FP11

DIAGNOSTIC NO. 1

MD-11-DFFPA-A

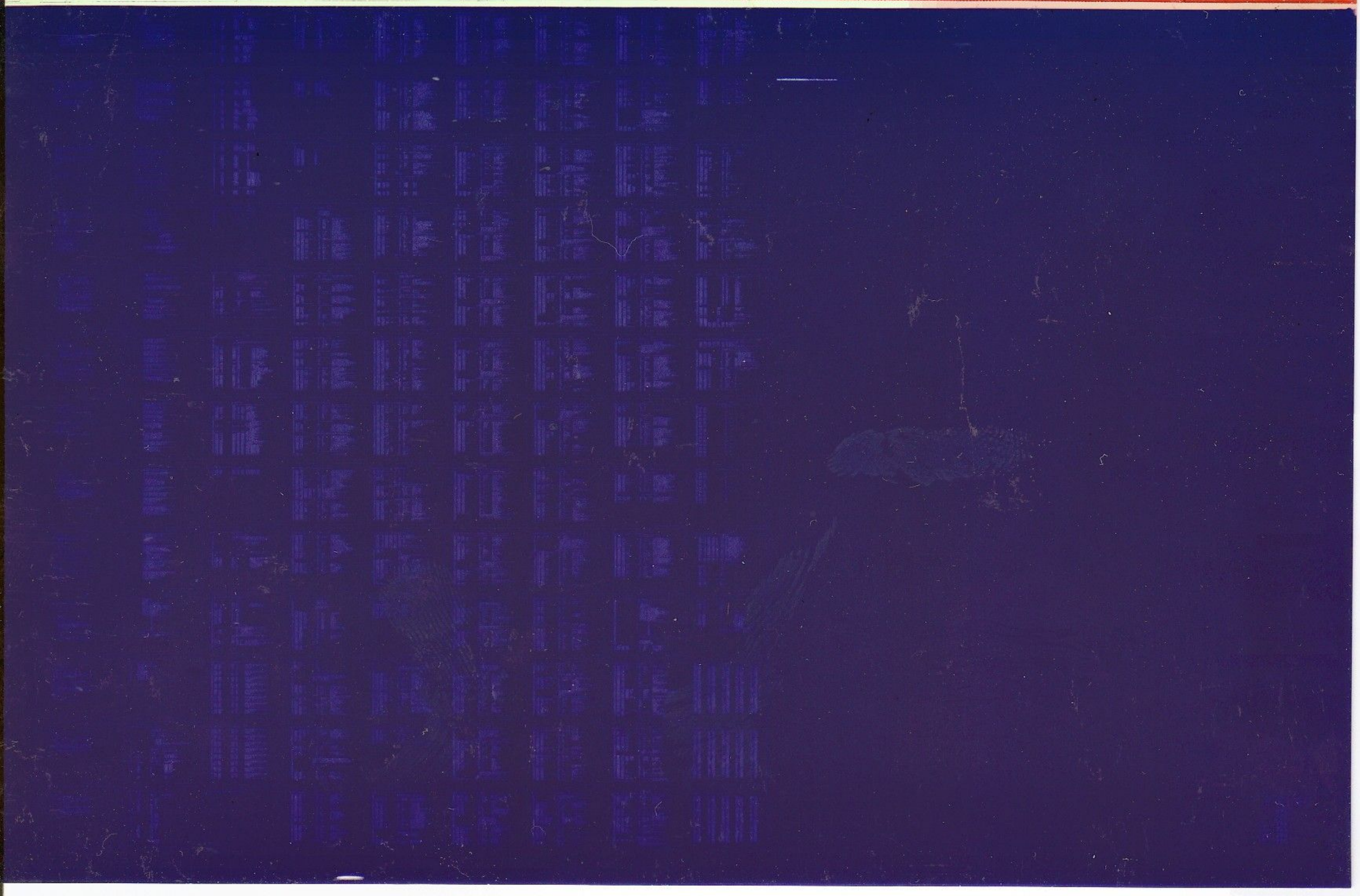
EP-DFFPA-A-DL-A

FICHE 1 OF 1

DEC 1976

digital

MADE IN USA



.REM &

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

2. REQUIREMENTS
   2.1 EQUIPMENT
   2.2 STORAGE
   2.3 PRELIMINARY PROGRAMS

3. LOADING PROCEDURE

4. STARTING PROCEDURE
   4.1 CONTROL SWITCH SETTINGS
   4.2 STARTING ADDRESS
   4.3 PROGRAM AND OPERATOR INTERACTION

5. OPERATING PROCEDURE
   5.1 OPERATIONAL SWITCH SETTINGS
   5.3 OPERATOR ACTION

6. ERRORS
   6.1 SUMMARY
   6.2 ERROR RECOVERY

7. RESTRICTIONS
   7.1 STARTING RESTRICTIONS
   7.2 OPERATING RESTRICTIONS

8. MISCELLANEOUS
   8.1 EXECUTION TIMES
   8.2 STACK POINTER
   8.3 PASS COUNT
   8.4 T-BIT TRAPPING
   8.5 SOFTWARE SWITCH REGISTER
   8.6 INTERRUPTS TEST
   8.7 ACT, APT AND XXDP COMPATIBILITY

9. PROGRAM DESCRIPTION
   9.1 DFFPAA

10. LISTING
   10.1 DFFPAA

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)
LDCTF 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<5> 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 1 AND 11 IN PROGRAM DFFPA BOTH SWITCHES 13 AND 7 MUST = 1.

6.2 ERROR RECOVERY

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

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

7. RESTRICTIONS

NONE

8. MISCELLANEOUS

8.1 EXECUTION TIMES

LESS THAN 10 SECONDS FOR EACH PROGRAM ON ANY PASS.

8.2 STACK POINTER

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

8.3 PASS COUNT

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

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 BIT 13 AND SWITCH REGISTER BIT 7 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 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,...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

673 ABOVE PATTERNS, THEN THE TROUBLE SHOOTER
674 MUST INTUITIVELY TRY TO FIND WHICH OF THE
675 ABOVE CASES (1 THROUGH 4) IS A 'BEST FIT' OF
676 THE SYMPTOMS.
677
678 TEST 12 FPP ACCUMULATORS DUAL ADDRESS TEST
679 ------- ----------
680
681 THIS TEST PERFORMS A DUAL ADDRESSING TEST ON THE
682 FLOATING ACCUMULATORS. NOTE THAT ACCUMULATOR ZERO
683 IS USED TO ACCESS ALL THE OTHERS.
684
685 TEST 13 FSRC MODE 0 WITH ILLEGAL ACCUMULATOR TEST
686 ------- ----------
687
688 THIS IS A TEST OF FSRC MODE 0 WITH ACCUMULATORS 5
689 AND 7. USE OF EITHER OF THESE NON-EXISTENT
690 ACCUMULATORS SHOULD RESULT IN A TRAP TO 244 WITH
691 FEC=2 (ILLEGAL FPP INSTRUCTION).
692
693
694 TEST 14 FSRC MODE 2 TEST
695 ------- ----------
696
697 THIS IS A TEST OF FSRC MODE 2, AUTO INCREMENT MODE.
698
699 TEST 15 FSRC MODE 4 TEST
700 ------- ----------
701
702 THIS IS A TEST OF FSRC MODE 4, AUTO DECREMENT MODE.
703
704 TEST 16 FSRC MODE 2, WITH FD=0, TEST
705 ------- ----------
706
707 THIS IS A TEST OF FSRC MODE 2 WITH FD=0. (AUTO
708 INCREMENT)
709
710 TEST 17 FSRC MODE 2 WITH GR7, IMMEDIATE MODE, TEST
711 ------- ----------
712
713 THIS IS A TEST OF FSRC MODE 2 USING GR7 (THE PC).
714 THIS IS IMMEDIATE MODE.
715
716 TEST 20 FSRC MODE 3 TEST
717 ------- ----------
718
719 THIS IS A TEST OF FSRC MODE 3, AUTO INCREMENT
720 DEFERRED
721
722 TEST 21 FSRC MODE 5 TEST
723 ------- ----------
724
725 THIS IS A TEST OF FSRC MODE 5, AUTO DECREMENT
726 DEFERRED.
727
728 TEST 22 FSRC MODE 6 TEST

```
		-------		----------------

			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 (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)

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.

10. LISTING

```
                                        $

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
```

```
                        $SWRMSK=200
                        TAB=11
                        CRLF=15

                        .SBTTL  BASIC DEFINITIONS

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

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

                        ;*GENERAL PURPOSE REGISTER DEFINITIONS
                        R0=     %0              ;;GENERAL REGISTER
                        R1=     %1              ;;GENERAL REGISTER
                        R2=     %2              ;;GENERAL REGISTER
                        R3=     %3              ;;GENERAL REGISTER
                        R4=     %4              ;;GENERAL REGISTER
                        R5=     %5              ;;GENERAL REGISTER
                        R6=     %6              ;;GENERAL REGISTER
                        R7=     %7              ;;GENERAL REGISTER
                        SP=     %6              ;;STACK POINTER
                        PC=     %7              ;;PROGRAM COUNTER

                        ;*PRIORITY LEVEL DEFINITIONS
                        PP0=    0               ;;PRIORITY LEVEL 0
                        PR1=    40              ;;PRIORITY LEVEL 1
                        PR2=    100             ;;PRIORITY LEVEL 2
                        PR3=    140             ;;PRIORITY LEVEL 3
                        PR4=    200             ;;PRIORITY LEVEL 4
                        PR5=    240             ;;PRIORITY LEVEL 5
                        PR6=    300             ;;PRIORITY LEVEL 6
                        PR7=    340             ;;PRIORITY LEVEL 7

                        ;*"SWITCH REGISTER" SWITCH DEFINITIONS
                        SW15=   100000
                        SW14=   40000
                        SW13=   20000
                        SW12=   10000
                        SW11=   4000
                        SW10=   2000
                        SW09=   1000
                        SW08=   400
                        SW07=   200
                        SW06=   100
```

```
                    000040          SW05=   40
                    000020          SW04=   20
                    000010          SW03=   10
                    000004          SW02=   4
                    000002          SW01=   2
                    000001          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)
                    100000          BIT15=  100000
                    040000          BIT14=  40000
                    020000          BIT13=  20000
                    010000          BIT12=  10000
                    004000          BIT11=  4000
                    002000          BIT10=  2000
                    001000          BIT09=  1000
                    000400          BIT08=  400
                    000200          BIT07=  200
                    000100          BIT06=  100
                    000040          BIT05=  40
                    000020          BIT04=  20
                    000010          BIT03=  10
                    000004          BIT02=  4
                    000002          BIT01=  2
                    000001          BIT00=  1
                                    .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
                    000004          ERRVEC= 4               ;;TIME OUT AND OTHER ERRORS
                    000010          RESVEC= 10              ;;RESERVED AND ILLEGAL INSTRUCTIONS
                    000014          TBITVEC=14              ;;"T" BIT
                    000014          TRTVEC= 14              ;;TRACE TRAP
                    000014          BPTVEC= 14              ;;BREAKPOINT TRAP (BPT)
                    000020          IOTVEC= 20              ;;INPUT/OUTPUT TRAP (IOT) **SCOPE**
                    000024          PWRVEC= 24              ;;POWER FAIL
                    000030          EMTVEC= 30              ;;EMULATOR TRAP (EMT) **ERROR**
                    000034          TRAPVEC=34              ;;"TRAP" TRAP
                    000060          TKVEC=  60              ;;TTY KEYBOARD VECTOR
```

```
                  000064                        TPVEC=  64              ;;TTY PRINTER VECTOR
                  000240                        PIRQVEC=240             ;;PROGRAM INTERRUPT REQUEST VECTOR
                                                        .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)
```

```
1088  001224  000000                $REG21: .WORD   0               ;;CONTAINS (($REGAD)+42)
1089  001226  000000                $REG22: .WORD   0               ;;CONTAINS (($REGAD)+44)
1090  001230  000000                $REG23: .WORD   0               ;;CONTAINS (($REGAD)+46)
1091  001232  000000                $TMP0:  .WORD   0               ;;USER DEFINED
1092  001234  000000                $TMP1:  .WORD   0               ;;USER DEFINED
1093  001236  000000                $TMP2:  .WORD   0               ;;USER DEFINED
1094  001240  000000                $TMP3:  .WORD   0               ;;USER DEFINED
1095  001242  000000                $TMP4:  .WORD   0               ;;USER DEFINED
1096  001244  000000                $TMP5:  .WORD   0               ;;USER DEFINED
1097  001246  000000                $TMP6:  .WORD   0               ;;USER DEFINED
1098  001250  000000                $TMP7:  .WORD   0               ;;USER DEFINED
1099  001252  000000                $TMP10: .WORD   0               ;;USER DEFINED
1100  001254  000000                $TMP11: .WORD   0               ;;USER DEFINED
1101  001256  000000                $TMP12: .WORD   0               ;;USER DEFINED
1102  001260  000000                $TMP13: .WORD   0               ;;USER DEFINED
1103  001262  000000                $TMP14: .WORD   0               ;;USER DEFINED
1104  001264  000000                $TMP15: .WORD   0               ;;USER DEFINED
1105  001266  000000                $TMP16: .WORD   0               ;;USER DEFINED
1106  001270  000000                $TMP17: .WORD   0               ;;USER DEFINED
1107  001272  000000                $TMP20: .WORD   0               ;;USER DEFINED
1108  001274  000000                $TMP21: .WORD   0               ;;USER DEFINED
1109  001276  000000                $TMP22: .WORD   0               ;;USER DEFINED
1110  001300  000000                $TMP23: .WORD   0               ;;USER DEFINED
1111  001302  000000                $TIMES: 0                       ;;MAX. NUMBER OF ITERATIONS
1112  001304  000000                $ESCAPE:0                       ;;ESCAPE ON ERROR ADDRESS
1113  001306  177607  000377        $BELL:  .ASCIZ  <207><377><377> ;;CODE FOR BELL
1114  001312     077                $QUES:  .ASCII  /?/             ;;QUESTION MARK
1115  001313     015                $CRLF:  .ASCII  <15>            ;;CARRIAGE RETURN
1116  001314  000012                $LF:    .ASCIZ  <12>            ;;LINE FEED
1117                                ;;*****************************************************************
1118                                .SBTTL  APT MAILBOX-ETABLE
1119
1120                                ;;*****************************************************************
1121                                .EVEN
1122  001316                        $MAIL:                          ;;APT MAILBOX
1123  001316  000000                $MSGTY: .WORD   AMSGTY          ;;MESSAGE TYPE CODE
1124  001320  000000                $FATAL: .WORD   AFATAL          ;;FATAL ERROR NUMBER
1125  001322  000000                $TESTN: .WORD   ATESTN          ;;TEST NUMBER
1126  001324  000000                $PASS:  .WORD   APASS           ;;PASS COUNT
1127  001326  000000                $DEVCT: .WORD   ADEVCT          ;;DEVICE COUNT
1128  001330  000000                $UNIT:  .WORD   AUNIT           ;;I/O UNIT NUMBER
1129  001332  000000                $MSGAD: .WORD   AMSGAD          ;;MESSAGE ADDRESS
1130  001334  000000                $MSGLG: .WORD   AMSGLG          ;;MESSAGE LENGTH
1131  001336                        $ETABLE:                        ;;APT ENVIRONMENT TABLE
1132  001336     000                $ENV:   .BYTE   AENV            ;;ENVIRONMENT BYTE
1133  001337     000                $ENVM:  .BYTE   AENVM           ;;ENVIRONMENT MODE BITS
1134  001340  000000                $SWREG: .WORD   ASWREG          ;;APT SWITCH REGISTER
1135  001342  000000                $USWR:  .WORD   AUSWR           ;;USER SWITCHES
1136  001344  000000                $CPUOP: .WORD   ACPUOP          ;;CPU TYPE,OPTIONS
1137                                ;*                              BITS 15-11=CPU TYPE
1138                                ;*                                      11/04=01,11/05=02,11/20=03,11/40=04,11/45=05
1139                                ;*                                      11/70=06,PDQ=07,Q=10
1140                                ;*                              BIT 10=REAL TIME CLOCK
1141                                ;*                              BIT  9=FLOATING POINT PROCESSOR
1142                                ;*                              BIT  8=MEMORY MANAGEMENT
1143  001346     000                $MAMS1: .BYTE   AMAMS1          ;;HIGH ADDRESS,M.S. BYTE
```

```
1144  001347  000                 $MTYP1: .BYTE   AMTYP1  ;;MEM. TYPE,BLK#1
1145                                :*                     MEM.TYPE BYTE   --  (HIGH BYTE
1146                                :*                           900 NSEC CORE=001
1147                                :*                           300 NSEC BIPOLAR=002
1148                                :*                           500 NSEC MOS=003
1149  001350  000000                $MADR1: .WORD   AMADR1  ;;HIGH ADDRESS,BLK#1
1150                                :*                       MEM.LAST ADDR.=3 BYTES.THIS WORD AND LOW OF "TYPE" ABOVE
1151  001352  000                 $MAMS2: .BYTE   AMAMS2  ;;HIGH ADDRESS,M.S. BYTE
1152  001353  000                 $MTYP2: .BYTE   AMTYP2  ;;MEM.TYPE,BLK#2
1153  001354  000000              $MADR2: .WORD   AMADR2  ;;MEM.LAST ADDRESS,BLK#2
1154  001356  000                 $MAMS3: .BYTE   AMAMS3  ;;HIGH ADDRESS,M.S.BYTE
1155  001357  000                 $MTYP3: .BYTE   AMTYP3  ;;MEM.TYPE,BLK#3
1156  001360  000000              $MADR3: .WORD   AMADR3  ;;MEM.LAST ADDRESS,BLK#3
1157  001362  000                 $MAMS4: .BYTE   AMAMS4  ;;HIGH ADDRESS,M.S.BYTE
1158  001363  000                 $MTYP4: .BYTE   AMTYP4  ;;MEM.TYPE,BLK#4
1159  001364  000000              $MADR4: .WORD   AMADR4  ;;MEM.LAST ADDRESS,BLK#4
1160  001366  000000              $VECT1: .WORD   AVECT1  ;;INTERRUPT VECTOR#1,BUS PRIORITY#1
1161  001370  000000              $VECT2: .WORD   AVECT2  ;;INTERRUPT VECTOR#2BUS PRIORITY#2
1162  001372  000000              $BASE:  .WORD   ABASE   ;;BASE ADDRESS OF EQUIPMENT UNDER TEST
1163  001374  000000              $DEVM:  .WORD   ADEVM   ;;DEVICE MAP
1164  001376  000000              $CDW1:  .WORD   ACDW1   ;;CONTROLLER DESCRIPTION WORD#1
1165  001400  003003              $CDW2:  .WORD   ACDW2   ;;CONTROLLER DESCRIPTION WORD#2
1166  001402  000000              $DDW0:  .WORD   ADDW0   ;;DEVICE DESCRIPTOR WORD#0
1167  001404  000000              $DDW1:  .WORD   ADDW1   ;;DEVICE DESCRIPTOR WORD#1
1168  001406  000000              $DDW2:  .WORD   ADDW2   ;;DEVICE DESCRIPTOR WORD#2
1169  001410  000000              $DDW3:  .WORD   ADDW3   ;;DEVICE DESCRIPTOR WORD#3
1170  001412  000000              $DDW4:  .WORD   ADDW4   ;;DEVICE DESCRIPTOR WORD#4
1171  001414  000000              $DDW5:  .WORD   ADDW5   ;;DEVICE DESCRIPTOR WORD#5
1172  001416  000000              $DDW6:  .WORD   ADDW6   ;;DEVICE DESCRIPTOR WORD#6
1173  001420  000000              $DDW7:  .WORD   ADDW7   ;;DEVICE DESCRIPTOR WORD#7
1174  001422  000000              $DDW8:  .WORD   ADDW8   ;;DEVICE DESCRIPTOR WORD#8
1175  001424  000000              $DDW9:  .WORD   ADDW9   ;;DEVICE DESCRIPTOR WORD#9
1176  001426  000000              $DDW10: .WORD   ADDW10  ;;DEVICE DESCRIPTOR WORD#10
1177  001430  000000              $DDW11: .WORD   ADDW11  ;;DEVICE DESCRIPTOR WORD#11
1178  001432  000000              $DDW12: .WORD   ADDW12  ;;DEVICE DESCRIPTOR WORD#12
1179  001434  000000              $DDW13: .WORD   ADDW13  ;;DEVICE DESCRIPTOR WORD#13
1180  001436  000000              $DDW14: .WORD   ADDW14  ;;DEVICE DESCRIPTOR WORD#14
1181  001440  000000              $DDW15: .WORD   ADDW15  ;;DEVICE DESCRIPTOR WORD#15
1182
1183
1184  001442                      $ETEND:
1185
```

```
                                .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  04356C  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  002042  045724  000000  070666          .WORD   EM41,DH41,DT41,DF41
 1299  002050  067204
 1300                                  ;ITEM 42
 1301  002052  045761  064703  070720          .WORD   EM42,DH42,DT42,DF42
 1302  002060  067220
 1303                                  ;ITEM 43
 1304  002062  046102  064703  070720          .WORD   EM43,DH43,DT43,DF43
 1305  002070  067220
 1306                                  ;ITEM 44
 1307  002072  046223  000000  070776          .WORD   EM44,DH44,DT44,DF44
 1308  002100  067246
 1309                                  ;ITEM 45
 1310  002102  046223  065005  071046          .WORD   EM45,DH45,DT45,DF45
 1311  002110  067271
 1312                                  ;ITEM 46
 1313  002112  046266  065024  071122          .WORD   EM46,DH46,DT46,DF46
 1314  002120  067316
 1315                                  ;ITEM 47
 1316  002122  046344  065005  071210          .WORD   EM47,DH47,DT47,DF47
 1317  002130  067350
 1318                                  ;ITEM 50
 1319  002132  046462  065050  070614          .WORD   EM50,DH50,DT50,DF50
 1320  002140  067365
 1321                                  ;ITEM 51
 1322  002142  046560  065050  071242          .WORD   EM51,DH51,DT51,DF51
 1323  002150  067411
 1324                                  ;ITEM 52
 1325  002152  046621  063620  071210          .WORD   EM52,DH52,DT52,DF52
 1326  002160  067350
 1327                                  ;ITEM 53
 1328  002162  046742  064512  071300          .WORD   EM53,DH53,DT53,DF53
 1329  002170  067427
 1330                                  ;ITEM 54
 1331  002172  047137  065122  071320          .WORD   EM54,DH54,DT54,DF54
 1332  002200  067436
 1333                                  ;ITEM 55
 1334  002202  047203  063620  071210          .WORD   EM55,DH55,DT55,DF55
 1335  002210  067350
 1336                                  ;ITEM 56
 1337  002212  047324  064512  071300          .WORD   EM56,DH56,DT56,DF56
 1338  002220  067427
 1339                                  ;ITEM 57
 1340  002222  047521  065122  071320          .WORD   EM57,DH57,DT57,DF57
 1341  002230  067436
 1342                                  ;ITEM 60
 1343  002232  047565  064512  071300          .WORD   EM60,DH60,DT60,DF60
 1344  002240  067427
 1345                                  ;ITEM 61
 1346  002242  047762  065122  071320          .WORD   EM61,DH61,DT61,DF61
 1347  002250  067436
 1348                                  ;ITEM 62
 1349  002252  050026  065122  071320          .WORD   EM62,DH62,DT62,DF62
 1350  002260  067436
 1351                                  ;ITEM 63
 1352  002262  050220  065122  071320          .WORD   EM63,DH63,DT63,DF63
 1353  002270  067436
```

N02


```
                                        ;ITEM 64
002272  050412  065232  0713[illegible]          .WORD   EM64,DH64,DT64,DF64
002300  067454
                                        ;ITEM 65
002302  050412  065163  071356          .WORD   EM65,DH65,DT65,DF65
002310  067454
                                        ;ITEM 66
002312  050546  065122  071320          .WORD   EM66,DH66,DT66,DF66
002320  067436
                                        ;ITEM 67
002322  050611  063620  070210          .WORD   EM67,DH67,DT67,DF67
002330  066742
                                        ;ITEM 70
002332  051042  063620  071376          .WORD   EM70,DH70,DT70,DF70
002340  067463
                                        ;ITEM 71
002342  051165  064450  071376          .WORD   EM71,DH71,DT71,DF71
002350  067463
                                        ;ITEM 72
002352  051267  064512  071444          .WORD   EM72,DH72,DT72,DF72
002360  067505
                                        ;ITEM 73
002362  051343  065122  071320          .WORD   EM73,DH73,DT73,DF73
002370  067436
                                        ;ITEM 74
002372  051403  063620  070210          .WORD   EM74,DH74,DT74,DF74
002400  066742
                                        ;ITEM 75
002402  051634  063620  071376          .WORD   EM75,DH75,DT75,DF75
002410  067463
                                        ;ITEM 76
002412  051757  064450  071376          .WORD   EM76,DH76,DT76,DF76
002420  067463
                                        ;ITEM 77
002422  052061  064512  071444          .WORD   EM77,DH77,DT77,DF77
002430  067505
                                        ;ITEM 100
002432  052135  065122  071320          .WORD   EM100,DH100,DT100,DF100
002440  067436
                                        ;ITEM 101
002442  052175  063620  071376          .WORD   EM101,DH101,DT101,DF101
002450  067463
                                        ;ITEM 102
002452  052321  064512  071376          .WORD   EM102,DH102,DT102,DF102
002460  067505
                                        ;ITEM 103
002462  052373  064450  071376          .WORD   EM103,DH103,DT103,DF103
002470  067463
                                        ;ITEM 104
002472  052476  065122  071320          .WORD   EM104,DH104,DT104,DF104
002500  067436
                                        ;ITEM 105
002502  052537  063620  071376          .WORD   EM105,DH105,DT105,DF105
002510  067463
                                        ;ITEM 106
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  053043  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
```

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

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

```
1578  003420  067571
1579                                  ;ITEM 177
1580  003422  061661  065122  071624          .WORD   EM177,DH177,DT177,DF177
1581  003430  067571
1582                                  ;ITEM 200
1583  003432  062016  065122  071624          .WORD   EM200,DH200,DT200,DF200
1584  003440  067571
1585                                  ;ITEM 201
1586  003442  062153  065122  071624          .WORD   EM201,DH201,DT201,DF201
1587  003450  067571
1588                                  ;ITEM 202
1589  003452  062310  065122  071624          .WORD   EM202,DH202,DT202,DF202
1590  003460  067571
1591                                  ;ITEM 203
1592  003462  062445  065122  071624          .WORD   EM203,DH203,DT203,DF203
1593  003470  067571
1594                                  ;ITEM 204
1595  003472  062602  065122  071624          .WORD   EM204,DH204,DT204,DF204
1596  003500  067571
1597                                  ;ITEM 205
1598  003502  062737  063740  067730          .WORD   EM205,DH205,DT205,DF205
1599  003510  066631
1600                                  ;ITEM 206
1601  003512  063004  065122  071624          .WORD   EM206,DH206,DT206,DF206
1602  003520  067571
1603                                  ;ITEM 207
1604  003522  063051  065122  071624          .WORD   EM207,DH207,DT207,DF207
1605  003530  067571
1606                                  ;ITEM 210
1607  003532  063173  065122  071624          .WORD   EM210,DH210,DT210,DF210
1608  003540  067571
1609                                  ;ITEM 211
1610  003542  043233  066544  071730          .WORD   EM211,DH211,DT211,DF211
1611  003550  067655
1612                                  ;ITEM 212
1613  003552  043267  063620  071746          .WORD   EM212,DH212,DT212,DF212
1614  003560  067655
1615                                  ;ITEM 213
1616  003562  043321  063620  071746          .WORD   EM213,DH213,DT213,DF213
1617  003570  067655
1618
1619
1620                                  .SBTTL  ACT11 HOOKS
1621
1622                                  ;;*****************************************************************
1623                                  ;HOOKS REQUIRED BY ACT11
1624          003572                          $SVPC=.                 ;SAVE PC
1625          000046                          .=46
1626  000046  034770                          $ENDAD                  ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .SEOP
1627          000052                          .=52
1628  000052  000000                          .WORD   0               ;;2)SET LOC.52 TO ZERO
1629          003572                          .=$SVPC                 ;; RESTORE PC
1630                                  .SBTTL  APT PARAMETER BLOCK
1631
1632                                  ;;*****************************************************************
1633                                  ;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
```

```
                                        MOV     #.,$LPERR       ;;SETUP THE ERROR LOOP ADDRESS
                                ;;SIZE FOR A HARDWARE SWITCH REGISTER. IF NOT FOUND OR IT IS
                                ;;EQUAL TO A "-1". SETUP FOR A SOFTWARE SWITCH REGISTER.
                                        MOV     @#ERRVEC,-(SP)  ;;SAVE ERROR VECTOR
                                        MOV     #67$,@#ERRVEC   ;;SET UP ERROR VECTOR
                                        MOV     #DSWR,SWR       ;;SETUP FOR A HARDWARE SWITCH REGISTER
                                        MOV     #DDISP,DISPLAY  ;;AND A HARDWARE DISPLAY REGISTER
                                        CMP     #-1,@SWR        ;;TRY TO REFERENCE HARDWARE SWR
                                        BNE     69$             ;;BRANCH IF NO TIMEOUT TRAP OCCURRED
                                                                ;;AND  THE HARDWARE SWR IS NOT = -1
                                        BR      68$             ;;BRANCH IF NO TIMEOUT
                                67$:    MOV     #68$,(SP)       ;;SET UP FOR TRAP RETURN
                                        RTI
                                68$:    MOV     #SWREG,SWR      ;;POINT TO SOFTWARE SWR
                                        MOV     #DISPREG,DISPLAY
                                69$:    MOV     (SP)+,@#ERRVEC  ;;RESTORE ERROR VECTOR

                                        CLR     $PASS           ;;CLEAR PASS COUNT
                                        BITB    #APTSIZE,$ENVM  ;;TEST USER SIZE UNDER APT
                                        BEQ     70$             ;;YES,USE NON-APT SWITCH
                                        MOV     #$SWREG,SWR     ;;NO,USE APT SWITCH REGISTER
                                70$:
                                .SBTTL  TYPE PROGRAM NAME
                                ;;TYPE THE NAME OF THE PROGRAM IF FIRST PASS
                                        INC     #-1             ;;FIRST TIME?
                                        BNE     71$             ;;BRANCH IF NO
                                        CMP     #$ENDAD,@#42    ;;ACT-11?
                                        BEQ     71$             ;;BRANCH IF YES
                                        TYPE    ,72$            ;;TYPE ASCIZ STRING
                                .SBTTL  GET VALUE FOR SOFTWARE SWITCH REGISTER
                                        TST     @#42            ;;ARE WE RUNNING UNDER XXDP/ACT?
                                        BNE     73$             ;;BRANCH IF YES
                                        CMPB    $ENV,#1         ;;ARE WE RUNNING UNDER APT?
                                        BEQ     73$             ;;BRANCH IF YES
                                        CMP     SWR,#SWREG      ;;SOFTWARE SWITCH REG SELECTED?
                                        BNE     74$             ;;BRANCH IF NO
                                        GTSWR                   ;;GET SOFT-SWR SETTINGS
                                        BR      74$
                                73$:    MOVB    #1,$AUTOB       ;;SET AUTO-MODE INDICATOR
                                74$:
                                        BR      71$             ;;GET OVER THE ASCIZ
                                ;;72$:  .ASCIZ  <CRLF>*DFFPA, FPII-A 11/34 FPP DIAGNOSTIC PART 1*<CRLF>
                                71$:

