.REM 3

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DFFPA-A-D

PRODUCT NAME: PDP-11/34 FPP DIAGNOSTIC PART 1

DATE: DECEMBER 1976

AUTHOR: ANTHONY VEZZA

CONTENTS

1. ABSTRACT

```
                              --------

THE THREE PROGRAMS:

        DFFPA DFFPB DFFPC

ARE DESIGN TO DETECT AND REPORT LOGIC FAULTS IN  THE
PDP  11/34  FP11-A  FLOATING  POINT  PROCESSOR.  THE
DESIGN IS AN ATTEMPT TO REACH ALL ROM  STATES,  TAKE
ALL  BRANCH  MICRO  TESTS (BUT'S) AND VARIFY ALL THE
LOGIC.  THEY CONSIST OF 155 (OCT)  INDIVIDUAL  TESTS
SEQUENCED  TO  DETECT AND ATTEMPT TO IDENTIFY FAULTS
WITH A MINIMUM  HARDWARE  OR  SOFTWARE  LEVEL.   THE
TESTS  ARE PARTIONED INTO THREE STAND-ALONE PROGRAMS
DESCRIBED BELOW.

NOTE THAT ERROR REPORTS IN THESE PROGRAMS ARE  BASED
UPON THE KNOWLEDGE THAT ALL PREVIOUS TESTS HAVE BEEN
RUN AND IN MOST CASE THAT THERE  IS  ONLY  A  SINGLE
POINT FAULT IN THE FP11-A.  IF THE PROGRAMS OR TESTS
ARE NOT RUN IN ORDER THEN ERROR MESSAGES MAY NOT  BE
ACCURATE.


A.      DFFPA

DFFPA 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.      DFFPB

DFFPB TESTS:

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

C.      DFFPC

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

2.2 STORAGE

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

2.3 PRELIMINARY PROGRAMS

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

3. LOADING PROCEDURE

THE PROGRAMS WILL BE SUPPLIED ON THE 11.34 DIAGNOSTIC MEDIA. REFER TO THE XXDP OPERATING MANUAL FOR FURTHER INFORMATION.

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 DFFPA.
SW<7>=1....	200	DESELECT CORRECT INTERRUPT TEST IN PROGRAM DFFPB. NOTE THAT THIS TEST WILL AUTOMATICALLY BE DESELECTED BY THE ABSENCE OF THE SPECIAL TEST EQUIPMENT DESIGNED TO CONDUCT THIS TEST. IF THIS EQUIPMENT IS NOT INSTALLED THERE IS NO NEED TO DESELECT THIS TEST. THIS APPLIES ONLY TO PROGRAM DFFPB!

6. ERRORS

6.1 SUMMARIES

IN PROGRAM DFFPA 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 IS 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 (I AND II IN PROGRAM DFFPA BOTH SWITCHES 13 AND 7 MUST = 1.

6.3 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.

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 DFFPB 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)=0 THIS TEST WILL BE RUN. IF SW(7)=1 THIS TEST WILL BE DESELECTED.

8.7 ACT, APT AND XXDP COMPATIBILTY

THESE PROGRAMS ARE FULLY COMPATIBLE WITH:

APT
ACT
XXDP MONITOR AND CHAIN PROGRAMS.

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 DMD AND SMD ARE USED. NOTE THAT A MASK MUST BE USED BECAUSE SOME OF THE FPS BITS CANNOT BE SET.

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

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

TEST 2 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.

TEST 4 ILLEGAL FPP OP CODES AND STST TEST

THIS IS A TEST OF THE FPP OPERATION CODES:

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),AC0
```

AND THE INSTRUCTION:

```
STD     AC0,(R0)
```

MOST OF THE FAILURES ARE ISOLATED TO THE SRC OR DST FLOWS. NOTE THAT THE INTEGRITY OF AC0 HAS NOT BEEN ASSURED. THIS MEANS THAT IN SOME CASES IT WILL BE IMPOSSIBLE TO ISOLATE CERTAIN DATA PATTERN FAILURES TO EITHER THE FLOWS OR THIS ACCUMULATOR.

TEST 7 FSRC MODE 0 TEST

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

TEST 10 FDST MODE 0 TEST

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

TEST 11 ACCUMULATORS DATA PATTERNS TEST

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 P
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 INTERMITTANT THEN THIS PROCEDURE WILL

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

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

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

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

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

BITS	AM2901 CHIP NUMBER
A15,A14,A13,A12	E61
A11,A10,A9,A8	E62
A7,A6,A5,A4	E90
A3,A2,A1,A0	E81
B15,B14,B13,B12	E86
B11,B10,B9,B8	E85
B7,B6,B5,B4	E83
B3,B2,B1,B0	E88
C15,C14,C13,C12	E79
C11,C10,C9,C8	E84
C7,C6,C5,C4	E89
C3,C2,C1,C0	E87
D15,D14,D13,D12	E78
D11,D10,D9,D8	E77
D7,D6,D5,D4	E82
D3,D2,D1,D0	E80

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.

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

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

THEN THE ABOVE TABLE SHOULD DIRECTLY IDENTIFY THE FAILING CHIP.

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

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

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

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

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

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

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

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

TEST 12         FPP ACCUMULATORS DUAL ADDRESS TEST
-------         ----------------------------------

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

TEST 13         FSRC MODE 0 WITH ILLEGAL ACCUMULATOR TEST
-------         -----------------------------------------

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


TEST 14         FSRC MODE 2 TEST
-------         ----------------

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

TEST 15         FSRC MODE 4 TEST
-------         ----------------

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

TEST 16         FSRC MODE 2, WITH FD=0, TEST
-------         ----------------------------

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

TEST 17         FSRC MODE 2 WITH GR7, IMMEDIATE MODE, TEST
-------         ------------------------------------------

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

TEST 20         FSRC MODE 3 TEST
-------         ----------------

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

TEST 21         FSRC MODE 5 TEST
-------         ----------------

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

TEST 22         FSRC MODE 6 TEST
```

THIS IS A TEST OF FSRC MODE 6, INDEX MODE

TEST 23 FSRC MODE 7 TEST

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

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

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

0
+NUM
-NUM
-0

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

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

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

TEST 26 ADDD AND SUB WITH FSRC=0

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

TEST 27 SUBD WITH AC=0 TEST

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

TEST 30 ADDD WITH AC=0 TEST

POSITIVE AND NEGATIVE FSRC'S ARE TRIED.

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

THIS IS A TEST OF THE ADD INSTRUCTION WITH THE OPERANDS HAVING EQUAL EXPONENTS. THE (BUF 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 USED FOR FLOATING (F FOR DOUBLE) USED IS CHECKED. THEN SIMPLE AND WORST CASE ALIGNMENT SITUATIONS ARE TRIED. NOTE E(AC) IS LESS THEN E(FSRC)

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 NEGATVE OPRANDS TEST

THIS IS A TEST OF THE ADDD INSTRUCTION WITH NEGATIVE OPERANDS. EVERY COMBINATION OF OPERAND 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.

```
                        $
000213                  MNUMBER=213
000001                  PROGNUM=1


                                .LIST   ME
                                .NLIST  MD,MC,CND



                                .ENABL  ABS


                        .TITLE  MAINDEC-11-DFFPA-A      PDP 11/34 FPP DIAGNOSTIC PART 1
                        ;*COPYRIGHT (C) SEP 1976
                        ;*DIGITAL EQUIPMENT CORP.
                        ;*MAYNARD, MASS. 01754
                        ;*
                        ;*PROGRAM BY ANTHONY S. VEZZA
                        ;*
                        ;*THIS PROGRAM WAS ASSEMBLED USING THE PDP-11 MAINDEC SYSMAC
                        ;*PACKAGE (MAINDEC-11-DZQAC-C2), SEPT 14, 1976.
                        ;*
000001                  $TN=1
160000                  $SWR=160000     ;;HALT ON ERROR, LOOP ON TEST, INHIBIT ERROR TYPOUT


000244                  FPVECT=244
177400                  $SWR=177400
```

```
        000200                  $SWRMSK=200 .
        000011                  TAB=11
        000015                  CRLF=15

                                .SBTTL  BASIC DEFINITIONS

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

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

                                ;*GENERAL PURPOSE REGISTER DEFINITIONS
        000000                  R0=     %0                      ;;GENERAL REGISTER
        000001                  R1=     %1                      ;;GENERAL REGISTER
        000002                  R2=     %2                      ;;GENERAL REGISTER
        000003                  R3=     %3                      ;;GENERAL REGISTER
        000004                  R4=     %4                      ;;GENERAL REGISTER
        000005                  R5=     %5                      ;;GENERAL REGISTER
        000006                  R6=     %6                      ;;GENERAL REGISTER
        000007                  R7=     %7                      ;;GENERAL REGISTER
        000006                  SP=     %6                      ;;STACK POINTER
        000007                  PC=     %7                      ;;PROGRAM COUNTER

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

                                ;*"SWITCH REGISTER" SWITCH DEFINITIONS
        100000                  SW15=   100000
        040000                  SW14=   40000
        020000                  SW13=   20000
        010000                  SW12=   10000
        004000                  SW11=   4000
        002000                  SW10=   2000
        001000                  SW09=   1000
        000400                  SW08=   400
        000200                  SW07=   200
        000100                  SW06=   100
```

```
MACRO-11-05FPA-A     PDP 11 34 FPP DIAGNOSTIC PART 1 MACY11 27-JUN-80  01-NOV-76  21:09  PAGE 18
05FPA1.P11    01-NOV-76 21:03          BASIC DEFINITIONS

                                SW05=   40
                                SW04=   20
                                SW03=   10
                                SW02=   4
                                SW01=   2
                                SW00=   1
                                .EQUIV  SW09,SW9
                                .EQUIV  SW08,SW8
                                .EQUIV  SW07,SW7
                                .EQUIV  SW06,SW6
                                .EQUIV  SW05,SW5
                                .EQUIV  SW04,SW4
                                .EQUIV  SW03,SW3
                                .EQUIV  SW02,SW2
                                .EQUIV  SW01,SW1
                                .EQUIV  SW00,SW0

                                ;*DATA BIT DEFINITIONS (BIT00 TO BIT15
                                BIT15=  100000
                                BIT14=  40000
                                BIT13=  20000
                                BIT12=  10000
                                BIT11=  4000
                                BIT10=  2000
                                BIT09=  1000
                                BIT08=  400
                                BIT07=  200
                                BIT06=  100
                                BIT05=  40
                                BIT04=  20
                                BIT03=  10
                                BIT02=  4
                                BIT01=  2
                                BIT00=  1
                                .EQUIV  BIT09,BIT9
                                .EQUIV  BIT08,BIT8
                                .EQUIV  BIT07,BIT7
                                .EQUIV  BIT06,BIT6
                                .EQUIV  BIT05,BIT5
                                .EQUIV  BIT04,BIT4
                                .EQUIV  BIT03,BIT3
                                .EQUIV  BIT02,BIT2
                                .EQUIV  BIT01,BIT1
                                .EQUIV  BIT00,BIT0

                                ;*BASIC "CPU" TRAP VECTOR ADDRESSES
                                ERRVEC= 4               ;;TIME OUT AND OTHER ERRORS
                                RESVEC= 10              ;;RESERVED AND ILLEGAL INSTRUCTIONS
                                TBITVEC=14              ;;"T" BIT
                                TRTVEC= 14              ;;TRACE TRAP
                                BPTVEC= 14              ;;BREAKPOINT TRAP (BPT)
                                IOTVEC= 20              ;;INPUT/OUTPUT TRAP (IOT) **SCOPE**
                                PWRVEC= 24              ;;POWER FAIL
                                EMTVEC= 30              ;;EMULATOR TRAP (EMT  **ERROR**
                                TRAPVEC=34              ;;"TRAP" TRAP
                                TKVEC=  60              ;;TTY KEYBOARD VECTOR
```

G02


```
                000064                  TPVEC=  64                  ;;TTY PRINTER VECTOR
                000240                  PIRQVEC=240                 ;;PROGRAM INTERRUPT REQUEST VECTOR
                                                .SBTTL  FPP REGISTER DEFINITIONS
                000000                  AC0     =%0
                000001                  AC1     =%1
                000002                  AC2     =%2
                000003                  AC3     =%3
                000004                  AC4     =%4
                000005                  AC5     =%5
                000006                  AC6     =%6
                000007                  AC7     =%7

                                        .SBTTL  TRAP CATCHER

                000000                          .=0
                                        ;*ALL UNUSED LOCATIONS FROM 4 - 776 CONTAIN A ".+2,HALT"
                                        ;*SEQUENCE TO CATCH ILLEGAL TRAPS AND INTERRUPTS
                                        ;*LOCATION 0 CONTAINS 0 TO CATCH IMPROPERLY LOADED VECTORS
                000174                          .=174
        000174  000000                  DISPREG: .WORD  0                   ;;SOFTWARE DISPLAY REGISTER
        000176  000000                  SWREG:   .WORD  0                   ;;SOFTWARE SWITCH REGISTER
                                        .SBTTL  STARTING ADDRESS(ES)
        000200  000137  003606                  JMP     @#START ;;JUMP TO STARTING ADDRESS OF PROGRAM
```

```
                                .SBTTL  COMMON TAGS

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

          001100                .=1100
001100                  $CMTAG:                                 ;;START OF COMMON TAGS
001100    000000                .WORD   0
001102       000        $TSTNM: .BYTE   0                       ;;CONTAINS THE TEST NUMBER
001103       000        $ERFLG: .BYTE   0                       ;;CONTAINS ERROR FLAG
001104    000000        $ICNT:  .WORD   0                       ;;CONTAINS SUBTEST ITERATION COUNT
001106    000000        $LPADR: .WORD   0                       ;;CONTAINS SCOPE LOOP ADDRESS
001110    000000        $LPERR: .WORD   0                       ;;CONTAINS SCOPE RETURN FOR ERRORS
001112    000000        $ERTTL: .WORD   0                       ;;CONTAINS TOTAL ERRORS DETECTED
001114       000        $ITEMB: .BYTE   0                       ;;CONTAINS ITEM CONTROL BYTE
001115       001        $ERMAX: .BYTE   1                       ;;CONTAINS MAX. ERRORS PER TEST
001116    000000        $ERRPC: .WORD   0                       ;;CONTAINS PC OF LAST ERROR INSTRUCTION
001120    000000        $GDADR: .WORD   0                       ;;CONTAINS ADDRESS OF 'GOOD' DATA
001122    000000        $BDADR: .WORD   0                       ;;CONTAINS ADDRESS OF 'BAD' DATA
001124    000000        $GDDAT: .WORD   0                       ;;CONTAINS 'GOOD' DATA
001126    000000        $BDDAT: .WORD   0                       ;;CONTAINS 'BAD' DATA
001130    000000                .WORD   0                       ;;RESERVED--NOT TO BE USED
001132    000000                .WORD   0
001134       000        $AUTOB: .BYTE   0                       ;;AUTOMATIC MODE INDICATOR
001135       000        $INTAG: .BYTE   0                       ;;INTERRUPT MODE INDICATOR
001136    000000                .WORD   0
001140    177570        SWR:    .WORD   DSWR                    ;;ADDRESS OF SWITCH REGISTER
001142    177570        DISPLAY: .WORD  DDISP                   ;;ADDRESS OF DISPLAY REGISTER
001144    177560        $TKS:   177560                          ;;TTY KBD STATUS
001146    177562        $TKB:   177562                          ;;TTY KBD BUFFER
001150    177564        $TPS:   177564                          ;;TTY PRINTER STATUS REG. ADDRESS
001152    177566        $TPB:   177566                          ;;TTY PRINTER BUFFER REG. ADDRESS
001154       000        $NULL:  .BYTE   0                       ;;CONTAINS NULL CHARACTER FOR FILLS
001155       002        $FILLS: .BYTE   2                       ;;CONTAINS # OF FILLER CHARACTERS REQUIRED
001156       012        $FILLC: .BYTE   12                      ;;INSERT FILL CHARS. AFTER A "LINE FEED"
001157       000        $TPFLG: .BYTE   0                       ;;"TERMINAL AVAILABLE" FLAG (BIT<07>=0=YES
001160    000000        $REGAD: .WORD   0                       ;;CONTAINS THE ADDRESS FROM
                                                                ;;WHICH ($REG0) WAS OBTAINED
001162    000000        $REG0:  .WORD   0                       ;;CONTAINS (($REGAD)+0)
001164    000000        $REG1:  .WORD   0                       ;;CONTAINS (($REGAD)+2)
001166    000000        $REG2:  .WORD   0                       ;;CONTAINS (($REGAD)+4)
001170    000000        $REG3:  .WORD   0                       ;;CONTAINS (($REGAD)+6)
001172    000000        $REG4:  .WORD   0                       ;;CONTAINS (($REGAD)+10)
001174    000000        $REG5:  .WORD   0                       ;;CONTAINS (($REGAD)+12)
001176    000000        $REG6:  .WORD   0                       ;;CONTAINS (($REGAD)+14)
001200    000000        $REG7:  .WORD   0                       ;;CONTAINS (($REGAD)+16)
001202    000000        $REG10: .WORD   0                       ;;CONTAINS (($REGAD)+20)
001204    000000        $REG11: .WORD   0                       ;;CONTAINS (($REGAD)+22)
001206    000000        $REG12: .WORD   0                       ;;CONTAINS (($REGAD)+24)
001210    000000        $REG13: .WORD   0                       ;;CONTAINS (($REGAD)+26)
001212    000000        $REG14: .WORD   0                       ;;CONTAINS (($REGAD)+30)
001214    000000        $REG15: .WORD   0                       ;;CONTAINS (($REGAD)+32)
001216    000000        $REG16: .WORD   0                       ;;CONTAINS (($REGAD)+34)
001220    000000        $REG17: .WORD   0                       ;;CONTAINS (($REGAD)+36)
001222    000000        $REG20: .WORD   0                       ;;CONTAINS (($REGAD)+40)
```

```
                                $REG21: .WORD   0               ;;CONTAINS (($REGAD)+42)
                                $REG22: .WORD   0               ;;CONTAINS (($REGAD)+44)
                                $REG23: .WORD   0               ;;CONTAINS (($REGAD)+46)
                                $TMP0:  .WORD   0               ;;USER DEFINED
                                $TMP1:  .WORD   0               ;;USER DEFINED
                                $TMP2:  .WORD   0               ;;USER DEFINED
                                $TMP3:  .WORD   0               ;;USER DEFINED
                                $TMP4:  .WORD   0               ;;USER DEFINED
                                $TMP5:  .WORD   0               ;;USER DEFINED
                                $TMP6:  .WORD   0               ;;USER DEFINED
                                $TMP7:  .WORD   0               ;;USER DEFINED
                                $TMP10: .WORD   0               ;;USER DEFINED
                                $TMP11: .WORD   0               ;;USER DEFINED
                                $TMP12: .WORD   0               ;;USER DEFINED
                                $TMP13: .WORD   0               ;;USER DEFINED
                                $TMP14: .WORD   0               ;;USER DEFINED
                                $TMP15: .WORD   0               ;;USER DEFINED
                                $TMP16: .WORD   0               ;;USER DEFINED
                                $TMP17: .WORD   0               ;;USER DEFINED
                                $TMP20: .WORD   0               ;;USER DEFINED
                                $TMP21: .WORD   0               ;;USER DEFINED
                                $TMP22: .WORD   0               ;;USER DEFINED
                                $TMP23: .WORD   0               ;;USER DEFINED
                                $TIMES: 0                       ;;MAX. NUMBER OF ITERATIONS
                                $ESCAPE:0                       ;;ESCAPE ON ERROR ADDRESS
                                $BELL:  .ASCIZ  <207><377><377> ;;CODE FOR BELL
                                $QUES:  .ASCII  /?/             ;;QUESTION MARK
                                $CRLF:  .ASCII  <15>            ;;CARRIAGE RETURN
                                $LF:    .ASCIZ  <12>            ;;LINE FEED
                                ;;**************************************************************
                                .SBTTL  APT MAILBOX-ETABLE

                                ;;**************************************************************
                                .EVEN
                                $MAIL:                          ;;APT MAILBOX
                                $MSGTY: .WORD   AMSGTY          ;;MESSAGE TYPE CODE
                                $FATAL: .WORD   AFATAL          ;;FATAL ERROR NUMBER
                                $TESTN: .WORD   ATESTN          ;;TEST NUMBER
                                $PASS:  .WORD   APASS           ;;PASS COUNT
                                $DEVCT: .WORD   ADEVCT          ;;DEVICE COUNT
                                $UNIT:  .WORD   AUNIT           ;;I/O UNIT NUMBER
                                $MSGAD: .WORD   AMSGAD          ;;MESSAGE ADDRESS
                                $MSGLG: .WORD   AMSGLG          ;;MESSAGE LENGTH
                                $ETABLE:                        ;;APT ENVIRONMENT TABLE
                                $ENV:   .BYTE   AENV            ;;ENVIRONMENT BYTE
                                $ENVM:  .BYTE   AENVM           ;;ENVIRONMENT MODE BITS
                                $SWREG: .WORD   ASWREG          ;;APT SWITCH REGISTER
                                $USWR:  .WORD   AUSWR           ;;USER SWITCHES
                                $CPUOP: .WORD   ACPUOP          ;;CPU TYPE,OPTIONS
                                ;*                              BITS 15-11=CPU TYPE
                                ;*                                      11/04=01,11/05=02,11/20=03,11/40=04,11/45=05
                                ;*                                      11/70=06,PDQ=07,Q=10
                                ;*                              BIT 10=REAL TIME CLOCK
                                ;*                              BIT  9=FLOATING POINT PROCESSOR
                                ;*                              BIT  8=MEMORY MANAGEMENT
                                $MAMS1: .BYTE   AMAMS1          ;;HIGH ADDRESS,M.S. BYTE
```

J02


```
001347  000           $MTYP1:  .BYTE   AMTYP1    ;;MEM. TYPE,BLK#1
                      ;*                           MEM.TYPE BYTE  --  (HIGH BYTE
                      ;*                                   900 NSEC CORE=001
                      ;*                                   300 NSEC BIPOLAR=002
                      ;*                                   500 NSEC MOS=003
001350  000000        $MADR1:  .WORD   AMADR1    ;;HIGH ADDRESS,BLK#1
                      ;*                             MEM.LAST ADDR.=3 BYTES,THIS WORD AND LOW OF ...
001352     000        $MAMS2:  .BYTE   AMAMS2    ;;HIGH ADDRESS,M.S. BYTE
001353     000        $MTYP2:  .BYTE   AMTYP2    ;;MEM.TYPE,BLK#2
001354  000000        $MADR2:  .WORD   AMADR2    ;;MEM.LAST ADDRESS,BLK#2
001356     000        $MAMS3:  .BYTE   AMAMS3    ;;HIGH ADDRESS,M.S.BYTE
001357     000        $MTYP3:  .BYTE   AMTYP3    ;;MEM.TYPE,BLK#3
001360  000000        $MADR3:  .WORD   AMADR3    ;;MEM.LAST ADDRESS,BLK#3
001362     000        $MAMS4:  .BYTE   AMAMS4    ;;HIGH ADDRESS,M.S.BYTE
001363     000        $MTYP4:  .BYTE   AMTYP4    ;;MEM.TYPE,BLK#4
001364  000000        $MADR4:  .WORD   AMADR4    ;;MEM.LAST ADDRESS,BLK#4
001366  000000        $VECT1:  .WORD   AVECT1    ;;INTERRUPT VECTOR#1,BUS PRIORITY#1
001370  000000        $VECT2:  .WORD   AVECT2    ;;INTERRUPT VECTOR#2BUS PRIORITY#2
001372  000000        $BASE:   .WORD   ABASE     ;;BASE ADDRESS OF EQUIPMENT UNDER TEST
001374  000000        $DEVM:   .WORD   ADEVM     ;;DEVICE MAP
001376  000000        $CDW1:   .WORD   ACDW1     ;;CONTROLLER DESCRIPTION WORD#1
001400  000000        $CDW2:   .WORD   ACDW2     ;;CONTROLLER DESCRIPTION WORD#2
001402  000000        $DDW0:   .WORD   ADDW0     ;;DEVICE DESCRIPTOR WORD#0
001404  000000        $DDW1:   .WORD   ADDW1     ;;DEVICE DESCRIPTOR WORD#1
001406  000000        $DDW2:   .WORD   ADDW2     ;;DEVICE DESCRIPTOR WORD#2
001410  000000        $DDW3:   .WORD   ADDW3     ;;DEVICE DESCRIPTOR WORD#3
001412  000000        $DDW4:   .WORD   ADDW4     ;;DEVICE DESCRIPTOR WORD#4
001414  000000        $DDW5:   .WORD   ADDW5     ;;DEVICE DESCRIPTOR WORD#5
001416  000000        $DDW6:   .WORD   ADDW6     ;;DEVICE DESCRIPTOR WORD#6
001420  000000        $DDW7:   .WORD   ADDW7     ;;DEVICE DESCRIPTOR WORD#7
001422  000000        $DDW8:   .WORD   ADDW8     ;;DEVICE DESCRIPTOR WORD#8
001424  000000        $DDW9:   .WORD   ADDW9     ;;DEVICE DESCRIPTOR WORD#9
001426  000000        $DDW10:  .WORD   ADDW10    ;;DEVICE DESCRIPTOR WORD#10
001430  000000        $DDW11:  .WORD   ADDW11    ;;DEVICE DESCRIPTOR WORD#11
001432  000000        $DDW12:  .WORD   ADDW12    ;;DEVICE DESCRIPTOR WORD#12
001434  000000        $DDW13:  .WORD   ADDW13    ;;DEVICE DESCRIPTOR WORD#13
001436  000000        $DDW14:  .WORD   ADDW14    ;;DEVICE DESCRIPTOR WORD#14
001440  000000        $DDW15:  .WORD   ADDW15    ;;DEVICE DESCRIPTOR WORD#15

001442                $ETEND:
```

```
                                      .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

 001442                               $ERRTB:
                                      ;ITEM 1
 001442  043025  063250  067664               .WORD   EM1,DH1,DT1,DF1
 001450  066611
                                      ;ITEM 2
 001452  043062  063340  067706               .WORD   EM2,DH2,DT2,DF2
 001460  066621
                                      ;ITEM 3
 001462  043126  063433  067730               .WORD   EM3,DH3,DT3,DF3
 001470  066631
                                      ;ITEM 4
 001472  043173  063524  067730               .WORD   EM4,DH4,DT4,DF4
 001500  066631
                                      ;ITEM 5
 001502  043233  063620  067752               .WORD   EM5,DH5,DT5,DF5
 001510  066641
                                      ;ITEM 6
 001512  043267  063620  070004               .WORD   EM6,DH6,DT6,DF6
 001520  066641
                                      ;ITEM 7
 001522  043321  063620  070004               .WORD   EM7,DH7,DT7,DF7
 001530  066641
                                      ;ITEM 10
 001532  043233  063620  070004               .WORD   EM10,DH10,DT10,DF10
 001540  066641
                                      ;ITEM 11
 001542  043354  063620  070004               .WORD   EM11,DH11,DT11,DF11
 001550  066641
                                      ;ITEM 12
 001552  000000  000000  070026               .WORD   EM12,DH12,DT12,DF12
 001560  066655
                                      ;ITEM 13
 001562  000000  000000  070114               .WORD   EM13,DH13,DT13,DF13
 001570  066707
                                      ;ITEM 14
 001572  043435  063620  070004               .WORD   EM14,DH14,DT14,DF14
 001600  066641
                                      ;ITEM 15
 001602  043560  063620  070004               .WORD   EM15,DH15,DT15,DF15
 001610  066641
                                      ;ITEM 16
 001612  043703  063660  070146               .WORD   EM16,DH16,DT16,DF16
```

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

```
1298  002242  045724  000000  070666          .WORD   EM41,DH41,DT41,DF41
1299  002250  067204
1300                                  ;ITEM 42
1301  002252  045761  064703  070720          .WORD   EM42,DH42,DT42,DF42
1302  002260  067220
1303                                  ;ITEM 43
1304  002262  046102  064703  070720          .WORD   EM43,DH43,DT43,DF43
1305  002270  067220
1306                                  ;ITEM 44
1307  002272  046223  000000  070776          .WORD   EM44,DH44,DT44,DF44
1308  002300  067246
1309                                  ;ITEM 45
1310  002302  046223  065005  071046          .WORD   EM45,DH45,DT45,DF45
1311  002310  067271
1312                                  ;ITEM 46
1313  002312  046266  065024  071122          .WORD   EM46,DH46,DT46,DF46
1314  002320  067316
1315                                  ;ITEM 47
1316  002322  046344  065005  071210          .WORD   EM47,DH47,DT47,DF47
1317  002330  067350
1318                                  ;ITEM 50
1319  002332  046462  065050  070614          .WORD   EM50,DH50,DT50,DF50
1320  002340  067365
1321                                  ;ITEM 51
1322  002342  046560  065050  071242          .WORD   EM51,DH51,DT51,DF51
1323  002350  067411
1324                                  ;ITEM 52
1325  002352  046621  063620  071210          .WORD   EM52.DH52,DT52,DF52
1326  002360  067350
1327                                  ;ITEM 53
1328  002362  046742  064512  071300          .WORD   EM53.DH53,DT53,DF53
1329  002370  067427
1330                                  ;ITEM 54
1331  002372  047137  065122  071320          .WORD   EM54,DH54,DT54,DF54
1332  002400  067436
1333                                  ;ITEM 55
1334  002402  047203  063620  071210          .WORD   EM55,DH55,DT55,DF55
1335  002410  067350
1336                                  ;ITEM 56
1337  002412  047324  064512  071300          .WORD   EM56,DH56,DT56,DF56
1338  002420  067427
1339                                  ;ITEM 57
1340  002422  047521  065122  071320          .WORD   EM57,DH57,DT57,DF57
1341  002430  067436
1342                                  ;ITEM 60
1343  002432  047565  064512  071300          .WORD   EM60,DH60.DT60.DF60
1344  002440  067427
1345                                  ;ITEM 61
1346  002442  047762  065122  071320          .WORD   EM61,DH61,DT61,DF61
1347  002450  067436
1348                                  ;ITEM 62
1349  002452  050026  065122  071320          .WORD   EM62.DH62,DT62.DF62
1350  002460  067436
1351                                  ;ITEM 63
1352  002462  050220  065122  071320          .WORD   EM63,DH63,DT63.DF63
1353  002470  067436
```

```
1354                                            ;ITEM 64
1355  002272  050412  065232  071356                .WORD   EM64,DH64,DT64,DF64
1356  002300  067454
1357                                            ;ITEM 65
1358  002302  050412  065163  071356                .WORD   EM65,DH65,DT65,DF65
1359  002310  067454
1360                                            ;ITEM 66
1361  002312  050546  065122  071320                .WORD   EM66,DH66,DT66,DF66
1362  002320  067436
1363                                            ;ITEM 67
1364  002322  050611  063620  070210                .WORD   EM67,DH67,DT67,DF67
1365  002330  066742
1366                                            ;ITEM 70
1367  002332  051042  063620  071376                .WORD   EM70,DH70,DT70,DF70
1368  002340  067463
1369                                            ;ITEM 71
1370  002342  051165  064450  071376                .WORD   EM71,DH71,DT71,DF71
1371  002350  067463
1372                                            ;ITEM 72
1373  002352  051267  064512  071444                .WORD   EM72,DH72,DT72,DF72
1374  002360  067505
1375                                            ;ITEM 73
1376  002362  051343  065122  071320                .WORD   EM73,DH73,DT73,DF73
1377  002370  067436
1378                                            ;ITEM 74
1379  0023    051403  063620  070210                .WORD   EM74,DH74,DT74,DF74
1380  002400  066742
1381                                            ;ITEM 75
1382  002402  051634  063620  071376                .WORD   EM75,DH75,DT75,DF75
1383  002410  067463
1384                                            ;ITEM 76
1385  002412  051757  064450  071376                .WORD   EM76,DH76,DT76,DF76
1386  002420  067463
1387                                            ;ITEM 77
1388  002422  052061  064512  071444                .WORD   EM77,DH77,DT77,DF77
1389  002430  067505
1390                                            ;ITEM 100
1391  002432  052135  065122  071320                .WORD   EM100,DH100,DT100,DF100
1392  002440  067436
1393                                            ;ITEM 101
1394  002442  052175  063620  071376                .WORD   EM101,DH101,DT101,DF101
1395  002450  067463
1396                                            ;ITEM 102
1397  002452  052321  064512  071376                .WORD   EM102,DH102,DT102,DF102
1398  002460  067505
1399                                            ;ITEM 103
1400  002462  052373  064450  071376                .WORD   EM103,DH103,DT103,DF103
1401  002470  067463
1402                                            ;ITEM 104
1403  002472  052476  065122  071320                .WORD   EM104,DH104,DT104,DF104
1404  002500  067436
1405                                            ;ITEM 105
1406  002502  052537  063620  071376                .WORD   EM105,DH105,DT105,DF105
1407  002510  067463
1408                                            ;ITEM 106
1409  002512  052664  064512  071444                .WORD   EM106,DH106,DT106,DF106
```

```
        002520  067505
                                        ;ITEM 107
        002522  052737  064450  071376          .WORD   EM107,DH107,DT107,DF107
        002530  067463
                                        ;ITEM 110
        002532  052043  065122  071320          .WORD   EM110,DH110,DT110,DF110
        002540  067436
                                        ;ITEM 111
        002542  053105  064450  071464          .WORD   EM111,DH111,DT111,DF111
        002550  067514
                                        ;ITEM 112
        002552  053105  065320  071464          .WORD   EM112,DH112,DT112,DF112
        002560  067514
                                        ;ITEM 113
        002562  053207  064450  071464          .WORD   EM113,DH113,DT113,DF113
        002570  067514
                                        ;ITEM 114
        002572  053207  065320  071464          .WORD   EM114,DH114,DT114,DF114
        002600  067514
                                        ;ITEM 115
        002602  053105  065537  071464          .WORD   EM115,DH115,DT115,DF115
        002610  067514
                                        ;ITEM 116
        002612  053207  065537  071464          .WORD   EM116,DH116,DT116,DF116
        002620  067514
                                        ;ITEM 117
        002622  053311  063740  067730          .WORD   EM117,DH117,DT117,DF117
        002630  066631
                                        ;ITEM 120
        002632  053445  066023  067730          .WORD   EM120,DH120,DT120,DF120
        002640  066631
                                        ;ITEM 121
        002642  053601  063620  071210          .WORD   EM121,DH121,DT121,DF121
        002650  067350
                                        ;ITEM 122
        002652  053720  065050  070614          .WORD   EM122,DH122,DT122,DF122
        002660  067365
                                        ;ITEM 123
        002662  054017  065050  071242          .WORD   EM123,DH123,DT123,DF123
        002670  067411
                                        ;ITEM 124
        002672  054060  063740  071476          .WORD   EM124,DH124,DT124,DF124
        002700  067520
                                        ;ITEM 125
        002702  054153  063740  071476          .WORD   EM125,DH125,DT125,DF125
        002710  067520
                                        ;ITEM 126
        002712  054243  063620  071464          .WORD   EM126,DH126,DT126,DF126
        002720  067514
                                        ;ITEM 127
        002722  054452  065122  071464          .WORD   EM127,DH127,DT127,DF127
        002730  067514
                                        ;ITEM 130
        002732  054665  066023  067730          .WORD   EM130,DH130,DT130,DF130
        002740  066631
                                        ;ITEM 131
```

```
002742  054765  065122  071562            .WORD   EM131,DH131,DT131,DF131
002750  067551
                               ;ITEM 132
002752  055025  065122  071562            .WORD   EM132,DH132,DT132,DF132
002760  067551
                               ;ITEM 133
002762  055065  066113  071624            .WORD   EM133,DH133,DT133,DF133
002770  067571
                               ;ITEM 134
002772  055124  066113  071624            .WORD   EM134,DH134,DT134,DF134
003000  067571
                               ;ITEM 135
003002  055163  066113  071624            .WORD   EM135,DH135,DT135,DF135
003010  067571
                               ;ITEM 136
003012  055222  066113  071624            .WORD   EM136,DH136,DT136,DF136
003020  067571
                               ;ITEM 137
003022  055065  066223  071676            .WORD   EM137,DH137,DT137,DF137
003030  067615
                               ;ITEM 140
003032  055124  066223  071676            .WORD   EM140,DH140,DT140,DF140
003040  067615
                               ;ITEM 141
003042  055163  066223  071676            .WORD   EM141,DH141,DT141,DF141
003050  067615
                               ;ITEM 142
003052  055222  066223  071676            .WORD   EM142,DH142,DT142,DF142
003060  067615
                               ;ITEM 143
003062  055261  066113  071624            .WORD   EM143,DH143,DT143,DF143
003070  067571
                               ;ITEM 144
003072  055314  066113  071624            .WORD   EM144,DH144,DT144,DF144
003100  067571
                               ;ITEM 145
003102  055261  066223  071676            .WORD   EM145,DH145,DT145,DF145
003110  067615
                               ;ITEM 146
003112  055314  066223  071676            .WORD   EM146,DH146,DT146,DF146
003120  067615
                               ;ITEM 147
003122  055347  065122  071624            .WORD   EM147,DH147,DT147,DF147
003130  067571
                               ;ITEM 150
003132  055347  066413  071624            .WORD   EM150,DH150,DT150,DF150
003140  067571
                               ;ITEM 151
003142  055347  066223  071676            .WORD   EM151,DH151,DT151,DF151
003150  067615
                               ;ITEM 152
003152  055401  066113  071624            .WORD   EM152,DH152,DT152,DF152
003160  067571
                               ;ITEM 153
003162  055401  066223  071676            .WORD   EM153,DH153,DT153,DF153
003170  067615
```

```
                                                ;ITEM 154
003172  055433  066504  071716                      .WORD   EM154,DH154,DT154,DF154
003200  067625
                                                ;ITEM 155
003202  055665  066504  071716                      .WORD   EM155,DH155,DT155,DF155
003210  067625
                                                ;ITEM 156
003212  056120  065122  071624                      .WORD   EM156,DH156,DT156,DF156
003220  067571
                                                ;ITEM 157
003222  056335  065122  071624                      .WORD   EM157,DH157,DT157,DF157
003230  067571
                                                ;ITEM 160
003232  056554  065122  071624                      .WORD   EM160,DH160,DT160,DF160
003240  067571
                                                ;ITEM 161
003242  056761  065122  071624                      .WORD   EM161,DH161,DT161,DF161
003250  067631
                                                ;ITEM 162
003252  057166  065122  071624                      .WORD   EM162,DH162,DT162,DF162
003260  067571
                                                ;ITEM 163
003262  057233  065122  071624                      .WORD   EM163,DH163,DT163,DF163
003270  067631
                                                ;ITEM 164
003272  057300  063740  067730                      .WORD   EM164,DH164,DT164,DF164
003300  066631
                                                ;ITEM 165
003302  057345  063740  067730                      .WORD   EM165,DH165,DT165,DF165
003310  066631
                                                ;ITEM 166
003312  057412  065122  071624                      .WORD   EM166,DH166,DT166,DF166
003320  067571
                                                ;ITEM 167
003322  057522  065122  071624                      .WORD   EM167,DH167,DT167,DF167
003330  067571
                                                ;ITEM 170
003332  057761  065122  071624                      .WORD   EM170,DH170,DT170,DF170
003340  067631
                                                ;ITEM 171
003342  060071  065122  071624                      .WORD   EM171,DH171,DT171,DF171
003350  067631
                                                ;ITEM 172
003352  060330  065122  071624                      .WORD   EM172,DH172,DT172,DF172
003360  067571
                                                ;ITEM 173
003362  060567  065122  071624                      .WORD   EM173,DH173,DT173,DF173
003370  067571
                                                ;ITEM 174
003372  061026  065122  071624                      .WORD   EM174,DH174,DT174,DF174
003400  067631
                                                ;ITEM 175
003402  061265  065122  071624                      .WORD   EM175,DH175,DT175,DF175
003410  067631
                                                ;ITEM 176
003412  061524  065122  071624                      .WORD   EM176,DH176,DT176,DF176
```

ERROR POINTER TABLE

```
003420  067571
                                ;ITEM 177
003422  061661  065122  071624          .WORD   EM177,DH177,DT177,DF177
003430  067571
                                ;ITEM 200
003432  062016  065122  071624          .WORD   EM200,DH200,DT200,DF200
003440  067571
                                ;ITEM 201
003442  062153  065122  071624          .WORD   EM201,DH201,DT201,DF201
003450  067571
                                ;ITEM 202
003452  062310  065122  071624          .WORD   EM202,DH202,DT202,DF202
003460  067571
                                ;ITEM 203
003462  062445  065122  071624          .WORD   EM203,DH203,DT203,DF203
003470  067571
                                ;ITEM 204
003472  062602  065122  071624          .WORD   EM204,DH204,DT204,DF204
003500  067571
                                ;ITEM 205
003502  062737  063740  067730          .WORD   EM205,DH205,DT205,DF205
003510  066631
                                ;ITEM 206
003512  063004  065122  071624          .WORD   EM206,DH206,DT206,DF206
003520  067571
                                ;ITEM 207
003522  063051  065122  071624          .WORD   EM207,DH207,DT207,DF207
003530  067571
                                ;ITEM 210
003532  063173  065122  071624          .WORD   EM210,DH210,DT210,DF210
003540  067571
                                ;ITEM 211
003542  043233  066544  071730          .WORD   EM211,DH211,DT211,DF211
003550  067655
                                ;ITEM 212
003552  043267  063620  071746          .WORD   EM212,DH212,DT212,DF212
003560  067655
                                ;ITEM 213
003562  043321  063620  071746          .WORD   EM213,DH213,DT213,DF213
003570  067655


                                .SBTTL  ACT11 HOOKS

                                ;;*****************************************************************
                                ;HOOKS REQUIRED BY ACT11
        003572                          $SVPC=.                 ;SAVE PC
        000046                          .=46
000046  034770                          $ENDAD                  ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .SEOP
        000052                          .=52
000052  000000                          .WORD   0               ;;2)SET LOC.52 TO ZERO
        003572                          .=$SVPC                 ;; RESTORE PC
                                .SBTTL  APT PARAMETER BLOCK

                                ;;*****************************************************************
                                ;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
```

```
                                        ;;*****************************************************************
        003572                                  .$X=.      ;;SAVE CURRENT LOCATION
        000024                                  .=24       ;;SET POWER FAIL TO POINT TO START OF PROGRAM
000024  000200                                  200        ;;FOR APT START UP
        000044                                  .=44       ;;POINT TO APT INDIRECT ADDRESS PNTR.
000044  003572                                  $APTHDR    ;;POINT TO APT HEADER BLOCK
        003572                                  .=.$X      ;;RESET LOCATION COUNTER
                                        ;;*****************************************************************
                                        ;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
                                        ;INTERFACE SPEC.

003572                                  $APTHD:
003572  000000                          $HIBTS: .WORD   0               ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
003574  001316                          $MBADR: .WORD   $MAIL           ;;ADDRESS OF APT MAILBOX (BITS 0-15)
003576  000010                          $TSTM:  .WORD   10              ;;RUN TIM OF LONGEST TEST
003600  000040                          $PASTM: .WORD   40              ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
003602  000000                          $UNITM: .WORD   0               ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
003604  000052                                  .WORD   $ETEND-$MAIL/2 ;;LENGTH MAILBOX-ETABLE(WORDS)


003606                                  START:
                                        .SBTTL  INITIALIZE THE COMMON TAGS
                                        ;;CLEAR THE COMMON TAGS ($CMTAG) AREA
003606  012706  001100                          MOV     #$CMTAG,R6              ;;FIRST LOCATION TO BE CLEARED
003612  005026                                  CLR     (R6)+                   ;;CLEAR MEMORY LOCATION
003614  022706  001140                          CMP     #SWR,R6 ;;DONE?
003620  001374                                  BNE     .-6                     ;;LOOP BACK IF NO
003622  012706  001100                          MOV     #STACK,SP               ;;SETUP THE STACK POINTER
                                        ;;INITIALIZE A FEW VECTORS
003626  012737  035050  000020                  MOV     #$SCOPE,@#IOTVEC ;;IOT VECTOR FOR SCOPE ROUTINE
003634  012737  000340  000022                  MOV     #340,@#IOTVEC+2 ;;LEVEL 7
003642  012737  035330  000030                  MOV     #$ERROR,@#EMTVEC ;;EMT VECTOR FOR ERROR ROUTINE
003650  012737  000340  000032                  MOV     #340,@#EMTVEC+2 ;;LEVEL 7
003656  012737  037276  000034                  MOV     #$TRAP,@#TRAPVEC ;;TRAP VECTOR FOR TRAP CALLS
003664  012737  000340  000036                  MOV     #340,@#TRAPVEC+2;LEVEL 7
003672  012737  037362  000024                  MOV     #$PWRDN,@#PWRVEC ;;POWER FAILURE VECTOR
003700  012737  000340  000026                  MOV     #340,@#PWRVEC+2 ;;LEVEL 7
003706  016767  030700  030670                  MOV     $ENDCT,$EOPCT   ;;SETUP END-OF-PROGRAM COUNTER
003714  005067  175362                          CLR     $TIMES          ;;INITIALIZE NUMBER OF ITERATIONS
003720  005067  175360                          CLR     $ESCAPE         ;;CLEAR THE ESCAPE ON ERROR ADDRESS
003724  112767  000001  175163                  MOVB    #1,$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".
003732  012737  035034  000014                  MOV     #$RTRN,@#TBITVEC ;;SET "T" BIT VECTOR TO $RTRN
003740  012737  000340  000016                  MOV     #340,@#TBITVEC+2 ;;LEVEL 7
003746  012767  000002  031060                  MOV     #RTI,$RTRN      ;;SET $RTRN TO A RTI
003754  012737  004002  000010                  MOV     #65$,@#RESVEC   ;;TRY TO DO A RTT
003762  005046                                  CLR     -(SP)           ;;DUMMY PS
003764  012746  003772                          MOV     #64$,-(SP)      ;;AND PC
003770  000006                                  RTT                     ;;TRY THE RTT
003772  012767  000006  031034  64$:            MOV     #RTT,$RTRN      ;;RTT IS LEGAL--SET $RTRN TO A RTT
004000  000402                                  BR      66$
004002  062706  000010          65$:            ADD     #10,SP          ;;RTT ILLEGAL--CLEAN OFF THE STACK
004006  012737  000012  000010  66$:            MOV     #RESVEC+2,@#RESVEC ;;RESTORE TRAP CATCHER
004014  005067  031022                          CLR     $TBIT           ;;CLEAR "T" BIT SWITCH
004020  012767  004020  175060                  MOV     #.,$LPADR       ;;INITIALIZE THE LOOP ADDRESS FOR SCOPE
```

```
004026  012767  004026  175054                  MOV     #.,$LPERR        ;;SETUP THE ERROR LOOP ADDRESS
                                        ;;SIZE FOR A HARDWARE SWITCH REGISTER. IF NOT FOUND OR IT IS
                                        ;;EQUAL TO A "-1", SETUP FOR A SOFTWARE SWITCH REGISTER.
004034  013746  000004                          MOV     @#ERRVEC,-(SP)   ;;SAVE ERROR VECTOR
004040  012737  004074  000004                  MOV     #67$,@#ERRVEC    ;;SET UP ERROR VECTOR
004046  012767  177570  175064                  MOV     #DSWR,SWR        ;;SETUP FOR A HARDWARE SWICH REGISTER
004054  012767  177570  175060                  MOV     #DDISP,DISPLAY   ;;AND A HARDWARE DISPLAY REGISTER
004062  022777  177777  175050                  CMP     #-1,@SWR         ;;TRY TO REFERENCE HARDWARE SWR
004070  001012                                  BNE     69$              ;;BRANCH IF NO TIMEOUT TRAP OCCURRED
                                                                         ;;AND  THE HARDWARE SWR IS NOT = -1
004072  000403                                  BR      68$              ;;BRANCH IF NO TIMEOUT
004074  012716  004102                  67$:    MOV     #68$,(SP)        ;;SET UP FOR TRAP RETURN
004100  000002                                  RTI
004102  012767  000176  175030          68$:    MOV     #SWREG,SWR       ;;POINT TO SOFTWARE SWR
004110  012767  000174  175024                  MOV     #DISPREG,DISPLAY
004116  012637  000004                  69$:    MOV     (SP)+,@#ERRVEC   ;;RESTORE ERROR VECTOR

004122  005067  175176                          CLR     $PASS            ;;CLEAR PASS COUNT
004126  132767  000200  175203                  BITB    #APTSIZE,$ENVM   ;;TEST USER SIZE UNDER APT
004134  001403                                  BEQ     70$              ;;YES,USE NON-APT SWITCH
004136  012767  001340  174774                  MOV     #$SWREG,SWR      ;;NO,USE APT SWITCH REGISTER
004144                                  70$:
                                        .SBTTL  TYPE PROGRAM NAME
                                        ;;TYPE THE NAME OF THE PROGRAM IF FIRST PASS
004144  005227  177777                          INC     #-1              ;;FIRST TIME?
004150  001055                                  BNE     71$              ;;BRANCH IF NO
004152  022737  034770  000042                  CMP     #$ENDAD,@#42     ;;ACT-11?
004160  001451                                  BEQ     71$              ;;BRANCH IF YES
004162  104401  004230                          TYPE    ,72$             ;;TYPE ASCIZ STRING
                                        .SBTTL  GET VALUE FOR SOFTWARE SWITCH REGISTER
004166  005737  000042                          TST     @#42             ;;ARE WE RUNNING UNDER XXDP ACT?
004172  001012                                  BNE     73$              ;;BRANCH IF YES
004174  126727  175136  000001                  CMPB    $ENV,#1          ;;ARE WE RUNNING UNDER APT?
004202  001406                                  BEQ     73$              ;;BRANCH IF YES
004204  026727  174730  000176                  CMP     SWR,#SWREG       ;;SOFTWARE SWITCH REG SELECTED?
004212  001005                                  BNE     74$              ;;BRANCH IF NO
004214  104405                                  GTSWR                    ;;GET SOFT-SWR SETTINGS
004216  000403                                  BR      74$
004220  112767  000001  174706          73$:    MOVB    #1,$AUTOB        ;;SET AUTO-MODE INDICATOR
004226                                  74$:
004226  000426                                  BR      71$              ;;GET OVER THE ASCIZ
                                        ;;72$:  .ASCIZ  <CRLF>*DFFPA, FP11-A 11/34 FPP DIAGNOSTIC PART 1* <CRLF>
004304                                  71$:

004304                                  LOOP:




                                        ;;*************************************************************
                                        ;*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 CP
                                        ;*WITH THE FPP AND ALSO RUNS INTERNALLY WITHIN THE FPP. ONLY SMC 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 (SW13)
                                        ;*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.
                                        ;*
                                        ;*****************************************************************
 004304  000004                         TST1:   SCOPE
 004306  005037  004560                         CLR     @#AERFLG
 004312  104413                                 LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
 004314  012700  177777                         MOV     #-1,R0                  ;INITIALIZE THE COUNT PATTERN.
 004320  012737  004562  000244                 MOV     #AERR1,@#FPVECT         ;SET UP FOR UNABLE TO DECODE
 004326  012737  004574  000010                 MOV     #AERR2,@#10             ;FPP INSTRUCTION TRAP TO 244 OR 10.
 004334  005002                                 CLR     R2                      ;R2 IS THE 'AND' OF BAD DATA.
 004336  005102                                 COM     R2
 004340  005003                                 CLR     R3                      ;R3 IS THE 'OR' OF BAD DATA.
 004342  012737  004626  000004                 MOV     #AERR3,@#ERRVECT        ;IF EITHER INSTRUCTION
                                                                                ;FAILS TO GO THROUGH THE
                                                                                ;CORRECT SRC OR DST MODE AN
                                                                                ;ODD ADDRESS TRAP WILL OCCUR.
 004350                                 A1:
 004350  010004                         A11:    MOV     R0,R4
 004352  042704  030020                         BIC     #30020,R4
 004356  170104                                 LDFPS   R4                      ;TEST INSTRUCTION.

 004360  012701  177777                         MOV     #-1,R1
 004364  170201                         A12:    STFPS   R1                      ;TEST INSTRUCTION.
 004366  012737  040200  000244                 MOV     #FPSPUR,@#FPVECT        ;SET UP FOR UNEXPECTED TRAPS.
 004374  010004                                 MOV     R0,R4                   ;MASK OFF UNSETTABLE BITS.
 004376  042704  030020                         BIC     #30020,R4
 004402  012737  040232  000004                 MOV     #CPSPUR,@#ERRVECT
 004410  012737  040250  000010                 MOV     #CPTWO,@#10
 004416  020401                                 CMP     R4,R1                   ;COMPARE DATA EXPECTED WITH
                                                                                ;THE DATA READ.
 004420  001002                                 BNE     A3                      ;IF NOT EQUAL GO REPORT ERROR.

 004422  077026                         A2:     SOB     R0,A1                   ;OTHERWISE DECREMENT COUNT PATTERN
 004424  000425                                 BR      A5                      ;UNTIL IT IS ZERO.

 004426  005237  004560                 A3:     INC     @#AERFLG                ;RECORD ERROR.
 004432  050003                                 BIS     R0,R3                   ;COMPUTE 'OR' OF FAILING PATTERNS.
```

```
004434  010005                                  MOV     R0,R5                   ;COMPUTE 'AND' OF FAILING PATTERNS
004436  005105                                  COM     R5
004440  040502                                  BIC     R5,R2

004442  022737  000005  004560                  CMP     #5,@#AERFLG             ;SEE IF MORE THAN 5 ERRORS HAVE
004450  103412                                  BLO     A05                     ;OCCURRED. BR IF YES.
                                                                                ;OTHERWISE
                                                                                ;REPORT ERROR.
004452  012737  004350  001236                  MOV     #A1,@#$TMP2
004460  010037  001240                          MOV     R0,@#$TMP3
004464  010137  001242                          MOV     R1,@#$TMP4
004470  010437  001244                          MOV     R4,@#$TMP5
004474  104001                          A4:     ERROR   1

004476  000751                          A05:    BR      A2                      ;CONTINUE TESTING.

004500  005737  004560                  A5:     TST     @#AERFLG                ;SEE IF ANY ERRORS OCCURRED.
004504  001471                                  BEQ     ADONE                   ;IF NOT GO TO NEXT TEST.
004506  032777  020000  174424                  BIT     #SW13,@SWR              ;OTHERWISE SEE IF A SUMMARY
004514  001404                                  BEQ     A6                      ;SHOULD BE TYPED.
004516  032777  000200  174414                  BIT     #SW7,@SWR
004524  001461                                  BEQ     ADONE

004526                                  A6:                                     ;TYPE ERROR SUMMARY.
004526  010237  001236                          MOV     R2,@#$TMP2
004532  010337  001240                          MOV     R3,@#$TMP3
004536  012737  004552  001116                  MOV     #A7,@#$ERRPC
004544  112737  000002  001114                  MOVB    #2,@#$ITEMB
004552  004737  037546                  A7:     JSR     PC,@#ERTYPE
004556  000444                                  BR      ADONE

004560  000000                          AERFLG: .WORD   0

                                        ;UNABLE TO DECODE FPP INSTRUCTION. TRAPPED TO 244.
004562  011637  001236                  AERR1:  MOV     (SP),@#$TMP2            ;SAVE PC OF TRAP.
004566  022626                                  CMP     (SP)+,(SP)+
004570  104010                          1$:     ERROR   10
004572  000436                                  BR      ADONE

                                        ;UNABLE TO DECODE INSTRUCTION. TRAPPED TO 10.
004574  021627  004352                  AERR2:  CMP     (SP),#A11+2             ;DID TRAP OCCUR OF FPP INSTRUCTION?
004600  001405                                  BEQ     1$
004602  021627  004366                          CMP     (SP),#A12+2
004606  001402                                  BEQ     1$
004610  000137  040250                          JMP     @#CPTWO                 ;IF NOT FPP INSTRUCTION THEN
                                                                                ;REPORT SPURIOUS TRAP TO 10.

004614  011637  001236                  1$:     MOV     (SP),@#$TMP2            ;OTHERWISE REPORT IR DECIDE ERROR.
004620  022626                                  CMP     (SP)+,(SP)+
004622  104011                          2$:     ERROR   11
004624  000421                                  BR      ADONE

                                        ;TRAP TO 4 HANDLER:
004626  021627  004352                  AERR3:  CMP     (SP),#A11+2             ;DID THE TRAP OCCUR ON THE
004632  001405                                  BEQ     1$                      ;LDFPS INSTRUCTION?
004634  021627  004366                          CMP     (SP),#A12+2             ;OR THE STFPS INSTRUCTION?
```

```
MAINDEC-11-DFFBA-A    FOR 11/34 FPP DIAGNOSTIC PART 1 MACY11 27(1006)  01-NOV-76  21:03  PAGE 35
DFFBA1.P11    01-NOV-76 21:03          T1      LDFPS, STFPS AND DATA PATHS TEST

                                       BEQ     2$
                                       JMP     @#CPSPUR                ;IF NEITHER THEN REPORT
                                                                       ;UNEXPECTED TRAP TO 4.

004646 011637 001236           1$:     MOV     (SP),@#$TMP2
004652 022626                          CMP     (SP)+,(SP)+
004654 104014                  15$:    ERROR   14
004656 000404                          BR      ADONE

004660 011637 001236           2$:     MOV     (SP),@#$TMP2
004664 022626                          CMP     (SP)+,(SP)+
004666 104015                  25$:    ERROR   15

004670                         ADONE:
004670 104412                          RESETUP                 ;GO INITIALIZE THE FPS AND STACK; AND
                                                               ;SEE IF THE USER HAS EXPRESSED
                                                               ;THE DESIRE TO CHANGE THE SOFTWARE
                                                               ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
                                                               ;THE USER TYPED CONTROL G?).


