

FP11.F

FP11F FLTG PNT PRT A  
CKFPACO

AH-F632C-MC  
FICHE 1 OF 2

AUG 1981  
COPYRIGHT © 79-81  
MADE IN USA





FP11.F

FP11F FLTG PNT PRT A  
CKFPACO

AH-F632C-MC  
FICHE 2 OF 2

AUG 1981  
COPYRIGHT © 79-81  
MADE IN USA





.REM    &

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

IDENTIFICATION

PRODUCT CODE:    AC-F630C-MC  
PRODUCT NAME:    CKFPACO FP11F FLTG PNT PRT A  
DATE CREATED:    APRIL, 1981  
MAINTAINER:      DIAGNOSTIC ENGINEERING  
AUTHOR:          DAN MILLEVILLE

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION.    DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY 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) 1979, 1981 BY DIGITAL EQUIPMENT CORPORATION



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

## HISTORY

NO CHANGES TO THE 11/34 FLOATING POINT DIAGNOSTIC PART 'A' WERE FOUND TO BE NEEDED TO ADAPT IT FOR USE ON THE 11/44.

THE FOLLOWING WAS ADDED TO THE 11/34 FLOATING POINT DIAGNOSTIC TO MAKE THE PART 'B' COVER THE 11/44:

1. TEST 22 - PROCESSOR LOOKS TO SEE IF APT IS CONTROLLING THE TEST, AND IF IT IS, CHECKS TO SEE IF THE USER HAS SELECTED THIS TEST BY CHECKING BIT 7 IN THE SWITCH REGISTER. IT HAS ALSO BEEN CHANGED SO THAT IF BIT 7 IS \*ONE\*, THE CODE WILL SELECT THE TEST.

THE FOLLOWING WAS ADDED TO THE 11/34 FLOATING POINT DIAGNOSTIC TO MAKE THE PART 'C' COVER THE 11/44:

1. TEST 76 - CHECKS THAT FP PROCESSOR DOESN'T ACCESS D-SPACE UNTIL CONDITIONS WARRANT.
2. TESTS 77 THROUGH 106 - CHECK THAT SR1 MATCHES WHAT ACTUALLY HAPPENED TO THE REGISTER OF THE INSTRUCTION, AND THAT THE VALUE OF AUTO INCREMENT /DECREMENT WAS PROPER.

ALL THREE PARTS WERE RE-RELEASED WITH A NEW SYSMAC THAT CHECKS BIT 0 OF THE CPU ERROR REGISTER (POWER MONITOR BIT). THE ADDITIONS WERE MADE IN THE SCOPE ROUTINE, EXECUTED AT THE BEGINNING OF EACH TEST. IF THE BIT BECOMES SET, AN ERROR IS CALLED FROM THE SCOPE ROUTINE. THE BIT IS CLEARED, AND THE TEST IS CONTINUED. IF THE BIT BECOMES SET IN THE MIDDLE OF A TEST, AND AN ERROR OCCURS FOR ANY REASON, THE ERROR ROUTINE WILL CALL \*TWO\* ERRORS, THE POWER MONITOR BIT ERROR FIRST, THEN THE ERROR ORIGINALLY CALLED. IN ADDITION, THE \$READ ROUTINE NOW CHECKS FOR A RANDOMLY INPUTED ^Q BEFORE A ^S IS TYPED. THIS BECAME NECESSARY WITH CERTAIN DATA CONNECTIONS OF SOME SYSTEMS.



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  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125  
126

CONTENTS

- 1. ABSTRACT
- 2. REQUIREMENTS
  - 2.1 EQUIPMENT
  - 2.2 STORAGE
  - 2.3 PRELIMINARY PROGRAMS
- 3. LOADING PROCEDURE
- 4. STARTING PROCEDURE
  - 4.1 CONTROL SWITCH SETTINGS
  - 4.2 STARTING ADDRESS
  - 4.3 PROGRAM 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 CKFPACO
- 10. LISTING
  - 10.1 CKFPACO



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  
169  
170  
171  
172  
173  
174  
175

1.

ABSTRACT

THE THREE PROGRAMS:

CKFPACO CKFPBBO CKFPCCO

ARE DESIGN TO DETECT AND REPORT LOGIC FAULTS IN THE PDP 11/44 FP11-F 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 157 (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 FP11-F. IF THE PROGRAMS OR TESTS ARE NOT RUN IN ORDER THEN ERROR MESSAGES MAY NOT BE ACCURATE.

A. CKFPACO

CKFPACO TESTS:

LDFPS  
STFPS  
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. CKFPBBO

CKFPBBO TESTS:

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



176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230

C. CKFPCCO

CKFPCCO TESTS:

STF AND STD (ALL MODES)  
STCFD AND STCDF  
CLRD AND CLRF  
NEGF AND NEGD  
ABSF AND ABSD  
TSTF AND TSTD  
NEGF, ABSF AND TSTF (ALL SOURCE MODES)  
NEGF, ABSF AND TSTF (ALL SOURCE MODES)  
LDFPS (ALL SOURCE MODES)  
LDCIF AND LDCLF  
LDCID AND LDCLD  
LDEXP  
STFPS (ALL DESTINATION MODES)  
STCFL AND STCFI  
STCDL AND STCDI  
STEXP  
STST

2. REQUIREMENTS

2.1 EQUIPMENT

A PDP 11/44 (WITH OR WITHOUT CONSOLE), LA30 (OR EQUIVALENT) AND AN FP11-F 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 CKFPBBO 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 11/44 CENTRAL PROCESSOR IS FAULTLESS, THEREFORE WHEN IN DOUBT RUN THE PDP 11/44 PROCESSOR DIAGNOSTICS BEFORE THESE FP11-F DIAGNOSTICS.

3. LOADING PROCEDURE

THE PROGRAMS WILL BE SUPPLIED ON THE 11/44 DIAGNOSTIC MEDIA. REFER TO THE XXDP OPERATING MANUAL FOR FURTHER INFORMATION.



231  
232  
233  
234  
235  
236  
237  
238  
239  
240  
241  
242  
243  
244  
245  
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257  
258  
259  
260  
261  
262  
263  
264  
265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275  
276

4. STARTING PROCEDURE

4.1 CONTROL SWITCH SETTINGS

SEE SECTION 5.1

4.2 PROGRAM AND OPERATOR ACTION

1. LOAD PROGRAM INTO MEMORY
2. LOAD ADDRESS 200
3. SET CONSOLE SWITCHES (IF CONSOLE IS PRESENT)
4. PRESS START  
ON FIRST PASS THE PROGRAM WILL IDENTIFY ITSELF. NOTE THAT IF THERE IS NO PHYSICAL CONSOLE THE PROGRAM WILL REQUEST THE OPERATOR FOR INITIAL VALUE FOR THE SOFTWARE SWITCH REGISTER (SEE SECTION 8.5). IF RUNNING UNDER ACT, APT OR CHAIN THIS DOES NOT APPLY.
5. THE PROGRAM WILL LOOP AND AN END OF PASS AND ERROR SUMMARY WILL BE TYPED AT THE END OF EVERY PASS.

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 CKFPACO.
SW<7>=1....	200	SELECT CORRECT INTERRUPT TEST IN PROGRAM CKFPBBO.



277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331  
332

6. ERRORS

6.1 SUMMARIES

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

6.2 ERROR RECOVERY

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

SW<15>=1... THE PROGRAM WILL HALT AFTER TYPING THE ERROR MESSAGE. PRESSING THE CONSOLE CONTINUE WILL CAUSE THE PROGRAM TO CONTINUE AS IF SW<15>=0.

7. RESTRICTIONS

NONE

8. MISCELLANEOUS

8.1 EXECUTION TIMES

LESS THAN 10 SECONDS FOR EACH PROGRAM ON ANY PASS.

8.2 STACK POINTER

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

8.3 PASS COUNT

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



333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382

#### 8.4 T-BIT TRAPPING

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

#### 8.5 SOFTWARE SWITCH REGISTER

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

#### 8.6 INTERRUPTS TEST

IN PROGRAM CKFPBBO THERE IS A SPECIAL TEST FOR CHECKING THE CORRECT FLOWS OF THE FPP. THIS TEST CAN BE RUN ONLY IF A SPECIAL TEST MODULE IS IN THE SYSTEM. THIS MODULE WILL PROBABLY ONLY BE USED IN MANUFACTURING. IF THIS MODULE IS NOT IN THE SYSTEM THIS TEST WILL AUTOMATICALLY BE DESELECTED. IF THIS TEST MODULE IS ON THE SYSTEM AND SW<7>=1 THIS TEST WILL BE RUN. IF SW<7>=0 THIS TEST WILL BE DESELECTED.

#### 8.7 ACT, APT AND XXDP COMPATIBILITY

THESE PROGRAMS ARE FULLY COMPATIBLE WITH:

APT  
ACT  
XXDP MONITOR AND CHAIN PROGRAMS.



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

## 9. PROGRAM DESCRIPTION

## TEST 1 LDFPS, STFPS AND DATA PATHS TEST

THIS IS A TEST OF THE LDFPS (LOAD FLOATING POINT STATUS) AND STFPS (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 DMO AND SMO 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 RECEIVE 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 CFCC TEST

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

## 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.



432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483

TEST 4                    ILLEGAL FPP OP CODES AND STST TEST

THIS IS A TEST OF THE FPP OPERATION CODES:

170003  
170004  
:  
170010  
170013  
170014  
:  
170077

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

TEST 5                    FID, INTERRUPT DISABLE, BIT TEST

THIS IS A TEST OF FPS BIT 14 (FID) OR FLOATING INTERRUPT DISABLE. AN ILLEGAL INSTRUCTION IS EXECUTED WITH FID=1. NO INTERRUPT SHOULD OCCUR.

TEST 6                    LDD AND STD, WITH SRC AND DST MODE 1, TEST

THIS IS A TEST OF BOTH THE INSTRUCTION:

LDD (R0),ACO

AND THE INSTRUCTION:

STD ACO,(R0)

MOST OF THE FAILURES ARE ISOLATED TO THE SRC OR DST FLOWS. NOTE THAT THE INTEGRITY OF ACO HAS NOT BEEN ASSURED. THIS MEANS THAT IN SOME CASES IT WILL BE IMPOSSIBLE TO ISOLATE CERTAIN DATA PATTERN FAILURES TO EITHER THE FLOWS OR THIS ACCUMULATOR.

TEST 7                    FSRC MODE 0 TEST

THIS IS A TEST OF FSRC MODE ZERO USING THE LDD AND LDF INSTRUCTIONS.

TEST 10                  FDST MODE 0 TEST

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



## TEST 11

## ACCUMULATORS DATA PATTERNS TEST

484  
485  
486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532

THIS IS A TEST OF THE FLOATING POINT PROCESSOR ACCUMULATORS.

EACH ACCUMULATOR IS TESTED IN TWO WAYS:

- 1 TEST PATTERN GENERATED BY FLOATING A ONE ACROSS A FIELD OF ZEROES.
- 2 TEST PATTERN GENERATED BY FLOATING A ZERO ACROSS A FIELD OF ONES.

EACH OF ACCUMULATORS AC0 THROUGH AC5 IS TESTED.

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 E. IF THE USER HAS THE INHIBIT ERROR TYPE OUT SW1 (SWR13) OFF, THEN THE USER WILL RECEIVE AN 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.

THE FOLLOWING PROCEDURE IS PRESENTED TO AID THE TROUBLE SHOOTER IN SITUATIONS WHERE AM2901 CHIP ISOLATION IS ATTEMPTED.

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

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

- A. FOUR SIXTEEN BIT NUMBERS LABELED THE LOGICAL 'AND' ('\*') OF THE FAILING DATA PATTERNS.
- B. FOUR SIXTEEN BIT NUMBERS LABELED THE LOGICAL 'OR' ('+') OF THE FAILING DATA PATTERNS.



533  
 534  
 535  
 536  
 537  
 538  
 539  
 540  
 541  
 542  
 543  
 544  
 545  
 546  
 547  
 548  
 549  
 550  
 551  
 552  
 553  
 554  
 555  
 556  
 557  
 558  
 559  
 560  
 561  
 562  
 563  
 564  
 565  
 566  
 567  
 568  
 569  
 570  
 571  
 572  
 573  
 574

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

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

THUS IF A FAILURE OCCURS:

- A. STUCK HIGHS WILL SHOW AS 0'S IN THE 'OR' PATTERN.
- B. STUCK LOWS WILL SHOW AS 1'S IN THE 'AND' PATTERN.

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

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

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

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

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



	BITS -----	AM2901 CHIP NUMBER -----
575		
576		
577		
578	A15,A14,A13,A12	E37
579	A11,A10,A9,A8	E45
580	A7,A6,A5,A4	E34
581	A3,A2,A1,A0	E42
582		
583	B15,B14,B13,B12	E33
584	B11,B10,B9,B8	E41
585	B7,B6,B5,B4	E36
586	B3,B2,B1,B0	E44
587		
588	C15,C14,C13,C12	E35
589	C11,C10,C9,C8	E43
590	C7,C6,C5,C4	E38
591	C3,C2,C1,C0	E46
592		
593	D15,D14,D13,D12	E39
594	D11,D10,D9,D8	E47
595	D7,D6,D5,D4	E40
596	D3,D2,D1,D0	E48

NOW FIVE IMPORTANT CASES WHICH WILL ARRISE WHEN A FAULTY AM2901 IS PRESENT CAN BE DESCRIBED:

- 1.) IF ONLY ONE BIT OF THE 64 BITS IS INCORRECT 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.

575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593  
594  
595  
596  
597  
598  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608  
609  
610  
611  
612  
613  
614  
615  
616  
617  
618  
619  
620  
621  
622



623  
624  
625  
626  
627  
628  
629  
630  
631  
632  
633  
634  
635  
636  
637  
638  
639  
640  
641  
642  
643  
644  
645  
646  
647  
648  
649  
650  
651  
652  
653  
654  
655  
656  
657  
658  
659  
660  
661  
662  
663  
664  
665  
666  
667  
668  
669  
670  
671  
672  
673  
674  
675  
676

- 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, AN+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  
ABOVE PATTERNS, THEN THE TROUBLE SHOOTER  
MUST INTUITIVELY TRY TO FIND WHICH OF THE  
ABOVE CASES (1 THROUGH 4) IS A 'BEST FIT' OF  
THE SYMPTOMS.

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.

TEST 13  
-----

FSRC MODE 0 WITH ILLEGAL ACCUMULATOR TEST  
-----

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



677  
678  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689  
690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
700  
701  
702  
703  
704  
705  
706  
707  
708  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719

TEST 14                    FSRC MODE 2 TEST  
                              THIS IS A TEST OF FSRC MODE 2, AUTO INCREMENT MODE.

TEST 15                    FSRC MODE 4 TEST  
                              THIS IS A TEST OF FSRC MODE 4, AUTO DECREMENT MODE.

TEST 16                    FSRC MODE 2, WITH FD=0, TEST  
                              THIS IS A TEST OF FSRC MODE 2 WITH FD=0. (AUTO INCREMENT)

TEST 17                    FSRC MODE 2 WITH GR7, IMMEDIATE MODE, TEST  
                              THIS IS A TEST OF FSRC MODE 2 USING GR7 (THE PC). THIS IS IMMEDIATE MODE.

TEST 20                    FSRC MODE 3 TEST  
                              THIS IS A TEST OF FSRC MODE 3, AUTO INCREMENT DEFERRED

TEST 21                    FSRC MODE 5 TEST  
                              THIS IS A TEST OF FSRC MODE 5, AUTO DECREMENT DEFERRED.

TEST 22                    FSRC MODE 6 TEST  
                              THIS IS A TEST OF FSRC MODE 6, INDEX MODE

TEST 23                    FSRC MODE 7 TEST  
                              THIS IS A TEST OF FSRC MODE 7, INDEX DEFERRED MODE.



720  
721  
722  
723  
724  
725  
726  
727  
728  
729  
730  
731  
732  
733  
734  
735  
736  
737  
738  
739  
740  
741  
742  
743  
744  
745  
746  
747  
748  
749  
750  
751  
752  
753  
754  
755  
756  
757  
758  
759  
760  
761  
762  
763  
764  
765  
766  
767  
768  
769  
770  
771  
772

TEST 24                    (BUT EZBT Y8), (BUT ENBT) AND (BUT FIUV) TEST

THIS IS A TEST OF THE (BUT EZBT Y8) FORK, THE (BUT ENBT) FORK AND (BUT FIUV) FORK IN THE LOAD INSTRUCTION FLOWS. EACH OF THE PATTERNS:

- 0
- +NUM
- NUM
- 0

IS LOADED TWICE, ONCE WITH AC>0 THEN WITH AC=0. AFTER EACH LOAD THE FPS IS CHECK TO INSURE THAT CONTROL WAS PASSED THROUGH WITH THE FORKS PROPERLY.

TEST 25                    ADDF, ADDD, SUBF AND SUBD WITH FSRC=AC=0 TEST

THIS IS A TEST OF ADD AND SUB WITH FSRC=AC=0

TEST 26                    ADDD AND SUB WITH FSRC=0

THIS IS A TEST OF ADD AND SUB WITH FSRC=0.

TEST 27                    SUBD WITH AC=0 TEST

THIS IS A TEST OF SUBD WITH AC=0. BOTH POSITIVE AND NEGATIVE FSRC'S ARE TRIED.

TEST 30                    ADDD WITH AC=0 TEST

POSITIVE AND NEGATIVE FSRC'S ARE TRIED.

TEST 31                    ADDF AND ADDD WITH E(AC)=E(FSRC) AND (BUT FT) TEST

THIS IS A TEST OF THE ADD INSTRUCTION WITH THE OPERANDS HAVING EQUAL EXPONENTS. THE (BUT FT) FORK IN THE ROUND/TRUNK FLOWS IS ALSO TESTED.

TEST 32                    ADDF AND ADDD WITH E(AC) LESS THAN E(FSRC) TEST

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



773  
774  
775  
776  
777  
778  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
800  
801  
802

TEST 33                    ADDF AND ADDD WITH E(AC) GREATER THAN E(FSRC) TEST

THIS IS A TEST OF THE ADDD AND ADDF INSTRUCTIONS AND THE ALIGN FSRC ALGORITHM FLOWS. FIRST THE CONSTANT USED IS CHECKED. THEN SIMPLE AND WORST CASE ALIGNMENT SITUATIONS ARE TRIED. NOTE E(AC) IS GREATER THAN E(FSRC).

TEST 34                    ADDD WITH NEGATIVE OPRANDS TEST

THIS IS A TEST OF THE ADDD INSTRUCTION WITH NEGATIVE OPRANDS. EVERY COMBINATION OF OPRAND SIGNS IS TRIED.

TEST 35                    SUBD TEST

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

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.



803  
804  
805  
806  
807  
808  
809  
1403  
1404  
1405  
1406  
1407  
1408  
1409

000000

000214  
000001

1410  
1411  
1412  
1413  
1414  
1415

000001  
160000  
000244  
177400  
000200  
000011  
000015

001100  
104000  
000004

000011  
000012  
000015  
000200  
177776  
177776  
177774  
177772  
177570  
177570

000000  
000001  
000002  
000003  
000004  
000005  
000006  
000007  
000006  
000007

10.

LISTING

```

&
MNUMBER=214
PROGNUM=1
.LIST ME
.NLIST MD,MC,CND
.ENABL ABS
.MCALL .HEADER, .SWRH, .EQUAT, .SETUP, .SCATCH, .SACT11, .SCMTAG
.MCALL .STYPE, .SEOP, .SSAVE, .STYPOCT
.MCALL .STYPDEC, .STRAJ, .SPOWER, .SAPTHDR, .SAPTBL
.MCALL .SAPTYPE, .SREAD
.MCALL .EQUIV ;*REMOVE FOR ASSEMBLY ON PDP-10
.TITLE CKFPACO FP11F FLTG PNT PRT A
;*COPYRIGHT (C) 1981
;*DIGITAL EQUIPMENT CORP.
;*MAYNARD, MASS. 01754
;*
;*
;*THIS PROGRAM WAS ASSEMBLED USING THE PDP-11 MAINDEC SYSMAC
;*PACKAGE (MAINDEC-11-DZQAC-C5), JAN, 1981.
;*
$TN=1
$SWR=160000 ;:HALT ON ERROR, LOOP ON TEST, INHIBIT ERROR TYP0UT
FPVECT=244
$SWR=177400
$SWRMSK=200
TAB=11
CRLF=15
.SBTTL BASIC DEFINITIONS
;*INITIAL ADDRESS OF THE STACK POINTER *** 1100 ***
STACK= 1100
ERROR=EMT
SCOPE=IOT
;*MISCELLANEOUS DEFINITIONS
HT= 11 ;:CODE FOR HORIZONTAL TAB
LF= 12 ;:CODE FOR LINE FEED
CR= 15 ;:CODE FOR CARRIAGE RETURN
CRLF= 200 ;:CODE FOR CARRIAGE RETURN-LINE FEED
PS= 177776 ;:PROCESSOR STATUS WORD
PSW=PS
STKLMT= 177774 ;:STACK LIMIT REGISTER
PIRQ= 177772 ;:PROGRAM INTERRUPT REQUEST REGISTER
DSWR= 177570 ;:HARDWARE SWITCH REGISTER
DDISP= 177570 ;:HARDWARE DISPLAY REGISTER
;*GENERAL PURPOSE REGISTER DEFINITIONS
R0= %0 ;:GENERAL REGISTER
R1= %1 ;:GENERAL REGISTER
R2= %2 ;:GENERAL REGISTER
R3= %3 ;:GENERAL REGISTER
R4= %4 ;:GENERAL REGISTER
R5= %5 ;:GENERAL REGISTER
R6= %6 ;:GENERAL REGISTER
R7= %7 ;:GENERAL REGISTER
SP= %6 ;:STACK POINTER
PC= %7 ;:PROGRAM COUNTER
;*PRIORITY LEVEL DEFINITIONS
    
```



000000	PR0=	0	::PRIORITY LEVEL 0
000040	PR1=	40	::PRIORITY LEVEL 1
000100	PR2=	100	::PRIORITY LEVEL 2
000140	PR3=	140	::PRIORITY LEVEL 3
000200	PR4=	200	::PRIORITY LEVEL 4
000240	PR5=	240	::PRIORITY LEVEL 5
000300	PR6=	300	::PRIORITY LEVEL 6
000340	PR7=	340	::PRIORITY LEVEL 7

;'SWITCH REGISTER' SWITCH DEFINITIONS

100000	SW15=	100000
040000	SW14=	40000
020000	SW13=	20000
010000	SW12=	10000
004000	SW11=	4000
002000	SW10=	2000
001000	SW09=	1000
000400	SW08=	400
000200	SW07=	200
000100	SW06=	100
000040	SW05=	40
000020	SW04=	20
000010	SW03=	10
000004	SW02=	4
000002	SW01=	2
000001	SW00=	1

SW9=SW09  
SW8=SW08  
SW7=SW07  
SW6=SW06  
SW5=SW05  
SW4=SW04  
SW3=SW03  
SW2=SW02  
SW1=SW01  
SW0=SW00

;'DATA BIT DEFINITIONS (BIT00 TO BIT15)

100000	BIT15=	100000
040000	BIT14=	40000
020000	BIT13=	20000
010000	BIT12=	10000
004000	BIT11=	4000
002000	BIT10=	2000
001000	BIT09=	1000
000400	BIT08=	400
000200	BIT07=	200
000100	BIT06=	100
000040	BIT05=	40
000020	BIT04=	20
000010	BIT03=	10
000004	BIT02=	4
000002	BIT01=	2
000001	BIT00=	1

BIT9=BIT09  
BIT8=BIT08  
BIT7=BIT07  
BIT6=BIT06  
BIT5=BIT05



000020  
000010  
000004  
000002  
000001

BIT4=BIT04  
BIT3=BIT03  
BIT2=BIT02  
BIT1=BIT01  
BIT0=BIT00

```
;*BASIC "CPU" TRAP VECTOR ADDRESSES
ERRVEC= 4          ;; TIME OUT AND OTHER ERRORS
RESVEC= 10         ;; RESERVED AND ILLEGAL INSTRUCTIONS
TBITVEC=14        ;; 'T' BIT
TRIVEC= 14        ;; TRACE TRAP
BPTVEC= 14        ;; BREAKPOINT TRAP (BPT)
IOTVEC= 20        ;; INPUT/OUTPUT TRAP (IOT) **SCOPE**
PWRVEC= 24        ;; POWER FAIL
EMTVEC= 30        ;; EMULATOR TRAP (EMT) **ERROR**
TRAPVEC=34        ;; "TRAP" TRAP
TKVEC= 60         ;; TTY KEYBOARD VECTOR
TPVEC= 64         ;; TTY PRINTER VECTOR
PIRQVEC=240       ;; PROGRAM INTERRUPT REQUEST VECTOR
```

1416  
1417  
1418  
1419  
1420  
1421  
1422  
1423  
1424  
1426

000000  
000001  
000002  
000003  
000004  
000005  
000006  
000007

.SBTTL FPP REGISTER DEFINITIONS

AC0 =%0  
AC1 =%1  
AC2 =%2  
AC3 =%3  
AC4 =%4  
AC5 =%5  
AC6 =%6  
AC7 =%7

.SBTTL TRAP CATCHER

000000

. =0  
;\*ALL UNUSED LOCATIONS FROM 4 - 776 CONTAIN A ".+2,HALT"  
;\*SEQUENCE TO CATCH ILLEGAL TRAPS AND INTERRUPTS  
;\*LOCATION 0 CONTAINS 0 TO CATCH IMPROPERLY LOADED VECTORS

000174 000174  
000174 000000  
000176 000000

. =174  
DISPREG: .WORD 0 ;; SOFTWARE DISPLAY REGISTER  
SWREG: .WORD 0 ;; SOFTWARE SWITCH REGISTER

000200 000137 003616

.SBTTL STARTING ADDRESS(ES)  
JMP @#START ;; JUMP TO STARTING ADDRESS OF PROGRAM



1427

.SBTTL COMMON TAGS

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

001100 001100  
001100 000000  
001102 000  
001103 000  
001104 000000  
001106 000000  
001110 000000  
001112 000000  
001114 000  
001115 001  
001116 000000  
001120 000000  
001122 000000  
001124 000000  
001126 000000  
001130 000000  
001132 000000  
001134 000  
001135 000  
001136 000000  
001140 177570  
001142 177570  
001144 177560  
001146 177562  
001150 177564  
001152 177566  
001154 000  
001155 002  
001156 012  
001157 000  
001160 000000

.=1100  
\$CMTAG: .WORD 0  
\$TSTNM: .BYTE 0  
\$ERFLG: .BYTE 0  
\$ICNT: .WORD 0  
\$LPADR: .WORD 0  
\$LPERR: .WORD 0  
\$ERTTL: .WORD 0  
\$ITEMB: .BYTE 0  
\$ERMAX: .BYTE 1  
\$ERRPC: .WORD 0  
\$GDADR: .WORD 0  
\$BDADR: .WORD 0  
\$GDDAT: .WORD 0  
\$BDDAT: .WORD 0  
\$AUTOB: .BYTE 0  
\$INTAG: .BYTE 0  
\$SWR: .WORD DSWR  
\$DISPLAY: .WORD DDISP  
\$TKS: 177560  
\$TKB: 177562  
\$TPS: 177564  
\$TPB: 177566  
\$NULL: .BYTE 0  
\$FILLS: .BYTE 2  
\$FILLC: .BYTE 12  
\$TPFLG: .BYTE 0  
\$REGAD: .WORD 0

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

000024  
001162 000000  
001164 000000  
001166 000000  
001170 000000  
001172 000000  
001174 000000  
001176 000000  
001200 000000  
001202 000000  
001204 000000  
001206 000000  
001210 000000  
001212 000000  
001214 000000  
001216 000000  
001220 000000  
001222 000000  
001224 000000  
001226 000000

.REPT \$CM3  
\$REG0: .WORD 0  
\$REG1: .WORD 0  
\$REG2: .WORD 0  
\$REG3: .WORD 0  
\$REG4: .WORD 0  
\$REG5: .WORD 0  
\$REG6: .WORD 0  
\$REG7: .WORD 0  
\$REG10: .WORD 0  
\$REG11: .WORD 0  
\$REG12: .WORD 0  
\$REG13: .WORD 0  
\$REG14: .WORD 0  
\$REG15: .WORD 0  
\$REG16: .WORD 0  
\$REG17: .WORD 0  
\$REG20: .WORD 0  
\$REG21: .WORD 0  
\$REG22: .WORD 0

:::CONTAINS ((\$REGAD)+0)  
:::CONTAINS ((\$REGAD)+2)  
:::CONTAINS ((\$REGAD)+4)  
:::CONTAINS ((\$REGAD)+6)  
:::CONTAINS ((\$REGAD)+10)  
:::CONTAINS ((\$REGAD)+12)  
:::CONTAINS ((\$REGAD)+14)  
:::CONTAINS ((\$REGAD)+16)  
:::CONTAINS ((\$REGAD)+20)  
:::CONTAINS ((\$REGAD)+22)  
:::CONTAINS ((\$REGAD)+24)  
:::CONTAINS ((\$REGAD)+26)  
:::CONTAINS ((\$REGAD)+30)  
:::CONTAINS ((\$REGAD)+32)  
:::CONTAINS ((\$REGAD)+34)  
:::CONTAINS ((\$REGAD)+36)  
:::CONTAINS ((\$REGAD)+40)  
:::CONTAINS ((\$REGAD)+42)  
:::CONTAINS ((\$REGAD)+44)



```
001230 000000 $REG23: .WORD 0 ::CONTAINS (($REGAD)+46)
001232 000024 .REPT 24
001234 000000 $TMP0: .WORD 0 ::USER DEFINED
001236 000000 $TMP1: .WORD 0 ::USER DEFINED
001240 000000 $TMP2: .WORD 0 ::USER DEFINED
001242 000000 $TMP3: .WORD 0 ::USER DEFINED
001244 000000 $TMP4: .WORD 0 ::USER DEFINED
001246 000000 $TMP5: .WORD 0 ::USER DEFINED
001250 000000 $TMP6: .WORD 0 ::USER DEFINED
001252 000000 $TMP7: .WORD 0 ::USER DEFINED
001254 000000 $TMP10: .WORD 0 ::USER DEFINED
001256 000000 $TMP11: .WORD 0 ::USER DEFINED
001260 000000 $TMP12: .WORD 0 ::USER DEFINED
001262 000000 $TMP13: .WORD 0 ::USER DEFINED
001264 000000 $TMP14: .WORD 0 ::USER DEFINED
001266 000000 $TMP15: .WORD 0 ::USER DEFINED
001270 000000 $TMP16: .WORD 0 ::USER DEFINED
001272 000000 $TMP17: .WORD 0 ::USER DEFINED
001274 000000 $TMP20: .WORD 0 ::USER DEFINED
001276 000000 $TMP21: .WORD 0 ::USER DEFINED
001300 000000 $TMP22: .WORD 0 ::USER DEFINED
001302 000000 $TMP23: .WORD 0 ::USER DEFINED
001304 000000 $TIMES: 0 ::MAX. NUMBER OF ITERATIONS
001306 207 377 377 $ESCAPE: 0 ::ESCAPE ON ERROR ADDRESS
001311 000 $BELL: .ASCIZ <207><377><377> ::CODE FOR BELL
001312 077 $QUES: .ASCII /?/ ::QUESTION MARK
001313 015 $CRLF: .ASCII <15> ::CARRIAGE RETURN
001314 012 000 $LF: .ASCIZ <12> ::LINE FEED
*****
.SBTTL APT MAILBOX-ETABLE
*****
.EVEN
001316 $MAIL: ::APT MAILBOX
001316 000000 $MSGTY: .WORD AMSGTY ::MESSAGE TYPE CODE
001320 000000 $FATAL: .WORD AFATAL ::FATAL ERROR NUMBER
001322 000000 $TESTN: .WORD ATESTN ::TEST NUMBER
001324 000000 $PASS: .WORD APASS ::PASS COUNT
001326 000000 $DEVCT: .WORD ADEVCT ::DEVICE COUNT
001330 000000 $UNIT: .WORD AUNIT ::I/O UNIT NUMBER
001332 000000 $MSGAD: .WORD AMSGAD ::MESSAGE ADDRESS
001334 000000 $MSGLG: .WORD AMSGLG ::MESSAGE LENGTH
001336 $ETABLE: ::APT ENVIRONMENT TABLE
001336 000 $ENV: .BYTE AENV ::ENVIRONMENT BYTE
001337 000 $ENVM: .BYTE AENVM ::ENVIRONMENT MODE BITS
001340 000000 $SWREG: .WORD ASWREG ::APT SWITCH REGISTER
001342 000000 $USWR: .WORD AUSWR ::USER SWITCHES
001344 000000 $CPUOP: .WORD ACPUOP ::CPU TYPE,OPTIONS
*
* BITS 15-11=CPU TYPE
* 11/04=01,11/05=02,11/20=03,11/40=04,11/45=05
* 11/70=06,PDQ=07,Q=10
*
* BIT 10=REAL TIME CLOCK
* BIT 9=FLOATING POINT PROCESSOR
* BIT 8=MEMORY MANAGEMENT
001346 000 $MAMS1: .BYTE AMAMS1 ::HIGH ADDRESS,M.S. BYTE
001347 000 $MTYP1: .BYTE AMTYP1 ::MEM. TYPE,BLK#1
*
* MEM.TYPE BYTE -- (HIGH BYTE)
```



```

          900 NSEC CORE=001
          300 NSEC BIPOLAR=002
          500 NSEC MOS=003
001350 000000 $MADR1: .WORD AMADR1 ::HIGH ADDRESS,BLK#1
          MEM.LAST ADDR.=3 BYTES,THIS WORD AND LOW OF "TYPE" ABOVE
001352 000 $MAMS2: .BYTE AMAMS2 ::HIGH ADDRESS,M.S. BYTE
001353 000 $MTYP2: .BYTE AMTYP2 ::MEM.TYPE,BLK#2
001354 000000 $MADR2: .WORD AMADR2 ::MEM.LAST ADDRESS,BLK#2
001355 000 $MAMS3: .BYTE AMAMS3 ::HIGH ADDRESS,M.S.BYTE
001357 000 $MTYP3: .BYTE AMTYP3 ::MEM.TYPE,BLK#3
001360 000000 $MADR3: .WORD AMADR3 ::MEM.LAST ADDRESS,BLK#3
001362 000 $MAMS4: .BYTE AMAMS4 ::HIGH ADDRESS,M.S.BYTE
001363 000 $MTYP4: .BYTE AMTYP4 ::MEM.TYPE,BLK#4
001364 000000 $MADR4: .WORD AMADR4 ::MEM.LAST ADDRESS,BLK#4
001366 000000 $VECT1: .WORD AVECT1 ::INTERRUPT VECTOR#1,BUS PRIORITY#1
001370 000000 $VECT2: .WORD AVECT2 ::INTERRUPT VECTOR#2BUS PRIORITY#2
001372 000000 $BASE: .WORD ABASE ::BASE ADDRESS OF EQUIPMENT UNDER TEST
001374 000000 $DEVM: .WORD ADEVM ::DEVICE MAP
001376 000000 $CDW1: .WORD ACDW1 ::CONTROLLER DESCRIPTION WORD#1
001400 000000 $CDW2: .WORD ACDW2 ::CONTROLLER DESCRIPTION WORD#2
001402 000000 $DDW0: .WORD ADDW0 ::DEVICE DESCRIPTOR WORD#0
001404 000000 $DDW1: .WORD ADDW1 ::DEVICE DESCRIPTOR WORD#1
001406 000000 $DDW2: .WORD ADDW2 ::DEVICE DESCRIPTOR WORD#2
001410 000000 $DDW3: .WORD ADDW3 ::DEVICE DESCRIPTOR WORD#3
001412 000000 $DDW4: .WORD ADDW4 ::DEVICE DESCRIPTOR WORD#4
001414 000000 $DDW5: .WORD ADDW5 ::DEVICE DESCRIPTOR WORD#5
001416 000000 $DDW6: .WORD ADDW6 ::DEVICE DESCRIPTOR WORD#6
001420 000000 $DDW7: .WORD ADDW7 ::DEVICE DESCRIPTOR WORD#7
001422 000000 $DDW8: .WORD ADDW8 ::DEVICE DESCRIPTOR WORD#8
001424 000000 $DDW9: .WORD ADDW9 ::DEVICE DESCRIPTOR WORD#9
001426 000000 $DDW10: .WORD ADDW10 ::DEVICE DESCRIPTOR WORD#10
001430 000000 $DDW11: .WORD ADDW11 ::DEVICE DESCRIPTOR WORD#11
001432 000000 $DDW12: .WORD ADDW12 ::DEVICE DESCRIPTOR WORD#12
001434 000000 $DDW13: .WORD ADDW13 ::DEVICE DESCRIPTOR WORD#13
001436 000000 $DDW14: .WORD ADDW14 ::DEVICE DESCRIPTOR WORD#14
001440 000000 $DDW15: .WORD ADDW15 ::DEVICE DESCRIPTOR WORD#15
001442 SETEND:

```



.SBTTL ERROR POINTER TABLE

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

INDEX	EM	DH	DT	DF	DESCRIPTION
1434 001442	044704	065163	071646	.WORD	EM1,DH1,DT1,DF1 :ERROR ITEM # 1
001452	044741	065253	071670	.WORD	EM2,DH2,DT2,DF2 :ERROR ITEM # 2
001462	045005	065346	071712	.WORD	EM3,DH3,DT3,DF3 :ERROR ITEM # 3
001472	045052	065437	071712	.WORD	EM4,DH4,DT4,DF4 :ERROR ITEM # 4
001502	045112	065533	071734	.WORD	EM5,DH5,DT5,DF5 :ERROR ITEM # 5
001512	045146	065533	071766	.WORD	EM6,DH6,DT6,DF6 :ERROR ITEM # 6
001522	045200	065533	071766	.WORD	EM7,DH7,DT7,DF7 :ERROR ITEM # 7
001532	045112	065533	071766	.WORD	EM10,DH10,DT10,DF10 :ERROR ITEM # 10
001542	045233	065533	071766	.WORD	EM11,DH11,DT11,DF11 :ERROR ITEM # 11
001552	000000	000000	072010	.WORD	EM12,DH12,DT12,DF12 :ERROR ITEM # 12
001562	000000	000000	072076	.WORD	EM13,DH13,DT13,DF13 :ERROR ITEM # 13
001572	045314	065533	071766	.WORD	EM14,DH14,DT14,DF14 :ERROR ITEM # 14
001602	045437	065533	071766	.WORD	EM15,DH15,DT15,DF15 :ERROR ITEM # 15
001612	045562	065573	072130	.WORD	EM16,DH16,DT16,DF16 :ERROR ITEM # 16
001622	045633	065653	071712	.WORD	EM17,DH17,DT17,DF17 :ERROR ITEM # 17
001632	046066	065743	072150	.WORD	EM20,DH20,DT20,DF20 :ERROR ITEM # 20
001642	046244	065533	072172	.WORD	EM21,DH21,DT21,DF21 :ERROR ITEM # 21
001652	046375	066031	072204	.WORD	EM22,DH22,DT22,DF22 :ERROR ITEM # 22
001662	046375	066066	072232	.WORD	EM23,DH23,DT23,DF23 :ERROR ITEM # 23
001672	046375	066224	072254	.WORD	EM24,DH24,DT24,DF24 :ERROR ITEM # 24
001702	046462	066363	072300	.WORD	EM25,DH25,DT25,DF25 :ERROR ITEM # 25
001712	046575	066425	072350	.WORD	EM26,DH26,DT26,DF26 :ERROR ITEM # 26
001722	046575	066425	072424	.WORD	EM27,DH27,DT27,DF27 :ERROR ITEM # 27
001732	046643	000000	072466	.WORD	EM30,DH30,DT30,DF30 :ERROR ITEM # 30
001742	046715	066425	072350	.WORD	EM31,DH31,DT31,DF31 :ERROR ITEM # 31
001752	046715	066425	072424	.WORD	EM32,DH32,DT32,DF32 :ERROR ITEM # 32
001762	046763	066513	072520	.WORD	EM33,DH33,DT33,DF33 :ERROR ITEM # 33
001772	047024	066513	072576	.WORD	EM34,DH34,DT34,DF34 :ERROR ITEM # 34
002002	047126	066513	072576	.WORD	EM35,DH35,DT35,DF35 :ERROR ITEM # 35
002012	047230	066513	072576	.WORD	EM36,DH36,DT36,DF36 :ERROR ITEM # 36
002022	047331	066513	072576	.WORD	EM37,DH37,DT37,DF37 :ERROR ITEM # 37
002032	047432	066513	072520	.WORD	EM40,DH40,DT40,DF40 :ERROR ITEM # 40
002042	047603	000000	072650	.WORD	EM41,DH41,DT41,DF41 :ERROR ITEM # 41
002052	047640	066616	072702	.WORD	EM42,DH42,DT42,DF42 :ERROR ITEM # 42
002062	047761	066616	072702	.WORD	EM43,DH43,DT43,DF43 :ERROR ITEM # 43
002072	050102	000000	072760	.WORD	EM44,DH44,DT44,DF44 :ERROR ITEM # 44
002102	050102	066720	073030	.WORD	EM45,DH45,DT45,DF45 :ERROR ITEM # 45
002112	050145	066737	073104	.WORD	EM46,DH46,DT46,DF46 :ERROR ITEM # 46
002122	050223	066720	073172	.WORD	EM47,DH47,DT47,DF47 :ERROR ITEM # 47
002132	050341	066763	072576	.WORD	EM50,DH50,DT50,DF50 :ERROR ITEM # 50
002142	050437	066763	073224	.WORD	EM51,DH51,DT51,DF51 :ERROR ITEM # 51
002152	050500	065533	073172	.WORD	EM52,DH52,DT52,DF52 :ERROR ITEM # 52
002162	050621	066425	073262	.WORD	EM53,DH53,DT53,DF53 :ERROR ITEM # 53
002172	051016	067035	073302	.WORD	EM54,DH54,DT54,DF54 :ERROR ITEM # 54
002202	051062	065533	073172	.WORD	EM55,DH55,DT55,DF55 :ERROR ITEM # 55
002212	051203	066425	073262	.WORD	EM56,DH56,DT56,DF56 :ERROR ITEM # 56



002222	051400	067035	073302	.WORD	EM57,DH57,DT57,DF57	:ERROR	ITEM # 57
002232	051444	066425	073262	.WORD	EM60,DH60,DT60,DF60	:ERROR	ITEM # 60
002242	051641	067035	073302	.WORD	EM61,DH61,DT61,DF61	:ERROR	ITEM # 61
002252	051705	067035	073302	.WORD	EM62,DH62,DT62,DF62	:ERROR	ITEM # 62
002262	052077	067035	073302	.WORD	EM63,DH63,DT63,DF63	:ERROR	ITEM # 63
002272	052271	067145	073340	.WORD	EM64,DH64,DT64,DF64	:ERROR	ITEM # 64
002302	052271	067076	073340	.WORD	EM65,DH65,DT65,DF65	:ERROR	ITEM # 65
002312	052425	067035	073302	.WORD	EM66,DH66,DT66,DF66	:ERROR	ITEM # 66
002322	052470	065533	072172	.WORD	EM67,DH67,DT67,DF67	:ERROR	ITEM # 67
002332	052721	065533	073360	.WORD	EM70,DH70,DT70,DF70	:ERROR	ITEM # 70
002342	053044	066363	073360	.WORD	EM71,DH71,DT71,DF71	:ERROR	ITEM # 71
002352	053146	066425	073426	.WORD	EM72,DH72,DT72,DF72	:ERROR	ITEM # 72
002362	053222	067035	073302	.WORD	EM73,DH73,DT73,DF73	:ERROR	ITEM # 73
002372	053262	065533	072172	.WORD	EM74,DH74,DT74,DF74	:ERROR	ITEM # 74
002402	053513	065533	073360	.WORD	EM75,DH75,DT75,DF75	:ERROR	ITEM # 75
002412	053636	066363	073360	.WORD	EM76,DH76,DT76,DF76	:ERROR	ITEM # 76
002422	053740	066425	073426	.WORD	EM77,DH77,DT77,DF77	:ERROR	ITEM # 77
002432	054014	067035	073302	.WORD	EM100,DH100,DT100,DF100	:ERROR	ITEM # 100
002442	054054	065533	073360	.WORD	EM101,DH101,DT101,DF101	:ERROR	ITEM # 101
002452	054200	066425	073360	.WORD	EM102,DH102,DT102,DF102	:ERROR	ITEM # 102
002462	054252	066363	073360	.WORD	EM103,DH103,DT103,DF103	:ERROR	ITEM # 103
002472	054355	067035	073302	.WORD	EM104,DH104,DT104,DF104	:ERROR	ITEM # 104
002502	054416	065533	073360	.WORD	EM105,DH105,DT105,DF105	:ERROR	ITEM # 105
002512	054543	066425	073426	.WORD	EM106,DH106,DT106,DF106	:ERROR	ITEM # 106
002522	054616	066363	073360	.WORD	EM107,DH107,DT107,DF107	:ERROR	ITEM # 107
002532	054722	067035	073302	.WORD	EM110,DH110,DT110,DF110	:ERROR	ITEM # 110
002542	054764	066363	073446	.WORD	EM111,DH111,DT111,DF111	:ERROR	ITEM # 111
002552	054764	067233	073446	.WORD	EM112,DH112,DT112,DF112	:ERROR	ITEM # 112
002562	055066	066363	073446	.WORD	EM113,DH113,DT113,DF113	:ERROR	ITEM # 113
002572	055066	067233	073446	.WORD	EM114,DH114,DT114,DF114	:ERROR	ITEM # 114
002602	054764	067452	073446	.WORD	EM115,DH115,DT115,DF115	:ERROR	ITEM # 115
002612	055066	067452	073446	.WORD	EM116,DH116,DT116,DF116	:ERROR	ITEM # 116
002622	055170	065653	071712	.WORD	EM117,DH117,DT117,DF117	:ERROR	ITEM # 117
002632	055324	067736	071712	.WORD	EM120,DH120,DT120,DF120	:ERROR	ITEM # 120
002642	055460	065533	073172	.WORD	EM121,DH121,DT121,DF121	:ERROR	ITEM # 121
002652	055577	066763	072576	.WORD	EM122,DH122,DT122,DF122	:ERROR	ITEM # 122
002662	055676	066763	073224	.WORD	EM123,DH123,DT123,DF123	:ERROR	ITEM # 123
002672	055737	065653	073460	.WORD	EM124,DH124,DT124,DF124	:ERROR	ITEM # 124
002702	056032	065653	073460	.WORD	EM125,DH125,DT125,DF125	:ERROR	ITEM # 125
002712	056122	065533	073446	.WORD	EM126,DH126,DT126,DF126	:ERROR	ITEM # 126
002722	056331	067035	073446	.WORD	EM127,DH127,DT127,DF127	:ERROR	ITEM # 127
002732	056544	067736	071712	.WORD	EM130,DH130,DT130,DF130	:ERROR	ITEM # 130
002742	056644	067035	073544	.WORD	EM131,DH131,DT131,DF131	:ERROR	ITEM # 131
002752	056704	067035	073544	.WORD	EM132,DH132,DT132,DF132	:ERROR	ITEM # 132
002762	056744	070026	073606	.WORD	EM133,DH133,DT133,DF133	:ERROR	ITEM # 133
002772	057003	070026	073606	.WORD	EM134,DH134,DT134,DF134	:ERROR	ITEM # 134
003002	057042	070026	073606	.WORD	EM135,DH135,DT135,DF135	:ERROR	ITEM # 135
003012	057101	070026	073606	.WORD	EM136,DH136,DT136,DF136	:ERROR	ITEM # 136
003022	056744	070136	073660	.WORD	EM137,DH137,DT137,DF137	:ERROR	ITEM # 137
003032	057003	070136	073660	.WORD	EM140,DH140,DT140,DF140	:ERROR	ITEM # 140
003042	057042	070136	073660	.WORD	EM141,DH141,DT141,DF141	:ERROR	ITEM # 141
003052	057101	070136	073660	.WORD	EM142,DH142,DT142,DF142	:ERROR	ITEM # 142
003062	057140	070026	073606	.WORD	EM143,DH143,DT143,DF143	:ERROR	ITEM # 143
003072	057173	070026	073606	.WORD	EM144,DH144,DT144,DF144	:ERROR	ITEM # 144
003102	057140	070136	073660	.WORD	EM145,DH145,DT145,DF145	:ERROR	ITEM # 145
003112	057173	070136	073660	.WORD	EM146,DH146,DT146,DF146	:ERROR	ITEM # 146
003122	057226	067035	073606	.WORD	EM147,DH147,DT147,DF147	:ERROR	ITEM # 147



003132	057226	070326	073606	.WORD	EM150,DH150,DT150,DF150	:ERROR ITEM # 150
003142	057226	070136	073660	.WORD	EM151,DH151,DT151,DF151	:ERROR ITEM # 151
003152	057260	070026	073606	.WORD	EM152,DH152,DT152,DF152	:ERROR ITEM # 152
003162	057260	070136	073660	.WORD	EM153,DH153,DT153,DF153	:ERROR ITEM # 153
003172	057312	070417	073700	.WORD	EM154,DH154,DT154,DF154	:ERROR ITEM # 154
003202	057544	070417	073700	.WORD	EM155,DH155,DT155,DF155	:ERROR ITEM # 155
003212	057777	067035	073606	.WORD	EM156,DH156,DT156,DF156	:ERROR ITEM # 156
003222	060214	067035	073606	.WORD	EM157,DH157,DT157,DF157	:ERROR ITEM # 157
003232	060433	067035	073606	.WORD	EM160,DH160,DT160,DF160	:ERROR ITEM # 160
003242	060640	067035	073606	.WORD	EM161,DH161,DT161,DF161	:ERROR ITEM # 161
003252	061045	067035	073606	.WORD	EM162,DH162,DT162,DF162	:ERROR ITEM # 162
003262	061112	067035	073606	.WORD	EM163,DH163,DT163,DF163	:ERROR ITEM # 163
003272	061157	065653	071712	.WORD	EM164,DH164,DT164,DF164	:ERROR ITEM # 164
003302	061224	065653	071712	.WORD	EM165,DH165,DT165,DF165	:ERROR ITEM # 165
003312	061271	067035	073606	.WORD	EM166,DH166,DT166,DF166	:ERROR ITEM # 166
003322	061401	067035	073606	.WORD	EM167,DH167,DT167,DF167	:ERROR ITEM # 167
003332	061640	067035	073606	.WORD	EM170,DH170,DT170,DF170	:ERROR ITEM # 170
003342	061750	067035	073606	.WORD	EM171,DH171,DT171,DF171	:ERROR ITEM # 171
003352	062207	067035	073606	.WORD	EM172,DH172,DT172,DF172	:ERROR ITEM # 172
003362	062446	067035	073606	.WORD	EM173,DH173,DT173,DF173	:ERROR ITEM # 173
003372	062705	067035	073606	.WORD	EM174,DH174,DT174,DF174	:ERROR ITEM # 174
003402	063144	067035	073606	.WORD	EM175,DH175,DT175,DF175	:ERROR ITEM # 175
003412	063403	067035	073606	.WORD	EM176,DH176,DT176,DF176	:ERROR ITEM # 176
003422	063540	070457	073712	.WORD	EM177,DH177,DT177,DF177	:ERROR ITEM # 177
003432	063574	067035	073606	.WORD	EM200,DH200,DT200,DF200	:ERROR ITEM # 200
003442	063731	067035	073606	.WORD	EM201,DH201,DT201,DF201	:ERROR ITEM # 201
003452	064066	067035	073606	.WORD	EM202,DH202,DT202,DF202	:ERROR ITEM # 202
003462	064223	067035	073606	.WORD	EM203,DH203,DT203,DF203	:ERROR ITEM # 203
003472	064360	067035	073606	.WORD	EM204,DH204,DT204,DF204	:ERROR ITEM # 204
003502	064515	065653	071712	.WORD	EM205,DH205,DT205,DF205	:ERROR ITEM # 205
003512	064562	067035	073606	.WORD	EM206,DH206,DT206,DF206	:ERROR ITEM # 206
003522	064627	067035	073606	.WORD	EM207,DH207,DT207,DF207	:ERROR ITEM # 207
003532	064751	067035	073606	.WORD	EM210,DH210,DT210,DF210	:ERROR ITEM # 210
003542	045112	070523	073722	.WORD	EM211,DH211,DT211,DF211	:ERROR ITEM # 211
003552	045146	065533	073740	.WORD	EM212,DH212,DT212,DF212	:ERROR ITEM # 212
003562	045200	065533	073740	.WORD	EM213,DH213,DT213,DF213	:ERROR ITEM # 213
003572	065026	067035	073606	.WORD	EM214,DH214,DT214,DF214	:ERROR ITEM # 214



1438

000046 003602  
000052 000046  
036110  
000052  
000000  
003602

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



1440

000024 003602  
000024 000074  
000044 000200  
000044 000044  
000044 003602  
000044 003602

003602  
003602 000000  
003604 001316  
003606 000010  
003610 000040  
003612 000000  
003614 000052

```
.SBTTL APT PARAMETER BLOCK
:*****
:SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
:*****
      .SX=.      ;;SAVE CURRENT LOCATION
      =24      ;;SET POWER FAIL TO POINT TO START OF PROGRAM
      200      ;;FOR APT START UP
      =44      ;;POINT TO APT INDIRECT ADDRESS PNTR.
$APTHDR ;;POINT TO APT HEADER BLOCK
      =.SX      ;;RESET LOCATION COUNTER
:*****
:SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
:INTERFACE SPEC.
$APTHD:
$HIBTS: .WORD 0      ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
$MBADR: .WORD $MAIL  ;;ADDRESS OF APT MAILBOX (BITS 0-15)
$TSTM:  .WORD 10     ;;RUN TIM OF LONGEST TEST
$PASTM: .WORD 40     ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
$UNITM: .WORD 0      ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
      .WORD $ETEND-$MAIL/2 ;;LENGTH MAILBOX-ETABLE(WORDS)
```



1442  
1443 003616

003616 012706 001100  
003622 005026  
003624 022706 001140  
003630 001374  
003632 012706 001100  
  
003636 012737 036170 000020  
003644 012737 000340 000022  
003652 012737 036506 000030  
003660 012737 000340 000032  
003666 012737 041072 000034  
003674 012737 000340 000036  
003702 012737 041156 000024  
003710 012737 000340 000026  
003716 013737 035736 035730  
003724 005037 001302  
003730 005037 001304  
003734 112737 000001 001115  
  
003742 012737 036154 000014  
003750 012737 000340 000016  
003756 012737 000002 036154  
003764 012737 004012 000010  
003772 005046  
003774 012746 004002  
004000 000006  
004002 012737 000006 036154 64\$:  
004010 000402  
004012 062706 000010 65\$:  
004016 012737 000012 000010 66\$:  
004024 005037 036162  
004030 012737 004030 001106  
004036 012737 004036 001110  
  
004044 013746 000004  
004050 012737 004104 000004  
004056 012737 177570 001140  
004064 012737 177570 001142  
004072 022777 177777 175040  
004100 001012  
  
004102 000403  
004104 012716 004112 67\$:  
004110 000002  
004112 012737 000176 001140 68\$:  
004120 012737 000174 001142  
004126 012637 000004 69\$:  
004132 005037 001324  
004136 132737 000200 001337  
004144 001403  
004146 012737 001340 001140

```
.SBTTL INITIALIZE THE COMMON TAGS
START:
.SBTTL INITIALIZE THE COMMON TAGS
::CLEAR THE COMMON TAGS ($CMTAG) AREA
MOV    #CMTAG,R6      ;;FIRST LOCATION TO BE CLEARED
CLR    (R6)+          ;;CLEAR MEMORY LOCATION
CMP    #SWR,R6      ;;DONE?
BNE    -6             ;;LOOP BACK IF NO
MOV    #STACK,SP     ;;SETUP THE STACK POINTER
::INITIALIZE A FEW VECTORS
MOV    #SCOPE,@#IOTVEC ;;IOT VECTOR FOR SCOPE ROUTINE
MOV    #340,@#IOTVEC+2 ;;LEVEL 7
MOV    #ERROR,@#EMTVEC ;;EMT VECTOR FOR ERROR ROUTINE
MOV    #340,@#EMTVEC+2 ;;LEVEL 7
MOV    #TRAP,@#TRAPVEC ;;TRAP VECTOR FOR TRAP CALLS
MOV    #340,@#TRAPVEC+2;LEVEL 7
MOV    #PWRDN,@#PWRVEC ;;POWER FAILURE VECTOR
MOV    #340,@#PWRVEC+2 ;;LEVEL 7
MOV    $ENDCT,$EOPCT  ;;SETUP END-OF-PROGRAM COUNTER
CLR    $TIMES         ;;INITIALIZE NUMBER OF ITERATIONS
CLR    $ESCAPE        ;;CLEAR THE ESCAPE ON ERROR ADDRESS
MOVB   #1,$SERMAX     ;;ALLOW ONE ERROR PER TEST
::INITIALIZE THE 'T-BIT' TRAP VECTOR. THEN LOAD LOCATION '$RTRN', IN
::THE 'END-OF-PASS' ($EOP) ROUTINE, WITH A 'RTI' OR 'RTT'.
MOV    #RTRN,@#TBITVEC ;;SET 'T' BIT VECTOR TO $RTRN
MOV    #340,@#TBITVEC+2 ;;LEVEL 7
MOV    #RTI,$RTRN      ;;SET $RTRN TO A RTI
MOV    #65$,@#RESVEC   ;;TRY TO DO A RTT
CLR    -(SP)          ;;DUMMY PS
MOV    #64$,-(SP)     ;;AND PC
RTT    ;;TRY THE RTT
MOV    #RTT,$RTRN     ;;RTT IS LEGAL--SET $RTRN TO A RTT
BR     66$
ADD    #10,SP         ;;RTT ILLEGAL--CLEAN OFF THE STACK
MOV    #RESVEC+2,@#RESVEC ;;RESTORE TRAP CATCHER
CLR    $TBIT         ;;CLEAR 'T' BIT SWITCH
MOV    #,$LPADR       ;;INITIALIZE THE LOOP ADDRESS FOR SCOPE
MOV    #,$LPERR       ;;SETUP THE ERROR LOOP ADDRESS
::SIZE FOR ^ HARDWARE SWITCH REGISTER. IF NOT FOUND OR IT IS
::EQUAL TO A '-1', SETUP FOR A SOFTWARE SWITCH REGISTER.
MOV    @#ERRVEC,-(SP) ;;SAVE ERROR VECTOR
MOV    #67$,@#ERRVEC  ;;SET UP ERROR VECTOR
MOV    #DSWR,SWR      ;;SETUP FOR A HARDWARE SWICH REGISTER
MOV    #DDISP,DISPLAY ;;AND A HARDWARE DISPLAY REGISTER
CMP    #-1,@SWR       ;;TRY TO REFERENCE HARDWARE SWR
BNE    69$           ;;BRANCH IF NO TIMEOUT TRAP OCCURRED
;;AND THE HARDWARE SWR IS NOT = -1
BR     68$           ;;BRANCH IF NO TIMEOUT
MOV    #68$,(SP)     ;;SET UP FOR TRAP RETURN
MOV    #SWREG,SWR    ;;POINT TO SOFTWARE SWR
MOV    #DISPREG,DISPLAY
MOV    (SP)+,@#ERRVEC ;;RESTORE ERROR VECTOR
CLR    $PASS         ;;CLEAR PASS COUNT
BITB   #APTSIZE,$ENVM ;;TEST USER SIZE UNDER APT
BEQ    70$           ;;YES,USE NON-APT SWITCH
MOV    #SSWREG,SWR   ;;NO,USE APT SWITCH REGISTER
```



```

1444 004154 005227 177777 70$:
      004160 001047          .SBTTL TYPE PROGRAM NAME
      004162 022737 036110 000042  ::TYPE THE NAME OF THE PROGRAM IF FIRST PASS
      004170 001443          INC #-1 ::FIRST TIME?
      004172 104401 004240      BNE 71$ ::BRANCH IF NO
      004176 005737 000042      CMP #SENDAD,@#42 ::ACT-11?
      004202 001012          BEQ 71$ ::BRANCH IF YES
      004204 123727 001336 000001  TYPE ,72$ ::TYPE ASCIZ STRING
      004212 001406          .SBTTL GET VALUE FOR SOFTWARE SWITCH REGISTER
      004214 023727 001140 000176  TST @#42 ::ARE WE RUNNING UNDER XXDP/ACT?
      004222 001005          BNE 73$ ::BRANCH IF YES
      004224 104406          CMPB $ENV,#1 ::ARE WE RUNNING UNDER APT?
      004226 000403          BEQ 73$ ::BRANCH IF YES
      004230 112737 000001 001134  CMP SWR,#SWREG ::SOFTWARE SWITCH REG SELECTED?
      004236 000420          BNE 74$ ::BRANCH IF NO
      004300 004300          GTSWR ::GET SOFT-SWR SETTINGS
      1445 004300          BR 74$
      73$: MOVB #1,$AUTOB ::SET AUTO-MODE INDICATOR
      74$: BR 71$ ::GET OVER THE ASCIZ
      ::72$: .ASCIZ <CRLF>*CKFPACO FP11F FLTG PNT PRT A*<CRLF>
      71$:
      LOOP:

```



1464

```

.SBTTL TEST # 1 - LDFPS, STFPS AND DATA PATHS TEST
:*****
:*TEST 1 LDFPS, STFPS AND DATA PATHS TEST
:*
:*THIS IS A TEST OF THE LDFPS (LOAD FLOATING POINT STATUS) AND STFPS
:*(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 DMO AND
:*SMO 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.
    
```

```

004300 000004
1465 004302 005037 004560
1466 004306 012737 004350 001110
1467 004314 012700 177777
1468 004320 012737 004562 000244
1469 004326 012737 004574 000010
1470 004334 005002
1471 004336 005102
1472 004340 005003
1473 004342 012737 004626 000004
1474
1475
1476
1477 004350
1478 004350 010004
1479 004352 042704 030020
1480 004356 170104
1481
1482 004360 012701 177777
1483 004364 170201
1484 004366 012737 042066 000244
1485 004374 010004
1486 004376 042704 030020
1487 004402 012737 042120 000004
1488 004410 012737 042136 000010
1489 004416 020401
1490
1491 004420 001002
    
```

```

TST1: SCOPE
      CLR AERFLG
      MOV #A1,$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
      MOV #-1,R0 ;INITIALIZE THE COUNT PATTERN.
      MOV #AERR1,FPVECT ;SET UP FOR UNABLE TO DECODE
      MOV #AERR2,10 ;FPP INSTRUCTION TRAP TO 244 OR 10.
      CLR R2 ;R2 IS THE 'AND' OF BAD DATA.
      COM R2
      CLR R3 ;R3 IS THE 'OR' OF BAD DATA.
      MOV #AERR3,ERRVECT ;IF EITHER INSTP'CTION
      ;FAILS TO GO THROUGH THE
      ;CORRECT SRC OR DST MODE AN
      ;ODD ADDRESS TRAP WILL OCCUR.

A1:
A11: MOV R0,R4
      BIC #30020,R4
      LDFPS R4 ;TEST INSTRUCTION.

A12: MOV #-1,R1
      STFPS R1 ;TEST INSTRUCTION.
      MOV #FPSPUR,FPVECT ;SET UP FOR UNEXPECTED TRAPS.
      MOV R0,R4 ;MASK OFF UNSETTABLE BITS.
      BIC #30020,R4
      MOV #CPSPUR,ERRVECT
      MOV #CPTWO,10
      CMP R4,R1 ;COMPARE DATA EXPECTED WITH
      ;THE DATA READ.
      BNE A3 ;IF NOT EQUAL GO REPORT ERROR.
    
```



```

1492
1493 004422 077026          A2:   SOB   R0,A1      ;OTHERWISE DECREMENT COUNT PATTERN
1494 004424 000425          BR    A5      ;UNTIL IT IS ZERO.
1495
1496 004426 005237 004560   A3:   INC   AERFLG   ;RECORD ERROR.
1497 004432 050003          BIS   R0,R3    ;COMPUTE 'OR' OF FAILING PATTERNS.
1498 004434 010005          MOV   R0,R5    ;COMPUTE 'AND' OF FAILING PATTERNS.
1499 004436 005105          COM   R5
1500 004440 040502          BIC   R5,R2
1501
1502 004442 022737 000005 004560   CMP   #5,AERFLG ;SEE IF MORE THAN 5 ERRORS HAVE
1503 004450 103412          BLO  A05      ;OCCURRED. BR IF YES.
1504
1505
1506 004452 012737 004350 001236   MOV   #A1,$TMP2
1507 004460 010037 001240          MOV   R0,$TMP3
1508 004464 010137 001242          MOV   R1,$TMP4
1509 004470 010437 001244          MOV   R4,$TMP5
1510 004474 104001          A4:   ERROR  +1
1511
1512 004476 000751          A05:  BR    A2      ;CONTINUE TESTING.
1513
1514 004500 005737 004560   A5:   TST   AERFLG   ;SEE IF ANY ERRORS OCCURRED.
1515 004504 001471          BEQ  ADONE     ;IF NOT GO TO NEXT TEST.
1516 004506 032777 020000 174424   BIT   #SW13,@SWR ;OTHERWISE SEE IF A SUMMARY
1517 004514 001404          BEQ  A6        ;SHOULD BE TYPED.
1518 004516 032777 000200 174414   BIT   #SW7,@SWR
1519 004524 001461          BEQ  ADONE
1520
1521 004526          A6:
1522 004526 010237 001236   MOV   R2,$TMP2
1523 004532 010337 001240   MOV   R3,$TMP3
1524 004536 012737 004552 001116   MOV   #A7,$ERRPC
1525 004544 112737 000002 001114   MOVB  #2,$ITEMB
1526 004552 004737 041342   A7:   JSR   PC,ERTYPE
1527 004556 000444          BR    ADONE
1528
1529 004560 000000   AERFLG: .WORD 0
1530
1531          ;UNABLE TO DECODE FPP INSTRUCTION. TRAPPED TO 244.
1532 004562 011637 001236   AERR1: MOV   (SP),$TMP2      ;SAVE PC OF TRAP.
1533 004566 022626          CMP   (SP)+,(SP)+
1534 004570 104010          1$:   ERROR +10
1535 004572 000436          BR    ADONE
1536
1537          ;UNABLE TO DECODE INSTRUCTION. TRAPPED TO 10.
1538 004574 021627 004352   AERR2: CMP   (SP),#A11+2    ;DID TRAP OCCUR OF FPP INSTRUCTION?
1539 004600 001405          BEQ  1$
1540 004602 021627 004366   CMP   (SP),#A12+2
1541 004606 001402          BEQ  1$
1542 004610 000137 042136   JMP   CPTWO
1543
1544          ;IF NOT FPP INSTRUCTION THEN
1545 004614 011637 001236   1$:   MOV   (SP),$TMP2      ;REPORT SPURIOUS TRAP TO 10.
1546 004620 022626          CMP   (SP)+,(SP)+
1547 004622 104011          2$:   ERROR +11
1548 004624 000421          BR    ADONE

```



1549  
 1550  
 1551 004626 021627 004352  
 1552 004632 001405  
 1553 004634 021627 004366  
 1554 004640 001407  
 1555 004642 000137 042120  
 1556  
 1557  
 1558 004646 011637 001236  
 1559 004652 022626  
 1560 004654 104014  
 1561 004656 000404  
 1562  
 1563 004660 011637 001236  
 1564 004664 022626  
 1565 004666 104015  
 1566  
 1567 004670  
 C04670 104413

:TRAP TO 4 HANDLER:  
 AERR3: CMP (SP),#A11+2  
 BEQ 1\$  
 CMP (SP),#A12+2  
 BEQ 2\$  
 JMP CPSPUR  
  
 1\$: MOV (SP), \$TMP2  
 CMP (SP)+, (SP)+  
 15\$: ERROR +14  
 BR ADONE  
  
 2\$: MOV (SP), \$TMP2  
 CMP (SP)+, (SP)+  
 25\$: ERROR +15  
  
 ADONE: RSETUP

:DID THE TRAP OCCUR ON THE  
 :LDFPS INSTRUCTION?  
 :OR THE STFPS INSTRUCTION?  
  
 :IF NEITHER THEN REPORT  
 :UNEXPECTED TRAP TO 4.

: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?).

1568  
 1569



1575

```
.SBTTL TEST # 2 - CFCC TEST  
:*****  
:TEST 2 CFCC TEST  
:*****  
:THIS IS A TEST OF THE COPY CONDITION CODES INSTRUCTION, CFCC.  
:*****
```

```
1576 004672 000004  
1577 004672 012737 004706 001110  
1578 004702 012700 000017  
1579 004706  
1580 004706 170100  
1581  
1582 004710  
1583 004710 170000  
1584  
1585 004712 013703 177776  
1586 004716 042703 177760  
1587 004722 020003  
1588 004724 001002  
1589  
1590 004726 077011  
1591 004730 000422  
1592  
1593 004732  
1594 004732 170201  
1595 004734 012737 004710 001236  
1596 004742 020001  
1597 004744 001006  
1598  
1599 004746 010337 001240  
1600 004752 010037 001242  
1601 004756 104003  
1602 004760 000762  
1603  
1604 004762  
1605 004762 010037 001240  
1606 004766 010137 001242  
1607 004772 104004  
1608 004774 000754  
1609  
1610 004776  
004776 104413
```

```
TST2: SCOPE  
MOV #B1,$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.  
MOV #17,R0 ;R0 CONTAINS TO TEST PATTERN.  
  
B1: LDFPS R0 ;LOAD THE TEST PATTERN  
  
B2: CFCC ;COPY CONDITION CODES.  
MOV PSW,R3 ;SEE IF PATTERN TRANSFERED.  
BIC #177760,R3  
CMP R0,R3  
BNE BERR  
  
B3: SOB R0,B1  
BR BDONE  
  
BERR: STFPS R1 ;WAS FPS MODIFIED BY CFCC?  
MOV #B2,$TMP2  
CMP R0,R1  
BNE BERR1  
  
B4: MOV R3,$TMP3  
MOV R0,$TMP4  
ERROR +3  
BR B3  
  
BERR1: MOV R0,$TMP3  
MOV R1,$TMP4  
B5: ERROR +4  
BR B3  
  
BDONE: 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?).
```

1611







```

1665
1666 005202          1$:
      005202 012737 005210 001110 C4:  MOV   #C4,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
1667 005210          CLR   RO
1668 005212 170100   LDFPS RO                ;CLEAR FPS.
1669 005214 012737 005222 001236 MOV   #C45,$TMP2
1670
1671 005222 170011   C45: SETD                ;SET FD BIT.
1672
1673 005224 170201   STFPS R1                ;GET RESULT.
1674 005226 012702 000200 MOV   #200,R2
1675 005232 020102   CMP   R1,R2            ;RESULT CORRECT?
1676 005234 001402   BEQ   1$
1677 005236 004737 005476 JSR   PC,CERR1
1678
1679 005242          1$:
      005242 012737 005250 001110 C5:  MOV   #C5,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
1680 005250 012737 000204 001250 MOV   #204,$TMP7
1681 005256 012737 043245 001252 MOV   #SETI1,$TMP10
1682 005264 005000   CLR   RO
1683
1684 005266 170100   LDFPS RO                ;CLEAR FPS
1685 005270 012737 005276 001236 MOV   #C55,$TMP2
1686
1687 005276 170002   C55: SETI                ;CLEAR FL BIT.
1688
1689 005300 170201   STFPS R1                ;GET RESULT.
1690 005302 005002   CLR   R2
1691 005304 020201   CMP   R2,R1            ;RESULT CORRECT?
1692 005306 001402   BEQ   1$
1693 005310 004737 005476 JSR   PC,CERR1
1694
1695 005314          1$:
      005314 012737 005322 001110 C6:  MOV   #C6,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
1696 005322 012700 147757          MOV   #147757,RO
1697 005326 170100   LDFPS RO                ;PUT 147757 INTO FPS
1698 005330 012737 005336 001236 MOV   #C65,$TMP2
1699
1700 005336 170002   C65: SETI                ;CLEAR FL BIT.
1701
1702 005340 170201   STFPS R1                ;GET THE RESULT.
1703 005342 012702 147657          MOV   #147657,R2
1704 005346 020102   CMP   R1,R2            ;RESULT CORRECT?
1705 005350 001402   BEQ   1$
1706 005352 004737 005574 JSR   PC,CERR2
1707
1708 005356          1$:
      005356 012737 005364 001110 C7:  MOV   #C7,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
1709 005364 012737 000205 001250 MOV   #205,$TMP7
1710 005372 012737 043253 001252 MOV   #SETL1,$TMP10
1711 005400 012700 147757          MOV   #147757,RO
1712 005404 170100   LDFPS RO                ;SET FPS TO 147757.
1713 005406 012737 005414 001236 MOV   #C75,$TMP2
1714
1715 005414 170012   C75: SETL                ;SET FL BIT.
1716
1717 005416 170201   STFPS R1                ;GET THE RESULT.
    
```



```

1718 005420 012702 147757          MOV    #147757,R2
1719 005424 020102          CMP    R1,R2                ;RESULT CORRECT?
1720 005426 001402          BEQ    1$
1721 005430 004737 005574          JSR    PC,CERR2
1722
1723 005434          1$:
      005434 012737 005442 001110      MOV    #C8,$LPERR          ;SET UP THE LOOP ON ERROR ADDRESS.
1724 005442 005000          C8:    CLR    R0
1725 005444 170100          LDFPS R0                    ;CLEAR FPS.
1726 005446 012737 005454 001236      MOV    #C85,$TMP2
1727
1728 005454 170012          C85:   SETL                   ;SET FL BIT.
1729
1730 005456 170201          STFPS R1
1731 005460 012702 000100      MOV    #100,R2
1732 005464 020102          CMP    R1,R2                ;RESULT CORRECT.
1733 005466 001402          BEQ    1$
1734 005470 004737 005476          JSR    PC,CERR1
1735
1736 005474 000522          1$:    BR     CDONE
1737
1738          ;THESE ARE ERROR ANALYSIS ROUTINES:
1739 005476 010103          CERR1: MOV    R1,R3
1740 005500 032703 177477          BIT    #177477,R3          ;ARE ANY OTHER BITS SET?
1741 005504 001401          BEQ    2$
1742 005506 000503          1$:    BR     CERR4
1743
1744 005510 022703 000300          2$:    CMP    #300,R3        ;ARE BOTH FD AND FL SET?
1745 005514 001774          BEQ    1$
1746 005516 032703 000300          BIT    #300,R3            ;ARE THEY BOTH CLEAR?
1747 005522 001771          BEQ    1$
1748
1749 005524 032703 000200          BIT    #200,R3            ;IS FD SET?
1750 005530 001407          BEQ    3$
1751 005532 012737 043237 001254      MOV    #SETD1,$TMP11
1752 005540 012737 000203 001246      MOV    #203,$TMP6
1753 005546 000452          BR     CERR3
1754
1755 005550 032703 000100          3$:    BIT    #100,R3        ;IS FL SET
1756 005554 001754          BEQ    1$
1757 005556 012737 043253 001254      MOV    #SETL1,$TMP11
1758 005564 012737 000205 001246      MOV    #205,$TMP6
1759 005572 000440          BR     CERR3
1760
1761 005574 010103          CERR2: MOV    R1,R3
1762 005576 005103          COM    R3
1763
1764 005600 032703 177477          BIT    #177477,R3          ;ARE ANY OTHER BITS SET?
1765 005604 001401          BEQ    2$
1766 005606 000443          1$:    BR     CERR4
1767
1768 005610 032703 000300          2$:    BIT    #300,R3        ;ARE BOTH FD AND FL SET?
1769 005614 001774          BEQ    1$
1770 005616 032701 000300          BIT    #300,R1            ;ARE THEY BOTH CLEAR?
1771 005622 001771          BEQ    1$
1772
1773 005624 032701 000200          BIT    #200,R1            ;IS FD CLEAR?
    
```



```

1774 005630 001007          BNE      3$
1775 005632 012737 043231 001254    MOV     #SETF1,$TMP11
1776 005640 012737 000202 001246    MOV     #202,$TMP6
1777 005646 000412          BR      CERR3
1778
1779 005650 032701 000100    3$:    BIT     #100,R1
1780 005654 001354          BNE     1$                ;IS FL CLEAR.
1781 005656 012737 043245 001254    MOV     #SETI1,$TMP11
1782 005664 012737 000204 001246    MOV     #204,$TMP6
1783 005672 000400          BR      CERR3
1784
1785          ;REPORT THE ERRORS:
1786 005674          CERR3:
1787 005674 010137 001240    MOV     R1,$TMP3
1788 005700 010237 001242    MOV     R2,$TMP4
1789 005704 012637 005740    MOV     (SP)+,CPC
1790 005710 104012          1$:    ERROR  +12
1791 005712 000177 000022    JMP     @CPC
1792
1793 005716          CERR4:
1794 005716 010137 001240    MOV     R1,$TMP3
1795 005722 010237 001242    MOV     R2,$TMP4
1796 005726 012637 005740    MOV     (SP)+,CPC
1797 005732 104013          1$:    ERROR  +13
1798 005734 000177 000000    JMP     @CPC
1799
1800 005740 000000          CPC:   .WORD  0
1801
1802 005742          CDONE:
      005742 104413    RSETUP
  
```

```

;GO INITIALIZE THE PPS 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?).
  
