Computer Organization and Structure

Homework #3
Due: 2010/11/16

1. Given the bit pattern:

   1010 1101 0001 0000 0000 0000 0000 0010

   what does it represent, assuming that it is

   a. A two's complement integer?
   b. An unsigned integer?
   c. A single precision floating-point number?
   d. A MIPS instruction?

2. We have shown how to add and subtract binary and decimal numbers. However, other numbering systems were also very popular when dealing with computers. Octal (base 8) numbering system was one of these. The following table shows pairs of octal numbers.

	A	**B**
1	5323	2275
2	0147	3257

   a. What is the sum of A and B if they represent unsigned 12-bit octal numbers? The result should be written in octal. Show your work.
   b. What is the sum of A and B if they represent signed 12-bit octal numbers stored in sign-magnitude format? The result should be written in octal. Show your work.
   c. Convert A into a decimal number, assuming it is unsigned. Repeat assuming it stored in sign-magnitude format. Show your work.

   The following table also shows pars of octal numbers.

	A	**B**
1	2762	2032
2	2646	1066

   d. What is A - B if they represent unsigned 12-bit octal numbers? The result should be written in octal. Show your work.
   e. What is A - B if they represent signed 12-bit octal numbers stored in sign-magnitude format? The result should be written in octal. Show your work.
   f. Convert A into a binary number. What makes base 8 (octal) an attractive numbering system for representing values in computers.

3. A majority function is generated in a combinational circuit when the output is equal to 1 if the input variables have more 1's than 0's. The output is 0 otherwise.

   a. Please write the truth table for a 4-input majority function.

b. Please use the Karnaugh map to find the minimum sum of products form and the minimum sum of products form for the complement.
c. Please draw the logic schematic by using AND, OR, and INVERT gates.

4. The ALU supported set on less than (slt) using just the sign bit of the adder. Let's try a set on less than operation using the values -7_{ten} and 6_{ten}. To make it simpler to follow the example, let's limit the binary representations to 4 bits: 1001_{two} and 0110_{two}.

   $1001_{two} - 0110_{two} = 1001_{two} + 1010_{two} = 0011_{two}$

   This result would suggest that $-7_{ten} > 6_{ten}$, which is clearly wrong. Hence we must factor in overflow in the decision. Modify the 1-bit ALU in the following figures to handle slt correctly.

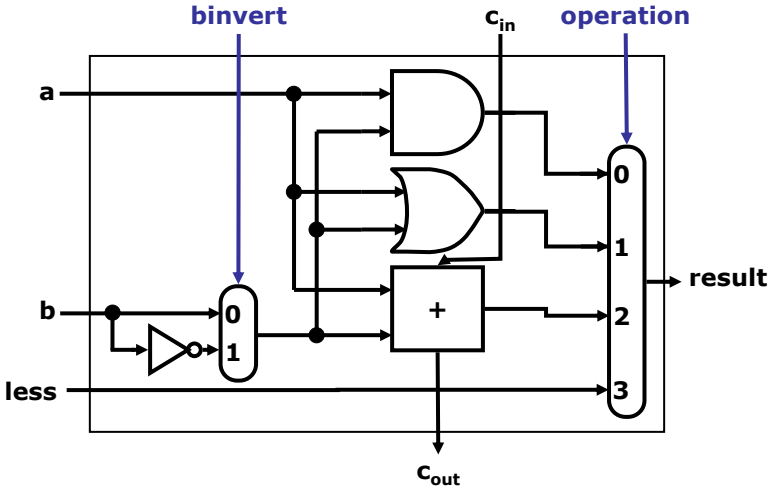

Figure 1: A 1-bit ALU that performs AND, OR, and addition on a and b or b'.

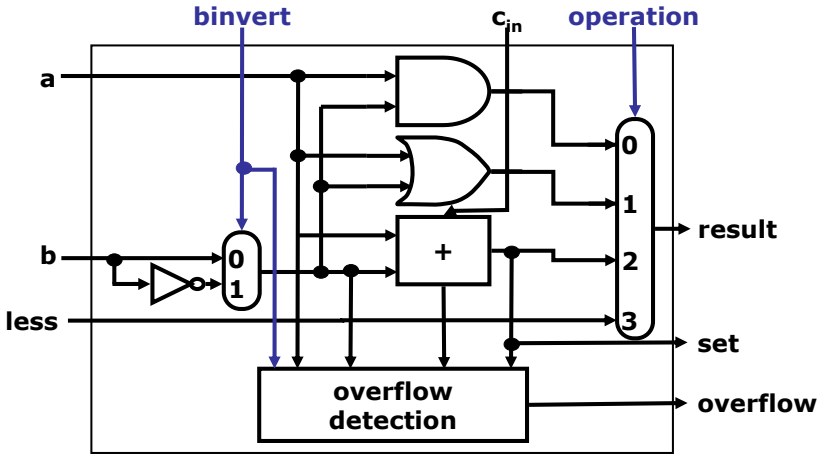

Figure 2: A 1-bit ALU for the most significant bit.