                                ;;*************************************************************
                                ;*TEST 2        CFCC TEST
                                ;*
                                ;*THIS IS A TEST OF THE COPY CONDITION CODES INSTRUCTION, CFCC.
                                ;*
                                ;;*************************************************************
004672 000004                  ST2:    SCOPE
004674 104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
004676 012700 000017                   MOV     #17,R0          ;R0 CONTAINS TO TEST PATTERN.

004702                         B1:
004702 170100                          LDFPS   R0              ;LOAD THE TEST PATTERN

004704                         B2:
004704 170000                          CFCC                    ;COPY CONDITION CODES.

004706 013703 177776                   MOV     @#PSW,R3        ;SEE IF PATTERN TRANSFERED.
004712 042703 177760                   BIC     #177760,R3
004716 020003                          CMP     R0,R3
004720 001002                          BNE     BERR

004722 077011                  B3:     SOB     R0,B1
004724 000422                          BR      BDONE

004726                         BERR:
004726 170201                          STFPS   R1              ;WAS FPS MODIFIED BY CFCC?
004730 012737 004704 001236            MOV     #B2,@#$TMP2

004736 020001                          CMP     R0,R1
004740 001006                          BNE     BERR1

004742 010337 001240                   MOV     R3,@#$TMP3
004746 010037 001242                   MOV     R0,@#$TMP4
004752 104003                  1$:     ERROR   3
004754 000762                          BR      B3
```

```
1914
1915  004756                          BERR1:
1916  004756  010037  001240                  MOV     R0,@#$TMP3
1917  004762  010137  001242                  MOV     R1,@#$TMP4
1918  004766  104004                  1$:     ERROR   4
1919  004770  000754                          BR      B3
1920
1921  004772                          BDONE:
1922  004772  104412                          RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
                                                                        ;SEE IF THE USER HAS EXPRESSED
                                                                        ;THE DESIRE TO CHANGE THE SOFTWARE
                                                                        ;VIRTUAL CONSOLE SWITCH REGISTER (SR
                                                                        ;THE USER TYPED CONTROL G?).

                                        ;;**********************************************************************
                                        ;*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.
                                        ;*
                                        ;;**********************************************************************
1937  004774  000004                  TST3:   SCOPE
1938  004776  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
1939  005000  012737  000760  001244          MOV     #760,@#$TMP5
1940  005006  012737  000202  001250  C1:     MOV     #202,@#$TMP7
1941  005014  012737  041352  001252          MOV     #SETF1,@#$TMP10
1942  005022  005000                          CLR     R0

1944  005024  170100                          LDFPS   R0                      ;CLEAR THE FPS.
1945  005026  012737  005034  001236          MOV     #C15,@#$TMP2

1947  005034  170001                  C15:    SETF                            ;TEST INSTRUCTION.

1949  005036  170201                          STFPS   R1                      ;GET RESULT.
1950  005040  005002                          CLR     R2
1951  005042  020201                          CMP     R2,R1                   ;DID AN ERROR OCCUR?
1952  005044  001402                          BEQ     1$
1953  005046  004737  005432                  JSR     PC,@#CERR1

1955  005052                          1$:
1956  005052  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
1957  005054  012700  147757          C2:     MOV     #147757,R0

1959  005060  170100                          LDFPS   R0                      ;PUT 147757 IS FPS
1960  005062  012737  005070  001236          MOV     #C25,@#$TMP2
1961  005070  170001                  C25:    SETF                            ;CLEAR FD BIT.

1963  005072  170201                          STFPS   R1                      ;GET RESULT
1964  005074  012702  147557                  MOV     #147557,R2
1965  005100  020102                          CMP     R1,R2                   ;RESULT CORRECT.
1966  005102  001402                          BEQ     1$
1967  005104  004737  005530                  JSR     PC,@#CERR2

1969  005110                          1$:
```

```
1970  005110  104413                  LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
1971  005112  012737  000203  001250  C3:   MOV   #203,@#$TMP7
1972  005120  012737  041360  001252        MOV   #SETD1,@#$TMP10
1973  005126  012700  147757             MOV   #147757,R0
1974
1975  005132  170100                  LDFPS R0                ;LOAD 147757 INTO FPS.
1976  005134  012737  005142  001236        MOV   #C35,@#$TMP2
1977  005142  170011                  C35:  SETD                    ;SETD FD BIT.
1978
1979  005144  170201                  STFPS R1
1980  005146  012702  147757             MOV   #147757,R2
1981  005152  020102                  CMP   R1,R2             ;RESULT CORRECT?
1982  005154  001402                  BEQ   1$
1983  005156  004737  005a30             JSR   PC,@#CERR2
1984
1985  005162                          1$:
1986  005162  104413                  LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
1987  005164  005000                  C4:   CLR   R0
1988  005166  170100                  LDFPS R0                ;CLEAR FPS.
1989  005170  012737  005176  001236        MOV   #C45,@#$TMP2
1990
1991  005176  170011                  C45:  SETD                    ;SET FD BIT.
1992
1993  005200  170201                  STFPS R1                ;GET RESULT.
1994  005202  012702  000200             MOV   #200,R2
1995  005206  020102                  CMP   R1,R2             ;RESULT CORRECT?
1996  005210  001402                  BEQ   1$
1997  005212  004737  005432             JSR   PC,@#CERR1
1998
1999  005216                          1$:
2000  005216  104413                  LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
2001  005220  012737  000204  001250  C5:   MOV   #204,@#$TMP7
2002  005226  012737  041366  001252        MOV   #SETI1,@#$TMP10
2003  005234  005000                  CLR   R0
2004
2005  005236  170100                  LDFPS R0                ;CLEAR FPS
2006  005240  012737  005246  001236        MOV   #C55,@#$TMP2
2007
2008  005246  170002                  C55:  SETI                    ;CLEAR FL BIT.
2009
2010  005250  170201                  STFPS R1                ;GET RESULT.
2011  005252  005002                  CLR   R2
2012  005254  020201                  CMP   R2,R1             ;RESULT CORRECT?
2013  005256  001402                  BEQ   1$
2014  005260  004737  005432             JSR   PC,@#CERR1
2015
2016  005264                          1$:
2017  005264  104413                  LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
2018  005266  012700  147757          C6:   MOV   #147757,R0
2019  005272  170100                  LDFPS R0                ;PUT 147757 INTO FPS
2020  005274  012737  005302  001236        MOV   #C65,@#$TMP2
2021
2022  005302  170002                  C65:  SETI                    ;CLEAR FL BIT.
2023
2024  005304  170201                  STFPS R1                ;GET THE RESULT.
2025  005306  012702  147657             MOV   #147657,R2
```

```
2026  005312  020102                        CMP     R1,R2                   ;RESULT CORRECT?
2027  005314  001402                        BEQ     1$
2028  005316  004737  005530                JSR     PC,@#CERR2
2029
2030  005322                        1$:
2031  005322  104413                        LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
2032  005324  012737  000205  001250  C7:   MOV     #205,@#$TMP7
2033  005332  012737  041374  001252        MOV     #SETL1,@#$TMP10
2034  005340  012700  147757               MOV     #147757,R0
2035  005344  170100                        LDFPS   R0                      ;SET FPS TO 147757.
2036  005346  012737  005354  001236        MOV     #C75,@#$TMP2
2037
2038  005354  170012                C75:    SETL                            ;SET FL BIT.
2039
2040  005356  170201                        STFPS   R1                      ;GET THE RESULT.
2041  005360  012702  147757               MOV     #147757,R2
2042  005364  020102                        CMP     R1,R2                   ;RESULT CORRECT?
2043  005366  001402                        BEQ     1$
2044  005370  004737  005530                JSR     PC,@#CERR2
2045
2046  005374                        1$:
2047  005374  104413                        LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
2048  005376  005000                C8:     CLR     R0
2049  005400  170100                        LDFPS   R0                      ;CLEAR FPS.
2050  005402  012737  005410  001236        MOV     #C85,@#$TMP2
2051
2052  005410  170012                C85:    SETL                            ;SET FL BIT.
2053
2054  005412  170201                        STFPS   R1
2055  005414  012702  000100               MOV     #100,R2
2056  005420  020102                        CMP     R1,R2                   ;RESULT CORRECT.
2057  005422  001402                        BEQ     1$
2058  005424  004737  005432                JSR     PC,@#CERR1
2059
2060  005430  000522                1$:     BR      CDONE
2061
2062                                ;THESE ARE ERROR ANALYSIS ROUTINES:
2063  005432  010103                CERR1:  MOV     R1,R3
2064  005434  032703  177477               BIT     #177477,R3              ;ARE ANY OTHER BITS SET?
2065  005440  001401                        BEQ     2$
2066  005442  000503                1$:     BR      CERR4
2067
2068  005444  022703  000300        2$:     CMP     #300,R3                 ;ARE BOTH FD AND FL SET?
2069  005450  001774                        BEQ     1$
2070  005452  032703  000300               BIT     #300,R3                 ;ARE THEY BOTH CLEAR?
2071  005456  001771                        BEQ     1$
2072
2073  005460  032703  000200               BIT     #200,R3                 ;IS FD SET?
2074  005464  001407                        BEQ     3$
2075  005466  012737  041360  001254        MOV     #SETD1,@#$TMP11
2076  005474  012737  000203  001246        MOV     #203,@#$TMP6
2077  005502  000452                        BR      CERR3
2078
2079  005504  032703  000100        3$:     BIT     #100,R3                 ;IS FL SET
2080  005510  001754                        BEQ     1$
2081  005512  012737  041374  001254        MOV     #SETL1,@#$TMP11
```

```
2082  005520  012737  000205  001246          MOV     #205,@#$TMP6
2083  005526  000440                          BR      CERR3
2084
2085  005530  010103                  CERR2:  MOV     R1,R3
2086  005532  005103                          COM     R3
2087
2088  005534  032703  177477                  BIT     #177477,R3              ;ARE ANY OTHER BITS SET?
2089  005540  001401                          BEQ     2$
2090  005542  000443                  1$:     BR      CERR4
2091
2092  005544  032703  000300          2$:     BIT     #300,R3                 ;ARE BOTH FD AND FL SET?
2093  005550  001774                          BEQ     1$
2094  005552  032701  000300                  BIT     #300,R1                 ;ARE THEY BOTH CLEAR?
2095  005556  001771                          BEQ     1$
2096
2097  005560  032701  000200                  BIT     #200,R1                 ;IS FD CLEAR?
2098  005564  001007                          BNE     3$
2099  005566  012737  041352  001254          MOV     #SETF1,@#$TMP11
2100  005574  012737  000202  001246          MOV     #202,@#$TMP6
2101  005602  000412                          BR      CERR3
2102
2103  005604  032701  000100          3$:     BIT     #100,R1
2104  005610  001354                          BNE     1$                      ;IS FL CLEAR.
2105  005612  012737  041366  001254          MOV     #SETI1,@#$TMP11
2106  005620  012737  000204  001246          MOV     #204,@#$TMP6
2107  005626  000400                          BR      CERR3
2108
2109                                  ;REPORT THE ERRORS:
2110  005630                          CERR3:
2111  005630  010137  001240                  MOV     R1,@#$TMP3
2112  005634  010237  001242                  MOV     R2,@#$TMP4
2113  005640  012637  005674                  MOV     (SP)+,@#CPC
2114  005644  104012                  1$:     ERROR   12
2115  005646  000177  000022                  JMP     @CPC
2116
2117  005652                          CERR4:
2118  005652  010137  001240                  MOV     R1,@#$TMP3
2119  005656  010237  001242                  MOV     R2,@#$TMP4
2120  005662  012637  005674                  MOV     (SP)+,@#CPC
2121  005666  104013                  1$:     ERROR   13
2122  005670  000177  000000                  JMP     @CPC
2123
2124  005674  000000                  CPC:    .WORD   0
2125
2126  005676                          CDONE:
2127  005676  104412                          RSETUP                          ;GO INITIALIZE THE FPS AND STACK; AND
2128                                                                          ;SEE IF THE USER HAS EXPRESSED
2129                                                                          ;THE DESIRE TO CHANGE THE SOFTWARE
2130                                                                          ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
2131                                                                          ;THE USER TYPED CONTROL G?).
2132
2133
2134                                  ;;*****************************************************************
2135                                  ;*TEST 4         ILLEGAL FPP OP CODES AND STST TEST
2136                                  ;*
2137                                  ;*THIS IS A TEST OF THE FPP OPERATION CODES:
```

```
                                        ;*                              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.
                                        ;*
                                        ;;************************************************************
005700  000004                          ST4:    SCOPE
005702  104413                                  LPERR                           ;SET UP THE LOOP ON ERROR ADDRESS.
005704  012705  170003                          MOV     #170003,R5              ;INITIAL OP CODE.
005710  012737  006114  000004                  MOV     #DERR2,@#ERRVECT
005716  012737  006020  000244                  MOV     #DERR1,@#FPVECT

005724  005000                          D1:     CLR     R0
005726  170100                                  LDFPS   R0                      ;CLEAR FPS.
005730  005002                                  CLR     R2
005732  010537  005750                          MOV     R5,@#D2                 ;SET UP THE ILLEGAL INSTRUCTION.
005736  010537  001244                          MOV     R5,@#$TMP5
005742  012737  005750  001236                  MOV     #D2,@#$TMP2
005750  000000                          D2:     .WORD   0
005752  170000                          D3:     CFCC
005754  005202                                  INC     R2
005756  005202                          D4:     INC     R2

005760  170201                                  STFPS   R1                      ;REPORT FAILURE. DID NOT TRAP.
005762  010137  001240                          MOV     R1,@#$TMP3
005766  104016                          1$:     ERROR   16

005770  022705  170010                  D5:     CMP     #170010,R5              ;COMPUTE NEXT OP CODE
005774  001003                                  BNE     D6
005776  012705  170013                          MOV     #170013,R5
006002  000750                                  BR      D1

006004  022705  170077                  D6:     CMP     #170077,R5
006010  001001                                  BNE     D7
006012  000452                                  BR      DDONE
006014  005205                          D7:     INC     R5
006016  000742                                  BR      D1

006020  022716  005752                  DERR1:  CMP     #D3,(SP)                ;DID TRAP OCCUR ON TEST INSTRUCTION?
006024  001402                                  BEQ     1$
006026  000137  040200                          JMP     @#FPSPUR

006032  022626                          1$:     CMP     (SP)+,(SP)+
006034  170201                                  STFPS   R1                      ;GET THE FPS AND SEE IF IT IS
006036  022701  100000                          CMP     #100000,R1              ;SET CORRECTLY.
006042  001406                                  BEQ     3$
```

```
006044  012737  100000  001240          MOV     #100000,@#$TMP3
006052  010437  001242                  MOV     R4,@#$TMP4
006056  104017                  2$:     ERROR   17
006060  012704  000001          3$:     MOV     #1,R4
006064  170304                  D9:     STST    R4                      ;GET THE FEC CODE. NOTE THAT
                                                                        ;IF THE DESTINATION MODE IS
                                                                        ;IMPROPERLY DECODED AN ODD
                                                                        ;ADDRESS TRAP TO 4 SHOULD OCCUR.
006066  022704  000002                  CMP     #2,R4                   ;WAS FEC CORRECT?
006072  001001                          BNE     D9
006074  000735                          BR      D5

006076                          D9:                                     ;REPORT STST FAILURE
006076  012737  006064  001240          MOV     #D8,@#$TMP3
006104  010437  001242                  MOV     R4,@#$TMP4
006110  104020                  1$:     ERROR   20
006112  000726                          BR      D5

006114  022716  006066          DERR2:  CMP     #D8+2,(SP)              ;DID THE TRAP OCCUR ON THE
006120  001402                          BEQ     D10                     ;STST INSTRUCTION?
006122  000137  040232                  JMP     @#CPSPUR

006126                          D10:
006126  011637  001236                  MOV     (SP),@#$TMP2
006132  022626                          CMP     (SP)+,(SP)+
006134  104021                  1$:     ERROR   21
006136  000714                          BR      D5

006140                          DDONE:
006140  104412                          RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
                                                                ;SEE IF THE USER HAS EXPRESSED
                                                                ;THE DESIRE TO CHANGE THE SOFTWARE
                                                                ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
                                                                ;THE USER TYPED CONTROL G?).


                                        ;;**************************************************************
                                        ;*TEST 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.
                                        ;*
                                        ;;**************************************************************
006142  000004                  TST5:   SCOPE
006144  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
006146  012737  006250  000244          MOV     #EERR2,@#FPVECT         ;SETUP FOR THE INTERRUPT.

006154  012700  040000          E1:     MOV     #40000,R0
006160  170100                          LDFPS   R0                      ;SET FID.
006162  012737  006170  001236          MOV     #E3,@#$TMP2
006170                          E2:
006170  170020                  E3:     .WORD   170020                  ;ILLEGAL FPP INSTRUCTION.
006172  170000                  E4:     CFCC
```

```
 006174  170201                         STFPS   R1                      ;SEE IF ERROR WAS DETECTED.
 006176  022701  140000                 CMP     #140000,R1
 006202  001005                         BNE     EERR0

 006204  170304                         STST    R4                      ;SEE IF FEC=2
 006206  022704  000002                 CMP     #2,R4
 006212  001010                         BNE     EERR1
 006214  000431                         BR      EDONE

 006216                         EERR0:                                  ;REPORT FPS INCORRECTLY SET.
 006216  010137  001240                 MOV     R1,@#$TMP3
 006222  012737  140000  001242         MOV     #140000,@#$TMP4
 006230  104022                 1$:     ERROR   22
 006232  000422                         BR      EDONE

 006234                         EERR1:                                  ;REPORT FEC NOT 2.
 006234  010537  001240                 MOV     R5,@#$TMP3
 006240  010437  001242                 MOV     R4,@#$TMP4
 006244  104023                 1$:     ERROR   23
 006246  000414                         BR      EDONE

 006250  021627  006172         EERR2:  CMP     (SP),#E4                ;DID THE ILLEGAL INSTRUCTION TRAP?
 006254  001402                         BEQ     1$
 006256  000137  040200                 JMP     @#FPSPUR

 006262                         1$:
 006262  011637  001236                 MOV     (SP),@#$TMP2
 006266  022626                         CMP     (SP)+,(SP)+
 006270  170201                         STFPS   R1
 006272  010137  001240                 MOV     R1,@#$TMP3
 006276  104024                 2$:     ERROR   24

 006300                         EDONE:
 006300  104412                         RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
                                                                ;SEE IF THE USER HAS EXPRESSED
                                                                ;THE DESIRE TO CHANGE THE SOFTWARE
                                                                ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
                                                                ;THE USER TYPED CONTROL G?).



                                ;;*****************************************************************
                                ;*TEST 6         LDD AND STD, WITH SRC AND DST MODE 1, TEST
                                ;*
                                ;*THIS IS A TEST OF BOTH THE INSTRUCTION:
                                ;*               LDD      (R0),AC0
                                ;*AND THE INSTRUCTION:
                                ;*               STD      AC0,(R0)
                                ;*MOST OF THE FAILURES ARE ISOLATED TO THE SRC OR DST FLOWS. NOTE
                                ;*THAT THE INTEGRITY OF AC0 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.
                                ;*
                                ;;*****************************************************************
 006302  000004                 TST6:   SCOPE
```

```
                                F1:
006304  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
006306  012737  006356  001236          MOV     #F3,@#$TMP2
006314  005000                          CLR     R0
006316  170100                          LDFPS   R0
006320  170011                          SETD
006322  012701  010110                  MOV     #FDAT10,R1              ;SET UP THE LOAD DATA.
006326  012702  010154                  MOV     #FXDAT0,R2
006332  012703  000010                  MOV     #10,R3

006336  012221                  F2:     MOV     (R2)+,(R1)+
006340  077302                          SOB     R3,F2

006342  012700  010120                  MOV     #FDAT14,R0              ;SETUP R0 FOR THE LDD (R0),AC0.
006346  012737  007574  000004          MOV     #FERR20,@#ERRVECT       ;IF THE SRC FLOWS FAIL, THEN
                                                                        ;AN ODD ADDRESS MAY OCCUR.
006354  005003                          CLR     R3

006356  172410                  F3:     LDD     (R0),AC0
006360  005203                  F4:     INC     R3
006362  005203                          INC     R3

006364  020027  010120                  CMP     R0,#FDAT14              ;WAS R0 AFFECTED?
006370  001402                          BEQ     F5
006372  000137  006740                  JMP     @#FERR1

006376  020327  000002          F5:     CMP     R3,#2                   ;SEE IF THE PC WAS ADVERSELY
006402  001402                          BEQ     1$                      ;AFFECTED DURING THE INSTRUCTION.
006404  000137  007036                  JMP     @#FERR2

006410  012701  010110          1$:     MOV     #FDAT10,R1              ;MAKE SURE THE SOURCE DATA WAS
006414  012702  010154                  MOV     #FXDAT0,R2              ;NOT AFFECTED.
006420  012703  000010                  MOV     #10,R3
006424  022122                  2$:     CMP     (R1)+,(R2)+
006426  001402                          BEQ     3$
006430  000137  006702                  JMP     @#FERR0
006434  077305                  3$:     SOB     R3,2$

006436  170201                          STFPS   R1                      ;MAKE SURE THE FPS IS CORRECT.
006440  022701  000200                  CMP     #200,R1
006444  001402                          BEQ     F6
006446  000137  007554                  JMP     @#FERR11

006452                          F6:
006452  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
006454  012737  006516  001236          MOV     #F10,@#$TMP2

006462  012703  177777                  MOV     #-1,R3
006466  012704  000010                  MOV     #10,R4
006472  012705  010132                  MOV     #FDAT00,R5              ;SET UP THE OUTPUT DATA BUFFER.
006476  010325                  F7:     MOV     R3,(R5)+
006500  077402                          SOB     R4,F7

006502  012700  010142                  MOV     #FDAT04,R0              ;SET UP R0 FOR DST MODE 1 REG 0.
006506  012737  007742  000004          MOV     #FERR25,@#ERRVECT       ;IF THE DST FLOWS FAIL AN ODD
                                                                        ;ADDRESS COULD OCCUR.
```

```
006514  005003                   CLR     R3

006516  174010           F10:    STD     AC0,(R0)            ;TEST INSTRUCTION.
006520  005203           F11:    INC     R3
006522  005203                   INC     R3

006524  020027  010142           CMP     R0,#FDAT04           WAS R0 MODIFIED?
006530  001402                   BEQ     F12
006532  000137  007076           JMP     @#FERR3

006536  020327  000002   F12:    CMP     R3,#2               ;WAS THE PC AFFECTED CORRECTLY?
006542  001402                   BEQ     F135
006544  000167  000320           JMP     FERR4

006550  012701  010132   F135:   MOV     #FDAT00,R1
006554  012702  010154           MOV     #FXDAT0,R2

006560  022122                   CMP     (R1)+,(R2)+         ;SEE IF THE DATA WAS OUTPUT
006562  001402                   BEQ     F13                 ;TO THE TARGET AREA CORRECTLY.
006564  000137  007174           JMP     @#FERR5

006570  022122           F13:    CMP     (R1)+,(R2)+
006572  001402                   BEQ     F14
006574  000137  007174           JMP     @#FERR5

006600  022122           F14:    CMP     (R1)+,(R2)+
006602  001402                   BEQ     F15
006604  000137  007174           JMP     @#FERR5

006610  022122           F15:    CMP     (R1)+,(R2)+
006612  001402                   BEQ     F16
006614  000137  007174           JMP     @#FERR5

006620  022122           F16:    CMP     (R1)+,(R2)+
006622  001402                   BEQ     F17
006624  000137  007520           JMP     @#FERR10

006630  022122           F17:    CMP     (R1)+,(R2)+
006632  001402                   BEQ     F20
006634  000137  007230           JMP     @#FERR6

006640  022122           F20:    CMP     (R1)+,(R2)+
006642  001402                   BEQ     F21
006644  000167  000514           JMP     FERR7

006650  022122           F21:    CMP     (R1)+,(R2)+
006652  001402                   BEQ     F22
006654  000137  007520           JMP     @#FERR10

006660  005001           F22:    CLR     R1
006662  170201                   STFPS   R1                  ;MAKE SURE FPS IS CORRECT.
006664  022701  000200           CMP     #200,R1
006670  001402                   BEQ     F23
006672  000137  007554           JMP     @#FERR11
006676  000137  010174   F23:    JMP     @#FDONE
```

```
                                FERR0:                                  ;SOURCE DATA AFFECTED BY
                                                                        ;THE LDD INSTRUCTION.
                        012737  010154  001240          MOV     #FXDAT0,@#$TMP3
                        012737  010166  001242          MOV     #FXDAT0+12,@#$TMP4
                        012737  010122  001244          MOV     #FDATI0,@#$TMP5
                        012737  010132  001246          MOV     #FDATI0+12,@#$TMP6
                        104025                  1$:     ERROR   25
                        000137  010174                  JMP     @#FDONE

                        012737  010120  001242  FERR1:  MOV     #FDATI4,@#$TMP4         ;FSRC FLOWS FAILURE.
                        010037  001240                  MOV     R0,@#$TMP3
                        012737  000762  001244          MOV     #762,@#$TMP5
                        012737  000321  001250          MOV     #321,@#$TMP7

                        022700  010110                  CMP     #FDATI0,R0              ;FSRC MODE 4
                        001004                          BNE     1$
                        012737  000324  001246          MOV     #324,@#$TMP6
                        000412                          BR      4$

                        022700  010130          1$:     CMP     #FDATI4+10,R0           ;FSRC MODE 2
                        001004                          BNE     2$
                        012737  000322  001246          MOV     #322,@#$TMP6
                        000403                          BR      4$

                                                2$:
                        104027                  3$:     ERROR   27
                        000137  010174                  JMP     @#FDONE

                                                4$:
                        104026                  5$:     ERROR   26
                        000137  010174                  JMP     @#FDONE

                        012701  006360          FERR2:  MOV     #F4,R1                  ;THE PC WAS INCORRECTLY AFFECTED
                                                                                        ;DURING THE INSTRUCTION.
                        010137  001242          FER2:   MOV     R1,@#$TMP4
                        162701  000004                  SUB     #4,R1
                        006303                          ASL     R3
                        060301                          ADD     R3,R1
                        010137  001240                  MOV     R1,@#$TMP3
                        104030                  1$:     ERROR   30
                        000137  010174                  JMP     @#FDONE

                        012701  006520          FERR4:  MOV     #F11,R1
                        000762                          BR      FER2

                        012737  010142  001242  FERR3:  MOV     #FDAT04,@#$TMP4         ;FAILURE IN THE FDST FLOWS.
                        010037  001240                  MOV     R0,@#$TMP3
                        012737  000527  001244          MOV     #527,@#$TMP5
                        012737  000641  001250          MOV     #641,@#$TMP7

                        022700  010132                  CMP     #FDAT00,R0              ;DST MODE 4
                        001004                          BNE     1$
                        012737  000644  001246          MOV     #644,@#$TMP6
                        000412                          BR      4$

                        022700  010152          1$:     CMP     #FDAT04+10,R0           ;DST MODE 2
                        001004                          BNE     2$
```

```
007150  012737  000642  001246                  MOV     #642,@#$TMP6
007156  000403                                  BR      4$

007160                                  2$:
007160  104032                          3$:     ERROR   32
007162  000137  010174                          JMP     @#FDONE

007166                                  4$:
007166  104031                          5$:     ERROR   31
007170  000137  010174                          JMP     @#FDONE

007174                                  FERR5:                                  ;FAILURE OF STD.
007174  010037  001240                          MOV     R0,@#$TMP3
007200  012737  010132  001242                  MOV     #FDAT00,@#$TMP4
007206  012737  010150  001244                  MOV     #FDAT07,@#$TMP5
007214  012737  010154  001246                  MOV     #FXDAT0,@#$TMP6
007222  104033                          1$:     ERROR   33
007224  000137  010174                          JMP     @#FDONE

007230  012701  010144                  FERR6:  MOV     #FDAT05,R1              ;DID (BUT GR7) FAIL IN THE FDST
007234  012702  177777                          MOV     #-1,R2                  ;FLOWS?
007240  012703  000003                          MOV     #3,R3
007244  020221                          1$:     CMP     R2,(R1)+
007246  001017                                  BNE     5$
007250  077303                                  SOB     R3,1$

                                                                                ;REPORT FAILURE OF (BUT GR7) IN
007252  010037  001240                          MOV     R0,@#$TMP3              ;THE FDST FLOWS.
007256  012737  000412  001244                  MOV     #412,@#$TMP5
007264  012737  000147  001246                  MOV     #147,@#$TMP6
007272  012737  000145  001250                  MOV     #145,@#$TMP7
007300  104034                          2$:     ERROR   34
007302  000137  010174                          JMP     @#FDONE

007306  012701  010144                  5$:     MOV     #FDAT05,R1              ;DID (BUT GR7) FAIL IN THE SRC FLOWS?
007312  012703  000003                          MOV     #3,R3
007316  005721                          6$:     TST     (R1)+
007320  001402                                  BEQ     7$
007322  000137  007520                          JMP     @#FERR10
007326  077305                          7$:     SOB     R3,6$

                                                                                ;REPORT FAILURE OF (BUT GR7) IN
007330  010037  001240                          MOV     R0,@#$TMP3              ;THE FSRC FLOWS.
007334  012737  000207  001244                  MOV     #207,@#$TMP5
007342  012737  000176  001246                  MOV     #176,@#$TMP6
007350  012737  000174  001250                  MOV     #174,@#$TMP7

007356  104035                          10$:    ERROR   35
007360  000137  010174                          JMP     @#FDONE

007364  012701  010146                  FERR7:  MOV     #FDAT06,R1              ;DID (BUT FD) FAIL IN THE FDST FLOWS?
007370  012702  177777                          MOV     #-1,R2
007374  012703  000002                          MOV     #2,R3
007400  020221                          1$:     CMP     R2,(R1)+
007402  001017                                  BNE     5$
007404  077303                                  SOB     R3,1$
```

```
                                                                        ;REPORT FAILURE OF  BUT FD  IN THE
                                                                        ;FDST FLOWS.
007406  010037  001240                  MOV     R0,@#$TMP3
007412  012737  000707  001244          MOV     #707,@#$TMP5
007420  012737  000244  001246          MOV     #244,@#$TMP6
007426  012737  000245  001250          MOV     #245,@#$TMP7
007434  104036          2$:     ERROR   36
007436  000137  010174                  JMP     @#FDONE

007442  012701  010146          5$:     MOV     #FDAT06,R1              ;DID  BUT FD, FAIL IN THE FSRC FLOWS
007446  012703  000002                  MOV     #2,R3
007452  005721          6$:     TST     (R1)+
007454  001402                  BEQ     7$
007456  000137  007520                  JMP     @#FERR10
007462  077305          7$:     SOB     R3,6$

                                                                        ;REPORT FAILURE OF  BUT FD  IN THE
                                                                        ;FSRC FLOWS.
007464  010037  001240                  MOV     R0,@#$TMP3
007470  012737  000441  001244          MOV     #441,@#$TMP5
007476  012737  000076  001246          MOV     #76,@#$TMP6
007504  012737  000077  001250          MOV     #77,@#$TMP7
007512  104037          10$:    ERROR 37
007514  000137  010174                  JMP     @#FDONE

007520          FERR1C:                                                 ;REPORT DATA ERROR.
007520  010037  001240                  MOV     R0,@#$TMP3
007524  012737  010142  001242          MOV     #FDAT04,@#$TMP4
007532  012737  010150  001244          MOV     #FDAT07,@#$TMP5
007540  012737  010164  001246          MOV     #FXDAT4,@#$TMP6
007546  104040          1$:     ERROR   40
007550  000137  010174                  JMP     @#FDONE

007554          FERR11:                                                 ;REPORT BAD FPS.
007554  010137  001240                  MOV     R1,@#$TMP3
007560  012737  000200  001242          MOV     #200,@#$TMP4
007566  104041          1$:     ERROR   41
007570  000137  010174                  JMP     @#FDONE

007574  012737  040411  001264  FERR20: MOV     #NULL,@#$TMP15          ;THE EXECUTION OF THE LDD
007602  005037  001252                  CLR     @#$TMP10                ;CAUSED A TRAP TO 4, BECAUSE
007606  011637  001236                  MOV     (SP),@#$TMP2            ;A FSRC FLOW FAILURE RESULTED
007612  012737  010120  001240          MOV     #FDATI4,@#$TMP3         ;IN AN ODD ADDRESS.
007620  012737  000321  001250          MOV     #321,@#$TMP7
007626  012737  000762  001244          MOV     #762,@#$TMP5

007634  021627  006362                  CMP     (SP),#F4+2              ;SEE IF FSRC MODE 6 OR 7 WAS
007640  001424                  BEQ     FERR21                  ;EXECUTED.

007642  020027  010116                  CMP     R0,#FDATI3              ;FSRC MODE 5?
007646  001006                  BNE     2$

                                                                        ;REPORT FSRC FLOW FAILURE TO
007650  012737  000325  001246          MOV     #325,@#$TMP6            ;MODE 5.
007656  022626                  CMP     (SP)+,(SP)+
007660  104042          1$:     ERROR   42
007662  000544                  BR      FDONE
```

J04


```
007664  020027  010122          2$:     CMP     R0,#FDAT15              ;FSRC MODE 3?
007670  001402                          BEQ     3$
007672  000137  040232                  JMP     @#CPSPUR

                                3$:                                     ;REPORT FSRC    FLOW FAILURE TO
007676                                                                  ;MODE 3.
007676  012737  000323  001246          MOV     #323,@#$TMP6
007704  022626                          CMP     (SP)+,(SP)+
007706  104042                  4$:     ERROR   42
007710  000531                          BR      FDONE

007712  022626                  FERR21: CMP     (SP)+,(SP)+             ;REPORT FSRC FLOW FAILURE TO
                                                                        ;MODE 6 OR MODE 7.
007714  012737  042156  001264          MOV     #MS16,@#$TMP15
007722  012737  000326  001246          MOV     #326,@#$TMP6
007730  012737  000327  001252          MOV     #327,@#$TMP10
007736  104042                  1$:     ERROR   42
007740  000515                          BR      FDONE

007742  012737  040411  001264  FERR25: MOV     #NULL,@#$TMP15          ;THE EXECUTION OF THE STD INSTRUCTION
007750  005037  001252                  CLR     @#$TMP10                ;TRAPPED TO 4, BECAUSE A FAILURE
007754  012737  010142  001240          MOV     #FDAT04,@#$TMP3         ;IN THE FDST FLOWS RESULTED
007762  011637  001236                  MOV     (SP),@#$TMP2            ;IN AN ODD ADDRESS.
007766  012737  000527  001244          MOV     #527,@#$TMP5
007774  012737  000641  001250          MOV     #641,@#$TMP7

010002  021627  006520                  CMP     (SP),#F10+2             ;FLOW FAILURE TO FDST MODE 6 OR 7?
010006  001424                          BEQ     FERR26

010010  020027  010140                  CMP     R0,#FDAT03              ;DID FDST FLOW FAIL TO MODE 5?
010014  001006                          BNE     2$

                                                                        ;REPORT FLOW FAILURE TO FDST
010016  012737  000645  001246          MOV     #645,@#$TMP6            ;MODE 5.
010024  022626                          CMP     (SP)+,(SP)+
010026  104043                  1$:     ERROR   43
010030  000461                          BR      FDONE

010032  020027  010144          2$:     CMP     R0,#FDAT05              ;DID FDST FLOW FAIL TO MODE 3?
010036  001402                          BEQ     3$
010040  000137  040232                  JMP     @#CPSPUR

010044                          3$:                                     ;REPORT FDST FLOW FAILED TO MODE 3.
010044  012737  000643  001246          MOV     #643,@#$TMP6
010052  022626                          CMP     (SP)+,(SP)+
010054  104043                  4$:     ERROR   43
010056  000446                          BR      FDONE

010060                          FERR26:                                 ;REPORT FDST FLOW FAILURE TO MODE
010060  012737  042156  001264          MOV     #MS16,@#$TMP15          ;6 OR MODE 7.
010066  012737  000646  001246          MOV     #646,@#$TMP6
010074  012737  000647  001252          MOV     #647,@#$TMP10
010102  022626                          CMP     (SP)+,(SP)+
010104  104043                  1$:     ERROR   43
010106  000432                          BR      FDONE
```

```
010110  177777          FDATI0: -1
010112  177777          FDATI1: -1
010114  177777          FDATI2: -1
010116  177777          FDATI3: -1
010120  177777          FDATI4: -1
010122  177777          FDATI5: -1
010124  177777          FDATI6: -1
010126  177777          FDATI7: -1
010130  177777                  -1
010132  177777          FDAT00: -1
010134  177777          FDAT01: -1
010136  177777          FDAT02: -1
010140  177777          FDAT03: -1
010142  177777          FDAT04: -1
010144  177777          FDAT05: -1
010146  177777          FDAT06: -1
010150  177777          FDAT07: -1
010152  177777                  -1
010154  177777          FXDAT0: -1
010156  177777          FXDAT1: -1
010160  177777          FXDAT2: -1
010162  177777          FXDAT3: -1
010164  052525          FXDAT4: 052525
010166  031463          FXDAT5: 031463
010170  007417          FXDAT6: 007417
010172  000477          FXDAT7: 000477


010174                  FDONE:
010174  104412                  RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
                                                        ;SEE IF THE USER HAS EXPRESSED
                                                        ;THE DESIRE TO CHANGE THE SOFTWARE
                                                        ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
                                                        ;THE USER TYPED CONTROL G?).


                        ;;**************************************************************
                        ;*TEST 7        FSRC MODE 0 TEST
                        ;*
                        ;*THIS IS A TEST OF FSRC MODE ZERO USING THE LDD AND LDF INSTRUCTIONS.
                        ;*
                        ;;**************************************************************
010176  000004          TST7:   SCOPE
010200  104413                  LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.

010202                  I1:
010202  170011                  SETD                            ;SET FD.
010204  012700  011026          MOV     #IDATI0,R0
010210  012701  010776          MOV     #IPATI0,R1
010214  012702  000004          MOV     #4,R2
010220  012120          I2:     MOV     (R1)+,(R0)+             ;SET UP THE INPUT DATA BUFFER.
010222  077202                  SOB     R2,I2

010224  012700  011026          MOV     #IDATI0,R0              ;LOAD AC1
010230  172510                  LDD     (R0),AC1
```

```
2698  010232  012700  011006          MOV     #IPAT20,R0              ;LOAD AC0
2699  010236  172410                  LDD     (R0),AC0
2700
2701  010240  012701  000001          MOV     #1,R1                   ;IN CASE THE FSRC FLOWS FAIL
2702  010244  012737  010576  000004  MOV     #IERR0,@#ERRVECT        ;AN ODD ADDRESS TRAP TO 4 MAY OCCUR.
2703  010252  012737  010266  001236  MOV     #I3,@#$TMP2
2704  010260  012737  042636  001240  MOV     #MS35,@#$TMP3
2705  010266  172401          I3:     LDD     AC1,AC0                 ;TEST INSTRUCTION.
2706  010270  000240          I4:     NOP
2707  010272  000240          I5:     NOP
2708
2709  010274  012700  011016          MOV     #IDAT00,R0
2710  010300  174010                  STD     AC0,(R0)                ;GET AC0, THE RESULTS.
2711
2712  010302  012700  011016          MOV     #IDAT00,R0              ;SEE IF DATA IS CORRECT.
2713  010306  012701  011026          MOV     #IDAT10,R1
2714  010312  012702  000004          MOV     #4,R2
2715  010316  022021          I6:     CMP     (R0)+,(R1)+
2716  010320  001424                  BEQ     I105
2717
2718  010322  012700  011022          MOV     #IDAT02,R0              ;SEE IF (BUT FD) FAILED.
2719  010326  012702  000002          MOV     #2,R2
2720  010332  005720          I7:     TST     (R0)+
2721  010334  001413                  BEQ     I10
2722
2723  010336  012700  011022          MOV     #IDAT02,R0
2724  010342  012702  000002          MOV     #2,R2
2725  010346  022720  177777  1$:     CMP     #-1,(R0)+
2726  010352  001402                  BEQ     2$
2727  010354  000137  010660          JMP     @#IERR1
2728  010360  077206          2$:     SOB     R2,1$
2729  010362  000401                  BR      I106
2730  010364  077216          I10:    SOB     R2,I7
2731  010366  000137  010700  I106:   JMP     @#IERR2
2732
2733  010372  077227          I105:   SOB     R2,I6
2734
2735                          ;NOW TEST THE LOAD INSTRUCTION WITH FSRC MODE ZERO AND FD CLEAR.
2736
2737  010374                  I11:
2738  010374  104413                  LPERR                           ;SET UP THE LOOP ON ERROR ADDRESS.
2739
2740  010376  012700  010776  I12:     MOV     #IPAT10,R0
2741  010402  012701  011026          MOV     #IDAT10,R1
2742  010406  012702  000004          MOV     #4,R2
2743  010412  012021          I13:    MOV     (R0)+,(R1)+
2744  010414  077202                  SOB     R2,I13
2745
2746  010416  012700  011026          MOV     #IDAT10,R0              ;SET UP AC1
2747  010422  172510                  LDD     (R0),AC1
2748
2749  010424  012700  011006          MOV     #IPAT20,R0              ;SET UP AC0
2750  010430  172410                  LDD     (R0),AC0
2751
2752  010432  012701  000001          MOV     #1,R1
2753  010436  012737  010454  001236  MOV     #I14,@#$TMP2
```

```
2754  010444  012737  042643  001240          MOV     #MS36,@#$TMP3
2755  010452  170001                          SETF                            ;CLEAR FD.
2756
2757  010454  172401                  I14:    LDF     AC1,AC0                 ;TEST INSTRUCTION.
2758  010456  000240                  I15:    NOP
2759  010460  000240                  I16:    NOP
2760
2761  010462  170200                          STFPS   R0                      ;SEE IF FPS IS STILL CLEAR.
2762  010464  022700  000004                  CMP     #4,R0
2763  010470  001402                          BEQ     I17
2764  010472  000137  010752                  JMP     @#IERR3
2765
2766  010476                          I17:                                    ;RESET TO DOUBLE MODE.
2767  010476  170011                          SETD
2768
2769  010500  012700  011016                  MOV     #IDAT00,R0
2770  010504  174010                          STD     AC0,(R0)                ;GET AC0
2771
2772  010506  012737  177777  011032          MOV     #-1,@#IDATI2
2773  010514  012737  177777  011034          MOV     #-1,@#IDATI3
2774  010522  012700  011016                  MOV     #IDAT00,R0
2775  010526  012701  011026                  MOV     #IDATI0,R1
2776  010532  012702  000004                  MOV     #4,R2
2777  010536  022021                  I20:    CMP     (R0)+,(R1)+             ;SEE IF AC0 WAS CORRECT.
2778  010540  001414                          BEQ     I23
2779
2780  010542  023737  011022  011002          CMP     @#IDAT02,@#IPAT12       ;DID (BUT FD) FAIL?
2781  010550  001402                          BEQ     I22
2782  010552  000137  010660          I21:    JMP     @#IERR1
2783  010556  023737  011024  011004  I22:    CMP     @#IDAT03,@#IPAT13
2784  010564  001372                          BNE     I21
2785  010566  000137  010726                  JMP     @#IERR4
2786
2787  010572  077217                  I23:    SOB     R2,I20
2788
2789  010574  000520                          BR      IDONE                   ;NO ERRORS.
2790
2791                                  ;IF AN ODD ADDRESS TRAP OCCURS COME HERE TO ANALYZE THE FSRC FAILURE.
2792  010576  022716  010270          IERR0:  CMP     #I4,(SP)                ;MAKE SURE THE TRAP OCCURRED
2793  010602  001413                          BEQ     1$                      ;ON THE INSTRUCTION BEING TESTED.
2794  010604  022716  010272                  CMP     #I5,(SP)
2795  010610  001410                          BEQ     1$
2796  010612  022716  010456                  CMP     #I15,(SP)
2797  010616  001405                          BEQ     1$
2798  010620  022716  010460                  CMP     #I16,(SP)
2799  010624  001402                          BEQ     1$
2800  010626  000137  040232                  JMP     @#CPSPUR
2801
2802  010632  011637  001236          1$:     MOV     (SP),@#$TMP2            ;REPORT FAILURE.
2803  010636  012737  000627  001240          MOV     #627,@#$TMP3
2804  010644  012737  000320  001242          MOV     #320,@#$TMP4
2805  010652  022626                          CMP     (SP)+,(SP)+
2806  010654  104047                  2$:     ERROR   47
2807  010656  000467                          BR      IDONE
2808
2809                                  ;REPORT DATA ERROR.
```

```
                                        T7      FSRC MODE 0 TEST

 2810  010660                         IERR1:
 2811  010660  012737  011026  001242          MOV     #IDATI0,@#$TMP4
 2812  010666  012737  011016  001244          MOV     #IDAT00,@#$TMP5
 2813  010674  104051                 1$:     ERROR   51
 2814  010676  000457                         BR      IDONE
 2815
 2816                                 ;REPORT FAILURE OF (BUT FD)
 2817  010700  012737  000153  001244 IERR2:  MOV     #153,@#$TMP5
 2818  010706  012737  000434  001246          MOV     #434,@#$TMP6
 2819  010714  012737  000435  001250          MOV     #435,@#$TMP7
 2820  010722                         IERR25:
 2821  010722  104050                 1$:     ERROR   50
 2822  010724  000444                         BR      IDONE
 2823  010726  012737  000153  001244 IERR4:  MOV     #153,@#$TMP5
 2824  010734  012737  000435  001246          MOV     #435,@#$TMP6
 2825  010742  012737  000434  001250          MOV     #434,@#$TMP7
 2826  010750  000764                         BR      IERR25
 2827
 2828                                 ;REPORT INCORRECT FPS AFTER LOAD INSTRUCTION.
 2829  010752                         IERR3:
 2830  010752  012737  010454  001236          MOV     #I14,@#$TMP2
 2831  010760  010037  001240                  MOV     R0,@#$TMP3
 2832  010764  012737  000004  001242          MOV     #4,@#$TMP4
 2833  010772  104041                 1$:     ERROR   41
 2834  010774  000420                         BR      IDONE
 2835
 2836
 2837  010776  000000                 IPAT10: 0
 2838  011000  170360                 IPAT11: 170360
 2839  011002  016161                 IPAT12: 016161
 2840  011004  052525                 IPAT13: 052525
 2841
 2842  011006  177777                 IPAT20: -1
 2843  011010  177777                 IPAT21: -1
 2844  011012  177777                 IPAT22: -1
 2845  011014  177777                 IPAT23: -1
 2846
 2847  011016  000000                 IDAT00: 0
 2848  011020  000000                 IDAT01: 0
 2849  011022  000000                 IDAT02: 0
 2850  011024  000000                 IDAT03: 0
 2851
 2852  011026  000000                 IDATI0: 0
 2853  011030  000000                 IDATI1: 0
 2854  011032  000000                 IDATI2: 0
 2855  011034  000000                 IDATI3: 0
 2856
 2857  011036                         IDONE:
 2858  011036  104412                         RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
 2859                                                                 ;SEE IF THE USER HAS EXPRESSED
 2860                                                                 ;THE DESIRE TO CHANGE THE SOFTWARE
 2861                                                                 ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
 2862                                                                 ;THE USER TYPED CONTROL G?).
 2863
 2864
 2865                                 ;;********************************** **************************
```

```
                                ;*TEST 10       FDST MODE 0 TEST
                                ;*
                                ;*THIS IS A TEST OF THE STORE INSTRUCTIONS, STD AND STF,  WITH FDST MODE 0.
                                ;*
                                ;*************************************************************************
 011040  000004                 TST10:  SCOPE
                                T1:
 011042  104413                         LPERR                          ;SET UP THE LOOP ON ERROR ADDRESS.
 011044  170011                         SETD                                   ;SET FD
 011046  012700  011604                 MOV     #TPAT10,R0
 011052  012701  011634                 MOV     #TDAT10,R1
 011056  012702  000004                 MOV     #4,R2
 011062  012021                 T2:     MOV     (R0)+,(R1)+                    ;SET UP THE INPUT DATA BUFFER.
 011064  077202                         SOB     R2,T2

 011066  012700  011634                 MOV     #TDAT10,R0                     ;LOAD AC0
 011072  172410                         LDD     (R0),AC0

 011074  012700  011614                 MOV     #TPAT20,R0                     ;LOAD AC1
 011100  172510                         LDD     (R0),AC1

 011102  012701  000001                 MOV     #1,R1                          ;IF THE (BUT FDST) FORM FAILS
 011106  012737  011412  000004         MOV     #TERR0,@#ERRVECT               ;AN ODD ADDRESS TRAP COULD RESULT.
 011114  012737  011130  001236         MOV     #T3,@#$TMP2
 011122  012737  042636  001240         MOV     #MS35,@#$TMP3
 011130  174001                 T3:     STD     AC0,AC1
 011132  000240                 T4:     NOP
 011134  000240                 T5:     NOP

 011136  012700  011624                 MOV     #TDAT00,R0
 011142  174110                         STD     AC1,(R0)                       ;GET THE DATA.

 011144  012703  011624                 MOV     #TDAT00,R3                     ;SEE IF THE DATA IS CORRECT.
 011150  012704  011634                 MOV     #TDAT10,R4
 011154  012705  000004                 MOV     #4,R5
 011160  022324                 T6:     CMP     (R3)+,(R4)+
 011162  001413                         BEQ     T105

 011164  012703  011630                 MOV     #TDAT02,R3                     ;DID (BUT FD) FAIL?
 011170  012705  000002                 MOV     #2,R5
 011174  005723                 T7:     TST     (R3)+
 011176  001402                         BEQ     T10
 011200  000137  011474                 JMP     @#TERR1
 011204  077505                 T10:    SOB     R5,T7
 011206  000137  011514                 JMP     @#TERR2

 011212  077516                 T105:   SOB     R5,T6

                                ;NOW TEST THE STF AC0,AC1 INSTRUCTION.

 011214                         T11:
 011214  104413                         LPERR                          ;SET UP THE LOOP ON ERROR ADDRESS.

 011216  012700  011604         T12:    MOV     #TPAT10,R0                     ;SET UP THE INPUT DATA BUFFER.
 011222  012701  011634                 MOV     #TDAT10,R1
 011226  012702  000004                 MOV     #4,R2
```

```
                                        T10     FDST MODE 0 TEST

011232  012021                          T13:    MOV     (R0)+,(R1)+
011234  077202                                  SOB     R2,T13

011236  012700  011634                          MOV     #TDATI0,R0              ;SET UP AC0
011242  172410                                  LDD     (R0),AC0

011244  012700  011614                          MOV     #TPAT20,R0              ;SET UP AC1
011250  172510                                  LDD     (R0),AC1

011252  012701  000001                          MOV     #1,R1
011256  012737  011274  001236                  MOV     #T14,@#STMP2
011264  012737  042643  001240                  MOV     #^M536,@#STMP3
011272  170001                                  SETF                            ;CLEAR FD
011274  174001                          T14:    STF     AC0,AC1
011276  000240                          T15:    NOP
011300  000240                          T16:    NOP

011302  005000                                  CLR     R0
011304  170200                                  STFPS   R0                      ;SEE IF FPS IS CLEAR.
011306  022700  000010                          CMP     #10,R0
011312  001401                                  BEQ     T17
011314  000521                                  BR      TERR3

011316                                  T17:
011316  170011                                  SETD                            ;SET FD.

011320  012700  011624                          MOV     #TDAT00,R0
011324  174110                                  STD     AC1,(R0)                ;PICK UP AC1.

011326  012737  177777  011640                  MOV     #-1,@#TDATI2
011334  012737  177777  011642                  MOV     #-1,@#TDATI3
011342  012703  011624                          MOV     #TDAT00,R3
011346  012704  011634                          MOV     #TDATI0,R4
011352  012705  000004                          MOV     #4,R5
011356  022324                          T20:    CMP     (R3)+,(R4)+             ;WAS THE DATA TRANSFERRED CORRECTLY?
011360  001412                                  BEQ     T23

011362  023737  011630  011610                  CMP     @#TDAT02,@#TPAT12       ;DID (BUT FD) FAIL.
011370  001401                                  BEQ     T22
011372  000440                          T21:    BR      TERR1
011374  023737  011632  011612          T22:    CMP     @#TDAT03,@#TPAT13
011402  301373                                  BNE     T21
011404  000456                                  BR      TERR4

011406  077515                          T23:    SOB     R5,T20
011410  000515                                  BR      TDONE


                                        ;TRAP HERE THROUGH VECTOR 4 IF AN ODD ADDRESS OCCURS.
011412  022716  011132                  TERR0:  CMP     #T4,(SP)                ;MAKE SURE THE TRAP WAS ON
011416  001413                                  BEQ     1$                      ;AN INSTRUCTION BEING TESTED.
011420  022716  011134                          CMP     #T5,(SP)
011424  001410                                  BEQ     1$
011426  022716  011276                          CMP     #T15,(SP)
011432  001405                                  BEQ     1$
011434  022716  011300                          CMP     #T16, SP
```

```
011440  001402                                   BEQ     1$
011442  000137  040232                           JMP     @#CPSPUR

011446  011637  001236          1$:      MOV     (SP),@#$TMP2
011452  022626                                   CMP     (SP)+,(SP)+
011454  012737  000527  001240           MOV     #527,@#$TMP3
011462  012737  000640  001242           MOV     #640,@#$TMP4
011470  104121                          2$:      ERROR   121
011472  000464                                   BR      TDONE

                                        ;REPORT DATA FAILURE.
011474                                  TERR1:
011474  012737  011634  001242           MOV     #TDATI0,@#$TMP4
011502  012737  011624  001244           MOV     #TDAT00,@#$TMP5
011510  104123                          1$:      ERROR   123
011512  000454                                   BR      TDONE

                                        ;REPORT FAILURE OF (BUT FD).
011514  012737  000160  001246  TERR2:  MOV     #160,@#$TMP6
011522  012737  000161  001250           MOV     #161,@#$TMP7
011530  012737  000640  001244  TERR25: MOV     #640,@#$TMP5
011536  104122                          1$:      ERROR   122
011540  000441                                   BR      TDONE
011542  012737  000161  001246  TERR4:  MOV     #161,@#$TMP5
011550  012737  000160  001250           MOV     #160,@#$TMP7
011556  000764                                   BR      TERR25

                                        ;REPORT INCORRECT FPS AFTER STORE INSTRUCTION.
011560                                  TERR3:
011560  012737  011276  001236           MOV     #T15,@#$TMP2
011566  010037  001240                   MOV     R0,@#$TMP3
011572  012737  000010  001242           MOV     #10,@#$TMP4
011600  104041                          1$:      ERROR   41
011602  000420                                   BR      TDONE

011604  000000                          TPAT10: 0
011606  170360                          TPAT11: 170360
011610  016161                          TPAT12: 016161
011612  052525                          TPAT13: 052525

011614  177777                          TPAT20: -1
011616  177777                          TPAT21: -1
011620  177777                          TPAT22: -1
011622  177777                          TPAT23: -1

011624  000000                          TDAT00: 0
011626  000000                          TDAT01: 0
011630  000000                          TDAT02: 0
011632  000000                          TDAT03: 0

011634  000000                          TDAT10: 0
011636  000000                          TDAT11: 0
011640  000000                          TDAT12: 0
011642  000000                          TDAT13: 0

011644                                  TDONE:
```

E05


```
014644  104412          RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
                                                ;SEE IF THE USER HAS EXPRESSED
                                                ;THE DESIRE TO CHANGE THE SOFTWARE
                                                ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
                                                ;THE USER TYPED CONTROL G?).



