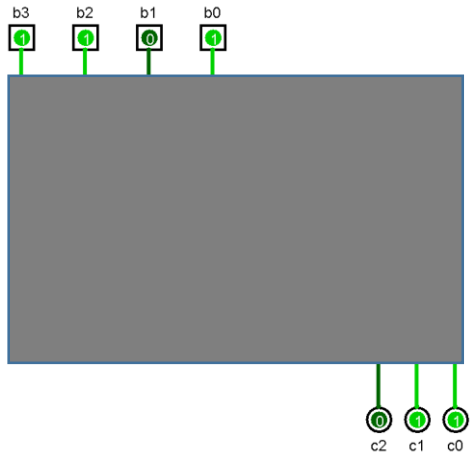
Simple Combinational Circuits

For this project, you will be using Logisim 2.7.1 to create some simple combinational circuits. You should familiarize yourself with Logisim by working your way through the built-in tutorial.

You might want to review the CS 2505 notes on data representation before starting to design your solution.

Q1: Count-ones Circuit for Nybbles [48 points]

Create a simple circuit that computes the number of 1-bits in a given nybble. The circuit will have 4 1-bit input pins and 3 1-bit output pins (in order to take advantage of Logisim's ability to generate a truth table). The interface must be laid out as shown below:



Note: you may not use the Bit Adder component or any addition circuit, that is provided in the Logisim Arithmetic library.

You may use any of the other circuit components provided by Logisim, including the other arithmetic components and the plexor components. You should probably consider creating subcircuits rather than a single, monolithic solution.

You must also create a plain text file showing your analysis that led to your circuit design; most likely this will consist of a truth table, and the associated Boolean expressions. You are not required to simplify the Boolean expressions, but doing so will make implementing the circuit easier.

The correctness of your solution will be determined by using Logisim to generate a truth table (16 rows), and comparing that to the truth table produced from the reference solution. If you don't lay out the input and output pins as shown, then Logisim may generate the truth table columns in a different order, and you will lose points. In Logisim, go to Project/Analyze Circuit to generate the truth table. The column headings should be:

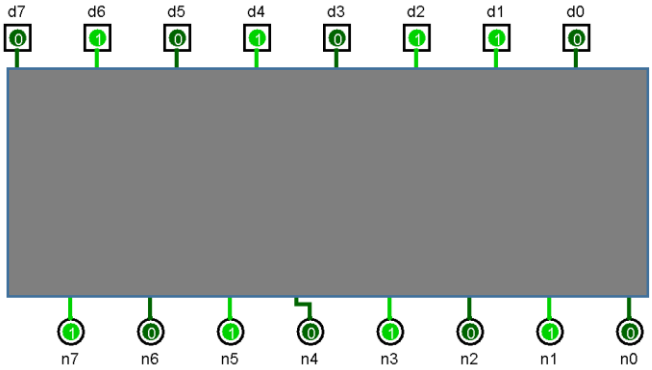
```
b3  b2  b1  b0  |  c2  c1  c0
```

If not, your circuit interface is incorrect.

Save your implementation of the circuit in a Logisim file named `bitadder4.circ`, and name your text file `BitAdder4.txt`. The use of other names for the files will irritate the person who grades your solution, and you may be charged for that irritation.

Q2: Arithmetic Negation Circuit for int8_t [20 points]

Create a simple circuit that computes the arithmetic negation of signed 8-bit integers, represented in 2's complement form. The interface must be laid out as shown below:



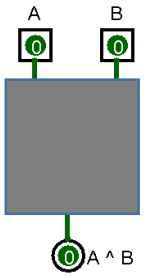
You may not use Logisim adder, subtractor, or negator components; all other Logisim components are allowed. In this case, you should consider how to accomplish the computation from a high level; creating a truth table (256 rows!) and Boolean expressions would be unproductive.

The correctness of your solution will be determined by using Logisim to generate a truth table (256 rows), and comparing that to the truth table produced from the reference solution. If you don't lay out the input and output pins as shown, then Logisim may generate the truth table rows in a different order, and you will lose points.

Save your implementation of the circuit in a Logisim file named `negate8.circ`. The use of other names for the files will irritate the person who grades your solution, and you may be charged for that irritation.

Q3: XOR Gate via Transistors [20 points]

Create an XOR gate by using transistors, pull resistors, and any other elements in Logisim's Wiring library. You may not use Logisim components from the other libraries. The interface must be laid out as shown below:



The correctness of your circuit will be determined by using Logisim to generate a truth table (4 rows), and comparing that to the truth table for the XOR gate. If you don't lay out the input and output pins as shown, then Logisim may generate the truth table rows in a different order, and you will lose points.

Save your implementation of the circuit in a Logisim file named `XOR.circ`. The use of another name for the file will irritate the person who grades your solution, and you may be charged for that irritation.

What to submit

You will submit a single, uncompressed, flat tar file containing your text file and your three Logisim files. If you worked with a partner, only one of you should make a submission, and be sure the text file contains a copy of the Partners Template (see the Assignments page), including the PID and name of each partner. Otherwise, only the partner who made the submission will receive credit.

The *Student Guide* and other pertinent information, such as the link to the proper submit page, can be found at:

http://www.cs.vt.edu/curator/