                                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 CPU
                                ;*WITH THE FPP AND ALSO RUNS INTERNALLY WITHIN THE FPP. ONLY DMC AND
                                ;*SMO ARE USED.
                                ;*NOTE THAT A MASK MUST BE USED BECAUSE SOME OF THE FPS BITS CANNOT
                                ;*BE SET.
                                ;*
                                ;*ONLY THE FIRST FIVE ERRORS WILL BE REPORTED INDIVIDUALLY.
                                ;*THIS IS TO PREVENT LOCKING OUT THE COMPLETION OF THE TEST BECAUSE
                                ;*OF VIRTUALLY ENDLESS NUMBER OF ERRORS. ONLY FIVE INDIVIDUAL ERRORS
                                ;*WILL BE REPORTED THEN THE TEST WILL BE COMPLETED AND AN ERROR
                                ;*SUMMARY GIVEN (SEE NOTE BELOW).
                                ;*
                                ;*NOTE THAT THIS TEST KEEPS A DYNAMIC RECORD OF THE LOGICAL 'AND' AND 'OR'
                                ;*OF THE FAILING DATA PATTERNS. THESE CAN BE VERY USEFUL IN DETERMINING
                                ;*STUCK BITS. IF THE USER HAS THE INHIBIT ERROR TYPE OUT SWITCH (SWR13)
                                ;*OFF, THEN THE USER WILL RECIEVE EACH INDIVIDUAL ERROR MESSAGE PLUS
                                ;*AN ERROR SUMMARY AT THE END OF THE TEST. INHIBITING ERROR PRINT OUT
                                ;*WILL INHIBIT ERROR SUMMARY PRINT OUT, EXCEPT IN THE CASE DESCRIBED BELOW.
                                ;*TO GET JUST THE ERROR SUMMARY WITH NO INDIVIDUAL ERROR REPORTS,
                                ;*SET SWITCH REGISTER BIT13 AND SWITCH REGISTER BIT7 BOTH ON.
                                ;*
                                ;**********************************************************************
004304  000004                  T1:     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.
```

```
1802  004434  010005                          MOV     R0,R5           ;COMPUTE 'AND' OF FAILING PATTERNS.
1803  004436  005105                          COM     R5
1804  004440  040502                          BIC     R5,R2
1805
1806  004442  022737  000005  004560          CMP     #5,@#AERFLG     ;SEE IF MORE THAN 5 ERRORS HAVE
1807  004450  103412                          BLO     A05             ;OCCURRED. BR IF YES.
1808                                                                  ;OTHERWISE
1809                                                                  ;REPORT ERROR.
1810  004452  012737  004350  001236          MOV     #A1,@#$TMP2
1811  004460  010037  001240                  MOV     R0,@#$TMP3
1812  004464  010137  001242                  MOV     R1,@#$TMP4
1813  004470  010437  001244                  MOV     R4,@#$TMP5
1814  004474  104001                  A4:     ERROR   1
1815
1816  004476  000751                  A05:    BR      A2              ;CONTINUE TESTING.
1817
1818  004500  005737  004560          A5:     TST     @#AERFLG        ;SEE IF ANY ERRORS OCCURRED.
1819  004504  001471                          BEQ     ADONE           ;IF NOT GO TO NEXT TEST.
1820  004506  032777  020000  174424          BIT     #SW13,@SWR      ;OTHERWISE SEE IF A SUMMARY
1821  004514  001404                          BEQ     A6              ;SHOULD BE TYPED.
1822  004516  032777  000200  174414          BIT     #SW7,@SWR
1823  004524  001461                          BEQ     ADONE
1824
1825  004526                          A6:                             ;TYPE ERROR SUMMARY.
1826  004526  010237  001236                  MOV     R2,@#$TMP2
1827  004532  010337  001240                  MOV     R3,@#$TMP3
1828  004536  012737  004552  001116          MOV     #A7,@#$ERRPC
1829  004544  112737  000002  001114          MOVB    #2,@#$ITEMB
1830  004552  004737  037546          A7:     JSR     PC,@#ERTYPE
1831  004556  000744                          BR      ADONE
1832
1833  00456C  000000                  AERFLG:.WORD    0
1834
1835                                  ;UNABLE TO DECODE FPP INSTRUCTION. TRAPPED TO 244.
1836  004562  011637  001236          AERR1:  MOV     (SP),@#$TMP2    ;SAVE PC OF TRAP.
1837  004566  022626                          CMP     (SP)+,(SP)+
1838  004570  104010                  1$:     ERROR   10
1839  004572  000436                          BR      ADONE
1840
1841                                  ;UNABLE TO DECODE INSTRUCTION. TRAPPED TO 10.
1842  004574  021627  004352          AERR2:  CMP     (SP),#A11+2     ;DID TRAP OCCUR OF FPP INSTRUCTION?
1843  004600  001405                          BEQ     1$
1844  004602  021627  004366                  CMP     (SP),#A12+2
1845  004606  001402                          BEQ     1$
1846  004610  000137  040250                  JMP     @#CPTWO         ;IF NOT FPP INSTRUCTION THEN
1847                                                                  ;REPORT SPURIOUS TRAP TO 10.
1848
1849  004614  011637  001236          1$:     MOV     (SP),@#$TMP2    ;OTHERWISE REPORT IR DECIDE ERROR.
1850  004620  022626                          CMP     (SP)+,(SP)+
1851  004622  104011                  2$:     ERROR   11
1852  004624  000421                          BR      ADONE
1853
1854                                  ;TRAP TO 4 HANDLER:
1855  004626  021627  004352          AERR3:  CMP     (SP),#A11+2     ;DID THE TRAP OCCUR ON THE
1856  004632  001405                          BEQ     1$              ;LDFPS INSTRUCTION?
1857  004634  021627  004366                  CMP     (SP),#A12+2     ;OR THE STFPS INSTRUCTION?
```

```
                                BEQ     2$
004642  000137  040232          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          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 2        CFCC TEST
                        ;*
                        ;*THIS IS A TEST OF THE COPY CONDITION CODES INSTRUCTION, CFCC.
                        ;*
                        ;;*****************************************************************
004672  000004  TST2:   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
```

```
                                BERR1:
004756                                  MOV     R0,@#$TMP3
004756  010037  001240                    MOV     R1,@#$TMP4
004762  010137  001242
004766  104004                  1$:       ERROR   4
004770  000754                            BR      B3

004772                          BDONE:
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 (HAS
                                                                    ;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.
                                ;*
                                ;;*******************************************************************
004774  000004                  TST3:   SCOPE
004776  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
005000  012737  000760  001244          MOV     #760,@#$TMPS
005006  012737  000202  001250  C1:     MOV     #202,@#$TMP7
005014  012737  041352  001252          MOV     #SETF1,@#$TMP10
005022  005000                          CLR     R0

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

005034  170001                  C15:    SETF                            ;TEST INSTRUCTION.

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

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

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

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

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  005530                  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$:     BR      CERR4
2067
2068  005444  032703  000300      2$:     BIT     #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 SYST 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                  TST4:   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,@#STMPS
005742  012737  005750  001236           MOV     #D2,@#STMP2
005750  000000          D2:     .WORD   0
005752  170000          D3:     CFCC
005754  005202                  D4:     INC     R2
005756  005202                          INC     R2

005760  170201                          STFPS   R1              ;REPORT FAILURE. DID NOT TRAP.
005762  010137  001240                   MOV     R1,@#STMP3
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,@#$TMP3
006052  010137  001242                MOV     R1,@#$TMP4
006056  104017                  2$:   ERROR   17
                                
006060  012704  000001          3$:   MOV     #1,R4
006064  170304                  D8:   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  006264  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                        RESTUP                    ;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,@#STMP3
        006222  012737  140000  001242           MOV     #140000,@#STMP4
        006230  104022                  1$:      ERROR   22
        006232  000422                           BR      EDONE

        006234                          EERR1:                                   ;REPORT FEC NOT 2.
        006234  010537  001240                   MOV     R5,@#STMP3
        006240  010437  001242                   MOV     R4,@#STMP4
        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$:
        006262  011637  001236                   MOV     (SP),@#STMP2
        006266  022626                           CMP     (SP)+,(SP)+
        006270  170201                           STFPS   R1
        006272  010137  001240                   MOV     R1,@#STMP3
        006276  104024                  2$:      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
```

```
006304                          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     #FDATI0,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     #FDATI4,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,#FDATI4              ;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$:     MOV     #FDATI0,R1              ;MAKE SURE THE SOURCE DATA WAS
006414 012702 010154                    MOV     #FXDAT0,R2              ;NOT AFFECTED.
006420 012703 000010                    MOV     #10,R3
006424 022122                   2$:     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  174210          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  022222                  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
006702                                                                          ;THE LDD INSTRUCTION.
006702  012737  010154  001240          MOV     #FXDAT0,@#$TMP3
006710  012737  010166  001242          MOV     #FXDAT0+12,@#$TMP4
006716  012737  010110  001244          MOV     #FDATI0,@#$TMP5
006724  012737  010122  001246          MOV     #FDATI0+12,@#$TMP6
006732  104025                  1$:     ERROR   25
006734  000137  010174                  JMP     @#FDONE

006740  012737  010120  001242  FERR1:  MOV     #FDATI4,@#$TMP4                ;FSRC FLOWS FAILURE.
006746  010037  001240                  MOV     R0,@#$TMP3
006752  012737  000762  001244          MOV     #762,@#$TMP5
006760  012737  000321  001250          MOV     #321,@#$TMP7

006766  022700  010110                  CMP     #FDATI0,R0                     ;FSRC MODE 4?
006772  001004                          BNE     1$
006774  012737  000324  001246          MOV     #324,@#$TMP6
007002  000412                          BR      4$

007004  022700  010130          1$:     CMP     #FDATI4+10,R0                  ;FSRC MODE 2?
007010  001004                          BNE     2$
007012  012737  000322  001246          MOV     #322,@#$TMP6
007020  000403                          BR      4$

007022                          2$:
007022  104027                  3$:     ERROR   27
007024  000137  010174                  JMP     @#FDONE

007030                          4$:
007030  104026                  5$:     ERROR   26
007032  000137  010174                  JMP     @#FDONE

007036  012701  006360          FERR2:  MOV     #F4,R1                         ;THE PC WAS INCORRECTLY AFFECTED
                                                                                ;DURING THE INSTRUCTION.
007042  010137  001242          FER2:   MOV     R1,@#$TMP4
007046  162701  000004                  SUB     #4,R1
007052  006303                          ASL     R3
007054  060301                          ADD     R3,R1
007056  010137  001240                  MOV     R1,@#$TMP3
007062  104030                  1$:     ERROR   30
007064  000137  010174                  JMP     @#FDONE

007070  012701  006520          FERR4:  MOV     #F11,R1
007074  000762                          BR      FER2

007076  012737  010142  001242  FERR3:  MOV     #FDAT04,@#$TMP4                ;FAILURE IN THE FDST FLOWS.
007104  010037  001240                  MOV     R0,@#$TMP3
007110  012737  000527  001244          MOV     #527,@#$TMP5
007116  012737  000641  001250          MOV     #641,@#$TMP7

007124  022700  010132                  CMP     #FDAT00,R0                     ;DST MODE 4?
007130  001004                          BNE     1$
007132  012737  000644  001246          MOV     #644,@#$TMP6
007140  000412                          BR      4$

007142  022700  010152          1$:     CMP     #FDAT04+10,R0                  ;DST MODE 2?
007146  001004                          BNE     2$
```

```
2474  007150  012737  000642  001246          MOV     #642,@#$TMP6
2475  007156  000403                          BR      4$
2476
2477  007160                          2$:
2478  007160  104032                 3$:      ERROR   32
2479  007162  000137  010174                  JMP     @#FDONE
2480
2481  007166                          4$:
2482  007166  104031                 5$:      ERROR   31
2483  007170  000137  010174                  JMP     @#FDONE
2484
2485  007174                          FERR5:                                  ;FAILURE OF STD.
2486  007174  010037  001240                  MOV     R0,@#$TMP3
2487  007200  012737  010132  001242          MOV     #FDAT00,@#$TMP4
2488  007206  012737  010150  001244          MOV     #FDAT07,@#$TMP5
2489  007214  012737  010154  001246          MOV     #FXDAT0,@#$TMP6
2490  007222  104033                 1$:      ERROR   33
2491  007224  000137  010174                  JMP     @#FDONE
2492
2493  007230  012701  010144          FERR6:  MOV     #FDAT05,R1              ;DID (BUT GR7) FAIL IN THE FDST
2494  007234  012702  177777                  MOV     #-1,R2                  ;FLOWS?
2495  007240  012703  000003                  MOV     #3,R3
2496  007244  020221                 1$:      CMP     R2,(R1)+
2497  007246  001017                          BNE     5$
2498  007250  077303                          SOB     R3,1$
2499
2500                                                                          ;REPORT FAILURE OF (BUT GR7) IN
2501  007252  010037  001240                  MOV     R0,@#$TMP3              ;THE FDST FLOWS.
2502  007256  012737  000412  001244          MOV     #412,@#$TMP5
2503  007264  012737  000147  001246          MOV     #147,@#$TMP6
2504  007272  012737  000145  001250          MOV     #145,@#$TMP7
2505  007300  104034                 2$:      ERROR   34
2506  007302  000137  010174                  JMP     @#FDONE
2507
2508  007306  012701  010144          5$:     MOV     #FDAT05,R1              ;DID (BUT GR7) FAIL IN THE SRC FLOWS?
2509  007312  012703  000003                  MOV     #3,R3
2510  007316  005721                 6$:      TST     (R1)+
2511  007320  001402                          BEQ     7$
2512  007322  000137  007520                  JMP     @#FERR10
2513  007326  077305                 7$:      SOB     R3,6$
2514
2515                                                                          ;REPORT FAILURE OF (BUT GR7) IN
2516  007330  010037  001240                  MOV     R0,@#$TMP3              ;THE FSRC FLOWS.
2517  007334  012737  000207  001244          MOV     #207,@#$TMP5
2518  007342  012737  000176  001246          MOV     #176,@#$TMP6
2519  007350  012737  000174  001250          MOV     #174,@#$TMP7
2520
2521  007356  104035                 10$:     ERROR   35
2522  007360  000137  010174                  JMP     @#FDONE
2523
2524  007364  012701  010146          FERR7:  MOV     #FDAT06,R1              ;DID (BUT FD) FAIL IN THE FDST FLOWS?
2525  007370  012702  177777                  MOV     #-1,R2
2526  007374  012703  000002                  MOV     #2,R3
2527  007400  020221                 1$:      CMP     R2,(R1)+
2528  007402  001017                          BNE     5$
2529  007404  077303                          SOB     R3,1$
```

```
2530
2531                                                        ;REPORT FAILURE OF (BUT FD) IN THE
2532  007406  010037  001240                  MOV     R0,@#$TMP3        ;FDST FLOWS.
2533  007412  012737  000707  001244          MOV     #707,@#$TMP5
2534  007420  012737  000244  001246          MOV     #244,@#$TMP6
2535  007426  012737  000245  001250          MOV     #245,@#$TMP7
2536  007434  104036            2$:   ERROR   36
2537  007436  000137  010174                  JMP     @#FDONE
2538
2539  007442  012701  010146          5$:     MOV     #FDAT06,R1        ;DID (BUT FD) FAIL IN THE FSRC FLOWS?
2540  007446  012703  000002                  MOV     #2,R3
2541  007452  005721            6$:   TST     (R1)+
2542  007454  001402                  BEQ     7$
2543  007456  000137  007520                  JMP     @#FERR10
2544  007462  077305            7$:   SOB     R3,6$
2545
2546                                                        ;REPORT FAILURE OF (BUT FD) IN THE
2547  007464  010037  001240                  MOV     R0,@#$TMP3        ;FSRC FLOWS.
2548  007470  012737  000441  001244          MOV     #441,@#$TMP5
2549  007476  012737  000076  001246          MOV     #76,@#$TMP6
2550  007504  012737  000077  001250          MOV     #77,@#$TMP7
2551  007512  104037            10$:  ERROR 37
2552  007514  000137  010174                  JMP     @#FDONE
2553
2554  007520            FERR10:                                   ;REPORT DATA ERROR.
2555  007520  010037  001240                  MOV     R0,@#$TMP3
2556  007524  012737  010142  001242          MOV     #FDAT04,@#$TMP4
2557  007532  012737  010150  001244          MOV     #FDAT07,@#$TMP5
2558  007540  012737  010164  001246          MOV     #FXDAT4,@#$TMP6
2559  007546  104040            1$:   ERROR   40
2560  007550  000137  010174                  JMP     @#FDONE
2561
2562  007554            FERR11:                                   ;REPORT BAD FPS.
2563  007554  010137  001240                  MOV     R1,@#$TMP3
2564  007560  012737  000200  001242          MOV     #200,@#$TMP4
2565  007566  104041            1$:   ERROR   41
2566  007570  000137  010174                  JMP     @#FDONE
2567
2568  007574  012737  040411  001264  FERR20: MOV     #NULL,@#$TMP15    ;THE EXECUTION OF THE LDD
2569  007602  005037  001252                  CLR     @#$TMP10          ;CAUSED A TRAP TO 4, BECAUSE
2570  007606  011637  001236                  MOV     (SP),@#$TMP2      ;A FSRC FLOW FAILURE RESULTED
2571  007612  012737  010120  001240          MOV     #FDATI4,@#$TMP3   ;I'M AN ODD ADDRESS.
2572  007620  012737  000321  001250          MOV     #321,@#$TMP7
2573  007626  012737  000762  001244          MOV     #762,@#$TMP5
2574
2575  007634  021627  006362                  CMP     (SP),#F4+2        ;SEE IF FSRC MODE 6 OR 7 WAS
2576  007640  001424                  BEQ     FERR21            ;EXECUTED.
2577
2578  007642  020027  010116                  CMP     R0,#FDATI3        ;FSRC MODE 5?
2579  007646  001006                  BNE     2$
2580
2581                                                        ;REPORT FSRC FLOW FAILURE TO
2582  007650  012737  000325  001246          MOV     #325,@#$TMP6      ;MODE 5.
2583  007656  022626                  CMP     (SP)+,(SP)+
2584  007660  104042            1$:   ERROR   42
2585  007662  000544                  BR      FDONE
```

```
2586
2587  007664  020027  010122          2$:     CMP     R0,#FDATI5              ;FSRC MODE 3?
2588  007670  001402                          BEQ     3$
2589  007672  000137  040232                  JMP     @#CPSPUR
2590
2591  007676                          3$:                                     ;REPORT FSRC     FLOW FAILURE TO
2592  007676  012737  000323  001246          MOV     #323,@#$TMP6            ;MODE 3.
2593  007704  022626                          CMP     (SP)+,(SP)+
2594  007706  104042                  4$:     ERROR   42
2595  007710  000531                          BR      FDONE
2596
2597  007712  022626                  FERR21: CMP     (SP)+,(SP)+             ;REPORT FSRC FLOW FAILURE TO
2598                                                                          ;MODE 6 OR MODE 7.
2599  007714  012737  042156  001264          MOV     #MS16,@#$TMP15
2600  007722  012737  000326  001246          MOV     #326,@#$TMP6
2601  007730  012737  000327  001252          MOV     #327,@#$TMP10
2602  007736  104042                  1$:     ERROR   42
2603  007740  000515                          BR      FDONE
2604
2605  007742  012737  040411  001264  FERR25: MOV     #NULL,@#$TMP15          ;THE EXECUTION OF THE STD INSTRUCTION
2606  007750  005037  001252                  CLR     @#$TMP10                ;TRAPPED TO 4, BECAUSE A FAILURE
2607  007754  012737  010142  001240          MOV     #FDAT04,@#$TMP3         ;IN THE FDST FLOWS RESULTED
2608  007762  011637  001236                  MOV     (SP),@#$TMP2            ;IN AN ODD ADDRESS.
2609  007766  012737  000527  001244          MOV     #527,@#$TMP5
2610  007774  012737  000641  001250          MOV     #641,@#$TMP7
2611
2612  010002  021627  006520                  CMP     (SP),#F10+2             ;FLOW FAILURE TO FDST MODE 6 OR 7?
2613  010006  001424                          BEQ     FERR26
2614
2615  010010  020027  010140                  CMP     R0,#FDAT03              ;DID FDST FLOW FAIL TO MODE 5?
2616  010014  001006                          BNE     2$
2617
2618                                                                          ;REPORT FLOW FAILURE TO FDST
2619  010016  012737  000645  001246          MOV     #645,@#$TMP6            ;MODE 5.
2620  010024  022626                          CMP     (SP)+,(SP)+
2621  010026  104043                  1$:     ERROR   43
2622  010030  000461                          BR      FDONE
2623
2624  010032  020027  010144          2$:     CMP     R0,#FDAT05              ;DID FDST FLOW FAIL TO MODE 3?
2625  010036  001402                          BEQ     3$
2626  010040  000137  040232                  JMP     @#CPSPUR
2627
2628  010044                          3$:                                     ;REPORT FDST FLOW FAILED TO MODE 3.
2629  010044  012737  000643  001246          MOV     #643,@#$TMP6
2630  010052  022626                          CMP     (SP)+,(SP)+
2631  010054  104043                  4$:     ERROR   43
2632  010056  000446                          BR      FDONE
2633
2634  010060                          FERR26:                                 ;REPORT FDST FLOW FAILURE TO MODE
2635  010060  012737  042156  001264          MOV     #MS16,@#$TMP15          ;6 OR MODE 7.
2636  010066  012737  000646  001246          MOV     #646,@#$TMP6
2637  010074  012737  000647  001252          MOV     #647,@#$TMP10
2638  010102  022626                          CMP     (SP)+,(SP)+
2639  010104  104043                  1$:     ERROR   43
2640  010106  000432                          BR      FDONE
2641
```

```
                                FDATI0: -1
                                FDATI1: -1
                                FDATI2: -1
                                FDATI3: -1
                                FDATI4: -1
                                FDATI5: -1
                                FDATI6: -1
                                FDATI7: -1
                                        -1
                                FDATO0: -1
                                FDATO1: -1
                                FDATO2: -1
                                FDATO3: -1
                                FDATO4: -1
                                FDATO5: -1
                                FDATO6: -1
                                FDATO7: -1
                                        -1
                                FXDAT0: -1
                                FXDAT1: -1
                                FXDAT2: -1
                                FXDAT3: -1
                                FXDAT4: 052525
                                FXDAT5: 031463
                                FXDAT6: 007417
                                FXDAT7: 000477

                                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
```

```
02698 010232 012700 011006          MOV   #IPAT20,R0          ;LOAD AC0
02699 010236 172410                 LDD   (R0),AC0
02700
02701 010240 012701 000001          MOV   #1,R1               ;IN CASE THE FSRC FLOWS FAIL
02702 010244 012737 010576 000004   MOV   #I6RR0,@#ERRVECT    ;AN ODD ADDRESS TRAP TO 4 MAY OCCUR.
02703 010252 012737 010266 001236   MOV   #I3,@#$TMP2
02704 010260 012737 042636 001240   MOV   #MS35,@#$TMP3
02705 010266 172401           I3:   LDD   AC1,AC0             ;TEST INSTRUCTION.
02706 010270 000240           I4:   NOP
02707 010272 000240           I5:   NOP
02708
02709 010274 012700 011016          MOV   #IDAT00,R0
02710 010300 174010                 STD   AC0,(R0)            ;GET AC0, THE RESULTS.
02711
02712 010302 012700 011016          MOV   #IDAT00,R0          ;SEE IF DATA IS CORRECT.
02713 010306 012701 011026          MOV   #IDAT10,R1
02714 010312 012702 000004          MOV   #4,R2
02715 010316 022021           I6:   CMP   (R0)+,(R1)+
02716 010320 001424                 BEQ   I105
02717
02718 010322 012700 011022          MOV   #IDAT02,R0          ;SEE IF (BUT FD) FAILED.
02719 010326 012702 000002          MOV   #2,R2
02720 010332 005720           I7:   TST   (R0)+
02721 010334 001413                 BEQ   I10
02722
02723 010336 012700 011022          MOV   #IDAT02,R0
02724 010342 012702 000002          MOV   #2,R2
02725 010346 022720 177777    1$:   CMP   #-1,(R0)+
02726 010352 001402                 BEQ   2$
02727 010354 000137 010660          JMP   @#IERR1
02728 010360 077206           2$:   SOB   R2,1$
02729 010362 000401                 BR    I106
02730 010364 077216           I10:  SOB   R2,I7
02731 010366 000137 010700    I106: JMP   @#IERR2
02732
02733 010372 077227           I105: SOB   R2,I6
02734
02735                               ;NOW TEST THE LOAD INSTRUCTION WITH FSRC MODE ZERO AND FD CLEAR.
02736
02737 010374                   I11:
02738 010374 104413                 LPERR                     ;SET UP THE LOOP ON ERROR ADDRESS.
02739
02740 010376 012700 010776    I12:  MOV   #IPAT10,R0
02741 010402 012701 011026          MOV   #IDAT10,R1
02742 010406 012702 000004          MOV   #4,R2
02743 010412 012021           I13:  MOV   (R0)+,(R1)+
02744 010414 077202                 SOB   R2,I13
02745
02746 010416 012700 011026          MOV   #IDAT10,R0          ;SET UP AC1
02747 010422 172510                 LDD   (R0),AC1
02748
02749 010424 012700 011006          MOV   #IPAT20,R0          ;SET UP AC0
02750 010430 172410                 LDD   (R0),AC0
02751
02752 010432 012701 000001          MOV   #1,R1
02753 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.
```

```
010660                                  IERR1:
010660  012737  011026  001242                  MOV     #IDATI0,@#$TMP4
010666  012737  011016  001244                  MOV     #IDAT00,@#$TMP5
010674  104051                          1$:     ERROR   51
010676  000457                                  BR      IDONE

                                        ;REPORT FAILURE OF (BUT FD)
010700  012737  000153  001244          IERR2:  MOV     #153,@#$TMP5
010706  012737  000434  001246                  MOV     #434,@#$TMP6
010714  012737  000435  001250                  MOV     #435,@#$TMP7
010722                                  IERR25:
010722  104050                          1$:     ERROR   50
010724  000444                                  BR      IDONE
010726  012737  000153  001244          IERR4:  MOV     #153,@#$TMP5
010734  012737  000435  001246                  MOV     #435,@#$TMP6
010742  012737  000434  001250                  MOV     #434,@#$TMP7
010750  000764                                  BR      IERR25

                                        ;REPORT INCORRECT FPS AFTER LOAD INSTRUCTION.
010752                                  IERR3:
010752  012737  010454  001236                  MOV     #I14,@#$TMP2
010760  010037  001240                          MOV     R0,@#$TMP3
010764  012737  000004  001242                  MOV     #4,@#$TMP4
010772  104041                          1$:     ERROR   41
010774  000420                                  BR      IDONE


010776  000000                          IPAT10: 0
011000  170360                          IPAT11: 170360
011002  016161                          IPAT12: 016161
011004  052525                          IPAT13: 052525

011006  177777                          IPAT20: -1
011010  177777                          IPAT21: -1
011012  177777                          IPAT22: -1
011014  177777                          IPAT23: -1

011016  000000                          IDAT00: 0
011020  000000                          IDAT01: 0
011022  000000                          IDAT02: 0
011024  000000                          IDAT03: 0

011026  000000                          IDATI0: 0
011030  000000                          IDATI1: 0
011032  000000                          IDATI2: 0
011034  000000                          IDATI3: 0

011036                                  IDONE:
011036  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 10       FDST MODE 0 TEST
                                ;*
                                ;*THIS IS A TEST OF THE STORE INSTRUCTIONS, STD AND STF,  WITH FDST MODE C.
                                ;*
                                ;**********************************************************************
011040  000004                T10TC:  SCOPE
011042                          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) FORK FAILS
011106  012737  011412  000004          MOV     #TERR0,@#ERRVECT        ;AN 000 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     T10S

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                  T10S:   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
```

```
2922  011232  012021              T13:  MOV   (R0)+,(R1)+
2923  011234  077202                    SOB   R2,T13
2924
2925  011236  012700  011634            MOV   #TDATI0,R0          ;SET UP AC0
2926  011242  172410                    LDD   (R0),AC0
2927
2928  011244  012700  011614            MOV   #TPAT20,R0          ;SET UP AC1
2929  011250  172510                    LDD   (R0),AC1
2930
2931  011252  012701  000001            MOV   #1,R1
2932  011256  012737  011274  001236    MOV   #T14,@#STMP2
2933  011264  012737  042643  001240    MOV   #^M36,@#STMP3
2934  011272  170001                    SETF                      ;CLEAR FD
2935  011274  174001              T14:  STF   AC0,AC1
2936  011276  000240              T15:  NOP
2937  011300  000240              T16:  NOP
2938
2939  011302  005000                    CLR   R0
2940  011304  170200                    STFPS R0                  ;SEE IF FPS IS CLEAR.
2941  011306  022700  000010            CMP   #10,R0
2942  011312  001401                    BEQ   T17
2943  011314  000521                    BR    TERR3
2944
2945  011316              T17:
2946  011316  170011                    SETD                      ;SET FD.
2947
2948  011320  012700  011624            MOV   #TDAT00,R0
2949  011324  174110                    STD   AC1,(R0)            ;PICK UP AC1.
2950
2951  011326  012737  177777  011640    MOV   #-1,@#TDATI2
2952  011334  012737  177777  011642    MOV   #-1,@#TDATI3
2953  011342  012703  011624            MOV   #TDAT00,R3
2954  011346  012704  011634            MOV   #TDATI0,R4
2955  011352  012705  000004            MOV   #4,R5
2956  011356  022324              T20:  CMP   (R3)+,(R4)+           ;WAS THE DATA TRANSFERRED CORRECTLY?
2957  011360  001412                    BEQ   T23
2958
2959  011362  023737  011630  011610    CMP   @#TDAT02,@#TPAT12    ;DID (BLT FD) FAIL.
2960  011370  001401                    BEQ   T22
2961  011372  000440              T21:  BR    TERR1
2962  011374  023737  011632  011612  T22:  CMP   @#TDAT03,@#TPAT13
2963  011402  001373                    BNE   T21
2964  011404  000456                    BR    TERR4
2965
2966  011406  077515              T23:  SOB   R5,T20
2967  011410  000515                    BR    TDONE
2968
2969
2970                                    ;TRAP HERE THROUGH VECTOR 4 IF AN ODD ADDRESS OCCURS.
2971  011412  022716  011132      TERR0:  CMP   #T4,(SP)            ;MAKE SURE THE TRAP WAS ON
2972  011416  001413                    BEQ   1$                  ;AN INSTRUCTION BEING TESTED.
2973  011420  022716  011134            CMP   #T5,(SP)
2974  011424  001410                    BEQ   1$
2975  011426  022716  011276            CMP   #T15,(SP)
2976  011432  001405                    BEQ   1$
2977  011434  022716  011300            CMP   #T16,(SP)
```

D05


```
      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,@#$TMP6
      011550  012737  000160  001250          MOV     #160,@#$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                  TDATI0: 0
      011636  000000                  TDATI1: 0
      011640  000000                  TDATI2: 0
      011642  000000                  TDATI3: 0

      011644                          TDONE:
```

```
3034  011644  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 LOW. 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                 E94
;*       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 N BEING MUST 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 BIT'S 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.
                        ;*
                        ;;*****************************************************************
011646  000004          TST11:  SCOPE
011650  170011                  SETD                            ;SET FD.
                        ;TEST ACCUMULATOR 0 WITH FLOATING ONE
011652  012737  042210  001244          MOV     #MNUM0,@#$TMP5
011660  012737  011710  001236          MOV     #G1,@#$TMP2
011666  012700  014130                  MOV     #GPAT00,R0
011672  012701  014170                  MOV     #GDAT00,R1
011676  104413                  LPERR                           ;SET UP THE LOOP ON ERROR ADDRESS.
011700  004737  013606                  JSR     PC,@#GSETUP             ;LOAD TEST PATTERN.
011704  012703  000102                  MOV     #102,R3
011710                  G1:
011710  172410                  LDD     (R0),AC0
011712  174000                  STD     AC0,AC0
011714  172400                  LDD     AC0,AC0         ;STORE THE TEST PATTERN.
011716  174011                  STD     AC0,(R1)
011720  004737  013704                  JSR     PC,@#GCMP               ;COMPARE THE DATA READ WITH
                                                        ;THAT WHICH WAS WRITTEN.
```