                        ;;*************************************************************
                        ;*TEST 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 INTERMITTANT THEN THIS PROCEDURE WILL STILL
;*APPLY!!
;*IF THE FAILURE MOVES FROM ONE BIT TO ANOTHER, OR FROM ONE
;*GROUP OF BITS TO ANOTHER GROUP OF BITS THEN THE FAULT WILL
;*PROBABLY NOT SHOW UP IN THE 'AND' OR THE 'OR' PATTERNS. IN THIS
;*CASE THE 'AND' PATTERN WILL BE ALL 0'S AND THE 'OR' PATTERN WILL
;*BE ALL 1'S. WHEN THIS OCCURS SOME OTHER METHOD OF REPAIR MUST
;*BE FOUND (SUCH AS INSPECTION OF EACH INDIVIDUAL ERROR REPORT
;*RATHER THAN USING THE SUMMARY).
;*
;*MAP THE FOLLOWING NOTATION ONTO EACH BIT POSITION IN THE 'AND'
;*AND THE 'OR' PATTERNS WHICH ARE TYPED IN THE ERROR SUMMARY.
;*
;*A15,A14,...A1,A0  B15,B14,...B1,B0  C15,C14,...C1,C0  D15,D14,...D1,D0
;*
;*IN THIS NOTATION A15 THROUGH A0 IS THE FIRST OF THE FOUR 16 BIT
;*OCTAL NUMBERS TYPED, B15 THROUGH B0 IS THE SECOND, ETC.
;*
;*THIS TABLE SHOWS THE CORRESPONDING AM2901 CHIP ('E' NUMBER)
;*WHICH IS RESPONSIBLE FOR EACH BIT POSITION USING THE ABOVE
;*NOTATION. NOTE THAT ECO'S TO THE HARDWARE MIGHT MAKE THIS
;*TABLE OBSOLETE IF IT IS NOT UP DATED. NOTE ALSO THAT THERE ARE
;*FOUR BITS FOR EACH AM2901 CHIP:
;*
;*      BITS                            AM2901 CHIP NUMBER
;*      ----                            ------------------
;*
;*      A15,A14,A13,A12                 E61
;*      A11,A10,A9,A8                   E62
;*      A7,A6,A5,A4                     E90
;*      A3,A2,A1,A0                     E81
;*
;*      B15,B14,B13,B12                 E86
;*      B11,B10,B9,B8                   E85
;*      B7,B6,B5,B4                     E83
;*      B3,B2,B1,B0                     E88
;*
;*      C15,C14,C13,C12                 E79
;*      C11,C10,C9,C8                   E84
;*      C7,C6,C5,C4                     E89
;*      C3,C2,C1,C0                     E87
;*
;*      D15,D14,D13,D12                 E78
;*      D11,D10,D9,D8                   E77
;*      D7,D6,D5,D4                     E82
;*      D3,D2,D1,D0                     E80
;*
;*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
```

```
;*                                   ANDN IS 15, 14, ... OR 0
;*      THEN IN GENERAL ANY OF THE FOUR CHIPS RESPONSIBLE FOR
;*      AN, BN, CN OR DN COULD BE AT FAULT, WITH LN BEING MOST PROBABLE
;*      FOR EXAMPLE IF BIT C12 IS FAULTY, THEN CHIP E79
;*      IS THE MOST PROBABLE SOURCE OF THE ERROR. IF REPAIRING
;*      THAT CHIP DOES NOT REMOVE THE FAULT THEN TRY EACH OF THE
;*      CHIPS ASSOCIATED WITH BITS A12, B12 AND D12 SHOULD BE TRIED
;*      WITH EQUAL PROBABILITY OF THE FAULT BEING
;*      IN ANY ONE OF THESE OTHER THREE CHIPS, TRY CHIPS E61, E86 AND E78.
;*2.)   IF THERE ARE FOUR CONSECUTIVE BITS IN ERROR, FOLLOWING THE
;*      PATTERN:
;*              LN, LN+1, LN+2 AND LN+3         WHERE 'L' IS A, B, C
;*                                              OR D.
;*                                              AND N=0,4,8 OR 12
;*      THEN THE ABOVE TABLE SHOULD DIRECTLY IDENTIFY THE
;*      FAILING CHIP.
;*
;*3.)   IF FOUR BITS ARE DROPPED WHICH FIT THE PATTERN:
;*              AN, BN, CN AND DN       WHERE N=15,14,... OR 0
;*      THEN ANY ONE OF THE FOUR CHIPS ASSOCIATED WITH EACH OF
;*      THE BITS AN, BN, CN AND DN COULD BE AT FAULT WITH
;*      EQUAL PROBABILITY.
;*
;*4.    IF 16 BITS ARE IN ERROR, FITTING THE PATTERN:
;*              AN, AN+1, AN+2, AN+3    WHERE N=0,4,8 OR 12
;*              BN, BN+1, BN+2, BN+3
;*              CN, CN+1, CN+2, CN+3
;*                      AND
;*              DN, DN+1, DN+2, AN+3
;*      THEN ANY ONE OF THE FOUR CHIPS ASSOCIATED
;*      WITH THESE BITS COULD BE AT FAULT WITH EQUAL PROBABILITY.
;*
;*5.)   IF THE FAILING BIT PATTERNS DISPLAYED IN THE 'AND' AND THE 'OR'
;*      DATA TYPED IN THE SUMMARY DOES NOT CONFORM EXPLICITELY TO
;*      ANY OF THE ABOVE PATTERNS, THEN THE TROUBLE SHOOTER MUST
;*      INTUITIVELY TRY TO FIND WHICH OF THE ABOVE CASES (1 THROUGH 4
;*      IS A 'BEST FIT' OF THE SYMPTOMS.
;*
;*************************************************************
ST11:   SCOPE
        SETD                                    ;SET FD.
;TEST ACCUMULATOR 0 WITH FLOATING ONE
        MOV     #MNUM0,@#$TMP5
        MOV     #G1,@#$TMP2
        MOV     #GPAT00,R0
        MOV     #GDAT00,R1
        LPERR                           ;SET UP THE LOOP ON ERROR ADDRESS.
        JSR     PC,@#GSETUP             ;LOAD TEST PATTERN.
        MOV     #102,R3
G1:
        LDD     (R0),AC0
        STD     AC0,AC0
        LDD     AC0,AC0                 ;STORE THE TEST PATTERN.
        STD     AC0,(R1)
        JSR     PC,@#GCMP               ;COMPARE THE DATA READ WITH
                                        ;THAT WHICH WAS WRITTEN.
```

```
011646  000004
011650  170011
011652  012737  042210  001244
011660  012737  011710  001236
011666  012700  014130
011672  012701  014170
011676  104413
011700  004737  013606
011704  012703  000102
011710
011710  172410
011712  174000
011714  172400
011716  174011
011720  004737  013704
```

```
011724  005737  014124                TST     @#GFLAG1
011730  001004                        BNE     G2
011732  005137  014124                COM     @#GFLAG1
011736  000261                        SEC
011740  000401                        BR      G3
011742  000241                G2:     CLC
011744  006160  000006        G3:     ROL     6(R0)                   ;GENERATE THE NEXT TEST PATTERN.
011750  006160  000004                ROL     4(R0)
011754  006160  000002                ROL     2(R0)
011760  006110                        ROL     (R0)
011762  004737  013664                JSR     PC,@#GRESET             ;RESET DEFAULT PATTERN IN OUTPUT
                                                                      ;BUFFER.
011766  077330                        SOB     R3,G1
011770  004737  014022                JSR     PC,@#GSUM               ;TYPE ERROR SUMMARY.

                              ;TEST ACCUMULATOR 0 WITH FLOATING ZERO
011774  012737  042210  001244        MOV     #MNUM0,@#$TMP5
012002  012737  012032  001236        MOV     #G4,@#$TMP2
012010  012700  014140                MOV     #GPAT10,R0
012014  012701  014170                MOV     #GDAT00,R1
012020  104413                        LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
012022  004737  013606                JSR     PC,@#GSETUP             ;LOAD TEST PATTERN.
012026  012703  000102                MOV     #102,R3
012032                        G4:
012032  172410                        LDD     (R0),AC0
012034  174000                        STD     AC0,AC0
012036  172400                        LDD     AC0,AC0         ;STORE THE TEST PATTERN.
012040  174011                        STD     AC0,(R1)
012042  004737  013704                JSR     PC,@#GCMP               ;COMPARE THE DATA READ WITH
                                                                      ;THAT WHICH WAS WRITTEN.
012046  005737  014124                TST     @#GFLAG1
012052  001004                        BNE     G5
012054  005137  014124                COM     @#GFLAG1
012060  000241                        CLC
012062  000401                        BR      G6
012064  000261                G5:     SEC
012066  006160  000006        G6:     ROL     6(R0)                   ;GENERATE THE NEXT TEST PATTERN.
012072  006160  000004                ROL     4(R0)
012076  006160  000002                ROL     2(R0)
012102  006110                        ROL     (R0)
012104  004737  013664                JSR     PC,@#GRESET             ;RESET DEFAULT PATTERN IN OUTPUT
                                                                      ;BUFFER.
012110  077330                        SOB     R3,G4
012112  004737  014022                JSR     PC,@#GSUM               ;TYPE ERROR SUMMARY.

                              ;TEST ACCUMULATOR 1 WITH FLOATING ONE
012116  012737  042216  001244        MOV     #MNUM1,@#$TMP5
012124  012737  012154  001236        MOV     #G7,@#$TMP2
012132  012700  014130                MOV     #GPAT00,R0
012136  012701  014170                MOV     #GDAT00,R1
012142  104413                        LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
012144  004737  013606                JSR     PC,@#GSETUP             ;LOAD TEST PATTERN.
012150  012703  000102                MOV     #102,R3
012154                        G7:
012154  172410                        LDD     (R0),AC0
012156  174001                        STD     AC0,AC1
```

```
                        ;11      ACCUMULATORS DATA PATTERNS TEST

012160  172401                        LDD     AC1,AC0         ;STORE THE TEST PATTERN.
012162  174011                        STD     AC0,(R1)
012164  004737  013704                JSR     PC,@#GCMP               ;COMPARE THE DATA READ WITH
                                                                ;THAT WHICH WAS WRITTEN.
012170  005737  014124                TST     @#GFLAG1
012174  001004                        BNE     G10
012176  005137  014124                COM     @#GFLAG1
012202  000261                        SEC
012204  000401                        BR      G11
012206  000241                G10:    CLC
012210  006160  000006        G11:    ROL     6(R0)                   ;GENERATE THE NEXT TEST PATTERN.
012214  006160  000004                ROL     4(R0)
012220  006160  000002                ROL     2(R0)
012224  006110                        ROL     (R0)
012226  004737  013664                JSR     PC,@#GRESET             ;RESET DEFAULT PATTERN IN OUTPUT
                                                                ;BUFFER.
012232  077330                        SOB     R3,G7
012234  004737  014022                JSR     PC,@#GSUM               ;TYPE ERROR SUMMARY.

                        ;TEST ACCUMULATOR 1 WITH FLOATING ZERO
012240  012737  042216  001244        MOV     #MNUM1,@#$TMP5
012246  012737  012276  001236        MOV     #G12,@#$TMP2
012254  012700  014140                MOV     #GPAT10,R0
012260  012701  014170                MOV     #GDAT00,R1
012264  104413                        LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
012266  004737  013606                JSR     PC,@#GSETUP             ;LOAD TEST PATTERN.
012272  012703  000102                MOV     #102,R3
012276                          G12:
012276  172410                        LDD     (R0),AC0
012300  174001                        STD     AC0,AC1
012302  172401                        LDD     AC1,AC0         ;STORE THE TEST PATTERN.
012304  174011                        STD     AC0,(R1)
012306  004737  013704                JSR     PC,@#GCMP               ;COMPARE THE DATA READ WITH
                                                                ;THAT WHICH WAS WRITTEN.
012312  005737  014124                TST     @#GFLAG1
012316  001004                        BNE     G13
012320  005137  014124                COM     @#GFLAG1
012324  000241                        CLC
012326  000401                        BR      G14
012330  000261                G13:    SEC
012332  006160  000006        G14:    ROL     6(R0)                   ;GENERATE THE NEXT TEST PATTERN.
012336  006160  000004                ROL     4(R0)
012342  006160  000002                ROL     2(R0)
012346  006110                        ROL     (R0)
012350  004737  013664                JSR     PC,@#GRESET             ;RESET DEFAULT PATTERN IN OUTPUT
                                                                ;BUFFER.
012354  077330                        SOB     R3,G12
012356  004737  014022                JSR     PC,@#GSUM               ;TYPE ERROR SUMMARY.

                        ;TEST ACCUMULATOR 2 WITH FLOATING ONE
012362  012737  042223  001244        MOV     #MNUM2,@#$TMP5
012370  012737  012420  001236        MOV     #G15,@#$TMP2
012376  012700  014130                MOV     #GPAT00,R0
012402  012701  014170                MOV     #GDAT00,R1
012406  104413                        LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
012410  004737  013606                JSR     PC,@#GSETUP             ;LOAD TEST PATTERN.
```

```
012414  012703  000102                  MOV     #102,R3
012420                          G15:
012420  172410                          LDD     (R0),AC0
012422  174002                          STD     AC0,AC2
012424  172402                          LDD     AC2,AC0         ;STORE THE TEST PATTERN.
012426  174011                          STD     AC0,(R1)
012430  004737  013704                  JSR     PC,@#GCMP               ;COMPARE THE DATA READ WITH
                                                                        ;THAT WHICH WAS WRITTEN.
012434  005737  014124                  TST     @#GFLAG1
012440  001004                          BNE     G16
012442  005137  014124                  COM     @#GFLAG1
012446  000261                          SEC
012450  000401                          BR      G17
012452  000241                  G16:    CLC
012454  006160  000006          G17:    ROL     6(R0)                   ;GENERATE THE NEXT TEST PATTERN.
012460  006160  000004                  ROL     4(R0)
012464  006160  000002                  ROL     2(R0)
012470  006110                          ROL     (R0)
012472  004737  013664                  JSR     PC,@#GRESET             ;RESET DEFAULT PATTERN IN OUTPUT
                                                                        ;BUFFER.
012476  077330                          SOB     R3,G15
012500  004737  014022                  JSR     PC,@#GSUM               ;TYPE ERROR SUMMARY.

                                ;TEST ACCUMULATOR 2 WITH FLOATING ZERO
012504  012737  042223  001244          MOV     #MNUM2,@#$TMPS
012512  012737  012542  001236          MOV     #G20,@#$TMP2
012520  012700  014140                  MOV     #GPAT10,R0
012524  012701  014170                  MOV     #GDAT00,R1
012530  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
012532  004737  013606                  JSR     PC,@#GSETUP             ;LOAD TEST PATTERN.
012536  012703  000102                  MOV     #102,R3
012542                          G20:
012542  172410                          LDD     (R0),AC0
012544  174002                          STD     AC0,AC2
012546  172402                          LDD     AC2,AC0         ;STORE THE TEST PATTERN.
012550  174011                          STD     AC0,(R1)
012552  004737  013704                  JSR     PC,@#GCMP               ;COMPARE THE DATA READ WITH
                                                                        ;THAT WHICH WAS WRITTEN.
012556  005737  014124                  TST     @#GFLAG1
012562  001004                          BNE     G21
012564  005137  014124                  COM     @#GFLAG1
012570  000241                          CLC
012572  000401                          BR      G22
012574  000261                  G21:    SEC
012576  006160  000006          G22:    ROL     6(R0)                   ;GENERATE THE NEXT TEST PATTERN.
012602  006160  000004                  ROL     4(R0)
012606  006160  000002                  ROL     2(R0)
012612  006110                          ROL     (R0)
012614  004737  013664                  JSR     PC,@#GRESET             ;RESET DEFAULT PATTERN IN OUTPUT
                                                                        ;BUFFER.
012620  077330                          SOB     R3,G20
012622  004737  014022                  JSR     PC,@#GSUM               ;TYPE ERROR SUMMARY.

                                ;TEST ACCUMULATOR 3 WITH FLOATING ONE
012626  012737  042230  001244          MOV     #MNUM3,@#$TMPS
012634  012737  012664  001236          MOV     #G23,@#$TMP2
```

```
3370  012642  012700  014130          MOV     #GPAT00,R0
3371  012646  012701  014170          MOV     #GDAT00,R1
3372  012652  104413                  LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
3373  012654  004737  013606          JSR     PC,@#GSETUP             ;LOAD TEST PATTERN.
3374  012660  012703  000102          MOV     #102,R3
3375  012664                  G23:
3376  012664  172410                  LDD     (R0),AC0
3377  012666  174003                  STD     AC0,AC3
3378  012670  172403                  LDD     AC3,AC0         ;STORE THE TEST PATTERN.
3379  012672  174011                  STD     AC0,(R1)
3380  012674  004737  013704          JSR     PC,@#GCMP               ;COMPARE THE DATA READ WITH
3381                                                           ;THAT WHICH WAS WRITTEN.
3382  012700  005737  014124          TST     @#GFLAG1
3383  012704  001004                  BNE     G24
3384  012706  005137  014124          COM     @#GFLAG1
3385  012712  000261                  SEC
3386  012714  000401                  BR      G25
3387  012716  000241          G24:    CLC
3388  012720  006160  000006  G25:    ROL     6(R0)                   ;GENERATE THE NEXT TEST PATTERN.
3389  012724  006160  000004          ROL     4(R0)
3390  012730  006160  000002          ROL     2(R0)
3391  012734  006110                  ROL     (R0)
3392  012736  004737  013664          JSR     PC,@#GRESET             ;RESET DEFAULT PATTERN IN OUTPUT
3393                                                           ;BUFFER.
3394  012742  077330                  SOB     R3,G23
3395  012744  004737  014022          JSR     PC,@#GSUM               ;TYPE ERROR SUMMARY.
3396
3397                  ;TEST ACCUMULATOR 3 WITH FLOATING ZERO
3398  012750  012737  042230  001244  MOV     #MNUM3,@#$TMP5
3399  012756  012737  013006  001236  MOV     #G26,@#$TMP2
3400  012764  012700  014140          MOV     #GPAT10,R0
3401  012770  012701  014170          MOV     #GDAT00,R1
3402  012774  104413                  LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
3403  012776  004737  013606          JSR     PC,@#GSETUP             ;LOAD TEST PATTERN.
3404  013002  012703  000102          MOV     #102,R3
3405  013006                  G26:
3406  013006  172410                  LDD     (R0),AC0
3407  013010  174003                  STD     AC0,AC3
3408  013012  172403                  LDD     AC3,AC0         ;STORE THE TEST PATTERN.
3409  013014  174011                  STD     AC0,(R1)
3410  013016  004737  013704          JSR     PC,@#GCMP               ;COMPARE THE DATA READ WITH
3411                                                           ;THAT WHICH WAS WRITTEN.
3412  013022  005737  014124          TST     @#GFLAG1
3413  013026  001004                  BNE     G27
3414  013030  005137  014124          COM     @#GFLAG1
3415  013034  000241                  CLC
3416  013036  000401                  BR      G30
3417  013040  000261          G27:    SEC
3418  013042  006160  000006  G30:    ROL     6(R0)                   ;GENERATE THE NEXT TEST PATTERN.
3419  013046  006160  000004          ROL     4(R0)
3420  013052  006160  000002          ROL     2(R0)
3421  013056  006110                  ROL     (R0)
3422  013060  004737  013664          JSR     PC,@#GRESET             ;RESET DEFAULT PATTERN IN OUTPUT
3423                                                           ;BUFFER.
3424  013064  077330                  SOB     R3,G26
3425  013066  004737  014022          JSR     PC,@#GSUM               ;TYPE ERROR SUMMARY.
```

```
3426
3427                                        ;TEST ACCUMULATOR 4 WITH FLOATING ONE
3428  013072  012737  042237  001244        MOV     #MNUM4,@#$TMP5
3429  013100  012737  013130  001236        MOV     #G31,@#$TMP2
3430  013106  012700  014130                MOV     #GPAT00,R0
3431  013112  012701  014170                MOV     #GDAT00,R1
3432  013116  104413                        LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
3433  013120  004737  013606                JSR     PC,@#GSETUP             ;LOAD TEST PATTERN.
3434  013124  012703  000102                MOV     #102,R3
3435  013130                          G31:
3436  013130  172410                        LDD     (R0),AC0
3437  013132  174004                        STD     AC0,AC4
3438  013134  172404                        LDD     AC4,AC0         ;STORE THE TEST PATTERN.
3439  013136  174011                        STD     AC0,(R1)
3440  013140  004737  013704                JSR     PC,@#GCMP               ;COMPARE THE DATA READ WITH
3441                                                                  ;THAT WHICH WAS WRITTEN.
3442  013144  005737  014124                TST     @#GFLAG1
3443  013150  001004                        BNE     G32
3444  013152  005137  014124                COM     @#GFLAG1
3445  013156  000261                        SEC
3446  013160  000401                        BR      G33
3447  013162  000241                 G32:   CLC
3448  013164  006160  000006           G33:  ROL     6(R0)                   ;GENERATE THE NEXT TEST PATTERN.
3449  013170  006160  000004                ROL     4(R0)
3450  013174  006160  000002                ROL     2(R0)
3451  013200  006110                        ROL     (R0)
3452  013202  004737  013664                JSR     PC,@#GRESET             ;RESET DEFAULT PATTERN IN OUTPUT
3453                                                                  ;BUFFER.
3454  013206  077330                        SOB     R3,G31
3455  013210  004737  014022                JSR     PC,@#GSUM               ;TYPE ERROR SUMMARY.
3456
3457                                        ;TEST ACCUMULATOR 4 WITH FLOATING ZERO
3458  013214  012737  042237  001244        MOV     #MNUM4,@#$TMP5
3459  013222  012737  013252  001236        MOV     #G34,@#$TMP2
3460  013230  012700  014140                MOV     #GPAT10,R0
3461  013234  012701  014170                MOV     #GDAT00,R1
3462  013240  104413                        LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
3463  013242  004737  013606                JSR     PC,@#GSETUP             ;LOAD TEST PATTERN.
3464  013246  012703  000102                MOV     #102,R3
3465  013252                          G34:
3466  013252  172410                        LDD     (R0),AC0
3467  013254  174004                        STD     AC0,AC4
3468  013256  172404                        LDD     AC4,AC0         ;STORE THE TEST PATTERN.
3469  013260  174011                        STD     AC0,(R1)
3470  013262  004737  013704                JSR     PC,@#GCMP               ;COMPARE THE DATA READ WITH
3471                                                                  ;THAT WHICH WAS WRITTEN.
3472  013266  005737  014124                TST     @#GFLAG1
3473  013272  001004                        BNE     G35
3474  013274  005137  014124                COM     @#GFLAG1
3475  013300  000241                        CLC
3476  013302  000401                        BR      G36
3477  013304  000261                 G35:   SEC
3478  013306  006160  000006           G36:  ROL     6(R0)                   ;GENERATE THE NEXT TEST PATTERN.
3479  013312  006160  000004                ROL     4(R0)
3480  013316  006160  000002                ROL     2(R0)
3481  013322  006110                        ROL     (R0)
```

```
3482  013324  004737  013664                  JSR     PC,@#GRESET             ;RESET DEFAULT PATTERN IN OUTPUT
3483                                                                          ;BUFFER.
3484  013330  077330                          SOB     R3,G34
3485  013332  004737  014022                  JSR     PC,@#GSUM               ;TYPE ERROR SUMMARY.
3486
3487                          ;TEST ACCUMULATOR 5 WITH FLOATING ONE
3488  013336  012737  042245  001244          MOV     #MNUM5,@#$TMP5
3489  013344  012737  013374  001236          MOV     #G37,@#$TMP2
3490  013352  012700  014130                  MOV     #GPAT00,R0
3491  013356  012701  014170                  MOV     #GDAT00,R1
3492  013362  104413                          LPERR                           ;SET UP THE LOOP ON ERROR ADDRESS.
3493  013364  004737  013606                  JSR     PC,@#GSETUP             ;LOAD TEST PATTERN.
3494  013370  012703  000102                  MOV     #102,R3
3495  013374                          G37:
3496  013374  172410                          LDD     (R0),AC0
3497  013376  174005                          STD     AC0,AC5
3498  013400  172405                          LDD     AC5,AC0                 ;STORE THE TEST PATTERN.
3499  013402  174011                          STD     AC0,(R1)
3500  013404  004737  013704                  JSR     PC,@#GCMP               ;COMPARE THE DATA READ WITH
3501                                                                          ;THAT WHICH WAS WRITTEN.
3502  013410  005737  014124                  TST     @#GFLAG1
3503  013414  001004                          BNE     G40
3504  013416  005137  014124                  COM     @#GFLAG1
3505  013422  000261                          SEC
3506  013424  000401                          BR      G41
3507  013426  000241                  G40:    CLC
3508  013430  006160  000006          G41:    ROL     6(R0)                   ;GENERATE THE NEXT TEST PATTERN.
3509  013434  006160  000004                  ROL     4(R0)
3510  013440  006160  000002                  ROL     2(R0)
3511  013444  006110                          ROL     (R0)
3512  013446  004737  013664                  JSR     PC,@#GRESET             ;RESET DEFAULT PATTERN IN OUTPUT
3513                                                                          ;BUFFER.
3514  013452  077330                          SOB     R3,G37
3515  013454  004737  014022                  JSR     PC,@#GSUM               ;TYPE ERROR SUMMARY.
3516
3517                          ;TEST ACCUMULATOR 5 WITH FLOATING ZERO
3518  013460  012737  042245  001244          MOV     #MNUM5,@#$TMP5
3519  013466  012737  013516  001236          MOV     #G42,@#$TMP2
3520  013474  012700  014140                  MOV     #GPAT10,R0
3521  013500  012701  014170                  MOV     #GDAT00,R1
3522  013504  104413                          LPERR                           ;SET UP THE LOOP ON ERROR ADDRESS.
3523  013506  004737  013606                  JSR     PC,@#GSETUP             ;LOAD TEST PATTERN.
3524  013512  012703  000102                  MOV     #102,R3
3525  013516                          G42:
3526  013516  172410                          LDD     (R0),AC0
3527  013520  174005                          STD     AC0,AC5
3528  013522  172405                          LDD     AC5,AC0                 ;STORE THE TEST PATTERN.
3529  013524  174011                          STD     AC0,(R1)
3530  013526  004737  013704                  JSR     PC,@#GCMP               ;COMPARE THE DATA READ WITH
3531                                                                          ;THAT WHICH WAS WRITTEN.
3532  013532  005737  014124                  TST     @#GFLAG1
3533  013536  001004                          BNE     G43
3534  013540  005137  014124                  COM     @#GFLAG1
3535  013544  000241                          CLC
3536  013546  000401                          BR      G44
3537  013550  000261                  G43:    SEC
```

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

```
	013754	010003				MOV	R0,R3			;COMPUTE 'AND' OF BAD DATA.
	013756	012705	014150			MOV	#GAND0,R5
	013762	012704	000004			MOV	#4,R4
	013766	012302			2$:	MOV	(R3)+,R2
	013770	005102				COM	R2
	013772	040225				BIC	R2,(R5)+
	013774	077404				SOB	R4,2$

	013776	005237	014126			INC	@#GFLAG2		;INCREMENT ERROR COUNT.
	014002	010037	001240			MOV	R0,@#$TMP3
	014006	012737	014170	001242		MOV	#GDAT00,@#$TMP4
	014014	104044			3$:	ERROR	44
	014016	000177	000156			JMP	@GADR

						;SEE IF ANY ERRORS HAVE OCCURRED AND WHETHER OR NOT AN ERROR SUMMAR
						;SHOULD BE TYPED.
	014022	005737	014126		GSUM:	TST	@#GFLAG2		;ANY ERRORS?
	014026	001435				BEQ	3$

	014030	032777	020000	165102		BIT	#SW13,@SWR		;INHIBIT ERROR PRINT OUT?
	014036	001404				BEQ	1$
	014040	032777	000200	165072		BIT	#SW7,@SWR		;PRINT SUMMARY?
	014046	001425				BEQ	3$

	014050	013737	014126	001246	1$:	MOV	@#GFLAG2,@#$TMP6	;YES PRINT SUMMARY.
	014056	012737	014150	001240		MOV	#GAND0,@#$TMP3
	014064	012737	014160	001242		MOV	#GOR0,@#$TMP4
	014072	012637	014200			MOV	(SP)+,@#GADR		;SAVE RETURN ADDRESS.
	014076	012737	014112	001116		MOV	#2$,@#$ERRPC
	014104	112737	000045	001114		MOVB	#45,@#$ITEMB
	014112	004737	027546		2$:	JSR	PC,@#ERTYPE
	014116	000177	000056			JMP	@GADR
	014122	000207			3$:	RTS	PC


	014124	000000			GFLAG1:	0
	014126	000000			GFLAG2:	0

	014130	000000			GPAT00:	0
	014132	000000			GPAT01:	0
	014134	000000			GPAT02:	0
	014136	000000			GPAT03:	0

	014140	177777			GPAT10:	-1
	014142	177777			GPAT11:	-1
	014144	177777			GPAT12:	-1
	014146	177777			GPAT13:	-1

	014150	177777			GAND0:	-1
	014152	177777			GAND1:	-1
	014154	177777			GAND2:	-1
	014156	177777			GAND3:	-1

	014160	000000			GOR0:	0
	014162	000000			GOR1:	0
```

```
                        ;11     ACCUMULATORS DATA PATTERNS TEST

014186  000000          $DR2:   0
014188  000000          $DR3:   0

014190  000000          $DAT00: 0
014192  000000          $DAT01: 0
014194  000000          $DAT02: 0
014196  000000          $DAT03: 0

014200  000000          $ADR:   0

014202                  $DONE:
014202  104412                  RESETF                  ;GO INITIALIZE THE FPS AND STACK; AND
                                                        ;SEE IF THE USER HAS EXPRESSED
                                                        ;THE DESIRE TO CHANGE THE SOFTWARE
                                                        ;VIRTUAL CONSOLE SWITCH REGISTER  AND
                                                        ;THE USER TYPED CONTROL G?.


                        ;;*****************************************************************
                        ;*TEST 12       FPP ACCUMULATORS DUAL ADDRESS TEST
                        ;*
                        ;*THIS TEST PERFORMS A DUAL ADDRESSING TEST ON THE FLOATING ACCUMULATORS.
                        ;*NOTE THAT ACCUMULATOR ZERO IS USED TO ACCESS ALL THE OTHERS.
                        ;*
                        ;;*****************************************************************
014204  000304          TST12:  SCOPE
014206  104413                  LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.

014210  005037  014734  H1:      CLR     @#HFLAG
014214  012700  014736          MOV     #HA1W,R0                ;INITIALIZE THE LOAD BUFFER DATA.
014220  012701  015056          MOV     #HDAT1,R1
014224  012703  000024          MOV     #24,R3
014230  012120  H2:      MOV     (R1)+,(R0)+
014232  077302          SOB     R3,H2

014234  004767  000422          JSR     PC,HCLR         ;CLEAR THE OUTPUT DATA BUFFER.

014240  170011          H3:     SETD
                        ;LOAD ACCUMULATOR 1
014242  012700  014736          MOV     #HA1W,R0
014246  172410                  LDD     (R0),AC0
014250  174001                  STD     AC0,AC1
                        ;LOAD ACCUMULATOR 2
014252  012700  014746          MOV     #HA2W,R0
014256  172410                  LDD     (R0),AC0
014260  174002                  STD     AC0,AC2
                        ;LOAD ACCUMULATOR 3
014262  012700  014756          MOV     #HA3W,R0
014266  172410                  LDD     (R0),AC0
014270  174003                  STD     AC0,AC3
                        ;LOAD ACCUMULATOR 4
014272  012700  014766          MOV     #HA4W,R0
014276  172410                  LDD     (R0),AC0
014300  174004                  STD     AC0,AC4
                        ;LOAD ACCUMULATOR 5
014302  012700  014776          MOV     #HA5W,R0
```

```
014306  172410                  LDD     (R0),AC0
014310  174005                  STD     AC0,AC5

014312  004737  014546          H4:     JSR     PC,@#HSTD               ;GO READ ALL ACCUMULATORS BACK.

014316  004737  014624                  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.
014322  012700  014736                  MOV     #HA1W,R0
014326  012702  000004                  MOV     #4,R2
014332  010001                  H5:     MOV     R0,R1
014334  005121                          COM     (R1)+
014336  172410                          LDD     (R0),AC0
014340  174001                          STD     AC0,AC1
014342  004737  014546                  JSR     PC,@#HSTD               ;READ ALL THE ACCUMULATORS BACK.
014346  004737  014624                  JSR     PC,@#HCMP               ;CHECK THE DATA.
014352  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.
014354  012700  014746                  MOV     #HA2W,R0
014360  012702  000004                  MOV     #4,R2
014364  010001                  H6:     MOV     R0,R1
014366  005121                          COM     (R1)+
014370  172410                          LDD     (R0),AC0
014372  174002                          STD     AC0,AC2
014374  004737  014546                  JSR     PC,@#HSTD               ;READ ALL THE ACCUMULATORS BACK.
014400  004737  014624                  JSR     PC,@#HCMP               ;CHECK THE DATA.
014404  077210                          SOB     R2,H6

                                ;COMPLIMENT EACH WORD OF THE DATA STORED IN ACCUMULATOR 3,
                                ;RELOAD THAT ACCUMULATOR, READ ALL THE ACCUMULATORS BACK AND CHECK
                                ;THE DATA.
014406  012700  014756                  MOV     #HA3W,R0
014412  012702  000004                  MOV     #4,R2
014416  010001                  H7:     MOV     R0,R1
014420  005121                          COM     (R1)+
014422  172410                          LDD     (R0),AC0
014424  174003                          STD     AC0,AC3
014426  004737  014546                  JSR     PC,@#HSTD               ;READ ALL THE ACCUMULATORS BACK.
014432  004737  014624                  JSR     PC,@#HCMP               ;CHECK THE DATA.
014436  077210                          SOB     R2,H7

                                ;COMPLIMENT EACH WORD OF THE DATA STORED IN ACCUMULATOR 4,
                                ;RELOAD THAT ACCUMULATOR, READ ALL THE ACCUMULATORS BACK AND CHECK
                                ;THE DATA.
014440  012700  014766                  MOV     #HA4W,R0
014444  012702  000004                  MOV     #4,R2
014450  010001                  H10:    MOV     R0,R1
014452  005121                          COM     (R1)+
014454  172410                          LDD     (R0),AC0
014456  174004                          STD     AC0,AC4
014460  004737  014546                  JSR     PC,@#HSTD               ;READ ALL THE ACCUMULATORS BACK.
```

```
                        712     FPP ACCUMULATORS DUAL ADDRESS TEST

[illegible]  014624  JSR     PC,@#HCMP               ;CHECK THE DATA.
[illegible]  014630  SOB     R2,H10

                        ;COMPLIMENT EACH WORD OF THE DATA STORED IN ACCUMULATOR 5,
                        ;RELOAD THAT ACCUMULATOR, READ ALL THE ACCUMULATORS BACK AND CHECK
                        ;THE DATA.
[illegible]  014472  MOV     #HA5W,R0
[illegible]  014476  MOV     #4,R2
[illegible]  014502  MOV     R0,R1
[illegible]  014504  H11:    COM     (R1)+
[illegible]  014506  LDD     (R0),AC0
[illegible]  014510  STD     AC0,AC5
[illegible]  014512  JSR     PC,@#HSTD               ;READ ALL THE ACCUMULATORS BACK.
[illegible]  014516  JSR     PC,@#HCMP               ;CHECK THE DATA.
[illegible]  014522  SOB     R2,H11

[illegible]  014524  005737  014734  TST     @#HFLAG
[illegible]  014530  001402          BEQ     H12
[illegible]  014532  000137  015126  JMP     @#HDONE

[illegible]  014536  005137  014734  H12:    COM     @#HFLAG
[illegible]  014542  000137  014240          JMP     @#H3

                        ;STORE ALL ACCUMULATORS IN THE OUTPUT BUFFERS.
[illegible]  014546  304737  014662  HSTD:   JSR     PC,@#HCLR               ;CLEAR ALL OUTPUT BUFFERS.
                        ;STORE ACCUMULATOR 1
[illegible]  014552  012704  015006          MOV     #HA1R,R4
[illegible]  014556  172401          LDD     AC1,AC0
[illegible]  014560  174014          STD     AC0,(R4)
                        ;STORE ACCUMULATOR 2
[illegible]  014562  012704  015016          MOV     #HA2R,R4
[illegible]  014566  172402          LDD     AC2,AC0
[illegible]  014570  174014          STD     AC0,(R4)
                        ;STORE ACCUMULATOR 3
[illegible]  014572  012704  015026          MOV     #HA3R,R4
[illegible]  014576  172403          LDD     AC3,AC0
[illegible]  014600  174014          STD     AC0,(R4)
                        ;STORE ACCUMULATOR 4
[illegible]  014602  012704  015036          MOV     #HA4R,R4
[illegible]  014606  172404          LDD     AC4,AC0
[illegible]  014610  174014          STD     AC0,(R4)
                        ;STORE ACCUMULATOR 5
[illegible]  014612  012704  015046          MOV     #HA5R,R4
[illegible]  014616  172405          LDD     AC5,AC0
[illegible]  014620  174014          STD     AC0,(R4)
[illegible]  014622  000207          RTS     PC

                        ;COMPARE DATA LOADED WITH DATA READ.
[illegible]  014624  012637  014732  HCMP:   MOV     (SP)+,@#HADR            ;SAVE RETURN ADDRESS.
[illegible]  014630  012703  014736          MOV     #HA1W,R3
[illegible]  014634  012704  015006          MOV     #HA1R,R4
[illegible]  014640  012705  000024          MOV     #24,R5
[illegible]  014644  022324          HCMP1:  CMP     (R3)+,(R4)+
[illegible]  014646  001402          BEQ     HCMP2
[illegible]  014650  000137  014700          JMP     @#HERROR
```

```
014654 077505                    HCMP2:  SOB     R5,HCMP1
014656 000177 000050                     JMP     @HADR

                                 ;CLEAR THE DATA OUTPUT BUFFER.
014662 012704 015006             HCLR:   MOV     #HA1R,R4
014666 012705 000024                     MOV     #24,R5
014672 005024                    HCLR1:  CLR     (R4)+
014674 077502                            SOB     R5,HCLR1
014676 000207                            RTS     PC

                                 ;REPORT ERROR.
014700                           HERROR:
014700 012703 014736                     MOV     #HA1W,R3
014704 012704 001236                     MOV     #STMP2,R4
014710 012705 000012                     MOV     #12,R5
014714 010324                    1$:     MOV     R3,(R4)+
014716 062703 000010                     ADD     #10,R3
014722 077504                            SOB     R5,1$
014724 104046                    2$:     ERROR   46
014726 000137 015126                     JMP     @#HDONE


014732 000000                    HADR:   0
014734 000000                    HFLAG:  0

014736 000000 000000 000000      HA1W:   .WORD   0,0,0,0
014744 000000
014746 000000 000000 000000      HA2W:   .WORD   0,0,0,0
014754 000000
014756 000000 000000 000000      HA3W:   .WORD   0,0,0,0
014764 000000
014766 000000 000000 000000      HA4W:   .WORD   0,0,0,0
014774 000000
014776 000000 000000 000000      HA5W:   .WORD   0,0,0,0
015004 000000

015006 000000 000000 000000      HA1R:   .WORD   0,0,0,0
015014 000000
015016 000000 000000 000000      HA2R:   .WORD   0,0,0,0
015024 000000
015026 000000 000000 000000      HA3R:   .WORD   0,0,0,0
015034 000000
015036 000000 000000 000000      HA4R:   .WORD   0,0,0,0
015044 000000
015046 000000 000000 000000      HA5R:   .WORD   0,0,0,0
015054 000000

015056 073567 073567 073567      HDAT1:  .WORD   73567,73567,73567,73567
015064 073567
015066 063146 063146 063146      HDAT2:  .WORD   63146,63146,63146,63146
015074 063146
015076 010421 010421 010421      HDAT3:  .WORD   10421,10421,10421,10421
015104 010421
015106 031463 031463 031463      HDAT4:  .WORD   31463,31463,31463,31463
015114 031463
015116 042104 042104 042104      HDAT5:  .WORD   42104,42104,42104,42104
```

```
015124  042104
        015126                          HDONE:
015126  104412                                  RSETUP                          ;GO INITIALIZE THE FPS AND STACK; AND
                                                                                ;SEE IF THE USER HAS EXPRESSED
                                                                                ;THE DESIRE TO CHANGE THE SOFTWARE
                                                                                ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
                                                                                ;THE USER TYPED CONTROL G?).


                                        ;;*************************************************************
                                        ;;*TEST 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).
                                        ;;*
                                        ;;*************************************************************
015130  000004                          TST13:  SCOPE
015132                                  S1:
015132  104413                                  LPERR                           ;SET UP THE LOOP ON ERROR ADDRESS.
015134  170011                                  SETD                                    ;SET FD
015136  012700  015642                          MOV     #SPAT10,R0                      ;LOAD AC0
015142  172410                                  LDD     (R0),AC0

015144  012737  015342  000244                  MOV     #SERR0,@#FPVECT                 ;USE OF THE NON-EXISTENT AC-
                                                                                        ;CUMULATOR SHOULD RESULT IN
                                                                                        ;A TRAP TO 244.
015152  012700  000001                          MOV     #1,R0                           ;A FAILURE IN THE FSRC FLOWS
                                                                                        ;WILL RESULT IN AN ODD ADDRESS
015156  012737  015552  000004                  MOV     #SERR1,@#ERRVECT                ;TRAP TO 4.
015164  005003                                  CLR     R3

015166  172407                          S2:     LDD     AC7,AC0
015170  170000                          S3:     CFCC
015172  005203                                  INC     R3
015174  005203                          S4:     INC     R3

015176  012701  015652                          MOV     #SDAT00,R1                      ;NO TRAP OCCURRED!!
015202  174011                                  STD     AC0,(R1)                        ;SEE IF AC0 WAS MODIFIED.

015204  012701  015652                          MOV     #SDAT00,R1
015210  012702  015642                          MOV     #SPAT10,R2
015214  012703  000004                          MOV     #4,R3
015220  022122                          S5:     CMP     (R1)+,(R2)+
015222  001402                                  BEQ     S6
015224  000137  015502                          JMP     @#SERR2
015230  077305                          S6:     SOB     R3,S5

015232  000137  015526                          JMP     @#SERR3

                                        ;NOW TEST AC6.
        015236                          S7:
015236  104413                                  LPERR                           ;SET UP THE LOOP ON ERROR ADDRESS.
015240  170011                                  SETD
```

H06


```
015242  012700  015642                  MOV     #SPAT10,R0              ;LOAD AC0
015246  172410                          LDD     (R0),AC0

015250  012737  015420  000244          MOV     #SERR4,@#FPVECT
015256  012700  000001                  MOV     #1,R0
015262  012737  015634  000004          MOV     #SERR5,@#ERRVECT
015270  005003                          CLR     R3

015272  172406                  S8:     LDD     AC6,AC0
015274  170000                  S9:     CFCC
015276  005203                          INC     R3
015300  005203                  S10:    INC     R3

015302  012701  015652                  MOV     #SDATO0,R1
015306  174011                          STD     AC0,(R1)                ;NO TRAP! GET AC0.

015310  012701  015652                  MOV     #SDATO0,R1              ;WAS AC0 MODIFIED.
015314  012702  015642                  MOV     #SPAT10,R2
015320  012703  000004                  MOV     #4,R3
015324  022122                  S11:    CMP     (R1)+,(R2)+
015326  001402                          BEQ     S12
015330  000137  015514                  JMP     @#SERR6
015334  077305                  S12:    SOB     R3,S11
015336  000137  015540                  JMP     @#SERR7

                                ;TRAPPED TO 244.
015342  021627  015170          SERR0:  CMP     (SP),#S3                ;PC OF TRAP CORRECT?
015346  001402                          BEQ     1$
015350  000137  040200                  JMP     @#FPSPUR

015354  012737  015236  015636  1$:     MOV     #S7,@#SADR

015362  011637  001236          SERR10: MOV     (SP),@#$TMP2
015366  022626                          CMP     (SP)+,(SP)+
015370  005004                          CLR     R4
015372  170204                          STFPS   R4                      ;IS FPS CORRECT?
015374  022704  100200                  CMP     #100200,R4
015400  001020                          BNE     SERR15

015402  005004                          CLR     R4
015404  170304                          STST    R4                      ;IS FEC CORRECT?
015406  022704  000002                  CMP     #2,R4
015412  001023                          BNE     SERR20
015414  000177  000216                  JMP     @SADR

015420  021627  015274          SERR4:  CMP     (SP),#S9
015424  001402                          BEQ     1$
015426  000137  040200                  JMP     @#FPSPUR
015432  012737  015662  015636  1$:     MOV     #SDONE,@#SADR
015440  000750                          BR      SERR10

                                ;REPORT FPS FAILURE:
015442  012737  100200  001242  SERR15: MOV     #100200,@#$TMP4
015450  010437  001240                  MOV     R4,@#$TMP3
015454  104117                  1$:     ERROR   117
015456  000177  000154                  JMP     @SADR
```

```
                                        ;REPORT FEC BAD:
015462  012737  000002  001242  SERR2C: MOV     #2,@#$TMP4
015470  010437  001240                  MOV     R4,@#$TMP3
015474  104120                  1$:     ERROR   120
015476  000177  000134                  JMP     @SADR

                                        ;ACC WAS MODIFIED. (BUT FSRC) FORK FAILED.
015502  012737  015166  001236  SERR2:  MOV     #S2,@#$TMP2
015510  104112                  1$:     ERROR   112
015512  000463                          BR      SDONE
015514  012737  015272  001236  SERR6:  MOV     #S8,@#$TMP2
015522  104114                  1$:     ERROR   114
015524  000456                          BR      SDONE

015526  012737  015166  001236  SERR3:  MOV     #S2,@#$TMP2
015534  104111                  1$:     ERROR   111
015536  000451                          BR      SDONE
015540  012737  015272  001236  SERR7:  MOV     #S8,@#$TMP2
015546  104113                  1$:     ERROR   113
015550  000444                          BR      SDONE

                                        ;FAILURE OF (BUT FSRC) CAUSED AN ODD ADDRESS TRAP TO 4.
015552  021627  015170          SERR1:  CMP     (SP),#S3                ;DID TRAP OCCUR ON TESTED INSTRUCTION?
015556  001405                          BEQ     1$
015560  021627  015174                  CMP     (SP),#S4
015564  001402                          BEQ     1$
015566  000137  040232                  JMP     @#CPSPUR

015572  011637  001236          1$:     MOV     (SP),@#$TMP2
015576  022626                          CMP     (SP)+,(SP)+
015600  104115                  2$:     ERROR   115
015602  000427                          BR      SDONE

015604  021627  015272          SERR5:  CMP     (SP),#S8                ;DID TRAP OCCUR ON TEST INSTRUCTION?
015610  001405                          BEQ     1$
015612  021627  015274                  CMP     (SP),#S9
015616  001402                          BEQ     1$
015620  000137  040232                  JMP     @#CPSPUR

015624  011637  001236          1$:     MOV     (SP),@#$TMP2
015630  022626                          CMP     (SP)+,(SP)+
015632  104116                  2$:     ERROR   116
015634  000412                          BR      SDONE

015636  000000                  SADR:   0
015640  177777                          -1
015642  010421                  SPAT10: 10421
015644  021042                  SPAT11: 21042
015646  031463                  SPAT12: 31463
015650  042104                  SPAT13: 42104

015652  000000                  SDAT00: 0
015654  000000                  SDAT01: 0
015656  000000                  SDAT02: 0
```

```
                SDAT03: 0
                SDONE:
                        RSETJP                  ;GO INITIALIZE THE FPS AND STACK: AND
                                                ;SEE IF THE USER HAS EXPRESSED
                                                ;THE DESIRE TO CHANGE THE SOFTWARE
                                                ;VIRTUAL CONSOLE SWITCH REGISTER  HAS
                                                ;THE USER TYPED CONTROL G?).

                ;;*****************************************************************
                ;*TEST 14       FSRC MODE 2 TEST
                ;*
                ;* THIS IS A TEST OF FSRC MODE 2, AUTO
                ;* INCREMENT MODE.
                ;*
                ;;*****************************************************************
015664  000004  TST14:  SCOPE
015666  104413          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.

015670          J1:
015670  170011          SETD                    ;SET DOUBLE MODE

015672  012700  016146          MOV     #JDAT0,R0
015676  172410          LDD     (R0),AC0        ;LOAD AC0

015700  012700  016126          MOV     #JDAT1,R0
015704  005003          CLR     R3
015706  012737  015776  000004  MOV     #JERR0,@#ERRVECT

015714  172420          J2:     LDD     (R0)+,AC0       ;TEST INSTRUCTION
015716  005203          J3:     INC     R3
015720  005203          J4:     INC     R3

015722  012701  016136          MOV     #JDAT00,R1
015726  174011          STD     AC0,(R1)        ;PICK UP RESULTS

015730  020027  016116          CMP     R0,#JBUF0       ;WAS AN AUTO
015734  001001          BNE     1$              ;DECREMENT EXECUTED?
015736  000442          BR      JERR1

015740  012702  016126  1$:     MOV     #JDAT1,R2       ;IS DATA CORRECT?
015744  012703  016136          MOV     #JDAT00,R3
015750  012704  000004          MOV     #4,R4
015754  022223          J5:     CMP     (R2)+,(R3)+
015756  001401          BEQ     J6
015760  000443          BR      JERR2
015762  077404          J6:     SOB     R4,J5

015764  022700  016136          CMP     #JDAT1+10,R0    ;WAS R0 INCREM.
015770  001401          BEQ     J7              ;BY 10 (OCTAL)
015772  000424          BR      JERR1

015774  000470          J7:     BR      JDONE

                ;IF A TRAP THROUGH 4 OCCURS COME HERE
```

```
4098  016776  021627  015716          JERR0:  CMP     (SP),#J3        ;SEE IF THE TRAP
4099  016002  001405                  BEQ     J10             ;OCCURRED ON THE
4100  016004  021627  015720          CMP     (SP),#J4        ;TESTED INSTRUCTION
4101  016010  001402                  BEQ     J10
4102  016012  000137  040232          JMP     @#CPSPUR
4103
4104  016016  012737  000762  001240  J10:    MOV     #762,@#$TMP3    ;REPORT FSRC FLOW
4105  016024  012737  000322  001242          MOV     #322,@#$TMP4    ;FAILURE
4106  016032  011637  001236          MOV     (SP),@#$TMP2
4107  016036  022626                  CMP     (SP)+,(SP)+
4108  016040  104052                  1$:     ERROR   52
4109  016042  000445                  BR      JDONE
4110
4111  016044                          JERR1:                          ;REPORT, R0 NOT
4112  016044  012737  015714  001236          MOV     #J2,@#$TMP2     ;CORRECTLY AFFECTED
4113  016052  010037  001240          MOV     R0,@#$TMP3
4114  016056  012737  016136  001242          MOV     #JDATI0+10,@#$TMP4
4115  016064  104053                  1$:     ERROR   53
4116  016066  000433                  BR      JDONE
4117
4118                                  ;REPORT DATA FAILURE
4119
4120  016070                          JERR2:
4121  016070  012737  015714  001236          MOV     #J2,@#$TMP2
4122  016076  012737  016126  001240          MOV     #JDATI0,@#$TMP3
4123  016104  012737  016136  001242          MOV     #JDAT00,@#$TMP4
4124  016112  104054                  1$:     ERROR   54
4125  016114  000420                  BR      JDONE
4126
4127  016116  010421                  JBUF0:  .WORD   010421
4128  016120  021042                  JBUF1:          021042
4129  016122  042104                  JBUF2:          042104
4130  016124  031463                  JBUF3:          031463
4131
4132  016126  052525                  JDATI0:         052525
4133  016130  114631                  JDATI1:         114631
4134  016132  063146                  JDATI2:         063146
4135  016134  073567                  JDATI3:         073567
4136
4137  016136  000000                  JDAT00:         0
4138  016140  000000                  JDAT01:         0
4139  016142  000000                  JDAT02:         0
4140  016144  000000                  JDAT03:         0
4141
4142  016146  177777                  JDAT0:          -1
4143  016150  177777                  JDAT1:          -1
4144  016152  177777                  JDAT2:          -1
4145  016154  177777                  JDAT3:          -1
4146
4147
4148  016156                          JDONE:
4149  016156  104412                          RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
4150                                                                  ;SEE IF THE USER HAS EXPRESSED
4151                                                                  ;THE DESIRE TO CHANGE THE SOFTWARE
4152                                                                  ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
4153                                                                  ;THE USER TYPED CONTROL G?).
```

```
                                        ;;*****************************************************************
                                        ;*TEST 15       FSRC MODE 4 TEST
                                        ;*
                                        ;* THIS IS A TEST OF FSRC MODE 4, AUTO
                                        ;* DECREMENT MODE.
                                        ;*
                                        ;;*****************************************************************
016160  000004                  TST15:  SCOPE
016162  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.

016164                          K1:
016164  170011                          SETD                    ;SET DOUBLE MODE

016166  012700  016440                  MOV     #KPAT0,R0
016172  172410                          LDD     (R0),AC0        ;LOAD A DEFAULT
                                                                ;PATTERN INTO.AC0
016174  012700  016420                  MOV     #KBUF0,R0
016200  005003                          CLR     R3
016202  012737  016272  000004          MOV     #KERR0,@#ERRVECT

016210  172440                  K2:     LDD     -(R0),AC0       ;TEST INSTRUCTION
016212  005203                  K3:     INC     R3
016214  005203                  K4:     INC     R3

016216  012701  016430                  MOV     #KDAT00,R1
016222  174011                          STD     AC0,(R1)        ;PICK UP THE RESULT

016224  020027  016430                  CMP     R0,#KBUF0+10    ;WAS AN AUTO
016230  001001                          BNE     1$              ;INCREMENT EXECUTED
016232  000441                          BR      KERR1

016234  012702  016410          1$:     MOV     #KDATI0,R2      ;IS DATA CORRECT?
016240  012703  016430                  MOV     #KDAT00,R3
016244  012704  000004                  MOV     #4,R4
016250  022223                  K5:     CMP     (R2)+,(R3)+
016252  001401                          BEQ     K6
016254  000442                          BR      KERR2
016256  077404                  K6:     SOB     R4,K5

016260  022700  016410                  CMP     #KBUF0-10,R0    ;WAS R0 DECREMENTED
016264  001401                          BEQ     K7              ;PROPERLY?
016266  000423                          BR      KERR1

016270  000457                  K7:     BR      KDONE

                                        ;TRAP TO HERE ON AN ODD ADDRESS ERROR

016272  021627  016212          KERR0:  CMP     (SP),#K3        ;SEE IF THE ERROR
016276  001405                          BEQ     K10             ;OCCURRED AT THE
016300  021627  016214                  CMP     (SP),#K4        ;INSTRUCTION TESTED.
016304  001402                          BEQ     K10
016306  000137  040232                  JMP     @#CPSPUR

016312  012737  000762  001240  K10:    MOV     #762,@#$TMP3    ;REPORT FAILURE IN
```

```
4210  016320  012737  000324  001242          MOV     #324,@#$TMP4    ;FSRC FLOWS
4211  016326  011637  001236                  MOV     (SP),@#$TMP2
4212  016332  104055                  1$:     ERROR   55
4213  016334  000445                          BR      KDONE
4214
4215  016336                          KERR1:                          ;REPORT, R0
4216  016336  012737  016210  001236          MOV     #K2,@#$TMP2     ;INCORRECTLY AFFECTED.
4217  016344  010037  001240                  MOV     R0,@#$TMP3
4218  016350  012737  016410  001242          MOV     #KDATI0,@#$TMP4
4219  016356  104056                  1$:     ERROR   56
4220  016360  000433                          BR      KDONE
4221
4222                                  ;REPORT DATA FAILURE
4223
4224  016362                          KERR2:
4225  016362  012737  016210  001236          MOV     #K2,@#$TMP2
4226  016370  012737  016410  001240          MOV     #KDATI0,@#$TMP3
4227  016376  012737  016430  001242          MOV     #KDAT00,@#$TMP4
4228  016404  104057                  1$:     ERROR   57
4229  016406  000420                          BR      KDONE
4230
4231  016410  052525                  KDATI0: .WORD   052525
4232  016412  114631                  KDATI1:         114631
4233  016414  063140                  KDATI2:         063140
4234  016416  073567                  KDATI3:         073567
4235
4236  016420  010421                  KBUF0:          010421
4237  016422  031463                  KBUF1:          031463
4238  016424  042104                  KBUF2:          042104
4239  016426  021042                  KBUF3:          021042
4240
4241  016430  000000                  KDAT00:         0
4242  016432  000000                  KDAT01:         0
4243  016434  000000                  KDAT02:         0
4244  016436  000000                  KDAT03:         0
4245
4246  016440  177777                  KPAT0:          -1
4247  016442  177777                  KPAT1:          -1
4248  016444  177777                  KPAT2:          -1
4249  016446  177777                  DPAT3:          -1
4250
4251  016450                          KDONE:
4252  016450  104412                          RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
4253                                                                  ;SEE IF THE USER HAS EXPRESSED
4254                                                                  ;THE DESIRE TO CHANGE THE SOFTWARE
4255                                                                  ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
4256                                                                  ;THE USER TYPED CONTROL G?).
4257
4258
4259                                  ;;*****************************************************************
4260                                  ;*TEST 16       FSRC MODE 2, WITH FD=0, TEST
4261                                  ;*
4262                                  ;* THIS IS A TEST OF FSRC MODE 2 WITH
4263                                  ;* FD=0.  (AUTO INCREMENT)
4264                                  ;*
4265                                  ;;*****************************************************************
```

```
4266  016452  000004                  TST16:  SCOPE
4267  016454  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
4268
4269  016456                          L1:
4270  016456  170011                          SETD                    ;SET DOUBLE MODE
4271
4272  016460  012700  016726                  MOV     #LPAT10,R0
4273  016464  172410                          LDD     (R0),AC0        ;LOAD AC0
4274
4275  016466  012700  016750                  MOV     #LDATI0,R0      ;SET UP THE INPUT
4276  016472  012701  016736                  MOV     #LPAT20,R1      ;DATA
4277  016476  012702  000004                  MOV     #4,R2
4278
4279  016502  012120                  1$:     MOV     (R1)+,(R0)+
4280  016504  077202                          SOB     R2,1$
4281
4282  016506  012700  016750                  MOV     #LDATI0,R0
4283  016512  005003                          CLR     R3
4284  016514  170001                          SETF                    ;CLEAR FD.
4285
4286  016516  172420                  L2:     LDF     (R0)+,AC0
4287  016520  005203                  L3:     INC     R3
4288
4289  016522                          L4:
4290  016522  170011                          SETD                    ;SET FD
4291
4292  016524  012701  016762                  MOV     #LDAT00,R1
4293  016530  174011                          STD     AC0,(R1)        ;PICK UP RESULTS
4294
4295  016532  020027  016754                  CMP     R0,#LDATI2      ;WAS R0 INCREMENTED
4296  016536  001401                          BEQ     1$              ;CORRECTLY BY 4
4297  016540  000421                          BR      LERR1
4298
4299  016542  012737  177777  016754  1$:     MOV     #-1,@#LDATI2
4300  016550  012737  177777  016756          MOV     #-1,@#LDATI3
4301  016556  012702  016750                  MOV     #LDATI0,R2      ;IS DATA CORRECT
4302  016562  012703  016762                  MOV     #LDAT00,R3
4303  016566  012704  000004                  MOV     #4,R4
   74
-  5  016572  022223                  L5:     CMP     (R2)+,(R3)+
4306  016574  001401                          BEQ     L6
4307  016576  000427                          BR      LERR2
4308  016600  077404                  L6:     SOB     R4,L5
4309
4310  016602  000473                          BR      LDONE
4311
4312  016604                          LERR1:                          ;REPORT FAILURE
4313  016604  012737  016516  001236          MOV     #L2,@#$TMP2     ;R0 NOT INCREMENTED
4314  016612  010037  001240                  MOV     R0,@#$TMP3      ;BY 4
4315  016616  012737  016754  001242          MOV     #LDATI2,@#$TMP4
4316  016624  104060                  1$:     ERROR   60
4317  016626  000461                          BR      LDONE
4318
4319  016630                          LERR3:                          ;REPORT DATA FAILURE.
4320  016630  012737  016516  001236          MOV     #L2,@#$TMP2
4321  016636  012737  016750  001240          MOV     #LDATI0,@#$TMP3
```

```
016644  012737  016762  001242          MOV     #LDAT00,@#$TMP4
016652  104061                  1$:     ERROR   61
016654  000446                          BR      LDONE

016656  012702  016736          LERR2:  MOV     #LPAT20,R2              ;DID (BUT FD)
016662  012703  016762                  MOV     #LDAT00,R3              ;FAIL.
016666  012704  000004                  MOV     #4,R4
016672  022223                  1$:     CMP     (R2)+,(R3)+
016674  001355                          BNE     LERR3
016676  077403                          SOB     R4,1$
016700  012737  016516  001236          MOV     #L2,@#$TMP2
016706  012737  016750  001240          MOV     #LDAT10,@#$TMP3
016714  012737  016764  001242          MOV     #LDAT01,@#$TMP4
016722  104062                  2$:     ERROR   62
016724  000422                          BR      LDONE

016726  177777                  LPAT10: .WORD   -1
016730  177777                  LPAT11:         -1
016732  177777                  LPAT12:         -1
016734  177777                  LPAT13:         -1

016736  052525                  LPAT20:         052525
016740  114631                  LPAT21:         114631
016742  063142                  LPAT22:         063142
016744  073567                  LPAT23:         073567
016746  000001                          .WORD   000001
016750  000000                  LDAT10:         0
016752  000000                  LDAT11:         0
016754  000000                  LDAT12:         0
016756  000000                  LDAT13:         0
016760  000001                          .WORD   00001
016762  000000                  LDAT00:         0
016764  000000                  LDAT01:         0
016766  000000                  LDAT02:         0
016770  000000                  LDAT03:         0

016772                          LDONE:
016772  104412                          RSETUP                          ;GO INITIALIZE THE FPS AND STACK; AND
                                                                        ;SEE IF THE USER HAS EXPRESSED
                                                                        ;THE DESIRE TO CHANGE THE SOFTWARE
                                                                        ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
                                                                        ;THE USER TYPED CONTROL G?).