```

1803  
1804



1824

```

.SBTTL TEST # 4 - ILLEGAL FPP OP CODES AND STST TEST
*****
*TEST 4 ILLEGAL FPP OP CODES AND STST TEST
*
*THIS IS A TEST OF THE FPP OPERATION CODES:
*
*      170003
*      170004
*      :
*      170010
*      170013
*      170014
*      :
*      170077
*THESE ARE ILLEGAL INSTRUCTIONS AND (WITH INTERRUPTS ENABLED)
*SHOULD CAUSE A TRAP TO 244.
*ALSO TESTED HERE IS THE INSTRUCTION:
*      STST R1
*WHICH SHOULD PUT THE FEC CODE 2 IN R1, AFTER ANY OF THE ABOVE
*OP CODES IS EXECUTED.
*
*****
  
```

```

1825 005744 000004
1826 005746 012737 005774 001110
1827 005754 012705 170003
1828 005760 012737 006164 000004
1829 005766 012737 006070 000244
1830 005774 005000
1831 005776 170100
1832 006000 005002
1833 006002 010537 006020
1834 006006 010537 001244
1835 006012 012737 006020 001236
1836 006020 000000
1837 006022 170000
1838 006024 005202
1839 006026 005202
1840
1841 006030 170201
1842 006032 010137 001240
1843 006036 104016
1844
1845 006040 022705 170010
1846 006044 001003
1847 006046 012705 170013
1848 006052 000750
1849
1850 006054 022705 170077
1851 006060 001001
1852 006062 000452
1853 006064 005205
1854 006066 000742
1855
1856 006070 022716 006022
1857 006074 001402
1858 006076 000137 042066
1859
  
```

```

TST4: SCOPE
      MOV #D1,$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
      MOV #170003,R5 ;INITIAL OP CODE.
      MOV #DERR2,ERRVECT
      MOV #DERR1,FPVECT

D1: CLR R0
    LDFPS R0 ;CLEAR FPS.
    CLR R2
    MOV R5,D2 ;SET UP THE ILLEGAL INSTRUCTION.
    MOV R5,$TMP5
    MOV #D2,$TMP2

D2: .WORD 0
D3: CFCC
D4: INC R2
    INC R2

1$: STFPS R1 ;REPORT FAILURE. DID NOT TRAP.
    MOV R1,$TMP3
    ERROR +16

D5: CMP #170010,R5 ;COMPUTE NEXT OP CODE
    BNE D6
    MOV #170013,R5
    BR D1

D6: CMP #170077,R5
    BNE D7
    BR DDONE

D7: INC R5
    BR D1

DERR1: CMP #D3,(SP) ;DID TRAP OCCUR ON TEST INSTRUCTION?
      BEQ 1$
      JMP FPSPUR
  
```



```

1860 006102 022626          1$:    CMP      (SP)+,(SP)+
1861 006104 170201          STFPS   R1           ;GET THE FPS AND SEE IF IT IS
1862 006106 022701 100000    CMP      #100000,R1  ;SET CORRECTLY.
1863 006112 001406          BEQ      3$
1864
1865 006114 012737 100000 001240    MOV      #100000,$TMP3
1866 006122 010137 001242    MOV      R1,$TMP4
1867 006126 104017          2$:    ERROR   +17
1868
1869 006130 012704 000001    3$:    MOV      #1,R4
1870 006134 170304          D8:    STST      R4           ;GET THE FEC CODE. NOTE THAT
1871                                     ;IF THE DESTINATION MODE IS
1872                                     ;IMPROPERLY DECODED AN ODD
1873                                     ;ADDRESS TRAP TO 4 SHOULD OCCUR.
1874 006136 022704 000002          CMP      #2,R4           ;WAS FEC CORRECT?
1875 006142 001001          BNE      D9
1876 006144 000735          BR       D5
1877
1878 006146          D9:    ;REPORT STST FAILURE
1879 006146 012737 006134 001240    MOV      #D8,$TMP3
1880 006154 010437 001242    MOV      R4,$TMP4
1881 006160 104020          1$:    ERROR   +20
1882 006162 000726          BR       D5
1883
1884 006164 022716 006136    DERR2:  CMP      #D8+2,(SP)   ;DID THE TRAP OCCUR ON THE
1885 006170 001402          BEQ      D10           ;STST INSTRUCTION?
1886 006172 000137 042120    JMP      CPSPUR
1887
1888 006176          D10:
1889 006176 011637 001236    MOV      (SP),$TMP2
1890 006202 022626          CMP      (SP)+,(SP)+
1891 006204 104021          1$:    ERROR   +21
1892 006206 000714          BR       D5
1893
1894 006210          DDONE:
      006210 104413          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?).
1895
1896
    
```



1904

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

```
1905 006212 000004
1906 006214 012737 006230 001110
1907 006222 012737 006324 000244
1908 006230 012700 040000
1909 006234 170100
1910 006236 012737 006244 001236
1911 006244
1912 006244 170020
1913 006246 170000
1914
1915 006250 170201
1916 006252 022701 140000
1917 006256 001005
1918
1919 006260 170304
1920 006262 022704 000002
1921 006266 001010
1922 006270 000431
1923
1924 006272
1925 006272 010137 001240
1926 006276 012737 140000 001242
1927 006304 104022
1928 006306 000422
1929
1930 006310
1931 006310 010537 001240
1932 006314 010437 001242
1933 006320 104023
1934 006322 000414
1935
1936 006324 021627 006246
1937 006330 001402
1938 006332 000137 042066
1939
1940 006336
1941 006336 011637 001236
1942 006342 022626
1943 006344 170201
1944 006346 010137 001240
1945 006352 104024
1946
1947 006354
006354 104413
```

```
TST5: SCOPE
MOV #E1,$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
MOV #EERR2,FPVECT ;SETUP FOR THE INTERRUPT.

E1: MOV #40000,R0
LDFPS R0 ;SET FID.
MOV #E3,$TMP2

E2:
E3: .WORD 170020 ;ILLEGAL FPP INSTRUCTION.
E4: CFCC

STFPS R1 ;SEE IF ERROR WAS DETECTED.
CMP #140000,R1
BNE EERR0

STST R4 ;SEE IF FEC=2
CMP #2,R4
BNE EERR1
BR EDONE

EERR0: ;REPORT FPS INCORRECTLY SET.
MOV R1,$TMP3
MOV #140000,$TMP4
1$: ERROR +22
BR EDONE

EERR1: ;REPORT FEC NOT 2.
MOV R5,$TMP3
MOV R4,$TMP4
1$: ERROR +23
BR EDONE

EERR2: ;DID THE ILLEGAL INSTRUCTION TRAP?
CMP (SP),#E4
BEQ 1$
JMP FPSPUR

1$: MOV (SP),$TMP2
CMP (SP)+,(SP)+
STFPS R1
MOV R1,$TMP3
2$: ERROR +24

EDONE: 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?).

1948  
1949



1962

```
.SBTTL TEST # 6 - LDD AND STD, WITH SRC AND DST MODE 1, TEST
*****
*TEST 6      LDD AND STD, WITH SRC AND DST MODE 1, TEST
*
*THIS IS A TEST OF BOTH THE INSTRUCTION:
*      LDD      (R0),ACO
*AND THE INSTRUCTION:
*      STD      ACO,(R0)
*
*MOST OF THE FAILURES ARE ISOLATED TO THE SRC OR DST FLOWS. NOTE
*THAT THE INTEGRITY OF ACO HAS NOT BEEN ASSURED. THIS MEANS THAT
*IN SOME CASES IT WILL BE IMPOSSIBLE TO ISOLATE CERTAIN DATA PATTERN
*FAILURES TO EITHER THE FLOWS OR THIS ACCUMULATOR.
*
*****
```

```
006356 000004
1963
1964 006360
      006360 012737 006360 001110
1965 006366 012737 006436 001236
1966 006374 005000
1967 006376 170100
1968 006400 170011
1969 006402 012701 010174
1970 006406 012702 010240
1971 006412 012703 000010
1972
1973 006416 012221
1974 006420 077302
1975
1976 006422 012700 010204
1977 006426 012737 007660 000004
1978
1979 006434 005003
1980
1981 006436 172410
1982 006440 005203
1983 006442 005203
1984
1985 006444 020027 010204
1986 006450 001402
1987 006452 000137 007024
1988
1989 006456 020227 000002
1990 006462 001402
1991 006464 000137 007122
1992
1993 006470 012701 010174
1994 006474 012702 010240
1995 006500 012703 000010
1996 006504 022122
1997 006506 001402
1998 006510 000137 006766
1999 006514 077305
2000
2001 006516 170201
2002 006520 022701 000200
2003 006524 001402
```

```
TST6: SCOPE
F1:  MOV #F1,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
      MOV #F3,$TMP2
      CLR R0
      LDFPS R0
      SETD
      MOV #FDAT10,R1      ;SET UP THE LOAD DATA.
      MOV #FXDAT0,R2
      MOV #10,R3
F2:  MOV (R2)+,(R1)+
      SOB R3,F2
      MOV #FDAT14,R0      ;SETUP R0 FOR THE LDD (R0),ACO.
      MOV #FERR20,ERRVECT ;IF THE SRC FLOWS FAIL THEN
                          ;AN ODD ADDRESS MAY OCCUR.
F3:  LDD (R0),ACO
F4:  INC R3
      INC R3
      CMP R0,#FDAT14      ;WAS R0 AFFECTED?
      BEQ F5
      JMP FERR1
F5:  CMP R3,#2
      BEQ 1$
      JMP FERR2
1$:  MOV #FDAT10,R1
      MOV #FXDAT0,R2
      MOV #10,R3
2$:  CMP (R1)+,(R2)+
      BEQ 3$
      JMP FERRO
3$:  SOB R3,2$
      STFPS R1
      CMP #200,R1
      BEQ F6
      ;MAKE SURE THE FPS IS CORRECT.
```



```

2004 006526 000137 007640          JMP      FERR11
2005
2006 006532          F6:      MOV      #1$, $LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
      006532 012737 006540 001110      MOV      #F10, $TMP2
2007 006540 012737 006602 001236 1$:      MOV
2008
2009 006546 012703 177777          MOV      #-1, R3
2010 006552 012704 000010          MOV      #10, R4
2011 006556 012705 010216          MOV      #FDAT00, R5      ;SET UP THE OUTPUT DATA BUFFER.
2012 006562 010325          F7:      MOV      R3, (R5)+
2013 006564 077402          SOB      R4, F7
2014
2015 006566 012700 010226          MOV      #FDAT04, R0      ;SET UP R0 FOR DST MODE 1 REG 0.
2016 006572 012737 010026 000004      MOV      #FERR25, ERRVECT ;IF THE DST FLOWS FAIL AN ODD
2017                                          ;ADDRESS COULD OCCUR.
2018 006600 005003          CLR      R3
2019
2020 006602 174010          F10:     STD      AC0, (R0)      ;TEST INSTRUCTION.
2021 006604 005203          F11:     INC      R3
2022 006606 005203          INC      R3
2023
2024 006610 020027 010226          CMP      R0, #FDAT04      ;WAS R0 MODIFIED?
2025 006614 001402          BEQ      F12
2026 006616 000137 007162          JMP      FERR3
2027
2028 006622 020327 000002          F12:     CMP      R3, #2      ;WAS THE PC AFFECTED CORRECTLY?
2029 006626 001402          BEQ      F135
2030 006630 000137 007154          JMP      FERR4
2031
2032 006634 012701 010216          F135:    MOV      #FDAT00, R1
2033 006640 012702 010240          MOV      #FXDAT0, R2
2034
2035 006644 022122          CMP      (R1)+, (R2)+      ;SEE IF THE DATA WAS OUTPUT
2036 006646 001402          BEQ      F13              ;TO THE TARGET AREA CORRECTLY.
2037 006650 000137 007260          JMP      FERR5
2038
2039 006654 022122          F13:     CMP      (R1)+, (R2)+
2040 006656 001402          BEQ      F14
2041 006660 000137 007260          JMP      FERR5
2042
2043 006664 022122          F14:     CMP      (R1)+, (R2)+
2044 006666 001402          BEQ      F15
2045 006670 000137 007260          JMP      FERR5
2046
2047 006674 022122          F15:     CMP      (R1)+, (R2)+
2048 006676 001402          BEQ      F16
2049 006700 000137 007260          JMP      FERR5
2050
2051 006704 022122          F16:     CMP      (R1)+, (R2)+
2052 006706 001402          BEQ      F17
2053 006710 000137 007604          JMP      FERR10
2054
2055 006714 022122          F17:     CMP      (R1)+, (R2)+
2056 006716 001402          BEQ      F20
2057 006720 000137 007314          JMP      FERR6
2058
2059 006724 022122          F20:     CMP      (R1)+, (R2)+
    
```



2060	006726	001402				BEQ	F21		
2061	006730	000137	007450			JMP	FERR7		
2062									
2063	006734	022122			F21:	CMP	(R1)+,(R2)+		
2064	006736	001402				BEQ	F22		
2065	006740	000137	007604			JMP	FERR10		
2066									
2067	006744	005001			F22:	CLR	R1		
2068	006746	170201				STFPS	R1		:MAKE SURE FPS IS CORRECT.
2069	006750	022701	000200			CMP	#200,R1		
2070	006754	001402				BEQ	F23		
2071	006756	000137	007640			JMP	FERR11		
2072	006762	000137	010260		F23:	JMP	FDONE		
2073									
2074	006766				FERR0:				:SOURCE DATA AFFECTED BY
2075	006766	012737	010240	001240		MOV	#FXDAT0,\$TMP3		:THE LDD INSTRUCTION.
2076	006774	012737	010252	001242		MOV	#FXDAT0+12,\$TMP4		
2077	007002	012737	010174	001244		MOV	#FDAT10,\$TMP5		
2078	007010	012737	010206	001246		MOV	#FDAT10+12,\$TMP6		
2079	007016	104025			1\$:	ERROR	+25		
2080	007020	000137	010260			JMP	FDONE		
2081									
2082	007024	012737	010204	001242	FERR1:	MOV	#FDAT14,\$TMP4		:FSRC FLOWS FAILURE.
2083	007032	010037	001240			MOV	R0,\$TMP3		
2084	007036	012737	000762	001244		MOV	#762,\$TMP5		
2085	007044	012737	000321	001250		MOV	#321,\$TMP7		
2086									
2087	007052	022700	010174			CMP	#FDAT10,R0		:FSRC MODE 4?
2088	007056	001004				BNE	1\$		
2089	007060	012737	000324	001246		MOV	#324,\$TMP6		
2090	007066	000412				BR	4\$		
2091									
2092	007070	022700	010214		1\$:	CMP	#FDAT14+10,R0		:FSRC MODE 2?
2093	007074	001004				BNE	2\$		
2094	007076	012737	000322	001246		MOV	#322,\$TMP6		
2095	007104	000403				BR	4\$		
2096									
2097	007106				2\$:				
2098	007106	104027			3\$:	ERROR	+27		
2099	007110	000137	010260			JMP	FDONE		
2100									
2101	007114				4\$:				
2102	007114	104026			5\$:	ERROR	+26		
2103	007116	000137	010260			JMP	FDONE		
2104									
2105	007122	012701	006440		FERR2:	MOV	#F4,R1		:THE PC WAS INCORRECTLY AFFECTED
2106									:DURING THE INSTRUCTION.
2107	007126	010137	001242		FER2:	MOV	R1,\$TMP4		
2108	007132	162701	000004			SUB	#4,R1		
2109	007136	006303				ASL	R3		
2110	007140	060301				ADD	R3,R1		
2111	007142	010137	001240			MOV	R1,\$TMP3		
2112	007146	104030			1\$:	ERROR	+30		
2113	007150	000137	010260			JMP	FDONE		
2114									
2115	007154	012701	006604		FERR4:	MOV	#F11,R1		
2116	007160	000762				BR	FER2		



2117										
2118	007162	012737	010226	001242	FERR3:	MOV	#FDAT04,\$TMP4			:FAILURE IN THE FDST FLOWS.
2119	007170	010037	001240			MOV	R0,\$TMP3			
2120	007174	012737	000527	001244		MOV	#527,\$TMP5			
2121	007202	012737	000641	001250		MOV	#641,\$TMP7			
2122										
2123	007210	022700	010216			CMP	#FDAT00,R0			:DST MODE 4?
2124	007214	001004				BNE	1\$			
2125	007216	012737	000644	001246		MOV	#644,\$TMP6			
2126	007224	000412				BR	4\$			
2127										
2128	007226	022700	010236		1\$:	CMP	#FDAT04+10,R0			:DST MODE 2?
2129	007232	001004				BNE	2\$			
2130	007234	012737	000642	001246		MOV	#642,\$TMP6			
2131	007242	000403				BR	4\$			
2132										
2133	007244				2\$:					
2134	007244	104032			3\$:	ERROR	+32			
2135	007246	000137	010260			JMP	FDONE			
2136										
2137	007252				4\$:					
2138	007252	104031			5\$:	ERROR	+31			
2139	007254	000137	010260			JMP	FDONE			
2140										
2141	007260				FERR5:					:FAILURE OF STD.
2142	007260	010037	001240			MOV	R0,\$TMP3			
2143	007264	012737	010216	001242		MOV	#FDAT00,\$TMP4			
2144	007272	012737	010234	001244		MOV	#FDAT07,\$TMP5			
2145	007300	012737	010240	001246		MOV	#FXDAT0,\$TMP6			
2146	007306	104033			1\$:	ERROR	+33			
2147	007310	000137	010260			JMP	FDONE			
2148										
2149	007314	012701	010230		FERR6:	MOV	#FDAT05,R1			:DID (BUT GR7) FAIL IN THE FDST
2150	007320	012702	177777			MOV	#-1,R2			:FLOWS?
2151	007324	012703	000003			MOV	#3,R3			
2152	007330	020221			1\$:	CMP	R2,(R1)+			
2153	007332	001017				BNE	5\$			
2154	007334	077303				SOB	R3,1\$			
2155										
2156										:REPORT FAILURE OF (BUT GR7) IN
2157	007336	010037	001240			MOV	R0,\$TMP3			:THE FDST FLOWS.
2158	007342	012737	000412	001244		MOV	#412,\$TMP5			
2159	007350	012737	000147	001246		MOV	#147,\$TMP6			
2160	007356	012737	000145	001250		MOV	#145,\$TMP7			
2161	007364	104034			2\$:	ERROR	+34			
2162	007366	000137	010260			JMP	FDONE			
2163										
2164	007372	012701	010230		5\$:	MOV	#FDAT05,R1			:DID (BUT GR7) FAIL IN THE SRC FLOWS?
2165	007376	012703	000003			MOV	#3,R3			
2166	007402	005721			6\$:	TST	(R1)+			
2167	007404	001402				BEQ	7\$			
2168	007406	000137	007604			JMP	FERR10			
2169	007412	077305			7\$:	SOB	R3,6\$			
2170										
2171										:REPORT FAILURE OF (BUT GR7) IN
2172	007414	010037	001240			MOV	R0,\$TMP3			:THE FSRC FLOWS.
2173	007420	012737	000207	001244		MOV	#207,\$TMP5			











```
2288 010142 000446 BR 7 F DONE
2289
2290 010144 FERR26: ;REPORT FDST FLOW FAILURE TO MODE
2291 010144 012737 044035 001264 MOV #MS16,$TMP15 ;6 OR MODE 7.
2292 010152 012737 000646 001246 MOV #646,$TMP6
2293 010160 012737 000647 001252 MOV #647,$TMP10
2294 010166 022626 CMP (SP)+,(SP)+
2295 010170 104043 1$: ERROR +43
2296 010172 000432 BR F DONE
2297
2298 010174 177777 FDATA0: -1
2299 010176 177777 FDATA1: -1
2300 010200 177777 FDATA2: -1
2301 010202 177777 FDATA3: -1
2302 010204 177777 FDATA4: -1
2303 010206 177777 FDATA5: -1
2304 010210 177777 FDATA6: -1
2305 010212 177777 FDATA7: -1
2306 010214 177777 -1
2307 010216 177777 FDATA00: -1
2308 010220 177777 FDATA01: -1
2309 010222 177777 FDATA02: -1
2310 010224 177777 FDATA03: -1
2311 010226 177777 FDATA04: -1
2312 010230 177777 FDATA05: -1
2313 010232 177777 FDATA06: -1
2314 010234 177777 FDATA07: -1
2315 010236 177777 -1
2316 010240 177777 FXDATA0: -1
2317 010242 177777 FXDATA1: -1
2318 010244 177777 FXDATA2: -1
2319 010246 177777 FXDATA3: -1
2320 010250 052525 FXDATA4: 052525
2321 010252 031463 FXDATA5: 031463
2322 010254 007417 FXDATA6: 007417
2323 010256 000477 FXDATA7: 000477
2324
2325
2326 010260 F DONE:
      010260 104413 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?).

2327
2328
```



2334

```
.SBTTL TEST # 7 - FSRC MODE 0 TEST
:*****
:*TEST 7 FSRC MODE 0 TEST
:*
:*THIS IS A TEST OF FSRC MODE ZERO USING THE LDD AND LDF INSTRUCTIONS.
:*
:*****
```

```
TST7: SCOPE
MOV #I1,$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.

I1: SETD ;SET FD.
MOV #IDATIO,R0
MOV #IPATIO,R1
MOV #4,R2
I2: MOV (R1)+,(R0)+ ;SET UP THE INPUT DATA BUFFER.
SOB R2,I2

MOV #IDATIO,R0 ;LOAD AC1
LDD (R0),AC1

MOV #IPATIO,R0 ;LOAD AC0
LDD (R0),AC0

MOV #1,R1 ;IN CASE THE FSRC FLOWS FAIL
MOV #IERR0,ERRVCT ;AN ODD ADDRESS TRAP TO 4 MAY OCCUR.
MOV #I3,$TMP2
MOV #MS35,$TMP3
I3: LDD AC1,AC0 ;TEST INSTRUCTION.
I4: NOP
I5: NOP

MOV #IDATIO,R0
STD AC0,(R0) ;GET AC0, THE RESULTS.

MOV #IDATIO,R0 ;SEE IF DATA IS CORRECT.
MOV #IDATIO,R1
MOV #4,R2
I6: CMP (R0)+,(R1)+
BEQ I105

MOV #IDATIO,R0 ;SEE IF (BUT FD) FAILED.
MOV #2,R2
I7: TST (R0)+
BEQ I10

MOV #IDATIO,R0
MOV #2,R2
I8: CMP #-1,(R0)+
BEQ 2$
JMP IERR1
I9: SOB R2,I9
BR I106
I10: SOB R2,I10
I106: JMP IERR2

I105: SOB R2,I105
```

```
010262 000004
2335 010264 012737 010272 001110
2336
2337 010272
2338 010272 170011
2339 010274 012700 011122
2340 010300 012701 011072
2341 010304 012702 000004
2342 010310 012120
2343 010312 077202
2344
2345 010314 012700 011122
2346 010320 172510
2347
2348 010322 012700 011102
2349 010326 172410
2350
2351 010330 012701 000001
2352 010334 012737 010672 000004
2353 010342 012737 010356 001236
2354 010350 012737 044515 001240
2355 010356 172401
2356 010360 000240
2357 010362 000240
2358
2359 010364 012700 011112
2360 010370 174010
2361
2362 010372 012700 011112
2363 010376 012701 011122
2364 010402 012702 000004
2365 010406 022021
2366 010410 001424
2367
2368 010412 012700 011116
2369 010416 012702 000002
2370 010422 005720
2371 010424 001413
2372
2373 010426 012700 011116
2374 010432 012702 000002
2375 010436 022720 177777
2376 010442 001402
2377 010444 000137 010754
2378 010450 077206
2379 010452 000401
2380 010454 077216
2381 010456 000137 010774
2382
2383 010462 077227
```



```

2384
2385           ;NOW TEST THE LOAD INSTRUCTION WITH FSRC MODE ZERO AND FD CLEAR.
2386
2387 010464      I11:      MOV      #I12,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
      010464 012737 010472 001110
2388 010472 012700 011072      I12:      MOV      #IPAT10,R0
2389 010476 012701 011122      MOV      #IDAT10,R1
2390 010502 012702 000004      MOV      #4,R2
2391 010506 012021      I13:      MOV      (R0)+,(R1)+
2392 010510 077202      SOB      R2,I13
2393
2394 010512 012700 011122      MOV      #IDAT10,R0      ;SET UP AC1
2395 010516 172510      LDD      (R0),AC1
2396
2397 010520 012700 011102      MOV      #IPAT20,R0      ;SET UP AC0
2398 010524 172410      LDD      (R0),AC0
2399
2400 010526 012701 000001      MOV      #1,R1
2401 010532 012737 010550 001236      MOV      #I14,$TMP2
2402 010540 012737 044522 001240      MOV      #MS36,$TMP3
2403 010546 170001      SETF
      ;CLEAR FD.
2404
2405 010550 172401      I14:      LDF      AC1,AC0      ;TEST INSTRUCTION.
2406 010552 000240      I15:      NOP
2407 010554 000240      I16:      NOP
2408
2409 010556 170200      STFPS   R0      ;SEE IF FPS IS STILL CLEAR.
2410 010560 022700 000004      CMP      #4,R0
2411 010564 001402      BEQ      I17
2412 010566 000137 011046      JMP      IERR3
2413
2414 010572      I17:      ;RESET TO DOUBLE MODE.
2415 010572 170011      SETD
2416
2417 010574 012700 011112      MOV      #IDAT00,R0
2418 010600 174010      STD      AC0,(R0)      ;GET AC0
2419
2420 010602 012737 177777 011126      MOV      #-1,IDAT12
2421 010610 012737 177777 011130      MOV      #-1,IDAT13
2422 010616 012700 011112      MOV      #IDAT00,R0
2423 010622 012701 011122      MOV      #IDAT10,R1
2424 010626 012702 000004      MOV      #4,R2
2425 010632 022021      I20:      CMP      (R0)+,(R1)+      ;SEE IF AC0 WAS CORRECT.
2426 010634 001414      BEQ      I23
2427
2428 010636 023737 011116 011076      CMP      IDAT02,IPAT12      ;DID (BUT FD) FAIL?
2429 010644 001402      BEQ      I22
2430 010646 000137 010754      I21:      JMP      IERR1
2431 010652 023737 011120 011100      I22:      CMP      IDAT03,IPAT13
2432 010660 001372      BNE      I21
2433 010662 000137 011022      JMP      IERR4
2434
2435 010666 077217      I23:      SOB      R2,I20
2436
2437 010670 000520      BR      IDONE      ;NO ERRORS.
2438
2439           ;IF AN ODD ADDRESS TRAP OCCURS COME HERE TO ANALYZE THE FSRC FAILURE.

```



```

2440 010672 022716 010360      IERR0:  CMP      #14,(SP)      ;MAKE SURE THE TRAP OCCURRED
2441 010676 001413              BEQ      1$              ;ON THE INSTRUCTION BEING TESTED.
2442 010700 022716 010362      CMP      #15,(SP)
2443 010704 001410              BEQ      1$
2444 010706 022716 010552      CMP      #115,(SP)
2445 010712 001405              BEQ      1$
2446 010714 022716 010554      CMP      #116,(SP)
2447 010720 001402              BEQ      1$
2448 010722 000137 042120      JMP      CPSPUR
2449
2450 010726 011637 001236      1$:      MOV      (SP), $TMP2      ;REPORT FAILURE.
2451 010732 012737 000627 001240  MOV      #627, $TMP3
2452 010740 012737 000320 001242  MOV      #320, $TMP4
2453 010746 022626              CMP      (SP)+, (SP)+
2454 010750 104047      2$:      ERROR    +47
2455 010752 000467              BR       IDONE
2456
2457              ;REPORT DATA ERROR.
2458 010754              IERR1:
2459 010754 012737 011122 001242  MOV      #IDATIO, $TMP4
2460 010762 012737 011112 001244  MOV      #IDAT00, $TMP5
2461 010770 104051      1$:      ERROR    +51
2462 010772 000457              BR       IDONE
2463
2464              ;REPORT FAILURE OF (BUT FD)
2465 010774 012737 000153 001244  IERR2:  MOV      #153, $TMP5
2466 011002 012737 000434 001246  MOV      #434, $TMP6
2467 011010 012737 000435 001250  MOV      #435, $TMP7
2468 011016
2469 011016 104050      IERR25:
2470 011020 000444      1$:      ERROR    +50
2471 011022 012737 000153 001244  BR       IDONE
2472 011030 012737 000435 001246  IERR4:  MOV      #153, $TMP5
2473 011036 012737 000434 001250  MOV      #435, $TMP6
2474 011044 000764  MOV      #434, $TMP7
2475              BR       IERR25
2476
2477              ;REPORT INCORRECT FPS AFTER LOAD INSTRUCTION.
2478 011046              IERR3:
2479 011054 010037 001240  MOV      #114, $TMP2
2480 011060 012737 000004 001242  MOV      R0, $TMP3
2481 011066 104041  MOV      #4, $TMP4
2482 011070 000420      1$:      ERROR    +41
2483              BR       IDONE
2484
2485 011072 000000      IPAT10: 0
2486 011074 170360      IPAT11: 170360
2487 011076 016161      IPAT12: 016161
2488 011100 052525      IPAT13: 052525
2489
2490 011102 177777      IPAT20: -1
2491 011104 177777      IPAT21: -1
2492 011106 177777      IPAT22: -1
2493 011110 177777      IPAT23: -1
2494
2495 011112 000000      IDAT00: 0
2496 011114 000000      IDAT01: 0

```



2497 011116 000000  
2498 011120 000000  
~~2499~~  
2500 011122 000000  
2501 011124 000000  
2502 011126 000000  
2503 011130 000000  
~~2504~~  
2505 011132  
011132 104413

IDAT02: 0  
IDAT03: 0  
  
IDAT10: 0  
IDAT11: 0  
IDAT12: 0  
IDAT13: 0  
  
IDONE: 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?).

2506  
2512



2513

```
.SBTTL TEST # 10 - FDST MODE 0 TEST
*****
:TEST 10      FDST MODE 0 TEST
:
:THIS IS A TEST OF THE STORE INSTRUCTIONS, STD AND STF, WITH FDST MODE 0.
:
*****
```

```
TST10: SCOPE
T1:
2514 011134 000004
011136
2515 011136 012737 011144 001110
011144 170011
2516 011146 012700 011710
2517 011152 012701 011740
2518 011156 012702 000004
2519 011162 012021
2520 011164 077202
2521
2522 011166 012700 011740
2523 011172 172410
2524
2525 011174 012700 011720
2526 011200 172510
2527
2528 011202 012701 000001
2529 011206 012737 011516 000004
2530 011214 012737 011230 001236
2531 011222 012737 044515 001240
2532 011230 174001
2533 011232 000240
2534 011234 000240
2535
2536 011236 012700 011730
2537 011242 174110
2538
2539 011244 012703 011730
2540 011250 012704 011740
2541 011254 012705 000004
2542 011260 022324
2543 011262 001413
2544
2545 011264 012703 011734
2546 011270 012705 000002
2547 011274 005723
2548 011276 001402
2549 011300 000137 011600
2550 011304 077505
2551 011306 000137 011620
2552
2553 011312 077516
2554
2555
2556
2557 011314
011314 012737 011322 001110
2558
2559 011322 012700 011710
2560 011326 012701 011740

T1:      MOV      #15,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
          SETD     ;SET FD
1$:      MOV      #TPAT10,R0
          MOV      #TDAT10,R1
          MOV      #4,R2
T2:      MOV      (R0), (R1)+      ;SET UP THE INPUT DATA BUFFER.
          SOB     R2,T2
          MOV      #TDAT10,R0      ;LOAD ACO
          LDD     (R0),ACO
          MOV      #TPAT20,R0      ;LOAD AC1
          LDD     (R0),AC1
          MOV      #1,R1           ;IF THE (BUT FDST) FORK FAILS
          MOV      #TERR0,ERRVECT ;AN ODD ADDRESS TRAP COULD RESULT.
          MOV      #T3,$TMP2
          MOV      #MS35,$TMP3
T3:      STD     ACO,AC1
T4:      NOP
T5:      NOP
          MOV      #TDAT00,R0
          STD     AC1,(R0)        ;GET THE DATA.
          MOV      #TDAT00,R3      ;SEE IF THE DATA IS CORRECT.
          MOV      #TDAT10,R4
          MOV      #4,R5
T6:      CMP     (R3)+,(R4)+
          BEQ     T105
          MOV      #TDAT02,R3      ;DID (BUT FD) FAIL?
          MOV      #2,R5
T7:      TST     (R3)+
          BEG     T10
          JMP     TERR1
T10:     SOB     R5,T7
          JMP     TERR2
T105:    SOB     R5,T6
          ;NOW TEST THE STF ACO,AC1 INSTRUCTION.
T11:     MOV      #T12,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
          MOV      #TPAT10,R0      ;SET UP THE INPUT DATA BUFFER.
          MOV      #TDAT10,R1
```



```

2561 011332 012702 000004          MOV    #4,R2
2562 011336 012021          T13:  MOV    (R0)+,(R1)+
2563 011340 077202          SOB    R2,T13
2564
2565 011342 012700 011740          MOV    #TDATIO,R0          ;SET UP ACO
2566 011346 172410          LDD    (R0),AC0
2567
2568 011350 012700 011720          MOV    #TPAT20,R0         ;SET UP AC1
2569 011354 172510          LDD    (R0),AC1
2570
2571 011356 012701 000001          MOV    #1,R1
2572 011362 012737 011400 001236    MOV    #T14,$TMP2
2573 011370 012737 044522 001240    MOV    #MS36,$TMP3
2574 011376 170001          SETF
2575 011400 174001          T14:  STF    AC0,AC1          ;CLEAR FD
2576 011402 000240          T15:  NOP
2577 011404 000240          T16:  NOP
2578
2579 011406 005000          CLR    R0
2580 011410 170200          STFPS R0          ;SEE IF FPS IS CLEAR.
2581 011412 022700 000010          CMP    #10,R0
2582 011416 001401          BEQ   T17
2583 011420 000521          BR    TERR3
2584
2585 011422          T17:
2586 011422 170011          SETD
          ;SET FD.
2587
2588 011424 012700 011730          MOV    #TDAT00,R0
2589 011430 174110          STD    AC1,(R0)          ;PICK UP AC1.
2590
2591 011432 012737 177777 011744    MOV    #-1,TDAT12
2592 011440 012737 177777 011746    MOV    #-1,TDAT13
2593 011446 012703 011730          MOV    #TDATC0,R3
2594 011452 012704 011740          MOV    #TDATIO,R4
2595 011456 012705 000004          MOV    #4,R5
2596 011462 022324          T20:  CMP    (R3)+,(R4)+
2597 011464 001412          BEQ   T23          ;WAS THE DATA TRANSFERRED CORRECTLY?
2598
2599 011466 023737 011734 011714    CMP    TDAT02,TPAT12
2600 011474 001401          BEQ   T22          ;DID (BUT FD) FAIL.
2601 011476 000440          BR    TERR1
2602 011500 023737 011736 011716    T21:  CMP    TDAT03,TPAT13
2603 011506 001373          T22:  BNE   T21
2604 011510 000456          BR    TERR4
2605
2606 011512 077515          T23:  SOB   R5,T20
2607 011514 000515          BR    TDONE
2608
2609
2610          ;TRAP HERE THROUGH VECTOR 4 IF AN ODD ADDRESS OCCURS.
2611 011516 022716 011232          TERRO: CMP    #T4,(SP)          ;MAKE SURE THE TRAP WAS ON
2612 011522 001413          BEQ   1$          ;AN INSTRUCTION BEING TESTED.
2613 011524 022716 011234          CMP    #T5,(SP)
2614 011530 001410          BEQ   1$
2615 011532 022716 011402          CMP    #T15,(SP)
2616 011536 001405          BEQ   1$
2617 011540 022716 011404          CMP    #T16,(SP)

```



```
2618 011544 001402          BEQ      1$
2619 011546 000137 042120    JMP      CPSPUR
2620
2621 011552 011637 001236    1$:     MOV      (SP), $TMP2
2622 011556 022626          CMP      (SP)+, (SP)+
2623 011560 012737 000527 001240    MOV      #527, $TMP3
2624 011566 012737 000640 001242    MOV      #640, $TMP4
2625 011574 104121          2$:     ERROR   +121
2626 011576 000464          BR       TDONE
2627
2628          ;REPORT DATA FAILURE.
2629 011600          TERR1:
2630 011600 012737 011740 001242    MOV      #TDATIO, $TMP4
2631 011606 012737 011730 001244    MOV      #TDAT00, $TMP5
2632 011614 104123          1$:     ERROR   +123
2633 011616 000454          BR       TDONE
2634
2635          ;REPORT FAILURE OF (BUT FD).
2636 011620 012737 000160 001246    TERR2:  MOV      #160, $TMP6
2637 011626 012737 000161 001250    MOV      #161, $TMP7
2638 011634 012737 000640 001244    TERR25: MOV      #640, $TMP5
2639 011642 104122          1$:     ERROR   +122
2640 011644 000441          BR       TDONE
2641 011646 012737 000161 001246    TERR4:  MOV      #161, $TMP6
2642 011654 012737 000160 001250    MOV      #160, $TMP7
2643 011662 000764          BR       TERR25
2644
2645          ;REPORT INCORRECT FPS AFTER STORE INSTRUCTION.
2646 011664          TERR3:
2647 011664 012737 011402 001236    MOV      #115, $TMP2
2648 011672 010037 001240          MOV      R0, $TMP3
2649 011676 012737 000010 001242    MOV      #10, $TMP4
2650 011704 104041          1$:     ERROR   +41
2651 011706 000420          BR       TDONE
2652
2653 011710 000000          TPAT10: 0
2654 011712 170360          TPAT11: 170360
2655 011714 016161          TPAT12: 016161
2656 011716 052525          TPAT13: 052525
2657
2658 011720 177777          TPAT20: -1
2659 011722 177777          TPAT21: -1
2660 011724 177777          TPAT22: -1
2661 011726 177777          TPAT23: -1
2662
2663 011730 000000          TDATA0: 0
2664 011732 000000          TDATA1: 0
2665 011734 000000          TDATA2: 0
2666 011736 000000          TDATA3: 0
2667
2668 011740 000000          TDATA0: 0
2669 011742 000000          TDATA1: 0
2670 011744 000000          TDATA2: 0
2671 011746 000000          TDATA3: 0
2672
2673 011750          TDONE:
      011750 104413          RSETUP          ;GO INITIALIZE THE FPS AND STACK; AND
```



CKFPACO FP11F FLTG PNT PRT A  
TEST # 10 - FDST MODE 0 TEST

MACRO M1113 28-MAR-81 12:56 PAGE 32-3

F 5

SEQUENCE 57

:SEE IF THE USER HAS EXPRESSED  
:THE DESIRE TO CHANGE THE SOFTWARE  
:VIRTUAL CONSOLE SWITCH REGISTER (HAS  
:THE USER TYPED CONTROL G?).

2674  
2675  
2808



2809

```
.SBTTL TEST # 11 - ACCUMULATORS DATA PATTERNS TEST
:*****
:*TEST 11 ACCUMULATORS DATA PATTERNS TEST
:*
:*THIS IS A TEST OF THE FLOATING POINT PROCESSOR ACCUMULATORS.
:*EACH ACCUMULATOR IS TESTED IN TWO WAYS:
:* 1 TEST PATTERN GENERATED BY FLOATING A ONE ACROSS
:* A FIELD OF ZEROES.
:* 2 TEST PATTERN GENERATED BY FLOATING A ZERO ACROSS
:* A FIELD OF ONES.
:*EACH OF ACCUMULATORS AC0 THROUGH AC5 IS TESTED.
:*
:*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.
:*
:*
:*THE FOLLOWING PROCEDURE IS PRESENTED TO AID THE TROUBLE
:*SHOOTER IN SITUATIONS WHERE AM2901 CHIP ISOLATION IS ATTEMPTED.
:*
:*WARNING: THIS PROCEDURE ASSUMES THAT THE FAULT IS IN ONE OF THE
:*AM2901 CHIPS. THIS ASSUMPTION IS NOT NECESSARILY VALID IN ALL
:*SITUATIONS. IT REMAINS TO BE SEEN WHAT NUMBER OF FAILURES CAN
:*PROBABLILISTICALLY ASSOCIATED WITH THEM. NOTE ALSO THAT THIS
:*INFORMATION SHOULD NOT BE TAKEN AS ABSOLUTE, THAT IS
:*THIS INFORMATION IS THE AUTHOR'S SUGGESTION FOR ACHIEVING ISOLATION
:*WHEN CHIP LEVEL REPAIR IS NECESSARY.
:*
:*WHEN THIS TEST HAS FINISHED RUNNING, IF ERRORS HAVE OCCURRED,
:*AN ERROR SUMMARY WILL BE TYPED. THIS SUMMARY WILL CONSIST OF TWO
:*IMPORTANT QUANTITIES:
:* A. FOUR SIXTEEN BIT NUMBERS LABELED THE LOGICAL 'AND' ('*')
:* OF THE FAILING DATA PATTERNS.
:* B. FOUR SIXTEEN BIT NUMBERS LABELED THE LOGICAL 'OR' ('+')
:* OF THE FAILING DATA PATTERNS.
:*
:*A BIT STUCK HIGH IN THE HARDWARE WILL SHOW UP AS A 0 IN THAT
:*BIT POSITION OF THE 'OR' OF THE FAILING DATA PATTERNS.
:*
:*A BIT STUCK LOW IN THE HARDWARE WILL SHOW UP AS A 1 IN THAT BIT
:*POSITION OF THE 'AND' OF THE FAILING DATA PATTERNS.
:*
:*THUS IF A FAILURE OCCURS:
:* A. STUCK HIGHS WILL SHOW AS 0'S IN THE 'OR' PATTERN.
:* B. STUCK LOWS WILL SHOW AS 1'S IN THE 'AND' PATTERN.
:*IF THE FAILURE IS INTERMITENT THEN THIS PROCEDURE WILL STILL APPLY!!!
:*IF THE FAILURE MOVES FROM ONE BIT TO ANOTHER, OR FROM ONE
:*GROUP OF BITS TO ANOTHER GROUP OF BITS THEN THE FAULT WILL
:*PROBABLY NOT SHOW UP IN THE 'AND' OR THE 'OR' PATTERNS; IN THIS
:*CASE THE 'AND' PATTERN WILL BE ALL 0'S AND THE 'OR' PATTERN WILL
:*BE ALL 1'S. WHEN THIS OCCURS SOME OTHER METHOD OF REPAIR MUST
:*BE FOUND (SUCH AS INSPECTION OF EACH INDIVIDUAL ERROR REPORT
```



\*RATHER THAN USING THE SUMMARY).  
 \*MAP THE FOLLOWING NOTATION ONTO EACH BIT POSITION IN THE 'AND'  
 \*AND THE 'OR' PATTERNS WHICH ARE TYPED IN THE ERROR SUMMARY.  
 \*A15,A14,....A1,A0 B15,B14,....B1,B0 C15,C14,....C1,C0 D15,D14,....D1,D0  
 \*IN THIS NOTATION A15 THROUGH A0 IS THE FIRST OF THE FOUR 16 BIT  
 \*OCTAL NUMBERS TYPED, B15 THROUGH B0 IS THE SECOND, ETC.  
 \*THIS TABLE SHOWS THE CORRESPONDING AM2901 CHIP ('E' NUMBER)  
 \*WHICH IS RESPONSIBLE FOR EACH BIT POSITION USING THE ABOVE  
 \*NOTATION. NOTE THAT ECO'S TO THE HARDWARE MIGHT MAKE THIS  
 \*TABLE OBSOLETE IF IT IS NOT UPDATED. NOTE ALSO THAT THERE ARE  
 \*FOUR BITS FOR EACH AM2901 CHIP:

BITS	AM2901 CHIP NUMBER
A15,A14,A13,A12	E37
A11,A10,A9,A8	E45
A7,A6,A5,A4	E34
A3,A2,A1,A0	E42
B15,B14,B13,B12	E33
B11,B10,B9,B8	E41
B7,B6,B5,B4	E36
B3,B2,B1,B0	E44
C15,C14,C13,C12	E35
C11,C10,C9,C8	E43
C7,C6,C5,C4	E38
C3,C2,C1,C0	E46
D15,D14,D13,D12	E39
D11,D10,D9,D8	E47
D7,D6,D5,D4	E40
D3,D2,D1,D0	E48

\*NOW FIVE IMPORTANT CASES WHICH WILL ARISE WHEN A FAULTY  
 \*AM2901 IS PRESENT CAN BE DESCRIBED:  
 \*1.) IF ONLY ONE BIT OF THE 64 BITS IS INCORRECT 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.
* AND 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
* 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, AN+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 ABOVE PATTERNS, THEN THE TROUBLE SHOOTER MUST
* INTUITIVELY TRY TO FIND WHICH OF THE ABOVE CASES (1 THROUGH 4)
* IS A 'BEST FIT' OF THE SYMPTOMS.
    
```

```

011752 000004
2810 011754 170011
2811
011756 012737 044067 001244
011764 012737 012020 001236
011772 012700 014760
011776 012701 015020
012002 012737 012020 001110
012010 004737 014112
012014 012703 000102
012020 172410
012022 174000
012024 172400
012026 174011
012030 004737 014210
012034 005737 014754
012040 001004
012042 005137 014754
012046 000261
012050 000401
012052 000241
012054 006160 000006
012060 006160 000004
012064 006160 000002
012070 006110
012072 004737 014170
    
```

```

*****
TST11: SCOPE
        SETD                ;SET FD.
;TEST ACCUMULATOR 0 WITH FLOATING ONE
        MOV #MNUM0,$TMP5
        MOV #G1,$TMP2
        MOV #GPAT00,R0
        MOV #GDAT00,R1
        MOV #G1,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
        JSR PC,GSETUP       ;LOAD TEST PATTERN.
        MOV #102,R3
G1:     LDD (R0),ACO
        STD ACO,ACO
        LDD ACO,ACO         ;STORE THE TEST PATTERN.
        STD ACO,(R1)
        JSR PC,GCMP        ;COMPARE THE DATA READ WITH
                           ;THAT WHICH WAS WRITTEN.
        TST GFLAG1
        BNE G2
        COM GFLAG1
        SEC
        BR G3
G2:     CLC
G3:     ROL 6(R0)           ;GENERATE THE NEXT TEST PATTERN.
        ROL 4(R0)
        ROL 2(R0)
        ROL (R0)
        JSR PC,GRESET     ;RESET DEFAULT PATTERN IN OUTPUT
                           ;BUFFER.
    
```



```

2812 012076 07733J SOB R3,G1
012100 004737 014322 JSR PC,GSUM ;TYPE ERROR SUMMARY.
;TEST ACCUMULATOR 0 WITH FLOATING ZERO
012104 012737 044067 001244 MOV #MNUM0,$TMP5
012112 012737 012146 001236 MOV #G4,$TMP2
012120 012700 014770 MOV #GPAT10,R0
012124 012701 015020 MOV #GDAT00,R1
012130 012737 012146 001110 MOV #G4,$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
012136 004737 014112 JSR PC,GSETUP ;LOAD TEST PATTERN.
012142 012703 000102 MOV #102,R3
012146 172410 G4: LDD (R0),ACO
012150 174000 STD ACO,ACO
012152 172400 LDD ACO,ACO ;STORE THE TEST PATTERN.
012154 174011 STD ACO,(R1)
012156 004737 014210 JSR PC,GCMP ;COMPARE THE DATA READ WITH
;THAT WHICH WAS WRITTEN.

012162 005737 014754 TST GFLAG1
012166 001004 BNE G5
012170 005137 014754 COM GFLAG1
012174 000241 CLC
012176 000401 BR G6
012200 000261 G5: SEC
012202 006160 000006 G6: ROL 6(R0) ;GENERATE THE NEXT TEST PATTERN.
012206 006160 000004 ROL 4(R0)
012212 006160 000002 ROL 2(R0)
012216 006110 ROL (R0)
012220 004737 014170 JSR PC,GRESET ;RESET DEFAULT PATTERN IN OUTPUT
;BUFFER.

2813 012224 07733J SOB R3,G4
012226 004737 014322 JSR PC,GSUM ;TYPE ERROR SUMMARY.
;TEST ACCUMULATOR 1 WITH FLOATING ONE
012232 012737 044075 001244 MOV #MNUM1,$TMP5
012240 012737 012274 001236 MOV #G7,$TMP2
012246 012700 014760 MOV #GPAT00,R0
012252 012701 015020 MOV #GDAT00,R1
012256 012737 012274 001110 MOV #G7,$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
012264 004737 014112 JSR PC,GSETUP ;LOAD TEST PATTERN.
012270 012703 000102 MOV #102,R3
012274 172410 G7: LDD (R0),ACO
012276 174001 STD ACO,AC1
012300 172401 LDD AC1,ACO ;STORE THE TEST PATTERN.
012302 174011 STD ACO,(R1)
012304 004737 014210 JSR PC,GCMP ;COMPARE THE DATA READ WITH
;THAT WHICH WAS WRITTEN.

012310 005737 014754 TST GFLAG1
012314 001004 BNE G10
012316 005137 014754 COM GFLAG1
012322 000261 SEC
012324 000401 BR G11
012326 000241 G10: CLC
012330 006160 000006 G11: ROL 6(R0) ;GENERATE THE NEXT TEST PATTERN.
012334 006160 000004 ROL 4(R0)
012340 006160 000002 ROL 2(R0)
012344 006110 ROL (R0)
012346 004737 014170 JSR PC,GRESET ;RESET DEFAULT PATTERN IN OUTPUT
;BUFFER.

012352 077330 SOB R3,G7
    
```



```

2814 012354 004737 014322          JSR    PC,GSUM          ;TYPE ERROR SUMMARY.
          ;TEST ACCUMULATOR 1 WITH FLOATING ZERO
012360 012737 044075 001244      MOV    #MNUM1,$TMP5
012366 012737 012422 001236      MOV    #G12,$TMP2
012374 012700 014770              MOV    #GPAT10,R0
012400 012701 015020              MOV    #GDAT00,R1
012404 012737 012422 001110      MOV    #G12,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
012412 004737 014112              JSR    PC,GSETUP        ;LOAD TEST PATTERN.
012416 012703 000102              MOV    #102,R3
012422 172410          G12:      LDD    (R0),AC0
012424 174001              STD    AC0,AC1
012426 172401              LDD    AC1,AC0          ;STORE THE TEST PATTERN.
012430 174011              STD    AC0,(R1)
012432 004737 014210          JSR    PC,GCMP          ;COMPARE THE DATA READ WITH
          ;THAT WHICH WAS WRITTEN.

012436 005737 014754              TST    GFLAG1
012442 001004              BNE    G13
012444 005137 014754              COM    GFLAG1
012450 000241              CLC
012452 000401              BR    G14
012454 000261          G13:      SEC
012456 006160 000006          G14:      ROL    6(R0)      ;GENERATE THE NEXT TEST PATTERN.
012462 006160 000004              ROL    4(R0)
012466 006160 000002              ROL    2(R0)
012472 006110              ROL    (R0)
012474 004737 014170          JSR    PC,GRESET      ;RESET DEFAULT PATTERN IN OUTPUT
          ;BUFFER.

012500 077330          SOB    R3,G12
2815 012502 004737 014322          JSR    PC,GSUM          ;TYPE ERROR SUMMARY.
          ;TEST ACCUMULATOR 2 WITH FLOATING ONE
012506 012737 044102 001244      MOV    #MNUM2,$TMP5
012514 012737 012550 001236      MOV    #G15,$TMP2
012522 012700 014760              MOV    #GPAT00,R0
012526 012701 015020              MOV    #GDAT00,R1
012532 012737 012550 001110      MOV    #G15,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
012540 004737 014112              JSR    PC,GSETUP        ;LOAD TEST PATTERN.
012544 012703 000102              MOV    #102,R3
012550 172410          G15:      LDD    (R0),AC0
012552 174002              STD    AC0,AC2
012554 172402              LDD    AC2,AC0          ;STORE THE TEST PATTERN.
012556 174011              STD    AC0,(R1)
012560 004737 014210          JSR    PC,GCMP          ;COMPARE THE DATA READ WITH
          ;THAT WHICH WAS WRITTEN.

012564 005737 014754              TST    GFLAG1
012570 001004              BNE    G16
012572 005137 014754              COM    GFLAG1
012576 000261              SEC
012600 000401              BR    G17
012602 000241          G16:      CLC
012604 006160 000006          G17:      ROL    6(R0)      ;GENERATE THE NEXT TEST PATTERN.
012610 006160 000004              ROL    4(R0)
012614 006160 000002              ROL    2(R0)
012620 006110              ROL    (R0)
012622 004737 014170          JSR    PC,GRESET      ;RESET DEFAULT PATTERN IN OUTPUT
          ;BUFFER.

012626 077330          SOB    R3,G15
012630 004737 014322          JSR    PC,GSUM          ;TYPE ERROR SUMMARY.
    
```



```

2816 ;TEST ACCUMULATOR 2 WITH FLOATING ZERO
012634 012737 044102 001244 MOV #MNUM2,$TMP5
012642 012737 012676 001236 MOV #G20,$TMP2
012650 012700 014770 MOV #GPAT10,R0
012654 012701 015020 MOV #GDAT00,R1
012660 012737 012676 001110 MOV #G20,$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
012666 004737 014112 JSR PC,GSETUP ;LOAD TEST PATTERN.
012672 012703 000102 MOV #102,R3
012676 172410 G20: LDD (R0),AC0
012700 174002 STD AC0,AC2 ;STORE THE TEST PATTERN.
012702 172402 LDD AC2,AC0
012704 174011 STD AC0,(R1)
012706 004737 014210 JSR PC,GCMP ;COMPARE THE DATA READ WITH
;THAT WHICH WAS WRITTEN.

012712 005737 014754 TST GFLAG1
012716 001004 BNE G21
012720 005137 014754 COM GFLAG1
012724 000241 CLC
012726 000401 BR G22
012730 000261 G21: SEC
012732 006160 000006 G22: ROL 6(R0) ;GENERATE THE NEXT TEST PATTERN.
012736 006160 000004 ROL 4(R0)
012742 006160 000002 ROL 2(R0)
012746 006110 ROL (R0)
012750 004737 014170 JSR PC,GRESET ;RESET DEFAULT PATTERN IN OUTPUT
;BUFFER.

012754 077330 SOB R3,G20
012756 004737 014322 JSR PC,GSUM ;TYPE ERROR SUMMARY.

2817 ;TEST ACCUMULATOR 3 WITH FLOATING ONE
012762 012737 044107 001244 MOV #MNUM3,$TMP5
012770 012737 013024 001236 MOV #G23,$TMP2
012776 012700 014760 MOV #GPAT00,R0
013002 012701 015020 MOV #GDAT00,R1
013006 012737 013024 001110 MOV #G23,$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
013014 004737 014112 JSR PC,GSETUP ;LOAD TEST PATTERN.
013020 012703 000102 MOV #102,R3
013024 172410 G23: LDD (R0),AC0
013026 174003 STD AC0,AC3 ;STORE THE TEST PATTERN.
013030 172403 LDD AC3,AC0
013032 174011 STD AC0,(R1)
013034 004737 014210 JSR PC,GCMP ;COMPARE THE DATA READ WITH
;THAT WHICH WAS WRITTEN.

013040 005737 014754 TST GFLAG1
013044 001004 BNE G24
013046 005137 014754 COM GFLAG1
013052 000261 SEC
013054 000401 BR G25
013056 000241 G24: CLC
013060 006160 000006 G25: ROL 6(R0) ;GENERATE THE NEXT TEST PATTERN.
013064 006160 000004 ROL 4(R0)
013070 006160 000002 ROL 2(R0)
013074 006110 ROL (R0)
013076 004737 014170 JSR PC,GRESET ;RESET DEFAULT PATTERN IN OUTPUT
;BUFFER.

013102 077330 SOB R3,G23
013104 004737 014322 JSR PC,GSUM ;TYPE ERROR SUMMARY.

2818 ;TEST ACCUMULATOR 3 WITH FLOATING ZERO
    
```



```

013110 012737 044107 001244      MOV      #MNUM3,$TMP5
013116 012737 013152 001236      MOV      #G26,$TMP2
013124 012700 014770              MOV      #GPAT10,R0
013130 012701 015020              MOV      #GDAT00,R1
013134 012737 013152 001110      MOV      #G26,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
013142 004737 014112              JSR      PC,GSETUP        ;LOAD TEST PATTERN.
013146 012703 000102              MOV      #102,R3
013152 172410              G26:    LDD      (R0),AC0
013154 174003              STD      AC0,AC3
013156 172403              LDD      AC3,AC0          ;STORE THE TEST PATTERN.
013160 174011              STD      AC0,(R1)
013162 004737 014210              JSR      PC,GCMP          ;COMPARE THE DATA READ WITH
                                ;THAT WHICH WAS WRITTEN.

013166 005737 014754              TST      GFLAG1
013172 001004              BNE      G27
013174 005137 014754              COM      GFLAG1
013200 000241              CLC
013202 000401              BR       G30
013204 000261              G27:    SEC
G30:    ROL      6(R0)          ;GENERATE THE NEXT TEST PATTERN.
013206 006160 000006              ROL      4(R0)
013212 006160 000004              ROL      2(R0)
013216 006160 000002              ROL      (R0)
013222 006110              JSR      PC,GRESET        ;RESET DEFAULT PATTERN IN OUTPUT
013224 004737 014170              ;BUFFER.

013230 077330              SOB      R3,G26
013232 004737 014322              JSR      PC,GSUM          ;TYPE ERROR SUMMARY.
                                ;TEST ACCUMULATOR 4 WITH FLOATING ONE
2819 013236 012737 044116 001244      MOV      #MNUM4,$TMP5
013244 012737 013300 001236      MOV      #G31,$TMP2
013252 012700 014760              MOV      #GPAT00,R0
013256 012701 015020              MOV      #GDAT00,R1
013262 012737 013300 001110      MOV      #G31,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
013270 004737 014112              JSR      PC,GSETUP        ;LOAD TEST PATTERN.
013274 012703 000102              MOV      #102,R3
013300 172410              G31:    LDD      (R0),AC0
013302 174004              STD      AC0,AC4
013304 172404              LDD      AC4,AC0          ;STORE THE TEST PATTERN.
013306 174011              STD      AC0,(R1)
013310 004737 014210              JSR      PC,GCMP          ;COMPARE THE DATA READ WITH
                                ;THAT WHICH WAS WRITTEN.

C1 14 013314 005737 014754              TST      GFLAG1
013320 001004              BNE      G32
013322 005137 014754              COM      GFLAG1
013326 000261              SEC
013330 000401              BR       G33
013332 000241              G32:    CLC
G33:    ROL      6(R0)          ;GENERATE THE NEXT TEST PATTERN.
013334 006160 000006              ROL      4(R0)
013340 006160 000004              ROL      2(R0)
013344 006160 000002              ROL      (R0)
013350 006110              JSR      PC,GRESET        ;RESET DEFAULT PATTERN IN OUTPUT
013352 004737 014170              ;BUFFER.

013356 077330              SOB      R3,G31
013360 004737 014322              JSR      PC,GSUM          ;TYPE ERROR SUMMARY.
                                ;TEST ACCUMULATOR 4 WITH FLOATING ZERO
2820 013364 012737 044116 001244      MOV      #MNUM4,$TMP5
    
```



```

013372 012737 013426 001236      MOV      #G34,$TMP2
013400 012700 014770      MOV      #GPAT10,R0
013404 012701 015020      MOV      #GDAT00,R1
013410 012737 013426 001110      MOV      #G34,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
013416 004737 014112      JSR      PC,GSETUP        ;LOAD TEST PATTERN.
013422 012703 000102      MOV      #102,R3
013426 172410      LDD      (R0),AC0      G34:
013430 174004      STD      AC0,AC4
013432 172404      LDD      AC4,AC0      ;STORE THE TEST PATTERN.
013434 174011      STD      AC0,(R1)
013436 004737 014210      JSR      PC,GCMP        ;COMPARE THE DATA READ WITH
                                ;THAT WHICH WAS WRITTEN.

013442 005737 014754      TST      GFLAG1
013446 001004      BNE     G35
013450 005137 014754      COM     GFLAG1
013454 000241      CLC
013456 000401      BR     G36
013460 000261      SEC      G35:
013462 006160 000006      ROL     6(R0)          G36:
013466 006160 000004      ROL     4(R0)          ;GENERATE THE NEXT TEST PATTERN.
013472 006160 000002      ROL     2(R0)
013476 006110      ROL     (R0)
013500 004737 014170      JSR      PC,GRESET      ;RESET DEFAULT PATTERN IN OUTPUT
                                ;BUFFER.

013504 077330      SOB     R3,G34
013506 004737 014322      JSR      PC,GSUM        ;TYPE ERROR SUMMARY.
2821 ;TEST ACCUMULATOR 5 WITH FLOATING ONE
013512 012737 044124 001244      MOV      #MNUM5,$TMP5
013520 012737 013554 001236      MOV      #G37,$TMP2
013526 012700 014760      MOV      #GPAT00,R0
013532 012701 015020      MOV      #GDAT00,R1
013536 012737 013554 001110      MOV      #G37,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
013544 004737 014112      JSR      PC,GSETUP        ;LOAD TEST PATTERN.
013550 012703 000102      MOV      #102,R3
013554 172410      LDD      (R0),AC0      G37:
013556 174005      STD      AC0,AC5
013560 172405      LDD      AC5,AC0      ;STORE THE TEST PATTERN.
013562 174011      STD      AC0,(R1)
013564 004737 014210      JSR      PC,GCMP        ;COMPARE THE DATA READ WITH
                                ;THAT WHICH WAS WRITTEN.

013570 005737 014754      TST      GFLAG1
013574 001004      BNE     G40
013576 005137 014754      COM     GFLAG1
013602 000261      SEC
013604 000401      BR     G41
013606 000241      CLC      G40:
013610 006160 000006      ROL     6(R0)          G41:
013614 006160 000004      ROL     4(R0)          ;GENERATE THE NEXT TEST PATTERN.
013620 006160 000002      ROL     2(R0)
013624 006110      ROL     (R0)
013626 004737 014170      JSR      PC,GRESET      ;RESET DEFAULT PATTERN IN OUTPUT
                                ;BUFFER.

013632 077330      SOB     R3,G37
013634 004737 014322      JSR      PC,GSUM        ;TYPE ERROR SUMMARY.
2822 ;TEST ACCUMULATOR 5 WITH FLOATING ZERO
013640 012737 044124 001244      MOV      #MNUM5,$TMP5
013646 012737 013702 001236      MOV      #G42,$TMP2
    
```



```

013654 012700 014770      MOV      #GPAT10,R0
013660 012701 015020      MOV      #GDAT00,R1
013664 012737 013702 001110  MOV      #G42,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
013672 004737 014112      JSR      PC,GSETUP        ;LOAD TEST PATTERN.
013676 012703 000102      MOV      #102,R3
013702 172410      G42:    LDD      (R0),AC0
013704 174005      STD      AC0,AC5
013706 172405      LDD      AC5,AC0      ;STORE THE TEST PATTERN.
013710 174011      STD      AC0,(R1)
013712 004737 014210      JSR      PC,GCMP        ;COMPARE THE DATA READ WITH
                                ;THAT WHICH WAS WRITTEN.

013716 005737 014754      TST      GFLAG1
013722 001004      BNE      G43
013724 005137 014754      COM      GFLAG1
013730 000241      CLC
013732 000401      BR      G44
013734 000261      G43:    SEC
013736 006160 000006      G44:    ROL      6(R0)      ;GENERATE THE NEXT TEST PATTERN.
013742 006160 000004      ROL      4(R0)
013746 006160 000002      ROL      2(R0)
013752 006110      ROL      (R0)
013754 004737 014170      JSR      PC,GRESET      ;RESET DEFAULT PATTERN IN OUTPUT
                                ;BUFFER.

013760 077330      SOB      R3,G42
013762 004737 014322      JSR      PC,GSUM        ;TYPE ERROR SUMMARY.

2823
2824 013766 000137 015032      JMP      GDONE
2825
2826
2827
2828
2829
2830
2831
2832 013772      105      063      067
2833
2834 014072 170000 007400 000360
2835
2836 014112 012705 014754
2837 014116 012704 000026
2838 014122 005025
2839 014124 077402
2840
2841 014126 012705 014770
2842 014132 012704 000010
2843 014136 005125
2844 014140 077402
2845
2846 014142 020037 014760
2847 014146 001401
2848 014150 000207
2849
2850 014152 012705 015020
2851 014156 012704 000004
2852 014162 005125
2853 014164 077402
2854 014166 000207

*****IMPORTANT*****
* IN CASE OF AN ECO OR CHIP NUMBER CHANGE, CHANGE THE 'CHPNUM' ASCII
* STRING AT THE APPROPRIATE LOCATIONS.
*****
A15 A11 A07 A03 B15 B11 B07 B03 C15 C11 C07 C03 D15 D11 D07 D03
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |
A12 A08 A04 A00 B12 B08 B04 B00 C12 C08 C04 C00 D12 D08 D04 D00
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |
CHPNUM: .ASCII ?E37 E45 E34 E42 E33 E41 E36 E44 E35 E43 E38 E46 E39 E47 E40 E48?
.EVEN
BITSTS: .WORD 170000,7400,360,17,7777,170377,177417,177760
;USE THIS ROUTINE TO INITIALIZE ALL THE DATA BUFFERS.
GSETUP: MOV      #GFLAG1,R5
MOV      #26,R4
1$:     CLR      (R5)+
SOB      R4,1$

MOV      #GPAT10,R5
MOV      #10,R4
2$:     COM      (R5)+
SOB      R4,2$

GS1:    CMP      R0,GPAT00
BEQ      3$
RTS      PC

3$:     MOV      #GDAT00,R5
MOV      #4,R4
4$:     COM      (R5)+
SOB      R4,4$
RTS      PC
    
```



```

2855
2856 014170 012705 015020      GRESET: MOV      #GDAT00,R5
2857 014174 012704 000004      MOV      #4,R4
2858 014200 005025      1$:      CLR      (R5)+
2859 014202 077402      SOB      R4,1$
2860 014204 000137 014142      JMP      GS1
2861
2862      ;SEE IF THE DATA WRITTEN MATCHES THE DATA READ.
2863 014210 012705 015020      GCMP:   MOV      #GDAT00,R5
2864 014214 012704 000004      MOV      #4,R4
2865 014220 010002      MOV      R0,R2
2866 014222 022225      1$:     CMP      (R2)+,(R5)+
2867 014224 001402      BEQ      2$
2868 014226 000137 014236      JMP      GERR1
2869 014232 077405      2$:     SOB      R4,1$
2870 014234 000207      RTS      PC
2871
2872      ;COME HERE TO REPORT AND RECORD ERRORS.
2873 014236 012637 015030      GERR1: MOV      (SP)+,GADR      ;SAVE THE RETURN ADDRESS.
2874 014242 010037 001240      MOV      R0,1240      ;COMPUTE 'OR' OF BAD DATA.
2875 014246 012705 015000      MOV      #GAND0,R5
2876 014252 012704 000004      MOV      #4,R4
2877 014256 051065 000010      1$:     BIS      (R0),10(R5)
2878 014262 012002      MOV      (R0)+,R2
2879 014264 005102      COM      R2
2880 014266 040225      BIC      R2,(R5)+
2881 014270 077406      SOB      R4,1$
2882 014272 013700 001240      MOV      1240,R0
2883 014276 005237 014756      INC      GFLAG2      ;INCREMENT ERROR COUNT.
2884 014302 010037 001240      MOV      P?,$TMP3
2885 014306 012737 015020 001242      MOV      #GDAT00,$TMP4
2886 014314 104044      3$:     ERROR   +44
2887 014316 000177 000506      JMP      @GADR
2888
2889      ;SEE IF ANY ERRORS HAVE OCCURRED AND WHETHER OR NOT AN ERROR SUMMARY
2890      ;SHOULD BE TYPED.
2891 014322 005737 014756      GSUM:   TST      GFLAG2      ;ANY ERRORS?
2892 014326 001410      BEQ      100$      ;BRANCH IF NOT
2893 014330 032777 020000 164602      BIT      #SW13,@SWR      ;INHIBIT ERROR PRINT OUT?
2894 014336 001405      BEQ      1$      ;BRANCH IF NOT INHIBITED
2895 014340 032777 000200 164572      BIT      #SW7,@SWR      ;PRINT SUMMARY?
2896 014346 001001      BNE      1$      ;BRANCH IF NOT
2897 014350 000207      100$:   RTS      PC      ;EXIT - NO ERRORS TO REPORT
2898 014352 013737 014756 001246      1$:     MOV      GFLAG2,$TMP6      ;YES PRINT SUMMARY.
2899 014360 012737 015000 001240      MOV      #GAND0,$TMP3
2900 014366 012737 015010 001242      MOV      #GOR0,$TMP4
2901 014374 012637 015030      MOV      (SP)+,GADR      ;SAVE RETURN ADDRESS FOR POSSIBLE LOOPING
2902 014400 012737 014414 001116      MOV      #2,$ERRPC
2903 014406 112737 000045 001114      MOV      #45,$ITEMB
2904 014414 004737 041342      2$:     JSR      PC,ERTYPE
2905 014420 010046      MOV      R0,-(SP)      ;SAVE R0
2906 014422 010146      MOV      R1,-(SP)      ;SAVE R1
2907 014424 010246      MOV      R2,-(SP)      ;SAVE R2
2908 014426 012700 013772      MOV      #CHPNUM,R0      ;MOVE ADDRESS OF CHIP NUMBER ASCII'S TO R0
2909 014432 012702 015000      MOV      #GAND0,R2      ;MOVE ADDRESS OF 'AND' DATA TO R2
2910 014436 112737 000077 014746      MOV      #'?,11$      ;MOVE ASCII '?' TO NEXT 3 LOCATIONS
2911 014444 112737 000077 014747      MOV      #'?,11$+1
    
```



```

2912 014452 112737 000077 014750      MOVB    #'?.11$+2
2913 014460 104401 014746      TYPE    .11$
2914 014464 012701 014072      3$:    MOV    #BITSTS,R1
2915 014470 032112      4$:    BIT    (R1)+,(R2)
2916 014472 001413      BEQ    5$
2917 014474 116037 000000 014746      MOVB    0(R0),11$+0
2918 014502 116037 000001 014747      MOVB    1(R0),11$+1
2919 014510 116037 000002 014750      MOVB    2(R0),11$+2
2920 014516 104401 014746      TYPE    .11$
2921 014522 062700 000004      5$:    ADD    #4,R0
2922 014526 022701 014102      CMP    #BITSTS+10,R1
2923 014532 001356      BNE    4$
2924 014534 062702 000002      ADD    #2,R2
2925 014540 022702 015010      CMP    #GAND0+10,R2
2926 014544 001347      BNE    3$
2927 014546 012700 013772      MOV    #CHPNUM,R0
2928 014552 012702 015010      MOV    #GORO,R2
2929 014556 011246      6$:    MOV    (R2),-(SP)
2930 014560 046116 000010      BIC    10(R1),(SP)
2931 014564 022126      CMP    (R1)+,(SP)+
2932 014566 001413      BEQ    7$
2933 014570 116037 000000 014746      MOVB    0(R0),11$
2934 014576 116037 000001 014747      MOVB    1(R0),11$+1
2935 014604 116037 000002 014750      MOVB    2(R0),11$+2
2936 014612 104401 014746      TYPE    .11$
2937 014616 062700 000004      7$:    ADD    #4,R0
2938 014622 022701 014102      CMP    #BITSTS+10,R1
2939 014626 001353      BNE    6$
2940 014630 012701 014072      MOV    #BITSTS,R1
2941 014634 062702 000002      ADD    #2,R2
2942 014640 022702 015010      CMP    #GAND0+10,R2
2943 014644 001344      BNE    6$
2944 014646 122737 000077 014746      CMPB   #'?.11$
2945 014654 001002      BNE    8$
2946 014656 104401 014746      TYPE    .11$
2947 014662 104401 001313      8$:    TYPE    .$CRLF
2948 014666 012602      MOV    (SP)+,R2
2949 014670 012601      MOV    (SP)+,R1
2950 014672 012600      MOV    (SP)+,R0
2951 014674 000177 000130      JMP    @GADR
2952 014700 000207      9$:    RTS    PC
2953
2954 014702      200      103      110      10$:    .ASCIZ <CRLF>'CHIP NUMBERS TO INITIALLY LOOK AT'<CRLF>
2955 014746      077      077      077      11$:    .ASCIZ '???'
2956
2957 014754 000000      GFLAG1: .WORD 0
2958 014756 000000      GFLAG2: .WORD 0
2959 014760 000000      GPAT00: .WORD 0
2960 014762 000000      GPAT01: .WORD 0
2961 014764 000000      GPAT02: .WORD 0
2962 014766 000000      GPAT03: .WORD 0
2963 014770 177777      GPAT10: .WORD -1
2964 014772 177777      GPAT11: .WORD -1
2965 014774 177777      GPAT12: .WORD -1
2966 014776 177777      GPAT13: .WORD -1
2967 015000 177777      GAND0:  .WORD -1
2968 015002 177777      GAND1:  .WORD -1
    
```



2969	015004	177777	GAND2:	.WORD	-1
2970	015006	177777	GAND3:	.WORD	-1
2971	015010	000000	GOR0:	.WORD	0
2972	015012	000000	GOR1:	.WORD	0
2973	015014	000000	GOR2:	.WORD	0
2974	015016	000000	GOR3:	.WORD	0
2975	015020	000000	GDATA0:	.WORD	0
2976	015022	000000	GDATA1:	.WORD	0
2977	015024	000000	GDATA2:	.WORD	0
2978	015026	000000	GDATA3:	.WORD	0
2979	015030	000000	GADR:	.WORD	0
2980	015032		GDONE:		
	015032	104413		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?).



2987

```

.SBTTL TEST # 12 - FPP ACCUMULATORS DUAL ADDRESS TEST
*****
: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.
:
*****
    
```

```

2988 015034 000004
2989 015036 012737 015044 001110
2990 015044 005037 015570
2991 015050 012700 015572
2992 015054 012701 015712
2993 015060 012703 000024
2994 015064 012120
2995 015066 077302
2996
2997 015070 004737 015516
2998
2999 015074 170011
3000
3001 015076 012700 015572
3002 015102 172410
3003 015104 174001
3004 015106 012700 015602
3005 015112 172410
3006 015114 174002
3007 015116 012700 015612
3008 015122 172410
3009 015124 174003
3010 015126 012700 015622
3011 015132 172410
3012 015134 174004
3013 015136 012700 015632
3014 015142 172410
3015 015144 174005
3016 015146 004737 015402
3017 015152 004737 015460
3018
3019
3020
3021
3022
3023
3024
3025
3026
3027
3028
3029
3030
3031
3032
3033
3034
3035
3036
3037
3038
3039
3040
3041
3042
3043
3044
3045
3046
3047
3048
3049
3050
3051
3052
3053
3054
3055
3056
3057
3058
3059
3060
3061
3062
3063
3064
3065
3066
3067
3068
3069
3070
3071
3072
3073
3074
3075
3076
3077
3078
3079
3080
3081
3082
3083
3084
3085
3086
3087
3088
3089
3090
3091
3092
3093
3094
3095
3096
3097
3098
3099
3100
    
```

```

TST12: SCOPE
MOV #H1,$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.

H1: CLR HFLAG
MOV #HA1W,R0 ;INITIALIZE THE LOAD BUFFER DATA.
MOV #HDAT1,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),ACO
STD ACO,AC1
:LOAD ACCUMULATOR 2
MOV #HA2W,R0
LDD (R0),ACO
STD ACO,AC2
:LOAD ACCUMULATOR 3
MOV #HA3W,R0
LDD (R0),ACO
STD ACO,AC3
:LOAD ACCUMULATOR 4
MOV #HA4W,R0
LDD (R0),ACO
STD ACO,AC4
:LOAD ACCUMULATOR 5
MOV #HA5W,R0
LDD (R0),ACO
STD ACO,AC5

H4: JSR PC,HSTD ;GO READ ALL ACCUMULATORS BACK.
JSR PC,HCMP ;SEE IF DATA IS CORRECT.

:COMPLIMENT EACH WORD OF THE DATA STORED IN ACCUMULATOR 1,
:RELOAD THAT ACCUMULATOR, READ ALL THE ACCUMULATORS BACK AND CHECK
:THE DATA.
MOV #HA1W,R0
MOV #4,R2
MOV R0,R1
H5: COM (R1)+
LDD (R0),ACO
STD ACO,AC1
JSR PC,HSTD ;READ ALL THE ACCUMULATORS BACK.
JSR PC,HCMP ;CHECK THE DATA.
    
```



3011 015206 077210

SOB R2,H5  
 :COMPLIMENT EACH WORD OF THE DATA STORED IN ACCUMULATOR 2,  
 :RELOAD THAT ACCUMULATOR, READ ALL THE ACCUMULATORS BACK AND CHECK  
 :THE DATA.

015210 012700 015602  
 015214 012702 000004  
 015220 010001  
 015222 005121  
 015224 172410  
 015226 174002  
 015230 004737 015402  
 015234 004737 015460  
 015240 077210

MOV #HA2W,R0  
 MOV #4,R2  
 MOV R0,R1  
 H6: COM (R1)+  
 LDD (R0),AC0  
 STD AC0,AC2  
 JSR PC,HSTD :READ ALL THE ACCUMULATORS BACK.  
 JSR PC,HCMP :CHECK THE DATA.  
 SOB R2,H6

3012

:COMPLIMENT EACH WORD OF THE DATA STORED IN ACCUMULATOR 3,  
 :RELOAD THAT ACCUMULATOR, READ ALL THE ACCUMULATORS BACK AND CHECK  
 :THE DATA.

015242 012700 015612  
 015246 012702 000004  
 015252 010001  
 015254 005121  
 015256 172410  
 015260 174003  
 015262 004737 015402  
 015266 004737 015460  
 015272 077210

MOV #HA3W,RC  
 MOV #4,R2  
 MOV R0,R1  
 H7: COM (R1)+  
 LDD (R0),AC0  
 STD AC0,AC3  
 JSR PC,HSTD :READ ALL THE ACCUMULATORS BACK.  
 JSR PC,HCMP :CHECK THE DATA.  
 SOB R2,H7

3013

:COMPLIMENT EACH WORD OF THE DATA STORED IN ACCUMULATOR 4,  
 :RELOAD THAT ACCUMULATOR, READ ALL THE ACCUMULATORS BACK AND CHECK  
 :THE DATA.

015274 012700 015622  
 015300 012702 000004  
 015304 010001  
 015306 005121  
 015310 172410  
 015312 174004  
 015314 004737 015402  
 015320 004737 015460  
 015324 077210

MOV #HA4W,R0  
 MOV #4,R2  
 MOV R0,R1  
 H10: COM (R1)+  
 LDD (R0),AC0  
 STD AC0,AC4  
 JSR PC,HSTD :READ ALL THE ACCUMULATORS BACK.  
 JSR PC,HCMP :CHECK THE DATA.  
 SOB R2,H10

3014

:COMPLIMENT EACH WORD OF THE DATA STORED IN ACCUMULATOR 5,  
 :RELOAD THAT ACCUMULATOR, READ ALL THE ACCUMULATORS BACK AND CHECK  
 :THE DATA.

015326 012700 015632  
 015332 012702 000004  
 015336 010001  
 015340 005121  
 015342 172410  
 015344 174005  
 015346 004737 015402  
 015352 004737 015460  
 015356 077210

MOV #HA5W,R0  
 MOV #4,R2  
 MOV R0,R1  
 H11: COM (R1)+  
 LDD (R0),AC0  
 STD AC0,AC5  
 JSR PC,HSTD :READ ALL THE ACCUMULATORS BACK.  
 JSR PC,HCMP :CHECK THE DATA.  
 SOB R2,H11

3015

3016 015360 005737 015570

3017 015364 001402

3018 015366 000137 015762

3019

3020 015372 005137 015570

3021 015376 000137 015074

3022

TST HFLAG  
 BEQ H12  
 JMP HDONE  
 H12: COM HFLAG  
 JMP H3



```

3023 ;STORE ALL ACCUMULATORS IN THE OUTPUT BUFFERS.
3024 015402 004737 015516 HSTD: JSR PC,HCLR ;CLEAR ALL OUTPUT BUFFERS.
3025 ;STORE ACCUMULATOR 1
      MOV #HA1R,R4
      LDD AC1,ACO
      STD ACO,(R4)
3026 ;STORE ACCUMULATOR 2
      MOV #HA2R,R4
      LDD AC2,ACO
      STD ACO,(R4)
3027 ;STORE ACCUMULATOR 3
      MOV #HA3R,R4
      LDD AC3,ACO
      STD ACO,(R4)
3028 ;STORE ACCUMULATOR 4
      MOV #HA4R,R4
      LDD AC4,ACO
      STD ACO,(R4)
3029 ;STORE ACCUMULATOR 5
      MOV #HA5R,R4
      LDD AC5,ACO
      STD ACO,(R4)
3030 RTS PC
3031
3032 ;COMPARE DATA LOADED WITH DATA READ.
3033 015460 012637 015566 HCMP: MOV (SP)+,HADR ;SAVE RETURN ADDRESS.
3034 015464 012703 015572      MOV #HA1W,R3
3035 015470 012704 015642      MOV #HA1R,R4
3036 015474 012705 000024      MOV #24,R5
3037 015500 022324 HCMP1: CMP (R3)+,(R4)+
3038 015502 001402      BEQ HCMP2
3039 015504 000137 015534      JMP HERROR
3040 015510 077505 HCMP2: SOB R5,HCMP1
3041 015512 000177 000050      JMP @HADR
3042
3043 ;CLEAR THE DATA OUTPUT BUFFER.
3044 015516 012704 015642 HCLR: MOV #HA1R,R4
3045 015522 012705 000024      MOV #24,R5
3046 015526 005024 HCLR1: CLR (R4)+
3047 015530 077502      SOB R5,HCLR1
3048 015532 000207      RTS PC
3049
3050 ;REPORT ERROR.
3051 015534 HERROR:
3052 015534 012703 015572      MOV #HA1W,R3
3053 015540 012704 001236      MOV #STMP2,R4
3054 015544 012705 000012      MOV #12,R5
3055 015550 010324 1$: MOV R3,(R4)+
3056 015552 062703 000010      ADD #10,R3
3057 015556 077504      SOB R5,1$
3058 015560 104046 2$: ERROR +46
3059 015562 000137 015762      JMP HDONE
3060
3061
3062 015566 000000 HADR: .WORD 0
3063 015570 000000 HFLAG: .WORD 0
3064
    
```



3065	015572	000000	000000	000000	HA1W:	.WORD	0,0,0,0
3066	015602	000000	000000	000000	HA2W:	.WORD	0,0,0,0
3067	015612	000000	000000	000000	HA3W:	.WORD	0,0,0,0
3068	015622	000000	000000	000000	HA4W:	.WORD	0,0,0,0
3069	015632	000000	000000	000000	HA5W:	.WORD	0,0,0,0
3070							
3071	015642	000000	000000	000000	HA1R:	.WORD	0,0,0,0
3072	015652	000000	000000	000000	HA2R:	.WORD	0,0,0,0
3073	015662	000000	000000	000000	HA3R:	.WORD	0,0,0,0
3074	015672	000000	000000	000000	HA4R:	.WORD	0,0,0,0
3075	015702	000000	000000	000000	HA5R:	.WORD	0,0,0,0
3076							
3077	015712	073567	073567	073567	HDATA1:	.WORD	73567,73567,73567,73567
3078	015722	063146	063146	063146	HDATA2:	.WORD	63146,63146,63146,63146
3079	015732	010421	010421	010421	HDATA3:	.WORD	10421,10421,10421,10421
3080	015742	031463	031463	031463	HDATA4:	.WORD	31463,31463,31463,31463
3081	015752	042104	042104	042104	HDATA5:	.WORD	42104,42104,42104,42104
3082	015762				HDONE:		
	015762	104413			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?).

