University of California at Berkeley
Department of Electrical Engineering and Computer Science

EECS150/EE141/EE241A, Fall 2014

Homework 1: Design flow, performance metrics and Verilog

Due: Friday Sep 12, 5pm

Please include your name, SID and specify either CS150, EE141 or EE241A at the top of your homework handin. Homeworks must be submitted electronically on bcourses (no paper submission allowed).

1. A wide range of computing systems are currently in production. Consider the following devices when answering the questions below: a laptop, a digital watch, a scientific calculator, a supercomputer, and a smartphone.

   (a) Sketch a curve showing computational performance of all these systems as a function of their cost. Put performance on the y-axis (arbitrary units), and cost on the x-axis (dollar estimate).

   (b) Similarly, show a curve that relates computational performance to system power consumption, with performance on the y-axis (arbitrary units), and power consumption on the x-axis (watt estimate). In the case of the smartphone, ignore the power consumption of the radio.

2. You are designing a CPU and are trying to evaluate two different implementations of the same design. The specifications for the two CPUs is given in the table below:

	Implementation A	Implementation B
Number of cores	1	2
Clock speed	2GHz	1GHz
Number of transistors	2 million	4 million
Supply voltage	1.2V	0.7V

   In order to decide which implementation to choose, you decide to benchmark a function, called `bmark` on these two processors. The function `bmark` takes $5000/N$ cycles to run, where N is the number of cores in the processor.

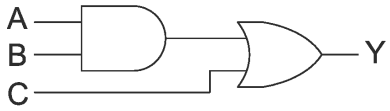
   (a) How long will it take (in seconds) to run `bmark` on each processor?

   (b) The power consumed by a single transistor can be modeled using the equation $P[\text{in watts}] = ACV^2f$, where $AC = 2 \times 10^{-14}$, V is voltage and f is clock frequency in Hz. What is the average power consumed by each processor implementation?

   (c) How much energy does each processor consume in calculating the function `bmark`? Hint: recall elementary physics, where Energy (in Joules) = Power (in Watts) x time (in seconds).

(d) In mobile phones, the primary concern is extending battery life. Therefore, which implementation would be better if you wanted to use the CPU in a mobile phone, and why?

3. (a) Draw a truth table for the logic circuit given below. It has 3 inputs and 1 output.
   (b) What happens if a register is placed between the output of the AND gate and the top input of the OR gate? You do not need to redraw the truth table, just explain how the addition of the register will affect how the circuit functions.



4. After a design is described using HDL, what are the necessary steps for it to become an actual circuit implementation on either:

   (a) an FPGA device? or
   (b) an ASIC?

   Explain what happens at each step, and what tool is used for that step. Note: you only need to describe the steps for either an FPGA or an ASIC, not both.

5. You are chief engineer at Pear Mobile Systems. You are currently developing a digital circuit for your latest product, the iMonocle. You are trying to decide whether you should build a custom ASIC for the digital circuit, or implement it on an FPGA.

   It costs $1 million to create the ASIC mask (once-off cost), and $6000 per wafer of 2000 ASICs thereafter. Each FPGA costs $100 (ignore volume discounts). You plan to sell x units of the iMonocle. For what values of x is it cheaper to implement a custom ASIC, and for what values of x is it cheaper to use FPGAs for your circuit?

6. Given the expression $Y = (A \oplus B) \cdot \overline{C}$, where $\oplus$ represents XOR, write a *structural* Verilog module that implements this expression. Your module should have inputs `A`, `B` and `C`, and output `Y`. Note that since you will be using structural Verilog, you will need to implement the necessary gates using expressions such as `and(out, in1, in2)`.

7. Repeat the question above, but this time using *behavioral* Verilog to implement the module.

8. **For EE241A only:** One-hot encoding is an alternative to binary encoding. For example, in 8-bit one-hot encoding, the number 0 is represented as 00000001, the number 1 is 00000010,

2 is 00000100, 3 is 00001000, ..., 7 is 10000000. It is called "one-hot" because only 1 bit is ever on at a time.

(a) Design and draw a circuit to convert 8-bit one-hot encoded numbers to 3-bit binary numbers.

(b) Write a Verilog module to implement this circuit. It should have an 8-bit input `oh_in` and a 3-bit output `bin_out`.