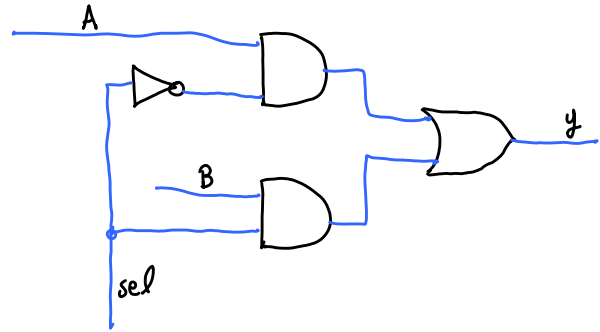


Lec-2-HW-2

(1.) At left is a drawing of a logic circuit. Recreate it in Electric yourself. Save it in your own "logic.jelib" library.

NB--Repositioning the pointer and right clicking will extend a wire from the last selected pin. It is easier to place straight segments and choose where to make turns than to let Electric decide for you. I usually first make connections to my objects (gates, for instance), extending a segment from each input or output. Then I connect these segments to each other as needed, or connect by giving wires the same name.



NB--By first selecting a connection point on a primitive, then using Electric's,

Edit.TechnologySpecific.TogglePortNegation you can add a negating bubble to the outputs of AND, OR, and BUFF primitive gates (it matters which connection point is selected when you do this). (You can also add bubbles to the inputs, but don't: it doesn't work.)

(2.) Write verilog testbench code to exercise the circuit and display its function. That is, in an initial block, set values for the inputs, wait a tic, and then display the inputs and output. Follow that with another delay and then change the input values, and so on. Continue until you have exhausted all possible input combinations. Remember to \$finish to stop the simulation. For each \$display, also show the simulation time.

NB--Recall from tutorial.jelib that you can drive any input wire or bus in the testbench by declaring an appropriate type "reg" to drive it. For instance, declaring "reg Asrc;" can be used to drive some wire "A":

```
reg Asrc;
assign A = Asrc;
reg [1:0] CbusSRC;
assign Cbus = CbusSRC;

initial begin
    Asrc = 1;
    CbusSRC = 2'b01; //--- Cbus[1] <== 0 , Cbus[0] <== 1
    ...
end
```

(3.) After completely testing it, speculate on what the function of this circuit is; that is, what is it good for?

What to turn in:

Add your logic.jelib library to your branch under lib/. Print on paper your simulation's output and the driver code you wrote. Turn that in with a cover sheet