EE 431 Lab 4 Report

AND Gate LVS and Post-Layout Simulation

Author: Ryan Cramer

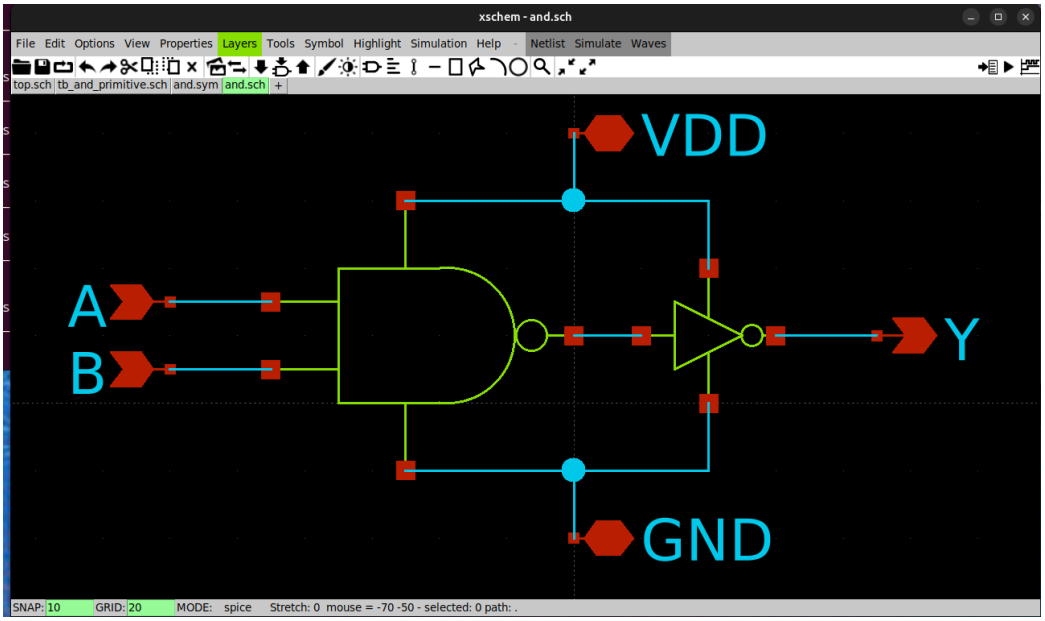
I. Introduction

In Lab 4, we do a hierarchical layout of an AND gate using our previous layouts of the NOT and NAND gates. By adding an inversion at the output of the NAND, we are able to achieve AND functionality with only two gates (6T). In this lab, we take it a step beyond LVS by simulating against the layout spice file, ensuring that our design works electrically.

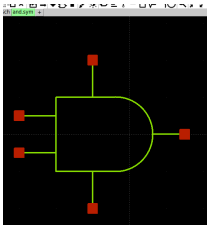
II. Methodology

To build the AND, we construct the gate using a two input NAND consisting of two parallel pMOS and two series nMOS, followed by an inverter which consists of one pMOS and one nMOS. By using our previous symbols, we avoid having to specify transistor sizes again. To build the NAND, we simply used two polysilicon gates, with ndiff and psd covered in nwell as our n and p transistors, respectively. We saw that the pMOS' shared a common output, Y, so we simply sandwich the Y node in between the two Polysilicon gates. The Y output feeds to the drain of the nMOS, which has to pass through two polysilicon gates to get to ground (series configuration). By asking Magic to load our cells, we are able to get the design, and then apply new poly and li to wire everything up, finishing by applying new labels and poly contacts.

