CS M152A Lab 2

Bolun Hu

Sung Joon Baek

**Introduction**

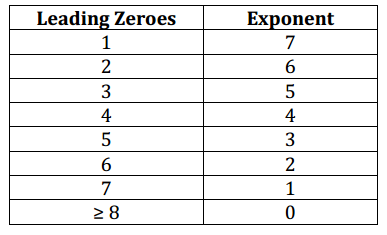
In this lab we are going to build a converter which translate a 12-digit binary input into an 8-bit floating point representation. The 8-bit output has the following desired format:



In the above representation, S indicates the sign bit; E represents the exponential value; F is the significand. The value of this representation can be calculated by the following equation:



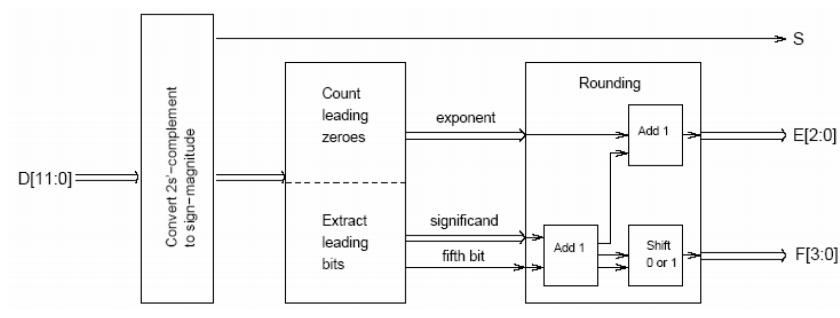
We can get the values of S, E and F from the 12-bit input. Specifically, S is represented by the first bit of the 12-bit inputs; E can be calculated by counting the number of leading zeros and subtract this number from 8. Namely:



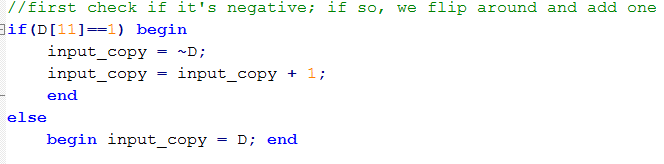
Significand is the 4-bit binary number after the last leading zero. We also have to deal with issues including overflow and rounding. We will talk about them in the later sections.

**Design Description**

Our overall design idea follows the guidance of the lab manual:



We first check the sign of the input. If the 12-bit input has 0 on its most significant bit, then we do nothing and proceed to next step. Otherwise, if it has 1 on its MSB, then we use 2’s complement rule to convert the input into its absolute value, namely by flipping the bits and add one. The related code is as follows:



After that, we then directly get the S from the most significant bit from the input: