ECE474 **Quiz 1** **Jan 18, 2008**

Name:_____

(last 4) ID:_____

Short answer questions, 3 points each.

(1) Why does simulation time for increase as definition of the design increases?

Increasing level of detail is expressed in the design. Moving from fairly abstract representation to detailed and concrete.

(2) At what point of simplicity does one stop when partitioning a design?

When you can wrap your brain around it. The individual partitions are all easily understood by the designer.

(3) What is the most prominent output of a logic synthesis tool?

A netlist of gates.

(4) What is the earliest point in the design process is a cycle-accurate simulation possible?

After the design has been fully specified and just prior to RTL coding.

(5) What is the earliest point in the design process at which an estimation of the minimum cycle time is possible?

Once synthesis is done and we have gates.

(6) What additional timing information is obtained in place and route?

Delays due to wiring.

(7) Successful use of the top down paradigm is primarily dependent upon clearly defining what part of the design?

Clear definition of the interfaces.

(8) Name the one nearly unavoidable downside of using top down design.

Duplication of effort.

(9) Good partitioning is vital to a clean small digital design. Name three aspects of a design that should inform partitioning.

Complexity, power supply voltage, clock domain, functionality

(10) What is the difference between a requirements document and a design specification?

Requirements doc: contradictory, confusing, possibly impossible to make
Design spec: exhaustively complete, workable design

True/false questions, 1 point each.

(11) You don't need to know what hardware structure the synthesis tool produces, that detail is excessive and is appropriately abstracted away.
False, you need to see if the tool made gates correctly.

(12) Writing synthesizable HDL code is very similar to writing an algorithm for most any other language.
False, you are describing a structure, not an algorithm.

(13) Synthesizable HDL code always reflects the structure of hardware.
False, inference of structure is the norm.

(14) Generally speaking, we look at datapath first, then control.
True.

Circuit question, 20 points.

(15) For the circuit given below, estimate the number of gates if:
2 input NOT/NAND/NOR/AND/OR/XOR/XNOR = 1 gate
1 D flip flop = 10 gates

XORs 16 gates * 1 gate/XOR = 16
dff 16 gates * 10gates/dff = 160
mux 16 gates * 3gates/mux = 48
not (1 per mux array) = 1
225 gates total