 -----

498  
499 THIS IS A TEST OF THE FLOATING POINT PROCESSOR  
500 ACCUMULATORS.  
501

502 EACH ACCUMULATOR IS TESTED IN TWO WAYS:  
503 1 TEST PATTERN GENERATED BY FLOATING A  
504 ONE ACROSS A FIELD OF ZEROES.

505 2 TEST PATTERN GENERATED BY FLOATING A  
506 ZERO ACROSS A FIELD OF ONES.  
507 EACH OF ACCUMULATORS AC0 THROUGH AC5 IS TESTED.  
508  
509 NOTE THAT THIS TEST KEEPS A DYNAMIC RECORD OF THE  
510 LOGICAL 'AND' AND 'OR' OF THE FAILING DATA PATTERNS.  
511 THESE CAN BE VERY USEFUL IN DETERMINING STUCK BITS.  
512 IF THE USER HAS THE INHIBIT ERROR TYPE OUT SWITCH  
513 (SWR13) OFF, THEN THE USER WILL RECIEVE EACH  
514 INDIVIDUAL ERROR MESSAGE PLUS AN ERROR SUMMARY AT  
515 THE END OF THE TEST. INHIBITING ERROR PRINT OUT  
516 WILL INHIBIT ERROR SUMMARY PRINT OUT, EXCEPT IN THE  
517 CASE DESCRIBED BELOW. TO GET JUST THE ERROR SUMMARY  
518 WITH NO INDIVIDUAL ERROR REPORTS, SET SWITCH  
519 REGISTER BIT13 AND SWITCH REGISTER BIT7 BOTH ON.  
520

521  
522 THE FOLLOWING PROCEDURE IS PRESENTED TO AID THE  
523 TROUBLE SHOOTER IN SITUATIONS WHERE AM2901 CHIP  
524 ISOLATION IS ATTEMPTED.  
525

526 WARNING: THIS PROCEDURE ASSUMES THAT THE FAULT IS  
527 IN ONE OF THE AM2901 CHIPS. THIS ASSUMPTION IS NOT  
528 NECESSARILY VALID IN ALL SITUATIONS. IT REMAINS TO  
529 BE SEEN WHAT NUMBER OF FAILURES CAN  
530 PROBABILISTICALLY ASSOCIATED WITH THEM. NOTE ALSO  
531 THAT THIS INFORMATION SHOULD NOT BE TAKEN AS  
532 ABSOLUTE, THAT IS THIS INFORMATION IS THE AUTHOR'S  
533 SUGGESTION FOR ACHIEVING ISOLATION WHEN CHIP LEVEL  
534 REPAIR IS NECESSARY.  
535

536 WHEN THIS TEST HAS FINISHED RUNNING, IF ERRORS HAVE  
537 OCCURRED, AN ERROR SUMMARY WILL BE TYPED. THIS  
538 SUMMARY WILL CONSIST OF TWO IMPORTANT QUANTITIES:

- 539 A. FOUR SIXTEEN BIT NUMBERS LABELED THE
- 540 LOGICAL 'AND' ('\*') OF THE FAILING
- 541 DATA PATTERNS.
- 542 B. FOUR SIXTEEN BIT NUMBERS LABELED THE
- 543 LOGICAL 'OR' ('+') OF THE FAILING
- 544 DATA PATTERNS.

545  
546 A BIT STUCK HIGH IN THE HARDWARE WILL SHOW UP AS A 0  
547 IN THAT BIT POSITION OF THE 'OR' OF THE FAILING DATA  
548 PATTERNS.  
549

550  
551 A BIT STUCK LOW IN THE HARDWARE WILL SHOW UP AS A 1  
552 IN THAT BIT POSITION OF THE 'AND' OF THE FAILING  
553 DATA PATTERNS.  
554

555  
556 THUS IF A FAILURE OCCURS:  
557 A. STUCK HIGHS WILL SHOW AS 0'S IN THE  
558 'OR' PATTERN.  
559 B. STUCK LOWS WILL SHOW AS 1'S IN THE  
560 'AND' PATTERN.

IF THE FAILURE IS INTERMITTANT THEN THIS PROCEDURE WILL

561 STILL APPLY!! IF THE FAILURE MOVES FROM ONE BIT TO  
 562 ANOTHER, OR FROM ONE GROUP OF BITS TO ANOTHER GROUP  
 563 OF BITS THEN THE FAULT WILL PROBABLY NOT SHOW UP IN  
 564 THE 'AND' OR THE 'OR' PATTERNS; IN THIS CASE THE  
 565 'AND' PATTERN WILL BE ALL 0'S AND THE 'OR' PATTERN  
 566 WILL BE ALL 1'S. WHEN THIS OCCURS SOME OTHER METHOD  
 567 OF REPAIR MUST BE FOUND (SUCH AS INSPECTION OF EACH  
 568 INDIVIDUAL ERROR REPORT RATHER THAN USING THE  
 569 SUMMARY).

570  
 571 MAP THE FOLLOWING NOTATION ONTO EACH BIT POSITION IN  
 572 THE 'AND' AND THE 'OR' PATTERNS WHICH ARE TYPED IN  
 573 THE ERROR SUMMARY.

574  
 575 A15,A14,...A1,A0 B15,B14,...B1,B0  
 576 C15,C14,...C1,C0 D15,D14,...D1,D0

577  
 578 IN THIS NOTATION A15 THROUGH A0 IS THE FIRST OF THE  
 579 FOUR 16 BIT OCTAL NUMBERS TYPED, B15 THROUGH B0 IS  
 580 THE SECOND, ETC.

581  
 582 THIS TABLE SHOWS THE CORRESPONDING AM2901 CHIP ('E'  
 583 NUMBER) WHICH IS RESPONSIBLE FOR EACH BIT POSITION  
 584 USING THE ABOVE NOTATION. NOTE THAT ECO'S TO THE  
 585 HARDWARE MIGHT MAKE THIS TABLE OBSOLETE IF IT IS NOT  
 586 UP DATED. NOTE ALSO THAT THERE ARE FOUR BITS FOR  
 587 EACH AM2901 CHIP:

588	BITS	AM2901 CHIP NUMBER
589	-----	-----
590		
591		
592	A15,A14,A13,A12	E61
593	A11,A10,A9,A8	E62
594	A7,A6,A5,A4	E90
595	A3,A2,A1,A0	E81
596		
597	B15,B14,B13,B12	E86
598	B11,B10,B9,B8	E85
599	B7,B6,B5,B4	E83
600	B3,B2,B1,B0	E88
601		
602	C15,C14,C13,C12	E79
603	C11,C10,C9,C8	E84
604	C7,C6,C5,C4	E89
605	C3,C2,C1,C0	E87
606		
607	D15,D14,D13,D12	E78
608	D11,D10,D9,D8	E77
609	D7,D6,D5,D4	E82
610	D3,D2,D1,D0	E80

611  
 612 NOW FIVE IMPORTANT CASES WHICH WILL ARISE WHEN A  
 613 FAULTY AM2901 IS PRESENT CAN BE DESCRIBED:

614  
 615  
 616 1.) IF ONLY ONE BIT OF THE 64 BITS IS INCORRECT

MICRO-BUSINESS FORMS, INC. HO  
 TORIA 1413  
 PRINTED IN U.S.A. 74

617  
618  
619  
620  
621  
622  
623  
624  
625  
626  
627  
628  
629  
630  
631  
632  
633  
634  
635  
636  
637  
638  
639  
640  
641  
642  
643  
644  
645  
646  
647  
648  
649  
650  
651  
652  
653  
654  
655  
656  
657  
658  
659  
660  
661  
662  
663  
664  
665  
666  
667  
668  
669  
670  
671  
672

THE CHIP INDICATED IN THE ABOVE TABLE IS MOST PROBABLY AT FAULT. BUT IF THAT CHIP IS REPLACED AND THE ERROR PERSISTS THEN SUPPOSE THAT BIT IS,

LN WHERE 'L' IS A, B, C OR D  
AND N IS 15, 14, ... OR 0

THEN IN GENERAL ANY OF THE FOUR CHIPS RESPONSIBLE FOR AN, BN, CN OR DN COULD BE AT FAULT, WITH LN BEING MOST PROBABLE.

FOR EXAMPLE IF BIT C12 IS FAULTY, THEN CHIP E79 IS THE MOST PROBABLE SOURCE OF THE ERROR. IF REPAIRING THAT CHIP DOES NOT REMOVE THE FAULT THEN TRY EACH OF THE CHIPS ASSOCIATED WITH BITS A12, B12 AND D12 SHOULD BE TRIED WITH EQUAL PROBABILITY OF THE FAULT BEING IN ANY ONE OF THESE OTHER THREE CHIPS, TRY CHIPS E61, E86 AND E78.

2.) IF THERE ARE FOUR CONSECUTIVE BITS IN ERROR, FOLLOWING THE PATTERN:

LN, LN+1, LN+2 AND LN+3 WHERE 'L' IS A, B, C OR D  
N=0, 4, 8 OR 12

THEN THE ABOVE TABLE SHOULD DIRECTLY IDENTIFY THE FAILING CHIP.

3.) IF FOUR BITS ARE DROPPED WHICH FIT THE PATTERN:

AN, BN, CN AND DN WHERE N=15, 14, ... OR 0  
OR 0

THEN ANY ONE OF THE FOUR CHIPS ASSOCIATED WITH EACH OF THE BITS AN, BN, CN AND DN COULD BE AT FAULT WITH EQUAL PROBABILITY.

4.) IF 16 BITS ARE IN ERROR, FITTING THE PATTERN:

AN, AN+1, AN+2, AN+3 WHERE N=0, 4, 8 OR 12  
BN, BN+1, BN+2, BN+3  
CN, CN+1, CN+2, CN+3  
AND  
DN, DN+1, DN+2, DN+3

THEN ANY ONE OF THE FOUR CHIPS ASSOCIATED WITH THESE BITS COULD BE AT FAULT WITH EQUAL PROBABILITY.

5.) IF THE FAILING BIT PATTERNS DISPLAYED IN THE 'AND' AND THE 'OR' DATA TYPED IN THE SUMMARY DOES NOT CONFORM EXPLICITELY TO ANY OF THE

673 ABOVE PATTERNS, THEN THE TROUBLE SHOOTER  
674 MUST INTUITIVELY TRY TO FIND WHICH OF THE  
675 ABOVE CASES (1 THROUGH 4) IS A 'BEST FIT' OF  
676 THE SYMPTOMS.  
677

678 TEST 12 FPP ACCUMULATORS DUAL ADDRESS TEST  
679 -----  
680

681 THIS TEST PERFORMS A DUAL ADDRESSING TEST ON THE  
682 FLOATING ACCUMULATORS. NOTE THAT ACCUMULATOR ZERO  
683 IS USED TO ACCESS ALL THE OTHERS.  
684

685 TEST 13 FSRC MODE 0 WITH ILLEGAL ACCUMULATOR TEST  
686 -----  
687

688 THIS IS A TEST OF FSRC MODE 0 WITH ACCUMULATORS 6  
689 AND 7. USE OF EITHER OF THESE NON-EXISTENT  
690 ACCUMULATORS SHOULD RESULT IN A TRAP TO 244 WITH  
691 FEC=2 (ILLEGAL FPP INSTRUCTION).  
692

693  
694 TEST 14 FSRC MODE 2 TEST  
695 -----  
696

697 THIS IS A TEST OF FSRC MODE 2, AUTO INCREMENT MODE.  
698

699 TEST 15 FSRC MODE 4 TEST  
700 -----  
701

702 THIS IS A TEST OF FSRC MODE 4, AUTO DECREMENT MODE.  
703

704 TEST 16 FSRC MODE 2, WITH FD=0, TEST  
705 -----  
706

707 THIS IS A TEST OF FSRC MODE 2 WITH FD=0. (AUTO  
708 INCREMENT)  
709

710 TEST 17 FSRC MODE 2 WITH GR7, IMMEDIATE MODE, TEST  
711 -----  
712

713 THIS IS A TEST OF FSRC MODE 2 USING GR7 (THE PC).  
714 THIS IS IMMEDIATE MODE.  
715

716 TEST 20 FSRC MODE 3 TEST  
717 -----  
718

719 THIS IS A TEST OF FSRC MODE 3, AUTO INCREMENT  
720 DEFERRED  
721

722 TEST 21 FSRC MODE 5 TEST  
723 -----  
724

725 THIS IS A TEST OF FSRC MODE 5, AUTO DECREMENT  
726 DEFERRED.  
727

728 TEST 22 FSRC MODE 6 TEST

FORM 1413 PRINTED IN U.S.A.

```
729 -----
730
731 THIS IS A TEST OF FSRC MODE 6, INDEX MODE
732
733 TEST 23 FSRC MODE 7 TEST
734 -----
735
736 THIS IS A TEST OF FSRC MODE 7, INDEX DEFERRED MODE.
737
738 TEST 24 (BUT EZBT Y8), (BUT ENBT) AND (BUT FIUV) TEST
739 -----
740
741 THIS IS A TEST OF THE (BUT EZBT Y8) FORK, THE (BUT
742 ENBT) FORK AND (BUT FIUV) FORK IN THE LOAD
743 INSTRUCTION FLOWS.
744 EACH OF THE PATTERNS:
745
746 0
747 +NUM
748 -NUM
749 -0
750
751 IS LOADED TWICE, ONCE WITH AC>0 THEN WITH AC=0.
752 AFTER EACH LOAD THE FPS IS CHECK TO INSURE THAT
753 CONTROL WAS PASSED THROUGH WITH THE FORKS PROPERLY.
754
755 TEST 25 ADDF, ADDD, SUBF AND SUBD WITH FSRC=AC=0 TEST
756 -----
757
758 THIS IS A TEST OF ADD AND SUB WITH FSRC=AC=0
759
760 TEST 26 ADDD AND SUB WITH FSRC=0
761 -----
762
763 THIS IS A TEST OF ADD AND SUB WITH FSRC=0.
764
765 TEST 27 SUBD WITH AC=0 TEST
766 -----
767
768 THIS IS A TEST OF SUBD WITH AC=0. BOTH POSITIVE AND
769 NEGATIVE FSRC'S ARE TRIED.
770
771 TEST 30 ADDD WITH AC=0 TEST
772 -----
773
774 POSITIVE AND NEGATIVE FSRC'S ARE TRIED.
775
776 TEST 31 ADDF AND ADDD WITH F(AC)=E(FSRC) AND (BUT FT) TEST
777 -----
778
779 THIS IS A TEST OF THE ADD INSTRUCTION WITH THE
780 OPERANDS HAVING EQUAL EXPONENTS. THE (BUT FT) FORK
781 IN THE ROUND/TRUNK FLOWS IS ALSO TESTED.
782
783 TEST 32 ADDF AND ADDD WITH E(AC) LESS THAN E(FSPC) TEST
784 -----
```

FORM 141 REV. 11-75 PRINTED IN U.S.A.

785  
786 THIS IS A TEST OF THE ADD AND ADDF INSTRUCTIONS AND  
787 THE ALIGN AC ALGORITHM FLOWS. THE CONSTANT (25 FOR  
788 FLOATING, 57 FOR DOUBLE) USED IS CHECKED. THEN  
789 SIMPLE AND WORST CASE ALIGNMENT SITUATIONS ARE  
790 TRIED. NOTE E(AC) IS LESS THAN E(FSRC)

791  
792  
793 TEST 33 ADD AND ADD WITH E(AC) GREATER THAN E(FSRC) TEST  
794 -----

795  
796 THIS IS A TEST OF THE ADD AND ADDF INSTRUCTIONS AND  
797 THE ALIGN FSPC ALGORITHM FLOWS. FIRST THE CONSTANT  
798 USED IS CHECKED. THEN SIMPLE AND WORST CASE  
799 ALIGNMENT SITUATIONS ARE TRIED. NOTE E(AC) IS  
800 GREATER THAN E(FSRC).

801  
802 TEST 34 ADD WITH NEGATIVE OPRANDS TEST  
803 -----

804  
805 THIS IS A TEST OF THE ADD INSTRUCTION WITH NEGATIVE  
806 OPERANDS. EVERY COMBINATION OF OPERAND SIGNS IS  
807 TRIED.

808  
809 TEST 35 SUBD TEST  
810 -----

811  
812 THIS IS A TEST OF THE SUBD INSTRUCTION. BOTH A  
813 POSITIVE AND A NEGATIVE NUMBER IS SUBTRACTED FROM IT  
814 SELF

815  
816 TEST 36 NORMALIZE ALGORITHM TEST  
817 -----

818  
819 THIS IS A TEST OF THE NORMALIZE FLOW ALGORITHM. TWO  
820 PATTERNS ARE USED, FIRST THE MINIMUM SITUATION  
821 REQUIRING ONE LEFT SHIFT AND THEN THE MAXIMUM  
822 SITUATION REQUIRING 56 SHIFTS.



841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
890  
891  
892  
893  
894  
895  
896

000213  
000001

MNUMBER=213  
PROGRAM=1

.LIST ME  
.NLIST MD,MC,CND

.ENABL ABS

.TITLE MAINDEC-11-DEFFPA-A PDP 11/34 FPP DIAGNOSTIC PART 1  
:\*COPYRIGHT (C) SEP 1976  
:\*DIGITAL EQUIPMENT CORP.  
:\*MAYNARD, MASS. 01754  
:\*  
:\*PROGRAM BY ANTHONY S. VEZZA  
:\*  
:\*THIS PROGRAM WAS ASSEMBLED USING THE PDP-11 MAINDEC SYSMAC  
:\*PACKAGE (MAINDEC-11-DZQAC-C2), SEPT 14, 1976.  
:\*  
STN=1  
SSWR=160000 ;HALT ON ERROR, LOOP ON TEST, INHIBIT ERROR TYP0UT

000001  
160000  
000244  
177400

FEVECT=244  
SSWR=177400

```

897      000200      $$WPMSK=200
898      000011      TAB=11
899      000015      CRLF=15
900
901      .SBTTL  BASIC DEFINITIONS
902
903      ;*INITIAL ADDRESS OF THE STACK POINTER *** 1100 ***
904      001100      STACK= 1100
905      .EQUIV  EMT,ERROR      ;;BASIC DEFINITION OF ERROR CALL
906      .EQUIV  IOT,SCOPE      ;;BASIC DEFINITION OF SCOPE CALL
907
908      ;*MISCELLANEOUS DEFINITIONS
909      000011      HT= 11      ;;CODE FOR HORIZONTAL TAB
910      000012      LF= 12      ;;CODE FOR LINE FEED
911      000015      CR= 15      ;;CODE FOR CARRIAGE RETURN
912      000200      CRLF= 200   ;;CODE FOR CARRIAGE RETURN-LINE FEED
913      177776      PS= 177776  ;;PROCESSOR STATUS WORD
914      .EQUIV  PS,PSW
915      177774      STKLMT= 177774 ;;STACK LIMIT REGISTER
916      177772      PIPQ= 177772 ;;PROGRAM INTERRUPT REQUEST REGISTER
917      177570      DSWP= 177570 ;;HARDWARE SWITCH REGISTER
918      177570      DDISP= 177570 ;;HARDWARE DISPLAY REGISTER
919
920      ;*GENERAL PURPOSE REGISTER DEFINITIONS
921      000000      R0= 00      ;;GENERAL REGISTER
922      000001      R1= 01      ;;GENERAL REGISTER
923      000002      R2= 02      ;;GENERAL REGISTER
924      000003      R3= 03      ;;GENERAL REGISTER
925      000004      R4= 04      ;;GENERAL REGISTER
926      000005      R5= 05      ;;GENERAL REGISTER
927      000006      R6= 06      ;;GENERAL REGISTER
928      000007      R7= 07      ;;GENERAL REGISTER
929      000006      SP= 06      ;;STACK POINTER
930      000007      PC= 07      ;;PROGRAM COUNTER
931
932      ;*PRIORITY LEVEL DEFINITIONS
933      000000      PR0= 0      ;;PRIORITY LEVEL 0
934      000040      PR1= 40      ;;PRIORITY LEVEL 1
935      000100      PR2= 100     ;;PRIORITY LEVEL 2
936      000140      PR3= 140     ;;PRIORITY LEVEL 3
937      000200      PR4= 200     ;;PRIORITY LEVEL 4
938      000240      PR5= 240     ;;PRIORITY LEVEL 5
939      000300      PR6= 300     ;;PRIORITY LEVEL 6
940      000340      PR7= 340     ;;PRIORITY LEVEL 7
941
942      ;*"SWITCH REGISTER" SWITCH DEFINITIONS
943      100000      SW15= 100000
944      040000      SW14= 40000
945      020000      SW13= 20000
946      010000      SW12= 10000
947      004000      SW11= 4000
948      002000      SW10= 2000
949      001000      SW09= 1000
950      000400      SW08= 400
951      000200      SW07= 200
952      000100      SW06= 100
  
```

```

953 000040 SW05= 40
954 000020 SW04= 20
955 000010 SW03= 10
956 000004 SW02= 4
957 000002 SW01= 2
958 000001 SW00= 1
959 .EQUIV SW09,SW9
960 .EQUIV SW08,SW8
961 .EQUIV SW07,SW7
962 .EQUIV SW06,SW6
963 .EQUIV SW05,SW5
964 .EQUIV SW04,SW4
965 .EQUIV SW03,SW3
966 .EQUIV SW02,SW2
967 .EQUIV SW01,SW1
968 .EQUIV SW00,SW0
  
```

;\*DATA BIT DEFINITIONS (BIT00 TO BIT15)

```

971 100000 BIT15= 100000
972 040000 BIT14= 40000
973 020000 BIT13= 20000
974 010000 BIT12= 10000
975 004000 BIT11= 4000
976 002000 BIT10= 2000
977 001000 BIT09= 1000
978 000400 BIT08= 400
979 000200 BIT07= 200
980 000100 BIT06= 100
981 000040 BIT05= 40
982 000020 BIT04= 20
983 000010 BIT03= 10
984 000004 BIT02= 4
985 000002 BIT01= 2
986 000001 BIT00= 1
987 .EQUIV BIT09,BIT9
988 .EQUIV BIT08,BIT8
989 .EQUIV BIT07,BIT7
990 .EQUIV BIT06,BIT6
991 .EQUIV BIT05,BIT5
992 .EQUIV BIT04,BIT4
993 .EQUIV BIT03,BIT3
994 .EQUIV BIT02,BIT2
995 .EQUIV BIT01,BIT1
996 .EQUIV BIT00,BIT0
  
```

;\*BASIC "CPU" TRAP VECTOR ADDRESSES

```

999 000004 ERRVEC= 4 ;:TIME OUT AND OTHER ERRORS
1000 000010 RESVEC= 10 ;:RESERVED AND ILLEGAL INSTRUCTIONS
1001 000014 TRITVEC=14 ;:"T" BIT
1002 000014 TRIVEC= 14 ;:TRACE TRAP
1003 000014 BPTVEC= 14 ;:BREAKPOINT TRAP (BPT)
1004 000020 IOTVEC= 20 ;:INPUT/OUTPUT TRAP (IOT) **SCOPE**
1005 000024 PWRVEC= 24 ;:POWER FAIL
1006 000030 EMTVEC= 30 ;:EMULATOR TRAP (EMT) **ERROR**
1007 000034 TRAPVEC=34 ;:"TRAP" TRAP
1008 000060 TKVEC= 60 ;:TTY KEYBOARD VECTOR
  
```

```

1009      000064      TPVEC= 64          ;;TTY PRINTER VECTOR
1010      000240      PIRQVEC=240       ;;PROGRAM INTERRUPT REQUEST VECTOR
1011      -           .SBTTL FPP REGISTER DEFINITIONS
1012      000000      AC0      =%0
1013      000001      AC1      =%1
1014      000002      AC2      =%2
1015      000003      AC3      =%3
1016      000004      AC4      =%4
1017      000005      AC5      =%5
1018      000006      AC6      =%6
1019      000007      AC7      =%7
1020
1021      -           .SBTTL TRAP CATCHER
1022
1023      000000      :=0
1024      -           ;*ALL UNUSED LOCATIONS FROM 4 - 776 CONTAIN A ".+2,HALT"
1025      -           ;*SEQUENCE TO CATCH ILLEGAL TRAPS AND INTERRUPTS
1026      -           ;*LOCATION 0 CONTAINS 0 TO CATCH IMPROPERLY LOADED VECTORS
1027      000174      :=174
1028      000174      000000      DISPREG: .WORD 0          ;;SOFTWARE DISPLAY REGISTER
1029      000176      000000      SWREG:   .WORD 0          ;;SOFTWARE SWITCH REGISTER
1030      -           .SBTTL STARTING ADDRESS(ES)
1031      000200      000137      003606      JMP      @#START ;;JUMP TO STARTING ADDRESS OF PROGRAM
  
```

Line	Address	Value	Label	Type	Description
1032			.SBTTI, COMMON TAGS		
1033					
1034			:*****		
1035			:*THIS TABLE CONTAINS VARIOUS COMMON STORAGE LOCATIONS		
1036			:*USED IN THE PROGRAM.		
1037					
1038	001100				
1039	001100		SCMTAG:		::START OF COMMON TAGS
1040	000000			WORD 0	
1041	001102	000	STSTNM:	BYTE 0	::CONTAINS THE TEST NUMBER
1042	001103	000	SERFLG:	BYTE 0	::CONTAINS ERROR FLAG
1043	001104	000000	SICNT:	WORD 0	::CONTAINS SUBTEST ITERATION COUNT
1044	001106	000000	SLPADR:	WORD 0	::CONTAINS SCOPE LOOP ADDRESS
1045	001110	000000	SLPERR:	WORD 0	::CONTAINS SCOPE RETURN FOR ERRORS
1046	001112	000000	SERTTL:	WORD 0	::CONTAINS TOTAL ERRORS DETECTED
1047	001114	000	SITEMB:	BYTE 0	::CONTAINS ITEM CONTROL BYTE
1048	001115	001	SERMAX:	BYTE 1	::CONTAINS MAX. ERRORS PER TEST
1049	001116	000000	SERRPC:	WORD 0	::CONTAINS PC OF LAST ERROR INSTRUCTION
1050	001120	000000	SGDADR:	WORD 0	::CONTAINS ADDRESS OF 'GOOD' DATA
1051	001122	000000	SBDADR:	WORD 0	::CONTAINS ADDRESS OF 'BAD' DATA
1052	001124	000000	SGDDAT:	WORD 0	::CONTAINS 'GOOD' DATA
1053	001126	000000	SBDDAT:	WORD 0	::CONTAINS 'BAD' DATA
1054	001130	000000		WORD 0	::RESERVED--NOT TO BE USED
1055	001132	000000		WORD 0	
1056	001134	000	SAUTOR:	BYTE 0	::AUTOMATIC MODE INDICATOR
1057	001135	000	SINTAG:	BYTE 0	::INTERRUPT MODE INDICATOR
1058	001136	000000		WORD 0	
1059	001140	177570	SWR:	WORD DSWR	::ADDRESS OF SWITCH REGISTER
1060	001142	177570	DISPLAY:	WORD DDISP	::ADDRESS OF DISPLAY REGISTER
1061	001144	177560	STKS:	177560	::TTY KBD STATUS
1062	001146	177562	STKB:	177562	::TTY KBD BUFFER
1063	001150	177564	STPS:	177564	::TTY PRINTER STATUS REG. ADDRESS
1064	001152	177566	STPB:	177566	::TTY PRINTER BUFFER REG. ADDRESS
1065	001154	000	SNULL:	BYTE 0	::CONTAINS NULL CHARACTER FOR FILLS
1066	001155	002	SFILLS:	BYTE 2	::CONTAINS # OF FILLER CHARACTERS REQUIRED
1067	001156	012	SFILLC:	BYTE 12	::INSERT FILL CHARS. AFTER A "LINE FEED"
1068	001157	000	STPFLG:	BYTE 0	::"TERMINAL AVAILABLE" FLAG (BIT<07>=0=YES)
1069	001160	000000	SREGAD:	WORD 0	::CONTAINS THE ADDRESS FROM
1070					::WHICH (SREG0) WAS OBTAINED
1071	001162	000000	SREG0:	WORD 0	::CONTAINS ((SREGAD)+0)
1072	001164	000000	SREG1:	WORD 0	::CONTAINS ((SREGAD)+2)
1073	001166	000000	SREG2:	WORD 0	::CONTAINS ((SREGAD)+4)
1074	001170	000000	SREG3:	WORD 0	::CONTAINS ((SREGAD)+6)
1075	001172	000000	SREG4:	WORD 0	::CONTAINS ((SREGAD)+10)
1076	001174	000000	SREG5:	WORD 0	::CONTAINS ((SREGAD)+12)
1077	001176	000000	SREG6:	WORD 0	::CONTAINS ((SREGAD)+14)
1078	001200	000000	SREG7:	WORD 0	::CONTAINS ((SREGAD)+16)
1079	001202	000000	SREG10:	WORD 0	::CONTAINS ((SREGAD)+20)
1080	001204	000000	SREG11:	WORD 0	::CONTAINS ((SREGAD)+22)
1081	001206	000000	SREG12:	WORD 0	::CONTAINS ((SREGAD)+24)
1082	001210	000000	SREG13:	WORD 0	::CONTAINS ((SREGAD)+26)
1083	001212	000000	SREG14:	WORD 0	::CONTAINS ((SREGAD)+30)
1084	001214	000000	SREG15:	WORD 0	::CONTAINS ((SREGAD)+32)
1085	001216	000000	SREG16:	WORD 0	::CONTAINS ((SREGAD)+34)
1086	001220	000000	SREG17:	WORD 0	::CONTAINS ((SREGAD)+36)
1087	001222	000000	SREG20:	WORD 0	::CONTAINS ((SREGAD)+40)

PRINTED IN U.S.A.

```

1088 001224 000000 SREG21: WORD 0 ;:CONTAINS ((SREGAD)+42)
1089 001226 000000 SREG22: WORD 0 ;:CONTAINS ((SREGAD)+44)
1090 001230 000000 SREG23: WORD 0 ;:CONTAINS ((SREGAD)+46)
1091 001232 000000 STMP0: WORD 0 ;:USER DEFINED
1092 001234 000000 STMP1: WORD 0 ;:USER DEFINED
1093 001236 000000 STMP2: WORD 0 ;:USER DEFINED
1094 001240 000000 STMP3: WORD 0 ;:USER DEFINED
1095 001242 000000 STMP4: WORD 0 ;:USER DEFINED
1096 001244 000000 STMP5: WORD 0 ;:USER DEFINED
1097 001246 000000 STMP6: WORD 0 ;:USER DEFINED
1098 001250 000000 STMP7: WORD 0 ;:USER DEFINED
1099 001252 000000 STMP10: WORD 0 ;:USER DEFINED
1100 001254 000000 STMP11: WORD 0 ;:USER DEFINED
1101 001256 000000 STMP12: WORD 0 ;:USER DEFINED
1102 001260 000000 STMP13: WORD 0 ;:USER DEFINED
1103 001262 000000 STMP14: WORD 0 ;:USER DEFINED
1104 001264 000000 STMP15: WORD 0 ;:USER DEFINED
1105 001266 000000 STMP16: WORD 0 ;:USER DEFINED
1106 001270 000000 STMP17: WORD 0 ;:USER DEFINED
1107 001272 000000 STMP20: WORD 0 ;:USER DEFINED
1108 001274 000000 STMP21: WORD 0 ;:USER DEFINED
1109 001276 000000 STMP22: WORD 0 ;:USER DEFINED
1110 001300 000000 STMP23: WORD 0 ;:USER DEFINED
1111 001302 000000 STIMES: 0 ;:MAX. NUMBER OF ITERATIONS
1112 001304 000000 SESCAPE: 0 ;:ESCAPE ON ERROR ADDRESS
1113 001306 177607 000377 SBELL: ASCII <207><377><377> ;:CODE FOR BELL
1114 001312 077 SQUES: ASCII /?/ ;:QUESTION MARK
1115 001313 015 SCRLF: ASCII <15> ;:CARRIAGE RETURN
1116 001314 000012 SLF: ASCII <12> ;:LINE FEED
1117 ;:*****
1118 .SBTTI APT MAILBOX-ETABLE
1119
1120 ;:*****
1121 .EVEN
1122 001316 SMAIL: ;:APT MAILBOX
1123 001316 000000 SMSCTY: WORD AMSGTY ;:MESSAGE TYPE CODE
1124 001320 000000 SFATAL: WORD AFATAL ;:FATAL ERROR NUMBER
1125 001322 000000 STESTN: WORD ATESTN ;:TEST NUMBER
1126 001324 000000 SPASS: WORD APASS ;:PASS COUNT
1127 001326 000000 SDEVCT: WORD ADEVCT ;:DEVICE COUNT
1128 001330 000000 SUNIT: WORD AUNIT ;:I/O UNIT NUMBER
1129 001332 000000 SMSGAD: WORD AMSGAD ;:MESSAGE ADDRESS
1130 001334 000000 SMSGLG: WORD AMSGLG ;:MESSAGE LENGTH
1131 001336 SETABLE: ;:APT ENVIRONMENT TABLE
1132 001336 000 SENV: BYTE AENV ;:ENVIRONMENT BYTE
1133 001337 000 SENVM: BYTE AENVM ;:ENVIRONMENT MODE BITS
1134 001340 000000 SSWREG: WORD ASWREG ;:APT SWITCH REGISTER
1135 001342 000000 SUSWR: WORD AUSWR ;:USER SWITCHES
1136 001344 000000 SCPUOP: WORD ACPUOP ;:CPU TYPE, OPTIONS
1137 ;:*
1138 ;:* 11/04=01, 11/05=02, 11/20=03, 11/40=04, 11/45=05
1139 ;:* 11/70=06, PDQ=07, Q=10
1140 ;:* BIT 10=REAL TIME CLOCK
1141 ;:* BIT 9=FLOATING POINT PROCESSOR
1142 ;:* BIT 8=MEMORY MANAGEMENT
1143 001346 000 SMAMS1: .BYTE AMAMS1 ;:HIGH ADDRESS, M.S. BYTE
    
```

JACKET BUSINESS FORMS, INC. HO  
 FORM 1413  
 PRINTED IN U.S.A.

```

1144 001347 000 SMTYP1: .BYTE AMTYP1 ;;MEM. TYPE,BLK#1
1145 :* MEM. TYPE BYTE -- (HIGH BYTE)
1146 :* 900 NSEC COPE=001
1147 :* 300 NSEC BIPOLAR=002
1148 :* 500 NSEC MOS=003
1149 001350 000000 $MADR1: .WORD AMADR1 ;;HIGH ADDRESS,BLK#1
1150 :* MEM. LAST ADDR.=3 BYTES,THIS WORD AND LOW OF "TYPE" ABOVE
1151 001352 000 $MAMS2: .BYTE AMAMS2 ;;HIGH ADDRESS,M.S. BYTE
1152 001353 000 SMTYP2: .BYTE AMTYP2 ;;MEM. TYPE,BLK#2
1153 001354 000000 $MADR2: .WORD AMADR2 ;;MEM. LAST ADDRESS,BLK#2
1154 001356 000 $MAMS3: .BYTE AMAMS3 ;;HIGH ADDRESS,M.S. BYTE
1155 001357 000 SMTYP3: .BYTE AMTYP3 ;;MEM. TYPE,BLK#3
1156 001360 000000 $MADR3: .WORD AMADR3 ;;MEM. LAST ADDRESS,BLK#3
1157 001362 000 $MAMS4: .BYTE AMAMS4 ;;HIGH ADDRESS,M.S. BYTE
1158 001363 000 SMTYP4: .BYTE AMTYP4 ;;MEM. TYPE,BLK#4
1159 001364 000000 $MADR4: .WORD AMADR4 ;;MEM. LAST ADDRESS,BLK#4
1160 001366 000000 $VECT1: .WORD AVECT1 ;;INTERRUPT VECTOR#1,BUS PRIORITY#1
1161 001370 000000 $VECT2: .WORD AVECT2 ;;INTERRUPT VECTOR#2BUS PRIORITY#2
1162 001372 000000 $BASE: .WORD ABASE ;;BASE ADDRESS OF EQUIPMENT UNDER TEST
1163 001374 000000 $DEVN: .WORD ADEVN ;;DEVICE MAP
1164 001376 000000 $CDW1: .WORD ACDW1 ;;CONTROLLER DESCRIPTION WORD#1
1165 001400 000000 $CDW2: .WORD ACDW2 ;;CONTROLLER DESCRIPTION WORD#2
1166 001402 000000 $DDW0: .WORD ADDW0 ;;DEVICE DESCRIPTOR WORD#0
1167 001404 000000 $DDW1: .WORD ADDW1 ;;DEVICE DESCRIPTOR WORD#1
1168 001406 000000 $DDW2: .WORD ADDW2 ;;DEVICE DESCRIPTOR WORD#2
1169 001410 000000 $DDW3: .WORD ADDW3 ;;DEVICE DESCRIPTOR WORD#3
1170 001412 000000 $DDW4: .WORD ADDW4 ;;DEVICE DESCRIPTOR WORD#4
1171 001414 000000 $DDW5: .WORD ADDW5 ;;DEVICE DESCRIPTOR WORD#5
1172 001416 000000 $DDW6: .WORD ADDW6 ;;DEVICE DESCRIPTOR WORD#6
1173 001420 000000 $DDW7: .WORD ADDW7 ;;DEVICE DESCRIPTOR WORD#7
1174 001422 000000 $DDW8: .WORD ADDW8 ;;DEVICE DESCRIPTOR WORD#8
1175 001424 000000 $DDW9: .WORD ADDW9 ;;DEVICE DESCRIPTOR WORD#9
1176 001426 000000 $DDW10: .WORD ADDW10 ;;DEVICE DESCRIPTOR WORD#10
1177 001430 000000 $DDW11: .WORD ADDW11 ;;DEVICE DESCRIPTOR WORD#11
1178 001432 000000 $DDW12: .WORD ADDW12 ;;DEVICE DESCRIPTOR WORD#12
1179 001434 000000 $DDW13: .WORD ADDW13 ;;DEVICE DESCRIPTOR WORD#13
1180 001436 000000 $DDW14: .WORD ADDW14 ;;DEVICE DESCRIPTOR WORD#14
1181 001440 000000 $DDW15: .WORD ADDW15 ;;DEVICE DESCRIPTOR WORD#15
1182
1183
1184 001442 $FTEND:
1185

```

FORM 11-100 (REV. 11-76) PRINTED IN U.S.A.

1186 .SBTTL ERROR POINTER TABLE

1187  
 1188 : \* THIS TABLE CONTAINS THE INFORMATION FOR EACH ERROR THAT CAN OCCUR.  
 1189 : \* THE INFORMATION IS OBTAINED BY USING THE INDEX NUMBER FOUND IN  
 1190 : \* LOCATION SITEMB. THIS NUMBER INDICATES WHICH ITEM IN THE TABLE IS PERTINENT.  
 1191 : \* NOTE1: IF SITEMB IS 0 THE ONLY PERTINENT DATA IS (SERRPC).  
 1192 : \* NOTE2: EACH ITEM IN THE TABLE CONTAINS 4 POINTERS EXPLAINED AS FOLLOWS:

- 1193
- 1194 : \* EM : : POINTS TO THE ERROR MESSAGE
- 1195 : \* DH : : POINTS TO THE DATA HEADER
- 1196 : \* DT : : POINTS TO THE DATA
- 1197 : \* DF : : POINTS TO THE DATA FORMAT
- 1198
- 1199

Line	SiteMB	EM	DH	DT	DF	Item
1200	001442					SERRTR:
1201						: ITEM 1
1202	001442	043025	063250	067664		WORD EM1,DH1,DT1,DF1
1203	001450	066611				
1204						: ITEM 2
1205	001452	043062	063340	067706		WORD EM2,DH2,DT2,DF2
1206	001460	066621				
1207						: ITEM 3
1208	001462	043126	063433	067730		WORD EM3,DH3,DT3,DF3
1209	001470	066631				
1210						: ITEM 4
1211	001472	043173	063524	067730		WORD EM4,DH4,DT4,DF4
1212	001500	066631				
1213						: ITEM 5
1214	001502	043233	063620	067752		WORD EM5,DH5,DT5,DF5
1215	001510	066641				
1216						: ITEM 6
1217	001512	043267	063620	070004		WORD EM6,DH6,DT6,DF6
1218	001520	066641				
1219						: ITEM 7
1220	001522	043321	063620	070004		WORD EM7,DH7,DT7,DF7
1221	001530	066641				
1222						: ITEM 10
1223	001532	043233	063620	070004		WORD EM10,DH10,DT10,DF10
1224	001540	066641				
1225						: ITEM 11
1226	001542	043354	063620	070004		WORD EM11,DH11,DT11,DF11
1227	001550	066641				
1228						: ITEM 12
1229	001552	000000	000000	070026		WORD EM12,DH12,DT12,DF12
1230	001560	066655				
1231						: ITEM 13
1232	001562	000000	000000	070114		WORD EM13,DH13,DT13,DF13
1233	001570	066707				
1234						: ITEM 14
1235	001572	043435	063620	070004		WORD EM14,DH14,DT14,DF14
1236	001600	066641				
1237						: ITEM 15
1238	001602	043560	063620	070004		WORD EM15,DH15,DT15,DF15
1239	001610	066641				
1240						: ITEM 16
1241	001612	043703	063660	070146		WORD EM16,DH16,DT16,DF16



1242	001620	066723				
1243					:ITEM 17	
1244	001622	043754	063740	067730	WORD	EM17,DH17,DT17,DF17
1245	001630	066631				
1246					:ITEM 20	
1247	001632	044207	064030	070166	WORD	EM20,DH20,DT20,DF20
1248	001640	066732				
1249					:ITEM 21	
1250	001642	044365	063620	070210	WORD	EM21,DH21,DT21,DF21
1251	001650	066742				
1252					:ITEM 22	
1253	001652	044516	064116	070222	WORD	EM22,DH22,DT22,DF22
1254	001660	066746				
1255					:ITEM 23	
1256	001662	044516	064153	070250	WORD	EM23,DH23,DT23,DF23
1257	001670	066760				
1258					:ITEM 24	
1259	001672	044516	064311	070272	WORD	EM24,DH24,DT24,DF24
1260	001700	066770				
1261					:ITEM 25	
1262	001702	044603	064450	070316	WORD	EM25,DH25,DT25,DF25
1263	001710	066777				
1264					:ITEM 26	
1265	001712	044716	064512	070366	WORD	EM26,DH26,DT26,DF26
1266	001720	067023				
1267					:ITEM 27	
1268	001722	044716	064512	070442	WORD	EM27,DH27,DT27,DF27
1269	001730	067050				
1270					:ITEM 30	
1271	001732	044764	000000	070504	WORD	EM30,DH30,DT30,DF30
1272	001740	067070				
1273					:ITEM 31	
1274	001742	045036	064512	070366	WORD	EM31,DH31,DT31,DF31
1275	001750	067023				
1276					:ITEM 32	
1277	001752	045036	064512	070442	WORD	EM32,DH32,DT32,DF32
1278	001760	067050				
1279					:ITEM 33	
1280	001762	045104	064600	070536	WORD	EM33,DH33,DT33,DF33
1281	001770	067104				
1282					:ITEM 34	
1283	001772	045145	064600	070614	WORD	EM34,DH34,DT34,DF34
1284	002000	067132				
1285					:ITEM 35	
1286	002002	045247	064600	070614	WORD	EM35,DH35,DT35,DF35
1287	002010	067132				
1288					:ITEM 36	
1289	002012	045351	064600	070614	WORD	EM36,DH36,DT36,DF36
1290	002020	067132				
1291					:ITEM 37	
1292	002022	045452	064600	070614	WORD	EM37,DH37,DT37,DF37
1293	002030	067132				
1294					:ITEM 40	
1295	002032	045553	064600	070536	WORD	EM40,DH40,DT40,DF40
1296	002040	067156				
1297					:ITEM 41	

1298	002042	045724	000000	070666	WORD	EM41,DH41,DT41,DF41
1299	002050	067204				
1300					:ITEM 42	
1301	002052	045761	064703	070720	WORD	EM42,DH42,DT42,DF42
1302	002060	067220				
1303					:ITEM 43	
1304	002062	046102	064703	070720	WORD	EM43,DH43,DT43,DF43
1305	002070	067220				
1306					:ITEM 44	
1307	002072	046223	000000	070776	WORD	EM44,DH44,DT44,DF44
1308	002100	067246				
1309					:ITEM 45	
1310	002102	046223	065005	071046	WORD	EM45,DH45,DT45,DF45
1311	002110	067271				
1312					:ITEM 46	
1313	002112	046266	065024	071122	WORD	EM46,DH46,DT46,DF46
1314	002120	067316				
1315					:ITEM 47	
1316	002122	046344	065005	071210	WORD	EM47,DH47,DT47,DF47
1317	002130	067350				
1318					:ITEM 50	
1319	002132	046462	065050	070614	WORD	EM50,DH50,DT50,DF50
1320	002140	067365				
1321					:ITEM 51	
1322	002142	046560	065050	071242	WORD	EM51,DH51,DT51,DF51
1323	002150	067411				
1324					:ITEM 52	
1325	002152	046621	063620	071210	WORD	EM52,DH52,DT52,DF52
1326	002160	067350				
1327					:ITEM 53	
1328	002162	046742	064512	071300	WORD	EM53,DH53,DT53,DF53
1329	002170	067427				
1330					:ITEM 54	
1331	002172	047137	065122	071320	WORD	EM54,DH54,DT54,DF54
1332	002200	067436				
1333					:ITEM 55	
1334	002202	047203	063620	071210	WORD	EM55,DH55,DT55,DF55
1335	002210	067350				
1336					:ITEM 56	
1337	002212	047324	064512	071300	WORD	EM56,DH56,DT56,DF56
1338	002220	067427				
1339					:ITEM 57	
1340	002222	047521	065122	071320	WORD	EM57,DH57,DT57,DF57
1341	002230	067436				
1342					:ITEM 60	
1343	002232	047565	064512	071300	WORD	EM60,DH60,DT60,DF60
1344	002240	067427				
1345					:ITEM 61	
1346	002242	047762	065122	071320	WORD	EM61,DH61,DT61,DF61
1347	002250	067436				
1348					:ITEM 62	
1349	002252	050026	065122	071320	WORD	EM62,DH62,DT62,DF62
1350	002260	067436				
1351					:ITEM 63	
1352	002262	050220	065122	071320	WORD	EM63,DH63,DT63,DF63
1353	002270	067436				

1354					:ITEM 64	
1355	002272	050412	065232	071356	.WORD	EM64,DH64,DT64,DF64
1356	002300	067454				
1357					:ITEM 65	
1358	002302	050412	065163	071356	.WORD	EM65,DH65,DT65,DF65
1359	002310	067454				
1360					:ITEM 66	
1361	002312	050546	065122	071320	.WORD	EM66,DH66,DT66,DF66
1362	002320	067436				
1363					:ITEM 67	
1364	002322	050611	063620	070210	.WORD	EM67,DH67,DT67,DF67
1365	002330	066742				
1366					:ITEM 70	
1367	002332	051042	063620	071376	.WORD	EM70,DH70,DT70,DF70
1368	002340	067463				
1369					:ITEM 71	
1370	002342	051165	064450	071376	.WORD	EM71,DH71,DT71,DF71
1371	002350	067463				
1372					:ITEM 72	
1373	002352	051267	064512	071444	.WORD	EM72,DH72,DT72,DF72
1374	002360	067505				
1375					:ITEM 73	
1376	002362	051343	065122	071320	.WORD	EM73,DH73,DT73,DF73
1377	002370	067436				
1378					:ITEM 74	
1379	002372	051403	063620	070210	.WORD	EM74,DH74,DT74,DF74
1380	002400	066742				
1381					:ITEM 75	
1382	002402	051634	063620	071376	.WORD	EM75,DH75,DT75,DF75
1383	002410	067463				
1384					:ITEM 76	
1385	002412	051757	064450	071376	.WORD	EM76,DH76,DT76,DF76
1386	002420	067463				
1387					:ITEM 77	
1388	002422	052061	064512	071444	.WORD	EM77,DH77,DT77,DF77
1389	002430	067505				
1390					:ITEM 100	
1391	002432	052135	065122	071320	.WORD	EM100,DH100,DT100,DF100
1392	002440	067436				
1393					:ITEM 101	
1394	002442	052175	063620	071376	.WORD	EM101,DH101,DT101,DF101
1395	002450	067463				
1396					:ITEM 102	
1397	002452	052321	064512	071376	.WORD	EM102,DH102,DT102,DF102
1398	002460	067505				
1399					:ITEM 103	
1400	002462	052373	064450	071376	.WORD	EM103,DH103,DT103,DF103
1401	002470	067463				
1402					:ITEM 104	
1403	002472	052476	065122	071320	.WORD	EM104,DH104,DT104,DF104
1404	002500	067436				
1405					:ITEM 105	
1406	002502	052537	063620	071376	.WORD	EM105,DH105,DT105,DF105
1407	002510	067463				
1408					:ITEM 106	
1409	002512	052664	064512	071444	.WORD	EM106,DH106,DT106,DF106

1410	002520	067505			
1411				:ITEM 107	
1412	002522	052737	064450	071376	.WORD EM107,DH107,DT107,DF107
1413	002530	067463			
1414				:ITEM 110	
1415	002532	053043	065122	071320	.WORD EM110,DH110,DT110,DF110
1416	002540	067436			
1417				:ITEM 111	
1418	002542	053105	064450	071464	.WORD EM111,DH111,DT111,DF111
1419	002550	067514			
1420				:ITEM 112	
1421	002552	053105	065320	071464	.WORD EM112,DH112,DT112,DF112
1422	002560	067514			
1423				:ITEM 113	
1424	002562	053207	064450	071464	.WORD EM113,DH113,DT113,DF113
1425	002570	067514			
1426				:ITEM 114	
1427	002572	053207	065320	071464	.WORD EM114,DH114,DT114,DF114
1428	002600	067514			
1429				:ITEM 115	
1430	002602	053105	065537	071464	.WORD EM115,DH115,DT115,DF115
1431	002610	067514			
1432				:ITEM 116	
1433	002612	053207	065537	071464	.WORD EM116,DH116,DT116,DF116
1434	002620	067514			
1435				:ITEM 117	
1436	002622	053311	063740	067730	.WORD EM117,DH117,DT117,DF117
1437	002630	066631			
1438				:ITEM 120	
1439	002632	053445	066023	067730	.WORD EM120,DH120,DT120,DF120
1440	002640	066631			
1441				:ITEM 121	
1442	002642	053601	063620	071210	.WORD EM121,DH121,DT121,DF121
1443	002650	067350			
1444				:ITEM 122	
1445	002652	053720	065050	070614	.WORD EM122,DH122,DT122,DF122
1446	002660	067365			
1447				:ITEM 123	
1448	002662	054017	065050	071242	.WORD EM123,DH123,DT123,DF123
1449	002670	067411			
1450				:ITEM 124	
1451	002672	054060	063740	071476	.WORD EM124,DH124,DT124,DF124
1452	002700	067520			
1453				:ITEM 125	
1454	002702	054153	063740	071476	.WORD EM125,DH125,DT125,DF125
1455	002710	067520			
1456				:ITEM 126	
1457	002712	054243	063620	071464	.WORD EM126,DH126,DT126,DF126
1458	002720	067514			
1459				:ITEM 127	
1460	002722	054452	065122	071464	.WORD EM127,DH127,DT127,DF127
1461	002730	067514			
1462				:ITEM 130	
1463	002732	054665	066023	067730	.WORD EM130,DH130,DT130,DF130
1464	002740	066631			
1465				:ITEM 131	

1466	002742	054765	065122	071562	.WORD	EM131,DH131,DT131,DF131
1467	002750	067551				
1468					:ITEM 132	
1469	002752	055025	065122	071562	.WORD	EM132,DH132,DT132,DF132
1470	002760	067551				
1471					:ITEM 133	
1472	002762	055065	066113	071624	.WORD	EM133,DH133,DT133,DF133
1473	002770	067571				
1474					:ITEM 134	
1475	002772	055124	066113	071624	.WORD	EM134,DH134,DT134,DF134
1476	003000	067571				
1477					:ITEM 135	
1478	003002	055163	066113	071624	.WORD	EM135,DH135,DT135,DF135
1479	003010	067571				
1480					:ITEM 136	
1481	003012	055222	066113	071624	.WORD	EM136,DH136,DT136,DF136
1482	003020	067571				
1483					:ITEM 137	
1484	003022	055065	066223	071676	.WORD	EM137,DH137,DT137,DF137
1485	003030	067615				
1486					:ITEM 140	
1487	003032	055124	066223	071676	.WORD	EM140,DH140,DT140,DF140
1488	003040	067615				
1489					:ITEM 141	
1490	003042	055163	066223	071676	.WORD	EM141,DH141,DT141,DF141
1491	003050	067615				
1492					:ITEM 142	
1493	003052	055222	066223	071676	.WORD	EM142,DH142,DT142,DF142
1494	003060	067615				
1495					:ITEM 143	
1496	003062	055261	066113	071624	.WORD	EM143,DH143,DT143,DF143
1497	003070	067571				
1498					:ITEM 144	
1499	003072	055314	066113	071624	.WORD	EM144,DH144,DT144,DF144
1500	003100	067571				
1501					:ITEM 145	
1502	003102	055261	066223	071676	.WORD	EM145,DH145,DT145,DF145
1503	003110	067615				
1504					:ITEM 146	
1505	003112	055314	066223	071676	.WORD	EM146,DH146,DT146,DF146
1506	003120	067615				
1507					:ITEM 147	
1508	003122	055347	065122	071624	.WORD	EM147,DH147,DT147,DF147
1509	003130	067571				
1510					:ITEM 150	
1511	003132	055347	066413	071624	.WORD	EM150,DH150,DT150,DF150
1512	003140	067571				
1513					:ITEM 151	
1514	003142	055347	066223	071676	.WORD	EM151,DH151,DT151,DF151
1515	003150	067615				
1516					:ITEM 152	
1517	003152	055401	066113	071624	.WORD	EM152,DH152,DT152,DF152
1518	003160	067571				
1519					:ITEM 153	
1520	003162	055401	066223	071676	.WORD	EM153,DH153,DT153,DF153
1521	003170	067615				

1522					:ITEM 154	
1523	003172	055433	066504	071716	.WORD	EM154,DH154,DT154,DF154
1524	003200	067625				
1525					:ITEM 155	
1526	003202	055665	066504	071716	.WORD	EM155,DH155,DT155,DF155
1527	003210	067625				
1528					:ITEM 156	
1529	003212	056120	065122	071624	.WORD	EM156,DH156,DT156,DF156
1530	003220	067571				
1531					:ITEM 157	
1532	003222	056335	065122	071624	.WORD	EM157,DH157,DT157,DF157
1533	003230	067571				
1534					:ITEM 160	
1535	003232	056554	065122	071624	.WORD	EM160,DH160,DT160,DF160
1536	003240	067571				
1537					:ITEM 161	
1538	003242	056761	065122	071624	.WORD	EM161,DH161,DT161,DF161
1539	003250	067631				
1540					:ITEM 162	
1541	003252	057166	065122	071624	.WORD	EM162,DH162,DT162,DF162
1542	003260	067571				
1543					:ITEM 163	
1544	003262	057233	065122	071624	.WORD	EM163,DH163,DT163,DF163
1545	003270	067631				
1546					:ITEM 164	
1547	003272	057300	063740	067730	.WORD	EM164,DH164,DT164,DF164
1548	003300	066631				
1549					:ITEM 165	
1550	003302	057345	063740	067730	.WORD	EM165,DH165,DT165,DF165
1551	003310	066631				
1552					:ITEM 166	
1553	003312	057412	065122	071624	.WORD	EM166,DH166,DT166,DF166
1554	003320	067571				
1555					:ITEM 167	
1556	003322	057522	065122	071624	.WORD	EM167,DH167,DT167,DF167
1557	003330	067571				
1558					:ITEM 170	
1559	003332	057761	065122	071624	.WORD	EM170,DH170,DT170,DF170
1560	003340	067631				
1561					:ITEM 171	
1562	003342	060071	065122	071624	.WORD	EM171,DH171,DT171,DF171
1563	003350	067631				
1564					:ITEM 172	
1565	003352	060330	065122	071624	.WORD	EM172,DH172,DT172,DF172
1566	003360	067571				
1567					:ITEM 173	
1568	003362	060567	065122	071624	.WORD	EM173,DH173,DT173,DF173
1569	003370	067571				
1570					:ITEM 174	
1571	003372	061026	065122	071624	.WORD	EM174,DH174,DT174,DF174
1572	003400	067631				
1573					:ITEM 175	
1574	003402	061265	065122	071624	.WORD	EM175,DH175,DT175,DF175
1575	003410	067631				
1576					:ITEM 176	
1577	003412	061524	065122	071624	.WORD	EM176,DH176,DT176,DF176

```

1578 003420 067571
1579                                     :ITEM 177
1580 003422 061661 065122 071624      .WORD  EM177,DH177,DT177,DF177
1581 003430 067571
1582                                     :ITEM 200
1583 003432 062016 065122 071624      .WORD  EM200,DH200,DT200,DF200
1584 003440 067571
1585                                     :ITEM 201
1586 003442 062153 065122 071624      .WORD  EM201,DH201,DT201,DF201
1587 003450 067571
1588                                     :ITEM 202
1589 003452 062310 065122 071624      .WORD  EM202,DH202,DT202,DF202
1590 003460 067571
1591                                     :ITEM 203
1592 003462 062445 065122 071624      .WORD  EM203,DH203,DT203,DF203
1593 003470 067571
1594                                     :ITEM 204
1595 003472 062602 065122 071624      .WORD  EM204,DH204,DT204,DF204
1596 003500 067571
1597                                     :ITEM 205
1598 003502 062737 063740 067730      .WORD  EM205,DH205,DT205,DF205
1599 003510 066631
1600                                     :ITEM 206
1601 003512 063004 065122 071624      .WORD  EM206,DH206,DT206,DF206
1602 003520 067571
1603                                     :ITEM 207
1604 003522 063051 065122 071624      .WORD  EM207,DH207,DT207,DF207
1605 003530 067571
1606                                     :ITEM 210
1607 003532 063173 065122 071624      .WORD  EM210,DH210,DT210,DF210
1608 003540 067571
1609                                     :ITEM 211
1610 003542 043233 066544 071730      .WORD  EM211,DH211,DT211,DF211
1611 003550 067655
1612                                     :ITEM 212
1613 003552 043267 063620 071746      .WORD  EM212,DH212,DT212,DF212
1614 003560 067655
1615                                     :ITEM 213
1616 003562 043321 063620 071746      .WORD  EM213,DH213,DT213,DF213
1617 003570 067655
1618
1619
1620                                     .SBTTI  ACT11 HOOKS
1621
1622                                     ;:*****
1623                                     :HOOKS REQUIRED BY ACT11
1624                                     $SVPC=          :SAVE PC
1625                                     =46
1626 000046 034770          $ENDAD          :;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .SEOP
1627                                     =52
1628 000052 000000          .WORD 0          :;2)SET LOC.52 TO ZERO
1629                                     =6SVPC         :; RESTORE PC
1630                                     .SETTL  APT PARAMETER BLOCK
1631
1632                                     ;:*****
1633                                     :SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
    
```

```

1634      ;:*****
1635      @03572      .SX=      ;:SAVE CUPRENT LOCATION
1636      @00024      .=24      ;:SET POWER FAIL TO POINT TO START OF PROGRAM
1637      @00024 @000200      200      ;:FOR APT START UP
1638      @00044      .=44      ;:POINT TO APT INDIRECT ADDRESS PNTR.
1639      @00044 @03572      $APTHDR ;:POINT TO APT HEADER BLOCK
1640      @03572      .=.SX      ;:RESET LOCATION COUNTER
1641      ;:*****
1642      ;:SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
1643      ;:INTERFACE SPEC.
1644
1645      @03572      $APTHD:
1646      @03572 @00000      $HIRTS: WORD 0      ;:TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
1647      @03574 @01316      $MBADP: WORD $MAIL ;:ADDRESS OF APT MAILBOX (BITS 0-15)
1648      @03576 @00010      $TSTM:  WORD 10     ;:RUN TIM OF LONGEST TEST
1649      @03600 @00040      $PASTM: WORD 40     ;:RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
1650      @03602 @00000      $UNITM: WORD 0      ;:ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
1651      @03604 @00052      .WORD  SETEND=$MAIL/2 ;:LENGTH MAILBOX-ETABLE(WORDS)
1652
1653
1654      @03606      START:
1655      .SBTT: INITIALIZE THE COMMON TAGS
1656      ;:CLEAR THE COMMON TAGS (SCMTAG) AREA
1657      @03606 @12706 @01100      MOV      #SCMTAG,R6      ;:FIRST LOCATION TO BE CLEARED
1658      @03612 @05026      CLR      (R6)+          ;:CLEAR MEMORY LOCATION
1659      @03614 @22706 @01140      CMP      #SWP,R6 ;:DONE?
1660      @03620 @01374      BNE      -=6          ;:LOOP BACK IF NO
1661      @03622 @12706 @01100      MOV      #STACK,SP     ;:SETUP THE STACK POINTER
1662
1663      ;:INITIALIZE A FEW VECTORS
1664      @03626 @12737 @35050 @00020      MOV      #SSCOPE,@#IOTVEC ;:IOT VECTOR FOR SCOPE ROUTINE
1665      @03634 @12737 @00340 @00022      MOV      #340,@#IOTVEC+2 ;:LEVEL 7
1666      @03642 @12737 @35330 @00030      MOV      #SEPPOR,@#EMTVEC ;:EMT VECTOR FOR ERROR ROUTINE
1667      @03650 @12737 @00340 @00032      MOV      #340,@#EMTVEC+2 ;:LEVEL 7
1668      @03656 @12737 @37276 @00034      MOV      #STPAP,@#TRAPVEC ;:TRAP VECTOR FOR TRAP CALLS
1669      @03664 @12737 @00340 @00036      MOV      #340,@#TRAPVEC+2;:LEVEL 7
1670      @03672 @12737 @37362 @00024      MOV      #SPWRDN,@#PWRVEC ;:POWER FAILURE VECTOR
1671      @03700 @12737 @00340 @00026      MOV      #340,@#PWRVEC+2 ;:LEVEL 7
1672      @03706 @16767 @30700 @30670      MOV      $ENDCT,$EOPCT ;:SETUP END-OF-PROGRAM COUNTER
1673      @03714 @05067 175362      CLR      $TIMES        ;:INITIALIZE NUMBER OF ITERATIONS
1674      @03720 @05067 175360      CLR      $ESCAPE      ;:CLEAR THE ESCAPE ON ERROR ADDRESS
1675      @03724 112767 @00001 175163      MOVB    #1,$EPMAX     ;:ALLOW ONE ERROR PER TEST
1676
1677      ;:INITIALIZE THE "T-BIT" TRAP VECTOR. THEN LOAD LOCATION "$RTRN", IN
1678      ;:THE "END-OF-PASS" (SEOP) ROUTINE, WITH A "PTI" OR "RTT".
1679      @03732 @12737 @35034 @00014      MOV      #RTRN,@#TBITVEC ;:SET "T" BIT VECTOR TO RTRN
1680      @03740 @12737 @00340 @00016      MOV      #340,@#TBITVEC+2 ;:LEVEL 7
1681      @03746 @12767 @00002 @31060      MOV      #RTI,$RTRN     ;:SET RTRN TO A RTI
1682      @03754 @12737 @004002 @00010      MOV      #65S,@#PESVEC  ;:TRY TO DO A RTT
1683      @03762 @05046      CLR      -(SP)         ;:DUMMY RS
1684      @03764 @12746 @03772      MOV      #64S,-(SP)    ;:AND PC
1685      @03770 @00006      RTT                     ;:TRY THE RTT
1686      @03772 @12767 @00006 @31034 64S:  MOV      #PTT,$RTRN     ;:RTT IS LEGAL--SET RTRN TO A RTT
1687      @04000 @00402      RR      66S          ;:
1688      @04002 @62706 @00010 65S:  ADD      #10,SP        ;:RTT ILLEGAL--CLEAN OFF THE STACK
1689      @04006 @12737 @00012 @00010 66S:  MOV      #RESVEC+2,@#RESVEC ;:RESTORE TRAP CATCHER
1690      @04014 @05067 @31022      CLR      $TBIT        ;:CLEAR "T" BIT SWITCH
1691      @04020 @12767 @00402 175060      MOV      #.,$LPADR     ;:INITIALIZE THE LOOP ADDRESS FOR SCOPE
    
```



```

1690 004026 012767 004026 175054      MOV    #..SLPEPR      ;; SETUP THE ERROR LOOP ADDRESS
1691                                     ;; SIZE FOR A HARDWARE SWITCH REGISTER. IF NOT FOUND OR IT IS
1692                                     ;; EQUAL TO A "-1", SETUP FOR A SOFTWARE SWITCH REGISTER.
1693 004034 013746 000004                MOV    @#ERRVEC, -(SP) ;; SAVE ERROR VECTOR
1694 004040 012737 004074 000004        MOV    #67s, @#ERRVEC ;; SET UP ERROR VECTOR
1695 004046 012767 177570 175064        MOV    #DSWR, SWR     ;; SETUP FOR A HARDWARE SWICH REGISTER
1696 004054 012767 177570 175060        MOV    #DDISP, DISPLAY ;; AND A HARDWARE DISPLAY REGISTER
1697 004062 022777 177777 175050        CMP    #-1, @SWR     ;; TRY TO REFERENCE HARDWARE SWR
1698 004070 001012                        BNE    69s           ;; BRANCH IF NO TIMEOUT TRAP OCCURRED
1699                                     ;; AND THE HARDWARE SWR IS NOT = -1
1700 004072 000403                        BR     68s           ;; BRANCH IF NO TIMEOUT
1701 004074 012716 004102 67s:          MOV    #68s, (SP)    ;; SET UP FOR TRAP RETURN
1702 004100 000002                        RTI
1703 004102 012767 000176 175030 68s:   MOV    #SWREG, SWR   ;; POINT TO SOFTWARE SWR
1704 004110 012767 000174 175024        MOV    #DISPREG, DISPLAY
1705 004116 012637 000004 69s:          MOV    (SP)+, @#ERRVEC ;; RESTORE ERROR VECTOR
1706
1707 004122 005067 175176                  CLR    $PASS         ;; CLEAR PASS COUNT
1708 004126 132767 000200 175203          BITB   #APTSIZE, $ENVM ;; TEST USER SIZE UNDER APT
1709 004134 001403                        BEQ    70s           ;; YES, USE NON-APT SWITCH
1710 004136 012767 001340 174774          MOV    #SSWREG, SWR  ;; NO, USE APT SWITCH REGISTER
1711 004144 70s:
1712                                     .SBTTL  TYPE PROGRAM NAME
1713                                     ;; TYPE THE NAME OF THE PROGRAM IF FIRST PASS
1714 004144 005227 177777                  INC    #-1           ;; FIRST TIME?
1715 004150 001055                        BNE    71s           ;; BRANCH IF NO
1716 004152 022737 034770 000042          CMP    #SENDAD, @#42 ;; ACT=11?
1717 004160 001451                        BEQ    71s           ;; BRANCH IF YES
1718 004162 104401 004230                  TYPE   ,72s         ;; TYPE ASCIZ STRING
1719                                     .SBTTL  GET VALUE FOR SOFTWARE SWITCH REGISTER
1720 004166 005737 000042                  TST    @#42         ;; ARE WE RUNNING UNDER XXDP/ACT?
1721 004172 001012                        BNE    73s           ;; BRANCH IF YES
1722 004174 126727 175136 000001          CMPB   $ENV, #1     ;; ARE WE RUNNING UNDER APT?
1723 004202 001406                        BEQ    73s           ;; BRANCH IF YES
1724 004204 026727 174730 000176          CMP    SWR, #SWREG  ;; SOFTWARE SWITCH REG SELECTED?
1725 004212 001005                        BNE    74s           ;; BRANCH IF NO
1726 004214 104405                        GTSWR                ;; GET SOFT-SWR SETTINGS
1727 004216 000403                        BR     74s
1728 004220 112767 000001 174706 73s:   MOVB   #1, $AUTOB   ;; SET AUTO-MODE INDICATOR
1729 004226 74s:
1730 004226 000426                        BR     71s           ;; GET OVER THE ASCIZ
1731                                     ;; 72s: .ASCIZ <CRLF>*DEFFPA, FP11-A 11/34 FPP DIAGNOSTIC PART 1* <CRLF>
1732 004304 71s:
1733
1734 004304      LOOP:
1735
1736
1737
1738
1739
1740                                     ;;*****
1741                                     ;;*TEST 1 ----- LDERS, STEPS AND DATA PATHS TEST
1742                                     ;;*
1743                                     ;;*THIS IS A TEST OF THE LDERS (LOAD FLOATING POINT STATUS) AND STEPS
1744                                     ;;*(STORE FLOATING POINT STATUS) INSTRUCTIONS. A COUNT PATTERN IS GENERATED
1745                                     ;;*AND RUN THROUGH THE FLOATING POINT STATUS REGISTER.

