



DATA GENERAL
CORPORATION

Southboro,
Massachusetts 01772
(617) 485-9100

PROGRAM

Floating Point to Octal

TAPES

Binary: 091-000014

ABSTRACT

This routine converts a number expressed in floating point notation to its internal two word octal representation.

1. REQUIREMENTS

1.1 Memory

2K or larger alterable memory

1.2 Equipment

A teletype with keyboard input (KSR)

1.3 External Subroutines

The Basic Floating Point Package is required
(Binary: 091-000012).

1.4 Other

None

2. OPERATING PROCEDURE

2.1 Calling Sequence

- a. Load the Basic Floating Point Package.
- b. Load Floating Point to Octal.
- c. Enter 000002 in the data switches, press RESET then START.

2.2 Input Format

An "F" will be typed on the teletype, indicating a floating point number is requested. A number should be typed in from the teletype keyboard. Numbers in the following form will be converted:

+nn...nE+mm(break)

where the "n's" represent the decimal mantissa (the first seven non-zero digits will be converted and the remaining digits ignored) and the "m's" represent the decimal characteristic. The signs of the mantissa and characteristic are optional with the default assumed "+". The break character is any character other than

- 1. a decimal digit
- or 2. an "E"

A decimal point is optional, as well as the characteristic. If an error is made on input, rubout can be pressed to clear the string and a new input will be requested.

Some examples of legal inputs are

1*
1.*
1.Ø15*
4E5*
+4E+Ø6*
-5E5Ø*
+3.1415926*
Ø.ØØØ2E-59*

where "*" has been used as the break character.

2.3 Output Format

Two digit octal words will be typed. These two words are the internal representation, in floating point format, of the input. If two consecutive memory words are defined to contain this octal information, they constitute a floating point constant of the input value. For example,

3.1415926 Ø4Ø462 Ø41766

might be a typical input and response. This octal representation can now be used to define "pi" for use with the Floating Point Interpreter.

2.4 Error Returns

If an error is detected in the input, "?" will be typed on another input will be requested.

2.5 State of Active Registers upon Exit

Not applicable

2.6 Cautions to User

None

3. DISCUSSION

3.1 Algorithms

The routine uses the Floating Point Interpreter to convert the input to binary and store it in two consecutive locations. The input conversion OK flag is checked to determine if the input was in error. If no error is detected, the two result words are converted to ASCII octal and printed on the teletype.

3.2 Limitations and Accuracy

Numbers in the approximate range

$$8.7 \cdot 10^{-78} \leq F \leq 7.2 \cdot 10^{75}$$

will be correctly converted.

3.3 Size and Timing

The routine requires 205 (octal) words.

Execution time is limited by the teletype output speed.

3.4 References

The Floating Point Interpreter is described in publication 093-000019.

3.5 Flow Diagrams

None

4. EXAMPLES AND APPLICATIONS

This routine provides a means for defining floating point constants at assembly time. Since the Assembler does not accept floating point values in expressions, it is necessary to know the octal representation for a floating point number in order to define it at assembly time. For example, "pi" could be defined by

```
PI: 40462          ; FIRST WORD OF PI
    41766          ; SECOND WORD OF PI
```

5. PROGRAM LISTING

A listing of floating point to octal follows.

```

; CONVERT A FLOATING POINT INPUT TO OCTAL

; INPUT:          A NUMBER IN ENGINEERING NOTATION TYPED
;                IN FROM THE .TTY KEYBOARD

;                NUMBER OF FORM:
;                (+)MM...MME(+)CC(BREAK)
;                -                -

;                WHERE M IS THE MANTISSA (FIRST SEVEN NON-ZERO
;                DIGITS CONVERTED) AND C IS THE CHARACTERISTIC.
;                PLUS SIGNS ARE OPTIONAL. CHARACTERISTIC IS
;                OPTIONAL AND ASSUMED E00 IF NOT GIVEN.
;                BREAK IS ANYTHING OTHER THAN AN "E" OR
;                A DECIMAL DIGIT. THE DECIMAL POINT MAY
;                BE PLACED ANYWHERE IN THE MANTISSA. IF NOT
;                PRESENT, IT IS ASSUMED TO BE AT THE RIGHT
;                (INTEGER).
;                IF A RUBOUT IS ENCOUNTERED, THE CONVERSION WILL
;                START OVER.

; OUTPUT:         OCTAL PRINTOUT ON THE TTY OF THE
;                TWO WORD FLOATING POINT NUMBER IN
;                PACKED FLOATING POINT FORMAT

; CALLING SEQUENCE:
;                THE ROUTINE IS STARTED BY ENTERING
;                000002 IN THE DATA SWITCHES, PRESSING
;                RESET, THEN START.
;                AN "F" WILL BE TYPED WHENEVER AN INPUT IS
;                REQUESTED

; ERROR CONDITIONS:  ERRORS IN THE INPUT WILL CAUSE
;                A "?" TO BE TYPED AND A NEW
;                INPUT REQUESTED

; CAUTION:        NUMBERS MUST BE IN THE APPROXIMATE
;                RANGE:
;                 $8.7 \times 10^{-78} \leq N \leq 7.2 \times 10^{75}$ 

; REQUIRES:       THE 1K FLOATING POINT INTERPRETER

