CSCB58:Computer Organization

Instructor: Brian Harrington

brian.harrington@utsc.utoronto.ca

* Original slides by Steve Engels

Why take CSCB58?

- To better understand computers!
- See what's going on "under the hood"
- Open the black box, get rid of the mystery
- Understand the whole pipeline, from atoms to assembly
  - Everything above assembly is virtualization and abstraction
  - Everything else is an **illusion!**
- Build a cool hardware-based project!
- Spend time with Brian

CSCB58 Asks the Big Questions

- What is a computer?
- What is memory? How does a computer store information
- Why do computers work in binary?
- What does a computer actually… compute?
- How does the code we write in python/java/C++/etc actually translate into things happening?

CSCB58 Course Goals

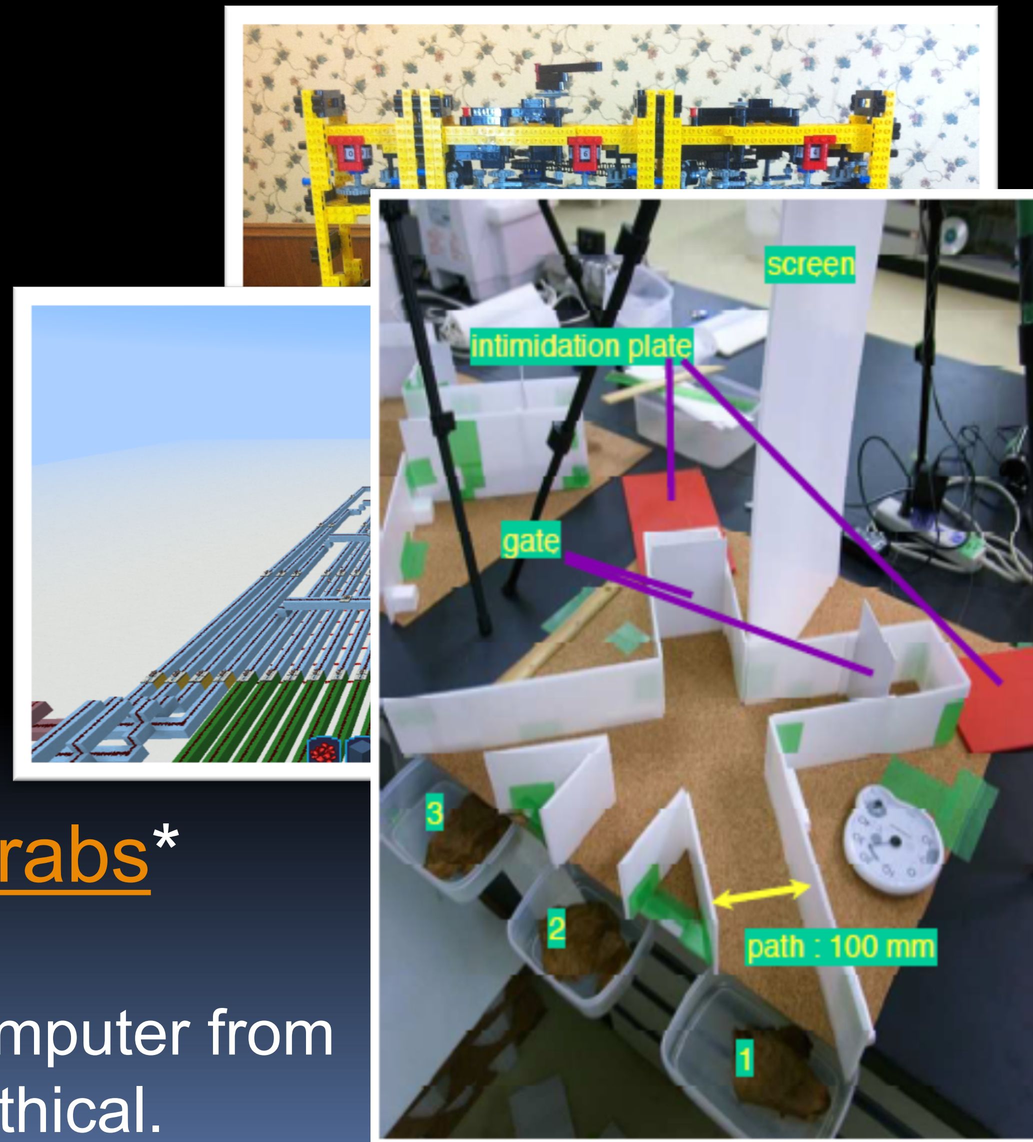
- Understand and design the underlying architecture (digital structures) of computer systems.
  - Build computers!
- Learn how to use these digital structures to do computation.
- Learn to build systems from components

CSCB58 Course Goals

- Given an infinite amount of time, and the ability to manipulate individual atoms, by the end of this course, you could:
  - Build transistors from atoms
  - Build logic gates from transistors
  - Build circuits from logic gates
  - Build memory/computational units from circuits
  - Build a computer from memory/computational units
- In short… you will be able to build a computer from the atomic level upwards

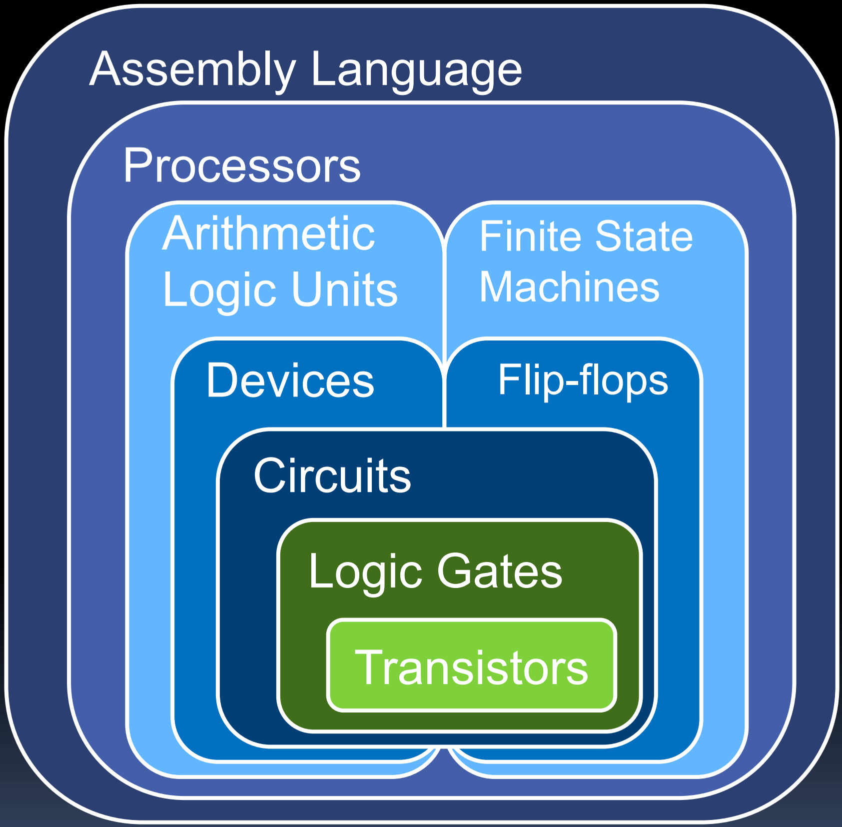
Build a Computer!

- …with LEGOs
- …in Minecraft
- ….from living crabs*



* Do not build a computer from live crabs. It is unethical.

The course at a glance…



Admin

- Course Website
  - https://www.utsc.utoronto.ca/~bharrington/cscb58
- Piazza Discussion Board
  - Use for all assignment or technical questions.
  - Read pinned posts (first posts at the top) before asking questions!
- E-mail (use official email only!)
  - For personal or other matters
- Office hours
  - See website

Course Structure

- Lectures
  - 2 Hour lecture= new material presented
  - 1 Hour lecture = review of previous week's material (inverted) or focus on labs
- Labs:
  - 7 labs (lab 0 is easy, 1-6 is much more work)
  - Must complete pre-lab exercises ahead of time.
  - There are no tutorials for this course (Just lectures and labs/practicals)
- Project:
  - 3 week project, large digital creation.
  - Done in teams of 2-4
- Exams:
  - Midterm – details TBA
  - Final exam - must get 40% to pass the course.

Marking Scheme

- Labs – 20%
  - Lab 0 - 2%
  - Labs 1-6 - 3% each
- Project – 20%
  - Proposal
  - Weekly Reports
  - Final Presentation + Video
  - Final Code Submission
- Term Tests – 25%
  - Details TBA
- Final Exam – 35%
  - Must score 40% on final to pass

Labs

- Starting week 2
  - Signed up online as practical sessions
  - All work will be done in pairs
    - Pairs come from your lab session (PRA001…)
    - Make your own pairs or we can randomly assign
- MUST complete pre-labs before you show up!
  - Lab time is very limited

Lab Submissions

- Before lab:
  - Print and bring lab handout
  - Do pre-lab work
- Bring to lab:
  - One filled lab printout per student
  - Print code and schematics: one per pair
- After lab:
  - Each student submits final code, including in-lab changes, to Quercus (same code for both students)
- You must remember to submit or you won’t get the marks for your work.

Things to note

- Lab time is limited, you're learning to work with limited time resources
- Whenever possible, use the tools.
- A lot of the material is simple on paper, but can be quite tricky to get right in hardware
  - "In theory there's no difference between practice and theory, but in practice there usually is"
- Lab rules are strictly enforced, NO food/drink allowed. Hardware is expensive, you break it, you're buying me a new one.

Preparing for Lab 0

- Experience is the best teacher.
  - Preparing a design. → Pre-Lab
  - Implementing your design.
  - Debugging the circuit.

  (Implementing your design + Debugging the circuit → In-Lab)

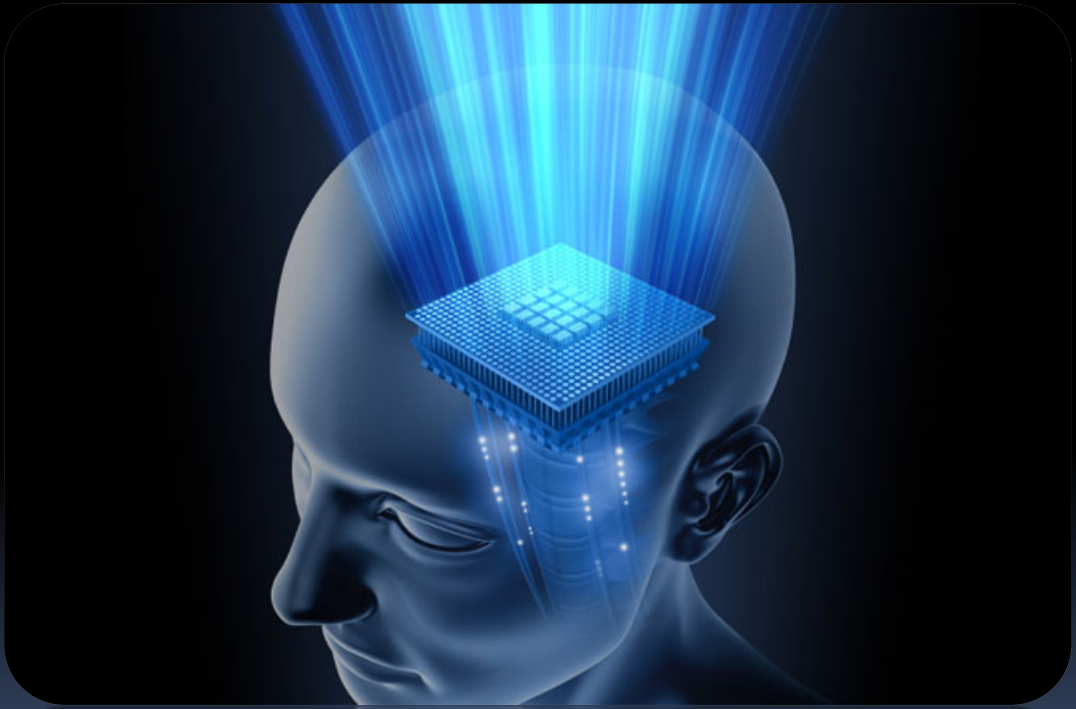
- Lab 0 will ask you to do things some things the “hard way”
  - We will learn systematic approaches as we go

How to do well in CSCB58

- Be Interested!
- Interact
  - lectures, labs (TAs), discussion board
- Work on pre-labs and labs
- Challenge yourself with a cool project
- Study for exams

Let the learning begin!