```

FORM 1413 PRINTED IN U.S.A.

```

1746      ;*THIS WILL TEST THE 16-BIT TRI STATE BUS WHICH CONNECTS THE CPU
1747      ;*WITH THE FPP AND ALSO RUNS INTERNALLY WITHIN THE FPP. ONLY DM0 AND
1748      ;*SM0 ARE USED.
1749      ;*NOTE THAT A MASK MUST BE USED BECAUSE SOME OF THE FPS BITS CANNOT
1750      ;*BE SET.
1751      ;*
1752      ;*ONLY THE FIRST FIVE ERRORS WILL BE REPORTED INDIVIDUALLY.
1753      ;*THIS IS TO PREVENT LOCKING OUT THE COMPLETION OF THE TEST BECAUSE
1754      ;*OF VIRTUALLY ENDLESS NUMBER OF ERRORS. ONLY FIVE INDIVIDUAL ERRORS
1755      ;*WILL BE REPORTED THEN THE TEST WILL BE COMPLETED AND AN ERROR
1756      ;*SUMMARY GIVEN (SEE NOTE BELOW).
1757      ;*
1758      ;*NOTE THAT THIS TEST KEEPS A DYNAMIC RECORD OF THE LOGICAL 'AND' AND 'OR'
1759      ;*OF THE FAILING DATA PATTERNS. THESE CAN BE VERY USEFUL IN DETERMINING
1760      ;*STUCK BITS. IF THE USER HAS THE INHIBIT ERROR TYPE OUT SWITCH (SWR13)
1761      ;*OFF, THEN THE USER WILL RECIEVE EACH INDIVIDUAL ERROR MESSAGE PLUS
1762      ;*AN ERROR SUMMARY AT THE END OF THE TEST. INHIBITING ERROR PRINT OUT
1763      ;*WILL INHIBIT ERROR SUMMARY PRINT OUT, EXCEPT IN THE CASE DESCRIBED BELOW.
1764      ;*TO GET JUST THE ERROR SUMMARY WITH NO INDIVIDUAL ERROR REPORTS,
1765      ;*SET SWITCH REGISTER BIT13 AND SWITCH REGISTER BIT7 BOTH ON.
1766      ;*
1767      ;*****
1768      TST1:  SCOPE
1769      @04304 @00004          CLR    @#AERFLG
1770      @04306 @05037 @04560  LPERF          ;SET UP THE LOOP ON ERROR ADDRESS.
1771      @04312 134413          MOV    #-1,R0          ;INITIALIZE THE COUNT PATTERN.
1772      @04314 @12700 177777  MOV    #AERR1,@#FPVECT ;SET UP FOR UNABLE TO DECODE
1773      @04320 @12737 @04562 @00244  MOV    #AERR2,@#10     ;FPP INSTRUCTION TRAP TO 244 OR 10.
1774      @04326 @12737 @04574 @00010  CLR    R2             ;R2 IS THE 'AND' OF BAD DATA.
1775      @04334 @05002          COM    R2
1776      @04336 @05102          CLR    R3             ;R3 IS THE 'OR' OF BAD DATA.
1777      @04340 @05003          MOV    #AERR3,@#ERRVECT ;IF EITHER INSTRUCTION
1778      @04342 @12737 @04626 @00004  ;FAILS TO GO THROUGH THE
1779      ;CORRECT SRC OR DST MODE AN
1780      ;ODD ADDRESS TRAP WILL OCCUR.
1781      A1:
1782      @04350 @10004          A11:   MOV    R0,R4
1783      @04352 @42704 @30020  BIC    #30020,R4
1784      @04356 170104          LDFPS  R4             ;TEST INSTRUCTION.
1785
1786      @04360 @12701 177777  MOV    #-1,R1
1787      @04364 170201          A12:   STEPS  R1          ;TEST INSTRUCTION.
1788      @04366 @12737 @40200 @00244  MOV    #FPSPUR,@#FPVECT ;SET UP FOR UNEXPECTED TRAPS.
1789      @04374 @10004          MOV    R0,R4          ;MASK OFF UNSETTABLE BITS.
1790      @04376 @42704 @30020  BIC    #30020,R4
1791      @04402 @12737 @40232 @00004  MOV    #CPSPUR,@#ERRVECT
1792      @04410 @12737 @40250 @00010  MOV    #CPTW0,@#10
1793      @04416 @20401          CMP    R4,R1          ;COMPARE DATA EXPECTED WITH
1794      ;THE DATA READ.
1795      @04420 @01002          BNE    A3             ;IF NOT EQUAL GO REPORT ERROR.
1796
1797      @04422 @77026          A2:   SOB    R0,A1      ;OTHERWISE DECREMENT COUNT PATTERN
1798      @04424 @00425          BP     A5             ;UNTIL IT IS ZERO.
1799
1800      @04426 @05237 @04560  A3:   INC    @#AERFLG    ;RECORD ERROR.
1801      @04432 @50003          BJS   R0,R3          ;COMPUTE 'OR' OF FAILING PATTERNS.
  
```

```

1802 004434 010005      MOV      R0,R5      :COMPUTE 'AND' OF FAILING PATTERNS.
1803 004436 005105      COM      R5
1804 004440 040502      BIC      R5,R2
1805
1806 004442 022737 000005 004560      CMP      #5,0#AERFLG :SEE IF MORE THAN 5 ERRORS HAVE
1807 004450 103412      BLO      A05        :OCCURRED. BR IF YES.
1808
1809
1810 004452 012737 004350 001236      MOV      #A1,0#STMP2
1811 004460 010037 001240      MOV      R0,0#STMP3
1812 004464 010137 001242      MOV      R1,0#STMP4
1813 004470 010437 001244      MOV      R4,0#STMP5
1814 004474 104001      A4:      ERROR      1
1815
1816 004476 000751      A05:     BR          A2        :CONTINUE TESTING.
1817
1818 004500 005737 004560      A5:      TST      0#AERFLG :SEE IF ANY ERRORS OCCURRED.
1819 004504 001471      BEQ      ADONE       :IF NOT GO TO NEXT TEST.
1820 004506 032777 020000 174424      BIT      #SW13,0SWP  :OTHERWISE SEE IF A SUMMARY
1821 004514 001404      BEQ      A6          :SHOULD BE TYPED.
1822 004516 032777 000200 174414      BIT      #SW7,0SWR
1823 004524 001461      BEQ      ADONE
1824
1825 004526      A6:      :TYPE ERROR SUMMARY.
1826 004526 010237 001236      MOV      R2,0#STMP2
1827 004532 010337 001240      MOV      R3,0#STMP3
1828 004536 012737 004552 001116      MOV      #A7,0#SERRPC
1829 004544 112737 000002 001114      MOVB    #2,0#SITEMB
1830 004552 004737 037546      A7:      JSR      PC,0#ERTYPE
1831 004556 000444      BR      ADONE
1832
1833 004560 000000      AERFLG: .WORD      0
1834
1835      :UNABLE TO DECODE FPP INSTRUCTION. TRAPPED TO 244.
1836 004562 011637 001236      AERR1:   MOV      (SP),0#STMP2 :SAVE PC OF TRAP.
1837 004566 022626      CMP      (SP)+,(SP)+
1838 004570 104010      1S:     ERROR      10
1839 004572 000436      BR      ADONE
1840
1841      :UNABLE TO DECODE INSTRUCTION. TRAPPED TO 10.
1842 004574 021627 004352      AERP2:   CMP      (SP),#A11+2 :DID TRAP OCCUR OF FPP INSTRUCTION?
1843 004600 001405      BEQ      1S
1844 004602 021627 004366      CMP      (SP),#A12+2
1845 004606 001402      BEQ      1S
1846 004610 000137 040250      JMP      0#CPTWO :IF NOT FPP INSTRUCTION THEN
1847
1848
1849 004614 011637 001236      1S:     MOV      (SP),0#STMP2 :OTHERWISE REPORT IF DECIDE ERROR.
1850 004620 022626      CMP      (SP)+,(SP)+
1851 004622 104011      2S:     ERROR      11
1852 004624 000421      BR      ADONE
1853
1854      :TRAP TO 4 HANDLER:
1855 004626 021627 004352      AERP3:   CMP      (SP),#A11+2 :DID THE TRAP OCCUR ON THE
1856 004632 001405      BEQ      1S :LDFPS INSTRUCTION?
1857 004634 021627 004366      CMP      (SP),#A12+2 :OR THE STEPS INSTRUCTION?

```

```

1858 004640 001407          BEQ      2S
1859 004642 000137 040232    JMP      @#CPSPUR          :IF NEITHER THEN REPORT
1860                               :UNEXPECTED TRAP TO 4.
1861
1862 004646 011637 001236    1S:     MOV      (SP),@#STMP2
1863 004652 022626          CMP      (SP)+,(SP)+
1864 004654 104014          15S:    ERROR    14
1865 004656 000404          BR       ADONE
1866
1867 004660 011637 001236    2S:     MOV      (SP),@#STMP2
1868 004664 022626          CMP      (SP)+,(SP)+
1869 004666 104015          25S:    ERROR    15
1870
1871 004670          ADONE:
1872 004670 104412          RSETUP          :GO INITIALIZE THE FPS AND STACK; AND
1873                               :SEE IF THE USER HAS EXPRESSED
1874                               :THE DESIRE TO CHANGE THE SOFTWARE
1875                               :VIRTUAL CONSOLE SWITCH REGISTER (HAS
1876                               :THE USER TYPED CONTROL G?).
1877
1878
1879                               :*****
1880                               :*TEST 2          CFCC TEST
1881                               :*
1882                               :*THIS IS A TEST OF THE COPY CONDITION CODES INSTRUCTION, CFCC.
1883                               :*
1884                               :*****
1885 004672 000004          TST2:    SCOPE
1886 004674 104413          IPERR
1887 004676 012700 000017    MOV      #17,R0          :SET UP THE LOOP ON ERROR ADDRESS,
1888                               :R0 CONTAINS TO TEST PATTERN.
1889
1889 004702          B1:
1890 004702 170100          LDFPS   P0          :LOAD THE TEST PATTERN
1891
1892 004704          B2:
1893 004704 170000          CFCC          :COPY CONDITION CODES.
1894
1895 004706 013703 177776    MOV      @#PSW,R3          :SEE IF PATTERN TRANSFERED.
1896 004712 042703 177760    BIC      #177760,R3
1897 004716 020003          CMP      R0,R3
1898 004720 001002          BNE     BERR
1899
1900 004722 077011          B3:     SOB      R0,B1
1901 004724 000422          BR       BDONE
1902
1903 004726          BERR:
1904 004726 170201          STFPS   P1          :WAS FPS MODIFIED BY CFCC?
1905 004730 012737 004704 001236    MOV      #R2,@#STMP2
1906
1907 004736 020001          CMP      R0,R1
1908 004740 001006          BNE     BERR1
1909
1910 004742 010337 001240    MOV      R3,@#STMP3
1911 004746 010037 001242    MOV      R0,@#STMP4
1912 004752 104003          1S:     ERROR    3
1913 004754 000762          BR       B3
  
```

```

1914
1915 004756 BEPR1:
1916 004756 010037 001240 MOV R0,#STMP3
1917 004762 010137 001242 MOV R1,#STMP4
1918 004766 104004 1S: ERROR 4
1919 004770 000754 BF B3
1920
1921 004772 BDONE:
1922 004772 104412 RSETUP :GO INITIALIZE THE FPS AND STACK; AND
1923 :SEE IF THE USER HAS EXPRESSED
1924 :THE DESIRE TO CHANGE THE SOFTWARE
1925 :VIRTUAL CONSOLE SWITCH REGISTER (HAS
1926 :THE USER TYPED CONTROL G?).
1927
1928 :*****
1929 :*TEST 3 SETF, SETD, SETI AND SETL TEST
1930 :*
1931 :*THIS IS A TEST OF THE SETF, SETD, SETI AND SETL INSTRUCTIONS.
1932 :*EACH INSTRUCTION IS EXECUTED WITH THE FPS CONTAINING
1933 :*ALL ONES AND ALSO WITH THE FPS CLEAR. THE RESULT OF EACH
1934 :* SITUATION IS CHECKED.
1935 :*
1936 :*****
1937 004774 000004 TST3: SCOPE
1938 004776 104413 LPERR :SET UP THE LOOP ON ERROR ADDRESS.
1939 005000 012737 000760 001244 MOV #760,#STMP5
1940 005006 012737 000202 001250 C1: MOV #202,#STMP7
1941 005014 012737 041352 001252 MOV #SETF1,#STMP10
1942 005022 005000 CLR R0
1943
1944 005024 170100 LDFPS R0 :CLEAR THE FPS.
1945 005026 012737 005034 001236 MOV #C15,#STMP2
1946
1947 005034 170001 C15: SETF :TEST INSTRUCTION.
1948
1949 005036 170201 STEPS R1 :GET RESULT.
1950 005040 005002 CLR R2
1951 005042 020201 CMP R2,R1 :DID AN ERROR OCCUR?
1952 005044 001402 BEQ 1S
1953 005046 004737 005432 JSR PC,#CERR1
1954
1955 005052 1S:
1956 005052 104413 LPERR :SET UP THE LOOP ON ERROR ADDRESS.
1957 005054 012700 147757 C2: MOV #147757,R0
1958
1959 005060 170100 LDFPS R0 :PUT 147757 IS FPS
1960 005062 012737 005070 001236 MOV #C25,#STMP2
1961 005070 170001 C25: SETF :CLEAR FD BIT.
1962
1963 005072 170201 STEPS R1 :GET RESULT
1964 005074 012702 147557 MOV #147557,R2
1965 005100 020102 CMP R1,R2 :RESULT CORRECT.
1966 005102 001402 BEQ 1S
1967 005104 004737 005530 JSR PC,#CERR2
1968
1969 005110 1S:

```

```

1970 005110 104413 LPERR :SET UP THE LOOP ON ERROR ADDRESS.
1971 005112 012737 000203 001250 C3: MOV #203,0#STMP7
1972 005120 012737 041360 001252 MOV #SETD1,0#STMP10
1973 005126 012700 147757 MOV #147757,R0
1974
1975 005132 170100 LDFPS R0 :LOAD 147757 INTO FPS.
1976 005134 012737 005142 001236 MOV #C35,0#STMP2
1977 005142 170011 C35: SETD :SETD FD BIT.
1978
1979 005144 170201 STEPS R1
1980 005146 012702 147757 MOV #147757,R2
1981 005152 020102 CMP R1,R2 :RESULT CORRECT?
1982 005154 001402 BEQ 1$
1983 005156 004737 005530 JSR PC,0#CERR2
1984
1985 005162 1$:
1986 005162 104413 LPERR :SET UP THE LOOP ON ERROR ADDRESS.
1987 005164 005000 C4: CLR R0
1988 005166 170100 LDFPS R0 :CLEAR FPS.
1989 005170 012737 005176 001236 MOV #C45,0#STMP2
1990
1991 005176 170011 C45: SETD :SET FD BIT.
1992
1993 005200 170201 STEPS R1 :GET RESULT.
1994 005202 012702 000200 MOV #200,R2
1995 005206 020102 CMP R1,R2 :RESULT CORRECT?
1996 005210 001402 BEQ 1$
1997 005212 004737 005432 JSR PC,0#CERR1
1998
1999 005216 1$:
2000 005216 104413 LPERR :SET UP THE LOOP ON ERROR ADDRESS.
2001 005220 012737 000204 001250 C5: MOV #204,0#STMP7
2002 005226 012737 041366 001252 MOV #SETI1,0#STMP10
2003 005234 005000 CLR R0
2004
2005 005236 170100 LDFPS R0 :CLEAR FPS
2006 005240 012737 005246 001236 MOV #C55,0#STMP2
2007
2008 005246 170002 C55: SETI :CLEAR FL BIT.
2009
2010 005250 170201 STEPS R1 :GET RESULT.
2011 005252 005002 CLR R2
2012 005254 020201 CMP R2,R1 :RESULT CORRECT?
2013 005256 001402 BEQ 1$
2014 005260 004737 005432 JSR PC,0#CERR1
2015
2016 005264 1$:
2017 005264 104413 LPERR :SET UP THE LOOP ON ERROR ADDRESS.
2018 005266 012700 147757 C6: MOV #147757,R0
2019 005272 170100 LDFPS R0 :PUT 147757 INTO FPS
2020 005274 012737 005302 001236 MOV #C65,0#STMP2
2021
2022 005302 170002 C65: SETI :CLEAR FL BIT.
2023
2024 005304 170201 STEPS R1 :GET THE RESULT.
2025 005306 012702 147657 MOV #147657,R2

```

```

2026 005312 020102          CMP      R1,R2          :RESULT CORRECT?
2027 005314 001402          BEQ      1$
2028 005316 004737 005530      JSR      PC,0#CERR2
2029
2030 005322          1$:
2031 005322 104413          LPERR          :SET UP THE LOOP ON ERROR ADDRESS.
2032 005324 012737 000205 001250 C7:      MOV      #205,0#STMP7
2033 005332 012737 041374 001252      MOV      #SETL1,0#STMP10
2034 005340 012700 147757      MOV      #147757,R0
2035 005344 170100          LDFPS      R0          :SET FPS TO 147757.
2036 005346 012737 005354 001236      MOV      #C75,0#STMP2
2037
2038 005354 170012          C75:      SETL          :SET FL BIT.
2039
2040 005356 170201          STFPS      R1          :GET THE RESULT.
2041 005360 012702 147757      MOV      #147757,R2
2042 005364 020102          CMP      R1,R2          :RESULT CORRECT?
2043 005366 001402          BEQ      1$
2044 005370 004737 005530      JSR      PC,0#CERR2
2045
2046 005374          1$:
2047 005374 104413          LPERR          :SET UP THE LOOP ON ERROR ADDRESS.
2048 005376 005000          C8:      CLR      R0
2049 005400 170100          LDFPS      R0          :CLEAR FPS.
2050 005402 012737 005410 001236      MOV      #C85,0#STMP2
2051
2052 005410 170012          C85:      SETL          :SET FL BIT.
2053
2054 005412 170201          STFPS      R1
2055 005414 012702 000100      MOV      #100,R2
2056 005420 020102          CMP      R1,R2          :RESULT CORRECT.
2057 005422 001402          BEQ      1$
2058 005424 004737 005432      JSR      PC,0#CERR1
2059
2060 005430 000522          1$:      BR      CDONE
2061
2062          :THESE ARE ERROR ANALYSIS ROUTINES:
2063 005432 010103          CERR1:      MOV      R1,R3
2064 005434 032703 177477      BIT      #177477,R3      :ARE ANY OTHER BITS SET?
2065 005440 001401          BEQ      2$
2066 005442 000503          1$:      BR      CERR4
2067
2068 005444 022703 000300          2$:      CMP      #300,R3      :ARE BOTH FD AND FL SET?
2069 005450 001774          BEQ      1$
2070 005452 032703 000300      BIT      #300,R3      :ARE THEY BOTH CLEAR?
2071 005456 001771          BEQ      1$
2072
2073 005460 032703 000200          BIT      #200,R3      :IS FD SET?
2074 005464 001407          BEQ      3$
2075 005466 012737 041360 001254      MOV      #SETD1,0#STMP11
2076 005474 012737 000203 001246      MOV      #203,0#STMP6
2077 005502 000452          BR      CERR3
2078
2079 005504 032703 000100          3$:      BIT      #100,R3      :IS FL SET
2080 005510 001754          BEQ      1$
2081 005512 012737 041374 001254      MOV      #SETL1,0#STMP11
  
```

```

2082 005520 012737 000205 001246      MOV      #205,0#STMP6
2083 005526 000440      BR       CERR3
2084
2085 005530 010103      CERR2:  MOV      R1,R3
2086 005532 005103      COM      R3
2087
2088 005534 032703 177477      BIT      #177477,R3      :ARE ANY OTHER BITS SET?
2089 005540 001401      BEQ      26
2090 005542 000443      1$:     BR       CERR4
2091
2092 005544 032703 000300      2$:     BIT      #300,R3      :ARE BOTH FD AND FL SET?
2093 005550 001774      BEQ      16
2094 005552 032701 000300      BIT      #300,R1      :ARE THEY BOTH CLEAR?
2095 005556 001771      BEQ      16
2096
2097 005560 032701 000200      BIT      #200,R1      :IS FD CLEAR?
2098 005564 001007      BNE      36
2099 005566 012737 041352 001254      MOV      #SETF1,0#STMP11
2100 005574 012737 000202 001246      MOV      #202,0#STMP6
2101 005602 000412      BR       CERR3
2102
2103 005604 032701 000100      3$:     BIT      #100,R1
2104 005610 001354      BNE      16      :IS FL CLEAR.
2105 005612 012737 041366 001254      MOV      #SETI1,0#STMP11
2106 005620 012737 000204 001246      MOV      #204,0#STMP6
2107 005626 000400      BR       CERR3
2108
2109      ;REPORT THE ERRORS:
2110 005630      CERR3:
2111 005630 010137 001240      MOV      R1,0#STMP3
2112 005634 010237 001242      MOV      R2,0#STMP4
2113 005640 012637 005674      MOV      (SP)+,0#CPC
2114 005644 104012      1$:     ERROR  12
2115 005646 000177 000022      JMP      @CPC
2116
2117 005652      CERR4:
2118 005652 010137 001240      MOV      R1,0#STMP3
2119 005656 010237 001242      MOV      R2,0#STMP4
2120 005662 012637 005674      MOV      (SP)+,0#CPC
2121 005666 104013      1$:     ERROR  13
2122 005670 000177 000000      JMP      @CPC
2123
2124 005674 000000      CPC:     .WORD  0
2125
2126 005676      CDONE:
2127 005676 104412      RSETUP      ;GO INITIALIZE THE FPS AND STACK; AND
2128      ;SEE IF THE USER HAS EXPRESSED
2129      ;THE DESIRE TO CHANGE THE SOFTWARE
2130      ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
2131      ;THE USER TYPED CONTROL G?).
2132
2133
2134      ;*****
2135      ;*TEST 4      ILLEGAL FPP OP CODES AND STST TEST
2136      ;*
2137      ;*THIS IS A TEST OF THE FPP OPERATION CODES:

```



```

2138      :*          170003
2139      :*          170004
2140      :*          :
2141      :*          170010
2142      :*          170013
2143      :*          170014
2144      :*          :
2145      :*          170077
2146      :*THESE ARE ILLEGAL INSTRUCTIONS AND (WITH INTERRUPTS ENABLED)
2147      :*SHOULD CAUSE A TRAP TO 244.
2148      :*ALSO TESTED HERE IS THE INSTRUCTION:
2149      :*          STST    R1
2150      :*WHICH SHOULD PUT THE FEC CODE 2 IN R1, AFTER ANY OF THE ABOVE
2151      :*OP CODES IS EXECUTED.
2152      :*
2153      :*****
2154  005700  000004      TST4:  SCOPE
2155  005702  104413      LPERR          ;SET UP THE LOOP ON ERROR ADDRESS.
2156  005704  012705  170003      MOV    #170003,R5      ;INITIAL OP CODE.
2157  005710  012737  006114  000004      MOV    #DERR2,#FRRVECT
2158  005716  012737  006020  000244      MOV    #DERR1,#FRRVECT
2159
2160  005724  005000      D1:    CLR    R0
2161  005726  170100      LDFPS  R0          ;CLEAR FPS.
2162  005730  005002      CLR    R2
2163  005732  010537  005750      MOV    R5,#D2          ;SET UP THE ILLEGAL INSTRUCTION.
2164  005736  010537  001244      MOV    R5,#STMP5
2165  005742  012737  005750  001236      MOV    #D2,#STMP2
2166  005750  000000      D2:    .WORD  0
2167  005752  170000      D3:    CFCC
2168  005754  005202      INC    R2
2169  005756  005202      D4:    INC    R2
2170
2171  005760  170201      STEPS  R1          ;REPORT FAILURE. DID NOT TRAP.
2172  005762  010137  001240      MOV    R1,#STMP3
2173  005766  104016      1S:   ERROR  16
2174
2175  005770  022705  170010      D5:    CMP    #170010,R5      ;COMPUTE NEXT OP CODE
2176  005774  001003      BNE   D6
2177  005776  012705  170013      MOV    #170013,R5
2178  006002  000750      BR    D1
2179
2180  006004  022705  170077      D6:    CMP    #170077,R5
2181  006010  001001      BNE   D7
2182  006012  000452      BR    DDONE
2183  006014  005205      D7:    INC    R5
2184  006016  000742      BR    D1
2185
2186  006020  022716  005752      DERR1: CMP    #D3,(SP)      ;DID TRAP OCCUR ON TEST INSTRUCTION?
2187  006024  001402      BEQ   1S
2188  006026  000137  040200      JNE   #FRRSPUR
2189
2190  006032  022626      1S:   CMP    (SP)+,(SP)+
2191  006034  170201      STEPS  R1          ;GET THE FPS AND SEE IF IT IS
2192  006036  022701  100000      CMP    #100000,R1      ;SET CORRECTLY.
2193  006042  001406      BEQ   3S
  
```

```

2194
2195 006044 012737 100000 001240      MOV      #100000,0#STMP3
2196 006052 010137 001242      MOV      R1,0#STMP4
2197 006056 104017      26:     ERROR      17
2198
2199 006060 012704 000001      36:     MOV      #1,R4
2200 006064 170304      D8:     STST      R4
2201
2202
2203
2204 006066 022704 000002      CMP      #2,R4
2205 006072 001001      BNE     D9
2206 006074 000735      BR      D5
2207
2208 006076      D9:
2209 006076 012737 006064 001240      MOV      #D8,0#STMP3
2210 006101 010437 001242      MOV      R4,0#STMP4
2211 006110 104020      16:     ERROR      20
2212 006112 000726      BR      D5
2213
2214 006114 022716 006066      DERR2:  CMP      #D8+2,(SP)
2215 006120 001402      BFG     D10
2216 006122 000137 040232      JMP     @#CPSPUR
2217
2218 006126      D10:
2219 006126 011637 001236      MOV      (SP),0#STMP2
2220 006132 022626      CMP     (SP)+,(SP)+
2221 006134 104021      16:     ERROR      21
2222 006136 000714      BR      D5
2223
2224 006140      DDONE:
2225 006140 104412      RSETUP
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240 006142 000004      TST5:   SCOPE
2241 006144 104413      LPERR
2242 006146 012737 006250 000244      MOV      #EFRR2,0#FPVTECT
2243
2244 006154 012700 040000      E1:     MOV      #40000,R0
2245 006160 170100      LDFPS  R0
2246 006162 012737 006170 001236      MOV      #E3,0#STMP2
2247 006170      E2:
2248 006170 170020      E3:     .WORD 170020
2249 006172 170000      E4:     CFCC

```

:GET THE FEC CODE, NOTE THAT  
 :IF THE DESTINATION MODE IS  
 :IMPROPERLY DECODED AN ODD  
 :ADDRESS TRAP TO 4 SHOULD OCCUR,  
 :WAS FEC CORRECT?

:REPORT STST FAILURE

:DID THE TRAP OCCUR ON THE  
 :STST INSTRUCTION?

:GO INITIALIZE THE FRS AND STACK; AND  
 :SEE IF THE USER HAS EXPRESSED  
 :THE DESIRE TO CHANGE THE SOFTWARE  
 :VIRTUAL CONSOLE SWITCH REGISTER (HAS  
 :THE USER TYPED CONTROL G?)

\*\*\*\*\*  
 :\*TEST 5 FID, INTERRUPT DISABLE, BIT TEST  
 :\*  
 :\*THIS IS A TEST OF FRS BIT 14 (FID) OR FLOATING INTERRUPT DISABLE.  
 :\*AN ILLEGAL INSTRUCTION IS EXECUTED WITH FID=1. NO INTERRUPT SHOULD  
 :\*OCCUR.  
 :\*

:SET UP THE LOOP ON ERROR ADDRESS,  
 :SETUP FOR THE INTERRUPT.

:SET FID.

:ILLEGAL FPP INSTRUCTION.



```

2306 006304          F1:
2307 006304 104413          LPERR          :SET UP THE LOOP ON ERROR ADDRESS.
2308 006306 012737 006356 001236      MOV          #F3,#STMP2
2309 006314 005000          CLR          R0
2310 006316 170100          LDFPS       R0
2311 006320 170011          SETD
2312 006322 012701 010110          MOV          #FDAT0,R1          :SET UP THE LOAD DATA.
2313 006326 012702 010154          MOV          #FXDAT0,R2
2314 006332 012703 000010          MOV          #10,R3
2315
2316 006336 012221          F2:          MOV          (R2)+,(R1)+
2317 006340 077302          SOB          R3,F2
2318
2319 006342 012700 010120          MOV          #FDAT4,R0          :SETUP R0 FOR THE LDD (R0),AC0.
2320 006346 012737 007574 000004          MOV          #FERR20,#ERRVECT :IF THE SRC FLOWS FAIL THEN
2321                                          :AN ODD ADDRESS MAY OCCUR.
2322 006354 005003          CLR          R3
2323
2324 006356 172410          F3:          LDD          (R0),AC0
2325 006360 005203          F4:          INC          R3
2326 006362 005203          INC          R3
2327
2328 006364 020027 010120          CMP          R0,#FDAT4          :WAS R0 AFFECTED?
2329 006370 001402          BEQ          F5
2330 006372 000137 006740          JMP          @#FERR1
2331
2332 006376 020327 000002          F5:          CMP          R3,#2          :SEE IF THE PC WAS ADVERSELY
2333 006402 001402          BEQ          1$          :AFFECTED DURING THE INSTRUCTION.
2334 006404 000137 007036          JMP          @#FERR2
2335
2336 006410 012701 010110          1$:          MOV          #FDAT0,R1          :MAKE SURE THE SOURCE DATA WAS
2337 006414 012702 010154          MOV          #FXDAT0,R2          :NOT AFFECTED.
2338 006420 012703 000010          MOV          #10,R3
2339 006424 022122          2$:          CMP          (R1)+,(R2)+
2340 006426 001402          BEQ          3$
2341 006430 000137 006702          JMP          @#FERR0
2342 006434 077305          3$:          SOB          R3,2$
2343
2344 006436 170201          STFPS       R1          :MAKE SURE THE FPS IS CORRECT.
2345 006440 022701 000200          CMP          #200,R1
2346 006444 001402          BEQ          F6
2347 006446 000137 007554          JMP          @#FERR11
2348
2349 006452          F6:
2350 006452 104413          LPERR          :SET UP THE LOOP ON ERROR ADDRESS.
2351 006454 012737 006516 001236      MOV          #F10,#STMP2
2352
2353 006462 012703 177777          MOV          #-1,R3
2354 006466 012704 000010          MOV          #10,R4
2355 006472 012705 010132          MOV          #FDAT00,R5          :SET UP THE OUTPUT DATA BUFFER.
2356 006476 010325          F7:          MOV          R3,(R5)+
2357 006500 077402          SOB          R4,F7
2358
2359 006502 012700 010142          MOV          #FDAT04,R0          :SET UP R0 FOR DST MODE 1 REG 0.
2360 006506 012737 007742 000004          MOV          #FERR25,#ERRVECT :IF THE DST FLOWS FAIL AN ODD
2361                                          :ADDRESS COULD OCCUR.

```

2362	006514	005003		CLR	R3	
2363						
2364	006516	174010		F10: STD	AC0,(R0)	:TEST INSTRUCTION.
2365	006520	005203		F11: INC	R3	
2366	006522	005203		INC	R3	
2367						
2368	006524	020027	010142	CMP	R0,#FDAT04	:WAS R0 MODIFIED?
2369	006530	001402		BEQ	F12	
2370	006532	000137	007076	JMP	0#FERR3	
2371						
2372	006536	020327	000002	F12: CMP	R3,#2	:WAS THE PC AFFECTED CORRECTLY?
2373	006542	001402		BEQ	F135	
2374	006544	000167	000320	JMP	FERR4	
2375						
2376	006550	012701	010132	F135: MOV	#FDAT00,R1	
2377	006554	012702	010154	MOV	#FXDAT0,R2	
2378						
2379	006560	022122		CMP	(R1)+,(R2)+	:SEE IF THE DATA WAS OUTPUT
2380	006562	001402		BEQ	F13	:TO THE TARGET AREA CORRECTLY.
2381	006564	000137	007174	JMP	0#FERR5	
2382						
2383	006570	022122		F13: CMP	(R1)+,(R2)+	
2384	006572	001402		BEQ	F14	
2385	006574	000137	007174	JMP	0#FERR5	
2386						
2387	006600	022122		F14: CMP	(R1)+,(R2)+	
2388	006602	001402		BEQ	F15	
2389	006604	000137	007174	JMP	0#FERR5	
2390						
2391	006610	022122		F15: CMP	(R1)+,(R2)+	
2392	006612	001402		BEQ	F16	
2393	006614	000137	007174	JMP	0#FERR5	
2394						
2395	006620	022122		F16: CMP	(R1)+,(R2)+	
2396	006622	001402		BEQ	F17	
2397	006624	000137	007520	JMP	0#FERR10	
2398						
2399	006630	022122		F17: CMP	(R1)+,(R2)+	
2400	006632	001402		BEQ	F20	
2401	006634	000137	007230	JMP	0#FERR6	
2402						
2403	006640	022122		F20: CMP	(R1)+,(R2)+	
2404	006642	001402		BEQ	F21	
2405	006644	000167	000514	JMP	FERR7	
2406						
2407	006650	022122		F21: CMP	(R1)+,(R2)+	
2408	006652	001402		BEQ	F22	
2409	006654	000137	007520	JMP	0#FERR10	
2410						
2411	006660	005001		F22: CLR	R1	
2412	006662	170201		STFPS	R1	:MAKE SURE FPS IS CORRECT.
2413	006664	022701	000200	CMP	#200,R1	
2414	006670	001402		BEQ	F23	
2415	006672	000137	007554	JMP	0#FERR11	
2416	006676	000137	010174	F23: JMP	0#FDONE	
2417						

2418	006702				FERR0:			:SOURCE DATA AFFECTED BY
2419	006702	012737	010154	001240	MOV	#FXDAT0,0#STMP3		:THE LDD INSTRUCTION.
2420	006710	012737	010166	001242	MOV	#FXDAT0+12,0#STMP4		
2421	006716	012737	010110	001244	MOV	#FDAT0,0#STMP5		
2422	006724	012737	010122	001246	MOV	#FDAT0+12,0#STMP6		
2423	006732	104025			1S:	ERR0R	25	
2424	006734	000137	010174		JMP	0#FDONE		
2425								
2426	006740	012737	010120	001242	FERR1:	MOV	#FDAT14,0#STMP4	:ESRC FLOWS FAILURE.
2427	006746	010037	001240		MOV	R0,0#STMP3		
2428	006752	012737	000762	001244	MOV	#762,0#STMP5		
2429	006760	012737	000321	001250	MOV	#321,0#STMP7		
2430								
2431	006766	022700	010110		CMP	#FDAT10,R0		:FSPC MODE 4?
2432	006772	001004			BNE	1S		
2433	006774	012737	000324	001246	MOV	#324,0#STMP6		
2434	007002	000412			BR	4S		
2435								
2436	007004	022700	010130		1S:	CMP	#FDAT14+10,R0	:FSPC MODE 2?
2437	007010	001004			BNE	2S		
2438	007012	012737	000322	001246	MOV	#322,0#STMP6		
2439	007020	000403			BR	4S		
2440								
2441	007022				2S:			
2442	007022	104027			3S:	ERR0R	27	
2443	007024	000137	010174		JMP	0#FDONE		
2444								
2445	007030				4S:			
2446	007030	104026			5S:	ERR0R	26	
2447	007032	000137	010174		JMP	0#FDONE		
2448								
2449	007036	012701	006360		FERR2:	MOV	#F4,R1	:THE PC WAS INCORRECTLY AFFECTED
2450								:DURING THE INSTRUCTION.
2451	007042	010137	001242		FERR2:	MOV	R1,0#STMP4	
2452	007046	162701	000004		SUB	#4,R1		
2453	007052	006303			ASL	R3		
2454	007054	060301			ADD	R3,R1		
2455	007056	010137	001240		MOV	R1,0#STMP3		
2456	007062	104030			1S:	ERR0R	30	
2457	007064	000137	010174		JMP	0#FDONE		
2458								
2459	007070	012701	006520		FERR4:	MOV	#F11,R1	
2460	007074	000762			BR	FERR2		
2461								
2462	007076	012737	010142	001242	FERR3:	MOV	#FDAT04,0#STMP4	:FAILURE IN THE FDST FLOWS.
2463	007104	010037	001240		MOV	R0,0#STMP3		
2464	007110	012737	000527	001244	MOV	#527,0#STMP5		
2465	007116	012737	000641	001250	MOV	#641,0#STMP7		
2466								
2467	007124	022700	010132		CMP	#FDAT00,R0		:DST MODE 4?
2468	007130	001004			BNE	1S		
2469	007132	012737	000644	001246	MOV	#644,0#STMP6		
2470	007140	000412			BR	4S		
2471								
2472	007142	022700	010152		1S:	CMP	#FDAT04+10,R0	:DST MODE 2?
2473	007146	001004			BNE	2S		

2474	007150	012737	000642	001246	MOV	#642,0#STMP6	
2475	007156	000403			BR	4S	
2476							
2477	007160				2S:		
2478	007160	104032			3S:	ERROR	32
2479	007162	000137	010174		JMP	0#FDONE	
2480							
2481	007166				4S:		
2482	007166	104031			5S:	ERROR	31
2483	007170	000137	010174		JMP	0#FDONE	
2484							
2485	007174				FERR5:		:FAILURE OF STD.
2486	007174	010037	001240		MOV	R0,0#STMP3	
2487	007200	012737	010132	001242	MOV	#FDAT00,0#STMP4	
2488	007206	012737	010150	001244	MOV	#FDAT07,0#STMP5	
2489	007214	012737	010154	001246	MOV	#FXDAT0,0#STMP6	
2490	007222	104033			1S:	ERROR	33
2491	007224	000137	010174		JMP	0#FDONE	
2492							
2493	007230	012701	010144		FERR6:	MOV #FDAT05,R1	:DID (BUT GR7) FAIL IN THE FDST
2494	007234	012702	177777		MOV	#-1,R2	:FLOWS?
2495	007240	012703	000003		MOV	#3,R3	
2496	007244	020221			1S:	CMP R2,(R1)+	
2497	007246	001017			BNE	5S	
2498	007250	077303			S0B	R3,1S	
2499							
2500							:REPORT FAILURE OF (BUT GR7) IN
2501	007252	010037	001240		MOV	R0,0#STMP3	:THE FDST FLOWS.
2502	007256	012737	000412	001244	MOV	#412,0#STMP5	
2503	007264	012737	000147	001246	MOV	#147,0#STMP6	
2504	007272	012737	000145	001250	MOV	#145,0#STMP7	
2505	007300	104034			2S:	ERROR	34
2506	007302	000137	010174		JMP	0#FDONE	
2507							
2508	007306	012701	010144		5S:	MOV #FDAT05,R1	:DID (BUT GR7) FAIL IN THE SRC FLOWS?
2509	007312	012703	000003		MOV	#3,R3	
2510	007316	005721			6S:	TST (R1)+	
2511	007320	001402			BEG	7S	
2512	007322	000137	007520		JMP	0#FERR10	
2513	007326	077305			7S:	S0B R3,6S	
2514							
2515							:REPORT FAILURE OF (BUT GR7) IN
2516	007330	010037	001240		MOV	R0,0#STMP3	:THE FSRC FLOWS.
2517	007334	012737	000207	001244	MOV	#207,0#STMP5	
2518	007342	012737	000176	001246	MOV	#176,0#STMP6	
2519	007350	012737	000174	001250	MOV	#174,0#STMP7	
2520							
2521	007356	104035			10S:	ERROR	35
2522	007360	000137	010174		JMP	0#FDONE	
2523							
2524	007364	012701	010146		FERR7:	MOV #FDAT06,R1	:DID (BUT FD) FAIL IN THE FDST FLOWS?
2525	007370	012702	177777		MOV	#-1,R2	
2526	007374	012703	000002		MOV	#2,R3	
2527	007400	020221			1S:	CMP R2,(R1)+	
2528	007402	001017			BNE	5S	
2529	007404	077303			S0B	R3,1S	

```

2530
2531                                     ;REPORT FAILURE OF (BUT FD) IN THE
2532 007406 010037 001240      MOV      R0,#STMP3      ;FDST FLOWS.
2533 007412 012737 000707 001244      MOV      #707,#STMP5
2534 007420 012737 000244 001246      MOV      #244,#STMP6
2535 007426 012737 000245 001250      MOV      #245,#STMP7
2536 007434 104036          2s:      ERROR    36
2537 007436 000137 010174          JMP      @#FDONE
2538
2539 007442 012701 010146          5s:      MOV      #FDAT06,R1      ;DID (BUT FD) FAIL IN THE FSRC FLOWS?
2540 007446 012703 000002          MOV      #2,R3
2541 007452 005721          6s:      TST      (R1)+
2542 007454 001402          BEQ      7s
2543 007456 000137 007520          JMP      @#FERR10
2544 007462 077305          7s:      SOB      R3,6s
2545
2546                                     ;REPORT FAILURE OF (BUT FD) IN THE
2547 007464 010037 001240      MOV      R0,#STMP3      ;FSRC FLOWS.
2548 007470 012737 000441 001244      MOV      #441,#STMP5
2549 007476 012737 000076 001246      MOV      #76,#STMP6
2550 007504 012737 000077 001250      MOV      #77,#STMP7
2551 007512 104037          10s:     ERROR    37
2552 007514 000137 010174          JMP      @#FDONE
2553
2554                                     FERR10:      ;REPORT DATA ERROR.
2555 007520 010037 001240      MOV      R0,#STMP3
2556 007524 012737 010142 001242      MOV      #FDAT04,#STMP4
2557 007532 012737 010150 001244      MOV      #FDAT07,#STMP5
2558 007540 012737 010164 001246      MOV      #FXDAT4,#STMP6
2559 007546 104040          1s:      ERROR    40
2560 007550 000137 010174          JMP      @#FDONE
2561
2562                                     FERR11:      ;REPORT BAD FPS.
2563 007554 010137 001240      MOV      R1,#STMP3
2564 007560 012737 000200 001242      MOV      #200,#STMP4
2565 007566 104041          1s:      ERROR    41
2566 007570 000137 010174          JMP      @#FDONE
2567
2568 007574 012737 040411 001264 FERR20: MOV      #NULL,#STMP15      ;THE EXECUTION OF THE LDD
2569 007602 025037 001252      CLR      @#STMP10      ;CAUSED A TRAP TO 4, BECAUSE
2570 007606 011637 001236      MOV      (SP),#STMP2      ;A FSRC FLOW FAILURE RESULTED
2571 007612 012737 010120 001240      MOV      #FDAT14,#STMP3      ;IN AN ODD ADDRESS.
2572 007620 012737 000321 001250      MOV      #321,#STMP7
2573 007626 012737 000762 001244      MOV      #762,#STMP5
2574
2575 007634 021627 006362          CMP      (SP),#F4+2      ;SEE IF FSRC MODE 6 OR 7 WAS
2576 007640 001424          BEQ      FERR21      ;EXECUTED.
2577
2578 007642 020027 010116          CMP      R0,#FDAT13      ;FSRC MODE 5?
2579 007646 001006          BNE      2s
2580
2581                                     ;REPORT FSRC FLOW FAILURE TO
2582 007650 012737 000325 001246      MOV      #325,#STMP6      ;MODE 5.
2583 007656 022626          CMP      (SP)+,(SP)+
2584 007660 104042          1s:      ERROR    42
2585 007662 000544          BR      FDONE
  
```

PRINTED IN U.S.A.



```

2586
2587 007664 020027 010122 2S:  CMP  R0,#FDAT15 :FSRC MODE 3?
2588 027670 001402      BEQ  3S
2589 027672 000137 040232      JMP  @#CPSPUR
2590
2591 007676      3S:      :REPORT FSRC FLOW FAILURE TO
2592 007676 012737 000323 001246      MOV  #323,@#STMP6 :MODE 3.
2593 007704 022626      CMP  (SP)+,(SP)+
2594 007706 104042      4S:      EPROR 42
2595 007710 000531      BR   FDONE
2596
2597 007712 022626      FERR21: CMP  (SP)+,(SP)+ :REPORT FSRC FLOW FAILURE TO
2598      :MODE 6 OR MODE 7.
2599 007714 012737 042156 001264      MOV  #MS16,@#STMP15
2600 007722 012737 000326 001246      MOV  #326,@#STMP6
2601 007730 012737 000327 001252      MOV  #327,@#STMP10
2602 007736 104042      1S:      ERROR 42
2603 007740 000515      BR   FDONE
2604
2605 007742 012737 040411 001264      FERR25: MOV  #NULL,@#STMP15 :THE EXECUTION OF THE STD INSTRUCTION
2606 007750 005037 001252      CLR  @#STMP10 :TRAPPED TO 4, BECAUSE A FAILURE
2607 007754 012737 010142 001240      MOV  #FDAT04,@#STMP3 :IN THE FDST FLOWS RESULTED
2608 007762 011637 001236      MOV  (SP),@#STMP2 :IN AN ODD ADDRESS.
2609 007766 012737 000527 001244      MOV  #527,@#STMP5
2610 007774 012737 000641 001250      MOV  #641,@#STMP7
2611
2612 010002 021627 006520      CMP  (SP),#F10+2 :FLOW FAILURE TO FDST MODE 6 OR 7?
2613 010006 001424      BEQ  FERR26
2614
2615 010010 020027 010140      CMP  R0,#FDAT03 :DID FDST FLOW FAIL TO MODE 5?
2616 010014 001006      BNE  2S
2617
2618      :REPORT FLOW FAILURE TO FDST
2619 010016 012737 000645 001246      MOV  #645,@#STMP6 :MODE 5.
2620 010024 022626      CMP  (SP)+,(SP)+
2621 010026 104043      1S:      EPROR 43
2622 010030 000461      BR   FDONE
2623
2624 010032 020027 010144      2S:      CMP  R0,#FDAT05 :DID FDST FLOW FAIL TO MODE 3?
2625 010036 001402      BEQ  3S
2626 010040 000137 040232      JMP  @#CPSPUR
2627
2628 010044      3S:      :REPORT FDST FLOW FAILED TO MODE 3.
2629 010044 012737 000643 001246      MOV  #643,@#STMP6
2630 010052 022626      CMP  (SP)+,(SP)+
2631 010054 104043      4S:      ERROR 43
2632 010056 000446      BR   FDONE
2633
2634 010060      FERR26: :REPORT FDST FLOW FAILURE TO MODE
2635 010060 012737 042156 001264      MOV  #MS16,@#STMP15 :6 OR MODE 7.
2636 010066 012737 000646 001246      MOV  #646,@#STMP6
2637 010074 012737 000647 001252      MOV  #647,@#STMP10
2638 010102 022626      CMP  (SP)+,(SP)+
2639 010104 104043      1S:      ERROR 43
2640 010106 000432      BR   FDONE
2641

```

```

2642 010110 177777      FDATA0: -1
2643 010112 177777      FDATA1: -1
2644 010114 177777      FDATA2: -1
2645 010116 177777      FDATA3: -1
2646 010120 177777      FDATA4: -1
2647 010122 177777      FDATA5: -1
2648 010124 177777      FDATA6: -1
2649 010126 177777      FDATA7: -1
2650 010130 177777      -1
2651 010132 177777      FDATA0: -1
2652 010134 177777      FDATA1: -1
2653 010136 177777      FDATA2: -1
2654 010140 177777      FDATA3: -1
2655 010142 177777      FDATA4: -1
2656 010144 177777      FDATA5: -1
2657 010146 177777      FDATA6: -1
2658 010150 177777      FDATA7: -1
2659 010152 177777      -1
2660 010154 177777      FXDATA0: -1
2661 010156 177777      FXDATA1: -1
2662 010160 177777      FXDATA2: -1
2663 010162 177777      FXDATA3: -1
2664 010164 052525      FXDATA4: 052525
2665 010166 031463      FXDATA5: 031463
2666 010170 007417      FXDATA6: 007417
2667 010172 000477      FXDATA7: 000477
2668
2669
2670 010174      FDCONE:
2671 010174 104412      RSETUP      ;GO INITIALIZE THE FPS AND STACK; AND
2672                                     ;SEE IF THE USER HAS EXPRESSED
2673                                     ;THE DESIRE TO CHANGE THE SOFTWARE
2674                                     ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
2675                                     ;THE USER TYPED CONTROL G?).
2676
2677
2678                                     ;:*****
2679                                     ;*TEST 7      FSRC MODE 0 TEST
2680                                     ;*
2681                                     ;*THIS IS A TEST OF FSRC MODE ZERO USING THE LDD AND LDF INSTRUCTIONS.
2682                                     ;*
2683                                     ;:*****
2684 010176 000004      TST7:      SCOPE
2685 010200 104413      LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
2686
2687 010202      I1:
2688 010202 170011      SETD      ;SET FD.
2689 010204 012700 011026      MOV      #IDATIO,R0
2690 010210 012701 010776      MOV      #IPATIO,R1
2691 010214 012702 000004      MOV      #4,R2
2692 010220 012120      I2:      MOV      (R1)+,(R0)+      ;SET UP THE INPUT DATA BUFFER.
2693 010222 077202      SOB      R2,I2
2694
2695 010224 012703 011026      MOV      #IDATIO,R0      ;LOAD AC1
2696 010230 172510      LDD      (R0),AC1
2697

```

```

2698 010232 012700 011006      MOV      #IPAT20,R0      ;LOAD AC0
2699 010236 172410      LDD      (R0),AC0
2700
2701 010240 012701 000001      MOV      #1,R1          ;IN CASE THE FSRC FLOWS FAIL
2702 010244 012737 010576 000004      MOV      #IERR0,#ERRVECT ;AN ODD ADDRESS TRAP TO 4 MAY OCCUR.
2703 010252 012737 010266 001236      MOV      #I3,#STMP2
2704 010260 012737 042636 001240      MOV      #MS35,#STMP3
2705 010266 172401      I3:     LDD      AC1,AC0      ;TEST INSTRUCTION.
2706 010270 000240      I4:     NOP
2707 010272 000240      I5:     NOP
2708
2709 010274 012700 011016      MOV      #IDAT00,R0
2710 010300 174010      STD      AC0,(R0)      ;GET AC0, THE RESULTS.
2711
2712 010302 012700 011016      MOV      #IDAT00,R0      ;SEE IF DATA IS CORRECT.
2713 010306 012701 011026      MOV      #IDATI0,R1
2714 010312 012702 000004      MOV      #4,R2
2715 010316 022021      I6:     CMP      (R0)+,(R1)+
2716 010320 001424      BEQ      I105
2717
2718 010322 012700 011022      MOV      #IDAT02,R0      ;SEE IF (BUT FD) FAILED.
2719 010326 012702 000002      MOV      #2,R2
2720 010332 005720      I7:     TST      (R0)+
2721 010334 001413      BEQ      I10
2722
2723 010336 012700 011022      MOV      #IDAT02,R0
2724 010342 012702 000002      MOV      #2,R2
2725 010346 022720 177777      I8:     CMP      #-1,(R0)+
2726 010352 001402      BEQ      2S
2727 010354 000137 010660      JMP      @#IERR1
2728 010360 077206      2S:     SOB      R2,1S
2729 010362 000401      BR      I106
2730 010364 077216      I10:    SOB      R2,I7
2731 010366 000137 010700      I106:   JMP      @#IERR2
2732
2733 010372 077227      I105:   SOB      R2,I6
2734
2735      ;NOW TEST THE LOAD INSTRUCTION WITH FSRC MODE ZERO AND FD CLEAR.
2736
2737      I11:
2738 010374 104413      LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
2739
2740 010376 012700 010776      I12:    MOV      #IPAT10,R0
2741 010402 012701 011026      MOV      #IDATI0,R1
2742 010406 012702 000004      MOV      #4,R2
2743 010412 012021      I13:    MOV      (R0)+,(R1)+
2744 010414 077202      SOB      R2,I13
2745
2746 010416 012700 011026      MOV      #IDATI0,R0      ;SET UP AC1
2747 010422 172510      LDD      (R0),AC1
2748
2749 010424 012700 011006      MOV      #IPAT20,R0      ;SET UP AC0
2750 010430 172410      LDD      (R0),AC0
2751
2752 010432 012701 000001      MOV      #1,R1
2753 010436 012737 010454 001236      MOV      #I14,#STMP2
  
```

```

2754 010144 012737 042643 001240      MOV      #MS36,0#STMP3
2755 010152 170001                      SETF                      :CLEAR FD.
2756
2757 010454 172401                      I14:   LDF      AC1,AC0      :TEST INSTRUCTION.
2758 010456 000240                      I15:   NOP
2759 010460 000240                      I16:   NOP
2760
2761 010462 170200                      STEPS  R0                  :SEE IF FPS IS STILL CLEAR.
2762 010464 022700 000004      CMP      #4,R0
2763 010470 001402      BEQ      I17
2764 010472 000137 010752      JMP      0#IERR3
2765
2766 010476                      I17:                      :RESET TO DOUBLE MODE.
2767 010476 170011      SETD
2768
2769 010500 012700 011016      MOV      #IDAT00,R0
2770 010504 174010      STD      AC0,(R0)          :GET AC0
2771
2772 010506 012737 177777 011032      MOV      #-1,0#IDATI2
2773 010514 012737 177777 011034      MOV      #-1,0#IDATI3
2774 010522 012700 011016      MOV      #IDAT00,R0
2775 010526 012701 011026      MOV      #IDATI0,R1
2776 010532 012702 000004      MOV      #4,R2
2777 010536 022021                      I20:   CMP      (R0)+,(R1)+  :SEE IF AC0 WAS CORRECT.
2778 010540 001414      BEQ      I23
2779
2780 010542 023737 011022 011002      CMP      0#IDAT02,0#IPAT12  :DID (BUT FD) FAIL?
2781 010550 001402      BEQ      I22
2782 010552 000137 010660                      I21:   JMP      0#IERR1
2783 010556 023737 011024 011004      I22:   CMP      0#IDAT03,0#IPAT13
2784 010564 001372      BNE      I21
2785 010566 000137 010726      JMP      0#IERR4
2786
2787 010572 077217                      I23:   SOB      R2,I20
2788
2789 010574 000520      BR      IDONE              :NO ERRORS.
2790
2791                      :IF AN ODD ADDRESS TRAP OCCURS COME HERE TO ANALYZE THE FSRC FAILURE.
2792 010576 022716 010270      IERR0:  CMP      #I4,(SP)          :MAKE SURE THE TRAP OCCURRED
2793 010602 001413      BEQ      1$                :ON THE INSTRUCTION BEING TESTED.
2794 010604 022716 010272      CMP      #I5,(SP)
2795 010610 001410      BEQ      1$
2796 010612 022716 010456      CMP      #I15,(SP)
2797 010616 001405      BEQ      1$
2798 010620 022716 010460      CMP      #I16,(SP)
2799 010624 001402      BEQ      1$
2800 010626 000137 040232      JMP      0#CPSPUR
2801
2802 010632 011637 001236                      1$:   MOV      (SP),0#STMP2      :REPORT FAILURE.
2803 010636 012737 000627 001240      MOV      #627,0#STMP3
2804 010644 012737 000320 001242      MOV      #320,0#STMP4
2805 010652 022626      CMP      (SP)+,(SP)+
2806 010654 104047                      2$:   ERROR  47
2807 010656 000467      BR      IDONE
2808
2809                      :REPORT DATA ERROR.
    
```

```

2810 010660 IERR1:
2811 010660 012737 011026 001242 MOV #IDATI0,0#STMP4
2812 010666 012737 011016 001244 MOV #IDATO0,0#STMP5
2813 010674 104051 IS: ERROR 51
2814 010676 000457 BR IDONE
2815
2816 ;REPORT FAILURE OF (BUT FD)
2817 010700 012737 000153 001244 IERR2: MOV #153,0#STMP5
2818 010706 012737 000434 001246 MOV #434,0#STMP6
2819 010714 012737 000435 001250 MOV #435,0#STMP7
2820 010722 IERR25:
2821 010722 104050 IS: ERROR 50
2822 010724 000444 BR IDONE
2823 010726 012737 000153 001244 IERR4: MOV #153,0#STMP5
2824 010734 012737 000435 001246 MOV #435,0#STMP6
2825 010742 012737 000434 001250 MOV #434,0#STMP7
2826 010750 000764 BR IEPR25
2827
2828 ;REPORT INCORRECT FPS AFTER LOAD INSTRUCTION.
2829 010752 IERR3:
2830 010752 012737 010454 001236 MOV #I14,0#STMP2
2831 010760 010037 001240 MOV R0,0#STMP3
2832 010764 012737 000004 001242 MOV #4,0#STMP4
2833 010772 104041 IS: ERROR 41
2834 010774 000420 BR IDONE
2835
2836
2837 010776 000004 IPATI0: 0
2838 011000 170360 IPATI1: 170360
2839 011002 016161 IPATI2: 016161
2840 011004 052525 IPATI3: 052525
2841
2842 011006 177777 IPAT20: -1
2843 011010 177777 IPAT21: -1
2844 011012 177777 IPAT22: -1
2845 011014 177777 IPAT23: -1
2846
2847 011016 000000 IDATO0: 0
2848 011020 000000 IDATO1: 0
2849 011022 000000 IDATO2: 0
2850 011024 000000 IDATO3: 0
2851
2852 011026 000000 IDATI0: 0
2853 011030 000000 IDATI1: 0
2854 011032 000000 IDATI2: 0
2855 011034 000000 IDATI3: 0
2856
2857 IDONE:
2858 011036 104412 RSETUP ;GO INITIALIZE THE FPS AND STACK; AND
2859 ;SEE IF THE USER HAS EXPRESSED
2860 ;THE DESIRE TO CHANGE THE SOFTWARE
2861 ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
2862 ;THE USER TYPED CONTROL G?).
2863
2864
2865 ;*****
  
```

MICRO BUSINESS FORMS, INC. HQ  
 FORM 1413  
 PRINTED IN U.S.A. 77

```

2866          :*TEST 10          FDST MODE 0 TEST
2867          :*
2868          :*THIS IS A TEST OF THE STORE INSTRUCTIONS, STD AND STF, WITH FDST MODE 0.
2869          :*
2870          :*****
2871 011040 000004          TST10: SCOPE
2872 011042          T1:
2873 011042 104413          LPEPR          :SET UP THE LOOP ON ERROR ADDRESS.
2874 011044 170011          SETD          :SET FD
2875 011046 012700 011604          MOV #TPAT10,R0
2876 011052 012701 011634          MOV #TDAT10,R1
2877 011056 012702 000004          MOV #4,R2
2878 011062 012021          T2: MOV (R0)+,(R1)+          :SET UP THE INPUT DATA BUFFER.
2879 011064 077202          SOB R2,T2
2880
2881 011066 012700 011634          MOV #TDAT10,R0          :LOAD AC0
2882 011072 172410          LDD (R0),AC0
2883
2884 011074 012700 011614          MOV #TPAT20,R0          :LOAD AC1
2885 011100 172510          LDD (R0),AC1
2886
2887 011102 012701 000001          MOV #1,R1          :IF THE (BUT FDST) FORK FAILS
2888 011106 012737 011412 000004          MOV #TERR0,#FRRVECT          :AN ODD ADDRESS TRAP COULD RESULT.
2889 011114 012737 011130 001236          MOV #T3,#STMP2
2890 011122 012737 042636 001240          MOV #MS35,#STMP3
2891 011130 174001          T3: STD AC0,AC1
2892 011132 000240          T4: NOP
2893 011134 000240          T5: NOP
2894
2895 011136 012700 011624          MOV #TDAT00,R0
2896 011142 174110          STD AC1,(R0)          :GET THE DATA.
2897
2898 011144 012703 011624          MOV #TDAT00,R3          :SEE IF THE DATA IS CORRECT.
2899 011150 012704 011634          MOV #TDAT10,R4
2900 011154 012705 000004          MOV #4,R5
2901 011160 022324          T6: CMP (R3)+,(R4)+
2902 011162 001413          BEQ T105
2903
2904 011164 012703 011630          MOV #TDAT02,R3          :DID (BUT FD) FAIL?
2905 011170 012705 000002          MOV #2,R5
2906 011174 005723          T7: TST (R3)+
2907 011176 001402          BEQ T10
2908 011200 000137 011474          JMP #TERR1
2909 011204 077505          T10: SOB R5,T7
2910 011206 000137 011514          JMP #TERR2
2911
2912 011212 077516          T105: SOB R5,T6
2913
2914          :NOW TEST THE STF AC0,AC1 INSTRUCTION.
2915
2916 011214          T11:
2917 011214 104413          LPEPR          :SET UP THE LOOP ON ERROR ADDRESS.
2918
2919 011216 012700 011604          T12: MOV #TPAT10,R0          :SET UP THE INPUT DATA BUFFER.
2920 011222 012701 011634          MOV #TDAT10,R1
2921 011226 012702 000004          MOV #4,R2
  
```

```

2922 011232 012021          T13:  MOV    (R0)+,(R1)+
2923 011234 077202          SOB    R2,T13
2924
2925 011236 012700 011634    MOV    #TDAT0,R0          :SET UP AC0
2926 011242 172410          LDD    (R0),AC0
2927
2928 011244 012700 011614    MOV    #TPAT0,R0          :SET UP AC1
2929 011250 172510          LDD    (R0),AC1
2930
2931 011252 012701 000001    MOV    #1,R1
2932 011256 012737 011274 001236  MOV    #T14,#STMP2
2933 011264 012737 042643 001240  MOV    #MS36,#STMP3
2934 011272 170001          SETF                     :CLEAR FD
2935 011274 174001          T14:  STF    AC0,AC1
2936 011276 000240          T15:  NOP
2937 011300 000240          T16:  NOP
2938
2939 011302 005000          CLP    R0
2940 011304 170200          STFPS  R0          :SEE IF FPS IS CLEAR.
2941 011306 022700 000010    CMP    #10,R0
2942 011312 001401          BEQ    T17
2943 011314 000521          BR     TERR3
2944
2945 011316          T17:
2946 011316 170011          SETD                     :SET FD.
2947
2948 011320 012700 011624    MOV    #TDAT00,R0
2949 011324 174110          STD    AC1,(R0)        :PICK UP AC1.
2950
2951 011326 012737 177777 011640  MOV    #-1,#TDAT12
2952 011334 012737 177777 011642  MOV    #-1,#TDAT13
2953 011342 012703 011624    MOV    #TDAT00,R3
2954 011346 012704 011634    MOV    #TDAT10,R4
2955 011352 012705 000004    MOV    #4,R5
2956 011356 022324          T20:  CMP    (R3)+,(R4)+    :WAS THE DATA TRANSFERRED CORRECTLY?
2957 011360 001412          BEQ    T23
2958
2959 011362 023737 011630 011610  CMP    #TDAT02,#TPAT12   :DID (BUT FD) FAIL.
2960 011370 001401          BEQ    T22
2961 011372 000440          T21:  BR     TERR1
2962 011374 023737 011632 011612  T22:  CMP    #TDAT03,#TPAT13
2963 011402 001373          BNE    T21
2964 011404 000456          BR     TERR4
2965
2966 011406 077515          T23:  SOB    R5,T20
2967 011410 000515          BR     TDONE
2968
2969
2970          :TRAP HERE THROUGH VECTOR 4 IF AN ODD ADDRESS OCCURS.
2971 011412 022716 011132  TERR0:  CMP    #T4,(SP)          :MAKE SURE THE TRAP WAS ON
2972 011416 001413          BEQ    1$              :AN INSTRUCTION BEING TESTED.
2973 011420 022716 011134          CMP    #T5,(SP)
2974 011424 001410          BEQ    1$
2975 011426 022716 011276          CMP    #T15,(SP)
2976 011432 001405          BEQ    1$
2977 011434 022716 011300          CMP    #T16,(SP)
    
```

DFFPA.P11 01-NOV-76 21:03 T10 FDST MODE 0 TEST

```

2978 011440 001402          BEQ      1$
2979 011442 000137 040232    JMP      @#CRSPUR
2980
2981 011446 011637 001236    1$:     MOV      (SP),@#STMP2
2982 011452 022626          CMP      (SP)+,(SP)+
2983 011454 012737 000527 001240    MOV      #527,@#STMP3
2984 011462 012737 000640 001242    MOV      #640,@#STMP4
2985 011470 104121          2$:     ERROR    121
2986 011472 000464          BR       TDONE
2987
2988          ;REPORT DATA FAILURE.
2989 011474          TERR1:
2990 011474 012737 011634 001242    MOV      #TDAT10,@#STMP4
2991 011502 012737 011624 001244    MOV      #TDAT00,@#STMP5
2992 011510 104123          1$:     ERROR    123
2993 011512 000454          BR       TDONE
2994
2995          ;REPORT FAILURE OF (BIT FD).
2996 011514 012737 000160 001246    TERR2:  MOV      #160,@#STMP6
2997 011522 012737 000161 001250    MOV      #161,@#STMP7
2998 011530 012737 000640 001244    TERR25: MOV      #640,@#STMP5
2999 011536 104122          1$:     ERROR    122
3000 011540 000441          BR       TDONE
3001 011542 012737 000161 001246    TERR4:  MOV      #161,@#STMP6
3002 011550 012737 000160 001250    MOV      #160,@#STMP7
3003 011556 000764          BR       TERR25
3004
3005          ;REPORT INCORRECT FPS AFTER STORE INSTRUCTION.
3006 011560          TERR3:
3007 011560 012737 011276 001236    MOV      #T15,@#STMP2
3008 011566 010037 001240          MOV      R0,@#STMP3
3009 011572 012737 000010 001242    MOV      #10,@#STMP4
3010 011600 104041          1$:     ERROR    41
3011 011602 000420          BR       TDONE
3012
3013 011604 000000          TPAT10: 0
3014 011606 170360          TPAT11: 170360
3015 011610 016161          TPAT12: 016161
3016 011612 052525          TPAT13: 052525
3017
3018 011614 177777          TPAT20: -1
3019 011616 177777          TPAT21: -1
3020 011620 177777          TPAT22: -1
3021 011622 177777          TPAT23: -1
3022
3023 011624 000000          TDAT00: 0
3024 011626 000000          TDAT01: 0
3025 011630 000000          TDAT02: 0
3026 011632 000000          TDAT03: 0
3027
3028 011634 000000          TDAT10: 0
3029 011636 000000          TDAT11: 0
3030 011640 000000          TDAT12: 0
3031 011642 000000          TDAT13: 0
3032
3033 011644          TDONE:

```

PRINTED IN U.S.A. FORM 1413



```
3034 011644 104412 RSETUP ;GO INITIALIZE THE FPS AND STACK; AND
3035 ;SEE IF THE USER HAS EXPRESSED
3036 ;THE DESIRE TO CHANGE THE SOFTWARE
3037 ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
3038 ;THE USER TYPED CONTROL G?),
3039
3040
3041
3042 :*****
3043 :*TEST 11 ACCUMULATORS DATA PATTERNS TEST
3044 :*
3045 :*THIS IS A TEST OF THE FLOATING POINT PROCESSOR ACCUMULATORS.
3046 :*EACH ACCUMULATOR IS TESTED IN TWO WAYS:
3047 :* 1 TEST PATTERN GENERATED BY FLOATING A ONE ACROSS
3048 :* A FIELD OF ZEROES.
3049 :* 2 TEST PATTERN GENERATED BY FLOATING A ZERO ACROSS
3050 :* A FIELD OF ONES.
3051 :*EACH OF ACCUMULATORS AC0 THROUGH AC5 IS TESTED.
3052 :*
3053 :*NOTE THAT THIS TEST KEEPS A DYNAMIC RECORD OF THE LOGICAL 'AND' AND 'OR'
3054 :*OF THE FAILING DATA PATTERNS. THESE CAN BE VERY USEFUL IN DETERMINING
3055 :*STUCK BITS. IF THE USER HAS THE INHIBIT ERROR TYPE OUT SWITCH (SWR13)
3056 :*OFF, THEN THE USER WILL RECIEVE EACH INDIVIDUAL ERROR MESSAGE PLUS
3057 :*AN ERROR SUMMARY AT THE END OF THE TEST. INHIBITING ERROR PRINT OUT
3058 :*WILL INHIBIT ERROR SUMMARY PRINT OUT, EXCEPT IN THE CASE DESCRIBED BELOW.
3059 :*TO GET JUST THE ERROR SUMMARY WITH NO INDIVIDUAL ERROR REPORTS,
3060 :*SET SWITCH REGISTER BIT13 AND SWITCH REGISTER BIT7 BOTH ON.
3061 :*
3062 :*
3063 :*THE FOLLOWING PROCEDURE IS PRESENTED TO AID THE TROUBLE
3064 :*SHOOTER IN SITUATIONS WHERE AM2901 CHIP ISOLATION IS ATTEMPTED.
3065 :*
3066 :*WARNING: THIS PROCEDURE ASSUMES THAT THE FAULT IS IN ONE OF THE
3067 :*AM2901 CHIPS. THIS ASSUMPTION IS NOT NECESSARILY VALID IN ALL
3068 :*SITUATIONS. IT REMAINS TO BE SEEN WHAT NUMBER OF FAILURES CAN
3069 :*PROBABILISTICALLY ASSOCIATED WITH THEM. NOTE ALSO THAT THIS
3070 :*INFORMATION SHOULD NOT BE TAKEN AS ABSOLUTE, THAT IS
3071 :*THIS INFORMATION IS THE AUTHOR'S SUGGESTION FOR ACHIEVING ISOLATION
3072 :*WHEN CHIP LEVEL REPAIR IS NECESSARY.
3073 :*
3074 :*WHEN THIS TEST HAS FINISHED RUNNING, IF ERRORS HAVE OCCURRED,
3075 :*AN ERROR SUMMARY WILL BE TYPED. THIS SUMMARY WILL CONSIST OF TWO
3076 :*IMPORTANT QUANTITIES:
3077 :* A. FOUR SIXTEEN BIT NUMBERS LABELED THE LOGICAL 'AND' ('*')
3078 :* OF THE FAILING DATA PATTERNS.
3079 :* B. FOUR SIXTEEN BIT NUMBERS LABELED THE LOGICAL 'OR' ('+')
3080 :* OF THE FAILING DATA PATTERNS.
3081 :*
3082 :*A BIT STUCK HIGH IN THE HARDWARE WILL SHOW UP AS A 0 IN THAT
3083 :*BIT POSITION OF THE 'OR' OF THE FAILING DATA PATTERNS.
3084 :*
3085 :*A BIT STUCK LOW IN THE HARDWARE WILL SHOW UP AS A 1 IN THAT BIT
3086 :*POSITION OF THE 'AND' OF THE FAILING DATA PATTERNS.
3087 :*
3088 :*THUS IF A FAILURE OCCURS:
3089 :* A. STUCK HIGHS WILL SHOW AS 0'S IN THE 'OR' PATTERN.
```

```

3090      :*      B.      STUCK LOWS WILL SHOW AS 1'S IN THE 'AND' PATTERN.
3091      :*IF THE FAILURE IS INTERMITTANT THEN THIS PROCEDURE WILL STILL
3092      :*APPLY!!
3093      :*IF THE FAILURE MOVES FROM ONE BIT TO ANOTHER, OR FROM ONE
3094      :*GROUP OF BITS TO ANOTHER GROUP OF BITS THEN THE FAULT WILL
3095      :*PROBABLY NOT SHOW UP IN THE 'AND' OR THE 'OR' PATTERNS, IN THIS
3096      :*CASE THE 'AND' PATTERN WILL BE ALL 0'S AND THE 'OR' PATTERN WILL
3097      :*BE ALL 1'S. WHEN THIS OCCURS SOME OTHER METHOD OF REPAIR MUST
3098      :*BE FOUND (SUCH AS INSPECTION OF EACH INDIVIDUAL ERROR REPORT
3099      :*RATHER THAN USING THE SUMMARY).
3100      :*
3101      :*MAP THE FOLLOWING NOTATION ONTO EACH BIT POSITION IN THE 'AND'
3102      :*AND THE 'OR' PATTERNS WHICH ARE TYPED IN THE ERROR SUMMARY.
3103      :*
3104      :*A15,A14,...A1,A0  B15,B14,...B1,B0  C15,C14,...C1,C0  D15,D14,...D1,D0
3105      :*
3106      :*IN THIS NOTATION A15 THROUGH A0 IS THE FIRST OF THE FOUR 16 BIT
3107      :*OCTAL NUMBERS TYPED, B15 THROUGH B0 IS THE SECOND, ETC.
3108      :*
3109      :*THIS TABLE SHOWS THE CORRESPONDING AM2901 CHIP ('E' NUMBER)
3110      :*WHICH IS RESPONSIBLE FOR EACH BIT POSITION USING THE ABOVE
3111      :*NOTATION. NOTE THAT ECO'S TO THE HARDWARE MIGHT MAKE THIS
3112      :*TABLE OBSOLETE IF IT IS NOT UP DATED. NOTE ALSO THAT THERE ARE
3113      :*FOUR BITS FOR EACH AM2901 CHIP:
3114      :*
3115      :*      BITS      AM2901 CHIP NUMBER
3116      :*      ----      -
3117      :*
3118      :*      A15,A14,A13,A12      E61
3119      :*      A11,A10,A9,A8      E62
3120      :*      A7,A6,A5,A4      E90
3121      :*      A3,A2,A1,A0      E81
3122      :*
3123      :*      B15,B14,B13,B12      E26
3124      :*      B11,B10,B9,B8      E85
3125      :*      B7,B6,B5,B4      E83
3126      :*      B3,B2,B1,B0      E88
3127      :*
3128      :*      C15,C14,C13,C12      E79
3129      :*      C11,C10,C9,C8      E94
3130      :*      C7,C6,C5,C4      E89
3131      :*      C3,C2,C1,C0      E87
3132      :*
3133      :*      D15,D14,D13,D12      E78
3134      :*      D11,D10,D9,D8      E77
3135      :*      D7,D6,D5,D4      E82
3136      :*      D3,D2,D1,D0      E80
3137      :*
3138      :*NOW FIVE IMPORTANT CASES WHICH WILL ARRISE WHEN A FAULTY
3139      :*AM2901 IS PRESENT CAN BE DESCRIBED:
3140      :*
3141      :*1.) IF ONLY ONE BIT OF THE 64 BITS IS INCORRECT THE CHIP INDICATED
3142      :*IN THE ABOVE TABLE IS MOST PROBABLY AT FAULT. BUT IF THAT
3143      :*CHIP IS REPLACED AND THE ERROR PERSISTS THEN SUPPOSE THAT
3144      :*BIT IS,
3145      :*      LN      WHERE 'L' IS A, B, C OR D
    
```

ACCUMULATORS DATA PATTERNS TEST  
 PRINTED IN U.S.A.

```

3146      :*                               ANDN IS 15, 14, ... OR 0
3147      :* THEN IN GENERAL ANY OF THE FOUR CHIPS RESPONSIBLE FOR
3148      :* AN, BN, CN OR DN COULD BE AT FAULT, WITH LN BEING MOST PROBABLE.
3149      :* FOR EXAMPLE IF BIT C12 IS FAULTY, THEN CHIP E79
3150      :* IS THE MOST PROBABLE SOURCE OF THE ERROR. IF REPAIRING
3151      :* THAT CHIP DOES NOT REMOVE THE FAULT THEN TRY EACH OF THE
3152      :* CHIPS ASSOCIATED WITH BITS A12, B12 AND D12 SHOULD BE TRIED
3153      :* WITH EQUAL PROBABILITY OF THE FAULT BEING
3154      :* IN ANY ONE OF THESE OTHER THREE CHIPS, TRY CHIPS E61, E86 AND E78.
3155      :*2.) IF THERE ARE FOUR CONSECUTIVE BITS IN ERROR, FOLLOWING THE
3156      :* PATTERN:
3157      :* LN, LN+1, LN+2 AND LN+3 WHERE 'L' IS A, B, C
3158      :* OR D.
3159      :* AND N=0,4,8 OR 12
3160      :* THEN THE ABOVE TABLE SHOULD DIRECTLY IDENTIFY THE
3161      :* FAILING CHIP.
3162      :*
3163      :*3.) IF FOUR BITS ARE DROPPED WHICH FIT THE PATTERN:
3164      :* AN, BN, CN AND DN WHERE N=15,14,... OR 0
3165      :* THEN ANY ONE OF THE FOUR CHIPS ASSOCIATED WITH EACH OF
3166      :* THE BITS AN, BN, CN AND DN COULD BE AT FAULT WITH
3167      :* EQUAL PROBABILITY.
3168      :*
3169      :*4.) IF 16 BITS ARE IN ERROR, FITTING THE PATTERN:
3170      :* AN, AN+1, AN+2, AN+3 WHERE N=0,4,8 OR 12
3171      :* BN, BN+1, BN+2, BN+3
3172      :* CN, CN+1, CN+2, CN+3
3173      :* AND
3174      :* DN, DN+1, DN+2, AN+3
3175      :* THEN ANY ONE OF THE FOUR CHIPS ASSOCIATED
3176      :* WITH THESE BITS COULD BE AT FAULT WITH EQUAL PROBABILITY.
3177      :*
3178      :*5.) IF THE FAILING BIT PATTERNS DISPLAYED IN THE 'AND' AND THE 'OR'
3179      :* DATA TYPED IN THE SUMMARY DOES NOT CONFORM EXPLICITELY TO
3180      :* ANY OF THE ABOVE PATTERNS, THEN THE TROUBLE SHOOTER MUST
3181      :* INTUITIVELY TRY TO FIND WHICH OF THE ABOVE CASES (1 THROUGH 4)
3182      :* IS A 'BEST FIT' OF THE SYMPTOMS.
3183      :*

```

```

3184      ;*****
3185 011646 000004 TST11: SCOPE
3186 011650 170011 SETD :SET FD.
3187 ;TEST ACCUMULATOR 0 WITH FLOATING ONE
3188 011652 012737 042210 001244 MOV #MNUM0,0#STMP5
3189 011660 012737 011710 001236 MOV #G1,0#STMP2
3190 011666 012700 014130 MOV #GPAT00,R0
3191 011672 012701 014170 MOV #GDAT00,R1
3192 011676 104413 LPEPR ;SET UP THE LOOP ON ERROR ADDRESS.
3193 011700 004737 013606 JSR PC,0#GSETUP ;LOAD TEST PATTERN.
3194 011704 012703 000102 MOV #102,R3
3195 011710 G1:
3196 011710 172410 LDD (R0),AC0
3197 011712 174000 STD AC0,AC0
3198 011714 172400 LDD AC0,AC0 ;STORE THE TEST PATTERN.
3199 011716 174011 STD AC0,(R1)
3200 011720 004737 013704 JSR PC,0#GCMP ;COMPARE THE DATA READ WITH
3201 ;THAT WHICH WAS WRITTEN.