3083  
3084



3092

```
.SBTTL TEST # 13 - FSRC MODE 0 WITH ILLEGAL ACCUMULATOR TEST
:*****
:TEST 13 FSRC MODE 0 WITH ILLEGAL ACCUMULATOR TEST
:*
:*THIS IS A TEST OF FSRC MODE 0 WITH ACCUMULATORS 6 AND 7. USE OF
:*EITHER OF THESE NON-EXISTENT ACCUMULATORS SHOULD RESULT IN A TRAP TO 244
:*WITH FEC=2 (ILLEGAL FPP INSTRUCTION).
:*
:*****
```

```
3093 015764 000004
3093 015766
3094 015766 012737 015774 001110
3094 015774 170011
3095 015776 012700 016506
3096 016002 172410
3097
3098 016004 012737 016206 000244
3099
3100
3101 016012 012700 000001
3102
3103 016016 012737 016416 000004
3104 016024 005003
3105
3106 016026 172407
3107 016030 170000
3108 016032 005203
3109 016034 005203
3110
3111 016036 012701 016516
3112 016042 174011
3113
3114 016044 012701 016516
3115 016050 012702 016506
3116 016054 012703 000004
3117 016060 022122
3118 016062 001402
3119 016064 000137 016346
3120 016070 077305
3121
3122 016072 000137 016372
3123
3124
3125 016076
3125 016076 012737 016104 001110
3126 016104 170011
3127
3128 016106 012700 016506
3129 016112 172410
3130
3131 016114 012737 016264 000244
3132 016122 012700 000001
3133 016126 012737 016450 000004
3134 016134 005003
3135
3136 016136 172406
3137 016140 170000
```

```
TST13: SCOPE
S1:
1$: MOV #1,$,LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
SETD ;SET FD
MOV #SPAT10,R0 ;LOAD ACO
LDD (R0),ACO
MOV #SERR0,FPVECT ;USE OF THE NON-EXISTENT AC-
;CUMULATOR SHOULD RESULT IN
;A TRAP TO 244.
MOV #1,R0 ;A FAILURE IN THE FSRC FLOWS
;WILL RESULT IN AN ODD ADDRESS
;TRAP TO 4.
MOV #SERR1,ERRVECT
CLR R3
S2: LDD AC7,ACO
S3: CFCC
INC R3
S4: INC R3
MOV #SDAT00,R1 ;NO TRAP OCCURRED!!
STD ACO,(R1) ;SEE IF ACO WAS MODIFIED.
MOV #SDAT00,R1
MOV #SPAT10,R2
MOV #4,R3
S5: CMP (R1)+,(R2)+
BEQ S6
JMP SERR2
S6: SOB R3,S5
JMP SERR3
;NOW TEST AC6.
S7:
1$: MOV #1,$,LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
SETD
MOV #SPAT10,R0 ;LOAD ACO
LDD (R0),ACO
MOV #SERR4,FPVECT
MOV #1,R0
MOV #SERR5,ERRVECT
CLR R3
S8: LDD AC6,ACO
S9: CFCC
```



```

3138 016142 005203          INC      R3
3139 016144 005203          S10:    INC      R3
3140
3141 016146 012701 016516          MOV     #SDAT00,R1
3142 016152 174011          STD     ACO,(R1)          ;NO TRAP! GET ACO.
3143
3144 016154 012701 016516          MOV     #SDAT00,R1          ;WAS ACO MODIFIED.
3145 016160 012702 016506          MOV     #SPAT10,R2
3146 016164 012703 000004          MOV     #4,R3
3147 016170 022122          S11:    CMP     (R1)+,(R2)+
3148 016172 001402          BEQ     S12
3149 016174 000137 016360          JMP     SERR6
3150 016200 077305          S12:    SOB     R3,S11
3151 016202 000137 016404          JMP     SERR7
3152
3153          ;TRAPPED TO 244.
3154 016206 021627 016030          SERR0:  CMP     (SP),#S3          ;PC OF TRAP CORRECT?
3155 016212 001402          BEQ     1$
3156 016214 000137 042066          JMP     FPSPUR
3157
3158 016220 012737 016076 016502 1$:    MOV     #S7,SADR
3159
3160 016226 011637 001236          SERR10: MOV     (SP),$TMP2
3161 016232 022626          CMP     (SP)+,(SP)+
3162 016234 005004          CLR     R4
3163 016236 170204          STFPS  R4          ;IS FPS CORRECT?
3164 016240 022704 100200          CMP     #100200,R4
3165 016244 001020          BNE     SERR15
3166
3167 016246 005004          CLR     R4
3168 016250 170304          STST   R4          ;IS FEC CORRECT?
3169 016252 022704 000002          CMP     #2,R4
3170 016256 001023          BNE     SERR20
3171 016260 000177 000216          JMP     @SADR
3172
3173 016264 021627 016140          SERR4:  CMP     (SP),#S9
3174 016270 001402          BEQ     1$
3175 016272 000137 042066          JMP     FPSPUR
3176 016276 012737 016526 016502 1$:    MOV     #SDONE,SADR
3177 016304 000750          BR      SERR10
3178
3179          ;REPORT FPS FAILURE:
3180 016306 012737 100200 001242 SERR15: MOV     #100200,$TMP4
3181 016314 010437 001240          MOV     R4,$TMP3
3182 016320 104117          1$:    ERROR  +117
3183 016322 000177 000154          JMP     @SADR
3184
3185          ;REPORT FEC BAD:
3186 016326 0:2737 000002 001242 SERR20: MOV     #2,$TMP4
3187 016334 010437 001240          MOV     R4,$TMP3
3188 016340 104120          1$:    ERROR  +120
3189 016342 000177 000134          JMP     @SADR
3190
3191
3192          ;ACO WAS MODIFIED. (BUT FSRC) FORK FAILED.
3193 016346 012737 016026 001236 SERR2:  MOV     #S2,$TMP2
3194 016354 104112          1$:    ERROR  +112
    
```



```

3195 016356 000463          BR      SDONE
3196 016360 012737 016136 001236 SERR6: MOV    #S8,$TMP2
3197 016366 104114          1$:   ERROR +114
3198 016370 000456          BR      SDONE
3199
3200 016372 012737 016026 001236 SERR3: MOV    #S2,$TMP2
3201 016400 104111          1$:   ERROR +111
3202 016402 000451          BR      SDONE
3203 016404 012737 016136 001236 SERR7: MOV    #S8,$TMP2
3204 016412 104113          1$:   ERROR +113
3205 016414 000444          BR      SDONE
3206
3207          ;FAILURE OF (BUT FSRC) CAUSED AN ODD ADDRESS TRAP TO 4.
3208 016416 021627 016030 SERR1: CMP    (SP),#S3          ;DID TRAP OCCUR ON TESTED INSTRUCTION?
3209 016422 001405          BEQ    1$
3210 016424 021627 016034          CMP    (SP),#S4
3211 016430 001402          BEQ    1$
3212 016432 000137 042120          JMP    CPSPUR
3213
3214 016436 011637 001236          1$:   MOV    (SP),$TMP2
3215 016442 022626          CMP    (SP)+,(SP)+
3216 016444 104115          2$:   ERROR +115
3217 016446 000427          BR      SDONE
3218
3219 016450 021627 016136 SERR5: CMP    (SP),#S8          ;DID TRAP OCCUR ON TEST INSTRUCTION?
3220 016454 001405          BEQ    1$
3221 016456 021627 016140          CMP    (SP),#S9
3222 016462 001402          BEQ    1$
3223 016464 000137 042120          JMP    CPSPUR
3224
3225 016470 011637 001236          1$:   MOV    (SP),$TMP2
3226 016474 022626          CMP    (SP)+,(SP)+
3227 016476 104116          2$:   ERROR +116
3228 016500 000412          BR      SDONE
3229
3230 016502 000000          SADR:  0
3231 016504 177777          -1
3232 016506 010421          SPAT10: .WORD 10421
3233 016510 021042          SPAT11: .WORD 21042
3234 016512 031463          SPAT12: .WORD 31463
3235 016514 042104          SPAT13: .WORD 42104
3236
3237 016516 000000          SDAT00: .WORD 0
3238 016520 000000          SDAT01: .WORD 0
3239 016522 000000          SDAT02: .WORD 0
3240 016524 000000          SDAT03: .WORD 0
3241
3242 016526          SDONE:
      016526 104413          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?).
    
```



3250

```
.SBTTL TEST # 14 - FSRC MODE 2 TEST
:*****
:*TEST 14      FSRC MODE 2 TEST
:*
:* THIS IS A TEST OF FSRC MODE 2, AUTO
:* INCREMENT MODE.
:*
:*****
```

```
TST14: SCOPE
3251 016530 000004 016540 001110  MOV    #J1,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
3252 016532 012737
3253 016540 170011  J1:      SETD           ;SET DOUBLE MODE
3254 016540 170011
3255
3256 016542 012700 017016  MOV    #JDATO,R0
3257 016546 172410  LDD    (R0),ACO      ;LOAD ACO
3258
3259 016550 012700 016776  MOV    #JDATIO,R0
3260 016554 005003  CLR    R3
3261 016556 012737 016646 000004  MOV    #JERRO,ERRVECT
3262
3263 016564 172420  J2:      LDD    (R0)+,ACO      ;TEST INSTRUCTION
3264 016566 005203  J3:      INC    R3
3265 016570 005203  J4:      INC    R3
3266
3267 016572 012701 017006  MOV    #JDATOO,R1
3268 016576 174011  STD    ACO,(R1)      ;PICK UP RESULTS
3269
3270 016600 020027 016766  CMP    R0,#JBUFO     ;WAS AN AUTO
3271 016604 001001  BNE    1$           ;DECREMENT EXECUTED?
3272 016606 000442  BR     JERR1
3273
3274 016610 012702 016776  1$:     MOV    #JDATIO,R2      ;IS DATA CORRECT?
3275 016614 012703 017006  MOV    #JDATOO,R3
3276 016620 012704 000004  MOV    #4,R4
3277 016624 022223  J5:     CMP    (R2)+,(R3)+
3278 016626 001401  BEQ    J6
3279 016630 000443  BR     JERR2
3280 016632 077404  J6:     SOB   R4,J5
3281
3282 016634 022700 017006  CMP    #JDATIO+10,R0 ;WAS R0 INCREM.
3283 016640 001401  BEQ    J7           ;BY 10 (OCTAL)
3284 016642 000424  BR     JERR1
3285
3286 016644 000470  J7:     BR     JDONE
3287
3288 ;IF A TRAP THROUGH 4 OCCURS COME HERE
3289
3290 016646 021627 016566  JERRO:  CMP    (SP),#J3      ;SEE IF THE TRAP
3291 016652 001405  BEQ    J10          ;OCCURRED ON THE
3292 016654 021627 016570  CMP    (SP),#J4      ;TESTED INSTRUCTION
3293 016660 001402  BEQ    J10
3294 016662 000137 042120  JMP    CPSPUR
3295
3296 016666 012737 000762 001240  J10:   MOV    #762,$TMP3     ;REPORT FSRC FLOW
3297 016674 012737 000322 001242  MOV    #322,$TMP4     ;FAILURE
3298 016702 011637 001236  MOV    (SP),$TMP2
```



```
3299 016706 022626
3300 016710 104052
3301 016712 000445
3302
3303 016714
3304 016714 012737 016564 001236
3305 016722 010037 001240
3306 016726 012737 017006 001242
3307 016734 104053
3308 016736 000433
3309
3310
3311
3312 016740
3313 016740 012737 016564 001236
3314 016746 012737 016776 001240
3315 016754 012737 017006 001242
3316 016762 104054
3317 016764 000420
3318
3319 016766 010421
3320 016770 021042
3321 016772 042104
3322 016774 031463
3323
3324 016776 052525
3325 017000 114631
3326 017002 063146
3327 017004 073567
3328
3329 017006 000000
3330 017010 000000
3331 017012 000000
3332 017014 000000
3333
3334 017016 177777
3335 017020 177777
3336 017022 177777
3337 017024 177777
3338
3339
3340 017026
      017026 104413

      1$:      CMP      (SP)+,(SP)+
      ERROR      +52
      BR         JDONE

JERR1:
      MOV      #J2,$TMP2      ;REPORT, RO NOT
      MOV      R0,$TMP3      ;CORRECTLY AFFECTED
      MOV      #JDATIO+10,$TMP4
      1$:      ERROR      +53
      BR         JDONE

;REPORT DATA FAILURE

JERR2:
      MOV      #J2,$TMP2
      MOV      #JDATIO,$TMP3
      MOV      #JDAT00,$TMP4
      1$:      ERROR      +54
      BR         JDONE

JBUF0: .WORD 010421
JBUF1: .WORD 021042
JBUF2: .WORD 042104
JBUF3: .WORD 031463

JDATIO: .WORD 052525
JDATIO1: .WORD 114631
JDATIO2: .WORD 063146
JDATIO3: .WORD 073567

JDAT00: .WORD 0
JDAT01: .WORD 0
JDAT02: .WORD 0
JDAT03: .WORD 0

JDAT0: .WORD -1
JDAT1: .WORD -1
JDAT2: .WORD -1
JDAT3: .WORD -1

JDONE:
      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?).

3341
3348
```



3349

```
.SBTTL TEST # 15 - FSRC MODE 4 TEST
:*****
:TEST 15 FSRC MODE 4 TEST
:*
:* THIS IS A TEST OF FSRC MODE 4, AUTO
:* DECREMENT MODE.
:*****
```

```

3350 017030 000004 017040 001110 TST15: SCOPE
3351 017032 012737 017040 001110 MOV #K1,$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
3352 017040 K1:
3353 017040 170011 SETD ;SET DOUBLE MODE
3354
3355 017042 012700 017314 MOV #KPATO,R0
3356 017046 172410 LDD (R0),ACO ;LOAD A DEFAULT
3357 ;PATTERN INTO ACO
3358 017050 012700 017274 MOV #KBUF0,R0
3359 017054 005003 CLR R3
3360 017056 012737 017146 000004 MOV #KERR0,ERRVECT
3361
3362 017064 172440 K2: LDD -(R0),ACO ;TEST INSTRUCTION
3363 017066 005203 K3: INC R3
3364 017070 005203 K4: INC R3
3365
3366 017072 012701 017304 MOV #KDAT00,R1
3367 017076 174011 STD ACO,(R1) ;PICK UP THE RESULT
3368
3369 017100 020027 017304 CMP R0,#KBUF0+10 ;WAS AN AUTO
3370 017104 001001 BNE 1$ ;INCREMENT EXECUTED
3371 017106 000441 BR KERR1
3372
3373 017110 012702 017264 1$: MOV #KDAT10,R2 ;IS DATA CORRECT?
3374 017114 012703 017304 MOV #KDAT00,R3
3375 017120 012704 000004 MOV #4,R4
3376 017124 022223 K5: CMP (R2)+,(R3)+
3377 017126 001401 BEQ K6
3378 017130 000442 BR KERR2
3379 017132 077404 K6: SOB R4,K5
3380
3381 017134 022700 017264 CMP #KBUF0-10,R0 ;WAS R0 DECREMENTED
3382 017140 001401 BEQ K7 ;PROPERLY?
3383 017142 000423 BR KERR1
3384
3385 017144 000467 K7: BR KDONE
3386
3387 ;TRAP TO HERE ON AN ODD ADDRESS ERROR
3388
3389 017146 021627 017066 KERR0: CMP (SP),#K3 ;SEE IF THE ERROR
3390 017152 001405 BEQ K10 ;OCCURRED AT THE
3391 017154 021627 017070 CMP (SP),#K4 ;INSTRUCTION TESTED.
3392 017160 001402 BEQ K10
3393 017162 000137 042120 JMP CPSPUR
3394
3395 017166 012737 000762 001240 K10: MOV #762,$TMP3 ;REPORT FAILURE IN
3396 017174 012737 000324 001242 MOV #324,$TMP4 ;FSRC FLOWS
3397 017202 C11637 001236 MOV (SP),$TMP2

```



3398 017206 104055  
3399 017210 000445  
3400  
3401 017212  
3402 017212 012737 017064 001236  
3403 017220 010037 001240  
3404 017224 012737 017264 001242  
3405 017232 104056  
3406 017234 000433  
3407  
3408  
3409  
3410 017236  
3411 017236 012737 017064 001236  
3412 017244 012737 017264 001240  
3413 017252 012737 017304 001242  
3414 017260 104057  
3415 017262 000420  
3416  
3417 017264 052525  
3418 017266 114631  
3419 017270 063140  
3420 017272 073567  
3421  
3422 017274 010421  
3423 017276 031463  
3424 017300 042104  
3425 017302 021042  
3426  
3427 017304 000000  
3428 017306 000000  
3429 017310 000000  
3430 017312 000000  
3431  
3432 017314 177777  
3433 017316 177777  
3434 017320 177777  
3435 017322 177777  
3436  
3437 017324 104413

1\$: ERROR +55  
BR KDONE  
  
KERR1: MOV #K2,\$TMP2  
MOV R0,\$TMP3  
MOV #KDAT10,\$TMP4  
1\$: ERROR +56  
BR KDONE  
  
:REPORT DATA FAILURE  
  
KERR2: MOV #K2,\$TMP2  
MOV #KDAT10,\$TMP3  
MOV #KDAT00,\$TMP4  
1\$: ERROR +57  
BR KDONE  
  
KDAT10: .WORD 052525  
KDAT11: .WORD 114631  
KDAT12: .WORD 063140  
KDAT13: .WORD 073567  
  
KBUF0: .WORD 010421  
KBUF1: .WORD 031463  
KBUF2: .WORD 042104  
KBUF3: .WORD 021042  
  
KDAT00: .WORD 0  
KDAT01: .WORD 0  
KDAT02: .WORD 0  
KDAT03: .WORD 0  
  
KPAT0: .WORD -1  
KPAT1: .WORD -1  
KPAT2: .WORD -1  
DPAT3: .WORD -1  
  
KDONE: RSETUP

:REPORT R0  
:INCORRECTLY AFFECTED.

: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?).

3438  
3445



3446

```
.SBTTL TEST # 16 - FSRC MODE 2, WITH FD=0, TEST
*****
*TEST 16      FSRC MODE 2, WITH FD=0, TEST
*
* THIS IS A TEST OF FSRC MODE 2 WITH
* FD=0. (AUTO INCREMENT)
*
*****
```

```
TST16: SCOPE
3447 017326 000004 017336 001110 MOV #L1,$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
3448 017330 012737
3449 017336 L1:
3450 017336 170011 SETD ;SET DOUBLE MODE
3451
3452 017340 012700 017606 MOV #LPAT10,R0
3453 017344 172410 LDD (R0),AC0 ;LOAD ACO
3454
3455 017346 012700 017630 MOV #LDAT10,R0 ;SET UP THE INPUT
3456 017352 012701 017616 MOV #LPAT20,R1 ;DATA
3457 017356 012702 000004 MOV #4,R2
3458
3459 017362 012120 1$: MOV (R1)+,(R0)+
3460 017364 077202 SOB R2,1$
3461
3462 017366 012700 017630 MOV #LDAT10,R0
3463 017372 005003 CLR R3
3464 017374 170001 SETF ;CLEAR FD.
3465
3466 017376 172420 L2: LDF (R0)+,AC0
3467 017400 005203 L3: INC R3
3468
3469 017402 L4:
3470 017402 170011 SETD ;SET FD
3471
3472 017404 012701 017642 MOV #LDAT00,R1
3473 017410 174011 STD ACO,(R1) ;PICK UP RESULTS
3474
3475 017412 020027 017634 CMP R0,#LDAT12 ;WAS R0 INCREMENTED
3476 017416 001401 BEQ 1$ ;CORRECTLY BY 4
3477 017420 000421 BR LERR1
3478
3479 017422 012737 177777 017634 1$: MOV #-1,LDAT12
3480 017430 012737 177777 017636 MOV #-1,LDAT13
3481 017436 012702 017630 MOV #LDAT10,R2 ;IS DATA CORRECT
3482 017442 012703 017642 MOV #LDAT00,R3
3483 017446 012704 000004 MOV #4,R4
3484
3485 017452 022223 L5: CMP (R2)+,(R3)+
3486 017454 001401 BEQ L6
3487 017456 000427 BR LERR2
3488 017460 077404 L6: SOB R4,L5
3489
3490 017462 000473 BR LDONE
3491
3492 017464 LERR1: ;REPORT FAILURE
3493 017464 012737 017376 001236 MOV #L2,$TMP2 ;RO NOT INCREMENTED
3494 017472 010037 001240 MOV R0,$TMP3 ;BY 4
```



```

3495 017476 012737 017634 001242      MOV      #LDAT12,$TMP4
3496 017504 104060      1$:      ERROR      +60
3497 017506 000461      BR        LDONE
3498
3499 017510      LERR3:      ;REPORT DATA FAILURE.
3500 017510 012737 017376 001236      MOV      #L2,$TMP2
3501 017516 012737 017630 001240      MOV      #LDAT10,$TMP3
3502 017524 012737 017642 001242      MOV      #LDAT00,$TMP4
3503 017532 104061      1$:      ERROR      +61
3504 017534 000446      BR        LDONE
3505
3506 017536 012702 017616      LERR2:      MOV      #LPAT20,R2      ;DID (BUT FD)
3507 017542 012703 017642      MOV      #LDAT00,R3      ;FAIL.
3508 017546 012704 000004      MOV      #4,R4
3509 017552 022223      1$:      CMP      (R2)+,(R3)+
3510 017554 001355      BNE      LERR3
3511 017556 077403      SOB      R4,1$
3512 017560 012737 017376 001236      MOV      #L2,$TMP2
3513 017566 012737 017630 001240      MOV      #LDAT10,$TMP3
3514 017574 012737 017644 001242      MOV      #LDAT01,$TMP4
3515 017602 104062      2$:      ERROR      +62
3516 017604 000422      BR        LDONE
3517
3518 017606 177777      LPAT10: .WORD      -1
3519 017610 177777      LPAT11: .WORD      -1
3520 017612 177777      LPAT12: .WORD      -1
3521 017614 177777      LPAT13: .WORD      -1
3522
3523 017616 052525      LPAT20: .WORD      052525
3524 017620 114631      LPAT21: .WORD      114631
3525 017622 063142      LPAT22: .WORD      063142
3526 017624 073567 000001      LPAT23: .WORD      073567,1
3527 017630 000000      LDAT10: .WORD      0
3528 017632 000000      LDAT11: .WORD      0
3529 017634 000000      LDAT12: .WORD      0
3530 017636 000000 000001      LDAT13: .WORD      0,1
3531 017642 000000      LDAT00: .WORD      0
3532 017644 000000      LDAT01: .WORD      0
3533 017646 000000      LDAT02: .WORD      0
3534 017650 000000      LDAT03: .WORD      0
3535
3536 017652      LDONE:      RSETUP
      017652 104413
;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?).
3537
3545
    
```



3546

```

.SBTTL TEST # 17 - FSRC MODE 2 WITH GR7, IMMEDIATE MODE, TEST
*****
*TEST 17 FSRC MODE 2 WITH GR7, IMMEDIATE MODE, TEST
*
* THIS IS A TEST OF FSRC MODE 2
* USING GR7 (THE PC). THIS IS IMMEDIATE
* MODE.
*****
TST17: SCOPE
    
```

3547	017654	000004				
3548	017656					
3549	017656	170011				
3550						
3551	017660	012700	020152			
3552	017664	172410				
3553						
3554	017666	005004				
3555	017670	012737	020112	000004		
3556						
3557	017676	172427	000000			
3558		017700				
3559	017700	005204				
3560	017702	005204				
3561	017704	005204				
3562	017706	005204				
3563						
3564	017710	020427	000003			
3565	017714	001401				
3566	017716	000443				
3567						
3568						
3569						
3570	017720	012700	020172			
3571	017724	174010				
3572						
3573	017726	012700	020172			
3574	017732	022720	005204			
3575	017736	001401				
3576	017740	000451				
3577	017742	012701	000003			
3578	017746	005720				
3579	017750	001002				
3580	017752	077103				
3581	017754	000512				
3582						
3583	017756	012700	020172			
3584	017762	012701	000004			
3585	017766	022720	005204			
3586	017772	001401				
3587	017774	000433				
3588	017776	077105				
3589						
3590	020000					
3591	020000	012737	017676	001236		
3592	020006	012737	020162	001240		
3593	020014	012737	020172	001242		

```

M1: SETD
MOV #MPAT10,R0
LDD (R0),ACO ;LOAD BACKGROUND
;PATTERN INTO ACO.
CLR R4
MOV #MERR3,ERRVECT
M15: LDD #0,ACO ;TEST INSTRUCTION
;=-2
;WORD 5204
M2: INC R4 ;NOTE THAT
M3: INC R4 ;005204=INC R4
M4: INC R4
CMP R4,#3 ;SEE IF THE PC
BEQ 1$ ;WAS INCREMENTED
BR MERR0 ;BY 2 DURING THE
;INSTRUCTION. IF
;NOT THEN A BAD
;CONSTANT WAS GENERATED
1$: MOV #MDAT00,R0
STD ACO,(R0) ;GET THE DATA
MOV #MDAT00,R0
CMP #5204,(R0)+ ;IS THE DATA CORRECT?
BEQ M5
BR MERR1
M5: MOV #3,R1
M6: TST (R0)+
BNE M7
SOB R1,M6
BR MDONE
M7: MOV #MDAT00,R0 ;DID (BUT GRM) FAIL?
MOV #4,R1
M8: CMP #5204,(R0)+
BEQ M9
BR MERR1
M9: SOB R1,M8
MERR2: MOV #M15,$TMP2 ;REPORT FAILURE
MOV #MPAT20,$TMP3 ;OF (BUT GR7)
MOV #MDAT00,$TMP4
    
```



```

3594 020022 104063          1$:  ERROR  +63
3595 020024 000466          BR      MDONE
3596
3597 020026 012705 017702  MERR0: MOV    #M2,R5      :REPORT FAILURE
3598 020032 010537 001242      MOV    R5,$TMP4      :PC INCREMENTED
3599 020036 162704 000003      SUB    #3,R4
3600 020042 006304          ASL    R4
3601 020044 160405          SUB    R4,R5
3602 020046 010537 001240      MOV    R5,$TMP3
3603 020052 012737 017676 001236  1$:  MOV    #M15,$TMP2
3604 020060 104064          ERROR  +64
3605 020062 000447          BR      MDONE
3606
3607 020064          MERR1:          :REPORT DATA
3608 020064 012737 017676 001236      MOV    #M15,$TMP2      :FAILURE
3609 020072 012737 020172 001240      MOV    #MDAT00,$TMP3
3610 020100 012737 020162 001242      MOV    #MPAT20,$TMP4
3611 020106 104066          1$:  ERROR  +66
3612 020110 000434          BR      MDONE
3613          :TRAP TO HERE THROUGH 4.
3614 020112 032716 000001  MERR3: BIT    #1,(SP)      :SEE IF THE
3615 020116 001002          BNE    1$              :TRAP TO 4 OCCURRED
3616 020120 000137 042120          JMP    CPSPUR          :BECAUSE OF AN
3617          :ODD ADDRESS
3618 020124 011637 001240  1$:  MOV    (SP),$TMP3      :IF YES REPORT
3619 020130 012737 017702 001242      MOV    #M2,$TMP4      :BAD CONSTANT
3620 020136 012737 017676 001236      MOV    #M15,$TMP2      :GENERATED
3621 020144 022626          CMP    (SP)+,(SP)+
3622 020146 104065          2$:  ERROR  +65
3623 020150 000414          BR      MDONE
3624
3625 020152 177777          MPAT10: .WORD  -1
3626 020154 177777          MPAT11: .WORD  -1
3627 020156 177777          MPAT12: .WORD  -1
3628 020160 177777          MPAT13: .WORD  -1
3629
3630 020162 005204          MPAT20: .WORD  5204
3631 020164 005204          MPAT21: .WORD  5204
3632 020166 005204          MPAT22: .WORD  5204
3633 020170 005204          MPAT23: .WORD  5204
3634
3635 020172 000000          MDAT00: .WORD  0
3636 020174 000000          MDAT01: .WORD  0
3637 020176 000000          MDAT02: .WORD  0
3638 020200 000000          MDAT03: .WORD  0
3639
3640 020202          MDONE:
      020202 104413          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?).
3641
3648
    
```



3649

```
.SBTTL TEST # 20 - FSRC MODE 3 TEST
:*****
:TEST 20 FSRC MODE 3 TEST
:
:* THIS IS A TEST OF FSRC MODE 3, AUTO INCREMENT
:* DEFERRED
:
:*****
```

```
020204 000004
3650
3651 020206
3652 020206 170011
3653
3654 020210 012700 020670
3655 020214 172410
3656
3657 020216 012700 020656
3658 020222 005003
3659 020224 012737 020400 000004
3660
3661
3662 020232 172430
3663 020234 005203
3664 020236 005203
3665
3666 020240 012701 020636
3667 020244 174011
3668
3669 020246 020027 020660
3670 020252 001437
3671
3672 020254 020027 020666
3673 020260 001001
3674 020262 000506
3675
3676 020264 020027 020646
3677 020270 001001
3678 020272 000520
3679
3680 020274 020027 020656
3681 020300 001023
3682
3683 020302 012702 020636
3684 020306 012703 000004
3685 020312 022227 177777
3686 020316 001002
3687 020320 077304
3688 020322 000510
3689
3690 020324 012702 020636
3691 020330 012703 020656
3692 020334 012704 000004
3693 020340 022223
3694 020342 001002
3695 020344 077403
3696 020346 000502
3697
```

```
TST20: SCOPE
N1: SETD ;SET FD MODE
MOV #NPAT10,R0
LDD (R0),AC0 ;LOAD AC0 WITH A DEFAULT
;PATTERN
PJV #NPAT20,R0
CLR R3
MOV #NERR0,ERRVECT ;IF A FAILURE OCCURS
;IN THE FSRC FLOWS AN
;ODD TRAP TO 4 COULD OCCUR
;TEST INSTRUCTION.
N2: LDD @(R0)+,AC0
N3: INC R3
N4: INC R3
MOV #NDAT00,R1
STD AC0,(R1) ;GET THE DATA
CMP R0,#NPAT20+2 ;WAS R0 INCREMENTED
BEQ N12 ;BY 2?
N5: CMP R0,#NPAT20+10 ;FSRC MODE ??
BNE N6
BR NERR1
N6: CMP R0,#NPAT20-10 ;FSRC MODE 4?
BNE N7
BR NERR2
N7: CMP R0,#NPAT20
BNE N11
MOV #NDAT00,R2 ;FSRC MODE 0?
MOV #4,R3
CMP (R2)+,#-1
BNE N9
SOB R3,N8
BR NERR3
N9: MOV #NDAT00,R2 ;FSRC MODE 1
MOV #NPAT20,R3
MOV #4,R4
N10: CMP (R2)+,(R3)+
BNE N11
SOB R4,N10
BR NERR4
```



```

3698 020350 000505      N11:  BR      NERR5
3699
3700 020352 012702 020636  N12:  MOV      #NDAT00,R2      ;DATA CORRECT?
3701 020356 012703 020700      MOV      #NDAT10,R3
3702 020362 012704 000004      MOV      #4,R4
3703 020366 022223      N13:  CMP      (R2)+,(R3)+
3704 020370 001002      BNE     N14
3705 020372 077403      SOB    R4,N13
3706 020374 000545      BR     NDONE
3707
3708 020376 000504      N14:  BR      NERR6
3709
3710      ;IF AN ODD ADDRESS TRAP OCCURS COME HERE
3711      ;TO SEE IF THE FAILURE WAS IN THE FSRC
3712      ;FLOWS
3713
3714 020400 022716 020236  NERR0:  CMP      #N4,(SP)      ;FSRC MODE 6 OR 7?
3715 020404 001412      BEQ    NERR10
3716 020406 022716 020234      CMP      #N3,(SP)
3717 020412 001402      BEQ    1$
3718 020414 000137 042120      JMP    CPSPUR
3719 020420 020027 020654  1$:    CMP      R0,#NPAT20-2    ;FSRC MODE 5?
3720 020424 001407      BEQ    NERR11
3721 020426 000137 042120      JMP    CPSPUR
3722
3723 020432      NERR10:  MOV      (SP),$TMP2      ;WENT TO FSRC
3724 020432 011637 001236      CMP      (SP)+,(SP)+    ;MODE 6 OR 7.
3725 020436 022626      1$:    ERROR   +67
3726 020440 104067      BR     NDONE
3727 020442 000522
3728
3729 020444 011637 001236  NERR11:  MOV      (SP),$TMP2      ;WENT TO FSRC
3730 020450 022626      CMP      (SP)+,(SP)+    ;MODE 5.
3731 020452 012737 000627 001244      MOV      #627,$TMP5
3732 020460 012737 000323 001250      MOV      #323,$TMP7
3733 020466 012737 000325 001246      MOV      #325,$TMP6
3734 020474 104070      1$:    ERROR   +70
3735 020476 000504      BR     NDONE
3736 020500 012737 000322 001246  NERR1:  MOV      #322,$TMP6      ;FSRC MODE 2.
3737 020506 012737 000627 001244  NERR20:  MOV      #627,$TMP5
3738 020514 012737 000323 001250      MOV      #323,$TMP7
3739 020522 012737 020232 001236      MOV      #N2,$TMP2
3740 020530 104071      1$:    ERROR   +71
3741 020532 000466      BR     NDONE
3742 020534 012737 000324 001246  NERR2:  MOV      #324,$TMP6      ;FSRC MODE 4
3743 020542 000761      BR     NERR20
3744 020544 012737 000320 001246  NERR3:  MOV      #320,$TMP6      ;FSRC MODE 0
3745 020552 000755      BR     NERR20
3746 020554 012737 000321 001246  NERR4:  MOV      #321,$TMP6      ;FSRC MODE 1
3747 020562 000751      BR     NERR20
3748
3749 020564 010037 001240  NERR5:  MOV      R0,$TMP3      ;R0 NOT
3750 020570 012737 020660 001242      MOV      #NPAT20+2,$TMP4 ;INCREMENTED
3751 020576 012737 020232 001236      MOV      #N2,$TMP2      ;PROPERLY.
3752 020604 104072      1$:    ERROR   +72
3753 020606 000440      BR     NDONE
3754

```



```
3755 020610
3756 020610 012737 020232 001236
3757 020616 012737 020636 001240
3758 020624 012737 020700 001242
3759 020632 104073
3760 020634 000425
3761
3762 020636 000000
3763 020640 000000
3764 020642 000000
3765 020644 000000 052525 052525
3766 020656 020700
3767 020660 070707
3768 020662 070707
3769 020664 070707 000001
3770 020670 177777
3771 020672 177777
3772 020674 177777
3773 020676 177777
3774
3775 020700 010421
3776 020702 021042
3777 020704 031463
3778 020706 042104
3779
3780 020710
      020710 104413
```

NERR6:           MOV       #N2,\$TMP2           ;DATA FAILURE.  
                  MOV       #NDAT00,\$TMP3  
                  MOV       #NDAT10,\$TMP4  
1\$:                ERROR     +73  
                  BR         NDONE

NDAT00: .WORD 0  
NDAT01: .WORD 0  
NDAT02: .WORD 0  
NDAT03: .WORD 0,52525,52525,52525,52525  
NPAT20: .WORD NDAT10  
NPAT21: .WORD 070707  
NPAT22: .WORD 070707  
NPAT23: .WORD 070707,1  
NPAT10: .WORD -1  
NPAT11: .WORD -1  
NPAT12: .WORD -1  
NPAT13: .WORD -1

NDAT10: .WORD 010421  
NDAT11: .WORD 021042  
NDAT12: .WORD 031463  
NDAT13: .WORD 042104

NDONE:           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?).
```

3781  
3788



3789

```

.SBTTL TEST # 21 - FSRC MODE 5 TEST
*****
*TEST 21      FSRC MODE 5 TEST
*
* THIS IS A TEST OF FSRC MODE 5, AUTO DECREMENT
* DEFERRED.
*
*****
  
```

```

020712 000004
3790
3791 020714
3792 020714 170011
3793
3794 020716 012700 021374
3795 020722 172410
3796
3797 020724 012700 021362
3798 020730 005003
3799 020732 012737 021104 000004
3800
3801
3802
3803 020740 172450
3804 020742 005203
3805 020744 005203
3806
3807 020746 012701 021342
3808 020752 174011
3809
3810 020754 020027 021360
3811 020760 001436
3812
3813 020762 020027 021372
3814 020766 001001
3815 020770 000505
3816
3817 020772 020027 021352
3818 020776 001001
3819 021000 000517
3820
3821 021002 020027 021362
3822
3823 021006 012702 021344
3824 021012 012703 000004
3825 021016 022227 177777
3826 021022 001002
3827 021024 077304
3828 021026 000510
3829
3830 021030 012702 021342
3831 021034 012703 021362
3832 021040 012704 000004
3833 021044 022223
3834 021046 001002
3835 021050 077403
3836 021052 000502
3837
  
```

```

TST21: SCOPE
01:      SETD          ;SET FD MODE
        MOV    #OPAT10,R0
        LDD    (R0),AC0 ;LOAD AC0 WITH A
        ;DEFAULT PATTERN.
        MOV    #OPAT21,R0
        CLR    R3
        MOV    #OERR0,ERRVEC ;IF A FAILURE
        ;OCCURS IN THE FSRC
        ;FLOWS AN ODD ADDR.
        ;TRAP TO 4 MAY OCCUR.
        ;TEST INSTRUCTION
02:      LDD    @-(R0),AC0
03:      INC    R3
04:      INC    R3
        MOV    #ODAT00,R1
        STD    AC0,(R1) ;GET THE DATA
        CMP    R0,#OPAT20 ;WAS R0 DECREMENTED
        BEQ    012 ;BY 2?
05:      CMP    R0,#OPAT21+10 ;FSRC MODE 2
        BNE    06
        BR    OERR1
06:      CMP    R0,#OPAT21-10 ;FSRC MODE 4?
        BNE    07
        BR    OERR2
07:      CMP    R0,#OPAT21
        ;FSRC MODE 0?
        MOV    #ODAT01,R2
        MOV    #4,R3
08:      CMP    (R2)+,#-1
        BNE    09
        SOB   R3,08
        BR    OERR3
09:      MOV    #ODAT00,R2 ;FSRC MODE 1?
        MOV    #OPAT21,R3
        MOV    #4,R4
10:      CMP    (R2)+,(R3)+
        BNE    011
        SOB   R4,010
        BR    OERR4
  
```



```

3838 021054 000505      011:  BR      CERR5
3839
3840 021056 012702 021342      012:  MOV      #ODAT00,R2      ;DATA CORRECT?
3841 021062 012703 021404      MOV      #ODATIO,R3
3842 021066 012704 000004      MOV      #4,R4
3843 021072 022223      013:  CMP      (R2)+,(R3)+
3844 021074 001002      BNE      014
3845 021076 077403      SOB      R4,013
3846 021100 000545      BR      ODONE
3847
3848 021102 000504      014:  BR      OERR6
3849
3850      ;IF AN ODD ADDRESS TRAP OCCURS COME
3851      ;HERE TO SEE IF THE FAILURE WAS IN THE
3852      ;FSRC FLOWS:
3853
3854 021104 022716 020744      OERR0:  CMP      #04,(SP)      ;FSRC MODE 6 OR 7?
3855 021110 001412      BEQ      OERR10
3856 021112 022716 020742      CMP      #03,(SP)
3857 021116 001402      BEQ      1$
3858 021120 000137 042120      JMP      CPSPUR
3859 021124 020027 021364      1$:    CMP      R0,#OPAT21+2      ;FSRC MODE 3?
3860 021130 001425      BEQ      OERR1
3861 021132 000137 042120      JMP      CPSPUR
3862
3863 021136      OERR10:  MOV      (SP),$TMP2      ;WENT TO FSRC
3864 021136 011637 001236      CMP      (SP)+,(SP)+      ;MODE 6 OR 7
3865 021142 022626      1$:    ERROR   +74
3866 021144 104074      BR      ODONE
3867 021146 000522
3868
3869 021150 011637 001240      OERR11:  MOV      (SP),$TMP3      ;WENT TO FSRC MODE
3870 021154 022626      CMP      (SP)+,(SP)+      ;3
3871 021156 012737 000627 001244      MOV      #627,$TMP5
3872 021164 012737 000325 001250      MOV      #325,$TMP7
3873 021172 012737 000323 001246      MOV      #323,$TMP6
3874 021200 104075      1$:    ERROR   +75
3875 021202 000504      BR      ODONE
3876
3877 021204 012737 000322 001246      OERR1:  MOV      #322,$TMP6      ;FSRC MODE2
3878 021212 012737 000627 001242      OERR20:  MOV      #627,$TMP4
3879 021220 012737 000325 001250      MOV      #325,$TMP7
3880 021226 012737 020740 001236      MOV      #02,$TMP2
3881 021234 104076      1$:    ERROR   +76
3882 021236 000466      BR      ODONE
3883 021240 012737 000324 001246      OERR2:  MOV      #324,$TMP6      ;FSRC MODE 4
3884 021246 000761      BR      OERR20
3885 021250 012737 000320 001246      OERR3:  MOV      #320,$TMP6      ;FSRC MODE 0
3886 021256 000755      BR      OERR20
3887 021260 012737 000321 001246      OERR4:  MOV      #321,$TMP6      ;FSRC MODE 1
3888 021266 000751      BR      OERR20
3889
3890 021270 010037 001240      OERR5:  MOV      R0,$TMP3      ;R0 NOT DECREMENTED
3891 021274 012737 021360 001242      MOV      #OPAT20,$TMP4      ;PROPERLY
3892 021302 012737 020744 001236      MOV      #04,$TMP2
3893 021310 104077      1$:    ERROR   +77
3894 021312 000440      BR      ODONE

```



```
3895
3896 021314
3897 021314 012737 020740 001236 OERR6:
3898 021322 012737 021342 001240 MOV #02,$TMP2 ;DATA FAILURE
3899 021330 012737 021404 001242 MOV #ODAT00,$TMP3
3900 021336 104100 1$: ERROR +100
3901 021340 000425 BR ODONE
3902
3903 021342 000000 ODAT00: .WORD 0
3904 021344 000000 ODAT01: .WORD 0
3905 021346 000000 ODAT02: .WORD 0
3906 021350 000000 052525 052525 ODAT03: .WORD 0,52525,52525,52525
3907 021360 021404 OPAT20: .WORD ODAT10
3908 021362 070707 OPAT21: .WORD 070707
3909 021364 070707 OPAT22: .WORD 070707
3910 021366 070707 OPAT23: .WORD 070707
3911 021370 070707 000001 OPAT24: .WORD 070707,1
3912 021374 177777 OPAT10: .WORD -1
3913 021376 177777 OPAT11: .WORD -1
3914 021400 177777 OPAT12: .WORD -1
3915 021402 177777 OPAT13: .WORD -1
3916
3917 021404 073557 ODAT10: .WORD 73557
3918 021406 004210 ODAT11: .WORD 004210
3919 021410 114631 ODAT12: .WORD 114631
3920 021412 125252 ODAT13: .WORD 125252
3921
3922 021414
021414 104413 ODONE:
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?).
```

3923  
3929



3930

021416 000004  
3931  
3932 021420  
3933 021420 170011  
3934  
3935 021422 012700 022040  
3936 021426 172410  
3937  
3938 021430 012737 021536 000004  
3939  
3940 021436 012700 021607  
3941  
3942 021442 172460 000241  
3943 021444  
3944  
3945 021446 012701 022060  
3946 021452 174011  
3947 021454 012703 000004  
3948 021460 012702 022050  
3949 021464 012701 022060  
3950 021470 022221  
3951 021472 001007  
3952 021474 077303  
3953 021476 022700 021607  
3954 021502 001401  
3955 021504 000512  
3956 021506 000137 022070  
3957  
3958 021512 012701 022060  
3959 021516 012703 000004  
3960 021522 022721 177777  
3961 021526 001401  
3962 021530 000512  
3963 021532 077305  
3964 021534 000523  
3965  
3966 021536 021627 021444  
3967 021542 001411  
3968 021544 021627 021446  
3969 021550 001402  
3970 021552 000137 042120  
3971  
3972 021556 012737 000327 001246  
3973 021564 000443  
3974 021566 022700 021607  
3975 021572 001004  
3976 021574 012737 000321 001246  
3977 021602 000434  
3978 021604 022700 021617  
3979 021610 001004

```

.SBTTL TEST # 22 - FSRC MODE 6 TEST
:*****
:*TEST 22      FSRC MODE 6 TEST
:*
:* THIS IS A TEST OF FSRC MODE 6, INDEX MODE
:*
:*****
TST22: SCOPE

P1:      SETD                ;SET FD MODE

        MOV      #PPAT10,R0
        LDD      (R0),AC0    ;LOAD A DEFAULT PATTERN
                                ;INTO AC0
        MOV      #PERRO,ERRVECT ;IF THE (BUT FSRC) FORQ
                                ;FAILS AN ODD ADDRESS TRAP
                                ;COULD OCCUR.

        MOV      #PDAT10-241,R0

P2:      LDD      241(R0),AC0
P3=P2+2

P4:      MOV      #PDAT00,R1
        STD      AC0,(R1)    ;GET THE DATA
        MOV      #4,R3
        MOV      #PDAT10,R2
        MOV      #PDAT00,R1
P5:      CMP      (R2)+,(R1)+ ;CHECK THE DATA
        BNE      P6
        SOB      R3,P5
        CMP      #PDAT10-241,R0 ;RO CORRECT?
        BEQ      1$
        BR      PERR21
1$:      JMP      PDONE

P6:      MOV      #PDAT00,R1
        MOV      #4,R3
P7:      CMP      #-1,(R1)+   ;WAS IT FSRC MODE 0?
        BEQ      P8
        BR      PERR1
P8:      SOB      R3,P7
        BR      PERR2
;TRAP TO HERE ON AN ODD ADDRESS
PERR0:  CMP      (SP),#P3
        BEQ      PERR11
        CMP      (SP),#P4   ;WAS IT FSRC MODE 7?
        BEQ      PERR10
        JMP      CPSPUR

PERR10: MOV      #327,$TMP6
        BR      PERR17
PERR11: CMP      #PDAT10-241,R0 ;WAS IT FSRC MODE 1
        BNE      PERR12
        MOV      #321,$TMP6
        BR      PERR17
PERR12: CMP      #PDAT10-241+10,R0 ;WAS IT FSRC MODE 2
        BNE      PERR13
    
```



```

3980 021612 012737 000322 001246      MOV      #322,$TMP6
3981 021620 000425                BR      PERR17
3982 021622 022700 021611      PERR13: CMP    #PDATIO-241+2,R0      ;WAS IT FSRC MODE 3
3983 021626 001004                BNE     PERR14
3984 021630 012737 000323 001246      MOV      #323,$TMP6
3985 021636 000416                BR      PERR17
3986 021640 022700 021577      PERR14: CMP    #PDATIO-241-10,R0    ;WAS IT FSRC MODE 4
3987 021644 001004                BNE     PERR15
3988 021646 012737 000324 001246      MOV      #324,$TMP6
3989 021654 000407                BR      PERR17
3990 021656 022700 021605      PERR15: CMP    #PDATIO-241-2,R0    ;WAS IT FSRC MODE 5
3991 021662 001401                BEQ     PERR16
3992 021664 000416                BR      PERR20
3993 021666 012737 000325 001246      PERR16: MOV    #325,$TMP6
3994
3995 021674 012737 000627 001244      PERR17: MOV    #627,$TMP5          ;REPORT FSRC
3996 021702 012737 000326 001250      MOV      #326,$TMP7          ;FLOWS FAILURE.
3997 021710 011637 001236      MOV      (SP),$TMP2
3998 021714 022626                CMP     (SP)+,(SP)+
3999 021716 104101                1$:      ERROR  +101
4000 021720 000463                BR      PDONE
4001
4002 021722 011637 001236      PERR20: MOV    (SP),$TMP2          ;REPORT R0 AFFECTED
4003 021726 022626                CMP     (SP)+,(SP)+
4004 021730 000403                BR      PERR22
4005 021732 012737 021442 001236      PERR21: MOV    #P2,$TMP2
4006 021740                PERR22:
4007 021740 010037 001240      MOV      R0,$TMP3
4008 021744 012737 021607 001242      MOV      #PDATIO-241,$TMP4
4009 021752 104102                1$:      ERROR  +102
4010 021754 000445                BR      PDONE
4011
4012 021756                PERR1:          ;DATA FAILURE.
4013 021756 012737 021442 001236      MOV      #P2,$TMP2
4014 021764 012737 022050 001240      MOV      #PDATIO,$TMP3
4015 021772 012737 022060 001242      MOV      #PDAT00,$TMP4
4016 022000 104104                1$:      ERROR  +104
4017 022002 000432                BR      PDONE
4018
4019 022004                PERR2:          ;FSRC FAILURE TO
4020 022004 012737 021442 001236      MOV      #P2,$TMP2          ;MODE 0
4021 022012 012737 000627 001244      MOV      #627,$TMP5
4022 022020 012737 000326 001250      MOV      #326,$TMP7
4023 022026 012737 000320 001246      MOV      #320,$TMP6
4024 022034 104103                1$:      ERROR  +103
4025 022036 000414                BR      PDONE
4026
4027 022040 177777      PPAT10: .WORD  -1
4028 022042 177777      PPAT11: .WORD  -1
4029 022044 177777      PPAT12: .WORD  -1
4030 022046 177777      PPAT13: .WORD  -1
4031
4032 022050 010421      PDATIO: .WORD  010421
4033 022052 031463      PDATIO: .WORD  031463
4034 022054 052525      PDATIO: .WORD  052525
4035 022056 073567      PDATIO: .WORD  073567
4036

```



4037 022060 000000  
4038 022062 000000  
4039 022064 000000  
4040 022066 000000  
4041  
4042 022070  
022070 104413

PDATA0: .WORD 0  
PDATA1: .WORD 0  
PDATA2: .WORD 0  
PDATA3: .WORD 0

PDONE: 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?).

4043  
4050



4051

```
.SBTTL TEST # 23 - FSRC MODE 7 TEST  
:*****  
:*TEST_23 . FSRC MODE 7 TEST  
:*  
:* THIS IS A TEST OF FSRC MODE 7, INDEX  
:* DEFERRED MODE.  
:*  
:*****  
TST23: SCOPE
```

022072 000004  
4052  
4053 022074  
4054 022074 170011  
4055  
4056 022076 012700 022530  
4057 022102 172410  
4058  
4059 022104 012737 022236 000004  
4060  
4061  
4062  
4063 022112 012700 022277  
4064  
4065 022116 172470 000241  
4066 022120  
4067  
4068 022122 012701 022550  
4069 022126 174011  
4070  
4071 022130 012703 000004  
4072 022134 012704 022550  
4073 022140 012705 022560  
4074 022144 022425  
4075 022146 001007  
4076 022150 077303  
4077  
4078 022152 022700 022277  
4079 022156 001401  
4080 022160 000514  
4081 022162 000137 022570  
4082  
4083 022166 012701 022550  
4084 022172 012703 000004  
4085 022176 022721 177777  
4086 022202 001002  
4087 022204 077304  
4088 022206 000513  
4089  
4090 022210 012701 022540  
4091 022214 012702 022550  
4092 022220 012703 000004  
4093 022224 022122  
4094 022226 001401  
4095 022230 000524  
4096 022232 077304  
4097 022234 000504  
4098  
4099

```
Q1: SETD  
MOV #QPAT10,R0  
LDD (R0),AC0 ;LOAD A DEFAULT  
;PATTERN INTO AC0  
MOV #QERRO,ERRVECT ;IF THE (BUT FSRC)  
;FORK FAILS AN  
;ODD ADR TRAP COULD  
;OCCUR  
MOV #QPAT20-241,R0  
Q2: LDD @241(R0),AC0  
Q3=Q2+2  
Q4: MOV #QDAT00,R1  
STD AC0,(R1) ;GET THE DATA  
MOV #4,R3  
MOV #QDAT00,R4  
MOV #QDAT10,R5  
Q5: CMP (R4)+,(R5)+ ;CHECK THE DATA  
BNE Q6  
SOB R3,Q5  
CMP #QPAT20-241,R0 ;CHECK R0.  
BEQ 1$  
BR QERR21  
1$: JMP QDONE  
Q6: MOV #QDAT00,R1  
MOV #4,R3  
Q7: CMP #-1,(R1)+ ;WAS IT FSRC MODE 0?  
BNE Q8  
SOB R3,Q7  
BR QERR2  
Q8: MOV #QPAT20,R1  
MOV #QDAT00,R2  
MOV #4,R3  
Q9: CMP (R1)+,(R2)+ ;WAS IT FSRC 6  
;OR DATA FAILURE  
BEQ Q10  
BR QERR1  
Q10: SOB R3,Q9  
BR QERR3
```

;TRAP TO HERE ON AN ODD ADR FAILURE



```

4100
4101 022236 021627 021444      QERR0:  CMP      (SP),#P3
4102 022242 000137 042120      JMP      CPSPUR
4103
4104 022246 022700 022277      QERR11: CMP      #QPAT20-241,R0 ;WAS IT FSRC
4105 022252 001004      BNE     QERR12 ;MODE 1?
4106 022254 012737 000321 001246      MOV     #321,$TMP6
4107 022262 000434      BR     QERR17
4108 022264 022700 022307      QERR12: CMP      #QPAT20-241+10,R0 ;WAS IT FSRC
4109 022270 001004      BNE     QERR13 ;MODE 2?
4110 022272 012737 000322 001246      MOV     #322,$TMP6
4111 022300 000425      BR     QERR17
4112 022302 022700 022301      QERR13: CMP      #QPAT20-241+2,R0 ;WAS IT FSRC
4113 022306 001004      BNE     QERR14 ;MODE 3?
4114 022310 012737 000323 001246      MOV     #323,$TMP6
4115 022316 000416      BR     QERR17
4116 022320 022700 022267      QERR14: CMP      #QPAT20-241-10,R0 ;WAS IT FSRC
4117 022324 001004      BNE     QERR15 ;MODE 4
4118 022326 012737 000324 001246      MOV     #324,$TMP6
4119 022334 000407      BR     QERR17
4120
4121 022336 022700 022275      QERR15: CMP      #QPAT20-241-2,R0 ;WAS IT FSRC
4122 022342 001401      BEQ     QERR16 ;MODE 5
4123 022344 000416      BR     QERR20
4124
4125 022346 012737 000325 001246      QERR16: MOV     #325,$TMP6
4126
4127 022354 012737 000627 001244      QERR17: MOV     #627,$TMP5 ;REPORT FSRC FAILURE
4128 022362 012737 000327 001250      MOV     #327,$TMP7
4129 022370 011637 001236      MOV     (SP),$TMP2
4130 022374 022626      CMP     (SP)+,(SP)+
4131 022376 104105      1$:     ERROR   +105
4132 022400 000473      BR     QDONE
4133
4134 022402 011637 001236      QERR20: MOV     (SP),$TMP2 ;REPORT R0 AFFECTED.
4135 022406 022626      CMP     (SP)+,(SP)+
4136 022410 000403      BR     QERR22
4137 022412 012737 022116 001236      QERR21: MOV     #Q2,$TMP2
4138 022420      QERR22:
4139 022420 010037 001240      MOV     R0,$TMP3
4140 022424 012737 022277 001242      MOV     #QPAT20-241,$TMP4
4141 022432 104106      1$:     ERROR   +106
4142 022434 000455      BR     QDONE
4143
4144 022436 012737 000320 001246      QERR2:  MOV     #320,$TMP6 ;WENT TO FSRC
4145 022444 000403      BR     QERR4 ;MODE 0
4146 022446 012737 000326 001246      QERR3:  MOV     #326,$TMP6 ;WENT TO FSRC
4147      ;MODE 6
4148 022454 012737 000627 001244      QERR4:  MOV     #627,$TMP5
4149 022462 012737 000327 001250      MOV     #327,$TMP7
4150 022470 012737 022116 001236      MOV     #Q2,$TMP2
4151 022476 104107      1$:     ERROR   +107
4152 022500 000433      BR     QDONE
4153
4154 022502      QERR1:  ;DATA FAILURE
4155 022502 012737 022116 001236      MOV     #Q2,$TMP2
4156 022510 012737 022560 001240      MOV     #QDAT10,$TMP3

```



```
4157 022516 012737 022550 001242      MOV      #QDAT00,$TMP4
4158 022524 104110      1$:      ERROR    +110
4159 022526 000420      BR       QDONE
4160
4161 022530 177777      QPAT10: .WORD   -1
4162 022532 177777      QPAT11: .WORD   -1
4163 022534 177777      QPAT12: .WORD   -1
4164 022536 177777      QPAT13: .WORD   -1
4165
4166 022540 022560      QPAT20: .WORD   QDAT10
4167 022542 052525      QPAT21: .WORD   52525
4168 022544 052525      QPAT22: .WORD   52525
4169 022546 052525      QPAT23: .WORD   52525
4170
4171 022550 000000      QDAT00: .WORD   0
4172 022552 000000      QDAT01: .WORD   0
4173 022554 000000      QDAT02: .WORD   0
4174 022556 000000      QDAT03: .WORD   0
4175
4176 022560 073567      QDAT10: .WORD   073567
4177 022562 052525      QDAT11: .WORD   052525
4178 022564 031463      QDAT12: .WORD   031463
4179 022566 010421      QDAT13: .WORD   010421
4180
4181 022570      QDONE:
      022570 104413      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?).
```

4182



4199

SBTTL TEST # 24 - (BUT EZBT Y8), (BUT ENBT) AND (BUT FIUV) TEST  
 \*\*\*\*\*  
 TEST 24 (BUT EZBT Y8), (BUT ENBT) AND (BUT FIUV) TEST  
 \*\*\*\*\*  
 THIS IS A TEST OF THE (BUT EZBT Y8) FORK, THE  
 (BUT ENBT) FORK AND (BUT FIUV) FORK IN THE  
 LOAD INSTRUCTION FLOWS.  
 EACH OF THE PATTERNS:  
 0  
 +NUM  
 -NUM  
 -0  
 IS LOADED TWICE, ONCE WITH AC>0 THEN  
 WITH AC=0. AFTER EACH LOAD THE FPS IS  
 CHECK TO INSURE THAT CONTROL WAS PASSED  
 THROUGH WITH THE FORKS PROPERLY.  
 \*\*\*\*\*

4200	022572	000004		
4201	022574	005037	027716	
4202	022600	012700	025646	
4203	022604	012701	000004	
4204	022610	012720	177777	
4205				
4206	022616	012737	000033	023720
4207	022624	012737	000023	023722
4208	022632	012737	023376	000244
4209	022640			
	022640	012737	022646	001110
4210	022646	012700	000200	
4211	022652	170100		
4212	022654	012700	023646	
4213	022660	172410		
4214	022662	013737	023720	023724
4215	022670	012737	000001	023726
4216	022676	012777	000254	023730
4217				
4218	022704	012700	023656	
4219	022710	172410		
4220	022712	010037	001252	
4221	022716	012737	022710	001236
4222				
4223	022724	012704	000204	
4224	022730	170205		
4225				
4226	022732	020405		
4227	022734	001402		
4228	022736	000137	023422	
4229				
4230	022742			
	022742	012737	022750	001110
4231	022750	012700	000200	
4232	022754	170100		
4233				
4234	022756	012700	023646	
4235	022762	172410		

TST24:	SCOPE		
	CLR	UFLAG	
	MOV	#UPAT00,R0	:SET UP AC#0 DATA.
	MOV	#4,R1	
U0:	MOV	#-1,(R0)+	
	SOB	R1,U0	
	MOV	#033,UTMP1	
	MOV	#023,UTMP2	
	MOV	#UERR0,FPVECT	:IN CASE (BUT FIUV FAILS)
U1:	MOV	#1\$, \$LPERR	:SET UP THE LOOP ON ERROR ADDRESS.
1\$:	MOV	#200,R0	
	LDFPS	R0	
	MOV	#UPAT00,R0	:LOAD AC0
	LDD	(R0),AC0	
	MOV	UTMP1,UROM1	
	MOV	#001,UROM2	
	MOV	#254,UROM3	
	MOV	#UPAT10,R0	:LOAD 0 INTO AC0
U2:	LDD	(R0),AC0	
	MOV	R0,\$TMP10	
	MOV	#J2,\$TMP2	
	MOV	#204,R4	
	STFPS	R5	:SEE IF FPS IS CORRECT
	CMP	R4,R5	
	BEQ	U3	
	JMP	UERR1	
U3:	MOV	#1\$, \$LPERR	:SET UP THE LOOP ON ERROR ADDRESS.
1\$:	MOV	#200,R0	
	LDFPS	R0	
	MOV	#UPAT00,R0	:LOAD AC0
	LDD	(R0),AC0	



4236	022764	013737	023722	023724	MOV	UTMP2,UROM1	
4237	022772	012737	000003	023726	MOV	#003,UROM2	
4238	023000	012737	000054	023730	MOV	#054,UROM3	
4239							
4240	023006	012700	023666		MOV	#UPAT20,R0	:LOAD A POSITIVE NUMBER
4241							:INTO ACO
4242	023012	172410			U4: LDD	(R0),ACO	
4243	023014	010037	001252		MOV	R0,\$TMP10	
4244	023020	012737	023012	001236	MOV	#U4,\$TMP2	
4245	023026	012704	000200		MOV	#200,R4	:FPS CORRECT?
4246	023032	170205			STFPS	R5	
4247	023034	020405			CMP	R4,R5	
4248	023036	001402			BEQ	U5	
4249	023040	000137	023506		JMP	UERR2	
4250	023044				U5:		
	023044	012737	023052	001110	MOV	#1\$, \$LPERR	:SET UP THE LOOP ON ERROR ADDRESS.
4251	023052	012700	000200		1\$: MOV	#200,R0	
4252	023056	170100			LDFPS	R0	
4253	023060	012700	023646		MOV	#UPAT00,R0	:LOAD ACO
4254	023064	172410			LDD	(R0),ACO	
4255	023066	013737	023722	023724	MOV	UTMP2,UROM1	
4256	023074	012737	000403	023726	MOV	#403,UROM2	
4257	023102	012737	000056	023730	MOV	#056,UROM3	
4258	023110	012700	023676		MOV	#UPAT30,R0	:LOAD A NEGATIVE
4259							:NUMBER INTO ALO
4260	023114	172410			U6: LDD	(R0),ACO	
4261	023116	010037	001252		MOV	R0,\$TMP10	
4262	023122	012737	023114	001236	MOV	#U6,\$TMP2	
4263	023130	012704	000210		MOV	#210,R4	:FPS CORRECT
4264	023134	170205			STFPS	R5	
4265	023136	020405			CMP	R4,R5	
4266	023140	001402			BEQ	U7	
4267	023142	000137	023506		JMP	UERR2	
4268	023146				U7:		
	023146	012737	023154	001110	MOV	#1\$, \$LPERR	:SET UP THE LOOP ON ERROR ADDRESS.
4269	023154	012700	000200		1\$: MOV	#200,R0	
4270	023160	170100			LDFPS	R0	
4271	023162	012700	023646		MOV	#UPAT00,R0	:LOAD ACO
4272	023166	172410			LDD	(R0),ACO	
4273	023170	013737	023720	023724	MOV	UTMP1,UROM1	
4274	023176	012737	000401	023726	MOV	#401,UROM2	
4275	023204	012737	000256	023730	MOV	#256,UROM3	
4276	023212	012700	023706		MOV	#UPAT40,R0	:LOAD -0 INTO ACO
4277	023216	172410			U10: LDD	(R0),ACO	
4278	023220	000240			U11: NOP		:TRAP FROM HERE IF
4279	023222	010037	001252		MOV	R0,\$TMP10	
4280	023226	012737	023216	001236	MOV	#U10,\$TMP2	:(BUT FIUV) FAULS!
4281	023234	012704	000214		MOV	#214,R4	:SEE IF FPS IS CORRECT.
4282	023240	170205			STFPS	R5	
4283	023242	020405			CMP	R4,R5	
4284	023244	001402			BEQ	U12	
4285	023246	000137	023422		JMP	UERR1	
4286	023252	005737	023716		U12: TST	UFLAG	:SEE IF ALL THE PATTERNS
4287	023256	001021			BNE	U14	:HAVE BEEN TEST WITH
4288							:BOTH AC NOT EQUAL TO 0 AND AC=0
4289	023260	012700	023646		MOV	#UPAT00,R0	:IF NOT GO BACK AND
4290	023264	012701	000004		MOV	#4,R1	:CHECK THEM WITH AC=0



```

4291 023270 005020          U13: CLR      (R0)+
4292 023272 077102          SOB      R1,U13
4293 023274 012737 177777 023716  MOV      #-1,UFLAG
4294 023302 012737 000233 023720  MOV      #233,UTMP1
4295 023310 012737 000223 023722  MOV      #223,UTMP2
4296 023316 000137 022640          JMP      U1
4297 023322          U14:
      023322 012737 023330 001110  MOV      #1$, $LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
4298          ;NOW SEE IF A TRAP CAN BE FORCED BY SETTING FIUV AND LOADING -0
4299 023330 012737 023572 000244 1$: MOV      #UERR3,FPVECT
4300 023336 012700 004200          MOV      #4200,R0 ;SET FD AND FIUV
4301 023342 170100          LDFPS   R0
4302 023344 012700 023646          MOV      #UPAT00,R0 ;SET UP ACO
4303 023350 172410          LDD     (R0),ACO
4304 023352 012700 023706          MOV      #UPAT40,R0 ;LOAD -0
4305 023356 172410          U15: LDD     (R0),ACO ;SHOULD TRAP TO 244
4306 023360 170000          U16: CFCC
4307 023362 000240          NOP
4308 023364 012737 023356 001236  MOV      #U15,$TMP2 ;REPORT ERROR.
4309          ;DIDN'T TRAP
4310 023372 104127          1$: ERROR  +127 ;(BUT FIUV) FAILED.
4311 023374 000556          BR      UDONE
4312
4313          ;TRAPPED TO 244. DID (BUT FIUV) FAIL?
4314 023376 021627 023220  UERR0: CMP      (SP),#U11
4315 023402 001402          BEQ     1$
4316 023404 000137 042066          JMP     FPSPUR
4317 023410 011637 001236          1$: MOV      (SP),$TMP2
4318 023414 022626          CMP     (SP)+,(SP)+
4319 023416 104126          2$: ERROR  +126
4320 023420 000544          BR      UDONE
4321
4322          ;COME HERE TO ANALYZE FPS ERRORS
4323
4324 023422 032705 000004          UERR1: BIT      #4,R5
4325 023426 001432          BEQ     UERR20
4326 023430 012737 000443 001244  UERR10: MOV     #443,$TMP5
4327 023436 013703 023730          MOV     UROM3,R3
4328 023442 010337 001250          MOV     R3,$TMP7
4329 023446 032703 000200          BIT     #200,R3
4330 023452 001403          BEQ     1$
4331 023454 042703 000200          BIC     #200,R3
4332 023460 000402          BR      2$
4333 023462 052703 000200          1$: BIS     #200,R3
4334 023466 010337 001246          2$: MOV     R3,$TMP6
4335 023472 010537 001240          UERR11: MOV    R5,$TMP3
4336 023476 010437 001242          MOV     R4,$TMP4
4337 023502 104124          1$: ERROR  +124
4338 023504 000512          BR      UDONE
4339 023506 032705 000004          UERR2: BIT      #4,R5
4340 023512 001746          BEQ     UERR10
4341 023514 013737 023724 001244  UERR20: MOV    UROM1,$TMP5
4342 023522 013703 023726          MOV     UROM2,R3
4343 023526 010337 001250          MOV     R3,$TMP7
4344 023532 032703 000400          BIT     #400,R3
4345 023536 001403          BEQ     1$
4346 023540 042703 000400          BIC     #400,R3
    
```







4411

.SBTTL TEST # 25 - ADDF, ADD, SUBF AND SUBD WITH FSRC=AC=0 TEST  
 \*\*\*\*\*  
 \*TEST 25 ADDF, ADD, SUBF AND SUBD WITH FSRC=AC=0 TEST

\*\*\*\*\*  
 \* THIS IS A TEST OF ADD AND SUB WITH FSRC=AC=0  
 \*\*\*\*\*

4412	023732	000004			TST25: SCOPE	
	023734				W1:	
	023734	012737	023742	001110	MOV #1\$, \$LPERR	;SET UP THE LOOP ON ERROR ADDRESS.
4413	023742	012700	000200		1\$: MOV #200, R0	
4414	023746	170100			LDFPS R0	;SET DOUBLE MODE
4415	023750	012700	024506		MOV #WPAT00, R0	;LOAD AC0=:
4416	023754	172410			LDD (R0), AC0	
4417	023756	012737	023770	001236	MOV #W2, \$TMP2	
4418	023764	012700	024506		MOV #WPAT00, R0	
4419	023770	172010			W2: ADD (R0), AC0	;TEST INSTRUCTION.
4420	023772	170205			STFPS R5	;GET FPS
4421	023774	170011			SETD	;SET DOUBLE MODE
4422	023776	012700	024506		MOV #WPAT00, R0	
4423	024002	174010			STD AC0, (R0)	;GET THE RESULT
4424	024004	012701	024506		MOV #WPAT00, R1	
4425	024010	012702	000004		MOV #4, R2	
4426	024014	022021			W3: CMP (R0)+, (R1)+	;IS RESULT CORRECT
4427	024016	001405			BEQ W4	
4428						;NO
4429	024020	004737	024454		JSR PC, WSETUP	
4430	024024	104133			1\$: ERROR +133	
4431	024026	000137	024526		JMP WDONE	
4432	024032	077210			W4: SOB R2, W3	
4433	024034	022705	000204		CMP #204, R5	;IS FPS CORRECT
4434	024040	001410			BEQ W5	
4435						;NO
4436	024042	012737	000204	001242	MOV #204, \$TMP4	
4437	024050	010537	001240		MOV R5, \$TMP3	
4438	024054	104137			1\$: ERROR +137	
4439	024056	000137	024526		JMP WDONE	
4440	024062				W5:	
	024062	012737	024070	001110	MOV #1\$, \$LPERR	;SET UP THE LOOP ON ERROR ADDRESS.
4441	024070	012700	000200		1\$: MOV #200, R0	
4442	024074	170100			LDFPS R0	;SET DOUBLE MODE
4443	024076	012700	024506		MOV #WPAT00, R0	;LOAD AC0=0
4444	024102	172410			LDD (R0), AC0	
4445	024104	012737	024122	001236	MOV #W6, \$TMP2	
4446	024112	005000			CLR R0	
4447	024114	170100			LDFPS R0	;GO TO FLOATING MODE
4448	024116	012700	024506		MOV #WPAT00, R0	
4449	024122	172010			W6: ADD (R0), AC0	;TEST INSTRUCTION
4450	024124	170205			STFPS R5	;GET FPS
4451	024126	170011			SETD	;RESET TO DOUBLE MODE
4452	024130	012700	024506		MOV #WPAT00, R0	
4453	024134	174010			STD AC0, (R0)	;GET THE RESULT
4454	024136	012701	024506		MOV #WPAT00, R1	
4455	024142	012702	000004		MOV #4, R2	
4456	024146	022021			W7: CMP (R0)+, (R1)+	;WAS THE RESULT







```

4512 024412 022021          W17:  CMP      (R0)+,(R1)+      ;IS RESULT CORRECT?
4513 024414 001404          BEQ      W20
4514                                     ;NO.
4515 024416 004737 024454          JSR      PC,WSETUP
4516 024422 104136          1$:  ERROR   +136
4517 024424 000440          BR       WDONE
4518 024426 077207          W20:  SOB      R2,W17
4519 024430 022705 000004          CMP      #4,R5      ;IS FPS CORRECT?
4520 024434 001434          BEQ      WDONE
4521                                     ;NO
4522 024436 012737 000004 001242          MOV      #4,$TMP4
4523 024444 010537 001240          MOV      R5,$TMP3
4524 024450 104142          1$:  ERROR   +142
4525 024452 000425          BR       WDONE
4526                                     ;SET UP FOR ERROR CALL
4527
4528
4529 024454 012737 024506 001240          WSETUP: MOV      #WPAT00,$TMP3
4530 024462 012737 024506 001242          MOV      #WPAT00,$TMP4
4531 024470 012737 024506 001246          MOV      #WPAT00,$TMP6
4532 024476 012737 024506 001244          MOV      #WPAT00,$TMP5
4533 024504 000207          RTS      PC
4534 024506 000000          WPAT00: .WORD   0
4535 024510 000000          WPAT01: .WORD   0
4536 024512 000000          WPAT02: .WORD   0
4537 024514 000000          WPAT03: .WORD   0
4538
4539 024516 000000          WDAPO0: .WORD   0
4540 024520 000000          WDAT01: .WORD   0
4541 024522 000000          WDAT02: .WORD   0
4542 024524 000000          WDAT03: .WORD   0
4543
4544 024526          WDONE:
         024526 104413          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?).
    
```

4545  
4546



4553

```

.SBTTL TEST # 26 - ADDD AND SUB WITH FSRC=0
*****
*TEST 26      ADDD AND SUB WITH FSRC=0
*
* THIS IS A TEST OF ADDD AND SUB WITH FSRC=0.
*
    
```

```

*****
TST26: SCOPE
X1:
4554 024530 000004
4554 024532
4554 024532 012737 024540 001110
4555 024540 012700 000200
4556 024544 170100
4557 024546 012700 025312
4558 024552 010037 025300
4559 024556 172410
4560 024560 012737 024572 001236
4561 024566 012700 025322
4562 024572 172010
4563 024574 170205
4564 024576 170011
4565 024600 012700 025302
4566 024604 174010
4567 024606 012701 025312
4568 024612 012702 000004
4569 024616 022021
4570 024620 001401
4571 024622 000561
4572 024624 077204
4573 024626 012704 000200
4574 024632 020405
4575 024634 001402
4576 024636 000137 025250
4577 024642
4578 024642 012737 024650 001110
4578 024650 012700 000200
4579 024654 170100
4580 024656 012700 025332
4581 024662 010037 025300
4582 024666 172410
4583 024670 012737 024702 001236
4584 024676 012700 025322
4585 024702 172010
4586 024704 170205
4587 024706 170011
4588 024710 012700 025302
4589 024714 174010
4590 024716 012701 025332
4591 024722 012702 000004
4592 024726 022021
4593 024730 001401
4594 024732 000515
4595 024734 077204
4596 024736 012704 000210
4597 024742 020405
4598 024744 001401
4599 024746 000540

X1:      MOV      #1$, $LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
1$:      MOV      #200,R0
          LDFPS   R0              ;SET DOUBLE MODE
          MOV      #XPAT00,R0      ;SET ACO TO POSITIVE
          MOV      R0,XTMP         ;NUMBER #0
          LDD     (R0),ACO
          MOV      #X2,$TMP2
          MOV      #XPAT10,R0      ;FSRC=0
X2:      ADDD    (R0),ACO         ;TEST INSTRUCTION
          STFPS   R5
          SETD
          MOV      #XDAT00,R0      ;GET RESULT.
          STD     ACO,(R0)
          MOV      #XPAT00,R1
          MOV      #4,R2
X3:      CMP     (R0)+,(R1)+      ;IS RESULT CORRECT?
          BEQ     X4
          BR      XERR1
X4:      SOB     R2,X3
          MOV      #200,R4
          CMP     R4,R5            ;IS FPS CORRECT?
          BEQ     X5
          JMP     XERR2
X5:      MOV      #1$, $LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
1$:      MOV      #200,R0
          LDFPS   R0              ;SET DOUBLE MODE
          MOV      #XPAT20,R0      ;SET ACO TO
          MOV      R0,XTMP         ;NEGATIVE NUMBER
          LDD     (R0),ACO
          MOV      #X6,$TMP2
          MOV      #XPAT10,R0      ;FSRC=0
X6:      ADDD    (R0),ACO         ;TEST INSTRUCTION
          STFPS   R5
          SETD
          MOV      #XDAT00,R0      ;GET RESULT
          STD     ACO,(R0)
          MOV      #XPAT20,R1
          MOV      #4,R2
X7:      CMP     (R0)+,(R1)+      ;IS RESULT CORRECT?
          BEQ     X10
          BR      XERR1
X10:     SOB     R2,X7
          MOV      #210,R4
          CMP     R4,R5            ;IS FPS CORRECT?
          BEQ     X11
          BR      XERR2
    
```



```

4600 024750          012737 024756 001110 X11:  MOV      #1$, $LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
      024750          012700 000200 1$:   MOV      #200,R0
4601 024756          170100          LDFPS   R0                ;SET DOUBLE MODE
4602 024762          012700 025312          MOV      #XPAT00,R0       ;SET ACO TO NON-ZERO
4603 024764          010037 025300          MOV      R0,XTMP         ;POSITIVE NUMBER
4604 024770          172410          LDD     (R0),ACO
4605 024774          012737 025010 001236  MOVL   #X12,$TMP2
4606 024776          012700 025322          MOV      #XPAT10,R0      ;FSRC=0
4607 025004          173010          SUBD   (R0),ACO         ;TEST INSTRUCTION
4608 025010          170205          STFPS  R5
4609 025012          170011          SETD
4610 025014          012700 025302          MOV      #XDAT00,R0      ;GET RESULT
4611 025016          174010          STD   ACO,(R0)
4612 025022          012701 025312          MOV      #XPAT00,R1
4613 025024          012702 000004          MOV      #4,R2
4614 025030          022021          X13:  CMP     (R0)+,(R1)+   ;IS RESULT CORRECT?
4615 025034          001401          BEQ    X14
4616 025036          000465          BR     XERR3
4617 025040          077204          X14:  SOB   R2,X13
4618 025042          012704 000200          MOV      #200,R4        ;IS FPS CORRECT?
4619 025044          020405          CMP    R4,R5
4620 025050          001401          BEQ    X15
4621 025052          000503          BR     XERR4
4622 025054
4623 025056          012737 025064 001110 X15:  MOV      #1$, $LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
      025056          012700 000200 1$:   MOV      #200,R0
4624 025064          170100          LDFPS   R0                ;SET DOUBLE MODE
4625 025070          012700 025332          MOV      #XPAT20,R0     ;SET ACO=A NEGATIVE
4626 025072          010037 025300          MOV      R0,XTMP         ;NUMBER
4627 025076          172410          LDD     (R0),ACO
4628 025102          012737 025116 001236  MOVL   #X16,$TMP2
4629 025104          012700 025322          MOV      #XPAT10,R0     ;FSRC=0
4630 025112          173010          SUBD   (R0),ACO         ;TEST INSTRUCTION.
4631 025116          170205          STFPS  R5
4632 025120          170011          SETD
4633 025122          012700 025302          MOV      #XDAT00,R0      ;GET RESULT
4634 025124          174010          STD   ACO,(R0)
4635 025130          012701 025332          MOV      #XPAT20,R1
4636 025132          012702 000004          MOV      #4,R2
4637 025136          022021          X17:  CMP     (R0)+,(R1)+   ;IS RESULT CORRECT?
4638 025142          001401          BEQ    X20
4639 025144          000422          BR     XERR3
4640 025146          077204          X20:  SOB   R2,X17
4641 025150          012704 000210          MOV      #210,R4        ;IS FPS CORRECT?
4642 025152          020405          CMP    R4,R5
4643 025156          001401          BEQ    X21
4644 025160          000440          BR     XERR4
4645 025162          000466          X21:  BR     XDONE
4646 025164
4647
4648          ;REPORT DATA ERRORS
4649
4650 025166          012737 025322 001240 XERR1: MOV      #XPAT10,$TMP3
4651 025174          012737 025300 001242          MOV      XTMP,$TMP4
4652 025202          012737 025302 001244          MOV      #XDAT00,$TMP5
4653 025210          104143          1$:   ERROR  +143
4654 025212          000453          BR     XDONE
    
```



```

4655 025214 012737 025322 001240 XERR3: MOV #XPAT10,$TMP3
4656 025222 013737 025300 001242 MOV XTMP,$TMP4
4657 025230 012737 025302 001244 MOV #XDAT00,$TMP5
4658 025236 013737 025300 001246 MOV XTMP,$TMP6
4659 025244 104144 1$: ERROR +144
4660 025246 000435 BR XDONE
4661
4662 ;REPORT FPS ERRORS
4663
4664 025250 XERR2:
4665 025250 010537 001240 MOV R5,$TMP3
4666 025254 010437 001242 MOV R4,$TMP4
4667 025260 104145 1$: ERROR +145
4668 025262 000427 BR XDONE
4669 025264 XERR4:
4670 025264 010537 001240 MOV R5,$TMP3
4671 025270 010437 001242 MOV R4,$TMP4
4672 025274 104146 1$: ERROR +146
4673 025276 000421 BR XDONE
4674 025300 XTMP: .WORD 0
4675 025302 XDAT00: .WORD 0
4676 025304 XDAT01: .WORD 0
4677 025306 XDAT02: .WORD 0
4678 025310 XDAT03: .WORD 0
4679
4680 025312 010421 XPAT00: .WORD 010421
4681 025314 021042 XPAT01: .WORD 021042
4682 025316 031463 XPAT02: .WORD 031463
4683 025320 042104 XPAT03: .WORD 042104
4684
4685 025322 000000 XPAT10: .WORD 0
4686 025324 000000 XAPT11: .WORD 0
4687 025326 000000 XPAT12: .WORD 0
4688 025330 000000 XPAT13: .WORD 0
4689 025332 104210 XPAT20: .WORD 104210
4690 025334 114631 XPAT21: .WORD 114631
4691 025336 125252 XPAT22: .WORD 125252
4692 025340 135673 XPAT23: .WORD 135673
4693
4694 025342 XDONE:
025342 104413 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?).
    
