CS 2110 Lab 4
Transistors: The building blocks of gates

Your TAs! :)

Fall 2018

1 Recitation Outline

1. DeMorgan's Law
2. States of a wire in CircuitSim
   (a) Low (dark green) (0) (0 volts)
   (b) High Impedance (blue) (undefined) (undefined voltage)
   (c) FIRE, exclamation mark, FIRE, exclamation mark, HELP ME, exclamation mark (red) (conflicting voltages)
3. Transistors
   (a) Essentially a very tiny switch
   (b) Source and drain
   (c) P type
      i. Normally closed (current passing through)
      ii. Connects to power
   (d) N type
      i. Normally open (**no** current passing through)
      ii. Connects to ground
4. Building Boolean Circuits With Transistors
   (a) Circuits model boolean expressions
   (b) Fully simplify a boolean expression
   (c) 1 (voltage) is True, 0 (ground) is False
   (d) Transistors in series: AND
   (e) Transistors in Parallel: OR
   (f) Pull-up network using P types
   (g) Pull-down network using N types (should be the negation of the pull-up network)

2 De-Morgan's Law

Laws for Boolean Algebra

- (not A and not B) = not (A or B)

  $\overline{A}$ && $\overline{B} = \overline{A||B}$

- (not A or not B) = not (A and B)

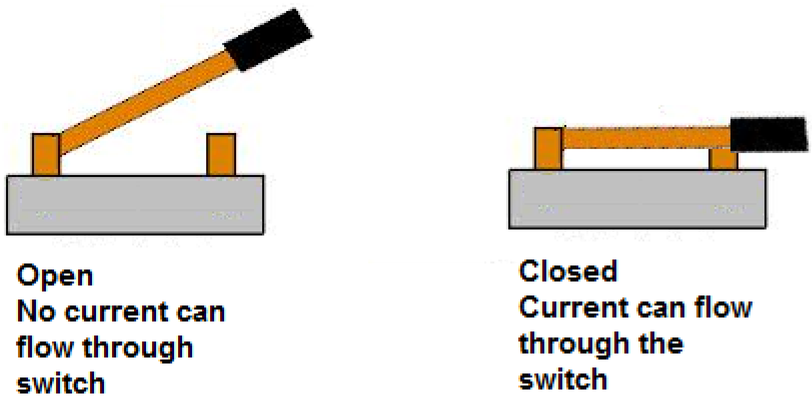
  $\overline{A}$ || $\overline{B} = \overline{A\&\&B}$

3 Which switch is which?

Draw a squiggly line on the board and claim it's a wire connected to nothing. It's just sitting there. **Question for the class**: what value is on this wire?
If you said 0, you're wrong. In this class, we will consider wires to have one of three values:

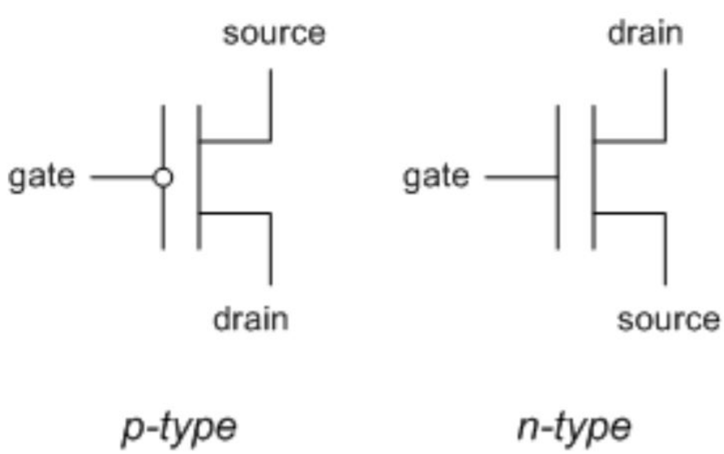
- 0 (current flows from ground). Dark green in CircuitSim
- 1 (current flows from power). Light green in CircuitSim
- high impedance / floating (no current flowing). Blue in CircuitSim

A switch either allows current to flow (from ground or power, aka 0 or 1) or prevents current from flowing (floating):



Note that when it's closed, that just means current flows. The current flowing through the switch could have a 0 or 1 value.
Now... what if we had a switch you could open or close based on the value a wire, made out of semiconductor so it's super compact? Might be useful, asking for a friend

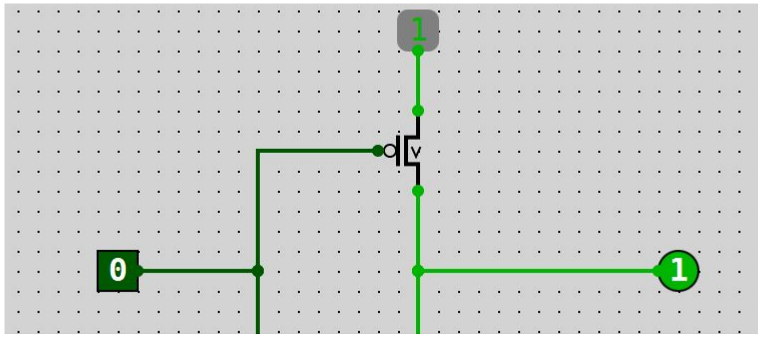
4 Transistors: The building blocks of gates



Each transistor has a source, drain, and gate. The input of the gate determines whether the switch will be closed, allowing current to flow from the source to the drain.

