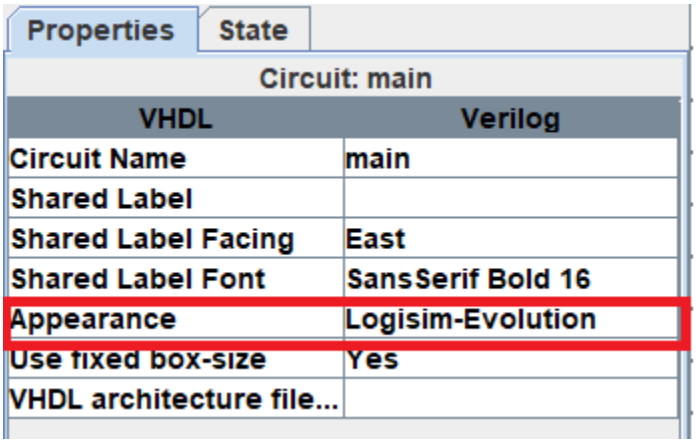
Logisim

1. Use the version from the class Google Drive of Logisim Evolution. Other versions may not work correctly.
2. Do not rename the files you receive. If you do so you will automatically fail the tester when you submit.
3. Put your solution for each problem into implementation subcircuit
4. Do not rename the implementation subcircuit anything else. If you do so you will automatically fail the tester when you submit.
5. Do not change the appearance of the implementation subcircuit from what it is set as. Doing so will cause you to automatically fail the tester when you submit.
    a. That is this field right here



6. Do not move the pins inside of the implementation subcircuit as that affects the appearance of the circuit on the outside as you saw in discussion. Doing so will cause you to automatically fail the tester when you submit.
    a. If you want to “move the pins” instead connect tunnels to the pins and move the tunnels around.
7. Do not name any of the subcirucits in your solution main. Doing so will cause you to automatically fail the tester when you submit.
8. You **can** create as many other subcircuits as you want in your solution. Just make sure your solution ends up in the implementation subcircuit

Restrictions

For all problems in this homework, you may only use

- All of the components under the wiring folder
- AND, OR, NOT, and XOR gates

Problem 1: fun.circ (25 points)

Create the **simplest** circuit in Logisim that implements the following function: $(x_0 * x_1 * \overline{x_2}) + (x_2 * x_3) + (x_0 * x_1 * \overline{x_3})$. Simplest means it uses the fewest gates.

Hint

Using the given formula directly may not implement the simplest circuit.

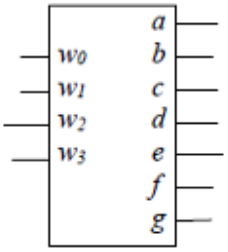
Problem 2: grey.circ (25 points)

Grey scale is an alternative method for representing binary numbers. In Grey scale, only a single bit changes between adjacent numbers. The following truth table shows the mapping from binary to grey scale. Create the **simplest** circuit in Logisim that implements the following truth table.

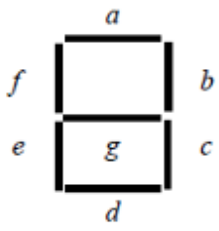
X_2	X_1	X_0	Grey Code
0	0	0	000
0	0	1	001
0	1	0	011
0	1	1	010
1	0	0	110
1	0	1	111
1	1	0	101
1	1	1	100

Problem 3: bcd.circ (50 points)

Given the following BCD-to-7-segment display, derive the minimal SOP function for the outputs, a, b, c, d, e, f, g of the 7-segment display, and then implement that function in Logisim.



Code converter



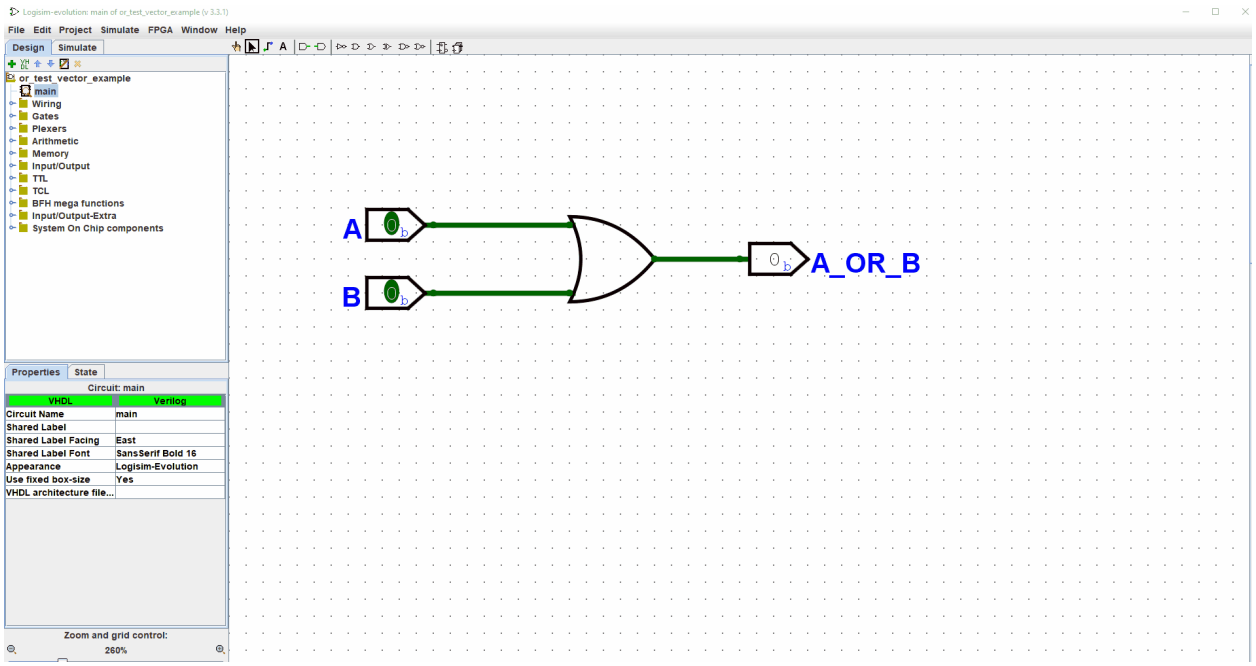
7-segment display

w_3	w_2	w_1	w_0	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1

Testing

Testing for this assignment should be fairly straight forward. All you have to do is load the included test vector for each problem into the implementation subcircuit. It will then tell you for each input whether the corresponding output is correct. If the test vector is says everything is good you should pass all of the test cases when you submit as long as you didn't break any of the rules at the beginning of this document.

To run a test vector, select Simulate -> Test Vector. Then in the pop-up window select Load Vector.



Submitting

Submit to

Logisim Homework 1 on GradeScope.

What to Submit

Submit a zip file that contains the following .circ files

1. fun.circ
2. grey.circ
3. bcd.circ

Make sure that you submit a zip that contains the files and **NOT** the **folder containing the files.** Check out the animation below for what to submit.