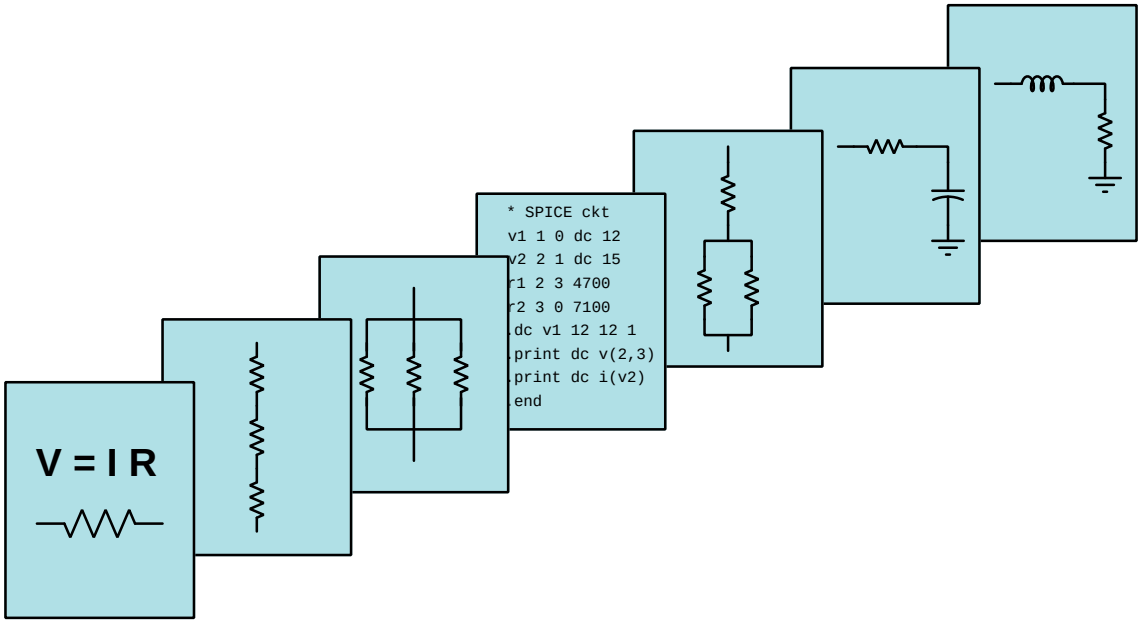
Modular Electronics Learning (ModEL) project



Polyphase AC

© 2018-2026 by Tony R. Kuphaldt – under the terms and conditions of the Creative Commons Attribution 4.0 International Public License

Last update = 11 March 2026

This is a copyrighted work, but licensed under the Creative Commons Attribution 4.0 International Public License. A copy of this license is found in the last Appendix of this document. Alternatively, you may visit `http://creativecommons.org/licenses/by/4.0/` or send a letter to Creative Commons: 171 Second Street, Suite 300, San Francisco, California, 94105, USA. The terms and conditions of this license allow for free copying, distribution, and/or modification of all licensed works by the general public.

Contents

1 Introduction **3**
- 1.1 Recommendations for students . . . 3
- 1.2 Challenging concepts related to polyphase AC circuits . . . 5
- 1.3 Recommendations for instructors . . . 6

2 Case Tutorial **7**
- 2.1 Example: wye-wound versus delta-wound motors . . . 8
- 2.2 Example: three-phase motor calculations with imperfect efficiency . . . 9
- 2.3 Example: wye-wound generator and wye-connected resistive load . . . 10
- 2.4 Example: delta-wound generator and delta-connected resistive load . . . 11
- 2.5 Example: wye-wound generator and delta-connected resistive load . . . 12
- 2.6 Example: delta-wound generator and wye-connected resistive load . . . 13
- 2.7 Example: experimental motor-generator set . . . 14
- 2.8 Example: phase rotation test circuit . . . 15

3 Tutorial **17**
- 3.1 Multi-phase AC generators . . . 18
- 3.2 Wye and Delta connections . . . 21
- 3.3 Wye and Delta grounding . . . 29
- 3.4 Three-phase rectification . . . 32
- 3.5 Harmonics in three-phase networks . . . 34

4 Historical References **43**
- 4.1 Prototype electrical power transmission system . . . 44

5 Derivations and Technical References **47**
- 5.1 Derivation of $\sqrt{3}$ factor . . . 48
- 5.2 Electrical safety . . . 51

6 Animations **59**
- 6.1 Rotating magnetic field animated . . . 60

7 Programming References **85**
7.1 Programming in C++ . . . 86
7.2 Programming in Python . . . 90
7.3 Modeling Wye and Delta networks using C++ . . . 95

8 Questions **99**
8.1 Conceptual reasoning . . . 103
8.1.1 Reading outline and reflections . . . 104
8.1.2 Foundational concepts . . . 105
8.1.3 Similar voltages and currents . . . 107
8.1.4 Testing for single-phase or three-phase . . . 108
8.2 Quantitative reasoning . . . 110
8.2.1 Miscellaneous physical constants . . . 111
8.2.2 Introduction to spreadsheets . . . 112
8.2.3 Total power dissipation . . . 115
8.2.4 Balanced Delta source and Delta load . . . 116
8.2.5 Balanced Delta source and Wye load . . . 117
8.2.6 Balanced Wye source and Delta load . . . 118
8.2.7 Three-phase versus single-phase power transmission . . . 119
8.2.8 Three-phase electric motor with CT current measurement . . . 120
8.2.9 Direct-coupled and transformer-coupled loads . . . 121
8.2.10 Wye source and Wye load with and without line fault . . . 122
8.2.11 Balanced Wye source with unbalanced Delta load . . . 123
8.2.12 Balanced Wye source with unbalanced Delta load and line fault . . . 124
8.2.13 Three-phase simulation program . . . 125
8.3 Diagnostic reasoning . . . 127
8.3.1 Winding faults within three-phase motors . . . 128

9 Projects and Experiments **129**
9.1 Recommended practices . . . 129
9.1.1 Safety first! . . . 130
9.1.2 Other helpful tips . . . 132
9.1.3 Terminal blocks for circuit construction . . . 133
9.1.4 Conducting experiments . . . 137
9.1.5 Constructing projects . . . 141
9.2 Experiment: SPICE modeling of a three-phase AC circuit . . . 142

A Problem-Solving Strategies **145**

B Instructional philosophy **147**
B.1 First principles of learning . . . 148
B.2 Proven strategies for instructors . . . 149
B.3 Proven strategies for students . . . 151
B.4 Design of these learning modules . . . 152

C Tools used **155**

D Creative Commons License **159**

E References **167**

F Version history **169**

Index **171**

Chapter 1

Introduction

1.1 Recommendations for students

Polyphase electric circuits are standard for transmitting and distributing large amounts of electrical energy over long distances, and it is the primary form of electric circuit encountered in industrial applications. *Three*-phase circuits are particularly popular, and as such comprise an essential topic of study for anyone seeking to understand electricity as used in the modern world.

The basic idea behind polyphase electric circuits is that multiple AC sources out-of-step with each other work together to deliver energy to loads. Polyphase circuits tend to deliver energy more continuously than single-phase electric circuits, and certain types of loads such as electric motors exhibit better operating characteristics than their single-phase counterparts.

Important concepts related to polyphase circuits include **alternating current** (AC), **electromagnetic induction**, **phase angle**, **frequency**, **phasors**, **phase sequence**, **transformers**, **delta** versus **wye** networks, **line** versus **phase** quantities, **Kirchhoff's Voltage Law**, **Kirchhoff's Current Law**, **RMS** quantities, **Ohm's Law**, the **Conservation of Energy**, **grounding**, **hot** versus **neutral** conductors, and **rectification**.

Here are some good questions to ask of yourself while studying this subject:

- How might an experiment be designed and conducted to explore the concept of phase versus line quantities in a three-phase AC system? What hypothesis (i.e. prediction) might you pose for that experiment, and what result(s) would either support or disprove that hypothesis?
- What are some practical applications of polyphase AC circuits?
- Why is AC generally preferred over DC for the transmission and use of electric power?
- Why is *polyphase* AC generally preferred over single-phase AC for the transmission and use of electric power?
- Which fundamental principles of electric circuits are useful for discerning equal line and phase quantities in a three-phase network?

- Which fundamental principles of electric circuits are useful for discerning *un*equal line and phase quantities in a three-phase network?
- How is phase rotation reversed in a three-phase electric circuit?
- Why are fuses always located on a "hot" conductor and never on a "neutral" or "ground" conductor?
- Why is an electric power system always grounded at the source and never at the load?
- Why do we find 120/208 Volt three-phase systems and not 120/240 Volt three-phase systems?
- What is the purpose of *rectification*?
- Why is three-phase rectification preferred over single-phase rectification?

1.2 Challenging concepts related to polyphase AC circuits

The following list cites concepts related to this module's topic that are easily misunderstood, along with suggestions for properly understanding them:

- **Delta versus Wye network properties** – a very common tendency among students is to memorize formulae for line and phase quantities in three-phase AC networks without actually understanding why those formulae are correct. Viewing line and phase elements as being either in series or in parallel with other elements is extremely helpful here, leveraging previously learned foundational concepts. For example, line and phase currents must be equal in a Wye network because each phase element of a Wye network is (obviously) connected in series with a respective line; line and phase voltages must be equal for Delta networks because each phase element of a Delta network is (obviously) connected in parallel with two lines. Once identical line/phase quantities are identified, unequal line/phase quantities must always differ by a factor of $\sqrt{3}$ (in a *balanced* three-phase system) with the line quantity always being larger than the phase quantity.

1.3 Recommendations for instructors

This section lists realistic student learning outcomes supported by the content of the module as well as suggested means of assessing (measuring) student learning. The outcomes state what learners should be able to do, and the assessments are specific challenges to prove students have learned.

- **Outcome** – Demonstrate effective technical reading and writing

  Assessment – Students present their outlines of this module's instructional chapters (e.g. Case Tutorial, Tutorial, Historical References, etc.) ideally as an entry to a larger Journal document chronicling their learning. These outlines should exhibit good-faith effort at summarizing major concepts explained in the text.

  Assessment – Students show how quantitative results were obtained by the author in the Tutorial chapter's examples.

  Assessment – Students present their "Conceptual Toolbox" Theory Journal entries listing important concepts applied in the Tutorial text(s). A "Conceptual Toolbox" serves as inspiration when solving problems, each concept suggesting a potential problem-solving action. Challenge students to collaboratively generate a Conceptual Toolbox for the day's topic during the class session and then use it when solving problems together, thus fostering the habit of reasoning from principles rather than recalling procedures!

- **Outcome** – Apply series and parallel concepts to three-phase networks

  Assessment – Identify electrical quantities that are guaranteed to be equal to each other in given three-phase networks; e.g. pose problems in the form of the "Similar voltages and currents" Conceptual Reasoning question.

- **Outcome** – Determine voltages, currents, and/or powers in three-phase circuits

  Assessment – Calculate various voltages, currents, and power values for balanced three-phase circuits given component values; e.g. pose problems in the form of the "Balanced Delta source and Delta load", "Balanced Delta source and Wye load", and "Balanced Wye source and Delta load" Quantitative Reasoning questions.

- **Outcome** – Independent research

  Assessment – Read and summarize in your own words reliable historical documents on the subject of polyphase electrical power systems.

Chapter 2

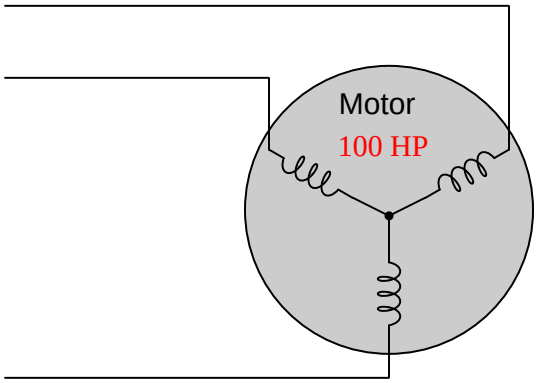
Case Tutorial

The idea behind a *Case Tutorial* is to explore new concepts by way of example. In this chapter you will read less presentation of theory compared to other Tutorial chapters, but by close observation and comparison of the given examples be able to discern patterns and principles much the same way as a scientific experimenter. Hopefully you will find these cases illuminating, and a good supplement to text-based tutorials.

These examples also serve well as challenges following your reading of the other Tutorial(s) in this module – can you explain *why* the circuits behave as they do?

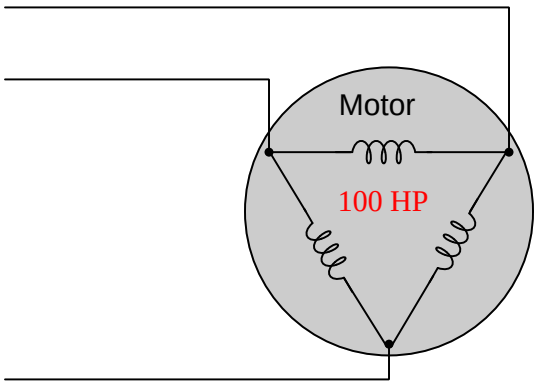
2.1 Example: wye-wound versus delta-wound motors

100 horsepower wye-wound electric motor energized by 460 Volts (line voltage) assuming 100% motor efficiency and a power factor of 1:



$I_{line} = 93.63$ A
$V_{phase} = 265.6$ V (i.e. across each of the motor's three windings)
$I_{phase} = 93.63$ A (i.e. through each of the motor's three windings)

100 horsepower delta-wound electric motor energized by 460 Volts (line voltage) assuming 100% motor efficiency and a power factor of 1:



$I_{line} = 93.63$ A
$V_{phase} = 460$ V (i.e. across each of the motor's three windings)
$I_{phase} = 54.06$ A (i.e. through each of the motor's three windings)

2.2 Example: three-phase motor calculations with imperfect efficiency

A three-phase AC electric motor delivers 1250 lb-ft of torque at 850 RPM to a mechanical load. Assuming a line voltage of 480 Volts and an energy-conversion efficiency of 92%, we may calculate the following:

Converting torque (lb-ft) and speed (RPM) values into horsepower:

$$P = \frac{S\tau}{5252.113} = \frac{(850 \text{ RPM})(1250 \text{ lb-ft})}{5252.113} = 202.30 \text{ HP}$$

Converting horsepower into Watts:

$$P = 202.30 \text{ HP} \times 745.7 \frac{\text{W}}{\text{HP}} = 150.85 \text{ kW}$$

This is the mechanical power output by the motor, expressed in units of Watts. The motor is 92 percent efficient ($\eta = 0.92$), which means the electrical power *input* to the motor must be greater:

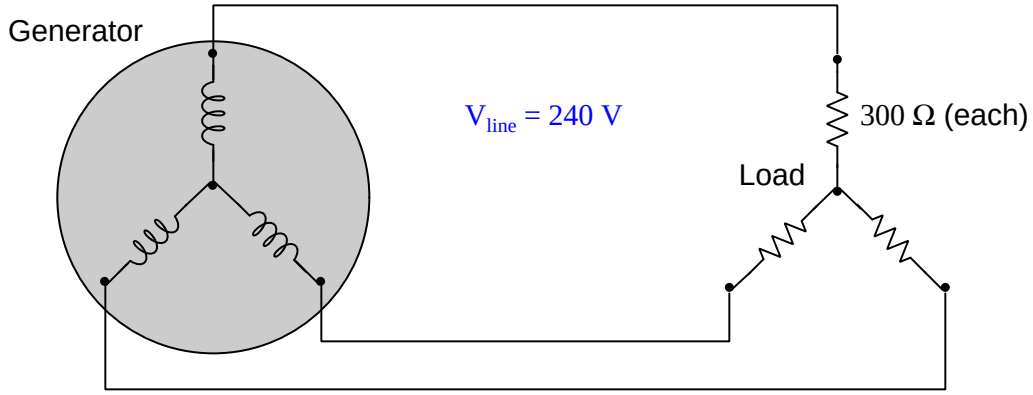
$$P_{out} = P_{in}\eta$$

$$P_{in} = \frac{P_{out}}{\eta} = \frac{150.85 \text{ kW}}{0.92} = 163.97 \text{ kW}$$

