Basic Curcitiry

Reference Manual

Contents

0 Quick Reference **4**
- 0.1 Pre-Midterm Equations . . . 5
- 0.2 Pre-Midterm Key Concepts . . . 5
- 0.3 Post-Midterm Equations . . . 5
- 0.4 Post-Midterm Key Concepts . . . 6

1 An Introduction to Making **7**
- 1.1 About This Guide . . . 8
- 1.2 Course Goals . . . 8
- 1.3 Class Projects . . . 8
- 1.4 Problem Solving . . . 9
- 1.5 Course Work . . . 9
- 1.6 Office Hours . . . 9
- 1.7 Exams . . . 9

2 Lab Reference **10**
- 2.1 Prelab . . . 11
- 2.2 Safety . . . 11
- 2.3 Cleanup . . . 11
- 2.4 Cable Colors . . . 11
- 2.5 Digital Multimeter . . . 11
- 2.6 Resistors . . . 12
- 2.7 Diodes . . . 13
- 2.8 Soldering . . . 13
- 2.9 Breadboarding . . . 14

3 Circuit Basics **16**
- 3.1 Terminology . . . 17
- 3.2 Charge . . . 17
- 3.3 Current . . . 18
- 3.4 Kirchoff's Current Law . . . 18
- 3.5 Voltage . . . 20
- 3.6 Kirchoff's Voltage Law . . . 20
- 3.7 Power . . . 22
- 3.8 Calculating Power . . . 22
- 3.9 Circuit Solving Tips . . . 23

4 Devices: Resistors **24**
- 4.1 I-V Curves . . . 25
- 4.2 Resistors . . . 25
- 4.3 Voltage Dividers . . . 26
- 4.4 Current Dividers . . . 29

5 Devices: Diodes **31**
5.1 Ideal Diodes . . . 31
5.2 Idealized Diode . . . 32
5.3 Solar Cells . . . 33

6 Solving Circuits **34**
6.1 Nodal Analysis . . . 34
6.2 Thevenin Equivalents . . . 38
6.3 Norton Equivalents . . . 43
6.4 Superposition . . . 48

7 Digital Logic Basics **51**
7.1 Switches . . . 51
7.2 Finite State Machines . . . 52
7.3 Boolean Logic . . . 55
7.4 Boolean Logic in Circuits . . . 55
7.5 Binary Representation . . . 56
7.6 Addition and Subtraction . . . 56
7.7 Two's Complement . . . 59

8 Transistors and Logic Gates **62**
8.1 MOSFETS . . . 63
8.2 nMOS . . . 63
8.3 pMOS . . . 65
8.4 Logic Gates . . . 67
8.5 Constructing Logic Gates . . . 68

9 Computers **71**
9.1 Arduino Basics . . . 71
9.2 Arduino Input/Output . . . 71
9.3 Arduino Programming . . . 72
9.4 Arduino Tips . . . 72
9.5 Codes . . . 72
9.6 Time Division Multiplexing . . . 72
9.7 Brightness . . . 72

10 The Fourier Transform **73**
10.1 The Big Idea . . . 73
10.2 Sum of Functions . . . 74
10.3 Animations . . . 74
10.4 The Fourier Series . . . 74
10.5 Square Wave . . . 75
10.6 Music . . . 75

11 Devices: Capacitors and Inductors **77**
11.1 Capacitors . . . 78
11.2 Inductors . . . 78
11.3 Capacitors in Parallel and in Series . . . 78
11.4 Inductors in Parallel and in Series . . . 79
11.5 RC Circuits . . . 80
11.6 RL Circuits . . . 81

12 Impedance and Filters **83**
12.1 Impedance . . . 83
12.2 Filters . . . 84
12.3 Bode Plots . . . 85

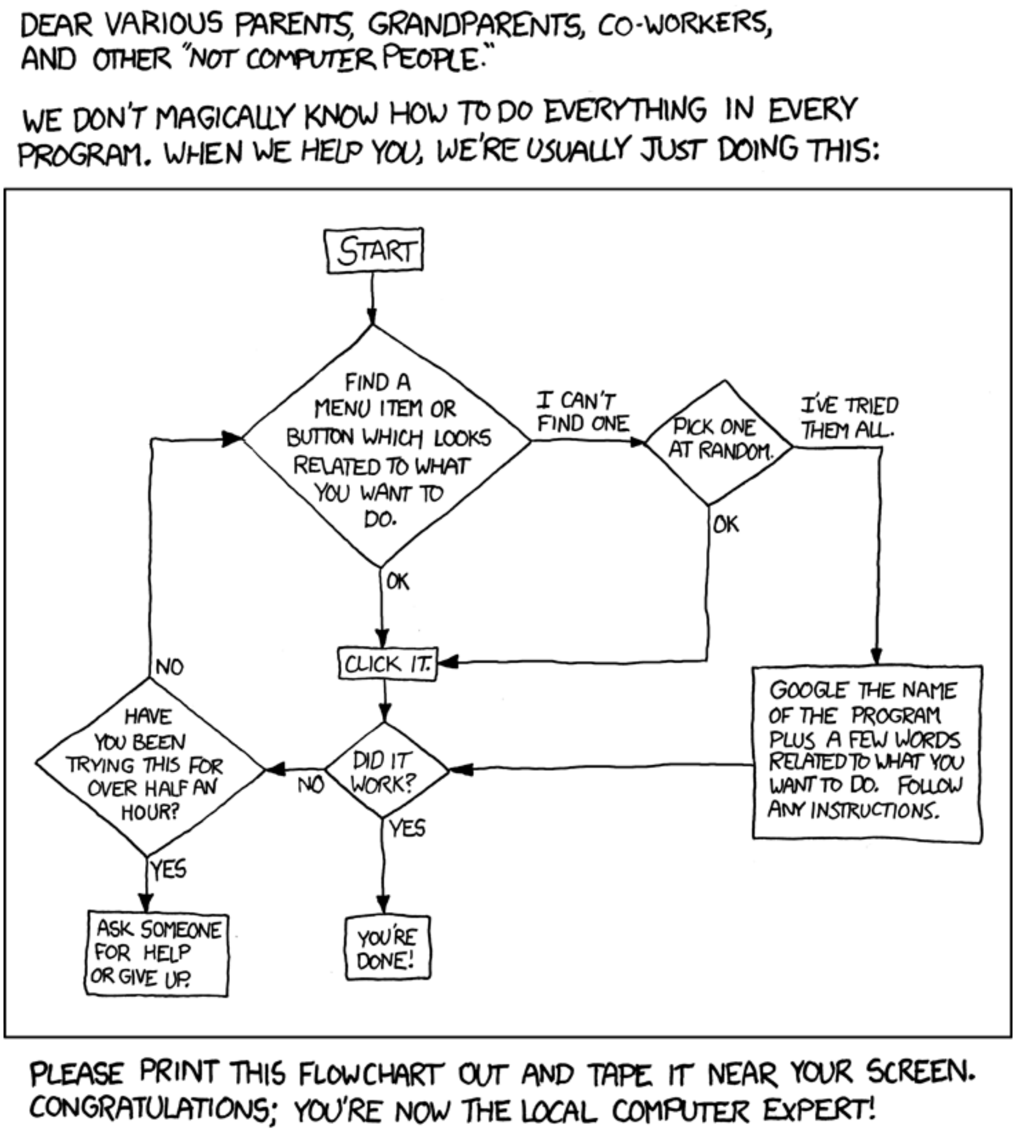
13 Operational Amplifiers **86**
13.1 Operational Amplifiers . . . 86
13.2 Golden Rules . . . 86
13.3 Non-inverting Amplifiers, Inverting Amplifiers, etc. . . . 87
13.4 Instrumentation Amplifiers . . . 87

14 More Filters **88**
14.1 Why Op-Amp Filters? . . . 89
14.2 Understanding Op-Amp Filters . . . 89

15 Power Converters **90**
15.1 Buck Converters . . . 90
15.2 Boost Converters . . . 90

Chapter 0

Quick Reference



In case you need a road map.

0.1 Pre-Midterm Equations

This is a short list of equations that are relevant to the material covered in class before the midterm.

$V = IR_{eq}$	$P = IV = I^2 R_{eq} = \frac{dE}{dt}$
Resistors in Series: $R_{eq} = R_1 + R_2$	Voltage Divider: $V_{out} = V_{in}(\frac{R_{out}}{R_{out}+R_1})$
Resistors in Parallel: $\frac{1}{R_{eq}} = \frac{1}{R_1} + \frac{1}{R_2}$	Current Divider: $I_{out} = I_{in}(\frac{R_1}{R_1+R_{out}})$

0.2 Pre-Midterm Key Concepts

This is just a summary of concepts that have been emphasized in this class prior to the midterm. Please be advised that this is not a comprehensive list and should not be treated as such. Anything covered in lecture or section can appear on the midterm or homework.

Labs Practices: Some concepts that have been emphasized during lab.

Solar Charger: Using a DMM, Soldering, Voltage Sources, Current Sources, Solar Panels, Diodes, Circuit Layout

Useless Box: Switches, Breadboarding, Motors, Transistors, Microcontrollers (Arduino), Circuit Design, Prototyping, Programming

Circuit Analysis and Components: Components and methods of analyzing circuits.

Voltage	Switches	Nodal Analysis
Current	IV Plots	Thevenin Circuit
Power	KVL	Norton Circuit
Energy	KCL	Relationship Between Thevenin and Norton Circuits
Resistors	Elements in Series	Finite State Machines
Diodes	Elements in Parallel	Boolean Logic
Solar Cells	Voltage Dividers	Transistor Gates
Transistors	Current Dividers	Two's Compliment
Pull Up Resistors	Superposition	Multiplexing

0.3 Post-Midterm Equations

This is just a summary of concepts that have been emphasized in this class after the midterm. Please be advised that this is not a comprehensive list and should not be treated as such. Anything covered in lecture or section can appear on the final or homework.

Capacitor Charge: $q = CV$

Capacitor Current: $i = C\frac{dV}{dt}$

Capacitor Power: $P_C = v(t)C\frac{dV}{dt}$
or $P_C = \frac{CV_{final}^2}{2}$, where $V_{initial} = 0$

Capacitor Energy: $E_C = \frac{1}{2}CV^2$

Capacitor Impedance: $Z_C = \frac{1}{2\pi fC}$

Circuit Elements in Parallel: $Z_{eq} = \frac{1}{Z_1} + \frac{1}{Z_2}$

Circuit Elements in Series: $Z_{eq} = Z_1 + Z_2$

Tones to Voltages: $v(t) = a_0 + \sum_{n=1}^{\infty}(a_n cos\frac{2n\pi t}{T} + b_n sin\frac{2n\pi t}{T})$

dB $= 10log(\frac{Power_{out}}{Power_{in}})$

Op Amp Equation: $v_{out} = A(v_p - v_n)$

Fundamental Frequency: $F = \frac{1}{T}$ where T is the period

Magnetic Flux Through Inductor: $\phi = iL$

Inductor Voltage: $V = \frac{d\phi}{dt} = L\frac{di}{dt}$

Inductor Power: $P_L = iL\frac{di}{dt}$
or $P_L = \frac{1}{2}Li_{final}^2$ where $i_{initial} = 0$

Inductor Energy: $E_L = \frac{1}{2}Li^2$

Inductor Impedance: $Z_L = 2\pi fL$

Current Divider: $I_{out} = I_{in}(\frac{Z_1}{Z_1+Z_{out}})$

Voltage Divider: $V_{out} = V_{in}(\frac{Z_{out}}{Z_{out}+Z_1})$

Voltages to Tones: $\int_0^T(v(t)cos\frac{2n\pi t}{T} = \frac{T}{2}a_n)dt$

Transfer Function: $H(F) = \frac{V_{out}(F)}{V_{in}(F)}$

Op Amp Gain: $G = \frac{v_{out}}{v_{source}}$

DC Component: $DC(averageV) = \frac{t_{high}}{T}$

0.4 Post-Midterm Key Concepts

This is just a summary of concepts that have been emphasized in this class after the midterm. Please be advised that this is not a comprehensive list and should not be treated as such. Anything covered in lecture or section can appear on the final or homework.

Labs Practices: Some concepts that have been emphasized during lab.

- **LED Cube:** Light-Emitting Diodes, Capacitors, Lab Tools (Power Supply, Signal Generator, Oscilloscope), Soldering, Breadboarding, Multiplexing
- **ECG:** Filters (1st and 2nd order), Operational Amplifiers, Inductors, Instrumentation Amplifiers, Debugging

Circuit Analysis: Components, general types of circuits and methods of analyzing circuits.

- Capacitors
- Relationship between Transistor Gates and Capacitors
- Operational Amplifiers
- Ideal Operation Amplifier Model
- Inverting Amplifier
- Non-Inverting Amplifier
- Summing Amplifier
- Current-to-Voltage Converter
- Differential Amplifier
- Instrumentation Amplifier
- Inductors
- RC Circuits
- RL Circuits
- RCL Circuits (ie. Boost and Buck Converters)
- Filters (1st and 2nd order)
- Bode Plots
- dB
- Corner Frequency
- Transfer Functions
- Sound
- Frequency Spectrum
- Fourier Series
- Sampling a Signal
- Fundamental Frequency
- Buck Converter
- Boost Converter

Chapter 1

An Introduction to Making

Making is breaking.

1.1 About This Guide

This study guide is intended to be a student-centric guide to understanding the concepts and developing the problem solving skills necessary for ENGR 40M. It is not meant to be a comprehensive text. Note that **this guide will likely be updated several times throughout the quarter**, so check the course website (engr40m.stanford.edu) often for updates. **Note: Many things in this reference manual, like the link here, and the entire table of contents are clickable! If you see anything that's referencing anything else, try clicking it!** Please let me know (rmu@stanford.edu) if you find any errors, typos, etc. (no matter how small, because they're definitely in here) or have any suggestions for improving this guide. Special thanks to Steven Bell and Mark Horowitz for their assistance and support in compiling this guide, and Sarah Van Sickle for creating the quick reference. Additionally, thanks to Henry Magun, Wendi Liu, Grace Young, Nathan Staffa, and Armando Herrera for submitting corrections.

1.2 Course Goals

Modern technology is a mix of electronics, computation, and mechanicals.

1. Leverage today's technology to see how YOU can create the future (or at least some cool projects).
2. Make things that combine electrical engineering, computer science, and mechanical engineering.
3. Develop an analytical understanding of simple analog and digital circuits.
4. Gain practical skills for constructing, programming, and debugging electronic devices.

1.3 Class Projects

Solar USB charger An introduction to the world of circuits.

Basic components: Voltage sources, current sources, and resistors.

Advanced components: Diodes and solar panels.

Making skills: Circuit layout and basic soldering.

Useless box An introduction to circuit design and combining software with hardware.

Basic components: Switches, motors, and breadboards.

Advanced components: Transistors and microcontrollers (Arduino).

Making skills: Circuit design, mechanical prototyping, and programming.

LED cube A demonstration of the possibilities unlocked by making skills and creativity.

Basic components: Light-emitting diodes.

Advanced components: Capacitors.

Making skills: Using laboratory tools (power supply, signal generator, and oscilloscope), efficient soldering, breadboard prototyping, and multiplexing.

Electrocardiogram An exercise in analog circuits, filters, and circuit debugging.

Basic components: Operational amplifiers.

Advanced components: Inductors, instrumentation amplifiers.

Making skills: Circuit debugging.

1.4 Problem Solving

Start with the information you know and see how far you can run with it.

- Work as far as you can, and find more information.
- Work in multiple directions at the same time.
- Learn to connect what you know.

1.5 Course Work

Assignments due every week:

- Homework assignment (Due Friday, before lecture).
- Prelab report (Due 24 hours before your assigned lab section, regardless of when you attend lab).
- Lab report (Due 1 week after the beginning of your assigned lab section).

1.6 Office Hours

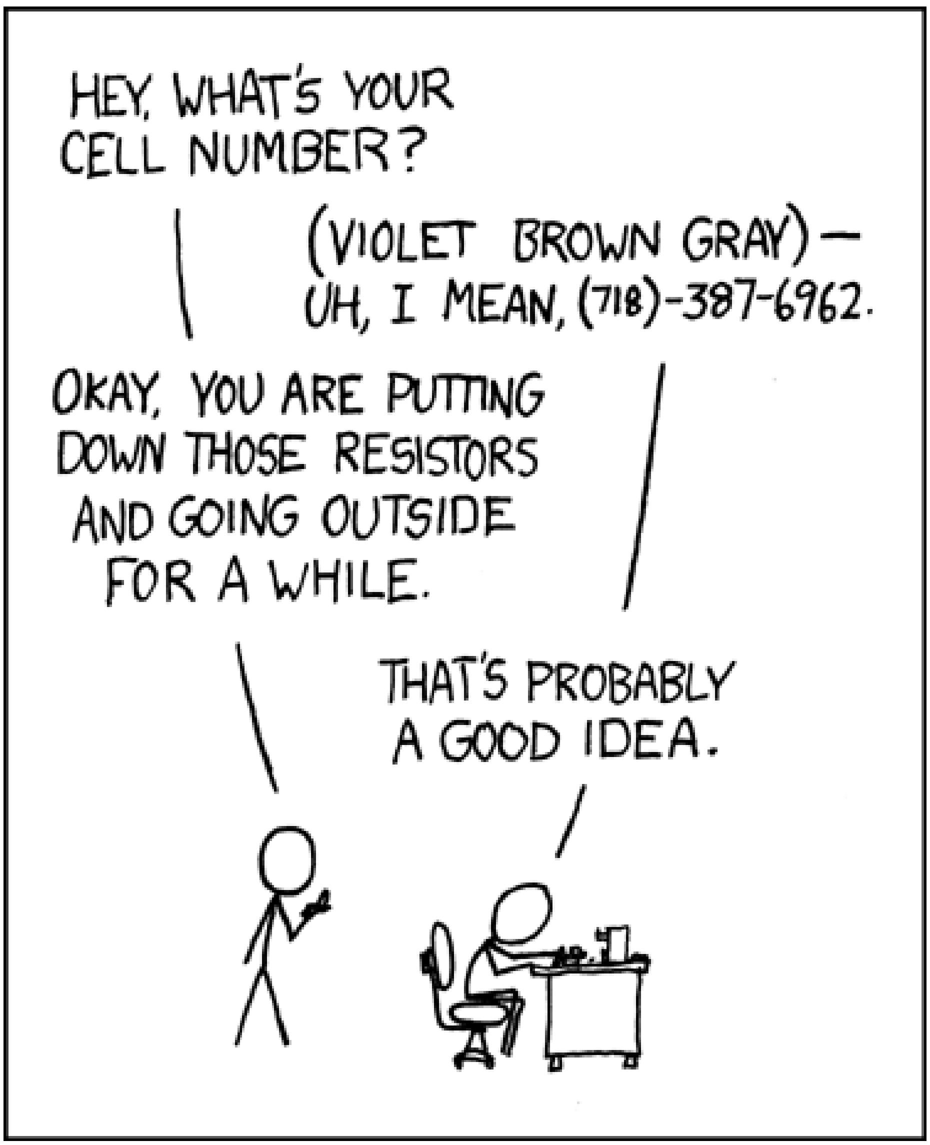
You are welcome to come to the lab whenever it is open to finish up projects, space permitting. Office hours are a great resource for getting help on a project or developing concept mastery.

1.7 Exams

The midterm will be in class on week 6. The final exam will be on Monday, June 6th, from 3:30 PM to 6:30 PM.

Chapter 2

Lab Reference



Long nights in the basement of Packard.

2.1 Prelab

Prelabs are important for your success in the labs – understanding the prelab will save you hours in lab! Come to office hours if you have questions. Note that they are due 24 hours before your assigned lab section, regardless of when you actually attend lab. This is so your TA is able to review your prelab and provide feedback before lab.

2.2 Safety

1. It is unlikely that you'll be electrocuted by anything other than the wall socket. Probably.
2. The following are bad: things getting hot, things sparking, things on fire, and things exploding. If something does get hot, disconnect the power immediately. If any of the other things happen, godspeed.
3. A hot soldering iron looks exactly like a cold soldering iron. Don't touch the end of a soldering iron.
4. Safety glasses (or some other form of eye protection) are highly recommended, but we can't exactly force you to use them. However, we don't exactly believe that metal in your eye is fun either.
5. Use the bench fans to avoid inhaling solder flux. While solder flux may smell like tree sap, we can't guarantee that it's doing any favors for your lungs.

2.3 Cleanup

1. Turn off your soldering iron, fan, oscilloscope, power supply and overhead light.
2. Return wires to the correct place on the wall wire rack.
3. Return reusable parts (spare plastic, unused components, >2" long pieces of solder, etc.)
4. Place scrap solder in solder waste cans (solder needs to disposed of separately due to its toxic composition, most notably lead).
5. Place all other trash in trash can.
6. Wash your hands with soap and hot water! (Delicious lead poisoning not included.)

Solder waste cans (the small plastic jars) are for solder only! (This means scrap wire doesn't go in the solder waste cans.)

2.4 Cable Colors

Red is power/positive, black is ground/negative. Try to be consistent when building your own projects.

2.5 Digital Multimeter

The digital multimeter is a tool for measuring current, voltage, and resistance.

Setup Black wire goes into the port labeled 'COM'. Red wire goes into the port labeled 'VΩmA'.

- Note: when measuring a large current (more than 200 mA), the red wire goes into the port labeled '10ADC' instead of the port labeled 'VΩmA'.

Measuring Voltage Put the voltage meter in parallel with the component that you want to measure.

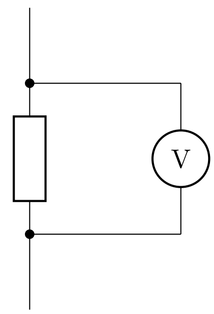


Figure 2.1: Measuring voltage

We don't put a voltage meter in series with the component that we want to measure because the voltage drop occurs across the component, and thus, the voltage meter has its own voltage drop, rather than measuring the voltage drop of the component. (Also, recall that by KVL, components in parallel have the same voltage.)

Ideally, a voltage meter would take no current (and thus, no power) from the circuit, but in reality, voltage meters take a small amount of current (which you will explore in lab).

Measuring Current Put the ammeter in series with the component that you want to measure.

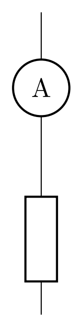


Figure 2.2: Measuring current

We don't put a current meter (ammeter) in parallel with the component that we want to measure because it acts as an alternate path for current to flow in the circuit, resulting in current flowing into the ammeter instead of the component (and thus, inaccurate measurements).

Ideally, a current meter would have no voltage drop (and thus, dissipate no power) from the circuit, but in reality, current meters have a small voltage drop (which you will explore in lab).

2.6 Resistors

Read resistors from **opposite** the side of the gold/silver bar. If there is no gold/silver bar, read from the side with a bar **closer to the end** of the resistor.

Color	First Digit	Second Digit	Multiplier	Tolerance
Black	0	0	$10^0 = 1$	
Brown	1	1	$10^1 = 10$	$\pm1\%$
Red	2	2	$10^2 = 100$	$\pm2\%$
Orange	3	3	$10^3 = 1,000$	
Yellow	4	4	$10^4 = 10,000$	
Green	5	5	$10^5 = 100,000$	
Blue	6	6	$10^6 = 1,000,000$	
Violet	7	7	$10^7 = 10,000,000$	
Gray	8	8	$10^8 = 100,000,000$	
White	9	9	$10^9 = 1,000,000,000$	
Gold			$10^{-1} = 0.1$	$\pm5\%$
Silver			$10^{-2} = 0.01$	$\pm10\%$

Resistor chart

For example, a resistor with bars brown, black, red, gold is a $10 \times 10^2 = 1000\Omega$ resistor with $\pm5\%$ tolerance.

2.7 Diodes

The end of the diode with the white band is the cathode/negative end (this matches the flat bar of a diode in a circuit diagram).

Most diodes have a forward voltage of 0.6 V; our Schottky diodes have a forward voltage of 0.3 V.

Light emitting diodes (LEDs) are a special type of diode which emit light when current passes through them. Red LEDs generally have a smaller forward voltage than green and blue LEDs.