```

4695



4703

```

.SBTTL TEST # 27 - SUBD WITH AC=0 TEST
:*****
:*TEST 27 SUBD WITH AC=0 TEST
:*
:* THIS IS A TEST OF SUBD WITH AC=0. BOTH POSITIVE
:* AND NEGATIVE FSRC'S ARE TRIED.
:*
    
```

```

025344 000004
4704 025346 005037 025702
4705 025352 012737 025722 025704
4706 025360 012737 025732 025706
4707 025366 012737 000210 025710
4708 025374
      025374 012737 025402 001110
4709 025402 012700 000200
4710 025406 170100
4711 025410 012700 025742
4712 025414 172410
4713 025416 013700 025704
4714 025422 173010
4715 025424 170205
4716 025426 170011
4717 025430 012700 025712
4718 025434 174010
4719 025436 012702 000004
4720 025442 013701 025706
4721 025446 022021
4722 025450 001026
4723 025452 077203
4724 025454 023705 025710
4725 025460 001401
4726 025462 000475
4727 025464 005737 025702
4728 025470 001015
4729 025472 012737 177777 025702
4730 025500 012737 025732 025704
4731 025506 012737 025722 025706
4732 025514 012737 000200 025710
4733 025522 000724
4734 025524 000512
4735 025526 012702 000004
4736 025532 012700 025704
4737 025536 012701 025712
4738 025542 022021
4739 025544 001002
4740 025546 077203
4741 025550 000421
4742 025552
4743 025552 012737 025422 001236
4744 025560 013737 025704 001240
4745 025566 012737 025742 001242
4746 025574 012737 025712 001244
4747 025602 013737 025706 001246
4748 025610 104147
4749 025612 000457
    
```

```

:*****
TST27: SCOPE
      CLR YFLAG
      MOV #YPAT00,YTMP1 ;P
      MOV #YPAT10,YTMP2 ;N
      MOV #210,YTMP3
Y1:   MOV #1$, $LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
1$:   MOV #200,R0
      LDFPS R0 ;SET DOUBLE MODE
      MOV #YPAT20,R0 ;SET AC0=0
      LDD (R0),AC0
      MOV YTMP1,R0
Y2:   SUBD (R0),AC0 ;TEST INSTRUCTION
      STFPS R5
      SETD
      MOV #YDAT00,R0 ;GET RESULT
      STD AC0,(R0)
      MOV #4,R2
      MOV YTMP2,R1 ;CHECK RESULT.
Y3:   CMP (R0)+,(R1)+
      BNE Y6
      SOB R2,Y3
      CMP YTMP3,R5 ;FPS CORRECT?
      BEQ Y4
      BR YERR3
Y4:   TST YFLAG ;FINISHED TEST?
      BNE Y5
      MOV #-1,YFLAG
      MOV #YPAT10,YTMP1
      MOV #YPAT00,YTMP2
      MOV #200,YTMP3
      BR Y1
Y5:   BR YDONE
Y6:   MOV #4,R2
      MOV #YTMP1,R0 ;DID XOR OF SIGN BIT
      MOV #YDAT00,R1 ;FAIL?
Y7:   CMP (R0)+,(R1)+
      BNE YERR1
      SOB R2,Y7
      BR YERR2
YERR1: MOV #Y2,$TMP2 ;DATA FAILURE
      MOV YTMP1,$TMP3
      MOV #YPAT20,$TMP4
      MOV #YDAT00,$TMP5
      MOV YTMP2,$TMP6
1$:   ERROR +147
      BR YDONE
    
```



4750	025614				YERR2:				:XOR OF SIGN BIT
4751	025614	012737	025422	001236		MOV	#Y2,\$TMP2		:FAILED
4752	025622	013737	025704	001240		MOV	YTMP1,\$TMP3		
4753	025630	012737	025742	001242		MOV	#YPAT20,\$TMP4		
4754	025636	012737	025712	001244		MOV	#YDAT00,\$TMP5		
4755	025644	013737	025706	001246		MOV	YTMP2,\$TMP6		
4756	025652	104150			1\$:	ERROR	+150		
4757	025654	000436				BR	YDONE		
4758	025656				YERR3:				:FPS WRONG.
4759	025656	012737	025422	001236		MOV	#Y2,\$TMP2		
4760	025664	010537	001240			MOV	R5,\$TMP3		
4761	025670	013737	025710	001242		MOV	YTMP3,\$TMP4		
4762	025676	104151			1\$:	ERROR	+151		
4763	025700	000424				BR	YDONE		
4764									
4765	025702	000000			YFLAG:	.WORD	0		
4766	025704	000000			YTMP1:	.WORD	0		
4767	025706	000000			YTMP2:	.WORD	0		
4768	025710	000000			YTMP3:	.WORD	0		
4769									
4770	025712	000000			YDAT00:	.WORD	0		
4771	025714	000000			YDAT01:	.WORD	0		
4772	025716	000000			YDAT02:	.WORD	0		
4773	025720	000000			YDAT03:	.WORD	0		
4774									
4775	025722	063146			YPAT00:	.WORD	063146		
4776	025724	052525			YPAT01:	.WORD	052525		
4777	025726	042104			YPAT02:	.WORD	042104		
4778	025730	167356			YPAT03:	.WORD	167356		
4779									
4780	025732	163146			YPAT10:	.WORD	163146		
4781	025734	052525			YPAT11:	.WORD	052525		
4782	025736	042104			YPAT12:	.WORD	042104		
4783	025740	167356			YPAT13:	.WORD	167356		
4784									
4785	025742	000000			YPAT20:	.WORD	0		
4786	025744	000000			YPAT21:	.WORD	0		
4787	025746	000000			YPAT22:	.WORD	0		
4788	025750	000000			YPAT23:	.WORD	0		
4789									
4790	025752	104413			YDONE:				
	025752					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?).



4799

..SBTTL TEST # 30 - ADDD WITH AC=0 TEST  
 .....  
 :\*TEST 30 ADDD WITH AC=0 TEST

..@ THIS IS A TEST OF ADDD WITH AC=0. BOTH  
 :\* POSITIVE AND NEGATIVE FSRC'S ARE TRIED.  
 :\*

```

025754 000004
4800 025756 005037 026212
4801 025762 012737 026230 026214
4802 025770 012737 000200 026216
4803 025776
      025776 012737 026004 001110
4804 026004 012700 000200
4805 026010 170100
4806 026012 012700 026250
4807 026016 172410
4808 026020 013700 026214
4809 026024 172010
4810 026026 170205
4811 026030 170011
4812 026032 012700 026220
4813 026036 174010
4814 026040 012702 000004
4815 026044 013701 026214
4816 026050 022021
4817 026052 001401
4818 026054 000423
4819 026056 077204
4820 026060 023705 026216
4821 026064 001401
4822 026066 000437
4823 026070 005737 026212
4824 026074 001012
4825 026076 012737 177777 026212
4826 026104 012737 026240 026214
4827 026112 012737 000210 026216
4828 026120 000726
4829 026122 000456
4830 026124
4831 026124 012737 026024 001236
4832 026132 013737 026214 001240
4833 026140 012737 026250 001242
4834 026146 012737 026220 001244
4835 026154 013737 026214 001246
4836 026162 104152
4837 026164 000435
4838 026166
4839 026166 012737 026024 001236
4840 026174 010537 001240
4841 026200 013737 026216 001242
4842 026206 104153
4843 026210 000423
4844
    
```

```

.....
TST30: SCOPE
        CLR      ZFLAG
        MOV      #ZPAT00,ZTMP1 ;P
        MOV      #200,ZTMP2
Z1:     MOV      #1$,$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
        MOV      #200,R0
        LDFPS   R0 ;SET DOUBLE MODE
        MOV      #ZPAT20,R0 ;SET ACO=0
        LDD     (R0),ACO
        MOV      ZTMP1,R0
Z2:     ADDD    (R0),ACO ;TEST INSTRUCTION
        STFPS  R5
        SETD
        MOV      #ZDAT00,R0 ;GET RESULT
        STD     ACO,(R0)
        MOV      #4,R2
        MOV      ZTMP1,R1 ;RESULT CORRECT?
Z3:     CMP     (R0)+,(R1)+
        BEQ     Z4
        BR      ZERR1
Z4:     SOB     R2,Z3 ;FPS CORRECT?
        CMP     ZTMP2,R5
        BEQ     Z5
        BR      ZERR2
Z5:     TST     ZFLAG ;FINISHED TEST?
        BNE     Z6
        MOV     #-1,ZFLAG
        MOV     #ZPAT10,ZTMP1
        MOV     #210,ZTMP2
        BR      Z1
Z6:     BR      ZDONE
ZERR1: ;DATA FAILURE
        MOV     #Z2,$TMP2
        MOV     ZTMP1,$TMP3
        MOV     #ZPAT20,$TMP4
        MOV     #ZDAT00,$TMP5
        MOV     ZTMP1,$TMP6
        ERROR  +152
        BR      ZDONE
ZERR2:
        MOV     #Z2,$TMP2
        MOV     R5,$TMP3
        MOV     ZTMP2,$TMP4
        ERROR  +153
        BR      ZDONE
    
```



4845	026212	000000	ZFLAG:	.WORD	0
4846	026214	000000	ZTMP1:	.WORD	0
4847	026216	000000	ZTMP2:	.WORD	0
4848					
4849	026220	000000	ZDAT00:	.WORD	0
4850	026222	000000	ZDAT01:	.WORD	0
4851	026224	000000	ZDAT02:	.WORD	0
4852	026226	000000	ZDAT03:	.WORD	0
4853					
4854	026230	031463	ZPAT00:	.WORD	031463
4855	026232	010421	ZPAT01:	.WORD	010421
4856	026234	146314	ZPAT02:	.WORD	146314
4857	026236	156735	ZPAT03:	.WORD	156735
4858					
4859	026240	156735	ZPAT10:	.WORD	156735
4860	026242	167356	ZPAT11:	.WORD	167356
4861	026244	135673	ZPAT12:	.WORD	135673
4862	026246	146314	ZPAT13:	.WORD	146314
4863					
4864	026250	000000	ZPAT20:	.WORD	0
4865	026252	000000	ZPAT21:	.WORD	0
4866	026254	000000	ZPAT22:	.WORD	0
4867	026256	000000	ZPAT23:	.WORD	0
4868					
4869	026260		ZDONE:		
	026260	104413		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?).

4870  
4871



4879

.SBTTL TEST # 31 - ADDF & ADDD E(AC)=E(FSRC) & (BUT FT) TEST  
 \*\*\*\*\*  
 \*TEST 31 ADDF & ADDD E(AC)=E(FSRC) & (BUT FT) TEST  
 \*  
 \* THIS IS A TEST OF THE ADD INSTRUCTION WITH THE  
 \* OPERANDS HAVING EQUAL EXPONENTS. THE (BUT FT)  
 \* FORK IN THE ROUND/TRUNK FLOWS IS ALSO TESTED.  
 \*  
 \*\*\*\*\*

```

4880 026262 000004
4880 026264 012737 026272 001110
4881 026272 012700 003240
4882 026276 170100
4883 026300 012737 026660 000244
4884 026306 012700 027236
4885
4886 026312 172410
4887 026314 012737 026326 001236
4888 026322 012700 027246
4889 026326 172010
4890
4891 026330 012700 027226
4892 026334 174010
4893 026336 012701 027256
4894 026342 012702 000004
4895 026346 022021
4896 026350 001414
4897 026352 012700 027266
4898 026356 012701 027226
4 9 026362 012702 000004
4 J 026366 022021
4901 026370 001401
4902 026372 000565
4903 026374 077204
4904 026376 000137 027002
4905 026402 077217
4906
4907
4908
4909 026404
4910 026412 012737 026412 001110
4911 026416 170100
4912 026420 012700 027236
4913 026424 172410
4914 026426 012737 026440 001236
4915 026434 012700 027246
4916 026440 172010
4917
4918 026442 012700 027226
4919 026446 174010
4920 026450 012701 027266
4921 026454 012702 000004
4922 026460 022021
4923 026462 001425
4924 026464 012700 027256
    
```

```

TST31: SCOPE
AA1:  MOV #1$, $LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
      MOV #3240, R0
      LDFPS R0 ;SET FIV FIV FD AND FT
      MOV #AAERRO, FPVECT ;IN CASE THE OVER/UNDER
      MOV #AAPATO, R0 ;FLOWS IN TRAP WILL
      LDD (R0), ACO ;OCCUR
      MOV #AA2, $TMP2 ;SET UP ACO
      MOV #AAPAT1, R0 ;OPERAND
      ADDD (R0), ACO ;TEST INSTRUCTION
      AA2: ;SHOULD TRUNCATE
      AA3: MOV #AADATO, R0
      STD ACO (R0) ;GET THE RESULT
      MOV #AAPAT2, R1
      MOV #4, R2
      AA4: CMP (R0)+, (R1)+ ;CORRECT?
      BEQ AA7
      MOV #AAPAT3, R0 ;DID (BUT FT) FAIL
      MOV #AADATO, R1
      MOV #4, R2
      AA5: CMP (R0)+, (R1)+
      BEQ AA6
      BR AAERR1 ;DATA ERROR
      A16: SOB R2, AA5
      JMP AAERR2 ;(BUT FT) ERROR
      AA7: SOB R2, AA4
      ;NOW TEST DOUBLE FLOATING ROUND MODE.
      AA10: MOV #1$, $LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
      MOV #3200, R0 ;SET FD FIV FIV. FT=0
      LDFPS R0
      MOV #AAPATO, R0
      LDD (R0), ACO ;SET UP ACO OPERAND
      MOV #AA11, $TMP2
      MOV #AAPAT1, R0
      AA11: ADDD (R0), ACO ;TEST INSTRUCTION
      ;SHOULD ROUND
      AA12: MOV #AADATO, R0
      STD ACO (R0) ;GET THE RESULT
      MOV #AAPAT3, R1
      MOV #4, R2
      AA13: CMP (R0)+, (R1)+ ;CORRECT?
      BEQ AA20
      MOV #AAPAT2, R0 ;DID (BUT FT) FAIL?
    
```



```

4925 026470 012701 027226      MOV      #AADATA0,R1
4926 026474 012702 000004      MOV      #4,R2
4927 026500 022021      AA14:   CMP      (R0)+,(R1)+
4928 026502 001413      BEQ      AA17
4929 026504 012700 027276      MOV      #AAPAT4,R0      :WAS THE FLOATING
4930 026510 012701 027226      MOV      #AADATA0,R1      :CONSTANT USED
4931 026514 012702 000004      MOV      #4,R2      :INSTEAD OF THE
4932 026520 022021      AA15:   CMP      (R0)+,(R1)+      :DOUBLE CONSTANT
4933 026522 001401      BEQ      AA16      :IN THE ROUND
4934 026524 000544      BR       AAERR3      :FLOWS?
4935 026526 077204      AA16:   SOB      R2,AA15      :DATA ERROR
4936 026530 000546      BR       AAERR4      :CONSTANT ERROR
4937 026532 077216      AA17:   SOB      R2,AA14
4938 026534 000562      BR       AAERR5      :(BUT FT) ERROR
4939 026536 077230      AA20:   SOB      R2,AA13
4940
4941      ;NOW TEST ADDF WITH FT=0, ROUND MODE
4942
4943 026540      AA21:
4944 026540 012737 026546 001110      MOV      #1$, $LPERR      :SET UP THE LOOP ON ERROR ADDRESS.
4945 026546 012700 003200      1$:     MOV      #3200,R0      ;FIV=1, FIV=1, FT=0
4946 026552 170100      LDFPS   R0
4947 026554 012700 027236      MOV      #AAPAT0,R0      :LOAD ACO OPERAND
4948 026560 172410      LDD     (R0),AC0
4949 026562 170001      SETF
4950 026564 012737 026576 001236      MOV      #AA22,$TMP2      :ENTER FLOATING MODE
4951 026572 012700 027306      MOV      #AAPAT5,R0
4952 026576 172010      AA22:   ADDF   (R0),AC0      :TEST INSTRUCTION
4953 026600      AA23:
4954 026600 170011      SETD
4955      :RESET TO DOUBLE
4956 026602 012700 027226      MOV      #AADATA0,R0      :MODE
4957 026606 174010      STD     AC0,(R0)      :GET THE RESULT
4958 026610 012701 027316      MOV      #AAPAT6,R1      :CORRECT?
4959 026614 012702 000002      MOV      #2,R2
4960 026620 022021      AA24:   CMP      (R0)+,(R1)+
4961 026622 001413      BEQ      AA27
4962 026624 012700 027256      MOV      #AAPAT2,R0      :WAS THE DOUBLE
4963 026630 012701 027226      MOV      #AADATA0,R1      :CONSTANT USED INSTEAD
4964 026634 012702 000002      MOV      #2,R2      :OF THE FLOATING
4965 026640 022011      AA25:   CMP      (R0)+,(R1)      :CONSTANT IN THE
4966 026642 001401      BEQ      AA26      :ROUND FLOWS?
4967 026644 000534      BR       AAERR6      :DATA ERROR
4968 026646 077204      AA26:   SOB      R2,AA25
4969 026650 000550      BR       AAERR7      :CONSTANT ERROR
4970 026652 077216      AA27:   SOB      R2,AA24
4971 026654 000137 027326      JMP     AADONE
4972
4973      ;COME HERE IF A TRAP OCCURS TO 244.
4974
4975 026660 013700 001236      AAERR0: MOV     $TMP2,R0      :SEE IF THE TRAP WAS
4976 026664 005720      TST     (R0)+      :AT A TEST INSTRUCTION
4977 026666 020016      CMP     R0,(SP)
4978 026670 001402      BEQ     1$
4979 026672 000137 042066      10$:   JMP     FPSPUR
4980 026676      1$:
    
```



```

4981 026676 170300          STST      R0          :GET FEC
4982 026700 020027 000010    CMP      R0,#10
4983 026704 001405          BEQ      20$         :OVERFLOW
4984 026706 020027 000012    CMP      R0,#12
4985 026712 001410          BEQ      30$         :UNDERFLOW
4986 026714 000766          BR       10$
4987 026716 026720          20$:
4988 026720 011637 001236    20$: MOV    (SP), $TMP2      :REPORT OVERFLOW ERROR
4989 026724 022626          CMP     (SP)+, (SP)+
4990 026726 104154          21$: ERROR +154
4991 026730 000137 027326    25$: JMP    AADONE
4992 026734 011637 001236    30$: MOV    (SP), $TMP2      :REPORT UNDERFLOW
4993 026740 022626          CMP     (SP)+, (SP)+      :ERROR
4994 026742 104155          31$: ERROR +155
4995 026744 000771          BR       25$
4996
4997
4998 026746 012737 027256 001246    :ADD RESULT INCORRECT
4999 026754 012737 027236 001242    AAERR1: MOV   #AAPAT2, $TMP6
5000 026762 012737 027246 001240    AAERR10: MOV  #AAPAT0, $TMP4
5001 026770 012737 027226 001244    MOV   #AAPAT1, $TMP3
5002 026776 104162          MOV   #AADATO, $TMP5
5003 027000 000552          1$: ERROR +162
5004 027002 012737 027256 001246    BR    AADONE
5005 027010 012737 027236 001242    AAERR2: MOV   #AAPAT2, $TMP6      :(BUT FT) FAILED.
5006 027016 012737 027246 001240    MOV   #AAPAT0, $TMP4
5007 027024 012737 027226 001244    MOV   #AAPAT1, $TMP3
5008 027032 104156          MOV   #AADATO, $TMP5
5009 027034 000534          1$: ERROR +156
5010 027036 012737 027266 001246    BR    AADONE
5011 027044 000743          AAERR3: MOV   #AAPAT3, $TMP6      :DATA ERROR.
5012 027046 012737 027266 001246    BR    AAERR10
5013 027054 012737 027236 001242    AAERR4: MOV   #AAPAT3, $TMP6      :BAD CONSTANT
5014 027062 012737 027246 001240    MOV   #AAPAT0, $TMP4
5015 027070 012737 027226 001244    MOV   #AAPAT1, $TMP3
5016 027076 104160          MOV   #AADATO, $TMP5
5017 027100 000512          1$: ERROR +160
5018 027102 012737 027266 001246    BR    AADONE
5019 027110 012737 027236 001242    AAERR5: MOV   #AAPAT3, $TMP6      :(BUT FT) FAILED.
5020 027116 012737 027246 001240    MOV   #AAPAT0, $TMP4
5021 027124 012737 027226 001244    MOV   #AAPAT1, $TMP3
5022 027132 104157          MOV   #AADATO, $TMP5
5023 027134 000474          1$: ERROR +157
5024 027136 012737 027306 001240    BR    AADONE
5025 027144 012737 027236 001242    AAERR6: MOV   #AAPAT5, $TMP3      :FD=0 AND
5026 027152 012737 027226 001244    MOV   #AAPAT0, $TMP4      :DATA ERROR
5027 027160 012737 027316 001246    MOV   #AADATO, $TMP5
5028 027166 104160          MOV   #AAPAT6, $TMP6
5029 027170 000456          1$: ERROR +160
5030 027172 012737 027306 001240    BR    AADONE
5031 027200 012737 027236 001242    AAERR7: MOV   #AAPAT5, $TMP3      :CONSTANT ERROR
5032 027206 012737 027226 001244    MOV   #AAPAT0, $TMP4
5033 027214 012737 027316 001246    MOV   #AADATO, $TMP5
5034 027222 104161          MOV   #AAPAT6, $TMP6
5035 027224 000440          1$: ERROR +161
5036 027226 000000 000000 000000    BR    AADONE
5037 027236 000200 000000 000000    AADATO: .WORD 0,0,0,0
                    AAPAT0: .WORD 200,0,0,0
    
```



5038	027246	000200	000000	000000	AAPAT1: .WORD	200,0,0,1
5039	027256	000400	000000	000000	AAPAT2: .WORD	400,0,0,0
5040	027266	000400	000000	000000	AAPAT3: .WORD	400,0,0,1
5041	027276	000400	000000	100000	AAPAT4: .WORD	400,0,100000,0
5042	027306	000200	000001	000000	AAPAT5: .WORD	200,1,0,0
5043	027316	000400	000001	000000	AAPAT6: .WORD	400,1,0,0
5044	027326				AADONE:	
	027326	104413			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?).



5055

```
.SBTTL TEST # 32 - ADDF & ADDD WITH E(AC) LESS THAN E(FSRC) TEST
*****
*TEST 32      ADDF & ADDD WITH E(AC) LESS THAN E(FSRC) TEST
*
*THIS IS A TEST OF THE ADDD AND ADDF
*INSTRUCTIONS AND THE ALIGN AC ALGORITHM
*FLOWS. THE CONSTANT (25 FOR FLOATING, 57 FOR
*DOUBLE) USED IS CHECKED. THEN SIMPLE
*AND WORST CASE ALIGNMENT SITUATIONS ARE
*TRIED. NOTE E(AC) IS LESS THEN E(FSRC)
*
*****
```

```
027330 000004
5056 027330
5057 027332
5058 027340
5059 027344
5060 027346
5061 027354
5062 027360
5063 027362
5064 027366
5065 027370
5066 027372
5067 027376
5068 027400
5069 027404
5070 027410
5071 027412
5072 027414
5073 027420
5074 027424
5075 027430
5076 027432
5077 027434
5078 027440
5079 027442
5080 027446
5081 027450
5082 027452
5083 027454
5084
5085 027460
5086 027466
5087 027472
5088 027474
5089 027502
5090 027506
5091 027510
5092 027514
5093 027516
5094 027520
5095 027524
5096 027526
5097 027532
```

```
TST32: SCOPE
;EXPONENT DIFFERENCE=57=71 (OCT) FD=1
CC1:
1$: MOV #1$, $LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
MOV #3200,R4 ;SET FIV,FIV, AND FD
LDFPS R4
MOV #CC2,$TMP2
MOV #CCP0,R0 ;SET ACO OPERAND
LDD (R0),ACO ;ACO
MOV #CCP2,R0
CC2: ADDD (R0),ACO ;TEST INSTRUCTION
STFPS R5 ;GET FPS
MOV #CCDAT0,R0 ;GET THE RESULT
STD ACO,(R0)
MOV #CCP2,R1 ;IS IT CORRECT
MOV #4,R2
CC3: CMP (R0)+,(R1)+
BEQ CC6
MOV #CCDAT0,R0 ;DID A BAD
MOV #CCP0,R1 ;CONSTANT (NOT 57)
MOV #4,R2 ;GET GENERATED
CC4: CMP (R0)+,(R1)+ ;FOR THE ALIGNMENT
BEQ CC5 ;FLOWS?
JMP CCER1 ;DATA ERROR.D
CC5: SOB R2,CC4
JMP CCER2 ;BAD CONSTANT.D
CC6: SOB R2,CC3
CMP R4,R5 ;FPS CORRECT?
BEQ CC7
JMP CCER0 ;BAD FPS.
;EXPONENT DIFFERENCE=56=70 (OCT) FD=1
CC7:
1$: MOV #1$, $LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
MOV #3200,R4 ;SET FIV,FIV, AND FD
LDFPS R4
MOV #CC8,$TMP2
MOV #CCP0,R0 ;SET ACO OPERAND
LDD (R0),ACO
MOV #CCP1,R0 ;FSRC
CC8: ADDD (R0),ACO ;TEST INSTRUCTION
STFPS R5 ;GET FPS
MOV #CCDAT0,R0 ;GET THE RESULT
STD ACO,(R0)
MOV #CCP7,R1 ;IS IT CORRECT
MOV #4,R2
```



5098	027536	022021			CC9:	CMP	(R0)+,(R1)+	
5099	027540	001415				BEQ	CC12	
5100	027542	012700	031004			MOV	#CCDATO,R0	:DID A BAD
5101	027546	012701	031024			MOV	#CCP1,R1	:CONSTANT (NOT 57)
5102	027552	012702	000004			MOV	#4,R2	:GET GENERATED
5103	027556	022021			CC10:	CMP	(R0)+,(R1)+	:FOR THE ALIGNMENT
5104	027560	001402				BEQ	CC11	:FLOWS?
5105	027562	000137	030476			JMP	CCER3	:DATA ERROR.D
5106	027566	077205			CC11:	SOB	R2,CC10	
5107	027570	000137	030514			JMP	CCER4	:BAD CONSTANT.D
5108	027574	077220			CC12:	SOB	R2,CC9	
5109	027576	020405				CMP	R4,R5	:FPS CORRECT?
5110	027600	001402				BEQ	CC13	
5111	027602	000137	030346			JMP	CCER0	:BAD FPS.
5112								:EXPONENT DIFFERENCE=25=31 (OCT) FD=0
5113	027606				CC13:			
	027606	012737	027614	001110		MOV	#1\$,\$LPERR	:SET UP THE LOOP ON ERROR ADDRESS.
5114	027614	012737	027642	001236	1\$:	MOV	#CC14,\$TMP2	
5115	027622	012700	031014			MOV	#CCP0,R0	:SET UP ACO OPERAND.
5116	027626	172410				LDD	(R0),ACO	
5117	027630	012704	003000			MOV	#3000,R4	:SET FIV,FIV. CLEAR FD.
5118	027634	170104				LDFPS	R4	
5119	027636	012700	031074			MOV	#CCP6,R0	:FSRC
5120	027642	172010			CC14:	ADDF	(R0),ACO	:TEST INSTRUCTION
5121	027644	170205				STFPS	R5	
5122	027646	170011				SETD		:REENTER DOUBLE MOVE
5123	027650	012700	031004			MOV	#CCDATO,R0	:GET THE RESULT
5124	027654	174010				STD	ACO,(R0)	
5125	027656	012701	031074			MOV	#CCP6,R1	:IS THE RESULT CORRECT?
5126	027662	012702	000002			MOV	#2,R2	
5127	027666	022021			CC15:	CMP	(R0)+,(R1)+	
5128	027670	001415				BEQ	CC18	
5129	027672	012700	031004			MOV	#CCDATO,R0	:WAS A BAD CONSTANT
5130	027676	012701	031044			MOV	#CCP3,R1	:USED (NOT 25) IN
5131	027702	012702	000002			MOV	#2,R2	:THE ALIGN FLOWS?
5132	027706	022021			CC16:	CMP	(R0)+,(R1)+	
5133	027710	001402				BEQ	CC17	
5134	027712	000137	030552			JMP	CCER5	:DATA ERROR F
5135	027716	077205			CC17:	SOB	R2,CC16	
5136	027720	000137	030606			JMP	CCER6	:BAD CONSTANT F
5137	027724	077220			CC18:	SOB	R2,CC15	
5138	027726	020405				CMP	R4,R5	
5139	027730	001402				BEQ	CC19	
5140	027732	000137	030364			JMP	CCER90	:BAD FPS.
5141								:EXPONENT DIFFERENC =24=30 (OCT) FD=0
5142	027736				CC19:			
	027736	012737	027744	001110		MOV	#1\$,\$LPERR	:SET UP THE LOOP ON ERROR ADDRESS.
5143	027744	012737	027772	001236	1\$:	MOV	#CC20,\$TMP2	
5144	027752	012700	031044			MOV	#CCP3,R0	:SET UP ACO OPERAND.
5145	027756	172410				LDD	(R0),ACO	
5146	027760	012704	003000			MOV	#3000,R4	:SET FIV,FIV. CLEAR FD.
5147	027764	170104				LDFPS	R4	
5148	027766	012700	031064			MOV	#CCP5,R0	:FSRC
5149	027772	172010			CC20:	ADDF	(R0),ACO	:TEST INSTRUCTION
5150	027774	170205				STFPS	R5	
5151	027776	170011				SETD		:REENTER DOUBLE MOVE
5152	030000	012700	031004			MOV	#CCDATO,R0	:GET THE RESULT



5153	030004	174010			STD	ACO,(R0)	
5154	030006	012701	031114		MOV	#CCP10,R1	;IS THE RESLT CORRECT?
5155	030012	012702	000002		MOV	#2,R2	
5156	030016	022021		CC21:	CMP	(R0)+,(R1)+	
5157	030020	001415			BEQ	CC24	
5158	030022	012700	031004		MOV	#CCDATO,R0	;WAS A BAD CONSTANT
5159	030026	012701	031064		MOV	#CCP5,R1	;USED (NOT 25) IN
5160	030032	012702	000002		MOV	#2,R2	;THE ALIGN FLOWS?
5161	030036	022021		CC22:	CMP	(R0)+,(R1)+	
5162	030040	001402			BEQ	CC23	
5163	030042	000137	030642		JMP	CCER7	;DATA ERROR F
5164	030046	077205		CC23:	SOB	R2,CC22	
5165	030050	000137	030660		JMP	CCER8	;BAD CONSTANT F
5166	030054	077220		CC24:	SOB	R2,CC21	
5167	030056	020405			CMP	R4,R5	
5168	030060	001402			BEQ	CC25	
5169	030062	000137	030364		JMP	CCER90	;BAD FPS.
5170							;EXPONENT DIFFERENCE=1 FD=1
5171	030066			CC25:			
	030066	012737	030074	001110	MOV	#1\$,\$LPERR	;SET UP THE LOOP ON ERROR ADDRESS.
5172	030074	012704	003200	1\$:	MOV	#3200,R4	;SET FIV,FIV, AND FD
5173	030100	170104			LDFPS	R4	
5174	030102	012737	030122	001236	MOV	#CC26,\$TMP2	
5175	030110	012700	031014		MOV	#CCP0,R0	;SET ACO OPERAND
5176	030114	172410			LDD	(R0),ACO	
5177	030116	012700	031044		MOV	#CCP3,R0	;FSRC
5178	030122	172010		CC26:	ADDD	(R0),ACO	;TEST INSTRUCTION
5179	030124	170205			STFPS	R5	;GET FPS
5180	030126	012700	031004		MOV	#CCDATO,R0	;GET THE RESULT
5181	030132	174010			STD	ACO,(R0)	
5182	030134	012701	031124		MOV	#CCP11,R1	;IS IT CORRECT
5183	030140	012702	000004		MOV	#4,R2	
5184	030144	022021		CC27:	CMP	(R0)+,(R1)+	
5185	030146	001415			BEQ	CC30	
5186	030150	012700	031004		MOV	#CCDATO,R0	;DID A BAD
5187	030154	012701	031044		MOV	#CCP3,R1	;CONSTANT (NOT 57)
5188	030160	012702	000004		MOV	#4,R2	;GET GENERATED
5189	030164	022021		CC28:	CMP	(R0)+,(R1)+	;FOR THE ALIGNMENT
5190	030166	001402			BEQ	CC29	;FLOWS?
5191	030170	000137	030714		JMP	CCER10	;DATA ERROR.D
5192	030174	077205		CC29:	SOB	R2,CC28	
5193	030176	000137	030732		JMP	CCER11	;BAD CONSTANT.D
5194	030202	077220		CC30:	SOB	R2,CC27	
5195	030204	020405			CMP	R4,R5	;FPS CORRECT?
5196	030206	001402			BEQ	CC31	
5197	030210	000137	030346		JMP	CCER0	;BAD FPS.
5198							;EXPONENT DIFFERENCE=100=144 (OCT) FD=1
5199	030214			CC31:			
	030214	012737	030222	001110	MOV	#1\$,\$LPERR	;SET UP THE LOOP ON ERROR ADDRESS.
5200	030222	012704	003200	1\$:	MOV	#3200,R4	;SET FIV,FIV, AND FD
5201	030226	170104			LDFPS	R4	
5202	030230	012737	030250	001236	MOV	#CC32,\$TMP2	
5203	030236	012700	031014		MOV	#CCP0,R0	;SET ACO OPERAND
5204	030242	172410			LDD	(R0),ACO	
5205	030244	012700	031054		MOV	#CCP4,R0	;FSRC
5206	030250	172010		CC32:	ADDD	(R0),ACO	;TEST INSTRUCTION
5207	030252	170205			STFPS	R5	;GET FPS



```

5208 030254 012700 031004      MOV      #CCDATO,R0      ;GET THE RESULT
5209 030260 174010      STD      ACO,(R0)
5210 030262 012701 031054      MOV      #CCP4,R1      ;IS IT CORRECT
5211 030266 012702 000004      MCV      #4,R2
5212 030272 022021      CC33:   CMP      (R0)+,(R1)+
5213 030274 001415      BEQ      CC36
5214 030276 012700 031004      MOV      #CCDATO,R0      ;DID A BAD
5215 030302 012701 031054      MOV      #CCP4,R1      ;CONSTANT (NOT 57)
5216 030306 012702 000004      MOV      #4,R2      ;GET GENERATED
5217 030312 022021      CC34:   CMP      (R0)+,(R1)+      ;FOR THE ALIGNMENT
5218 030314 001402      BEQ      CC35      ;FLOWS?
5219 030316 000137 030750      JMP      CCER12      ;DATA ERROR.D
5220 030322 077205      CC35:   SOB      R2,CC34
5221 030324 000137 030766      JMP      CCER13      ;BAD CONSTANT.D
5222 030330 077220      CC36:   SOB      R2,CC33
5223 030332 020405      CMP      R4,R5      ;FPS CORRECT?
5224 030334 001402      BEQ      CC37
5225 030336 000137 030346      JMP      CCER0      ;BAD FPS.
5226 030342 000137 031144      CC37:   JMP      CCDONE
5227 030346 010437 001242      CCER0:  MOV      R4,$TMP4      ;FPS ERROR D
5228 030352 010537 001240      MOV      R5,$TMP3
5229 030356 104164      1$:    ERROR      +164
5230 030360 000137 031144      JMP      CCDONE
5231 030364 010437 001242      CCER90: MOV      R4,$TMP4      ;FPS ERROR F
5232 030370 010537 001240      MCV      R5,$TMP3
5233 030374 104165      1$:    ERROR      +165
5234 030376 000137 031144      JMP      CCDONE
5235 030402 012737 031034 001240      CCER1:  MOV      #CCP2,$TMP3      ;DATA ERROR D
5236 030410 012737 031034 001246      MOV      #CCP2,$TMP6
5237 030416 012737 031014 001242      CCER50: MOV      #CCP0,$TMP4
5238 030424 012737 031004 001244      MOV      #CCDATO,$TMP5
5239 030432 104166      1$:    ERROR      +166
5240 030434 000137 031144      JMP      CCDONE
5241 030440 012737 031034 001240      CCER2:  MOV      #CCP2,$TMP3      ;CONSTANT BAD D(B)
5242 030446 012737 031034 001246      MOV      #CCP2,$TMP6
5243 030454 012737 031014 001242      CCER22: MOV      #CCP0,$TMP4
5244 030462 012737 031004 001244      MOV      #CCDATO,$TMP5
5245 030470 104172      1$:    ERROR      +172
5246 030472 000137 031144      JMP      CCDONE
5247 030476 012737 031024 001240      CCER3:  MOV      #CCP1,$TMP3
5248 030504 012737 031104 001246      MOV      #CCP7,$TMP6
5249 030512 000741      BR      CCER50
5250 030514 012737 031024 001240      CCER4:  MOV      #CCP1,$TMP3      ;CONSTANT BAD D(G)
5251 030522 012737 031104 001246      MOV      #CCP7,$TMP6
5252 030530 012737 031014 001242      CCER44: MOV      #CCP0,$TMP4
5253 030536 012737 031004 001244      MOV      #CCDATO,$TMP5
5254 030544 104173      1$:    ERROR      +173
5255 030546 000137 031144      JMP      CCDONE
5256 030552 012737 031074 001240      CCER5:  MOV      #CCP6,$TMP3      ;DATA ERROR F
5257 030560 012737 031074 001246      MOV      #CCP6,$TMP6
5258 030566 012737 031014 001242      CCER55: MOV      #CCP0,$TMP4
5259 030574 012737 031004 001244      MOV      #CCDATO,$TMP5
5260 030602 104170      1$:    ERROR      +170
5261 030604 000557      BR      CCDONE
5262 030606 012737 031074 001240      CCER6:  MOV      #CCP6,$TMP3      ;CONSTANT BAD F(B)
5263 030614 012737 031074 001246      MOV      #CCP6,$TMP6
5264 030622 012737 031014 001242      MOV      #CCP0,$TMP4

```



```

5265 030630 012737 031004 001244      MOV      #CCDATO,$TMP5
5266 030636 104174      1$:      ERROR      +174
5267 030640 000541      BR      CCDONE
5268 030642 012737 031064 001240  CCER7:  MOV      #CCP5,$TMP3      ;DATA ERROR F
5269 030650 012737 031114 001246      MOV      #CCP10,$TMP6
5270 030656 000743      BR      CCER55
5271 030660 012737 031064 001240  CCER8:  MOV      #CCP5,$TMP3      ;CONSTANT BAD F(G)
5272 030666 012737 031114 001246      MOV      #CCP10,$TMP6
5273 030674 012737 031004 001244      MOV      #CCDATO,$TMP5
5274 030702 012737 031014 001242      MOV      #CCP0,$TMP4
5275 030710 104175      1$:      ERROR      +175
5276 030712 000514      BR      CCDONE
5277 030714 012737 031044 001240  CCER10: MOV      #CCP3,$TMP3      ;DATA ERROR D
5278 030722 012737 031124 001246      MOV      #CCP11,$TMP6
5279 030730 000632      BR      CCER50
5280 030732 012737 031044 001240  CCER11: MOV      #CCP3,$TMP3      ;CONSTANT BAD D(G)
5281 030740 012737 031124 001246      MOV      #CCP11,$TMP6
5282 030746 000670      BR      CCER44
5283 030750 012737 031054 001240  CCER12: MOV      #CCP4,$TMP3      ;DATA ERROR D
5284 030756 012737 031054 001246      MOV      #CCP4,$TMP6
5285 030764 000614      ER      CCER50
5286 030766 012737 031054 001240  CCER13: MOV      #CCP4,$TMP3      ;CONSTANT BAD D(B)
5287 030774 012737 031054 001246      MOV      #CCP4,$TMP6
5288 031002 000624      BR      CCER22
5289 031004 000000 000000 000000  CCDATO: .WORD    0,0,0,0
5290 031014 000200 000000 000000  CCP0:   .WORD    200,0,0,0      ;E(AC)=1
5291 031024 016200 000000 000000  CCP1:   .WORD    16200,0,0,0     ;E(FSRC)=E(AC)+56=57
5292 031034 016400 000000 000000  CCP2:   .WORD    16400,0,0,0     ;E(FSRC)=E(AC)+57=58
5293 031044 000400 000000 000000  CCP3:   .WORD    400,0,0,0       ;E(FSRC)=E(AC)+1=2
5294 031054 031200 000000 000000  CCP4:   .WORD    31200,0,0,0     ;E(FSRC)=E(AC)+100=101=145(OCT)
5295 031064 006200 000000 000000  CCP5:   .WORD    6200,0,0,0      ;E(FSRC)=E(AC)+24=25=31(OCT)
5296 031074 006400 000000 000000  CCP6:   .WORD    6400,0,0,0      ;E(FSRC)=E(AC)+25=26=32(OCT)
5297 031104 016200 000000 000000  CCP7:   .WORD    16200,0,0,1
5298 031114 006200 000001 000000  CCP10:  .WORD    6200,1,0,0
5299 031124 000500 000000 000000  CCP11:  .WORD    500,0,0,0
5300 031134 000200 000000 000000  CCP12:  .WORD    200,0,0,0
5301 031144 104413      CCDONE:
                                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?).
    
```



5313

.SBTTL TEST # 33 - ADDF & ADDD WITH E(AC) GREATER THAN E(FSRC) TEST

\*\*\*\*\*  
:TEST 33 ADDF & ADDD WITH E(AC) GREATER THAN E(FSRC) TEST

\*\*\*\*\*  
:THIS IS A TEST OF THE ADDD AND ADDF  
:INSTRUCTIONS AND THE ALIGN FSRC ALGORITHM  
:FLOWS. FIRST THE CONSTANT USED IS CHECKED.  
:THEN SIMPLE AND WORST CASE ALIGNMENT  
:SITUATIONS ARE TRIED. NOTE E(AC)  
:IS GREATER THAN E(FSRC).  
\*\*\*\*\*

TST33: SCOPE  
:EXPONENT DIFFERENCE=57=71 (OCT) FD=1

5314 031146 000004  
5315 031150  
5316 031156 012737 031156 001110  
5317 031162 170104  
5318 031164 012737 032036 000244  
5319 031172 012737 031212 001236  
5320  
5321 031200 012700 032400  
5322 031204 172410  
5323 031206 012700 032370  
5324 031212 172010  
5325 031214 170205  
5326 031216 012700 032350  
5327 031222 174010  
5328 031224 012701 032400  
5329 031230 012702 000004  
5330 031234 022021  
5331 031236 001402  
5332 031240 000137 032076  
5333 031244 077205  
5334  
5335 031246 020405  
5336 031250 001402  
5337 031252 000137 032036  
5338  
5339 031256  
5340 031264 012737 031264 001110  
5341 031270 170104  
5342 031272 012737 031312 001236  
5343 031300 012700 032420  
5344 031304 172410  
5345 031306 012700 032370  
5346 031312 172010  
5347 031314 170205  
5348 031316 012700 032350  
5349 031322 174010  
5350 031324 012701 032460  
5351 031330 012702 000004  
5352 031334 022021  
5353 031336 001415  
5354 031340 012700 032350  
5355 031344 012701 032420

BB1: MOV #1\$, \$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.  
:MOV #3200, R4 ;SET FIV FIV, AND FD  
LDFPS R4  
MOV #BBERO, FPVECT ;SET UP FOR ERROR  
MOV #BB2, \$TMP2 ;IN CASE THE OVER  
;UNDER FLOWS FAIL.  
;SET ACO OPERAND.  
MOV #BBPAT2, R0  
LDD (R0), ACO  
MOV #BBPAT1, R0 ;FSRC  
BB2: ADDD (R0), ACO ;TEST INSTRUCTION  
STFPS R5  
BB3: MOV #BBDATO, R0 ;GET THE RESULT  
STD ACO, (R0)  
MOV #BBPAT2, R1 ;RESULT CORRECT?  
MOV #4, R2  
BB4: CMP (R0)+, (R1)+  
BEQ BB5  
JMF BBER1 ;DATA ERROR D  
BB5: SOB R2, BB4 ;WAS FPS CORRECT?  
CMP R4, R5  
BEQ BB6  
JMP BBERO ;FPS ERROR

:EXPONENT DIFFERENCE=56=70 (OCT) FD=1

BB6: MOV #1\$, \$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.  
:MOV #3200, R4 ;SET FIV, FIV, AND FD  
LDFPS R4  
MOV #BB7, \$TMP2  
MOV #BBPAT4, R0 ;SET ACO OPERAND  
LDD (R0), ACO  
MOV #BBPAT1, R0 ;FSRC  
BB7: ADDD (R0), ACO ;TEST INSTRUCTION  
STFPS R5 ;GET FPS  
MOV #BBDATO, R0 ;GET THE RESULT  
STD ACO, (R0)  
MOV #BBP10, R1 ;IS IT CORRECT  
MOV #4, R2  
BB10: CMP (R0)+, (R1)+  
BEQ BB13  
MOV #BBDATO, R0 ;DID A BAD  
MOV #BBPAT4, R1 ;CONSTANT (NOT 57)



```

5356 031350 012702 000004          MOV      #4,R2          ;GET GENERATED
5357 031354 022021          BB11:  CMP      (R0)+,(R1)+ ;FOR THE ALIGNMENT
5358 031356 001402          BEQ      B22          ;FLOWS?
5359 031360 000137 032134          JMP      BBER2        ;DATA ERROR.D
5360 031364 077205          BB12:  SOB      R2,BB11
5361 031366 000137 032152          JMP      BBER3        ;BAD CONSTANT.D
5362 031372 077220          BB13:  SOB      R2,BB10
5363 031374 020405          CMP      R4,R5        ;FPS CORRECT?
5364 031376 001402          BEQ      BB14
5365 031400 000137 032036          JMP      BBER0        ;BAD FPS.
5366
5367 031404          ;EXPONENT DIFFERENCE=25=31 (OCT) FD=0
031404 012737 031412 001110          BB14:  MOV      #1$, $LPERR    ;SET UP THE LOOP ON ERROR ADDRESS.
031412 012737 031440 001236          1$:   MOV      #BB15,$TMP2
031420 012700 032360          MOV      #BBPAT0,R0    ;SET UP ACO OPERAND
031424 172410          LDD      (R0),AC0
031426 012704 003000          MOV      #3000,R4      ;SET FIV AND FIV
5372                                     ;CLEAR FD
031432 170104          LDFPS   R4
031434 012700 032370          BB15:  MOV      #BBPAT1,R0    ;FSRC
031440 172010          ADDF    (R0),AC0      ;TEST INSTRUCTION
031442 170205          STFPS   R5
031444 170011          SETD
031446 012700 032350          MOV      #BBDAT0,R0    ;REENTERED DOUBLE MODE.
031452 174010          STD     ACO,(R0)      ;GET THE RESULT
031454 012701 032360          MOV      #BBPAT0,R1
031460 012702 000002          MOV      #2,R2        ;IS THE RESULT
031464 022021          BB16:  CMP      (R0)+,(R1)+  ;CORRECT?
031466 001402          BEQ      BB17
031470 000137 032206          JMP      BBER4        ;DATA ERROR F
031474 077205          BB17:  SOB      R2,BB16
031476 020405          CMP      R4,R5        ;IS FPS CORRECT?
031500 001402          BEQ      BB20
031502 000137 032056          JMP      BBER10       ;FPS ERROR.
5389
5390 031506          ;EXPONENT DIFFERENCE=24=30 (OCT)
031506 012737 031514 001110          BB20:  MOV      #1$, $LPERR    ;SET UP THE LOOP ON ERROR ADDRESS.
031514 012737 031542 001236          1$:   MOV      #BE21,$TMP2
031522 012700 032410          MOV      #BBPAT3,R0    ;SET UP ACO OPERAND.
031526 172410          LDD      (R0),AC0
031530 012704 003000          MOV      #3000,R4      ;SET FIU,FIV. CLEAR FD.
031534 170104          LDFPS   R4
031536 012700 032370          BB21:  MOV      #BBPAT1,R0    ;FSRC
031542 172010          ADDF    (R0),AC0      ;TEST INSTRUCTION
031544 170205          STFPS   R5
031546 170011          SETD
031550 012700 032350          MOV      #BBDAT0,R0    ;REENTER DOUBLE MODE
031554 174010          STD     ACO,(R0)      ;GET THE RESULT
031556 012701 032450          MOV      #BBPAT7,R1
031562 012702 000002          MOV      #2,R2        ;IS THE RESULT CORRECT?
031566 022021          BB22:  CMP      (R0)+,(R1)+
031570 001415          BEQ      BB25
031572 012700 032350          MOV      #BBDAT0,R0    ;WAS A BAD CONSTANT
031576 012701 032410          MOV      #BBPAT3,R1    ;USED (NOT 25) IN
031602 012702 000002          MOV      #2,R2        ;THE ALLIGN FLOWS?
031606 022021          BB23:  CMP      (R0)+,(R1)+
031610 001402          BEQ      BB24

```



```

5411 031612 000137 032242          JMP      BBER5          ;DATA ERROR F
5412 031616 077205          SOB      R2,BB23       ;BAD CONSTANT F
5413 031620 000137 032260          JMP      BBER6
5414 031624 077220          SOB      R2,BB22
5415 031626 020405          CMP      R4,R5
5416 031630 001402          BEQ      BB26
5417 031632 000137 032056          JMP      BBER10        ;BAD FPS.
5418
5419 031636          ;EXPONENT DIFFERENCE=1
031636 012737 031644 001110          BB26:  MOV      #15,$LPERR    ;SET UP THE LOOP ON ERROR ADDRESS.
031644 012737 031672 001236          1$:   MOV      #BB27,$TMP2
031652 012704 003200          MOV      #3200,R4
031656 170104          LDFPS   R4              ;SET UP ACO OPERAND
031660 012700 032430          MOV      #BBPAT5,R0
031664 172410          LDD     (R0),ACO
031666 012700 032370          MOV      #BBPAT1,R0    ;FSRC
031672 172010          BB27:  ADDD   (R0),ACO    ;TEST INSTRUCTION
031674 170205          STFPS  R5
031676 012700 032350          MOV      #BBDAT0,R0    ;GET THE RESULT.
031702 174010          STD     ACO,(R0)
031704 012701 032470          MOV      #BBP11,R1    ;IS IT CORRECT?
031710 012702 000004          MOV      #4,R2
031714 022021          BB30:  CMP     (R0)+,(R1)+
031716 001402          BEQ     BB31
031720 000137 032314          JMP      BBER7          ;DATA ERROR D
031724 077205          BB31:  SOB     R2,BB30
031726 020405          CMP     R4,R5          ;IS FPS CORRECT
031730 001402          BEQ     BB32
031732 000137 032036          JMP      BBER0
5439
5440 031736          ;EXPONENT DIFFERENCE=100=144 (OCT)
031736 012737 031744 001110          BB32:  MOV      #15,$LPERR    ;SET UP THE LOOP ON ERROR ADDRESS.
031744 012737 031772 001236          1$:   MOV      #BB33,$TMP2
031752 012704 003200          MOV      #3200,R4
031756 170104          LDFPS   R4              ;SET FIV,FIV AND FD
031760 012700 032440          MOV      #BBPAT6,R0    ;SET UP ACO OPERAND.
031764 172410          LDD     (R0),ACO
031766 012700 032370          MOV      #BBPAT1,R0    ;FSRC
031772 172010          BB33:  ADDD   (R0),ACO    ;TEST INSTRUCTION
031774 170205          STFPS  R5
031776 012700 032350          MOV      #BBDAT0,R0    ;GET THE RESULT
032002 174010          STD     ACO,(R0)
032004 012701 032440          MOV      #BBPAT6,R1    ;IS IT CORRECT
032010 012702 000004          MOV      #4,R2
032014 022021          BB34:  CMP     (R0)+,(R1)+
032016 001402          BEQ     BB35
032020 000137 032332          JMP      BBER8          ;DATA ERROR D
032024 077205          BB35:  SOB     R2,BB34
032026 020405          CMP     R4,R5          ;IS FPS CORRECT
032030 001002          BNE     BBER0
032032 000137 032500          JMP      BBDONE
032036 010437 001242          BBER0: MOV      R4,$TMP4    ;FPS ERROR D
032042 010537 001240          MOV      R5,$TMP3
5462 032046 104164          1$:   ERROR  +164
5463 032050 104413          RSETUP
;GO INITIALIZE THE FPS AND STACK; AND
;SEE IF THE USER HAS EXPRESSED
;THE DESIRE TO CHANGE THE SOFTWARE

```



5464 032052 000137 032500  
 5465 032056 010437 001242  
 5466 032062 010537 001240  
 5467 032066 104165  
 5468 032070 104413

BBER10: JMP BBDONE  
 MOV R4,\$TMP4  
 MOV R5,\$TMP3  
 1\$: ERROR +165  
 RSETUP

:VIRTUAL CONSOLE SWITCH REGISTER (HAS  
 :THE USER TYPED CONTROL G?).

:FPS ERROR F

: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?).

5469 032072 000137 032500  
 5470 032076 012737 032400 001242  
 5471 032104 012737 032400 001246  
 5472 032112 012737 032370 001240  
 5473 032120 012737 032350 001244  
 5474 032126 104166  
 5475 032130 000137 032500  
 5476 032134 012737 032420 001242  
 5477 032142 012737 032460 001246  
 5478 032150 000760  
 5479 032152 012737 032420 001242  
 5480 032160 012737 032460 001246  
 5481 032166 012737 032370 001240  
 5482 032174 012737 032350 001244  
 5483 032202 104167  
 5484 032204 000535  
 5485 032206 012737 032360 001242  
 5486 032214 012737 032360 001246  
 5487 032222 012737 032370 001240  
 5488 032230 012737 032350 001244  
 5489 032236 104170  
 5490 032240 000517  
 5491 032242 012737 032410 001242  
 5492 032250 012737 032450 001246  
 5493 032256 000761  
 5494 032260 012737 032410 001242  
 5495 032266 012737 032450 001246  
 5496 032274 012737 032370 001240  
 5497 032302 012737 032350 001244  
 5498 032310 104171  
 5499 032312 000472  
 5500 032314 012737 032430 001242  
 5501 032322 012737 032370 001246  
 5502 032330 000670  
 5503 032332 012737 032440 001242  
 5504 032340 012737 032440 001246  
 5505 032346 000661  
 5506 032350 000000 000000 000000  
 5507 032360 006400 000000 000000  
 5508 032370 000200 000000 000000  
 5509 032400 016400 000000 000000  
 5510 032410 006200 000000 000000  
 5511 032420 016200 000000 000000  
 5512 032430 000400 000000 000000  
 5513 032440 031200 000000 000000  
 5514 032450 006200 000001 000000

BBER1: JMP BBDONE  
 MOV #BBPAT2,\$TMP4  
 BBER11: MOV #BBPAT2,\$TMP6  
 MOV #BBPAT1,\$TMP3  
 MOV #BBDAT0,\$TMP5  
 1\$: ERROR +166  
 JMP BBDONE  
 BBER2: MOV #BBPAT4,\$TMP4  
 MOV #BBP10,\$TMP6  
 BR BBER11  
 BBER3: MOV #BBPAT4,\$TMP4  
 MOV #BBP10,\$TMP6  
 MOV #BBPAT1,\$TMP3  
 MOV #BBDAT0,\$TMP5  
 1\$: ERROR +167  
 BR BBDONE  
 BBER4: MOV #BBPAT0,\$TMP4  
 MOV #BBPAT0,\$TMP6  
 BBER40: MOV #BBPAT1,\$TMP3  
 MOV #BBDAT0,\$TMP5  
 1\$: ERROR +170  
 BR BBDONE  
 BBER5: MOV #BBPAT3,\$TMP4  
 MOV #BBP7,\$TMP6  
 BR BBER40  
 BBER6: MOV #BBPAT3,\$TMP4  
 MOV #BBP7,\$TMP6  
 MOV #BBPAT1,\$TMP3  
 MOV #BBDAT0,\$TMP5  
 1\$: ERROR +171  
 BR BBDONE  
 BBER7: MOV #BBPAT5,\$TMP4  
 MOV #BBPAT11,\$TMP6  
 BR BBER11  
 BBER8: MOV #BBPAT6,\$TMP4  
 MOV #BBPAT6,\$TMP6  
 BR BBER11  
 BBDAT0: .WORD 0,0,0,0  
 BBPAT0: .WORD 6400,0,0,0  
 BBPAT1: .WORD 200,0,0,0  
 BBPAT2: .WORD 16400,0,0,0  
 BBPAT3: .WORD 6200,0,0,0  
 BBPAT4: .WORD 16200,0,0,0  
 BBPAT5: .WORD 400,0,0,0  
 BBPAT6: .WORD 31200,0,0,0  
 BBP7: .WORD 6200,1,0,0

:DATA ERROR D

:BAD CONSTANT D

:DATA ERROR F

:CONSTANT ERROR F

:F(AC)=E(FSRC)+25=26=32(OCT)  
 :E(FSRC)=1  
 :E(AC)=E(FSRC)+57=58=72(OCT)  
 :E(AC)=E(FSRC)+24=25=31(OCT)  
 :E(AC)=E(FSRC)+56=57=71(OCT)  
 :E(AC)=E(FSRC)+1=2  
 :E(AC)=E(FSRC)+100=101=145(OCT)  
 :BBPAT3 RES



5515	032460	016200	000000	000000	BBP10:	.WORD	16200,0,0,1	:BBPAT4 RES
5516	032470	000500	000000	000000	BBP11:	.WORD	500,0,0,0	:BBPAT5 RES
5517	032500				BBDONE:			
	032500	104413				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?).



5525

```

.SBTTL TEST # 34 - ADD WITH NEGATIVE OPRANDS TEST
.....
*TEST 34      ADD WITH NEGATIVE OPRANDS TEST
*
*THIS IS A TEST OF THE ADD INSTRUCTION
*WITH NEGATIVE OPERANDS. EVERY COMBINATION OF
*OPERAND SIGNS IS TRIED.
.....
    
```

```

5526 032502 000004
5527 032504
5528 032512 012737 032512 001110
5529 032516 170104
5530 032520 012737 032540 001236
5531 032526 012700 034420
5532 032532 172410
5533 032534 012700 034420
5534 032540 172010
5535 032542 170205
5536 032544 012700 034400
5537 032550 174010
5538 032552 012701 034520
5539 032556 012702 000004
5540 032562 022021
5541 032564 001415
5542 032566 012700 034400
5543 032572 012701 034450
5544 032576 012702 000004
5545 032602 022021
5546 032604 001402
5547 032606 000137 033630
5548 032612 077205
5549 032614 000137 033666
5550 032620 077220
5551 032622 052704 000010
5552 032626 020405
5553 032630 001402
5554 032632 000137 033612
5555
5556 032636
5557 032644 012704 003200 001110
5558 032650 170104
5559 032652 012737 032672 001236
5560 032660 012700 034430
5561 032664 172410
5562 032666 012700 034420
5563 032672 172010
5564 032674 170205
5565 032676 012700 034400
5566 032702 174010
5567 032704 012701 034410
5568 032710 012702 000004
5569 032714 022021
5570 032716 001402
    
```

```

TST34: SCOPE
:BOTH OPERANDS NEGATIVE
DD1:
18:  MOV #18,$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
     MOV #3200,R4 ;SET F10, F1V, AND FD
     LDFPS R4
     MOV #DD2,$TMP2
     MOV #DDP1,R0 ;SET ACO OPERAND
     LDD (R0),ACO
     MOV #DDP1,R0 ;ESRC
     ADDD (R0),ACO ;TEST INSTRUCTION
     STFPS R5 ;GET FPS
     MOV #DDDATO,R0 ;GET THE RESULT
     STD ACO,(R0)
     MOV #DDP9,R1 ;IS IT CORRECT
     MOV #4,R2
DD3:  CMP (R0)+,(R1)+
     BEQ DD6
     MOV #DDDATO,R0 ;DID A ADD-SUB
     MOV #DDP4,R1 ;FLOW A FAILURE
     MOV #4,R2
DD4:  CMP (R0)+,(R1)+
     BEQ DD5 ;216,442,500
     JMP DDER1 ;DATA ERROR,D
DD5:  SOB R2,DD4
     JMP DDER2 ;FLOW FAILURE,D
DD6:  SOB R2,DD3
     BIS #10,R4
     CMP R4,R5 ;FPS CORRECT?
     BEQ DD7
     JMP DDER0 ;BAD,FPS
:AC POS FSRC NEG AC=-FSRC
DD7:
18:  MOV #18,$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
     MOV #3200,R4 ;SET F10, F1V, AND FD
     LDFPS R4
     MOV #DD8,$TMP2
     MOV #DDP2,R0 ;SET ACO OPERAND
     LDD (R0),ACO
     MOV #DDP1,R0 ;FSPC
     ADDD (R0),ACO ;TEST INSTRUCTION
     STFPS R5 ;GET FPS
     MOV #DDDATO,R0 ;GET THE RESULT
     STD ACO,(R0)
     MOV #DDP0,R1 ;IS IT CORRECT
     MOV #4,R2
DD10:  CMP (R0)+,(R1)+
     BEQ DD11
    
```



```

5571 032720 000137 033724          JMP      DDER3          :FLOW FAILURE
5572 032724 077205          DD11:  SOB      R2,DD10
5573 032726 052704 000004          BIS      #4,R4
5574 032732 020405          CMP      R4,R5          :FPS CORRECT?
5575 032734 001402          BEQ      DD12
5576 032736 000137 033612          JMP      DDER0          :BAD FPS
5577                                     :AC NEG FSRC POS          AC=-FSRC
5578 032742          DD12:  MOV      #1$, $LPERR          :SET UP THE LOOP ON ERROR ADDRESS.
5579 032742 012737 032750 001110          MOV      #3200,R4          :SET FIU, FIV, AND FD
5580 032754 170104          LDFPS   R4
5581 032756 012737 032776 001236          MOV      #DD13,$TMP2
5582 032764 012700 034420          MOV      #DDP1,R0          :SET ACO OPERAND
5583 032770 172410          LDD     (R0),ACO
5584 032772 012700 034430          MOV      #DDP2,R0          :FSRC
5585 032776 172010          DD13:  ADDD   (R0),ACO          :TEST INSTRUCTION
5586 033000 170205          STFPS   R5          :GET FPS
5587 033002 012700 034400          MOV      #DDDATO,R0          :GET THE RESULT
5588 033006 174010          STD     ACO,(R0)
5589 033010 012701 034410          MOV      #DDP0,R1          :IS IT CORRECT
5590 033014 012702 000004          MOV      #4,R2
5591 033020 022021          DD14:  CMP     (R0)+,(R1)+
5592 033022 001402          BEQ     DD15
5593 033024 000137 033762          JMP     DDER4          :FLOW FAILURE 216,440,121
5594 033030 077205          DD15:  SOB     R2,DD14
5595 033032 052704 000004          BIS     #4,R4
5596 033036 020405          CMP     R4,R5          :EPS CORRECT?
5597 033040 001402          BEQ     DD16
5598 033042 000137 033612          JMP     DDER0          :BAD FPS
5599                                     :ACO POC          /AC/ > /FSRC/
5600 033046          DD16:  MOV     #1$, $LPERR          :SET UP THE LOOP ON ERROR ADDRESS.
5601 033046 012737 033054 001110          MOV     #3200,R4          :SET FIV, FIV AND FD
5602 033060 170104          LDFPS   R4
5603 033062 012737 033102 001236          MOV     #DD17,$TMP2
5604 033070 012700 034440          MOV     #DDP3,R0          :SET ACO OPERAND
5605 033074 172410          LDD     (R0),ACO
5606 033076 012700 034470          MOV     #DDP6,R0          :ESPC
5607 033102 172010          DD17:  ADDD   (R0),ACO          :TEST INSTRUCTION
5608 033104 170205          STFPS   R5          :GET FPS
5609 033106 012700 034400          MOV     #DDDATO,R0          :GET THE RESULT
5610 033112 174010          STD     ACO,(R0)
5611 033114 012701 034500          MOV     #DDP7,R1          :IS IT CORRECT
5612 033120 012702 000004          MOV     #4,R2
5613 033124 022021          DD18:  CMP     (R0)+,(R1)+
5614 033126 001415          BEQ     DD21
5615 033130 012700 034400          MOV     #DDDATO,R0          :FLOWS FAILURE
5616 033134 012701 034510          MOV     #DDP8,R1          :216,440,101
5617 033140 012702 000004          MOV     #4,R2          :GET GENERATED
5618 033144 022021          DD19:  CMP     (R0)+,(R1)+
5619 033146 001402          BEQ     DD20
5620 033150 000137 034020          JMP     DDER5          :DATA ERROR.
5621 033154 077205          DD20:  SOB     R2,DD19
5622 033156 000137 034056          JMP     DDER6
5623 033162 077220          DD21:  SOB     R2,DD18
5624 033164 020405          CMP     R4,R5          :EPS CORRECT?
5625 033166 001402          BEQ     DD22
  
```



5626	033170	000137	033612		JMP	DDERO		:BAD FPS
5627					:AC NEG	FSRC	POS /FSRC/ > /AC/	
5628	033174				DD22:			
	033174	012737	033202	001110	MOV	#1\$, \$LPERR		:SET UP THE LOOP ON ERROR ADDRESS.
5629	033202	012704	003200		1\$:	MOV	#3200, R4	:SET FIO, FIV, AND FD
5630	033206	170104			LDFPS	R4		
5631	033210	012737	033230	001236	MOV	#DD23, \$TMP2		
5632	033216	012700	034470		MOV	#DDP6, R0		:SET ACO OPERAND
5633	033222	172410			LDD	(R0), ACO		
5634	033224	012700	034440		MOV	#DDP3, R0		:FSPC
5635	033230	172010			DD23:	ADDD	(R0), ACO	:TEST INSTRUCTION
5636	033232	170205			STFPS	R5		:GET FPS
5637	033234	012700	034400		MOV	#DDDATO, R0		:GET THE RESULT
5638	033240	174010			STD	ACO, (R0)		
5639	033242	012701	034500		MOV	#DDP7, R1		:IS IT CORRECT?
5640	033246	012702	000004		MOV	#4, R2		
5641	033252	022021			DD24:	CMP	(R0)+, (R1)+	
5642	033254	001415			BEQ	DD27		
5643	033256	012700	034400		MOV	#DDDATO, R0		:FLO,S FAILURE
5644	033262	012701	034510		MOV	#DDP9, R1		:CONSTANT (NOT 57)
5645	033266	012702	000004		MOV	#4, R2		:216,042,101
5646	033272	021011			DD25:	CMP	(R0), (R1)	
5647	033274	001402			BEQ	DD26		
5648	033276	000137	034114		JMP	DDER7		:DATA ERROR.
5649	033302	077205			DD26:	SOB	R2, DD25	
5650	033304	000137	034152		JMP	DDER8		
5651	033310	077220			DD27:	SOB	R2, DD24	
5652	033312	020405			CMP	R4, R5		:FPS CORRECT?
5653	033314	001402			BEQ	DD30		
5654	033316	000137	033612		JMP	DDERO		:BAD FPS
5655					:ACO POS	FSRC	NEG /AC/ < /FSRC/	
5656	033322				DD30:			
	033322	012737	033330	001110	MOV	#1\$, \$LPERR		:SET UP THE LOOP ON ERROR ADDRESS.
5657	033330	012704	003200		1\$:	MOV	#3200, R4	:SET FIO, FIV, AND FD
5658	033334	170104			LDFPS	R4		
5659	033336	012737	033356	001236	MOV	#DD31, \$TMP2		
5660	033344	012700	034450		MOV	#DDP4, R0		:SET ACO OPERAND
5661	033350	172410			LDD	(R0), ACO		
5662	033352	012700	034460		MOV	#DDP5, R0		:FSPC
5663	033356	172010			DD31:	ADDD	(R0), ACO	:TEST INSTRUCTION
5664	033360	170205			STFPS	R5		:GET FPS
5665	033362	012700	034400		MOV	#DDDATO, R0		:GET THE RESULT
5666	033366	174010			STD	ACO, (R0)		
5667	033370	012701	034510		MOV	#DDP8, R1		:IS IT CORRECT
5668	033374	012702	000004		MOV	#4, R2		
5669	033400	022021			DD32:	CMP	(R0)+, (R1)+	
5670	033402	001415			BEQ	DD35		:ADD-SUB
5671	033404	012700	034400		MOV	#DDDATO, R0		:FLOWAS FAILURE
5672	033410	012701	034500		MOV	#DDP7, R1		:CON 216 N440 NOT 141
5673	033414	012702	000004		MOV	#4, R2		:GET GENERATED
5674	033420	022021			DD33:	CMP	(R0)+, (R1)+	:FOR THE ALLIGNMENT
5675	033422	001402			BEQ	DD34		:FLOWS?
5676	033424	000137	034210		JMP	DDER9		:DATA ERROR, D
5677	033430	077205			DD34:	SOB	R2, DD33	
5678	033432	000137	034246		JMP	DDER10		
5679	033436	077220			DD35:	SOB	R2, DD32	
5680	033440	052704	000010		BIS	#10, R4		



5681	033444	020405				CMP	R4,R5		:FPS CORRECT?
5682	033446	001402				BEQ	DD36		
5683	033450	000137	033612			JMP	DDERO		:BAD FPS
5684							FSRC	POS	/FSRC/</AC/
5685	033454								
	033454	012737	033462	001110		DD36:	MOV	#1\$, \$LPERR	:SET UP THE LOOP ON ERROR ADDRESS.
5686	033462	012704	003200			1\$:	MOV	#3200,R4	:SET FIO, FIV, AND FD
5687	033466	170104					LDFPS	R4	
5688	033470	012737	033510	001236			MOV	#DD37,\$TMP2	
5689	033476	012700	034460				MOV	#DDP5,R0	:SET ACO OPERAND
5690	033502	172410					LDD	(R0),ACO	
5691	033504	012700	034450				MOV	#DDP4,R0	:FSPC
5692	033510	172010				DD37:	ADDD	(R0),ACO	:TEST INSTRUCTION
5693	033512	170205					STFPS	R5	:GET FPS
5694	033514	012700	034400				MOV	#DDDATO,R0	:GET THE RESULT
5695	033520	174010					STD	ACO,(R0)	
5696	033522	012701	034510				MOV	#DDP8,R1	:IS IT CORRECT
5697	033526	012702	000004				MOV	#4,R2	
5698	033532	022021				DD38:	CMP	(R0)+,(R1)+	
5699	033534	001415					BEQ	DD41	
5700	033536	012700	034400				MOV	#DDDATO,R0	:ADD SUB
5701	033542	012701	034500				MOV	#DDP7,R1	:FLOWS FAILURES
5702	033546	012702	000004				MOV	#4,R2	:GET 216,042,141
5703	033552	022021				DD39:	CMP	(R0)+,(R1)+	:FOR THE ALLIGNMENT
5704	033554	001402					BEQ	DD40	:FLOWS?
5705	033556	000137	034304				JMP	DDER11	:DATA ERROR. D
5706	033562	077205				DD40:	SOB	R2,DD39	
5707	033564	000137	034342				JMP	DDER12	:BAD CONSTANT.D
5708	033570	077220				DD41:	SOB	R2,DD38	
5709	033572	052704	000010				BIS	#10,R4	
5710	033576	020405					CMP	R4,R5	:FPS CORRECT?
5711	033600	001402					BEQ	DD42	
5712	033602	000137	033612				JMP	DDERO	:BAD FPS
5713	033606	000137	034530			DD42:	JMP	DDDONE	
5714	033612	010437	001242			DDERO:	MOV	R4,\$TMP4	:FPS ERROR
5715	033616	010537	001240				MOV	R5,\$TMP3	
5716	033622	104164				1\$:	ERROR	+164	
5717	033624	000137	034530				JMP	DDDONE	
5718	033630					DDER1:			
	033630	012737	034420	001240			MOV	#DDP1,\$TMP3	
	033636	012737	034420	001242			MOV	#DDP1,\$TMP4	
	033644	012737	034400	001244			MOV	#DDDATO,\$TMP5	
	033652	012737	034520	001246			MOV	#DDP9,\$TMP6	
	033660	104165				1\$:	ERROR	+165	
	033662	000137	034530				JMP	DDDONE	
5719	033666					DDER2:			
	033666	012737	034420	001240			MOV	#DDP1,\$TMP3	
	033674	012737	034420	001242			MOV	#DDP1,\$TMP4	
	033702	012737	034400	001244			MOV	#DDDATO,\$TMP5	
	033710	012737	034520	001246			MOV	#DDP9,\$TMP6	
	033716	104176				1\$:	ERROR	+176	
	033720	000137	034530				JMP	DDDONE	
5720	033724					DDER3:			
	033724	012737	034420	001240			MOV	#DDP1,\$TMP3	
	033732	012737	034430	001242			MOV	#DDP2,\$TMP4	
	033740	012737	034400	001244			MOV	#DDDATO,\$TMP5	
	033746	012737	034410	001246			MOV	#DDP0,\$TMP6	



	033754	104214			1\$:	ERROR	+214
	033756	000137	034530			JMP	DDDONE
5721	033762				DDER4:	MOV	#DDP2,\$TMP3
	033762	012737	034430	001240		MOV	#DDP1,\$ P4
	033770	012737	034420	001242		MOV	#DDDATO,\$TMP5
	033776	012737	034400	001244		MOV	#DDP0,\$TMP6
	034004	012737	034410	001246	1\$:	ERROR	+200
	034012	104200				JMP	DDDONE
5722	034014	000137	034530		DDER5:	MOV	#DDP6,\$TMP3
	034020					MOV	#DDP3,\$TMP4
	034020	012737	034470	001240		MOV	#DDDATO,\$TMP5
	034026	012737	034440	001242		MOV	#DDP7,\$TMP6
	034034	012737	034400	001244	1\$:	ERROR	+165
	034042	012737	034500	001246		JMP	DDDONE
	034050	104165			DDER6:	MOV	#DDP6,\$TMP3
5723	034052	000137	034530			MOV	#DDP3,\$TMP4
	034056					MOV	#DDDATO,\$TMP5
	034056	012737	034470	001240		MOV	#DDP7,\$TMP6
	034064	012737	034440	001242	1\$:	ERROR	+201
	034072	012737	034400	001244		JMP	DDDONE
	034100	012737	034500	001246	DDER7:	MOV	#DDP3,\$TMP3
	034106	104201				MOV	#DDP6,\$TMP4
5724	034110	000137	034530			MOV	#DDDATO,\$TMP5
	034114					MOV	#DDP7,\$TMP6
	034114	012737	034440	001240	1\$:	ERROR	+165
	034122	012737	034470	001242		JMP	DDDONE
	034130	012737	034400	001244	DDER8:	MOV	#DDP3,\$TMP3
	034136	012737	034500	001246		MOV	#DDP6,\$TMP4
	034144	104165				MOV	#DDDATO,\$TMP5
5725	034146	000137	034530			MOV	#DDP7,\$TMP6
	034152				1\$:	ERROR	+202
	034152	012737	034440	001240		JMP	DDDONE
	034160	012737	034470	001242	DDER9:	MOV	#DDP5,\$TMP3
	034166	012737	034400	001244		MOV	#DDP4,\$TMP4
	034174	012737	034500	001246		MOV	#DDDATO,\$TMP5
	034202	104202				MOV	#DDP8,\$TMP6
	034204	000137	034530		1\$:	ERROR	+165
5726	034210					JMP	DDDONE
	034210	012737	034460	001240	DDER10:	MOV	#DDP5,\$TMP3
	034216	012737	034450	001242		MOV	#DDP4,\$TMP4
	034224	012737	034400	001244		MOV	#DDDATO,\$TMP5
	034232	012737	034510	001246		MOV	#DDP8,\$TMP6
	034240	104165			1\$:	ERROR	+203
5727	034242	000137	034530			JMP	DDDONE
	034246				DDER11:	MOV	#DDP4,\$TMP3
	034246	012737	034460	001240		MOV	#DDP5,\$TMP4
	034254	012737	034450	001242		MOV	#DDDATO,\$TMP5
	034262	012737	034400	001244		MOV	#DDP8,\$TMP6
	034270	012737	034510	001246	1\$:	ERROR	+203
	034276	104203				JMP	DDDONE
5728	034300	000137	034530				
	034304				DDER11:	MOV	#DDP4,\$TMP3
	034304	012737	034450	001240		MOV	#DDP5,\$TMP4
	034312	012737	034460	001242		MOV	#DDDATO,\$TMP5
	034320	012737	034400	001244		MOV	#DDP8,\$TMP6
	034326	012737	034510	001246	1\$:	ERROR	+165
	034334	104165					



```

5729 034336 000137 034530          JMP      DDDONE
      034342          DDERR12:  MOV     #DDP4,$TMP3
      034342 012737 034450 001240    MOV     #DDP5,$TMP4
      034350 012737 034460 001242    MOV     #DDDATO,$TMP5
      034356 012737 034400 001244    MOV     #DDP8,$TMP6
      034364 012737 034510 001246    1$:    ERROR  +204
      034372 104204          JMP      DDDONE
      034374 000137 034530          DDDATO: .WORD 0,0,0,0
5730 034400 000000 000000 000000  DDP0:   .WORD 0,0,0,0
5731 034410 000000 000000 000000  DDP1:   .WORD 100200,0,0,0    :-DDP2
5732 034420 100200 000000 000000  DDP2:   .WORD 200,0,0,0      :-DDP1
5733 034430 000200 000000 000000  DDP3:   .WORD 1100,0,0,0     :EXP=4 :FRAC=...110...
5734 034440 001100 000000 000000  DDP4:   .WORD 600,0,0,0     :EXP=3 :FRAC=...100...
5735 034450 000600 000000 000000  DDP5:   .WORD 101100,0,0,0  :-DDP3
5736 034460 101100 000000 000000  DDP6:   .WORD 100600,0,0,0  :-DDP4
5737 034470 100600 000000 000000  DDP7:   .WORD 1000,0,0,0    :DDP3+DDP6
5738 034500 001000 000000 000000  DDP8:   .WORD 101000,0,0,0  :DDP5+DDP4
5739 034510 101000 000000 000000  DDP9:   .WORD 100400,0,0,0  :DDP1+DDP1
5740 034520 100400 000000 000000  DDDONE: RSETUP
5741 034530 104413

```

: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?).



5749

```
.SBTTL TEST # 35 - SUBD TEST
:*****
:*TEST 35      SUBD TEST
:*
:* THIS IS A TEST OF THE SUBD INSTRUCTION.
:* BOTH A POSITIVE AND A NEGATIVE NUMBER
:* IS SUBTRACTED FROM IT SELF
:*
:*****
```

```
TST35: SCOPE
:      USE POSITIVE OPERANDS
EE1:   MOV      #1$, $LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
:      MOV      #3200,R4        ;SET FIO, FIV, AND FD
:      LDFPS    R4
:      MOV      #EE2,$TMP2
:      MOV      #EEP1,R0        ;SET ACO OPERAND
:      LDD      (R0),ACO
:      MOV      #EEP1,R0        ;FSPC
EE2:   SUBD     (R0),ACO        ;TEST INSTRUCTION
:      STFPS    R5             ;GET FPS
:      MOV      #EEDATO,R0      ;GET THE RESULT
:      STD      ACO,(R0)
:      MOV      #EEO,R1        ;IS IT CORRECT?
:      MOV      #4,R2
EE3:   CMP      (R0)+,(R1)+
:      BEQ      EE6
:      MOV      #EEDATO,R0      ;DID A BAD
:      MOV      #EEP2,R1        ;CONSTANT (NOT 57)
:      MOV      #4,R2          ;GET GENERATED
EE4:   CMP      (R0)+,(R1)+    ;FOR THE ALLIGNMENT
:      BEQ      EE5           ;FLOWS?
:      JMP      EEER1          ;DATA ERROR.D
EE5:   SOB      R2,EE4
:      JMP      EEER2          ;BAD CONSTANT.D
EE6:   SOB      R2,EE3
:      BIS      #4,R4
:      CMP      R4,R5          ;FPS CORRECT?
:      BEQ      EE7
:      JMP      EEERO          ;BAD FPS
:      ;USE NEGATIVE OPERANDS
EE7:   MOV      #1$, $LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
:      MOV      #3200,R4        ;SET FIO, FIV, AND FD
:      LDFPS    R4
:      MOV      #EE8,$TMP2
:      MOV      #EEP3,R0        ;SET ACO OPERAND
:      LDD      (R0),ACO
:      MOV      #EEP3,R0        ;FSPC
EE8:   SUBD     (R0),ACO        ;TEST INSTRUCTION
:      STFPS    R5             ;GET FPS
:      MOV      #EEDATO,R0      ;GET THE RESULT
:      STD      ACO,(R0)
:      MOV      #EEO,R1        ;IS IT CORRECT?
:      MOV      #4,R2
EE9:   CMP      (R0)+,(R1)+
:      BEQ      EE12
```



5795	034750	012700	035232			MOV	#EEDATO,R0	
5796	034754	012701	035304			MOV	#EEP4,R1	
5797	034760	012702	000004			MOV	#4,R2	
5798	034764	022021			EE10:	CMP	(R0)+,(R1)+	
5799	034766	001402				BEQ	EE11	
5800	034770	000137	035136			JMP	EEER3	
5801	034774	077205			EE11:	SOB	R2,EE10	
5802	034776	000137	035174			JMP	EEER4	
5803	035002	077220			EE12:	SOB	R2,EE9	
5804	035004	052704	000004			BIS	#4,R4	
5805	035010	020405				CMP	R4,R5	
5806	035012	001402				BEQ	EE13	
5807	035014	000137	035024			JMP	EEER0	
5808	035020	000137	035314		EE13:	JMP	EEDONE	
5809	035024	010437	001242		EEER0:	MOV	R4,\$TMP4	
5810	035030	010537	001240			MOV	R5,\$TMP3	
5811	035034	104205			1\$:	ERROR	+205	
5812	035036	000137	035314			JMP	EEDONE	
5813	035042				EEER1:			
	C35042	012737	035254	001240		MOV	#EEP1,\$TMP3	
	035050	012737	035254	001242		MOV	#EEP1,\$TMP4	
	035056	012737	035232	001244		MOV	#EEDATO,\$TMP5	
	035064	012737	035242	001246		MOV	#EEO,\$TMP6	
	035072	104206			1\$:	ERROR	+206	
	035074	000137	035314			JMP	EEDONE	
5814	035100				EEER2:			
	035100	012737	035254	001240		MOV	#EEP1,\$TMP3	
	035106	012737	035254	001242		MOV	#EEP1,\$TMP4	
	035114	012737	035232	001244		MOV	#EEDATO,\$TMP5	
	035122	012737	035242	001246		MOV	#EEO,\$TMP6	
	035130	104207			1\$:	ERROR	+207	
	035132	000137	035314			JMP	EEDONE	
5815	035136				EEER3:			
	035136	012737	035274	001240		MOV	#EEP3,\$TMP3	
	035144	012737	035274	001242		MOV	#EEP3,\$TMP4	
	035152	012737	035232	001244		MOV	#EEDATO,\$TMP5	
	035160	012737	035242	001246		MOV	#EEO,\$TMP6	
	035166	104206			1\$:	ERROR	+206	
	035170	000137	035314			JMP	EEDONE	
5816	035174				EEER4:			
	035174	012737	035274	001240		MOV	#EEP3,\$TMP3	
	035202	012737	035274	001242		MOV	#EEP3,\$TMP4	
	035210	012737	035232	001244		MOV	#EEDATO,\$TMP5	
	035216	012737	035242	001246		MOV	#EEO,\$TMP6	
	035224	104207			1\$:	ERROR	+207	
	035226	000137	035314			JMP	EEDONE	
5817	035232	000000	000000	000000	EEDATO:	.WORD	0,0,0,0	
5818	035242	000000	000000	000000	EEO:	.WORD	0,0,0,0,0	
5819	035254	000200	000000	000000	EEP1:	.WORD	200,0,0,0	
5820	035264	000400	000000	000000	EEP2:	.WORD	400,0,0,0	
5821	035274	100200	000000	000000	EEP3:	.WORD	100200,0,0,0	
5822	035304	100400	000000	000000	EEP4:	.WORD	100400,0,0,0	
5823	035314				EEDONE:			
	035314	104413				RSETUP		

```

;DID A BAD
;CONSTANT (NOT 57)
;GET GENERATED
;FOR THE ALLIGNMENT
;FLOWS?
;DATA ERROR.D
;BAD CONSTANT.D
;FPS CORRECT?
;BAD FPS.
;BAD FPS

```

```

;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?).



