Professor Fearing EECS150/Problem Set 1 Solution Fall 2013

**Due at 10 am, Thu. Sep. 12**

Reading Harris and Harris: 1.1-1.6, 2.1-2.6, 4.1-4.3, 4.5, 4.8

1. **(10 pts) Base Representation.** Perform the following number system conversions (assume unsigned binary):

   a. $0010\ 0011\ 1001_2 = 1071_8 = 569_{10} = 239_{16}$

   b. $111.1001_2 = 7.44_8 = 7.5625_{10} + 7.9_{16}$

   c. $0001\ 1010\ 0000_2 = 640_8 = 416_{10} = 1A0_{16}$

   d. $11.0001\ 0011\ 0011_2 = 3.0463_8 = 3.07495117188_{10} = 3.133_{16}$

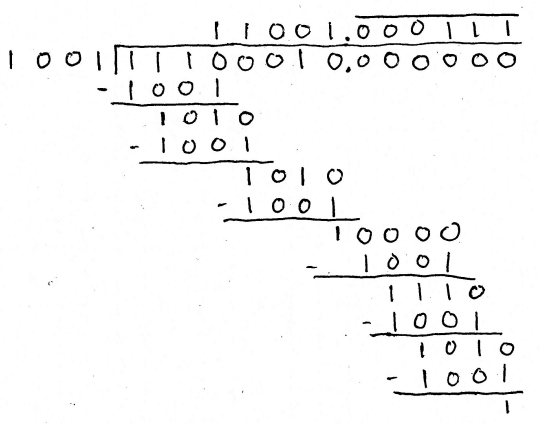
   e. $1110\ 1000.1_2 = 350.4_8 = 232.5_{10} = E8.8_{16}$

2. **(15 pts) Arithmetic.** Showing your calculations:

   a. Add $108_{10}$ and $117_{10}$ in base 2, 10, and sixteen.
   $108_{10} + 117_{10} = 1110\ 0001_2 = 225_{10} = E1_{16}$

   b. Multiply $123_4$ and $231_4$ in base 2, 10, and sixteen.
   $123_4 * 231_4 = 0100\ 1011\ 1111_2 = 1215_{10} = 4BF_{16}$

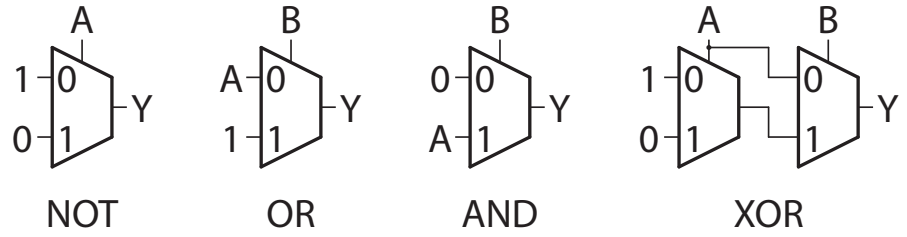
   c. Use long division to find $1110\ 0010_2$ divided by $1001_2$.
   $1110\ 0010_2/1001_2 = 0001\ 1001.\overline{000111}$



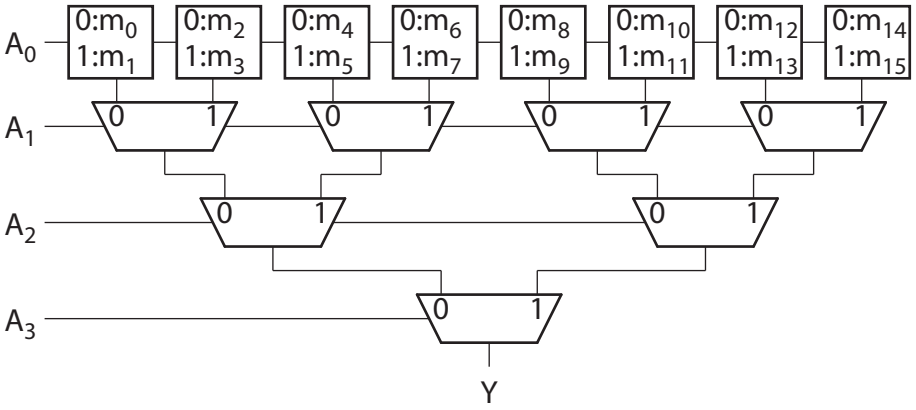

3. **(15 pts) DC Transfer Characteristics.** Harris & Harris exercise 1.78.

   Yes, the device can be used as an inverter with $V_{IL} = 2.5V, V_{OL} = 1.5V, V_{IH} = 3V, V_{OH} = 4V$, with $NM_L = NM_H = 1V$. These points were chosen to be at values of the function where the slope passes through -1 (this the heuristic method mentioned in the book) which maximizes the noise margins. Other points satisfy the -1 slope condition, but no other combination of points satisfies the conditions $V_{OL} < V_{IL} < V_{IH} < V_{OH}$.

4. **(10 pts) 2-input Mux.** Using a minimum number of 2:1 muxes, draw a diagram showing designs for NOT, and 2-input AND, OR, and XOR functions. You may use static inputs "0" and "1" to the muxes in addition to the function inputs.




5. **(10 pts) Look-up Tables.** You are given a look-up table (LUT) with 2 binary addresses and one input. Show how these 1 input LUTs can be combined to perform ANY 4 input logic function.




6. **(10 pts) Boolean algebra** Reduce the following Boolean expressions using theorems and identities and express in Sum-of-Products form (luckily now done by synthesis tools):

   a. $F = \overline{A(\overline{B+\overline{C}})D} = \overline{A} + B + \overline{C} + \overline{D}$

   b. $F = \overline{A \cdot \overline{B} \cdot \overline{C}} = \overline{A} + B + C$

   c. $F = (A + \overline{B})(\overline{A} + B + C)(\overline{A} + B + \overline{C}) = A \cdot B + \overline{A} \cdot \overline{B}$

7. **(15 pts) Verilog.** Write a Verilog module (with correct syntax) called `majority`. It receives three inputs `a, b,` and `c`. It produces one output `y` which is TRUE if 2 or more of the inputs are TRUE.

```
module majority(input a,b,c, output y);

  assign y = (a&b) | (b&c) | (a&c);

endmodule
```

8. **(15 pts) Verilog.** Write a Verilog module (with correct syntax) called `PriorityEncoder` with 8 inputs `A[7:0]`, 3 bit output `Y[2:0]`, and binary output `None`. `Y` encodes the most significant "1" input bit in `A`. Output `None` is TRUE if all inputs `A[7:0]` are "0". For example if `A =` 0110 $1111_2$ then `Y =` $110_2$. (This is a priority encoder.)

```
module PriorityEncoder(input A[7:0], output Y[2:0], None);
```

```
  assign None = A == 8'd0;
  assign Y = A[7] ? 3'd7 :
             (A[6] ? 3'd6 :
             (A[5] ? 3'd5 :
             (A[4] ? 3'd4 :
             (A[3] ? 3'd3 :
             (A[2] ? 3'd2 :
             (A[1] ? 3'd1 : 3'd0))))));

endmodule
```