2.8 Soldering

Soldering is a way of permanently attaching wires to terminals. You'll be doing a lot of soldering over the course of the quarter.

1. Turn the soldering iron on and allow the iron to heat up. (The green light will turn on while the soldering iron is heating up, and will blink when the soldering iron has reached the set temperature.)
2. Coat the tip of your soldering iron with solder before soldering. (This is also called tinning.)
3. Use the soldering iron to heat the components to be soldered.
4. Feed solder between the components and the soldering iron.
    - Keep in mind that the solder will melt towards the heat source.
    - Use the minimum amount of solder that will allow for a solid connection. (You need less solder than you think.)
5. Once an adequate amount of solder has been applied to the solder joint, remove the soldering iron and allow the joint to cool.
6. Clean the tip of your soldering iron on either a damp sponge (use the spray bottle at your lab station to dampen the sponge) or a brass tip cleaner.
7. Test the solder joint by pulling gently on the ends of the components. The joint should stay intact.

Solder contains a material called flux that burns away as you heat the solder. This is the "smoke" that emerges from the soldering iron. Flux allows for solder to bind to components and helps prevent metal oxidation (which lowers electrical conductivity). **You want a constant stream of flux when soldering.**

Note that solder is meant to be an electrical connection; do not rely on it for mechanical strength.

2.9 Breadboarding

Breadboards are flat blocks that reduce the amount of wiring and soldering necessary to build circuits.

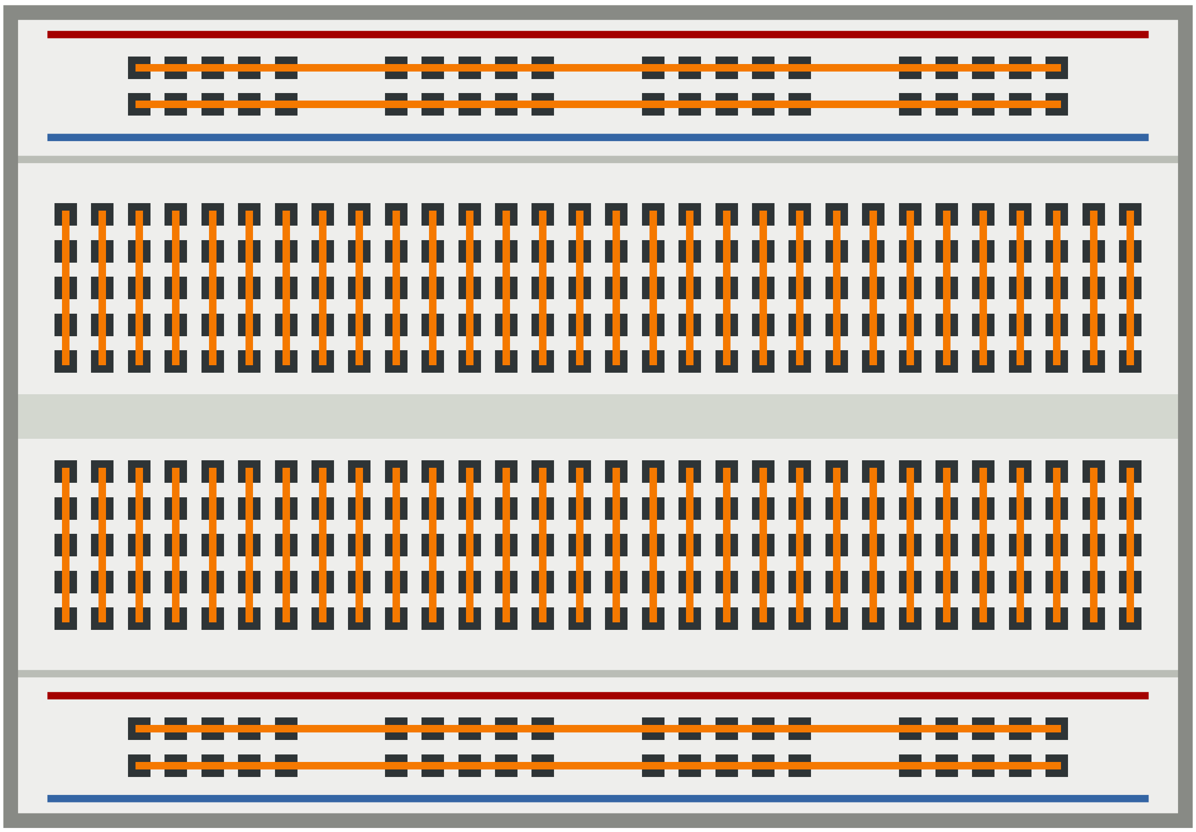


Figure 2.3: Breadboard. Orange lines indicate connected ports.

This means that you should never connect both ends of a component to the same line.
Note: sometimes the rails (the long lines at the top and bottom of the breadboard) are connected all the way across the breadboard, other times they are only connected halfway across the breadboard. Your breadboard should have markings to help you determine which is the case.

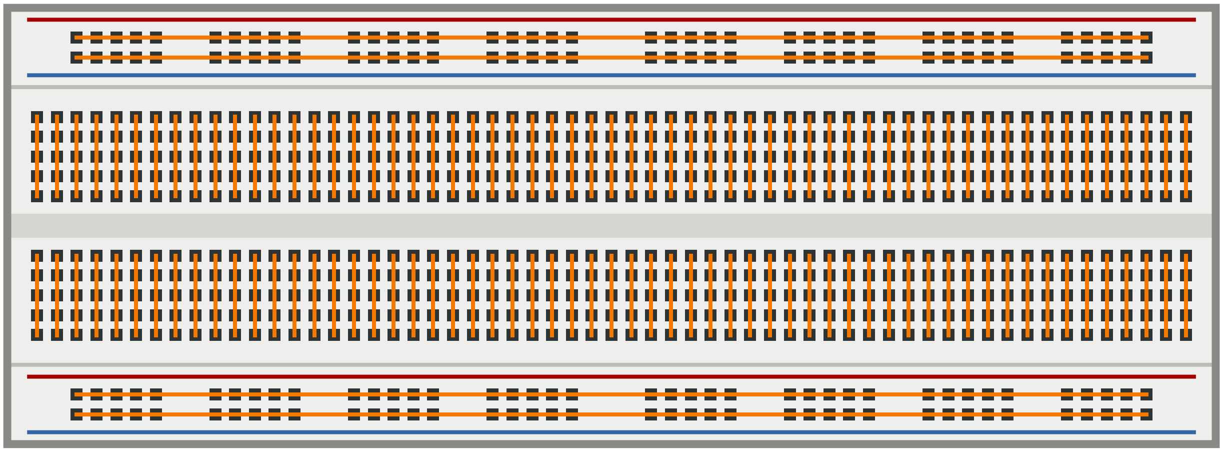


Figure 2.4: Breadboard with full rails.

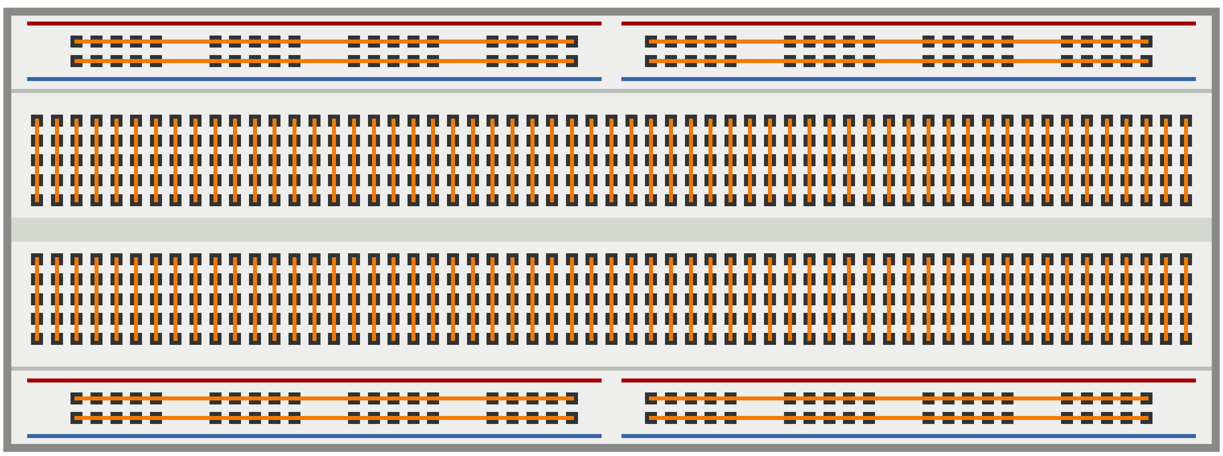


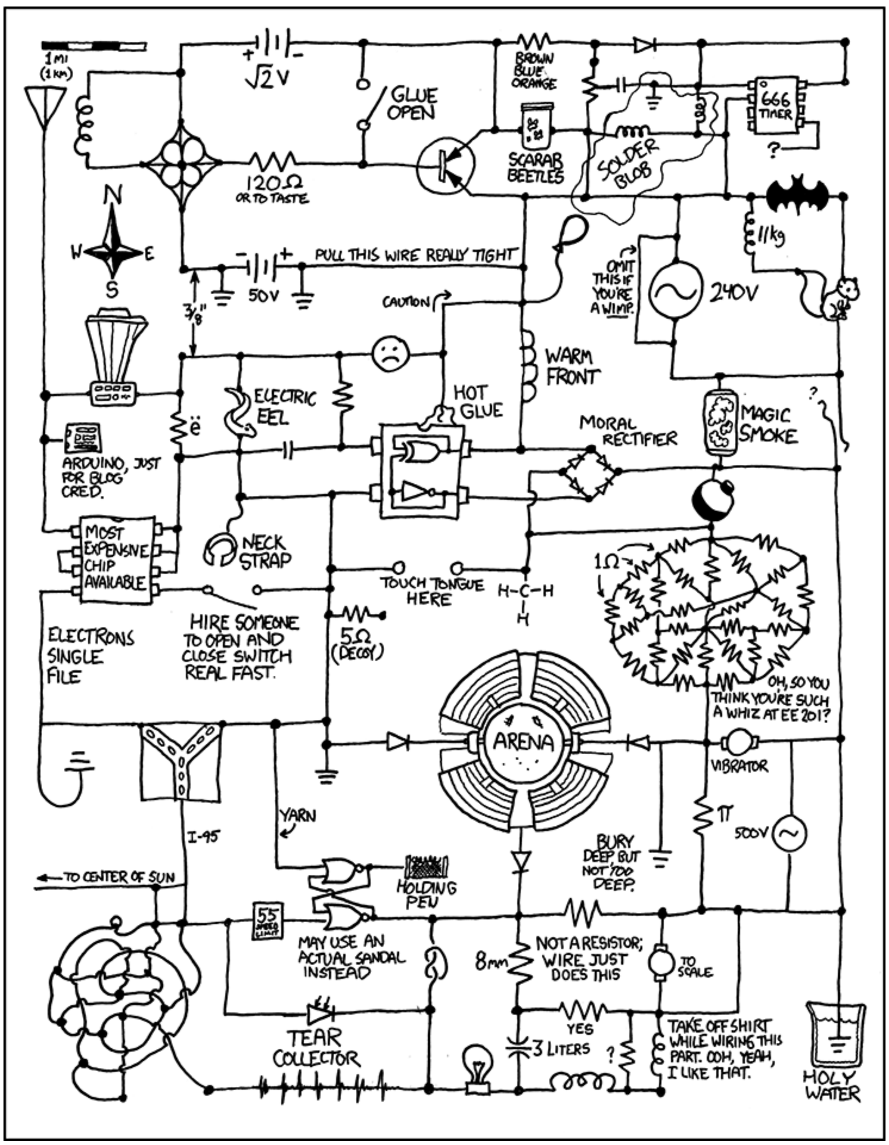
Figure 2.5: Breadboard with half rails. Note the different red and blue rail markings.

Breadboarding tips:

1. Avoid using jumper cables whenever possible.
2. Use the shortest wires possible when connecting components.
3. Cut components with long wires/leads to an appropriate length (this applies especially to resistors — you don't want components sticking out above your breadboard.)
4. Use a capacitor across the power rails to reduce noise from the power supply.
5. Integrated circuits (chips) are not necessarily the same even if they are in the same package! Read the markings on the IC to figure out what it is.

Chapter 3

Circuit Basics



I have no idea what I'm doing.

3.1 Terminology

Active Active devices are devices which provide net energy to the circuit over time. They have a negative power dissipated.

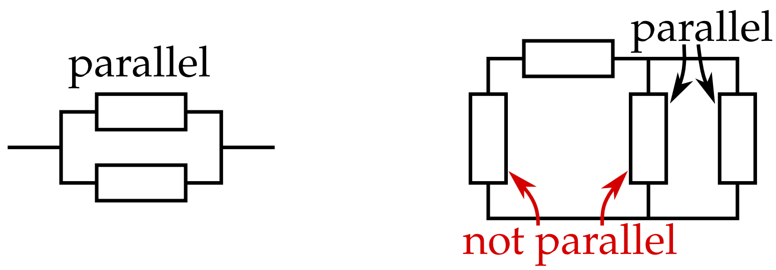
Closed Circuit An closed circuit refers to a circuit that is connected.

Energy Energy is the integral of power over time ($E = \int P \, dt$). For constant power, $E = P \cdot t$.

Ground Ground refers to the point in a circuit that all voltages are measured relative to. It is always at 0 volts, since the voltage of the ground node is measured relative to itself.

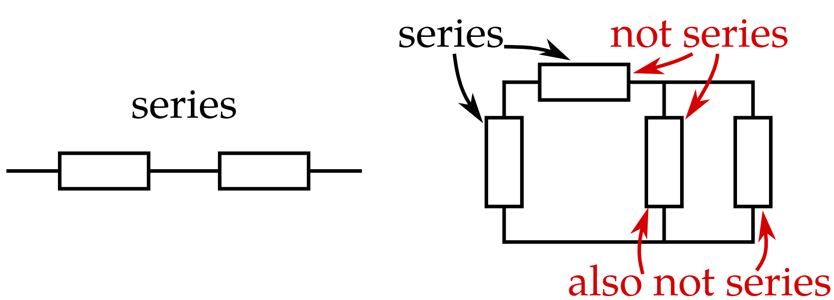
Open Circuit An open circuit refers to a circuit that is disconnected.

Parallel Parallel refers to two components where the two terminals of the first component are connected (not necessarily exclusively) to different terminals of the second component. (Components in parallel have the same voltage.)



Passive Passive devices are devices which consume net energy from the circuit over time. They have a positive power dissipated.

Series Series refers to two components where one of the terminals of the first component is connected **exclusively** to one of the terminals of the second component. (Components in series have the same current.)



Short A short refers to a wire. The name is derived from the term "short circuit."

3.2 Charge

Charge is the basis of all electrical systems.

1. Electrons are (negative) charge carriers.
2. Charge is measured in coulombs (C).
3. An electron has a charge of -1.6×10^{-19} coulombs.

3.3 Current

Current is moving charge.

1. Current is measured in amperes (A), which is coulombs per second. (This is charge per unit of time.)
2. 1 amp(ere) is 1 coulomb per second. ($1A = 1\ {}^{C}\!/_{s}$)

You can reverse the direction of a current if you flip the sign on the current label. This is because saying that you have some amount of current flowing in one direction and saying that you have negative some amount of current flowing in the opposite direction mean the same thing.

Change the direction, change the sign.

3.4 Kirchoff's Current Law

Kirchoff's current law (KCL) states that all bodies are charge neutral. Thus, current in is equal to current out.

1. Current along a wire is the same.

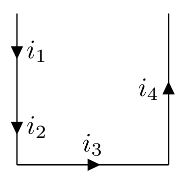


Figure 3.1: $i_1 = i_2 = i_3 = i_4$

2. Current into a device is equal to current out a device.

Figure 3.2: $i_1 = i_2$

3. The net current into a node (or device) is zero.

Figure 3.3: $i_1 + i_2 = 0$

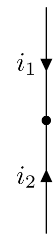


Figure 3.4: $i_1 + i_2 = 0$

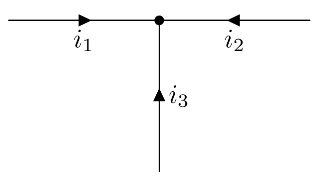


Figure 3.5: $i_1 + i_2 + i_3 = 0$

4. A useful result of KCL is that components in series have the same current.

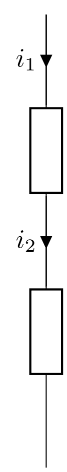


Figure 3.6: $i_1 = i_2$

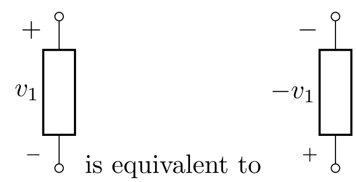
3.5 Voltage

Voltage is electric potential. It is the electrical analogue of the potential energy that results from gravity; voltage causes charge to move.

1. Voltage is measured in volts (V), which is joules per coulomb. (This is energy per unit of charge.)
2. 1 volt is 1 joule per coulomb. ($1V = 1\ ^{J}/_{C}$)
3. Voltage is relative; it is only defined between two points.

When voltages are labeled in a circuit diagram, they are accompanied by polarity labels. In combination, these signs are read as: "The '+' terminal is (voltage label) volts higher than the '-' terminal.

You can reverse the polarity labels of a component if you flip the sign on the voltage label. (Consider why.)

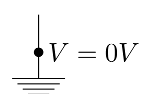


Remember that voltages are relative. **Polarity labels without a voltage label are meaningless!**

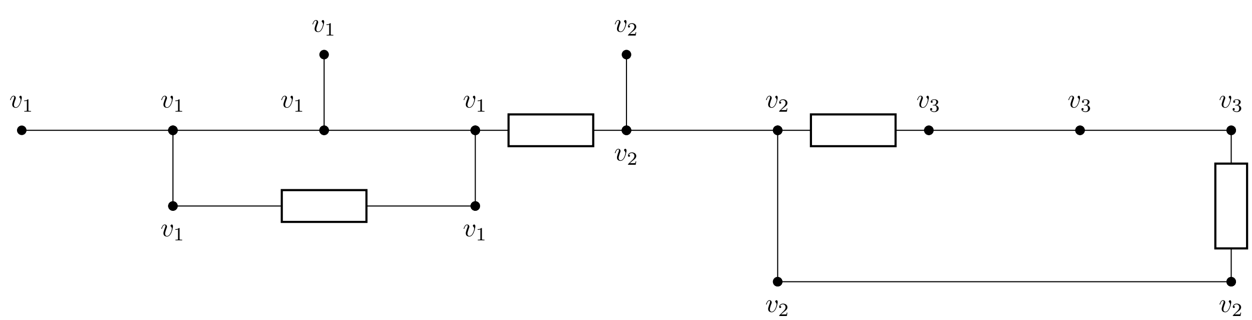
3.6 Kirchoff's Voltage Law

Kirchoff's voltage law (KVL) states that potential energy is consistent. Thus, the sum of component voltages around any loop in a circuit is zero.

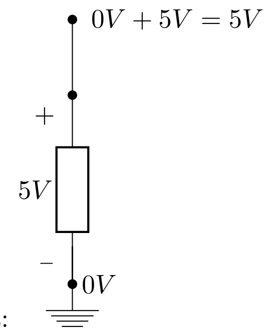
1. We define ground to be the reference that all voltages are compared to. Ground is always at zero volts because the voltage at ground is measured relative to itself.

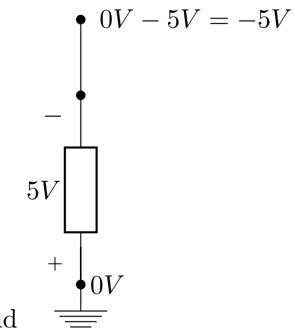


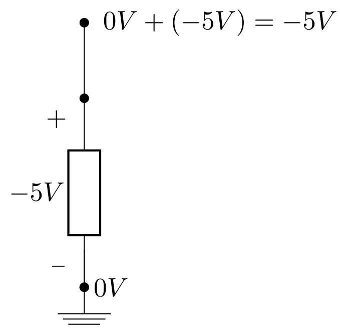
2. All voltages along a single wire (all points reachable by a path with no components) are the same.

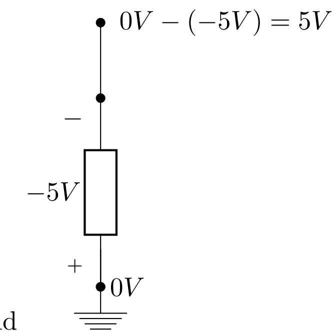


3. To calculate voltage at a node across a component:
   (a) When going from the **negative** polarity label **to** the **positive** polarity label, we **add** the voltage on the voltage label.
   (b) When going from the **positive** polarity label **to** the **negative** polarity label we **subtract** the voltage on the voltage label.

Components with positive voltages: 

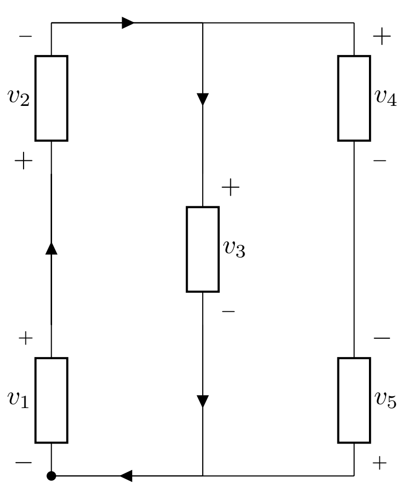
and 

Similarly, components with negative voltages: 

and 

4. Voltage is path independent. This means that if we pick a starting point and follow **any path** around a circuit, adding and subtracting component voltages as we go around, we should end up at the same voltage when we return to our starting point.

   tldr: The sum of component voltages around a loop is zero. This is useful for finding unknown voltages.



Using KVL and starting from the node in the bottom left corner and following the loop indicated by the arrows, we see that $v_1 - v_2 - v_3 = 0$.

We can use this to solve for the voltages of components in the loop. If we know the values of v_1 and v_2, we can solve for v_3 using $v_3 = v_1 - v_2$.

Alternatively, we could start from the same node in the bottom left corner and follow the loop clockwise, but this time through the path on the far right, resulting in $v_1 - v_2 - v_4 + v_5 = 0$.

We can use the combination of these two equations to solve for the voltages of two unknown components in the circuit. If we know the values of v_1, v_3, and v_5, we can solve for v_2 and v_4 by using $v_2 = v_1 - v_3$ and $v_4 = v_1 - v_2 + v_5$.

5. A useful result of KVL is that components in parallel have the same voltage.

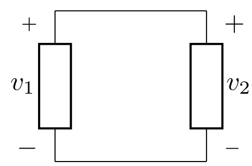


Figure 3.7: $v_1 - v_2 = 0$, therefore $v_1 = v_2$

3.7 Power

Power is energy per unit of time. (Think power ratings for light bulbs.)

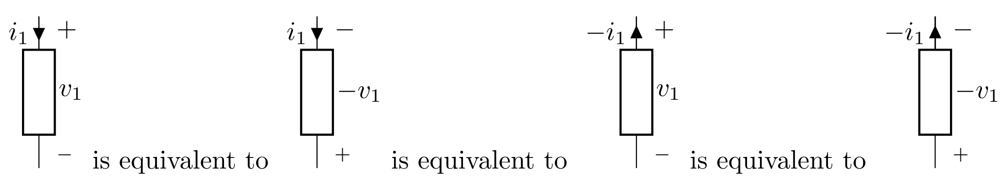
1. Power is measured in watts (W), which is joules per second. (This is energy per unit of time.)
2. 1 watt is 1 joule per second. ($1W = 1\ ^J/_s$)
3. 1 watt is 1 amp times 1 volt. ($1W = 1A \times 1V$)
   (This is because $1A = 1\ ^C/_s$ and $1V = 1\ ^J/_C$, so $1A \times 1V = \frac{C}{s} \times \frac{J}{C} = \frac{J}{s} = 1W$)
4. Power is equal to current times voltage. ($P = i \times V$)

3.8 Calculating Power

To calculate power, find the numerical values (absolute values) of the current and voltage of the device, and multiply them. Then to find the sign, find the terminal with the higher voltage, and find whether current is flowing into or out of that terminal. If current is flowing into that terminal, the device is dissipating energy, and the power dissipated is positive. If current if flowing out of that terminal, the device is providing energy, and the power dissipated is negative.

