

FP11-F

FP11F FLTG PNT PRT A
CKFPADO

AH-F632D-MC
FICHE 1 OF 2

APR 1982
COPYRIGHT © 79-82
MADE IN USA



The main body of the document is a large grid of data, likely a flight log or maintenance record. It consists of approximately 15 columns and 25 rows of small, dense text. The text is too small to be legible in this image, but it appears to be organized into a structured table format. The data points within the grid are arranged in a regular pattern, suggesting a systematic recording of information over time or across different categories.

FP11-F

FP11F FLTG PNT PRT A
CKFPADO

AH-F632D-MC
FICHE 2 OF 2

APR 1982
COPYRIGHT © 79-82
MADE IN USA



.REM 8

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-F630D-MC
PRODUCT NAME: CKFPAD0 FP11F FLTG PNT PRT A
DATE CREATED: OCTOBER, 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, 1982 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
84
85
86
87
88
89
90
91
92

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.

THE FOLLOWING WAS ADDED TO PART 'D':

THE ABILITY OF THE PROGRAM TO PRINT AN END-OF-PASS MESSAGE ONLY EVERY 1000TH PASS. EOP MESSAGES CAN BE DISABLED ALTOGETHER BY TYPING ANY KEY, AND CAN BE REENABLED BY AGAIN TYPING ANY KEY. THIS IS A PAPER-SAVING ADDITION WHEN THE DIAGNOSTIC IS TO BE RUN OVERNIGHT WITH A HARD COPY TERMINAL, NOT TO MENTION TURNING THE DIAGNOSTIC INTO A HARDWARE INTENSIVE TEST INSTEAD OF A TEST OF THE TERMINAL.

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
127
128
129
130
131
132
133
134
135

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 CKFPADO
- 10. LISTING
 - 10.1 CKFPADO

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
176
177
178
179
180
181
182
183
184

1.

ABSTRACT

THE THREE PROGRAMS:

CKFPADO CKFPBCO CKFPCDO

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. CKFPADO

CKFPADO 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. CKFPBCO

CKFPBCO TESTS:

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

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
231
232
233
234
235
236
237
238
239

C. CKFPCDO

CKFPCDO 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 CKFPBCO 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.

240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285

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 CKFPAD0.
SW<7>=1....	200	SELECT CORRECT INTERRUPT TEST IN PROGRAM CKFPBC0.

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
333
334
335
336
337
338
339
340
341

6. ERRORS

6.1 SUMMARIES

IN PROGRAM CKFPADO 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 CKFPADO 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.

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
383
384
385
386
387
388
389
390
391

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 CKFPBCO 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.

392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440

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.

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
484
485
486
487
488
489
490
491
492

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.

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
533
534
535
536
537
538
539
540
541TEST 11ACCUMULATORS 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 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.

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.

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

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 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.

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
677
678
679
680
681
682
683
684
685

- 2.) IF THERE ARE FOUR CONSECUTIVE BITS IN ERROR,
FOLLOWING THE PATTERN:
LN, LN+1, LN+2 AND LN+3 WHERE 'L' IS A, B, C OR D
N=0, 4, 8 OR 12
THEN THE ABOVE TABLE SHOULD DIRECTLY
IDENTIFY THE FAILING CHIP.
- 3.) IF FOUR BITS ARE DROPPED WHICH FIT THE
PATTERN:
AN, BN, CN AND DN WHERE N=15, 14, ... OR 0
OR 0
THEN ANY ONE OF THE FOUR CHIPS ASSOCIATED
WITH EACH OF THE BITS AN, BN, CN AND DN
COULD BE AT FAULT WITH EQUAL PROBABILITY.
- 4.) IF 16 BITS ARE IN ERROR, FITTING THE
PATTERN:
AN, AN+1, AN+2, AN+3 WHERE N=0, 4, 8 OR 12
BN, BN+1, BN+2, BN+3
CN, CN+1, CN+2, CN+3
AND
DN, DN+1, DN+2, DN+3
THEN ANY ONE OF THE FOUR CHIPS ASSOCIATED
WITH THESE BITS COULD BE AT FAULT WITH EQUAL
PROBABILITY.
- 5.) IF THE FAILING BIT PATTERNS DISPLAYED IN THE
'AND' AND THE 'OR' DATA TYPED IN THE SUMMARY
DOES NOT CONFORM EXPLICITELY TO ANY OF THE
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).

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
773
774
775
776
777
778
779
780
781

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, ADD, 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)

782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811

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.

812
813
814
815
816
817
818
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949

000214
000001

10. LISTING

```

&
MNUMBER=214
PROGNUM=1
.LIST ME
.NLIST MD,MC,CND
:*****
:*****
:***** .SEOP *****
:
: .SEOP IS USED TO HANDLE THE END OF A PASS
:
: ARGUMENTS ARE:
:
: 1) ADR -- WILL RETURN TO ADR IF NON-BLANK
:           WILL RETURN TO 200 IF BLANK
:
: 2) NUM -- SPECIFIES THE NUMBER OF PROGRAM PASSES
:           BEFORE INDICATING AN END OF PROGRAM
:           NOTE: IF BLANK IT DEFAULTS TO 1
:
: 3) OPTION -- SPECIFIES HOW THE END OF PROGRAM IS TO BE INDICATED.
:              THE AVAILABLE ARGUMENTS AND THEIR INDICATIONS ARE:
:              1) BELL RING TTY BELL
:              2) END TYPE 'END PASS'
:              3) PASCNT TYPE 'END PASS #XXXXX'
:                   WHERE XXXXX IS A DECIMAL NUMBER
:                   IS THE NAME OF A MACRO THAT
:                   MUST BE DEFINED IN THE USERS
:                   PROGRAM
:              4) ENDPAS
:              5) REPORT TYPE 'END PASS #XXXXX TOTAL ERRORS
:                   SINCE LAST REPORT YYYYY'
:                   WHERE XXXXX AND YYYYY ARE DECIMAL NUMBERS
:
:              THE DEFAULT IS NO INDICATION
:
: 4) INSTR1 -- THIS WILL BE THE FIRST INSTRUCTION IF NON-BLANK
:              SCOPE WILL BE THE FIRST INSTRUCTION IF BLANK
:
: 5) INSTR2 -- THIS ARGUMENT ALLOWS CODE TO BE INSERTED BETWEEN
:              THE TESTING OF LOCATION '42' AND THE 'RESET'
:              INSTRUCTION IMMEDIATELY PRECEDING THE 'XXDP' OR 'ACT11'
:              RETURN LINKAGE.
:              NOTE: CAN BE A MACRO
:
: NOTE: THIS ROUTINE IS CONDITIONALLY ASSEMBLED BY $SWR FOR SWR<12>
:
: ROUTINES REQUIRED:
:
: 1) TYPE AN ASCIZ STRING (.STYPE) DEPENDING ON $SWR
:
: 2) CHANGE BINARY TO DECIMAL AND TYPE (.STYPDEC).
:     ONLY REQUIRED WHEN 'OPTION' IS DEFINED AS 'PASCNT'
:     OR 'REPORT'

```


950
951
1771 000000
1772
1773
1774
1775
1776
1777

1778
1779
1780
1781
1782
1783

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

000000
000040
000100
000140
000200

```

:*****
:ENABL ABS
:      .MCALL .HEADER, .SWRHI, .EQUAT, .SETUP, .SCATCH, .SACT11, .SCMTAG
:      .MCALL .STYPE, .SSAVE
:      .MCALL .STYPDEC, .STRAP, .SPOWER, .SAPTHDR, .SAPTBL5
:      .MCALL .SAPTYPE, .SREAD
:      .MCALL .EQUIV ;*REMOVE FOR ASSEMBLY ON PDP-10
:TITLE CKFPADO 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
PR0= 0 ;:PRIORITY LEVEL 0
PR1= 40 ;:PRIORITY LEVEL 1
PR2= 100 ;:PRIORITY LEVEL 2
PR3= 140 ;:PRIORITY LEVEL 3
PR4= 200 ;:PRIORITY LEVEL 4

```

```
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
001000 SW9=SW09
000400 SW8=SW08
000200 SW7=SW07
000100 SW6=SW06
000040 SW5=SW05
000020 SW4=SW04
000010 SW3=SW03
000004 SW2=SW02
000002 SW1=SW01
000001 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
001000 BIT9=BIT09
000400 BIT8=BIT08
000200 BIT7=BIT07
000100 BIT6=BIT06
000040 BIT5=BIT05
000020 BIT4=BIT04
000010 BIT3=BIT03
000004 BIT2=BIT02
000002 BIT1=BIT01
000001 BIT0=BIT00
```



```

; *BASIC "CPU" TRAP VECTOR ADDRESSES
000004 ERRVEC= 4          ;; TIME OUT AND OTHER ERRORS
000010 RESVEC= 10       ;; RESERVED AND ILLEGAL INSTRUCTIONS
000014 TBITVEC=14      ;; "T" BIT
000014 TRTVEC= 14      ;; TRACE TRAP
000014 BPTVEC= 14      ;; BREAKPOINT TRAP (BPT)
000020 IOTVEC= 20      ;; INPUT/OUTPUT TRAP (IOT) **SCOPE**
000024 PWRVEC= 24      ;; POWER FAIL
000030 EMTVEC= 30      ;; EMULATOR TRAP (EMT) **ERROR**
000034 TRAPVEC=34      ;; "TRAP" TRAP
000060 TKVEC= 60        ;; TTY KEYBOARD VECTOR
000064 TPVEC= 64        ;; TTY PRINTER VECTOR
000240 PIRQVEC=240     ;; PROGRAM INTERRUPT REQUEST VECTOR

1784      .SBTTL FPP REGISTER DEFINITIONS
1785      AC0          =%0
1786      AC1          =%1
1787      AC2          =%2
1788      AC3          =%3
1789      AC4          =%4
1790      AC5          =%5
1791      AC6          =%6
1792      AC7          =%7
1794      .SBTTL TRAP CATCHER
          =0
; *ALL UNUSED LOCATIONS FROM 4 - 776 CONTAIN A ".+2,HALT"
; *SEQUENCE TO CATCH ILLEGAL TRAPS AND INTERRUPTS
; *LOCATION 0 CONTAINS 0 TO CATCH IMPROPERLY LOADED VECTORS
          =174
000174    000174    000000    DISPREG: .WORD 0          ;; SOFTWARE DISPLAY REGISTER
000176    000000    SWREG:   .WORD 0          ;; SOFTWARE SWITCH REGISTER
000200    000137    003616    .SBTTL STARTING ADDRESS(ES)
          JMP @#START ;; JUMP TO STARTING ADDRESS OF PROGRAM
    
```

1795

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

001162 000024
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

.SBTTL COMMON TAGS

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

.=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

.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

:::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

:::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
001232 000000 $TMP0: .WORD 0 ;;USER DEFINED
001234 000000 $TMP1: .WORD 0 ;;USER DEFINED
001236 000000 $TMP2: .WORD 0 ;;USER DEFINED
001240 000000 $TMP3: .WORD 0 ;;USER DEFINED
001242 000000 $TMP4: .WORD 0 ;;USER DEFINED
001244 000000 $TMP5: .WORD 0 ;;USER DEFINED
001246 000000 $TMP6: .WORD 0 ;;USER DEFINED
001250 000000 $TMP7: .WORD 0 ;;USER DEFINED
001252 000000 $TMP10: .WORD 0 ;;USER DEFINED
001254 000000 $TMP11: .WORD 0 ;;USER DEFINED
001256 000000 $TMP12: .WORD 0 ;;USER DEFINED
001260 000000 $TMP13: .WORD 0 ;;USER DEFINED
001262 000000 $TMP14: .WORD 0 ;;USER DEFINED
001264 000000 $TMP15: .WORD 0 ;;USER DEFINED
001266 000000 $TMP16: .WORD 0 ;;USER DEFINED
001270 000000 $TMP17: .WORD 0 ;;USER DEFINED
001272 000000 $TMP20: .WORD 0 ;;USER DEFINED
001274 000000 $TMP21: .WORD 0 ;;USER DEFINED
001276 000000 $TMP22: .WORD 0 ;;USER DEFINED
001300 000000 $TMP23: .WORD 0 ;;USER DEFINED
001302 000000 $TIMES: 0 ;;MAX. NUMBER OF ITERATIONS
001304 000000 $ESCAPE: 0 ;;ESCAPE ON ERROR ADDRESS
001306 207 377 377 $BELL: .ASCIIZ <207><377><377> ;;CODE FOR BELL
001311 000
001312 077
001313 015
001314 012 000 $QUES: .ASCII /?/ ;;QUESTION MARK
$CRLF: .ASCII <15> ;;CARRIAGE RETURN
$LF: .ASCIIZ <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,OPTIGNS
;*
;* 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)
```



```
.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
```

1802	001442	045352	065631	072314	\$ERRTB:				
	001442	045352	065631	072314	.WORD	EM1,DH1,DT1,DF1	:ERROR	ITEM	# 1
	001452	045407	065721	072336	.WORD	EM2,DH2,DT2,DF2	:ERROR	ITEM	# 2
	001462	045453	066014	072360	.WORD	EM3,DH3,DT3,DF3	:ERROR	ITEM	# 3
	001472	045520	066105	072360	.WORD	EM4,DH4,DT4,DF4	:ERROR	ITEM	# 4
	001502	045560	066201	072402	.WORD	EM5,DH5,DT5,DF5	:ERROR	ITEM	# 5
	001512	045614	066201	072434	.WORD	EM6,DH6,DT6,DF6	:ERROR	ITEM	# 6
	001522	045646	066201	072434	.WORD	EM7,DH7,DT7,DF7	:ERROR	ITEM	# 7
	001532	045560	066201	072434	.WORD	EM10,DH10,DT10,DF10	:ERROR	ITEM	# 10
	001542	045701	066201	072434	.WORD	EM11,DH11,DT11,DF11	:ERROR	ITEM	# 11
	001552	000000	000000	072456	.WORD	EM12,DH12,DT12,DF12	:ERROR	ITEM	# 12
	001562	000000	000000	072544	.WORD	EM13,DH13,DT13,DF13	:ERROR	ITEM	# 13
	001572	045762	066201	072434	.WORD	EM14,DH14,DT14,DF14	:ERROR	ITEM	# 14
	001602	046105	066201	072434	.WORD	EM15,DH15,DT15,DF15	:ERROR	ITEM	# 15
	001612	046230	066241	072576	.WORD	EM16,DH16,DT16,DF16	:ERROR	ITEM	# 16
	001622	046301	066321	072360	.WORD	EM17,DH17,DT17,DF17	:ERROR	ITEM	# 17
	001632	046534	066411	072616	.WORD	EM20,DH20,DT20,DF20	:ERROR	ITEM	# 20
	001642	046712	066201	072640	.WORD	EM21,DH21,DT21,DF21	:ERROR	ITEM	# 21
	001652	047043	066477	072652	.WORD	EM22,DH22,DT22,DF22	:ERROR	ITEM	# 22
	001662	047043	066534	072700	.WORD	EM23,DH23,DT23,DF23	:ERROR	ITEM	# 23
	001672	047043	066672	072722	.WORD	EM24,DH24,DT24,DF24	:ERROR	ITEM	# 24
	001702	047130	067031	072746	.WORD	EM25,DH25,DT25,DF25	:ERROR	ITEM	# 25
	001712	047243	067073	073016	.WORD	EM26,DH26,DT26,DF26	:ERROR	ITEM	# 26
	001722	047243	067073	073072	.WORD	EM27,DH27,DT27,DF27	:ERROR	ITEM	# 27
	001732	047311	000000	073134	.WORD	EM30,DH30,DT30,DF30	:ERROR	ITEM	# 30
	001742	047363	067073	073016	.WORD	EM31,DH31,DT31,DF31	:ERROR	ITEM	# 31
	001752	047363	067073	073072	.WORD	EM32,DH32,DT32,DF32	:ERROR	ITEM	# 32
	001762	047431	067161	073166	.WORD	EM33,DH33,DT33,DF33	:ERROR	ITEM	# 33
	001772	047472	067161	073244	.WORD	EM34,DH34,DT34,DF34	:ERROR	ITEM	# 34
	002002	047574	067161	073244	.WORD	EM35,DH35,DT35,DF35	:ERROR	ITEM	# 35
	002012	047676	067161	073244	.WORD	EM36,DH36,DT36,DF36	:ERROR	ITEM	# 36
	002022	047777	067161	073244	.WORD	EM37,DH37,DT37,DF37	:ERROR	ITEM	# 37
	002032	050100	067161	073166	.WORD	EM40,DH40,DT40,DF40	:ERROR	ITEM	# 40
	002042	050251	000000	073316	.WORD	EM41,DH41,DT41,DF41	:ERROR	ITEM	# 41
	002052	050306	067264	073350	.WORD	EM42,DH42,DT42,DF42	:ERROR	ITEM	# 42
	002062	050427	067264	073350	.WORD	EM43,DH43,DT43,DF43	:ERROR	ITEM	# 43
	002072	050550	000000	073426	.WORD	EM44,DH44,DT44,DF44	:ERROR	ITEM	# 44
	002102	050550	067366	073476	.WORD	EM45,DH45,DT45,DF45	:ERROR	ITEM	# 45
	002112	050613	067405	073552	.WORD	EM46,DH46,DT46,DF46	:ERRCR	ITEM	# 46
	002122	050671	067366	073640	.WORD	EM47,DH47,DT47,DF47	:ERROR	ITEM	# 47
	002132	051007	067431	073244	.WORD	EM50,DH50,DT50,DF50	:ERROR	ITEM	# 50
	002142	051105	067431	073672	.WORD	EM51,DH51,DT51,DF51	:ERROR	ITEM	# 51
	002152	051146	066201	073640	.WORD	EM52,DH52,DT52,DF52	:ERROR	ITEM	# 52
	002162	051267	067073	073730	.WORD	EM53,DH53,DT53,DF53	:ERROR	ITEM	# 53
	002172	051464	067503	073750	.WORD	EM54,DH54,DT54,DF54	:ERROR	ITEM	# 54
	002202	051530	066201	073640	.WORD	EM55,DH55,DT55,DF55	:ERROR	ITEM	# 55
	002212	051651	067073	073730	.WORD	EM56,DH56,DT56,DF56	:ERROR	ITEM	# 56

002222	052046	067503	073750	.WORD	EM57,DH57,DT57,DF57	:ERROR	ITEM # 57
002232	052112	067073	073730	.WORD	EM60,DH60,DT60,DF60	:ERROR	ITEM # 60
002242	052307	067503	073750	.WORD	EM61,DH61,DT61,DF61	:ERROR	ITEM # 61
002252	052353	067503	073750	.WORD	EM62,DH62,DT62,DF62	:ERROR	ITEM # 62
002262	052545	067503	073750	.WORD	EM63,DH63,DT63,DF63	:ERROR	ITEM # 63
002272	052737	067613	074006	.WORD	EM64,DH64,DT64,DF64	:ERROR	ITEM # 64
002302	052737	067544	074006	.WORD	EM65,DH65,DT65,DF65	:ERROR	ITEM # 65
002312	053073	067503	073750	.WORD	EM66,DH66,DT66,DF66	:ERROR	ITEM # 66
002322	053136	066201	072640	.WORD	EM67,DH67,DT67,DF67	:ERROR	ITEM # 67
002332	053367	066201	074026	.WORD	EM70,DH70,DT70,DF70	:ERROR	ITEM # 70
002342	053512	067031	074026	.WORD	EM71,DH71,DT71,DF71	:ERROR	ITEM # 71
002352	053614	067073	074074	.WORD	EM72,DH72,DT72,DF72	:ERROR	ITEM # 72
002362	053670	067503	073750	.WORD	EM73,DH73,DT73,DF73	:ERROR	ITEM # 73
002372	053730	066201	072640	.WORD	EM74,DH74,DT74,DF74	:ERROR	ITEM # 74
002402	054161	066201	074026	.WORD	EM75,DH75,DT75,DF75	:ERROR	ITEM # 75
002412	054304	067031	074026	.WORD	EM76,DH76,DT76,DF76	:ERROR	ITEM # 76
002422	054406	067073	074074	.WORD	EM77,DH77,DT77,DF77	:ERROR	ITEM # 77
002432	054462	067503	073750	.WORD	EM100,DH100,DT100,DF100	:ERROR	ITEM # 100
002442	054522	066201	074026	.WORD	EM101,DH101,DT101,DF101	:ERROR	ITEM # 101
002452	054646	067073	074026	.WORD	EM102,DH102,DT102,DF102	:ERROR	ITEM # 102
002462	054720	067031	074026	.WORD	EM103,DH103,DT103,DF103	:ERROR	ITEM # 103
002472	055023	067503	073750	.WORD	EM104,DH104,DT104,DF104	:ERROR	ITEM # 104
002502	055064	066201	074026	.WORD	EM105,DH105,DT105,DF105	:ERROR	ITEM # 105
002512	055211	067073	074074	.WORD	EM106,DH106,DT106,DF106	:ERROR	ITEM # 106
002522	055264	067031	074026	.WORD	EM107,DH107,DT107,DF107	:ERROR	ITEM # 107
002532	055370	067503	073750	.WORD	EM110,DH110,DT110,DF110	:ERROR	ITEM # 110
002542	055432	067031	074114	.WORD	EM111,DH111,DT111,DF111	:ERROR	ITEM # 111
002552	055432	067701	074114	.WORD	EM112,DH112,DT112,DF112	:ERROR	ITEM # 112
002562	055534	067031	074114	.WORD	EM113,DH113,DT113,DF113	:ERROR	ITEM # 113
002572	055534	067701	074114	.WORD	EM114,DH114,DT114,DF114	:ERROR	ITEM # 114
002602	055432	070120	074114	.WORD	EM115,DH115,DT115,DF115	:ERROR	ITEM # 115
002612	055534	070120	074114	.WORD	EM116,DH116,DT116,DF116	:ERROR	ITEM # 116
002622	055636	066321	072360	.WORD	EM117,DH117,DT117,DF117	:ERROR	ITEM # 117
002632	055772	070404	072360	.WORD	EM120,DH120,DT120,DF120	:ERROR	ITEM # 120
002642	056126	066201	073640	.WORD	EM121,DH121,DT121,DF121	:ERROR	ITEM # 121
002652	056245	067431	073244	.WORD	EM122,DH122,DT122,DF122	:ERROR	ITEM # 122
002662	056344	067431	073672	.WORD	EM123,DH123,DT123,DF123	:ERROR	ITEM # 123
002672	056405	066321	074126	.WORD	EM124,DH124,DT124,DF124	:ERROR	ITEM # 124
002702	056500	066321	074126	.WORD	EM125,DH125,DT125,DF125	:ERROR	ITEM # 125
002712	056570	066201	074114	.WORD	EM126,DH126,DT126,DF126	:ERROR	ITEM # 126
002722	056777	067503	074114	.WORD	EM127,DH127,DT127,DF127	:ERROR	ITEM # 127
002732	057212	070404	072360	.WORD	EM130,DH130,DT130,DF130	:ERROR	ITEM # 130
002742	057312	067503	074212	.WORD	EM131,DH131,DT131,DF131	:ERROR	ITEM # 131
002752	057352	067503	074212	.WORD	EM132,DH132,DT132,DF132	:ERROR	ITEM # 132
002762	057412	070474	074254	.WORD	EM133,DH133,DT133,DF133	:ERROR	ITEM # 133
002772	057451	070474	074254	.WORD	EM134,DH134,DT134,DF134	:ERROR	ITEM # 134
003002	057510	070474	074254	.WORD	EM135,DH135,DT135,DF135	:ERROR	ITEM # 135
003012	057547	070474	074254	.WORD	EM136,DH136,DT136,DF136	:ERROR	ITEM # 136
003022	057412	070604	074326	.WORD	EM137,DH137,DT137,DF137	:ERROR	ITEM # 137
003032	057451	070604	074326	.WORD	EM140,DH140,DT140,DF140	:ERROR	ITEM # 140
003042	057510	070604	074326	.WORD	EM141,DH141,DT141,DF141	:ERROR	ITEM # 141
003052	057547	070604	074326	.WORD	EM142,DH142,DT142,DF142	:ERROR	ITEM # 142
003062	057606	070474	074254	.WORD	EM143,DH143,DT143,DF143	:ERROR	ITEM # 143
003072	057641	070474	074254	.WORD	EM144,DH144,DT144,DF144	:ERROR	ITEM # 144
003102	057606	070604	074326	.WORD	EM145,DH145,DT145,DF145	:ERROR	ITEM # 145
003112	057641	070604	074326	.WORD	EM146,DH146,DT146,DF146	:ERROR	ITEM # 146
003122	057674	067503	074254	.WORD	EM147,DH147,DT147,DF147	:ERROR	ITEM # 147

003132	057674	070774	074254	.WORD	EM150,DH150,DT150,DF150	:ERROR ITEM # 150
003142	057674	070604	074326	.WORD	EM151,DH151,DT151,DF151	:ERROR ITEM # 151
003152	057726	070474	074254	.WORD	EM152,DH152,DT152,DF152	:ERROR ITEM # 152
003162	057726	070604	074326	.WORD	EM153,DH153,DT153,DF153	:ERROR ITEM # 153
003172	057760	071065	074346	.WORD	EM154,DH154,DT154,DF154	:ERROR ITEM # 154
003202	060212	071065	074346	.WORD	EM155,DH155,DT155,DF155	:ERROR ITEM # 155
003212	060445	067503	074254	.WORD	EM156,DH156,DT156,DF156	:ERROR ITEM # 156
003222	060662	067503	074254	.WORD	EM157,DH157,DT157,DF157	:ERROR ITEM # 157
003232	061101	067503	074254	.WORD	EM160,DH160,DT160,DF160	:ERROR ITEM # 160
003242	061306	067503	074254	.WORD	EM161,DH161,DT161,DF161	:ERROR ITEM # 161
003252	061513	067503	074254	.WORD	EM162,DH162,DT162,DF162	:ERROR ITEM # 162
003262	061560	067503	074254	.WORD	EM163,DH163,DT163,DF163	:ERROR ITEM # 163
003272	061625	066321	072360	.WORD	EM164,DH164,DT164,DF164	:ERROR ITEM # 164
003302	061672	066321	072360	.WORD	EM165,DH165,DT165,DF165	:ERROR ITEM # 165
003312	061737	067503	074254	.WORD	EM166,DH166,DT166,DF166	:ERROR ITEM # 166
003322	062047	067503	074254	.WORD	EM167,DH167,DT167,DF167	:ERROR ITEM # 167
003332	062306	067503	074254	.WORD	EM170,DH170,DT170,DF170	:ERROR ITEM # 170
003342	062416	067503	074254	.WORD	EM171,DH171,DT171,DF171	:ERROR ITEM # 171
003352	062655	067503	074254	.WORD	EM172,DH172,DT172,DF172	:ERROR ITEM # 172
003362	063114	067503	074254	.WORD	EM173,DH173,DT173,DF173	:ERROR ITEM # 173
003372	063353	067503	074254	.WORD	EM174,DH174,DT174,DF174	:ERROR ITEM # 174
003402	063612	067503	074254	.WORD	EM175,DH175,DT175,DF175	:ERROR ITEM # 175
003412	064051	067503	074254	.WORD	EM176,DH176,DT176,DF176	:ERROR ITEM # 176
003422	064206	071125	074360	.WORD	EM177,DH177,DT177,DF177	:ERROR ITEM # 177
003432	064242	067503	074254	.WORD	EM200,DH200,DT200,DF200	:ERROR ITEM # 200
003442	064377	067503	074254	.WORD	EM201,DH201,DT201,DF201	:ERROR ITEM # 201
003452	064534	067503	074254	.WORD	EM202,DH202,DT202,DF202	:ERROR ITEM # 202
003462	064671	067503	074254	.WORD	EM203,DH203,DT203,DF203	:ERROR ITEM # 203
003472	065026	067503	074254	.WORD	EM204,DH204,DT204,DF204	:ERROR ITEM # 204
003502	065163	066321	072360	.WORD	EM205,DH205,DT205,DF205	:ERROR ITEM # 205
003512	065230	067503	074254	.WORD	EM206,DH206,DT206,DF206	:ERROR ITEM # 206
003522	065275	067503	074254	.WORD	EM207,DH207,DT207,DF207	:ERROR ITEM # 207
003532	065417	067503	074254	.WORD	EM210,DH210,DT210,DF210	:ERROR ITEM # 210
003542	045560	071171	074370	.WORD	EM211,DH211,DT211,DF211	:ERROR ITEM # 211
003552	045614	066201	074406	.WORD	EM212,DH212,DT212,DF212	:ERROR ITEM # 212
003562	045646	066201	074406	.WORD	EM213,DH213,DT213,DF213	:ERROR ITEM # 213
003572	065474	067503	074254	.WORD	EM214,DH214,DT214,DF214	:ERROR ITEM # 214

1806

000046 003602
000052 000046
000052 036432
000052 000052
000052 000000
000052 003602

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


1808

000024 003602
000044 000024
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
:*****
.$X=      ;;SAVE CURRENT LOCATION
.=24     ;;SET POWER FAIL TO POINT TO START OF PROGRAM
200      ;;FOR APT START UP
.=44     ;;POINT TO APT INDIRECT ADDRESS PNTR.
$APTHDR  ;;POINT TO APT HEADER BLOCK
.=.$X    ;;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)
$STMT:  .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)
```

1810
1811 003616

```

003616 012706 001100
003622 005026
003624 022706 001140
003630 001374
003632 012706 001100

003636 012737 036516 000020
003644 012737 000340 000022
003652 012737 037146 000030
003660 012737 000340 000032
003666 012737 041540 000034
003674 012737 000340 000036
003702 012737 041624 000024
003710 012737 000340 000026
003716 013737 036124 036112
003724 005037 001302
003730 005037 001304
003734 112737 000001 001115

003742 012737 036476 000014
003750 012737 000340 000016
003756 012737 000002 036476
003764 012737 004012 000010
003772 005046
003774 012746 004002
004000 000006
004002 012737 000006 036476 64$
004010 000402
004012 062706 000010 65$
004016 012737 000012 000010 66$
004024 005037 036504
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 $ESCAPE ::CLEAR THE ESCAPE ON ERROR ADDRESS
MOVB #1,$ERMAX ::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$,-(SP) ::AND PC
RTT ::TRY THE RTT
64$: MOV #RTT,$RTRN ::RTT IS LEGAL--SET $RTRN TO A RTT
BR 66$
65$: ADD #10,SP ::RTT ILLEGAL--CLEAN OFF THE STACK
66$: MOV #RESVEC+2,@RESVEC ::RESTORE TRAP CATCHER
CLR $TBIT ::CLEAR "T" BIT SWITCH
MOV #,$LPADR ::INITIALIZE THE LOOP ADDRESS FOR SCOPE
MOV #,$LPERR ::SETUP THE ERROR LOOP ADDRESS
::SIZE FOR A HARDWARE SWITCH REGISTER. IF NOT FOUND OR IT IS
::EQUAL TO A "-1", SETUP FOR A SOFTWARE SWITCH REGISTER.
MOV @ERRVEC,-(SP) ::SAVE ERROR VECTOR
MOV #67$,@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
67$: MOV #68$,(SP) ::SET UP FOR TRAP RETURN
RTI
68$: MOV #SWREG,SWR ::POINT TO SOFTWARE SWR
MOV #DISPREG,DISPLAY
69$: 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 #SWREG,SWR ::NO,USE APT SWITCH REGISTER
    
```



```

1812 004154 005227 177777
004160 001047
004162 022737 036432 000042
004170 001443
004172 104401 004240
004176 005737 000042
004202 001012
004204 123727 001336 000001
004212 001406
004214 023727 001140 000176
004222 001005
004224 104406
004226 000403
004230 112737 000001 001134
004236 000420
004236 000420
1813 004300 104401 004306
004304 000431
004370 104401 004376
004374 000426
1815 004452 005037 036512
1816 004456

70$:
.SBTTL TYPE PROGRAM NAME
::TYPE THE NAME OF THE PROGRAM IF FIRST PASS
INC #-1 ::FIRST TIME?
BNE 71$ ::BRANCH IF NO
CMP #SENDAD,@#42 ::ACT-11?
BEQ 71$ ::BRANCH IF YES
TYPE ,72$ ::TYPE ASCIZ STRING
.SBTTL GET VALUE FOR SOFTWARE SWITCH REGISTER
TST @#42 ::ARE WE RUNNING UNDER XXDP/ACT?
BNE 73$ ::BRANCH IF YES
CMPB $ENV,#1 ::ARE WE RUNNING UNDER APT?
BEQ 73$ ::BRANCH IF YES
CMP SWR,#SWREG ::SOFTWARE SWITCH REG SELECTED?
BNE 74$ ::BRANCH IF NO
GTSWR ::GET SOFT-SWR SETTINGS
BR 74$
MOV B #1,$AUTOB ::SET AUTO-MODE INDICATOR
BR 71$ ::GET OVER THE ASCIZ
::72$: .ASCIZ <CRLF>*CKFPADO FP11F FLTG PNT PRT A*<CRLF>
71$:
TYPE ,76$ ::TYPE ASCIZ STRING
BR 75$ ::GET OVER THE ASCIZ
::76$: .ASCIZ !EOP MSGS WILL PRINT EVERY 4 PASSES (15 SECONDS)!<CRLF>
75$:
TYPE ,78$ ::TYPE ASCIZ STRING
BR 77$ ::GET OVER THE ASCIZ
::78$: .ASCIZ !HIT ANY KEY TO DISABLE/ENABLE EOP MESSAGES!<CRLF>
77$:
CLR EPENDS ;CLR THE ENABLE/DISABLE EOP FLAG ;DPM002
LOOP:

```

1835

```
.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
*SMD ARE USED.
*NOTE THAT A MASK MUST BE USED BECAUSE SOME OF THE FPS BITS CANNOT
*BE SFT.
*
*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.
*
*****
```

```
1836 004456 000004
1837 004460 005037 004736
1838 004464 012737 004526 001110
1839 004472 012700 177777
1840 004476 012737 004740 000244
1841 004504 012737 004752 000010
1842 004512 005002
1843 004514 005102
1844 004516 005003
1845 004520 012737 005004 000004
1846
1847
1848 004526
1849 004526 010004
1850 004530 042704 030020
1851 004534 170104
1852
1853 004536 012701 177777
1854 004542 170201
1855 004544 012737 042534 000244
1856 004552 010004
1857 004554 042704 030020
1858 004560 012737 042566 000004
1859 004566 012737 042604 000010
1860 004574 020401
1861
1862 004576 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 INSTRUCTION
                               ;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.
```



```

1863
1864 004600 077026          A2:   SOB      R0,A1      ;OTHERWISE DECREMENT COUNT PATTERN
1865 004602 000425          BR      A5      ;UNTIL IT IS ZERO.
1866
1867 004604 005237 004736   A3:   INC      AERFLG   ;RECORD ERROR.
1868 004610 050003          BIS      R0,R3    ;COMPUTE 'OR' OF FAILING PATTERNS.
1869 004612 010005          MOV      R0,R5    ;COMPUTE 'AND' OF FAILING PATTERNS.
1870 004614 005105          COM      R5
1871 004616 040502          BIC      R5,R2
1872
1873 004620 022737 000005 004736   CMP      #5,AERFLG ;SEE IF MORE THAN 5 ERRORS HAVE
1874 004626 103412          BLO      A05     ;OCCURRED. BR IF YES.
1875
1876
1877 004630 012737 004520 001236   MOV      #A1,$TMP2 ;OTHERWISE
1878 004636 010037 001240          MOV      R0,$TMP3 ;REPORT ERROR.
1879 004642 010137 001242          MOV      R1,$TMP4
1880 004646 010437 001244          MOV      R4,$TMP5
1881 004652 104001          A4:   ERROR    +1
1882
1883 004654 000751          A05:   BR      A2      ;CONTINUE TESTING.
1884
1885 004656 005737 004736   A5:   TST      AERFLG   ;SEE IF ANY ERRORS OCCURRED.
1886 004662 001471          BEQ      ADONE    ;IF NOT GO TO NEXT TEST.
1887 004664 032777 020000 174246   BIT      #SW13,@SWR ;OTHERWISE SEE IF A SUMMARY
1888 004672 001404          BEQ      A6       ;SHOULD BE TYPED.
1889 004674 032777 000200 174236   BIT      #SW7,@SWR
1890 004702 001461          BEQ      ADONE
1891
1892 004704          A6:
1893 004704 010237 001236   MOV      R2,$TMP2 ;TYPE ERROR SUMMARY.
1894 004710 010337 001240   MOV      R3,$TMP3
1895 004714 012737 004730 001116   MOV      #A7,$ERRPC
1896 004722 112737 000002 001114   MOV      #2,$ITEMB
1897 004730 004737 042010   A7:   JSR      PC,ERTYPE
1898 004734 000444          BR      ADONE
1899
1900 004736 000000   AERFLG: .WORD 0
1901
1902          ;UNABLE TO DECODE FPP INSTRUCTION. TRAPPED TO 244.
1903 004740 011637 001236   AERR1: MOV      (SP),$TMP2 ;SAVE PC OF TRAP.
1904 004744 022626          CMP      (SP)+,(SP)+
1905 004746 104010   1$:   ERFOR    +10
1906 004750 000436          BR      ADONE
1907
1908          ;UNABLE TO DECODE INSTRUCTION. TRAPPED TO 10.
1909 004752 021627 004530   AERR2: CMP      (SP),#A11+2 ;DID TRAP OCCUR OF FPP INSTRUCTION?
1910 004756 001405          BEQ      1$
1911 004760 021627 004544          CMP      (SP),#A12+2
1912 004764 001402          BEQ      1$
1913 004766 000137 042604          JMP      CPTWO
1914
1915
1916 004772 011637 001236   1$:   MOV      (SP),$TMP2 ;IF NOT FPP INSTRUCTION THEN
1917 004776 022626          CMP      (SP)+,(SP)+ ;REPORT SPURIOUS TRAP TO 10.
1918 005000 104011   2$:   ERROR    +11
1919 005002 000421          BR      ADONE ;OTHERWISE REPORT IR DECIDE ERROR.
    
```


1946

```

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

```

```

1947 00505C 000004
1947 005052 012737 005064 001110
1948 005060 012700 000017
1949
1950 005064
1951 005064 170100
1952
1953 005066
1954 005066 170000
1955
1956 005070 013703 177776
1957 005074 042703 177760
1958 005100 020003
1959 005102 001002
1960
1961 005104 077011
1962 005106 000422
1963
1964 005110
1965 005110 170201
1966 005112 012737 005066 001236
1967 005120 020001
1968 005122 001006
1969
1970 005124 010337 001240
1971 005130 010037 001242
1972 005134 104003
1973 005136 000762
1974
1975 005140
1976 005140 010037 001240
1977 005144 010137 001242
1978 005150 104004
1979 005152 000754
1980
1981 005154
      005154 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

1$:    MOV    R3,$TMP3
        MOV    R0,$TMP4
        ERROR +3
        BR     B3

BERR1: MOV    R0,$TMP3
        MOV    R1,$TMP4

1$:    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?).

```

1982


```

2036
2037 005360          1$:
      005360 012737 005366 001110  MOV    #C4,$LPERR    ;SET UP THE LOOP ON ERROR ADDRESS.
2038 005366 005000  C4:    CLR    R0            ;CLEAR FPS.
2039 005370 170100    LDFPS  R0
2040 005372 012737 005400 001236  MOV    #C45,$TMP2
2041
2042 005400 170011    C45:   SETD           ;SET FD BIT.
2043
2044 005402 170201    STFPS  R1            ;GET RESULT.
2045 005404 012702 000200  MOV    #200,R2
2046 005410 020102    CMP    R1,R2        ;RESULT CORRECT?
2047 005412 001402    BEQ   1$
2048 005414 004737 005654  JSR   PC,CERR1
2049
2050 005420          1$:
      005420 012737 005426 001110  MOV    #C5,$LPERR    ;SET UP THE LOOP ON ERROR ADDRESS.
2051 005426 012737 000204 001250  C5:    MOV    #204,$TMP7
2052 005434 012737 043713 001252  MOV    #SETI1,$TMP10
2053 005442 005000    CLR    R0
2054
2055 005444 170100    LDFPS  R0            ;CLEAR FPS
2056 005446 012737 005454 001236  MOV    #C55,$TMP2
2057
2058 005454 170002    C55:   SETI           ;CLEAR FL BIT.
2059
2060 005456 170201    STFPS  R1            ;GET RESULT.
2061 005460 005002    CLR    R2
2062 005462 020201    CMP    R2,R1        ;RESULT CORRECT?
2063 005464 001402    BEQ   1$
2064 005466 004737 005654  JSR   PC,CERR1
2065
2066 005472          1$:
      005472 012737 005500 001110  MOV    #C6,$LPERR    ;SET UP THE LOOP ON ERROR ADDRESS.
2067 005500 012700 147757  C6:    MOV    #147757,R0   ;PUT 147757 INTO FPS
2068 005504 170100    LDFPS  R0
2069 005506 012737 005514 001236  MOV    #C65,$TMP2
2070
2071 005514 170002    C65:   SETI           ;CLEAR FL BIT.
2072
2073 005516 170201    STFPS  R1            ;GET THE RESULT.
2074 005520 012702 147657  MOV    #147657,R2   ;RESULT CORRECT?
2075 005524 020102    CMF   R1,R2
2076 005526 001402    BEQ   1$
2077 005530 004737 005752  JSR   PC,CERR2
2078
2079 005534          1$:
      005534 012737 005542 001110  MOV    #C7,$LPERR    ;SET UP THE LOOP ON ERROR ADDRESS.
2080 005542 012737 000205 001250  C7:    MOV    #205,$TMP7
2081 005550 012737 043721 001252  MOV    #SETL1,$TMP10
2082 005556 012700 147757  MOV    #147757,R0
2083 005562 170100    LDFPS  R0            ;SET FPS TO 147757.
2084 005564 012737 005572 001236  MOV    #C75,$TMP2
2085
2086 005572 170012    C75:   SETL           ;SET FL BIT.
2087
2088 005574 170201    STFPS  R1            ;GET THE RESULT.
    
```

```

2089 005576 012702 147757          MOV    #147757,R2
2090 005602 020102          CMP    R1,R2                ;RESULT CORRECT?
2091 005604 001402          BEQ   1$
2092 005606 004737 005752          JSR   PC,CERR2
2093
2094 005612          1$:
      005612 012737 005620 001110          MOV    #C8,$LPERR        ;SET UP THE LOOP ON ERROR ADDRESS.
2095 005620 005000          C8:    CLR    R0
2096 005622 170100          LDFPS R0                    ;CLEAR FPS.
2097 005624 012737 005632 001236          MOV    #C85,$TMP2
2098
2099 005632 170012          C85:  SETL                   ;SET FL BIT.
2100
2101 005634 170201          STFPS R1
2102 005636 012702 000100          MOV    #100,R2
2103 005642 020102          CMP    R1,R2                ;RESULT CORRECT.
2104 005644 001402          BEQ   1$
2105 005646 004737 005654          JSR   PC,CERR1
2106
2107 005652 000522          1$:   BR     CDONE
2108
2109          ;THESE ARE ERROR ANALYSIS ROUTINES:
2110 005654 010103          CERR1: MOV    R1,R3
2111 005656 032703 177477          BIT    #177477,R3          ;ARE ANY OTHER BITS SET?
2112 005662 001401          BEQ   2$
2113 005664 000503          1$:   BR     CERR4
2114
2115 005666 022703 000300          2$:   CMP    #300,R3        ;ARE BOTH FD AND FL SET?
2116 005672 001774          BEQ   1$
2117 005674 032703 000300          BIT    #300,R3            ;ARE THEY BOTH CLEAR?
2118 005700 001771          BEQ   1$
2119
2120 005702 032703 000200          BIT    #200,R3            ;IS FD SET?
2121 005706 001407          BEQ   3$
2122 005710 012737 043705 001254          MOV    #SETD1,$TMP11
2123 005716 012737 000203 001246          MOV    #203,$TMP6
2124 005724 000452          BR     CERR3
2125
2126 005726 032703 000100          3$:   BIT    #100,R3        ;IS FL SET
2127 005732 001754          BEQ   1$
2128 005734 012737 043721 001254          MOV    #SETL1,$TMP11
2129 005742 012737 000205 001246          MOV    #205,$TMP6
2130 005750 000440          BR     CERR3
2131
2132 005752 010103          CERR2: MOV    R1,R3
2133 005754 005103          COM   R3
2134
2135 005756 032703 177477          BIT    #177477,R3          ;ARE ANY OTHER BITS SET?
2136 005762 001401          BEQ   2$
2137 005764 000443          1$:   BR     CERR4
2138
2139 005766 032703 000300          2$:   BIT    #300,R3        ;ARE BOTH FD AND FL SET?
2140 005772 001774          BEQ   1$
2141 005774 032701 000300          BIT    #300,R1            ;ARE THEY BOTH CLEAR?
2142 006000 001771          BEQ   1$
2143
2144 006002 032701 000200          BIT    #200,R1            ;IS FD CLEAR?
    
```



```

2145 006006 001007          BNE      3$
2146 006010 012737 043677 001254  MOV     #SETF1,$TMP11
2147 006016 012737 000202 001246  MOV     #202,$TMP6
2148 006024 000412          BR      CERR3
2149
2150 006026 032701 000100      3$:    BIT     #100,R1
2151 006032 001354          BNE     1$                ;IS FL CLEAR.
2152 006034 012737 043713 001254  MOV     #SETI1,$TMP11
2153 006042 012737 000204 001246  MOV     #204,$TMP6
2154 006050 000400          BR      CERR3
2155
2156          ;REPORT THE ERRORS:
2157 006052          CERR3:
2158 006052 010137 001240      MOV     R1,$TMP3
2159 006056 010237 001242      MOV     R2,$TMP4
2160 006062 012637 006116      MOV     (SP)+,CPC
2161 006066 104012          1$:    ERROR  +12
2162 006070 000177 000022      JMP     @CPC
2163
2164          CERR4:
2165 006074 010137 001240      MOV     R1,$TMP3
2166 006100 010237 001242      MOV     R2,$TMP4
2167 006104 012637 006116      MOV     (SP)+,CPC
2168 006110 104013          1$:    ERROR  +13
2169 006112 000177 000000      JMP     @CPC
2170
2171 006116 000000          CPC:   .WORD  0
2172
2173 006120          CDONE:  RSETUP
      006120 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?).
    
```

2174
2175

2195

```

.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.
:
:*****
    
```

```

2196 006122 000004
2196 006124 012737 006152 001110
2197 006132 012705 170003
2198 006136 012737 006342 000004
2199 006144 012737 006246 000244
2200
2201 006152 005000
2202 006154 170100
2203 006156 005002
2204 006160 010537 006176
2205 006164 010537 001244
2206 006170 012737 006176 001236
2207 006176 000000
2208 006200 170000
2209 006202 005202
2210 006204 005202
2211
2212 006206 170201
2213 006210 010137 001240
2214 006214 104016
2215
2216 006216 022705 170010
2217 006222 001003
2218 006224 012705 170013
2219 006230 000750
2220
2221 006232 022705 170077
2222 006236 001001
2223 006240 000452
2224 006242 005205
2225 006244 000742
2226
2227 006246 022716 006200
2228 006252 001402
2229 006254 000137 042534
2230

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
      CLR  R2
      MOV  R5,D2
      MOV  R5,$TMP5
      MOV  #D2,$TMP2

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

      STFPS R1
      MOV  R1,$TMP3
      ERROR +16

D5:    CMF  #170010,R5
      BNE  D6
      MOV  #170013,R5
      BR   D1

D6:    CMP  #170077,R5
      BNE  D7
      BR   DDONE

D7:    INC  R5
      BR   D1

DERR1: CMP  #D3,(SP)
      BEQ  1$
      JMP  FPSPUR
    
```



```

2231 006260 022626          1$:  CMP      (SP)+,(SP)+
2232 006262 170201          STFPS   R1           ;GET THE FPS AND SEE IF IT IS
2233 006264 022701 100000    CMP      #100000,R1    ;SET CORRECTLY.
2234 006270 001406          BEQ     3$
2235
2236 006272 012737 100000 001240  MOV     #100000,$TMP3
2237 006300 010137 001242    MOV     R1,$TMP4
2238 006304 104017          2$:  ERROR  +17
2239
2240 006306 012704 000001    3$:  MOV     #1,R4
2241 006312 170304          D8:  STST   R4           ;GET THE FEC CODE. NOTE THAT
2242
2243
2244
2245 006314 022704 000002    CMP     #2,R4           ;IF THE DESTINATION MODE IS
2246 006320 001001          BNE    D9              ;IMPROPERLY DECODED AN ODD
2247 006322 000735          BR     D5              ;ADDRESS TRAP TO 4 SHOULD OCCUR.
2248
2249 006324          D9:  ;REPORT STST FAILURE
2250 006324 012737 006312 001240  MOV     #D8,$TMP3
2251 006332 010437 001242    MOV     R4,$TMP4
2252 006336 104020          1$:  ERROR  +20
2253 006340 000726          BR     D5
2254
2255 006342 022716 006314    DERR2: CMP     #D8+2,(SP) ;DID THE TRAP OCCUR ON THE
2256 006346 001402          BEQ    D10            ;STST INSTRUCTION?
2257 006350 000137 042566    JMP     CPSPUR
2258
2259 006354          D10:
2260 006354 011637 001236    MOV     (SP),$TMP2
2261 006360 022626          CMP     (SP)+,(SP)+
2262 006362 104021          1$:  ERROR  +21
2263 006364 000714          BR     D5
2264
2265 006366          DDONE:
      006366 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?).

2266
2267
  
```

2275

```
.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.
*
```

```
*****
TST5:  SCOPE
2276 006370 030004          MOV      #E1,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
2277 006372 012737 006406 001110  MOV      #EERR2,FPVECT  ;SETUP FOR THE INTERRUPT.
2277 006400 012737 006502 000244
2278
2279 006406 012700 040000  E1:      MOV      #40000,R0
2280 006412 170100          LDFPS   R0              ;SET FID.
2281 006414 012737 006422 001236  MOV      #E3,$TMP2
2282 006422
2283 006422 170020          E2:      .WORD   170020    ;ILLEGAL FPP INSTRUCTION.
2284 006424 170000          E3:
2285          E4:      CFCC
2286 006426 170201          STFPS   R1              ;SEE IF ERROR WAS DETECTED.
2287 006430 022701 140000  CMP      #140000,R1
2288 006434 001005          BNE     EERR0
2289
2290 006436 170304          STST    R4              ;SEE IF FEC=2
2291 006440 022704 000002  CMP      #2,R4
2292 006444 001010          BNE     EERR1
2293 006446 000431          BR      EDONE
2294
2295 006450          EERR0:          ;REPORT FPS INCORRECTLY SET.
2296 006450 010137 001240  MOV      R1,$TMP3
2297 006454 012737 140000 001242  MOV      #140000,$TMP4
2298 006462 104022          1$:      ERROR   +22
2299 006464 000422          BR      EDONE
2300
2301 006466          EERR1:          ;REPORT FEC NOT 2.
2302 006466 010537 001240  MOV      R5,$TMP3
2303 006472 010437 001242  MOV      R4,$TMP4
2304 006476 104023          1$:      ERROR   +23
2305 006500 000414          BR      EDONE
2306
2307 006502 021627 006424  EERR2:  CMP      (SP),#E4      ;DID THE ILLEGAL INSTRUCTION TRAP?
2308 006506 001402          BEC     1$
2309 006510 000137 042534          JMP     FPSPUR
2310
2311 006514          1$:
2312 006514 011637 001236  MOV      (SP),$TMP2
2313 006520 022626          CMP     (SP)+,(SP)+
2314 006522 170201          STFPS   R1
2315 006524 010137 001240  MOV      R1,$TMP3
2316 006530 104024          2$:      ERROR   +24
2317
2318 006532          EDONE:
2318 006532 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?).

2319
2320

2333

```
.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.
*****
```

```
TST6:  SCOPE
2334 006534 000004
2335 006536 012737 006536 001110 F1:  MOV      #F1,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
2336 006544 012737 006614 001236    MOV      #F3,$TMP2
2337 006552 005000    CLR      R0
2338 006554 170100    LDFPS   R0
2339 006556 170011    SETD
2340 006560 012701 010352    MOV      #FDAT10,R1      ;SET UP THE LOAD DATA.
2341 006564 012702 010416    MOV      #FXDAT0,R2
2342 006570 012703 000010    MOV      #10,R3
2343
2344 006574 012221    F2:  MOV      (R2)+,(R1)+
2345 006576 077302    SOB
2346
2347 006600 012700 010362    MOV      #FDAT14,R0      ;SETUP R0 FOR THE LDD (R0),ACO.
2348 006604 012737 010036 000004    MOV      #FERR20,ERRVECT ;IF THE SRC FLOWS FAIL THEN
2349
2350 006612 005003    CLR      R3              ;AN ODD ADDRESS MAY OCCUR.
2351
2352 006614 172410    F3:  LDD      (R0),ACO
2353 006616 005203    F4:  INC      R3
2354 006620 005203    INC      R3
2355
2356 006622 020027 010362    CMP      R0,#FDAT14      ;WAS R0 AFFECTED?
2357 006626 001402    BEQ      F5
2358 006630 000137 007202    JMP      FERR1
2359
2360 006634 020327 000002    F5:  CMF      R3,#2
2361 006640 001402    BEQ      1$              ;SEE IF THE PC WAS ADVERSELY
2362 006642 000137 007300    JMP      FERR2          ;AFFECTED DURING THE INSTRUCTION.
2363
2364 006646 012701 010352    1$:  MOV      #FDAT10,R1      ;MAKE SURE THE SOURCE DATA WAS
2365 006652 012702 010416    MOV      #FXDAT0,R2      ;NOT AFFECTED.
2366 006656 012703 000010    MOV      #10,R3
2367 006662 022122    2$:  CMP      (R1)+,(R2)+
2368 006664 001402    BEQ      3$
2369 006666 000137 007144    JMP      FERRO
2370 006672 077305    3$:  SOB
2371
2372 006674 170201    STFPS   R1              ;MAKE SURE THE FPS IS CORRECT.
2373 006676 022701 000200    CMP      #200,R1
2374 006702 001402    BEQ      F6
```


2375	006704	000137	010016		JMP	FERR11	
2376							
2377	006710			F6:			
	006710	012737	006716	001110	MOV	#1\$, \$LPERR	:SET UP THE LOOP ON ERROR ADDRESS.
2378	006716	012737	006760	001236	1\$: MOV	#F10, \$TMP2	
2379							
2380	006724	012703	177777		MOV	#-1, R3	
2381	006730	012704	000010		MOV	#10, R4	
2382	006734	012705	010374		MOV	#FDAT00, R5	:SET UP THE OUTPUT DATA BUFFER.
2383	006740	010325		F7:	MOV	R3, (R5)+	
2384	006742	077402			SOB	R4, F7	
2385							
2386	006744	012700	010404		MOV	#FDAT04, R0	:SET UP R0 FOR DST MODE 1 REG 0.
2387	006750	012737	010204	000004	MOV	#FERR25, ERRVECT	:IF THE DST FLOWS FAIL AN ODD :ADDRESS COULD OCCUR.
2388							
2389	006756	005003			CLR	R3	
2390							
2391	006760	174010		F10:	STD	AC0, (R0)	:TEST INSTRUCTION.
2392	006762	005203		F11:	INC	R3	
2393	006764	005203			INC	R3	
2394							
2395	006766	020027	010404		CMP	R0, #FDAT04	:WAS R0 MODIFIED?
2396	006772	001402			BEQ	F12	
2397	006774	000137	007340		JMP	FERR3	
2398							
2399	007000	020327	000002	F12:	CMP	R3, #2	:WAS THE PC AFFECTED CORRECTLY?
2400	007004	001402			BEQ	F135	
2401	007006	000137	007332		JMP	FERR4	
2402							
2403	007012	012701	010374	F135:	MOV	#FDAT00, R1	
2404	007016	012702	010416		MOV	#FXDAT0, R2	
2405							
2406	007022	022122			CMP	(R1)+, (R2)+	:SEE IF THE DATA WAS OUTPUT
2407	007024	001402			BEQ	F13	:TO THE TARGET AREA CORRECTLY.
2408	007026	000137	007436		JMP	FERR5	
2409							
2410	007032	022122		F13:	CMP	(R1)+, (R2)+	
2411	007034	001402			BEQ	F14	
2412	007036	000137	007436		JMP	FERR5	
2413							
2414	007042	022122		F14:	CMP	(R1)+, (R2)+	
2415	007044	001402			BEQ	F15	
2416	007046	000137	007436		JMF	FERR5	
2417							
2418	007052	022122		F15:	CMP	(R1)+, (R2)+	
2419	007054	001402			BEQ	F16	
2420	007056	000137	007436		JMP	FERR5	
2421							
2422	007062	022122		F16:	CMP	(R1)+, (R2)+	
2423	007064	001402			BEQ	F17	
2424	007066	000137	007762		JMP	FERR10	
2425							
2426	007072	022122		F17:	CMP	(R1)+, (R2)+	
2427	007074	001402			BEQ	F20	
2428	007076	000137	007472		JMP	FERR6	
2429							
2430	007102	022122		F20:	CMP	(R1)+, (R2)+	

2431	007104	001402				BEQ	F21		
2432	007106	000137	007626			JMP	FERR7		
2433									
2434	007112	022122			F21:	CMP	(R1)+,(R2)+		
2435	007114	001402				BEQ	F22		
2436	007116	000137	007762			JMP	FERR10		
2437									
2438	007122	005001			F22:	CLR	R1		
2439	007124	170201				STFPS	R1		:MAKE SURE FPS IS CORRECT.
2440	007126	022701	000200			CMP	#200,R1		
2441	007132	001402				BEQ	F23		
2442	007134	000137	010016			JMP	FERR11		
2443	007140	000137	010436		F23:	JMP	FDONE		
2444									
2445	007144				FERR0:				:SOURCE DATA AFFECTED BY
2446	007144	012737	010416	001240		MOV	#FXDAT0,\$TMP3		:THE LDD INSTRUCTION.
2447	007152	012737	010430	001242		MOV	#FXDAT0+12,\$TMP4		
2448	007160	012737	010352	001244		MOV	#FDAT10,\$TMP5		
2449	007166	012737	010364	001246		MOV	#FDAT10+12,\$TMP6		
2450	007174	104025			1\$:	ERROR	+25		
2451	007176	000137	010436			JMP	FDONE		
2452									
2453	007202	012737	010362	001242	FERR1:	MOV	#FDAT14,\$TMP4		:FSRC FLOWS FAILURE.
2454	007210	010037	001240			MOV	R0,\$TMP3		
2455	007214	012737	000762	001244		MOV	#762,\$TMP5		
2456	007222	012737	000321	001250		MOV	#321,\$TMP7		
2457									
2458	007230	022700	010352			CMP	#FDAT10,R0		:FSRC MODE 4?
2459	007234	001004				BNE	1\$		
2460	007236	012737	000324	001246		MOV	#324,\$TMP6		
2461	007244	000412				BR	4\$		
2462									
2463	007246	022700	010372		1\$:	CMP	#FDAT14+10,R0		:FSRC MODE 2?
2464	007252	001004				BNE	2\$		
2465	007254	012737	000322	001246		MOV	#322,\$TMP6		
2466	007262	000403				BR	4\$		
2467									
2468	007264				2\$:				
2469	007264	104027			3\$:	ERROR	+27		
2470	007266	000137	010436			JMP	FDONE		
2471									
2472	007272				4\$:				
2473	007272	104026			5\$:	ERROR	+26		
2474	007274	000137	010436			JMP	FDONE		
2475									
2476	007300	012701	006616		FERR2:	MOV	#F4,R1		:THE PC WAS INCORRECTLY AFFECTED
2477									:DURING THE INSTRUCTION.
2478	007304	010137	001242		FER2:	MOV	R1,\$TMP4		
2479	007310	162701	000004			SUB	#4,R1		
2480	007314	006303				ASL	R3		
2481	007316	060301				ADD	R3,R1		
2482	007320	010137	001240			MOV	R1,\$TMP3		
2483	007324	104030			1\$:	ERROR	+30		
2484	007326	000137	010436			JMP	FDONE		
2485									
2486	007332	012701	006762		FERR4:	MOV	#F11,R1		
2487	007336	000762				BR	FER2		


```

2488
2489 007340 012737 010404 001242 FERR3: MOV #FDAT04,$TMP4 ;FAILURE IN THE FDST FLOWS.
2490 007346 010037 001240 MOV R0,$TMP3
2491 007352 012737 000527 001244 MOV #527,$TMP5
2492 007360 012737 000641 001250 MOV #641,$TMP7
2493
2494 007366 022700 010374 CMP #FDAT00,R0 ;DST MODE 4?
2495 007372 001004 BNE 1$
2496 007374 012737 000644 001246 MOV #644,$TMP6
2497 007402 000412 BR 4$
2498
2499 007404 022700 010414 1$: CMP #FDAT04+10,R0 ;DST MODE 2?
2500 007410 001004 BNE 2$
2501 007412 012737 000642 001246 MOV #642,$TMP6
2502 007420 000403 BR 4$
2503
2504 007422 2$:
2505 007422 104032 3$: ERROR +32
2506 007424 000137 010436 JMP FDONE
2507
2508 007430 4$:
2509 007430 104031 5$: ERROR +31
2510 007432 000137 010436 JMP FDONE
2511
2512 007436 FERR5: ;FAILURE OF STD.
2513 007436 010037 001240 MOV R0,$TMP3
2514 007442 012737 010374 001242 MOV #FDAT00,$TMP4
2515 007450 012737 010412 001244 MOV #FDAT07,$TMP5
2516 007456 012737 010416 001246 MOV #FXDAT0,$TMP6
2517 007464 104033 1$: ERROR +33
2518 007466 000137 010436 JMP FDONE
2519
2520 007472 012701 010406 FERR6: MOV #FDAT05,R1 ;DID (BUT GR7) FAIL IN THE FDST
2521 007476 012702 177777 MOV #-1,R2 ;FLOWS?
2522 007502 012703 000003 MOV #3,R3
2523 007506 020221 1$: CMP R2,(R1)+
2524 007510 001017 BNE 5$
2525 007512 077303 SOB R3,1$
2526
2527 ;REPORT FAILURE OF (BUT GR7) IN
2528 007514 010037 001240 MOV R0,$TMP3 ;THE FDST FLOWS.
2529 007520 012737 000412 001244 MOV #412,$TMP5
2530 007526 012737 000147 001246 MOV #147,$TMP6
2531 007534 012737 000145 001250 MOV #145,$TMP7
2532 007542 104034 2$: ERROR +34
2533 007544 000137 010436 JMP FDONE
2534
2535 007550 012701 010406 5$: MOV #FDAT05,R1 ;DID (BUT GR7) FAIL IN THE SRC FLOWS?
2536 007554 012703 000003 MOV #3,R3
2537 007560 005721 6$: TST (R1)+
2538 007562 001402 BEQ 7$
2539 007564 000137 007762 JMP FERR10
2540 007570 077305 7$: SOB R3,6$
2541
2542 ;REPORT FAILURE OF (BUT GR7) IN
2543 007572 010037 001240 MOV R0,$TMP3 ;THE FSRC FLOWS.
2544 007576 012737 000207 001244 MOV #207,$TMP5
    
```

2545	007604	012737	000176	001246		MOV	#176,\$TMP6	
2546	007612	012737	000174	001250		MOV	#174,\$TMP7	
2547								
2548	007620	104035			10\$:	ERROR	+35	
2549	007622	000137	010436			JMP	FDONE	
2550								
2551	007626	012701	010410		FERR7:	MOV	#FDAT06,R1	;DID (BUT FD) FAIL IN THE FDST FLOWS?
2552	007632	012702	177777			MOV	#-1,R2	
2553	007636	012703	000002			MOV	#2,R3	
2554	007642	020221			1\$:	CMP	R2,(R1)+	
2555	007644	001017				BNE	5\$	
2556	007646	077303				SOB	R3,1\$	
2557								
2558								;REPORT FAILURE OF (BUT FD) IN THE
2559	007650	010037	001240			MOV	R0,\$TMP3	;FDST FLOWS.
2560	007654	012737	000707	001244		MOV	#707,\$TMP5	
2561	007662	012737	000244	001246		MOV	#244,\$TMP6	
2562	007670	012737	000245	001250		MOV	#245,\$TMP7	
2563	007676	104036			2\$:	ERROR	+36	
2564	007700	000137	010436			JMP	FDONE	
2565								
2566	007704	012701	010410		5\$:	MOV	#FDAT06,R1	;DID (BUT FD) FAIL IN THE FSRC FLOWS?
2567	007710	012703	000002			MOV	#2,R3	
2568	007714	005721			6\$:	TST	(R1)+	
2569	007716	001402				BEQ	7\$	
2570	007720	000137	007762			JMP	FERR10	
2571	007724	077305			7\$:	SOB	R3,6\$	
2572								
2573								;REPORT FAILURE OF (BUT FD) IN THE
2574	007726	010037	001240			MOV	R0,\$TMP3	;FSRC FLOWS.
2575	007732	012737	000441	001244		MOV	#441,\$TMP5	
2576	007740	012737	000076	001246		MOV	#76,\$TMP6	
2577	007746	012737	000077	001250		MOV	#77,\$TMP7	
2578	007754	104037			10\$:	ERROR	+37	
2579	007756	000137	010436			JMP	FDONE	
2580								
2581	007762				FERR10:			;REPORT DATA ERROR.
2582	007762	010037	001240			MOV	R0,\$TMP3	
2583	007766	012737	010404	001242		MOV	#FDAT04,\$TMP4	
2584	007774	012737	010412	001244		MOV	#FDAT07,\$TMP5	
2585	010002	012737	010426	001246		MOV	#FXDAT4,\$TMP6	
2586	010010	104040			1\$:	ERROR	+40	
2587	010012	000137	010436			JMF	FDONE	
2588								
2589	010016				FERR11:			;REPORT BAD FPS.
2590	010016	010137	001240			MOV	R1,\$TMP3	
2591	010022	012737	000200	001242		MOV	#200,\$TMP4	
2592	010030	104041			1\$:	ERROR	+41	
2593	010032	000137	010436			JMP	FDONE	
2594								
2595	010036	012737	042737	001264	FERR20:	MOV	#NULL,\$TMP15	;THE EXECUTION OF THE LDD
2596	010044	005037	001252			CLR	\$TMP10	;CAUSED A TRAP TO 4, BECAUSE
2597	010050	011637	001236			MOV	(SP),\$TMP2	;A FSRC FLOW FAILURE RESULTED
2598	010054	012737	010362	001240		MOV	#FDAT14,\$TMP3	;IN AN ODD ADDRESS.
2599	010062	012737	000321	001250		MOV	#321,\$TMP7	
2600	010070	012737	000762	001244		MOV	#762,\$TMP5	
2601								


```
2659 010320 000446 BR FDONE
2660
2661 010322 FERR26: ;REPORT FDST FLOW FAILURE TO MODE
2662 010322 012737 044503 001264 MOV #MS16,$TMP15 ;6 OR MODE 7.
2663 010330 012737 000646 001246 MOV #646,$TMP6
2664 010336 012737 000647 001252 MOV #647,$TMP10
2665 010344 022626 CMP (SP)+,(SP)+
2666 010346 104043 1$: ERROR +43
2667 010350 000432 BR FDONE
2668
2669 010352 177777 FDATA0: -1
2670 010354 177777 FDATA1: -1
2671 010356 177777 FDATA2: -1
2672 010360 177777 FDATA3: -1
2673 010362 177777 FDATA4: -1
2674 010364 177777 FDATA5: -1
2675 010366 177777 FDATA6: -1
2676 010370 177777 FDATA7: -1
2677 010372 177777 -1
2678 010374 177777 FDATA0: -1
2679 010376 177777 FDATA1: -1
2680 010400 177777 FDATA2: -1
2681 010402 177777 FDATA3: -1
2682 010404 177777 FDATA4: -1
2683 010406 177777 FDATA5: -1
2684 010410 177777 FDATA6: -1
2685 010412 177777 FDATA7: -1
2686 010414 177777 -1
2687 010416 177777 FXDATA0: -1
2688 010420 177777 FXDATA1: -1
2689 010422 177777 FXDATA2: -1
2690 010424 177777 FXDATA3: -1
2691 010426 052525 FXDATA4: 052525
2692 010430 031463 FXDATA5: 031463
2693 010432 007417 FXDATA6: 007417
2694 010434 000477 FXDATA7: 000477
2695
2696
2697 010436 FDONE: ;GO INITIALIZE THE FPS AND STACK; AND
010436 104413 RSETUP ;SEE IF THE USER HAS EXPRESSED
;THE DESIRE TO CHANGE THE SOFTWARE
;VIRTUAL CONSOLE SWITCH REGISTER (HAS
;THE USER TYPED CONTROL G?).
2698
2699
```


2705

```

.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.
*
*****
  
```

2706	010440	000004			TST7:	SCOPE		
2707	010442	012737	010450	001110		MOV	#I1,\$LPERR	;SET UP THE LOOP ON ERROR ADDRESS.
2708	010450				I1:			
2709	010450	170011				SETD		;SET FD.
2710	010452	012700	011300			MOV	#IDATIO,R0	
2711	010456	012701	011250			MOV	#IPATIO,R1	
2712	010462	012702	000004			MOV	#4,R2	
2713	010466	012120			I2:	MOV	(R1)+,(R0)+	;SET UP THE INPUT DATA BUFFER.
2714	010470	077202				SOB	R2,I2	
2715								
2716	010472	012700	011300			MOV	#IDATIO,R0	;LOAD AC1
2717	010476	172510				LDD	(R0),AC1	
2718								
2719	010500	012700	011260			MOV	#IPAT20,R0	;LOAD ACO
2720	010504	172410				LDD	(R0),ACO	
2721								
2722	010506	012701	000001			MOV	#1,R1	;IN CASE THE FSRC FLOWS FAIL
2723	010512	012737	011050	000004		MOV	#IERR0,ERRVECT	;AN ODD ADDRESS TRAP TO 4 MAY OCCUR.
2724	010520	012737	010534	001236		MOV	#I3,\$TMP2	
2725	010526	012737	045163	001240		MOV	#MS35,\$TMP3	
2726	010534	172401			I3:	LDD	AC1,ACO	;TEST INSTRUCTION.
2727	010536	000240			I4:	NOP		
2728	010540	000240			I5:	NOP		
2729								
2730	010542	012700	011270			MOV	#IDAT00,R0	
2731	010546	174010				STD	ACO,(R0)	;GET ACO, THE RESULTS.
2732								
2733	010550	012700	011270			MOV	#IDAT00,R0	;SEE IF DATA IS CORRECT.
2734	010554	012701	011300			MOV	#IDATIO,R1	
2735	010560	012702	000004			MOV	#4,R2	
2736	010564	022021			I6:	CMP	(R0)+,(R1)+	
2737	010566	001424				BEQ	I105	
2738								
2739	010570	012700	011274			MOV	#IDAT02,R0	;SEE IF (BUT FD) FAILED.
2740	010574	012702	000002			MOV	#2,R2	
2741	010600	005720			I7:	TST	(R0)+	
2742	010602	001413				BEQ	I10	
2743								
2744	010604	012700	011274			MOV	#IDAT02,R0	
2745	010610	012702	000002			MOV	#2,R2	
2746	010614	022720	177777		I\$:	CMP	#-1,(R0)+	
2747	010620	001402				BEQ	2\$	
2748	010622	000137	011132			JMP	IERR1	
2749	010626	077206			2\$:	SOB	R2,I\$	
2750	010630	000401				BR	I106	
2751	010632	077216			I10:	SOB	R2,I7	
2752	010634	000137	011152		I106:	JMP	IERR2	
2753								
2754	010640	077227			I105:	SOB	R2,I6	

```

2755
2756 ;NOW TEST THE LOAD INSTRUCTION WITH FSRC MODE ZERO AND FD CLEAR.
2757
2758 010642 012737 010650 001110 I11: MOV #I12,$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
      010642 012700 011250 I12: MOV #IPAT10,R0
2759 010650 012700 011250 I12: MOV #IDATIO,R1
2760 010654 012701 011300 I12: MOV #4,R2
2761 010660 012702 000004 I13: MOV (R0)+,(R1)+
2762 010664 012021 I13: SOB R2,I13
2763 010666 077202
2764
2765 010670 012700 011300 MOV #IDATIO,R0 ;SET UP AC1
2766 010674 172510 LDD (R0),AC1
2767
2768 010676 012700 011260 MOV #IPAT20,R0 ;SET UP AC0
2769 010702 172410 LDD (R0),AC0
2770
2771 010704 012701 000001 MOV #1,R1
2772 010710 012737 010726 001236 MOV #I14,$TMP2
2773 010716 012737 045170 001240 MOV #MS36,$TMP3
2774 010724 170001 SETF ;CLEAR FD.
2775
2776 010726 172401 I14: LDF AC1,AC0 ;TEST INSTRUCTION.
2777 010730 000240 I15: NOP
2778 010732 000240 I16: NOP
2779
2780 010734 170200 STFPS R0 ;SEE IF FPS IS STILL CLEAR.
2781 010736 022700 000004 CMP #4,R0
2782 010742 001402 BEQ I17
2783 010744 000137 011224 JMP IERR3
2784
2785 010750 I17: ;RESET TO DOUBLE MODE.
2786 010750 170011 SETD
2787
2788 010752 012700 011270 MOV #IDATIO,R0
2789 010756 174010 STD AC0,(R0) ;GET AC0
2790
2791 010760 012737 177777 011304 MOV #-1,IDATIO
2792 010766 012737 177777 011306 MOV #-1,IDATIO
2793 010774 012700 011270 MOV #IDATIO,R0
2794 011000 012701 011300 MOV #IDATIO,R1
2795 011004 012702 000004 MOV #4,R2
2796 011010 022021 I20: CMF (R0)+,(R1)+ ;SEE IF AC0 WAS CORRECT.
2797 011012 001414 BEQ I23
2798
2799 011014 023737 011274 011254 CMP IDATIO,IPAT12 ;DID (BUT FD) FAIL?
2800 011022 001402 BEQ I22
2801 011024 000137 011132 I21: JMP IERR1
2802 011030 023737 011276 011256 I22: CMP IDATIO,IPAT13
2803 011036 001372 BNE I21
2804 011040 000137 011200 JMP IERR4
2805
2806 011044 077217 I23: SOB R2,I20
2807
2808 011046 000520 BR IDONE ;NO ERRORS.
2809
2810 ;IF AN ODD ADDRESS TRAP OCCURS COME HERE TO ANALYZE THE FSRC FAILURE.
  
```


2811	011050	022716	010536			IERR0:	CMP	#14,(SP)		
2812	011054	001413					BEQ	1\$:MAKE SURE THE TRAP OCCURRED
2813	011056	022716	010540				CMP	#15,(SP)		:ON THE INSTRUCTION BEING TESTED.
2814	011062	001410					BEQ	1\$		
2815	011064	022716	010730				CMP	#115,(SP)		
2816	011070	001405					BEQ	1\$		
2817	011072	022716	010732				CMP	#116,(SP)		
2818	011076	001402					BEQ	1\$		
2819	011100	000137	042566				JMP	CPSPUR		
2820										
2821	011104	011637	001236			1\$:	MOV	(SP), \$TMP2		:REPORT FAILURE.
2822	011110	012737	000627	001240			MOV	#627, \$TMP3		
2823	011116	012737	000320	001242			MOV	#320, \$TMP4		
2824	011124	022626					CMP	(SP)+, (SP)+		
2825	011126	104047				2\$:	ERROR	+47		
2826	011130	000467					BR	IDONE		
2827										
2828										
2829	011132									:REPORT DATA ERROR.
2830	011132	012737	011300	001242		IERR1:	MOV	#IDATIO, \$TMP4		
2831	011140	012737	011270	001244			MOV	#IDAT00, \$TMP5		
2832	011146	104051				1\$:	ERROR	+51		
2833	011150	000457					BR	IDONE		
2834										
2835										:REPORT FAILURE OF (BUT FD)
2836	011152	012737	000153	001244		IERR2:	MOV	#153, \$TMP5		
2837	011160	012737	000434	001246			MOV	#434, \$TMP6		
2838	011166	012737	000435	001250			MOV	#435, \$TMP7		
2839	011174					IERR25:				
2840	011174	104050				1\$:	ERROR	+50		
2841	011176	000444					BR	IDONE		
2842	011200	012737	000153	001244		IERR4:	MOV	#153, \$TMP5		
2843	011206	012737	000435	001246			MOV	#435, \$TMP6		
2844	011214	012737	000434	001250			MOV	#434, \$TMP7		
2845	011222	000764					BR	IERR25		
2846										
2847										:REPORT INCORRECT FPS AFTER LOAD INSTRUCTION.
2848	011224					IERR3:				
2849	011224	012737	010726	001236			MOV	#114, \$TMP2		
2850	011232	010037	001240				MOV	R0, \$TMP3		
2851	011236	012737	000004	001242			MOV	#4, \$TMP4		
2852	011244	104041				1\$:	ERROR	+41		
2853	011246	000420					BR	IDONE		
2854										
2855										
2856	011250	000000				IPAT10:	0			
2857	011252	170360				IPAT11:	170360			
2858	011254	016161				IPAT12:	016161			
2859	011256	052525				IPAT13:	052525			
2860										
2861	011260	177777				IPAT20:	-1			
2862	011262	177777				IPAT21:	-1			
2863	011264	177777				IPAT22:	-1			
2864	011266	177777				IPAT23:	-1			
2865										
2866	011270	000000				IDAT00:	0			
2867	011272	000000				IDAT01:	0			

2868	011274	000000	IDATO2:	0
2869	011276	000000	IDATO3:	0
2870				
2871	011300	000000	IDATI0:	0
2872	011302	000000	IDATI1:	0
2873	011304	000000	IDATI2:	0
2874	011306	000000	IDATI3:	0
2875				
2876	011310		IDONE:	
	011310	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?).

2877
2883

2884

```

2885 011312 000004
2885 011314
2886 011314 012737 011322 001110
2887 011322 170011
2888 011324 012700 012066
2889 011330 012701 012116
2890 011334 012702 000004
2891 011340 012021
2892 011342 077202
2893 011344 012700 012116
2894 011350 172410
2895
2896 011352 012700 012076
2897 011356 172510
2898
2899 011360 012701 000001
2900 011364 012737 011674 000004
2901 011372 012737 011406 001236
2902 011400 012737 045163 001240
2903 011406 174001
2904 011410 000240
2905 011412 000240
2906
2907 011414 012700 012106
2908 011420 174110
2909
2910 011422 012703 012106
2911 011426 012704 012116
2912 011432 012705 000004
2913 011436 022324
2914 011440 001413
2915
2916 011442 012703 012112
2917 011446 012705 000002
2918 011452 005723
2919 011454 001402
2920 011456 000137 011756
2921 011462 077505
2922 011464 000137 011776
2923
2924 011470 077516
2925
2926
2927
2928 011472
2928 011472 012737 011500 001110
2929
2930 011500 012700 012066
2931 011504 012701 012116
    
```

```

.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:
T1:  MOV    #1$, $LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
      SETD
      ;SET FD
      MOV    #TPAT10,R0
      MOV    #TDAT10,R1
      MOV    #4,R2
T2:  MOV    (R0)+,(R1)+      ;SET UP THE INPUT DATA BUFFER.
      SOB   R2,T2
      MOV    #TDAT10,R0      ;LOAD AC0
      LDD   (R0),AC0
      MOV    #TPAT20,R0      ;LOAD AC1
      LDD   (R0),AC1
      MOV    #1,R1           ;IF THE (BUT FDST) FORK FAILS
      MOV    #TERRO,ERRVECT ;AN ODD ADDRESS TRAP COULD RESULT.
      MOV    #T3,$TMP2
      MOV    #MS35,$TMP3
T3:  STD   AC0,AC1
T4:  NOP
T5:  NOP
      MOV    #TDAT00,R0
      STD   AC1,(R0)        ;GET THE DATA.
      MOV    #TDAT00,R3      ;SEE IF THE DATA IS CORRECT.
      MOV    #TDAT10,R4
      MOV    #4,R5
T6:  CMP   (R3)+,(R4)+
      BEQ   T105
      MOV    #TDAT02,R3      ;DID (BUT FD) FAIL?
      MOV    #2,R5
T7:  TST   (R3)+
      BEQ   T10
      JMP   TERR1
T10: SOB   R5,T7
      JMP   TERR2
T105: SOB   R5,T6
;NOW TEST THE STF AC0,AC1 INSTRUCTION.
T11:
T11:  MOV    #T12,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
      MOV    #TPAT10,R0
      MOV    #TDAT10,R1      ;SET UP THE INPUT DATA BUFFER.
    
