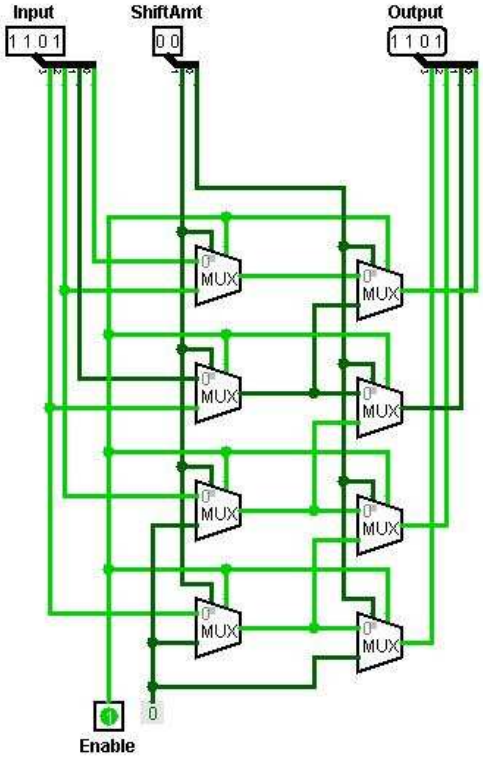
You may work in pairs for this assignment. If you choose to work with a partner, make sure only one of you makes a submission a solution and that the file lists names and PIDs for both of you as described in the assignment below. Prepare your answer to each of the following questions in a single Logisim `circ` file. Create a tar file that contains your two Logisim files, and the text files containing your explanations, and submit the tar file to the Curator system by the posted deadline for this assignment. No late submissions will be accepted.

You will submit your answers to the Curator System (www.cs.vt.edu/curator) under the heading `HW17`.

1. [50 points] Begin by reviewing the discussion of the 4-bit barrel logical right shifter in the course notes. For this assignment, you will make some modifications to the implementation shown in the course notes, so it's essential that you understand exactly why the one shown in the course notes works. For reference, here is a slightly edited picture of that circuit:



Download and install Logisim; access the Logisim help menu and read through the Tutorial section; it's quite good. Create a new Logisim file named `BarrelShifters.circ`. Use the Logisim Text Tool (on the Base Library tree) to create a text box containing your name and VT email PID (do this for both partners if you work in a pair).

Implement the 4-bit barrel logical right shifter above, and test it. This will be the "main" circuit in the Logisim file; you should rename it "4-bit Barrel CRS" (click on "main" and change the circuit name in the dialog).

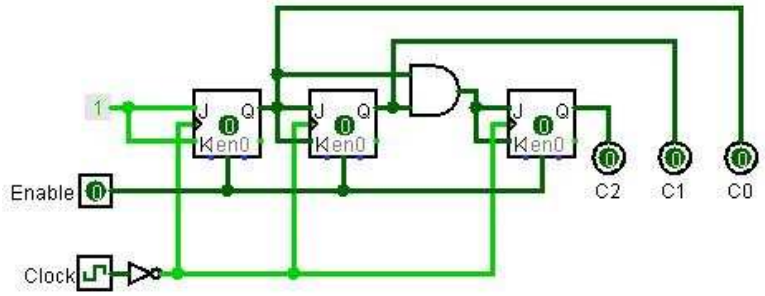
You will now modify the design of circuit "4-bit Barrel CRS " so that it performs a circular right shift of its input. In a circular shift, the bits that shift out at one end are shifted in at the other end; for example:

`1101 → 1110 → 0111 → 1011 → 1101 . . .`

Don't change the "interface" for the circuit. There should still be a 4-bit Input, a 4-bit Output, a 2-bit ShiftAmt and an Enable input.

Write a text file with a brief explanation of how you analyzed this problem and arrived at your solution. This should be clear and concise; a single sentence would not be sufficient; a full page would probably be excessive.

2. [50 points] Begin by reviewing the discussion of the mod-8 up-counter, implemented using JK flip-flops, in the course notes. For this assignment, you will make some modifications to the implementation shown in the course notes, so it's essential that you understand exactly why the one shown in the course notes works. For reference, here it is:



Create a new Logisim file named `Mod16Downcounter.circ`. Use the Logisim Text Tool (on the Base Library tree) to create a text box containing your name and VT email PID (do this for both partners if you work in a pair).

Follow the example in the course notes to analyze the problem of designing a mod-16 counter that counts down, rather than up. That is, the counter value changes according to the pattern:

`1111 → 1110 → 1101 → 1100 → . . . → 0000 → 1111 . . .`

Your analysis should include the table for the next state function, and show the derivation of the necessary Boolean functions. Be sure to simplify those functions, otherwise you'll be creating a lot of unnecessary gates in your implementation. Write your analysis in a plain text file (use ~, * and + for the operators).

Then use Logisim to implement your design; you should use the interface in the circuit above as a guide.