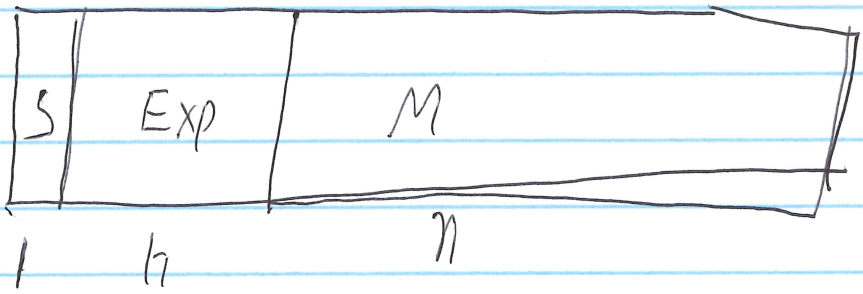
CS2318

09/17/2018

Floating Point Operations



$$(-1)^{S} \times (1.M) \times 2^{E - Bias}$$

Bias is $2^{h-1} - 1$

In IEEE not in assignment

The Exp of $2^{E - bias}$ for max(E) is considered as the indication of exception

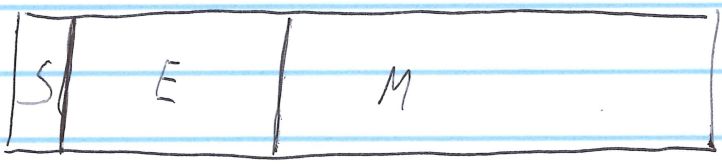
E is "treated" as unsigned

$Max(E) = 2^{h} - 2^{h-1} - 1$ Bias $2^{h-1} - 1$

$\Rightarrow 2^{h-1}$ is the exception

2

Conversion from FP to Decimal



1 5 4 $E = 15$

$(-1)^S \times (1.M) \times 2^{E-15}$

FP to Dec.

1 10110 1101 $= -(1\frac{13}{16})2^7$

0.000075 $(1.M) = ?$ $\frac{1M}{5}$

Dec. to FP

step of $1+|M|+1$ (*2

Significand

3

0.00075	0 not in numby
0.0015	0
0.003	0 not really not
0.006	0 accurate
0.012	0
0.024	0
0.048	0
0.096	0 $1.M = 1.1000 \times 2^{-3}$
0.192	0
0.384	0 -3 BC we moved
0.768	0
0.536	1 the point to the right
0.072	1
	0
	0
	0

0 0 0 0 0 0 0 0 1 1 0 0 0 [0]

I placed 3 instead of 11 might be to small

1 M [2] [1]

LO

4 $A = S_A E_A M_A$

$+ \quad B = S_B E_B M_B$

$C = S_C E_C M_C$

On paper

S + M Rep. might be of h

For computes

3.5×10^{10}

$+ \; 0.5 \times 10^{3}$

2's

$S_A M_A \Rightarrow X$

2's

$+ \quad S_B M_B \Rightarrow Y$

Sig

$S_C M_C \leq 2$

D) Align Exp Assume $E_B < E_A$

$E_A - E_B = h$ SHR [1, M_B] by h

Example $3.5 \times 10^{5} + 0.5 \times 10^{3}$

$= 0.005 \times 10^{5}$

5

1) Align Exp current Exp is E_A

2) Produce $l_1 M_A + l_1 M_B$

Normalize
- Get up to $|M_A| + 2$ significant bits In the case of $|M_A| + 2$
- Shift point to the ~~right~~ left
- => shifting H to the right
- => increment Exp $E_C = E_A + 1$
- In the case of $|M_A| + 1$ $E_C = E_A$

3) ~~E_C~~ ~~$E_C = E_A$~~

$C = S_C E_C M_C$ S_C M_C obtained in the addition

E_A

E_C $E_A + 1$

4) Check for ovF (BC $E_A + 1$ is to big)

5) Normalize again (might be needed)

Example

(6)

A 0 1 0 0 1 1 1 1 0 $+ \left(1\frac{7}{8}\right) \times 2^4$

B + 0 1 0 0 1 0 1 1 0 0 $+ \left(1\frac{3}{4}\right) \times 2^3$

$E_A = 4 \quad E_B \; 3$

$1.M_A = 1.1110 \quad 1.M_B = 1.1100$

$E_C' = 4$

After Align $1.M_B \quad 0.1110$

Add in 2's Result in S+M

$$\begin{array}{r} 1.1110 \\ 0.1110 \\ \hline 10.1100 \end{array} \Rightarrow 1.0110$$

inc E_C' $E_C = 5$ / un biased

0 1 0 1 0 0 0 1 1 0

7 Addition / subtraction

$$\pm \begin{matrix} S_A E_A M_A \\ S_B E_B M_B \end{matrix} \qquad S_C E_C M_C$$

1) Align Exp

2) Add / Sub $1.M_A \pm 1.M_B$

3) Normalize and Round

4) Check for overflow

5) Normalize and round again

$$10.5 \times 10^3 \times 0.05 \times 10^4$$

8 * $S_A E_A M_A$
/ $S_B E_B M_B$ $S_C E_C M_C$

1) $S_C = S_A \oplus S_B$

2) $E_C' = E_A + E_B$ check for ovf

3) $1.M_C = (1.M_A) * (1.M_B)$

Result is $2 \times (1+|M_A|)$

4) Normalize (select the 5 significant bits and Round

5 check for ovf

6) Normalized and round

9

0 1001 1110

* 1 0010 1100

$S_c = 1$

Compiler
- $E_A = X$ - bias $E_B = Y$ - bias
- $E_A + E_B$ at the biased notation
- Subtract 1 bias from $E_A + E_B$

Un biased $E_A = 4$ $E_B = 3$

in bias 7 + 15 1011

11011 XXXX

$1.M_A$ 1.1110 1.111

$1.M_B$ 1.1100 ① 1.110

① .0000 Shift and Add

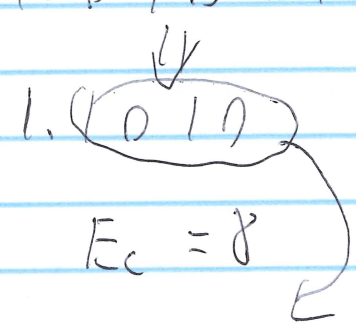
1 0 1111

1111

1111

11 1.01 0010

to 11.0101010



point to the left

Enc Ec

11011110110

Normalize might be needed infinite

1) convert to Binary

2) on addition / multiplication

use only 1+n bits for computing

when we have more than

~~1+m~~ 1+n bits $n = |M_A|$

assume we get

XXX...X

0

000...01.XXXXX

Select the n+1 bits 1.XXXXX...

11 Example

1.1111 0 need to select 4

↓

1.1111

1.1101 xxx

↷

Add for 1.11

1.1101 1

1 n

1.1101.0 round down cut

1.1101 1 Round up add 1

↷ add bit n+2 for roun

1.1101 1 becomes 1.1110

↷