CS2022 ASSIGNMENT 5 (Due Date: Oct 14, 2022)

Question 1

To represent a decimal number in a binary format, fixed-point binary number representation is one option. As re-iterated many times in the lectures, only a finite number of *decimal numbers* (equi. real numbers) can be represented by a fixed-point binary number without error. For a decimal number which cannot be exactly represented by a fixed-point binary number, it will still be enforced to be represented. Clearly, rounding error will be incurred.

Let x be the decimal number to be represented and $\hat{x}$ be the value of the fixed-point binary number representing the decimal number x. Rounding error is denoted by $\epsilon(x)$ which is a function of x. The rounding error of x is defined as follows :

$$\epsilon(x) = |x - \hat{x}|,$$

the absolute value between the actual decimal number and the value of its corresponding binary number.

Sign-Magnitude

It is assumed that the fixed-point binary number format is defined as follows :

$$saaaaabbbbbbbbbb$$

where one bit is used as the sign bit, five bits are used for representing the integer part and 10 bits are used for representing the fractional part.

(a) What is the minimum value of the numbers being encoded in the above sign-magnitude fixed-point format?

(b) What is the maximum value of the numbers being encoded in the above sign-magnitude fixed-point format?

(c) What is the binary pattern of the fixed-point binary number for the decimal number +6.4 and the rounding method is round-to-nearest?

(d) Based on the answer obtained in (c), what is the rounding error?

(e) What is the binary pattern of the fixed-point binary number for the decimal number −6.4 and the rounding method is round-to-nearest?

(f) Based on the answer obtained in (e), what is the rounding error?

(g) Based on the answer obtained in (d) and (f), are these rounding errors the same?

Answer:

(a) $-(2^5 - 2^{-10})$.

(b) $(2^5 - 2^{-10})$.

(c) 0001100110011010.

(d) 0.000390625.

(e) 1001100110011010.

(f) 0.000390625.

(g) Yes.

2's Complement

In the lecture note, the method for obtaining the 2's complement of a negative fixed-point binary number has been introduced.

(h) What is the minimum value of the numbers being encoded in the 2's complement fixed-point format?

(i) What is the maximum value of the numbers being encoded in the 2's complement fixed-point format?

(j) What is the binary pattern of the 2's complement fixed-point binary number for the decimal number +6.4 and the rounding method is round-to-nearest?

(k) Based on the answer obtained in (c), what is the rounding error?

(l) What is the binary pattern of the 2's complement fixed-point binary number for the decimal number −6.4 and the rounding method is round-to-nearest?

(m) Based on the answer obtained in (l), what is the rounding error?

(n) Based on the answer obtained in (k) and (m), are these rounding errors the same?

Answer:

(h) $-(2^5 - 2^{-10})$.

(i) $(2^5 - 2^{-10})$.

(j) 0001100110011010.

(k) 0.000390625.

(l) 1110011001100110.

(m) 0.000390625.

(n) Yes.

Question 2

(a) What is the rounding error of +6.4 which is represented by the 16-bit sign-magnitude fixed-point number format defined with round-to-nearest rounding method?

(b) What is the rounding error of −6.4 which is represented by the 16-bit sign-magnitude fixed-point number format defined with round-to-nearest rounding method?

(c) What is the rounding error of +6.4 which is represented by the 16-bit 2's complement fixed-point number format defined with round-to-nearest rounding method?

(d) What is the rounding error of −6.4 which is represented by the 16-bit 2's complement fixed-point number format defined with round-to-nearest rounding method?

(e) What is the rounding error of +6.4 which is represented by the 16-bit sign-magnitude fixed-point number format defined with round-by-chop rounding method?

(f) What is the rounding error of −6.4 which is represented by the 16-bit sign-magnitude fixed-point number format defined with round-by-chop rounding method?

(g) What is the rounding error of +6.4 which is represented by the 16-bit 2's complement fixed-point number format defined with round-by-chop rounding method?

(h) What is the rounding error of −6.4 which is represented by the 16-bit 2's complement fixed-point number format defined with round-by-chop rounding method?

Answer:

(a) 0.000390625.

(b) 0.000390625.

(c) 0.000390625.

(d) 0.000390625.

(e) Note that the binary number of +6.4 is represented by 00001100110011001 if the rounding method is round-by-chop. So, the rounding error is 5.859375×10^{-4}.

(f) The binary number of +6.4 is represented by 10001100110011001. So, the rounding error is again 5.859375×10^{-4}.

(g) 5.859375×10^{-4}.

(h) 5.859375×10^{-4}.

Question 3

(a) What is the maximum rounding error of a positive decimal number which is represented by the 16-bit sign-magnitude fixed-point number format defined with round-to-nearest rounding method?

(b) What is the maximum rounding error of a negative decimal number which is represented by the 16-bit sign-magnitude fixed-point number format defined with round-to-nearest rounding method?

(c) What is the maximum rounding error of a positive decimal number which is represented by the 16-bit 2's complement fixed-point number format defined with round-to-nearest rounding method?