```

```

2932 011510 012702 000004          MOV    #4,R2
2933 011514 012021          T13:  MOV    (R0)+,(R1)+
2934 011516 077202          SOB    R2,T13
2935
2936 011520 012700 012116          MOV    #TDAT10,R0          ;SET UP AC0
2937 011524 172410          LDD    (R0),AC0
2938
2939 011526 012700 012076          MOV    #TPAT20,R0          ;SET UP AC1
2940 011532 172510          LDD    (R0),AC1
2941
2942 011534 012701 000001          MOV    #1,R1
2943 011540 012737 011556 001236          MOV    #T14,$TMP2
2944 011546 012737 045170 001240          MOV    #MS36,$TMP3
2945 011554 170001          SETF
2946 011556 174001          T14:  STF    AC0,AC1          ;CLEAR FD
2947 011560 000240          T15:  NOP
2948 011562 000240          T16:  NOP
2949
2950 011564 005000          CLR    R0
2951 011566 170200          STFPS R0          ;SEE IF FPS IS CLEAR.
2952 011570 022700 000010          CMP    #10,R0
2953 011574 001401          BEQ    T17
2954 011576 000521          BR     TERR3
2955
2956 011600          T17:  SETD
2957 011600 170011          ;SET FD.
2958
2959 011602 012700 012106          MOV    #TDAT00,R0
2960 011606 174110          STD    AC1,(R0)          ;PICK UP AC1.
2961
2962 011610 012737 177777 012122          MOV    #-1,TDAT12
2963 011616 012737 177777 012124          MOV    #-1,TDAT13
2964 011624 012703 012106          MOV    #TDAT00,R3
2965 011630 012704 012116          MOV    #TDAT10,R4
2966 011634 012705 000004          MOV    #4,R5
2967 011640 022324          T20:  CMP    (R3)+,(R4)+          ;WAS THE DATA TRANSFERRED CORRECTLY?
2968 011642 001412          BEQ    T23
2969
2970 011644 023737 012112 012072          CMP    TDAT02,TPAT12          ;DID (BUT FD) FAIL.
2971 011652 001401          BEQ    T22
2972 011654 000440          BR     TERR1
2973 011656 023737 012114 012074          T21:  CMP    TDAT03,TPAT13
2974 011664 001373          T22:  BNE   T21
2975 011666 000456          BR     TERR4
2976
2977 011670 077515          T23:  SOB   R5,T20
2978 011672 000515          BR     TDONE
2979
2980
2981          ;TRAP HERE THROUGH VECTOR 4 IF AN ODD ADDRESS OCCURS.
2982 011674 022716 011410          TERRO: CMP    #T4,(SP)          ;MAKE SURE THE TRAP WAS ON
2983 011700 001413          BEQ    1$          ;AN INSTRUCTION BEING TESTED.
2984 011702 022716 011412          CMP    #T5,(SP)
2985 011706 001410          BEQ    1$
2986 011710 022716 011560          CMP    #T15,(SP)
2987 011714 001405          BEQ    1$
2988 011716 022716 011562          CMP    #T16,(SP)
  
```



```

2989 011722 001402          BEQ      1$
2990 011724 000137 042566    JMP      CPSPUR
2991
2992 011730 011637 001236    1$:     MOV      (SP), $TMP2
2993 011734 022626          CMP      (SP)+, (SP)+
2994 011736 012737 000527 001240    MOV      #527, $TMP3
2995 011744 012737 000640 001242    MOV      #640, $TMP4
2996 011752 104121          2$:     ERROR   +121
2997 011754 000464          BR       TDONE
2998
2999          :REPORT DATA FAILURE.
3000 011756          TERR1:
3001 011756 012737 012116 001242    MOV      #TDAT10, $TMP4
3002 011764 012737 012106 001244    MOV      #TDAT00, $TMP5
3003 011772 104123          1$:     ERROR   +123
3004 011774 000454          BR       TDONE
3005
3006          :REPORT FAILURE OF (BUT FD).
3007 011776 012737 000160 001246    TERR2:  MOV      #160, $TMP6
3008 012004 012737 000161 001250    MOV      #161, $TMP7
3009 012012 012737 000640 001244    TERR25: MOV      #640, $TMP5
3010 012020 104122          1$:     ERROR   +122
3011 012022 000441          BR       TDONE
3012 012024 012737 000161 001246    TERR4:  MOV      #161, $TMP6
3013 012032 012737 000160 001250    MOV      #160, $TMP7
3014 012040 000764          BR       TERR25
3015
3016          :REPORT INCORRECT FPS AFTER STORE INSTRUCTION.
3017 012042          TERR3:
3018 012042 012737 011560 001236    MOV      #T15, $TMP2
3019 012050 010037 001240    MOV      R0, $TMP3
3020 012054 012737 000010 001242    MOV      #10, $TMP4
3021 012062 104041          1$:     ERROR   +41
3022 012064 000420          BR       TDONE
3023
3024 012066 000000          TPAT10: 0
3025 012070 170360          TPAT11: 170360
3026 012072 016161          TPAT12: 016161
3027 012074 052525          TPAT13: 052525
3028
3029 012076 177777          TPAT20: -1
3030 012100 177777          TPAT21: -1
3031 012102 177777          TPAT22: -1
3032 012104 177777          TPAT23: -1
3033
3034 012106 000000          TDAT00: 0
3035 012110 000000          TDAT01: 0
3036 012112 000000          TDAT02: 0
3037 012114 000000          TDAT03: 0
3038
3039 012116 000000          TDAT10: 0
3040 012120 000000          TDAT11: 0
3041 012122 000000          TDAT12: 0
3042 012124 000000          TDAT13: 0
3043
3044 012126          TDONE:
      012126 104413          RSETUP          :GO INITIALIZE THE FPS AND STACK; AND
  
```

3045
3046
3179

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

3180

```
.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. THUS 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


```

012254 077330          SOB      R3,G1
012256 004737 014500  ;TEST ACCUMULATOR 0 WITH FLOATING ZERO ;TYPE ERROR SUMMARY.
3183 012262 012737 044535 001244  MOV      #MNUM0,$TMP5
012270 012737 012324 001236  MOV      #G4,$TMP2
012276 012700 015146          MOV      #GPAT10,R0
012302 012701 015176          MOV      #GDAT00,R1
012306 012737 012324 001110  MOV      #G4,$LPERR          ;SET UP THE LOOP ON ERROR ADDRESS.
012314 004737 014270          JSR      PC,GSETUP          ;LOAD TEST PATTERN.
012320 012703 000102          MOV      #102,R3
012324 172410          G4:  LDD      (R0),AC0
012326 174000          STD      AC0,AC0
012330 172400          LDD      AC0,AC0          ;STORE THE TEST PATTERN.
012332 174011          STD      AC0,(R1)
012334 004737 014366          JSR      PC,GCMP          ;COMPARE THE DATA READ WITH
                                ;THAT WHICH WAS WRITTEN.

012340 005737 015132          TST      GFLAG1
012344 001004          BNE      G5
012346 005137 015132          COM      GFLAG1
012352 000241          CLC
012354 000401          BR      G5
012356 000261          G5:  SEC
012360 006160 000006          G6:  ROL      6(R0)          ;GENERATE THE NEXT TEST PATTERN.
012364 006160 000004          ROL      4(R0)
012370 006160 000002          ROL      2(R0)
012374 006110          ROL      (R0)
012376 004737 014346          JSR      PC,GRESET          ;RESET DEFAULT PATTERN IN OUTPUT
                                ;BUFFER.

012402 077330          SOB      R3,G4
012404 004737 014500  ;TEST ACCUMULATOR 1 WITH FLOATING ONE ;TYPE ERROR SUMMARY.
3184 012410 012737 044543 001244  MOV      #MNUM1,$TMP5
012416 012737 012452 001236  MOV      #G7,$TMP2
012424 012700 015136          MOV      #GPAT00,R0
012430 012701 015176          MOV      #GDAT00,R1
012434 012737 012452 001110  MOV      #G7,$LPERR          ;SET UP THE LOOP ON ERROR ADDRESS.
012442 004737 014270          JSR      PC,GSETUP          ;LOAD TEST PATTERN.
012446 012703 000102          MOV      #102,R3
012452 172410          G7:  LDD      (R0),AC0
012454 174001          STD      AC0,AC1
012456 172401          LDD      AC1,AC0          ;STORE THE TEST PATTERN.
012460 174011          STD      AC0,(R1)
012462 004737 014366          JSR      PC,GCMP          ;COMPARE THE DATA READ WITH
                                ;THAT WHICH WAS WRITTEN.

012466 005737 015132          TST      GFLAG1
012472 001004          BNE      G10
012474 005137 015132          COM      GFLAG1
012500 000261          SEC
012502 000401          BR      G10
012504 000241          G10: CLC
012506 006160 000006          G11: ROL      6(R0)          ;GENERATE THE NEXT TEST PATTERN.
012512 006160 000004          ROL      4(R0)
012516 006160 000002          ROL      2(R0)
012522 006110          ROL      (R0)
012524 004737 014346          JSR      PC,GRESET          ;RESET DEFAULT PATTERN IN OUTPUT
                                ;BUFFER.

012530 077330          SOB      R3,G7

```



```

3185 012532 004737 014500          JSR      PC,GSUM          ;TYPE ERROR SUMMARY.
;TEST ACCUMULATOR 1 WITH FLOATING ZERO
012536 012737 044543 001244      MOV      #MNUM1,$TMP5
012544 012737 012600 001236      MOV      #G12,$TMP2
012552 012700 015146                MOV      #GPAT10,R0
012556 012701 015176                MOV      #GDAT00,R1
012562 012737 012600 001110      MOV      #G12,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
012570 004737 014270                JSR      PC,GSETUP        ;LOAD TEST PATTERN.
012574 012703 000102                MOV      #102,R3
012600 172410          G12:      LDD      (R0),AC0
012602 174001          STD      AC0,AC1
012604 172401          LDD      AC1,AC0          ;STORE THE TEST PATTERN.
012606 174011          STD      AC0,(R1)
012610 004737 014366          JSR      PC,GCMP          ;COMPARE THE DATA READ WITH
;THAT WHICH WAS WRITTEN.

012614 005737 015132          TST      GFLAG1
012620 001004          BNE      G13
012622 005137 015132          COM      GFLAG1
012626 000241          CLC
012630 000401          BR
012632 000261          G13:      SEC          G14
012634 006160 000006          G14:      ROL      6(R0)      ;GENERATE THE NEXT TEST PATTERN.
012640 006160 000004          ROL      4(R0)
012644 006160 000002          ROL      2(R0)
012650 006110          ROL      (R0)
012652 004737 014346          JSR      PC,GRESET      ;RESET DEFAULT PATTERN IN OUTPUT
;BUFFER.

012656 077330          SOB      R3,G12
012660 004737 014500          JSR      PC,GSUM          ;TYPE ERROR SUMMARY.
3186 ;TEST ACCUMULATOR 2 WITH FLOATING ONE
012664 012737 044550 001244      MOV      #MNUM2,$TMP5
012672 012737 012726 001236      MOV      #G15,$TMP2
012700 012700 015136                MOV      #GPAT00,R0
012704 012701 015176                MOV      #GDAT00,R1
012710 012737 012726 001110      MOV      #G15,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
012716 004737 014270                JSR      PC,GSETUP        ;LOAD TEST PATTERN.
012722 012703 000102                MOV      #102,R3
012726 172410          G15:      LDD      (R0),AC0
012730 174002          STD      AC0,AC2
012732 172402          LDD      AC2,AC0          ;STORE THE TEST PATTERN.
012734 174011          STD      AC0,(R1)
012736 004737 014366          JSR      PC,GCMP          ;COMPARE THE DATA READ WITH
;THAT WHICH WAS WRITTEN.

012742 005737 015132          TST      GFLAG1
012746 001004          BNE      G16
012750 005137 015132          COM      GFLAG1
012754 000261          SEC
012756 000401          BR
012760 000241          G16:      CLC          G17
012762 006160 000006          G17:      ROL      6(R0)      ;GENERATE THE NEXT TEST PATTERN.
012766 006160 000004          ROL      4(R0)
012772 006160 000002          ROL      2(R0)
012776 006110          ROL      (R0)
013000 004737 014346          JSR      PC,GRESET      ;RESET DEFAULT PATTERN IN OUTPUT
;BUFFER.

013004 077330          SOB      R3,G15
013006 004737 014500          JSR      PC,GSUM          ;TYPE ERROR SUMMARY.
    
```

3187

```

;TEST ACCUMULATOR 2 WITH FLOATING ZERO
013012 012737 044550 001244 MOV #MNUM2,$TMP5
013020 012737 013054 001236 MOV #G20,$TMP2
013026 012700 015146 MOV #GPAT10,R0
013032 012701 015176 MOV #GDAT00,R1
013036 012737 013054 001110 MOV #G20,$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
013044 004737 014270 JSR PC,GSETUP ;LOAD TEST PATTERN.
013050 012703 000102 MOV #102,R3
013054 172410 G20: LDD (R0),AC0
013056 174002 STD AC0,AC2
013060 172402 LDD AC2,AC0 ;STORE THE TEST PATTERN.
013062 174011 STD AC0,(R1)
013064 004737 014366 JSR PC,GCMP ;COMPARE THE DATA READ WITH
;THAT WHICH WAS WRITTEN.

013070 005737 015132 TST GFLAG1
013074 001004 BNE G21
013076 005137 015132 COM GFLAG1
013102 000241 CLC
013104 000401 BR G22
013106 000261 G21: SEC
013110 006160 000006 G22: ROL 6(R0) ;GENERATE THE NEXT TEST PATTERN.
013114 006160 000004 ROL 4(R0)
013120 006160 000002 ROL 2(R0)
013124 006110 ROL (R0)
013126 004737 014346 JSR PC,GRESET ;RESET DEFAULT PATTERN IN OUTPUT
;BUFFER.

013132 077330 SOB R3,G20
013134 004737 014500 JSR PC,GSUM ;TYPE ERROR SUMMARY.

```

3188

```

;TEST ACCUMULATOR 3 WITH FLOATING ONE
013140 012737 044555 001244 MOV #MNUM3,$TMP5
013146 012737 013202 001236 MOV #G23,$TMP2
013154 012700 015136 MOV #GPAT00,R0
013160 012701 015176 MOV #GDAT00,R1
013164 012737 013202 001110 MOV #G23,$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
013172 004737 014270 JSR PC,GSETUP ;LOAD TEST PATTERN.
013176 012703 000102 MOV #102,R3
013202 172410 G23: LDD (R0),AC0
013204 174003 STD AC0,AC3
013206 172403 LDD AC3,AC0 ;STORE THE TEST PATTERN.
013210 174011 STD AC0,(R1)
013212 004737 014366 JSR PC,GCMP ;COMPARE THE DATA READ WITH
;THAT WHICH WAS WRITTEN.

013216 005737 015132 TST GFLAG1
013222 001004 BNE G24
013224 005137 015132 COM GFLAG1
013230 000261 SEC
013232 000401 BR G25
013234 000241 G24: CLC
013236 006160 000006 G25: ROL 6(R0) ;GENERATE THE NEXT TEST PATTERN.
013242 006160 000004 ROL 4(R0)
013246 006160 000002 ROL 2(R0)
013252 006110 ROL (R0)
013254 004737 014346 JSR PC,GRESET ;RESET DEFAULT PATTERN IN OUTPUT
;BUFFER.

013260 077330 SOB R3,G23
013262 004737 014500 JSR PC,GSUM ;TYPE ERROR SUMMARY.

```

3189

;TEST ACCUMULATOR 3 WITH FLOATING ZERO


```

013266 012737 044555 001244      MOV      #MNUM3,$TMP5
013274 012737 013330 001236      MOV      #G26,$TMP2
013302 012700 015146                MOV      #GPAT10,R0
013306 012701 015176                MOV      #GDAT00,R1
013312 012737 013330 001110      MOV      #G26,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
013320 004737 014270                JSR      PC,GSETUP        ;LOAD TEST PATTERN.
013324 012703 000102                MOV      #102,R3
013330 172410                G26:    LDD      (R0),AC0
013332 174003                STD      AC0,AC3
013334 172403                LDD      AC3,AC0          ;STORE THE TEST PATTERN.
013336 174011                STD      AC0,(R1)
013340 004737 014366                JSR      PC,GCMP          ;COMPARE THE DATA READ WITH
                                ;THAT WHICH WAS WRITTEN.

013344 005737 015132                TST      GFLAG1
013350 001004                BNE      G27
013352 005137 015132                COM      GFLAG1
013356 000241                CLC
013360 000401                BR       G30
013362 000261                G27:    SEC
013364 006160 000006                G30:    ROL      6(R0)      ;GENERATE THE NEXT TEST PATTERN.
013370 006160 000004                ROL      4(R0)
013374 006160 000002                ROL      2(R0)
013400 006110                ROL      (R0)
013402 004737 014346                JSR      PC,GRESET        ;RESET DEFAULT PATTERN IN OUTPUT
                                ;BUFFER.

013406 077330                SOB      R3,G26
013410 004737 014500                JSR      PC,GSUM          ;TYPE ERROR SUMMARY.
3190 ;TEST ACCUMULATOR 4 WITH FLOATING ONE
013414 012737 044564 001244      MOV      #MNUM4,$TMP5
013422 012737 013456 001236      MOV      #G31,$TMP2
013430 012700 015136                MOV      #GPAT00,R0
013434 012701 015176                MOV      #GDAT00,R1
013440 012737 013456 001110      MOV      #G31,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
013446 004737 014270                JSR      PC,GSETUP        ;LOAD TEST PATTERN.
013452 012703 000102                MOV      #102,R3
013456 172410                G31:    LDD      (R0),AC0
013460 174004                STD      AC0,AC4
013462 172404                LDD      AC4,AC0          ;STORE THE TEST PATTERN.
013464 174011                STD      AC0,(R1)
013466 004737 014366                JSR      PC,GCMP          ;COMPARE THE DATA READ WITH
                                ;THAT WHICH WAS WRITTEN.

013472 005737 015132                TST      GFLAG1
013476 001004                BNE      G32
013500 005137 015132                COM      GFLAG1
013504 000261                SEC
013506 000401                BR       G33
013510 000241                G32:    CLC
013512 006160 000006                G33:    ROL      6(R0)      ;GENERATE THE NEXT TEST PATTERN.
013516 006160 000004                ROL      4(R0)
013522 006160 000002                ROL      2(R0)
013526 006110                ROL      (R0)
013530 004737 014346                JSR      PC,GRESET        ;RESET DEFAULT PATTERN IN OUTPUT
                                ;BUFFER.

013534 077330                SOB      R3,G31
013536 004737 014500                JSR      PC,GSUM          ;TYPE ERROR SUMMARY.
3191 ;TEST ACCUMULATOR 4 WITH FLOATING ZERO
013542 012737 044564 001244      MOV      #MNUM4,$TMP5
    
```

```

013550 012737 013604 001236      MOV      #G34,$TMP2
013556 012700 015146      MOV      #GPAT10,R0
013562 012701 015176      MOV      #GDAT00,R1
013566 012737 013604  G01110     MOV      #G34,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
013574 004737 014270      JSR      PC,GSETUP        ;LOAD TEST PATTERN.
013600 012703 000102      MOV      #102,R3
013604 172410      G34:    LDD      (R0),AC0
013606 174004      STD      AC0,AC4
013610 172404      LDD      AC4,AC0          ;STORE THE TEST PATTERN.
013612 174011      STD      AC0,(R1)
013614 004737 014366      JSR      PC,GCMP          ;COMPARE THE DATA READ WITH
                                ;THAT WHICH WAS WRITTEN.

013620 005737 015132      TST      GFLAG1
013624 001004      BNE      G35
013626 005137 015132      COM      GFLAG1
013632 000241      CLC
013634 000401      BR       G36
013636 000261      G35:    SEC
013640 006160 000006      G36:    ROL      6(R0)      ;GENERATE THE NEXT TEST PATTERN.
013644 006160 000004      ROL      4(R0)
013650 006160 000002      ROL      2(R0)
013654 006110      ROL      (R0)
013656 004737 014346      JSR      PC,GRESET       ;RESET DEFAULT PATTERN IN OUTPUT
                                ;BUFFER.

013662 077330      SOB      R3,G34
013664 004737 014500      JSR      PC,GSUM         ;TYPE ERROR SUMMARY.
3192  ;TEST ACCUMULATOR 5 WITH FLOATING ONE
013670 012737 044572 001244      MOV      #MNUM5,$TMP5
013676 012737 013732 001236      MOV      #G37,$TMP2
013704 012700 015136      MOV      #GPAT00,R0
013710 012701 015176      MOV      #GDAT00,R1
013714 012737 013732 001110     MOV      #G37,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
013722 004737 014270      JSR      PC,GSETUP        ;LOAD TEST PATTERN.
013726 012703 000102      MOV      #102,R3
013732 172410      G37:    LDD      (R0),AC0
013734 174005      STD      AC0,AC5
013736 172405      LDD      AC5,AC0          ;STORE THE TEST PATTERN.
013740 174011      STD      AC0,(R1)
013742 004737 014366      JSR      PC,GCMP          ;COMPARE THE DATA READ WITH
                                ;THAT WHICH WAS WRITTEN.

013746 005737 015132      TST      GFLAG1
013752 001004      BNE      G40
013754 005137 015132      COM      GFLAG1
013760 000261      SEC
013762 000401      BR       G41
013764 000241      G40:    CLC
013766 006160 000006      G41:    ROL      6(R0)      ;GENERATE THE NEXT TEST PATTERN.
013772 006160 000004      ROL      4(R0)
013776 006160 000002      ROL      2(R0)
014002 006110      ROL      (R0)
014004 004737 014346      JSR      PC,GRESET       ;RESET DEFAULT PATTERN IN OUTPUT
                                ;BUFFER.

014010 077330      SOB      R3,G37
014012 004737 014500      JSR      PC,GSUM         ;TYPE ERROR SUMMARY.
3193  ;TEST ACCUMULATOR 5 WITH FLOATING ZERO
014016 012737 044572 001244      MOV      #MNUM5,$TMP5
014024 012737 014060 001236      MOV      #G42,$TMP2

```



```

014032 012700 015146      MOV      #GPAT10,R0
014036 012701 015176      MOV      #GDAT00,R1
014042 012737 014060 001110  MOV      #G42,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
014050 004737 014270      JSR      PC,GSETUP        ;LOAD TEST PATTERN.
014054 012703 000102      MOV      #102,R3
014060 17241C      G42:    LDD      (R0),AC0
014062 174005      STD      AC0,AC5
014064 172405      LDD      AC5,AC0        ;STORE THE TEST PATTERN.
014066 174011      STD      AC0,(R1)
014070 004737 014366      JSR      PC,GCMP        ;COMPARE THE DATA READ WITH
                                ;THAT WHICH WAS WRITTEN.

014074 005737 015132      TST      GFLAG1
014100 001004      BNE      G43
014102 005137 015132      COM      GFLAG1
014106 000241      CLC
014110 000401      BR      G44
014112 000261      G43:    SEC
014114 006160 000006      G44:    ROL      6(R0)      ;GENERATE THE NEXT TEST PATTERN.
014120 006160 000004      ROL      4(R0)
014124 006160 000002      ROL      2(R0)
014130 006110      ROL      (R0)
014132 004737 014346      JSR      PC,GRESET      ;RESET DEFAULT PATTERN IN OUTPUT
                                ;BUFFER.

014136 077330      SOB      R3,G42
014140 004737 014500      JSR      PC,GSUM        ;TYPE ERROR SUMMARY.

3194
3195 014144 000137 015210      JMP      GDONE
3196
3197
3198
3199
3200
3201
3202
3203 014150      105      063      067  *****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
3204  CHPNUM: .ASCII ?E37 E45 E34 E42 E33 E41 E36 E44 E35 E43 E38 E46 E39 E47 E40 E48?
3205 014250 170000 007400 000360 BITSTS: .EVEN
                                .WORD 170000,7400,360,17,7777,170377,177417,177760
3206  ;USE THIS ROUTINE TO INITIALIZE ALL THE DATA BUFFERS.
3207 014270 012705 015132  GSETUP: MOV      #GFLAG1,R5
3208 014274 012704 000026      MOV      #26,R4
3209 014300 005025      1$:    CLR      (R5)+
3210 014302 077402      SOB      R4,1$
3211
3212 014304 012705 015146      MOV      #GPAT10,R5
3213 014310 012704 000010      MOV      #10,R4
3214 014314 005125      2$:    COM      (R5)+
3215 014316 077402      SOB      R4,2$
3216
3217 014320 020037 015136      GS1:    CMP      R0,GPAT00
3218 014324 001401      BEQ      3$
3219 014326 000207      RTS      PC
3220
3221 014330 012705 015176      3$:    MOV      #GDAT00,R5
3222 014334 012704 000004      MOV      #4,R4
3223 014340 005125      4$:    COM      (R5)+
3224 014342 077402      SOB      R4,4$
3225 014344 000207      RTS      PC

```

```

3226
3227 014346 012705 015176 GRESET: MOV #GDAT00,R5
3228 014352 012704 000004 MOV #4,R4
3229 014356 005025 1$: CLR (R5)+
3230 014360 077402 SOB R4,1$
3231 014362 000137 014320 JMP GS1
3232
3233 ;SEE IF THE DATA WRITTEN MATCHES THE DATA READ.
3234 014366 012705 015176 GCMP: MOV #GDAT00,R5
3235 014372 012704 000004 MOV #4,R4
3236 014376 010002 MOV R0,R2
3237 014400 022225 1$: CMP (R2)+,(R5)+
3238 014402 001402 BEQ 2$
3239 014404 000137 014414 JMP GERR1
3240 014410 077405 2$: SOB R4,1$
3241 014412 000207 RTS PC
3242
3243 ;COME HERE TO REPORT AND RECORD ERRORS.
3244 014414 012637 015206 GERR1: MOV (SP)+,GADR ;SAVE THE RETURN ADDRESS.
3245 014420 010037 001240 MOV R0,1240 ;COMPUTE 'OR' OF BAD DATA.
3246 014424 012705 015156 MOV #GAND0,R5
3247 014430 012704 000004 MOV #4,R4
3248 014434 051065 000010 1$: BIS (R0),10(R5)
3249 014440 012002 MOV (R0)+,R2
3250 014442 005102 COM R2
3251 014444 040225 BIC R2,(R5)+
3252 014446 077406 SOB R4,1$
3253 014450 013700 001240 MOV 1240,R0
3254 014454 005237 015134 INC GFLAG2 ;INCREMENT ERROR COUNT.
3255 014460 010037 001240 MOV R0,$TMP3
3256 014464 012737 015176 001242 MOV #GDAT00,$TMP4
3257 014472 104044 3$: ERROR +44
3258 014474 000177 000506 JMP @GADR
3259
3260 ;SEE IF ANY ERRORS HAVE OCCURRED AND WHETHER OR NOT AN ERROR SUMMARY
3261 ;SHOULD BE TYPED.
3262 014500 005737 015134 GSUM: TST GFLAG2 ;ANY ERRORS?
3263 014504 001410 BEQ 100$ ;BRANCH IF NOT
3264 014506 032777 020000 164424 BIT #SW13,@SWR ;INHIBIT ERROR PRINT OUT?
3265 014514 001405 BEQ 1$ ;BRANCH IF NOT INHIBITED
3266 014516 032777 000200 164414 BIT #SW7,@SWR ;PRINT SUMMARY?
3267 014524 001001 BNE 1$ ;BRANCH IF NOT
3268 014526 000207 100$: RTS PC ;EXIT - NO ERRORS TO REPORT
3269 014530 013737 015134 001246 1$: MOV GFLAG2,$TMP6 ;YES PRINT SUMMARY.
3270 014536 012737 015156 001240 MOV #GAND0,$TMP3
3271 014544 012737 015166 001242 MOV #GOR0,$TMP4
3272 014552 012637 015206 MOV (SP)+,GADR ;SAVE RETURN ADDRESS FOR POSSIBLE LOOPING
3273 014556 012737 014572 001116 MOV #2$,$ERRPC
3274 014564 112737 000045 001114 MOV #45,$ITEMB
3275 014572 004737 042010 2$: JSR PC,ERTYPE
3276 014576 010046 MOV R0,-(SP) ;SAVE R0
3277 014600 010146 MOV R1,-(SP) ;SAVE R1
3278 014602 010246 MOV R2,-(SP) ;SAVE R2
3279 014604 012700 014150 MOV #CHPNJM,R0 ;MOVE ADDRESS OF CHIP NUMBER ASCII'S TO R0
3280 014610 012702 015156 MOV #GAND0,R2 ;MOVE ADDRESS OF 'AND' DATA TO R2
3281 014614 112737 000077 015124 MOV #?,11$ ;MOVE ASCII '?' TO NEXT 3 LOCATIONS
3282 014622 112737 000077 015125 MOV #?,11$+1

```


3283	014630	112737	000077	015126		MOVW	#?,11\$+2		
3284	014636	104401	015124			TYPE	,11\$:TYPE ERROR MESSAGE BELOW
3285	014642	012701	014250		3\$:	MOV	#BITSTS,R1		:MOVE ADDRESS OF BIT TESTING TO R1
3286	014646	032112			4\$:	BIT	(R1)+,(R2)		:SEE IF ANY BITS IN THE GROUP WERE SET
3287	014650	001413				BEQ	5\$:BRANCH AROUND CHIP # PRINT IF NOT
3288	014652	116037	000000	015124		MOVW	0(R0),11\$+0		:MOVE 1ST ASCII DIGIT TO LOCATION
3289	014660	116037	000001	015125		MOVW	1(R0),11\$+1		:MOVE 2ND ASCII DIGIT TO LOCATION
3290	014666	116037	000002	015126		MOVW	2(R0),11\$+2		:MOVE 3RD ASCII DIGIT TO LOCATION
3291	014674	104401	015124			TYPE	,11\$:TYPE THE CHIP NUMBER
3292	014700	062700	000004		5\$:	ADD	#4,R0		:MOVE ASCII POINTER TO NEXT CHIP NUMBER
3293	014704	022701	014260			CMP	#BITSTS+10,R1		:SEE IF WE ARE DONE CHECKING THE 'AND' SETS
3294	014710	001356				BNE	4\$:BRANCH BACK IF NOT
3295	014712	062702	000002			ADD	#2,R2		:MAP TO NEXT 'AND' LOCATION
3296	014716	022702	015166			CMP	#GAND0+10,R2		:SEE IF ALL 'AND' LOCATIONS HAVE BEEN CHECKED
3297	014722	001347				BNE	3\$:BRANCH BACK IF NOT
3298	014724	012700	014150			MOV	#CHPNUM,R0		:RESET CHIP NUMBER ASCII POINTER IN R0
3299	014730	012702	015166			MOV	#GORO,R2		:MOVE 'OR' STARTING ADDRESS TO R2
3300	014734	011246			6\$:	MOV	(R2),-(SP)		:PUT 'OR' DATA ON STACK
3301	014736	046116	000010			BIC	10(R1),(SP)		:CLEAR BITS NOT UNDER TEST
3302	014742	022126				CMP	(R1)+,(SP)+		:SEE IF ANY BITS WERE FOUND CLEAR
3303	014744	001413				BEQ	7\$:BRANCH AROUND CHIP # PRINTING IF OK
3304	014746	116037	000000	015124		MOVW	0(R0),11\$:MOVE 1ST ASCII DIGIT TO LOCATION
3305	014754	116037	000001	015125		MOVW	1(R0),11\$+1		:MOVE 2ND ASCII DIGIT TO LOCATION
3306	014762	116037	000002	015126		MOVW	2(R0),11\$+2		:MOVE 3RD ASCII DIGIT TO LOCATION
3307	014770	104401	015124			TYPE	,11\$:TYPE THE CHIP NUMBER
3308	014774	062700	000004		7\$:	ADD	#4,R0		:POINT R0 TO NEXT ASCII CHIP REPRESENTATION
3309	015000	022701	014260			CMP	#BITSTS+10,R1		:SEE IF WE ARE DONE CHECKING THE 'AND' SETS
3310	015004	001353				BNE	6\$:BRANCH BACK IF NOT
3311	015006	012701	014250			MOV	#BITSTS,R1		:RESET BIT TEST CHECK WORDS ADDRESS TO R1
3312	015012	062702	000002			ADD	#2,R2		:MAP TO NEXT 'AND' LOCATION
3313	015016	022702	015166			CMP	#GAND0+10,R2		:SEE IF ALL 'AND' LOCATIONS HAVE BEEN CHECKED
3314	015022	001344				BNE	6\$:BRANCH BACK IF NOT
3315	015024	122737	000077	015124		CMPB	#?,11\$:SEE IF ANY CHIP NUMBER HAS BEEN LOADED
3316	015032	001002				BNE	8\$:BRANCH TO RETURN JUMP IF SO
3317	015034	104401	015124			TYPE	,11\$:TYPE THE QUESTION MARKS - COULDN'T ISOLATE TO A CHIP
3318	015040	104401	001313		8\$:	TYPE	,\$CRLF		:TYPE A <CRLF>
3319	015044	012602				MOV	(SP)+,R2		:RESTORE R2
3320	015046	012601				MOV	(SP)+,R1		:RESTORE R1
3321	015050	012600				MOV	(SP)+,R0		:RESTORE R0
3322	015052	000177	000130			JMP	@GADR		
3323	015056	000207			9\$:	RTS	PC		
3324									
3325	015060	200	103	110	10\$:	.ASCIZ	<CRLF>'CHIP NUMBERS TO INITIALLY LOOK AT'<CRLF>		
3326	015124	077	077	077	11\$:	.ASCIZ	'???'		
3327									
3328	015132	000000				GFLAG1:	.WORD 0		
3329	015134	000000				GFLAG2:	.WORD 0		
3330	015136	000000				GPAT00:	.WORD 0		
3331	015140	000000				GPAT01:	.WORD 0		
3332	015142	000000				GPAT02:	.WORD 0		
3333	015144	000000				GPAT03:	.WORD 0		
3334	015146	177777				GPAT10:	.WORD -1		
3335	015150	177777				GPAT11:	.WORD -1		
3336	015152	177777				GPAT12:	.WORD -1		
3337	015154	177777				GPAT13:	.WORD -1		
3338	015156	177777				GAND0:	.WORD -1		
3339	015160	177777				GAND1:	.WORD -1		

3340	015162	177777	GAND2:	.WORD	-1
3341	015164	177777	GAND3:	.WORD	-1
3342	015166	000000	GOR0:	.WORD	0
3343	015170	000000	GOR1:	.WORD	0
3344	015172	000000	GOR2:	.WORD	0
3345	015174	000000	GOR3:	.WORD	0
3346	015176	000000	GDATA0:	.WORD	0
3347	015200	000000	GDATA1:	.WORD	0
3348	015202	000000	GDATA2:	.WORD	0
3349	015204	000000	GDATA3:	.WORD	0
3350	015206	000000	GADR:	.WORD	0
3351	015210		GDONE:		
	015210	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?).

3358

```

.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.
:
*****
    
```

```

3359 015212 000004
3360 015214 012737 015222 001110
3361 015222 005037 015746
3362 015226 012700 015750
3363 015232 012701 016070
3364 015236 012703 000024
3365 015242 012120
3366 015244 077302
3367
3368 015246 004737 015674
3369
3370 015252 170011
3371
3372 015254 012700 015750
3373 015260 172410
3374 015262 174001
3375 015264 012700 015760
3376 015270 172410
3377 015272 174002
3378 015274 012700 015770
3379 015300 172410
3380 015302 174003
3381 015304 012700 016000
3382 015310 172410
3383 015312 174004
3384 015314 012700 016010
3385 015320 172410
3386 015322 174005
3387 015324 004737 015560
3388 015324 004737 015560
3389 015330 004737 015636
3390
3391
3392 015334 012700 015750
3393 015340 012702 000004
3394 015344 010001
3395 015346 005121
3396 015350 172410
3397 015352 174001
3398 015354 004737 015560
3399 015360 004737 015636
    
```

```

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: JSF 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.
    
```

3382 015364 077210
 015366 012700 015760
 015372 012702 000004
 015376 010001
 015400 005121
 015402 172410
 015404 174002
 015406 004737 015560
 015412 004737 015636
 015416 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.
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
    
```

3383 015420 012700 015770
 015424 012702 000004
 015430 010001
 015432 005121
 015434 172410
 015436 174003
 015440 004737 015560
 015444 004737 015636
 015450 077210

```

:COMPLIMENT EACH WORD OF THE DATA STORED IN ACCUMULATOR 3,
:RELOAD THAT ACCUMULATOR, READ ALL THE ACCUMULATORS BACK AND CHECK
:THE DATA.
MOV      #HA3W,R0
MOV      #4,R2
MOV      R0,R1
H7:      COM      (R1)+
          LDD      (R0),ACC
          STD      AC0,AC3
          JSR      PC,HSTD      ;READ ALL THE ACCUMULATORS BACK.
          JSR      PC,HCMP     ;CHECK THE DATA.
          SOB      R2,H7
    
```

3384 015452 012700 016000
 015456 012702 000004
 015462 010001
 015464 005121
 015466 172410
 015470 174004
 015472 004737 015560
 015476 004737 015636
 015502 077210

```

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

3385 015504 012700 016010
 015510 012702 000004
 015514 010001
 015516 005121
 015520 172410
 015522 174005
 015524 004737 015560
 015530 004737 015636
 015534 077210

```

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

3386
 3387 015536 005737 015746
 3388 015542 001402
 3389 015544 000137 016140
 3390
 3391 015550 005137 015746
 3392 015554 000137 015252
 3393

```

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



```

3394 ;STORE ALL ACCUMULATORS IN THE OUTPUT BUFFERS.
3395 015560 004737 015674 HSTD: JSR PC,HCLR ;CLEAR ALL OUTPUT BUFFERS.
3396 ;STORE ACCUMULATOR 1
      MOV #HA1R,R4
      LDD AC1,AC0
      STD AC0,(R4)
3397 ;STORE ACCUMULATOR 2
      MOV #HA2R,R4
      LDD AC2,AC0
      STD AC0,(R4)
3398 ;STORE ACCUMULATOR 3
      MOV #HA3R,R4
      LDD AC3,AC0
      STD AC0,(R4)
3399 ;STORE ACCUMULATOR 4
      MOV #HA4R,R4
      LDD AC4,AC0
      STD AC0,(R4)
3400 ;STORE ACCUMULATOR 5
      MOV #HA5R,R4
      LDD AC5,AC0
      STD AC0,(R4)
      RTS PC
3401 015614 012704 016050
3402 015620 172404
3403 015622 174014
3403 ;COMPARE DATA LOADED WITH DATA READ.
3404 015636 012637 015744 HCMP: MOV (SP)+,HADR ;SAVE RETURN ADDRESS.
3405 015642 012703 015750      MOV #HA1W,R3
3406 015646 012704 016020      MOV #HA1R,R4
3407 015652 012705 000024      MOV #24,R5
3408 015656 022324 HCMP1: CMP (R3)+,(R4)+
3409 015660 001402      BEQ HCMP2
3410 015662 000137 015712      JMP HERROR
3411 015666 077505 HCMP2: SOB R5,HCMP1
3412 015670 000177 000050      JMP @HADR
3413
3414 ;CLEAR THE DATA OUTPUT BUFFER.
3415 015674 012704 016020 HCLR: MOV #HA1R,R4
3416 015700 012705 000024      MOV #24,R5
3417 015704 005024 HCLR1: CLR (R4)+
3418 015706 077502      SOB R5,HCLR1
3419 015710 000207      RTS PC
3420
3421 ;REPORT ERF. OR.
3422 015712 HERROR:
3423 015712 012703 015750      MOV #HA1W,R3
3424 015716 012704 001236      MOV #STMP2,R4
3425 015722 012705 000012      MOV #12,R5
3426 015726 010324 1$: MOV R3,(R4)+
3427 015730 062703 000010      ADD #10,R3
3428 015734 077504      SOB R5,1$
3429 015736 104046 2$: ERROR +46
3430 015740 000137 016140      JMP HDONE
3431
3432
3433 015744 000000 HADR: .WORD 0
3434 015746 000000 HFLAG: .WORD 0
3435

```

3436	015750	000000	000000	000000	HA1W:	.WORD	0,0,0,0
3437	015760	000000	000000	000000	HA2W:	.WORD	0,0,0,0
3438	015770	000000	000000	000000	HA3W:	.WORD	0,0,0,0
3439	016000	000000	000000	000000	HA4W:	.WORD	0,0,0,0
3440	016010	000000	000000	000000	HA5W:	.WORD	0,0,0,0
3441							
3442	016020	000000	000000	000000	HA1R:	.WORD	0,0,0,0
3443	016030	000000	000000	000000	HA2R:	.WORD	0,0,0,0
3444	016040	000000	000000	000000	HA3R:	.WORD	0,0,0,0
3445	016050	000000	000000	000000	HA4R:	.WORD	0,0,0,0
3446	016060	000000	000000	000000	HA5R:	.WORD	0,0,0,0
3447							
3448	016070	073567	073567	073567	HDATA1:	.WORD	73567,73567,73567,73567
3449	016100	063146	063146	063146	HDATA2:	.WORD	63146,63146,63146,63146
3450	016110	010421	010421	010421	HDATA3:	.WORD	10421,10421,10421,10421
3451	016120	031463	031463	031463	HDATA4:	.WORD	31463,31463,31463,31463
3452	016130	042104	042104	042104	HDATA5:	.WORD	42104,42104,42104,42104
3453	016140				HDATA:		
	016140	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?).
    
```

3454
3455

3463

```
.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).
*
*****
```

```
TST13: SCOPE
S1:
3464 016142 000004
3464 016144
3465 016144 012737 016152 001110
3465 016152 170011
3466 016154 012700 016664
3466 016154 172410
3467 016160
3467 016160 172410
3468
3469 016162 012737 016364 000244
3470
3471
3472 016170 012700 000001
3473
3474 016174 012737 016574 000004
3475 016202 005003
3476
3477 016204 172407
3478 016206 170000
3479 016210 005203
3480 016212 005203
3481
3482 016214 012701 016674
3483 016220 174011
3484
3485 016222 012701 016674
3486 016226 012702 016664
3487 016232 012703 000004
3488 016236 022122
3489 016240 001402
3490 016242 000137 016524
3491 016246 077305
3492
3493 016250 000137 016550
3494
3495
3496 016254
3496 016254 012737 016262 001110
3497 016262 170011
3498
3499 016264 012700 016664
3500 016270 172410
3501
3502 016272 012737 016442 000244
3503 016300 012700 000001
3504 016304 012737 016626 000004
3505 016312 005003
3506
3507 016314 172406
3508 016316 170000

S2: LDD AC7,AC0
S3: CFCC
S4: INC R3
S5: CMP (R1)+,(R2)+
S6: BEQ S6
S7: SOB SERR2
S8: R3,S5
S9: JMP SERR3

;NO TRAP OCCURRED!!
;SEE IF ACO WAS MODIFIED.

;NOW TEST AC6.
S7:
S7: MOV #1$,$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
S7: SETD ;SET FD
S7: MOV #SPAT10,R0 ;LOAD ACO
S7: LDD (R0),AC0
S7: MOV #SERR4,FPVECT
S7: MOV #1,R0
S7: MOV #SERR5,ERRVECT
S7: CLR R3
S7: LDD AC6,AC0
S7: CFCC
```

```

3509 016320 005203          INC      R3
3510 016322 005203          S10:   INC      R3
3511
3512 016324 012701 016674          MOV     #SDAT00,R1
3513 016330 174011          STD     ACO,(R1)          ;NO TRAP! GET ACC
3514
3515 016332 012701 016674          MOV     #SDAT00,R1          ;WAS ACO MODIFIED.
3516 016336 012702 016664          MOV     #SPAT10,R2
3517 016342 012703 000004          MOV     #4,R3
3518 016346 022122          S11:   CMP     (R1)+,(R2)+
3519 016350 001402          BEQ     S12
3520 016352 000137 016536          JMP     SERR6
3521 016356 077305          S12:   SOB     R3,S11
3522 016360 000137 016562          JMP     SERR7
3523
3524          ;TRAPPED TO 244.
3525 016364 021627 016206          SERR0: CMP     (SP),#S3          ;PC OF TRAP CORRECT?
3526 016370 001402          BEQ     1$
3527 016372 000137 042534          JMP     FPSPUR
3528
3529 016376 012737 016254 016660 1$:   MOV     #S7,SADR
3530
3531 016404 011637 001236          SERR10: MOV     (SP),$TMP2
3532 016410 022626          CMP     (SP)+,(SP)+
3533 016412 005004          CLR     R4
3534 016414 170204          STFPS  R4          ;IS FPS CORRECT?
3535 016416 022704 100200          CMP     #100200,R4
3536 016422 001020          BNE     SERR15
3537
3538 016424 005004          CLR     R4
3539 016426 170304          STST   R4          ;IS FEC CORRECT?
3540 016430 022704 000002          CMP     #2,R4
3541 016434 001023          BNE     SERR20
3542 016436 000177 000216          JMP     @SADR
3543
3544 016442 021627 016316          SERR4: CMP     (SP),#S9
3545 016446 001402          BEQ     1$
3546 016450 000137 042534          JMP     FPSPUR
3547 016454 012737 016704 016660 1$:   MOV     #SDONE,SADR
3548 016462 000750          BR     SERR10
3549
3550          ;REPORT FPS FAILURE:
3551 016464 012737 100200 001242 SERR15: MOV     #100200,$TMP4
3552 016472 010437 001240          MOV     R4,$TMP3
3553 016476 104117          1$:   ERROR +117
3554 016500 000177 000154          JMP     @SADR
3555
3556          ;REPORT FEC BAD:
3557 016504 012737 000002 001242 SERR20: MOV     #2,$TMP4
3558 016512 010437 001240          MOV     R4,$TMP3
3559 016516 104120          1$:   ERROR +120
3560 016520 000177 000134          JMP     @SADR
3561
3562
3563          ;ACO WAS MODIFIED. (BUT FSRC) FORK FAILED.
3564 016524 012737 016204 001236 SERR2: MOV     #S2,$TMP2
3565 016532 104112          1$:   ERROR +112
    
```



```

3566 016534 000463          BR      SDONE
3567 016536 012737 016314 001236 SERR6: MOV    #S8,$TMP2
3568 016544 104114          1$:    ERROR  +114
3569 016546 000456          BR      SDONE
3570
3571 016550 012737 016204 001236 SERR3: MOV    #S2,$TMP2
3572 016556 104111          1$:    ERROR  +111
3573 016560 000451          BR      SDONE
3574 016562 012737 016314 001236 SERR7: MOV    #S8,$TMP2
3575 016570 104113          1$:    ERROR  +113
3576 016572 000444          BR      SDONE
3577
3578          ;FAILURE OF (BUT FSRC) CAUSED AN ODD ADDRESS TRAP TO 4.
3579 016574 021627 016206 SERR1: CMP    (SP),#S3          ;DID TRAP OCCUR ON TESTED INSTRUCTION?
3580 016600 001405          BEQ    1$
3581 016602 021627 016212          CMP    (SP),#S4
3582 016606 001402          BEQ    1$
3583 016610 000137 042566          JMP    CPSPUR
3584
3585 016614 011637 001236          1$:    MOV    (SP),$TMP2
3586 016620 022626          CMP    (SP)+,(SP)+
3587 016622 104115          2$:    ERROR  +115
3588 016624 000427          BR      SDONE
3589
3590 016626 021627 016314 SERR5: CMP    (SP),#S8          ;DID TRAP OCCUR ON TEST INSTRUCTION?
3591 016632 001405          BEQ    1$
3592 016634 021627 016316          CMP    (SP),#S9
3593 016640 001402          BEQ    1$
3594 016642 000137 042566          JMP    CPSPUR
3595
3596 016646 011637 001236          1$:    MOV    (SP),$TMP2
3597 016652 022626          CMP    (SP)+,(SP)+
3598 016654 104116          2$:    ERROR  +116
3599 016656 000412          BR      SDONE
3600
3601 016660 000000          SADR:  0
3602 016662 177777          -1
3603 016664 010421          SPAT10: .WORD 10421
3604 016666 021042          SPAT11: .WORD 21042
3605 016670 031463          SPAT12: .WORD 31463
3606 016672 042104          SPAT13: .WORD 42104
3607
3608 016674 000000          SDAT00: .WORD 0
3609 016676 000000          SDAT01: .WORD 0
3610 016700 000000          SDAT02: .WORD 0
3611 016702 000000          SDAT03: .WORD 0
3612
3613 016704          SDONE:
      016704 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?).
    
```

3614

3621

```
.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
3622 016706 000004 016716 001110  MOV      #J1,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
3623 016710 012737
3624 016716
3625 016716 170011  J1:      SETD              ;SET DOUBLE MODE
3626
3627 016720 012700 017174  MOV      #JDAT0,R0
3628 016724 172410  LDD      (R0),ACO      ;LOAD ACO
3629
3630 016726 012700 017154  MOV      #JDATIO,R0
3631 016732 005003  CLR      R3
3632 016734 012737 017024 000004  MOV      #JERR0,ERRVECT
3633
3634 016742 172420  J2:      LDD      (R0)+,ACO      ;TEST INSTRUCTION
3635 016744 005203  J3:      INC      R3
3636 016746 005203  J4:      INC      R3
3637
3638 016750 012701 017164  MOV      #JDAT00,R1
3639 016754 174011  STD      ACO,(R1)      ;PICK UP RESULTS
3640
3641 016756 020027 017144  CMP      R0,#JBUFO      ;WAS AN AUTO
3642 016762 001001  BNE      1$             ;DECREMENT EXECUTED?
3643 016764 000442  BR       JERR1
3644
3645 016766 012702 017154  1$:      MOV      #JDATIO,R2      ;IS DATA CORRECT?
3646 016772 012703 017164  MOV      #JDAT00,R3
3647 016776 012704 000004  MOV      #4,R4
3648 017002 022223  J5:      CMP      (R2)+,(R3)+
3649 017004 001401  BEQ      J6
3650 017006 000443  BR       JERR2
3651 017010 077404  J6:      SOB      R4,J5
3652
3653 017012 022700 017164  CMP      #JDATIO+10,R0  ;WAS R0 INCREM.
3654 017016 001401  BEQ      J7             ;BY 10 (OCTAL)
3655 017020 000424  BR       JERR1
3656
3657 017022 000470  J7:      BR       JDONE
3658
3659 ;IF A TRAP THROUGH 4 OCCURS COME HERE
3660
3661 017024 021627 016744  JERR0:   CMP      (SP),#J3      ;SEE IF THE TRAP
3662 017030 001405  BEQ      J10            ;OCCURRED ON THE
3663 017032 021627 016746  CMP      (SP),#J4      ;TESTED INSTRUCTION
3664 017036 001402  BEQ      J10
3665 017040 000137 042566  JMP      CPSPUR
3666
3667 017044 012737 000762 001240  J10:    MOV      #762,$TMP3      ;REPORT FSRC FLOW
3668 017052 012737 000322 001242  MOV      #322,$TMP4      ;FAILURE
3669 017060 011637 001236  MOV      (SP),$TMP2
```



```
3670 017064 022626
3671 017066 104052
3672 017070 000445
3673
3674 017072
3675 017072 012737 016742 001236
3676 017100 010037 001240
3677 017104 012737 017164 001242
3678 017112 104053
3679 017114 000433
3680
3681
3682
3683 017116
3684 017116 012737 016742 001236
3685 017124 012737 017154 001240
3686 017132 012737 017164 001242
3687 017140 104054
3688 017142 000420
3689
3690 017144 010421
3691 017146 021042
3692 017150 042104
3693 017152 031463
3694
3695 017154 052525
3696 017156 114631
3697 017160 063146
3698 017162 073567
3699
3700 017164 000000
3701 017166 000000
3702 017170 000000
3703 017172 000000
3704
3705 017174 177777
3706 017176 177777
3707 017200 177777
3708 017202 177777
3709
3710
3711 017204
      017204 104413
      JERR1:
      JERR2:
      JBUF0: .WORD 010421
      JBUF1: .WORD 021042
      JBUF2: .WORD 042104
      JBUF3: .WORD 031463
      JDAT10: .WORD 052525
      JDAT11: .WORD 114631
      JDAT12: .WORD 063146
      JDAT13: .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
      :REPORT, RO NOT
      :CORRECTLY AFFECTED
      :REPORT DATA FAILURE
      :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?).
      1$:
      1$:
      1$:
      1$:
```

3720

```

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

```

```

3721 017206 000004 017216 001110 TST15: SCOPE
3722 017210 012737          MOV      #K1,$LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
3723 017216          K1:
3724 017216 170011          SETD           ;SET DOUBLE MODE
3725
3726 017220 012700 017472          MOV      #KPATO,R0
3727 017224 172410          LDD      (R0),ACO      ;LOAD A DEFAULT
3728                                ;PATTERN INTO ACO
3729 017226 012700 017452          MOV      #KBUFO,R0
3730 017232 005003          CLR      R3
3731 017234 012737 017324 000004    MOV      #KERR0,ERRVECT
3732
3733 017242 172440          K2: LDD      -(R0),ACO      ;TEST INSTRUCTION
3734 017244 005203          K3: INC      R3
3735 017246 005203          K4: INC      R3
3736
3737 017250 012701 017462          MOV      #KDAT00,R1
3738 017254 174011          STD      ACO,(R1)      ;PICK UP THE RESULT
3739
3740 017256 020027 017462          CMP      R0,#KBUFO+10  ;WAS AN AUTO
3741 017262 001001          BNE     1$            ;INCREMENT EXECUTED
3742 017264 000441          BR      KERR1
3743
3744 017266 012702 017442          1$: MOV      #KDAT10,R2      ;IS DATA CORRECT?
3745 017272 012703 017462          MOV      #KDAT00,R3
3746 017276 012704 000004          MOV      #4,R4
3747 017302 022223          K5: CMP      (R2)+,(R3)+
3748 017304 001401          BEQ     K6
3749 017306 000442          BR      KERR2
3750 017310 077404          K6: SOB     R4,K5
3751
3752 017312 022700 017442          CMP      #KBUFO-10,R0  ;WAS R0 DECREMENTED
3753 017316 001401          BEQ     K7            ;PROPERLY?
3754 017320 000423          BR      KERR1
3755
3756 017322 000467          K7: BR      KDONE
3757
3758                                ;TRAP TO HERE ON AN ODD ADDRESS ERROR
3759
3760 017324 021627 017244          KERR0: CMP     (SP),#K3      ;SEE IF THE ERROR
3761 017330 001405          BEQ     K10           ;OCCURRED AT THE
3762 017332 021627 017246          CMP     (SP),#K4      ;INSTRUCTION TESTED.
3763 017336 001402          BEQ     K10
3764 017340 000137 042566          JMP     CPSPUR
3765
3766 017344 012737 000762 001240    K10: MOV     #762,$TMP3      ;REPORT FAILURE IN
3767 017352 012737 000324 001242    MOV     #324,$TMP4      ;FSRC FLOWS
3768 017360 011637 001236          MOV     (SP),$TMP2

```



```

3769 017364 104055          1$:      ERROR      +55
3770 017366 000445          BR          KDONE
3771
3772 017370
3773 017370 012737 017242 001236 KERR1:    MOV      #K2,$TMP2      :REPORT, RO
3774 017376 010037 001240          MOV      R0,$TMP3      :INCORRECTLY AFFECTED.
3775 017402 012737 017442 001242          MOV      #KDAT10,$TMP4
3776 017410 104056          1$:      ERROR      +56
3777 017412 000433          BR          KDONE
3778
3779          ;REPORT DATA FAILURE
3780
3781 017414
3782 017414 012737 017242 001236 KERR2:    MOV      #K2,$TMP2
3783 017422 012737 017442 001240          MOV      #KDAT10,$TMP3
3784 017430 012737 017462 001242          MOV      #KDAT00,$TMP4
3785 017436 104057          1$:      ERROR      +57
3786 017440 000420          BR          KDONE
3787
3788 017442 052525          KDAT10:  .WORD    052525
3789 017444 114631          KDAT11:  .WORD    114631
3790 017446 063140          KDAT12:  .WORD    063140
3791 017450 073567          KDAT13:  .WORD    073567
3792
3793 017452 010421          KBUF0:   .WORD    010421
3794 017454 031463          KBUF1:   .WORD    031463
3795 017456 042104          KBUF2:   .WORD    042104
3796 017460 021042          KBUF3:   .WORD    021042
3797
3798 017462 000000          KDAT00:  .WORD    0
3799 017464 000000          KDAT01:  .WORD    0
3800 017466 000000          KDAT02:  .WORD    0
3801 017470 000000          KDAT03:  .WORD    0
3802
3803 017472 177777          KPAT0:   .WORD   -1
3804 017474 177777          KPAT1:   .WORD   -1
3805 017476 177777          KPAT2:   .WORD   -1
3806 017500 177777          DPAT3:   .WORD   -1
3807
3808 017502          KDONE:
      017502 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?).