It also turns out that you can calculate power if you adjust the current and voltage so that current is flowing into the positive terminal of the device, then multiply the current and voltage (keeping the signs this time).

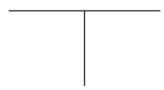
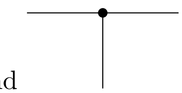
In order: normal, voltage flip, current flip, voltage and current flip.



The power dissipated by each of these components is $P = i_1 \times v_1$.

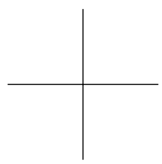
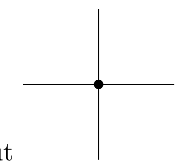
3.9 Circuit Solving Tips

1. Voltages at nodes are labeled relative to ground. Thus, ground is at 0 volts by definition.
2. KCL and KVL always hold. If you solve a circuit and find that they are not consistent, go back and check your work.
3. Three-way junctions are connected.

   Both  and  are **connected**.

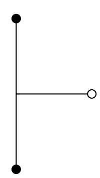
   The dot is optional – it is only for clarity.

4. Four-way junctions are not necessarily connected.

    is **not connected**, but  is **connected**.

   The dot is important – it indicates that the junction is connected.

5. An open circle **typically** indicates that a terminal is not connected to anything – no current flows through that terminal.



The right terminal, and therefore the entire right branch, has no current flowing through it.
Note that this does not apply to switches, which are also drawn with open circles.

Chapter 4

Devices: Resistors

It is left as an exercise to the reader to prove this result.

4.1 I-V Curves

I-V curves are a way of representing the relationship between current and voltage for circuit components. They show current flow as a function of voltage.

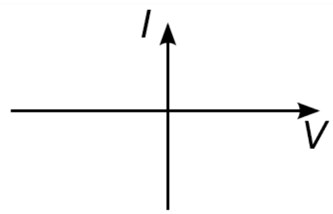


Figure 4.1: I-V curve axes

Voltage Sources Voltage sources supply or absorb a constant voltage regardless of current. An example is a battery.

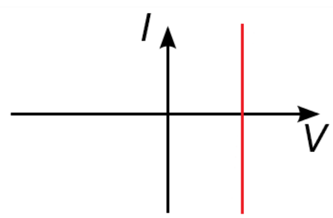


Figure 4.2: Ideal voltage source

Current Sources Current sources supply or absorb a constant current regardless of voltage. There are very few real-life current sources.

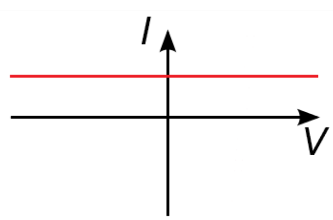


Figure 4.3: Ideal current source

4.2 Resistors

Resistors define a linear relationship between voltage and current.

Specifically, Ohm's law defines the linear relationship $\mathbf{V} = \mathbf{iR}$, where V is voltage, i is current, and R is resistance.

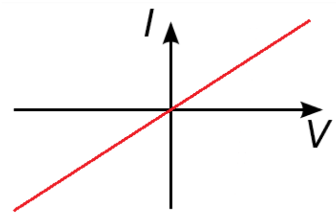
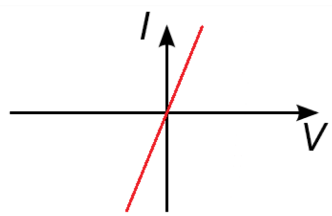
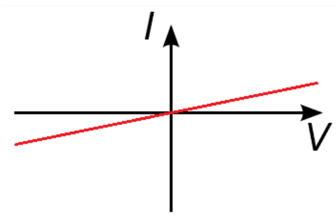


Figure 4.4: Resistor

Notice that the curve travels only through the first (top right) and third (bottom left) quadrants. In the first quadrant, voltage and current are both positive, so power dissipated is positive. In the third quadrant, voltage and current are both negative, so power dissipated is also positive. This means that a resistor is **always consuming power**.

Small resistance: 

Large resistance: 

Since $V = iR$ implies that $i = \frac{1}{R}V$, we have that the slope of the I-V curve ($\frac{1}{R}$) is inversely proportional to the resistance of the resistor (R) being represented.

The reason that we model resistance is because conductors are not perfect and consume some energy when carrying current. The worse the conductor, the greater the resistance to current flow, and the more energy consumed when carrying current. Components designed to have a particular resistance are called **resistors**.

4.3 Voltage Dividers

Multiple resistors can be arranged to form devices that perform various tasks. One of these devices is a voltage divider.

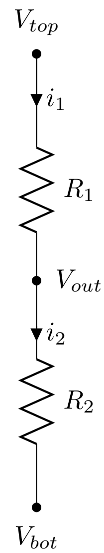


Figure 4.5: What is a voltage divider? What is V_{out}? Why did I take this class?

I know this looks complicated, so let's break it down using KCL and Ohm's law.

Recall that when calculating the power of devices, we want current flowing into the positive terminal of the device. Let's arbitrarily pick the upper end of each resistor to be the positive terminal.

Then, by our rules for calculating the voltage across a device (we take the voltage at the positive terminal and subtract the voltage at the negative terminal), we have that

1. the voltage across the upper resistor is $V_{top} - V_{out}$
2. the voltage across the lower resistor is $V_{out} - V_{bot}$

Similarly, by our rules for calculating the current into a device (we take the current flowing into the positive terminal), we have that

1. the current into the upper resistor is i_1
2. the current into the lower resistor is i_2

Applying KCL (see figure 3.2), we get that $i_1 = i_2$.
Thus, the current into either resistor is i_1.

From Ohm's law, we have $V = iR$. Thus,

1. Across the upper resistor, $V_{top} - V_{out} = i_1 \times R_1$
2. Across the lower resistor, $V_{out} - V_{bot} = i_1 \times R_2$

Adding the two equations together, we get $V_{top} - V_{out} + V_{out} - V_{bot} = i_1 \times R_1 + i_1 \times R_2$.
Simplifying, we get $(V_{top} - V_{bot}) = i_1 \times (R_1 + R_2)$

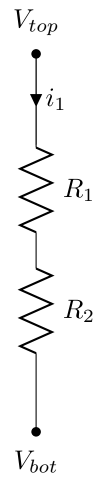


Figure 4.6: Same circuit as above with fewer things with words next to them

Hmm . . . this looks familiar, doesn't it? It looks like Ohm's law ($V = iR$) again!

But this time, there are **two resistors**. This means that **resistances in series add**.

$$R_{series} = R_1 + R_2 + \ldots + R_n$$

Intuitively, this makes sense because there are two conductors, each with some imperfections that consume energy when carrying current. Putting these conductors end to end forces all of the current to flow through both conductors, so each conductor consumes its individual share of the energy.

Returning to finding V_{out}, we observe that $(V_{top} - V_{bot}) = i_1 \times (R_1 + R_2)$ implies that $i_1 = \frac{V_{top}-V_{bot}}{R_1+R_2}$

From Ohm's law across the lower resistor, we have $V_{out} - V_{bot} = i_1 \times R_2$ which implies $V_{out} = i_1 R_2 + V_{bot}$.

Substituting for i_1, we get $V_{out} = \frac{V_{top}-V_{bot}}{R_1+R_2} R_2 + V_{bot}$.

Rearranging, we get $V_{out} = \frac{R_2}{R_1+R_2}(V_{top} - V_{bot}) + V_{bot}$.

Notice that if $V_{bot} = 0$ (the bottom of the circuit is connected to ground), then $V_{out} = \frac{R_2}{R_1+R_2} \times V_{top}$, that is, V_{out} is a fraction of V_{top} based on the ratio of R_1 and R_2. This is why this "component" is called a voltage divider – the resistance of the two resistors determines how the voltage is divided in the circuit.

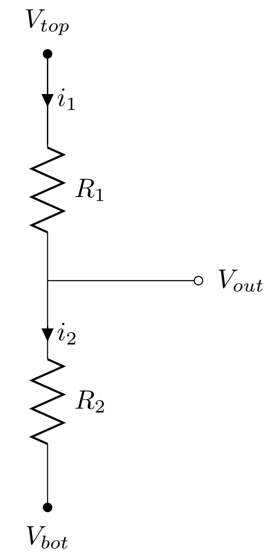


Figure 4.7: $V_{out} = \frac{R_2}{R_1+R_2}(V_{top} - V_{bot}) + V_{bot}$

Note that the open circle next to V_{out} indicates that there is no current in that terminal; hence, the entire right branch of the circuit has 0 Amps of current flowing through it, and this circuit is equivalent to the ones above.

4.4 Current Dividers

Another device that multiple resistors can be arranged to form is a current divider. I won't bore you with the same level of detail on current dividers. That being said . . .

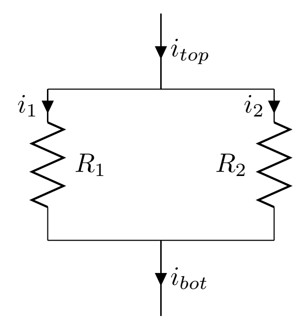


Figure 4.8: Current divider: $i_1R_1 = i_2R_2$

Showing how a current divider works using Ohm's law and KVL is not very tedious. Recall from Ohm's law that $V = iR$. Similarly, recall from KVL that the voltage across each resistor must be the same (see figure 3.7). Thus, we have that $i_1R_1 = i_2R_2$. This means that **the current flowing through a resistor in a current divider is inversely proportional to its resistance**. ("Path of least resistance.")

By KCL, we know that $i_{top} = i_1 + i_2 = i_{bot}$ (see figure 3.5). Then we can figure out the equivalent resistance of two resistors in parallel by using Ohm's law and KVL.

$i_{top}R_{equivalent} = i_1R_1 = i_2R_2$ and $i_{top} = i_1 + i_2$

Multiplying each by a factor of one, we get $\frac{R_{equivalent}}{R_{equivalent}} i_{top} = \frac{R_1}{R_1} i_1 + \frac{R_2}{R_2} i_2$.

Rearranging, this results in $\frac{i_{top} R_{equivalent}}{R_{equivalent}} = \frac{i_1 R_1}{R_1} + \frac{i_2 R_2}{R_2}$.

Dividing through using $i_{top} R_{equivalent} = i_1 R_1 = i_2 R_2$, we get that $\frac{1}{R_{equivalent}} = \frac{1}{R_1} + \frac{1}{R_2}$.

This means that **resistance in parallel is the reciprocal of the sum of the recipricols of the individual resistances**. (Confused yet?)

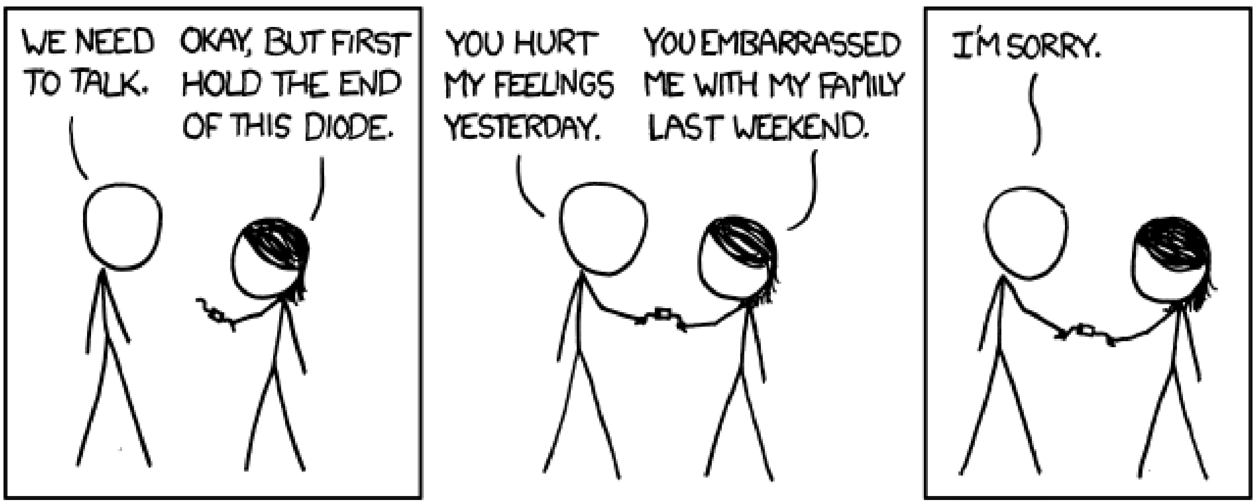
$$\frac{1}{R_{parallel}} = \frac{1}{R_1} + \frac{1}{R_2} + \ldots + \frac{1}{R_n}$$

This also results in the overall resistance of resistors in parallel to be less than any individual resistor in parallel. (Strange, right?)

Intuitively, this is because there are now multiple paths for current to take. We define conductance to be the reciprocal of resistance, so you are adding conductance when adding more paths for current to take, resulting in a lower resistance than any individual resistor.

Chapter 5

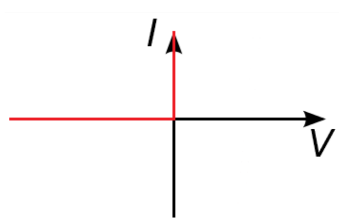
Devices: Diodes



Resistance is futile – diodes are better.

5.1 Ideal Diodes

Diodes are devices that allow current to flow in one direction, but not the other.



Diodes are drawn asymmetrically, with different positive and negative ends.

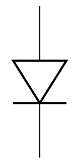


Figure 5.1: Ideal diode in a circuit

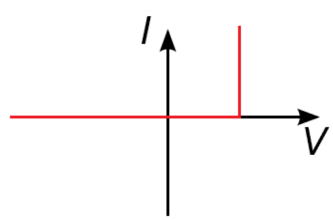
In this diagram, current can flow from top to bottom but not the other way.

To solve a circuit containing ideal diodes:

1. Assume that current through the diode is zero (i.e., that the diode is off, and acts like an open circuit). Find the voltage across the diode under this assumption.
2. If the diode voltage is negative, then the diode is in fact off, and our analysis holds.
3. If the diode voltage is positive, then the assumption was incorrect, and the diode is on. The diode voltage can't actually be positive, so re-solve the circuit assuming that the diode is on (i.e., it acts a short circuit and can be treated like a wire).

5.2 Idealized Diode

Idealized diodes are similar to ideal diodes, but have a threshold voltage (called the forward voltage) before they allow current to flow in one direction.



The voltage at which current starts flowing (the value on the x-axis at the "corner" of the curve) is known as the forward voltage (V_f).

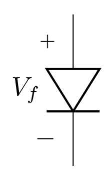


Figure 5.2: Idealized diode in a circuit

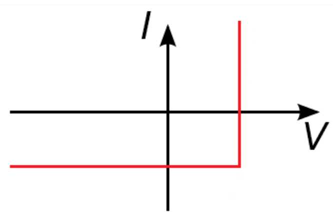
Ideal diodes are just a special case of idealized diodes where $V_f = 0V$.

To solve a circuit containing ideal diodes:

1. Assume that current through the diode is zero (i.e., that the diode is off, and acts like an open circuit). Find the voltage across the diode under this assumption.
2. If the diode voltage is less than V_f, then the diode is in fact off, and our analysis holds.
3. If the diode voltage is V_f or greater, then the assumption was incorrect, and the diode is on. The diode voltage is no greater than V_f, so re-solve the circuit assuming that the diode is on (i.e., it acts a voltage source with $V = V_f$).

5.3 Solar Cells

Idealized solar cells are similar to idealized diodes, but can supply power.



Recall that we describe negative power dissipated as supplying power to the circuit. In the fourth quadrant (bottom right), we have positive voltage and negative current, so power dissipated is negative. Thus, the solar cell is supplying power to the circuit when it is operating in the fourth quadrant.

Solar cells can be modeled in a circuit as a current source in parallel with an idealized diode.

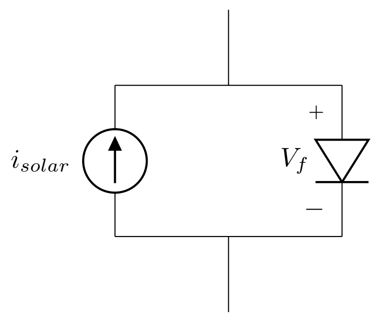


Figure 5.3: Solar cell model

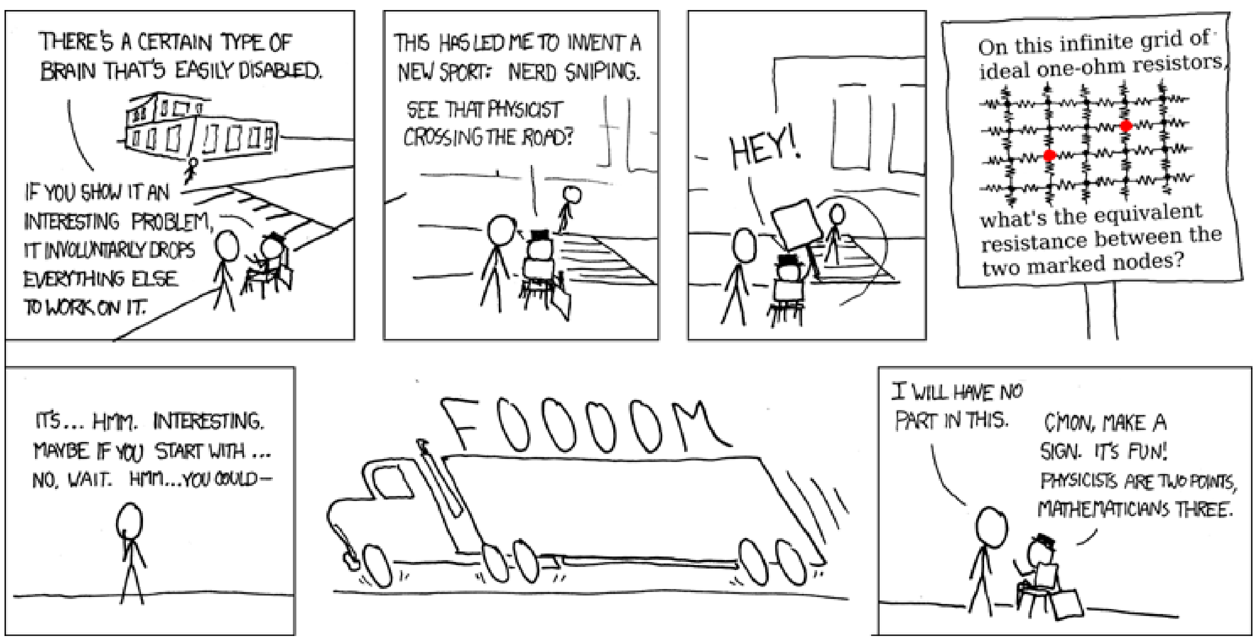
When light shines on the diode, it generates a current in the solar cell. The amount of current is proportional to the intensity of the light.

We can characterize the power output of a solar cell by measuring the open circuit voltage (where the red line crosses the voltage axis) and the short circuit current (where the red line crosses the current axis). Since $P = iV$, for an idealized solar cell, the power output is the product of the open circuit voltage and short circuit current, or the area of the rectangle in the fourth quadrant of the graph.

Note that in a real solar cell, the corner is not perfectly square, so real power output is less than the product of the open circuit voltage and closed circuit current.

Chapter 6

Solving Circuits



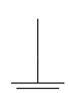
Life just got a lot harder.

6.1 Nodal Analysis

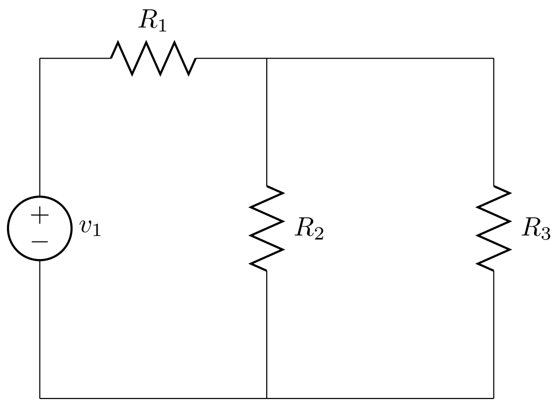
Nodal analysis is the process of choosing "interesting" nodes and applying KCL and KVL to them.

Note that device voltages are just the difference of the voltages of the nodes they are connected to, so nodal analysis can be used to solve for device voltages as well.

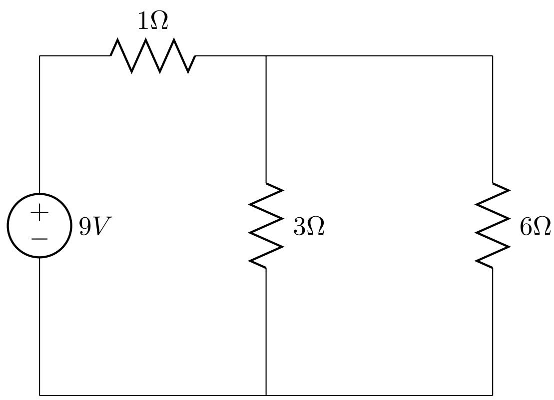
Recall that voltages are relative, and are only defined between two points. Thus, we need to pick a node as a reference, and call it ground (0 V). This node is sometimes indicated on circuit diagrams, and looks like this:



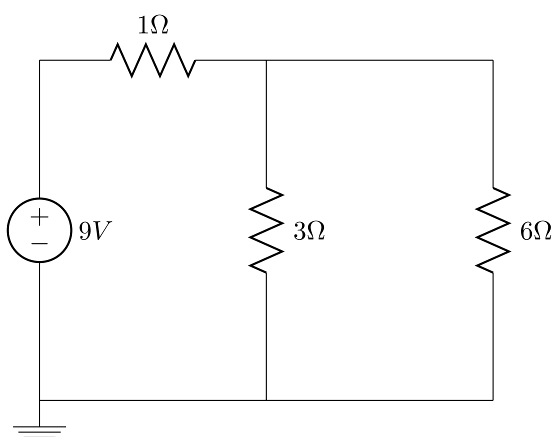
Nodal analysis is best explained with an example.



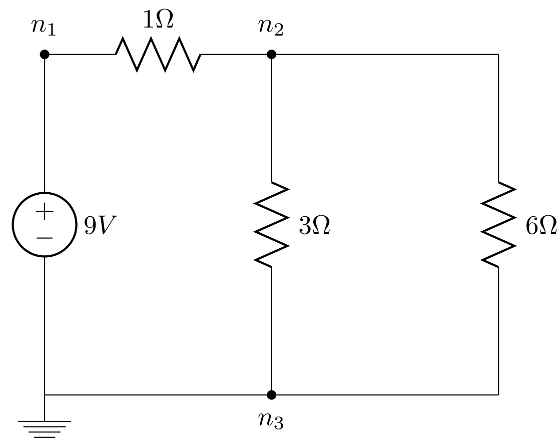
To make things more concrete, let's choose values for v_1, R_1, R_2, R_3.



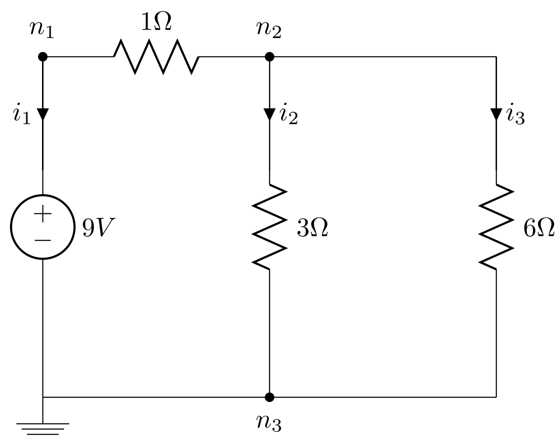
Alright, now let's pick a reference point (ground), since none are given in the circuit. Generally, it's a good idea to put ground next to a voltage source, so you know the voltages on both sides of the source.