                                ;;*************************************************************
                                ;*TEST 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.
                                ;*
                                ;;*************************************************************
016774  000004                  TST17:  SCOPE

016776                          M1:
016776  170011                          SETD
```

```
        017000  012700  017272                  MOV     #MPAT10,R0
        017004  172410                          LDD     (R0),AC0        ;LOAD BACKROUND
                                                                        ;PATTERN INTO ACC.
        017006  005004                          CLR     R4
        017010  012737  017232  000004          MOV     #MERR3,@#ERRVECT

        017016  172427  000000          M15:    LDD     #0,AC0          ;TEST INSTRUCTION
                017020                          .=.-2
        017020  005204                          .WORD   5204
        017022  005204                  M2:     INC     R4              ;NOTE THAT
        017024  005204                  M3:     INC     R4              ;005204=INC R4
        017026  005204                  M4:     INC     R4

        017030  020427  000003                  CMP     R4,#3           ;SEE IF THE PC
        017034  001401                          BEQ     1$              ;WAS INCREMENTED
        017036  000443                          BR      MERR0           ;BY 2 DURING THE
                                                                        ;INSTRUCTION.  IF
                                                                        ;NOT THEN A BAD
                                                                        ;CONSTANT WAS GENERATED
        017040  012700  017312          1$:     MOV     #MDAT00,R0
        017044  174010                          STD     AC0,(R0)        ;GET THE DATA

        017046  012700  017312                  MOV     #MDAT00,R0
        017052  022720  005204                  CMP     #5204,(R0)+     ;IS THE DATA CORRECT?
        017056  001401                          BEQ     M5
        017060  000451                          BR      MERR1
        017062  012701  000003          M5:     MOV     #3,R1
        017066  005720                  M6:     TST     (R0)+
        017070  001002                          BNE     M7
        017072  077103                          SOB     R1,M6
        017074  000512                          BR      MDONE

        017076  012700  017312          M7:     MOV     #MDAT00,R0      ;DID (BUT GRM) FAIL?
        017102  012701  000004                  MOV     #4,R1
        017106  022720  005204          M8:     CMP     #5204,(R0)+
        017112  001401                          BEQ     M9
        017114  000433                          BR      MERR1
        017116  077105                  M9:     SOB     R1,M8

        017120                          MERR2:                          ;REPORT FAILURE
        017120  012737  017016  001236          MOV     #M15,@#$TMP2    ;OF (BUT GR7)
        017126  012737  017302  001240          MOV     #MPAT20,@#$TMP3
        017134  012737  017312  001242          MOV     #MDAT00,@#$TMP4
        017142  104063                  1$:     ERROR   63
        017144  000466                          BR      MDONE

        017146  012705  017022          MERR0:  MOV     #M2,R5          ;REPORT FAILURE
        017152  010537  001242                  MOV     R5,@#$TMP4      ;PC INCREMENTED
        017156  162704  000003                  SUB     #3,R4
        017162  006304                          ASL     R4
        017164  160405                          SUB     R4,R5
        017166  010537  001240                  MOV     R5,@#$TMP3
        017172  012737  017016  001236          MOV     #M15,@#$TMP2
        017200  104064                  1$:     ERROR   64
        017202  000447                          BR      MDONE
```

```
017204                          MERR1:                                  ;REPORT DATA
017204  012737  017016  001236          MOV     #M15,@#STMP2            ;FAILURE
017212  012737  017312  001240          MOV     #MDAT00,@#STMP3
017220  012737  017302  001242          MOV     #MPAT20,@#STMP4
017226  104066                  1$:     ERROR   66
017230  000434                          BR      MDONE
                                ;TRAP TO HERE THROUGH 4.
017232  032716  000001          MERR3:  BIT     #1,(SP)                 ;SEE IF THE
017236  001002                          BNE     1$                      ;TRAP TO 4 OCCURRED
017240  000137  040232                  JMP     @#CPSPUR                ;BECAUSE OF AN
                                                                        ;ODD ADDRESS
017244  011637  001240          1$:     MOV     (SP),@#STMP3            ;IF YES REPORT
017250  012737  017022  001242          MOV     #M2,@#STMP4             ;BAD CONSTANT
017256  012737  017016  001236          MOV     #M15,@#STMP2            ;GENERATED
017264  022626                          CMP     (SP)+,(SP)+
017266  104065                  2$:     ERROR   65
017270  000414                          BR      MDONE

017272  177777                  MPAT10:         -1
017274  177777                  MPAT11:         -1
017276  177777                  MPAT12:         -1
017300  177777                  MPAT13:         -1

017302  005204                  MPAT20:         5204
017304  005204                  MPAT21:         5204
017306  005204                  MPAT22:         5204
017310  005204                  MPAT23:         5204

017312  000000                  MDAT00:         0
017314  000000                  MDAT01:         0
017316  000000                  MDAT02:         0
017320  000000                  MDAT03:         0

017322                          MDONE:
017322  104412                          RSETUP                          ;GO INITIALIZE THE FPS AND STACK; AND
                                                                        ;SEE IF THE USER HAS EXPRESSED
                                                                        ;THE DESIRE TO CHANGE THE SOFTWARE
                                                                        ;VIRTUAL CONSOLE SWITCH REGISTER.(HAS
                                                                        ;THE USER TYPED CONTROL G?).





                                ;;*****************************************************************
                                ;*TEST 20       FSRC MODE 3 TEST
                                ;*
                                ;* THIS IS A TEST OF FSRC MODE 3, AUTO INCREMENT
                                ;* DEFERRED
                                ;*
                                ;;*****************************************************************
017324  000004                  TST20:  SCOPE

017326                          N1:
017326  170011                          SETD                            ;SET FD MODE

017330  012700  020010                  MOV     #NPAT10,R0
017334  172410                          LDD     (R0),AC0                ;LOAD AC0 WITH A DEFAULT
```

```
                                                        ;PATTERN
017336  012700  017776          MOV     #NPAT20,R0
017342  005003                  CLR     R3
017344  012737  017620  000004  MOV     #NERR0,2#EPRVECT        ;IF A FAILURE OCCURS
                                                        ;IN THE FSRC FLOWS AN
                                                        ;ODD TRAP TO 4 COULD OCCUR
017352  172430          N2:     LDD     @(R0)+,AC0      ;TEST INSTRUCTION.
017354  005203          N3:     INC     R3
017356  005203          N4:     INC     R3

017360  012701  017756          MOV     #NDAT00,R1
017364  174011                  STD     AC0,(R1)        ;GET THE DATA

017366  020027  020000          CMP     R0,#NPAT20+2    ;WAS R0 INCREMENTED
017372  001437                  BEQ     N12             ;BY 2?

017374  020027  020006  N5:     CMP     R0,#NPAT20+10   ;FSRC MODE 2?
017400  001001                  BNE     N6
017402  000506                  BR      NERR1

017404  020027  017756  N6:     CMP     R0,#NPAT20-10   ;FSRC MODE 4?
017410  001001                  BNE     N7
017412  000520                  BR      NERR2

017414  020027  017776  N7:     CMP     R0,#NPAT20
017420  001023                  BNE     N11

017422  012702  017756          MOV     #NDAT00,R2      ;FSRC MODE 3?
017426  012703  000004          MOV     #4,R3
017432  022227  177777  N8:     CMP     (R2)+,#-1
017436  001002                  BNE     N9
017440  077304                  SOB     R3,N8
017442  000510                  BR      NERR3

017444  012702  017756  N9:     MOV     #NDAT00,R2      ;FSRC MODE 1
017450  012703  017776          MOV     #NPAT20,R3
017454  012704  000004          MOV     #4,R4
017460  022223          N10:    CMP     (R2)+,(R3)+
017462  001002                  BNE     N11
017464  077403                  SOB     R4,N10
017466  000502                  BR      NERR4

017470  000505          N11:    BR      NERR5

017472  012702  017756  N12:    MOV     #NDAT00,R2      ;DATA CORRECT?
017476  012703  020020          MOV     #NDAT10,R3
017502  012704  000004          MOV     #4,R4
017506  022223          N13:    CMP     (R2)+,(R3)+
017510  001002                  BNE     N14
017512  077403                  SOB     R4,N13
017514  000545                  BR      NDONE

017516  000504          N14:    BR      NERR6

                        ;IF AN ODD ADDRESS TRAP OCCURS COME HERE
                        ;TO SEE IF THE FAILURE WAS IN THE FSRC
```

```
                                        ;FLOWS
017520  022716  017356          NERR0:  CMP     #N4,(SP)        ;FSRC MODE 6 OR 7?
017524  001412                  BEQ     NERR10
017526  022716  017354          CMP     #N3,(SP)
017532  001403                  BEQ     1$
017534  000137  040232          JMP     @#CPSPUR
017540  020027  017774  1$:     CMP     R0,#NPAT20-2    ;FSRC MODE 5?
017544  001407                  BEQ     NERR11
017546  000137  040232          JMP     @#CPSPUR

017552                          NERR10:                 ;WENT TO FSRC
017552  011637  001236          MOV     (SP),@#$TMP2    ;MODE 6 OR 7.
017556  022626                  CMP     (SP)+,(SP)+
017560  104067          1$:     ERROR   67
017562  000522                  BR      NDONE

017564  011637  001236  NERR11: MOV     (SP),@#$TMP2    ;WENT TO FSRC
017570  022626                  CMP     (SP)+,(SP)+     ;MODE 5.
017572  012737  000627  001244  NERR1:  MOV     #627,@#$TMP5
017600  012737  000323  001250  NERR2C: MOV     #323,@#$TMP7
017606  012737  000325  001246          MOV     #325,@#$TMP6
017614  104070          1$:     ERROR   70
017616  000504                  BR      NDONE
017620  012737  000322  001246  NERR1:  MOV     #322,@#$TMP6    ;FSRC MODE 3.
017626  012737  000627  001244  NERR2C: MOV     #627,@#$TMP5
017634  012737  000323  001250          MOV     #323,@#$TMP7
017642  012737  017352  001236          MOV     #N2,@#$TMP2
017650  104071          1$:     ERROR   71
017652  000466                  BR      NDONE
017654  012737  000324  001246  NERR2:  MOV     #324,@#$TMP6    ;FSRC MODE 4
017662  000761                  BR      NERR20
017664  012737  000320  001246  NERR3:  MOV     #320,@#$TMP6    ;FSRC MODE 0
017672  000755                  BR      NERR20
017674  012737  000321  001246  NERR4:  MOV     #321,@#$TMP6    ;FSRC MODE 1
017702  000751                  BR      NERR20

017704  010037  001240  NERR5:  MOV     R0,@#$TMP3      ;R0 NOT
017710  012737  020000  001242          MOV     #NPAT20+2,@#$TMP4       ;INCREMENTED
017716  012737  017352  001236          MOV     #N2,@#$TMP2     ;PROPERLY.
017724  104072          1$:     ERROR   72
017726  000440                  BR      NDONE

017730                          NERR6:                  ;DATA FAILURE.
017730  012737  017352  001236          MOV     #N2,@#$TMP2
017736  012737  017756  001240          MOV     #NDAT00,@#$TMP3
017744  012737  020020  001242          MOV     #NDAT10,@#$TMP4
017752  104073          1$:     ERROR   73
017754  000425                  BR      NDONE

017756  000000          NDAT00: .WORD   0
017760  000000          NDAT01:         0
017762  000000          NDAT02:         0
017764  000000          NDAT03:         0

017766  052525  052525  052525          .WORD   52525,52525,52525,52525
```

```
020020  010421          NDAT10: .WORD   010421
020022  021042          NDAT11:         021042
020024  031463          NDAT12:         031463
020026  042104          NDAT13:         042104

020030                  NDONE:
020030  104412                  RESETUP                 ;GO INITIALIZE THE FPS AND STACK; AND
                                                        ;SEE IF THE USER HAS EXPRESSED
                                                        ;THE DESIRE TO CHANGE THE SOFTWARE
                                                        ;VIRTUAL CONSOLE SWITCH REGISTER  WAS
                                                        ;THE USER TYPED CONTROL G? .


                                ;;*******************************************************************
                                ;*TEST 21       FSRC MODE 5 TEST
                                ;*
                                ;* THIS IS A TEST OF FSRC MODE 5, AUTO DECREMENT
                                ;* DEFERRED.
                                ;*
                                ;;*******************************************************************
020032  000004          TST21:  SCOPE

020034                  01:
020034  170011                  SETD                    ;SET FD MODE

020036  012700  020514          MOV     #OPAT10,R0
020042  172410                  LDD     (R0),AC0        ;LOAD AC0 WITH A
                                                        ;DEFAULT PATTERN.
020044  012700  020502          MOV     #OPAT21,R0
020050  005003                  CLR     R3
020052  012737  020224  000004  MOV     #OERR0,@#ERRVEC ;IF A FAILURE
                                                        ;OCCURS IN THE FSRC
                                                        ;FLOWS AN ODD ADDR.
                                                        ;TRAP TO 4 MAY OCCUR.
020060  172450          02:     LDD     @-(R0),AC0      ;TEST INSTRUCTION
020062  005203          03:     INC     R3
020064  005203          04:     INC     R3

020066  012701  020462          MOV     #ODAT00,R1
020072  174011                  STD     AC0,(R1)        ;GET THE DATA

020074  020027  020500          CMP     R0,#OPAT20      ;WAS R0 DECREMENTED
020100  001436                  BEQ     012             ;BY 2?

020102  020027  020512  05:     CMP     R0,#OPAT21+10   ;FSRC MODE 2
```

```
020106  001001                 BNE     06
020110  000505                 BR      OERR1

020112  020027  020472  06:    CMP     R0,#OPAT21-10   ;FSRC MODE 4?
020116  001001                 BNE     07
020120  000517                 BR      OERR2

020122  020027  020502  07:    CMP     R0,#OPAT21

020126  012702  020464         MOV     #ODAT01,R2      ;FSRC MODE 0?
020132  012703  000004         MOV     #4,R3
020136  022227  177777  08:    CMP     (R2)+,#-1
020142  001002                 BNE     09
020144  077304                 SOB     R3,08
020146  000510                 BR      OERR3

020150  012702  020462  09:    MOV     #ODAT00,R2      ;FSRC MODE 1?
020154  012703  020502         MOV     #OPAT21,R3
020160  012704  000004         MOV     #4,R4
020164  022223          010:   CMP     (R2)+,(R3)+
020166  001002                 BNE     011
020170  077403                 SOB     R4,010
020172  000502                 BR      OERR4

020174  000505          011:   BR      OERR5

020176  012702  020462  012:   MOV     #ODAT00,R2      ;DATA CORRECT?
020202  012703  020524         MOV     #ODAT10,R3
020206  012704  000004         MOV     #4,R4
020212  022223          013:   CMP     (R2)+,(R3)+
020214  001002                 BNE     014
020216  077403                 SOB     R4,013
020220  000545                 BR      ODONE

020222  000504          014:   BR      OERR6

                        ;IF AN ODD ADDRESS TRAP OCCURS COME
                        ;HERE TO SEE IF THE FAILURE WAS IN THE
                        ;FSRC FLOWS:

020224  022716  020064  OERR0: CMP     #04,(SP)        ;FSRC MODE 6 OR 7?
020230  001412                 BEQ     OERR10
020232  022716  020062         CMP     #03,(SP)
020236  001402                 BEQ     1$
020240  000137  040232         JMP     @#CPSPUR
020244  020027  020504  1$:    CMP     R0,#OPAT21+2    ;FSRC MODE 3?
020250  001425                 BEQ     OERR1
020252  000137  040232         JMP     @#CPSPUR

020256                  OERR10:                        ;WENT TO FSRC
020256  011637  001236         MOV     (SP),@#$TMP2    ;MODE 6 OR 7
020262  022626                 CMP     (SP)+,(SP)+
020264  104074          1$:    ERROR   74
020266  000522                 BR      ODONE

020270  011637  001240  OERR11: MOV    (SP),@#$TMP3    ;WENT TO FSRC MODE
```

```
                                                T21     FSRC MODE 5 TEST

[illegible]  [illegible]  [illegible]                           CMP     (SP)+,(SP)+     ;3
[illegible]  [illegible]  [illegible]  000627  001244           MOV     #627,@#$TMP5
[illegible]  [illegible]  [illegible]  000325  001250           MOV     #325,@#$TMP7
[illegible]  [illegible]  [illegible]  000323  001246           MOV     #323,@#$TMP6
[illegible]  [illegible]  [illegible]                   1$:     ERROR   75
[illegible]  [illegible]  [illegible]                           BR      ODONE

020324  012737  000322  001246  OERR1:  MOV     #322,@#$TMP6    ;FSRC MODE2
020332  012737  000627  001242  OERR20: MOV     #627,@#$TMP4
020340  012737  000325  001250          MOV     #325,@#$TMP7
020346  012737  020060  001236          MOV     #02,@#$TMP2
020354  104076                  1$:     ERROR   76
020356  000466                          BR      ODONE
020360  012737  000324  001246  OERR2:  MOV     #324,@#$TMP6    ;FSRC MODE 4
020366  000761                          BR      OERR20
020370  012737  000320  001246  OERR3:  MOV     #320,@#$TMP6    ;FSRC MODE 0
020376  000755                          BR      OERR20
020400  012737  000321  001246  OERR4:  MOV     #321,@#$TMP6    ;FSRC MODE 1
020406  000751                          BR      OERR20

020410  010037  001240          OERR5:  MOV     R0,@#$TMP3      ;R0 NOT DECREMENTED
020414  012737  020500  001242          MOV     #OPAT20,@#$TMP4 ;PROPERLY
020422  012737  020064  001236          MOV     #04,@#$TMP2
020430  104077                  1$:     ERROR   77
020432  000440                          BR      ODONE

020434                          OERR6:                          ;DATA FAILURE
020434  012737  020060  001236          MOV     #02,@#$TMP2
020442  012737  020462  001240          MOV     #ODAT00,@#$TMP3
020450  012737  020524  001242          MOV     #ODAT10,@#$TMP4
020456  104100                  1$:     ERROR   100
020460  000425                          BR      ODONE

020462  000000          ODAT00: .WORD   0
020464  000000          ODAT01:         0
020466  000000          ODAT02:         0
020470  000000          ODAT03:         0

020472  052525  052525  052525          .WORD   52525,52525,52525
020500  020524          CPAT20: .WORD   ODAT10
020502  070707          OPAT21: 070707
020504  070707          OPAT22: 070707
020506  070707          OPAT23: 070707
020510  070707          OPAT24: 070707
020512  000001                  .WORD   1
020514  177777          OPAT10: .WORD   -1
020516  177777          OPAT11:         -1
020520  177777          OPAT12:         -1
020522  177777          OPAT13:         -1

020524  073567          ODAT10: .WORD   73567
020526  004210          ODAT11:         004210
020530  114631          ODAT12:         114631
020532  125252          ODAT13:         125252

020534                  ODONE:
```

```
020534  104412                          RSETUP                  ;GO INITIALIZE THE FPP AND STACK; AND
                                                                ;SEE IF THE USER HAS EXPRESSED
                                                                ;THE DESIRE TO CHANGE THE SOFTWARE
                                                                ;VIRTUAL CONSOLE SWITCH REGISTER (WAS
                                                                ;THE USER TYPED CONTROL G?).


                                ;;**************************************************************
                                ;*TEST 22       FSRC MODE 6 TEST
                                ;*
                                ;* THIS IS A TEST OF FSRC MODE 6, INDEX MODE
                                ;*
                                ;;**************************************************************
020536  000004                  TST22:  SCOPE

020540                          P1:
020540  170011                          SETD                    ;SET FD MODE

020542  012700  021160                  MOV     #PPAT10,R0
020546  172410                          LDD     (R0),AC0        ;LOAD A DEFAULT PATTERN
                                                                ;INTO AC0
020550  012737  020656  000004          MOV     #PERR0,@#ERRVECT        ;IF THE (BUT FSRC) FCRO
                                                                ;FAILS AN ODD ADDRESS TRAP
020556  012700  020727                  MOV     #PDAT10-241,R0  ;COULD OCCUR.

020562  172460  000241          P2:     LDD     241(R0),AC0
                020564          P3=P2+2

020566  012701  021200          P4:     MOV     #PDAT00,R1
020572  174011                          STD     AC0,(R1)        ;GET THE DATA
020574  012703  000004                  MOV     #4,R3
020600  012702  021170                  MOV     #PDAT10,R2
020604  012701  021200                  MOV     #PDAT00,R1
020610  022221          P5:     CMP     (R2)+,(R1)+     ;CHECK THE DATA
020612  001007                  BNE     P6
020614  077303                  SOB     R3,P5
020616  022700  020727                  CMP     #PDAT10-241.R0  ;R0 CORRECT?
020622  001401                  BEQ     1$
020624  000512                  BR      PERR21
020626  000137  021210          1$:     JMP     @#PDONE

020632  012701  021200          P6:     MOV     #PDAT00,R1
020636  012703  000004                  MOV     #4,R3
020642  022721  177777          P7:     CMP     #-1,(R1)+       ;WAS IT FSRC MODE 0?
020646  001401                  BEQ     P8
020650  000512                  BR      PERR1
020652  077305          P8:     SOB     R3,P7
020654  000523                  BR      PERR2
                                ;TRAP TO HERE ON AN ODD ADDRESS
020656  021627  020564          PERR0:  CMP     (SP),#P3
020662  001411                  BEQ     PERR11
020664  021627  020566                  CMP     (SP),#P4        ;WAS IT FSRC MODE 7?
020670  001402                  BEQ     PERR10
020672  000137  040232                  JMP     @#CPSPUR

020676  012737  000327  001246  PERR10: MOV     #327,@#$TMP6
```

```
 4826  020704  000443                        BR      PERR17
 4827  020706  022700  020727        PERR11: CMP     #PDATI0-241,R0  ;WAS IT FSRC MODE 1
 4828  020712  001004                        BNE     PERR12
 4829  020714  012737  000321  001246        MOV     #321,@#$TMP6
 4830  020722  000434                        BR      PERR17
 4831  020724  022700  020737        PERR12: CMP     #PDATI0-241+10,R0       ;WAS IT FSRC MODE 2
 4832  020730  001004                        BNE     PERR13
 4833  020732  012737  000322  001246        MOV     #322,@#$TMP6
 4834  020740  000425                        BR      PERR17
 4835  020742  022700  020731        PERR13: CMP     #PDATI0-241+2,R0        ;WAS IT FSRC MODE 3
 4836  020746  001004                        BNE     PERR14
 4837  020750  012737  000323  001246        MOV     #323,@#$TMP6
 4838  020756  000416                        BR      PERR17
 4839  020760  022700  020717        PERR14: CMP     #PDATI0-241-10,R0       ;WAS IT FSRC MODE 4
 4840  020764  001004                        BNE     PERR15
 4841  020766  012737  000324  001246        MOV     #324,@#$TMP6
 4842  020774  000407                        BR      PERR17
 4843  020776  022700  020725        PERR15: CMP     #PDATI0-241-2,R0        ;WAS IT FSRC MODE 5
 4844  021002  001401                        BEQ     PERR16
 4845  021004  000416                        BR      PERR20
 4846  021006  012737  000325  001246  PERR16: MOV     #325,@#$TMP6
 4847
 4848  021014  012737  000627  001244  PERR17: MOV     #627,@#$TMP5    ;REPORT FSRC
 4849  021022  012737  000326  001250        MOV     #326,@#$TMP7    ;FLOWS FAILURE.
 4850  021030  011637  001236                MOV     (SP),@#$TMP2
 4851  021034  022626                        CMP     (SP)+,(SP)+
 4852  021036  104101                  1$:   ERROR   101
 4853  021040  000463                        BR      PDONE
 4854
 4855  021042  011637  001236        PERR20: MOV     (SP),@#$TMP2    ;REPORT R0 AFFECTED
 4856  021046  022626                        CMP     (SP)+,(SP)+
 4857  021050  000403                        BR      PERR22
 4858  021052  012737  020562  001236  PERR21: MOV     #P2,@#$TMP2
 4859  021060                        PERR22:
 4860  021060  010037  001240                MOV     R0,@#$TMP3
 4861  021064  012737  020727  001242        MOV     #PDATI0-241,@#$TMP4
 4862  021072  104102                  1$:   ERROR   102
 4863  021074  000445                        BR      PDONE
 4864
 4865  021076                        PERR1:                          ;DATA FAILURE.
 4866  021076  012737  020562  001236        MOV     #P2,@#$TMP2
 4867  021104  012737  021170  001240        MOV     #PDATI0,@#$TMP3
 4868  021112  012737  021200  001242        MOV     #PDAT00,@#$TMP4
 4869  021120  104104                  1$:   ERROR   104
 4870  021122  000432                        BR      PDONE
 4871
 4872  021124                        PERR2:                          ;FSRC FAILURE TO
 4873  021124  012737  020562  001236        MOV     #P2,@#$TMP2     ;MODE 0
 4874  021132  012737  000627  001244        MOV     #627,@#$TMP5
 4875  021140  012737  000326  001250        MOV     #326,@#$TMP7
 4876  021146  012737  000320  001246        MOV     #320,@#$TMP6
 4877  021154  104103                  1$:   ERROR   103
 4878  021156  000414                        BR      PDONE
 4879
 4880  021160  177777                PPAT10: .WORD   -1
 4881  021162  177777                PPAT11:         -1
```

```
4882  021164  177777                  PPAT12:         -1
4883  021166  177777                  PPAT13:         -1
4884
4885  021170  010421                  PDATI0: .WORD   010421
4886  021172  031463                  PDATI1:         031463
4887  021174  052525                  PDATI2:         052525
4888  021176  073567                  PDATI3:         073567
4889
4890  021200  000000                  PDAT00: .WORD   0
4891  021202  000000                  PDAT01:         0
4892  021204  000000                  PDAT02:         0
4893  021206  000000                  PDAT03:         0
4894
4895  021210                          PDONE:
4896  021210  104412                          RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
4897                                                                  ;SEE IF THE USER HAS EXPRESSED
4898                                                                  ;THE DESIRE TO CHANGE THE SOFTWARE
4899                                                                  ;VIRTUAL CONSOLE SWITCH REGISTER .HAS
4900                                                                  ;THE USER TYPED CONTROL G?.
4901
4902
4903                                  ;;*****************************************************************
4904                                  ;*TEST 23       FSRC MODE 7 TEST
4905                                  ;*
4906                                  ;* THIS IS A TEST OF FSRC MODE 7, INDEX
4907                                  ;* DEFERRED MODE.
4908                                  ;*
4909                                  ;;*****************************************************************
4910  021212  000004                  ST23:   SCOPE
4911
4912  021214                          Q1:
4913  021214  170011                          SETD
4914
4915  021216  012700  021650                  MOV     #QPAT10,R0
4916  021222  172410                          LDD     (R0),AC0        ;LOAD A DEFAULT
4917                                                                  ;PATTERN INTO AC0
4918  021224  012737  021356  000004          MOV     #QERR0,@#ERRVECT        ;IF THE (BUT FSRC)
4919                                                                  ;FORK FAILS AN
4920                                                                  ;ODD ADR TRAP COULD
4921                                                                  ;OCCUR
4922  021232  012700  021417                  MOV     #QPAT20-241,R0
4923
4924  021236  172470  000241          Q2:     LDD     @241(R0),AC0
4925          021240                  Q3=Q2+2
4926
4927  021242  012701  021670          Q4:     MOV     #QDAT00,R1
4928  021246  174011                          STD     AC0,(R1)        ;GET THE DATA
4929
4930  021250  012703  000004                  MOV     #4,R3
4931  021254  012704  021670                  MOV     #QDAT00,R4
4932  021260  012705  021700                  MOV     #QDATI0,R5
4933  021264  022425                  Q5:     CMP     (R4)+,(R5)+     ;CHECK THE DATA
4934  021266  001007                          BNE     Q6
4935  021270  077303                          SOB     R3,Q5
4936
4937  021272  022700  021417                  CMP     #QPAT20-241,R0  ;CHECK R0.
```

```
4938  021276  001401                      BEQ     1$
4939  021300  000514                      BR      QERR21
4940  021302  000137  021710      1$:     JMP     @#QDONE
4941
4942  021306  012701  021670      Q6:     MOV     #QDAT00,R1
4943  021312  012703  000004              MOV     #4,R3
4944  021316  022721  177777      Q7:     CMP     #-1,(R1)+       ;WAS IT FSRC MODE 0?
4945  021322  001002                      BNE     Q8
4946  021324  077304                      SOB     R3,Q7
4947  021326  000513                      BR      QERR2
4948
4949  021330  012701  021660      Q8:     MOV     #QPAT20,R1
4950  021334  012702  021670              MOV     #QDAT00,R2
4951  021340  012703  000004              MOV     #4,R3
4952  021344  022122              Q9:     CMP     (R1)+,(R2)+     ;WAS IT FSRC 6
4953  021346  001401                      BEQ     Q10             ;OR DATA FAILURE
4954  021350  000524                      BR      QERR1
4955  021352  077304              Q10:    SOB     R3,Q9
4956  021354  000504                      BR      QERR3
4957
4958                                      ;TRAP TO HERE ON AN ODD ADR FAILURE
4959
4960  021356  021627  020564      QERR0:  CMP     (SP),#P3
4961  021362  000137  040232              JMP     @#CPSPUR
4962
4963  021366  022700  021417      QERR11: CMP     #QPAT20-241,R0  ;WAS IT FSRC
4964  021372  001004                      BNE     QERR12          ;MODE 1?
4965  021374  012737  000321  001246      MOV     #321,@#$TMP6
4966  021402  000434                      BR      QERR17
4967  021404  022700  021427      QERR12: CMP     #QPAT20-241+10,R0       ;WAS IT FSRC
4968  021410  001004                      BNE     QERR13          ;MODE 2?
4969  021412  012737  000322  001246      MOV     #322,@#$TMP6
4970  021420  000425                      BR      QERR17
4971  021422  022700  021421      QERR13: CMP     #QPAT20-241+2,R0        ;WAS IT FSRC
4972  021426  001004                      BNE     QERR14          ;MODE 3?
4973  021430  012737  000323  001246      MOV     #323,@#$TMP6
4974  021436  000416                      BR      QERR17
4975  021440  022700  021407      QERR14: CMP     #QPAT20-241-10,R0       ;WAS IT FSRC
4976  021444  001004                      BNE     QERR15          ;MODE 4
4977  021446  012737  000324  001246      MOV     #324,@#$TMP6
4978  021454  000407                      BR      QERR17
4979
4980  021456  022700  021415      QERR15: CMP     #QPAT20-241-2,R0        ;WAS IT FSRC
4981  021462  001401                      BEQ     QERR16          ;MODE 5
4982  021464  000416                      BR      QERR20
4983
4984  021466  012737  000325  001246  QERR16: MOV     #325,@#$TMP6
4985
4986  021474  012737  000627  001244  QERR17: MOV     #627,@#$TMP5    ;REPORT FSRC FAILURE
4987  021502  012737  000327  001250          MOV     #327,@#$TMP7
4988  021510  011637  001236                  MOV     (SP),@#$TMP2
4989  021514  022626                          CMP     (SP)+,(SP)+
4990  021516  104105              1$:     ERROR   105
4991  021520  000473                      BR      QDONE
4992
4993  021522  011637  001236      QERR20: MOV     (SP),@#$TMP2    ;REPORT R0 AFFECTED.
```

```
4994  021526  022626                          CMP     (SP)+,(SP)+
4995  021530  000403                          BR      QERR22
4996  021532  012737  021236  001236  QERR21: MOV     #Q2,@#$TMP2
4997  021540                          QERR22:
4998  021540  010037  001240                  MOV     R0,@#$TMP3
4999  021544  012737  021417  001242          MOV     #QPAT20-241,@#$TMP4
5000  021552  104106                  1$:     ERROR   106
5001  021554  000455                          BR      QDONE
5002
5003  021556  012737  000320  001246  QERR2:  MOV     #320,@#$TMP6    ;WENT TO FSRC
5004  021564  000403                          BR      QERR4           ;MODE 0
5005  021566  012737  000326  001246  QERR3:  MOV     #326,@#$TMP6    ;WENT TO FSRC
5006                                                                  ;MODE 6
5007  021574  012737  000627  001244  QERR4:  MOV     #627,@#$TMP5
5008  021602  012737  000327  001250          MOV     #327,@#$TMP7
5009  021610  012737  021236  001236          MOV     #Q2,@#$TMP2
5010  021616  104107                  1$:     ERROR   107
5011  021620  000433                          BR      QDONE
5012
5013  021622                          QERR1:                          ;DATA FAILURE
5014  021622  012737  021236  001236          MOV     #Q2,@#$TMP2
5015  021630  012737  021700  001240          MOV     #QDATI0,@#$TMP3
5016  021636  012737  021670  001242          MOV     #QDAT00,@#$TMP4
5017  021644  104110                  1$:     ERROR   110
5018  021646  000420                          BR      QDONE
5019
5020  021650  177777                  QPAT10: .WORD   -1
5021  021652  177777                  QPAT11:         -1
5022  021654  177777                  QPAT12:         -1
5023  021656  177777                  QPAT13:         -1
5024
5025  021660  021700                  QPAT20: .WORD   QDATI0
5026  021662  052525                  QPAT21:         52525
5027  021664  052525                  QPAT22:         52525
5028  021666  052525                  QPAT23:         52525
5029
5030  021670  000000                  QDAT00: .WORD   0
5031  021672  000000                  QDAT01:         0
5032  021674  000000                  QDAT02:         0
5033  021676  000000                  QDAT03:         0
5034
5035  021700  073567                  QDATI0: .WORD   073567
5036  021702  052525                  QDATI1: .WORD   052525
5037  021704  031463                  QDATI2: .WORD   031463
5038  021706  010421                  QDATI3: .WORD   010421
5039
5040  021710                          QDONE:
5041  021710  104412                          RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
5042                                                                  ;SEE IF THE USER HAS EXPRESSED
5043                                                                  ;THE DESIRE TO CHANGE THE SOFTWARE
5044                                                                  ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
5045                                                                  ;THE USER TYPED CONTROL G^).
5046
5047                                  ;;*****************************************************************
5048                                  ;*TEST 24       (BUT EZBT Y8),(BUT ENBT) AND (BUT FIUV) TEST
5049                                  ;*
```

```
                                        ;* 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.
                                        ;*
                                        ;;*****************************************************************
021712  000004                  T24:    SCOPE
021714  005037  023012                  CLR     @#UFLAG
021720  012700  022742                  MOV     #UPAT00,R0      ;SET UP AC#0 DATA.
021724  012701  000004                  MOV     #4,R1
021730  012720  177777          .       MOV     #-1,(R0)+
021734  077103                  U0:     SOB     R1,U0
                                
021736  012737  000033  023014          MOV     #033,@#UTMP1
021744  012737  000023  023016          MOV     #023,@#UTMP2
021752  012737  022472  000244          MOV     #UERR0,@#FPVECT ;IN CASE (BUT FIUV FAILS)
021760                          U1:
021760  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
021762  012700  000200                  MOV     #200,R0
021766  170100                          LDFPS   R0
021770  012700  022742                  MOV     #UPAT00,R0      ;LOAD ACO
021774  172410                          LDD     (R0),AC0
021776  013737  023014  023020          MOV     @#UTMP1,@#UROM1
022004  012737  000001  023022          MOV     #001,@#UROM2
022012  012737  000254  023024          MOV     #254,@#UROM3
                                
022020  012700  022752                  MOV     #UPAT10,R0      ;LOAD C INTO ACO
022024  172410                  U2:     LDD     (R0),AC0
022026  010037  001252                  MOV     R0,@#STMP10
022032  012737  022024  001236          MOV     #U2,@#STMP2
                                
022040  012704  000204                  MOV     #204,R4         ;SEE IF FPS IS CORRECT
022044  170205                          STFPS   R5
                                
022046  020405                          CMP     R4,R5
022050  001402                          BEQ     U3
022052  000137  022516                  JMP     @#UERR1
                                
022056                          U3:
022056  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
022060  012700  000200                  MOV     #200,R0
022064  170100                          LDFPS   R0
                                
022066  012700  022742                  MOV     #UPAT00,R0      ;LOAD ACO
022072  172410                          LDD     (R0),AC0
022074  013737  023016  023020          MOV     @#UTMP2,@#UROM1
022102  012737  000003  023022          MOV     #003,@#UROM2
022110  012737  000054  023024          MOV     #054,@#UROM3
```

```
022116  012700  022762          MOV     #UPAT2D,R0      ;LOAD A POSITIVE NUMBER
                                                        ;INTO ACD
022122  172410                  U4:     LDD     (R0),AC0
022124  010037  001252                  MOV     R0,@#STMP10
022130  012737  022122  001236          MOV     #U4,@#STMP2
022136  012704  000200                  MOV     #200,R4         ;FPS CORRECT
022142  170205                  STFPS   R5

022144  020405                  CMP     R4,R5
022146  001402                  BEQ     U5
022150  000137  022602          JMP     @#UERR2
022154                          U5:
022154  104413                  LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
022156  012700  000200          MOV     #200,R0
022162  170100                  LDFPS   R0
022164  012700  022742          MOV     #UPAT0D,R0      ;LOAD ACD
022170  172410                  LDD     (R0),AC0
022172  013737  023016  023020  MOV     @#UTMP2,@#UROM1
022200  012737  000403  023022  MOV     #403,@#UROM2
022206  012737  000056  023024  MOV     #056,@#UROM3
022214  012700  022772          MOV     #UPAT3D,R0      ;LOAD A NEGATIUE
                                                        ;NUMBER INTO ALC
022220  172410                  U6:     LDD     (R0),AC0
022222  010037  001252          MOV     R0,@#STMP10
022226  012737  022220  001236  MOV     #U6,@#STMP2
022234  012704  000210          MOV     #210,R4         ;FPS CORRECT
022240  170205                  STFPS   R5
022242  020405                  CMP     R4,R5
022244  001402                  BEQ     U7
022246  000137  022602          JMP     @#UERR2
022252                          U7:
022252  104413                  LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
022254  012700  000200          MOV     #200,R0
022260  170100                  LDFPS   R0
022262  012700  022742          MOV     #UPAT0D,R0      ;LOAD ACD
022266  172410                  LDD     (R0),AC0
022270  013737  023014  023020  MOV     @#UTMP1,@#UROM1
022276  012737  000401  023022  MOV     #401,@#UROM2
022304  012737  000256  023024  MOV     #256,@#UROM3
022312  012700  023002          MOV     #UPAT4D,R0      ;LOAD -0 INTO ACD
022316  172410                  U10:    LDD     (R0),AC0
022320  000240                  U11:    NOP                     ;TRAP FROM HERE IF
022322  010037  001252          MOV     R0,@#STMP10
022326  012737  022316  001236  MOV     #U10,@#STMP2    ;(BUT FIUV) FAULS!
022334  012704  000214          MOV     #214,R4         ;SEE IF FPS IS CORRECT.
022340  170205                  STFPS   R5
022342  020405                  CMP     R4,R5
022344  001402                  BEQ     U12
022346  000137  022516          JMP     @#UERR1
022352  005737  023012          U12:    TST     @#UFLAG ;SEE IF ALL THE PATTERNS
022356  001021                  BNE     U14             ;HAVE BEEN TEST WITH
                                                        ;BOTH AC NOT EQUAL TO 0 AND AC=0
022360  012700  022742          MOV     #UPAT0D,R0      ;IF NOT GO BACK AND
022364  012701  000004          MOV     #4,R1           ;CHECK THEM WITH AC=0
022370  005020                  U13:    CLR     (R0)+
```

```
022372  077102                                SOB     R1,U13
022374  012737  177777  023012                MOV     #-1,@#UFLAG
022402  012737  000233  023014                MOV     #233,@#UTMP1
022410  012737  000223  023016                MOV     #223,@#UTMP2
022416  000137  021760                        JMP     @#U1
022422                                U14:
022422  104413                                LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
                                      ;NOW SEE IF A TRAP CAN BE FORCED BY SETTING FIUV AND LOADING -0
022424  012737  022666  000244                MOV     #UERR3,@#FPVECT
022432  012700  004200                        MOV     #4200,R0        ;SET FD AND FIUV
022436  170100                                LDFPS   R0
022440  012700  022742                        MOV     #UPAT00,R0      ;SET UP AC0
022444  172410                                LDD     (R0),AC0
022446  012700  023002                        MOV     #UPAT40,R0      ;LOAD -0
022452  172410                        U15:    LDD     (R0),AC0        ;SHOULD TRAP TO 244
022454  170000                        U16:    CFCC
022456  000240                                NOP
022460  012737  022452  001236                MOV     #U15,@#$TMP2    ;REPORT ERROR.
                                                                      ;DIDN'T TRAP
022466  104127                        1$:     ERROR   127             ;(BUT FIUV) FAILED.
022470  000556                                BR      UDONE

                                      ;TRAPPED TO 244.  DID (BUT FIUV) FAIL?
022472  021627  022320                UERR0:  CMP     (SP),#U11
022476  001402                                BEQ     1$
022500  000137  040200                        JMP     @#FPSPUR
022504  011637  001236                1$:     MOV     (SP),@#$TMP2
022510  022626                                CMP     (SP)+,(SP)+
022512  104126                        2$:     ERROR   126
022514  000544                                BR      UDONE

                                      ;COME HERE TO ANALYZE FPS ERRORS

022516  032705  000004                UERR1:  BIT     #4,R5
022522  001432                                BEQ     UERR20
022524  012737  000443  001244        UERR10: MOV     #443,@#$TMP5
022532  013703  023024                        MOV     @#UROM3,R3
022536  010337  001250                        MOV     R3,@#$TMP7
022542  032703  000200                        BIT     #200,R3
022546  001403                                BEQ     1$
022550  042703  000200                        BIC     #200,R3
022554  000402                                BR      2$
022556  052703  000200                1$:     BIS     #200,R3
022562  010337  001246                2$:     MOV     R3,@#$TMP6
022566  010537  001240                UERR11: MOV     R5,@#$TMP3
022572  010437  001242                        MOV     R4,@#$TMP4
022576  104124                        1$:     ERROR   124
022600  000512                                BR      UDONE
022602  032705  000004                UERR2:  BIT     #4,R5
022606  001746                                BEQ     UERR10
022610  013737  023020  001244        UERR20: MOV     @#UROM1,@#$TMP5
022616  013703  023022                        MOV     @#UROM2,R3
022622  010337  001250                        MOV     R3,@#$TMP7
022626  032703  000400                        BIT     #400,R3
022632  001403                                BEQ     1$
022634  042703  000400                        BIC     #400,R3
```

```
022640  000402                          BR      2$
022642  052703  000400          1$:     BIS     #400,R3
022646  010337  001246          2$:     MOV     R3,@#$TMP6
022652  010537  001240          UERR2:: MOV     R5,@#$TMP3
022656  010437  001242                  MOV     R4,@#$TMP4
022662  104125                  1$:     ERROR   125
022664  000462                          BR      UDONE

                                ;INTERRUPT HERE WHEN FIUV SET AND ATTEMPTED TO LOAD-0
022666  021627  022454          UERR3:  CMP     (SP),#U16
022672  001402                          BEQ     1$
022674  000137  040200                  JMP     @#FPSPUR
022700  022626                  1$:     CMP     (SP)+,(SP)+
022702  005000                          CLR     R0
022704  170300                          STST    R0                      ;GET FEC.
022706  022700  000014                  CMP     #14,R0                  ;CORRECT
022712  001001                          BNE     UERR4
022714  000444                          BR      UDONE
022716  012737  022452  001236  UERR4:  MOV     #U15,@#$TMP2
022724  012737  000012  001242          MOV     #12,@#$TMP4
022732  010037  001240                  MOV     R0,@#$TMP3
022736  104130                  1$:     ERROR   130
022740  000432                          BR      UDONE
022742  000000                  UPAT00: .WORD   0
022744  000000                  UPAT01:         0
022746  000000                  UPAT02:         0
022750  000000                  UPAT03:         0

022752  000000                  UPAT10: .WORD   0                       ;0
022754  000000                  UPAT11:         0
022756  000000                  UPAT12:         0
022760  000000                  UPAT13:         0

022762  010421                  UPAT20: .WORD   010421                  ;POS NUM
022764  114631                  UPAT21:         114631
022766  125252                  UPAT22:         125252
022770  177777                  UPAT23:         177777

022772  114631                  UPAT30:         114631                  ;NEG NUM
022774  135673                  UPAT31:         135673
022776  146314                  UPAT32:         146314
023000  167356                  UPAT33:         167356

023002  100000                  UPAT40:         100000                  ;NEG ZERO
023004  000000                  UPAT41:         0
023006  000000                  UPAT42:         0
023010  000000                  UPAT43:         0

023012  000000                  UFLAG:  .WORD   0
023014  000000                  UTMP1:          0
023016  000000                  UTMP2:          0
023020  000000                  UROM1:          0
023022  000000                  UROM2:          0
023024  000000                  UROM3:          0
023026                          UDONE:
```

F08


```
                                        ;;*************************************************************
                                        ;;*TEST 25      ADDF,ADDD,SUBF AND SUBD WITH FSRC=AC=0 TEST
                                        ;;*
                                        ;;* THIS IS A TEST OF ADD AND SUB WITH FSRC=AC=0
                                        ;;*
                                        ;;*************************************************************
023026  000004                          TST25:  SCOPE
023030                                  W1:
023030  104413                                  LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
023032  012700  000200                          MOV     #200,R0
023036  170100                                  LDFPS   R0              ;SET DOUBLE MODE
023040  012700  023562                          MOV     #WPAT00,R0      ;LOAD AC0=:
023044  172410                                  LDD     (R0),AC0
023046  012737  023060  001236                  MOV     #W2,@#$TMP2
023054  012700  023562                          MOV     #WPAT00,R0
023060  172010                          W2:     ADDD    (R0),AC0        ;TEST INSTRUCTION.
023062  170205                                  STFPS   R5              ;GET FPS
023064  170011                                  SETD                    ;SET DOUBLE MODE
023066  012700  023562                          MOV     #WPAT00,R0
023072  174010                                  STD     AC0,(R0)        ;GET THE RESULT
023074  012701  023562                          MOV     #WPAT00,R1
023100  012702  000004                          MOV     #4,R2
023104  022021                          W3:     CMP     (R0)+,(R1)+     ;IS RESULT CORRECT
023106  001405                                  BEQ     W4
                                                                        ;NO
023110  004737  023530                          JSR     PC,@#WSETUP
023114  104133                          1$:     ERROR   133
023116  000137  023602                          JMP     @#WDONE
023122  077210                          W4:     SOB     R2,W3
023124  022705  000204                          CMP     #204,R5         ;IS FPS CORRECT
023130  001410                                  BEQ     W5
                                                                        ;NO
023132  012737  000204  001242                  MOV     #204,@#$TMP4
023140  010537  001240                          MOV     R5,@#$TMP3
023144  104137                          1$:     ERROR   137
023146  000137  023602                          JMP     @#WDONE
023152                                  W5:
023152  104413                                  LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
023154  012700  000200                          MOV     #200,R0
023160  170100                                  LDFPS   R0              ;SET DOUBLE MODE
023162  012700  023562                          MOV     #WPAT00,R0      ;LOAD AC0=0
023166  172410                                  LDD     (R0),AC0
023170  012737  023206  001236                  MOV     #W6,@#$TMP2
023176  005000                                  CLR     R0
023200  170100                                  LDFPS   R0              ;GO TO FLOATING MODE
023202  012700  023562                          MOV     #WPAT00,R0
023206  172010                          W6:     ADDF    (R0),AC0        ;TEST INSTRUCTION
023210  170205                                  STFPS   R5              ;GET FPS
023212  170011                                  SETD                    ;RESET TO DOUBLE MODE
023214  012700  023562                          MOV     #WPAT00,R0
023220  174010                                  STD     AC0,(R0)        ;GET THE RESULT
023222  012701  023562                          MOV     #WPAT00,R1
023226  012702  000004                          MOV     #4,R2
```

```
                                W7:     CMP     (R0)+,(R1)+     ;WAS THE RESULT
                                        BEQ     W10             ;NO. REPORT FAILURE.
                                1$:     ERROR   134
                                        BR      WDONE
                                W10:    SOB     R2,W7
                                        CMP     #4,R5           ;WAS FPS CORRECT
023252  001407                          BEQ     W11
                                                                ;INCORRECT FPS.
023254  012737  000004  001242          MOV     #4,@#$TMP4
023262  010537  001240                  MOV     R5,@#$TMP3
023264  104140                  1$:     ERROR   140
023266  000545                          BR      WDONE
023270                          W11:
023270  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
023272  012700  000200                  MOV     #200,R0
023276  170100                          LDFPS   R0              ;SET DOUBLE MODE
023300  012700  023562                  MOV     #WPAT00,R0      ;LOAD AC0=0
023304  172410                          LDD     (R0),AC0
023306  012737  023320  001236          MOV     #W12,@#$TMP2
023314  012700  023562                  MOV     #WPAT00,R0
023320  173010                  W12:    SUBD    (R0),AC0        ;TEST INSTRUCTION
023322  170205                          STFPS   R5              ;GET FPS
023324  170011                          SETD                    ;SET DOUBLE MODE
023326  012700  023562                  MOV     #WPAT00,R0
023332  174010                          STD     AC0,(R0)        ;GET THE RESULT
023334  012701  023562                  MOV     #WPAT00,R1
023340  012702  000004                  MOV     #4,R2
023344  022021                  W13:    CMP     (R0)+,(R1)+     ;IS RESULT CORRECT?
023346  001404                          BEQ     W14
                                                                ;NO.
023350  004737  023530                  JSR     PC,@#WSETUP
023354  104135                  1$:     ERROR   135
023356  000511                          BR      WDONE
023360  077207                  W14:    SOB     R2,W13
023362  022705  000204                  CMP     #204,R5         ;IS FPS CORRECT
023366  001407                          BEQ     W15
                                                                ;NO.
023370  012737  000204  001242          MOV     #204,@#$TMP4
023376  010537  001240                  MOV     R5,@#$TMP3
023402  104141                  1$:     ERROR   141
023404  000476                          BR      WDONE
023406                          W15:
023406  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
023410  012700  000200                  MOV     #200,R0
023414  170100                          LDFPS   R0              ;SET DOUBLE MODE
023416  012700  023562                  MOV     #WPAT00,R0      ;LOAD AC0=0
023422  172410                          LDD     (R0),AC0
023424  012737  023442  001236          MOV     #W16,@#$TMP2
023432  005000                          CLR     R0
023434  170100                          LDFPS   R0              ;ENTER FLOATING MODE.
023436  012700  023562                  MOV     #WPAT00,R0
023442  173010                  W16:    SUBF    (R0),AC0        ;TEST INSTRUCTION.
023444  170205                          STFPS   R5              ;GET FPS
023446  170011                          SETD                    ;RESET TO DOUBLE MODE
023450  012700  023562                  MOV     #WPAT00,R0      ;GET THE RESULT.
023454  174010                          STD     AC0,(R0)
```

```
023456 012701 023562                     MOV     #WPAT00,R1
023462 012702 000004                     MOV     #4,R2
023466 022021                    W17:    CMP     (R0)+,(R1)+         ;IS RESULT CORRECT?
023470 001404                            BEQ     W20
                                                                     ;NO.
023472 004737 023530                     JSR     PC,@#WSETUP
023476 104136                    1$:     ERROR   136
023500 000440                            BR      WDONE
023502 077207                    W20:    SOB     R2,W17
023504 022705 000004                     CMP     #4,R5               ;IS FPS CORRECT?
023510 001434                            BEQ     WDONE
                                                                     ;NO
023512 012737 000004 001242              MOV     #4,@#$TMP4
023520 010537 001240                     MOV     R5,@#$TMP3
023524 104142                    1$:     ERROR   142
023526 000425                            BR      WDONE

                                 ;SET UP FOR ERROR CALL

023530 012737 023562 001240      WSETUP: MOV     #WPAT00,@#$TMP3
023536 012737 023562 001242              MOV     #WPAT00,@#$TMP4
023544 012737 023562 001246              MOV     #WPAT00,@#$TMP6
023552 012737 023562 001244              MOV     #WPAT00,@#$TMP5
023560 000207                            RTS     PC
023562 000000                    WPAT00: .WORD   0
023564 000000                    WPAT01:         0
023566 000000                    WPAT02:         0
023570 000000                    WPAT03:         0

023572 000000                    WDAP00: .WORD   0
023574 000000                    WDAT01:         0
023576 000000                    WDAT02:         0
023600 000000                    WDAT03:         0

023602                           WDONE:
023602 104412                            RSETUP                      ;GO INITIALIZE THE FPS AND STACK; AND
                                                                     ;SEE IF THE USER HAS EXPRESSED
                                                                     ;THE DESIRE TO CHANGE THE SOFTWARE
                                                                     ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
                                                                     ;THE USER TYPED CONTROL G?).