3809
3816
  
```

3817

.....
 .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)
 *

3818	017504	000004							
3819	017506	012737	017514	001110	TST16:	SCOPE			
3820	017514					MOV	#L1,\$LPERR		;SET UP THE LOOP ON ERROR ADDRESS.
3821	017514	170011			L1:	SETD			;SET DOUBLE MODE
3822									
3823	017516	012700	017764			MOV	#LPAT10,R0		
3824	017522	172410				LDD	(R0),AC0		;LOAD AC0
3825									
3826	017524	012700	020006			MOV	#LDAT10,R0		;SET UP THE INPUT
3827	017530	012701	017774			MOV	#LPAT20,R1		;DATA
3828	017534	012702	000004			MOV	#4,R2		
3829									
3830	017540	012120			1\$:	MOV	(R1)+,(R0)+		
3831	017542	077202				SOB	R2,1\$		
3832									
3833	017544	012700	020006			MOV	#LDAT10,R0		
3834	017550	005003				CLR	R3		
3835	017552	170001				SETF			;CLEAR FD.
3836									
3837	017554	172420			L2:	LDF	(R0)+,AC0		
3838	017556	005203			L3:	INC	R3		
3839									
3840	017560				L4:	SETD			;SET FD
3841	017560	170011							
3842									
3843	017562	012701	020020			MOV	#LDAT00,R1		
3844	017566	174011				STD	AC0,(R1)		;PICK UP RESULTS
3845									
3846	017570	020027	020012			CMP	R0,#LDAT12		;WAS R0 INCREMENTED
3847	017574	001401				BEQ	1\$;CORRECTLY BY 4
3848	017576	000421				BR	LERR1		
3849									
3850	017600	012737	177777	020012	1\$:	MOV	#-1,LDAT12		
3851	017606	012737	177777	020014		MOV	#-1,LDAT13		
3852	017614	012702	020006			MOV	#LDAT10,R2		;IS DATA CORRECT
3853	017620	012703	020020			MOV	#LDAT00,R3		
3854	017624	012704	000004			MOV	#4,R4		
3855									
3856	017630	022223			L5:	CMP	(R2)+,(R3)+		
3857	017632	001401				BEQ	L6		
3858	017634	000427				BR	LERR2		
3859	017636	077404			L6:	SOB	R4,L5		
3860									
3861	017640	000473				BR	LDONE		
3862									
3863	017642				LERR1:				;REPORT FAILURE
3864	017642	012737	017554	001236		MOV	#L2,\$TMP2		;R0 NOT INCREMENTED
3865	017650	010037	001240			MOV	R0,\$TMP3		;BY 4


```

3866 017654 012737 020012 001242      MOV      #LDAT12,$TMP4
3867 017662 104060      1$:     ERROR    +60
3868 017664 000461          BR      LDONE
3869
3870 017666          LERR3:          ;REPORT DATA FAILURE.
3871 017666 012737 017554 001236      MOV      #L2,$TMP2
3872 017674 012737 020006 001240      MOV      #LDAT10,$TMP3
3873 017702 012737 020020 001242      MOV      #LDAT00,$TMP4
3874 017710 104061      1$:     ERROR    +61
3875 017712 000446          BR      LDONE
3876
3877 017714 012702 017774          LERR2: MOV      #LPAT20,R2      ;DID (BUT FD)
3878 017720 012703 020020          MOV      #LDAT00,R3      ;FAIL.
3879 017724 012704 000004          MOV      #4,R4
3880 017730 022223      1$:     CMP      (R2)+,(R3)+
3881 017732 001355          BNE     LERR3
3882 017734 077403          SOB     R4,1$
3883 017736 012737 017554 001236      MOV      #L2,$TMP2
3884 017744 012737 020006 001240      MOV      #LDAT10,$TMP3
3885 017752 012737 020022 001242      MOV      #LDAT01,$TMP4
3886 017760 104062      2$:     ERROR    +62
3887 017762 000422          BR      LDONE
3888
3889 017764 177777          LPAT10: .WORD    -1
3890 017766 177777          LPAT11: .WORD    -1
3891 017770 177777          LPAT12: .WORD    -1
3892 017772 177777          LPAT13: .WORD    -1
3893
3894 017774 052525          LPAT20: .WORD    052525
3895 017776 114631          LPAT21: .WORD    114631
3896 020000 063142          LPAT22: .WORD    063142
3897 020002 073567 000001          LPAT23: .WORD    073567,1
3898 020006 000000          LDAT10: .WORD    0
3899 020010 000000          LDAT11: .WORD    0
3900 020012 000000          LDAT12: .WORD    0
3901 020014 000000 000001          LDAT13: .WORD    0,1
3902 020020 000000          LDAT00: .WORD    0
3903 020022 000000          LDAT01: .WORD    0
3904 020024 000000          LDAT02: .WORD    0
3905 020026 000000          LDAT03: .WORD    0
3906
3907 020030          LDONE:
      020030 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?).
3908
3916
    
```

3917

```
.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
```

```

020032 000004
3918
3919 020034
3920 020034 170011
3921
3922 020036 012700 020330
3923 020042 172410
3924
3925 020044 005004
3926 020046 012737 020270 000004
3927
3928 020054 172427 000000
3929 020056
3930 020056 005204
3931 020060 005204
3932 020062 005204
3933 020064 005204
3934
3935 020066 020427 000003
3936 020072 001401
3937 020074 000443
3938
3939
3940
3941 020076 012700 020350
3942 020102 174010
3943
3944 020104 012700 020350
3945 020110 022720 005204
3946 020114 001401
3947 020116 000451
3948 020120 012701 000003
3949 020124 005720
3950 020126 001002
3951 020130 077103
3952 020132 000512
3953
3954 020134 012700 020350
3955 020140 012701 000004
3956 020144 022720 005204
3957 020150 001401
3958 020152 000433
3959 020154 077105
3960
3961 020156
3962 020156 012737 020054 001236
3963 020164 012737 020340 001240
3964 020172 012737 020350 001242

M1: SETD
MOV #MPAT10,R0
LDD (R0),AC0 ;LOAD BACKGROUND
;PATTERN INTO AC0.
CLR R4
MOV #MERR3,ERRVECT
M15: LDD #0,AC0 ;TEST INSTRUCTION
;=-2
;WORD
INC R4 ;NOTE THAT
M2: INC R4 ;005204=INC R4
M3: 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$: MOV #MDAT00,R0
STD AC0,(R0) ;GET THE DATA
MOV #MDAT00,R0
CMP #5204,(R0)+ ;IS THE DATA CORRECT?
BEQ M5
BR MERR1
M5: MOV #3,R1
M6: TST (R0)+
BNE M7
SOB R1,M6
BR MDONE
M7: MOV #MDAT00,R0 ;DID (BUT GRM) FAIL?
MOV #4,R1
M8: CMP #5204,(R0)+
BEQ M9
BR MERR1
M9: SOB R1,M8
MERR2: MOV #M15,$TMP2 ;REPORT FAILURE
MOV #MPAT20,$TMP3 ;OF (BUT GR7)
MOV #MDAT00,$TMP4
```

```

3965 020200 104063          1$:  ERROR  +63
3966 020202 000466          BR      MDONE
3967
3968 020204 012705 020060      MERR0: MOV    #M2,R5          :REPORT FAILURE
3969 020210 010537 001242      MOV    R5,$TMP4        :PC INCREMENTED
3970 020214 162704 000003      SUB    #3,R4
3971 020220 006304          ASL    R4
3972 020222 160405          SUB    R4,R5
3973 020224 010537 001240      MOV    R5,$TMP3
3974 020230 012737 020054 001236      MOV    #M15,$TMP2
3975 020236 104064          1$:  ERROR  +64
3976 020240 000447          BR      MDONE
3977
3978 020242          MERR1:          :REPORT DATA
3979 020242 012737 020054 001236      MOV    #M15,$TMP2        :FAILURE
3980 020250 012737 020350 001240      MOV    #MDAT00,$TMP3
3981 020256 012737 020340 001242      MOV    #MPAT20,$TMP4
3982 020264 104066          1$:  ERROR  +66
3983 020266 000434          BR      MDONE
3984          ;TRAP TO HERE THROUGH 4.
3985 020270 032716 000001      MERR3: BIT    #1,(SP)      :SEE IF THE
3986 020274 001002          BNE    1$                :TRAP TO 4 OCCURRED
3987 020276 000137 042566          JMP    CPSPUR            :BECAUSE OF AN
3988          :ODD ADDRESS
3989 020302 011637 001240      1$:  MOV    (SP),$TMP3      :IF YES REPORT
3990 020306 012737 020060 001242      MOV    #M2,$TMP4        :BAD CONSTANT
3991 020314 012737 020054 001236      MOV    #M15,$TMP2      :GENERATED
3992 020322 022626          CMP    (SP)+,(SP)+
3993 020324 104065          2$:  ERROR  +65
3994 020326 000414          BR      MDONE
3995
3996 020330 177777          MPAT10: .WORD  -1
3997 020332 177777          MPAT11: .WORD  -1
3998 020334 177777          MPAT12: .WORD  -1
3999 020336 177777          MPAT13: .WORD  -1
4000
4001 020340 005204          MPAT20: .WORD  5204
4002 020342 005204          MPAT21: .WORD  5204
4003 020344 005204          MPAT22: .WORD  5204
4004 020346 005204          MPAT23: .WORD  5204
4005
4006 020350 000000          MDAT00: .WORD  0
4007 020352 000000          MDAT01: .WORD  0
4008 020354 000000          MDAT02: .WORD  0
4009 020356 000000          MDAT03: .WORD  0
4010
4011 020360          MDONE:
      020360 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?).
4012
4019
    
```


4020

```
.SBTTL TEST # 20 - FSRC MODE 3 TEST  
:*****  
:TEST 20 FSRC MODE 3 TEST  
:*****  
:* THIS IS A TEST OF FSRC MODE 3, AUTO INCREMENT  
:* DEFERRED  
:*****  
TST20: SCOPE
```

4021	020362	000004			
4022	020364				
4023	020364	170011			
4024					
4025	020366	012700	021046		
4026	020372	172410			
4027					
4028	020374	012700	021034		
4029	020400	005003			
4030	020402	012737	020556	000004	
4031					
4032					
4033	020410	172430			
4034	020412	005203			
4035	020414	005203			
4036					
4037	020416	012701	021014		
4038	020422	174011			
4039					
4040	020424	020027	021036		
4041	020430	001437			
4042					
4043	020432	020027	021044		
4044	020436	001001			
4045	020440	000506			
4046					
4047	020442	020027	021024		
4048	020446	001001			
4049	020450	000520			
4050					
4051	020452	020027	021034		
4052	020456	001023			
4053					
4054	020460	012702	021014		
4055	020464	012703	000004		
4056	020470	022227	177777		
4057	020474	001002			
4058	020476	077304			
4059	020500	000510			
4060					
4061	020502	012702	021014		
4062	020506	012703	021034		
4063	020512	012704	000004		
4064	020516	022223			
4065	020520	001002			
4066	020522	077403			
4067	020524	000502			
4068					

```
N1: SETD ;SET FD MODE  
MOV #NPAT10,R0  
LDD (R0),ACO ;LOAD ACO WITH A DEFAULT  
;PATTERN  
MOV #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)+,ACO  
N3: INC R3  
N4: INC R3  
MOV #NDAT00,R1  
STD ACO,(R1) ;GET THE DATA  
CMP R0,#NPAT20+2 ;WAS R0 INCREMENTED  
BEQ N12 ;BY 2?  
N5: CMP R0,#NPAT20+10 ;FSRC MODE 2?  
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  
N8: 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
```

```

4069 020526 000505      N11:  BR      NERR5
4070
4071 020530 012702 021014  N12:  MOV      #NDAT00,R2      ;DATA CORRECT?
4072 020534 012703 021056      MOV      #NDAT10,R3
4073 020540 012704 000004      MOV      #4,R4
4074 020544 022223      N13:  CMP      (R2)+,(R3)+
4075 020546 001002      BNE      N14
4076 020550 077403      SOB      R4,N13
4077 020552 000545      BR       NDONE
4078
4079 020554 000504      N14:  BR       NERR6
4080
4081      ;IF AN ODD ADDRESS TRAP OCCURS COME HERE
4082      ;TO SEE IF THE FAILURE WAS IN THE FSRC
4083      ;FLOWS
4084
4085 020556 022716 020414  NERR0:  CMP      #N4,(SP)      ;FSRC MODE 6 OR 7?
4086 020562 001412      BEQ      NERR10
4087 020564 022716 020412      CMP      #N3,(SP)
4088 020570 001402      BEQ      1$
4089 020572 000137 042566      JMP      CPSPUR
4090 020576 020027 021032  1$:    CMP      R0,#NPAT20-2      ;FSRC MODE 5?
4091 020602 001407      BEQ      NERR11
4092 020604 000137 042566      JMP      CPSPUR
4093
4094 020610      NERR10:  MOV      (SP),$TMP2      ;WENT TO FSRC
4095 020610 011637 001236      CMP      (SP)+,(SP)+      ;MODE 6 OR 7.
4096 020614 022626      1$:    ERROR   +67
4097 020616 104067      BR       NDONE
4098 020620 000522
4099
4100 020622 011637 001236  NERR11:  MOV      (SP),$TMP2      ;WENT TO FSRC
4101 020626 022626      CMP      (SP)+,(SP)+      ;MODE 5.
4102 020630 012737 000627 001244      MOV      #627,$TMP5
4103 020636 012737 000323 001250      MOV      #323,$TMP7
4104 020644 012737 000325 001246      MOV      #325,$TMP6
4105 020652 104070      1$:    ERROR   +70
4106 020654 000504      BR       NDONE
4107 020656 012737 000322 001246  NERR1:  MOV      #322,$TMP6      ;FSRC MODE 2.
4108 020664 012737 000627 001244  NERR20:  MOV      #627,$TMP5
4109 020672 012737 000323 001250      MOV      #323,$TMP7
4110 020700 012737 020410 001236      MOV      #N2,$TMP2
4111 020706 104071      1$:    ERROR   +71
4112 020710 000466      BR       NDONE
4113 020712 012737 000324 001246  NERR2:  MOV      #324,$TMP6      ;FSRC MODE 4
4114 020720 000761      BR       NERR20
4115 020722 012737 000320 001246  NERR3:  MOV      #320,$TMP6      ;FSRC MODE 0
4116 020730 000755      BR       NERR20
4117 020732 012737 000321 001246  NERR4:  MOV      #321,$TMP6      ;FSRC MODE 1
4118 020740 000751      BR       NERR20
4119
4120 020742 010037 001240      NERR5:  MOV      R0,$TMP3      ;R0 NOT
4121 020746 012737 021036 001242      MOV      #NPAT20+2,$TMP4  ;INCREMENTED
4122 020754 012737 020410 001236      MOV      #N2,$TMP2      ;PROPERLY.
4123 020762 104072      1$:    ERROR   +72
4124 020764 000440      BR       NDONE
4125

```

```
4126 020766          NERR6:          ;DATA FAILURE.
4127 020766 012737 020410 001236      MOV      #N2,$TMP2
4128 020774 012737 021014 001240      MOV      #NDAT00,$TMP3
4129 021002 012737 021056 001242      MOV      #NDAT10,$TMP4
4130 021010 104073          1$:      ERROR    +73
4131 021012 000425          BR        NDONE
4132
4133 021014 000000          NDAT00: .WORD 0
4134 021016 000000          NDAT01: .WORD 0
4135 021020 000000          NDAT02: .WORD 0
4136 021022 000000 052525 052525  NDAT03: .WORD 0,52525,52525,52525,52525
4137 021034 021056          NPAT20: .WORD NDAT10
4138 021036 070707          NPAT21: .WORD 070707
4139 021040 070707          NPAT22: .WORD 070707
4140 021042 070707 000001  NPAT23: .WORD 070707,1
4141 021046 177777          NPAT10: .WORD -1
4142 021050 177777          NPAT11: .WORD -1
4143 021052 177777          NPAT12: .WORD -1
4144 021054 177777          NPAT13: .WORD -1
4145
4146 021056 010421          NDAT10: .WORD 010421
4147 021060 021042          NDAT11: .WORD 021042
4148 021062 031463          NDAT12: .WORD 031463
4149 021064 042104          NDAT13: .WORD 042104
4150
4151 021066          NDONE:
      021066 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?).
```

4152
4159

4160

```

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

4161	021070	000004				
4162	021072			01:	SETD	;SET FD MODE
4163	021072	170011				
4164						
4165	021074	012700	021552		MOV #OPAT10,R0	
4166	021100	172410			LDD (R0),AC0	;LOAD AC0 WITH A
4167						;DEFAULT PATTERN.
4168	021102	012700	021540		MOV #OPAT21,R0	
4169	021106	005003			CLR R3	
4170	021110	012737	021262	000004	MOV #OERR0,ERRVEC	;IF A FAILURE
4171						;OCCURS IN THE FSRC
4172						;FLOWS AN ODD ADDR.
4173						;TRAP TO 4 MAY OCCUR.
4174	021116	172450		02:	LDD @-(R0),AC0	;TEST INSTRUCTION
4175	021120	005203		03:	INC R3	
4176	021122	005203		04:	INC R3	
4177						
4178	021124	012701	021520		MOV #ODAT00,R1	
4179	021130	174011			STD AC0,(R1)	;GET THE DATA
4180						
4181	021132	020027	021536		CMP R0,#OPAT20	;WAS R0 DECREMENTED
4182	021136	001436			BEQ 012	;BY 2?
4183						
4184	021140	020027	021550	05:	CMP R0,#OPAT21+10	;FSRC MODE 2
4185	021144	001001			BNE 06	
4186	021146	000505			BR OERR1	
4187						
4188	021150	020027	021530	06:	CMP R0,#OPAT21-10	;FSRC MODE 4?
4189	021154	001001			BNE 07	
4190	021156	000517			BR OERR2	
4191						
4192	021160	020027	021540	07:	CMP R0,#OPAT21	
4193						
4194	021164	012702	021522		MOV #ODAT01,R2	;FSRC MODE 0?
4195	021170	012703	000004		MOV #4,R3	
4196	021174	022227	177777	08:	CMP (R2)+,#-1	
4197	021200	001002			BNE 09	
4198	021202	077304			SOB R3,08	
4199	021204	000510			BR OERR3	
4200						
4201	021206	012702	021520	09:	MOV #ODAT00,R2	;FSRC MODE 1?
4202	021212	012703	021540		MOV #OPAT21,R3	
4203	021216	012704	000004		MOV #4,R4	
4204	021222	022223		010:	CMP (R2)+,(R3)+	
4205	021224	001002			BNE 011	
4206	021226	077403			SOB R4,010	
4207	021230	000502			BR OERR4	
4208						

```

4209 021232 000505      011:  BR      OERR5
4210
4211 021234 012702 021520      012:  MOV      #ODAT00,R2      ;DATA CORRECT?
4212 021240 012703 021562      MOV      #ODATIO,R3
4213 021244 012704 000004      MOV      #4,R4
4214 021250 022223      013:  CMP      (R2)+,(R3)+
4215 021252 001002      BNE      014
4216 021254 077403      SOB      R4,013
4217 021256 000545      BR       ODONE
4218
4219 021260 000504      014:  BR       OERR6
4220
4221      ;IF AN ODD ADDRESS TRAP OCCURS COME
4222      ;HERE TO SEE IF THE FAILURE WAS IN THE
4223      ;FSRC FLOWS:
4224
4225 021262 022716 021122      OERR0:  CMP      #04,(SP)      ;FSRC MODE 6 OR 7?
4226 021266 001412      BEQ      OERR10
4227 021270 022716 021120      CMP      #03,(SP)
4228 021274 001402      BEQ      1$
4229 021276 000137 042566      JMP      CPSPUR
4230 021302 020027 021542      1$:    CMP      R0,#OPAT21+2      ;FSRC MODE 3?
4231 021306 001425      BEQ      OERR1
4232 021310 000137 042566      JMP      CPSPUR
4233
4234 021314      OERR10:  MOV      (SP),$TMP2      ;WENT TO FSRC
4235 021314 011637 001236      CMP      (SP)+,(SP)+      ;MODE 6 OR 7
4236 021320 022626      1$:    ERROR   +74
4237 021322 104074      BR       ODONE
4238 021324 000522
4239
4240 021326 011637 001240      OERR11:  MOV      (SP),$TMP3      ;WENT TO FSRC MODE
4241 021332 022626      CMP      (SP)+,(SP)+      ;3
4242 021334 012737 000627 001244      MOV      #627,$TMP5
4243 021342 012737 000325 001250      MOV      #325,$TMP7
4244 021350 012737 000323 001246      MOV      #323,$TMP6
4245 021356 104075      1$:    ERROR   +75
4246 021360 000504      BR       ODONE
4247
4248 021362 012737 000322 001246      OERR1:  MOV      #322,$TMP6      ;FSRC MODE2
4249 021370 012737 000627 001242      OERR20:  MOV      #627,$TMP4
4250 021376 012737 000325 001250      MOV      #325,$TMP7
4251 021404 012737 021116 001236      MOV      #02,$TMP2
4252 021412 104076      1$:    ERROR   +76
4253 021414 000466      BR       ODONE
4254 021416 012737 000324 001246      OERR2:  MOV      #324,$TMP6      ;FSRC MODE 4
4255 021424 000761      BR       OERR20
4256 021426 012737 000320 001246      OERR3:  MOV      #320,$TMP6      ;FSRC MODE 0
4257 021434 000755      BR       OERR20
4258 021436 012737 000321 001246      OERR4:  MOV      #321,$TMP6      ;FSRC MODE 1
4259 021444 000751      BR       OERR20
4260
4261 021446 010037 001240      OERR5:  MOV      R0,$TMP3      ;R0 NOT DECREMENTED
4262 021452 012737 021536 001242      MOV      #OPAT20,$TMP
4263 021460 012737 021122 001236      MOV      #04,$TMP2      ;PROPERLY
4264 021466 104077      1$:    ERROR   +77
4265 021470 000440      BR       ODONE

```

```
4266  
4267 021472  
4268 021472 012737 021116 001236  
4269 021500 012737 021520 001240  
4270 021506 012737 021562 001242  
4271 021514 10410C  
4272 021516 000425  
4273  
4274 021520 000000  
4275 021522 000000  
4276 021524 000000  
4277 021526 000000 052525 052525  
4278 021536 021562  
4279 021540 070707  
4280 021542 070707  
4281 021544 070707  
4282 021546 070707 000001  
4283 021552 177777  
4284 021554 177777  
4285 021556 177777  
4286 021560 177777  
4287  
4288 021562 073567  
4289 021564 004210  
4290 021566 114631  
4291 021570 125252  
4292  
4293 021572  
021572 104413
```

```
QERR6: ;DATA FAILURE  
MOV #02,$TMP2  
MOV #ODAT00,$TMP3  
MOV #ODAT10,$TMP4  
1$: ERROR +100  
BR ODONE
```

```
ODAT00: .WORD 0  
ODAT01: .WORD 0  
ODAT02: .WORD 0  
ODAT03: .WORD 0,52525,52525,52525  
OPAT20: .WORD ODAT10  
OPAT21: .WORD 070707  
OPAT22: .WORD 070707  
OPAT23: .WORD 070707  
OPAT24: .WORD 070707,1  
OPAT10: .WORD -1  
OPAT11: .WORD -1  
OPAT12: .WORD -1  
OPAT13: .WORD -1  
ODAT10: .WORD 73567  
ODAT11: .WORD 004210  
ODAT12: .WORD 114631  
ODAT13: .WORD 125252
```

```
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?).
```

4294
4300

4301

```

.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
  
```

4302	021574	000004			
4303	021576				
4304	021576	170011			
4305					
4306	021600	012700	022216		
4307	021604	172410			
4308					
4309	021606	012737	021714	000004	
4310					
4311	021614	012700	021765		
4312					
4313	021620	172460	000241		
4314		021622			
4315					
4316	021624	012701	022236		
4317	021630	174011			
4318	021632	012703	000004		
4319	021636	012702	022226		
4320	021642	012701	022236		
4321	021646	022221			
4322	021650	001007			
4323	021652	077303			
4324	021654	022700	021765		
4325	021660	001401			
4326	021662	000512			
4327	021664	000137	022246		
4328					
4329	021670	012701	022236		
4330	021674	012703	000004		
4331	021700	022721	177777		
4332	021704	001401			
4333	021706	000512			
4334	021710	077305			
4335	021712	000523			
4336					
4337	021714	021627	021622		
4338	021720	001411			
4339	021722	021627	021624		
4340	021726	001402			
4341	021730	000137	042566		
4342					
4343	021734	012737	000327	001246	
4344	021742	000443			
4345	021744	022700	021765		
4346	021750	001004			
4347	021752	012737	000321	001246	
4348	021760	000434			
4349	021762	022700	021775		
4350	021766	001004			

```

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.
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
  
```

```

4351 021770 012737 000322 001246      MOV      #322,$TMP6
4352 021776 000425                BR      PERR17
4353 022000 022700 021767      PERR13: CMP      #PDAT10-241+2,R0      ;WAS IT FSRC MODE 3
4354 022004 001004                BNE     PERR14
4355 022006 012737 000323 001246      MOV      #323,$TMP6
4356 022014 000416                BR      PERR17
4357 022016 022700 021755      PERR14: CMP      #PDAT10-241-10,R0     ;WAS IT FSRC MODE 4
4358 022022 001004                BNE     PERR15
4359 022024 012737 000324 001246      MOV      #324,$TMP6
4360 022032 000407                BR      PERR17
4361 022034 022700 021763      PERR15: CMP      #PDAT10-241-2,R0     ;WAS IT FSRC MODE 5
4362 022040 001401                BEQ     PERR16
4363 022042 000416                BR      PERR20
4364 022044 012737 000325 001246      PERR16: MOV      #325,$TMP6
4365
4366 022052 012737 000627 001244      PERR17: MOV      #627,$TMP5      ;REPORT FSRC
4367 022060 012737 000326 001250      MOV      #326,$TMP7      ;FLOWS FAILURE.
4368 022066 011637 001236      MOV      (SP),$TMP2
4369 022072 022626      CMP      (SP)+,(SP)+
4370 022074 104101      1$:      ERROR   +101
4371 022076 000463                BR      PDONE
4372
4373 022100 011637 001236      PERR20: MOV      (SP),$TMP2      ;REPORT R0 AFFECTED
4374 022104 022626      CMP      (SP)+,(SP)+
4375 022106 000403                BR      PERR22
4376 022110 012737 021620 001236      PERR21: MOV      #P2,$TMP2
4377 022116                PERR22:
4378 022116 010037 001240      MOV      R0,$TMP3
4379 022122 012737 021765 001242      MOV      #PDAT10-241,$TMP4
4380 022130 104102      1$:      ERROR   +102
4381 022132 000445                BR      PDONE
4382
4383 022134                PERR1:      ;DATA FAILURE.
4384 022134 012737 021620 001236      MOV      #P2,$TMP2
4385 022142 012737 022226 001240      MOV      #PDAT10,$TMP3
4386 022150 012737 022236 001242      MOV      #PDAT00,$TMP4
4387 022156 104104      1$:      ERROR   +104
4388 022160 000432                BR      PDONE
4389
4390 022162                PERR2:      ;FSRC FAILURE TO
4391 022162 012737 021620 001236      MOV      #P2,$TMP2      ;MODE 0
4392 022170 012737 000627 001244      MOV      #627,$TMP5
4393 022176 012737 000326 001250      MOV      #326,$TMP7
4394 022204 012737 000320 001246      MOV      #320,$TMP6
4395 022212 104103      1$:      ERROR   +103
4396 022214 000414                BR      PDONE
4397
4398 022216 177777      PPAT10: .WORD   -1
4399 022220 177777      PPAT11: .WORD   -1
4400 022222 177777      PPAT12: .WORD   -1
4401 022224 177777      PPAT13: .WORD   -1
4402
4403 022226 010421      PDAT10: .WORD   010421
4404 022230 031463      PDAT11: .WORD   031463
4405 022232 052525      PDAT12: .WORD   052525
4406 022234 073567      PDAT13: .WORD   073567
4407

```

4408 022236 000000
4409 022240 000000
4410 022242 000000
4411 022244 000000
4412
4413 022246 104413
022246

PDAT00: .WORD 0
PDAT01: .WORD 0
PDAT02: .WORD 0
PDAT03: .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?).

4414
4421

4422

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

```
022250 000004
4423
4424 022252
4425 022252 170011
4426
4427 022254 012700 022706
4428 022260 172410
4429
4430 022262 012737 022414 000004
4431
4432
4433
4434 022270 012700 022455
4435
4436 022274 172470 000241
4437 022276
4438
4439 022300 012701 022726
4440 022304 174011
4441
4442 022306 012703 000004
4443 022312 012704 022726
4444 022316 012705 022736
4445 022322 022425
4446 022324 001007
4447 022326 077303
4448
4449 022330 022700 022455
4450 022334 001401
4451 022336 000514
4452 022340 000137 022746
4453
4454 022344 012701 022726
4455 022350 012703 000004
4456 022354 022721 177777
4457 022360 001002
4458 022362 077304
4459 022364 000513
4460
4461 022366 012701 022716
4462 022372 012702 022726
4463 022376 012703 000004
4464 022402 022122
4465 022404 001401
4466 022406 000524
4467 022410 077304
4468 022412 000504
4469
4470
```

```
TST23: SCOPE
Q1:      SETD
          MOV    #QPAT10,R0
          LDD    (R0),AC0      ;LOAD A DEFAULT
                                ;PATTERN INTO ACO
          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    ACO,(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:      CMF    #-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
```

```

4471
4472 022414 021627 021622      QERR0:  CMP      (SP),#P3
4473 022420 000137 042566      JMP      C PSPUR
4474
4475 022424 022700 022455      QERR11: CMP      #QPAT20-241,R0 ;WAS IT FSRC
4476 022430 001004                BNE      QERR12 ;MODE 1?
4477 022432 012737 000321 001246      MOV      #321,$TMP6
4478 022440 000434                BR       QERR17
4479 022442 022700 022465      QERR12: CMP      #QPAT20-241+10,R0 ;WAS IT FSRC
4480 022446 001004                BNE      QERR13 ;MODE 2?
4481 022450 012737 000322 001246      MOV      #322,$TMP6
4482 022456 000425                BR       QERR17
4483 022460 022700 022457      QERR13: CMP      #QPAT20-241+2,R0 ;WAS IT FSRC
4484 022464 001004                BNE      QERR14 ;MODE 3?
4485 022466 012737 000323 001246      MOV      #323,$TMP6
4486 022474 000416                BR       QERR17
4487 022476 022700 022445      QERR14: CMP      #QPAT20-241-10,R0 ;WAS IT FSRC
4488 022502 001004                BNE      QERR15 ;MODE 4
4489 022504 012737 000324 001246      MOV      #324,$TMP6
4490 022512 000407                BR       QERR17
4491
4492 022514 022700 022453      QERR15: CMP      #QPAT20-241-2,R0 ;WAS IT FSRC
4493 022520 001401                BEQ      QERR16 ;MODE 5
4494 022522 000416                BR       QERR20
4495
4496 022524 012737 000325 001246      QERR16: MOV      #325,$TMP6
4497
4498 022532 012737 000627 001244      QERR17: MOV      #627,$TMP5 ;REPORT FSRC FAILURE
4499 022540 012737 000327 001250      MOV      #327,$TMP7
4500 022546 011637 001236      MOV      (SP),$TMP2
4501 022552 022626      CMP      (SP)+,(SP)+
4502 022554 104105      1$:      ERROR   +105
4503 022556 000473                BR       QDONE
4504
4505 022560 011637 001236      QERR20: MOV      (SP),$TMP2 ;REPORT R0 AFFECTED.
4506 022564 022626      CMP      (SP)+,(SP)+
4507 022566 000403                BR       QERR22
4508 022570 012737 022274 001236      QERR21: MOV      #Q2,$TMP2
4509 022576                QERR22:
4510 022576 010037 001240                MOV      R0,$TMP3
4511 022602 012737 022455 001242      MOV      #QPAT20-241,$TMP4
4512 022610 104106      1$:      ERROR   +106
4513 022612 000455                BR       QDONE
4514
4515 022614 012737 000320 001246      QERR2:  MOV      #320,$TMP6 ;WENT TO FSRC
4516 022622 000403                BR       QERR4 ;MODE 0
4517 022624 012737 000326 001246      QERR3:  MOV      #326,$TMP6 ;WENT TO FSRC
4518                                ;MODE 6
4519 022632 012737 000627 001244      QERR4:  MOV      #627,$TMP5
4520 022640 012737 000327 001250      MOV      #327,$TMP7
4521 022646 012737 022274 001236      MOV      #Q2,$TMP2
4522 022654 104107      1$:      ERROR   +107
4523 022656 000433                BR       QDONE
4524
4525 022660                QERR1:  ;DATA FAILURE
4526 022660 012737 022274 001236      MOV      #Q2,$TMP2
4527 022666 012737 022736 001240      MOV      #QDATIO,$TMP3

```

```
4528 022674 012737 022726 001242      MOV      #QDAT00,$TMP4
4529 022702 104110      1$:      ERROR      +110
4530 022704 000420           BR      QDONE
4531
4532 022706 177777      QPAT10: .WORD      -1
4533 022710 177777      QPAT11: .WORD      -1
4534 022712 177777      QPAT12: .WORD      -1
4535 022714 177777      QPAT13: .WORD      -1
4536
4537 022716 022736      QPAT20: .WORD      QDAT10
4538 022720 052525      QPAT21: .WORD      52525
4539 022722 052525      QPAT22: .WORD      52525
4540 022724 052525      QPAT23: .WORD      52525
4541
4542 022726 000000      QDAT00: .WORD      0
4543 022730 000000      QDAT01: .WORD      0
4544 022732 000000      QDAT02: .WORD      0
4545 022734 000000      QDAT03: .WORD      0
4546
4547 022736 073567      QDAT10: .WORD      073567
4548 022740 052525      QDAT11: .WORD      052525
4549 022742 031463      QDAT12: .WORD      031463
4550 022744 010421      QDAT13: .WORD      010421
4551
4552 022746      QDONE:
      022746 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?).
```

4553

4570

```

.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.
*
*****
    
```

4571	022750	000004					
4572	022752	005037	024074				
4573	022756	012700	024024				
4574	022762	012701	000004				
4575	022766	012720	177777				
4576	022772	077103					
4577	022774	012737	000033	024076			
4578	023002	012737	000023	024100			
4579	023010	012737	023554	000244			
4580	023016						
	023016	012737	023024	001110			
4581	023024	012700	000200				
4582	023030	170100					
4583	023032	012700	024024				
4584	023036	172410					
4585	023040	013737	024076	024102			
4586	023046	012737	000001	024104			
4587	023054	012737	000254	024106			
4588							
4589	023062	012700	024034				
4590	023066	172410					
4591	023070	010037	001252				
4592	023074	012737	023066	001236			
4593							
4594	023102	012704	000204				
4595	023106	170205					
4596							
4597	023110	020405					
4598	023112	001402					
4599	023114	000137	023600				
4600							
4601	023120						
	023120	012737	023126	001110			
4602	023126	012700	000200				
4603	023132	170100					
4604							
4605	023134	012700	024024				
4606	023140	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\$,SLPERR	;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	#U2,\$TMP2	
	MOV	#204,R4	
	STFPS	R5	;SEE IF FPS IS CORRECT
	CMP	R4,R5	
	BEQ	U3	
	JMP	UERR1	
U3:	MOV	#1\$,SLPERR	;SET UP THE LOOP ON ERROR ADDRESS.
1\$:	MOV	#200,R0	
	LDFPS	R0	
	MOV	#UPAT00,R0	;LOAD AC0
	LDD	(R0),AC0	

4607	023142	013737	024100	024102	MOV	UTMP2,UROM1	
4608	023150	012737	000003	024104	MOV	#003,UROM2	
4609	023156	012737	000054	024106	MOV	#054,UROM3	
4610							
4611	023164	012700	024044		MOV	#UPAT20,R0	:LOAD A POSITIVE NUMBER
4612							:INTO ACO
4613	023170	172410			U4: LDD	(R0),ACO	
4614	023172	010037	001252		MOV	R0,\$TMP10	
4615	023176	012737	023170	001236	MOV	#U4,\$TMP2	
4616	023204	012704	000200		MOV	#200,R4	:FPS CORRECT?
4617	023210	170205			STFPS	R5	
4618	023212	020405			CMP	R4,R5	
4619	023214	001402			BEQ	U5	
4620	023216	000137	023664		JMP	UERR2	
4621	023222				U5: MOV	#1\$,SLPERR	:SET UP THE LOOP ON ERROR ADDRESS.
4622	023222	012737	023230	001110	MOV	#200,R0	
4623	023230	012700	000200		1\$: LDFPS	R0	
4624	023234	170100			MOV	#UPAT00,R0	:LOAD ACO
4625	023236	012700	024024		LDD	(R0),ACO	
4626	023242	172410			MOV	UTMP2,UROM1	
4627	023244	013737	024100	024102	MOV	#403,UROM2	
4628	023252	012737	000403	024104	MOV	#056,UROM3	
4629	023260	012737	000056	024106	MOV	#UPAT30,R0	:LOAD A NEGATIVE
4630	023266	012700	024054				:NUMBER INTO ALO
4631	023272	172410			U6: LDD	(R0),ACO	
4632	023274	010037	001252		MOV	R0,\$TMP10	
4633	023300	012737	023272	001236	MOV	#U6,\$TMP2	
4634	023306	012704	000210		MOV	#210,R4	:FPS CORRECT
4635	023312	170205			STFPS	R5	
4636	023314	020405			CMP	R4,R5	
4637	023316	001402			BEQ	U7	
4638	023320	000137	023664		JMP	UERR2	
4639	023324				U7: MOV	#1\$,SLPERR	:SET UP THE LOOP ON ERROR ADDRESS.
4640	023324	012737	023332	001110	MOV	#200,R0	
4641	023332	012700	000200		1\$: LDFPS	R0	
4642	023336	170100			MOV	#UPAT00,R0	:LOAD ACO
4643	023340	012700	024024		LDD	(R0),ACO	
4644	023344	172410			MOV	UTMP1,UROM1	
4645	023346	013737	024076	024102	MOV	#401,UROM2	
4646	023354	012737	000401	024104	MOV	#256,UROM3	
4647	023362	012737	000256	024106	MOV	#UPAT40,R0	:LOAD -0 INTO ACO
4648	023370	012700	024064		U10: LDD	(R0),ACO	
4649	023374	172410			U11: NOP		:TRAP FROM HERE IF
4650	023376	000240			MOV	R0,\$TMP10	
4651	023400	010037	001252		MOV	#U10,\$TMP2	:(BUT FIUV) FAULTS!
4652	023404	012737	023374	001236	MOV	#214,R4	:SEE IF FPS IS CORRECT.
4653	023412	012704	000214		STFPS	R5	
4654	023416	170205			CMP	R4,R5	
4655	023420	020405			BEQ	U12	
4656	023422	001402			JMP	UERR1	
4657	023424	000137	023600		U12: TST	UFLAG	:SEE IF ALL THE PATTERNS
4658	023430	005737	024074		BNE	U14	:HAVE BEEN TEST WITH
4659	023434	001021					:BOTH AC NOT EQUAL TO 0 AND AC=0
4660	023436	012700	024024		MOV	#UPAT00,R0	:IF NOT GO BACK AND
4661	023442	012701	000004		MOV	#4,R1	:CHECK THEM WITH AC=0

```

4662 023446 005020          U13:  CLR      (R0)+
4663 023450 077102          SOB      R1,U13
4664 023452 012737 177777 024074  MOV      #-1,UFLAG
4665 023460 012737 000233 024076  MOV      #233,UTMP1
4666 023466 012737 000223 024100  MOV      #223,UTMP2
4667 023474 000137 023016          JMP      U1
4668 023500          U14:  MOV      #1$,SLPERR          ;SET UP THE LOOP ON ERROR ADDRESS.
      023500 012737 023506 001110  ;NOW SEE IF A TRAP CAN BE FORCED BY SETTING FIUV AND LOADING -0
4669          1$:  MOV      #UERR3,FPVECT
4670 023506 012737 023750 000244  MOV      #4200,R0          ;SET FD AND FIUV
4671 023514 012700 004200          LDFPS   R0
4672 023520 170100          MOV      #UPAT00,R0        ;SET UP ACO
4673 023522 012700 024024          LDD     (R0),AC0
4674 023526 172410          MOV      #UPAT40,R0        ;LOAD -0
4675 023530 012700 024064          LDD     (R0),AC0          ;SHOULD TRAP TO 244
4676 023534 172410          U15:  LDD     (R0),AC0
4677 023536 170000          U16:  CFCC
4678 023540 000240          NOP
4679 023542 012737 023534 001236  MOV      #U15,$TMP2        ;REPORT ERROR.
4680          ;DIDN'T TRAP
4681 023550 104127          1$:  ERROR  +127          ;(BUT FIUV) FAILED.
4682 023552 000556          BR      UDONE
4683
4684          ;TRAPPED TO 244. DID (BUT FIUV) FAIL?
4685 023554 021627 023376  UERR0:  CMP      (SP),#U11
4686 023560 001402          BEQ     1$
4687 023562 000137 042534          JMP     FPSPUR
4688 023566 011637 001236          1$:  MOV      (SP),$TMP2
4689 023572 022626          CMP     (SP)+,(SP)+
4690 023574 104126          2$:  ERROR  +126
4691 023576 000544          BR      UDONE
4692
4693          ;COME HERE TO ANALYZE FPS ERRORS
4694
4695 023600 032705 000004          UERR1:  BIT      #4,R5
4696 023604 001432          BEQ     UERR20
4697 023606 012737 000443 001244  UERR10:  MOV      #443,$TMP5
4698 023614 013703 024106          MOV     UROM3,R3
4699 023620 010337 001250          MOV     R3,$TMP7
4700 023624 032703 000200          BIT     #200,R3
4701 023630 001403          BEQ     1$
4702 023632 042703 000200          BIC     #200,R3
4703 023636 000402          BR      2$
4704 023640 052703 000200          1$:  BIS     #200,R3
4705 023644 010337 001246          2$:  MOV     R3,$TMP6
4706 023650 010537 001240          UERR11:  MOV     R5,$TMP3
4707 023654 010437 001242          MOV     R4,$TMP4
4708 023660 104124          1$:  ERROR  +124
4709 023662 000512          BR      UDONE
4710 023664 032705 000004          UERR2:  BIT      #4,R5
4711 023670 001746          BEQ     UERR10
4712 023672 013737 024102 001244  UERR20:  MOV     UROM1,$TMP5
4713 023700 013703 024104          MOV     UROM2,R3
4714 023704 010337 001250          MOV     R3,$TMP7
4715 023710 032703 000400          BIT     #400,R3
4716 023714 001403          BEQ     1$
4717 023716 042703 000400          BIC     #400,R3
    
```



```

4718 023722 000402
4719 023724 052703 000400
4720 023730 010337 001246
4721 023734 010537 001240
4722 023740 010437 001242
4723 023744 104125
4724 023746 000460
4725
4726
4727 023750 021627 023536
4728 023754 001402
4729 023756 000137 042534
4730 023762 022626
4731 023764 005000
4732 023766 170300
4733 023770 022700 000014
4734 023774 001001
4735 023776 000444
4736 024000 012737 023534 001236
4737 024006 012737 000012 001242
4738 024014 010037 001240
4739 024020 104130
4740 024022 000432
4741 024024 000000
4742 024026 000000
4743 024030 000000
4744 024032 000000
4745
4746 024034 000000
4747 024036 000000
4748 024040 000000
4749 024042 000000
4750
4751 024044 010421
4752 024046 114631
4753 024050 125252
4754 024052 177777
4755
4756 024054 114631
4757 024056 135673
4758 024060 146314
4759 024062 167356
4760
4761 024064 100000
4762 024066 000000
4763 024070 000000
4764 024072 000000
4765
4766 024074 000000
4767 024076 000000
4768 024100 000000
4769 024102 000000
4770 024104 000000
4771 024106 000000
4772 024110
4773
4774

1$: BR 2$
1$: BIS #400,R3
2$: MOV R3,$TMP6
UERR21: MOV R5,$TMP3
MOV R4,$TMP4
1$: ERROR +125
BR UDONE

; INTERRUPT HERE WHEN FIUV SET AND ATTEMPTED TO LOAD-0
UERR3: CMP (SP),#U16
BEQ 1$
JMP FPSPUR
1$: CMP (SP)+,(SP)+
CLR R0
STST R0 ;GET FEC.
CMP #14,R0 ;CORRECT
BNE UERR4
BR UDONE
UERR4: MOV #U15,$TMP2
MOV #12,$TMP4
MOV R0,$TMP3
1$: ERROR +130
BR UDONE

UPAT00: .WORD 0
UPAT01: .WORD 0
UPAT02: .WORD 0
UPAT03: .WORD 0
UPAT10: .WORD 0 ;0
UPAT11: .WORD 0
UPAT12: .WORD 0
UPAT13: .WORD 0
UPAT20: .WORD 010421 ;POS NUM
UPAT21: .WORD 114631
UPAT22: .WORD 125252
UPAT23: .WORD 177777
UPAT30: .WORD 114631 ;NEG NUM
UPAT31: .WORD 135673
UPAT32: .WORD 146314
UPAT33: .WORD 167356
UPAT40: .WORD 100000 ;NEG ZERO
UPAT41: .WORD 0
UPAT42: .WORD 0
UPAT43: .WORD 0
UFLAG: .WORD 0
UTMP1: .WORD 0
UTMP2: .WORD 0
UROM1: .WORD 0
UROM2: .WORD 0
UROM3: .WORD 0
UDONE:
    
```

4782

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

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

:*****

4783	024110	030004			TEST25: SCOPE	
	024112				W1:	
	024112	012737	024120	001110	MOV	#1\$, \$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
4784	024120	012700	000200		1\$:	MOV #200, R0
4785	024124	170100				LDFPS R0 ;SET DOUBLE MODE
4786	024126	012700	024664			MOV #WPAT00, R0 ;LOAD AC0=:
4787	024132	172410				LDD (R0), AC0
4788	024134	012737	024146	001236		MOV #W2, \$TMP2
4789	024142	012700	024664			MOV #WPAT00, R0
4790	024146	172010			W2:	ADDD (R0), AC0 ;TEST INSTRUCTION.
4791	024150	170205				STFPS R5 ;GET FPS
4792	024152	170011				SETD ;SET DOUBLE MODE
4793	024154	012700	024664			MOV #WPAT00, R0
4794	024160	174010				STD AC0, (R0) ;GET THE RESULT
4795	024162	012701	024664			MOV #WPAT00, R1
4796	024166	012702	000004			MOV #4, R2
4797	024172	022021			W3:	CMP (R0)+, (R1)+ ;IS RESULT CORRECT
4798	024174	001405				BEQ W4 ;NO
4799						
4800	024176	004737	024632			JSR PC, WSETUP
4801	024202	104133			1\$:	ERROR +133
4802	024204	000137	024704			JMP WDONE
4803	024210	077210			W4:	SOB R2, W3
4804	024212	022705	000204			CMP #204, R5 ;IS FPS CORRECT
4805	024216	001410				BEQ W5 ;NO
4806						
4807	024220	012737	000204	001242		MOV #204, \$TMP4
4808	024226	010537	001240			MOV R5, \$TMP3
4809	024232	104137			1\$:	ERROR +137
4810	024234	000137	024704			JMP WDONE
4811	024240				W5:	
	024240	012737	024246	001110		MOV #1\$, \$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
4812	024246	012700	000200		1\$:	MOV #200, R0
4813	024252	170100				LDFPS R0 ;SET DOUBLE MODE
4814	024254	012700	024664			MOV #WPAT00, R0 ;LOAD AC0=0
4815	024260	172410				LDD (R0), AC0
4816	024262	012737	024300	001236		MOV #W6, \$TMP2
4817	024270	005000				CLR R0
4818	024272	170100				LDFPS R0 ;GO TO FLOATING MODE
4819	024274	012700	024664			MOV #WPAT00, R0
4820	024300	172010			W6:	ADDF (R0), AC0 ;TEST INSTRUCTION
4821	024302	170205				STFPS R5 ;GET FPS
4822	024304	170011				SETD ;RESET TO DOUBLE MODE
4823	024306	012700	024664			MOV #WPAT00, R0
4824	024312	174010				STD AC0, (R0) ;GET THE RESULT
4825	024314	012701	024664			MOV #WPAT00, R1
4826	024320	012702	000004			MOV #4, R2
4827	024324	022021			W7:	CMP (R0)+, (R1)+ ;WAS THE RESULT

4828	024326	001402			BEQ	W10	:NO. REPORT FAILURE.
4829	024330	104134		1\$:	ERROR	+134	
4830	024332	000564			BR	WDONE	
4831	024334	077205		W10:	SOB	R2,W7	
4832	024336	022705	000004		CMP	#4,R5	:WAS FPS CORRECT
4833	024342	001407			BEQ	W11	:INCORRECT FPS.
4834							
4835	024344	012737	000004	001242	MOV	#4,\$TMP4	
4836	024352	010537	001240		MOV	R5,\$TMP3	
4837	024356	104140		1\$:	ERROR	+140	
4838	024360	000551			BR	WDONE	
4839	024362			W11:			
4840	024370	012737	024370	001110	MOV	#1\$,\$LPERR	:SET UP THE LOOP ON ERROR ADDRESS.
4841	024374	170100	000200	1\$:	MOV	#200,R0	
4842	024376	012700	024664		LDFPS	R0	:SET DOUBLE MODE
4843	024402	172410			MOV	#WPAT00,R0	:LOAD AC0=0
4844	024404	012737	024416	001236	LDD	(R0),AC0	
4845	024412	012700	024664		MOV	#W12,\$TMP2	
4846	024416	173010		W12:	MOV	#WPAT00,R0	
4847	024420	170205			SUBD	(R0),AC0	:TEST INSTRUCTION
4848	024422	170011			STFPS	R5	:GET FPS
4849	024424	012700	024664		SETD		:SET DOUBLE MODE
4850	024430	174010			MOV	#WPAT00,R0	
4851	024432	012701	024664		STD	AC0,(R0)	:GET THE RESULT
4852	024436	012702	000004		MOV	#WPAT00,R1	
4853	024442	022021		W13:	MOV	#4,R2	
4854	024444	001404			CMP	(R0)+,(R1)+	:IS RESULT CORRECT?
4855					BEQ	W14	:NO.
4856	024446	004737	024632		JSR	PC,WSETUP	
4857	024452	104135		1\$:	ERROR	+135	
4858	024454	000513			BR	WDONE	
4859	024456	077207		W14:	SOB	R2,W13	
4860	024460	022705	000204		CMP	#204,R5	:IS FPS CORRECT?
4861	024464	001407			BEQ	W15	:NO.
4862							
4863	024466	012737	000204	001242	MOV	#204,\$TMP4	
4864	024474	010537	001240		MOV	R5,\$TMP3	
4865	024500	104141		1\$:	ERROR	+141	
4866	024502	000500			BR	WDONE	
4867	024504			W15:			
4868	024512	012737	024512	001110	MOV	#1\$,\$LPERR	:SET UP THE LOOP ON ERROR ADDRESS.
4869	024516	170100	000200	1\$:	MOV	#200,R0	
4870	024520	012700	024664		LDFPS	R0	:SET DOUBLE MODE
4871	024524	172410			MOV	#WPAT00,R0	:LOAD AC0=0
4872	024526	012737	024544	001236	LDD	(R0),AC0	
4873	024534	005000			MOV	#W16,\$TMP2	
4874	024536	170100			CLR	R0	
4875	024540	012700	024664		LDFPS	R0	:ENTER FLOATING MODE.
4876	024544	173010		W16:	MOV	#WPAT00,R0	
4877	024546	170205			SUBF	(R0),AC0	:TEST INSTRUCTION.
4878	024550	170011			STFPS	R5	:GET FPS
4879	024552	012700	024664		SETD		:RESET TO DOUBLE MODE
4880	024556	174010			MOV	#WPAT00,R0	:GET THE RESULT.
4881	024560	012701	024664		STD	AC0,(R0)	
4882	024564	012702	000004		MOV	#WPAT00,R1	
					MOV	#4,R2	


```

4883 024570 022021          W17:  CMP      (R0)+,(R1)+  ;IS RESULT CORRECT?
4884 024572 001404          BEQ      W20
4885                                ;NO.
4886 024574 004737 024632          JSR      PC,WSETUP
4887 024600 104136          1$:  ERROR  +136
4888 024602 000440          BR      WDONE
4889 024604 077207          W20:  SOB      R2,W17
4890 024606 022705 000004          CMP      #4,R5  ;IS FPS CORRECT?
4891 024612 001434          BEQ      WDONE
4892                                ;NO
4893 024614 012737 000004 001242          MOV      #4,$TMP4
4894 024622 010537 001240          MOV      R5,$TMP3
4895 024626 104142          1$:  ERROR  +142
4896 024630 000425          BR      WDONE
4897
4898                                ;SET UP FOR ERROR CALL
4899
4900 024632 012737 024664 001240 WSETUP: MOV      #WPAT00,$TMP3
4901 024640 012737 024664 001242          MOV      #WPAT00,$TMP4
4902 024646 012737 024664 001246          MOV      #WPAT00,$TMP6
4903 024654 012737 024664 001244          MOV      #WPAT00,$TMP5
4904 024662 000207          RTS      PC
4905 024664 000000          WPAT00: .WORD 0
4906 024666 000000          WPAT01: .WORD 0
4907 024670 000000          WPAT02: .WORD 0
4908 024672 000000          WPAT03: .WORD 0
4909
4910 024674 000000          WDAPO0: .WORD 0
4911 024676 000000          WDAT01: .WORD 0
4912 024700 000000          WDAT02: .WORD 0
4913 024702 000000          WDAT03: .WORD 0
4914
4915 024704          WDONE:
      024704 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?).
4916
4917
    
```

4924

```

.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:
4925 024706 000004
4925 024710 012737 024716 001110
4926 024716 012700 000200
4927 024722 170100
4928 024724 012700 025470
4929 024730 010037 025456
4930 024734 172410
4931 024736 012737 024750 001236
4932 024744 012700 025500
4933 024750 172010
4934 024752 170205
4935 024754 170011
4936 024756 012700 025460
4937 024762 174010
4938 024764 012701 025470
4939 024770 012702 000004
4940 024774 022021
4941 024776 001401
4942 025000 000561
4943 025002 077204
4944 025004 012704 000200
4945 025010 020405
4946 025012 001402
4947 025014 000137 025426
4948 025020
4949 025020 012737 025026 001110
4950 025026 012700 000200
4951 025032 170100
4952 025034 012700 025510
4953 025040 010037 025456
4954 025044 172410
4955 025046 012737 025060 001236
4956 025054 012700 025500
4957 025060 172010
4958 025062 170205
4959 025064 170011
4960 025066 012700 025460
4961 025072 174010
4962 025074 012701 025510
4963 025100 012702 000004
4964 025104 022021
4965 025106 001401
4966 025110 000515
4967 025112 077204
4968 025114 012704 000210
4969 025120 020405
4970 025122 001401
4970 025124 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,$TMP2
          MOV      #XPAT10,R0    ;FSRC=0
X2:      ADDD    (R0),ACC        ;TEST INSTRUCTION
          STFPS   R5
          SETD
          MOV      #XDAT00,R0    ;GET RESULT.
          STD     ACO,(R0)
          MOV      #XPAT00,R1
          MOV      #4,R2
X3:      CMP     (R0)+,(R1)+    ;IS RESULT CORRECT?
          BEQ     X4
          BR      XERR1
X4:      SOB     R2,X3
          MOV      #200,R4
          CMP     R4,R5          ;IS FPS CORRECT?
          BEQ     X5
          JMP     XERR2
X5:      MOV      #1$, $LPERR    ;SET UP THE LOOP ON ERROR ADDRESS.
1$:      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),ACC        ;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
    
```

```

4971 025126          012737 025134 001110  X11:  MOV    #1$,SLPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
      025126          012700 000200  1$:    MOV    #200,R0
4972 025134          170100          LDFPS  R0              ;SET DOUBLE MODE
4973 025140          012700 025470          MOV    #XPAT00,R0      ;SET ACO TO NON-ZERO
4974 025142          010037 025456          MOV    R0,XTMP        ;POSITIVE NUMBER
4975 025146          172410          LDD    (R0),ACO
4976 025152          012737 025166 001236  MOV    #X12,$TMP2
4977 025154          012700 025500          MOV    #XPAT10,R0     ;FSRC=0
4978 025162          173010          SUBD   (R0),ACO       ;TEST INSTRUCTION
4979 025166          170205          STFPS  R5
4980 025170          170011          SETD
4981 025172          012700 025460          MOV    #XDAT00,R0     ;GET RESULT
4982 025174          174010          STD    ACO,(R0)
4983 025200          012701 025470          MOV    #XPAT00,R1
4984 025202          012702 000004          MOV    #4,R2
4985 025206          022021          X13:  CMP    (R0)+,(R1)+   ;IS RESULT CORRECT?
4986 025212          001401          BEQ    X14
4987 025214          000465          BR     XERR3
4988 025216          077204          X14:  SOB   R2,X13
4989 025220          012704 000200          MOV    #200,R4        ;IS FPS CORRECT?
4990 025222          020405          CMP    R4,R5
4991 025226          001401          BEQ    X15
4992 025230          000503          BR     XERR4
4993 025232
4994 025234          012737 025242 001110  X15:  MOV    #1$,SLPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
      025234          012700 000200  1$:    MOV    #200,R0
4995 025242          170100          LDFPS  R0              ;SET DOUBLE MODE
4996 025246          012700 025510          MOV    #XPAT20,R0     ;SET ACO=A NEGATIVE
4997 025250          010037 025456          MOV    R0,XTMP        ;NUMBER
4998 025254          172410          LDD    (R0),ACO
4999 025260          012737 025274 001236  MOV    #X16,$TMP2
5000 025262          012700 025500          MOV    #XPAT10,R0     ;FSRC=0
5001 025270          173010          SUBD   (R0),ACO       ;TEST INSTRUCTION.
5002 025274          170205          STFPS  R5
5003 025276          170011          SETD
5004 025300          012700 025460          MOV    #XDAT00,R0     ;GET RESULT
5005 025302          174010          STD    ACO,(R0)
5006 025306          012701 025510          MOV    #XPAT20,R1
5007 025310          012702 000004          MOV    #4,R2
5008 025314          022021          X17:  CMP    (R0)+,(R1)+   ;IS RESULT CORRECT?
5009 025320          001401          BEQ    X20
5010 025322          000422          BR     XERR3
5011 025324          077204          X20:  SOB   R2,X17
5012 025326          012704 000210          MOV    #210,R4        ;IS FPS CORRECT?
5013 025330          020405          CMP    R4,R5
5014 025334          001401          BEQ    X21
5015 025336          000440          BR     XERR4
5016 025340          000466          X21:  BR     XDONE
5017 025342
5018
5019          ;REPORT DATA ERRORS
5020
5021 025344          012737 025500 001240  XERR1: MOV    #XPAT10,$TMP3
5022 025352          013737 025456 001242  MOV    XTMP,$TMP4
5023 025360          012737 025460 001244  MOV    #XDAT00,$TMP5
5024 025366          104143          1$:    ERROR  +143
5025 025370          000453          BR     XDONE
    
```



```

5026 025372 012737 025500 001240 XERR3: MOV #XPAT10,$TMP3
5027 025400 013737 025456 001242 MOV $TMP,$TMP4
5028 025406 012737 025460 001244 MOV #XDAT00,$TMP5
5029 025414 013737 025456 001246 MOV $TMP,$TMP6
5030 025422 104144 1$: ERROR +144
5031 025424 000435 BR XDONE
5032
5033 ;REPORT FPS ERRORS
5034
5035 025426 XERR2:
5036 025426 010537 001240 MOV R5,$TMP3
5037 025432 010437 001242 MOV R4,$TMP4
5038 025436 104145 1$: ERROR +145
5039 025440 000427 BR XDONE
5040 025442 XERR4:
5041 025442 010537 001240 MOV R5,$TMP3
5042 025446 010437 001242 MOV R4,$TMP4
5043 025452 104146 1$: ERROR +146
5044 025454 000421 BR XDONE
5045 025456 000000 XTMP: .WORD 0
5046 025460 000000 XDAT00: .WORD 0
5047 025462 000000 XDAT01: .WORD 0
5048 025464 000000 XDAT02: .WORD 0
5049 025466 000000 XDAT03: .WORD 0
5050
5051 025470 010421 XPAT00: .WORD 010421
5052 025472 021042 XPAT01: .WORD 021042
5053 025474 031463 XPAT02: .WORD 031463
5054 025476 042104 XPAT03: .WORD 042104
5055
5056 025500 000000 XPAT10: .WORD 0
5057 025502 000000 XPAT11: .WORD 0
5058 025504 000000 XPAT12: .WORD 0
5059 025506 000000 XPAT13: .WORD 0
5060 025510 104210 XPAT20: .WORD 104210
5061 025512 114631 XPAT21: .WORD 114631
5062 025514 125252 XPAT22: .WORD 125252
5063 025516 135673 XPAT23: .WORD 135673
5064
5065 025520 XDONE:
    025520 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?).
    
```

5066

5074

```

.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.
:
:*****
    
```

```

5075 025522 030004
5075 025524 005037 026060
5076 025530 012737 026100 026062
5077 025536 012737 026110 026064
5078 025544 012737 000210 026066
5079 025552
5080 025552 012737 025560 001110
5080 025560 012700 000200
5081 025564 170100
5082 025566 012700 026120
5083 025572 172410
5084 025574 013700 026062
5085 025600 173010
5086 025602 170205
5087 025604 170011
5088 025606 012700 026070
5089 025612 174010
5090 025614 012702 000004
5091 025620 013701 026064
5092 025624 022021
5093 025626 001026
5094 025630 077203
5095 025632 023705 026066
5096 025636 001401
5097 025640 000475
5098 025642 005737 026060
5099 025646 001015
5100 025650 012737 177777 026060
5101 025656 012737 026110 026062
5102 025664 012737 026100 026064
5103 025672 012737 000200 026066
5104 025700 000724
5105 025702 000512
5106 025704 012702 000004
5107 025710 012700 026062
5108 025714 012701 026070
5109 025720 022021
5110 025722 001002
5111 025724 077203
5112 025726 000421
5113 025730
5114 025730 012737 025600 001236
5115 025736 013737 026062 001240
5116 025744 012737 026120 001242
5117 025752 012737 026070 001244
5118 025760 013737 026064 001246
5119 025766 104147
5120 025770 000457

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

```

5121 025772
5122 025772 012737 025600 001236
5123 026000 013737 026062 001240
5124 026006 012737 026120 001242
5125 026014 012737 026070 001244
5126 026022 013737 026064 001246
5127 026030 104150
5128 026032 000436
5129 026034
5130 026034 012737 025600 001236
5131 026042 010537 001240
5132 026046 013737 026066 001242
5133 026054 104151
5134 026056 000424
5135
5136 026060 000000
5137 026062 000000
5138 026064 000000
5139 026066 000000
5140
5141 026070 000000
5142 026072 000000
5143 026074 000000
5144 026076 000000
5145
5146 026100 063146
5147 026102 052525
5148 026104 042104
5149 026106 167356
5150
5151 026110 163146
5152 026112 052525
5153 026114 042104
5154 026116 167356
5155
5156 026120 000000
5157 026122 000000
5158 026124 000000
5159 026126 000000
5160
5161 026130
    026130 104413

VERR2:
MOV #Y2,$TMP2
MOV YTMP1,$TMP3
MOV #YPAT20,$TMP4
MOV #YDAT00,$TMP5
MOV YTMP2,$TMP6
1$: ERROR +150
BR YDONE
: XOR OF SIGN BIT
: FAILED

VERR3:
MOV #Y2,$TMP2
MOV R5,$TMP3
MOV YTMP3,$TMP4
1$: ERROR +151
BR YDONE
: FPS WRONG.

YFLAG: .WORD 0
YTMP1: .WORD 0
YTMP2: .WORD 0
YTMP3: .WORD 0

YDAT00: .WORD 0
YDAT01: .WORD 0
YDAT02: .WORD 0
YDAT03: .WORD 0

YPAT00: .WORD 063146
YPAT01: .WORD 052525
YPAT02: .WORD 042104
YPAT03: .WORD 167356

YPAT10: .WORD 163146
YPAT11: .WORD 052525
YPAT12: .WORD 042104
YPAT13: .WORD 167356

YPAT20: .WORD 0
YPAT21: .WORD 0
YPAT22: .WORD 0
YPAT23: .WORD 0

YDONE:
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?).
    
```


5170

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

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

:*****

5171	026132	000004			TST30:	SCOPE		
5172	026134	005037	026370			CLR	ZFLAG	
5173	026140	012737	026406	026372		MOV	#ZPAT00,ZTMP1	:P
5174	026146	012737	000200	026374		MOV	#200,ZTMP2	
	026154				Z1:			
5175	026154	012737	026162	001110		MOV	#1\$, \$LPERR	:SET UP THE LOOP ON ERROR ADDRESS.
5176	026162	012700	000200		1\$:	MOV	#200,R0	
5177	026166	170100				LDFPS	R0	:SET DOUBLE MODE
5178	026170	012700	026426			MOV	#ZPAT20,R0	:SET AC0=0
5179	026174	172410				LDD	(R0),AC0	
5180	026176	013700	026372		Z2:	MOV	ZTMP1,R0	
5181	026202	172010				ADD	(R0),AC0	:TEST INSTRUCTION
5182	026204	170205				STFPS	R5	
5183	026206	170011				SETD		
5184	026210	012700	026376			MOV	#ZDAT00,R0	:GET RESULT
5185	026214	174010				STD	AC0,(R0)	
5186	026216	012702	000004			MOV	#4,R2	
5187	026222	013701	026372		Z3:	MOV	ZTMP1,R1	:RESULT CORRECT?
5188	026226	022021				CMP	(R0)+,(R1)+	
5189	026230	001401				BEQ	Z4	
5190	026232	000423			Z4:	BR	ZERR1	
5191	026234	077204				SOB	R2,Z3	
5192	026236	023705	026374			CMP	ZTMP2,R5	:FPS CORRECT?
5193	026242	001401				BEQ	Z5	
5194	026244	000437				BR	ZERR2	
5195	026246	005737	026370		Z5:	TST	ZFLAG	:FINISHED TEST?
5196	026252	001012				BNE	Z6	
5197	026254	012737	177777	026370		MOV	#-1,ZFLAG	
5198	026262	012737	026416	026372		MOV	#ZPAT10,ZTMP1	
5199	026270	012737	000210	026374		MOV	#210,ZTMP2	
5200	026276	000726				BR	Z1	
5201	026300	000456			Z6:	BR	ZDONE	
5202	026302				ZERR1:			:DATA FAILURE
5203	026302	012737	026202	001236		MOV	#Z2,\$TMP2	
5204	026302	013737	026372	001240		MOV	ZTMP1,\$TMP3	
5205	026310	012737	026426	001242		MOV	#ZPAT20,\$TMP4	
5206	026316	012737	026376	001244		MOV	#ZDAT00,\$TMP5	
5207	026324	012737	026372	001246		MOV	ZTMP1,\$TMP6	
5208	026332	013737			1\$:	ERROR	+152	
5209	026340	104152				BR	ZDONE	
5210	026342	000435			ZERR2:			
5211	026344	012737	026202	001236		MOV	#Z2,\$TMP2	
5212	026344	010537	001240			MOV	R5,\$TMP3	
5213	026352	013737	026374	001242		MOV	ZTMP2,\$TMP4	
5214	026356	013737			1\$:	ERROR	+153	
5215	026364	104153				BR	ZDONE	
5216	026366	000423						

5216	026370	000000	ZFLAG:	.WORD	0
5217	026372	000000	ZTMP1:	.WORD	0
5218	026374	000000	ZTMP2:	.WORD	0
5219					
5220	026376	000000	ZDAT00:	.WORD	0
5221	026400	000000	ZDAT01:	.WORD	0
5222	026402	000000	ZDAT02:	.WORD	0
5223	026404	000000	ZDAT03:	.WORD	0
5224					
5225	026406	031463	ZPAT00:	.WORD	031463
5226	026410	010421	ZPAT01:	.WORD	010421
5227	026412	146314	ZPAT02:	.WORD	146314
5228	026414	156735	ZPAT03:	.WORD	156735
5229					
5230	026416	156735	ZPAT10:	.WORD	156735
5231	026420	167356	ZPAT11:	.WORD	167356
5232	026422	135673	ZPAT12:	.WORD	135673
5233	026424	146314	ZPAT13:	.WORD	146314
5234					
5235	026426	000000	ZPAT20:	.WORD	0
5236	026430	000000	ZPAT21:	.WORD	0
5237	026432	000000	ZPAT22:	.WORD	0
5238	026434	000000	ZPAT23:	.WORD	0
5239					
5240	026436		ZDONE:		
	026436	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?).

5241
5242

5250

.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.
 *


```

5251 026440 030004
5251 026442 012737 026450 001110
5252 026450 012700 003240
5253 026454 170100
5254 026456 012737 027036 000244
5255 026464 012700 027414
5256
5257 026470 172410
5258 026472 012737 026504 001236
5259 026500 012700 027424
5260 026504 172010
5261
5262 026506 012700 027404
5263 026512 174010
5264 026514 012701 027434
5265 026520 012702 000004
5266 026524 022021
5267 026526 001414
5268 026530 012700 027444
5269 026534 012701 027404
5270 026540 012702 000004
5271 026544 022021
5272 026546 001401
5273 026550 000565
5274 026552 077204
5275 026554 000137 027160
5276 026560 077217
5277
5278
5279
5280 026562
5280 026562 012737 026570 001110
5281 026570 012700 003200
5282 026574 170100
5283 026576 012700 027414
5284 026602 172410
5285 026604 012737 026616 001236
5286 026612 012700 027424
5287 026616 172010
5288
5289 026620 012700 027404
5290 026624 174010
5291 026626 012701 027444
5292 026632 012702 000004
5293 026636 022021
5294 026640 001425
5295 026642 012700 027434
    
```

TEST31: SCOPE
 AA1: MOV #1\$, \$LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
 1\$: MOV #3240, R0 ;SET FIV FIV FD AND FT
 LDFPS R0 ;IN CASE THE OVER/UNDER
 MOV #AAERRO, FPVECT ;FLOWS IN TRAP WILL
 MOV #AAPATO, R0 ;OCCUR
 LDD (R0), ACO ;SET UP ACO
 MOV #AA2, \$TMP2 ;OPERAND
 MOV #AAPAT1, R0
 AA2: ADDD (R0), ACO ;TEST INSTRUCTION
 ;SHOULD TRUNCATE
 AA3: MOV #AADATO, R0 ;GET THE RESULT
 STD ACO, (R0)
 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
 AA6: 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.
 1\$: 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 ;GET THE RESULT
 STD ACO, (R0)
 MOV #AAPAT3, R1
 MOV #4, R2
 AA13: CMP (R0)+, (R1)+ ;CORRECT?
 BEQ AA20
 MOV #AAPAT2, R0 ;DID (BUT FT) FAIL?