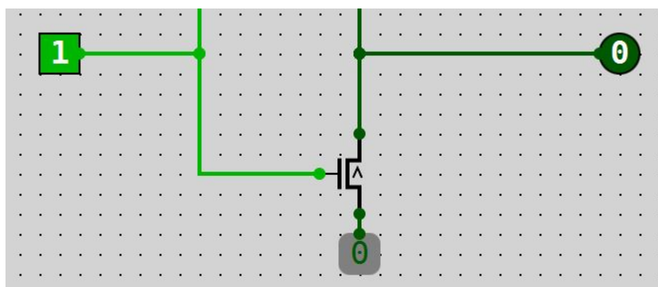
4.1 P-type transistors

- P-type transistors connect to Power(Voltage/logical 1).
- Whenever the gate value is '0', the p-type switch closes, allowing power to send a '1' to the output



4.2 N-type transistors

- N-type transistors connect to Ground(No Voltage/logical 0).
- Whenever the gate value is '1', the n-type switch closes, allowing ground to send a '0' to the output



5 Using transistors to build circuits

5.1 Rules for converting from transistors to circuits

1. Begin with a fully simplified boolean expression, let's refer to this expression as F. (Ex. F = A && B)
2. Expression F tells you when you want the output to be true, you want to use F to create the 'pull-up' network that is connected to the voltage.
   - Use p-type switches
   - Connect the switches in series for an 'and' network
   - Connect the switches in parallel for an 'or' network
   - Since p-type switches pull from voltage only when the input is '0', they work on the negated values of inputs.
3. F also tells you when you want the output to be false, you want to use expression F to create the 'pull-down' network that is connected to the ground
   - Use n-type switches
   - The 'pull-down' expression is the negated version of the 'pull-up' expression

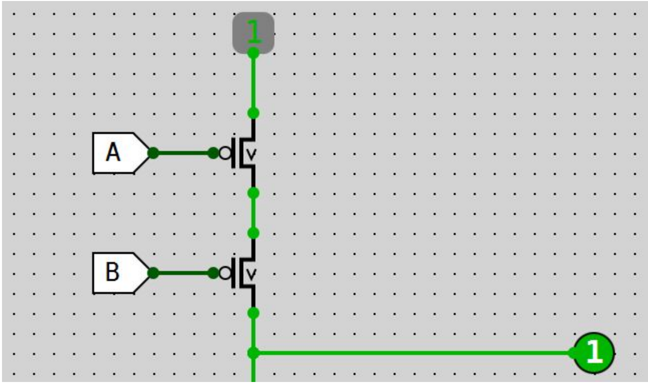
The easiest way to convert from transistors to circuits is by using a simplified expression
Ex. NOR

1. F = $\overline{A \mid\mid B}$

2. Using DeMorgan's Theorem:

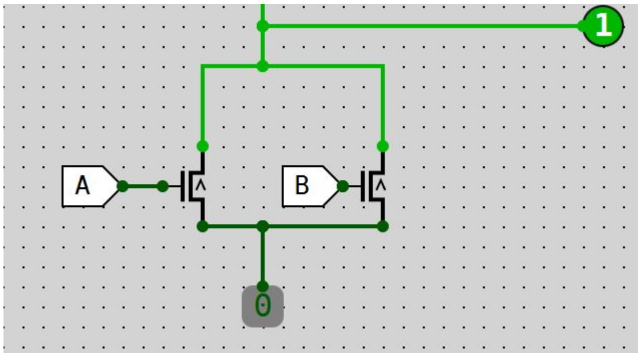
   (a) Pull up expression: $\overline{A \mid\mid B} == \overline{A}\ \&\&\ \overline{B}$

   (b) With the 'pull-up' network / network connected to voltage, since it is an 'and' operation, connect the complement of the inputs to the switches in series

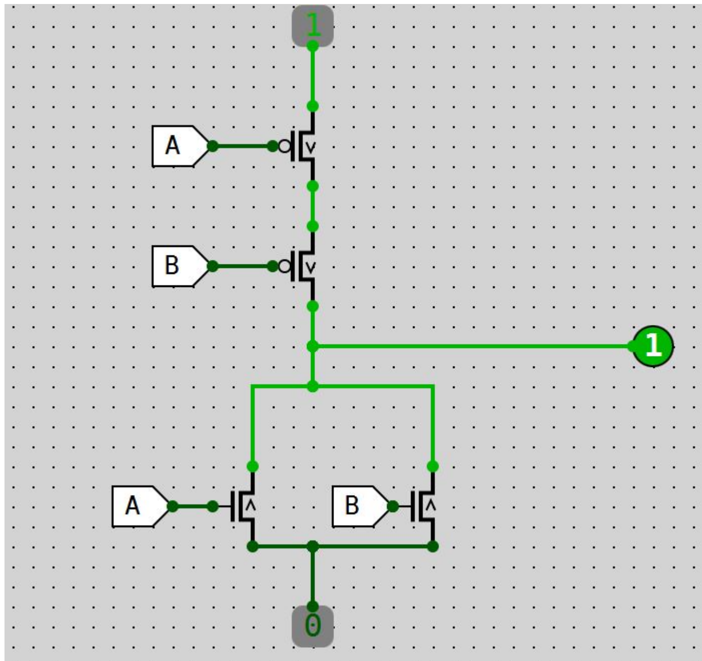
   

   (c) Pull down expression: $\overline{\overline{A}\&\&\overline{B}} = A||B$

   (d) With the 'pull-down' network / network connected to ground, since it is an 'or' operation, connect the inputs to the switches in parallel

   (e) NOR circuit using transistors



(f) This method is a fool-safe method of creating NAND and NOR gates. For an AND or OR gate, you will need to create a NAND/NOR gate respectively and add the NOT circuitry afterwards.

6 Hints when making transistors in circuitsim

- Make sure the arrow of the transistor is pointed towards the output