                                 ;;*************************************************************
                                 ;*TEST 26        ADDD AND SUB WITH FSRC=0
                                 ;*
                                 ;* THIS IS A TEST OF ADD AND SUB WITH FSRC=0.
                                 ;*

                                 ;;*************************************************************
023604 000004                    TST26:  SCOPE
023606                           X1:
023606 104413                            LPERR                       ;SET UP THE LOOP ON ERROR ADDRESS.
023610 012700 000200                     MOV     #200,R0
023614 170100                            LDFPS   R0                  ;SET DOUBLE MODE
023616 012700 024346                     MOV     #XPAT00,R0          ;SET AC0 TO POSITIVE
023622 010037 024334                     MOV     R0,@#XTMP           ;NUMBER #0
```

```
023626  172410                      LDD     (R0),AC0
023630  012737  023642  001236      MOV     #X2,@#$TMP2
023636  012700  024356              MOV     #XPAT10,R0      ;FSRC=0
023642  172010              X2:     ADDD    (R0),AC0        ;TEST INSTRUCTION
023644  170205                      STFPS   R5
023646  170011                      SETD
023650  012700  024336              MOV     #XDAT00,R0      ;GET RESULT.
023654  174010                      STD     AC0,(R0)
023656  012701  024346              MOV     #XPAT00,R1
023662  012702  000004              MOV     #4,R2
023666  022021              X3:     CMP     (R0)+,(R1)+     ;IS RESULT CORRECT?
023670  001401                      BEQ     X4
023672  000553                      BR      XERR1
023674  077204              X4:     SOB     R2,X3
023676  012704  000200              MOV     #200,R4
023702  020405                      CMP     R4,R5           ;IS FPS CORRECT?
023704  001402                      BEQ     X5
023706  000137  024304              JMP     @#XERR2
023712              X5:
023712  104413                      LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
023714  012700  000200              MOV     #200,R0
023720  170100                      LDFPS   R0              ;SET DOUBLE MODE
023722  012700  024366              MOV     #XPAT20,R0      ;SET ACO TO
023726  010037  024334              MOV     R0,@#XTMP       ;NEGATIVE NUMBER
023732  172410                      LDD     (R0),AC0
023734  012737  023746  001236      MOV     #X6,@#$TMP2
023742  012700  024356              MOV     #XPAT10,R0      ;FSRC=0
023746  172010              X6:     ADDD    (R0),AC0        ;TEST INSTRUCTION
023750  170205                      STFPS   R5
023752  170011                      SETD
023754  012700  024336              MOV     #XDAT00,R0      ;GET RESULT
023760  174010                      STD     AC0,(R0)
023762  012701  024366              MOV     #XPAT20,R1
023766  012702  000004              MOV     #4,R2
023772  022021              X7:     CMP     (R0)+,(R1)+     ;IS RESULT CORRECT?
023774  001401                      BEQ     X10
023776  000511                      BR      XERR1
024000  077204              X10:    SOB     R2,X7
024002  012704  000210              MOV     #210,R4
024006  020405                      CMP     R4,R5           ;IS FPS CORRECT?
024010  001401                      BEQ     X11
024012  000534                      BR      XERR2
024014              X11:
024014  104413                      LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
024016  012700  000200              MOV     #200,R0
024022  170100                      LDFPS   R0              ;SET DOUBLE MODE
024024  012700  024346              MOV     #XPAT00,R0      ;SET ACO TO NON-ZERO
024030  010037  024334              MOV     R0,@#XTMP       ;POSITIVE NUMBER
024034  172410                      LDD     (R0),AC0
024036  012737  024050  001236      MOV     #X12,@#$TMP2
024044  012700  024356              MOV     #XPAT10,R0      ;FSRC=0
024050  173010              X12:    SUBD    (R0),AC0        ;TEST INSTRUCTION
024052  170205                      STFPS   R5
024054  170011                      SETD
024056  012700  024336              MOV     #XDAT00,R0      ;GET RESULT
024062  174010                      STD     AC0,(R0)
```

```
024064 012701 024346                    MOV     #XPAT00,R1
024070 012702 000004                    MOV     #4,R2
024074 022021                   X13:    CMP     (R0)+,(R1)+     ;IS RESULT CORRECT?
024076 001401                           BEQ     X14
024100 000463                           BR      XERR3
024102 077204                   X14:    SOB     R2,X13
024104 012704 000200                    MOV     #200,R4         ;IS FPS CORRECT?
024110 020405                           CMP     R4,R5
024112 001401                           BEQ     X15
024114 000501                           BR      XERR4
024116                          X15:
024116 104413                           LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
024120 012700 000200                    MOV     #200,R0
024124 170100                           LDFPS   R0              ;SET DOUBLE MODE
024126 012700 024366                    MOV     #XPAT20,R0      ;SET ACC=A NEGATIVE
024132 010037 024334                    MOV     R0,@#XTMP       ;NUMBER
024136 172410                           LDD     (R0),AC0
024140 012737 024152 001236             MOV     #X16,@#$TMP2
024146 012700 024356                    MOV     #XPAT10,R0      ;FSRC=0
024152 173010                   X16:    SUBD    (R0),AC0        ;TEST INSTRUCTION.
024154 170205                           STFPS   R5
024156 170011                           SETD
024160 012700 024336                    MOV     #XDAT00,R0      ;GET RESULT
024164 174010                           STD     AC0,(R0)
024166 012701 024366                    MOV     #XPAT20,R1
024172 012702 000004                    MOV     #4,R2
024176 022021                   X17:    CMP     (R0)+,(R1)+     ;IS RESULT CORRECT?
024200 001401                           BEQ     X20
024202 000422                           BR      XERR3
024204 077204                   X20:    SOB     R2,X17
024206 012704 000210                    MOV     #210,R4         ;IS FPS CORRECT?
024212 020405                           CMP     R4,R5
024214 001401                           BEQ     X21
024216 000440                           BR      XERR4
024220 000466                   X21:    BR      XDONE

                                ;REPORT DATA ERRORS

024222 012737 024356 001240     XERR1:  MOV     #XPAT10,@#$TMP3
024230 013737 024334 001242             MOV     @#XTMP,@#$TMP4
024236 012737 024336 001244             MOV     #XDAT00,@#$TMP5
024244 104143                   1$:     ERROR   143
024246 000453                           BR      XDONE
024250 012737 024356 001240     XERR3:  MOV     #XPAT10,@#$TMP3
024256 013737 024334 001242             MOV     @#XTMP,@#$TMP4
024264 012737 024336 001244             MOV     #XDAT00,@#$TMP5
024272 013737 024334 001246             MOV     @#XTMP,@#$TMP6
024300 104144                   1$:     ERROR   144
024302 000435                           BR      XDONE

                                ;REPORT FPS ERRORS

024304                          XERR2:
024304 010537 001240                    MOV     R5,@#$TMP3
024310 010437 001242                    MOV     R4,@#$TMP4
024314 104145                   1$:     ERROR   145
```

```
                        T26     ADDD AND SUB WITH FSRC=0

5554   024316  000427                   BR      XDONE
5555   024320                  XERR4:
5556   024320  010537  001240           MOV     R5,@#$TMP3
5557   024324  010437  001242           MOV     R4,@#$TMP4
5558   024330  104146          1$:      ERROR   146
5559   024332  000421                   BR      XDONE
5560   024334  000000          XTMP:    .WORD   0
5561   024336  000000          XDAT00:  .WORD   0
5562   024340  000000          XDAT01:          0
5563   024342  000000          XDAT02:          0
5564   024344  000000          XDAT03:          0
5565
5566   024346  010421          XPAT00:  .WORD   010421
5567   024350  021042          XPAT01:          021042
5568   024352  031463          XPAT02:          031463
5569   024354  042104          XPAT03:          042104
5570
5571   024356  000000          XPAT10:  .WORD   0
5572   024360  000000          XAPT11:          0
5573   024362  000000          XPAT12:          0
5574   024364  000000          XPAT13:          0
5575   024366  104210          XPAT20:  .WORD   104210
5576   024370  114631          XPAT21:          114631
5577   024372  125252          XPAT22:          125252
5578   024374  135673          XPAT23:          135673
5579
5580   024376                  XDONE:
5581   024376  104412                   RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
5582                                                            ;SEE IF THE USER HAS EXPRESSED
5583                                                            ;THE DESIRE TO CHANGE THE SOFTWARE
5584                                                            ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
5585                                                            ;THE USER TYPED CONTROL G?).
5586
5587                            ;;*****************************************************************
5588                            ;*TEST 27        SUBD WITH AC=0 TEST
5589                            ;*
5590                            ;* THIS IS A TEST OF SUBD WITH AC=0.  BOTH POSITIVE
5591                            ;* AND NEGATIVE FSRC'S ARE TRIED.
5592                            ;*
5593
5594                            ;;*****************************************************************
5595   024400  000004          TST27:   SCOPE
5596   024402  005037  024732           CLR     @#YFLAG
5597   024406  012737  024752  024734   MOV     #YPAT00,@#YTMP1 ;P
5598   024414  012737  024762  024736   MOV     #YPAT10,@#YTMP2 ;N
5599   024422  012737  000210  024740   MOV     #210,@#YTMP3
5600   024430                  Y1:
5601   024430  104413                   LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
5602   024432  012700  000200           MOV     #200,R0
5603   024436  170100                   LDFPS   R0              ;SET DOUBLE MODE
5604   024440  012700  024772           MOV     #YPAT20,R0      ;SET AC0=0
5605   024444  172410                   LDD     (R0),AC0
5606   024446  013700  024734           MOV     @#YTMP1,R0
5607   024452  173010          Y2:      SUBD    (R0),AC0        ;TEST INSTRUCTION
5608   024454  170205                   STFPS   R5
5609   024456  170C11                   SETD
```

```
5610  024460  012700  024742            MOV     #YDAT00,R0      ;GET RESULT
5611  024464  174010                    STD     AC0,(R0)
5612  024466  012702  000004            MOV     #4,R2
5613  024472  013701  024736            MOV     @#YTMP2,R1      ;CHECK RESULT.
5614  024476  022021            Y3:     CMP     (R0)+,(R1)+
5615  024500  001026                    BNE     Y6
5616  024502  077203                    SOB     R2,Y3
5617  024504  023705  024740            CMP     @#YTMP3,R5      ;FPS CORRECT?
5618  024510  001401                    BEQ     Y4
5619  024512  000475                    BR      YERR3
5620  024514  005737  024732    Y4:     TST     @#YFLAG         ;FINISHED TEST?
5621  024520  001015                    BNE     Y5
5622  024522  012737  177777  024732    MOV     #-1,@#YFLAG
5623  024530  012737  024762  024734    MOV     #YPAT10,@#YTMP1
5624  024536  012737  024752  024736    MOV     #YPAT00,@#YTMP2
5625  024544  012737  000200  024740    MOV     #200,@#YTMP3
5626  024552  000726                    BR      Y1
5627  024554  000512            Y5:     BR      YDONE
5628  024556  012702  000004    Y6:     MOV     #4,R2
5629  024562  012700  024734            MOV     #YTMP1,R0       ;DID XOR OF SIGN BIT
5630  024566  012701  024742            MOV     #YDAT00,R1      ;FAIL?
5631  024572  022021            Y7:     CMP     (R0)+,(R1)+
5632  024574  001002                    BNE     YERR1
5633  024576  077203                    SOB     R2,Y7
5634  024600  000421                    BR      YERR2
5635  024602            YERR1:                                  ;DATA FAILURE
5636  024602  012737  024452  001236    MOV     #Y2,@#$TMP2
5637  024610  013737  024734  001240    MOV     @#YTMP1,@#$TMP3
5638  024616  012737  024772  001242    MOV     #YPAT20,@#$TMP4
5639  024624  012737  024742  001244    MOV     #YDAT00,@#$TMP5
5640  024632  013737  024736  001246    MOV     @#YTMP2,@#$TMP6
5641  024640  104147            1$:     ERROR   147
5642  024642  000457                    BR      YDONE
5643  024644            YERR2:                                  ;XOR OF SIGN BIT
5644  024644  012737  024452  001236    MOV     #Y2,@#$TMP2     ;FAILED
5645  024652  013737  024734  001240    MOV     @#YTMP1,@#$TMP3
5646  024660  012737  024772  001242    MOV     #YPAT20,@#$TMP4
5647  024666  012737  024742  001244    MOV     #YDAT00,@#$TMP5
5648  024674  013737  024736  001246    MOV     @#YTMP2,@#$TMP6
5649  024702  104150            1$:     ERROR   150
5650  024704  000436                    BR      YDONE
5651  024706            YERR3:                                  ;FPS WRONG.
5652  024706  012737  024452  001236    MOV     #Y2,@#$TMP2
5653  024714  010537  001240            MOV     R5,@#$TMP3
5654  024720  013737  024740  001242    MOV     @#YTMP3,@#$TMP4
5655  024726  104151            1$:     ERROR   151
5656  024730  000424                    BR      YDONE
5657
5658  024732  000000    YFLAG:  .WORD   0
5659  024734  000000    YTMP1:          0
5660  024736  000000    YTMP2:          0
5661  024740  000000    YTMP3:          0
5662
5663  024742  000000    YDAT00: .WORD   0
5664  024744  000000    YDAT01:         0
5665  024746  000000    YDAT02:         0
```

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

```
5722  025112  000437                          BR      ZERR2
5723  025114  005737  025236          Z5:     TST     @#ZFLAG         ;FINISHED TEST?
5724  025120  001012                          BNE     Z6
5725  025122  012737  177777  025236          MOV     #-1,@#ZFLAG
5726  025130  012737  025264  025240          MOV     #ZPAT10,@#ZTMP1
5727  025136  012737  000210  025242          MOV     #210,@#ZTMP2
5728  025144  000730                          BR      Z1
5729  025146  000456                  Z6:     BR      ZDONE
5730  025150                          ZERR1:                          ;DATA FAILURE
5731  025150  012737  025050  001236          MOV     #Z2,@#$TMP2
5732  025156  013737  025240  001240          MOV     @#ZTMP1,@#$TMP3
5733  025164  012737  025274  001242          MOV     #ZPAT20,@#$TMP4
5734  025172  012737  025244  001244          MOV     #ZDAT00,@#$TMP5
5735  025200  013737  025240  001246          MOV     @#ZTMP1,@#$TMP6
5736  025206  104152                  1$:     ERROR   152
5737  025210  000435                          BR      ZDONE
5738  025212                          ZERR2:
5739  025212  012737  025050  001236          MOV     #Z2,@#$TMP2
5740  025220  010537  001240                  MOV     R5,@#$TMP3
5741  025224  013737  025242  001242          MOV     @#ZTMP2,@#$TMP4
5742  025232  104153                  1$:     ERROR   153
5743  025234  000423                          BR      ZDONE
5744
5745  025236  000000                  ZFLAG:  .WORD   0
5746  025240  000000                  ZTMP1:          0
5747  025242  000000                  ZTMP2:          0
5748
5749  025244  000000                  ZDAT00: .WORD   0
5750  025246  000000                  ZDAT01:         0
5751  025250  000000                  ZDAT02:         0
5752  025252  000000                  ZDAT03:         0
5753
5754  025254  031463                  ZPAT00:         031463
5755  025256  010421                  ZPAT01:         010421
5756  025260  146314                  ZPAT02:         146314
5757  025262  156735                  ZPAT03:         156735
5758
5759  025264  156735                  ZPAT10:         156735
5760  025266  167356                  ZPAT11:         167356
5761  025270  135673                  ZPAT12:         135673
5762  025272  146314                  ZPAT13:         146314
5763
5764  025274  000000                  ZPAT20:         0
5765  025276  000000                  ZPAT21:         0
5766  025300  000000                  ZPAT22:         0
5767  025302  000000                  ZPAT23:         0
5768
5769  025304                          ZDONE:
5770  025304  104412                          RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
5771                                                                  ;SEE IF THE USER HAS EXPRESSED
5772                                                                  ;THE DESIRE TO CHANGE THE SOFTWARE
5773                                                                  ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
5774                                                                  ;THE USER TYPED CONTROL G?).
5775
5776
5777                                  ;;*****************************************************************
```

```
                                        ;*TEST 31:      ADDF AND ADDD WITH E(AC)=E(FSRC) TEST AND  BUT FT  TEST
                                        ;*
                                        ;* THIS IS A TEST OF THE ADD INSTRUCTION WITH THE
                                        ;* OPERANDS HAVING EQUAL EXPONENTS.  THE (BUF FT)
                                        ;* FORK IN THE ROUND TRUNK FLOWS IS ALSO TESTED.
                                        ;*
                                        ;;*****************************************************************
025326  000004                          ST31:   SCOPE
                                        AA1:
                                                .PERR                           ;SET UP THE LOOP ON ERROR ADDRESS.
                                                MOV     #3240,R0
003240                                          LDFPS   R0                      ;SET FIV FIV FD AND FT
025330  013737  025670  000244                  MOV     #AAERR0,@#FPVECT        ;IN CASE THE OVER/UNDER
025336  012705  026346                          MOV     #AAPAT0,R0              ;FLOWS IN TRAP WILL
                                                                                ;OCCUR
025342  172410                                  LDD     (R0),AC0                ;SET UP AC0
025344  012737  025346  001236                  MOV     #AA2,@#$TMP2            ;OPERAND
025352  012700  026256                          MOV     #AAPAT1,R0
025356  172010                          AA2:    ADDD    (R0),AC0                ;TEST INSTRUCTION
                                                                                ;SHOULD "RUNCATE
025360  012700  026236                  AA3:    MOV     #AADAT0,R0
025364  174010                                  STD     AC0,(R0)                ;GET THE RESULT
025366  012701  026266                          MOV     #AAPAT2,R1
025372  012702  000004                          MOV     #4,R2
025376  022021                          AA4:    CMP     (R0)+,(R1)+             ;CORRECT?
025400  001414                                  BEQ     AA7
025402  012700  026276                          MOV     #AAPAT3,R0              ;DID (BUT FT) FAIL
025406  012701  026236                          MOV     #AADAT0,R1
025412  012702  000004                          MOV     #4,R2
025416  022021                          AA5:    CMP     (R0)+,(R1)+
025420  001401                                  BEQ     AA6
025422  000561                                  BR      AAERR1  ;DATA ERROR
025424  077204                          AA6:    SOB     R2,AA5
025426  000137  026012                          JMP     @#AAERR2                ;(BUT FT) ERROR
025432  077217                          AA7:    SOB     R2,AA4

                                        ;NOW TEST DOUBLE FLOATING ROUND MODE.

025434                                  AA10:
025434  104413                                  .PERR                           ;SET UP THE LOOP ON ERROR ADDRESS.
025436  012700  003200                          MOV     #3200,R0                ;SET FD FIV FIV. FT=0
025442  170100                                  LDFPS   R0
025444  012700  026246                          MOV     #AAPAT0,R0
025450  172410                                  LDD     (R0),AC0                ;SET UP AC0 OPERAND
025452  012737  025454  001236                  MOV     #AA11,@#$TMP2
025460  012700  026256                          MOV     #AAPAT1,R0
025464  172010                          AA11:   ADDD    (R0),AC0                ;TEST INSTRUCTION
                                                                                ;SHOULD ROUND
025466  012700  026236                  AA12:   MOV     #AADAT0,R0
025472  174010                                  STD     AC0,(R0)                ;GET THE RESULT
025474  012701  026276                          MOV     #AAPAT3,R1
025500  012702  000004                          MOV     #4,R2
025504  022021                          AA13:   CMP     (R0)+,(R1)+             ;CORRECT?
025506  001426                                  BEQ     AA20
025510  012700  026266                          MOV     #AAPAT2,R0              ;DID (BUT FT) FAIL?
025514  012701  026236                          MOV     #AADAT0,R1
```

```
025510  012702  000004                     MOV     #4,R2
025514  022021                     AA14:   CMP     (R0)+,(R1)+
025516  001413                             BEQ     AA17
025520  012700  026306                     MOV     #AAPAT4,R0      ;WAS THE FLOATING
025524  012701  026236                     MOV     #AADAT0,R1      ;CONSTANT USED
025530  012702  000004                     MOV     #4,R2           ;INSTEAD OF THE
025534  022021                     AA15:   CMP     (R0)+,(R1)+     ;DOUBLE CONSTANT
025536  001401                             BEQ     AA16            ;IN THE ROUND
025540  000542                             BR      AAERR3  ;FLOWS?
025542  077204                     AA16:   SOB     R2,AA15         ;DATA ERROR
025544  000544                             BR      AAERR4  ;CONSTANT ERROR
025546  077216                     AA17:   SOB     R2,AA14
025550  000560                             BR      AAERR5  ;(BUT FT) ERROR
025552  077230                     AA20:   SOB     R2,AA13

                                   ;NOW TEST ADDF WITH FT=0, ROUND MODE

025554                             AA21:
025554  104413                             LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
025556  012700  003200                     MOV     #3200,R0                ;FIV=1, FIV=1, FT=0
025562  170100                             LDFPS   R0
025564  012700  026246                     MOV     #AAPAT0,R0      ;LOAD AC0 OPERAND
025570  172410                             LDD     (R0),AC0
025572  170001                             SETF                    ;ENTER FLOATING MODE
025574  012737  025606  001236             MOV     #AA22,@#$TMP2
025602  012700  026316                     MOV     #AAPAT5,R0
025606  172010                     AA22:   ADDF    (R0),AC0        ;TEST INSTRUCTION
                                                                   ;SHOULD ROUND
025610                             AA23:
025610  170011                             SETD                    ;RESET TO DOUBLE
                                                                   ;MODE
025612  012700  026236                     MOV     #AADAT0,R0      ;GET THE RESULT
025616  174010                             STD     AC0,(R0)
025620  012701  026326                     MOV     #AAPAT6,R1      ;CORRECT?
025624  012702  000002                     MOV     #2,R2
025630  022021                     AA24:   CMP     (R0)+,(R1)+
025632  001413                             BEQ     AA27
025634  012700  026266                     MOV     #AAPAT2,R0      ;WAS THE DOUBLE
025640  012701  026236                     MOV     #AADAT0,R1      ;CONSTANT USED INSTEAD
025644  012702  000002                     MOV     #2,R2           ;OF THE FLOATING
025650  022011                     AA25:   CMP     (R0)+,(R1)      ;CONSTANT IN THE
025652  001401                             BEQ     AA26            ;ROUND FLOWS?
025654  000534                             BR      AAERR6  ;DATA ERROR
025656  077204                     AA26:   SOB     R2,AA25
025660  000550                             BR      AAERR7  ;CONSTANT ERROR
025662  077216                     AA27:   SOB     R2,AA24
025664  000137  026336                     JMP     @#AADONE

                                   ;COME HERE IF A TRAP OCCURS TO 244.

025670  013700  001236             AAERR0: MOV     @#$TMP2,R0      ;SEE IF THE TRAP WAS
025674  005720                             TST     (R0)+   ;AT A TEST INSTRUCTION
025676  020016                             CMP     R0,(SP)
025700  001402                             BEQ     1$
025702  000137  040200             10$:    JMP     @#FPSPUR
025706                             1$:
```

```
 025706  170300                              STST    R0                     ;GET FEC
 025710  020027  000010                      CMP     R0,#10
 025714  001405                              BEQ     20$                    ;OVERFLOW
 025716  020027  000012                      CMP     R0,#12
 025722  001410                              BEQ     30$                    ;UNDERFLOW
 025724  000766                              BR      10$
 025726  025730                      20$
 025730  011637  001236              20$:    MOV     (SP),@#STMP2           ;REPORT OVERFLOW ERROR
 025734  022626                              CMP     (SP)+,(SP)+
 025736  104154                      21$:    ERROR   154
 025740  000137  026336              25$:    JMP     @#AADONE
 025744  011637  001236              30$:    MOV     (SP),@#STMP2           ;REPORT UNDERFLOW
 025750  022626                              CMP     (SP)+,(SP)+            ;ERROR
 025752  104155                      31$:    ERROR   155
 025754  000771                              BR      25$

                                     ;ADDD RESULT INCORRECT
 025756  012737  026266  001246      AAERR1: MOV     #AAPAT2,@#STMP6
 025764  012737  026246  001242      AAER10: MOV     #AAPAT0,@#STMP4
 025772  012737  026256  001240              MOV     #AAPAT1,@#STMP3
 026000  012737  026236  001244              MOV     #AADAT0,@#STMP5
 026006  104162                      1$:     ERROR   162
 026010  000552                              BR      AADONE
 026012  012737  026266  001246      AAERR2: MOV     #AAPAT2,@#STMP6 ;(BUT FT) FAILED.
 026020  012737  026246  001242              MOV     #AAPAT0,@#STMP4
 026026  012737  026256  001240              MOV     #AAPAT1,@#STMP3
 026034  012737  026236  001244              MOV     #AADAT0,@#STMP5
 026042  104156                      1$:     ERROR   156
 026044  000534                              BR      AADONE
 026046  012737  026276  001246      AAERR3: MOV     #AAPAT3,@#STMP6 ;DATA ERROR.
 026054  000743                              BR      AAER10
 026056  012737  026276  001246      AAERR4: MOV     #AAPAT3,@#STMP6 ;BAD CONSTANT
 026064  012737  026246  001242              MOV     #AAPAT0,@#STMP4
 026072  012737  026256  001240              MOV     #AAPAT1,@#STMP3
 026100  012737  026236  001244              MOV     #AADAT0,@#STMP5
 026106  104160                      1$:     ERROR   160
 026110  000512                              BR      AADONE
 026112  012737  026276  001246      AAERR5: MOV     #AAPAT3,@#STMP6 ;(BUT FT) FAILED.
 026120  012737  026246  001242              MOV     #AAPAT0,@#STMP4
 026126  012737  026256  001240              MOV     #AAPAT1,@#STMP3
 026134  012737  026236  001244              MOV     #AADAT0,@#STMP5
 026142  104157                      1$:     ERROR   157
 026144  000474                              BR      AADONE
 026146  012737  026316  001240      AAERR6: MOV     #AAPAT5,@#STMP3 ;FD=0 AND
 026154  012737  026246  001242              MOV     #AAPAT0,@#STMP4 ;DATA ERROR
 026162  012737  026236  001244              MOV     #AADAT0,@#STMP5
 026170  012737  026326  001246              MOV     #AAPAT6,@#STMP6
 026176  104160                      1$:     ERROR   160
 026200  000456                              BR      AADONE
 026202  012737  026316  001240      AAERR7: MOV     #AAPAT5,@#STMP3 ;CONSTANT ERROR
 026210  012737  026246  001242              MOV     #AAPAT0,@#STMP4
 026216  012737  026236  001244              MOV     #AADAT0,@#STMP5
 026224  012737  026326  001246              MOV     #AAPAT6,@#STMP6
 026232  104161                      1$:     ERROR   161
 026234  000440                              BR      AADONE
 026236  000000                      AADAT0:         0
```

```
026240  000000                        0
026242  000000                        0
026244  000000                        0
026246  000200         AAPAT0:        200
026250  000000                        0
026252  000000                        0
026254  000000                        0
026256  000200         AAPAT1:        200
026260  000000                        0
026262  000000                        0
026264  000001                        1
026266  000400         AAPAT2:        400
026270  000000                        0
026272  000000                        0
026274  000000                        0
026276  000400         AAPAT3:        400
026300  000000                        0
026302  000000                        0
026304  000001                        1
026306  000400         AAPAT4:        400
026310  000000                        0
026312  100000                        100000
026314  000000                        0
026316  000200         AAPAT5:        200
026320  000001                        1
026322  000000                        0
026324  000000                        0
026326  000400         AAPAT6:        400
026330  000001                        1
026332  000000                        0
026334  000000                        0
026336                 AADONE:
026336  104412                 RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
                                                       ;SEE IF THE USER HAS EXPRESSED
                                                       ;THE DESIRE TO CHANGE THE SOFTWARE
                                                       ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
                                                       ;THE USER TYPED CONTROL G?).

                       ;;*************************************************************
                       ;*TEST 32        ADDF AND ADDD WITH E(AC) LESS THAN E(FSRC) TEST
                       ;*
                       ;*THIS IS ATEST 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)
                       ;*
                       ;;*************************************************************
026340  000004         TST32:  SCOPE
                       ;EXPONENT DIFFERENCE=57=71 (OCT) FD=1
026342                 CC1:
026342  104413                 LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
026344  012704  003200         MOV     #3200,R4        ;SET FIV,FIV, AND FD
026350  170104                 LDFPS   R4
026352  012737  026372  001236 MOV     #CC2,@#$TMP2
026360  012700  027774         MOV     #CCP0,R0        ;SET AC0 OPERAND
```

```
026364  172410                           LDD     (R0),AC0                :AC0
026366  012700  030014                   MOV     #CCP2,R0
026372  172010                   CC2:    ADDD    (R0),AC0                :TEST INSTRUCTION
026374  170205                           STFPS   R5                      :GET FPS
026376  012700  027764                   MOV     #CCDATO,R0              :GET THE RESULT
026402  174010                           STD     AC0,(R0)
026404  012701  030014                   MOV     #CCP2,R1                :IS IT CORRECT
026410  012702  000004                   MOV     #4,R2
026414  022021                   CC3:    CMP     (R0)+,(R1)+
026416  001415                           BEQ     CC6
026420  012700  027764                   MOV     #CCDATO,R0              :DID A BAD
026424  012701  027774                   MOV     #CCP0,R1                :CONSTANT (NOT 57)
026430  012702  000004                   MOV     #4,R2                   :GET GENERATED
026434  022021                   CC4:    CMP     (R0)+,(R1)+             :FOR THE ALIGNMENT
026436  001402                           BEQ     CC5                     :FLOWS?
026440  000137  027362                   JMP     @#CCER1                 :DATA ERROR.D
026444  077205                   CC5:    SOB     R2,CC4
026446  000137  027420                   JMP     @#CCER2                 :BAD CONSTANT.D
026452  077220                   CC6:    SOB     R2,CC3
026454  020405                           CMP     R4,R5                   :FPS CORRECT?
026456  001402                           BEQ     CC7
026460  000137  027326                   JMP     @#CCER0                 :BAD FPS.
                                 ;EXPONENT DIFFERENCE=56=70 (OCT) FD=1
026464                           CC7:
026464  104413                           LPERR                           :SET UP THE LOOP ON ERROR ADDRESS.
026466  012704  003200                   MOV     #3200,R4                :SET FIV,FIV, AND FD
026472  170104                           LDFPS   R4
026474  012737  026514  001236           MOV     #CC8,@#$TMP2
026502  012700  027774                   MOV     #CCP0,R0                :SET AC0 OPERAND
026506  172410                           LDD     (R0),AC0
026510  012700  030034                   MOV     #CCP1,R0                :FSRC
026514  172010                   CC8:    ADDD    (R0),AC0                :TEST INSTRUCTION
026516  170205                           STFPS   R5                      :GET FPS
026520  012700  027764                   MOV     #CCDATO,R0              :GET THE RESULT
026524  174010                           STD     AC0,(R0)
026526  012701  030064                   MOV     #CCP7,R1                :IS IT CORRECT
026532  012702  000004                   MOV     #4,R2
026536  022021                   CC9:    CMP     (R0)+,(R1)+
026540  001415                           BEQ     CC12
026542  012700  027764                   MOV     #CCDATO,R0              :DID A BAD
026546  012701  030004                   MOV     #CCP1,R1                :CONSTANT (NOT 57)
026552  012702  000004                   MOV     #4,R2                   :GET GENERATED
026556  022021                   CC10:   CMP     (R0)+,(R1)+             :FOR THE ALIGNMENT
026560  001402                           BEQ     CC11                    :FLOWS?
026562  000137  027456                   JMP     @#CCER3                 :DATA ERROR.D
026566  077205                   CC11:   SOB     R2,CC10
026570  000137  027474                   JMP     @#CCER4                 :BAD CONSTANT.D
026574  077220                   CC12:   SOB     R2,CC9
026576  020405                           CMP     R4,R5                   :FPS CORRECT?
026600  001402                           BEQ     CC13
026602  000137  027326                   JMP     @#CCER0                 :BAD FPS.
                                 ;EXPONENT DIFFERENCE=25=31 (OCT) FD=0
026606                           CC13:
026606  104413                           LPERR                           :SET UP THE LOOP ON ERROR ADDRESS.
026610  012737  026636  001236           MOV     #CC14,@#$TMP2
026616  012700  027774                   MOV     #CCP0,R0                :SET UP AC0 OPERAND.
```

```
026622  172410                          LDD     (R0),AC0
026624  012704  003000                  MOV     #3000,R4        ;SET FIV,FIV. CLEAR FD.
026630  170104                          LDFPS   R4
026632  012700  030054                  MOV     #CCP6,R0        ;FSRC
026636  172010                  CC14:   ADDF    (R0),AC0        ;TEST INSTRUCTION
026640  170205                          STFPS   R5
026642  170011                          SETD                    ;REENTER DOUBLE MOVE
026644  012700  027764                  MOV     #CCDATO,R0      ;GET THE RESULT
026650  174010                          STD     AC0,(R0)
026652  012701  030054                  MOV     #CCP6,R1        ;IS THE RESULT CORRECT?
026656  012702  000002                  MOV     #2,R2
026662  022021                  CC15:   CMP     (R0)+,(R1)+
026664  001415                          BEQ     CC18
026666  012700  027764                  MOV     #CCDATO,R0      ;WAS A BAD CONSTANT
026672  012701  030024                  MOV     #CCP3,R1        ;USED (NOT 25) IN
026676  012702  000002                  MOV     #2,R2           ;THE ALIGN FLOWS?
026702  022021                  CC16:   CMP     (R0)+,(R1)+
026704  001402                          BEQ     CC17
026706  000137  027532                  JMP     @#CCER5         ;DATA ERROR F
026712  077205                  CC17:   SOB     R2,CC16
026714  000137  027566                  JMP     @#CCER6         ;BAD CONSTANT F
026720  077220                  CC18:   SOB     R2,CC15
026722  020405                          CMP     R4,R5
026724  001402                          BEQ     CC19
026726  000137  027344                  JMP     @#CCER90        ;BAD FPS.
                                ;EXPONENT DIFFERENCE=24=30 (OCT) FD=0
026732                          CC19:
026732  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
026734  012737  026762  001236          MOV     #CC20,@#$TMP2
026742  012700  030024                  MOV     #CCP3,R0        ;SET UP AC0 OPERAND.
026746  172410                          LDD     (R0),AC0
026750  012704  003000                  MOV     #3000,R4        ;SET FIV,FIV. CLEAR FD.
026754  170104                          LDFPS   R4
026756  012700  030044                  MOV     #CCP5,R0        ;FSRC
026762  172010                  CC20:   ADDF    (R0),AC0        ;TEST INSTRUCTION
026764  170205                          STFPS   R5
026766  170011                          SETD                    ;REENTER DOUBLE MOVE
026770  012700  027764                  MOV     #CCDATO,R0      ;GET THE RESULT
026774  174010                          STD     AC0,(R0)
026776  012701  030074                  MOV     #CCP10,R1       ;IS THE RESLT CORRECT?
027002  012702  000002                  MOV     #2,R2
027006  022021                  CC21:   CMP     (R0)+,(R1)+
027010  001415                          BEQ     CC24
027012  012700  027764                  MOV     #CCDATO,R0      ;WAS A BAD CONSTANT
027016  012701  030044                  MOV     #CCP5,R1        ;USED (NOT 25) IN
027022  012702  000002                  MOV     #2,R2           ;THE ALIGN FLOWS?
027026  022021                  CC22:   CMP     (R0)+,(R1)+
027030  001402                          BEQ     CC23
027032  000137  027622                  JMP     @#CCER7         ;DATA ERROR F
027036  077205                  CC23:   SOB     R2,CC22
027040  000137  027640                  JMP     @#CCER8 ;BAD CONSTANT F
027044  077220                  CC24:   SOB     R2,CC21
027046  020405                          CMP     R4,R5
027050  001402                          BEQ     CC25
027052  000137  027344                  JMP     @#CCER90        ;BAD FPS.
                                ;EXPONENT DIFFERENCE=1 FD=1
```

H09


```
027056                            CC25:
027056  104413                            LPERR                  ;SET UP THE LOOP ON ERROR ADDRESS.
027060  012704  003200                    MOV     #3200,R4       ;SET FIV,FIV, AND FD
027064  170104                            LDFPS   R4
027066  012737  027106  001236            MOV     #CC26,@#$TMP2
027074  012700  027774                    MOV     #CCP0,R0       ;SET AC0 OPERAND
027100  172410                            LDD     (R0),AC0
027102  012700  030024                    MOV     #CCP3,R0       ;FSRC
027106  172010                    CC26:   ADDD    (R0),AC0       ;TEST INSTRUCTION
027110  170205                            STFPS   R5             ;GET FPS
027112  012700  027764                    MOV     #CCDAT0,R0     ;GET THE RESULT
027116  174010                            STD     AC0,(R0)
027120  012701  030104                    MOV     #CCP11,R1      ;IS IT CORRECT
027124  012702  000004                    MOV     #4,R2
027130  022021                    CC27:   CMP     (R0)+,(R1)+
027132  001415                            BEQ     CC30
027134  012700  027764                    MOV     #CCDAT0,R0     ;DID A BAD
027140  012701  030024                    MOV     #CCP3,R1       ;CONSTANT (NOT 57)
027144  012702  000004                    MOV     #4,R2          ;GET GENERATED
027150  022021                    CC28:   CMP     (R0)+,(R1)+    ;FOR THE ALIGNMENT
027152  001402                            BEQ     CC29           ;FLOWS?
027154  000137  027674                    JMP     @#CCER10       ;DATA ERROR.D
027160  077205                    CC29:   SOB     R2,CC28
027162  000137  027712                    JMP     @#CCER11       ;BAD CONSTANT.D
027166  077220                    CC30:   SOB     R2,CC27
027170  020405                            CMP     R4,R5          ;FPS CORRECT?
027172  001402                            BEQ     CC31
027174  000137  027326                    JMP     @#CCER0        ;BAD FPS.
                                  ;EXPONENT DIFFERENCE=100=144 (OCT) FD=1
027200                            CC31:
027200  104413                            LPERR                  ;SET UP THE LOOP ON ERROR ADDRESS.
027202  012704  003200                    MOV     #3200,R4       ;SET FIV,FIV, AND FD
027206  170104                            LDFPS   R4
027210  012737  027230  001236            MOV     #CC32,@#$TMP2
027216  012700  027774                    MOV     #CCP0,R0       ;SET AC0 OPERAND
027222  172410                            LDD     (R0),AC0
027224  012700  030034                    MOV     #CCP4,R0       ;FSRC
027230  172010                    CC32:   ADDD    (R0),AC0       ;TEST INSTRUCTION
027232  170205                            STFPS   R5             ;GET FPS
027234  012700  027764                    MOV     #CCDAT0,R0     ;GET THE RESULT
027240  174010                            STD     AC0,(R0)
027242  012701  030034                    MOV     #CCP4,R1       ;IS IT CORRECT
027246  012702  000004                    MOV     #4,R2
027252  022021                    CC33:   CMP     (R0)+,(R1)+
027254  001415                            BEQ     CC36
027256  012700  027764                    MOV     #CCDAT0,R0     ;DID A BAD
027262  012701  030034                    MOV     #CCP4,R1       ;CONSTANT (NOT 57)
027266  012702  000004                    MOV     #4,R2          ;GET GENERATED
027272  022021                    CC34:   CMP     (R0)+,(R1)+    ;FOR THE ALIGNMENT
027274  001402                            BEQ     CC35           ;FLOWS?
027276  000137  027730                    JMP     @#CCER12       ;DATA ERROR.D
027302  077205                    CC35:   SOB     R2,CC34
027304  000137  027746                    JMP     @#CCER13       ;BAD CONSTANT.D
027310  077220                    CC36:   SOB     R2,CC33
027312  020405                            CMP     R4,R5          ;FPS CORRECT?
027314  001402                            BEQ     CC37
```

```
027316  000137  027326                    JMP     @#CCERO          ;BAD FPS.
027322  000137  030124            CC37:   JMP     @#CCDONE
027326  010437  001242            CCERO:  MOV     R4,@#$TMP4       ;FPS ERROR D
027332  010537  001240                    MOV     R5,@#$TMP3
027336  104164                    1$:     ERROR   164
027340  000137  030124                    JMP     @#CCDONE
027344  010437  001242            CCER90: MOV     R4,@#$TMP4       ;FPS ERROR F
027350  010537  001240                    MOV     R5,@#$TMP3
027354  104165                    1$:     ERROR   165
027356  000137  030124                    JMP     @#CCDONE
027362  012737  030014  001240    CCER1:  MOV     #CCP2,@#$TMP3    ;DATA ERROR D
027370  012737  030014  001246            MOV     #CCP2,@#$TMP6
027376  012737  027774  001242    CCER50: MOV     #CCP0,@#$TMP4
027404  012737  027764  001244            MOV     #CCDAT0,@#$TMP5
027412  104166                    1$:     ERROR   166
027414  000137  030124                    JMP     @#CCDONE
027420  012737  030014  001240    CCER2:  MOV     #CCP2,@#$TMP3    ;CONSTANT BAD D(B)
027426  012737  030014  001246            MOV     #CCP2,@#$TMP6
027434  012737  027774  001242    CCER22: MOV     #CCP0,@#$TMP4
027442  012737  027764  001244            MOV     #CCDAT0,@#$TMP5
027450  104172                    1$:     ERROR   172
027452  000137  030124                    JMP     @#CCDONE
027456  012737  030004  001240    CCER3:  MOV     #CCP1,@#$TMP3
027464  012737  030064  001246            MOV     #CCP7,@#$TMP6
027472  000741                            BR      CCER50
027474  012737  030004  001240    CCER4:  MOV     #CCP1,@#$TMP3    ;CONSTANT BAD D(G)
027502  012737  030064  001246            MOV     #CCP7,@#$TMP6
027510  012737  027774  001242    CCER44: MOV     #CCP0,@#$TMP4
027516  012737  027764  001244            MOV     #CCDAT0,@#$TMP5
027524  104173                    1$:     ERROR   173
027526  000137  030124                    JMP     @#CCDONE
027532  012737  030054  001240    CCER5:  MOV     #CCP6,@#$TMP3    ;DATA ERROR F
027540  012737  030054  001246            MOV     #CCP6,@#$TMP6
027546  012737  027774  001242    CCER55: MOV     #CCP0,@#$TMP4
027554  012737  027764  001244            MOV     #CCDAT0,@#$TMP5
027562  104170                    1$:     ERROR   170
027564  000557                            BR      CCDONE
027566  012737  030054  001240    CCER6:  MOV     #CCP6,@#$TMP3    ;CONSTANT BAD F(B)
027574  012737  030054  001246            MOV     #CCP6,@#$TMP6
027602  012737  027774  001242            MOV     #CCP0,@#$TMP4
027610  012737  027764  001244            MOV     #CCDAT0,@#$TMP5
027616  104174                    1$:     ERROR   174
027620  000541                            BR      CCDONE
027622  012737  030044  001240    CCER7:  MOV     #CCP5,@#$TMP3    ;DATA ERROR F
027630  012737  030074  001246            MOV     #CCP10,@#$TMP6
027636  000743                            BR      CCER55
027640  012737  030044  001240    CCER8:  MOV     #CCP5,@#$TMP3    ;CONSTANT BAD F(G)
027646  012737  030074  001246            MOV     #CCP10,@#$TMP6
027654  012737  027764  001244            MOV     #CCDAT0,@#$TMP5
027662  012737  027774  001242            MOV     #CCP0,@#$TMP4
027670  104175                    1$:     ERROR   175
027672  000514                            BR      CCDONE
027674  012737  030024  001240    CCER10: MOV     #CCP3,@#$TMP3    ;DATA ERROR D
027702  012737  030104  001246            MOV     #CCP11,@#$TMP6
027710  000632                            BR      CCER50
027712  012737  030024  001240    CCER11: MOV     #CCP3,@#$TMP3    ;CONSTANT BAD D(G)
```

```
027720  012737  030104  001246              MOV     #CCP11,@#$TMP6
027726  000670                              BR      CCER44
027730  012737  030034  001240   CCER12:    MOV     #CCP4,@#$TMP3      ;DATA ERROR D
027736  012737  030034  001246              MOV     #CCP4,@#$TMP6
027744  000614                              BR      CCER50
027746  012737  030034  001240   CCER13:    MOV     #CCP4,@#$TMP3      ;CONSTANT BAD D(B)
027754  012737  030034  001246              MOV     #CCP4,@#$TMP6
027762  000624                              BR      CCER22
027764  000000                   CCDAT0:    0
027766  000000                              0
027770  000000                              0
027772  000000                              0
027774  000200                   CCP0:      200                        ;E(AC)=1
027776  000000                              0
030000  000000                              0
030002  000000                              0
030004  016200                   CCP1:      16200                      ;E(FSRC)=E(AC)+56=57
030006  000000                              0                          ;                 =71(OCT)
030010  000000                              0
030012  000000                              0
030014  016400                   CCP2:      16400                      ;E(FSRC)=E(AC)+57=58
030016  000000                              0                          ;                 =72(OCT)
030020  000000                              0
030022  000000                              0
030024  000400                   CCP3:      400                        ;E(FSRC)=E(AC)+1=2
030026  000000                              0
030030  000000                              0
030032  000000                              0
030034  031200                   CCP4:      31200                      ;E(FSRC)=E(AC)+100=101=145(OCT)
030036  000000                              0
030040  000000                              0
030042  000000                              0
030044  006200                   CCP5:      6200                       ;E(FSRC)=E(AC)+24=25=31(OCT)
030046  000000                              0
030050  000000                              0
030052  000000                              0
030054  006400                   CCP6:      6400                       ;E(FSRC)=E(AC)+25=26=32(OCT)
030056  000000                              0
030060  000000                              0
030062  000000                              0
030064  016200                   CCP7:      16200                      ;CCP1 RES
030066  000000                              0
030070  000000                              0
030072  000001                              1
030074  006200                   CCP10:     6200                       ;CCP5 RES
030076  000001                              1
030100  000000                              0
030102  000000                              0
030104  000500                   CCP11:     500                        ;CCP3 RES
030106  000000                              0
030110  000000                              0
030112  000000                              0
030114  000200                   CCP12:     200                        ;BAD CONSTANT
030116  000000                              0                          ;RES CCP2,CCP4
030120  000000                              0
030122  000000                              0
```

```
                                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?).

                                ;;*****************************************************************
                                ;*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).
                                ;*
                                ;;*****************************************************************
                                ST33:  SCOPE
                                ;EXPONENT DIFFERENCE=57=71 (OCT) FD=1
                                BB1:
030130  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
030132  012704  003200                  MOV     #3200,R4        ;SET FIV FIV, AND FD
030136  170104                          LDFPS   R4
030140  012737  030766  000244          MOV     #BBERR,@#FPVECT ;SET UP FOR ERROR
030146  012737  030166  001236          MOV     #BB2,@#$TMP2    ;IN CASE THE OVER\
                                                                ;UNDER FLOWS FAIL.
030154  012700  031330                  MOV     #BBPAT2,R0      ;SET AC0 OPERAND.
030160  172410                          LDD     (R0),AC0
030162  012700  031320                  MOV     #BBPAT1,R0      ;FSRC
030166  172010                  BB2:    ADDD    (R0),AC0        ;TEST INSTRUCTION
030170  170205                          STFPS   R5
030172  012700  031300          BB3:    MOV     #BBDAT0,R0      ;GET THE RESULT
030176  174010                          STD     AC0,(R0)
030200  012701  031330                  MOV     #BBPAT2,R1      ;RESULT CORRECT?
030204  012702  000004                  MOV     #4,R2
030210  022021                  BB4:    CMP     (R0)+,(R1)+
030212  001402                          BEQ     BB5
030214  000137  031026                  JMP     @#BBER1         ;DATA ERROR D
030220  077205                  BB5:    SOB     R2,BB4
                                                                ;WAS FPS CORRECT?
030222  020405                          CMP     R4,R5
030224  001402                          BEQ     BB6
030226  000137  030766                  JMP     @#BBERR         ;FPS ERROR
                                ;EXPONENT DIFFERENCE=56=70 (OCT) FD=1
030232                          BB6:
030232  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
030234  012704  003200                  MOV     #3200,R4        ;SET FIV,FIV, AND FD
030240  170104                          LDFPS   R4
030242  012737  030262  001236          MOV     #BB7,@#$TMP2
030250  012700  031350                  MOV     #BBPAT4,R0      ;SET AC0 OPERAND
030254  172410                          LDD     (R0),AC0
030256  012700  031320                  MOV     #BBPAT1,R0      ;FSRC
030262  172010                  BB7:    ADDD    (R0),AC0        ;TEST INSTRUCTION
030264  170205                          STFPS   R5              ;GET FPS
```

```
6338  030266  012700  031300          MOV     #BBDAT0,R0      ;GET THE RESULT
6339  030272  174010                  STD     AC0,(R0)
6340  030274  012701  031410          MOV     #BBP10,R1       ;IS IT CORRECT
6341  030300  012702  000004          MOV     #4,R2
6342  030304  022021          BB10:   CMP     (R0)+,(R1)+
6343  030306  001415                  BEQ     BB13
6344  030310  012700  031300          MOV     #BBDAT0,R0      ;DID A BAD
6345  030314  012701  031350          MOV     #BBPAT4,R1      ;CONSTANT (NOT 57)
6346  030320  012702  000004          MOV     #4,R2           ;GET GENERATED
6347  030324  022021          BB11:   CMP     (R0)+,(R1)+     ;FOR THE ALIGNMENT
6348  030326  001402                  BEQ     BB12            ;FLOWS?
6349  030330  000137  031064          JMP     @#BBER2         ;DATA ERROR.D
6350  030334  077205          BB12:   SOB     R2,BB11
6351  030336  000137  031102          JMP     @#BBER3         ;BAD CONSTANT.D
6352  030342  077220          BB13:   SOB     R2,BB10
6353  030344  020405                  CMP     R4,R5           ;FPS CORRECT?
6354  030346  001402                  BEQ     BB14
6355  030350  000137  030766          JMP     @#BBER0         ;BAD FPS.
6356                          ;EXPONENT DIFFERENCE=25=31 (OCT) FD=0
6357  030354                  BB14:
6358  030354  104413                  LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
6359  030356  012737  030404  001236  MOV     #BB15,@#$TMP2
6360  030364  012700  031310          MOV     #BBPAT0,R0      ;SET UP AC0 OPERAND
6361  030370  172410                  LDD     (R0),AC0
6362  030372  012704  003000          MOV     #3000,R4        ;SET FIV AND FIV
6363                                                          ;CLEAR FD
6364  030376  170104                  LDFPS   R4
6365  030400  012700  031320          MOV     #BBPAT1,R0      ;FSRC
6366  030404  172010          BB15:   ADDF    (R0),AC0        ;TEST INSTRUCTION
6367  030406  170205                  STFPS   R5
6368  030410  170011                  SETD                    ;REENTED DOUBLE MODE.
6369  030412  012700  031300          MOV     #BBDAT0,R0      ;GET THE RESULT
6370  030416  174010                  STD     AC0,(R0)
6371  030420  012701  031310          MOV     #BBPAT0,R1      ;IS THE RESULT
6372  030424  012702  000002          MOV     #2,R2           ;CORRECT?
6373  030430  022021          BB16:   CMP     (R0)+,(R1)+
6374  030432  001402                  BEQ     BB17
6375  030434  000137  031136          JMP     @#BBER4         ;DATA ERROR F
6376  030440  077205          BB17:   SOB     R2,BB16
6377  030442  020405                  CMP     R4,R5           ;IS FPS CORRECT?
6378  030444  001402                  BEQ     BB20
6379  030446  000137  031006          JMP     @#BBER10        ;FPS ERROR.
6380                          ;EXPONENT DIFFERENCE=24=30 (OCT)
6381  030452                  BB20:
6382  030452  104413                  LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
6383  030454  012737  030502  001236  MOV     #BB21,@#$TMP2
6384  030462  012700  031340          MOV     #BBPAT3,R0      ;SET UP AC0 OPERAND.
6385  030466  172410                  LDD     (R0),AC0
6386  030470  012704  003000          MOV     #3000,R4        ;SET FIU,FIV. CLEAR FD.
6387  030474  170104                  LDFPS   R4
6388  030476  012700  031320          MOV     #BBPAT1,R0      ;FSRC
6389  030502  172010          BB21:   ADDF    (R0),AC0        ;TEST INSTRUCTION
6390  030504  170205                  STFPS   R5
6391  030506  170011                  SETD                    ;REENTER DOUBLE MODE
6392  030510  012700  031300          MOV     #BBDAT0,R0      ;GET THE RESULT
6393  030514  174010                  STD     AC0,(R0)
```

```
6394  030516  012701  031400                  MOV     #BBP7,R1        ;IS THE RESULT CORRECT?
6395  030522  012702  000002                  MOV     #2,R2
6396  030526  022021                  BB22:   CMP     (R0)+,(R1)+
6397  030530  001415                          BEQ     BB25
6398  030532  012700  031300                  MOV     #BBDAT0,R0      ;WAS A BAD CONSTANT
6399  030536  012701  031340                  MOV     #BBPAT3,R1      ;USED (NOT 25) IN
6400  030542  012702  000002                  MOV     #2,R2           ;THE ALLIGN FLOWS?
6401  030546  022021                  BB23:   CMP     (R0)+,(R1)+
6402  030550  001402                          BEQ     BB24
6403  030552  000137  031172                  JMP     @#BBER5         ;DATA ERROR F
6404  030556  077205                  BB24:   SOB     R2,BB23
6405  030560  000137  031210                  JMP     @#BBER6         ;BAD CONSTANT F
6406  030564  077220                  BB25:   SOB     R2,BB22
6407  030566  020405                          CMP     R4,R5
6408  030570  001402                          BEQ     BB26
6409  030572  000137  031006                  JMP     @#BBER10        ;BAD FPS.
6410                                  ;EXPONENT DIFFERENCE=1
6411  030576                          BB26:
6412  030576  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
6413  030600  012737  030626  001236          MOV     #BB27,@#$TMP2
6414  030606  012704  003200                  MOV     #3200,R4
6415  030612  170104                          LDFPS   R4              ;SET UP AC0 OPERAND
6416  030614  012700  031360                  MOV     #BBPAT5,R0
6417  030620  172410                          LDD     (R0),AC0
6418  030622  012700  031320                  MOV     #BBPAT1,R0      ;FSRC
6419  030626  172010                  BB27:   ADDD    (R0),AC0        ;TEST INSTRUCTION
6420  030630  170205                          STFPS   R5
6421  030632  012700  031300                  MOV     #BBDAT0,R0      ;GET THE RESULT.
6422  030636  174010                          STD     AC0,(R0)
6423  030640  012701  031420                  MOV     #BBP11,R1       ;IS IT CORRECT?
6424  030644  012702  000004                  MOV     #4,R2
6425  030650  022021                  BB30:   CMP     (R0)+,(R1)+
6426  030652  001402                          BEQ     BB31
6427  030654  000137  031244                  JMP     @#BBER7         ;DATA ERROR D
6428  030660  077205                  BB31:   SOB     R2,BB30
6429  030662  020405                          CMP     R4,R5           ;IS FPS CORRECT
6430  030664  001402                          BEQ     BB32
6431  030666  000137  030766                  JMP     @#BBER0
6432                                  ;EXPONENT DIFFERENCE=100=144 (OCT)
6433  030672                          BB32:
6434  030672  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
6435  030674  012737  030722  001236          MOV     #BB33,@#$TMP2
6436  030702  012704  003200                  MOV     #3200,R4
6437  030706  170104                          LDFPS   R4              ;SET FIV,FIV AND FD
6438  030710  012700  031370                  MOV     #BBPAT6,R0      ;SET UP AC0 OPERAND.
6439  030714  172410                          LDD     (R0),AC0
6440  030716  012700  031320                  MOV     #BBPAT1,R0      ;FSRC
6441  030722  172010                  BB33:   ADDD    (R0),AC0        ;TEST INSTRUCTION
6442  030724  170205                          STFPS   R5
6443  030726  012700  031300                  MOV     #BBDAT0,R0      ;GET THE RESULT
6444  030732  174010                          STD     AC0,(R0)
6445  030734  012701  031370                  MOV     #BBPAT6,R1      ;IS IT CORRECT
6446  030740  012702  000004                  MOV     #4,R2
6447  030744  022021                  BB34:   CMP     (R0)+,(R1)+
6448  030746  001402                          BEQ     BB35
6449  030750  000137  031262                  JMP     @#BBER8         ;DATA ERROR D
```

```
6450  030754  077205                  BB35:   SOB     R2,BB34
6451  030756  020405                          CMP     R4,R5           ;IS FPS CORRECT
6452  030760  001002                          BNE     BBERD
6453  030762  000167  000442                  JMP     BBDONE
6454  030766  010437  001242          BBERD:  MOV     R4,@#$TMP4      ;FPS ERROR D
6455  030772  010537  001240                  MOV     R5,@#$TMP3
6456  030776  104164                  1$:     ERROR   164
6457  031000  104412                          RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
6458                                                                  ;SEE IF THE USER HAS EXPRESSED
6459                                                                  ;THE DESIRE TO CHANGE THE SOFTWARE
6460                                                                  ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
6461                                                                  ;THE USER TYPED CONTROL G?).
6462  031002  000137  031430                  JMP     @#BBDONE
6463  031006  010437  001242          BBER10: MOV     R4,@#$TMP4      ;FPS ERROR F
6464  031012  010537  001240                  MOV     R5,@#$TMP3
6465  031016  104165                  1$:     ERROR   165
6466  031020  104412                          RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
6467                                                                  ;SEE IF THE USER HAS EXPRESSED
6468                                                                  ;THE DESIRE TO CHANGE THE SOFTWARE
6469                                                                  ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
6470                                                                  ;THE USER TYPED CONTROL G?).
6471  031022  000137  031430                  JMP     @#BBDONE
6472  031026  012737  031330  001242  BBER1:  MOV     #BBPAT2,@#$TMP4 ;DATA ERROR D
6473  031034  012737  031330  001246          MOV     #BBPAT2,@#$TMP6
6474  031042  012737  031320  001240  BBER11: MOV     #BBPAT1,@#$TMP3
6475  031050  012737  031300  001244          MOV     #BBDAT0,@#$TMP5
6476  031056  104166                  1$:     ERROR   166
6477  031060  000137  031430                  JMP     @#BBDONE
6478  031064  012737  031350  001242  BBER2:  MOV     #BBPAT4,@#$TMP4
6479  031072  012737  031410  001246          MOV     #BBP10,@#$TMP6
6480  031100  000760                          BR      BBER11
6481  031102  012737  031350  001242  BBER3:  MOV     #BBPAT4,@#$TMP4 ;BAD CONSTANT D
6482  031110  012737  031410  001246          MOV     #BBP10,@#$TMP6
6483  031116  012737  031320  001240          MOV     #BBPAT1,@#$TMP3
6484  031124  012737  031300  001244          MOV     #BBDAT0,@#$TMP5
6485  031132  104167                  1$:     ERROR   167
6486  031134  000535                          BR      BBDONE
6487  031136  012737  031310  001242  BBER4:  MOV     #BBPAT0,@#$TMP4 ;DATA ERROR F
6488  031144  012737  031310  001246          MOV     #BBPAT0,@#$TMP6
6489  031152  012737  031320  001240  BBER40: MOV     #BBPAT1,@#$TMP3
6490  031160  012737  031300  001244          MOV     #BBDAT0,@#$TMP5
6491  031166  104170                  1$:     ERROR   170
6492  031170  000517                          BR      BBDONE
6493  031172  012737  031340  001242  BBER5:  MOV     #BBPAT3,@#$TMP4
6494  031200  012737  031400  001246          MOV     #BBP7,@#$TMP6
6495  031206  000761                          BR      BBER40
6496  031210  012737  031340  001242  BBER6:  MOV     #BBPAT3,@#$TMP4 ;CONSTANT ERROR F
6497  031216  012737  031400  001246          MOV     #BBP7,@#$TMP6
6498  031224  012737  031320  001240          MOV     #BBPAT1,@#$TMP3
6499  031232  012737  031300  001244          MOV     #BBDAT0,@#$TMP5
6500  031240  104171                  1$:     ERROR   171
6501  031242  000472                          BR      BBDONE
6502  031244  012737  031360  001242  BBER7:  MOV     #BBPAT5,@#$TMP4
6503  031252  012737  031320  001246          MOV     #BBPAT11,@#$TMP6
6504  031260  000670                          BR      BBER11
6505  031262  012737  031370  001242  BBER8:  MOV     #BBPAT6,@#$TMP4
```

B10


