WS 23/24 Numerics Notes

Igor Dimitrov

2023-10-30

Table of contents

Preface 3

1 Floating Point Numbers 4

1.1 ANSI/IEEE 64 Bit 4

Preface

Notes for the lecture "WS 23/24 Numerics 0" at Uni Heidelberg.

1 Floating Point Numbers

1.1 ANSI/IEEE 64 Bit

Let $\tilde{a}$ be a 64 bit IEEE floating point number. $\tilde{a}$ is represented as

```
S E ... E M ... M
```

Where S is the sign bit, 11 E's are the exponent bits and 52 M's are mantissa bits. Interpretation (Case analysis on value of E):

1. S | 0 … 0 | M:
    1. $M = 0 \Rightarrow \tilde{a} = (-1)^S 0$
    2. $M \neq 0 \Rightarrow \tilde{a} = (-1)^S \times 2^{-1022} \times 0.M$ (**subnormal range**)
2. $1 \leq E \leq 2046 \Rightarrow \tilde{a} = (-1)^S \times 2^{E-1023} \times 1.M$ (**normal range**)
3. S | 1 … 1 | M:
    1. $M = 0 \Rightarrow \tilde{a} = (-1)^S$`inf`
    2. $M \neq 0 \Rightarrow \tilde{a} =$ `NaN`(Not a Number) (**exceptions**)

See Figure 1.1 for a visual summary.

Examples:

- `realmin`: smallest normalized positive machine number in FP64 $= [\![$`0|0 ... 01|0 ... 0`$]\!]_{FP64} = 2^{1-1023} \times 1.0 = 2^{-1022}$
- `realmax`: greatest normalized machine number in FP64 $=$ [`0|1...10|1...1`]$_{\texttt{FP64}} \approx$ 1.7977E308
- $1 = 2^0 \times 1.0 = 2^{1023-1023} \times 1.0 =$ [`0|01...1|0...0`]$_{\texttt{FP64}}$
- `eps`: spacing in the interval $(1, 2)$. Note that the spacing is constant for each such interval, but grows as we go further down the number line. That is, the spacing in $(1000, 1001)$ is constant, but larger.
- number right after 1: [`0|01...1|0...1`]$_{\texttt{FP64}}$. Then the spacing, i.e. `eps` from above is 2^{-52}

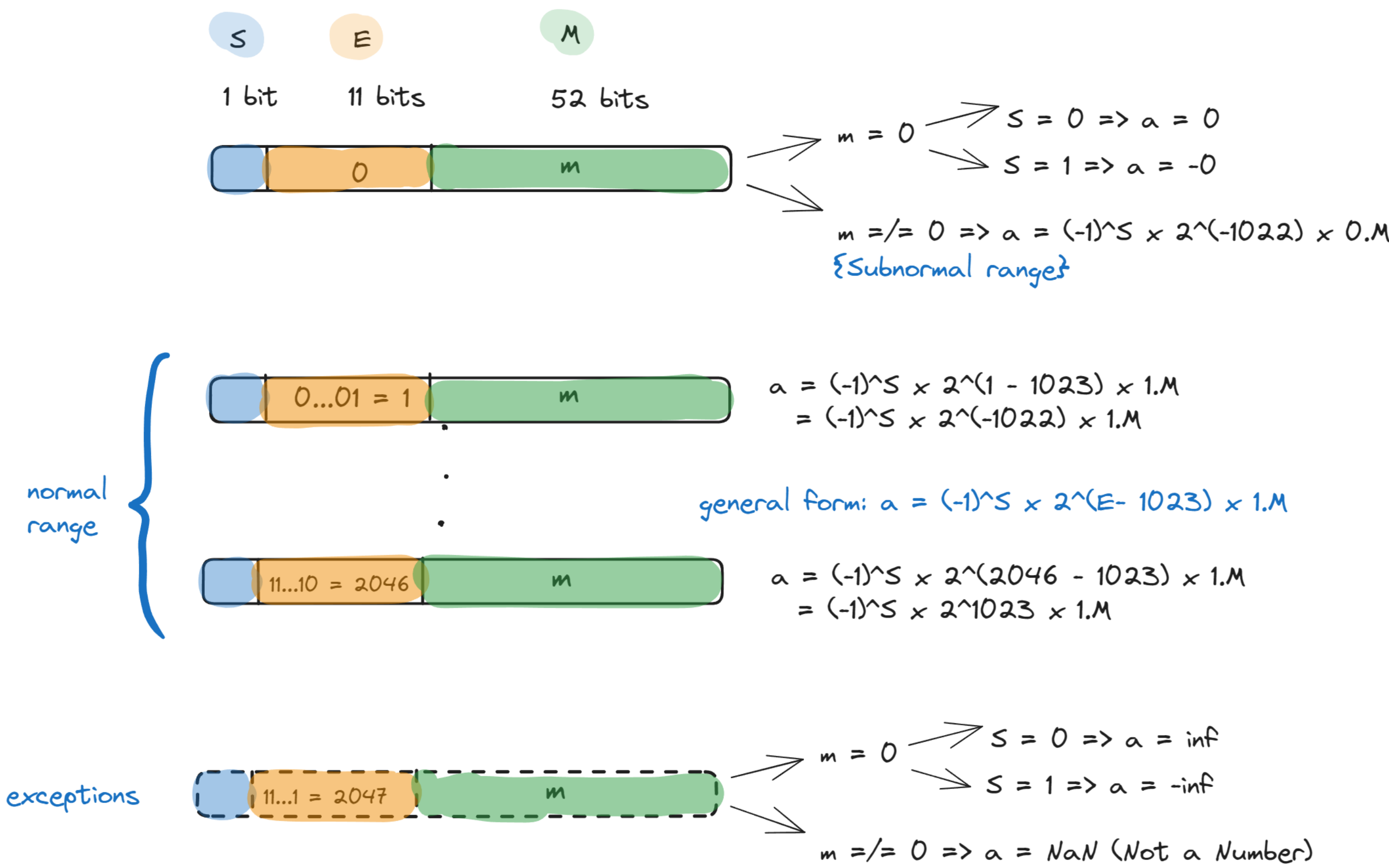

Figure 1.1: floating-point