```
 3202  011724  005737  014124          TST     @#GFLAG1
 3203  011730  001004                  BNE     G2
 3204  011732  005137  014124          COM     @#GFLAG1
 3205  011736  000261                  SEC
 3206  011740  000401                  BR      G3
 3207  011742  000241          G2:     CLC
 3208  011744  006160  000006  G3:     ROL     6(R0)                   ;GENERATE THE NEXT TEST PATTERN.
 3209  011750  006160  000004          ROL     4(R0)
 3210  011754  006160  000002          ROL     2(R0)
 3211  011760  006110                  ROL     (R0)
 3212  011762  004737  013664          JSR     PC,@#GRESET             ;RESET DEFAULT PATTERN IN OUTPUT
 3213                                                                  ;BUFFER.
 3214  011766  077330                  SOB     R3,G1
 3215  011770  004737  014022          JSR     PC,@#GSUM               ;TYPE ERROR SUMMARY.
 3216
 3217                          ;TEST ACCUMULATOR 0 WITH FLOATING ZERO
 3218  011774  012737  042210  001244        MOV     #MNUM0,@#$TMP5
 3219  012002  012737  012032  001236        MOV     #G4,@#$TMP2
 3220  012010  012700  014140                MOV     #GPAT10,R0
 3221  012014  012701  014170                MOV     #GDAT00,R1
 3222  012020  104413                        LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
 3223  012022  004737  013626                JSR     PC,@#GSETUP             ;LOAD TEST PATTERN.
 3224  012026  012703  000102                MOV     #102,R3
 3225  012032                          G4:
 3226  012032  172410                        LDD     (R0),AC0
 3227  012034  174000                        STD     AC0,AC0
 3228  012036  172400                        LDD     AC0,AC0         ;STORE THE TEST PATTERN.
 3229  012040  174011                        STD     AC0,(R1)
 3230  012042  004737  013704                JSF     PC,@#GCMP               ;COMPARE THE DATA READ WITH
 3231                                                                  ;THAT WHICH WAS WRITTEN.
 3232  012046  005737  014124                TST     @#GFLAG1
 3233  012052  001004                        BNE     G5
 3234  012054  005137  014124                COM     @#GFLAG1
 3235  012060  000241                        CLC
 3236  012062  000401                        BR      G6
 3237  012064  000261          G5:     SEC
 3238  012066  006160  000006  G6:     ROL     6(R0)                   ;GENERATE THE NEXT TEST PATTERN.
 3239  012072  006160  000004          ROL     4(R0)
 3240  012076  006160  000002          ROL     2(R0)
 3241  012102  006110                  ROL     (R0)
 3242  012104  004737  013664          JSR     PC,@#GRESET             ;RESET DEFAULT PATTERN IN OUTPUT
 3243                                                                  ;BUFFER.
 3244  012110  077330                  SOB     R3,G4
 3245  012112  004737  014022          JSR     PC,@#GSUM               ;TYPE ERROR SUMMARY.
 3246
 3247                          ;TEST ACCUMULATOR 1 WITH FLOATING ONE
 3248  012116  012737  042216  001244        MOV     #MNUM1,@#$TMP5
 3249  012124  012737  012154  001236        MOV     #G7,@#$TMP2
 3250  012132  012700  014130                MOV     #GPAT00,R0
 3251  012136  012701  014170                MOV     #GDAT00,R1
 3252  012142  104413                        LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
 3253  012144  004737  013606                JSR     PC,@#GSETUP             ;LOAD TEST PATTERN.
 3254  012150  012703  000102                MOV     #102,R3
 3255  012154                          G7:
 3256  012154  172410                        LDD     (R0),AC0
 3257  012156  174001                        STD     AC0,AC1
```

```
3258  012160  172401                          LDD     AC1,AC0         ;STORE THE TEST PATTERN.
3259  012162  174011                          STD     AC0,(R1)
3260  012164  004737  013704                  JSR     PC,@#GCMP               ;COMPARE THE DATA READ WITH
3261                                                                    ;THAT WHICH WAS WRITTEN.
3262  012170  005737  014124                  TST     @#GFLAG1
3263  012174  001004                          BNE     G10
3264  012176  005137  014124                  COM     @#GFLAG1
3265  012202  000261                          SEC
3266  012204  000401                          BR      G11
3267  012206  000241          G10:            CLC
3268  012210  006160  000006  G11:            ROL     6(R0)                   ;GENERATE THE NEXT TEST PATTERN.
3269  012214  006160  000004                  ROL     4(R0)
3270  012220  006160  000002                  ROL     2(R0)
3271  012224  006110                          ROL     (R0)
3272  012226  004737  013664                  JSR     PC,@#GRESET             ;RESET DEFAULT PATTERN IN OUTPUT
3273                                                                    ;BUFFER.
3274  012232  077330                          SOB     R3,G7
3275  012234  004737  014022                  JSR     PC,@#GSUM               ;TYPE ERROR SUMMARY.
3276
3277                          ;TEST ACCUMULATOR 1 WITH FLOATING ZERO
3278  012240  012737  042216  001244          MOV     #MNUM1,@#$TMP5
3279  012246  012737  012276  001236          MOV     #G12,@#$TMP2
3280  012254  012700  014140                  MOV     #GPAT10,R0
3281  012260  012701  014170                  MOV     #GDAT00,R1
3282  012264  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
3283  012266  004737  013606                  JSR     PC,@#GSETUP             ;LOAD TEST PATTERN.
3284  012272  012703  000102                  MOV     #102,R3
3285  012276          G12:
3286  012276  172410                          LDD     (R0),AC0
3287  012300  174001                          STD     AC0,AC1
3288  012302  172401                          LDD     AC1,AC0         ;STORE THE TEST PATTERN.
3289  012304  174011                          STD     AC0,(R1).
3290  012306  004737  013704                  JSR     PC,@#GCMP               ;COMPARE THE DATA READ WITH
3291                                                                    ;THAT WHICH WAS WRITTEN.
3292  012312  005737  014124                  TST     @#GFLAG1
3293  012316  001004                          BNE     G13
3294  012320  005137  014124                  COM     @#GFLAG1
3295  012324  000241                          CLC
3296  012326  000401                          BR      G14
3297  012330  000261          G13:            SEC
3298  012332  006160  000006  G14:            ROL     6(R0)                   ;GENERATE THE NEXT TEST PATTERN.
3299  012336  006160  000004                  ROL     4(R0)
3300  012342  006160  000002                  ROL     2(R0)
3301  012346  006110                          ROL     (R0)
3302  012350  004737  013664                  JSR     PC,@#GRESET             ;RESET DEFAULT PATTERN IN OUTPUT
3303                                                                    ;BUFFER.
3304  012354  077330                          SOB     R3,G12
3305  012356  004737  014022                  JSR     PC,@#GSUM               ;TYPE ERROR SUMMARY.
3306
3307                          ;TEST ACCUMULATOR 2 WITH FLOATING ONE
3308  012362  012737  042223  001244          MOV     #MNUM2,@#$TMP5
3309  012370  012737  012420  001236          MOV     #G15,@#$TMP2
3310  012376  012700  014130                  MOV     #GPAT00,R0
3311  012402  012701  014170                  MOV     #GDAT00,R1
3312  012406  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
3313  012410  004737  013606                  JSR     PC,@#GSETUP             ;LOAD TEST PATTERN.
```

```
3314  012414  012703  000102            MOV     #102,R3
3315  012420                    G15:
3316  012420  172410                    LDD     (R0),AC0
3317  012422  174002                    STD     AC0,AC2
3318  012424  172402                    LDD     AC2,AC0         ;STORE THE TEST PATTERN.
3319  012426  174011                    STD     AC0,(R1)
3320  012430  004737  013704            JSR     PC,@#GCMP               ;COMPARE THE DATA READ WITH
3321                                                                    ;THAT WHICH WAS WRITTEN.
3322  012434  005737  014124            TST     @#GFLAG1
3323  012440  001004                    BNE     G16
3324  012442  005137  014124            COM     @#GFLAG1
3325  012446  000261                    SEC
3326  012450  000401                    BR      G17
3327  012452  000241            G16:    CLC
3328  012454  006160  000006    G17:    ROL     6(R0)                   ;GENERATE THE NEXT TEST PATTERN.
3329  012460  006160  000004            ROL     4(R0)
3330  012464  006160  000002            ROL     2(R0)
3331  012470  006110                    ROL     (R0)
3332  012472  004737  013664            JSR     PC,@#GRESET             ;RESET DEFAULT PATTERN IN OUTPUT
3333                                                                    ;BUFFER.
3334  012476  077330                    SOB     R3,G15
3335  012500  004737  014022            JSR     PC,@#GSUM               ;TYPE ERROR SUMMARY.
3336
3337                            ;TEST ACCUMULATOR 2 WITH FLOATING ZERO
3338  012504  012737  042223  001244    MOV     #MNUM2,@#$TMP5
3339  012512  012737  012542  001236    MOV     #G20,@#$TMP2
3340  012520  012700  014140            MOV     #GPAT10,R0
3341  012524  012701  014170            MOV     #GDAT00,R1
3342  012530  104413                    LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
3343  012532  004737  013606            JSR     PC,@#GSETUP             ;LOAD TEST PATTERN.
3344  012536  012703  000102            MOV     #102,R3
3345  012542                    G20:
3346  012542  172410                    LDD     (R0),AC0
3347  012544  174002                    STD     AC0,AC2
3348  012546  172402                    LDD     AC2,AC0         ;STORE THE TEST PATTERN.
3349  012550  174011                    STD     AC0,(R1)
3350  012552  004737  013704            JSR     PC,@#GCMP               ;COMPARE THE DATA READ WITH
3351                                                                    ;THAT WHICH WAS WRITTEN.
3352  012556  005737  014124            TST     @#GFLAG1
3353  012562  001004                    BNE     G21
3354  012564  005137  014124            COM     @#GFLAG1
3355  012570  000241                    CLC
3356  012572  000401                    BR      G22
3357  012574  000261            G21:    SEC
3358  012576  006160  000006    G22:    ROL     6(R0)                   ;GENERATE THE NEXT TEST PATTERN.
3359  012602  006160  000004            ROL     4(R0)
3360  012606  006160  000002            ROL     2(R0)
3361  012612  006110                    ROL     (R0)
3362  012614  004737  013664            JSR     PC,@#GRESET             ;RESET DEFAULT PATTERN IN OUTPUT
3363                                                                    ;BUFFER.
3364  012620  077330                    SOB     R3,G20
3365  012622  004737  014022            JSR     PC,@#GSUM               ;TYPE ERROR SUMMARY.
3366
3367                            ;TEST ACCUMULATOR 3 WITH FLOATING ONE
3368  012626  012737  042230  001244    MOV     #MNUM3,@#$TMP5
3369  012634  012737  012664  001236    MOV     #G23,@#$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,@#GSLM               ;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$:     BIS     (R3)+,(R5)+
3593  013752  077402                          SOB     R4,1$
```

```
                                        MOV     R0,R3                   ;COMPUTE '[redacted]DATA.
                                        MOV     #GAND0,R5
                                        MOV     #4,R4
                                2$:     MOV     (R3)+,R2
                                        COM     R2
                                        BIC     R2,(R5)+
                                        SOB     R4,2$

                                        INC     @#GFLAG2                ;INCREMENT ERROR COUNT.
                                        MOV     R0,@#STMP3
                                        MOV     #GDAT00,@#STMP4
                                3$:     ERROR   44
                                        JMP     @GADR

                                ;SEE IF ANY ERRORS HAVE OCCURRED AND WHETHER OR NOT AN ERROR SUMMARY
                                ;SHOULD BE TYPED.
                                GSJM:   TST     @#GFLAG2                ;ANY ERRORS?
                                        BEQ     3$

                                        BIT     #SW13,@#SWR             ;INHIBIT ERROR PRINT OUT?
                                        BEQ     1$
                                        BIT     #SW7,@#SWR              ;PRINT SUMMARY?
                                        BEQ     3$

                                1$:     MOV     @#GFLAG2,@#STMP6        ;YES PRINT SUMMARY.
                                        MOV     #GAND0,@#STMP3
                                        MOV     #GOR0,@#STMP4
                                        MOV     (SP)+,@#GADR            ;SAVE RETURN ADDRESS.
                                        MOV     #2$,@#ERRPC
                                        MOVB    #45,@#ITEMB
                                2$:     JSR     PC,@#ERTYPE
                                        JMP     @GADR
                                3$:     RTS     PC

                                GFLAG1: 0
                                GFLAG2: 0

                                GPAT00: 0
                                GPAT01: 0
                                GPAT02: 0
                                GPAT03: 0

                                GPAT10: -1
                                GPAT11: -1
                                GPAT12: -1
                                GPAT13: -1

                                GAND0:  -1
                                GAND1:  -1
                                GAND2:  -1
                                GAND3:  -1

                                GOR0:   0
                                GOR1:   0
```

```
3650  014164  000000                  GOR2:   0
3651  014166  000000                  GOR3:   0
3652
3653  014170  000000                  GDAT00: 0
3654  014172  000000                  GDAT01: 0
3655  014174  000000                  GDAT02: 0
3656  014176  000000                  GDAT03: 0
3657
3658  014200  000000                  GADR:   0
3659
3660  014202                          GDONE:
3661  014202  104412                          RSET,JF                 ;GO INITIALIZE THE FPS AND STACK; AND
3662                                                                  ;SEE IF THE USER HAS EXPRESSED
3663                                                                  ;THE DESIRE TO CHANGE THE SOFTWARE
3664                                                                  ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
3665                                                                  ;THE USER TYPED CONTROL G?).
3666
3667
3668                                  ;;*******************************************************************
3669                                  ;*TEST 12       FPP ACCUMULATORS DUAL ADDRESS TEST
3670                                  ;*
3671                                  ;*THIS TEST PERFORMS A DUAL ADDRESSING TEST ON THE FLOATING ACCUMULATORS.
3672                                  ;*NOTE THAT ACCUMULATOR ZERO IS USED TO ACCESS ALL THE OTHERS.
3673                                  ;*
3674                                  ;;*******************************************************************
3675  014204  000004          TST12:  SCOPE
3676  014206  104413                  LPERR                           ;SET UP THE LOOP ON ERROR ADDRESS.
3677
3678  014210  005037  014734  H1:      CLR     @#HFLAG
3679  014214  012700  014736          MOV     #HA1W,R0                ;INITIALIZE THE LOAD BUFFER DATA.
3680  014220  012701  015056          MOV     #HDAT1,R1
3681  014224  012703  000024          MOV     #24,R3
3682  014230  012120  H2:      MOV     (R1)+,(R0)+
3683  014232  077302                  SOB     R3,H2
3684
3685  014234  004767  000422          JSR     PC,HCLR                 ;CLEAR THE OUTPUT DATA BUFFER.
3686
3687  014240  170011  H3:      SETD
3688                                  ;LOAD ACCUMULATOR 1
3689  014242  012700  014736          MOV     #HA1W,R0
3690  014246  172410                  LDD     (R0),AC0
3691  014250  174001                  STD     AC0,AC1
3692                                  ;LOAD ACCUMULATOR 2
3693  014252  012700  014746          MOV     #HA2W,R0
3694  014256  172410                  LDD     (R0),AC0
3695  014260  174002                  STD     AC0,AC2
3696                                  ;LOAD ACCUMULATOR 3
3697  014262  012700  014756          MOV     #HA3W,R0
3698  014266  172410                  LDD     (R0),AC0
3699  014270  174003                  STD     AC0,AC3
3700                                  ;LOAD ACCUMULATOR 4
3701  014272  012700  014766          MOV     #HA4W,R0
3702  014276  172410                  LDD     (R0),AC0
3703  014300  174004                  STD     AC0,AC4
3704                                  ;LOAD ACCUMULATOR 5
3705  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                        MOV     R0,R1
014334 005121                H5:     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                        MOV     R0,R1
014366 005121                H6:     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                        MOV     R0,R1
014420 005121                H7:     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                        MOV     R0,R1
014452 005121                H10:    COM     (R1)+
014454 172410                        LDD     (R0),AC0
014456 174004                        STD     AC0,AC4
014460 004737 014546                 JSR     PC,@#HSTD               ;READ ALL THE ACCUMULATORS BACK.
```

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

```
 3818  014654  077505                  HCMP2:  SOB     R5,HCMP1
 3819  014656  000177  000050                  JMP     @HADR
 3820
 3821                                  ;CLEAR THE DATA OUTPUT BUFFER.
 3822  014662  012704  015006          HCLR:   MOV     #HA1R,R4
 3823  014666  012705  000024                  MOV     #24,R5
 3824  014672  005024                  HCLR1:  CLR     (R4)+
 3825  014674  077502                          SOB     R5,HCLR1
 3826  014676  000207                          RTS     PC
 3827
 3828                                  ;REPORT ERROR.
 3829  014700                          HERROR:
 3830  014700  012703  014736                  MOV     #HA1W,R3
 3831  014704  012704  001236                  MOV     #$TMP2,R4
 3832  014710  012705  000012                  MOV     #12,R5
 3833  014714  010324                  1$:     MOV     R3,(R4)+
 3834  014716  062703  000010                  ADD     #10,R3
 3835  014722  077504                          SOB     R5,1$
 3836  014724  104046                  2$:     ERROR   46
 3837  014726  000137  015126                  JMP     @#HDONE
 3838
 3839
 3840  014732  000000                  HADR:   0
 3841  014734  000000                  HFLAG:  0
 3842
 3843  014736  000000  000000  000000  HA1W:   .WORD   0,0,0,0
 3844  014744  000000
 3845  014746  000000  000000  000000  HA2W:   .WORD   0,0,0,0
 3846  014754  000000
 3847  014756  000000  000000  000000  HA3W:   .WORD   0,0,0,0
 3848  014764  000000
 3849  014766  000000  000000  000000  HA4W:   .WORD   0,0,0,0
 3850  014774  000000
 3851  014776  000000  000000  000000  HA5W:   .WORD   0,0,0,0
 3852  015004  000000
 3853
 3854  015006  000000  000000  000000  HA1R:   .WORD   0,0,0,0
 3855  015014  000000
 3856  015016  000000  000000  000000  HA2R:   .WORD   0,0,0,0
 3857  015024  000000
 3858  015026  000000  000000  000000  HA3R:   .WORD   0,0,0,0
 3859  015034  000000
 3860  015036  000000  000000  000000  HA4R:   .WORD   0,0,0,0
 3861  015044  000000
 3862  015046  000000  000000  000000  HA5R:   .WORD   0,0,0,0
 3863  015054  000000
 3864
 3865  015056  073567  073567  073567  HDAT1:  .WORD   73567,73567,73567,73567
 3866  015064  073567
 3867  015066  063146  063146  063146  HDAT2:  .WORD   63146,63146,63146,63146
 3868  015074  063146
 3869  015076  010421  010421  010421  HDAT3:  .WORD   10421,10421,10421,10421
 3870  015104  010421
 3871  015106  031463  031463  031463  HDAT4:  .WORD   31463,31463,31463,31463
 3872  015114  031463
 3873  015116  042104  042104  042104  HDAT5:  .WORD   42104,42104,42104,42104
```

```
015124  042104

015126                          4DONE:
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
```

```
3930  015242  012700  015642          MOV     #SPAT10,R0              ;LOAD AC0
3931  015246  172410                  LDD     (R0),AC0
3932
3933  015250  012737  015420  000244  MOV     #SERR4,@#FPVECT
3934  015256  012700  000001          MOV     #1,R0
3935  015262  012737  015634  000004  MOV     #SERR5,@#ERRVECT
3936  015270  005003                  CLR     R3
3937
3938  015272  172406          S8:     LDD     AC6,AC0
3939  015274  170000          S9:     CFCC
3940  015276  005203                  INC     R3
3941  015300  005203          S10:    INC     R3
3942
3943  015302  012701  015652          MOV     #SDAT0C,R1
3944  015306  174011                  STD     AC0,(R1)                ;NO TRAP! GET AC0.
3945
3946  015310  012701  015652          MOV     #SDAT0C,R1              ;WAS AC0 MODIFIED.
3947  015314  012702  015642          MOV     #SPAT10,R2
3948  015320  012703  000004          MOV     #4,R3
3949  015324  022122          S11:    CMP     (R1)+,(R2)+
3950  015326  001402                  BEQ     S12
3951  015330  000137  015514          JMP     @#SERR6
3952  015334  077305          S12:    SOB     R3,S11
3953  015336  000137  015540          JMP     @#SERR7
3954
3955                          ;TRAPPED TO 244.
3956  015342  021627  015170  SERR0:  CMP     (SP),#S3                ;PC OF TRAP CORRECT?
3957  015346  001402                  BEQ     1$
3958  015350  000137  040200          JMP     @#FPSPUR
3959
3960  015354  012737  015236  015636  1$:     MOV     #S7,@#SADR
3961
3962  015362  011637  001236  SERR10: MOV     (SP),@#STMP2
3963  015366  022626                  CMP     (SP)+,(SP)+
3964  015370  005004                  CLR     R4
3965  015372  170204                  STFPS   R4                      ;IS FPS CORRECT?
3966  015374  022704  100200          CMP     #100200,R4
3967  015400  001020                  BNE     SERR15
3968
3969  015402  005004                  CLR     R4
3970  015404  170304                  STST    R4                      ;IS FEC CORRECT?
3971  015406  022704  000002          CMP     #2,R4
3972  015412  001023                  BNE     SERR20
3973  015414  000177  000216          JMP     @SADR
3974
3975  015420  021627  015274  SERR4:  CMP     (SP),#S9
3976  015424  001402                  BEQ     1$
3977  015426  000137  040200          JMP     @#FPSPUR
3978  015432  012737  015662  015636  1$:     MOV     #SDONE,@#SADR
3979  015440  000750                  BR      SERR10
3980
3981                          ;REPORT FPS FAILURE:
3982  015442  012737  100200  001242  SERR15: MOV     #100200,@#STMP4
3983  015450  010437  001240          MOV     R4,@#STMP3
3984  015454  104117          1$:     ERROR   117
3985  015456  000177  000154          JMP     @SADR
```

```
                                                ;REPORT FEC BAD:
015462  012737  000002  001242  SERR20: MOV     #2,@#$TMP4
015466  010437  001240              MOV     R4,@#$TMP3
015472  104120              1$:     ERROR   120
015474  000177  000134              JMP     @SADR

                                                ;ACO 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
```

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

```
                                JERR0:  CMP     (SP),#J3        ;SEE IF THE TRAP
                                        BEQ     J10             ;OCCURRED ON THE
                                        CMP     (SP),#J4        ;TESTED INSTRUCTION
                                        BEQ     J10
                                        JMP     @#CPSPUR

                                J10:    MOV     #762,@#$TMP3    ;REPORT FSRC FLOW
                                        MOV     #322,@#$TMP4    ;FAILURE
                                        MOV     (SP),@#$TMP2
                                        CMP     (SP)+,(SP)+
                                1$:     ERROR   52
                                        BR      JDONE

                                JERR1:                          ;REPORT, R0 NOT
                                        MOV     #J2,@#$TMP2     ;CORRECTLY AFFECTED
                                        MOV     R0,@#$TMP3
                                        MOV     #JDATI0+10,@#$TMP4
                                1$:     ERROR   53
                                        BR      JDONE

                                ;REPORT DATA FAILURE

                                JERR2:
                                        MOV     #J2,@#$TMP2
                                        MOV     #JDATI0,@#$TMP3
                                        MOV     #JDAT00,@#$TMP4
                                1$:     ERROR   54
                                        BR      JDONE

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

                                JDATI0:         052525
                                JDATI1:         114631
                                JDATI2:         063146
                                JDATI3:         073567

                                JDAT00:         0
                                JDAT01:         0
                                JDAT02:         0
                                JDAT03:         0

                                JDAT0:          -1
                                JDAT1:          -1
                                JDAT2:          -1
                                JDAT3:          -1