Now, let's pick some nodes to analyze. Generally, good nodes to pick are nodes between two components and T-junctions.



Then, we need to define currents so we can write KCL equations for every node.



n_1 The current around node n_1 isn't very interesting, it is just i_1.

n_2 The current around node n_2 can be described by $(-i_1) + (-i_2) + (-i_3) = 0$.

n_3 The current around node n_3 can be described by $i_1 + i_2 + i_3 = 0$.

Hmm, not very fruitful is it? The current equations for n_2 and n_3 are the same, so we ended up with one equation, $i_1 + i_2 + i_3 = 0$.

Well then, let's look at the current flowing through each device in terms of the node voltages. Let v_{n_1} be the voltage at n_1, v_{n_2} be the voltage at n_2, and v_{n_3} be the voltage at n_3. There are four devices: (Pay close attention to signs!)

(Eq. 1) 9V Source This is a voltage source, so $v_{n_1} - v_{n_3} = 9V$.

(Eq. 2) 1Ω Resistor This is a resistor, so $V = iR$, and $v_{n_1} - v_{n_2} = (-i_1) \cdot 1\Omega$

(Eq. 3) 3Ω Resistor This is a resistor, so $V = iR$, and $v_{n_2} - v_{n_3} = i_2 \cdot 3\Omega$

(Eq. 4) 6Ω Resistor This is a resistor, so $V = iR$, and $v_{n_2} - v_{n_3} = i_3 \cdot 6\Omega$

Now, we can solve the system of equations to find each of the node voltages $(v_{n_1}, v_{n_2}, v_{n_3})$ and currents (i_1, i_2, i_3).

Adding -6 times Eq. 2, 2 times Eq. 3, and 1 time Eq. 4, we get

$$-6(v_{n_1} - v_{n_2}) + 2(v_{n_2} - v_{n_3}) + (v_{n_2} - v_{n_3}) = -6((-i_1) \cdot 1\Omega) + 2(i_2 \cdot 3\Omega) + (i_3 \cdot 6\Omega)$$

Simplifying results in

$$-6v_{n_1} + 9v_{n_2} - 3v_{n_3} = i_1 \cdot 6\Omega + i_2 \cdot 6\Omega + i_3 \cdot 6\Omega$$

Factoring

$$-6v_{n_1} + 9v_{n_2} - 3v_{n_3} = (i_1 + i_2 + i_3)6\Omega$$

and since $i_1 + i_2 + i_3 = 0$,

$$-6v_{n_1} + 9v_{n_2} - 3v_{n_3} = 0$$

Since we defined ground, we know that $v_{n_3} = 0V$. From Eq. 1 ($v_{n_1} - v_{n_3} = 9V$), we know that $v_{n_1} = 9V$.

Plugging in to the above equation, we get $-6(9V) + 9v_{n_2} - 3(0V) = 0$ which implies that $9v_{n_2} = 54V$ and $v_{n_2} = 6V$.

Then we can solve for i_1, i_2, i_3 using Eq. 2, Eq. 3, and Eq. 4.

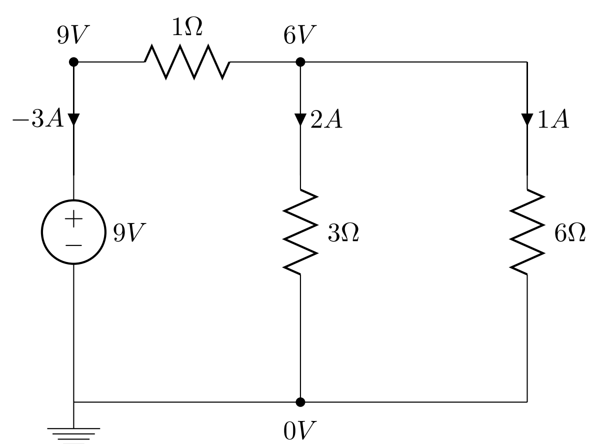
$9V - 6V = (-i_1)1\Omega$, so $i_1 = -3A$.

$6V - 0V = i_2 \cdot 3\Omega$, so $i_2 = 2A$.

$6V - 0V = i_3 \cdot 6\Omega$, so $i_3 = 1A$.

Thus, we have that

$v_{n_1} = 9V, v_{n_2} = 6V, v_{n_3} = 0V$, $i_1 = -3A, i_2 = 2A, i_3 = 1A$.



In summary, whenever you solve a circuit using nodal analysis,

1. Pick a reference point (ground), if necessary.
2. Define the "interesting" nodes in the circuit.
3. Use KCL to write equations relating the current at every node.
4. Write the equations for each devices current as a function of node voltages.

5. Solve the resulting system of equations.

Kind of a pain, isn't it? It's slow and tedious, but it's reliable. You can use this if you want, but there has to be a faster way . . .

6.2 Thevenin Equivalents

Dreading solving circuits yet? Introducing Thevenin equivalent circuits, the "no math" (. . . or at least, less linear algebra . . .) way of solving circuits!

The basic idea behind Thevenin equivalent circuits is that **any** circuit consisting of only linear components (in this class, voltage sources, current sources, and resistors) can be reduced to a single voltage source in series with a single resistor.

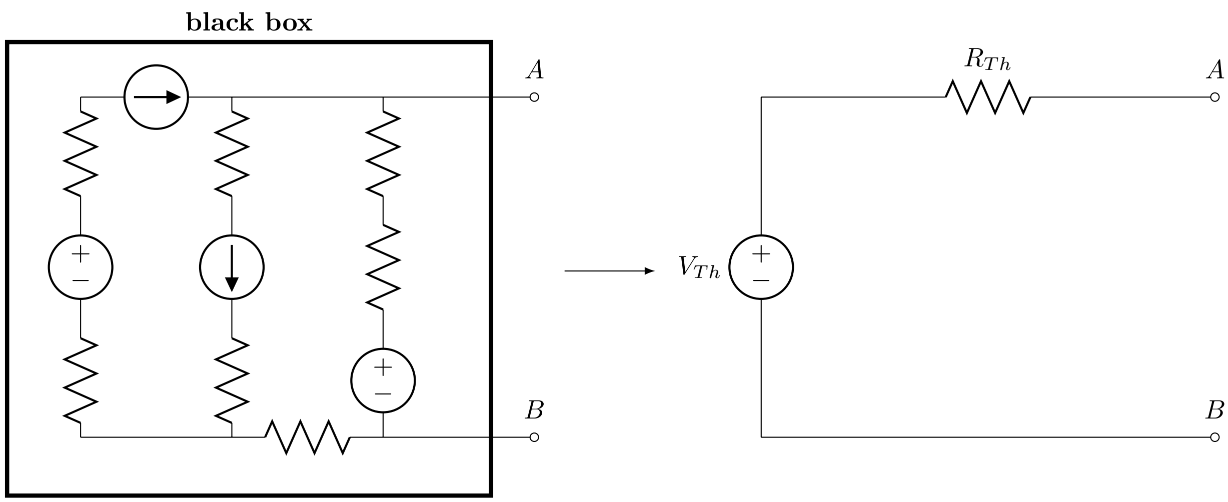
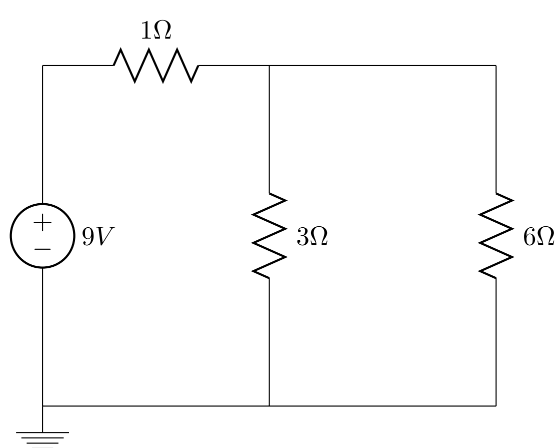


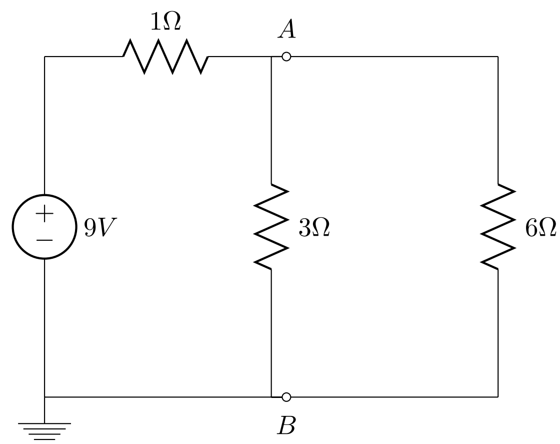
Figure 6.1: Thevenin equivalent – magic!

So, how does it work? Without going into detail here, the general idea is that voltage sources, current sources, and resistors all have linear I-V curves. When you combine components in a circuit, you get a new I-V curve. It happens that any number of linear components combined together in a circuit form another linear curve. Since linear curves can be represented by $y = mx + b$, they can also be represented by a voltage source (b) in series with a resistor (mx).

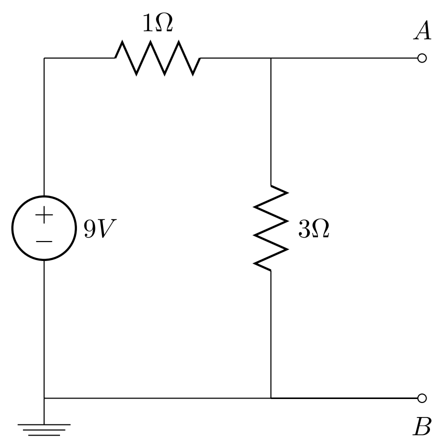
Let's take a look at our nodal analysis problem again (note that we have already chosen a ground):



We need to begin by defining nodes for our black box. I'm going to break off a part of the circuit that looks manageable:

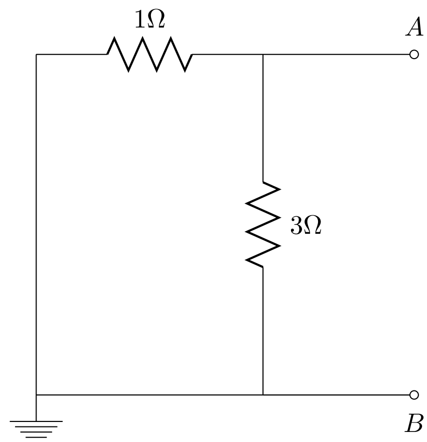


Let's take a look at the left side of the circuit:



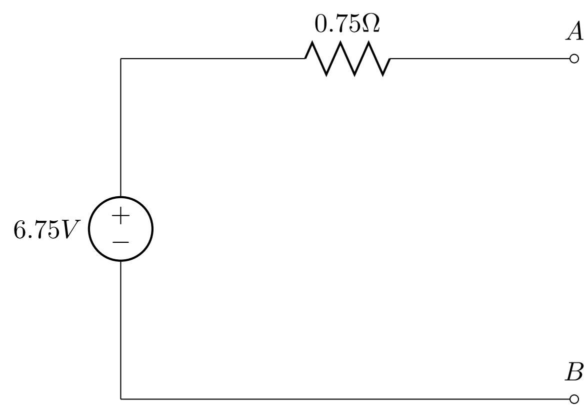
We need to find the open circuit voltage (the voltage difference between A and B); this is the Thevenin equivalent voltage, V_{Th}. Using our voltage divider formula (see figure 4.7), we get that the open circuit voltage is $\frac{3\Omega}{1\Omega+3\Omega}(9V - 0V) + 0V = \frac{27}{4}V = \mathbf{6.75V}$

We also need to find the equivalent resistance, R_{Th}. We do this by setting all sources to zero. Voltage sources become 0 volts, so they act like wires. Current sources become 0 amps, so they act like broken wires. We want to find the equivalent resistance between A and B in the following figure:

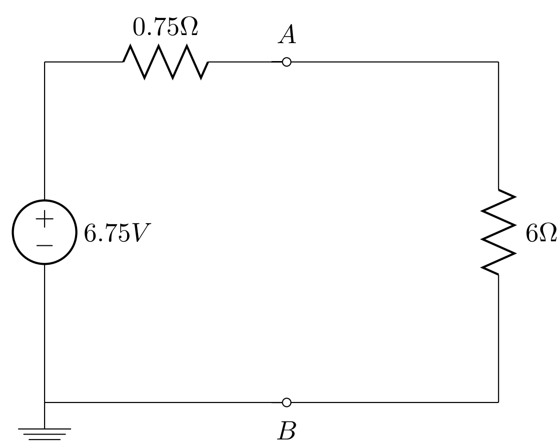


From our formula for resistors in parallel (see figure 4.8), we have that $\frac{1}{R_{parallel}} = \frac{1}{1\Omega} + \frac{1}{3\Omega}$, therefore $R_{parallel} = \frac{3}{4}\Omega = \mathbf{0.75\Omega}$.

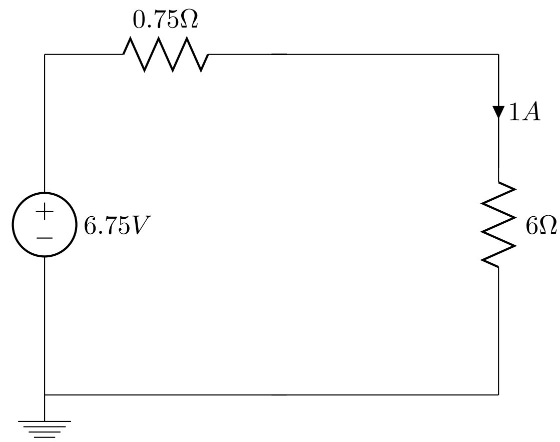
Thus, our Thevenin equivalent circuit looks like this:



Substituting back into the original circuit,



From our formula for current in voltage dividers (see figure 4.6), we have that the current flowing through the 6Ω resistor is $\frac{6.75V}{6\Omega+0.75\Omega} = \frac{6.75V}{6.75\Omega} = 1A$.



This **exactly the same** result we got from nodal analysis. You can now solve the rest of the circuit fairly easily using KCL, KVL, and Ohm's law.

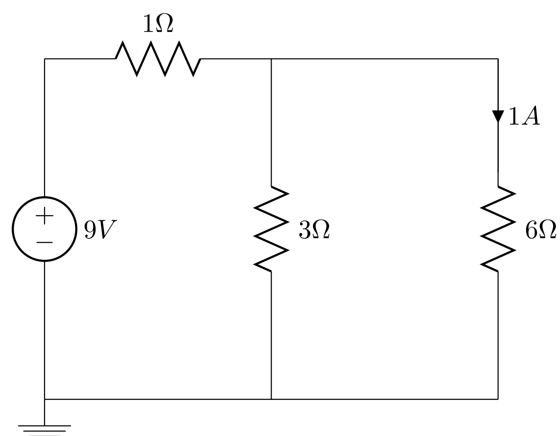
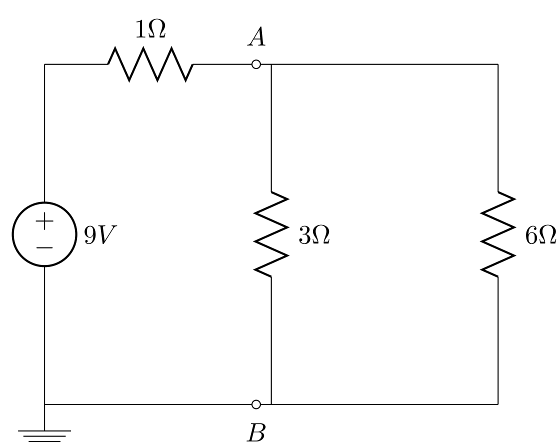
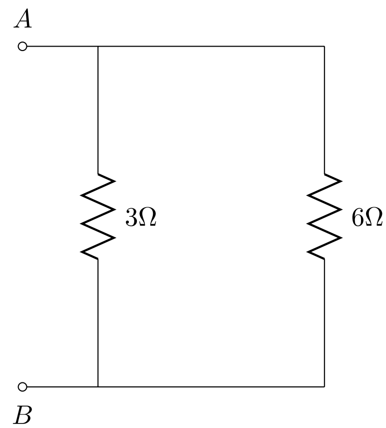


Figure 6.2: Thevenin equivalent result

Note that where you split the circuit does matter. Choosing your split points wisely can make your life easier. For example, consider splitting to the left of the T-junction instead of the right.



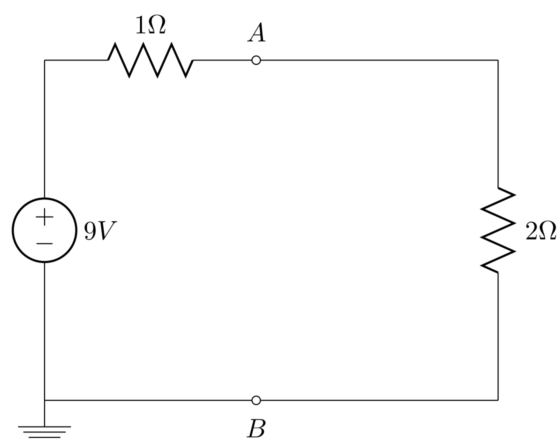
Let's take a look at the right side of the circuit:



We need to find the open circuit voltage (the voltage difference between A and B); this is the Thevenin equivalent voltage, V_{Th}. There are no sources in this circuit, so $V_{Th} = \mathbf{0V}$.

We also need to find the equivalent resistance, R_{Th}. We do this by setting all sources to zero, but there are no sources to set to 0. The equivalent resistance between A and B is given by the formula for resistors in parallel (see figure 4.8): $\frac{1}{R_{parallel}} = \frac{1}{3\Omega} + \frac{1}{6\Omega}$, therefore $R_{parallel} = \mathbf{2\Omega}$.

Substituting back into the original circuit,



From our formula for current in voltage dividers (see figure 4.6), we have that the current flowing through the 1Ω resistor is $\frac{9V}{1\Omega+2\Omega} = \frac{9V}{3\Omega} = 3A$.

Again, this is the same result we got from nodal analysis. (remember that when you change direction, you change the sign).

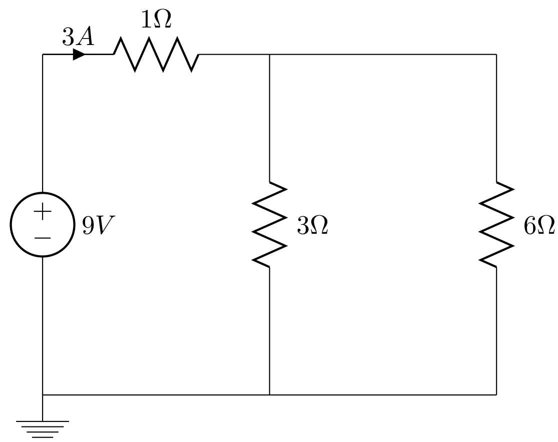


Figure 6.3: Thevenin equivalent result

In summary, whenever you solve a circuit using Thevenin equivalents,

1. Pick a reference point (ground), if necessary.
2. Break the circuit into manageable chunks.
3. Find the Thevenin equivalent circuit for each chunk.
4. Recombine the pieces and continue reducing the circuit until it is solvable.
5. Solve the remainder of the circuit using KCL, KVL, and Ohm's law.

To find the Thevenin equivalent circuit,

1. Find the open circuit voltage (the voltage difference between the nodes A and B when there is nothing connecting A and B), this is the Thevenin equivalent voltage, V_{Th}.
2. Find the equivalent resistance (set all sources to zero: voltage sources become wires (closed circuits) and current sources become broken wires (open circuits), then find the resistance between the nodes A and B), this is the Thevenin equivalent resistance, R_{Th}).

Much better! But whenever we talk about Thevenin equivalents, we also need to talk about its mirror image, Norton equivalents.

6.3 Norton Equivalents

Norton equivalent circuits are pretty much the same thing as Thevenin equivalent circuits, except that you have a current source in parallel with a resistor instead of a voltage source in series with a resistor.

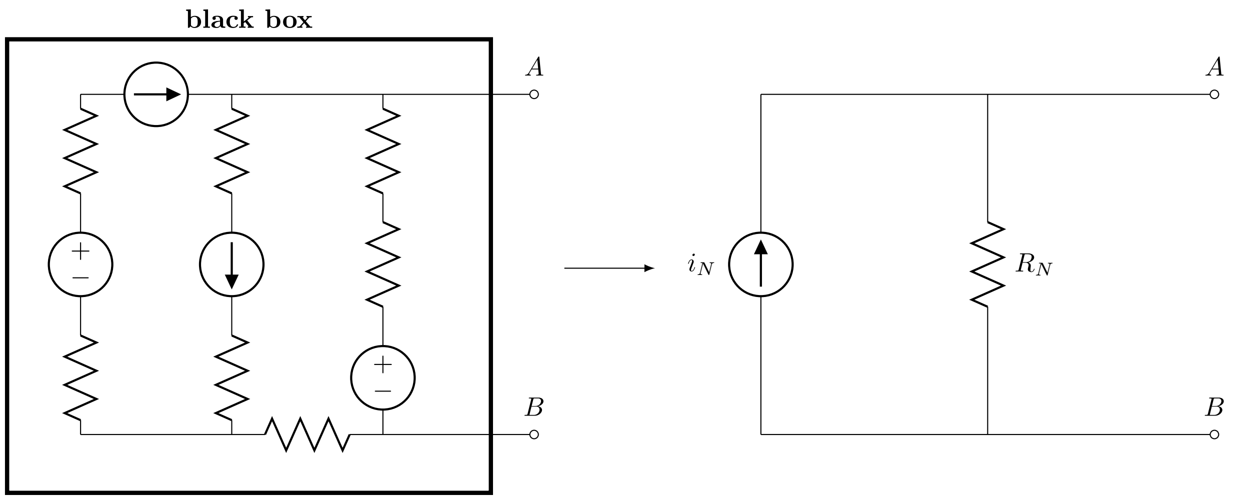
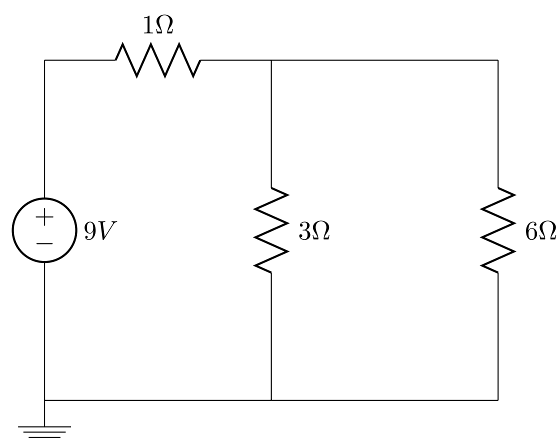
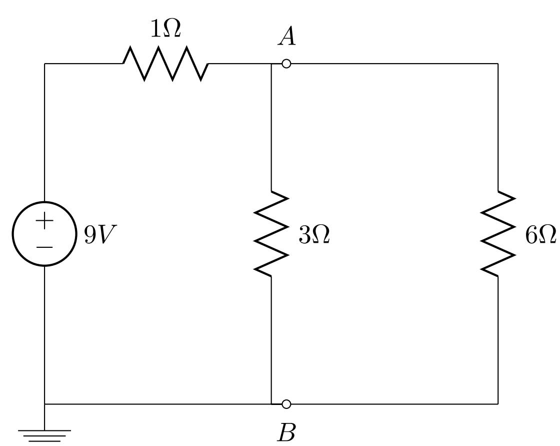


Figure 6.4: Norton equivalent – also magic!

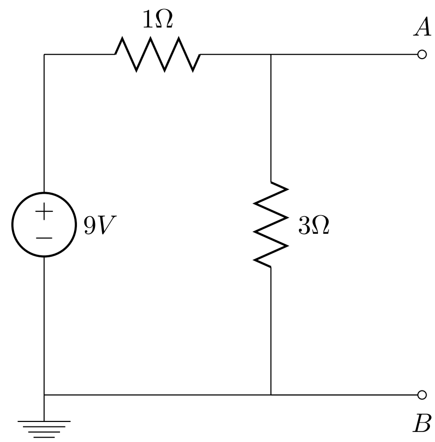
Let's take a look at our nodal analysis problem again (note that we have already chosen a ground):



We need to begin by defining nodes for our black box. Break off a part of the circuit that looks manageable:



Let's take a look at the left side of the circuit:



We need to find the **closed circuit current** (the amount of current that would flow if there was a wire between A and B); this is the Norton equivalent current, i_N.