```

```
000002      •LOC 2
000002 002044      JMP      @44      ; STARTING LOCATION
000003 002044      JMP      @44      ; RESTART LOCATION

000007      •LOC 7
000007 005400 WA:   STR1-100      ; LEAVE 100 WORDS FOR
                                ; INTERPRETER

000040      •LOC 40
000400 005554      IPT              ; ADDRESS OF INPUT ROUTINE
000401 005572      OPT              ; ADDRESS OF OUTPUT ROUTINE

000044      •LOC 44
000440 005500      STRT           ; STARTING ADDRESS

005500      •LOC 5600-100      ; START OF INTERPRETER - LENGTH
                                ; OF THIS ROUTINE

000001 COK=1

055000 006005 STRT:  FINI              ; INITIALIZE INTERPRETER
055001 006004 FDOC:  FEIR              ; ENTER
055002 120001      FDFCI 0           ; CONVERT INPUT
055003 040430      FSTA 0,NUM        ; STORE RESULTS
055004 100000      FEXT              ; EXIT FOR PRINTING
055005 030007      LDA 2,WA          ; GET WIRTABLE AREA ADDRESS
055006 031001      LDA 2,COK,2       ;
055007 151004      MOV 2,2,SRK       ; CHECK FOR INPUT ERROR
055100 000404      JMP FDOC1         ; NO
055110 020416      LDA 0,QUES        ; YES, SEND "?"
055112 004460      JSR OPT
055113 000407      JMP FDOC2

055114 030417 FDOC1: LDA 2,NUM        ; GET FIRST OF TWO WORDS
055115 004420      JSR CVRT          ; CONVERT AND PRINT
055116 020414      LDA 0,SP
055117 004453      JSR OPT           ; OUTPUT SPACE
055200 030414      LDA 2,NUM+1       ; GET SECOND WORD
055201 004414      JSR CVRT          ; CONVERT AND PRINT
055202 020406 FDOC2: LDA 0,CR        ; CARRIGE RETURN
055203 004447      JSR OPT           ; OUTPUT CR, LF
055204 020405      LDA 0,LF         ; LINE FEED
055205 004445      JSR OPT
055206 000753      JMP FDOC          ; LOOP FOR NEXT NUMBER

055207 000077 OUES:  "?"
055300 000015 CR:    15
055301 000012 LF:    12
055302 000040 SP:    40
000002 NUM:        •BLK 2          ; STORAGE FOR FLOATING POINT
                                ; NUMBER
```

↑↑↑

; CONVERT AC2 TO OCTAL AND PRINT RESULTS

```
05535 054415 CVRT: STA 3,SAVR ; SAVE RETURN
05536 126621 SURZ 1,1,SKP ; 100000 TO AC1
05537 132401 CVR1: SUR 1,2,SKP ; DECREASE CURRENT DIGIT BY 1
05540 020413 CVR2: LDA 0,057 ; OCTAL 57
05541 101400 INC 0,0 ; FORM ASCII OUTPUT DIGIT
05542 132533 SURZL# 1,2,SNC ; MINUS IMPLIES DIGIT COMPLETE
05543 000774 JMP CVR1
05544 004426 JSR OPT ; OUTPUT DIGIT
05545 125220 MOVZ 1,1
05546 125220 MOVZ 1,1
05547 125224 MOVZ 1,1,SZ  ; POSITION "1" FOR NEXT DIGIT
05550 000770 JMP CVR2 ; NOT DONE YET
05551 002401 JMP @SAVR ; DONE, RETURN

05552 000000 SAVR: 0
05553 000057 057: 57
```

; INPUT A CHARACTER TO AC0

```
05554 060110 IPT: NIOS TTI ; START AND GET A CHARACTER
05555 063610 SKPDN TTI
05556 000777 JMP .-1
05557 060610 DIAC 0,TTI
05560 024416 LDA 1,MSK
05561 123400 AND 1,0 ; MASK TO 7 BITS
05562 054770 STA 3,SAVR ; SAVE RETURN
05563 004407 JSR OPT ; OUTPUT CHARACTER
05564 024744 LDA 1,CR ; TEST FOR CR
05565 106414 SUB# 0,1,SZ  ;
05566 002764 JMP @SAVR ; NO RETURN
05567 020742 LDA 0,LF ; YES, OUTPUT LINE FEED
05570 004402 JSR OPT
05571 002761 JMP @SAVR
```

; OUTPUT A CHARACTER IN AC0 TO TTY

```
05572 063511 OPT: SKPRZ TTO
05573 000777 JMP .-1
05574 061111 DOAS 0,TT0
05575 001400 JMP 0,3

05576 000177 MSK: 177 ; INPUT MASK
```

.END