But you already know some important concepts…

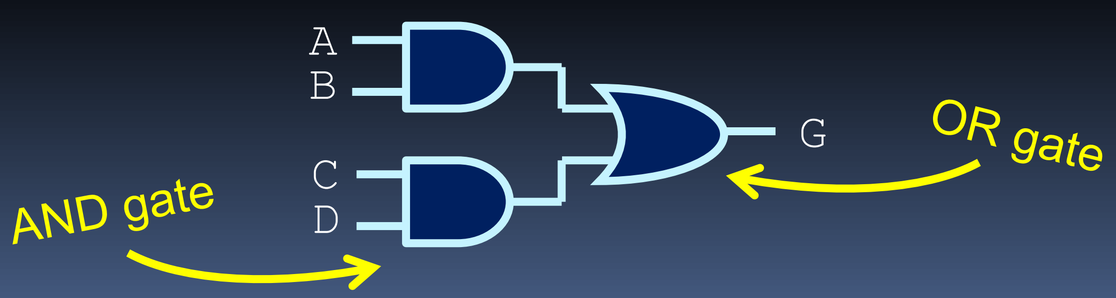


Boolean Logic from CSCA67

- <u>Example:</u> Create an expression that is true if the variables `A` and `B` are true, or `C` and `D` are true.

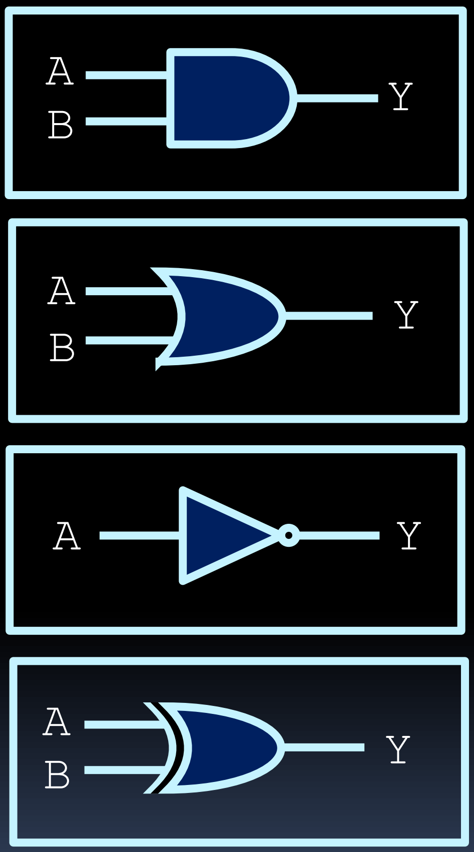
```
G = A & B | C & D
```

- Now create a circuit that does the same thing:

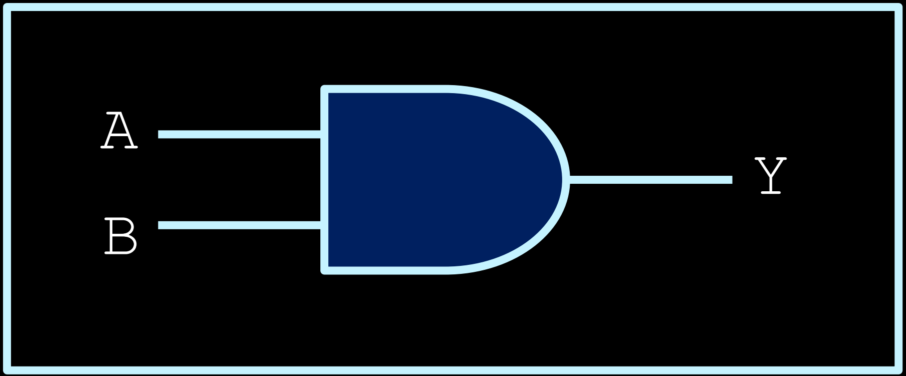


Logic Gates

- If you know how to create simple logical expressions, you already know the basics of putting logic gates together to form simple circuits.
- Just need to know which logic operations are represented by which gate!

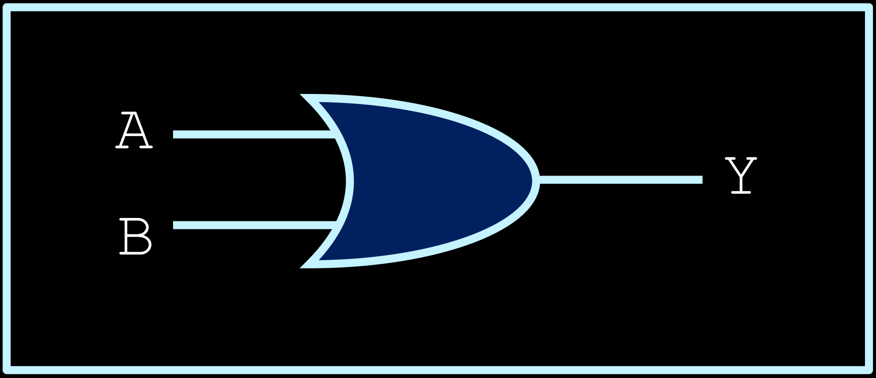


AND Gate



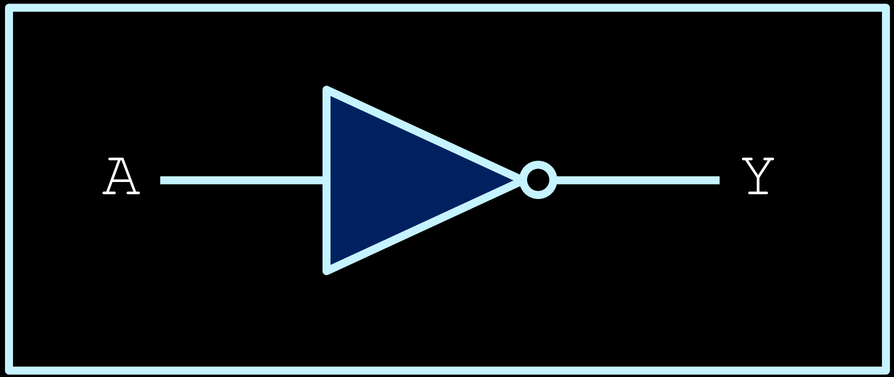
A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

OR Gate



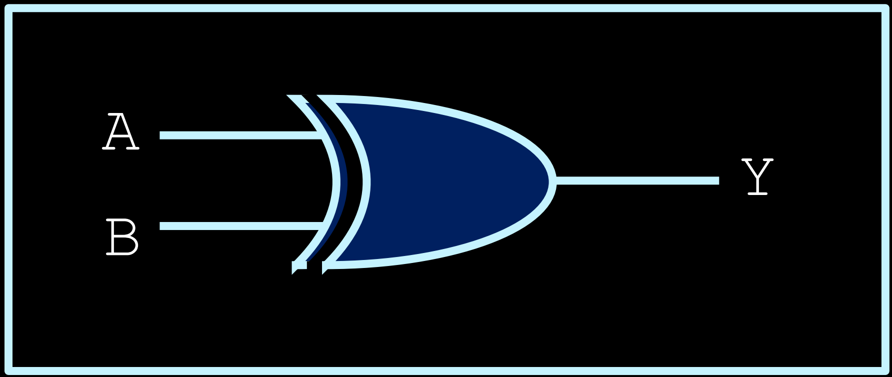
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

NOT Gates



A	Y
0	1
1	0

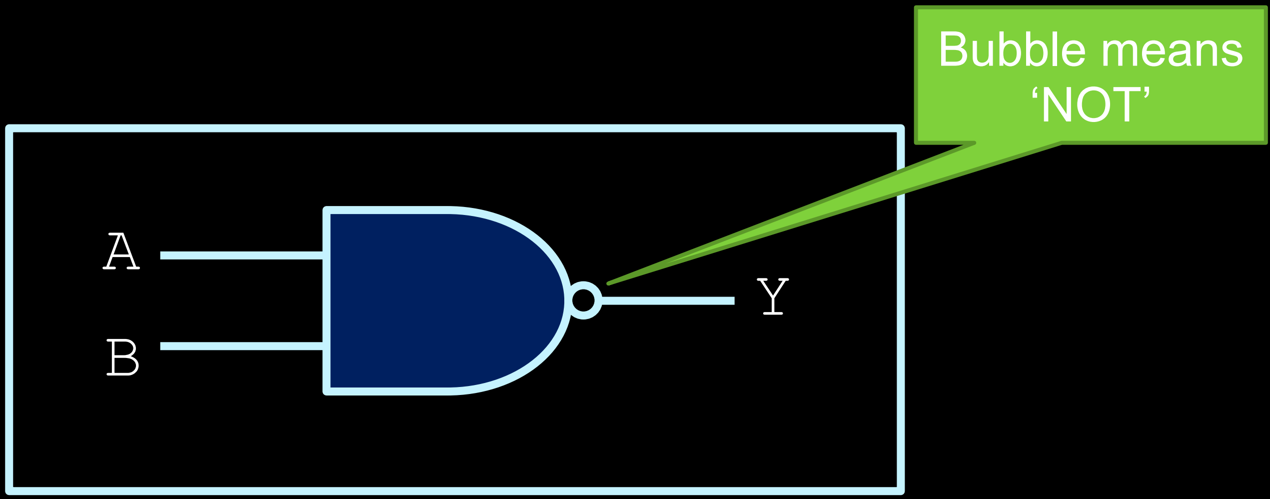
XOR Gates



A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

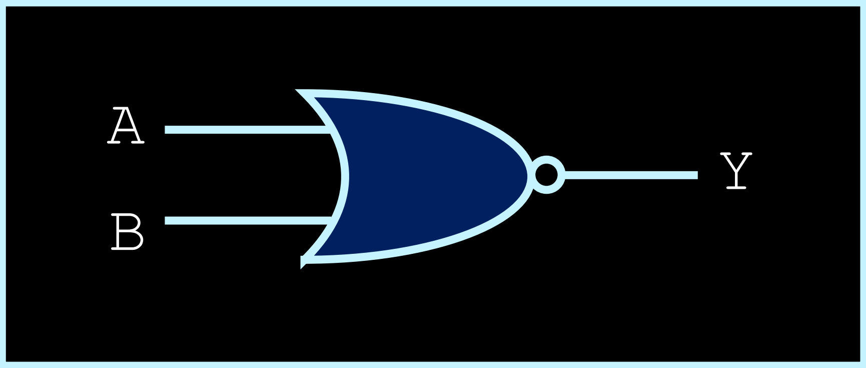
Bill Gates

NAND Gates



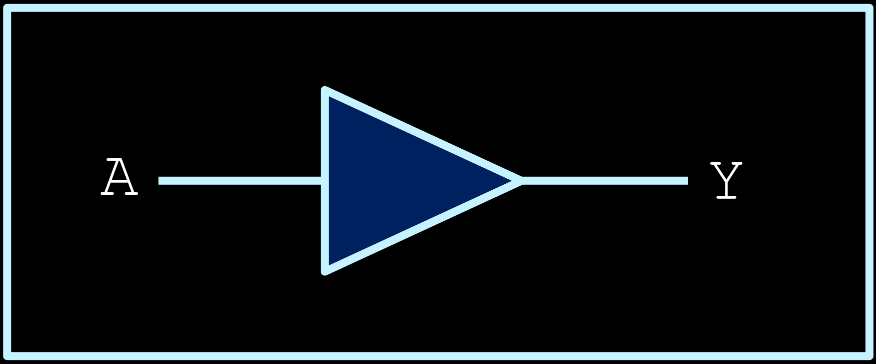
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

NOR Gates



A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

Buffer

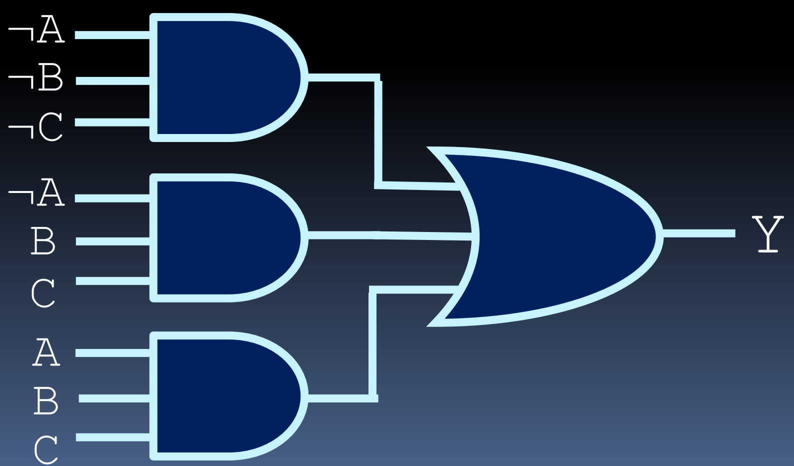


A	Y
0	0
1	1

Circuit Design (1)