```
6506  031270  012737  031370  031246            MOV     #BBPAT6,@#STMP6
6507  031276  000661                                    BR      BBER11
6508  031300  000000                    BBDAT0: 0
6509  031302  000000                            0
6510  031304  000000                            0
6511  031306  000000                            0
6512  031310  006400                    BBPAT0: 6400                            ;E(AC)=E(FSRC)+25=26
6513  031312  000000                            0                               ;                =32(OCT)
6514  031314  000000                            0
6515  031316  000000                            0
6516  031320  000200                    BBPAT1: 200
6517  031322  000000                            0                               ;E(FSRC)=1
6518  031324  000000                            0
6519  031326  000000                            0
6520  031330  016400                    BBPAT2: 16400
6521  031332  000000                            0                               ;E(AC)=E(FSRC)+57=58
6522  031334  000000                            0                               ;                =72(OCT)
6523  031336  000000                            0
6524  031340  006200                    BBPAT3: 6200                            ;E(AC)=E(FSRC)+24=25
6525  031342  000000                            0                               ;                =31(OCT)
6526  031344  000000                            0
6527  031346  000000                            0
6528  031350  016200                    BBPAT4: 16200                           ;E(AC)=E(FSRE)+56=57
6529  031352  000000                            0                               ;                =71(OCT)
6530  031354  000000                            0
6531  031356  000000                            0
6532  031360  000400                    BBPAT5: 400                             ;E(AC)=E(FSRC)+1=2
6533  031362  000000                            0
6534  031364  000000                            0
6535  031366  000000                            0
6536  031370  031200                    BBPAT6: 31200                           ;E(AC)=E(FSRC)+100=101
6537  031372  000000                            0                               ;                =145(OCT)
6538  031374  000000                            0
6539  031376  000000                            0
6540  031400  006200                    BBP7:   6200                            ;BBPAT3 RES
6541  031402  000001                            1
6542  031404  000000                            0
6543  031406  000000                            0
6544  031410  016200                    BBP10:  16200                           ;BBPAT4 RES
6545  031412  000000                            0
6546  031414  000000                            0
6547  031416  000001                            1
6548  031420  000500                    BBP11:  500                             ;BBPAT5 RES
6549  031422  000000                            0
6550  031424  000000                            0
6551  031426  000000                            0
6552  031430                            BBDONE:
6553  031430  104412                            RSETUP                          ;GO INITIALIZE THE FPS AND STACK; AND
6554                                                                            ;SEE IF THE USER HAS EXPRESSED
6555                                                                            ;THE DESIRE TO CHANGE THE SOFTWARE
6556                                                                            ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
6557                                                                            ;THE USER TYPED CONTROL G?).
6558                                    ;;*********************************************************************
6559                                    ;*TEST 34       ADDD WITH NEGATVE OPRANDS TEST
6560                                    ;*
6561                                    ;*THIS IS A TEST OF THE ADDD INSTRUCTION
```

```
                                        ;*WITH NEGATIVE OPERANDS.  EVERY COMBINATION OF
                                        ;*OPERAND SIGNS IS TRIED.
                                        ;*
                                        ;*************************************************************
031432  000004                          ST34:   SCOPE
                                        ;BOTH OPERANDS NEGATIVE
031434                                  DD1:
031434  104413                                  LPERR                           ;SET UP THE LOOP ON ERROR ADDRESS.
031436  012704  003200                          MOV     #3200,R4                ;SET FIO, FIV, AND FD
031442  170104                                  LDFPS   R4
031444  012737  031464  001236                  MOV     #DD2,@#$TMP2
031452  012700  033314                          MOV     #DDP1,R0                        ;SET ACO OPERAND
031456  172410                                  LDD     (R0),AC0
031460  012700  033314                          MOV     #DDP1,R0                        ;FSRC
031464  172010                          DD2:    ADDD    (R0),AC0                        ;TEST INSTRUCTION
031466  170205                                  STFPS   R5                              ;GET FPS
031470  012700  033274                          MOV     #DDDATO,R0                      ;GET THE RESULT
031474  174010                                  STD     AC0,(R0)
031476  012701  033414                          MOV     #DDP9,R1                        ;IS IT CORRECT
031502  012702  000004                          MOV     #4,R2
031506  022021                          DD3:    CMP     (R0)+,(R1)+
031510  001415                                  BEQ     DD6
031512  012700  033274                          MOV     #DDDATO,R0                      ;DID A ADD-SUB
031516  012701  033344                          MOV     #DDP4,R1                        ;FLOW A FAILURE
031522  012702  000004                          MOV     #4,R2
031526  022021                          DD4:    CMP     (R0)+,(R1)+
031530  001402                                  BEQ     DD5                             ;216,442,503
031532  000137  032524                          JMP     @#DDER1                         ;DATA ERROR,D
031536  077205                          DD5:    SOB     R2,DD4
031540  000137  032562                          JMP     @#DDER2                         ;FLOW FAILURE,D
031544  077220                          DD6:    SOB     R2,DD3
031546  052704  000010                          BIS     #10,R4
031552  020405                                  CMP     R4,R5                           ;FPS CORRECT?
031554  001402                                  BEQ     DD7
031556  000137  032506                          JMP     @#DDER0                         ;BAD,FPS
                                        ;AC POS FSRC NEG        AC=-FSRC
031562                                  DD7:
031562  104413                                  LPERR                           ;SET UP THE LOOP ON ERROR ADDRESS.
031564  012704  003200                          MOV     #3200,R4                        ;SET FIO, FIV, AND FD
031570  170104                                  LDFPS   R4
031572  012737  031612  001236                  MOV     #DD8,@#$TMP2
031600  012700  033324                          MOV     #DDP2,R0                        ;SET ACO OPERAND
031604  172410                                  LDD     (R0),AC0
031606  012700  033314                          MOV     #DDP1,R0                        ;FSRC
031612  172010                          DD8:    ADDD    (R0),AC0                        ;TEST INSTRUCTION
031614  170205                                  STFPS   R5                              ;GET FPS
031616  012700  033274                          MOV     #DDDATO,R0                      ;GET THE RESULT
031622  174010                                  STD     AC0,(R0)
031624  012701  033304                          MOV     #DDP0,R1                        ;IS IT CORRECT
031630  012702  000004                          MOV     #4,R2
031634  022021                          DD10:   CMP     (R0)+,(R1)+
031636  001402                                  BEQ     DD11
031640  000137  032620                          JMP     @#DDER3                         ;FLOW FAILURE
031644  077205                          DD11:   SOB     R2,DD10
031646  052704  000004                          BIS     #4,R4
031652  020405                                  CMP     R4,R5                           ;FPS CORRECT?
```

```
031654  001402                          BEQ     DD12
031656  000137  032506                  JMP     @#DDER0                         ;BAD FPS
                                ;AC NEG FSRC POS                AC=-FSRC
031662                          DD12:
031662  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
031664  012704  003200                  MOV     #3200,R4                        ;SET FIU, FIV, AND FD
031670  170104                          LDFPS   R4
031672  012737  031712  001236          MOV     #DD13,@#$TMP2
031700  012700  033314                  MOV     #DDP1,R0                        ;SET AC0 OPERAND
031704  172410                          LDD     (R0),AC0
031706  012700  033324                  MOV     #DDP2,R0                        ;FSRC
031712  172010                  DD13:   ADDD    (R0),AC0                        ;TEST INSTRUCTION
031714  170205                          STFPS   R5                              ;GET FPS
031716  012700  033274                  MOV     #DDDAT0,R0                      ;GET THE RESULT
031722  174010                          STD     AC0,(R0)
031724  012701  033304                  MOV     #DDPC,R1                        ;IS IT CORRECT
031730  012702  000004                  MOV     #4,R2
031734  022021                  DD14:   CMP     (R0)+,(R1)+
031736  001402                          BEQ     DD15
031740  000137  032656                  JMP     @#DDER4                         ;FLOW FAILURE 216,440,121
031744  077205                  DD15:   SOB     R2,DD14
031746  052704  000004                  BIS     #4,R4
031752  020405                          CMP     R4,R5                           ;EPS CORRECT?
031754  001402                          BEQ     DD16
031756  000137  032506                  JMP     @#DDER0                         ;BAD FPS
                                ;ACC POS                FSRC    NEG     AC/ \   FSRC/
031762                          DD16:
031762  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
031764  012704  003200                  MOV     #3200,R4                        ;SET FIV, FIV AND FD
031770  170104                          LDFPS   R4
031772  012737  032012  001236          MOV     #DD17,@#$TMP2
032000  012700  033334                  MOV     #DDP3,R0                        ;SET AC0 OPERAND
032004  172410                          LDD     (R0),AC0
032006  012700  033364                  MOV     #DDP6,R0                        ;ESPC
032012  172010                  DD17:   ADDD    (R0),AC0                        ;TEST INSTRUCTION
032014  170205                          STFPS   R5                              ;GET FPS
032016  012700  033274                  MOV     #DDDAT0,R0                      ;GET THE RESULT
032022  174010                          STD     AC0,(R0)
032024  012701  033374                  MOV     #DDP7,R1                        ;IS IT CORRECT
032030  012702  000004                  MOV     #4,R2
032034  022021                  DD18:   CMP     (R0)+,(R1)+
032036  001415                          BEQ     DD21
032040  012700  033274                  MOV     #DDDAT0,R0                      ;FLOWS FAILURE
032044  012701  033404                  MOV     #DDP8,R1                        ;216,440,101
032050  012702  000004                  MOV     #4,R2                           ;GET GENERATED
032054  022021                  DD19:   CMP     (R0)+,(R1)+
032056  001402                          BEQ     DD20
032060  000137  032714                  JMP     @#DDER5                 ;DATA ERROR.
032064  077205                  DD20:   SOB     R2,DD19
032066  000137  032752                  JMP     @#DDER6
032072  077220                  DD21:   SOB     R2,DD18
032074  020405                          CMP     R4,R5                           ;EPS CORRECT?
032076  001402                          BEQ     DD22
032100  000137  032506                  JMP     @#DDER0                         ;BAD FPS
                                ;AC NEG FSRC    POS     /FSRC/ \ AC
032104                          DD22:
```

```
032104  104413                          LPERR                           ;SET UP THE LOOP ON ERROR ADDRESS.
032106  012704  003200                  MOV     #3200,R4                        ;SET FIC,FIV, AND FD
032112  170104                          LDFPS   R4
032114  012737  032134  001236          MOV     #DD23,@#$TMP2
032122  012700  033364                  MOV     #DDP6,R0                        ;SET AC0 OPERAND
032126  172410                          LDD     (R0),AC0
032130  012700  033334                  MOV     #DDP3,R0                        ;FSPC
032134  172010                  DD23:   ADDD    (R0),AC0                        ;TEST INSTRUCTION
032136  170205                          STFPS   R5                              ;GET FPS
032140  012700  033274                  MOV     #DDDAT0,R0                      ;GET THE RESULT
032144  174010                          STD     AC0,(R0)
032146  012701  033374                  MOV     #DDP7,R1                        ;IS IT CORRECT?
032152  012702  000004                  MOV     #4,R2
032156  022021                  DD24:   CMP     (R0)+,(R1)+
032160  001415                          BEQ     DD27
032162  012700  033274                  MOV     #DDDAT0,R0                      ;FLO.S FAILURE
032166  012701  033404                  MOV     #DDP8,R1                        ;CONSTANT (NOT 57)
032172  012702  000004                  MOV     #4,R2                           ;216,042,101
032176  021011                  DD25:   CMP     (R0),(R1
032200  001402                          BEQ     DD26
032202  000137  033010                  JMP     @#DDER7                         ;DATA ERROR.
032206  077205                  DD26:   SOB     R2,DD25
032210  000137  033046                  JMP     @#DDER8
032214  077220                  DD27:   SOB     R2,DD24
032216  020405                          CMP     R4,R5                           ;FPS CORRECT?
032220  001402                          BEQ     DD30
032222  000137  032506                  JMP     @#DDER0                         ;BAD FPS
                                ;ACC POS        FSRC    NEG     (AC) (FRSRC)
032226                          DD30:
032226  104413                          LPERR                           ;SET UP THE LOOP ON ERROR ADDRESS.
032230  012704  003200                  MOV     #3200,R4                        ;SET FIC.FIV.AND FD
032234  170104                          LDFPS   R4
032236  012737  032256  001236          MOV     #DD31,@#$TMP2
032244  012700  033344                  MOV     #DDP4,R0                        ;SET AC0 OPERAND
032250  172410                          LDD     (R0),AC0
032252  012700  033354                  MOV     #DDP5,R0                        ;FSPC
032256  172010                  DD31:   ADDD    (R0),AC0                        ;TEST INSTRUCTION
032260  170205                          STFPS   R5                              ;GET FPS
032262  012700  033274                  MOV     #DDDAT0,R0                      ;GET THE RESULT
032266  174010                          STD     AC0,(R0)
032270  012701  033404                  MOV     #DDP8,R1                        ;IS IT CORRECT
032274  012702  000004                  MOV     #4,R2
032300  022021                  DD32:   CMP     (R0)+,(R1)+
032302  001415                          BEQ     DD35                            ;ADD-SUB
032304  012700  033274                  MOV     #DDDAT0,R0                      ;FLOWAS FAILURE
032310  012701  033374                  MOV     #DDP7,R1                        ;CON 216 N440 NOT 141
032314  012702  000004                  MOV     #4,R2                   ;GET GENERATED
032320  022021                  DD33:   CMP     (R0)+,(R1)+                     ;FOR THE ALLIGNMENT
032322  001402                          BEQ     DD34                            ;FLOWS?
032324  000137  033104                  JMP     @#DDER9                         ;DATA ERROR, D
032330  077205                  DD34:   SOB     R2,DD33
032332  000137  033142                  JMP     @#DDER10
032336  077220                  DD35:   SOB     R2,DD32
032340  052704  000010                  BIS     #10,R4
032344  020405                          CMP     R4,R5                           ;FPS CORRECT?
032346  001402                          BEQ     #DD36
```

F10


```
032350  000137  032506                   JMP     @#DDERO                 ;BAD FPS
                                 ;ACC NEG        FSRC    POS     FSRC    AC.
032354                           DD36:
032354  104413                           LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
032356  012704  003200                   MOV     #3200,R4                ;SET FIO, FIV, AND FD
032362  170104                           LDFPS   R4
032364  012737  032404  001236           MOV     #DD37,@#$TMP2
032372  012700  033354                   MOV     #DDP5,R0                ;SET ACC OPERAND
032376  172410                           LDD     (R0),AC0
032400  012700  033344                   MOV     #DDP4,R0                ;FSRC
032404  172010                   DD37:   ADDD    (R0),AC0                ;TEST INSTRUCTION
032406  170205                           STFPS   R5                      ;GET FPS
032410  012700  033274                   MOV     #DDDAT0,R0              ;GET THE RESULT
032414  174010                           STD     AC0,(R0)
032416  012701  033404                   MOV     #DDP8,R1                ;IS IT CORRECT
032422  012702  000004                   MOV     #4,R2
032426  022021                   DD38:   CMP     (R0)+,(R1)+
032430  001415                           BEQ     DD41
032432  012700  033274                   MOV     #DDDAT0,R0              ;ADD SUB
032436  012701  033374                   MOV     #DDP7,R1                ;FLOWS FAILURES
032442  012702  000004                   MOV     #4,R2                   ;GET 216,042,141
032446  022021                   DD39:   CMP     (R0)+,(R1)+             ;FOR THE ALLIGNMENT
032450  001402                           BEQ     DD40                    ;FLOWS?
032452  000137  033200                   JMP     @#DDER11                ;DATA ERROR. D
032456  077206                   DD40:   SOB     R2,DD39
032460  000137  033236                   JMP     @#DDER12                ;BAD CONSTANT.D
032464  077220                   DD41:   SOB     R2,DD38
032466  052704  000010                   BIS     #10,R4
032472  020405                           CMP     R4,R5                   ;FPS CORRECT?
032474  001402                           BEQ     DD42
032476  000137  032506                   JMP     @#DDERO                 ;BAD FPS
032502  000137  033424           DD42:   JMP     @#DDDONE
032506  010437  001242           DDERO:  MOV     R4,@#$TMP4              ;FPS ERROR
032512  010537  001240                   MOV     R5,@#$TMP3
032516  104164                   1$:     ERROR   164
032520  000137  033424                   JMP     @#DDDONE
032524                           DDER1:
032524  012737  033314  001240           MOV     #DDP1,@#$TMP3
032532  012737  033314  001242           MOV     #DDP1,@#$TMP4
032540  012737  033274  001244           MOV     #DDDAT0,@#$TMP5
032546  012737  033414  001246           MOV     #DDP9,@#$TMP6
032554  104165                   1$:     ERROR   165
032556  000137  033424                   JMP     @#DDDONE
032562                           DDER2:
032562  012737  033314  001240           MOV     #DDP1,@#$TMP3
032570  012737  033314  001242           MOV     #DDP1,@#$TMP4
032576  012737  033274  001244           MOV     #DDDAT0,@#$TMP5
032604  012737  033414  001246           MOV     #DDP9,@#$TMP6
032612  104176                   1$:     ERROR   176
032614  000137  033424                   JMP     @#DDDONE
032620                           DDER3:
032620  012737  033314  001240           MOV     #DDP1,@#$TMP3
032626  012737  033324  001242           MOV     #DDP2,@#$TMP4
032634  012737  033274  001244           MOV     #DDDAT0,@#$TMP5
032642  012737  033304  001246           MOV     #DDP0,@#$TMP6
032650  104177                   1$:     ERROR   177
```

```
032652	000137	033424			JMP	@#DDDONE
032656				DDER4:
032656	012737	033324	001240		MOV	#DDP2,@#$TMP3
032664	012737	033314	001242		MOV	#DDP1,@#$TMP4
032672	012737	033274	001244		MOV	#DDDAT0,@#$TMP5
032700	012737	033304	001246		MOV	#DDP0,@#$TMP6
032706	104200			1$:	ERROR	200
032710	000137	033424			JMP	@#DDDONE
032714				DDER5:
032714	012737	033364	001240		MOV	#DDP6,@#$TMP3
032722	012737	033334	001242		MOV	#DDP3,@#$TMP4
032730	012737	033274	001244		MOV	#DDDAT0,@#$TMP5
032736	012737	033374	001246		MOV	#DDP7,@#$TMP6
032744	104165			1$:	ERROR	165
032746	000137	033424			JMP	@#DDDONE
032752				DDER6:
032752	012737	033364	001240		MOV	#DDP6,@#$TMP3
032760	012737	033334	001242		MOV	#DDP3,@#$TMP4
032766	012737	033274	001244		MOV	#DDDAT0,@#$TMP5
032774	012737	033374	001246		MOV	#DDP7,@#$TMP6
033002	104201			1$:	ERROR	201
033004	000137	033424			JMP	@#DDDONE
033010				DDER7:
033010	012737	033334	001240		MOV	#DDP3,@#$TMP3
033016	012737	033364	001242		MOV	#DDP6,@#$TMP4
033024	012737	033274	001244		MOV	#DDDAT0,@#$TMP5
033032	012737	033374	001246		MOV	#DDP7,@#$TMP6
033040	104165			1$:	ERROR	165
033042	000137	033424			JMP	@#DDDONE
033046				DDER8:
033046	012737	033334	001240		MOV	#DDP3,@#$TMP3
033054	012737	033364	001242		MOV	#DDP6,@#$TMP4
033062	012737	033274	001244		MOV	#DDDAT0,@#$TMP5
033070	012737	033374	001246		MOV	#DDP7,@#$TMP6
033076	104202			1$:	ERROR	202
033100	000137	033424			JMP	@#DDDONE
033104				DDER9:
033104	012737	033354	001240		MOV	#DDP5,@#$TMP3
033112	012737	033344	001242		MOV	#DDP4,@#$TMP4
033120	012737	033274	001244		MOV	#DDDAT0,@#$TMP5
033126	012737	033404	001246		MOV	#DDP8,@#$TMP6
033134	104165			1$:	ERROR	165
033136	000137	033424			JMP	@#DDDONE
033142				DDER10:
033142	012737	033354	001240		MOV	#DDP5,@#$TMP3
033150	012737	033344	001242		MOV	#DDP4,@#$TMP4
033156	012737	033274	001244		MOV	#DDDAT0,@#$TMP5
033164	012737	033404	001246		MOV	#DDP8,@#$TMP6
033172	104203			1$:	ERROR	203
033174	000137	033424			JMP	@#DDDONE
033200				DDER11:
033200	012737	033344	001240		MOV	#DDP4,@#$TMP3
033206	012737	033354	001242		MOV	#DDP5,@#$TMP4
033214	012737	033274	001244		MOV	#DDDAT0,@#$TMP5
033222	012737	033404	001246		MOV	#DDP8,@#$TMP6
033230	104165			1$:	ERROR	165
```

H10


```
033232  000137  033424                    JMP     @#DDDONE
033236                          DDER12:
033236  012737  033344  001240            MOV     #DDP4,@#STMP3
033244  012737  033354  001242            MOV     #DDP5,@#STMP4
033252  012737  033274  001244            MOV     #DDDAT0,@#STMP5
033260  012737  033404  001246            MOV     #DDP8,@#STMP6
033266  104204                  1$:       ERROR   204
033270  000137  033424                    JMP     @#DDDONE
033274  000000                  DDDAT0:   0
033276  000000                            0
033300  000000                            0
033302  000000                            0
033304  000000                  DDP0:     0
033306  000000                            0
033310  000000                            0
033312  000000                            0
033314  100200                  DDP1:     100200                      ;-DDP2
033316  000000                            0
033320  000000                            0
033322  000000                            0
033324  000200                  DDP2:     200                         ;-DDP1
033326  000000                            0
033330  000000                            0
033332  000000                            0
033334  001100                  DDP3:     1100                        ;EXP=4
                                                                      ;FRAC=...110...
033336  000000                            0
033340  000000                            0
033342  000000                            0
033344  000600                  DDP4:     600                         ;EXP=3
                                                                      ;FRAC=...100...
033346  000000                            0
033350  000000                            0
033352  000000                            0
033354  101100                  DDP5:     101100                      ;-DDP3
033356  000000                            0
033360  000000                            0
033362  000000                            0
033364  100600                  DDP6:     100600                      ;-DDP4
033366  000000                            0
033370  000000                            0
033372  000000                            0
033374  001000                  DDP7:     1000                        ;DDP3+DDP6
033376  000000                            0
033400  000000                            0
033402  000000                            0
033404  101000                  DDP8:     101000                      ;DDP5+DDP4
033406  000000                            0
033410  000000                            0
033412  000000                            0
033414  100400                  DDP9:     100400                      ;DDP1+DDP1
033416  000000                            0
033420  000000                            0
033422  000000                            0
033424  000005                  DDDONE:           RESET
                                ;;************************************************************
                                ;*TEST 35         SUBD TEST
                                ;*
```

```
                                          ;* THIS IS A TEST OF THE SUBD INSTRUCTION.
                                          ;* BOTH A POSITIVE AND A NEGATIVE NUMBER
                                          ;* IS SUBTRACTED FROM IT SELF
                                          ;*
                                          ;;***********************************************************************
        033426  000004                    ST35:   SCOPE
                                                  ;USE POSITIVE OPERANDS
        033430                            EE1:
        033430  104413                            LPERR                             ;SET UP THE LOOP ON ERROR ADDRESS.
        033432  012704  003200                    MOV     #3200,R4                  ;SET FIO, FIV, AND FD
        033436  170104                            LDFPS   R4
        033440  012737  033460  001236            MOV     #EE2,@#$TMP2
        033446  012700  034140                    MOV     #EEP1,R0                  ;SET ACC OPERAND
        033452  172410                            LDD     (R0),AC0
        033454  012700  034140                    MOV     #EEP1,R0                  ;FSPC
        033460  173010                    EE2:    SUBD    (R0),AC0                  ;TEST INSTRUCTION
        033462  170205                            STFPS   R5                        ;GET FPS
        033464  012700  034116                    MOV     #EEDAT0,R0                ;GET THE RESULT
        033470  174010                            STD     AC0,(R0)
        033472  012701  034126                    MOV     #EEP0,R1                  ;IS IT CORRECT?
        033476  012702  000004                    MOV     #4,R2
        033502  022021                    EE3:    CMP     (R0)+,(R1)+
        033504  001415                            BEQ     EE6
        033506  012700  034116                    MOV     #EEDAT0,R0                ;DID A BAD
        033512  012701  034150                    MOV     #EEP2,R1                  ;CONSTANT (NOT 57)
        033516  012702  000004                    MOV     #4,R2                     ;GET GENERATED
        033522  022021                    EE4:    CMP     (R0)+,(R1)+               ;FOR THE ALLIGNMENT
        033524  001402                            BEQ     EE5                       ;FLOWS?
        033526  000137  033726                    JMP     @#EEER1                   ;DATA ERROR.D
        033532  077205                    EE5:    SOB     R2,EE4
        033534  000137  033764                    JMP     @#EEER2                   ;BAD CONSTANT.D
        033540  077220                    EE6:    SOB     R2,EE3
        033542  052704  000004                    BIS     #4,R4
        033546  020405                            CMP     R4,R5                     ;FPS CORRECT?
        033550  001402                            BEQ     EE7
        033552  000137  033710                    JMP     @#EEER0                   ;BAD FPS
                                                  ;USE NEGATIVE OPERANDS
        033556                            EE7:
        033556  104413                            LPERR                             ;SET UP THE LOOP ON ERROR ADDRESS.
        033560  012704  003200                    MOV     #3200,R4                  ;SET FIO, FIV, AND FD
        033564  170104                            LDFPS   R4
        033566  012737  033606  001236            MOV     #EE8,@#$TMP2
        033574  012700  034160                    MOV     #EEP3,R0                  ;SET ACC OPERAND
        033600  172410                            LDD     (R0),AC0
        033602  012700  034160                    MOV     #EEP3,R0                  ;FSPC
        033606  173010                    EE8:    SUBD    (R0),AC0                  ;TEST INSTRUCTION
        033610  170205                            STFPS   R5                        ;GET FPS
        033612  012700  034116                    MOV     #EEDAT0,R0                ;GET THE RESULT
        033616  174010                            STD     AC0,(R0)
        033620  012701  034126                    MOV     #EEP0,R1                  ;IS IT CORRECT?
        033624  012702  000004                    MOV     #4,R2
        033630  022021                    EE9:    CMP     (R0)+,(R1)+
        033632  001415                            BEQ     EE12
        033634  012700  034116                    MOV     #EEDAT0,R0                ;DID A BAD
        033640  012701  034170                    MOV     #EEP4,R1                  ;CONSTANT (NOT 57)
        033644  012702  000004                    MOV     #4,R2                     ;GET GENERATED
```

```
033650  022021                           EE10:   CMP     (R0)+,(R1)+             ;FOR THE ALLIGNMENT
033652  001402                                   BEQ     EE11                    ;FLOWS?
033654  000137  034022                           JMP     @#EEER3                 ;DATA ERROR.D
033660  077205                           EE11:   SOB     R2,EE10
033662  000137  034060                           JMP     @#EEER4                 ;BAD CONSTANT.D
033666  077220                           EE12:   SOB     R2,EE9
033670  052704  000004                           BIS     #4,R4
033674  020405                                   CMP     R4,R5                   ;FPS CORRECT?
033676  001402                                   BEQ     EE13
033700  000137  033710                           JMP     @#EEER0                 ;BAD FPS.
033704  000137  034200                   EE13:   JMP     @#EEDONE
033710  010437  001242                   EEER0:  MOV     R4,@#$TMP4              ;BAD FPS
033714  010537  001240                           MOV     R5,@#$TMP3
033720  104205                           1$:     ERROR   205
033722  000137  034200                           JMP     @#EEDONE
033726                                   EEER1:
033726  012737  034140  001240                   MOV     #EEP1,@#$TMP3
033734  012737  034140  001242                   MOV     #EEP1,@#$TMP4
033742  012737  034116  001244                   MOV     #EEDAT0,@#$TMP5
033750  012737  034126  001246                   MOV     #EEP0,@#$TMP6
033756  104206                           1$:     ERROR   206
033760  000137  034200                           JMP     @#EEDONE
033764                                   EEER2:
033764  012737  034140  001240                   MOV     #EEP1,@#$TMP3
033772  012737  034140  001242                   MOV     #EEP1,@#$TMP4
034000  012737  034116  001244                   MOV     #EEDAT0,@#$TMP5
034006  012737  034126  001246                   MOV     #EEP0,@#$TMP6
034014  104207                           1$:     ERROR   207
034016  000137  034200                           JMP     @#EEDONE
034022                                   EEER3:
034022  012737  034160  001240                   MOV     #EEP3,@#$TMP3
034030  012737  034160  001242                   MOV     #EEP3,@#$TMP4
034036  012737  034116  001244                   MOV     #EEDAT0,@#$TMP5
034044  012737  034126  001246                   MOV     #EEP0,@#$TMP6
034052  104206                           1$:     ERROR   206
034054  000137  034200                           JMP     @#EEDONE
034060                                   EEER4:
034060  012737  034160  001240                   MOV     #EEP3,@#$TMP3
034066  012737  034160  001242                   MOV     #EEP3,@#$TMP4
034074  012737  034116  001244                   MOV     #EEDAT0,@#$TMP5
034102  012737  034126  001246                   MOV     #EEP0,@#$TMP6
034110  104207                           1$:     ERROR   207
034112  000137  034200                           JMP     @#EEDONE
034116  000000                           EEDAT0: 0
034120  000000                                   0
034122  000000                                   0
034124  000000                                   0
034126  000000                           EEP0:   0
034130  000000                                   0
034132  000000                           00000
034134  000000                                   0
034136  000000                                   0
034140  000200                           EEP1:   200
034142  000000                                   0
034144  000000                                   0
034146  000000                                   0
```

```
                        T35     SUBD TEST

034160  000400                  EEP2:   400
034162  000000                          0
034164  000000                          0
034166  000000                          0
034170  100200                  EEP3:   100200
034172  000000                          0
034174  000000                          0
034176  000000                          0
034200  100400                  EEP4:   100400
034202  000000                          0
034204  000000                          0
034206  000000                          0
034210  104412                  EEDONE:
                                        RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
                                                                ;SEE IF THE USER HAS EXPRESSED
                                                                ;THE DESIRE TO CHANGE THE SOFTWARE
                                                                ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
                                                                ;THE USER TYPED CONTROL G?).
                                ;;*****************************************************************
                                ;*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.
                                ;*
                                ;*****************************************************************
034212  000004                  ST36:   SCOPE
                                ;USE DATA PATTERNS THAT REQUIRE ONLY ONE LEFT SHIFT TO NORMALIZE
034214                          FF1:
034214  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
034216  012704  003200                  MOV     #3200,R4                ;SET FIU, FIV, AND FD
034222  170104                          LDFPS   R4
034224  012737  034234  001236          MOV     #FF2,@#$TMP2
034232  012700  034526                  MOV     #FFP2,R0                ;SET ACO OPERAND
034236  172410                          LDD     (R0),AC0
034240  012700  034536                  MOV     #FFP3,R0                ;FSPC
034244  172010                  FF2:    ADDD    (R0),AC0                ;TEST INSTRUCTION
034246  170205                          STFPS   R5                      ;GET FPS
034252  012700  034476                  MOV     #FFDAT0,R0              ;GET THE RESULT
034256  174010                          STD     AC0,(R0)
034260  012701  034546                  MOV     #FFP4, R1               ;IS IT CORRECT
034264  012702  000004                  MOV     #4,R2
034270  022021                  FF3:    CMP     (R0)+,(R1)+
034272  001401                          BEQ     FF4
034274  000466                          BR      FFER2                   ;BAD DATA
034276  077204                  FF4:    SOB     R2,FF3
034300  020405                          CMP     R4,R5                   ;FPS CORRECT?
034302  001401                          BEQ     FF5
034304  000435                          BR      FFER0                   ;BAD FPS
                                ;USE DATA PATTERNS WHICH REQUIRE 56 LEFT SHIFTS TO NORMALIZE
                                ;THE RESULT
034306                          FF5:
034306  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
034310  012704  003200                  MOV     #3200,R4                ;SET FIU, FIV, AND FD
```

```
034302  170104                      LDFPS   R4
034304  013737  034324  001236      MOV     #FF6,@#$TMP2
034312  012700  034506              MOV     #FFP0,R0                ;SET ACO OPERAND
034316  172410                      LDD     (R0),AC0
034320  012700  034516              MOV     #FFP1,R0                ;FSRC
034324  172010          FF6:        ADDD    (R0),AC0                ;TEST INSTRUCTION
034326  170205                      STFPS   R5                      ;GET FPS
034330  012700  034476              MOV     #FFDAT0,R0              ;GET THE RESULT
034334  174010                      STD     AC0,(R0)
034336  012701  034546              MOV     #FFP4,R1                ;IS IT CORRECT
034342  012702  000004              MOV     #4,R2
034346  022021          FF7:        CMP     (R0)+,(R1)+
034350  001401                      BEQ     FF10
034352  000413                      BR      FFER1                   ;BAD
034354  077204          FF10:       SOB     R2,FF7
034356  020405                      CMP     R4,R5                   ;FPS CORRECT
034360  001401                      BEQ     FF11
034362  000401                      BR      FFER0           ;BAD FPS
034364  000474          FF11:       BR      FFDONE

034366  010537  001240  FFER0:      MOV     R5,@#$TMP3
034372  010437  001242              MOV     R4,@#$TMP4
034376  104164          1$:         ERROR   164
034400  000466                      BR      FFDONE

034402                  FFER1:
034402  012737  034516  001240      MOV     #FFP1,@#$TMP3
034410  012737  034506  001242      MOV     #FFP0,@#$TMP4
034416  012737  034476  001244      MOV     #FFDAT0,@#$TMP5
034424  012737  034546  001246      MOV     #FFP4,@#$TMP6
034432  104210          1$:         ERROR   210
034434  000137  034556              JMP     @#FFDONE

034440                  FFER2:
034440  012737  034536  001240      MOV     #FFP3,@#$TMP3
034446  012737  034526  001242      MOV     #FFP2,@#$TMP4
034454  012737  034476  001244      MOV     #FFDAT0,@#$TMP5
034462  012737  034546  001246      MOV     #FFP4,@#$TMP6
034470  104210          1$:         ERROR   210
034472  000137  034556              JMP     @#FFDONE


034476  000000          FFDAT0:     0
034500  000000                      0
034502  000000                      0
034504  000000                      0

034506  016000          FFP0:       16000
034510  000000                      0
034512  000000                      0
034514  000001                      1
034516  116000          FFP1:       116000
034520  000000                      0
034522  000000                      0
034524  000000                      0
034526  000500          FFP2:       500
```

```
7122  034530  000000                          0
7123  034532  000000                          0
7124  034534  000000                          0
7125  034536  100400          FFP3:           100400
7126  034540  000000                          0
7127  034542  000000                          0
7128  034544  000000                          0
7129  034546  000200          FFP4:           200                     ;FFP4=FFP0+FFP1
7130  034550  000000                          0                       ;    =FFP3+FFP4
7131  034552  000000                          0
7132  034554  000000                          0
7133
7134  034556                  FFDONE:
7135
7136
7137  034556                  TST37:
7138
7139
7140
7141                          .SBTTL  END OF PASS ROUTINE
7142
7143                          ;;*****************************************************************
7144                          ;*INCREMENT THE PASS NUMBER ($PASS)
7145                          ;*INDICATE END-OF-PROGRAM AFTER 1 PASSES THRU THE PROGRAM
7146                          ;*IF SW12=1 INHIBIT TRACE TRAP
7147                          ;*IF THERES A MONITOR GO TO IT
7148                          ;*IF THERE ISN'T JUMP TO LOOP
7149
7150  034556                  $EOP:
7151  034556  000004                  SCOPE
7152  034560  005067  144316          CLR     $TSTNM          ;;ZERO THE TEST NUMBER
7153  034564  005067  144512          CLR     $TIMES          ;;ZERO THE NUMBER OF ITERATIONS
7154  034570  005267  144530          INC     $PASS           ;;INCREMENT THE PASS NUMBER
7155  034574  042767  100000  144522  BIC     #100000,$PASS   ;;DON'T ALLOW A NEG. NUMBER
7156  034602  005327                  DEC     (PC)+           ;;LOOP?
7157  034604  000001          $EOPCT: .WORD   1
7158  034606  003074                  BGT     $DOAGN          ;;YES
7159  034610  012737                  MOV     (PC)+,@(PC)+    ;;RESTORE COUNTER
7160  034612  000001          $ENDCT: .WORD   1
7161  034614  034604                  $EOPCT
7162  034616  104401  034624          TYPE    ,65$            ;;TYPE ASCIZ STRING
7163  034622  000407                  BR      64$             ;;GET OVER THE ASCIZ
7164                          ;;65$:  .ASCIZ  <12><15>/END PASS #/
7165  034642                  64$:
7166  034642  016746  144456          MOV     $PASS,-(SP)     ;;SAVE $PASS FOR TYPEOUT
7167                                                          ;;TYPE PASS NUMBER IN OCTAL
7168  034646  104403                  TYPOS                   ;;GO TYPE--OCTAL ASCII
7169  034650     006                  .BYTE   6               ;;TYPE 6 DIGITS
7170  034651     000                  .BYTE   0               ;;SUPPRESS LEADING ZEROS
7171  034652  104401  034660          TYPE    ,67$            ;;TYPE ASCIZ STRING
7172  034656  000421                  BR      66$             ;;GET OVER THE ASCIZ
7173                          ;;67$:  .ASCIZ  /  TOTAL ERRORS SINCE LAST REPORT /
7174  034722                  66$:
7175  034722  016746  144164          MOV     $ERTTL,-(SP)    ;;SAVE $ERTTL FOR TYPEOUT
7176                                                          ;;TOTAL NUMBER OF ERRORS IN OCTAL
7177  034726  104403                  TYPOS                   ;;GO TYPE--OCTAL ASCII
```

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

B11


```
035062  000416                           $XTSTR: BR      6$                      ;;IF RUNNING ON THE "XOR" TESTER CHANGE
                                                                                 ;;THIS INSTRUCTION TO A "NOP" (NOP=240)
035064  013746  000004                           MOV     @#ERRVEC,-(SP)          ;;SAVE THE CONTENTS OF THE ERROR VECTOR
035070  012737  035110  000004                   MOV     #5$,@#ERRVEC            ;;SET FOR TIMEOUT
035076  005737  177060                           TST     @#177060                ;;TIME OUT ON XOR?
035102  012637  000004                           MOV     (SP)+,@#ERRVEC          ;;RESTORE THE ERROR VECTOR
035106  000463                                   BR      $SVLAD                  ;;GO TO THE NEXT TEST
035110  022626                           5$:     CMP     (SP)+,(SP)+             ;;CLEAR THE STACK AFTER A TIME OUT
035112  012637  000004                           MOV     (SP)+,@#ERRVEC          ;;RESTORE THE ERROR VECTOR
035116  000423                                   BR      7$                      ;;LOOP ON THE PRESENT TEST
035120                                   6$::*****END OF CODE FOR THE XOR TESTER*****
035120  032777  000400  144012                   BIT     #BIT08,@SWR             ;;LOOP ON SPEC. TEST?
035126  001404                                   BEQ     2$                      ;;BR IF NO
035130  127767  144004  143744                   CMPB    @SWR,$TSTNM             ;;ON THE RIGHT TEST?    SWR<7:0>
035136  001465                                   BEQ     $OVER                   ;;BR IF YES
035140  105767  143737                   2$:     TSTB    $ERFLG                  ;;HAS AN ERROR OCCURRED?
035144  001421                                   BEQ     3$                      ;;BR IF NO
035146  126767  143743  143727                   CMPB    $ERMAX,$ERFLG           ;;MAX. ERRORS FOR THIS TEST OCCURRED?
035154  101015                                   BHI     3$                      ;;BR IF NO
035156  032777  001000  143754                   BIT     #BIT09,@SWR             ;;LOOP ON ERROR?
035164  001404                                   BEQ     4$                      ;;BR IF NO
035166  016767  143716  143712           7$:     MOV     $LPERR,$LPADR           ;;SET LOOP ADDRESS TO LAST SCOPE
035174  000446                                   BR      $OVER
035176  105067  143701                   4$:     CLRB    $ERFLG                  ;;ZERO THE ERROR FLAG
035202  005067  144074                           CLR     $TIMES                  ;;CLEAR THE NUMBER OF ITERATIONS TO MAKE
035206  000415                                   BR      1$                      ;;ESCAPE TO THE NEXT TEST
035210  032777  004000  143722           3$:     BIT     #BIT11,@SWR             ;;INHIBIT ITERATIONS?
035216  001011                                   BNE     1$                      ;;BR IF YES
035220  005767  144100                           TST     $PASS                   ;;IF FIRST PASS OF PROGRAM
035224  001406                                   BEQ     1$                      ;;      INHIBIT ITERATIONS
035226  005267  143652                           INC     $ICNT                   ;;INCREMENT ITERATION COUNT
035232  026767  144044  143644                   CMP     $TIMES,$ICNT            ;;CHECK THE NUMBER OF ITERATIONS MADE
035240  002024                                   BGE     $OVER                   ;;BR IF MORE ITERATION REQUIRED
035242  012767  000001  143634           1$:     MOV     #1,$ICNT                ;;REINITIALIZE THE ITERATION COUNTER
035250  016767  000052  144024                   MOV     $MXCNT,$TIMES           ;;SET NUMBER OF ITERATIONS TO DO
035256  105267  143620                   $SVLAD: INCB    $TSTNM                  ;;COUNT TEST NUMBERS
035262  116767  143614  144032                   MOVB    $TSTNM,$TESTN           ;;SET TEST NUMBER IN APT MAILBOX
035270  011667  143612                           MOV     (SP),$LPADR             ;;SAVE SCOPE LOOP ADDRESS
035274  011667  143610                           MOV     (SP),$LPERR             ;;SAVE ERROR LOOP ADDRESS
035300  005067  144000                           CLR     $ESCAPE                 ;;CLEAR THE ESCAPE FROM ERROR ADDRESS
035304  112767  000001  143603                   MOVB    #1,$ERMAX               ;;ONLY ALLOW ONE(1) ERROR ON NEXT TEST
035312  016777  143564  143622           $OVER:  MOV     $TSTNM,@DISPLAY         ;;DISPLAY TEST NUMBER
035320  016716  143562                           MOV     $LPADR,(SP)             ;;FUDGE RETURN ADDRESS
035324  000002                                   RTI                             ;;FIXES PS
035326  000001                           $MXCNT: 1                               ;;MAX. NUMBER OF ITERATIONS

                                         .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
```

```
                                        ERROR HANDLER ROUTINE

                                        ;*SWC9=1        LOOP ON ERROR
                                        ;*CALL
                                        ;*      ERROR   N       ;;ERROR=EMT AND N=ERROR ITEM NUMBER

035330                                  $ERROR:
035330  104406                                  CKSWR                           ;;TEST FOR CHANGE IN SOFT-SWR
035332  105267  143545                  7$:     INCB    $ERFLG                  ;;SET THE ERROR FLAG
035336  001775                                  BEQ     7$                      ;;DON'T LET THE FLAG GO TO ZERO
035340  016777  143536  143574                  MOV     $TSTNM,@DISPLAY         ;;DISPLAY TEST NUMBER AND ERROR FLAG
035346  032777  002000  143564                  BIT     #BIT10,@SWR             ;;BELL ON ERROR?
035354  001402                                  BEQ     1$                      ;;NO - SKIP
035356  104401  001306                          TYPE    ,$BELL                  ;;RING BELL
035362  005267  143524                  1$:     INC     $ERTTL                  ;;COUNT THE NUMBER OF ERRORS
035366  011667  143524                          MOV     (SP),$ERRPC             ;;GET ADDRESS OF ERROR INSTRUCTION
035372  162767  000002  143516                  SUB     #2,$ERRPC
035400  117767  143512  143506                  MOVB    @$ERRPC,$ITEMB          ;;STRIP AND SAVE THE ERROR ITEM CODE
035406  032777  020000  143524                  BIT     #BIT13,@SWR             ;;SKIP TYPEOUT IF SET
035414  001004                                  BNE     20$                     ;;SKIP TYPEOUTS
035416  004767  002124                          JSR     PC,ERTYPE               ;;GO TO USER ERROR ROUTINE
035422  104401  001313                          TYPE    ,$CRLF
035426                                  20$:
035426  122767  000001  143732                  CMPB    #APTENV,$ENV            ;;RUNNING IN APT MODE
035434  001007                                  BNE     2$                      ;;NO,SKIP APT ERROR REPORT
035436  116767  143452  000004                  MOVB    $ITEMB,21$              ;;SET ITEM NUMBER AS ERROR NUMBER
035444  004767  000740                          JSR     PC,$ATY4                ;;REPORT FATAL ERROR TO APT
035450     000                          21$:    .BYTE   0
035451     000                                  .BYTE   0
035452  000777                          22$:    BR      22$                     ;;APT ERROR LOOP
035454  005777  143460                  2$:     TST     @SWR                    ;;HALT ON ERROR
035460  100002                                  BPL     3$                      ;;SKIP IF CONTINUE
035462  000000                                  HALT                            ;;HALT ON ERROR!
035464  104406                                  CKSWR                           ;;TEST FOR CHANGE IN SOFT-SWR
035466  032777  001000  143444          3$:     BIT     #BIT09,@SWR             ;;LOOP ON ERROR SWITCH SET?
035474  001402                                  BEQ     4$                      ;;BR IF NO
035476  016716  143406                          MOV     $LPERR,(SP)             ;;FUDGE RETURN FOR LOOPING
035502  005767  143576                  4$:     TST     $ESCAPE                 ;;CHECK FOR AN ESCAPE ADDRESS
035506  001402                                  BEQ     5$                      ;;BR IF NONE
035510  016716  143570                          MOV     $ESCAPE,(SP)            ;;FUDGE RETURN ADDRESS FOR ESCAPE
035514                                  5$:
035514  022737  034770  000042                  CMP     #$ENDAD,@#42            ;;ACT-11 AUTO-ACCEPT?
035522  001001                                  BNE     6$                      ;;BRANCH IF NO
035524  000000                                  HALT                            ;;YES
035526                                  6$:
035526  032777  001000  143404                  BIT     #BIT09,@SWR
035534  001013                                  BNE     ERM10
035536  011637  001162                          MOV     (SP),@#$REGO                    ;SEE IF ERROR #377
035542  062737  177776  001162                  ADD     #-2,@#$REGO
035550  122777  000377  143404                  CMPB    #377,@$REGO
035556  001002                                  BNE     ERM10
035560  062716  000002                          ADD     #2,(SP)
035564  000002                          ERM10:  RTI


                                        .SBTTL  SAVE AND RESTORE R0-R5 ROUTINES

                                        ;;**************************************************************
```

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

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

                                        .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:         $FILLS CONTAINS THE CHARACTER TO FILL AFTER.
                                        ;*
                                        ;*CALL:
                                        ;*1) USING A TRAP INSTRUCTION
                                        ;*      TYPE    ,MESADR         ;;MESADR IS FIRST ADDRESS OF AN ASCII STRING
                                        ;*OR
```

```
                                        ;*      TYPE
                                        ;*      MESADR
                                        ;*
035662  105767  142271          $TYPE:  TSTB    $TPFLG          ;;IS THERE A TERMINAL?
035666  100002                          BPL     1$              ;;BR IF YES
XXXXXX  XXXXXX                          HALT                    ;;HALT HERE IF NO TERMINAL
XXXXXX  000430                          BR      3$              ;;LEAVE
035674  010046                  1$:     MOV     R0,-(SP)        ;;SAVE R0
035676  016600  000002                  MOV     2(SP),R0        ;;GET ADDRESS OF ASCIZ STRING
035702  122767  000001  143426          CMPB    #APTENV,$ENV    ;;RUNNING IN APT MODE
035710  001011                          BNE     62$             ;;NO,GO CHECK FOR APT CONSOLE
035712  132767  000100  143417          BITB    #APTSPOOL,$ENVM ;;SPOOL MESSAGE TO APT
035720  001405                          BEQ     62$             ;;NO,GO CHECK FOR CONSOLE
035722  010067  000004                  MOV     R0,61$          ;;SETUP MESSAGE ADDRESS FOR APT
035726  004767  000446                  JSR     PC,$ATY3        ;;SPOOL MESSAGE TO APT
035732  000000                  61$:    .WORD   0               ;;MESSAGE ADDRESS
035734  132767  000040  143375  62$:    BITB    #APTCSUP,$ENVM  ;;APT CONSOLE SUPPRESSED .
035742  001003                          BNE     60$             ;;YES,SKIP TYPE OUT
035744  112046                  2$:     MOVB    (R0)+,-(SP)     ;;PUSH CHARACTER TO BE TYPED ONTO STACK
035746  001005                          BNE     4$              ;;BR IF IT ISN'T THE TERMINATOR
035750  005726                          TST     (SP)+           ;;IF TERMINATOR POP IT OFF THE STACK
035752  012600                  60$:    MOV     (SP)+,R0        ;;RESTORE R0
035754  062716  000002          3$:     ADD     #2,(SP)         ;;ADJUST RETURN PC
035760  000002                          RTI                     ;;RETURN
035762  122716  000011          4$:     CMPB    #HT,(SP)        ;;BRANCH IF <HT>
035766  001430                          BEQ     9$
035770  122716  000200                  CMPB    #CRLF,(SP)      ;;BRANCH IF NOT <CRLF>
035774  001006                          BNE     5$
035776  005726                          TST     (SP)+           ;;POP <CR><LF> EQUIV
036000  104401                          TYPE                    ;;TYPE A CR AND LF
036002  001313                          $CRLF
036004  105067  000130                  CLRB    $CHARCNT        ;;CLEAR CHARACTER COUNT
036010  000755                          BR      2$              ;;GET NEXT CHARACTER
036012  004767  000056          5$:     JSR     PC,$TYPEC       ;;GO TYPE THIS CHARACTER
036016  126726  143134          6$:     CMPB    $FILLC,(SP)+    ;;IS IT TIME FOR FILLER CHARS.?
036022  001350                          BNE     2$              ;;IF NO GO GET NEXT CHAR.
036024  016746  143124                  MOV     $NULL,-(SP)     ;;GET # OF FILLER CHARS. NEEDED
                                                                ;;AND THE NULL CHAR.
036030  105366  000001          7$:     DECB    1(SP)           ;;DOES A NULL NEED TO BE TYPED?
036034  002770                          BLT     6$              ;;BR IF NO--GO POP THE NULL OFF OF STACK
036036  004767  000032                  JSR     PC,$TYPEC       ;;GO TYPE A NULL
036042  105367  000072                  DECB    $CHARCNT        ;;DO NOT COUNT AS A COUNT
036046  000770                          BR      7$              ;;LOOP

                                ;HORIZONTAL TAB PROCESSOR

036050  112716  000040          8$:     MOVB    #' ,(SP)        ;;REPLACE TAB WITH SPACE
036054  004767  000014          9$:     JSR     PC,$TYPEC       ;;TYPE A SPACE
036060  132767  000007  000052          BITB    #7,$CHARCNT     ;;BRANCH IF NOT AT
036066  001372                          BNE     9$              ;;TAB STOP
036070  005726                          TST     (SP)+           ;;POP SPACE OFF STACK
036072  000724                          BR      2$              ;;GET NEXT CHARACTER
036074  105777  143050          $TYPEC: TSTB    @$TPS           ;;WAIT UNTIL PRINTER IS READY
036100  100375                          BPL     $TYPEC
036102  116677  000002  143042          MOVB    2(SP),@$TPB     ;;LOAD CHAR TO BE TYPED INTO DATA REG.
```

