Cpr E 281 HW07

ELECTRICAL AND COMPUTER ENGINEERING
IOWA STATE UNIVERSITY

Arithmetic Circuits
Assigned Date: Eighth Week
Due Date: Oct. 21, 2013

P1. (10 points) Show that the circuit in Figure 3.4 implements the full-adder specified in Figure 3.3(a).

P2. (10 points) By considering the signs of the operands and the result in the addition of two n-bit 2's complement numbers, prove that Overflow = $c_n \oplus c_{n-1}$.

P3. (10 points) Consider the 1-bit subtract unit described in the lecture notes. Write the canonical SOP expressions for functions R and W (Borrow out) based on A, B and W(in).

P4. (10 points) Design a circuit to add 1 to a given n-bit number (i.e., design an increment-by-1 circuit) using n half-adders.

P5. (10 points) Represent the decimal number -9.125 in IEEE 754 single-precision floating-point format.

P6. (6 points) What is the decimal value of the following IEEE 754 single-precision floating point number?

00111110 11001000 00000000 00000000

P7. (10 points) Design a 8x1 multiplexer using seven 2x1 multiplexers.

P8. (10 points) The question considers the design of a 8x1 multiplexer using gates. Assume the data inputs are I0,...,I7 and the select inputs are S2, S1 and S0.

(a) Write a sum-of-products expression for the 8x1 multiplexer.

(b) Implement the expression in part (a) using NOT and NAND gates with any number of inputs. Please use as few gates as possible.

Arithmetic Circuits
Assigned Date: Eighth Week
Due Date: Oct. 21, 2013

P9. (10 points) A 4-bit rotator circuit that rotates the input bits to the left by 0, 1, 2 or 3 bit positions is called a left-rotator. A right-rotator is the same as a left-rotator except that it rotates to the right instead of to the left. In this question, you are asked to design a right-rotator circuit based on a left-rotator.

(a) Design a right-rotator by modifying the inputs X Y to the left-rotator by some AND, OR, and NOT gates.

(b) Design a right-rotator with a left-rotator and without any other gate.

P10. (10 points) Consider an arithmetic circuit. Its input consists of two 3-bit 2's complement numbers $A=A_2\ A_1\ A_0$, $B=B_2\ B_1\ B_0$, and two selection bits S1 and S0. Its output consists of a 3-bit 2's complement number $F=F_2\ F_1\ F_0$. The circuit performs the following operations. We will ignore overflow in this question. Implement this circuit using one 3-bit adder, one 3-bit 2-to-1 multiplexer and a minimal number of gates.

S1	S0	F
0	0	A+B
0	1	A+1
1	0	A-B
1	1	A-1