LSU EE 3755 **Homework 3** **Due: 15 October 2012**

Problem 1: The module below performs subtraction naïvely, using two adders. If the synthesis program does not see it, the resulting hardware will use the two adders. Re-write the module so that it uses one adder, even before the synthesis program applies optimization.

```
module subtract(diff, a, b);
   input a, b;
   output diff;

   wire [31:0] a, b, diff;

   wire [31:0] bnot = ~ b;   // Perform bitwise-negation.
   wire        cout1, cout2; // Ignore values.
   wire [31:0] bneg;

   // Ripple_add_32 ports: carry-out, sum, addend, addend, carry-in.
   ripple_add_32 a1(cout1, bneg, bnot, 32'd1, 1'b0);
   ripple_add_32 a2(cout2, diff, a,    bneg,  1'b0);

endmodule
```

Problem 2: Memorize the Boolean expressions to compute the generate and propagate signals in a carry lookahead adder. Show a Boolean expression needed to generate carry in 5 of a flat carry lookahead adder, where carry in 0 is the carry in to the least significant bit. In your Boolean expression use p_i for the propagate signal from bit i, for $i \geq 0$ and use g_i for the generate signal from bit i, for $i \geq 0$.

Problem 3: Show the work for long-hand multiplication of $6b_{16} \times 27_{16}$ using radix-16 multiplication (four bits at a time). See the examples in lecture set 7, `http://www.ece.lsu.edu/ee3755/2012f/l07.v.html`.

Problem 4: Show the work for long-hand multiplication of $6b_{16} \times 27_{16}$ using radix-4 Booth recoding. Remember that with Booth recoding some of the partial products can be negative, so remember to sign extend as long as necessary. Please make an effort to arrive at the correct answer, which is $104d_{16}$.