CS2130 - Computer Systems and Organization 1.
Due monday, 5:30 pm. Via Gradescope.

Homework 2.

Important: Do not draw the circuits by hand; instead, you should use the CircuitVerse simulation tool. https://circuitverse.org/simulator **When you are done, export your file (Project → Export File). Name your file FirstName_LastName_HW2.cv** and upload it to Gradescope. This will export a single file containing all three problems. Note: Save each problem as a different circuit.

Collaboration Policy: **For this homework only!** You may collaborate with other students in this class. As an **exception** to the usual collaboration policy, you do not need to tell us about casual interactions of the "I got *X*, what did you get?" variety. But **do** cite any close collaboration or major corrections; for example, if the answer to the above hypothetical was "I think *X* is wrong, here's why," and then you change your answer, add a note like "mst3k suggested this answer" next to your answer. However, we expect that everyone will work on the assignment to better understand circuits, so **you may not directly copy another student's answer**.

Problem 1 *4-bit Shifter.*

Design a circuit that accepts two inputs: a 4-bit number and a 2-bit wide input shift amount. The circuit's purpose is to generate a 4-bit output that reflects the input number after a bitwise left shift. Importantly, this circuit should operate without relying on a clock signal. Use ground for inputs that are zeros. Please label all your inputs and outputs.

Problem 2 *4-bit Increment.*

In class, we considered a counter circuit which counted from `0b000` to `0b111` and then back to `0b000`. Design a counter circuit that "stops" at `x = 0b1111`. That is, if `x` is not all 1s, then increment by 1. If `x` is all 1s, then increment by 0, i.e., `z = 0b1111`. Feel free to use the Full Adder component under Misc. Think about how you can turn a Full Adder into a half adder by setting inputs. Also, feel free to wire up the reset line on your flip flops to reset them for testing. Please label all your inputs and outputs.

Problem 3 *Register File.*

Create a register file comprising four register banks, each capable of storing 4 bits. The register file should have four input signals: a clock signal, a write enable line, a 2-bit wide register number selector, and a data line. Furthermore, the register file should feature a single 4-bit wide output, representing the value currently stored in the selected register. Please label all your inputs and outputs.