```

5296 026646 012701 027404      MOV      #AADATO,R1
5297 026652 012702 000004      MOV      #4,R2
5298 026656 022021      AA14:   CMP      (R0)+,(R1)+
5299 026660 001413      BEQ
5300 026662 012700 027454      MOV      #AAPAT4,R0      ;WAS THE FLOATING
5301 026666 012701 027404      MOV      #AADATO,R1      ;CONSTANT USED
5302 026672 012702 000004      MOV      #4,R2      ;INSTEAD OF THE
5303 026676 022021      AA15:   CMP      (R0)+,(R1)+      ;DOUBLE CONSTANT
5304 026700 001401      BEQ      AA16      ;IN THE ROUND
5305 026702 000544      BR      AAERR3      ;FLOWS?
5306 026704 077204      AA16:   SOB      R2,AA15      ;DATA ERROR
5307 026706 000546      BR      AAERR4      ;CONSTANT ERROR
5308 026710 077216      AA17:   SOB      R2,AA14
5309 026712 000562      BR      AAERR5      ;(BUT FT) ERROR
5310 026714 077230      AA20:   SOB      R2,AA13
5311
5312      ;NOW TEST ADDF WITH FT=0, ROUND MODE
5313
5314 026716      AA21:
5315 026716 012737 026724 001110      1$:   MOV      #1$,SLPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
5316 026724 012700 003200      MOV      #3200,R0      ;FIV=1, FIV=1, FT=0
5317 026730 170100      LDFPS   RC
5318 026732 012700 027414      MOV      #AAPATO,R0      ;LOAD ACO OPERAND
5319 026736 172410      LDD      (R0),AC0
5320 026740 170001      SETF
5321 026742 012737 026754 001236      MOV      #AA22,$TMP2      ;ENTER FLOATING MODE
5322 026750 012700 027464      MOV      #AAPAT5,R0
5323 026754 172010      AA22:   ADDF      (R0),AC0      ;TEST INSTRUCTION
5324 026756      AA23:   ;SHOULD ROUND
5325 026756 170011      SETD
5326      ;RESET TO DOUBLE
5327 026760 012700 027404      MOV      #AADATO,R0      ;MODE
5328 026764 174010      STD      AC0,(R0)      ;GET THE RESULT
5329 026766 012701 027474      MOV      #AAPAT6,R1      ;CORRECT?
5330 026772 012702 000002      MOV      #2,R2
5331 026776 022021      AA24:   CMP      (R0)+,(R1)+
5332 027000 001413      BEQ      AA27
5333 027002 012700 027434      MOV      #AAPAT2,R0      ;WAS THE DOUBLE
5334 027006 012701 027404      MOV      #AADATO,R1      ;CONSTANT USED INSTEAD
5335 027012 012702 000002      MOV      #2,R2      ;OF THE FLOATING
5336 027016 022011      AA25:   CMP      (R0)+,(R1)      ;CONSTANT IN THE
5337 027020 001401      BEQ      AA26      ;ROUND FLOWS?
5338 027022 000534      BR      AAERR6      ;DATA ERROR
5339 027024 077204      AA26:   SOB      R2,AA25
5340 027026 000550      BR      AAERR7      ;CONSTANT ERROR
5341 027030 077216      AA27:   SOB      R2,AA24
5342 027032 000137 027504      JMP      AADONE
5343
5344      ;COME HERE IF A TRAP OCCURS TO 244.
5345
5346 027036 013700 001236      AAERR0: MOV      $TMP2,R0      ;SEE IF THE TRAP WAS
5347 027042 005720      TST      (R0)+      ;AT A TEST INSTRUCTION
5348 027044 020016      CMP      R0,(SP)
5349 027046 001402      BEQ      1$
5350 027050 000137 042534      10$:   JMP      FPSPUR
5351 027054      1$:
    
```

```

5352 027054 170300          STST  R0          ;GET FEC
5353 027056 020027 000010  CMP   R0,#10
5354 027062 001405          BEQ   20$         ;OVERFLOW
5355 027064 020027 000012  CMP   R0,#12
5356 027070 001410          BEQ   30$         ;UNDERFLOW
5357 027072 000766          BR    10$
5358 027074 027076          20$:
5359 027076 011637 001236  MOV   (SP), $TMP2 ;REPORT OVERFLOW ERROR
5360 027102 022626          CMP   (SP)+, (SP)+
5361 027104 104154          21$: ERROR +154
5362 027106 000137 027504  25$: JMP   AADONE
5363 027112 011637 001236  30$: MOV   (SP), $TMP2 ;REPORT UNDERFLOW
5364 027116 022626          CMP   (SP)+, (SP)+ ;ERROR
5365 027120 104155          31$: ERROR +155
5366 027122 000771          BR    25$
5367
5368          ;ADD RESULT INCORRECT
5369 027124 012737 027434 001246 AAERR1: MOV   #AAPAT2, $TMP6
5370 027132 012737 027414 001242 AAERR10: MOV  #AAPAT0, $TMP4
5371 027140 012737 027424 001240      MOV  #AAPAT1, $TMP3
5372 027146 012737 027404 001244      MOV  #AADATO, $TMP5
5373 027154 104162          1$: ERROR +162
5374 027156 000552          BR    AADONE
5375 027160 012737 027434 001246 AAERR2: MOV   #AAPAT2, $TMP6 ;(BUT FT) FAILED.
5376 027166 012737 027414 001242      MOV  #AAPAT0, $TMP4
5377 027174 012737 027424 001240      MOV  #AAPAT1, $TMP3
5378 027202 012737 027404 001244      MOV  #AADATO, $TMP5
5379 027210 104156          1$: ERROR +156
5380 027212 000534          BR    AADONE
5381 027214 012737 027444 001246 AAERR3: MOV   #AAPAT3, $TMP6 ;DATA ERROR.
5382 027222 000743          BR    AAERR10
5383 027224 012737 027444 001246 AAERR4: MOV   #AAPAT3, $TMP6 ;BAD CONSTANT
5384 027232 012737 027414 001242      MOV  #AAPAT0, $TMP4
5385 027240 012737 027424 001240      MOV  #AAPAT1, $TMP3
5386 027246 012737 027404 001244      MOV  #AADATO, $TMP5
5387 027254 104160          1$: ERROR +160
5388 027256 000512          BR    AADONE
5389 027260 012737 027444 001246 AAERR5: MOV   #AAPAT3, $TMP6 ;(BUT FT) FAILED.
5390 027266 012737 027414 001242      MOV  #AAPAT0, $TMP4
5391 027274 012737 027424 001240      MOV  #AAPAT1, $TMP3
5392 027302 012737 027404 001244      MOV  #AADATO, $TMP5
5393 027310 104157          1$: ERROR +157
5394 027312 000474          BR    AADONE
5395 027314 012737 027464 001240 AAERR6: MOV   #AAPAT5, $TMP3 ;FD=0 AND
5396 027322 012737 027414 001242      MOV  #AAPAT0, $TMP4 ;DATA ERROR
5397 027330 012737 027404 001244      MOV  #AADATO, $TMP5
5398 027336 012737 027474 001246      MOV  #AAPAT6, $TMP6
5399 027344 104160          1$: ERROR +160
5400 027346 000456          BR    AADONE
5401 027350 012737 027464 001240 AAERR7: MOV   #AAPAT5, $TMP3 ;CONSTANT ERROR
5402 027356 012737 027414 001242      MOV  #AAPAT0, $TMP4
5403 027364 012737 027404 001244      MOV  #AADATO, $TMP5
5404 027372 012737 027474 001246      MOV  #AAPAT6, $TMP6
5405 027400 104161          1$: ERROR +161
5406 027402 000440          BR    AADONE
5407 027404 000000 000000 000000 AADATO: .WORD 0,0,0,0
5408 027414 000200 000000 000000 AAPATO: .WORD 200,0,0,0
    
```

5409	027424	000200	000000	000000	AAPAT1: .WORD	200,0,0,1
5410	027434	000400	000000	000000	AAPAT2: .WORD	400,0,0,0
5411	027444	000400	000000	000000	AAPAT3: .WORD	400,0,0,1
5412	027454	000400	000000	100000	AAPAT4: .WORD	400,0,100000,0
5413	027464	000200	000001	000000	AAPAT5: .WORD	200,1,0,0
5414	027474	000400	000001	000000	AAPAT6: .WORD	400,1,0,0
5415	027504				AADONE:	
	027504	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?).

5426

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

```
TST32: SCOPE
:EXPONENT DIFFERENCE=57=71 (OCT) FD=1
5427 027506 000004
5428 027510
027510 012737 027516 001110 CC1:
5429 027516 012704 003200 1$: MOV #1$, $LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
5430 027522 170104 LDFPS #3200,R4 ;SET FIV,FIV, AND FD
5431 027524 012737 027544 001236 MOV #CC2,$TMP2
5432 027532 012700 031172 MOV #CCP0,R0 ;SET ACO OPERAND
5433 027536 172410 LDD (R0),AC0 ;ACO
5434 027540 012700 031212 MOV #CCP2,R0
5435 027544 172010 CC2: ADDD (R0),AC0 ;TEST INSTRUCTION
5436 027546 170205 STFPS R5 ;GET FPS
5437 027550 012700 031162 MOV #CCDAT0,R0 ;GET THE RESULT
5438 027554 174010 STD AC0,(R0)
5439 027556 012701 031212 MOV #CCP2,R1 ;IS IT CORRECT
5440 027562 012702 000004 MOV #4,R2
5441 027566 022021 CC3: CMP (R0)+,(R1)+
5442 027570 001415 BEQ CC6
5443 027572 012700 031162 MOV #CCDAT0,R0 ;DID A BAD
5444 027576 012701 031172 MOV #CCP0,R1 ;CONSTANT (NOT 57)
5445 027602 012702 000004 MOV #4,R2 ;GET GENERATED
5446 027606 022021 CC4: CMP (R0)+,(R1)+ ;FOR THE ALIGNMENT
5447 027610 001402 BEQ CC5 ;FLOWS?
5448 027612 000137 030560 JMP CCER1 ;DATA ERROR.D
5449 027616 077205 CC5: SOB R2,CC4
5450 027620 000137 030616 JMP CCER2 ;BAD CONSTANT.D
5451 027624 077220 CC6: SOB R2,CC3
5452 027626 020405 CMP R4,R5 ;FPS CORRECT?
5453 027630 001402 BEQ CC7
5454 027632 000137 030524 JMP CCER0 ;BAD FPS.
5455
:EXPONENT DIFFERENCE=56=70 (OCT) FD=1
5456 027636
027636 012737 027644 001110 CC7:
5457 027644 012704 003200 1$: MOV #1$, $LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
5458 027650 170104 LDFPS #3200,R4 ;SET FIV,FIV, AND FD
5459 027652 012737 027672 001236 MOV #CC8,$TMP2
5460 027660 012700 031172 MOV #CCP0,R0 ;SET ACO OPERAND
5461 027664 172410 LDD (R0),AC0
5462 027666 012700 031202 MOV #CCP1,R0 ;FSRC
5463 027672 172010 CC8: ADDD (R0),AC0 ;TEST INSTRUCTION
5464 027674 170205 STFPS R5 ;GET FPS
5465 027676 012700 031162 MOV #CCDAT0,R0 ;GET THE RESULT
5466 027702 174010 STD AC0,(R0)
5467 027704 012701 031262 MOV #CCP7,R1 ;IS IT CORRECT
5468 027710 012702 000004 MOV #4,R2
```

```

5469 027714 022021          CC9:  CMP      (R0)+,(R1)+
5470 027716 001415          BEQ      CC12
5471 027720 012700 031162      MOV      #CCDAT0,R0      ;DID A BAD
5472 027724 012701 031202      MOV      #CCP1,R1      ;CONSTANT (NOT 57)
5473 027730 012702 000004      MOV      #4,R2          ;GET GENERATED
5474 027734 022021          CC10: CMP      (R0)+,(R1)+   ;FOR THE ALIGNMENT
5475 027736 001402          BEQ      CC11           ;FLOWS?
5476 027740 000137 030654      JMP      CCER3          ;DATA ERROR.D
5477 027744 077205          CC11: SOB      R2,CC10
5478 027746 000137 030672      JMP      CCER4          ;BAD CONSTANT.D
5479 027752 077220          CC12: SOB      R2,CC9
5480 027754 020405          CMP      R4,R5          ;FPS CORRECT?
5481 027756 001402          BEQ      CC13
5482 027760 000137 030524      JMP      CCER0          ;BAD FPS.
5483
5484 027764          ;EXPONENT DIFFERENCE=25=31 (OCT) FD=0
027764 012737 027772 001110      CC13:  MOV      #1$, $LPERR   ;SET UP THE LOOP ON ERROR ADDRESS.
027764 012737 030020 001236      1$:   MOV      #CC14,$TMP2
5485 027772 012737 030020 001236      MOV      #CCP0,R0      ;SET UP ACO OPERAND.
5486 030000 012700 031172      MOV      (R0),ACO
5487 030004 172410          LDD      (R0),ACO
5488 030006 012704 003000      MOV      #3000,R4      ;SET FIV,FIV. CLEAR FD.
5489 030012 170104          LDFPS   R4
5490 030014 012700 031252      MOV      #CCP6,R0      ;FSRC
5491 030020 172010          CC14: ADDF     (R0),ACO   ;TEST INSTRUCTION
5492 030022 170205          STFPS   R5
5493 030024 170011          SETD
5494 030026 012700 031162      MOV      #CCDAT0,R0   ;REENTER DOUBLE MOVE
5495 030032 174010          STD     ACO,(R0)      ;GET THE RESULT
5496 030034 012701 031252      MOV      #CCP6,R1
5497 030040 012702 000002      MOV      #2,R2        ;IS THE RESULT CORRECT?
5498 030044 022021          CC15: CMP      (R0)+,(R1)+
5499 030046 001415          BEQ      CC18
5500 030050 012700 031162      MOV      #CCDAT0,R0   ;WAS A BAD CONSTANT
5501 030054 012701 031222      MOV      #CCP3,R1     ;USED (NOT 25) IN
5502 030060 012702 000002      MOV      #2,R2        ;THE ALIGN FLOWS?
5503 030064 022021          CC16: CMP      (R0)+,(R1)+
5504 030066 001402          BEQ      CC17
5505 030070 000137 030730      JMP      CCER5          ;DATA ERROR F
5506 030074 077205          CC17: SOB      R2,CC16
5507 030076 000137 030764      JMP      CCER6          ;BAD CONSTANT F
5508 030102 077220          CC18: SOB      R2,CC15
5509 030104 020405          CMP      R4,R5
5510 030106 001402          BEQ      CC19
5511 030110 000137 030542      JMP      CCER90        ;BAD FPS.
5512
5513 030114          ;EXPONENT DIFFERENCE=24=30 (OCT) FD=0
030114 012737 030122 001110      CC19:  MOV      #1$, $LPERR   ;SET UP THE LOOP ON ERROR ADDRESS.
030114 012737 030150 001236      1$:   MOV      #CC20,$TMP2
5514 030122 012737 030150 001236      MOV      #CCP3,R0      ;SET UP ACO OPERAND.
5515 030130 012700 031222      MOV      (R0),ACO
5516 030134 172410          LDD      (R0),ACO
5517 030136 012704 003000      MOV      #3000,R4      ;SET FIV,FIV. CLEAR FD.
5518 030142 170104          LDFPS   R4
5519 030144 012700 031242      MOV      #CCP5,R0      ;FSRC
5520 030150 172010          CC20: ADDF     (R0),ACO   ;TEST INSTRUCTION
5521 030152 170205          STFPS   R5
5522 030154 170011          SETD
5523 030156 012700 031162      MOV      #CCDAT0,R0   ;REENTER DOUBLE MOVE
                    ;GET THE RESULT
    
```



```

5524 030162 174010          STD      ACO,(R0)
5525 030164 012701 031272    MOV      #CCP10,R1      :IS THE RESULT CORRECT?
5526 030170 012702 000002    MOV      #2,R2
5527 030174 022021          CC21:   CMP      (R0)+,(R1)+
5528 030176 001415          BEQ      CC24
5529 030200 012700 031162    MOV      #CCDATO,R0    :WAS A BAD CONSTANT
5530 030204 012701 031242    MOV      #CCP5,R1      :USED (NOT 25) IN
5531 030210 012702 000002    MOV      #2,R2          :THE ALIGN FLOWS?
5532 030214 022021          CC22:   CMP      (R0)+,(R1)+
5533 030216 001402          BEQ      CC23
5534 030220 000137 031020    JMP      CCER7          :DATA ERROR F
5535 030224 077205          CC23:   SOB      R2,CC22
5536 030226 000137 031036    JMP      CCER8          :BAD CONSTANT F
5537 030232 077220          CC24:   SOB      R2,CC21
5538 030234 020405          CMP      R4,R5
5539 030236 001402          BEQ      CC25
5540 030240 000137 030542    JMP      CCER90         :BAD FPS.
5541
5542 030244          :EXPONENT DIFFERENCE=1 FD=1
5543 030244 012737 030252 001110  CC25:   MOV      #1$,SLPERR    :SET UP THE LOOP ON ERROR ADDRESS.
5544 030252 012704 003200          1$:   MOV      #3200,R4      :SET FIV,FIV, AND FD
5545 030256 170104          LDFPS   R4
5546 030260 012737 030300 001236    MOV      #CC26,$TMP2
5547 030266 012700 031172          MOV      #CCP0,R0      :SET ACO OPERAND
5548 030272 172410          LDD     (R0),ACO
5549 030274 012700 031222          MOV      #CCP3,R0      :FSRC
5550 030300 172010          CC26:   ADDD   (R0),ACO     :TEST INSTRUCTION
5551 030302 170205          STFPS  R5              :GET FPS
5552 030304 012700 031162    MOV      #CCDATO,R0    :GET THE RESULT
5553 030310 174010          STD     ACO,(R0)
5554 030312 012701 031302    MOV      #CCP11,R1     :IS IT CORRECT
5555 030316 012702 000004    MOV      #4,R2
5556 030322 022021          CC27:   CMP      (R0)+,(R1)+
5557 030324 001415          BEQ      CC30
5558 030326 012700 031162    MOV      #CCDATO,R0    :DID A BAD
5559 030332 012701 031222    MOV      #CCP3,R1      :CONSTANT (NOT 57)
5560 030336 012702 000004    MOV      #4,R2          :GET GENERATED
5561 030342 022021          CC28:   CMP      (R0)+,(R1)+  :FOR THE ALIGNMENT
5562 030344 001402          BEQ      CC29          :FLOWS?
5563 030346 000137 031072    JMP      CCER10        :DATA ERROR.D
5564 030352 077205          CC29:   SOB      R2,CC28
5565 030354 000137 031110    JMP      CCER11        :BAD CONSTANT.D
5566 030360 077220          CC30:   SOB      R2,CC27
5567 030362 020405          CMP      R4,R5          :FPS CORRECT?
5568 030364 001402          BEQ      CC31
5569 030366 000137 030524    JMP      CCER0          :BAD FPS.
5570 030372          :EXPONENT DIFFERENCE=100=144 (OCT) FD=1
5571 030372 012737 030400 001110  CC31:   MOV      #1$,SLPERR    :SET UP THE LOOP ON ERROR ADDRESS.
5572 030400 012704 003200          1$:   MOV      #3200,R4      :SET FIV,FIV, AND FD
5573 030404 170104          LDFPS  R4
5574 030406 012737 030426 001236    MOV      #CC32,$TMP2
5575 030414 012700 031172          MOV      #CCP0,R0      :SET ACO OPERAND
5576 030420 172410          LDD     (R0),ACO
5577 030422 012700 031232          MOV      #CCP4,R0      :FSRC
5578 030426 172010          CC32:   ADDD   (R0),ACO     :TEST INSTRUCTION
5579 030430 170205          STFPS  R5              :GET FPS
    
```


5579	030432	012700	031162		MOV	#CCDATO,R0	:GET THE RESULT	
5580	030436	174010			STD	AC0,(R0)		
5581	030440	012701	031232		MOV	#CCP4,R1	:IS IT CORRECT	
5582	030444	012702	000004		MOV	#4,R2		
5583	030450	022021		CC33:	CMP	(R0)+,(R1)+		
5584	030452	001415			BEQ	CC36		
5585	030454	012700	031162		MOV	#CCDATO,R0	:DID A BAD	
5586	030460	012701	031232		MOV	#CCP4,R1	:CONSTANT (NOT 57)	
5587	030464	012702	000004		MOV	#4,R2	:GET GENERATED	
5588	030470	022021		CC34:	CMP	(R0)+,(R1)+	:FOR THE ALIGNMENT	
5589	030472	001402			BEQ	CC35	:FLOWS?	
5590	030474	000137	031126		JMP	CCER12	:DATA ERROR.D	
5591	030500	077205		CC35:	SOB	R2,CC34		
5592	030502	000137	031144		JMP	CCER13	:BAD CONSTANT.D	
5593	030506	077220		CC36:	SOB	R2,CC33		
5594	030510	020405			CMP	R4,R5	:FPS CORRECT?	
5595	030512	001402			BEQ	CC37		
5596	030514	000137	030524		JMP	CCER0	:BAD FPS.	
5597	030520	000137	031322	CC37:	JMP	CCDONE		
5598	030524	010437	001242	CCER0:	MOV	R4,\$TMP4	:FPS ERROR D	
5599	030530	010537	001240		MOV	R5,\$TMP3		
5600	030534	104164		1\$:	ERROR	+164		
5601	030536	000137	031322		JMP	CCDONE		
5602	030542	010437	001242	CCER90:	MOV	R4,\$TMP4	:FPS ERROR F	
5603	030546	010537	001240		MOV	R5,\$TMP3		
5604	030552	104165		1\$:	ERROR	+165		
5605	030554	000137	031322		JMP	CCDONE		
5606	030560	012737	031212	001240	CCER1:	MOV	#CCP2,\$TMP3	:DATA ERROR D
5607	030566	012737	031212	001246		MOV	#CCP2,\$TMP6	
5608	030574	012737	031172	001242	CCER50:	MOV	#CCP0,\$TMP4	
5609	030602	012737	031162	001244		MOV	#CCDATO,\$TMP5	
5610	030610	104166		1\$:	ERROR	+166		
5611	030612	000137	031322		JMP	CCDONE		
5612	030616	012737	031212	001240	CCER2:	MOV	#CCP2,\$TMP3	:CONSTANT BAD D(B)
5613	030624	012737	031212	001246		MOV	#CCP2,\$TMP6	
5614	030632	012737	031172	001242	CCER22:	MOV	#CCP0,\$TMP4	
5615	030640	012737	031162	001244		MOV	#CCDATO,\$TMP5	
5616	030646	104172		1\$:	ERROR	+172		
5617	030650	000137	031322		JMP	CCDONE		
5618	030654	012737	031202	001240	CCER3:	MOV	#CCP1,\$TMP3	
5619	030662	012737	031262	001246		MOV	#CCP7,\$TMP6	
5620	030670	000741			BR	CCER50		
5621	030672	012737	031202	001240	CCER4:	MOV	#CCP1,\$TMP3	:CONSTANT BAD D(G)
5622	030700	012737	031262	001246		MOV	#CCP7,\$TMP6	
5623	030706	012737	031172	001242	CCER44:	MOV	#CCP0,\$TMP4	
5624	030714	012737	031162	001244		MOV	#CCDATO,\$TMP5	
5625	030722	104173		1\$:	ERROR	+173		
5626	030724	000137	031322		JMP	CCDONE		
5627	030730	012737	031252	001240	CCER5:	MOV	#CCP6,\$TMP3	:DATA ERROR F
5628	030736	012737	031252	001246		MOV	#CCP6,\$TMP6	
5629	030744	012737	031172	001242	CCER55:	MOV	#CCP0,\$TMP4	
5630	030752	012737	031162	001244		MOV	#CCDATO,\$TMP5	
5631	030760	104170		1\$:	ERROR	+170		
5632	030762	000557			BR	CCDONE		
5633	030764	012737	031252	001240	CCER6:	MOV	#CCP6,\$TMP3	:CONSTANT BAD F(B)
5634	030772	012737	031252	001246		MOV	#CCP6,\$TMP6	
5635	031000	012737	031172	001242		MOV	#CCP0,\$TMP4	

```

5636 031006 012737 031162 001244      MOV      #CCDAT0,$STMP5
5637 031014 104174      1$:      ERROR    +174
5638 031016 000541      BR       CCDONE
5639 031020 012737 031242 001240  CCER7:  MOV      #CCP5,$STMP3      ;DATA ERROR F
5640 031026 012737 031272 001246      MOV      #CCP10,$STMP6
5641 031034 000743      BR       CCER55
5642 031036 012737 031242 001240  CCER8:  MOV      #CCP5,$STMP3      ;CONSTANT BAD F(G)
5643 031044 012737 031272 001246      MOV      #CCP10,$STMP6
5644 031052 012737 031162 001244      MOV      #CCDAT0,$STMP5
5645 031060 012737 031172 001242      MOV      #CCP0,$STMP4
5646 031066 104175      1$:      ERROR    +175
5647 031070 000514      BR       CCDONE
5648 031072 012737 031222 001240  CCER10: MOV      #CCP3,$STMP3      ;DATA ERROR D
5649 031100 012737 031302 001246      MOV      #CCP11,$STMP6
5650 031106 000632      BR       CCER50
5651 031110 012737 031222 001240  CCER11: MOV      #CCP3,$STMP3      ;CONSTANT BAD D(G)
5652 031116 012737 031302 001246      MOV      #CCP11,$STMP6
5653 031124 000670      BR       CCER44
5654 031126 012737 031232 001240  CCER12: MOV      #CCP4,$STMP3      ;DATA ERROR D
5655 031134 012737 031232 001246      MOV      #CCP4,$STMP6
5656 031142 000614      BR       CCER50
5657 031144 012737 031232 001240  CCER13: MOV      #CCP4,$STMP3      ;CONSTANT BAD D(B)
5658 031152 012737 031232 001246      MOV      #CCP4,$STMP6
5659 031160 000624      BR       CCER22
5660 031162 000000 000000 000000  CCDAT0: .WORD    0,0,0,0
5661 031172 000200 000000 000000  CCP0:   .WORD    200,0,0,0      ;E(AC)=1
5662 031202 016200 000000 000000  CCP1:   .WORD    16200,0,0,0     ;E(FSRC)=E(AC)+56=57
5663 031212 016400 000000 000000  CCP2:   .WORD    16400,0,0,0     ;E(FSRC)=E(AC)+57=58
5664 031222 000400 000000 000000  CCP3:   .WORD    400,0,0,0       ;E(FSRC)=E(AC)+1=2
5665 031232 031200 000000 000000  CCP4:   .WORD    31200,0,0,0     ;E(FSRC)=E(AC)+100=101=145(OCT)
5666 031242 006200 000000 000000  CCP5:   .WORD    6200,0,0,0      ;E(FSRC)=E(AC)+24=25=31(OCT)
5667 031252 006400 000000 000000  CCP6:   .WORD    6400,0,0,0      ;E(FSRC)=E(AC)+25=26=32(OCT)
5668 031262 016200 000000 000000  CCP7:   .WORD    16200,0,0,1
5669 031272 006200 000001 000000  CCP10:  .WORD    6200,1,0,0
5670 031302 000500 000000 000000  CCP11:  .WORD    500,0,0,0
5671 031312 000200 000000 000000  CCP12:  .WORD    200,0,0,0
5672 031322 000000 000000 000000  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?).
    
```

5673

5684

```

.SBTTL TEST # 33 - ADDF & ADD WITH E(AC) GREATER THAN E(FSRC) TEST
:*****
:TEST 33      ADDF & ADD WITH E(AC) GREATER THAN E(FSRC) TEST
:
:THIS IS A TEST OF THE ADD 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).
:
:*****
    
```

```

5685 031324 000004
5686 031326
5687 031334 012737 031334 001110
5688 031340 170104 003200
5689 031342 012737 032214 000244
5690 031350 012737 031370 001236
5691
5692 031356 012700 032556
5693 031362 172410
5694 031364 012700 032546
5695 031370 172010
5696 031372 170205
5697 031374 012700 032526
5698 031400 174010
5699 031402 012701 032556
5700 031406 012702 000004
5701 031412 022021
5702 031414 001402
5703 031416 000137 032254
5704 031422 077205
5705
5706 031424 020405
5707 031426 001402
5708 031430 000137 032214
5709
5710 031434
5711 031434 012737 031442 001110
5712 031442 012704 003200
5713 031446 170104 001236
5714 031450 012737 031470
5715 031456 012700 032576
5716 031462 172410
5717 031464 012700 032546
5718 031470 172010
5719 031472 170205
5720 031474 012700 032526
5721 031500 174010
5722 031502 012701 032636
5723 031506 012702 000004
5724 031512 022021
5725 031514 001415
5726 031516 012700 032526
5727 031522 012701 032576
    
```

```

TST33: SCOPE
:EXPONENT DIFFERENCE=57=71 (OCT) FD=1
BB1:
1$:  MOV #1$, $LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
      MOV #3200,R4        ;SET FIV FIV, AND FD
      LDFPS R4
      MOV #BBERO,FPVECT  ;SET UP FOR ERROR
      MOV #BB2,$TMP2     ;IN CASE THE OVER\
                          ;UNDER FLOWS FAIL.
                          ;SET ACO OPERAND.
      MOV #BBPAT2,R0
      LDD (R0),AC0
      MOV #BBPAT1,R0     ;FSRC
BB2:  ADDD (R0),AC0      ;TEST INSTRUCTION
      STFPS R5
BB3:  MOV #BBDAT0,R0     ;GET THE RESULT
      STD ACO,(R0)
      MOV #BBPAT2,R1     ;RESULT CORRECT?
      MOV #4,R2
BB4:  CMP (R0)+,(R1)+
      BEQ BB5
BB5:  JMP BBER1          ;DATA ERROR D
      SOB R2,BB4        ;WAS FPS CORRECT?
      CMP R4,R5
      BEQ BB6
      JMP BBER0          ;FPS ERROR
:EXPONENT DIFFERENCE=56=70 (OCT) FD=1
BB6:
1$:  MOV #1$, $LPERR      ;SET UP THE LOOP ON ERROR ADDRESS.
      MOV #3200,R4        ;SET FIV,FIV, AND FD
      LDFPS R4
      MOV #BB7,$TMP2
      MOV #BBPAT4,R0     ;SET ACO OPERAND
      LDD (R0),AC0
      MOV #BBPAT1,R0     ;FSRC
BB7:  ADDD (R0),AC0      ;TEST INSTRUCTION
      STFPS R5
      MOV #BBDAT0,R0     ;GET FPS
      STD ACO,(R0)      ;GET THE RESULT
      MOV #BBP10,R1     ;IS IT CORRECT
      MOV #4,R2
BB10: CMP (R0)+,(R1)+
      BEQ BB13
      MOV #BBDAT0,R0    ;DID A BAD
      MOV #BBPAT4,R1   ;CONSTANT (NOT 57)
    
```



```

5727 031526 012702 000004          MOV      #4,R2          :GET GENERATED
5728 031532 022021          BB11:  CMP      (R0)+,(R1)+ :FOR THE ALIGNMENT
5729 031534 001402          BEQ      BB12          :FLOWS?
5730 031536 000137 032312          JMP      BBER2        :DATA ERROR.D
5731 031542 077205          BB12:  SOB      R2,BB11
5732 031544 000137 032330          JMP      BBER3        :BAD CONSTANT.D
5733 031550 077220          BB13:  SOB      R2,BB10
5734 031552 020405          CMP      R4,R5        :FPS CORRECT?
5735 031554 001402          BEQ      BB14
5736 031556 000137 032214          JMP      BBER0        :BAD FPS.
5737          :EXPONENT DIFFERENCE=25=31 (OCT) FD=0
5738 031562          BB14:
5739 031562 012737 031570 001110          MOV      #1$,$LPERR    :SET UP THE LOOP ON ERROR ADDRESS.
5740 031570 012737 031616 001236          1$:  MOV      #BB15,$TMP2
5741 031576 012700 032536          MOV      #BBPAT0,R0    :SET UP ACO OPERAND
5742 031604 012704 003000          LDD      (R0),AC0
5743          MOV      #3000,R4    :SET FIV AND FIV
5744 031610 170104          LDFPS   R4             :CLEAR FD
5745 031612 012700 032546          BB15:  MOV      #BBPAT1,R0    :FSRC
5746 031616 172010          ADDF    (R0),AC0      :TEST INSTRUCTION
5747 031620 170205          STFPS  R5
5748 031622 170011          SETD
5749 031624 012700 032526          MOV      #BBDAT0,R0    :REENTERED DOUBLE MODE.
5750 031630 174010          STD     AC0,(R0)      :GET THE RESULT
5751 031632 012701 032536          MOV      #BBPAT0,R1
5752 031636 012702 000002          MOV      #2,R2
5753 031642 022021          BB16:  CMP      (R0)+,(R1)+  :IS THE RESULT
5754 031644 001402          BEQ      BB17          :CORRECT?
5755 031646 000137 032364          JMP      BBER4        :DATA ERROR F
5756 031652 077205          BB17:  SOB      R2,BB16
5757 031654 020405          CMP      R4,R5        :IS FPS CORRECT?
5758 031656 001402          BEQ      BB20
5759 031660 000137 032234          JMP      BBER10       :FPS ERROR.
5760          :EXPONENT DIFFERENCE=24=30 (OCT)
5761 031664          BB20:
5762 031664 012737 031672 001110          MOV      #1$,$LPERR    :SET UP THE LOOP ON ERROR ADDRESS.
5763 031672 012737 031720 001236          1$:  MOV      #BB21,$TMP2
5764 031700 012700 032566          MOV      #BBPAT3,R0    :SET UP ACO OPERAND.
5765 031704 172410          LDD      (R0),AC0
5766 031706 012704 003000          MOV      #3000,R4
5767 031712 170104          LDFPS   R4             :SET FIU,FIV. CLEAR FD.
5768 031714 012700 032546          BB21:  MOV      #BBPAT1,R0    :FSRC
5769 031722 172010          ADDF    (R0),AC0      :TEST INSTRUCTION
5770 031724 170205          STFPS  R5
5771 031726 012700 032526          SETD
5772 031732 174010          MOV      #BBDAT0,R0    :REENTER DOUBLE MODE
5773 031734 012701 032626          STD     AC0,(R0)      :GET THE RESULT
5774 031740 012702 000002          MOV      #BBP7,R1
5775 031744 022021          BB22:  MOV      #2,R2
5776 031746 001415          CMP      (R0)+,(R1)+  :IS THE RESULT CORRECT?
5777 031750 012700 032526          BEQ      BB25
5778 031754 012701 032566          MOV      #BBDAT0,R0    :WAS A BAD CONSTANT
5779 031760 012702 000002          MOV      #BBPAT3,R1    :USED (NOT 25) IN
5780 031764 022021          BB23:  MOV      #2,R2
5781 031766 001402          CMP      (R0)+,(R1)+  :THE ALLIGN FLOWS?
5781          BEQ      BB24
    
```

```

5782 031770 000137 032420          JMP      BBER5          ;DATA ERROR F
5783 031774 077205          SOB      R2,BB23
5784 031776 000137 032436          JMP      BBER6          ;BAD CONSTANT F
5785 032002 077220          SOB      R2,BB22
5786 032004 020405          CMP      R4,R5
5787 032006 001402          BEQ      BB26
5788 032010 000137 032234          JMP      BBER10        ;BAD FPS.
5789
5790 032014 012737 032022 001110      ;EXPONENT DIFFERENCE=1
      032014 012737 032022 001236      BB26:
5791 032022 012737 032050          MOV      #1$, $LPERR   ;SET UP THE LOOP ON ERROR ADDRESS.
5792 032030 012704 003200          MOV      #BB27,$TMP2
5793 032034 170104          MOV      #3200,R4
5794 032036 012700 032606          LDFPS   R4             ;SET UP ACO OPERAND
5795 032042 172410          MOV      #BBPAT5,R0
5796 032044 012700 032546          LDD      (R0),AC0
5797 032050 172010          MOV      #BBPAT1,R0   ;FSRC
5798 032052 170205          BB27:  ADDD   (R0),AC0  ;TEST INSTRUCTION
5799 032054 012700 032526          STFPS   R5
5800 032060 174010          MOV      #BBDAT0,R0   ;GET THE RESULT.
5801 032062 012701 032646          STD      ACO,(R0)
5802 032066 012702 000004          MOV      #BBP11,R1   ;IS IT CORRECT?
5803 032072 022021          MOV      #4,R2
5804 032074 001402          BB30:  CMP      (R0)+,(R1)+
5805 032076 000137 032472          BEQ      BB31          ;DATA ERROR D
5806 032102 077205          BB31:  JMP      BBER7
5807 032104 020405          SOB      R2,BB30
5808 032106 001402          CMP      R4,R5        ;IS FPS CORRECT
5809 032110 000137 032214          BEQ      BB32
5810
5811 032114 012737 032122 001110      ;EXPONENT DIFFERENCE=100=144 (OCT)
      032114 012737 032122 001236      BB32:
5812 032122 012737 032150          MOV      #1$, $LPERR   ;SET UP THE LOOP ON ERROR ADDRESS.
5813 032130 012704 003200          MOV      #BB33,$TMP2
5814 032134 170104          MOV      #3200,R4
5815 032136 012700 032616          LDFPS   R4             ;SET FIV,FIV AND FD
5816 032142 172410          MOV      #BBPAT6,R0   ;SET UP ACO OPERAND.
5817 032144 012700 032546          LDD      (R0),AC0
5818 032150 172010          MOV      #BBPAT1,R0   ;FSRC
5819 032152 170205          BB33:  ADDD   (R0),AC0  ;TEST INSTRUCTION
5820 032154 012700 032526          STFPS   R5
5821 032160 174010          MOV      #BBDAT0,R0   ;GET THE RESULT
5822 032162 012701 032616          STD      ACO,(R0)
5823 032166 012702 000004          MOV      #BBPAT6,R1   ;IS IT CORRECT
5824 032172 022021          MOV      #4,R2
5825 032174 001402          BB34:  CMP      (R0)+,(R1)+
5826 032176 000137 032510          BEQ      BB35          ;DATA ERROR D
5827 032202 077205          BB35:  JMP      BBER8
5828 032204 020405          SOB      R2,BB34
5829 032206 001002          CMP      R4,R5        ;IS FPS CORRECT
5830 032210 000137 032656          BNE      BBER0
5831 032214 010437 001242          JMP      BBDONE
5832 032220 010537 001240          BBER0: MOV      R4,$TMP4   ;FPS ERROR D
5833 032224 104164          MOV      R5,$TMP3
5834 032226 104413          1$:  ERROR +164
      RSETUP
          ;GO INITIALIZE THE FPS AND STACK; AND
          ;SEE IF THE USER HAS EXPRESSED
          ;THE DESIRE TO CHANGE THE SOFTWARE
    
```


5835 032230 000137 032656
 5836 032234 010437 001242
 5837 032240 010537 001240
 5838 032244 104165
 5839 032246 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?).

5840 032250 000137 032656
 5841 032254 012737 032556 001242
 5842 032262 012737 032556 001246
 5843 032270 012737 032546 001240
 5844 032276 012737 032526 001244
 5845 032304 104166
 5846 032306 000137 032656
 5847 032312 012737 032576 001242
 5848 032320 012737 032636 001246
 5849 032326 000760
 5850 032330 012737 032576 001242
 5851 032336 012737 032636 001246
 5852 032344 012737 032546 001240
 5853 032352 012737 032526 001244
 5854 032360 104167
 5855 032362 000535
 5856 032364 012737 032536 001242
 5857 032372 012737 032536 001246
 5858 032400 012737 032546 001240
 5859 032406 012737 032526 001244
 5860 032414 104170
 5861 032416 000517
 5862 032420 012737 032566 001242
 5863 032426 012737 032626 001246
 5864 032434 000761
 5865 032436 012737 032566 001242
 5866 032444 012737 032626 001246
 5867 032452 012737 032546 001240
 5868 032460 012737 032526 001244
 5869 032466 104171
 5870 032470 000472
 5871 032472 012737 032606 001242
 5872 032500 012737 032546 001246
 5873 032506 000670
 5874 032510 012737 032616 001242
 5875 032516 012737 032616 001246
 5876 032524 000661
 5877 032526 000000 000000 000000
 5878 032536 006400 000000 000000
 5879 032546 000200 000000 000000
 5880 032556 016400 000000 000000
 5881 032566 006200 000000 000000
 5882 032576 016200 000000 000000
 5883 032606 000400 000000 000000
 5884 032616 031200 000000 000000
 5885 032626 006200 000001 000000

BBER1: JMP BBDONE
 MOV #BBPAT2,\$TMP4
 MOV #BBPAT2,\$TMP6
 BBER11: 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

5886 032636 016200 000000 000000 BBP10: .WORD 16200,0,0,1 :BBPAT4 RES
5887 032646 000500 000000 000000 BBP11: .WORD 500,0,0,0 :BBPAT5 RES
5888 032656
032656 104413 BB DONE: 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?).

5942	033076	000137	034102		JMP	DDER3			:FLOW FAILURE
5943	033102	077205		DD11:	SOB	R2,DD10			
5944	033104	052704	000004		BIS	#4,R4			
5945	033110	020405			CMP	R4,R5			:FPS CORRECT?
5946	033112	001402			BEQ	DD12			
5947	033114	000137	033770		JMP	DDERO			:BAD FPS
5948								AC=-FSRC	
5949	033120				:AC NEG	FSRC POS			
5950	033120	012737	033126	001110	DD12:				
5951	033126	012704	003200		1\$:	MOV #1\$, \$LPERR			:SET UP THE LOOP ON ERROR ADDRESS.
5952	033132	170104			MOV	#3200,R4			:SET FIU, FIV, AND FD
5953	033134	012737	033154	001236	LDFPS	R4			
5954	033142	012700	034576		MOV	#DD13,\$TMP2			
5955	033146	172410			MOV	#DDP1,R0			:SET ACO OPERAND
5956	033150	012700	034606		LDD	(R0),ACO			:FSRC
5957	033154	172010		DD13:	MOV	#DDP2,R0			:TEST INSTRUCTION
5958	033156	170205			ADDD	(R0),ACO			:GET FPS
5959	033160	012700	034556		STFPS	R5			:GET THE RESULT
5960	033164	174010			MOV	#DDDATO,R0			
5961	033166	012701	034566		STD	ACO,(R0)			:IS IT CORRECT
5962	033172	012702	000004		MOV	#DDP0,R1			
5963	033176	022021		DD14:	MOV	#4,R2			
5964	033200	001402			CMP	(R0)+,(R1)+			
5965	033202	000137	034140		BEQ	DD15			:FLOW FAILURE 216,440,121
5966	033206	077205		DD15:	JMP	DDER4			
5967	033210	052704	000004		SOB	R2,DD14			
5968	033214	020405			BIS	#4,R4			
5969	033216	001402			CMP	R4,R5			:EPS CORRECT?
5970	033220	000137	033770		BEQ	DD16			
5971	033224				JMP	DDERO			:BAD FPS
5972	033224	012737	033232	001110	:ACO POC	FSRC NEG		/AC/ > /FSRC/	
5973	033232	012704	003200		DD16:				
5974	033236	170104			1\$:	MOV #1\$, \$LPERR			:SET UP THE LOOP ON ERROR ADDRESS.
5975	033240	012737	033260	001236	MOV	#3200,R4			:SET FIV, FIV AND FD
5976	033246	012700	034616		LDFPS	R4			
5977	033252	172410			MOV	#DD17,\$TMP2			
5978	033254	012700	034646		MOV	#DDP3,R0			:SET ACO OPERAND
5979	033260	172010		DD17:	LDD	(R0),ACO			:ESPC
5980	033262	170205			MOV	#DDP6,R0			:TEST INSTRUCTION
5981	033264	012700	034556		ADDD	(R0),ACO			:GET FPS
5982	033270	174010			STFPS	R5			:GET THE RESULT
5983	033272	012701	034656		MOV	#DDDATO,R0			
5984	033276	012702	000004		STD	ACO,(R0)			:IS IT CORRECT
5985	033302	022021		DD18:	MOV	#DDP7,R1			
5986	033304	001415			MOV	#4,R2			
5987	033306	012700	034556		CMP	(R0)+,(R1)+			:FLOWS FAILURE
5988	033312	012701	034666		BEQ	DD21			:216,440,101
5989	033316	012702	000004		MOV	#DDDATO,R0			:GET GENERATED
5990	033322	022021		DD19:	MOV	#DDP8,R1			
5991	033324	001402			MOV	#4,R2			
5992	033326	000137	034176		CMP	(R0)+,(R1)+			
5993	033332	077205		DD20:	BEQ	DD20			:DATA ERROR.
5994	033334	000137	034234		JMP	DDER5			
5995	033340	077220		DD21:	SOB	R2,DD19			
5996	033342	020405			JMP	DDER6			
5997	033344	001402			SOB	R2,DD18			:EPS CORRECT?
					CMP	R4,R5			
					BEQ	DD22			

5997	033346	000137	033770		JMP	DDER0			:BAD FPS
5998					:AC NEG	FSRC	POS	/FSRC/ > /AC/	
5999	033352				DD22:				
6000	033352	012737	033360	001110	MOV	#1\$, \$LPERR			:SET UP THE LOOP ON ERROR ADDRESS.
6001	033360	012704	003200		1\$:	MOV	#3200, R4		:SET FIO, FIV, AND FD
6002	033364	170104			LDFPS	R4			
6003	033366	012737	033406	001236	MOV	#DD23, \$TMP2			
6004	033374	012700	034646		MOV	#DDP6, R0			:SET ACO OPERAND
6005	033400	172410			LDD	(R0), ACO			
6006	033402	012700	034616		MOV	#DDP3, R0			:FSPC
6007	033406	172010			DD23:	ADDD	(R0), ACO		:TEST INSTRUCTION
6008	033410	170205			STFPS	R5			:GET FPS
6009	033412	012700	034556		MOV	#DDDATO, R0			:GET THE RESULT
6010	033416	174010			STD	ACO, (R0)			
6011	033420	012701	034656		MOV	#DDP7, R1			:IS IT CORRECT?
6012	033424	012702	000004		MOV	#4, R2			
6013	033430	022021			DD24:	CMP	(R0)+, (R1)+		
6014	033432	001415			BEQ	DD27			
6015	033434	012700	034556		MOV	#DDDATO, R0			:FLO, S FAILURE
6016	033440	012701	034666		MOV	#DDP8, R1			:CONSTANT (NOT 57)
6017	033444	012702	000004		MOV	#4, R2			:216, 042, 101
6018	033450	021011			DD25:	CMP	(R0), (R1)		
6019	033452	001402			BEQ	DD26			
6020	033454	000137	034272		JMP	DDER7			:DATA ERROR.
6021	033460	077205			DD26:	SOB	R2, DD25		
6022	033462	000137	034330		JMP	DDER8			
6023	033466	077220			DD27:	SOB	R2, DD24		
6024	033470	020405			CMP	R4, R5			:FPS CORRECT?
6025	033472	001402			BEQ	DD30			
6026	033474	000137	033770		JMP	DDER0			:BAD FPS
6027					:ACO POS	FSRC	NEG	/AC/ < /FRSRC/	
6028	033500				DD30:				
6029	033500	012737	033506	001110	MOV	#1\$, \$LPERR			:SET UP THE LOOP ON ERROR ADDRESS.
6030	033506	012704	003200		1\$:	MOV	#3200, R4		:SET FIO, FIV, AND FD
6031	033512	170104			LDFPS	R4			
6032	033514	012737	033534	001236	MOV	#DD31, \$TMP2			
6033	033522	012700	034626		MOV	#DDP4, R0			:SET ACO OPERAND
6034	033526	172410			LDD	(R0), ACO			
6035	033530	012700	034636		MOV	#DDP5, R0			:FSPC
6036	033534	172010			DD31:	ADDD	(R0), ACO		:TEST INSTRUCTION
6037	033536	170205			STFPS	R5			:GET FPS
6038	033540	012700	034556		MOV	#DDDATO, R0			:GET THE RESULT
6039	033544	174010			STD	ACO, (R0)			
6040	033546	012701	034666		MOV	#DDP8, R1			:IS IT CORRECT
6041	033552	012702	000004		MOV	#4, R2			
6042	033556	022021			DD32:	CMP	(R0)+, (R1)+		
6043	033560	001415			BEQ	DD35			:ADD-SUB
6044	033562	012700	034556		MOV	#DDDATO, R0			:FLOWAS FAILURE
6045	033566	012701	034656		MOV	#DDP7, R1			:CON 216 N440 NOT 141
6046	033572	012702	000004		MOV	#4, R2			:GET GENERATED
6047	033576	022021			DD33:	CMP	(R0)+, (R1)+		:FOR THE ALLIGNMENT
6048	033600	001402			BEQ	DD34			:FLOWS?
6049	033602	000137	034366		JMP	DDER9			:DATA ERROR, D
6050	033606	077205			DD34:	SOB	R2, DD33		
6051	033610	000137	034424		JMP	DDER10			
6052	033614	077220			DD35:	SOB	R2, DD32		
6053	033616	052704	000010		BIS	#10, R4			

	034132	104214			1\$:	ERROR	+214
	034134	000137	034706			JMP	DDDONE
6092	034140				DDER4:		
	034140	012737	034606	001240		MOV	#DDP2,\$TMP3
	034146	012737	034576	001242		MOV	#DDP1,\$TMP4
	034154	012737	034556	001244		MOV	#DDDATO,\$TMP5
	034162	012737	034566	001246		MOV	#DDPO,\$TMP6
	034170	104200			1\$:	ERROR	+200
	034172	000137	034706			JMP	DDDONE
6093	034176				DDER5:		
	034176	012737	034646	001240		MOV	#DDP6,\$TMP3
	034204	012737	034616	001242		MOV	#DDP3,\$TMP4
	034212	012737	034556	001244		MOV	#DDDATO,\$TMP5
	034220	012737	034656	001246		MOV	#DDP7,\$TMP6
	034226	104165			1\$:	ERROR	+165
	034230	000137	034706			JMP	DDDONE
6094	034234				DDER6:		
	034234	012737	034646	001240		MOV	#DDP6,\$TMP3
	034242	012737	034616	001242		MOV	#DDP3,\$TMP4
	034250	012737	034556	001244		MOV	#DDDATO,\$TMP5
	034256	012737	034656	001246		MOV	#DDP7,\$TMP6
	034264	104201			1\$:	ERROR	+201
	034266	000137	034706			JMP	DDDONE
6095	034272				DDER7:		
	034272	012737	034616	001240		MOV	#DDP3,\$TMP3
	034300	012737	034646	001242		MOV	#DDP6,\$TMP4
	034306	012737	034556	001244		MOV	#DDDATO,\$TMP5
	034314	012737	034656	001246		MOV	#DDP7,\$TMP6
	034322	104165			1\$:	ERROR	+165
	034324	000137	034706			JMP	DDDONE
6096	034330				DDER8:		
	034330	012737	034616	001240		MOV	#DDP3,\$TMP3
	034336	012737	034646	001242		MOV	#DDP6,\$TMP4
	034344	012737	034556	001244		MOV	#DDDATO,\$TMP5
	034352	012737	034656	001246		MOV	#DDP7,\$TMP6
	034360	104202			1\$:	ERROR	+202
	034362	000137	034706			JMP	DDDONE
6097	034366				DDER9:		
	034366	012737	034636	001240		MOV	#DDP5,\$TMP3
	034374	012737	034626	001242		MOV	#DDP4,\$TMP4
	034402	012737	034556	001244		MOV	#DDDATO,\$TMP5
	034410	012737	034666	001246		MOV	#DDP8,\$TMP6
	034416	104165			1\$:	ERROR	+165
	034420	000137	034706			JMP	DDDONE
6098	034424				DDER10:		
	034424	012737	034636	001240		MOV	#DDP5,\$TMP3
	034432	012737	034626	001242		MOV	#DDP4,\$TMP4
	034440	012737	034556	001244		MOV	#DDDATO,\$TMP5
	034446	012737	034666	001246		MOV	#DDP8,\$TMP6
	034454	104203			1\$:	ERROR	+203
	034456	000137	034706			JMP	DDDONE
6099	034462				DDER11:		
	034462	012737	034626	001240		MOV	#DDP4,\$TMP3
	034470	012737	034636	001242		MOV	#DDP5,\$TMP4
	034476	012737	034556	001244		MOV	#DDDATO,\$TMP5
	034504	012737	034666	001246		MOV	#DDP8,\$TMP6
	034512	104165			1\$:	ERROR	+165


```

6100 034514 000137 034706          JMP      DDDONE
      034520          DDERR12:  MOV      #DDP4,$TMP3
      034520 012737 034626 001240          MOV      #DDP5,$TMP4
      034526 012737 034636 001242          MOV      #DDDAT0,$TMP5
      034534 012737 034556 001244          MOV      #DDP8,$TMP6
      034542 012737 034666 001246          1$:      ERROR      +204
      034550 104204          JMP      DDDONE
      034552 000137 034706          DDDAT0: .WORD    0,0,0,0
6101 034556 000000 000000 000000 DDP0:    .WORD    0,0,0,0
6102 034566 000000 000000 000000 DDP1:    .WORD    100200,0,0,0    :-DDP2
6103 034576 100200 000000 000000 DDP2:    .WORD    200,0,0,0      :-DDP1
6104 034606 000200 000000 000000 DDP3:    .WORD    1100,0,0,0     :EXP=4 :FRAC=...110...
6105 034616 001100 000000 000000 DDP4:    .WORD    600,0,0,0     :EXP=3 :FRAC=...100...
6106 034626 000600 000000 000000 DDP5:    .WORD    101100,0,0,0  :-DDP3
6107 034636 101100 000000 000000 DDP6:    .WORD    100600,0,0,0  :-DDP4
6108 034646 100600 000000 000000 DDP7:    .WORD    1000,0,0,0    :DDP3+DDP6
6109 034656 001000 000000 000000 DDP8:    .WORD    101000,0,0,0  :DDP5+DDP4
6110 034666 101000 000000 000000 DDP9:    .WORD    100400,0,0,0  :DDP1+DDP1
6111 034676 100400 000000 000000 DDDONE:
6112 034706          RSETUP

      034706 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?).
    
```

6120

.....
: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
:.....

6121 034710 030004
6122 034712
6123 034712 012737 034720 001110
6124 034720 012704 003200
6125 034724 170104
6126 034726 012737 034746 001236
6127 034734 012700 035432
6128 034740 172410
6129 034742 012700 035432
6130 034746 173010
6131 034750 170205
6132 034752 012700 035410
6133 034756 174010
6134 034760 012701 035420
6135 034764 012702 000004
6136 034770 022021
6137 034772 001415
6138 034774 012700 035410
6139 035000 012701 035442
6140 035004 012702 000004
6141 035010 022021
6142 035012 001402
6143 035014 000137 035220
6144 035020 077205
6145 035022 000137 035256
6146 035026 077220
6147 035030 052704 000004
6148 035034 020405
6149 035036 001402
6150 035040 000137 035202
6151 035044
6152 035044 012737 035052 001110
6153 035052 012704 003200
6154 035056 170104
6155 035060 012737 035100 001236
6156 035066 012700 035452
6157 035072 172410
6158 035074 012700 035452
6159 035100 173010
6160 035102 170205
6161 035104 012700 035410
6162 035110 174010
6163 035112 012701 035420
6164 035116 012702 000004
6165 035122 022021
6165 035124 001415

TEST35: SCOPE
: USE POSITIVE OPERANDS
EE1: ;SET UP THE LOOP ON ERROR ADDRESS.
1\$: MOV #1\$, \$LPERR ;SET FIO, FIV, AND FD
MOV #3200, R4
LDFPS R4
MOV #EE2, \$TMP2
MOV #EEP1, R0 ;SET ACO OPERAND
LDD (R0), ACO
MOV #EEP1, R0 ;FSPC
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 ;DID A BAD
MOV #EEDATO, R0 ;CONSTANT (NOT 57)
MOV #EEP2, R1 ;GET GENERATED
MOV #4, R2 ;FOR THE ALLIGNMENT
EE4: CMP (R0)+, (R1)+
BEQ EE5 ;FLOWS?
JMP EEER1 ;DATA ERROR.D
EE5: SOB R2, EE4 ;BAD CONSTANT.D
JMP EEER2
EE6: SOB R2, EE3
BIS #4, R4
CMP R4, R5 ;FPS CORRECT?
BEQ EE7 ;BAD FPS
JMP EEER0
: USE NEGATIVE OPERANDS
EE7: ;SET UP THE LOOP ON ERROR ADDRESS.
1\$: MOV #1\$, \$LPERR ;SET FIO, FIV, AND FD
MOV #3200, R4
LDFPS R4
MOV #EE8, \$TMP2
MOV #EEP3, R0 ;SET ACO OPERAND
LDD (R0), ACO
MOV #EEP3, R0 ;FSPC
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

6166	035126	012700	035410		MOV	#EEDATO,R0		:DID A BAD
6167	035132	012701	035462		MOV	#EEP4,R1		:CONSTANT (NOT 57)
6168	035136	012702	000J04		MOV	#4,R2		:GET GENERATED
6169	035142	022021		EE10:	CMP	(R0)+,(R1)+		:FOR THE ALLIGNMENT
6170	035144	001402			BEQ	EE11		:FLOWS?
6171	035146	000137	035314		JMP	EEER3		:DATA ERROR.D
6172	035152	077205		EE11:	SOB	R2,EE10		
6173	035154	000137	035352		JMP	EEER4		:BAD CONSTANT.D
6174	035160	077220		EE12:	SOB	R2,EE9		
6175	035162	052704	000004		BIS	#4,R4		
6176	035166	020405			CMP	R4,R5		:FPS CORRECT?
6177	035170	001402			BEQ	EE13		
6178	035172	000137	035202		JMP	EEER0		:BAD FPS.
6179	035176	000137	035472	EE13:	JMP	EEDONE		
6180	035202	010437	001242	EEER0:	MOV	R4,\$TMP4		:BAD FPS
6181	035206	010537	001240		MOV	R5,\$TMP3		
6182	035212	104205		1\$:	ERROR	+205		
6183	035214	000137	035472		JMP	EEDONE		
6184	035220			EEER1:				
	035220	012737	035432	001240	MOV	#EEP1,\$TMP3		
	035226	012737	035432	001242	MOV	#EEP1,\$TMP4		
	035234	012737	035410	001244	MOV	#EEDATO,\$TMP5		
	035242	012737	035420	001246	MOV	#EEO,\$TMP6		
	035250	104206		1\$:	ERROR	+206		
	035252	000137	035472		JMP	EEDONE		
6185	035256			EEER2:				
	035256	012737	035432	001240	MOV	#EEP1,\$TMP3		
	035264	012737	035432	001242	MOV	#EEP1,\$TMP4		
	035272	012737	035410	001244	MOV	#EEDATO,\$TMP5		
	035300	012737	035420	001246	MOV	#EEO,\$TMP6		
	035306	104207		1\$:	ERROR	+207		
	035310	000137	035472		JMP	EEDONE		
6186	035314			EEER3:				
	035314	012737	035452	001240	MOV	#EEP3,\$TMP3		
	035322	012737	035452	001242	MOV	#EEP3,\$TMP4		
	035330	012737	035410	001244	MOV	#EEDATO,\$TMP5		
	035336	012737	035420	001246	MOV	#EEO,\$TMP6		
	035344	104206		1\$:	ERROR	+206		
	035346	000137	035472		JMP	EEDONE		
6187	035352			EEER4:				
	035352	012737	035452	001240	MOV	#EEP3,\$TMP3		
	035360	012737	035452	001242	MOV	#EEP3,\$TMP4		
	035366	012737	035410	001244	MOV	#EEDATO,\$TMP5		
	035374	012737	035420	001246	MOV	#EEO,\$TMP6		
	035402	104207		1\$:	ERROR	+207		
	035404	000137	035472		JMP	EEDONE		
6188	035410	000000	000000	000000	EEDATO:	.WORD	0,0,0,0	
6189	035420	000000	000000	000000	EEO:	.WORD	0,0,0,0,0	
6190	035432	000200	000000	000000	EEP1:	.WORD	200,0,0,0	
6191	035442	000400	000000	000000	EEP2:	.WORD	400,0,0,0	
6192	035452	100200	000000	000000	EEP3:	.WORD	100200,0,0,0	
6193	035462	100400	000000	000000	EEP4:	.WORD	100400,0,0,0	
6194	035472				EEDONE:			
	035472	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?).

6204

```
.SBTTL TEST # 36 - NORMALIZE ALGORITHM TEST
*****
:TEST 36      NORMALIZE ALGORITHM TEST
:
: THIS IS A TEST OF THE NORMALIZE
: FLOW ALGORITHM. TWO PATTEPNS ARE USED,
: FIRST THE MINIMUM SITUATION REQUIRING ONE
: LEFT SHIFT AND THEN THE MAXIMUM SITUATION
: REQUIRING 56 SHIFTS.
:
:*****
```

```
6205 035474 000004
6206 035476
6207 035476 012737 035504 001110
6208 035504 012704 003200
6209 035510 170104
6210 035512 012737 035532 001236
6211 035520 012700 036030
6212 035524 172410
6213 035526 012700 036040
6214 035532 172010
6215 035534 170205
6216 035536 012700 036000
6217 035542 174010
6218 035544 012701 036050
6219 035550 012702 000004
6220 035554 022021
6221 035556 001401
6222 035560 000470
6223 035562 077204
6224 035564 020405
6225 035566 001401
6226 035570 000437
6227
6228 035572
6229 035572 012737 035600 001110
6230 035600 012704 003200
6231 035604 170104
6232 035606 012737 035626 001236
6233 035614 012700 036010
6234 035620 172410
6235 035622 012700 036020
6236 035626 172010
6237 035630 012700 036000
6238 035632 174010
6239 035636 012701 036050
6240 035640 012702 000004
6241 035644 022021
6242 035650 001401
6243 035652 000413
6244 035654 077204
6245 035656 020405
6246 035660 001401
6247 035662 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, $TMP2
MOV #FFP2, R0 ;SET ACO OPERAND
LDD (R0), ACO
MOV #FFP3, R0 ;FSPC
FF2: ADDD (R0), ACO ;TEST INSTRUCTION
STFPS R5 ;GET FPS
MOV #FFDAT0, R0 ;GET THE RESULT
STD ACO, (R0)
MOV #FFP4, R1 ;IS IT CORRECT
MOV #4, R2
FF3: CMP (R0)+, (R1)+
BEQ FF4
BR FFER2 ;BAD DATA
FF4: SOB R2, FF3
CMP R4, R5 ;FPS CORRECT?
BEQ FF5
BR FFER0 ;BAD FPS
:USE DATA PATTERNS WHICH REQUIRE 56 LEFT SHIFTS TO NORMALIZE
:THE RESULT
FF5:
1$: MOV #1$, $LPERR ;SET UP THE LOOP ON ERROR ADDRESS.
MOV #3200, R4 ;SET FIU, FIV, AND FD
LDFPS R4
MOV #FF6, $TMP2
MOV #FFP0, R0 ;SET ACO OPERAND
LDI (R0), ACO
MOV #FFP1, R0 ;FSRC
FF6: 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
```

```

6248 035666 000474          FF11:  BR      FFDONE
6249
6250 035670 010537 001240  FFER0:  MOV      R5,$TMP3
6251 035674 010437 001242      MOV      R4,$TMP4
6252 035700 104164          1$:     ERROR   +164
6253 035702 000466          BR      FFDONE
6254
6255 035704          FFER1:  MOV      #FFP1,$TMP3
        035704 012737 036020 001240      MOV      #FFP0,$TMP4
        035712 012737 036010 001242      MOV      #FFDAT0,$TMP5
        035720 012737 036000 001244      MOV      #FFP4,$TMP6
        035726 012737 036050 001246      1$:     ERROR   +210
        035734 104210          JMP     FFDONE
        035736 000137 036060
6256
6257 035742          FFER2:  MOV      #FFP3,$TMP3
        035742 012737 036040 001240      MOV      #FFP2,$TMP4
        035750 012737 036030 001242      MOV      #FFDAT0,$TMP5
        035756 012737 036000 001244      MOV      #FFP4,$TMP6
        035764 012737 036050 001246      1$:     ERROR   +210
        035772 104210          JMP     FFDONE
        035774 000137 036060
6258
6259
6260 036000 000000 000000 000000  FFDAT0: .WORD  0,0,0,0
6261 036010 016000 000000 000000  FFPO:   .WORD  16000,0,0,1
6262 036020 116000 000000 000000  FFP1:   .WORD  116000,0,0,0
6263 036030 000500 000000 000000  FFP2:   .WORD  500,0,0,0
6264 036040 100400 000000 000000  FFP3:   .WORD  100400,0,0,0
6265 036050 000200 000000 000000  FFP4:   .WORD  200,0,0,0
6266 036060          :FFP4=FFP0+FFP1=FFP3+FFP4
6267 036060          FFDONE:
          TST37:
    
```


6269

.SBTTL END OF PASS ROUTINE

```
*****
*INCREMENT THE PASS NUMBER ($PASS)
*INDICATE END-OF-PROGRAM AFTER 1 PASSES THRU THE PROGRAM
*IF SW12=1 INHIBIT TRACE TRAP
*IF THERES A MONITOR GO TO IT
*IF THERE ISN'T JUMP TO LOOP
$EOP:
036060 000004
036060 005037 001102      SCOPE
036062 005037 001302      CLR $TSTNM          ;;ZERO THE TEST NUMBER
036066 005037 001302      CLR $TIMES          ;;ZERO THE NUMBER OF ITERATIONS
036072 005237 001324      INC $PASS           ;;INCREMENT THE PASS NUMBER
036076 100004              BPL 1000$           ;;BRANCH IF STILL PLUS          :DPM002
036100 005037 001324      CLR $PASS           ;;CLEAR THE PASS COUNTER      :DPM002
036104 005237 036514      INC $PASS2          ;;INCREMENT OVERFLOW PASS COUNTER :DPM002
036110 005327              1000$: DEC (PC)+          ;;LOOP?
036112 000001              $EOPCT: .WORD 1
036114 003402              BLE 999$            ;NO          :DPM002
036116 000137 036442      JMP $DOAGN
036122 012737              999$: MOV (PC)+,a(PC)+  ;;RESTORE COUNTER
036124 000001              $ENDCT: .WORD 1
036126 036112              $EOPCT
036130 005737 001112      TST $ERTTL          ;SEE IF ANY ERRORS THIS PASS  :DPM002
036134 001007              BNE 5000$           ;BRANCH IF SO TO PRINT THE EOP :DPM002
036136 005737 036512      TST EPENDS          ;SEE IF EOP MSGS ARE DISABLED  :DPM002
036142 001120              BNE $GET42          ;BRANCH IF SO                  :DPM002
036144 032737 000003 001324 BIT #3,$PASS         ;PRINT EOP EVERY 4TH PASS      :DPM002
036152 001114              BNE $GET42          ;BRANCH IF NOT MULTIPLE OF 10  :DPM002
036154
036154 104401 036162      5000$: TYPE ,65$      ;;TYPE ASCIZ STRING
036160 000407              BR 64$             ;;GET OVER THE ASCIZ
036200
036200 005737 036514      ;;65$: .ASCIZ <12><15>/END PASS # /
036204 001440              64$: TST $PASS2          ;SEE IF OVERFLOW HAS NON-ZERO VALUE ;DPM002
036206 013746 036514      BEQ 4900$           ;BRANCH IF ZERO
036212 104403              MOV $PASS2,-(SP)    ;;SAVE $PASS2 FOR TYPEOUT
036214 006              ;;TYPE OVERFLOW PASS NUMBER IN OCTAL ;DPM002
036215 000              TYPOS
036216 005737 001324      .BYTE 6            ;;GO TYPE--OCTAL ASCII
036222 001007              .BYTE 0            ;;TYPE 6 DIGITS
036224 104401 036232      TST $PASS           ;;SUPPRESS LEADING ZEROS
036230 000403              BNE 3000$           ;SEE IF PASS COUNT IS ZERO      ;DPM002
036240
036240 000426              TYPE ,67$          ;;BRANCH IF NOT                  ;DPM002
036242 012737 070000 001244 BR 66$             ;;TYPE ASCIZ STRING
036250 033737 001244 001324 3000$: MOV #70000,$TMP5    ;;GET OVER THE ASCIZ
036256 001013              4000$: BIT $TMP5,$PASS ;GO TEST $ERTTL                  ;DPM002
036260 104401 036266      BNE 4900$           ;CHECK 5TH OCTAL DIGIT FIRST    ;DPM002
036264 000401              TYPE ,69$          ;CHECK TO SEE IF OCTAL DIGIT IS ZERO ;DPM002
036270
036270 006237 001244      BR 68$             ;BRANCH OUT IF ZERO
036274 006237 001244      .ASCIZ :0:        ;;TYPE ASCIZ STRING
036274 006237 001244      ASR $TMP5          ;;GET OVER THE ASCIZ
036274 006237 001244      ASR $TMP5          ;SHIFT THE THREE BITS RIGHT 3 PLACES ;DPM002
036274 006237 001244      ;                  ;DPM002
```

036300	006237	001244		ASR	\$TMP5	:			
036304	000761			BR	4000\$:	BRANCH BACK TO CHECK \$PASS		:DPM002
036306			4900\$:			:			:DPM002
036306	013746	001324		MOV	\$PASS,-(SP)	:	SAVE \$PASS FOR TYPEOUT		
						:	TYPE PASS NUMBER IN OCTAL		
036312	104403			TYPOS		:	GO TYPE--OCTAL ASCII		
036314	006			.BYTE	6	:	TYPE 6 DIGITS		
036315	000			.BYTE	0	:	SUPPRESS LEADING ZEROS		
036316	005737	001112	4910\$:	TST	\$ERTTL	:	SEE IF ANY ERRORS THIS PASS		:DPM002
036322	031426			BEQ	5001\$:	BRANCH AROUND REPORT IF NONE		:DPM002
036324	104401	036332		TYPE	71\$:	TYPE ASCII STRING		
036330	000415			BR	70\$:	GET OVER THE ASCIIZ		
			::71\$:	.ASCIIZ	/ TOTAL ERRORS	:	THIS PASS /		
			70\$:			:			
036364				MOV	\$ERTTL,-(SP)	:	SAVE \$ERTTL FOR TYPEOUT		
036364	013746	001112				:	TOTAL NUMBER OF ERRORS IN OCTAL		
						:	GO TYPE--OCTAL ASCII		
036370	104403			TYPOS		:	GO TYPE--OCTAL ASCII		
036372	006			.BYTE	6	:	TYPE 6 DIGITS		
036373	000			.BYTE	0	:	SUPPRESS LEADING ZEROS		
036374	005037	001112		CLR	\$ERTTL	:	CLEAR ERROR TOTAL		
036400	104401	001313	5001\$:	TYPE	,\$CRLF	:	TYPE CARRIAGE RETURN, LINE FEED		
036404	013700	000042	\$GET42:	MOV	@#42,R0	:	GET MONITOR ADDRESS		
036410	001414			BEQ	\$DOAGN	:	BRANCH IF NO MONITOR		
036412	005046			CLR	-(SP)	:	INSURE THE 'T' BIT IS CLEAR		
036414	012746	036422		MOV	#\$CLR.T,-(SP)	:	SETUP FOR AN RTI OR RTT		
036420	000426			BR	\$RTRN	:	GO DO AN RTI OR RTT TO LOAD THE PSW		
						:	WITH A CLEARED 'T' BIT		
			\$CLR.T:			:			
036422				MOV	@#42,R0	:	INSURE R0 CONTAINS THE MONITORS		
036422	013700	000042		BEQ	\$DOAGN	:	RETURN ADDRESS		
036426	001405			RESET		:	CLEAR THE WORLD		
036430	000005			JSR	PC,(R0)	:	GO TO MONITOR		
036432	004710		\$ENDAD:	NOP		:	SAVE ROOM		
036434	000240			NOP		:	FOR		
036436	000240			NOP		:	ACT11		
036440	000240					:			
036442			\$DOAGN:			:			
036442	104400			TRAP		:	PUSH OLD PSW AND PC ON STACK		
036444	042716	000020		BIC	#20,(SP)	:	CLEAR THE 'T' BIT		
036450	032777	010000	142462	BIT	#BIT12,@SWR	:	RUN WITH TRACE TRAP?		
036456	001005			BNE	1\$:	BR IF NO		
036460	005137	036504		COM	\$TBIT	:	IS IT TIME FOR TRACE TRAP		
036464	100402			BMI	1\$:	BR IF NO		
036466	052716	000020		BIS	#20,(SP)	:	SET TRACE TRAP		
036472	012746	036500	1\$:	MOV	#\$LOOP,-(SP)	:	JUMP TO START OF TEST		
036476	000002		\$RTRN:	RTI		:	RETURN--THIS IS CHANGED TO		
						:	AN 'RTT' IF 'RTT' IS A LEGAL		
						:	INSTRUCTION		
			\$LOOP:			:			
036500				JMP	@(PC)+	:	RETURN		
036500	000137			.WORD	LOOP	:			
036502	004456		\$RTNAD:	.WORD	0	:	'T' BIT STATE INDICATOR		
036504	000000		\$TBIT:	.WORD	0	:	NULL CHARACTER STRING		
036506	377	377	000	SENULL:	.BYTE	-1,-1,0			
				.EVEN					
036512	000000		EPENDS:	.WORD	0	:	LOCATION FOR EOP PRINT FLAG		:DPM002
036514	000000		\$PASS2:	.WORD	0	:	LOCATION FOR PASS COUNT OVERFLOW		:DPM002

6271

.SBTTL SCOPE HANDLER ROUTINE