```

```

3202 011724 005737 014124      TST      0#GFLAG1
3203 011730 001004      BNE      G2
3204 011732 005137 014124      COM      0#GFLAG1
3205 011736 000261      SEC
3206 011740 000401      BR       G3
3207 011742 000241      G2:     CLC
3208 011744 006160 000006      G3:     ROL      6(R0)      :GENERATE THE NEXT TEST PATTERN.
3209 011750 006160 000004      ROL      4(R0)
3210 011754 006160 000002      ROL      2(R0)
3211 011760 006110      ROL      (R0)
3212 011762 004737 013664      JSR      PC,0#GRESET      :RESET DEFAULT PATTERN IN OUTPUT
3213                                :BUFFER.
3214 011766 077330      SOB      R3,G1
3215 011770 004737 014022      JSR      PC,0#GSUM      :TYPE ERROR SUMMARY.
3216
3217                                :TEST ACCUMULATOR 0 WITH FLOATING ZERO
3218 011774 012737 042210 001244      MOV      #MNUM0,0#STMP5
3219 012002 012737 012032 001236      MOV      #G4,0#STMP2
3220 012010 012700 014140      MOV      #GPAT10,R0
3221 012014 012701 014170      MOV      #GDAT00,R1
3222 012020 104413      LPERR                                :SET UP THE LOOP ON ERROR ADDRESS.
3223 012022 004737 013606      JSR      PC,0#GSETUP      :LOAD TEST PATTERN.
3224 012026 012703 000102      MOV      #102,R3
3225 012032      G4:
3226 012032 172410      LDD      (R0),AC0
3227 012034 174000      STD      AC0,AC0
3228 012036 172400      LDD      AC0,AC0      :STORE THE TEST PATTERN.
3229 012040 174011      STD      AC0,(R1)
3230 012042 004737 013704      JSR      PC,0#GCMP      :COMPARE THE DATA READ WITH
3231                                :THAT WHICH WAS WRITTEN.
3232 012046 005737 014124      TST      0#GFLAG1
3233 012052 001004      BNE      G5
3234 012054 005137 014124      COM      0#GFLAG1
3235 012060 000241      CLC
3236 012062 000401      BR       G6
3237 012064 000261      G5:     SEC
3238 012066 006160 000006      G6:     ROL      6(R0)      :GENERATE THE NEXT TEST PATTERN.
3239 012072 006160 000004      ROL      4(R0)
3240 012076 006160 000002      ROL      2(R0)
3241 012102 006110      ROL      (R0)
3242 012104 004737 013664      JSR      PC,0#GRESET      :RESET DEFAULT PATTERN IN OUTPUT
3243                                :BUFFER.
3244 012110 077330      SOB      R3,G4
3245 012112 004737 014022      JSR      PC,0#GSUM      :TYPE ERROR SUMMARY.
3246
3247                                :TEST ACCUMULATOR 1 WITH FLOATING ONE
3248 012116 012737 042216 001244      MOV      #MNUM1,0#STMP5
3249 012124 012737 012154 001236      MOV      #G7,0#STMP2
3250 012132 012700 014130      MOV      #GPAT00,R0
3251 012136 012701 014170      MOV      #GDAT00,R1
3252 012142 104413      LPERR                                :SET UP THE LOOP ON ERROR ADDRESS.
3253 012144 004737 013606      JSR      PC,0#GSETUP      :LOAD TEST PATTERN.
3254 012150 012703 000102      MOV      #102,R3
3255 012154      G7:
3256 012154 172410      LDD      (R0),AC0
3257 012156 174001      STD      AC0,AC1
    
```

```

3258 012160 172401          LDD    AC1,AC0          :STORE THE TEST PATTERN.
3259 012162 174011          STD    AC0,(R1)
3260 012164 004737 013704    JSR    PC,#GCMP        :COMPARE THE DATA READ WITH
3261                                :THAT WHICH WAS WRITTEN.
3262 012170 005737 014124    TST    #GFLAG1
3263 012174 001004          BNE    G10
3264 012176 005137 014124    COM    #GFLAG1
3265 012202 000261          SEC
3266 012204 000401          BR     G11
3267 012206 000241          G10:  CLC
3268 012210 006160 000006    G11:  ROL    6(R0)      :GENERATE THE NEXT TEST PATTERN.
3269 012214 006160 000004    ROL    4(R0)
3270 012220 006160 000002    ROL    2(R0)
3271 012224 006110          ROL    (R0)
3272 012226 004737 013664    JSR    PC,#GRESET     :RESET DEFAULT PATTERN IN OUTPUT
3273                                :BUFFER.
3274 012232 077330          SOB    R3,G7
3275 012234 004737 014022    JSR    PC,#GSUM       :TYPE ERROR SUMMARY.
3276
3277                                :TEST ACCUMULATOR 1 WITH FLOATING ZERO
3278 012240 012737 042216 001244    MOV    #MNUM1,#STMP5
3279 012246 012737 012276 001236    MOV    #G12,#STMP2
3280 012254 012700 014140          MOV    #GPAT10,R0
3281 012260 012701 014170          MOV    #GDAT00,R1
3282 012264 104413          LPERR                                :SET UP THE LOOP ON ERROR ADDRESS.
3283 012266 004737 013606    JSR    PC,#GSETUP     :LOAD TEST PATTERN.
3284 012272 012703 000102          MOV    #102,R3
3285 012276          G12:
3286 012276 172410          LDD    (R0),AC0
3287 012300 174001          STD    AC0,AC1
3288 012302 172401          LDD    AC1,AC0          :STORE THE TEST PATTERN.
3289 012304 174011          STD    AC0,(R1)
3290 012306 004737 013704    JSR    PC,#GCMP        :COMPARE THE DATA READ WITH
3291                                :THAT WHICH WAS WRITTEN.
3292 012312 005737 014124    TST    #GFLAG1
3293 012316 001004          BNE    G13
3294 012320 005137 014124    COM    #GFLAG1
3295 012324 000241          CLC
3296 012326 000401          BR     G14
3297 012330 000261          G13:  SEC
3298 012332 006160 000006    G14:  ROL    6(R0)      :GENERATE THE NEXT TEST PATTERN.
3299 012336 006160 000004    ROL    4(R0)
3300 012342 006160 000002    ROL    2(R0)
3301 012346 006110          ROL    (R0)
3302 012350 004737 013664    JSR    PC,#GRESET     :RESET DEFAULT PATTERN IN OUTPUT
3303                                :BUFFER.
3304 012354 077330          SOB    R3,G12
3305 012356 004737 014022    JSR    PC,#GSUM       :TYPE ERROR SUMMARY.
3306
3307                                :TEST ACCUMULATOR 2 WITH FLOATING ONE
3308 012362 012737 042223 001244    MOV    #MNUM2,#STMP5
3309 012370 012737 012420 001236    MOV    #G15,#STMP2
3310 012376 012700 014130          MOV    #GPAT00,R0
3311 012402 012701 014170          MOV    #GDAT00,R1
3312 012406 104413          LPERR                                :SET UP THE LOOP ON ERROR ADDRESS.
3313 012410 004737 013606    JSR    PC,#GSETUP     :LOAD TEST PATTERN.

```

FORM 1413 REPRINTED IN U.S.A.

```

3314 012414 012703 000102      MOV      #102,R3
3315 012420      G15:
3316 012420 172410      LDD      (R0),AC0
3317 012422 174002      STD      AC0,AC2
3318 012424 172402      LDD      AC2,AC0      ;STORE THE TEST PATTERN.
3319 012426 174011      STD      AC0,(R1)
3320 012430 004737 013704      JSR      PC,#GCMP      ;COMPARE THE DATA READ WITH
3321      ;THAT WHICH WAS WRITTEN.
3322 012434 005737 014124      TST      @#GFLAG1
3323 012440 001004      BNE      G16
3324 012442 005137 014124      COM      @#GFLAG1
3325 012446 000261      SEC
3326 012450 000401      BR       G17
3327 012452 000241      G16:  CLC
3328 012454 006160 000006      G17:  ROL      6(R0)      ;GENERATE THE NEXT TEST PATTERN.
3329 012460 006160 000004      ROL      4(R0)
3330 012464 006160 000002      ROL      2(R0)
3331 012470 006110      ROL      (R0)
3332 012472 004737 013664      JSR      PC,#GRESET      ;RESET DEFAULT PATTERN IN OUTPUT
3333      ;BUFFER.
3334 012476 077330      SOB      R3,G15
3335 012500 004737 014022      JSR      PC,#GSUM      ;TYPE ERROR SUMMARY.
3336
3337      ;TEST ACCUMULATOR 2 WITH FLOATING ZERO
3338 012504 012737 042223 001244      MOV      #MNUM2,@#STMP5
3339 012512 012737 012542 001236      MOV      #G20,@#STMP2
3340 012520 012700 014140      MOV      #GPAT10,R0
3341 012524 012701 014170      MOV      #GDAT00,R1
3342 012530 104413      LPERR      ;SET UP THE LOOP ON ERROR ADDRESS,
3343 012532 004737 013606      JSR      PC,#GSETUP      ;LOAD TEST PATTERN.
3344 012536 012703 000102      MOV      #102,R3
3345 012542      G20:
3346 012542 172410      LDD      (R0),AC0
3347 012544 174002      STD      AC0,AC2
3348 012546 172402      LDD      AC2,AC0      ;STORE THE TEST PATTERN.
3349 012550 174011      STD      AC0,(R1)
3350 012552 004737 013704      JSR      PC,#GCMP      ;COMPARE THE DATA READ WITH
3351      ;THAT WHICH WAS WRITTEN.
3352 012556 005737 014124      TST      @#GFLAG1
3353 012562 001004      BNE      G21
3354 012564 005137 014124      COM      @#GFLAG1
3355 012570 000241      CLC
3356 012572 000401      BR       G22
3357 012574 000261      G21:  SEC
3358 012576 006160 000006      G22:  ROL      6(R0)      ;GENERATE THE NEXT TEST PATTERN.
3359 012602 006160 000004      ROL      4(R0)
3360 012606 006160 000002      ROL      2(R0)
3361 012612 006110      ROL      (R0)
3362 012614 004737 013664      JSR      PC,#GRESET      ;RESET DEFAULT PATTERN IN OUTPUT
3363      ;BUFFER.
3364 012620 077330      SOB      R3,G20
3365 012622 004737 014022      JSR      PC,#GSUM      ;TYPE ERROR SUMMARY.
3366
3367      ;TEST ACCUMULATOR 3 WITH FLOATING ONE
3368 012626 012737 042230 001244      MOV      #MNUM3,@#STMP5
3369 012634 012737 012664 001236      MOV      #G23,@#STMP2

```

```

3370 012642 012700 014130      MOV      #GPAT00,R0
3371 012646 012701 014170      MOV      #GDAT00,R1
3372 012652 104413              LPERR                    :SET UP THE LOOP ON ERROR ADDRESS.
3373 012654 004737 013606      JSR      PC,#GSETUP      :LOAD TEST PATTERN.
3374 012660 012703 000102      MOV      #102,R3
3375 012664                      G23:
3376 012664 172410              LDD      (R0),AC0
3377 012666 174003              STD      AC0,AC3
3378 012670 172403              LDD      AC3,AC0        :STORE THE TEST PATTERN.
3379 012672 174011              STD      AC0,(R1)
3380 012674 004737 013704      JSR      PC,#GCMP        :COMPARE THE DATA READ WITH
3381                                :THAT WHICH WAS WRITTEN.
3382 012700 005737 014124      TST      @#GFLAG1
3383 012704 001004              RNE      G24
3384 012706 005137 014124      COM      @#GFLAG1
3385 012712 000261              SEC
3386 012714 000401              BR       G25
3387 012716 000241              CLC      G24:
3388 012720 006160 000006      ROL      6(R0)          :GENERATE THE NEXT TEST PATTERN.
3389 012724 006160 000004      ROL      4(R0)
3390 012730 006160 000002      ROL      2(R0)
3391 012734 006110              ROL      (R0)
3392 012736 004737 013664      JSR      PC,#GRESET     :RESET DEFAULT PATTERN IN OUTPUT
3393                                :BUFFER.
3394 012742 077330              SOB      R3,G23
3395 012744 004737 014022      JSR      PC,#GSUM       :TYPE ERROR SUMMARY.
3396
3397                                :TEST ACCUMULATOR 3 WITH FLOATING ZERO
3398 012750 012737 042230 001244      MOV      #MNUM3,@#STMP5
3399 012756 012737 013006 001236      MOV      #G26,@#STMP2
3400 012764 012700 014140      MOV      #GPAT10,R0
3401 012770 012701 014170      MOV      #GDAT00,R1
3402 012774 104413              LPERR                    :SET UP THE LOOP ON ERROR ADDRESS.
3403 012776 004737 013606      JSR      PC,#GSETUP      :LOAD TEST PATTERN.
3404 013002 012703 000102      MOV      #102,R3
3405 013006                      G26:
3406 013006 172410              LDD      (R0),AC0
3407 013010 174003              STD      AC0,AC3
3408 013012 172403              LDD      AC3,AC0        :STORE THE TEST PATTERN.
3409 013014 174011              STD      AC0,(R1)
3410 013016 004737 013704      JSR      PC,#GCMP        :COMPARE THE DATA READ WITH
3411                                :THAT WHICH WAS WRITTEN.
3412 013022 005737 014124      TST      @#GFLAG1
3413 013026 001004              RNE      G27
3414 013030 005137 014124      COM      @#GFLAG1
3415 013034 000241              CLC
3416 013036 000401              BR       G30
3417 013040 000261              SEC      G27:
3418 013042 006160 000006      ROL      6(R0)          :GENERATE THE NEXT TEST PATTERN.
3419 013046 006160 000004      ROL      4(R0)
3420 013052 006160 000002      ROL      2(R0)
3421 013056 006110              ROL      (R0)
3422 013060 004737 013664      JSR      PC,#GRESET     :RESET DEFAULT PATTERN IN OUTPUT
3423                                :BUFFER.
3424 013064 077330              SOB      R3,G26
3425 013066 004737 014022      JSR      PC,#GSUM       :TYPE ERROR SUMMARY.

```

```

3426
3427          ;TEST ACCUMULATOR 4 WITH FLOATING ONE
3428 013072 012737 042237 001244      MOV      #MNUM1,0#STMP5
3429 013100 012737 013130 001236      MOV      #G31,0#STMP2
3430 013106 012700 014130              MOV      #GPAT00,R0
3431 013112 012701 014170              MOV      #GDAT00,R1
3432 013116 104413                      LPERR                    ;SET UP THE LOOP ON ERROR ADDRESS,
3433 013120 004737 013606              JSR      PC,0#GSETUP      ;LOAD TEST PATTERN.
3434 013124 012703 000102              MOV      #102,R3
3435 013130          G31:
3436 013130 172410                      LDD      (R0),AC0
3437 013132 174004                      STD      AC0,AC4
3438 013134 172404                      LDD      AC4,AC0          ;STORE THE TEST PATTERN.
3439 013136 174011                      STD      AC0,(R1)
3440 013140 004737 013704              JSR      PC,0#GCMP        ;COMPARE THE DATA READ WITH
3441                                     ;THAT WHICH WAS WRITTEN.
3442 013144 005737 014124              TST      0#GFLAG1
3443 013150 001004                      BNE      G32
3444 013152 005137 014124              COM      0#GFLAG1
3445 013156 000261                      SEC
3446 013160 000401                      BR       G33
3447 013162 000241          G32:
3448 013164 006160 000006          G33:  ROL      6(R0)          ;GENERATE THE NEXT TEST PATTERN.
3449 013170 006160 000004          ROL      4(R0)
3450 013174 006160 000002          ROL      2(R0)
3451 013200 006110          ROL      (R0)
3452 013202 004737 013664              JSR      PC,0#GRESET      ;RESET DEFAULT PATTERN IN OUTPUT
3453                                     ;BUFFER.
3454 013206 077330                      SOB      R3,G31
3455 013210 004737 014022              JSR      PC,0#GSUM        ;TYPE ERROR SUMMARY.
3456
3457          ;TEST ACCUMULATOR 4 WITH FLOATING ZERO
3458 013214 012737 042237 001244      MOV      #MNUM4,0#STMP5
3459 013222 012737 013252 001236      MOV      #G34,0#STMP2
3460 013230 012700 014140              MOV      #GPAT10,R0
3461 013234 012701 014170              MOV      #GDAT00,R1
3462 013240 104413                      LPERR                    ;SET UP THE LOOP ON ERROR ADDRESS.
3463 013242 004737 013606              JSR      PC,0#GSETUP      ;LOAD TEST PATTERN.
3464 013246 012703 000102              MOV      #102,R3
3465 013252          G34:
3466 013252 172410                      LDD      (R0),AC0
3467 013254 174004                      STD      AC0,AC4
3468 013256 172404                      LDD      AC4,AC0          ;STORE THE TEST PATTERN.
3469 013260 174011                      STD      AC0,(R1)
3470 013262 004737 013704              JSR      PC,0#GCMP        ;COMPARE THE DATA READ WITH
3471                                     ;THAT WHICH WAS WRITTEN.
3472 013266 005737 014124              TST      0#GFLAG1
3473 013272 001004                      BNE      G35
3474 013274 005137 014124              COM      0#GFLAG1
3475 013300 000241                      CLC
3476 013302 000401                      BR       G36
3477 013304 000261          G35:  SEC
3478 013306 006160 000006          G36:  ROL      6(R0)          ;GENERATE THE NEXT TEST PATTERN.
3479 013312 006160 000004          ROL      4(R0)
3480 013316 006160 000002          ROL      2(R0)
3481 013322 006110          ROL      (R0)

```



```

3482 013324 004737 013664      JSR      PC,#GRESET      :RESET DEFAULT PATTERN IN OUTPUT
3483                          :BUFFER.
3484 013330 077330          SOB      R3,G34
3485 013332 004737 014022      JSR      PC,#GSUM        :TYPE ERROR SUMMARY.
3486
3487                          :TEST ACCUMULATOR 5 WITH FLOATING ONE
3488 013336 012737 042245 001244    MOV      #MNUM5,#STMP5
3489 013344 012737 013374 001236    MOV      #G37,#STMP2
3490 013352 012700 014130          MOV      #GPAT00,R0
3491 013356 012701 014170          MOV      #GDAT00,R1
3492 013362 104413          LPERR                     :SET UP THE LOOP ON ERROR ADDRESS.
3493 013364 004737 013606      JSR      PC,#GSETUP      :LOAD TEST PATTERN.
3494 013370 012703 000102          MOV      #102,R3
3495 013374          G37:
3496 013374 172410          LDD      (R0),AC0
3497 013376 174005          STD      AC0,AC5
3498 013400 172405          LDD      AC5,AC0          :STORE THE TEST PATTERN.
3499 013402 174011          STD      AC0,(R1)
3500 013404 004737 013704      JSR      PC,#GCMP        :COMPARE THE DATA READ WITH
3501                          :THAT WHICH WAS WRITTEN.
3502 013410 005737 014124          TST      0#GFLAG1
3503 013414 001004          BNE      G40
3504 013416 005137 014124          COM      0#GFLAG1
3505 013422 000261          SEC
3506 013424 000401          BR       G41
3507 013426 000241          G40: CLC
3508 013430 006160 000006          G41: ROL      6(R0)          :GENERATE THE NEXT TEST PATTERN.
3509 013434 006160 000404          ROL      4(R0)
3510 013440 006160 000002          ROL      2(R0)
3511 013444 006110          ROL      (R0)
3512 013446 004737 013664      JSR      PC,#GRESET      :RESET DEFAULT PATTERN IN OUTPUT
3513                          :BUFFER.
3514 013452 077330          SOB      R3,G37
3515 013454 004737 014022      JSR      PC,#GSUM        :TYPE ERROR SUMMARY.
3516
3517                          :TEST ACCUMULATOR 5 WITH FLOATING ZERO
3518 013460 012737 042245 001244    MOV      #MNUM5,#STMP5
3519 013466 012737 013516 001236    MOV      #G42,#STMP2
3520 013474 012700 014140          MOV      #GPAT10,R0
3521 013500 012701 014170          MOV      #GDAT00,R1
3522 013504 104413          LPERR                     :SET UP THE LOOP ON ERROR ADDRESS.
3523 013506 004737 013606      JSR      PC,#GSETUP      :LOAD TEST PATTERN.
3524 013512 012703 000102          MOV      #102,R3
3525 013516          G42:
3526 013516 172410          LDD      (R0),AC0
3527 013520 174005          STD      AC0,AC5
3528 013522 172405          LDD      AC5,AC0          :STORE THE TEST PATTERN.
3529 013524 174011          STD      AC0,(R1)
3530 013526 004737 013704      JSR      PC,#GCMP        :COMPARE THE DATA READ WITH
3531                          :THAT WHICH WAS WRITTEN.
3532 013532 005737 014124          TST      0#GFLAG1
3533 013536 001004          BNE      G43
3534 013540 005137 014124          COM      0#GFLAG1
3535 013544 000241          CLC
3536 013546 000401          BR       G44
3537 013550 000261          G43: SEC

```

```

3538 013552 006160 000006      G44:  ROL    6(R0)          ;GENERATE THE NEXT TEST PATTERN.
3539 013556 006160 000004      ROL    4(R0)
3540 013562 006160 000002      ROL    2(R0)
3541 013566 006110 000000      ROL    (R0)
3542 013570 004737 013664      JSR    PC,#GRESET          ;RESET DEFAULT PATTERN IN OUTPUT
3543                                ;BUFFER.
3544 013574 077330              SOB    R3,G42
3545 013576 004737 014022      JSR    PC,#GSUM           ;TYPE ERROR SUMMARY.
3546
3547
3548 013602 000137 014202      JMP    @#GDONE
3549
3550                                ;USE THIS ROUTINE TO INITIALIZE ALL THE DATA BUFFERS.
3551 013606 012705 014124      GSETUP: MOV    #GFLAG1,R5
3552 013612 012704 000026      MOV    #26,R4
3553 013616 005025              1$:  CLR    (R5)+
3554 013620 077402              SOB    R4,1$
3555
3556 013622 012705 014140      MOV    #GPAT10,R5
3557 013626 012704 000010      MOV    #10,R4
3558 013632 005125              2$:  COM    (R5)+
3559 013634 077402              SOB    R4,2$
3560
3561 013636 020067 000266      GS1:  CMP    R0,GPAT00
3562 013642 001401              BEQ    3$
3563 013644 000207              RTS    PC
3564
3565 013646 012705 014170      3$:  MOV    #GDAT00,R5
3566 013652 012704 000004      MOV    #4,R4
3567 013656 005125              4$:  COM    (R5)+
3568 013660 077402              SOB    R4,4$
3569 013662 000207              RTS    PC
3570
3571 013664 012705 014170      GRESET: MOV   #GDAT00,R5
3572 013670 012704 000004      MOV    #4,R4
3573 013674 005025              1$:  CLR    (R5)+
3574 013676 077402              SOB    R4,1$
3575 013700 000137 013636      JMP    @#GS1
3576
3577                                ;SEE IF THE DATA WRITTEN MATCHES THE DATA READ.
3578 013704 012705 014170      GCMP:  MOV    #GDAT00,R5
3579 013710 012704 000004      MOV    #4,R4
3580 013714 010002              MOV    R0,R2
3581 013716 022225              1$:  CMP    (R2)+,(R5)+
3582 013720 001402              BEQ    2$
3583 013722 000137 013732      JMP    @#GERR1
3584 013726 077405              2$:  SOB    R4,1$
3585 013730 000207              RTS    PC
3586
3587                                ;COME HERE TO REPORT AND RECORD ERRORS.
3588 013732 012637 014200      GERR1: MOV    (SP)+,@#GADR          ;SAVE THE RETURN ADDRESS.
3589 013736 010003              MOV    R0,R3                    ;COMPUTE 'OR' OF BAD DATA.
3590 013740 012705 014160      MOV    #GOR0,R5
3591 013744 012704 000004      MOV    #4,R4
3592 013750 052325              1$:  BIS    (R3)+,(R5)+
3593 013752 077402              SOB    R4,1$

```

```

3594
3595 013754 010003      MOV      R0,R3      :COMPUTE 'AND' OF BAD DATA.
3596 013756 012705 014150      MOV      #GAND0,R5
3597 013762 012704 000004      MOV      #4,R4
3598 013766 012302      2S:      MOV      (R3)+,R2
3599 013770 005102      COM      R2
3600 013772 040225      BIC      R2,(R5)+
3601 013774 077404      SOB      R4,2S
3602
3603 013776 005237 014126      INC      @#GFLAG2      :INCREMENT ERROR COUNT.
3604 014002 010037 001240      MOV      R0,@#STMP3
3605 014006 012737 014170 001242      MOV      #GDAT00,@#STMP4
3606 014014 104044      3S:      ERROR    44
3607 014016 000177 000156      JMP      @GADR
3608
3609      :SEE IF ANY ERRORS HAVE OCCURRED AND WHETHER OR NOT AN ERROR SUMMARY
3610      :SHOULD BE TYPED.
3611 014022 005737 014126      GSUM:    TST      @#GFLAG2      :ANY ERRORS?
3612 014026 001135      BEQ      3S
3613
3614 014030 032777 020000 165102      BIT      #SW13,@SWR      :INHIBIT ERROR PRINT OUT?
3615 014036 001404      BEQ      1S
3616 014040 032777 000200 165072      BIT      #SW7,@SWR      :PRINT SUMMARY?
3617 014046 001425      BEQ      3S
3618
3619 014050 013737 014126 001246 1S:      MOV      @#GFLAG2,@#STMP6      :YES PRINT SUMMARY.
3620 014056 012737 014150 001240      MOV      #GAND0,@#STMP3
3621 014064 012737 014160 001242      MOV      #GOR0,@#STMP4
3622 014072 012637 014200      MOV      (SP)+,@#GADR      :SAVE RETURN ADDRESS.
3623 014076 012737 014112 001116      MOV      #2S,@#SERRPC
3624 014104 112737 000045 001114      MOV      #45,@#SITEMB
3625 014112 004737 037546      2S:      JSR      PC,@#ERTYPE
3626 014116 000177 000056      JMP      @GADR
3627 014122 000207      3S:      RTS      PC
3628
3629
3630 014124 000000      GFLAG1: 0
3631 014126 000000      GFLAG2: 0
3632
3633 014130 000000      GPAT00: 0
3634 014132 000000      GPAT01: 0
3635 014134 000000      GPAT02: 0
3636 014136 000000      GPAT03: 0
3637
3638 014140 177777      GPAT10: -1
3639 014142 177777      GPAT11: -1
3640 014144 177777      GPAT12: -1
3641 014146 177777      GPAT13: -1
3642
3643 014150 177777      GAND0:  -1
3644 014152 177777      GAND1:  -1
3645 014154 177777      GAND2:  -1
3646 014156 177777      GAND3:  -1
3647
3648 014160 000000      GOR0:   0
3649 014162 000000      GOR1:   0
    
```

3650 014164 000000  
 3651 014166 000000  
 3652  
 3653 014170 000000  
 3654 014172 000000  
 3655 014174 000000  
 3656 014176 000000  
 3657  
 3658 014200 000000  
 3659  
 3660 014202  
 3661 014202 104412  
 3662  
 3663  
 3664  
 3665  
 3666  
 3667  
 3668  
 3669  
 3670  
 3671  
 3672  
 3673  
 3674  
 3675 014204 000004  
 3676 014206 104413  
 3677  
 3678 014210 005037 014734  
 3679 014214 012700 014736  
 3680 014220 012701 015056  
 3681 014224 012703 000024  
 3682 014230 012120  
 3683 014232 077302  
 3684  
 3685 014234 004767 000422  
 3686  
 3687 014240 170011  
 3688  
 3689 014242 012700 014736  
 3690 014246 172410  
 3691 014250 174001  
 3692  
 3693 014252 012700 014746  
 3694 014256 172410  
 3695 014260 174002  
 3696  
 3697 014262 012700 014756  
 3698 014266 172410  
 3699 014270 174003  
 3700  
 3701 014272 012700 014766  
 3702 014276 172410  
 3703 014300 174004  
 3704  
 3705 014302 012700 014776

GOR2: 0  
 GOR3: 0  
 GDATA0: 0  
 GDATA1: 0  
 GDATA2: 0  
 GDATA3: 0  
 GADR: 0  
 GDONE:  
 RSETUP :GO INITIALIZE THE FPS AND STACK; AND  
 :SEE IF THE USER HAS EXPRESSED  
 :THE DESIRE TO CHANGE THE SOFTWARE  
 :VIRTUAL CONSOLE SWITCH REGISTER (HAS  
 :THE USER TYPED CONTROL G?).  
 :\*\*\*\*\*  
 :\*TEST 12 FPP ACCUMULATORS DUAL ADDRESS TEST  
 :\*  
 :\*THIS TEST PERFORMS A DUAL ADDRESSING TEST ON THE FLOATING ACCUMULATORS.  
 :\*NOTE THAT ACCUMULATOR ZERO IS USED TO ACCESS ALL THE OTHERS.  
 :\*  
 :\*\*\*\*\*  
 TST12: SCOPE :SET UP THE LOOP ON ERROR ADDRESS.  
 LPERR  
 H1: CLR 0#HFLAG  
 MOV #HA1W,R0 ;INITIALIZE THE LOAD BUFFER DATA.  
 MOV #HDATA1,R1  
 MOV #24,R3  
 H2: MOV (R1)+,(R0)+  
 SOB R3,H2  
 JSR PC,HCLR ;CLEAR THE OUTPUT DATA BUFFER.  
 H3: SETD  
 :LOAD ACCUMULATOR 1  
 MOV #HA1W,R0  
 LDD (R0),AC0  
 STD AC0,AC1  
 :LOAD ACCUMULATOR 2  
 MOV #HA2W,R0  
 LDD (R0),AC0  
 STD AC0,AC2  
 :LOAD ACCUMULATOR 3  
 MOV #HA3W,R0  
 LDD (R0),AC0  
 STD AC0,AC3  
 :LOAD ACCUMULATOR 4  
 MOV #HA4W,R0  
 LDD (R0),AC0  
 STD AC0,AC4  
 :LOAD ACCUMULATOR 5  
 MOV #HA5W,R0

```

3706 @14306 172410 LDD (R0),AC0
3707 @14310 174005 STD AC0,AC5
3708
3709 @14312 @04737 @14546 H4: JSR PC,@#HSTD ;GO READ ALL ACCUMULATORS BACK.
3710
3711 @14316 @04737 @14624 JSR PC,@#HCMP ;SEE IF DATA IS CORRECT.
3712
3713 ;COMPLIMENT EACH WORD OF THE DATA STORED IN ACCUMULATOR 1,
3714 ;RELOAD THAT ACCUMULATOR, READ ALL THE ACCUMULATORS BACK AND CHECK
3715 ;THE DATA.
3716 @14322 @12700 @14736 MOV #HA1W,R0
3717 @14326 @12702 @000004 MOV #4,R2
3718 @14332 @10001 MOV R0,R1
3719 @14334 @05121 H5: COM (R1)+
3720 @14336 172410 LDD (R0),AC0
3721 @14340 174001 STD AC0,AC1
3722 @14342 @04737 @14546 JSR PC,@#HSTD ;READ ALL THE ACCUMULATORS BACK.
3723 @14346 @04737 @14624 JSR PC,@#HCMP ;CHECK THE DATA.
3724 @14352 @77210 SOB R2,H5
3725
3726 ;COMPLIMENT EACH WORD OF THE DATA STORED IN ACCUMULATOR 2,
3727 ;RELOAD THAT ACCUMULATOR, READ ALL THE ACCUMULATORS BACK AND CHECK
3728 ;THE DATA.
3729 @14354 @12700 @14746 MOV #HA2W,R0
3730 @14360 @12702 @000004 MOV #4,R2
3731 @14364 @10001 MOV R0,R1
3732 @14366 @05121 H6: COM (R1)+
3733 @14370 172410 LDD (R0),AC0
3734 @14372 174002 STD AC0,AC2
3735 @14374 @04737 @14546 JSR PC,@#HSTD ;READ ALL THE ACCUMULATORS BACK.
3736 @14400 @04737 @14624 JSR PC,@#HCMP ;CHECK THE DATA.
3737 @14404 @77210 SOB R2,H6
3738
3739 ;COMPLIMENT EACH WORD OF THE DATA STORED IN ACCUMULATOR 3,
3740 ;RELOAD THAT ACCUMULATOR, READ ALL THE ACCUMULATORS BACK AND CHECK
3741 ;THE DATA.
3742 @14406 @12700 @14756 MOV #HA3W,R0
3743 @14412 @12702 @000004 MOV #4,R2
3744 @14416 @10001 MOV R0,R1
3745 @14420 @05121 H7: COM (R1)+
3746 @14422 172410 LDD (R0),AC0
3747 @14424 174003 STD AC0,AC3
3748 @14426 @04737 @14546 JSR PC,@#HSTD ;READ ALL THE ACCUMULATORS BACK.
3749 @14432 @04737 @14624 JSR PC,@#HCMP ;CHECK THE DATA.
3750 @14436 @77210 SOB R2,H7
3751
3752 ;COMPLIMENT EACH WORD OF THE DATA STORED IN ACCUMULATOR 4,
3753 ;RELOAD THAT ACCUMULATOR, READ ALL THE ACCUMULATORS BACK AND CHECK
3754 ;THE DATA.
3755 @14440 @12700 @14766 MOV #HA4W,R0
3756 @14444 @12702 @000004 MOV #4,R2
3757 @14450 @10001 MOV R0,R1
3758 @14452 @05121 H10: COM (R1)+
3759 @14454 172410 LDD (R0),AC0
3760 @14456 174004 STD AC0,AC4
3761 @14460 @04737 @14546 JSR PC,@#HSTD ;READ ALL THE ACCUMULATORS BACK.

```

```

3762 014464 004737 014624      JSR    PC,#HCMP      ;CHECK THE DATA.
3763 014470 077210      SOB    R2,H10
3764
3765      ;COMPLIMENT EACH WORD OF THE DATA STORED IN ACCUMULATOR 5,
3766      ;RELOAD THAT ACCUMULATOR, READ ALL THE ACCUMULATORS BACK AND CHECK
3767      ;THE DATA.
3768 014472 012700 014776      MOV    #HA5W,R0
3769 014476 012702 000004      MOV    #4,R2
3770 014502 010001      MOV    R0,R1
3771 014504 005121      H11:   COM    (R1)+
3772 014506 172410      LDD    (R0),AC0
3773 014510 174005      STD    AC0,AC5
3774 014512 004737 014546      JSR    PC,#HSTD      ;READ ALL THE ACCUMULATORS BACK.
3775 014516 004737 014624      JSR    PC,#HCMP      ;CHECK THE DATA.
3776 014522 077210      SOB    R2,H11
3777
3778
3779 014524 005737 014734      TST    @#HFLAG
3780 014530 001402      BEQ    H12
3781 014532 000137 015126      JMP    @#HDONE
3782
3783 014536 005137 014734      H12:   COM    @#HFLAG
3784 014542 000137 014240      JMP    @#H3
3785
3786      ;STORE ALL ACCUMULATORS IN THE OUTPUT BUFFERS.
3787 014546 004737 014662      HSTD:  JSR    PC,#HCLR      ;CLEAR ALL OUTPUT BUFFERS.
3788      ;STORE ACCUMULATOR 1
3789 014552 012704 015006      MOV    #HA1R,R4
3790 014556 172401      LDD    AC1,AC0
3791 014560 174014      STD    AC0,(R4)
3792      ;STORE ACCUMULATOR 2
3793 014562 012704 015016      MOV    #HA2R,R4
3794 014566 172402      LDD    AC2,AC0
3795 014570 174014      STD    AC0,(R4)
3796      ;STORE ACCUMULATOR 3
3797 014572 012704 015026      MOV    #HA3R,R4
3798 014576 172403      LDD    AC3,AC0
3799 014600 174014      STD    AC0,(R4)
3800      ;STORE ACCUMULATOR 4
3801 014602 012704 015036      MOV    #HA4R,R4
3802 014606 172404      LDD    AC4,AC0
3803 014610 174014      STD    AC0,(R4)
3804      ;STORE ACCUMULATOR 5
3805 014612 012704 015046      MOV    #HA5R,R4
3806 014616 172405      LDD    AC5,AC0
3807 014620 174014      STD    AC0,(R4)
3808 014622 000207      RTS    PC
3809
3810      ;COMPARE DATA LOADED WITH DATA READ.
3811 014624 012637 014732      HCMP:  MOV    (SP)+,@#HADR      ;SAVE RETURN ADDRESS.
3812 014630 012703 014736      MOV    #HA1W,R3
3813 014634 012704 015006      MOV    #HA1R,R4
3814 014640 012705 000024      MOV    #24,R5
3815 014644 022324      HCMP1: CMP    (R3)+,(R4)+
3816 014646 001402      BEQ    HCMP2
3817 014650 000137 014700      JMP    @#HEPERROR
    
```

```

3818 014654 077505          HCMP2: SOB      R5,HCMP1
3819 014656 000177 000050          JMP      @HADR
3820
3821          ;CLEAR THE DATA OUTPUT BUFFER.
3822 014662 012704 015006          HCLR:  MOV      #HA1R,R4
3823 014666 012705 000024          MOV      #24,R5
3824 014672 005024          HCLR1: CLR      (R4)+
3825 014674 077502          SOB      R5,HCLR1
3826 014676 000207          RTS      PC
3827
3828          ;REPORT ERROR.
3829 014700          HEPROP:
3830 014700 012703 014736          MOV      #HA1W,R3
3831 014704 012704 001236          MOV      #STMP2,R4
3832 014710 012705 000012          MOV      #12,R5
3833 014714 010324          1S:    MOV      R3,(R4)+
3834 014716 062703 000010          ADD      #10,R3
3835 014722 077504          SOB      R5,1S
3836 014724 104046          2S:    ERROR   46
3837 014726 000137 015126          JMP      @#HDONE
3838
3839
3840 014732 000000          HADR:  0
3841 014734 000000          HFLAG: 0
3842
3843 014736 000000 000000 000000 HA1W:  .WORD   0,0,0,0
3844 014744 000000          HA2W:  .WORD   0,0,0,0
3845 014746 000000 000000 000000 HA3W:  .WORD   0,0,0,0
3846 014754 000000          HA4W:  .WORD   0,0,0,0
3847 014756 000000 000000 000000 HA5W:  .WORD   0,0,0,0
3848 014764 000000
3849 014766 000000 000000 000000 HA1R:  .WORD   0,0,0,0
3850 014774 000000
3851 014776 000000 000000 000000 HA2R:  .WORD   0,0,0,0
3852 015004 000000
3853
3854 015006 000000 000000 000000 HA3R:  .WORD   0,0,0,0
3855 015014 000000
3856 015016 000000 000000 000000 HA4R:  .WORD   0,0,0,0
3857 015024 000000
3858 015026 000000 000000 000000 HA5R:  .WORD   0,0,0,0
3859 015034 000000
3860 015036 000000 000000 000000 HDAT1: .WORD  73567,73567,73567,73567
3861 015044 000000
3862 015046 000000 000000 000000 HDAT2: .WORD  63146,63146,63146,63146
3863 015054 000000
3864
3865 015056 073567 073567 073567 HDAT3: .WORD  10421,10421,10421,10421
3866 015064 073567
3867 015066 063146 063146 063146 HDAT4: .WORD  31463,31463,31463,31463
3868 015074 063146
3869 015076 010421 010421 010421 HDAT5: .WORD  42104,42104,42104,42104
3870 015104 010421
3871 015106 031463 031463 031463
3872 015114 031463
3873 015116 042104 042104 042104
    
```

```

3874 015124 042104
3875
3876 015125 HDONE:
3877 015126 104412 RSETUP ;GO INITIALIZE THE FPS AND STACK; AND
3878 ;SEE IF THE USER HAS EXPRESSED
3879 ;THE DESIRE TO CHANGE THE SOFTWARE
3880 ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
3881 ;THE USER TYPED CONTROL G?).
3882
3883
3884 ;*****
3885 ;*TEST 13 FSRC MODE 0 WITH ILLEGAL ACCUMULATOR TEST
3886 ;*
3887 ;*THIS IS A TEST OF FSRC MODE 0 WITH ACCUMULATORS 6 AND 7. USE OF
3888 ;*EITHER OF THESE NON-EXISTENT ACCUMULATORS SHOULD RESULT IN A TRAP TO 244
3889 ;*WITH FEC=2 (ILLEGAL FPP INSTRUCTION).
3890 ;*
3891 ;*****
3892 015130 000004 TST13: SCOPE
3893 015132 S1:
3894 015132 104413 LPERR ;SET UP THE LOOP ON ERROR ADDRESS,
3895 015134 170011 SETD ;SET FD
3896 015136 012700 015642 MOV #SPAT10,R0 ;LOAD AC0
3897 015142 172410 LDD (R0),AC0
3898
3899 015144 012737 015342 000244 MOV #SERR0,@#FPVECT ;USE OF THE NON-EXISTENT AC=
3900 ;CUMULATOR SHOULD RESULT IN
3901 ;A TRAP TO 244.
3902 015152 012700 000001 MOV #1,R0 ;A FAILURE IN THE FSRC FLOWS
3903 ;WILL RESULT IN AN ODD ADDRESS
3904 015156 012737 015552 000004 MOV #SERR1,@#ERRVECT ;TRAP TO 4.
3905 015164 005003 CLR R3
3906
3907 015166 172407 S2: LDD AC7,AC0
3908 015170 170000 S3: CFCC
3909 015172 005203 INC R3
3910 015174 005203 S4: INC R3
3911
3912 015176 012701 015652 MOV #SDAT00,R1 ;NO TRAP OCCURRED!!
3913 015202 174011 STD AC0,(R1) ;SEE IF AC0 WAS MODIFIED.
3914
3915 015204 012701 015652 MOV #SDAT00,R1
3916 015210 012702 015642 MOV #SPAT10,R2
3917 015214 012703 000004 MOV #4,R3
3918 015220 022122 S5: CMP (P1)+,(P2)+
3919 015222 001402 BEQ S6
3920 015224 000137 015502 JMP @#SERR2
3921 015230 077305 S6: SOB R3,S5
3922
3923 015232 000137 015526 JMP @#SERR3
3924
3925 ;NOW TEST AC6.
3926 015236 S7:
3927 015236 104413 LPERR ;SET UP THE LOOP ON ERROR ADDRESS,
3928 015240 170011 SETD
3929

```



```

3930 015242 012700 015642      MOV      #SPAT10,R0      :LOAD AC0
3931 015246 172410      LDD      (R0),AC0
3932
3933 015250 012737 015420 000244      MOV      #SERR4,#FPVECT
3934 015256 012700 000001      MOV      #1,R0
3935 015262 012737 015604 000004      MOV      #SERR5,#ERPVECT
3936 015270 005003      CLR      R3
3937
3938 015272 172406      S8:     LDD      AC6,AC0
3939 015274 170000      S9:     CFCC
3940 015276 005203      INC      R3
3941 015300 005203      S10:    INC      R3
3942
3943 015302 012701 015652      MOV      #SDAT00,R1
3944 015306 174011      STD      AC0,(R1)      :NO TRAP! GET AC0.
3945
3946 015310 012701 015652      MOV      #SDAT00,R1      :WAS AC0 MODIFIED.
3947 015314 012702 015642      MOV      #SPAT10,R2
3948 015320 012703 000004      MOV      #4,R3
3949 015324 022122      S11:    CMP      (R1)+,(R2)+
3950 015326 001402      BEQ      S12
3951 015330 000137 015514      JMP      @#SERR6
3952 015334 077305      S12:    SOB      R3,S11
3953 015336 000137 015540      JMP      @#SERR7
3954
3955      :TRAPPED TO 244.
3956 015342 021627 015170      SERR0:  CMP      (SP),#S3      :PC OF TRAP CORRECT?
3957 015346 001402      BEQ      1$
3958 015350 000137 040200      JMP      @#FPSPUR
3959
3960 015354 012737 015236 015636 1$:     MOV      #S7,@#SADP
3961
3962 015362 011637 001236      SERR10: MOV      (SP),@#STMP2
3963 015366 022626      CMP      (SP)+,(SP)+
3964 015370 005004      CLR      R4
3965 015372 170204      STEPS   R4      :IS FPS CORRECT?
3966 015374 022704 100200      CMP      #100200,R4
3967 015400 001020      BNE     SERR15
3968
3969 015402 005004      CLR      R4
3970 015404 170304      STST   R4      :IS FEC CORRECT?
3971 015406 022704 000002      CMP      #2,R4
3972 015412 001023      BNE     SERR20
3973 015414 000177 000216      JMP      @SADR
3974
3975 015420 021627 015274      SERR1:  CMP      (SP),#S9
3976 015424 001402      BEQ      1$
3977 015426 000137 040200      JMP      @#FPSPUR
3978 015432 012737 015662 015636 1$:     MOV      #SDONE,@#SADR
3979 015440 000750      BR      SERR10
3980
3981      :REPORT FPS FAILURE:
3982 015442 012737 100200 001242  SERR15: MOV      #100200,@#STMP4
3983 015450 010437 001240      MOV      R4,@#STMP3
3984 015454 104117      1$:     ERROR   117
3985 015456 000177 000154      JMP      @SADR
    
```

3986  
3987  
3988  
3989  
3990  
3991  
3992  
3993  
3994  
3995  
3996  
3997  
3998  
3999  
4000  
4001  
4002  
4003  
4004  
4005  
4006  
4007  
4008  
4009  
4010  
4011  
4012  
4013  
4014  
4015  
4016  
4017  
4018  
4019  
4020  
4021  
4022  
4023  
4024  
4025  
4026  
4027  
4028  
4029  
4030  
4031  
4032  
4033  
4034  
4035  
4036  
4037  
4038  
4039  
4040  
4041

015462 012737 000002 001242  
 015470 010437 001240  
 015474 104120  
 015476 000177 000134  
  
 015502 012737 015166 001236  
 015510 104112  
 015512 000463  
 015514 012737 015272 001236  
 015522 104114  
 015524 000456  
  
 015526 012737 015166 001236  
 015534 104111  
 015536 000451  
 015540 012737 015272 001236  
 015546 104113  
 015550 000444  
  
 015552 021627 015170  
 015556 001405  
 015560 021627 015174  
 015564 001402  
 015566 001137 040232  
  
 015572 011637 001236  
 015576 022626  
 015600 104115  
 015602 000427  
  
 015604 021627 015272  
 015610 001405  
 015612 021627 015274  
 015616 001402  
 015620 000137 040232  
  
 015624 011637 001236  
 015630 022626  
 015632 104116  
 015634 000412  
  
 015636 000000  
 015640 177777  
 015642 010421  
 015644 021042  
 015646 031463  
 015650 042104  
  
 015652 000000  
 015654 000000  
 015656 000000

:REPORT REC RAD:

SERR20: MOV #2,0#STMP4  
 MOV #4,0#STMP3  
 1S: ERROR 120  
 JMP #SADR

:ACC WAS MODIFIED, (BUT FSRG) FORK FAILED.

SERR2: MOV #S2,0#STMP2  
 1S: ERROR 112  
 BR SDONE  
 SERR6: MOV #S8,0#STMP2  
 1S: ERROR 114  
 BR SDONE

SERR3: MOV #S2,0#STMP2  
 1S: ERROR 111  
 BR SDONE  
 SERR7: MOV #S8,0#STMP2  
 1S: ERROR 113  
 BR SDONE

:FAILURE OF (BUT FSRG) CAUSED AN ODD ADDRESS TRAP TO 4. DID TRAP OCCUR ON TESTED INSTRUCTION?

SERR1: CMP (SP),#S2  
 BEQ 1S  
 CMP (SP),#S4  
 BEQ 1S  
 JMP #CPSPUR  
 1S: MOV (SP),0#STMP2  
 CMP (SP)+,(SP)+  
 2S: ERROR 115  
 BR SDONE

:DID TRAP OCCUR ON TEST INSTRUCTION?

SERR5: CMP (SP),#S9  
 BEQ 1S  
 CMP (SP),#S9  
 BEQ 1S  
 JMP #CPSPUR  
 1S: MOV (SP),0#STMP2  
 CMP (SP)+,(SP)+  
 2S: ERROR 116  
 BR SDONE

SADR: 0  
 -1  
 SPAT10: 10421  
 SPAT11: 21042  
 SPAT12: 31463  
 SPAT13: 42104

SDAT00: 0  
 SDAT01: 0  
 SDAT02: 0

```

4042 015660 000000          SDAT03: 0
4043
4044 015662          SDONE:
4045 015662 104412          RSETUP          ;GO INITIALIZE THE FPS AND STACK; AND
4046                                     ;SEE IF THE USER HAS EXPRESSED
4047                                     ;THE DESIRE TO CHANGE THE SOFTWARE
4048                                     ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
4049                                     ;THE USER TYPED CONTROL G?).
4050
4051          ;*****
4052          ;*TEST 14          FSRC MODE 2 TEST
4053          ;*
4054          ;* THIS IS A TEST OF FSRC MODE 2, AUTO
4055          ;* INCREMENT MODE.
4056          ;*
4057          ;*****
4058 015664 000004          TST14: SCOPE
4059 015666 104413          LPERR          ;SET UP THE LOOP ON ERROR ADDRESS.
4060
4061 015670          J1:
4062 015670 170011          SETD          ;SET DOUBLE MODE
4063
4064 015672 012700 016146          MOV          #JDAT0,R0
4065 015676 172410          LDD          (R0),AC0          ;LOAD AC0
4066
4067 015700 012700 016126          MOV          #JDATI0,R0
4068 015704 005003          CLR          R3
4069 015706 012737 015776 000004          MOV          #JERR0,#ERRVECT
4070
4071 015714 172420          J2:          LDD          (R0)+,AC0          ;TEST INSTRUCTION
4072 015716 005203          J3:          INC          R3
4073 015720 005203          J4:          INC          R3
4074
4075 015722 012701 016136          MOV          #JDAT00,R1
4076 015726 174011          STD          AC0,(R1)          ;PICK UP RESULTS
4077
4078 015730 020027 016116          CMP          R0,#JBUF0          ;WAS AN AUTO
4079 015734 001001          BNE          1$          ;DECREMENT EXECUTED?
4080 015736 000442          BR          JERR1
4081
4082 015740 012702 016126          1$:          MOV          #JDATI0,R2          ;IS DATA CORRECT?
4083 015744 012703 016136          MOV          #JDAT00,R3
4084 015750 012704 000004          MOV          #4,R4
4085 015754 022223          J5:          CMP          (R2)+,(R3)+
4086 015756 001401          BEQ          J6
4087 015760 000443          BR          JERR2
4088 015762 077404          J6:          SOB          R4,J5
4089
4090 015764 022700 016136          CMP          #JDATI0+10,R0          ;WAS R0 INCREM.
4091 015770 001401          BEQ          J7          ;BY 10 (OCTAL)
4092 015772 000424          BR          JERR1
4093
4094 015774 000470          J7:          BR          JDONE
4095
4096          ;IF A TRAP THROUGH 4 OCCURS COME HERE
4097

```

```

4098 015776 021627 015716      JERR0:  CME      (SP),#J3      ;SEE IF THE TRAP
4099 016002 001405              BEG      J10              ;OCCURRED ON THE
4100 016004 021627 015720      CMP      (SP),#J4      ;TESTED INSTRUCTION
4101 016010 001402              BEG      J10
4102 016012 000137 040232      JMP      @#CPSPUR
4103
4104 016016 012737 000762 001240  J10:    MOV      #762,@#STMP3    ;REPORT FSRC FLOW
4105 016024 012737 000322 001242      MOV      #322,@#STMP4    ;FAILURE
4106 016032 011637 001236              MOV      (SP),@#STMP2
4107 016036 022626              CMP      (SP)+,(SP)+
4108 016040 104052              1S:     ERROR    52
4109 016042 000445              BR       JDONE
4110
4111 016044              JERR1:              ;REPORT, R0 NOT
4112 016044 012737 015714 001236      MOV      #J2,@#STMP2    ;CORRECTLY AFFECTED
4113 016052 010037 001240              MOV      R0,@#STMP3
4114 016056 012737 016136 001242      MOV      #JDATI0+10,@#STMP4
4115 016064 104053              1S:     ERROR    53
4116 016066 000433              BR       JDONE
4117
4118              ;REPORT DATA FAILURE
4119
4120 016070              JERR2:
4121 016070 012737 015714 001236      MOV      #J2,@#STMP2
4122 016076 012737 016126 001240      MOV      #JDATI0,@#STMP3
4123 016104 012737 016136 001242      MOV      #JDATO0,@#STMP4
4124 016112 104054              1S:     ERROR    54
4125 016114 000420              BR       JDONE
4126
4127 016116 010421              JBUF0:  .WORD    010421
4128 016120 021042              JBUF1:              021042
4129 016122 042104              JBUF2:              042104
4130 016124 031463              JBUF3:              031463
4131
4132 016126 052525              JDATI0:              052525
4133 016130 114631              JDATI1:              114631
4134 016132 063146              JDATI2:              063146
4135 016134 073567              JDATI3:              073567
4136
4137 016136 000000              JDATO0:              0
4138 016140 000000              JDATO1:              0
4139 016142 000000              JDATO2:              0
4140 016144 000000              JDATO3:              0
4141
4142 016146 177777              JDAT0:              -1
4143 016150 177777              JDAT1:              -1
4144 016152 177777              JDAT2:              -1
4145 016154 177777              JDAT3:              -1
4146
4147
4148 016156              JDONE:
4149 016156 104412      RSETUP              ;GO INITIALIZE THE FPS AND STACK; AND
4150              ;SEE IF THE USER HAS EXPRESSED
4151              ;THE DESIRE TO CHANGE THE SOFTWARE
4152              ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
4153              ;THE USER TYPED CONTROL G?).