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

```
                                ;;*****************************************************************
                                ;*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  [illegible]             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  000467                  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                  KBJF3:          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     #LDAT10,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     #LDAT10,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,#LDAT12      ;WAS R0 INCREMENTED
4296  016536  001401                          BEQ     1$              ;CORRECTLY BY 4
4297  016540  000421                          BR      LERR1
4298
4299  016542  012737  177777  016754  1$:     MOV     #-1,@#LDAT12
4300  016550  012737  177777  016756          MOV     #-1,@#LDAT13
4301  016556  012702  016750                  MOV     #LDAT10,R2      ;IS DATA CORRECT
4302  016562  012703  016762                  MOV     #LDAT00,R3
4303  016566  012704  000004                  MOV     #4,R4
4304
4305  016572  022223                  L5:     CMP     (R2)+,(R3)+
4306  016574  001401                          BEQ     L6
4307  016576  000427                          BR      LERR2
4308  016600  077404                  L6:     SOB     R4,L5
4309
4310  016602  000473                          BR      LDONE
4311
4312  016604                          LERR1:                          ;REPORT FAILURE
4313  016604  012737  016516  001236          MOV     #L2,@#$TMP2     ;R0 NOT INCREMENTED
4314  016612  010037  001240                  MOV     R0,@#$TMP3      ;BY 4
4315  016616  012737  016754  001242          MOV     #LDAT12,@#$TMP4
4316  016624  104060                  1$:     ERROR   6C
4317  016626  000461                          BR      LDONE
4318
4319  016630                          LERR3:                          ;REPORT DATA FAILURE.
4320  016630  012737  016516  001236          MOV     #L2,@#$TMP2
4321  016636  012737  016750  001240          MOV     #LDAT10,@#$TMP3
```

```
4322  016644  012737  016762  001242          MOV     #LDAT00,@#STMP4
4323  016652  104061                  1$:     ERROR   61
4324  016654  000446                          BR      LDONE
4325
4326  016656  012702  016736          LERR2:  MOV     #LPAT20,R2      ;DID .BUT PC.
4327  016662  012703  016762                  MOV     #LDAT00,R3      ;FAIL.
4328  016666  012704  000004                  MOV     #4,R4
4329  016672  022223                  1$:     CMP     (R2)+,(R3)+
4330  016674  001355                          BNE     LERR3
4331  016676  077403                          SOB     R4,1$
4332  016700  012737  016516  001236          MOV     #L2,@#STMP2
4333  016706  012737  016750  001240          MOV     #LDATI0,@#STMP3
4334  016714  012737  016764  001242          MOV     #LDAT01,@#STMP4
4335  016722  104062                  2$:     ERROR   62
4336  016724  000422                          BR      LDONE
4337
4338  016726  177777          LPAT10: .WORD   -1
4339  016730  177777          LPAT11:         -1
4340  016732  177777          LPAT12:         -1
4341  016734  177777          LPAT13:         -1
4342
4343  016736  052525          LPAT20:         052525
4344  016740  114631          LPAT21:         114631
4345  016742  063142          LPAT22:         063142
4346  016744  073567          LPAT23:         073567
4347  016746  000001                  .WORD   00001
4348  016750  000000          LDATI0:         0
4349  016752  000000          LDATI1:         0
4350  016754  000000          LDATI2:         0
4351  016756  000000          LDATI3:         0
4352  016760  000001                  .WORD   00001
4353  016762  000000          LDAT00:         0
4354  016764  000000          LDAT01:         0
4355  016766  000000          LDAT02:         0
4356  016770  000000          LDAT03:         0
4357
4358  016772                  LDONE:
4359  016772  104412                  RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
4360                                                          ;SEE IF THE USER HAS EXPRESSED
4361                                                          ;THE DESIRE TO CHANGE THE SOFTWARE
4362                                                          ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
4363                                                          ;THE USER TYPED CONTROL G?).
4364
4365
4366                          ;;*****************************************************************
4367                          ;*TEST 17       FSRC MODE 2 WITH GR7, IMMEDIATE MODE. TEST
4368                          ;*
4369                          ;* THIS IS A TEST OF FSRC MODE 2
4370                          ;* USING GR7 (THE PC).  THIS IS IMMEDIATE
4371                          ;* MODE.
4372                          ;*
4373                          ;;*****************************************************************
4374  016774  000004          TST17:  SCOPE
4375
4376  016776                  M1:
4377  016776  170011                  SETD
```

```
017000  012700  017272                 MOV     #MPAT10,R0
017004  172410                         LDD     (R0),AC0          ;LOAD BACKROUND
                                                                 ;PATTERN INTO AC0.
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,@#STMP2      ;OF (BUT GR7)
017126  012737  017302  001240         MOV     #MPAT20,@#STMP3
017134  012737  017312  001242         MOV     #MDAT00,@#STMP4
017142  104063                 1$:     ERROR   63
017144  000466                         BR      MDONE

017146  012705  017022         MERR0:  MOV     #M2,R5            ;REPORT FAILURE
017152  010537  001242                 MOV     R5,@#STMP4        ;PC INCREMENTED
017156  162704  000003                 SUB     #3,R4
017162  006304                         ASL     R4
017164  160405                         SUB     R4,R5
017166  010537  001240                 MOV     R5,@#STMP3
017172  012737  017016  001236         MOV     #M15,@#STMP2
017200  104064                 1$:     ERROR   64
017202  000447                         BR      MDONE
```

```

017204                              MERR1:                                  ;REPORT DATA
017204  012737  017016  001236              MOV     #M15,@#$TMP2            ;FAILURE
017212  012737  017312  001240              MOV     #MDAT00,@#$TMP3
017220  012737  017302  001242              MOV     #MPAT20,@#$TMP4
017226  104066                      1$:     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$:     MOV     (SP),@#$TMP3            ;IF YES REPORT
017250  012737  017022  001242              MOV     #M2,@#$TMP4             ;BAD CONSTANT
017256  012737  017016  001236              MOV     #M15,@#$TMP2            ;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  017520  000004          MOV     #NERR0,@#ERRVECT        ;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 0?
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  001402                          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          MOV     #627,@#$TMP5
017600  012737  000323  001250          MOV     #323,@#$TMP7
017606  012737  000325  001246          MOV     #325,@#$TMP6
017614  104070                  1$:     ERROR   70
017616  000504                          BR      NDONE
017620  012737  000322  001246  NERR1:  MOV     #322,@#$TMP6    ;FSRC MODE 2.
017626  012737  000627  001244  NERR2:  MOV     #627,@#$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,@#$TMP6    ;FSRC MODE 3
017672  000755                          BR      NERR20
017674  012737  000321  001246  NERR4:  MOV     #321,@#$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     #NDATI0,@#$TMP4
017752  104073                  1$:     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
```

```
 4622  017774  052525
 4623  017776  020020                NPAT20: .WORD   NDATIO
 4624  020000  070707                NPAT21:         070707
 4625  020002  070707                NPAT22:         070707
 4626  020004  070707                NPAT23:         070707
 4627  020006  000001                        .WORD   1
 4628  020010  177777                NPAT10: .WORD   -1
 4629  020012  177777                NPAT11:         -1
 4630  020014  177777                NPAT12:         -1
 4631  020016  177777                NPAT13:         -1
 4632
 4633  020020  010421                NDATIO: .WORD   010421
 4634  020022  021042                NDATI1:         021042
 4635  020024  031463                NDATI2:         031463
 4636  020026  042104                NDATI3:         042104
 4637
 4638  020030                        NDONE:
 4639  020030  104412                        RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
 4640                                                                ;SEE IF THE USER HAS EXPRESSED
 4641                                                                ;THE DESIRE TO CHANGE THE SOFTWARE
 4642                                                                ;VIRTUAL CONSOLE SWITCH REGISTER  HAS
 4643                                                                ;THE USER TYPED CONTROL G").
 4644
 4645
 4646                                ;;******************************************************************
 4647                                ;*TEST 21       FSRC MODE 5 TEST
 4648                                ;*
 4649                                ;* THIS IS A TEST OF FSRC MODE 5, AUTO DECREMENT
 4650                                ;* DEFERRED.
 4651                                ;*
 4652                                ;;******************************************************************
 4653  020032  000004                TST21:  SCOPE
 4654
 4655  020034                        01:
 4656  020034  170011                        SETD                    ;SET FD MODE
 4657
 4658  020036  012700  020514                MOV     #OPAT10,R0
 4659  020042  172410                        LDD     (R0),AC0        ;LOAD AC0 WITH A
 4660                                                                ;DEFAULT PATTERN.
 4661  020044  012700  020502                MOV     #OPAT21,R0
 4662  020050  005003                        CLR     R3
 4663  020052  012737  020224  000004       MOV     #OERR0,@#ERRVEC ;IF A FAILURE
 4664                                                                ;OCCURS IN THE FSRC
 4665                                                                ;FLOWS AN ODD ADDR.
 4666                                                                ;TRAP TO 4 MAY OCCUR.
 4667  020060  172450                02:     LDD     @-(R0),AC0      ;TEST INSTRUCTION
 4668  020062  005203                03:     INC     R3
 4669  020064  005203                04:     INC     R3
 4670
 4671  020066  012701  020462                MOV     #ODAT00,R1
 4672  020072  174011                        STD     AC0,(R1)        ;GET THE DATA
 4673
 4674  020074  020027  020500                CMP     R0,#OPAT20      ;WAS R0 DECREMENTED
 4675  020100  001436                        BEQ     012             ;BY 2?
 4676
 4677  020102  020027  020512        05:     CMP     R0,#OPAT21+10   ;FSRC MODE 2
```

```
 4658  020106  001001                        BNE     .+6
 4659  020110  000505                        BR      .ERR1
 4660
 4661  020112  020027  020472        .+6:    CMP     R0,#.PAT21-.D   ;FSRC MODE 4?
 4662  020116  001001                        BNE     .+7
 4663  020120  000517                        BR      .ERR2
 4664
 4665  020122  020027  020502        .+7:    CMP     R0,#.PAT21
 4666
 4667  020126  012702  020464                MOV     #.DAT01,R2      ;FSRC MODE 5?
 4668  020132  012703  000004                MOV     #4,R3
 4669  020136  022227  177777        .08:    CMP     (R2)+,#-1
 4670  020142  001002                        BNE     .+9
 4671  020144  077304                        SOB     R3,.+8
 4672  020146  000510                        BR      .ERR3
 4673
 4674  020150  012702  020462        .+9:    MOV     #.DAT00,R2      ;FSRC MODE 1?
 4675  020154  012703  020502                MOV     #.PAT21,R3
 4676  020160  012704  000004                MOV     #4,R4
 4677  020164  022223                .010:   CMP     (R2)+,(R3)+
 4678  020166  001002                        BNE     .+11
 4679  020170  077403                        SOB     R4,.+10
 4680  020172  000502                        BR      .ERR4
 4681
 4682  020174  000505                .+11:   BR      .ERR5
 4683
 4684  020176  012702  020462        .+12:   MOV     #.DAT00,R2      ;DATA CORRECT?
 4685  020202  012703  020524                MOV     #.DAT10,R3
 4686  020206  012704  000004                MOV     #4,R4
 4687  020212  022223                .013:   CMP     (R2)+,(R3)+
 4688  020214  001002                        BNE     .+14
 4689  020216  077403                        SOB     R4,.+13
 4690  020220  000545                        BR      .DONE
 4691
 4692  020222  000504                .+14:   BR      .ERR6
 4693
 4694                                  ;IF AN ODD ADDRESS TRAP OCCURS COME
 4695                                  ;HERE TO SEE IF THE FAILURE WAS IN THE
 4696                                  ;FSRC FLOWS:
 4697
 4698  020224  022716  020064        .ERR0:  CMP     #04,(SP)        ;FSRC MODE 6 OR 7?
 4699  020230  001412                        BEQ     .ERR10
 4700  020232  022716  020062                CMP     #03,(SP)
 4701  020236  001402                        BEQ     1$
 4702  020240  000137  040232                JMP     @#CPSPUR
 4703  020244  020027  020504        1$:     CMP     R0,#.PAT21+2    ;FSRC MODE 3?
 4704  020250  001425                        BEQ     .ERR1
 4705  020252  000137  040232                JMP     @#CPSPUR
 4706
 4707  020256                        .ERR10:                         ;WENT TO FSRC
 4708  020256  011637  001236                MOV     (SP),@#$TMP2    ;MODE 6 OR 7
 4709  020262  022626                        CMP     (SP)+,(SP)+
 4710  020264  104074                1$:     ERROR   74
 4711  020266  000522                        BR      .DONE
 4712
 4713  020270  011637  001240        .ERR11: MOV     (SP),@#$TMP3    ;WENT TO FSRC MODE
```

```
020274  022626                           CMP     (SP)+,(SP)+      ;3
020276  012737  000627  001244           MOV     #627,@#$TMP5
020304  012737  000325  001250           MOV     #325,@#$TMP7
020312  012737  000323  001246           MOV     #323,@#$TMP6
020320  104075                   1$:     ERROR   75
020322  000504                           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     #ODATI0,@#$TMP4
020456  104100                   1$:     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                   OPAT20: .WORD   ODATI0
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                   ODATI0: .WORD   73567
020526  004210                   ODATI1:         004210
020530  114631                   ODATI2:         114631
020532  125252                   ODATI3:         125252

020534                           ODONE:
```

```
4770  020534  104412                    RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
4771                                                              ;SEE IF THE USER HAS EXPRESSED
4772                                                              ;THE DESIRE TO CHANGE THE SOFTWARE
4773                                                              ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
4774                                                              ;THE USER TYPED CONTROL G?).
4775
4776
4777                              ;;*****************************************************************
4778                              ;*TEST 22       FSRC MODE 6 TEST
4779                              ;*
4780                              ;* THIS IS A TEST OF FSRC MODE 6, INDEX MODE
4781                              ;*
4782                              ;;*****************************************************************
4783  020536  000004              TST22:  SCOPE
4784
4785  020540                      P1:
4786  020540  170011                     SETD                    ;SET FD MODE
4787
4788  020542  012700  021160              MOV     #PPAT10,R0
4789  020546  172410                      LDD     (R0),AC0        ;LOAD A DEFAULT PATTERN
4790                                                              ;INTO AC0
4791  020550  012737  020656  000004      MOV     #PERR0,@#ERRVECT        ;IF THE (BUT FSRC) FORG
4792                                                              ;FAILS AN ODD ADDRESS TRAP
4793  020556  012700  020727              MOV     #PDAT10-241,R0  ;COULD occur.
4794
4795  020562  172460  000241    P2:     LDD     241(R0),AC0
4796          020564              P3=P2+2
4797
4798  020566  012701  021200      P4:     MOV     #PDAT00,R1
4799  020572  174011                      STD     AC0,(R1)        ;GET THE DATA
4800  020574  012703  000004              MOV     #4,R3
4801  020600  012702  021170              MOV     #PDAT10,R2
4802  020604  012701  021200              MOV     #PDAT00,R1
4803  020610  022221              P5:     CMP     (R2)+,(R1)+     ;CHECK THE DATA
4804  020612  001007                      BNE     P6
4805  020614  077303                      SOB     R3,P5
4806  020616  022700  020727              CMP     #PDAT10-241,R0  ;R0 CORRECT?
4807  020622  001401                      BEQ     1$
4808  020624  000512                      BR      PERR21
4809  020626  000137  021210      1$:     JMP     @#PDONE
4810
4811  020632  012701  021200      P6:     MOV     #PDAT00,R1
4812  020636  012703  000004              MOV     #4,R3
4813  020642  022721  177777      P7:     CMP     #-1,(R1)+       ;WAS IT FSRC MODE 0?
4814  020646  001401                      BEQ     P8
4815  020650  000512                      BR      PERR1
4816  020652  077305              P8:     SOB     R3,P7
4817  020654  000523                      BR      PERR2
4818                              ;TRAP TO HERE ON AN ODD ADDRESS
4819  020656  021627  020564      PERR0:  CMP     (SP),#P3
4820  020662  001411                      BEQ     PERR11
4821  020664  021627  020566              CMP     (SP),#P4        ;WAS IT FSRC MODE 7?
4822  020670  001402                      BEQ     PERR10
4823  020672  000137  040232              JMP     @#CPSPUR
4824
4825  020676  012737  000327  001246  PERR10: MOV     #327,@#$TMP6
```

```
 4826  020704  000443                          BR      PERR17
 4827  020706  022700  020727          PERR11: CMP     #PDATIO-241,R0  ;WAS IT FSRC MODE 1
 4828  020712  001004                          BNE     PERR12
 4829  020714  012737  000321  001246          MOV     #321,@#$TMP6
 4830  020722  000434                          BR      PERR17
 4831  020724  022700  020737          PERR12: CMP     #PDATIO-241+10,R0       ;WAS IT FSRC MODE 2
 4832  020730  001004                          BNE     PERR13
 4833  020732  012737  000322  001246          MOV     #322,@#$TMP6
 4834  020740  000425                          BR      PERR17
 4835  020742  022700  020731          PERR13: CMP     #PDATIO-241+2,R0        ;WAS IT FSRC MODE 3
 4836  020746  001004                          BNE     PERR14
 4837  020750  012737  000323  001246          MOV     #323,@#$TMP6
 4838  020756  000416                          BR      PERR17
 4839  020760  022700  020717          PERR14: CMP     #PDATIO-241-10,R0       ;WAS IT FSRC MODE 4
 4840  020764  001004                          BNE     PERR15
 4841  020766  012737  000324  001246          MOV     #324,@#$TMP6
 4842  020774  000407                          BR      PERR17
 4843  020776  022700  020725          PERR15: CMP     #PDATIO-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     #PDATIO-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,@#$TMP2
 4867  021104  012737  021170  001240          MOV     #PDATIO,@#$TMP3
 4868  021112  012737  021200  001242          MOV     #PDATOO,@#$TMP4
 4869  021120  104104                  1$:     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                  PDATO0: .WORD   0
 4891  021202  000000                  PDATO1:         0
 4892  021204  000000                  PDATO2:         0
 4893  021206  000000                  PDATO3:         0
 4894
 4895  021210                          PDONE:
 4896  021210  104412                          RSETUP                  ;GO INITIALIZE THE FFS AND STACK; AND
 4897                                                                  ;SEE IF THE USER HAS EXPRESSED
 4898                                                                  ;THE DESIRE TO CHANGE THE SOFTWARE
 4899                                                                  ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
 4900                                                                  ;THE USER TYPED CONTROL G?).
 4901
 4902
 4903                                  ;;*****************************************************************
 4904                                  ;*TEST 23       FSRC MODE 7 TEST
 4905                                  ;*
 4906                                  ;* THIS IS A TEST OF FSRC MODE 7, INDEX
 4907                                  ;* DEFERRED MODE.
 4908                                  ;*
 4909                                  ;;*****************************************************************
 4910  021212  000004                  TST23:  SCOPE
 4911
 4912  021214                          Q1:
 4913  021214  170011                          SETD
 4914
 4915  021216  012700  021650                  MOV     #QPAT10,R0
 4916  021222  172410                          LDD     (R0),AC0        ;LOAD A DEFAULT
 4917                                                                  ;PATTERN INTO AC0
 4918  021224  012737  021356  000C04          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,@#$TMP2
 5015  021630  012737  021700  001240          MOV     #QDATIO,@#$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   QDATIO
 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          QDATIO: .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 8 ,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                         TST24:  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                 U0:     MOV     #-1,(R0)+
   021734  077103                                 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 AC0
   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 0 INTO AC0
   022024  172410                         U2:     LDD     (R0),AC0
   022026  010037  001252                         MOV     R0,@#$TMP10
   022032  012737  022024  001236                 MOV     #U2,@#$TMP2

   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 AC0
   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     #UPAT20,R0      ;LOAD A POSITIVE NUMBER
                                                        ;INTO ACO
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
                        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     #UPAT00,R0      ;LOAD AC0
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     #UPAT30,R0      ;LOAD A NEGATIVE
                                                        ;NUMBER INTO AL0
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
                        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     #UPAT00,R0      ;LOAD AC0
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     #UPAT40,R0      ;LOAD -0 INTO AC0
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     #UPAT00,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  001402                          BEQ     1$
        022634  042703  000400                  BIC     #400,R3
```

```
                                BR      2$
                1$:             BIS     #400,R3
                2$:             MOV     R3,@#$TMP6
                UERP21:         MOV     R5,@#$TMP3
                                MOV     R4,@#$TMP4
                1$:             ERROR   125
                                BR      UDONE

                                ;INTERRUPT HERE WHEN FIUV SET AND ATTEMPTED TO LOAD-0
                UERR3:          CMP     (SP),#U16
                                BEQ     1$
                                JMP     @#FPSPUR
                1$:             CMP     (SP)+,(SP)+
                                CLR     R0
                                STST    R0                      ;GET FEC.
                                CMP     #14,R0                  ;CORRECT
                                BNE     UERR4
                                BR      UDONE
                UERR4:          MOV     #U15,@#$TMP2
                                MOV     #12,@#$TMP4
                                MOV     R0,@#$TMP3
                1$:             ERROR   130
                                BR      UDONE
                UPAT00: .WORD   0
                UPAT01:         0
                UPAT02:         0
                UPAT03:         0

                UPAT10: .WORD   0                       ;0
                UPAT11:         0
                UPAT12:         0
                UPAT13:         0

                UPAT20: .WORD   010421                  ;POS NUM
                UPAT21:         114631
                UPAT22:         125252
                UPAT23:         177777

                UPAT30:         114631                  ;NEG NUM
                UPAT31:         135673
                UPAT32:         146314
                UPAT33:         167356

                UPAT40:         100000                  ;NEG ZERO
                UPAT41:         0
                UPAT42:         0
                UPAT43:         0

                UFLAG:  .WORD   0
                UTMP1:          0
                UTMP2:          0
                UROM1:          0
                UROM2:          0
                UROM3:          0
                UDONE:
```

```
                                ;;**********************************************************************
                                ;*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                  ^T25:  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,@#STMP2
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$:     ERROR   133
023116  000137  023602                  JMP     @#WDONE
023122  077210                  W4:     SOB     R2,W3
023124  022705  000204                  CMP     #204,R5         ;IS FPS CORRECT
023130  001410                          BEQ     W5
                                                                ;NO
023132  012737  000204  001242          MOV     #204,@#STMP4
023140  010537  001240                  MOV     R5,@#STMP3
023144  104137                  1$:     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,@#STMP2
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
```

```
023232  022021                  W7:     CMP     (R0)+,(R1)+     ;WAS THE RESULT
023234  001402                          BEQ     W10             ;NO. REPORT FAILURE.
023236  104134                  1$:     ERROR   134
023240  000560                          BR      WDONE
023242  077205                  W10:    SOB     R2,W7
023244  022705  000004                  CMP     #4,R5           ;WAS FPS CORRECT
023250  001407                          BEQ     W11
                                                                ;INCORRECT FPS.
023252  012737  000004  001242          MOV     #4,@#$TMP4
023260  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$:     ERROR   136
023500  000440                  BR      WDONE
023502  077203          W20:    SOB     R2,W17
023504  022705  000004          CMP     #4,R5           ;IS FPS CORRECT?
023510  001434                  BEQ     WDONE
                                                        ;NO
023512  012737  000004  001242  MOV     #4,@#STMP4
023520  010537  001240          MOV     R5,@#STMP3
023524  104142          1$:     ERROR   142
023526  000425                  BR      WDONE

                                ;SET UP FOR ERROR CALL

023530  012737  023562  001240  WSETUP: MOV     #WPAT00,@#STMP3
023536  012737  023562  001242          MOV     #WPAT00,@#STMP4
023544  012737  023562  001246          MOV     #WPAT00,@#STMP6
023552  012737  023562  001244          MOV     #WPAT00,@#STMP5
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
023602  104412          WDONE:
                                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          ST26:   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
```

```
5442  023626  172410                          LDD     (R0),AC0
5443  023630  012737  023642  001236          MOV     #X2,@#$TMP2
5444  023636  012700  024356                  MOV     #XPAT10,R0      ;FSRC=0
5445  023642  172010                  X2:     ADDD    (R0),AC0        ;TEST INSTRUCTION
5446  023644  170205                          STFPS   R5
5447  023646  170011                          SETD
5448  023650  012700  024336                  MOV     #XDAT00,R0      ;GET RESULT.
5449  023654  174010                          STD     AC0,(R0)
5450  023656  012701  024346                  MOV     #XPAT00,R1
5451  023662  012702  000004                  MOV     #4,R2
5452  023666  022021                  X3:     CMP     (R0)+,(R1)+     ;IS RESULT CORRECT?
5453  023670  001401                          BEQ     X4
5454  023672  000553                          BR      XERR1
5455  023674  077204                  X4:     SOB     R2,X3
5456  023676  012704  000200                  MOV     #200,R4
5457  023702  020405                          CMP     R4,R5           ;IS FPS CORRECT?
5458  023704  001402                          BEQ     X5
5459  023706  000137  024304                  JMP     @#XERR2
5460  023712                           X5:
5461  023712  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
5462  023714  012700  000200                  MOV     #200,R0
5463  023720  170100                          LDFPS   R0              ;SET DOUBLE MODE
5464  023722  012700  024366                  MOV     #XPAT20,R0      ;SET AC0 TO
5465  023726  010037  024334                  MOV     R0,@#XTMP       ;NEGATIVE NUMBER
5466  023732  172410                          LDD     (R0),AC0
5467  023734  012737  023746  001236          MOV     #X6,@#$TMP2
5468  023742  012700  024356                  MOV     #XPAT10,R0      ;FSRC=0
5469  023746  172010                  X6:     ADDD    (R0),AC0        ;TEST INSTRUCTION
5470  023750  170205                          STFPS   R5
5471  023752  170011                          SETD
5472  023754  012700  024336                  MOV     #XDAT00,R0      ;GET RESULT
5473  023760  174010                          STD     AC0,(R0)
5474  023762  012701  024366                  MOV     #XPAT20,R1
5475  023766  012702  000004                  MOV     #4,R2
5476  023772  022021                  X7:     CMP     (R0)+,(R1)+     ;IS RESULT CORRECT?
5477  023774  001401                          BEQ     X10
5478  023776  000511                          BR      XERR1
5479  024000  077204                  X10:    SOB     R2,X7
5480  024002  012704  000210                  MOV     #210,R4
5481  024006  020405                          CMP     R4,R5           ;IS FPS CORRECT?
5482  024010  001401                          BEQ     X11
5483  024012  000534                          BR      XERR2
5484  024014                           X11:
5485  024014  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
5486  024016  012700  000200                  MOV     #200,R0
5487  024022  170100                          LDFPS   R0              ;SET DOUBLE MODE
5488  024024  012700  024346                  MOV     #XPAT00,R0      ;SET AC0 TO NON-ZERO
5489  024030  010037  024334                  MOV     R0,@#XTMP       ;POSITIVE NUMBER
5490  024034  172410                          LDD     (R0),AC0
5491  024036  012737  024050  001236          MOV     #X12,@#$TMP2
5492  024044  012700  024356                  MOV     #XPAT10,R0      ;FSRC=0
5493  024050  173010                  X12:    SUBD    (R0),AC0        ;TEST INSTRUCTION
5494  024052  170205                          STFPS   R5
5495  024054  170011                          SETD
5496  024056  012700  024336                  MOV     #XDAT00,R0      ;GET RESULT
5497  024062  174010                          STD     AC0,(R0)
```

```
5498  024064  012701  024346          MOV     #XPAT00,R1
5499  024070  012702  000004          MOV     #4,R2
5500  024074  022021          X13:    CMP     (R0)+,(R1)+     ;IS RESULT CORRECT?
5501  024076  001401                  BEQ     X14
5502  024100  000463                  BR      XERR3
5503  024102  077204          X14:    SOB     R2,X13
5504  024104  012704  000200          MOV     #200,R4         ;IS FPS CORRECT?
5505  024110  020405                  CMP     R4,R5
5506  024112  001401                  BEQ     X15
5507  024114  000501                  BR      XERR4
5508  024116          X15:
5509  024116  104413                  LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
5510  024120  012700  000200          MOV     #200,R0
5511  024124  170100                  LDFPS   R0              ;SET DOUBLE MODE
5512  024126  012700  024366          MOV     #XPAT20,R0      ;SET ACO=A NEGATIVE
5513  024132  010037  024334          MOV     R0,@#XTMP       ;NUMBER
5514  024136  172410                  LDD     (R0),AC0
5515  024140  012737  024152  001236  MOV     #X16,@#$TMP2
5516  024146  012700  024356          MOV     #XPAT10,R0      ;FSRC=0
5517  024152  173010          X16:    SUBD    (R0),AC0        ;TEST INSTRUCTION.
5518  024154  170205                  STFPS   R5
5519  024156  170011                  SETD
5520  024160  012700  024336          MOV     #XDAT00,R0      ;GET RESULT
5521  024164  174010                  STD     AC0,(R0)
5522  024166  012701  024366          MOV     #XPAT20,R1
5523  024172  012702  000004          MOV     #4,R2
5524  024176  022021          X17:    CMP     (R0)+,(R1)+     ;IS RESULT CORRECT?
5525  024200  001401                  BEQ     X20
5526  024202  000422                  BR      XERR3
5527  024204  077204          X20:    SOB     R2,X17
5528  024206  012704  000210          MOV     #210,R4         ;IS FPS CORRECT?
5529  024212  020405                  CMP     R4,R5
5530  024214  001401                  BEQ     X21
5531  024216  000440                  BR      XERR4
5532  024220  000466          X21:    BR      XDONE
5533
5534                          ;REPORT DATA ERRORS
5535
5536  024222  012737  024356  001240  XERR1:  MOV     #XPAT10,@#$TMP3
5537  024230  013737  024334  001242          MOV     @#XTMP,@#$TMP4
5538  024236  012737  024336  001244          MOV     #XDAT00,@#$TMP5
5539  024244  104143          1$:     ERROR   143
5540  024246  000453                  BR      XDONE
5541  024250  012737  024356  001240  XERR3:  MOV     #XPAT10,@#$TMP3
5542  024256  013737  024334  001242          MOV     @#XTMP,@#$TMP4
5543  024264  012737  024336  001244          MOV     #XDAT00,@#$TMP5
5544  024272  013737  024334  001246          MOV     @#XTMP,@#$TMP6
5545  024300  104144          1$:     ERROR   144
5546  024302  000435                  BR      XDONE
5547
5548                          ;REPORT FPS ERRORS
5549
5550  024304          XERR2:
5551  024304  010537  001240          MOV     R5,@#$TMP3
5552  024310  010437  001242          MOV     R4,@#$TMP4
5553  024314  104145          1$:     ERROR   145
```

```
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  170011                          SETD
```

```
005610  024460  012700  024742          MOV     #YDAT00,R0      ;GET RESULT
005611  024464  174010                  STD     AC0,(R0)
005612  024466  012702  000004          MOV     #4,R2
005613  024472  013701  024736          MOV     @#YTMP2,R1      ;CHECK RESULT.
005614  024476  022021          Y3:     CMP     (R0)+,(R1)+
005615  024500  001026                  BNE     Y6
005616  024502  077203                  SOB     R2,Y3
005617  024504  023705  024740          CMP     @#YTMP3,R5      ;FPS CORRECT?
005618  024510  001401                  BEQ     Y4
005619  024512  000475                  BR      YERR3
005620  024514  005737  024732  Y4:     TST     @#YFLAG         ;FINISHED TEST?
005621  024520  001015                  BNE     Y5
005622  024522  012737  177777  024732  MOV     #-1,@#YFLAG
005623  024530  012737  024762  024734  MOV     #YPAT10,@#YTMP1
005624  024536  012737  024752  024736  MOV     #YPAT00,@#YTMP2
005625  024544  012737  000200  024740  MOV     #200,@#YTMP3
005626  024552  000726                  BR      Y1
005627  024554  000512          Y5:     BR      YDONE
005628  024556  012702  000004  Y6:     MOV     #4,R2
005629  024562  012700  024734          MOV     #YTMP1,R0       ;DID XOR OF SIGN BIT
005630  024566  012701  024742          MOV     #YDAT00,R1      ;FAIL?
005631  024572  022021          Y7:     CMP     (R0)+,(R1)+
005632  024574  001002                  BNE     YERR1
005633  024576  077203                  SOB     R2,Y7
005634  024600  000421                  BR      YERR2
005635  024602          YERR1:                                  ;DATA FAILURE
005636  024602  012737  024452  001236  MOV     #Y2,@#$TMP2
005637  024610  013737  024734  001240  MOV     @#YTMP1,@#$TMP3
005638  024616  012737  024772  001242  MOV     #YPAT20,@#$TMP4
005639  024624  012737  024742  001244  MOV     #YDAT00,@#$TMP5
005640  024632  013737  024736  001246  MOV     @#YTMP2,@#$TMP6
005641  024640  104147          1$:     ERROR   147
005642  024642  000457                  BR      YDONE
005643  024644          YERR2:                                  ;XOR OF SIGN BIT
005644  024644  012737  024452  001236  MOV     #Y2,@#$TMP2     ;FAILED
005645  024652  013737  024734  001240  MOV     @#YTMP1,@#$TMP3
005646  024660  012737  024772  001242  MOV     #YPAT20,@#$TMP4
005647  024666  012737  024742  001244  MOV     #YDAT00,@#$TMP5
005648  024674  013737  024736  001246  MOV     @#YTMP2,@#$TMP6
005649  024702  104150          1$:     ERROR   150
005650  024704  000436                  BR      YDONE
005651  024706          YERR3:                                  ;FPS WRONG.
005652  024706  012737  024452  001236  MOV     #Y2,@#$TMP2
005653  024714  010537  001240          MOV     R5,@#$TMP3
005654  024720  013737  024740  001242  MOV     @#YTMP3,@#$TMP4
005655  024726  104151          1$:     ERROR   151
005656  024730  000424                  BR      YDONE
005657
005658  024732  000000          YFLAG:  .WORD   0
005659  024734  000000          YTMP1:          0
005660  024736  000000          YTMP2:          0
005661  024740  000000          YTMP3:          0
005662
005663  024742  000000          YDAT00: .WORD   0
005664  024744  000000          YDAT01:         0
005665  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                          ;;*****************************************************************
```

B09


```
                                        ;*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.
                                        ;*
                                        ;;**************************************************************
 025306  000004                         TST31:  SCOPE
 025310                                 AA1:
 025310  104413                                 LPERR                           ;SET UP THE LOOP ON ERROR ADDRESS.
 025312  012700  003240                         MOV     #3240,R0
 025316  170100                                 LDFPS   R0                      ;SET FIV FIV FD AND FT
 025320  012737  025670  000244                 MOV     #AAERR0,@#FPVECT        ;IN CASE THE OVER/UNDER
 025326  012700  026246                         MOV     #AAPAT0,R0              ;FLOWS IN TRAP WILL
                                                                                ;OCCUR
 025332  172410                                 LDD     (R0),AC0                ;SET UP AC0
 025334  012737  025346  001236                 MOV     #AA2,@#$TMP2            ;OPERAND
 025342  012700  026256                         MOV     #AAPAT1,R0
 025346  172010                         AA2:    ADDD    (R0),AC0                ;TEST INSTRUCTION
                                                                                ;SHOULD TRUNCATE
 025350  012700  026236                 AA3:    MOV     #AADAT0,R0
 025354  174010                                 STD     AC0,(R0)                ;GET THE RESULT
 025356  012701  026266                         MOV     #AAPAT2,R1
 025362  012702  000004                         MOV     #4,R2
 025366  022021                         AA4:    CMP     (R0)+,(R1)+             ;CORRECT?
 025370  001414                                 BEQ     AA7
 025372  012700  026276                         MOV     #AAPAT3,R0              ;DID (BUT FT) FAIL
 025376  012701  026236                         MOV     #AADAT0,R1
 025402  012702  000004                         MOV     #4,R2
 025406  022021                         AA5:    CMP     (R0)+,(R1)+
 025410  001401                                 BEQ     AA6
 025412  000561                                 BR      AAERR1  ;DATA ERROR
 025414  077204                         AA6:    SOB     R2,AA5
 025416  000137  026012                         JMP     @#AAERR2                ;(BUT FT) ERROR
 025422  077217                         AA7:    SOB     R2,AA4

                                        ;NOW TEST DOUBLE FLOATING ROUND MODE.

 025424                                 AA10:
 025424  104413                                 LPERR                           ;SET UP THE LOOP ON ERROR ADDRESS.
 025426  012700  003200                         MOV     #3200,R0                ;SET FD FIV FIV. FT=0
 025432  170100                                 LDFPS   R0
 025434  012700  026246                         MOV     #AAPAT0,R0
 025440  172410                                 LDD     (R0),AC0                ;SET UP AC0 OPERAND
 025442  012737  025454  001236                 MOV     #AA11,@#$TMP2
 025450  012700  026256                         MOV     #AAPAT1,R0
 025454  172010                         AA11:   ADDD    (R0),AC0                ;TEST INSTRUCTION
                                                                                ;SHOULD ROUND
 025456  012700  026236                 AA12:   MOV     #AADAT0,R0
 025462  174010                                 STD     AC0,(R0)                ;GET THE RESULT
 025464  012701  026276                         MOV     #AAPAT3,R1
 025470  012702  000004                         MOV     #4,R2
 025474  022021                         AA13:   CMP     (R0)+,(R1)+             ;CORRECT?
 025476  001425                                 BEQ     AA20
 025500  012700  026266                         MOV     #AAPAT2,R0              ;DID (BUT FT) FAIL?
 025504  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,@#STMP2
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     @#STMP2,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$:    JMP     @#FPSPUR
025706                          1$:
```

