Computer Sciences Department

University of Wisconsin-Madison

CS/ECE 552 – Introduction to Computer Architecture

In-Class Exercise (04/28)

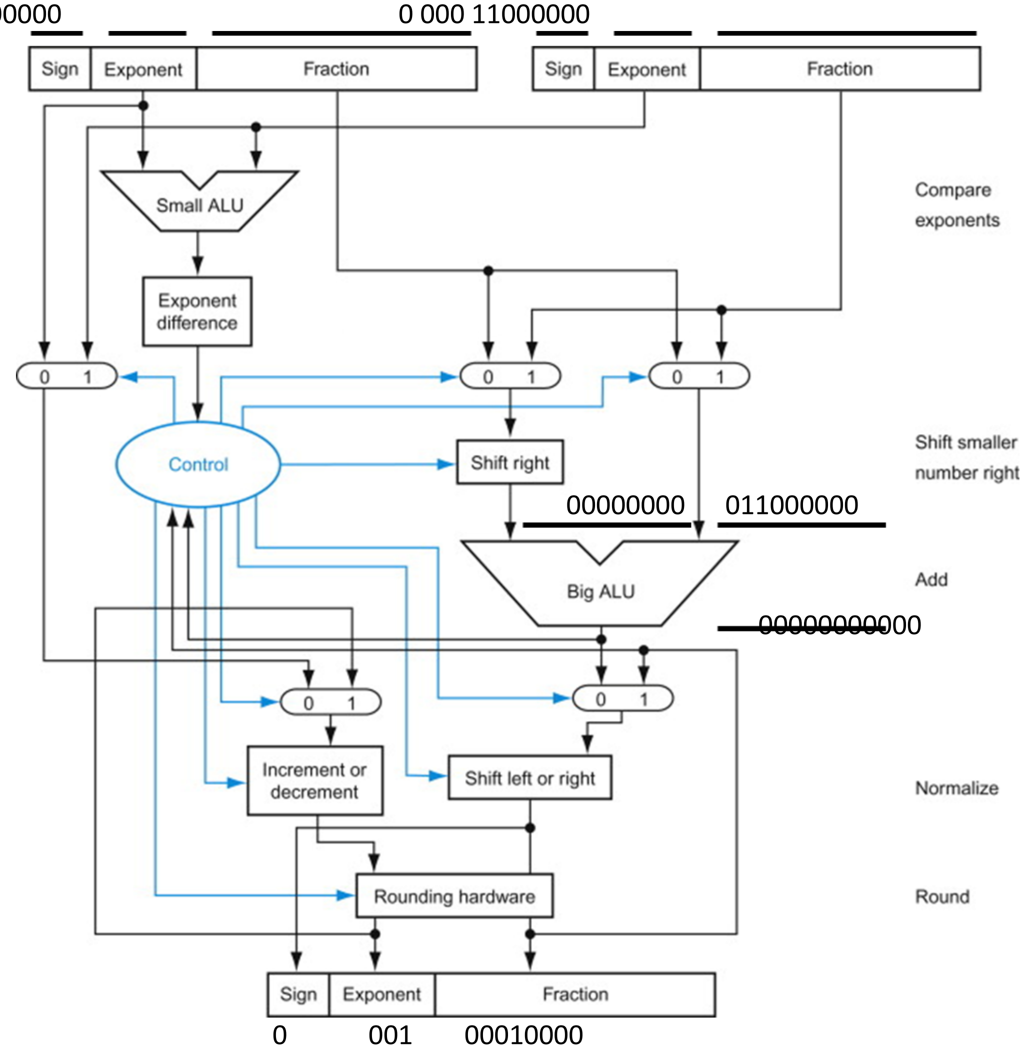
Answers to all questions should be uploaded on Canvas.

1. [7 points] Assume a simplified normalized floating-point representation:

S	Exp			Significand							

Similar to the IEEE 754 floating point standard, this simplified floating point representation has a 1-bit sign (S), a 3-bit biased exponent (E) and an 8-bit significand/fraction (F) with an **implicit leading 1**. The exponent E uses a **bias of 3**. Fill in the blanks in the adder diagram below (Figure 3.15 from your textbook) when adding decimal numbers **0.375 (left) + 1.75 (right)**:

0 110 10000000 0 000 11000000



2. [3 points] You are asked to design a new architecture for machine learning on the edge. The architecture should be extremely low power. You observe that for your machine learning applications, the IEEE 754 floating point representation is greatly over-provisioned. You propose to use a custom normalized floating-point representation that has far less exponent and significand bits. Your representation should have the following simplified format:

Exp (with Bias)	Significand (with implicit leading 1)

With your representation, the value is simply computed as:

$$\mathbf{1.Significand \times 2^{(Exp - Bias)}}$$

Assume that you do not need a sign bit nor special values (e.g., zero, infinity). After profiling your machine learning applications, you find that you only need to represent numbers within the range of **0.140625 and 19.25**. Design a custom floating-point representation (in the format above) that requires the **minimum number of bits** to represent both 0.140625 and 19.25. You do not need extra precision for numbers in between. Specifically, specify:

(i) Number of bits for **Significand**
(ii) Number of bits for **Exp**
(iii) Value of **Bias**

6 2 7