```

```

4154
4155
4156 ;*****
4157 ;*TEST 15 FSRC MODE 4 TEST
4158 ;*
4159 ;* THIS IS A TEST OF FSRC MODE 4, AUTO
4160 ;* DECREMENT MODE,
4161 ;*
4162 ;*****
4163 016160 000004 TST15: SCOPE
4164 016162 104413 LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
4165
4166 016164 K1:
4167 016164 170011 SETD ;SET DOUBLE MODE
4168
4169 016166 012700 016440 MOV #KPAT0,R0
4170 016172 172410 LDD (R0),AC0 ;LOAD A DEFAULT
4171 ;PATTERN INTO ACCO
4172 016174 012700 016420 MOV #KBUF0,R0
4173 016200 005003 CLR R3
4174 016202 012737 016272 000004 MOV #KERR0,#ERRVECT
4175
4176 016210 172440 K2: LDD -(R0),AC0 ;TEST INSTRUCTION
4177 016212 005203 K3: INC R3
4178 016214 005203 K4: INC R3
4179
4180 016216 012701 016430 MOV #KDAT00,R1
4181 016222 174011 STD AC0,(R1) ;PICK UP THE RESULT
4182
4183 016224 020027 016430 CMP R0,#KBUF0+10 ;WAS AN AUTO
4184 016230 001001 BNE 1$ ;INCREMENT EXECUTED
4185 016232 000441 BR KERR1
4186
4187 016234 012702 016410 1$: MOV #KDAT10,R2 ;IS DATA CORRECT?
4188 016240 012703 016430 MOV #KDAT00,R3
4189 016244 012704 000004 MOV #4,R4
4190 016250 022223 K5: CMP (R2)+,(R3)+
4191 016252 001401 BEQ K6
4192 016254 000442 BR KERR2
4193 016256 077404 K6: SOB R4,K5
4194
4195 016260 022700 016410 CMP #KBUF0-10,R0 ;WAS R0 DECREMENTED
4196 016264 001401 BEQ K7 ;PROPERLY?
4197 016266 000423 BR KERR1
4198
4199 016270 000467 K7: BR KDONE
4200
4201 ;TRAP TO HERE ON AN ODD ADDRESS ERROR
4202
4203 016272 021627 016212 KERR0: CMP (SP),#K3 ;SEE IF THE ERROR
4204 016276 001405 BEQ K10 ;OCCURRED AT THE
4205 016300 021627 016214 CMP (SP),#K4 ;INSTRUCTION TESTED.
4206 016304 001402 BEQ K10
4207 016306 000137 040232 JMP #CPSPUR
4208
4209 016312 012737 000762 001240 K10: MOV #762,#STMP3 ;REPORT FAILURE IN
    
```

```

4210 016320 012737 000324 001242      MOV      #324,0#STMP4      ;FSRC FLOWS
4211 016326 011637 001236      MOV      (SP),0#STMP2
4212 016332 104055      1$:     ERROR      55
4213 016334 000445      BR       KDONE
4214
4215 016336      KERR1:      ;REPORT, R0
4216 016336 012737 016210 001236      MOV      #K2,0#STMP2      ;INCORRECTLY AFFECTED,
4217 016344 010037 001240      MOV      R0,0#STMP3
4218 016350 012737 016410 001242      MOV      #KDATI0,0#STMP4
4219 016356 104056      1$:     ERROR      56
4220 016360 000433      BR       KDONE
4221
4222      ;REPORT DATA FAILUPE
4223
4224 016362      KERR2:
4225 016362 012737 016210 001236      MOV      #K2,0#STMP2
4226 016370 012737 016410 001240      MOV      #KDATI0,0#STMP3
4227 016376 012737 016430 001242      MOV      #KDAT00,0#STMP4
4228 016404 104057      1$:     ERROR      57
4229 016406 000420      BR       KDONE
4230
4231 016410 052525      KDATI0:  .WORD      052525
4232 016412 114631      KDATI1:      114631
4233 016414 063140      KDATI2:      063140
4234 016416 073567      KDATI3:      073567
4235
4236 016420 010421      KBUF0:      010421
4237 016422 031463      KBUF1:      031463
4238 016424 042104      KBUF2:      042104
4239 016426 021042      KBUF3:      021042
4240
4241 016430 000000      KDAT00:      0
4242 016432 000000      KDAT01:      0
4243 016434 000000      KDAT02:      0
4244 016436 000000      KDAT03:      0
4245
4246 016440 177777      KPAT0:      -1
4247 016442 177777      KPAT1:      -1
4248 016444 177777      KPAT2:      -1
4249 016446 177777      DPAT3:      -1
4250
4251 016450      KDONE:
4252 016450 104412      RSETUP      ;GO INITIALIZE THE FPS AND STACK; AND
4253      ;SEE IF THE USER HAS EXPRESSED
4254      ;THE DESIRE TO CHANGE THE SOFTWARE
4255      ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
4256      ;THE USER TYPED CONTROL G?).
4257
4258
4259      ;*****
4260      ;*TEST 16      FSRC MODE 2, WITH FD=0, TEST
4261      ;*
4262      ;* THIS IS A TEST OF FSRC MODE 2 WITH
4263      ;* FD=0. (AUTO INCREMENT)
4264      ;*
4265      ;*****

```

```

4266 016452 000004          IST16: SCOPE
4267 016454 104413          LPERR          ;SET UP THE LOOP ON ERROR ADDRESS.
4268
4269 016456          L1:
4270 016456 170011          SETD          ;SET DOUBLE MODE
4271
4272 016460 012700 016726          MOV          #LPAT10,R0
4273 016464 172410          LDD          (R0),AC0          ;LOAD AC0
4274
4275 016466 012700 016750          MOV          #LDATI0,R0          ;SET UP THE INPUT
4276 016472 012701 016736          MOV          #LPAT20,R1          ;DATA
4277 016476 012702 000004          MOV          #4,R2
4278
4279 016502 012120          1$: MOV          (R1)+,(R0)+
4280 016504 077202          SOB          R2,1$
4281
4282 016506 012700 016750          MOV          #LDATI0,R0
4283 016512 005003          CLR          R3
4284 016514 170001          SETF          ;CLEAR FD.
4285
4286 016516 172420          L2: LDF          (R0)+,AC0
4287 016520 005203          L3: INC          R3
4288
4289 016522          L4:
4290 016522 170011          SETD          ;SET FD
4291
4292 016524 012701 016762          MOV          #LDATI0,R1
4293 016530 174011          STD          AC0,(R1)          ;PICK UP RESULTS
4294
4295 016532 020027 016754          CMP          R0,#LDATI2          ;WAS R0 INCREMENTED
4296 016536 001401          BEQ          1$          ;CORRECTLY BY 4
4297 016540 000421          BR          LERR1
4298
4299 016542 012737 177777 016754 1$: MOV          #-1,#LDATI2
4300 016550 012737 177777 016756          MOV          #-1,#LDATI3
4301 016556 012702 016750          MOV          #LDATI0,R2          ;IS DATA CORRECT
4302 016562 012703 016762          MOV          #LDATI0,R3
4303 016566 012704 000004          MOV          #4,R4
4304
4305 016572 022223          L5: CMP          (R2)+,(R3)+
4306 016574 001401          BEQ          L6
4307 016576 000427          BR          LERR2
4308 016600 077404          L6: SOB          R4,L5
4309
4310 016602 000473          BR          LDONE
4311
4312 016604          LERR1:          ;REPORT FAILURE
4313 016604 012737 016516 001236          MOV          #L2,@$STMP2          ;R0 NOT INCREMENTED
4314 016612 010037 001240          MOV          R0,@$STMP3          ;BY 4
4315 016616 012737 016754 001242          MOV          #LDATI2,@$STMP4
4316 016624 104060          1$: ERROR          60
4317 016626 000461          BR          LDONE
4318
4319 016630          LERR3:          ;REPORT DATA FAILURE.
4320 016630 012737 016516 001236          MOV          #L2,@$STMP2
4321 016636 012737 016750 001240          MOV          #LDATI0,@$STMP3

```

```

4322 016644 012737 016762 001242      MOV      #LDAT00,@#STMP4
4323 016652 104061      1S:     ERROR  61
4324 016654 000446      BR      LDONE
4325
4326 016656 012702 016736      LERR2:  MOV      #LPAT20,R2      ;DID (BUT FD)
4327 016662 012703 016762      MOV      #LDAT00,R3      ;FAIL,
4328 016666 012704 000004      MOV      #4,R4
4329 016672 022223      1S:     CMP      (R2)+,(R3)+
4330 016674 001355      BNE     LERR3
4331 016676 077403      SOB     R4,1S
4332 016700 012737 016516 001236      MOV      #L2,@#STMP2
4333 016706 012737 016750 001240      MOV      #LDATI0,@#STMP3
4334 016714 012737 016764 001242      MOV      #LDAT01,@#STMP4
4335 016722 104062      2S:     ERROR  62
4336 016724 000422      BR      LDONE
4337
4338 016726 177777      LPAT10: .WORD  -1
4339 016730 177777      LPAT11:      -1
4340 016732 177777      LPAT12:      -1
4341 016734 177777      LPAT13:      -1
4342
4343 016736 052525      LPAT20:      052525
4344 016740 114631      LPAT21:      114631
4345 016742 063142      LPAT22:      063142
4346 016744 073567      LPAT23:      073567
4347 016746 000001      .WORD  000001
4348 016750 000000      LDATI0:      0
4349 016752 000000      LDATI1:      0
4350 016754 000000      LDATI2:      0
4351 016756 000000      LDATI3:      0
4352 016760 000001      .WORD  000001
4353 016762 000000      LDAT00:      0
4354 016764 000000      LDAT01:      0
4355 016766 000000      LDAT02:      0
4356 016770 000000      LDAT03:      0
4357
4358 016772      LDONE:
4359 016772 104412      PSETUP      ;GO INITIALIZE THE FPS AND STACK; AND
4360                                     ;SEE IF THE USER HAS EXPRESSED
4361                                     ;THE DESIRE TO CHANGE THE SOFTWARE
4362                                     ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
4363                                     ;THE USER TYPED CONTROL G?).
4364
4365
4366      ;*****
4367      ;*TEST 17      FSRC MODE 2 WITH GR7, IMMEDIATE MODE, TEST
4368      ;*
4369      ;* THIS IS A TEST OF FSRC MODE 2
4370      ;* USING GR7 (THE PC). THIS IS IMMEDIATE
4371      ;* MODE,
4372      ;*
4373      ;*****
4374 016774 000004      TST17:  SCOPE
4375
4376 016776      M1:
4377 016776 170011      SETD
    
```



```

4378
4379 017000 012700 017272      MOV      #MPAT10,R0
4380 017004 172410      LOD      (R0),AC0      ;LOAD BACKGROUND
4381                                ;PATTERN INTO AC0.
4382 017006 005004      CLR      R4
4383 017010 012737 017232 000004  MOV      #MERR3,@#ERRVECT
4384
4385 017016 172427 000000      M15:    LOD      #0,AC0      ;TEST INSTRUCTION
4386                                .=-2
4387 017020 005204      .WORD   5204
4388 017022 005204      M2:    INC      R4      ;NOTE THAT
4389 017024 005204      M3:    INC      R4      ;005204=INC R4
4390 017026 005204      M4:    INC      R4
4391
4392 017030 020427 000003      CMP      R4,#3      ;SEE IF THE PC
4393 017034 001401      BEQ     1$          ;WAS INCREMENTED
4394 017036 000443      BR      MERR0      ;BY 2 DURING THE
4395                                ;INSTRUCTION. IF
4396                                ;NOT THEN A BAD
4397                                ;CONSTANT WAS GENERATED
4398 017040 012700 017312      1$:    MOV      #MDAT00,R0
4399 017044 174010      STD      AC0,(R0)   ;GET THE DATA
4400
4401 017046 012700 017312      MOV      #MDAT00,R0
4402 017052 022720 005204      CMP      #5204,(R0)+ ;IS THE DATA CORRECT?
4403 017056 001401      BEQ     M5
4404 017060 000451      BR      MERR1
4405 017062 012701 000003      M5:    MOV      #3,R1
4406 017066 005720      M6:    TST      (R0)+
4407 017070 001002      BNE     M7
4408 017072 077103      SOB     R1,M6
4409 017074 000512      BR      MDONE
4410
4411 017076 012700 017312      M7:    MOV      #MDAT00,R0      ;DID (BUT GRM) FAIL?
4412 017102 012701 000004      MOV      #4,R1
4413 017106 022720 005204      M8:    CMP      #5204,(R0)+
4414 017112 001401      BEQ     M9
4415 017114 000433      BR      MERR1
4416 017116 077105      M9:    SOB     R1,M8
4417
4418 017120      MERR2:                                ;REPORT FAILURE
4419 017120 012737 017016 001236      MOV      #M15,@#STMP2      ;OF (BUT GR7)
4420 017126 012737 017302 001240      MOV      #MPAT20,@#STMP3
4421 017134 012737 017312 001242      MOV      #MDAT00,@#STMP4
4422 017142 104063      1$:    ERROR   63
4423 017144 000466      BR      MDONE
4424
4425 017146 012705 017022      MERR0: MOV      #M2,R5      ;REPORT FAILURE
4426 017152 010537 001242      MOV      R5,@#STMP4      ;PC INCREMENTED
4427 017156 162704 000003      SUB      #3,R4
4428 017162 006304      ASL     R4
4429 017164 160405      SUB     R4,R5
4430 017166 010537 001240      MOV      R5,@#STMP3
4431 017172 012737 017016 001236      MOV      #M15,@#STMP2
4432 017200 104064      1$:    ERROR   64
4433 017202 000447      BR      MDONE
    
```

PRINTED IN U.S.A.  
 FORM 1412  
 NOV 76  
 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100

```

4434
4435 017204 MERR1: ;REPORT DATA
4436 017204 012737 017016 001236 MOV #M15,@#STMP2 ;FAILURE
4437 017212 012737 017312 001240 MOV #MDAT00,@#STMP3
4438 017220 012737 017302 001242 MOV #MPAT20,@#STMP4
4439 017226 104066 1S: ERROR 66
4440 017230 000434 BR MDONE
4441 ;TRAP TO HERE THROUGH 4.
4442 017232 032716 000001 MERR3: BIT #1,(SP) ;SEE IF THE
4443 017236 001002 BNE 1S ;TRAP TO 4 OCCURRED
4444 017240 000137 040232 JMP @#CPSPUR ;BECAUSE OF AN
4445 ;ODD ADDRESS
4446 017244 011637 001240 1S: MOV (SP),@#STMP3 ;IF YES REPORT
4447 017250 012737 017022 001242 MOV #M2,@#STMP4 ;BAD CONSTANT
4448 017256 012737 017016 001236 MOV #M15,@#STMP2 ;GENERATED
4449 017264 022626 CMP (SP)+,(SP)+
4450 017266 104065 2S: ERROR 65
4451 017270 000414 BR MDONE
4452
4453 017272 177777 MPAT10: -1
4454 017274 177777 MPAT11: -1
4455 017276 177777 MPAT12: -1
4456 017300 177777 MPAT13: -1
4457
4458 017302 005204 MPAT20: 5204
4459 017304 005204 MPAT21: 5204
4460 017306 005204 MPAT22: 5204
4461 017310 005204 MPAT23: 5204
4462
4463 017312 000000 MDAT00: 0
4464 017314 000000 MDAT01: 0
4465 017316 000000 MDAT02: 0
4466 017320 000000 MDAT03: 0
4467
4468 017322 MDONE:
4469 017322 104412 RSETUP ;GO INITIALIZE THE FPS AND STACK; AND
4470 ;SEE IF THE USER HAS EXPRESSED
4471 ;THE DESIRE TO CHANGE THE SOFTWARE
4472 ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
4473 ;THE USER TYPED CONTROL G?).
4474
4475
4476 ;*****
4477 ;*TEST 20 FSRC MODE 3 TEST
4478 ;*
4479 ;* THIS IS A TEST OF FSRC MODE 3, AUTO INCREMENT
4480 ;* DEFERRED
4481 ;*
4482 ;*****
4483 017324 000004 TST20: SCOPE
4484
4485 017326 N1:
4486 017326 170011 SETD ;SET FD MODE
4487
4488 017330 012700 020010 MOV #NPAT10,R0
4489 017334 172410 LDD (R0),AC0 ;LOAD AC0 WITH A DEFAULT
    
```

```

4490                                     ;PATTERN
4491 017336 012702 017776             MOV    #NPAT20,R0
4492 017342 005003                     CLR    R3
4493 017344 012737 017520 000004     MOV    #NERR0,#ERRVECT           ;IF A FAILURE OCCURS
4494                                     ;IN THE FSRC FLOWS AN
4495                                     ;ODD TRAP TO 4 COULD OCCUR
4496 017352 172430                     N2:    LOD    @(R0)+,AC0         ;TEST INSTRUCTION.
4497 017354 005203                     N3:    INC    R3
4498 017356 005203                     N4:    INC    R3
4499
4500 017360 012701 017756             MOV    #NDAT00,R1
4501 017364 174011                     STD    AC0,(R1)                 ;GET THE DATA
4502
4503 017366 020027 020000             CMP    R0,#NPAT20+2           ;WAS R0 INCREMENTED
4504 017372 001437                     BEQ    N12                      ;BY 2?
4505
4506 017374 020027 020006             N5:    CMP    R0,#NPAT20+10    ;FSRC MODE 2?
4507 017400 001001                     BNE    N6
4508 017402 000506                     BR     NERR1
4509
4510 017404 020027 017766             N6:    CMP    R0,#NPAT20-10    ;FSRC MODE 4?
4511 017410 001001                     BNE    N7
4512 017412 000520                     BR     NERP2
4513
4514 017414 020027 017776             N7:    CMP    R0,#NPAT20
4515 017420 001023                     BNE    N11
4516
4517 017422 012702 017756             MOV    #NDAT00,R2           ;FSRC MODE 0?
4518 017426 012703 000004             MOV    #4,R3
4519 017432 022227 177777             N8:    CMP    (R2)+,#-1
4520 017436 001002                     BNE    N9
4521 017440 077304                     SOB    R3,N8
4522 017442 000510                     BR     NERR3
4523
4524 017444 012702 017756             N9:    MOV    #NDAT00,R2           ;FSRC MODE 1
4525 017450 012703 017776             MOV    #NPAT20,R3
4526 017454 012704 000004             MOV    #4,R4
4527 017460 022223                     N10:   CMP    (R2)+,(R3)+
4528 017462 001002                     BNE    N11
4529 017464 077403                     SOB    R4,N10
4530 017466 000502                     BR     NERR4
4531
4532 017470 000505                     N11:   BR     NERR5
4533
4534 017472 012702 017756             N12:   MOV    #NDAT00,R2           ;DATA CORRECT?
4535 017476 012703 020020             MOV    #NDATI0,R3
4536 017502 012704 000004             MOV    #4,R4
4537 017506 022223                     N13:   CMP    (R2)+,(R3)+
4538 017510 001002                     BNE    N14
4539 017512 077403                     SOB    R4,N13
4540 017514 000545                     BR     NDONE
4541
4542 017516 000504                     N14:   BR     NERR6
4543
4544                                     ;IF AN ODD ADDRESS TRAP OCCURS COME HERE
4545                                     ;TO SEE IF THE FAILURE WAS IN THE FSRC
    
```

```

4546 ; FLOWS
4547
4548 017520 022716 017356 NERR0: CMP #N4,(SP) ;FSRC MODE 6 OR 7?
4549 017524 001412 BEQ NERR10
4550 017526 022716 017354 CMP #N3,(SP)
4551 017532 001402 BEQ 1S
4552 017534 000137 040232 JMP @#CPSPUR
4553 017540 020027 017774 1S: CMP R0,#NPAT20-2 ;FSRC MODE 5?
4554 017544 001407 BEQ NERR11
4555 017546 000137 040232 JMP @#CPSPUR
4556
4557 NERR10: ;WENT TO FSRC
4558 017552 011637 001236 MOV (SP),@#STMP2 ;MODE 6 OR 7.
4559 017556 022626 CMP (SP)+,(SP)+
4560 017560 104067 1S: ERROR 67
4561 017562 000522 BR NDONE
4562
4563 017564 011637 001236 NERR11: MOV (SP),@#STMP2 ;WENT TO FSRC
4564 017570 022626 CMP (SP)+,(SP)+ ;MODE 5.
4565 017572 012737 000627 001244 MOV #627,@#STMP5
4566 017600 012737 000323 001250 MOV #323,@#STMP7
4567 017606 012737 000325 001246 MOV #325,@#STMP6
4568 017614 104070 1S: ERROR 70
4569 017616 000504 BR NDONE
4570 017620 012737 000322 001246 NERR1: MOV #322,@#STMP6 ;FSRC MODE 2.
4571 017626 012737 000627 001244 NERR20: MOV #627,@#STMP5
4572 017631 012737 000323 001250 MOV #323,@#STMP7
4573 017642 012737 017352 001236 MOV #N2,@#STMP2
4574 017650 104071 1S: ERROR 71
4575 017652 000466 BR NDONE
4576 017654 012737 000324 001246 NERR2: MOV #324,@#STMP6 ;FSRC MODE 4
4577 017662 000761 BR NERR20
4578 017664 012737 000320 001246 NERR3: MOV #320,@#STMP6 ;FSRC MODE 0
4579 017672 000755 BR NERR20
4580 017674 012737 000321 001246 NERR4: MOV #321,@#STMP6 ;FSRC MODE 1
4581 017702 000751 BR NERR20
4582
4583 017704 010037 001240 NERR5: MOV R0,@#STMP3 ;R0 NOT
4584 017710 012737 020000 001242 MOV #NPAT20+2,@#STMP4 ;INCREMENTED
4585 017716 012737 017352 001236 MOV #N2,@#STMP2 ;PROPERLY.
4586 017724 104072 1S: ERROR 72
4587 017726 000440 BR NDONE
4588
4589 NERR6: ;DATA FAILURE.
4590 017730 012737 017352 001236 MOV #N2,@#STMP2
4591 017736 012737 017756 001240 MOV #NDAT00,@#STMP3
4592 017744 012737 020020 001242 MOV #NDATI0,@#STMP4
4593 017752 104073 1S: ERROR 73
4594 017754 000425 BR NDONE
4595
4596 017756 000000 NDAT00: .WORD 0
4597 017760 000000 NDAT01: 0
4598 017762 000000 NDAT02: 0
4599 017764 000000 NDAT03: 0
4600
4601 017766 052525 052525 052525 .WORD 52525,52525,52525,52525

```

```

4602 017774 052525
4603 017776 020020 NPAT20: .WORD NDAT10
4604 020000 070707 NPAT21: 070707
4605 020002 070707 NPAT22: 070707
4606 020004 070707 NPAT23: 070707
4607 020006 000001 .WORD 1
4608 020010 177777 NPAT10: .WORD -1
4609 020012 177777 NPAT11: -1
4610 020014 177777 NPAT12: -1
4611 020016 177777 NPAT13: -1
4612
4613 020020 010421 NDAT10: .WORD 010421
4614 020022 021042 NDAT11: 021042
4615 020024 031463 NDAT12: 031463
4616 020026 042104 NDAT13: 042104
4617
4618 020030 NDONE:
4619 020030 104412 RSETUP ;GO INITIALIZE THE FPS AND STACK; AND
4620 ;SEE IF THE USER HAS EXPRESSED
4621 ;THE DESIRE TO CHANGE THE SOFTWARE
4622 ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
4623 ;THE USER TYPED CONTROL G?).
4624
4625
4626 ;*****
4627 ;*TEST 21 FSRC MODE 5 TEST
4628 ;*
4629 ;* THIS IS A TEST OF FSRC MODE 5, AUTO DECREMENT
4630 ;* DEFERRED.
4631 ;*
4632 ;*****
4633 020032 000004 TST21: SCOPE
4634
4635 020034 01:
4636 020034 170011 SETD ;SET FD MODE
4637
4638 020036 012700 020514 MOV #OPAT10,R0
4639 020042 172410 LDD (R0),AC0 ;LOAD AC0 WITH A
4640 ;DEFAULT PATTERN.
4641 020044 012700 020502 MOV #OPAT21,R0
4642 020050 005003 CLR R3
4643 020052 012737 020224 000004 MOV #OERR0,#ERRVEC ;IF A FAILURE
4644 ;OCCURS IN THE FSRC
4645 ;FLOWS AN ODD ADDR.
4646 ;TRAP TO 4 MAY OCCUR.
4647 020060 172450 02: LDD @-(R0),AC0 ;TEST INSTRUCTION
4648 020062 005203 03: INC R3
4649 020064 005203 04: INC R3
4650
4651 020066 012701 020462 MOV #ODAT00,R1
4652 020072 174011 STD AC0,(R1) ;GET THE DATA
4653
4654 020074 020027 020500 CMP R0,#OPAT20 ;WAS R0 DECREMENTED
4655 020100 001436 BEQ 012 ;BY 2?
4656
4657 020102 020027 020512 05: CMP R0,#OPAT21+10 ;FSRC MODE 2

```

```

4658 020106 001001 BNE 06
4659 020110 000505 BR OERR1
4660
4661 020112 020027 020472 06: CMP R0,#OPAT21-10 ;FSRC MODE 4?
4662 020116 001001 BNE 07
4663 020120 000517 BR OERR2
4664
4665 020122 020027 020502 07: CMP R0,#OPAT21
4666
4667 020126 012702 020464 MOV #ODAT01,R2 ;FSRC MODE 0?
4668 020132 012703 000004 MOV #4,R3
4669 020136 022227 177777 08: CMP (R2)+,#-1
4670 020142 001002 BNE 09
4671 020144 077304 SOB R3,08
4672 020146 000510 BR OERR3
4673
4674 020150 012702 020462 09: MOV #ODAT00,R2 ;FSRC MODE 1?
4675 020154 012703 020502 MOV #OPAT21,R3
4676 020160 012704 000004 MOV #4,R4
4677 020164 022223 010: CMP (R2)+,(R3)+
4678 020166 001002 BNE 011
4679 020170 077403 SOB R4,010
4680 020172 000502 BR OERR4
4681
4682 020174 000505 011: BR OERR5
4683
4684 020176 012702 020462 012: MOV #ODAT00,R2 ;DATA CORRECT?
4685 020202 012703 020524 MOV #ODAT10,R3
4686 020206 012704 000004 MOV #4,R4
4687 020212 022223 013: CMP (R2)+,(R3)+
4688 020214 001002 BNE 014
4689 020216 077403 SOB R4,013
4690 020220 000545 BR ODONE
4691
4692 020222 000504 014: BR OERR6
4693
4694 ;IF AN ODD ADDRESS TRAP OCCURS COME
4695 ;HERE TO SEE IF THE FAILURE WAS IN THE
4696 ;FSRC FLOWS:
4697
4698 020224 022716 020064 OERR0: CMP #04,(SP) ;FSRC MODE 6 OR 7?
4699 020230 001412 BEQ OERR10
4700 020232 022716 020062 CMP #03,(SP)
4701 020236 001402 BEQ 1$
4702 020240 000137 040232 JMP @#CPSPUR
4703 020244 020027 020504 1$: CMP R0,#OPAT21+2 ;FSRC MODE 3?
4704 020250 001425 BEQ OERR1
4705 020252 000137 040232 JMP @#CPSPUR
4706
4707 020256 OERR10: ;WENT TO FSRC
4708 020256 011637 001236 MOV (SP),@#STMP2 ;MODE 6 OR 7
4709 020262 022626 CMP (SP)+,(SP)+
4710 020264 104074 1$: ERROR 74
4711 020266 000522 BR ODONE
4712
4713 020270 011637 001240 OERR11: MOV (SP),@#STMP3 ;WENT TO FSRC MODE
    
```

4714	020274	022626				CMP	(SP)+,(SP)+	;3
4715	020276	012737	000627	001244		MOV	#627,@#STMP5	
4716	020304	012737	000325	001250		MOV	#325,@#STMP7	
4717	020312	012737	000323	001246		MOV	#323,@#STMP6	
4718	020320	104075			1s:	ERROR	75	
4719	020322	000504				BR	ODONE	
4720								
4721	020324	012737	000322	001246	OERR1:	MOV	#322,@#STMP6	;FSRC MODE2
4722	020332	012737	000627	001242	OERR20:	MOV	#627,@#STMP4	
4723	020340	012737	000325	001250		MOV	#325,@#STMP7	
4724	020346	012737	020060	001236		MOV	#02,@#STMP2	
4725	020354	104076			1s:	ERROR	76	
4726	020356	000466				BR	ODONE	
4727	020360	012737	000324	001246	OERR2:	MOV	#324,@#STMP6	;FSRC MODE 4
4728	020366	000761				BR	OERR20	
4729	020370	012737	000320	001246	OERR3:	MOV	#320,@#STMP6	;FSRC MODE 0
4730	020376	000755				BR	OERR20	
4731	020400	012737	000321	001246	OERR4:	MOV	#321,@#STMP6	;FSRC MODE 1
4732	020406	000751				BR	OERR20	
4733								
4734	020410	010037	001240		OERR5:	MOV	R0,@#STMP3	;R0 NOT DECREMENTED
4735	020414	012737	020500	001242		MOV	#OPAT20,@#STMP4	;PROPERLY
4736	020422	012737	020064	001236		MOV	#04,@#STMP2	
4737	020430	104077			1s:	ERROR	77	
4738	020432	000440				BR	ODONE	
4739								
4740	020434				OERR6:			;DATA FAILURE
4741	020434	012737	020060	001236		MOV	#02,@#STMP2	
4742	020442	012737	020462	001240		MOV	#ODAT00,@#STMP3	
4743	020450	012737	020524	001242		MOV	#ODATI0,@#STMP4	
4744	020456	104100			1s:	ERROR	100	
4745	020460	000425				BR	ODONE	
4746								
4747	020462	000000			ODAT00:	.WORD	0	
4748	020464	000000			ODAT01:		0	
4749	020466	000000			ODAT02:		0	
4750	020470	000000			ODAT03:		0	
4751								
4752	020472	052525	052525	052525		.WORD	52525,52525,52525	
4753	020500	020524			OPAT20:	.WORD	ODATI0	
4754	020502	070707			OPAT21:	070707		
4755	020504	070707			OPAT22:	070707		
4756	020506	070707			OPAT23:	070707		
4757	020510	070707			OPAT24:	070707		
4758	020512	000001				.WORD	1	
4759	020514	177777			OPAT10:	.WORD	-1	
4760	020516	177777			OPAT11:		-1	
4761	020520	177777			OPAT12:		-1	
4762	020522	177777			OPAT13:		-1	
4763								
4764	020524	073567			ODATI0:	.WORD	73567	
4765	020526	004210			ODATI1:		004210	
4766	020530	114631			ODATI2:		114631	
4767	020532	125252			ODATI3:		125252	
4768								
4769	020534				ODONE:			

```

4770 020534 104412 RSETUP ;GO INITIALIZE THE FPS AND STACK; AND
4771 ;SEE IF THE USER HAS EXPRESSED
4772 ;THE DESIRE TO CHANGE THE SOFTWARE
4773 ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
4774 ;THE USER TYPED CONTROL G?).
4775
4776
4777 ;*****
4778 ;*TEST 22 FSRC MODE 6 TEST
4779 ;*
4780 ;* THIS IS A TEST OF FSRC MODE 6, INDEX MODE
4781 ;*
4782 ;*****
4783 020536 000004 TST22: SCOPE
4784
4785 020540 P1:
4786 020540 170011 SETD ;SET FD MODE
4787
4788 020542 012700 021160 MOV #PPAT10,R0
4789 020546 172410 LDD (R0),AC0 ;LOAD A DEFAULT PATTERN
4790 ;INTO AC0
4791 020550 012737 020656 000004 MOV #PERR0,#ERRVECT ;IF THE (BUT FSRC) FORQ
4792 ;FAILS AN ODD ADDRESS TRAP
4793 020556 012700 020727 MOV #PDAT10-241,R0 ;COULD OCCUR.
4794
4795 020562 172460 000241 P2: LDD 241(R0),AC0
4796 020564 P3=P2+2
4797
4798 020566 012701 021200 P4: MOV #PDAT00,R1
4799 020572 174011 STD AC0,(R1) ;GET THE DATA
4800 020574 012703 000004 MOV #4,R3
4801 020600 012702 021170 MOV #PDAT10,R2
4802 020604 012701 021200 MOV #PDAT00,R1
4803 020610 022221 P5: CMP (R2)+,(R1)+ ;CHECK THE DATA
4804 020612 001007 BNE P6
4805 020614 077303 SOB R3,P5
4806 020616 022700 020727 CMP #PDAT10-241,R0 ;R0 CORRECT?
4807 020622 001401 BEQ 1$
4808 020624 000512 BR PERR21
4809 020626 000137 021210 1$: JMP @#PDONE
4810
4811 020632 012701 021200 P6: MOV #PDAT00,R1
4812 020636 012703 000004 MOV #4,R3
4813 020642 022721 177777 P7: CMP #-1,(R1)+ ;WAS IT FSRC MODE 0?
4814 020646 001401 BEQ P8
4815 020650 000512 BR PERR1
4816 020652 077305 P8: SOB R3,P7
4817 020654 000523 BR PERR2
4818 ;TRAP TO HERE ON AN ODD ADDRESS
4819 020656 021627 020564 PERR0: CMP (SP),#P3
4820 020662 001411 BEQ PERR11
4821 020664 021627 020566 CMP (SP),#P4 ;WAS IT FSRC MODE 7?
4822 020670 001402 BEQ PERR10
4823 020672 000137 040232 JMP @#CPSPUR
4824
4825 020676 012737 000327 001246 PERR10: MOV #327,@#STMP6
    
```



```

4826 020704 000443 BR PERR17
4827 020706 022700 020727 PERR11: CMP #PDATI0-241,R0 ;WAS IT FSRC MODE 1
4828 020712 001004 BNE PERR12
4829 020714 012737 000321 001246 MOV #321,@#STMP6
4830 020722 000434 BR PERR17
4831 020724 022700 020737 PERR12: CMP #PDATI0-241+10,R0 ;WAS IT FSRC MODE 2
4832 020730 001004 BNE PERR13
4833 020732 012737 000322 001246 MOV #322,@#STMP6
4834 020740 000425 BR PERR17
4835 020742 022700 020731 PERR13: CMP #PDATI0-241+2,R0 ;WAS IT FSRC MODE 3
4836 020746 001004 BNE PERR14
4837 020750 012737 000323 001246 MOV #323,@#STMP6
4838 020756 000416 BR PERR17
4839 020760 022700 020717 PERR14: CMP #PDATI0-241-10,R0 ;WAS IT FSRC MODE 4
4840 020764 001004 BNE PERR15
4841 020766 012737 000324 001246 MOV #324,@#STMP6
4842 020774 000407 BR PERR17
4843 020776 022700 020725 PERR15: CMP #PDATI0-241-2,R0 ;WAS IT FSRC MODE 5
4844 021002 001401 BEQ PERR16
4845 021004 000416 BR PERR20
4846 021006 012737 000325 001246 PERR16: MOV #325,@#STMP6
4847
4848 021014 012737 000627 001244 PERR17: MOV #627,@#STMP5 ;REPORT FSRC
4849 021022 012737 000326 001250 MOV #326,@#STMP7 ;FLOWS FAILURE.
4850 021030 011637 001236 MOV (SP),@#STMP2
4851 021034 022626 CMP (SP)+,(SP)+
4852 021036 104101 1s: ERROR 101
4853 021040 000463 BR PDONE
4854
4855 021042 011637 001236 PERR20: MOV (SP),@#STMP2 ;REPORT R0 AFFECTED
4856 021046 022626 CMP (SP)+,(SP)+
4857 021050 000403 BR PERR22
4858 021052 012737 020562 001236 PERR21: MOV #P2,@#STMP2
4859 021060 PERR22:
4860 021060 010037 001240 MOV R0,@#STMP3
4861 021064 012737 020727 001242 MOV #PDATI0-241,@#STMP4
4862 021072 104102 1s: ERROR 102
4863 021074 000445 BR PDONE
4864
4865 021076 PERR1: ;DATA FAILURE.
4866 021076 012737 020562 001236 MOV #P2,@#STMP2
4867 021104 012737 021170 001240 MOV #PDATI0,@#STMP3
4868 021112 012737 021200 001242 MOV #PDATI0,@#STMP4
4869 021120 104104 1s: ERROR 104
4870 021122 000432 BR PDONE
4871
4872 021124 PERR2: ;FSRC FAILURE TO
4873 021124 012737 020562 001236 MOV #P2,@#STMP2 ;MODE 0
4874 021132 012737 000627 001244 MOV #627,@#STMP5
4875 021140 012737 000326 001250 MOV #326,@#STMP7
4876 021146 012737 000320 001246 MOV #320,@#STMP6
4877 021154 104103 1s: ERROR 103
4878 021156 000414 BR PDONE
4879
4880 021160 177777 PPAT10: .WORD -1
4881 021162 177777 PPAT11: -1
    
```

```

4882 021164 177777 PPAT12: -1
4883 021166 177777 PPAT13: -1
4884
4885 021170 010421 PDATI0: .WORD 010421
4886 021172 031463 PDATI1: 031463
4887 021174 052525 PDATI2: 052525
4888 021176 073567 PDATI3: 073567
4889
4890 021200 000000 PDAT00: .WORD 0
4891 021202 000000 PDAT01: 0
4892 021204 000000 PDAT02: 0
4893 021206 000000 PDAT03: 0
4894
4895 021210 PDONE:
4896 021210 104412 RSETUP ;GO INITIALIZE THE FPS AND STACK; AND
4897 ;SEE IF THE USER HAS EXPRESSED
4898 ;THE DESIRE TO CHANGE THE SOFTWARE
4899 ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
4900 ;THE USER TYPED CONTROL G?).
4901
4902
4903 ;*****
4904 ;*TEST 23 FSRC MODE 7 TEST
4905 ;*
4906 ;* THIS IS A TEST OF FSRC MODE 7, INDEX
4907 ;* DEFERRED MODE.
4908 ;*
4909 ;*****
4910 021212 000004 TST23: SCOPE
4911
4912 021214 Q1:
4913 021214 170011 SETD
4914
4915 021216 012700 021650 MOV #QPAT10,R0
4916 021222 172410 LDD (R0),AC0 ;LOAD A DEFAULT
4917 ;PATTERN INTO AC0
4918 021224 012737 021356 000004 MOV #QERR0,@#ERRVECT ;IF THE (BUT FSRC)
4919 ;FORK FAILS AN
4920 ;ODD ADR TRAP COULD
4921 ;OCCUR
4922 021232 012700 021417 MOV #QPAT20-241,R0
4923
4924 021236 172470 000241 Q2: LDD @241(R0),AC0
4925 021240 Q3=Q2+2
4926
4927 021242 012701 021670 Q4: MOV #QDAT00,R1
4928 021246 174011 STD AC0,(R1) ;GET THE DATA
4929
4930 021250 012703 000004 MOV #4,R3
4931 021254 012704 021670 MOV #QDAT00,R4
4932 021260 012705 021700 MOV #QDATI0,R5
4933 021264 022425 Q5: CMP (R4)+,(R5)+ ;CHECK THE DATA
4934 021266 001007 BNE Q6
4935 021270 077303 SOB R3,Q5
4936
4937 021272 022700 021417 CMP #QPAT20-241,R0 ;CHECK R0,

```

PRINTED IN U.S.A. FORM 1413

```

4938 021276 001401          BEQ      1S
4939 021300 000514          BR       QERR21
4940 021302 000137 021710    1S:     JMP      @#QDONE
4941
4942 021306 012701 021670    Q6:     MOV      #QDAT00,R1
4943 021312 012703 000004          MOV      #4,R3
4944 021316 022721 177777    Q7:     CMP      #-1,(R1)+      ;WAS IT FSRC MODE 0?
4945 021322 001002          BNE      Q8
4946 021324 077304          SOB      R3,Q7
4947 021326 000513          BR       QERR2
4948
4949 021330 012701 021660    Q8:     MOV      #QPAT20,R1
4950 021334 012702 021670          MOV      #QDAT00,R2
4951 021340 012703 000004          MOV      #4,R3
4952 021344 022122          Q9:     CMP      (R1)+,(R2)+      ;WAS IT FSRC 6
4953 021346 001401          BEQ      Q10                ;OR DATA FAILURE
4954 021350 000524          BR       QERR1
4955 021352 077304          Q10:    SOB      R3,Q9
4956 021354 000504          BR       QERR3
4957
4958                      ;TRAP TO HERE ON AN ODD ADR FAILURE
4959
4960 021356 021627 020564          QERR0:  CMP      (SP),#P3
4961 021362 000137 040232          JMP      @#CPSPUR
4962
4963 021366 022700 021417          QERR11: CMP      #QPAT20-241,R0      ;WAS IT FSRC
4964 021372 001004          BNE      QERR12            ;MODE 1?
4965 021374 012737 000321 001246          MOV      #321,@#STMP6
4966 021402 000434          BR       QERR17
4967 021404 022700 021427          QERR12: CMP      #QPAT20-241+10,R0      ;WAS IT FSRC
4968 021410 001004          BNE      QERR13            ;MODE 2?
4969 021412 012737 000322 001246          MOV      #322,@#STMP6
4970 021420 000425          BR       QERR17
4971 021422 022700 021421          QERR13: CMP      #QPAT20-241+2,R0      ;WAS IT FSRC
4972 021426 001004          BNE      QERR14            ;MODE 3?
4973 021430 012737 000323 001246          MOV      #323,@#STMP6
4974 021436 000416          BR       QERR17
4975 021440 022700 021407          QERR14: CMP      #QPAT20-241-10,R0      ;WAS IT FSRC
4976 021444 001004          BNE      QERR15            ;MODE 4
4977 021446 012737 000324 001246          MOV      #324,@#STMP6
4978 021454 000407          BR       QERR17
4979
4980 021456 022700 021415          QERR15: CMP      #QPAT20-241-2,R0      ;WAS IT FSRC
4981 021462 001401          BEQ      QERR16            ;MODE 5
4982 021464 000416          BR       QERR20
4983
4984 021466 012737 000325 001246          QERR16: MOV      #325,@#STMP6
4985
4986 021474 012737 000627 001244          QERR17: MOV      #627,@#STMP5      ;REPORT FSRC FAILURE
4987 021502 012737 000327 001250          MOV      #327,@#STMP7
4988 021510 011637 001236          MOV      (SP),@#STMP2
4989 021514 022626          CMP      (SP)+,(SP)+
4990 021516 104105          1S:     ERROR  105
4991 021520 000473          BR       QDONE
4992
4993 021522 011637 001236          QERR20: MOV      (SP),@#STMP2      ;REPORT R0 AFFECTED.

```

```

4994 021526 022626          CMP      (SP)+,(SP)+
4995 021530 000403          BR       QERR2
4996 021532 012737 021236 001236 QERR21: MOV      #Q2,@#STMP2
4997 021540          QERR22:
4998 021540 010037 001240          MOV      R0,@#STMP3
4999 021544 012737 021417 001242          MOV      #QPAT20-241,@#STMP4
5000 021552 104106          1S:     ERROR   106
5001 021554 000455          BR       QDONE
5002
5003 021556 012737 000320 001246 QERR2:  MOV      #320,@#STMP6      ;WENT TO FSRC
5004 021564 000403          BR       QERR4                ;MODE 0
5005 021566 012737 000326 001246 QERR3:  MOV      #326,@#STMP6      ;WENT TO FSRC
5006                                ;MODE 6
5007 021574 012737 000627 001244 QERR4:  MOV      #627,@#STMP5
5008 021602 012737 000327 001250          MOV      #327,@#STMP7
5009 021610 012737 021236 001236          MOV      #Q2,@#STMP2
5010 021616 104107          1S:     ERROR   107
5011 021620 000433          BR       QDONE
5012
5013 021622          QERR1:                                ;DATA FAILURE
5014 021622 012737 021236 001236          MOV      #Q2,@#STMP2
5015 021630 012737 021700 001240          MOV      #QDATI0,@#STMP3
5016 021636 012737 021670 001242          MOV      #QDAT00,@#STMP4
5017 021644 104110          1S:     ERROR   110
5018 021646 000420          BR       QDONE
5019
5020 021650 177777          QPAT10: .WORD  -1
5021 021652 177777          QPAT11:                -1
5022 021654 177777          QPAT12:                -1
5023 021656 177777          QPAT13:                -1
5024
5025 021660 021700          QPAT20: .WORD  QDATI0
5026 021662 052525          QPAT21:                52525
5027 021664 052525          QPAT22:                52525
5028 021666 052525          QPAT23:                52525
5029
5030 021670 000000          QDAT00: .WORD  0
5031 021672 000000          QDAT01:                0
5032 021674 000000          QDAT02:                0
5033 021676 000000          QDAT03:                0
5034
5035 021700 073567          QDATI0: .WORD  073567
5036 021702 052525          QDATI1: .WORD  052525
5037 021704 031463          QDATI2: .WORD  031463
5038 021706 010421          QDATI3: .WORD  010421
5039
5040 021710          QDONE:
5041 021710 104412          RSETUP                ;GO INITIALIZE THE FPS AND STACK; AND
5042                                ;SEE IF THE USER HAS EXPRESSED
5043                                ;THE DESIRE TO CHANGE THE SOFTWARE
5044                                ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
5045                                ;THE USER TYPED CONTROL G?).
5046
5047                                ;*****
5048                                ;*TEST 24      (BUT EZBT Y8),(BUT ENBT) AND (BUT FIUV) TEST
5049                                ;*
    
```

ANHEIM BUSINESS FORMS, INC. HO  
 FORM 1413  
 PRINTED IN U.S.A.

```

5050 ;* THIS IS A TEST OF THE (BUT EZBT Y8) FORK, THE
5051 ;* (BUT ENBT) FORK AND (BUT FIUV) FORK IN THE
5052 ;* LOAD INSTRUCTION FLOWS.
5053 ;* EACH OF THE PATTERNS:
5054 ;*
5055 ;* +NUM
5056 ;* -NUM
5057 ;* -0
5058 ;* IS LOADED TWICE, ONCE WITH AC>0 THEN
5059 ;* WITH AC=0. AFTER EACH LOAD THE FPS IS
5060 ;* CHECK TO INSURE THAT CONTROL WAS PASSED
5061 ;* THROUGH WITH THE FORKS PROPERLY.
5062 ;*
5063 ;*****
5064 021712 000004 TST24: SCOPE
5065 021714 005037 023012 CLR #UFLAG
5066 021720 012700 022742 MOV #UPAT00,R0 ;SET UP AC#0 DATA.
5067 021724 012701 000004 MOV #4,R1
5068 021730 012720 177777 U0: MOV #-1,(R0)+
5069 021734 077103 SOB R1,U0
5070
5071 021736 012737 000033 023014 MOV #033,@#UTMP1
5072 021744 012737 000023 023016 MOV #023,@#UTMP2
5073 021752 012737 022472 000244 MOV #UERR0,@#FPVECT ;IN CASE (BUT FIUV FAILS)
5074 021760 U1:
5075 021760 104413 LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
5076 021762 012700 000200 MOV #200,R0
5077 021766 170100 LDFPS R0
5078 021770 012700 022742 MOV #UPAT00,R0 ;LOAD AC0
5079 021774 172410 LDD (R0),AC0
5080 021776 013737 023014 023020 MOV @#UTMP1,@#UROM1
5081 022004 012737 000001 023022 MOV #001,@#UROM2
5082 022012 012737 000254 023024 MOV #254,@#UROM3
5083
5084 022020 012700 022752 MOV #UPAT10,R0 ;LOAD 0 INTO AC0
5085 022024 -7---0 U2: LDD (R0),AC0
5086 022026 010037 001252 MOV R0,@#STMP10
5087 022032 012737 022024 001236 MOV #U2,@#STMP2
5088
5089 022040 012704 000204 MOV #204,R4 ;SEE IF FPS IS CORRECT
5090 022044 170205 STFPS R5
5091
5092 022046 020405 CMP R4,R5
5093 022050 001402 BEQ U3
5094 022052 000137 022516 JMP @#UERR1
5095
5096 022056 U3:
5097 022056 104413 LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
5098 022060 012700 000200 MOV #200,R0
5099 022064 170100 LDFPS R0
5100
5101 022066 012700 022742 MOV #UPAT00,R0 ;LOAD AC0
5102 022072 172410 LDD (R0),AC0
5103 022074 013737 023016 023020 MOV @#UTMP2,@#UROM1
5104 022102 012737 000003 023022 MOV #003,@#UROM2
5105 022110 012737 000054 023024 MOV #054,@#UROM3
    
```

PRINTED IN U.S.A.

```

5106
5107 022116 012700 022762      MOV      #UPAT20,R0      ;LOAD A POSITIVE NUMBER
5108                                ;INTO AC0
5109 022122 172410      U4:      LDD      (R0),AC0
5110 022124 010037 001252      MOV      R0,#STMP10
5111 022130 012737 022122 001236  MOV      #U4,#STMP2
5112 022136 012704 000200      MOV      #200,R4      ;FPS CORRECT?
5113 022142 170205      STFPS   R5
5114
5115 022144 020405      CMP      R4,R5
5116 022146 001402      BEQ      U5
5117 022150 000137 022602      JMP      @#UERR2
5118 022154      U5:
5119 022154 104413      LPERR   ;SET UP THE LOOP ON ERROR ADDRESS.
5120 022156 012700 000200      MOV      #200,R0
5121 022162 170100      LDFPS   R0
5122 022164 012700 022742      MOV      #UPAT00,R0      ;LOAD AC0
5123 022170 172410      LDD      (R0),AC0
5124 022172 013737 023016 023020      MOV      @#UTMP2,@#UROM1
5125 022200 012737 000403 023022      MOV      #403,@#UROM2
5126 022206 012737 000056 023024      MOV      #056,@#UROM3
5127 022214 012700 022772      MOV      #UPAT30,R0      ;LOAD A NEGATIVE
5128                                ;NUMBER INTO AL0
5129 022220 172410      U6:      LDD      (R0),AC0
5130 022222 010037 001252      MOV      R0,#STMP10
5131 022226 012737 022220 001236  MOV      #U6,#STMP2
5132 022234 012704 000210      MOV      #210,R4      ;FPS CORRECT
5133 022240 170205      STFPS   R5
5134 022242 020405      CMP      R4,R5
5135 022244 001402      BEQ      U7
5136 022246 000137 022602      JMP      @#UERR2
5137 022252      U7:
5138 022252 104413      LPERR   ;SET UP THE LOOP ON ERROR ADDRESS.
5139 022254 012700 000200      MOV      #200,R0
5140 022260 170100      LDFPS   R0
5141 022262 012700 022742      MOV      #UPAT00,R0      ;LOAD AC0
5142 022266 172410      LDD      (R0),AC0
5143 022270 013737 023014 023020      MOV      @#UTMP1,@#UROM1
5144 022276 012737 000401 023022      MOV      #401,@#UROM2
5145 022304 012737 000256 023024      MOV      #256,@#UROM3
5146 022312 012700 023002      MOV      #UPAT40,R0      ;LOAD -0 INTO AC0
5147 022316 172410      U10:    LDD      (R0),AC0
5148 022320 000240      U11:    NOP
5149 022322 010037 001252      MOV      R0,#STMP10
5150 022326 012737 022316 001236  MOV      #U10,#STMP2      ;(BUT FIUV) FAULTS!
5151 022334 012704 000214      MOV      #214,R4      ;SEE IF FPS IS CORRECT.
5152 022340 170205      STFPS   R5
5153 022342 020405      CMP      R4,R5
5154 022344 001402      BEQ      U12
5155 022346 000137 022516      JMP      @#UERR1
5156 022352 005737 023012      U12:    TST      @#UFLAG ;SEE IF ALL THE PATTERNS
5157 022356 001021      BNE     U14      ;HAVE BEEN TEST WITH
5158                                ;BOTH AC NOT EQUAL TO 0 AND AC=0
5159 022360 012700 022742      MOV      #UPAT00,R0      ;IF NOT GO BACK AND
5160 022364 012701 000004      MOV      #4,R1      ;CHECK THEM WITH AC=0
5161 022370 005020      U13:    CLR      (R0)+
    
```

```

5162 022372 077102          SOB      R1,U13
5163 022374 012737 177777 023012      MOV      #-1,@#UFLAG
5164 022402 012737 000233 023014      MOV      #233,@#UTMP1
5165 022410 012737 000223 023016      MOV      #223,@#UTMP2
5166 022416 000137 021760          JMP      @#U1
5167 022422
5168 022422 104413          U14:
5169          LPERF          ;SET UP THE LOOP ON ERROR ADDRESS.
          ;NOW SEE IF A TRAP CAN BE FORCED BY SETTING FIUV AND LOADING -0
5170 022424 012737 022666 000244      MOV      #UERR3,@#FPVECT
5171 022432 012700 004200      MOV      #4200,R0          ;SET FD AND FIUV
5172 022436 170100          LDFPS   R0
5173 022440 012700 022742      MOV      #UPAT00,R0      ;SET UP AC0
5174 022444 172410          LDD     (R0),AC0
5175 022446 012700 023002      MOV      #UPAT40,R0      ;LOAD -0
5176 022452 172410          U15:   LDD     (R0),AC0      ;SHOULD TRAP TO 244
5177 022454 170000          U16:   CFCC
5178 022456 000240          NOP
5179 022460 012737 022452 001236      MOV      #U15,@#STMP2    ;REPORT ERROR.
5180          ;DIDN'T TRAP
5181 022466 104127          1$:   ERROR   127        ;(BUT FIUV) FAILED.
5182 022470 000556          BR      UDONE
5183
5184          ;TRAPPED TO 244. DID (BUT FIUV) FAIL?
5185 022472 021627 022320      UERR0:  CMP     (SP),#U11
5186 022476 001402          BEQ     1$
5187 022500 000137 040200      JMP     @#FPSPUR
5188 022504 011637 001236          1$:   MOV     (SP),@#STMP2
5189 022510 022626          CMP     (SP)+,(SP)+
5190 022512 104126          2$:   ERROR   126
5191 022514 000544          BR      UDONE
5192
5193          ;COME HERE TO ANALYZE FPS ERRORS
5194
5195 022516 032705 000004          UERR1:  BIT     #4,R5
5196 022522 001432          BEQ     UERR20
5197 022524 012737 000443 001244      UERR10: MOV     #443,@#STMP5
5198 022532 013703 023024          MOV     @#UROM3,R3
5199 022536 010337 001250          MOV     R3,@#STMP7
5200 022542 032703 000200          BIT     #200,R3
5201 022546 001403          BEQ     1$
5202 022550 042703 000200          BIC     #200,R3
5203 022554 000402          BR      2$
5204 022556 052703 000200          1$:   BIS     #200,R3
5205 022562 010337 001246          2$:   MOV     R3,@#STMP6
5206 022566 010537 001240          UERR11: MOV     R5,@#STMP3
5207 022572 010437 001242          MOV     R4,@#STMP4
5208 022576 104124          1$:   ERROR   124
5209 022600 000512          BR      UDONE
5210 022602 032705 000004          UERR2:  BIT     #4,R5
5211 022606 001746          BEQ     UERR10
5212 022610 013737 023020 001244      UERR20: MOV     @#UROM1,@#STMP5
5213 022616 013703 023022          MOV     @#UROM2,R3
5214 022622 010337 001250          MOV     R3,@#STMP7
5215 022626 032703 000400          BIT     #400,R3
5216 022632 001403          BEQ     1$
5217 022634 042703 000400          BIC     #400,R3
    
```

```

5218 022640 000402          BR      2S
5219 022642 052703 000400    1S:    BIS      #400,R3
5220 022646 010337 001246    2S:    MOV      R3,#STMP6
5221 022652 010537 001240    UERR21: MOV     R5,#STMP3
5222 022656 010437 001242          MOV     R4,#STMP4
5223 022662 134125          1S:    ERROR   125
5224 022664 000460          BR      UDONE
5225
5226          ;INTERRUPT HERE WHEN FIUV SET AND ATTEMPTED TO LOAD=0
5227 022666 021627 022454    UERR3:  CMP     (SP),#U16
5228 022672 001402          BEQ     1S
5229 022674 000137 040200          JMP     @#FPPSPUR
5230 022700 022626          1S:    CMP     (SP)+,(SP)+
5231 022702 005000          CLR     R0
5232 022704 170300          STST   R0          ;GET FEC.
5233 022706 022700 000014    CMP     #14,R0     ;CORRECT
5234 022712 001001          BNE     UERR4
5235 022714 000444          BR      UDONE
5236 022716 012737 022452 001236  UERR4:  MOV     #U15,#STMP2
5237 022724 012737 000012 001242    MOV     #12,#STMP4
5238 022732 010037 001240          MOV     R0,#STMP3
5239 022736 104130          1S:    ERROR   130
5240 022740 000432          BR      UDONE
5241 022742 000000          UPAT00: .WORD  0
5242 022744 000000          UPAT01:          0
5243 022746 000000          UPAT02:          0
5244 022750 000000          UPAT03:          0
5245
5246 022752 000000          UPAT10: .WORD  0          ;0
5247 022754 000000          UPAT11:          0
5248 022756 000000          UPAT12:          0
5249 022760 000000          UPAT13:          0
5250
5251 022762 010421          UPAT20: .WORD  010421     ;POS NUM
5252 022764 114631          UPAT21:          114631
5253 022766 125252          UPAT22:          125252
5254 022770 177777          UPAT23:          177777
5255
5256 022772 114631          UPAT30:          114631     ;NEG NUM
5257 022774 135673          UPAT31:          135673
5258 022776 146314          UPAT32:          146314
5259 023000 167356          UPAT33:          167356
5260
5261 023002 100000          UPAT40:          100000     ;NEG ZERO
5262 023004 000000          UPAT41:          0
5263 023006 000000          UPAT42:          0
5264 023010 000000          UPAT43:          0
5265
5266 023012 000000          UFLAG:  .WORD  0
5267 023014 000000          UTMP1:          0
5268 023016 000000          UTMP2:          0
5269 023020 000000          UROM1:          0
5270 023022 000000          UROM2:          0
5271 023024 000000          UROM3:          0
5272 023026          UDONE:
5273

```



```

5274
5275 ;*****
5276 ;*TEST 25      ADDF,ADD0,SUBF AND SUBD WITH FSRC=AC=0 TEST
5277
5278 ;*
5279 ;* THIS IS A TEST OF ADD AND SUB WITH FSRC=AC=0
5280 ;*
5281
5282 ;*****
5283 023026 000004 TST25: SCOPE
5284 023030 W1:
5285 023030 104413 LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
5286 023032 012700 000200 MOV #200,R0
5287 023036 170100 LDFPS R0 ;SET DOUBLE MODE
5288 023040 012700 023562 MOV #WPAT00,R0 ;LOAD AC0=:
5289 023044 172410 LDD (R0),AC0
5290 023046 012737 023060 001236 MOV #W2,@#STMP2
5291 023054 012700 023562 MOV #WPAT00,R0
5292 023060 172010 W2: ADDD (R0),AC0 ;TEST INSTRUCTION.
5293 023062 170205 STFPS R5 ;GET FPS
5294 023064 170011 SETD ;SET DOUBLE MODE
5295 023066 012700 023562 MOV #WPAT00,R0
5296 023072 174010 STD AC0,(R0) ;GET THE RESULT
5297 023074 012701 023562 MOV #WPAT00,R1
5298 023100 012702 000004 MOV #4,R2
5299 023104 022021 W3: CMP (R0)+,(R1)+ ;IS RESULT CORRECT
5300 023106 001405 BEQ W4 ;NO
5301
5302 023110 004737 023530 JSR PC,@#WSETUP
5303 023114 104133 1$: ERROR 133
5304 023116 000137 023602 JMP @#WDONE
5305 023122 077210 W4: SOB R2,W3
5306 023124 022705 000204 CMP #204,R5 ;IS FPS CORRECT
5307 023130 001410 BEQ W5 ;NO
5308
5309 023132 012737 000204 001242 MOV #204,@#STMP4
5310 023140 010537 001240 MOV R5,@#STMP3
5311 023144 104137 1$: ERROR 137
5312 023146 000137 023602 JMP @#WDONE
5313 023152 W5:
5314 023152 104413 LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
5315 023154 012700 000200 MOV #200,R0
5316 023160 170100 LDFPS R0 ;SET DOUBLE MODE
5317 023162 012700 023562 MOV #WPAT00,R0 ;LOAD AC0=0
5318 023166 172410 LDD (R0),AC0
5319 023170 012737 023206 001236 MOV #W6,@#STMP2
5320 023176 005000 CLR R0
5321 023200 170100 LDFPS R0 ;GO TO FLOATING MODE
5322 023202 012700 023562 MOV #WPAT00,R0
5323 023206 172010 W6: ADDF (R0),AC0 ;TEST INSTRUCTION
5324 023210 170205 STFPS R5 ;GET FPS
5325 023212 170011 SETD ;RESET TO DOUBLE MODE
5326 023214 012700 023562 MOV #WPAT00,R0
5327 023220 174010 STD AC0,(R0) ;GET THE RESULT
5328 023222 012701 023562 MOV #WPAT00,R1
5329 023226 012702 000004 MOV #4,R2
    
```

5330	023232	022021			W7:	CMP	(R0)+,(R1)+	; WAS THE RESULT
5331	023234	001402				BEQ	W10	; NO. REPORT FAILURE.
5332	023236	104134			1S:	ERROR	134	
5333	023240	000560				BR	WDONE	
5334	023242	077205			W10:	SOB	R2,W7	
5335	023244	022705	000004			CMP	#4,R5	; WAS FPS CORRECT
5336	023250	001407				BEQ	W11	; INCORRECT FPS.
5337								
5338	023252	012737	000004	001242		MOV	#4,@#STMP4	
5339	023260	010537	001240			MOV	R5,@#STMP3	
5340	023264	104140			1S:	ERROR	140	
5341	023266	000545				BR	WDONE	
5342	023270				W11:			
5343	023270	104413				LPERR		; SET UP THE LOOP ON ERROR ADDRESS.
5344	023272	012700	000200			MOV	#200,R0	
5345	023276	170100				LDFPS	R0	; SET DOUBLE MODE
5346	023300	012700	023562			MOV	#WPAT00,R0	; LOAD AC0=0
5347	023304	172410				LDD	(R0),AC0	
5348	023306	012737	023320	001236		MOV	#W12,@#STMP2	
5349	023314	012700	023562			MOV	#WPAT00,R0	
5350	023320	173010			W12:	SUBD	(R0),AC0	; TEST INSTRUCTION
5351	023322	170205				STFPS	R5	; GET FPS
5352	023324	170011				SETD		; SET DOUBLE MODE
5353	023326	012700	023562			MOV	#WPAT00,R0	
5354	023332	174010				STD	AC0,(R0)	; GET THE RESULT
5355	023334	012701	023562			MOV	#WPAT00,R1	
5356	023340	012702	000004			MOV	#4,R2	
5357	023344	022021			W13:	CMP	(R0)+,(R1)+	; IS RESULT CORRECT?
5358	023346	001404				BEQ	W14	; NO.
5359								
5360	023350	004737	023530			JSR	PC,@#WSETUP	
5361	023354	104135			1S:	ERROR	135	
5362	023356	000511				BR	WDONE	
5363	023360	077207			W14:	SOB	R2,W13	
5364	023362	022705	000204			CMP	#204,R5	; IS FPS CORRECT?
5365	023366	001407				BEQ	W15	; NO.
5366								
5367	023370	012737	000204	001242		MOV	#204,@#STMP4	
5368	023376	010537	001240			MOV	R5,@#STMP3	
5369	023402	104141			1S:	ERROR	141	
5370	023404	000476				BR	WDONE	
5371	023406				W15:			
5372	023406	104413				LPERR		; SET UP THE LOOP ON ERROR ADDRESS.
5373	023410	012700	000200			MOV	#200,R0	
5374	023414	170100				LDFPS	R0	; SET DOUBLE MODE
5375	023416	012700	023562			MOV	#WPAT00,R0	; LOAD AC0=0
5376	023422	172410				LDD	(R0),AC0	
5377	023424	012737	023442	001236		MOV	#W16,@#STMP2	
5378	023432	005000				CLR	R0	
5379	023434	170100				LDFPS	R0	; ENTER FLOATING MODE.
5380	023436	012700	023562			MOV	#WPAT00,R0	
5381	023442	173010			W16:	SUBF	(R0),AC0	; TEST INSTRUCTION.
5382	023444	170205				STFPS	R5	; GET FPS
5383	023446	170011				SETD		; RESET TO DOUBLE MODE
5384	023450	012700	023562			MOV	#WPAT00,R0	; GET THE RESULT.
5385	023454	174010				STD	AC0,(R0)	

```

5386 023456 012701 023562          MOV    #WPAT00,R1
5387 023462 012702 000004          MOV    #4,R2
5388 023466 022021          W17:   CMP    (R0)+,(R1)+    ;IS RESULT CORRECT?
5389 023470 001404          BEQ    W20
5390                                ;NO.
5391 023472 004737 023530          JSR    PC,@#WSETUP
5392 023476 104136          1S:   ERROR 136
5393 023500 000440          BR    WDONE
5394 023502 077207          W20:   SOB    R2,W17
5395 023504 022705 000004          CMP    #4,R5    ;IS FPS CORRECT?
5396 023510 001434          BEQ    WDONE
5397                                ;NO
5398 023512 012737 000004 001242          MOV    #4,@#STMP4
5399 023520 010537 001240          MOV    R5,@#STMP3
5400 023524 104142          1S:   ERROR 142
5401 023526 000125          BR    WDONE
5402
5403                                ;SET UP FOR ERROR CALL
5404
5405 023530 012737 023562 001240 WSETUP: MOV    #WPAT00,@#STMP3
5406 023536 012737 023562 001242          MOV    #WPAT00,@#STMP4
5407 023544 012737 023562 001246          MOV    #WPAT00,@#STMP6
5408 023552 012737 023562 001244          MOV    #WPAT00,@#STMP5
5409 023560 000207          RTS    PC
5410 023562 000000          WPAT00: .WORD 0
5411 023564 000000          WPAT01:      0
5412 023566 000000          WPAT02:      0
5413 023570 000000          WPAT03:      0
5414
5415 023572 000000          WDAP00: .WORD 0
5416 023574 000000          WDAT01:      0
5417 023576 000000          WDAT02:      0
5418 023600 000000          WDAT03:      0
5419
5420 023602          WDONE:
5421 023602 104412          RSETUP    ;GO INITIALIZE THE FPS AND STACK; AND
5422                                ;SEE IF THE USER HAS EXPRESSED
5423                                ;THE DESIRE TO CHANGE THE SOFTWARE
5424                                ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
5425                                ;THE USER TYPED CONTROL G?).
5426
5427
5428                                ;*****
5429                                ;*TEST 26      ADDD AND SUB WITH FSRC=0
5430                                ;*
5431                                ;* THIS IS A TEST OF ADD AND SUB WITH FSRC=0.
5432                                ;*
5433
5434                                ;*****
5435 023604 000004          IST26: SCOPE
5436 023606          X1:
5437 023606 104413          ↓PERR    ;SET UP THE LOOP ON ERROR ADDRESS.
5438 023610 012700 000200          MOV    #200,R0
5439 023614 170100          LDFPS  R0    ;SET DOUBLE MODE
5440 023616 012700 024346          MOV    #XPAT00,R0 ;SET ACQ TO POSITIVE
5441 023622 010037 024334          MOV    R0,@#XTMP ;NUMBER #0
    
```

5442	023626	172410			LDD	(R0),AC0	
5443	023630	012737	023642	001236	MOV	#X2,#STMP2	
5444	023636	012700	024356		MOV	#XPAT10,R0	;FSRC=0
5445	023642	172010		X2:	ADDD	(R0),AC0	;TEST INSTRUCTION
5446	023644	170205			STFPS	R5	
5447	023646	170011			SETD		
5448	023650	012700	024336		MOV	#XDAT00,R0	;GET RESULT.
5449	023654	174010			STD	AC0,(R0)	
5450	023656	012701	024346		MOV	#XPAT00,R1	
5451	023662	012702	000004		MOV	#4,R2	
5452	023666	022021		X3:	CMP	(R0)+,(R1)+	;IS RESULT CORRECT?
5453	023670	001401			BEQ	X4	
5454	023672	000553			BR	XERR1	
5455	023674	077204		X4:	SOB	R2,X3	
5456	023676	012704	000200		MOV	#200,R4	
5457	023702	020405			CMP	R4,R5	;IS FPS CORRECT?
5458	023704	001402			BEQ	X5	
5459	023706	000137	024304		JMP	#XERR2	
5460	023712			X5:			
5461	023712	104413			LPERR		;SET UP THE LOOP ON ERROR ADDRESS.
5462	023714	012700	000200		MOV	#200,R0	
5463	023720	170100			LDFPS	R0	;SET DOUBLE MODE
5464	023722	012700	024366		MOV	#XPAT20,R0	;SET AC0 TO
5465	023726	010037	024334		MOV	R0,#XTMP	;NEGATIVE NUMBER
5466	023732	172410			LDD	(R0),AC0	
5467	023734	012737	023746	001236	MOV	#X6,#STMP2	
5468	023742	012700	024356		MOV	#XPAT10,R0	;FSRC=0
5469	023746	172010		X6:	ADDD	(R0),AC0	;TEST INSTRUCTION
5470	023750	170205			STFPS	R5	
5471	023752	170011			SETD		
5472	023754	012700	024336		MOV	#XDAT00,R0	;GET RESULT
5473	023760	174010			STD	AC0,(R0)	
5474	023762	012701	024366		MOV	#XPAT20,R1	
5475	023766	012702	000004		MOV	#4,R2	
5476	023772	022021		X7:	CMP	(R0)+,(R1)+	;IS RESULT CORRECT?
5477	023774	001401			BEQ	X10	
5478	023776	000511			BR	XERR1	
5479	024000	077204		X10:	SOB	R2,X7	
5480	024002	012704	000210		MOV	#210,R4	
5481	024006	020405			CMP	R4,R5	;IS FPS CORRECT?
5482	024010	001401			BEQ	X11	
5483	024012	000534			BR	XERR2	
5484	024014			X11:			
5485	024014	104413			LPERR		;SET UP THE LOOP ON ERROR ADDRESS.
5486	024016	012700	000200		MOV	#200,R0	
5487	024022	170100			LDFPS	R0	;SET DOUBLE MODE
5488	024024	012700	024346		MOV	#XPAT00,R0	;SET AC0 TO NON-ZERO
5489	024030	010037	024334		MOV	R0,#XTMP	;POSITIVE NUMBER
5490	024034	172410			LDD	(R0),AC0	
5491	024036	012737	024050	001236	MOV	#X12,#STMP2	
5492	024044	012700	024356		MOV	#XPAT10,R0	;FSRC=0
5493	024050	173010		X12:	SUBD	(R0),AC0	;TEST INSTRUCTION
5494	024052	170205			STFPS	R5	
5495	024054	170011			SETD		
5496	024056	012700	024336		MOV	#XDAT00,R0	;GET RESULT
5497	024062	174010			STD	AC0,(R0)	

```

5498 024064 012701 024346      MOV      #XPAT00,R1
5499 024070 012702 000004      MOV      #4,R2
5500 024074 022021      X13:    CMP      (R0)+,(R1)+      ;IS RESULT CORRECT?
5501 024076 001401      BEQ      X14
5502 024100 000463      BR       XERR3
5503 024102 077204      X14:    SOB      R2,X13
5504 024104 012704 000200      MOV      #200,R4      ;IS FPS CORRECT?
5505 024110 020405      CMP      R4,R5
5506 024112 001401      BEQ      X15
5507 024114 000501      BR       XERR4
5508 024116      X15:
5509 024116 104413      LPERR           ;SET UP THE LOOP ON ERROR ADDRESS,
5510 024120 012700 000200      MOV      #200,R0
5511 024124 170100      LDFPS  R0      ;SET DOUBLE MODE
5512 024126 012700 024366      MOV      #XPAT20,R0   ;SET AC0=A NEGATIVE
5513 024132 010037 024334      MOV      R0,#XTMP    ;NUMBER
5514 024136 172410      LDD      (R0),AC0
5515 024140 012737 024152 001236      MOV      #X16,#STMP2
5516 024146 012700 024356      MOV      #XPAT10,R0   ;FSRC=0
5517 024152 173010      X16:    SUBD     (R0),AC0   ;TEST INSTRUCTION.
5518 024154 170205      STFPS  R5
5519 024156 170011      SETD
5520 024160 012700 024336      MOV      #XDAT00,R0   ;GET RESULT
5521 024164 174010      STD     AC0,(R0)
5522 024166 012701 024366      MOV      #XPAT20,R1
5523 024172 012702 000004      MOV      #4,R2
5524 024176 022021      X17:    CMP      (R0)+,(R1)+      ;IS RESULT CORRECT?
5525 024200 001401      BEQ      X20
5526 024202 000422      BR       XERR3
5527 024204 077204      X20:    SOB      R2,X17
5528 024206 012704 000210      MOV      #210,R4      ;IS FPS CORRECT?
5529 024212 020405      CMP      R4,R5
5530 024214 001401      BEQ      X21
5531 024216 000440      BR       XERR4
5532 024220 000466      X21:    BR       XDONE
5533
5534      ;REPORT DATA ERRORS
5535
5536 024222 012737 024356 001240 XERR1:  MOV      #XPAT10,#STMP3
5537 024230 013737 024334 001242      MOV      #XTMP,#STMP4
5538 024236 012737 024336 001244      MOV      #XDAT00,#STMP5
5539 024244 104143      1$:    ERROR  143
5540 024246 000453      BR       XDONE
5541 024250 012737 024356 001240 XERR3:  MOV      #XPAT10,#STMP3
5542 024256 013737 024334 001242      MOV      #XTMP,#STMP4
5543 024264 012737 024336 001244      MOV      #XDAT00,#STMP5
5544 024272 013737 024334 001246      MOV      #XTMP,#STMP6
5545 024300 104144      1$:    ERROR  144
5546 024302 000435      BR       XDONE
5547
5548      ;REPORT FPS ERRORS
5549
5550      XERR2:
5551 024304 010537 001240      MOV      R5,#STMP3
5552 024310 010437 001242      MOV      R4,#STMP4
5553 024314 104145      1$:    ERROR  145
    
```

```

5554 024316 000427          BR      XDONE
5555 024320          XERR4:
5556 024320 010537 001240      MOV     R5,0#STMP3
5557 024324 010437 001242      MOV     R4,0#STMP4
5558 024330 104146          1S:    ERROR  146
5559 024332 000421          BR      XDONE
5560 024334 000000          XTMP:   .WORD  0
5561 024336 000000          XDAT00: .WORD  0
5562 024340 000000          XDAT01:      0
5563 024342 000000          XDAT02:      0
5564 024344 000000          XDAT03:      0
5565
5566 024346 010421          XPAT00: .WORD  010421
5567 024350 021042          XPAT01:      021042
5568 024352 031463          XPAT02:      031463
5569 024354 042104          XPAT03:      042104
5570
5571 024356 000000          XPAT10: .WORD  0
5572 024360 000000          XPAT11:      0
5573 024362 000000          XPAT12:      0
5574 024364 000000          XPAT13:      0
5575 024366 104210          XPAT20: .WORD  104210
5576 024370 114631          XPAT21:      114631
5577 024372 125252          XPAT22:      125252
5578 024374 135673          XPAT23:      135673
5579
5580 024376          XDONE:
5581 024376 104412          RSETUP      ;GO INITIALIZE THE FPS AND STACK; AND
5582                                     ;SEE IF THE USER HAS EXPRESSED
5583                                     ;THE DESIRE TO CHANGE THE SOFTWARE
5584                                     ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
5585                                     ;THE USER TYPED CONTROL G?).
5586
5587                                     ;*****
5588                                     ;*TEST 27      SUBD WITH AC=0 TEST
5589                                     ;*
5590                                     ;* THIS IS A TEST OF SUBD WITH AC=0.  BOTH POSITIVE
5591                                     ;* AND NEGATIVE FSRC'S ARE TRIED.
5592                                     ;*
5593
5594                                     ;*****
5595 024400 000004          TST27:  SCOPE
5596 024402 005037 024732          CLR     0#YFLAG
5597 024406 012737 024752 024734      MOV     #YPAT00,0#YTMP1 ;P
5598 024414 012737 024762 024736      MOV     #YPAT10,0#YTMP2 ;N
5599 024422 012737 000210 024740      MOV     #210,0#YTMP3
5600 024430
5601 024430 104413          Y1:      LPERF      ;SET UP THE LOOP ON ERROR ADDRESS.
5602 024432 012700 000200          MOV     #200,R0
5603 024436 170100          LDFPS   R0      ;SET DOUBLE MODE
5604 024440 012700 024772          MOV     #YPAT20,R0  ;SET AC0=0
5605 024444 172410          LDD     (R0),AC0
5606 024446 013700 024734          MOV     0#YTMP1,R0
5607 024452 173010          Y2:      SUBD     (R0),AC0  ;TEST INSTRUCTION
5608 024454 170205          STFPS   R5
5609 024456 170011          SETD

