ES 4 Midterm study guide

1. Numbers

- Convert an unsigned integer from decimal to binary and back
- Add numbers in binary

2. Boolean equations, truth tables, and circuits

- Write a 2-input truth table for each of the following logic operations: AND, OR, NOT, NAND, NOR, XOR, XNOR
- Write truth tables for AND, OR, NAND, and NOR gates with 3 or more inputs.
- Write a boolean equation from an English description of a logic system, and vice versa, using the rules of operator precedence.
- Draw a logic gate diagram from a boolean equation, and write a boolean equation from a logic gate diagram
- Write a truth table from a logic equation by evaluating it for each value of the inputs
- Write a boolean equation from a truth table using either canonical form (sum of products and product of sums)

Lab 1

- Interpret datasheets for 74-series logic gate ICs, and use this information to draw a pin/package diagram from a logic diagram

3. Manipulating boolean equations

- Use boolean algebra to manipulate boolean equations, using the 12 theorems listed in the book (Commutativity, associativity, distributivity, De Morgan)
- Use "bubble pushing" to manipulate the logic of a circuit
- Use the above logic manipulation techniques to minimize or implement a circuit with constraints (e.g., only NAND gates)

4. Minimizing logic

- Draw a Karnaugh map for a 3 or 4-input truth table, and use it to create a minimal boolean equation for the truth table.
- Find a minimal boolean equation for a truth table with "don't care" outputs

5. Multiplexers and FPGAs

- Explain what a multiplexer is, and write out a truth table for a N:1 multiplexer (where N is 2, 4, 8, 16)
- Draw a logic diagram using a 2^M-input multiplexer to implement a boolean equation with M variables
- Describe the basic structure of an FPGA (i.e., explain to someone who just started the class what an FPGA is)

6. Timing combinational logic

- Given a circuit and timing information about the gates, calculate the propagation delay and contamination delay
- Optimize a simple combinational circuit for speed by implementing the same function with fewer or faster gates.
- Explain what glitches are and why they occur

7. VHDL for combinational logic

- Using a reference sheet, write the code for a complete VHDL entity
- Use VHDL to implement a boolean equation or truth table

8. Number systems

- Convert an integer from decimal to hexadecimal and back
- Convert a signed integer from decimal to 2's complement binary and back
- Explain why we prefer 2's complement to sign-magnitude

9. Testing and testbenches

- Use the VHDL process structure for combinational logic and print-style debugging.
- Use a sensitivity list to control when a process executes
- Write a test bench for combinational logic
- Explain the difference between synthesizeable and non-synthesizeable constructs

10. Adders and other combinational circuits

- Explain the basic principles of a carry-lookahead adder, and why it is faster than a ripple-carry adder
- Explain what an ALU is, and describe the purpose of the control signals and flags