- Creating circuit logic can be similar to creating Boolean logic in Python, C or Java:

```
Y = (!A and !B and !C) or (!A and
    B and C) or (A and B and C)
```



Circuit Design (2)

- Given a truth table or description, find a circuit that implements it.
- Many ways of tackling the problem
- More next week…

A	B	C	Y
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

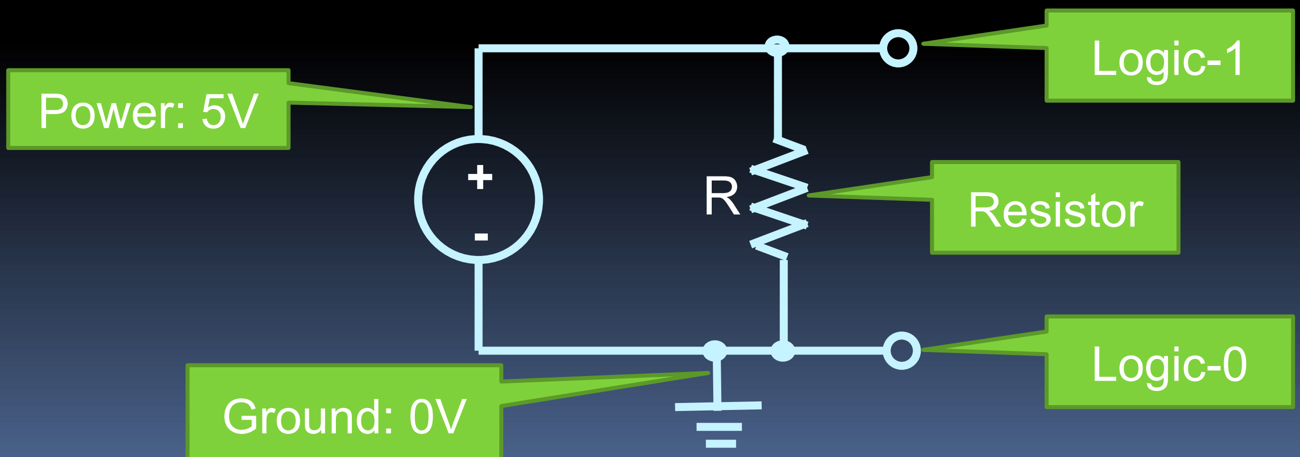
Hardware Design

- Designing Hardware fundamentally different from designing software
- Software – describe **a sequence of steps**, an algorithm
- Hardware – **describe circuits**, connections between logic gates and devices

- Very important concept to grasp early in this course!

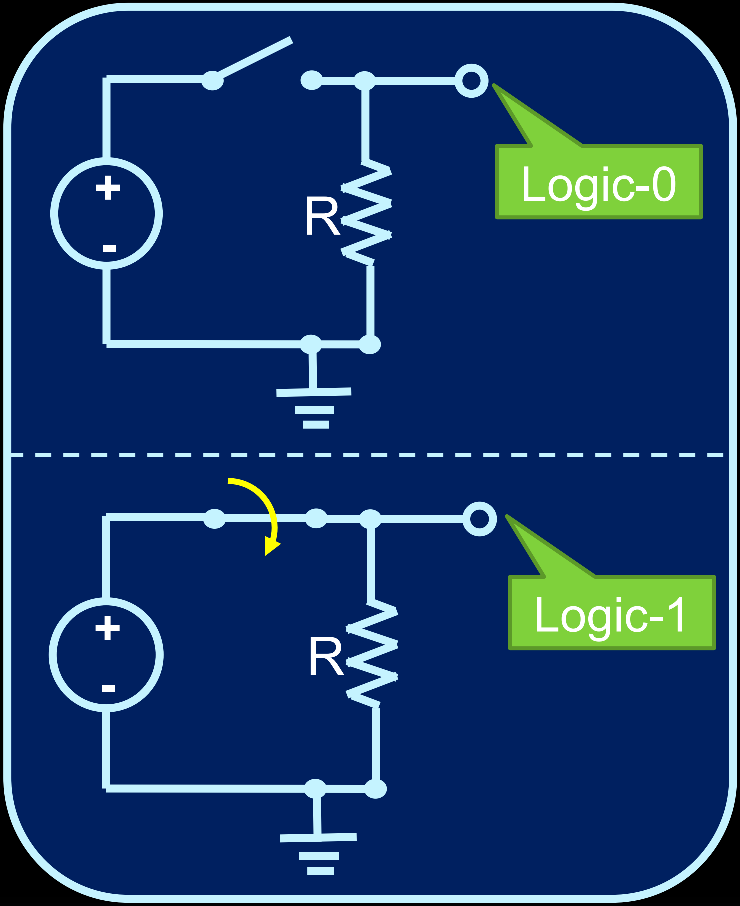
Physical Representation of Boolean values

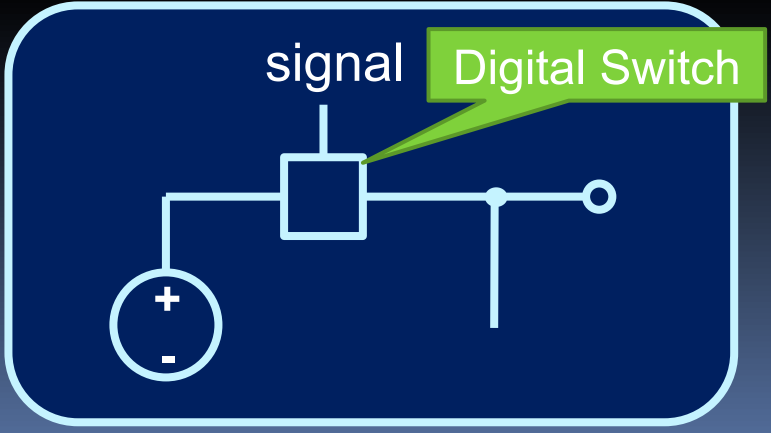
- “true” and “false” (1’s and 0’s) are represented by different electrical voltage values on the wires
- Logic-0: 0 volts, a low voltage
- Logic-1: 5 volts, a high voltage



Building logic gates

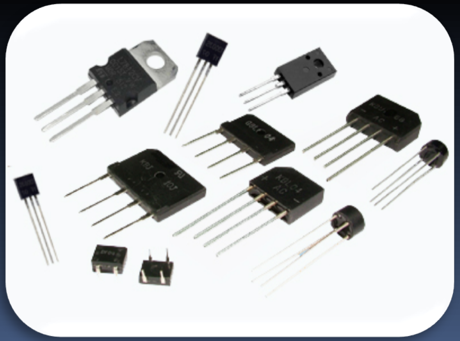
- Logic Gates behave like switches that control whether an output wire will have a high value (5V) or a low value (0V)
  - Switches are physical devices for manually controlling a circuit.
  - Transistors are semi-conductor devices that control a circuit electrically.



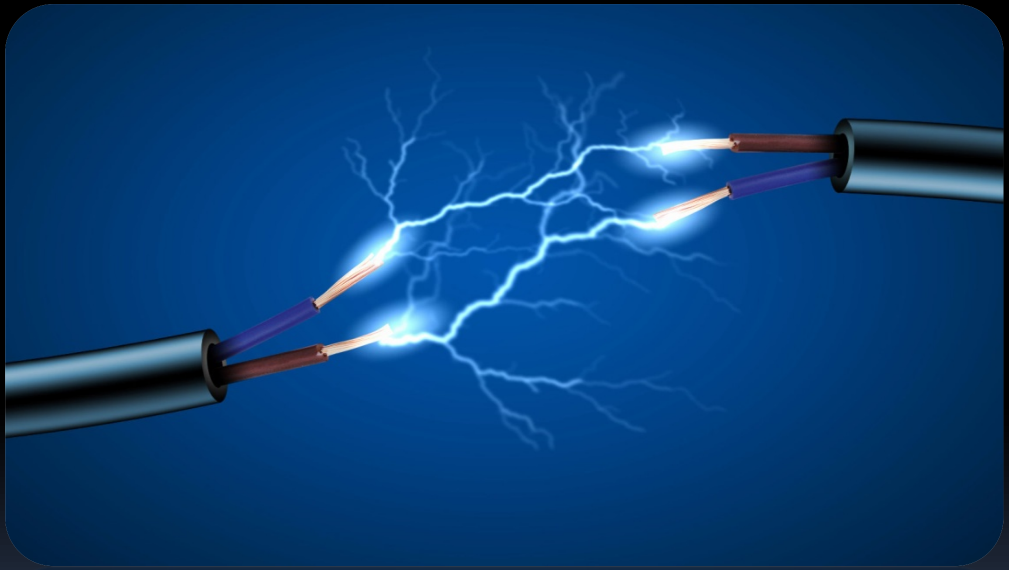


Starting from the bottom

- But how do the digital switches work?
  - First, we need to understand electricity.
  - Then, we need to understand transistors.

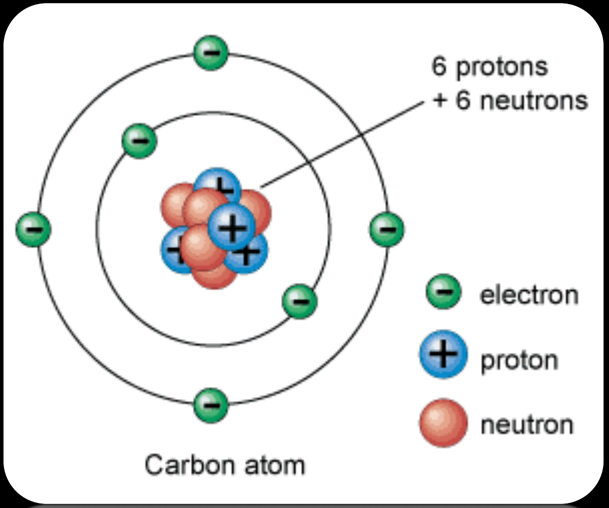


Intro to Electricity

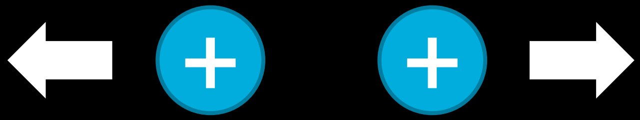



BRACE YOURSELVES
PHYSICS AND CHEMISTRY ARE COMING
imgflip.com

Electrical Particles



- Protons are positive
- Electrons are negative
- Same charge repel

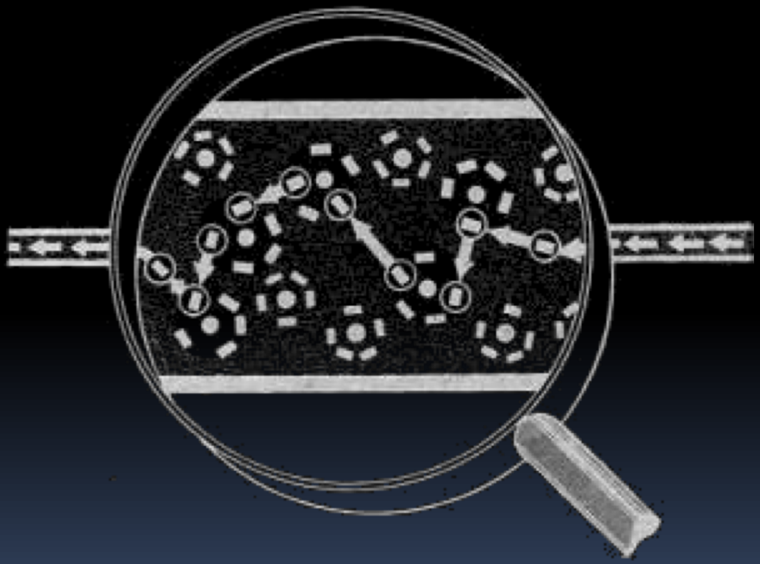


- Opposite charge attract

- What if you have many electrons in one place?

Electricity

- Electricity is the flow of charged particles (usually electrons) through a material.



