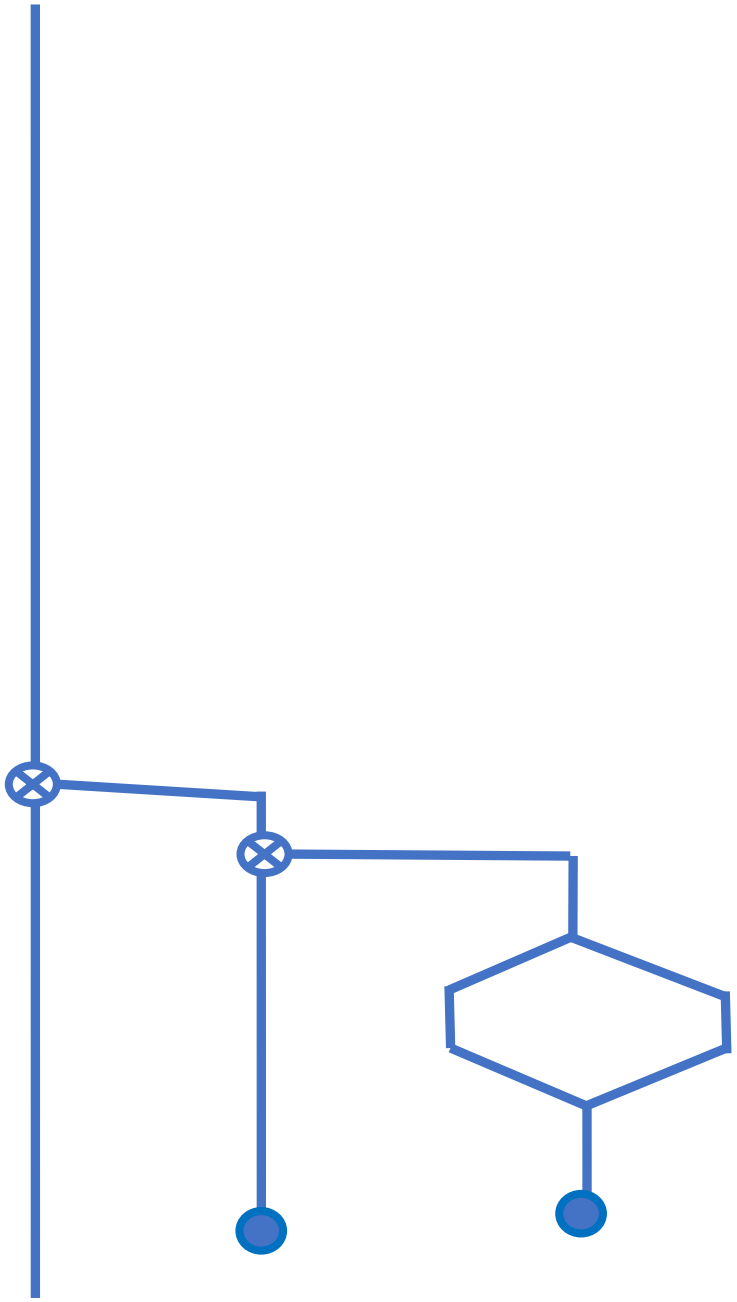

REPRESENTing Floats

SVFIG

May 24, 2025

Bill Ragsdale

REPRESENT

The most powerful word in ANS Forth.

Breaks apart a IEEE 64- bit floating point number.

Useful for printing custom float formats.

IEEE 64 Bit Floating Point

The Sign, 1 bit

The Exponent, 11 bits

The Significand, 52 bits

2.23×10^{-308} to 1.79×10^{308}

16 decimal digits precision.

Definition

REPRESENT (f: f1 ---) (ds: addr n1 --- exp sign flag) From a floating-point number f1, return a counted ascii string of n1 digits at addr rounded to 'precision' significant digits, its decimal exponent, sign and a flag with a TRUE flag.

When flag is FALSE, exp is 7FFFFFFF. The contents of addr are the first n1 characters of text detailing the source of error. e.g. NAN, Infinity and others.

The Actions

- Extract the sign.
- Extract the exponent into a decimal value.
- Extract the Significand, 0 to 1.0
- Round to ‘precision’ digits.
- Format to ‘n’ significant digits.
- Present as an ASCII counted string.
- Allow for NAN, infinity and other exceptions.

The Process

- Prepare two text buffers, $ftemp and output.
- Specify precision and output digits.
- Use REPRESENT
- Then, format your output from left to right.
- Print from ‘addr count’.

My Goal

- Take a float: -1230000.57e
- And present as: $-1,230,000.57

Support in Win32Forth

PLACE (c-addr1 len1 c-addr2 ---) Place string c-addr1,len1 at c-addr2 as a counted string.

c-addr1 is source; c-addr2 is output string.

+PLACE (c-addr1 len1 c-addr2 ---) Append string addr1, len1 to the counted string at addr2.

C+PLACE (c addr ---) Append char c to the counted string at addr.

Support

```
0 value $offset
: Singles      ( singles  --- )
    $ftemp $offset + over pad +place +to $offset ;
: Triple       (  --- )
    $ftemp $offset +    3 pad +place 3 +to $offset ;
: Place-decimal  ascii .  pad c+place ;
: Place-comma    ascii ,  pad c+place ;
```

Formatting, $ and -

```
: DemoB ( fs: f1 ) ( decimals --- addr count )
      {: decimals | exponent sign  :}
   pad off  0 to $offset
   fdup  $ftemp decimals represent 0= abort" error"
   to sign  to exponent              \ find exponent
   decimals exponent + set-precision
   $ftemp  precision  represent drop
           to sign to exponent  \ final process
          ascii $ pad c+place
   sign if ascii - pad c+place then
```

Final Formatting

```
exponent 3 /mod                  \  singles triples
 case
    0 of \ singles ( .x x. xx. )
         ( n ) singles place-decimal endof
    1 of  case ( xxx. x,xxx. xx,xxx. )
          0 of     triple place-decimal                    endof
          1 of  1 singles place-comma triple place-decimal endof
          2 of  2 singles place-comma triple place-decimal endof
          endcase  endof
   2 of  case ( xxx,xxx.,  x,xxx,xxx., xx.xxx.xxx. )
         0 of    triple  place-comma  triple place-decimal endof
         1 of  1 singles place-comma  triple place-comma
                                      triple place-decimal endof
         2 of  2 singles place-comma  triple place-comma
                                      triple place decimal endof
         endcase  endof
      endcase
  decimals singles
  pad count ;
```

Examples

```
Testing DemoB
      -1.5789   $-1.58
     -12.5789   $-12.58
    -123.5789   $-123.58
   -1230.5789   $-1,230.58
  -12300.5789   $-12,300.58
 -123000.5789   $-123,000.58
-1230000.5789   $-1,230,000.58 ok
```