```
*****  
*THIS ROUTINE CONTROLS THE LOOPING OF SUBTESTS. IT WILL INCREMENT  
*AND LOAD THE TEST NUMBER($STNM) INTO THE DISPLAY REG.(DISPLAY<7:0>)  
*AND LOAD THE ERROR FLAG ($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
```

```
036516 105777 142422 $SCOPE: TSTB @STKS ;IS A CHARACTER WAITING? ;DPM002  
036522 100042 BPL 9000$ ;BRANCH IF NOT ;DPM002  
036524 013737 001146 001244 MOV $TKB,$TMP5 ;WASTE THE CHARACTER, CLEARING READY ;DPM002  
036532 005737 036512 TST EPENDS ;SEE WHICH STATE ENABLE/DISABLE IS IN ;DPM002  
036536 001017 BNE 8000$ ;BRANCH IF EOP'S DISABLED ;DPM002  
036540 005237 036512 INC EPENDS ;SET FLAG DISABLING PRINTOUTS ;DPM002  
036544 104401 036552 TYPE .65$ ;:TYPE ASCIZ STRING  
036550 000411 BR 64$ ;:GET OVER THE ASCIZ  
;:65$: .ASCIZ <CRLF>!EOP'S DISABLED!<CRLF>  
64$:  
036574 000415 BR 9000$ ;BRANCH OVER ENABLE ROUTINE ;DPM002  
036576 005037 036512 8000$: CLR EPENDS ;CLEAR FLAG ENABLING PRINTOUTS ;DPM002  
036602 104401 036610 TYPE .67$ ;:TYPE ASCIZ STRING  
036606 000410 BR 66$ ;:GET OVER THE ASCIZ  
;:67$: .ASCIZ <CRLF>!EOP'S ENABLED!<CRLF>  
66$:  
036630 9000$:  
036630 104407 CKSWR ;:TEST FOR CHANGE IN SOFT-SWR  
036632 032777 040000 142300 1$: BIT #BIT14,@SWR ;:LOOP ON PRESENT TEST?  
036640 001131 BNE $OVER ;:YES IF SW14=1  
;:#####START OF CODE FOR THE XOR TESTER#####  
036642 000416 $XTSTR: BR 6$ ;:IF RUNNING ON THE "XOR" TESTER CHANGE  
;:THIS INSTRUCTION TO A "NOP" (NOP=240)  
036644 013746 000004 MOV @#ERRVEC,-(SP) ;:SAVE THE CONTENTS OF THE ERROR VECTOR  
036650 012737 036670 000004 MOV #5$,@#ERRVEC ;:SET FOR TIMEOUT  
036656 005737 177060 TST @#177060 ;:TIME OUT ON XOR?  
036662 012637 000004 MOV (SP)+,@#ERRVEC ;:RESTORE THE ERROR VECTOR  
036666 000500 BR $SVLAD ;:GO TO THE NEXT TEST  
036670 022626 5$: CMF (SP)+,(SP)+ ;:CLEAR THE STACK AFTER A TIME OUT  
036672 012637 000004 MOV (SP)+,@#ERRVEC ;:RESTORE THE ERROR VECTOR  
036676 000440 BR 7$ ;:LOOP ON THE PRESENT TEST  
036700 6$:;#####END OF CODE FOR THE XOR TESTER#####  
036700 032777 000400 142232 BIT #BIT08,@SWR ;:LOOP ON SPEC. TEST?  
036706 001404 BEQ 2$ ;:BR IF NO  
036710 127737 142224 001102 CMPB @SWR,$STNM ;:ON THE RIGHT TEST? SWR<7:0>  
036716 001502 BEQ $OVER ;:BR IF YES  
036720 013737 177766 037142 2$: MOV 177766,CPSAVE ;:MOVE CPU ERR REG VALUE TO LOC FOR TST ;DPM001  
036726 032737 000001 037142 BIT #BIT00,CPSAVE ;:SEE IF THE POWER MONITOR BIT IS ON ;DPM001  
036734 001406 BEQ 2000$ ;:BRANCH TO CONTINUE ROUTINE IF CLEAR ;DPM001  
036736 042737 000001 177766 BIC #BIT00,177766 ;:CLEAR THE BIT FOUND TO BE SET ;DPM001  
036744 104177 ERROR +177 ;:CALL SPECIAL POWER FAIL BIT ERROR CALL ;DPM001  
036746 105037 001103 CLRB $ERFLG ;:CLEAR THE ERROR FLAG FOR NEXT TEST ;DPM001  
036752 105737 001103 2000$: TSTB $ERFLG ;:HAS AN ERROR OCCURRED?
```



```

036756 001421          BEQ      3$          ;;BR IF NO
036760 123737 001115 001103  CMPB   $ERMAX,$ERFLG  ;;MAX. ERRORS FOR THIS TEST OCCURRED?
036766 101015          BHI     3$          ;;BR IF NO
036770 032777 001000 142142  BIT    #BIT09,@SWR    ;;LOOP ON ERROR?
036776 001404          BEQ     4$          ;;BR IF NO
037000 013737 001110 001106  7$:   MOV    $LPERR,$LPADR  ;;SET LOOP ADDRESS TO LAST SCOPE
037006 000446          BR      $OVER
037010 105037 001103          4$:   CLRB   $ERFLG          ;;ZERO THE ERROR FLAG
037014 005037 001302          CLR    $TIMES         ;;CLEAR THE NUMBER OF ITERATIONS TO MAKE
037020 000415          BR      1$          ;;ESCAPE TO THE NEXT TEST
037022 032777 004000 142110  3$:   BIT    #BIT11,@SWR    ;;INHIBIT ITERATIONS?
037030 001011          BNE    1$          ;;BR IF YES
037032 005737 001324          TST    $PASS         ;;IF FIRST PASS OF PROGRAM
037036 001406          BEQ    1$          ;;      INHIBIT ITERATIONS
037040 005237 001104          INC    $ICNT         ;;INCREMENT ITERATION COUNT
037044 023737 001302 001104  CMP    $TIMES,$ICNT   ;;CHECK THE NUMBER OF ITERATIONS MADE
037052 002024          BGE    $OVER         ;;BR IF MORE ITERATION REQUIRED
037054 012737 000001 001104  1$:   MOV    #1,$ICNT      ;;REINITIALIZE THE ITERATION COUNTER
037062 013737 037140 001302  MOV    $MXCNT,$TIMES  ;;SET NUMBER OF ITERATIONS TO DO
037070 105237 001102          $SVLAD: INCB   $STSTNM     ;;COUNT TEST NUMBERS
037074 113737 001102 001322  MOVB   $STSTNM,$TESTN ;;SET TEST NUMBER IN APT MAILBOX
037102 011637 001106          MOV    (SP),$LPADR    ;;SAVE SCOPE LOOP ADDRESS
037106 011637 001110          MOV    (SP),$LPERR    ;;SAVE ERROR LOOP ADDRESS
037112 005037 001304          CLR    $ESCAPE       ;;CLEAR THE ESCAPE FROM ERROR ADDRESS
037116 112737 000001 001115  MOVB   #1,$ERMAX      ;;ONLY ALLOW ONE(1) ERROR ON NEXT TEST
037124 013777 001102 142010  $OVER: MOV    $STSTNM,@DISPLAY ;;DISPLAY TEST NUMBER
037132 013716 001106          MOV    $LPADR,(SP)   ;;FUDGE RETURN ADDRESS
037136 000002          RTI
037140 000001          $MXCNT: 1          ;;MAX. NUMBER OF ITERATIONS
037142 000000          CPSAVE: .WORD 0    ;;LOCATION TO SAVE CPU ERR REG CONTENTS ;DPM001

```

6273

.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
  
```

```

037144 000000          IBSAVE: .WORD 0          ;LOC'N TO HOLD $ITEMB DURING DUAL ERR ;DPM001
037146 105037 037144 $ERROR: CLRB IBSAVE          ;CLEAR THE ITEM BYTE SAVE LOCATION ;DPM001
037152 104407          CKSWR          ;:TEST FOR CHANGE IN SOFT-SWR
037154 105237 001103 7$: INCB $ERFLG          ;:SET THE ERROR FLAG
037160 001775          BEQ 7$          ;:DON'T LET THE FLAG GO TO ZERO
037162 013777 001102 141752 MOV $TSTNM,@DISPLAY ;:DISPLAY TEST NUMBER AND ERROR FLAG
037170 032777 002000 141742 BIT #BIT10,@SWR          ;:BELL ON ERROR?
037176 001402          BEQ 1$          ;:NO - SKIP
037200 104401 001306          TYPE ,SBELL          ;:RING BELL
037204 005237 001112 1$: INC $ERTTL          ;:COUNT THE NUMBER OF ERRORS
037210 011637 001116          MOV (SP), $ERRPC          ;:GET ADDRESS OF ERROR INSTRUCTION
037214 162737 000002 001116 SUB #2,$ERRPC
037222 117737 141670 001114 MOVB @ $ERRPC,$ITEMB          ;:STRIP AND SAVE THE ERROR ITEM CODE
037230 122737 000177 001114 CMPB #177,$ITEMB          ;:SEE IF THIS IS THE POWER FAIL CALL ;DPM001
037236 001421          BEQ 1000$          ;:BRANCH AROUND PMB TEST IF IT IS ;DPM001
037240 013737 177766 037142 MOV 177766,CPSAVE          ;:MOVE CPU ERR REG TO CPSAVE FOR TEST ;DPM001
037246 032737 000001 037142 BIT #BIT00,CPSAVE          ;:SEE IF POWER MONITOR BIT IS SET ;DPM001
037254 001412          BEQ 1000$          ;:BRANCH IF OK ;DPM001
037256 042737 000001 177766 BIC #BIT00,177766          ;:CLEAR THE BIT FOUND SET ;DPM001
037264 013737 001116 037144 MOV $ERRPC,IBSAVE          ;:SAVE $ERRPC ;DPM001
037272 104177          ERROR +177          ;:CALL SPECIAL POWER MONITOR BIT ERROR ;DPM001
037274 013737 037144 001116 MOV IBSAVE,$ERRPC          ;:RESTORE $ERRPC
037302          1000$:
037302 032777 020000 141630 BIT #BIT13,@SWR          ;:SKIP TYPEOUT IF SET
037310 001004          BNE 20$          ;:SKIP TYPEOUTS
037312 004737 042010          JSR PC,ERTYPE          ;:GO TO USER ERROR ROUTINE
037316 104401 001313          TYPE , $CRLF
037322          20$:
037322 122737 000001 001336 CMFB #APTENV,$ENV          ;:RUNNING IN APT MODE
037330 001007          BNE 2$          ;:NO, SKIP APT ERROR REPORT
037332 113737 001114 037344 MOVB $ITEMB,21$          ;:SET ITEM NUMBER AS ERROR NUMBER
037340 004737 040642          JSR PC,$ATY4          ;:REPORT FATAL ERROR TO APT
037344 000          21$: .BYTE 0
037345 000          .BYTE 0
037346 000777          22$: BR 22$          ;:APT ERROR LOOP
037350 005737 037144 2$: TST IBSAVE          ;:SEE IF POWER FAIL ERROR CALL ;DPM001
037354 001005          BNE 3$          ;:BRANCH IF NOT - HALT NOT ALLOWED ;DPM001
037356 005777 141556          TST @SWR          ;:HALT ON ERROR
037362 100002          BPL 3$          ;:SKIP IF CONTINUE
037364 000000          HALT          ;:HALT ON ERROR!
037366 104407          CKSWR          ;:TEST FOR CHANGE IN SOFT-SWR
037370 032777 001000 141542 3$: BIT #BIT09,@SWR          ;:LOOP ON ERROR SWITCH SET?
037376 001405          BEQ 4$          ;:BR IF NO
  
```

```
037400 005737 037144          TST    IBSAVE          ;SEE IF THIS IS THE PWR MONITOR ERROR ;DPM001
037404 001263                   BNE    7$              ;BRANCH BACK IF SO - NO FUDGING      ;DPM001
037406 013716 001110          MOV    $LPERR,(SP)    ;:FUDGE RETURN FOR LOOPING
037412 005737 001304          4$:   TST    $ESCAPE     ;:CHECK FOR AN ESCAPE ADDRESS
037416 001405                   BEQ    5$              ;:BR IF NONE
037420 005737 037144          TST    IBSAVE          ;SEE IF THIS IS THE PWR MONITOR ERROR ;DPM001
037424 001253                   BNE    7$              ;BRANCH BACK IF SO - NO FUDGING      ;DPM001
037426 013716 001304          MOV    $ESCAPE,(SP)  ;:FUDGE RETURN ADDRESS FOR ESCAPE
037432                                5$:
037432 022737 036432 000042    CMP    #$ENDAD,@#42   ;:ACT-11 AUTO-ACCEPT?
037440 001001                   BNE    6$              ;:BRANCH IF NO
037442 000000                   HALT
037444                                6$:
037444 032777 001000 141466    BIT    #BIT09,@SWR
037452 001013                   BNE    ERM10
037454 011637 001162          MOV    (SP), $REG0    ;SEE IF ERROR #377
037460 062737 177776 001162    ADD    #-2,$REG0
037466 122777 000377 141466    CMPB  #377,@$REG0
037474 001002                   BNE    ERM10
037476 062716 000002          ADD    #2,(SP)
037502 000002          ERM10: RTI
```


6275

.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
 \$SAVREG:

037504			MOV	R0,-(SP)	::PUSH R0 ON STACK
037504	010046		MOV	R1,-(SP)	::PUSH R1 ON STACK
037506	010146		MOV	R2,-(SP)	::PUSH R2 ON STACK
037510	010246		MOV	R3,-(SP)	::PUSH R3 ON STACK
037512	010346		MOV	R4,-(SP)	::PUSH R4 ON STACK
037514	010446		MOV	R5,-(SP)	::PUSH R5 ON STACK
037516	010546		MOV	22(SP),-(SP)	::SAVE PS OF MAIN FLOW
037520	016646	000022	MOV	22(SP),-(SP)	::SAVE PC OF MAIN FLOW
037524	016646	000022	MOV	22(SP),-(SP)	::SAVE PS OF CALL
037530	016646	000022	MOV	22(SP),-(SP)	::SAVE PC OF CALL
037534	016646	000022	RTI		
037540	000002				

*RESTORE R0-R5

*CALL:

* RESREG

\$RESREG:

037542			MOV	(SP)+,22(SP)	::RESTORE PC OF CALL
037542	012666	000022	MOV	(SP)+,22(SP)	::RESTORE PS OF CALL
037546	012666	000022	MOV	(SP)+,22(SP)	::RESTORE PC OF MAIN FLOW
037552	012666	000022	MOV	(SP)+,22(SP)	::RESTORE PS OF MAIN FLOW
037556	012666	000022	MOV	(SP)+,R5	::POP STACK INTO R5
037562	012605		MOV	(SP)+,R4	::POP STACK INTO R4
037564	012604		MOV	(SP)+,R3	::POP STACK INTO R3
037566	012603		MOV	(SP)+,R2	::POP STACK INTO R2
037570	012602		MOV	(SP)+,R1	::POP STACK INTO R1
037572	012601		MOV	(SP)+,R0	::POP STACK INTO R0
037574	012600		RTI		
037576	000002				

6277

```

.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
*
037600 105737 001157 $TYPE: TSTB $/PFLG ;;IS THERE A TERMINAL?
037604 100002 BPL 1$ ;;BR IF YES
037606 000000 HALT ;;HALT HERE IF NO TERMINAL
037610 000430 BR 3$ ;;LEAVE
037612 010046 1$: MOV R0,-(SP) ;;SAVE R0
037614 017600 000002 MOV @2(SP),R0 ;;GET ADDRESS OF ASCIZ STRING
037620 122737 000001 001336 CMPB #APTENV,$ENV ;;RUNNING IN APT MODE
037626 001011 BNE 62$ ;;NO,GO CHECK FOR APT CONSOLE
037630 132737 000100 001337 BITB #APTSPOOL,$ENVM ;;SPOOL MESSAGE TO APT
037636 001405 BEQ 62$ ;;NO,GO CHECK FOR CONSOLE
037640 010037 037650 MOV R0,61$ ;;SETUP MESSAGE ADDRESS FOR APT
037644 004737 040632 JSR PC,$ATY3 ;;SPOOL MESSAGE TO APT
037650 000000 61$: .WORD 0 ;;MESSAGE ADDRESS
037652 132737 000040 001337 62$: BITB #APTCSUP,$ENVM ;;APT CONSOLE SUPPRESSED
037660 001003 BNE 60$ ;;YES,SKIP TYPE OUT
037662 112046 2$: MOVB (R0)+,-(SP) ;;PUSH CHARACTER TO BE TYPED ONTO STACK
037664 001005 BNE 4$ ;;BR IF IT ISN'T THE TERMINATOR
037666 005726 TST (SP)+ ;;IF TERMINATOR POP IT OFF THE STACK
037670 012600 60$: MOV (SP)+,R0 ;;RESTORE R0
037672 062716 000002 3$: ADD #2,(SP) ;;ADJUST RETURN PC
037676 000002 RTI ;;RETURN
037700 122716 000011 4$: CMPB #HT,(SP) ;;BRANCH IF <HT>
037704 001430 BEQ 8$
037706 122716 000200 CMPB #CRLF,(SP) ;;BRANCH IF NOT <CRLF>
037712 001006 BNE 5$
037714 005726 TST (SP)+ ;;POP <CR><LF> EQUIV
037716 104401 TYPE ;;TYPE A CR AND LF
037720 001313 $CRLF
037722 105037 040140 CLRB $CHARCNT ;;CLEAR CHARACTER COUNT
037726 000755 BR 2$ ;;GET NEXT CHARACTER
037730 004737 040012 5$: JSR PC,$TYPEC ;;GO TYPE THIS CHARACTER
037734 123726 001156 6$: CMPB $FILLC,(SP)+ ;;IS IT TIME FOR FILLER CHARS.?
037740 001350 BNE 2$ ;;IF NO GO GET NEXT CHAR.
037742 013746 001154 MOV $NULL,-(SP) ;;GET # OF FILLER CHARS. NEEDED
;;AND THE NULL CHAR.
037746 105366 000001 7$: DECB 1(SP) ;;DOES A NULL NEED TO BE TYPED?
037752 002770 BLT 6$ ;;BR IF NO--GO POP THE NULL OFF OF STACK
037754 004737 040012 JSR PC,$TYPEC ;;GO TYPE A NULL
037760 105337 040140 DECB $CHARCNT ;;DO NOT COUNT AS A COUNT
037764 000770 BR 7$ ;;LOOP
:HORIZONTAL TAB PROCESSOR
037766 112716 000040 8$: MOVB #' ,(SP) ;;REPLACE TAB WITH SPACE

```

```

037772 004737 040012 9$: JSR PC,$TYPEC ::TYPE A SPACE
037776 132737 000007 040140 BITB #7,$CHARCNT ::BRANCH IF NOT AT
040004 001372 BNE 9$ ::TAB STOP
040006 005726 TST (SP)+ ::POP SPACE OFF STACK
040010 000724 BR 2$ ::GET NEXT CHARACTER
040012 $TYPEC:
040012 105777 141126 TSTB @STKS ::CHAR IN KYBD BUFFER? :MJD001
040016 100022 BPL 10$ ::BR IF NOT :MJD001
040020 017746 141122 MOV @STKB,-(SP) ::GET CHAR :MJD001
040024 042716 177600 BIC #177600,(SP) ::STRIP EXTRANEIOUS BITS :MJD001
040030 122716 000023 CMPB #$XOFF,(SP) ::WAS CHAR XOFF :MJD001
040034 001012 BNE 102$ ::BR IF NOT :MJD001
040036 105777 141102 101$: TSTB @STKS ::WAIT FOR CHAR :MJD001
040042 100375 BPL 101$ :MJD001
040044 117716 141076 MOVB @STKB,(SP) ::GET CHAR :MJD001
040050 042716 177600 BIC #177600,(SP) ::STRIP IT :MJD001
040054 122716 000021 CMPB #$XON,(SP) ::WAS IT XON? :MJD001
040060 001366 BNE 101$ ::BR IF NOT :MJD001
040062 102$: TST (SP)+ ::FIX STACK :MJD001
040064 105777 141060 10$: TSTB @STPS ::WAIT UNTIL PRINTER IS READY :MJD001
040070 100375 BPL 10$ :MJD001
040072 126627 000002 000021 CMPB 2(SP),#$XON ::IS CHARACTER A RANDOM XON? :MJD001
040100 001420 BEQ $TYPEX ::BRANCH IF YES :RAN001
040102 116677 000002 141042 MOVB 2(SP),@STPB ::LOAD CHAR TO BE TYPED INTO DATA REG. :RAN001
040110 122766 000015 000002 CMPB #CR,2(SP) ::IS CHARACTER A CARRIAGE Peturn?
040116 001003 BNE 1$ ::BRANCH IF NO
040120 105037 040140 CLRB $CHARCNT ::YES--CLEAR CHARACTER COUNT
040124 000406 BR $TYPEX ::EXIT
040126 122766 000012 000002 1$: CMPB #LF,2(SP) ::IS CHARACTER A LINE FEED?
040134 001402 BEQ $TYPEX ::BRANCH IF YES
040136 105227 INCB (PC)+ ::COUNT THE CHARACTER
040140 000000 $CHARCNT: .WORD 0 ::CHARACTER COUNT STORAGE
040142 000207 $TYPEX: RTS PC

```


6279

.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

```

040144	017646	000000		\$TYPOS:	MOV	@(SP),-(SP)	;;PICKUP THE MODE
040150	116637	000001	040375		MOV	1(SP), \$OFILL	;;LOAD ZERO FILL SWITCH
040156	112637	040377			MOV	(SP)+, \$OMODE+1	;;NUMBER OF DIGITS TO TYPE
040162	062716	000002			ADD	#2, (SP)	;;ADJUST RETURN ADDRESS
040166	000406				BR	\$TYPON	
040170	112737	000001	040375	\$TYPOC:	MOV	#1, \$OFILL	;;SET THE ZERO FILL SWITCH
040176	112737	000006	040377		MOV	#6, \$OMODE+1	;;SET FOR SIX(6) DIGITS
040204	112737	000005	040374	\$TYPON:	MOV	#5, \$OCNT	;;SET THE ITERATION COUNT
040212	010346				MOV	R3, -(SP)	;;SAVE R3
040214	010446				MOV	R4, -(SP)	;;SAVE R4
040216	010546				MOV	R5, -(SP)	;;SAVE R5
040220	113704	040377			MOV	\$OMODE+1, R4	;;GET THE NUMBER OF DIGITS TO TYPE
040224	005404				NEG	R4	
040226	062704	000006			ADD	#6, R4	;;SUBTRACT IT FOR MAX. ALLOWED
040232	110437	040376			MOV	R4, \$OMODE	;;SAVE IT FOR USE
040236	113704	040375			MOV	\$OFILL, R4	;;GET THE ZERO FILL SWITCH
040242	016605	000012			MOV	12(SP), R5	;;PICKUP THE INPUT NUMBER
040246	005003				CLF	R3	;;CLEAR THE OUTPUT WORD
040250	006105			1\$:	ROL	R5	;;ROTATE MSB INTO 'C'
040252	000404				BR	3\$;;GO DO MSB
040254	006105			2\$:	ROL	R5	;;FORM THIS DIGIT
040256	006105				ROL	R5	
040260	006105				ROL	R5	
040262	010503				MOV	R5, R3	
040264	006103			3\$:	ROL	R3	;;GET LSB OF THIS DIGIT
040266	105337	040376			DECB	\$OMODE	;;TYPE THIS DIGIT?
040272	100021				BPL	7\$;;BR IF NO
040274	042703	177770			BIC	#177770, R3	;;GET RID OF JUNK
040300	001002				BNE	4\$;;TEST FOR 0
040302	005704				TST	R4	;;SUPPRESS THIS 0?
040304	001403				BEQ	5\$;;BR IF YES
040306	005204			4\$:	INC	R4	;;DON'T SUPPRESS ANYMORE 0'S

040310	052703	000060		BIS	#'0,R3	::MAKE THIS DIGIT ASCII	
040314	052703	000040	5\$:	BIS	#' ,R3	::MAKE ASCII IF NOT ALREADY	
040320	122703	000040		CMPB	#' ,R3	::IS THIS A SPACE CHARACTER?	:DPM002
040324	001404			BEQ	7\$::BRANCH IF SO - DON'T TYPE	:DPM002
040326	110337	040372		MOVB	R3,8\$::SAVE FOR TYPING	
040332	104401	040372		TYPE	,8\$::GO TYPE THIS DIGIT	
040336	105337	040374	7\$:	DECB	\$OCNT	::COUNT BY 1	
040342	003344			BGT	2\$::BR IF MORE TO DO	
040344	002402			BLT	6\$::BR IF DONE	
040346	005204			INC	R4	::INSURE LAST DIGIT ISN'T A BLANK	
040350	000741			BR	2\$::GO DO THE LAST DIGIT	
040352	012605		6\$:	MOV	(SP)+,R5	::RESTORE R5	
040354	012604			MOV	(SP)+,R4	::RESTORE R4	
040356	012603			MOV	(SP)+,R3	::RESTORE R3	
040360	016666	000002 000004		MOV	2(SP),4(SP)	::SET THE STACK FOR RETURNING	
040366	012616			MOV	(SP)+,(SP)		
040370	000002			RTI		::RETURN	
040372	000		8\$:	.BYTE	0	::STORAGE FOR ASCII DIGIT	
040373	000			.BYTE	0	::TERMINATOR FOR TYPE ROUTINE	
040374	000		\$OCNT:	.BYTE	0	::OCTAL DIGIT COUNTER	
040375	000		\$OFILL:	.BYTE	0	::ZERO FILL SWITCH	
040376	000000		\$OMODE:	.WORD	0	::NUMBER OF DIGITS TO TYPE	

6281

```

.SBTT: 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
$TYPDS:
MOV     R0,-(SP)           ;;PUSH R0 ON STACK
MOV     R1,-(SP)           ;;PUSH R1 ON STACK
MOV     R2,-(SP)           ;;PUSH R2 ON STACK
MOV     R3,-(SP)           ;;PUSH R3 ON STACK
MOV     R5,-(SP)           ;;PUSH R5 ON STACK
MOV     #20200,-(SP)       ;;SET BLANK SWITCH AND SIGN
MOV     20(SP),R5          ;;GET THE INPUT NUMBER
BPL     1$                 ;;BR IF INPUT IS POS.
NEG     R5                 ;;MAKE THE BINARY NUMBER POS.
MOVB   #'-,1(SP)          ;;MAKE THE ASCII NUMBER NEG.
1$:    CLR     RC           ;;ZERO THE CONSTANTS INDEX
MOV     #SDBLK,R3         ;;SETUP THE OUTPUT POINTER
MOVB   #' ,(R3)+          ;;SET THE FIRST CHARACTER TO A BLANK
2$:    CLR     R2           ;;CLEAR THE BCD NUMBER
MOV     $DTBL(R0),R1      ;;GET THE CONSTANT
3$:    SUB     R1,R5        ;;FORM THIS BCD DIGIT
BLT     4$                 ;;BR IF DONE
INC     R2                 ;;INCREASE THE BCD DIGIT BY 1
BR      3$
4$:    ADD     R1,R5        ;;ADD BACK THE CONSTANT
TST     R2                 ;;CHECK IF BCD DIGIT=0
BNE     5$                 ;;FALL THROUGH IF 0
TSTB   (SP)               ;;STILL DOING LEADING 0'S?
BMI     7$                 ;;BR IF YES
5$:    ASLB   (SP)          ;;MSD?
BCC     6$                 ;;BR IF NO
MOVB   1(SP),-1(R3)       ;;YES--SET THE SIGN
6$:    BIS     #'0,R2       ;;MAKE THE BCD DIGIT ASCII
7$:    BIS     #' ,R2       ;;MAKE IT A SPACE IF NOT ALREADY A DIGIT
MOVB   R2,(R3)+          ;;PUT THIS CHARACTER IN THE OUTPUT BUFFER
TST     (R0)+             ;;JUST INCREMENTING
CMF     R0,#10            ;;CHECK THE TABLE INDEX
BLT     2$                 ;;GO DO THE NEXT DIGIT
BGT     8$                 ;;GO TO EXIT
MOV     R5,R2             ;;GET THE LSD
BR      6$                 ;;GO CHANGE TO ASCII
8$:    TSTB   (SP)+        ;;WAS THE LSD THE FIRST NON-ZERO?
BPL     9$                 ;;BR IF NO
MOVB   -1(SP),-2(R3)     ;;YES--SET THE SIGN FOR TYPING
9$:    CLRB   (R3)         ;;SET THE TERMINATOR
MOV     (SP)+,R5         ;;POP STACK INTO R5
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
040400 010046 020200 000020 040614 040604 040614 040614 040614
040400 010046 020200 000020 040614 040604 040614 040614 040614
040402 010146 020200 000020 040614 040604 040614 040614 040614
040404 010246 020200 000020 040614 040604 040614 040614 040614
040406 010346 020200 000020 040614 040604 040614 040614 040614
040410 010546 020200 000020 040614 040604 040614 040614 040614
040412 012746 020200 000020 040614 040604 040614 040614 040614
040416 016605 000020 000055 000001 040614 040604 040614 040614
040422 100004 000055 000001 040614 040604 040614 040614 040614
040424 005405 000055 000001 040614 040604 040614 040614 040614
040426 112766 000055 000001 040614 040604 040614 040614 040614
040434 005000 040614 040604 040614 040604 040614 040614 040614
040436 012703 040614 040604 040614 040604 040614 040614 040614
040442 112723 000040 040614 040604 040614 040604 040614 040614
040446 005002 040614 040604 040614 040604 040614 040604 040614
040450 016001 040604 040614 040604 040614 040604 040614 040614
040454 160105 040614 040604 040614 040604 040614 040604 040614
040456 002402 040614 040604 040614 040604 040614 040604 040614
040460 005202 040614 040604 040614 040604 040614 040604 040614
040462 000774 040614 040604 040614 040604 040614 040604 040614
040464 060105 040614 040604 040614 040604 040614 040604 040614
040466 005702 040614 040604 040614 040604 040614 040604 040614
040470 001002 040614 040604 040614 040604 040614 040604 040614
040472 105716 040614 040604 040614 040604 040614 040604 040614
040474 100407 040614 040604 040614 040604 040614 040604 040614
040476 106316 040614 040604 040614 040604 040614 040604 040614
040500 103003 040614 040604 040614 040604 040614 040604 040614
040502 116663 000001 177777 040614 040604 040614 040604 040614
040510 052702 000060 040614 040604 040614 040604 040614 040614
040514 052702 000040 040614 040604 040614 040604 040614 040614
040520 110223 040614 040604 040614 040604 040614 040604 040614
040522 005720 040614 040604 040614 040604 040614 040604 040614
040524 020027 000010 040614 040604 040614 040604 040614 040614
040530 002746 040614 040604 040614 040604 040614 040604 040614
040532 003002 040614 040604 040614 040604 040614 040604 040614
040534 010502 040614 040604 040614 040604 040614 040604 040614
040536 000764 040614 040604 040614 040604 040614 040604 040614
040540 105726 040614 040604 040614 040604 040614 040604 040614
040542 100003 040614 040604 040614 040604 040614 040604 040614
040544 116663 177777 177776 040614 040604 040614 040604 040614
040552 105013 040614 040604 040614 040604 040614 040604 040614
040554 012605 040614 040604 040614 040604 040614 040604 040614
040556 012603 040614 040604 040614 040604 040614 040604 040614
040560 012602 040614 040604 040614 040604 040614 040604 040614
040562 012601 040614 040604 040614 040604 040614 040604 040614
040564 012600 040614 040604 040614 040604 040614 040604 040614
040566 104401 040614 040614 040614 040614 040614 040614 040614
    
```



```
040572 016666 000002 000004      MOV      2(SP),4(SP)      ;;ADJUST THE STACK
040600 012616                      MOV      (SP)+,(SP)
040602 000002                      RTI                          ;;RETURN TO USER
040604 023420      $DTBL: 10000.
040606 001750                      1000.
040610 000144                      100.
040612 000012                      10.
040614                      $DBLK: .BLKW 4
```

6283

.SBTTL APT COMMUNICATIONS ROUTINE

```

040624 112737 000001 041070 $ATY1: MOVB #1,$FFLG ::TO REPORT FATAL ERROR
040632 112737 000001 041066 $ATY3: MOVB #1,$MFLG ::TO TYPE A MESSAGE
040640 000403
040642 112737 000001 041070 $ATY4: MOVB #1,$FFLG ::TO ONLY REPORT FATAL ERROR
040650 $ATYC:
040650 010046 MOV R0,-(SP) ::PUSH R0 ON STACK
040652 010146 MOV R1,-(SP) ::PUSH R1 ON STACK
040654 105737 041066 TSTB $MFLG ::SHOULD TYPE A MESSAGE?
040660 001450 BEQ 5$ ::IF NOT: BR
040662 122737 000001 001336 CMPB #APTENV,$ENV ::OPERATING UNDER APT?
040670 001031 BNE 3$ ::IF NOT: BR
040672 132737 000100 001337 BITB #APTSPOOL,$ENVM ::SHOULD SPOOL MESSAGES?
040700 001425 BEQ 3$ ::IF NOT: BR
040702 017600 000004 MOV @4(SP),R0 ::GET MESSAGE ADDR.
040706 062766 000002 000004 ADD #2,4(SP) ::BUMP RETURN ADDR.
040714 005737 001316 1$: TST $MSGTYPE ::SEE IF DONE W/ LAST XMISSION?
040720 001375 BNE 1$ ::IF NOT: WAIT
040722 010037 001332 MOV R0,$MSGAD ::PUT ADDR IN MAILBOX
040726 105720 2$: TSTB (R0)+ ::FIND END OF MESSAGE
040730 001376 BNE 2$
040732 163700 001332 SUB $MSGAD,R0 ::SUB START OF MESSAGE
040736 006700 ASR R0 ::GET MESSAGE LNTH IN WORDS
040740 010037 001334 MOV R0,$MSGGLT ::PUT LENGTH IN MAILBOX
040744 012737 000004 001316 MOV #4,$MSGTYPE ::TELL APT TO TAKE MSG.
040752 000413 BR 5$
040754 017637 000004 041000 3$: MOV @4(SP),4$ ::PUT MSG ADDR IN JSR LINKAGE
040762 062766 000002 000004 ADD #2,4(SP) ::BUMP RETURN ADDRESS
040770 013746 177776 MOV 177776,-(SP) ::PUSH 177776 ON STACK
040774 004737 037600 JSR PC,$TYPE ::CALL TYPE MACRO
041000 000000 4$: .WORD 0
041002 5$:
041002 105737 041070 10$: TSTB $FFLG ::SHOULD REPORT FATAL ERROR?
041006 001416 BEQ 12$ ::IF NOT: BR
041010 005737 001336 TST $ENV ::RUNNING UNDER APT?
041014 001413 BEQ 12$ ::IF NOT: BR
041016 005737 001316 11$: TST $MSGTYPE ::FINISHED LAST MESSAGE?
041022 001375 BNE 11$ ::IF NOT: WAIT
041024 017637 000004 001320 MOV @4(SP),$FATAL ::GET ERROR #
041032 062766 000002 000004 ADD #2,4(SP) ::BUMP RETURN ADDR.
041040 005237 001316 INC $MSGTYPE ::TELL APT TO TAKE ERROR
041044 105037 041070 12$: CLFB $FFLG ::CLEAR FATAL FLAG
041050 105037 041067 CLRFB $LFLG ::CLEAR LOG FLAG
041054 105037 041066 CLRFB $MFLG ::CLEAR MESSAGE FLAG
041060 012601 MOV (SP)+,R1 ::POP STACK INTO R1
041062 012600 MOV (SP)+,R0 ::POP STACK INTO R0
041064 000207 RTS PC ::RETURN
041066 000 $MFLG: .BYTE 0 ::MESSG. FLAG
041067 000 $LFLG: .BYTE 0 ::LOG FLAG
041070 000 $FFLG: .BYTE 0 ::FATAL FLAG
.EVEN
000200 APTSIZE=200
000001 APTENV=001
000100 APTSPOOL=100
000040 APTCSUP=040

```

6285

```

.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.

041072 022737 000176 001140 $CKSWR: CMP #SWREG,SWR ;;IS THE SOFT-SWR SELECTED?
041100 031074 BNE 15$ ;;BRANCH IF NO
041102 105777 140036 TSTB @STKS ;;CHAR THERE?
041106 100071 BPL 15$ ;;IF NO, DON'T WAIT AROUND
041110 117746 140032 MOVB @STKB,-(SP) ;;SAVE THE CHAR
041114 042716 177600 BIC #^C177,(SP) ;;STRIP-OFF THE ASCII
041120 022726 000007 CMP #7,(SP)+ ;;IS IT A CONTROL G?
041124 001062 BNE 15$ ;;NO, RETURN TO USER
041126 123727 001134 000001 CMPB $AUTOB,#1 ;;ARE WE RUNNING IN AUTO-MODE?
041134 001456 BEQ 15$ ;;BRANCH IF YES
041136 104401 041511 TYPE ,SCNTLG ;;ECHO THE CONTROL-G (^G)
041142 104401 041516 $GTSWR: TYPE ,SMSWR ;;TYPE CURRENT CONTENTS
041146 013746 000176 MOV SWREG,-(SP) ;;SAVE SWREG FOR TYPEOUT
041152 104402 TYPOC ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
041154 104401 041527 TYPE ,SMNEW ;;PROMPT FOR NEW SWR
041160 005046 19$: CLR -(SP) ;;CLEAR COUNTER
041162 005046 CLR -(SP) ;;THE NEW SWR
041164 105777 137754 7$: TSTB @STKS ;;CHAR THERE?
041170 100375 BPL 7$ ;;IF NOT TRY AGAIN
041172 117746 137750 MOVB @STKB,-(SP) ;;PICK UP CHAR
041176 042716 177600 BIC #^C177,(SP) ;;MAKE IT 7-BIT ASCII
041202 021627 000025 9$: CMP (SP),#25 ;;IS IT A CONTROL-U?
041206 001005 BNE 10$ ;;BRANCH IF NOT
041210 104401 041504 TYPE ,SCNTLU ;;YES, ECHO CONTROL-U (^U)
041214 062706 000006 20$: ADD #6,SP ;;IGNORE PREVIOUS INPUT
041220 000757 BR 19$ ;;LET'S TRY IT AGAIN
041222 021627 000015 10$: CMP (SP),#15 ;;IS IT A <CR>?
041226 001022 BNE 16$ ;;BRANCH IF NO
041230 005766 000004 TST 4(SP) ;;YES, IS IT THE FIRST CHAR?
041234 001403 BEQ 11$ ;;BRANCH IF YES
041236 016677 000002 137674 MOV 2(SP),@SWR ;;SAVE NEW SWR
041244 062706 000006 11$: ADD #6,SP ;;CLEAR UP STACK
041250 104401 001313 14$: TYPE ,SCRLF ;;ECHO <CR> AND <LF>
041254 123727 001135 000001 CMPB $INTAG,#1 ;;RE-ENABLE TTY KBD INTERRUPTS?
041262 001003 BNE 15$ ;;BRANCH IF NOT
041264 012777 000100 137652 MOV #100,@STKS ;;RE-ENABLE TTY KBD INTERRUPTS
041272 000002 15$: RTI ;;RETURN
041274 004737 040012 16$: JSR PC,$TYPEC ;;ECHO CHAR
041300 021627 000060 CMP (SP),#60 ;;CHAR < 0?
041304 002420 BLT 18$ ;;BRANCH IF YES
041306 021627 000067 CMP (SP),#67 ;;CHAR > 7?
041312 003015 BGT 18$ ;;BRANCH IF YES
041314 042726 000060 BIC #60,(SP)+ ;;STRIP-OFF ASCII
041320 005766 000002 TST 2(SP) ;;IS THIS THE FIRST CHAR
041324 001403 BEQ 17$ ;;BRANCH IF YES
041326 006316 ASL (SP) ;;NO, SHIFT PRESENT
041330 006316 ASL (SP) ;; CHAR OVER TO MAKE
041332 006316 ASL (SP) ;; ROOM FOR NEW ONE.
041334 005266 000002 .17$: INC 2(SP) ;;KEEP COUNT OF CHAR

```


041340 056616 177776
041344 000707
041346 104401 001312
041352 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

041354 011646
041356 016666 000004 000002
041364 105777 137554
041370 100375
041372 117766 137550 000004
041400 042766 177600 000004
041406 026627 000004 000023
041414 001013
041416 105777 137522
041422 100375
041424 117746 137516
041430 042716 177600
041434 022627 000021
041440 001366
041442 000750
041444 026627 000004 000021
041452 001744
041454 026627 000004 000140
041462 002407
041464 026627 000004 000175
041472 003003
041474 042766 000040 000004
041502 000002
041504 136 125 015
041511 136 107 015
041516 015 012 123
041527 040 040 116

\$RDCHR: MOV (SP), -(SP) ::PUSH DOWN THE PC
MOV 4(SP), 2(SP) ::SAVE THE PS
1\$: TSTB @STKS ::WAIT FOR
BPL 1\$::A CHARACTER
MOV 4(SP), 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 @STKS ::WAIT FOR A CHARACTER
BPL 2\$::LOOP UNTIL ITS THERE
MOV 4(SP), -(SP) ::GET CHARACTER
BIC #^C177, (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
4\$: RTI ::GO BACK TO USER
\$CNTLU: .ASCIZ /*U/<15><12> ::CONTROL 'U'
\$CNTLG: .ASCIZ /*G/<15><12> ::CONTROL 'G'
\$MSWR: .ASCIZ <15><12>/SWR = /
\$MNEW: .ASCIZ / NEW = /

:RAN001
:RAN001

6287

041540	010046	
041542	016600	000002
041546	005740	
041550	111000	
041552	006300	
041554	016000	041574
041560	000200	

041562	011646		
041564	016666	000004	000002
041572	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    R0,-(SP)           ;;SAVE R0
        MOV    2(SP),R0         ;;GET TRAP ADDRESS
        TST    -(R0)           ;;BACKUP BY 2
        MOVB   (R0),R0         ;;GET RIGHT BYTE OF TRAP
        ASL    R0              ;;POSITION FOR INDEXING
        MOV    $TRPAD(R0),R0    ;;INDEX TO TABLE
        RTS    R0              ;;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
:-----

```

041574	041562
041576	037600
041600	040170
041602	040144
041604	040204
041606	040400
041610	041142
041612	041072
041614	041354
041616	037504
041620	037542
6288	041622
6289	042622
	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

```

6291

.SBTTL POWER DOWN AND UP ROUTINES

```

*****
:POWER DOWN ROUTINE
041624 012737 042002 000024 $PWRDN: MOV $ILLUP,@#PWRVEC ;;SET FOR FAST UP
041632 012737 000340 000026 MOV #340,@#PWRVEC+2 ;;PRIO:7
041640 010046 MOV R0,-(SP) ;;PUSH R0 ON STACK
041642 010146 MOV R1,-(SP) ;;PUSH R1 ON STACK
041644 010246 MOV R2,-(SP) ;;PUSH R2 ON STACK
041646 010346 MOV R3,-(SP) ;;PUSH R3 ON STACK
041650 010446 MOV R4,-(SP) ;;PUSH R4 ON STACK
041652 010546 MOV R5,-(SP) ;;PUSH R5 ON STACK
041654 017746 137260 MOV @SWR,-(SP) ;;PUSH @SWR ON STACK
041660 010637 042006 MOV SP,$SAVR6 ;;SAVE SP
041664 012737 041676 000024 MOV #PWRUP,@#PWRVEC ;;SET UP VECTOR
041672 000000 HALT
041674 000776 BR -2 ;;HANG UP
*****
:POWER UP ROUTINE
041676 012737 042002 000024 $PWRUP: MOV $ILLUP,@#PWRVEC ;;SET FOR FAST DOWN
041704 013706 042006 MOV $SAVR6,SP ;;GET SP
041710 005037 042006 CLR $SAVR6 ;;WAIT LOOP FOR THE TTY
041714 005237 042006 1$: INC $SAVR6 ;;WAIT FOR THE INC
041720 001375 BNE 1$ ;;OF WORD
041722 012677 137212 MOV (SP)+,@SWR ;;POP STACK INTO @SWR
041726 012605 MOV (SP)+,R5 ;;POP STACK INTO R5
041730 012604 MOV (SP)+,R4 ;;POP STACK INTO R4
041732 012603 MOV (SP)+,R3 ;;POP STACK INTO R3
041734 012602 MOV (SP)+,R2 ;;POP STACK INTO R2
041736 012601 MOV (SP)+,R1 ;;POP STACK INTO R1
041740 012600 MOV (SP)+,R0 ;;POP STACK INTO R0
041742 012737 041624 000024 MOV #PWRDN,@#PWRVEC ;;SET UP THE POWER DOWN VECTOR
041750 012737 000340 000026 MOV #340,@#PWRVEC+2 ;;PRIO:7
041756 104401 TYPE ;;REPORT THE POWER FAILURE
041760 042672 $PWRMG: .WORD POWERM ;;POWER FAIL MESSAGE POINTER
041762 012716 MOV (PC)+,(SP) ;;RESTART AT START
041764 003616 $PWRAD: .WORD START ;;RESTART ADDRESS
041766 042766 000020 000002 BIC #20,2(SP) ;;CLEAR 'T' BIT
041774 005037 036504 CLR $TBIT ;;CLEAR THE 'T' BIT FLAG
042000 000002 RTI
042002 000000 $ILLUP: HALT ;;THE POWER UP SEQUENCE WAS STARTED
042004 000776 BR -2 ;; BEFORE THE POWER DOWN WAS COMPLETE
042006 000000 $SAVR6: 0 ;;PUT THE SP HERE

```


6293
 6294
 6295
 6296
 6297
 6298
 6299
 6300
 6301 042010 104401 001313
 6302 042014 113737 001102 001232
 6303 042022 042737 177400 001232
 6304 042030 013737 001116 001234
 6305 042036 010046
 6306
 6307 042040 113700 001114
 6308 042044 042700 177400
 6309 042050 001005
 6310
 6311 042052 013746 001116
 6312 042056 104402
 6313 042060 000137 042406
 6314
 6315 042064 022700 000377
 6316 042070 001005
 6317 042072 016600 000004
 6318 042076 011000
 6319 042100 062700 000400
 6320 042104 010037 001320
 6321 042110 005300
 6322 042112 006300
 6323 042114 006300
 6324 042116 006300
 6325 042120 062700 001442
 6326
 6327 042124 012037 042134
 6328 042130 001404
 6329 042132 104401
 6330 042134 000000
 6331 042136 104401
 6332 042140 001313
 6333
 6334 042142 012037 042152
 6335 042146 001404
 6336 042150 104401
 6337 042152 000000
 6338 042154 104401
 6339 042156 001313
 6340
 6341 042160 010146
 6342 042162 010246
 6343 042164 010346
 6344
 6345 042166 012001
 6346 042170 001501
 6347
 6348 042172 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 $EPROR 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
        MOV      $STNM,$TMP0
        BIC      #177400,$TMP0
        MOV      $ERRPC,$TMP1      ;GET PC OF CALL
        MOV      RO,-(SP)          ;SAVE RO
        MOV      $ITEMB,RO          ;GET THE ITEM NUMBER.
        BIC      #177400,RO        ;CLEAR UPPER BYTE EXPOSING OFFSET
        BNE      1$                ;BRANCH IF OFFSET EXISTS
        MOV      $ERRPC,-(SP)      ;IF ZERO THEN JUST
        TYPOC   ERT5              ;PRINT THE PC
        JMP
1$:     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$:   MOV      (RO)+,2$          ;PICK UP THE ADDRESS OF THE EM, ERROR MESSAGE
        BEQ      3$                ;BRANCH IF NONE
        TYPE
2$:     .WORD    0
        TYPE
        .WORD    $CRLF
3$:     MOV      (RO)+,4$          ;GET THE DH,DATA HEADER
        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          ;GET THE ADDRESS OF THE DATA TABLE
        BEQ      ERT4              ;RETURN IF NO DATA.
        MOV      (RO),RO          ;GET A POINTER TO THE DATA FORMAT TABLE
  
```



```
6406 042356 000000      17S:  HALT                ;UNDEFINED FORMAT FOR DATA?????
6407
6408 042360 104401      ERT2:  TYPE                ;PRINT A TAB AFTER TYPING
6409 042362 042740      .WORD  STAB                ;AN DATA TABLE ENTRY
6410
6411
6412
6413 042364 005200      ERT3:  INC      R0          ;POINT TO THE NEXT FORMAT
6414 042366 005711      TST      (R1)             ;END OF DATA TABLE.
6415 042370 001401      BEQ      ERT4
6416 042372 000700      BR       ERT1
6417
6418 042374 104401      ERT4:  TYPE                ;DONE.
6419 042376 001313      .WORD  $CRLF
6420 042400 012603      MOV      (SP)+,R3        ;RESTORE R1,R2 AND R3
6421 042402 012602      MOV      (SP)+,R2
6422 042404 012601      MOV      (SP)+,R1
6423 042406 012600      ERT5:  MOV      (SP)+,R0  ;RESTORE R0.
6424 042410 000207      RTS      PC              ;AND RETURN.
6425
6426
6427 042412 042422 042456 042506 PFECWS: .WORD  PFECM,PFECDH,PFECAD,PFECFT ;ADRSES OF DATA/ASCII BELOW ;DPM001
6428 042422      120      117      127 PFECM: .ASCIZ ?POWER MONITOR BIT FOUND SET? ;ERROR MESSAGE ;DPM001
6429 042456      124      105      123 PFECDH: .ASCIZ ?TESTNO ERR PC CPUERR? ;ERROR DATA HEADER ;DPM001
6430
6431 042506 001322 001116 037142 PFECAD: .WORD  $TESTN,$ERRPC,CPSAVE,0 ;ADDRESSES OF DATA ;DPM001
6432 042516      000      000      000 PFECFT: .BYTE  0,0,0,0 ;FORMAT TABLE ;DPM001
6433
6434 042522 012246
6435 042524 104402
6436 042526 104401 042742
6437 042532 000207
TOCTNM: MOV      (R2)+,-(SP) ;MOVE THE NUMBER TO THE STACK FOR PRINTING
        TYPOC ;TYPE AN OCTAL NUMBER
        TYPE  ,SPACE ;TYPE A SPACE CHARACTER
        RTS   PC ;EXIT BACK
```


6438
6439
6440
6441
6442
6443
6444 042534 011637 001236
6445 042540 022626
6446 042542 170200
6447 042544 010037 001240
6448 042550 170300
6449 042552 010037 001242
6450 042556 104211
6451 042560 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 EFEN RECORDED
:THESE ALONG WITH THE FEC, FPS AND PC OF TRAP ARE REPORIED.
:*

FPPSPUR: MOV (SP), \$TMP2 ;SAVE PC OF TRAP.
CMP (SP)+, (SP)+ ;RESTORE SP.
STFPS R0 ;GET FPS
MOV R0, \$TMP3
STST R0 ;GET FEC
MOV R0, \$TMP4
1\$: ERROR +211
RSETUP
;GO INITIALIZE THE FPS AND STACK; AND
;SEE IF THE USER HAS EXPRESSED
;THE DESIRE TO CHANGE THE SOFTWARE
;VIRTUAL CONSOLE SWITCH REGISTER (HAS
;THE USER TYPED CONTROL G?).

6452 042562 000137 036060 JMP \$EOP

6453
6454
6455
6456
6457 042566 011637 001236
6458 042572 022626
6459 042574 104212
6460 042576 104413

6461 042600 000137 036060

```
.SBTTL CPU SPURIOUS TRAP TO 4 HANDLER  
:*****  
:*****  
:*THIS ROUTINE REPORTS UNEXPECTED CPU TRAPS TO VECTOR 4.  
:*  
CPSUR: MOV (SP), $TMP2 ;SAVE PC OF TRAP.  
CMP (SP)+, (SP)+  
1$: ERROR +212  
RSETUP  
  
:GO INITIALIZE THE FPS AND STACK; AND  
:SEE IF THE USER HAS EXPRESSED  
:THE DESIRE TO CHANGE THE SOFTWARE  
:VIRTUAL CONSOLE SWITCH REGISTER (HAS  
:THE USER TYPED CONTROL G?).  
  
JMP $EOP
```

6462
6463
6464
6465
6466 042604 011637 001236
6467 042610 022626
6468 042612 104213
6469 042614 104413

```
.SBTTL CPU SPURIOUS TRAP TO 10 HANDLER  
:*****  
:*****  
:THIS ROUTINE REPORTS UNEXPECTED CPU TRAPS TO VECTOR 10.  
:*****  
:*****  
CPTWO: MOV (SP), $TMP2 ;SAVE PC OF TRAP.  
CMP (SP)+, (SP)+  
1$: ERROR +213  
RSETUP
```

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

6470 042616 000137 036060 JMP \$EOP


```

6471                                     .SBTTL FLAG RESET AND CONSOLE TEST ROUTINE
6472                                     :*****
6473                                     :*THIS ROUTINE WILL BE CALLED AT THE END OF EACH TEST TO
6474                                     :*RESET THE STACK, CLEAR THE FPS AND SEE IF THE USER HAS TYPED
6475                                     :* CONTROL G ON THE TERMINAL. IF THE USER HAS TYPED CONTROL G AND
6476                                     :*THERE IS NO PHYSICAL CONSOLE SWITCH REGISTER THEN THE CONTENTS
6477                                     :*OF THE SOFTWARE SWITCH REGISTER WILL BE TYPED IN OCTAL ON THE
6478                                     :*TELETYPE AND THE USER CAN MODIFY IT.
6479                                     :*
6480 042622 023727 001140 177570 .RSET: CMP     SWR,#177570      :SEE IF THERE IS A PHYSICAL
6481                                     :CONSOLE SWITCH REGISTER.
6482 042630 001001                 BNE     1$              :BRANCH IF NO.
6483 042632 104407                 CKSWR                    :OTHERWISE TYPE THE CONTENTS
6484                                     :OF THE PROGRAM VIRTUAL SWITCH REGISTER
6485                                     :AND GIVE THE USER A CHANCE TO
6486                                     :MODIFY IT.
6487 042634 012737 042534 000244 1$: MOV     #FPSPUR,FPVECT
6488 042642 012737 042566 000004     MOV     #CPSPUR,ERRVECT
6489 042650 012737 042604 000010     MOV     #CPTWO,10
6490 042656 011600                 MOV     (SP),R0        :SAVE RETURN ADDRESS.
6491 042660 012706 001100         MOV     #STACK,SP     :RESET THE STACK POINTER.
6492 042664 005004                 CLR     R4             :CLEAR THE FPS.
6493 042666 170104                 LDFPS  R4
6494 042670 000110                 JMP     (R0)           :RETURN.
  
```

6495						.SBTTL	SPECIAL MESSAGES
6496	042672	200	120	117	POWERM:	.ASCIZ	<CRLF>'POWER FAILURE. PROGRAM RESTARTING.'<CRLF>
6497	042737	000			NULL:	.BYTE	0
6498	042740	011	000		\$TAB:	.ASCIZ	<TAB>
6499	042742	040	040	000	SPACE:	.ASCIZ	'
6500	042745	200	120	103	LFIE1:	.ASCIZ	<CRLF>'PC OF LAST FPP INSTRUCTION EXECUTED: '<TAB>
6501	043015	200	114	101	LFIE2:	.ASCIZ	<CRLF>'LAST FPP INSTRUCTION EXECUTED: '<TAB>
6502	043057	200	106	114	FPSMS:	.ASCIZ	<CRLF>'FLOATING POINT STATUS REGISTER: '
6503	043123	200	106	105	FECMS:	.ASCIZ	<CRLF>'FEC: '
6504	043134	124	110	105	\$THE:	.ASCIZ	'THE '
6505	043141	011	040	111	NOOP1:	.ASCIZ	<TAB>' INSTRUCTION FAILED.'<CRLF>
6506	043170	105	111	124	NOOP15:	.ASCII	'EITHER A BAD CONSTANT WAS GENERATED OR MICROPROGRAM FLOW WENT'
6507	043265	200	106	122	NOOP2:	.ASCIZ	<CRLF>'FROM STATE '
6508	043302	124	117	040	NOOP3:	.ASCIZ	'TO STATE '
6509	043314	200	111	116	NOOP4:	.ASCIZ	<CRLF>'INSTEAD OF '
6510	043331	200	124	110	NOOP5:	.ASCIZ	<CRLF>'THEREBY EXECUTING A '
6511	043357	011	040	111	NOOP6:	.ASCIZ	<TAB>' INSTEAD OF A '
6512	043377	011	040	111	NOOP7:	.ASCIZ	<TAB>' INSTRUCTION.'<CRLF>
6513	043417	040	040	124	NOOP10:	.ASCII	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'<TAB>
6514	043460	107	117	124		.ASCIZ	'GOT FPS. EXPECTED FPS.'<CRLF>
6515	043510	101	040	102	NOOP11:	.ASCIZ	'A BAD CONSTANT MAY HAVE BEEN USED.'<CRLF>
6516	043554	011	114	104	LFPS1:	.ASCIZ	<TAB>'LDFPS'<TAB>'REG'
6517	043567	011	114	104	LD1:	.ASCIZ	<TAB>'LDD'<TAB>'(REG),A'<TAB>'//FSRC#0//'
6518	043617	011	114	104	LD2:	.ASCIZ	<TAB>'LDD'<TAB>'A,A'
6519	043630	011	123	124	STFS1:	.ASCIZ	<TAB>'STFPS'<TAB>'REG'
6520	043643	011	123	124	ST1:	.ASCIZ	<TAB>'STD'<TAB>'A,(REG)'
6521	043660	011	123	124	ST2:	.ASCIZ	<TAB>'STD'<TAB>'A,A'
6522	043671	011	103	106	CFCC1:	.ASCIZ	<TAB>'CFCC'
6523	043677	011	123	105	SETF1:	.ASCIZ	<TAB>'SETF'
6524	043705	011	123	105	SETD1:	.ASCIZ	<TAB>'SETD'
6525	043713	011	123	105	SETI1:	.ASCIZ	<TAB>'SETI'
6526	043721	011	123	105	SETL1:	.ASCIZ	<TAB>'SETL'
6527	043727	011	111	114	ILL1:	.ASCIZ	<TAB>'ILLEGAL FPP INSTRUCTION'
6528	043760	011	123	124	STST1:	.ASCIZ	<TAB>'STST'<TAB>'REG'
6529	043772	011	111	114	ILL2:	.ASCIZ	<TAB>'ILLEGAL FPP INSTRUCTION (FID=1)'
6530	044033	040	040	124	ILLMS:	.ASCIZ	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF TRAP.'<TAB>'FPS.'<CRLF>
6531	044101	105	130	120	MS1:	.ASCIZ	'EXPECTED '
6532	044113	107	117	124	MS2:	.ASCIZ	'GOT '
6533	044120	103	117	116	MS3:	.ASCIZ	'CONTENTS OF LOCATIONS '
6534	044147	040	124	110	MS4:	.ASCIZ	' THROUGH '
6535	044161	106	101	111	MS5:	.ASCIZ	'FAILURE IN THE MICROPROGRAM FLOW.'
6536	044223	103	117	116	MS6:	.ASCIZ	'CONTROL WENT '
6537	044241	106	122	117	MS7:	.ASCIZ	'FROM STATE '
6538	044255	040	124	117	MS10:	.ASCIZ	' TO STATE '
6539	044270	102	125	124	MS11:	.ASCIZ	'BUT SHOULD HAVE GONE'
6540	044315	103	117	116	MS12:	.ASCIZ	'CONTROL FLOW SHOULD HAVE GONE'
6541	044353	102	125	124	MS13:	.ASCIZ	'BUT DID NOT.'
6542	044370	040	040	124	MS14:	.ASCII	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'<TAB>
6543	044431	107	117	124		.ASCIZ	'GOT PC.'<TAB>'EXPECTED PC.'
6544	044456	111	116	123	MS15:	.ASCIZ	'INSTRUCTION TESTED: '
6545	044503	040	117	122	MS16:	.ASCIZ	' OR '
6546	044510	124	105	123	MS17:	.ASCIZ	'TESTING ACCUMULATOR '
6547	044535	132	105	122	MNUM0:	.ASCIZ	'ZERO '
6548	044543	117	116	105	MNUM1:	.ASCIZ	'ONE '
6549	044550	124	127	117	MNUM2:	.ASCIZ	'TWO '
6550	044555	124	110	122	MNUM3:	.ASCIZ	'THREE '
6551	044564	106	117	125	MNUM4:	.ASCIZ	'FOUR '

6552	044572	106	111	126	MNUM5:	.ASCIZ	'FIVE '
6553	044600	040	040	124	MS20:	.ASCIZ	' TEST.' <tab>'PC OF CALL.'<tab>'PC OF ERROR.'</tab></tab>
6554	044641	104	101	124	MS21:	.ASCIZ	'DATA (FLOATING POINT NUMBER): '
6555	044700	114	117	107	MS22:	.ASCIZ	'LOGICAL AND OF FAILING '
6556	044730	114	117	107	MS23:	.ASCIZ	'LOGICAL OR OF FAILING '
6557	044757	040	040	124	MS24:	.ASCII	' TEST.' <tab>'PC OF CALL.'<tab>'PC OF ERRORS.'<tab></tab></tab></tab>
6558	045021	116	125	115		.ASCIZ	'NUMBER OF ERRORS(OCTAL).'
6559	045052	105	130	120	MS25:	.ASCIZ	'EXPECTED DATA IN '
6560	045074	107	117	124	MS26:	.ASCIZ	'GOT DATA IN '
6561	045111	200	101	103	MS27:	.ASCIZ	<CRLF>'AC0= '
6562	045120	200	101	103	MS30:	.ASCIZ	<CRLF>'AC1= '
6563	045127	200	101	103	MS31:	.ASCIZ	<CRLF>'AC2= '
6564	045136	200	101	103	MS32:	.ASCIZ	<CRLF>'AC3= '
6565	045145	200	101	103	MS33:	.ASCIZ	<CRLF>'AC4= '
6566	045154	200	101	103	MS34:	.ASCIZ	<CRLF>'AC5= '
6567	045163	123	105	124	MS35:	.ASCIZ	'SET '
6568	045170	103	114	105	MS36:	.ASCIZ	'CLEAR '
6569	045177	114	117	101	MS37:	.ASCIZ	'LOADED DATA: '
6570	045215	122	105	101	MS40:	.ASCIZ	'READ DATA: '
6571	045231	105	130	120	MS415:	.ASCIZ	'EXPECTED DATA: '
6572	045251	104	101	124	MS41:	.ASCIZ	'DATA IN (RO) FSRC: '
6573	045275	104	101	124	MS42:	.ASCIZ	'DATA IN AC0: '
6574	045313	107	117	124	MS43:	.ASCIZ	'GOT RESULT: '
6575	045330	105	130	120	MS44:	.ASCIZ	'EXPECTED RESULT: '

Address	Instruction	PC	PSW	EM	Message
6576					.SBTTL ERROR MESSAGES
6577	045352	114	104	106	EM1: .ASCIZ 'LDFPS AND STFPS TEST FAILED.'
6578	045407	114	104	106	EM2: .ASCIZ 'LDFPS AND STFPS TEST ERROR SUMMARY.'
6579	045453	103	106	103	EM3: .ASCIZ 'CFCC TRANSFERED BAD DATA TO THE PSW.'
6580	045520	103	106	103	EM4: .ASCIZ 'CFCC MODIFIED THE FPS REGISTER.'
6581	045560	125	116	105	EM5: .ASCIZ 'UNEXPECTED FPP TRAP TO 244.'
6582	045614	125	116	105	EM6: .ASCIZ 'UNEXPECTED CPU TRAP TO 4.'
6583	045646	125	116	105	EM7: .ASCIZ 'UNEXPECTED CPU TRAP TO 10.'
6584		045560			EM10=EM5
6585	045701	125	116	101	EM11: .ASCIZ 'UNABLE TO DECODE FPP INSTRUCTION. TRAPPED TO 10.'
6586		000000			EM12=0
6587		000000			EM13=0
6588	045762	114	104	106	EM14: .ASCII 'LDFPS R0 FAILED IN THE FSRC FLOWS.'
6589	046024	040	124	122	.ASCII ' TRAPPED TO 4.'
6590	046042	200	104	111	.ASCIZ <CRLF>'DID NOT GO FROM STATE 400 TO 670.'
6591	046105	123	124	106	EM15: .ASCII 'STFPS R1 FAILED IN THE FDST FLOWS.'
6592	046147	040	124	122	.ASCII ' TRAPPED TO 4.'
6593	046165	200	104	111	.ASCIZ <CRLF>'DID NOT GO FROM STATE 634 TO 710.'
6594	046230	101	116	040	EM16: .ASCIZ 'AN ILLEGAL FPP INSTRUCTION DID NOT TRAP.'
6595	046301	101	116	040	EM17: .ASCII 'AN ILLEGAL FPP INSTRUCTION'
6596	046333	200	124	122	.ASCII <CRLF>'TRAPPED TO 244, BUT FAILED TO SET '
6597	046376	124	110	105	.ASCII 'THE FPS CORRECTLY.'<CRLF>'EITHER A BAD CONSTANT '
6598	046447	127	101	123	.ASCIZ 'WAS GENERATED OR THE ALU LOGICAL OR FUNCTION FAILED.'
6599	046534	101	116	040	EM20: .ASCII 'AN ILLEGAL FPP INSTRUCTION'
6600	046566	040	124	122	.ASCII ' TRAPPED TO 244, BUT A SUBSEQUENT '
6601	046630	040	123	124	.ASCII ' STST'<CRLF>
6602	046636	106	101	111	.ASCIZ 'FAILED TO PICK UP THE CORRECT FEC CODE = 2.'
6603	046712	123	124	123	EM21: .ASCII 'STST R4 FAILED IN THE DESTINATION FLOWS.'
6604	046762	040	124	122	.ASCII ' TRAPPED TO 4.'<CRLF>
6605	047001	104	111	104	.ASCIZ 'DID NOT GO FROM STATE 636 TO 710.'
6606	047043	101	116	040	EM22: .ASCII 'AN ILLEGAL FPP INSTRUCTION.'
6607	047076	127	111	124	.ASCIZ 'WITH INTERRUPTS DISABLED.'
6608		047043			EM23=EM22
6609		047043			EM24=EM22
6610	047130	123	117	125	EM25: .ASCII 'SOURCE LOCATIONS MODIFIED BY, LDD.'
6611	047172	200	101	040	.ASCIZ <CRLF>'A DATO WAS PERFORMED INSTEAD OF A DATI.'
6612	047243	114	104	104	EM26: .ASCII 'LDD (R0),ACO FAILED.'<CRLF>
6613	047270	122	060	040	.ASCIZ 'R0 WAS MODIFIED.'
6614		047243			EM27=EM26
6615	047311	124	110	105	EM30: .ASCII 'THE PC WAS BAD AFTER '
6616	047337	101	116	040	.ASCIZ 'AN FPP INSTRUCTION.'
6617	047363	123	124	104	EM31: .ASCII 'STD ACO, (R0) FAILED.'<CRLF>
6618	047410	122	060	040	.ASCIZ 'R0 WAS MODIFIED.'
6619		047363			EM32=EM31
6620	047431	123	124	104	EM33: .ASCII 'STD ACO, (R0) FAILED.'<CRLF>
6621	047456	117	125	124	.ASCIZ 'OUTPUT BAD.'
6622	047472	123	124	104	EM34: .ASCII 'STD ACO, (R0) FAILED IN THE FDST FLOWS.'
6623	047540	200	124	110	.ASCIZ <CRLF>'THE (BUT GR7) FORK FAILED.'
6624	047574	114	104	104	EM35: .ASCII 'LDD (R0),ACO FAILED IN THE FSRC FLOWS.'
6625	047642	200	124	110	.ASCIZ <CRLF>'THE (BUT GR7) FORK FAILED.'
6626	047676	123	124	104	EM36: .ASCII 'STD ACO, (R0) FAILED IN THE FDST FLOWS.'
6627	047744	200	124	110	.ASCIZ <CRLF>'THE (BUT FD) FORK FAILED.'
6628	047777	114	104	104	EM37: .ASCII 'LDD (R0),ACO FAILED IN THE FSRC FLOWS.'
6629	050045	200	124	110	.ASCIZ <CRLF>'THE (BUT FD) FORK FAILED.'
6630	050100	114	104	104	EM40: .ASCII 'LDD (R0),ACO OR THE STD ACO, (R0) FAILED.'
6631	050150	200	102	101	.ASCIZ <CRLF>'BAD DATA WAS DETECTED AFTER A SEQUENCE OF THE TWO INSTRUCTIONS.'
6632	050251	106	120	123	EM41: .ASCIZ 'FPS BAD AFTER EXECUTION OF: '

6637	050306				EM42:	
	050306	114	104	104	.ASCII	/LDD (RO),ACO FAILED IN THE FSRC FLOWS./<CRLF>
	050355	124	110	105	.ASCIZ	/THE (BUT FSRC) FORK FAILED. TRAPPED TO 4./
6638	050427				EM43:	
	050427	123	124	104	.ASCII	/STD ACO,(RO) FAILED IN THE FDST FLOWS./<CRLF>
	050476	124	110	105	.ASCIZ	/THE (BUT FDST) FORK FAILED. TRAPPED TO 4./
6639	050550	106	120	120	EM44:	'FPP ACCUMULATORS DATA TEST FAILED.'
6640		050550			EM45=EM44	
6641	050613	106	120	120	EM46:	'FPP ACCUMULATORS DUAL ADDRESSING TEST FAILED.'
6646	050671				EM47:	
	050671	114	104	040	.ASCII	/LD AC1,ACO FAILED IN THE FSRC FLOWS./
	050735	124	110	105	.ASCIZ	/THE (BUT FSRC) FORK FAILED. TRAPPED TO 4./
6647	051007	114	104	040	EM50:	'LD AC1,ACO FAILED IN THE FSRC FLOWS.'
6648	051053	124	110	105	.ASCIZ	'THE (BUT FD) FORK FAILED.'
6649	051105	114	104	040	EM51:	'LD AC1,ACO TRANSFERRED BAD DATA.'
6659	051146				EM52:	
	051146	114	104	104	.ASCII	/LDD (RO)+,ACO FAILED IN THE FSRC FLOWS./
	051215	124	110	105	.ASCIZ	/THE (BUT FSRC) FORK FAILED. TRAPPED TO 4./
6660	051267				EM53:	
	051267	114	104	104	.ASCII	/LDD (RO)+,ACO FAILED IN THE FSRC FLOWS./
	051336	200	122	060	.ASCII	<CRLF>'RO WAS BAD.'<CRLF>
	051353	105	111	124	.ASCII	'EITHER A BAD CONSTANT WAS GENERATED OR'<CRLF>
	051422	104	111	104	.ASCIZ	\DID NOT GO FROM STATE 627 TO 322.\
6661	051464				EM54:	
	051464	114	104	104	.ASCIZ	/LDD (RO)+,ACO TRANSFERRED BAD DATA./
6662	051530				EM55:	
	051530	114	104	104	.ASCII	/LDD -(RO),ACO FAILED IN THE FSRC FLOWS./
	051577	124	110	105	.ASCIZ	/THE (BUT FSRC) FORK FAILED. TRAPPED TO 4./
6663	051651				EM56:	
	051651	114	104	104	.ASCII	/LDD -(RO),ACO FAILED IN THE FSRC FLOWS./
	051720	200	122	060	.ASCII	<CRLF>'RO WAS BAD.'<CRLF>
	051735	105	111	124	.ASCII	'EITHER A BAD CONSTANT WAS GENERATED OR'<CRLF>
	052004	104	111	104	.ASCIZ	\DID NOT GO FROM STATE 627 TO 324.\
6664	052046				EM57:	
	052046	114	104	104	.ASCIZ	/LDD -(RO),ACO TRANSFERRED BAD DATA./
6665	052112				EM60:	
	052112	114	104	106	.ASCII	/LDF (RO)+,ACO FAILED IN THE FSRC FLOWS./
	052161	200	122	060	.ASCII	<CRLF>'RO WAS BAD.'<CRLF>
	052176	105	111	124	.ASCII	'EITHER A BAD CONSTANT WAS GENERATED OR'<CRLF>
	052245	104	111	104	.ASCIZ	\DID NOT GO FROM STATE 627 TO 322.\
6666	052307				EM61:	
	052307	114	104	106	.ASCIZ	/LDF (RO)+,ACO TRANSFERRED BAD DATA./
6667	052353	114	104	106	EM62:	'LDF (RO)+,ACO FAILED IN THE FSRC FLOWS.'
6668	052422	200	124	110	.ASCII	<CRLF>'THE (BUT FD) FORK FAILED.'<CRLF>
6669	052455	127	105	116	.ASCII	'WENT FROM STATE 441 TO 077.'<CRLF>
6670	052511	111	116	123	.ASCIZ	'INSTEAD OF FROM 441 TO 076.'
6671	052545	114	104	104	EM63:	'LDD #NUM,ACO FAILED IN THE FSRC FLOWS.'
6672	052613	200	124	110	.ASCII	<CRLF>'THE (BUT GR7) FORK FAILED.'<CRLF>
6673	052647	127	105	116	.ASCII	'WENT FROM STATE 207 TO 174.'<CRLF>
6674	052703	111	116	123	.ASCIZ	'INSTEAD OF FROM 207 TO 176.'
6675	052737	114	104	104	EM64:	'LDD #NUM,ACO FAILED IN THE FSRC FLOWS.'
6676	053005	200	101	040	.ASCIZ	<CRLF>'A BAD CONSTANT WAS USED WHEN THE PC WAS INCREMENTED.'
6677		052737			EM65=EM64	
6678	053073				EM66:	
	053073	114	104	104	.ASCIZ	/LDD #NUM,ACO TRANSFERRED BAD DATA./
6700	053136				EM67:	
	053136	114	104	104	.ASCII	'LDD @ (RO)+,ACO FAILED IN THE FSRC FLOWS.'