Given a line voltage of 480 Volts, the line current must be 197.228 Amperes:

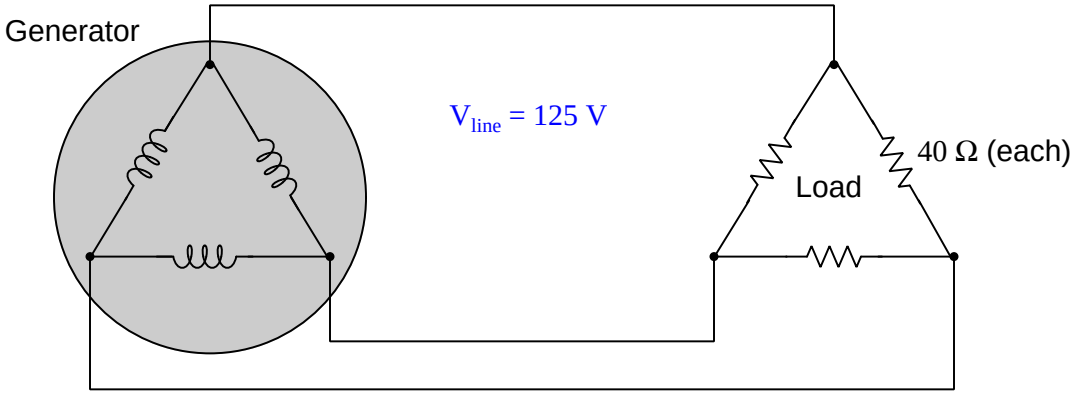
$$I_{line} = \frac{P_{total}}{V_{line}\sqrt{3}} = \frac{163.97 \text{ kW}}{(480 \text{ V})\sqrt{3}} = 197.228 \text{ A}$$

2.3 Example: wye-wound generator and wye-connected resistive load



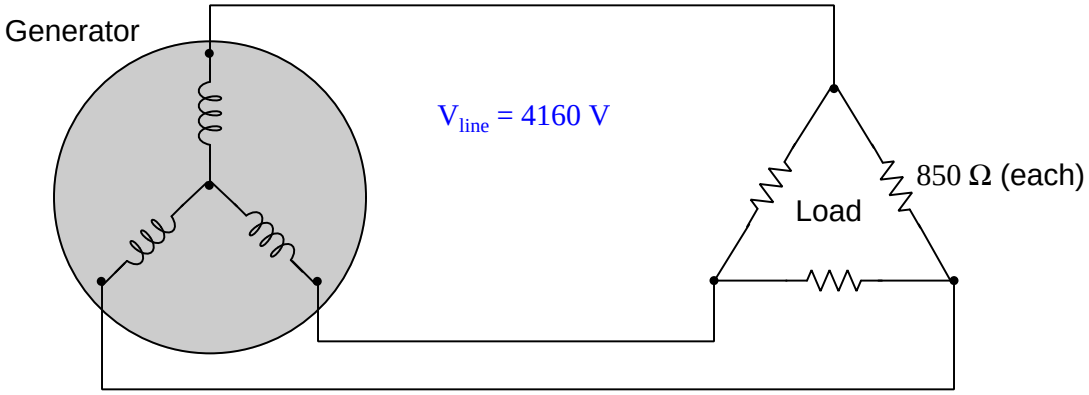
V_{line} = 240 Volts
I_{line} = 0.46188 Amperes
V_{phase} (generator) = 138.564 Volts
I_{phase} (generator) = 0.46188 Amperes
V_{phase} (load) = 138.564 Volts
I_{phase} (load) = 0.46188 Amperes
P_{total} = 192 Watts

2.4 Example: delta-wound generator and delta-connected resistive load



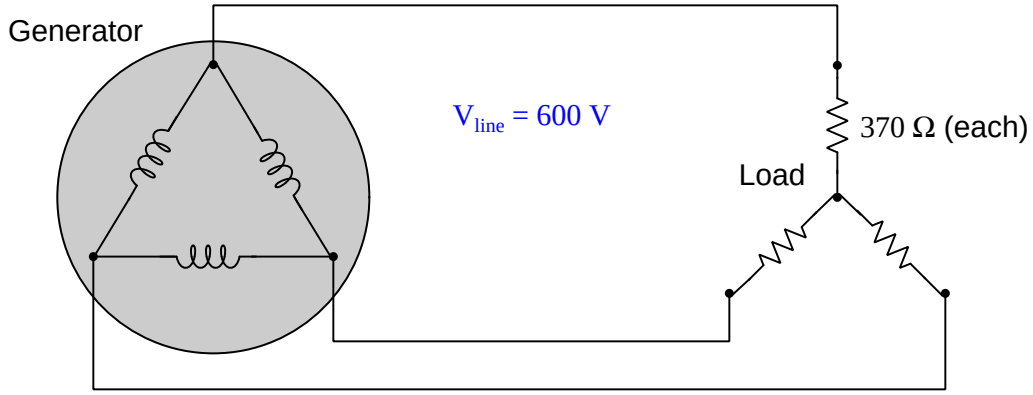
V_{line} = 125 Volts
I_{line} = 5.4127 Amperes
V_{phase} (generator) = 125 Volts
I_{phase} (generator) = 3.125 Amperes
V_{phase} (load) = 125 Volts
I_{phase} (load) = 3.125 Amperes
P_{total} = 1171.9 Watts

2.5 Example: wye-wound generator and delta-connected resistive load



V_{line} = 4160 Volts
I_{line} = 8.477 Amperes
V_{phase} (generator) = 2401.8 Volts
I_{phase} (generator) = 8.477 Amperes
V_{phase} (load) = 4160 Volts
I_{phase} (load) = 4.894 Amperes
P_{total} = 61.079 kiloWatts

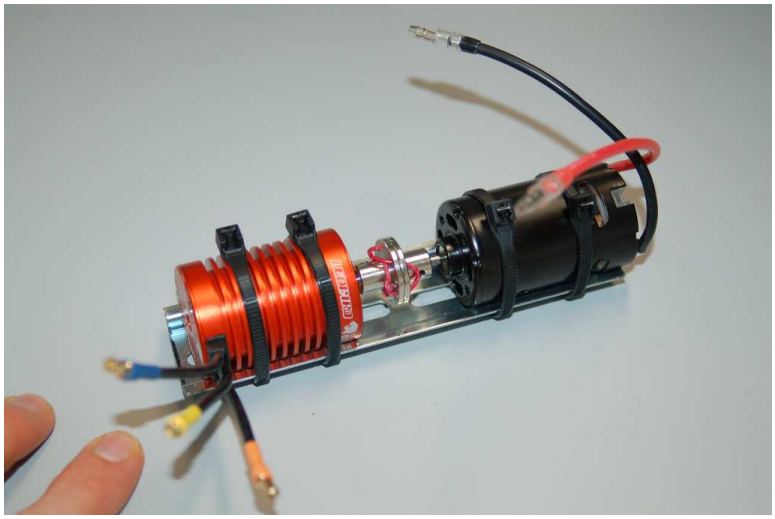
2.6 Example: delta-wound generator and wye-connected resistive load



V_{line} = 600 Volts
I_{line} = 0.9362 Amperes
V_{phase} (generator) = 600 Volts
I_{phase} (generator) = 0.5405 Amperes
V_{phase} (load) = 346.4 Volts
I_{phase} (load) = 0.9362 Amperes
P_{total} = 972.97 Watts

2.7 Example: experimental motor-generator set

Three-phase electricity is not available in most residences, and this means performing three-phase AC electrical experiments at home is usually considered impossible. However, it is possible to build a simple motor-generator set to produce safe (i.e. low-voltage, low-current) three-phase AC electricity suitable for running small experiments:



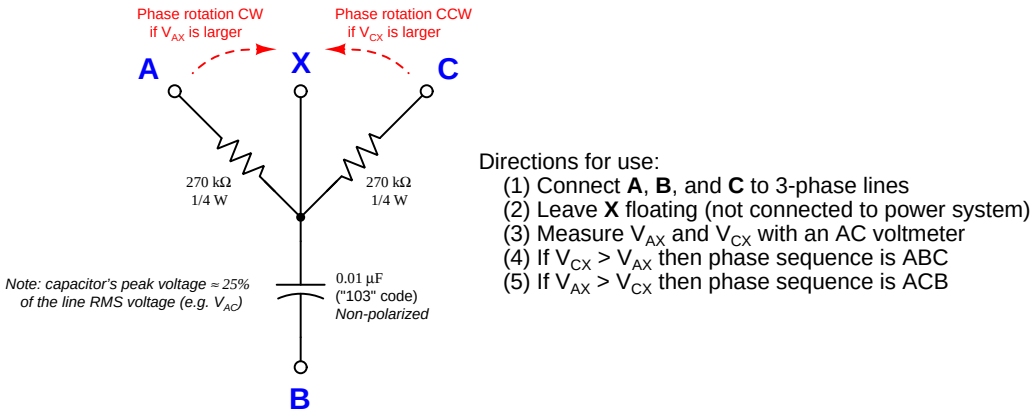
In this example, we are using two electric motors designed for use in radio-controlled (RC) hobby vehicles. The motor on the right is a standard brush-type DC permanent-magnet motor. The motor on the left is a *brushless* motor with the same diameter case (36 millimeters)[1]. A brushless motor is really a synchronous three-phase AC motor with a permanent-magnet rotor, which means it naturally produces three-phase AC simply by rotating the shaft. This size of motor rests securely within the recess of a 35 mm DIN rail, and when lashed in place with cable ties are nicely aligned shaft-to-shaft. A pair of flange couplings attach to the motor shafts using set-screws, and those flanges are coupled by lacing a 22 AWG copper wire through the four holes, making a simple shaft-to-shaft coupling so that the brush-type motor is able to spin the brushless motor (generator).

Brushless motors made for RC vehicle use are rated by their number of poles (just like any other AC motor) and the shaft speed per Volt. Most brushless motors are four-pole which means they must rotate at 1800 RPM to output a frequency of 60 Hz. Two-pole motors (like the one shown) rotate at 3600 RPM to produce 60 Hz. The RPM/Volt rating is usually specified as a number followed by the letters "KV". In this example the orange-colored brushless motor is 3000KV which means 3000 RPM per Volt. Spinning at 3600 RPM was found to generate approximately 0.88 Volts RMS between any two of the three wires, which admittedly is not much, but which may be stepped up through a three-phase transformer bank. Greater three-phase voltage could be obtained by using a brushless motor with a lesser "KV" number (representing more turns of wire on each stator pole and/or a stronger permanent magnet on the rotor). The black-colored brush-type motor required 4.75 Volts to drive the other motor to that speed.

[1] Both the "3650" can size and "550" can size RC motors have the exact same outer diameter. In this particular example the brush-type motor has a 550 "can" and the brushless has a 3650 "can".

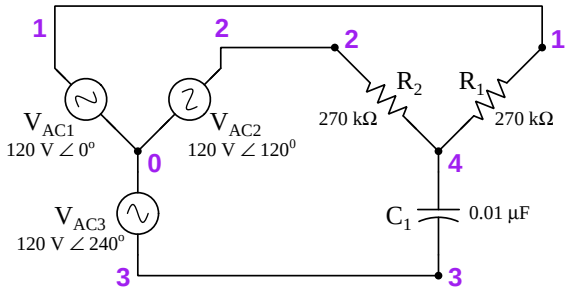
2.8 Example: phase rotation test circuit

Multi-channel oscilloscopes work well to indicate phase rotation in three-phase circuits, but other techniques exist as well for determining the ABC versus CBA rotation of a three-phase circuit. The schematic below shows how you may build a three-phase voltage divider resistor network to create a three-phase voltage divider for safely testing phase rotation in cases where the line voltage could damage the test instrument's inputs. This Wye-connected resistor network also provides a "ground" reference if the power system lacks one:



The natural phase shift offered by the capacitor interacts with the shifted phases at terminals A and C to produce a higher current through one resistor than through the other. As such, this is a great example of an *unbalanced* three-phase network, phase voltages and currents being intentionally unequal.

Here we see a SPICE analysis of this rotation-testing circuit:



SPICE source file (netlist):

```
* Phase rotation tester made of two resistors and one capacitor
* Phase rotation in this case is 2-1-3 (a.k.a. 3-2-1)
VAC1 1 0 ac 120 0 sin
VAC2 2 0 ac 120 120 sin
VAC3 3 0 ac 120 -120 sin
R1 1 4 270000
R2 2 4 270000
C1 3 4 1E-08 IC=0
.OPTION ITL5=0 NUMDGT=6
.AC LIN 1 60 60
.PRINT AC VM(1,4) VM(2,4)
.END
```

SPICE analysis summary:

```
--------------------------------------------------------------------------------
Index   frequency       mag(v(1)-v(4))  mag(v(2)-v(4))
--------------------------------------------------------------------------------
0       6.000000e+01    4.839760e+01    1.805242e+02
```

With the phase rotation such that node 2 leads node 1 (both with respect to ground which is always node 0 in SPICE), the voltage between node 2 and the center-point of the Wye (node 4) is larger than the voltage between node 1 and the center-point. In this particular analysis, `mag(v(2)-v(4))` is 180.5242 Volts while `mag(v(1(-v(4))` is 48.39760 Volts.