```

FORM 1413 PRINTED IN U.S.A.

```

5610 024460 012700 024742      MOV      #YDAT00,R0      ;GET RESULT
5611 024464 174010      STD      AC0,(R0)
5612 024466 012702 000004      MOV      #4,R2
5613 024472 013701 024736      MOV      @#YTMP2,R1      ;CHECK RESULT.
5614 024476 022021      Y3:     CMP      (R0)+,(R1)+
5615 024500 001026      BNE     Y6
5616 024502 077203      SOB     R2,Y3
5617 024504 023705 024740      CMP      @#YTMP3,R5      ;FPS CORRECT?
5618 024510 001401      BEQ     Y4
5619 024512 000475      BR      YERR3
5620 024514 005737 024732      Y4:     TST      @#YFLAG      ;FINISHED TEST?
5621 024520 001015      BNE     Y5
5622 024522 012737 177777 024732      MOV      #-1,@#YFLAG
5623 024530 012737 024762 024734      MOV      #YPAT10,@#YTMP1
5624 024536 012737 024752 024736      MOV      #YPAT00,@#YTMP2
5625 024544 012737 000200 024740      MOV      #200,@#YTMP3
5626 024552 000726      BR      Y1
5627 024554 000512      Y5:     BR      YDONE
5628 024556 012702 000004      Y6:     MOV      #4,R2
5629 024562 012700 024734      MOV      @#YTMP1,R0      ;DID XOR OF SIGN BIT
5630 024566 012701 024742      MOV      #YDAT00,R1      ;FAIL?
5631 024572 022021      Y7:     CMP      (R0)+,(R1)+
5632 024574 001002      BNE     YERR1
5633 024576 077203      SOB     R2,Y7
5634 024600 000421      BR      YERR2
5635 024602      YERR1:      ;DATA FAILURE
5636 024602 012737 024452 001236      MOV      #Y2,@#STMP2
5637 024610 013737 024734 001240      MOV      @#YTMP1,@#STMP3
5638 024616 012737 024772 001242      MOV      #YPAT20,@#STMP4
5639 024624 012737 024742 001244      MOV      #YDAT00,@#STMP5
5640 024632 013737 024736 001246      MOV      @#YTMP2,@#STMP6
5641 024640 104147      1S:     ERROR    147
5642 024642 000457      BR      YDONE
5643 024644      YERR2:      ;XOR OF SIGN BIT
5644 024644 012737 024452 001236      MOV      #Y2,@#STMP2      ;FAILED
5645 024652 013737 024734 001240      MOV      @#YTMP1,@#STMP3
5646 024660 012737 024772 001242      MOV      #YPAT20,@#STMP4
5647 024666 012737 024742 001244      MOV      #YDAT00,@#STMP5
5648 024674 013737 024736 001246      MOV      @#YTMP2,@#STMP6
5649 024702 104150      1S:     ERROR    150
5650 024704 000436      BR      YDONE
5651 024706      YERR3:      ;FPS WRONG.
5652 024706 012737 024452 001236      MOV      #Y2,@#STMP2
5653 024714 010537 001240      MOV      R5,@#STMP3
5654 024720 013737 024740 001242      MOV      @#YTMP3,@#STMP4
5655 024726 104151      1S:     ERROR    151
5656 024730 000424      BR      YDONE
5657
5658 024732 000000      YFLAG:   .WORD    0
5659 024734 000000      YTMP1:   0
5660 024736 000000      YTMP2:   0
5661 024740 000000      YTMP3:   0
5662
5663 024742 000000      YDAT00:  .WORD    0
5664 024744 000000      YDAT01:  0
5665 024746 000000      YDAT02:  0
    
```

PRINTED FORMS DIV 100  
 FORM 100  
 PRINTED IN U.S.A.

```

5666 024750 000000 YDAT03: 0
5667
5668 024752 063146 YPAT00: 063146
5669 024754 052525 YPAT01: 052525
5670 024756 042104 YPAT02: 042104
5671 024760 167356 YPAT03: 167356
5672
5673 024762 163146 YPAT10: 163146
5674 024764 052525 YPAT11: 052525
5675 024766 042104 YPAT12: 042104
5676 024770 167356 YPAT13: 167356
5677
5678 024772 000000 YPAT20: 0
5679 024774 000000 YPAT21: 0
5680 024776 000000 YPAT22: 0
5681 025000 000000 YPAT23: 0
5682
5683 025002 YDONE:
5684 025002 104412 RSETUP ;GO INITIALIZE THE FPS AND STACK; AND
5685 ;SEE IF THE USER HAS EXPRESSED
5686 ;THE DESIRE TO CHANGE THE SOFTWARE
5687 ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
5688 ;THE USER TYPED CONTROL G?).
5689 ;*****
5690 ;*TEST 30 ADDD WITH AC=0 TEST
5691
5692 ;@
5693 ;@ THIS IS A TEST OF ADDD WITH AC=0. BOTH
5694 ;* POSITIVE AND NEGATIVE FSRC'S ARE TRIED.
5695 ;*
5696
5697 ;*****
5698 025004 000004 TST30: SCOPE
5699 025006 005067 000224 CLR ZFLAG
5700 025012 012737 025254 025240 MOV #ZPAT00,0#ZTMP1 ;P
5701 025020 012737 000200 025242 MOV #200,0#ZTMP2
5702 025026
5703 025026 104413 Z1: LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
5704 025030 012700 000200 MOV #200,R0
5705 025034 170100 LDFPS R0 ;SET DOUBLE MODE
5706 025036 012700 025274 MOV #ZPAT20,R0 ;SET AC0=0
5707 025042 172410 LDD (R0),AC0
5708 025044 013700 025240 MOV 0#ZTMP1,R0
5709 025050 172010 Z2: ADDD (R0),AC0 ;TEST INSTRUCTION
5710 025052 170205 STFPS R5
5711 025054 170011 SETD
5712 025056 012700 025244 MOV #ZDAT00,R0 ;GET RESULT
5713 025062 174010 STD AC0,(R0)
5714 025064 012702 000004 MOV #4,R2
5715 025070 013701 025240 MOV 0#ZTMP1,R1 ;RESULT CORRECT?
5716 025074 022021 Z3: CMP (R0)+,(R1)+
5717 025076 001401 BEQ Z4
5718 025100 000423 BR ZERR1
5719 025102 077204 Z4: SOB R2,Z3
5720 025104 023705 025242 CMP 0#ZTMP2,R5 ;FPS CORRECT?
5721 025110 001401 BEQ Z5
  
```

FORM 1413  
 PRINTED IN U.S.A.  
 MICRO BUSINESS FORMS, INC. HO



```

5722 025112 000437          BR      ZERR2
5723 025114 005737 025236  Z5:    IST      @#ZFLAG      ;FINISHED TEST?
5724 025120 001012          BNE     Z6
5725 025122 012737 177777 025236  MOV     #-1,@#ZFLAG
5726 025130 012737 025264 025240  MOV     #ZPAT10,@#ZTMP1
5727 025136 012737 000210 025242  MOV     #210,@#ZTMP2
5728 025144 000730          BR      Z1
5729 025146 000456          Z6:    BR      ZDONE
5730 025150          ZERR1:          ;DATA FAILURE
5731 025150 012737 025050 001236  MOV     #Z2,@#STMP2
5732 025156 013737 025240 001240  MOV     @#ZTMP1,@#STMP3
5733 025164 012737 025274 001242  MOV     #ZPAT20,@#STMP4
5734 025172 012737 025244 001244  MOV     #ZDAT00,@#STMP5
5735 025200 013737 025240 001246  MOV     @#ZTMP1,@#STMP6
5736 025206 104152          1S:    ERROR    152
5737 025210 000435          BR      ZDONE
5738 025212          ZERR2:
5739 025212 012737 025050 001236  MOV     #Z2,@#STMP2
5740 025220 010537 001240  MOV     R5,@#STMP3
5741 025224 013737 025242 001242  MOV     @#ZTMP2,@#STMP4
5742 025232 104153          1S:    ERROR    153
5743 025234 000423          BR      ZDONE
5744
5745 025236 000000          ZFLAG:  .WORD    0
5746 025240 000000          ZTMP1:          0
5747 025242 000000          ZTMP2:          0
5748
5749 025244 000000          ZDAT00: .WORD    0
5750 025246 000000          ZDAT01:          0
5751 025250 000000          ZDAT02:          0
5752 025252 000000          ZDAT03:          0
5753
5754 025254 031463          ZPAT00:          031463
5755 025256 010421          ZPAT01:          010421
5756 025260 146314          ZPAT02:          146314
5757 025262 156735          ZPAT03:          156735
5758
5759 025264 156735          ZPAT10:          156735
5760 025266 167356          ZPAT11:          167356
5761 025270 135673          ZPAT12:          135673
5762 025272 146314          ZPAT13:          146314
5763
5764 025274 000000          ZPAT20:          0
5765 025276 000000          ZPAT21:          0
5766 025300 000000          ZPAT22:          0
5767 025302 000000          ZPAT23:          0
5768
5769 025304          ZDONE:
5770 025304 104412          RSETUP          ;GO INITIALIZE THE FPS AND STACK; AND
5771                                     ;SEE IF THE USER HAS EXPRESSED
5772                                     ;THE DESIRE TO CHANGE THE SOFTWARE
5773                                     ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
5774                                     ;THE USER TYPED CONTROL G?).
5775
5776
5777 ;*****

```

01-NOV-76 21:09 PDP 11/34 FPP DIAGNOSTIC PART 1 MACY11 27(1006) PAGE 104

```

5778 ;*TEST 31 ADDF AND ADDD WITH E(AC)=E(FSRC) TEST AND (BUT FT) TEST
5779 ;*
5780 ;* THIS IS A TEST OF THE ADD INSTRUCTION WITH THE
5781 ;* OPERANDS HAVING EQUAL EXPONENTS. THE (BUT FT)
5782 ;* FORK IN THE ROUND/TRUNK FLOWS IS ALSO TESTED.
5783 ;*
5784 ;*****
5785 025306 000004 TST31: SCOPE
5786 025310 AA1:
5787 025310 104413 LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
5788 025312 012700 003240 MOV #3240,R0 ;SET FIV FIV FD AND FT
5789 025316 170100 LDFPS R0 ;IN CASE THE OVER/UNDER
5790 025320 012737 025670 000244 MOV #AAERR0,#FPVECT ;FLOWS IN TRAP WILL
5791 025326 012700 026246 MOV #AAPAT0,R0 ;OCCUR
5792 ;SET UP AC0
5793 025332 172410 LDD (R0),AC0 ;OPERAND
5794 025334 012737 025346 001236 MOV #AA2,#STMP2
5795 025342 012700 026256 MOV #AAPAT1,R0
5796 025346 172010 AA2: ADDD (R0),AC0 ;TEST INSTRUCTION
5797 ;SHOULD TRUNCATE
5798 025350 012700 026236 AA3: MOV #AADAT0,R0
5799 025354 174010 STD AC0,(R0) ;GET THE RESULT
5800 025356 012701 026266 MOV #AAPAT2,R1
5801 025362 012702 000004 MOV #4,R2
5802 025366 022021 AA4: CMP (R0)+,(R1)+ ;CORRECT?
5803 025370 001414 BEQ AA7
5804 025372 012700 026276 MOV #AAPAT3,R0 ;DID (BUT FT) FAIL
5805 025376 012701 026236 MOV #AADAT0,R1
5806 025402 012702 000004 MOV #4,R2
5807 025406 022021 AA5: CMP (R0)+,(R1)+
5808 025410 001401 BEQ AA6
5809 025412 000561 BR AAERR1 ;DATA ERROR
5810 025414 077204 AA6: SOB R2,AA5
5811 025416 000137 026012 JMP #AAERR2 ;(BUT FT) ERROR
5812 025422 077217 AA7: SOB R2,AA4
5813
5814 ;NOW TEST DOUBLE FLOATING ROUND MODE.
5815
5816 025424 AA10:
5817 025424 104413 LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
5818 025426 012700 003200 MOV #3200,R0 ;SET FD FIV FIV. FT=0
5819 025432 170100 LDFPS R0
5820 025434 012700 026246 MOV #AAPAT0,R0
5821 025440 172410 LDD (R0),AC0 ;SET UP AC0 OPERAND
5822 025442 012737 025454 001236 MOV #AA11,#STMP2
5823 025450 012700 026256 MOV #AAPAT1,R0
5824 025454 172010 AA11: ADDD (R0),AC0 ;TEST INSTRUCTION
5825 ;SHOULD ROUND
5826 025456 012700 026236 AA12: MOV #AADAT0,R0
5827 025462 174010 STD AC0,(R0) ;GET THE RESULT
5828 025464 012701 026276 MOV #AAPAT3,R1
5829 025470 012702 000004 MOV #4,R2
5830 025474 022021 AA13: CMP (R0)+,(R1)+ ;CORRECT?
5831 025476 001425 BEQ AA20
5832 025500 012700 026266 MOV #AAPAT2,R0 ;DID (BUT FT) FAIL?
5833 025504 012701 026236 MOV #AADAT0,R1
    
```

PRINTED IN U.S.A.

```

5834 025510 012702 000004          MOV      #4,R2
5835 025514 022021          AA14:   CMP      (R0)+,(R1)+
5836 025516 001413          BEQ      AA17
5837 025520 012700 026306          MOV      #AAPAT4,R0          ;WAS THE FLOATING
5838 025524 012701 026236          MOV      #AADATO,R1          ;CONSTANT USED
5839 025530 012702 000004          MOV      #4,R2          ;INSTEAD OF THE
5840 025534 022021          AA15:   CMP      (R0)+,(R1)+          ;DOUBLE CONSTANT
5841 025536 001401          BEQ      AA16          ;IN THE ROUND
5842 025540 000542          BR      AAERR3 ; FLOWS?
5843 025542 077204          AA16:   SOB      R2,AA15          ;DATA ERROR
5844 025544 000544          BR      AAERR4 ; CONSTANT ERROR
5845 025546 077216          AA17:   SOB      R2,AA14
5846 025550 000560          BR      AAERR5 ; (BUT FT) ERROR
5847 025552 077230          AA20:   SOB      R2,AA13
5848
5849          ;NOW TEST ADDF WITH FT=0, ROUND MODE
5850
5851 025554          AA21:
5852 025554 104413          LPERR          ;SET UP THE LOOP ON ERROR ADDRESS,
5853 025556 012700 003200          MOV      #3200,R0          ;FIV=1, FIV=1, FT=0
5854 025562 170100          LDFPS      R0
5855 025564 012700 026246          MOV      #AAPAT0,R0          ;LOAD AC0 OPERAND
5856 025570 172410          LDD      (R0),AC0
5857 025572 170001          SETF          ;ENTER FLOATING MODE
5858 025574 012737 025606 001236          MOV      #AA22,@$TMP2
5859 025602 012700 026316          MOV      #AAPAT5,R0
5860 025606 172010          AA22:   ADDF      (R0),AC0          ;TEST INSTRUCTION
5861          ;SHOULD ROUND
5862 025610          AA23:
5863 025610 170011          SETD          ;RESET TO DOUBLE
5864          ;MODE
5865 025612 012700 026236          MOV      #AADATO,R0          ;GET THE RESULT
5866 025616 174010          STD      AC0,(R0)
5867 025620 012701 026326          MOV      #AAPAT6,R1          ;CORRECT?
5868 025624 012702 000002          MOV      #2,R2
5869 025630 022021          AA24:   CMP      (R0)+,(R1)+
5870 025632 001413          BEQ      AA27
5871 025634 012700 026266          MOV      #AAPAT2,R0          ;WAS THE DOUBLE
5872 025640 012701 026236          MOV      #AADATO,R1          ;CONSTANT USED INSTEAD
5873 025644 012702 000002          MOV      #2,R2          ;OF THE FLOATING
5874 025650 022011          AA25:   CMP      (R0)+,(R1)          ;CONSTANT IN THE
5875 025652 001401          BEQ      AA26          ;ROUND FLOWS?
5876 025654 000534          BR      AAERR6 ; DATA ERROR
5877 025656 077204          AA26:   SOB      R2,AA25
5878 025660 000550          BR      AAERR7 ; CONSTANT ERROR
5879 025662 077216          AA27:   SOB      R2,AA24
5880 025664 000137 026336          JMP      @#AADONE
5881
5882          ;COME HERE IF A TRAP OCCURS TO 244.
5883
5884 025670 013700 001236          AAERR0: MOV      @$TMP2,R0          ;SEE IF THE TRAP WAS
5885 025674 005720          TST      (R0)+          ;AT A TEST INSTRUCTION
5886 025676 020016          CMP      R0,(SP)
5887 025700 001402          BEQ      1$
5888 025702 000137 040200          10$:   JMP      @#FPSPUR
5889 025706          1$:

```

```

5890 025706 170300 STST R0 ;GET FEC
5891 025710 022027 000010 CMP R0,#10
5892 025714 001405 BEQ 20$ ;OVERFLOW
5893 025716 020027 000012 CMP R0,#12
5894 025722 001410 BEQ 30$ ;UNDERFLOW
5895 025724 000766 BR 10$
5896 025726 025730 20$
5897 025730 011637 001236 20$: MOV (SP),@#STMP2 ;REPORT OVERFLOW ERROR
5898 025734 022626 CMP (SP)+,(SP)+
5899 025736 104154 21$: ERROR 154
5900 025740 000137 026336 25$: JMP @#AADONE
5901 025744 011637 001236 30$: MOV (SP),@#STMP2 ;REPORT UNDERFLOW
5902 025750 022626 CMP (SP)+,(SP)+ ;ERFOR
5903 025752 104155 31$: ERROR 155
5904 025754 000771 BR 25$
5905
5906 ;ADDD RESULT INCORRECT
5907 025756 012737 026266 001246 AAERR1: MOV #AAPAT2,@#STMP6
5908 025764 012737 026246 001242 AAERR10: MOV #AAPAT0,@#STMP4
5909 025772 012737 026256 001240 MOV #AAPAT1,@#STMP3
5910 026000 012737 026236 001244 MOV #AADATO,@#STMP5
5911 026006 104162 1$: ERROR 162
5912 026010 000552 BR AADONE
5913 026012 012737 026266 001246 AAERR2: MOV #AAPAT2,@#STMP6 ;(BUT FT) FAILED.
5914 026020 012737 026246 001242 MOV #AAPAT0,@#STMP4
5915 026026 012737 026256 001240 MOV #AAPAT1,@#STMP3
5916 026034 012737 026236 001244 MOV #AADATO,@#STMP5
5917 026042 104156 1$: ERROR 156
5918 026044 000534 BR AADONE
5919 026046 012737 026276 001246 AAERR3: MOV #AAPAT3,@#STMP6 ;DATA ERROR.
5920 026054 000743 BR AAERR10
5921 026056 012737 026276 001246 AAERR4: MOV #AAPAT3,@#STMP6 ;BAD CONSTANT
5922 026064 012737 026246 001242 MOV #AAPAT0,@#STMP4
5923 026072 012737 026256 001240 MOV #AAPAT1,@#STMP3
5924 026100 012737 026236 001244 MOV #AADATO,@#STMP5
5925 026106 104160 1$: ERROR 160
5926 026110 000512 BR AADONE
5927 026112 012737 026276 001246 AAERR5: MOV #AAPAT3,@#STMP6 ;(BUT FT) FAILED.
5928 026120 012737 026246 001242 MOV #AAPAT0,@#STMP4
5929 026126 012737 026256 001240 MOV #AAPAT1,@#STMP3
5930 026134 012737 026236 001244 MOV #AADATO,@#STMP5
5931 026142 104157 1$: ERROR 157
5932 026144 000474 BR AADONE
5933 026146 012737 026316 001240 AAERR6: MOV #AAPAT5,@#STMP3 ;FD=0 AND
5934 026154 012737 026246 001242 MOV #AAPAT0,@#STMP4 ;DATA ERROR
5935 026162 012737 026236 001244 MOV #AADATO,@#STMP5
5936 026170 012737 026326 001246 MOV #AAPAT6,@#STMP6
5937 026176 104160 1$: ERROR 160
5938 026200 000456 BR AADONE
5939 026202 012737 026316 001240 AAERR7: MOV #AAPAT5,@#STMP3 ;CONSTANT ERROR
5940 026210 012737 026246 001242 MOV #AAPAT0,@#STMP4
5941 026216 012737 026236 001244 MOV #AADATO,@#STMP5
5942 026224 012737 026326 001246 MOV #AAPAT6,@#STMP6
5943 026232 104161 1$: ERROR 161
5944 026234 000440 BR AADONE
5945 026236 000000 AADATO: 0
    
```

```

5946 026240 000000 0
5947 026242 000000 0
5948 026244 000000 0
5949 026246 000200 AAPAT0: 200
5950 026250 000000 0
5951 026252 000000 0
5952 026254 000000 0
5953 026256 000200 AAPAT1: 200
5954 026260 000000 0
5955 026262 000000 0
5956 026264 000001 1
5957 026266 000400 AAPAT2: 400
5958 026270 000000 0
5959 026272 000000 0
5960 026274 000000 0
5961 026276 000400 AAPAT3: 400
5962 026300 000000 0
5963 026302 000000 0
5964 026304 000001 1
5965 026306 000400 AAPAT4: 400
5966 026310 000000 0
5967 026312 100000 100000
5968 026314 000000 0
5969 026316 000200 AAPAT5: 200
5970 026320 000001 1
5971 026322 000000 0
5972 026324 000000 0
5973 026326 000400 AAPAT6: 400
5974 026330 000001 1
5975 026332 000000 0
5976 026334 000000 0
5977 026336 AADONE:
5978 026336 104412 RSETUP ;GO INITIALIZE THE FPS AND STACK; AND
5979 ;SEE IF THE USER HAS EXPRESSED
5980 ;THE DESIRE TO CHANGE THE SOFTWARE
5981 ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
5982 ;THE USER TYPED CONTROL G?).
5983 ;*****
5984 ;*TEST 32 ADDF AND ADD WITH E(AC) LESS THAN E(FSRC) TEST
5985 ;*
5986 ;*THIS IS A TEST OF THE ADDD AND ADDF
5987 ;*INSTRUCTIONS AND THE ALIGN AC ALGORITHM
5988 ;*FLOWS. THE CONSTANT (25 FOR FLOATING, 57 FOR
5989 ;*DOUBLE) USED IS CHECKED. THEN SIMPLE
5990 ;*AND WORST CASE ALIGNMENT SITUATIONS ARE
5991 ;*TRIED. NOTE E(AC) IS LESS THEN E(FSRC)
5992 ;*
5993 ;*****
5994 026340 000004 TST32: SCOPE
5995 ;EXPONENT DIFFERENCE=57=71 (OCT) FD=1
5996 026342 CC1:
5997 026342 104413 LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
5998 026344 012704 003200 MOV #3200,R4 ;SET FIV,FIV, AND FD
5999 026350 170104 LDFPS R4
6000 026352 012737 026372 001236 MOV #CC2,#STMP2
6001 026360 012700 027774 MOV #CCP0,R0 ;SET AC0 OPERAND
    
```

6002	026364	172410			LDD	(R0),AC0		;AC0
6003	026366	012720	030014		MOV	#CCP2,R0		
6004	026372	172010		CC2:	ADDD	(R0),AC0		;TEST INSTRUCTION
6005	026374	170205			STFPS	R5		;GET FPS
6006	026376	012700	027764		MOV	#CCDAT0,R0		;GET THE RESULT
6007	026402	174010			STD	AC0,(R0)		
6008	026404	012701	030014		MOV	#CCP2,R1		;IS IT CORRECT
6009	026410	012702	000004		MOV	#4,R2		
6010	026414	022021		CC3:	CMP	(R0)+,(R1)+		
6011	026416	001415			BEQ	CC6		
6012	026420	012700	027764		MOV	#CCDAT0,R0		;DID A BAD
6013	026424	012701	027774		MOV	#CCP0,R1		;CONSTANT (NOT 57)
6014	026430	012702	000004		MOV	#4,R2		;GET GENERATED
6015	026434	022021		CC4:	CMP	(R0)+,(R1)+		;FOR THE ALIGNMENT
6016	026436	001402			BEQ	CC5		;FLOWS?
6017	026440	000137	027362		JMP	@#CCER1		;DATA ERROR.D
6018	026444	077205		CC5:	SOB	R2,CC4		
6019	026446	000137	027420		JMP	@#CCER2		;BAD CONSTANT.D
6020	026452	077220		CC6:	SOB	R2,CC3		
6021	026454	020405			CMP	R4,R5		;FPS CORRECT?
6022	026456	001402			BEQ	CC7		
6023	026460	000137	027326		JMP	@#CCER0		;BAD FPS.
6024								;EXPONENT DIFFERENCE=56=70 (OCT) FD=1
6025	026464			CC7:				
6026	026464	104413			LPERR			;SET UP THE LOOP ON ERROR ADDRESS.
6027	026466	012704	003200		MOV	#3200,R4		;SET FIV,FIV, AND FD
6028	026472	170104			LDFPS	R4		
6029	026474	012737	026514 001236		MOV	#CC8,@#STMP2		
6030	026502	012700	027774		MOV	#CCP0,R0		;SET AC0 OPERAND
6031	026506	172410			LDD	(R0),AC0		
6032	026510	012700	030004		MOV	#CCP1,R0		;FSRC
6033	026514	172010		CC8:	ADDD	(R0),AC0		;TEST INSTRUCTION
6034	026516	170205			STFPS	R5		;GET FPS
6035	026520	012700	027764		MOV	#CCDAT0,R0		;GET THE RESULT
6036	026524	174010			STD	AC0,(R0)		
6037	026526	012701	030064		MOV	#CCP7,R1		;IS IT CORRECT
6038	026532	012702	000004		MOV	#4,R2		
6039	026536	022021		CC9:	CMP	(R0)+,(R1)+		
6040	026540	001415			BEQ	CC12		
6041	026542	012700	027764		MOV	#CCDAT0,R0		;DID A BAD
6042	026546	012701	030004		MOV	#CCP1,R1		;CONSTANT (NOT 57)
6043	026552	012702	000004		MOV	#4,R2		;GET GENERATED
6044	026556	022021		CC10:	CMP	(R0)+,(R1)+		;FOR THE ALIGNMENT
6045	026560	001402			BEQ	CC11		;FLOWS?
6046	026562	000137	027456		JMP	@#CCER3		;DATA ERROR.D
6047	026566	077205		CC11:	SOB	R2,CC10		
6048	026570	000137	027474		JMP	@#CCER4		;BAD CONSTANT.D
6049	026574	077220		CC12:	SOB	R2,CC9		
6050	026576	020405			CMP	R4,R5		;FPS CORRECT?
6051	026600	001402			BEQ	CC13		
6052	026602	000137	027326		JMP	@#CCER0		;BAD FPS.
6053								;EXPONENT DIFFERENCE=25=31 (OCT) FD=0
6054	026606			CC13:				
6055	026606	104413			LPERR			;SET UP THE LOOP ON ERROR ADDRESS.
6056	026610	012737	026636 001236		MOV	#CC14,@#STMP2		
6057	026616	012700	027774		MOV	#CCP0,R0		;SET UP AC0 OPERAND.

6058	026622	172410		LDD	(R0),AC0	
6059	026624	012704	030000	MOV	#3000,R4	;SET FIV,FIV. CLEAR FD.
6060	026630	170104		LDFPS	R4	
6061	026632	012700	030054	MOV	#CCP6,R0	;FSRC
6062	026636	172010		CC14: ADDF	(R0),AC0	;TEST INSTRUCTION
6063	026640	170205		STFPS	R5	
6064	026642	170011		SETD		;REENTER DOUBLE MOVE
6065	026644	012700	027764	MOV	#CCDAT0,R0	;GET THE RESULT
6066	026650	174010		STD	AC0,(R0)	
6067	026652	012701	030054	MOV	#CCP6,R1	;IS THE RESULT CORRECT?
6068	026656	012702	000002	MOV	#2,R2	
6069	026662	022021		CC15: CMP	(R0)+,(R1)+	
6070	026664	001415		BEQ	CC18	
6071	026666	012700	027764	MOV	#CCDAT0,R0	;WAS A BAD CONSTANT
6072	026672	012701	030024	MOV	#CCP3,R1	;USED (NOT 25) IN
6073	026676	012702	000002	MOV	#2,R2	;THE ALIGN FLOWS?
6074	026702	022021		CC16: CMP	(R0)+,(R1)+	
6075	026704	001402		BEQ	CC17	
6076	026706	000137	027532	JMP	#CCER5	;DATA ERROR F
6077	026712	077205		CC17: SOB	R2,CC16	
6078	026714	000137	027566	JMP	#CCER6	;BAD CONSTANT F
6079	026720	077220		CC18: SOB	R2,CC15	
6080	026722	020405		CMP	R4,R5	
6081	026724	001402		BEQ	CC19	
6082	026726	000137	027344	JMP	#CCER90	;BAD FPS.
6083						;EXPONENT DIFFERENCE=24=30 (OCT) FD=0
6084	026732			CC19:		
6085	026732	104413		LPERR		;SET UP THE LOOP ON ERROR ADDRESS.
6086	026734	012737	026762 001236	MOV	#CC20,#\$TMP2	
6087	026742	012700	030024	MOV	#CCP3,R0	;SET UP AC0 OPERAND.
6088	026746	172410		LDD	(R0),AC0	
6089	026750	012704	003000	MOV	#3000,R4	;SET FIV,FIV. CLEAR FD.
6090	026754	170104		LDFPS	R4	
6091	026756	012700	030044	MOV	#CCP5,R0	;FSRC
6092	026762	172010		CC20: ADDF	(R0),AC0	;TEST INSTRUCTION
6093	026764	170205		STFPS	R5	
6094	026766	170011		SETD		;REENTER DOUBLE MOVE
6095	026770	012700	027764	MOV	#CCDAT0,R0	;GET THE RESULT
6096	026774	174010		STD	AC0,(R0)	
6097	026776	012701	030074	MOV	#CCP10,R1	;IS THE RESULT CORRECT?
6098	027002	012702	000002	MOV	#2,R2	
6099	027006	022021		CC21: CMP	(R0)+,(R1)+	
6100	027010	001415		BEQ	CC24	
6101	027012	012700	027764	MOV	#CCDAT0,R0	;WAS A BAD CONSTANT
6102	027016	012701	030044	MOV	#CCP5,R1	;USED (NOT 25) IN
6103	027022	012702	000002	MOV	#2,R2	;THE ALIGN FLOWS?
6104	027026	022021		CC22: CMP	(R0)+,(R1)+	
6105	027030	001402		BEQ	CC23	
6106	027032	000137	027622	JMP	#CCER7	;DATA ERROR F
6107	027036	077205		CC23: SOB	R2,CC22	
6108	027040	000137	027640	JMP	#CCER8	;BAD CONSTANT F
6109	027044	077220		CC24: SOB	R2,CC21	
6110	027046	020405		CMP	R4,R5	
6111	027050	001402		BEQ	CC25	
6112	027052	000137	027344	JMP	#CCER90	;BAD FPS.
6113						;EXPONENT DIFFERENCE=1 FD=1

```

6114 027056          CC25:
6115 027056 104413          LPERR          ;SET UP THE LOOP ON ERROR ADDRESS,
6116 027060 012704 003200          MOV          #3200,R4          ;SET FIV,FIV, AND FD
6117 027064 170104          LDFPS        R4
6118 027066 012737 027106 001236          MOV          #CC26,@#STMP2
6119 027074 012700 027774          MOV          #CCP0,R0          ;SET AC0 OPERAND
6120 027100 172410          LDD          (R0),AC0
6121 027102 012700 030024          MOV          #CCP3,R0          ;FSRC
6122 027106 172010          CC26: ADDD          (R0),AC0          ;TEST INSTRUCTION
6123 027110 170205          STFPS        R5          ;GET FPS
6124 027112 012700 027764          MOV          #CCDAT0,R0          ;GET THE RESULT
6125 027116 174010          STD          AC0,(R0)
6126 027120 012701 030104          MOV          #CCP11,R1          ;IS IT CORRECT
6127 027124 012702 000004          MOV          #4,R2
6128 027130 022021          CC27: CMP          (R0)+,(R1)+
6129 027132 001415          BEQ          CC30
6130 027134 012700 027764          MOV          #CCDAT0,R0          ;DID A BAD
6131 027140 012701 030024          MOV          #CCP3,R1          ;CONSTANT (NOT 57)
6132 027144 012702 000004          MOV          #4,R2          ;GET GENERATED
6133 027150 022021          CC28: CMP          (R0)+,(R1)+          ;FOR THE ALIGNMENT
6134 027152 001402          BEQ          CC29          ;FLOWS?
6135 027154 000137 027674          JMP          @#CCER10          ;DATA ERROR,D
6136 027160 077205          CC29: SOB          R2,CC28
6137 027162 000137 027712          JMP          @#CCER11          ;BAD CONSTANT,D
6138 027166 077220          CC30: SOB          R2,CC27
6139 027170 020405          CMP          R4,R5          ;FPS CORRECT?
6140 027172 001402          BEQ          CC31
6141 027174 000137 027326          JMP          @#CCER0          ;BAD FPS,
6142          ;EXPONENT DIFFERENCE=100=144 (OCT) FD=1
6143 027200          CC31:
6144 027200 104413          LPERR          ;SET UP THE LOOP ON ERROR ADDRESS,
6145 027202 012704 003200          MOV          #3200,R4          ;SET FIV,FIV, AND FD
6146 027206 170104          LDFPS        R4
6147 027210 012737 027230 001236          MOV          #CC32,@#STMP2
6148 027216 012700 027774          MOV          #CCP0,R0          ;SET AC0 OPERAND
6149 027222 172410          LDD          (R0),AC0
6150 027224 012700 030034          MOV          #CCP4,R0          ;FSRC
6151 027230 172010          CC32: ADDD          (R0),AC0          ;TEST INSTRUCTION
6152 027232 170205          STFPS        R5          ;GET FPS
6153 027234 012700 027764          MOV          #CCDAT0,R0          ;GET THE RESULT
6154 027240 174010          STD          AC0,(R0)
6155 027242 012701 030034          MOV          #CCP4,R1          ;IS IT CORRECT
6156 027246 012702 000004          MOV          #4,R2
6157 027252 022021          CC33: CMP          (R0)+,(R1)+
6158 027254 001415          BEQ          CC36
6159 027256 012700 027764          MOV          #CCDAT0,R0          ;DID A BAD
6160 027262 012701 030034          MOV          #CCP4,R1          ;CONSTANT (NOT 57)
6161 027266 012702 000004          MOV          #4,R2          ;GET GENERATED
6162 027272 022021          CC34: CMP          (R0)+,(R1)+          ;FOR THE ALIGNMENT
6163 027274 001402          BEQ          CC35          ;FLOWS?
6164 027276 000137 027730          JMP          @#CCER12          ;DATA ERROR,D
6165 027302 077205          CC35: SOB          R2,CC34
6166 027304 000137 027746          JMP          @#CCER13          ;BAD CONSTANT,D
6167 027310 077220          CC36: SOB          R2,CC33
6168 027312 020405          CMP          R4,R5          ;FPS CORRECT?
6169 027314 001402          BEQ          CC37
    
```

FORM 1413  
 PRINTED IN U.S.A.  
 27



6170	027316	000137	027326			JMP	#CCER0		;BAD FPS.
6171	027322	000137	030124		CC37:	JMP	#CCDONE		
6172	027326	010437	001242		CCER0:	MOV	R4,#STMP4		;FPS ERROR D
6173	027332	010537	001240			MOV	R5,#STMP3		
6174	027336	104164			1S:	ERROR	164		
6175	027340	000137	030124			JMP	#CCDONE		
6176	027344	010437	001242		CCER90:	MOV	R4,#STMP4		;FPS ERROR F
6177	027350	010537	001240			MOV	R5,#STMP3		
6178	027354	104165			1S:	ERROR	165		
6179	027356	000137	030124			JMP	#CCDONE		
6180	027362	012737	030014	001240	CCER1:	MOV	#CCP2,#STMP3		;DATA ERROR D
6181	027370	012737	030014	001246		MOV	#CCP2,#STMP6		
6182	027376	012737	027774	001242	CCER50:	MOV	#CCP0,#STMP4		
6183	027404	012737	027764	001244		MOV	#CCDAT0,#STMP5		
6184	027412	104166			1S:	ERROR	166		
6185	027414	000137	030124			JMP	#CCDONE		
6186	027420	012737	030014	001240	CCER2:	MOV	#CCP2,#STMP3		;CONSTANT BAD D(B)
6187	027426	012737	030014	001246		MOV	#CCP2,#STMP6		
6188	027434	012737	027774	001242	CCER22:	MOV	#CCP0,#STMP4		
6189	027442	012737	027764	001244		MOV	#CCDAT0,#STMP5		
6190	027450	104172			1S:	ERROR	172		
6191	027452	000137	030124			JMP	#CCDONE		
6192	027456	012737	030004	001240	CCER3:	MOV	#CCP1,#STMP3		
6193	027464	012737	030064	001246		MOV	#CCP7,#STMP6		
6194	027472	000741				BR	CCER50		
6195	027474	012737	030004	001240	CCER4:	MOV	#CCP1,#STMP3		;CONSTANT BAD D(G)
6196	027502	012737	030064	001246		MOV	#CCP7,#STMP6		
6197	027510	012737	027774	001242	CCER44:	MOV	#CCP0,#STMP4		
6198	027516	012737	027764	001244		MOV	#CCDAT0,#STMP5		
6199	027524	104173			1S:	ERROR	173		
6200	027526	000137	030124			JMP	#CCDONE		
6201	027532	012737	030054	001240	CCER5:	MOV	#CCP6,#STMP3		;DATA ERROR F
6202	027540	012737	030054	001246		MOV	#CCP6,#STMP6		
6203	027546	012737	027774	001242	CCER55:	MOV	#CCP0,#STMP4		
6204	027554	012737	027764	001244		MOV	#CCDAT0,#STMP5		
6205	027562	104170			1S:	ERROR	170		
6206	027564	000557				BR	CCDONE		
6207	027566	012737	030054	001240	CCER6:	MOV	#CCP6,#STMP3		;CONSTANT BAD F(B)
6208	027574	012737	030054	001246		MOV	#CCP6,#STMP6		
6209	027602	012737	027774	001242		MOV	#CCP0,#STMP4		
6210	027610	012737	027764	001244		MOV	#CCDAT0,#STMP5		
6211	027616	104174			1S:	ERROR	174		
6212	027620	000541				BR	CCDONE		
6213	027622	012737	030044	001240	CCER7:	MOV	#CCP5,#STMP3		;DATA ERROR F
6214	027630	012737	030074	001246		MOV	#CCP10,#STMP6		
6215	027636	000743				BR	CCER55		
6216	027640	012737	030044	001240	CCEP8:	MOV	#CCP5,#STMP3		;CONSTANT BAD F(G)
6217	027646	012737	030074	001246		MOV	#CCP10,#STMP6		
6218	027654	012737	027764	001244		MOV	#CCDAT0,#STMP5		
6219	027662	012737	027774	001242		MOV	#CCP0,#STMP4		
6220	027670	104175			1S:	ERROR	175		
6221	027672	000514				BR	CCDONE		
6222	027674	012737	030024	001240	CCER10:	MOV	#CCP3,#STMP3		;DATA ERROR D
6223	027702	012737	030104	001246		MOV	#CCP11,#STMP6		
6224	027710	000632				BR	CCER50		
6225	027712	012737	030024	001240	CCER11:	MOV	#CCP3,#STMP3		;CONSTANT BAD D(G)

6226	027720	012737	030104	001246	MOV	#CCP11,@#STMP6	
6227	027726	000670			BR	CCER44	
6228	027730	012737	030034	001240	CCER12: MOV	#CCP4,@#STMP3	;DATA ERROR D
6229	027736	012737	030034	001246	MOV	#CCP4,@#STMP6	
6230	027744	000614			BR	CCER50	
6231	027746	012737	030034	001240	CCER13: MOV	#CCP4,@#STMP3	;CONSTANT BAD D(B)
6232	027754	012737	030034	001246	MOV	#CCP4,@#STMP6	
6233	027762	000624			BR	CCER22	
6234	027764	000000			CCDAT0:	0	
6235	027766	000000				0	
6236	027770	000000				0	
6237	027772	000000				0	
6238	027774	000200			CCP0:	200	;E(AC)=1
6239	027776	000000				0	
6240	030000	000000				0	
6241	030002	000000				0	
6242	030004	016200			CCP1:	16200	;E(FSRC)=E(AC)+56=57
6243	030006	000000				0	;=71(OCT)
6244	030010	000000				0	
6245	030012	000000				0	
6246	030014	016400			CCP2:	16400	;E(FSRC)=E(AC)+57=58
6247	030016	000000				0	;=72(OCT)
6248	030020	000000				0	
6249	030022	000000				0	
6250	030024	000400			CCP3:	400	;E(FSRC)=E(AC)+1=2
6251	030026	000000				0	
6252	030030	000000				0	
6253	030032	000000				0	
6254	030034	031200			CCP4:	31200	;E(FSRC)=E(AC)+100=101=145(OCT)
6255	030036	000000				0	
6256	030040	000000				0	
6257	030042	000000				0	
6258	030044	006200			CCP5:	6200	;E(FSRC)=E(AC)+24=25=31(OCT)
6259	030046	000000				0	
6260	030050	000000				0	
6261	030052	000000				0	
6262	030054	006400			CCP6:	6400	;E(FSRC)=E(AC)+25=26=32(OCT)
6263	030056	000000				0	
6264	030060	000000				0	
6265	030062	000000				0	
6266	030064	016200			CCP7:	16200	;CCP1 RES
6267	030066	000000				0	
6268	030070	000000				0	
6269	030072	000001				1	
6270	030074	006200			CCP10:	6200	;CCP5 RES
6271	030076	000001				1	
6272	030100	000000				0	
6273	030102	000000				0	
6274	030104	000500			CCP11:	500	;CCP3 RES
6275	030106	000000				0	
6276	030110	000000				0	
6277	030112	000000				0	
6278	030114	000200			CCP12:	200	;BAD CONSTANT
6279	030116	000000				0	;RES CCP2,CCP4
6280	030120	000000				0	
6281	030122	000000				0	

FORM 143  
PRINTED IN U.S.A.

```

6282
6283 030124          CCDONE:
6284 030124 104412          RSETUP          ;GO INITIALIZE THE FPS AND STACK; AND
6285                                     ;SEE IF THE USER HAS EXPRESSED
6286                                     ;THE DESIRE TO CHANGE THE SOFTWARE
6287                                     ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
6288                                     ;THE USER TYPED CONTROL G?),
6289
6290                                     ;*****
6291                                     ;*TEST 33      ADDF AND ADDD WITH E(AC) GREATER THAN E(FSRC) TEST
6292                                     ;*
6293                                     ;*THIS IS A TEST OF THE ADDD AND ADDF
6294                                     ;*INSTRUCTIONS AND THE ALIGN FSRC ALGORITHM
6295                                     ;*FLOWS.  FIRST THE CONSTANT USED IS CHECKED.
6296                                     ;*THEN SIMPLE AND WORST CASE ALIGNMENT
6297                                     ;*SITUATIONS ARE TRIED.  NOTE E(AC)
6298                                     ;*IS GREATER THAN E(FSRC).
6299                                     ;*
6300                                     ;*****
6301 030126 000004          TST33:  SCOPE
6302                                     ;EXPONENT DIFFERENCE=57=71 (OCT) FD=1
6303                                     BB1:
6304 030130 104413          LPERR          ;SET UP THE LOOP ON ERROR ADDRESS.
6305 030132 012704 003200          MOV          #3200,R4          ;SET FIV FIV, AND FD
6306 030136 170104          LDFPS          R4
6307 030140 012737 030766 000244          MOV          #BBER0,@#FPVECT ;SET UP FOR ERROR
6308 030146 012737 030166 001236          MOV          #BB2,@#STMP2    ;IN CASE THE OVER\
6309                                     ;UNDER FLOWS FAIL.
6310 030154 012700 031330          MOV          #BBPAT2,R0      ;SET AC0 OPERAND.
6311 030160 172410          LDD          (R0),AC0
6312 030162 012700 031320          MOV          #BBPAT1,R0      ;FSRC
6313 030166 172010          BB2:  ADDD          (R0),AC0    ;TEST INSTRUCTION
6314 030170 170205          STFPS          R5
6315 030172 012700 031300          BB3:  MOV          #BBDAT0,R0  ;GET THE RESULT
6316 030176 174010          STD          AC0,(R0)
6317 030200 012701 031330          MOV          #BBPAT2,R1      ;RESULT CORRECT?
6318 030204 012702 000004          MOV          #4,R2
6319 030210 022021          BB4:  CMP          (R0)+,(R1)+
6320 030212 001402          BEQ          BB5
6321 030214 000137 031026          JMP          @#BBER1          ;DATA ERROR D
6322 030220 077205          BB5:  SOB          R2,BB4
6323                                     ;WAS FPS CORRECT?
6324 030222 020405          CMP          R4,R5
6325 030224 001402          BEQ          BB6
6326 030226 000137 030766          JMP          @#BBER0          ;FPS ERROR
6327                                     ;EXPONENT DIFFERENCE=56=70 (OCT) FD=1
6328                                     BB6:
6329 030232 104413          LPERR          ;SET UP THE LOOP ON ERROR ADDRESS.
6330 030234 012704 003200          MOV          #3200,R4          ;SET FIV,FIV, AND FD
6331 030240 170104          LDFPS          R4
6332 030242 012737 030262 001236          MOV          #BB7,@#STMP2
6333 030250 012700 031350          MOV          #BBPAT4,R0      ;SET AC0 OPERAND
6334 030254 172410          LDD          (R0),AC0
6335 030256 012700 031320          MOV          #BBPAT1,R0      ;FSRC
6336 030262 172010          BB7:  ADDD          (R0),AC0    ;TEST INSTRUCTION
6337 030264 170205          STFPS          R5          ;GET FPS

```

```

6338 030266 012700 031300      MOV      #BBDAT0,R0      ;GET THE RESULT
6339 030272 174010      STD      AC0,(R0)
6340 030274 012701 031410      MOV      #BBP10,R1      ;IS IT CORRECT
6341 030300 012702 000004      MOV      #4,R2
6342 030304 022021      BB10:   CMP      (R0)+,(R1)+
6343 030306 001415      BEQ     BB13
6344 030310 012700 031300      MOV      #BBDAT0,R0      ;DID A BAD
6345 030314 012701 031350      MOV      #BBPAT4,R1      ;CONSTANT (NOT 57)
6346 030320 012702 000004      MOV      #4,R2      ;GET GENERATED
6347 030324 022021      BB11:   CMP      (R0)+,(R1)+ ;FOR THE ALIGNMENT
6348 030326 001402      BEQ     BB12      ;FLOWS?
6349 030330 000137 031064      JMP     @#BBER2      ;DATA ERROR,D
6350 030334 077205      BB12:   SOB     R2,BB11
6351 030336 000137 031102      JMP     @#BBER3      ;BAD CONSTANT,D
6352 030342 077220      BB13:   SOB     R2,BB10
6353 030344 020405      CMP     R4,R5      ;FPS CORRECT?
6354 030346 001402      BEQ     BB14
6355 030350 000137 030766      JMP     @#BBER0      ;BAD FPS,
6356      ;EXPONENT DIFFERENCE=25=31 (OCT) FD=0
6357 030354      BB14:   LPERR      ;SET UP THE LOOP ON ERROR ADDRESS,
6358 030354 104413      LPERR
6359 030356 012737 030404 001236      MOV     #BB15,@#STMP2
6360 030364 012700 031310      MOV     #BBPAT0,R0      ;SET UP AC0 OPERAND
6361 030370 172410      LDD     (R0),AC0
6362 030372 012704 003000      MOV     #3000,R4      ;SET FIV AND FIV
6363      ;CLEAR FD
6364 030376 170104      LDFPS  R4
6365 030400 012700 031320      MOV     #BBPAT1,R0      ;FSRC
6366 030404 172010      BB15:   ADDF   (R0),AC0      ;TEST INSTRUCTION
6367 030406 170205      STFPS  R5
6368 030410 170011      SETD   ;REENTED DOUBLE MODE,
6369 030412 012700 031300      MOV     #BBDAT0,R0      ;GET THE RESULT
6370 030416 174010      STD     AC0,(R0)
6371 030420 012701 031310      MOV     #BBPAT0,R1      ;IS THE RESULT
6372 030424 012702 000002      MOV     #2,R2      ;CORRECT?
6373 030430 022021      BB16:   CMP     (R0)+,(R1)+
6374 030432 001402      BEQ     BB17
6375 030434 000137 031136      JMP     @#BBER4      ;DATA ERROR F
6376 030440 077205      BB17:   SOB     R2,BB16
6377 030442 020405      CMP     R4,R5      ;IS FPS CORRECT?
6378 030444 001402      BEQ     BB20
6379 030446 000137 031006      JMP     @#BBER10     ;FPS ERROR,
6380      ;EXPONENT DIFFERENCE=24=30 (OCT)
6381 030452      BB20:   LPERR      ;SET UP THE LOOP ON ERROR ADDRESS,
6382 030452 104413      LPERR
6383 030454 012737 030502 001236      MOV     #BB21,@#STMP2
6384 030462 012700 031340      MOV     #BBPAT3,R0      ;SET UP AC0 OPERAND,
6385 030466 172410      LDD     (R0),AC0
6386 030470 012704 003000      MOV     #3000,R4      ;SET FIU,FIV, CLEAR FD,
6387 030474 170104      LDFPS  R4
6388 030476 012700 031320      MOV     #BBPAT1,R0      ;FSRC
6389 030502 172010      BB21:   ADDF   (R0),AC0      ;TEST INSTRUCTION
6390 030504 170205      STFPS  R5
6391 030506 170011      SETD   ;REENTER DOUBLE MODE
6392 030510 012700 031300      MOV     #BBDAT0,R0      ;GET THE RESULT
6393 030514 174010      STD     AC0,(R0)
    
```

6394	030516	012701	031400		MOV	#BBP7,R1	;IS THE RESULT CORRECT?
6395	030522	012702	000002		MOV	#2,R2	
6396	030526	022021		BB22:	CMP	(R0)+,(R1)+	
6397	030530	001415			BEQ	BB25	
6398	030532	012700	031300		MOV	#BBDATO,R0	;WAS A BAD CONSTANT
6399	030536	012701	031340		MOV	#BBPAT3,R1	;USED (NOT 25) IN
6400	030542	012702	000002		MOV	#2,R2	;THE ALLIGN FLOWS?
6401	030546	022021		BB23:	CMP	(R0)+,(R1)+	
6402	030550	001402			BEQ	BB24	
6403	030552	000137	031172		JMP	@#BBER5	;DATA ERROR F
6404	030556	077205		BB24:	SOB	R2,BB23	
6405	030560	000137	031210		JMP	@#BBER6	;BAD CONSTANT F
6406	030564	077220		BB25:	SOB	R2,BB22	
6407	030566	020405			CMP	R4,R5	
6408	030570	001402			BEQ	BB26	
6409	030572	000137	031006		JMP	@#BBER10	;BAD FPS.
6410							
6411	030576						
6412	030576	104413		BB26:	LPERR		;SET UP THE LOOP ON ERROR ADDRESS.
6413	030600	012737	030626 001236		MOV	#BB27,@#STMP2	
6414	030606	012704	003200		MOV	#3200,R4	
6415	030612	170104			LDFPS	R4	;SET UP AC0 OPERAND
6416	030614	012700	031360		MOV	#BBPAT5,R0	
6417	030620	172410			LDD	(R0),AC0	
6418	030622	012700	031320		MOV	#BBPAT1,R0	;FSRC
6419	030626	172010		BB27:	ADDD	(R0),AC0	;TEST INSTRUCTION
6420	030630	170205			STFPS	R5	
6421	030632	012700	031300		MOV	#BBDATO,R0	;GET THE RESULT.
6422	030636	174010			STD	AC0,(R0)	
6423	030640	012701	031420		MOV	#BBP11,R1	;IS IT CORRECT?
6424	030644	012702	000004		MOV	#4,R2	
6425	030650	022021		BB30:	CMP	(R0)+,(R1)+	
6426	030652	001402			BEQ	BB31	
6427	030654	000137	031244		JMP	@#BBER7	;DATA ERROR D
6428	030660	077205		BB31:	SOB	R2,BB30	
6429	030662	020405			CMP	R4,R5	;IS FPS CORRECT
6430	030664	001402			BEQ	BB32	
6431	030666	000137	030766		JMP	@#BBER0	
6432							
6433	030672						
6434	030672	104413		BB32:	LPERR		;SET UP THE LOOP ON ERROR ADDRESS.
6435	030674	012737	030722 001236		MOV	#BB33,@#STMP2	
6436	030702	012704	003200		MOV	#3200,R4	
6437	030706	170104			LDFPS	R4	;SET FIV,FIV AND FD
6438	030710	012700	031370		MOV	#BBPAT6,R0	;SET UP AC0 OPERAND.
6439	030714	172410			LDD	(R0),AC0	
6440	030716	012700	031320		MOV	#BBPAT1,R0	;FSRC
6441	030722	172010		BB33:	ADDD	(R0),AC0	;TEST INSTRUCTION
6442	030724	170205			STFPS	R5	
6443	030726	012700	031300		MOV	#BBDATO,R0	;GET THE RESULT
6444	030732	174010			STD	AC0,(R0)	
6445	030734	012701	031370		MOV	#BBPAT6,R1	;IS IT CORRECT
6446	030740	012702	000004		MOV	#4,R2	
6447	030744	022021		BB34:	CMP	(R0)+,(R1)+	
6448	030746	001402			BEQ	BB35	
6449	030750	000137	031262		JMP	@#BBER8	;DATA ERROR D

```

6450 030754 077205          BB35: SOB      R2, BB34
6451 030756 020405          CMP      R4, R5          ;IS FPS CORRECT
6452 030760 001002          BNE     BBER0
6453 030762 000167 000442    JMP     BBDONE
6454 030766 010437 001242    BBER0: MOV     R4, @#STMP4 ;FPS ERROR D
6455 030772 010537 001240    MOV     R5, @#STMP3
6456 030776 104164          1$:     ERROR   164
6457 031000 104412          RSETUP
6458
6459
6460
6461
6462 031002 000137 031430    JMP     @#BBDONE
6463 031006 010437 001242    BBER10: MOV    R4, @#STMP4 ;FPS ERROR F
6464 031012 010537 001240    MOV    R5, @#STMP3
6465 031016 104165          1$:     ERROR   165
6466 031020 104412          RSETUP
6467
6468
6469
6470
6471 031022 000137 031430    JMP     @#BBDONE
6472 031026 012737 031330 001242    BBER1:  MOV    #BBPAT2, @#STMP4 ;DATA ERROR D
6473 031034 012737 031330 001246    MOV    #BBPAT2, @#STMP6
6474 031042 012737 031320 001240    BBER11: MOV   #BBPAT1, @#STMP3
6475 031050 012737 031300 001244    MOV   #BBDAT0, @#STMP5
6476 031056 104166          1$:     ERROR   166
6477 031060 000137 031430    JMP     @#BBDONE
6478 031064 012737 031350 001242    BBER2:  MOV    #BBPAT4, @#STMP4
6479 031072 012737 031410 001246    MOV    #BBP10, @#STMP6
6480 031100 000760          BR     BBER11
6481 031102 012737 031350 001242    BBER3:  MOV    #BBPAT4, @#STMP4 ;BAD CONSTANT D
6482 031110 012737 031410 001246    MOV    #BBP10, @#STMP6
6483 031116 012737 031320 001240    MOV    #BBPAT1, @#STMP3
6484 031124 012737 031300 001244    MOV    #BBDAT0, @#STMP5
6485 031132 104167          1$:     ERROR   167
6486 031134 000535          BR     BBDONE
6487 031136 012737 031310 001242    BBER4:  MOV    #BBPAT0, @#STMP4 ;DATA ERROR F
6488 031144 012737 031310 001246    MOV    #BBPAT0, @#STMP6
6489 031152 012737 031320 001240    BBER40: MOV   #BBPAT1, @#STMP3
6490 031160 012737 031300 001244    MOV   #BBDAT0, @#STMP5
6491 031166 104170          1$:     ERROR   170
6492 031170 000517          BR     BBDONE
6493 031172 012737 031340 001242    BBER5:  MOV    #BBPAT3, @#STMP4
6494 031200 012737 031400 001246    MOV    #BBP7, @#STMP6
6495 031206 000761          BR     BBER40
6496 031210 012737 031340 001242    BBER6:  MOV    #BBPAT3, @#STMP4 ;CONSTANT ERROR F
6497 031216 012737 031400 001246    MOV    #BBP7, @#STMP6
6498 031224 012737 031320 001240    MOV    #BBPAT1, @#STMP3
6499 031232 012737 031300 001244    MOV    #BBDAT0, @#STMP5
6500 031240 104171          1$:     ERROR   171
6501 031242 000472          BR     BBDONE
6502 031244 012737 031360 001242    BBER7:  MOV    #BBPAT5, @#STMP4
6503 031252 012737 031320 001246    MOV    #BBPAT11, @#STMP6
6504 031260 000670          BR     BBER11
6505 031262 012737 031370 001242    BBER8:  MOV    #BBPAT6, @#STMP4

```

```
6506 031270 012737 031370 001246      MOV      #BBPAT6,0#sTMP6
6507 031276 000661      BR      BBER11
6508 031300 000000      BB DATO: 0
6509 031302 000000      0
6510 031304 000000      0
6511 031306 000000      0
6512 031310 006400      BB PAT0: 6400      ;F(AC)=E(FSRC)+25=26
6513 031312 000000      0      ;      =32(OCT)
6514 031314 000000      0
6515 031316 000000      0
6516 031320 000200      BB PAT1: 200
6517 031322 000000      0      ;E(FSPC)=1
6518 031324 000000      0
6519 031326 000000      0
6520 031330 016400      BB PAT2: 16400
6521 031332 000000      0      ;E(AC)=E(FSRC)+57=58
6522 031334 000000      0      ;      =72(OCT)
6523 031336 000000      0
6524 031340 006200      BB PAT3: 6200      ;E(AC)=E(FSRC)+24=25
6525 031342 000000      0      ;      =31(OCT)
6526 031344 000000      0
6527 031346 000000      0
6528 031350 016200      BB PAT4: 16200      ;E(AC)=E(FSR6)+56=57
6529 031352 000000      0      ;      =71(OCT)
6530 031354 000000      0
6531 031356 000000      0
6532 031360 000400      BB PAT5: 400      ;E(AC)=E(FSRC)+1=2
6533 031362 000000      0
6534 031364 000000      0
6535 031366 000000      0
6536 031370 031200      BB PAT6: 31200      ;E(AC)=E(FSRC)+100=101
6537 031372 000000      0      ;      =145(OCT)
6538 031374 000000      0
6539 031376 000000      0
6540 031400 006200      BB P7: 6200      ;BBPAT3 RES
6541 031402 000001      1
6542 031404 000000      0
6543 031406 000000      0
6544 031410 016200      BB P10: 16200      ;BBPAT4 RES
6545 031412 000000      0
6546 031414 000000      0
6547 031416 000001      1
6548 031420 000500      BB P11: 500      ;BBPAT5 RES
6549 031422 000000      0
6550 031424 000000      0
6551 031426 000000      0
6552 031430      BB DONE:
6553 031430 104412      RSETUP      ;GO INITIALIZE THE FPS AND STACK; AND
6554      ;SEE IF THE USER HAS EXPRESSED
6555      ;THE DESIPE TO CHANGE THE SOFTWARE
6556      ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
6557      ;THE USER TYPED CONTROL G?).
6558      ;*****
6559      ;*TEST 34      ADDD WITH NEGATIVE OPRANDS TEST
6560      ;*
6561      ;*THIS IS A TEST OF THE ADDD INSTRUCTION
```

```

6562 ;*WITH NEGATIVE OPERANDS. EVERY COMBINATION OF
6563 ;*OPERAND SIGNS IS TRIED.
6564 ;*
6565 ;*****
6566 031432 000004 TST34: SCOPE
6567 ;BOTH OPERANDS NEGATIVE
6568 031434 DD1:
6569 031434 104413 LPERR ;SET UP THE LOOP ON ERROR ADDRESS,
6570 031436 012704 003200 MOV #3200,R4 ;SET FIO, FIV, AND FD
6571 031442 170104 LDFPS R4
6572 031444 012737 031464 001236 MOV #DD2,#STMP2
6573 031452 012700 033314 MOV #DDP1,R0 ;SET AC0 OPERAND
6574 031456 172410 LDD (R0),AC0
6575 031460 012700 033314 MOV #DDP1,R0 ;ESRC
6576 031464 172010 DD2: ADDD (R0),AC0 ;TEST INSTRUCTION
6577 031466 170205 STFPS R5 ;GET FPS
6578 031470 012700 033274 MOV #DDDAT0,R0 ;GET THE RESULT
6579 031474 174010 STD AC0,(R0)
6580 031476 012701 033414 MOV #DDP9,R1 ;IS IT CORRECT
6581 031502 012702 000004 MOV #4,R2
6582 031506 022021 DD3: CMP (R0)+,(R1)+
6583 031510 001415 BEQ DD6
6584 031512 012700 033274 MOV #DDDAT0,R0 ;DID A ADD-SUB
6585 031516 012701 033344 MOV #DDP4,R1 ;FLOW A FAILURE
6586 031522 012702 000004 MOV #4,R2
6587 031526 022021 DD4: CMP (R0)+,(R1)+
6588 031530 001402 BEQ DD5 ;216,442,500
6589 031532 000137 032524 JMP @#DDER1 ;DATA ERROR,D
6590 031536 077205 DD5: SOB R2,DD4
6591 031540 000137 032562 JMP @#DDER2 ;FLOW FAILURE,D
6592 031544 077220 DD6: SOB R2,DD3
6593 031546 052704 000010 BIS #10,R4
6594 031552 020405 CMP R4,R5 ;FPS CORRECT?
6595 031554 001402 BEQ DD7
6596 031556 000137 032506 JMP @#DDER0 ;BAD,FPS
6597 ;AC POS FSRC NEG AC=-FSRC
6598 031562 DD7:
6599 031562 104413 LPERR ;SET UP THE LOOP ON ERROR ADDRESS,
6600 031564 012704 003200 MOV #3200,R4 ;SET FIO, FIV, AND FD
6601 031570 170104 LDFPS R4
6602 031572 012737 031612 001236 MOV #DD8,#STMP2
6603 031600 012700 033324 MOV #DDP2,R0 ;SET AC0 OPERAND
6604 031604 172410 LDD (R0),AC0
6605 031606 012700 033314 MOV #DDP1,R0 ;FSPC
6606 031612 172010 DD8: ADDD (R0),AC0 ;TEST INSTRUCTION
6607 031614 170205 STFPS R5 ;GET FPS
6608 031616 012700 033274 MOV #DDDAT0,R0 ;GET THE RESULT
6609 031622 174010 STD AC0,(R0)
6610 031624 012701 033304 MOV #DDP0,R1 ;IS IT CORRECT
6611 031630 012702 000004 MOV #4,R2
6612 031634 022021 DD10: CMP (R0)+,(R1)+
6613 031636 001402 BEQ DD11
6614 031640 000137 032620 JMP @#DDER3 ;FLOW FAILURE
6615 031644 077205 DD11: SOB R2,DD10
6616 031646 052704 000004 BIS #4,R4
6617 031652 020405 CMP R4,R5 ;FPS CORRECT?