F11


```
036110  122766  000015  000002          CMPB    #CR,2(SP)       ;;IS CHARACTER A CARRIAGE RETURN?
036116  001003                          BNE     1$              ;;BRANCH IF NO
036120  105067  000014                  CLRB    $CHARCNT        ;;YES--CLEAR CHARACTER COUNT
036124  000406                          BR      $TYPEX          ;;EXIT
036126  122766  000012  000002  1$:     CMPB    #LF,2(SP)       ;;IS CHARACTER A LINE FEED?
036134  001402                          BEQ     $TYPEX          ;;BRANCH IF YES
036136  105227                          INCB    (PC)+           ;;COUNT THE CHARACTER
036140  000000                  $CHARCNT:.WORD  0               ;;CHARACTER COUNT STORAGE
036142  000207                  $TYPEX: RTS     PC

                                .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

036144  017646  000000          $TYPOS: MOV     @(SP),-(SP)     ;;PICKUP THE MODE
036150  116667  000001  000211          MOVB    1(SP),$OFILL    ;;LOAD ZERO FILL SWITCH
036156  112667  000207                  MOVB    (SP)+,$OMODE+1  ;;NUMBER OF DIGITS TO TYPE
036162  062716  000002                  ADD     #2,(SP)         ;;ADJUST RETURN ADDRESS
036166  000406                          BR      $TYPON
036170  112767  000001  000171  $TYPOC: MOVB    #1,$OFILL       ;;SET THE ZERO FILL SWITCH
036176  112767  000006  000165          MOVB    #6,$OMODE+1     ;;SET FOR SIX(6) DIGITS
036204  112767  000005  000154  $TYPON: MOVB    #5,$OCNT        ;;SET THE ITERATION COUNT
036212  010346                          MOV     R3,-(SP)        ;;SAVE R3
036214  010446                          MOV     R4,-(SP)        ;;SAVE R4
036216  010546                          MOV     R5,-(SP)        ;;SAVE R5
036220  116704  000145                  MOVB    $OMODE+1,R4     ;;GET THE NUMBER OF DIGITS TO TYPE
036224  005404                          NEG     R4
036226  062704  000006                  ADD     #6,R4           ;;SUBTRACT IT FOR MAX. ALLOWED
036232  110467  000132                  MOVB    R4,$OMODE       ;;SAVE IT FOR USE
036236  116704  000125                  MOVB    $OFILL,R4       ;;GET THE ZERO FILL SWITCH
036242  016605  000012                  MOV     12(SP),R5       ;;PICKUP THE INPUT NUMBER
036246  005003                          CLR     R3              ;;CLEAR THE OUTPUT WORD
036250  006105                  1$:     ROL     R5              ;;ROTATE MSB INTO "C"
036252  000404                          BR      3$              ;;GO DO MSB
```

```
036254  006105                      2$:      ROL     R5                  ;;FORM THIS DIGIT
036256  006105                               ROL     R5
036260  006105                               ROL     R5
036262  010503                               MOV     R5,R3
036264  006103                      3$:      ROL     R3                  ;;GET LSB OF THIS DIGIT
036266  105367  000076                       DECB    $OMCODE             ;;TYPE THIS DIGIT?
036272  100016                               BPL     7$                  ;;BR IF NO
036274  042703  177770                       BIC     #177770,R3          ;;GET RID OF JUNK
036300  001002                               BNE     4$                  ;;TEST FOR 0
036302  005704                               TST     R4                  ;;SUPPRESS THIS 0?
036304  001403                               BEQ     5$                  ;;BR IF YES
036306  005204                      4$:      INC     R4                  ;;DON'T SUPPRESS ANYMORE 0'S
036310  052703  000060                       BIS     #'0,R3              ;;MAKE THIS DIGIT ASCII
036314  052703  000040              5$:      BIS     #' ,R3              ;;MAKE ASCII IF NOT ALREADY
036320  110367  000040                       MOVB    R3,8$               ;;SAVE FOR TYPING
036324  104401  036364                       TYPE    ,8$                 ;;GO TYPE THIS DIGIT
036330  105367  000032              7$:      DECB    $OCNT               ;;COUNT BY 1
036334  003347                               BGT     2$                  ;;BR IF MORE TO DO
036336  002402                               BLT     6$                  ;;BR IF DONE
036340  005204                               INC     R4                  ;;INSURE LAST DIGIT ISN'T A BLANK
036342  000744                               BR      2$                  ;;GO DO THE LAST DIGIT
036344  012605                      6$:      MOV     (SP)+,R5            ;;RESTORE R5
036346  012604                               MOV     (SP)+,R4            ;;RESTORE R4
036350  012603                               MOV     (SP)+,R3            ;;RESTORE R3
036352  016666  000002  000004               MOV     2(SP),4(SP)         ;;SET THE STACK FOR RETURNING
036360  012616                               MOV     (SP)+,(SP)
036362  000002                               RTI                         ;;RETURN
036364     000                      8$:      .BYTE   0                   ;;STORAGE FOR ASCII DIGIT
036365     000                               .BYTE   0                   ;;TERMINATOR FOR TYPE ROUTINE
036366     000                      $OCNT:   .BYTE   0                   ;;OCTAL DIGIT COUNTER
036367     000                      $OFILL:  .BYTE   0                   ;;ZERO FILL SWITCH
036370  000000                      $OMODE:  .WORD   0                   ;;NUMBER OF DIGITS TO TYPE

                                    .SBTTL  APT COMMUNICATIONS ROUTINE

                                    ;;*******************************************************************
036372  112767  000001  000236      $ATY1:   MOVB    #1,$FFLG            ;;TO REPORT FATAL ERROR
036400  112767  000001  000226      $ATY3:   MOVB    #1,$MFLG            ;;TO TYPE A MESSAGE
036406  000403                               BR      $ATYC
036410  112767  000001  000220      $ATY4:   MOVB    #1,$FFLG            ;;TO ONLY REPORT FATAL ERROR
036416                              $ATYC:
036416  010046                               MOV     R0,-(SP)            ;;PUSH R0 ON STACK
036420  010146                               MOV     R1,-(SP)            ;;PUSH R1 ON STACK
036422  105767  000206                       TSTB    $MFLG               ;;SHOULD TYPE A MESSAGE?
036426  001450                               BEQ     5$                  ;;IF NOT: BR
036430  122767  000001  142700               CMPB    #APTENV,$ENV        ;;OPERATING UNDER APT?
036436  001031                               BNE     3$                  ;;IF NOT: BR
036440  132767  000100  142671               BITB    #APTSPOOL,$ENVM     ;;SHOULD SPOOL MESSAGES?
036446  001425                               BEQ     3$                  ;;IF NOT: BR
036450  017600  000004                       MOV     @4(SP),R0           ;;GET MESSAGE ADDR.
036454  062766  000002  000004               ADD     #2,4(SP)            ;;BUMP RETURN ADDR.
036462  005767  142630              1$:      TST     $MSGTYPE            ;;SEE IF DONE W/ LAST XMISSION?
036466  001375                               BNE     1$                  ;;IF NOT: WAIT
036470  010067  142636                       MOV     R0,$MSGAD           ;;PUT ADDR IN MAILBOX
036474  105720                      2$:      TSTB    (R0)+               ;;FIND END OF MESSAGE
036476  001376                               BNE     2$
```

```
                                        SUB     $MSGAD,R0       ;;SUB START OF MESSAGE
                                        ASR     R0              ;;GET MESSAGE LNGTH IN WORDS
                                        MOV     R0,$MSGLGT      ;;PUT LENGTH IN MAILBOX
                                        MOV     #4,$MSGTYPE     ;;TELL APT TO TAKE MSG.
                                        SR      5$
                                3$:     MOV     @4(SP),4$       ;;PUT MSG ADDR IN JSR LINKAGE
                                        ADD     #2,4(SP)                ;;BUMP RETURN ADDRESS
                                        MOV     177776,-(SP)    ;;PUSH 177776 ON STACK
                                        JSR     PC,$TYPE        ;;CALL TYPE MACRO
                                4$:     .WORD   0
                                5$:
                                10$:    TSTB    $FFLG           ;;SHOULD REPORT FATAL ERROR?
                                        BEQ     12$             ;;IF NOT:  BR
                                        TST     $ENV            ;;RUNNING UNDER APT?
                                        BEQ     12$             ;;IF NOT:  BR
                                11$:    TST     $MSGTYPE        ;;FINISHED LAST MESSAGE?
                                        BNE     11$             ;;IF NOT:  WAIT
                                        MOV     @4(SP),$FATAL   ;;GET ERROR #
                                        ADD     #2,4(SP)                ;;BUMP RETURN ADDR.
                                        INC     $MSGTYPE        ;;TELL APT TO TAKE ERROR
                                12$:    CLRB    $FFLG           ;;CLEAR FATAL FLAG
                                        CLRB    $LFLG           ;;CLEAR LOG FLAG
                                        CLRB    $MFLG           ;;CLEAR MESSAGE FLAG
                                        MOV     (SP)+,R1        ;;POP STACK INTO R1
                                        MOV     (SP)+,R0        ;;POP STACK INTO R0
                                        RTS     PC              ;;RETURN
                                $MFLG:  .BYTE   0               ;;MESSG. FLAG
                                $LFLG:  .BYTE   0               ;;LOG FLAG
                                $FFLG:  .BYTE   0               ;;FATAL FLAG
                                        .EVEN
                                APTSIZE=200
                                APTENV=001
                                APTSPOOL=100
                                APTCSUP=040

                                .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.
                                $CKSWR: CMP     #SWREG,SWR      ;;IS THE SOFT-SWR SELECTED?
                                        BNE     15$             ;;BRANCH IF NO
                                        TSTB    @$TKS           ;;CHAR THERE?
                                        BPL     15$             ;;IF NO, DON'T WAIT AROUND
                                        MOVB    @$TKB,-(SP)     ;;SAVE THE CHAR
                                        BIC     #^C177,(SP)     ;;STRIP-OFF THE ASCII
                                        CMP     #7,(SP)+        ;;IS IT A CONTROL G?
                                        BNE     15$             ;;NO, RETURN TO USER
                                        CMPB    $AUTOB,#1       ;;ARE WE RUNNING IN AUTO-MODE?
                                        BEQ     15$             ;;BRANCH IF YES
```

```
036704  104401  037247              TYPE    .$CNTLG         ;;ECHO THE CONTROL-G  ^G
036710  104401  037254     $GTSWR:  TYPE    ,$MSWR          ;;TYPE CURRENT CONTENTS
036714  016746  141256              MOV     SWREG,-(SP)     ;;SAVE SWREG FOR TYPEOUT
036720  104402                      TYPOC                   ;;GO TYPE--OCTAL ASCII ALL DIGITS
036722  104401  037265              TYPE    ,$MNEW          ;;PROMPT FOR NEW SWR
036726  005046             19$:     CLR     -(SP)           ;;CLEAR COUNTER
036730  005046                      CLR     -(SP)           ;;THE NEW SWR
036732  105777  142206     7$:      TSTB    @$TKS           ;;CHAR THERE?
036736  100375                      BPL     7$              ;;IF NOT TRY AGAIN

036740  117746  142202              MOVB    @$TKB,-(SP)     ;;PICK UP CHAR
036744  042716  177600              BIC     #^C177,(SP)     ;;MAKE IT 7-BIT ASCII



036750  021627  000025     9$:      CMP     (SP),#25        ;;IS IT A CONTROL-U?
036754  001005                      BNE     10$             ;;BRANCH IF NOT
036756  104401  037242              TYPE    ,$CNTLU         ;;YES, ECHO CONTROL-U (^U)
036762  062706  000006     20$:     ADD     #6,SP           ;;IGNORE PREVIOUS INPUT
036766  000757                      BR      19$             ;;LET'S TRY IT AGAIN


036770  021627  000015     10$:     CMP     (SP),#15        ;;IS IT A <CR>?
036774  001022                      BNE     16$             ;;BRANCH IF NO
036776  005766  000004              TST     4(SP)           ;;YES, IS IT THE FIRST CHAR?
037002  001403                      BEQ     11$             ;;BRANCH IF YES
037004  016677  000002  142126      MOV     2(SP),@SWR      ;;SAVE NEW SWR
037012  062706  000006     11$:     ADD     #6,SP           ;;CLEAR UP STACK
037016  104401  001313     14$:     TYPE    ,$CRLF          ;;ECHO <CR> AND <LF>
037022  126727  142107  000001      CMPB    $INTAG,#1       ;;RE-ENABLE TTY KBD INTERRUPTS?
037030  001003                      BNE     15$             ;;BRANCH IF NOT
037032  012777  000100  142104      MOV     #100,@$TKS      ;;RE-ENABLE TTY KBD INTERRUPTS
037040  000002             15$:     RTI                     ;;RETURN
037042  004767  177026     16$:     JSR     PC,$TYPEC       ;;ECHO CHAR
037046  021627  000060              CMP     (SP),#60        ;;CHAR < 0?
037052  002420                      BLT     18$             ;;BRANCH IF YES
037054  021627  000067              CMP     (SP),#67        ;;CHAR > 7?
037060  003015                      BGT     18$             ;;BRANCH IF YES
037062  042726  000060              BIC     #60,(SP)+       ;;STRIP-OFF ASCII
037066  005766  000002              TST     2(SP)           ;;IS THIS THE FIRST CHAR
037072  001403                      BEQ     17$             ;;BRANCH IF YES
037074  006316                      ASL     (SP)            ;;NO, SHIFT PRESENT
037076  006316                      ASL     (SP)            ;;   CHAR OVER TO MAKE
037100  006316                      ASL     (SP)            ;;   ROOM FOR NEW ONE.
037102  005266  000002     17$:     INC     2(SP)           ;;KEEP COUNT OF CHAR
037106  056616  177776              BIS     -2(SP),(SP)     ;;SET IN NEW CHAR
037112  000707                      BR      7$              ;;GET THE NEXT ONE
037114  104401  001312     18$:     TYPE    ,$QUES          ;;TYPE ?<CR><LF>
037120  000720                      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
```

J11


```
                                        ;*      RETURN HERE             ;;CHARACTER IS ON THE STACK
                                        ;*                              ;;WITH PARITY BIT STRIPPED OFF
                                        ;
037122  011646                          $RDCHR: MOV     (SP),-(SP)      ;;PUSH DOWN THE PC
037124  016666  000004  000002                  MOV     4(SP),2(SP)     ;;SAVE THE PS
037132  105777  142006                  1$:     TSTB    @$TKS           ;;WAIT FOR
037136  100375                                  BPL     1$              ;;A CHARACTER
037140  117766  142002  000004                  MOVB    @$TKB,4(SP)     ;;READ THE TTY
037146  042766  177600  000004                  BIC     #^C<177>,4(SP)  ;;GET RID OF JUNK IF ANY
037154  026627  000004  000023                  CMP     4(SP),#23       ;;IS IT A CONTROL-S?
037162  001013                                  BNE     3$              ;;BRANCH IF NO
037164  105777  141754                  2$:     TSTB    @$TKS           ;;WAIT FOR A CHARACTER
037170  100375                                  BPL     2$              ;;LOOP UNTIL ITS THERE
037172  117746  141750                          MOVB    @$TKB,-(SP)     ;;GET CHARACTER
037176  042716  177600                          BIC     #^C177,(SP)     ;;MAKE IT 7-BIT ASCII
037202  022627  000021                          CMP     (SP)+,#21       ;;IS IT A CONTROL-Q?
037206  001366                                  BNE     2$              ;;IF NOT DISCARD IT
037210  000750                                  BR      1$              ;;YES, RESUME
037212  026627  000004  000140          3$:     CMP     4(SP),#140      ;;IS IT UPPER CASE?
037220  002407                                  BLT     4$              ;;BRANCH IF YES
037222  026627  000004  000175                  CMP     4(SP),#175      ;;IS IT A SPECIAL CHAR?
037230  003003                                  BGT     4$              ;;BRANCH IF YES
037232  042766  000040  000004                  BIC     #40,4(SP)       ;;MAKE IT UPPER CASE
037240  000002                          4$:     RTI                     ;;GO BACK TO USER
037242  052536  005015     000          $CNTLU: .ASCIZ  /^U/<15><12>    ;;CONTROL "U"
037247     136  006507  000012          $CNTLG: .ASCIZ  /^G/<15><12>    ;;CONTROL "G"
037254  005015  053523  020122          $MSWR:  .ASCIZ  <15><12>/SWR = /
037262  020075     000
037265     040  047040  053505          $MNEW:  .ASCIZ  /   NEW = /
037272  036440  000040

                                        .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.

037276  010046                          $TRAP:  MOV     R0,-(SP)        ;;SAVE R0
037300  016600  000002                          MOV     2(SP),R0        ;;GET TRAP ADDRESS
037304  005740                                  TST     -(R0)           ;;BACKUP BY 2
037306  111000                                  MOVB    (R0),R0         ;;GET RIGHT BYTE OF TRAP
037310  006300                                  ASL     R0              ;;POSITION FOR INDEXING
037312  016000  037332                          MOV     $TRPAD(R0),R0   ;;INDEX TO TABLE
037316  000200                                  RTS     R0              ;;GO TO ROUTINE

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

037320  011646                          $TRAP2: MOV     (SP),-(SP)      ;;MOVE THE PC DOWN
037322  016666  000004  000002                  MOV     4(SP),2(SP)     ;;MOVE THE PSW DOWN
037330  000002                                  RTI                     ;;RESTORE THE PSW

                                        .SBTTL  TRAP TABLE
```

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

                                        ;               ROUTINE
                                        ;               -------
037332  037320                          $TRPAD: .WORD   $TRAP2
037334  035662                                  $TYPE   ;;CALL=TYPE     TRAP+1(104401)  TTY TYPEOUT ROUTINE
037336  036170                                  $TYPOC  ;;CALL=TYPOC    TRAP+2(104402)  TYPE OCTAL NUMBER (WITH LEADING ZEROS)
037340  036144                                  $TYPOS  ;;CALL=TYPOS    TRAP+3(104403)  TYPE OCTAL NUMBER (NO LEADING ZEROS)
037342  036204                                  $TYPON  ;;CALL=TYPON    TRAP+4(104404)  TYPE OCTAL NUMBER (AS PER LAST CALL)

037344  036710                                  $GTSWR  ;;CALL=GTSWR    TRAP+5(104405)  GET SOFT-SWR SETTING

037346  036640                                  $CKSWR  ;;CALL=CKSWR    TRAP+6(104406)  TEST FOR CHANGE IN SOFT-SWR
037350  037122                                  $RDCHR  ;;CALL=RDCHR    TRAP+7(104407)  TTY TYPEIN CHARACTER ROUTINE
037352  035566                                  $SAVREG ;;CALL=SAVREG   TRAP+10(104410) SAVE R0-R5 ROUTINE
037354  035624                                  $RESREG ;;CALL=RESREG   TRAP+11(104411) RESTORE R0-R5 ROUTINE
037356  040274                                  .RSET   ;;CALL=RSETUP   TRAP+12(104412) ROUTINE TO RESET STACK AND FPS
037360  040266                                  .LPER   ;;CALL=LPERR    TRAP+13(104413) ROUTINE TO SET LOOP ON ERROR ADDRESS
        000030                          $TERM=.-$TRPAD

                                        .SBTTL  POWER DOWN AND UP ROUTINES

                                        ;;*****************************************************************
                                        ;POWER DOWN ROUTINE
037362  012737  037540  000024          $PWRDN: MOV     #$ILLUP,@#PWRVEC ;;SET FOR FAST UP
037370  012737  000340  000026                  MOV     #340,@#PWRVEC+2 ;;PRIO:7
037376  010046                                  MOV     R0,-(SP)        ;;PUSH R0 ON STACK
037400  010146                                  MOV     R1,-(SP)        ;;PUSH R1 ON STACK
037402  010246                                  MOV     R2,-(SP)        ;;PUSH R2 ON STACK
037404  010346                                  MOV     R3,-(SP)        ;;PUSH R3 ON STACK
037406  010446                                  MOV     R4,-(SP)        ;;PUSH R4 ON STACK
037410  010546                                  MOV     R5,-(SP)        ;;PUSH R5 ON STACK
037412  017746  141522                          MOV     @SWR,-(SP)      ;;PUSH @SWR ON STACK
037416  010667  000122                          MOV     SP,$SAVR6       ;;SAVE SP
037422  012737  037434  000024                  MOV     #$PWRUP,@#PWRVEC ;;SET UP VECTOR
037430  000000                                  HALT
037432  000776                                  BR      .-2             ;;HANG UP

                                        ;;*****************************************************************
                                        ;POWER UP ROUTINE
037434  012737  037540  000024          $PWRUP: MOV     #$ILLUP,@#PWRVEC ;;SET FOR FAST DOWN
037442  016706  000076                          MOV     $SAVR6,SP       ;;GET SP
037446  005067  000072                          CLR     $SAVR6          ;;WAIT LOOP FOR THE TTY
037452  005267  000066                  1$:     INC     $SAVR6          ;;WAIT FOR THE INC
037456  001375                                  BNE     1$              ;;OF  WORD
037460  012677  141454                          MOV     (SP)+,@SWR      ;;POP STACK INTO @SWR
037464  012605                                  MOV     (SP)+,R5        ;;POP STACK INTO R5
037466  012604                                  MOV     (SP)+,R4        ;;POP STACK INTO R4
037470  012603                                  MOV     (SP)+,R3        ;;POP STACK INTO R3
037472  012602                                  MOV     (SP)+,R2        ;;POP STACK INTO R2
037474  012601                                  MOV     (SP)+,R1        ;;POP STACK INTO R1
037476  012600                                  MOV     (SP)+,R0        ;;POP STACK INTO R0
037500  012737  037362  000024                  MOV     #$PWRDN,@#PWRVEC ;;SET UP THE POWER DOWN VECTOR
037506  012737  000340  000026                  MOV     #340,@#PWRVEC+2 ;;PRIO:7
```

```
 7794  037514  104401                    TYPE                            ;;REPORT THE POWER FAILURE
 7795  037516  040344            $PWRMG: .WORD   POWERM                  ;;POWER FAIL MESSAGE POINTER
 7796  037520  012716                    MOV     (PC)+,(SP)              ;;RESTART AT START
 7797  037522  003606            $PWRAD: .WORD   START                   ;;RESTART ADDRESS
 7798  037524  042766  000020  000002    BIC     #20,2(SP)               ;;CLEAR "T" BIT
 7799  037532  005067  175304            CLR     $TBIT                   ;;CLEAR THE "T" BIT FLAG
 7800  037536  000002                    RTI
 7801  037540  000000            $ILLUP: HALT                            ;;THE POWER UP SEQUENCE WAS STARTED
 7802  037542  000776                    BR      .-2                     ;; BEFORE THE POWER DOWN WAS COMPLETE
 7803  037544  000000            $SAVR6: 0                               ;;PUT THE SP HERE
 7804
 7805
 7806                                    .SBTTL  ERROR TYPE OUT ROUTINE
 7807                            ;;**************************************************************
 7808                            ;;**************************************************************
 7809                            ;*THIS ROUTINE IS CALLED TO TYPE AN ERROR MESSAGE WHICH IS INCLUDED
 7810                            ;*IN THE ERROR MESSAGE DATA TABLE. IT IS CALLED BY THE $ERROR ROUTINE
 7811                            ;*OR BY FIRST SETTING $ITEMB EQUAL TO THE ERROR TABLE ITEM TO BE PRINTED
 7812                            ;*OUT AND THEN EXECUTING A:
 7813                            ;*              JSR     PC,ERTYPE
 7814                            ;*
 7815  037546  104401            ERTYPE: TYPE                                    ;TYPE A CRLF
 7816  037550  001313                    .WORD   $CRLF
 7817  037552  113737  001102  001232    MOVB    @#$TSTNM,@#$TMPO
 7818  037560  042737  177400  001232    BIC     #177400,@#$TMPO
 7819  037566  013737  001116  001234    MOV     @#$ERRPC,@#$TMP1                ;GET PC OF CALL
 7820  037574  010046                    MOV     R0,-(SP)                        ;SAVE R0
 7821
 7822  037576  113700  001114            MOVB    @#$ITEMB,R0                     ;GET THE ITEM NUMBER.
 7823  037602  042700  177400            BIC     #177400,R0
 7824  037606  001005                    BNE     1$
 7825
 7826  037610  013746  001116            MOV     @#$ERRPC,-(SP)                  ;IF ZERO THEN JUST
 7827  037614  104402                    TYPOC                                   ;PRINT THE PC
 7828  037616  000137  040174            JMP     @#ERT5
 7829
 7830  037622  022700  000377    1$:     CMP     #377,R0
 7831  037626  001005                    BNE     20$
 7832  037630  016600  000004            MOV     4(SP),R0
 7833  037634  011000                    MOV     (R0),R0
 7834  037636  062700  000400            ADD     #400,R0
 7835  037642  005300            20$:    DEC     R0                              ;OTHERWISE MAKE R0 AN
 7836  037644  006300                    ASL     R0                              ;INDEX FOR THE TABLE.
 7837  037646  006300                    ASL     R0
 7838  037650  006300                    ASL     R0
 7839  037652  062700  001442            ADD     #$ERRTB,R0
 7840
 7841  037656  012037  037666            MOV     (R0)+,@#2$                      ;PICK UP THE ADDRESS
 7842  037662  001404                    BEQ     3$                              ;OF THE EM. ERROR MESSAGE
 7843  037664  104401                    TYPE
 7844  037666  000000            2$:     .WORD   0
 7845  037670  104401                    TYPE
 7846  037672  001313                    .WORD   $CRLF
 7847
 7848  037674  012037  037704    3$:     MOV     (R0)+,@#4$                      ;GET THE DH.DATA HEADER
 7849  037700  001404                    BEQ     5$
```

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

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

```
                        .SBTTL  FPP SPURIOUS TRAP TO 244 HANDLER
                ;;*******************************************************************
                ;;*******************************************************************
                ;*THIS ROUTINE HANDLES UNEXPECTED TRAPS TO THE FPP TRAP VECTOR AT 244.
                ;*THE LAST FPP INSTRUCTION EXECUTED AND ITS ADDRESS HAS BEEN RECORDED
                ;*THESE ALONG WITH THE FEC, FPS AND PC OF TRAP ARE REPORTED.
                ;*
040236  011637  001236  FPSPUR: MOV     (SP),@#$TMP2            ;SAVE PC OF TRAP.
040242  022626          1$:     CMP     (SP)+,(SP)+             ;RESTORE SP.
040244  170200                  STFPS   R0                      ;GET FPS
040246  010067  001240          MOV     R0,@#$TMP3
040252  170300                  STST    R0                      ;GET FEC
040254  010067  001242          MOV     R0,@#$TMP4
040260  104213                  ERROR   211
040262  104412                  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?).
040264  000137  034556          JMP     @#$EOP

                        .SBTTL  CPU SPURIOUS TRAP TO 4 HANDLER
                ;;*******************************************************************
                ;;*******************************************************************
                ;*THIS ROUTINE REPORTS UNEXPECTED CPU TRAPS TO VECTOR 4.
                ;*
040270  011637  001236  CPSPUR: MOV     (SP),@#$TMP2            ;SAVE PC OF TRAP.
040274  022626          1$:     CMP     (SP)+,(SP)+
040276  104213                  ERROR   212
040300  104412                  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?).
040302  000137  034556          JMP     @#$EOP

                        .SBTTL  CPU SPURIOUS TRAP TO 10 HANDLER
                ;;*******************************************************************
                ;;*******************************************************************
                ;*THIS ROUTINE REPORTS UNEXPECTED CPU TRAPS TO VECTOR 10.
                ;*
040306  011637  001236  CPTWO:  MOV     (SP),@#$TMP2            ;SAVE PC OF TRAP.
040312  022626          1$:     CMP     (SP)+,(SP)+
040314  104213                  ERROR   213
040316  104412                  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?).
040320  000137  034556          JMP     @#$EOP
```

```
                                        .SBTTL  SET LOOP ON ERROR ADDRESS ROUTINE
                                ;;*************************************************************
                                ;;*************************************************************
                                ;*
040266  011637  001110          .LPER:  MOV     (SP),@#$LPERR
040272  000002                          RTI

                                        .SBTTL  FLAG RESET AND CONSOLE TEST ROUTINE
                                ;;*************************************************************
                                ;;*************************************************************
                                ;*THIS ROUTINE WILL BE CALLED AT THE END OF EACH TEST TO
                                ;*RESET THE STACK, CLEAR THE FPS AND SEE IF THE USER HAS TYPED
                                ;* CONTROL G ON THE TERMINAL. IF THE USER HAS TYPED CONTROL G AND
                                ;*THERE IS NO PHYSICAL CONSOLE SWITCH REGISTER THEN THE CONTENTS
                                ;*OF THE SOFTWARE SWITCH REGISTER WILL BE TYPED IN OCTAL ON THE
                                ;*TELETYPE AND THE USER CAN MODIFY IT.
                                ;*
040274  023727  001140  177570  .RSET:  CMP     @#SWR,#177570           ;SEE IF THERE IS A PHYSICAL
                                                                        ;CONSOLE SWITCH REGISTER.
040302  001001                          BNE     1$                      ;BRANCH IF NO.
040304  104406                          CKSWR                           ;OTHERWISE TYPE THE CONTENTS
                                                                        ;OF THE PROGRAM VIRTUAL SWITCH REGISTER
                                                                        ;AND GIVE THE USER A CHANCE TO
                                                                        ;MODIFY IT.
040306  012737  040200  000244  1$:     MOV     #FPSPUR,@#FPVECT
040314  012737  040232  000004          MOV     #CPSPUR,@#ERRVECT
040322  012737  040250  000010          MOV     #CPTWO,@#10
040330  011600                          MOV     (SP),R0                 ;SAVE RETURN ADDRESS.
040332  012706  001100                  MOV     #STACK,SP               ;RESET THE STACK POINTER.
040336  005004                          CLR     R4                      ;CLEAR THE FPS.
040340  170104                          LDFPS   R4
040342  000110                          JMP     (R0)                    ;RETURN.

                                        .NLIST  BEX

                                ;SPECIAL MESSAGES:

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

D12


```
041004  052203  042510  042522  NOOP5:   .ASCIZ  <CRLF>'THEREBY EXECUTING A '
041032  020011  047111  052123  NOOP6:   .ASCIZ  <TAB>' INSTEAD OF A '
041052  020011  047111  052123  NOOP7:   .ASCIZ  <TAB>' INSTRUCTION.'<CRLF>
041072  020040  042524  052123  NOOP10:  .ASCII  '   TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'<TAB>
041133     107  052117  043040           .ASCIZ  'GOT FPS. EXPECTED FPS.'<CRLF>
041163     101  041040  042101  NOOP11:  .ASCIZ  'A BAD CONSTANT MAY HAVE BEEN USED.'<CRLF>
041227     011  042114  050106  LFPS1:   .ASCIZ  <TAB>'LDFPS'<TAB>'REG'
041242  046011  042104  024011  LD1:     .ASCIZ  <TAB>'LDD'<TAB>'.REG.,A'<TAB>'  FSRC '
041272  046011  042104  040411  LD2:     .ASCIZ  <TAB>'LDD'<TAB>'A,A'
041303     011  052123  050106  STFS1:   .ASCIZ  <TAB>'STFPS'<TAB>'REG'
041316  051411  042124  040411  ST1:     .ASCIZ  <TAB>'STD'<TAB>'A,.REG '
041333     011  052123  004504  ST2:     .ASCIZ  <TAB>'STD'<TAB>'A,A'
041344  041411  041506  000103  CFCC1:   .ASCIZ  <TAB>'CFCC'
041352  051411  052105  000106  SETF1:   .ASCIZ  <TAB>'SETF'
041360  051411  052105  000104  SETD1:   .ASCIZ  <TAB>'SETD'
041366  051411  052105  000111  SETI1:   .ASCIZ  <TAB>'SETI'
041374  051411  052105  000114  SETL1:   .ASCIZ  <TAB>'SETL'
041402  044411  046114  043505  ILL1:    .ASCIZ  <TAB>'ILLEGAL FPP INSTRUCTION'
041433     011  052123  052123  STST1:   .ASCIZ  <TAB>'STST'<TAB>'REG'
041445     011  046111  042514  ILL2:    .ASCIZ  <TAB>'ILLEGAL FPP INSTRUCTION (FID=1)'
041506  020040  042524  052123  ILLMS:   .ASCIZ  '   TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF TRAP.'<TAB>'FPS.'<CRLF>
041554  054105  042520  052103  MS1:     .ASCIZ  'EXPECTED '
041566  047507  020124     000  MS2:     .ASCIZ  'GOT '
041573     103  047117  042524  MS3:     .ASCIZ  'CONTENTS OF LOCATIONS '
041622  052040  051110  052517  MS4:     .ASCIZ  ' THROUGH '
041634  040506  046111  051125  MS5:     .ASCIZ  'FAILURE IN THE MICROPROGRAM FLOW.'
041676  047503  052116  047522  MS6:     .ASCIZ  'CONTROL WENT '
041714  051106  046517  051440  MS7:     .ASCIZ  'FROM STATE '
041730  052040  020117  052123  MS10:    .ASCIZ  ' TO STATE '
041743     102  052125  051440  MS11:    .ASCIZ  'BUT SHOULD HAVE GONE'
041770  047503  052116  047522  MS12:    .ASCIZ  'CONTROL FLOW SHOULD HAVE GONE'
042026  052502  020124  044504  MS13:    .ASCIZ  'BUT DID NOT.'
042043     040  052040  051505  MS14:    .ASCII  '   TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'<TAB>
042104  047507  020124  041520           .ASCIZ  'GOT PC.'<TAB>'EXPECTED PC.'
042131     111  051516  051124  MS15:    .ASCIZ  'INSTRUCTION TESTED: '
042156  047440  020122     000  MS16:    .ASCIZ  ' OR '
042163     124  051505  044524  MS17:    .ASCIZ  'TESTING ACCUMULATOR '
042210  042532  047522  000040  MNUM0:   .ASCIZ  'ZERO '
042216  047117  020105     000  MNUM1:   .ASCIZ  'ONE '
042223     124  047527  000040  MNUM2:   .ASCIZ  'TWO '
042230  044124  042522  020105  MNUM3:   .ASCIZ  'THREE '
042237     106  052517  020122  MNUM4:   .ASCIZ  'FOUR '
042245     106  053111  020105  MNUM5:   .ASCIZ  'FIVE '
042253     040  052040  051505  MS20:    .ASCIZ  '   TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
042314  040504  040524  024040  MS21:    .ASCIZ  'DATA (FLOATING POINT NUMBER): '
042353     114  043517  041511  MS22:    .ASCIZ  'LOGICAL AND OF FAILING '
042403     114  043517  041511  MS23:    .ASCIZ  'LOGICAL OR OF FAILING '
042432  020040  042524  052123  MS24:    .ASCII  '   TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERRORS.'<TAB>
042474  052516  041115  051105           .ASCIZ  'NUMBER OF ERRORS(OCTAL).'
042525     105  050130  041505  MS25:    .ASCIZ  'EXPECTED DATA IN '
042547     107  052117  042040  MS26:    .ASCIZ  'GOT DATA IN '
042564  040600  030103  020075  MS27:    .ASCIZ  <CRLF>'AC0= '
042573     200  041501  036461  MS30:    .ASCIZ  <CRLF>'AC1= '
042602  040600  031103  020075  MS31:    .ASCIZ  <CRLF>'AC2= '
042611     200  041501  036463  MS32:    .ASCIZ  <CRLF>'AC3= '
042620  040600  032103  020075  MS33:    .ASCIZ  <CRLF>'AC4= '
```

```
[illegible]    [illegible]  [illegible]  036465  MS34:   .ASCIZ  <CRLF>'ACS= '
[illegible]    [illegible]  [illegible]     000  MS35:   .ASCIZ  'SET '
[illegible]    [illegible]  [illegible]  051101  MS36:   .ASCIZ  'CLEAR '
[illegible]    [illegible]  [illegible]  042105  MS37:   .ASCIZ  'LOADED DATA: '
[illegible]    042522       [illegible]  042040  MS40:   .ASCIZ  'READ DATA: '
[illegible]    [illegible]  042520       052103  MS41S:  .ASCIZ  'EXPECTED DATA: '
[illegible]    [illegible]  040524       044440  MS41:   .ASCIZ  'DATA IN (R0) FSRC: '
[illegible]    [illegible]  040524       044440  MS42:   .ASCIZ  'DATA IN AC0: '
[illegible]    [illegible]  [illegible]  042522  MS43:   .ASCIZ  'GOT RESULT: '
[illegible]       [illegible]  [illegible]  [illegible]  MS44:   .ASCIZ  'EXPECTED RESULT: '


                                         ;ERROR MESSAGES:

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

F12


```
             045036                            EM32=EM31
 [illegible] 052123   020104   041501  EM33:    .ASCII  'STD AC0,(R0) FAILED.'<CRLF>
 [illegible] [illegible] 052125 052520          .ASCIZ  'OUTPUT BAD.'
 [illegible] [illegible] 042124 040440 EM34:    .ASCII  'STD AC0,(R0) FAILED IN THE FDST FLOWS.'
 [illegible]    200   044124   020105           .ASCIZ  <CRLF>'THE (BUT GR7) FORK FAILED.'
 [illegible]    114   042104   024040  EM35:    .ASCII  'LDD (R0),AC0 FAILED IN THE FSRC FLOWS.'
 [illegible]    200   044124   020105           .ASCIZ  <CRLF>'THE (BUT GR7) FORK FAILED.'
 [illegible]    123   042124   040440  EM36:    .ASCII  'STD AC0,(R0) FAILED IN THE FDST FLOWS.'
 [illegible]    200   044124   020105           .ASCIZ  <CRLF>'THE (BUT FD) FORK FAILED.'
 045452      042114   020104   051050  EM37:    .ASCII  'LDD (R0),AC0 FAILED IN THE FSRC FLOWS.'
 045520      052200   042510   024040           .ASCIZ  <CRLF>'THE (BUT FD) FORK FAILED.'
 [illegible]    114   042104   024040  EM40:    .ASCII  'LDD (R0),AC0 OR THE STD AC0,(R0) FAILED.'
 045622         200   040502   020104           .ASCIZ  <CRLF>'BAD DATA WAS DETECTED AFTER A SEQUENCE OF THE TWO INSTRUCTIONS.'
 045724      050106   020123   040502  EM41:    .ASCIZ  'FPS BAD AFTER EXECUTION OF: '
 045761                                EM42:
 045761         114   042104   024040           .ASCII  /LDD (R0),AC0 FAILED IN THE FSRC FLOWS./<CRLF>
 046030      044124   020105   041050           .ASCIZ  /THE (BUT FSRC) FORK FAILED. TRAPPED TO 4./
 046102                                EM43:
 046102      052123   020104   041501           .ASCII  /STD AC0,(R0) FAILED IN THE FDST FLOWS./<CRLF>
 046151         124   042510   024040           .ASCIZ  /THE (BUT FDST) FORK FAILED. TRAPPED TO 4./
 046223         106   050120   040440  EM44:    .ASCIZ  'FPP ACCUMULATORS DATA TEST FAILED.'
             046223                            EM45=EM44
 046266      050106   020120   041501  EM46:    .ASCIZ  'FPP ACCUMULATORS DUAL ADDRESSING TEST FAILED.'
 046344                                EM47:
 046344      042114   040440   030503           .ASCII  /LD AC1,AC0 FAILED IN THE FSRC FLOWS./
 046410      044124   020105   041050           .ASCIZ  /THE (BUT FSRC) FORK FAILED. TRAPPED TO 4./
 046462      042114   040440   030503  EM50:    .ASCII  'LD AC1,AC0 FAILED IN THE FSRC FLOWS.'
 046526      044124   020105   041050           .ASCIZ  'THE (BUT FD) FORK FAILED.'
 046560      042114   040440   030503  EM51:    .ASCIZ  'LD AC1,AC0 TRANSFERRED BAD DATA.'
 046621                                EM52:
 046621         114   042104   024040           .ASCII  /LDD (R0)+,AC0 FAILED IN THE FSRC FLOWS./
 046670      044124   020105   041050           .ASCIZ  /THE (BUT FSRC) FORK FAILED. TRAPPED TO 4./
 046742                                EM53:
 046742      042114   020104   051050           .ASCII  /LDD (R0)+,AC0 FAILED IN THE FSRC FLOWS.
 047011         200   030122   053440           .ASCII  <CRLF>'R0 WAS BAD.'<CRLF>
 047026      044505   044124   051105           .ASCII  'EITHER A BAD CONSTANT WAS GENERATED OR'<CRLF>
 047075         104   042111   047040           .ASCIZ  \DID NOT GO FROM STATE 627 TO 322.\
 047137                                EM54:
 047137         114   042104   024040           .ASCIZ  /LDD (R0)+,AC0 TRANSFERRED BAD DATA./
 047203                                EM55:
 047203         114   042104   026440           .ASCII  /LDD -(R0),AC0 FAILED IN THE FSRC FLOWS./
 047252      044124   020105   041050           .ASCIZ  /THE (BUT FSRC) FORK FAILED. TRAPPED TO 4.
 047324                                EM56:
 047324      042114   020104   024055           .ASCII  /LDD -(R0),AC0 FAILED IN THE FSRC FLOWS./
 047373         200   030122   053440           .ASCII  <CRLF>'R0 WAS BAD.'<CRLF>
 047410      044505   044124   051105           .ASCII  'EITHER A BAD CONSTANT WAS GENERATED OR'<CRLF>
 047457         104   042111   047040           .ASCIZ  \DID NOT GO FROM STATE 627 TO 324.\
 047521                                EM57:
 047521         114   042104   026440           .ASCIZ  /LDD -(R0),AC0 TRANSFERRED BAD DATA./
 047565                                EM60:
 047565         114   043104   024040           .ASCII  /LDF (R0)+,AC0 FAILED IN THE FSRC FLOWS./
 047634      051200   020060   040527           .ASCII  <CRLF>'R0 WAS BAD.'<CRLF>
 047651         105   052111   042510           .ASCII  'EITHER A BAD CONSTANT WAS GENERATED OR'<CRLF>
 047720      044504   020104   047516           .ASCIZ  \DID NOT GO FROM STATE 627 TO 322.\
 047762                                EM61:
 047762      042114   020106   051050           .ASCIZ  /LDF (R0)+,AC0 TRANSFERRED BAD DATA./
```

G12


```
[illegible]  042114  020106  051050  EM62:   .ASCII  'LDF (R0)+,AC0 FAILED IN THE FSRC FLOWS.'
[illegible]     200  044124  020105          .ASCII  <CRLF>'THE (BUT FD) FORK FAILED.'<CRLF>
[illegible]  042527  052116  043040          .ASCII  'WENT FROM STATE 441 TO 077.'<CRLF>
[illegible]  047111  052123  040505          .ASCIZ  'INSTEAD OF FROM 441 TO 076.'
[illegible]  042114  020104  047043  EM63:   .ASCII  'LDD #NUM,AC0 FAILED IN THE FSRC FLOWS.'
[illegible]6 052200  042510  024040          .ASCII  <CRLF>'THE (BUT GR7) FORK FAILED.'<CRLF>
[illegible]2 042527  052116  043040          .ASCII  'WENT FROM STATE 207 TO 174.'<CRLF>
[illegible]6 047111  052123  040505          .ASCIZ  'INSTEAD OF FROM 207 TO 176.'
[illegible]  042114  020104  047043  EM64:   .ASCII  'LDD #NUM,AC0 FAILED IN THE FSRC FLOWS.'
050460  040600  041040  042101          .ASCIZ  <CRLF>'A BAD CONSTANT WAS USED WHEN THE PC WAS INCREMENTED.'
        050412                  EM65=EM64
050546                          EM66:
050546  042114  020104  047043          .ASCIZ   LDD #NUM,AC0 TRANSFERRED BAD DATA./
[illegible]                     EM67:
050611     114  042104  040040          .ASCII  'LDD @(R0)+,AC0 FAILED IN THE FSRC FLOWS.'
050661     200  044124  020105          .ASCII  <CRLF>'THE (BUT FSRC) FORK FAILED. TRAPPED TO 4.'
050733     200  042527  052116          .ASCII  <CRLF>\WENT FROM STATE 627 TO EITHER 326 OR 326.
051005     200  047111  052123          .ASCIZ  <CRLF>\INSTEAD OF FROM 627 TO 323.\
051042                          EM70:
051042  042114  020104  024100          .ASCII  'LDD @(R0)+,AC0 FAILED IN THE FSRC FLOWS.'
051112  052200  042510  024040          .ASCIZ  <CRLF>'THE (BUT FSRC) FORK FAILED. TRAPPED TO 4.'
051165                          EM71:
051165     114  042104  040040          .ASCII  'LDD @(R0)+,AC0 FAILED IN THE FSRC FLOWS.'
051235     124  042510  024040          .ASCIZ  'THE (BUT FD) FORK FAILED.'
051267                          EM72:
051267     114  042104  040040          .ASCII  'LDD @(R0)+,AC0'<CRLF>
051306  040506  046111  042105          .ASCIZ  'FAILED TO INCREMENT R0 BY 2.'
051343                          EM73:
051343     114  042104  040040          .ASCIZ  'LDD @(R0)+,AC0 LOADED BAD DATA.'
051403                          EM74:
051403     114  042104  040040          .ASCII  'LDD @-(R0),AC0 FAILED IN THE FSRC FLOWS.'
051453     200  044124  020105          .ASCII  <CRLF>'THE (BUT FSRC) FORK FAILED. TRAPPED TO 4.'
051525     200  042527  052116          .ASCII  <CRLF>\WENT FROM STATE 627 TO EITHER 326 OR 326.
051577     200  047111  052123          .ASCIZ  <CRLF>\INSTEAD OF FROM 627 TO 325.\
051634                          EM75:
051634  042114  020104  026500          .ASCII  'LDD @-(R0),AC0 FAILED IN THE FSRC FLOWS.'
051704  052200  042510  024040          .ASCIZ  <CRLF>'THE (BUT FSRC) FORK FAILED. TRAPPED TO 4.'
051757                          EM76:
051757     114  042104  040040          .ASCII  'LDD @-(R0),AC0 FAILED IN THE FSRC FLOWS.'
052027     124  042510  024040          .ASCIZ  'THE (BUT FD) FORK FAILED.'
052061                          EM77:
052061     114  042104  040040          .ASCII  'LDD @-(R0),AC0'<CRLF>
052100  040506  046111  042105          .ASCIZ  'FAILED TO DECREMENT R0 BY 2.'
052135                          EM100:
052135     114  042104  040040          .ASCIZ  'LDD @-(R0),AC0 LOADED BAD DATA.'
052175                          EM101:
052175     114  042104  047040          .ASCII  'LDD NUM(R0),AC0 FAILED IN THE FSRC FLOWS.'
052246  052200  042510  024040          .ASCIZ  <CRLF>'THE (BUT FSRC) FORK FAILED. TRAPPED TO 4.'
052321                          EM102:
052321     114  042104  047040          .ASCII  'LDD NUM(R0),AC0'<CRLF>
052341     106  044501  042514          .ASCIZ  'FAILED TO AFFECT R0 BY 2.'
052373                          EM103:
052373     114  042104  047040          .ASCII  'LDD NUM(R0),AC0 FAILED IN THE FSRC FLOWS.'
052444  044124  020105  041050          .ASCIZ  'THE (BUT FD) FORK FAILED.'
052476                          EM104:
052476  042114  020104  052516          .ASCIZ  'LDD NUM(R0),AC0 LOADED BAD DATA.'
```

```
                                        EM105:
                                                .ASCII  'LDD @NUM(R0),AC0 FAILED IN THE FSRC FLOWS.'
                                                .ASCIZ  <CRLF>'THE (BUT FSRC) FORK FAILED. TRAPPED TO 4.'
                                        EM106:
                                                .ASCII  'LDD @NUM(R0),AC0'<CRLF>
                                                .ASCIZ  'FAILED TO AFFECT R0 BY 2.'
                                        EM107:
                                                .ASCII  'LDD @NUM(R0),AC0 FAILED IN THE FSRC FLOWS.'
                                                .ASCIZ  'THE (BUT FD) FORK FAILED.'
                                        EM110:
                                                .ASCIZ  'LDD @NUM(R0),AC0 LOADED BAD DATA.'
                                        EM111:
                                                .ASCII  /LDD AC7,AC0 FAILED TO TRAP TO 244./
                                                .ASCIZ  <CRLF>/AC7 IS AN ILLEGAL ACCUMULATOR./
                    053105                      EM112=EM111
                                        EM113:
                                                .ASCII  /LDD AC6,AC0 FAILED TO TRAP TO 244./
                                                .ASCIZ  <CRLF>/AC6 IS AN ILLEGAL ACCUMULATOR./
                    053207                      EM114=EM113
                    053106                      EM115=EM111
                    053207                      EM116=EM113
                                        EM117:
                                                .ASCII  'USE OF AN ILLEGAL ACCUMULATOR WITH FSRC MODE ZERO.'
                                                .ASCIZ  <CRLF>'TRAPPED BUT FAILED TO SET FPS CORRECTLY.'
                                        EM120:
                                                .ASCII  'USE OF AN ILLEGAL ACCUMULATOR WITH FSRC MODE ZERO.'
                                                .ASCIZ  <CRLF>'TRAPPED BUT FAILED TO SET FEC CORRECTLY.'
                                        EM121:  .ASCII  'ST AC0,AC1 FAILED IN THE FDST FLOWS.'
                                                .ASCIZ  <CRLF>'THE (BUT FDST) FORK FAILED. TRAPPED TO 4.'
                                        EM122:  .ASCII  'ST AC0,AC1 FAILED IN THE FDST FLOWS.'
                                                .ASCIZ  <CRLF>'THE (BUT FD) FORK FAILED.'
                                        EM123:  .ASCIZ  'ST AC0,AC1 TRANSFERRED BAD DATA.'
                                        EM124:
                                                .ASCII  'FPS BAD AFTER LDD (R0),AC0.'
                                                .ASCIZ  <CRLF>'THE (BUT EZBT Y8) FORK FAILED.
                                        EM125:
                                                .ASCII  'FPS BAD AFTER LDD (R0),AC0.'
                                                .ASCIZ  <CRLF>'THE (BUT ENBT) FORK FAILED.
                                        EM126:  .ASCII  'LDD (R0),AC0 TRAPPED TO 244.'
                                                .ASCII  ' FSRC= -0 AND FIUV= 0.'<CRLF>
                                                .ASCII  'THE (BUT FIUV) FORK FAILED.'
                                                .ASCII  <CRLF>'WENT FROM STATE 256 TO 354.'
                                                .ASCIZ  <CRLF>'INSTEAD OF FROM 256 TO 254.'
                                        EM127:  .ASCII  'LDD (R0),AC0 FAILED TO TRAP TO 244.'
                                                .ASCII  ' FSRC= -0, FIUV= 1.'
                                                .ASCII  <CRLF>'THE (BUT FIUV) FOR FAILED.'<CRLF
                                                .ASCII  'WENT FROM STATE 256 TO 254.'
                                                .ASCIZ  <CRLF>'INSTEAD OF FROM 256 THE 354.'
                                        EM130:  .ASCII  'LDD (R0),AC0 TRAPPED TO 244.'
                                                .ASCII  'FSRC= -0, FIUV= 1.'<CRLF>
                                                .ASCIZ  'BUT FEC WAS BAD.'
                                        EM131:
                                                .ASCIZ  /LDCFD (R0),AC0 LOADED BAD DATA.
                                        EM132:
                                                .ASCIZ  /LDCDF (R0),AC0 LOADED BAD DATA.
                                        EM133:
```

I12


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

J12


```
057300                                  EM164:
057300  044124  020105  050106                  .ASCIZ  \THE FPS WAS BAD AFTER ADDD (R0),AC0.\
057345                                  EM165:
057345     124  042510  043040                  .ASCIZ  \THE FPS WAS BAD AFTER ADDF (R0),AC0.\
057412                                  EM166:
057412  042101  042104  024040                  .ASCII  /ADDD (R0),AC0 PRODUCED A BAD RESULT./<CRLF>
057457     120  047522  040502                  .ASCIZ  'PROBABLE ERROR IN THE ALIGN FLOWS.'
057522                                  EM167:
057522  042101  042104  024040                  .ASCII  /ADDD (R0),AC0 FAILED IN THE ALIGN FLOWS./<CRLF>
057573     106  047514  020127                  .ASCII  \FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 111, TO 214.
057663     200  020101  040502                  .ASCII  <CRLF>\A BAD CONSTANT (NOT 57 DEC) \
057720  040527  020123  051525                  .ASCIZ  'WAS USED IN THE ALIGN ALGORITHM.'
057761                                  EM170:
057761     101  042104  020106                  .ASCII  /ADDF (R0),AC0 PRODUCED A BAD RESULT./<CRLF>
060026  051120  041117  041101                  .ASCIZ  'PROBABLE ERROR IN THE ALIGN FLOWS.'
060071                                  EM171:
060071     101  042104  020106                  .ASCII  /ADDF (R0),AC0 FAILED IN THE ALIGN FLOWS./<CRLF>
060142  046106  053517  042040                  .ASCII  \FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 111, TO 214.
060232  040600  041040  042101                  .ASCII  <CRLF>\A BAD CONSTANT (NOT 25 DEC) \
060267     127  051501  052440                  .ASCIZ  'WAS USED IN THE ALIGN ALGORITHM.'
060330                                  EM172:
060330  042101  042104  024040                  .ASCII  /ADDD (R0),AC0 FAILED IN THE ALIGN FLOWS./<CRLF>
060401     106  047514  020127                  .ASCII  \FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 011, TO 215.
060471     200  020101  040502                  .ASCII  <CRLF>\A BAD CONSTANT (NOT 57 DEC) \
060526  040527  020123  051525                  .ASCIZ  'WAS USED IN THE ALIGN ALGORITHM.'
060567                                  EM173:
060567     101  042104  020104                  .ASCII  /ADDD (R0),AC0 FAILED IN THE ALIGN FLOWS./<CRLF>
060640  046106  053517  042040                  .ASCII  \FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 011, TO 215.
060730  040600  041040  042101                  .ASCII  <CRLF>\A BAD CONSTANT (NOT 57 DEC) \
060765     127  051501  052440                  .ASCIZ  'WAS USED IN THE ALIGN ALGORITHM.'
061026                                  EM174:
061026  042101  043104  024040                  .ASCII  /ADDF (R0),AC0 FAILED IN THE ALIGN FLOWS./<CRLF>
061077     106  047514  020127                  .ASCII  \FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 011, TO 215.
061167     200  020101  040502                  .ASCII  <CRLF>\A BAD CONSTANT (NOT 25 DEC) \
061224  040527  020123  051525                  .ASCIZ  'WAS USED IN THE ALIGN ALGORITHM.'
061265                                  EM175:
061265     101  042104  020106                  .ASCII  /ADDF (R0),AC0 FAILED IN THE ALIGN FLOWS./<CRLF>
061336  046106  053517  042040                  .ASCII  \FLOW DID NOT FOLLOW THE PATH: STATE 476, TO 011, TO 215.
061426  040600  041040  042101                  .ASCII  <CRLF>\A BAD CONSTANT (NOT 25 DEC) \
061463     127  051501  052440                  .ASCIZ  'WAS USED IN THE ALIGN ALGORITHM.'
061524                                  EM176:
061524  042101  042104  024040                  .ASCII  'ADDD (R0),AC0 FAILED IN THE ADD-SUB FLOWS.'<CRLF>
061577     104  042111  047040                  .ASCIZ  \DID NOT TAKE THE PATH: STATE 216, TO 442, TO 500.\
061661                                  EM177:
061661     101  042104  020104                  .ASCII  'ADDD (R0),AC0 FAILED IN THE ADD-SUB FLOWS.'<CRLF>
061734  044504  020104  047516                  .ASCIZ  \DID NOT TAKE THE PATH: STATE 216, TO 042, TO 121.\
062016                                  EM200:
062016  042101  042104  024040                  .ASCII  'ADDD (R0),AC0 FAILED IN THE ADD-SUB FLOWS.'<CRLF>
062071     104  042111  047040                  .ASCIZ  \DID NOT TAKE THE PATH: STATE 216, TO 440, TO 121.\
062153                                  EM201:
062153     101  042104  020104                  .ASCII  'ADDD (R0),AC0 FAILED IN THE ADD-SUB FLOWS.'<CRLF>
062226  044504  020104  047516                  .ASCIZ  \DID NOT TAKE THE PATH: STATE 216, TO 440, TO 101.\
062310                                  EM202:
062310  042101  042104  024040                  .ASCII  'ADDD (R0),AC0 FAILED IN THE ADD-SUB FLOWS.'<CRLF>
062363     104  042111  047040                  .ASCIZ  \DID NOT TAKE THE PATH: STATE 216, TO 042, TO 101.\
062445                                  EM203:
```

```
062445     101  042104  020104          .ASCII  'ADDD (R0),AC0 FAILED IN THE ADD-SUB FLOWS.'<CRLF>
062520  044504  020104  047516          .ASCIZ  \DID NOT TAKE THE PATH: STATE 216, TO 440, TO 141.\
062602                          EM204:
062602  042101  042104  024040          .ASCII  'ADDD (R0),AC0 FAILED IN THE ADD-SUB FLOWS.'<CRLF>
062655     104  042111  047040          .ASCIZ  \DID NOT TAKE THE PATH: STATE 216, TO 042, TO 141.
062737                          EM205:
062737     124  042510  043040          .ASCIZ  \THE FPS WAS BAD AFTER SUBD (R0),AC0.\
063004                          EM206:
063004  052523  042102  024040          .ASCIZ  /SUBD (R0),AC0 PRODUCED A BAD RESULT./
063051     123  041125  020104  EM207:  .ASCII  'SUBD (R0),AC0 PRODUCED A BAD RESULT.'
063116     200  044124  020105          .ASCIZ  <CRLF>'THE XOR OF THE SIGN BIT FAILED IN STATE 024.'
063143     101  042104  020104  EM210:  .ASCIZ  'ADDD (R0),AC0 FAILED IN THE NORMALIZE FLOWS.'
        043233                  EM211=EM5
        043267                  EM212=EM6
        043321                  EM213=EM7

                                ;DATA HEADERS

063250  020040  042524  052123  DH1:    .ASCII  '   TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
063310  053411  047522  042524          .ASCIZ  <TAB>'WROTE.'<TAB>'READ.'<TAB>'EXPECTED.'
063340  020040  042524  052123  DH2:    .ASCII  '   TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
063400  047101  020104  040502          .ASCIZ  'AND BAD DATA.'<TAB>'OR BAD DATA.'
063433     040  052040  051505  DH3:    .ASCII  '   TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
063473     011  042522  042101          .ASCIZ  <TAB>'READ PSW.'<TAB>'EXPECTED PSW.'
063524  020040  042524  052123  DH4:    .ASCII  '   TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
063564  053411  047522  042524          .ASCIZ  <TAB>'WROTE FPS.'<TAB>'FPS AFTER CFCC.'
063620  020040  042524  052123  DH5:    .ASCIZ  '   TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF TRAP.'
        063620                  DH6=DH5
        063620                  DH7=DH5
        063620                  DH10=DH5
        063620                  DH11=DH5
        000000                  DH12=0
        000000                  DH13=0
        063620                  DH14=DH5
        063620                  DH15=DH5
063660  020040  042524  052123  DH16:   .ASCII  '   TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
063720  047411  020120  047503          .ASCIZ  <TAB>'OP CODE.   FPS.'
063740  020040  042524  052123  DH17:   .ASCII  '   TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
064000  043411  052117  043040          .ASCIZ  <TAB>'GOT FPS.'<TAB>'EXPECTED FPS.'
064030  020040  042524  052123  DH20:   .ASCII  '   TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF TRAP.'
064067     011  041520  047440          .ASCIZ  <TAB>'PC OF STST.'<TAB>'READ FEC.'
        063620                  DH21=DH5
064116  040506  046111  042105  DH22:   .ASCIZ  'FAILED TO CORRECTLY SET FPS.'
064153     106  044501  042514  DH23:   .ASCII  'FAILED TO CORRECTLY SET FEC TO 000002.'<CRLF>
064222  020040  042524  052123          .ASCII  '   TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
064262  050011  020103  043117          .ASCIZ  <TAB>'PC OF STST.'<TAB>'READ FEC.'
064311     124  040522  050120  DH24:   .ASCII  'TRAPPED TO 244. FLOW WENT FROM STATE 554 TO STATE 430.'
064377     200  047111  052123          .ASCIZ  <CRLF>'INSTEAD OF FROM STATE 554 TO STATE 432.'
064450  020040  042524  052123  DH25:   .ASCIZ  '   TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'<TAB>
064512  020040  042524  052123  DH26:   .ASCII  '   TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
064552  043411  052117  051040          .ASCIZ  <TAB>'GOT R0.'<TAB>'EXPECTED R0.'
        064512                  DH27=DH26
        000000                  DH30=0
        064512                  DH31=DH26
        064512                  DH32=DH26
064603  020040  042524  052123  DH33:   .ASCII  '   TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
```

L12