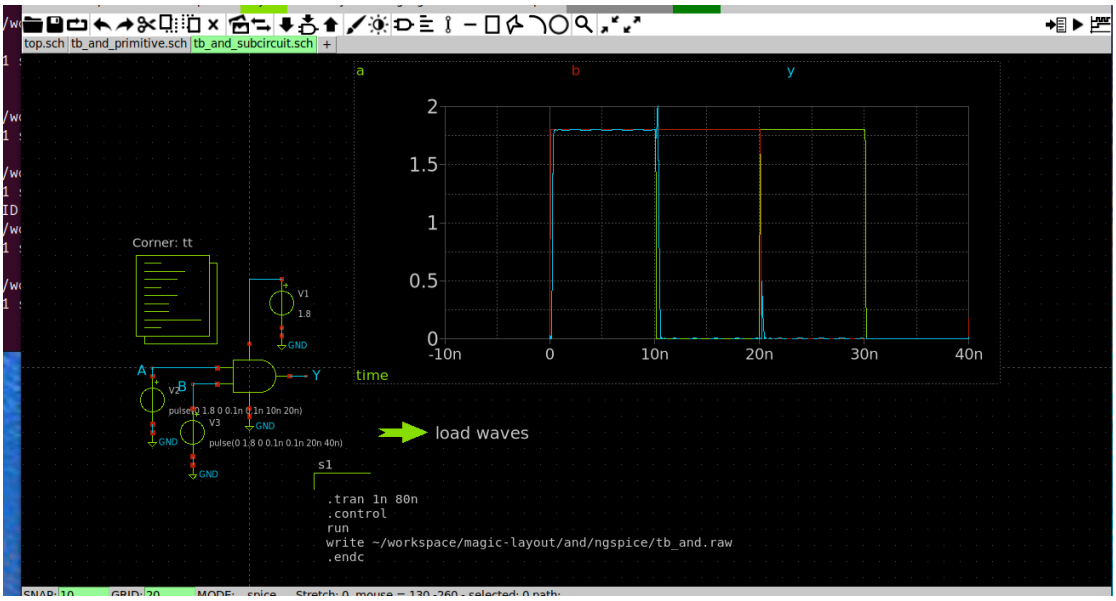
AND Schematic

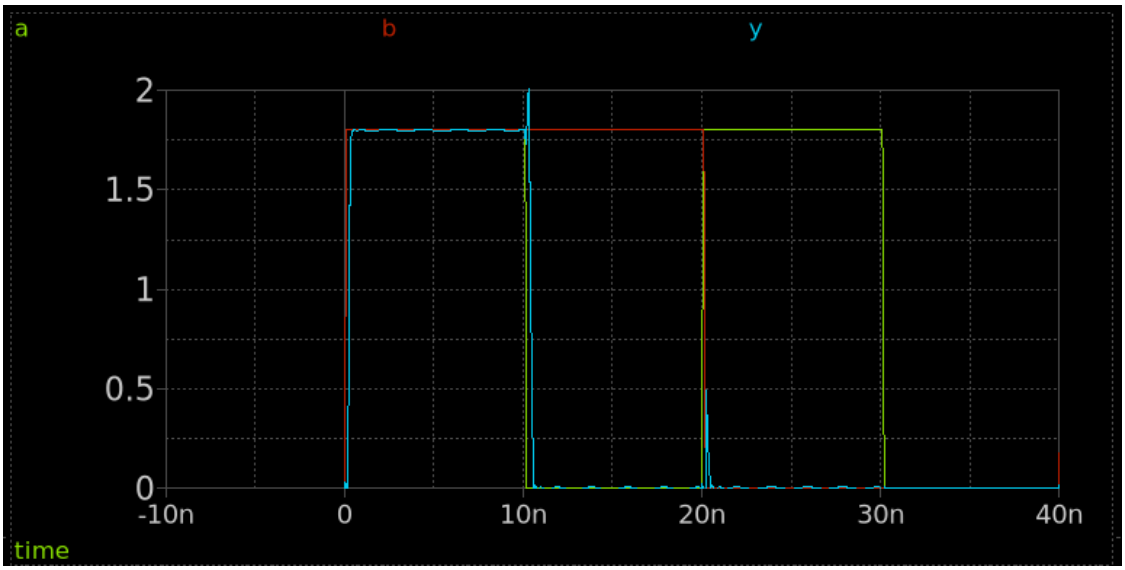