```



```

6618 031654 001402          BEQ      DD12
6619 031656 000137 032506    JMP      @#DDER0          ;BAD FPS
6620          ;AC NEG FSRC POS          AC=-FSRC
6621 031662          DD12:
6622 031662 104413          LPERR          ;SET UP THE LOOP ON ERROR ADDRESS,
6623 031664 012704 003200    MOV      #3200,R4          ;SET FIU, FIV, AND FD
6624 031670 170104          LDFPS     R4
6625 031672 012737 031712 001236    MOV      #DD13,@#STMP2
6626 031700 012700 033314    MOV      #DDP1,R0          ;SET AC0 OPERAND
6627 031704 172410          LDD      (R0),AC0
6628 031706 012700 033324    MOV      #DDP2,R0          ;FSPC
6629 031712 172010          DD13: ADDD     (R0),AC0          ;TEST INSTRUCTION
6630 031714 170205          STFPS     R5          ;GET FPS
6631 031716 012700 033274    MOV      #DDDAT0,R0          ;GET THE RESULT
6632 031722 174010          STD      AC0,(R0)
6633 031724 012701 033304    MOV      #DDP0,R1          ;IS IT CORRECT
6634 031730 012702 000004    MOV      #4,R2
6635 031734 022021          DD14: CMP      (R0)+,(R1)+
6636 031736 001402          BEQ      DD15
6637 031740 000137 032656    JMP      @#DDER4          ;FLOW FAILURE 216,440,121
6638 031744 077205          DD15: SOB      R2,DD14
6639 031746 052704 000004    BIS      #4,R4
6640 031752 020405          CMP      R4,R5          ;EPS CORRECT?
6641 031754 001402          BEQ      DD16
6642 031756 000137 032506    JMP      @#DDER0          ;BAD FPS
6643          ;AC0 POC          FSRC      NEG      /AC/ > /FSRC/
6644 031762          DD16:
6645 031762 104413          LPERR          ;SET UP THE LOOP ON ERROR ADDRESS,
6646 031764 012704 003200    MOV      #3200,R4          ;SET FIV, FIV AND FD
6647 031770 170104          LDFPS     R4
6648 031772 012737 032012 001236    MOV      #DD17,@#STMP2
6649 032000 012700 033334    MOV      #DDP3,R0          ;SET AC0 OPERAND
6650 032004 172410          LDD      (R0),AC0
6651 032006 012700 033364    MOV      #DDP6,R0          ;ESPC
6652 032012 172010          DD17: ADDD     (R0),AC0          ;TEST INSTRUCTION
6653 032014 170205          STFPS     R5          ;GET FPS
6654 032016 012700 033274    MOV      #DDDAT0,R0          ;GET THE RESULT
6655 032022 174010          STD      AC0,(R0)
6656 032024 012701 033374    MOV      #DDP7,R1          ;IS IT CORRECT
6657 032030 012702 000004    MOV      #4,R2
6658 032034 022021          DD18: CMP      (R0)+,(R1)+
6659 032036 001415          BEQ      DD21
6660 032040 012700 033274    MOV      #DDDAT0,R0          ;FLOWS FAILURE
6661 032044 012701 033404    MOV      #DDP8,R1          ;216,440,101
6662 032050 012702 000004    MOV      #4,R2          ;GET GENERATED
6663 032054 022021          DD19: CMP      (R0)+,(R1)+
6664 032056 001402          BEQ      DD20
6665 032060 000137 032714    JMP      @#DDER5          ;DATA ERROR.
6666 032064 077205          DD20: SOB      R2,DD19
6667 032066 000137 032752    JMP      @#DDER6
6668 032072 077220          DD21: SOB      R2,DD18
6669 032074 020405          CMP      R4,R5          ;EPS CORRECT?
6670 032076 001402          BEQ      DD22
6671 032100 000137 032506    JMP      @#DDER0          ;BAD FPS
6672          ;AC NEG FSRC          POS      /FSPC/ > /AC/
6673 032104          DD22:
    
```

```

6674 032104 104413 LPERP ;SET UP THE LOOP ON ERROR ADDRESS.
6675 032106 012704 003200 MOV #3200,R4 ;SET FIO,FIV, AND FD
6676 032112 170104 LDFPS R4
6677 032114 012737 032134 001236 MOV #DD23,@#STMP2
6678 032122 012700 033364 MOV #DDP6,R0 ;SET AC0 OPERAND
6679 032126 172410 LDD (R0),AC0
6680 032130 012700 033334 MOV #DDP3,R0 ;FSPC
6681 032134 172010 DD23: ADDD (R0),AC0 ;TEST INSTRUCTION
6682 032136 170205 STFPS R5 ;GET FPS
6683 032140 012700 033274 MOV #DDDAT0,R0 ;GET THE RESULT
6684 032144 174010 STD AC0,(R0)
6685 032146 012701 033374 MOV #DDP7,R1 ;IS IT CORRECT?
6686 032152 012702 000004 MOV #4,R2
6687 032156 022021 DD24: CMP (R0)+,(R1)+
6688 032160 001415 BEQ DD27
6689 032162 012700 033274 MOV #DDDAT0,R0 ;FLO,S FAILURE
6690 032166 012701 033404 MOV #DDP8,R1 ;CONSTANT (NOT 57)
6691 032172 012702 000004 MOV #4,R2 ;216,042,101
6692 032176 021011 DD25: CMP (R0),(R1)
6693 032200 001402 BEQ DD26
6694 032202 000137 033010 JMP #DDER7 ;DATA ERROR.
6695 032206 077205 DD26: SOB R2,DD25
6696 032210 000137 033046 JMP #DDER8
6697 032214 077220 DD27: SOB R2,DD24
6698 032216 020405 CMP R4,R5 ;FPS CORRECT?
6699 032220 001402 BEQ DD30
6700 032222 000137 032506 JMP #DDER0 ;BAD FPS
6701 ;AC0 POS FSRC NEG /AC/</FRSRC/
6702 DD30:
6703 032226 104413 LPERP ;SET UP THE LOOP ON ERROR ADDRESS.
6704 032230 012704 003200 MOV #3200,R4 ;SET FIO,FIV,AND FD
6705 032234 170104 LDFPS R4
6706 032236 012737 032256 001236 MOV #DD31,@#STMP2
6707 032244 012700 033344 MOV #DDP4,R0 ;SET AC0 OPERAND
6708 032250 172410 LDD (R0),AC0
6709 032252 012700 033354 MOV #DDP5,R0 ;FSPC
6710 032256 172010 DD31: ADDD (R0),AC0 ;TEST INSTRUCTION
6711 032260 170205 STFPS R5 ;GET FPS
6712 032262 012700 033274 MOV #DDDAT0,R0 ;GET THE RESULT
6713 032266 174010 STD AC0,(R0)
6714 032270 012701 033404 MOV #DDP8,R1 ;IS IT CORRECT
6715 032274 012702 000004 MOV #4,R2
6716 032300 022021 DD32: CMP (R0)+,(R1)+
6717 032302 001415 BEQ DD35 ;ADD-SUB
6718 032304 012700 033274 MOV #DDDAT0,R0 ;FLOWAS FAILURE
6719 032310 012701 033374 MOV #DDP7,R1 ;CON 216 N440 NOT 141
6720 032314 012702 000004 MOV #4,R2 ;GET GENERATED
6721 032320 022021 DD33: CMP (R0)+,(R1)+ ;FOR THE ALLIGNMENT
6722 032322 001402 BEQ DD34 ;FLOWS?
6723 032324 000137 033104 JMP #DDER9 ;DATA ERROR, D
6724 032330 077205 DD34: SOB R2,DD33
6725 032332 000137 033142 JMP #DDER10
6726 032336 077220 DD35: SOB R2,DD32
6727 032340 052704 000010 BIS #10,R4
6728 032344 020405 CMP R4,P5 ;FPS CORRECT?
6729 032346 001402 BEQ #DD36
    
```

```

6730 032350 000137 032506          JMP      @#DDER0          ;BAD FPS
6731          ;AC0 NEG      FSRC      POS      /FSRC/</AC/
6732 032354          DD36:          LPERR          ;SET UP THE LOOP ON ERROR ADDRESS.
6733 032354 104413          MOV      #3200,R4          ;SET F10, F1V, AND FD
6734 032356 012704 003200          LDEFS   R4
6735 032362 170104          MOV      #DD37,@#STMP2
6736 032364 012737 032404 001236          MOV      #DDP5,R0          ;SET AC0 OPERAND
6737 032372 012700 033354          LDD     (R0),AC0
6738 032376 172410          MOV      #DDP4,R0          ;FSPC
6739 032400 012700 033344          DD37:  ADDD   (R0),AC0          ;TEST INSTRUCTION
6740 032404 172010          STEPS   R5          ;GET FPS
6741 032406 170205          MOV      #DDDAT0,R0          ;GET THE RESULT
6742 032410 012700 033274          STD     AC0,(R0)
6743 032414 174010          MOV      #DDP8,R1          ;IS IT CORRECT
6744 032416 012701 033404          MOV      #4,R2
6745 032422 012702 000004          DD38:  CMP     (R0)+,(R1)+
6746 032426 022021          BEQ     DD41
6747 032430 001415          MOV      #DDDAT0,R0          ;ADD SUB
6748 032432 012700 033274          MOV      #DDP7,R1          ;FLOWS FAILURES
6749 032436 012701 033374          MOV      #4,R2          ;GET 216,042,141
6750 032442 012702 000004          DD39:  CMP     (R0)+,(R1)+          ;FOR THE ALLIGNMENT
6751 032446 022021          BEQ     DD40          ;FLOWS?
6752 032450 001402          JMP     @#DDER11          ;DATA ERROR. D
6753 032452 000137 033200          SOB     R2,DD39
6754 032456 077205          DD40:  SOB     R2,DD38
6755 032460 000137 033236          DD41:  SOB     R2,DD38
6756 032464 077220          BIS     #10,R4
6757 032466 052704 000010          CMP     R4,R5          ;FPS CORRECT?
6758 032472 020405          BEQ     DD42
6759 032474 001402          JMP     @#DDER0          ;BAD FPS
6760 032476 000137 032506          DD42:  JMP     @#DDDONE
6761 032502 000137 033424          DDER0:  MOV     R4,@#STMP4          ;FPS ERROR
6762 032506 010437 001242          MOV     R5,@#STMP3
6763 032512 010537 001240          1S:    ERROR 164
6764 032516 104164          JMP     @#DDDONE
6765 032520 000137 033424          DDER1:  MOV     #DDP1,@#STMP3
6766 032524          MOV     #DDP1,@#STMP4
6767 032524 012737 033314 001240          MOV     #DDDAT0,@#STMP5
6768 032532 012737 033314 001242          MOV     #DDP9,@#STMP6
6769 032540 012737 033274 001244          1S:    ERROR 165
6770 032546 012737 033414 001246          JMP     @#DDDONE
6771 032554 104165          DDER2:  MOV     #DDP1,@#STMP3
6772 032556 000137 033424          MOV     #DDP1,@#STMP4
6773 032562          MOV     #DDDAT0,@#STMP5
6774 032562 012737 033314 001240          MOV     #DDP9,@#STMP6
6775 032570 012737 033314 001242          1S:    ERROR 176
6776 032576 012737 033274 001244          JMP     @#DDDONE
6777 032604 012737 033414 001246          DDER3:  MOV     #DDP1,@#STMP3
6778 032612 104176          MOV     #DDP2,@#STMP4
6779 032614 000137 033424          MOV     #DDDAT0,@#STMP5
6780 032620          MOV     #DDP0,@#STMP6
6781 032620 012737 033314 001240          1S:    ERROR 177
6782 032626 012737 033324 001242
6783 032634 012737 033274 001244
6784 032642 012737 033304 001246
6785 032650 104177

```

6786	032652	000137	033424		JMP	@#DDDDONE
6787	032656			DDER4:		
6788	032656	012737	033324	001240	MOV	#DDP2,@#STMP3
6789	032664	012737	033314	001242	MOV	#DDP1,@#STMP4
6790	032672	012737	033274	001244	MOV	#DDDATO,@#STMP5
6791	032700	012737	033304	001246	MOV	#DDP0,@#STMP6
6792	032706	104200			1S:	ERROR 200
6793	032710	000137	033424		JMP	@#DDDDONE
6794	032714			DDER5:		
6795	032714	012737	033364	001240	MOV	#DDP6,@#STMP3
6796	032722	012737	033334	001242	MOV	#DDP3,@#STMP4
6797	032730	012737	033274	001244	MOV	#DDDATO,@#STMP5
6798	032736	012737	033374	001246	MOV	#DDP7,@#STMP6
6799	032744	104165			1S:	ERROR 165
6800	032746	000137	033424		JMP	@#DDDDONE
6801	032752			DDER6:		
6802	032752	012737	033364	001240	MOV	#DDP6,@#STMP3
6803	032760	012737	033334	001242	MOV	#DDP3,@#STMP4
6804	032766	012737	033274	001244	MOV	#DDDATO,@#STMP5
6805	032774	012737	033374	001246	MOV	#DDP7,@#STMP6
6806	033002	104201			1S:	ERROR 201
6807	033004	000137	033424		JMP	@#DDDDONE
6808	033010			DDER7:		
6809	033010	012737	033334	001240	MOV	#DDP3,@#STMP3
6810	033016	012737	033364	001242	MOV	#DDP6,@#STMP4
6811	033024	012737	033274	001244	MOV	#DDDATO,@#STMP5
6812	033032	012737	033374	001246	MOV	#DDP7,@#STMP6
6813	033040	104165			1S:	ERROR 165
6814	033042	000137	033424		JMP	@#DDDDONE
6815	033046			DDER8:		
6816	033046	012737	033334	001240	MOV	#DDP3,@#STMP3
6817	033054	012737	033364	001242	MOV	#DDP6,@#STMP4
6818	033062	012737	033274	001244	MOV	#DDDATO,@#STMP5
6819	033070	012737	033374	001246	MOV	#DDP7,@#STMP6
6820	033076	104202			1S:	ERROR 202
6821	033100	000137	033424		JMP	@#DDDDONE
6822	033104			DDER9:		
6823	033104	012737	033354	001240	MOV	#DDP5,@#STMP3
6824	033112	012737	033344	001242	MOV	#DDP4,@#STMP4
6825	033120	012737	033274	001244	MOV	#DDDATO,@#STMP5
6826	033126	012737	033404	001246	MOV	#DDP8,@#STMP6
6827	033134	104165			1S:	ERROR 165
6828	033136	000137	033424		JMP	@#DDDDONE
6829	033142			DDER10:		
6830	033142	012737	033354	001240	MOV	#DDP5,@#STMP3
6831	033150	012737	033344	001242	MOV	#DDP4,@#STMP4
6832	033156	012737	033274	001244	MOV	#DDDATO,@#STMP5
6833	033164	012737	033404	001246	MOV	#DDP8,@#STMP6
6834	033172	104203			1S:	ERROR 203
6835	033174	000137	033424		JMP	@#DDDDONE
6836	033200			DDER11:		
6837	033200	012737	033344	001240	MOV	#DDP4,@#STMP3
6838	033206	012737	033354	001242	MOV	#DDP5,@#STMP4
6839	033214	012737	033274	001244	MOV	#DDDATO,@#STMP5
6840	033222	012737	033404	001246	MOV	#DDP8,@#STMP6
6841	033230	104165			1S:	ERROR 165

```

6842 033232 000137 033424          JMP      @#DDDONE
6843 033236          DDER12:
6844 033236 012737 033344 001240      MOV      #DDP4,@#STMP3
6845 033244 012737 033354 001242      MOV      #DDP5,@#STMP4
6846 033252 012737 033274 001244      MOV      #DDDAT0,@#STMP5
6847 033260 012737 033404 001246      MOV      #DDP8,@#STMP6
6848 033266 104204          1S:     ERROR    204
6849 033270 000137 033424          JMP      @#DDDONE
6850 033274 000000          DDDAT0: 0
6851 033276 000000          0
6852 033300 000000          0
6853 033302 000000          0
6854 033304 000000          DDP0:   0
6855 033306 000000          0
6856 033310 000000          0
6857 033312 000000          0
6858 033314 100200          DDP1:   100200      ;=DDP2
6859 033316 000000          0
6860 033320 000000          0
6861 033322 000000          0
6862 033324 000200          DDP2:   200      ;=DDP1
6863 033326 000000          0
6864 033330 000000          0
6865 033332 000000          0
6866 033334 001100          DDP3:   1100      ;EXP=4
6867 033336 000000          0      ;FRAC=...110...
6868 033340 000000          0
6869 033342 000000          0
6870 033344 000600          DDP4:   600      ;EXP=3
6871 033346 000000          0      ;FRAC=...100...
6872 033350 000000          0
6873 033352 000000          0
6874 033354 101100          DDP5:   101100      ;=DDP3
6875 033356 000000          0
6876 033360 000000          0
6877 033362 000000          0
6878 033364 100600          DDP6:   100600      ;=DDP4
6879 033366 000000          0
6880 033370 000000          0
6881 033372 000000          0
6882 033374 001000          DDP7:   1000      ;DDP3+DDP6
6883 033376 000000          0
6884 033400 000000          0
6885 033402 000000          0
6886 033404 101000          DDP8:   101000      ;DDP5+DDP4
6887 033406 000000          0
6888 033410 000000          0
6889 033412 000000          0
6890 033414 100400          DDP9:   100400      ;DDP1+DDP1
6891 033416 000000          0
6892 033420 000000          0
6893 033422 000000          0
6894 033424 000005          DDDONE:  RESET
6895          ;*****
6896          ;*TEST 35      SUBD TEST
6897          ;*
    
```

MICRO BUSINESS FORMS, INC. PO BOX 1413 PRINTED IN U.S.A. 27

```

6898 ;* THIS IS A TEST OF THE SUBD INSTRUCTION.
6899 ;* BOTH A POSITIVE AND A NEGATIVE NUMBER
6900 ;* IS SUBTRACTED FROM IT SELF
6901 ;*
6902 ;*****
6903 033426 000004 IST35: SCOPE
6904 ; USE POSITIVE OPERANDS
6905 033430 EE1:
6906 033430 104413 LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
6907 033432 012704 003200 MOV #3200,R4 ;SET F10, F1V, AND FD
6908 033436 170104 LDFPS R4
6909 033440 012737 033460 001236 MOV #EE2,#STMP2
6910 033446 012700 034140 MOV #EEP1,R0 ;SET AC0 OPERAND
6911 033452 172410 LDD (R0),AC0
6912 033454 012700 034140 MOV #EEP1,R0 ;FSPC
6913 033460 173010 EE2: SUBD (R0),AC0 ;TEST INSTRUCTION
6914 033462 170205 STFPS R5 ;GET FPS
6915 033464 012700 034116 MOV #EEDATO,R0 ;GET THE RESULT
6916 033470 174010 STD AC0,(R0)
6917 033472 012701 034126 MOV #EEP0,R1 ;IS IT CORRECT?
6918 033476 012702 000004 MOV #4,R2
6919 033502 022021 EE3: CMP (R0)+,(R1)+
6920 033504 001415 BEQ EE6
6921 033506 012700 034116 MOV #EEDATO,R0 ;DID A BAD
6922 033512 012701 034150 MOV #EEP2,R1 ;CONSTANT (NOT 57)
6923 033516 012702 000004 MOV #4,R2 ;GET GENERATED
6924 033522 022021 EE4: CMP (R0)+,(R1)+ ;FOR THE ALLIGNMENT
6925 033524 001402 BEQ EE5 ;FLOWS?
6926 033526 000137 033726 JMP @#EEER1 ;DATA ERROR.D
6927 033532 077205 EE5: SOB R2,EE4
6928 033534 000137 033764 JMP @#EEER2 ;BAD CONSTANT.D
6929 033540 077220 EE6: SOB R2,EE3
6930 033542 052704 000004 HIS #4,R4
6931 033546 020405 CMP R4,R5 ;FPS CORRECT?
6932 033550 001402 BEQ EE7
6933 033552 000137 033710 JMP @#EEER0 ;BAD FPS
6934 ;USE NEGATIVE OPERANDS
6935 EE7:
6936 033556 104413 LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
6937 033560 012704 003200 MOV #3200,R4 ;SET F10, F1V, AND FD
6938 033564 170104 LDFPS R4
6939 033566 012737 033606 001236 MOV #EE8,#STMP2
6940 033574 012700 034160 MOV #EEP3,R0 ;SET AC0 OPERAND
6941 033600 172410 LDD (R0),AC0
6942 033602 012700 034160 MOV #EEP3,R0 ;FSPC
6943 033606 173010 EE8: SUBD (R0),AC0 ;TEST INSTRUCTION
6944 033610 170205 STFPS R5 ;GET FPS
6945 033612 012700 034116 MOV #EEDATO,R0 ;GET THE RESULT
6946 033616 174010 STD AC0,(R0)
6947 033620 012701 034126 MOV #EEP0,R1 ;IS IT CORRECT?
6948 033624 012702 000004 MOV #4,R2
6949 033630 022021 EE9: CMP (R0)+,(R1)+
6950 033632 001415 BEQ EE12
6951 033634 012700 034116 MOV #EEDATO,R0 ;DID A BAD
6952 033640 012701 034170 MOV #EEP4,R1 ;CONSTANT (NOT 57)
6953 033644 012702 000004 MOV #4,R2 ;GET GENERATED
    
```

```

6954 033650 022021          EE10:  CMP      (R0)+,(R1)+          ;FOR THE ALLIGNMENT
6955 033652 001402          BEQ      EE11                      ;FLOWS?
6956 033654 000137 034022    JMP      @#EEER3                    ;DATA ERROR.D
6957 033660 077205          EE11:  SOB      R2,EE10
6958 033662 000137 034060    JMP      @#EEER4                    ;BAD CONSTANT.D
6959 033666 077220          EE12:  SOB      R2,EE9
6960 033670 052704 000004    BIS      #4,R4
6961 033674 020405          CMP      R4,R5                      ;FPS CORRECT?
6962 033676 001402          BEQ      EE13
6963 033700 000137 033710    JMP      @#EEER0                    ;BAD FPS.
6964 033704 000137 034200    EE13:  JMP      @#EEDONE
6965 033710 010437 001242    EEER0:  MOV      R4,@#STMP4          ;BAD FPS
6966 033714 010537 001240    MOV      R5,@#STMP3
6967 033720 104205          1S:   ERROR    205
6968 033722 000137 034200    JMP      @#EEDONE
6969 033726          EEER1:
6970 033726 012737 034140 001240    MOV      #EEP1,@#STMP3
6971 033734 012737 034140 001242    MOV      #EEP1,@#STMP4
6972 033742 012737 034116 001244    MOV      #EEDATO,@#STMP5
6973 033750 012737 034126 001246    MOV      #EEP0,@#STMP6
6974 033756 104206          1S:   ERROR    206
6975 033760 000137 034200    JMP      @#EEDONE
6976 033764          EEER2:
6977 033764 012737 034140 001240    MOV      #EEP1,@#STMP3
6978 033772 012737 034140 001242    MOV      #EEP1,@#STMP4
6979 034000 012737 034116 001244    MOV      #EEDATO,@#STMP5
6980 034006 012737 034126 001246    MOV      #EEP0,@#STMP6
6981 034014 104207          1S:   ERROR    207
6982 034016 000137 034200    JMP      @#EEDONE
6983 034022          EEER3:
6984 034022 012737 034160 001240    MOV      #EEP3,@#STMP3
6985 034030 012737 034160 001242    MOV      #EEP3,@#STMP4
6986 034036 012737 034116 001244    MOV      #EEDATO,@#STMP5
6987 034044 012737 034126 001246    MOV      #EEP0,@#STMP6
6988 034052 104206          1S:   ERROR    206
6989 034054 000137 034200    JMP      @#EEDONE
6990 034060          EEER4:
6991 034060 012737 034160 001240    MOV      #EEP3,@#STMP3
6992 034066 012737 034160 001242    MOV      #EEP3,@#STMP4
6993 034074 012737 034116 001244    MOV      #EEDATO,@#STMP5
6994 034102 012737 034126 001246    MOV      #EEP0,@#STMP6
6995 034110 104207          1S:   ERROR    207
6996 034112 000137 034200    JMP      @#EEDONE
6997 034116 000000          EEDATO: 0
6998 034120 000000          0
6999 034122 000000          0
7000 034124 000000          0
7001 034126 000000          EEER0: 0
7002 034130 000000          0
7003 034132 000000          000000
7004 034134 000000          0
7005 034136 000000          0
7006 034140 000200          EEP1:  200
7007 034142 000000          0
7008 034144 000000          0
7009 034146 000000          0
    
```

7010 034150 000400  
 7011 034152 000000  
 7012 034154 000000  
 7013 034156 000000  
 7014 034160 100200  
 7015 034162 000000  
 7016 034164 000000  
 7017 034166 000000  
 7018 034170 100400  
 7019 034172 000000  
 7020 034174 000000  
 7021 034176 000000  
 7022 034200  
 7023 034200 104412

EEP2: 400  
 0  
 0  
 0  
 EEP3: 100200  
 0  
 0  
 0  
 EEP4: 100400  
 0  
 0  
 0  
 EEDONE:

RSETUP ;GO INITIALIZE THE FPS AND STACK; AND  
 ;SEE IF THE USER HAS EXPRESSED  
 ;THE DESIRE TO CHANGE THE SOFTWARE  
 ;VIRTUAL CONSOLE SWITCH REGISTER (HAS  
 ;THE USER TYPED CONTROL G?).

7024  
 7025  
 7026  
 7027  
 7028  
 7029  
 7030  
 7031  
 7032  
 7033  
 7034  
 7035  
 7036  
 7037

\*\*\*\*\*  
 ;\*TEST 36 NORMALIZE ALGORITHM TEST  
 ;\*  
 ;\* THIS IS A TEST OF THE NORMALIZE  
 ;\* FLOW ALGORITHM. TWO PATTERNS ARE USED,  
 ;\* FIRST THE MINIMUM SITUATION REQUIRING ONE  
 ;\* LEFT SHIFT AND THEN THE MAXIMUM SITUATION  
 ;\* REQUIRING 56 SHIFTS.  
 ;\*  
 ;\*\*\*\*\*

7038 034202 000004  
 7039  
 7040 034204  
 7041 034204 104413  
 7042 034206 012704 003200  
 7043 034212 170104  
 7044 034214 012737 034234 001236  
 7045 034222 012700 034526  
 7046 034226 172410  
 7047 034230 012700 034536  
 7048 034234 172010  
 7049 034236 170205  
 7050 034240 012700 034476  
 7051 034244 174010  
 7052 034246 012701 034546  
 7053 034252 012702 000004  
 7054 034256 022021  
 7055 034260 001401  
 7056 034262 000466  
 7057 034264 077204  
 7058 034266 020405  
 7059 034270 001401  
 7060 034272 000435

TST36: SCOPE  
 ;USE DATA PATTERNS THAT REQUIRE ONLY ONE LEFT SHIFT TO NORMALIZE  
 FF1: LPERR ;SET UP THE LOOP ON ERROR ADDRESS,  
 MOV #3200,R4 ;SET F10, F1V, AND F1D  
 LDFPS R4  
 MOV #FF2,0#STMP2  
 MOV #FFP2,R0 ;SET AC0 OPERAND  
 LDD (R0),AC0  
 MOV #FFP3,R0 ;FSPC  
 FF2: ADDD (R0),AC0 ;TEST INSTRUCTION  
 STFPS R5 ;GET FPS  
 MOV #FFDAT0,R0 ;GET THE RESULT  
 STD AC0,(R0)  
 MOV #FFP4,R1 ;IS IT CORRECT  
 MOV #4,R2  
 FF3: CMP (R0)+,(R1)+  
 BEQ FF4  
 BR FFER2 ;BAD DATA  
 FF4: SOB R2,FF3  
 CMP R4,R5 ;FPS CORRECT?  
 BEQ FF5  
 BR FFER0 ;BAD FPS

7061  
 7062  
 7063 034274  
 7064 034274 104413  
 7065 034276 012704 003200

;USE DATA PATTERNS WHICH REQUIRE 56 LEFT SHIFTS TO NORMALIZE  
 ;THE RESULT  
 FF5: LPERR ;SET UP THE LOOP ON ERROR ADDRESS,  
 MOV #3200,R4 ;SET F1U, F1V, AND F1D



```

7066 034302 170104 LDFPS R4
7067 034304 012737 034324 001236 MOV #FF6,@#STMP2
7068 034312 012700 034506 MOV #FFP0,R0 ;SET AC0 OPERAND
7069 034316 172410 LDD (R0),AC0
7070 034320 012700 034516 MOV #FFP1,R0 ;FSRC
7071 034324 172010 FF6: ADDD (R0),AC0 ;TEST INSTRUCTION
7072 034326 170205 STFPS R5 ;GET FPS
7073 034330 012700 034476 MOV #FFDAT0,R0 ;GET THE RESULT
7074 034334 174010 STD AC0,(R0)
7075 034336 012701 034546 MOV #FFP4,R1 ;IS IT CORRECT
7076 034342 012702 000004 MOV #4,R2
7077 034346 022021 FF7: CMP (R0)+,(R1)+
7078 034350 001401 BEQ FF10
7079 034352 000413 BR FFER1 ;BATA
7080 034354 077204 FF10: SOB R2,FF7
7081 034356 020405 CMP R4,R5 ;FPS CORRECT?
7082 034360 001401 BEQ FF11
7083 034362 000401 BR FFER0 ;BAD FPS
7084 034364 000474 FF11: BR FFDONE
7085
7086 034366 010537 001240 FFER0: MOV R5,@#STMP3
7087 034372 010437 001242 MOV R4,@#STMP4
7088 034376 104164 1S: ERROR 164
7089 034400 000466 BR FFDONE
7090
7091 034402 FFER1:
7092 034402 012737 034516 001240 MOV #FFP1,@#STMP3
7093 034410 012737 034506 001242 MOV #FFP0,@#STMP4
7094 034416 012737 034476 001244 MOV #FFDAT0,@#STMP5
7095 034424 012737 034546 001246 MOV #FFP4,@#STMP6
7096 034432 104210 1S: ERROR 210
7097 034434 000137 034556 JMP @#FFDONE
7098
7099 034440 FFER2:
7100 034440 012737 034536 001240 MOV #FFP3,@#STMP3
7101 034446 012737 034526 001242 MOV #FFP2,@#STMP4
7102 034454 012737 034476 001244 MOV #FFDAT0,@#STMP5
7103 034462 012737 034546 001246 MOV #FFP4,@#STMP6
7104 034470 104210 1S: ERROR 210
7105 034472 000137 034556 JMP @#FFDONE
7106
7107
7108 034476 000000 FFDAT0: 0
7109 034500 000000 0
7110 034502 000000 0
7111 034504 000000 0
7112
7113 034506 016000 FFP0: 16000
7114 034510 000000 0
7115 034512 000000 0
7116 034514 000001 1
7117 034516 116000 FFP1: 116000
7118 034520 000000 0
7119 034522 000000 0
7120 034524 000000 0
7121 034526 000500 FFP2: 500
    
```

```

7122 034530 000000      0
7123 034532 000000      0
7124 034534 000000      0
7125 034536 100400      FFP3: 100400
7126 034540 000000      0
7127 034542 000000      0
7128 034544 000000      0
7129 034546 000200      FFP4: 200      ;FFP4=FFP0+FFP1
7130 034550 000000      0      ;      =FFP3+FFP4
7131 034552 000000      0
7132 034554 000000      0
7133
7134 034556      FFDONE:
7135
7136
7137 034556      IST37:
7138
7139
7140
7141      .SBTTL  END OF PASS ROUTINE
7142
7143      ;*****
7144      ;*INCREMENT THE PASS NUMBER (SPASS)
7145      ;*INDICATE END-OF-PROGRAM AFTER 1 PASSES THRU THE PROGRAM
7146      ;*IF SW12=1 INHIBIT TRACE TRAP
7147      ;*IF THERES A MONITOR GO TO IT
7148      ;*IF THERE ISN'T JUMP TO LOOP
7149
7150 034556      SEOP:
7151 034556 000004      SCOPE
7152 034560 005067 144316      CLR      STSTNM      ;;ZERO THE TEST NUMBER
7153 034564 005067 144512      CLR      STIMES      ;;ZERO THE NUMBER OF ITERATIONS
7154 034570 005267 144530      INC      SPASS      ;;INCREMENT THE PASS NUMBER
7155 034574 042767 100000 144522      BIC      #100000,SPASS      ;;DON'T ALLOW A NEG. NUMBER
7156 034602 005327      DEC      (PC)+      ;;LOOP?
7157 034604 000001      SEOPCT: .WORD 1
7158 034606 003074      BGT      SDOAGN      ;;YES
7159 034610 012737      MOV      (PC)+,@(PC)+      ;;RESTORE COUNTER
7160 034612 000001      SENDCT: .WORD 1
7161 034614 034604      SEOPCT
7162 034616 104401 034624      TYPE    ,65$      ;;TYPE ASCIZ STRING
7163 034622 000407      BR      64$      ;;GET OVER THE ASCIZ
7164      ;;65$: .ASCIZ <12><15>/END PASS #/
7165      64$:
7166 034642 016746 144456      MOV      SPASS,-(SP)      ;;SAVE SPASS FOR TYPEOUT
7167      ;;TYPE PASS NUMBER IN OCTAL
7168 034646 104403      TYPOS      ;;GO TYPE--OCTAL ASCII
7169 034650 006      .BYTE 6      ;;TYPE 6 DIGITS
7170 034651 000      .BYTE 0      ;;SUPPRESS LEADING ZEROS
7171 034652 104401 034660      TYPE    ,67$      ;;TYPE ASCIZ STRING
7172 034656 000421      BR      66$      ;;GET OVER THE ASCIZ
7173      ;;67$: .ASCIZ / TOTAL ERRORS SINCE LAST REPORT /
7174      66$:
7175 034722 016746 144164      MOV      SERTTL,-(SP)      ;;SAVE SERTTL FOR TYPEOUT
7176      ;;TOTAL NUMBER OF ERRORS IN OCTAL
7177 034726 104403      TYPOS      ;;GO TYPE--OCTAL ASCII
  
```

```

7178 034730 006 .BYTE 6 ;;TYPE 6 DIGITS
7179 034731 000 .BYTE 0 ;;SUPPRESS LEADING ZEROS
7180 034732 104401 001313 TYPE ,SCPLF ;;TYPE CARRIAGE RETURN, LINE FEED
7181 034736 005067 144150 CLP $ERTTL ;;CLEAR ERROR TOTAL
7182 034742 013700 000042 $GET42: MOV @#42,R0 ;;GET MONITOR ADDRESS
7183 034746 001414 BEQ $DOAGN ;;BRANCH IF NO MONITOR
7184 034750 005046 CLR -(SP) ;;INSURE THE "T" BIT IS CLEAR
7185 034752 012746 034760 MOV $SCLR.T,-(SP) ;;SETUP FOR AN RTI OR RTT
7186 034756 000426 BR $RTRN ;;GO DO AN RTI OR RTT TO LOAD THE PSW
7187 ;;WITH A CLEARED "T" BIT
7188 034760 $SCLR.T:
7189 034760 013700 000042 MOV @#42,R0 ;;INSURE R0 CONTAINS THE MONITORS
7190 034764 001405 BEQ $DOAGN ;;RETURN ADDRESS
7191 034766 000005 RESET ;;CLEAR THE WORLD
7192 034770 004710 $ENDAD: JSR PC,(R0) ;;GO TO MONITOR
7193 034772 000240 NOP ;;SAVE ROOM
7194 034774 000240 NOP ;;FOR
7195 034776 000240 NOP ;;ACT11
7196 035000 $DOAGN:
7197 035000 104400 TRAP ;;PUSH OLD PSW AND PC ON STACK
7198 035002 042716 000020 BIC #20,(SP) ;;CLEAR THE "T" BIT
7199 035006 032777 010000 144124 BIT #BIT12,@SWR ;;RUN WITH TRACE TRAP?
7200 035014 001005 BNE 1$ ;;BR IF NO
7201 035016 005167 000020 COM $TBIT ;;IS IT TIME FOR TRACE TRAP
7202 035022 100402 BMI 1$ ;;BR IF NO
7203 035024 052716 000020 BIS #20,(SP) ;;SET TRACE TRAP
7204 035030 012746 035036 1$: MOV $SLOOP,-(SP) ;;JUMP TO START OF TEST
7205 035034 000002 $RTRN: RTI ;;RETURN--THIS IS CHANGED TO
7206 ;;AN "RTT" IF "RTI" IS A LEGAL
7207 ;;INSTRUCTION
7208 035036 $LOOP:
7209 035036 000137 JMP @(PC)+ ;;RETURN
7210 035040 004304 $RTNAD: .WORD LOOP
7211 035042 000000 $TBIT: .WORD 0 ;;"T" BIT STATE INDICATOR
7212 035044 377 377 000 $ENULL: .BYTE -1,-1,0 ;;NULL CHARACTER STRING
7213 .EVEN
7214
7215 .SBTTL SCOPE HANDLER ROUTINE
7216
7217 ;;*****
7218 ;*THIS ROUTINE CONTROLS THE LOOPING OF SUBTESTS. IT WILL INCREMENT
7219 ;*AND LOAD THE TEST NUMBER($TSTNM) INTO THE DISPLAY REG.(DISPLAY<7:0>)
7220 ;*AND LOAD THE ERROR FLAG ($ERFLG) INTO DISPLAY<15:08>
7221 ;*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
7222 ;*SW14=1 LOOP ON TEST
7223 ;*SW11=1 INHIBIT ITERATIONS
7224 ;*SW09=1 LOOP ON ERROR
7225 ;*SW08=1 LOOP ON TEST IN SWR<7:0>
7226 ;*CALL
7227 ;* SCOPE ;;SCOPE=IOT
7228
7229 035050 $SCOPE:
7230 035050 104406 CKSWR ;;TEST FOR CHANGE IN SOFT-SWR
7231 035052 032777 040000 144060 1$: BIT #BIT14,@SWR ;;LOOP ON PRESENT TEST?
7232 035060 001114 BNE $OVER ;;YES IF SW14=1
7233 ;####START OF CODE FOR THE XOR TESTER####
    
```

```

7234 035062 000416          SXTSTR: BR      6$          ;; IF RUNNING ON THE "XOR" TESTER CHANGE
7235                                     ;; THIS INSTRUCTION TO A "NOP" (NOP=240)
7236 035064 013746 000004          MOV      @#ERRVEC,-(SP)      ;; SAVE THE CONTENTS OF THE ERROR VECTOR
7237 035070 012737 035110 000004          MOV      $5$,@#ERRVEC      ;; SET FOR TIMEOUT
7238 035076 005737 177060          TST      @#177060          ;; TIME OUT ON XOR?
7239 035102 012637 000004          MOV      (SP)+,@#ERRVEC    ;; RESTORE THE ERROR VECTOR
7240 035106 000463          BR      $SVLAD              ;; GO TO THE NEXT TEST
7241 035110 022626          5$:      CMP      (SP)+,(SP)+    ;; CLEAR THE STACK AFTER A TIME OUT
7242 035112 012637 000004          MOV      (SP)+,@#ERRVEC    ;; RESTORE THE ERROR VECTOR
7243 035116 000423          BR      7$                  ;; LOOP ON THE PRESENT TEST
7244 035120          6$:;####END OF CODE FOR THE XOR TESTER####
7245 035120 032777 000400 144012          BIT      #BIT08,@SWR        ;; LOOP ON SPEC. TEST?
7246 035126 001404          BEQ      2$                  ;; BR IF NO
7247 035130 127767 144004 143744          CMPB    @SWR,$STSTNM        ;; ON THE RIGHT TEST?   SWR<7:0>
7248 035136 001465          BEQ      $OVER              ;; BR IF YES
7249 035140 105767 143737          2$:      TSTB    $ERFLG          ;; HAS AN ERROR OCCURRED?
7250 035144 001421          BEQ      3$                  ;; BR IF NO
7251 035146 126767 143743 143727          CMPB    $ERMAX,$ERFLG      ;; MAX. ERRORS FOR THIS TEST OCCURRED?
7252 035154 101015          BHI      3$                  ;; BR IF NO
7253 035156 032777 001000 143754          BIT      #BIT09,@SWR        ;; LOOP ON ERROR?
7254 035164 001404          BEQ      4$                  ;; BR IF NO
7255 035166 016767 143716 143712          7$:      MOV      $LPERR,$LPADR    ;; SET LOOP ADDRESS TO LAST SCOPE
7256 035174 000446          BR      $OVER
7257 035176 105067 143701          4$:      CLRB    $ERFLG          ;; ZERO THE ERROR FLAG
7258 035202 005067 144074          CLR      $TIMES            ;; CLEAR THE NUMBER OF ITERATIONS TO MAKE
7259 035206 000415          BR      1$                  ;; ESCAPE TO THE NEXT TEST
7260 035210 032777 004000 143722          3$:      BIT      #BIT11,@SWR      ;; INHIBIT ITERATIONS?
7261 035216 001011          BNE      1$                  ;; BR IF YES
7262 035220 005767 144100          TST      $PASS             ;; IF FIRST PASS OF PROGRAM
7263 035224 001406          BEQ      1$                  ;; INHIBIT ITERATIONS
7264 035226 005267 143652          INC      $ICNT             ;; INCREMENT ITERATION COUNT
7265 035232 026767 144044 143644          CMP      $TIMES,$ICNT      ;; CHECK THE NUMBER OF ITERATIONS MADE
7266 035240 002024          BGE      $OVER             ;; BR IF MORE ITERATION REQUIRED
7267 035242 012767 000001 143634          1$:      MOV      #1,$ICNT      ;; REINITIALIZE THE ITERATION COUNTER
7268 035250 016767 000052 144024          MOV      $MXCNT,$TIMES    ;; SET NUMBER OF ITERATIONS TO DO
7269 035256 105267 143620          $SVLAD: INCB    $STSTNM      ;; COUNT TEST NUMBERS
7270 035262 116767 143614 144032          MOVB    $STSTNM,$TESTN    ;; SET TEST NUMBER IN APT MAILBOX
7271 035270 011667 143612          MOV      (SP),$LPADR       ;; SAVE SCOPE LOOP ADDRESS
7272 035274 011667 143610          MOV      (SP),$LPERR       ;; SAVE ERROR LOOP ADDRESS
7273 035300 005067 144000          CLR      $ESCAPE           ;; CLEAR THE ESCAPE FROM ERROR ADDRESS
7274 035304 112767 000001 143603          MOVB    #1,$ERMAX         ;; ONLY ALLOW ONE(1) ERROR ON NEXT TEST
7275 035312 016777 143564 143622          $OVER:  MOV      $STSTNM,@DISPLAY ;; DISPLAY TEST NUMBER
7276 035320 016716 143562          MOV      $LPADR,(SP)      ;; FUDGE RETURN ADDRESS
7277 035324 000002          RTI                       ;; FIXES PS
7278 035326 000001          $MXCNT: 1                  ;; MAX. NUMBER OF ITERATIONS
7279
7280          .SBTTL  ERROR HANDLER ROUTINE
7281
7282          ;*****
7283          ;*THIS ROUTINE WILL INCREMENT THE ERROR FLAG AND THE ERROR COUNT,
7284          ;*SAVE THE ERROR ITEM NUMBER AND THE ADDRESS OF THE ERROR CALL
7285          ;*AND GO TO ERTYPE ON ERROR
7286          ;*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
7287          ;*SW15=1      HALT ON ERROR
7288          ;*SW13=1      INHIBIT ERROR TYPEOUTS
7289          ;*SW10=1     BELL ON ERROR
    
```

```

7290 ;*SW09=1 LOOP ON ERROR
7291 ;*CALL
7292 ;* ERROR N ;;ERROR=EMT AND N=ERROR ITEM NUMBER
7293
7294 035330 SERROR:
7295 035330 104406 CKSWR ;;TEST FOR CHANGE IN SOFT-SWR
7296 035332 105267 143545 7s: INCB SEFFLG ;;SET THE ERROR FLAG
7297 035336 001775 BEQ 7s ;;DON'T LET THE FLAG GO TO ZERO
7298 035340 016777 143536 143574 MOV STSTNM,@DISPLAY ;;DISPLAY TEST NUMBER AND ERROR FLAG
7299 035346 032777 002000 143564 BIT #BIT10,@SWR ;;BELL ON ERROR?
7300 035354 001402 BEQ 1s ;;NO - SKIP
7301 035356 104401 001306 TYPE ,SBELL ;;RING BELL
7302 035362 005267 143524 1s: INC SERTTL ;;COUNT THE NUMBER OF ERRORS
7303 035366 011667 143524 MOV (SP),SERRPC ;;GET ADDRESS OF ERROR INSTRUCTION
7304 035372 162767 000002 143516 SUB #2,SERRPC
7305 035400 117767 143512 143506 MOVB @SERRPC,SITEMB ;;STRIP AND SAVE THE ERROR ITEM CODE
7306 035406 032777 020000 143524 BIT #BIT13,@SWR ;;SKIP TYPEOUT IF SET
7307 035414 001004 BNE 20s ;;SKIP TYPEOUTS
7308 035416 004767 002124 JSR PC,ERTYPE ;;GO TO USER ERROR ROUTINE
7309 035422 104401 001313 TYPE ,SCRLF
7310 035426 20s:
7311 035426 122767 000001 143702 CMPB #APTENV,SENV ;;RUNNING IN APT MODE
7312 035434 001007 BNE 2s ;;NO,SKIP APT ERROR REPORT
7313 035436 116767 143452 000004 MOVB SITEMB,21s ;;SET ITEM NUMBER AS ERROR NUMBER
7314 035444 004767 000740 JSR PC,SATY4 ;;REPORT FATAL ERROR TO APT
7315 035450 000 .BYTE 0
7316 035451 000 .BYTE 0
7317 035452 000777 22s: BR 22s ;;APT ERROR LOOP
7318 035454 005777 143460 2s: TST @SWR ;;HALT ON ERROR
7319 035460 100002 BPL 3s ;;SKIP IF CONTINUE
7320 035462 000000 HALT ;;HALT ON ERROR!
7321 035464 104406 CKSWR ;;TEST FOR CHANGE IN SOFT-SWR
7322 035466 032777 001000 143444 3s: BIT #BIT09,@SWR ;;LOOP ON ERROR SWITCH SET?
7323 035474 001402 BEQ 4s ;;BR IF NO
7324 035476 016716 143406 MOV $LPERR,(SP) ;;FUDGE RETURN FOR LOOPING
7325 035502 005767 143576 4s: TST $ESCAPE ;;CHECK FOR AN ESCAPE ADDRESS
7326 035506 001402 BEQ 5s ;;BR IF NONE
7327 035510 016716 143570 MOV $ESCAPE,(SP) ;;FUDGE RETURN ADDRESS FOR ESCAPE
7328 035514 5s:
7329 035514 022737 034770 000042 CMP #SENDAD,@#42 ;;ACT-11 AUTO-ACCEPT?
7330 035522 001001 BNE 6s ;;BRANCH IF NO
7331 035524 000000 HALT ;;YES
7332 035526 6s:
7333 035526 032777 001000 143404 BIT #BIT09,@SWR
7334 035534 001013 BNE ERM10
7335 035536 011637 001162 MOV (SP),@#$REG0 ;SEE IF ERROR #377
7336 035542 062737 177776 001162 ADD #-2,@#$REG0
7337 035550 122777 000377 143404 CMPB #377,@$REG0
7338 035556 001002 BNE ERM10
7339 035560 062716 000002 ADD #2,(SP)
7340 035564 000002 ERM10: RTI
7341
7342
7343 ,SBTTL SAVE AND RESTORE R0-R5 ROUTINES
7344
7345 ;*****
  
```

```

7346 ;*SAVE R0-R5
7347 ;*CALL:
7348 ;* SAVREG
7349 ;*UPON RETURN FROM $$SAVREG THE STACK WILL LOOK LIKE:
7350 ;*
7351 ;*TOP---(+16)
7352 ;* +2---(+18)
7353 ;* +4---R5
7354 ;* +6---R4
7355 ;* +8---R3
7356 ;*+10---R2
7357 ;*+12---R1
7358 ;*+14---R0
7359
7360 $35566 $SAVREG:
7361 $35566 $10046 MOV R0,-(SP) ;;PUSH R0 ON STACK
7362 $35570 $10146 MOV R1,-(SP) ;;PUSH R1 ON STACK
7363 $35572 $10246 MOV R2,-(SP) ;;PUSH R2 ON STACK
7364 $35574 $10346 MOV R3,-(SP) ;;PUSH R3 ON STACK
7365 $35576 $10446 MOV R4,-(SP) ;;PUSH R4 ON STACK
7366 $35600 $10546 MOV R5,-(SP) ;;PUSH R5 ON STACK
7367 $35602 $16646 000022 MOV 22(SP),-(SP) ;;SAVE PS OF MAIN FLOW
7368 $35606 $16646 000022 MOV 22(SP),-(SP) ;;SAVE PC OF MAIN FLOW
7369 $35612 $16646 000022 MOV 22(SP),-(SP) ;;SAVE PS OF CALL
7370 $35616 $16646 000022 MOV 22(SP),-(SP) ;;SAVE PC OF CALL
7371 $35622 $000022 RTI
7372
7373 ;*RESTORE R0-R5
7374 ;*CALL:
7375 ;* RESREG
7376 $35624 $RESREG:
7377 $35624 $12666 000022 MOV (SP)+,22(SP) ;;RESTORE PC OF CALL
7378 $35630 $12666 000022 MOV (SP)+,22(SP) ;;RESTORE PS OF CALL
7379 $35634 $12666 000022 MOV (SP)+,22(SP) ;;RESTORE PC OF MAIN FLOW
7380 $35640 $12666 000022 MOV (SP)+,22(SP) ;;RESTORE PS OF MAIN FLOW
7381 $35644 $12605 MOV (SP)+,R5 ;;POP STACK INTO R5
7382 $35646 $12604 MOV (SP)+,R4 ;;POP STACK INTO R4
7383 $35650 $12603 MOV (SP)+,R3 ;;POP STACK INTO R3
7384 $35652 $12602 MOV (SP)+,R2 ;;POP STACK INTO R2
7385 $35654 $12601 MOV (SP)+,R1 ;;POP STACK INTO R1
7386 $35656 $12600 MOV (SP)+,R0 ;;POP STACK INTO R0
7387 $35660 $000022 RTI
7388
7389 ;SBTTL TYPE ROUTINE
7390
7391 ;*****
7392 ;*ROUTINE TO TYPE ASCIZ MESSAGE. MESSAGE MUST TERMINATE WITH A 0 BYTE.
7393 ;*THE ROUTINE WILL INSERT A NUMBER OF NULL CHARACTERS AFTER A LINE FEED.
7394 ;*NOTE1: $NULL CONTAINS THE CHARACTER TO BE USED AS THE FILLER CHARACTER.
7395 ;*NOTE2: $FILLS CONTAINS THE NUMBER OF FILLER CHARACTERS REQUIRED.
7396 ;*NOTE3: $FILLC CONTAINS THE CHARACTER TO FILL AFTER.
7397 ;*
7398 ;*CALL:
7399 ;*1) USING A TRAP INSTRUCTION
7400 ;* TYPE ,MESADR ;;MESADR IS FIRST ADDRESS OF AN ASCIZ STRING
7401 ;*OR
    
```

MODEL BUSINESS FORMS, INC. HO  
 1084 1413  
 PRINTED IN U.S.A. 22

```

7402          ;*      TYPE
7403          ;*      MESADR
7404          ;*
7405
7406 035662 105767 143271      STYPE:  TSTB   STPFLG      ;; IS THERE A TERMINAL?
7407 035666 100002          BPL     1$          ;; BR IF YES
7408 035670 000000          HALT          ;; HALT HERE IF NO TERMINAL
7409 035672 000430          BR      3$          ;; LEAVE
7410 035674 010046          1$:   MOV    R0,-(SP)      ;; SAVE R0
7411 035676 017600 000002    MOV    @2(SP),R0      ;; GET ADDRESS OF ASCIZ STRING
7412 035702 122767 000001 143426  CMPB  #APTENV,SENV    ;; RUNNING IN APT MODE
7413 035710 001011          BNE    62$          ;; NO, GO CHECK FOR APT CONSOLE
7414 035712 132767 000100 143417  BITB  #APTPOOL,SENVM  ;; SPOOL MESSAGE TO APT
7415 035720 001405          BEQ    62$          ;; NO, GO CHECK FOR CONSOLE
7416 035722 010067 000004    MOV    R0,61$       ;; SETUP MESSAGE ADDRESS FOR APT
7417 035726 004767 000446    JSR   PC,$ATY3      ;; SPOOL MESSAGE TO APT
7418 035732 000000          61$:  .WORD  0          ;; MESSAGE ADDRESS
7419 035734 132767 000040 143375  62$:  BITB  #APTCSUP,SENVM  ;; APT CONSOLE SUPPRESSED
7420 035742 001003          BNE    60$          ;; YES, SKIP TYPE OUT
7421 035744 112046          2$:   MOVB  (R0)+,-(SP)  ;; PUSH CHARACTER TO BE TYPED ONTO STACK
7422 035746 001005          BNE    4$          ;; BR IF IT ISN'T THE TERMINATOR
7423 035750 005726          TST   (SP)+         ;; IF TERMINATOR POP IT OFF THE STACK
7424 035752 012600          60$:  MOV    (SP)+,R0      ;; RESTORE R0
7425 035754 062716 000002    3$:   ADD    #2,(SP)     ;; ADJUST RETURN PC
7426 035760 000002          RTI          ;; RETURN
7427 035762 122716 000011    4$:   CMPB  #HT,(SP)    ;; BRANCH IF <HT>
7428 035766 001430          BEQ    8$          ;; BRANCH IF NOT <CRLF>
7429 035770 122716 000200    RNE    5$          ;; POP <CR><LF> EQUIV
7430 035774 001006          TST   (SP)+         ;; TYPE A CR AND LF
7431 035776 005726          TYPE          ;; CLEAR CHARACTER COUNT
7432 036000 104401          SCRLF          ;; GET NEXT CHARACTER
7433 036002 001313          CLRBR  $CHARCNT     ;; GO TYPE THIS CHARACTER
7434 036004 105067 000130    BR     2$          ;; IS IT TIME FOR FILLER CHARS.?
7435 036010 000755          5$:   JSR   PC,$TYPEC    ;; IF NO GO GET NEXT CHAR.
7436 036012 004767 000056    6$:   CMPB  $FILLC,(SP)+  ;; GET # OF FILLER CHARS. NEEDED
7437 036016 126726 143134    BNE    2$          ;; AND THE NULL CHAR.
7438 036022 001350          MOV    $NULL,-(SP)  ;; DOES A NULL NEED TO BE TYPED?
7439 036024 016746 143124    7$:   DECB  1(SP)       ;; BR IF NO--GO POP THE NULL OFF OF STACK
7440          BLT     6$          ;; GO TYPE A NULL
7441 036030 105366 000001    JSR   PC,$TYPEC    ;; DO NOT COUNT AS A COUNT
7442 036034 002770          DECB  $CHARCNT     ;; LOOP
7443 036036 004767 000032    BR     7$
7444 036042 105367 000072
7445 036046 000770
7446
7447          ;HORIZONTAL TAB PROCESSOR
7448
7449 036050 112716 000040    8$:   MOVB  #' ,(SP)    ;; REPLACE TAB WITH SPACE
7450 036054 004767 000014    9$:   JSR   PC,$TYPEC    ;; TYPE A SPACE
7451 036060 132767 000007 000052  BITB  #7,$CHARCNT    ;; BRANCH IF NOT AT
7452 036066 001372          BNE    9$          ;; TAB STOP
7453 036070 005726          TST   (SP)+         ;; POP SPACE OFF STACK
7454 036072 000724          BR     2$          ;; GET NEXT CHARACTER
7455 036074 105777 143050    STYPEC: TSTB  @STPS    ;; WAIT UNTIL PRINTER IS READY
7456 036100 100375          BPL   $TYPEC
7457 036102 116677 000002 143042  MOVB  2(SP),@STPB    ;; LOAD CHAR TO BE TYPED INTO DATA REG.
  
```

```

7458 036110 122766 000015 000002      CMPB   #CR,2(SP)      ;; IS CHARACTER A CARRIAGE RETURN?
7459 036116 001003                      BNE    1$            ;; BRANCH IF NO
7460 036120 145067 000014                      CLRB   SCHARCNT     ;; YES--CLEAR CHARACTER COUNT
7461 036124 000406                      BR     STYPEX       ;; EXIT
7462 036126 122766 000012 000002 1$:    CMPB   #LF,2(SP)     ;; IS CHARACTER A LINE FEED?
7463 036134 001402                      BEQ    STYPEX       ;; BRANCH IF YES
7464 036136 105227                      INCB   (PC)+        ;; COUNT THE CHARACTER
7465 036140 000000                      SCHARCNT: WORD 0    ;; CHARACTER COUNT STORAGE
7466 036142 000207                      STYPEX: RTS        PC
7467
7468
7469                      .SBTTL  BINARY TO OCTAL (ASCII) AND TYPE
7470
7471                      ;*****
7472                      ;*THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 6-DIGIT
7473                      ;*OCTAL (ASCII) NUMBER AND TYPE IT.
7474                      ;*STYPOS---ENTER HERE TO SETUP SUPPRESS ZEROS AND NUMBER OF DIGITS TO TYPE
7475                      ;*CALL:
7476                      ;*      MOV     NUM,-(SP)      ;;NUMBER TO BE TYPED
7477                      ;*      TYPOS                    ;;CALL FOR TYPEOUT
7478                      ;*      .BYTE  N              ;;N=1 TO 6 FOR NUMBER OF DIGITS TO TYPE
7479                      ;*      .BYTE  M              ;;M=1 OR 0
7480                      ;*                                  ;;1=TYPE LEADING ZEROS
7481                      ;*                                  ;;0=SUPPRESS LEADING ZEROS
7482                      ;*
7483                      ;*STYPON---ENTER HERE TO TYPE OUT WITH THE SAME PARAMETERS AS THE LAST
7484                      ;*STYPOS OR STYPOC
7485                      ;*CALL:
7486                      ;*      MOV     NUM,-(SP)      ;;NUMBER TO BE TYPED
7487                      ;*      TYPON                    ;;CALL FOR TYPEOUT
7488                      ;*
7489                      ;*STYPOC---ENTER HERE FOR TYPEOUT OF A 16 BIT NUMBER
7490                      ;*CALL:
7491                      ;*      MOV     NUM,-(SP)      ;;NUMBER TO BE TYPED
7492                      ;*      TYPOC                    ;;CALL FOR TYPEOUT
7493
7494 036144 017646 000000                      STYPOS: MOV     0(SP),-(SP)  ;; PICKUP THE MODE
7495 036150 116667 000001 000211          MOVB   1(SP),S0FILL      ;; LOAD ZERO FILL SWITCH
7496 036156 112667 000207                      MOVB   (SP)+,S0MODE+1   ;; NUMBER OF DIGITS TO TYPE
7497 036162 062716 000002                      ADD    #2,(SP)          ;; ADJUST RETURN ADDRESS
7498 036166 000406                      BR     STYPON
7499 036170 112767 000001 000171          STYPOC: MOVB   #1,S0FILL  ;; SET THE ZERO FILL SWITCH
7500 036176 112767 000006 000165          MOVB   #6,S0MODE+1     ;; SET FOR SIX(6) DIGITS
7501 036204 112767 000005 000154          STYPON: MOVB   #5,S0CNT  ;; SET THE ITERATION COUNT
7502 036212 010346                      MOV    R3,-(SP)        ;; SAVE R3
7503 036214 010446                      MOV    R4,-(SP)        ;; SAVE R4
7504 036216 010546                      MOV    R5,-(SP)        ;; SAVE R5
7505 036220 116704 000145                      MOVB   S0MODE+1,R4     ;; GET THE NUMBER OF DIGITS TO TYPE
7506 036224 005404                      NEG    R4
7507 036226 062704 000006                      ADD    #6,R4           ;; SUBTRACT IT FOR MAX, ALLOWED
7508 036232 110467 000132                      MOVB   R4,S0MODE       ;; SAVE IT FOR USE
7509 036236 116704 000125                      MOVB   S0FILL,R4       ;; GET THE ZERO FILL SWITCH
7510 036242 016605 000012                      MOV    12(SP),R5      ;; PICKUP THE INPUT NUMBER
7511 036246 005003                      CLR    R3              ;; CLEAR THE OUTPUT WORD
7512 036250 006105                      1$:    ROL    R5         ;; ROTATE MSB INTO "C"
7513 036252 000404                      BR     3$              ;; GO DO MSB
    
```

FORM 1413 PRINTED IN U.S.A.



```

7514 036254 006105          2S:   ROL   R5           ;;FORM THIS DIGIT
7515 036256 006105          ROL   R5
7516 036260 006105          ROL   P5
7517 036262 010503          MOV   R5,R3
7518 036264 006103          3S:   ROL   R3           ;;GET LSH OF THIS DIGIT
7519 036266 105367 000076  DECB  $OMODE        ;;TYPE THIS DIGIT?
7520 036272 100016          BPL   7$           ;;BR IF NO
7521 036274 042703 177770  BIC   #177770,R3   ;;GET RID OF JUNK
7522 036300 001002          BNE   4$           ;;TEST FOR 0
7523 036302 005704          TST   R4           ;;SUPPRESS THIS 0?
7524 036304 001403          BEQ   5$           ;;BR IF YES
7525 036306 005204          4S:   INC   R4           ;;DON'T SUPPRESS ANYMORE 0'S
7526 036310 052703 000060  BIS   #'0,R3       ;;MAKE THIS DIGIT ASCII
7527 036314 052703 000040  5S:   BIS   #' ,R3   ;;MAKE ASCII IF NOT ALREADY
7528 036320 110367 000040  MOVB  R3,R6        ;;SAVE FOR TYPING
7529 036324 104401 036364  TYPE  ,8$          ;;GO TYPE THIS DIGIT
7530 036330 105367 000032  7S:   DECB  $OCNT    ;;COUNT BY 1
7531 036334 003347          BGT   2$           ;;BR IF MORE TO DO
7532 036336 002402          BLT   6$           ;;BR IF DONE
7533 036340 005204          INC   R4           ;;INSURE LAST DIGIT ISN'T A BLANK
7534 036342 000744          BR    2$           ;;GO DO THE LAST DIGIT
7535 036344 012605          6S:   MOV   (SP)+,R5  ;;RESTORE R5
7536 036346 012604          MOV   (SP)+,R4    ;;RESTORE R4
7537 036350 012603          MOV   (SP)+,R3    ;;RESTORE R3
7538 036352 016666 000002 000004  MOV   2(SP),4(SP)  ;;SET THE STACK FOR RETURNING
7539 036360 012616          MOV   (SP)+,(SP)
7540 036362 000002          RTI                    ;;RETURN
7541 036364 000          8S:   .BYTE 0          ;;STORAGE FOR ASCII DIGIT
7542 036365 000          .BYTE 0          ;;TERMINATOR FOR TYPE ROUTINE
7543 036366 000          $OCNT: .BYTE 0     ;;OCTAL DIGIT COUNTER
7544 036367 000          $OFILL: .BYTE 0   ;;ZERO FILL SWITCH
7545 036370 000000          $OMODE: .WORD 0    ;;NUMBER OF DIGITS TO TYPE
7546
7547          .SBTTL  APT COMMUNICATIONS ROUTINE
7548
7549          ;*****
7550 036372 112767 000001 000236  SATY1: MOVB  #1,$FFLG  ;;TO REPORT FATAL ERROR
7551 036400 112767 000001 000226  SATY3: MOVB  #1,$MFLG  ;;TO TYPE A MESSAGE
7552 036406 000403          BR    SATYC
7553 036410 112767 000001 000220  SATY4: MOVB  #1,$FFLG  ;;TO ONLY REPORT FATAL ERROR
7554 036416          SATYC:
7555 036416 010046          MOV   R0,-(SP)     ;;PUSH R0 ON STACK
7556 036420 010146          MOV   R1,-(SP)     ;;PUSH R1 ON STACK
7557 036422 105767 000206          TSTB  $MFLG        ;;SHOULD TYPE A MESSAGE?
7558 036426 001450          BEQ   5$           ;;IF NOT: BR
7559 036430 122767 000001 142700  CMPB  #APTENV,$ENV  ;;OPERATING UNDER APT?
7560 036436 001031          ENE   3$           ;;IF NOT: BR
7561 036440 132767 000100 142671  BITB  #APTPOOL,$ENVM ;;SHOULD SPOOL MESSAGES?
7562 036446 001425          BEQ   3$           ;;IF NOT: BR
7563 036450 017600 000004          MOV   #4(SP),R0    ;;GET MESSAGE ADDR.
7564 036454 062766 000002 000004  ADD   #2,4(SP)      ;;BUMP RETURN ADDR.
7565 036462 005767 142630          1S:   TST   $MSGTYPE  ;;SEE IF DONE W/ LAST XMISSION?
7566 036466 001375          BNE   1$           ;;IF NOT: WAIT
7567 036470 010067 142636          MOV   R0,$MSGAD    ;;PUT ADDR IN MAILBOX
7568 036474 105720          2S:   TSTB  (R0)+     ;;FIND END OF MESSAGE
7569 036476 001376          BNE   2$

```

```

7570 036500 166700 142626 SUB $MSGAD,R0 ;;SUB START OF MESSAGE
7571 036501 006200 ASR R0 ;;GET MESSAGE LNGLH IN WORDS
7572 036505 010067 142622 MOV R0,$MSGGLT ;;PUT LENGTH IN MAILBOX
7573 036512 012767 000004 142576 MOV #4,$MSGTYPE ;;TELL APT TO TAKE MSG.
7574 036520 000413 BR 5S
7575 036522 017667 000004 000016 3S: MOV @4(SP),4S ;;PUT MSG ADDR IN JSR LINKAGE
7576 036530 062766 000002 000004 ADD #2,4(SP) ;;BUMP RETURN ADDRESS
7577 036536 016746 141234 MOV 177776,-(SP) ;;PUSH 177776 ON STACK
7578 036542 004767 177114 JSR PC,STYPE ;;CALL TYPE MACRO
7579 036546 000000 4S: .WORD 0
7580 036550 5S:
7581 036550 105767 000062 10S: TSTB $FFLG ;;SHOULD REPORT FATAL ERROR?
7582 036554 001416 BEQ 12S ;;IF NOT: BR
7583 036556 005767 142554 TST $ENV ;;RUNNING UNDER APT?
7584 036562 001413 BEQ 12S ;;IF NOT: BR
7585 036564 005767 142526 11S: TST $MSGTYPE ;;FINISHED LAST MESSAGE?
7586 036570 001375 BNE 11S ;;IF NOT: WAIT
7587 036572 017667 000004 142520 MOV @4(SP),$FATAL ;;GET ERROR #
7588 036600 062766 000002 000004 ADD #2,4(SP) ;;BUMP RETURN ADDR.
7589 036606 005267 142504 INC $MSGTYPE ;;TELL APT TO TAKE ERROR
7590 036612 105067 000020 12S: CLRB $FFLG ;;CLEAR FATAL FLAG
7591 036616 105067 000013 CLRB $LFLG ;;CLEAR LOG FLAG
7592 036622 105067 000006 CLRB $MFLG ;;CLEAR MESSAGE FLAG
7593 036626 012601 MOV (SP)+,R1 ;;POP STACK INTO R1
7594 036630 012600 MOV (SP)+,R0 ;;POP STACK INTO R0
7595 036632 000207 RTS PC ;;RETURN
7596 036634 000 SMFLG: .BYTE 0 ;;MESSG. FLAG
7597 036635 000 SLFLG: .BYTE 0 ;;LOG FLAG
7598 036636 000 SFFLG: .BYTE 0 ;;FATAL FLAG
7599 036640 .EVEN
7600 000200 APTSIZE=200
7601 000001 APTENV=001
7602 000100 APTSPool=100
7603 000040 APTCSUP=040
7604
7605 .SBTTL TTY INPUT ROUTINE
7606
7607 ;*****
7608 .ENABL LSB
7609
7610 ;*****
7611 ;*SOFTWARE SWITCH REGISTER CHANGE ROUTINE.
7612 ;*ROUTINE IS ENTERED FROM THE TRAP HANDLER, AND WILL
7613 ;*SERVICE THE TEST FOR CHANGE IN SOFTWARE SWITCH REGISTER TRAP CALL
7614 ;*WHEN OPERATING IN TTY FLAG MODE.
7615 036640 022767 000176 142272 SCKSWR: CMP #SWREG,SWR ;;IS THE SOFT-SWR SELECTED?
7616 036646 001074 BNE 15S ;;BRANCH IF NO
7617 036650 105777 142270 TSTB @STKS ;;CHAR THERE?
7618 036654 100071 BPL 15S ;;IF NO, DON'T WAIT AROUND
7619 036656 117746 142264 MOVB @STKB,-(SP) ;;SAVE THE CHAR
7620 036662 042716 177600 BIC #~C177,(SP) ;;STRIP-OFF THE ASCII
7621 036666 022726 000007 CMP #7,(SP)+ ;;IS IT A CONTROL G?
7622 036672 001062 BNE 15S ;;NO, RETURN TO USER
7623 036674 126727 142234 000001 CMPB $AUTOB,#1 ;;ARE WE RUNNING IN AUTO-MODE?
7624 036702 001456 BEQ 15S ;;BRANCH IF YES
7625

```

```

7626 036704 104401 037247          TYPE      ,SCNTLG      ;;ECHO THE CONTROL-G (^G)
7627 036710 104401 037254          SGTSWR: TYPE      ,SMSWR      ;;TYPE CURRENT CONTENTS
7628 036714 016746 141256          MOV       SWREG,-(SP)  ;;SAVE SWREG FOR TYPEOUT
7629 036720 104402          TYPEOC     ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
7630 036722 104401 037265          TYPE      ,SMNEW      ;;PROMPT FOR NEW SWR
7631 036726 005046          19$: CLR      -(SP)    ;;CLEAR COUNTER
7632 036730 005046          CLR      -(SP)      ;;THE NEW SWR
7633 036732 105777 142206          7$: TSTB     @STKS     ;;CHAR THERE?
7634 036736 100375          BPL       7$        ;;IF NOT TRY AGAIN
7635
7636 036740 117746 142202          MOVB     @STKB,-(SP)  ;;PICK UP CHAR
7637 036744 042716 177600          BIC      #'C177,(SP) ;;MAKE IT 7-BIT ASCII
7638
7639
7640
7641 036750 021627 000025          9$: CMP      (SP),#25  ;;IS IT A CONTROL-U?
7642 036754 001005          BNE      10$        ;;BRANCH IF NOT
7643 036756 104401 037242          TYPE      ,SCNTLU    ;;YES, ECHO CONTROL-U (^U)
7644 036762 062706 000006          20$: ADD     #6,SP    ;;IGNORE PREVIOUS INPUT
7645 036766 000757          BR       19$        ;;LET'S TRY IT AGAIN
7646
7647
7648 036770 021627 000015          10$: CMP     (SP),#15  ;;IS IT A <CR>?
7649 036774 001022          BNE      16$        ;;BRANCH IF NO
7650 036776 005766 000004          TST      4(SP)      ;;YES, IS IT THE FIRST CHAR?
7651 037002 001403          BEQ      11$        ;;BRANCH IF YES
7652 037004 016677 000002 142126          MOV      2(SP),@SWR  ;;SAVE NEW SWR
7653 037012 062706 000006          11$: ADD     #6,SP    ;;CLEAR UP STACK
7654 037016 104401 001313          14$: TYPE    ,SCRLF    ;;ECHO <CR> AND <LF>
7655 037022 126727 142107 000001          CMPB     $INTAG,#1  ;;RE-ENABLE TTY KBD INTERRUPTS?
7656 037030 001003          BNE      15$        ;;BRANCH IF NOT
7657 037032 012777 000100 142104          MOV      #100,@STKS ;;RE-ENABLE TTY KBD INTERRUPTS
7658 037040 000002          15$: RTI                    ;;RETURN
7659 037042 004767 177026          16$: JSR     PC,$TYPEC  ;;ECHO CHAR
7660 037046 021627 000060          CMP      (SP),#60   ;;CHAR < 0?
7661 037052 002420          BLT      18$        ;;BRANCH IF YES
7662 037054 021627 000067          CMP      (SP),#67   ;;CHAR > 7?
7663 037060 003015          BGT      18$        ;;BRANCH IF YES
7664 037062 042726 000060          BIC      #60,(SP)+  ;;STRIP-OFF ASCII
7665 037066 005766 000002          TST      2(SP)      ;;IS THIS THE FIRST CHAR
7666 037072 001403          BEQ      17$        ;;BRANCH IF YES
7667 037074 006316          ASL      (SP)       ;;NO, SHIFT PRESENT
7668 037076 006316          ASL      (SP)       ;; CHAR OVER TO MAKE
7669 037100 006316          ASL      (SP)       ;; ROOM FOR NEW ONE,
7670 037102 005266 000002          17$: INC     2(SP)    ;;KEEP COUNT OF CHAR
7671 037106 056616 177776          BIS      -2(SP),(SP) ;;SET IN NEW CHAR
7672 037112 000707          BP       7$        ;;GET THE NEXT ONE
7673 037114 104401 001312          18$: TYPE    ,SQUES    ;;TYPE ?<CR><LF>
7674 037120 000720          BR       20$        ;;SIMULATE CONTROL-U
7675          ,DSABL  LSB
7676
7677
7678          ;*****
7679          ;*THIS ROUTINE WILL INPUT A SINGLE CHARACTER FROM THE TTY
7680          ;*CALL:
7681          ;*      RDCHR          ;;INPUT A SINGLE CHARACTER FROM THE TTY
    
```