Potential and Flow

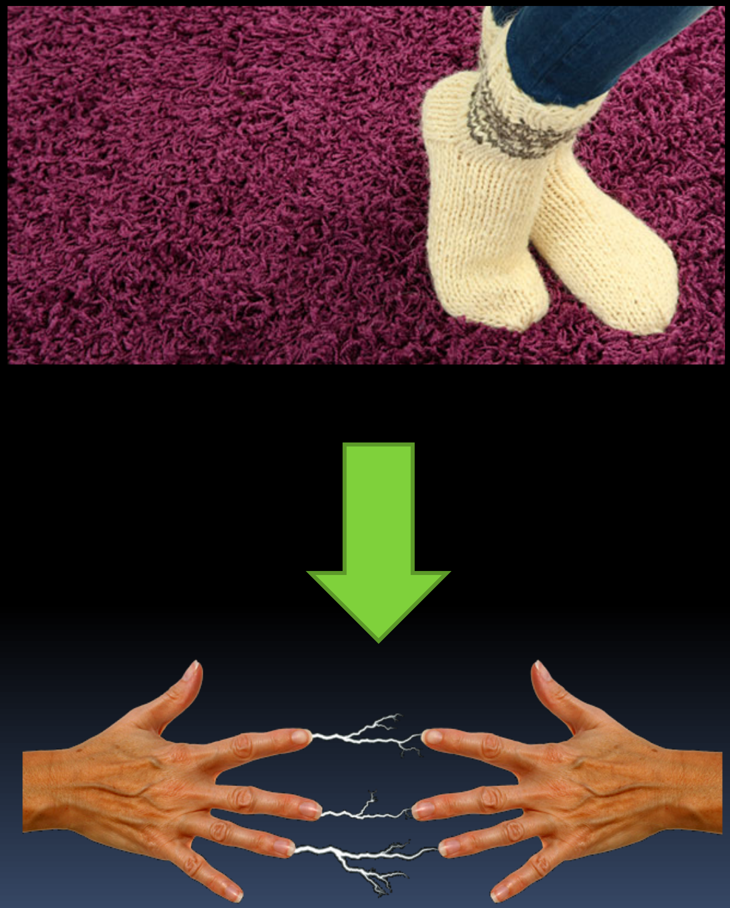
- Electrical particles want to flow from regions of high electrical potential (many electrons) to regions of low electrical potential (few electrons)
  - Similar to gravitational potential
- This difference in potential is referred to as voltage.
- The rate of this flow is called the current.

Water Analogy

- To help picture this concept of voltage and current, imagine a reservoir:
  - Electrons flow from high to low potential like water would flow from the reservoir to the ground.
  - Voltage is like the elevation of the water above the ground.
  - Current is the rate at which the water flows.
- The relationship between voltage (V) and current (I) is called resistance: $R = V/I$

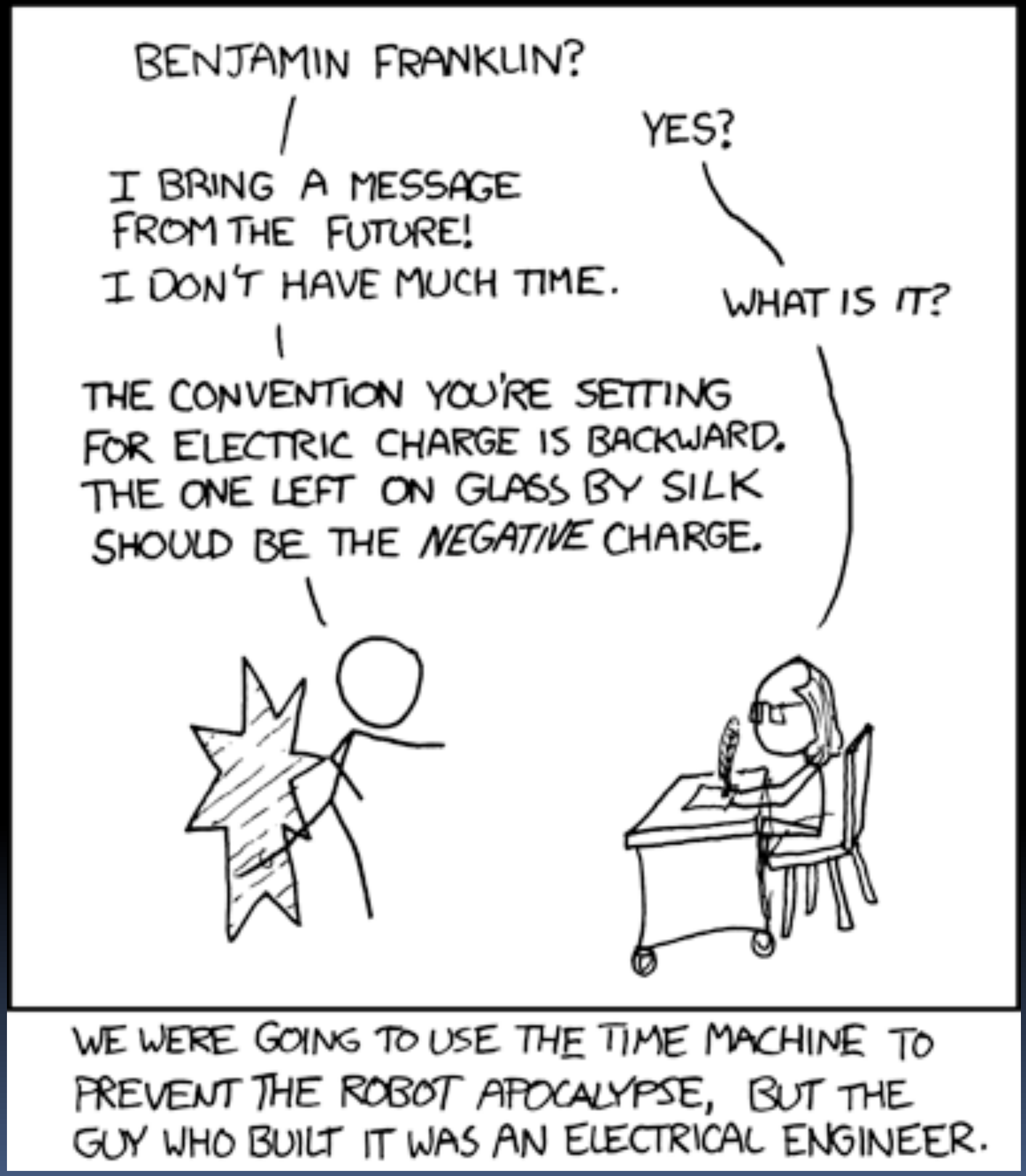
Static electricity example

- When you shuffle your feet back and forth on a carpet, you pick up extra electrons in your body and develop an electrical imbalance, relative to the ground.
- When you touch an object or person who is electrically balanced, those extra electrons transfer over to that object or person.



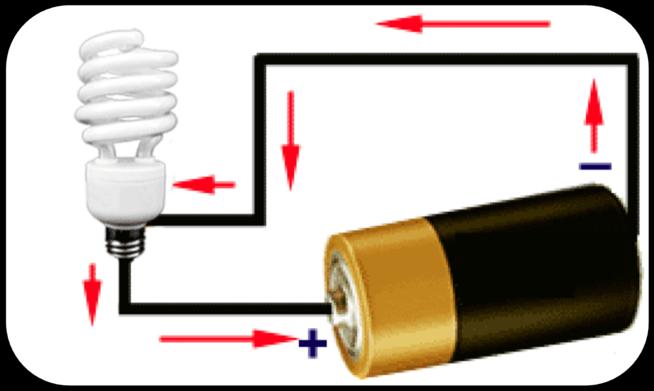
About the path of electricity

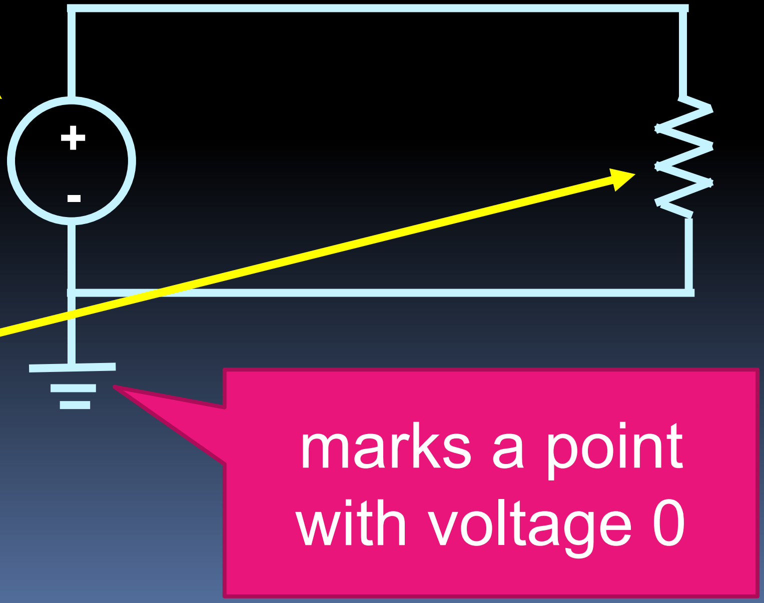
- Current always flows toward the zero voltage point of a circuit.
  - Commonly referred to as ground.
- Electricity always likes to take the path of least resistance, from source to ground.
- Even though electrical current is the flow of electrons through a medium, its direction is typically expressed as *the movement of the positive charges*.


BENJAMIN FRANKLIN?
YES?
I BRING A MESSAGE FROM THE FUTURE!
I DON'T HAVE MUCH TIME.
WHAT IS IT?
THE CONVENTION YOU'RE SETTING FOR ELECTRIC CHARGE IS BACKWARD. THE ONE LEFT ON GLASS BY SILK SHOULD BE THE *NEGATIVE* CHARGE.
WE WERE GOING TO USE THE TIME MACHINE TO PREVENT THE ROBOT APOCALYPSE, BUT THE GUY WHO BUILT IT WAS AN ELECTRICAL ENGINEER.

Using electricity

- Electronic particles want to travel from areas of high concentration to areas of low concentration.
- We can use this knowledge to drive circuits.
  - A circuit has a source of electrical particles, some path between this source and the ground, and some resistance along this path that dissipates these electrons.