AND Symbol

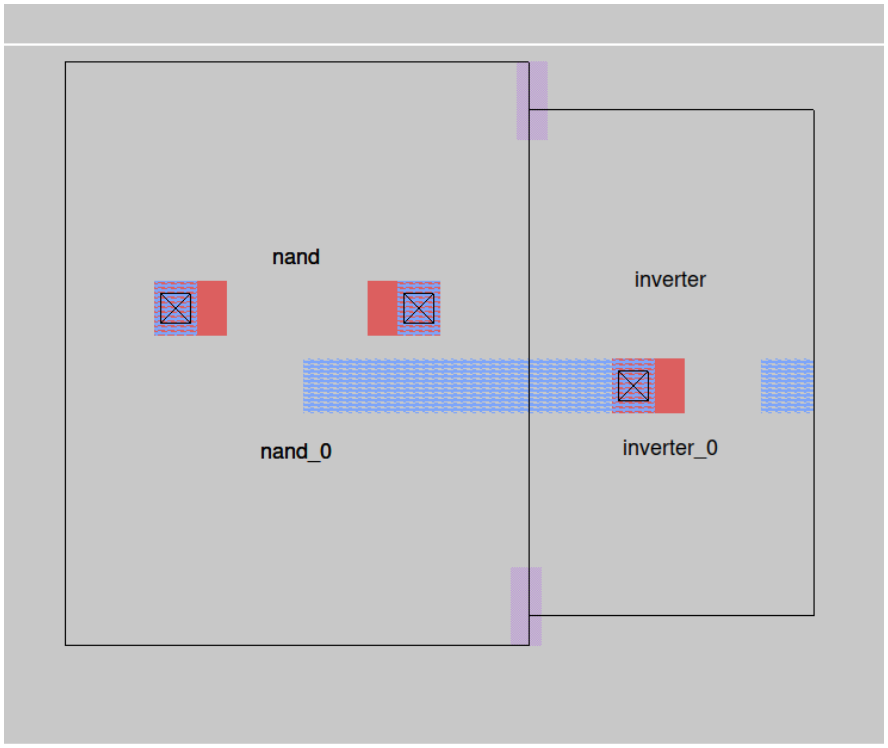


Testbench Schematic (subcircuit defined)