5833

```

.SBTTL TEST # 36 - NORMALIZE ALGORITHM TEST
*****
: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.
:
*****
    
```

```

5834 035316 000004
5835 035320
5836 035326 012737 035326 001110
5837 035332 170104
5838 035334 012737 035354 001236
5839 035342 012700 035652
5840 035346 172410
5841 035350 012700 035662
5842 035354 172010
5843 035356 170205
5844 035360 012700 035622
5845 035364 174010
5846 035366 012701 035672
5847 035372 012702 000004
5848 035376 022021
5849 035400 001401
5850 035402 000470
5851 035404 077204
5852 035406 020405
5853 035410 001401
5854 035412 000437
5855
5856
5857 035414
5858 035422 012737 035422 001110
5859 035426 170104
5860 035430 012737 035450 001236
5861 035436 012700 035632
5862 035442 172410
5863 035444 012700 035642
5864 035450 172010
5865 035452 170205
5866 035454 012700 035622
5867 035460 174010
5868 035462 012701 035672
5869 035466 012702 000004
5870 035472 022021
5871 035474 001401
5872 035476 000413
5873 035500 077204
5874 035502 020405
5875 035504 001401
5876 035506 000401
    
```

```

TST36: SCOPE
:USE DATA PATTERNS THAT REQUIRE ONLY ONE LEFT SHIFT TO NORMALIZE
FF1:
1$:      MOV      #1$, $LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
          MOV      #3200, R4      ;SET F10, F1V, AND FD
          LDFPS   R4
          MOV      #FF2, $TMP2
          MOV      #FFP2, R0      ;SET ACO OPERAND
          LDD     (R0), ACO
          MOV      #FFP3, R0      ;FSPC
          ADDD   (R0), ACO      ;TEST INSTRUCTION
          STFPS   R5              ;GET FPS
          MOV      #FFDAT0, R0    ;GET THE RESULT
          STD     ACO, (R0)
          MOV      #FFP4, R1      ;IS IT CORRECT
          MOV      #4, R2
FF3:     CMP      (R0)+, (R1)+
          BEQ     FF4
          BR      FFER2          ;BAD DATA
FF4:     SOB     R2, FF3
          CMP     R4, R5          ;FPS CORRECT?
          BEQ     FF5
          BR      FFER0          ;BAD FPS
:USE DATA PATTERNS WHICH REQUIRE 56 LEFT SHIFTS TO NORMALIZE
:THE RESULT
FF5:
1$:      MOV      #1$, $LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
          MOV      #3200, R4      ;SET F1U, F1V, AND FD
          LDFPS   R4
          MOV      #FF6, $TMP2
          MOV      #FFP0, R0      ;SET ACO OPERAND
          LDD     (R0), ACO
          MOV      #FFP1, R0      ;FSRC
          ADDD   (R0), ACO      ;TEST INSTRUCTION
          STFPS   R5              ;GET FPS
          MOV      #FFDAT0, R0    ;GET THE RESULT
          STD     ACO, (R0)
          MOV      #FFP4, R1      ;IS IT CORRECT
          MOV      #4, R2
FF7:     CMP      (R0)+, (R1)+
          BEQ     FF10
          BR      FFER1          ;BATA
FF10:    SOB     R2, FF7
          CMP     R4, R5          ;FPS CORRECT?
          BEQ     FF11
          BR      FFER0          ;BAD FPS
    
```



```

5877 035510 000474          FF11:  BR      FFDONE
5878
5879 035512 010537 001240  FFER0:  MOV     R5,$TMP3
5880 035516 010437 001242      MOV     R4,$TMP4
5881 035522 104164          1$:     ERROR  +164
5882 035524 000466          BR      FFDONE
5883
5884 035526          FFER1:  MOV     #FFP1,$TMP3
      035526 012737 035642 001240      MOV     #FFP0,$TMP4
      035534 012737 035632 001242      MOV     #FFDAT0,$TMP5
      035542 012737 035622 001244      MOV     #FFP4,$TMP6
      035550 012737 035672 001246      1$:     ERROR  +210
      035556 104210          JMP     FFDONE
      035560 000137 035702
5885
5886 035564          FFER2:  MOV     #FFP3,$TMP3
      035564 012737 035662 001240      MOV     #FFP2,$TMP4
      035572 012737 035652 001242      MOV     #FFDAT0,$TMP5
      035600 012737 035622 001244      MOV     #FFP4,$TMP6
      035606 012737 035672 001246      1$:     ERROR  +210
      035614 104210          JMP     FFDONE
      035616 000137 035702
5887
5888
5889 035622 000000 000000 000000  FFDAT0: .WORD  0,0,0,0
5890 035632 016000 000000 000000  FFP0:   .WORD  16000,0,0,1
5891 035642 116000 000000 000000  FFP1:   .WORD  116000,0,0,0
5892 035652 000500 000000 000000  FFP2:   .WORD  500,0,0,0
5893 035662 100400 000000 000000  FFP3:   .WORD  100400,0,0,0
5894 035672 000200 000000 000000  FFP4:   .WORD  200,0,0,0
5895 035702          FFDONE:
5896 035702          TST37:

```

:FFP4=FFP0+FFP1=FFP3+FFP4



5898

```

035702
035702 000004
035704 005037 001102
035710 005037 001302
035714 005237 001324
035720 042737 100000 001324
035726 005327
035730 000001
035732 003072
035734 012737
035736 000001
035740 035730
035742 104401 035750
035746 000407

035766
035766 013746 001324

035772 104405
035774 104401 036002
036000 000421

036044
036044 013746 001112

036050 104405
036052 104401 001313
036056 005037 001112
036062 013700 000042
036066 001414
036070 005046
036072 012746 036100
036076 000426

036100
036100 013700 000042
036104 001405
036106 000005
036110 004710
036112 000240
036114 000240
036116 000240
036120
036120 104400
036122 042716 000020
036126 032777 010000 143004
036134 001005
036136 005137 036162
    
```

```

.SBTTL END OF PASS ROUTINE
*****
*INCREMENT THE PASS NUMBER ($PASS)
*INDICATE END-OF-PROGRAM AFTER 1 PASSES THRU THE PROGRAM
*TYPE 'END PASS #XXXXX TOTAL NUMBER OF ERRORS SINCE LAST REPORT YYYYY'
*WHERE XXXXX AND YYYYY ARE DECIMAL NUMBERS
*IF SW12=1 INHIBIT TRACE TRAP
*IF THERES A MONITOR GO TO IT
*IF THERE ISN'T JUMP TO LOOP
$EOP:
SCOPE
CLR $STNM ::ZERO THE TEST NUMBER
CLR $TIMES ::ZERO THE NUMBER OF ITERATIONS
INC $PASS ::INCREMENT THE PASS NUMBER
BIC #100000,$PASS ::DON'T ALLOW A NEG. NUMBER
DEC (PC)+ ::LOOP?
$EOPCT: .WORD 1
BGT $DOAGN ::YES
MOV (PC)+,@(PC)+ ::RESTORE COUNTER
$ENDCT: .WORD 1
$EOPCT
TYPE ,65$ ::TYPE ASCIZ STRING
BR 64$ ::GET OVER THE ASCIZ
::65$: .ASCIZ <12><15>/END PASS #/
64$:
MOV $PASS,-(SP) ::SAVE $PASS FOR TYPEOUT
::TYPE PASS NUMBER
TYPDS
TYPE ,67$ ::GO TYPE--DECIMAL ASCII WITH SIGN
BR 66$ ::TYPE ASCIZ STRING
::67$: .ASCIZ / TOTAL ERRORS SINCE LAST REPORT /
66$:
MOV $ERTTL,-(SP) ::SAVE $ERTTL FOR TYPEOUT
::TOTAL NUMBER OF ERRORS
TYPDS
TYPE , $CRLF ::GO TYPE--DECIMAL ASCII WITH SIGN
::TYPE CARRIAGE RETURN, LINE FEED
$GET42: CLR $ERTTL ::CLEAR ERROR TOTAL
MOV @#42,R0 ::GET MONITOR ADDRESS
BEQ $DOAGN ::BRANCH IF NO MONITOR
CLR -(SP) ::INSURE THE 'T' BIT IS CLEAR
MOV # $CLR.T,-(SP) ::SETUP FOR AN RTI OR RTT
BR $RTRN ::GO DO AN RTI OR RTT TO LOAD THE PSW
::WITH A CLEARED 'T' BIT

$CLR.T:
MOV @#42,R0 ::INSURE R0 CONTAINS THE MONITORS
BEQ $DOAGN ::RETURN ADDRESS
RESET ::CLEAR THE WORLD
$ENDAD: JSR PC,(R0) ::GO TO MONITOR
NOP ::SAVE ROOM
NOP ::FOR
NOP ::ACT11

$DOAGN:
TRAP ::PUSH OLD PSW AND PC ON STACK
BIC #20,(SP) ::CLEAR THE 'T' BIT
BIT #BIT12,@SWR ::RUN WITH TRACE TRAP?
BNE 1$ ::BR IF NO
COM $TBIT ::IS IT TIME FOR TRACE TRAP
    
```



036142 100402  
036144 052716 000020  
036150 012746 036156  
036154 000002

1\$: BMI 1\$  
BIS #20,(SP)  
MOV #SLOOP,-(SP)  
\$RTN: RTI

::BR IF NO  
::SET TRACE TRAP  
::JUMP TO START OF TEST  
::RETURN--THIS IS CHANGED TO  
::AN 'RTI' IF 'RTI' IS A LEGAL  
::INSTRUCTION

036156  
036156 000137  
036160 004300  
036162 000000  
036164 377 000

\$LOOP: JMP @ (PC)+  
\$RTNAD: .WORD LOOP  
\$TBIT: .WORD 0  
\$ENULL: .BYTE -1,-1,0  
.EVEN

::RETURN  
::'T' BIT STATE INDICATOR  
::NULL CHARACTER STRING



5900

.SBTTL SCOPE HANDLER ROUTINE

```

*****
*THIS ROUTINE CONTROLS THE LOOPING OF SUBTESTS. IT WILL INCREMENT
*AND LOAD THE TEST NUMBER($TSTNM) INTO THE DISPLAY REG.(DISPLAY<7:0>)
*AND LOAD THE ERROR FLAG ($ERFLG) INTO DISPLAY<15:08>
*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
*SW14=1      LOOP ON TEST
*SW11=1      INHIBIT ITERATIONS
*SW09=1      LOOP ON ERROR
*SW08=1      LOOP ON TEST IN SWR<7:0>
*CALL
*          SCOPE          ;;SCOPE=IOT
    
```

```

036170          $SCOPE:
036170 104407          CKSWR          ;;TEST FOR CHANGE IN SOFT-SWR
036172 032777 040000 142740 1$:      BIT      #BIT14,@SWR          ;;LOOP ON PRESENT TEST?
036200 001131          BNE      $OVER          ;;YES IF SW14=1
          ;;*****START OF CODE FOR THE XOR TESTER*****
036202 000416          $XTSTR: BR      6$          ;;IF RUNNING ON THE 'XOR' TESTER CHANGE
          ;;THIS INSTRUCTION TO A 'NOP' (NOP=240)
036204 013746 000004          MOV      @#ERRVEC,-(SP)          ;;SAVE THE CONTENTS OF THE ERROR VECTOR
036210 012737 036230 000004          MOV      #5$,@#ERRVEC          ;;SET FOR TIMEOUT
036216 005737 177060          TST      @#177060          ;;TIME OUT ON XOR?
036222 012637 000004          MOV      (SP)+,@#ERRVEC          ;;RESTORE THE ERROR VECTOR
036226 000500          BR      $SVLAD          ;;GO TO THE NEXT TEST
036230 022626          5$:      CMP      (SP)+,(SP)+          ;;CLEAR THE STACK AFTER A TIME OUT
036232 012637 000004          MOV      (SP)+,@#ERRVEC          ;;RESTORE THE ERROR VECTOR
036236 000440          BR      7$          ;;LOOP ON THE PRESENT TEST
036240          6$:;*****END OF CODE FOR THE XOR TESTER*****
036240 032777 000400 142672          BIT      #BIT08,@SWR          ;;LOOP ON SPEC. TEST?
036246 001404          BEQ      2$          ;;BR IF NO
036250 127737 142664 001102          CMPB   @SWR,$TSTNM          ;;ON THE RIGHT TEST? SWR<7:0>
036256 001502          BEQ      $OVER          ;;BR IF YES
036260 013737 177765 036502 2$:      MOV      177766,CPSAVE          ;;MOVE CPU ERR REG VALUE TO LOC FOR TST :DPM001
036266 032737 000001 036502          BIT      #BIT00,CPSAVE          ;;SEE IF THE POWER MONITOR BIT IS ON :DPM001
036274 001406          BEQ      2000$          ;;BRANCH TO CONTINUE ROUTINE IF CLEAR :DPM001
036276 042737 000001 177766          BIC      #BIT00,177766          ;;CLEAR THE BIT FOUND TO BE SET :DPM001
036304 104177          ERROR +177          ;;CALL SPECIAL POWER FAIL BIT ERROR CALL :DPM001
036306 105037 001103          CLRB   $ERFLG          ;;CLEAR THE ERROR FLAG FOR NEXT TEST :DPM001
036312 105737 001103          2000$: TSTB   $ERFLG          ;;HAS AN ERROR OCCURRED?
036316 001421          BEQ      3$          ;;BR IF NO
036320 123737 001115 001103          CMPB   $ERMAX,$ERFLG          ;;MAX. ERRORS FOR THIS TEST OCCURRED?
036326 101015          BHI      3$          ;;BR IF NO
036330 032777 001000 142602          BIT      #BIT09,@SWR          ;;LOOP ON ERROR?
036336 001404          BEQ      4$          ;;BR IF NO
036340 013737 001110 001106 7$:      MOV      $LPERR,$LPADR          ;;SET LOOP ADDRESS TO LAST SCOPE
036346 000446          BR      $OVER
036350 105037 001103          4$:      CLRB   $ERFLG          ;;ZERO THE ERROR FLAG
036354 005037 001302          CLR      $TIMES          ;;CLEAR THE NUMBER OF ITERATIONS TO MAKE
036360 000415          BR      1$          ;;ESCAPE TO THE NEXT TEST
036362 032777 004000 142550 3$:      BIT      #BIT11,@SWR          ;;INHIBIT ITERATIONS?
036370 001011          BNE      1$          ;;BR IF YES
036372 005737 001324          TST      $PASS          ;;IF FIRST PASS OF PROGRAM
036376 001406          BEQ      1$          ;;INHIBIT ITERATIONS
036400 005237 001104          INC      $ICNT          ;;INCREMENT ITERATION COUNT
036404 023737 001302 001104          CMP      $TIMES,$ICNT          ;;CHECK THE NUMBER OF ITERATIONS MADE
    
```



036412	002024			BGE	\$OVER	::BR IF MORE ITERATION REQUIRED
036414	012737	000001	001104	1\$: MOV	#1,\$ICNT	::REINITIALIZE THE ITERATION COUNTER
036422	013737	036500	001302	MOV	\$MXCNT,\$TIMES	::SET NUMBER OF ITERATIONS TO DO
036430	105237	001102		\$SVLAD: INCB	\$STNM	::COUNT TEST NUMBERS
036434	113737	001102	001322	MOVB	\$STNM,\$TESTN	::SET TEST NUMBER IN APT MAILBOX
036442	011637	001106		MOV	(SP),\$LPADR	::SAVE SCOPE LOOP ADDRESS
036446	011637	001110		MOV	(SP),\$LPERR	::SAVE ERROR LOOP ADDRESS
036452	005037	001304		CLR	\$ESCAPE	::CLEAR THE ESCAPE FROM ERROR ADDRESS
036456	112737	000001	001115	MOVB	#1,\$ERMAX	::ONLY ALLOW ONE(1) ERROR ON NEXT TEST
036464	013777	001102	142450	\$OVER: MOV	\$STNM,@DISPLAY	::DISPLAY TEST NUMBER
036472	013716	001106		MOV	\$LPADR,(SP)	::FUDGE RETURN ADDRESS
036476	000002			RTI		::FIXES PS
036500	000001			\$MXCNT: 1		::MAX. NUMBER OF ITERATIONS
036502	000000			CPSAVE: .WORD	0	::LOCATION TO SAVE CPU ERR REG CONTENTS ;DPM001



5902

.SBTTL ERROR HANDLER ROUTINE

\*\*\*\*\*  
\*THIS ROUTINE WILL INCREMENT THE ERROR FLAG AND THE ERROR COUNT,  
\*SAVE THE ERROR ITEM NUMBER AND THE ADDRESS OF THE ERROR CALL  
\*AND GO TO ERTYPE ON ERROR  
\*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:  
\*SW15=1 HALT ON ERROR  
\*SW13=1 INHIBIT ERROR TYPEOUTS  
\*SW10=1 BELL ON ERROR  
\*SW09=1 LOOP ON ERROR  
\*CALL  
\* ERROR N ;;ERROR=EMT AND N=ERROR ITEM NUMBER

```
036504 000000 IBSAVE: .WORD 0 ;LOC'N TO HOLD $ITEMB DURING DUAL ERR ;DPM001
036506 105037 036504 $ERROR: CLRB IBSAVE ;CLEAR THE ITEM BYTE SAVE LOCATION ;DPM001
036512 104407 CKSWR ;TEST FOR CHANGE IN SOFT-SWR
036514 105237 001103 7$: INCB $ERFLG ;SET THE ERROR FLAG
036520 001775 BEQ 7$ ;DON'T LET THE FLAG GO TO ZERO
036522 013777 001102 142412 MOV $TSTNM,@DISPLAY ;DISPLAY TEST NUMBER AND ERROR FLAG
036530 032777 002000 142402 BIT #BIT10,@SWR ;BELL ON ERROR?
036536 001402 BEQ 1$ ;NO - SKIP
036540 104401 001306 TYPE ,SBELL ;RING BELL
036544 005237 001112 1$: INC $ERTTL ;COUNT THE NUMBER OF ERRORS
036550 011637 001116 MOV (SP), $ERRPC ;GET ADDRESS OF ERROR INSTRUCTION
036554 162737 000002 001116 SUB #2, $ERRPC
036562 117737 142330 001114 MOV $ERRPC, $ITEMB ;STRIP AND SAVE THE ERROR ITEM CODE
036570 122737 000177 001114 CMPB #177, $ITEMB ;SEE IF THIS IS THE POWER FAIL CALL ;DPM001
036576 001421 BEQ 1000$ ;BRANCH AROUND PMB TEST IF IT IS ;DPM001
036600 013737 177766 036502 MOV 177766, CPSAVE ;MOVE CPU ERR REG TO CPSAVE FOR TEST ;DPM001
036606 032737 000001 036502 BIT #BIT00, CPSAVE ;SEE IF POWER MONITOR BIT IS SET ;DPM001
036614 001412 BEQ 1000$ ;BRANCH IF OK ;DPM001
036616 042737 000001 177766 BIC #BIT00, 177766 ;CLEAR THE BIT FOUND SET ;DPM001
036624 013737 001116 036504 MOV $ERRPC, IBSAVE ;SAVE $ERRPC ;DPM001
036632 104177 ERROR +177 ;CALL SPECIAL POWER MONITOR BIT ERROR ;DPM001
036634 013737 036504 001116 MOV IBSAVE, $ERRPC ;RESTORE $ERRPC
036642 1000$: BIT #BIT13, @SWR ;SKIP TYPEOUT IF SET
036642 032777 020000 142270 BNE 20$ ;SKIP TYPEOUTS
036650 001004 JSR PC, ERTYPE ;GO TO USER ERROR ROUTINE
036652 004737 041342 TYPE , $CRLF
036656 104401 001313 20$: CMPB #APTENV, $ENV ;RUNNING IN APT MODE
036662 122737 000001 001336 BNE 2$ ;NO, SKIP APT ERROR REPORT
036670 001007 MOV $ITEMB, 21$ ;SET ITEM NUMBER AS ERROR NUMBER
036672 113737 001114 036704 JSR PC, $ATY4 ;REPORT FATAL ERROR TO APT
036700 004737 040174 21$: .BYTE 0
036704 000 .BYTE 0
036705 000 BR 22$ ;APT ERROR LOOP
036706 000777 22$: TST IBSAVE ;SEE IF POWER FAIL ERROR CALL ;DPM001
036710 005737 036504 2$: BNE 3$ ;BRANCH IF NOT - HALT NOT ALLOWED ;DPM001
036714 001005 TST @SWR ;HALT ON ERROR
036716 005777 142216 BPL 3$ ;SKIP IF CONTINUE
036722 100002 HALT ;HALT ON ERROR!
036724 000000 CKSWR ;TEST FOR CHANGE IN SOFT-SWR
036726 104407 3$: BIT #BIT09, @SWR ;LOOP ON ERROR SWITCH SET?
036730 032777 001000 142202 BEQ 4$ ;BR IF NO
036736 001405
```



```

036740 005737 036504      TST      IBSAVE      ;SEE IF THIS IS THE PWR MONITOR ERROR ;DPM001
036744 001263              BNE      7$          ;BRANCH BACK IF SO - NO FUDGING      ;DPM001
036746 013716 001110      MOV      $LPERR,(SP) ;:FUDGE RETURN FOR LOOPING
036752 005737 001304      4$:     TST      $ESCAPE ;:CHECK FOR AN ESCAPE ADDRESS
036756 001405              BEQ      5$          ;:BR IF NONE
036760 005737 036504      TST      IBSAVE      ;SEE IF THIS IS THE PWR MONITOR ERROR ;DPM001
036764 001253              BNE      7$          ;BRANCH BACK IF SO - NO FUDGING      ;DPM001
036766 013716 001304      MOV      $ESCAPE,(SP);:FUDGE RETURN ADDRESS FOR ESCAPE
036772              5$:
036772 022737 036110 000042  CMP      #$ENDAD,@#42 ;:ACT-11 AUTO-ACCEPT?
037000 001001              BNE      6$          ;:BRANCH IF NO
037002 000000              HALT                ;:YES
037004              6$:
037004 032777 001000 142126  BIT      #BIT09,@SWR
037012 001013              BNE      ERM10
037014 011637 001162      MOV      (SP),$RFGO ;SEE IF ERROR #377
037020 062737 177776 001162  ADD      #-2,$REGO
037026 122777 000377 142126  CMPB    #377,@$REGO
037034 001002              BNE      ERM10
037036 062716 000002      ADD      #2,(SP)
037042 000002      ERM10: RTI
  
```



5904

.SBTTL SAVE AND RESTORE R0-R5 ROUTINES  
:\*\*\*\*\*:\*\*\*\*\*:\*\*\*\*\*

:\*SAVE R0-R5  
:\*CALL:  
:\* SAVREG  
:\*UPON RETURN FROM \$SAVREG THE STACK WILL LOOK LIKE:

:\*TOP---(+16)  
:\* +2---(+18)  
:\* +4---R5  
:\* +6---R4  
:\* +8---R3  
:\*+10---R2  
:\*+12---R1  
:\*+14---R0

037044  
037044 010046  
037046 010146  
037050 010246  
037052 010346  
037054 010446  
037056 010546  
037060 016646 000022  
037064 016646 000022  
037070 016646 000022  
037074 016646 000022  
037100 000002

\$SAVREG:  
MOV R0,-(SP) ::PUSH R0 ON STACK  
MOV R1,-(SP) ::PUSH R1 ON STACK  
MOV R2,-(SP) ::PUSH R2 ON STACK  
MOV R3,-(SP) ::PUSH R3 ON STACK  
MOV R4,-(SP) ::PUSH R4 ON STACK  
MOV R5,-(SP) ::PUSH R5 ON STACK  
MOV 22(SP),-(SP) ::SAVE PS OF MAIN FLOW  
MOV 22(SP),-(SP) ::SAVE PC OF MAIN FLOW  
MOV 22(SP),-(SP) ::SAVE PS OF CALL  
MOV 22(SP),-(SP) ::SAVE PC OF CALL  
RTI

:\*RESTORE R0-R5  
:\*CALL:  
:\* RESREG

037102  
037102 012666 000022  
037106 012666 000022  
037112 012666 000022  
037116 012666 000022  
037122 012605  
037124 012604  
037126 012603  
037130 012602  
037132 012601  
037134 012600  
037136 000002

\$RESREG:  
MOV (SP)+,22(SP) ::RESTORE PC OF CALL  
MOV (SP)+,22(SP) ::RESTORE PS OF CALL  
MOV (SP)+,22(SP) ::RESTORE PC OF MAIN FLOW  
MOV (SP)+,22(SP) ::RESTORE PS OF MAIN FLOW  
MOV (SP)+,R5 ::POP STACK INTO R5  
MOV (SP)+,R4 ::POP STACK INTO R4  
MOV (SP)+,R3 ::POP STACK INTO R3  
MOV (SP)+,R2 ::POP STACK INTO R2  
MOV (SP)+,R1 ::POP STACK INTO R1  
MOV (SP)+,R0 ::POP STACK INTO R0  
RTI



5906

.SBTTL TYPE ROUTINE

\*\*\*\*\*  
 \*ROUTINE TO TYPE ASCIZ MESSAGE. MESSAGE MUST TERMINATE WITH A 0 BYTE.  
 \*THE ROUTINE WILL INSERT A NUMBER OF NULL CHARACTERS AFTER A LINE FEED.  
 \*NOTE1: \$NULL CONTAINS THE CHARACTER TO BE USED AS THE FILLER CHARACTER.  
 \*NOTE2: \$FILLS CONTAINS THE NUMBER OF FILLER CHARACTERS REQUIRED.  
 \*NOTE3: \$FILLC CONTAINS THE CHARACTER TO FILL AFTER.  
 \*

\*CALL:  
 \*1) USING A TRAP INSTRUCTION  
 \* TYPE ,MESADR ;:MESADR IS FIRST ADDRESS OF AN ASCIZ STRING  
 \*OR  
 \* TYPE  
 \* MESADR

037140	105737	001157	\$TYPE:	TSTB	\$TPFLG	:: IS THERE A TERMINAL?
037144	100002			BPL	1\$	:: BR IF YES
037146	000000			HALT		:: HALT HERE IF NO TERMINAL
037150	000430			BR	3\$	:: LEAVE
037152	010046		1\$:	MOV	RO,-(SP)	:: SAVE RO
037154	017600	000002		MOV	@2(SP),RO	:: GET ADDRESS OF ASCIZ STRING
037160	122737	000001	001336	CMPB	#APTENV,\$ENV	:: RUNNING IN APT MODE
037166	001011			BNE	62\$	:: NO,GO CHECK FOR APT CONSOLE
037170	132737	000100	001337	BITB	#APTSPOOL,\$ENVM	:: SPOOL MESSAGE TO APT
037176	001405			BEQ	62\$	:: NO,GO CHECK FOR CONSOLE
037200	010037	037210		MOV	RO,61\$	:: SETUP MESSAGE ADDRESS FOR APT
037204	004737	040164		JSR	PC,\$ATY3	:: SPOOL MESSAGE TO APT
037210	000000		61\$:	.WORD	0	:: MESSAGE ADDRESS
037212	132737	000040	001337	BITB	#APTCSUP,\$ENVM	:: APT CONSOLE SUPPRESSED
037220	001003			BNE	60\$	:: YES,SKIP TYPE OUT
037222	112046		2\$:	MOVB	(RO)+,-(SP)	:: PUSH CHARACTER TO BE TYPED ONTO STACK
037224	001005			BNE	4\$	:: BR IF IT ISN'T THE TERMINATOR
037226	005726			TST	(SP)+	:: IF TERMINATOR POP IT OFF THE STACK
037230	012600		60\$:	MOV	(SP)+,RO	:: RESTORE RO
037232	067716	000002	3\$:	ADD	#2,(SP)	:: ADJUST RETURN PC
037236	000002			RTI		:: RETURN
037240	122716	000011	4\$:	CMPB	#HT,(SP)	:: BRANCH IF <HT>
037244	001430			BEQ	8\$	
037246	122716	000200		CMPB	#CRLF,(SP)	:: BRANCH IF NOT <CRLF>
037252	001006			BNE	5\$	
037254	005726			TST	(SP)+	:: POP <CR><LF> EQUIV
037256	104401			TYPE		:: TYPE A CR AND LF
037260	001313			\$CRLF		
037262	105037	037500		CLRB	\$CHARCNT	:: CLEAR CHARACTER COUNT
037266	000755			BR	2\$	:: GET NEXT CHARACTER
037270	004737	037352	5\$:	JSR	PC,\$TYPEC	:: GO TYPE THIS CHARACTER
037274	123726	001156	6\$:	CMPB	\$FILLC,(SP)+	:: IS IT TIME FOR FILLER CHARS.?
037300	001350			BNE	2\$	:: IF NO GO GET NEXT CHAR.
037302	013746	001154		MOV	\$NULL,-(SP)	:: GET # OF FILLER CHARS. NEEDED
						:: AND THE NULL CHAR.
037306	105366	000001	7\$:	DECB	1(SP)	:: DOES A NULL NEED TO BE TYPED?
037312	002770			BLT	6\$	:: BR IF NO--GO POP THE NULL OFF OF STACK
037314	004737	037352		JSR	PC,\$TYPEC	:: GO TYPE A NULL
037320	105337	037500		DECB	\$CHARCNT	:: DO NOT COUNT AS A COUNT
037324	000770			BR	7\$	:: LOOP
				:HORIZONTAL TAB	PROCESSOR	
037326	112716	000040	8\$:	MOVB	#' ,(SP)	:: REPLACE TAB WITH SPACE



```

037332 004737 037352 9$: JSR PC,$TYPEC ;;TYPE A SPACE
037336 132737 000007 037500 BITB #7,$CHARCNT ;;BRANCH IF NOT AT
037344 001372 BNE 9$ ;;TAB STOP
037346 005726 TST (SP)+ ;;POP SPACE OFF STACK
037350 000724 BR 2$ ;;GET NEXT CHARACTER
037352 $TYPEC:
037352 105777 141556 TSTB @$TKS ;;CHAR IN KYBD BUFFER? :MJD001
037356 100022 BPL 10$ ;;BR IF NOT :MJD001
037360 017746 141562 MOV @$TKB,-(SP) ;;GET CHAR :MJD001
037364 042716 177600 BIC #177600,(SP) ;;STRIP EXTRANEIOUS BITS :MJD001
037370 122716 000023 CMPB #$XOFF,(SP) ;;WAS CHAR XOFF :MJD001
037374 001012 BNE 102$ ;;BR IF NOT :MJD001
037376 105777 141542 101$: TSTB @$TKS ;;WAIT FOR CHAR :MJD001
037402 100375 BPL 101$ :MJD001
037404 117716 141536 MOVB @$TKB,(SP) ;;GET CHAR :MJD001
037410 042716 177600 BIC #177600,(SP) ;;STRIP IT :MJD001
037414 122716 000021 CMPB #$XON,(SP) ;;WAS IT XON? :MJD001
037420 001366 BNE 101$ ;;BR IF NOT :MJD001
037422 102$: TST (SP)+ ;;FIX STACK :MJD001
037424 10$: TSTB @$TPS ;;WAIT UNTIL PRINTER IS READY :MJD001
037424 105777 141520 BPL 10$ :MJD001
037430 100375 CMPB 2(SP),#$XON ;;IS CHARACTER A RANDOM XON? :RAN001
037432 126627 000002 000021 BEQ $TYPEX ;;BRANCH IF YES :RAN001
037440 001420 MOVB 2(SP),@$TPB ;;LOAD CHAR TO BE TYPED INTO DATA REG.
037442 116677 000002 141502 CMPB #CR,2(SP) ;;IS CHARACTER A CARRIAGE RETURN?
037450 122766 000015 000002 BNE 1$ ;;BRANCH IF NO
037456 001003 CLRB $CHARCNT ;;YES--CLEAR CHARACTER COUNT
037460 105037 037500 BR $TYPEX ;;EXIT
037464 000406 1$: CMPB #LF,2(SP) ;;IS CHARACTER A LINE FEED?
037466 122766 000012 000002 BEQ $TYPEX ;;BRANCH IF YES
037474 001402 INCB (PC)+ ;;COUNT THE CHARACTER
037476 105227 $CHARCNT: .WORD 0 ;;CHARACTER COUNT STORAGE
037500 000000 $TYPEX: RTS PC
037502 000207

```



5908

```

.SBTTL BINARY TO OCTAL (ASCII) AND TYPE
*****
*THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 6-DIGIT
*OCTAL (ASCII) NUMBER AND TYPE IT.
*$TYPOS---ENTER HERE TO SETUP SUPPRESS ZEROS AND NUMBER OF DIGITS TO TYPE
*CALL:
*   MOV     NUM,-(SP)      ;;NUMBER TO BE TYPED
*   TYPOS   ;;CALL FOR TYPEOUT
*   .BYTE  N              ;;N=1 TO 6 FOR NUMBER OF DIGITS TO TYPE
*   .BYTE  M              ;;M=1 OR 0
*                               ;;1=TYPE LEADING ZEROS
*                               ;;0=SUPPRESS LEADING ZEROS
*$TYPON----ENTER HERE TO TYPE OUT WITH THE SAME PARAMETERS AS THE LAST
*$TYPOS OR $TYPOC
*CALL:
*   MOV     NUM,-(SP)      ;;NUMBER TO BE TYPED
*   TYPON   ;;CALL FOR TYPEOUT
*$TYPOC---ENTER HERE FOR TYPEOUT OF A 16 BIT NUMBER
*CALL:
*   MOV     NUM,-(SP)      ;;NUMBER TO BE TYPED
*   TYPOC   ;;CALL FOR TYPEOUT
037504 017646 000000 037727 $TYPOS: MOV @ (SP),-(SP) ;;PICKUP THE MODE
037510 116637 000001 037727 MOV 1(SP), $OFILL ;;LOAD ZERO FILL SWITCH
037516 112637 037731 MOV (SP)+, $OMODE+1 ;;NUMBER OF DIGITS TO TYPE
037522 062716 000002 ADD #2, (SP) ;;ADJUST RETURN ADDRESS
037526 000406 BR $TYPON
037530 112737 000001 037727 $TYPOC: MOV #1, $OFILL ;;SET THE ZERO FILL SWITCH
037536 112737 000006 037731 MOV #6, $OMODE+1 ;;SET FOR SIX(6) DIGITS
037544 112737 000005 037726 $TYPON: MOV #5, $OCNT ;;SET THE ITERATION COUNT
037552 010346 MOV R3, -(SP) ;;SAVE R3
037554 010446 MOV R4, -(SP) ;;SAVE R4
037556 010546 MOV R5, -(SP) ;;SAVE R5
037560 113704 037731 MOV $OMODE+1, R4 ;;GET THE NUMBER OF DIGITS TO TYPE
037564 005404 NEG R4
037566 062704 000006 ADD #6, R4 ;;SUBTRACT IT FOR MAX. ALLOWED
037572 110437 037730 MOV R4, $OMODE ;;SAVE IT FOR USE
037576 113704 037727 MOV $OFILL, R4 ;;GET THE ZERO FILL SWITCH
037602 016605 000012 MOV 12(SP), R5 ;;PICKUP THE INPUT NUMBER
037606 005003 CLR R3 ;;CLEAR THE OUTPUT WORD
037610 006105 1$: ROL R5 ;;ROTATE MSB INTO 'C'
037612 000404 BR 3$ ;;GO DO MSB
037614 006105 2$: ROL R5 ;;FORM THIS DIGIT
037616 006105 ROL R5
037620 006105 ROL R5
037622 010503 MOV R5, R3
037624 006103 3$: ROL R3 ;;GET LSB OF THIS DIGIT
037626 105337 037730 DECB $OMODE ;;TYPE THIS DIGIT?
037632 100016 BPL 7$ ;;BR IF NO
037634 042703 177770 BIC #177770, R3 ;;GET RID OF JUNK
037640 001002 BNE 4$ ;;TEST FOR 0
037642 005704 TST R4 ;;SUPPRESS THIS 0?
037644 001403 BEQ 5$ ;;BR IF YES
037646 005204 4$: INC R4 ;;DON'T SUPPRESS ANYMORE 0'S
037650 052703 000060 BIS #'0, R3 ;;MAKE THIS DIGIT ASCII
037654 052703 000040 5$: BIS #' , R3 ;;MAKE ASCII IF NOT ALREADY

```



037660	110337	037724		MOVB	R3,8\$	::SAVE FOR TYPING
037664	104401	037724		TYPE	.8\$	::GO TYPE THIS DIGIT
037670	105337	037726	7\$:	DECB	\$OCNT	::COUNT BY 1
037674	003347			BGT	2\$	::BR IF MORE TO DO
037676	002402			BLT	6\$	::BR IF DONE
037700	005204			INC	R4	::INSURE LAST DIGIT ISN'T A BLANK
037702	000744			BR	2\$	::GO DO THE LAST DIGIT
037704	012605		6\$:	MOV	(SP)+,R5	::RESTORE R5
037706	012604			MOV	(SP)+,R4	::RESTORE R4
037710	012603			MOV	(SP)+,R3	::RESTORE R3
037712	016666	000002 000004		MOV	2(SP),4(SP)	::SET THE STACK FOR RETURNING
037720	012616			MOV	(SP)+,(SP)	
037722	000002			RTI		::RETURN
037724	000		8\$:	.BYTE	0	::STORAGE FOR ASCII DIGIT
037725	000			.BYTE	0	::TERMINATOR FOR TYPE ROUTINE
037726	000		\$OCNT:	.BYTE	0	::OCTAL DIGIT COUNTER
037727	000		\$OFILL:	.BYTE	0	::ZERO FILL SWITCH
037730	000000		\$OMODE:	.WORD	0	::NUMBER OF DIGITS TO TYPE



5910

```

.SBTTL CONVERT BINARY TO DECIMAL AND TYPE ROUTINE
:*****
:THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 5-DIGIT
:SIGNED DECIMAL (ASCII) NUMBER AND TYPE IT. DEPENDING ON WHETHER THE
:NUMBER IS POSITIVE OR NEGATIVE A SPACE OR A MINUS SIGN WILL BE TYPED
:BEFORE THE FIRST DIGIT OF THE NUMBER. LEADING ZEROS WILL ALWAYS BE
:REPLACED WITH SPACES.
:CALL:
:*      MOV      NUM,-(SP)      ::PUT THE BINARY NUMBER ON THE STACK
:*      TYPDS                    ::GO TO THE ROUTINE
$TYPDS:
037732 010046      MOV      R0,-(SP)      ::PUSH R0 ON STACK
037732 010146      MOV      R1,-(SP)      ::PUSH R1 ON STACK
037734 010246      MOV      R2,-(SP)      ::PUSH R2 ON STACK
037736 010346      MOV      R3,-(SP)      ::PUSH R3 ON STACK
037740 010546      MOV      R5,-(SP)      ::PUSH R5 ON STACK
037742 012746 020200      MOV      #20200,-(SP)  ::SET BLANK SWITCH AND SIGN
037744 016605 000020      MOV      20(SP),R5    ::GET THE INPUT NUMBER
037750 100004      BPL      1$           ::BR IF INPUT IS POS.
037754 005405      NEG      R5           ::MAKE THE BINARY NUMBER POS.
037756 112766 000055 000001      MOVVB   #'-,1(SP)    ::MAKE THE ASCII NUMBER NEG.
037760 005000      CLR      R0           ::ZERO THE CONSTANTS INDEX
037766 012703 040146      MOV      #$DBLK,R3    ::SETUP THE OUTPUT POINTER
037770 112723 000040      MOVVB   #' ,(R3)+    ::SET THE FIRST CHARACTER TO A BLANK
040000 005002      CLR      R2           ::CLEAR THE BCD NUMBER
040002 016001 040136      MOV      $DTBL(R0),R1 ::GET THE CONSTANT
040006 160105      3$:     SUB      R1,R5    ::FORM THIS BCD DIGIT
040010 002402      BLT     4$           ::BR IF DONE
040012 005202      INC     R2           ::INCREASE THE BCD DIGIT BY 1
040014 000774      BR      3$
040016 060105      4$:     ADD     R1,R5    ::ADD BACK THE CONSTANT
040020 005702      TST     R2           ::CHECK IF BCD DIGIT=0
040022 001002      BNE     5$           ::FALL THROUGH IF 0
040024 105716      TS'B    (SP)         ::STILL DOING LEADING 0'S?
040026 100407      BMI     7$           ::BR IF YES
040030 106316      5$:     ASLB   (SP)         ::MSD?
040032 103003      BCC     6$           ::BR IF NO
040034 116663 000001 177777      MOVVB   1(SP),-1(R3)  ::YES--SET THE SIGN
040042 052702 000060      6$:     BIS     #'0,R2    ::MAKE THE BCD DIGIT ASCII
040046 052702 000040      7$:     BIS     #' ,R2    ::MAKE IT A SPACE IF NOT ALREADY A DIGIT
040052 110223      MOVVB   R2,(R3)+    ::PUT THIS CHARACTER IN THE OUTPUT BUFFER
040054 005720      TST     (R0)+       ::JUST INCREMENTING
040056 020027 000010      CMP     R0,#10      ::CHECK THE TABLE INDEX
040062 002746      BLT     2$           ::GO DO THE NEXT DIGIT
040064 003002      BGT     8$           ::GO TO EXIT
040066 010502      MOV     R5,R2       ::GET THE LSD
040070 000764      BR      6$           ::GO CHANGE TO ASCII
040072 105726      8$:     TSTB   (SP)+     ::WAS THE LSD THE FIRST NON-ZERO?
040074 100003      BPL     9$           ::BR IF NO
040076 116663 177777 177776      MOVVB   -1(SP),-2(R3) ::YES--SET THE SIGN FOR TYPING
040104 105013      9$:     CLRB   (R3)     ::SET THE TERMINATOR
040106 012605      MOV     (SP)+,R5    ::POP STACK INTO R5
040110 012603      MOV     (SP)+,R3    ::POP STACK INTO R3
040112 012602      MOV     (SP)+,R2    ::POP STACK INTO R2
040114 012601      MOV     (SP)+,R1    ::POP STACK INTO R1
040116 012600      MOV     (SP)+,R0    ::POP STACK INTO R0
040120 104401 040146      TYPE   , $DBLK     ::NOW TYPE THE NUMBER
    
```



```
040124 016666 000002 000004      MOV      2(SP),4(SP)      ;;ADJUST THE STACK
040132 012616                      MOV      (SP)+,(SP)
040134 000002                      RTI          ;;RETURN TO USER
040136 023420      $DTBL: 10000.
040140 001750                      1000.
040142 000144                      100.
040144 000012                      10.
040146                      $DBLK: .BLKW 4
```



5912

.SBTTL APT COMMUNICATIONS ROUTINE

```

*****
040156 112737 000001 040422 $ATY1: MOVB #1,$FFLG      ;;TO REPORT FATAL ERROR
040164 112737 000001 040420 $ATY3: MOVB #1,$MFLG      ;;TO TYPE A MESSAGE
040172 000403
040174 112737 000001 040422 $ATY4: MOVB #1,$FFLG      ;;TO ONLY REPORT FATAL ERROR
040202 $ATYC:
040202 010046          MOV R0,-(SP)      ;;PUSH R0 ON STACK
040204 010146          MOV R1,-(SP)      ;;PUSH R1 ON STACK
040206 105737 040420          TSTB $MFLG      ;;SHOULD TYPE A MESSAGE?
040212 001450          BEQ 5$          ;;IF NOT: BR
040214 122737 000001 001336      CMPB #APTENV,$ENV      ;;OPERATING UNDER APT?
040222 001031          BNE 3$          ;;IF NOT: BR
040224 132737 000100 001337      BITB #APTSPOOL,$ENVM   ;;SHOULD SPOOL MESSAGES?
040232 001425          BEQ 3$          ;;IF NOT: BR
040234 017600 000004          MOV @4(SP),R0      ;;GET MESSAGE ADDR.
040240 062766 000002 000004          ADD #2,4(SP)      ;;BUMP RETURN ADDR.
040246 005737 001316          1$: TST $MSGTYPE      ;;SEE IF DONE W/ LAST XMISSION?
040252 001375          BNE 1$          ;;IF NOT: WAIT
040254 010037 001332          MOV R0,$MSGAD      ;;PUT ADDR IN MAILBOX
040260 105720          2$: TSTB (R0)+      ;;FIND END OF MESSAGE
040262 001376          BNE 2$
040264 163700 001332          SUB $MSGAD,R0      ;;SUB START OF MESSAGE
040270 006200          ASR R0          ;;GET MESSAGE LNTH IN WORDS
040272 010037 001334          MOV R0,$MSGGLT      ;;PUT LENGTH IN MAILBOX
040276 012737 000004 001316          MOV #4,$MSGTYPE      ;;TELL APT TO TAKE MSG.
040304 000413          BR 5$
040306 017637 000004 040332 3$: MOV @4(SP),4$      ;;PUT MSG ADDR IN JSR LINKAGE
040314 062766 000002 000004          ADD #2,4(SP)      ;;BUMP RETURN ADDRESS
040322 013746 177776          MOV 177776,-(SP)    ;;PUSH 177776 ON STACK
040326 004737 037140          JSR PC,$TYPE      ;;CALL TYPE MACRO
040332 000000          4$: .WORD 0
040334          5$:
040334 105737 040422          10$: TSTB $FFLG      ;;SHOULD REPORT FATAL ERROR?
040340 001416          BEQ 12$         ;;IF NOT: BR
040342 005737 001336          TST $ENV        ;;RUNNING UNDER APT?
040346 001413          BEQ 12$         ;;IF NOT: BR
040350 005737 001316          11$: TST $MSGTYPE      ;;FINISHED LAST MESSAGE?
040354 001375          BNE 11$        ;;IF NOT: WAIT
040356 017637 000004 001320          MOV @4(SP),$FATAL   ;;GET ERROR #
040364 062766 000002 000004          ADD #2,4(SP)      ;;BUMP RETURN ADDR.
040372 005237 001316          INC $MSGTYPE      ;;TELL APT TO TAKE ERROR
040376 105037 040422          12$: CLRB $FFLG      ;;CLEAR FATAL FLAG
040402 105037 040421          CLRB $LFLG      ;;CLEAR LOG FLAG
040406 105037 040420          CLRB $MFLG      ;;CLEAR MESSAGE FLAG
040412 012601          MOV (SP)+,R1      ;;POP STACK INTO R1
040414 012600          MOV (SP)+,R0      ;;POP STACK INTO R0
040416 000207          RTS PC          ;;RETURN
040420 000          $MFLG: .BYTE 0      ;;MESSG. FLAG
040421 000          $LFLG: .BYTE 0      ;;LOG FLAG
040422 000          $FFLG: .BYTE 0      ;;FATAL FLAG
          .EVEN
          APTSIZE=200
          APTENV=001
          APTSPOOL=100
          APTCSUP=040
000200
000001
000100
000040

```



5914

```

.SBTTL TTY INPUT ROUTINE
*****
.ENABL LSB
*****
*SOFTWARE SWITCH REGISTER CHANGE ROUTINE.
*ROUTINE IS ENTERED FROM THE TRAP HANDLER, AND WILL
*SERVICE THE TEST FOR CHANGE IN SOFTWARE SWITCH REGISTER TRAP CALL
*WHEN OPERATING IN TTY FLAG MODE.
040424 022737 000176 001140 $CKSWR: CMP #SWREG,SWR ;;IS THE SOFT-SWR SELECTED?
040432 001074 BNE 15$ ;;BRANCH IF NO
040434 105777 140504 TSTB @STKS ;;CHAR THERE?
040440 100071 BPL 15$ ;;IF NO, DON'T WAIT AROUND
040442 117746 140500 MOVB @STKB,-(SP) ;;SAVE THE CHAR
040446 042716 177600 BIC #^C177,(SP) ;;STRIP-OFF THE ASCII
040452 022726 000007 CMP #7,(SP)+ ;;IS IT A CONTROL G?
040456 001062 BNE 15$ ;;NO, RETURN TO USER
040460 123727 001134 000001 CMPB $AUTOB,#1 ;;ARE WE RUNNING IN AUTO-MODE?
040466 001456 BEQ 15$ ;;BRANCH IF YES
040470 104401 041043 TYPE ,SCNTLG ;;ECHO THE CONTROL-G (^G)
040474 104401 041050 $GTSWR: TYPE ,SMSWR ;;TYPE CURRENT CONTENTS
040500 013746 000176 MOV SWREG,-(SP) ;;SAVE SWREG FOR TYPEOUT
040504 104402 TYPOC ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
040506 104401 041061 TYPE ,SMNEW ;;PROMPT FOR NEW SWR
040512 005046 19$: CLR -(SP) ;;CLEAR COUNTER
040514 005046 CLR -(SP) ;;THE NEW SWR
040516 105777 140422 7$: TSTB @STKS ;;CHAR THERE?
040522 100375 BPL 7$ ;;IF NOT TRY AGAIN
040524 117746 140416 MOVB @STKB,-(SP) ;;PICK UP CHAR
040530 042716 177600 BIC #^C177,(SP) ;;MAKE IT 7-BIT ASCII
040534 021627 000025 9$: CMP (SP),#25 ;;IS IT A CONTROL-U?
040540 001005 BNE 10$ ;;BRANCH IF NOT
040542 104401 041036 TYPE ,SCNTLU ;;YES, ECHO CONTROL-U (^U)
040546 062706 000006 20$: ADD #6,SP ;;IGNORE PREVIOUS INPUT
040552 000757 BR 19$ ;;LET'S TRY IT AGAIN
040554 021627 000015 10$: CMP (SP),#15 ;;IS IT A <CR>?
040560 001022 BNE 16$ ;;BRANCH IF NO
040562 005766 000004 TST 4(SP) ;;YES, IS IT THE FIRST CHAR?
040566 001403 BEQ 11$ ;;BRANCH IF YES
040570 016677 000002 140342 MOV 2(SP),@SWR ;;SAVE NEW SWR
040576 062706 000006 11$: ADD #6,SP ;;CLEAR UP STACK
040602 104401 001313 14$: TYPE ,SCRLF ;;ECHO <CR> AND <LF>
040606 123727 001135 000001 CMPB $INTAG,#1 ;;RE-ENABLE TTY KBD INTERRUPTS?
040614 001003 BNE 15$ ;;BRANCH IF NOT
040616 012777 000100 140320 MOV #100,@STKS ;;RE-ENABLE TTY KBD INTERRUPTS
040624 000002 RTI ;;RETURN
040626 004737 037352 16$: JSR PC,$TYPEC ;;ECHO CHAR
040632 021627 000060 CMP (SP),#60 ;;CHAR < 0?
040636 002420 BLT 18$ ;;BRANCH IF YES
040640 021627 000067 CMP (SP),#67 ;;CHAR > 7?
040644 003015 BGT 18$ ;;BRANCH IF YES
040646 042726 000060 BIC #60,(SP)+ ;;STRIP-OFF ASCII
040652 005766 000002 TST 2(SP) ;;IS THIS THE FIRST CHAR
040656 001403 BEQ 17$ ;;BRANCH IF YES
040660 006316 ASL (SP) ;;NO, SHIFT PRESENT
040662 006316 ASL (SP) ;; CHAR OVER TO MAKE
040664 006316 ASL (SP) ;; ROOM FOR NEW ONE.
040666 005266 000002 17$: INC 2(SP) ;;KEEP COUNT OF CHAR

```



040672 056616 177776  
040676 000707  
040700 104401 001312  
040704 000720

```

BIS -2(SP), (SP)      ::SET IN NEW CHAR
BR 7$                  ::GET THE NEXT ONE
18$: TYPE $QUES       ::TYPE ?<CR><LF>
BR 20$                 ::SIMULATE CONTROL-U
.DSABL LSB

```

```

*****
*THIS ROUTINE WILL INPUT A SINGLE CHARACTER FROM THE TTY
*CALL:
*   RDCHR                ::INPUT A SINGLE CHARACTER FROM THE TTY
*   RETURN HERE          ::CHARACTER IS ON THE STACK
*                          ::WITH PARITY BIT STRIPPED OFF

```

040706 011646  
040710 016666 000004 000002  
040716 105777 140222  
040722 100375  
040724 117766 140216 000004  
040732 042766 177600 000004  
040740 026627 000004 000023  
040746 001013  
040750 105777 140170  
040754 100375  
040756 117746 140164  
040762 042716 177600  
040766 022627 000021  
040772 001366  
040774 000750  
040776 026627 000004 000021  
041004 001744  
041006 026627 000004 000140  
041014 002407  
041016 026627 000004 000175  
041024 003003  
041026 042766 000040 000004  
041034 000002  
041036 136 125 015  
041043 136 107 015  
041050 015 012 123  
041061 040 040 116

```

$RDCHR: MOV (SP), -(SP)      ::PUSH DOWN THE PC
MOV 4(SP), 2(SP)           ::SAVE THE PS
1$: TSTB @$TKS              ::WAIT FOR
BPL 1$                      ::A CHARACTER
MOVB @$TKB, 4(SP)          ::READ THE TTY
BIC #^C<177>, 4(SP)        ::GET RID OF JUNK IF ANY
CMP 4(SP), #23             ::IS IT A CONTROL-S?
BNE 3$                      ::BRANCH IF NO
2$: TSTB @$TKS              ::WAIT FOR A CHARACTER
BPL 2$                      ::LOOP UNTIL ITS THERE
MOVB @$TKB, -(SP)          ::GET CHARACTER
BIC #^C<177>, (SP)         ::MAKE IT 7-BIT ASCII
CMP (SP)+, #21             ::IS IT A CONTROL-Q?
BNE 2$                      ::IF NOT DISCARD IT
BR 1$                       ::YES, RESUME
3$: CMP 4(SP), #XON         ::IS IT A RANDOM XON?
BEQ 1$                      ::BRANCH IF YES
CMP 4(SP), #140            ::IS IT UPPER CASE?
BLT 4$                      ::BRANCH IF YES
CMP 4(SP), #175           ::IS IT A SPECIAL CHAR?
BGT 4$                      ::BRANCH IF YES
BIC #40, 4(SP)             ::MAKE IT UPPER CASE
RTI                        ::GO BACK TO USER
4$: $CNTLU: .ASCIZ /^U/<15><12> ::CONTROL 'U'
$CNTLG: .ASCIZ /^G/<15><12> ::CONTROL 'G'
$MSWR: .ASCIZ <15><12>/SWR = /
$MNEW: .ASCIZ / NEW = /

```

:RAN001  
:RAN001



5916

041072 010046  
041074 016600 000002  
041100 005740  
041102 111000  
041104 006300  
041106 016000 041126  
041112 000200  
  
041114 011646  
041116 016666 000004 000002  
041124 000002

.SBTTL TRAP DECODER

\*\*\*\*\*  
\*THIS ROUTINE WILL PICKUP THE LOWER BYTE OF THE "TRAP" INSTRUCTION  
\*AND USE IT TO INDEX THROUGH THE TRAP TABLE FOR THE STARTING ADDRESS  
\*OF THE DESIRED ROUTINE. THEN USING THE ADDRESS OBTAINED IT WILL  
\*GO TO THAT ROUTINE.

```
$TRAP: MOV    RO,-(SP)      ;;SAVE RO
        MOV    2(SP),RO    ;;GET TRAP ADDRESS
        TST   -(RO)       ;;BACKUP BY 2
        MOVB  (RO),RO     ;;GET RIGHT BYTE OF TRAP
        ASL   RO          ;;POSITION FOR INDEXING
        MOV   $TRPAD(RO),RO ;;INDEX TO TABLE
        RTS   RO          ;;GO TO ROUTINE
```

;;THIS IS USE TO HANDLE THE "GETPRI" MACRO

```
$TRAP2: MOV   (SP),-(SP)  ;;MOVE THE PC DOWN
        MOV   4(SP),2(SP) ;;MOVE THE PSW DOWN
        RTI                          ;;RESTORE THE PSW
```

.SBTTL TRAP TABLE

\*THIS TABLE CONTAINS THE STARTING ADDRESSES OF THE ROUTINES CALLED  
\*BY THE "TRAP" INSTRUCTION.

ROUTINE

041126 041114  
041130 037140  
041132 037530  
041134 037504  
041136 037544  
041140 037732  
041142 040474  
041144 040424  
041146 040706  
041150 037044  
041152 037102  
5917 041154 042154  
5918 000030

```
$TRPAD: .WORD  $TRAP2
        $TYPE  ;;CALL=TYPE      TRAP+1(104401) TTY TYPEOUT ROUTINE
        $TYPOC ;;CALL=TYPOC    TRAP+2(104402) TYPE OCTAL NUMBER (WITH LEADING ZEROS)
        $TYPOS ;;CALL=TYPOS    TRAP+3(104403) TYPE OCTAL NUMBER (NO LEADING ZEROS)
        $TYPON ;;CALL=TYPON    TRAP+4(104404) TYPE OCTAL NUMBER (AS PER LAST CALL)
        $TYPDS ;;CALL=TYPDS    TRAP+5(104405) TYPE DECIMAL NUMBER (WITH SIGN)
        $GTSWR ;;CALL=GTSWR    TRAP+6(104406) GET SOFT-SWR SETTING
        $CKSWR ;;CALL=CKSWR    TRAP+7(104407) TEST FOR CHANGE IN SOFT-SWR
        $RDCHR ;;CALL=RDCHR    TRAP+10(104410) TTY TYPEIN CHARACTER ROUTINE
        $SAVREG ;;CALL=SAVREG  TRAP+11(104411) SAVE R0-R5 ROUTINE
        $RESREG ;;CALL=RESREG  TRAP+12(104412) RESTORE R0-R5 ROUTINE
        .RSET  ;;CALL=RSETUP   TRAP+13(104413) ROUTINE TO RESET STACK AND FPS
$TERM=-.$TRPAD
```



5920

```

.SBTTL POWER DOWN AND UP ROUTINES
:*****
:POWER DOWN ROUTINE
041156 012737 041334 000024 $PWRDN: MOV    #$ILLUP,@#PWRVEC  ;;SET FOR FAST UP
041164 012737 000340 000026      MOV    #340,@#PWRVEC+2  ;;PRIO:7
041172 010046      MOV    R0,-(SP)        ;;PUSH R0 ON STACK
041174 010146      MOV    R1,-(SP)        ;;PUSH R1 ON STACK
041176 010246      MOV    R2,-(SP)        ;;PUSH R2 ON STACK
041200 010346      MOV    R3,-(SP)        ;;PUSH R3 ON STACK
041202 010446      MOV    R4,-(SP)        ;;PUSH R4 ON STACK
041204 010546      MOV    R5,-(SP)        ;;PUSH R5 ON STACK
041206 017746 137726      MOV    @SWR,-(SP)      ;;PUSH @SWR ON STACK
041212 010637 041340      MOV    SP,$SAVR6      ;;SAVE SP
041216 012737 041230 000024      MOV    #$PWRUP,@#PWRVEC ;;SET UP VECTOR
041224 000000      HALT
041226 000776      BR     -2             ;;HANG UP
:*****
:POWER UP ROUTINE
041230 012737 041334 000024 $PWRUP: MOV    #$ILLUP,@#PWRVEC  ;;SET FOR FAST DOWN
041236 013706 041340      MOV    $SAVR6,SP      ;;GET SP
041242 005037 041340      CLR    $SAVR6        ;;WAIT LOOP FOR THE TTY
041246 005237 041340 1$:   INC    $SAVR6        ;;WAIT FOR THE INC
041252 001375      BNE    1$            ;;OF WORD
041254 012677 137660      MOV    (SP)+,@SWR     ;;POP STACK INTO @SWR
041260 012605      MOV    (SP)+,R5      ;;POP STACK INTO R5
041262 012604      MOV    (SP)+,R4      ;;POP STACK INTO R4
041264 012603      MOV    (SP)+,R3      ;;POP STACK INTO R3
041266 012602      MOV    (SP)+,R2      ;;POP STACK INTO R2
041270 012601      MOV    (SP)+,R1      ;;POP STACK INTO R1
041272 012600      MOV    (SP)+,R0      ;;POP STACK INTO R0
041274 012737 041156 000024      MOV    #$PWRDN,@#PWRVEC ;;SET UP THE POWER DOWN VECTOR
041302 012737 000340 000026      MOV    #340,@#PWRVEC+2 ;;PRIO:7
041310 104401      TYPE    POWERM        ;;REPORT THE POWER FAILURE
041312 042224      $PWRMG: .WORD POWERM  ;;POWER FAIL MESSAGE POINTER
041314 012716      MOV    (PC)+,(SP)    ;;RESTART AT START
041316 003616      $PWRAD: .WORD START   ;;RESTART ADDRESS
041320 042766 000020 000002      BIC    #20,2(SP)     ;;CLEAR 'T' BIT
041326 005037 036162      CLR    $TBIT        ;;CLEAR THE 'T' BIT FLAG
041332 000002      RTI
041334 000000      $ILLUP: HALT        ;;THE POWER UP SEQUENCE WAS STARTED
041336 000776      BR     -2           ;;BEFORE THE POWER DOWN WAS COMPLETE
041340 000000      $SAVR6: 0           ;;PUT THE SP HERE

```



```

5922
5923
5924
5925
5926
5927
5928
5929
5930 041342 104401 001313
5931 041346 113737 001102 001232
5932 041354 042737 177400 001232
5933 041362 013737 001116 001234
5934 041370 010046
5935
5936 041372 113700 001114
5937 041376 042700 177400
5938 041402 001005
5939
5940 041404 013746 001116
5941 041410 104402
5942 041412 000137 041740
5943
5944 041416 022700 000377
5945 041422 001005
5946 041424 016600 000004
5947 041430 011000
5948 041432 062700 000400
5949 041436 010037 001320
5950 041442 005300
5951 041444 006300
5952 041446 006300
5953 041450 006300
5954 041452 062700 001442
5955
5956 041456 012037 041466
5957 041462 001404
5958 041464 104401
5959 041466 000000
5960 041470 104401
5961 041472 001313
5962
5963 041474 012037 041504
5964 041500 001404
5965 041502 104401
5966 041504 000000
5967 041506 104401
5968 041510 001313
5969
5970 041512 010146
5971 041514 010246
5972 041516 010346
5973
5974 041520 012001
5975 041522 001501
5976
5977 041524 011000
  
```

```

.SBTTL ERROR TYPE OUT ROUTINE
*****
*****
*THIS ROUTINE IS CALLED TO TYPE AN ERROR MESSAGE WHICH IS INCLUDED
*IN THE ERROR MESSAGE DATA TABLE. IT IS CALLED BY THE $ERROR ROUTINE
*OR BY FIRST SETTING $ITEMB EQUAL TO THE ERROR TABLE ITEM TO BE PRINTED
*OUT AND THEN EXECUTING A:
*
*      JSR      PC,ERTYPE
*
ERTYPE: TYPE      $CRLF      ;TYPE A CRLF
        MOVB     $TSTNM,$TMP0
        BIC      #177400,$TMP0
        MOV      $ERRPC,$TMP1 ;GET PC OF CALL
        MOV      RO,-(SP)     ;SAVE RO
        MOVB     $ITEMB,RO    ;GET THE ITEM NUMBER.
        BIC      #177400,RO   ;CLEAR UPPER BYTE EXPOSING OFFSET
        BNE      1$          ;BRANCH IF OFFSET EXISTS
        MOV      $ERRPC,-(SP) ;IF ZERO THEN JUST
        TYPOC   ;PRINT THE PC
        JMP      ERT5
1$:     CMP      #377,RO
        BNE      20$
        MOV      4(SP),RO
        MOV      (RO),RO
        ADD      #400,RO
20$:    MOV      RO,$FATAL    ;MOVE ITEM NUMBER TO $FATAL FOR APT ;DPM001
        DEC      RO
        ASL      RO
        ASL      RO
        ASL      RO
        ADD      # $ERRTB,RO
110$:   MOV      (RO)+,2$
        BEQ      3$          ;PICK UP THE ADDRESS OF THE EM, ERROR MESSAGE
        TYPE
        BEQ      2$          ;BRANCH IF NONE
2$:     .WORD    0
        TYPE
        .WORD    $CRLF
3$:     MOV      (RO)+,4$
        BEQ      5$
        TYPE
4$:     .WORD    0
        TYPE
        .WORD    $CRLF
5$:     MOV      R1,-(SP)    ;SAVE R1,R2 AND R3
        MOV      R2,-(SP)
        MOV      R3,-(SP)
        MOV      (RO)+,R1
        BEQ      ERT4      ;GET THE ADDRESS OF THE DATA TABLE
        ;RETURN IF NO DATA.
        MOV      (RO),RO   ;GET A POINTER TO THE DATA FORMAT TABLE
  
```



5978	041526	105710		ERT1:	TSTB	(R0)	:FORMAT ZERO?
5979	041530	001003			BNE	8\$	:BRANCH IF NOT
5980							
5981	041532	013146			MOV	@(R1)+,-(SP)	:FORMAT ZERO SO TYPE
5982	041534	104402			TYPOC		:AN OCTAL NUMBER.
5983	041536	000465			BR	ERT2	
5984							
5985	041540	122710	000002	8\$:	CMPB	#2,(R0)	:FORMAT TWO?
5986	041544	001006			BNE	9\$	
5987							
5988	041546	013102			MOV	@(R1)+,R2	:FORMAT TWO SO TYPE TWO OCTAL NUMBERS
5989	041550	004737	042054		JSR	PC,TOCTNM	:TYPE 1 NUMBER
5990	041554	004737	042054		JSR	PC,TOCTNM	:TYPE 2ND NUMBER
5991	041560	000454			BR	ERT2	
5992							
5993	041562	122710	000003	9\$:	CMPB	#3,(R0)	:FORMAT THREE?
5994	041566	001007			BNE	10\$	:BRANCH IF NOT
5995							
5996	041570	013102			MOV	@(R1)+,R2	:FORMAT THREE SO TYPE FOUR OCTAL NUMBERS.
5997	041572	012703	000004		MOV	#4,R3	:TYPE 4 NUMBERS
5998	041576	004737	042054	90\$:	JSR	PC,TOCTNM	:TYPE AN OCTAL NUMBER
5999	041602	077303			SOB	R3,90\$	:SUBTRACT 1 AND BRANCH IF NOT DONE
6000	041604	000442			BR	ERT2	
6001							
6002	041606	122710	000004	10\$:	CMPB	#4,(R0)	:FORMAT FOUR?
6003	041612	001004			BNE	11\$	:BRANCH IF NOT
6004							
6005	041614	013146			MOV	@(R1)+,-(SP)	:FORMAR FOUR SO TYPE
6006	041616	104403			TYPOS		:AN OCTAL NUMBER
6007	041620	016			.BYTE	16	:SUPPRESSING LEADING ZEROES.
6008	041621	000			.BYTE	0	
6009	041622	000433			BR	ERT2	
6010							
6011	041624	122710	000005	11\$:	CMPB	#5,(R0)	:FORMAT FIVE?
6012	041630	001005			BNE	13\$	
6013							
6014	041632	012137	041640		MOV	(R1)+,12\$	:FORMAT FIVE SO TYPE AN
6015	041636	104401			TYPE		:ASCIZ STRING.
6016	041640	000000		12\$:	.WORD	0	
6017	041642	000425			BR	ERT3	
6018							
6019	041644	122710	000011	13\$:	CMPB	#11,(R0)	:FORMAT ELEVEN?
6020	041650	001005			BNE	15\$	:BRANCH IF NOT
6021	041652	013137	041660		MOV	@(R1)+,14\$	:FORMAT ELEVEN SO PICK
6022	041656	104401			TYPE		:A POINTER TO AN ASCIZ
6023	041660	000000		14\$:	.WORD	0	:STRING.
6024	041662	000415			BR	ERT3	
6025							
6026	041664	122710	000012	15\$:	CMPB	#12,(R0)	:FORMAT TWELVE?
6027	041670	001007			BNE	17\$	:BRANCH IF NOT
6028							
6029	041672	013102			MOV	@(R1)+,R2	:FORMAT TWELVE SO TYPE
6030	041674	012703	000006		MOV	#6,R3	:TYPE SIX OCTAL NUMBERS
6031	041700	004737	042054	16\$:	JSR	PC,TOCTNM	:TYPE AN OCTAL NUMBER
6032	041704	077303			SOB	R3,16\$	:SUBTRACT 1 AND BRANCH IF NOT DONE YET
6033	041706	000401			BR	ERT2	
6034							



```

6035 041710 000000      17$:      HALT                ;UNDEFINED FORMAT FOR DATA?????
6036
6037 041712 104401      ERT2:     TYPE                ;PRINT A TAB AFTER TYPING
6038 041714 042272      .WORD     $TAB                ;AN DATA TABLE ENTRY
6039                                     ;OF ALL FORMATS EXCEPT
6040                                     ;ASCIZ, FORMATS 5 OR 11
6041
6042 041716 005200      ERT3:     INC      R0          ;POINT TO THE NEXT FORMAT
6043 041720 005711      TST      (R1)                ;END OF DATA TABLE.
6044 041722 001401      REG      ERT4
6045 041724 000700      BR       ERT1
6046
6047 041726 104401      ERT4:     TYPE                ;DONE.
6048 041730 001313      .WORD     $CRLF
6049 041732 012603      MOV      (SP)+,R3            ;RESTORE R1,R2 AND R3
6050 041734 012602      MOV      (SP)+,R2
6051 041736 012601      MOV      (SP)+,R1
6052 041740 012600      ERT5:     MOV      (SP)+,R0    ;RESTORE R0.
6053 041742 000207      RTS      PC                  ;AND RETURN.
6054
6055
6056 041744 041754 042010 042040 PFECWS: .WORD PFECM,PFECDH,PFECAD,PFECFT ;ADRSES OF DATA/ASCII BELOW ;DPM001
6057 041754      120      117      127 PFECM: .ASCIZ ?POWER MONITOR BIT FOUND SET? ;ERROR MESSAGE ;DPM001
6058 042010      124      105      123 PFECDH: .ASCIZ ?TESTNO ERR PC CPUERR? ;ERROR DATA HEADER ;DPM001
6059                                     .EVEN
6060 042040 001322 001116 036502 PFECAD: .WORD $TESTN,$ERRPC,$PSAVE,0 ;ADDRESSES OF DATA ;DPM001
6061 042050      000      000      000 PFECFT: .BYTE 0,0,0,0 ;FORMAT TABLE ;DPM001
6062
6063 042054 012246      TOCTNM: MOV      (R2)+,-(SP) ;MOVE THE NUMBER TO THE STACK FOR PRINTING
6064 042056 104402      TYPOC   ;TYPE AN OCTAL NUMBER
6065 042060 104401 042274      TYPE    ,SPACE ;TYPE A SPACE CHARACTER
6066 042064 000207      RTS     PC                  ;EXIT BACK
    
```



6067  
6068  
6069  
6070  
6071  
6072  
6073 042066 011637 001236  
6074 042073 022626  
6075 042074 170200  
6076 042076 010037 001240  
6077 042102 170300  
6078 042104 010037 001242  
6079 042110 104211  
6080 042112 104413

.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), $TMP2      ;SAVE PC OF TRAP.
          CMP      (SP)+, (SP)+   ;RESTORE SP.
          STFPS    R0              ;GET FPS
          MOV      R0, $TMP3
          STST     R0              ;GET FEC
1$:      MOV      R0, $TMP4
          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?).

```

6081 042114 000137 035702                    JMP    \$EOP







6091  
6092

6093  
6094

6095 042136 011637 001236  
6096 042142 022626  
6097 042144 104213  
6098 042146 104413

6099 042150 000137 035702

.SBTTL CPU SPURIOUS TRAP TO 10 HANDLER

\*\*\*\*\*  
\*\*\*\*\*  
\*THIS ROUTINE REPORTS UNEXPECTED CPU TRAPS TO VECTOR 10.\*

\*  
CPTWO: MOV (SP), \$TMP2 ;SAVE PC OF TRAP.  
CMF (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 \$EOP



6100  
 6101

.SBTTL FLAG RESET AND CONSOLE TEST ROUTINE

```

*****
*****
*THIS ROUTINE WILL BE CALLED AT THE END OF EACH TEST TO
*RESET THE STACK, CLEAR THE FPS AND SEE IF THE USER HAS TYPED
* CONTROL G ON THE TERMINAL. IF THE USER HAS TYPED CONTROL G AND
*THERE IS NO PHYSICAL CONSOLE SWITCH REGISTER THEN THE CONTENTS
*OF THE SOFTWARE SWITCH REGISTER WILL BE TYPED IN OCTAL ON THE
*TELETYPE AND THE USER CAN MODIFY IT.
*

```

6102  
 6103  
 6104  
 6105  
 6106  
 6107  
 6108  
 6109 042154 023727 001140 177570  
 6110  
 6111 042162 001001  
 6112 042164 104407  
 6113  
 6114  
 6115  
 6116 042166 012737 042066 000244  
 6117 042174 012737 042120 000004  
 6118 042202 012737 042136 000010  
 6119 042210 011600  
 6120 042212 012706 001100  
 6121 042216 005004  
 6122 042220 170104  
 6123 042222 000110

```

.RSET:  CMP     SWR,#177570
        BNE     1$
        CKSWR
1$:     MOV     #FPSPUR,FPVECT
        MOV     #CPSPUR,ERRVECT
        MOV     #CPTWO,10
        MOV     (SP),R0
        MOV     #STACK,SP
        CLR     R4
        LDFPS  R4
        JMP     (R0)

```

```

;SEE IF THERE IS A PHYSICAL
;CONSOLE SWITCH REGISTER.
;BRANCH IF NO.
;OTHERWISE TYPE THE CONTENTS
;OF THE PROGRAM VIRTUAL SWITCH REGISTER
;AND GIVE THE USER A CHANCE TO
;MODIFY IT.

;SAVE RETURN ADDRESS.
;RESET THE STACK POINTER.
;CLEAR THE FPS.

;RETURN.

```



Address	Hex	Dec	Hex	Dec	Label	Value	Description
6124						.SBTTL	SPECIAL MESSAGES
6125	042224	200	120	117	POWERM:	.ASCIZ	<CRLF>'POWER FAILURE. PROGRAM RESTARTING.'<CRLF>
6126	042271	000			NULL:	.BYTE	0
6127	042272	011	000		\$TAB:	.ASCIZ	<TAB>
6128	042274	040	040	000	SPACE:	.ASCIZ	' '
6129	042277	200	120	103	LFIEX1:	.ASCIZ	<CRLF>'PC OF LAST FPP INSTRUCTION EXECUTED: '<TAB>
6130	042347	200	114	101	LFIEX2:	.ASCIZ	<CRLF>'LAST FPP INSTRUCTION EXECUTED: '<TAB>
6131	042411	200	106	114	FPSMS:	.ASCIZ	<CRLF>'FLOATING POINT STATUS REGISTER: ' '
6132	042455	200	106	105	FECMS:	.ASCIZ	<CRLF>'FEC: ' '
6133	042466	124	110	105	\$THE:	.ASCIZ	'THE ' '
6134	042473	011	040	111	NOOP1:	.ASCIZ	<TAB>' INSTRUCTION FAILED.'<CRLF>
6135	042522	105	111	124	NOOP15:	.ASCII	'EITHER A BAD CONSTANT WAS GENERATED OR MICROPROGRAM FLOW WENT'
6136	042617	200	106	122	NOOP2:	.ASCIZ	<CRLF>'FROM STATE ' '
6137	042634	124	117	040	NOOP3:	.ASCIZ	'TO STATE ' '
6138	042646	200	111	116	NOOP4:	.ASCIZ	<CRLF>'INSTEAD OF ' '
6139	042663	200	124	110	NOOP5:	.ASCIZ	<CRLF>'THEREBY EXECUTING A ' '
6140	042711	011	040	111	NOOP6:	.ASCIZ	<TAB>' INSTEAD OF A ' '
6141	042731	011	040	111	NOOP7:	.ASCIZ	<TAB>' INSTRUCTION.'<CRLF>
6142	042751	040	040	124	NOOP10:	.ASCII	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'<TAB>
6143	043012	107	117	124		.ASCIZ	'GOT FPS. EXPECTED FPS.'<CRLF>
6144	043042	101	040	102	NOOP11:	.ASCIZ	'A BAD CONSTANT MAY HAVE BEEN USED.'<CRLF>
6145	043106	011	114	104	LFPS1:	.ASCIZ	<TAB>'LDFPS'<TAB>'REG'
6146	043121	011	114	104	LD1:	.ASCIZ	<TAB>'LDD'<TAB>'(REG),A'<TAB>'//FSRC#0//'
6147	043151	011	114	104	LD2:	.ASCIZ	<TAB>'LDD'<TAB>'A,A'
6148	043162	011	123	124	STFS1:	.ASCIZ	<TAB>'STFPS'<TAB>'REG'
6149	043175	011	123	124	ST1:	.ASCIZ	<TAB>'STD'<TAB>'A,(REG)'
6150	043212	011	123	124	ST2:	.ASCIZ	<TAB>'STD'<TAB>'A,A'
6151	043223	011	103	106	CFCC1:	.ASCIZ	<TAB>'CFCC'
6152	043231	011	123	105	SETF1:	.ASCIZ	<TAB>'SETF'
6153	043237	011	123	105	SETD1:	.ASCIZ	<TAB>'SETD'
6154	043245	011	123	105	SETI1:	.ASCIZ	<TAB>'SETI'
6155	043253	011	123	105	SETL1:	.ASCIZ	<TAB>'SETL'
6156	043261	011	111	114	ILL1:	.ASCIZ	<TAB>'ILLEGAL FPP INSTRUCTION'
6157	043312	011	123	124	STST1:	.ASCIZ	<TAB>'STST'<TAB>'REG'
6158	043324	011	111	114	ILL2:	.ASCIZ	<TAB>'ILLEGAL FPP INSTRUCTION (FID=1)'
6159	043365	040	040	124	ILLMS:	.ASCIZ	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF TRAP.'<TAB>'FPS.'<CRLF>
6160	043433	105	130	120	MS1:	.ASCIZ	'EXPECTED ' '
6161	043445	107	117	124	MS2:	.ASCIZ	'GOT ' '
6162	043452	103	117	116	MS3:	.ASCIZ	'CONTENTS OF LOCATIONS ' '
6163	043501	040	124	110	MS4:	.ASCIZ	' THROUGH ' '
6164	043513	106	101	111	MS5:	.ASCIZ	'FAILURE IN THE MICROPROGRAM FLOW.' '
6165	043555	103	117	116	MS6:	.ASCIZ	'CONTROL WENT ' '
6166	043573	106	122	117	MS7:	.ASCIZ	'FROM STATE ' '
6167	043607	040	124	117	MS10:	.ASCIZ	' TO STATE ' '
6168	043622	102	125	124	MS11:	.ASCIZ	'BUT SHOULD HAVE GONE ' '
6169	043647	103	117	116	MS12:	.ASCIZ	'CONTROL FLOW SHOULD HAVE GONE ' '
6170	043705	102	125	124	MS13:	.ASCIZ	'BUT DID NOT.' '
6171	043722	040	040	124	MS14:	.ASCII	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'<TAB>
6172	043763	107	117	124		.ASCIZ	'GOT PC.'<TAB>'EXPECTED PC.' '
6173	044010	111	116	123	MS15:	.ASCIZ	'INSTRUCTION TESTED: ' '
6174	044035	040	117	122	MS16:	.ASCIZ	' OR ' '
6175	044042	124	105	123	MS17:	.ASCIZ	'TESTING ACCUMULATOR ' '
6176	044067	132	105	122	MNUM0:	.ASCIZ	'ZERO ' '
6177	044075	117	116	105	MNUM1:	.ASCIZ	'ONE ' '
6178	044102	124	127	117	MNUM2:	.ASCIZ	'TWO ' '
6179	044107	124	110	122	MNUM3:	.ASCIZ	'THREE ' '
6180	044116	106	117	125	MNUM4:	.ASCIZ	'FOUR ' '



6181	044124	106	111	126	MNUM5:	.ASCIZ	'FIVE '
6182	044132	040	040	124	MS20:	.ASCIZ	' TEST.' <tab>'PC OF CALL.'<tab>'PC OF ERROR.'</tab></tab>
6183	044173	104	101	124	MS21:	.ASCIZ	'DATA (FLOATING POINT NUMBER): '
6184	044232	114	117	107	MS22:	.ASCIZ	'LOGICAL AND OF FAILING '
6185	044262	114	117	107	MS23:	.ASCIZ	'LOGICAL OR OF FAILING '
6186	044311	040	040	124	MS24:	.ASCII	' TEST.' <tab>'PC OF CALL.'<tab>'PC OF ERRORS.'<tab></tab></tab></tab>
6187	044353	116	125	115		.ASCIZ	'NUMBER OF ERRORS(OCTAL).'
6188	044404	105	130	120	MS25:	.ASCIZ	'EXPECTED DATA IN '
6189	044426	107	117	124	MS26:	.ASCIZ	'GOT DATA IN '
6190	044443	200	101	103	MS27:	.ASCIZ	<CRLF>'AC0= '
6191	044452	200	101	103	MS30:	.ASCIZ	<CRLF>'AC1= '
6192	044461	200	101	103	MS31:	.ASCIZ	<CRLF>'AC2= '
6193	044470	200	101	103	MS32:	.ASCIZ	<CRLF>'AC3= '
6194	044477	200	101	103	MS33:	.ASCIZ	<CRLF>'AC4= '
6195	044506	200	101	103	MS34:	.ASCIZ	<CRLF>'AC5= '
6196	044515	123	105	124	MS35:	.ASCIZ	'SET '
6197	044522	103	114	105	MS36:	.ASCIZ	'CLEAR '
6198	044531	114	117	101	MS37:	.ASCIZ	'LOADED DATA: '
6199	044547	122	105	101	MS40:	.ASCIZ	'READ DATA: '
6200	044563	105	130	120	MS415:	.ASCIZ	'EXPECTED DATA: '
6201	044603	104	101	124	MS41:	.ASCIZ	'DATA IN (RO) FSRC: '
6202	044627	104	101	124	MS42:	.ASCIZ	'DATA IN ACO: '
6203	044645	107	117	124	MS43:	.ASCIZ	'GOT RESULT: '
6204	044662	105	130	120	MS44:	.ASCIZ	'EXPECTED RESULT: '



Line	Address	PC	SP	BP	EM	Code	Description
6205						.SBTTL	ERROR MESSAGES
6206	044704	114	104	106	EM1:	.ASCIZ	'LDFPS AND STFPS TEST FAILED.'
6207	044741	114	104	106	EM2:	.ASCIZ	'LDFPS AND STFPS TEST ERROR SUMMARY.'
6208	045005	103	106	103	EM3:	.ASCIZ	'CFCC TRANSFERED BAD DATA TO THE PSW.'
6209	045052	103	106	103	EM4:	.ASCIZ	'CFCC MODIFIED THE FPS REGISTER.'
6210	045112	125	116	105	EM5:	.ASCIZ	'UNEXPECTED FPP TRAP TO 244.'
6211	045146	125	116	105	EM6:	.ASCIZ	'UNEXPECTED CPU TRAP TO 4.'
6212	045200	125	116	105	EM7:	.ASCIZ	'UNEXPECTED CPU TRAP TO 10.'
6213		045112			EM10=EM11:		
6214	045233	125	116	101	EM11:	.ASCIZ	'UNABLE TO DECODE FPP INSTRUCTION. TRAPPED TO 10.'
6215		000000			EM12=0		
6216		000000			EM13=0		
6217	045314	114	104	106	EM14:	.ASCIZ	'LDFPS R0 FAILED IN THE FSRC FLOWS.'
6218	045356	040	124	122		.ASCIZ	' TRAPPED TO 4.'
6219	045374	200	104	111		.ASCIZ	<CRLF>'DID NOT GO FROM STATE 400 TO 670.'
6220	045437	123	124	106	EM15:	.ASCIZ	'STFPS R1 FAILED IN THE FDST FLOWS.'
6221	045501	040	124	122		.ASCIZ	' TRAPPED TO 4.'
6222	045517	200	104	111		.ASCIZ	<CRLF>'DID NOT GO FROM STATE 634 TO 710.'
6223	045562	101	116	040	EM16:	.ASCIZ	'AN ILLEGAL FPP INSTRUCTION DID NOT TRAP.'
6224	045633	101	116	040	EM17:	.ASCIZ	'AN ILLEGAL FPP INSTRUCTION'
6225	045665	200	124	122		.ASCIZ	<CRLF>'TRAPPED TO 244, BUT FAILED TO SET '
6226	045730	124	110	105		.ASCIZ	'THE FPS CORRECTLY.'<CRLF>'EITHER A BAD CONSTANT '
6227	046001	127	101	123		.ASCIZ	'WAS GENERATED OR THE ALU LOGICAL OR FUNCTION FAILED.'
6228	046066	101	116	040	EM20:	.ASCIZ	'AN ILLEGAL FPP INSTRUCTION'
6229	046120	040	124	122		.ASCIZ	' TRAPPED TO 244, BUT A SUBSEQUENT '
6230	046162	040	123	124		.ASCIZ	' STST'<CRLF>
6231	046170	106	101	111		.ASCIZ	'FAILED TO PICK UP THE CORRECT FEC CODE = 2.'
6232	046244	123	124	123	EM21:	.ASCIZ	'STST R4 FAILED IN THE DESTINATION FLOWS.'
6233	046314	040	124	122		.ASCIZ	' TRAPPED TO 4.'<CRLF>
6234	046333	104	111	104		.ASCIZ	'DID NOT GO FROM STATE 636 TO 710.'
6235	046375	101	116	040	EM22:	.ASCIZ	'AN ILLEGAL FPP INSTRUCTION.'
6236	046430	127	111	124		.ASCIZ	'WITH INTERRUPTS DISABLED.'
6237		046375			EM23=EM22		
6238		046375			EM24=EM22		
6239	046462	123	117	125	EM25:	.ASCIZ	'SOURCE LOCATIONS MODIFIED BY, LDD.'
6240	046524	200	101	040		.ASCIZ	<CRLF>'A DATO WAS PERFORMED INSTEAD OF A DATI.'
6241	046575	114	104	104	EM26:	.ASCIZ	'LDD (R0),ACO FAILED.'<CRLF>
6242	046622	122	060	040		.ASCIZ	'R0 WAS MODIFIED.'
6243		046575			EM27=EM26		
6244	046643	124	110	105	EM30:	.ASCIZ	'THE PC WAS BAD AFTER '
6245	046671	101	116	040		.ASCIZ	'AN FPP INSTRUCTION.'
6246	046715	123	124	104	EM31:	.ASCIZ	'STD ACO,(R0) FAILED.'<CRLF>
6247	046742	122	060	040		.ASCIZ	'R0 WAS MODIFIED.'
6248		046715			EM32=EM31		
6249	046763	123	124	104	EM33:	.ASCIZ	'STD ACO,(R0) FAILED.'<CRLF>
6250	047010	117	125	124		.ASCIZ	'OUTPUT BAD.'
6251	047024	123	124	104	EM34:	.ASCIZ	'STD ACO,(R0) FAILED IN THE FDST FLOWS.'
6252	047072	200	124	110		.ASCIZ	<CRLF>'THE (BUT GR7) FORK FAILED.'
6253	047126	114	104	104	EM35:	.ASCIZ	'LDD (R0),ACO FAILED IN THE FSRC FLOWS.'
6254	047174	200	124	110		.ASCIZ	<CRLF>'THE (BUT GR7) FORK FAILED.'
6255	047230	123	124	104	EM36:	.ASCIZ	'STD ACO,(R0) FAILED IN THE FDST FLOWS.'
6256	047276	200	124	110		.ASCIZ	<CRLF>'THE (BUT FD) FORK FAILED.'
6257	047331	114	104	104	EM37:	.ASCIZ	'LDD (R0),ACO FAILED IN THE FSRC FLOWS.'
6258	047377	200	124	110		.ASCIZ	<CRLF>'THE (BUT FD) FORK FAILED.'
6259	047432	114	104	104	EM40:	.ASCIZ	'LDD (R0),ACO OR THE STD ACO,(R0) FAILED.'
6260	047502	200	102	101		.ASCIZ	<CRLF>'BAD DATA WAS DETECTED AFTER A SEQUENCE OF THE TWO INSTRUCTIONS.'
6261	047603	106	120	123	EM41:	.ASCIZ	'FPS BAD AFTER EXECUTION OF: '



6266	047640				EM42:	.ASCII	/LDD (RO),ACO FAILED IN THE FSRC FLOWS./<CRLF>
	047640	114	104	104		.ASCIZ	/THE (BUT FSRC) FORK FAILED. TRAPPED TO 4./
	047707	124	110	105			
6267	047761				EM43:	.ASCII	/STD ACO,(RO) FAILED IN THE FDST FLOWS./<CRLF>
	047761	123	124	104		.ASCIZ	/THE (BUT FDST) FORK FAILED. TRAPPED TO 4./
	050030	124	110	105			
6268	050102	106	120	120	EM44:	.ASCIZ	'FPP ACCUMULATORS DATA TEST FAILED.'
6269		050102			EM45=EM44		
6270	050145	106	120	120	EM46:	.ASCIZ	'FPP ACCUMULATORS DUAL ADDRESSING TEST FAILED.'
6275	050223				EM47:		
	050223	114	104	040		.ASCII	/LD AC1,ACO FAILED IN THE FSRC FLOWS./
	050267	124	110	105		.ASCIZ	/THE (BUT FSRC) FORK FAILED. TRAPPED TO 4./
6276	050341	114	104	040	EM50:	.ASCII	'LD AC1,ACO FAILED IN THE FSRC FLOWS.'
6277	050405	124	110	105		.ASCIZ	'THE (BUT FD) FORK FAILED.'
6278	050437	114	104	040	EM51:	.ASCIZ	'LD AC1,ACO TRANSFERRED BAD DATA.'
6288	050500				EM52:		
	050500	114	104	104		.ASCII	/LDD (RO)+,ACO FAILED IN THE FSRC FLOWS./
	050547	124	110	105		.ASCIZ	/THE (BUT FSRC) FORK FAILED. TRAPPED TO 4./
6289	050621				EM53:		
	050621	114	104	104		.ASCII	/LDD (RO)+,ACO FAILED IN THE FSRC FLOWS./
	050670	200	122	060		.ASCII	<CRLF>'RO WAS BAD.'<CRLF>
	050705	105	111	124		.ASCII	'EITHER A BAD CONSTANT WAS GENERATED OR'<CRLF>
	050754	104	111	104		.ASCIZ	\DID NOT GO FROM STATE 627 TO 322.\
6290	051016				EM54:		
	051016	114	104	104		.ASCIZ	/LDD (RO)+,ACO TRANSFERRED BAD DATA./
6291	051062				EM55:		
	051062	114	104	104		.ASCII	/LDD -(RO),ACO FAILED IN THE FSRC FLOWS./
	051131	124	110	105		.ASCIZ	/THE (BUT FSRC) FORK FAILED. TRAPPED TO 4./
6292	051203				EM56:		
	051203	114	104	104		.ASCII	/LDD -(RO),ACO FAILED IN THE FSRC FLOWS./
	051252	200	122	060		.ASCII	<CRLF>'RO WAS BAD.'<CRLF>
	051267	105	111	124		.ASCII	'EITHER A BAD CONSTANT WAS GENERATED OR'<CRLF>
	051336	104	111	104		.ASCIZ	\DID NOT GO FROM STATE 627 TO 324.\
6293	051400				EM57:		
	051400	114	104	104		.ASCIZ	/LDD -(RO),ACO TRANSFERRED BAD DATA./
6294	051444				EM60:		
	051444	114	104	106		.ASCII	/LDF (RO)+,ACO FAILED IN THE FSRC FLOWS./
	051513	200	122	060		.ASCII	<CRLF>'RO WAS BAD.'<CRLF>
	051530	105	111	124		.ASCII	'EITHER A BAD CONSTANT WAS GENERATED OR'<CRLF>
	051577	104	111	104		.ASCIZ	\DID NOT GO FROM STATE 627 TO 322.\
6295	051641				EM61:		
	051641	114	104	106		.ASCIZ	/LDF (RO)+,ACO TRANSFERRED BAD DATA./
6296	051705	114	104	106	EM62:	.ASCII	'LDF (RO)+,ACO FAILED IN THE FSRC FLOWS.'
6297	051754	200	124	110		.ASCII	<CRLF>'THE (BUT FD) FORK FAILED.'<CRLF>
6298	052007	127	105	116		.ASCII	'WENT FROM STATE 441 TO 077.'<CRLF>
6299	052043	111	116	123		.ASCIZ	'INSTEAD OF FROM 441 TO 076.'
6300	052077	114	104	104	EM63:	.ASCII	'LDD #NUM,ACO FAILED IN THE FSRC FLOWS.'
6301	052145	200	124	110		.ASCII	<CRLF>'THE (BUT GR7) FORK FAILED.'<CRLF>
6302	052201	127	105	116		.ASCII	'WENT FROM STATE 207 TO 174.'<CRLF>
6303	052235	111	116	123		.ASCIZ	'INSTEAD OF FROM 207 TO 176.'
6304	052271	114	104	104	EM64:	.ASCII	'LDD #NUM,ACO FAILED IN THE FSRC FLOWS.'
6305	052337	200	101	040		.ASCIZ	<CRLF>'A BAD CONSTANT WAS USED WHEN THE PC WAS INCREMENTED.'
6306		052271			EM65=EM64		
6307	052425				EM66:		
	052425	114	104	104		.ASCIZ	/LDD #NUM,ACO TRANSFERRED BAD DATA./
6329	052470				EM67:		
	052470	114	104	104		.ASCII	'LDD @ (RO)+,ACO FAILED IN THE FSRC FLOWS.'



	052540	200	124	110	.ASCII	<CRLF>'THE (BUT FSRC) FORK FAILED. TRAPPED TO 4.'
	052612	200	127	105	.ASCII	<CRLF>\WENT FROM STATE 627 TO EITHER 326 OR 326.\
	052664	200	111	116	.ASCIZ	<CRLF>\INSTEAD OF FROM 627 TO 323.\
6330	052721				EM70:	
	052721	114	104	104	.ASCII	'LDD @ (RO)+,ACO FAILED IN THE FSRC FLOWS.'
	052771	200	124	110	.ASCIZ	<CRLF>'THE (BUT FSRC) FORK FAILED. TRAPPED TO 4.'
6331	053044				EM71:	
	053044	114	104	104	.ASCII	'LDD @ (RO)+,ACO FAILED IN THE FSRC FLOWS.'
	053114	124	110	105	.ASCIZ	'THE (BUT FD) FORK FAILED.'
6332	053146				EM72:	
	053146	114	104	104	.ASCII	'LDD @ (RO)+,ACO'<CRLF>
	053165	106	101	111	.ASCIZ	'FAILED TO INCREMENT RO BY 2.'
6333	053222				EM73:	
	053222	114	104	104	.ASCIZ	'LDD @ (RO)+,ACO LOADED BAD DATA.'
6334	053262				EM74:	
	053262	114	104	104	.ASCII	'LDD @-(RO),ACO FAILED IN THE FSRC FLOWS.'
	053332	200	124	110	.ASCII	<CRLF>'THE (BUT FSRC) FORK FAILED. TRAPPED TO 4.'
	053404	200	127	105	.ASCII	<CRLF>\WENT FROM STATE 627 TO EITHER 326 OR 326.\
	053456	200	111	116	.ASCIZ	<CRLF>\INSTEAD OF FROM 627 TO 325.\
6335	053513				EM75:	
	053513	114	104	104	.ASCII	'LDD @-(RO),ACO FAILED IN THE FSRC FLOWS.'
	053563	200	124	110	.ASCIZ	<CRLF>'THE (BUT FSRC) FORK FAILED. TRAPPED TO 4.'
6336	053636				EM76:	
	053636	114	104	104	.ASCII	'LDD @-(RO),ACO FAILED IN THE FSRC FLOWS.'
	053706	124	110	105	.ASCIZ	'THE (BUT FD) FORK FAILED.'
6337	053740				EM77:	
	053740	114	104	104	.ASCII	'LDD @-(RO),ACO'<CRLF>
	053757	106	101	111	.ASCIZ	'FAILED TO DECREMENT RO BY 2.'
6338	054014				EM100:	
	054014	114	104	104	.ASCIZ	'LDD @-(RO),ACO LOADED BAD DATA.'
6339	054054				EM101:	
	054054	114	104	104	.ASCII	'LDD NUM(RO),ACO FAILED IN THE FSRC FLOWS.'
	054125	200	124	110	.ASCIZ	<CRLF>'THE (BUT FSRC) FORK FAILED. TRAPPED TO 4.'
6340	054200				EM102:	
	054200	114	104	104	.ASCII	'LDD NUM(RO),ACO'<CRLF>
	054220	106	101	111	.ASCIZ	'FAILED TO AFFECT RO BY 2.'
6341	054252				EM103:	
	054252	114	104	104	.ASCII	'LDD NUM(RO),ACO FAILED IN THE FSRC FLOWS.'
	054323	124	110	105	.ASCIZ	'THE (BUT FD) FORK FAILED.'
6342	054355				EM104:	
	054355	114	104	104	.ASCIZ	'LDD NUM(RO),ACO LOADED BAD DATA.'
6343	054416				EM105:	
	054416	114	104	104	.ASCII	'LDD @NUM(RO),ACO FAILED IN THE FSRC FLOWS.'
	054470	200	124	110	.ASCIZ	<CRLF>'THE (BUT FSRC) FORK FAILED. TRAPPED TO 4.'
6344	054543				EM106:	
	054543	114	104	104	.ASCII	'LDD @NUM(RO),ACO'<CRLF>
	054564	106	101	111	.ASCIZ	'FAILED TO AFFECT RO BY 2.'
6345	054616				EM107:	
	054616	114	104	104	.ASCII	'LDD @NUM(RO),ACO FAILED IN THE FSRC FLOWS.'
	054670	124	110	105	.ASCIZ	'THE (BUT FD) FORK FAILED.'
6346	054722				EM110:	
	054722	114	104	104	.ASCIZ	'LDD @NUM(RO),ACO LOADED BAD DATA.'
6362	054764				EM111:	
	054764	114	104	104	.ASCII	/LDD AC7,ACO FAILED TO TRAP TO 244./
	055026	200	101	103	.ASCIZ	<CRLF>/AC7 IS AN ILLEGAL ACCUMULATOR./
6363		054764			EM112=EM111	
6364	055066				EM113:	



	055066	114	104	104	.ASCII	/LDD AC6,ACO FAILED TO TRAP TO 244./
	055130	200	101	103	.ASCIZ	<CRLF>/AC6 IS AN ILLEGAL ACCUMULATOR./
6365	055066				EM114=EM113	
6366	054764				EM115=EM111	
6367	055066				EM116=EM113	
6368	055170				EM117:	
	055170	125	123	105	.ASCII	'USE OF AN ILLEGAL ACCUMULATOR WITH FSRC MODE ZERO.'
	055252	200	124	122	.ASCIZ	<CRLF>'TRAPPED BUT FAILED TO SET FPS CORRECTLY.'
6369	055324				EM120:	
	055324	125	123	105	.ASCII	'USE OF AN ILLEGAL ACCUMULATOR WITH FSRC MODE ZERO.'
	055406	200	124	122	.ASCIZ	<CRLF>'TRAPPED BUT FAILED TO SET FEC CORRECTLY.'
6370	055460	123	124	040	EM121:	.ASCII 'ST ACO,AC1 FAILED IN THE FDST FLOWS.'
6371	055524	200	124	110	.ASCIZ	<CRLF>'THE (BUT FDST) FORK FAILED. TRAPPED TO 4.'
6372	055577	123	124	040	EM122:	.ASCII 'ST ACO,AC1 FAILED IN THE FDST FLOWS.'
6373	055643	200	124	110	.ASCIZ	<CRLF>'THE (BUT FD) FORK FAILED.'
6374	055676	123	124	040	EM123:	.ASCIZ 'ST ACO,AC1 TRANSFERRED BAD DATA.'
6375	055737				EM124:	
	055737	106	120	123	.ASCII	'FPS BAD AFTER LDD (R0),ACO.'
	055772	200	124	110	.ASCIZ	<CRLF>\THE (BUT EZBT Y8) FORK FAILED.\
6376	056032				EM125:	
	056032	106	120	123	.ASCII	'FPS BAD AFTER LDD (R0),ACO.'
	056065	200	124	110	.ASCIZ	<CRLF>\THE (BUT ENBT) FORK FAILED.\
6377	056122	114	104	104	EM126:	.ASCII 'LDD (R0),ACO TRAPPED TO 244.'
6378	056156	040	106	123	.ASCII	' FSRC= -0 AND FIUV= 0.'<CRLF>
6379	056205	124	110	105	.ASCII	'THE (BUT FIUV) FORK FAILED.'
6380	056240	200	127	105	.ASCII	<CRLF>'WENT FROM STATE 256 TO 354.'
6381	056274	200	111	116	.ASCIZ	<CRLF>'INSTEAD OF FROM 256 TO 254.'
6382	056331	114	104	104	EM127:	.ASCII 'LDD (R0),ACO FAILED TO TRAP TO 244.'
6383	056374	040	106	123	.ASCII	' FSRC= -0, FIUV= 1.'
6384	056417	200	124	110	.ASCII	<CRLF>'THE (BUT FIUV) FORK FAILED.'<CRLF>
6385	056453	127	105	116	.ASCII	'WENT FROM STATE 256 TO 254.'
6386	056506	200	111	116	.ASCIZ	<CRLF>'INSTEAD OF FROM 256 THE 354.'
6387	056544	114	104	104	EM130:	.ASCII 'LDD (R0),ACO TRAPPED TO 244.'
6388	056600	106	123	122	.ASCII	'FSRC= -0, FIUV= 1.'<CRLF>
6389	056623	102	125	124	.ASCIZ	'BUT FEC WAS BAD.'
6390	056644				EM131:	
	056644	114	104	103	.ASCIZ	/LDCFD (R0),ACO LOADED BAD DATA./
6391	056704				EM132:	
	056704	114	104	103	.ASCIZ	/LDCDF (R0),ACO LOADED BAD DATA./
6432	056744				EM133:	
	056744	101	104	104	.ASCIZ	/ADDD (R0),ACO WITH (R0)=ACO=0 /
6433	057003				EM134:	
	057003	101	104	104	.ASCIZ	/ADDF (R0),ACO WITH (R0)=ACO=0 /
6434	057042				EM135:	
	057042	123	125	102	.ASCIZ	/SUBD (R0),ACO WITH (R0)=ACO=0 /
6435	057101				EM136:	
	057101	123	125	102	.ASCIZ	/SUBF (R0),ACO WITH (R0)=ACO=0 /
6436	056744				EM137=EM133	
6437	057003				EM140=EM134	
6438	057042				EM141=EM135	
6439	057101				EM142=EM136	
6440	057140				EM143:	
	057140	101	104	104	.ASCIZ	/ADDD (R0),ACO WITH (R0)=0 /
6441	057173				EM144:	
	057173	123	125	102	.ASCIZ	/SUBD (R0),ACO WITH (R0)=0 /
6442	057140				EM145=EM143	
6443	057173				EM146=EM144	



6444	057226				EM147:	
	057226	123	125	102	.ASCIZ	/SUBD (R0),ACO WITH ACO=0 /
6445		057226			EM150=EM147	
6446		057226			EM151=EM147	
6447	057260				EM152:	
	057260	101	104	104	.ASCIZ	/ADDD (R0),ACO WITH ACO=0 /
6448		057260			EM153=EM152	
6449	057312				EM154:	
	057312	101	116	040	.ASCII	'AN OVERFLOW ERROR OCCURRED ON ADD'<CRLF>
	057354	103	101	125	.ASCII	'CAUSING A TRAP TO 244.'
	057402	200	050	102	.ASCII	<CRLF>'(BUT EZBT Y9 Y8) FORK IN STATE 420 OF OVER\UNDER FAILED.'
	057473	200	123	110	.ASCIZ	<CRLF>'SHOULD HAVE GONE FROM STATE 420 TO 131.'
6450	057544				EM155:	
	057544	101	116	040	.ASCII	'AN UNDERFLOW ERROR OCCURRED ON ADD'<CRLF>
	057607	103	101	125	.ASCII	'CAUSING A TRAP TO 244.'
	057635	200	050	102	.ASCII	<CRLF>'(BUT EZBT Y9 Y8) FORK IN STATE 420 OF OVER\UNDER FAILED.'
	057726	200	123	110	.ASCIZ	<CRLF>'SHOULD HAVE GONE FROM STATE 420 TO 131.'
6451	057777				EM156:	
	057777	101	104	104	.ASCII	/ADDD (R0),ACO FAILED IN THE ROUND\TRUNK FLOWS./
	060055	200	124	110	.ASCII	<CRLF>'THE (BUT FD) FORK FAILED. WENT'
	060114	106	122	117	.ASCII	\FROM STATE 665 TO 113.\<CRLF>
	060143	111	116	123	.ASCIZ	\INSTEAD OF FROM 665 TO 313.\<CRLF>\WITH FT SET.\
6452	060214				EM157:	
	060214	101	104	104	.ASCII	/ADDD (R0),ACO FAILED IN THE ROUND\TRUNK FLOWS./
	060272	200	124	110	.ASCII	<CRLF>'THE (BUT FD) FORK FAILE'. WENT'
	060331	106	122	117	.ASCII	\FROM STATE 665 TO 313.\<CRLF>
	060360	111	116	123	.ASCIZ	\INSTEAD OF FROM 665 TO 113.\<CRLF>\WITH FT CLEAR.\
6453	060433				EM160:	
	060433	101	104	104	.ASCII	/ADDD (R0),ACO FAILED IN THE ROUND\TRUNK FLOWS./<CRLF>
	060512	124	110	105	.ASCII	'THE FLOATING CONSTANT WAS USED INSTEAD OF THE DOUBLE CONSTANT'<CRLF>
	060610	111	116	040	.ASCIZ	'IN THE ROUND ALGORITHM.'
6454	060640				EM161:	
	060640	101	104	104	.ASCII	/ADDF (R0),ACO FAILED IN THE ROUND\TRUNK FLOWS./<CRLF>
	060717	124	110	105	.ASCII	'THE DOUBLE CONSTANT WAS USED INSTEAD OF THE FLOATING CONSTANT'<CRLF>
	061015	111	116	040	.ASCIZ	'IN THE ROUND ALGORITHM.'
6455	061045				EM162:	
	061045	101	104	104	.ASCIZ	/ADDD (R0),ACO PRODUCED A BAD RESULT./
6456	061112				EM163:	
	061112	101	104	104	.ASCIZ	/ADDF (R0),ACO PRODUCED A BAD RESULT./
6457	061157				EM164:	
	061157	124	110	105	.ASCIZ	\THE FPS WAS BAD AFTER ADDD (R0),ACO.\
6458	061224				EM165:	
	061224	124	110	105	.ASCIZ	\THE FPS WAS BAD AFTER ADDF (R0),ACO.\
6459	061271				EM166:	
	061271	101	104	104	.ASCII	/ADDD (R0),ACO PRODUCED A BAD RESULT./<CRLF>
	061336	120	122	117	.ASCIZ	'PROBABLE ERROR IN THE ALIGN FLOWS.'
6460	061401				EM167:	
	061401	101	104	104	.ASCII	/ADDD (R0),ACO FAILED IN THE ALIGN FLOWS./<CRLF>
	061452	106	114	117	.ASCII	\FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 111, TO 014.\
	061542	200	101	040	.ASCII	<CRLF>\A BAD CONSTANT (NOT 57 DEC) \
	061577	127	101	123	.ASCIZ	'WAS USED IN THE ALIGN ALGORITHM.'
6461	061640				EM170:	
	061640	101	104	104	.ASCII	/ADDF (R0),ACO PRODUCED A BAD RESULT./<CRLF>
	061705	120	122	117	.ASCIZ	'PROBABLE ERROR IN THE ALIGN FLOWS.'
6462	061750				EM171:	
	061750	101	104	104	.ASCII	/ADDF (R0),ACO FAILED IN THE ALIGN FLOWS./<CRLF>
	062021	106	114	117	.ASCII	\FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 111, TO 014.\



	062111	200	101	040	.ASCII	<CRLF>\A BAD CONSTANT (NOT 25 DEC) \
	062146	127	101	123	.ASCIZ	'WAS USED IN THE ALIGN ALGORITHM.'
6463	062207				EM172:	
	062207	101	104	104	.ASCII	/ADDD (RO),ACO FAILED IN THE ALIGN FLOWS./<CRLF>
	062260	106	114	117	.ASCII	\FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 011, TO 015.\
	062350	200	101	040	.ASCII	<CRLF>\A BAD CONSTANT (NOT 57 DEC) \
	062405	127	101	123	.ASCIZ	'WAS USED IN THE ALIGN ALGORITHM.'
6464	062446				EM173:	
	062446	101	104	104	.ASCII	/ADDD (RO),ACO FAILED IN THE ALIGN FLOWS./<CRLF>
	062517	106	114	117	.ASCII	\FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 011, TO 215.\
	062607	200	101	040	.ASCII	<CRLF>\A BAD CONSTANT (NOT 57 DEC) \
	062644	127	101	123	.ASCIZ	'WAS USED IN THE ALIGN ALGORITHM.'
6465	062705				EM174:	
	062705	101	104	104	.ASCII	/ADDF (RO),ACO FAILED IN THE ALIGN FLOWS./<CRLF>
	062756	106	114	117	.ASCII	\FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 011, TO 015.\
	063046	200	101	040	.ASCII	<CRLF>\A BAD CONSTANT (NOT 25 DEC) \
	063103	127	101	123	.ASCIZ	'WAS USED IN THE ALIGN ALGORITHM.'
6466	063144				EM175:	
	063144	101	104	104	.ASCII	/ADDF (RO),ACO FAILED IN THE ALIGN FLOWS./<CRLF>
	063215	106	114	117	.ASCII	\FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 011, TO 215.\
	063305	200	101	040	.ASCII	<CRLF>\A BAD CONSTANT (NOT 25 DEC) \
	063342	127	101	123	.ASCIZ	'WAS USED IN THE ALIGN ALGORITHM.'
6467	063403				EM176:	
	063403	101	104	104	.ASCII	'ADDD (RO),ACO FAILED IN THE ADD-SUB FLOWS.'<CRLF>
	063456	104	111	104	.ASCIZ	\DID NOT TAKE THE PATH: STATE 216, TO 442, TO 500.\
6468	063540	120	117	127	EM177:	'POWER MONITOR BIT FOUND SET'
6469	063574				EM200:	
	063574	101	104	104	.ASCII	'ADDD (RO),ACO FAILED IN THE ADD-SUB FLOWS.'<CRLF>
	063647	104	111	104	.ASCIZ	\DID NOT TAKE THE PATH: STATE 216, TO 440, TO 121.\
6470	063731				EM201:	
	063731	101	104	104	.ASCII	'ADDD (RO),ACO FAILED IN THE ADD-SUB FLOWS.'<CRLF>
	064004	104	111	104	.ASCIZ	\DID NOT TAKE THE PATH: STATE 216, TO 440, TO 101.\
6471	064066				EM202:	
	064066	101	104	104	.ASCII	'ADDD (RO),ACO FAILED IN THE ADD-SUB FLOWS.'<CRLF>
	064141	104	111	104	.ASCIZ	\DID NOT TAKE THE PATH: STATE 216, TO 042, TO 101.\
6472	064223				EM203:	
	064223	101	104	104	.ASCII	'ADDD (RO),ACO FAILED IN THE ADD-SUB FLOWS.'<CRLF>
	064276	104	111	104	.ASCIZ	\DID NOT TAKE THE PATH: STATE 216, TO 440, TO 141.\
6473	064360				EM204:	
	064360	101	104	104	.ASCII	'ADDD (RO),ACO FAILED IN THE ADD-SUB FLOWS.'<CRLF>
	064433	104	111	104	.ASCIZ	\DID NOT TAKE THE PATH: STATE 216, TO 042, TO 141.\
6474	064515				EM205:	
	064515	124	110	105	.ASCIZ	\THE FPS WAS BAD AFTER SUBD (RO),ACO.\
6475	064562				EM206:	
	064562	123	125	102	.ASCIZ	/SUBD (RO),ACO PRODUCED A BAD RESULT./
6476	064627	123	125	102	EM207:	'SUBD (RO),ACO PRODUCED A BAD RESULT.'
6477	064673	200	124	110	.ASCIZ	<CRLF>'THE XOR OF THE SIGN BIT FAILED IN STATE 024.'
6478	064751	101	104	104	EM210:	'ADDD (RO),ACO FAILED IN THE NORMALIZE FLOWS.'
6479		045112			EM211=EM5	
6480		045146			EM212=EM6	
6481		045200			EM213=EM7	
6482	065026				EM214:	



065026	101	104	104		.ASCII	'ADD (RO),ACO FAILED IN THE ADD-SUB FLOWS.'
065101	104	111	104		.ASCIZ	\DID NOT TAKE THE PATH: STATE 216, TO 042, TO 121.\
6483					.SBTTL	DATA HEADERS
6484 065163	040	040	124	DH1:	.ASCII	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6485 065223	011	127	122		.ASCIZ	<TAB>'WROTE.'<TAB>'READ.'<TAB>'EXPECTED.'
6486 065253	040	040	124	DH2:	.ASCII	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6487 065313	101	116	104		.ASCIZ	'AND BAD DATA.'<TAB>'OR BAD DATA.'
6488 065346	040	040	124	DH3:	.ASCII	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6489 065406	011	122	105		.ASCIZ	<TAB>'READ PSW.'<TAB>'EXPECTED PSW.'
6490 065437	040	040	124	DH4:	.ASCII	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6491 065477	011	127	122		.ASCIZ	<TAB>'WROTE FPS.'<TAB>'FPS AFTER CFCC.'
6492 065533	040	040	124	DH5:	.ASCIZ	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF TRAP.'
6493	065533					DH6=DH5
6494	065533					DH7=DH5
6495	065533					DH10=DH5
6496	065533					DH11=DH5
6497	000000					DH12=0
6498	000000					DH13=0
6499	065533					DH14=DH5
6500	065533					DH15=DH5
6501 065573	040	040	124	DH16:	.ASCII	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6502 065633	011	117	120		.ASCIZ	<TAB>'OP CODE. FPS.'
6503 065653	040	040	124	DH17:	.ASCII	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6504 065713	011	107	117		.ASCIZ	<TAB>'GOT FPS.'<TAB>'EXPECTED FPS.'
6505 065743	040	040	124	DH20:	.ASCII	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF TRAP.'
6506 066002	011	120	103		.ASCIZ	<TAB>'PC OF STST.'<TAB>'READ FEC.'
6507	065533					DH21=DH5
6508 066031	106	101	111	DH22:	.ASCIZ	'FAILED TO CORRECTLY SET FPS.'
6509 066066	106	101	111	DH23:	.ASCII	'FAILED TO CORRECTLY SET FEC TO 000002.'
6510 066135	040	040	124		.ASCII	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6511 066175	011	120	103		.ASCIZ	<TAB>'PC OF STST.'<TAB>'READ FEC.'
6512 066224	124	122	101	DH24:	.ASCII	'TRAPPED TO 244. FLOW WENT FROM STATE 554 TO STATE 430.'
6513 066312	200	111	116		.ASCIZ	<CRLF>'INSTEAD OF FROM STATE 554 TO STATE 432.'
6514 066363	040	040	124	DH25:	.ASCIZ	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'<TAB>
6515 066425	040	040	124	DH26:	.ASCII	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6516 066465	011	107	117		.ASCIZ	<TAB>'GOT RO.'<TAB>'EXPECTED RO.'
6517	066425					DH27=DH26
6518	000000					DH30=0
6519	066425					DH31=DH26
6520	066425					DH32=DH26
6521 066513	040	040	124	DH33:	.ASCII	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6522 066553	011	122	060		.ASCIZ	<TAB>'RO (TARGET LOCATIONS FOR OUTPUT).'
6523	066513					DH34=DH33
6524	066513					DH35=DH33
6525	066513					DH36=DH33
6526	066513					DH37=DH33
6527	066513					DH40=DH33
6528	000000					DH41=0
6529 066616	040	040	124	DH42:	.ASCII	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF TRAP.'
6530 066655	011	122	060		.ASCIZ	<TAB>'RO (TARGET LOCATIONS FOR OUTPUT).'
6531	066616					DH43=DH42
6532	000000					DH44=0
6533 066720	105	122	122	DH45:	.ASCIZ	'ERROR SUMMARY.'
6534 066737	040	040	124	DH46:	.ASCIZ	' TEST.'<TAB>'CALL AT PC.'
6535	066720					DH47=DH45
6536 066763	040	040	124	DH50:	.ASCII	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6537 067023	011	127	111		.ASCIZ	<TAB>'WITH FD.'



6538		066763				DH51=DH50	
6539		065533				DH52=DH5	
6540		066425				DH53=DH26	
6541	067035	040	040	124	DH54:	.ASCIZ ' TEST.' <tab>'PC OF CALL.'<tab>'PC OF ERROR.'</tab></tab>	
6542		065533			DH55=DH5		
6543		066425			DH56=DH26		
6544		067035			DH57=DH54		
6545		066425			DH60=DH26		
6546		067035			DH61=DH54		
6547		067035			DH62=DH54		
6548		067035			DH63=DH54		
6549	067076	122	105	123	DH65:	.ASCII 'RESULTING IN AN ODD ADDRESS TRAP TO 4.'	
6550	067144	200				<CRLF>	
6551	067145	040	040	124	DH64:	.ASCII ' TEST.' <tab>'PC OF CALL.'<tab>'PC OF ERROR.'</tab></tab>	
6552	067205	011	107	117		.ASCIZ <TAB>'GOT PC.' <tab&gt;'expected pc.'<="" td=""></tab&gt;'expected>	
6553		067035			DH66=DH54		
6554		065533			DH67=DH5		
6555		065533			DH70=DH5		
6556		066363			DH71=DH25		
6557		066425			DH72=DH26		
6558		067035			DH73=DH54		
6559		065533			DH74=DH5		
6560		065533			DH75=DH5		
6561		066363			DH76=DH25		
6562		066425			DH77=DH26		
6563		067035			DH100=DH54		
6564		065533			DH101=DH5		
6565		066425			DH102=DH26		
6566		066363			DH103=DH25		
6567		067035			DH104=DH54		
6568		065533			DH105=DH5		
6569		066425			DH106=DH26		
6570		066363			DH107=DH25		
6571		067035			DH110=DH54		
6572		066363			DH111=DH25		
6573	067233	124	110	105	DH112:	.ASCII 'THE (BUT FSRC) FORK FAILED.' <crlf&gt;< td=""></crlf&gt;<>	
6574	067267	103	117	116		.ASCII 'CONTROL WENT FROM STATE 762 TO STATE 627.'	
6575	067340	200	111	116		.ASCII <CRLF>'INSTEAD OF FROM STATE 762 TO STATE 627.' <crlf&gt;< td=""></crlf&gt;<>	
6576	067411	040	040	124		.ASCIZ ' TEST.' <tab>'PC OF CALL.'<tab>'PC OF ERROR.'</tab></tab>	
6577		066363			DH113=DH25		
6578		067233			DH114=DH112		
6579	067452	124	110	105	DH115:	.ASCII 'THE (BUT FSRC) FORK FAILED RESULTING IN AN ODD ADDRESS TRAP TO 4.'	
6580	067553	200	103	117		.ASCII <CRLF>'CONTROL WENT FROM STATE 762 TO STATE 627.' <crlf&gt;< td=""></crlf&gt;<>	
6581	067626	111	116	123		.ASCII 'INSTEAD OF FROM STATE 762 TO STATE 627.' <crlf&gt;< td=""></crlf&gt;<>	
6582	067676	040	040	124		.ASCIZ ' TEST.' <tab>'PC OF CALL.'<tab>'PC OF TRAP.'</tab></tab>	
6583		067452			DH116=DH115		
6584		065653			DH117=DH17		
6585	067736	040	040	124	DH120:	.ASCII ' TEST.' <tab>'PC OF CALL.'<tab>'PC OF ERROR.'</tab></tab>	
6586	067776	011	107	117		.ASCIZ <TAB>'GOT FEC.' <tab&gt;'expected fec.'<="" td=""></tab&gt;'expected>	
6587		065533			DH121=DH5		
6588		066763			DH122=DH50		
6589		066763			DH123=DH50		
6590		065653			DH124=DH17		
6591		065653			DH125=DH17		
6592		065533			DH126=DH5		
6593		067035			DH127=DH54		
6594		067736			DH130=DH120		



6595		067035				DH131=DH54	
6596		067035				DH132=DH54	
6597	070026	106	101	111		DH133: .ASCII	'FAILED TO PRODUCE THE CORRECT RESULTS.' <crlf&gt;< td=""></crlf&gt;<>
6598	070075	040	040	124		.ASCIZ	' TEST.' <tab&gt;'pc call.'<tab&gt;'pc="" error.'<="" of="" td=""></tab&gt;'pc>
6599		070026				DH134=DH133	
6600		070026				DH135=DH133	
6601		070026				DH136=DH133	
6602	070136	120	122	117		DH137: .ASCII	'PRODUCED THE CORRECT RESULT BUT FAILED TO SET THE FPS CORRECTLY.'
6603	070236	040	040	124		.ASCII	' TEST.' <tab&gt;'pc call.'<tab&gt;'pc="" error.'<="" of="" td=""></tab&gt;'pc>
6604	070276	011	107	117		.ASCIZ	<TAB>'GOT FPS.' <tab&gt;'expected fps.'<="" td=""></tab&gt;'expected>
6605		070136				DH140=DH137	
6606		070136				DH141=DH137	
6607		070136				DH142=DH137	
6608		070026				DH143=DH133	
6609		070026				DH144=DH133	
6610		070136				DH145=DH137	
6611		070136				DH146=DH137	
6612		067035				DH147=DH54	
6613	070326	130	117	122		DH150: .ASCII	'XOR OF SIGN BIT FAILED.' <crlf&gt;< td=""></crlf&gt;<>
6614	070356	040	040	124		.ASCIZ	' TEST.' <tab&gt;'pc call.'<tab&gt;'pc="" error.'<="" of="" td=""></tab&gt;'pc>
6615		070136				DH151=DH137	
6616		070026				DH152=DH133	
6617		070136				DH153=DH137	
6618	070417	040	040	124		DH154: .ASCIZ	' TEST.' <tab&gt;'pc call.'<tab&gt;'pc="" of="" td="" trap.'<=""></tab&gt;'pc>
6619		070417				DH155=DH154	
6620		067035				DH156=DH54	
6621		067035				DH157=DH54	
6622		067035				DH160=DH54	
6623		067035				DH161=DH54	
6624		067035				DH162=DH54	
6625		067035				DH163=DH54	
6626		065653				DH164=DH17	
6627		065653				DH165=DH17	
6628		067035				DH166=DH54	
6629		067035				DH167=DH54	
6630		067035				DH170=DH54	
6631		067035				DH171=DH54	
6632		067035				DH172=DH54	
6633		067035				DH173=DH54	
6634		067035				DH174=DH54	
6635		067035				DH175=DH54	
6636		067035				DH176=DH54	
6637	070457	040	040	124		DH177: .ASCIZ	' TEST. ERR PC CPU ERROR REGISTER'
6638		067035				DH200=DH54	
6639		067035				DH201=DH54	
6640		067035				DH202=DH54	
6641		067035				DH203=DH54	
6642		067035				DH204=DH54	
6643		065653				DH205=DH17	
6644		067035				DH206=DH54	
6645		067035				DH207=DH54	
6646		067035				DH210=DH54	
6647	070525	040	040	124		DH211: .ASCIZ	' TEST.' <tab&gt;'pc call.'<tab&gt;'pc="" of="" td="" trap.'<tab&gt;'fec.'<=""></tab&gt;'pc>
6648		065533				DH212=DH5	
6649		065533				DH213=DH5	
6650		067035				DH214=DH54	



Line No.	Code	Mode	DF	Type	Format
6651				.SBTTL	DATA FORMATS
6652	070570	004	000	005 DF1:	.BYTE 4,0,5,0,5,0,0,0
6653	070600	004	000	005 DF2:	.BYTE 4,0,5,4,5,0,5,0
6654	070610	004	000	005 DF3:	.BYTE 4,0,5,0,5,0,5,0
6655		070610			DF4=DF3
6656	070620	004	000	005 DF5:	.BYTE 4,0,5,0,5,0,5,11,5,0,5,0
6657		070620			DF6=DF5
6658		070620			DF7=DF5
6659		070620			DF10=DF5
6660		070620			DF11=DF5
6661	070634	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
6662	070666	005	011	005 DF13:	.BYTE 5,11,5,5,5,4,0,5,0,5,0,0
6663		070620			DF14=DF6
6664		070620			DF15=DF6
6665	070702	004	000	005 DF16:	.BYTE 4,0,5,0,5,0
6666		070610			DF17=DF3
6667	070711	004	000	005 DF20:	.BYTE 4,0,5,0,5,0,5,0
6668	070721	004	000	005 DF21:	.BYTE 4,0,5,0
6669	070725	005	005	004 DF22:	.BYTE 5,5,4,0,5,0,5,0,5,0
6670	070737	004	000	005 DF23:	.BYTE 4,0,5,0,5,0,5,0
6671	070747	005	004	000 DF24:	.BYTE 5,4,0,5,0,5,0
6672	070756	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
6673	071002	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
6674	071027	004	000	005 DF27:	.BYTE 4,0,5,0,5,0,0,5,5,5,5,4,5,4,5,5
6675	071047	005	011	005 DF30:	.BYTE 5,11,5,5,5,4,0,5,0,5,0,0
6676		071002			DF31=DF26
6677		071027			DF32=DF27
6678	071063	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
6679	071111	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
6680		071111			DF35=DF34
6681		071111			DF36=DF34
6682		071111			DF37=DF34
6683	071135	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
6684	071163	011	005	005 DF41:	.BYTE 11,5,5,5,4,0,5,0,5,0,5,0
6685	071177	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
6686		071177			DF43=DF42
6687	071225	005	011	005 DF44:	.BYTE 5,11,5,5,5,4,0,5,0,5,5,5,5,3,5,5,5,5,3
6688	071250	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
6689	071275	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
6690	071327	004	000	005 DF47:	.BYTE 4,0,5,0,5,5,5,4,5,4,5,5
6691	071344	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
6692	071370	004	000	005 DF51:	.BYTE 4,0,5,0,5,11,5,5,5,3,5,5,5,3
6693		071327			DF52=DF47
6694	071406	004	000	005 DF53:	.BYTE 4,0,5,0,5,0,0
6695	071415	004	000	005 DF54:	.BYTE 4,0,5,0,5,5,5,5,3,5,5,5,5,3
6696		071327			DF55=DF47
6697		071406			DF56=DF53
6698		071415			DF57=DF54
6699		071406			DF60=DF53
6700		071415			DF61=DF54
6701		071415			DF62=DF54
6702		071415			DF63=DF54
6703	071433	004	000	005 DF64:	.BYTE 4,0,5,0,5,0,0
6704		071433			DF65=DF64
6705		071415			DF66=DF54
6706		070721			DF67=DF21
6707	071442	004	000	005 DF70:	.BYTE 4,0,5,0,5,5,5,5,4,5,4,5,5,5,5,4,5,4



6708		071442			DF71=DF70	
6709	071464	004	000	005	DF72: .BYTE	4.0.5.0.5.0.0
6710		071415			DF73=DF54	
6711		070721			DF74=DF21	
6712		071442			DF75=DF70	
6713		071442			DF76=DF70	
6714		071464			DF77=DF72	
6715		071415			DF100=DF54	
6716		071442			DF101=DF70	
6717		071464			DF102=DF72	
6718		071442			DF103=DF70	
6719		071415			DF104=DF54	
6720		071442			DF105=DF70	
6721		071464			DF106=DF72	
6722		071442			DF107=DF70	
6723		071415			DF110=DF54	
6724	071473	004	000	005	DF111: .BYTE	4.0.5.0
6725		071473			DF112=DF111	
6726		071473			DF113=DF111	
6727		071473			DF114=DF111	
6728		071473			DF115=DF111	
6729		071473			DF116=DF111	
6730		070610			DF117=DF3	
6731		070610			DF120=DF3	
6732		071327			DF121=DF47	
6733		071344			DF122=DF50	
6734		071370			DF123=DF51	
6735	071477	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
6736		071477			DF125=DF124	
6737		071473			DF126=DF111	
6738		071473			DF127=DF111	
6739		070610			DF130=DF3	
6740	071530	004	000	005	DF131: .BYTE	4.0.5.0.5.5.5.3.5.5.5.3.5.5.5.3
6741		071530			DF132=DF131	
6742	071550	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
6743		071550			DF134=DF133	
6744		071550			DF135=DF133	
6745		071550			DF136=DF133	
6746	071574	004	000	005	DF137: .BYTE	4.0.5.0.5.0.5.0
6747		071574			DF140=DF137	
6748		071574			DF141=DF137	
6749		071574			DF142=DF137	
6750		071550			DF143=DF133	
6751		071550			DF144=DF133	
6752		071574			DF145=DF137	
6753		071574			DF146=DF137	
6754		071550			DF147=DF133	
6755		071550			DF150=DF133	
6756		071574			DF151=DF137	
6757		071550			DF152=DF133	
6758		071574			DF153=DF137	
6759	071604	004	000	005	DF154: .BYTE	4.0.5.0
6760		071604			DF155=DF154	
6761		071550			DF156=DF133	
6762		071550			DF157=DF133	
6763		071550			DF160=DF133	
6764	071610	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



6765	071550			DF162=DF133	
6766	071610			DF163=DF161	
6767	070610			DF164=DF3	
6768	070610			DF165=DF3	
6769	071550			DF166=DF133	
6770	071550			DF167=DF133	
6771	071610			DF170=DF161	
6772	071610			DF171=DF161	
6773	071550			DF172=DF133	
6774	071550			DF173=DF133	
6775	071610			DF174=DF161	
6776	071610			DF175=DF161	
6777	071550			DF176=DF133	
6778	071634	000	000	DF177: .BYTE	4.0.0
6779	071550			DF200=DF133	
6780	071550			DF201=DF133	
6781	071550			DF202=DF133	
6782	071550			DF203=DF133	
6783	071550			DF204=DF133	
6784	070610			DF205=DF3	
6785	071550			DF206=DF133	
6786	071550			DF207=DF133	
6787	071550			DF210=DF133	
6788	071637	000	005	DF211: .BYTE	4.0.5.0.5.0
6789	071637			DF212=DF211	
6790	071637			DF213=DF211	
6791	071550			DF214=DF133	
6792				.EVEN	



6793						.SBTTL	DATA TABLES
6794	071646	001232	001234	042272	DT1:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
6795	071662	001242	001244	000000		.WORD	\$TMP4,\$TMP5,0
6796	071670	001232	001234	042272	DT2:	.WORD	\$TMP0,\$TMP1,\$TAB,AERFLG,\$TAB,\$TMP2,\$TAB,\$TMP3,0
6797	071712	001232	001234	042272	DT3:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
6798	071726	042272	001242	000000		.WORD	\$TAB,\$TMP4,0
6799		071712			DT4=DT3		
6800	071734	001232	001234	042272	DT5:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,LF IEX1,\$TMP21,LF IEX2
6801	071752	001272	042411	001240		.WORD	\$TMP20,FPSMS,\$TMP3,FECMS,\$TMP4,0
6802	071766	001232	001234	042272	DT6:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,LF IEX1,\$TMP21,LF IEX2,\$TMP20,0
6803		071766			DT7=DT6		
6804		071766			DT10=DT6		
6805		071766			DT11=DT6		
6806	072010	042466	001252	042473	DT12:	.WORD	\$THE,\$TMP10,NOOP1,NOOP15,NOOP2,\$TMP5
6807	072024	042634	001246	042646		.WORD	NOOP3,\$TMP6,NOOP4,NOOP2,\$TMP5,NOOP3,\$TMP7,NOOP5,\$TMP11
6808	072046	042711	001252	042731		.WORD	NOOP6,\$TMP10,NOOP7,NOOP10,\$TMP0,\$TMP1,\$TAB,\$TMP2
6809	072066	042272	001240	001242		.WORD	\$TAB,\$TMP3,\$TMP4,0
6810	072076	042466	001252	042473	DT13:	.WORD	\$THE,\$TMP10,NOOP1,NOOP11,NOOP10,\$TMP0,\$TMP1,\$TAB
6811	072116	001236	042272	001240		.WORD	\$TMP2,\$TAB,\$TMP3,\$TMP4,0
6812		071766			DT14=DT6		
6813		071766			DT15=DT6		
6814	072130	001232	001234	042272	DT16:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP5,\$TMP3,0
6815		071712			DT17=DT3		
6816	072150	001232	001234	042272	DT20:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
6817	072164	042272	001242	000000		.WORD	\$TAB,\$TMP4,0
6818	072172	001232	001234	042272	DT21:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,0
6819	072204	065346	001313		DT22:	.WORD	DH3,\$CRLF
6820	072210	001232	001234	042272		.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
6821	072224	042272	001242	000000		.WORD	\$TAB,\$TMP4,0
6822	072232	001232	001234	042272	DT23:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
6823	072246	042272	001242	000000		.WORD	\$TAB,\$TMP4,0
6824	072254	043365			DT24:	.WORD	ILLMS
6825	072256	001232	001234	042272		.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
6826	072272	042272	001242	000000		.WORD	\$TAB,\$TMP4,0
6827	072300	001232	001234	042272	DT25:	.WORD	\$TMP0,\$TMP1,\$TAB,\$CRLF,MS1,MS3,\$TMP3,MS4,\$TMP4,\$CRLF
6828	072324	001242	001313	043445		.WORD	\$TMP4,\$CRLF,MS2,MS3,\$TMP5,MS4,\$TMP6,\$CRLF,\$TMP5,0
6829	072350	001232	001234	042272	DT26:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3,\$TMP4,\$CRLF
6830	072370	043555	001313	043573		.WORD	MS6,\$CRLF,MS7,\$TMP5,MS10,\$TMP6,\$CRLF
6831	072406	043622	001313	043573		.WORD	MS11,\$CRLF,MS7,\$TMP5,MS10,\$TMP7,0
6832	072424	001232	001234	042272	DT27:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
6833	072440	001242	001313	043647		.WORD	\$TMP4,\$CRLF,MS12,\$CRLF,MS7,\$TMP5,MS10,\$TMP7,\$CRLF,MS13,0
6834	072466	044010	001272	001313	DT30:	.WORD	MS15,\$TMP20,\$CRLF,MS14,\$CRLF
6835	072500	001232	001234	042272		.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
6836	072514	001242	000000			.WORD	\$TMP4,0
6837		072350			DT31=DT26		
6838		072424			DT32=DT27		
6839	072520	001232	001234	042272	DT33:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
6840	072534	001313	043433	043452		.WORD	\$CRLF,MS1,MS3,\$TMP4,MS4,\$TMP5,\$CRLF,\$TMP6,\$CRLF
6841	072556	043445	043452	001242		.WORD	MS2,MS3,\$TMP4,MS4,\$TMP5,\$CRLF,\$TMP4,0
6842	072576	001232	001234	042272	DT34:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
6843	072612	001313	043555	001313		.WORD	\$CRLF,MS6,\$CRLF,MS7,\$TMP5,MS10,\$TMP6,\$CRLF
6844	072632	043622	001313	043573		.WORD	MS11,\$CRLF,MS7,\$TMP5,MS10,\$TMP7,0
6845		072576			DT35=DT34		
6846		072576			DT36=DT34		
6847		072576			DT37=DT34		
6848		072520			DT40=DT33		
6849	072650	001272	001313	065346	DT41:	.WORD	\$TMP20,\$CRLF,DH3,\$CRLF



6850	072660	001232	001234	042272	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
6851	072674	042272	001242	000000	.WORD	\$TAB,\$TMP4,0
6852	072702	001232	001234	042272	DT42: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
6853	072716	001313	043555	001313	.WORD	\$CRLF,MS6,\$CRLF,MS7,\$TMP5,MS10,\$TMP6,\$TMP15,\$TMP10
6854	072740	001313	043622	001313	.WORD	\$CRLF,MS11,\$CRLF,MS7,\$TMP5,MS10,\$TMP7,0
6855		072702			DT43=DT42	
6856	072760	044042	001244	001313	DT44: .WORD	MS17,\$TMP5,\$CRLF,MS20,\$CRLF,\$TMP0,\$TMP1,\$TAB,\$TMP2
6857	073002	001313	043433	044173	.WORD	\$CRLF,MS1,MS21,\$CRLF,\$TMP3,\$CRLF,MS2,MS21,\$CRLF,\$TMP4,0
6858	073030	044042	001244	001313	DT45: .WORD	MS17,\$TMP5,\$CRLF,MS24,\$CRLF,\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB
6859	073054	001246	001313	044232	.WORD	\$TMP6,\$CRLF,MS22,MS21,\$CRLF,\$TMP3,\$CRLF
6860	073072	044262	044173	001313	.WORD	MS23,MS21,\$CRLF,\$TMP4,0
6861	073104	001232	001234	001313	DT46: .WORD	\$TMP0,\$TMP1,\$CRLF,MS25,MS30,\$TMP2,MS31,\$TMP3
6862	073124	044470	001242	044477	.WORD	MS32,\$TMP4,MS33,\$TMP5,MS34,\$TMP6,\$CRLF,MS26
6863	073144	044452	001250	044461	.WORD	MS30,\$TMP7,MS31,\$TMP10
6864	073154	044470	001254	044477	.WORD	MS32,\$TMP11,MS33,\$TMP12,MS34,\$TMP13,0
6865	073172	001232	001234	042272	DT47: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$CRLF,MS12,MS7,\$TMP3,MS10
6866	073214	001242	001313	043705	.WORD	\$TMP4,\$CRLF,MS13,0
6867		072576			DT50=DT34	
6868	073224	001232	001234	042272	DT51: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
6869	073240	001313	044404	044443	.WORD	\$CRLF,MS25,MS27,\$TMP4,\$CRLF,MS26,MS27,\$TMP5,0
6870		073172			DT52=DT47	
6871	073262	001232	001234	042272	DT53: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
6872	073276	001242	000000		.WORD	\$TMP4,0
6873	073302	001232	001234	042272	DT54: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$CRLF,MS1,MS21,\$CRLF,\$TMP3
6874	073324	001313	043445	044173	.WORD	\$CRLF,MS2,MS21,\$CRLF,\$TMP3,0
6875		073172			DT55=DT47	
6876		073262			DT56=DT53	
6877		073302			DT57=DT54	
6878		073262			DT60=DT53	
6879		073302			DT61=DT54	
6880		073302			DT62=DT54	
6881		073302			DT63=DT54	
6882	073340	001232	001234	042272	DT64: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
6883	073354	001242	000000		.WORD	\$TMP4,0
6884		073340			DT65=DT64	
6885		073302			DT66=DT54	
6886		072172			DT67=DT21	
6887	073360	001232	001234	042272	DT70: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$CRLF,MS6,\$CRLF,MS7,\$TMP5
6888	073402	043607	001246	001313	.WORD	MS10,\$TMP6,\$CRLF,MS11,\$CRLF,MS7,\$TMP5,MS10,\$TMP7,0
6889		073360			DT71=DT70	
6890	073426	001232	001234	042272	DT72: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3,\$TMP4,0
6891		073302			DT73=DT54	
6892		072172			DT74=DT21	
6893		073360			DT75=DT70	
6894		073360			DT76=DT70	
6895		073426			DT77=DT72	
6896		073302			DT100=DT54	
6897		073360			DT101=DT70	
6898		073360			DT102=DT71	
6899		073360			DT103=DT70	
6900		073302			DT104=DT54	
6901		073360			DT105=DT70	
6902		073426			DT106=DT72	
6903		073360			DT107=DT70	
6904		073302			DT110=DT54	
6905	073446	001232	001234	042272	DT111: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,0
6906		073446			DT112=DT111	



6907	073446				DT113=DT111	
6908	073446				DT114=DT111	
6909	073446				DT115=DT111	
6910	073446				DT116=DT111	
6911	071712				DT117=DT3	
6912	071712				DT120=DT3	
6913	073172				DT121=DT47	
6914	072576				DT122=DT34	
6915	073224				DT123=DT51	
6916	073460	001232	001234	042272	DT124: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3,\$TMP4,\$CRLF
6917	073500	043555	001313	043573	.WORD	MS6,\$CRLF,MS7,\$TMP5,MS10,\$TMP6,\$CRLF
6918	073516	043622	001313	043573	.WORD	MS11,\$CRLF,MS7,\$TMP5,MS10,\$TMP7,\$CRLF,MS37,\$CRLF,\$TMP10,0
6919	073460				DT125=DT124	
6920	073446				DT126=DT111	
6921	073446				DT127=DT111	
6922	071712				DT130=DT3	
6923	073544	001232	001234	042272	DT131: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$CRLF,MS37,\$CRLF,\$TMP3
6924	073564	001313	044547	001313	.WORD	\$CRLF,MS40,\$CRLF,\$TMP4,\$CRLF,MS415,\$CRLF,\$TMP5,0
6925	073544				DT132=DT131	
6926	073606	001232	001234	042272	DT133: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$CRLF,MS41,\$CRLF,\$TMP3
6927	073626	001313	044627	001313	.WORD	\$CRLF,MS42,\$CRLF,\$TMP4,\$CRLF,MS43,\$CRLF,\$TMP5
6928	073646	001313	044662	001313	.WORD	\$CRLF,MS44,\$CRLF,\$TMP6,0
6929	073606				DT134=DT133	
6930	073606				DT135=DT133	
6931	073606				DT136=DT133	
6932	073660	001232	001234	042272	DT137: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TMP10,\$TAB,\$TMP11,0
6933	073660				DT140=DT137	
6934	073660				DT141=DT137	
6935	073660				DT142=DT137	
6936	073606				DT143=DT133	
6937	073606				DT144=DT133	
6938	073660				DT145=DT137	
6939	073660				DT146=DT137	
6940	073606				DT147=DT133	
6941	073606				DT150=DT133	
6942	073660				DT151=DT137	
6943	073606				DT152=DT133	
6944	073660				DT153=DT137	
6945	073700	001232	001234	042272	DT154: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,0
6946	073700				DT155=DT154	
6947	073606				DT156=DT133	
6948	073606				DT157=DT133	
6949	073606				DT160=DT133	
6950	073606				DT161=DT133	
6951	073606				DT162=DT133	
6952	073606				DT163=DT133	
6953	071712				DT164=DT3	
6954	071712				DT165=DT3	
6955	073606				DT166=DT133	
6956	073606				DT167=DT133	
6957	073606				DT170=DT133	
6958	073606				DT171=DT133	
6959	073606				DT172=DT133	
6960	073606				DT173=DT133	
6961	073606				DT174=DT133	
6962	073606				DT175=DT133	
6963	073606				DT176=DT133	



DATA TABLES

6964	073712	001232	001234	036502	DT177: .WORD	\$TMP0,\$TMP1,CPSAVE,0
6965		073606			DT200=DT133	
6966		073606			DT201=DT133	
6967		073606			DT202=DT133	
6968		073606			DT203=DT133	
6969		073606			DT204=DT133	
6970		071712			DT205=DT3	
6971		073606			DT206=DT133	
6972		073606			DT207=DT133	
6973		073606			DT210=DT133	
6974	073722	001232	001234	042272	DT211: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3,0
6975	073740	001232	001234	042272	DT212: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,0
6976		073740			DT213=DT212	
6977		073606			DT214=DT133	
6978		000001			.END	



SYMBOL TABLE

AA DATO	027226	ADDW12=	000000	A6	004526	BERR1	004762	CCP2	031034
AA DONE	027326	ADDW13=	000000	A7	004552	BITSTS	014072	CCP3	031044
AA ERRO	026660	ADDW14=	000000	BB DATO	032350	BIT0 =	000001	CCP4	031054
AA ERR1	026746	ADDW15=	000000	BB DONE	032500	BIT00 =	000001	CCP5	031064
AA ERR2	027002	ADDW2 =	000000	BBERO	032036	BIT01 =	000002	CCP6	031074
AA ERR3	027036	ADDW3 =	000000	BBER1	032076	BIT02 =	000004	CCP7	031104
AA ERR4	027046	ADDW4 =	000000	BBER10	032056	BIT03 =	000010	CC1	027332
AA ERR5	027102	ADDW5 =	000000	BBER11	032112	BIT04 =	000020	CC10	027556
AA ERR6	027136	ADDW6 =	000000	BBER2	032134	BIT05 =	000040	CC11	027566
AA ERR7	027172	ADDW7 =	000000	BBER3	032152	BIT06 =	000100	CC12	027574
AA ER10	026754	ADDW8 =	000000	BBER4	032206	BIT07 =	000200	CC13	027606
AA PATO	027236	ADDW9 =	000000	BBER40	032222	BIT08 =	000400	CC14	027642
AA PAT1	027246	ADEVCT=	000000	BBER5	032242	BIT09 =	001000	CC15	027666
AA PAT2	027256	ADEVVM =	000000	BBER6	032260	BIT1 =	000002	CC16	027706
AA PAT3	027266	ADONE =	004670	BBER7	032314	BIT10 =	002000	CC17	027716
AA PAT4	027276	AENV =	000000	BBER8	032332	BIT11 =	004000	CC18	027724
AA PAT5	027306	AENVVM =	000000	BBPAT0	032360	BIT12 =	010000	CC19	027736
AA PAT6	027316	AERFLG	004560	BBPAT1	032370	BIT13 =	020000	CC2	027366
AA1	026264	AERR1	004562	BBPAT2	032400	BIT14 =	040000	CC20	027772
AA10	026404	AERR2	004574	BBPAT3	032410	BIT15 =	100000	CC21	030016
AA11	026440	AERR3	004626	BBPAT4	032420	BIT2 =	000004	CC22	030036
AA12	026442	AFATAL=	000000	BBPAT5	032430	BIT3 =	000010	CC23	030046
AA13	026460	AMADR1=	000000	BBPAT6	032440	BIT4 =	000020	CC24	030054
AA14	026500	AMADR2=	000000	BBP10	032460	BIT5 =	000040	CC25	030066
AA15	026520	AMADR3=	000000	BBP11	032470	BIT6 =	000100	CC26	030122
AA16	026526	AMADR4=	000000	BBP7	032450	BIT7 =	000200	CC27	030144
AA17	026532	AMAMS1=	000000	BB1	031150	BIT8 =	000400	CC28	030164
AA2	026326	AMAMS2=	000000	BB10	031334	BIT9 =	001000	CC29	030174
AA20	026536	AMAMS3=	000000	BB11	031354	BPTVEC=	000014	CC3	027410
AA21	026540	AMAMS4=	000000	BB12	031364	B1	004706	CC30	030202
AA22	026576	AMSGAD=	000000	BB13	031372	B2	004710	CC31	030214
AA23	026600	AMSGLG=	000000	BB14	031404	B3	004726	CC32	030250
AA24	026620	AMSGTY=	000000	BB15	031440	CC DATO	031004	CC33	030272
AA25	026640	AMTYP1=	000000	BB16	031464	CC DONE	031144	CC34	030312
AA26	026646	AMTYP2=	000000	BB17	031474	CCER0	030346	CC35	030322
AA27	026652	AMTYP3=	000000	BB2	031212	CCER1	030402	CC36	030330
AA3	026330	AMTYP4=	000000	BB20	031506	CCER10	030714	CC37	030342
AA4	026346	APASS =	000000	BB21	031542	CCER11	030732	CC4	027430
AA5	026366	APRIOR=	000000	BB22	031566	CCER12	030750	CC5	027440
AA6	026374	APTCSU=	000040	BB23	031606	CCER13	030766	CC6	027446
AA7	026402	APTENV=	000001	BB24	031616	CCER2	030440	CC7	027460
ABASE =	000000	APTSIZ=	000200	BB25	031624	CCER22	030454	CC8	027514
ACDW1 =	000000	APTSPO=	000100	BB26	031636	CCER3	030476	CC9	027536
ACDW2 =	000000	ASWREG=	000000	BB27	031672	CCER4	030514	CDONE	005742
ACPUOP=	000000	ATESTN=	000000	BB3	031216	CCER44	030530	CERR1	005476
AC0 =	%000000	AUNIT =	000000	BB30	031714	CCER5	030552	CERR2	005574
AC1 =	%000001	AUSWR =	000000	BB31	031724	CCER50	030416	CERR3	005674
AC2 =	%000002	AVECT1=	000000	BB32	031736	CCER55	030566	CERR4	005716
AC3 =	%000003	AVECT2=	000000	BB33	031772	CCER6	030606	CFCC1	043223
AC4 =	%000004	A05	004476	BB34	032014	CCER7	030642	CHPNUM	013772
AC5 =	%000005	A1	004350	BB35	032024	CCER8	030660	CKSWR =	104407
AC6 =	%000006	A11	004350	BB4	031234	CCER90	030364	CNT =	000214
AC7 =	%000007	A12	004364	BB5	031244	CCP0	031014	CPC	005740
ADDW0 =	000000	A2	004422	BB6	031256	CCP1	031024	CPSAVE	036502
ADDW1 =	000000	A3	004426	BB7	031312	CCP10	031114	CPSPUR	042120
ADDW10=	000000	A4	004474	BDONE	004776	CCP11	031124	CPTWO	042136
ADDW11=	000000	A5	004500	BERR	004732	CCP12	031134	CR =	000015



CRLF = 000200  
C1 005016  
C15 005044  
C2 005070  
C25 005104  
C3 005132  
C35 005162  
C4 005210  
C45 005222  
C5 005250  
C55 005276  
C6 005322  
C65 005336  
C7 005364  
C75 005414  
C8 005442  
C85 005454  
DDDATO 034400  
DDDONE 034530  
DDERO 033612  
DDER1 033630  
DDER10 034246  
DDER11 034304  
DDER12 034342  
DDER2 033666  
DDER3 033724  
DDER4 033762  
DDER5 034020  
DDER6 034056  
DDER7 034114  
DDER8 034152  
DDER9 034210  
DDISP = 177570  
DDONE 006210  
DDPO 034410  
DDP1 034420  
DDP2 034430  
DDP3 034440  
DDP4 034450  
DDP5 034460  
DDP6 034470  
DDP7 034500  
DDP8 034510  
DDP9 034520  
DD1 032504  
DD10 032714  
DD11 032724  
DD12 032742  
DD13 032776  
DD14 033020  
DD15 033030  
DD16 033046  
DD17 033102  
DD18 033124  
DD19 033144  
DD2 032540  
DD20 033154

DD21 033162  
DD22 033174  
DD23 033230  
DD24 033252  
DD25 033272  
DD26 033302  
DD27 033310  
DD3 032562  
DD30 033322  
DD31 033356  
DD32 033400  
DD33 033420  
DD34 033430  
DD35 033436  
DD36 033454  
DD37 033510  
DD38 033532  
DD39 033552  
DD4 032602  
DD40 033562  
DD41 033570  
DD42 033606  
DD5 032612  
DD6 032620  
DD7 032636  
DD8 032672  
DERR1 006070  
DERR2 006164  
DF 1 070570  
DF 10 = 070620  
DF 100 = 071415  
DF 101 = 071442  
DF 102 = 071464  
DF 103 = 071442  
DF 104 = 071415  
DF 105 = 071442  
DF 106 = 071464  
DF 107 = 071442  
DF 11 = 070620  
DF 110 = 071415  
DF 111 = 071473  
DF 112 = 071473  
DF 113 = 071473  
DF 114 = 071473  
DF 115 = 071473  
DF 116 = 071473  
DF 117 = 070610  
DF 12 = 070634  
DF 120 = 070610  
DF 121 = 071327  
DF 122 = 071344  
DF 123 = 071370  
DF 124 = 071477  
DF 125 = 071477  
DF 126 = 071473  
DF 127 = 071473  
DF 13 = 070666

DF 130 = 070610  
DF 131 = 071530  
DF 132 = 071530  
DF 133 = 071550  
DF 134 = 071550  
DF 135 = 071550  
DF 136 = 071550  
DF 137 = 071574  
DF 14 = 070620  
DF 140 = 071574  
DF 141 = 071574  
DF 142 = 071574  
DF 143 = 071550  
DF 144 = 071550  
DF 145 = 071574  
DF 146 = 071574  
DF 147 = 071550  
DF 15 = 070620  
DF 150 = 071550  
DF 151 = 071574  
DF 152 = 071550  
DF 153 = 071574  
DF 154 = 071604  
DF 155 = 071604  
DF 156 = 071550  
DF 157 = 071550  
DF 16 = 070702  
DF 160 = 071550  
DF 161 = 071610  
DF 162 = 071550  
DF 163 = 071610  
DF 164 = 070610  
DF 165 = 070610  
DF 166 = 071550  
DF 167 = 071550  
DF 17 = 070610  
DF 170 = 071610  
DF 171 = 071610  
DF 172 = 071550  
DF 173 = 071550  
DF 174 = 071610  
DF 175 = 071610  
DF 176 = 071550  
DF 177 = 071634  
DF 2 = 070600  
DF 20 = 070711  
DF 200 = 071550  
DF 201 = 071550  
DF 202 = 071550  
DF 203 = 071550  
DF 204 = 071550  
DF 205 = 070610  
DF 206 = 071550  
DF 207 = 071550  
DF 21 = 070721  
DF 210 = 071550  
DF 211 = 071637

DF 212 = 071637  
DF 213 = 071637  
DF 214 = 071550  
DF 22 = 070725  
DF 23 = 070737  
DF 24 = 070747  
DF 25 = 070756  
DF 26 = 071002  
DF 27 = 071027  
DF 3 = 070610  
DF 30 = 071047  
DF 31 = 071002  
DF 32 = 071027  
DF 33 = 071063  
DF 34 = 071111  
DF 35 = 071111  
DF 36 = 071111  
DF 37 = 071111  
DF 4 = 070610  
DF 40 = 071135  
DF 41 = 071163  
DF 42 = 071177  
DF 43 = 071177  
DF 44 = 071225  
DF 45 = 071250  
DF 46 = 071275  
DF 47 = 071327  
DF 5 = 070620  
DF 50 = 071344  
DF 51 = 071370  
DF 52 = 071327  
DF 53 = 071406  
DF 54 = 071415  
DF 55 = 071327  
DF 56 = 071406  
DF 57 = 071415  
DF 6 = 070620  
DF 60 = 071406  
DF 61 = 071415  
DF 62 = 071415  
DF 63 = 071415  
DF 64 = 071433  
DF 65 = 071433  
DF 66 = 071415  
DF 67 = 070721  
DF 7 = 070620  
DF 70 = 071442  
DF 71 = 071442  
DF 72 = 071464  
DF 73 = 071415  
DF 74 = 070721  
DF 75 = 071442  
DF 76 = 071442  
DF 77 = 071464  
DH 1 = 065163  
DH 10 = 065533  
DH 100 = 067035

DH 101 = 065533  
DH 102 = 066425  
DH 103 = 066363  
DH 104 = 067035  
DH 105 = 065533  
DH 106 = 066425  
DH 107 = 066363  
DH 11 = 065533  
DH 110 = 067035  
DH 111 = 066363  
DH 112 = 067233  
DH 113 = 066363  
DH 114 = 067233  
DH 115 = 067452  
DH 116 = 067452  
DH 117 = 065653  
DH 12 = 000000  
DH 120 = 067736  
DH 121 = 065533  
DH 122 = 066763  
DH 123 = 066763  
DH 124 = 065653  
DH 125 = 065653  
DH 126 = 065533  
DH 127 = 067035  
DH 13 = 000000  
DH 130 = 067736  
DH 131 = 067035  
DH 132 = 067035  
DH 133 = 070026  
DH 134 = 070026  
DH 135 = 070026  