D09


```
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$:    MOV     (SP),@#$TMP2       ;REPORT OVERFLOW ERROR
025734  022626                            CMP     (SP)+,(SP)+
025736  104154                    21$:    ERROR   154
025740  000137  026336            25$:    JMP     @#AADONE
025744  011637  001236            30$:    MOV     (SP),@#$TMP2       ;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,@#$TMP6
025764  012737  026246  001242    AAER10: MOV     #AAPAT0,@#$TMP4
025772  012737  026256  001240            MOV     #AAPAT1,@#$TMP3
026000  012737  026236  001244            MOV     #AADAT0,@#$TMP5
026006  104162                    1$:     ERROR   162
026010  000552                            BR      AADONE
026012  012737  026266  001246    AAERR2: MOV     #AAPAT2,@#$TMP6 ;(BUT FT) FAILED.
026020  012737  026246  001242            MOV     #AAPAT0,@#$TMP4
026026  012737  026256  001240            MOV     #AAPAT1,@#$TMP3
026034  012737  026236  001244            MOV     #AADAT0,@#$TMP5
026042  104156                    1$:     ERROR   156
026044  000534                            BR      AADONE
026046  012737  026276  001246    AAERR3: MOV     #AAPAT3,@#$TMP6 ;DATA ERROR.
026054  000743                            BR      AAER10
026056  012737  026276  001246    AAERR4: MOV     #AAPAT3,@#$TMP6 ;BAD CONSTANT
026064  012737  026246  001242            MOV     #AAPAT0,@#$TMP4
026072  012737  026256  001240            MOV     #AAPAT1,@#$TMP3
026100  012737  026236  001244            MOV     #AADAT0,@#$TMP5
026106  104160                    1$:     ERROR   160
026110  000512                            BR      AADONE
026112  012737  026276  001246    AAERR5: MOV     #AAPAT3,@#$TMP6 ;(BUT FT) FAILED.
026120  012737  026246  001242            MOV     #AAPAT0,@#$TMP4
026126  012737  026256  001240            MOV     #AAPAT1,@#$TMP3
026134  012737  026236  001244            MOV     #AADAT0,@#$TMP5
026142  104157                    1$:     ERROR   157
026144  000474                            BR      AADONE
026146  012737  026316  001240    AAERR6: MOV     #AAPAT5,@#$TMP3 ;FD=0 AND
026154  012737  026246  001242            MOV     #AAPAT0,@#$TMP4 ;DATA ERROR
026162  012737  026236  001244            MOV     #AADAT0,@#$TMP5
026170  012737  026326  001246            MOV     #AAPAT6,@#$TMP6
026176  104160                    1$:     ERROR   160
026200  000456                            BR      AADONE
026202  012737  026316  001240    AAERR7: MOV     #AAPAT5,@#$TMP3 ;CONSTANT ERROR
026210  012737  026246  001242            MOV     #AAPAT0,@#$TMP4
026216  012737  026236  001244            MOV     #AADAT0,@#$TMP5
026224  012737  026326  001246            MOV     #AAPAT6,@#$TMP6
026232  104161                    1$:     ERROR   161
026234  000440                            BR      AADONE
026236  000000                    AADAT0:         0
```

```
 5946  026240  000000                                  0
 5947  026242  000000                                  0
 5948  026244  000000                                  0
 5949  026246  000200          AAPAT0:                 200
 5950  026250  000000                                  0
 5951  026252  000000                                  0
 5952  026254  000000                                  0
 5953  026256  000200          AAPAT1:                 200
 5954  026260  000000                                  0
 5955  026262  000000                                  0
 5956  026264  000001                                  1
 5957  026266  000400          AAPAT2:                 400
 5958  026270  000000                                  0
 5959  026272  000000                                  0
 5960  026274  000000                                  0
 5961  026276  000400          AAPAT3:                 400
 5962  026300  000000                                  0
 5963  026302  000000                                  0
 5964  026304  000001                                  1
 5965  026306  000400          AAPAT4:                 400
 5966  026310  000000                                  0
 5967  026312  100000                                  100000
 5968  026314  000000                                  0
 5969  026316  000200          AAPAT5:                 200
 5970  026320  000001                                  1
 5971  026322  000000                                  0
 5972  026324  000000                                  0
 5973  026326  000400          AAPAT6:                 400
 5974  026330  000001                                  1
 5975  026332  000000                                  0
 5976  026334  000000                                  0
 5977  026336                  AADONE:
 5978  026336  104412                  RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
 5979                                                          ;SEE IF THE USER HAS EXPRESSED
 5980                                                          ;THE DESIRE TO CHANGE THE SOFTWARE
 5981                                                          ;VIRTJAL CONSOLE SWITCH REGISTER (HAS
 5982                                                          ;THE USER TYPED CONTROL G?).
 5983                  ;;*******************************************************************
 5984                  ;*TEST 32       ADDF AND ADDD WITH E(AC) LESS THAN E(FSRC) TEST
 5985                  ;*
 5986                  ;*THIS IS ATEST OF THE ADDD AND ADDF
 5987                  ;*INSTRUCTIONS AND THE ALIGN AC ALGORITHM
 5988                  ;*FLOWS.  THE CONSTANT (25 FOR FLOATING, 57 FOR
 5989                  ;*DOUBLE) USED IS CHECKED.  THEN SIMPLE
 5990                  ;*AND WORST CASE ALIGNMENT SITUATIONS ARE
 5991                  ;*TRIED.  NOTE E(AC) IS LESS THEN E(FSRC)
 5992                  ;*
 5993                  ;;*******************************************************************
 5994  026340  000004  ^ST32:  SCOPE
 5995                  ;EXPONENT DIFFERENCE=57=71 (OCT) FD=1
 5996  026342          CC1:
 5997  026342  104413                  LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
 5998  026344  012704  003200          MOV     #3200,R4        ;SET FIV,FIV, AND FD
 5999  026350  170104                  LDFPS   R4
 6000  026352  012737  026372  001236  MOV     #CC2,@#$TMP2
 6001  026360  012700  027774          MOV     #CCP0,R0        ;SET ACO 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     #CCDAT0,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     #CCDAT0,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  030004                  MOV     #CCP1,R0                ;FSRC
026514  172010                  CC8:    ADDD    (R0),AC0                ;TEST INSTRUCTION
026516  170205                          STFPS   R5                      ;GET FPS
026520  012700  027764                  MOV     #CCDAT0,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     #CCDAT0,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.
```

```
 6058  026622  172410                          LDD     (R0),AC0
 6059  026624  012704  003000                  MOV     #3000,R4        ;SET FIV,FIV. CLEAR FD.
 6060  026630  170104                          LDFPS   R4
 6061  026632  012700  030054                  MOV     #CCP6,R0        ;FSRC
 6062  026636  172010                  CC14:   ADDF    (R0),AC0        ;TEST INSTRUCTION
 6063  026640  170205                          STFPS   R5
 6064  026642  170011                          SETD                    ;REENTER DOUBLE MOVE
 6065  026644  012700  027764                  MOV     #CCDAT0,R0      ;GET THE RESULT
 6066  026650  174010                          STD     AC0,(R0)
 6067  026652  012701  030054                  MOV     #CCP6,R1        ;IS THE RESULT CORRECT?
 6068  026656  012702  000002                  MOV     #2,R2
 6069  026662  022021                  CC15:   CMP     (R0)+,(R1)+
 6070  026664  001415                          BEQ     CC18
 6071  026666  012700  027764                  MOV     #CCDAT0,R0      ;WAS A BAD CONSTANT
 6072  026672  012701  030034                  MOV     #CCP3,R1        ;USED (NOT 25) IN
 6073  026676  012702  000002                  MOV     #2,R2           ;THE ALIGN FLOWS?
 6074  026702  022021                  CC16:   CMP     (R0)+,(R1)+
 6075  026704  001402                          BEQ     CC17
 6076  026706  000137  027532                  JMP     @#CCER5         ;DATA ERROR F
 6077  026712  077205                  CC17:   SOB     R2,CC16
 6078  026714  000137  027566                  JMP     @#CCER6         ;BAD CONSTANT F
 6079  026720  077220                  CC18:   SOB     R2,CC15
 6080  026722  020405                          CMP     R4,R5
 6081  026724  001402                          BEQ     CC19
 6082  026726  000137  027344                  JMP     @#CCER90        ;BAD FPS.
 6083                                  ;EXPONENT DIFFERENCE=24=30 (OCT) FD=0
 6084  026732                          CC19:
 6085  026732  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
 6086  026734  012737  026762  001236          MOV     #CC20,@#$TMP2
 6087  026742  012700  030024                  MOV     #CCP3,R0        ;SET UP AC0 OPERAND.
 6088  026746  172410                          LDD     (R0),AC0
 6089  026750  012704  003000                  MOV     #3000,R4        ;SET FIV,FIV. CLEAR FD.
 6090  026754  170104                          LDFPS   R4
 6091  026756  012700  030044                  MOV     #CCP5,R0        ;FSRC
 6092  026762  172010                  CC20:   ADDF    (R0),AC0        ;TEST INSTRUCTION
 6093  026764  170205                          STFPS   R5
 6094  026766  170011                          SETD                    ;REENTER DOUBLE MOVE
 6095  026770  012700  027764                  MOV     #CCDAT0,R0      ;GET THE RESULT
 6096  026774  174010                          STD     AC0,(R0)
 6097  026776  012701  030074                  MOV     #CCP10,R1       ;IS THE RESLT CORRECT?
 6098  027002  012702  000002                  MOV     #2,R2
 6099  027006  022021                  CC21:   CMP     (R0)+,(R1)+
 6100  027010  001415                          BEQ     CC24
 6101  027012  012700  027764                  MOV     #CCDAT0,R0      ;WAS A BAD CONSTANT
 6102  027016  012701  030044                  MOV     #CCP5,R1        ;USED (NOT 25) IN
 6103  027022  012702  000002                  MOV     #2,R2           ;THE ALIGN FLOWS?
 6104  027026  022021                  CC22:   CMP     (R0)+,(R1)+
 6105  027030  001402                          BEQ     CC23
 6106  027032  000137  027622                  JMP     @#CCER7         ;DATA ERROR F
 6107  027036  077205                  CC23:   SOB     R2,CC22
 6108  027040  000137  027640                  JMP     @#CCER8 ;BAD CONSTANT F
 6109  027044  077220                  CC24:   SOB     R2,CC21
 6110  027046  020405                          CMP     R4,R5
 6111  027050  001402                          BEQ     CC25
 6112  027052  000137  027344                  JMP     @#CCER90        ;BAD FPS.
 6113                                  ;EXPONENT DIFFERENCE=1 FD=1
```

```
6114  027056                          CC25:
6115  027056  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
6116  027060  012704  003200                  MOV     #3200,R4        ;SET FIV,FIV, AND FD
6117  027064  170104                          LDFPS   R4
6118  027066  012737  027106  001236          MOV     #CC26,@#$TMP2
6119  027074  012700  027774                  MOV     #CCP0,R0        ;SET ACO OPERAND
6120  027100  172410                          LDD     (R0),AC0
6121  027102  012700  030024                  MOV     #CCP3,R0        ;FSRC
6122  027106  172010                  CC26:   ADDD    (R0),AC0        ;TEST INSTRUCTION
6123  027110  170205                          STFPS   R5              ;GET FPS
6124  027112  012700  027764                  MOV     #CCDAT0,R0      ;GET THE RESULT
6125  027116  174010                          STD     AC0,(R0)
6126  027120  012701  030104                  MOV     #CCP11,R1       ;IS IT CORRECT
6127  027124  012702  000004                  MOV     #4,R2
6128  027130  022021                  CC27:   CMP     (R0)+,(R1)+
6129  027132  001415                          BEQ     CC30
6130  027134  012700  027764                  MOV     #CCDAT0,R0      ;DID A BAD
6131  027140  012701  030024                  MOV     #CCP3,R1        ;CONSTANT (NOT 57,
6132  027144  012702  000004                  MOV     #4,R2           ;GET GENERATED
6133  027150  022021                  CC28:   CMP     (R0)+,(R1)+     ;FOR THE ALIGNMENT
6134  027152  001402                          BEQ     CC29            ;FLOWS?
6135  027154  000137  027674                  JMP     @#CCER10        ;DATA ERROR.D
6136  027160  077205                  CC29:   SOB     R2,CC28
6137  027162  000137  027712                  JMP     @#CCER11        ;BAD CONSTANT.D
6138  027166  077220                  CC30:   SOB     R2,CC27
6139  027170  020405                          CMP     R4,R5           ;FPS CORRECT?
6140  027172  001402                          BEQ     CC31
6141  027174  000137  027326                  JMP     @#CCER0         ;BAD FPS.
6142                          ;EXPONENT DIFFERENCE=100=144 (OCT) FD=1
6143  027200                  CC31:
6144  027200  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
6145  027202  012704  003200                  MOV     #3200,R4        ;SET FIV,FIV, AND FD
6146  027206  170104                          LDFPS   R4
6147  027210  012737  027230  001236          MOV     #CC32,@#$TMP2
6148  027216  012700  027774                  MOV     #CCP0,R0        ;SET ACO OPERAND
6149  027222  172410                          LDD     (R0),AC0
6150  027224  012700  030034                  MOV     #CCP4,R0        ;FSRC
6151  027230  172010                  CC32:   ADDD    (R0),AC0        ;TEST INSTRUCTION
6152  027232  170205                          STFPS   R5              ;GET FPS
6153  027234  012700  027764                  MOV     #CCDAT0,R0      ;GET THE RESULT
6154  027240  174010                          STD     AC0,(R0)
6155  027242  012701  030034                  MOV     #CCP4,R1        ;IS IT CORRECT
6156  027246  012702  000004                  MOV     #4,R2
6157  027252  022021                  CC33:   CMP     (R0)+,(R1)+
6158  027254  001415                          BEQ     CC36
6159  027256  012700  027764                  MOV     #CCDAT0,R0      ;DID A BAD
6160  027262  012701  030034                  MOV     #CCP4,R1        ;CONSTANT (NOT 57)
6161  027266  012702  000004                  MOV     #4,R2           ;GET GENERATED
6162  027272  022021                  CC34:   CMP     (R0)+,(R1)+     ;FOR THE ALIGNMENT
6163  027274  001402                          BEQ     CC35            ;FLOWS?
6164  027276  000137  027730                  JMP     @#CCER12        ;DATA ERROR.D
6165  027302  077205                  CC35:   SOB     R2,CC34
6166  027304  000137  027746                  JMP     @#CCER13        ;BAD CONSTANT.D
6167  027310  077220                  CC36:   SOB     R2,CC33
6168  027312  020405                          CMP     R4,R5           ;FPS CORRECT?
6169  027314  001402                          BEQ     CC37
```

```
6170  027316  000137  027326                    JMP     @#CCERO         ;BAD FPS.
6171  027322  000137  030124          CC37:     JMP     @#CCDONE
6172  027326  010437  001242          CCERO:    MOV     R4,@#$TMP4      ;FPS ERROR D
6173  027332  010537  001240                    MOV     R5,@#$TMP3
6174  027336  104164                  1$:       ERROR   164
6175  027340  000137  030124                    JMP     @#CCDONE
6176  027344  010437  001242          CCER90:   MOV     R4,@#$TMP4      ;FPS ERROR F
6177  027350  010537  001240                    MOV     R5,@#$TMP3
6178  027354  104165                  1$:       ERROR   165
6179  027356  000137  030124                    JMP     @#CCDONE
6180  027362  012737  030014  001240  CCER1:   MOV     #CCP2,@#$TMP3   ;DATA ERROR D
6181  027370  012737  030014  001246            MOV     #CCP2,@#$TMP6
6182  027376  012737  027774  001242  CCER50:  MOV     #CCP0,@#$TMP4
6183  027404  012737  027764  001244            MOV     #CCDAT0,@#$TMP5
6184  027412  104166                  1$:       ERROR   166
6185  027414  000157  030124                    JMP     @#CCDONE
6186  027420  012737  030014  001240  CCER2:   MOV     #CCP2,@#$TMP3   ;CONSTANT BAD D(B)
6187  027426  012737  030014  001246            MOV     #CCP2,@#$TMP6
6188  027434  012737  027774  001242  CCER22:  MOV     #CCP0,@#$TMP4
6189  027442  012737  027764  001244            MOV     #CCDAT0,@#$TMP5
6190  027450  104172                  1$:       ERROR   172
6191  027452  000137  030124                    JMP     @#CCDONE
6192  027456  012737  030004  001240  CCER3:   MOV     #CCP1,@#$TMP3
6193  027464  012737  030064  001246            MOV     #CCP7,@#$TMP6
6194  027472  000741                            BR      CCER50
6195  027474  012737  030004  001240  CCER4:   MOV     #CCP1,@#$TMP3   ;CONSTANT BAD D(G)
6196  027502  012737  030064  001246            MOV     #CCP7,@#$TMP6
6197  027510  012737  027774  001242  CCER44:  MOV     #CCP0,@#$TMP4
6198  027516  012737  027764  001244            MOV     #CCDAT0,@#$TMP5
6199  027524  104173                  1$:       ERROR   173
6200  027526  000137  030124                    JMP     @#CCDONE
6201  027532  012737  030054  001240  CCER5:   MOV     #CCP6,@#$TMP3   ;DATA ERROR F
6202  027540  012737  030054  001246            MOV     #CCP6,@#$TMP6
6203  027546  012737  027774  001242  CCER55:  MOV     #CCP0,@#$TMP4
6204  027554  012737  027764  001244            MOV     #CCDAT0,@#$TMP5
6205  027562  104170                  1$:       ERROR   170
6206  027564  000557                            BR      CCDONE
6207  027566  012737  030054  001240  CCER6:   MOV     #CCP6,@#$TMP3   ;CONSTANT BAD F(B)
6208  027574  012737  030054  001246            MOV     #CCP6,@#$TMP6
6209  027602  012737  027774  001242            MOV     #CCP0,@#$TMP4
6210  027610  012737  027764  001244            MOV     #CCDAT0,@#$TMP5
6211  027616  104174                  1$:       ERROR   174
6212  027620  000541                            BR      CCDONE
6213  027622  012737  030044  001240  CCER7:   MOV     #CCP5,@#$TMP3   ;DATA ERROR F
6214  027630  012737  030074  001246            MOV     #CCP10,@#$TMP6
6215  027636  000743                            BR      CCER55
6216  027640  012737  030044  001240  CCER8:   MOV     #CCP5,@#$TMP3   ;CONSTANT BAD F(G)
6217  027646  012737  030074  001246            MOV     #CCP10,@#$TMP6
6218  027654  012737  027764  001244            MOV     #CCDAT0,@#$TMP5
6219  027662  012737  027774  001242            MOV     #CCP0,@#$TMP4
6220  027670  104175                  1$:       ERROR   175
6221  027672  000514                            BR      CCDONE
6222  027674  012737  030024  001240  CCER10:  MOV     #CCP3,@#$TMP3   ;DATA ERROR D
6223  027702  012737  030104  001246            MOV     #CCP11,@#$TMP6
6224  027710  000632                            BR      CCER50
6225  027712  012737  030024  001240  CCER11:  MOV     #CCP3,@#$TMP3   ;CONSTANT BAD D(G)
```

```
6226  027722  012737  030104  001246  .       MOV     #CCP11,@#STMP6
6227  027726  000670                  BR      CCER44
6228  027730  012737  030034  001240  CCER12: MOV     #CCP4,@#STMP3   ;DATA ERROR D
6229  027736  012737  030034  001246          MOV     #CCP4,@#STMP6
6230  027744  000614                  BR      CCER50
6231  027746  012737  030034  001240  CCER13: MOV     #CCP4,@#STMP3   ;CONSTANT BAD D(B)
6232  027754  012737  030034  001246          MOV     #CCP4,@#STMP6
6233  027762  000624                  BR      CCER22
6234  027764  000000          CCDAT0: 0
6235  027766  000000                  0
6236  027770  000000                  0
6237  027772  030000                  0
6238  027774  000200          CCP0:   200                     ;E(AC)=1
6239  027776  000000                  0
6240  030000  00000C                  0
6241  030002  000000                  0
6242  030004  016200          CCP1:   16200                   ;E(FSRC)=E(AC)+56=57
6243  030006  000000                  0                       ;               =71(OCT)
6244  030010  000000                  0
6245  030012  000000                  0
6246  030014  016400          CCP2:   16400                   ;E(FSRC)=E(AC)+57=58
6247  030016  000000                  0                       ;               =72(OCT)
6248  030020  000000                  0
6249  030022  000000                  0
6250  030024  000400          CCP3:   400                     ;E(FSRC)=E(AC)+1=2
6251  030026  000000                  0
6252  030030  000000                  0
6253  030032  000000                  0
6254  030034  031200          CCP4:   31200                   ;E(FSRC)=E(AC)+100=101=145(OCT)
6255  030036  000000                  0
6256  030040  000000                  0
6257  030042  000000                  0
6258  030044  006200          CCP5:   6200                    ;E(FSRC)=E(AC)+24=25=31(OCT)
6259  030046  000000                  0
6260  030050  000000                  0
6261  030052  000000                  0
6262  030054  006400          CCP6:   6400                    ;E(FSRC)=E(AC)+25=26=32(OCT)
6263  030056  000000                  0
6264  030060  000000                  0
6265  030062  000000                  0
6266  030064  016200          CCP7:   16200                   ;CCP1 RES
6267  030066  000000                  0
6268  030070  000000                  0
6269  030072  000001                  1
6270  030074  006200          CCP10:  6200                    ;CCP5 RES
6271  030076  000001                  1
6272  030100  000000                  0
6273  030102  000000                  0
6274  030104  000500          CCP11:  500                     ;CCP3 RES
6275  030106  000000                  0
6276  030110  000000                  0
6277  030112  000000                  0
6278  030114  000200          CCP12:  200                     ;BAD CONSTANT
6279  030116  000000                  0                       ;RES CCP2,CCP4
6280  030120  000000                  0
6281  030122  000000                  0
```

```
6282
6283  030124                          CCDONE:
6284  030124  104412                          RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
6285                                                                  ;SEE IF THE USER HAS EXPRESSED
6286                                                                  ;THE DESIRE TO CHANGE THE SOFTWARE
6287                                                                  ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
6288                                                                  ;THE USER TYPED CONTROL G?).
6289
6290                                  ;;*****************************************************************
6291                                  ;*TEST 33       ADDF AND ADDD WITH E(AC) GREATER THAN E(FSRC) TEST
6292                                  ;*
6293                                  ;*THIS IS A TEST OF THE ADDD AND ADDF
6294                                  ;*INSTRUCTIONS AND THE ALIGN FSRC ALGORITHM
6295                                  ;*FLOWS.  FIRST THE CONSTANT USED IS CHECKED.
6296                                  ;*THEN SIMPLE AND WORST CASE ALIGNMENT
6297                                  ;*SITUATIONS ARE TRIED.  NOTE E(AC)
6298                                  ;*IS GREATER THAN E(FSRC).
6299                                  ;*
6300                                  ;;*****************************************************************
6301  030126  000004                  TST33:  SCOPE
6302                                  ;EXPONENT DIFFERENCE=57=71 (OCT) FD=1
6303  030130                          BB1:
6304  030130  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
6305  030132  012704  003200                  MOV     #3200,R4        ;SET FIV FIV, AND FD
6306  030136  170104                          LDFPS   R4
6307  030140  012737  030766  000244          MOV     #BBER0,@#FPVECT ;SET UP FOR ERROR
6308  030146  012737  030166  001236          MOV     #BB2,@#$TMP2    ;IN CASE THE OVER\
6309                                                                  ;UNDER FLOWS FAIL.
6310  030154  012700  031330                  MOV     #BBPAT2,R0      ;SET AC0 OPERAND.
6311  030160  172410                          LDD     (R0),AC0
6312  030162  012700  031320                  MOV     #BBPAT1,R0      ;FSRC
6313  030166  172010                  BB2:    ADDD    (R0),AC0        ;TEST INSTRUCTION
6314  030170  170205                          STFPS   R5
6315  030172  012700  031300          BB3:    MOV     #BBDAT0,R0      ;GET THE RESULT
6316  030176  174010                          STD     AC0,(R0)
6317  030200  012701  031330                  MOV     #BBPAT2,R1      ;RESULT CORRECT?
6318  030204  012702  000004                  MOV     #4,R2
6319  030210  022021                  BB4:    CMP     (R0)+,(R1)+
6320  030212  001402                          BEQ     BB5
6321  030214  000137  031026                  JMP     @#BBER1         ;DATA ERROR D
6322  030220  077205                  BB5:    SOB     R2,BB4
6323                                                                  ;WAS FPS CORRECT?
6324  030222  020405                          CMP     R4,R5
6325  030224  001402                          BEQ     BB6
6326  030226  000137  030766                  JMP     @#BBER0         ;FPS ERROR
6327                                  ;EXPONENT DIFFERENCE=56=70 (OCT) FD=1
6328  030232                          BB6:
6329  030232  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
6330  030234  012704  003200                  MOV     #3200,R4        ;SET FIV,FIV, AND FD
6331  030240  170104                          LDFPS   R4
6332  030242  012737  030262  001236          MOV     #BB7,@#$TMP2
6333  030250  012700  031350                  MOV     #BBPAT4,R0      ;SET AC0 OPERAND
6334  030254  172410                          LDD     (R0),AC0
6335  030256  012700  031320                  MOV     #BBPAT1,R0      ;FSRC
6336  030262  172010                  BB7:    ADDD    (R0),AC0        ;TEST INSTR' CTION
6337  030264  170205                          STFPS   R5              ;GET FPS
```

```
6338  030266  012700  031300              MOV     #BBDAT0,R0      ;GET THE RESULT
6339  030272  174010                      STD     AC0,(R0)
6340  030274  012701  031410              MOV     #BBP10,R1       ;IS IT CORRECT
6341  030300  012702  000004              MOV     #4,R2
6342  030304  022021              BB10:   CMP     (R0)+,(R1)+
6343  030306  001415                      BEQ     BB13
6344  030310  012700  031300              MOV     #BBDAT0,R0      ;DID A BAD
6345  030314  012701  031350              MOV     #BBPAT4,R1      ;CONSTANT (NOT 57)
6346  030320  012702  000004              MOV     #4,R2           ;GET GENERATED
6347  030324  022021              BB11:   CMP     (R0)+,(R1)+     ;FOR THE ALIGNMENT
6348  030326  001402                      BEQ     BB12            ;FLOWS?
6349  030330  000137  031064              JMP     @#BBER2         ;DATA ERROR.D
6350  030334  077205              BB12:   SOB     R2,BB11
6351  030336  000137  031102              JMP     @#BBER3         ;BAD CONSTANT.D
6352  030342  077220              BB13:   SOB     R2,BB10
6353  030344  020405                      CMP     R4,R5           ;FPS CORRECT?
6354  030346  001402                      BEQ     BB14
6355  030350  000137  030766              JMP     @#BBERD         ;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,@#STMP2
6414  030606  012704  003200              MOV     #3200,R4
6415  030612  170104                      LDFPS   R4              ;SET UP ACO 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,@#STMP2
6436  030702  012704  003200              MOV     #3200,R4
6437  030706  170104                      LDFPS   R4              ;SET FIV,FIV AND FD
6438  030710  012700  031370              MOV     #BBPAT6,R0      ;SET UP AC0 OPERAND.
6439  030714  172410                      LDD     (R0),AC0
6440  030716  012700  031320              MOV     #BBPAT1,R0      ;FSRC
6441  030722  172010              BB33:   ADDD    (R0),AC0        ;TEST INSTRUCTION
6442  030724  170205                      STFPS   R5
6443  030726  012700  031300              MOV     #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
```

```
                                BB35:   SOB     R2,BB34
                                        CMP     R4,R5           ;IS FPS CORRECT
                                        BNE     BBERO
                                        JMP     BBDONE
                                BBERO:  MOV     R4,@#$TMP4      ;FPS ERROR D
                                        MOV     R5,@#$TMP3
                                1$:     ERROR   164
                                        RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
                                                                ;SEE IF THE USER HAS EXPRESSED
                                                                ;THE DESIRE TO CHANGE THE SOFTWARE
                                                                ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
                                                                ;THE USER TYPED CONTROL G?).
                                        JMP     @#BBDONE
                                BBER1D: MOV     R4,@#$TMP4      ;FPS ERROR F
                                        MOV     R5,@#$TMP3
                                1$:     ERROR   165
                                        RSETUP                  ;GO INITIALIZE THE FPS AND STACK; AND
                                                                ;SEE IF THE USER HAS EXPRESSED
                                                                ;THE DESIRE TO CHANGE THE SOFTWARE
                                                                ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
                                                                ;THE USER TYPED CONTROL G?).
                                        JMP     @#BBDONE
                                BBER1:  MOV     #BBPAT2,@#$TMP4 ;DATA ERROR D
                                        MOV     #BBPAT2,@#$TMP6
                                BBER11: MOV     #BBPAT1,@#$TMP3
                                        MOV     #BBDAT0,@#$TMP5
                                1$:     ERROR   166
                                        JMP     @#BBDONE
                                BBER2:  MOV     #BBPAT4,@#$TMP4
                                        MOV     #BBP10,@#$TMP6
                                        BR      BBER11
                                BBER3:  MOV     #BBPAT4,@#$TMP4 ;BAD CONSTANT D
                                        MOV     #BBP10,@#$TMP6
                                        MOV     #BBPAT1,@#$TMP3
                                        MOV     #BBDAT0,@#$TMP5
                                1$:     ERROR   167
                                        BR      BBDONE
                                BBER4:  MOV     #BBPAT0,@#$TMP4 ;DATA ERROR F
                                        MOV     #BBPAT0,@#$TMP6
                                BBER40: MOV     #BBPAT1,@#$TMP3
                                        MOV     #BBDAT0,@#$TMP5
                                1$:     ERROR   170
                                        BR      BBDONE
                                BBER5:  MOV     #BBPAT3,@#$TMP4
                                        MOV     #BBP7,@#$TMP6
                                        BR      BBER40
                                BBER6:  MOV     #BBPAT3,@#$TMP4 ;CONSTANT ERROR F
                                        MOV     #BBP7,@#$TMP6
                                        MOV     #BBPAT1,@#$TMP3
                                        MOV     #BBDAT0,@#$TMP5
                                1$:     ERROR   171
                                        BR      BBDONE
                                BBER7:  MOV     #BBPAT5,@#$TMP4
                                        MOV     #BBPAT11,@#$TMP6
                                        BR      BBER11
                                BBER8:  MOV     #BBPAT6,@#$TMP4
```

```
031270  012737  031370  001246          MOV     #BBPAT6,@#STMP6
031276  000661                          BR      BBER11
031300  000000                  BBDAT0: 0
031302  000000                          0
031304  000000                          0
031306  000000                          0
031310  006400                  BBPAT0: 6400                    ;E(AC)=E(FSRC)+25=26
031312  000000                          0                       ;                 =32(OCT)
031314  000000                          0
031316  000000                          0
031320  000200                  BBPAT1: 200
031322  000000                          0                       ;E(FSRC)=1
031324  000000                          0
031326  000000                          0
031330  016400                  BBPAT2: 16400
031332  000000                          0                       ;E(AC)=E(FSRC)+57=58
031334  000000                          0                       ;                 =72(OCT)
031336  000000                          0
031340  006200                  BBPAT3: 6200                    ;E(AC)=E(FSRC)+24=25
031342  000000                          0                       ;                 =31(OCT)
031344  000000                          0
031346  000000                          0
031350  016200                  BBPAT4: 16200                   ;E(AC)=E(FSR6)+56=57
031352  000000                          0                       ;                 =71(OCT)
031354  000000                          0
031356  000000                          0
031360  000400                  BBPAT5: 400                     ;E(AC)=E(FSRC)+1=2
031362  000000                          0
031364  000000                          0
031366  000000                          0
031370  031200                  BBPAT6: 31200                   ;E(AC)=E(FSRC)+100=101
031372  000000                          0                       ;                   =145(OCT)
031374  000000                          0
031376  000000                          0
031400  006200                  BBP7:   6200                    ;BBPAT3 RES
031402  000001                          1
031404  000000                          0
031406  000000                          0
031410  016200                  BBP10:  16200                   ;BBPAT4 RES
031412  000000                          0
031414  000000                          0
031416  000001                          1
031420  000500                  BBP11:  500                     ;BBPAT5 RES
031422  000000                          0
031424  000000                          0
031426  000000                          0
031430                          BBDONE:
031430  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 34        ADDD WITH NEGATVE OPRANDS TEST
                                ;*
                                ;*THIS IS A TEST OF THE ADDD INSTRUCTION
```

```
                                        ;*WITH NEGATIVE OPERANDS.  EVERY COMBINATION OF
                                        ;*OPERAND SIGNS IS TRIED.
                                        ;*
                                        ;;*********************************************************
 031432  000034                         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 AC0 OPERAND
 031456  172410                                 LDD     (R0),AC0
 031460  012700  033314                         MOV     #DDP1,R0                        ;ESRC
 031464  172010                         DD2:    ADDD    (R0),AC0                        ;TEST INSTRUCTION
 031466  170205                                 STFPS   R5                              ;GET FPS
 031470  012700  033274                         MOV     #DDDAT0,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     #DDDAT0,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 AC0 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     #DDDAT0,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     @#DDERO                         ;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,@#STMP2
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     #DDDATO,R0                      ;GET THE RESULT
031722  174010                          STD     AC0,(R0)
031724  012701  033304                  MOV     #DDP0,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     @#DDERO                         ;BAD FPS
                                ;AC0 POC        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,@#STMP2
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     #DDDATO,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     #DDDATO,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     @#DDERO                         ;BAD FPS
                                ;AC NEG FSRC    POS     /FSRC/ > /AC/
032104                          DD22:
```

```
6674  032124  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
6675  032126  012704  003200                  MOV     #3200,R4        ;SET FIU,FIV, AND FD
6676  032132  170104                          LDFPS   R4
6677  032134  012737  032134  001236          MOV     #DD23,@#STMP2
6678  032142  012700  033364                  MOV     #DDP6,R0        ;SET ACO OPERAND
6679  032146  172410                          LDD     (R0),AC0
6680  032150  012700  033334                  MOV     #DDP3,R0        ;FSRC
6681  032154  172010          DD23:           ADDD    (R0),AC0        ;TEST INSTRUCTION
6682  032156  170205                          STFPS   R5              ;GET FPS
6683  032160  012700  033274                  MOV     #DDDAT0,R0      ;GET THE RESULT
6684  032164  174010                          STD     AC0,(R0)
6685  032166  012701  033374                  MOV     #DDP7,R1        ;IS IT CORRECT?
6686  032172  012702  000004                  MOV     #4,R2
6687  032176  022021          DD24:           CMP     (R0)+,(R1)+
6688  032200  001415                          BEQ     DD27
6689  032202  012700  033274                  MOV     #DDDAT0,R0      ;FLO,S FAILURE
6690  032206  012701  033404                  MOV     #DDP8,R1        ;CONSTANT (NOT 57,
6691  032212  012702  000004                  MOV     #4,R2           ;216.042,101
6692  032216  021011          DD25:           CMP     (R0),(R1)
6693  032220  001402                          BEQ     DD26
6694  032222  000137  033010                  JMP     @#DDER7         ;DATA ERROR.
6695  032226  077205          DD26:           SOB     R2,DD25
6696  032230  000137  033046                  JMP     @#DDER8
6697  032234  077220          DD27:           SOB     R2,DD24
6698  032236  020405                          CMP     R4,R5           ;FPS CORRECT?
6699  032240  001402                          BEQ     DD30
6700  032242  000137  032506                  JMP     @#DDER0         ;BAD FPS
6701                          ;AC0 POS        FSRC    NEG     ;AC/(.FRSRC
6702  032246                  DD30:
6703  032246  104413                          LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
6704  032250  012704  003200                  MOV     #3200,R4        ;SET FIC,FIV,AND FD
6705  032254  170104                          LDFPS   R4
6706  032256  012737  032256  001236          MOV     #DD31,@#STMP2
6707  032264  012700  033344                  MOV     #DDP4,R0        ;SET ACO OPERAND
6708  032270  172410                          LDD     (R0),AC0
6709  032272  012700  033354                  MOV     #DDP5,R0        ;FSRC
6710  032276  172010          DD31:           ADDD    (R0),AC0        ;TEST INSTRUCTION
6711  032300  170205                          STFPS   R5              ;GET FPS
6712  032302  012700  033274                  MOV     #DDDAT0,R0      ;GET THE RESULT
6713  032306  174010                          STD     AC0,(R0)
6714  032310  012701  033404                  MOV     #DDP8,R1        ;IS IT CORRECT
6715  032314  012702  000004                  MOV     #4,R2
6716  032320  022021          DD32:           CMP     (R0)+,(R1)+
6717  032322  001415                          BEQ     DD35            ;ADD-SUB
6718  032324  012700  033274                  MOV     #DDDAT0,R0      ;FLOWAS FAILURE
6719  032330  012701  033374                  MOV     #DDP7,R1        ;CON 216 N440 NOT 141
6720  032334  012702  000004                  MOV     #4,R2           ;GET GENERATED
6721  032340  022021          DD33:           CMP     (R0)+,(R1)+     ;FOR THE ALLIGNMENT
6722  032342  001402                          BEQ     DD34            ;FLOWS?
6723  032344  000137  033104                  JMP     @#DDER9         ;DATA ERROR, D
6724  032350  077205          DD34:           SOB     R2,DD33
6725  032352  000137  033142                  JMP     @#DDER10
6726  032356  077220          DD35:           SOB     R2,DD32
6727  032360  052704  000010                  BIS     #10,R4
6728  032364  020405                          CMP     R4,R5           ;FPS CORRECT?
6729  032366  001402                          BEQ     #DD36
```

```
032360  000137  032506                JMP     @#DDERO                 ;BAD FPS
                                ;ACO 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,@#$TMP2
032372  012700  033354                MOV     #DDP5,R0                ;SET ACO OPERAND
032376  172410                        LDD     (R0),AC0
032400  012700  033344                MOV     #DDP4,R0                ;FSPC
032404  172010          DD37:         ADDD    (R0),AC0                ;TEST INSTRUCTION
032406  170205                        STFPS   R5                      ;GET FPS
032410  012700  033274                MOV     #DDDAT0,R0              ;GET THE RESULT
032414  174010                        STD     AC0,(R0)
032416  012701  033404                MOV     #DDP8,R1                ;IS IT CORRECT
032422  012702  000004                MOV     #4,R2
032426  022021          DD38:         CMP     (R0)+,(R1)+
032430  001415                        BEQ     DD41
032432  012700  033274                MOV     #DDDAT0,R0              ;ADD SUB
032436  012701  033374                MOV     #DDP7,R1                ;FLOWS FAILURES
032442  012702  000004                MOV     #4,R2                   ;GET 216,042,141
032446  022021          DD39:         CMP     (R0)+,(R1)+             ;FOR THE ALLIGNMENT
032450  001402                        BEQ     DD40                    ;FLOWS?
032452  000137  033200                JMP     @#DDER11                ;DATA ERROR. D
032456  077205          DD40:         SOB     R2,DD39
032460  000137  033236                JMP     @#DDER12                ;BAD CONSTANT.D
032464  077220          DD41:         SOB     R2,DD38
032466  052704  000010                BIS     #10,R4
032472  020405                        CMP     R4,R5                   ;FPS CORRECT?
032474  001402                        BEQ     DD42
032476  000137  032506                JMP     @#DDERO                 ;BAD FPS
032502  000137  033424          DD42: JMP     @#DDDONE
032506  010437  001242          DDERO: MOV    R4,@#$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
```

```
 6796  032652  000137  033424                    JMP     @#DDDONE
 6797  032656                            DDER4:
 6798  032656  012737  033324  001240            MOV     #DDP2,@#$TMP3
 6799  032664  012737  033314  001242            MOV     #DDP1,@#$TMP4
 6800  032672  012737  033274  001244            MOV     #DDDAT0,@#$TMP5
 6801  032700  012737  033304  001246            MOV     #DDP0,@#$TMP6
 6802  032706  104200                    1$:     ERROR   200
 6803  032710  000137  033424                    JMP     @#DDDONE
 6804  032714                            DDER5:
 6805  032714  012737  033334  001240            MOV     #DDP6,@#$TMP3
 6806  032722  012737  033334  001242            MOV     #DDP3,@#$TMP4
 6807  032730  012737  033274  001244            MOV     #DDDAT0,@#$TMP5
 6808  032736  012737  033374  001246            MOV     #DDP7,@#$TMP6
 6809  032744  104165                    1$:     ERROR   165
 6810  032746  000137  033424                    JMP     @#DDDONE
 6811  032752                            DDER6:
 6812  032752  012737  033364  001240            MOV     #DDP6,@#$TMP3
 6813  032760  012737  033334  001242            MOV     #DDP3,@#$TMP4
 6814  032766  012737  033274  001244            MOV     #DDDAT0,@#$TMP5
 6815  032774  012737  033374  001246            MOV     #DDP7,@#$TMP6
 6816  033002  104201                    1$:     ERROR   201
 6817  033004  000137  033424                    JMP     @#DDDONE
 6818  033010                            DDER7:
 6819  033010  012737  033334  001240            MOV     #DDP3,@#$TMP3
 6820  033016  012737  033364  001242            MOV     #DDP6,@#$TMP4
 6821  033024  012737  033274  001244            MOV     #DDDAT0,@#$TMP5
 6822  033032  012737  033374  001246            MOV     #DDP7,@#$TMP6
 6823  033040  104165                    1$:     ERROR   165
 6824  033042  000137  033424                    JMP     @#DDDONE
 6825  033046                            DDER8:
 6826  033046  012737  033334  001240            MOV     #DDP3,@#$TMP3
 6827  033054  012737  033364  001242            MOV     #DDP6,@#$TMP4
 6828  033062  012737  033274  001244            MOV     #DDDAT0,@#$TMP5
 6829  033070  012737  033374  001246            MOV     #DDP7,@#$TMP6
 6830  033076  104202                    1$:     ERROR   202
 6831  033103  000137  033424                    JMP     @#DDDONE
 6832  033104                            DDER9:
 6833  033104  012737  033354  001240            MOV     #DDP5,@#$TMP3
 6834  033112  012737  033344  001242            MOV     #DDP4,@#$TMP4
 6835  033120  012737  033274  001244            MOV     #DDDAT0,@#$TMP5
 6836  033126  012737  033404  001246            MOV     #DDP8,@#$TMP6
 6837  033134  104165                    1$:     ERROR   165
 6838  033136  000137  033424                    JMP     @#DDDONE
 6839  033142                            DDER10:
 6840  033142  012737  033354  001240            MOV     #DDP5,@#$TMP3
 6841  033150  012737  033344  001242            MOV     #DDP4,@#$TMP4
 6842  033156  012737  033274  001244            MOV     #DDDAT0,@#$TMP5
 6843  033164  012737  033404  001246            MOV     #DDP8,@#$TMP6
 6844  033172  104203                    1$:     ERROR   203
 6845  033174  000137  033424                    JMP     @#DDDONE
 6846  033200                            DDER11:
 6847  033200  012737  033344  001240            MOV     #DDP4,@#$TMP3
 6848  033206  012737  033354  001242            MOV     #DDP5,@#$TMP4
 6849  033214  012737  033274  001244            MOV     #DDDAT0,@#$TMP5
 6850  033222  012737  033404  001246            MOV     #DDP8,@#$TMP6
 6851  033230  104165                    1$:     ERROR   165
```

```
 6842  033232  000137  033424                 JMP     @#DDDONE
 6843  033236                         DDER12:
 6844  033236  012737  033344  001240         MOV     #DDP4,@#$TMP3
 6845  033244  012737  033354  001242         MOV     #DDP5,@#$TMP4
 6846  033252  012737  033274  001244         MOV     #DDDAT0,@#$TMP5
 6847  033260  012737  033404  001246         MOV     #DDP8,@#$TMP6
 6848  033266  104204                 1$:     ERROR   204
 6849  033270  000137  033424                 JMP     @#DDDONE
 6850  033274  000000                 DDDAT0: 0
 6851  033276  000000                         0
 6852  033300  000000                         0
 6853  033302  000000                         0
 6854  033304  000000                 DDPC:   0
 6855  033306  000000                         0
 6856  033310  000000                         0
 6857  033312  000000                         0
 6858  033314  100200                 DDP1:   100200                          ;-DDP2
 6859  033316  000000                         0
 6860  033320  000000                         0
 6861  033322  000000                         0
 6862  033324  000200                 DDP2:   200                             ;-DDP1
 6863  033326  000000                         0
 6864  033330  000000                         0
 6865  033332  000000                         0
 6866  033334  001100                 DDP3:   1100                            ;EXP=4
 6867  033336  000000                         0                               ;FRAC=...110...
 6868  033340  000000                         0
 6869  033342  000000                         0
 6870  033344  000600                 DDP4:   600                             ;EXP=3
 6871  033346  000000                         0                               ;FRAC=...100...
 6872  033350  000000                         0
 6873  033352  000000                         0
 6874  033354  101100                 DDP5:   101100                          ;-DDP3
 6875  033356  000000                         0
 6876  033360  000000                         0
 6877  033362  000000                         0
 6878  033364  100600                 DDP6:   100600                          ;-DDP4
 6879  033366  000000                         0
 6880  033370  000000                         0
 6881  033372  000000                         0
 6882  033374  001000                 DDP7:   1000                            ;DDP3+DDP6
 6883  033376  000000                         0
 6884  033400  000000                         0
 6885  033402  000000                         0
 6886  033404  101000                 DDP8:   101000                          ;DDP5+DDP4
 6887  033406  000000                         0
 6888  033410  000000                         0
 6889  033412  000000                         0
 6890  033414  100400                 DDP9:   100400                          ;DDP1+DDP1
 6891  033416  000000                         0
 6892  033420  000000                         0
 6893  033422  000000                         0
 6894  033424  000005                 DDDONE:         RESET
 6895                                 ;;*************************************************************
 6896                                 ;*TEST 35        SUBD TEST
 6897                                 ;*
```

```
                                        ;* THIS IS A TEST OF THE SUBD INSTRUCTION.
                                        ;* BOTH A POSITIVE AND A NEGATIVE NUMBER
                                        ;* IS SUBTRACTED FROM IT SELF
                                        ;*
                                        ;;*****************************************************************
 033426  000004                         TST35:  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 AC0 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 AC0 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
```

```
6954  033650  022021                    EE10:   CMP     (R0)+,(R1)+             ;FOR THE ALLIGNMENT
6955  033652  001402                            BEQ     EE11                   ;FLOWS?
6956  033654  000137  034022                    JMP     @#EEER3                ;DATA ERROR.D
6957  033660  077205                    EE11:   SOB     R2,EE10
6958  033662  000137  034060                    JMP     @#EEER4                ;BAD CONSTANT.D
6959  033666  077220                    EE12:   SOB     R2,EE9
6960  033670  052704  000004                    BIS     #4,R4
6961  033674  020405                            CMP     R4,R5                  ;FPS CORRECT?
6962  033676  001402                            BEQ     EE13
6963  033700  000137  033710                    JMP     @#EEER0                ;BAD FPS.
6964  033704  000137  034200            EE13:   JMP     @#EEDONE
6965  033710  010437  001242            EEER0:  MOV     R4,@#$TMP4             ;BAD FPS
6966  033714  010537  001240                    MOV     R5,@#$TMP3
6967  033720  104205                    1$:     ERROR   205
6968  033722  000137  034200                    JMP     @#EEDONE
6969  033726                            EEER1:
6970  033726  012737  034140  001240            MOV     #EEP1,@#$TMP3
6971  033734  012737  034140  001242            MOV     #EEP1,@#$TMP4
6972  033742  012737  034116  001244            MOV     #EEDAT0,@#$TMP5
6973  033750  012737  034126  001246            MOV     #EEP0,@#$TMP6
6974  033756  104206                    1$:     ERROR   206
6975  033760  000137  034200                    JMP     @#EEDONE
6976  033764                            EEER2:
6977  033764  012737  034140  001240            MOV     #EEP1,@#$TMP3
6978  033772  012737  034140  001242            MOV     #EEP1,@#$TMP4
6979  034000  012737  034116  001244            MOV     #EEDAT0,@#$TMP5
6980  034006  012737  034126  001246            MOV     #EEP0,@#$TMP6
6981  034014  104207                    1$:     ERROR   207
6982  034016  000137  034200                    JMP     @#EEDONE
6983  034022                            EEER3:
6984  034022  012737  034160  001240            MOV     #EEP3,@#$TMP3
6985  034030  012737  034160  001242            MOV     #EEP3,@#$TMP4
6986  034036  012737  034116  001244            MOV     #EEDAT0,@#$TMP5
6987  034044  012737  034126  001246            MOV     #EEP0,@#$TMP6
6988  034052  104206                    1$:     ERROR   206
6989  034054  000137  034200                    JMP     @#EEDONE
6990  034060                            EEER4:
6991  034060  012737  034160  001240            MOV     #EEP3,@#$TMP3
6992  034066  012737  034160  001242            MOV     #EEP3,@#$TMP4
6993  034074  012737  034116  001244            MOV     #EEDAT0,@#$TMP5
6994  034102  012737  034126  001246            MOV     #EEP0,@#$TMP6
6995  034110  104207                    1$:     ERROR   207
6996  034112  000137  034200                    JMP     @#EEDONE
6997  034116  000000                    EEDAT0: 0
6998  034120  000000                            0
6999  034122  000000                            0
7000  034124  000000                            0
7001  034126  000000                    EEP0:   0
7002  034130  000000                            0
7003  034132  000000                    00000
7004  034134  000000                            0
7005  034136  000000                            0
7006  034140  000200                    EEP1:   200
7007  034142  000000                            0
7008  034144  000000                            0
7009  034146  000000                            0
```

```
 7010  034150  000400                EEP2:   400
 7011  034152  000000                        0
 7012  034154  000000                        0
 7013  034156  000000                        0
 7014  034160  100200                EEP3:   100200
 7015  034162  000000                        0
 7016  034164  000000                        0
 7017  034166  000000                        0
 7018  034170  100400                EEP4:   100400
 7019  034172  000000                        0
 7020  034174  000000                        0
 7021  034176  000000                        0
 7022  034200                        EECONE:
 7023  034200  104412                        RSETJP                  ;GO INITIALIZE THE FPS AND STACK; AND
 7024                                                                ;SEE IF THE USER HAS EXPRESSED
 7025                                                                ;THE DESIRE TO CHANGE THE SOFTWARE
 7026                                                                ;VIRTUAL CONSOLE SWITCH REGISTER (HAS
 7027                                                                ;THE USER TYPED CONTROL G?).
 7028                                ;;*****************************************************************
 7029                                ;*TEST 36       NORMALIZE ALGORITHM TEST
 7030                                ;*
 7031                                ;* THIS IS A TEST OF THE NORMALIZE
 7032                                ;* FLOW ALGORITHM.  TWO PATTERNS ARE USED,
 7033                                ;* FIRST THE MINIMUM SITUATION REQUIRING ONE
 7034                                ;* LEFT SHIFT AND THEN THE MAXIMUM SITUATION
 7035                                ;* REQUIRING 56 SHIFTS.
 7036                                ;*
 7037                                ;;*****************************************************************
 7038  034202  000004                ST36:   SCOPE
 7039                                ;USE DATA PATTERNS THAT REQUIRE ONLY ONE LEFT SHIFT TO NORMALIZE
 7040  034204                        FF1:
 7041  034204  104413                        LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
 7042  034206  012704  003200                MOV     #3200,R4                ;SET FIU, FIV, AND FD
 7043  034212  170104                        LDFPS   R4
 7044  034214  012737  034234  001236        MOV     #FF2,@#$TMP2
 7045  034222  012700  034526                MOV     #FFP2,R0                ;SET ACO OPERAND
 7046  034226  172410                        LDD     (R0),AC0
 7047  034230  012700  034536                MOV     #FFP3,R0                ;FSPC
 7048  034234  172010                FF2:    ADDD    (R0),AC0                ;TEST INSTRUCTION
 7049  034236  170205                        STFPS   R5                      ;GET FPS
 7050  034240  012700  034476                MOV     #FFDAT0,R0              ;GET THE RESULT
 7051  034244  174010                        STD     AC0,(R0)
 7052  034246  012701  034546                MOV     #FFP4, R1               ;IS IT CORRECT
 7053  034252  012702  000004                MOV     #4,R2
 7054  034256  022021                FF3:    CMP     (R0)+,(R1)+
 7055  034260  001401                        BEQ     FF4
 7056  034262  000466                        BR      FFER2                   ;BAD DATA
 7057  034264  077204                FF4:    SOB     R2,FF3
 7058  034266  020405                        CMP     R4,R5                   ;FPS CORRECT?
 7059  034270  001401                        BEQ     FF5
 7060  034272  000435                        BR      FFER0                   ;BAD FPS
 7061                                ;LSE DATA PATTERNS WHICH REQUIRE 56 LEFT SHIFTS TO NORMALIZE
 7062                                ;THE RESULT
 7063  034274                        FF5:
 7064  034274  104413                        LPERR                   ;SET UP THE LOOP ON ERROR ADDRESS.
 7065  034276  012704  003200                MOV     #3200,R4                ;SET FIU, FIV, AND FD
```

```
 7066  034302  170104                          LDFPS   R4
 7067  034304  012737  034324  001236          MOV     #FF6,@#$TMP2
 7068  034312  012700  034506                  MOV     #FFP0,R0                ;SET ACO OPERAND
 7069  034316  172410                          LDD     (R0),AC0
 7070  034320  012700  034516                  MOV     #FFP1,R0                ;FSRC
 7071  034324  172010                  FF6:    ADDD    (R0),AC0                ;TEST INSTRUCTION
 7072  034326  170205                          STFPS   R5                      ;GET FPS
 7073  034330  012700  034476                  MOV     #FFDAT0,R0              ;GET THE RESULT
 7074  034334  174010                          STD     AC0,(R0)
 7075  034336  012701  034546                  MOV     #FFP4,R1                ;IS IT CORRECT
 7076  034342  012702  000004                  MOV     #4,R2
 7077  034346  022021                  FF7:    CMP     (R0)+,(R1)+
 7078  034350  001401                          BEQ     FF10
 7079  034352  000413                          BR      FFER1                   ;BAD A
 7080  034354  077204                  FF10:   SOB     R2,FF7
 7081  034356  020405                          CMP     R4,R5                   ;FPS CORRECT
 7082  034360  001401                          BEQ     FF11
 7083  034362  000401                          BR      FFER0           ;BAD FPS
 7084  034364  000474                  FF11:   BR      FFDONE
 7085
 7086  034366  010537  001240          FFER0:  MOV     R5,@#$TMP3
 7087  034372  010437  001242                  MOV     R4,@#$TMP4
 7088  034376  104164                  1$:     ERROR   164
 7089  034400  000466                          BR      FFDONE
 7090
 7091  034402                          FFER1:
 7092  034402  012737  034516  001240                  MOV     #FFP1,@#$TMP3
 7093  034410  012737  034506  001242                  MOV     #FFP0,@#$TMP4
 7094  034416  012737  034476  001244                  MOV     #FFDAT0,@#$TMP5
 7095  034424  012737  034546  001246                  MOV     #FFP4,@#$TMP6
 7096  034432  104210                  1$:     ERROR   210
 7097  034434  000137  034556                  JMP     @#FFDONE
 7098
 7099  034440                          FFER2:
 7100  034440  012737  034536  001240                  MOV     #FFP3,@#$TMP3
 7101  034446  012737  034526  001242                  MOV     #FFP2,@#$TMP4
 7102  034454  012737  034476  001244                  MOV     #FFDAT0,@#$TMP5
 7103  034462  012737  034546  001246                  MOV     #FFP4,@#$TMP6
 7104  034470  104210                  1$:     ERROR   210
 7105  034472  000137  034556                  JMP     @#FFDONE
 7106
 7107
 7108  034476  000000                  FFDAT0: 0
 7109  034500  000000                          0
 7110  034502  000000                          0
 7111  034504  000000                          0
 7112
 7113  034506  016000                  FFP0:   16000
 7114  034510  000000                          0
 7115  034512  000000                          0
 7116  034514  000001                          1
 7117  034516  116000                  FFP1:   116000
 7118  034520  000000                          0
 7119  034522  000000                          0
 7120  034524  000000                          0
 7121  034526  000500                  FFP2:   500
```

```
7122  034530  000000                  0
7123  034532  000000                  0
7124  034534  000000                  0
7125  034536  100400          FFP2:   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  001435                        BEQ     $DOAGN          ;;RETURN ADDRESS
 7191  034766  000005                        RESET                   ;;CLEAR THE WORLD
 7192  034770  004710                $SENDAD: 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.(DISPLAY<7:0>)
 7220                                ;*AND LOAD THE ERROR FLAG ($ERFLG) INTO DISPLAY<15:08>
 7221                                ;*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
 7222                                ;*SW14=1         LOOP ON TEST
 7223                                ;*SW11=1         INHIBIT ITERATIONS
 7224                                ;*SW09=1         LOOP ON ERROR
 7225                                ;*SW08=1         LOOP ON TEST IN SWR<7:0>
 7226                                ;*CALL
 7227                                ;*      SCOPE           ;;SCOPE=IOT
 7228
 7229  035050                        $SCOPE:
 7230  035050  104406                        CKSWR                   ;;TEST  FOR CHANGE IN SOFT-SWR
 7231  035052  032777  040000  144060  1$:     BIT     #BIT14,@SWR     ;;LOOP ON PRESENT TEST?
 7232  035060  001114                        BNE     $OVER           ;;YES IF SW14=1
 7233                                ;*****START OF CODE FOR THE XOR TESTER*****
```