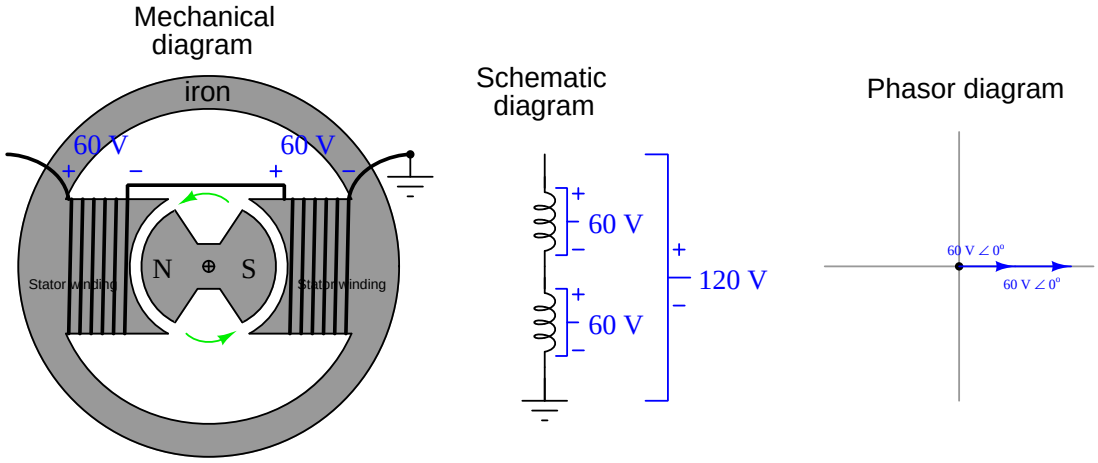
Chapter 3

Tutorial

3.1 Multi-phase AC generators

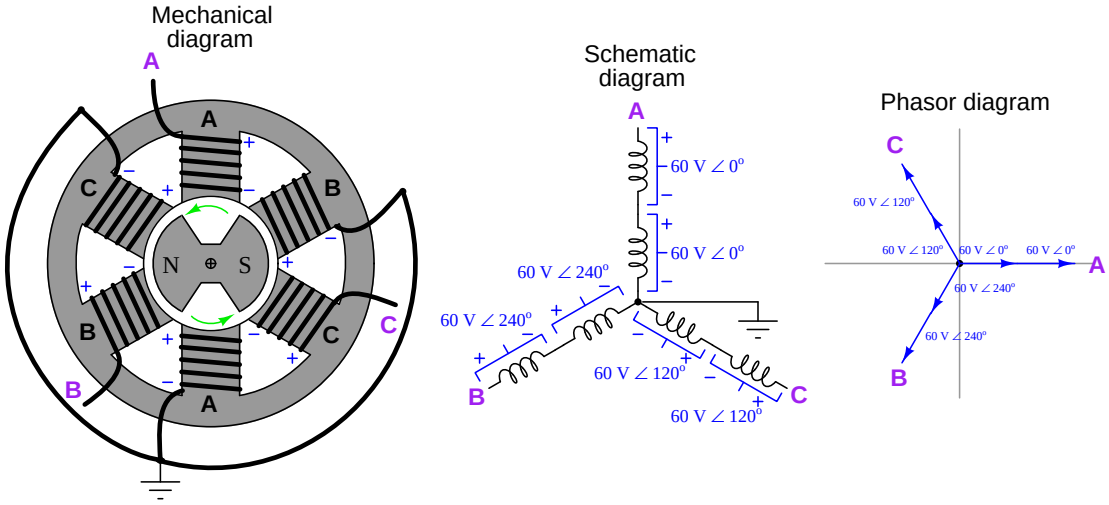
The term *polyphase* means "many phases" which describes a form of AC electrical system where multiple sinusoidal voltages exist out-of-step with each other. The most common form of polyphase AC power in industry is *three-phase*, where the respective voltages and currents are phase-shifted from one another by 120^o, but all polyphase systems share similar traits. This tutorial will focus on three-phase AC circuits because this is by far the most common form of polyphase circuit.

A simple *alternator* or *AC generator* is nothing more than a magnetized rotor spinning between a pair of electromagnetic poles, the stationary wire coils ("stator windings") developing AC voltage as the spinning rotor's magnet passes by:



In this particular example each stator winding outputs 60 Volts AC, and since they're connected in series-aiding fashion the total output voltage will be 120 Volts. This machine is properly called a *single-phase* generator, because all its stator winding voltages are in-phase with each other. Recall that AC polarity symbols (+ and −) indicate the actual polarity of each voltage during the positive half-cycle of the alternating waveform.

A much more common design of AC generator uses *three* sets of stator windings to generate three AC voltages phase-shifted from one another by 120^o. The reason these three AC voltages are not in-phase with each other is precisely because the three stator poles are not physically aligned with each other, which means the magnetic poles of the spinning rotor will pass by each stator pole pair at different times:



As the generator's rotor spins counter-clockwise, stator winding pair "A" will reach its positive peak value before stator winding pair "B", and that before stator winding pair "C". This A-B-C order is called the *phase sequence* or *phase rotation* of the generator, and each stator winding set is simply called a "phase". Phase rotation is easiest to apprehend by viewing the schematic diagram, noting the phase order as you rotate counter-clockwise around the schematic (in the same direction as the rotor): at the top is "A" (arbitrarily defined as having a 0^o phase angle) followed by "B" 120 degrees later (-120^o or 240^o phase-shifted from "A") followed by "C" 120 degrees after that.

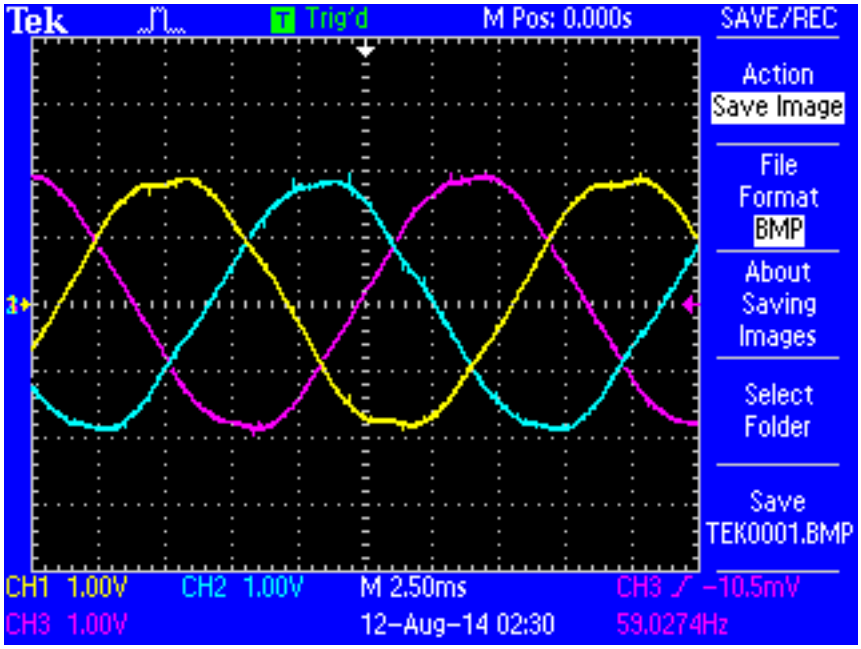
The phasor diagram also declares an A-B-C phase rotation: the blue vectors represent stator voltages at that instant in time when stator winding pair "A" is at its positive peak value (i.e. 0^o on a cosine wave), and in a phasor diagram it is the vectors themselves that rotate counter-clockwise so that a stationary observer would notice vector tips passing by in an A-B-C order.

If we were to connect an AC voltmeter between any one phase terminal on this generator and ground, it would register 120 Volts: the sum of the individual stator winding voltages (60 Volts each) comprising that phase. However, if we were to connect an AC voltmeter between any two of the un-grounded terminals on this three-phase generator, it would register 207.85 Volts. For example, connecting the red lead of the voltmeter to terminal A and the black lead to terminal B gives us this difference of potential:

$$V_A - V_B = (120 \text{ V}\angle 0^o) - (120 \text{ V}\angle 240^o) = 207.84 \text{ V}\angle 30^o$$

This is the length and direction of a resultant vector starting at B and ending at A on the phasor diagram.

The following oscilloscope screenshot shows the output of a three-phase generator, each channel of the oscilloscope connected across one stator winding set (e.g. Channel 1 measuring phase "A", Channel 2 measuring phase "B", and Channel 3 measuring phase "C"):



In this oscillograph we clearly see a 120^o phase shift between successive waveforms, and we may also discern the phase rotation of the generator: surveying the three sinusoids from left to right (the forward direction of time) on the oscillograph, we see channel 1 (Yellow) reaches its positive peak 120 degrees before channel 2 (Cyan), which reaches its positive peak 120 degrees before channel 3 (Magenta): a 1-2-3 phase rotation. It should be noted that a 1-2-3 phase rotation is synonymous with a 2-3-1 or a 3-1-2 phase rotation, since we may find any of these triads within the sequence as written in longer form: 1-2-3-1-2-3-1-2-3.

Phase rotation may be reversed by either reversing the alternator's shaft's direction of rotation, or by exchanging the number labels for any two out of three phases. The first method of phase rotation reversal requires no explanation: if the alternator shaft spins in the opposite direction, the sequence must switch from 1-2-3 to 3-2-1. The second method is a bit less intuitive, but may be easily seen by taking the long "1-2-3" sequence previously mentioned (1-2-3-1-2-3-1-2-3) and re-writing it with any two numbers swapped throughout the entire sequence:

Swapping 1 and 2: `2-1-3-2-1-3-2-1-3` *(equivalent to 3-2-1)*

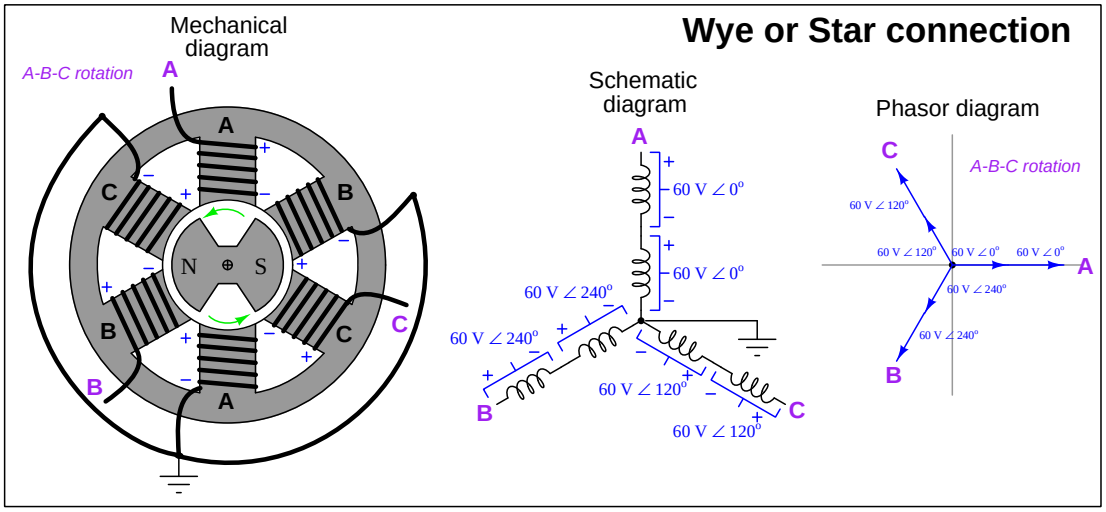
Swapping 2 and 3: `1-3-2-1-3-2-1-3-2` *(equivalent to 3-2-1)*

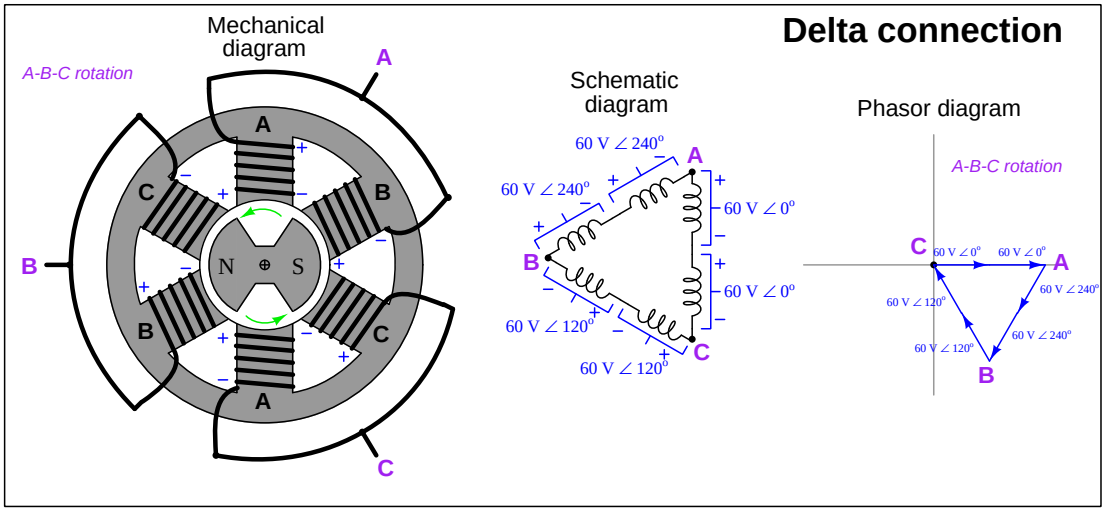
Swapping 1 and 3: `3-2-1-3-2-1-3-2-1` *(obviously 3-2-1)*

No matter which two phases we swap, the original 1-2-3 rotation turns into a 3-2-1 rotation.

3.2 Wye and Delta connections

The three stator winding sets in a three-phase AC generator may be connected in either a "Wye" (also called a "Star") or a "Delta" fashion, each with its own characteristics:

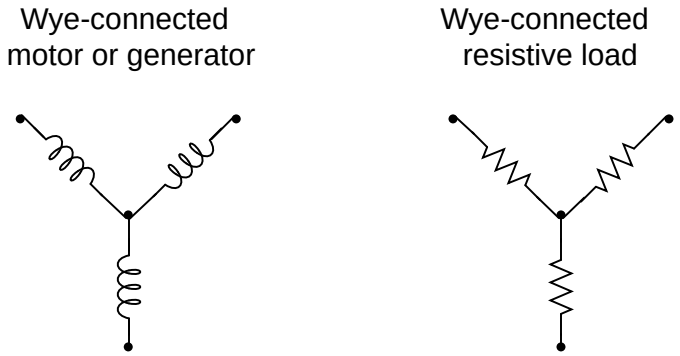




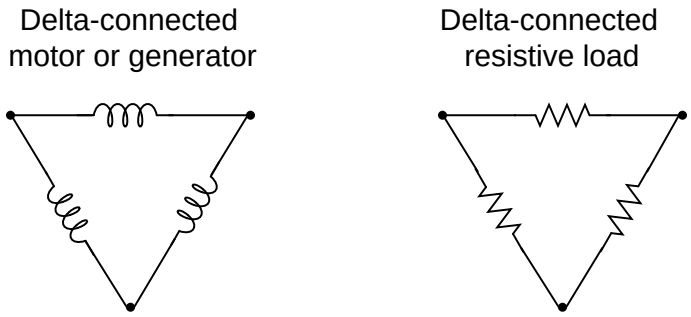
Please note the orientations of the polarities on the schematic diagrams and of the voltage vectors in the phasor diagrams, as they are not arbitrary. A Wye-connected generator's stator windings must join together at a point sharing a common polarity: in the example show above, all the − terminals join at the center of the Wye. A Delta-connected generator's stator windings must join together such that the phasor sum of all stator winding voltages is zero (− joins to + at each node). These proper

connections are not just important for generator windings, but also for three-phase *transformer*[1] windings as well.

It is customary to represent polyphase generators in the simplest possible form on a schematic diagram, using a single coil symbol to denote each "phase" of the stator winding rather than separately denote each portion of each stator winding as previously shown. Examples of Wye-connected sources and loads appears in the following diagram:



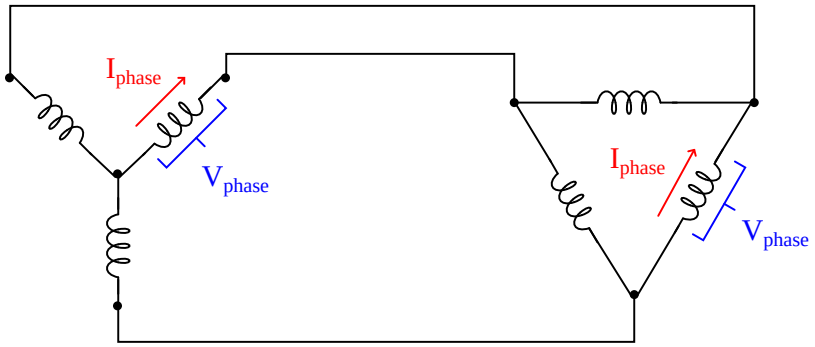
By contrast, a "Delta-connected" device has its three elements joined as the sides of a triangle:



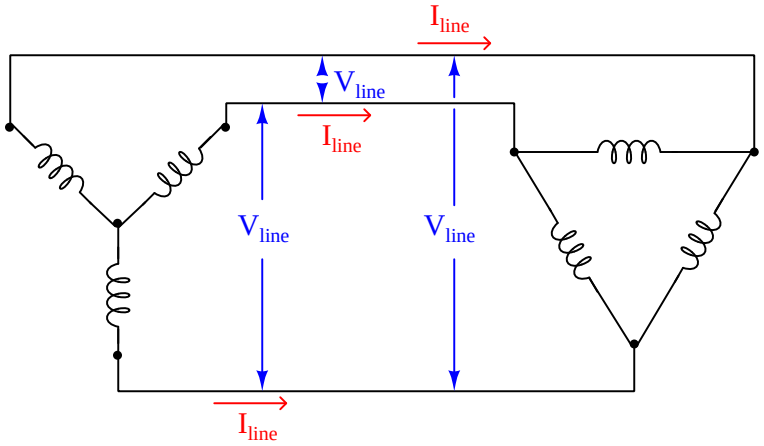
Each configuration has its own unique advantages and disadvantages in the larger context of a three-phase electrical power system. Either source type may connect to either load type (e.g. Delta to Wye, Delta to Delta, Wye to Delta, Wye to Wye) so long as the voltage and current ratings of all components are compatible.