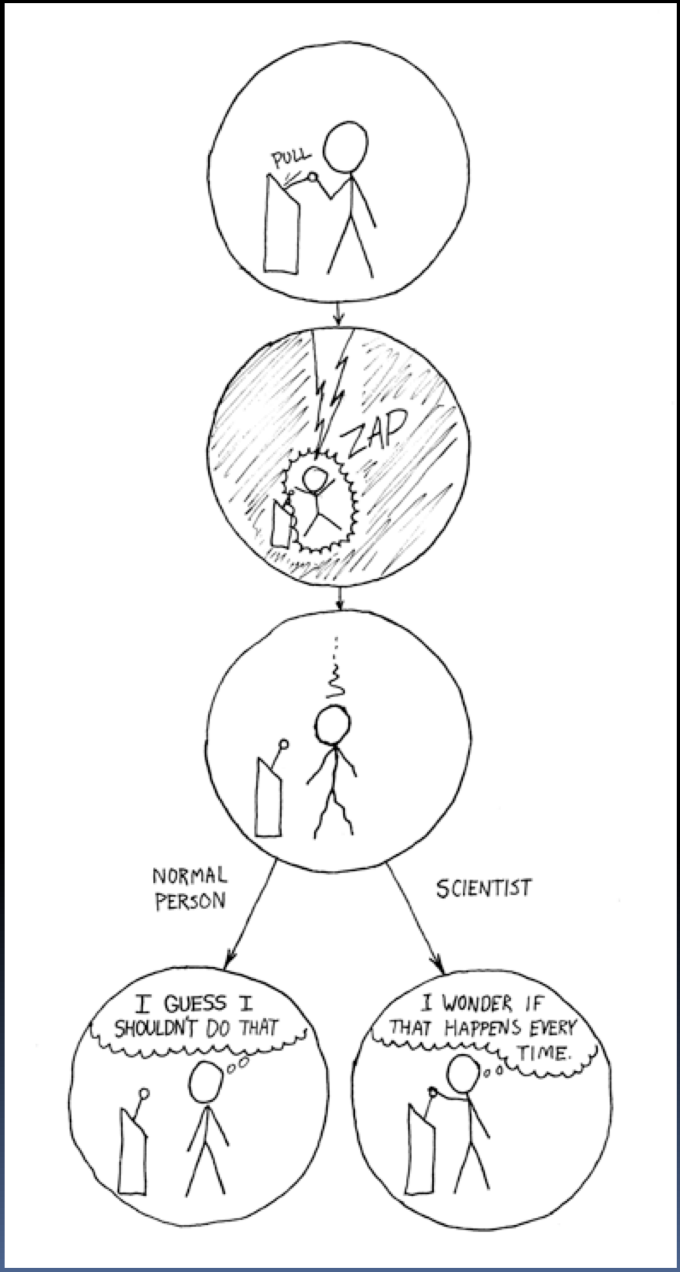




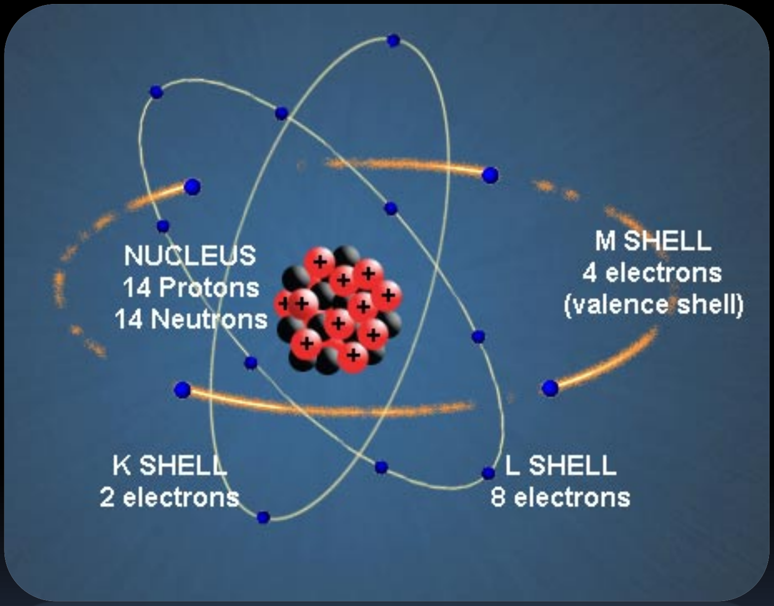
Resistance

- In the water analogy, resistance would be measure how restrictive the pipe is that connects the reservoir to the ground.
  - Wide, smooth pipe = low resistance
  - Narrow, twisty pipe = high resistance
- Electrical resistance indicates how well a material allows electricity to flow through it:
  - High resistance (aka insulators) don't conduct electricity at all, or only under special circumstances.
  - Low resistance (aka conductors) conduct electricity well, and are generally used for wires.
  - These are largely determined by the position on the element on the periodic table.
  - Measured in ohms (Ω). More ohms, more resistance.
- Semiconductors are somewhere in between conductors and insulators.

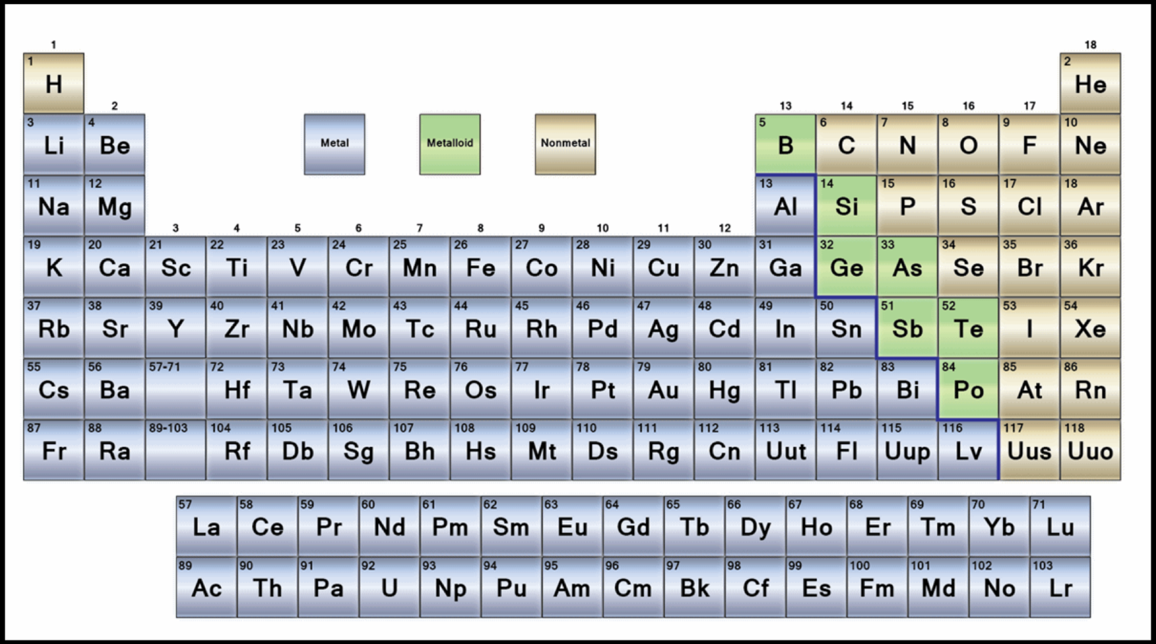
Break


PULL
ZAP
NORMAL PERSON
SCIENTIST
I GUESS I SHOULDN'T DO THAT
I WONDER IF THAT HAPPENS EVERY TIME.

Semiconductors



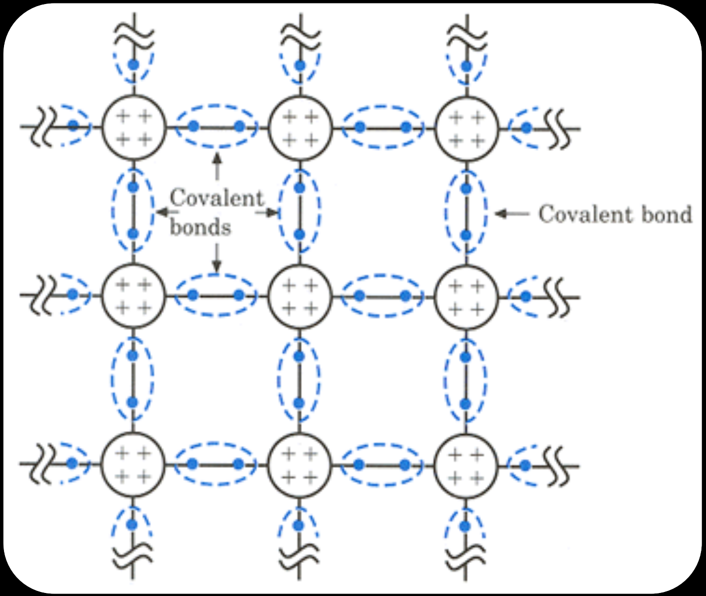
Conductivity



Semiconductor materials (silicon, germanium) straddle the boundary between conductors and insulators, behaving like one or the other, depending on factors like temperature and impurities in the material.

Semiconductor Conductivity

- Semiconductors are solid and stable at room temperature, but energy can make electrons from the valence layer become loose.

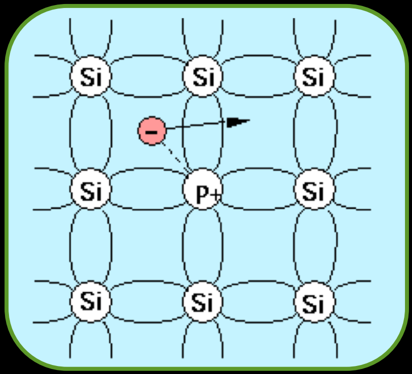


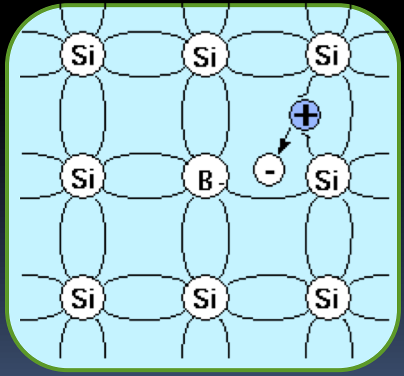
- At room temperature, a weak current will flow through the material, much less than that of a conductor.

Doping

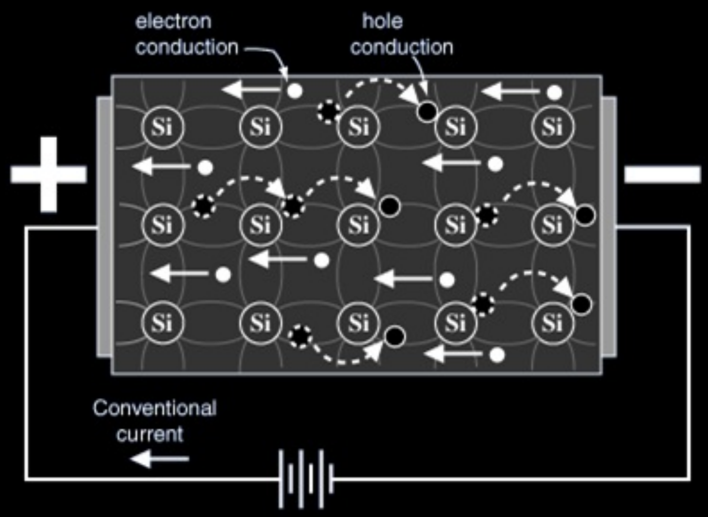
Doping

- Semiconductors don't conduct electricity naturally.
  - Because the valence electrons of each atom are joined to their neighbours.
- To encourage the semiconductor's conductivity, impurities can be introduced in the fabrication process, increasing the number of free charge carriers.
  - n-type: adding elements which have 5 electrons in their valence layer (e.g. phosphorus, arsenic).
  - p-type: adding elements which have 3 electrons in their valence layer (e.g. boron).
  - This process is referred to as doping the semiconductor.
- These are neutral (there is no charge)





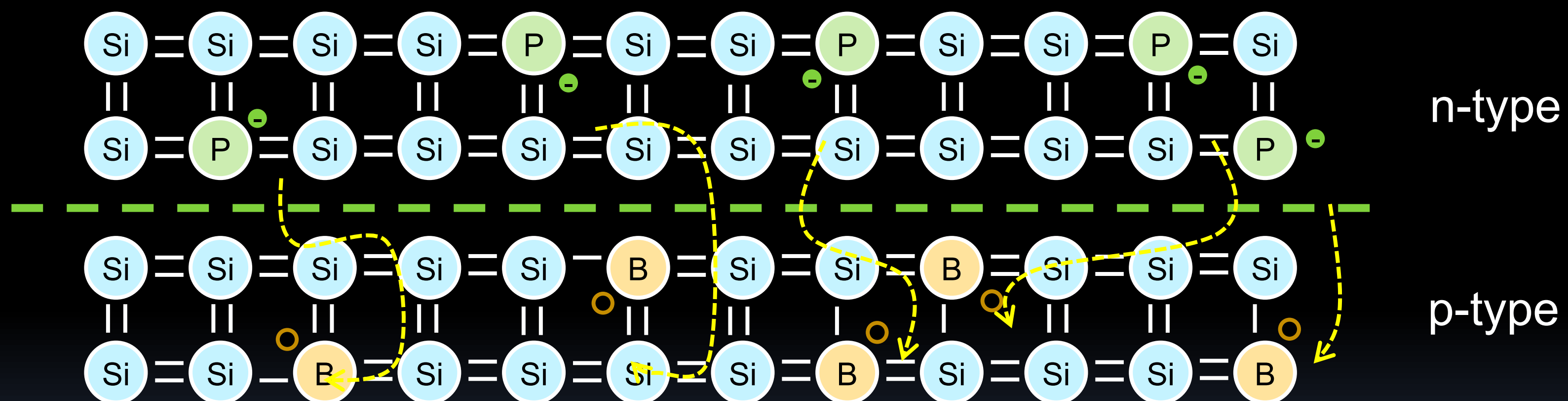
Charge Carriers



- n-type semiconductors, the charge carriers are electrons that are not bound to the crystal
- p-type semiconductors, the carriers are called holes, to represent the absence of an electron

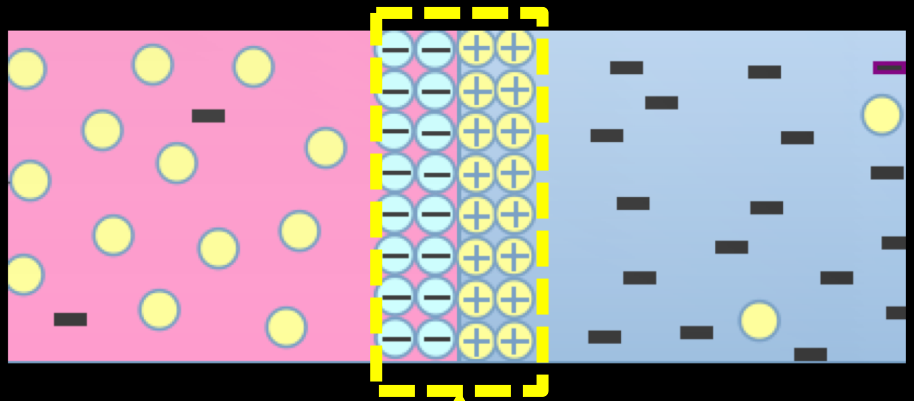
Bringing p and n together

- What would happen if you brought some p-type material into contact with some n-type material?



