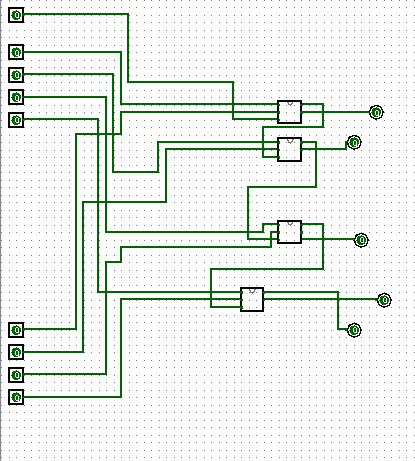
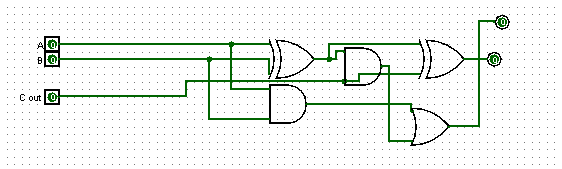
**Binary Calculator**

**Description of Function**

The circuit that is being built is a binary calculator. This binary calculator is made up of two main systems, the adder and the multiplier. These systems are toggled by the two switches labeled as Add and Multiply. To calculate a number, you must first input the first number to compute in binary into the switches labeled I0-I3 where I0 is the first 20 and I3 being the 2­3 digit. Once the desired number is inputted it must be inserted into the memory, so the user activates the shift switch which will store the number. The shift switch is then left on and the desired operation is selected either Add or Multiply. The second number may now be inputted, and the desired result should emit at the outputs labeled O0 to O4. It should be noted that when multiplying only the inputs I0 and I1 should be used as it can only compute 5 bit numbers and having 3 bit inputs for multiplication may cause there to be 6 bit outputs.

**Breakdown of how the circuit works**

The circuit appears very basic but requires some advanced mechanics. The circuit takes the input and when the shift button is pressed it activates a part of the d flip flop which prevents the flip flop data from being altered. Then the circuit checks which operation is being used, this requires a significant 12 and gates. Once through the memory process and the operation distinguishing the circuit then proceed to the actual calculations. The multiplier is a very simple 2 by 2 bit binary multiplier. The adder however is more complicated. The 4 bit adder is 4 full adders connected in where the carry out connects to the carry in on the other full adder. 