[1] The subject of polyphase transformer wiring is treated in its own learning module.

The voltage appearing across the terminals of each element in a polyphase device is called the *phase voltage*, and the current through each element in a polyphase device is called the *phase current*:



Voltage appearing between any two of the connecting conductors (power lines) is called the *line voltage* of the polyphase system, and current through any of the connecting conductors (power lines) is called the *line current*:



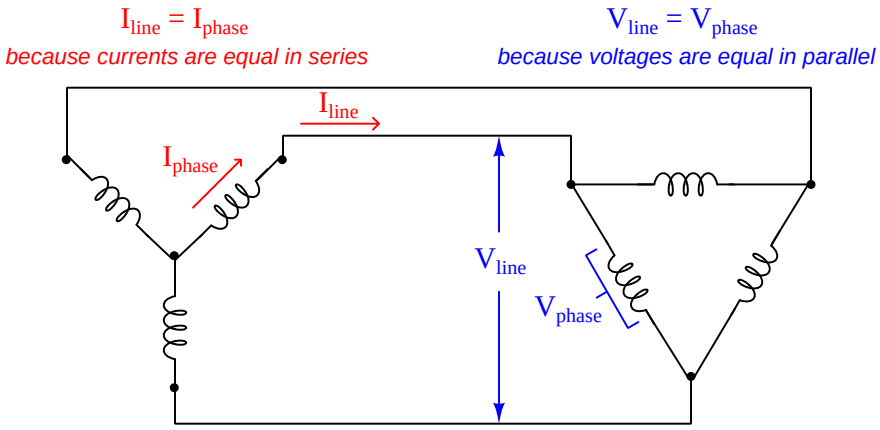
Line and phase quantities relate to each other differently between Delta versus Wye networks. Line voltage for a balanced[2] Wye device exceeds phase voltage by a factor of $\sqrt{3}$, while line current for a balanced Delta device exceeds phase current by the same factor[3]:

System type	Voltage	Current
Wye (Y)	$V_{line} = \sqrt{3} \times V_{phase}$	$I_{line} = I_{phase}$
Delta (Δ)	$V_{line} = V_{phase}$	$I_{line} = \sqrt{3} \times I_{phase}$

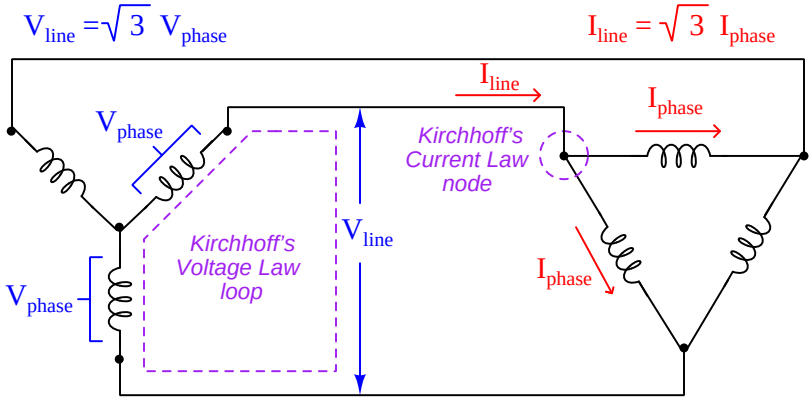
[2]Here, the term "balanced" refers to a condition where all phase voltages and currents are symmetrically equal. Unbalanced conditions can and do exist in real polyphase power systems, but the degree of imbalance is usually quite small except in cases of component faults.

[3]This factor is mathematically derived in section 5.1 beginning on page 48.

While it may be tempting to simply memorize these mathematical relationships, it is far better to *understand* why they are so[4]. In a Wye network, $I_{line} = I_{phase}$ because each line is in series with a phase element, and we know series-connected elements share the same current. Likewise, $V_{line} = V_{phase}$ in a Delta network because each line pair is in parallel with a phase element, and we know parallel-connected elements share the same voltage:



Phase and line voltages are unequal in Wye networks, as are phase and line currents in Delta networks. In each of these cases, though, we may see once again by visual inspection that these line and phase quantities cannot be equal because the line quantities are the result of *two* additive phase quantities. In a Wye network, line voltage is the (phasor) sum of two phase voltages in accordance with Kirchhoff's Voltage Law. In a Delta network, line current is the (phasor) sum of two currents in accordance with Kirchhoff's Current Law. If we know the system in question is balanced, however, we may be assured that the multiplying factor between these line and phase quantities will be the square-root of three ($\sqrt{3}$), therefore line voltage is $\sqrt{3}$ times greater than Wye phase voltage, and line current is $\sqrt{3}$ times greater than Delta phase current:



[4]This is a general principle of learning: always seek to discover the reasons why. Rote memorization is of limited value.

As an example of phase and line voltages in a Wye-connected system, we see a great many three-phase industrial power circuits in the United States of the "480/277" Volt configuration. These are Wye-connected devices exhibiting having phase voltages of 277 Volts RMS (each) and a balanced line voltage of 480 Volts RMS, because $277 \times \sqrt{3} \approx 480$.

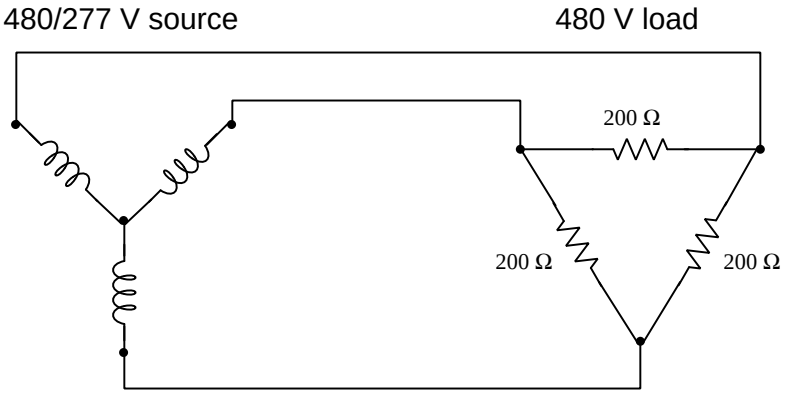
In the Wye-connected system we see how two smaller phase voltages add to form a larger line voltage. In the Delta-connected system we see how a larger line current splits to form two smaller phase currents. The key to understanding these mathematical relationships is to recognize where the properties of series and parallel networks and Kirchhoff's Laws dictate quantities be identical or different, and then all we need to remember is that if the two are different, the line quantity will be greater by a factor of $\sqrt{3}$.

One of the major advantages of polyphase electrical power systems over single-phase electrical power systems is a more continuous transfer of energy. In a three-phase circuit there is never a moment in time where all currents or all voltages are zero: at every moment in time work is being done, and energy transferred from source to load. Single-phase AC power systems, however, experience moments in time where energy transfer halts: whenever the voltage and/or current waveforms cross zero. This distinction is analogous to a multi-cylinder piston engine compared to a single-cylinder piston engine: the multi-cylinder engine's power output is more continuous over time because there is always at least one piston performing work while the others are not, while the single-cylinder engine's output "pulses" as the one piston must successively perform intake, compression, power, and exhaust strokes.

This continuity of energy transfer results in more power transmitted from source to load for a given amount of line current and line voltage than for a single-phase system. This is important in the design of high-power electrical transmission lines, where greater voltage requires larger insulators and greater current requires heavier-gauge wire. The advantage of less voltage and/or less current required to transmit the same amount of power in a three-wire three-phase system versus a two-wire single-phase is great enough that it offsets the additional cost of the third conductor.