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

```
                                        ;*SW09=1        LOOP ON ERROR
                                        ;CALL
                                        ;*      ERROR   N       ;;ERROR=EMT AND N=ERROR ITEM NUMBER

                                        $ERROR:
                                                CKSWR                   ;;TEST FOR CHANGE IN SOFT-SWR
                                        7$:     INCB    $ERFLG          ;;SET THE ERROR FLAG
                                                BEQ     7$              ;;DON'T LET THE FLAG GO TO ZERO
                                                MOV     $STNM,@DISPLAY  ;;DISPLAY TEST NUMBER AND ERROR FLAG
                                                BIT     #BIT10,@SWR     ;;BELL ON ERROR?
                                                BEQ     1$              ;;NO - SKIP
                                                TYPE    ,$BELL          ;;RING BELL
                                        1$:     INC     $ERTTL          ;;COUNT THE NUMBER OF ERRORS
                                                MOV     (SP),$ERRPC     ;;GET ADDRESS OF ERROR INSTRUCTION
                                                SUB     #2,$ERRPC
                                                MOVB    @$ERRPC,$ITEMB  ;;STRIP AND SAVE THE ERROR ITEM CODE
                                                BIT     #BIT13,@SWR     ;;SKIP TYPEOUT IF SET
                                                BNE     20$             ;;SKIP TYPEOUTS
                                                JSR     PC,ERTYPE       ;;GO TO USER ERROR ROUTINE
                                                TYPE    ,$CRLF
                                        20$:
                                                CMPB    #APTENV,$ENV    ;;RUNNING IN APT MODE
                                                BNE     2$              ;;NO,SKIP APT ERROR REPORT
                                                MOVB    $ITEMB,21$      ;;SET ITEM NUMBER AS ERROR NUMBER
                                                JSR     PC,$ATY4        ;;REPORT FATAL ERROR TO APT
                                        21$:    .BYTE   0
                                                .BYTE   0
                                        22$:    BR      22$             ;;APT ERROR LOOP
                                        2$:     TST     @SWR            ;;HALT ON ERROR
                                                BPL     3$              ;;SKIP IF CONTINUE
                                                HALT                    ;;HALT ON ERROR!
                                                CKSWR                   ;;TEST FOR CHANGE IN SOFT-SWR
                                        3$:     BIT     #BIT09,@SWR     ;;LOOP ON ERROR SWITCH SET?
                                                BEQ     4$              ;;BR IF NO
                                                MOV     $LPERR,(SP)     ;;FUDGE RETURN FOR LOOPING
                                        4$:     TST     $ESCAPE         ;;CHECK FOR AN ESCAPE ADDRESS
                                                BEQ     5$              ;;BR IF NONE
                                                MOV     $ESCAPE,(SP)    ;;FUDGE RETURN ADDRESS FOR ESCAPE
                                        5$:
                                                CMP     #$ENDAD,@#42    ;;ACT-11 AUTO-ACCEPT?
                                                BNE     6$              ;;BRANCH IF NO
                                                HALT                    ;;YES
                                        6$:
                                                BIT     #BIT09,@SWR
                                                BNE     ERM10
                                                MOV     (SP),@#$REGO            ;SEE IF ERROR #377
                                                ADD     #-2,@#$REGO
                                                CMPB    #377,@$REGO
                                                BNE     ERM10
                                                ADD     #2,(SP)
                                        ERM10:  RTI

                                        .SBTTL  SAVE AND RESTORE R0-R5 ROUTINES

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

```
                                        ;*SAVE R0-R5
                                        ;*CALL:
                                        ;*      SAVREG
                                        ;*UPON RETURN FROM $SAVREG THE STACK WILL LOOK LIKE:
                                        ;*
                                        ;*TOP---(+16)
                                        ;* +2---(+18)
                                        ;* +4---R5
                                        ;* +6---R4
                                        ;* +8---R3
                                        ;*+10---R2
                                        ;*+12---R1
                                        ;*+14---R0

035566                                  $SAVREG:
035566  010046                                  MOV     R0,-(SP)        ;;PUSH R0 ON STACK
035570  010146                                  MOV     R1,-(SP)        ;;PUSH R1 ON STACK
035572  010246                                  MOV     R2,-(SP)        ;;PUSH R2 ON STACK
035574  010346                                  MOV     R3,-(SP)        ;;PUSH R3 ON STACK
035576  010446                                  MOV     R4,-(SP)        ;;PUSH R4 ON STACK
035600  010546                                  MOV     R5,-(SP)        ;;PUSH R5 ON STACK
035602  016646  000022                          MOV     22(SP),-(SP)    ;;SAVE PS OF MAIN FLOW
035606  016646  000022                          MOV     22(SP),-(SP)    ;;SAVE PC OF MAIN FLOW
035612  016646  000022                          MOV     22(SP),-(SP)    ;;SAVE PS OF CALL
035616  016646  000022                          MOV     22(SP),-(SP)    ;;SAVE PC OF CALL
035622  000002                                  RTI

                                        ;*RESTORE R0-R5
                                        ;*CALL:
                                        ;*      RESREG
035624                                  $RESREG:
035624  012666  000022                          MOV     (SP)+,22(SP)    ;;RESTORE PC OF CALL
035630  012666  000022                          MOV     (SP)+,22(SP)    ;;RESTORE PS OF CALL
035634  012666  000022                          MOV     (SP)+,22(SP)    ;;RESTORE PC OF MAIN FLOW
035640  012666  000022                          MOV     (SP)+,22(SP)    ;;RESTORE PS OF MAIN FLOW
035644  012605                                  MOV     (SP)+,R5        ;;POP STACK INTO R5
035646  012604                                  MOV     (SP)+,R4        ;;POP STACK INTO R4
035650  012603                                  MOV     (SP)+,R3        ;;POP STACK INTO R3
035652  012602                                  MOV     (SP)+,R2        ;;POP STACK INTO R2
035654  012601                                  MOV     (SP)+,R1        ;;POP STACK INTO R1
035656  012600                                  MOV     (SP)+,R0        ;;POP STACK INTO R0
035660  000002                                  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:         $FILLC CONTAINS THE CHARACTER TO FILL AFTER.
                                        ;*
                                        ;*CALL:
                                        ;*1) USING A TRAP INSTRUCTION
                                        ;*      TYPE    ,MESADR         ;;MESADR IS FIRST ADDRESS OF AN ASCIZ STRING
                                        ;*OR
```

```
                                        ;*      TYPE
                                        ;*      MESADR
                                        ;*
                                        ;
035662  105767  143271          $TYPE:  TSTB    $TPFLG          ;;IS THERE A TERMINAL?
035666  100002                          BPL     :$              ;;BR IF YES
035670  000000                          HALT                    ;;HALT HERE IF NO TERMINAL
035672  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     6C$             ;;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                  6C$:    MOV     (SP)+,R0        ;;RESTORE R0
035754  062716  000002          3$:     ADD     #2,(SP)         ;;ADJUST RETURN PC
035760  000002                          RTI                     ;;RETURN
035762  122716  000011          4$:     CMPB    #HT,(SP)        ;;BRANCH IF <HT>
035766  001430                          BEQ     8$
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.
```

```
 7458  036110  122766  000015  000002          CMPB    #CR,2(SP)       ;;IS CHARACTER A CARRIAGE RETURN?
 7459  036116  001003                          BNE     1$              ;;BRANCH IF NO
 7460  036120  105067  000014                  CLRB    $CHARCNT        ;;YES--CLEAR CHARACTER COUNT
 7461  036124  000406                          BR      $TYPEX          ;;EXIT
 7462  036126  122766  000012  000002  1$:     CMPB    #LF,2(SP)       ;;IS CHARACTER A LINE FEED?
 7463  036134  001402                          BEQ     $TYPEX          ;;BRANCH IF YES
 7464  036136  105227                          INCB    (PC)+           ;;COUNT THE CHARACTER
 7465  036140  000000                $CHARCNT:.WORD  0               ;;CHARACTER COUNT STORAGE
 7466  036142  000207                $TYPEX: RTS     PC
 7467
 7468
 7469                                .SBTTL  BINARY TO OCTAL (ASCII) AND TYPE
 7470
 7471                                ;;**************************************************************
 7472                                ;*THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 6-DIGIT
 7473                                ;*OCTAL (ASCII) NUMBER AND TYPE IT.
 7474                                ;*$TYPOS---ENTER HERE TO SETUP SUPPRESS ZEROS AND NUMBER OF DIGITS TO TYPE
 7475                                ;*CALL:
 7476                                ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
 7477                                ;*      TYPOS                   ;;CALL FOR TYPEOUT
 7478                                ;*      .BYTE   N               ;;N=1 TO 6 FOR NUMBER OF DIGITS TO TYPE
 7479                                ;*      .BYTE   M               ;;M=1 OR 0
 7480                                ;*                                      ;;1=TYPE LEADING ZEROS
 7481                                ;*                                      ;;0=SUPPRESS LEADING ZEROS
 7482                                ;*
 7483                                ;*$TYPON----ENTER HERE TO TYPE OUT WITH THE SAME PARAMETERS AS THE LAST
 7484                                ;*$TYPOS OR $TYPOC
 7485                                ;*CALL:
 7486                                ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
 7487                                ;*      TYPON                   ;;CALL FOR TYPEOUT
 7488                                ;*
 7489                                ;*$TYPOC---ENTER HERE FOR TYPEOUT OF A 16 BIT NUMBER
 7490                                ;*CALL:
 7491                                ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
 7492                                ;*      TYPOC                   ;;CALL FOR TYPEOUT
 7493
 7494  036144  017646  000000        $TYPOS: MOV     @(SP),-(SP)     ;;PICKUP THE MODE
 7495  036150  116667  000001  000211        MOVB    1(SP),$OFILL    ;;LOAD ZERO FILL SWITCH
 7496  036156  112667  000207                MOVB    (SP)+,$OMODE+1  ;;NUMBER OF DIGITS TO TYPE
 7497  036162  062716  000002                ADD     #2,(SP)         ;;ADJUST RETURN ADDRESS
 7498  036166  000406                        BR      $TYPON
 7499  036170  112767  000001  000171  $TYPOC: MOVB    #1,$OFILL       ;;SET THE ZERO FILL SWITCH
 7500  036176  112767  000006  000165        MOVB    #6,$OMODE+1     ;;SET FOR SIX(6) DIGITS
 7501  036204  112767  000005  000154  $TYPON: MOVB    #5,$OCNT        ;;SET THE ITERATION COUNT
 7502  036212  010346                        MOV     R3,-(SP)        ;;SAVE R3
 7503  036214  010446                        MOV     R4,-(SP)        ;;SAVE R4
 7504  036216  010546                        MOV     R5,-(SP)        ;;SAVE R5
 7505  036220  116704  000145                MOVB    $OMODE+1,R4     ;;GET THE NUMBER OF DIGITS TO TYPE
 7506  036224  005404                        NEG     R4
 7507  036226  062704  000006                ADD     #6,R4           ;;SUBTRACT IT FOR MAX. ALLOWED
 7508  036232  110467  000132                MOVB    R4,$OMODE       ;;SAVE IT FOR USE
 7509  036236  116704  000125                MOVB    $OFILL,R4       ;;GET THE ZERO FILL SWITCH
 7510  036242  016605  000012                MOV     12(SP),R5       ;;PICKUP THE INPUT NUMBER
 7511  036246  005003                        CLR     R3              ;;CLEAR THE OUTPUT WORD
 7512  036250  006105                1$:     ROL     R5              ;;ROTATE MSB INTO "C"
 7513  036252  000404                        BR      3$              ;;GO DO MSB
```