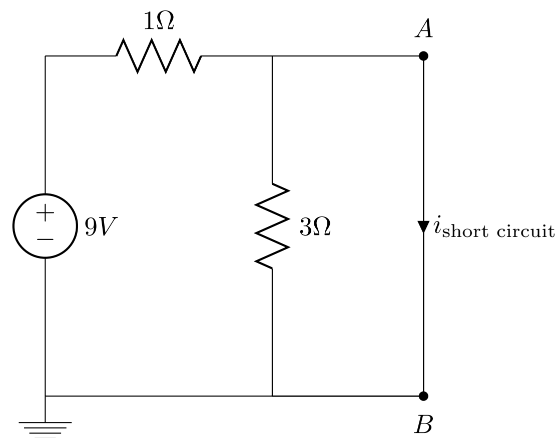
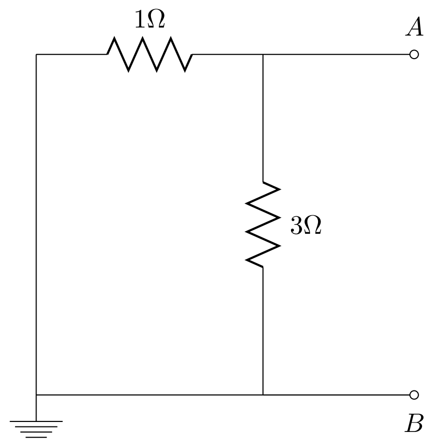


Figure 6.5: Finding the short circuit current

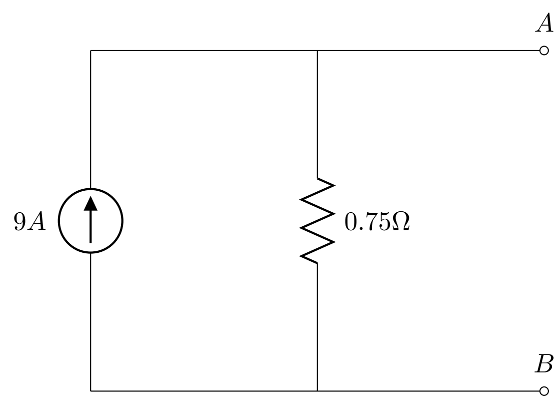
Using KVL through the loop on the far right, we get that the voltage drop across the 1Ω resistor is $9V$. Applying Ohm's law ($V = iR$), we get that the closed circuit current is $\frac{9V}{1\Omega} = \mathbf{9A}$.

We also need to find the equivalent resistance, R_N. We do this by setting all sources to zero. Voltage sources become 0 volts, so they act like wires. Current sources become 0 amps, so they act like broken wires. We want to find the equivalent resistance between A and B in the following figure:

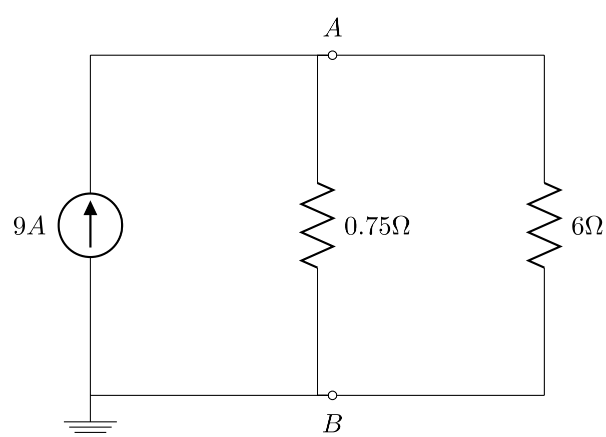


From our formula for resistors in parallel (see figure 4.8), we have that $\frac{1}{R_{parallel}} = \frac{1}{1\Omega} + \frac{1}{3\Omega}$, therefore $R_{parallel} = \frac{3}{4}\Omega = \mathbf{0.75\Omega}$.

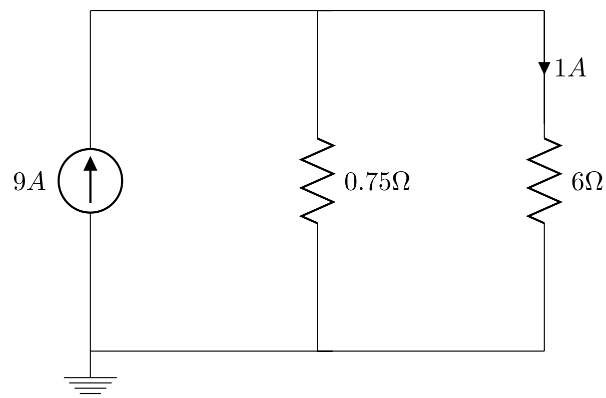
Thus, our Norton equivalent circuit looks like this:



Substituting back into the original circuit,



From our formula for current in current dividers (see figure 4.8), we have that the current flowing through the 6Ω resistor is $9A\frac{0.75\Omega}{6\Omega+0.75\Omega} = 9A\frac{0.75V}{6.75\Omega} = 1A$.



This **exactly the same** result we got from nodal analysis and Thevenin equivalents. You can now solve the rest of the circuit fairly easily using KCL, KVL, and Ohm's law.

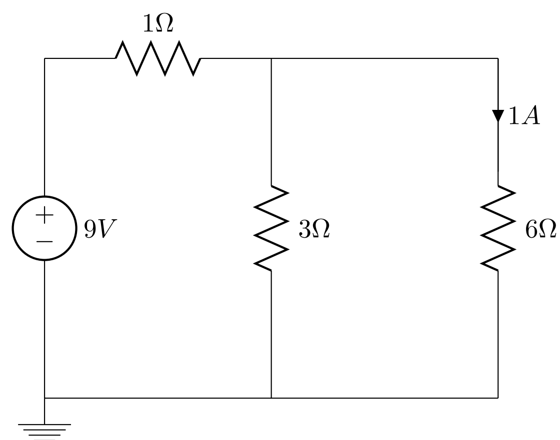


Figure 6.6: Norton equivalent result

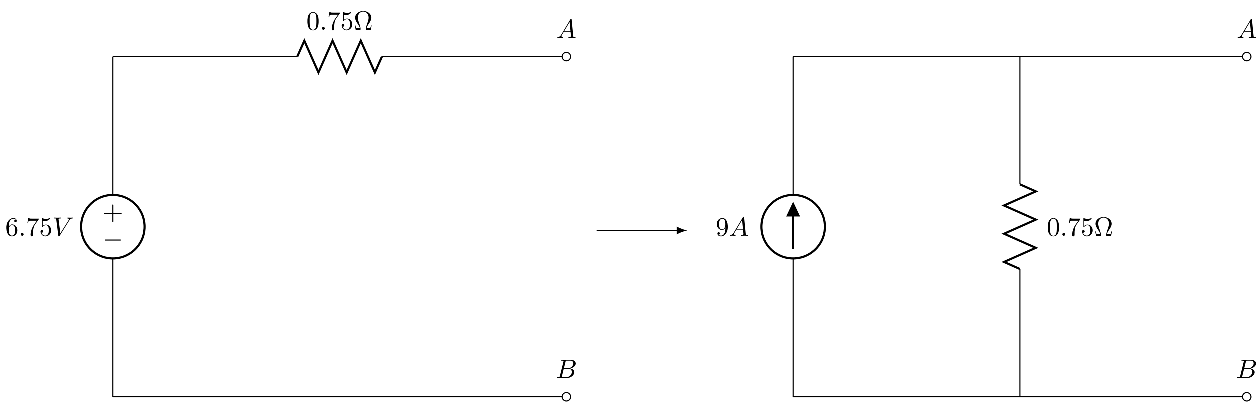
In summary, whenever you solve a circuit using Norton equivalents,

1. Pick a reference point (ground), if necessary.
2. Break the circuit into manageable chunks.
3. Find the Norton equivalent circuit for each chunk.
4. Recombine the pieces and continue reducing the circuit until it is solvable.
5. Solve the remainder of the circuit using KCL, KVL, and Ohm's law.

To find the Norton equivalent circuit,

1. Find the closed circuit current (the amount of current that would flow if there was a wire between A and B); this is the Norton equivalent current, i_N.
2. Find the equivalent resistance (set all sources to zero: voltage sources become wires (closed circuits) and current sources become broken wires (open circuits), then find the resistance between the nodes A and B), this is the Norton equivalent resistance, R_N).

You might have noticed some similarities between the Thevenin equivalent and Norton equivalent for the left branch. This isn't a coincidence.



Hmm $\ldots 6.75V = 9A \cdot 0.75\Omega \ldots V = iR$?!

It turns out that $R_{Th} = R_N$ and $V_{Th} = I_N R_N$.

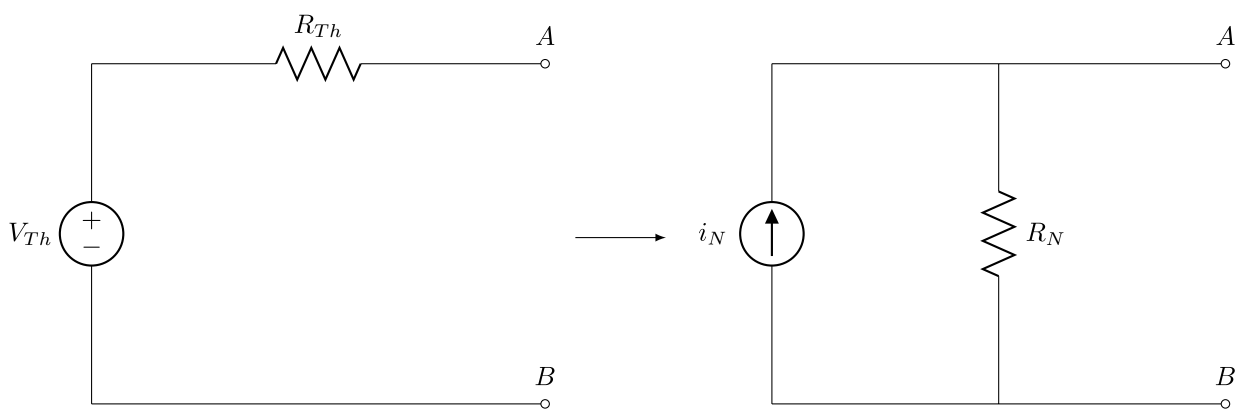


Figure 6.7: Magic = Magic

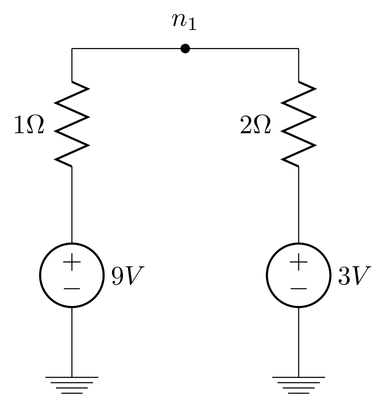
This equivalence can be used to change Thevenin circuits to Norton circuits and vice versa, which can make solving some circuits easier.

6.4 Superposition

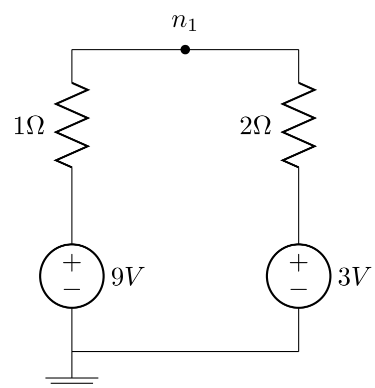
For certain circuits (in our case, circuits with only voltage sources, current sources, and resistors), we can simplify them using a technique called superposition.

This technique involves going through each voltage and current source one at a time (turning off the others, as described previously) and solving the circuit for each source, then going back and adding all of the voltage/current values from each individual source to find the total voltage/current.

This works by a similar principle to Thevenin and Norton equivalents; the linearity of components means that the order of summation (all sources together or one source at a time) is irrelevant.
A brief example:



Wait what? Two grounds? Remember that we define ground to be a reference node, so all grounds are connected. Thus, we can redraw the circuit like this.



Say we want to find the voltage at n_1, called v_{n_1}. We can use superposition to do so!
Let's try first by turning on only the source on the left. Then, the source on the right goes to $0V$, which is like a connected wire (remember that a current source goes to $0A$, which is like a broken wire).

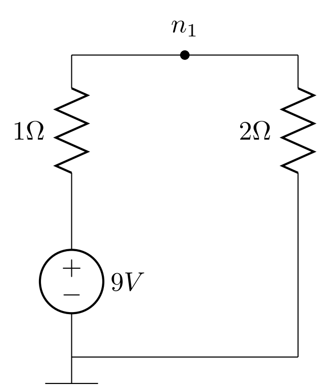
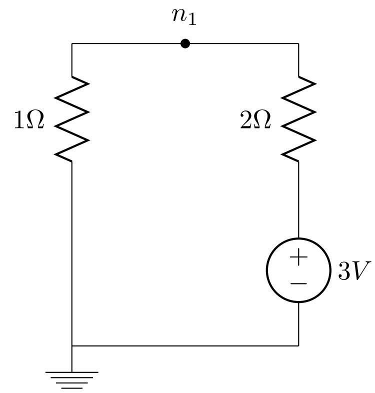


Figure 6.8: Left source only

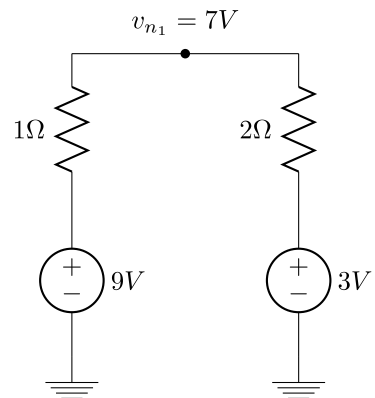
By our voltage divider formula (see fig 4.7), the voltage at n_1 from only the left source is $\frac{2\Omega}{1\Omega+2\Omega}(9V) = 6V$. ($V_{bot} = 0V$)

Similarly, for turning on only the source on the right, we set the source on the left to $0V$ (connected wire).



By our voltage divider formula (see fig 4.7), the voltage at n_1 from only the right source is $\frac{1\Omega}{2\Omega+1\Omega}(3V) = 1V$.

Then, by superposition, since we have counted all sources, we can sum up the voltage contributions from each source and find the final voltage. The contribution from the left source is $6V$ and the contribution from the right source is $1V$, so the voltage at n_1 in the original circuit is $6V + 1V = 7V$.

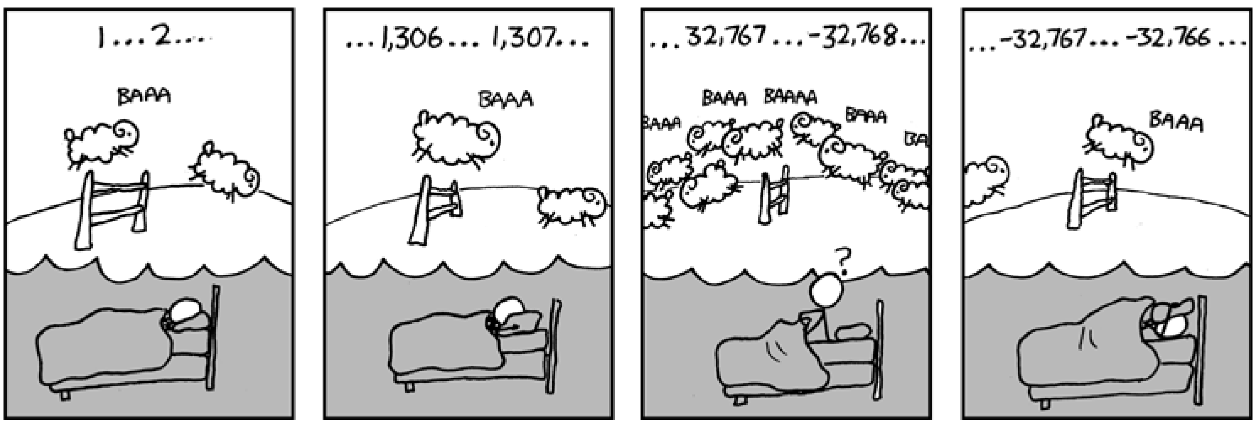


Note that superposition can be applied to find **both voltage and current** at a given node. (We only did voltage in this example).
In summary, whenever you solve a circuit using superposition,

1. Pick a node or device of interest.
2. Choose whether you want to find the voltage or current (or both) at that node, and pick a reference direction if necessary.
3. Turn on one source at a time (turn off all other sources) and solve for the voltage or current at the node of interest. (Remember to keep your signs straight!)
4. Combine the contributions from each source to find the final value at the node.
5. Solve the remainder of the circuit using KCL, KVL, and Ohm's law.

Chapter 7

Digital Logic Basics



What happens when ~~you take this class~~ androids dream of electric sheep.

7.1 Switches

Switches are mechanical devices that control the connection of electrical terminals.

Poles Number of switch outputs.

Throws Number of mechanically connected switches.

Major switch types:

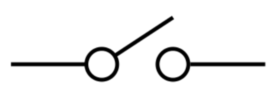


Figure 7.1: Single pole, single throw switch (SPST)

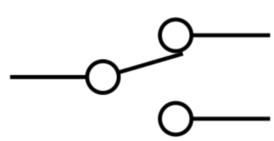


Figure 7.2: Single pole, double throw switch (SPDT)

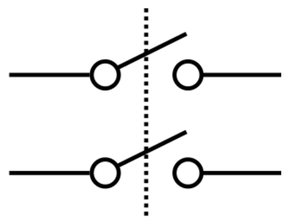


Figure 7.3: Double pole, single throw switch (DPST)

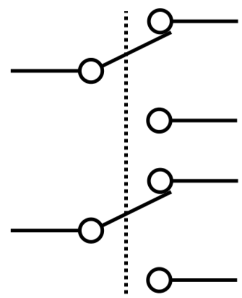


Figure 7.4: Double pole, double throw switch (DPDT)

7.2 Finite State Machines

Deterministic finite state machines (FSM) are models of a computing device. If you've taken CS 103, this is similar to (but less restrictive than) deterministic finite automata. In ENGR 40M, we use finite state machines to discuss inputs, outputs, and feedback. In particular, we want to discuss the operation of our useless box.

Finite state machines are a set of states connected by transitions. Generally, transitions correspond to inputs a computer can receive, and states correspond to a set of reactions to inputs. A finite state machine can only be in one state at a time. Inputs (transitions) can cause the machine to change the state that it is in.

Consider a light switch. It has two states: light_off and light_on. The transitions between the states are switch_connected and switch_disconnected. The FSM would look like this:

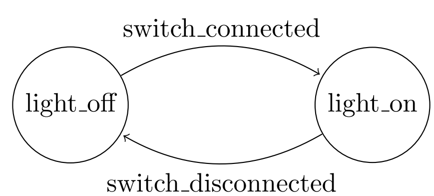


Figure 7.5: Finite state machine model of a light switch

We can use a similar model for more complex systems, like our useless box. First, consider the states necessary to represent the useless box. We want the useless box to be able to go forwards, go backwards, and stop. Thus, we will have three states:

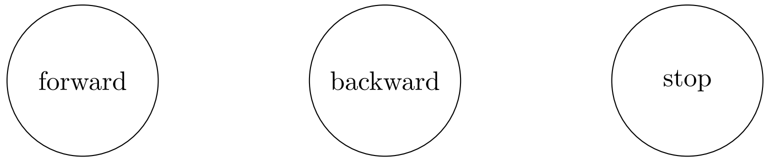


Figure 7.6: States in the finite state machine model of the useless box

Next, we want to consider the possible inputs to the useless box. We have a DPDT toggle switch and a SPST limit switch. We will call these inputs toggle_up, toggle_down, and limit_pressed.

Now, for each state, we will consider the reaction to each input.

1. For the forward state:

   toggle_up The arm is already moving forward, so we stay in the forward state.

   toggle_down The arm is already moving forward, but now we want it moving backward, so we move to the backward state.

   limit_pressed The arm is moving forward, so this shouldn't happen. However, if it does happen, we can just ignore it since we're in no danger of over-reversing. Thus, we stay in the forward state.

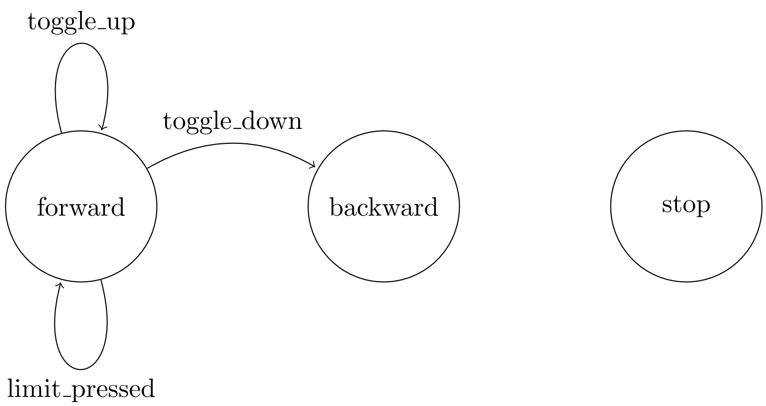


Figure 7.7: FSM model of useless box with transitions from forward state

2. For the backward state:

   toggle_up The arm is already moving backward, but now we want it moving forward, so we move to the forward state.

   toggle_down The arm is already moving backward, so we stay in the backward state.

   limit_pressed The arm is moving backward, so when the limit switch is pressed, we want to stop reversing to prevent over-reversing. Thus, we move to the stop state.

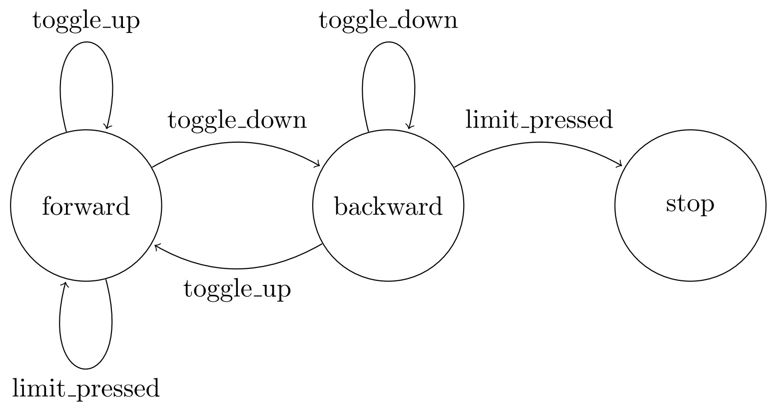


Figure 7.8: FSM model of useless box with transitions from forward state and backward state

3. For the stop state:

   toggle_up The arm is stopped, but now we want it moving forward, so we move to the forward state.

   toggle_down The arm is stopped, but we don't want it to reverse (since that would cause over-reversing), so we stay in the stop state.

   limit_pressed The arm is stopped and sitting on the limit switch, so the limit switch being pressed shouldn't matter. Thus, we stay in the stop state.

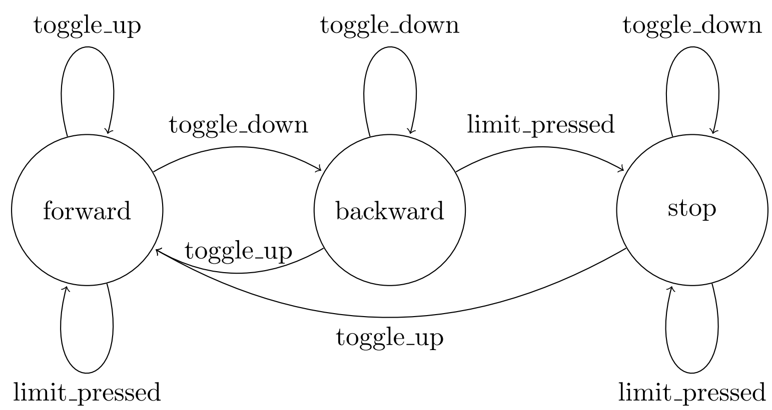


Figure 7.9: FSM model of useless box with transitions from all states

However, drawing a finite state machine in this way quickly becomes cluttered. To improve the clarity of the diagram, we omit all transitions that do not lead to a new state.