- The electrons at the surface of the n-type material are drawn to the holes in the p-type.

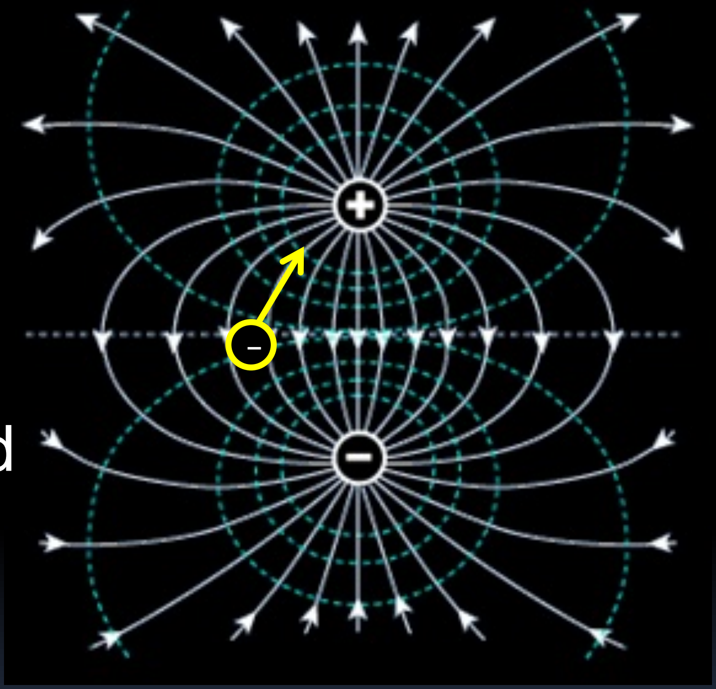
p-n Junctions (1)



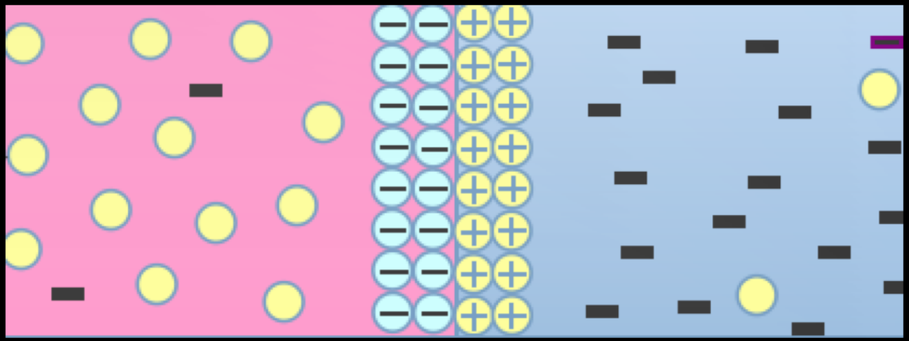
- When left alone, the electrons from the n section of the junction will mix with the holes of the p section, cancelling each other out, and creating a section without charge carries: called the depletion layer.
- Once this depletion layer is wide enough, the doping atoms that remain will create an electric field in that region.

Electric fields

- What is an electric field?
  - Vector of forces that a charge would feel relative to other charges
- If an electron was dropped between the two charges on the right, it would be attracted to the +ve charge and repelled by the –ve, in the direction of the field lines.

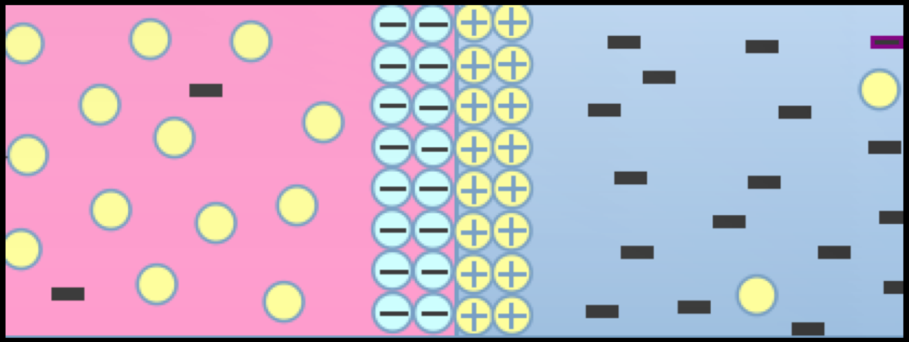


p-n Junctions (2)



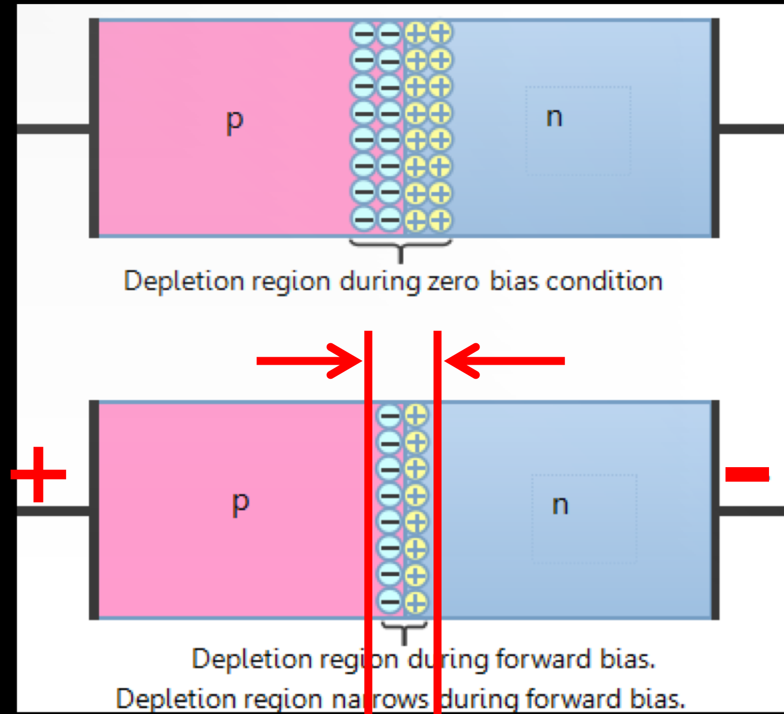
- When a phosphorus atom (n-type) loses its electron, the atom develops an overall positive charge.
- When a boron atom (p-type) takes on an extra electron, that atom develops an overall negative charge.

p-n Junctions (3)



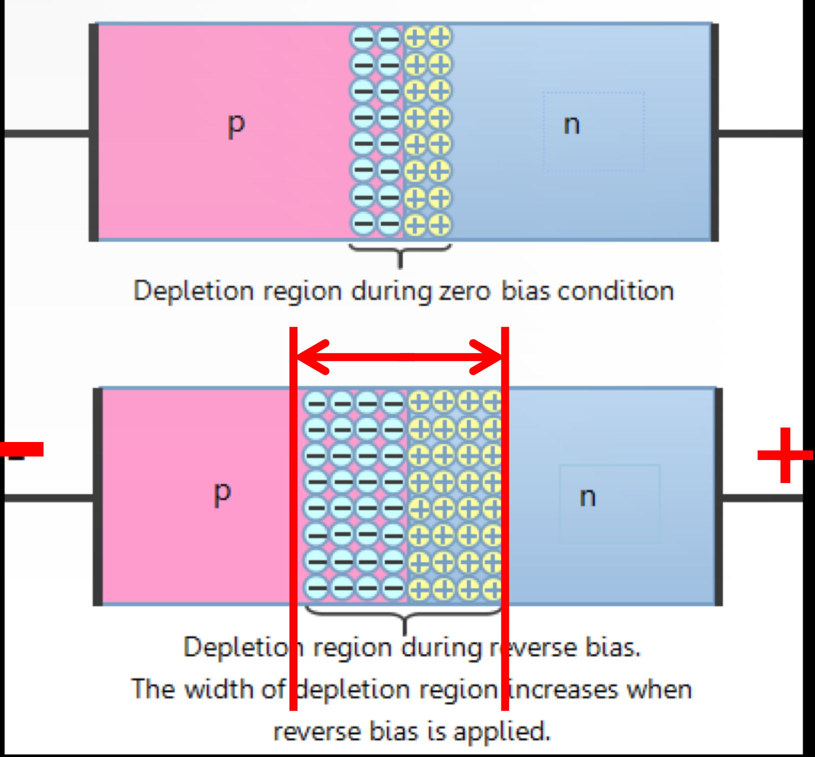
- The electric field caused by these atoms will cause holes to flow back to the p section, and electrons to flow back to the n section.
- The current caused by this electric field is called the drift current.
- The current caused by the initial electron/hole recombination is called the diffusion current.
- At rest, these two currents reach equilibrium.

Forward Bias

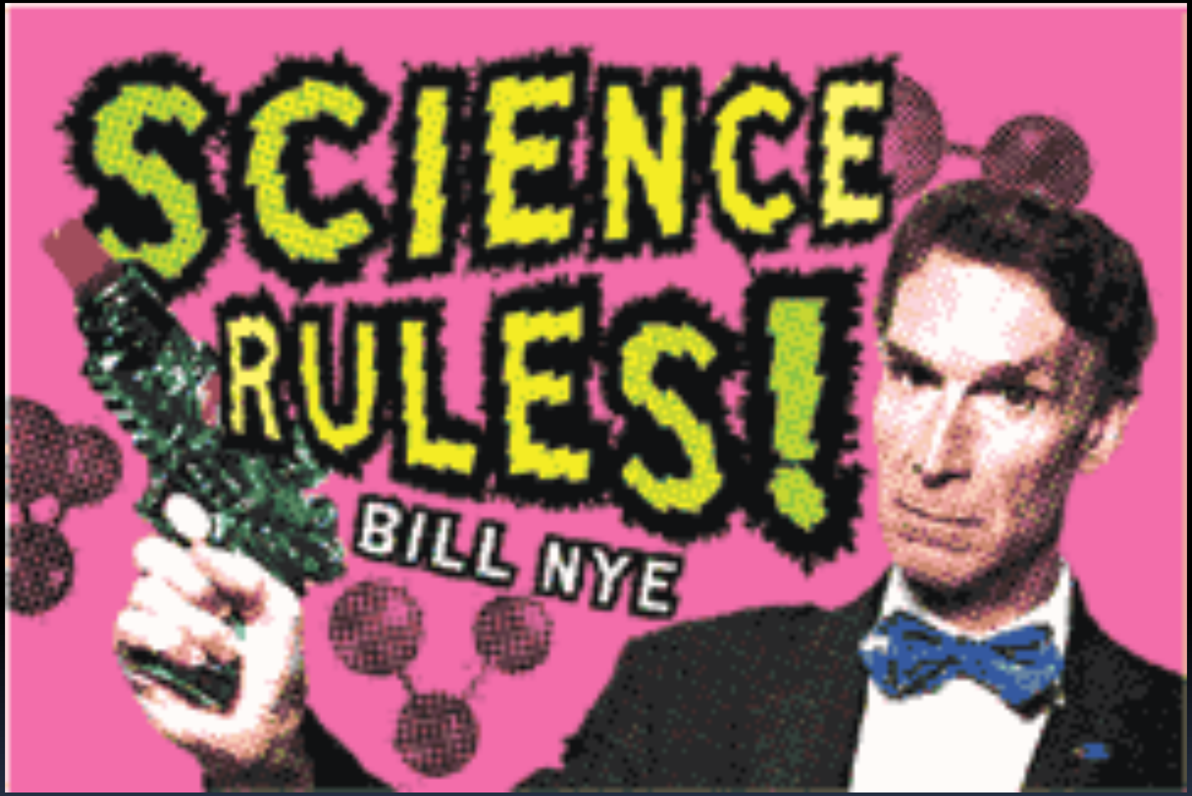


- What happens when a voltage is applied to this junction?
  - It depends on the direction in which the voltage is applied.
- Forward bias:
  - When a positive voltage is applied to the p end of this junction, electrons are injected into the n-type section.
  - This narrows the depletion layer and increasing the electron diffusion current.
- With a smaller depletion layer, the electrons travel more easily through pn-junction