053206	200	124	110	.ASCII	<CRLF>'THE (BUT FSRC) FORK FAILED. TRAPPED TO 4.'
053260	200	127	105	.ASCII	<CRLF>\WENT FROM STATE 627 TO EITHER 326 OR 326,\
053332	200	111	116	.ASCIIZ	<CRLF>\INSTEAD OF FROM 627 TO 323.\
6701 053367				EM70:	
053367	114	104	104	.ASCII	'LDD @ (RO)+,ACO FAILED IN THE FSRC FLOWS.'
053437	200	124	110	.ASCIIZ	<CRLF>'THE (BUT FSRC) FORK FAILED. TRAPPED TO 4.'
6702 053512				EM71:	
053512	114	104	104	.ASCII	'LDD @ (RO)+,ACO FAILED IN THE FSRC FLOWS.'
053562	124	110	105	.ASCIIZ	'THE (BUT FD) FORK FAILED.'
6703 053614				EM72:	
053614	114	104	104	.ASCII	'LDD @ (RO)+,ACO'<CRLF>
053633	106	101	111	.ASCIIZ	'FAILED TO INCREMENT RO BY 2.'
6704 053670				EM73:	
053670	114	104	104	.ASCIIZ	'LDD @ (RO)+,ACO LOADED BAD DATA.'
6705 053730				EM74:	
053730	114	104	104	.ASCII	'LDD @-(RO),ACO FAILED IN THE FSRC FLOWS.'
054000	200	124	110	.ASCII	<CRLF>'THE (BUT FSRC) FORK FAILED. TRAPPED TO 4.'
054052	200	127	105	.ASCII	<CRLF>\WENT FROM STATE 627 TO EITHER 326 OR 326,\
054124	200	111	116	.ASCIIZ	<CRLF>\INSTEAD OF FROM 627 TO 325.\
6706 054161				EM75:	
054161	114	104	104	.ASCII	'LDD @-(RO),ACO FAILED IN THE FSRC FLOWS.'
054231	200	124	110	.ASCIIZ	<CRLF>'THE (BUT FSRC) FORK FAILED. TRAPPED TO 4.'
6707 054304				EM76:	
054304	114	104	104	.ASCII	'LDD @-(RO),ACO FAILED IN THE FSRC FLOWS.'
054354	124	110	105	.ASCIIZ	'THE (BUT FD) FORK FAILED.'
6708 054406				EM77:	
054406	114	104	104	.ASCII	'LDD @-(RO),ACO'<CRLF>
054425	106	101	111	.ASCIIZ	'FAILED TO DECREMENT RO BY 2.'
6709 054462				EM100:	
054462	114	104	104	.ASCIIZ	'LDD @-(RO),ACO LOADED BAD DATA.'
6710 054522				EM101:	
054522	114	104	104	.ASCII	'LDD NUM(RO),ACO FAILED IN THE FSRC FLOWS.'
054573	200	124	110	.ASCIIZ	<CRLF>'THE (BUT FSRC) FORK FAILED. TRAPPED TO 4.'
6711 054646				EM102:	
054646	114	104	104	.ASCII	'LDD NUM(RO),ACO'<CRLF>
054666	106	101	111	.ASCIIZ	'FAILED TO AFFECT RO BY 2.'
6712 054720				EM103:	
054720	114	104	104	.ASCII	'LDD NUM(RO),ACO FAILED IN THE FSRC FLOWS.'
054771	124	110	105	.ASCIIZ	'THE (BUT FD) FORK FAILED.'
6713 055023				EM104:	
055023	114	104	104	.ASCIIZ	'LDD NUM(RO),ACO LOADED BAD DATA.'
6714 055064				EM105:	
055064	114	104	104	.ASCII	'LDD @NUM(RO),ACO FAILED IN THE FSRC FLOWS.'
055136	200	124	110	.ASCIIZ	<CRLF>'THE (BUT FSRC) FORK FAILED. TRAPPED TO 4.'
6715 055211				EM106:	
055211	114	104	104	.ASCII	'LDD @NUM(RO),ACO'<CRLF>
055232	106	101	111	.ASCIIZ	'FAILED TO AFFECT RO BY 2.'
6716 055264				EM107:	
055264	114	104	104	.ASCII	'LDD @NUM(RO),ACO FAILED IN THE FSRC FLOWS.'
055336	124	110	105	.ASCIIZ	'THE (BUT FD) FORK FAILED.'
6717 055370				EM110:	
055370	114	104	104	.ASCIIZ	'LDD @NUM(RO),ACO LOADED BAD DATA.'
6733 055432				EM111:	
055432	114	104	104	.ASCII	/LDD AC7,ACO FAILED TO TRAP TO 244./
055474	200	101	103	.ASCIIZ	<CRLF>/AC7 IS AN ILLEGAL ACCUMULATOR./
6734 055534	055432			EM112=EM111	
6735				EM113:	

	055534	114	104	104	.ASCII	/LDD AC6,ACO FAILED TO TRAP TO 244./
	055576	200	101	103	.ASCIIZ	<CRLF>/AC6 IS AN ILLEGAL ACCUMULATOR./
6736		055534			EM114=EM113	
6737		055432			EM115=EM111	
6738		055534			EM116=EM113	
6739	055636				EM117:	
	055636	125	123	105	.ASCII	'USE OF AN ILLEGAL ACCUMULATOR WITH FSRC MODE ZERO.'
	055720	200	124	122	.ASCIIZ	<CRLF>'TRAPPED BUT FAILED TO SET FPS CORRECTLY.'
6740	055772				EM120:	
	055772	125	123	105	.ASCII	'USE OF AN ILLEGAL ACCUMULATOR WITH FSRC MODE ZERO.'
	056054	200	124	122	.ASCIIZ	<CRLF>'TRAPPED BUT FAILED TO SET FEC CORRECTLY.'
6741	056126	123	124	040	EM121:	.ASCII 'ST ACO,AC1 FAILED IN THE FDST FLOWS.'
6742	056172	200	124	110	.ASCIIZ	<CRLF>'THE (BUT FDST) FORK FAILED. TRAPPED TO 4.'
6743	056245	123	124	040	EM122:	.ASCII 'ST ACO,AC1 FAILED IN THE FDST FLOWS.'
6744	056311	200	124	110	.ASCIIZ	<CRLF>'THE (BUT FD) FORK FAILED.'
6745	056344	123	124	040	EM123:	.ASCIIZ 'ST ACO,AC1 TRANSFERRED BAD DATA.'
6746	056405				EM124:	
	056405	106	120	123	.ASCII	'FPS BAD AFTER LDD (RO),ACO.'
	056440	200	124	110	.ASCIIZ	<CRLF>\THE (BUT EZBT Y8) FORK FAILED.\
6747	056500				EM125:	
	056500	106	120	123	.ASCII	'FPS BAD AFTER LDD (RO),ACO.'
	056533	200	124	110	.ASCIIZ	<CRLF>\THE (BUT ENBT) FORK FAILED.\
6748	056570	114	104	104	EM126:	.ASCII 'LDD (RO),ACO TRAPPED TO 244.'
6749	056624	040	106	123	.ASCII	' FSRC= -0 AND FIUV= 0.'<CRLF>
6750	056653	124	110	105	.ASCII	'THE (BUT FIUV) FORK FAILED.'
6751	056706	200	127	105	.ASCII	<CRLF>'WENT FROM STATE 256 TO 354.'
6752	056742	200	111	116	.ASCIIZ	<CRLF>'INSTEAD OF FROM 256 TO 254.'
6753	056777	114	104	104	EM127:	.ASCII 'LDD (RO),ACO FAILED TO TRAP TO 244.'
6754	057042	040	106	123	.ASCII	' FSRC= -0, FIUV= 1.'
6755	057065	200	124	110	.ASCII	<CRLF>'THE (BUT FIUV) FORK FAILED.'<CRLF>
6756	057121	127	105	116	.ASCII	'WENT FROM STATE 256 TO 254.'
6757	057154	200	111	116	.ASCIIZ	<CRLF>'INSTEAD OF FROM 256 THE 354.'
6758	057212	114	104	104	EM130:	.ASCII 'LDD (RO),ACO TRAPPED TO 244.'
6759	057246	106	123	122	.ASCII	'FSRC= -0, FIUV= 1.'<CRLF>
6760	057271	102	125	124	.ASCIIZ	'BUT FEC WAS BAD.'
6761	057312				EM131:	
	057312	114	104	103	.ASCIIZ	/LDCFD (RO),ACO LOADED BAD DATA./
6762	057352				EM132:	
	057352	114	104	103	.ASCIIZ	/LDCDF (RO),ACO LOADED BAD DATA./
6803	057412				EM133:	
	057412	101	104	104	.ASCIIZ	/ADDD (RO),ACO WITH (RO)=ACO=0 /
6804	057451				EM134:	
	057451	101	104	104	.ASCIIZ	/ADDF (RO),ACO WITH (RO)=ACO=0 /
6805	057510				EM135:	
	057510	123	125	102	.ASCIIZ	/SUBD (RO),ACO WITH (RO)=ACO=0 /
6806	057547				EM136:	
	057547	123	125	102	.ASCIIZ	/SUBF (RO),ACO WITH (RO)=ACO=0 /
6807		057412			EM137=EM133	
6808		057451			EM140=EM134	
6809		057510			EM141=EM135	
6810		057547			EM142=EM136	
6811	057606				EM143:	
	057606	101	104	104	.ASCIIZ	/ADDD (RO),ACO WITH (RO)=0 /
6812	057641				EM144:	
	057641	123	125	102	.ASCIIZ	/SUBD (RO),ACO WITH (RO)=0 /
6813		057606			EM145=EM143	
6814		057641			EM146=EM144	

6815	057674				EM147:	
	057674	123	125	102	.ASCIZ	/SUBD (R0),ACO WITH ACO=0 /
6816		057674			EM150=EM147	
6817		057674			EM151=EM147	
6818	057726				EM152:	
	057726	101	104	104	.ASCIZ	/ADDD (R0),ACO WITH ACO=0 /
6819		057726			EM153=EM152	
6820	057760				EM154:	
	057760	101	116	040	.ASCII	'AN OVERFLOW ERROR OCCURRED ON ADD'<CRLF>
	060022	103	101	125	.ASCII	'CAUSING A TRAP TO 244.'
	060050	200	050	102	.ASCII	<CRLF>'(BUT EZBT Y9 Y8) FORK IN STATE 420 OF OVER\UNDER FAILED.'
	060141	200	123	110	.ASCIZ	<CRLF>'SHOULD HAVE GONE FROM STATE 420 TO 131.'
6821	060212				EM155:	
	060212	101	116	040	.ASCII	'AN UNDERFLOW ERROR OCCURRED ON ADD'<CRLF>
	060255	103	101	125	.ASCII	'CAUSING A TRAP TO 244.'
	060303	200	050	102	.ASCII	<CRLF>'(BUT EZBT Y9 Y8) FORK IN STATE 420 OF OVER\UNDER FAILED.'
	060374	200	123	110	.ASCIZ	<CRLF>'SHOULD HAVE GONE FROM STATE 420 TO 131.'
6822	060445				EM156:	
	060445	101	104	104	.ASCII	/ADDD (R0),ACO FAILED IN THE ROUND\TRUNK FLOWS./
	060523	200	124	110	.ASCII	<CRLF>'THE (BUT FD) FORK FAILED. WENT'
	060562	106	122	117	.ASCII	\FROM STATE 665 TO 113.\<CRLF>
	060611	111	116	123	.ASCIZ	\INSTEAD OF FROM 665 TO 313.\<CRLF>\WITH FT SET.\
6823	060662				EM157:	
	060662	101	104	104	.ASCII	/ADDD (R0),ACO FAILED IN THE ROUND\TRUNK FLOWS./
	060740	200	124	110	.ASCII	<CRLF>'THE (BUT FD) FORK FAILED. WENT'
	060777	106	122	117	.ASCII	\FROM STATE 665 TO 313.\<CRLF>
	061026	111	116	123	.ASCIZ	\INSTEAD OF FROM 665 TO 113.\<CRLF>\WITH FT CLEAR.\
6824	061101				EM160:	
	061101	101	104	104	.ASCII	/ADDD (R0),ACO FAILED IN THE ROUND\TRUNK FLOWS./<CRLF>
	061160	124	110	105	.ASCII	'THE FLOATING CONSTANT WAS USED INSTEAD OF THE DOUBLE CONSTANT'<CRLF>
	061256	111	116	040	.ASCIZ	'IN THE ROUND ALGORITHM.'
6825	061306				EM161:	
	061306	101	104	104	.ASCII	/ADDF (R0),ACO FAILED IN THE ROUND\TRUNK FLOWS./<CRLF>
	061365	124	110	105	.ASCII	'THE DOUBLE CONSTANT WAS USED INSTEAD OF THE FLOATING CONSTANT'<CRLF>
	061463	111	116	040	.ASCIZ	'IN THE ROUND ALGORITHM.'
6826	061513				EM162:	
	061513	101	104	104	.ASCIZ	/ADDD (R0),ACO PRODUCED A BAD RESULT./
6827	061560				EM163:	
	061560	101	104	104	.ASCIZ	/ADDF (R0),ACO PRODUCED A BAD RESULT./
6828	061625				EM164:	
	061625	124	110	105	.ASCIZ	\THE FPS WAS BAD AFTER ADDD (R0),ACO.\
6829	061672				EM165:	
	061672	124	110	105	.ASCIZ	\THE FPS WAS BAD AFTER ADDF (R0),ACO.\
6830	061737				EM166:	
	061737	101	104	104	.ASCII	/ADDD (R0),ACO PRODUCED A BAD RESULT./<CRLF>
	062004	120	122	117	.ASCIZ	'PROBABLE ERROR IN THE ALIGN FLOWS.'
6831	062047				EM167:	
	062047	101	104	104	.ASCII	/ADDD (R0),ACO FAILED IN THE ALIGN FLOWS./<CRLF>
	062120	106	114	117	.ASCII	\FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 111, TO 014.\
	062210	200	101	040	.ASCII	<CRLF>\A BAD CONSTANT (NOT 57 DEC) \
	062245	127	101	123	.ASCIZ	'WAS USED IN THE ALIGN ALGORITHM.'
6832	062306				EM170:	
	062306	101	104	104	.ASCII	/ADDF (R0),ACO PRODUCED A BAD RESULT./<CRLF>
	062353	120	122	117	.ASCIZ	'PROBABLE ERROR IN THE ALIGN FLOWS.'
6833	062416				EM171:	
	062416	101	104	104	.ASCII	/ADDF (R0),ACO FAILED IN THE ALIGN FLOWS./<CRLF>
	062467	106	114	117	.ASCII	\FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 111, TO 014.\

	062557	200	101	040		.ASCII	<CRLF>\A BAD CONSTANT (NOT 25 DEC) \
	062614	127	101	123		.ASCIZ	'WAS USED IN THE ALIGN ALGORITHM.'
6834	062655				EM172:		
	062655	101	104	104		.ASCII	/ADDD (R0),ACO FAILED IN THE ALIGN FLOWS./<CRLF>
	062726	106	114	117		.ASCII	\FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 011, TO 015.\
	063016	200	101	040		.ASCII	<CRLF>\A BAD CONSTANT (NOT 57 DEC) \
	063053	127	101	123		.ASCIZ	'WAS USED IN THE ALIGN ALGORITHM.'
6835	063114				EM173:		
	063114	101	104	104		.ASCII	/ADDD (R0),ACO FAILED IN THE ALIGN FLOWS./<CRLF>
	063165	106	114	117		.ASCII	\FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 011, TO 215.\
	063255	200	101	040		.ASCII	<CRLF>\A BAD CONSTANT (NOT 57 DEC) \
	063312	127	101	123		.ASCIZ	'WAS USED IN THE ALIGN ALGORITHM.'
6836	063353				EM174:		
	063353	101	104	104		.ASCII	/ADDF (R0),ACO FAILED IN THE ALIGN FLOWS./<CRLF>
	063424	106	114	117		.ASCII	\FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 011, TO 015.\
	063514	200	101	040		.ASCII	<CRLF>\A BAD CONSTANT (NOT 25 DEC) \
	063551	127	101	123		.ASCIZ	'WAS USED IN THE ALIGN ALGORITHM.'
6837	063612				EM175:		
	063612	101	104	104		.ASCII	/ADDF (R0),ACO FAILED IN THE ALIGN FLOWS./<CRLF>
	063663	106	114	117		.ASCII	\FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 011, TO 215.\
	063753	200	101	040		.ASCII	<CRLF>\A BAD CONSTANT (NOT 25 DEC) \
	064010	127	101	123		.ASCIZ	'WAS USED IN THE ALIGN ALGORITHM.'
6838	064051				EM176:		
	064051	101	104	104		.ASCII	'ADDD (R0),ACO FAILED IN THE ADD-SUB FLOWS.'<CRLF>
	064124	104	111	104		.ASCIZ	\DID NOT TAKE THE PATH: STATE 216, TO 442, TO 500.\
6839	064206	120	117	127	EM177:	.ASCIZ	'POWER MONITOR BIT FOUND SET'
6840	064242				EM200:		
	064242	101	104	104		.ASCII	'ADDD (R0),ACO FAILED IN THE ADD-SUB FLOWS.'<CRLF>
	064315	104	111	104		.ASCIZ	\DID NOT TAKE THE PATH: STATE 216, TO 440, TO 121.\
6841	064377				EM201:		
	064377	101	104	104		.ASCII	'ADDD (R0),ACO FAILED IN THE ADD-SUB FLOWS.'<CRLF>
	064452	104	111	104		.ASCIZ	\DID NOT TAKE THE PATH: STATE 216, TO 440, TO 101.\
6842	064534				EM202:		
	064534	101	104	104		.ASCII	'ADDD (R0),ACO FAILED IN THE ADD-SUB FLOWS.'<CRLF>
	064607	104	111	104		.ASCIZ	\DID NOT TAKE THE PATH: STATE 216, TO 042, TO 101.\
6843	064671				EM203:		
	064671	101	104	104		.ASCII	'ADDD (R0),ACO FAILED IN THE ADD-SUB FLOWS.'<CRLF>
	064744	104	111	104		.ASCIZ	\DID NOT TAKE THE PATH: STATE 216, TO 440, TO 141.\
6844	065026				EM204:		
	065026	101	104	104		.ASCII	'ADDD (R0),ACO FAILED IN THE ADD-SUB FLOWS.'<CRLF>
	065101	104	111	104		.ASCIZ	\DID NOT TAKE THE PATH: STATE 216, TO 042, TO 141.\
6845	065163				EM205:		
	065163	124	110	105		.ASCIZ	\THE FPS WAS BAD AFTER SUBD (R0),ACO.\
6846	065230				EM206:		
	065230	123	125	102		.ASCIZ	/SUBD (R0),ACO PRODUCED A BAD RESULT./
6847	065275	123	125	102	EM207:	.ASCII	'SUBD (R0),ACO PRODUCED A BAD RESULT.'
6848	065341	200	124	110		.ASCIZ	<CRLF>'THE XOR OF THE SIGN BIT FAILED IN STATE 024.'
6849	065417	101	104	104	EM210:	.ASCIZ	'ADDD (R0),ACO FAILED IN THE NORMALIZE FLOWS.'
6850		045560			EM211=EM5		
6851		045614			EM212=EM6		
6852		045646			EM213=EM7		
6853	065474				EM214:		

Line	Address	Op	PC	PC	PC	Op	Text
	065474		101	104	104	.ASCII	'ADD (RO),ACO FAILED IN THE ADD-SUB FLOWS.'<CRLF>
	065547		104	111	104	.ASCII	\DID NOT TAKE THE PATH: STATE 216, TO 042, TO 121.\
6854						.SBTTL	DATA HEADERS
6855	065631		040	040	124	DH1:	.ASCII ' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6856	065671		011	127	122		<TAB>'WROTE.'<TAB>'READ.'<TAB>'EXPECTED.'
6857	065721		040	040	124	DH2:	.ASCII ' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6858	065761		101	116	104		'AND BAD DATA.'<TAB>'OR BAD DATA.'
6859	066014		040	040	124	DH3:	.ASCII ' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6860	066054		011	122	105		<TAB>'READ PSW.'<TAB>'EXPECTED PSW.'
6861	066105		040	040	124	DH4:	.ASCII ' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6862	066145		011	127	122		<TAB>'WROTE FPS.'<TAB>'FPS AFTER CFCC.'
6863	066201		040	040	124	DH5:	.ASCII ' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF TRAP.'
6864		066201					DH6=DH5
6865		066201					DH7=DH5
6866		066201					DH10=DH5
6867		066201					DH11=DH5
6868		000000					DH12=0
6869		000000					DH13=0
6870		066201					DH14=DH5
6871		066201					DH15=DH5
6872	066241		040	040	124	DH16:	.ASCII ' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6873	066301		011	117	120		<TAB>'OP CODE. FPS.'
6874	066321		040	040	124	DH17:	.ASCII ' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6875	066361		011	107	117		<TAB>'GOT FPS.'<TAB>'EXPECTED FPS.'
6876	066411		040	040	124	DH20:	.ASCII ' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF TRAP.'
6877	066450		011	120	103		<TAB>'PC OF STST.'<TAB>'READ FEC.'
6878		066201					DH21=DH5
6879	066477		106	101	111	DH22:	.ASCII 'FAILED TO CORRECTLY SET FPS.'
6880	066534		106	101	111	DH23:	.ASCII 'FAILED TO CORRECTLY SET FEC TO 000002.'<CRLF>
6881	066603		040	040	124		' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6882	066643		011	120	103		<TAB>'PC OF STST.'<TAB>'READ FEC.'
6883	066672		124	122	101	DH24:	.ASCII 'TRAPPED TO 244. FLOW WENT FROM STATE 554 TO STATE 430.'
6884	066760		200	111	116		<CRLF>'INSTEAD OF FROM STATE 554 TO STATE 432.'
6885	067031		040	040	124	DH25:	.ASCII ' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'<TAB>
6886	067073		040	040	124	DH26:	.ASCII ' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6887	067133		011	107	117		<TAB>'GOT RO.'<TAB>'EXPECTED RO.'
6888		067073					DH27=DH26
6889		000000					DH30=0
6890		067073					DH31=DH26
6891		067073					DH32=DH26
6892	067161		040	040	124	DH33:	.ASCII ' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6893	067221		011	122	060		<TAB>'RO (TARGET LOCATIONS FOR OUTPUT).'
6894		067161					DH34=DH33
6895		067161					DH35=DH33
6896		067161					DH36=DH33
6897		067161					DH37=DH33
6898		067161					DH40=DH33
6899		000000					DH41=0
6900	067264		040	040	124	DH42:	.ASCII ' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF TRAP.'
6901	067323		011	122	060		<TAB>'RO (TARGET LOCATIONS FOR OUTPUT).'
6902		067264					DH43=DH42
6903		000000					DH44=0
6904	067366		105	122	122	DH45:	.ASCII 'ERROR SUMMARY.'
6905	067405		040	040	124	DH46:	.ASCII ' TEST.'<TAB>'CALL AT PC.'
6906		067366					DH47=DH45
6907	067431		040	040	124	DH50:	.ASCII ' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6908	067471		011	127	111		<TAB>'WITH FD.'

6909		067431				DH51=DH50	
6910		066201				DH52=DH5	
6911		067073				DH53=DH26	
6912	067503	040	040	124		DH54: .ASCIZ	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6913		066201				DH55=DH5	
6914		067073				DH56=DH26	
6915		067503				DH57=DH54	
6916		067073				DH60=DH26	
6917		067503				DH61=DH54	
6918		067503				DH62=DH54	
6919		067503				DH63=DH54	
6920	067544	122	105	123		DH65: .ASCII	'RESULTING IN AN ODD ADDRESS TRAP TO 4.'
6921	067612	200				.ASCII	<CRLF>
6922	067613	040	040	124		DH64: .ASCII	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6923	067653	011	107	117		.ASCIZ	<TAB>'GOT PC.'<TAB>'EXPECTED PC.'
6924		067503				DH66=DH54	
6925		066201				DH67=DH5	
6926		066201				DH70=DH5	
6927		067031				DH71=DH25	
6928		067073				DH72=DH26	
6929		067503				DH73=DH54	
6930		066201				DH74=DH5	
6931		066201				DH75=DH5	
6932		067031				DH76=DH25	
6933		067073				DH77=DH26	
6934		067503				DH100=DH54	
6935		066201				DH101=DH5	
6936		067073				DH102=DH26	
6937		067031				DH103=DH25	
6938		067503				DH104=DH54	
6939		066201				DH105=DH5	
6940		067073				DH106=DH26	
6941		067031				DH107=DH25	
6942		067503				DH110=DH54	
6943		067031				DH111=DH25	
6944	067701	124	110	105		DH112: .ASCII	'THE (BUT FSRC) FORK FAILED.'<CRLF>
6945	067735	103	117	116		.ASCII	'CONTROL WENT FROM STATE 762 TO STATE 627.'
6946	070006	200	111	116		.ASCII	<CRLF>'INSTEAD OF FROM STATE 762 TO STATE 637.'<CRLF>
6947	070057	040	040	124		.ASCIZ	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6948		067031				DH113=DH25	
6949		067701				DH114=DH112	
6950	070120	124	110	105		DH115: .ASCII	'THE (BUT FSRC) FORK FAILED RESULTING IN AN ODD ADDRESS TRAP TO 4.'
6951	070221	200	103	117		.ASCII	<CRLF>'CONTROL WENT FROM STATE 762 TO STATE 627.'<CRLF>
6952	070274	111	116	123		.ASCII	'INSTEAD OF FROM STATE 762 TO STATE 627.'<CRLF>
6953	070344	040	040	124		.ASCIZ	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF TRAP.'
6954		070120				DH116=DH115	
6955		066321				DH117=DH17	
6956	070404	040	040	124		DH120: .ASCII	' TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
6957	070444	011	107	117		.ASCIZ	<TAB>'GOT FEC.'<TAB>'EXPECTED FEC.'
6958		066201				DH121=DH5	
6959		067431				DH122=DH50	
6960		067431				DH123=DH50	
6961		066321				DH124=DH17	
6962		066321				DH125=DH17	
6963		066201				DH126=DH5	
6964		067503				DH127=DH54	
6965		070404				DH130=DH120	

Line	Code	Value	Unit	Code	DF	Type	Data
7022						.SBTTL	DATA FORMATS
7023	071236	004	000	005	DF1:	.BYTE	4.0.5.0.5.0.0.0
7024	071246	004	000	005	DF2:	.BYTE	4.0.5.4.5.0.5.0
7025	071256	004	000	005	DF3:	.BYTE	4.0.5.0.5.0.5.0
7026		071256			DF4=DF3		
7027	071266	004	000	005	DF5:	.BYTE	4.0.5.0.5.0.5.11.5.0.5.0
7028		071266			DF6=DF5		
7029		071266			DF7=DF5		
7030		071266			DF10=DF5		
7031		071266			DF11=DF5		
7032	071302	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
7033	071334	005	011	005	DF13:	.BYTE	5.11.5.5.5.4.0.5.0.5.0.0
7034		071266			DF14=DF6		
7035		071266			DF15=DF6		
7036	071350	004	000	005	DF16:	.BYTE	4.0.5.0.5.0.0
7037		071256			DF17=DF3		
7038	071357	004	000	005	DF20:	.BYTE	4.0.5.0.5.0.5.0
7039	071367	004	000	005	DF21:	.BYTE	4.0.5.0
7040	071373	005	005	004	DF22:	.BYTE	5.5.4.0.5.0.5.0.5.0
7041	071405	004	000	005	DF23:	.BYTE	4.0.5.0.5.0.5.0
7042	071415	005	004	000	DF24:	.BYTE	5.4.0.5.0.5.0
7043	071424	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
7044	071450	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
7045	071475	004	000	005	DF27:	.BYTE	4.0.5.0.5.0.0.5.5.5.5.4.5.4.5.5
7046	071515	005	011	005	DF30:	.BYTE	5.11.5.5.5.4.0.5.0.5.0.0
7047		071450			DF31=DF26		
7048		071475			DF32=DF27		
7049	071531	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
7050	071557	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
7051		071557			DF35=DF34		
7052		071557			DF36=DF34		
7053		071557			DF37=DF34		
7054	071603	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
7055	071631	011	005	005	DF41:	.BYTE	11.5.5.5.4.0.5.0.5.0.5.0
7056	071645	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
7057		071645			DF43=DF42		
7058	071673	005	011	005	DF44:	.BYTE	5.11.5.5.5.4.0.5.0.5.5.5.5.3.5.5.5.5.3
7059	071716	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
7060	071743	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
7061	071775	004	000	005	DF47:	.BYTE	4.0.5.0.5.5.5.4.5.4.5.5
7062	072012	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
7063	072036	004	000	005	DF51:	.BYTE	4.0.5.0.5.11.5.5.5.3.5.5.5.3
7064		071775			DF52=DF47		
7065	072054	004	000	005	DF53:	.BYTE	4.0.5.0.5.0.0
7066	072063	004	000	005	DF54:	.BYTE	4.0.5.0.5.5.5.5.3.5.5.5.5.3
7067		071775			DF55=DF47		
7068		072054			DF56=DF53		
7069		072063			DF57=DF54		
7070		072054			DF60=DF53		
7071		072063			DF61=DF54		
7072		072063			DF62=DF54		
7073		072063			DF63=DF54		
7074	072101	004	000	005	DF64:	.BYTE	4.0.5.0.5.0.0
7075		072101			DF65=DF64		
7076		072063			DF66=DF54		
7077		071367			DF67=DF21		
7078	072110	004	000	005	DF70:	.BYTE	4.0.5.0.5.5.5.5.4.5.4.5.5.5.5.4.5.4

7079		072110			DF71=DF70	
7080	072132	004	000	005	DF72: .BYTE	4.0.5.0.5.0.0
7081		072063			DF73=DF54	
7082		071367			DF74=DF21	
7083		072110			DF75=DF70	
7084		072110			DF76=DF70	
7085		072132			DF77=DF72	
7086		072063			DF100=DF54	
7087		072110			DF101=DF70	
7088		072132			DF102=DF72	
7089		072110			DF103=DF70	
7090		072063			DF104=DF54	
7091		072110			DF105=DF70	
7092		072132			DF106=DF72	
7093		072110			DF107=DF70	
7094		072063			DF110=DF54	
7095	072141	004	000	005	DF111: .BYTE	4.0.5.0
7096		072141			DF112=DF111	
7097		072141			DF113=DF111	
7098		072141			DF114=DF111	
7099		072141			DF115=DF111	
7100		072141			DF116=DF111	
7101		071256			DF117=DF3	
7102		071256			DF120=DF3	
7103		071775			DF121=DF47	
7104		072012			DF122=DF50	
7105		072036			DF123=DF51	
7106	072145	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
7107		072145			DF125=DF124	
7108		072141			DF126=DF111	
7109		072141			DF127=DF111	
7110		071256			DF130=DF3	
7111	072176	004	000	005	DF131: .BYTE	4.0.5.0.5.5.5.3.5.5.5.3.5.5.5.3
7112		072176			DF132=DF131	
7113	072216	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
7114		072216			DF134=DF133	
7115		072216			DF135=DF133	
7116		072216			DF136=DF133	
7117	072242	004	000	005	DF137: .BYTE	4.0.5.0.5.0.5.0
7118		072242			DF140=DF137	
7119		072242			DF141=DF137	
7120		072242			DF142=DF137	
7121		072216			DF143=DF133	
7122		072216			DF144=DF133	
7123		072242			DF145=DF137	
7124		072242			DF146=DF137	
7125		072216			DF147=DF133	
7126		072216			DF150=DF133	
7127		072242			DF151=DF137	
7128		072216			DF152=DF133	
7129		072242			DF153=DF137	
7130	072252	004	000	005	DF154: .BYTE	4.0.5.0
7131		072252			DF155=DF154	
7132		072216			DF156=DF133	
7133		072216			DF157=DF133	
7134		072216			DF160=DF133	
7135	072256	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

Line	Code	DF	DF	DF	DF	DF
7136		072216				DF162=DF133
7137		072256				DF163=DF161
7138		071256				DF164=DF3
7139		071256				DF165=DF3
7140		072216				DF166=DF133
7141		072216				DF167=DF133
7142		072256				DF170=DF161
7143		072256				DF171=DF161
7144		072216				DF172=DF133
7145		072216				DF173=DF133
7146		072256				DF174=DF161
7147		072256				DF175=DF161
7148		072216				DF176=DF133
7149	072302	004	000	000		DF177: .BYTE 4,0,0
7150		072216				DF200=DF133
7151		072216				DF201=DF133
7152		072216				DF202=DF133
7153		072216				DF203=DF133
7154		072216				DF204=DF133
7155		071256				DF205=DF3
7156		072216				DF206=DF133
7157		072216				DF207=DF133
7158		072216				DF210=DF133
7159	072305	004	000	005		DF211: .BYTE 4,0,5,0,5,0
7160		072305				DF212=DF211
7161		072305				DF213=DF211
7162		072216				DF214=DF133
7163						.EVEN

7164						.SBTTL	DATA TABLES
7165	072314	001232	001234	042740	DT1:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
7166	072330	001242	001244	000000		.WORD	\$TMP4,\$TMP5,0
7167	072336	001232	001234	042740	DT2:	.WORD	\$TMP0,\$TMP1,\$TAB,AERFLG,\$TAB,\$TMP2,\$TAB,\$TMP3,0
7168	072360	001232	001234	042740	DT3:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
7169	072374	042740	001242	000000		.WORD	\$TAB,\$TMP4,0
7170		072360			DT4=DT3		
7171	072402	001232	001234	042740	DT5:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,LF IEX1,\$TMP21,LF IEX2
7172	072420	001272	043057	001240		.WORD	\$TMP20,FPSMS,\$TMP3,FECMS,\$TMP4,0
7173	072434	001232	001234	042740	DT6:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,LF IEX1,\$TMP21,LF IEX2,\$TMP20,0
7174		072434			DT7=DT6		
7175		072434			DT10=DT6		
7176		072434			DT11=DT6		
7177	072456	043134	001252	043141	DT12:	.WORD	\$THE,\$TMP10,NOOP1,NOOP15,NOOP2,\$TMP5
7178	072472	043302	001246	043314		.WORD	NOOP3,\$TMP6,NOOP4,NOOP2,\$TMP5,NOOP3,\$TMP7,NOOP5,\$TMP11
7179	072514	043357	001252	043377		.WORD	NOOP6,\$TMP10,NOOP7,NOOP10,\$TMP0,\$TMP1,\$TAB,\$TMP2
7180	072534	042740	001240	001242		.WORD	\$TAB,\$TMP3,\$TMP4,0
7181	072544	043134	001252	043141	DT13:	.WORD	\$THE,\$TMP10,NOOP1,NOOP11,NOOP10,\$TMP0,\$TMP1,\$TAB
7182	072564	001236	042740	001240		.WORD	\$TMP2,\$TAB,\$TMP3,\$TMP4,0
7183		072434			DT14=DT6		
7184		072434			DT15=DT6		
7185	072576	001232	001234	042740	DT16:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP5,\$TMP3,0
7186		072360			DT17=DT3		
7187	072616	001232	001234	042740	DT20:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
7188	072632	042740	001242	000000		.WORD	\$TAB,\$TMP4,0
7189	072640	001232	001234	042740	DT21:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,0
7190	072652	066014	001313		DT22:	.WORD	DH3,\$CRLF
7191	072656	001232	001234	042740		.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
7192	072672	042740	001242	000000		.WORD	\$TAB,\$TMP4,0
7193	072700	001232	001234	042740	DT23:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
7194	072714	042740	001242	000000		.WORD	\$TAB,\$TMP4,0
7195	072722	044033			DT24:	.WORD	ILLMS
7196	072724	001232	001234	042740		.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
7197	072740	042740	001242	000000		.WORD	\$TAB,\$TMP4,0
7198	072746	001232	001234	042740	DT25:	.WORD	\$TMP0,\$TMP1,\$TAB,\$CRLF,MS1,MS3,\$TMP3,MS4,\$TMP4,\$CRLF
7199	072772	001242	001313	044113		.WORD	\$TMP4,\$CRLF,MS2,MS3,\$TMP5,MS4,\$TMP6,\$CRLF,\$TMP5,0
7200	073016	001232	001234	042740	DT26:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3,\$TMP4,\$CRLF
7201	073036	044223	001313	044241		.WORD	MS6,\$CRLF,MS7,\$TMP5,MS10,\$TMP6,\$CRLF
7202	073054	044270	001313	044241		.WORD	MS11,\$CRLF,MS7,\$TMP5,MS10,\$TMP7,0
7203	073072	001232	001234	042740	DT27:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
7204	073106	001242	001313	044315		.WORD	\$TMP4,\$CRLF,MS12,\$CRLF,MS7,\$TMP5,MS10,\$TMP7,\$CRLF,MS13,0
7205	073134	044456	001272	001313	DT30:	.WORD	MS15,\$TMP20,\$CRLF,MS14,\$CRLF
7206	073146	001232	001234	042740		.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
7207	073162	001242	000000			.WORD	\$TMP4,0
7208		073016			DT31=DT26		
7209		073072			DT32=DT27		
7210	073166	001232	001234	042740	DT33:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
7211	073202	001313	044101	044120		.WORD	\$CRLF,MS1,MS3,\$TMP4,MS4,\$TMP5,\$CRLF,\$TMP6,\$CRLF
7212	073224	044113	044120	001242		.WORD	MS2,MS3,\$TMP4,MS4,\$TMP5,\$CRLF,\$TMP4,0
7213	073244	001232	001234	042740	DT34:	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
7214	073260	001313	044223	001313		.WORD	\$CRLF,MS6,\$CRLF,MS7,\$TMP5,MS10,\$TMP6,\$CRLF
7215	073300	044270	001313	044241		.WORD	MS11,\$CRLF,MS7,\$TMP5,MS10,\$TMP7,0
7216		073244			DT35=DT34		
7217		073244			DT36=DT34		
7218		073244			DT37=DT34		
7219		073166			DT40=DT33		
7220	073316	001272	001313	066014	DT41:	.WORD	\$TMP20,\$CRLF,DH3,\$CRLF

7221	073326	001232	001234	042740	.WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
7222	073342	042740	001242	000000	.WORD	\$TAB,\$TMP4,0
7223	073350	001232	001234	042740	DT42: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
7224	073364	001313	044223	001313	.WORD	\$CRLF,MS6,\$CRLF,MS7,\$TMP5,MS10,\$TMP6,\$TMP15,\$TMP10
7225	073406	001313	044270	001313	.WORD	\$CRLF,MS11,\$CRLF,MS7,\$TMP5,MS10,\$TMP7,0
7226		073350			DT43=DT42	
7227	073426	044510	001244	001313	DT44: .WORD	MS17,\$TMP5,\$CRLF,MS20,\$CRLF,\$TMP0,\$TMP1,\$TAB,\$TMP2
7228	073450	001313	044101	044641	.WORD	\$CRLF,MS1,MS21,\$CRLF,\$TMP3,\$CRLF,MS2,MS21,\$CRLF,\$TMP4,0
7229	073476	044510	001244	001313	DT45: .WORD	MS17,\$TMP5,\$CRLF,MS24,\$CRLF,\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB
7230	073522	001246	001313	044700	.WORD	\$TMP6,\$CRLF,MS22,MS21,\$CRLF,\$TMP3,\$CRLF
7231	073540	044730	044641	001313	.WORD	MS23,MS21,\$CRLF,\$TMP4,0
7232	073552	001232	001234	001313	DT46: .WORD	\$TMP0,\$TMP1,\$CRLF,MS25,MS30,\$TMP2,MS31,\$TMP3
7233	073572	045136	001242	045145	.WORD	MS32,\$TMP4,MS33,\$TMP5,MS34,\$TMP6,\$CRLF,MS26
7234	073612	045120	001250	045127	.WORD	MS30,\$TMP7,MS31,\$TMP10
7235	073622	045136	001254	045145	.WORD	MS32,\$TMP11,MS33,\$TMP12,MS34,\$TMP13,0
7236	073640	001232	001234	042740	DT47: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$CRLF,MS12,MS7,\$TMP3,MS10
7237	073662	001242	001313	044353	.WORD	\$TMP4,\$CRLF,MS13,0
7238		073244			DT50=DT34	
7239	073672	001232	001234	042740	DT51: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
7240	073706	001313	045052	045111	.WORD	\$CRLF,MS25,MS27,\$TMP4,\$CRLF,MS26,MS27,\$TMP5,0
7241		073640			DT52=DT47	
7242	073730	001232	001234	042740	DT53: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
7243	073744	001242	000000		.WORD	\$TMP4,0
7244	073750	001232	001234	042740	DT54: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$CRLF,MS1,MS21,\$CRLF,\$TMP3
7245	073772	001313	044113	044641	.WORD	\$CRLF,MS2,MS21,\$CRLF,\$TMP3,0
7246		073640			DT55=DT47	
7247		073730			DT56=DT53	
7248		073750			DT57=DT54	
7249		073730			DT60=DT53	
7250		073750			DT61=DT54	
7251		073750			DT62=DT54	
7252		073750			DT63=DT54	
7253	074006	001232	001234	042740	DT64: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3
7254	074022	001242	000000		.WORD	\$TMP4,0
7255		074006			DT65=DT64	
7256		073750			DT66=DT54	
7257		072640			DT67=DT21	
7258	074026	001232	001234	042740	DT70: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$CRLF,MS6,\$CRLF,MS7,\$TMP5
7259	074050	044255	001246	001313	.WORD	MS10,\$TMP6,\$CRLF,MS11,\$CRLF,MS7,\$TMP5,MS10,\$TMP7,0
7260		074026			DT71=DT70	
7261	074074	001232	001234	042740	DT72: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3,\$TMP4,0
7262		073750			DT73=DT54	
7263		072640			DT74=DT21	
7264		074026			DT75=DT70	
7265		074026			DT76=DT70	
7266		074074			DT77=DT72	
7267		073750			DT100=DT54	
7268		074026			DT101=DT70	
7269		074026			DT102=DT71	
7270		074026			DT103=DT70	
7271		073750			DT104=DT54	
7272		074026			DT105=DT70	
7273		074074			DT106=DT72	
7274		074026			DT107=DT70	
7275		073750			DT110=DT54	
7276	074114	001232	001234	042740	DT111: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,0
7277		074114			DT112=DT111	

7278		074114			DT113=DT111	
7279		074114			DT114=DT111	
7280		074114			DT115=DT111	
7281		074114			DT116=DT111	
7282		072360			DT117=DT3	
7283		072360			DT120=DT3	
7284		073640			DT121=DT47	
7285		073244			DT122=DT34	
7286		073672			DT123=DT51	
7287	074126	001232	001234	042740	DT124: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3,\$TMP4,\$CRLF
7288	074146	044223	001313	044241	.WORD	MS6,\$CRLF,MS7,\$TMP5,MS10,\$TMP6,\$CRLF
7289	074164	044270	001313	044241	.WORD	MS11,\$CRLF,MS7,\$TMP5,MS10,\$TMP7,\$CRLF,MS37,\$CRLF,\$TMP10,0
7290		074126			DT125=DT124	
7291		074114			DT126=DT111	
7292		074114			DT127=DT111	
7293		072360			DT130=DT3	
7294	074212	001232	001234	042740	DT131: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$CRLF,MS37,\$CRLF,\$TMP3
7295	074232	001313	045215	001313	.WORD	\$CRLF,MS40,\$CRLF,\$TMP4,\$CRLF,MS415,\$CRLF,\$TMP5,0
7296		074212			DT132=DT131	
7297	074254	001232	001234	042740	DT133: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$CRLF,MS41,\$CRLF,\$TMP3
7298	074274	001313	045275	001313	.WORD	\$CRLF,MS42,\$CRLF,\$TMP4,\$CRLF,MS43,\$CRLF,\$TMP5
7299	074314	001313	045330	001313	.WORD	\$CRLF,MS44,\$CRLF,\$TMP6,0
7300		074254			DT134=DT133	
7301		074254			DT135=DT133	
7302		074254			DT136=DT133	
7303	074326	001232	001234	042740	DT137: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TMP10,\$TAB,\$TMP11,0
7304		074326			DT140=DT137	
7305		074326			DT141=DT137	
7306		074326			DT142=DT137	
7307		074254			DT143=DT133	
7308		074254			DT144=DT133	
7309		074326			DT145=DT137	
7310		074326			DT146=DT137	
7311		074254			DT147=DT133	
7312		074254			DT150=DT133	
7313		074326			DT151=DT137	
7314		074254			DT152=DT133	
7315		074326			DT153=DT137	
7316	074346	001232	001234	042740	DT154: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,0
7317		074346			DT155=DT154	
7318		074254			DT156=DT133	
7319		074254			DT157=DT133	
7320		074254			DT160=DT133	
7321		074254			DT161=DT133	
7322		074254			DT162=DT133	
7323		074254			DT163=DT133	
7324		072360			DT164=DT3	
7325		072360			DT165=DT3	
7326		074254			DT166=DT133	
7327		074254			DT167=DT133	
7328		074254			DT170=DT133	
7329		074254			DT171=DT133	
7330		074254			DT172=DT133	
7331		074254			DT173=DT133	
7332		074254			DT174=DT133	
7333		074254			DT175=DT133	
7334		074254			DT176=DT133	

DATA TABLES

7335	074360	001232	001234	037142	DT177: .WORD	\$TMP0,\$TMP1,CPSAVE,0
7336		074254			DT200=DT133	
7337		074254			DT201=DT133	
7338		074254			DT202=DT133	
7339		074254			DT203=DT133	
7340		074254			DT204=DT133	
7341		072360			DT205=DT3	
7342		074254			DT206=DT133	
7343		074254			DT207=DT133	
7344		074254			DT210=DT133	
7345	074370	001232	001234	042740	DT211: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,\$TAB,\$TMP3,0
7346	074406	001232	001234	042740	DT212: .WORD	\$TMP0,\$TMP1,\$TAB,\$TMP2,0
7347		074406			DT213=DT212	
7348		074254			DT214=DT133	
7349		000001			.END	

SYMBOL TABLE

AADATO	027404	ADDW12=	000000	A6	004704	BERR1	005140	CCP2	031212
AADONE	027504	ADDW13=	000000	A7	004730	BITSTS	014250	CCP3	031222
AAERRO	027036	ADDW14=	000000	BBDATO	032526	BIT0 =	000001	CCP4	031232
AAERR1	027124	ADDW15=	000000	BBDONE	032656	BIT00 =	000001	CCP5	031242
AAERR2	027160	ADDW2 =	000000	BBERO	032214	BIT01 =	000002	CCP6	031252
AAERR3	027214	ADDW3 =	000000	BBER1	032254	BIT02 =	000004	CCP7	031262
AAERR4	027224	ADDW4 =	000000	BBER10	032234	BIT03 =	000010	CC1	027510
AAERR5	027260	ADDW5 =	000000	BBER11	032270	BIT04 =	000020	CC10	027734
AAERR6	027314	ADDW6 =	000000	BBER2	032312	BIT05 =	000040	CC11	027744
AAERR7	027350	ADDW7 =	000000	BBER3	032330	BIT06 =	000100	CC12	027752
AAER10	027132	ADDW8 =	000000	BBER4	032364	BIT07 =	000200	CC13	027764
AAPATO	027414	ADDW9 =	000000	BBER40	032400	BIT08 =	000400	CC14	030020
AAPAT1	027424	ADEVCT=	000000	BBER5	032420	BIT09 =	001000	CC15	030044
AAPAT2	027434	ADEVVM =	000000	BBER6	032436	BIT1 =	000002	CC16	030064
AAPAT3	027444	ADONE	005046	BBER7	032472	BIT10 =	002000	CC17	030074
AAPAT4	027454	AENV =	000000	BBER8	032510	BIT11 =	004000	CC18	030102
AAPAT5	027464	AENVVM =	000000	BBPAT0	032536	BIT12 =	010000	CC19	030114
AAPAT6	027474	AERFLG	004736	BBPAT1	032546	BIT13 =	020000	CC2	027544
AA1	026442	AERR1	004740	BBPAT2	032556	BIT14 =	040000	CC20	030150
AA10	026562	AERR2	004752	BBPAT3	032566	BIT15 =	100000	CC21	030174
AA11	026616	AERR3	005004	BBPAT4	032576	BIT2 =	000004	CC22	030214
AA12	026620	AFATAL=	000000	BBPAT5	032606	BIT3 =	000010	CC23	030224
AA13	026636	AMADR1=	000000	BBPAT6	032616	BIT4 =	000020	CC24	030232
AA14	026656	AMADR2=	000000	BBP10	032636	BIT5 =	000040	CC25	030244
AA15	026676	AMADR3=	000000	BBP11	032646	BIT6 =	000100	CC26	030300
AA16	026704	AMADR4=	000000	BBP7	032626	BIT7 =	000200	CC27	030322
AA17	026710	AMAMS1=	000000	BB1	031326	BIT8 =	000400	CC28	030342
AA2	026504	AMAMS2=	000000	BB10	031512	BIT9 =	001000	CC29	030352
AA20	026714	AMAMS3=	000000	BB11	031532	BPTVEC=	000014	CC3	027566
AA21	026716	AMAMS4=	000000	BB12	031542	B1	005064	CC30	030360
AA22	026754	AMSGAD=	000000	BB13	031550	B2	005066	CC31	030372
AA23	026756	AMSGLG=	000000	BB14	031562	B3	005104	CC32	030426
AA24	026776	AMSGTY=	000000	BB15	031616	CCDATO	031162	CC33	030450
AA25	027016	AMTYP1=	000000	BB16	031642	CCDONE	031322	CC34	030470
AA26	027024	AMTYP2=	000000	BB17	031652	CCERO	030524	CC35	030500
AA27	027030	AMTYP3=	000000	BB2	031370	CCER1	030560	CC36	030506
AA3	026506	AMTYP4=	000000	BB20	031664	CCER10	031072	CC37	030520
AA4	026524	APASS =	000000	BB21	031720	CCER11	031110	CC4	027606
AA5	026544	APRIOR=	000000	BB22	031744	CCER12	031126	CC5	027616
AA6	026552	APTCSU=	000040	BB23	031764	CCER13	031144	CC6	027624
AA7	026560	APTENV=	000001	BB24	031774	CCER2	030616	CC7	027636
ABASE =	000000	APTSIZ=	000200	BB25	032002	CCER22	030632	CC8	027672
ACDW1 =	000000	APTSPO=	000100	BB26	032014	CCER3	030654	CC9	027714
ACDW2 =	000000	ASWREG=	000000	BB27	032050	CCER4	030672	CDONE	006120
ACPUOP=	000000	AATESTN=	000000	BB3	031374	CCER44	030706	CERR1	005654
AC0 =	%000000	AUNIT =	000000	BB30	032072	CCER5	030730	CERR2	005752
AC1 =	%000001	AUSWR =	000000	BB31	032102	CCER50	030574	CERR3	006052
AC2 =	%000002	AVECT1=	000000	BB32	032114	CCER55	030744	CERR4	006074
AC3 =	%000003	AVECT2=	000000	BB33	032150	CCER6	030764	CFCC1	043671
AC4 =	%000004	A05	004654	BB34	032172	CCER7	031020	CHPNUM	014150
AC5 =	%000005	A1	004526	BB35	032202	CCER8	031036	CKSWR =	104407
AC6 =	%000006	A11	004526	BB4	031412	CCER90	030542	CNT =	000214
AC7 =	%000007	A12	004542	BB5	031422	CCP0	031172	CPC	006116
ADDW0 =	000000	A2	004600	BB6	031434	CCP1	031202	CPSAVE	037142
ADDW1 =	000000	A3	004604	BB7	031470	CCP10	031272	CPSPUR	042566
ADDW10=	000000	A4	004652	BDONE	005154	CCP11	031302	CPTWO	042604
ADDW11=	000000	A5	004656	BERR	005110	CCP12	031312	CR =	000015

CRLF = 000200	DD21 033340	DF130 = 071256	DF212 = 072305	DH101 = 066201
C1 005174	DD22 033352	DF131 = 072176	DF213 = 072305	DH102 = 067073
C15 005222	DD23 033406	DF132 = 072176	DF214 = 072216	DH103 = 067031
C2 005246	DD24 033430	DF133 = 072216	DF22 071373	DH104 = 067503
C25 005262	DD25 033450	DF134 = 072216	DF23 071405	DH105 = 066201
C3 005310	DD26 033460	DF135 = 072216	DF24 071415	DH106 = 067073
C35 005340	DD27 033466	DF136 = 072216	DF25 071424	DH107 = 067031
C4 005366	DD3 032740	DF137 = 072242	DF26 071450	DH11 = 066201
C45 005400	DD30 033500	DF14 = 071266	DF27 071475	DH110 = 067503
C5 005426	DD31 033534	DF140 = 072242	DF3 071256	DH111 = 067031
C55 005454	DD32 033556	DF141 = 072242	DF30 071515	DH112 = 067701
C6 005500	DD33 033576	DF142 = 072242	DF31 = 071450	DH113 = 067031
C65 005514	DD34 033606	DF143 = 072216	DF32 = 071475	DH114 = 067701
C7 005542	DD35 033614	DF144 = 072216	DF33 071531	DH115 = 070120
C75 005572	DD36 033632	DF145 = 072242	DF34 071557	DH116 = 070120
C8 005620	DD37 033666	DF146 = 072242	DF35 = 071557	DH117 = 066321
C85 005632	DD38 033710	DF147 = 072216	DF36 = 071557	DH12 = 000000
DDDATO 034556	DD39 033730	DF15 = 071266	DF37 = 071557	DH120 = 070404
DDDONE 034706	DD4 032760	DF150 = 072216	DF4 = 071256	DH121 = 066201
DDERO 033770	DD40 033740	DF151 = 072242	DF40 071603	DH122 = 067431
DDER1 034006	DD41 033746	DF152 = 072216	DF41 071631	DH123 = 067431
DDER10 034424	DD42 033764	DF153 = 072242	DF42 071645	DH124 = 066321
DDER11 034462	DD5 032770	DF154 072252	DF43 = 071645	DH125 = 066321
DDER12 034520	DD6 032776	DF155 = 072252	DF44 071673	DH126 = 066201
DDER2 034044	DD7 033014	DF156 = 072216	DF45 071716	DH127 = 067503
DDER3 034102	DD8 033050	DF157 = 072216	DF46 071743	DH13 = 000000
DDER4 034140	DERR1 006246	DF16 071350	DF47 071775	DH130 = 070404
DDER5 034176	DERR2 006342	DF160 = 072216	DF5 071266	DH131 = 067503
DDER6 034234	DF1 071236	DF161 072256	DF50 072012	DH132 = 067503
DDER7 034272	DF10 = 071266	DF162 = 072216	DF51 072036	DH133 070474
DDER8 034330	DF100 = 072063	DF163 = 072256	DF52 = 071775	DH134 = 070474
DDER9 034366	DF101 = 072110	DF164 = 071256	DF53 072054	DH135 = 070474
DDISP = 177570	DF102 = 072132	DF165 = 071256	DF54 072063	DH136 = 070474
DDONE 006366	DF103 = 072110	DF166 = 072216	DF55 = 071775	DH137 070604
DDP0 034566	DF104 = 072063	DF167 = 072216	DF56 = 072054	DH14 = 066201
DDP1 034576	DF105 = 072110	DF17 = 071256	DF57 = 072063	DH140 = 070604
DDP2 034606	DF106 = 072132	DF170 = 072256	DF6 = 071266	DH141 = 070604
DDP3 034616	DF107 = 072110	DF171 = 072256	DF60 = 072054	DH142 = 070604
DDP4 034626	DF11 = 071266	DF172 = 072216	DF61 = 072063	DH143 = 070474
DDP5 034636	DF110 = 072063	DF173 = 072216	DF62 = 072063	DH144 = 070474
DDP6 034646	DF111 072141	DF174 = 072256	DF63 = 072063	DH145 = 070604
DDP7 034656	DF112 = 072141	DF175 = 072256	DF64 072101	DH146 = 070604
DDP8 034666	DF113 = 072141	DF176 = 072216	DF65 = 072101	DH147 = 067503
DDP9 034676	DF114 = 072141	DF177 072302	DF66 = 072063	DH15 = 066201
DD1 032662	DF115 = 072141	DF2 071246	DF67 = 071367	DH150 070774
DD10 033072	DF116 = 072141	DF20 071357	DF7 = 071266	DH151 = 070604
DD11 033102	DF117 = 071256	DF200 = 072216	DF70 072110	DH152 = 070474
DD12 033120	DF12 071302	DF201 = 072216	DF71 = 072110	DH153 = 070604
DD13 033154	DF120 = 071256	DF202 = 072216	DF72 072132	DH154 071065
DD14 033176	DF121 = 071775	DF203 = 072216	DF73 = 072063	DH155 = 071065
DD15 033206	DF122 = 072012	DF204 = 072216	DF74 = 071367	DH156 = 067503
DD16 033224	DF123 = 072036	DF205 = 071256	DF75 = 072110	DH157 = 067503
DD17 033260	DF124 072145	DF206 = 072216	DF76 = 072110	DH16 066241
DD18 033302	DF125 = 072145	DF207 = 072216	DF77 = 072132	DH160 = 067503
DD19 033322	DF126 = 072141	DF21 071367	DH1 065631	DH161 = 067503
DD2 032716	DF127 = 072141	DF210 = 072216	DH10 = 066201	DH162 = 067503
DD20 033332	DF13 071334	DF211 072305	DH100 = 067503	DH163 = 067503

DH164 = 066321	DH53 = 067073	DT131 = 074212	DT213 = 074406	D4 = 006204
DH165 = 066321	DH54 = 067503	DT132 = 074212	DT214 = 074254	D5 = 006216
DH166 = 067503	DH55 = 066201	DT133 = 074254	DT22 = 072652	D6 = 006232
DH167 = 067503	DH56 = 067073	DT134 = 074254	DT23 = 072700	D7 = 006242
DH17 = 066321	DH57 = 067503	DT135 = 074254	DT24 = 072722	DR = 006312
DH170 = 067503	DH6 = 066201	DT136 = 074254	DT25 = 072746	D9 = 006324
DH171 = 067503	DH60 = 067073	DT137 = 074326	DT26 = 073016	EDONE = 006532
DH172 = 067503	DH61 = 067503	DT14 = 072434	DT27 = 073072	EEDATO = 035410
DH173 = 067503	DH62 = 067503	DT140 = 074326	DT3 = 072360	EEDONE = 035472
DH174 = 067503	DH63 = 067503	DT141 = 074326	DT30 = 073134	EEERO = 035202
DH175 = 067503	DH64 = 067613	DT142 = 074326	DT31 = 073016	EEER1 = 035220
DH176 = 067503	DH65 = 067544	DT143 = 074254	DT32 = 073072	EEER2 = 035256
DH177 = 071125	DH66 = 067503	DT144 = 074254	DT33 = 073166	EEER3 = 035314
DH2 = 065721	DH67 = 066201	DT145 = 074326	DT34 = 073244	EEER4 = 035352
DH20 = 066411	DH7 = 066201	DT146 = 074326	DT35 = 073244	EEPO = 035420
DH200 = 067503	DH70 = 066201	DT147 = 074254	DT36 = 073244	EEP1 = 035432
DH201 = 067503	DH71 = 067031	DT15 = 072434	DT37 = 073244	EEP2 = 035442
DH202 = 067503	DH72 = 067073	DT150 = 074254	DT4 = 072360	EEP3 = 035452
DH203 = 067503	DH73 = 067503	DT151 = 074326	DT40 = 073166	EEP4 = 035462
DH204 = 067503	DH74 = 066201	DT152 = 074254	DT41 = 073316	EERRO = 006450
DH205 = 066321	DH75 = 066201	DT153 = 074326	DT42 = 073350	EERR1 = 006466
DH206 = 067503	DH76 = 067031	DT154 = 074346	DT43 = 073350	EERR2 = 006502
DH207 = 067503	DH77 = 067073	DT155 = 074346	DT44 = 073426	EE1 = 034712
DH21 = 066201	DISPLA = 001142	DT156 = 074254	DT45 = 073476	EE10 = 035142
DH210 = 067503	DISPRE = 000174	DT157 = 074254	DT46 = 073552	EE11 = 035152
DH211 = 071171	DPAT3 = 017500	DT16 = 072576	DT47 = 073640	EE12 = 035160
DH212 = 066201	DSWR = 177570	DT160 = 074254	DT5 = 072402	EE13 = 035176
DH213 = 066201	DT1 = 072314	DT161 = 074254	DT50 = 073244	EE2 = 034746
DH214 = 067503	DT10 = 072434	DT162 = 074254	DT51 = 073672	EE3 = 034770
DH22 = 066477	DT100 = 073750	DT163 = 074254	DT52 = 073640	EE4 = 035010
DH23 = 066534	DT101 = 074026	DT164 = 072360	DT53 = 073730	EE5 = 035020
DH24 = 066672	DT102 = 074026	DT165 = 072360	DT54 = 073750	EE6 = 035026
DH25 = 067031	DT103 = 074026	DT166 = 074254	DT55 = 073640	EE7 = 035044
DH26 = 067073	DT104 = 073750	DT167 = 074254	DT56 = 073730	EE8 = 035100
DH27 = 067073	DT105 = 074026	DT17 = 072360	DT57 = 073750	EE9 = 035122
DH3 = 066014	DT106 = 074074	DT170 = 074254	DT6 = 072434	EMTVEC = 000030
DH30 = 000000	DT107 = 074026	DT171 = 074254	DT60 = 073730	EM1 = 045352
DH31 = 067073	DT11 = 072434	DT172 = 074254	DT61 = 073750	EM10 = 045560
DH32 = 067073	DT110 = 073750	DT173 = 074254	DT62 = 073750	EM100 = 054462
DH33 = 067161	DT111 = 074114	DT174 = 074254	DT63 = 073750	EM101 = 054522
DH34 = 067161	DT112 = 074114	DT175 = 074254	DT64 = 074006	EM102 = 054646
DH35 = 067161	DT113 = 074114	DT176 = 074254	DT65 = 074006	EM103 = 054720
DH36 = 067161	DT114 = 074114	DT177 = 074360	DT66 = 073750	EM104 = 055023
DH37 = 067161	DT115 = 074114	DT2 = 072336	DT67 = 072640	EM105 = 055064
DH4 = 066105	DT116 = 074114	DT20 = 072616	DT7 = 072434	EM106 = 055211
DH40 = 067161	DT117 = 072360	DT200 = 074254	DT70 = 074026	EM107 = 055264
DH41 = 000000	DT12 = 072456	DT201 = 074254	DT71 = 074026	EM11 = 045701
DH42 = 067264	DT120 = 072360	DT202 = 074254	DT72 = 074074	EM110 = 055370
DH43 = 067264	DT121 = 073640	DT203 = 074254	DT73 = 073750	EM111 = 055432
DH44 = 000000	DT122 = 073244	DT204 = 074254	DT74 = 072640	EM112 = 055432
DH45 = 067366	DT123 = 073672	DT205 = 072360	DT75 = 074026	EM113 = 055534
DH46 = 067405	DT124 = 074126	DT206 = 074254	DT76 = 074026	EM114 = 055534
DH47 = 067366	DT125 = 074126	DT207 = 074254	DT77 = 074074	EM115 = 055432
DH5 = 066201	DT126 = 074114	DT21 = 072640	D1 = 006152	EM116 = 055534
DH50 = 067431	DT127 = 074114	DT210 = 074254	D10 = 006354	EM117 = 055636
DH51 = 067431	DT13 = 072544	DT211 = 074370	D2 = 006176	EM12 = 000000
DH52 = 066201	DT130 = 072360	DT212 = 074406	D3 = 006200	EM120 = 055772