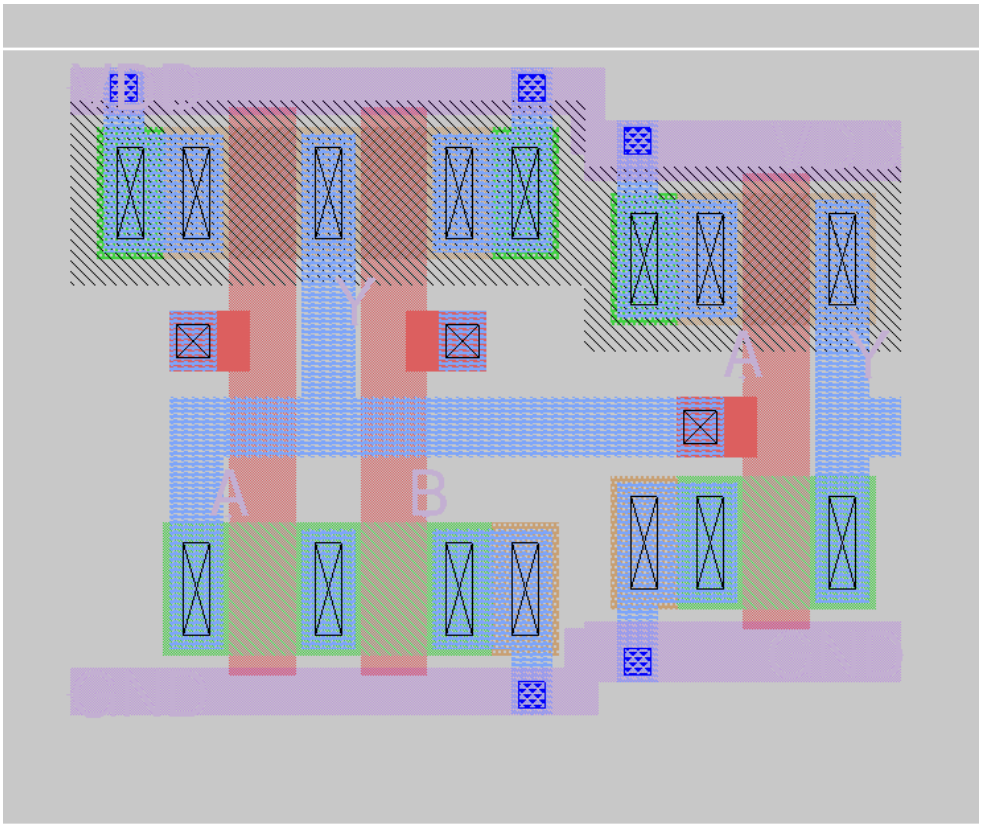




AND Layout (Subcells Hidden)

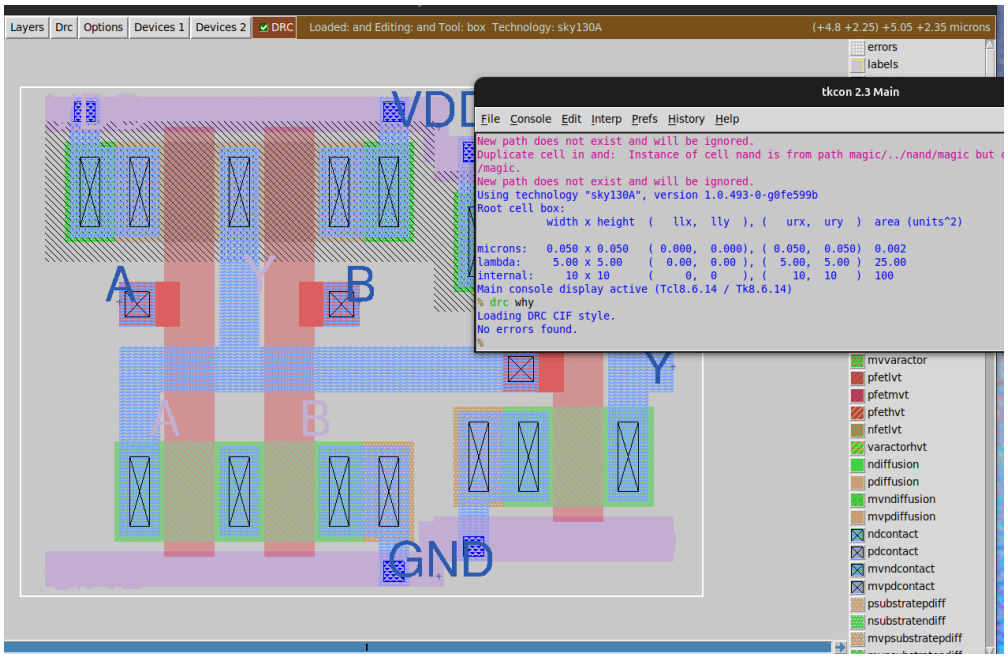


AND Layout (Full)