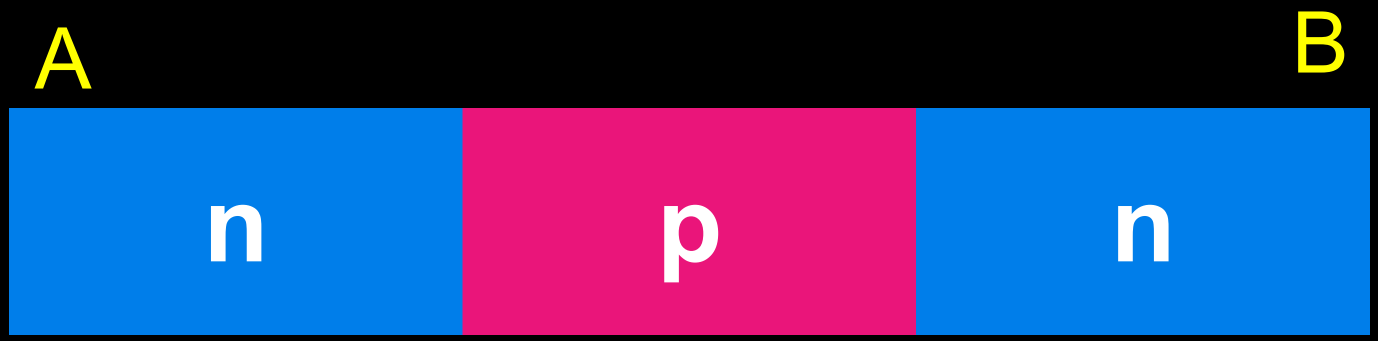
Reverse Bias



- Reverse bias:
  - When a positive voltage is applied to the n side of the junction, the depletion region at the junction becomes wider
  - A small current still flows through the circuit, but it is weak and does not increase with an increase in the applied voltage.
- So when a junction is forward biased, it becomes like a virtual short-circuit, and when the junction is reverse biased, it becomes like a virtual open-circuit.
  - This is the basis of transistors!


SCIENCE RULES!
BILL NYE

What Happens If I Do This…?



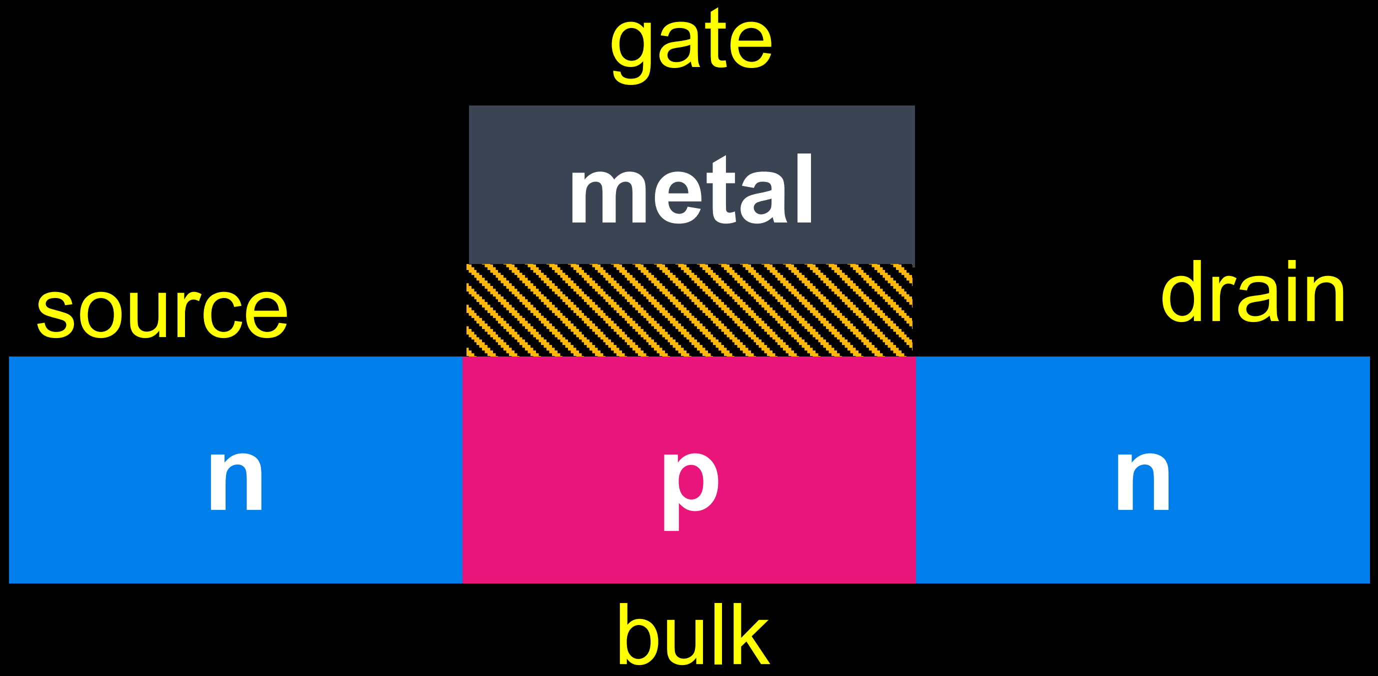
- Source on A, drain on B?

→ No current

- Source on B, drain on A?

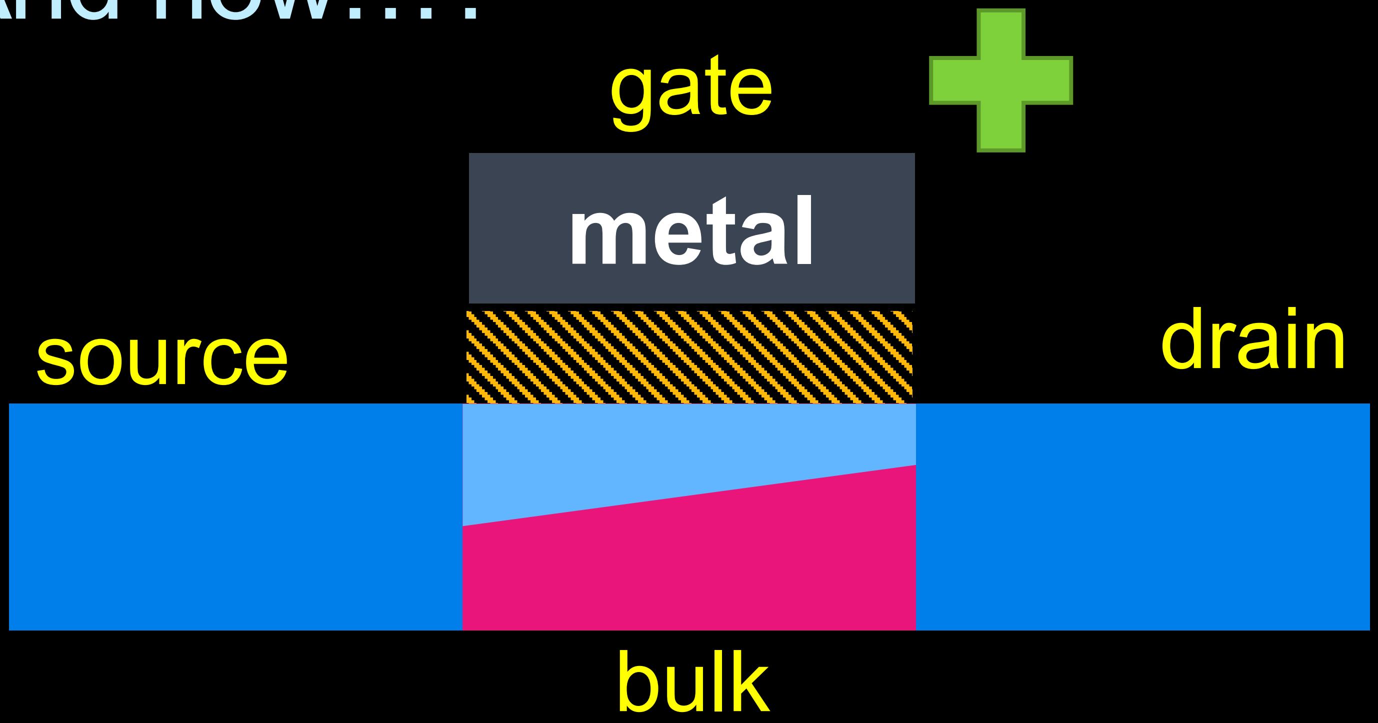
→ No current

A Tweak



- We added a metal gate
- We added an silicon insulator

And now…?



- Positive charge on gate attracts electrons.
- Electrons form an n-type channel
- Current can flow from source to drain!

Transistors

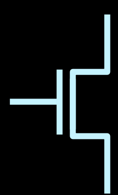
- Transistors form the basic building blocks of all computer hardware.
- Invented by William Shockley, John Bardeen and Walter Brattain in 1947, replacing previous vacuum-tube technology.
  - Won Nobel Prize for Physics in 1956.
- Used for applications such as amplification, switching and digital logic design.

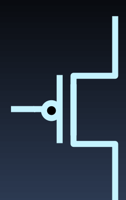
MOSFET

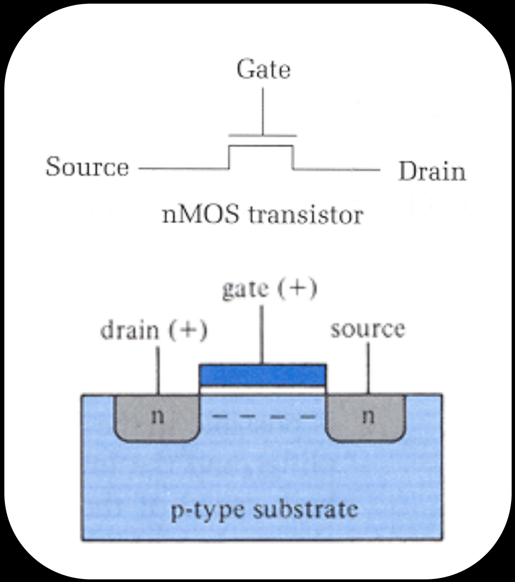
- Metal Oxide Semiconductor Field Effect Transistors are composed of a layer of semiconductor material, with two layers on top of the semiconductor:
  - An oxide layer that doesn't conduct electricity,
  - A metal layer (called the gate), that can have an electric charge applied to it
  - These are the M and O components of MOSFETs.

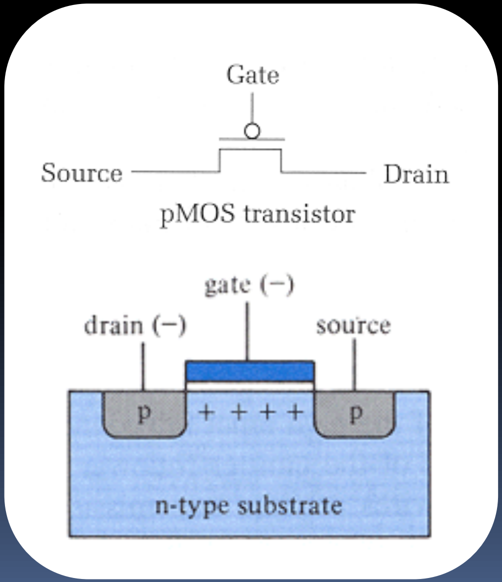
nMOS vs pMOS

- Two types of MOSFETs exist, based on the semiconductor type in the drain and source, and the channel formed.
  - nMOS transistors (the design described so far) conduct electricity when a positive voltage (5V) is applied to the gate.
  - pMOS transistors (indicated by a small circle above the gate) conduct electricity (i.e., act as a closed switch) when the gate voltage is logic-zero.