```
                                2$:     ROL     R5              ;;FORM THIS DIGIT
                                        ROL     R5
                                        ROL     R5
                                        MOV     R5,R3
                                3$:     ROL     R3              ;;GET LSB OF THIS DIGIT
036264  105367  000076                DECB    $OMODE          ;;TYPE THIS DIGIT?
036270  100016                        BPL     7$              ;;BR IF NO
036272  042703  177770                BIC     #177770,R3      ;;GET RID OF JUNK
036276  001002                        BNE     4$              ;;TEST FOR 0
036300  005704                        TST     R4              ;;SUPPRESS THIS 0?
036302  001403                        BEQ     5$              ;;BR IF YES
036304  005204                4$:     INC     R4              ;;DON'T SUPPRESS ANYMORE 0'S
036306  052703  000060                BIS     #'0,R3          ;;MAKE THIS DIGIT ASCII
036312  052703  000040        5$:     BIS     #' ,R3          ;;MAKE ASCII IF NOT ALREADY
036316  110367  000040                MOVB    R3,8$           ;;SAVE FOR TYPING
036322  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$
```

```
 7570  036500  166700  142626                  SUB     $MSGAD,R0       ;;SUB START OF MESSAGE
 7571  036504  006200                          ASR     R0              ;;GET MESSAGE LNGTH IN WORDS
 7572  036506  010067  142622                  MOV     R0,$MSGLGT      ;;PUT LENGTH IN MAILBOX
 7573  036512  012767  000004  142576          MOV     #4,$MSGTYPE     ;;TELL APT TO TAKE MSG.
 7574  036520  000413                          BR      5$
 7575  036522  017667  000004  000016  3$:     MOV     @4(SP),4$       ;;PUT MSG ADDR IN JSR LINKAGE
 7576  036530  062766  000002  000004          ADD     #2,4(SP)                ;;BUMP RETURN ADDRESS
 7577  036536  016746  141234                  MOV     177776,-(SP)    ;;PUSH 177776 ON STACK
 7578  036542  004767  177114                  JSR     PC,$TYPE        ;;CALL TYPE MACRO
 7579  036546  000000                  4$:     .WORD   0
 7580  036550                          5$:
 7581  036550  105767  000062          10$:    TSTB    $FFLG           ;;SHOULD REPORT FATAL ERROR?
 7582  036554  001416                          BEQ     12$             ;;IF NOT:  BR
 7583  036556  005767  142554                  TST     $ENV            ;;RUNNING UNDER APT?
 7584  036562  001413                          BEQ     12$             ;;IF NOT:  BR
 7585  036564  005767  142526          11$:    TST     $MSGTYPE        ;;FINISHED LAST MESSAGE?
 7586  036570  001375                          BNE     11$             ;;IF NOT:  WAIT
 7587  036572  017667  000004  142520          MOV     @4(SP),$FATAL   ;;GET ERROR #
 7588  036600  062766  000002  000004          ADD     #2,4(SP)                ;;BUMP RETURN ADDR.
 7589  036606  005267  142504                  INC     $MSGTYPE        ;;TELL APT TO TAKE ERROR
 7590  036612  105067  000020          12$:    CLRB    $FFLG           ;;CLEAR FATAL FLAG
 7591  036616  105067  000013                  CLRB    $LFLG           ;;CLEAR LOG FLAG
 7592  036622  105067  000006                  CLRB    $MFLG           ;;CLEAR MESSAGE FLAG
 7593  036626  012601                          MOV     (SP)+,R1        ;;POP STACK INTO R1
 7594  036630  012600                          MOV     (SP)+,R0        ;;POP STACK INTO R0
 7595  036632  000207                          RTS     PC              ;;RETURN
 7596  036634     000                  $MFLG:  .BYTE   0               ;;MESSG. FLAG
 7597  036635     000                  $LFLG:  .BYTE   0               ;;LOG FLAG
 7598  036636     000                  $FFLG:  .BYTE   0               ;;FATAL FLAG
 7599          036640                          .EVEN
 7600          000200                  APTSIZE=200
 7601          000001                  APTENV=001
 7602          000100                  APTSPOOL=100
 7603          000040                  APTCSUP=040
 7604
 7605                                  .SBTTL  TTY INPUT ROUTINE
 7606
 7607                                  ;;**********************************************************************
 7608                                  .ENABL  LSB
 7609
 7610                                  ;;**********************************************************************
 7611                                  ;*SOFTWARE SWITCH REGISTER CHANGE ROUTINE.
 7612                                  ;*ROUTINE IS ENTERED FROM THE TRAP HANDLER, AND WILL
 7613                                  ;*SERVICE THE TEST FOR CHANGE IN SOFTWARE SWITCH REGISTER TRAP CALL
 7614                                  ;*WHEN OPERATING IN TTY FLAG MODE.
 7615  036640  022767  000176  142272  $CKSWR: CMP     #SWREG,SWR      ;;IS THE SOFT-SWR SELECTED?
 7616  036646  001074                          BNE     15$             ;;BRANCH IF NO
 7617  036650  105777  142270                  TSTB    @$TKS           ;;CHAR THERE?
 7618  036654  100071                          BPL     15$             ;;IF NO, DON'T WAIT AROUND
 7619  036656  117746  142264                  MOVB    @$TKB,-(SP)     ;;SAVE THE CHAR
 7620  036662  042716  177600                  BIC     #^C177,(SP)     ;;STRIP-OFF THE ASCII
 7621  036666  022726  000007                  CMP     #7,(SP)+        ;;IS IT A CONTROL G?
 7622  036672  001062                          BNE     15$             ;;NO, RETURN TO USER
 7623  036674  126727  142234  000001          CMPB    $AUTOB,#1       ;;ARE WE RUNNING IN AUTO-MODE?
 7624  036702  001456                          BEQ     15$             ;;BRANCH IF YES
 7625
```

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

```
                                        ;*      RETURN HERE             ;;CHARACTER IS ON THE STACK
                                        ;*                              ;;WITH PARITY BIT STRIPPED OFF
                                        ;
                                        $RDCHR: MOV     (SP),-(SP)      ;;PUSH DOWN THE PC
037124 016666 000004 000002             MOV     4(SP),2(SP)             ;;SAVE THE PS
037132 105777 142006                    1$:     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
```

```
                        TYPE                            ;;REPORT THE POWER FAILURE
037514  104401
$PWRMG: .WORD   POWERM                                  ;;POWER FAIL MESSAGE POINTER
037516  040344
                MOV     (PC)+,(SP)                      ;;RESTART AT START
037520  012716
$PWPAD: .WORD   START                                   ;;RESTART ADDRESS
037522  003606
                BIC     #20,2(SP)                       ;;CLEAR "T" BIT
037524  042766  000020  000002
                CLR     $TBIT                           ;;CLEAR THE "T" BIT FLAG
037532  005067  175304
                RTI
037536  000002
$ILLUP: HALT                                            ;;THE POWER UP SEQUENCE WAS STARTED
037540  000000
                BR      .-2                             ;; BEFORE THE POWER DOWN WAS COMPLETE
037542  000776
$SAVR6: 0                                               ;;PUT THE SP HERE
037544  000000


                .SBTTL  ERROR TYPE OUT ROUTINE
                ;;**************************************************************
                ;;**************************************************************
                ;*THIS ROUTINE IS CALLED TO TYPE AN ERROR MESSAGE WHICH IS INCLUDED
                ;*IN THE ERROR MESSAGE DATA TABLE. IT IS CALLED BY THE $ERROR ROUTINE
                ;*OR BY FIRST SETTING $ITEMB EQUAL TO THE ERROR TABLE ITEM TO BE PRINTED
                ;*OUT AND THEN EXECUTING A:
                ;*               JSR     PC,ERTYPE
                ;*
ERTYPE: TYPE                                                            ;TYPE A CRLF
037546  104401
                .WORD   $CRLF
037550  001313
                MOVB    @#$TSTNM,@#$TMP0
037552  113737  001102  001232
                BIC     #177400,@#$TMP0
037560  042737  177400  001232
                MOV     @#$ERRPC,@#$TMP1                        ;GET PC OF CALL
037566  013737  001116  001234
                MOV     R0,-(SP)                                ;SAVE R0
037574  010046

                MOVB    @#$ITEMB,R0                             ;GET THE ITEM NUMBER.
037576  113700  001114
                BIC     #177400,R0
037602  042700  177400
                BNE     1$
037606  001005

                MOV     @#$ERRPC,-(SP)                          ;IF ZERO THEN JUST
037610  013746  001116
                TYP0C                                           ;PRINT THE PC
037614  104402
                JMP     @#ERT5
037616  000137  040174

1$:             CMP     #377,R0
037622  022700  000377
                BNE     20$
037626  001005
                MOV     4(SP),R0
037630  016600  000004
                MOV     (R0),R0
037634  011000
                ADD     #400,R0
037636  062700  000400
20$:            DEC     R0                                      ;OTHERWISE MAKE R0 AN
037642  005300
                ASL     R0                                      ;INDEX FOR THE TABLE.
037644  006300
                ASL     R0
037646  006300
                ASL     R0
037650  006300
                ADD     #$ERRTB,R0
037652  062700  001442

                MOV     (R0)+,@#2$                              ;PICK UP THE ADDRESS
037656  012037  037666
                BEQ     3$                                      ;OF THE EM, ERROR MESSAGE
037662  001404
2$:             TYPE
037664  104401
                .WORD   0
037666  000000
                TYPE
037670  104401
                .WORD   $CRLF
037672  001313

3$:             MOV     (R0)+,@#4$                              ;GET THE DH,DATA HEADER
037674  012037  037704
                BEQ     5$
037700  001404
```

```
 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     2(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)+,2#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     2(R1)+,2#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     2(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.
                                ;*
040230  011637  001236          FPSPUR: MOV     (SP),@#$TMP2            ;SAVE PC OF TRAP.
040234  022626                          CMP     (SP)+,(SP)+             ;RESTORE SP.
040236  170200                          STFPS   R0                      ;GET FPS
040240  010037  001240                  MOV     R0,@#$TMP3
040244  170300                          STST    R0                      ;GET FEC
040246  010037  001242                  MOV     R0,@#$TMP4
040252  104211                  1$:     ERROR   211
040254  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?).
040256  000137  034556                  JMP     @#$EOP

                                        .SBTTL  CPU SPURIOUS TRAP TO 4 HANDLER
                                ;;**************************************************************
                                ;;**************************************************************
                                ;*THIS ROUTINE REPORTS UNEXPECTED CPU TRAPS TO VECTOR 4.
                                ;*
040262  011637  001236          CPSPUR: MOV     (SP),@#$TMP2            ;SAVE PC OF TRAP.
040266  022626                          CMP     (SP)+,(SP)+
040270  104212                  1$:     ERROR   212
040272  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?).
040274  000137  034556                  JMP     @#$EOP

                                        .SBTTL  CPU SPURIOUS TRAP TO 10 HANDLER
                                ;;**************************************************************
                                ;;**************************************************************
                                ;*THIS ROUTINE REPORTS UNEXPECTED CPU TRAPS TO VECTOR 10.
                                ;*
040300  011637  001236          CPTWO:  MOV     (SP),@#$TMP2            ;SAVE PC OF TRAP.
040304  022626                          CMP     (SP)+,(SP)+
040306  104213                  1$:     ERROR   213
040310  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?).
040312  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 '
```

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

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

                                ;ERROR MESSAGES:

043026      114  043104  051520  EM1:    .ASCIZ  'LDFPS AND STFPS TEST FAILED.'
043062  042114  050106  020123  EM2:    .ASCIZ  'LDFPS AND STFPS TEST ERROR SUMMARY.'
043136  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.'
043615      200  044504  020104          .ASCIZ  <CRLF>'DID NOT GO FROM STATE 400 TO 670.'
043660  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 '
044052      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.'
```

```
                045C36                          EM32=EM31
        045104  052123  020104  041501  EM33:   .ASCII  'STD AC0,(R0) FAILED.'<CRLF>
        045131     117  052125  052520          .ASCIZ  'OUTPUT BAD.'
        045145     123  042124  040440  EM34:   .ASCII  'STD AC0,(R0) FAILED IN THE FDST FLOWS.'
        045213     200  044124  020105          .ASCIZ  <CRLF>'THE (BUT GR7) FORK FAILED.'
        045247     114  042104  024040  EM35:   .ASCII  'LDD (R0),AC0 FAILED IN THE FSRC FLOWS.'
        045315     200  044124  020105          .ASCIZ  <CRLF>'THE (BUT GR7) FORK FAILED.'
        045351     123  042124  040440  EM36:   .ASCII  'STD AC0,(R0) FAILED IN THE FDST FLOWS.'
        045417     200  044124  020105          .ASCIZ  <CRLF>'THE (BUT FD) FORK FAILED.'
        045452  042114  020104  051050  EM37:   .ASCII  'LDD (R0),AC0 FAILED IN THE FSRC FLOWS.'
        045520  052200  042510  024040          .ASCIZ  <CRLF>'THE (BUT FD) FORK FAILED.'
        045553     114  042104  024040  EM40:   .ASCII  'LDD (R0),AC0 OR THE STD AC0,(R0) FAILED.'
        045623     200  040502  020104          .ASCIZ  <CRLF>'BAD DATA WAS DETECTED AFTER A SEQUENCE OF THE TWO INSTRUCTIONS.'
        045724  050106  020123  040502  EM41:   .ASCIZ  'FPS BAD AFTER EXECUTION OF: '
(0)     045761                          EM42:
(1)     045761     114  042104  024040          .ASCII  /LDD (R0),AC0 FAILED IN THE FSRC FLOWS./<CRLF>
(1)     046030  044124  020105  041050          .ASCIZ  /THE (BUT FSRC) FORK FAILED. TRAPPED TO 4./
(0)     046102                          EM43:
(1)     046102  052123  020104  041501          .ASCII  'STD AC0,(R0) FAILED IN THE FDST FLOWS./<CRLF>
(1)     046151     124  042510  024040          .ASCIZ  'THE (BUT FDST) FORK FAILED. TRAPPED TO 4.'
        046223     106  050120  040440  EM44:   .ASCIZ  'FPP ACCUMULATORS DATA TEST FAILED.'
                046223                          EM45=EM44
        046266  050106  020120  041501  EM46:   .ASCIZ  'FPP ACCUMULATORS DUAL ADDRESSING TEST FAILED.'
(0)     046344                          EM47:
(1)     046344  042114  040440  030503          .ASCII  /LD AC1,AC0 FAILED IN THE FSRC FLOWS./
(1)     046410  044124  020105  041050          .ASCIZ  /THE (BUT FSRC) FORK FAILED. TRAPPED TO 4./
        046462  042114  040440  030503  EM50:   .ASCII  'LD AC1,AC0 FAILED IN THE FSRC FLOWS.'
        046526  044124  020105  041050          .ASCIZ  'THE (BUT FD) FORK FAILED.'
        046560  042114  040440  030503  EM51:   .ASCIZ  'LD AC1,AC0 TRANSFERRED BAD DATA.'
(0)     046621                          EM52:
(1)     046621     114  042104  024040          .ASCII  /LDD (R0)+,AC0 FAILED IN THE FSRC FLOWS./
(1)     046670  044124  020105  041050          .ASCIZ  /THE (BUT FSRC) FORK FAILED. TRAPPED TO 4./
(0)     046742                          EM53:
(1)     046742  042114  020104  051050          .ASCII  /LDD (R0)+,AC0 FAILED IN THE FSRC FLOWS./
(1)     047011     200  030122  053440          .ASCII  <CRLF>'R0 WAS BAD.'<CRLF>
(1)     047026  044505  044124  051105          .ASCII  'EITHER A BAD CONSTANT WAS GENERATED OR'<CRLF>
(1)     047075     104  042111  047040          .ASCIZ  \DID NOT GO FROM STATE 627 TO 322.\
(0)     047137                          EM54:
(1)     047137     114  042104  024040          .ASCIZ  /LDD (R0)+,AC0 TRANSFERRED BAD DATA./
(0)     047203                          EM55:
(1)     047203     114  042104  026440          .ASCII  /LDD -(R0),AC0 FAILED IN THE FSRC FLOWS./
(1)     047252  044124  020105  041050          .ASCIZ  /THE (BUT FSRC) FORK FAILED. TRAPPED TO 4./
(0)     047324                          EM56:
(1)     047324  042114  020104  024055          .ASCII  /LDD -(R0),AC0 FAILED IN THE FSRC FLOWS./
(1)     047373     200  030122  053440          .ASCII  <CRLF>'R0 WAS BAD.'<CRLF>
(1)     047410  044505  044124  051105          .ASCII  'EITHER A BAD CONSTANT WAS GENERATED OR'<CRLF>
(1)     047457     104  042111  047040          .ASCIZ  \DID NOT GO FROM STATE 627 TO 324.\
(0)     047521                          EM57:
(1)     047521     114  042104  026440          .ASCIZ  /LDD -(R0),AC0 TRANSFERRED BAD DATA./
(0)     047565                          EM60:
(1)     047565     114  043104  024040          .ASCII  /LDF (R0)+,AC0 FAILED IN THE FSRC FLOWS./
(1)     047634  051200  020060  040527          .ASCII  <CRLF>'R0 WAS BAD.'<CRLF>
(1)     047651     105  052111  042510          .ASCII  'EITHER A BAD CONSTANT WAS GENERATED OR'<CRLF>
(1)     047720  044504  020104  047516          .ASCIZ  \DID NOT GO FROM STATE 627 TO 322.\
(0)     047762                          EM61:
(1)     047762  042114  020106  051050          .ASCIZ  /LDF (R0)+,AC0 TRANSFERRED BAD DATA.
```

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

H12


```
(0)  052537                               EM105:
(1)  052537    114  042104  040040                 .ASCII  'LDD @NUM,RO),ACO FAILED IN THE FSRC FLOWS.'
(1)  052611    200  044124  020105                 .ASCIZ  <CRLF>'THE (BUT FSRC) FORK FAILED. TRAPPED TO 4.'
(0)  052664                               EM106:
(1)  052664 042114  020104  047100                 .ASCII  'LDD @NUM(RO),ACO'<CRLF>
(1)  052705    106  044501  042514                 .ASCIZ  'FAILED TO AFFECT RO BY 2.'
(0)  052737                               EM107:
(1)  052737    114  042104  040040                 .ASCII  'LDD @NUM(RO),ACO FAILED IN THE FSRC FLOWS.'
(1)  053011    124  042510  024040                 .ASCIZ  'THE (BUT FD) FORK FAILED.'
(0)  053043                               EM110:
(1)  053043    114  042104  040040                 .ASCIZ  'LDD @NUM(RO),ACO LOADED BAD DATA.'
(0)  053105                               EM111:
(1)  053105    114  042104  040440                 .ASCII  /LDD AC7,ACO FAILED TO TRAP TO 244./
(1)  053147    200  041501  020067                 .ASCIZ  <CRLF>/AC7 IS AN ILLEGAL ACCUMULATOR./
            053105                        EM112=EM111
(0)  053207                               EM113:
(1)  053207    114  042104  040440                 .ASCII  /LDD AC6,ACO FAILED TO TRAP TO 244./
(1)  053251    200  041501  020066                 .ASCIZ  <CRLF>/AC6 IS AN ILLEGAL ACCUMULATOR./
            053207                        EM114=EM113
            053105                        EM115=EM111
            053207                        EM116=EM113
(0)  053311                               EM117:
(1)  053311    125  042523  047440                 .ASCII  'USE OF AN ILLEGAL ACCUMULATOR WITH FSRC MODE ZERO,'
(1)  053373    200  051124  050101                 .ASCIZ  <CRLF>'TRAPPED BUT FAILED TO SET FPS CORRECTLY.'
(0)  053445                               EM120:
(1)  053445    125  042523  047440                 .ASCII  'USE OF AN ILLEGAL ACCUMULATOR WITH FSRC MODE ZERO,'
(1)  053527    200  051124  050101                 .ASCIZ  <CRLF>'TRAPPED BUT FAILED TO SET FEC CORRECTLY.'
     053601    123  020124  041501        EM121:   .ASCII  'ST ACO,AC1 FAILED IN THE FDST FLOWS.'
     053645    200  044124  020105                 .ASCIZ  <CRLF>'THE (BUT FDST) FORK FAILED. TRAPPED TO 4.'
     053720 052123  040440  030103        EM122:   .ASCII  'ST ACO,AC1 FAILED IN THE FDST FLOWS.'
     053764 052200  042510  024040                 .ASCIZ  <CRLF>'THE (BUT FD) FORK FAILED.'
     054017    123  020124  041501        EM123:   .ASCIZ  'ST ACO,AC1 TRANSFERRED BAD DATA.'
(0)  054060                               EM124:
(1)  054060 050106  020123  040502                 .ASCII  'FPS BAD AFTER LDD (RO),ACO.'
(1)  054113    200  044124  020105                 .ASCIZ  <CRLF>\THE (BUT EZBT Y8) FORK FAILED.\
(0)  054153                               EM125:
(1)  054153    106  051520  041040                 .ASCII  'FPS BAD AFTER LDD (RO),ACO.'
(1)  054206 052200  042510  024040                 .ASCIZ  <CRLF>\THE (BUT ENBT) FORK FAILED.\
     054243    114  042104  024040        EM126:   .ASCII  'LDD (RO),ACO TRAPPED TO 244.'
     054277    040  051506  041522                 .ASCII  ' FSRC= -0 AND FIUV= 0.'<CRLF>
     054326 044124  020105  041050                 .ASCII  'THE (BUT FIUV) FORK FAILED.'
     054361    200  042527  052116                 .ASCIZ  <CRLF>'WENT FROM STATE 256 TO 354.'
     054415    200  047111  052123                 .ASCIZ  <CRLF>'INSTEAD OF FROM 256 TO 254.'
     054452 042114  020104  051050        EM127:   .ASCII  'LDD (RO),ACO FAILED TO TRAP TO 244,'
     054515    040  051506  041522                 .ASCII  ' FSRC= -0, FIUV= 1.'
     054540 052200  042510  024040                 .ASCII  <CRLF>'THE (BUT FIUV) FOR FAILED.'<CRLF>
     054574 042527  052116  043040                 .ASCII  'WENT FROM STATE 256 TO 254,'
     054627    200  047111  052123                 .ASCIZ  <CRLF>'INSTEAD OF FROM 256 THE 354.'
     054665    114  042104  024040        EM130:   .ASCII  'LDD (RO),ACO TRAPPED TO 244,'
     054721    106  051123  036503                 .ASCII  'FSRC= -0, FIUV= 1.'<CRLF>
     054744 052502  020124  042506                 .ASCIZ  'BUT FEC WAS BAD.'
(0)  054765                               EM131:
(1)  054765    114  041504  042106                 .ASCIZ  /LDCFD (RO),ACO LOADED BAD DATA./
(0)  055025                               EM132:
(1)  055025    114  041504  043104                 .ASCIZ  /LDCDF (RO),ACO LOADED BAD DATA./
(0)  055065                               EM133:
```

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

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

```
(1)  062445      101  042104  020104            .ASCII  'ADDD (R0),AC0 FAILED IN THE ACC-SUB FLOWS.'<CRLF>
(1)  062520  044504  020104  047516            .ASCIZ  \DID NOT TAKE THE PATH: STATE 216, TO 440, TO 141.\
(0)  062602                              EM204:
(1)  062602  042101  042104  024040            .ASCII  'ADDD (R0),AC0 FAILED IN THE ADD-SUB FLOWS.'<CRLF>
(1)  062655      104  042111  047040            .ASCIZ  \DID NOT TAKE THE PATH: STATE 216, TO 042, TO 141.
(0)  062737                              EM205:
(1)  062737      124  042510  043040            .ASCIZ  \THE FPS WAS BAD AFTER SUBD (R0),AC0.\
(C)  063024                              EM206:
(1)  063004  052523  042102  024040            .ASCIZ  /SUBD (R0),AC0 PRODUCED A BAD RESULT./
     063051      123  041125  020104  EM207:    .ASCII  'SUBD (R0),AC0 PRODUCED A BAD RESULT.'
     063115      200  044124  020105            .ASCIZ  <CRLF>'THE XOR OF THE SIGN BIT FAILED IN STATE 024.'
     063173      101  042104  020104  EM210:    .ASCIZ  'ADDD (R0),AC0 FAILED IN THE NORMALIZE FLOWS.'
          043233                      EM211=EM5
          043267                      EM212=EM6
          043321                      EM213=EM7

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

```
064640  051011  020060  052050          .ASCIZ  <TAB>'RO (TARGET LOCATIONS FOR OUTPUT).'
                064600          DH34=DH33
                064600          DH35=DH33
                064600          DH36=DH33
                064620          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  053040          .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    052040  042524  052123  DH211:  .ASCIZ  '  TEST.'<TAB>'PC OF CALL.'<TAB>'PC OF TRAP.'<TAB>'PSW.'
                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


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

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

                    067664                  .EVEN
                                            ;DATA TABLES:

          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
```

```
                067730                          DT4=DT3
        067752  001232  001234  040412  DT5:    .WORD   $TMP0,$TMP1,$TAB,$TMP2,LFIEX1,$TMP21,LFIEX2
        067770  001272  040531  001240          .WORD   $TMP20,FPSMS,$TMP3,FEOMS,$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
        070260  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,$CRLF
        070652  041743  001313  041714          .WORD   MS11,$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
070776  042163  001244  001313  DT44:   .WORD   MS17,$TMP5,$CRLF,MS20,$CRLF,$TMP0,$TMP1,$TAB,$TMP2
071020  001313  041554  042314          .WORD   $CRLF,MS1,MS21,$CRLF,$TMP3,$CRLF,MS2,MS21,$CRLF,$TMP4,0
071046  042163  001244  001313  DT45:   .WORD   MS17,$TMP5,$CRLF,MS24,$CRLF,$TMP0,$TMP1,$TAB,$TMP2,$TAB
071072  001246  001313  042353          .WORD   $TMP6,$CRLF,MS22,MS21,$CRLF,$TMP3,$CRLF
071110  042403  042314  001313          .WORD   MS23,MS21,$CRLF,$TMP4,0
071122  001232  001234  001313  DT46:   .WORD   $TMP0,$TMP1,$CRLF,MS25,MS30,$TMP2,MS31,$TMP3
071142  042611  001242  042620          .WORD   MS32,$TMP4,MS33,$TMP5,MS34,$TMP6,$CRLF,MS26
071162  042573  001250  042602          .WORD   MS30,$TMP7,MS31,$TMP10
071172  042611  001254  042620          .WORD   MS32,$TMP11,MS33,$TMP12,MS34,$TMP13,0
071210  001232  001234  040412  DT47:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$CRLF,MS12,MS7,$TMP3,MS10
071232  001242  001313  042026          .WORD   $TMP4,$CRLF,MS13,0
                070614                  DT50=DT34
071242  001232  001234  040412  DT51:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
071256  001313  042525  042564          .WORD   $CRLF,MS25,MS27,$TMP4,$CRLF,MS26,MS27,$TMP5,0
                071210                  DT52=DT47
071300  001232  001234  040412  DT53:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
071314  001242  000000                  .WORD   $TMP4,0
071320  001232  001234  040412  DT54:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$CRLF,MS1,MS21,$CRLF,$TMP3
071342  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
071356  001232  001234  040412  DT64:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3
071372  001242  000000                  .WORD   $TMP4,0
                071356                  DT65=DT64
                071320                  DT66=DT54
                070210                  DT67=DT21
071376  001232  001234  040412  DT70:   .WORD   $TMP0,$TMP1,$TAB,$TMP2,$CRLF,MS6,$CRLF,MS7,$TMP5
071420  041730  001246  001313          .WORD   MS10,$TMP6,$CRLF,MS11,$CRLF,MS7,$TMP5,MS10,$TMP7,0
                071376                  DT71=DT70
071444  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
071464  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
```