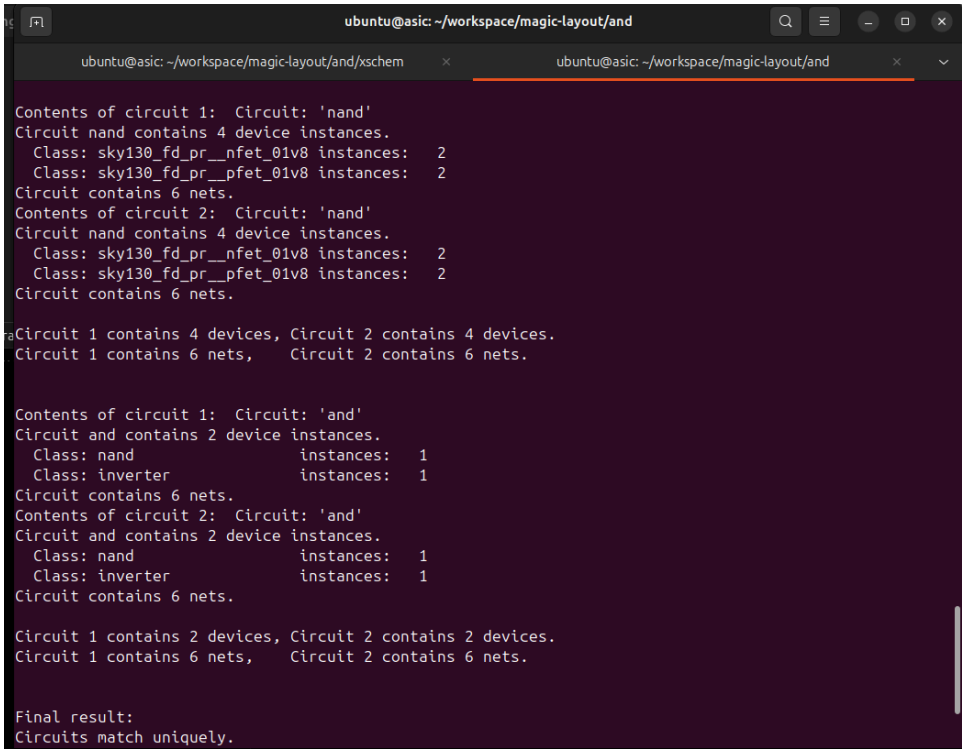


III. Results

Clean DRC Results

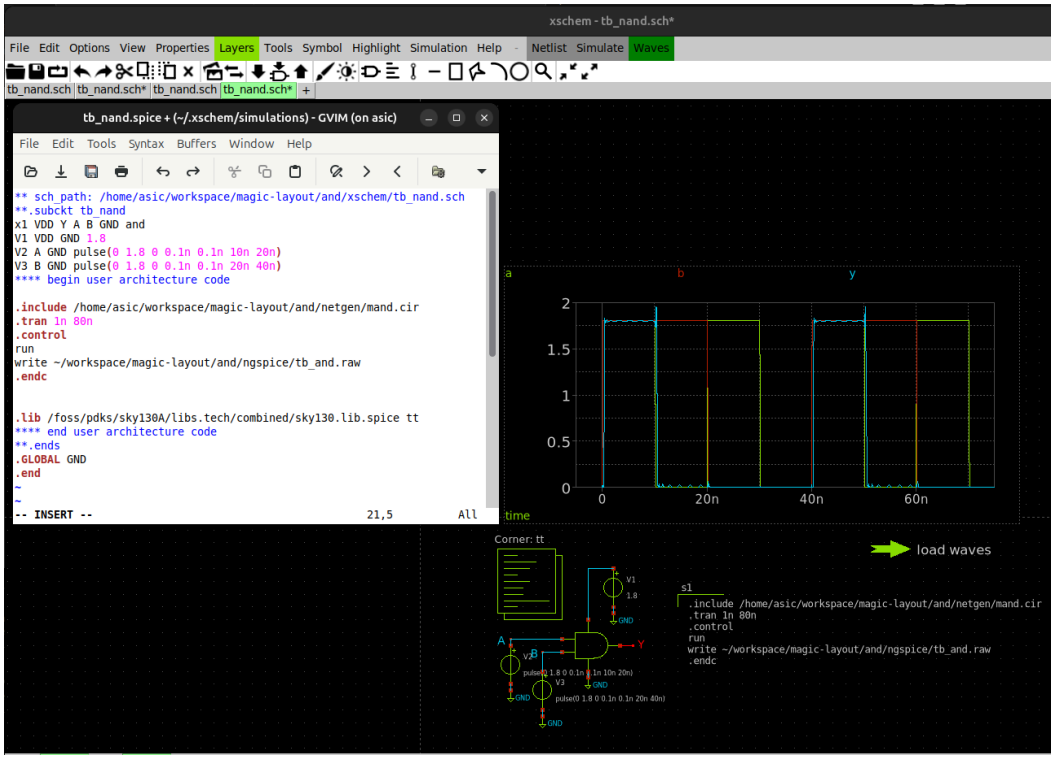


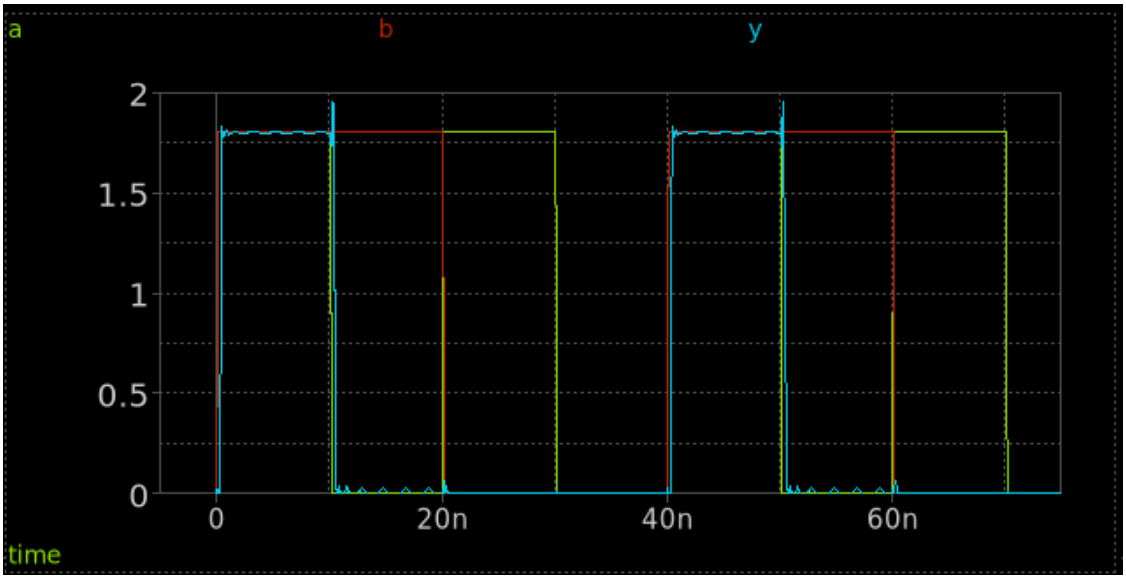
Clean LVS Results



(Circuits match uniquely for INVERTER, NAND, and AND)

Post Layout Testbench Schematic and Simulation (primitive defined)





So from our simulation results, we can see that the post-simulation layout spice actually simulated cleaner than the regular testbench. On the original simulation, we see a voltage spike in the output at the B falling, A rising transition. However, in our post-simulation results, we see more ripples at ‘1’ and ‘0’, but smaller voltage spikes at the transitions.

IV. Discussion

In this lab, I felt like I really understood how to use xschem, netgen, and Magic afterwards. I did not understand the difference between a hierarchical and flat spice netlist, and now I do. I was having issues running LVS until I realized that my spice netlist from xschem had the subcircuit definition commented out. After fixing it, it worked. Another thing I learned was that if you don't extract using the right commands, the ext2spice will put extra nodes in your layout netlist because of parasitic capacitances. To avoid this, I had to use ext2spice lvs to be able to avoid seeing those extra nodes. I also didn't realize that you couldn't just connect two ports in xschem, you actually need to define a wire or it will appear like your devices are not connected. Now I understand the tooling much better, and I feel like I could easily do a hierarchical layout and do post-layout simulation with it.