

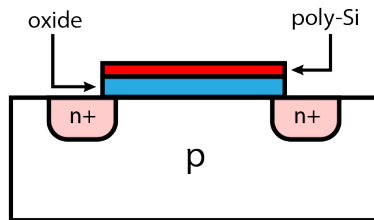
EECS 151/251A Homework 5

Due Monday, Mar 1th, 2021

Please include a short (1-2 sentence) explanation with your responses to each question unless otherwise directed.

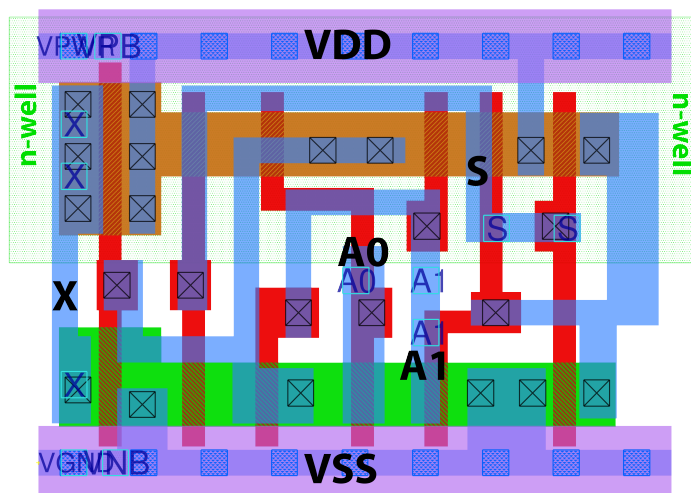
Problem 1: Basic IC Processing

In modern IC fabrication, metal layers are commonly added using a process called *Chemical Vapor Deposition (CVD)* (Wikipedia Article). Starting from the finished transistor below, describe the fabrication steps to manufacture the first metal layer, contacting the terminals of the transistor. The blue layer represents the oxide.



Problem 2: Standard Cell Layout Reverse Engineering

Take a look at the standard cell layout below from the SkyWater 130 nm open-source PDK. The supply rails, pins, and n-well have been labeled. What function does it implement? Provide a schematic at the transistor level.



Problem 3: MOS Characteristics

Using the same 16nm predictive LTSpice model for transistors as Homework 1, set up a circuit to measure I_{DS} vs. V_{GS} with $V_{DS} = 0.9\text{ V}$ for both NMOS and PMOS. Start with both devices at $0.1\text{ }\mu\text{m}$ wide. Repeat for $W = 0.05\text{ }\mu\text{m}$ and $W = 0.3\text{ }\mu\text{m}$. Plot the three current vs. voltage curves on one waveform for each MOS and submit it along with a screenshot of your schematic.

251A only — Optional Challenge Question for 151

In MOSFETs, there is a region of operation known as *subthreshold*. From your results in Problem 3, choose new voltage sweep endpoints to demonstrate this region of operation and plot the drain currents for the same devices as Problem 3. Does the current in the MOSFET still follow the square law relationship with the gate voltage in this region? Turn in screenshots of your schematic setup and waveforms.

Problem 4: CMOS

Build a complex CMOS gate that performs the following function:

$$y = (a'b' + c')(d + e)'$$

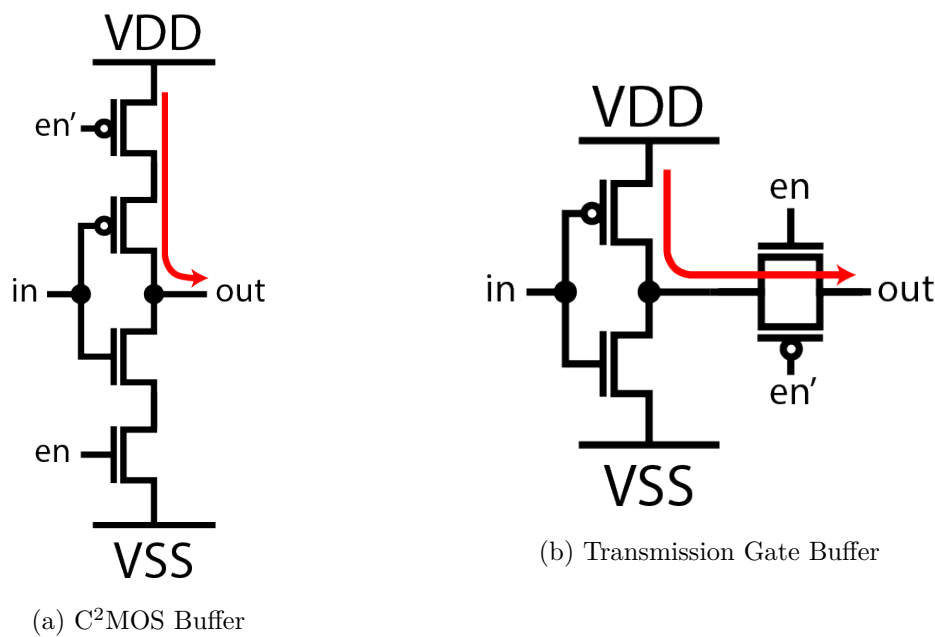
Turn in a transistor-level schematic of your CMOS gate. You may **not** use complemented inputs.

Problem 5: Transmission Gate Logic

A *rotator* is a circuit that takes a set of input bits and rotates them in either direction by some number of bits, configurable with another input. For example, a 3-bit rotator could rotate the bits either right or left by 0, 1, or 2 bits (3 bit rotation would be the same as a 0 bit rotation). Design a 4-bit rotator using transmission gate logic, using only 1 transmission gate per input/output path. The rotator will always rotate to the **right** (for example, 0110 will become 0011 with a 1-bit rotation). (*Hint: Note that the number of possible rotations is equal to the number of input bits. This lends itself well to an $N \times N$ grid of circuit elements to implement the function.*)

Problem 6: Tri-State Buffer

In the lecture, you learned about tri-state buffers and how they are made with either a transmission gate or C^2 MOS (stacking CMOS). Both of these approaches use two devices in series (transmission gate uses the driving static gate's devices as well as the transmission gate device itself).



Current paths in both tri-state buffer topologies. Note the two series devices.

There is another way to implement a tri-state buffer, with only one device along the path from either supply rail to the output. Design such a buffer and provide a circuit schematic. Explain its operation. In what situation would such a circuit be preferred over the classic way of making tri-state buffers?

Problem 7: Latch

A latch is a circuit that holds its output steady when an input control signal is either high or low. When the control signal allows the latch to update to its data input, we call the latch *transparent* to the input. A positive latch will be transparent when the control signal is high, and a negative latch will be transparent when the control signal is low. For a positive latch, the latch will remain transparent as long as the control signal is high, and vice-versa for the negative latch. Once the control signal changes, the latch will then ignore its input and retain whatever its last output was until the control signal requests the latch become transparent again. One way to implement such a latch is with multiplexers. Design a positive latch based around a 2-MUX and provide the circuit schematic (you may abstract the mux itself as just a block, but the rest of the circuit should be transistor-level). How might such a circuit be used to design a positive edge-triggered flip-flop?