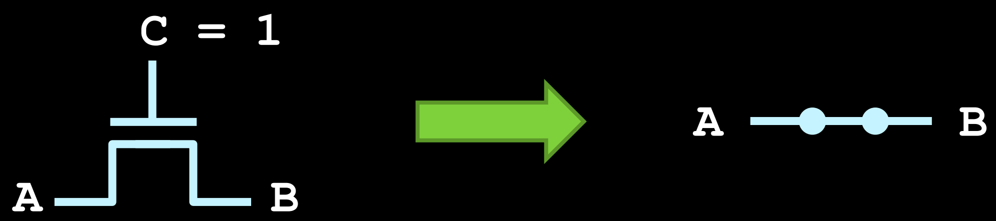
© Steve Engels, University of Toronto

Veritassium: How Does a Transistor Work?

https://youtu.be/IcrBqCFLHIY

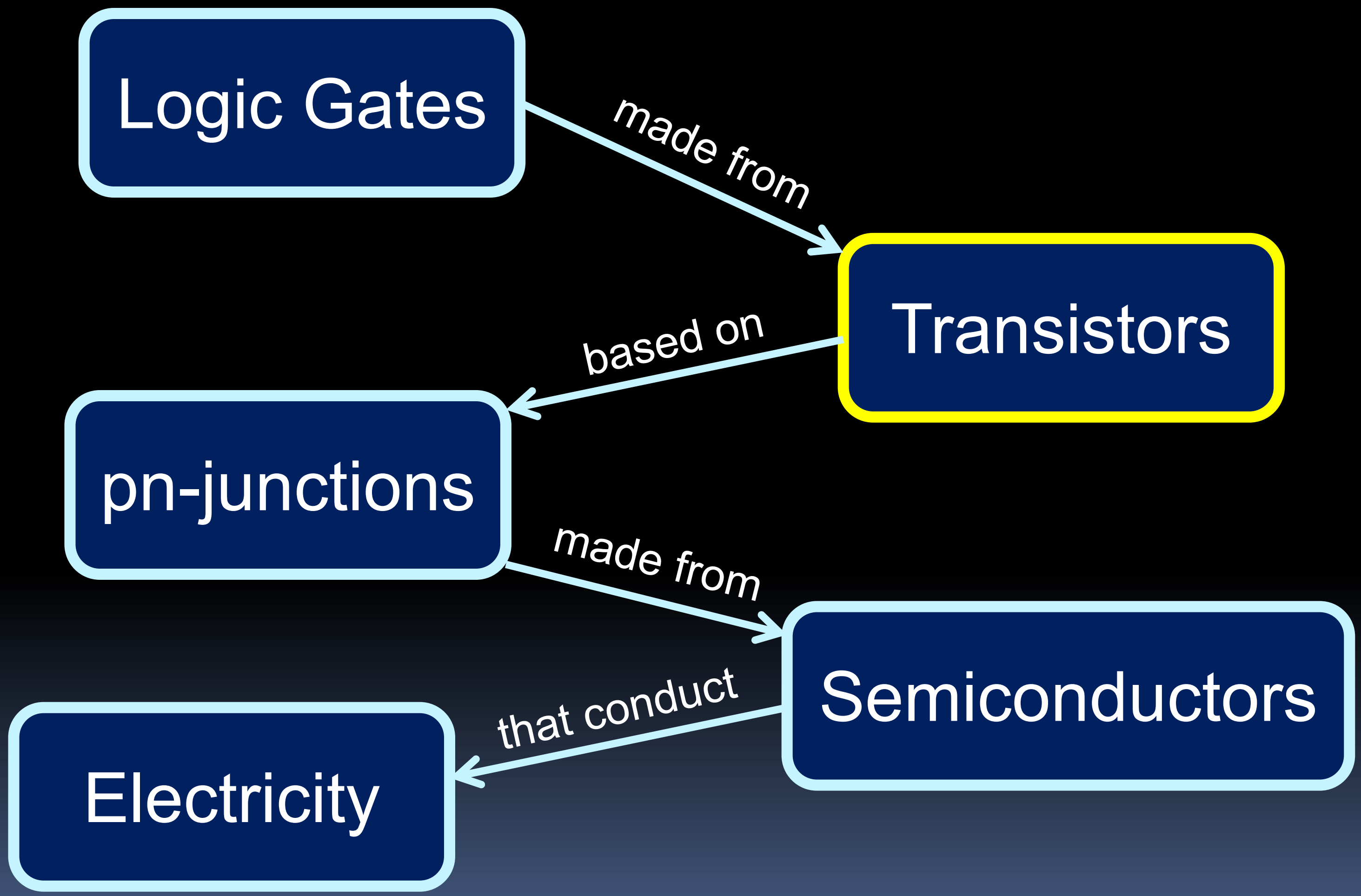
What do transistors do?

- Transistors connect Point A to Point B, based on the value at Point C (the gate).
- If the value at gate is high, A and B are connected.



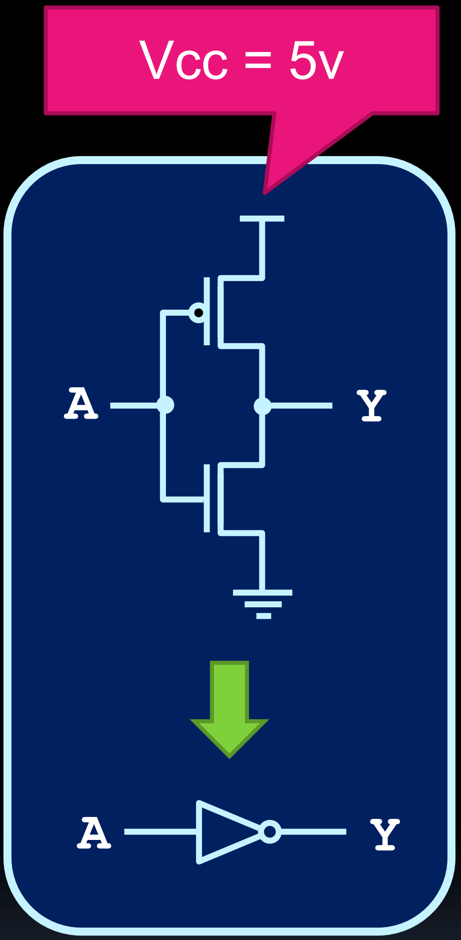
- And if the value at gate is low, A and B are disconnected.

Where do transistors fit?



Making gates

- Since these transistors aren't simply on/off switches, digital logic gates (AND, OR, NOT) are created by a combination of transistors
- Combine pMOS and nMOS
  - Examples: NOT gate circuit in diagram.
- Called CMOS for Complementary MOSFET

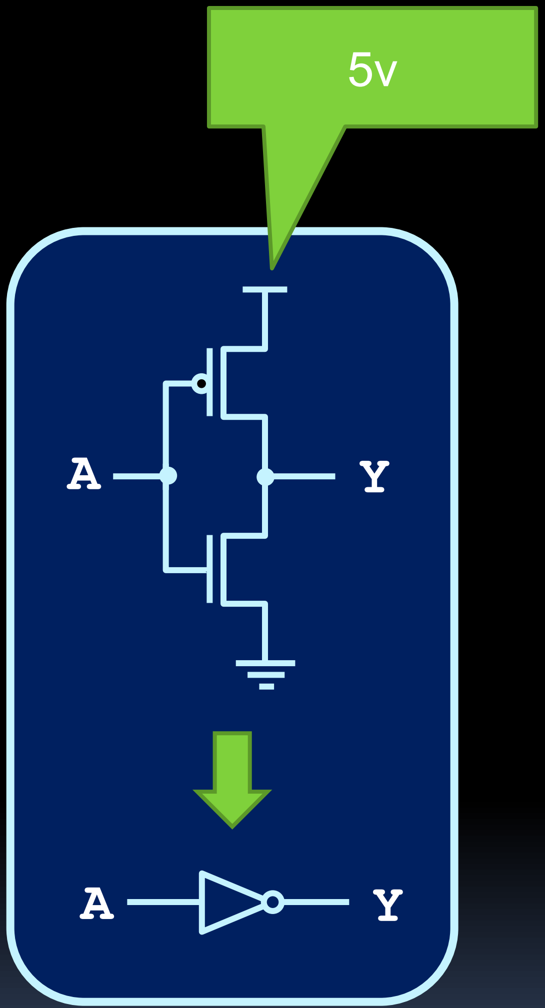


© Steve Engels, University of Toronto

NOT gate

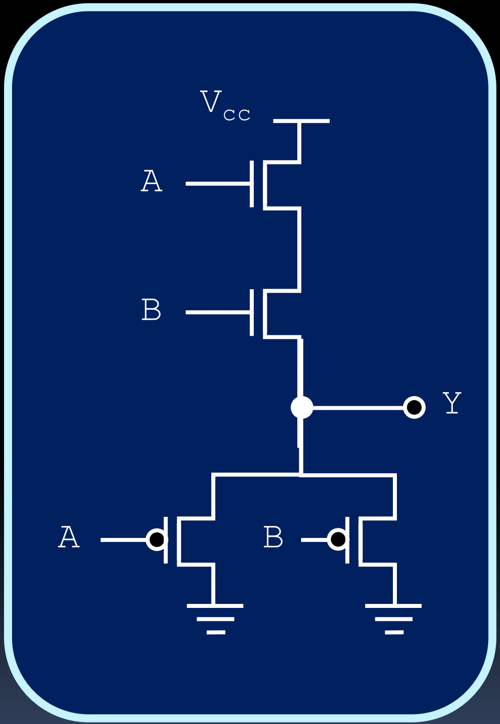
Note that:

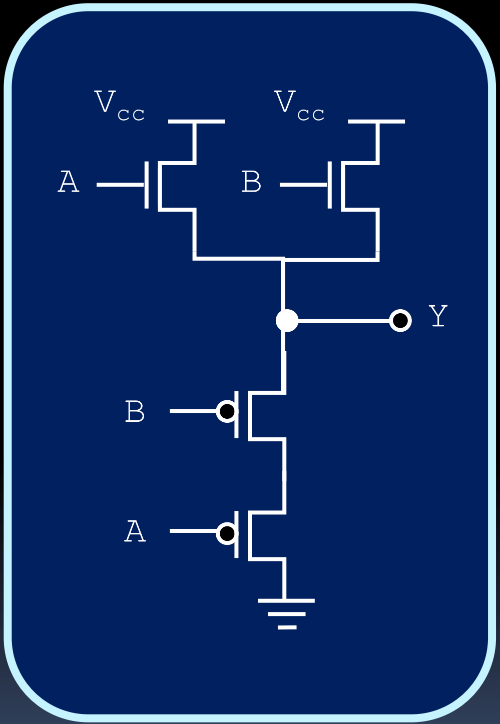
- Every input assignment makes a path from output to either Vcc or ground
- Never both.
- Output is never left “floating” (high-Z)
  - There are exceptions

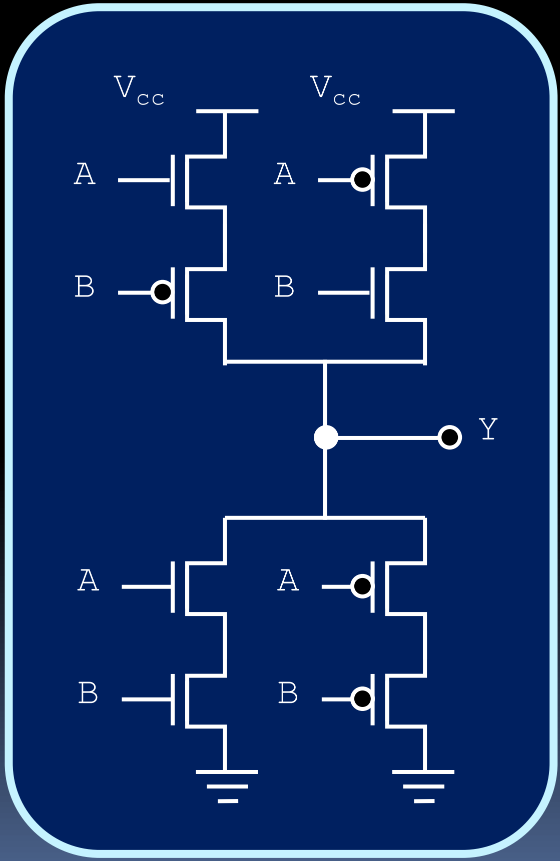


More circuit examples

What logic gates do these circuits implement?







Challenge Yourself

- Design a circuit for NAND
  - What's so great about NAND?
  - Can you make AND, OR, NOT, etc using only NAND?