PRINTED IN U.S.A.  
 FORM 1413  
 PERIODICALS DIVISION, GPO

```

7682 ;* RETURN HERE ;; CHARACTER IS ON THE STACK
7683 ;* ;; WITH PARITY BIT STRIPPED OFF
7684 ;
7685
7686 037122 011646 SRDCHR: MOV (SP),-(SP) ;; PUSH DOWN THE PC
7687 037124 016566 000004 000002 MOV 4(SP),2(SP) ;; SAVE THE PS
7688 037132 105777 142006 1S: TSTB @STKS ;; WAIT FOR
7689 037136 120375 BFL 1S ;; A CHARACTER
7690 037140 117766 142002 000004 MOVB @STKB,4(SP) ;; READ THE TTY
7691 037146 042766 177600 000004 BIC #^C<177>,4(SP) ;; GET RID OF JUNK IF ANY
7692 037154 026627 000004 000023 CMP 4(SP),#23 ;; IS IT A CONTROL-S?
7693 037162 001013 BNE 3S ;; BRANCH IF NO
7694 037164 105777 141754 2S: TSTB @STKS ;; WAIT FOR A CHARACTER
7695 037170 100375 BPL 2S ;; LOOP UNTIL ITS THERE
7696 037172 117746 141750 MOVB @STKB,-(SP) ;; GET CHARACTER
7697 037176 042716 177600 BIC #^C177,(SP) ;; MAKE IT 7-BIT ASCII
7698 037202 022627 000021 CMP (SP)+,#21 ;; IS IT A CONTROL-Q?
7699 037206 001366 BNE 2S ;; IF NOT DISCARD IT
7700 037210 000750 BR 1S ;; YES, RESUME
7701 037212 026627 000004 000140 3S: CMP 4(SP),#140 ;; IS IT UPPER CASE?
7702 037220 002407 BLT 4S ;; BRANCH IF YES
7703 037222 026627 000004 000175 CMP 4(SP),#175 ;; IS IT A SPECIAL CHAR?
7704 037230 003003 BGT 4S ;; BRANCH IF YES
7705 037232 042766 000040 000004 BIC #40,4(SP) ;; MAKE IT UPPER CASE
7706 037240 000002 4S: RTI ;; GO BACK TO USER
7707 037242 052536 005015 000 SCNTLU: .ASCIZ /^U/<15><12> ;; CONTROL "U"
7708 037247 136 006507 000012 SCNTLG: .ASCIZ /^G/<15><12> ;; CONTROL "G"
7709 037254 005015 053523 020122 SMSWR: .ASCIZ <15><12>/SWR = /
7710 037262 020075 000
7711 037265 040 047040 053505 SMNEW: .ASCIZ / NEW = /
7712 037272 036440 000040
7713
7714 .SBTTL TRAP DECODER
7715
7716 ;*****
7717 ;*THIS ROUTINE WILL PICKUP THE LOWER BYTE OF THE "TRAP" INSTRUCTION
7718 ;*AND USE IT TO INDEX THROUGH THE TRAP TABLE FOR THE STARTING ADDRESS
7719 ;*OF THE DESIRED ROUTINE. THEN USING THE ADDRESS OBTAINED IT WILL
7720 ;*GO TO THAT ROUTINE.
7721
7722 037276 010046 STRAP: MOV R0,-(SP) ;; SAVE R0
7723 037300 016600 000002 MOV 2(SP),R0 ;; GET TRAP ADDRESS
7724 037304 005740 TST -(R0) ;; BACKUP BY 2
7725 037306 111000 MOVB (R0),R0 ;; GET RIGHT BYTE OF TRAP
7726 037310 006300 ASL R0 ;; POSITION FOR INDEXING
7727 037312 016000 037332 MOV STRPAD(R0),R0 ;; INDEX TO TABLE
7728 037316 000200 RTS R0 ;; GO TO ROUTINE
7729
7730
7731 ;;THIS IS USE TO HANDLE THE "GETPRI" MACRO
7732
7733 037320 011646 STRAP2: MOV (SP),-(SP) ;; MOVE THE PC DOWN
7734 037322 016666 000004 000002 MOV 4(SP),2(SP) ;; MOVE THE PSW DOWN
7735 037330 000002 RTI ;; RESTORE THE PSW
7736
7737 .SBTTL TRAP TABLE
  
```

```

7738
7739 ;*THIS TABLE CONTAINS THE STARTING ADDRESSES OF THE ROUTINES CALLED
7740 ;*BY THE "TRAP" INSTRUCTION.
7741
7742 ; ROUTINE
7743 ; -----
7744 037332 037320 STRPAD: .WORD STRPAD
7745 037334 035662 $TYPE ;;CALL=TYPE TRAP+1(104401) TTY TYPEOUT ROUTINE
7746 037336 036170 $TYPOC ;;CALL=TYPOC TRAP+2(104402) TYPE OCTAL NUMBER (WITH LEADING ZEROS)
7747 037340 036144 $TYPOS ;;CALL=TYPOS TRAP+3(104403) TYPE OCTAL NUMBER (NO LEADING ZEROS)
7748 037342 036204 $TYPON ;;CALL=TYPON TRAP+4(104404) TYPE OCTAL NUMBER (AS PER LAST CALL)
7749
7750 037344 036710 $GTSWR ;;CALL=GTSWP TRAP+5(104405) GET SOFT-SWR SETTING
7751
7752 037346 036640 $CKSWR ;;CALL=CKSWP TRAP+6(104406) TEST FOR CHANGE IN SOFT-SWR
7753 037350 037122 $RDCHR ;;CALL=RDCHR TRAP+7(104407) TTY TYPEIN CHARACTER ROUTINE
7754 037352 035566 $SAVREG ;;CALL=SAVREG TRAP+10(104410) SAVE R0-R5 ROUTINE
7755 037354 035624 $RESREG ;;CALL=RESREG TRAP+11(104411) RESTORE R0-R5 ROUTINE
7756 037356 040274 .RSET ;;CALL=RSETUP TRAP+12(104412) ROUTINE TO RESET STACK AND FPS
7757 037360 040266 .LPER ;;CALL=LPERF TRAP+13(104413) ROUTINE TO SET LOOP ON ERROR ADDRESS
7758 000030
7759
7760 .SBTTL POWER DOWN AND UP ROUTINES
7761
7762 ;*****
7763 ;POWER DOWN ROUTINE
7764 037362 012737 037540 000024 $PWRDN: MOV #SILLUP,@#PWRVEC ;;SET FOR FAST UP
7765 037370 012737 000340 000026 MOV #340,@#PWRVEC+2 ;;PRIO:7
7766 037376 010046 MOV R0,-(SP) ;;PUSH R0 ON STACK
7767 037400 010146 MOV R1,-(SP) ;;PUSH R1 ON STACK
7768 037402 010246 MOV R2,-(SP) ;;PUSH R2 ON STACK
7769 037404 010346 MOV R3,-(SP) ;;PUSH R3 ON STACK
7770 037406 010446 MOV R4,-(SP) ;;PUSH R4 ON STACK
7771 037410 010546 MOV R5,-(SP) ;;PUSH R5 ON STACK
7772 037412 017746 141522 MOV @SWR,-(SP) ;;PUSH @SWR ON STACK
7773 037416 010667 000122 MOV SP,$SAVP6 ;;SAVE SP
7774 037422 012737 037434 000024 MOV #SPWRUP,@#PWRVEC ;;SET UP VECTOR
7775 037430 000000 HALT
7776 037432 000776 BR .-2 ;;HANG UP
7777
7778 ;*****
7779 ;POWER UP ROUTINE
7780 037434 012737 037540 000024 $PWRUP: MOV #SILLUP,@#PWRVEC ;;SET FOR FAST DOWN
7781 037442 016706 000076 MOV $SAVR6,SP ;;GET SP
7782 037446 005067 000072 CLR $SAVR6 ;;WAIT LOOP FOR THE TTY
7783 037452 005267 000066 1$: INC $SAVR6 ;;WAIT FOR THE INC
7784 037456 001375 BNE 1$ ;;OF WORD
7785 037460 012677 141454 MOV (SP)+,@SWR ;;POP STACK INTO @SWR
7786 037464 012605 MOV (SP)+,R5 ;;POP STACK INTO R5
7787 037466 012604 MOV (SP)+,R4 ;;POP STACK INTO R4
7788 037470 012603 MOV (SP)+,R3 ;;POP STACK INTO R3
7789 037472 012602 MOV (SP)+,R2 ;;POP STACK INTO R2
7790 037474 012601 MOV (SP)+,R1 ;;POP STACK INTO R1
7791 037476 012600 MOV (SP)+,R0 ;;POP STACK INTO R0
7792 037500 012737 037362 000024 MOV #SPWRDN,@#PWRVEC ;;SET UP THE POWER DOWN VECTOR
7793 037506 012737 000340 000026 MOV #340,@#PWRVEC+2 ;;PRIO:7

```

```

7794 037514 104401                                TYPE                                ;;REPORT THE POWER FAILURE
7795 037516 040344                                SPWRMG: .WORD POWERM                ;;POWER FAIL MESSAGE POINTER
7796 037520 012716                                MOV (PC)+,(SP)                       ;;RESTART AT START
7797 037522 003606                                SPWRAD: .WORD START                  ;;RESTART ADDRESS
7798 037524 042766 000020 000002                BIC #20,2(SP)                         ;;CLEAR "T" BIT
7799 037532 005067 175304                        CLR STBIT                               ;;CLEAR THE "T" BIT FLAG
7800 037536 000002                                RTI
7801 037540 000000                                $ILLUP: HALT                          ;;THE POWER UP SEQUENCE WAS STARTED
7802 037542 000776                                BR .-2                                 ;; BEFORE THE POWER DOWN WAS COMPLETE
7803 037544 000000                                $$SAVR6: 0                             ;;PUT THE SP HERE
7804
7805
7806
7807                                .SBTTL ERROR TYPE OUT ROUTINE
7808                                ;*****
7809                                ;*****
7810                                ;*THIS ROUTINE IS CALLED TO TYPE AN ERROR MESSAGE WHICH IS INCLUDED
7811                                ;*IN THE ERROR MESSAGE DATA TABLE. IT IS CALLED BY THE SERROR ROUTINE
7812                                ;*OR BY FIRST SETTING $ITEMB EQUAL TO THE ERROR TABLE ITEM TO BE PRINTED
7813                                ;*OUT AND THEN EXECUTING A;
7814                                ;*
7815                                JSP PC,ERTYPE
7816                                ;*
7817                                ERTYPE: TYPE                                ;TYPE A CRLF
7818                                .WORD $CRLF
7819                                MOVB @#$STSTNM,@$STMP0
7820                                BIC #177400,@$STMP0
7821                                MOV @$ERRPC,@$STMP1                                ;GET PC OF CALL
7822                                MOV R0,-(SP)                                ;SAVE R0
7823                                MOVB @$ITEMB,R0                                ;GET THE ITEM NUMBER.
7824                                BIC #177400,R0
7825                                BNE 1$
7826                                MOV @$ERRPC,-(SP)                                ;IF ZERO THEN JUST
7827                                TYPOC                                ;PRINT THE PC
7828                                JMP @#ERT5
7829
7830                                1$: CMP #377,R0
7831                                BNE 20$
7832                                MOV 4(SP),R0
7833                                MOV (R0),R0
7834                                ADD #400,R0
7835                                20$: DEC R0                                ;OTHERWISE MAKE R0 AN
7836                                ASL R0                                ;INDEX FOR THE TABLE.
7837                                ASL R0
7838                                ASL R0
7839                                ADD #$ERRTB,R0
7840
7841                                MOV (R0)+,@#2$                                ;PICK UP THE ADDRESS
7842                                BEQ 3$                                ;OF THE EM, ERROR MESSAGE
7843                                TYPE
7844                                2$: .WORD 0
7845                                TYPE
7846                                .WORD $CRLF
7847
7848                                3$: MOV (R0)+,@#4$                                ;GET THE DH,DATA HEADER
7849                                BEQ 5$
  
```

FORM 1413  
 PRINTED IN U.S.A.  
 MICROFILMED FOR THE IBM

7850	037702	104401			TYPE		
7851	037704	000000		4\$:	.WORD	0	
7852	037706	104401			TYPE		
7853	037710	001313			.WORD	SCRLF	
7854							
7855	037712	010146		5\$:	MOV	R1,-(SP)	;SAVE R1,R2 AND R3
7856	037714	010246			MOV	R2,-(SP)	
7857	037716	010346			MOV	R3,-(SP)	
7858							
7859	037720	012001			MOV	(R0)+,R1	;GET THE ADDRESS OF THE
7860							;DATA TABLE.
7861	037722	001001			BNE	6\$	
7862	037724	000516			BR	ERT4	;RETURN IF NO DATA.
7863							
7864	037726	011000		6\$:	MOV	(R0),R0	;GET A POINTER TO THE DATA
7865							;FORMAT TABLE.
7866	037730	105710		ERT1:	TSTB	(R0)	;FORMAT ZERO?
7867	037732	001003			BNE	7\$	
7868							
7869	037734	013146			MOV	@(R1)+,-(SP)	;FORMAT ZERO SO TYPE
7870	037736	104402			TYPOC		;AN OCTAL NUMBER.
7871	037740	000502			BR	ERT2	
7872							
7873	037742			7\$:			
7874	037742	122710	000002	8\$:	CMPB	#2,(R0)	;FORMAT TWO?
7875	037746	001010			BNE	9\$	
7876							
7877	037750	013102			MOV	@(R1)+,R2	;FORMAT TWO SO TYPE TWO
7878	037752	012246			MOV	(R2)+,-(SP)	;OCTAL NUMBERS.
7879	037754	104402			TYPOC		
7880	037756	104401			TYPE		
7881	037760	040414			.WORD	SPACE	
7882	037762	011246			MOV	(R2),-(SP)	
7883	037764	104402			TYPOC		
7884	037766	000467			BR	ERT2	
7885							
7886	037770	122710	000003	9\$:	CMPB	#3,(R0)	;FORMAT THREE?
7887	037774	001020			BNE	10\$	
7888							
7889	037776	013102			MOV	@(R1)+,R2	;FORMAT THREE SO TYPE
7890	040000	012246			MOV	(R2)+,-(SP)	;FOUR OCTAL NUMBERS.
7891	040002	104402			TYPOC		
7892	040004	104401			TYPE		
7893	040006	040414			.WORD	SPACE	
7894	040010	012246			MOV	(R2)+,-(SP)	
7895	040012	104402			TYPOC		
7896	040014	104401			TYPE		
7897	040016	040414			.WORD	SPACE	
7898	040020	012246			MOV	(R2)+,-(SP)	
7899	040022	104402			TYPOC		
7900	040024	104401			TYPE		
7901	040026	040414			.WORD	SPACE	
7902	040030	011246			MOV	(R2),-(SP)	
7903	040032	104402			TYPOC		
7904	040034	000444			BR	ERT2	
7905							

7906	040036	122710	000004	10s:	CMPB	#4,(R0)	;FORMAT FOUR?
7907	040042	001004			BNE	11s	
7908							
7909	040044	013146			MOV	@(R1)+,-(SP)	;FORMAT FOUR SO TYPE
7910	040046	104403			TYPOS		;AN OCTAL NUMBER
7911	040050	016			.BYTE	16	;SUPPRESSING LEADING ZEROES.
7912	040051	000			.BYTE	0	
7913	040052	000435			BR	ERT2	
7914							
7915	040054	122710	000005	11s:	CMPB	#5,(R0)	;FORMAT FIVE?
7916	040060	001005			BNE	13s	
7917							
7918	040062	012137	040070		MOV	(R1)+,@#12s	;FORMAT FIVE SO TYPE AN
7919	040066	104401			TYPE		;ASCIZ STRING.
7920	040070	000000		12s:	.WORD	0	
7921	040072	000427			BR	ERT3	
7922							
7923	040074	122710	000011	13s:	CMPB	#11,(R0)	;FORMAT ELEVEN?
7924	040100	001005			BNE	15s	
7925							
7926	040102	013137	040110		MOV	@(R1)+,@#14s	;FORMAT ELEVEN SO PICK
7927	040106	104401			TYPE		;A POINTER TO AN ASCIZ
7928	040110	000000		14s:	.WORD	0	;STRING.
7929	040112	000417			BR	ERT3	
7930							
7931	040114	122710	000012	15s:	CMPB	#12,(R0)	;FORMAT TWELVE?
7932	040120	001011			BNE	17s	
7933							
7934	040122	013102			MOV	@(R1)+,R2	;FORMAT TWELVE SO TYPE
7935	040124	012703	000006		MOV	#6,R3	;TYPE SIX OCTAL NUMBERS
7936	040130	012246		16s:	MOV	(R2)+,-(SP)	
7937	040132	104402			TYPOC		
7938	040134	104401			TYPE		
7939	040136	040414			.WORD	SPACE	
7940	040140	077305			SQB	R3,16s	
7941	040142	000401			BR	ERT2	
7942							
7943	040144	000000		17s:	HALT		;UNDEFINED FORMAT FOR DATA?????
7944							
7945	040146	104401		ERT2:	TYPE		;PRINT A TAB AFTER TYPING
7946	040150	040412			.WORD	\$TAB	;AN DATA TABLE ENTRY
7947							;OF ALL FORMATS EXCEPT
7948							;ASCIZ, FORMATS 5 OR 11
7949							
7950	040152	005200		ERT3:	INC	R0	;POINT TO THE NEXT FORMAT
7951	040154	005711			TST	(R1)	;END OF DATA TABLE.
7952	040156	001401			BEQ	ERT4	
7953	040160	000663			BR	ERT1	
7954							
7955	040162	104401		ERT4:	TYPE		;DONE.
7956	040164	001313			.WORD	\$CRLF	
7957	040166	012603			MOV	(SP)+,R3	;RESTORE R1,R2 AND R3
7958	040170	012602			MOV	(SP)+,R2	
7959	040172	012601			MOV	(SP)+,R1	
7960	040174	012600		ERT5:	MOV	(SP)+,R0	;RESTORE R0.
7961	040176	000207			RTS	PC	;AND RETURN.



7962  
 7963  
 7964  
 7965  
 7966  
 7967  
 7968  
 7969  
 7970  
 7971  
 7972  
 7973  
 7974  
 7975  
 7976  
 7977  
 7978  
 7979  
 7980  
 7981  
 7982  
 7983  
 7984  
 7985  
 7986  
 7987  
 7988  
 7989  
 7990  
 7991  
 7992  
 7993  
 7994  
 7995  
 7996  
 7997  
 7998  
 7999  
 8000  
 8001  
 8002  
 8003  
 8004  
 8005  
 8006  
 8007  
 8008  
 8009  
 8010  
 8011  
 8012  
 8013  
 8014  
 8015  
 8016  
 8017

040200 011637 001236  
 040204 022626  
 040206 170200  
 040210 010037 001240  
 040214 170300  
 040216 010037 001242  
 040222 104211  
 040224 104412  
 040226 000137 034556  
 040232 011637 001236  
 040236 022626  
 040240 104212  
 040242 104412  
 040244 000137 034556  
 040250 011637 001236  
 040254 022626  
 040256 104213  
 040260 104412  
 040262 000137 034556

```

      .SBTTL FPP SPURIOUS TRAP TO 244 HANDLER
      ;*****
      ;*****
      ;*THIS ROUTINE HANDLES UNEXPECTED TRAPS TO THE FPP TRAP VECTOR AT 244.
      ;*THE LAST FPP INSTRUCTION EXECUTED AND ITS ADDRESS HAS BEEN RECORDED
      ;*THESE ALONG WITH THE FEC, FPS AND PC OF TRAP ARE REPORTED.
      ;*
      FPPSPUR: MOV      (SP),@#STMP2          ;SAVE PC OF TRAP.
                CMP      (SP)+,(SP)+        ;RESTORE SP.
                STEPS   R0                   ;GET FPS
                MOV      R0,@#STMP3
                STST    R0                   ;GET FEC
                MOV      R0,@#STMP4
1$:      ERROR   211
                RSETUP
                ;GO INITIALIZE THE FPS AND STACK; AND
                ;SEE IF THE USER HAS EXPRESSED
                ;THE DESIRE TO CHANGE THE SOFTWARE
                ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
                ;THE USER TYPED CONTROL G?).
                JMP      @#SEOP

      .SBTTL CPU SPURIOUS TRAP TO 4 HANDLER
      ;*****
      ;*****
      ;*THIS ROUTINE REPORTS UNEXPECTED CPU TRAPS TO VECTOR 4.
      ;*
      CPSPUR:  MOV      (SP),@#STMP2          ;SAVE PC OF TRAP.
                CMP      (SP)+,(SP)+
1$:      ERROR   212
                RSETUP
                ;GO INITIALIZE THE FPS AND STACK; AND
                ;SEE IF THE USER HAS EXPRESSED
                ;THE DESIRE TO CHANGE THE SOFTWARE
                ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
                ;THE USER TYPED CONTROL G?).
                JMP      @#SEOP

      .SBTTL CPU SPURIOUS TRAP TO 10 HANDLER
      ;*****
      ;*****
      ;*THIS ROUTINE REPORTS UNEXPECTED CPU TRAPS TO VECTOR 10.
      ;*
      CPTWO:   MOV      (SP),@#STMP2          ;SAVE PC OF TRAP.
                CMP      (SP)+,(SP)+
1$:      ERROR   213
                RSETUP
                ;GO INITIALIZE THE FPS AND STACK; AND
                ;SEE IF THE USER HAS EXPRESSED
                ;THE DESIRE TO CHANGE THE SOFTWARE
                ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
                ;THE USER TYPED CONTROL G?).
                JMP      @#SEOP
  
```

```

8018
8019
8020
8021
8022          .SBTTL  SET LOOP ON ERROR ADDRESS ROUTINE
8023          ;*****
8024          ;*****
8025          ;*
8026 040260 011637 001110 .LPER:  MOV      (SP),@#SLPERR
8027 040272 000002          RTI
8028
8029          .SBTTL  FLAG RESET AND CONSOLE TEST ROUTINE
8030          ;*****
8031          ;*****
8032          ;*THIS ROUTINE WILL BE CALLED AT THE END OF EACH TEST TO
8033          ;*RESET THE STACK, CLEAR THE FPS AND SEE IF THE USER HAS TYPED
8034          ;*CONTROL G ON THE TERMINAL. IF THE USER HAS TYPED CONTROL G AND
8035          ;*THERE IS NO PHYSICAL CONSOLE SWITCH REGISTER THEN THE CONTENTS
8036          ;*OF THE SOFTWARE SWITCH REGISTER WILL BE TYPED IN OCTAL ON THE
8037          ;*TELETYPE AND THE USER CAN MODIFY IT.
8038          ;*
8039 040274 023727 001140 177570 .RSET:  CMP      @#SWR,#177570          ;SEE IF THERE IS A PHYSICAL
8040          ;CONSOLE SWITCH REGISTER,
8041 040302 001001          BNE      1S          ;BRANCH IF NO.
8042 040304 104406          CKSWR          ;OTHERWISE TYPE THE CONTENTS
8043          ;OF THE PROGRAM VIRTUAL SWITCH REGISTER
8044          ;AND GIVE THE USER A CHANCE TO
8045          ;MODIFY IT.
8046 040306 012737 040200 000244 1S:  MOV      #FPSPUR,@#FPVECT
8047 040314 012737 040232 000004      MOV      #CPSPUR,@#ERRVECT
8048 040322 012737 040250 000010      MOV      #CPTWO,@#10
8049 040330 011600          MOV      (SP),R0          ;SAVE RETURN ADDRESS.
8050 040332 012706 001100          MOV      #STACK,SP          ;RESET THE STACK POINTER.
8051 040336 005004          CLR      R4          ;CLEAR THE FPS.
8052 040340 170104          LDFPS   R4
8053 040342 000110          JMP      (R0)          ;RETURN.
8054
8055          .NLIST  BEX
    
```

;SPECIAL MESSAGES:

```

040344 050200 053517 051105 POWERM: .ASCIZ  <CRLF>'POWER FAILURE. PROGRAM RESTARTING.'<CRLF>
040411      000          NULL:   .BYTE   0
040412 000011          STAB:  .ASCIZ  <TAB>
040414 020040      000          SPACE: .ASCIZ  ' '
040417      200 041520 047440 LFIEX1: .ASCIZ  <CRLF>'PC OF LAST FPP INSTRUCTION EXECUTED: '<TAB>
040467      200 040514 052123 LFIEX2: .ASCIZ  <CRLF>'LAST FPP INSTRUCTION EXECUTED: '<TAB>
040531      200 046106 040517 FPSMS:  .ASCIZ  <CRLF>'FLOATING POINT STATUS REGISTER: '
040575      200 042506 035103 FECMS:  .ASCIZ  <CRLF>'FEC: '
040606 044124 020105      000 STHE:   .ASCIZ  'THE '
040613      011 044440 051516 NOOP1:  .ASCIZ  <TAB>' INSTRUCTION FAILED.'<CRLF>
040642 044505 044124 051105 NOOP15: .ASCII  'EITHER A BAD CONSTANT WAS GENERATED OR '
040711      115 041511 047522          .ASCIZ  'MICROPROGRAM FLOW WENT '
040740 043200 047522 020115 NOOP2:  .ASCIZ  <CRLF>'FROM STATE '
040755      124 020117 052123 NOOP3:  .ASCIZ  'TO STATE '
040767      200 047111 052123 NOOP4:  .ASCIZ  <CRLF>'INSTEAD OF '
    
```

```

041004 052200 042510 042522 NOOP5: .ASCIZ <CRLF>'THEREBY EXECUTING A '
041032 020011 047111 052123 NOOP6: .ASCIZ <TAB>' INSTEAD OF A '
041052 020011 047111 052123 NOOP7: .ASCIZ <TAB>' INSTRUCTION.'<CRLF>
041072 020040 042524 052123 NOOP10: .ASCII ' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'<TAB>
041133 107 052117 043040 .ASCIZ 'GOT FPS. EXPECTED FPS.'<CRLF>
041163 101 041040 042101 NOOP11: .ASCIZ 'A BAD CONSTANT MAY HAVE BEEN USED.'<CRLF>
041227 011 042114 050106 LFPS1: .ASCIZ <TAB>'LDFPS'<TAB>'REG'
041242 046011 042104 024011 LD1: .ASCIZ <TAB>'LDD'<TAB>'(REG),A'<TAB>'//FSRC#0//'
041272 046011 042104 040411 LD2: .ASCIZ <TAB>'LDD'<TAB>'A,A'
041303 011 052123 050106 STFS1: .ASCIZ <TAB>'STFPS'<TAB>'REG'
041316 051411 042124 040411 ST1: .ASCIZ <TAB>'STD'<TAB>'A,(REG)'
041333 011 052123 004504 ST2: .ASCIZ <TAB>'STD'<TAB>'A,A'
041344 041411 041506 000103 CFCC1: .ASCIZ <TAB>'CFCC'
041352 051411 052105 000106 SETF1: .ASCIZ <TAB>'SETF'
041360 051411 052105 000104 SETD1: .ASCIZ <TAB>'SETD'
041366 051411 052105 000111 SETI1: .ASCIZ <TAB>'SETI'
041374 051411 052105 000114 SETL1: .ASCIZ <TAB>'SETL'
041402 044411 046114 043505 ILL1: .ASCIZ <TAB>'ILLEGAL FPP INSTRUCTION'
041433 011 052123 052123 STST1: .ASCIZ <TAB>'STST'<TAB>'REG'
041445 011 046111 042514 ILL2: .ASCIZ <TAB>'ILLEGAL FPP INSTRUCTION (FID=1)'
041506 020040 042524 052123 ILLMS: .ASCIZ ' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF TRAP.'<TAB>'FPS.'<CRLF>
041554 054105 042520 052103 MS1: .ASCIZ 'EXPECTED '
041566 047507 020124 000 MS2: .ASCIZ 'GOT '
041573 103 047117 042524 MS3: .ASCIZ 'CONTENTS OF LOCATIONS '
041622 052040 051110 052517 MS4: .ASCIZ ' THROUGH '
041634 040506 046111 051125 MS5: .ASCIZ 'FAILURE IN THE MICROPROGRAM FLOW.'
041676 047503 052116 047522 MS6: .ASCIZ 'CONTROL WENT '
041714 051106 046517 051440 MS7: .ASCIZ 'FROM STATE '
041730 052040 020117 052123 MS10: .ASCIZ ' TO STATE '
041743 102 052125 051440 MS11: .ASCIZ 'BUT SHOULD HAVE GONE'
041770 047503 052116 047522 MS12: .ASCIZ 'CONTROL FLOW SHOULD HAVE GONE'
042026 052502 020124 044504 MS13: .ASCIZ 'BUT DID NOT.'
042043 040 052040 051505 MS14: .ASCII ' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'<TAB>
042104 047507 020124 041520 .ASCIZ 'GOT PC.'<TAB>'EXPECTED PC.'
042131 111 051516 051124 MS15: .ASCIZ 'INSTRUCTION TESTED: '
042156 047440 020122 000 MS16: .ASCIZ ' OR '
042163 124 051505 044524 MS17: .ASCIZ 'TESTING ACCUMULATOR '
042210 042532 047522 000040 MNUM0: .ASCIZ 'ZERO '
042216 047117 020105 000 MNUM1: .ASCIZ 'ONE '
042223 124 047527 000040 MNUM2: .ASCIZ 'TWO '
042230 044124 042522 020105 MNUM3: .ASCIZ 'THREE '
042237 106 052517 020122 MNUM4: .ASCIZ 'FOUR '
042245 106 053111 020105 MNUM5: .ASCIZ 'FIVE '
042253 040 052040 051505 MS20: .ASCIZ ' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
042314 040504 040524 024040 MS21: .ASCIZ 'DATA (FLOATING POINT NUMBER): '
042353 114 043517 041511 MS22: .ASCIZ 'LOGICAL AND OF FAILING '
042403 114 043517 041511 MS23: .ASCIZ 'LOGICAL OR OF FAILING '
042432 020040 042524 052123 MS24: .ASCII ' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERRORS.'<TAB>
042474 052516 041115 051105 .ASCIZ 'NUMBER OF ERRORS(OCTAL),'
042525 105 050130 041505 MS25: .ASCIZ 'EXPECTED DATA IN '
042547 107 052117 042040 MS26: .ASCIZ 'GOT DATA IN '
042564 040600 030103 020075 MS27: .ASCIZ <CRLF>'AC0= '
042573 200 041501 036461 MS30: .ASCIZ <CRLF>'AC1= '
042602 040600 031103 020075 MS31: .ASCIZ <CRLF>'AC2= '
042611 200 041501 036463 MS32: .ASCIZ <CRLF>'AC3= '
042620 040600 032103 020075 MS33: .ASCIZ <CRLF>'AC4= '

```

042627	200	041501	036465	MS34:	.ASCIZ	<CRLF>'AC5= '
042636	042523	020124	000	MS35:	.ASCIZ	'SET '
042643	103	042514	051101	MS36:	.ASCIZ	'CLEAR '
042652	047514	042101	042105	MS37:	.ASCIZ	'LOADED DATA: '
042670	042522	042101	042040	MS40:	.ASCIZ	'READ DATA: '
042704	054105	042520	052103	MS415:	.ASCIZ	'EXPECTED DATA: '
042724	040504	040524	044440	MS41:	.ASCIZ	'DATA IN (R0) FSRC: '
042750	040504	040524	044440	MS42:	.ASCIZ	'DATA IN AC0: '
042766	047507	020124	042522	MS43:	.ASCIZ	'GOT RESULT: '
043003	105	050130	041505	MS44:	.ASCIZ	'EXPECTED RESULT: '

;ERROR MESSAGES:

043025	114	043104	051520	EM1:	.ASCIZ	'LDFPS AND STFPS TEST FAILED.'
043062	042114	050106	020123	EM2:	.ASCIZ	'LDFPS AND STFPS TEST ERROR SUMMARY.'
043126	043103	041503	052040	EM3:	.ASCIZ	'CFCC TRANSFERED BAD DATA TO THE PSW.'
043173	103	041506	020103	EM4:	.ASCIZ	'CFCC MODIFIED THE FPS REGISTER.'
043233	125	042516	050130	EM5:	.ASCIZ	'UNEXPECTED FPP TRAP TO 244.'
043267	125	042516	050130	EM6:	.ASCIZ	'UNEXPECTED CPU TRAP TO 4.'
043321	125	042516	050130	EM7:	.ASCIZ	'UNEXPECTED CPU TRAP TO 10.'
	043233			EM10=EM5		
043354	047125	041101	042514	EM11:	.ASCIZ	'UNABLE TO DECODE FPP INSTRUCTION, TRAPPED TO 10.'
	000000			EM12=0		
	000000			EM13=0		
043435	114	043104	051520	EM14:	.ASCII	'LDFPS R0 FAILED IN THE FSRC FLOWS.'
043477	040	051124	050101		.ASCII	' TRAPPED TO 4.'
043515	200	044504	020104		.ASCIZ	<CRLF>'DID NOT GO FROM STATE 400 TO 670.'
043560	052123	050106	020123	EM15:	.ASCII	'STFPS R1 FAILED IN THE FDST FLOWS.'
043622	052040	040522	050120		.ASCII	' TRAPPED TO 4.'
043640	042200	042111	047040		.ASCIZ	<CRLF>'DID NOT GO FROM STATE 634 TO 710.'
043703	101	020116	046111	EM16:	.ASCIZ	'AN ILLEGAL FPP INSTRUCTION DID NOT TRAP.'
043754	047101	044440	046114	EM17:	.ASCII	'AN ILLEGAL FPP INSTRUCTION'
044006	052200	040522	050120		.ASCII	<CRLF>'TRAPPED TO 244, BUT FAILED TO SET '
044051	124	042510	043040		.ASCII	'THE FPS CORRECTLY.'<CRLF>'EITHER A BAD CONSTANT '
044122	040527	020123	042507		.ASCIZ	'WAS GENERATED OR THE ALU LOGICAL OR FUNCTION FAILED.'
044207	101	020116	046111	EM20:	.ASCII	'AN ILLEGAL FPP INSTRUCTION'
044241	040	051124	050101		.ASCII	' TRAPPED TO 244, BUT A SUBSEQUENT '
044303	040	052123	052123		.ASCII	' STST'<CRLF>
044311	106	044501	042514		.ASCIZ	'FAILED TO PICK UP THE CORRECT FEC CODE = 2.'
044365	123	051524	020124	EM21:	.ASCII	'STST R4 FAILED IN THE DESTINATION FLOWS.'
044435	040	051124	050101		.ASCII	' TRAPPED TO 4.'<CRLF>
044454	044504	020104	047516		.ASCIZ	'DID NOT GO FROM STATE 636 TO 710.'
044516	047101	044440	046114	EM22:	.ASCII	'AN ILLEGAL FPP INSTRUCTION,'
044551	127	052111	020110		.ASCIZ	'WITH INTERRUPTS DISABLED,'
	044516			EM23=EM22		
	044516			EM24=EM22		
044603	123	052517	041522	EM25:	.ASCII	'SOURCE LOCATIONS MODIFIED BY, LDD.'
044645	200	020101	040504		.ASCIZ	<CRLF>'A DATO WAS PERFORMED INSTEAD OF A DATI.'
044716	042114	020104	051050	EM26:	.ASCII	'LDD (R0),AC0 FAILED.'<CRLF>
044743	122	020060	040527		.ASCIZ	'R0 WAS MODIFIED.'
	044716			EM27=EM26		
044764	044124	020105	041520	EM30:	.ASCII	'THE PC WAS BAD AFTER '
045012	047101	043040	050120		.ASCIZ	'AN FPP INSTRUCTION.'
045036	052123	020104	041501	EM31:	.ASCII	'STD AC0,(R0) FAILED.'<CRLF>
045063	122	020060	040527		.ASCIZ	'R0 WAS MODIFIED.'

Address	Offset	Op1	Op2	Op3	Op4	Label	Instruction
	045036					EM32=EM31	
045104	052123	020104	041501			EM33:	.ASCII 'STD AC0,(R0) FAILED.' <crlf&gt;< td=""> </crlf&gt;<>
045131	117	052125	052520				.ASCIZ 'OUTPUT BAD.'
045145	123	042124	040440			EM34:	.ASCII 'STD AC0,(R0) FAILED IN THE FDST FLOWS.'
045213	200	044124	020105				.ASCIZ <CRLF>'THE (BUT GR7) FORK FAILED.'
045247	114	042104	024040			EM35:	.ASCII 'LDD (R0),AC0 FAILED IN THE FSRC FLOWS.'
045315	200	044124	020105				.ASCIZ <CRLF>'THE (BUT GR7) FORK FAILED.'
045351	123	042124	040440			EM36:	.ASCII 'STD AC0,(R0) FAILED IN THE FDST FLOWS.'
045417	200	044124	020105				.ASCIZ <CRLF>'THE (BUT FD) FORK FAILED.'
045452	042114	020104	051050			EM37:	.ASCII 'LDD (R0),AC0 FAILED IN THE FSRC FLOWS.'
045520	052200	042510	024040				.ASCIZ <CRLF>'THE (BUT FD) FORK FAILED.'
045553	114	042104	024040			EM40:	.ASCII 'LDD (R0),AC0 OR THE STD AC0,(R0) FAILED.'
045623	200	040502	020104				.ASCIZ <CRLF>'BAD DATA WAS DETECTED AFTER A SEQUENCE OF THE TWO INSTRUCTIONS.'
045724	050106	020123	040502			EM41:	.ASCIZ 'FPS BAD AFTER EXECUTION OF: '
(0) 045761						EM42:	
(1) 045761	114	042104	024040				.ASCII /LDD (R0),AC0 FAILED IN THE FSRC FLOWS./<CRLF>
(1) 046030	044124	020105	041050				.ASCIZ /THE (BUT FSRC) FORK FAILED. TRAPPED TO 4./
(0) 046102						EM43:	
(1) 046102	052123	020104	041501				.ASCII /STD AC0,(R0) FAILED IN THE FDST FLOWS./<CRLF>
(1) 046151	124	042510	024040				.ASCIZ /THE (BUT FDST) FORK FAILED. TRAPPED TO 4./
046223	106	050120	040440			EM44:	.ASCIZ 'FPP ACCUMULATORS DATA TEST FAILED.'
	046223					EM45=EM44	
(0) 046266	050106	020120	041501			EM46:	.ASCIZ 'FPP ACCUMULATORS DUAL ADDRESSING TEST FAILED.'
(0) 046344						EM47:	
(1) 046344	042114	040440	030503				.ASCII /LD AC1,AC0 FAILED IN THE FSRC FLOWS./
(1) 046410	044124	020105	041050				.ASCIZ /THE (BUT FSRC) FORK FAILED. TRAPPED TO 4./
046462	042114	040440	030503			EM50:	.ASCII 'LD AC1,AC0 FAILED IN THE FSRC FLOWS.'
046526	044124	020105	041050				.ASCIZ 'THE (BUT FD) FORK FAILED.'
046560	042114	040440	030503			EM51:	.ASCIZ 'LD AC1,AC0 TRANSFERRED BAD DATA.'
(0) 046621						EM52:	
(1) 046621	114	042104	024040				.ASCII /LDD (R0)+,AC0 FAILED IN THE FSRC FLOWS./
(1) 046670	044124	020105	041050				.ASCIZ /THE (BUT FSRC) FORK FAILED. TRAPPED TO 4./
(0) 046742						EM53:	
(1) 046742	042114	020104	051050				.ASCII /LDD (R0)+,AC0 FAILED IN THE FSRC FLOWS./
(1) 047011	200	030122	053440				.ASCII <CRLF>'R0 WAS BAD.' <crlf&gt;< td=""> </crlf&gt;<>
(1) 047026	044505	044124	051105				.ASCII 'EITHER A BAD CONSTANT WAS GENERATED OR' <crlf&gt;< td=""> </crlf&gt;<>
(1) 047075	104	042111	047040				.ASCIZ \DID NOT GO FROM STATE 627 TO 322.\
(0) 047137						EM54:	
(1) 047137	114	042104	024040				.ASCIZ /LDD (R0)+,AC0 TRANSFERRED BAD DATA./
(0) 047203						EM55:	
(1) 047203	114	042104	026440				.ASCII /LDD -(R0),AC0 FAILED IN THE FSRC FLOWS./
(1) 047252	044124	020105	041050				.ASCIZ /THE (BUT FSRC) FORK FAILED. TRAPPED TO 4./
(0) 047324						EM56:	
(1) 047324	042114	020104	024055				.ASCII /LDD -(R0),AC0 FAILED IN THE FSRC FLOWS./
(1) 047373	200	030122	053440				.ASCII <CRLF>'R0 WAS BAD.' <crlf&gt;< td=""> </crlf&gt;<>
(1) 047410	044505	044124	051105				.ASCII 'EITHER A BAD CONSTANT WAS GENERATED OR' <crlf&gt;< td=""> </crlf&gt;<>
(1) 047457	104	042111	047040				.ASCIZ \DID NOT GO FROM STATE 627 TO 324.\
(0) 047521						EM57:	
(1) 047521	114	042104	026440				.ASCIZ /LDD -(R0),AC0 TRANSFERRED BAD DATA./
(0) 047565						EM60:	
(1) 047565	114	043104	024040				.ASCII /LDF (R0)+,AC0 FAILED IN THE FSRC FLOWS./
(1) 047634	051200	020060	040527				.ASCII <CRLF>'R0 WAS BAD.' <crlf&gt;< td=""> </crlf&gt;<>
(1) 047651	105	052111	042510				.ASCII 'EITHER A BAD CONSTANT WAS GENERATED OR' <crlf&gt;< td=""> </crlf&gt;<>
(1) 047720	044504	020104	047516				.ASCIZ \DID NOT GO FROM STATE 627 TO 322.\
(0) 047762						EM61:	
(1) 047762	042114	020106	051050				.ASCIZ /LDF (R0)+,AC0 TRANSFERRED BAD DATA./

	050026	042114	020106	051050	EM62:	.ASCII	'LDF (R0)+,AC0 FAILED IN THE FSRC FLOWS.'
	050075	200	044124	020105		.ASCII	<CRLF>'THE (BUT FD) FORK FAILED.'<CRLF>
	050130	042527	052116	043040		.ASCII	'WENT FROM STATE 441 TO 077.'<CRLF>
	050164	047111	052123	040505		.ASCIZ	'INSTEAD OF FROM 441 TO 076.'
	050220	042114	020104	047043	EM63:	.ASCII	'LDD #NUM,AC0 FAILED IN THE FSRC FLOWS.'
	050266	052200	042510	024040		.ASCII	<CRLF>'THE (BUT GR7) FORK FAILED.'<CRLF>
	050322	042527	052116	043040		.ASCII	'WENT FROM STATE 207 TO 174.'<CRLF>
	050356	047111	052123	040505		.ASCIZ	'INSTEAD OF FROM 207 TO 176.'
	050412	042114	020104	047043	EM64:	.ASCII	'LDD #NUM,AC0 FAILED IN THE FSRC FLOWS.'
	050460	040600	041040	042101		.ASCIZ	<CRLF>'A BAD CONSTANT WAS USED WHEN THE PC WAS INCREMENTED.'
		050412			EM65=EM64		
(0)	050546				EM66:		
(1)	050546	042114	020104	047043		.ASCIZ	/LDD #NUM,AC0 TRANSFERRED BAD DATA./
(0)	050611				EM67:		
(1)	050611	114	042104	040040		.ASCII	'LDD @(R0)+,AC0 FAILED IN THE FSRC FLOWS.'
(1)	050661	200	044124	020105		.ASCII	<CRLF>'THE (BUT FSRC) FORK FAILED, TRAPPED TO 4.'
(1)	050733	200	042527	052116		.ASCII	<CRLF>'WENT FROM STATE 627 TO EITHER 326 OR 326,\
(1)	051005	200	047111	052123		.ASCIZ	<CRLF>'INSTEAD OF FROM 627 TO 323.\
(0)	051042				EM70:		
(1)	051042	042114	020104	024100		.ASCII	'LDD @(R0)+,AC0 FAILED IN THE FSRC FLOWS.'
(1)	051112	052200	042510	024040		.ASCIZ	<CRLF>'THE (BUT FSRC) FORK FAILED, TRAPPED TO 4.'
(0)	051165				EM71:		
(1)	051165	114	042104	040040		.ASCII	'LDD @(R0)+,AC0 FAILED IN THE FSRC FLOWS.'
(1)	051235	124	042510	024040		.ASCIZ	'THE (BUT FD) FORK FAILED.'
(0)	051267				EM72:		
(1)	051267	114	042104	040040		.ASCII	'LDD @(R0)+,AC0'<CRLF>
(1)	051306	040506	046111	042105		.ASCIZ	'FAILED TO INCREMENT R0 BY 2.'
(0)	051343				EM73:		
(1)	051343	114	042104	040040		.ASCIZ	'LDD @(R0)+,AC0 LOADED BAD DATA.'
(0)	051403				EM74:		
(1)	051403	114	042104	040040		.ASCII	'LDD @-(R0),AC0 FAILED IN THE FSRC FLOWS.'
(1)	051453	200	044124	020105		.ASCII	<CRLF>'THE (BUT FSRC) FORK FAILED, TRAPPED TO 4.'
(1)	051525	200	042527	052116		.ASCII	<CRLF>'WENT FROM STATE 627 TO EITHER 326 OR 326,\
(1)	051577	200	047111	052123		.ASCIZ	<CRLF>'INSTEAD OF FROM 627 TO 325.\
(0)	051634				EM75:		
(1)	051634	042114	020104	026500		.ASCII	'LDD @-(R0),AC0 FAILED IN THE FSRC FLOWS.'
(1)	051704	052200	042510	024040		.ASCIZ	<CRLF>'THE (BUT FSRC) FORK FAILED, TRAPPED TO 4.'
(0)	051757				EM76:		
(1)	051757	114	042104	040040		.ASCII	'LDD @-(R0),AC0 FAILED IN THE FSRC FLOWS.'
(1)	052027	124	042510	024040		.ASCIZ	'THE (BUT FD) FORK FAILED.'
(0)	052061				EM77:		
(1)	052061	114	042104	040040		.ASCII	'LDD @-(R0),AC0'<CRLF>
(1)	052100	040506	046111	042105		.ASCIZ	'FAILED TO DECREMENT R0 BY 2.'
(0)	052135				EM100:		
(1)	052135	114	042104	040040		.ASCIZ	'LDD @-(R0),AC0 LOADED BAD DATA.'
(0)	052175				EM101:		
(1)	052175	114	042104	047040		.ASCII	'LDD NUM(R0),AC0 FAILED IN THE FSRC FLOWS.'
(1)	052246	052200	042510	024040		.ASCIZ	<CRLF>'THE (BUT FSRC) FORK FAILED, TRAPPED TO 4.'
(0)	052321				EM102:		
(1)	052321	114	042104	047040		.ASCII	'LDD NUM(R0),AC0'<CRLF>
(1)	052341	106	044501	042514		.ASCIZ	'FAILED TO AFFECT R0 BY 2.'
(0)	052373				EM103:		
(1)	052373	114	042104	047040		.ASCII	'LDD NUM(R0),AC0 FAILED IN THE FSRC FLOWS.'
(1)	052444	044124	020105	041050		.ASCIZ	'THE (BUT FD) FORK FAILED.'
(0)	052476				EM104:		
(1)	052476	042114	020104	052516		.ASCIZ	'LDD NUM(R0),AC0 LOADED BAD DATA.'

(0)	052537				EM105:	
(1)	052537	114	042104	040040	.ASCII	'LDD @NUM(R0),AC0 FAILED IN THE FSRC FLOWS.'
(1)	052611	200	044124	020105	.ASCIZ	<CRLF>'THE (BUT FSRC) FORK FAILED, TRAPPED TO 4.'
(0)	052664				EM106:	
(1)	052664	042114	020104	047100	.ASCII	'LDD @NUM(R0),AC0<CRLF>
(1)	052705	106	044501	042514	.ASCIZ	'FAILED TO AFFECT R0 BY 2.'
(0)	052737				EM107:	
(1)	052737	114	042104	040040	.ASCII	'LDD @NUM(R0),AC0 FAILED IN THE FSRC FLOWS.'
(1)	053011	124	042510	024040	.ASCIZ	'THE (BUT FD) FORK FAILED.'
(0)	053043				EM110:	
(1)	053043	114	042104	040040	.ASCIZ	'LDD @NUM(R0),AC0 LOADED BAD DATA.'
(0)	053105				EM111:	
(1)	053105	114	042104	040440	.ASCII	/'LDD AC7,AC0 FAILED TO TRAP TO 244./
(1)	053147	200	041501	020067	.ASCIZ	<CRLF>/'AC7 IS AN ILLEGAL ACCUMULATOR./
(0)	053207	053105			EM112=EM111	
(1)	053207	114	042104	040440	.ASCII	/'LDD AC6,AC0 FAILED TO TRAP TO 244./
(1)	053251	200	041501	020066	.ASCIZ	<CRLF>/'AC6 IS AN ILLEGAL ACCUMULATOR./
(0)	053311	053207			EM114=EM113	
(1)	053311	125	042523	047440	.ASCII	'USE OF AN ILLEGAL ACCUMULATOR WITH FSRC MODE ZERO.'
(1)	053373	200	051124	050101	.ASCIZ	<CRLF>'TRAPPED BUT FAILED TO SET FPS CORRECTLY.'
(0)	053445	053105			EM120:	
(1)	053445	125	042523	047440	.ASCII	'USE OF AN ILLEGAL ACCUMULATOR WITH FSRC MODE ZERO.'
(1)	053527	200	051124	050101	.ASCIZ	<CRLF>'TRAPPED BUT FAILED TO SET FEC CORRECTLY.'
(0)	053601	053207			EM121:	
(1)	053601	123	020124	041501	.ASCII	'ST AC0,AC1 FAILED IN THE FDST FLOWS.'
(0)	053645	053105			EM122:	
(1)	053645	200	044124	020105	.ASCIZ	<CRLF>'THE (BUT FDST) FORK FAILED, TRAPPED TO 4.'
(0)	053720	053207			EM123:	
(1)	053720	052123	040440	030103	.ASCII	'ST AC0,AC1 FAILED IN THE FDST FLOWS.'
(0)	053764	053105			EM124:	
(1)	053764	052200	042510	024040	.ASCIZ	<CRLF>'THE (BUT FD) FORK FAILED.'
(0)	054017	053207			EM125:	
(1)	054017	123	020124	041501	.ASCIZ	'ST AC0,AC1 TRANSFERRED BAD DATA.'
(0)	054060				EM126:	
(1)	054060	050106	020123	040502	.ASCII	'FPS BAD AFTER LDD (R0),AC0.'
(1)	054113	200	044124	020105	.ASCIZ	<CRLF>'THE (BUT EZBT Y8) FORK FAILED.\
(0)	054153	053105			EM127:	
(1)	054153	106	051520	041040	.ASCII	'FPS BAD AFTER LDD (R0),AC0.'
(1)	054206	052200	042510	024040	.ASCIZ	<CRLF>'THE (BUT ENBT) FORK FAILED.\
(0)	054243	053207			EM128:	
(1)	054243	114	042104	024040	.ASCII	'LDD (R0),AC0 TRAPPED TO 244.'
(0)	054277	053105			EM129:	
(1)	054277	040	051506	041522	.ASCII	'FSRC= -0 AND FIUV= 0.<CRLF>
(0)	054326	053207			EM130:	
(1)	054326	044124	020105	041050	.ASCII	'THE (BUT FIUV) FORK FAILED.'
(0)	054361	053105			EM131:	
(1)	054361	200	042527	052116	.ASCII	<CRLF>'WENT FROM STATE 256 TO 354.'
(0)	054415	053207			EM132:	
(1)	054415	200	047111	052123	.ASCIZ	<CRLF>'INSTEAD OF FROM 256 TO 254.'
(0)	054452	053105			EM133:	
(1)	054452	042114	020104	051050	.ASCII	'LDD (R0),AC0 FAILED TO TRAP TO 244.'
(0)	054515	053207			EM134:	
(1)	054515	040	051506	041522	.ASCII	'FSRC= -0, FIUV= 1.'
(0)	054540	053105			EM135:	
(1)	054540	052200	042510	024040	.ASCII	<CRLF>'THE (BUT FIUV) FORK FAILED.<CRLF>
(0)	054574	053207			EM136:	
(1)	054574	042527	052116	043040	.ASCII	'WENT FROM STATE 256 TO 254.'
(0)	054627	053105			EM137:	
(1)	054627	200	047111	052123	.ASCIZ	<CRLF>'INSTEAD OF FROM 256 THE 354.'
(0)	054665	053207			EM138:	
(1)	054665	114	042104	024040	.ASCII	'LDD (R0),AC0 TRAPPED TO 244.'
(0)	054721	053105			EM139:	
(1)	054721	106	051123	036503	.ASCII	'FSRC= -0, FIUV= 1.<CRLF>
(0)	054744	053207			EM140:	
(1)	054744	052502	020124	042506	.ASCIZ	'BUT FEC WAS BAD.'
(0)	054765				EM141:	
(1)	054765	114	041504	042106	.ASCIZ	/'LDCFD (R0),AC0 LOADED BAD DATA./
(0)	055025				EM142:	
(1)	055025	114	041504	043104	.ASCIZ	/'LDCDF (R0),AC0 LOADED BAD DATA./
(0)	055065				EM143:	

(1)	055065	101	042104	020104	.ASCIZ	/ADDD (R0),AC0 WITH (R0)=AC0=0 /
(0)	055124				EM134:	
(1)	055124	042101	043104	024040	.ASCIZ	/ADDF (R0),AC0 WITH (R0)=AC0=0 /
(0)	055163				EM135:	
(1)	055163	123	041125	020104	.ASCIZ	/SUBD (R0),AC0 WITH (R0)=AC0=0 /
(0)	055222				EM136:	
(1)	055222	052523	043102	024040	.ASCIZ	/SUBF (R0),AC0 WITH (R0)=AC0=0 /
	055265				EM137=EM133	
	055124				EM140=EM134	
	055163				EM141=EM135	
	055222				EM142=EM136	
(0)	055261				EM143:	
(1)	055261	101	042104	020104	.ASCIZ	/ADDD (R0),AC0 WITH (R0)=0 /
(0)	055314				EM144:	
(1)	055314	052523	042102	024040	.ASCIZ	/SUBD (R0),AC0 WITH (R0)=0 /
	055261				EM145=EM143	
	055314				EM146=EM144	
(0)	055347				EM147:	
(1)	055347	123	041125	020104	.ASCIZ	/SUBD (R0),AC0 WITH AC0=0 /
	055347				EM150=EM147	
	055347				EM151=EM147	
(0)	055401				EM152:	
(1)	055401	101	042104	020104	.ASCIZ	/ADDD (R0),AC0 WITH AC0=0 /
	055401				EM153=EM152	
(0)	055433				EM154:	
(1)	055433	101	020116	053117	.ASCII	*AN OVERFLOW ERROR OCCURRED ON ADD* <crlf></crlf>
(1)	055475	103	052501	044523	.ASCII	*CAUSING A TRAP TO 244.*
(1)	055523	200	041050	052125	.ASCII	<CRLF>*(BUT EZBT Y9 Y8) FORK IN STATE 420 OF OVER\UNDER FAILED,*
(1)	055614	051600	047510	046125	.ASCIZ	<CRLF>*SHOULD HAVE GONE FROM STATE 420 TO 131.*
(0)	055665				EM155:	
(1)	055665	101	020116	047125	.ASCII	*AN UNDERFLOW ERROR OCCURRED ON ADD* <crlf></crlf>
(1)	055730	040503	051525	047111	.ASCII	*CAUSING A TRAP TO 244.*
(1)	055756	024200	052502	020124	.ASCII	<CRLF>*(BUT EZBT Y9 Y8) FORK IN STATE 420 OF OVER\UNDER FAILED,*
(1)	056047	200	044123	052517	.ASCIZ	<CRLF>*SHOULD HAVE GONE FROM STATE 420 TO 131.*
(0)	056120				EM156:	
(1)	056120	042101	042104	024040	.ASCII	/ADDD (R0),AC0 FAILED IN THE ROUND\TRUNK FLOWS./
(1)	056176	052200	042510	024040	.ASCII	<CRLF>*THE (BUT FD) FORK FAILED, WENT*
(1)	056235	106	047522	020115	.ASCII	\FROM STATE 665 TO 113.\<CRLF>
(1)	056264	047111	052123	040505	.ASCIZ	\INSTEAD OF FROM 665 TO 313.\<CRLF>\WITH FT SET.\
(0)	056335				EM157:	
(1)	056335	101	042104	020104	.ASCII	/ADDD (R0),AC0 FAILED IN THE ROUND\TRUNK FLOWS./
(1)	056413	200	044124	020105	.ASCII	<CRLF>*THE (BUT FD) FORK FAILED, WENT*
(1)	056452	051106	046517	051440	.ASCII	\FROM STATE 665 TO 313.\<CRLF>
(1)	056501	111	051516	042524	.ASCIZ	\INSTEAD OF FROM 665 TO 113.\<CRLF>\WITH FT CLEAR.\
(0)	056554				EM160:	
(1)	056554	042101	042104	024040	.ASCII	/ADDD (R0),AC0 FAILED IN THE ROUND\TRUNK FLOWS./<CRLF>
(1)	056633	124	042510	043040	.ASCII	*THE FLOATING CONSTANT WAS USED INSTEAD OF THE DOUBLE CONSTANT* <crlf></crlf>
(1)	056731	111	020116	044124	.ASCIZ	*IN THE ROUND ALGORITHM.*
(0)	056761				EM161:	
(1)	056761	101	042104	020106	.ASCII	/ADDF (R0),AC0 FAILED IN THE ROUND\TRUNK FLOWS./<CRLF>
(1)	057040	044124	020105	047504	.ASCII	*THE DOUBLE CONSTANT WAS USED INSTEAD OF THE FLOATING CONSTANT* <crlf></crlf>
(1)	057136	047111	052040	042510	.ASCIZ	*IN THE ROUND ALGORITHM.*
(0)	057166				EM162:	
(1)	057166	042101	042104	024040	.ASCIZ	/ADDD (R0),AC0 PRODUCED A BAD RESULT./
(0)	057233				EM163:	
(1)	057233	101	042104	020106	.ASCIZ	/ADDF (R0),AC0 PRODUCED A BAD RESULT./



(0)	057300				EM164:	
(1)	057300	044124	020105	050106	.ASCIZ	\THE FPS WAS BAD AFTER ADDD (R0),AC0.\
(0)	057345				EM165:	
(1)	057345	124	042510	043040	.ASCIZ	\THE FPS WAS BAD AFTER ADDE (R0),AC0.\
(0)	057412				EM166:	
(1)	057412	042101	042104	024040	.ASCII	/ADDD (R0),AC0 PRODUCED A BAD RESULT,/<CRLF>
(1)	057457	120	047522	040502	.ASCIZ	'PROBABLE ERROR IN THE ALIGN FLOWS.'
(0)	057522				EM167:	
(1)	057522	042101	042104	024040	.ASCII	/ADDD (R0),AC0 FAILED IN THE ALIGN FLOWS,/<CRLF>
(1)	057573	126	047514	020127	.ASCII	\FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 111, TO 014.\
(1)	057663	200	020101	040502	.ASCII	<CRLF>\A BAD CONSTANT (NOT 57 DEC) \
(1)	057720	040527	020123	051525	.ASCIZ	'WAS USED IN THE ALIGN ALGORITHM.'
(0)	057761				EM170:	
(1)	057761	101	042104	020106	.ASCII	/ADDE (R0),AC0 PRODUCED A BAD RESULT,/<CRLF>
(1)	060026	051120	041117	041101	.ASCIZ	'PROBABLE ERROR IN THE ALIGN FLOWS.'
(0)	060071				EM171:	
(1)	060071	101	042104	020106	.ASCII	/ADDE (R0),AC0 FAILED IN THE ALIGN FLOWS,/<CRLF>
(1)	060142	046106	053517	042040	.ASCII	\FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 111, TO 014.\
(1)	060232	040600	041040	042101	.ASCII	<CRLF>\A BAD CONSTANT (NOT 25 DEC) \
(1)	060267	127	051501	052440	.ASCIZ	'WAS USED IN THE ALIGN ALGORITHM.'
(0)	060330				EM172:	
(1)	060330	042101	042104	024040	.ASCII	/ADDD (R0),AC0 FAILED IN THE ALIGN FLOWS,/<CRLF>
(1)	060401	106	047514	020127	.ASCII	\FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 011, TO 015.\
(1)	060471	200	020101	040502	.ASCII	<CRLF>\A BAD CONSTANT (NOT 57 DEC) \
(1)	060526	040527	020123	051525	.ASCIZ	'WAS USED IN THE ALIGN ALGORITHM.'
(0)	060567				EM173:	
(1)	060567	101	042104	020104	.ASCII	/ADDD (R0),AC0 FAILED IN THE ALIGN FLOWS,/<CRLF>
(1)	060640	046106	053517	042040	.ASCII	\FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 011, TO 215.\
(1)	060730	040600	041040	042101	.ASCII	<CRLF>\A BAD CONSTANT (NOT 57 DEC) \
(1)	060765	127	051501	052440	.ASCIZ	'WAS USED IN THE ALIGN ALGORITHM.'
(0)	061026				EM174:	
(1)	061026	042101	043104	024040	.ASCII	/ADDE (R0),AC0 FAILED IN THE ALIGN FLOWS,/<CRLF>
(1)	061077	106	047514	020127	.ASCII	\FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 011, TO 015.\
(1)	061167	200	020101	040502	.ASCII	<CRLF>\A BAD CONSTANT (NOT 25 DEC) \
(1)	061224	040527	020123	051525	.ASCIZ	'WAS USED IN THE ALIGN ALGORITHM.'
(0)	061265				EM175:	
(1)	061265	101	042104	020106	.ASCII	/ADDE (R0),AC0 FAILED IN THE ALIGN FLOWS,/<CRLF>
(1)	061336	046106	053517	042040	.ASCII	\FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 011, TO 215.\
(1)	061426	040600	041040	042101	.ASCII	<CRLF>\A BAD CONSTANT (NOT 25 DEC) \
(1)	061463	127	051501	052440	.ASCIZ	'WAS USED IN THE ALIGN ALGORITHM.'
(0)	061524				EM176:	
(1)	061524	042101	042104	024040	.ASCII	'ADDD (R0),AC0 FAILED IN THE ADD-SUB FLOWS,<CRLF>
(1)	061577	104	042111	047040	.ASCIZ	\DID NOT TAKE THE PATH: STATE 216, TO 442, TO 500.\
(0)	061661				EM177:	
(1)	061661	101	042104	020104	.ASCII	'ADDD (R0),AC0 FAILED IN THE ADD-SUB FLOWS,<CRLF>
(1)	061734	044504	020104	047516	.ASCIZ	\DID NOT TAKE THE PATH: STATE 216, TO 042, TO 121.\
(0)	062016				EM200:	
(1)	062016	042101	042104	024040	.ASCII	'ADDD (R0),AC0 FAILED IN THE ADD-SUB FLOWS,<CRLF>
(1)	062071	104	042111	047040	.ASCIZ	\DID NOT TAKE THE PATH: STATE 216, TO 440, TO 121.\
(0)	062153				EM201:	
(1)	062153	101	042104	020104	.ASCII	'ADDD (R0),AC0 FAILED IN THE ADD-SUB FLOWS,<CRLF>
(1)	062226	044504	020104	047516	.ASCIZ	\DID NOT TAKE THE PATH: STATE 216, TO 440, TO 101.\
(0)	062310				EM202:	
(1)	062310	042101	042104	024040	.ASCII	'ADDD (R0),AC0 FAILED IN THE ADD-SUB FLOWS,<CRLF>
(1)	062363	104	042111	047040	.ASCIZ	\DID NOT TAKE THE PATH: STATE 216, TO 042, TO 101.\
(0)	062445				EM203:	

```

(1) 062445      101 042104 020104      .ASCII 'ADDD (R0),AC0 FAILED IN THE ADD-SUB FLOWS,'<CRLF>
(1) 062520 044504 020104 047516      .ASCIZ \DID NOT TAKE THE PATH: STATE 216, TO 440, TO 141,\
(2) 062602      EM204:
(1) 062602 042101 042104 024040      .ASCII 'ADDD (R0),AC0 FAILED IN THE ADD-SUB FLOWS,'<CRLF>
(1) 062655      104 042111 047040      .ASCIZ \DID NOT TAKE THE PATH: STATE 216, TO 042, TO 141,\
(2) 062737      EM205:
(1) 062737      124 042510 043040      .ASCIZ \THE FPS WAS BAD AFTER SUBD (R0),AC0,\
(2) 063004      EM206:
(1) 063004 052523 042102 024040      .ASCIZ /SUBD (R0),AC0 PRODUCED A BAD RESULT./
    063051      123 041125 020104      EM207: .ASCII 'SUBD (R0),AC0 PRODUCED A BAD RESULT,'
    063115      200 044124 020105      .ASCIZ <CRLF>'THE XOR OF THE SIGN BIT FAILED IN STATE 024,'
    063173      101 042104 020104      EM210: .ASCIZ 'ADDD (R0),AC0 FAILED IN THE NORMALIZE FLOWS,'
    043233      EM211=EM5
    043267      EM212=EM6
    043321      EM213=EM7
    
```