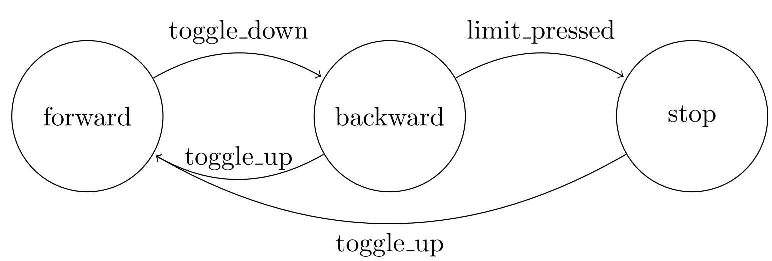


Figure 7.10: FSM model of useless box

In summary, whenever you build a finite state machine,

1. Define the states necessary to represent your computer.
2. Define the possible inputs to the computer.
3. Determine transitions between states by going through each state and assigning transitions to inputs.

7.3 Boolean Logic

Boolean logic is the logic of TRUE and FALSE.

Boolean variables are variables that have a value of either TRUE or FALSE. They are generally represented by lowercase letters like $p, q,$ and r.

We have basic operators NOT, AND, OR that operate on boolean variables.

NOT (!) If p is true, $!p$ is false. Also vice versa: if p is false, then $!p$ is true.

p	$!p$
T	F
F	T

Figure 7.11: Logical NOT operator

AND (&&) p && q is true if (and only if) both p and q are true.

p	q	p && q
T	T	T
T	F	F
F	T	F
F	F	F

Figure 7.12: Logical AND operator

OR (||) p \|\| q is true if at least one of p or q is true.

p	q	p || q
T	T	T
T	F	T
F	T	T
F	F	F

Figure 7.13: Logical OR operator

7.4 Boolean Logic in Circuits

TRUE and FALSE are generally represented in a circuit using two voltages: V_{dd} and Gnd (ground).

- V_{dd} corresponds to TRUE and 1. Gnd (ground) corresponds to FALSE and 0.
- V_{dd} varies between systems, but for the Arduino Nanos we will be using for this class, $V_{dd} = 5V$.

- *Gnd* can also vary between systems, but is usually $Gnd = 0V$.

It turns out that 1s and 0s can be used to represent all types of data that are stored by a computer – through binary representation.

7.5 Binary Representation

Computers represent TRUE and FALSE using binary, where TRUE = 1 and FALSE = 0.

Integers can be represented in binary through base 2. (Decimal is base 10).

For example, consider the decimal number 6234.

10^3	10^2	10^1	10^0
6	2	3	4

Figure 7.14: Decimal 6234

The value is simply $(6 \cdot 10^3) + (2 \cdot 10^2) + (3 \cdot 10^1) + (4 \cdot 10^0) = (6 \cdot 1000) + (2 \cdot 100) + (3 \cdot 10) + (4 \cdot 1) = 6234$.

Then, consider the binary number 1101. In a similar fashion:

2^3	2^2	2^1	2^0
0	1	0	1

Figure 7.15: Binary 0101

The value is $(0 \cdot 2^3) + (1 \cdot 2^2) + (0 \cdot 2^1) + (1 \cdot 2^0) = (0 \cdot 8) + (1 \cdot 4) + (0 \cdot 2) + (1 \cdot 1) = 5$.

7.6 Addition and Subtraction

Addition and subtraction in binary are very similar to their decimal counterparts. You add by carrying 1s and subtract by borrowing 2s (instead of borrowing 10s).
An example for addition:

Binary 11 + Binary 3

1	0	1	1
0	0	1	1

Addition step 1

		1	
1	0	1	1
0	0	1	1
			0

Addition step 2

	1	1	
1	0	1	1
0	0	1	1
		1	0

Addition step 3

	1	1	
1	0	1	1
0	0	1	1
	1	1	0

Addition step 4

	1	1	
1	0	1	1
0	0	1	1
1	1	1	0

Addition step 5

Binary 11 + Binary 3 = Binary 14

And, for subtraction:

Binary 9 - Binary 3

1	0	0	1
0	0	1	1

Subtraction step 1

1	0	0	1
0	0	1	1
			0

Subtraction step 2

	2		
~~1~~ 0	0	0	1
0	0	1	1
			0

Subtraction step 3

	~~2~~ 1	2	
~~1~~ 0	0	0	1
0	0	1	1
			0

Subtraction step 4

	~~2~~ 1	2	
~~1~~ 0	0	0	1
0	0	1	1
		1	0

Subtraction step 5

	~~2~~ 1	2	
~~1~~ 0	0	0	1
0	0	1	1
	1	1	0

Subtraction step 6

	~~2~~ 1	2	
~~1~~ 0	0	0	1
0	0	1	1
0	1	1	0

Subtraction step 7

Binary 9 - Binary 3 = Binary 6

It turns out that computers have a limited number of places to store numbers. The number of bits (places) that a computer has to store a number depends on the computer. Arduino Nanos use 16 bits to store integers.

There is a maximum value that can be stored in an integer. For example, the largest 4 digit decimal number is 9999. Analogously, the largest 4 bit binary number is $1111 = 2^0 + 2^1 + 2^2 + 2^3 = 1 + 2 + 4 + 8 = 15$.

It turns out that the maximum value that can be stored in an n bit binary number is equal to $2^n - 1$ (this can be shown by using the formula for a sum of a geometric sequence). Thus, the largest **unsigned** integer that the Arudino can store is $2^{16} - 1 = 65535$.

This limited number of places to store numbers can cause problems when we want to get numbers larger than the maximum or smaller than the minimum.

For example, when we add 1 to the maximum integer, we get **integer overflow**.

1	1	1	
1	1	1	1
0	0	0	1
0	0	0	0

Integer overflow. $15 + 1 = 0$?

This problem stems from the fact that the leftmost value place has a carried 1 that can't be added.

Similarly, when we subtract 1 from the minimum integer, we get **integer underflow**.

~~2~~ 1	~~2~~ 1	~~2~~ 1	2
0	0	0	0
0	0	0	1
1	1	1	1

Integer underflow. $0 - 1 = 15$?

This problem stems from the fact that the leftmost value place has to borrow a 2 that doesn't actually exist.

7.7 Two's Complement

Alright, we have a method of representing non-negative integers using binary. But what if we need negative integers?

Hmm. . . let's make the **most significant bit** (the leftmost bit) a sign bit. What's 2 plus negative 1?

	1		
0	0	1	0
1	0	1	1
1	1	0	1

$2 + (-1) = -5$? This is bad.

Hmm. . . , looks like addition and subtraction no longer work. Also, we have both positive and negative zero. What's negative zero?

Let's try something different. Let's use subtraction to get negative numbers. What's 0 minus 1?

~~2~~ 1	~~2~~ 1	~~2~~ 1	2
0	0	0	0
0	0	0	1
1	1	1	1

Binary 1111 is -1 and 15

Well, we can call binary 1111 to be -1 if we want (even though by our previous definition, binary 1111 = 15).

Actually... this could make some sense. Notice that we have 4 bits, and $2^4 = 16$. If you're familiar with the concept of modulo, we can note here that $15 \equiv -1 (\text{mod } 16)$. If you're not familiar with the concept of modulo, just note that $15 - 16 = -1$.

What's 0 minus 2?

~~2~~ 1	~~2~~ 1	2	
0	0	0	0
0	0	1	0
1	1	1	0

Binary 1110 is -2 and 14

Well, we can call binary 1110 to be -2 if we want (even though by our previous definition, binary 1110 = 14).

Again, if you're familiar with the concept of modulo, we can note here that $14 \equiv -2 (\text{mod } 16)$. If you're not familiar with the concept of modulo, just note that $14 - 16 = -2$.

Wait, since numbers wrap around due to integer overflow and underflow, we can just designate that certain numbers are negative, and use modulo (or for 4 bit numbers, subtracting 16) to determine negative numbers!

This concept of using this circular wrap-around for negative numbers is known as **two's complement**.

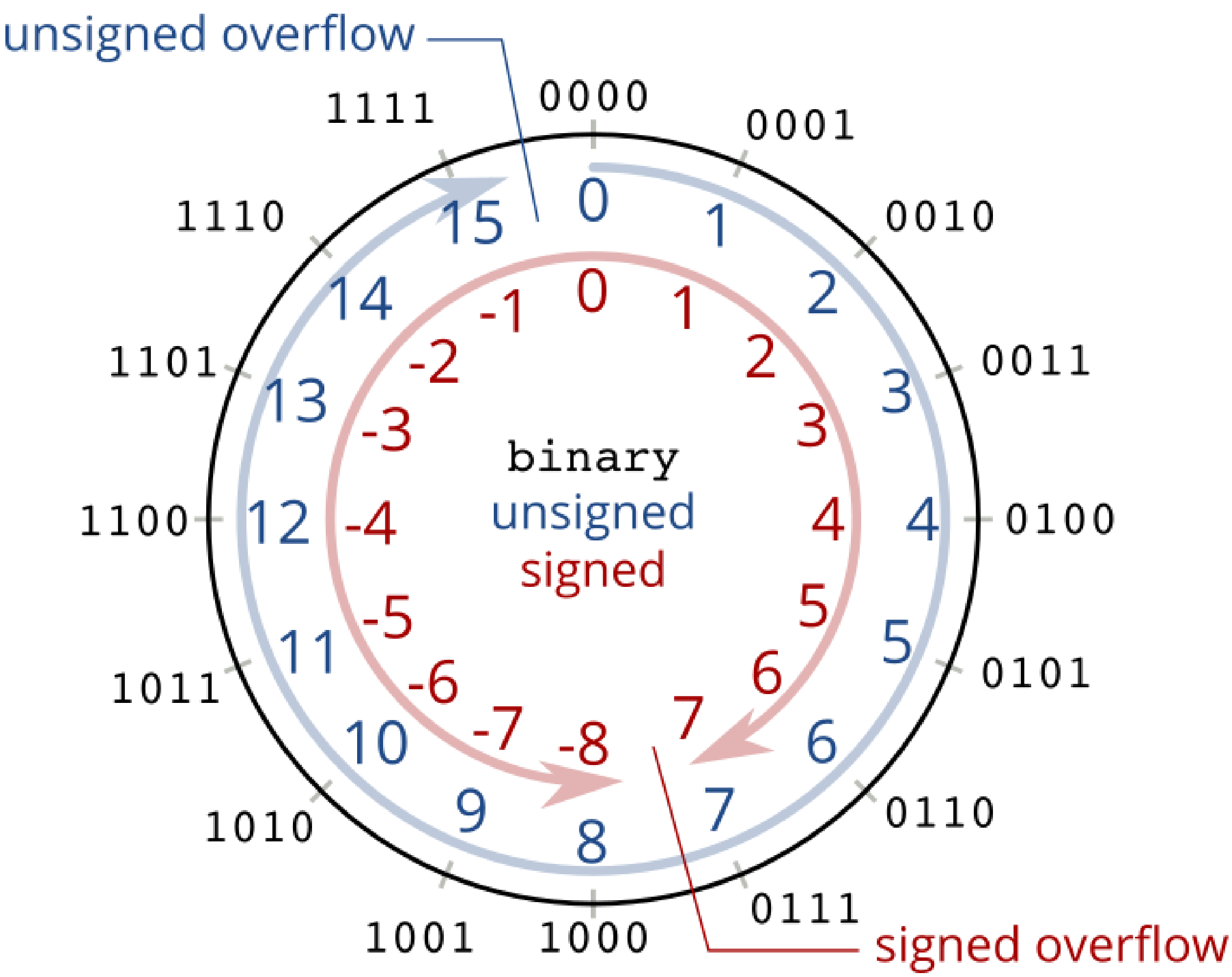


Figure 7.16: Two's complement circle

Since we don't want the bits interpreted to be both 2 and -14 at the same time, we're going to define two conventions for interpreting bits: unsigned and signed.

Unsigned convention is exactly what you've seen in the previous section (binary without any negative numbers). Overflow and underflow occur at the maximum integer and 0.

Signed convention is a little bit different. We want a way to represent negative numbers, so all numbers whose most significant bit (the leftmost bit) is 1 will be negative. This also means that overflow and underflow occur at a different location (see the two's complement circle).

For example, consider the interpretation of binary 1011.

Unsigned: We get $2^0 + 2^1 + 2^3 = \mathbf{11}$.

Signed: We have unsigned 11, but the most significant bit is 1.
Thus, we subtract $2^{\text{number of bits}} = 2^4 = 16$, so we get $11 - 16 = \mathbf{-5}$.

And binary 0111:

Unsigned: We get $2^0 + 2^1 + 2^2 = \mathbf{5}$.

Signed: We have unsigned 5, and the most significant bit is 0. Thus, we have **5**.

Also, changing the sign (multiplying by negative 1) is easy: just invert all the bits and add 1.

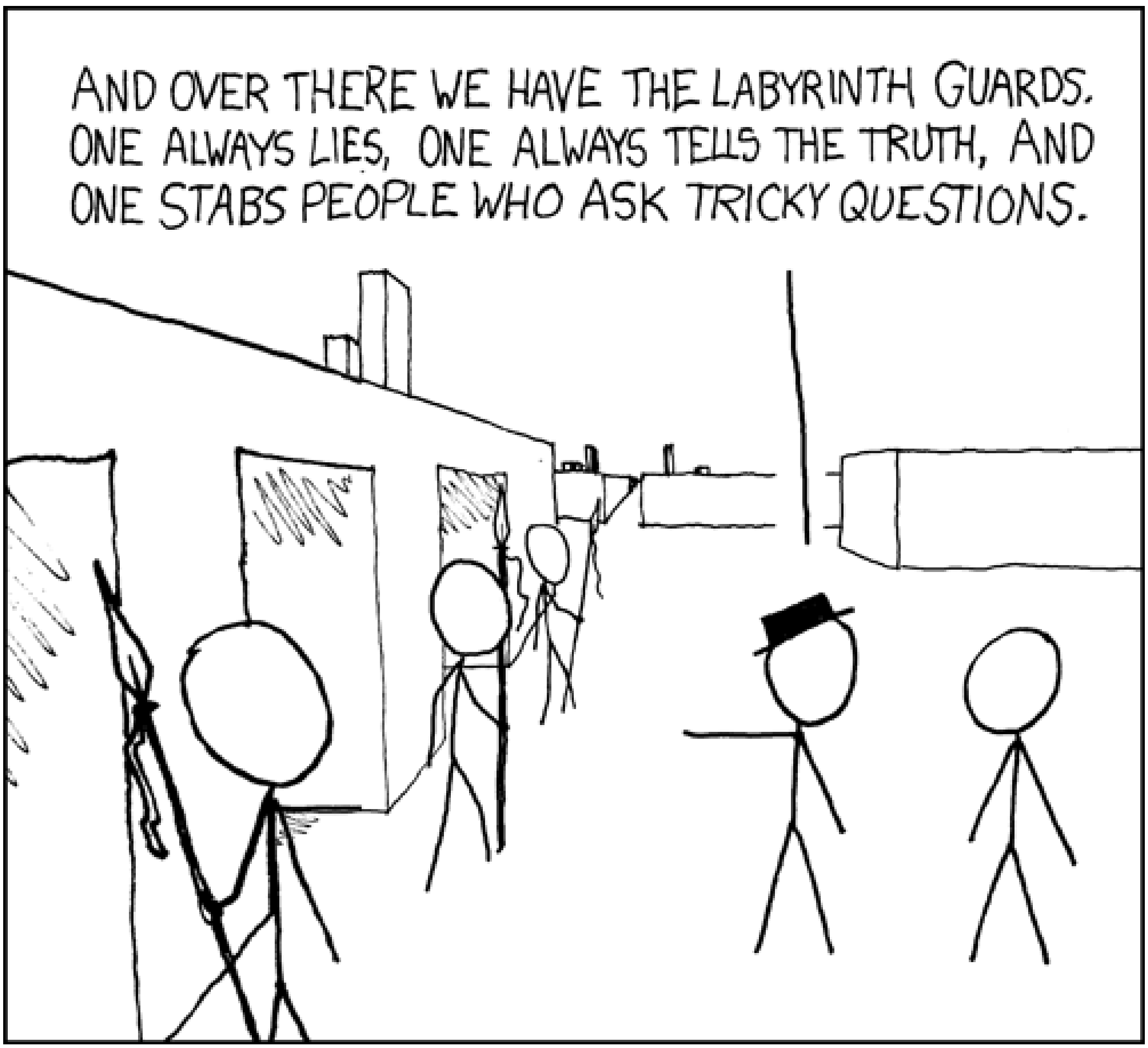
1. Decimal 5 = Binary 0101
2. Invert bits: 1010
3. Add 1: 1011
4. Decimal -5 = Binary 1011
5. Invert bits: 0100
6. Add 1: 0101
7. Decimal 5 = Binary 0101

Magic!

It turns out that, under two's complement, the most positive number is a 0 followed by all 1s, while the most negative number is a 1 followed by all 0s.

Chapter 8

Transistors and Logic Gates



...it may or may not be too late to drop the class.

8.1 MOSFETS

MOSFETs (**M**etal **o**xide **s**emiconductor **f**ield-**e**ffect **t**ransistor**s**) are a class of electronic components with interesting properties.

In this class, we will focus on their usage as electrical switches. In general, they act like SPST switches that are open or closed depending on the voltage at a third terminal (which we will call the **gate terminal**.)

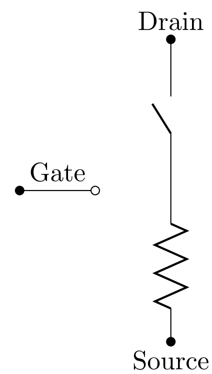


Figure 8.1: Switch model of a transistor

The input to the switch is called the **source terminal**, and the output of the switch (the throw the switch can be connected to) is called the **drain terminal**.

The two major types of transistors are nMOS and pMOS transistors. For the purposes of this class, they differ only in terms of the voltages at each of the terminals (**g**ate, **s**ource, **d**rain) necessary to open or close the switch. We will begin with an examination of nMOS transistors.

8.2 nMOS

nMOS transistors are transistors where the switch is closed if the voltage at the gate terminal is **greater than** the voltage at the source (switch input) terminal (otherwise the switch is open).

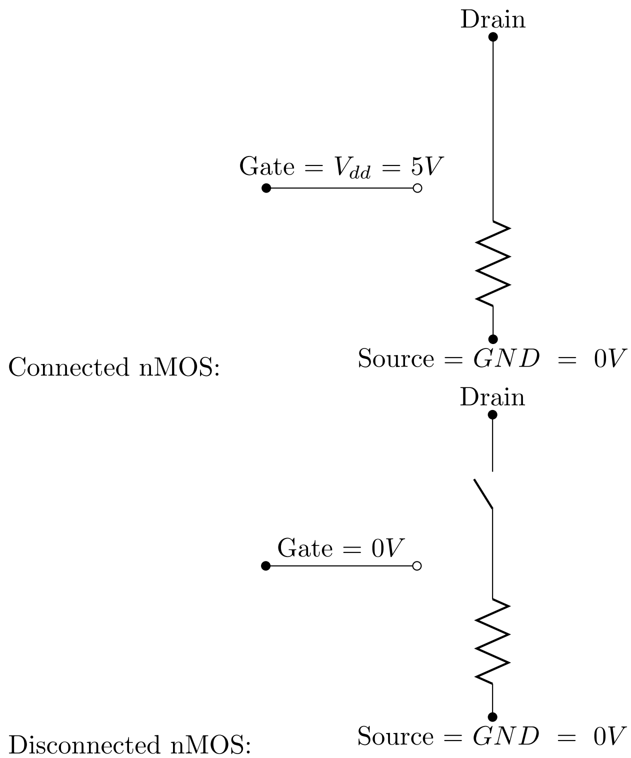


Figure 8.2: Switch model of an nMOS transistor

Note that the voltage at the gate usually needs to be at least 1 volt higher than the voltage at the source for the switch to close.

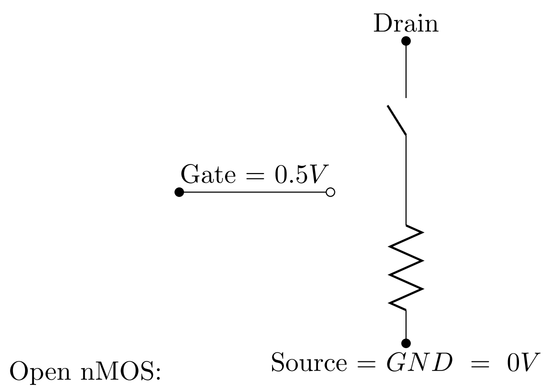


Figure 8.3: Gate voltage higher, but not more than 1 volt greater than the source

For simplicity, nMOS transistors are drawn without the switch, and whether the switch is open or closed is inferred from the voltage at the gate and source terminals.

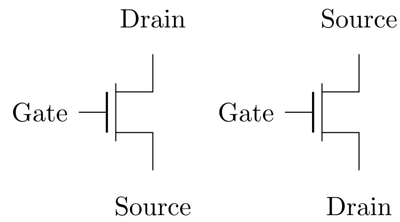


Figure 8.4: Standard nMOS diagrams

Additionally, the drain and source terminals are generally left **unlabeled**, and inferred from the components connected to the terminal (generally, only one of the terminals will have a known voltage input; this is the **source** terminal).

The voltage at the output of an nMOS transistor, the **drain** terminal, depends on the inputs. It will either be connected to the source (and therefore have the same voltage as the source) or disconnected from the source (in which case the voltage is floating, or undefined).

Gate	Source	Drain
0 V	0 V	Floating
2 V	0 V	0 V
5 V	0 V	0 V
0 V	2 V	Floating
2 V	2 V	Floating
2.5 V	2 V	Floating
3 V	2 V	2 V
5 V	2 V	2 V
0 V	5 V	Floating
2 V	5 V	Floating
5 V	5 V	Floating

Figure 8.5: nMOS inputs and output

Notice that the drain is floating unless the gate is at least 1 volt higher than the source, in which case the drain is connected to the source and thus has the same voltage as the source.

For the purposes of this class, the gate terminal does not draw any current (that is, current into the gate terminal is 0 Amps).

8.3 pMOS

pMOS transistors are nearly identical to nMOS transistors, except that the switch is closed if the voltage at the gate terminal is **less than** the voltage at the source (switch input) terminal (otherwise the switch is open).

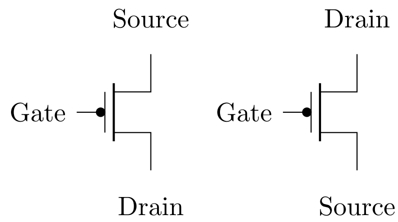


Figure 8.6: Standard pMOS diagrams

The diagrams for pMOS transistors are identical to those of nMOS transistors, except that they are drawn with a bubble at the gate terminal that represents "not". (The reason for this will become clear shortly.) Again, the drain and source terminals are generally left **unlabeled**, and are inferred from the components connected to the terminals.