DH 136 = 070026  
DH 137 = 070136  
DH 14 = 065533  
DH 140 = 070136  
DH 141 = 070136  
DH 142 = 070136  
DH 143 = 070026  
DH 144 = 070026  
DH 145 = 070136  
DH 146 = 070136  
DH 147 = 067035  
DH 15 = 065533  
DH 150 = 070326  
DH 151 = 070136  
DH 152 = 070026  
DH 153 = 070136  
DH 154 = 070417  
DH 155 = 070417  
DH 156 = 067035  
DH 157 = 067035  
DH 16 = 065573  
DH 160 = 067035  
DH 161 = 067035  
DH 162 = 067035  
DH 163 = 067035



DH164 = 065653  
DH165 = 065653  
DH166 = 067035  
DH167 = 067035  
DH17 = 065653  
DH170 = 067035  
DH171 = 067035  
DH172 = 067035  
DH173 = 067035  
DH174 = 067035  
DH175 = 067035  
DH176 = 067035  
DH177 = 07045  
DH2 = 065253  
DH20 = 065743  
DH200 = 067035  
DH201 = 067035  
DH202 = 067035  
DH203 = 067035  
DH204 = 067035  
DH205 = 065653  
DH206 = 067035  
DH207 = 067035  
DH21 = 065533  
DH210 = 067035  
DH211 = 070523  
DH212 = 065533  
DH213 = 065533  
DH214 = 067035  
DH22 = 066031  
DH23 = 066066  
DH24 = 066224  
DH25 = 066363  
DH26 = 066425  
DH27 = 066425  
DH3 = 065346  
DH30 = 000000  
DH31 = 066425  
DH32 = 066425  
DH33 = 066513  
DH34 = 066513  
DH35 = 066513  
DH36 = 066513  
DH37 = 066513  
DH4 = 065437  
DH40 = 066513  
DH41 = 000000  
DH42 = 066616  
DH43 = 066616  
DH44 = 000000  
DH45 = 066720  
DH46 = 066737  
DH47 = 066720  
DH5 = 065533  
DH50 = 066763  
DH51 = 066763  
DH52 = 065533

DH53 = 066425  
DH54 = 067035  
DH55 = 065533  
DH56 = 066425  
DH57 = 067035  
DH6 = 065533  
DH60 = 066425  
DH61 = 067035  
DH62 = 067035  
DH63 = 067035  
DH64 = 067145  
DH65 = 067076  
DH66 = 067035  
DH67 = 065533  
DH7 = 065533  
DH70 = 065533  
DH71 = 066363  
DH72 = 066425  
DH73 = 067035  
DH74 = 065533  
DH75 = 065533  
DH76 = 066363  
DH77 = 066425  
DISPLA 001142  
DISPRE 000174  
DPAT3 017322  
DSWR = 177570  
DT1 = 071646  
DT10 = 071766  
DT100 = 073302  
DT101 = 073360  
DT102 = 073360  
DT103 = 073360  
DT104 = 073302  
DT105 = 073360  
DT106 = 073426  
DT107 = 073360  
DT11 = 071766  
DT110 = 073302  
DT111 = 073446  
DT112 = 073446  
DT113 = 073446  
DT114 = 073446  
DT115 = 073446  
DT116 = 073446  
DT117 = 071712  
DT12 = 072010  
DT120 = 071712  
DT121 = 073172  
DT122 = 072576  
DT123 = 073224  
DT124 = 073460  
DT125 = 073460  
DT126 = 073446  
DT127 = 073446  
DT13 = 072076  
DT130 = 071712

DT131 = 073544  
DT132 = 073544  
DT133 = 073606  
DT134 = 073606  
DT135 = 073606  
DT136 = 073606  
DT137 = 073660  
DT14 = 071766  
DT140 = 073660  
DT141 = 073660  
DT142 = 073660  
DT143 = 073606  
DT144 = 073606  
DT145 = 073660  
DT146 = 073660  
DT147 = 073606  
DT15 = 071766  
DT150 = 073606  
DT151 = 073660  
DT152 = 073606  
DT153 = 073660  
DT154 = 073700  
DT155 = 073700  
DT156 = 073606  
DT157 = 073606  
DT16 = 072130  
DT160 = 073606  
DT161 = 073606  
DT162 = 073606  
DT163 = 073606  
DT164 = 071712  
DT165 = 071712  
DT166 = 073606  
DT167 = 073606  
DT17 = 071712  
DT170 = 073606  
DT171 = 073606  
DT172 = 073606  
DT173 = 073606  
DT174 = 073606  
DT175 = 073606  
DT176 = 073606  
DT177 = 073712  
DT2 = 071670  
DT20 = 072150  
DT200 = 073606  
DT201 = 073606  
DT202 = 073606  
DT203 = 073606  
DT204 = 073606  
DT205 = 071712  
DT206 = 073606  
DT207 = 073606  
DT21 = 072172  
DT210 = 073606  
DT211 = 073722  
DT212 = 073740

DT213 = 073740  
DT214 = 073606  
DT22 = 072204  
DT23 = 072232  
DT24 = 072254  
DT25 = 072300  
DT26 = 072350  
DT27 = 072424  
DT3 = 071712  
DT30 = 072466  
DT31 = 072350  
DT32 = 072424  
DT33 = 072520  
DT34 = 072576  
DT35 = 072576  
DT36 = 072576  
DT37 = 072576  
DT4 = 071712  
DT40 = 072520  
DT41 = 072650  
DT42 = 072702  
DT43 = 072702  
DT44 = 072760  
DT45 = 073030  
DT46 = 073104  
DT47 = 073172  
DT5 = 071734  
DT50 = 072576  
DT51 = 073224  
DT52 = 073172  
DT53 = 073262  
DT54 = 073302  
DT55 = 073172  
DT56 = 073262  
DT57 = 073302  
DT6 = 071766  
DT60 = 073262  
DT61 = 073302  
DT62 = 073302  
DT63 = 073302  
DT64 = 073340  
DT65 = 073340  
DT66 = 073302  
DT67 = 072172  
DT7 = 071766  
DT70 = 073360  
DT71 = 073360  
DT72 = 073426  
DT73 = 073302  
DT74 = 072172  
DT75 = 073360  
DT76 = 073360  
DT77 = 073426  
D1 = 005774  
D10 = 006176  
D2 = 006020  
D3 = 006022

D4 = 006026  
D5 = 006040  
D6 = 006054  
D7 = 006064  
D8 = 006134  
D9 = 006146  
EDONE = 006354  
EEDATO = 035232  
EEDONE = 035314  
EEERO = 035024  
EEER1 = 035042  
EEER2 = 035100  
EEER3 = 035136  
EEER4 = 035174  
EEP0 = 035242  
EEP1 = 035254  
EEP2 = 035264  
EEP3 = 035274  
EEP4 = 035304  
EERRO = 006272  
EERR1 = 006310  
EERR2 = 006324  
EE1 = 034534  
EE10 = 034764  
EE11 = 034774  
EE12 = 035002  
EE13 = 035020  
EE2 = 034570  
EE3 = 034612  
EE4 = 034632  
EE5 = 034642  
EE6 = 034650  
EE7 = 034666  
EE8 = 034722  
EE9 = 034744  
EMTVEC = 000030  
EM1 = 044704  
EM10 = 045112  
EM100 = 054014  
EM101 = 054054  
EM102 = 054200  
EM103 = 054252  
EM104 = 054355  
EM105 = 054416  
EM106 = 054543  
EM107 = 054616  
EM11 = 045233  
EM110 = 054722  
EM111 = 054764  
EM112 = 054764  
EM113 = 055066  
EM114 = 055066  
EM115 = 054764  
EM116 = 055066  
EM117 = 055170  
EM12 = 000000  
EM120 = 055324



EM121	055460	EM203	064223	EM73	053222	FFP1	035642	GFI AG2	014756
EM122	055577	EM204	064360	EM74	053262	FFP2	035652	GOR0	015010
EM123	055676	EM205	064515	EM75	053513	FFP3	035662	GOR1	015012
EM124	055737	EM206	064562	EM76	053636	FFP4	035672	GOR2	015014
EM125	056032	EM207	064627	EM77	053740	FF1	035320	GOR3	015016
EM126	056122	EM21	046244	ERM10	037042	FF10	035500	GPAT00	014760
EM127	056331	EM210	064751	ERROR =	104000	FF11	035510	GPAT01	014762
EM13 =	000000	EM211 =	045112	ERRVEC =	000004	FF2	035354	GPAT02	014764
EM130	056544	EM212 =	045146	ERTYPE	041342	FF3	035376	GPAT03	014766
EM131	056644	EM213 =	045200	ERT1	041526	FF4	035404	GPAT10	014770
FM132	056704	EM214	055026	ERT2	041712	FF5	035414	GPAT11	014772
EM133	056744	EM22	046375	ERT3	041716	FF6	035450	GPAT12	014774
EM134	057003	EM23 =	046375	ERT4	041726	FF7	035472	GPAT13	014776
EM135	057042	EM24 =	046375	ERT5	041740	FPSMS	042411	GRESET	014170
EM136	057101	EM25	046462	E1	006230	FPSPUR	042066	GSETUP	014112
EM137 =	056744	EM26	046575	E2	006244	FPVECT =	000244	GSUM	014322
EM14	045314	EM27 =	046575	E3	006244	FXDAT0	010240	GS1	014142
EM140 =	057003	EM3	045005	E4	006246	FXDAT1	010242	GTSWR =	104406
EM141 =	057042	EM30	046643	FDATIO	010174	FXDAT2	010244	G1	012020
EM142 =	057101	EM31	046715	FDATI1	010176	FXDAT3	010246	G10	012326
EM143	057140	EM32 =	046715	FDATI2	010200	FXDAT4	010250	G11	012330
EM144	057173	EM33	046763	FDATI3	010202	FXDAT5	010252	G12	012422
EM145 =	057140	EM34	047024	FDATI4	010204	FXDAT6	010254	G13	012454
EM146 =	057173	EM35	047126	FDATI5	010206	FXDAT7	010256	G14	012456
EM147	057226	EM36	047230	FDATI6	010210	F1	006360	G15	012550
EM15 =	045437	EM37	047331	FDATI7	010212	F10	006602	G16	012602
EM150 =	057226	EM4	045052	FDAT00	010216	F11	006604	G17	012604
EM151 =	057226	EM40	047432	FDAT01	010220	F12	006622	G2	012052
EM152	057260	EM41	047603	FDAT02	010222	F13	006654	G20	012676
EM153 =	057260	EM42	047640	FDAT03	010224	F135	006634	G21	012730
EM154	057312	EM43	047761	FDAT04	010226	F14	006664	G22	012732
EM155	057544	EM44	050102	FDAT05	010230	F15	006674	G23	013024
EM156	057777	EM45 =	050102	FDAT06	010232	F16	006704	G24	013056
EM157	060214	EM46	050145	FDAT07	010234	F17	006714	G25	013060
EM16	045562	EM47	050223	FDONE	010260	F2	006416	G26	013152
EM160	060433	EM5	045112	FECMS	042455	F20	006724	G27	013204
EM161	060640	EM50	050341	FERP0	006766	F21	006734	G3	012054
EM162	061045	EM51	050437	FERR1	007024	F22	006744	G30	013206
EM163	061112	EM52	050500	FERR10	007604	F23	006762	G31	013300
EM164	061157	EM53	050621	FERR11	007640	F3	006436	G32	013332
EM165	061224	EM54	051016	FERR2	007122	F4	006440	G33	013334
EM166	061271	EM55	051062	FERR20	007660	F5	006456	G34	013426
EM167	061401	EM56	051203	FERR21	007776	F6	006532	G35	013460
EM17	045633	EM57	051400	FERR25	010026	F7	006562	G36	013462
EM170	061640	EM6	045146	FERR26	010144	GADR	015030	G37	013554
EM171	061750	EM60	051444	FERR3	007162	GAND0	015000	G4	012146
EM172	062207	EM61	051641	FERR4	007154	GAND1	015002	G40	013606
EM173	062446	EM62	051705	FERR5	007260	GAND2	015004	G41	013610
EM174	062705	EM63	052077	FERR6	007314	GAND3	015006	G42	013702
EM175	063144	FM64	052271	FERR7	007450	GCMF	014210	G43	013734
EM176	063403	EM65 =	052271	FER2	007126	GDAT00	015020	G44	013736
EM177	063540	EM66	052425	FFDAT0	035622	GDAT01	015022	G5	012200
EM2	044741	EM67	052470	FFDONE	035702	GDAT02	015024	G6	012202
EM20	046066	EM7	045200	FFER0	035512	GDAT03	015026	G7	012274
EM200	063574	EM70	052721	FFER1	035526	GDONE	015032	HADR	015566
EM201	063731	EM71	053044	FFER2	035564	GERR1	014236	HA1R	015642
EM202	064066	EM72	053146	FFP0	035632	GFLAG1	014754	HA1W	015572



HA2R	015652	IPAT20	011102	KDAT10	017264	MDAT01	020174	MS5	043513
HA2w	015602	IPAT21	011104	KDAT11	017266	MDAT02	020176	MS6	043555
HA3R	015662	IPAT22	011106	KDAT12	017270	MDAT03	020200	MS7	043573
HA3w	015612	IPAT23	011110	KDAT13	017272	MDONE	020202	M1	017656
HA4R	015672	I1	010272	KDAT00	017304	MERR0	020026	M15	017676
HA4w	015622	I10	010454	KDAT01	017306	MERR1	020064	M2	017702
HA5R	015702	I105	010462	KDAT02	017310	MERR2	020000	M3	017704
HA5w	015632	I106	010456	KDAT03	017312	MERR3	020112	M4	017706
HCLR	015516	I11	010464	KDONE	017324	MNUMBE=	000214	M5	017742
HCLR1	015526	I12	010472	KERR0	017146	MNUM0	044067	M6	017746
HCMP	015460	I13	010506	KERR1	017212	MNUM1	044075	M7	017756
HCMP1	015500	I14	010550	KERR2	017236	MNUM2	044102	M8	017766
HCMP2	015510	I15	010552	KPAT0	017314	MNUM3	044107	M9	017776
HDAT1	015712	I16	010554	KPAT1	017316	MNUM4	044116	NDAT10	020700
HDAT2	015722	I17	010572	KPAT2	017320	MNUM5	044124	NDAT11	020702
HDAT3	015732	I2	010310	K1	017040	MPAT10	020152	NDAT12	020704
HDAT4	015742	I20	010632	K10	017166	MPAT11	020154	NDAT13	020706
HDAT5	015752	I21	010646	K2	017064	MPAT12	020156	NDAT00	020636
HDONE	015762	I22	010652	K3	017066	MPAT13	020160	NDAT01	020640
HERROR	015534	I23	010666	K4	017070	MPAT20	020162	NDAT02	020642
HFLAG	015570	I3	010356	K5	017124	MPAT21	020164	NDAT03	020644
HSTD	015402	I4	010360	K6	017132	MPAT22	020166	NDONE	020710
HT =	000011	I5	010362	K7	017144	MPAT23	020170	NERR0	020400
H1	015044	I6	010406	LDAT10	017630	MS1	043433	NERR1	020500
H10	015306	I7	010422	LDAT11	017632	MS10	043607	NERR10	020432
H11	015340	JBUF0	016766	LDAT12	017634	MS11	043622	NERR11	020444
H12	015372	JBUF1	016770	LDAT13	017636	MS12	043647	NERR2	020534
H2	015064	JBUF2	016772	LDAT00	017642	MS13	043705	NERR20	020506
H3	015074	JBUF3	016774	LDAT01	017644	MS14	043722	NERR3	020544
H4	015146	JDAT10	016776	LDAT02	017646	MS15	044010	NERR4	020554
H5	015170	JDAT11	017000	LDAT03	017650	MS16	044035	NERR5	020564
H6	015222	JDAT12	017002	LDONE	017652	MS17	044042	NERR6	020610
H7	015254	JDAT13	017004	LD1	043121	MS2	043445	NOOP1	042473
IBSAVE	036504	JDAT00	017006	LD2	043151	MS20	044132	NOOP10	042751
IDAT10	011122	JDAT0	017016	LERR1	017464	MS21	044173	NOOP11	043042
IDAT11	011124	JDAT01	017010	LERR2	017536	MS22	044232	NOOP15	042522
IDAT12	011126	JDAT02	017012	LERR3	017510	MS23	044262	NOOP2	042617
IDAT13	011130	JDAT03	017014	LF =	000012	MS24	044311	NOOP3	042634
IDAT00	011112	JDAT1	017020	LF IEX1	042277	MS25	044404	NOOP4	042646
IDAT01	011114	JDAT2	017022	LF IEX2	042347	MS26	044426	NOOP5	042663
IDAT02	011116	JDAT3	017024	LFPS1	043106	MS27	044443	NOOP6	042711
IDAT03	011120	JDONE	017026	LOOP	004300	MS3	043452	NOOP7	042731
IDONE	011132	JERR0	016646	LPAT10	017606	MS30	044452	NPAT10	020670
IERR0	010672	JERR1	016714	LPAT11	017610	MS31	044461	NPAT11	020672
IERR1	010754	JERR2	016740	LPAT12	017612	MS32	044470	NPAT12	020674
IERR2	010774	J1	016540	LPAT13	017614	MS33	044477	NPAT13	020676
IERR25	011016	J10	016666	LPAT20	017616	MS34	044506	NPAT20	020656
IERR3	011046	J2	016564	LPAT21	017620	MS35	044515	NPAT21	020660
IERR4	011022	J3	016566	LPAT22	017622	MS36	044522	NPAT22	020662
ILLMS	043365	J4	016570	LPAT23	017624	MS37	044531	NPAT23	020664
ILL1	043261	J5	016624	L1	017336	MS4	043501	NULL	042271
ILL2	043324	J6	016632	L2	017376	MS40	044547	N1	020206
IOTVEC=	000020	J7	016644	L3	017400	MS41	044603	N10	020340
IPAT10	011072	KBUF0	017274	L4	017402	MS415	044563	N11	020350
IPAT11	011074	KBUF1	017276	L5	017452	MS42	044627	N12	020352
IPAT12	011076	KBUF2	017300	L6	017460	MS43	044645	N13	020366
IPAT13	011100	KBUF3	017302	MDAT00	020172	MS44	044662	N14	020376



N2	020232	PDAT03	022066	QERR0	022236	SERR7	016404	TAB	= 000011
N3	020234	PDONE	022070	QERR1	022502	SETD1	043237	TBITVE	= 000014
N4	020236	PERR0	021536	QERR11	022246	SETF1	043231	TDAT10	011740
N5	020254	PERR1	021756	QERR12	022264	SETI1	043245	TDAT11	011742
N6	020264	PERR10	021556	QERR13	022302	SETL1	043253	TDAT12	011744
N7	020274	PERR11	021566	QERR14	022320	SPACE	042274	TDAT13	011746
N8	020312	PERR12	021604	QERR15	022336	SPAT10	016506	TDAT00	011730
N9	020324	PERR13	021622	QERR16	022346	SPAT11	016510	TDAT01	011732
ODAT10	021404	PERR14	021640	QERR17	022354	SPAT12	016512	TDAT02	011734
ODAT11	021406	PERR15	021656	QERR2	022436	SPAT13	016514	TDAT03	011736
ODAT12	021410	PERR16	021666	QERR20	022402	STACK	= 001100	TDONE	011750
ODAT13	021412	PERR17	021674	QERR21	022412	START	003616	TERR0	011516
ODAT00	021342	PERR2	022004	QERR22	022420	STFS1	043162	TERR1	011600
ODAT01	021344	PERR20	021722	QERR3	022446	STKLMT	= 177774	TERR2	011620
ODAT02	021346	PERR21	021732	QERR4	022454	STST1	043312	TERR25	011634
ODAT03	021350	PERR22	021740	QPAT10	022530	ST1	043175	TERR3	011664
ODONE	021414	PFECAD	042040	QPAT11	022532	ST2	043212	TERR4	011646
OERR0	021104	PFECDH	042010	QPAT12	022534	SWR	001145	TKVEC	= 000060
OERR1	021204	PFECM	041754	QPAT13	022536	SWREG	000176	TOCTNM	042054
OERR10	021156	PFECFT	042050	QPAT20	022540	SW0	= 000001	TPAT10	011710
OERR11	021150	PFECWS	041744	QPAT21	022542	SW00	= 000001	TPAT11	011712
OERR2	021240	PIRQ	= 177772	QPAT22	022544	SW01	= 000002	TPAT12	011714
OERR20	021212	PIRQVE	= 000240	QPAT23	022546	SW02	= 000004	TPAT13	011716
OERR3	021250	POWERM	042224	Q1	022074	SW03	= 000010	TPAT20	011720
OERR4	021260	PPAT10	022040	Q10	022232	SW04	= 000020	TPAT21	011722
OERR5	021270	PPAT11	022042	Q2	022116	SW05	= 000040	TPAT22	011724
OERR6	021314	PPAT12	022044	Q3	= 022120	SW06	= 000100	TPAT23	011726
OPAT10	021374	PPAT13	022046	Q4	022122	SW07	= 000200	TPVEC	= 000064
OPAT11	021376	PROGNU	= 000001	Q5	022144	SW08	= 000400	TRAPVE	= 000034
OPAT12	021400	PRO	= 000000	Q6	022166	SW09	= 001000	TRTVEC	= 000014
OPAT13	021402	PR1	= 000040	Q7	022176	SW1	= 000002	TST1	004300
OPAT20	021360	PR2	= 000100	Q8	022210	SW10	= 002000	TST10	011134
OPAT21	021362	PR3	= 000140	Q9	022224	SW11	= 004000	TST11	011752
OPAT22	021364	PR4	= 000200	RDCHR	= 104410	SW12	= 010000	TST12	015034
OPAT23	021366	PR5	= 000240	RESREG	= 104412	SW13	= 020000	TST13	015764
OPAT24	021370	PR6	= 000300	RESVEC	= 000010	SW14	= 040000	TST14	016530
O1	020714	PR7	= 000340	RSETUP	= 104413	SW15	= 100000	TST15	017030
O10	021044	PS	= 177776	R6	= %000006	SW2	= 000004	TST16	017326
O11	021054	PSW	= 177776	R7	= %000007	SW3	= 000010	TST17	017654
O12	021056	PWRVEC	= 000024	SADR	016502	SW4	= 000020	TST2	004672
O13	021072	P1	021420	SAVREG	= 104411	SW5	= 000040	TST20	020204
O14	021102	P2	021442	SCOPE	= 000004	SW6	= 000100	TST21	020712
O2	020740	P3	= 021444	SDAT00	016516	SW7	= 000200	TST22	021416
O3	020742	P4	021446	SDAT01	016520	SW8	= 000400	TST23	022072
O4	020744	P5	021470	SDAT02	016522	SW9	= 001000	TST24	022572
O5	020762	P6	021512	SDAT03	016524	S1	015766	TST25	023732
O6	020772	P7	021522	SDONE	016526	S10	016144	TST26	024530
O7	021002	P8	021532	SERR0	016206	S11	016170	TST27	025344
O8	021016	QDAT10	022560	SERR1	016416	S12	016200	TST3	005000
O9	021030	QDAT11	022562	SERR10	016226	S2	016026	TST30	025754
PDAT10	022050	QDAT12	022564	SERR15	016306	S3	016030	TST31	026262
PDAT11	022052	QDAT13	022566	SERR2	016346	S4	016034	TST32	027330
PDAT12	022054	QDAT00	022550	SERR20	016326	S5	016060	TST33	031146
PDAT13	022056	QDAT01	022552	SERR3	016372	S6	016070	TST34	032502
PDAT00	022060	QDAT02	022554	SERR4	016264	S7	016076	TST35	034532
PDAT01	022062	QDAT03	022556	SERR5	016450	S8	016136	TST36	035316
PDAT02	022064	QDONE	022570	SERR6	016360	S9	016140	TST37	035702



TST4	005744	UPAT41	023710	XERR3	025214	Y3	025446	\$DBLK	040146
TST5	006212	UPAT42	023712	XERR4	025264	Y4	025464	\$DDW0	001402
TST6	006356	UPAT43	023714	XPAT00	025312	Y5	025524	\$DDW1	001404
TST7	010262	UROM1	023724	XPAT01	025314	Y6	025526	\$DDW10	001426
TYPDS =	104405	UROM2	023726	XPAT02	025316	Y7	025542	\$DDW11	001430
TYPE =	104401	UROM3	023730	XPAT03	025320	ZDAT00	026220	\$DDW12	001432
TYPOC =	104402	UTMP1	023720	XPAT10	025322	ZDAT01	026222	\$DDW13	001434
TYPON =	104404	UTMP2	023722	XPAT12	025326	ZDAT02	026224	\$DDW14	001436
TYPOS =	104403	U0	022610	XPAT13	025330	ZDAT03	026226	\$DDW15	001440
T1	011136	U1	022640	XPAT20	025332	ZDONE	026260	\$DDW2	001406
T10	011304	U10	023216	XPAT21	025334	ZERR1	026124	\$DDW3	001410
T105	011312	U11	023220	XPAT22	025336	ZERR2	026166	\$DDW4	001412
T11	011314	U12	023252	XPAT23	025340	ZFLAG	026212	\$DDW5	001414
T12	011322	U13	023270	XTMP	025300	ZPAT00	026230	\$DDW6	001416
T13	011336	U14	023322	X1	024532	ZPAT01	026232	\$DDW7	001420
T14	011400	U15	023356	X10	024734	ZPAT02	026234	\$DDW8	001422
T15	011402	U16	023360	X11	024750	ZPAT03	026236	\$DDW9	001424
T16	011404	U2	022710	X12	025010	ZPAT10	026240	\$DEVCT	001326
T17	011422	U3	022742	X13	025034	ZPAT11	026242	\$DEVN	001374
T2	011162	U4	023012	X14	025042	ZPAT12	026244	\$DOAGN	036120
T20	011462	U5	023044	X15	025056	ZPAT13	026246	\$DTBL	040136
T21	011476	U6	023114	X16	025116	ZPAT20	026250	\$ENDAD	036110
T22	011500	U7	023146	X17	025142	ZPAT21	026252	\$ENDCT	035736
T23	011512	WDAPO0	024516	X2	024572	ZPAT22	026254	\$ENULL	036164
T3	011230	WDAT01	024520	X20	025150	ZPAT23	026256	\$ENV	001336
T4	011232	WDAT02	024522	X21	025164	ZTMP1	026214	\$ENVM	001337
T5	011234	WDAT03	024524	X3	024616	ZTMP2	026216	\$EOP	035702
T6	011260	WDONE	024526	X4	024624	Z1	025776	\$EOPCT	035730
T7	011274	WPAT00	024506	X5	024642	Z2	026024	\$ERFLG	001103
UDONE	023732	WPAT01	024510	X6	024702	Z3	026050	\$ERMAX	001115
UERRO	023376	WPAT02	024512	X7	024726	Z4	026056	\$ERROR	036506
UERR1	023422	WPAT03	024514	YDAT00	025712	Z5	026070	\$ERRPC	001116
UERR10	023430	WSETUP	024454	YDAT01	025714	Z6	026122	\$ERRTB	001442
UERR11	023472	W1	023734	YDAT02	025716	\$APTHD	003602	\$ERTTL	001112
UERR2	023506	W10	024156	YDAT03	025720	\$ATYC	040202	\$ESCAP	001304
UERR20	023514	W11	024204	YDONE	025752	\$ATY1	040156	\$ETABL	001336
UERR21	023556	W12	024240	YERR1	025552	\$ATY3	040164	\$ETEND	001442
UERR3	023572	W13	024264	YERR2	025614	\$ATY4	040174	\$FATAL	001320
UERR4	023622	W14	024300	YERR3	025656	\$AUTOB	001134	\$FFLG	040422
UFLAG	023716	W15	024326	YFLAG	025702	\$BASE	001372	\$FILLC	001156
UPAT00	023646	W16	024366	YPAT00	025722	\$BADR	001122	\$FILLS	001155
UPAT01	023650	W17	024412	YPAT01	025724	\$BDDAT	001126	\$GDADR	001120
UPAT02	023652	W2	023770	YPAT02	025726	\$BELL	001306	\$GDDAT	001124
UPAT03	023654	W20	024426	YPAT03	025730	\$CDW1	001376	\$GET42	036062
UPAT10	023656	W3	024014	YPAT10	025732	\$CDW2	001400	\$GTSWR	040474
UPAT11	023660	W4	024032	YPAT11	025734	\$CHARC	037500	\$HD =	000003
UPAT12	023662	W5	024062	YPAT12	025736	\$CKSWR	040424	\$HIBTS	003602
UPAT13	023664	W6	024122	YPAT13	025740	\$CLR.T	036100	\$ICNT	001104
UPAT20	023666	W7	024146	YPAT20	025742	\$CMTAG	001100	\$ILLUP	041334
UPAT21	023670	XAPT11	025324	YPAT21	025744	\$CM1 =	000024	\$INTAG	001135
UPAT22	023672	XDAT00	025302	YPAT22	025746	\$CM2 =	000050	\$ITEMB	001114
UPAT23	023674	XDAT01	025304	YPAT23	025750	\$CM3 =	000024	\$LF	001314
UPAT30	023676	XDAT02	025306	YTMP1	025704	\$CM4 =	000024	\$LFLG	040421
UPAT31	023700	XDAT03	025310	YTMP2	025706	\$CNTLG	041043	\$LOOP	036156
UPAT32	023702	XDONE	025342	YTMP3	025710	\$CNTLU	041036	\$LPADR	001106
UPAT33	023704	XERR1	025166	Y1	025374	\$CPUOP	001344	\$LPERR	001110
UPAT40	023706	XERR2	025250	Y2	025422	\$CRLF	001313	\$MADR1	001350



\$MADR2	001354	\$PASTM	003610	\$REG4	001172	\$TKS	001144	\$STRAP	041072
\$MADR3	001360	\$PWAD	041316	\$REG5	001174	\$TMP0	001232	\$STRAP2	041114
\$MADR4	001364	\$PWRDN	041156	\$REG6	001176	\$TMP1	001234	\$TRP =	000014
\$MAIL	001316	\$PWARMG	041312	\$REG7	001200	\$TMP10	001252	\$TRPAD	041126
\$MAMS1	001346	\$PWRUP	041230	\$RESRE	037102	\$TMP11	001254	\$STSTM	003606
\$MAMS2	001352	\$QUES	001312	\$RTNAD	036160	\$TMP12	001256	\$STSTM	001102
\$MAMS3	001356	\$RDCHR	040706	\$RTRN	036154	\$TMP13	001260	\$STPDS	037732
\$MAMS4	001362	\$RDSZ =	000001	\$SAVRE	037044	\$TMP14	001262	\$STYPE	037140
\$MBADR	003604	\$REGAD	001160	\$SAVR6	041340	\$TMP15	001264	\$STYPEC	037352
\$MFLG	040420	\$REG0	001162	\$SCOPE	036170	\$TMP16	001266	\$STYPEX	037502
\$MNEW	041061	\$REG1	001164	\$SETUP =	000137	\$TMP17	001270	\$STYPOC	037530
\$MSGAD	001332	\$REG10	001202	\$STUP =	177777	\$TMP2	001236	\$STYPOD	037544
\$MSGLG	001334	\$REG11	001204	\$SVLAD	036430	\$TMP20	001272	\$STYPOS	037504
\$MSGTY	001316	\$REG12	001206	\$SVPC =	003602	\$TMP21	001274	\$UNIT	001330
\$MSWR	041050	\$REG13	001210	\$SWR =	177400	\$TMP22	001276	\$UNITM	003612
\$MTYP1	001347	\$REG14	001212	\$SWREG	001340	\$TMP23	001300	\$USWR	001342
\$MTYP2	001353	\$REG15	001214	\$SWRMK =	000000	\$TMP3	001240	\$VECT1	001366
\$MTYP3	001357	\$REG16	001216	\$SWRMS =	000200	\$TMP4	001242	\$VECT2	001370
\$MTYP4	001363	\$REG17	001220	\$TAB	042272	\$TMP5	001244	\$XOFF =	000023
\$MXCNT	036500	\$REG2	001166	\$TBIT	036162	\$TMP6	001246	\$XON =	000021
\$NULL	001154	\$REG20	001222	\$TERM =	000030	\$TMP7	001250	\$XTSTR	036202
\$NWTST =	000001	\$REG21	001224	\$TESTN	001322	\$TN =	000037	\$GET4 =	000001
\$OCNT	037726	\$REG22	001226	\$THE	042466	\$TPB	001152	\$OFILL	037727
\$OMODE	037730	\$REG23	001230	\$TIMES	001302	\$TPFLG	001157	.RSET	042154
\$OVER	036464	\$REG3	001170	\$TKB	001146	\$TPS	001150	.\$X =	003602
\$PASS	001324								

. ABS. 073752 000  
000000 001

ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 58024 WORDS ( 227 PAGES)  
DYNAMIC MEMORY: 20034 WORDS ( 77 PAGES)  
ELAPSED TIME: 00:13:11  
CKFPAC.BIN,CKFPAC/CR/-SP/NL:TOC=CKFPAC.MLB/ML,CKFPAC.P11



SYMBOL	CROSS REFERENCE	REFERENCES								
SYMBOL	VALUE									
AADATO	027226	49-4891	49-4898	49-4918	49-4925	49-4930	49-4956	49-4963	49-5001	49-5007
		49-5015	49-5021	49-5026	49-5032	#49-5036				
AADONE	027326	49-4971	49-4991	49-5003	49-5009	49-5017	49-5023	49-5029	49-5035	#49-5044
AAERRO	026660	49-4883	#49-4975							
AAERR1	026746	49-4902	#49-4998							
AAERR2	027002	49-4904	#49-5004							
AAERR3	027036	49-4934	#49-5010							
AAERR4	027046	49-4936	#49-5012							
AAERR5	027102	49-4938	#49-5018							
AAERR6	027136	49-4967	#49-5024							
AAERR7	027172	49-4969	#49-5030							
AAER10	026754	#49-4999	49-5011							
AAPATO	027236	49-4884	49-4912	49-4946	49-4999	49-5005	49-5013	49-5019	49-5025	49-5031
		#49-5037								
AAPAT1	027246	49-4888	49-4915	49-5000	49-5006	49-5014	49-5020	#49-5038		
AAPAT2	027256	49-4893	49-4924	49-4962	49-4998	49-5004	#49-5039			
AAPAT3	027266	49-4897	49-4920	49-5010	49-5012	49-5018	#49-5040			
AAPAT4	027276	49-4929	#49-5041							
AAPAT5	027306	49-4950	49-5024	49-5030	#49-5042					
AAPAT6	027316	49-4958	49-5027	49-5033	#49-5043					
AA1	026264	#49-4880								
AA10	026404	#49-4909								
AA11	026440	49-4914	#49-4916							
AA12	026442	#49-4918								
AA13	026460	#49-4922	49-4939							
AA14	026500	#49-4927	49-4937							
AA15	026520	#49-4932	49-4935							
AA16	026526	49-4933	#49-4935							
AA17	026532	49-4928	#49-4937							
AA2	026326	49-4887	#49-4889							
AA20	026536	49-4923	#49-4939							
AA21	026540	#49-4943								
AA22	026576	49-4949	#49-4951							
AA23	026600	#49-4953								
AA24	026620	#49-4960	49-4970							
AA25	026640	#49-4965	49-4968							
AA26	026646	49-4966	#49-4968							
AA27	026652	49-4961	#49-4970							
AA3	026330	#49-4891								
AA4	026346	#49-4895	49-4905							
AA5	026366	#49-4900	49-4903							
AA6	026374	49-4901	#49-4903							
AA7	026402	49-4896	#49-4905							
ABASE	= 000000	20-1427	20-1427							
ACDW1	= 000000	20-1427	20-1427							
ACDW2	= 000000	20-1427	20-1427							
ACPUOP	= 000000	20-1427	20-1427							
ACO	= %000000	#19-1417	*30-1981	30-2020	*31-2349	*31-2355	31-2360	*31-2398	*31-2405	31-2418
		*32-2523	32-2532	*32-2566	32-2575	*33-2811	33-2811	*33-2811	33-2811	*33-2811
		33-2811	*33-2812	33-2812	*33-2812	33-2812	*33-2812	33-2812	*33-2813	33-2813
		*33-2813	33-2813	*33-2814	33-2814	*33-2814	33-2814	*33-2815	33-2815	*33-2815
		33-2815	*33-2816	33-2816	*33-2816	33-2816	*33-2817	33-2817	*33-2817	33-2817



SYMBOL CROSS REFERENCE  
SYMBOL VALUE

REFERENCES

SYMBOL	VALUE	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES
		*33-2818	33-2818	*33-2818	33-2818	*33-2819	33-2819	*33-2819	33-2819
		33-2820	*33-2820	33-2820	*33-2821	33-2821	*33-2821	33-2821	*33-2822
		*33-2822	33-2822	*34-3000	34-3000	*34-3001	34-3001	*34-3002	34-3002
		34-3003	*34-3004	34-3004	*34-3010	34-3010	*34-3011	34-3011	*34-3012
		*34-3013	34-3013	*34-3014	34-3014	*34-3025	34-3025	*34-3026	34-3026
		34-3027	*34-3028	34-3028	*34-3029	34-3029	*35-3096	*35-3106	35-3112
		*35-3136	35-3142	*36-3257	*36-3263	36-3268	*37-3356	*37-3362	37-3367
		*38-3466	38-3473	*39-3552	*39-3557	39-3571	*40-3655	*40-3662	40-3667
		*41-3803	41-3808	*42-3936	*42-3942	42-3946	*43-4057	*43-4065	43-4069
		*44-4219	*44-4235	*44-4242	*44-4254	*44-4260	*44-4272	*44-4277	*44-4303
		*45-4416	*45-4419	45-4423	*45-4444	*45-4449	45-4453	*45-4472	*45-4475
		*45-4500	*45-4505	45-4509	*46-4559	*46-4562	46-4566	*46-4582	*46-4585
		*46-4605	*46-4608	46-4612	*46-4628	*46-4631	46-4635	*47-4712	*47-4714
		*48-4807	*48-4809	48-4813	*49-4886	*49-4889	49-4892	*49-4913	*49-4916
		*49-4947	*49-4951	49-4957	*50-5062	*50-5064	50-5067	*50-5090	*50-5092
		*50-5116	*50-5120	50-5124	*50-5145	*50-5149	50-5153	*50-5176	*50-5178
		50-5204	*50-5206	50-5209	*51-5322	*51-5324	51-5327	*51-5344	*51-5346
		*51-5370	*51-5375	51-5379	*51-5393	*51-5397	51-5401	*51-5424	*51-5426
		*51-5445	*51-5447	51-5450	*52-5532	*52-5534	52-5537	*52-5561	*52-5563
		*52-5583	*52-5585	52-5588	*52-5605	*52-5607	52-5610	*52-5633	*52-5635
		*52-5661	*52-5663	52-5666	*52-5690	*52-5692	52-5695	*53-5756	*53-5758
		*53-5785	*53-5787	53-5790	*54-5840	*54-5842	54-5845	*54-5862	*54-5864
AC1	=%000001	#19-1418	*31-2346	31-2355	*31-2395	31-2405	*32-2526	*32-2532	32-2537
		*32-2575	32-2589	*33-2813	33-2813	*33-2814	33-2814	*34-3000	*34-3010
AC2	=%000002	#19-1419	*33-2815	33-2815	*33-2816	33-2816	*34-3001	*34-3011	34-3026
AC3	=%000003	#19-1420	*33-2817	33-2817	*33-2818	33-2818	*34-3002	*34-3012	34-3027
AC4	=%000004	#19-1421	*33-2819	33-2819	*33-2820	33-2820	*34-3003	*34-3013	34-3028
AC5	=%000005	#19-1422	*33-2821	33-2821	*33-2822	33-2822	*34-3004	*34-3014	34-3029
AC6	=%000006	#19-1423	35-3136						
AC7	=%000007	#19-1424	35-3106						
ADDWC	= 000000	20-1427	20-1427						
ADDW1	= 000000	20-1427	20-1427						
ADDW10	= 000000	20-1427	20-1427						
ADDW11	= 000000	20-1427	20-1427						
ADDW12	= 000000	20-1427	20-1427						
ADDW13	= 000000	20-1427	20-1427						
ADDW14	= 000000	20-1427	20-1427						
ADDW15	= 000000	20-1427	20-1427						
ADDW2	= 000000	20-1427	20-1427						
ADDW3	= 000000	20-1427	20-1427						
ADDW4	= 000000	20-1427	20-1427						
ADDW5	= 000000	20-1427	20-1427						
ADDW6	= 000000	20-1427	20-1427						
ADDW7	= 000000	20-1427	20-1427						
ADDW8	= 000000	20-1427	20-1427						
ADDW9	= 000000	20-1427	20-1427						
ADEVCT	= 000000	20-1427	20-1427						
ADEVN	= 000000	20-1427	20-1427						
ADONE	004670	25-1515	25-1519	25-1527	25-1535	25-1548	25-1561	#25-1567	
AENV	= 000000	20-1427	20-1427						
AENVN	= 000000	20-1427	20-1427						
AERFLG	004560	*25-1465	*25-1496	25-1502	25-1514	#25-1529	75-6796		







SYMBOL	CROSS REFERENCE	VALUE	REFERENCES							
BBER5		032242	51-5411	#51-5491						
BBER6		032260	51-5413	#51-5494						
BBER7		032314	51-5434	#51-5500						
BBER8		032332	51-5455	#51-5503						
BBPAT0		032360	51-5369	51-5380	51-5485	51-5486	#51-5507			
BBPAT1		032370	51-5323	51-5345	51-5374	51-5396	51-5425	51-5446	51-5472	51-5481
			51-5496	51-5501	#51-5508					51-5487
BBPAT2		032400	51-5321	51-5328	51-5470	51-5471	#51-5509			
BBPAT3		032410	51-5392	51-5407	51-5491	51-5494	#51-5510			
BBPAT4		032420	51-5343	51-5355	51-5476	51-5479	#51-5511			
BBPAT5		032430	51-5423	51-5500	#51-5512					
BBPAT6		032440	51-5444	51-5451	51-5503	51-5504	#51-5513			
BBP10		032460	51-5350	51-5477	51-5480	#51-5515				
BBP11		032470	51-5430	#51-5516						
BBP7		032450	51-5402	51-5492	51-5495	#51-5514				
BB1		031150	#51-5315							
BB10		031334	#51-5352	51-5362						
BB11		031354	#51-5357	51-5360						
BB12		031364	51-5358	#51-5360						
BB13		031372	51-5353	#51-5362						
BB14		031404	51-5364	#51-5367						
BB15		031440	51-5368	#51-5375						
BB16		031464	#51-5382	51-5385						
BB17		031474	51-5383	#51-5385						
BB2		031212	51-5319	#51-5324						
BB20		031506	51-5387	#51-5390						
BB21		031542	51-5391	#51-5397						
BB22		031566	#51-5404	51-5414						
BB23		031606	#51-5409	51-5412						
BB24		031616	51-5410	#51-5412						
BB25		031624	51-5405	#51-5414						
BB26		031636	51-5416	#51-5419						
BB27		031672	51-5420	#51-5426						
BB3		031216	#51-5326							
BB30		031714	#51-5432	51-5435						
BB31		031724	51-5433	#51-5435						
BB32		031736	51-5437	#51-5440						
BB33		031772	51-5441	#51-5447						
BB34		032014	#51-5453	51-5456						
BB35		032024	51-5454	#51-5456						
BB4		031234	#51-5330	51-5333						
BB5		031244	51-5331	#51-5333						
BB6		031256	51-5336	#51-5339						
BB7		031312	51-5342	#51-5346						
BDONE		004776	26-1591	#26-1610						
BERR		004732	26-1588	#26-1593						
BERR1		004762	26-1597	#26-1604						
BITSTS		014072	#33-2834	33-2914	33-2922	33-2938	33-2940			
BIT0	=	000001	#19-1415							
BIT0C	=	000001	#19-1415	19-1415	56-5900	56-5900	57-5902	57-5902		
BIT01	=	000002	#19-1415	19-1415						
BIT02	=	000004	#19-1415	19-1415						



SYMBOL	CROSS REFERENCE	VALUE	REFERENCES											
BIT03	=	000010	#19-1415	19-1415										
BIT04	=	000020	#19-1415	19-1415										
BIT05	=	000040	#19-1415	19-1415										
BIT06	=	000100	#19-1415	19-1415										
BIT07	=	000200	#19-1415	19-1415										
BIT08	=	000400	#19-1415	19-1415	56-5900									
BIT09	=	001000	#19-1415	19-1415	56-5900	57-5902	57-5902							
BIT1	=	000002	#19-1415											
BIT10	=	002000	#19-1415	57-5902										
BIT11	=	004000	#19-1415	56-5900										
BIT12	=	010000	#19-1415	55-5898										
BIT13	=	020000	#19-1415	57-5902										
BIT14	=	040000	#19-1415	56-5900										
BIT15	=	100000	#19-1415											
BIT2	=	000004	#19-1415											
BIT3	=	000010	#19-1415											
BIT4	=	000020	#19-1415											
BIT5	=	000040	#19-1415											
BIT6	=	000100	#19-1415											
BIT7	=	000200	#19-1415											
BIT8	=	000400	#19-1415											
BIT9	=	001000	#19-1415											
BPTVEC	=	000014	#19-1415											
B1		004706	26-1576	#26-1579	26-1590									
B2		004710	#26-1582	26-1595										
B3		004726	#26-1590	26-1602	26-1608									
CCDATO		031004	50-5066	50-5072	50-5094	50-5100	50-5123	50-5129	50-5152	50-5158	50-5180			
			50-5186	50-5208	50-5214	50-5238	50-5244	50-5253	50-5259	50-5265	50-5273			
			#50-5289											
CCDONE		031144	50-5226	50-5230	50-5234	50-5240	50-5246	50-5255	50-5261	50-5267	50-5276			
			#50-5301											
CCERO		030346	50-5083	50-5111	50-5197	50-5225	#50-5227							
CCER1		030402	50-5077	#50-5235										
CCER10		030714	50-5191	#50-5277										
CCER11		030732	50-5193	#50-5280										
CCER12		030750	50-5219	#50-5283										
CCER13		030766	50-5221	#50-5286										
CCER2		030440	50-5079	#50-5241										
CCER22		030454	#50-5243	50-5288										
CCER3		030476	50-5105	#50-5247										
CCER4		030514	50-5107	#50-5250										
CCER44		030530	#50-5252	50-5282										
CCER5		030552	50-5134	#50-5256										
CCER50		030416	#50-5237	50-5249	50-5279	50-5285								
CCER55		030566	#50-5258	50-5270										
CCER6		030606	50-5136	#50-5262										
CCER7		030642	50-5163	#50-5268										
CCER8		030660	50-5165	#50-5271										
CCER90		030364	50-5140	50-5169	#50-5231									
CCPO		031014	50-5061	50-5073	50-5089	50-5115	50-5175	50-5203	50-5237	50-5243	50-5252			
			50-5258	50-5264	50-5274	#50-5290								
CCP1		031024	50-5091	50-5101	50-5247	50-5250	#50-5291							



SYMBOL	CROSS REFERENCE VALUE	REFERENCES	CREF	V01
CCP10	031114	50-5154	50-5269	50-5272 #50-5298
CCP11	031124	50-5182	50-5278	50-5281 #50-5299
CCP12	031134	#50-5300		
CCP2	031034	50-5063	50-5068	50-5235 50-5236 50-5241 50-5242 #50-5292
CCP3	031044	50-5130	50-5144	50-5177 50-5187 50-5277 50-5280 #50-5293
CCP4	031054	50-5205	50-5210	50-5215 50-5283 50-5284 50-5286 50-5287 #50-5294
CCP5	031064	50-5148	50-5159	50-5268 50-5271 #50-5295
CCP6	031074	50-5119	50-5125	50-5256 50-5257 50-5262 50-5263 #50-5296
CCP7	031104	50-5096	50-5248	50-5251 #50-5297
CC1	027332	#50-5057		
CC10	027556	#50-5103	50-5106	
CC11	027566	50-5104	#50-5106	
CC12	027574	50-5099	#50-5108	
CC13	027606	50-5110	#50-5113	
CC14	027642	50-5114	#50-5120	
CC15	027666	#50-5127	50-5137	
CC16	027706	#50-5132	50-5135	
CC17	027716	50-5133	#50-5135	
CC18	027724	50-5128	#50-5137	
CC.9	027736	50-5139	#50-5142	
CC2	027366	50-5060	#50-5064	
CC20	027772	50-5143	#50-5149	
CC21	030016	#50-5156	50-5166	
CC22	030036	#50-5161	50-5164	
CC23	030046	-5162	#50-5164	
CC24	030054	50-5157	#50-5166	
CC25	030066	50-5168	#50-5171	
CC26	030122	50-5174	#50-5178	
CC27	030144	#50-5184	50-5194	
CC28	030164	#50-5189	50-5192	
CC29	030174	50-5190	#50-5192	
CC3	027410	#50-5070	50-5080	
CC30	030202	50-5185	#50-5194	
CC31	030214	50-5196	#50-5199	
CC32	030250	50-5202	#50-5206	
CC33	030272	#50-5212	50-5222	
CC34	030312	#50-5217	50-5220	
CC35	030322	50-5218	#50-5220	
CC36	030330	50-5213	#50-5222	
CC37	030342	50-5224	#50-5226	
CC4	027430	#50-5075	50-5078	
CC5	027440	50-5076	#50-5078	
CC6	027446	50-5071	#50-5080	
CC7	027460	50-5082	#50-5085	
CC8	027514	50-5088	#50-5092	
CC9	027536	#50-5098	50-5108	
CDONE	005742	27-1736	#27-1802	
CERR1	005476	27-1636	27-1677	27-1693 27-1734 #27-1739
CERR2	005574	27-1649	27-1664	27-1706 27-1721 #27-1761
CERR3	005674	27-1753	27-1759	27-1777 27-1783 #27-1786
CERR4	005716	27-1742	27-1766	#27-1793
CFCC1	043223	#71-6151		







SYMBOL CROSS REFERENCE  
SYMBOL VALUE

REFERENCES

SYMBOL	CROSS REFERENCE VALUE	REFERENCES	37-3393	39-3616	40-3718	40-3721	41-3858	41-3861	42-3970	43-4102
		36-3294	37-3393	39-3616	40-3718	40-3721	41-3858	41-3861	42-3970	43-4102
		#68-6086	70-6117							
		25-1488	25-1542	#69-6095	70-6118					
CPTWO	042136	#19-1415	59-5906	59-5906						
CR	= 000015	#19-1414	#19-1415	24-1444	24-1444	33-2954	33-2954	59-5906	59-5906	71-6125
CRLF	= 000200	71-6125	71-6129	71-6130	71-6131	71-6132	71-6134	71-6136	71-6138	71-6139
		71-6141	71-6143	71-6144	71-6159	71-6190	71-6191	71-6192	71-6193	71-6194
		71-6195	72-6219	72-6222	72-6225	72-6226	72-6230	72-6233	72-6240	72-6241
		72-6246	72-6249	72-6252	72-6254	72-6256	72-6258	72-6260	72-6266	72-6267
		72-6289	72-6289	72-6289	72-6292	72-6292	72-6292	72-6294	72-6294	72-6294
		72-6297	72-6297	72-6298	72-6301	72-6301	72-6302	72-6305	72-6329	72-6329
		72-6329	72-6330	72-6332	72-6334	72-6334	72-6334	72-6335	72-6337	72-6339
		72-6340	72-6343	72-6344	72-6362	72-6364	72-6368	72-6369	72-6371	72-6373
		72-6375	72-6376	72-6378	72-6380	72-6381	72-6384	72-6384	72-6386	72-6388
		72-6449	72-6449	72-6449	72-6450	72-6450	72-6450	72-6451	72-6451	72-6451
		72-6452	72-6452	72-6452	72-6453	72-6453	72-6454	72-6454	72-6454	72-6459
		72-6460	72-6461	72-6462	72-6462	72-6463	72-6463	72-6464	72-6464	72-6465
		72-6465	72-6466	72-6466	72-6467	72-6469	72-6470	72-6471	72-6472	72-6473
		72-6477	73-6482	73-6509	73-6513	73-6550	73-6573	73-6575	73-6575	73-6580
		73-6580	73-6581	73-6597	73-6613					
C1	005016	27-1621	#27-1623							
C15	005044	27-1628	#27-1630							
C2	005070	27-1638	#27-1639							
C25	005104	27-1642	#27-1643							
C3	005132	27-1651	#27-1652							
C35	005162	27-1657	#27-1658							
C4	005210	27-1666	#27-1667							
C45	005222	27-1669	#27-1671							
C5	005250	27-1679	#27-1680							
C55	005276	27-1685	#27-1687							
C6	005322	27-1695	#27-1696							
C65	005336	27-1698	#27-1700							
C7	005364	27-1708	#27-1709							
C75	005414	27-1713	#27-1715							
C8	005442	27-1723	#27-1724							
C85	005454	27-1726	#27-1728							
DDDATO	034400	52-5536	52-5542	52-5565	52-5587	52-5609	52-5615	52-5637	52-5643	52-5665
		52-5671	52-5694	52-5700	52-5718	52-5719	52-5720	52-5721	52-5722	52-5723
		52-5724	52-5725	52-5726	52-5727	52-5728	52-5729	#52-5730		
DDDONE	034530	52-5713	52-5717	52-5718	52-5719	52-5720	52-5721	52-5722	52-5723	52-5724
		52-5725	52-5726	52-5727	52-5728	52-5729	#52-5741			
DDERO	033612	52-5554	52-5576	52-5598	52-5626	52-5654	52-5683	52-5712	#52-5714	
DDER1	033630	52-5547	#52-5718							
DDER10	034246	52-5678	#52-5727							
DDER11	034304	52-5705	#52-5728							
DDER12	034342	52-5707	#52-5729							
DDER2	033666	52-5549	#52-5719							
DDER3	033724	52-5571	#52-5720							
DDER4	033762	52-5593	#52-5721							
DDER5	034020	52-5620	#52-5722							
DDER6	034056	52-5622	#52-5723							
DDER7	034114	52-5648	#52-5724							



SYMBOL	CROSS REFERENCE	VALUE	REFERENCES
DDER8		034152	52-5650 #52-5725
DDER9		034210	52-5676 #52-5726
DDISP	=	177570	#19-1415 20-1427 24-1443
DDONE		006210	28-1852 #28-1894
DDP0		034410	52-5567 52-5589 52-5720 52-5721 #52-5731
DDP1		034420	52-5531 52-5533 52-5562 52-5582 52-5718 52-5719 52-5719 52-5720
			52-5721 #52-5732
DDP2		034430	52-5560 52-5584 52-5720 52-5721 #52-5733
DDP3		034440	52-5604 52-5634 52-5722 52-5723 52-5724 52-5725 #52-5734
DDP4		034450	52-5543 52-5660 52-5691 52-5726 52-5727 52-5728 52-5729 #52-5735
DDP5		034460	52-5662 52-5689 52-5726 52-5727 52-5728 52-5729 #52-5736
DDP6		034470	52-5606 52-5632 52-5722 52-5723 52-5724 52-5725 #52-5737
DDP7		034500	52-5611 52-5639 52-5672 52-5701 52-5722 52-5723 52-5724 52-5725 #52-5738
DDP8		034510	52-5616 52-5644 52-5667 52-5696 52-5726 52-5727 52-5728 52-5729 #52-5739
DDP9		034520	52-5538 52-5718 52-5719 #52-5740
DD1		032504	#52-5527
DD10		032714	#52-5569 52-5572
DD11		032724	52-5570 #52-5572
DD12		032742	52-5575 #52-5578
DD13		032776	52-5581 #52-5585
DD14		033020	#52-5591 52-5594
DD15		033030	52-5592 #52-5594
DD16		033046	52-5597 #52-5600
DD17		033102	52-5603 #52-5607
DD18		033124	#52-5613 52-5623
DD19		033144	#52-5618 52-5621
DD2		032540	52-5530 #52-5534
DD20		033154	52-5619 #52-5621
DD21		033162	52-5614 #52-5623
DD22		033174	52-5625 #52-5628
DD23		033230	52-5631 #52-5635
DD24		033252	#52-5641 52-5651
DD25		033272	#52-5646 52-5649
DD26		033302	52-5647 #52-5649
DD27		033310	52-5642 #52-5651
DD3		032562	#52-5540 52-5550
DD30		033322	52-5653 #52-5656
DD31		033356	52-5659 #52-5663
DD32		033400	#52-5669 52-5679
DD33		033420	#52-5674 52-5677
DD34		033430	52-5675 #52-5677
DD35		033436	52-5670 #52-5679
DD36		033454	52-5682 #52-5685
DD37		033510	52-5688 #52-5692
DD38		033532	#52-5698 52-5708
DD39		033552	#52-5703 52-5706
DD4		032602	#52-5545 52-5548
DD40		033562	52-5704 #52-5706
DD41		033570	52-5699 #52-5708
DD42		033606	52-5711 #52-5713
DD5		032612	52-5546 #52-5548
DD6		032620	52-5541 #52-5550



SYMBOL CROSS REFERENCE		REFERENCES										
SYMBOL	VALUE											
DD7	032636	52-5553	#52-5556									
DD8	032672	52-5559	#52-5563									
DERR1	006070	28-1828	#28-1856									
DERR2	006164	28-1827	#28-1884									
DF1	070570	21-1434	#74-6652									
DF10	= 070620	21-1434	#74-6659									
DF100	= 071415	21-1434	#74-6715									
DF101	= 071442	21-1434	#74-6716									
DF102	= 071464	21-1434	#74-6717									
DF103	= 071442	21-1434	#74-6718									
DF104	= 071415	21-1434	#74-6719									
DF105	= 071442	21-1434	#74-6720									
DF106	= 071464	21-1434	#74-6721									
DF107	= 071442	21-1434	#74-6722									
DF11	= 070620	21-1434	#74-6660									
DF110	= 071415	21-1434	#74-6723									
DF111	= 071473	21-1434	#74-6724	74-6725	74-6726	74-6727	74-6728	74-6729	74-6737	74-6738		
DF112	= 071473	21-1434	#74-6725									
DF113	= 071473	21-1434	#74-6726									
DF114	= 071473	21-1434	#74-6727									
DF115	= 071473	21-1434	#74-6728									
DF116	= 071473	21-1434	#74-6729									
DF117	= 070610	21-1434	#74-6730									
DF12	070634	21-1434	#74-6661									
DF120	= 070610	21-1434	#74-6731									
DF121	= 071327	21-1434	#74-6732									
DF122	= 071344	21-1434	#74-6733									
DF123	= 071370	21-1434	#74-6734									
DF124	= 071477	21-1434	#74-6735	74-6736								
DF125	= 071477	21-1434	#74-6736									
DF126	= 071473	21-1434	#74-6737									
DF127	= 071473	21-1434	#74-6738									
DF13	070666	21-1434	#74-6662									
DF130	= 070610	21-1434	#74-6739									
DF131	= 071530	21-1434	#74-6740	74-6741								
DF132	= 071530	21-1434	#74-6741									
DF133	= 071550	21-1434	#74-6742	74-6743	74-6744	74-6745	74-6750	74-6751	74-6754	74-6755		
				74-6757	74-6761	74-6762	74-6763	74-6765	74-6769	74-6770	74-6773	74-6774
				74-6777	74-6779	74-6780	74-6781	74-6782	74-6783	74-6785	74-6786	74-6787
				74-6791								
DF134	= 071550	21-1434	#74-6743									
DF135	= 071550	21-1434	#74-6744									
DF136	= 071550	21-1434	#74-6745									
DF137	= 071574	21-1434	#74-6746	74-6747	74-6748	74-6749	74-6752	74-6753	74-6756	74-6758		
DF14	= 070620	21-1434	#74-6663									
DF140	= 071574	21-1434	#74-6747									
DF141	= 071574	21-1434	#74-6748									
DF142	= 071574	21-1434	#74-6749									
DF143	= 071550	21-1434	#74-6750									
DF144	= 071550	21-1434	#74-6751									
DF145	= 071574	21-1434	#74-6752									
DF146	= 071574	21-1434	#74-6753									



SYMBOL CROSS REFERENCE

SYMBOL VALUE REFERENCES

SYMBOL	CROSS REFERENCE VALUE	REFERENCES							
DF147	= 071550	21-1434 #74-6754							
DF15	= 070620	21-1434 #74-6664							
DF150	= 071550	21-1434 #74-6755							
DF151	= 071574	21-1434 #74-6756							
DF152	= 071550	21-1434 #74-6757							
DF153	= 071574	21-1434 #74-6758							
DF154	071604	21-1434 #74-6759	74-6760						
DF155	= 071604	21-1434 #74-6760							
DF156	= 071550	21-1434 #74-6761							
DF157	= 071550	21-1434 #74-6762							
DF16	070702	21-1434 #74-6665							
DF160	= 071550	21-1434 #74-6763							
DF161	071610	21-1434 #74-6764	74-6766	74-6771	74-6772	74-6775	74-6776		
DF162	= 071550	21-1434 #74-6765							
DF163	= 071610	21-1434 #74-6766							
DF164	= 070610	21-1434 #74-6767							
DF165	= 070610	21-1434 #74-6768							
DF166	= 071550	21-1434 #74-6769							
DF167	= 071550	21-1434 #74-6770							
DF17	= 070610	21-1434 #74-6666							
DF170	= 071610	21-1434 #74-6771							
DF171	= 071610	21-1434 #74-6772							
DF172	= 071550	21-1434 #74-6773							
DF173	= 071550	21-1434 #74-6774							
DF174	= 071610	21-1434 #74-6775							
DF175	= 071610	21-1434 #74-6776							
DF176	= 071550	21-1434 #74-6777							
DF177	071634	21-1434 #74-6778							
DF2	070600	21-1434 #74-6653							
DF20	070711	21-1434 #74-6667							
DF200	= 071550	21-1434 #74-6779							
DF201	= 071550	21-1434 #74-6780							
DF202	= 071550	21-1434 #74-6781							
DF203	= 071550	21-1434 #74-6782							
DF204	= 071550	21-1434 #74-6783							
DF205	= 070610	21-1434 #74-6784							
DF206	= 071550	21-1434 #74-6785							
DF207	= 071550	21-1434 #74-6786							
DF21	070721	21-1434 #74-6668	74-6706	74-6711					
DF210	= 071550	21-1434 #74-6787							
DF211	071637	21-1434 #74-6788	74-6789	74-6790					
DF212	= 071637	21-1434 #74-6789							
DF213	= 071637	21-1434 #74-6790							
DF214	= 071550	21-1434 #74-6791							
DF22	070725	21-1434 #74-6669							
DF23	070737	21-1434 #74-6670							
DF24	070747	21-1434 #74-6671							
DF25	070756	21-1434 #74-6672							
DF26	071002	21-1434 #74-6673	74-6676						
DF27	071027	21-1434 #74-6674	74-6677						
DF3	070610	21-1434 #74-6654	74-6655	74-6666	74-6730	74-6731	74-6739	74-6767	74-6768
		74-6784							



SYMBOL	CROSS REFERENCE	REFERENCES								
SYMBOL	VALUE									
DF30	071047	21-1434	#74-6675							
DF31	= 071002	21-1434	#74-6676							
DF32	= 071027	21-1434	#74-6677							
DF33	071063	21-1434	#74-6678							
DF34	071111	21-1434	#74-6679	74-6680	74-6681	74-6682				
DF35	= 071111	21-1434	#74-6680							
DF36	= 071111	21-1434	#74-6681							
DF37	= 071111	21-1434	#74-6682							
DF4	= 070610	21-1434	#74-6655							
DF40	071135	21-1434	#74-6683							
DF41	071163	21-1434	#74-6684							
DF42	071177	21-1434	#74-6685	74-6686						
DF43	= 071177	21-1434	#74-6686							
DF44	071225	21-1434	#74-6687							
DF45	071250	21-1434	#74-6688							
DF46	071275	21-1434	#74-6689							
DF47	071327	21-1434	#74-6690	74-6693	74-6696	74-6732				
DF5	070620	21-1434	#74-6656	74-6657	74-6658	74-6659	74-6660			
DF50	071344	21-1434	#74-6691	74-6733						
DF51	071370	21-1434	#74-6692	74-6734						
DF52	= 071327	21-1434	#74-6693							
DF53	071406	21-1434	#74-6694	74-6697	74-6699					
DF54	071415	21-1434	#74-6695	74-6698	74-6700	74-6701	74-6702	74-6705	74-6710	74-6715
			74-6719	74-6723						
DF55	= 071327	21-1434	#74-6696							
DF56	= 071406	21-1434	#74-6697							
DF57	= 071415	21-1434	#74-6698							
DF6	= 070620	21-1434	#74-6657	74-6663	74-6664					
DF60	= 071406	21-1434	#74-6699							
DF61	= 071415	21-1434	#74-6700							
DF62	= 071415	21-1434	#74-6701							
DF63	= 071415	21-1434	#74-6702							
DF64	071433	21-1434	#74-6703	74-6704						
DF65	= 071433	21-1434	#74-6704							
DF66	= 071415	21-1434	#74-6705							
DF67	= 070721	21-1434	#74-6706							
DF7	= 070620	21-1434	#74-6658							
DF70	071442	21-1434	#74-6707	74-6708	74-6712	74-6713	74-6716	74-6718	74-6720	74-6722
DF71	= 071442	21-1434	#74-6708							
DF72	071464	21-1434	#74-6709	74-6714	74-6717	74-6721				
DF73	= 071415	21-1434	#74-6710							
DF74	= 070721	21-1434	#74-6711							
DF75	= 071442	21-1434	#74-6712							
DF76	= 071442	21-1434	#74-6713							
DF77	= 071464	21-1434	#74-6714							
DH1	065163	21-1434	#73-6484							
DH10	= 065533	21-1434	#73-6495							
DH100	= 067035	21-1434	#73-6563							
DH101	= 065533	21-1434	#73-6564							
DH102	= 066425	21-1434	#73-6565							
DH103	= 066363	21-1434	#73-6566							
DH104	= 067035	21-1434	#73-6567							



SYMBOL CROSS REFERENCE		REFERENCES								
SYMBOL	VALUE									
DH105	= 065533	21-1434	#73-6568							
DH106	= 066425	21-1434	#73-6569							
DH107	= 066363	21-1434	#73-6570							
DH11	= 065533	21-1434	#73-6496							
DH110	= 067035	21-1434	#73-6571							
DH111	= 066363	21-1434	#73-6572							
DH112	= 067233	21-1434	#73-6573	73-6578						
DH113	= 066363	21-1434	#73-6577							
DH114	= 067233	21-1434	#73-6578							
DH115	= 067452	21-1434	#73-6579	73-6583						
DH116	= 067452	21-1434	#73-6583							
DH117	= 065653	21-1434	#73-6584							
DH12	= 000000	21-1434	#73-6497							
DH120	= 067736	21-1434	#73-6585	73-6594						
DH121	= 065533	21-1434	#73-6587							
DH122	= 066763	21-1434	#73-6588							
DH123	= 066763	21-1434	#73-6589							
DH124	= 065653	21-1434	#73-6590							
DH125	= 065653	21-1434	#73-6591							
DH126	= 065533	21-1434	#73-6592							
DH127	= 067035	21-1434	#73-6593							
DH13	= 000000	21-1434	#73-6498							
DH130	= 067736	21-1434	#73-6594							
DH131	= 067035	21-1434	#73-6595							
DH132	= 067035	21-1434	#73-6596							
DH133	= 070026	21-1434	#73-6597	73-6599	73-6600	73-6601	73-6608	73-6609	73-6616	
DH134	= 070026	21-1434	#73-6599							
DH135	= 070026	21-1434	#73-6600							
DH136	= 070026	21-1434	#73-6601							
DH137	= 070136	21-1434	#73-6602	73-6605	73-6606	73-6607	73-6610	73-6611	73-6615	73-6617
DH14	= 065533	21-1434	#73-6499							
DH140	= 070136	21-1434	#73-6605							
DH141	= 070136	21-1434	#73-6606							
DH142	= 070136	21-1434	#73-6607							
DH143	= 070026	21-1434	#73-6608							
DH144	= 070026	21-1434	#73-6609							
DH145	= 070136	21-1434	#73-6610							
DH146	= 070136	21-1434	#73-6611							
DH147	= 067035	21-1434	#73-6612							
DH15	= 065533	21-1434	#73-6500							
DH150	= 070326	21-1434	#73-6613							
DH151	= 070136	21-1434	#73-6615							
DH152	= 070026	21-1434	#73-6616							
DH153	= 070136	21-1434	#73-6617							
DH154	= 070417	21-1434	#73-6618	73-6619						
DH155	= 070417	21-1434	#73-6619							
DH156	= 067035	21-1434	#73-6620							
DH157	= 067035	21-1434	#73-6621							
DH16	= 065573	21-1434	#73-6501							
DH160	= 067035	21-1434	#73-6622							
DH161	= 067035	21-1434	#73-6623							
DH162	= 067035	21-1434	#73-6624							



SYMBOL CROSS REFERENCE

SYMBOL	VALUE	REFERENCES						
DH163	= 067035	21-1434	#73-6625					
DH164	= 065653	21-1434	#73-6626					
DH165	= 065653	21-1434	#73-6627					
DH166	= 067035	21-1434	#73-6628					
DH167	= 067035	21-1434	#73-6629					
DH17	= 065653	21-1434	#73-6503	73-6584	73-6590	73-6591	73-6626	73-6627 73-6643
DH170	= 067035	21-1434	#73-6630					
DH171	= 067035	21-1434	#73-6631					
DH172	= 067035	21-1434	#73-6632					
DH173	= 067035	21-1434	#73-6633					
DH174	= 067035	21-1434	#73-6634					
DH175	= 067035	21-1434	#73-6635					
DH176	= 067035	21-1434	#73-6636					
DH177	= 070457	21-1434	#73-6637					
DH2	= 065253	21-1434	#73-6486					
DH20	= 065743	21-1434	#73-6505					
DH200	= 067035	21-1434	#73-6638					
DH201	= 067035	21-1434	#73-6639					
DH202	= 067035	21-1434	#73-6640					
DH203	= 067035	21-1434	#73-6641					
DH204	= 067035	21-1434	#73-6642					
DH205	= 065653	21-1434	#73-6643					
DH206	= 067035	21-1434	#73-6644					
DH207	= 067035	21-1434	#73-6645					
DH21	= 065533	21-1434	#73-6507					
DH210	= 067035	21-1434	#73-6646					
DH211	= 070523	21-1434	#73-6647					
DH212	= 065533	21-1434	#73-6648					
DH213	= 065533	21-1434	#73-6649					
DH214	= 067035	21-1434	#73-6650					
DH22	= 066031	21-1434	#73-6508					
DH23	= 066066	21-1434	#73-6509					
DH24	= 066224	21-1434	#73-6512					
DH25	= 066363	21-1434	#73-6514	73-6556	73-6561	73-6566	73-6570	73-6572 73-6577
DH26	= 066425	21-1434	#73-6515	73-6517	73-6519	73-6520	73-6540	73-6543 73-6545 73-6557
		73-6562	73-6565	73-6569				
DH27	= 066425	21-1434	#73-6517					
DH3	= 065346	21-1434	#73-6488	75-6819	75-6849			
DH30	= 000000	21-1434	#73-6518					
DH31	= 066425	21-1434	#73-6519					
DH32	= 066425	21-1434	#73-6520					
DH33	= 066513	21-1434	#73-6521	73-6523	73-6524	73-6525	73-6526	73-6527
DH34	= 066513	21-1434	#73-6523					
DH35	= 066513	21-1434	#73-6524					
DH36	= 066513	21-1434	#73-6525					
DH37	= 066513	21-1434	#73-6526					
DH4	= 065437	21-1434	#73-6490					
DH40	= 066513	21-1434	#73-6527					
DH41	= 000000	21-1434	#73-6528					
DH42	= 066616	21-1434	#73-6529	73-6531				
DH43	= 066616	21-1434	#73-6531					
DH44	= 000000	21-1434	#73-6532					







SYMBOL	CROSS REFERENCE VALUE	REFERENCES	75-6906	75-6907	75-6908	75-6909	75-6910	75-6920	75-6921
DT111	= 073446	21-1434 #75-6905							
DT112	= 073446	21-1434 #75-6906							
DT113	= 073446	21-1434 #75-6907							
DT114	= 073446	21-1434 #75-6908							
DT115	= 073446	21-1434 #75-6909							
DT116	= 073446	21-1434 #75-6910							
DT117	= 071712	21-1434 #75-6911							
DT12	= 072010	21-1434 #75-6806							
DT120	= 071712	21-1434 #75-6912							
DT121	= 073172	21-1434 #75-6913							
DT122	= 072576	21-1434 #75-6914							
DT123	= 073224	21-1434 #75-6915							
DT124	= 073460	21-1434 #75-6916	75-6919						
DT125	= 073460	21-1434 #75-6919							
DT126	= 073446	21-1434 #75-6920							
DT127	= 073446	21-1434 #75-6921							
DT13	= 072076	21-1434 #75-6810							
DT130	= 071712	21-1434 #75-6922							
DT131	= 073544	21-1434 #75-6923	75-6925						
DT132	= 073544	21-1434 #75-6925							
DT133	= 073606	21-1434 #75-6926	75-6929	75-6930	75-6931	75-6936	75-6937	75-6940	75-6941
		75-6943 #75-6947	75-6948	75-6949	75-6950	75-6951	75-6952	75-6955	75-6956
		75-6957 #75-6958	75-6959	75-6960	75-6961	75-6962	75-6963	75-6965	75-6966
		75-6967 #75-6968	75-6969	75-6971	75-6972	75-6973	75-6977		
DT134	= 073606	21-1434 #75-6929							
DT135	= 073606	21-1434 #75-6930							
DT136	= 073606	21-1434 #75-6931							
DT137	= 073660	21-1434 #75-6932	75-6933	75-6934	75-6935	75-6938	75-6939	75-6942	75-6944
DT14	= 071766	21-1434 #75-6812							
DT140	= 073660	21-1434 #75-6933							
DT141	= 073660	21-1434 #75-6934							
DT142	= 073660	21-1434 #75-6935							
DT143	= 073606	21-1434 #75-6936							
DT144	= 073606	21-1434 #75-6937							
DT145	= 073660	21-1434 #75-6938							
DT146	= 073660	21-1434 #75-6939							
DT147	= 073606	21-1434 #75-6940							
DT15	= 071766	21-1434 #75-6813							
DT150	= 073606	21-1434 #75-6941							
DT151	= 073660	21-1434 #75-6942							
DT152	= 073606	21-1434 #75-6943							
DT153	= 073660	21-1434 #75-6944							
DT154	= 073700	21-1434 #75-6945	75-6946						
DT155	= 073700	21-1434 #75-6946							
DT156	= 073606	21-1434 #75-6947							
DT157	= 073606	21-1434 #75-6948							
DT16	= 072130	21-1434 #75-6814							
DT160	= 073606	21-1434 #75-6949							
DT161	= 073606	21-1434 #75-6950							
DT162	= 073606	21-1434 #75-6951							
DT163	= 073606	21-1434 #75-6952							
DT164	= 071712	21-1434 #75-6953							



SYMBOL	CROSS REFERENCE	VALUE	REFERENCES							
DT165	=	071712	21-1434	#75-6954						
DT166	=	073606	21-1434	#75-6955						
DT167	=	073606	21-1434	#75-6956						
DT17	=	071712	21-1434	#75-6815						
DT170	=	073606	21-1434	#75-6957						
DT171	=	073606	21-1434	#75-6958						
DT172	=	073606	21-1434	#75-6959						
DT173	=	073606	21-1434	#75-6960						
DT174	=	073606	21-1434	#75-6961						
DT175	=	073606	21-1434	#75-6962						
DT176	=	073606	21-1434	#75-6963						
DT177	=	073712	21-1434	#75-6964						
DT2	=	071670	21-1434	#75-6796						
DT20	=	072150	21-1434	#75-6816						
DT200	=	073606	21-1434	#75-6965						
DT201	=	073606	21-1434	#75-6966						
DT202	=	073606	21-1434	#75-6967						
DT203	=	073606	21-1434	#75-6968						
DT204	=	073606	21-1434	#75-6969						
DT205	=	071712	21-1434	#75-6970						
DT206	=	073606	21-1434	#75-6971						
DT207	=	073606	21-1434	#75-6972						
DT21	=	072172	21-1434	#75-6818	75-6886		75-6892			
DT210	=	073606	21-1434	#75-6973						
DT211	=	073722	21-1434	#75-6974						
DT212	=	073740	21-1434	#75-6975	75-6976					
DT213	=	073740	21-1434	#75-6976						
DT214	=	073606	21-1434	#75-6977						
DT22	=	072204	21-1434	#75-6819						
DT23	=	072232	21-1434	#75-6822						
DT24	=	072254	21-1434	#75-6824						
DT25	=	072300	21-1434	#75-6827						
DT26	=	072350	21-1434	#75-6829	75-6837					
DT27	=	072424	21-1434	#75-6832	75-6838					
DT3	=	071712	21-1434	#75-6797	75-6799	75-6815	75-6911	75-6912	75-6922	75-6953 75-6954
DT30	=	072466	21-1434	#75-6834						
DT31	=	072350	21-1434	#75-6837						
DT32	=	072424	21-1434	#75-6838						
DT33	=	072520	21-1434	#75-6839	75-6848					
DT34	=	072576	21-1434	#75-6842	75-6845	75-6846	75-6847	75-6867	75-6914	
DT35	=	072576	21-1434	#75-6845						
DT36	=	072576	21-1434	#75-6846						
DT37	=	072576	21-1434	#75-6847						
DT4	=	071712	21-1434	#75-6799						
DT40	=	072520	21-1434	#75-6848						
DT41	=	072650	21-1434	#75-6849						
DT42	=	072702	21-1434	#75-6852	75-6855					
DT43	=	072702	21-1434	#75-6855						
DT44	=	072760	21-1434	#75-6856						
DT45	=	073030	21-1434	#75-6858						
DT46	=	073104	21-1434	#75-6861						



SYMBOL CROSS REFERENCE

SYMBOL	CROSS REFERENCE	VALUE	REFERENCES	CREF	V01						
DT47		073172	21-1434	#75-6865	75-6870	75-6875	75-6913				
DT5		071734	21-1434	#75-6800							
DT50	=	072576	21-1434	#75-6867							
DT51		073224	21-1434	#75-6868	75-6915						
DT52	=	073172	21-1434	#75-6870							
DT53		073262	21-1434	#75-6871	75-6876	75-6878					
DT54		073302	21-1434	#75-6873	75-6877	75-6879	75-6880	75-6881	75-6885	75-6891	75-6896
			75-6900	75-6904							
DT55	=	073172	21-1434	#75-6875							
DT56	=	073262	21-1434	#75-6876							
DT57	=	073302	21-1434	#75-6877							
DT6		071766	21-1434	#75-6802	75-6803	75-6804	75-6805	75-6812	75-6813		
DT60	=	073262	21-1434	#75-6878							
DT61	=	073302	21-1434	#75-6879							
DT62	=	073302	21-1434	#75-6880							
DT63	=	073302	21-1434	#75-6881							
DT64		073340	21-1434	#75-6882	75-6884						
DT65	=	073340	21-1434	#75-6884							
DT66	=	073302	21-1434	#75-6885							
DT67	=	072172	21-1434	#75-6886							
DT7	=	071766	21-1434	#75-6803							
DT70		073360	21-1434	#75-6887	75-6889	75-6893	75-6894	75-6897	75-6899	75-6901	75-6903
DT71	=	073360	21-1434	#75-6889	75-6898						