;DATA HEADERS

```

063250 020040 042524 052123 DH1:  .ASCII ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF ERROR,'
063310 053411 047522 042524      .ASCIZ <TAB>'WROTE,'<TAB>'READ,'<TAB>'EXPECTED,'
063340 020040 042524 052123 DH2:  .ASCII ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF ERROR,'
063400 047101 020104 040502      .ASCIZ 'AND BAD DATA,'<TAB>'OR BAD DATA,'
063433      040 052040 051505 DH3:  .ASCII ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF ERROR,'
063473      011 042522 042101      .ASCIZ <TAB>'READ PSW,'<TAB>'EXPECTED PSW,'
063524 020040 042524 052123 DH4:  .ASCII ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF ERROR,'
063564 053411 047522 042524      .ASCIZ <TAB>'WROTE FPS,'<TAB>'FPS AFTER CFCC,'
063620 020040 042524 052123 DH5:  .ASCIZ ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF TRAP,'
    063620      DH6=DH5
    063620      DH7=DH5
    063620      DH10=DH5
    063620      DH11=DH5
    000000      DH12=0
    000000      DH13=0
    063620      DH14=DH5
    063620      DH15=DH5
063660 020040 042524 052123 DH16: .ASCII ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF ERROR,'
063720 047411 020120 047503      .ASCIZ <TAB>'OP CODE, FPS,'
063740 020040 042524 052123 DH17: .ASCII ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF ERROR,'
064000 043411 052117 043040      .ASCIZ <TAB>'GOT FPS,'<TAB>'EXPECTED FPS,'
064030 020040 042524 052123 DH20: .ASCII ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF TRAP,'
064067      011 041520 047440      .ASCIZ <TAB>'PC OF STST,'<TAB>'READ FEC,'
    063620      DH21=DH5
064116 040506 046111 042105 DH22: .ASCIZ 'FAILED TO CORRECTLY SET FPS,'
064153      106 044501 042514 DH23: .ASCII 'FAILED TO CORRECTLY SET FEC TO 000002,'<CRLF>
064222 020040 042524 052123      .ASCII ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF ERROR,'
064262 050011 020103 043117      .ASCIZ <TAB>'PC OF STST,'<TAB>'READ FEC,'
064311      124 040522 050120 DH24: .ASCII 'TRAPPED TO 244. FLOW WENT FROM STATE 554 TO STATE 430,'
064377      200 047111 052123      .ASCIZ <CRLF>'INSTEAD OF FROM STATE 554 TO STATE 432,'
064450 020040 042524 052123 DH25: .ASCIZ ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF ERROR,'<TAB>
064512 020040 042524 052123 DH26: .ASCII ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF ERROR,'
064552 043411 052117 051040      .ASCIZ <TAB>'GOT R0,'<TAB>'EXPECTED R0,'
    064512      DH27=DH26
    000000      DH30=0
    064512      DH31=DH26
    064512      DH32=DH26
064600 020040 042524 052123 DH33: .ASCII ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF ERROR,'
    
```

```

064640 051011 020060 052050          .ASCIZ  <TAB>'R0 (TARGET LOCATIONS FOR OUTPUT),'
          064600          DH34=DH33
          064600          DH35=DH33
          064600          DH36=DH33
          064600          DH37=DH33
          064600          DH40=DH33
          000000          DH41=0
064703 040 052040 051505 DH42:  .ASCII  ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF TRAP,'
064742 051011 020060 052050          .ASCIZ  <TAB>'R0 (TARGET LOCATIONS FOR OUTPUT),'
          064703          DH43=DH42
          000000          DH44=0
065005 105 051122 051117 DH45:  .ASCIZ  'ERROR SUMMARY,'
065024 020040 042524 052123 DH46:  .ASCIZ  ' TEST,'<TAB>'CALL AT PC,'
          065005          DH47=DH45
065050 020040 042524 052123 DH50:  .ASCII  ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF ERROR,'
065110 053411 052111 020110          .ASCIZ  <TAB>'WITH FD,'
          065050          DH51=DH50
          063620          DH52=DH5
          064512          DH53=DH26
065122 020040 042524 052123 DH54:  .ASCIZ  ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF ERROR,'
          063620          DH55=DH5
          064512          DH56=DH26
          065122          DH57=DH54
          064512          DH60=DH26
          065122          DH61=DH54
          065122          DH62=DH54
          065122          DH63=DH54
065163 122 051505 046125 DH65:  .ASCII  'RESULTING IN AN ODD ADDRESS TRAP TO 4,'
065231 200          .ASCII  <CRLF>
065232 020040 042524 052123 DH64:  .ASCII  ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF ERROR,'
065272 043411 052117 050040          .ASCIZ  <TAB>'GOT PC,'<TAB>'EXPECTED PC,'
          065122          DH66=DH54
          063620          DH67=DH5
          063620          DH70=DH5
          064450          DH71=DH25
          064512          DH72=DH26
          065122          DH73=DH54
          063620          DH74=DH5
          063620          DH75=DH5
          064450          DH76=DH25
          064512          DH77=DH26
          065122          DH100=DH54
          063620          DH101=DH5
          064512          DH102=DH26
          064450          DH103=DH25
          065122          DH104=DH54
          063620          DH105=DH5
          064512          DH106=DH26
          064450          DH107=DH25
          065122          DH110=DH54
          064450          DH111=DH25
065320 044124 020105 041050 DH112: .ASCII  'THE (BUT FSRC) FORK FAILED,'<CRLF>
065354 047503 052116 047522          .ASCII  'CONTROL WENT FROM STATE 762 TO STATE 627,'
065425 200 047111 052123          .ASCII  <CRLF>'INSTEAD OF FROM STATE 762 TO STATE 637,'<CRLF>
065475 020040 042524 052123          .ASCIZ  ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF ERROR,'
          064450          DH113=DH25
  
```

```

065320 DH114=DH112
065537 124 042510 024040 DH115: .ASCII 'THE (BUT FSRC) FORK FAILED RESULTING IN AN ODD ADDRESS TRAP TO 4.'
065640 041600 047117 051124 .ASCII '<CRLF>'CONTROL WENT FROM STATE 762 TO STATE 627.'<CRLF>
065713 111 051516 042524 .ASCII 'INSTEAD OF FROM STATE 762 TO STATE 627.'<CRLF>
065763 040 052040 051505 .ASCIZ ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF TRAP,'
065537 DH116=DH115
063740 DH117=DH17
066023 040 052040 051505 DH120: .ASCII ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF ERROR,'
066063 011 047507 020124 .ASCIZ <TAB>'GOT FEC,'<TAB>'EXPECTED FEC,'
063620 DH121=DH5
065050 DH122=DH50
065050 DH123=DH50
063740 DH124=DH17
063740 DH125=DH17
063620 DH126=DH5
065122 DH127=DH54
066023 DH130=DH120
065122 DH131=DH54
065122 DH132=DH54
066113 106 044501 042514 DH133: .ASCII 'FAILED TO PRODUCE THE CORRECT RESULTS.'<CRLF>
066162 020040 042524 052123 .ASCIZ ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF ERROR,'
066113 DH134=DH133
066113 DH135=DH133
066113 DH136=DH133
066223 120 047522 052504 DH137: .ASCII 'PRODUCED THE CORRECT RESULT BUT FAILED TO SET THE FPS CORRECTLY.'
066323 040 052040 051505 .ASCII ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF ERROR,'
066363 011 047507 020124 .ASCIZ <TAB>'GOT FPS,'<TAB>'EXPECTED FPS,'
066223 DH140=DH137
066223 DH141=DH137
066223 DH142=DH137
066113 DH143=DH133
066113 DH144=DH133
066223 DH145=DH137
066223 DH146=DH137
065122 DH147=DH54
066413 130 051117 047440 DH150: .ASCII 'XOR OF SIGN BIT FAILED.'<CRLF>
066443 040 052040 051505 .ASCIZ ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF ERROR,'
066223 DH151=DH137
066113 DH152=DH133
066223 DH153=DH137
066504 020040 042524 052123 DH154: .ASCIZ ' TEST,'<TAB>'PC OF CALL,'<TAB>'PC OF TRAP,'
066504 DH155=DH154
065122 DH156=DH54
065122 DH157=DH54
065122 DH160=DH54
065122 DH161=DH54
065122 DH162=DH54
065122 DH163=DH54
063740 DH164=DH17
063740 DH165=DH17
065122 DH166=DH54
065122 DH167=DH54
065122 DH170=DH54
065122 DH171=DH54
065122 DH172=DH54
065122 DH173=DH54

```

```

065122 DH174=DH54
065122 DH175=DH54
065122 DH176=DH54
065122 DH177=DH54
065122 DH200=DH54
065122 DH201=DH54
065122 DH202=DH54
065122 DH203=DH54
065122 DH204=DH54
063740 DH205=DH17
065122 DH206=DH54
065122 DH207=DH54
065122 DH210=DH54
066544 020040 042524 052123 DH211: .ASCIZ ' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF TRAP.'<TAB>'FEC.'
063620 DH212=DH5
063620 DH213=DH5
  
```

;DATA FORMATS:

```

066611 004 000 005 DF1: .BYTE 4,0,5,0,5,0,0,0
066621 004 000 005 DF2: .BYTE 4,0,5,4,5,0,5,0
066631 004 000 005 DF3: .BYTE 4,0,5,0,5,0,5,0
066641 066631 004 000 005 DF4=DF3
066641 004 000 005 DF5: .BYTE 4,0,5,0,5,0,5,11,5,0,5,0
066641 DF6=DF5
066641 DF7=DF5
066641 DF10=DF5
066641 DF11=DF5
066655 005 011 005 DF12: .BYTE 5,11,5,5,5,4,5,4,5,5,4,5,4,5,11,5,11,5,5,4,0,5,0,5,0,0
066707 005 011 005 DF13: .BYTE 5,11,5,5,5,4,0,5,0,5,0,0
066641 DF14=DF6
066641 DF15=DF6
066723 004 000 005 DF16: .BYTE 4,0,5,0,5,0,0
066631 DF17=DF3
066732 004 000 005 DF20: .BYTE 4,0,5,0,5,0,5,0
066742 004 000 005 DF21: .BYTE 4,0,5,0
066746 005 005 004 DF22: .BYTE 5,5,4,0,5,0,5,0,5,0
066760 004 000 005 DF23: .BYTE 4,0,5,0,5,0,5,0
066770 005 004 000 DF24: .BYTE 5,4,0,5,0,5,0
066777 004 000 005 DF25: .BYTE 4,0,5,0,5,5,5,0,5,0,5,0,5,5,5,0,5,0,5,0
067023 004 000 005 DF26: .BYTE 4,0,5,0,5,0,0,5,5,5,5,4,5,4,5,5,5,5,4,5,4
067050 004 000 005 DF27: .BYTE 4,0,5,0,5,0,0,5,5,5,5,4,5,4,5,5
067070 005 011 005 DF30: .BYTE 5,11,5,5,5,4,0,5,0,5,0,0
067023 DF31=DF26
067050 DF32=DF27
067104 004 000 005 DF33: .BYTE 4,0,5,0,5,0,5,5,5,0,5,0,5,12,5,5,5,0,5,0,5,12
067132 004 000 005 DF34: .BYTE 4,0,5,0,5,0,5,5,5,5,4,5,4,5,5,5,5,4,5,4
067132 DF35=DF34
067132 DF36=DF34
067132 DF37=DF34
067156 004 000 005 DF40: .BYTE 4,0,5,0,5,0,5,5,5,0,5,0,5,3,5,5,5,0,5,0,5,3
067204 011 005 005 DF41: .BYTE 11,5,5,5,4,0,5,0,5,0,5,0
067220 004 000 005 DF42: .BYTE 4,0,5,0,5,0,5,5,5,5,4,5,4,11,4,5,5,5,5,4,5,4
067220 DF43=DF42
067246 005 011 005 DF44: .BYTE 5,11,5,5,5,4,0,5,0,5,5,5,5,3,5,5,5,5,3
  
```

067271	005	011	005	DF45:	.BYTE	5,11,5,5,5,4,0,5,0,5,4,5,5,5,5,3,5,5,5,5,3
067316	004	000	005	DF46:	.BYTE	4,0,5,5,5,3,5,3,5,3,5,3,5,3,5,5,5,3,5,3,5,3,5,3
067350	004	000	005	DF47:	.BYTE	4,0,5,0,5,5,5,4,5,4,5,5,
067365	004	000	005	DF50:	.BYTE	4,0,5,0,5,11,5,5,5,5,4,5,4,5,5,5,5,4,5,4
067411	004	000	005	DF51:	.BYTE	4,0,5,0,5,11,5,5,5,3,5,5,5,3
	067350			DF52=DF47		
067427	004	000	005	DF53:	.BYTE	4,0,5,0,5,0,0
067436	004	000	005	DF54:	.BYTE	4,0,5,0,5,5,5,5,3,5,5,5,5,3
	067350			DF55=DF47		
	067427			DF56=DF53		
	067436			DF57=DF54		
	067427			DF60=DF53		
	067436			DF61=DF54		
	067436			DF62=DF54		
	067436			DF63=DF54		
067454	004	000	005	DF64:	.BYTE	4,0,5,0,5,0,0
	067454			DF65=DF64		
	067436			DF66=DF54		
	066742			DF67=DF21		
067463	004	000	005	DF70:	.BYTE	4,0,5,0,5,5,5,5,4,5,4,5,5,5,5,4,5,4
	067463			DF71=DF70		
067505	004	000	005	DF72:	.BYTE	4,0,5,0,5,0,0
	067436			DF73=DF54		
	066742			DF74=DF21		
	067463			DF75=DF70		
	067463			DF76=DF70		
	067505			DF77=DF72		
	067436			DF100=DF54		
	067463			DF101=DF70		
	067505			DF102=DF72		
	067463			DF103=DF70		
	067436			DF104=DF54		
	067463			DF105=DF70		
	067505			DF106=DF72		
	067463			DF107=DF70		
	067436			DF110=DF54		
067514	004	000	005	DF111:	.BYTE	4,0,5,0
	067514			DF112=DF111		
	067514			DF113=DF111		
	067514			DF114=DF111		
	067514			DF115=DF111		
	067514			DF116=DF111		
	066631			DF117=DF3		
	066631			DF120=DF3		
	067350			DF121=DF47		
	067365			DF122=DF50		
	067411			DF123=DF51		
067520	004	000	005	DF124:	.BYTE	4,0,5,0,5,0,0,5,5,5,5,4,5,4,5,5,5,5,4,5,4,5,5,5,3
	067520			DF125=DF124		
	067514			DF126=DF111		
	067514			DF127=DF111		
	066631			DF130=DF3		
067551	004	000	005	DF131:	.BYTE	4,0,5,0,5,5,5,3,5,5,5,3,5,5,5,3
	067551			DF132=DF131		
067571	004	000	005	DF133:	.BYTE	4,0,5,0,5,5,5,3,5,5,5,3,5,5,5,3,5,5,5,3
	067571			DF134=DF133		

```

067571 DF135=DF133
067571 DF136=DF133
067615 004 000 005 DF137: .BYTE 4,0,5,0,5,0,5,0
067615 DF140=DF137
067615 DF141=DF137
067615 DF142=DF137
067571 DF143=DF133
067571 DF144=DF133
067615 DF145=DF137
067615 DF146=DF137
067571 DF147=DF133
067571 DF150=DF133
067615 DF151=DF137
067571 DF152=DF133
067615 DF153=DF137
067625 004 000 005 DF154: .BYTE 4,0,5,0
067625 DF155=DF154
067571 DF156=DF133
067571 DF157=DF133
067571 DF160=DF133
067631 004 000 005 DF161: .BYTE 4,0,5,0,5,5,5,2,5,5,5,2,5,5,5,2,5,5,5,2
067571 DF162=DF133
067631 DF163=DF161
066631 DF164=DF3
066631 DF165=DF3
067571 DF166=DF133
067571 DF167=DF133
067631 DF170=DF161
067631 DF171=DF161
067571 DF172=DF133
067571 DF173=DF133
067631 DF174=DF161
067631 DF175=DF161
067571 DF176=DF133
067571 DF177=DF133
067571 DF200=DF133
067571 DF201=DF133
067571 DF202=DF133
067571 DF203=DF133
067571 DF204=DF133
066631 DF205=DF3
067571 DF206=DF133
067571 DF207=DF133
067571 DF210=DF133
067655 004 000 005 DF211: .BYTE 4,0,5,0,5,0
067655 DF212=DF211
067655 DF213=DF211

067664 .EVEN
;DATA TABLES:

067661 001232 001234 040412 DT1: .WORD STMP0,STMP1,STAB,STMP2,STAB,STMP3
067700 001242 001244 000000 .WORD STMP4,STMP5,0
067706 001232 001234 040412 DT2: .WORD STMP0,STMP1,STAB,AEFFLG,STAB,STMP2,STAB,STMP3,0
067730 001232 001234 040412 DT3: .WORD STMP0,STMP1,STAB,STMP2,STAB,STMP3
067744 240412 001242 000000 .WORD STAB,STMP4,0
    
```

```

067752 067730 DT4=DT3
067770 001232 001234 040412 DT5: .WORD $TMP0,$TMP1,$TAB,$TMP2,LFIEX1,$TMP21,LFIEX2
070004 001272 040531 001240 .WORD $TMP20,FPSMS,$TMP3,FECMS,$TMP4,0
070004 001232 001234 040412 DT6: .WORD $TMP0,$TMP1,$TAB,$TMP2,LFIEX1,$TMP21,LFIEX2,$TMP20,0
070004 DT7=DT6
070004 DT10=DT6
070004 DT11=DT6
070026 040606 001252 040613 DT12: .WORD $THE,$TMP10,NOOP1,NOOP15,NOOP2,$TMP5
070042 040755 001246 040767 .WORD NOOP3,$TMP6,NOOP4,NOOP2,$TMP5,NOOP3,$TMP7,NOOP5,$TMP11
070064 041032 001252 041052 .WORD NOOP6,$TMP10,NOOP7,NOOP10,$TMP0,$TMP1,$TAB,$TMP2
070104 040412 001240 001242 .WORD $TAB,$TMP3,$TMP4,0
070114 040606 001252 040613 DT13: .WORD $THE,$TMP10,NOOP1,NOOP11,NOOP10,$TMP0,$TMP1,$TAB
070134 001236 040412 001240 .WORD $TMP2,$TAB,$TMP3,$TMP4,0
070146 001232 001234 040412 DT14=DT6
070146 001232 001234 040412 DT15=DT6
070146 001232 001234 040412 DT16: .WORD $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP5,$TMP3,0
070166 067730 DT17=DT3
070202 040412 001242 000000 DT20: .WORD $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
070210 001232 001234 040412 .WORD $TAB,$TMP4,0
070222 063433 001313 DT21: .WORD $TMP0,$TMP1,$TAB,$TMP2,0
070226 001232 001234 040412 DT22: .WORD DH3,$CRLF
070242 040412 001242 000000 .WORD $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
070250 001232 001234 040412 .WORD $TAB,$TMP4,0
070264 040412 001242 000000 DT23: .WORD $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
070272 041506 .WORD $TAB,$TMP4,0
070274 001232 001234 040412 DT24: .WORD ILLMS
070310 040412 001242 000000 .WORD $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
070316 001232 001234 040412 .WORD $TAB,$TMP4,0
070342 001242 001313 041566 DT25: .WORD $TMP0,$TMP1,$TAB,$CRLF,MS1,MS3,$TMP3,MS4,$TMP4,$CRLF
070366 001232 001234 040412 .WORD $TMP4,$CRLF,MS2,MS3,$TMP5,MS4,$TMP6,$CRLF,$TMP5,0
070400 041676 001313 041714 DT26: .WORD $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3,$TMP4,$CRLF
070424 041743 001313 041714 .WORD MS6,$CRLF,MS7,$TMP5,MS10,$TMP6,$CRLF
070442 001232 001234 040412 .WORD MS11,$CRLF,MS7,$TMP5,MS10,$TMP7,0
070456 001242 001313 041770 DT27: .WORD $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
070504 042131 001272 001313 .WORD $TMP4,$CRLF,MS12,$CRLF,MS7,$TMP5,MS10,$TMP7,$CRLF,MS13,0
070516 001232 001234 040412 DT30: .WORD MS15,$TMP20,$CRLF,MS14,$CRLF
070532 001242 000000 .WORD $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
070536 070366 DT31=DT26
070552 001313 041554 041573 DT32=DT27
070574 041566 041573 001242 DT33: .WORD $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
070614 001232 001234 040412 .WORD $CRLF,MS1,MS3,$TMP4,MS4,$TMP5,$CRLF,$TMP6,$CRLF
070630 001313 041676 001313 .WORD MS2,MS3,$TMP4,MS4,$TMP5,$CRLF,$TMP4,0
070650 041743 001313 041714 DT34: .WORD $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
070666 070614 .WORD $CRLF,MS6,$CRLF,MS7,$TMP5,MS10,$TMP6,$CRLF
070676 001232 001234 040412 DT35=DT34
070712 040412 001242 000000 DT36=DT34
070720 001232 001234 040412 DT37=DT34
070734 001313 041676 001313 DT40=DT33
070756 001313 041743 001313 DT41: .WORD $TMP20,$CRLF,DH3,$CRLF
070756 001232 001234 040412 .WORD $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
070770 040412 001242 000000 .WORD $TAB,$TMP4,0
070780 001232 001234 040412 DT42: .WORD $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
070794 001313 041676 001313 .WORD $CRLF,MS6,$CRLF,MS7,$TMP5,MS10,$TMP6,$TMP15,$TMP10
070806 001313 041743 001313 .WORD $CRLF,MS11,$CRLF,MS7,$TMP5,MS10,$TMP7,0

```



```

070720 DT43=DT42
070776 042163 001244 001313 DT44: .WORD MS17,STMP5,SCRLF,MS20,SCRLF,STMP0,STMP1,STAB,STMP2
071020 001313 041554 042314 .WORD SCRLF,MS1,MS21,SCRLF,STMP3,SCRLF,MS2,MS21,SCRLF,STMP4,0
071046 042163 001244 001313 DT45: .WORD MS17,STMP5,SCRLF,MS24,SCRLF,STMP0,STMP1,STAB,STMP2,STAB
071072 001246 001313 042353 .WORD STMP6,SCRLF,MS22,MS21,SCRLF,STMP3,SCRLF
071110 042403 042314 001313 .WORD MS23,MS21,SCRLF,STMP4,0
071122 001232 001234 001313 DT46: .WORD STMP0,STMP1,SCRLF,MS25,MS30,STMP2,MS31,STMP3
071142 042611 001242 042620 .WORD MS32,STMP4,MS33,STMP5,MS34,STMP6,SCRLF,MS26
071162 042573 001250 042602 .WORD MS30,STMP7,MS31,STMP10
071172 042611 001254 042620 .WORD MS32,STMP11,MS33,STMP12,MS34,STMP13,0
071210 001232 001234 040412 DT47: .WORD STMP0,STMP1,STAB,STMP2,SCRLF,MS12,MS7,STMP3,MS10
071232 001242 001313 042026 .WORD STMP4,SCRLF,MS13,0

070614 DT50=DT34
071242 001232 001234 040412 DT51: .WORD STMP0,STMP1,STAB,STMP2,STAB,STMP3
071256 001313 042525 042564 .WORD SCRLF,MS25,MS27,STMP4,SCRLF,MS26,MS27,STMP5,0

071210 DT52=DT47
071300 001232 001234 040412 DT53: .WORD STMP0,STMP1,STAB,STMP2,STAB,STMP3
071314 001242 000000 .WORD STMP4,0
071320 001232 001234 040412 DT54: .WORD STMP0,STMP1,STAB,STMP2,SCRLF,MS1,MS21,SCRLF,STMP3
071342 001313 041566 042314 .WORD SCRLF,MS2,MS21,SCRLF,STMP3,0

071210 DT55=DT47
071300 DT56=DT53
071320 DT57=DT54
071300 DT60=DT53
071320 DT61=DT54
071320 DT62=DT54
071320 DT63=DT54
071356 001232 001234 040412 DT64: .WORD STMP0,STMP1,STAB,STMP2,STAB,STMP3
071372 001242 000000 .WORD STMP4,0

071356 DT65=DT64
071320 DT66=DT54
070210 DT67=DT21
071376 001232 001234 040412 DT70: .WORD STMP0,STMP1,STAB,STMP2,SCRLF,MS6,SCRLF,MS7,STMP5
071420 041730 001246 001313 .WORD MS10,STMP6,SCRLF,MS11,SCRLF,MS7,STMP5,MS10,STMP7,0

071376 DT71=DT70
071444 001232 001234 040412 DT72: .WORD STMP0,STMP1,STAB,STMP2,STAB,STMP3,STMP4,0
071320 DT73=DT54
070210 DT74=DT21
071376 DT75=DT70
071376 DT76=DT70
071444 DT77=DT72
071320 DT100=DT54
071376 DT101=DT70
071376 DT102=DT71
071376 DT103=DT70
071320 DT104=DT54
071376 DT105=DT70
071444 DT106=DT72
071376 DT107=DT70
071320 DT110=DT54
071464 001232 001234 040412 DT111: .WORD STMP0,STMP1,STAB,STMP2,0
071464 DT112=DT111
071464 DT113=DT111
071464 DT114=DT111
071464 DT115=DT111
071464 DT116=DT111
  
```

	067730			DT117=DT3	
	067730			DT120=DT3	
	071210			DT121=DT47	
	070614			DT122=DT34	
	071242			DT123=DT51	
071476	001232	001234	040412	DT124: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3,\$TMP4,\$CRLF
071516	041676	001313	041714	.WORD	MS6,\$CRLF,MS7,\$TMP5,MS10,\$TMP6,\$CRLF
071534	041743	001313	041714	.WORD	MS11,\$CRLF,MS7,\$TMP5,MS10,\$TMP7,\$CRLF,MS37,\$CRLF,\$TMP10,0
	071476			DT125=DT124	
	071464			DT126=DT111	
	071464			DT127=DT111	
	067730			DT130=DT3	
071562	001232	001234	040412	DT131: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$CRLF,MS37,\$CRLF,\$TMP3
071602	001313	042670	001313	.WORD	\$CRLF,MS40,\$CRLF,\$TMP4,\$CRLF,MS415,\$CRLF,\$TMP5,0
	071562			DT132=DT131	
071624	001232	001234	040412	DT133: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$CRLF,MS41,\$CRLF,\$TMP3
071644	001313	042750	001313	.WORD	\$CRLF,MS42,\$CRLF,\$TMP4,\$CRLF,MS43,\$CRLF,\$TMP5
071664	001313	043003	001313	.WORD	\$CRLF,MS44,\$CRLF,\$TMP6,0
	071624			DT134=DT133	
	071624			DT135=DT133	
	071624			DT136=DT133	
071676	001232	001234	040412	DT137: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TMP10,\$TAB,\$TMP11,0
	071676			DT140=DT137	
	071676			DT141=DT137	
	071676			DT142=DT137	
	071624			DT143=DT133	
	071624			DT144=DT133	
	071676			DT145=DT137	
	071676			DT146=DT137	
	071624			DT147=DT133	
	071624			DT150=DT133	
	071676			DT151=DT137	
	071624			DT152=DT133	
	071676			DT153=DT137	
071716	001232	001234	040412	DT154: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,0
	071716			DT155=DT154	
	071624			DT156=DT133	
	071624			DT157=DT133	
	071624			DT160=DT133	
	071624			DT161=DT133	
	071624			DT162=DT133	
	071624			DT163=DT133	
	067730			DT164=DT3	
	067730			DT165=DT3	
	071624			DT166=DT133	
	071624			DT167=DT133	
	071624			DT170=DT133	
	071624			DT171=DT133	
	071624			DT172=DT133	
	071624			DT173=DT133	
	071624			DT174=DT133	
	071624			DT175=DT133	
	071624			DT176=DT133	
	071624			DT177=DT133	
	071624			DT200=DT133	
	071624			DT201=DT133	

```
071624 DT202=DT133
071624 DT203=DT133
071624 DT204=DT133
067730 DT205=DT3
071624 DT206=DT133
071624 DT207=DT133
071624 DT210=DT133
071730 001232 001234 040412 DT211: .WORD STMP0,STMP1,STAB,STMP2,STAB,STMP3,0
071746 001232 001234 040412 DT212: .WORD STMP0,STMP1,STAB,STMP2,0
071746 DT213=DT212
```

```
000001 ;12345 .END
```

AADATO	026236	ADDW0 = 000000	A05	004476	BB30	030650	CCER3	027456		
AADONE	026336	ADDW1 = 000000	A1	004350	BB31	030660	CCER4	027474		
AAERR0	025670	ADDW10= 000000	A11	004350	BB32	030672	CCER44	027510		
AAERR1	025756	ADDW11= 000000	A12	004364	BB33	030722	CCER5	027532		
AAERR2	026012	ADDW12= 000000	A2	004422	BB34	030744	CCER50	027376		
AAERR3	026046	ADDW13= 000000	A3	004426	BB35	030754	CCER55	027546		
AAERR4	026056	ADDW14= 000000	A4	004474	BB4	030210	CCER6	027566		
AAERR5	026112	ADDW15= 000000	A5	004500	BB5	030220	CCER7	027622		
AAERR6	026146	ADDW2 = 000000	A6	004526	BB6	030232	CCER8	027640		
AAERR7	026202	ADDW3 = 000000	A7	004552	BB7	030262	CCER90	027344		
AAER10	025764	ADDW4 = 000000	BBDATO	031300	BDONE	004772	CCP0	027774		
AAPAT0	026246	ADDW5 = 000000	BBDONE	031430	BERR	004726	CCP1	030004		
AAPAT1	026256	ADDW6 = 000000	BBEP0	030766	BERR1	004756	CCP10	030074		
AAPAT2	026266	ADDW7 = 000000	BBER1	031026	BIT0 =	000001	CCP11	030104		
AAPAT3	026276	ADDW8 = 000000	BBEP10	031006	BIT00 =	000001	CCP12	030114		
AAPAT4	026306	ADDW9 = 000000	BBER11	031042	BIT01 =	000002	CCP2	030014		
AAPAT5	026316	ADEVCT= 000000	BBER2	031064	BIT02 =	000004	CCP3	030024		
AAPAT6	026326	ADEVVM = 000000	BBER3	031102	BIT03 =	000010	CCP4	030034		
AA1	025310	ADONE	004670	BBER4	031136	BIT04 =	000020	CCP5	030044	
AA10	025424	AENV = 000000	AENV	000000	BBER40	031152	BIT05 =	000040	CCP6	030054
AA11	025454	AENVVM = 000000	AENVVM	000000	BBER5	031172	BIT06 =	000100	CCP7	030064
AA12	025456	AERFLG	004560	BBER6	031210	BIT07 =	000200	CC1	026342	
AA13	025474	AERR1	004562	BBER7	031244	BIT08 =	000400	CC10	026556	
AA14	025514	AERR2	004574	BBER8	031262	BIT09 =	001000	CC11	026566	
AA15	025534	AERR3	004626	BBPAT0	031310	BIT1 =	000002	CC12	026574	
AA16	025542	AFATAL= 000000	AFATAL	000000	BBPAT1	031320	BIT10 =	002000	CC13	026606
AA17	025546	AMADR1= 000000	AMADR1	000000	BBPAT2	031330	BIT11 =	004000	CC14	026636
AA2	025346	AMADR2= 000000	AMADR2	000000	BBPAT3	031340	BIT12 =	010000	CC15	026662
AA20	025552	AMADR3= 000000	AMADR3	000000	BBPAT4	031350	BIT13 =	020000	CC16	026702
AA21	025554	AMADR4= 000000	AMADR4	000000	BBPAT5	031360	BIT14 =	040000	CC17	026712
AA22	025606	AMAMS1= 000000	AMAMS1	000000	BBPAT6	031370	BIT15 =	100000	CC18	026720
AA23	025610	AMAMS2= 000000	AMAMS2	000000	BBP10	031410	BIT2 =	000004	CC19	026732
AA24	025630	AMAMS3= 000000	AMAMS3	000000	BBP11	031420	BIT3 =	000010	CC2	026372
AA25	025650	AMAMS4= 000000	AMAMS4	000000	BBP7	031400	BIT4 =	000020	CC20	026762
AA26	025656	AMSGAD= 000000	AMSGAD	000000	BB1	030130	BIT5 =	000040	CC21	027006
AA27	025662	AMSGLG= 000000	AMSGLG	000000	BB10	030304	BIT6 =	000100	CC22	027026
AA3	025350	AMSGTY= 000000	AMSGTY	000000	BB11	030324	BIT7 =	000200	CC23	027036
AA4	025366	AMTYP1= 000000	AMTYP1	000000	BB12	030334	BIT8 =	000400	CC24	027044
AA5	025406	AMTYP2= 000000	AMTYP2	000000	BB13	030342	BIT9 =	001000	CC25	027056
AA6	025414	AMTYP3= 000000	AMTYP3	000000	BB14	030354	BPTVEC=	000014	CC26	027106
AA7	025422	AMTYP4= 000000	AMTYP4	000000	BB15	030404	B1	004702	CC27	027130
ABASE =	000000	APASS = 000000	APASS	000000	BB16	030430	B2	004704	CC28	027150
ACDW1 =	000000	APRIOR= 000000	APRIOR	000000	BB17	030440	B3	004722	CC29	027160
ACDW2 =	000000	APTCSU= 000040	APTCSU	000040	BB2	030166	CCDATO	027764	CC3	026414
ACPUOP=	000000	APTENV= 000001	APTENV	000001	BB20	030452	CCDONE	030124	CC30	027166
AC0 =	00000000	APTSIZ= 000200	APTSIZ	000200	BB21	030502	CCER0	027326	CC31	027200
AC1 =	00000001	APTSPO= 000100	APTSPO	000100	BB22	030526	CCER1	027362	CC32	027230
AC2 =	00000002	ASWREG= 000000	ASWREG	000000	BB23	030546	CCER10	027674	CC33	027252
AC3 =	00000003	ATESTN= 000000	ATESTN	000000	BB24	030556	CCER11	027712	CC34	027272
AC4 =	00000004	AUNIT = 000000	AUNIT	000000	BB25	030564	CCER12	027730	CC35	027302
AC5 =	00000005	AUSWR = 000000	AUSWR	000000	BB26	030576	CCER13	027746	CC36	027310
AC6 =	00000006	AVECT1= 000000	AVECT1	000000	BB27	030626	CCER2	027420	CC37	027322
AC7 =	00000007	AVECT2= 000000	AVECT2	000000	BB3	030172	CCER22	027434	CC4	026434

CC5	026444	DDP2	033324	DF102 =	067505	DF161	067631	DF46	067316
CC6	026452	DDP3	033334	DF103 =	067463	DF162 =	067571	DF47	067350
CC7	026464	DDP4	033344	DF104 =	067436	DF163 =	067631	DF5	066641
CC8	026514	DDP5	033354	DF105 =	067463	DF164 =	066631	DF50	067365
CC9	026536	DDP6	033364	DF106 =	067505	DF165 =	066631	DF51	067411
CDONE	005676	DDP7	033374	DF107 =	067463	DF166 =	067571	DF52 =	067350
CERR1	005432	DDP8	033404	DF11	066641	DF167 =	067571	DF53	067427
CERR2	005530	DDP9	033414	DF110 =	067436	DF17	066631	DF54	067436
CERR3	005630	DD1	031434	DF111	067514	DF170 =	067631	DF55 =	067350
CERR4	005652	DD10	031634	DF112 =	067514	DF171 =	067631	DF56 =	067427
CFCC1	041344	DD11	031644	DF113 =	067514	DF172 =	067571	DF57 =	067436
CKSWR =	104406	DD12	031662	DF114 =	067514	DF173 =	067571	DF6	066641
CNT =	000214	DD13	031712	DF115 =	067514	DF174 =	067631	DF60	067427
CPC	005674	DD14	031734	DF116 =	067514	DF175 =	067631	DF61	067436
CPSUR	040232	DD15	031744	DF117 =	066631	DF176 =	067571	DF62	067436
CPTWO	040250	DD16	031762	DF12	066655	DF177 =	067571	DF63	067436
CR =	000015	DD17	032012	DF120 =	066631	DF2	066621	DF64	067454
CRLF =	000200	DD18	032034	DF121 =	067350	DF20	066732	DF65	067454
C1	005006	DD19	032054	DF122 =	067365	DF200 =	067571	DF66	067436
C15	005034	DD2	031464	DF123 =	067411	DF201 =	067571	DF67	066742
C2	005054	DD20	032064	DF124	067520	DF202 =	067571	DF7	066641
C25	005070	DD21	032072	DF125 =	067520	DF203 =	067571	DF70	067463
C3	005117	DD22	032104	DF126 =	067514	DF204 =	067571	DF71	067463
C35	005142	DD23	032134	DF127 =	067514	DF205 =	066631	DF72	067505
C4	005164	DD24	032156	DF13	066707	DF206 =	067571	DF73	067436
C45	005176	DD25	032176	DF130 =	066631	DF207 =	067571	DF74	066742
C5	005220	DD26	032206	DF131	067551	DF21	066742	DF75	067463
C55	005246	DD27	032214	DF132 =	067551	DF210 =	067571	DF76	067463
C6	005266	DD3	031506	DF133	067571	DF211	067655	DF77	067505
C65	005302	DD30	032226	DF134 =	067571	DF212 =	067655	DH1	063250
C7	005324	DD31	032256	DF135 =	067571	DF213 =	067655	DH10	063620
C75	005354	DD32	032300	DF136 =	067571	DF22	066746	DH100	065122
C8	005376	DD33	032320	DF137	067615	DF23	066760	DH101	063620
C85	005410	DD34	032330	DF14	066641	DF24	066770	DH102	064512
DDATO	033274	DD35	032336	DF140 =	067615	DF25	066777	DH103	064450
DDONE	033424	DD36	032354	DF141 =	067615	DF26	067023	DH104	065122
DDER0	032506	DD37	032404	DF142 =	067615	DF27	067050	DH105	063620
DDER1	032524	DD38	032426	DF143 =	067571	DF3	066631	DH106	064512
DDER10	033142	DD39	032446	DF144 =	067571	DF30	067070	DH107	064450
DDER11	033200	DD4	031526	DF145 =	067615	DF31 =	067023	DH11	063620
DDER12	033236	DD40	032456	DF146 =	067615	DF32 =	067050	DH110	065122
DDER2	032562	DD41	032464	DF147 =	067571	DF33	067104	DH111	064450
DDER3	032620	DD42	032502	DF15	066641	DF34	067132	DH112	065320
DDER4	032656	DD5	031536	DF150 =	067571	DF35 =	067132	DH113	064450
DDER5	032714	DD6	031544	DF151 =	067615	DF36 =	067132	DH114	065320
DDER6	032752	DD7	031562	DF152 =	067571	DF37 =	067132	DH115	065537
DDER7	033010	DD8	031612	DF153 =	067615	DF4	066631	DH116	065537
DDER8	033046	DERR1	006020	DF154	067625	DF40	067156	DH117	063740
DDER9	033104	DERR2	006114	DF155 =	067625	DF41	067204	DH12	000000
DDISP =	177570	DF1	066611	DF156 =	067571	DF42	067220	DH120	066023
DDONE	006140	DF10 =	066641	DF157 =	067571	DF43 =	067220	DH121	063620
DDP0	033304	DF100 =	067436	DF16	066723	DF44	067246	DH122	065050
DDP1	033314	DF101 =	067463	DF160 =	067571	DF45	067271	DH123	065050

SYMBOL TABLE

DH124 = 063740	DH202 = 065122	DH7 = 063620	DT142 = 071676	DT27 = 070442
DH125 = 063740	DH203 = 065122	DH70 = 063620	DT143 = 071624	DT3 = 067730
DH126 = 063620	DH204 = 065122	DH71 = 064450	DT144 = 071624	DT30 = 070504
DH127 = 065122	DH205 = 063740	DH72 = 064512	DT145 = 071676	DT31 = 070366
DH13 = 000000	DH206 = 065122	DH73 = 065122	DT146 = 071676	DT32 = 070442
DH130 = 066023	DH207 = 065122	DH74 = 063620	DT147 = 071624	DT33 = 070536
DH131 = 065122	DH21 = 063620	DH75 = 063620	DT15 = 070004	DT34 = 070614
DH132 = 065122	DH210 = 065122	DH76 = 064450	DT150 = 071624	DT35 = 070614
DH133 = 066113	DH211 = 066544	DH77 = 064512	DT151 = 071676	DT36 = 070614
DH134 = 066113	DH212 = 063620	DISPLA 001142	DT152 = 071624	DT37 = 070614
DH135 = 066113	DH213 = 063620	DISPRE 000174	DT153 = 071676	DT4 = 067730
DH136 = 066113	DH22 = 064116	DPAT3 016446	DT154 = 071716	DT40 = 070536
DH137 = 066223	DH23 = 064153	DSWR = 177570	DT155 = 071716	DT41 = 070666
DH14 = 063620	DH24 = 064311	DT1 = 067664	DT156 = 071624	DT42 = 070720
DH140 = 066223	DH25 = 064450	DT10 = 070004	DT157 = 071624	DT43 = 070720
DH141 = 066223	DH26 = 064512	DT100 = 071320	DT16 = 070146	DT44 = 070776
DH142 = 066223	DH27 = 064512	DT101 = 071376	DT160 = 071624	DT45 = 071046
DH143 = 066113	DH3 = 063433	DT102 = 071376	DT161 = 071624	DT46 = 071122
DH144 = 066113	DH30 = 000000	DT103 = 071376	DT162 = 071624	DT47 = 071210
DH145 = 066223	DH31 = 064512	DT104 = 071320	DT163 = 071624	DT5 = 067752
DH146 = 066223	DH32 = 064512	DT105 = 071376	DT164 = 067730	DT50 = 070614
DH147 = 065122	DH33 = 064600	DT106 = 071444	DT165 = 067730	DT51 = 071242
DH15 = 063620	DH34 = 064600	DT107 = 071376	DT166 = 071624	DT52 = 071210
DH150 = 066413	DH35 = 064600	DT11 = 070004	DT167 = 071624	DT53 = 071300
DH151 = 066223	DH36 = 064600	DT110 = 071320	DT17 = 067730	DT54 = 071320
DH152 = 066113	DH37 = 064600	DT111 = 071464	DT170 = 071624	DT55 = 071210
DH153 = 066223	DH4 = 063524	DT112 = 071464	DT171 = 071624	DT56 = 071300
DH154 = 066504	DH40 = 064600	DT113 = 071464	DT172 = 071624	DT57 = 071320
DH155 = 066504	DH41 = 000000	DT114 = 071464	DT173 = 071624	DT6 = 070004
DH156 = 065122	DH42 = 064703	DT115 = 071464	DT174 = 071624	DT60 = 071300
DH157 = 065122	DH43 = 064703	DT116 = 071464	DT175 = 071624	DT61 = 071320
DH16 = 063660	DH44 = 000000	DT117 = 067730	DT176 = 071624	DT62 = 071320
DH160 = 065122	DH45 = 065005	DT12 = 070026	DT177 = 071624	DT63 = 071320
DH161 = 065122	DH46 = 065024	DT120 = 067730	DT2 = 067706	DT64 = 071356
DH162 = 065122	DH47 = 065005	DT121 = 071210	DT20 = 070166	DT65 = 071356
DH163 = 065122	DH5 = 063620	DT122 = 070614	DT200 = 071624	DT66 = 071320
DH164 = 063740	DH50 = 065050	DT123 = 071242	DT201 = 071624	DT67 = 070210
DH165 = 063740	DH51 = 065050	DT124 = 071476	DT202 = 071624	DT7 = 070004
DH166 = 065122	DH52 = 063620	DT125 = 071476	DT203 = 071624	DT70 = 071376
DH167 = 065122	DH53 = 064512	DT126 = 071464	DT204 = 071624	DT71 = 071376
DH17 = 063740	DH54 = 065122	DT127 = 071464	DT205 = 067730	DT72 = 071444
DH170 = 065122	DH55 = 063620	DT13 = 070114	DT206 = 071624	DT73 = 071320
DH171 = 065122	DH56 = 064512	DT130 = 067730	DT207 = 071624	DT74 = 070210
DH172 = 065122	DH57 = 065122	DT131 = 071562	DT21 = 070210	DT75 = 071376
DH173 = 065122	DH6 = 063620	DT132 = 071562	DT210 = 071624	DT76 = 071376
DH174 = 065122	DH60 = 064512	DT133 = 071624	DT211 = 071730	DT77 = 071444
DH175 = 065122	DH61 = 065122	DT134 = 071624	DT212 = 071746	D1 = 005724
DH176 = 065122	DH62 = 065122	DT135 = 071624	DT213 = 071746	D10 = 006126
DH177 = 065122	DH63 = 065122	DT136 = 071624	DT22 = 070222	D2 = 005750
DH2 = 063340	DH64 = 065232	DT137 = 071676	DT23 = 070250	D3 = 005752
DH20 = 064030	DH65 = 065163	DT14 = 070004	DT24 = 070272	D4 = 005756
DH200 = 065122	DH66 = 065122	DT140 = 071676	DT25 = 070316	D5 = 005770
DH201 = 065122	DH67 = 063620	DT141 = 071676	DT26 = 070366	D6 = 006004

D7	006014	EM120	053445	EM2	043062	EM64	050412	FERR4	007070
D8	006064	EM121	053601	EM20	044207	EM65	= 050412	FERR5	007174
D9	006076	EM122	053720	EM200	062016	EM66	050546	FERR6	007230
EDONE	006300	EM123	054017	EM201	062153	EM67	050611	FERR7	007364
EEDATO	034116	EM124	054060	EM202	062310	EM7	043321	FER2	007042
EEDONE	034200	EM125	054153	EM203	062445	EM70	051042	FFDAT0	034476
EEER0	033710	EM126	054243	EM204	062602	EM71	051165	FFDONE	034556
EEER1	033726	EM127	054452	EM205	062737	EM72	051267	FFER0	034366
EEER2	033764	EM13	= 000000	EM206	063004	EM73	051343	FFER1	034402
EEER3	034022	EM130	054665	EM207	063051	EM74	051403	FFER2	034440
EEER4	034060	EM131	054765	EM21	044365	EM75	051634	FFP0	034506
EEP0	034126	EM132	055025	EM210	063173	EM76	051757	FFP1	034516
EEP1	034140	EM133	055065	EM211	= 043233	EM77	052061	FFP2	034526
EEP2	034150	EM134	055124	EM212	= 043267	ERM10	035564	FFP3	034536
EEP3	034160	EM135	055163	EM213	= 043321	ERRVEC	= 000004	FFP4	034546
EEP4	034170	EM136	055222	EM22	044516	ERTYPE	037546	FF1	034204
EERR0	006216	EM137	= 055065	EM23	= 044516	ERT1	037730	FF10	034354
EERR1	006234	EM14	043435	EM24	= 044516	ERT2	040146	FF11	034364
EERR2	006250	EM140	= 055124	EM25	044603	ERT3	040152	FF2	034234
EE1	033430	EM141	= 055163	EM26	044716	ERT4	040162	FF3	034256
EE10	033650	EM142	= 055222	EM27	= 044716	ERT5	040174	FF4	034264
EE11	033660	EM143	055261	EM3	043126	E1	006154	FF5	034274
EE12	033666	EM144	055314	EM30	044764	E2	006170	FF6	034324
EE13	033704	EM145	= 055261	EM31	045036	E3	006170	FF7	034346
EE2	033460	EM146	= 055314	EM32	= 045036	E4	006172	FPSMS	040531
EE3	033502	EM147	055347	EM33	045104	FDATIO	010110	FPSPUR	040200
EE4	033522	EM15	043560	EM34	045145	FDATI1	010112	FPVECT	= 000244
EE5	033532	EM150	= 055347	EM35	045247	FDATI2	010114	FXDAT0	010154
EE6	033540	EM151	= 055347	EM36	045351	FDATI3	010116	FXDAT1	010156
EE7	033556	EM152	055401	EM37	045452	FDATI4	010120	FXDAT2	010160
EE8	033606	EM153	= 055401	EM4	043173	FDATI5	010122	FXDAT3	010162
EE9	033630	EM154	055433	EM40	045553	FDATI6	010124	FXDAT4	010164
EMTVEC	= 000030	EM155	055665	EM41	045724	FDATI7	010126	FXDAT5	010166
EM1	043025	EM156	056120	EM42	045761	FDATO0	010132	FXDAT6	010170
EM10	= 043233	EM157	056335	EM43	046102	FDATO1	010134	FXDAT7	010172
EM100	052135	EM16	043703	EM44	046223	FDATO2	010136	F1	006304
EM101	052175	EM160	056554	EM45	= 046223	FDATO3	010140	F10	006516
EM102	052321	EM161	056761	EM46	046266	FDATO4	010142	F11	006520
EM103	052373	EM162	057166	EM47	046344	FDATO5	010144	F12	006536
EM104	052476	EM163	057233	EM5	043233	FDATO6	010146	F13	006570
EM105	052537	EM164	057300	EM50	046462	FDATO7	010150	F135	006550
EM106	052664	EM165	057345	EM51	046560	FDONE	010174	F14	006600
EM107	052737	EM166	057412	EM52	046621	FECMS	040575	F15	006610
EM11	043354	EM167	057522	EM53	046742	FERR0	006702	F16	006620
EM110	053043	EM17	043754	EM54	047137	FERR1	006740	F17	006630
EM111	053105	EM170	057761	EM55	047203	FERR10	007520	F2	006336
EM112	= 053105	EM171	060071	EM56	047324	FERR11	007554	F20	006640
EM113	053207	EM172	060330	EM57	047521	FERR2	007036	F21	006650
EM114	= 053207	EM173	060567	EM6	043267	FERR20	007574	F22	006660
EM115	= 053105	EM174	061026	EM60	047565	FERR21	007712	F23	006676
EM116	= 053207	EM175	061265	EM61	047762	FERR25	007742	F3	006356
EM117	053311	EM176	061524	EM62	050026	FERR26	010060	F4	006360
EM12	= 000000	EM177	061661	EM63	050220	FERR3	007076	F5	006376

F6	006452	G31	013130	IDATI1	011030	JDAT12	016132	LDAT00	016762
F7	006476	G32	013162	IDATI2	011032	JDAT13	016134	LDAT01	016764
GADR	014200	G33	013164	IDATI3	011034	JDAT00	016136	LDAT02	016766
GAND0	014150	G34	013252	IDAT00	011016	JDAT0	016146	LDAT03	016770
GAND1	014152	G35	013304	IDAT01	011020	JDAT01	016140	LDONE	016772
GAND2	014154	G36	013306	IDAT02	011022	JDAT02	016142	LD1	041242
GAND3	014156	G37	013374	IDAT03	011024	JDAT03	016144	LD2	041272
GCMP	013704	G4	012032	IDONE	011036	JDAT1	016150	LERR1	016604
GDAT00	014170	G40	013426	IERR0	010576	JDAT2	016152	LERR2	016656
GDAT01	014172	G41	013430	IERR1	010660	JDAT3	016154	LERR3	016630
GDAT02	014174	G42	013516	IERR2	010700	JDONE	016156	LF	= 000012
GDAT03	014176	G43	013550	IERR25	010722	JERR0	015776	LFIEX1	040417
GDONE	014202	G44	013552	IERR3	010752	JERR1	016044	LFIEX2	040467
GERR1	013732	G5	012064	IERR4	010726	JERR2	016070	LFPS1	041227
GFLAG1	014124	G6	012066	ILLMS	041506	J1	015670	LOOP	004304
GFLAG2	014126	G7	012154	ILL1	041402	J10	016016	LPAT10	016726
GOR0	014160	HADP	014732	ILL2	041445	J2	015714	LPAT11	016730
GOR1	014162	HA1R	015006	IOTVEC=	000020	J3	015716	LPAT12	016732
GOR2	014164	HA1W	014736	IPAT10	010776	J4	015720	LPAT13	016734
GOR3	014166	HA2R	015016	IPAT11	011000	J5	015754	LPAT20	016736
GPAT00	014130	HA2W	014746	IPAT12	011002	J6	015762	LPAT21	016740
GPAT01	014132	HA3R	015026	IPAT13	011004	J7	015774	LPAT22	016742
GPAT02	014134	HA3W	014756	IPAT20	011006	KBUF0	016420	LPAT23	016744
GPAT03	014136	HA4R	015036	IPAT21	011010	KBUF1	016422	LPERR =	104413
GPAT10	014140	HA4W	014766	IPAT22	011012	KBUF2	016424	L1	016456
GPAT11	014142	HA5R	015046	IPAT23	011014	KBUF3	016426	L2	016516
GPAT12	014144	HA5W	014776	I1	010202	KDATI0	016410	L3	016520
GPAT13	014146	HCLR	014662	I10	010364	KDATI1	016412	L4	016522
GRESET	013064	HCLR1	014672	I105	010372	KDATI2	016414	L5	016572
GSETUP	013006	HCMP	014624	I106	010366	KDATI3	016416	L6	016600
GSUM	014022	HCMP1	014644	I11	010374	KDAT00	016430	MDAT00	017312
GS1	013636	HCMP2	014654	I12	010376	KDAT01	016432	MDAT01	017314
GTSWR =	104405	HDAT1	015056	I13	010412	KDAT02	016434	MDAT02	017316
G1	011710	HDAT2	015066	I14	010454	KDAT03	016436	MDAT03	017320
G10	012206	HDAT3	015076	I15	010456	KDONE	016450	MDONE	017322
G11	012210	HDAT4	015106	I16	010460	KERR0	016272	MERR0	017146
G12	012276	HDAT5	015116	I17	010476	KERR1	016336	MERR1	017204
G13	012330	HDONE	015126	I2	010220	KERR2	016362	MERR2	017120
G14	012332	HERROR	014700	I20	010536	KPATO	016440	MERR3	017232
G15	012420	HFLAG	014734	I21	010552	KPAT1	016442	MNUMBE=	000213
G16	012452	HSTD	014546	I22	010556	KPAT2	016444	MNUM0	042210
G17	012454	HT =	000011	I23	010572	K1	016164	MNUM1	042216
G2	011742	H1	014210	I3	010266	K10	016312	MNUM2	042223
G20	012542	H10	014452	I4	010270	K2	016210	MNUM3	042230
G21	012574	H11	014504	I5	010272	K3	016212	MNUM4	042237
G22	012576	H12	014536	I6	010316	K4	016214	MNUM5	042245
G23	012664	H2	014230	I7	010332	K5	016250	MPAT10	017272
G24	012716	H3	014240	JRUF0	016116	K6	016256	MPAT11	017274
G25	012720	H4	014312	JBUF1	016120	K7	016270	MPAT12	017276
G26	013006	H5	014334	JBUF2	016122	LDATI0	016750	MPAT13	017300
G27	013040	H6	014366	JBUF3	016124	LDATI1	016752	MPAT20	017302
G3	011744	H7	014420	JDATI0	016126	LDATI2	016754	MPAT21	017304
G30	013042	IDATI0	011026	JDATI1	016130	LDATI3	016756	MPAT22	017306



MPAT23	017310	NDAT01	017760	ODAT02	020466	PERR16	021006	QERR21	021532
MS1	041554	NDAT02	017762	ODAT03	020470	PERR17	021014	QERR22	021540
MS10	041730	NDAT03	017764	ODONE	020534	PERR2	021124	QERR3	021566
MS11	041743	NDONE	020030	QERR0	020224	PERR20	021042	QERR4	021574
MS12	041770	NERR0	017520	QERR1	020324	PERR21	021052	QPAT10	021650
MS13	042026	NERR1	017620	QERR10	020256	PERR22	021060	QPAT11	021652
MS14	042043	NERR10	017552	QERR11	020270	PIRG =	177772	QPAT12	021654
MS15	042131	NERR11	017564	QERR2	020360	PIRQVE=	000240	QPAT13	021656
MS16	042156	NERR2	017654	QERR20	020332	POWERM	040344	QPAT20	021660
MS17	042163	NERR20	017626	QERR3	020370	PPAT10	021160	QPAT21	021662
MS2	041566	NERR3	017664	QERR4	020400	PPAT11	021162	QPAT22	021664
MS20	042253	NERR4	017674	QERR5	020410	PPAT12	021164	QPAT23	021666
MS21	042314	NERR5	017704	QERR6	020434	PPAT13	021166	Q1	021214
MS22	042353	NERR6	017730	OPAT10	020514	PROGNUM=	000001	Q10	021352
MS23	042403	NOOP1	040613	OPAT11	020516	PR0 =	000000	Q2	021236
MS24	042432	NOOP10	041072	OPAT12	020520	PR1 =	000040	Q3 =	021240
MS25	042525	NOOP11	041163	OPAT13	020522	PR2 =	000100	Q4	021242
MS26	042547	NOOP15	040642	OPAT20	020500	PR3 =	000140	Q5	021264
MS27	042564	NOOP2	040740	OPAT21	020502	PR4 =	000200	Q6	021306
MS3	041573	NOOP3	040755	OPAT22	020504	PR5 =	000240	Q7	021316
MS30	042573	NOOP4	040767	OPAT23	020506	PR6 =	000300	Q8	021330
MS31	042602	NOOP5	041004	OPAT24	020510	PR7 =	000340	Q9	021344
MS32	042611	NOOP6	041032	01	020034	PS =	177776	RDCHR =	104407
MS33	042620	NOOP7	041052	010	020164	PSW =	177776	RESREG=	104411
MS34	042627	NPAT10	020010	011	020174	PWRVEC=	000024	RESVEC=	000010
MS35	042636	NPAT11	020012	012	020176	P1	020540	RSETUP=	104412
MS36	042643	NPAT12	020014	013	020212	P2	020562	R6 =	000006
MS37	042652	NPAT13	020016	014	020222	P3 =	020564	R7 =	000007
MS4	041622	NPAT20	017776	02	020060	P4	020566	SADR	015636
MS40	042670	NPAT21	020000	03	020062	P5	020610	SAVREG=	104410
MS41	042724	NPAT22	020002	04	020064	P6	020632	SDAT00	015652
MS415	042704	NPAT23	020004	05	020102	P7	020642	SDAT01	015654
MS42	042750	NULL	040411	06	020112	P8	020652	SDAT02	015656
MS43	042766	N1	017326	07	020122	QDATI0	021700	SDAT03	015660
MS44	043003	N10	017460	08	020136	QDATI1	021702	SDONE	015662
MS5	041634	N11	017470	09	020150	QDATI2	021704	SERR0	015342
MS6	041676	N12	017472	PDATI0	021170	QDATI3	021706	SERR1	015552
MS7	041714	N13	017506	PDATI1	021172	QDAT00	021670	SERR10	015362
M1	016776	N14	017516	PDATI2	021174	QDAT01	021672	SERR15	015442
M15	017016	N2	017352	PDATI3	021176	QDAT02	021674	SERR2	015502
M2	017022	N3	017354	PDAT00	021200	QDAT03	021676	SERR20	015462
M3	017024	N4	017356	PDAT01	021202	QDONE	021710	SERR3	015526
M4	017026	N5	017374	PDAT02	021204	QERR0	021356	SERR4	015420
M5	017062	N6	017404	PDAT03	021206	QERR1	021622	SERR5	015604
M6	017066	N7	017414	PDONE	021210	QERR11	021366	SERR6	015514
M7	017076	N8	017432	PERR0	020656	QERR12	021404	SERR7	015540
M8	017106	N9	017444	PERP1	021076	QERR13	021422	SETD1	041360
M9	017116	ODATI0	020524	PERR10	020676	QERR14	021440	SETF1	041352
NDATI0	020020	ODATI1	020526	PERR11	020706	QERR15	021456	SETI1	041366
NDATI1	020022	ODATI2	020530	PERR12	020724	QERR16	021466	SETL1	041374
NDATI2	020024	ODATI3	020532	PERR13	020742	QERR17	021474	SPACE	040414
NDATI3	020026	ODAT00	020462	PERP14	020760	QERR2	021556	SPAT10	015642
NDAT00	017756	ODAT01	020464	PERR15	020776	QERR20	021522	SPAT11	015644

SYMBOL TABLE

SPAT12 = 015646	TDAT12 011640	TST5 006142	UPAT33 023000	XDAT01 024340
SPAT13 = 015650	TDAT13 011642	TST6 006302	UPAT40 023002	XDAT02 024342
STACK = 001100	TDAT00 011624	TST7 010176	UPAT41 023004	XDAT03 024344
START 003606	TDAT01 011626	TYPE = 104401	UPAT42 023006	XDONE 024376
STFS1 041303	TDAT02 011630	TYPOC = 104402	UPAT43 023010	XERR1 024222
STKLMT= 177774	TDAT03 011632	TYPON = 104404	UROM1 023020	XERR2 024304
STSI1 041433	TDONE 011644	TYPOS = 104403	UROM2 023022	XERR3 024250
ST1 041316	TERR0 011412	T1 011042	UROM3 023024	XERR4 024320
ST2 041333	TERR1 011474	T10 011204	UTMP1 023014	XPAT00 024346
SWR 001140	TERR2 011514	T105 011212	UTMP2 023016	XPAT01 024350
SWREG 000176	TERR25 011530	T11 011214	U0 021730	XPAT02 024352
SW0 = 000001	TERR3 011560	T12 011216	U1 021760	XPAT03 024354
SW00 = 000001	TERR4 011542	T13 011232	U10 022316	XPAT10 024356
SW01 = 000002	TKVEC = 000060	T14 011274	U11 022320	XPAT12 024362
SW02 = 000004	TPAT10 011604	T15 011276	U12 022352	XPAT13 024364
SW03 = 000010	TPAT11 011606	T16 011300	U13 022370	XPAT20 024366
SW04 = 000020	TPAT12 011610	T17 011316	U14 022422	XPAT21 024370
SW05 = 000040	TPAT13 011612	T2 011062	U15 022452	XPAT22 024372
SW06 = 000100	TPAT20 011614	T20 011356	U16 022454	XPAT23 024374
SW07 = 000200	TPAT21 011616	T21 011372	U2 022024	XTMP 024334
SW08 = 000400	TPAT22 011620	T22 011374	U3 022056	X1 023606
SW09 = 001000	TPAT23 011622	T23 011406	U4 022122	X10 024000
SW1 = 000002	TPVEC = 000064	T3 011130	U5 022154	X11 024014
SW10 = 002000	TRAPVE= 000034	T4 011132	U6 022220	X12 024050
SW11 = 004000	TRTVEC= 000014	T5 011134	U7 022252	X13 024074
SW12 = 010000	TST1 004304	T6 011160	WDAP00 023572	X14 024102
SW13 = 020000	TST10 011040	T7 011174	WDAT01 023574	X15 024116
SW14 = 040000	TST11 011646	UDONE 023026	WDAT02 023576	X16 024152
SW15 = 100000	TST12 014204	UERR0 022472	WDAT03 023600	X17 024176
SW2 = 000004	TST13 015130	UERR1 022516	WDONE 023602	X2 023642
SW3 = 000010	TST14 015664	UERR10 022524	WPAT00 023562	X20 024204
SW4 = 000020	TST15 016160	UERR11 022566	WPAT01 023564	X21 024220
SW5 = 000040	TST16 016452	UERR2 022602	WPAT02 023566	X3 023666
SW6 = 000100	TST17 016774	UERR20 022610	WPAT03 023570	X4 023674
SW7 = 000200	TST2 004672	UERR21 022652	WSETUP 023530	X5 023712
SW8 = 000400	TST20 017324	UERR3 022666	W1 023030	X6 023746
SW9 = 001000	TST21 020032	UERR4 022716	W10 023242	X7 023772
S1 015132	TST22 020536	UFLAG 023012	W11 023270	YDAT00 024742
S10 015300	TST23 021212	UPAT00 022742	W12 023320	YDAT01 024744
S11 015324	TST24 021712	UPAT01 022744	W13 023344	YDAT02 024746
S12 015334	TST25 023026	UPAT02 022746	W14 023360	YDAT03 024750
S2 015166	TST26 023604	UPAT03 022750	W15 023406	YDONE 025002
S3 015170	TST27 024400	UPAT10 022752	W16 023442	YERR1 024602
S4 015174	TST3 004774	UPAT11 022754	W17 023466	YERR2 024644
S5 015220	TST30 025004	UPAT12 022756	W2 023060	YERR3 024706
S6 015230	TST31 025306	UPAT13 022760	W20 023502	YFLAG 024732
S7 015236	TST32 026340	UPAT20 022762	W3 023104	YPAT00 024752
S8 015272	TST33 030126	UPAT21 022764	W4 023122	YPAT01 024754
S9 015274	TST34 031432	UPAT22 022766	W5 023152	YPAT02 024756
TAB = 000011	TST35 033426	UPAT23 022770	W6 023206	YPAT03 024760
TBITVE= 000014	TST36 034202	UPAT30 022772	w7 023232	YPAT10 024762
TDAT10 011634	TST37 034556	UPAT31 022774	XAPT11 024360	YPAT11 024764
TDAT11 011636	TST4 005700	UPAT32 022776	XDAT00 024336	YPAT12 024766

YPAT13	024770	\$BDADR	001122	\$ETABL	001336	\$PWRDN	037362	\$TMP1	001234
YPAT20	024772	\$BDDAT	001126	\$ETEND	001442	\$PWRMG	037516	\$TMP10	001252
YPAT21	024774	\$BELL	001306	\$FATAL	001320	\$PWRUP	037434	\$TMP11	001254
YPAT22	024776	\$CDW1	001376	\$FFLG	036636	\$QUES	001312	\$TMP12	001256
YPAT23	025000	\$CDW2	001400	\$FILLC	001156	\$RDCHR	037122	\$TMP13	001260
YIMP1	024734	\$CHARC	036140	\$FILLS	001155	\$RDSZ =	000001	\$TMP14	001262
YTMP2	024736	\$CKSWR	036640	\$GDADR	001120	\$REGAD	001160	\$TMP15	001264
YTMP3	024740	\$CLR.T	034760	\$GDAT	001124	\$REG0	001162	\$TMP16	001266
Y1	024430	\$CMTAG	001100	\$GET42	034742	\$REG1	001164	\$TMP17	001270
Y2	024452	\$CM1 =	000024	\$GTSWR	036710	\$REG10	001202	\$TMP2	001236
Y3	024476	\$CM2 =	000050	\$HD =	000003	\$REG11	001204	\$TMP20	001272
Y4	024514	\$CM3 =	000024	\$HIBTS	003572	\$REG12	001206	\$TMP21	001274
Y5	024554	\$CM4 =	000024	\$ICNT	001104	\$REG13	001210	\$TMP22	001276
Y6	024556	\$CNTLG	037247	\$ILLUP	037540	\$REG14	001212	\$TMP23	001300
Y7	024572	\$CNTLU	037242	\$INTAG	001135	\$REG15	001214	\$TMP3	001240
ZDAT00	025244	\$CRUOP	001344	\$ITEMB	001114	\$REG16	001216	\$TMP4	001242
ZDAT01	025246	\$CRLF	001313	\$LF	001314	\$REG17	001220	\$TMP5	001244
ZDAT02	025250	\$DDW0	001402	\$LFLG	036635	\$REG2	001166	\$TMP6	001246
ZDAT03	025252	\$DDW1	001404	\$LOOP	035036	\$REG20	001222	\$TMP7	001250
ZDONE	025304	\$DDW10	001426	\$LPADR	001106	\$REG21	001224	\$TN =	000037
ZERR1	025150	\$DDW11	001430	\$LPERR	001110	\$REG22	001226	\$TPB	001152
ZERR2	025212	\$DDW12	001432	\$MADR1	001350	\$REG23	001230	\$TPFLG	001157
ZFLAG	025236	\$DDW13	001434	\$MADR2	001354	\$REG3	001170	\$TPS	001150
ZPAT00	025254	\$DDW14	001436	\$MADR3	001360	\$REG4	001172	\$TRAP	037276
ZPAT01	025256	\$DDW15	001440	\$MADR4	001364	\$REG5	001174	\$TRAP2	037320
ZPAT02	025260	\$DDW2	001406	\$MAIL	001316	\$REG6	001176	\$TRP =	000014
ZPAT03	025262	\$DDW3	001410	\$MAMS1	001346	\$REG7	001200	\$TRPAD	037332
ZPAT10	025264	\$DDW4	001412	\$MAMS2	001352	\$RESRE	035624	\$TSTM	003576
ZPAT11	025266	\$DDW5	001414	\$MAMS3	001356	\$RTNAD	035040	\$TSTM	001102
ZPAT12	025270	\$DDW6	001416	\$MAMS4	001362	\$RTRN	035034	\$TYPE	035662
ZPAT13	025272	\$DDW7	001420	\$MBADR	003574	\$SAVRE	035566	\$TYPEC	036074
ZPAT20	025274	\$DDW8	001422	\$MFLG	036634	\$SAVR6	037544	\$TYPEX	036142
ZPAT21	025276	\$DDW9	001424	\$MNEW	037265	\$SCOPE	035050	\$TYPOC	036170
ZPAT22	025300	\$DEVCT	001326	\$MSGAD	001332	\$SETUP=	000137	\$TYPON	036204
ZPAT23	025302	\$DEVN	001374	\$MSGLG	001334	\$STUP =	177777	\$TYPOS	036144
ZTMP1	025240	\$DOAGN	035000	\$MSGTY	001316	\$SVLAD	035256	\$UNIT	001330
ZTMP2	025242	\$ENDAD	034770	\$MSWR	037254	\$SVPC =	003572	\$UNITM	003602
Z1	025026	\$ENDCT	034612	\$MTYP1	001347	\$SWP =	177400	\$USWR	001342
Z2	025050	\$ENULL	035044	\$MTYP2	001353	\$SWREG	001340	\$VECT1	001366
Z3	025074	\$ENV	001336	\$MTYP3	001357	\$SWRMK=	000000	\$VECT2	001370
Z4	025102	\$ENVM	001337	\$MTYP4	001363	\$SWRMS=	000200	\$XTSTR	035062
Z5	025114	\$EOP	034556	\$MXCNT	035326	\$TAB	040412	\$SGET4=	000001
Z6	025146	\$EOPCT	034604	\$NULL	001154	\$TBIT	035042	\$OFILL	036367
\$APIHD	003572	\$ERFLG	001103	\$NWTST=	000001	\$TERM =	000030	. =	071760
\$ATYC	036416	\$ERMAX	001115	\$OCNT	036366	\$TESTN	001322	.LPER	040266
\$ATY1	036372	\$ERPOR	035330	\$OMODE	036370	\$THE	040606	.RSET	040274
\$ATY3	036400	\$ERRPC	001116	\$OVER	035312	\$TIMES	001302	.SX =	003572
\$ATY4	036410	\$ERRTB	001442	\$PASS	001324	\$TKB	001146		
\$AUTOR	001134	\$ERTTL	001112	\$PASTM	003600	\$TKS	001144		
\$BASE	001372	\$ESCAP	001304	\$PWRAD	037522	\$TMP0	001232		

. ABS. 071760 000

ERRORS DETECTED: 0  
 DEFAULT GLOBALS GENERATED: 0

DSKZ:DEFPAA.BIN,DSKZ:DEFPAA.SEO/SOL\_DEFPAA.P11  
RUN-TIME: 86 73 7 SECONDS  
RUN-TIME RATIO: 244/167=1.4  
CORE USED: 31K (61 PAGES)