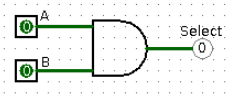
CS 441/641 Homework 0

Due **in class** 2012-09-06. Covers course prerequisites, background material, and survey.
Name: _____ UAF Email: _____@alaska.edu

For each of these circuits, I'd like you to tell me all possible inputs that will make the "select" line true.
You can get runnable .circ files for Logisim from http://www.cs.uaf.edu/2012/fall/cs441/hw0/

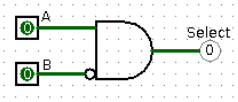
---

hw0_0: Basic logic circuit.



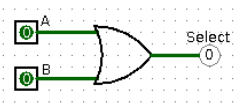
---

hw0_1:



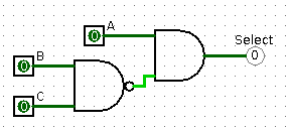
---

hw0_2:



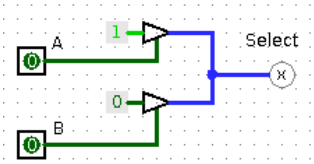
---

hw0_3: This is a three input decoder.



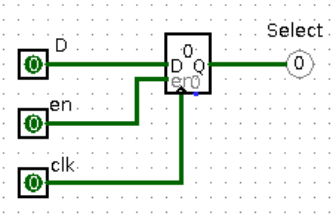
---

hw0_4: This is a tristate bus, with non-inverting tristate drivers.



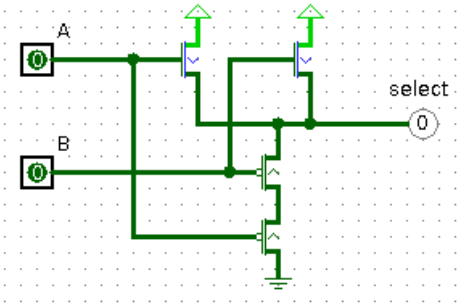
---

hw0_5: This is a "D" type flip-flop.

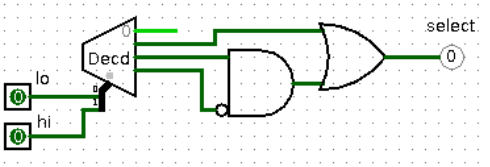


---

hw0_6: This is a logic circuit built from analog FET transistors.



hw0_7: This is a two-input decoder, driving some downstream logic.



hw0_8.) An RV bagel toaster is consuming 100 amps at 12 vdc. Is the toaster about to catch on fire?

hw0_9.) Write some assembly code on the left, to implement the C language function on the right:

Assembly code:	extern int bar(int y); /* function prototype */
	int foo(int x,int y) {
	  return 7+bar(y);
	}

(Pick any real CPU you like!)

Survey.) Which of these things would you like to hear more about? Please rate your interest from 0 (not at all interested) through 3 (worth about a week) through 10 (willing to devote your life to this subject).

__ mechanics of PCB fabrication (copper traces, multi-layer boards, soldering)

__ mechanics of semiconductor fabrication (photolithography, etching, masking)

__ high performance computing general techniques

  __ benchmarking __ load balancing __ tuning __ autotuners

__ GPU performance tuning generally, and graphics card massively parallel programming models:

  __ CUDA (popular, but NVIDIA specific) __ OpenCL (portable, but verbose) __ MS AMP

__ SIMD performance tuning generally, and the single instruction multiple data programming models:

  __ x86 four-float SSE __ x86 eight-float AVX __ ARM NEON VFP

__ multithreaded machine programming models

  __ OpenMP __ pthreads __ processes (fork)

__ multithreaded correctness: memory consistency models, locking, atomic operations

__ cloud computing, such as Amazon's Elastic Compute Cloud

__ processor virtualization support, such as VT-x CPU extensions

__ latest and greatest CPU designs (e.g., Ivy Bridge, AMD Opteron)

__ biological computing (computing using DNA nucleotides, protein soups, etc.)

__ quantum computing (computing with “qubits”, which can store the “superposition” of 0 and 1)

__ _____ (please suggest an interesting topic!)