(d) What is the maximum rounding error of a negative decimal number which is represented by the 16-bit 2's complement fixed-point number format defined with round-to-nearest rounding method?

(e) What is the maximum rounding error of a positive decimal number which is represented by the 16-bit sign-magnitude fixed-point number format defined with round-by-chop rounding method?

(f) What is the maximum rounding error of a negative decimal number which is represented by the 16-bit sign-magnitude fixed-point number format defined with round-by-chop rounding method?

(g) What is the maximum rounding error of a positive decimal number which is represented by the 16-bit 2's complement fixed-point number format defined with round-by-chop rounding method?

(h) What is the maximum rounding error of a negative decimal number which is represented by the 16-bit 2's complement fixed-point number format defined with round-by-chop rounding method?

Answer:

(a) 2^{-11}.

(b) 2^{-11}.

(c) 2^{-11}.

(d) 2^{-11}.

(e) 2^{-10}.

(f) 2^{-10}.

(g) 2^{-10}.

(h) 2^{-10}.

Question 4

Consider two positive decimal numbers x and y whose values are +6.4 and +3.6. The sum of these two numbers is clearly +10. Let z and $\hat{z}$ be the actual result of $x + y$ and result obtained after the binary addition.

$$z = x + y, \quad \hat{z} = \hat{x} + \hat{y}.$$

(a) What is the binary pattern of the decimal number +3.6 if it is represented by 16-bit sign-magnitude fixed-point binary number format as defined in Question 1 and the rounding method is round-to-nearest?

(b) Based on the result in (a), what is the rounding error?

(c) Performing the binary addition of $\hat{x}$ and $\hat{y}$, where $\hat{x}$ is the fixed-point binary number for +6.4 and $\hat{y}$ is the fixed-point binary number for +3.6, what is the fixed-point binary number for the result $\hat{z}$?

(d) Based on the result in (c), what is the rounding error $|z - \hat{z}|$?

(e) What is the binary pattern of the decimal number +3.6 if it is represented by 16-bit sign-magnitude fixed-point binary number format as defined in Question 1 and the rounding method is round-by-chop?

(f) Based on the result in (e), what is the rounding error?

(g) Performing the binary addition of $\hat{x}$ and $\hat{y}$, where $\hat{x}$ is the fixed-point binary number for +6.4 obtained in Question 2(e) and $\hat{y}$ is the fixed-point binary number for +3.6 obtained in (e), what is the fixed-point binary number for the result $\hat{z}$?

(h) Based on the result in (g), what is the rounding error $|z - \hat{z}|$?

Answer:

(a) 0000111001100110.

(b) The number in decimal is +3.599609375. So, the rounding error is 3.90625×10^{-4}.

(c) +10.

(d) 0.

(e) 0000111001100110.

(f) 3.90625×10^{-4}.

(g) 9.9990234375.

(h) 9.765625×10^{-4}.

Question 5

Eight-bit unsigned integer format is used for the representation of the light intensity of a pixel in an image. Light intensity with zero is defined as no light, i.e. the darkness. Light intensity with 255 is defined as the brightest. For a color image which is encoded in bitmap format, each pixel is encoded by three positive integers. One integer is for the red color light intensity. One integer is for the green color light intensity. One integer is for the blue color light intensity.

(a) For a color image of size 1024×1024, how many pixels are there in the image?

(b) Excluding the necessary information to be encoded in the file header, what is the total number of bytes for encoding the above image?

(c) If the light intensity level is encoded by the IEEE 754 floating-point format, what is the total number of bytes for encoding the above image?

(d) For a computer, the size of a color screen could be of the size 1280×1024. Excluding the necessary information to be encoded in the file header, what is the total number of bytes for encoding a graphical display on a computer screen?

Answer:

(a) 2^{20} (equivalently, 1048576) pixels.

(b) 3×2^{20} (equivalently, 3145728) bytes.

(c) As each number to be encoded by the IEEE 754 floating-point format requires 32 bits (equivalently, 4 bytes), it needs 12×2^{20} bytes memory for the image.

(d) The total number of pixels of a graphical display is 10×2^{18}. It needs 30×2^{18} bytes memory for encoding a graphical display.

Question 6

(a) For the six logic gates that have been introduced in the lectures, one of them is considered to be very special as any logic gate can be implemented by a combination of this logic gate. What is it?

(b) Today, almost all computers their architectures must consist of a processor broad, a memory broad, an input device, an output device and a network communication card. What is this architecture originated from?

(c) State the reason(s) why we need to have the floating-point format for number representation?

(d) In any processor, there must have so many logic gates, switches and connectors. What are the purposes for having those switches and connectors in the processor?

Answer:

(a) The NAND gate.

(b) von Neumann architecture (equivalently, von Neumann model).

(c) First, the range of numbers to be represented by floating-point format is much wider than the range of numbers to be represented by fixed-point format of the same number of bits. Second, the rounding error (resp. precision) of a number is much smaller (resp. much higher).

(d) The switches and the connectors are used for controlling the flow of signals in the processor and the read/write action to a register.