CptS260: Introduction to Computer Architecture

Homework 2 Solution

School of Electrical and Computer Engineering
Spring 2022

Part 1: Float and Double Numbers

1. Consider the 32-bit binary number (11000001010010000000000000000000). Convert this binary to its equivalent decimal, assuming if:

   a) It is a 2's complement signed number

$$N = -2^{31} + 2^{30} + 2^{24} + 2^{22} + 2^{19} = -1052246016$$

   b) It is an unsigned number

$$N = 2^{31} + 2^{30} + 2^{24} + 2^{22} + 2^{19} = 3242721280$$

   c) Sign-magnitude number

   Most significant bit is the sign

$$N = -(2^{30} + 2^{24} + 2^{22} + 2^{19}) = -1095237632$$

   d) Single precision format (IEEE 754)

1 10000010 10010000000000000000000

Sign bit = 1 → negative number
Exponent = 10000010 = 130 – bias = 3
Fraction = $1001 = \frac{1}{2} + \frac{1}{16} = 0.5625$

$$\rightarrow -1.5625 \times 2^{3} = -12.5$$

2. Consider these two decimal numbers: -112.15625, +183.515625.
   a) Show the two numbers in the normalized form.

$$-112.15625 = -1.1215625 \times 10^2, \qquad +183.515625 = +1.83515625 \times 10^2$$

b) Convert the numbers into binary representation.

$$N1 = -1110000.00101000000\ldots0 \qquad N2 = +10110111.100001000\ldots00$$

c) Using the IEEE754 single precision representation, show step by step procedure for addition and multiplication of these numbers.

From part (b), we normalize each number and convert to IEEE 754 representation as follows:

$$N1 = 1\ 10000101\ \ 11000000101000000\ldots0$$
$$N2 = 0\ 10000110\ \ 0110111100001000\ldots00$$

For addition, we first must ensure that each number has the same exponent. We see that N1 has a biased exponent of 133 and N2 has a biased exponent of 134. We always increase the smaller exponent to match the larger exponent. Since we are performing addition, it is also useful to add the implicit 1 to the mantissa so that it is taken into account for addition. The mantissas of both numbers with the implicit 1 is,

M1 = 1$11000000101000000\ldots0$
M2 = 1$0110111100001000\ldots00$

Next, we equalize the exponents. Since N1 has a smaller exponent, we right shift it by 1. Therefore, the mantissas become:

M1 = 01$1100000010100000\ldots0$
M2 = 1$0110111100001000\ldots00$

Now, note that one of the numbers, i.e. N1 is negative. This means that the first mantissa is negative as well. As a result, the numbers need to be subtracted. It can either be done using 2's complement method or direct subtraction. In the following, we will use the 2's complement method and then convert the result back to sign-magnitude form. The 2's complement representation of the two mantissas is

0 0111 0000 0010 1000 0000 0000

1 1000 1111 1101 0111 1111 1111
0 0000 0000 0000 0000 0000 0001

1 1000 1111 1101 1000 0000 0000
0 1011 0111 1000 0100 0000 0000

0 0100 0111 0101 1100 0000 0000

M1 = 1 1000 1111 1101 1000 0000 0000
M2 = 0 1011 0111 1000 0100 0000 0000
M1+M2 = 0 0100 0111 0101 1100 0000 0000

The sign is already positive, which means that we do not need to convert back from 2's complement. Therefore, the mantissa of the sum (with the implicit 1 is)

0100 0111 0101 1100 0000 0000

Notice that the mantissa is not normalized since the implicit 1 is not in the left most position. To normalize the mantissa, we must decrease the exponent by 1 and left shift the mantissa. Therefore, the components of the sum after removing the implicit 1 are:

Sign: 0
Exponent: 10000101
Mantissa: 000 1110 1011 1000 0000 0000

N1 + N2 = 1000111.01011100…0 ➔ 0 10000101 000111010111000…0

The multiplication can be performed similarly by first calculating the exponent and then multiplying the mantissa.

N1*N2 = 01000110101000001100110011011001

d) Show the first number in the IEEE754 double precision format.

N1 = 1 10000000101 000111010111000…0

Difference : 1) bias = 1023, 11 bits exponent

3. $(-1.5625 \times 10^{-1}) \rightarrow (0.15625) \rightarrow$

$0.15625 * 2$	0
$0.3125 * 2$	0
$0.625 * 2$	1
$0.25 * 2$	0
$0.5 * 2$	1

$$(0.15625)_{(10)} \rightarrow (0.00101)_{(2)} \rightarrow (1.01 \times 2^{-3})_{(2)}$$

Sign bit = 1
Mantissa = 0100000000
Exponent = $-3 + 15 = 12_{(10)} = (1100)_{(2)}$

IEEE 754-2008 format : 1 1100 0100000000

4. $(3.41796875 \times 10^{-3} \times 6.34765625 \times 10^{-3}) \times 1.05625 \times 10^{2}$

(a) We first convert each number to its binary form.

$.00341796875 * 2$	0
$0.0068359375 * 2$	0
$0.013671875 * 2$	0
$0.02734375 * 2$	0
$0.0546875 * 2$	0
$0.109375 * 2$	0
$0.21875 * 2$	0
$0.4375 * 2$	0
$0.875 * 2$	1
$0.75 * 2$	1
$0.5 * 2$	1

$$3.41796875 \times 10^{-3} \rightarrow 1.1100000000 \times 2^{-9}$$

(b) $4.150390625 \times 10^{-3} \rightarrow 1.0001000000 \times 2^{-8}$

(c) $1.05625 \times 10^{2} \rightarrow 1.1010011010 \times 2^{6}$

Exp: -9-8 = - 17

Signs: both positive, result positive

```
(A)                      1.1100000000
(B)                    × 1.0001000000
                        --------------
                11100000000
            11100000000
          ----------------------
         1.11011100000000000000
A×B      1.1101110000 00 00000000
```

Guard: 0, Round: 0 and Sticky: 0, there no round is needed

Moreover, the exponent of the result is -17, which results in an underflow. This, the number cannot be represented in the 16-fit half precision floating point.

Part 2: Sequential Multiplier

1. You are asked to multiply two binary numbers using the multiplier provided in lectures for week two. This two binary numbers are 1101, and 10001.
   a) Show your work for multiplication step by step.
   b) How many bits do you need to hold the result?

	Cycle 1	Cycle 2	Cycle 3	Cycle 4	Cycle 5
Product	000000000	000010001	000010001	001010101	011011101
Multiplicand	000010001	000100010	001000100	010001000	100010000
Multiplier	1101	0110	0011	0001	0000

The Result needs 4+5 = 9 bits, and the product is 011011101= 221

2. Convert the two decimal numbers listed below to binary, multiply them using sequential multiplier, and fill the table below with the value of different registers in each clock cycle in multiplication.

a. 7 and 19

$7 = (111)_2$ and $19 = (10011)_2$

	Cycle 1	Cycle 2	Cycle 3	Cycle 4
Product	000000000	000010011	000111001	010000101
Multiplicand	000010011	000100110	001001100	010011000
Multiplier	111	011	001	000

b. 1000 and 25

$1000 = (1111101000)_2$ and $25 = (11001)_2$

	Cycle 1	Cycle 2	Cycle 3	Cycle 4	Cycle 5	Cycle 6
Product	000000000000000	000001111101000	000001111101000	000001111101000	010001100101000	110000110101000
Multiplicand	000001111101000	000011111010000	000111110100000	001111101000000	011111010000000	111110100000000
Multiplier	11001	01100	00110	00011	00001	00000