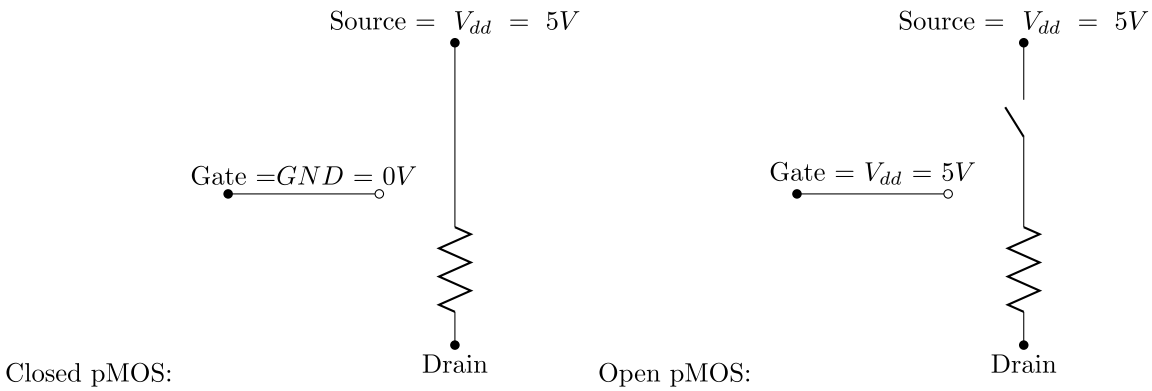


Figure 8.7: Switch model of a pMOS transistor

The reason that pMOS transistors are drawn with a bubble at the gate terminal that represents "not" is due to their functional logic. Consider the following:

Gate	Source	Drain
0 V	0 V	Floating
2 V	0 V	Floating
5 V	0 V	Floating
0 V	3 V	3V
2 V	3 V	3V
2.5 V	3 V	Floating
3 V	3 V	Floating
5 V	3 V	Floating
0 V	5 V	5 V
2 V	5 V	5 V
5 V	5 V	Floating

Figure 8.8: nMOS inputs and output

For digital logic, we usually define voltages for V_{dd} and GND. We use V_{dd} to represent a boolean TRUE (1) and GND to represent a boolean FALSE (0). If we consider the **gate** of the pMOS transistor to be the **input** and the **drain** of the pMOS transistor to be the **output** (the reason for this will become clear when we discuss building logic gates), we see that the drain (output) is significantly higher than the gate (input). In particular, when we have an input of FALSE (GND) and an adequate source (V_{dd}), we have an output of TRUE (V_{dd}), so our input was inverted!

Note that the drain is floating unless the gate is at least 1 volt lower than the source, in which case the drain is connected to the source and thus has the same voltage as the source.

8.4 Logic Gates

Okay, so now we have these magical electrically controlled switches, but how are they useful?

It turns out that transistors can be wired together in ways that serve to act like boolean logic operators – NOT, AND, OR. Additionally, they can be wired together in ways to form even more complex logic operators.

NOT Gate: If **A** is V_{dd}, **B** is GND. Also vice versa: if **A** is GND, then **B** is V_{dd}.

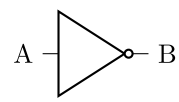


Figure 8.9: NOT gate

AND Gate: **C** is V_{dd} if both **A** and **B** are V_{dd}. Otherwise, **C** is GND.

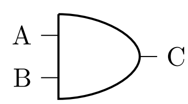


Figure 8.10: AND gate

OR Gate: **C** is V_{dd} if at least one of **A** or **B** is V_{dd}.

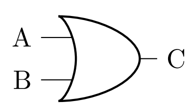


Figure 8.11: OR gate

NAND Gate: **C** is V_{dd} if at least one of **A** or **B** is GND. (Negated AND gate.)

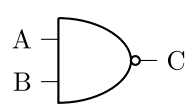


Figure 8.12: NAND gate (NOT-AND gate)

NOR Gate: **C** is GND if at least one of **A** or **B** is V_{dd}. (Negated OR gate.)

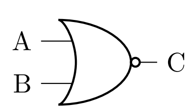


Figure 8.13: NOR gate (NOT-OR gate)

XOR Gate: C is V_{dd} if **exactly one** of **A** or **B** is V_{dd}. (Exclusive OR gate.)

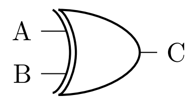


Figure 8.14: XOR gate (Exclusive OR gate)

These gates can be put together in ways to do arithmetic and serve as memory – but that's beyond the scope of this class. Take EE 108 if you want to learn more!

8.5 Constructing Logic Gates

So now we have transistors and we know what logic gates do. How do we go about building these logic gates?

First of all, since nMOS transistors are closed if the voltage at the gate terminal is greater than the voltage at the source terminal, we generally **connect the source of an nMOS transistor to GND**. Similarly, since pMOS transistors are closed if the voltage at the gate terminal is less than the voltage at the source terminal, we generally **connect the source of a pMOS transistor to $\mathbf{V_{dd}}$**.

Let's try building a NOT gate. Consider the requirements. If the input is V_{dd}, we want the output to be GND. If the input is GND, we want the output to be V_{dd}.

Let's see...we want an output of GND when the input is V_{dd}. Since we want a GND output, and we generally connect nMOS transistors to GND sources, let's connect an nMOS to the input and GND.

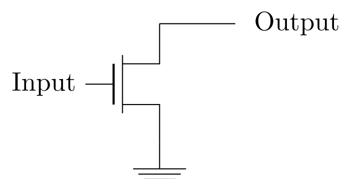


Figure 8.15: Half of a NOT gate

When we connect V_{dd} to the input, we get that the output is GND. Excellent.

We also want an output of V_{dd} when the input is GND. Since we want a V_{dd} output, and we generally connect pMOS transistors to V_{dd} sources, we connect a pMOS to the input and V_{dd}.

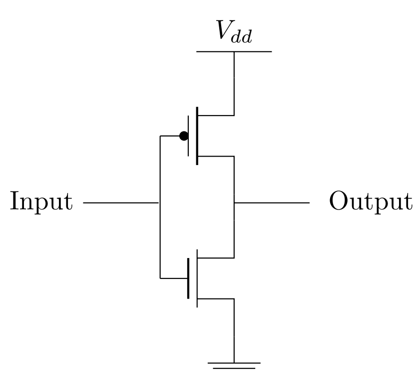


Figure 8.16: Complete NOT gate

When we connect V_{dd} to the input, only the nMOS transistor is closed and the output is GND. When we connect GND to the input, only the pMOS transistor is closed and the output is V_{dd}.

Let's try a little more complex gate. Let's try building a NAND gate. A NAND gate is an inverted AND gate:

p	q	p NAND q
T	T	F
T	F	T
F	T	T
F	F	T

Figure 8.17: Logical NAND operator

Now we have two inputs and one output. We will call the inputs A and B.

If the inputs A and B are both V_{dd}, we want the output to be GND. Otherwise, we want the output to be V_{dd}. This happens if at least one of the inputs is GND.

Let's start with inputs A and B both being V_{dd} and the output as GND. Since we want a GND output, we will connect a nMOS transistors to the inputs and GND.

But we only want a connection to GND when both inputs are V_{dd}. One input at V_{dd} isn't good enough. How do we require that both inputs are at V_{dd}?

We put the transistors in series so that both transistors must be "closed" in order for current to flow.

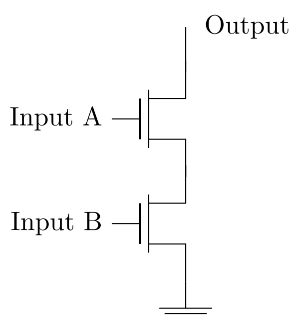


Figure 8.18: Fraction of a NAND gate

We also want a connection to V_{dd} when at least one of inputs A and B is at GND. Since we want a V_{dd} output, we will connect a pMOS transistor to the inputs and V_{dd}.

We want a connection when either or both A and B is at GND, so we put the transistors in parallel, so if either transistor is "closed" current will flow.

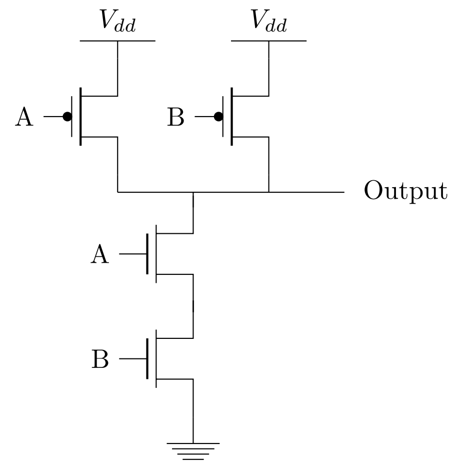


Figure 8.19: Complete NAND gate

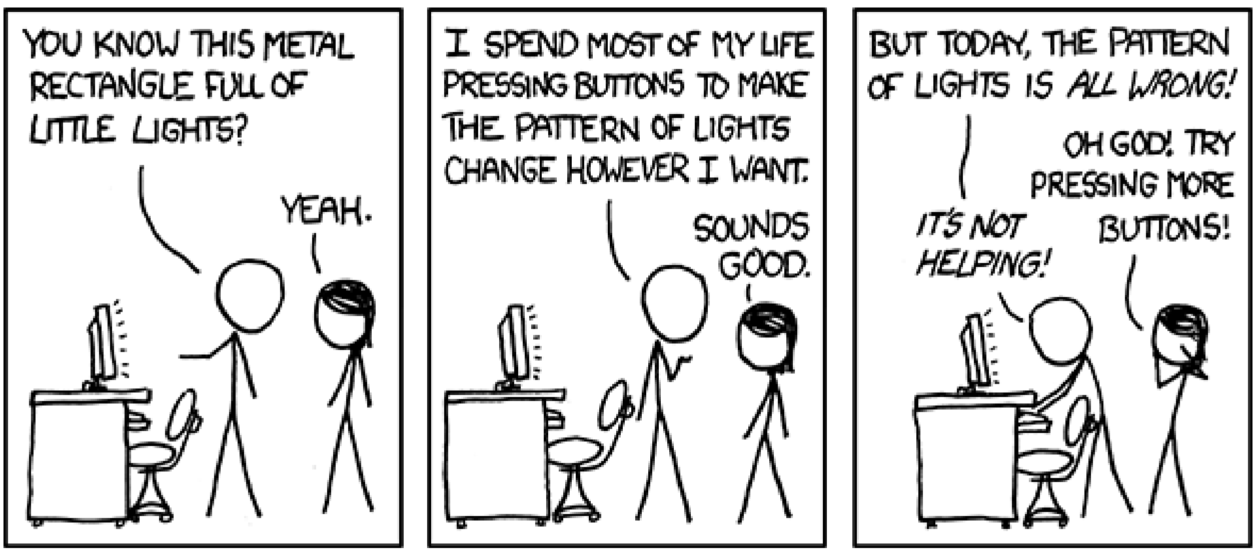
The key takeaway here is that if you need **all inputs** to be at a certain state for a particular output, put them in **series**; if you need **at least one input** to be at a certain state for a particular output, put them in **parallel**.

In summary, whenever you build a logic gate,

1. Choose an output voltage.
2. Determine what type of transistor is necessary and whether they are connected to V_{dd} or GND.
3. Determine the arrangement of the transistors (parallel, series, etc.)
4. Repeat for all output voltages.
5. Verify your gate by checking that every combination of inputs yields the expected output.

Chapter 9

Computers



Basically your life.

9.1 Arduino Basics

Microcontrollers are small computers designed to control stuff. They typically incorporate a processor, RAM, flash memory, etc. all on a single chip.

Arduino is a specific software development system to control Arduino microcontrollers (like the ones in your lab kit.)

9.2 Arduino Input/Output

The Arduino has a relatively basic input/output (I/O) system.

The Arduino has **digital pins** that can read either LOW (GND) or HIGH (V_{dd}). These pins are indicated by pin numbers, e.g. 1, 4, 13. These pins can also be programmed to output any voltage GND and V_{dd}.[1]

[1] The Arduino doesn't actually set the pin to the voltage; rather, it generates a square wave with a duty cycle such that the average voltage of the square wave is the desired voltage.

The Arduino also has **analog pins** that can do everything that the digital pins do; in addition, they **can read voltages between GND and $\mathbf{V_{dd}}$** (instead of just GND and V_{dd}). The pin number is prefaced with an A, e.g. $A0$.

There are three pin modes for pins in digital mode.

OUTPUT This means that the Arduino will drive the voltage of the pin.

INPUT This means that the Arduino will read the voltage of the pin. If the input is ever floating, the values the Arduino reads will be unpredictable; thus, this pin mode should not be used if the input is ever floating.

INPUT_PULLUP This means that the Arduino will attach the input to an internal pull-up resistor and read the voltage of the pin. This means that if the input is ever floating, the pull-up resistor will pull the voltage of the pin to HIGH (V_{dd}).

9.3 Arduino Programming

The Arduino is programmed with a C/C++ dialect; an important thing to note is the presence of the **setup** and **loop** functions. The **setup function** is run once, when the Arduino starts up; the **loop function** is run forever after the setup function has been run once.

9.4 Arduino Tips

The Arduino Nano is pretty slow (16 MHz, while modern computers often run at 4 GHz which is 250 times faster). Thus, avoid using floating-point operations, and minimize the size of your variables (e.g. using byte (8 bits) or int (16 bits on the Arduino) instead of long (32 bits on the Arduino)).

9.5 Codes

No, this isn't the kind of code you upload to your Arduino. Code essentially means that we can use tricks (like linear algebra) to reduce the amount of information we need to send/store/etc. It's certainly an important concept, and it's covered in chapter 5 of the reader. I'm going to put this here and get to writing about this later…

9.6 Time Division Multiplexing

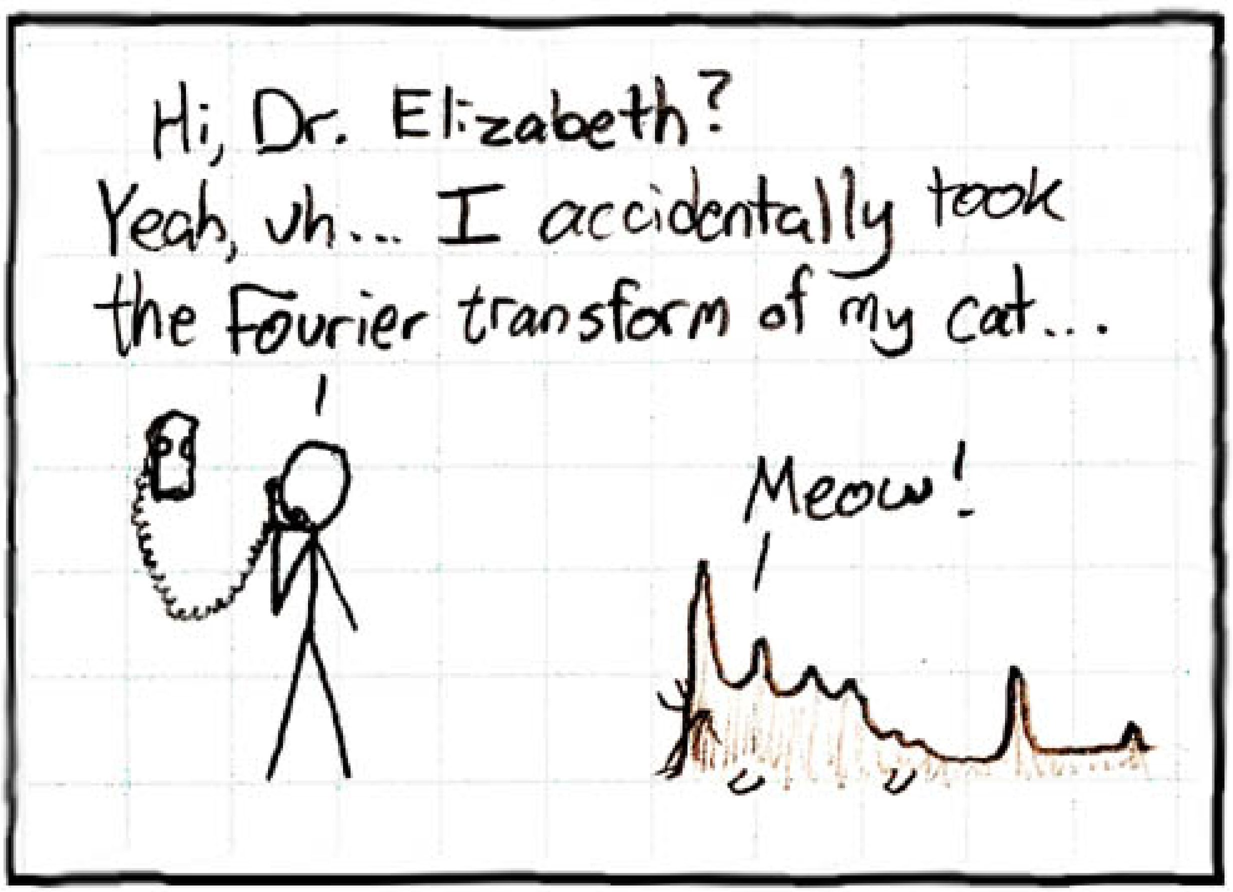
This probably should have it's own chapter. Anyways, time multiplexing on the LED cube exploits optical persistence (i.e. the fact that your eyes perceive a rapidly blinking light to be on). Instead of wiring two wires to every LED of an 8 by 8 plane, (128 wires, have fun soldering that!), we can use eight anodes and eight cathodes (16 wires) and just iterate through them quickly so that the lights look like they're on all the time (even though they aren't). I'm going to assume that working through the LED cube display function has given you a good idea of what this is about, so onwards!

9.7 Brightness

I'm not really sure where this belongs…possibly in another section for LEDs? Anyways, total visible light emitted is measured in lumens, and lux is lumens per meter squared (that is, light per unit area.)

Chapter 10

The Fourier Transform



Looks like it was Schroedinger's cat, not Gauss's.

10.1 The Big Idea

The Fourier transform is taking a function and writing it as the sum of sine waves.

10.2 Sum of Functions

Consider the functions $f_1 = 1$ and $f_2 = 2$. Then, if $f_3 = f_1 + f_2$, then $f_3 = 3$.

Consider the functions $f_4 = t + 5$ and $f_5 = 2t + 4$. Then, if $f_6 = f_4 + f_5$, then $f_6 = 3t + 9$.

The Fourier transform is no different. Essentially, what it says is that **any (repeating) function can be represented as the sum of sine waves**.

Just as above, we can pick f_{final} to be any arbitrary periodic (repeating) function, and we can represent it as the sum of $f_{s1}, f_{s2}, f_{s3}, \ldots$, where $f_{s1} = A_1 sin(2\pi 1t), f_{s2} = A_2 sin(2\pi 2t), f_{s2} = A_2 sin(2\pi 3t)$, etc.

10.3 Animations

A good way of intuiting the Fourier transform is through animations.

Note: these links and most other links in this document (like the table of contents) are clickable!

- https://commons.wikimedia.org/wiki/File:Fourier_series_square_wave_circles_animation.gif
- https://commons.wikimedia.org/wiki/File:Fourier_transform_time_and_frequency_domains.gif
- http://codepen.io/anon/pen/jPGJMK

Also, play around with the fourier applet (http://www.falstad.com/fourier/) if you haven't already. **If you have issues running the applet** on the website, you can click "Download zip archive of this applet" and then **run the .jar file**.

10.4 The Fourier Series

Say we have a function, a voltage that is a function of time, $v(t)$. What the Fourier transform says is that we can represent this as a sum of sinusoidal waves.

$$v(t) = a_0 + \sum_{n=1}^{\infty} \left(a_n \cos \frac{2\pi nt}{T} + b_n \sin \frac{2\pi nt}{T} \right)$$

That's a little complicated, isn't it? Let's take a slightly simpler model:

$$v(t) = a_0 + \sum_{n=1}^{\infty} \left(a_n \cos(2\pi nft) \right)$$

What happened here is that we removed the sine wave component and changed $\frac{1}{T}$ to f, or frequency. We're able to make this change because T is a period of time, and the reciprocal of a period of time is a frequency. In fact, this frequency, f or $\frac{1}{T}$ is quite important. It is the **fundamental frequency** of the Fourier series, the lowest frequency that the Fourier series can represent. If you have a signal that repeats every T seconds, the fundamental frequency is $\frac{1}{T}$, since the Fourier series must be able to represent that repetition every T seconds.

Let's analyze:

$$v(t) = \mathbf{a_0} + \sum_{n=1}^{\infty} \left(a_n \cos(2\pi nft) \right)$$

This constant term is the DC offset. However, we also conveniently know that $\cos(0) = 1$, so we can just fold this constant term into the summation, like so:

$$v(t) = \sum_{n=0}^{\infty} (a_n \cos(2\pi n f t))$$

Okay, I see this summation but I still have no clue what this means. Let's break it into words.

What this means is for every cosine wave ($\cos(2\pi n f t)$) that has a frequency that is a non-negative integer multiple (n) of the fundamental frequency (f), there is a corresponding amplitude (a_n) that determines how large of a contribution (if at all, since a_n can equal zero) that cosine wave has to the function being represented. Remember that we have the case of 0 frequency; this is the DC offset.

If you are familiar with linear algebra, a helpful (if somewhat inaccurate) analogy is that any periodic function can be represented as a linear combination of sine waves. In other words, sine waves form a basis for the space of all periodic functions.

10.5 Square Wave

A square wave can be represented as the sum of sine waves with frequencies that are an odd multiple of the fundamental frequency, with the amplitude decreasing as n increases.

$$v_{squarewave}(t) = \sum_{n=0}^{\infty} \left(\frac{1}{2n+1} \sin(2\pi(2n+1)ft) \right)$$

Note: The form $(2n + 1)$ is used to represent odd numbers.

10.6 Music

A natural application of the Fourier transform is to music. Musical notes (tones) are sine waves. (That is, if you connect a sine wave voltage source to a speaker, you will hear a single note.) The frequency of the sine wave determines a note's pitch, while the amplitude of the sine wave determines how loud it is.
An equalizer visualizer is a good analogy for the Fourier transform: you are breaking a sound wave into different notes. (If you are familiar with music, this is analogous to taking a chord and separating it into its constituent notes.)

Figure 10.1: A music visualizer.

Chapter 11

Devices: Capacitors and Inductors

Caps.

11.1 Capacitors

Capacitors are linear components that model the energy stored in electric fields.

1. A capacitor is formed by two terminals that are not connected, so the average current is 0.
2. Current through a capacitor is described by $i = C\frac{dV}{dt}$.
3. Capacitors store energy. This means that they can absorb energy from the circuit (charging capacitor) and provide that energy back to the circuit later (discharging capacitor).
4. Energy stored in a capacitor is described by $E = \frac{1}{2}CV^2$.
5. Capacitors try to **keep their voltage from changing rapidly** and look like **voltage sources** for short time periods. However, this means that current can change abruptly.

You might be wondering why there's no I-V curve here. That's because capacitors change over time (hence the $\frac{dV}{dt}$ in the capacitor characteristic equation).

11.2 Inductors

Inductors are linear components that model energy stored in magnetic fields. They are the analogue of capacitors for current.

1. An inductor is pretty much just a really long wire.
2. Voltage across an inductor is described by $V = L\frac{di}{dt}$.
3. Energy stored in an inductor is described by $E = \frac{1}{2}Li^2$.
4. Inductors try to **keep their current from changing rapidly** and look like **current sources** for short time periods. However, this means that voltage can change abruptly.

11.3 Capacitors in Parallel and in Series

Consider what happens when you have two capacitors in parallel.

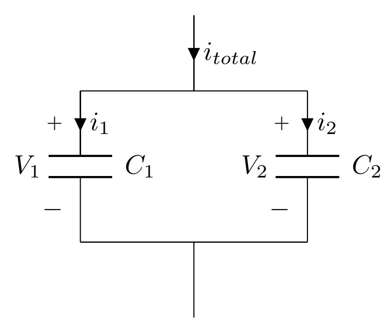


Figure 11.1: By KVL, $V_1 - V_2 = 0$, therefore $V_1 = V_2$

Recall that by KVL that two components in parallel have the same voltage. Therefore, since the voltages are the same at all points in time, $\frac{dV_1}{dt} = \frac{dV_2}{dt}$.

From KCL, we have that $i_{total} = i_1 + i_2 = C_1\frac{dV_1}{dt} + C_2\frac{dV_2}{dt} = C_1\frac{dV_1}{dt} + C_2\frac{dV_1}{dt} = (C_1 + C_2)\frac{dV_1}{dt}$.

This looks like $i = C\frac{dV}{dt}$ again, doesn't it?

This means that **capacitors in parallel add**. That is, $C_{parallel} = C_1 + C_2 + \ldots + C_n$.

Similarly, consider what happens when you have two capacitors in series.

