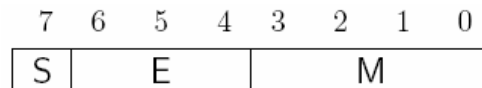


**CS m51A: Logic Design of Digital Systems**  
**UCLA Computer Science Department**  
**Fall 2009**

**Project 2 – Large Digital System Design**  
**Due Friday, December 4, 2009 12:00 Noon Sharp**

Provide a schematic that implements the multiplication of two floating point values, and outputs the resulting product in normalized form.

We will use a simplified floating-point representation consisting of one sign bit, a 3-bit *exponent*, and a 4-bit *significand* (also called the *fraction* or, somewhat inaccurately, the *mantissa*):



The value represented by an 8-bit byte in this format is

$$V = (-1)^S \cdot M \cdot 2^E$$

The *S* bit signifies the sign of the number. The 4-bit significant *M* ranges from (0000) = 0 to (1111) = 15, and the exponent ranges from (000) = 0 to (111) = 7.

The following table shows the values corresponding to several FP representations.

Floating Point Representation Examples		
FP representation	Formula	Value
0   000   0000	$+0 \times 2^0$	0
1   010   1010	$-10 \times 2^2$	-40
0   011   0111	$+7 \times 2^3$	56
0   010   1110	$14 \times 2^2$	56

The last two rows of the above table demonstrate that some numbers have multiple FP representations. The preferred representation is the one in which the most significant bit of the significant is 1; this representation is said to be *normalized*.

It is possible to have your input values not be normalized. Also, it is possible to have as an input a dirty zero, where the significant is zero, but the exponent is not zero. However, output values must be normalized and output zeros must be clean zeros, with a significant and an exponent of zero.

In case of an overflow, the overflow bit is set.

The pins for the logic block have the following uses:

Input	SA	Sign bit of the first floating point value
Input	FA[3:0]	4-bit significand of the first floating point value
Input	EA[2:0]	3-bit exponent of the first floating point value
Input	SB	Sign bit of the second floating point value
Input	FB[3:0]	4-bit significand of the second floating point value
Input	EB[2:0]	3-bit exponent of the second floating point value
Output	SR	Sign bit of the product
Output	FR[3:0]	4-bit normalized significand of the product
Output	ER[2:0]	3-bit exponent of the product
Output	Overflow	Bit set if multiplication resulted in an overflow

You can use the modules and gates presented in class or given in the text book. The modules that you design must have specific functionality. For extra credit, you can simulate your design and provide a timing diagram, with various appropriate inputs tested.

### **Materials to Turn in Using Your Submit Folder**

1. Final Design
2. Cleanly presented material demonstrating your design process
3. A maximum of a two page write-up explaining your design

### **Grading**

Readability – 10%

Correctness – 60%

Design Quality – 30%

Extra Credit – up to 12% for simulating your results