EM121	056126	EM203	064671	EM73	053670	FFP0	036010	GFLAG1	015132
EM122	056245	EM204	065026	EM74	053730	FFP1	036020	GFLAG2	015134
EM123	056344	EM205	065163	EM75	054161	FFP2	036030	GORO	015166
EM124	056405	EM206	065230	EM76	054304	FFP3	036040	GOR1	015170
EM125	056500	EM207	065275	EM77	054406	FFP4	036050	COR2	015172
EM126	056570	EM21	046712	EPENDS	036512	FF1	035476	GOR3	015174
EM127	056777	EM210	065417	ERM10	037502	FFi0	035656	GPAT00	015136
EM13 =	000000	EM211 =	045560	ERROR =	104000	FF11	035666	GPAT01	015140
EM130	057212	EM212 =	045614	ERRVEC =	000004	FF2	035532	GPAT02	015142
EM131	057312	EM213 =	045646	ERTYPE	042010	FF3	035554	GPAT03	015144
EM132	057352	EM214	065474	ERT1	042174	FF4	035562	GPAT10	015146
EM133	057412	EM22	047043	ERT2	042360	FF5	035572	GPAT11	015150
EM134	057451	EM23 =	047043	ERT3	042364	FF6	035626	GPAT12	015152
EM135	057510	EM24 =	047043	ERT4	042374	FF7	035650	GPAT13	015154
EM136	057547	EM25	047130	ERT5	042406	FPSMS	043057	GRESET	014346
EM137 =	057412	EM26	047243	E1	006406	FPSPUR	042534	GSETUP	014270
EM14	045762	EM27 =	047243	E2	006422	FPVECT =	000244	GSUM	014500
EM140 =	057451	EM3	045453	E3	006422	FXDAT0	010416	GS1	014320
EM141 =	057510	EM30	047311	E4	006424	FXDAT1	010420	GTSWR =	104406
EM142 =	057547	EM31	047363	FDATIO	010352	FXDAT2	010422	G1	012176
EM143	057606	EM32 =	047363	FDATI1	010354	FXDAT3	010424	G10	012504
EM144	057641	EM33	047431	FDATI2	010356	FXDAT4	010426	G11	012506
EM145 =	057606	EM34	047472	FDATI3	010360	FXDAT5	010430	G12	012600
EM146 =	057641	EM35	047574	FDATI4	010362	FXDAT6	010432	G13	012632
EM147	057674	EM36	047676	FDATI5	010364	FXDAT7	010434	G14	012634
EM15	046105	EM37	047777	FDATI6	010366	F1	006536	G15	012726
EM150 =	057674	EM4	045520	FDATI7	010370	F10	006760	G16	012760
EM151 =	057674	EM40	050100	FDAT00	010374	F11	006762	G17	012762
EM152	057726	EM41	050251	FDAT01	010376	F12	007000	G2	012230
EM153 =	057726	EM42	050306	FDAT02	010400	F13	007032	G20	013054
EM154	057760	EM43	050427	FDAT03	010402	F135	007012	G21	013106
EM155	060212	EM44	050550	FDAT04	010404	F14	007042	G22	013110
EM156	060445	EM45 =	050550	FDAT05	010406	F15	007052	G23	013202
EM157	060662	EM46	050613	FDAT06	010410	F16	007062	G24	013234
EM16	046230	EM47	050671	FDAT07	010412	F17	007072	G25	013236
EM160	061101	EM5	045560	FDONE	010436	F2	006574	G26	013330
EM161	061306	EM50	051007	FECMS	043123	F20	007102	G27	013362
EM162	061513	EM51	051105	FERR0	007144	F21	007112	G3	012232
EM163	061560	EM52	051146	FERR1	007202	F22	007122	G30	013364
EM164	061625	EM53	051267	FERR10	007762	F23	007140	G31	013456
EM165	061672	EM54	051464	FERR11	010016	F3	006614	G32	013510
EM166	061737	EM55	051530	FERR2	007300	F4	006616	G33	013512
EM167	062047	EM56	051651	FERR20	010036	F5	006634	G34	013604
EM17	046301	EM57	052046	FERR21	010154	F6	006710	G35	013636
EM170	062306	EM6	045614	FERR25	010204	F7	006740	G36	013640
EM171	062416	EM60	052112	FERR26	010322	GADR	015206	G37	013732
EM172	062655	EM61	052307	FERR3	007340	GAND0	015156	G4	012324
EM173	063114	EM62	052353	FERR4	007332	GAND1	015160	G40	013764
EM174	063353	EM63	052545	FERR5	007436	GAND2	015162	G41	013766
EM175	063612	EM64	052737	FERR6	007472	GAND3	015164	G42	014060
EM176	064051	EM65 =	052737	FERR7	007626	GCMP	014366	G43	014112
EM177	064206	EM66	053073	FER2	007304	GDAT00	015176	G44	014114
EM2	045407	EM67	053136	FFDATO	036000	GDAT01	015200	G5	012356
EM20	046534	EM7	045646	FFDONE	036060	GDAT02	015202	G6	012360
EM200	064242	EM70	053367	FFERO	035670	GDAT03	015204	G7	012452
EM201	064377	EM71	053512	FFER1	035704	GDONE	015210	HADR	015744
EM202	064534	EM72	053614	FFER2	035742	GERR1	014414	HA1R	016020

HA1W	015750	IPAT13	011256	KBUF3	017460	MDAT00	020350	MS44	045330
HA2R	016030	IPAT20	011260	KDAT10	017442	MDAT01	020352	MS5	044161
HA2W	015760	IPAT21	011262	KDAT11	017444	MDAT02	020354	MS6	044223
HA3R	016040	IPAT22	011264	KDAT12	017446	MDAT03	020356	MS7	044241
HA3W	015770	IPAT23	011266	KDAT13	017450	MDONE	020360	M1	020034
HA4R	016050	I1	010450	KDAT00	017462	MERR0	020204	M15	020054
HA4W	016000	I10	010632	KDAT01	017464	MERR1	020242	M2	020060
HA5R	016060	I105	010640	KDAT02	017466	MERR2	020156	M3	020062
HA5W	016010	I106	010634	KDAT03	017470	MERR3	020270	M4	020064
HCLR	015674	I11	010642	KDONE	017502	MNUMBE=	000214	M5	020120
HCLR1	015704	I12	010650	KERR0	017324	MNUM0	044535	M6	020124
HCMP	015636	I13	010664	KERR1	017370	MNUM1	044543	M7	020134
HCMP1	015656	I14	010726	KERR2	017414	MNUM2	044550	M8	020144
HCMP2	015666	I15	010730	KPAT0	017472	MNUM3	044555	M9	020154
HDAT1	016070	I16	010732	KPAT1	017474	MNUM4	044564	NDAT10	021056
HDAT2	016100	I17	010750	KPAT2	017476	MNUM5	044572	NDAT11	021060
HDAT3	016110	I2	010466	K1	017216	MPAT10	020330	NDAT12	021062
HDAT4	016120	I20	011010	K10	017344	MPAT11	020332	NDAT13	021064
HDAT5	016130	I21	011024	K2	017242	MPAT12	020334	NDAT00	021014
HDONE	016140	I22	011030	K3	017244	MPAT13	020336	NDAT01	021016
HERROR	015712	I23	011044	K4	017246	MPAT20	020340	NDAT02	021020
HFLAG	015746	I3	010534	K5	017302	MPAT21	020342	NDAT03	021022
HSTD	015560	I4	010536	K6	017310	MPAT22	020344	NDONE	021066
HT =	000011	I5	010540	K7	017322	MPAT23	020346	NERR0	020556
H1	015222	I6	010564	LDAT10	020006	MS1	044101	NERR1	020656
H10	015464	I7	010600	LDAT11	020010	MS10	044255	NERR10	020610
H11	015516	JBUF0	017144	LDAT12	020012	MS11	044270	NERR11	020622
H12	015550	JBUF1	017146	LDAT13	020014	MS12	044315	NERR2	020712
H2	015242	JBUF2	017150	LDAT00	020020	MS13	044353	NERR20	020664
H3	015252	JBUF3	017152	LDAT01	020022	MS14	044370	NERR3	020722
H4	015324	JDAT10	017154	LDAT02	020024	MS15	044456	NERR4	020732
H5	015346	JDAT11	017156	LDAT03	020026	MS16	044503	NERR5	020742
H6	015400	JDAT12	017160	LDONE	020030	MS17	044510	NERR6	020766
H7	015432	JDAT13	017162	LD1	043567	MS2	044113	NOOP1	043141
IBSAVE	037144	JDAT00	017164	LD2	043617	MS20	044600	NOOP10	043417
IDAT10	011300	JDAT0	017174	LERR1	017642	MS21	044641	NOOP11	043510
IDAT11	011302	JDAT01	017166	LERR2	017714	MS22	044700	NOOP15	043170
IDAT12	011304	JDAT02	017170	LERR3	017666	MS23	044730	NOOP2	043265
IDAT13	011306	JDAT03	017172	LF =	000012	MS24	044757	NOOP3	043302
IDAT00	011270	JDAT1	017176	LF1EX1	042745	MS25	045052	NOOP4	043314
IDAT01	011272	JDAT2	017200	LF1EX2	043015	MS26	045074	NOOP5	043331
IDAT02	011274	JDAT3	017202	LFPS1	043554	MS27	045111	NOOP6	043357
IDAT03	011276	JDONE	017204	LOGP	004456	MS3	044120	NOOP7	043377
IDONE	011310	JERR0	017024	LPAT10	017764	MS30	045120	NPAT10	021046
IERR0	011050	JERR1	017072	LPAT11	017766	MS31	045127	NPAT11	021050
IERR1	011132	JERR2	017116	LPAT12	017770	MS32	045136	NPAT12	021052
IERR2	011152	J1	016716	LPAT13	017772	MS33	045145	NPAT13	021054
IERR25	011174	J10	017044	LPAT20	017774	MS34	045154	NPAT20	021034
IERR3	011224	J2	016742	LPAT21	017776	MS35	045163	NPAT21	021036
IERR4	011200	J3	016744	LPAT22	020000	MS36	045170	NPAT22	021040
ILLMS	044033	J4	016746	LPAT23	020002	MS37	045177	NPAT23	021042
ILL1	043727	J5	017002	L1	017514	MS4	044147	NULL	042737
ILL2	043772	J6	017010	L2	017554	MS40	045215	N1	020364
IOTVEC=	000020	J7	017022	L3	017556	MS41	045251	N10	020516
IPAT10	011250	KBUF0	017452	L4	017560	MS415	045231	N11	020526
IPAT11	011252	KBUF1	017454	L5	017630	MS42	045275	N12	020530
IPAT12	011254	KBUF2	017456	L6	017636	MS43	045313	N13	020544

N14	020554	PDAT02	022242	QDONE	022746	SERR6	016536	S9	016316
N2	020410	PDAT03	022244	QERR0	022414	SERR7	016562	TAB	= 000011
N3	020412	PDONE	022246	QERR1	022660	SETD1	043705	TB!TVE	= C00014
N4	020414	PERR0	021714	QERR11	022424	SETF1	043677	TDAT10	012116
N5	020432	PERR1	022134	QERR12	022442	SETI1	043713	TPAT11	012120
N6	020442	PERR10	021734	QERR13	022460	SETL1	043721	TDAT12	012122
N7	020452	PERR11	021744	QERR14	022476	SPACE	042742	TDAT13	012124
N8	020470	PERR12	021762	QERR15	022514	SPAT10	016664	TDAT00	012106
N9	020502	PERR13	022000	QERR16	022524	SPAT11	016666	TDAT01	012110
ODAT10	021562	PERR14	022016	QERR17	022532	SPAT12	016670	TDAT02	012112
ODAT11	021564	PERR15	022034	QERR2	022614	SPAT13	016672	TDAT03	012114
ODAT12	021566	PERR16	022044	QERR20	022560	STACK	= 001100	TDONE	012126
ODAT13	021570	PERR17	022052	QERR21	022570	START	003616	TERR0	011674
ODAT00	021520	PERR2	022162	QERR22	022576	STFS1	043630	TERR1	011756
ODAT01	021522	PERR20	022100	QERR3	022624	STKLMT	= 177774	TERR2	011776
ODAT02	021524	PERR21	022110	QERR4	022632	STST1	043760	TERR25	012012
ODAT03	021526	PERR22	022116	QPAT10	022706	ST1	043643	TERR3	012042
ODONE	021572	PFECAD	042506	QPAT11	022710	ST2	043660	TERR4	012024
QERR0	021262	PFECDH	042456	QPAT12	022712	SWR	001140	TKVEC	= 000060
QERR1	021362	PFECM	042422	QPAT13	022714	SWREG	000176	TOCTNM	042522
QERR10	021314	PFECFT	042516	QPAT20	022716	SW0	= 000001	TPAT10	012066
QERR11	021326	PFECWS	042412	QPAT21	022720	SW00	= 000001	TPAT11	012070
QERR2	021416	PIRQ	= 177772	QPAT22	022722	SW01	= 000002	TPAT12	012072
QERR20	021370	PIRQVE	= 000240	QPAT23	022724	SW02	= 000004	TPAT13	012074
QERR3	021426	POWERM	042672	Q1	022252	SW03	= 000010	TPAT20	012076
QERR4	021436	PPAT10	022216	Q10	022410	SW04	= 000020	TPAT21	012100
QERR5	021446	PPAT11	022220	Q2	022274	SW05	= 000040	TPAT22	012102
QERR6	021472	PPAT12	022222	Q3	= 022276	SW06	= 000100	TPAT23	012104
OPAT10	021552	PPAT13	022224	Q4	022300	SW07	= 000200	TPVEC	= 000064
OPAT11	021554	PROGNUM	= 000001	Q5	022322	SW08	= 000400	TRAPVE	= 000034
OPAT12	021556	PRO	= 000000	Q6	022344	SW09	= 001000	TRTVEC	= 000014
OPAT13	021560	PR1	= 000040	Q7	022354	SW1	= 000002	TST1	004456
OPAT20	021536	PR2	= 000100	Q8	022366	SW10	= 002000	TST10	011312
OPAT21	021540	PR3	= 000140	Q9	022402	SW11	= 004000	TST11	012130
OPAT22	021542	PR4	= 000200	RDCHR	= 104410	SW12	= 010000	TST12	015212
OPAT23	021544	PR5	= 000240	RESREG	= 104412	SW13	= 020000	TST13	016142
OPAT24	021546	PR6	= 000300	RESVEC	= 000010	SW14	= 040000	TST14	016706
01	021072	PR7	= 000340	RSETUP	= 104413	SW15	= 100000	TST15	017206
010	021222	PS	= 177776	R6	= %000006	SW2	= 000004	TST16	017504
011	021232	PSW	= 177776	R7	= %000007	SW3	= 000010	TST17	020032
012	021234	PWRVEC	= 000024	SADR	016660	SW4	= 000020	TST2	005050
013	021250	P1	021576	SAVREG	= 104411	SW5	= 000040	TST20	020362
014	021260	P2	021620	SCGPE	= 000004	SW6	= 000100	TST21	021070
02	021116	P3	= 021622	SDAT00	016674	SW7	= 000200	TST22	021574
03	021120	P4	021624	SDAT01	016676	SW8	= 000400	TST23	022250
04	021122	P5	021646	SDAT02	016700	SW9	= 001000	TST24	022750
05	021140	P6	021670	SDAT03	016702	S1	016144	TST25	024110
06	021150	P7	021700	SDONE	016704	S10	016322	TST26	024706
07	021160	P8	021710	SERR0	016364	S11	016346	TST27	025522
08	021174	QDAT10	022736	SERR1	016574	S12	016356	TST3	005156
09	021206	QDAT11	022740	SERR10	016404	S2	016204	TST30	026132
PDAT10	022226	QDAT12	022742	SERR15	016464	S3	016206	TST31	026440
PDAT11	022230	QDAT13	022744	SERR2	016524	S4	016212	TST32	027506
PDAT12	022232	QDAT00	022726	SERR20	016504	S5	016236	TST33	031324
PDAT13	022234	QDAT01	022730	SERR3	016550	S6	016246	TST34	032660
PDAT00	022236	QDAT02	022732	SERR4	016442	S7	016254	TST35	034710
PDAT01	022240	QDAT03	022734	SERR5	016626	S8	016314	TST36	035474

TST37	036060	UPAT40	024064	XERR2	025426	Y2	025600	\$CRLF	001313
TST4	006122	UPAT41	024066	XERR3	025372	Y3	025624	\$DBLK	040614
TST5	006370	UPAT42	024070	XERR4	025442	Y4	025642	\$DDW0	001402
TST6	006534	UPAT43	024072	XPAT00	025470	Y5	025702	\$DDW1	001404
TST7	010440	UROM1	024102	XPAT01	025472	Y6	025704	\$DDW10	001426
TYPDS =	104405	UROM2	024104	XPAT02	025474	Y7	025720	\$DDW11	001430
TYPE =	104401	UROM3	024106	XPAT03	025476	ZDAT00	026376	\$DDW12	001432
TYPOC =	104402	UTMP1	024076	XPAT10	025500	ZDAT01	026400	\$DDW13	001434
TYPON =	104404	UTMP2	024100	XPAT12	025504	ZDAT02	026402	\$DDW14	001436
TYPOS =	104403	U0	022766	XPAT13	025506	ZDAT03	026404	\$DDW15	001440
T1	011314	U1	023016	XPAT20	025510	ZDONE	026436	\$DDW2	001406
T10	011462	U10	023374	XPAT21	025512	ZERR1	026302	\$DDW3	001410
T105	011470	U11	023376	XPAT22	025514	ZERR2	026344	\$DDW4	001412
T11	011472	U12	023430	XPAT23	025516	ZFLAG	026370	\$DDW5	001414
T12	011500	U13	023446	XTMP	025456	ZPAT00	026406	\$DDW6	001416
T13	011514	U14	023500	X1	024710	ZPAT01	026410	\$DDW7	001420
T14	011556	U15	023534	X10	025112	ZPAT02	026412	\$DDW8	001422
T15	011560	U16	023536	X11	025126	ZPAT03	026414	\$DDW9	001424
T16	011562	U2	023066	X12	025166	ZPAT10	026416	\$DEVCT	001326
T17	011600	U3	023120	X13	025212	ZPAT11	026420	\$DEVN	001374
T2	011340	U4	023170	X14	025220	ZPAT12	026422	\$DJAGN	036442
T20	011640	U5	023222	X15	025234	ZPAT13	026424	\$DTBL	040604
T21	011654	U6	023272	X16	025274	ZPAT20	026426	\$ENDAD	036432
T22	011656	U7	023324	X17	025320	ZPAT21	026430	\$ENDCT	036124
T23	011670	WDAPO0	024674	X2	024750	ZPAT22	026432	\$ENULL	036506
T3	011406	WDAT01	024676	X20	025326	ZPAT23	026434	\$ENV	001336
T4	011410	WDAT02	024700	X21	025342	ZTMP1	026372	\$ENVM	001337
T5	011412	WDAT03	024702	X3	024774	ZTMP2	026374	\$EOP	036060
T6	011436	WDONE	024704	X4	025002	Z1	026154	\$EOPCT	036112
T7	011452	WPAT00	024664	X5	025020	Z2	026202	\$ERFLG	001103
UDONE	024110	WPAT01	024666	X6	025060	Z3	026226	\$ERMAX	001115
UERR0	023554	WPAT02	024670	X7	025104	Z4	026234	\$EPROR	037146
UERR1	023600	WPAT03	024672	YDAT00	026070	Z5	026246	\$ERRPC	001116
UERR10	023606	WSETUP	024632	YDAT01	026072	Z6	026300	\$ERRTB	001442
UERR11	023650	W1	024112	YDAT02	026074	\$APTHD	003602	\$ERTTL	001112
UERR2	023664	W10	024334	YDAT03	026076	\$ATYC	040650	\$ESCAP	001304
UERR20	023672	W11	024362	YDONE	026130	\$ATY1	040624	\$ETABL	001336
UERR21	023734	W12	024416	YERR1	025730	\$ATY3	040632	\$ETEND	001442
UERR3	023750	W13	024442	YERR2	025772	\$ATY4	040642	\$FATAL	001320
UERR4	024000	W14	024456	YERR3	026034	\$AUTOB	001134	\$FFLG	041070
UFLAG	024074	W15	024504	YFLAG	026060	\$BASE	001372	\$FILLC	001156
UPAT00	024024	W16	024544	YPAT00	026100	\$BDADR	001122	\$FILLS	001155
UPAT01	024026	W17	024570	YPAT01	026102	\$BDDAT	001126	\$GDADR	001120
UPAT02	024030	W2	024146	YPAT02	026104	\$BELL	001306	\$GDDAT	001124
UPAT03	024032	W20	024604	YPAT03	026106	\$CDW1	001376	\$GET42	036404
UPAT10	024034	W3	024172	YPAT10	026110	\$CDW2	001400	\$GTSWR	041142
UPAT11	024036	W4	024210	YPAT11	026112	\$CHARC	040140	\$HD =	000003
UPAT12	024040	W5	024240	YPAT12	026114	\$CKSWR	041072	\$HIBTS	003602
UPAT13	024042	W6	024300	YPAT13	026116	\$CLR.T	036422	\$ICNT	001104
UPAT20	024044	W7	024324	YPAT20	026120	\$CMTAG	001100	\$ILLUP	042002
UPAT21	024046	XAPT11	025502	YPAT21	026122	\$CM1 =	000024	\$INTAG	001135
UPAT22	024050	XDAT00	025460	YPAT22	026124	\$CM2 =	000050	\$ITEMB	001114
UPAT23	024052	XDAT01	025462	YPAT23	026126	\$CM3 =	000024	\$LF	001314
UPAT30	024054	XDAT02	025464	YTMP1	026062	\$CM4 =	000024	\$LFLG	041067
UPAT31	024056	XDAT03	025466	YTMP2	026064	\$CNTLG	041511	\$LOOP	036500
UPAT32	024060	XDONE	025520	YTMP3	026066	\$CNTLU	041504	\$LPADR	001106
UPAT33	024062	XERR1	025344	Y1	025552	\$CPUOP	001344	\$LPERR	001110

SMADR1	001350	\$PASS	001324	\$REG3	001170	\$TKS	001144	\$STRAP	041540
SMADR2	001354	\$PASS2	036514	\$REG4	001172	\$TMP0	001232	\$STRAP2	041562
SMADR3	001360	\$PASTM	003610	\$REG5	001174	\$TMP1	001234	\$STRP =	000014
SMADR4	001364	\$PWAD	041764	\$REG6	001176	\$TMP10	001252	\$STRPAD	041574
\$MAIL	001316	\$PWADN	041624	\$REG7	001200	\$TMP11	001254	\$TSTM	003606
\$MAMS1	001346	\$PWARMG	041760	\$RESRE	037542	\$TMP12	001256	\$TSTNM	001102
\$MAMS2	001352	\$PWURUP	041676	\$RTNAD	036502	\$TMP13	001260	\$TYPDS	040400
\$MAMS3	001356	\$QUES	001312	\$RTRN	036476	\$TMP14	001262	\$TYPE	037600
\$MAMS4	001362	\$RDCHR	041354	\$SAVRE	037504	\$TMP15	001264	\$TYPEC	040012
\$MBADR	003604	\$RDSZ =	000001	\$SAVR6	042006	\$TMP16	001266	\$TYPEX	040142
\$MFLG	041066	\$REGAD	001160	\$SCOPE	036516	\$TMP17	001270	\$TYPOC	040170
\$MNEW	041527	\$REG0	001162	\$SETUP=	000137	\$TMP2	001236	\$TYPON	040204
\$MSGAD	001332	\$REG1	001164	\$STUP =	177777	\$TMP20	001272	\$TYPOS	040144
\$MSGLG	001334	\$REG10	001202	\$SVLAD	037070	\$TMP21	001274	\$UNIT	001330
\$MSGTY	001316	\$REG11	001204	\$SVPC =	003602	\$TMP22	001276	\$UNITM	003612
\$MSWR	041516	\$REG12	001206	\$SWR =	177400	\$TMP23	001300	\$USWR	001342
\$MTYP1	001347	\$REG13	001210	\$SWREG	001340	\$TMP3	001240	\$VECT1	001366
\$MTYP2	001353	\$REG14	001212	\$SWRMK=	000000	\$TMP4	001242	\$VECT2	001370
\$MTYP3	001357	\$REG15	001214	\$SWRMS=	000200	\$TMP5	001244	\$XOFF =	000023
\$MTYP4	001363	\$REG16	001216	\$TAB	042740	\$TMP6	001246	\$XON =	000021
\$MXCNT	037140	\$REG17	001220	\$TBIT	036504	\$TMP7	001250	\$XTSTR	036642
\$NULL	001154	\$REG2	001166	\$TERM =	000030	\$TN =	000037	\$SET4=	000001
\$NWTST=	000001	\$REG20	001222	\$TESTN	001322	\$TPB	001152	\$OFILL	040375
\$OCNT	040374	\$REG21	001224	\$THE	043134	\$TPFLG	001157	\$.RSET	042622
\$OMODE	040376	\$REG22	001226	\$TIMES	001302	\$TPS	001150	\$.X =	003602
\$OVER	037124	\$REG23	001230	\$TKB	001146				

. ABS. 074420 000
000000 001
EPRORS DETECTED: 0

VIRTUAL MEMORY USED: 59424 WORDS (233 PAGES)
DYNAMIC MEMORY: 20034 WORDS (77 PAGES)
ELAPSED TIME: 02:44:50
CKFPAD.BIN,CKFPAD/CR/-SP/NL:TOC=CKFPAD.MLB/ML,CKFPAD.P11

SYMBOL	CROSS REFERENCE	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES
SYMBOL	VALUE									
AADATO	027404	49-5262	49-5269	49-5289	49-5296	49-5301	49-5327	49-5334	49-5372	49-5378
AADONE	027504	49-5386	49-5392	49-5397	49-5403	#49-5407				
AAERRO	027036	49-5342	49-5362	49-5374	49-5380	49-5388	49-5394	49-5400	49-5406	#49-5415
AAERR1	027124	49-5254	#49-5346							
AAERR2	027160	49-5273	#49-5369							
AAERR3	027214	49-5275	#49-5375							
AAERR4	027224	49-5305	#49-5381							
AAERR5	027260	49-5307	#49-5383							
AAERR6	027314	49-5309	#49-5389							
AAERR7	027350	49-5338	#49-5395							
AAER10	027132	49-5340	#49-5401							
AAPATO	027414	#49-5370	49-5382							
AAPAT1	027424	49-5255	49-5283	49-5317	49-5370	49-5376	49-5384	49-5390	49-5396	49-5402
AAPAT2	027434	#49-5408								
AAPAT3	027444	49-5259	49-5286	49-5371	49-5377	49-5385	49-5391	#49-5409		
AAPAT4	027454	49-5264	49-5295	49-5333	49-5369	49-5375	#49-5410			
AAPAT5	027464	49-5268	49-5291	49-5381	49-5383	49-5389	#49-5411			
AAPAT6	027474	49-5300	#49-5412							
AA1	026442	49-5321	49-5395	49-5401	#49-5413					
AA10	026562	49-5329	49-5398	49-5404	#49-5414					
AA11	026616	#49-5251								
AA12	026620	#49-5280								
AA13	026636	49-5285	#49-5287							
AA14	026656	#49-5289								
AA15	026676	#49-5293	49-5310							
AA16	026704	#49-5298	49-5308							
AA17	026710	#49-5303	49-5306							
AA2	026504	49-5304	#49-5306							
AA20	026714	49-5299	#49-5308							
AA21	026716	49-5258	#49-5260							
AA22	026754	49-5294	#49-5310							
AA23	026756	#49-5314								
AA24	026776	49-5320	#49-5322							
AA25	027016	#49-5324								
AA26	027024	#49-5331	49-5341							
AA27	027030	#49-5336	49-5339							
AA3	026506	49-5337	#49-5339							
AA4	026524	49-5332	#49-5341							
AA5	026544	#49-5262								
AA6	026552	#49-5266	49-5276							
AA7	026560	#49-5271	49-5274							
ABASE	= 000000	49-5272	#49-5274							
ACDW1	= 000000	49-5267	#49-5276							
ACDW2	= 000000	20-1795	20-1795							
ACPUOP	= 000000	20-1795	20-1795							
ACO	= %000000	20-1795	20-1795							
		#19-1785	*30-2352	30-2391	*31-2720	*31-2726	31-2731	*31-2769	*31-2776	31-2789
		*32-2894	32-2903	*32-2937	32-2946	*33-3182	33-3182	*33-3182	33-3182	*33-3182
		33-3182	*33-3183	33-3183	*33-3183	33-3183	*33-3183	33-3183	*33-3184	33-3184
		*33-3184	33-3184	*33-3185	33-3185	*33-3185	33-3185	*33-3186	33-3186	*33-3186
		33-3186	*33-3187	33-3187	*33-3187	33-3187	*33-3188	33-3188	*33-3188	33-3188

SYMBOL CROSS REFERENCE
 SYMBOL VALUE

REFERENCES

SYMBOL	CROSS REFERENCE VALUE	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES
		*33-3189	33-3189	*33-3189	33-3189	*33-3190	33-3190	*33-3190
		33-3191	*33-3191	33-3191	*33-3192	33-3192	*33-3192	33-3192
		*33-3193	33-3193	*34-3371	34-3371	*34-3372	34-3372	*34-3373
		34-3374	*34-3375	34-3375	*34-3381	34-3381	*34-3382	34-3382
		*34-3384	34-3384	*34-3385	34-3385	*34-3396	34-3396	*34-3397
		34-3398	*34-3399	34-3399	*34-3400	34-3400	*35-3467	*35-3477
		*35-3507	35-3513	*36-3628	*36-3634	36-3639	*37-3727	*37-3733
		*38-3837	38-3844	*39-3923	*39-3928	39-3942	*40-4026	*40-4033
		*41-4174	41-4179	*42-4307	*42-4313	42-4317	*43-4428	*43-4436
		*44-4590	*44-4606	*44-4613	*44-4625	*44-4631	*44-4643	*44-4648
		*45-4787	*45-4790	45-4794	*45-4815	*45-4820	45-4824	*45-4843
		*45-4871	*45-4876	45-4880	*46-4930	*46-4933	46-4937	*46-4953
		*46-4976	*46-4979	46-4983	*46-4999	*46-5002	46-5006	*47-5083
		*48-5178	*48-5180	48-5184	*49-5257	*49-5260	49-5263	*49-5284
		*49-5318	*49-5322	49-5328	*50-5433	*50-5435	50-5438	*50-5461
		*50-5487	*50-5491	50-5495	*50-5516	*50-5520	50-5524	*50-5547
		*50-5575	*50-5577	50-5580	*51-5693	*51-5695	51-5698	*51-5715
		*51-5741	*51-5746	51-5750	*51-5764	*51-5768	51-5772	*51-5795
		*51-5816	*51-5818	51-5821	*52-5903	*52-5905	52-5908	*52-5932
		*52-5954	*52-5956	52-5959	*52-5976	*52-5978	52-5981	*52-6004
		*52-6032	*52-6034	52-6037	*52-6061	*52-6063	52-6066	*53-6127
		*53-6156	*53-6158	53-6161	*54-6211	*54-6213	54-6216	*54-6233
AC1	=%000001	#19-1786	*31-2717	31-2726	*31-2766	31-2776	*32-2897	*32-2903
		*32-2946	32-2960	*33-3184	33-3184	*33-3185	33-3185	*34-3371
AC2	=%000002	#19-1787	*33-3186	33-3186	*33-3187	33-3187	*34-3372	*34-3382
AC3	=%000003	#19-1788	*33-3188	33-3188	*33-3189	33-3189	*34-3373	*34-3383
AC4	=%000004	#19-1789	*33-3190	33-3190	*33-3191	33-3191	*34-3374	*34-3384
AC5	=%000005	#19-1790	*33-3192	33-3192	*33-3193	33-3193	*34-3375	*34-3385
AC6	=%000006	#19-1791	35-3507					34-3400
AC7	=%000007	#19-1792	35-3477					
ADDW0	= 000000	20-1795	20-1795					
ADDW1	= 000000	20-1795	20-1795					
ADDW10	= 000000	20-1795	20-1795					
ADDW11	= 000000	20-1795	20-1795					
ADDW12	= 000000	20-1795	20-1795					
ADDW13	= 000000	20-1795	20-1795					
ADDW14	= 000000	20-1795	20-1795					
ADDW15	= 000000	20-1795	20-1795					
ADDW2	= 000000	20-1795	20-1795					
ADDW3	= 000000	20-1795	20-1795					
ADDW4	= 000000	20-1795	20-1795					
ADDW5	= 000C00	20-1795	20-1795					
ADDW6	= 000000	20-1795	20-1795					
ADDW7	= 000000	20-1795	20-1795					
ADDW8	= 000000	20-1795	20-1795					
ADDW9	= 000000	20-1795	20-1795					
ADEVCT	= 000000	20-1795	20-1795					
ADEVN	= 000000	20-1795	20-1795					
ADONE	= 005046	25-1886	25-1890	25-1898	25-1906	25-1919	25-1932	#25-1938
AENV	= 000000	20-1795	20-1795					
AENVN	= 000000	20-1795	20-1795					
AERFLG	= 004736	*25-1836	*25-1867	25-1873	25-1885	#25-1900	75-7167	

SYMBOL	CROSS REFERENCE	VALUE	REFERENCES							
BBER5		032420	51-5782	#51-5862						
BBER6		032436	51-5784	#51-5865						
BBER7		032472	51-5805	#51-5871						
BBER8		032510	51-5826	#51-5874						
BBPAT0		032536	51-5740	51-5751	51-5856	51-5857	#51-5878			
BBPAT1		032546	51-5694	51-5716	51-5745	51-5767	51-5796	51-5817	51-5843	51-5852
			51-5867	51-5872	#51-5879					51-5858
BBPAT2		032556	51-5692	51-5699	51-5841	51-5842	#51-5880			
BBPAT3		032566	51-5763	51-5778	51-5862	51-5865	#51-5881			
BBPAT4		032576	51-5714	51-5726	51-5847	51-5850	#51-5882			
BBPAT5		032606	51-5794	51-5871	#51-5883					
BBPAT6		032616	51-5815	51-5822	51-5874	51-5875	#51-5884			
BBP10		032636	51-5721	51-5848	51-5851	#51-5886				
BBP11		032646	51-5801	#51-5887						
BBP7		032626	51-5773	51-5863	51-5866	#51-5885				
BB1		031326	#51-5686							
BB10		031512	#51-5723	51-5733						
BB11		031532	#51-5728	51-5731						
BB12		031542	51-5729	#51-5731						
BB13		031550	51-5724	#51-5733						
BB14		031562	51-5735	#51-5738						
BB15		031616	51-5739	#51-5746						
BB16		031642	#51-5753	51-5756						
BB17		031652	51-5754	#51-5756						
BB2		031370	51-5690	#51-5695						
BB20		031664	51-5758	#51-5761						
BB21		031720	51-5762	#51-5768						
BB22		031744	#51-5775	51-5785						
BB23		031764	#51-5780	51-5783						
BB24		031774	51-5781	#51-5783						
BB25		032002	51-5776	#51-5785						
BB26		032014	51-5787	#51-5790						
BB27		032050	51-5791	#51-5797						
BB3		031374	#51-5697							
BB30		032072	#51-5803	51-5806						
BB31		032102	51-5804	#51-5806						
BB32		032114	51-5808	#51-5811						
BB33		032150	51-5812	#51-5818						
BB34		032172	#51-5824	51-5827						
BB35		032202	51-5825	#51-5827						
BB4		031412	#51-5701	51-5704						
BB5		031422	51-5702	#51-5704						
BB6		031434	51-5707	#51-5710						
BB7		031470	51-5713	#51-5717						
BDONE		005154	26-1962	#26-1981						
BERR		005110	26-1959	#26-1964						
BERR1		005140	26-1968	#26-1975						
BITSTS		014250	#33-3205	33-3285	33-3293	33-3309	33-3311			
BIT0	=	000001	#19-1783							
BIT00	=	000001	#19-1783	19-1783	56-6271	56-6271	57-6273	57-6273		
BIT01	=	000002	#19-1783	19-1783						
BIT02	=	000004	#19-1783	19-1783						

SYMBOL	CROSS REFERENCE VALUE	REFERENCES	CREF	V01
CCP10	031272	50-5525	50-5640	50-5643 #50-5669
CCP11	031302	50-5553	50-5649	50-5652 #50-5670
CCP12	031312	#50-5671		
CCP2	031212	50-5434	50-5439	50-5606 50-5607 50-5612
CCP3	031222	50-5501	50-5515	50-5548 50-5558 50-5648
CCP4	031232	50-5576	50-5581	50-5586 50-5654 50-5655 50-5657
CCP5	031242	50-5519	50-5530	50-5639 50-5642 #50-5666
CCP6	031252	50-5490	50-5496	50-5627 50-5628 50-5633
CCP7	031262	50-5467	50-5619	50-5622 #50-5668
CC1	027510	#50-5428		
CC10	027734	#50-5474	50-5477	
CC11	027744	50-5475	#50-5477	
CC12	027752	50-5470	#50-5479	
CC13	027764	50-5481	#50-5484	
CC14	030020	50-5485	#50-5491	
CC15	030044	#50-5498	50-5508	
CC16	030064	#50-5503	50-5506	
CC17	030074	50-5504	#50-5506	
CC18	030102	50-5499	#50-5508	
CC19	030114	50-5510	#50-5513	
CC2	027544	50-5431	#50-5435	
CC20	030150	50-5514	#50-5520	
CC21	030174	#50-5527	50-5537	
CC22	030214	#50-5532	50-5535	
CC23	030224	50-5533	#50-5535	
CC24	030232	50-5528	#50-5537	
CC25	030244	50-5539	#50-5542	
CC26	030300	50-5545	#50-5549	
CC27	030322	#50-5555	50-5565	
CC28	030342	#50-5560	50-5563	
CC29	030352	50-5561	#50-5563	
CC3	027566	#50-5441	50-5451	
CC30	030360	50-5556	#50-5565	
CC31	030372	50-5567	#50-5570	
CC32	030426	50-5573	#50-5577	
CC33	030450	#50-5583	50-5593	
CC34	030470	#50-5588	50-5591	
CC35	030500	50-5589	#50-5591	
CC36	030506	50-5584	#50-5593	
CC37	030520	50-5595	#50-5597	
CC4	027606	#50-5446	50-5449	
CC5	027616	50-5447	#50-5449	
CC6	027624	50-5442	#50-5451	
CC7	027636	50-5453	#50-5456	
CC8	027672	50-5459	#50-5463	
CC9	027714	#50-5469	50-5479	
CDONE	006120	27-2107	#27-2173	
CERR1	005654	27-2007	27-2048	27-2064 27-2105 #27-2110
CERR2	005752	27-2020	27-2035	27-2077 27-2092 #27-2132
CERR3	006052	27-2124	27-2130	27-2148 27-2154 #27-2157
CERR4	006074	27-2113	27-2137	#27-2164
CFCC1	043671	#71-6522		

SYMBOL CROSS REFERENCE		REFERENCES								
SYMBOL	VALUE									
		36-3665	37-3764	39-3987	40-4089	40-4092	41-4229	41-4232	42-4341	43-4473
		#68-6457	70-6488							
CPTWO	042604	25-1859	25-1913	#69-6466	70-6489					
CR	= 000015	#19-1783	59-6277	59-6277						
CRLF	= 000200	#19-1782	#19-1783	24-1812	24-1812	24-1813	24-1814	33-3325	33-3325	56-6271
		56-6271	56-6271	56-6271	59-6277	59-6277	71-6496	71-6496	71-6500	71-6501
		71-6502	71-6503	71-6505	71-6507	71-6509	71-6510	71-6512	71-6514	71-6515
		71-6530	71-6561	71-6562	71-6563	71-6564	71-6565	71-6566	72-6590	72-6593
		72-6596	72-6597	72-6601	72-6604	72-6611	72-6612	72-6617	72-6620	72-6623
		72-6625	72-6627	72-6629	72-6631	72-6637	72-6638	72-6660	72-6660	72-6660
		72-6663	72-6663	72-6663	72-6665	72-6665	72-6665	72-6668	72-6668	72-6669
		72-6672	72-6672	72-6673	72-6676	72-6700	72-6700	72-6700	72-6701	72-6703
		72-6705	72-6705	72-6705	72-6706	72-6708	72-6710	72-6711	72-6714	72-6715
		72-6733	72-6735	72-6739	72-6740	72-6742	72-6744	72-6746	72-6747	72-6749
		72-6751	72-6752	72-6755	72-6755	72-6757	72-6759	72-6820	72-6820	72-6820
		72-6821	72-6821	72-6821	72-6822	72-6822	72-6822	72-6823	72-6823	72-6823
		72-6824	72-6824	72-6825	72-6825	72-6830	72-6831	72-6831	72-6832	72-6833
		72-6833	72-6834	72-6834	72-6835	72-6835	72-6836	72-6836	72-6837	72-6837
		72-6838	72-6840	72-6841	72-6842	72-6843	72-6844	72-6848	73-6853	73-6880
		73-6884	73-6921	73-6944	73-6946	73-6946	73-6951	73-6951	73-6952	73-6968
		73-6984								
C1	005174	27-1992	#27-1994							
C15	005222	27-1999	#27-2001							
C2	005246	27-2009	#27-2010							
C25	005262	27-2013	#27-2014							
C3	005310	27-2022	#27-2023							
C35	005340	27-2028	#27-2029							
C4	005366	27-2037	#27-2038							
C45	005400	27-2040	#27-2042							
C5	005426	27-2050	#27-2051							
C55	005454	27-2056	#27-2058							
C6	005500	27-2066	#27-2067							
C65	005514	27-2069	#27-2071							
C7	005542	27-2079	#27-2080							
C75	005572	27-2084	#27-2086							
C8	005620	27-2094	#27-2095							
C85	005632	27-2097	#27-2099							
DDDATO	034556	52-5907	52-5913	52-5936	52-5958	52-5980	52-5986	52-6008	52-6014	52-6036
		52-6042	52-6065	52-6071	52-6089	52-6090	52-6091	52-6092	52-6093	52-6094
		52-6095	52-6096	52-6097	52-6098	52-6099	52-6100	#52-6101		
DDDONE	034706	52-6084	52-6088	52-6089	52-6090	52-6091	52-6092	52-6093	52-6094	52-6095
		52-6096	52-6097	52-6098	52-6099	52-6100	#52-6112			
DDERO	033770	52-5925	52-5947	52-5969	52-5997	52-6025	52-6054	52-6083	#52-6085	
DDER1	034006	52-5918	#52-6089							
DDER10	034424	52-6049	#52-6098							
DDER11	034462	52-6076	#52-6099							
DDER12	034520	52-6078	#52-6100							
DDER2	034044	52-5920	#52-6090							
DDER3	034102	52-5942	#52-6091							
DDER4	034140	52-5964	#52-6092							
DDER5	034176	52-5991	#52-6093							
DDER6	034234	52-5993	#52-6094							

SYMBOL CROSS REFERENCE		REFERENCES								
SYMBOL	VALUE									
DDER7	034272	52-6019	#52-6095							
DDER8	034330	52-6021	#52-6096							
DDER9	034366	52-6047	#52-6097							
DDISP	= 177570	#19-1783	20-1795	24-1811						
DDONE	006366	28-2223	#28-2265							
DDPO	034566	52-5938	52-5960	52-6091	52-6092	#52-6102				
DDP1	034576	52-5902	52-5904	52-5933	52-5953	52-6089	52-6089	52-6090	52-6090	52-6091
		52-6092	#52-6103							
DDP2	034606	52-5931	52-5955	52-6091	52-6092	#52-6104				
DDP3	034616	52-5975	52-6005	52-6093	52-6094	52-6095	52-6096	#52-6105		
DDP4	034626	52-5914	52-6031	52-6062	52-6097	52-6098	52-6099	52-6100	#52-6106	
DDP5	034636	52-6033	52-6060	52-6097	52-6098	52-6099	52-6100	#52-6107		
DDP6	034646	52-5977	52-6003	52-6093	52-6094	52-6095	52-6096	#52-6108		
DDP7	034656	52-5982	52-6010	52-6043	52-6072	52-6093	52-6094	52-6095	52-6096	#52-6109
DDP8	034666	52-5987	52-6015	52-6038	52-6067	52-6097	52-6098	52-6099	52-6100	#52-6110
DDP9	034676	52-5909	52-6089	52-6090	#52-6111					
DD1	032662	#52-5898								
DD10	033072	#52-5940	52-5943							
DD11	033102	52-5941	#52-5943							
DD12	033120	52-5946	#52-5949							
DD13	033154	52-5952	#52-5956							
DD14	033176	#52-5962	52-5965							
DD15	033206	52-5963	#52-5965							
DD16	033224	52-5968	#52-5971							
DD17	033260	52-5974	#52-5978							
DD18	033302	#52-5984	52-5994							
DD19	033322	#52-5989	52-5992							
DD2	032716	52-5901	#52-5905							
DD20	033332	52-5990	#52-5992							
DD21	033340	52-5985	#52-5994							
DD22	033352	52-5996	#52-5999							
DD23	033406	52-6002	#52-6006							
DD24	033430	#52-6012	52-6022							
DD25	033450	#52-6017	52-6020							
DD26	033460	52-6018	#52-6020							
DD27	033466	52-6013	#52-6022							
DD3	032740	#52-5911	52-5921							
DD30	033500	52-6024	#52-6027							
DD31	033534	52-6030	#52-6034							
DD32	033556	#52-6040	52-6050							
DD33	033576	#52-6045	52-6048							
DD34	033606	52-6046	#52-6048							
DD35	033614	52-6041	#52-6050							
DD36	033632	52-6053	#52-6056							
DD37	033666	52-6059	#52-6063							
DD38	033710	#52-6069	52-6079							
DD39	033730	#52-6074	52-6077							
DD4	032760	#52-5916	52-5919							
DD40	033740	52-6075	#52-6077							
DD41	033746	52-6070	#52-6079							
DD42	033764	52-6082	#52-6084							
DD5	032770	52-5917	#52-5919							

SYMBOL CROSS REFERENCE		REFERENCES									
SYMBOL	VALUE										
DD6	032776	52-5912	#52-5921								
DD7	033014	52-5924	#52-5927								
DD8	033050	52-5930	#52-5934								
DERR1	006246	28-2199	#28-2227								
DERR2	006342	28-2198	#28-2255								
DF1	071236	21-1802	#74-7023								
DF10	= 071266	21-1802	#74-7030								
DF100	= 072063	21-1802	#74-7086								
DF101	= 072110	21-1802	#74-7087								
DF102	= 072132	21-1802	#74-7088								
DF103	= 072110	21-1802	#74-7089								
DF104	= 072063	21-1802	#74-7090								
DF105	= 072110	21-1802	#74-7091								
DF106	= 072132	21-1802	#74-7092								
DF107	= 072110	21-1802	#74-7093								
DF11	= 071266	21-1802	#74-7031								
DF110	= 072063	21-1802	#74-7094								
DF111	= 072141	21-1802	#74-7095	74-7096	74-7097	74-7098	74-7099	74-7100	74-7108	74-7109	
DF112	= 072141	21-1802	#74-7096								
DF113	= 072141	21-1802	#74-7097								
DF114	= 072141	21-1802	#74-7098								
DF115	= 072141	21-1802	#74-7099								
DF116	= 072141	21-1802	#74-7100								
DF117	= 071256	21-1802	#74-7101								
DF12	= 071502	21-1802	#74-7032								
DF120	= 071256	21-1802	#74-7102								
DF121	= 071775	21-1802	#74-7103								
DF122	= 072012	21-1802	#74-7104								
DF123	= 072036	21-1802	#74-7105								
DF124	= 072145	21-1802	#74-7106	74-7107							
DF125	= 072145	21-1802	#74-7107								
DF126	= 072141	21-1802	#74-7108								
DF127	= 072141	21-1802	#74-7109								
DF13	= 071334	21-1802	#74-7033								
DF130	= 071256	21-1802	#74-7110								
DF131	= 072176	21-1802	#74-7111	74-7112							
DF132	= 072176	21-1802	#74-7112								
DF133	= 072216	21-1802	#74-7113	74-7114	74-7115	74-7116	74-7121	74-7122	74-7125	74-7126	
				74-7128	74-7132	74-7133	74-7134	74-7136	74-7140	74-7141	
				74-7148	74-7150	74-7151	74-7152	74-7153	74-7154	74-7156	
				74-7157	74-7158						
				74-7162							
DF134	= 072216	21-1802	#74-7114								
DF135	= 072216	21-1802	#74-7115								
DF136	= 072216	21-1802	#74-7116								
DF137	= 072242	21-1802	#74-7117	74-7118	74-7119	74-7120	74-7123	74-7124	74-7127	74-7129	
DF14	= 071266	21-1802	#74-7034								
DF140	= 072242	21-1802	#74-7118								
DF141	= 072242	21-1802	#74-7119								
DF142	= 072242	21-1802	#74-7120								
DF143	= 072216	21-1802	#74-7121								
DF144	= 072216	21-1802	#74-7122								
DF145	= 072242	21-1802	#74-7123								

SYMBOL	CROSS REFERENCE	VALUE	REFERENCES						
DF146	=	072242	21-1802	#74-7124					
DF147	=	072216	21-1802	#74-7125					
DF15	=	071266	21-1802	#74-7035					
DF150	=	072216	21-1802	#74-7126					
DF151	=	072242	21-1802	#74-7127					
DF152	=	072216	21-1802	#74-7128					
DF153	=	072242	21-1802	#74-7129					
DF154	=	072252	21-1802	#74-7130	74-7131				
DF155	=	072252	21-1802	#74-7131					
DF156	=	072216	21-1802	#74-7132					
DF157	=	072216	21-1802	#74-7133					
DF16	=	071350	21-1802	#74-7036					
DF160	=	072216	21-1802	#74-7134					
DF161	=	072256	21-1802	#74-7135	74-7137	74-7142	74-7143	74-7146	74-7147
DF162	=	072216	21-1802	#74-7136					
DF163	=	072256	21-1802	#74-7137					
DF164	=	071256	21-1802	#74-7138					
DF165	=	071256	21-1802	#74-7139					
DF166	=	072216	21-1802	#74-7140					
DF167	=	072216	21-1802	#74-7141					
DF17	=	071256	21-1802	#74-7037					
DF170	=	072256	21-1802	#74-7142					
DF171	=	072256	21-1802	#74-7143					
DF172	=	072216	21-1802	#74-7144					
DF173	=	072216	21-1802	#74-7145					
DF174	=	072256	21-1802	#74-7146					
DF175	=	072256	21-1802	#74-7147					
DF176	=	072216	21-1802	#74-7148					
DF177	=	072302	21-1802	#74-7149					
DF2	=	071246	21-1802	#74-7024					
DF20	=	071357	21-1802	#74-7038					
DF200	=	072216	21-1802	#74-7150					
DF201	=	072216	21-1802	#74-7151					
DF202	=	072216	21-1802	#74-7152					
DF203	=	072216	21-1802	#74-7153					
DF204	=	072216	21-1802	#74-7154					
DF205	=	071256	21-1802	#74-7155					
DF206	=	072216	21-1802	#74-7156					
DF207	=	072216	21-1802	#74-7157					
DF21	=	071367	21-1802	#74-7039	74-7077	74-7082			
DF210	=	072216	21-1802	#74-7158					
DF211	=	072305	21-1802	#74-7159	74-7160	74-7161			
DF212	=	072305	21-1802	#74-7160					
DF213	=	072305	21-1802	#74-7161					
DF214	=	072216	21-1802	#74-7162					
DF22	=	071373	21-1802	#74-7040					
DF23	=	071405	21-1802	#74-7041					
DF24	=	071415	21-1802	#74-7042					
DF25	=	071424	21-1802	#74-7043					
DF26	=	071450	21-1802	#74-7044	74-7047				
DF27	=	071475	21-1802	#74-7045	74-7048				
DF3	=	071256	21-1802	#74-7025	74-7026	74-7037	74-7101	74-7102	74-7110
								74-7138	74-7139

SYMBOL	CROSS REFERENCE VALUE	REFERENCES	REF	V01
DF30	= 071515	74-7155		
DF31	= 071450	21-1802 #74-7046		
DF32	= 071475	21-1802 #74-7047		
DF33	= 071531	21-1802 #74-7048		
DF34	= 071557	21-1802 #74-7049		
DF35	= 071557	21-1802 #74-7050	74-7051	74-7052 74-7053
DF36	= 071557	21-1802 #74-7051		
DF37	= 071557	21-1802 #74-7052		
DF4	= 071256	21-1802 #74-7053		
DF40	= 071603	21-1802 #74-7026		
DF41	= 071631	21-1802 #74-7054		
DF42	= 071645	21-1802 #74-7055		
DF43	= 071645	21-1802 #74-7056	74-7057	
DF44	= 071673	21-1802 #74-7057		
DF45	= 071716	21-1802 #74-7058		
DF46	= 071743	21-1802 #74-7059		
DF47	= 071775	21-1802 #74-7060		
DF5	= 071266	21-1802 #74-7061	74-7064	74-7067 74-7103
DF50	= 072012	21-1802 #74-7027	74-7028	74-7029 74-7030 74-7031
DF51	= 072036	21-1802 #74-7062	74-7104	
DF52	= 071775	21-1802 #74-7063	74-7105	
DF53	= 072054	21-1802 #74-7064		
DF54	= 072063	21-1802 #74-7065	74-7068	74-7070
		74-7090 #74-7066	74-7069	74-7071 74-7072 74-7073 74-7076 74-7081 74-7086
		74-7094		
DF55	= 071775	21-1802 #74-7067		
DF56	= 072054	21-1802 #74-7068		
DF57	= 072063	21-1802 #74-7069		
DF6	= 071266	21-1802 #74-7028	74-7034	74-7035
DF60	= 072054	21-1802 #74-7070		
DF61	= 072063	21-1802 #74-7071		
DF62	= 072063	21-1802 #74-7072		
DF63	= 072063	21-1802 #74-7073		
DF64	= 072101	21-1802 #74-7074	74-7075	
DF65	= 072101	21-1802 #74-7075		
DF66	= 072063	21-1802 #74-7076		
DF67	= 071367	21-1802 #74-7077		
DF7	= 071266	21-1802 #74-7029		
DF70	= 072110	21-1802 #74-7078	74-7079	74-7083 74-7084 74-7087 74-7089 74-7091 74-7093
DF71	= 072110	21-1802 #74-7079		
DF72	= 072132	21-1802 #74-7080	74-7085	74-7088 74-7092
DF73	= 072063	21-1802 #74-7081		
DF74	= 071367	21-1802 #74-7082		
DF75	= 072110	21-1802 #74-7083		
DF76	= 072110	21-1802 #74-7084		
DF77	= 072132	21-1802 #74-7085		
DH1	= 065631	21-1802 #73-6855		
DH10	= 066201	21-1802 #73-6866		
DH100	= 067503	21-1802 #73-6934		
DH101	= 066201	21-1802 #73-6935		
DH102	= 067073	21-1802 #73-6936		
DH103	= 067031	21-1802 #73-6937		

SYMBOL CROSS REFERENCE		REFERENCES								
SYMBOL	VALUE									
DH162	= 067503	21-1802	#73-6995							
DH163	= 067503	21-1802	#73-6996							
DH164	= 066321	21-1802	#73-6997							
DH165	= 066321	21-1802	#73-6998							
DH166	= 067503	21-1802	#73-6999							
DH167	= 067503	21-1802	#73-7000							
DH17	066321	21-1802	#73-6874	73-6955	73-6961	73-6962	73-6997	73-6998	73-7014	
DH170	= 067503	21-1802	#73-7001							
DH171	= 067503	21-1802	#73-7002							
DH172	= 067503	21-1802	#73-7003							
DH173	= 067503	21-1802	#73-7004							
DH174	= 067503	21-1802	#73-7005							
DH175	= 067503	21-1802	#73-7006							
DH176	= 067503	21-1802	#73-7007							
DH177	071125	21-1802	#73-7008							
DH2	065721	21-1802	#73-6857							
DH20	066411	21-1802	#73-6876							
DH200	= 067503	21-1802	#73-7009							
DH201	= 067503	21-1802	#73-7010							
DH202	= 067503	21-1802	#73-7011							
DH203	= 067503	21-1802	#73-7012							
DH204	= 067503	21-1802	#73-7013							
DH205	= 066321	21-1802	#73-7014							
DH206	= 067503	21-1802	#73-7015							
DH207	= 067503	21-1802	#73-7016							
DH21	= 066201	21-1802	#73-6878							
DH210	= 067503	21-1802	#73-7017							
DH211	071171	21-1802	#73-7018							
DH212	= 066201	21-1802	#73-7019							
DH213	= 066201	21-1802	#73-7020							
DH214	= 067503	21-1802	#73-7021							
DH22	066477	21-1802	#73-6879							
DH23	066534	21-1802	#73-6880							
DH24	066672	21-1802	#73-6883							
DH25	067031	21-1802	#73-6885	73-6927	73-6932	73-6937	73-6941	73-6943	73-6948	
DH26	067073	21-1802	#73-6886	73-6888	73-6890	73-6891	73-6911	73-6914	73-6916	73-6928
			73-6933	73-6936	73-6940					
DH27	= 067073	21-1802	#73-6888							
DH3	066014	21-1802	#73-6859	75-7190	75-7220					
DH30	= 000000	21-1802	#73-6889							
DH31	= 067073	21-1802	#73-6890							
DH32	= 067073	21-1802	#73-6891							
DH33	067161	21-1802	#73-6892	73-6894	73-6895	73-6896	73-6897	73-6898		
DH34	= 067161	21-1802	#73-6894							
DH35	= 067161	21-1802	#73-6895							
DH36	= 067161	21-1802	#73-6896							
DH37	= 067161	21-1802	#73-6897							
DH4	066105	21-1802	#73-6861							
DH40	= 067161	21-1802	#73-6898							
DH41	= 000000	21-1802	#73-6899							
DH42	067264	21-1802	#73-6900	73-6902						
DH43	= 067264	21-1802	#73-6902							

SYMBOL CROSS REFERENCE		REFERENCES									
SYMBOL	VALUE										
DH44	= 000000	21-1802	#73-6903								
DH45	= 067366	21-1802	#73-6904	73-6906							
DH46	= 067405	21-1802	#73-6905								
DH47	= 067366	21-1802	#73-6906								
DH5	= 066201	21-1802	#73-6863	73-6864	73-6865	73-6866	73-6867	73-6870	73-6871	73-6878	
		73-6910	73-6913	73-6925	73-6926	73-6930	73-6931	73-6935	73-6939	73-6958	
		73-6963	73-7019	73-7020							
DH50	= 067431	21-1802	#73-6907	73-6909	73-6959	73-6960					
DH51	= 067431	21-1802	#73-6909								
DH52	= 066201	21-1802	#73-6910								
DH53	= 067073	21-1802	#73-6911								
DH54	= 067503	21-1802	#73-6912	73-6915	73-6917	73-6918	73-6919	73-6924	73-6929	73-6934	
		73-6938	73-6942	73-6964	73-6966	73-6967	73-6983	73-6991	73-6992	73-6993	
		73-6994	73-6995	73-6996	73-6999	73-7000	73-7001	73-7002	73-7003	73-7004	
		73-7005	73-7006	73-7007	73-7009	73-7010	73-7011	73-7012	73-7013	73-7015	
		73-7016	73-7017	73-7021							
DH55	= 066201	21-1802	#73-6913								
DH56	= 067073	21-1802	#73-6914								
DH57	= 067503	21-1802	#73-6915								
DH6	= 066201	21-1802	#73-6864								
DH60	= 067073	21-1802	#73-6916								
DH61	= 067503	21-1802	#73-6917								
DH62	= 067503	21-1802	#73-6918								
DH63	= 067503	21-1802	#73-6919								
DH64	= 067613	21-1802	#73-6922								
DH65	= 067544	21-1802	#73-6920								
DH66	= 067503	21-1802	#73-6924								
DH67	= 066201	21-1802	#73-6925								
DH7	= 066201	21-1802	#73-6865								
DH70	= 066201	21-1802	#73-6926								
DH71	= 067031	21-1802	#73-6927								
DH72	= 067073	21-1802	#73-6928								
DH73	= 067503	21-1802	#73-6929								
DH74	= 066201	21-1802	#73-6930								
DH75	= 066201	21-1802	#73-6931								
DH76	= 067031	21-1802	#73-6932								
DH77	= 067073	21-1802	#73-6933								
DISPLA	= 001142	#20-1795	*24-1811	*24-1811	56-6271	57-6273					
DISPRE	= 000174	#19-1794	24-1811								
DPAT3	= 017500	#37-3806									
DSWR	= 177570	#19-1783	20-1795	24-1811							
DT1	= 072314	21-1802	#75-7165								
DT10	= 072434	21-1802	#75-7175								
DT100	= 073750	21-1802	#75-7267								
DT101	= 074026	21-1802	#75-7268								
DT102	= 074026	21-1802	#75-7269								
DT103	= 074026	21-1802	#75-7270								
DT104	= 073750	21-1802	#75-7271								
DT105	= 074026	21-1802	#75-7272								
DT106	= 074074	21-1802	#75-7273								
DT107	= 074026	21-1802	#75-7274								
DT11	= 072434	21-1802	#75-7176								

SYMBOL	CROSS REFERENCE	VALUE	REFERENCES								
DT164	=	072360	21-1802	#75-7324							
DT165	=	072360	21-1802	#75-7325							
DT166	=	074254	21-1802	#75-7326							
DT167	=	074254	21-1802	#75-7327							
DT17	=	072360	21-1802	#75-7186							
DT170	=	074254	21-1802	#75-7328							
DT171	=	074254	21-1802	#75-7329							
DT172	=	074254	21-1802	#75-7330							
DT173	=	074254	21-1802	#75-7331							
DT174	=	074254	21-1802	#75-7332							
DT175	=	074254	21-1802	#75-7333							
DT176	=	074254	21-1802	#75-7334							
DT177	=	074360	21-1802	#75-7335							
DT2		072336	21-1802	#75-7167							
DT20		072616	21-1802	#75-7187							
DT200	=	074254	21-1802	#75-7336							
DT201	=	074254	21-1802	#75-7337							
DT202	=	074254	21-1802	#75-7338							
DT203	=	074254	21-1802	#75-7339							
DT204	=	074254	21-1802	#75-7340							
DT205	=	072360	21-1802	#75-7341							
DT206	=	074254	21-1802	#75-7342							
DT207	=	074254	21-1802	#75-7343							
DT21		072640	21-1802	#75-7189	75-7257		75-7263				
DT210	=	074254	21-1802	#75-7344							
DT211		074370	21-1802	#75-7345							
DT212		074406	21-1802	#75-7346	75-7347						
DT213	=	074406	21-1802	#75-7347							
DT214	=	074254	21-1802	#75-7348							
DT22		072652	21-1802	#75-7190							
DT23		072700	21-1802	#75-7193							
DT24		072722	21-1802	#75-7195							
DT25		072746	21-1802	#75-7198							
DT26		073016	21-1802	#75-7200	75-7208						
DT27		073072	21-1802	#75-7203	75-7209						
DT3		072360	21-1802	#75-7168	75-7170	75-7186	75-7282	75-7283	75-7293	75-7324	75-7325
			75-7341								
DT30		073134	21-1802	#75-7205							
DT31	=	073016	21-1802	#75-7208							
DT32	=	073072	21-1802	#75-7209							
DT33		073166	21-1802	#75-7210	75-7219						
DT34		073244	21-1802	#75-7213	75-7216	75-7217	75-7218	75-7238	75-7285		
DT35	=	073244	21-1802	#75-7216							
DT36	=	073244	21-1802	#75-7217							
DT37	=	073244	21-1802	#75-7218							
DT4	=	072360	21-1802	#75-7170							
DT40	=	073166	21-1802	#75-7219							
DT41		073316	21-1802	#75-7220							
DT42		073350	21-1802	#75-7223	75-7226						
DT43	=	073350	21-1802	#75-7226							
DT44		073426	21-1802	#75-7227							
DT45		073476	21-1802	#75-7229							

SYMBOL	CROSS REFERENCE	VALUE	REFERENCES	SEQUENCE	CREF	V01						
DT46		073552	21-1802	#75-7232								
DT47		073640	21-1802	#75-7236	75-7241	75-7246	75-7284					
DT5		072402	21-1802	#75-7171								
DT50	=	073244	21-1802	#75-7238								
DT51		073672	21-1802	#75-7239	75-7286							
DT52	=	073640	21-1802	#75-7241								
DT53		073730	21-1802	#75-7242	75-7247	75-7249						
DT54		073750	21-1802	#75-7244	75-7248	75-7250	75-7251	75-7252	75-7256	75-7262	75-7267	
			75-7271	75-7275								
DT55	=	073640	21-1802	#75-7246								
DT56	=	073730	21-1802	#75-7247								
DT57	=	073750	21-1802	#75-7248								
DT6		072434	21-1802	#75-7173	75-7174	75-7175	75-7176	75-7183	75-7184			
DT60	=	073730	21-1802	#75-7249								
DT61	=	073750	21-1802	#75-7250								
DT62	=	073750	21-1802	#75-7251								
DT63	=	073750	21-1802	#75-7252								
DT64		074006	21-1802	#75-7253	75-7255							
DT65	=	074006	21-1802	#75-7255								
DT66	=	073750	21-1802	#75-7256								
DT67	=	072640	21-1802	#75-7257								
DT7	=	072434	21-1802	#75-7174								
DT70		074026	21-1802	#75-7258	75-7260	75-7264	75-7265	75-7268	75-7270	75-7272	75-7274	
DT71	=	074026	21-1802	#75-7260	75-7269							
DT72		074074	21-1802	#75-7261	75-7266	75-7273						
DT73	=	073750	21-1802	#75-7262								
DT74	=	072640	21-1802	#75-7263								
DT75	=	074026	21-1802	#75-7264								
DT76	=	074026	21-1802	#75-7265								
DT77	=	074074	21-1802	#75-7266								
D1		006152	28-2196	#28-2201	28-2219	28-2225						
D10		006354	28-2256	#28-2259								
D2		006176	*28-2204	28-2206	#28-2207							
D3		006200	#28-2208	28-2227								
D4		006204	#28-2210									
D5		006216	#28-2216	28-2247	28-2253	28-2263						
D6		006232	28-2217	#28-2221								
D7		006242	28-2222	#28-2224								
D8		006312	#28-2241	28-2250	28-2255							
D9		006324	28-2246	#28-2249								
EDONE		006532	29-2293	29-2299	29-2305	#29-2318						
EEDATO		035410	53-6131	53-6137	53-6160	53-6166	53-6184	53-6185	53-6186	53-6187	#53-6188	
EEDONE		035472	53-6179	53-6183	53-6184	53-6185	53-6186	53-6187	#53-6194			
EEERO		035202	53-6149	53-6178	#53-6180							
EEER1		035220	53-6142	#53-6184								
EEER2		035256	53-6144	#53-6185								
EEER3		035314	53-6171	#53-6186								
EEER4		035352	53-6173	#53-6187								
EEO		035420	53-6133	53-6162	53-6184	53-6185	53-6186	53-6187	#53-6189			
EEO1		035432	53-6126	53-6128	53-6184	53-6184	53-6185	53-6185	#53-6190			
EEO2		035442	53-6138	#53-6191								
EEO3		035452	53-6155	53-6157	53-6186	53-6186	53-6187	53-6187	#53-6192			

SYMBOL	CROSS REFERENCE	VALUE	REFERENCES		
EEP4		035462	53-6167	#53-6193	
EERRO		006450	29-2288	#29-2295	
EERR1		006466	29-2292	#29-2301	
EERR2		006502	29-2277	#29-2307	
EE1		034712	#53-6122		
EE10		035142	#53-6169	53-6172	
EE11		035152	53-6170	#53-6172	
EE12		035160	53-6165	#53-6174	
EE13		035176	53-6177	#53-6179	
EE2		034746	53-6125	#53-6129	
EE3		034770	#53-6135	53-6145	
EE4		035010	#53-6140	53-6143	
EE5		035020	53-6141	#53-6143	
EE6		035026	53-6136	#53-6145	
EE7		035044	53-6148	#53-6151	
EE8		035100	53-6154	#53-6158	
EE9		035122	#53-6164	53-6174	
EMTVEC	=	000030	#19-1783	*24-1811	
EM1		045352	21-1802	#72-6577	
EM10	=	045560	21-1802	#72-6584	
EM100		054462	21-1802	#72-6709	
EM101		054522	21-1802	#72-6710	
EM102		054646	21-1802	#72-6711	
EM103		054720	21-1802	#72-6712	
EM104		055023	21-1802	#72-6713	
EM105		055064	21-1802	#72-6714	
EM106		055211	21-1802	#72-6715	
EM107		055264	21-1802	#72-6716	
EM11		045701	21-1802	#72-6585	
EM110		055370	21-1802	#72-6717	
EM111		055432	21-1802	#72-6733	72-6734 72-6737
EM112	=	055432	21-1802	#72-6734	
EM113	=	055534	21-1802	#72-6735	72-6736 72-6738
EM114	=	055534	21-1802	#72-6736	
EM115	=	055432	21-1802	#72-6737	
EM116	=	055534	21-1802	#72-6738	
EM117		055636	21-1802	#72-6739	
EM12	=	000000	21-1802	#72-6586	
EM120		055772	21-1802	#72-6740	
EM121		056126	21-1802	#72-6741	
EM122		056245	21-1802	#72-6743	
EM123		056344	21-1802	#72-6745	
EM124		056405	21-1802	#72-6746	
EM125		056500	21-1802	#72-6747	
EM126		056570	21-1802	#72-6748	
EM127		056777	21-1802	#72-6753	
EM13	=	000000	21-1802	#72-6587	
EM130		057212	21-1802	#72-6758	
EM131		057312	21-1802	#72-6761	
EM132		057352	21-1802	#72-6762	
EM133		057412	21-1802	#72-6803	72-6807
EM134		057451	21-1802	#72-6804	72-6808

SYMBOL	CROSS REFERENCE	VALUE	REFERENCES		
EM135		057510	21-1802	#72-6805	72-6809
EM136		057547	21-1802	#72-6806	72-6810
EM137	=	057412	21-1802	#72-6807	
EM14		045762	21-1802	#72-6588	
EM140	=	057451	21-1802	#72-6808	
EM141	=	057510	21-1802	#72-6809	
EM142	=	057547	21-1802	#72-6810	
EM143		057606	21-1802	#72-6811	72-6813
EM144		057641	21-1802	#72-6812	72-6814
EM145	=	057606	21-1802	#72-6813	
EM146	=	057641	21-1802	#72-6814	
EM147		057674	21-1802	#72-6815	72-6816
EM15		046105	21-1802	#72-6591	72-6817
EM150	=	057674	21-1802	#72-6816	
EM151	=	057674	21-1802	#72-6817	
EM152		057726	21-1802	#72-6818	72-6819
EM153	=	057726	21-1802	#72-6819	
EM154		057760	21-1802	#72-6820	
EM155		060212	21-1802	#72-6821	
EM156		060445	21-1802	#72-6822	
EM157		060662	21-1802	#72-6823	
EM16		046230	21-1802	#72-6594	
EM160		061101	21-1802	#72-6824	
EM161		061306	21-1802	#72-6825	
EM162		061513	21-1802	#72-6826	
EM163		061560	21-1802	#72-6827	
EM164		061625	21-1802	#72-6828	
EM165		061672	21-1802	#72-6829	
EM166		061737	21-1802	#72-6830	
EM167		062047	21-1802	#72-6831	
EM17		046301	21-1802	#72-6595	
EM170		062306	21-1802	#72-6832	
EM171		062416	21-1802	#72-6833	
EM172		062655	21-1802	#72-6834	
EM173		063114	21-1802	#72-6835	
EM174		063353	21-1802	#72-6836	
EM175		063612	21-1802	#72-6837	
EM176		064051	21-1802	#72-6838	
EM177		064206	21-1802	#72-6839	
EM2		045407	21-1802	#72-6578	
EM20		046534	21-1802	#72-6599	
EM200		064242	21-1802	#72-6840	
EM201		064377	21-1802	#72-6841	
EM202		064534	21-1802	#72-6842	
EM203		064671	21-1802	#72-6843	
EM204		065026	21-1802	#72-6844	
EM205		065163	21-1802	#72-6845	
EM206		065230	21-1802	#72-6846	
EM207		065275	21-1802	#72-6847	
EM21		046712	21-1802	#72-6603	
EM210		065417	21-1802	#72-6849	
EM211	=	045560	21-1802	#72-6850	

SYMBOL	CROSS REFERENCE	VALUE	REFERENCES		
EM212	=	045614	21-1802	#72-6851	
EM213	=	045646	21-1802	#72-6852	
EM214		065474	21-1802	#72-6853	
EM22		047043	21-1802	#72-6606	72-6608 72-6609
EM23	=	047043	21-1802	#72-6608	
EM24	=	047043	21-1802	#72-6609	
EM25		047130	21-1802	#72-6610	
EM26		047243	21-1802	#72-6612	72-6614
EM27	=	047243	21-1802	#72-6614	
EM3		045453	21-1802	#72-6579	
EM30		047311	21-1802	#72-6615	
EM31		047363	21-1802	#72-6617	72-6619
EM32	=	047363	21-1802	#72-6619	
EM33		047431	21-1802	#72-6620	
EM34		047472	21-1802	#72-6622	
EM35		047574	21-1802	#72-6624	
EM36		047676	21-1802	#72-6626	
EM37		047777	21-1802	#72-6628	
EM4		045520	21-1802	#72-6580	
EM40		050100	21-1802	#72-6630	
EM41		050251	21-1802	#72-6632	
EM42		050306	21-1802	#72-6637	
EM43		050427	21-1802	#72-6638	
EM44		050550	21-1802	#72-6639	72-6640
EM45	=	050550	21-1802	#72-6640	
EM46		050613	21-1802	#72-6641	
EM47		050671	21-1802	#72-6646	
EM5		045560	21-1802	#72-6581	72-6584 72-6850
EM50		051007	21-1802	#72-6647	
EM51		051105	21-1802	#72-6649	
EM52		051146	21-1802	#72-6659	
EM53		051267	21-1802	#72-6660	
EM54		051464	21-1802	#72-6661	
EM55		051530	21-1802	#72-6662	
EM56		051651	21-1802	#72-6663	
EM57		052046	21-1802	#72-6664	
EM6		045614	21-1802	#72-6582	72-6851
EM60		052112	21-1802	#72-6665	
EM61		052307	21-1802	#72-6666	
EM62		052353	21-1802	#72-6667	
EM63		052545	21-1802	#72-6671	
EM64		052737	21-1802	#72-6675	72-6677
EM65	=	052737	21-1802	#72-6677	
EM66		053073	21-1802	#72-6678	
EM67		053136	21-1802	#72-6700	
EM7		045646	21-1802	#72-6583	72-6852
EM70		053367	21-1802	#72-6701	
EM71		053512	21-1802	#72-6702	
EM72		053614	21-1802	#72-6703	
EM73		053670	21-1802	#72-6704	
EM74		053730	21-1802	#72-6705	
EM75		054161	21-1802	#72-6706	

SYMBOL	CROSS REFERENCE VALUE	REFERENCES	30-2549	30-2564	30-2579	30-2587	30-2593	30-2612	30-2622	30-2630	30-2649
FECMS	043123	30-2549	30-2564	30-2579	30-2587	30-2593	30-2612	30-2622	30-2630	30-2649	
FERRO	007144	30-2659	30-2667	#30-2697							
FERR1	007202	#71-6503	75-7172								
FERR10	007762	30-2369	#30-2445								
FERR11	010016	30-2358	#30-2453								
FERR2	007300	30-2424	30-2436	30-2539	30-2570	#30-2581					
FERR20	010036	30-2375	30-2442	#30-2589							
FERR21	010154	30-2362	#30-2476								
FERR25	010204	30-2348	#30-2595								
FERR26	010322	30-2603	#30-2624								
FERR3	007340	30-2387	#30-2632								
FERR4	007332	30-2640	#30-2661								
FERR5	007436	30-2397	#30-2489								
FERR6	007472	30-2401	#30-2486								
FERR7	007626	30-2408	30-2412	30-2416	30-2420	#30-2512					
FER2	007304	30-2428	#30-2520								
FFDATO	036000	30-2432	#30-2551								
FFDONE	036060	#30-2478	30-2487								
FFERO	035670	54-6215	54-6237	54-6255	54-6257	#54-6260					
FFER1	035704	54-6248	54-6253	54-6255	54-6257	#54-6266					
FFER2	035742	54-6225	54-6247	#54-6250							
FFP0	036010	54-6243	#54-6255								
FFP1	036020	54-6221	#54-6257								
FFP2	036030	54-6232	54-6255	#54-6261							
FFP3	036040	54-6234	54-6255	#54-6262							
FFP4	036050	54-6210	54-6257	#54-6263							
FF1	035476	54-6212	54-6257	#54-6264							
FF10	035656	54-6217	54-6239	54-6255	54-6257	#54-6265					
FF11	035666	#54-6206									
FF2	035532	54-6242	#54-6244								
FF3	035554	54-6246	#54-6248								
FF4	035562	54-6209	#54-6213								
FF5	035572	#54-6219	54-6222								
FF6	035626	54-6220	#54-6222								
FF7	035650	54-6224	#54-6228								
FPSMS	043057	54-6231	#54-6235								
FPSPUR	042534	#54-6241	54-6244								
FPVECT	= 000244	#71-6502	75-7172								
FXDAT0	010416	25-1855	28-2229	29-2309	35-3527	35-3546	44-4687	44-4729	49-5350	#67-6444	
FXDAT1	010420	70-6487									
FXDAT2	010422	#19-1778	*25-1839	*25-1855	*28-2199	*29-2277	*35-3469	*35-3502	*44-4579	*44-4670	
FXDAT3	010424	*49-5254	*51-5689	*70-6487							
FXDAT4	010426	30-2341	30-2365	30-2404	30-2446	30-2447	30-2516	#30-2687			
FXDAT5	010430	#30-2688									
FXDAT6	010432	#30-2689									
FXDAT7	010434	#30-2690									
F1	006536	30-2585	#30-2691								
F10	006760	#30-2692									
		#30-2693									
		#30-2694									
		#30-2335	30-2335								
		30-2378	#30-2391	30-2639							

SYMBOL	CROSS REFERENCE	REFERENCES	SEQUENCE	CREF	V01					
SYMBOL	VALUE									
GRESET	014346	33-3182	33-3183	33-3184	33-3185	33-3186	33-3187	33-3188	33-3189	33-3190
		33-3191	33-3192	33-3193	#33-3227					
GSETUP	014270	33-3182	33-3183	33-3184	33-3185	33-3186	33-3187	33-3188	33-3189	33-3190
		33-3191	33-3192	33-3193	#33-3207					
GSUM	014500	33-3182	33-3183	33-3184	33-3185	33-3186	33-3187	33-3188	33-3189	33-3190
		33-3191	33-3192	33-3193	#33-3262					
GS1	014320	#33-3217	33-3231							
GTSWR	= 104406	24-1812	#64-6287							
G1	012176	33-3182	33-3182	#33-3182	33-3182					
G10	012504	33-3184	#33-3184							
G11	012506	33-3184	#33-3184							
G12	012600	33-3185	33-3185	#33-3185	33-3185					
G13	012632	33-3185	#33-3185							
G14	012634	33-3185	#33-3185							
G15	012726	33-3186	33-3186	#33-3186	33-3186					
G16	012760	33-3186	#33-3186							
G17	012762	33-3186	#33-3186							
G2	012230	33-3182	#33-3182							
G20	013054	33-3187	33-3187	#33-3187	33-3187					
G21	013106	33-3187	#33-3187							
G22	013110	33-3187	#33-3187							
G23	013202	33-3188	33-3188	#33-3188	33-3188					
G24	013234	33-3188	#33-3188							
G25	013236	33-3188	#33-3188							
G26	013330	33-3189	33-3189	#33-3189	33-3189					
G27	013362	33-3189	#33-3189							
G3	012232	33-3182	#33-3182							
G30	013364	33-3189	#33-3189							
G31	013456	33-3190	33-3190	#33-3190	33-3190					
G32	013510	33-3190	#33-3190							
G33	013512	33-3190	#33-3190							
G34	013604	33-3191	33-3191	#33-3191	33-3191					
G35	013636	33-3191	#33-3191							
G36	013640	33-3191	#33-3191							
G37	013732	33-3192	33-3192	#33-3192	33-3192					
G4	012324	33-3183	33-3183	#33-3183	33-3183					
G40	013764	33-3192	#33-3192							
G41	013766	33-3192	#33-3192							
G42	014060	33-3193	33-3193	#33-3193	33-3193					
G43	014112	33-3193	#33-3193							
G44	014114	33-3193	#33-3193							
G5	012356	33-3183	#33-3183							
G6	012360	33-3183	#33-3183							
G7	012452	33-3184	33-3184	#33-3184	33-3184					
HADR	015744	*34-3404	34-3412	#34-3433						
HA1R	016020	34-3396	34-3406	34-3415	#34-3442					
HA1W	015750	34-3362	34-3371	34-3381	34-3405	34-3423	#34-3436			
HA2R	016030	34-3397	#34-3443							
HA2W	015760	34-3372	34-3382	#34-3437						
HA3R	016040	34-3398	#34-3444							
HA3W	015770	34-3373	34-3383	#34-3438						
HA4R	016050	34-3399	#34-3445							

SYMBOL	CROSS REFERENCE	VALUE	REFERENCES						
HA4W		016000	34-3374	34-3384	#34-3439				
HA5R		016060	34-3400	#34-3446					
HA5W		016010	34-3375	34-3385	#34-3440				
HCLR		015674	34-3368	34-3395	#34-3415				
HCLR1		015704	#34-3417	34-3418					
HCMP		015636	34-3379	34-3381	34-3382	34-3383	34-3384	34-3385	#34-3404
HCMP1		015656	#34-3408	34-3411					
HCMP2		015666	34-3409	#34-3411					
HDAT1		016070	34-3363	#34-3448					
HDAT2		016100	#34-3449						
HDAT3		016110	#34-3450						
HDAT4		016120	#34-3451						
HDAT5		016130	#34-3452						
HDONE		016140	34-3389	34-3430	#34-3453				
HERROR		015712	34-3410	#34-3422					
HFLAG		015746	*34-3361	34-3387	*34-3391	#34-3434			
HSTD		015560	34-3377	34-3381	34-3382	34-3383	34-3384	34-3385	#34-3395
HT	=	000011	#19-1783	59-6277	59-6277				
H1		015222	34-3359	#34-3361					
H10		015464	#34-3384	34-3384					
H11		015516	#34-3385	34-3385					
H12		015550	34-3388	#34-3391					
H2		015242	#34-3365	34-3366					
H3		015252	#34-3370	34-3392					
H4		015324	#34-3377						
H5		015346	#34-3381	34-3381					
H6		015400	#34-3382	34-3382					
H7		015432	#34-3383	34-3383					
IBSAVE		037144	#57-6273	*57-6273	*57-6273	57-6273	57-6273	57-6273	57-6273
IDAT10		011300	31-2710	31-2716	31-2734	31-2760	31-2765	31-2794	31-2830
IDAT11		011302	#31-2872						#31-2871
IDAT12		011304	*31-2791	#31-2873					
IDAT13		011306	*31-2792	#31-2874					
IDAT00		011270	31-2730	31-2733	31-2788	31-2793	31-2831	#31-2866	
IDAT01		011272	#31-2867						
IDAT02		011274	31-2739	31-2744	31-2799	#31-2868			
IDAT03		011276	31-2802	#31-2869					
IDONE		011310	31-2808	31-2826	31-2833	31-2841	31-2853	#31-2876	
IERR0		011050	31-2723	#31-2811					
IERR1		011132	31-2748	31-2801	#31-2829				
IERR2		011152	31-2752	#31-2836					
IERR25		011174	#31-2839	31-2845					
IERR3		011224	31-2783	#31-2848					
IERR4		011200	31-2804	#31-2842					
ILLMS		044033	#71-6530	75-7195					
ILL1		043727	#71-6527						
ILL2		043772	#71-6529						
IOTVEC	=	000020	#19-1783	*24-1811	*24-1811				
IPAT10		011250	31-2711	31-2759	#31-2856				
IPAT11		011252	#31-2857						
IPAT12		011254	31-2799	#31-2858					
IPAT13		011256	31-2802	#31-2859					

SYMBOL	CROSS REFERENCE	VALUE	REFERENCES						
IPAT20		011260	31-2719	31-2768	#31-2861				
IPAT21		011262	#31-2862						
IPAT22		011264	#31-2863						
IPAT23		011266	#31-2864						
I1		010450	31-2706	#31-2708					
I10		010632	31-2742	#31-2751					
I105		010640	31-2737	#31-2754					
I106		010634	31-2750	#31-2752					
I11		010642	#31-2758						
I12		010650	31-2758	#31-2759					
I13		010664	#31-2762	31-2763					
I14		010726	31-2772	#31-2776	31-2849				
I15		010730	#31-2777	31-2815					
I16		010732	#31-2778	31-2817					
I17		010750	31-2782	#31-2785					
I2		010466	#31-2713	31-2714					
I20		011010	#31-2796	31-2806					
I21		011024	#31-2801	31-2803					
I22		011030	31-2800	#31-2802					
I23		011044	31-2797	#31-2806					
I3		010534	31-2724	#31-2726					
I4		010536	#31-2727	31-2811					
I5		010540	#31-2728	31-2813					
I6		010564	#31-2736	31-2754					
I7		010600	#31-2741	31-2751					
JBUF0		017144	36-3641	#36-3690					
JBUF1		017146	#36-3691						
JBUF2		017150	#36-3692						
JBUF3		017152	#36-3693						
JDAT10		017154	36-3630	36-3645	36-3653	36-3677	36-3685	#36-3695	
JDAT11		017156	#36-3696						
JDAT12		017160	#36-3697						
JDAT13		017162	#36-3698						
JDAT00		017164	36-3638	36-3646	36-3686	#36-3700			
JDAT0		017174	36-3627	#36-3705					
JDAT01		017166	#36-3701						
JDAT02		017170	#36-3702						
JDAT03		017172	#36-3703						
JDAT1		017176	#36-3706						
JDAT2		017200	#36-3707						
JDAT3		017202	#36-3708						
JDONE		017204	36-3657	36-3672	36-3679	36-3688	#36-3711		
JERRO		017024	36-3632	#36-3661					
JERR1		017072	36-3643	36-3655	#36-3674				
JERR2		017116	36-3650	#36-3683					
J1		016716	36-3622	#36-3624					
J10		017044	36-3662	36-3664	#36-3667				
J2		016742	#36-3634	36-3675	36-3684				
J3		016744	#36-3635	36-3661					
J4		016746	#36-3636	36-3663					
J5		017002	#36-3648	36-3651					
J6		017010	36-3649	#36-3651					

SYMBOL	CROSS REFERENCE	VALUE	REFERENCES				
J7		017022	36-3654	#36-3657			
KBUF0		017452	37-3729	37-3740	37-3752	#37-3793	
KBUF1		017454	#37-3794				
KBUF2		017456	#37-3795				
KBUF3		017460	#37-3796				
KDAT10		017442	37-3744	37-3775	37-3783	#37-3788	
KDAT11		017444	#37-3789				
KDAT12		017446	#37-3790				
KDAT13		017450	#37-3791				
KDAT00		017462	37-3737	37-3745	37-3784	#37-3798	
KDAT01		017464	#37-3799				
KDAT02		017466	#37-3800				
KDAT03		017470	#37-3801				
KDONE		017502	37-3756	37-3770	37-3777	37-3786	#37-3808
KERRO		017324	37-3731	#37-3760			
KERR1		017370	37-3742	37-3754	#37-3772		
KERR2		017414	37-3749	#37-3781			
KPATO		017472	37-3726	#37-3803			
KPAT1		017474	#37-3804				
KPAT2		017476	#37-3805				
K1		017216	37-3721	#37-3723			
K10		017344	37-3761	37-3763	#37-3766		
K2		017242	#37-3733	37-3773	37-3782		
K3		017244	#37-3734	37-3760			
K4		017246	#37-3735	37-3762			
K5		017302	#37-3747	37-3750			
K6		017310	37-3748	#37-3750			
K7		017322	37-3753	#37-3756			
LDAT10		020006	38-3826	38-3833	38-3852	38-3872	38-3884 #38-3898
LDAT11		020010	#38-3899				
LDAT12		020012	38-3846	*38-3850	38-3866	#38-3900	
LDAT13		020014	*38-3851	#38-3901			
LDAT00		020020	38-3843	38-3853	38-3873	38-3878	#38-3902
LDAT01		020022	38-3885	#38-3903			
LDAT02		020024	#38-3904				
LDAT03		020026	#38-3905				
LDONE		020030	38-3861	38-3868	38-3875	38-3887	#38-3907
LD1		043567	#71-6517				
LD2		043617	#71-6518				
LERR1		017642	38-3848	#38-3863			
LERR2		017714	38-3858	#38-3877			
LERR3		017666	#38-3870	38-3881			
LF	=	000012	#19-1783	59-6277	59-6277		
LFIX1		042745	#71-6500	75-7171	75-7173		
LFIX2		043015	#71-6501	75-7171	75-7173		
LFPS1		043554	#71-6516				
LOOP		004456	#24-1816	55-6269			
LPAT10		017764	38-3823	#38-3889			
LPAT11		017766	#38-3890				
LPAT12		017770	#38-3891				
LPAT13		017772	#38-3892				
LPAT20		017774	38-3827	38-3877	#38-3894		

SYMBOL	CROSS REFERENCE	VALUE	REFERENCES
LPAT21		017776	#38-3895
LPAT22		020000	#38-3896
LPAT23		020002	#38-3897
L1		017514	38-3818 #38-3820
L2		017554	#38-3837 38-3864 38-3871 38-3883
L3		017556	#38-3838
L4		017560	#38-3840
L5		017630	#38-3856 38-3859
L6		017636	38-3857 #38-3859
MDAT00		020350	39-3941 39-3944 39-3954 39-3964 39-3980 #39-4006
MDAT01		020352	#39-4007
MDAT02		020354	#39-4008
MDAT03		020356	#39-4009
MDONE		020360	39-3952 39-3966 39-3976 39-3983 39-3994 #39-4011
MERRO		020204	39-3937 #39-3968
MERR1		020242	39-3947 39-3958 #39-3978
MERR2		020156	#39-3961
MERR3		020270	39-3926 #39-3985
MNUMBE	=	000214	#19-815 21-1799
MNUM0		044535	33-3182 33-3183 #71-6547
MNUM1		044543	33-3184 33-3185 #71-6548
MNUM2		044550	33-3186 33-3187 #71-6549
MNUM3		044555	33-3188 33-3189 #71-6550
MNUM4		044564	33-3190 33-3191 #71-6551
MNUM5		044572	33-3192 33-3193 #71-6552
MPAT10		020330	39-3922 #39-3996
MPAT11		020332	#39-3997
MPAT12		020334	#39-3998
MPAT13		020336	#39-3999
MPAT20		020340	39-3963 39-3981 #39-4001
MPAT21		020342	#39-4002
MPAT22		020344	#39-4003
MPAT23		020346	#39-4004
MS1		044101	#71-6531 75-7198 75-7211 75-7228 75-7244
MS10		044255	#71-6538 75-7201 75-7202 75-7204 75-7214 75-7215 75-7224 75-7225 75-7236
			75-7259 75-7259 75-7288 75-7289
MS11		044270	#71-6539 75-7202 75-7215 75-7225 75-7259 75-7289
MS12		044315	#71-6540 75-7204 75-7236
MS13		044353	#71-6541 75-7204 75-7237
MS14		044370	#71-6542 75-7205
MS15		044456	#71-6544 75-7205
MS16		044503	30-2626 30-2662 #71-6545
MS17		044510	#71-6546 75-7227 75-7229
MS2		044113	#71-6532 75-7199 75-7212 75-7228 75-7245
MS20		044600	#71-6553 75-7227
MS21		044641	#71-6554 75-7228 75-7228 75-7230 75-7231 75-7244 75-7245
MS22		044700	#71-6555 75-7230
MS23		044730	#71-6556 75-7231
MS24		044757	#71-6557 75-7229
MS25		045052	#71-6559 75-7232 75-7240
MS26		045074	#71-6560 75-7233 75-7240
MS27		045111	#71-6561 75-7240 75-7240

SYMBOL	CROSS REFERENCE	REFERENCES	SEQUENCE	CREF	V01					
SYMBOL	VALUE									
MS3	044120	#71-6533	75-7198	75-7199	75-7211	75-7212				
MS30	045120	#71-6562	75-7232	75-7234						
MS31	045127	#71-6563	75-7232	75-7234						
MS32	045136	#71-6564	75-7233	75-7235						
MS33	045145	#71-6565	75-7233	75-7235						
MS34	045154	#71-6566	75-7233	75-7235						
MS35	045163	31-2725	32-2902	#71-6567						
MS36	045170	31-2773	32-2944	#71-6568						
MS37	045177	#71-6569	75-7289	75-7294						
MS4	044147	#71-6534	75-7198	75-7199	75-7211	75-7212				
MS40	045215	#71-6570	75-7295							
MS41	045251	#71-6572	75-7297							
MS415	045231	#71-6571	75-7295							
MS42	045275	#71-6573	75-7298							
MS43	045313	#71-6574	75-7298							
MS44	045330	#71-6575	75-7299							
MS5	044161	#71-6535								
MS6	044223	#71-6536	75-7201	75-7214	75-7224	75-7258	75-7288			
MS7	044241	#71-6537	75-7201	75-7202	75-7204	75-7214	75-7215	75-7224	75-7225	75-7236
		75-7258	75-7259	75-7288	75-7289					
M1	020034	#39-3919								
M15	020054	#39-3928	39-3962	39-3974	39-3979	39-3991				
M2	020060	#39-3931	39-3968	39-3990						
M3	020062	#39-3932								
M4	020064	#39-3933								
M5	020120	39-3946	#39-3948							
M6	020124	#39-3949	39-3951							
M7	020134	39-3950	#39-3954							
M8	020144	#39-3956	39-3959							
M9	020154	39-3957	#39-3959							
NDAT10	021056	40-4072	40-4129	40-4137	#40-4146					
NDAT11	021060	#40-4147								
NDAT12	021062	#40-4148								
NDAT13	021064	#40-4149								
NDAT00	021014	40-4037	40-4054	40-4061	40-4071	40-4128	#40-4133			
NDAT01	021016	#40-4134								
NDAT02	021020	#40-4135								
NDAT03	021022	#40-4136								
NDONE	021066	40-4077	40-4098	40-4106	40-4112	40-4124	40-4131	#40-4151		
NERRO	020556	40-4030	#40-4085							
NERR1	020656	40-4045	#40-4107							
NERR10	020610	40-4086	#40-4094							
NERR11	020622	40-4091	#40-4100							
NERR2	020712	40-4049	#40-4113							
NERR20	020664	#40-4108	40-4114	40-4116	40-4118					
NERR3	020722	40-4059	#40-4115							
NERR4	020732	40-4067	#40-4117							
NERR5	020742	40-4069	#40-4120							
NERR6	020766	40-4079	#40-4126							
NOOP1	043141	#71-6505	75-7177	75-7181						
NOOP10	043417	#71-6513	75-7179	75-7181						
NOOP11	043510	#71-6515	75-7181							

SYMBOL	CROSS REFERENCE	VALUE	REFERENCES
OPAT13		021560	#41-4286
OPAT20		021536	41-4181 41-4262 #41-4278
OPAT21		021540	41-4168 41-4184 41-4188 41-4192 41-4202 41-4230 #41-4279
OPAT22		021542	#41-4280
OPAT23		021544	#41-4281
OPAT24		021546	#41-4282
01		021072	#41-4162
010		021222	#41-4204 41-4206
011		021232	41-4205 #41-4209
012		021234	41-4182 #41-4211
013		021250	#41-4214 41-4216
014		021260	41-4215 #41-4219
02		021116	#41-4174 41-4251 41-4268
03		021120	#41-4175 41-4227
04		021122	#41-4176 41-4225 41-4263
05		021140	#41-4184
06		021150	41-4185 #41-4188
07		021160	41-4189 #41-4192
08		021174	#41-4196 41-4198
09		021206	41-4197 #41-4201
PDAT10		022226	42-4311 42-4319 42-4324 42-4345 42-4349 42-4353 42-4357 42-4361 42-4379
			42-4385 #42-4403
PDAT11		022230	#42-4404
PDAT12		022232	#42-4405
PDAT13		022234	#42-4406
PDAT00		022236	42-4316 42-4320 42-4329 42-4386 #42-4408
PDAT01		022240	#42-4409
PDAT02		022242	#42-4410
PDAT03		022244	#42-4411
PDONE		022246	42-4327 42-4371 42-4381 42-4388 42-4396 #42-4413
PERRO		021714	42-4309 #42-4337
PERR1		022134	42-4333 #42-4383
PERR10		021734	42-4340 #42-4343
PERR11		021744	42-4338 #42-4345
PERR12		021762	42-4346 #42-4349
PERR13		022000	42-4350 #42-4353
PERR14		022016	42-4354 #42-4357
PERR15		022034	42-4358 #42-4361
PERR16		022044	42-4362 #42-4364
PERR17		022052	42-4344 42-4348 42-4352 42-4356 42-4360 #42-4366
PERR2		022162	42-4335 #42-4390
PERR20		022100	42-4363 #42-4373
PERR21		022110	42-4326 #42-4376
PERR22		022116	42-4375 #42-4377
PFECAD		042506	66-6427 #66-6431
PFECDH		042456	66-6427 #66-6429
PFECM		042422	66-6427 #66-6428
PFECFT		042516	66-6427 #66-6432
PFECWS		042412	#66-6427
PIRQ	=	177772	#19-1783
PIRQE	=	000240	#19-1783
POWERM		042672	65-6291 #71-6496

SYMBOL	CROSS REFERENCE	VALUE	REFERENCES									
SPAT11		016666	#35-3604									
SPAT12		016670	#35-3605									
SPAT13		016672	#35-3606									
STACK	=	001100	#19-1783	24-1811	70-6491							
START		003616	19-1794	#24-1811	65-6291							
STFS1		043630	#71-6519									
STKLMT	=	177774	#19-1783									
STST1		043760	#71-6528									
ST1		043643	#71-6520									
ST2		043660	#71-6521									
SWR		001140	#20-1795	24-1811	*24-1811	24-1811	*24-1811	*24-1811	24-1812	25-1887	25-1889	
			33-3264	33-3266	55-6269	56-6271	56-6271	56-6271	56-6271	56-6271	57-6273	
			57-6273	57-6273	57-6273	57-6273	63-6285	63-6285	65-6291	65-6291	70-6480	
SWREG		000176	#19-1794	24-1811	24-1812	63-6285	63-6285					
SW0	=	000001	#19-1783									
SW00	=	000001	#19-1783	19-1783								
SW01	=	000002	#19-1783	19-1783								
SW02	=	000004	#19-1783	19-1783								
SW03	=	000010	#19-1783	19-1783								
SW04	=	000020	#19-1783	19-1783								
SW05	=	000040	#19-1783	19-1783								
SW06	=	000100	#19-1783	19-1783								
SW07	=	000200	#19-1783	19-1783								
SW08	=	000400	#19-1783	19-1783								
SW09	=	001000	#19-1783	19-1783								
SW1	=	000002	#19-1783									
SW10	=	002000	#19-1783									
SW11	=	004000	#19-1783									
SW12	=	010000	#19-1783									
SW13	=	020000	#19-1783	25-1887	33-3264							
SW14	=	040000	#19-1783									
SW15	=	100000	#19-1783									
SW2	=	000004	#19-1783									
SW3	=	000010	#19-1783									
SW4	=	000020	#19-1783									
SW5	=	000040	#19-1783									
SW6	=	000100	#19-1783									
SW7	=	000200	#19-1783	25-1889	33-3266							
SW8	=	000400	#19-1783									
SW9	=	001000	#19-1783									
S1		016144	#35-3464									
S10		016322	#35-3510									
S11		016346	#35-3518	35-3521								
S12		016356	35-3519	#35-3521								
S2		016204	#35-3477	35-3564	35-3571							
S3		016206	#35-3478	35-3525	35-3579							
S4		016212	#35-3480	35-3581								
S5		016236	#35-3488	35-3491								
S6		016246	35-3489	#35-3491								
S7		016254	#35-3496	35-3529								
S8		016314	#35-3507	35-3567	35-3574	35-3590						
S9		016316	#35-3508	35-3544	35-3592							

SYMBOL CROSS REFERENCE

SYMBOL VALUE
TAB = 000011

REFERENCES

#19-1781	71-6498	71-6500	71-6501	71-6505	71-6511	71-6512	71-6513	71-6513
71-6513	71-6516	71-6516	71-6517	71-6517	71-6517	71-6518	71-6518	71-6519
71-6519	71-6520	71-6520	71-6521	71-6521	71-6522	71-6523	71-6524	71-6525
71-6526	71-6527	71-6528	71-6528	71-6529	71-6530	71-6530	71-6530	71-6542
71-6542	71-6542	71-6543	71-6553	71-6553	71-6557	71-6557	71-6557	73-6855
73-6855	73-6856	73-6856	73-6856	73-6857	73-6857	73-6858	73-6858	73-6859
73-6860	73-6860	73-6861	73-6861	73-6862	73-6862	73-6863	73-6863	73-6872
73-6872	73-6873	73-6874	73-6874	73-6875	73-6875	73-6876	73-6876	73-6877
73-6877	73-6881	73-6881	73-6882	73-6882	73-6885	73-6885	73-6885	73-6886
73-6886	73-6887	73-6887	73-6892	73-6892	73-6893	73-6900	73-6900	73-6901
73-6905	73-6907	73-6907	73-6908	73-6912	73-6912	73-6922	73-6922	73-6923
73-6923	73-6947	73-6947	73-6953	73-6953	73-6956	73-6956	73-6957	73-6957
73-6969	73-6969	73-6974	73-6974	73-6975	73-6975	73-6985	73-6985	73-6989
73-6989	73-7018	73-7018	73-7018					

TBITVE = 000014

TDAT10	012116	#19-1783	*24-1811	*24-1811	32-2931	32-2936	32-2965	32-3001	#32-3039
TDAT11	012120	32-2888	32-2893	32-2911					
TDAT12	012122	#32-3040							
TDAT13	012124	*32-2962	#32-3041						
TDAT00	012106	*32-2963	#32-3042						
TDAT01	012110	32-2907	32-2910	32-2959	32-2964	32-3002	#32-3034		
TDAT02	012112	#32-3035							
TDAT03	012114	32-2916	32-2970	#32-3036					
TDONE	012126	32-2973	#32-3037						
TERR0	011674	32-2978	32-2997	32-3004	32-3011	32-3022	#32-3044		
TERR1	011756	32-2900	#32-2982						
TERR2	011776	32-2920	32-2972	#32-3000					
TERR25	012012	32-2922	#32-3007						
TERR3	012042	#32-3009	32-3014						
TERR4	012024	32-2954	#32-3017						
TKVEC	= 000060	32-2975	#32-3012						
TOCTNM	042522	#19-1783							
TPAT10	012066	66-6360	66-6361	66-6369	66-6402	#66-6434			
TPAT11	012070	32-2887	32-2930	#32-3024					
TPAT12	012072	#32-3025							
TPAT13	012074	32-2970	#32-3026						
TPAT20	012076	32-2973	#32-3027						
TPAT21	012100	32-2896	32-2939	#32-3029					
TPAT22	012102	#32-3030							
TPAT23	012104	#32-3031							
TPVEC	= 000064	#32-3032							
TRAPVE	= 000034	#19-1783							
TRTVEC	= 000014	#19-1783	*24-1811	*24-1811					
TST1	004456	#19-1783							
TST10	011312	#25-1835							
TST11	012130	#32-2884							
TST12	015212	#33-3180							
TST13	016142	#34-3358							
TST14	016706	#35-3463							
TST15	017206	#36-3621							
TST16	017504	#37-3720							
TST17	020032	#38-3817							
		#39-3917							

SYMBOL CROSS REFERENCE		REFERENCES								
SYMBOL	VALUE									
TST2	005050	#26-1946								
TST20	020362	#40-4020								
TST21	021070	#41-4160								
TST22	021574	#42-4301								
TST23	022250	#43-4422								
TST24	022750	#44-4570								
TST25	024110	#45-4782								
TST26	024706	#46-4924								
TST27	025522	#47-5074								
TST3	005156	#27-1991								
TST30	026132	#48-5170								
TST31	026440	#49-5250								
TST32	027506	#50-5426								
TST33	031324	#51-5684								
TST34	032660	#52-5896								
TST35	034710	#53-6120								
TST36	035474	#54-6204								
TST37	036060	#54-6267								
TST4	006122	#28-2195								
TST5	006370	#29-2275								
TST6	006534	#30-2333								
TST7	010440	#31-2705								
TYPDS	= 104405	#64-6287								
TYPE	= 104401	24-1812	24-1813	24-1814	33-3284	33-3291	33-3307	33-3317	33-3318	55-6269
		55-6269	55-6269	55-6269	55-6269	56-6271	56-6271	57-6273	57-6273	59-6277
		60-6279	61-6281	63-6285	63-6285	63-6285	63-6285	63-6285	63-6285	#64-6287
		65-6291	66-6301	66-6329	66-6331	66-6336	66-6338	66-6386	66-6393	66-6408
		66-6418	66-6436							
		63-6285	#64-6287	66-6312	66-6353	66-6435				
TYPOC	= 104402									
TYPON	= 104404	#64-6287								
TYPOS	= 104403	55-6269	55-6269	55-6269	#64-6287	66-6377				
T1	011314	#32-2885								
T10	011462	32-2919	#32-2921							
T105	011470	32-2914	#32-2924							
T11	011472	#32-2928								
T12	011500	32-2928	#32-2930							
T13	011514	#32-2933	32-2934							
T14	011556	32-2943	#32-2946							
T15	011560	#32-2947	32-2986	32-3018						
T16	011562	#32-2948	32-2988							
T17	011600	32-2953	#32-2956							
T2	011340	#32-2890	32-2891							
T20	011640	#32-2967	32-2977							
T21	011654	#32-2972	32-2974							
T22	011656	32-2971	#32-2973							
T23	011670	32-2968	#32-2977							
T3	011406	32-2901	#32-2903							
T4	011410	#32-2904	32-2982							
T5	011412	#32-2905	32-2984							
T6	011436	#32-2913	32-2924							
T7	011452	#32-2918	32-2921							
UDONE	024110	44-4682	44-4691	44-4709	44-4724	44-4735	44-4740	#44-4772		

SYMBOL CROSS REFERENCE		REFERENCES							
SYMBOL	VALUE								
UERR0	023554	44-4579	#44-4685						
UERR1	023600	44-4599	44-4656	#44-4695					
UERR10	023606	#44-4697	44-4711						
UERR11	023650	#44-4706							
UERR2	023664	44-4620	44-4638	#44-4710					
UERR20	023672	44-4696	#44-4712						
UERR21	023734	#44-4721							
UERR3	023750	44-4670	#44-4727						
UERR4	024000	44-4734	#44-4736						
UFLAG	024074	*44-4571	44-4657	*44-4664	#44-4766				
UPAT00	024024	44-4572	44-4583	44-4605	44-4624	44-4642	44-4660	44-4673	#44-4741
UPAT01	024026	#44-4742							
UPAT02	024030	#44-4743							
UPAT03	024032	#44-4744							
UPAT10	024034	44-4589	#44-4746						
UPAT11	024036	#44-4747							
UPAT12	024040	#44-4748							
UPAT13	024042	#44-4749							
UPAT20	024044	44-4611	#44-4751						
UPAT21	024046	#44-4752							
UPAT22	024050	#44-4753							
UPAT23	024052	#44-4754							
UPAT30	024054	44-4629	#44-4756						
UPAT31	024056	#44-4757							
UPAT32	024060	#44-4758							
UPAT33	024062	#44-4759							
UPAT40	024064	44-4647	44-4675	#44-4761					
UPAT41	024066	#44-4762							
UPAT42	024070	#44-4763							
UPAT43	024072	#44-4764							
UROM1	024102	*44-4585	*44-4607	*44-4626	*44-4644	44-4712	#44-4769		
UROM2	024104	*44-4586	*44-4608	*44-4627	*44-4645	44-4713	#44-4770		
UROM3	024106	*44-4587	*44-4609	*44-4628	*44-4646	44-4698	#44-4771		
UTMP1	024076	*44-4577	44-4585	44-4644	*44-4665	#44-4767			
UTMP2	024100	*44-4578	44-4607	44-4626	*44-4666	#44-4768			
U0	022766	#44-4574	44-4575						
U1	023016	#44-4580	44-4667						
U10	023374	#44-4648	44-4651						
U11	023376	#44-4649	44-4685						
U12	023430	44-4655	#44-4657						
U13	023446	#44-4662	44-4663						
U14	023500	44-4658	#44-4668						
U15	023534	#44-4676	44-4679	44-4736					
U16	023536	#44-4677	44-4727						
U2	023066	#44-4590	44-4592						
U3	023120	44-4598	#44-4601						
U4	023170	#44-4613	44-4615						
U5	023222	44-4619	#44-4621						
U6	023272	#44-4631	44-4633						
U7	023324	44-4637	#44-4639						
WDAPO0	024674	#45-4910							
WDAPO1	024676	#45-4911							

SYMBOL	CROSS REFERENCE	REFERENCES								
SYMBOL	VALUE									
WDAT02	024700	#45-4912								
WDAT03	024702	#45-4913								
WDONE	024704	45-4802	45-4810	45-4830	45-4838	45-4858	45-4866	45-4888	45-4891	45-4896
		#45-4915								
WPAT00	024664	45-4786	45-4789	45-4793	45-4795	45-4814	45-4819	45-4823	45-4825	45-4842
		45-4845	45-4849	45-4851	45-4870	45-4875	45-4879	45-4881	45-4900	45-4901
		45-4902	45-4903	#45-4905						
WPAT01	024666	#45-4906								
WPAT02	024670	#45-4907								
WPAT03	024672	#45-4908								
WSETUP	024632	45-4800	45-4856	45-4886	#45-4900					
W1	024112	#45-4783								
W10	024334	45-4828	#45-4831							
W11	024362	45-4833	#45-4839							
W12	024416	45-4844	#45-4846							
W13	024442	#45-4853	45-4859							
W14	024456	45-4854	#45-4859							
W15	024504	45-4861	#45-4867							
W16	024544	45-4872	#45-4876							
W17	024570	#45-4883	45-4889							
W2	024146	45-4788	#45-4790							
W20	024604	45-4884	#45-4889							
W3	024172	#45-4797	45-4803							
W4	024210	45-4798	#45-4803							
W5	024240	45-4805	#45-4811							
W6	024300	45-4816	#45-4820							
W7	024324	#45-4827	45-4831							
XAPT11	025502	#46-5057								
XDAT00	025460	46-4936	46-4959	46-4982	46-5005	46-5023	46-5028	#46-5046		
XDAT01	025462	#46-5047								
XDAT02	025464	#46-5048								
XDAT03	025466	#46-5049								
XDONE	025520	46-5017	46-5025	46-5031	46-5039	46-5044	#46-5065			
XERR1	025344	46-4942	46-4965	#46-5021						
XERR2	025426	46-4947	46-4970	#46-5035						
XERR3	025372	46-4988	46-5011	#46-5026						
XERR4	025442	46-4993	46-5016	#46-5040						
XPAT00	025470	46-4928	46-4938	46-4974	46-4984	#46-5051				
XPAT01	025472	#46-5052								
XPAT02	025474	#46-5053								
XPAT03	025476	#46-5054								
XPAT10	025500	46-4932	46-4955	46-4978	46-5001	46-5021	46-5026	#46-5056		
XPAT12	025504	#46-5058								
XPAT13	025506	#46-5059								
XPAT20	025510	46-4951	46-4961	46-4997	46-5007	#46-5060				
XPAT21	025512	#46-5061								
XPAT22	025514	#46-5062								
XPAT23	025516	#46-5063								
XTMP	025456	*46-4929	*46-4952	*46-4975	*46-4998	46-5022	46-5027	46-5029	#46-5045	
X1	024710	#46-4925								
X10	025112	46-4964	#46-4966							
X11	025126	46-4969	#46-4971							

SYMBOL	CROSS REFERENCE VALUE	REFERENCES					
X12	025166	46-4977	#46-4979				
X13	025212	#46-4986	46-4989				
X14	025220	46-4987	#46-4989				
X15	025234	46-4992	#46-4994				
X16	025274	46-5000	#46-5002				
X17	025320	#46-5009	46-5012				
X2	024750	46-4931	#46-4933				
X20	025326	46-5010	#46-5012				
X21	025342	46-5015	#46-5017				
X3	024774	#46-4940	46-4943				
X4	025002	46-4941	#46-4943				
X5	025020	46-4946	#46-4948				
X6	025060	46-4954	#46-4956				
X7	025104	#46-4963	46-4966				
YDAT00	026070	47-5088	47-5108	47-5117	47-5125	#47-5141	
YDAT01	026072	#47-5142					
YDAT02	026074	#47-5143					
YDAT03	026076	#47-5144					
YDONE	026130	47-5105	47-5120	47-5128	47-5134	#47-5161	
YERR1	025730	47-5110	#47-5113				
YERR2	025772	47-5112	#47-5121				
YERR3	026034	47-5097	#47-5129				
YFLAG	026060	*47-5075	47-5098	*47-5100	#47-5136		
YPAT00	026100	47-5076	47-5102	#47-5146			
YPAT01	026102	#47-5147					
YPAT02	026104	#47-5148					
YPAT03	026106	#47-5149					
YPAT10	026110	47-5077	47-5101	#47-5151			
YPAT11	026112	#47-5152					
YPAT12	026114	#47-5153					
YPAT13	026116	#47-5154					
YPAT20	026120	47-5082	47-5116	47-5124	#47-5156		
YPAT21	026122	#47-5157					
YPAT22	026124	#47-5158					
YPAT23	026126	#47-5159					
YTMP1	026062	*47-5076	47-5084	*47-5101	47-5107	47-5115	47-5123 #47-5137
YTMP2	026064	*47-5077	47-5091	*47-5102	47-5118	47-5126	#47-5138
YTMP3	026066	*47-5078	47-5095	*47-5103	47-5132	#47-5139	
Y1	025552	#47-5079	47-5104				
Y2	025600	#47-5085	47-5114	47-5122	47-5130		
Y3	025624	#47-5092	47-5094				
Y4	025642	47-5096	#47-5098				
Y5	025702	47-5099	#47-5105				
Y6	025704	47-5093	#47-5106				
Y7	025720	#47-5109	47-5111				
ZDAT00	026376	48-5183	48-5205	#48-5220			
ZDAT01	026400	#48-5221					
ZDAT02	026402	#48-5222					
ZDAT03	026404	#48-5223					
ZDONE	026436	48-5200	48-5208	48-5214	#48-5240		
ZERR1	026302	48-5189	#48-5201				
ZERR2	026344	48-5193	#48-5209				

SYMBOL CROSS REFERENCE

SYMBOL	VALUE	REFERENCES								
\$CM3	= 000024	#20-1795	20-1795	20-1795						
\$CM4	= 000024	#20-1795	20-1795	20-1795	#20-1795	20-1795	20-1795	#20-1795	20-1795	20-1795
		#20-1795	20-1795	20-1795	#20-1795	20-1795	20-1795	#20-1795	20-1795	20-1795
		#20-1795	20-1795	20-1795	#20-1795	20-1795	20-1795	#20-1795	20-1795	20-1795
		#20-1795	20-1795	20-1795	#20-1795	20-1795	20-1795	#20-1795	20-1795	20-1795
		#20-1795	20-1795	20-1795	#20-1795	20-1795	20-1795	#20-1795	20-1795	20-1795
		#20-1795	20-1795	20-1795	#20-1795	20-1795	20-1795	#20-1795	20-1795	20-1795
		#20-1795	20-1795	20-1795	#20-1795	20-1795	20-1795	#20-1795	20-1795	20-1795
		#20-1795	20-1795	20-1795	#20-1795	20-1795	20-1795	#20-1795	20-1795	20-1795
\$CNTLG	041511	63-6285	#63-6285							
\$CNTLU	041504	63-6285	#63-6285							
\$CPUOP	001344	#20-1795								
\$CRLF	001313	#20-1795	33-3318	55-6269	57-6273	57-6273	57-6273	59-6277	59-6277	59-6277
		63-6285	66-6301	66-6332	66-6339	66-6419	75-7190	75-7198	75-7198	75-7199
		75-7199	75-7200	75-7201	75-7201	75-7202	75-7204	75-7204	75-7204	75-7205
		75-7205	75-7211	75-7211	75-7211	75-7212	75-7214	75-7214	75-7214	75-7215
		75-7220	75-7220	75-7224	75-7224	75-7225	75-7225	75-7227	75-7227	75-7228
		75-7228	75-7228	75-7228	75-7229	75-7229	75-7230	75-7230	75-7230	75-7231
		75-7232	75-7233	75-7236	75-7237	75-7240	75-7240	75-7244	75-7244	75-7245
		75-7245	75-7258	75-7258	75-7259	75-7259	75-7287	75-7288	75-7288	75-7289
		75-7289	75-7289	75-7294	75-7294	75-7295	75-7295	75-7295	75-7295	75-7297
		75-7297	75-7298	75-7298	75-7298	75-7298	75-7299	75-7299	75-7299	
\$DBLK	040614	61-6281	61-6281	#61-6281						
\$DDW0	001402	#20-1795								
\$DDW1	001404	#20-1795								
\$DDW10	001426	#20-1795								
\$DDW11	001430	#20-1795								
\$DDW12	001432	#20-1795								
\$DDW13	001434	#20-1795								
\$DDW14	001436	#20-1795								
\$DDW15	001440	#20-1795								
\$DDW2	001406	#20-1795								
\$DDW3	001410	#20-1795								
\$DDW4	001412	#20-1795								
\$DDW5	001414	#20-1795								
\$DDW6	001416	#20-1795								
\$DDW7	001420	#20-1795								
\$DDW8	001422	#20-1795								
\$DDW9	001424	#20-1795								
\$DEVCT	001326	#20-1795								
\$DEVM	001374	#20-1795								
\$DOAGN	036442	55-6269	55-6269	55-6269	#55-6269					
\$DTBL	040604	61-6281	#61-6281							
\$ENDAD	036432	22-1806	24-1812	#55-6269	57-6273					
\$ENDCT	036124	24-1811	#55-6269							
\$ENULL	036506	#55-6269								
\$ENV	001336	#20-1795	24-1812	57-6273	59-6277	62-6283	62-6283			
\$ENVM	001337	#20-1795	24-1811	59-6277	59-6277	62-6283				
\$EOP	036060	#55-6269	67-6452	68-6461	69-6470					
\$EOPCT	036112	*24-1811	#55-6269	55-6269						
\$ERFLG	001103	#20-1795	56-6271	*56-6271	56-6271	56-6271	*56-6271	56-6271	56-6271	*57-6273
		57-6273	57-6273							
\$ERMAX	001115	#20-1795	*24-1811	56-6271	*56-6271	56-6271	56-6271	56-6271	56-6271	

SYMBOL	CROSS REFERENCE VALUE	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES
\$ERROR	037146	24-1811	#57-6273							
\$ERRPC	001116	#20-1795	*25-1895	*33-3273	*57-6273	*57-6273	57-6273	57-6273	*57-6273	57-6273
		57-6273	66-6304	66-6311	66-6431					
\$ERRTB	001442	#21-1795	66-6325							
\$ERTTL	001112	#20-1795	55-6269	55-6269	55-6269	*55-6269	*57-6273	57-6273	57-6273	57-6273
\$ESCAP	001304	#20-1795	*24-1811	*56-6271	57-6273	57-6273	57-6273	57-6273		
\$ETABL	001336	#20-1795								
\$ETEND	001442	#20-1795	23-1808							
\$FATAL	001320	#20-1795	*62-6283	*66-6320						
\$FFLG	041070	*62-6283	*62-6283	62-6283	*62-6283	#62-6283				
\$FILLC	001156	#20-1795	59-6277	59-6277	59-6277					
\$FILLS	001155	#20-1795	59-6277	59-6277						
\$GDADR	001120	#20-1795								
\$GDDAT	001124	#20-1795								
\$GET42	036404	55-6269	55-6269	#55-6269						
\$GTSWR	041142	#63-6285	64-6287	64-6287						
\$HD	= 000003	19-1777	19-1777	19-1777						
\$HIBTS	003602	#23-1808								
\$ICNT	001104	#20-1795	*56-6271	56-6271	*56-6271	56-6271	56-6271	56-6271		
\$ILLUP	042002	65-6291	65-6291	#65-6291						
\$INTAG	001135	#20-1795	63-6285	63-6285	63-6285					
\$ITEMB	001114	#20-1795	*25-1896	*33-3274	*57-6273	57-6273	57-6273	57-6273	57-6273	66-6307
\$LF	001314	#20-1795	57-6273	57-6273	59-6277	59-6277				
\$LFLG	041067	*62-6283	#62-6283							
\$LOOP	036500	55-6269	#55-6269							
\$LPADR	001106	#20-1795	*24-1811	*56-6271	*56-6271	56-6271	56-6271	56-6271	56-6271	56-6271
\$LPERR	001110	#20-1795	*24-1811	*25-1837	*26-1947	*27-1992	*27-2009	*27-2022	*27-2037	*27-2050
		*27-2066	*27-2079	*27-2094	*28-2196	*29-2276	*30-2335	*30-2377	*31-2706	*31-2758
		*32-2885	*32-2928	*33-3182	*33-3183	*33-3184	*33-3185	*33-3186	*33-3187	*33-3188
		*33-3189	*33-3190	*33-3191	*33-3192	*33-3193	*34-3359	*35-3464	*35-3496	*36-3622
		*37-3721	*38-3818	*44-4580	*44-4601	*44-4621	*44-4639	*44-4668	*45-4783	*45-4811
		*45-4839	*45-4867	*46-4925	*46-4948	*46-4971	*46-4994	*47-5079	*48-5174	*49-5251
		*49-5280	*49-5314	*50-5428	*50-5456	*50-5484	*50-5513	*50-5542	*50-5570	*51-5686
		*51-5710	*51-5738	*51-5761	*51-5790	*51-5811	*52-5898	*52-5927	*52-5949	*52-5971
		*52-5999	*52-6027	*52-6056	*53-6122	*53-6151	*54-6206	*54-6228	56-6271	*56-6271
		56-6271	56-6271	57-6273						
\$MADR1	001350	#20-1795								
\$MADR2	001354	#20-1795								
\$MADR3	001360	#20-1795								
\$MADR4	001364	#20-1795								
\$MAIL	001316	#20-1795	23-1808	23-1808	24-1811	24-1812	56-6271	57-6273	59-6277	
\$MAMS1	001346	#20-1795								
\$MAMS2	001352	#20-1795								
\$MAMS3	001356	#20-1795								
\$MAMS4	001362	#20-1795								
\$MBADR	003604	#23-1808								
\$MFLG	041066	*62-6283	62-6283	*62-6283	#62-6283					
\$MNEW	041527	63-6285	#63-6285							
\$MSGAD	001332	#20-1795	*62-6283	62-6283						
\$MSGLG	001334	#20-1795	*62-6283							
\$MSGTY	001316	#20-1795	62-6283	*62-6283	62-6283	*62-6283				
\$MSWR	041516	63-6285	#63-6285							

SYMBOL	CROSS REFERENCE	VALUE	REFERENCES
SMTYP1		001347	#20-1795
SMTYP2		001353	#20-1795
SMTYP3		001357	#20-1795
SMTYP4		001363	#20-1795
SMXCNT		037140	56-6271 56-6271 56-6271 #56-6271
SNULL		001154	#20-1795 59-6277 59-6277 59-6277
SNWTST	=	000001	#24-1835 24-1835 #25-1835 25-1835 #25-1946 25-1946 #26-1946 26-1946 #26-1991 26-1991 #27-1991 27-1991 #27-2195 27-2195 #28-2195 28-2195 #28-2275 28-2275 #29-2275 29-2275 #29-2333 29-2333 #30-2333 30-2333 #30-2705 30-2705 #31-2705 31-2705 #31-2884 31-2884 #32-2884 32-2884 #32-3180 32-3180 #33-3180 33-3180 #33-3358 33-3358 #34-3358 34-3358 #34-3463 34-3463 #35-3463 35-3463 #35-3621 35-3621 #36-3621 36-3621 #36-3720 36-3720 #37-3720 37-3720 #37-3817 37-3817 #38-3817 38-3817 #38-3917 38-3917 #39-3917 39-3917 #39-4020 39-4020 #40-4020 40-4020 #40-4160 40-4160 #41-4160 41-4160 #41-4301 41-4301 #42-4301 42-4301 #42-4422 42-4422 #43-4422 43-4422 #43-4570 43-4570 #44-4570 44-4570 #44-4782 44-4782 #45-4782 45-4782 #45-4924 45-4924 #46-4924 46-4924 #46-5074 46-5074 #46-5074 47-5074 #47-5170 47-5170 #48-5170 48-5170 #48-5250 48-5250 #49-5250 49-5250 #49-5426 49-5426 #50-5426 50-5426 #50-5684 50-5684 #51-5684 51-5684 #51-5896 51-5896 #52-5896 52-5896 #52-6120 52-6120 #53-6120 53-6120 #53-6204 53-6204 #54-6204 54-6204
SOCNT		040374	*60-6279 *60-6279 #60-6279
SOMODE		040376	*60-6279 *60-6279 60-6279 *60-6279 *60-6279 #60-6279
SOVER		037124	56-6271 56-6271 56-6271 56-6271 #56-6271
SPASS		001324	#20-1795 *24-1811 *55-6269 *55-6269 55-6269 55-6269 55-6269
SPASS2		036514	*55-6269 55-6269 55-6269 #55-6269
SPASTM		003610	#23-1808
SPWRAD		041764	#65-6291
SPWRDN		041624	24-1811 #65-6291 65-6291
SPWRMG		041760	#65-6291
SPWRUP		041676	65-6291 #65-6291
SQUES		001312	#20-1795 57-6273 57-6273 59-6277 59-6277 63-6285
SRDCHR		041354	#63-6285 64-6287 64-6287
SRDDEC	=	*****	64-6287
SRDLIN	=	*****	64-6287
SRDOCT	=	*****	64-6287
SRDSZ	=	000001	#63-6285 63-6285
SREGAD		001160	#20-1795
SREGO		001162	#20-1795 *57-6273 *57-6273 57-6273
SREG1		001164	#20-1795
SREG10		001202	#20-1795
SREG11		001204	#20-1795
SREG12		001206	#20-1795
SREG13		001210	#20-1795
SREG14		001212	#20-1795
SREG15		001214	#20-1795
SREG16		001216	#20-1795
SREG17		001220	#20-1795
SREG2		001166	#20-1795
SREG20		001222	#20-1795
SREG21		001224	#20-1795
SREG22		001226	#20-1795

SYMBOL CROSS REFERENCE		REFERENCES									
SYMBOL	VALUE	SYMBOL	VALUE	SYMBOL	VALUE	SYMBOL	VALUE	SYMBOL	VALUE	SYMBOL	VALUE
\$REG23	001230	#20-1795									
\$REG3	001170	#20-1795									
\$REG4	001172	#20-1795									
\$REG5	001174	#20-1795									
\$REG6	001176	#20-1795									
\$REG7	001200	#20-1795									
\$RESRE	037542	#58-6275	64-6287								
\$RTNAD	036502	#55-6269									
\$RTRN	036476	24-1811	*24-1811	*24-1811	55-6269	#55-6269					
\$R2A	= *****	64-6287									
\$SAVRE	037504	#58-6275	64-6287	64-6287							
\$SAVR6	042006	*65-6291	65-6291	*65-6291	*65-6291	#65-6291					
\$SCOPE	036516	24-1811	#56-6271								
\$SETUP	= 000137	#19-1793	19-1793	#19-1793	19-1793	#19-1793	19-1793	#19-1793	19-1793	#19-1793	#19-1793
		19-1793	#19-1793	19-1793	#19-1793	24-1811	24-1811	24-1811	24-1811	24-1811	24-1811
		24-1811	24-1811	24-1811	24-1811	24-1811	24-1811	24-1811	24-1811	24-1812	24-1812
		24-1812	55-6269	55-6269	56-6271	57-6273	57-6273	57-6273	57-6273	57-6273	63-6285
		63-6285	65-6291								
\$STUP	= 177777	#19-1793	#19-1793	19-1793	#19-1793	#19-1793	19-1793	#19-1793	#19-1793	#19-1793	19-1793
		#19-1793	#19-1793	19-1793	#19-1793	#19-1793	19-1793	#19-1793	#19-1793	#19-1793	19-1793
\$SVLAD	037070	56-6271	#56-6271								
\$SVPC	= 003602	#22-1806	22-1806								
\$SWR	= 177400	19-1777	#19-1777	#19-1779	20-1795	20-1795	20-1795	24-1811	24-1811	24-1811	24-1811
		24-1811	24-1811	25-1835	26-1946	27-1991	28-2195	29-2275	30-2333	31-2705	
		32-2884	33-3180	34-3358	35-3463	36-3621	37-3720	38-3817	39-3917	40-4020	
		41-4160	42-4301	43-4422	44-4570	45-4782	46-4924	47-5074	48-5170	49-5250	
		50-5426	51-5684	52-5896	53-6120	54-6204	55-6269	55-6269	55-6269	55-6269	
		55-6269	56-6271	56-6271	56-6271	56-6271	56-6271	56-6271	56-6271	56-6271	
		56-6271	56-6271	56-6271	56-6271	56-6271	56-6271	56-6271	56-6271	56-6271	
		56-6271	56-6271	56-6271	57-6273	57-6273	57-6273	57-6273	57-6273	57-6273	
		57-6273	57-6273	57-6273	57-6273	57-6273	65-6291				
\$SWREG	001340	#20-1795	24-1811								
\$SWRMK	= 000000	56-6271	56-6271	56-6271	56-6271	56-6271	56-6271	56-6271	56-6271	56-6271	56-6271
		56-6271									
\$SWRMS	= 000200	#19-1780									
\$TAB	042740	66-6409	#71-6498	75-7165	75-7165	75-7167	75-7167	75-7167	75-7168	75-7168	75-7168
		75-7169	75-7171	75-7173	75-7179	75-7180	75-7181	75-7182	75-7185	75-7185	75-7185
		75-7187	75-7187	75-7188	75-7189	75-7191	75-7191	75-7192	75-7193	75-7193	75-7193
		75-7194	75-7196	75-7196	75-7197	75-7198	75-7200	75-7200	75-7203	75-7203	75-7203
		75-7206	75-7206	75-7210	75-7210	75-7213	75-7213	75-7221	75-7221	75-7222	75-7222
		75-7223	75-7223	75-7227	75-7229	75-7229	75-7236	75-7239	75-7239	75-7242	75-7242
		75-7242	75-7244	75-7253	75-7253	75-7258	75-7261	75-7261	75-7276	75-7287	75-7287
		75-7287	75-7294	75-7297	75-7303	75-7303	75-7316	75-7345	75-7345	75-7346	75-7346
\$TBIT	036504	*24-1811	*55-6269	55-6269	55-6269	#55-6269	*65-6291				
\$TERM	= 000030	#64-6289									
\$TESTN	001322	#20-1795	*56-6271	66-6431							
\$THE	043134	#71-6504	75-7177	75-7181							
\$TIMES	001302	#20-1795	*24-1811	*55-6269	*56-6271	56-6271	*56-6271	56-6271	56-6271	56-6271	63-6285
\$TKB	001146	#20-1795	56-6271	59-6277	59-6277	59-6277	59-6277	63-6285	63-6285	63-6285	63-6285
		63-6285	63-6285	63-6285							
\$TKS	001144	#20-1795	56-6271	59-6277	59-6277	59-6277	59-6277	63-6285	63-6285	63-6285	63-6285
		63-6285	63-6285	63-6285	63-6285						

SYMBOL CROSS REFERENCE
SYMBOL VALUE

REFERENCES

		*47-5115	*47-5123	*47-5131	*48-5203	*48-5211	*49-5371	*49-5377	*49-5385	*49-5391
		*49-5395	*49-5401	*50-5599	*50-5603	*50-5606	*50-5612	*50-5618	*50-5621	*50-5627
		*50-5633	*50-5639	*50-5642	*50-5648	*50-5651	*50-5654	*50-5657	*51-5832	*51-5837
		*51-5843	*51-5852	*51-5858	*51-5867	*52-6086	*52-6089	*52-6090	*52-6091	*52-6092
		*52-6093	*52-6094	*52-6095	*52-6096	*52-6097	*52-6098	*52-6099	*52-6100	*53-6181
		*53-6184	*53-6185	*53-6186	*53-6187	*54-6250	*54-6255	*54-6257	*67-6447	75-7165
		75-7167	75-7168	75-7172	75-7180	75-7182	75-7185	75-7187	75-7191	75-7193
		75-7196	75-7198	75-7200	75-7203	75-7206	75-7210	75-7213	75-7221	75-7223
		75-7228	75-7230	75-7232	75-7236	75-7239	75-7242	75-7244	75-7245	75-7253
		75-7261	75-7287	75-7294	75-7297	75-7345				
\$TMP4	001242	#20-1795	*25-1879	*26-1971	*26-1977	*27-2159	*27-2166	*28-2237	*28-2251	*29-2297
		*29-2303	*30-2447	*30-2453	*30-2478	*30-2489	*30-2514	*30-2583	*30-2591	*31-2823
		*31-2830	*31-2851	*32-2995	*32-3001	*32-3020	*33-3256	*33-3271	*35-3551	*35-3557
		*36-3668	*36-3677	*36-3686	*37-3767	*37-3775	*37-3784	*38-3866	*38-3873	*38-3885
		*39-3964	*39-3969	*39-3981	*39-3990	*40-4121	*40-4129	*41-4249	*41-4262	*41-4270
		*42-4379	*42-4386	*43-4511	*43-4528	*44-4707	*44-4722	*44-4737	*45-4807	*45-4835
		*45-4863	*45-4893	*45-4901	*46-5022	*46-5027	*46-5037	*46-5042	*47-5116	*47-5124
		*47-5132	*48-5204	*48-5212	*49-5370	*49-5376	*49-5384	*49-5390	*49-5396	*49-5402
		*50-5598	*50-5602	*50-5608	*50-5614	*50-5623	*50-5629	*50-5635	*50-5645	*51-5831
		*51-5836	*51-5841	*51-5847	*51-5850	*51-5856	*51-5862	*51-5865	*51-5871	*51-5874
		*52-5885	*52-6089	*52-6090	*52-6091	*52-6092	*52-6093	*52-6094	*52-6095	*52-6096
		*52-6097	*52-6098	*52-6099	*52-6100	*53-6180	*53-6184	*53-6185	*53-6186	*53-6187
		*54-6251	*54-6255	*54-6257	*67-6449	75-7166	75-7169	75-7172	75-7180	75-7182
		75-7188	75-7192	75-7194	75-7197	75-7198	75-7199	75-7200	75-7204	75-7207
		75-7211	75-7212	75-7212	75-7222	75-7228	75-7231	75-7233	75-7237	75-7240
		75-7243	75-7254	75-7261	75-7287	75-7295	75-7298			
\$TMP5	001244	#20-1795	*25-1880	*27-1993	*28-2205	*30-2448	*30-2455	*30-2491	*30-2515	*30-2529
		*30-2544	*30-2560	*30-2575	*30-2584	*30-2600	*30-2636	*31-2831	*31-2836	*31-2842
		*32-3002	*32-3009	*33-3182	*33-3183	*33-3184	*33-3185	*33-3186	*33-3187	*33-3188
		*33-3189	*33-3190	*33-3191	*33-3192	*33-3193	*40-4102	*40-4108	*41-4242	*42-4366
		*42-4392	*43-4498	*43-4519	*44-4697	*44-4712	*45-4903	*46-5023	*46-5028	*47-5117
		*47-5125	*48-5205	*49-5372	*49-5378	*49-5386	*49-5392	*49-5397	*49-5403	*50-5609
		*50-5615	*50-5624	*50-5630	*50-5636	*50-5644	*51-5844	*51-5853	*51-5859	*51-5868
		*52-6089	*52-6090	*52-6091	*52-6092	*52-6093	*52-6094	*52-6095	*52-6096	*52-6097
		*52-6098	*52-6099	*52-6100	*53-6184	*53-6185	*53-6186	*53-6187	*54-6255	*54-6257
		*55-6269	55-6269	*55-6269	*55-6269	*55-6269	*56-6271	75-7166	75-7177	75-7178
		75-7185	75-7199	75-7199	75-7201	75-7202	75-7204	75-7211	75-7212	75-7214
		75-7215	75-7224	75-7225	75-7227	75-7229	75-7233	75-7240	75-7258	75-7259
		75-7288	75-7289	75-7295	75-7298					
\$TMP6	001246	#20-1795	*27-2123	*27-2129	*27-2147	*27-2153	*30-2449	*30-2460	*30-2465	*30-2496
		*30-2501	*30-2516	*30-2530	*30-2545	*30-2561	*30-2576	*30-2585	*30-2609	*30-2619
		*30-2627	*30-2646	*30-2656	*30-2663	*31-2837	*31-2843	*32-3007	*32-3012	*33-3269
		*40-4104	*40-4107	*40-4113	*40-4115	*40-4117	*41-4244	*41-4248	*41-4254	*41-4256
		*41-4258	*42-4343	*42-4347	*42-4351	*42-4355	*42-4359	*42-4364	*42-4394	*43-4477
		*43-4481	*43-4485	*43-4489	*43-4496	*43-4515	*43-4517	*44-4705	*44-4720	*45-4902
		*46-5029	*47-5118	*47-5126	*48-5206	*49-5369	*49-5375	*49-5381	*49-5383	*49-5389
		*49-5398	*49-5404	*50-5607	*50-5613	*50-5619	*50-5622	*50-5628	*50-5634	*50-5640
		*50-5643	*50-5649	*50-5652	*50-5655	*50-5658	*51-5842	*51-5848	*51-5851	*51-5857
		*51-5863	*51-5866	*51-5872	*51-5875	*52-6089	*52-6090	*52-6091	*52-6092	*52-6093
		*52-6094	*52-6095	*52-6096	*52-6097	*52-6098	*52-6099	*52-6100	*53-6184	*53-6185
		*53-6186	*53-6187	*54-6255	*54-6257	75-7178	75-7199	75-7201	75-7211	75-7214
		75-7224	75-7230	75-7233	75-7259	75-7288	75-7299			

SYMBOL	CROSS REFERENCE	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES	REFERENCES
SYMBOL	VALUE									
\$TMP7	001250	#20-1795	*27-1994	*27-2023	*27-2051	*27-2080	*30-2456	*30-2492	*30-2531	*30-2546
		*30-2552	*30-2577	*30-2599	*30-2637	*31-2838	*31-2844	*32-3008	*32-3013	*40-4103
		*40-4109	*41-4243	*41-4250	*42-4367	*42-4393	*43-4499	*43-4520	*44-4699	*44-4714
		75-7178	75-7202	75-7204	75-7215	75-7225	75-7234	75-7259	75-7289	
STN	= 000037	19-1777	#19-1777	24-1835	25-1835	#25-1835	25-1946	26-1946	#26-1946	26-1946
		27-1991	#27-1991	27-2195	28-2195	#28-2195	28-2275	29-2275	#29-2275	29-2275
		30-2333	#30-2333	30-2705	31-2705	#31-2705	31-2884	32-2884	#32-2884	32-2884
		33-3180	#33-3180	33-3358	34-3358	#34-3358	34-3463	35-3463	#35-3463	35-3463
		36-3621	#36-3621	36-3720	37-3720	#37-3720	37-3817	38-3817	#38-3817	38-3817
		39-3917	#39-3917	39-4020	40-4020	#40-4020	40-4160	41-4160	#41-4160	41-4160
		42-4301	#42-4301	42-4422	43-4422	#43-4422	43-4570	44-4570	#44-4570	44-4570
		45-4782	#45-4782	45-4924	46-4924	#46-4924	46-5074	47-5074	#47-5074	47-5074
		48-5170	#48-5170	48-5250	49-5250	#49-5250	49-5426	50-5426	#50-5426	50-5426
		51-5684	#51-5684	51-5896	52-5896	#52-5896	52-6120	53-6120	#53-6120	53-6120
		54-6204	#54-6204	54-6267						
STPB	001152	#20-1795	59-6277	59-6277	59-6277					
STPFLG	001157	#20-1795	59-6277	59-6277	59-6277					
STPS	001150	#20-1795	59-6277	59-6277	59-6277					
STRAP	041540	24-1811	#64-6287							
STRAP2	041562	#64-6287	64-6287							
STRP	= 000014	#64-6287	64-6287	64-6287	64-6287	64-6287	#64-6287	64-6287	64-6287	64-6287
		64-6287	#64-6287	64-6287	64-6287	64-6287	64-6287	#64-6287	64-6287	64-6287
		64-6287	64-6287	#64-6287	64-6287	64-6287	64-6287	64-6287	#64-6287	64-6287
		64-6287	64-6287	64-6287	#64-6287	64-6287	64-6287	64-6287	64-6287	#64-6287
		64-6287	64-6287	64-6287	64-6287	#64-6287	64-6287	64-6287	64-6287	64-6287
		#64-6287	64-6287	64-6287	64-6287	64-6287	#64-6287	64-6287	64-6287	64-6287
		64-6288	#64-6288							
STRPAD	041574	64-6287	#64-6287	64-6289						
STSTM	003606	#23-1808								
STSTM	001102	#20-1795	*55-6269	56-6271	56-6271	*56-6271	56-6271	56-6271	56-6271	56-6271
		57-6273	57-6273	57-6273	66-6302					
STYPBN	= *****	64-6287								
STYPDS	040400	#61-6281	64-6287	64-6287						
STYPE	037600	#59-6277	62-6283	64-6287	64-6287					
STYPEC	040012	59-6277	59-6277	59-6277	#59-6277	63-6285				
STYPEX	040142	59-6277	59-6277	59-6277	#59-6277					
STYPOC	040170	#60-6279	64-6287	64-6287						
STYPON	040204	60-6279	#60-6279	64-6287						
STYPOS	040144	#60-6279	64-6287							
SUNIT	001330	#20-1795								
SUNITM	003612	#23-1808								
SUSWR	001342	#20-1795								
SVECT1	001366	#20-1795								
SVECT2	001370	#20-1795								
SXOFF	= 000023	59-6277	59-6277							
SXON	= 000021	59-6277	59-6277	59-6277	63-6285					
SXTSTR	036642	#56-6271								
SSGET4	= 000001	#55-6269	#55-6269	55-6269						
SOFILL	040375	*60-6279	*60-6279	60-6279	#60-6279					
\$4OCAT	= *****	56-6271	57-6273							
.RSET	042622	64-6288	#70-6480							
.\$ASTA	= *****	62-6283	62-6283							

SYMBOL	CROSS REFERENCE	REFERENCES
SYMBOL	VALUE	#23-1808 23-1808
.\$X	= 003602	