```
064640  051011  020060  052050           .ASCIZ  <TAB>'RO (TARGET LOCATIONS FOR OUTPUT).'
        064600                   DH34=DH33
        064600                   DH35=DH33
        064600                   DH36=DH33
        064600                   DH37=DH33
        064600                   DH40=DH33
        000000                   DH41=0
064703     040  052040  051505   DH42:   .ASCII  '  TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF TRAP.'
064742  051011  020060  052050           .ASCIZ  <TAB>'RO (TARGET LOCATIONS FOR OUTPUT).'
        064703                   DH43=DH42
        000000                   DH44=0
065005     105  051122  051117   DH45:   .ASCIZ  'ERROR SUMMARY.'
065024  020040  042524  052123   DH46:   .ASCIZ  '  TEST.'<TAB>'CALL AT PC.'
        065005                   DH47=DH45
065050  020040  042524  052123   DH50:   .ASCII  '  TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
065110  053411  052111  020110           .ASCIZ  <TAB>'WITH FD.'
        065050                   DH51=DH50
        063620                   DH52=DH5
        064512                   DH53=DH26
065122  020040  042524  052123   DH54:   .ASCIZ  '  TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
        063620                   DH55=DH5
        064512                   DH56=DH26
        065122                   DH57=DH54
        064512                   DH60=DH26
        065122                   DH61=DH54
        065122                   DH62=DH54
        065122                   DH63=DH54
065163     122  051505  046125   DH65:   .ASCII  'RESULTING IN AN ODD ADDRESS TRAP TO 4.'
065231     200                           .ASCII  <CRLF>
065232  020040  042524  052123   DH64:   .ASCII  '  TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
065272  043411  052117  050040           .ASCIZ  <TAB>'GOT PC.'<TAB>'EXPECTED PC.'
        065122                   DH66=DH54
        063620                   DH67=DH5
        063620                   DH70=DH5
        064450                   DH71=DH25
        064512                   DH72=DH26
        065122                   DH73=DH54
        063620                   DH74=DH5
        063620                   DH75=DH5
        064450                   DH76=DH25
        064512                   DH77=DH26
        065122                   DH100=DH54
        063620                   DH101=DH5
        064512                   DH102=DH26
        064450                   DH103=DH25
        065122                   DH104=DH54
        063620                   DH105=DH5
        064512                   DH106=DH26
        064450                   DH107=DH25
        065122                   DH110=DH54
        064450                   DH111=DH25
065320  044124  020105  041050   DH112:  .ASCII  'THE (BUT FSRC) FORK FAILED.'<CRLF>
065354  047503  052116  047522           .ASCII  'CONTROL WENT FROM STATE 762 TO STATE 627.'
065425     200  047111  052123           .ASCII  <CRLF>'INSTEAD OF FROM STATE 762 TO STATE 637.'<CRLF>
065476  020040  042524  052123           .ASCIZ  '  TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF ERROR.'
        064450                   DH113=DH25
```

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

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


                                        ;DATA FORMATS:

066611      004     000     005 DF1:    .BYTE   4,0,5,0,5,0,0,0
066621      004     000     005 DF2:    .BYTE   4,0,5,4,5,0,5,0
066631      004     000     005 DF3:    .BYTE   4,0.5,0,5,0,5,0
                066631                  DF4=DF3
066641      004     000     005 DF5:    .BYTE   4,0.5,0,5,0,5,11,5,0,5,0
                066641                  DF6=DF5
                066641                  DF7=DF5
                066641                  DF10=DF5
                066641                  DF11=DF5
066655      005     011     005 DF12:   .BYTE   5,11,5,5,5,4,5.4,5,5,4,5,4.5,11,5.11.5,5,4,0,5,0.5,0,0
066707      005     011     005 DF13:   .BYTE   5,11,5,5,5,4,0,5,0,5,0,0
                066641                  DF14=DF6
                066641                  DF15=DF6
066723      004     000     005 DF16:   .BYTE   4,0,5,0,5,0,0
                066631                  DF17=DF3
066732      004     000     005 DF20:   .BYTE   4,0.5,0,5,0,5,0
066742      004     000     005 DF21:   .BYTE   4,0,5,0
066746      005     005     004 DF22:   .BYTE   5,5,4,0,5,0,5,0,5,0
066760      004     000     005 DF23:   .BYTE   4,0.5,0,5.0,5,0
066770      005     004     000 DF24:   .BYTE   5,4,0,5.0,5,0
066777      004     000     005 DF25:   .BYTE   4,0,5,0,5,5,5.0,5,0,5,0,5.5.5.0,5,0.5.0
067023      004     000     005 DF26:   .BYTE   4,0.5,0,5,0,0,5,5,5,5,4.5,4,5,5,5.5.4,5,4
067050      004     000     005 DF27:   .BYTE   4,0,5,0,5,0,0,5,5,5,5,4,5.4.5.5
067070      005     011     005 DF30:   .BYTE   5,11.5,5,5,4,0,5,0,5,0,0
                067023                  DF31=DF26
                067050                  DF32=DF27
067104      004     000     005 DF33:   .BYTE   4,0,5,0,5,0,5,5,5,0,5,0,5,12,5,5,5,0,5,0,5.12
067132      004     000     005 DF34:   .BYTE   4,0,5,0,5,0,5,5,5,5,4,5,4,5,5,5,5,4,5,4
                067132                  DF35=DF34
                067132                  DF36=DF34
                067132                  DF37=DF34
067156      004     000     005 DF40:   .BYTE   4,0,5,0,5,0,5,5,5,0,5,0,5,3,5,5,5,0,5,0.5,3
067204      011     005     005 DF41:   .BYTE   11,5,5,5,4,0,5,0,5,0,5,0
067220      004     000     005 DF42:   .BYTE   4.0,5,0,5,0,5,5,5,5,4,5,4,11,4,5,5,5,5,4.5.4
                067220                  DF43=DF42
067246      005     011     005 DF44:   .BYTE   5,11,5,5,5,4,0,5,0,5,5,5,5,3,5,5,5,5.3
```

B13


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

C13


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

              067664                        .EVEN
                                            ;DATA TABLES:

      067664  001232   001234   040412      DT1:      .WORD    $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
      067700  001242   001244   000000                .WORD    $TMP4,$TMP5,0
      067706  001232   001234   040412      DT2:      .WORD    $TMP0,$TMP1,$TAB,AERFLG,$TAB,$TMP2,$TAB,$TMP3,0
      067730  001232   001234   040412      DT3:      .WORD    $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
      067744  040412   001242   000000                .WORD    $TAB,$TMP4,0
```

D13


```
         067730                   DT4=DT3
067752   001232  001234  040412   DT5:    .WORD   $TMP0,$TMP1,$TAB,$TMP2,LFIEX1,$TMP21,LFIEX2
067770   001272  040531  001240           .WORD   $TMP20,FPSMS,$TMP3,FECMS,$TMP4,0
070004   001232  001234  040412   DT6:    .WORD   $TMP0,$TMP1,$TAB,$TMP2,LFIEX1,$TMP21,LFIEX2,$TMP20,0
         070004                   DT7=DT6
         070004                   DT10=DT6
         070004                   DT11=DT6
070026   040606  001252  040613   DT12:   .WORD   $THE,$TMP10,NOOP1,NOOP15,NOOP2,$TMP5
070042   040755  001246  040767           .WORD   NOOP3,$TMP6,NOOP4,NOOP2,$TMP5,NOOP3,$TMP7,NOOP5,$TMP11
070064   041032  001252  041052           .WORD   NOOP6,$TMP10,NOOP7,NOOP10,$TMP0,$TMP1,$TAB,$TMP2
070104   040412  001240  001242           .WORD   $TAB,$TMP3,$TMP4,0
070114   040606  001252  040613   DT13:   .WORD   $THE,$TMP10,NOOP1,NOOP11,NOOP10,$TMP0,$TMP1,$TAB
070134   001236  040412  001240           .WORD   $TMP2,$TAB,$TMP3,$TMP4,0
         070004                   DT14=DT6
         070004                   DT15=DT6
070146   001232  001234  040412   DT16:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP5,$TMP3,0
         067730                   DT17=DT3
070166   001232  001234  040412   DT20:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
070202   040412  001242  000000           .WORD   $TAB,$TMP4,0
070210   001232  001234  040412   DT21:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,0
070222   063433  001313           DT22:   .WORD   DH3,$CRLF
070226   001232  001234  040412           .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
070242   040412  001242  000000           .WORD   $TAB,$TMP4,0
070250   001232  001234  040412   DT23:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
070264   040412  001242  000000           .WORD   $TAB,$TMP4,0
070272   041506                   DT24:   .WORD   ILLMS
070274   001232  001234  040412           .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
070310   040412  001242  000000           .WORD   $TAB,$TMP4,0
070316   001232  001234  040412   DT25:   .WORD   $TMP0,$TMP1,$TAB,$CRLF,MS1,MS3,$TMP3,MS4,$TMP4,$CRLF
070342   001242  001313  041566           .WORD   $TMP4,$CRLF,MS2,MS3,$TMP5,MS4,$TMP6,$CRLF,$TMP5,0
070366   001232  001234  040412   DT26:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3,$TMP4,$CRLF
070406   041676  001313  041714           .WORD   MS6,$CRLF,MS7,$TMP5,MS10,$TMP6,$CRLF
070424   041743  001313  041714           .WORD   MS11,$CRLF,MS7,$TMP5,MS10,$TMP7,0
070442   001232  001234  040412   DT27:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
070456   001242  001313  041770           .WORD   $TMP4,$CRLF,MS12,$CRLF,MS7,$TMP5,MS10,$TMP7,$CRLF,MS13,0
070504   042131  001272  001313   DT30:   .WORD   MS15,$TMP20,$CRLF,MS14,$CRLF
070516   001232  001234  040412           .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
070532   001242  000000                   .WORD   $TMP4,0
         070366                   DT31=DT26
         070442                   DT32=DT27
070536   001232  001234  040412   DT33:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
070552   001313  041554  041573           .WORD   $CRLF,MS1,MS3,$TMP4,MS4,$TMP5,$CRLF,$TMP6,$CRLF
070574   041566  041573  001242           .WORD   MS2,MS3,$TMP4,MS4,$TMP5,$CRLF,$TMP4,0
070614   001232  001234  040412   DT34:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
070630   001313  041676  001313           .WORD   $CRLF,MS6,$CRLF,MS7,$TMP5,MS10,$TMP6,$CRLF
070650   041743  001313  041714           .WORD   MS11,$CRLF,MS7,$TMP5,MS10,$TMP7,0
         070614                   DT35=DT34
         070614                   DT36=DT34
         070614                   DT37=DT34
         070536                   DT40=DT33
070666   001272  001313  063433   DT41:   .WORD   $TMP20,$CRLF,DH3,$CRLF
070676   001232  001234  040412           .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
070712   040412  001242  000000           .WORD   $TAB,$TMP4,0
070720   001232  001234  040412   DT42:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
070734   001313  041676  001313           .WORD   $CRLF,MS6,$CRLF,MS7,$TMP5,MS10,$TMP6,$TMP15,$TMP10
070756   001313  041743  001313           .WORD   $CRLF,MS11,$CRLF,MS7,$TMP5,MS10,$TMP7,0
```

```
                    070720                  DT43=DT42
000776  042163  001244  001313  DT44:   .WORD   MS17,$TMP5,$CRLF,MS20,$CRLF,$TMP0,$TMP1,$TAB,$TMP2
001020  001313  041554  042314          .WORD   $CRLF,MS1,MS21,$CRLF,$TMP3,$CRLF,MS2,MS21,$CRLF,$TMP4,0
001046  042163  001244  001313  DT45:   .WORD   MS17,$TMP5,$CRLF,MS24,$CRLF,$TMP0,$TMP1,$TAB,$TMP2,$TAB
001072  001246  001313  042353          .WORD   $TMP6,$CRLF,MS22,MS21,$CRLF,$TMP3,$CRLF
001116  042403  042314  001313          .WORD   MS23,MS21,$CRLF,$TMP4,0
001142  001232  001234  001313  DT46:   .WORD   $TMP0,$TMP1,$CRLF,MS25,MS30,$TMP2,MS31,$TMP3
001166  042611  001242  042620          .WORD   MS32,$TMP4,MS33,$TMP5,MS34,$TMP6,$CRLF,MS26
001214  042573  001250  042602          .WORD   MS30,$TMP7,MS31,$TMP10
001240  042611  001254  042620          .WORD   MS32,$TMP11,MS33,$TMP12,MS34,$TMP13,0
001266  001232  001234  040412  DT47:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$CRLF,MS12,MS7,$TMP3,MS10
001312  001242  001313  042026          .WORD   $TMP4,$CRLF,MS13,0
                    070614                  DT50=DT34
001242  001232  001234  040412  DT51:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
001256  001313  042525  042564          .WORD   $CRLF,MS25,MS27,$TMP4,$CRLF,MS26,MS27,$TMP5,0
                    071210                  DT52=DT47
001300  001232  001234  040412  DT53:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
001314  001242  000000          .WORD   $TMP4,0
001320  001232  001234  040412  DT54:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$CRLF,MS1,MS21,$CRLF,$TMP3
001342  001313  041566  042314          .WORD   $CRLF,MS2,MS21,$CRLF,$TMP3,0
                    071210                  DT55=DT47
                    071300                  DT56=DT53
                    071320                  DT57=DT54
                    071300                  DT60=DT53
                    071320                  DT61=DT54
                    071320                  DT62=DT54
                    071320                  DT63=DT54
001356  001232  001234  040412  DT64:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
001372  001242  000000          .WORD   $TMP4,0
                    071356                  DT65=DT64
                    071320                  DT66=DT54
                    070210                  DT67=DT21
001376  001232  001234  040412  DT70:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$CRLF,MS6,$CRLF,MS7,$TMP5
001420  041730  001246  001313          .WORD   MS10,$TMP6,$CRLF,MS11,$CRLF,MS7,$TMP5,MS10,$TMP7,0
                    071376                  DT71=DT70
001444  001232  001234  040412  DT72:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3,$TMP4,0
                    071320                  DT73=DT54
                    070210                  DT74=DT21
                    071376                  DT75=DT70
                    071376                  DT76=DT70
                    071444                  DT77=DT72
                    071320                  DT100=DT54
                    071376                  DT101=DT70
                    071376                  DT102=DT71
                    071376                  DT103=DT70
                    071320                  DT104=DT54
                    071376                  DT105=DT70
                    071444                  DT106=DT72
                    071376                  DT107=DT70
                    071320                  DT110=DT54
001464  001232  001234  040412  DT111:  .WORD   $TMP0,$TMP1,$TAB,$TMP2,0
                    071464                  DT112=DT111
                    071464                  DT113=DT111
                    071464                  DT114=DT111
                    071464                  DT115=DT111
                    071464                  DT116=DT111
```

F13


```
        [illegible]                  DT117=DT3
        [illegible]                  DT120=DT3
        [illegible]                  DT121=DT47
        [illegible]                  DT122=DT34
        [illegible]                  DT123=DT51
071476  001232  001234  040412       DT124:  .WORD   STMP0,STMP1,STAB,STMP2,STAB,STMP3,STMP4,SCRLF
071516  [illegible]  001313  041714          .WORD   MS6,SCRLF,MS7,STMP5,MS10,STMP6,SCRLF
071534  [illegible]  001313  041714          .WORD   MS11,SCRLF,MS7,STMP5,MS10,STMP7,SCRLF,MS37,SCRLF,STMP10,0
        [illegible]                  DT125=DT124
        [illegible]                  DT126=DT111
        [illegible]                  DT127=DT111
        [illegible]                  DT130=DT3
071562  [illegible]  001234  040412  DT131:  .WORD   STMP0,STMP1,STAB,STMP2,SCRLF,MS37,SCRLF,STMP3
071602  001313  042670  001313               .WORD   SCRLF,MS40,SCRLF,STMP4,SCRLF,MS415,SCRLF,STMP5,0
        071562                       DT132=DT131
071624  001232  001234  040412       DT133:  .WORD   STMP0,STMP1,STAB,STMP2,SCRLF,MS41,SCRLF,STMP3
071644  001313  042750  001313               .WORD   SCRLF,MS42,SCRLF,STMP4,SCRLF,MS43,SCRLF,STMP5
071664  001313  043003  001313               .WORD   SCRLF,MS44,SCRLF,STMP6,0
        071624                       DT134=DT133
        071624                       DT135=DT133
        071624                       DT136=DT133
071676  001232  001234  040412       DT137:  .WORD   STMP0,STMP1,STAB,STMP2,STMP10,STAB,STMP11,0
        071676                       DT140=DT137
        071676                       DT141=DT137
        071676                       DT142=DT137
        071624                       DT143=DT133
        071624                       DT144=DT133
        071676                       DT145=DT137
        071676                       DT146=DT137
        071624                       DT147=DT133
        071624                       DT150=DT133
        071676                       DT151=DT137
        071624                       DT152=DT133
        071676                       DT153=DT137
071716  001232  001234  040412       DT154:  .WORD   STMP0,STMP1,STAB,STMP2,0
        071716                       DT155=DT154
        071624                       DT156=DT133
        071624                       DT157=DT133
        071624                       DT160=DT133
        071624                       DT161=DT133
        071624                       DT162=DT133
        071624                       DT163=DT133
        067730                       DT164=DT3
        067730                       DT165=DT3
        071624                       DT166=DT133
        071624                       DT167=DT133
        071624                       DT170=DT133
        071624                       DT171=DT133
        071624                       DT172=DT133
        071624                       DT173=DT133
        071624                       DT174=DT133
        071624                       DT175=DT133
        071624                       DT176=DT133
        071624                       DT177=DT133
        071624                       DT200=DT133
        071624                       DT201=DT133
```

G13


```
                071624                           DT202=DT133
                071624                           DT203=DT133
                071624                           DT204=DT133
                067730                           DT205=DT3
                071624                           DT206=DT133
                071624                           DT207=DT133
                071624                           DT210=DT133
        071730  001232   001234   040412   DT211:   .WORD    $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3,0
        071746  001232   001234   040412   DT212:   .WORD    $TMP0,$TMP1,$TAB,$TMP2,0
                071746                           DT213=DT212


                                                 ;12345
                000001                                   .END
```

H13

[illegible]	[illegible]	ADDWO =	000000	AC5	004476	BB30	030650	[illegible]	[illegible]
[illegible]	[illegible]	ADDW1 =	000000	A1	004350	BB31	030650	[illegible]	[illegible]
[illegible]	[illegible]	ADDW10=	000000	A11	004350	BB32	030672	[illegible]	[illegible]
[illegible]	[illegible]	ADDW11=	000000	A12	004364	BB33	030722	[illegible]	[illegible]
[illegible]	[illegible]	ADDW12=	000000	A2	004422	BB34	030744	[illegible]	[illegible]
[illegible]	[illegible]	ADDW13=	000000	A3	004426	BB35	030754	[illegible]	[illegible]
[illegible]	[illegible]	ADDW14=	000000	A4	004474	BB4	030210	[illegible]	[illegible]
[illegible]	[illegible]	ADDW15=	000000	A5	004500	BB5	030220	[illegible]	[illegible]
[illegible]	[illegible]	ADDW2 =	000000	A6	004526	BB6	030232	[illegible]	[illegible]
[illegible]	[illegible]	ADDW3 =	000000	A7	004552	BB7	030262	[illegible]	[illegible]
[illegible]	025764	ADDW4 =	000000	BBDATO	031300	BDONE	004772	[illegible]	[illegible]
[illegible]	026246	ADDW5 =	000000	BBDONE	031430	BERR	004726	[illegible]	[illegible]
[illegible]	026256	ADDW6 =	000000	BBERO	030766	BERR1	004756	[illegible]	[illegible]
[illegible]	026266	ADDW7 =	000000	BBER1	031026	BIT0 =	000001	[illegible]	[illegible]
[illegible]	026276	ADDW8 =	000000	BBER10	031006	BIT00 =	000001	[illegible]	[illegible]
[illegible]	026306	ADDW9 =	000000	BBER11	031042	BIT01 =	000002	[illegible]	[illegible]
[illegible]	026316	ADEVCT=	000000	BBER2	031064	BIT02 =	000004	[illegible]	030024
[illegible]	026326	ADEVM =	000000	BBER3	031102	BIT03 =	000010	CCP4	030034
[illegible]	025310	ADONE	004670	BBER4	031136	BIT04 =	000020	CCP5	030044
[illegible]	025424	AENV =	000000	BBER40	031152	BIT05 =	000040	CCP6	030054
[illegible]	025444	AENVM =	000000	BBER5	031172	BIT06 =	000100	CCP7	030064
[illegible]	025456	AERFLG	004560	BBER6	031210	BIT07 =	000200	CC1	026342
[illegible]	025474	AERR1	004562	BBER7	031244	BIT08 =	000400	CC10	026556
[illegible]	025514	AERR2	004574	BBER8	031262	BIT09 =	001000	CC11	026566
[illegible]	025534	AERR3	004626	BBPATO	031310	BIT1 =	000002	CC12	026574
[illegible]	025542	AFATAL=	000000	BBPAT1	031320	BIT10 =	002000	CC13	026606
[illegible]	025546	AMADR1=	000000	BBPAT2	031330	BIT11 =	004000	CC14	026636
[illegible]	025346	AMADR2=	000000	BBPAT3	031340	BIT12 =	010000	CC15	026662
[illegible]	025552	AMADR3=	000000	BBPAT4	031350	BIT13 =	020000	CC16	026702
[illegible]	025554	AMADR4=	000000	BBPAT5	031360	BIT14 =	040000	CC17	026712
[illegible]	025606	AMAMS1=	000000	BBPAT6	031370	BIT15 =	100000	CC18	026720
[illegible]	025610	AMAMS2=	000000	BBP10	031410	BIT2 =	000004	CC19	026732
[illegible]	025630	AMAMS3=	000000	BBP11	031420	BIT3 =	000010	CC2	026372
[illegible]	025650	AMAMS4=	000000	BBP7	031400	BIT4 =	000020	CC20	026762
[illegible]	025656	AMSGAD=	000000	BB1	030130	BIT5 =	000040	CC21	027006
[illegible]	025662	AMSGLG=	000000	BB10	030304	BIT6 =	000100	CC22	027026
[illegible]	025350	AMSGTY=	000000	BB11	030324	BIT7 =	000200	CC23	027036
[illegible]	025366	AMTYP1=	000000	BB12	030334	BIT8 =	000400	CC24	027044
[illegible]	025406	AMTYP2=	000000	BB13	030342	BIT9 =	001000	CC25	027056
[illegible]	025414	AMTYP3=	000000	BB14	030354	BPTVEC=	000014	CC26	027106
[illegible]	025422	AMTYP4=	000000	BB15	030404	B1	004702	CC27	027130
[illegible]BASE =	000000	APASS =	000000	BB16	030430	B2	004704	CC28	027150
[illegible]DW1 =	000000	APRIOR=	000000	BB17	030440	B3	004722	CC29	027160
[illegible]DW2 =	000000	APTCSU=	000040	BB2	030166	CCDATO	027764	CC3	026414
[illegible]PLOP=	000000	APTENV=	000001	BB20	030452	CCDONE	030124	CC30	027166
[illegible] =%	000000	APTSIZ=	000200	BB21	030502	CCERO	027326	CC31	027200
[illegible] =%	000001	APTSPO=	000100	BB22	030526	CCER1	027362	CC32	027230
[illegible] =%	000002	ASWREG=	000000	BB23	030546	CCER10	027674	CC33	027252
[illegible] =%	000003	ATESTN=	000000	BB24	030556	CCER11	027712	CC34	027272
[illegible] =%	000004	AUNIT =	000000	BB25	030564	CCER12	027730	CC35	027302
[illegible] =%	000005	AUSWR =	000000	BB26	030576	CCER13	027746	CC36	027310
[illegible] =%	000006	AVECT1=	000000	BB27	030626	CCER2	027420	[illegible]	027332
[illegible] =%	000007	AVECT2=	000000	BB3	030172	CCER22	027434	CC4	026434

Symbol		Value
[illegible]		[illegible]
[illegible]		[illegible]452
[illegible]		[illegible]
[illegible]		[illegible]
[illegible]		026536
[illegible]NE		[illegible]
[illegible]ERR1		[illegible]
[illegible]ERR2		[illegible]
[illegible]ERR3		[illegible]
[illegible]		[illegible]
[illegible]		041344
[illegible]	=	034436
[illegible]	=	[illegible]
[illegible]		[illegible]
[illegible]		040232
[illegible]		040250
[illegible]	=	000015
[illegible]	=	000200
[illegible]		005206
[illegible]		005034
[illegible]		005054
[illegible]		005070
[illegible]		005112
[illegible]		005142
[illegible]		005164
[illegible]		005176
[illegible]		005220
[illegible]		005246
[illegible]		005266
[illegible]		005302
[illegible]		005324
[illegible]		005354
[illegible]		005376
[illegible]BS		005410
[illegible]CAT[illegible]		033274
[illegible]ONE		033424
[illegible]CERO		032506
[illegible]CER1		032524
[illegible]CER10		033142
[illegible]CER11		033200
[illegible]CER12		033236
[illegible]CER2		032562
[illegible]CER3		032620
[illegible]CER4		032656
[illegible]CER5		032714
[illegible]CER6		032752
[illegible]CER7		033010
[illegible]CER8		033046
[illegible]CER9		033104
[illegible]	=	177570
[illegible]E		006140
[illegible]P0		033304
[illegible]P1		033314
DDP2		033324
DDP3		033334
DDP4		033344
DDP5		033354
DDP6		033364
DDP7		033374
DDP8		033404
DDP9		033414
DD1		031434
DD10		031634
DD11		031644
DD12		031662
DD13		031712
DD14		031734
DD15		031744
DD16		031762
DD17		032012
DD18		032034
DD19		032054
DD2		031464
DD20		032064
DD21		032072
DD22		032104
DD23		032134
DD24		032156
DD25		032176
DD26		032206
DD27		032214
DD3		031506
DD30		032226
DD31		032256
DD32		032300
DD33		032320
DD34		032330
DD35		032336
DD36		032354
DD37		032404
DD38		032426
DD39		032446
DD4		031526
DD40		032456
DD41		032464
DD42		032502
DD5		031536
DD6		031544
DD7		031562
DD8		031612
DERR1		006020
DERR2		006114
DF1		066611
DF10	=	066641
DF100	=	067436
DF101	=	067463
DF102	=	067505
DF103	=	067463
DF104	=	067436
DF105	=	067463
DF106	=	067505
DF107	=	067463
DF11	=	066641
DF110	=	067436
DF111		067514
DF112	=	067514
DF113	=	067514
DF114	=	067514
DF115	=	067514
DF116	=	067514
DF117	=	066631
DF12		065655
DF120	=	066631
DF121	=	067350
DF122	=	067365
DF123	=	067411
DF124		067520
DF125	=	067520
DF126	=	067514
DF127	=	067514
DF13		066707
DF130	=	066631
DF131		067551
DF132	=	067551
DF133		067571
DF134	=	067571
DF135	=	067571
DF136	=	067571
DF137		067615
DF14	=	066641
DF140	=	067615
DF141	=	067615
DF142	=	067615
DF143	=	067571
DF144	=	067571
DF145	=	067615
DF146	=	067615
DF147	=	067571
DF15	=	066641
DF150	=	067571
DF151	=	067615
DF152	=	067571
DF153	=	067615
DF154		067625
DF155	=	067625
DF156	=	067571
DF157	=	067571
DF16		066723
DF160	=	067571
DF161		0676[illegible]
DF162	=	[illegible]
DF163	=	067631
DF164	=	066631
DF165	=	06663[illegible]
DF166	=	[illegible]
DF167	=	067571
DF17	=	066631
DF170	=	067631
DF171	=	067631
DF172	=	067571
DF173	=	067571
DF174	=	067631
DF175	=	067631
DF176	=	067571
DF177	=	067571
DF2		066621
DF20		066732
DF200	=	067571
DF201	=	067571
DF202	=	067571
DF203	=	067571
DF204	=	067571
DF205	=	066631
DF206	=	067571
DF207	=	067571
DF21		066742
DF210	=	067571
DF211		067655
DF212	=	067655
DF213	=	067655
DF22		066746
DF23		066760
DF24		066770
DF25		066777
DF26		067023
DF27		067050
DF3		066631
DF30		067070
DF31	=	067023
DF32	=	067050
DF33		067104
DF34		067132
DF35	=	067132
DF36	=	067132
DF37	=	067132
DF4	=	066631
DF40		067156
DF41		067204
DF42		067220
DF43	=	067220
DF44		067246
DF45		067271
[illegible]		[illegible]
[illegible]		[illegible]
[illegible]		[illegible]
[illegible]		[illegible]
[illegible]		[illegible]
[illegible]	=	[illegible]
[illegible]		[illegible]
[illegible]		[illegible]
[illegible]		[illegible]
[illegible]	=	[illegible]
[illegible]	=	[illegible]
[illegible]	=	067436
[illegible]	=	[illegible]
[illegible]	=	[illegible]
DF6[illegible]	=	[illegible]
DF60	=	[illegible]
DF61	=	067436
DF62	=	067436
DF63	=	067436
DF64		[illegible]
DF65	=	067436
DF66	=	067436
DF67	=	066742
DF7	=	[illegible]
DF70		067463
DF71	=	067463
DF72		06750[illegible]
DF73	=	067436
DF74	=	066742
DF75	=	067463
DF76	=	067463
DF77	=	067505
DH1		063250
DH10	=	063620
DH100	=	065122
DH101	=	063620
DH102	=	064512
DH103	=	064450
DH104	=	065122
DH105	=	063620
DH106	=	064512
DH107	=	064450
DH11	=	063620
DH110	=	065122
DH111	=	064450
DH112		065320
DH113	=	064450
DH114	=	065320
DH115		065537
DH116	=	065537
DH117	=	063740
DH12	=	000000
DH120		066023
DH121	=	063620
DH122	=	065050
DH123	=	065050

SYMBOL TABLE

```
DH    = 063620
DH70  = 063620
DH71  = 064450
DH72  = 064512
DH73  = 065122
DH74  = 063620
DH75  = 063620
DH76  = 064450
DH77  = 064512
DISPLA  001142
DISPRE  000174
DPAT3   016446
DSWR    = 177570
DT1     067664
DT10    = 070004
DT100   = 071320
DT101   = 071376
DT102   = 071376
DT103   = 071376
DT104   = 071320
DT105   = 071376
DT106   = 071444
DT107   = 071376
DT11    = 070004
DT110   = 071320
DT111   = 071464
DT112   = 071464
DT113   = 071464
DT114   = 071464
DT115   = 071464
DT116   = 071464
DT117   = 067730
DT12    = 070026
DT120   = 067730
DT121   = 071210
DT122   = 070614
DT123   = 071242
DT124   = 071476
DT125   = 071476
DT126   = 071464
DT127   = 071464
DT13    = 070114
DT130   = 067730
DT131   = 071562
DT132   = 071562
DT133   = 071624
DT134   = 071624
DT135   = 071624
DT136   = 071624
DT137   = 071676
DT14    = 070004
DT140   = 071676
DT141   = 071676
DT142   = 071676
DT143   = 071624
DT144   = 071624
DT145   = 071676
DT146   = 071624
DT147   = 071624
DT15    = 070004
DT150   = 071624
DT151   = 071676
DT152   = 071624
DT153   = 071676
DT154   = 071716
DT155   = 071716
DT156   = 071624
DT157   = 071624
DT16    = 070146
DT160   = 071624
DT161   = 071624
DT162   = 071624
DT163   = 071624
DT164   = 067730
DT165   = 067730
DT166   = 071624
DT167   = 071624
DT17    = 067730
DT170   = 071624
DT171   = 071624
DT172   = 071624
DT173   = 071624
DT174   = 071624
DT175   = 071624
DT176   = 071624
DT177   = 071624
DT2     067706
DT20    070166
DT200   = 071624
DT201   = 071624
DT202   = 071624
DT203   = 071624
DT204   = 071624
DT205   = 067730
DT206   = 071624
DT207   = 071624
DT21    070210
DT210   = 071624
DT211   071730
DT212   071746
DT213   = 071746
DT22    070222
DT23    070250
DT24    070272
DT25    070316
DT26    070366
```

K13


```
D7        006014
D8        006064
D9        006076
EDONE     006300
EEDATO    034116
EEDONE    034200
EEER0     033710
EEER1     033726
EEER2     033764
EEER3     034022
EEER4     034060
EEP0      034126
EEP1      034140
EEP2      034150
EEP3      034160
EEP4      034170
EERR0     006216
EERR1     006234
EERR2     006250
EE1       033430
EE10      033650
EE11      033660
EE12      033666
EE13      033704
EE2       033460
EE3       033502
EE4       033522
EE5       033532
EE6       033540
EE7       033556
EE8       033606
EE9       033630
EMTVEC=   000030
EM1       043025
EM10   =  043233
EM100     052135
EM101     052175
EM102     052321
EM103     052373
EM104     052476
EM105     052537
EM106     052664
EM107     052737
EM11      043354
EM110     053043
EM111     053105
EM112  =  053105
EM113     053207
EM114  =  053207
EM115  =  053105
EM116  =  053207
EM117     053311
EM12   =  000000

EM120     053445
EM121     053601
EM122     053720
EM123     054017
EM124     054060
EM125     054153
EM126     054243
EM127     054452
EM13   =  000000
EM130     054665
EM131     054765
EM132     055025
EM133     055065
EM134     055124
EM135     055163
EM136     055222
EM137  =  055065
EM14      043435
EM140  =  055124
EM141  =  055163
EM142  =  055222
EM143     055261
EM144     055314
EM145  =  055261
EM146  =  055314
EM147     055347
EM15      043560
EM150  =  055347
EM151  =  055347
EM152     055401
EM153  =  055401
EM154     055433
EM155     055665
EM156     056120
EM157     056335
EM16      043703
EM160     056554
EM161     056761
EM162     057166
EM163     057233
EM164     057300
EM165     057345
EM166     057412
EM167     057522
EM17      043754
EM170     057761
EM171     060071
EM172     060330
EM173     060567
EM174     061026
EM175     061265
EM176     061524
EM177     061661

EM2       043062
EM20      044207
EM200     062016
EM201     062153
EM202     062310
EM203     062445
EM204     062602
EM205     062737
EM206     063004
EM207     063051
EM21      044365
EM210     063173
EM211  =  043233
EM212  =  043267
EM213  =  043321
EM22      044516
EM23   =  044516
EM24   =  044516
EM25      044603
EM26      044716
EM27   =  044716
EM3       043126
EM30      044764
EM31      045036
EM32   =  045036
EM33      045104
EM34      045145
EM35      045247
EM36      045351
EM37      045452
EM4       043173
EM40      045553
EM41      045724
EM42      045761
EM43      046102
EM44      046223
EM45   =  046223
EM46      046266
EM47      046344
EM5       043233
EM50      046462
EM51      046560
EM52      046621
EM53      046742
EM54      047137
EM55      047203
EM56      047324
EM57      047521
EM6       043267
EM60      047565
EM61      047762
EM62      050026
EM63      050220

EM64      050412
EM65   =  050412
EM66      050546
EM67      050611
EM7       043321
EM70      051042
EM71      051165
EM72      051267
EM73      051343
EM74      051403
EM75      051634
EM76      051757
EM77      052061
ERM10     035564
ERRVEC=   000004
ERTYPE    037546
ERT1      037730
ERT2      040146
ERT3      040152
ERT4      040162
ERT5      040174
E1        006154
E2        006170
E3        006170
E4        006172
FDATI0    010110
FDATI1    010112
FDATI2    010114
FDATI3    010116
FDATI4    010120
FDATI5    010122
FDATI6    010124
FDATI7    010126
FDATO0    010132
FDATO1    010134
FDATO2    010136
FDATO3    010140
FDATO4    010142
FDATO5    010144
FDATO6    010146
FDATO7    010150
FDONE     010174
FECMS     040575
FERR0     006702
FERR1     006740
FERR10    007520
FERR11    007554
FERR2     007036
FERR20    007574
FERR21    007712
FERR25    007742
FERR26    010060
FERR3     007076

FERR4     [illegible]
FERR5     [illegible]
FERR6     007230
FERR7     007364
FER2      007342
FFDATO    034476
FFDONE    034556
FFER0     034366
FFER1     034402
FFER2     034440
FFP0      034506
FFP1      034516
FFP2      034526
FFP3      034536
FFP4      034546
FF1       034204
FF10      034354
FF11      034364
FF2       034234
FF3       034256
FF4       034264
FF5       034274
FF6       034324
FF7       034346
FPSMS     040531
FPSPUR    040200
FPVECT=   000244
FXDATO    010154
FXDAT1    010156
FXDAT2    010160
FXDAT3    010162
FXDAT4    010164
FXDAT5    010166
FXDAT6    010170
FXDAT7    010172
F1        006304
F10       006516
F11       006520
F12       006536
F13       006570
F135      006550
F14       006600
F15       006610
F16       006620
F17       006630
F2        006336
F20       006540
F21       006650
F22       006660
F23       006676
F3        006356
F4        006360
F5        006376
```

SYMBOL TABLE

[illegible]	[illegible]	G31	013130	IDATI1	011030	JDATI2	016132	LDAT00	016760
[illegible]	[illegible]	G32	013162	IDATI2	011032	JDATI3	016134	LDAT01	016762
[illegible]	[illegible]	G33	013164	IDATI3	011034	JDATO0	016136	LDAT02	016766
[illegible]	[illegible]	G34	013252	IDATO0	011016	JDATO	016146	LDAT03	016770
[illegible]	[illegible]	G35	013304	IDATO1	011020	JDATO1	016140	LDONE	016772
[illegible]	[illegible]	G36	013306	IDATO2	011022	JDATO2	016142	LD1	041242
[illegible]	[illegible]	G37	013374	IDATO3	011024	JDATO3	016144	LD2	041272
[illegible]	[illegible]	G4	012032	IDONE	011036	JDAT1	016150	LERR1	016604
[illegible]	[illegible]	G40	013426	IERR0	010576	JDAT2	016152	LERR2	016656
[illegible]	[illegible]	G41	013430	IERR1	010660	JDAT3	016154	LERR3	016630
[illegible]	[illegible]	G42	013516	IERR2	010700	JDONE	016156	LF =	000012
[illegible]	[illegible]	G43	013550	IERR25	010722	JERR0	015776	LFIEX1	040417
[illegible]	[illegible]	G44	013552	IERR3	010752	JERR1	016044	LFIEX2	040467
[illegible]	[illegible]	G5	012064	IERR4	010726	JERR2	016070	LFPS1	041227
[illegible]	[illegible]	G6	012066	ILLMS	041506	J1	015670	LOOP	004304
[illegible]	[illegible]	G7	012154	ILL1	041402	J10	016016	LPAT10	016726
[illegible]	[illegible]	HADR	014732	ILL2	041445	J2	015714	LPAT11	016730
[illegible]	[illegible]	HA1R	015006	IOTVEC=	000020	J3	015716	LPAT12	016732
[illegible]	[illegible]	HA1W	014736	IPAT10	010776	J4	015720	LPAT13	016734
[illegible]	[illegible]	HA2R	015016	IPAT11	011000	J5	015754	LPAT20	016736
[illegible]	[illegible]	HA2W	014746	IPAT12	011002	J6	015762	LPAT21	016740
[illegible]	[illegible]	HA3R	015026	IPAT13	011004	J7	015774	LPAT22	016742
[illegible]	[illegible]	HA3W	014756	IPAT20	011006	KBUF0	016420	LPAT23	016744
[illegible]	[illegible]	HA4R	015036	IPAT21	011010	KBUF1	016422	LPERR =	104413
[illegible]	[illegible]	HA4W	014766	IPAT22	011012	KBUF2	016424	L1	016456
[illegible]	[illegible]	HA5R	015046	IPAT23	011014	KBUF3	016426	L2	016516
[illegible]	[illegible]	HA5W	014776	I1	010202	KDATI0	016410	L3	016520
[illegible]	[illegible]	HCLR	014662	I10	010364	KDATI1	016412	L4	016522
[illegible]	[illegible]	HCLR1	014672	I105	010372	KDATI2	016414	L5	016572
[illegible]	[illegible]	HCMP	014624	I106	010366	kDATI3	016416	L6	016600
[illegible]	[illegible]	HCMP1	014644	I11	010374	KDATO0	016430	MDAT00	017312
[illegible]	[illegible]	HCMP2	014654	I12	010376	KDATO1	016432	MDAT01	017314
[illegible]	[illegible]	HDAT1	015056	I13	010412	KDATO2	016434	MDAT02	017316
[illegible]	[illegible]	HDAT2	015066	I14	010454	KDATO3	016436	MDAT03	017320
[illegible]	[illegible]	HDAT3	015076	I15	010456	KDONE	016450	MDONE	017322
[illegible]	[illegible]	HDAT4	015106	I16	010460	KERR0	016272	MERR0	017146
[illegible]	[illegible]	HDAT5	015116	I17	010476	KERR1	016336	MERR1	017204
[illegible]	[illegible]	HDONE	015126	I2	010220	KERR2	016362	MERR2	017120
[illegible]	[illegible]	HERROR	014700	I20	010536	KPAT0	016440	MERR3	017232
[illegible]	[illegible]	HFLAG	014734	I21	010552	KPAT1	016442	MNUMBE=	000213
[illegible]	[illegible]	HSTD	014546	I22	010556	KPAT2	016444	MNUM0	042210
[illegible]	[illegible]	HT =	000011	I23	010572	K1	016164	MNUM1	042216
[illegible]	[illegible]	H1	014210	I3	010266	K10	016312	MNUM2	042223
[illegible]	[illegible]	H10	014452	I4	010270	K2	016210	MNUM3	042230
[illegible]	[illegible]	H11	014504	I5	010272	K3	016212	MNUM4	042237
[illegible]	[illegible]	H12	014536	I6	010316	K4	016214	MNUM5	042245
[illegible]	[illegible]	H2	014230	I7	010332	K5	016250	MPAT10	017272
[illegible]	[illegible]	H3	014240	JBUF0	016116	K6	016256	MPAT11	017274
[illegible]	[illegible]	H4	014312	JBUF1	016120	K7	016270	MPAT12	017276
[illegible]	[illegible]	H5	014334	JBUF2	016122	LDATI0	016750	MPAT13	017300
[illegible]	[illegible]	H6	014366	JBUF3	016124	LDATI1	016752	MPAT20	017302
[illegible]	[illegible]	H7	014420	JDATI0	016126	LDATI2	016754	MPAT21	017304
[illegible]	[illegible]	IDATI0	011026	JDATI1	016130	LDATI3	016756	MPAT22	017306

MPAT23	017310	NDAT01	017760	ODAT02	020466	PERR16	021006	QERR21	021532
[illegible]	[illegible]	NDAT02	017762	ODAT03	020470	PERR17	021014	QERR22	021540
[illegible]10	[illegible]	NDAT03	017764	ODONE	020534	PERR2	021124	QERR3	021566
[illegible]11	[illegible]	NDONE	020030	OERR0	020224	PERR20	021042	QERR4	021574
[illegible]12	[illegible]	NERR0	017520	OERR1	020324	PERR21	021052	QPAT10	021650
[illegible]13	[illegible]	NERR1	017620	OERR10	020256	PERR22	021060	QPAT11	021652
[illegible]14	042043	NERR10	017552	OERR11	020270	PIRQ =	177772	QPAT12	021654
[illegible]15	042131	NERR11	017564	OERR2	020360	PIRQVE=	000240	QPAT13	021656
[illegible]16	042156	NERR2	017654	OERR20	020332	POWERM	040344	QPAT20	021660
[illegible]17	042163	NERR20	017626	OERR3	020370	PPAT10	021160	QPAT21	021662
MS2	041566	NERR3	017664	OERR4	020400	PPAT11	021162	QPAT22	021664
MS20	042253	NERR4	017674	OERR5	020410	PPAT12	021164	QPAT23	021666
MS21	042314	NERR5	017704	OERR6	020434	PPAT13	021166	Q1	021214
MS22	042353	NERR6	017730	OPAT10	020514	PROGNU=	000001	Q10	021352
MS23	042403	NOOP1	040613	OPAT11	020516	PR0 =	000000	Q2	021236
MS24	042432	NOOP10	041072	OPAT12	020520	PR1 =	000040	Q3 =	021240
MS25	042525	NOOP11	041163	OPAT13	020522	PR2 =	000100	Q4	021242
MS26	042547	NOOP15	040642	OPAT20	020500	PR3 =	000140	Q5	021264
MS27	042564	NOOP2	040740	OPAT21	020502	PR4 =	000200	Q6	021306
MS3	041573	NOOP3	040755	OPAT22	020504	PR5 =	000240	Q7	021316
MS30	042573	NOOP4	040767	OPAT23	020506	PR6 =	000300	Q8	021330
MS31	042602	NOOP5	041004	OPAT24	020510	PR7 =	000340	Q9	021344
MS32	042611	NOOP6	041032	O1	020034	PS =	177776	RDCHR =	104407
MS33	042620	NOOP7	041052	O10	020164	PSW =	177776	RESREG=	104411
MS34	042627	NPAT10	020010	O11	020174	PWRVEC=	000024	RESVEC=	000010
MS35	042636	NPAT11	020012	O12	020176	P1	020540	RSETUP=	104412
MS36	042643	NPAT12	020014	O13	020212	P2	020562	R6	=%000006
MS37	042652	NPAT13	020016	O14	020222	P3 =	020564	R7	=%000007
MS4	041622	NPAT20	017776	O2	020060	P4	020566	SADR	015636
MS40	042670	NPAT21	020000	O3	020062	P5	020610	SAVREG=	104410
MS41	042724	NPAT22	020002	O4	020064	P6	020632	SDAT00	015652
MS415	042704	NPAT23	020004	O5	020102	P7	020642	SDAT01	015654
MS42	042750	NULL	040411	O6	020112	P8	020652	SDAT02	015656
MS43	042766	N1	017326	O7	020122	QDATI0	021700	SDAT03	015660
MS44	043003	N10	017460	O8	020136	QDATI1	021702	SDONE	015662
MS5	041634	N11	017470	O9	020150	QDATI2	021704	SERR0	015342
MS6	041676	N12	017472	PDATI0	021170	QDATI3	021706	SERR1	015552
MS7	041714	N13	017506	PDATI1	021172	QDAT00	021670	SERR10	015362
M1	016776	N14	017516	PDATI2	021174	QDAT01	021672	SERR15	015442
M15	017016	N2	017352	PDATI3	021176	QDAT02	021674	SERR2	015502
M2	017022	N3	017354	PDAT00	021200	QDAT03	021676	SERR20	015462
M3	017024	N4	017356	PDAT01	021202	QDONE	021710	SERR3	015526
M4	017026	N5	017374	PDAT02	021204	QERR0	021356	SERR4	015420
M5	017062	N6	017404	PDAT03	021206	QERR1	021622	SERR5	015604
M6	017066	N7	017414	PDONE	021210	QERR11	021366	SERR6	015514
M7	017076	N8	017432	PERR0	020656	QERR12	021404	SERR7	015540
M8	017106	N9	017444	PERR1	021076	QERR13	021422	SETD1	041360
M9	017116	ODATI0	020524	PERR10	020676	QERR14	021440	SETF1	041352
NDATI0	020020	ODATI1	020526	PERR11	020706	QERR15	021456	SETI1	041366
NDATI1	020022	ODATI2	020530	PERR12	020724	QERR16	021466	SETL1	041374
NDATI2	020024	ODATI3	020532	PERR13	020742	QERR17	021474	SPACE	040414
NDATI3	020026	ODAT00	020462	PERR14	020760	QERR2	021556	SPAT10	015642
NDAT00	017756	ODAT01	020464	PERR15	020776	QERR20	021522	SPAT11	015644

SPAT12	015646	TDATI2	011640	TST5	006142	UPAT33	023000	XDATO1	02-34
SPAT13	015650	TDATI3	011642	TST6	006302	UPAT40	023002	XDATO2	024342
STACK =	001100	TDATO0	011624	TST7	010176	UPAT41	023004	XDATO3	024344
START	003606	TDATO1	011626	TYPE =	104401	UPAT42	023006	XDONE	024376
STFS1	041303	TDATO2	011630	TYPOC =	104402	UPAT43	023010	XERR1	024222
STKLMT=	177774	TDATO3	011632	TYPON =	104404	UROM1	023020	XERR2	024304
STST1	041433	TDONE	011644	TYPOS =	104403	UROM2	023022	XERR3	024250
ST1	041316	TERR0	011412	T1	011042	UROM3	023024	XERR4	024320
ST2	041333	TERR1	011474	T10	011204	UTMP1	023014	XPAT00	024346
SWR	001140	TERR2	011514	T105	011212	UTMP2	023016	XPAT01	024350
SWREG	000176	TERR25	011530	T11	011214	U0	021730	XPAT02	024352
SW0 =	000001	TERR3	011560	T12	011216	U1	021760	XPAT03	024354
SW00 =	000001	TERR4	011542	T13	011232	U10	022316	XPAT10	024356
SW01 =	000002	TKVEC =	000060	T14	011274	U11	022320	XPAT12	024362
SW02 =	000004	TPAT10	011604	T15	011276	U12	022352	XPAT13	024364
SW03 =	000010	TPAT11	011606	T16	011300	U13	022370	XPAT20	024366
SW04 =	000020	TPAT12	011610	T17	011316	U14	022422	XPAT21	024370
SW05 =	000040	TPAT13	011612	T2	011062	U15	022452	XPAT22	024372
SW06 =	000100	TPAT20	011614	T20	011356	U16	022454	XPAT23	024374
SW07 =	000200	TPAT21	011616	T21	011372	U2	022024	XTMP	024334
SW08 =	000400	TPAT22	011620	T22	011374	U3	022056	X1	023606
SW09 =	001000	TPAT23	011622	T23	011406	U4	022122	X10	024000
SW1 =	000002	TPVEC =	000064	T3	011130	U5	022154	X11	024014
SW10 =	002000	TRAPVE=	000034	T4	011132	U6	022220	X12	024050
SW11 =	004000	TRTVEC=	000014	T5	011134	U7	022252	X13	024074
SW12 =	010000	TST1	004304	T6	011160	WDAP00	023572	X14	024102
SW13 =	020000	TST10	011040	T7	011174	WDAT01	023574	X15	024116
SW14 =	040000	TST11	011646	UDONE	023026	WDAT02	023576	X16	024152
SW15 =	100000	TST12	014204	UERR0	022472	WDAT03	023600	X17	024176
SW2 =	000004	TST13	015130	UERR1	022516	WDONE	023602	X2	023642
SW3 =	000010	TST14	015664	UERR10	022524	WPAT00	023562	X20	024204
SW4 =	000020	TST15	016160	UERR11	022566	WPAT01	023564	X21	024220
SW5 =	000040	TST16	016452	UERR2	022602	WPAT02	023566	X3	023666
SW6 =	000100	TST17	016774	UERR20	022610	WPAT03	023570	X4	023674
SW7 =	000200	TST2	004672	UERR21	022652	WSETUP	023530	X5	023712
SW8 =	000400	TST20	017324	UERR3	022666	W1	023030	X6	023746
SW9 =	001000	TST21	020032	UERR4	022716	W10	023242	X7	023772
S1	015132	TST22	020536	UFLAG	023012	W11	023270	YDAT00	024742
S10	015300	TST23	021212	UPAT00	022742	W12	023320	YDAT01	024744
S11	015324	TST24	021712	UPAT01	022744	W13	023344	YDAT02	024746
S12	015334	TST25	023026	UPAT02	022746	W14	023360	YDAT03	024750
S2	015166	TST26	023604	UPAT03	022750	W15	023406	YDONE	025002
S3	015170	TST27	024400	UPAT10	022752	W16	023442	YERR1	024602
S4	015174	TST3	004774	UPAT11	022754	W17	023466	YERR2	024644
S5	015220	TST30	025004	UPAT12	022756	W2	023060	YERR3	024706
S6	015230	TST31	025306	UPAT13	022760	W20	023502	YFLAG	024732
S7	015236	TST32	026340	UPAT20	022762	W3	023104	YPAT00	024752
S8	015272	TST33	030126	UPAT21	022764	W4	023122	YPAT01	024754
S9	015274	TST34	031432	UPAT22	022766	W5	023152	YPAT02	024756
TAB =	000011	TST35	033426	UPAT23	022770	W6	023206	YPAT03	024760
TBITVE=	000014	TST36	034202	UPAT30	022772	W7	023232	YPAT10	024762
TDATIC	011634	TST37	034556	UPAT31	022774	XAPT11	024360	YPAT11	024764
TDATI1	011636	TST4	005700	UPAT32	022776	XDAT00	024336	YPAT12	024766

[illegible]	[illegible]	$BDADR	001122	$ETABL	001336	$PWRDN	037362	$TMP1	001234
[illegible]	[illegible]	$BDDAT	001126	$ETEND	001442	$PWRMG	037516	$TMP10	001252
[illegible]	[illegible]	$BELL	001306	$FATAL	001320	$PWRUP	037434	$TMP11	001254
[illegible]	[illegible]	$CDW1	001376	$FFLG	036636	$QUES	001312	$TMP12	001256
[illegible]	[illegible]	$CDW2	001400	$FILLC	001156	$RDCHR	037122	$TMP13	001260
[illegible]	[illegible]	$CHARC	036140	$FILLS	001155	$RDSZ =	000001	$TMP14	001262
[illegible]	[illegible]	$CKSWR	036640	$GDADR	001120	$REGAD	001160	$TMP15	001264
[illegible]	[illegible]	$CLR.T	034760	$GDDAT	001124	$REG0	001162	$TMP16	001266
[illegible]	[illegible]	$CNTAG	001100	$GET42	034742	$REG1	001164	$TMP17	001270
[illegible]	[illegible]	$CM1 =	000024	$GTSWR	036710	$REG10	001202	$TMP2	001236
[illegible]	[illegible]	$CM2 =	000050	$HD =	000003	$REG11	001204	$TMP20	001272
[illegible]	[illegible]	$CM3 =	000024	$HIBTS	003572	$REG12	001206	$TMP21	001274
[illegible]	[illegible]	$CM4 =	000024	$ICNT	001104	$REG13	001210	$TMP22	001276
[illegible]	[illegible]	$CNTLG	037240	$ILLUP	037540	$REG14	001212	$TMP23	001300
[illegible]	[illegible]	$CNTLU	037242	$INTAG	001135	$REG15	001214	$TMP3	001240
[illegible]	0252[illegible]	$CPUOP	001344	$ITEMB	001114	$REG16	001216	$TMP4	001242
[illegible]	025246	$CRLF	001313	$LF	001314	$REG17	001220	$TMP5	001244
[illegible]	025250	$DDW0	001402	$LFLG	036635	$REG2	001166	$TMP6	001246
[illegible]	025252	$DDW1	001404	$LOOP	035036	$REG20	001222	$TMP7	001250
[illegible]ONE	025302	$DDW10	001426	$LPADR	001106	$REG21	001224	$TN =	000037
[illegible]ERR	025[illegible]	$DDW11	001430	$LPERR	001110	$REG22	001226	$TPB	001152
[illegible]ERR2	025[illegible]	$DDW12	001432	$MADR1	001350	$REG23	001230	$TPFLG	001157
[illegible]FLAG	025236	$DDW13	001434	$MADR2	001354	$REG3	001170	$TPS	001150
[illegible]	025254	$DDW14	001436	$MADR3	001360	$REG4	001172	$TRAP	037276
[illegible]	025256	$DDW15	001440	$MADR4	001364	$REG5	001174	$TRAP2	037320
[illegible]	025260	$DDW2	001406	$MAIL	001316	$REG6	001176	$TRP =	000014
[illegible]	025262	$DDW3	001410	$MAMS1	001346	$REG7	001200	$TRPAD	037332
[illegible]	025264	$DDW4	001412	$MAMS2	001352	$RESRE	035624	$TSTM	003576
[illegible]	025266	$DDW5	001414	$MAMS3	001356	$RTNAD	035040	$TSTNM	001102
[illegible]	025270	$DDW6	001416	$MAMS4	001362	$RTRN	035034	$TYPE	035662
[illegible]	025272	$DDW7	001420	$MBADR	003574	$SAVRE	035566	$TYPEC	036074
[illegible]	025274	$DDW8	001422	$MFLG	036634	$SAVR6	037544	$TYPEX	036142
[illegible]	025276	$DDW9	001424	$MNEW	037265	$SCOPE	035050	$TYPOC	036170
[illegible]	025300	$DEVCT	001326	$MSGAD	001332	$SETUP=	000137	$TYPON	036204
[illegible]	025302	$DEVM	001374	$MSGLG	001334	$STUP =	177777	$TYPOS	036144
[illegible]	025240	$DOAGN	035000	$MSGTY	001316	$SVLAD	035256	$UNIT	001330
[illegible]	025242	$ENDAD	034770	$MSWR	037254	$SVPC =	003572	$UNITM	003602
[illegible]	025026	$ENDCT	034612	$MTYP1	001347	$SWR =	177400	$USWR	001342
[illegible]	025050	$ENULL	035044	$MTYP2	001353	$SWREG	001340	$VECT1	001366
[illegible]	025074	$ENV	001336	$MTYP3	001357	$SWRMK=	000000	$VECT2	001370
[illegible]	025102	$ENVM	001337	$MTYP4	001363	$SWRMS=	000200	$XTSTR	035062
[illegible]	025114	$EOP	034556	$MXCNT	035326	$TAB	040412	$$GET4=	000001
[illegible]	025146	$EOPCT	034604	$NULL	001154	$TBIT	035042	$OFILL	036367
$APTHD	003572	$ERFLG	001103	$NWTST=	000001	$TERM =	000030	. =	071760
$ATYC	036416	$ERMAX	001115	$OCNT	036366	$TESTN	001322	.LPER	040266
$ATY1	036272	$ERROR	035330	$OMODE	036370	$THE	040606	.RSET	040274
$ATY3	036400	$ERRPC	001116	$OVER	035312	$TIMES	001302	.$X =	003572
$ATY4	036410	$ERRTB	001442	$PASS	001324	$TKB	001146		
$AUTOB	001134	$ERTTL	001112	$PASTM	003600	$TKS	001144		
$BASE	001372	$ESCAP	001304	$PWRAD	037522	$TMP0	001232		

. ABS. 071760 000

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

C14


```
DS:DFFPAA.BIN,DS:DFFPAA.SEQ SOL=DFFPAA.P11
RUN-TIME: 86 3 SECONDS
RUN-TIME RATIO: 2   16 1.4
CORE USED: 31   61 PAGES
```