Full Adder 4 bit adder

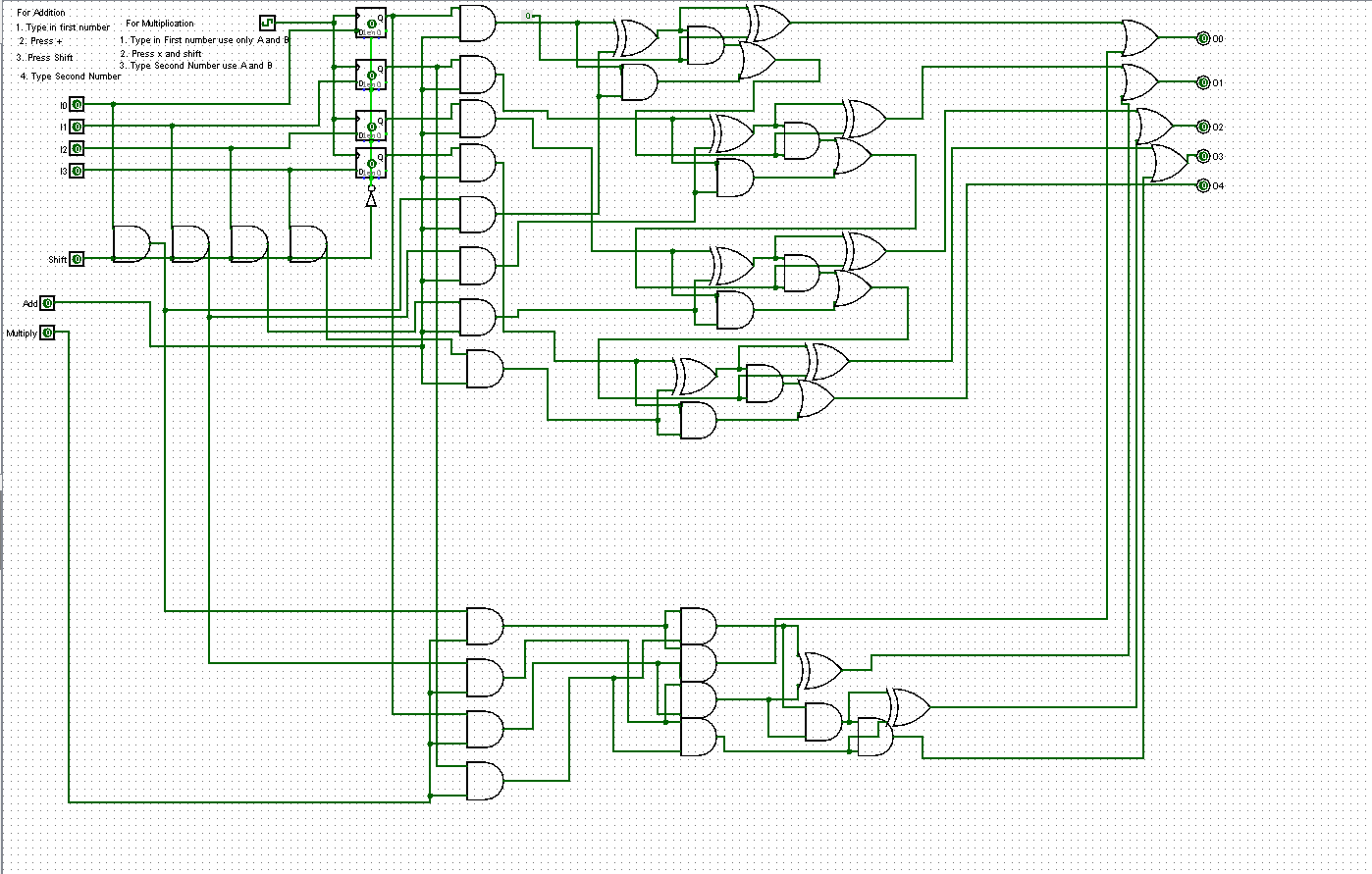
Finally the circuit uses an or gate so that the circuit can function with two inputs connecting to one output. This OR gate can be replaced by a diode however Logisim has no alternative.

**The D-Flip Flop**

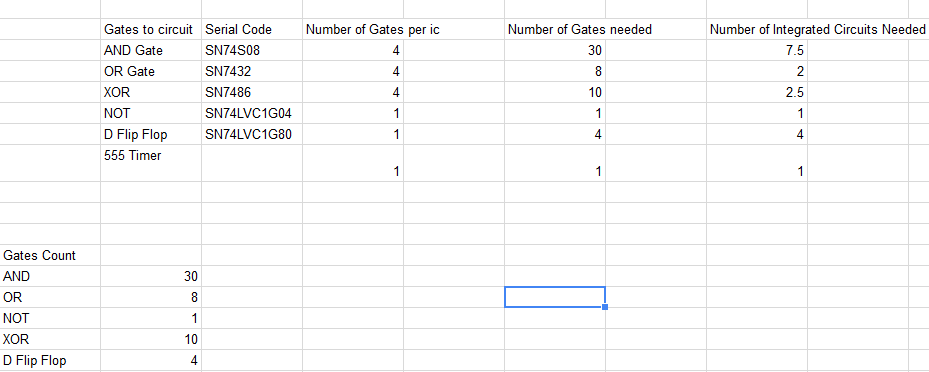
The d-Flip flop was not taught in class but is critical to the function of this circuit. Most D-Flip flops have many inputs but in this case we only require 3. We need the clock input, the data input and the toggle. The way the d flip flop works is every time the clock oscillates then the value of the data input is stored and that storage is outputted. There is another pin on the south side of the flip flop and when this is true than values may be stored however when it is false no data is stored so we use this so our data cannot be overwritten.

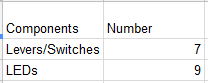
**NOTE about the Truth Table**

There is an issue with the truth table referenced below. As it can be seen much of the data is 0. This is because these truth table cannot fully represent the data in this circuit. This circuit uses D-Flip flops so that means it can store different values at different times. If one were to do a truth table it would have an infinite amount of rows as any possible combinations can be stored in these flip flops and these flip flops will change in time. This truth table is technically correct however functionally useless.

**Full Logisim Circuit without sub circuits**

**Parts List**





**Truth Table**