```
                    067730                  DT117=DT3
                    067730                  DT120=DT3
                    071210                  DT121=DT47
                    070614                  DT122=DT34
                    071242                  DT123=DT51
071476  001232  001234  040412  DT124:  .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TAB,$TMP3,$TMP4,$CRLF
071516  041676  001313  041714          .WORD   MS6,$CRLF,MS7,$TMP5,MS10,$TMP6,$CRLF
071534  041743  001313  041714          .WORD   MS11,$CRLF,MS7,$TMP5,MS10,$TMP7,$CRLF,MS37,$CRLF,$TMP10,0
                    071446                  DT125=DT124
                    071464                  DT126=DT111
                    071464                  DT127=DT111
                    067730                  DT130=DT3
071562  001232  001234  040412  DT131:  .WORD   $TMP0,$TMP1,$TAB,$TMP2,$CRLF,MS37,$CRLF,$TMP3
071602  001313  042670  001313          .WORD   $CRLF,MS40,$CRLF,$TMP4,$CRLF,MS415,$CRLF,$TMP5,0
                    071562                  DT132=DT131
071624  001232  001234  040412  DT133:  .WORD   $TMP0,$TMP1,$TAB,$TMP2,$CRLF,MS41,$CRLF,$TMP3
071644  001313  042750  001313          .WORD   $CRLF,MS42,$CRLF,$TMP4,$CRLF,MS43,$CRLF,$TMP5
071664  001313  043003  001313          .WORD   $CRLF,MS44,$CRLF,$TMP6,0
                    071624                  DT134=DT133
                    071624                  DT135=DT133
                    071624                  DT136=DT133
071676  001232  001234  040412  DT137:  .WORD   $TMP0,$TMP1,$TAB,$TMP2,$TMP10,$TAB,$TMP11,0
                    071676                  DT140=DT137
                    071676                  DT141=DT137
                    071676                  DT142=DT137
                    071624                  DT143=DT133
                    071624                  DT144=DT133
                    071676                  DT145=DT137
                    071676                  DT146=DT137
                    071624                  DT147=DT133
                    071624                  DT150=DT133
                    071676                  DT151=DT137
                    071624                  DT152=DT133
                    071676                  DT153=DT137
071716  001232  001234  040412  DT154:  .WORD   $TMP0,$TMP1,$TAB,$TMP2,0
                    071716                  DT155=DT154
                    071624                  DT156=DT133
                    071624                  DT157=DT133
                    071624                  DT160=DT133
                    071624                  DT161=DT133
                    071624                  DT162=DT133
                    071624                  DT163=DT133
                    067730                  DT164=DT3
                    067730                  DT165=DT3
                    071624                  DT166=DT133
                    071624                  DT167=DT133
                    071624                  DT170=DT133
                    071624                  DT171=DT133
                    071624                  DT172=DT133
                    071624                  DT173=DT133
                    071624                  DT174=DT133
                    071624                  DT175=DT133
                    071624                  DT176=DT133
                    071624                  DT177=DT133
                    071624                  DT200=DT133
                    071624                  DT201=DT133
```

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



                                        ;12345
                00000:                          .END
```

```
ADDATO  026236
ADDONE  026336
ADERR0  025670
ADERR1  025756
ADERR2  026012
ADERR3  026046
ADERR4  026056
ADERR5  026112
ADERR6  026146
ADERR7  026202
ADERIC  025764
ADPATO  026246
ADPAT1  026256
ADPAT2  026266
ADPAT3  026276
ADPAT4  026306
ADPAT5  026316
ADPAT6  026326
AR1     025310
AR10    025424
AR11    025454
AR12    025456
AR13    025474
AR14    025514
AR15    025534
AR16    025542
AR17    025546
AR2     025346
AR3     025552
AR4     025554
AR5     025606
AR6     025610
AR7     025630
AR8     025650
AR9     025656
ARBASE  = 000030
ARCSW1  = 000300
ARCSW2  = 000030
ARCPUOP = 000090
ARO     =%000000
AR1     =%000001
AR2     =%000002
AR3     =%000003
AR4     =%000004
AR5     =%000005
AR6     =%000006
AR7     =%000007
ADDWC = 000000
ADDW1 = 000000
ADDW10= 000000
ADDW11= 000000
ADDW12= 000000
ADDW13= 000000
ADDW14= 000000
ADDW15= 000000
ADDW2 = 000000
ADDW3 = 000000
ADDW4 = 000000
ADDW5 = 000000
ADDW6 = 000000
ADDW7 = 000000
ADDW8 = 000000
ADDW9 = 000000
ADEVCT= 000000
ADEVM = 000000
ADDONE  004670
AENV    = 000000
AENVM   = 000000
AERFLG  004560
AERR1   004562
AERR2   004574
AERR3   004626
AFATA_= 000000
AMADR1= 000000
AMADR2= 000000
AMADR3= 000000
AMADR4= 000000
AMAMS1= 000000
AMAMS2= 000000
AMAMS3= 000000
AMAMS4= 000000
AMSGAD= 000000
AMSGLG= 000000
AMSGTY= 000000
AMTYP1= 000000
AMTYP2= 000000
AMTYP3= 000000
AMTYP4= 000000
APASS = 000000
APRIOR= 000000
APTCSU= 000040
APTENV= 000001
APTSIZ= 000200
APTSPO= 000100
ASWREG= 000000
ATESTN= 000000
AUNIT = 000000
AUSWR = 000000
AVECT1= 000000
AVECT2= 000000
AC5     004476
A1      004350
A11     004350
A12     004364
A2      004422
A3      004426
A4      004474
A5      004500
A6      004526
A7      004552
BBDATO  031300
BBDONE  031430
BBERO   030766
BBER1   031026
BBER1C  031006
BBER11  031042
BBER2   031064
BBER3   031102
BBER4   031136
BBER40  031152
BBER5   031172
BBER6   031210
BBER7   031244
BBER8   031262
BBPATO  031310
BBPAT1  031320
BBPAT2  031330
BBPAT3  031340
BBPAT4  031350
BBPAT5  031360
BBPAT6  031370
BBP10   031410
BBP11   031420
BBP7    031400
BB1     030130
BB10    030304
BB11    030324
BB12    030334
BB13    030342
BB14    030354
BB15    030404
BB16    030430
BB17    030440
BB2     030166
BB20    030452
BB21    030502
BB22    030526
BB23    030546
BB24    030556
BB25    030564
BB26    030576
BB27    030626
BB3     030172
BB30    030650
BB31    030660
BB32    030672
BB33    030722
BB34    030744
BB35    030754
BB4     030210
BB5     030220
BB6     030232
BB7     030262
BDONE   004772
BERR    004726
BERR1   004756
BIT0    = 000001
BIT00   = 000001
BIT01   = 000002
BIT02   = 000004
BIT03   = 000010
BIT04   = 000020
BIT05   = 000040
BIT06   = 000100
BIT07   = 000200
BIT08   = 000400
BIT09   = 001000
BIT1    = 000002
BIT10   = 002000
BIT11   = 004000
BIT12   = 010000
BIT13   = 020000
BIT14   = 040000
BIT15   = 100000
BIT2    = 000004
BIT3    = 000010
BIT4    = 000020
BIT5    = 000040
BIT6    = 000100
BIT7    = 000200
BIT8    = 000400
BIT9    = 001000
BPTVEC= 000014
B1      004702
B2      004704
B3      004722
CCDATO  027764
CCDONE  030124
CCERO   027326
CCER1   027362
CCER10  027674
CCER11  027712
CCER12  027730
CCER13  027746
CCER2   027420
CCER22  027434
CCER3   027456
CCER4   027474
CCER44  027510
CCER5   027532
CCER50  027376
CCER55  027546
CCER6   027566
CCER7   027622
CCER8   027640
CCER90  027344
CCP0    027774
CCP1    030004
CCP10   030074
CCP11   030104
CCP12   030114
CCP2    030014
CCP3    030024
CCP4    030034
CCP5    030044
CCP6    030054
CCP7    030064
CC1     026342
CC10    026556
CC11    026566
CC12    026574
CC13    026606
CC14    026636
CC15    026662
CC16    026702
CC17    026712
CC18    026720
CC19    026732
CC2     026372
CC20    026762
CC21    027006
CC22    027026
CC23    027036
CC24    027044
CC25    027056
CC26    027106
CC27    027130
CC28    027150
CC29    027160
CC3     026414
CC30    027166
CC31    027200
CC32    027230
CC33    027252
CC34    027272
CC35    027302
CC36    027310
CC37    027322
CC4     026434
```

```
CC5        026444
CC6        026452
CC7        026464
CC8        026514
CC9        026536
CDONE      005676
CERR1      005432
CERR2      005530
CERR3      005630
CERR4      005652
CFCC1      041344
CKSWR  =   104406
CNT    =   000214
CPC        005674
CPSPUR     040232
CPTWC      040250
CR     =   000015
CRLF   =   000200
C1         005006
C15        005034
C2         005054
C25        005070
C3         005112
C35        005142
C4         005164
C45        005176
C5         005220
C55        005246
C6         005266
C65        005302
C7         005324
C75        005354
C8         005376
C85        005410
DDDATO     032274
DDDONE     033424
DDER0      032506
DDER1      032524
DDER10     033142
DDER11     033200
DDER12     033236
DDER2      032562
DDER3      032620
DDER4      032656
DDER5      032714
DDER6      032752
DDER7      033010
DDER8      033046
DDER9      033104
DDISP  =   177570
DDONE      006140
DDP0       033304
DDP1       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       066655
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      067631
DF162  =   067571
DF163  =   067631
DF164  =   066631
DF165  =   066631
DF166  =   067571
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

DF46       067316
DF47       067350
DF5        066641
DF50       067365
DF51       067411
DF52   =   067350
DF53       067427
DF54       067436
DF55   =   067350
DF56   =   067427
DF57   =   067436
DF6    =   066641
DF60   =   067427
DF61   =   067436
DF62   =   067436
DF63   =   067436
DF64       067454
DF65   =   067454
DF66   =   067436
DF67   =   066742
DF7    =   066641
DF70       067463
DF71   =   067463
DF72       067505
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
```

```
DH124 = 063740          DH202 = 065122          DH7   = 063620          DT142 = 071676          DT27    = 070442
DH125 = 063740          DH203 = 065122          DH70  = 063620          DT143 = 071624          DT3     = 067730
DH126 = 063620          DH204 = 065122          DH71  = 064450          DT144 = 071624          DT30    = 070504
DH127 = 065122          DH205 = 063740          DH72  = 064512          DT145 = 071676          DT31    = 070366
DH13  = 000000          DH206 = 065122          DH73  = 065122          DT146 = 071676          DT32    = 070442
DH130 = 066023          DH207 = 065122          DH74  = 063620          DT147 = 071624          DT33    = 070536
DH131 = 065122          DH21  = 063620          DH75  = 063620          DT15  = 070004          DT34    = 070614
DH132 = 065122          DH210 = 065122          DH76  = 064450          DT150 = 071624          DT35    = 070614
DH133 = 066113          DH211 = 066544          DH77  = 064512          DT151 = 071676          DT36    = 070614
DH134 = 066113          DH212 = 063620          DISP_A  001142          DT152 = 071624          DT37    = 070614
DH135 = 066113          DH213 = 063620          DISPRE  000174          DT153 = 071676          DT4     = 067730
DH136 = 066113          DH22  = 064116          DPAT3   016446          DT154 = 071716          DT40    = 070536
DH137 = 066223          DH23  = 064153          DSWR  = 177570          DT155 = 071716          DT41    = 070666
DH14  = 063620          DH24  = 064311          DT1     067664          DT156 = 071624          DT42    = 070720
DH140 = 066223          DH25  = 064450          DT10  = 070004          DT157 = 071624          DT43    = 070720
DH141 = 066223          DH26  = 064512          DT100 = 071320          DT16    070146          DT44    = 070776
DH142 = 066223          DH27  = 064512          DT101 = 071376          DT160 = 071624          DT45    = 071046
DH143 = 066113          DH3   = 063433          DT102 = 071376          DT161 = 071624          DT46    = 071122
DH144 = 066113          DH30  = 000000          DT103 = 071376          DT162 = 071624          DT47    = 071210
DH145 = 066223          DH31  = 064512          DT104 = 071320          DT163 = 071624          DT5     = 067752
DH146 = 066223          DH32  = 064512          DT105 = 071376          DT164 = 067730          DT50    = 070614
DH147 = 065122          DH33  = 064600          DT106 = 071444          DT165 = 067730          DT51    = 071242
DH15  = 063620          DH34  = 064600          DT107 = 071376          DT166 = 071624          DT52    = 071210
DH150 = 066413          DH35  = 064600          DT11  = 070004          DT167 = 071624          DT53    = 071300
DH151 = 066223          DH36  = 064600          DT110 = 071320          DT17  = 067730          DT54    = 071320
DH152 = 066113          DH37  = 064600          DT111   071464          DT170 = 071624          DT55    = 071210
DH153 = 066223          DH4   = 063524          DT112 = 071464          DT171 = 071624          DT56    = 071300
DH154 = 066504          DH40  = 064600          DT113 = 071464          DT172 = 071624          DT57    = 071320
DH155 = 066504          DH41  = 000000          DT114 = 071464          DT173 = 071624          DT6     = 070004
DH156 = 065122          DH42  = 064703          DT115 = 071464          DT174 = 071624          DT60    = 071300
DH157 = 065122          DH43  = 064703          DT116 = 071464          DT175 = 071624          DT61    = 071320
DH16  = 063620          DH44  = 000000          DT117 = 067730          DT176 = 071624          DT62    = 071320
DH160 = 065122          DH45  = 065005          DT12    070026          DT177 = 071624          DT63    = 071320
DH161 = 065122          DH46  = 065024          DT120 = 067730          DT2     067706          DT64    = 071356
DH162 = 065122          DH47  = 065005          DT121 = 071210          DT20    070166          DT65    = 071356
DH163 = 065122          DH5   = 063620          DT122 = 070614          DT200 = 071624          DT66    = 071320
DH164 = 063740          DH50  = 065050          DT123 = 071242          DT201 = 071624          DT67    = 070210
DH165 = 063740          DH51  = 065050          DT124   071476          DT202 = 071624          DT7     = 070004
DH166 = 065122          DH52  = 063620          DT125 = 071476          DT203 = 071624          DT70    = 071376
DH167 = 065122          DH53  = 064512          DT126 = 071464          DT204 = 071624          DT71    = 071376
DH17  = 063740          DH54  = 065122          DT127 = 071464          DT205 = 067730          DT72    = 071444
DH170 = 065122          DH55  = 063620          DT13    070114          DT206 = 071624          DT73    = 071320
DH171 = 065122          DH56  = 064512          DT130 = 067730          DT207 = 071624          DT74    = 070210
DH172 = 065122          DH57  = 065122          DT131   071562          DT21    070210          DT75    = 071376
DH173 = 065122          DH6   = 063620          DT132 = 071562          DT210 = 071624          DT76    = 071376
DH174 = 065122          DH60  = 064512          DT133   071624          DT211   071730          DT77    = 071444
DH175 = 065122          DH61  = 065122          DT134 = 071624          DT212   071746          D1      005724
DH176 = 065122          DH62  = 065122          DT135 = 071624          DT213 = 071746          D10     006126
DH177 = 065122          DH63  = 065122          DT136 = 071624          DT22    070222          D2      005750
DH2   = 063440          DH64  = 065232          DT137   071676          DT23    070250          D3      005752
DH20  = 064030          DH65  = 065163          DT14  = 070004          DT24    070272          D4      005756
DH200 = 065122          DH66  = 065122          DT140 = 071676          DT25    070316          D5      005770
DH201 = 065122          DH67  = 063620          DT141 = 071676          DT26    070366          D6      006004
```

```
D7         006014
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      0523[illegible]3
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
FEOMS      040575
FERR0      006702
FERR1      006740
FERR10     007520
FERR11     007554
FERR2      007036
FERR20     007574
FERR21     007712
FERR25     007742
FERR26     010060
FERR3      007076
FERR4      00707[illegible]
FERR5      007174
FERR6      007230
FERR7      007364
FEP2       007042
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
FXDAT0     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        006640
F21        006650
F22        006660
F23        006676
F3         006356
F4         006360
F5         006376
```

```
F6      000452
F7      000476
GADR    014200
GAND0   014150
GAND1   014152
GAND2   014154
GAND3   014156
GCMP    013704
GDAT00  014170
GDAT01  014172
GDAT02  014174
GDAT03  014176
GDONE   014202
GERR1   013732
GFLAG1  014124
GFLAG2  014126
GOR0    014160
GOR1    014162
GOR2    014164
GOR3    014166
GPAT00  014130
GPAT01  014132
GPAT02  014134
GPAT03  014136
GPAT10  014140
GPAT11  014142
GPAT12  014144
GPAT13  014146
GRESET  013664
GSETUP  013606
GSUM    014022
GSW     013636
GTSWR =  104405
G1      011710
G10     012206
G11     012210
G12     012276
G13     012330
G14     012332
G15     012420
G16     012452
G17     012454
G2      011742
G20     012542
G21     012574
G22     012576
G23     012664
G24     012716
G25     012720
G26     013006
G27     013040
G3      011744
G30     013042
G31     013130
G32     013162
G33     013164
G34     013252
G35     013304
G36     013306
G37     013374
G4      012032
G40     013426
G41     013430
G42     013516
G43     013550
G44     013552
G5      012064
G6      012066
G7      012154
HADR    014732
HA1R    015006
HA1W    014736
HA2R    015016
HA2W    014746
HA3R    015026
HA3W    014756
HA4R    015036
HA4W    014766
HA5R    015046
HA5W    014776
HCLR    014662
HCLR1   014672
HCMP    014624
HCMP1   014644
HCMP2   014654
HDAT1   015056
HDAT2   015066
HDAT3   015076
HDAT4   015106
HDAT5   015116
HDONE   015126
HERROR  014700
HFLAG   014734
HSTD    014546
HT    = 000011
H1      014210
H10     014452
H11     014504
H12     014536
H2      014230
H3      014240
H4      014312
H5      014334
H6      014366
H7      014420
IDAT10  011026
IDAT11  011030
IDAT12  011032
IDAT13  011034
IDAT00  011016
IDAT01  011020
IDAT02  011022
IDAT03  011024
IDONE   011036
IERR0   010576
IERR1   010660
IERR2   010700
IERR25  010722
IERR3   010752
IERR4   010726
ILLMS   041506
ILL1    041402
ILL2    041446
IOTVEC= 000020
IPAT10  010776
IPAT11  011000
IPAT12  011002
IPAT13  011004
IPAT20  011006
IPAT21  011010
IPAT22  011012
IPAT23  011014
I1      010202
I10     010364
I105    010372
I106    010366
I11     010374
I12     010376
I13     010412
I14     010454
I15     010456
I16     010460
I17     010476
I2      010220
I20     010536
I21     010552
I22     010556
I23     010572
I3      010266
I4      010270
I5      010272
I6      010316
I7      010332
JBUF0   016116
JBUF1   016120
JBUF2   016122
JBUF3   016124
JDAT10  016126
JDAT11  016130
JDAT12  016132
JDAT13  016134
JDAT00  016136
JDAT0   016146
JDAT01  016140
JDAT02  016142
JDAT03  016144
JDAT1   016150
JDAT2   016152
JDAT3   016154
JDONE   016156
JERR0   015776
JERR1   016044
JERR2   016070
J1      015670
J10     016016
J2      015714
J3      015716
J4      015720
J5      015754
J6      015762
J7      015774
KBUF0   016420
KBUF1   016422
KBUF2   016424
KBUF3   016426
KDAT10  016410
KDAT11  016412
KDAT12  016414
KDAT13  016416
KDAT00  016430
KDAT01  016432
KDAT02  016434
KDAT03  016436
KDONE   016450
KERR0   016272
KERR1   016336
KERR2   016362
KPAT0   016440
KPAT1   016442
KPAT2   016444
K1      016164
K10     016312
K2      016210
K3      016212
K4      016214
K5      016250
K6      016256
K7      016270
LDAT10  016750
LDAT11  016752
LDAT12  016754
LDAT13  016756
LDAT00  016760
LDAT01  016762
LDAT02  016764
LDAT03  016770
LDONE   016772
LD1     041242
LD2     041272
LEPR1   016604
LEPR2   016656
LERR3   016630
LF    = 000012
LFIEX1  040417
LFIEX2  040467
LFPS1   041227
LOOP    004304
LPAT10  016726
LPAT11  016730
LPAT12  016732
LPAT13  016734
LPAT20  016736
LPAT21  016740
LPAT22  016742
LPAT23  016744
LPERR =  104413
L1      016456
L2      016516
L3      016520
L4      016522
L5      016572
L6      016600
MDAT00  017312
MDAT01  017314
MDAT02  017316
MDAT03  017320
MDONE   017322
MERR0   017146
MERR1   017204
MERR2   017120
MERR3   017232
MNUMBE= 000213
MNUM0   042210
MNUM1   042216
MNUM2   042223
MNUM3   042230
MNUM4   042237
MNUM5   042245
MPAT10  017272
MPAT11  017274
MPAT12  017276
MPAT13  017300
MPAT20  017302
MPAT21  017304
MPAT22  017306
```

```
MPAT23  017310    NDAT01  017760    ODAT02  020466    PERR16  021006    QERR21  021532
MS1     041554    NDAT02  017762    ODAT03  020470    PERR17  021014    QERR22  021540
MS10    041730    NDAT03  017764    ODONE   020534    PERR2   021124    QERR3   021566
MS11    041743    NDONE   020030    OERR0   020224    PERR20  021042    QERR4   021574
MS12    041770    NERR0   017520    OERR1   020324    PERR21  021052    QPAT10  021650
MS13    042026    NERR1   017620    OERR10  020256    PERR22  021060    QPAT11  021652
MS14    042043    NERR10  017552    OERR11  020270    PIRQ    = 177772  QPAT12  021654
MS15    042131    NERR11  017564    OERR2   020360    PIRQVE= 000240    QPAT13  021656
MS16    042156    NERR2   017654    OERR20  020332    POWERM  040344    QPAT20  021660
MS17    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    01      020034    PS      = 177776  RDCHR   = 104407
MS33    042620    NOOP7   041052    010     020164    PSW     = 177776  RESREG= 104411
MS34    042627    NPAT10  020010    011     020174    PWRVEC= 000024    RESVEC= 000010
MS35    042636    NPAT11  020012    012     020176    P1      020540    RSETUP= 104412
MS36    042643    NPAT12  020014    013     020212    P2      020562    R6      =%000006
MS37    042652    NPAT13  020016    014     020222    P3      = 020564  R7      =%000007
MS4     041622    NPAT20  017776    02      020060    P4      020566    SADR    015636
MS40    042670    NPAT21  020000    03      020062    P5      020610    SAVREG= 104410
MS41    042724    NPAT22  020002    04      020064    P6      020632    SDAT00  015652
MS415   042704    NPAT23  020004    05      020102    P7      020642    SDAT01  015654
MS42    042750    NULL    040411    06      020112    P8      020652    SDAT02  015656
MS43    042766    N1      017326    07      020122    QDATI0  021700    SDAT03  015660
MS44    043003    N10     017460    08      020136    QDATI1  021702    SDONE   015662
MS5     041634    N11     017470    09      020150    QDATI2  021704    SERR0   015342
MS6     041676    N12     017472    PDATI0  021170    QDATI3  021706    SERR1   015552
MS7     041714    N13     017506    PDATI1  021172    QDAT00  021670    SERR10  015362
M1      016776    N14     017516    PDATI2  021174    QDAT01  021672    SERR15  015442
M15     017016    N2      017352    PDATI3  021176    QDAT02  021674    SERR2   015502
M2      017022    N3      017354    PDAT00  021200    QDAT03  021676    SERR20  015462
M3      017024    N4      017356    PDAT01  021202    QDONE   021710    SERR3   015526
M4      017026    N5      017374    PDAT02  021204    QERR0   021356    SERR4   015420
M5      017062    N6      017404    PDAT03  021206    QERR1   021622    SERR5   015604
M6      017066    N7      017414    PDONE   021210    QERR11  021366    SERR6   015514
M7      017076    N8      017432    PERR0   020656    QERR12  021404    SERR7   015540
M8      017106    N9      017444    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          TDAT12  011640          TST5    006142          UPAT33  023000          XDAT01  024340
SPAT13  015650          TDAT13  011642          TST6    006302          UPAT40  023002          XDAT02  024342
STACK = 001100          TDAT00  011624          TST7    010176          UPAT41  023004          XDAT03  024344
START   003606          TDAT01  011626          TYPE  = 104401          UPAT42  023006          XDONE   024376
STFS1   041303          TDAT02  011630          TYPOC = 104402          UPAT43  023010          XERR1   024222
STKLMT= 177774          TDAT03  011632          TYPON = 104404          URCM1   023020          XERR2   024304
STST1   041433          TDONE   011644          TYPOS = 104403          URCM2   023022          XERR3   024260
STT1    041316          TERR0   011412          T1      011042          URCM3   023024          XERR4   024320
STT2    041333          TERR1   011474          T10     011204          UTMP1   023014          XPAT20  024346
STW     041140          TERR2   011514          T105    011212          UTMP2   023016          XPAT01  024350
TBEG    000176          TERR25  011530          T11     011214          U0      021730          XPAT02  024352
TBIT0 = 000001          TERR3   011560          T12     011216          U1      021760          XPAT03  024354
TBIT1 = 000002          TERR4   011542          T13     011332          U10     022316          XPAT10  024356
TBIT2 = 000004          TVVEC = 000060          T14     011274          U11     022320          XPAT12  024362
TBIT3 = 000010          TPAT10  011604          T15     011276          U12     022352          XPAT13  024364
TBIT4 = 000020          TPAT11  011606          T16     011300          U13     022370          XPAT20  024366
TBIT5 = 000040          TPAT12  011610          T17     011316          U14     022422          XPAT21  024370
TBIT6 = 000100          TPAT13  011612          T2      011062          U15     022452          XPAT22  024372
TBIT7 = 000200          TPAT20  011614          T20     011350          U16     022454          XPAT23  024374
TBIT8 = 000400          TPAT21  011616          T21     011372          U2      022024          XTMP    024334
TBIT9 = 001000          TPAT22  011620          T22     011374          U3      022056          X1      023606
TBIT10= 002000          TPAT23  011622          T23     011406          U4      022122          X10     024000
TBIT11= 004000          TPVEC = 000064          T3      011130          U5      022154          X11     024014
TBIT12= 010000          TRAPVE= 000034          T4      011132          U6      022220          X12     024050
TBIT13= 020000          TRTVEC= 000014          T5      011134          U7      022252          X13     024074
TBIT14= 040000          TST1    004304          T6      011160          WDAP00  023572          X14     024102
TBIT15= 100000          TST10   011040          T7      011174          WDAT01  023574          X15     024116
TBUF    000004          TST11   011646          UDONE   023026          WDAT02  023576          X16     024152
TBUFM1  000010          TST12   014204          UERR0   022472          WDAT03  023600          X17     024176
TBUFM2  000020          TST13   015130          UERR1   022516          WDONE   023602          Y2      023642
TBUFM3  000040          TST14   015664          UERR10  022524          WPAT00  023562          X20     024204
TCCR    000100          TST15   016160          UERR11  022566          WPAT01  023564          X21     024220
TCSR    000200          TST16   016452          UERR2   022602          WPAT02  023566          X3      023666
TDB     000400          TST17   016774          UERR20  022610          WPAT03  023570          X4      023674
TDBUF   000402          TST2    004672          UERR21  022652          WSETUP  023530          X5      023712
TDCSR   000400          TST20   017324          UERR3   022666          W1      023030          X6      023746
TDDB    001000          TST21   020032          UERR4   022716          W10     023242          X7      023772
TDONE   011644          TST22   020536          UFLAG   023012          W11     023270          YDAT00  024742
TDRBUF  000402          TST23   021212          UPAT00  022742          W12     023320          YDAT01  024744
TDRCSR  000400          TST24   021712          UPAT01  022744          W13     023344          YDAT02  024746
TDRDB   001000          TST25   023026          UPAT02  022746          W14     023360          YDAT03  024750
TDTCSR  000402          TST26   023604          UPAT03  022750          W15     023406          YDONE   025002
TDTDB   001002          TST27   024400          UPAT10  022752          W16     023442          YERR1   024602
TDTSR   000404          TST3    004774          UPAT11  022754          W17     023466          YERR2   024644
TDVEC = 000300          TST30   025004          UPAT12  022756          W2      023060          YERR3   024706
TERR    000100          TST31   025306          UPAT13  022760          W20     023502          YFLAG   024732
TERR1   000200          TST32   026340          UPAT20  022762          W3      023104          YPAT00  024752
TERR2   000400          TST33   030126          UPAT21  022764          W4      023122          YPAT01  024754
TERR3   001000          TST34   031432          UPAT22  022766          W5      023152          YPAT02  024756
TERR4   002000          TST35   033426          UPAT23  022770          W6      023206          YPAT03  024760
TERR5   004000          TST36   034202          UPAT30  022772          W7      023232          YPAT10  024762
TERR6   010000          TST37   034556          UPAT31  022774          XAPT11  024360          YPAT11  024764
TERR7   020000          TST4    005700          UPAT32  022776          XDAT00  024336          YPAT12  024766
```

YPAT13	024770
YPAT20	024772
YPAT21	024774
YPAT22	024776
YPAT23	025000
YTMP1	024734
YTMP2	024736
[illegible]	[illegible]
[illegible]	[illegible]
Y3	[illegible]
Y4	[illegible]
Y5	024554
Y6	024556
[illegible]	024572
ZDAT00	025244
ZDAT01	025246
ZDAT02	025250
ZDAT03	025252
ZDONE	025324
ZERR1	025150
ZERR2	025212
ZFLAG	025236
ZPAT00	025254
ZPAT01	025256
ZPAT02	025260
ZPAT03	025262
ZPAT10	025264
ZPAT11	025266
ZPAT12	025270
ZPAT13	025272
ZPAT20	025274
ZPAT21	025276
ZPAT22	025300
ZPAT23	025302
ZTMP1	025240
ZTMP2	025242
Z1	025026
Z2	025050
Z3	025074
Z4	025102
Z5	025114
Z6	025146
$APTHD	003572
$ATYC	036416
$ATY1	036372
$ATY3	036400
$ATY4	036410
$AUTOB	001134
$BASE	001372
$BDADR	001122
$BDDAT	001126
$BELL	001306
$CDW1	001376
$CDW2	001400
$CHARC	036140
$CKSWR	036640
$CLR.T	034760
$CMTAG	001100
$CM1 =	000024
$CM2 =	000050
$CM3 =	000024
$CM4 =	000024
$CNTLG	037247
$CNTLU	037242
$CPUOP	001344
$CRLF	001313
$DDW0	001402
$DDW1	001404
$DDW10	001426
$DDW11	001430
$DDW12	001432
$DDW13	001434
$DDW14	001436
$DDW15	001440
$DDW2	001406
$DDW3	001410
$DDW4	001412
$DDW5	001414
$DDW6	001416
$DDW7	001420
$DDW8	001422
$DDW9	001424
$DEVCT	001326
$DEVM	001374
$DIAGN	035000
$ENDAD	034770
$ENDCT	034612
$ENULL	035044
$ENV	001336
$ENVM	001337
$EOP	034556
$EOPCT	034604
$ERFLG	001103
$ERMAX	001115
$ERROR	035330
$ERRPC	001116
$ERRTB	001442
$ERTTL	001112
$ESCAP	001304
$ETABL	001336
$ETEND	001442
$FATAL	001320
$FFLG	036636
$FILLC	001156
$FILLS	001155
$GDADR	001120
$GDDAT	001124
$GET42	034742
$GTSWR	036710
$HD =	000003
$HIBTS	003572
$ICNT	001104
$ILLUP	037540
$INTAG	001135
$ITEMB	001114
$LF	001314
$LFLG	036635
$LOOP	035036
$LPADR	001106
$LPERR	001110
$MADR1	001350
$MADR2	001354
$MADR3	001360
$MADR4	001364
$MAIL	001316
$MAMS1	001346
$MAMS2	001352
$MAMS3	001356
$MAMS4	001362
$MBADR	003574
$MFLG	036634
$MNEW	037265
$MSGAD	001332
$MSGLG	001334
$MSGTY	001316
$MSWR	037254
$MTYP1	001347
$MTYP2	001353
$MTYP3	001357
$MTYP4	001363
$MXCNT	035326
$NULL	001154
$NWTST=	000001
$OCNT	036366
$OMODE	036370
$OVER	035312
$PASS	001324
$PASTM	003600
$PWRAD	037522
$PWRDN	037362
$PWRMG	037516
$PWRUP	037434
$QUES	001312
$RDCHR	037122
$RDSZ =	000001
$REGAD	001160
$REG0	001162
$REG1	001164
$REG10	001202
$REG11	001204
$REG12	001206
$REG13	001210
$REG14	001212
$REG15	001214
$REG16	001216
$REG17	001220
$REG2	001166
$REG20	001222
$REG21	001224
$REG22	001226
$REG23	001230
$REG3	001170
$REG4	001172
$REG5	001174
$REG6	001176
$REG7	001200
$RESRE	035624
$RTNAD	035040
$RTRN	035034
$SAVRE	035566
$SAVR6	037544
$SCOPE	035050
$SETUP=	000137
$STUP =	177777
$SVLAD	035256
$SVPC =	003572
$SWR =	177400
$SWREG	001340
$SWRMK=	000000
$SWRMS=	000200
$TAB	040412
$TBIT	035042
$TERM =	000030
$TESTN	001322
$THE	040606
$TIMES	001302
$TKB	001146
$TKS	001144
$TMP0	001232
$TMP1	001234
$TMP10	001252
$TMP11	001254
$TMP12	001256
$TMP13	001260
$TMP14	001262
$TMP15	001264
$TMP16	001266
$TMP17	001270
$TMP2	001236
$TMP20	001272
$TMP21	001274
$TMP22	001276
$TMP23	001300
$TMP3	001240
$TMP4	001242
$TMP5	001244
$TMP6	001246
$TMP7	001250
$TN =	000037
$TPB	001152
$TPFLG	001157
$TPS	001150
$TRAP	037276
$TRAP2	037320
$TRP =	000014
$TRPAD	037332
$TSTM	003576
$TSTNM	001102
$TYPE	035662
$TYPEC	036074
$TYPEX	036142
$TYPOC	036170
$TYPON	036204
$TYPOS	036144
$UNIT	001330
$UNITM	003602
$USWR	001342
$VECT1	001366
$VECT2	001370
$XTSTR	035062
$$GET4=	000001
$OFILL	036367
. =	071760
.LPER	040266
.RSET	040274
.$X =	003572

. ABS.   071760   000

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

```
MAINDEC-11-DFFPA-A      PDP 11 34 FPP DIAGNOSTIC PART 1 MACY11 27(1006)  01-NOV-76  21:09  PAGE .72
DFFPAA.P11    01-NOV-76 21:03          SYMBOL TABLE


DSK2:DFFPAA.BIN,DSK2:DFFPAA.SEG SOL=DFFPAA.P11
RUN-TIME:  86.73  SECONDS
RUN-TIME RATIO: 244/167=1.4
CORE USED:  31K  (61 PAGES)
```

D14

Spooler runtime 29 Seconds, 123 KCS, 757 disk reads, 4 disk writes, 171 pages

[illegible] Monitor IPC-D 602 (600) *END**

[illegible]

0000000011111111112222222222333333333344444444445555555555666666666677777777 [illegible]