DT72		073426	21-1434	#75-6890	75-6895	75-6902					
DT73	=	073302	21-1434	#75-6891							
DT74	=	072172	21-1434	#75-6892							
DT75	=	073360	21-1434	#75-6893							
DT76	=	073360	21-1434	#75-6894							
DT77	=	073426	21-1434	#75-6895							
D1		005774	28-1825	#28-1830	28-1848	28-1854					
D10		006176	28-1885	#28-1888							
D2		006020	*28-1833	28-1835	#28-1836						
D3		006022	#28-1837	28-1856							
D4		006026	#28-1839								
D5		006040	#28-1845	28-1876	28-1882	28-1892					
D6		006054	28-1846	#28-1850							
D7		006064	28-1851	#28-1853							
D8		006134	#28-1870	28-1879	28-1884						
D9		006146	28-1875	#28-1878							
EDONE		006354	29-1922	29-1928	29-1934	#29-1947					
EEDATO		035232	53-5760	53-5766	53-5789	53-5795	53-5813	53-5814	53-5815	53-5816	#53-5817
EEDONE		035314	53-5808	53-5812	53-5813	53-5814	53-5815	53-5816	#53-5823		
EEERO		035024	53-5778	53-5807	#53-5809						
EEER1		035042	53-5771	#53-5813							
EEER2		035100	53-5773	#53-5814							
EEER3		035136	53-5800	#53-5815							
EEER4		035174	53-5802	#53-5816							
EEO		035242	53-5762	53-5791	53-5813	53-5814	53-5815	53-5816	#53-5818		
EEO1		035254	53-5755	53-5757	53-5813	53-5813	53-5814	53-5814	#53-5819		
EEO2		035264	53-5767	#53-5820							
EEO3		035274	53-5784	53-5786	53-5815	53-5815	53-5816	53-5816	#53-5821		
EEO4		035304	53-5796	#53-5822							



SYMBOL CROSS REFERENCE		REFERENCES	
SYMBOL	VALUE		
EER0	006272	29-1917	#29-1924
EER1	006310	29-1921	#29-1930
EER2	006324	29-1906	#29-1936
EE1	034534	#53-5751	
EE10	034764	#53-5798	53-5801
EE11	034774	53-5799	#53-5801
EE12	035002	53-5794	#53-5803
EE13	035020	53-5806	#53-5808
EE2	034570	53-5754	#53-5758
EE3	034612	#53-5764	53-5774
EE4	034632	#53-5769	53-5772
EE5	034642	53-5770	#53-5772
EE6	034650	53-5765	#53-5774
EE7	034656	53-5777	#53-5780
EE8	034722	53-5783	#53-5787
EE9	034744	#53-5793	53-5803
EMTVEC	= 000030	#19-1415	*24-1443 *24-1443
EM1	044704	21-1434	#72-6203
EM10	= 045112	21-1434	#72-6213
EM100	054014	21-1434	#72-6338
EM101	054054	21-1434	#72-6339
EM102	054200	21-1434	#72-6340
EM103	054252	21-1434	#72-6341
EM104	054355	21-1434	#72-6342
EM105	054416	21-1434	#72-6343
EM106	054543	21-1434	#72-6344
EM107	054616	21-1434	#72-6345
EM11	045233	21-1434	#72-6214
EM110	054722	21-1434	#72-6346
EM111	054764	-1434	#72-6362 72-6363 72-6366
EM112	= 054764	21-1434	#72-6363
EM113	055066	21-1434	#72-6364 72-6365 72-6367
EM114	= 055066	21-1434	#72-6365
EM115	= 054764	21-1434	#72-6366
EM116	= 055066	21-1434	#72-6367
EM117	055170	21-1434	#72-6368
EM12	= 000000	21-1434	#72-6215
EM120	055324	21-1434	#72-6369
EM121	055460	21-1434	#72-6370
EM122	055577	21-1434	#72-6372
EM123	055676	21-1434	#72-6374
EM124	055737	21-1434	#72-6375
EM125	056032	21-1434	#72-6376
EM126	056122	21-1434	#72-6377
EM127	056331	21-1434	#72-6382
EM13	= 000000	21-1434	#72-6216
EM130	056544	21-1434	#72-6387
EM131	056644	21-1434	#72-6390
EM132	056704	21-1434	#72-6391
EM133	056744	21-1434	#72-6432 72-6436
EM134	057003	21-1434	#72-6433 72-6437
EM135	057042	21-1434	#72-6434 72-6438



CKFPAC

CREATED BY MACRO ON 28-MAR-81 AT 13:02

PAGE 20

SEQUENCE 206

CREF V01

SYMBOL	CROSS REFERENCE	REFERENCES		
SYMBOL	VALUE			
EM136	= 057101	21-1434	#72-6435	72-6439
EM137	= 056744	21-1434	#72-6436	
EM14	= 045314	21-1434	#72-6217	
EM140	= 057003	21-1434	#72-6437	
EM141	= 057042	21-1434	#72-6438	
EM142	= 057101	21-1434	#72-6439	
EM143	= 057140	21-1434	#72-6440	72-6442
EM144	= 057173	21-1434	#72-6441	72-6443
EM145	= 057140	21-1434	#72-6442	
EM146	= 057173	21-1434	#72-6443	
EM147	= 057226	21-1434	#72-6444	72-6445 72-6446
EM15	= 045437	21-1434	#72-6220	
EM150	= 057226	21-1434	#72-6445	
EM151	= 057226	21-1434	#72-6446	
EM152	= 057260	21-1434	#72-6447	72-6448
EM153	= 057260	21-1434	#72-6448	
EM154	= 057312	21-1434	#72-6449	
EM155	= 057544	21-1434	#72-6450	
EM156	= 057777	21-1434	#72-6451	
EM157	= 060214	21-1434	#72-6452	
EM16	= 045562	21-1434	#72-6223	
EM160	= 060433	21-1434	#72-6453	
EM161	= 060640	21-1434	#72-6454	
EM162	= 061045	21-1434	#72-6455	
EM163	= 061112	21-1434	#72-6456	
EM164	= 061157	21-1434	#72-6457	
EM165	= 061224	21-1434	#72-6458	
EM166	= 061271	21-1434	#72-6459	
EM167	= 061401	21-1434	#72-6460	
EM17	= 045633	21-1434	#72-6224	
EM170	= 061640	21-1434	#72-6461	
EM171	= 061750	21-1434	#72-6462	
EM172	= 062207	21-1434	#72-6463	
EM173	= 062446	21-1434	#72-6464	
EM174	= 062705	21-1434	#72-6465	
EM175	= 063144	21-1434	#72-6466	
EM176	= 063403	21-1434	#72-6467	
EM177	= 063540	21-1434	#72-6468	
EM2	= 044741	21-1434	#72-6207	
EM20	= 046066	21-1434	#72-6228	
EM200	= 063574	21-1434	#72-6469	
EM201	= 063731	21-1434	#72-6470	
EM202	= 064066	21-1434	#72-6471	
EM203	= 064223	21-1434	#72-6472	
EM204	= 064360	21-1434	#72-6473	
EM205	= 064515	21-1434	#72-6474	
EM206	= 064562	21-1434	#72-6475	
EM207	= 064627	21-1434	#72-6476	
EM21	= 046244	21-1434	#72-6232	
EM210	= 064751	21-1434	#72-6478	
EM211	= 045112	21-1434	#72-6479	
EM212	= 045146	21-1434	#72-6480	



SYMBOL CROSS REFERENCE		REFERENCES			
SYMBOL	VALUE				
EM213	= 045200	21-1434	#72-6481		
EM214	065026	21-1434	#72-6482		
EM22	046375	21-1434	#72-6235	72-6237	72-6238
EM23	= 046375	21-1434	#72-6237		
EM24	= 046375	21-1434	#72-6238		
EM25	046462	21-1434	#72-6239		
EM26	046575	21-1434	#72-6241	72-6243	
EM27	= 046575	21-1434	#72-6243		
EM3	045005	21-1434	#72-6208		
EM30	046643	21-1434	#72-6244		
EM31	046715	21-1434	#72-6246	72-6248	
EM32	= 046715	21-1434	#72-6248		
EM33	046763	21-1434	#72-6249		
EM34	047024	21-1434	#72-6251		
EM35	047126	21-1434	#72-6253		
EM36	047230	21-1434	#72-6255		
EM37	047331	21-1434	#72-6257		
EM4	045052	21-1434	#72-6209		
EM40	047432	21-1434	#72-6259		
EM41	047603	21-1434	#72-6261		
EM42	047640	21-1434	#72-6266		
EM43	047761	21-1434	#72-6267		
EM44	050102	21-1434	#72-6268	72-6269	
EM45	= 050102	21-1434	#72-6269		
EM46	050145	21-1434	#72-6270		
EM47	050223	21-1434	#72-6275		
EM5	045112	21-1434	#72-6210	72-6213	72-6479
EM50	050341	21-1434	#72-6276		
EM51	050437	21-1434	#72-6278		
EM52	050500	21-1434	#72-6298		
EM53	050621	21-1434	#72-6299		
EM54	051016	21-1434	#72-6290		
EM55	051062	21-1434	#72-6291		
EM56	051203	21-1434	#72-6292		
EM57	051400	21-1434	#72-6293		
EM6	045146	21-1434	#72-6211	72-6480	
EM60	051444	21-1434	#72-6294		
EM61	051641	21-1434	#72-6295		
EM62	051705	21-1434	#72-6296		
EM63	052077	21-1434	#72-6300		
EM64	052271	21-1434	#72-6304	72-6306	
EM65	= 052271	21-1434	#72-6306		
EM66	052425	21-1434	#72-6307		
EM67	052470	21-1434	#72-6329		
EM7	045200	21-1434	#72-5212	72-6481	
EM70	052721	21-1434	#72-6330		
EM71	053044	21-1434	#72-6331		
EM72	053146	21-1434	#72-6332		
EM73	053222	21-1434	#72-6333		
EM74	053262	21-1434	#72-6334		
EM75	053513	21-1434	#72-6335		
EM76	053636	21-1434	#72-6336		



SYMBOL CROSS REFERENCE  
SYMBOL VALUE

EM77 053740  
ERM10 037042  
ERROR = 104000

REFERENCES

21-1434	#72-6337									
57-5902	57-5902	#57-5902								
#19-1415	25-1510	25-1534	25-1547	25-1560	25-1565	26-1601	26-1607	27-1790		
27-1797	28-1843	28-1867	28-1881	28-1891	29-1927	29-1933	29-1945	30-2079		
30-2098	30-2102	30-2112	30-2134	30-2138	30-2146	30-2161	30-2177	30-2192		
30-2207	30-2215	30-2221	30-2240	30-2250	30-2258	30-2277	30-2287	30-2295		
31-2454	31-2461	31-2469	31-2481	32-2625	32-2632	32-2639	32-2650	33-2886		
34-3058	35-3182	35-3188	35-3194	35-3197	35-3201	35-3204	35-3216	35-3227		
36-3300	36-3307	36-3316	37-3398	37-3405	37-3414	38-3496	38-3503	38-3515		
39-3594	39-3604	39-3611	39-3622	40-3726	40-3734	40-3740	40-3752	40-3759		
41-3866	41-3874	41-3881	41-3893	41-3900	42-3999	42-4009	42-4016	42-4024		
43-4131	43-4141	43-4151	43-4158	44-4310	44-4319	44-4337	44-4352	44-4368		
45-4430	45-4438	45-4458	45-4466	45-4486	45-4494	45-4516	45-4524	46-4653		
46-4659	46-4667	46-4672	47-4748	47-4756	47-4762	48-4836	48-4842	49-4990		
49-4994	49-5002	49-5008	49-5016	49-5022	49-5028	49-5034	50-5229	50-5233		
50-5239	50-5245	50-5254	50-5260	50-5266	50-5275	51-5462	51-5467	51-5474		
51-5483	51-5489	51-5498	52-5716	52-5718	52-5719	52-5720	52-5721	52-5722		
52-5723	52-5724	52-5725	52-5726	52-5727	52-5728	52-5729	53-5811	53-5813		
53-5814	53-5815	53-5816	54-5881	54-5884	54-5886	56-5900	57-5902	67-6079		
68-6088	69-6097									
#19-1415	24-1443	*24-1443	*24-1443	*25-1473	*25-1487	*28-1827	*30-1977	*30-2016		
*31-2352	*32-2529	*35-3103	*35-3133	*36-3261	*37-3360	*39-3555	*40-3659	*41-3799		
*42-3938	*43-4059	56-5900	*56-5900	*56-5900	*56-5900	*70-6117				
25-1526	33-2904	57-5902	#66-5930							
#66-5978	66-6045									
66-5983	66-5991	66-6000	66-6009	66-6033	#66-6037					
66-6017	66-6024	#66-6042								
66-5975	66-6044	#66-6047								
66-5942	#66-6052									
29-1905	#29-1908									
#29-1911										
29-1910	#29-1912									
#29-1913	29-1936									
30-1969	30-1993	30-2077	30-2078	30-2087	#30-2298					
#30-2259										
#30-2300										
30-2234	#30-2301									
30-1976	30-1985	30-2082	30-2092	30-2227	#30-2302					
30-2243	#30-2303									
#30-2304										
#30-2305										
30-2011	30-2032	30-2123	30-2143	#30-2307						
#30-2308										
#30-2309										
30-2271	#30-2310									
30-2015	30-2024	30-2118	30-2128	30-2212	30-2263	#30-2311				
30-2149	30-2164	30-2280	#30-2312							
30-2180	30-2195	#30-2313								
30-2144	30-2213	#30-2314								
30-2072	30-2080	30-2099	30-2103	30-2113	30-2135	30-2139	30-2147	30-2162		
30-2178	30-2193	30-2208	30-2216	30-2222	30-2241	30-2251	30-2259	30-2278		
30-2288	30-2296	#30-2326								

ERRVEC = 000004

ERTYPE 041342  
ERT1 041526  
ERT2 041712  
ERT3 041716  
ERT4 041726  
ERT5 041740  
E1 006230  
E2 006244  
E3 006244  
E4 006246  
FDAT10 010174  
FDAT11 010176  
FDAT12 010200  
FDAT13 010202  
FDAT14 010204  
FDAT15 010206  
FDAT16 010210  
FDAT17 010212  
FDATOC 010216  
FDATO1 010220  
FDATO2 010222  
FDATO3 010224  
FLATC 010226  
FDATO5 010230  
FDATO6 010232  
FDATO7 010234  
FDONE 010260



SYMBOL	CROSS REFERENCE	REFERENCES								
SYMBOL	VALUE									
FECMS	042455	#71-6132	75-6801							
FERRO	006766	30-1998	#30-2074							
FERR1	007024	30-1987	#30-2082							
FERR10	007604	30-2053	30-2065	30-2168	30-2199	#30-2210				
FERR11	007640	30-2004	30-2071	#30-2218						
FERR2	007122	30-1991	#30-2105							
FERR20	007660	30-1977	#30-2224							
FERR21	007776	30-2232	#30-2253							
FERR25	010026	30-2016	#30-2261							
FERR26	010144	30-2269	#30-2290							
FERR3	007162	30-2026	#30-2118							
FERR4	007154	30-2030	#30-2115							
FERR5	007260	30-2037	30-2041	30-2045	30-2049	#30-2141				
FERR6	007314	30-2057	#30-2149							
FERR7	007450	30-2061	#30-2180							
FER2	007126	#30-2107	30-2116							
FFDAT0	035622	54-5844	54-5866	54-5884	54-5886	#54-5889				
FFDONE	035702	54-5877	54-5882	54-5884	54-5886	#54-5895				
FFERO	035512	54-5854	54-5876	#54-5879						
FFER1	035526	54-5872	#54-5884							
FFER2	035564	54-5850	#54-5886							
FFP0	035632	54-5861	54-5884	#54-5890						
FFP1	035642	54-5863	54-5884	#54-5891						
FFP2	035652	54-5839	54-5886	#54-5892						
FFP3	035662	54-5841	54-5886	#54-5893						
FFP4	035672	54-5846	54-5868	54-5884	54-5886	#54-5894				
FF1	035320	#54-5835								
FF10	035500	54-5871	#54-5873							
FF11	035510	54-5875	#54-5877							
FF2	035354	54-5838	#54-5842							
FF3	035376	#54-5848	54-5851							
FF4	035404	54-5849	#54-5851							
FF5	035414	54-5853	#54-5857							
FF6	035450	54-5860	#54-5864							
FF7	035472	#54-5870	54-5873							
FPSMS	042411	#71-6131	75-6801							
FPSPUR	042066	25-1484	28-1858	29-1938	35-3156	35-3175	44-4316	44-4358	49-4979	#67-6073
		70-6116								
FPVECT	= 000244	#19-1410	*25-1468	*25-1484	*28-1828	*29-1906	*35-3098	*35-3131	*44-4208	*44-4299
		*49-4883	*51-5318	*70-6116						
FXDAT0	010240	30-1970	30-1994	30-2033	30-2075	30-2076	30-2145	#30-2316		
FXDAT1	010242	#30-2317								
FXDAT2	010244	#30-2318								
FXDAT3	010246	#30-2319								
FXDAT4	010250	30-2214	#30-2320							
FXDAT5	010252	#30-2321								
FXDAT6	010254	#30-2322								
FXDAT7	010256	#30-2323								
F1	006360	#30-1964	30-1964							
F10	006602	30-2007	#30-2020	30-2268						
F11	006604	#30-2021	30-2115							
F12	006622	30-2025	#30-2028							



SYMBOL CROSS REFERENCE		REFERENCES									
SYMBOL	VALUE										
F13	006654	30-2036	#30-2039								
F135	006634	30-2029	#30-2032								
F14	006664	30-2040	#30-2043								
F15	006674	30-2044	#30-2047								
F16	006704	30-2048	#30-2051								
F17	006714	30-2052	#30-2055								
F2	006416	#30-1973	30-1974								
F20	006724	30-2056	#30-2059								
F21	006734	30-2060	#30-2063								
F22	006744	30-2064	#30-2067								
F23	006762	30-2070	#30-2072								
F3	006436	30-1965	#30-1981								
F4	006440	#30-1982	30-2105	30-2231							
F5	006456	30-1986	#30-1989								
F6	006532	30-2003	#30-2006								
F7	006562	#30-2012	30-2013								
GADR	015030	*33-2873	33-2887	*33-2901	33-2951	#33-2979					
GAND0	015000	33-2875	33-2899	33-2909	33-2925	33-2942	#33-2967				
GAND1	015002	#33-2968									
GAND2	015004	#33-2969									
GAND3	015006	#33-2970									
GCMP	014210	33-2811	33-2812	33-2813	33-2814	33-2815	33-2816	33-2817	33-2818	33-2819	
		33-2820	33-2821	33-2822	#33-2863						
GDATA00	015020	33-2811	33-2812	33-2813	33-2814	33-2815	33-2816	33-2817	33-2818	33-2819	
		33-2820	33-2821	33-2822	33-2850	33-2856	33-2863	33-2885	#33-2975		
GDATA01	015022	#33-2976									
GDATA02	015024	#33-2977									
GDATA03	015026	#33-2978									
GDONE	015032	33-2824	#33-2980								
GERR1	014236	33-2868	#33-2873								
GFLAG1	014754	33-2811	*33-2811	33-2812	*33-2812	33-2813	*33-2813	33-2814	*33-2814	33-2815	
		*33-2815	33-2816	*33-2816	33-2817	*33-2817	33-2818	*33-2818	33-2819	*33-2819	
		33-2820	*33-2820	33-2821	*33-2821	33-2822	*33-2822	33-2836	#33-2957		
GFLAG2	014756	*33-2883	33-2891	33-2898	#33-2958						
GNS	= *****	19-1426	19-1426	24-1444	55-5898	55-5898	64-5916	64-5916	64-5916	64-5916	64-5916
		64-5916	64-5916	64-5916	64-5916	64-5916	64-5916	64-5916	64-5916	64-5916	64-5916
		64-5916	64-5916	64-5916	64-5916	64-5916	64-5916	64-5916	64-5917	64-5917	
		33-2900	33-2928	#33-2971							
GORO	015010										
GOR1	015012	#33-2972									
GOR2	015014	#33-2973									
GOR3	015016	#33-2974									
GPAT00	014760	33-2811	33-2813	33-2815	33-2817	33-2819	33-2821	33-2846	#33-2959		
GPAT01	014762	#33-2960									
GPAT02	014764	#33-2961									
GPAT03	014766	#33-2962									
GPAT10	014770	33-2812	33-2814	33-2816	33-2818	33-2820	33-2822	33-2841	#33-2963		
GPAT11	014772	#33-2964									
GPAT12	014774	#33-2965									
GPAT13	014776	#33-2966									
GRESET	014170	33-2811	33-2812	33-2813	33-2814	33-2815	33-2816	33-2817	33-2818	33-2819	
		33-2820	33-2821	33-2822	#33-2856						
GSETUP	014112	33-2811	33-2812	33-2813	33-2814	33-2815	33-2816	33-2817	33-2818	33-2819	



SYMBOL	CROSS REFERENCE VALUE	REFERENCES	33-2821	33-2822	#33-2836	33-2815	33-2816	33-2817	33-2818	33-2819
GSUM	014322	33-2820	33-2821	33-2822	#33-2836					
		33-2811	33-2812	33-2813	33-2814	33-2815	33-2816	33-2817	33-2818	33-2819
		33-2820	33-2821	33-2822	#33-2891					
GS1	014142	#33-2846	33-2860							
GTSWR	= 104406	24-1444	#64-5916							
G1	012020	33-2811	33-2811	#33-2811	33-2811					
G10	012326	33-2813	#33-2813							
G11	012330	33-2813	#33-2813							
G12	012422	33-2814	33-2814	#33-2814	33-2814					
G13	012454	33-2814	#33-2814							
G14	012456	33-2814	#33-2814							
G15	012550	33-2815	33-2815	#33-2815	33-2815					
C16	012602	33-2815	#33-2815							
G17	012604	33-2815	#33-2815							
G2	012052	33-2811	#33-2811							
G20	012676	33-2816	33-2816	#33-2816	33-2816					
G21	012730	33-2816	#33-2816							
G22	012732	33-2816	#33-2816							
G23	013024	33-2817	33-2817	#33-2817	33-2817					
G24	013056	33-2817	#33-2817							
G25	013060	33-2817	#33-2817							
G26	013152	33-2818	33-2818	#33-2818	33-2818					
G27	013204	33-2818	#33-2818							
G3	012054	33-2811	#33-2811							
G30	013206	33-2818	#33-2818							
G31	013300	33-2819	33-2819	#33-2819	33-2819					
G32	013332	33-2819	#33-2819							
G33	013334	33-2819	#33-2819							
G34	013426	33-2820	33-2820	#33-2820	33-2820					
G35	013460	33-2820	#33-2820							
G36	013462	33-2820	#33-2820							
G37	013554	33-2821	33-2821	#33-2821	33-2821					
G4	012146	33-2812	33-2812	#33-2812	33-2812					
G40	013606	33-2821	#33-2821							
G41	013610	33-2821	#33-2821							
G42	013702	33-2822	33-2822	#33-2822	33-2822					
G43	013734	33-2822	#33-2822							
G44	013736	33-2822	#33-2822							
G5	012200	33-2812	#33-2812							
G6	012202	33-2812	#33-2812							
G7	012274	33-2813	33-2813	#33-2813	33-2813					
HADR	015566	*34-3033	34-3041	#34-3062						
HA1R	015642	34-3025	34-3035	34-3044	#34-3071					
HA1W	015572	34-2991	34-3000	34-3010	34-3034	34-3052	#34-3065			
HA2R	015652	34-3026	#34-3072							
HA2W	015602	34-3001	34-3011	#34-3066						
HA3R	015662	34-3027	#34-3073							
HA3W	015612	34-3002	34-3012	#34-3067						
HA4R	015672	34-3028	#34-3074							
HA4W	015622	34-3003	34-3013	#34-3068						
HA5R	015702	34-3029	#34-3075							
HA5W	015632	34-3004	34-3014	#34-3069						



SYMBOL	CROSS REFERENCE	VALUE	REFERENCES	CREF	V01					
HCLR		015516	34-2997	34-3024	#34-3044					
HCLR1		015526	#34-3046	34-3047						
HCMP		015460	34-3008	34-3010	34-3011	34-3012	34-3013	34-3014	#34-3033	
HCMP1		015500	#34-3037	34-3040						
HCMP2		015510	34-3038	#34-3040						
HDAT1		015712	34-2992	#34-3077						
HDAT2		015722	#34-3078							
HDAT3		015732	#34-3079							
HDAT4		015742	#34-3080							
HDAT5		015752	#34-3081							
HDONE		015762	34-3018	34-3059	#34-3082					
HERROR		015534	34-3039	#34-3051						
HFLAG		015570	*34-2990	34-3016	*34-3020	#34-3063				
HSTD		015402	34-3006	34-3010	34-3011	34-3012	34-3013	34-3014	#34-3024	
HT	=	000011	#19-1415	59-5906	59-5906					
H1		015044	34-2988	#34-2990						
H10		015306	#34-3013	34-3013						
H11		015340	#34-3014	34-3014						
H12		015372	34-3017	#34-3020						
H2		015064	#34-2994	34-2995						
H3		015074	#34-2999	34-3021						
H4		015146	#34-3006							
H5		015170	#34-3010	34-3010						
H6		015222	#34-3011	34-3011						
H7		015254	#34-3012	34-3012						
IBSAVE		036504	#57-5902	*57-5902	*57-5902	57-5902	57-5902	57-5902	57-5902	
IDATIO		011122	31-2339	31-2345	31-2363	31-2389	31-2394	31-2423	31-2459	#31-2500
IDATI1		011124	#31-2501							
IDATI2		011126	*31-2420	#31-2502						
IDATI3		011130	*31-2421	#31-2503						
IDAT00		011112	31-2359	31-2362	31-2417	31-2422	31-2460	#31-2495		
IDAT01		011114	#31-2496							
IDAT02		011116	31-2368	31-2373	31-2428	#31-2497				
IDAT03		011120	31-2431	#31-2498						
IDONE		011132	31-2437	31-2455	31-2462	31-2470	31-2482	#31-2505		
IERR0		010672	31-2352	#31-2440						
IERR1		010754	31-2377	31-2430	#31-2458					
IERR2		010774	31-2381	#31-2465						
IERR25		011016	#31-2468	31-2474						
IERR3		011046	31-2412	#31-2477						
IERR4		011022	31-2433	#31-2471						
ILLMS		043365	#71-6159	75-6824						
ILL1		043261	#71-6156							
ILL2		043324	#71-6158							
IOTVEC	=	000020	#19-1415	*24-1443	*24-1443					
IPAT10		011072	31-2340	31-2388	#31-2485					
IPAT11		011074	#31-2486							
IPAT12		011076	31-2428	#31-2487						
IPAT13		011100	31-2431	#31-2488						
IPAT20		011102	31-2348	31-2397	#31-2490					
IPAT21		011104	#31-2491							
IPAT22		011106	#31-2492							



SYMBOL	CROSS REFERENCE	REFERENCES
SYMBOL	VALUE	
IPAT23	011110	#31-2493
I1	010272	31-2335 #31-2337
I10	010454	31-2371 #31-2380
I105	010462	31-2366 #31-2383
I106	010456	31-2379 #31-2381
I11	010464	#31-2387
I12	010472	31-2387 #31-2388
I13	010506	#31-2391 31-2392
I14	010550	31-2401 #31-2405 31-2478
I15	010552	#31-2406 31-2444
I16	010554	#31-2407 31-2446
I17	010572	31-2411 #31-2414
I2	010310	#31-2342 31-2343
I20	010632	#31-2425 31-2435
I21	010646	#31-2430 31-2432
I22	010652	31-2429 #31-2431
I23	010666	31-2426 #31-2435
I3	010356	31-2353 #31-2355
I4	010360	#31-2356 31-2440
I5	010362	#31-2357 31-2442
I6	010406	#31-2365 31-2383
I7	010422	#31-2370 31-2380
JBUF0	016766	36-3270 #36-3319
JBUF1	016770	#36-3320
JBUF2	016772	#36-3321
JBUF3	016774	#36-3322
JDAT10	016776	36-3259 36-3274 36-3282 36-3306 36-3314 #36-3324
JDAT11	017000	#36-3325
JDAT12	017002	#36-3326
JDAT13	017004	#36-3327
JDAT00	017006	36-3267 36-3275 36-3315 #36-3329
JDAT0	017016	36-3256 #36-3334
JDAT01	017010	#36-3330
JDAT02	017012	#36-3331
JDAT03	017014	#36-3332
JDAT1	017020	#36-3335
JDAT2	017022	#36-3336
JDAT3	017024	#36-3337
JDONE	017026	36-3286 36-3301 36-3308 36-3317 #36-3340
JERRO	016646	36-3261 #36-3290
JERR1	016714	36-3272 36-3284 #36-3303
JERR2	016740	36-3279 #36-3312
J1	016540	36-3251 #36-3253
J10	016666	36-3291 36-3293 #36-3296
J2	016564	#36-3263 36-3304 36-3313
J3	016566	#36-3264 36-3290
J4	016570	#36-3265 36-3292
J5	016624	#36-3277 36-3280
J6	016632	36-3278 #36-3280
J7	016644	36-3283 #36-3286
KBUF0	017274	37-3358 37-3369 37-3381 #37-3422
KBUF1	017276	#37-3423



SYMBOL	CROSS REFERENCE	VALUE	REFERENCES
KBUF 2		017300	#37-3424
KBUF 3		017302	#37-3425
KDATI0		017264	37-3373 37-3404 37-3412 #37-3417
KDATI1		017266	#37-3418
KDATI2		017270	#37-3419
KDATI3		017272	#37-3420
KDATO0		017304	37-3366 37-3374 37-3413 #37-3427
KDATO1		017306	#37-3428
KDATO2		017310	#37-3429
KDATO3		017312	#37-3430
KDONE		017324	37-3385 37-3399 37-3406 37-3415 #37-3437
KERR0		017146	37-3360 #37-3389
KERR1		017212	37-3371 37-3383 #37-3401
KERR2		017236	37-3378 #37-3410
KPATO		017314	37-3355 #37-3432
KPAT1		017316	#37-3433
KPAT2		017320	#37-3434
K1		017040	37-3350 #37-3352
K10		017166	37-3390 37-3392 #37-3395
K2		017064	#37-3362 37-3402 37-3411
K3		017066	#37-3363 37-3389
K4		017070	#37-3364 37-3391
K5		017124	#37-3376 37-3379
K6		017132	37-3377 #37-3379
K7		017144	37-3382 #37-3385
LDATI0		017630	38-3455 38-3462 38-3481 38-3501 38-3513 #38-3527
LDATI1		017632	#38-3528
LDATI2		017634	38-3475 *38-3479 38-3495 #38-3529
LDATI3		017636	*38-3480 #38-3530
LDATO0		017642	38-3472 38-3482 38-3502 38-3507 #38-3531
LDATO1		017644	38-3514 #38-3532
LDATO2		017646	#38-3533
LDATO3		017650	#38-3534
LDONE		017652	38-3490 38-3497 38-3504 38-3516 #38-3536
LD1		043121	#71-6146
LD2		043151	#71-6147
LERR1		017464	38-3477 #38-3492
LERR2		017536	38-3487 #38-3506
LERR3		017510	#38-3499 38-3510
LF	=	000012	#19-1415 59-5906 59-5906
LF IEX1		042277	#71-6129 75-6800 75-6802
LF IEX2		042347	#71-6130 75-6800 75-6802
LFPS1		043106	#71-6145
LOOP		004300	#24-1445 55-5898
LPAT10		017606	38-3452 #38-3518
LPAT11		017610	#38-3519
LPAT12		017612	#38-3520
LPAT13		017614	#38-3521
LPAT20		017616	38-3456 38-3506 #38-3523
LPAT21		017620	#38-3524
LPAT22		017622	#38-3525
LPAT23		017624	#38-3526







SYMBOL	CROSS REFERENCE	REFERENCES								
SYMBOL	VALUE									
MS32	044470	#71-6193	75-6862	75-6864						
MS33	044477	#71-6194	75-6862	75-6864						
MS34	044506	#71-6195	75-6862	75-6864						
MS35	044515	31-2354	32-2531	#71-6196						
MS36	044522	31-2402	32-2573	#71-6197						
MS37	044531	#71-6198	75-6918	75-6923						
MS4	043501	#71-6163	75-6827	75-6828	75-6840	75-6841				
MS40	044547	#71-6199	75-6924							
MS41	044603	#71-6201	75-6926							
MS415	044563	#71-6200	75-6924							
MS42	044627	#71-6202	75-6927							
MS43	044645	#71-6203	75-6927							
MS44	044662	#71-6204	75-6928							
MS5	043513	#71-6164								
MS6	043555	#71-6165	75-6830	75-6843	75-6853	75-6887	75-6917			
MS7	043573	#71-6166	75-6830	75-6831	75-6833	75-6843	75-6844	75-6853	75-6854	75-6865
		75-6887	75-6888	75-6917	75-6918					
M1	017656	#39-3548								
M15	017676	#39-3557	39-3591	39-3603	39-3608	39-3620				
M2	017702	#39-3560	39-3597	39-3619						
M3	017704	#39-3561								
M4	017706	#39-3562								
M5	017742	39-3575	#39-3577							
M6	017746	#39-3578	39-3580							
M7	017756	39-3579	#39-3583							
M8	017766	#39-3585	39-3588							
M9	017776	39-3586	#39-3588							
NDAT10	020700	40-3701	40-3758	40-3766	#40-3775					
NDAT11	020702	#40-3776								
NDAT12	020704	#40-3777								
NDAT13	020706	#40-3778								
NDAT00	020636	40-3666	40-3683	40-3690	40-3700	40-3757	#40-3762			
NDAT01	020640	#40-3763								
NDAT02	020642	#40-3764								
NDAT03	020644	#40-3765								
NDONE	020710	40-3706	40-3727	40-3735	40-3741	40-3753	40-3760	#40-3780		
NERR0	020400	40-3659	#40-3714							
NERR1	020500	40-3674	#40-3736							
NERR10	020432	40-3715	#40-3723							
NERR11	020444	40-3720	#40-3729							
NERR2	020534	40-3678	#40-3742							
NERR20	020506	#40-3737	40-3743	40-3745	40-3747					
NERR3	020544	40-3688	#40-3744							
NERR4	020554	40-3696	#40-3746							
NERR5	020564	40-3698	#40-3749							
NERR6	020610	40-3708	#40-3755							
NOOP1	042473	#71-6134	75-6806	75-6810						
NOOP10	042751	#71-6142	75-6808	75-6810						
NOOP11	043042	#71-6144	75-6810							
NOOP15	042522	#71-6135	75-6806							
NOOP2	042617	#71-6136	75-6806	75-6807						
NOOP3	042634	#71-6137	75-6807	75-6807						



SYMBOL	CROSS REFERENCE	REFERENCES								
SYMBOL	VALUE									
NOOP4	042646	#71-6138	75-6807							
NOOP5	042663	#71-6139	75-6807							
NOOP6	042711	#71-6140	75-6808							
NOOP7	042731	#71-6141	75-6808							
NPAT10	020670	40-3654	#40-3770							
NPAT11	020672	#40-3771								
NPAT12	020674	#40-3772								
NPAT13	020676	#40-3773								
NPAT20	020656	40-3657	40-3669	40-3672	40-3676	40-3680	40-3691	40-3719	40-3750	#40-3766
NPAT21	020660	#40-3767								
NPAT22	020662	#40-3768								
NPAT23	020664	#40-3769								
NULL	042271	30-2224	30-2261	#71-6126						
N1	020206	#40-3651								
N10	020340	#40-3693	40-3695							
N11	020350	40-3681	40-3694	#40-3698						
N12	020352	40-3670	#40-3700							
N13	020366	#40-3703	40-3705							
N14	020376	40-3704	#40-3708							
N2	020232	#40-3662	40-3739	40-3751	40-3756					
N3	020234	#40-3663	40-3716							
N4	020236	#40-3664	40-3714							
N5	020254	#40-3672								
N6	020264	40-3673	#40-3676							
N7	020274	40-3677	#40-3680							
N8	020312	#40-3685	40-3687							
N9	020324	40-3686	#40-3690							
ODAT10	021404	41-3841	41-3899	41-3907	#41-3917					
ODAT11	021406	#41-3918								
ODAT12	021410	#41-3919								
ODAT13	021412	#41-3920								
ODAT00	021342	41-3807	41-3830	41-3840	41-3898	#41-3903				
ODAT01	021344	41-3823	#41-3904							
ODAT02	021346	#41-3905								
ODAT03	021350	#41-3906								
ODONE	021414	41-3846	41-3867	41-3875	41-3882	41-3894	41-3901	#41-3922		
OERR0	021104	41-3799	#41-3854							
OERR1	021204	41-3815	41-3860	#41-3877						
OERR10	021136	41-3855	#41-3863							
OERR11	021150	#41-3869								
OERR2	021240	41-3819	#41-3883							
OERR20	021212	#41-3878	41-3884	41-3886	41-3888					
OERR3	021250	41-3828	#41-3885							
OERR4	021260	41-3836	#41-3887							
OERR5	021270	41-3838	#41-3890							
OERR6	021314	41-3848	#41-3896							
OPAT10	021374	41-3794	#41-3912							
OPAT11	021376	#41-3913								
OPAT12	021400	#41-3914								
OPAT13	021402	#41-3915								
OPAT20	021360	41-3810	41-3891	#41-3907						
OPAT21	021362	41-3797	41-3813	41-3817	41-3821	41-3831	41-3859	#41-3908		







SYMBOL	CROSS REFERENCE	VALUE	REFERENCES							
PPAT13		022046	#42-4030							
PROGNUM	=	000001	#19-807	24-1444						
PRO	=	000000	#19-1415							
PR1	=	000040	#19-1415							
PR2	=	000100	#19-1415							
PR3	=	000140	#19-1415							
PR4	=	000200	#19-1415							
PR5	=	000240	#19-1415							
PR6	=	000300	#19-1415							
PR7	=	000340	#19-1415							
PS	=	177776	#19-1415	19-1415						
PSW	=	177776	#19-1415	26-1585						
PWRVEC	=	000024	#19-1415	*24-1443	*24-1443	*65-5920	*65-5920	*65-5920	*65-5920	*65-5920
P1		021420	#42-3932							
P2		021442	#42-3942	42-3943	42-4005	42-4013	42-4020			
P3	=	021444	#42-3943	42-3966	43-4101					
P4		021446	#42-3945	42-3968						
P5		021470	#42-3950	42-3952						
P6		021512	42-3951	#42-3958						
P7		021522	#42-3960	42-3963						
P8		021532	42-3961	#42-3963						
QDAT10		022560	43-4073	43-4156	43-4166	#43-4176				
QDAT11		022562	#43-4177							
QDAT12		022564	#43-4178							
QDAT13		022566	#43-4179							
QDAT00		022550	43-4068	43-4072	43-4083	43-4091	43-4157	#43-4171		
QDAT01		022552	#43-4172							
QDAT02		022554	#43-4173							
QDAT03		022556	#43-4174							
QDONE		022570	43-4081	43-4132	43-4142	43-4152	43-4159	#43-4181		
QERRO		022236	43-4059	#43-4101						
QERR1		022502	43-4095	#43-4154						
QERR11		022246	#43-4104							
QERR12		022264	43-4105	#43-4108						
QERR13		022302	43-4109	#43-4112						
QERR14		022320	43-4113	#43-4116						
QERR15		022336	43-4117	#43-4121						
QERR16		022346	43-4122	#43-4125						
QERR17		022354	43-4107	43-4111	43-4115	43-4119	#43-4127			
QERR2		022436	43-4088	#43-4144						
QERR20		022402	43-4123	#43-4134						
QERR21		022412	43-4080	#43-4137						
QERR22		022420	43-4136	#43-4138						
QERR3		022446	43-4097	#43-4146						
QERR4		022454	43-4145	#43-4148						
QPAT10		022530	43-4056	#43-4161						
QPAT11		022532	#43-4162							
QPAT12		022534	#43-4163							
QPAT13		022536	#43-4164							
QPAT20		022540	43-4063	43-4078	43-4090	43-4104	43-4108	43-4112	43-4116	43-4121
			#43-4166							
QPAT21		022542	#43-4167							







SYMBOL CROSS REFERENCE

SYMBOL	CROSS REFERENCE VALUE	REFERENCES								
STACK	= 001100	#19-1415	24-1443	70-6120						
START	003616	19-1426	#24-1443	65-5920						
STFS1	043162	#71-6148								
STKLMT	= 177774	#19-1415								
STST1	043312	#71-6157								
ST1	043175	#71-6149								
ST2	043212	#71-6150								
SWR	001140	#20-1427	24-1443	*24-1443	24-1443	*24-1443	*24-1443	24-1444	25-1516	25-1518
		33-2893	33-2895	55-5898	56-5900	56-5900	56-5900	56-5900	56-5900	57-5902
		57-5902	57-5902	57-5902	57-5902	63-5914	63-5914	65-5920	65-5920	70-6109
SWREG	000176	#19-1426	24-1443	24-1444	63-5914	63-5914				
SW0	= 000001	#19-1415								
SW00	= 000001	#19-1415	19-1415							
SW01	= 000002	#19-1415	19-1415							
SW02	= 000004	#19-1415	19-1415							
SW03	= 000010	#19-1415	19-1415							
SW04	= 000020	#19-1415	19-1415							
SW05	= 000040	#19-1415	19-1415							
SW06	= 000100	#19-1415	19-1415							
SW07	= 000200	#19-1415	19-1415							
SW08	= 000400	#19-1415	19-1415							
SW09	= 001000	#19-1415	19-1415							
SW1	= 000002	#19-1415								
SW10	= 002000	#19-1415								
SW11	= 004000	#19-1415								
SW12	= 010000	#19-1415								
SW13	= 020000	#19-1415	25-1516	33-2893						
SW14	= 040000	#19-1415								
SW15	= 100000	#19-1415								
SW2	= 000004	#19-1415								
SW3	= 000010	#19-1415								
SW4	= 000020	#19-1415								
SW5	= 000040	#19-1415								
SW6	= 000100	#19-1415								
SW7	= 000200	#19-1415	25-1518	33-2895						
SW8	= 000400	#19-1415								
SW9	= 001000	#19-1415								
S1	015766	#35-3093								
S10	016144	#35-3139								
S11	016170	#35-3147	35-3150							
S12	016200	35-3148	#35-3150							
S2	016026	#35-3106	35-3193	35-3200						
S3	016030	#35-3107	35-3154	35-3208						
S4	016034	#35-3109	35-3210							
S5	016060	#35-3117	35-3120							
S6	016070	35-3118	#35-3120							
S7	016076	#35-3125	35-3158							
S8	016136	#35-3136	35-3196	35-3203	35-3219					
S9	016140	#35-3137	35-3173	35-3221						
TAB	= 000011	#19-1413	71-6127	71-6129	71-6130	71-6134	71-6140	71-6141	71-6142	71-6142
		71-6142	71-6145	71-6145	71-6146	71-6146	71-6146	71-6147	71-6147	71-6148
		71-6148	71-6149	71-6149	71-6150	71-6150	71-6151	71-6152	71-6153	71-6154







SYMBOL CROSS REFERENCE

SYMBOL	CROSS REFERENCE VALUE	REFERENCES							
TST22	021416	#42-3930							
TST23	022072	#43-4051							
TST24	022572	#44-4199							
TST25	023732	#45-4411							
TST26	024530	#46-4553							
TST27	025344	#47-4703							
TST3	005000	#27-1620							
TST30	025754	#48-4799							
TST31	026262	#49-4879							
TST32	027330	#50-5055							
TST33	031146	#51-5313							
TST34	032502	#52-5525							
TST35	034532	#53-5749							
TST36	035316	#54-5833							
TST37	035702	#54-5896							
TST4	005744	#28-1824							
TST5	006212	#29-1904							
TST6	006356	#30-1962							
TST7	010262	#31-2334							
TYPDS	= 104405	55-5898	55-5898	#64-5916					
TYPE	= 104401	24-1444	33-2913	33-2920	33-2936	33-2946	33-2947	55-5898	55-5898
		57-5902	57-5902	59-5906	60-5908	61-5910	63-5914	63-5914	63-5914
		63-5914	63-5914	#64-5916	65-5920	66-5930	66-5958	66-5960	66-5965
		66-6015	66-6022	66-6037	66-6047	66-6065			
		63-5914	#64-5916	66-5941	66-5982	66-6064			
TYPOC	= 104402	#64-5916							
TYPON	= 104404	#64-5916							
TYPOS	= 104403	#64-5916	66-6006						
T1	011136	#32-2514							
T10	011304	32-2548	#32-2550						
T105	011312	32-2543	#32-2553						
T11	011314	#32-2557							
T12	011322	32-2557	#32-2559						
T13	011336	#32-2562	32-2563						
T14	011400	32-2572	#32-2575						
T15	011402	#32-2576	32-2615	32-2647					
T16	011404	#32-2577	32-2617						
T17	011422	32-2582	#32-2585						
T2	011162	#32-2519	32-2520						
T20	011462	#32-2596	32-2606						
T21	011476	#32-2601	32-2603						
T22	011500	32-2600	#32-2602						
T23	011512	32-2597	#32-2606						
T3	011230	32-2530	#32-2532						
T4	011232	#32-2533	32-2611						
T5	011234	#32-2534	32-2613						
T6	011260	#32-2542	32-2553						
T7	011274	#32-2547	32-2550						
UDONE	023732	44-4311	44-4320	44-4338	44-4353	44-4364	44-4369	#44-4401	
UERR0	023376	44-4208	#44-4314						
UERR1	023422	44-4228	44-4285	#44-4324					
UERR10	023430	#44-4326	44-4340						
UERR11	023472	#44-4335							







SYMBOL	CROSS REFERENCE VALUE	REFERENCES								
WPAT00	024506	45-4415	45-4418	45-4422	45-4424	45-4443	45-4448	45-4452	45-4454	45-4471
		45-4474	45-4478	45-4480	45-4499	45-4504	45-4508	45-4510	45-4529	45-4530
		45-4531	45-4532	#45-4534						
WPAT01	024510	#45-4535								
WPAT02	024512	#45-4536								
WPAT03	024514	#45-4537								
WSETUP	024454	45-4429	45-4485	45-4515	#45-4529					
W1	023734	#45-4412								
W10	024156	45-4457	#45-4460							
W11	024204	45-4462	#45-4468							
W12	024240	45-4473	#45-4475							
W13	024264	#45-4482	45-4488							
W14	024300	45-4483	#45-4488							
W15	024326	45-4490	#45-4496							
W16	024366	45-4501	#45-4505							
W17	024412	#45-4512	45-4518							
W2	023770	45-4417	#45-4419							
W20	024426	45-4513	#45-4518							
W3	024014	#45-4426	45-4432							
W4	024032	45-4427	#45-4432							
W5	024062	45-4434	#45-4440							
W6	024122	45-4445	#45-4449							
W7	024146	#45-4456	45-4460							
XAPT11	025324	#46-4686								
XDAT00	025302	46-4565	46-4588	46-4611	46-4634	46-4652	46-4657	#46-4675		
XDAT01	025304	#46-4676								
XDAT02	025306	#46-4677								
XDAT03	025310	#46-4678								
XDONE	025342	46-4646	46-4654	46-4660	46-4668	46-4673	#46-4694			
XERR1	025166	46-4571	46-4594	#46-4650						
XERR2	025250	46-4576	46-4599	#46-4664						
XERR3	025214	46-4617	46-4640	#46-4655						
XERR4	025264	46-4622	46-4645	#46-4669						
XPAT00	025312	46-4557	46-4567	46-4603	46-4613	#46-4680				
XPAT01	025314	#46-4681								
XPAT02	025316	#46-4682								
XPAT03	025320	#46-4683								
XPAT10	025322	46-4561	46-4584	46-4607	46-4630	46-4650	46-4655	#46-4685		
XPAT12	025326	#46-4687								
XPAT13	025330	#46-4688								
XPAT20	025332	46-4580	46-4590	46-4626	46-4636	#46-4689				
XPAT21	025334	#46-4690								
XPAT22	025336	#46-4691								
XPAT23	025340	#46-4692								
XTMP	025300	*46-4558	*46-4581	*46-4604	*46-4627	46-4651	46-4656	46-4658	#46-4674	
X1	024532	#46-4554								
X10	024734	46-4593	#46-4595							
X11	024750	46-4598	#46-4600							
X12	025010	46-4606	#46-4608							
X13	025034	#46-4615	46-4618							
X14	025042	46-4616	#46-4618							
X15	025056	46-4621	#46-4623							



SYMBOL	CROSS REFERENCE	VALUE	REFERENCES
X16		025116	46-4629 #46-4631
X17		025142	#46-4638 46-4641
X2		024572	46-4560 #46-4562
X20		025150	46-4639 #46-4641
X21		025164	46-4644 #46-4646
X3		024616	#46-4569 46-4572
X4		024624	46-4570 #46-4572
X5		024642	46-4575 #46-4577
X6		024702	46-4583 #46-4585
X7		024726	#46-4592 46-4595
YDAT00		025712	47-4717 47-4737 47-4746 47-4754 #47-4770
YDAT01		025714	#47-4771
YDAT02		025716	#47-4772
YDAT03		025720	#47-4773
YDONE		025752	47-4734 47-4749 47-4757 47-4763 #47-4790
YERR1		025552	47-4739 #47-4742
YERR2		025614	47-4741 #47-4750
YERR3		025656	47-4726 #47-4758
YFLAG		025702	*47-4704 47-4727 #47-4765
YPAT00		025722	47-4705 47-4731 #47-4775
YPAT01		025724	#47-4776
YPAT02		025726	#47-4777
YPAT03		025730	#47-4778
YPAT10		025732	47-4706 47-4730 #47-4780
YPAT11		025734	#47-4781
YPAT12		025736	#47-4782
YPAT13		025740	#47-4783
YPAT20		025742	47-4711 47-4745 47-4753 #47-4785
YPAT21		025744	#47-4786
YPAT22		025746	#47-4787
YPAT23		025750	#47-4788
YTMP1		025704	*47-4705 47-4713 *47-4730 47-4736 47-4744 47-4752 #47-4766
YTMP2		025706	*47-4706 47-4720 *47-4731 47-4747 47-4755 #47-4767
YTMP3		025710	*47-4707 47-4724 *47-4732 47-4761 #47-4768
Y1		025374	#47-4708 47-4733
Y2		025422	#47-4714 47-4743 47-4751 47-4759
Y3		025446	#47-4721 47-4723
Y4		025464	47-4725 #47-4727
Y5		025524	47-4728 #47-4734
Y6		025526	47-4722 #47-4735
Y7		025542	#47-4738 47-4740
ZDAT00		026220	48-4812 48-4834 #48-4849
ZDAT01		026222	#48-4850
ZDAT02		026224	#48-4851
ZDAT03		026226	#48-4852
ZDONE		026260	48-4829 48-4837 48-4843 #48-4869
ZERR1		026124	48-4818 #48-4830
ZERR2		026166	48-4822 #48-4838
ZFLAG		026212	*48-4800 48-4823 *48-4825 #48-4845
ZPAT00		026230	48-4801 #48-4854
ZPAT01		026232	#48-4855
ZPAT02		026234	#48-4856







SYMBOL	CROSS REFERENCE VALUE	REFERENCES	CREF	V01						
		#20-1427	20-1427	20-1427	#20-1427	20-1427	20-1427	#20-1427	20-1427	20-1427
		#20-1427	20-1427	20-1427	#20-1427	20-1427	20-1427	#20-1427	20-1427	20-1427
		#20-1427	20-1427	20-1427	#20-1427	20-1427	20-1427	#20-1427	20-1427	20-1427
		#20-1427	20-1427	20-1427	#20-1427	20-1427	20-1427	#20-1427	20-1427	20-1427
\$CNTLG	041043	63-5914	#63-5914							
\$CNTLU	041036	63-5914	#63-5914							
\$CPUOP	001344	#20-1427								
\$CRLF	001313	#20-1427	33-2947	55-5898	57-5902	57-5902	57-5902	59-5906	59-5906	59-5906
		63-5914	66-5930	66-5961	66-5968	66-6048	75-6819	75-6827	75-6827	75-6828
		75-6828	75-6829	75-6830	75-6830	75-6831	75-6833	75-6833	75-6833	75-6834
		75-6834	75-6840	75-6840	75-6840	75-6841	75-6843	75-6843	75-6843	75-6844
		75-6849	75-6849	75-6853	75-6853	75-6854	75-6854	75-6856	75-6856	75-6857
		75-6857	75-6857	75-6857	75-6858	75-6858	75-6859	75-6859	75-6859	75-6860
		75-6861	75-6862	75-6865	75-6866	75-6869	75-6869	75-6873	75-6873	75-6874
		75-6874	75-6887	75-6887	75-6888	75-6888	75-6916	75-6917	75-6917	75-6918
		75-6918	75-6918	75-6923	75-6923	75-6924	75-6924	75-6924	75-6924	75-6926
		75-6926	75-6927	75-6927	75-6927	75-6927	75-6928	75-6928	75-6928	
		61-5910	61-5910	#61-5910						
\$DBLK	040146	#20-1427								
\$DDW0	001402	#20-1427								
\$DDW1	001404	#20-1427								
\$DDW10	001426	#20-1427								
\$DDW11	001430	#20-1427								
\$DDW12	001432	#20-1427								
\$DDW13	001434	#20-1427								
\$DDW14	001436	#20-1427								
\$DDW15	001440	#20-1427								
\$DDW2	001406	#20-1427								
\$DDW3	001410	#20-1427								
\$DDW4	001412	#20-1427								
\$DDW5	001414	#20-1427								
\$DDW6	001416	#20-1427								
\$DDW7	001420	#20-1427								
\$DDW8	001422	#20-1427								
\$DDW9	001424	#20-1427								
\$DEVCT	001326	#20-1427								
\$DEVVM	001374	#20-1427								
\$DOAGN	036120	55-5898	55-5898	55-5898	#55-5898					
\$DTBL	040136	61-5910	#61-5910							
\$ENDAD	036110	22-1438	24-1444	#55-5898	57-5902					
\$ENDCT	035736	24-1443	#55-5898							
\$ENULL	036164	#55-5898								
\$ENV	001336	#20-1427	24-1444	57-5902	59-5906	62-5912	62-5912			
\$ENVM	001337	#20-1427	24-1443	59-5906	59-5906	62-5912				
\$EOP	035702	#55-5898	67-6081	68-6090	69-6099					
\$EOPCT	035730	*24-1443	#55-5898	55-5898						
\$ERFLG	001103	#20-1427	56-5900	*56-5900	56-5900	56-5900	*56-5900	56-5900	56-5900	*57-5902
		57-5902	57-5902							
\$ERMAX	001115	#20-1427	*24-1443	56-5900	*56-5900	56-5900	56-5900			
\$ERROR	036506	24-1443	#57-5902							
\$ERRPC	001116	#20-1427	*25-1524	*33-2902	*57-5902	*57-5902	57-5902	57-5902	*57-5902	57-5902
		57-5902	66-5933	66-5940	66-6060					
\$ERRTB	001442	#21-1427	66-5954							







SYMBOL CROSS REFERENCE

SYMBOL	VALUE	REFERENCES
\$MXCNT	036500	56-5900 #56-5900
\$NULL	001154	#20-1427 59-5906
\$NWTST	= 000001	#24-1464 24-1464 #25-1464 25-1464 #25-1575 25-1575 #26-1575 26-1575 #26-1620 26-1620 #27-1620 27-1620 #27-1824 27-1824 #28-1824 28-1824 #28-1904 28-1904 #29-1904 29-1904 #29-1962 29-1962 #30-1962 30-1962 #30-2334 30-2334 #31-2334 31-2334 #31-2513 31-2513 #32-2513 32-2513 #32-2809 32-2809 #33-2809 33-2809 #33-2987 33-2987 #34-2987 34-2987 #34-3092 34-3092 #35-3092 35-3092 #35-3250 35-3250 #36-3250 36-3250 #36-3349 36-3349 #37-3349 37-3349 #37-3446 37-3446 #38-3446 38-3446 #38-3546 38-3546 #39-3546 39-3546 #39-3649 39-3649 #40-3649 40-3649 #40-3789 40-3789 #41-3789 41-3789 #41-3930 41-3930 #42-3930 42-3930 #42-4051 42-4051 #43-4051 43-4051 #43-4199 43-4199 #44-4199 44-4199 #44-4411 44-4411 #45-4411 45-4411 #45-4553 45-4553 #46-4553 46-4553 #46-4703 46-4703 #47-4703 47-4703 #47-4799 47-4799 #48-4799 48-4799 #48-4879 48-4879 #49-4879 49-4879 #49-5055 49-5055 #50-5055 50-5055 #50-5313 50-5313 #51-5313 51-5313 #51-5525 51-5525 #52-5525 52-5525 #52-5749 52-5749 #53-5749 53-5749 #53-5833 53-5833 #54-5833 54-5833
\$OCNT	037726	*60-5908 #60-5908
\$OMODE	037730	*60-5908 #60-5908
\$OVER	036464	56-5900 #60-5908
\$PASS	001324	#20-1427 *24-1443 *55-5898 #60-5908
\$PASTM	003610	56-5900 #60-5908
\$PWRAD	041316	#23-1440 #60-5908
\$PWRDN	041156	24-1443 #65-5920 65-5920
\$PWRMG	041312	#65-5920
\$PWRUP	041230	65-5920 #65-5920
\$QUES	001312	#20-1427 57-5902 57-5902 59-5906 59-5906 63-5914
\$RDCHR	040706	#63-5914 64-5916 64-5916
\$RDDEC	= *****	64-5916
\$RDLIN	= *****	64-5916
\$RDOCT	= *****	64-5916
\$RDSZ	= 000001	#63-5914 63-5914
\$REGAD	001160	#20-1427
\$REGO	001162	#20-1427 *57-5902 *57-5902 57-5902
\$REG1	001164	#20-1427
\$REG10	001202	#20-1427
\$REG11	001204	#20-1427
\$REG12	001206	#20-1427
\$REG13	001210	#20-1427
\$REG14	001212	#20-1427
\$REG15	001214	#20-1427
\$REG16	001216	#20-1427
\$REG17	001220	#20-1427
\$REG2	001166	#20-1427
\$REG20	001222	#20-1427
\$REG21	001224	#20-1427
\$REG22	001226	#20-1427
\$REG23	001230	#20-1427
\$REG3	001170	#20-1427
\$REG4	001172	#20-1427
\$REG5	001174	#20-1427
\$REG6	001176	#20-1427



SYMBOL	CROSS REFERENCE	VALUE	REFERENCES								
\$REG7		001200	#20-1427								
\$RESRE		037102	#58-5904	64-5916							
\$RTNAD		036160	#55-5898								
\$RTRN		036154	24-1443	*24-1443	*24-1443	55-5898	#55-5898				
\$R2A	=	*****	64-5916								
\$SAVRE		037044	#58-5904	64-5916	64-5916						
\$SAVR6		041340	*65-5920	65-5920	*65-5920	*65-5920	#65-5920				
\$SCOPE		036170	24-1443	#56-5900							
\$SETUP	=	000137	#19-1425	19-1425	#19-1425	19-1425	#19-1425	19-1425	#19-1425	19-1425	#19-1425
			19-1425	#19-1425	19-1425	#19-1425	24-1443	24-1443	24-1443	24-1443	24-1443
			24-1443	24-1443	24-1443	24-1443	24-1443	24-1443	24-1443	24-1444	24-1444
			24-1444	55-5898	55-5898	56-5900	57-5902	57-5902	57-5902	57-5902	63-5914
			63-5914	65-5920							
\$STUP	=	177777	#19-1425	#19-1425	19-1425	#19-1425	#19-1425	19-1425	#19-1425	#19-1425	19-1425
			#19-1425	#19-1425	19-1425	#19-1425	#19-1425	19-1425	#19-1425	#19-1425	19-1425
\$SVLAD		036430	56-5900	#56-5900							
\$SVPC	=	003602	#22-1438	22-1438							
\$SWR	=	177400	19-1409	#19-1409	#19-1411	20-1427	20-1427	20-1427	24-1443	24-1443	24-1443
			24-1443	24-1443	25-1464	26-1575	27-1620	28-1824	29-1904	30-1962	31-2334
			32-2513	33-2809	34-2987	35-3092	36-3250	37-3349	38-3446	39-3546	40-3649
			41-3789	42-3930	43-4051	44-4199	45-4411	46-4553	47-4703	48-4799	49-4879
			50-5055	51-5313	52-5525	53-5749	54-5833	55-5898	55-5898	55-5898	55-5898
			55-5898	56-5900	56-5900	56-5900	56-5900	56-5900	56-5900	56-5900	56-5900
			56-5900	56-5900	56-5900	56-5900	56-5900	56-5900	56-5900	56-5900	56-5900
			56-5900	56-5900	56-5900	57-5902	57-5902	57-5902	57-5902	57-5902	57-5902
			57-5902	57-5902	57-5902	57-5902	57-5902	65-5920			
\$SWREG		001340	#20-1427	24-1443							
\$SWRMK	=	000000	56-5900	56-5900	56-5900	56-5900	56-5900	56-5900	56-5900	56-5900	56-5900
			56-5900								
\$SWRMS	=	000200	#19-1412								
\$TAB		042272	66-6038	#71-6127	75-6794	75-6794	75-6796	75-6796	75-6796	75-6797	75-6797
			75-6798	75-6800	75-6802	75-6808	75-6809	75-6810	75-6811	75-6814	75-6814
			75-6816	75-6816	75-6817	75-6818	75-6820	75-6820	75-6821	75-6822	75-6822
			75-6823	75-6825	75-6825	75-6826	75-6827	75-6829	75-6829	75-6832	75-6832
			75-6835	75-6835	75-6839	75-6839	75-6842	75-6842	75-6850	75-6850	75-6851
			75-6852	75-6852	75-6856	75-6858	75-6858	75-6865	75-6868	75-6868	75-6871
			75-6871	75-6873	75-6882	75-6882	75-6887	75-6890	75-6890	75-6905	75-6916
			75-6916	75-6923	75-6926	75-6932	75-6932	75-6945	75-6974	75-6974	75-6975
\$TBIT		036162	*24-1443	*55-5898	55-5898	55-5898	#55-5898	*65-5920			
\$TERM	=	000030	#64-5918								
\$TESTN		001322	#20-1427	*56-5900	66-6060						
\$THE		042466	#71-6133	75-6806	75-6810						
\$TIMES		001302	#20-1427	*24-1443	*55-5898	*56-5900	56-5900	*56-5900	56-5900	56-5900	56-5900
\$TKB		001146	#20-1427	59-5906	59-5906	59-5906	59-5906	63-5914	63-5914	63-5914	63-5914
			63-5914	63-5914							
\$TKS		001144	#20-1427	59-5906	59-5906	59-5906	59-5906	63-5914	63-5914	63-5914	63-5914
			63-5914	63-5914	63-5914						
\$TMPO		001232	#20-1427	*66-5931	*66-5932	75-6794	75-6796	75-6797	75-6800	75-6802	75-6808
			75-6810	75-6814	75-6816	75-6818	75-6820	75-6822	75-6825	75-6827	75-6829
			75-6832	75-6835	75-6839	75-6842	75-6850	75-6852	75-6856	75-6858	75-6861
			75-6865	75-6868	75-6871	75-6873	75-6882	75-6887	75-6890	75-6905	75-6916
			75-6923	75-6926	75-6932	75-6945	75-6964	75-6974	75-6975		



SYMBOL	CROSS REFERENCE	REFERENCES	CREF	V01						
SYMBOL	VALUE									
\$TMP1	001234	#20-1427	*66-5933	75-6794	75-6796	75-6797	75-6800	75-6802	75-6808	75-6810
		75-6814	75-6816	75-6818	75-6820	75-6822	75-6825	75-6827	75-6829	75-6832
		75-6835	75-6839	75-6842	75-6850	75-6852	75-6856	75-6858	75-6861	75-6865
		75-6868	75-6871	75-6873	75-6882	75-6887	75-6890	75-6905	75-6916	75-6923
		75-6926	75-6932	75-6945	75-6964	75-6974	75-6975			
\$TMP10	001252	#20-1427	*27-1624	*27-1653	*27-1681	*27-1710	*30-2225	*30-2257	*30-2262	*30-2293
		*44-4220	*44-4243	*44-4261	*44-4279	75-6806	75-6808	75-6810	75-6853	75-6863
		75-6918	75-6932							
\$TMP11	001254	#20-1427	*27-1751	*27-1757	*27-1775	*27-1781	75-6807	75-6864	75-6932	
\$TMP12	001256	#20-1427	75-6864							
\$TMP13	001260	#20-1427	75-6864							
\$TMP14	001262	#20-1427								
\$TMP15	001264	#20-1427	*30-2224	*30-2255	*30-2261	*30-2291	75-6853			
\$TMP16	001266	#20-1427								
\$TMP17	001270	#20-1427								
\$TMP2	001236	#20-1427	*25-1506	*25-1522	*25-1532	*25-1545	*25-1558	*25-1563	*26-1595	*27-1628
		*27-1642	*27-1657	*27-1669	*27-1685	*27-1698	*27-1713	*27-1726	*28-1835	*28-1889
		*29-1910	*29-1941	*30-1965	*30-2007	*30-2226	*30-2264	*31-2353	*31-2401	*31-2450
		*31-2478	*32-2530	*32-2572	*32-2621	*32-2647	*33-2811	*33-2812	*33-2813	*33-2814
		*33-2815	*33-2816	*33-2817	*33-2818	*33-2819	*33-2820	*33-2821	*33-2822	34-3053
		*35-3160	*35-3193	*35-3196	*35-3200	*35-3203	*35-3214	*35-3225	*36-3298	*36-3304
		*36-3313	*37-3397	*37-3402	*37-3411	*38-3493	*38-3500	*38-3512	*39-3591	*39-3603
		*39-3608	*39-3620	*40-3724	*40-3729	*40-3739	*40-3751	*40-3756	*41-3864	*41-3880
		*41-3892	*41-3897	*42-3997	*42-4002	*42-4005	*42-4013	*42-4020	*43-4129	*43-4134
		*43-4137	*43-4150	*43-4155	*44-4221	*44-4244	*44-4262	*44-4280	*44-4308	*44-4317
		*44-4365	*45-4417	*45-4445	*45-4473	*45-4501	*46-4560	*46-4583	*46-4606	*46-4629
		*47-4743	*47-4751	*47-4759	*48-4831	*48-4839	*49-4887	*49-4914	*49-4949	49-4975
		*49-4988	*49-4992	*50-5060	*50-5088	*50-5114	*50-5143	*50-5174	*50-5202	*51-5319
		*51-5342	*51-5368	*51-5391	*51-5420	*51-5441	*52-5530	*52-5559	*52-5581	*52-5603
		*52-5631	*52-5659	*52-5688	*53-5754	*53-5783	*54-5838	*54-5860	*67-6073	*68-6086
		*69-6095	75-6794	75-6796	75-6797	75-6800	75-6802	75-6808	75-6811	75-6814
		75-6816	75-6818	75-6820	75-6822	75-6825	75-6829	75-6832	75-6835	75-6839
		75-6842	75-6850	75-6852	75-6856	75-6858	75-6861	75-6865	75-6868	75-6871
		75-6873	75-6882	75-6887	75-6890	75-6905	75-6916	75-6923	75-6926	75-6932
		75-6945	75-6974	75-6975						
\$TMP20	001272	#20-1427	75-6801	75-6802	75-6834	75-6849				
\$TMP21	001274	#20-1427	75-6800	75-6802						
\$TMP22	001276	#20-1427								
\$TMP23	001300	#20-1427								
\$TMP3	001240	#20-1427	*25-1507	*25-1523	*26-1599	*26-1605	*27-1787	*27-1794	*28-1842	*28-1865
		*28-1879	*29-1925	*29-1931	*29-1944	*30-2075	*30-2083	*30-2111	*30-2119	*30-2142
		*30-2157	*30-2172	*30-2188	*30-2203	*30-2211	*30-2219	*30-2227	*30-2263	*31-2354
		*31-2402	*31-2451	*31-2479	*32-2531	*32-2573	*32-2623	*32-2648	*33-2884	*33-2899
		*35-3181	*35-3187	*36-3296	*36-3305	*36-3314	*37-3395	*37-3403	*37-3412	*38-3494
		*38-3501	*38-3513	*39-3592	*39-3602	*39-3609	*39-3618	*40-3749	*40-3757	*41-3869
		*41-3890	*41-3898	*42-4007	*42-4014	*43-4139	*43-4156	*44-4335	*44-4350	*44-4367
		*45-4437	*45-4465	*45-4493	*45-4523	*45-4529	*46-4650	*46-4655	*46-4665	*46-4670
		*47-4744	*47-4752	*47-4760	*48-4832	*48-4840	*49-5000	*49-5006	*49-5014	*49-5020
		*49-5024	*49-5030	*50-5228	*50-5232	*50-5235	*50-5241	*50-5247	*50-5250	*50-5256
		*50-5262	*50-5268	*50-5271	*50-5277	*50-5280	*50-5283	*50-5286	*51-5461	*51-5466
		*51-5472	*51-5481	*51-5487	*51-5496	*52-5715	*52-5718	*52-5719	*52-5720	*52-5721
		*52-5722	*52-5723	*52-5724	*52-5725	*52-5726	*52-5727	*52-5728	*52-5729	*53-5810



SYMBOL CROSS REFERENCE  
SYMBOL VALUE

REFERENCES

		*53-5813	*53-5814	*53-5815	*53-5816	*54-5879	*54-5884	*54-5886	*67-6076	75-6794
		75-6796	75-6797	75-6801	75-6809	75-6811	75-6814	75-6816	75-6820	75-6822
		75-6825	75-6827	75-6829	75-6832	75-6835	75-6839	75-6842	75-6850	75-6852
		75-6857	75-6859	75-6861	75-6865	75-6868	75-6871	75-6873	75-6874	75-6882
		75-6890	75-6916	75-6923	75-6926	75-6974				
\$TMP4	001242	#20-1427	*25-1508	*26-1600	*26-1606	*27-1788	*27-1795	*28-1866	*28-1880	*29-1926
		*29-1932	*30-2076	*30-2082	*30-2107	*30-2118	*30-2143	*30-2212	*30-2220	*31-2452
		*31-2459	*31-2480	*32-2624	*32-2630	*32-2649	*33-2885	*33-2900	*35-3180	*35-3186
		*36-3297	*36-3306	*36-3315	*37-3396	*37-3404	*37-3413	*38-3495	*38-3502	*38-3514
		*39-3593	*39-3598	*39-3610	*39-3619	*40-3750	*40-3758	*41-3878	*41-3891	*41-3899
		*42-4008	*42-4015	*43-4140	*43-4157	*44-4336	*44-4351	*44-4366	*45-4436	*45-4464
		*45-4492	*45-4522	*45-4530	*46-4651	*46-4656	*46-4666	*46-4671	*47-4745	*47-4753
		*47-4761	*48-4833	*48-4841	*49-4999	*49-5005	*49-5013	*49-5019	*49-5025	*49-5031
		*50-5227	*50-5231	*50-5237	*50-5243	*50-5252	*50-5258	*50-5264	*50-5274	*51-5460
		*51-5465	*51-5470	*51-5476	*51-5479	*51-5485	*51-5491	*51-5494	*51-5500	*51-5503
		*52-5714	*52-5718	*52-5719	*52-5720	*52-5721	*52-5722	*52-5723	*52-5724	*52-5725
		*52-5726	*52-5727	*52-5728	*52-5729	*53-5809	*53-5813	*53-5814	*53-5815	*53-5816
		*54-5880	*54-5884	*54-5886	*67-6078	75-6795	75-6798	75-6801	75-6809	75-6811
		75-6817	75-6821	75-6823	75-6826	75-6827	75-6828	75-6829	75-6833	75-6836
		75-6840	75-6841	75-6841	75-6851	75-6857	75-6860	75-6862	75-6866	75-6869
		75-6872	75-6883	75-6890	75-6916	75-6924	75-6927			
\$TMP5	001244	#20-1427	*25-1509	*27-1622	*28-1834	*30-2077	*30-2084	*30-2120	*30-2144	*30-2158
		*30-2173	*30-2189	*30-2204	*30-2213	*30-2229	*30-2265	*31-2460	*31-2465	*31-2471
		*32-2631	*32-2638	*33-2811	*33-2812	*33-2813	*33-2814	*33-2815	*33-2816	*33-2817
		*33-2818	*33-2819	*33-2820	*33-2821	*33-2822	*40-3731	*40-3737	*41-3871	*42-3995
		*42-4021	*43-4127	*43-4148	*44-4326	*44-4341	*45-4532	*46-4652	*46-4657	*47-4746
		*47-4754	*48-4834	*49-5001	*49-5007	*49-5015	*49-5021	*49-5026	*49-5032	*50-5238
		*50-5244	*50-5253	*50-5259	*50-5265	*50-5273	*51-5473	*51-5482	*51-5488	*51-5497
		*52-5718	*52-5719	*52-5720	*52-5721	*52-5722	*52-5723	*52-5724	*52-5725	*52-5726
		*52-5727	*52-5728	*52-5729	*53-5813	*53-5814	*53-5815	*53-5816	*54-5884	*54-5886
		75-6795	75-6806	75-6807	75-6814	75-6828	75-6828	75-6830	75-6831	75-6833
		75-6840	75-6841	75-6843	75-6844	75-6853	75-6854	75-6856	75-6858	75-6862
		75-6869	75-6887	75-6888	75-6917	75-6918	75-6924	75-6927		
\$TMP6	001246	#20-1427	*27-1752	*27-1758	*27-1776	*27-1782	*30-2078	*30-2089	*30-2094	*30-2125
		*30-2130	*30-2145	*30-2159	*30-2174	*30-2190	*30-2205	*30-2214	*30-2238	*30-2248
		*30-2256	*30-2275	*30-2285	*30-2292	*31-2466	*31-2472	*32-2636	*32-2641	*33-2898
		*40-3733	*40-3736	*40-3742	*40-3744	*40-3746	*41-3873	*41-3877	*41-3883	*41-3885
		*41-3887	*42-3972	*42-3976	*42-3980	*42-3984	*42-3988	*42-3993	*42-4023	*43-4106
		*43-4110	*43-4114	*43-4118	*43-4125	*43-4144	*43-4146	*44-4334	*44-4349	*45-4531
		*46-4658	*47-4747	*47-4755	*48-4835	*49-4998	*49-5004	*49-5010	*49-5012	*49-5018
		*49-5027	*49-5033	*50-5236	*50-5242	*50-5248	*50-5251	*50-5257	*50-5263	*50-5269
		*50-5272	*50-5278	*50-5281	*50-5284	*50-5287	*51-5471	*51-5477	*51-5480	*51-5486
		*51-5492	*51-5495	*51-5501	*51-5504	*52-5718	*52-5719	*52-5720	*52-5721	*52-5722
		*52-5723	*52-5724	*52-5725	*52-5726	*52-5727	*52-5728	*52-5729	*53-5813	*53-5814
		*53-5815	*53-5816	*54-5884	*54-5886	75-6807	75-6828	75-6830	75-6840	75-6843
		75-6853	75-6859	75-6862	75-6888	75-6917	75-6928			
\$TMP7	001250	#20-1427	*27-1623	*27-1652	*27-1680	*27-1709	*30-2085	*30-2121	*30-2160	*30-2175
		*30-2191	*30-2206	*30-2228	*30-2266	*31-2467	*31-2473	*32-2637	*32-2642	*40-3732
		*40-3738	*41-3872	*41-3879	*42-3996	*42-4022	*43-4128	*43-4149	*44-4328	*44-4343
		75-6807	75-6831	75-6833	75-6844	75-6854	75-6863	75-6888	75-6918	
\$STN	= 000037	19-1409	#19-1409	24-1464	25-1464	#25-1464	25-1575	26-1575	#26-1575	26-1620
		27-1620	#27-1620	27-1824	28-1824	#28-1824	28-1904	29-1904	#29-1904	29-1962



SYMBOL	CROSS REFERENCE	REFERENCES	30-1962	30-2334	31-2334	#31-2334	31-2513	32-2513	#32-2513	32-2809
SYMBOL	VALUE									
		30-1962	#30-1962	30-2334	31-2334	#31-2334	31-2513	32-2513	#32-2513	32-2809
		33-2809	#33-2809	33-2987	34-2987	#34-2987	34-3092	35-3092	#35-3092	35-3250
		36-3250	#36-3250	36-3349	37-3349	#37-3349	37-3446	38-3446	#38-3446	38-3546
		39-3546	#39-3546	39-3649	40-3649	#40-3649	40-3789	41-3789	#41-3789	41-3930
		42-3930	#42-3930	42-4051	43-4051	#43-4051	43-4199	44-4199	#44-4199	44-4411
		45-4411	#45-4411	45-4553	46-4553	#46-4553	46-4703	47-4703	#47-4703	47-4799
		48-4799	#48-4799	48-4879	49-4879	#49-4879	49-5055	50-5055	#50-5055	50-5313
		51-5313	#51-5313	51-5525	52-5525	#52-5525	52-5749	53-5749	#53-5749	53-5833
		54-5833	#54-5833	54-5896						
\$TPB	001152	#20-1427	59-5906	59-5906	59-5906					
\$TPFLG	001157	#20-1427	59-5906	59-5906	59-5906					
\$TPS	001150	#20-1427	59-5906	59-5906	59-5906					
\$TRAP	041072	24-1443	#64-5916							
\$TRAP2	041114	#64-5916	64-5916							
\$TRP	= 000014	#64-5916	64-5916	64-5916	64-5916	64-5916	#64-5916	64-5916	64-5916	64-5916
		64-5916	#64-5916	64-5916	64-5916	64-5916	64-5916	#64-5916	64-5916	64-5916
		64-5916	64-5916	#64-5916	64-5916	64-5916	64-5916	64-5916	#64-5916	64-5916
		64-5916	64-5916	64-5916	#64-5916	64-5916	64-5916	64-5916	64-5916	#64-5916
		64-5916	64-5916	64-5916	64-5916	#64-5916	64-5916	64-5916	64-5916	64-5916
		#64-5916	64-5916	64-5916	64-5916	64-5916	#64-5916	64-5916	64-5916	64-5916
		64-5917	#64-5917							
\$TRPAD	041126	64-5916	#64-5916	64-5918						
\$STSM	003606	#23-1440								
\$STSNM	001102	#20-1427	*55-5898	56-5900	56-5900	*56-5900	56-5900	56-5900	56-5900	56-5900
		57-5902	57-5902	57-5902	66-5931					
\$TYPBN	= *****	64-5916								
\$TYPDS	037732	#61-5910	64-5916	64-5916						
\$TYPE	037140	#59-5906	62-5912	64-5916	64-5916					
\$TYPEC	037352	59-5906	59-5906	59-5906	#59-5906	63-5914				
\$TYPEX	037502	59-5906	59-5906	59-5906	#59-5906					
\$TYPOC	037530	#60-5908	64-5916	64-5916						
\$TYPON	037544	60-5908	#60-5908	64-5916						
\$TYPOS	037504	#60-5908	64-5916							
\$UNIT	001330	#20-1427								
\$UNITM	003612	#23-1440								
\$USWR	001342	#20-1427								
\$VECT1	001366	#20-1427								
\$VECT2	001370	#20-1427								
\$XOFF	= 000023	59-5906	59-5906							
\$XON	= 000021	59-5906	59-5906	59-5906	63-5914					
\$XTSTR	036202	#56-5900								
\$\$GET4	= 000001	#55-5898	#55-5898	55-5898						
\$OFILL	037727	*60-5908	*60-5908	60-5908	#60-5908					
\$4OCAT	= *****	56-5900	57-5902							
.RSET	042154	64-5917	#70-6109							
.\$ASTA	= *****	62-5912	62-5912							
.\$X	= 003602	#23-1440	23-1440							