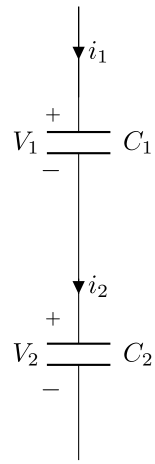


Figure 11.2: By KCL, $i_1 = i_2$

$i_1 = C_1 \frac{dV_1}{dt}$, $i_2 = C_2 \frac{dV_2}{dt}$. Thus $\frac{i_1}{C_1} = \frac{dV_1}{dt}$ and $\frac{i_2}{C_2} = \frac{dV_2}{dt}$.

Recall that by KCL, $i_1 = i_2$.

The total voltage across both capacitors is $\frac{dV_{total}}{dt} = \frac{dV_1}{dt} + \frac{dV_2}{dt} = \frac{i_1}{C_1} + \frac{i_2}{C_2} = \frac{i_1}{C_1} + \frac{i_1}{C_2} = i_1(\frac{1}{C_1} + \frac{1}{C_2})$.

Then $i_1 = \frac{1}{\frac{1}{C_1}+\frac{1}{C_2}} \frac{dV_{total}}{dt}$. This means that **capacitance in series is the reciprocal of the sum of the recipricols of the individual capacitances**. (You might recall that this is the same formula as for resistors in parallel.)

$$\frac{1}{C_{series}} = \frac{1}{C_1} + \frac{1}{C_2} + \ldots + \frac{1}{C_n}$$

11.4 Inductors in Parallel and in Series

Consider what happens when you have two inductors in parallel.

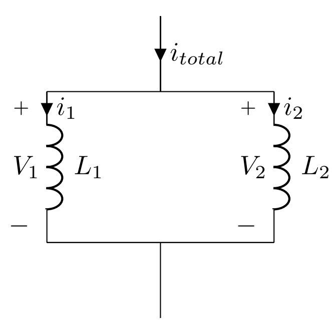


Figure 11.3: By KVL, $V_1 - V_2 = 0$, therefore $V_1 = V_2$

$V_1 = L_1 \frac{di_1}{dt}$, $V_2 = L_2 \frac{di_2}{dt}$. Thus $\frac{V_1}{L_1} = \frac{di_1}{dt}$ and $\frac{V_2}{L_2} = \frac{di_2}{dt}$.

Recall that by KVL, $V_1 = V_2$.

From KCL, we have that $i_{total} = i_1 + i_2$, thus $\frac{di_{total}}{dt} = \frac{di_1}{dt} + \frac{di_2}{dt} = \frac{V_1}{L_1} + \frac{V_2}{L_2} = \frac{V_1}{L_1} + \frac{V_1}{L_2} = V_1(\frac{1}{L_1} + \frac{1}{L_2})$.

Then $V_1 = \frac{1}{\frac{1}{L_1}+\frac{1}{L_2}} \frac{di_{total}}{dt}$. This means that **inductance in parallel is the reciprocal of the sum of the recipricols of the individual inductances**. (You might recall that this is the same formula as for resistors in parallel.)

$$\frac{1}{L_{parallel}} = \frac{1}{L_1} + \frac{1}{L_2} + \ldots + \frac{1}{L_n}$$

Similarly, consider what happens when you have two inductors in series.

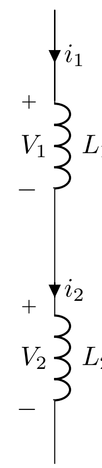


Figure 11.4: By KCL, $i_1 = i_2$

Recall that by KCL that two components in series have the same current. Therefore, since the voltages are the same at all points in time, $\frac{di_1}{dt} = \frac{di_2}{dt}$.

From KVL, we have that $V_{total} = V_1 + V_2 = L_1\frac{di_1}{dt} + L_2\frac{di_2}{dt} = L_1\frac{di_1}{dt} + L_2\frac{di_1}{dt} = (L_1 + L_2)\frac{di_1}{dt}$.

This looks like $V = L\frac{di}{dt}$ again, doesn't it?

This means that **inductors in series add**. That is, $L_{series} = L_1 + L_2 + \ldots + L_n$.

11.5 RC Circuits

You need to know the charge and discharge equations. The time constant for an RC circuit is $\tau = RC$.

Charge equation:

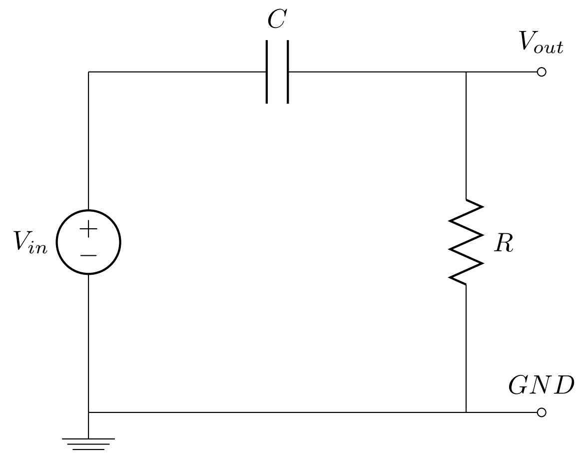


Figure 11.5: $V_{out} = V_{in}\left(1 - e^{-\frac{t}{RC}}\right)$

Discharge equation:

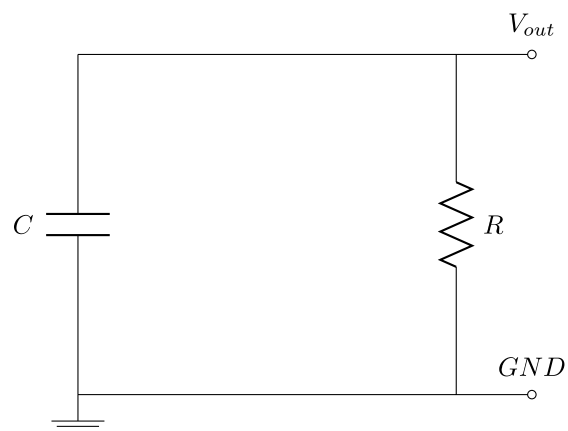


Figure 11.6: $V_{out} = V_{capacitor,initial}e^{-\frac{t}{RC}}$

Essentially, voltages in an RC circuit grow and decay exponentially. If you're curious about the derivation, check out chapter 6 of the reader.

11.6 RL Circuits

Similarly, two equations, except with current this time around. The time constant for an RL circuit is $\tau = \frac{L}{R}$.

Charge equation:

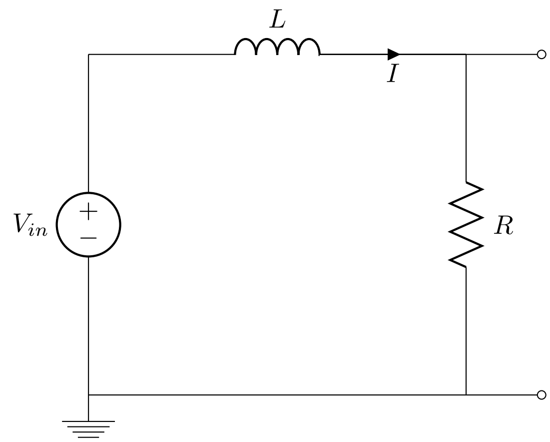


Figure 11.7: $I = \frac{V_{in}}{R}\left(1 - e^{-\frac{R}{L}t}\right)$

Discharge equation:

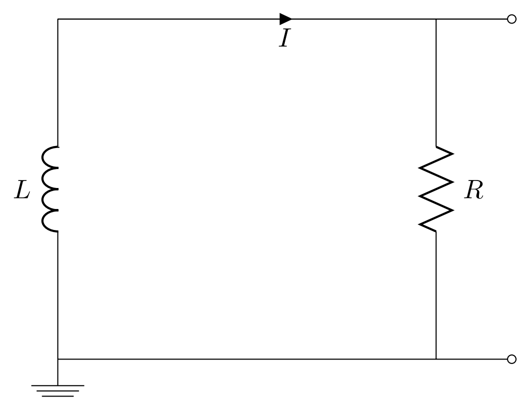
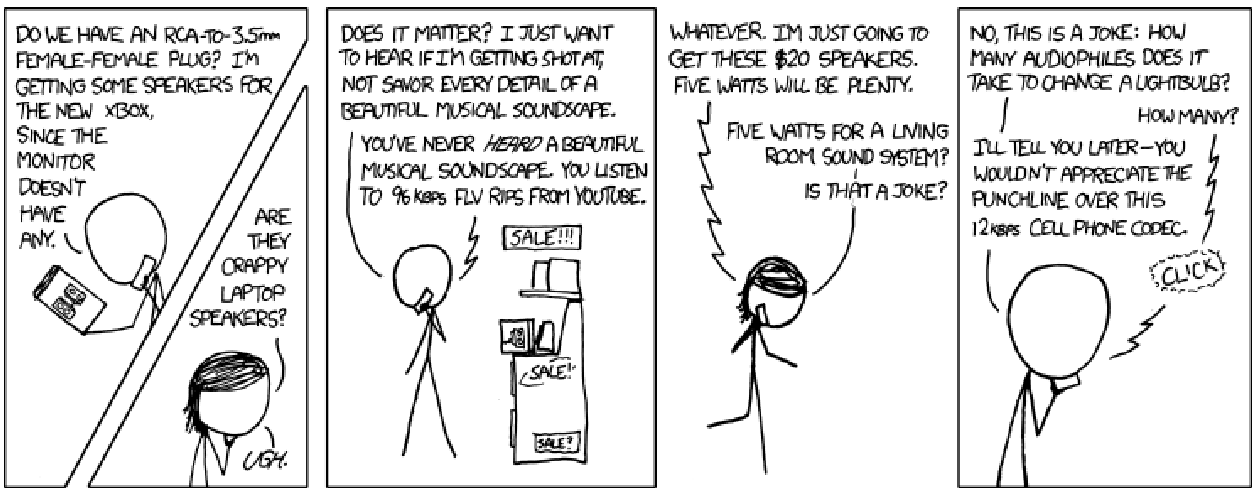


Figure 11.8: $I = I_{initial}e^{-\frac{R}{L}t}$

Similarly, currents in an LC circuit grow and decay exponentially. The derivation is covered in chapter 9 of the reader.

Chapter 12

Impedance and Filters



Filters. Like the audio kind.

12.1 Impedance

Impedance for a device is defined as the ratio of the sine wave voltage across the device to the sine wave current through the device.

Impedance is somewhat like resistance. For the purposes of this class, you can think of impedance as resistance for sinusoidal input waveforms.

For resistors, the impedance is simply

$$Z_{resistor} = R$$

For the purposes of this class, when calculating the gain of a circuit, you can replace capacitors and inductors with what are essentially variable-impedance resistors. (The impedance depends on the frequency of the signal being passed through.)

For capacitors, the impedance is

$$Z_{capacitor} = \frac{1}{2\pi FC}$$

For inductors, the impedance is

$$Z_{inductor} = 2\pi FL$$

Once you find the impedance of all of the components of the circuit, you can solve it in a manner similar to the way you solved circuits previously; just treat the impedances as resistances for sinusoidal input waves.

12.2 Filters

Filters are useful for isolating particular frequencies within a signal. This is particularly useful for removing high frequency (say, 60 Hz noise from the wall) or low frequency (say, a DC offset) noise from a circuit, but filters also have many other uses.
Filters generally fall into three broad catagories: low-pass, high-pass, and band-pass. They can be as simple as an RC circuit. The categories are based on the signals the filter lets **pass through**, i.e. the signals the filter **does not attenuate** (make much smaller). That is, filters are named for the frequencies for which they do not have a tiny gain for.

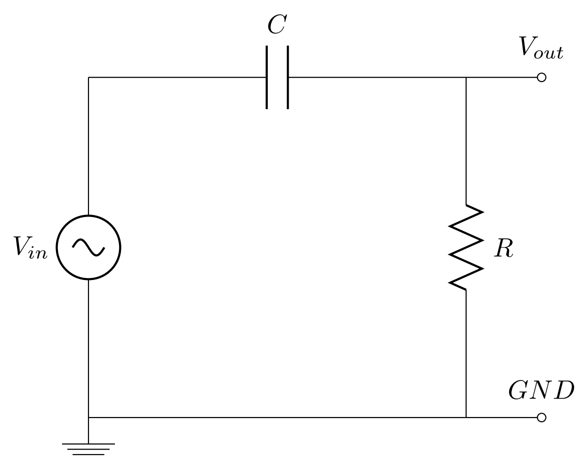


Figure 12.1: RC High Pass Filter

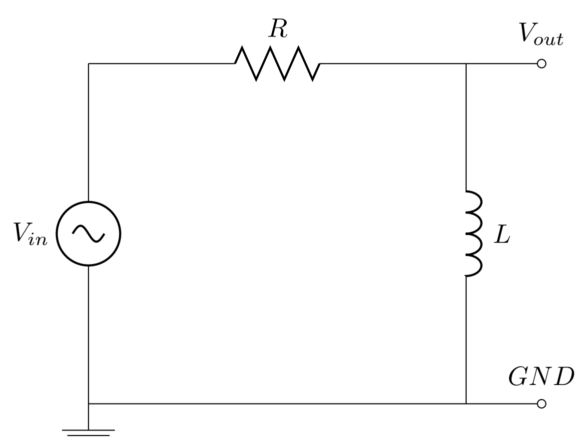


Figure 12.2: RL High Pass Filter

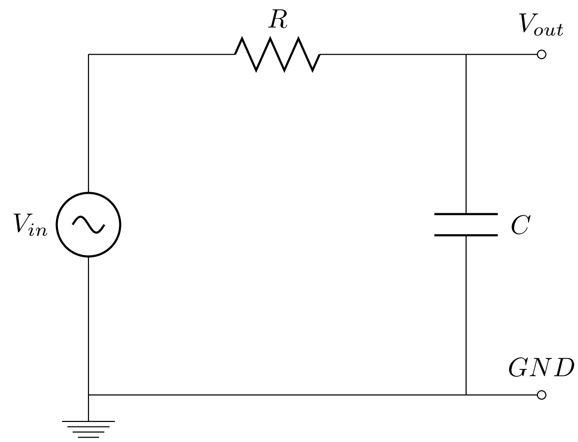


Figure 12.3: RC Low Pass Filter

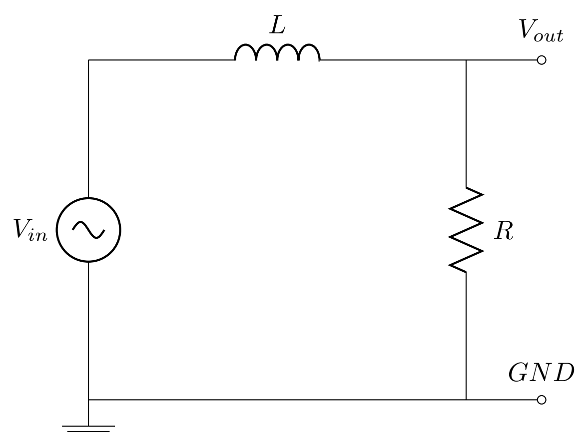


Figure 12.4: RL Low Pass Filter

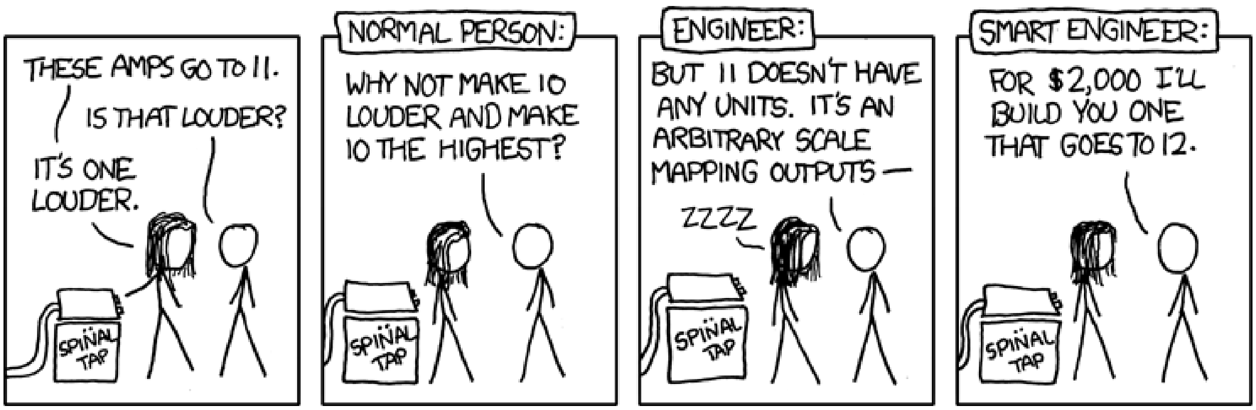
Band-pass filters are covered in chapter 7 of the reader.

12.3 Bode Plots

Steven has an amazing handout on Bode plots, and I will defer to that handout here: http://web.stanford.edu/class/engr40m/reader/bodeplots.pdf. (Note: the link is clickable!) If you have questions about gain, decibels, or anything of the sort, look there! Chapter 7 of the reader might also be helpful. There are really only a few components to drawing Bode plots in this class: whether the filter is a high-pass, low-pass, or band-pass filter, the gain (in decibels) of the plateau, the corner frequency, and the slope of the drop-off.

Chapter 13

Operational Amplifiers



Not that kind of amp. Not that kind of Amp either.

13.1 Operational Amplifiers

The output of an operational amplifier is given by:

$$V_{out} = A(V_{in}^{+} - V_{in}^{-})$$

Where A is the **gain of the operational amplifier** (not to be confused with the **gain of a filter** that contains an operational amplifier.). For ideal op-amps, $A = \infty$. For real op-amps, this value is usually around 10^6.

An important thing to note is that real op-amps cannot supply voltages (V_{out}) that are outside the range of its power supply voltages, i.e. $V-$ and $V+$. So if $A(V_{in}^{+} - V_{in}^{-})$ becomes greater than $V+$ or less than $V-$, the real op-amp will output a voltage that is close to $V+$ or $V-$, respectively. This is known as **railing** (or hitting the rails of the op-amp power supply).

13.2 Golden Rules

For **ideal op-amps** where **the output is fed back into the inverting (V_{in}^{-}) input**, there are really only two things you need to know to solve every circuit containing them (for this class).

1. The voltage at the two input terminals (V_{in}^{+} and V_{in}^{-}) are the same.
2. No current flows into either of the two input terminals.

Note: **Op-amps are not non-invering amps, inverting amps or instrumentation amps!**

13.3 Non-inverting Amplifiers, Inverting Amplifiers, etc.

Let me make this very clear: non-inverting amplifiers, inverting amplifiers, and instrumentation amplifiers are not different kinds of op-amps. Rather, they are **different circuits that contain op-amps**. That means that all non-inverting amplifiers, inverting amplifiers, and instrumentation amplifiers **contain components other than op-amps**, namely, resistors.

Confession time: I don't actually know how to build a non-inverting or inverting amplifier from memory. I vaguely know that non-inverting amplifiers have an output voltage that is a non-negative multiple of their input voltage, while inverting amplifiers have an output voltage that is a negative multiple of their input voltage. But using the golden rules for op-amps, it's relatively easy to derive the properties of an amplifier circuit. If that's not comforting to you, copy the op-amp circuits to your cheat sheet. Actually, copy the op-amp circuits to your cheat sheet anyway.[1] You already know everything you need to know to derive the properties - just take your basic circuit solving skills and the golden rules of op-amps to derive the properties by solving for voltages and currents. Essentially, **I don't particularly care what you call an amplifier circuit; I'm just going to apply my circuit solving skills with the op-amp golden rules to find** V_{out}.

13.4 Instrumentation Amplifiers

Instrumentation amplifiers are an interesting case of amplifier circuits. Instrumentation amplifiers have an output voltage that is similar to that of a differential amplifier circuit. For the instrumentation amplifier we use in this class, the gain of the instrumentation amplifier is programmable through a resistor, R_G. We can set the gain of the instrumentation amplifier to a much smaller value than an op-amp gain, say, on the order of 100.

$$V_{out} = A(V_{in}^{+} - V_{in}^{-}) + V_{REF}$$

However, very importantly, instrumentation amplifiers, unlike differential amplifier circuits, **draw no current from the inputs**. (Note that the **op-amp** in a differential amplifier circuit still does not draw current from its inputs, but the differential amplifier circuit also contains a resistor path from the input to ground, so the overall circuit can still draw current from the input.) Essentially, an instrumentation amplifier is a differential amplifier buffered by two non-inverting amplifiers to prevent drawing current from the input. See Chapter 8 of the reader for details on the analysis of an instrumentation amplifier, as well as its role in removing common node noise.

[1] Though actually, this will probably save you time on the exam.

Chapter 14

More Filters

Band-pass filter.

14.1 Why Op-Amp Filters?

You might be wondering why we need filters that contain op-amps. The reason is that RC/LC filters can't ever have a gain of more than 1. (Think about it this way: RC/LC filters are like voltage dividers for sine waves. Having a voltage divider means that you will always have a fraction of the total voltage, so you can't ever get a larger signal out than you put in.) However, op-amps allow us to generate voltages that are larger than our signals.[1]

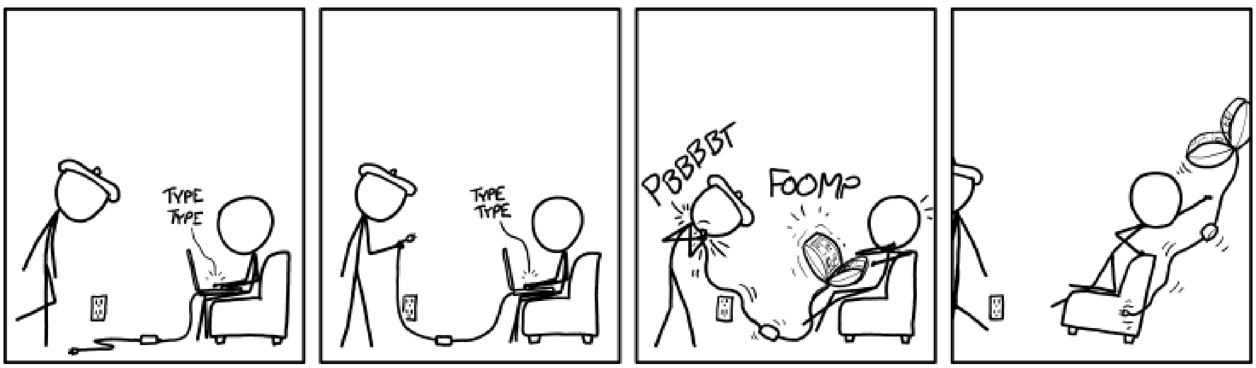
14.2 Understanding Op-Amp Filters

Op-amp filters can be analyzed exactly as the amplifier circuits in the previous chapter were, except that you now have capacitors and inductors in the mix. It's not a big deal though, just convert them to impedances and solve like you would any other impedance circuit. Thus, to solve op-amp filters, convert all capacitors and inductors to impedances, then use your circuit solving skills and the golden rules of op-amps in order to solve for the output voltage (which should be a function of frequency).

[1] The output voltage is limited by the op-amp power rails, however.

Chapter 15

Power Converters



If only.

15.1 Buck Converters

Buck converters aren't too bad. The general intuition behind them is that you can take a high voltage supply and turn it into a lower voltage supply. This is done by using a square wave with variable duty cycle such that the average voltage of the square wave equal to the desired lower voltage. However, the problem with this is that we now have a supply that has the correct average voltage, but at any given time, the output voltage is either too high or too low. Thus, we use inductors and capacitors to store energy and average out the output voltage. When the voltage from the supply is higher than our desired voltage, we store the extra energy in inductors and capacitors. When the voltage from the supply is lower than our desired voltage, we take the energy we stored in inductors and capacitors earlier to keep the output voltage at our desired voltage. You can also think of it as a filter for the square wave; we filter out the square part of the wave and are left with the DC offset.

Thus, V_{out} for a buck converter is:

$$V_{out} = \frac{T_{high}}{T_{cycle}} V_{in}$$

Note: There's a lot of analysis that I didn't cover here. Chapter 9 of the reader is your friend.

15.2 Boost Converters

Well then, boost converters are intuitively just buck converters flipped the other way around. Refer to chapter 9 of the reader for now…