However, polyphase AC circuits were actually invented as an alternative to single-phase AC circuits for yet another reason: to facilitate a better design of *AC electric motor*. As it so happens, AC induction-style motors are simpler in design and construction than single-phase induction-style motors, and therefore tend to be more rugged and reliable than the latter. Three-phase AC induction motors may also be easily reversed: simply swap any two lines feeding power to the motor (thereby reversing the phase rotation), and the motor spins in the opposite direction as before. This economy of motor design, more than anything else, is what led to the development and widespread adoption of polyphase AC generators and power systems.

In order to better understand power calculations in three-phase circuits, we will consider a specific example. Here we have a 480/277 Wye-connected alternator supplying electrical power to a Delta-connected load consisting of three 200 Ohm resistive heating elements:



To begin our calculation of all electrical quantities in this circuit, we will apply Ohm's Law to the calculation of phase current at the load, since we already know the load's phase voltage (480 Volts RMS) and phase resistance (200 Ohms):

$$I_{phase(load)} = \frac{480 \text{ V}}{200 \ \Omega} = 2.4 \text{ A}$$

Now that we know phase current at the Delta-connected load, we may calculate line current for the whole system by multiplying by the square-root of three:

$$I_{line} = (\sqrt{3})(2.4 \text{ A}) = 4.157 \text{ A}$$

This line current must be the same as the phase current in the Wye-connected alternator, since line and phase currents are equal in Wye-connected devices by virtue of their series connection. We already know the phase voltage of the alternator (277 Volts) because that was given to us, but we could just as well calculate it from the line voltage of 480 Volts as such:

$$V_{line} = (\sqrt{3})(V_{phase(source)})$$

$$V_{phase(source)} = \frac{V_{line}}{\sqrt{3}}$$

$$V_{phase(source)} = \frac{480 \text{ V}}{\sqrt{3}} = 277.1 \text{ V} \approx 277 \text{ V}$$

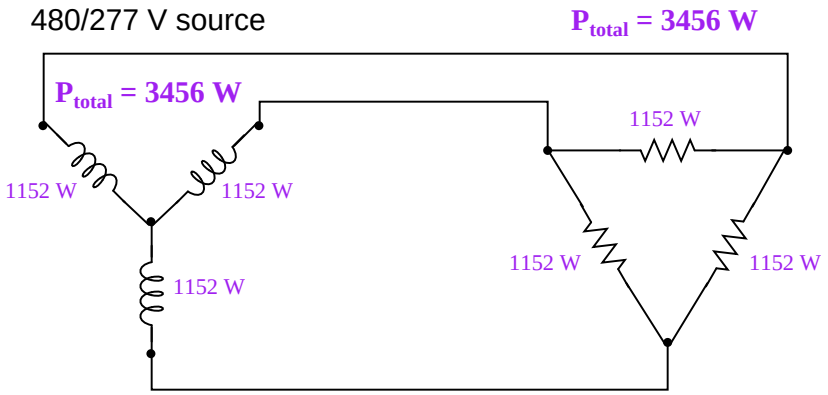
Tabulating all phase voltages and currents in our balanced system with a line voltage of 480 V and a line current of 4.157 A:

Quantity	Source	Load
V_{phase}	277 V	480 V
I_{phase}	4.157 A	2.4 A

Power for each of the three source or three load elements in this balanced system is simply the product of phase voltage and phase current ($P = IV$) as it is for any resistive load. Expanding our table to include the power for each phase element:

Quantity	Source	Load
V_{phase}	277 V	480 V
I_{phase}	4.157 A	2.4 A
P_{phase}	1152 W	1152 W

Total generated power at the alternator (as well as total dissipated power at the resistive heater) is the simple sum of all three phase elements: 3456 watts in each case. No "square-root-of-three" factor is required in this calculation, because power (work over time) is not a phasor quantity[5]. The Law of Energy Conservation demands that all power be accounted for, and thus three resistors dissipating 1152 watts each must be together dissipating 3456 watts total:



[5]You may recall from basic physics that while force and displacement are both vector quantities (having direction as well as magnitude), *work* and *energy* are not. Since power is nothing more than the rate of work over time, and neither work nor time are vector quantities, power is not a vector quantity either. This is closely analogous to voltage, current, and power in polyphase electrical networks, where both voltage and current are phasor quantities (having phase angle "direction" as well as magnitude) but where power merely has magnitude. We call such "directionless" quantities *scalar*. Scalar arithmetic is simple, with quantities adding and subtracting directly rather than trigonometrically.

In the interest of convenience, though, it is helpful to have a formula to calculate power in a balanced three-phase system knowing just the line voltage and current rather than phase voltages and currents. You will note how the simple product of line voltage (480 V) and line current (4.157 A) does *not* yield total power (3456 W) in this system. We may develop a proper formula for calculating total power from line voltage and current by beginning with the formula described in calculating total power from the power of each resistor at the Delta-connected load:

$$P_{total} = (3)(P_{phase})$$

$$P_{total} = (3)(I_{phase})(V_{phase})$$

We may substitute I_{line} for I_{phase} and V_{line} for V_{phase} in this equation if we properly relate them for the Delta[6] connection. While $V_{line} = V_{phase}$ in a Delta configuration, $I_{phase} = \frac{I_{line}}{\sqrt{3}}$:

$$P_{total} = (3)\left(\frac{I_{line}}{\sqrt{3}}\right)(V_{line})$$

We may consolidate the two constants in this formula (3 and $\sqrt{3}$) by re-writing the number 3 as the product $\sqrt{3}\sqrt{3}$, then canceling one of these with the $\sqrt{3}$ in the denominator:

$$P_{total} = (\sqrt{3}\sqrt{3})\left(\frac{I_{line}}{\sqrt{3}}\right)(V_{line})$$

$$P_{total} = (\sqrt{3})(I_{line})(V_{line})$$

As a test of our conclusion, we may check to see that this new formula accurately calculates the total power[7] of our balanced three-phase system:

$$P_{total} = (\sqrt{3})(4.157 \text{ A})(480 \text{ V})$$

$$P_{total} = 3456 \text{ W}$$

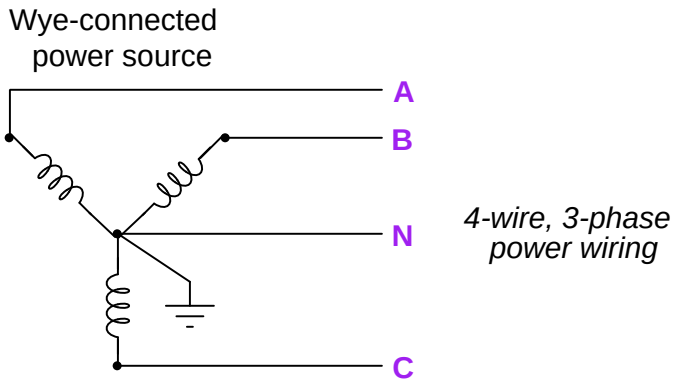
[6] We end up with the same final result if we substitute line quantities in a Wye-connected system, too. Instead of $V_{line} = V_{phase}$ and $I_{phase} = \frac{I_{line}}{\sqrt{3}}$ in the Delta connection we have $I_{line} = I_{phase}$ and $V_{phase} = \frac{V_{line}}{\sqrt{3}}$ in the Wye connection. The end-result is still $P_{total} = (\sqrt{3})(I_{line})(V_{line})$ based on line quantities.

[7] In the United States, total system power is often expressed in units of *horsepower*, with 745.7 Watts equal to 1 horsepower. Therefore, 3456 W could be expressed as 4.635 HP. The equivalence of 745.7 Watts to 1 horsepower is important to know when working with electric motors which (at least in the United States) are usually rated in horsepower.

3.3 Wye and Delta grounding

So far all the three-phase configurations shown have been *ungrounded*: that is, none of the terminals or conductors have any direct connection to Earth ground. While it is possible (and practical in many cases) to use polyphase power without an explicit Earth ground connection, it is not always the safest. Firmly connecting the source to Earth limits the voltage which may develop between any non-grounded ("hot") conductor and Earth ground for all loads powered by that source. This is especially important in power systems with overhead lines, where lightning strikes may dramatically elevate common-mode[8] voltages in the system.

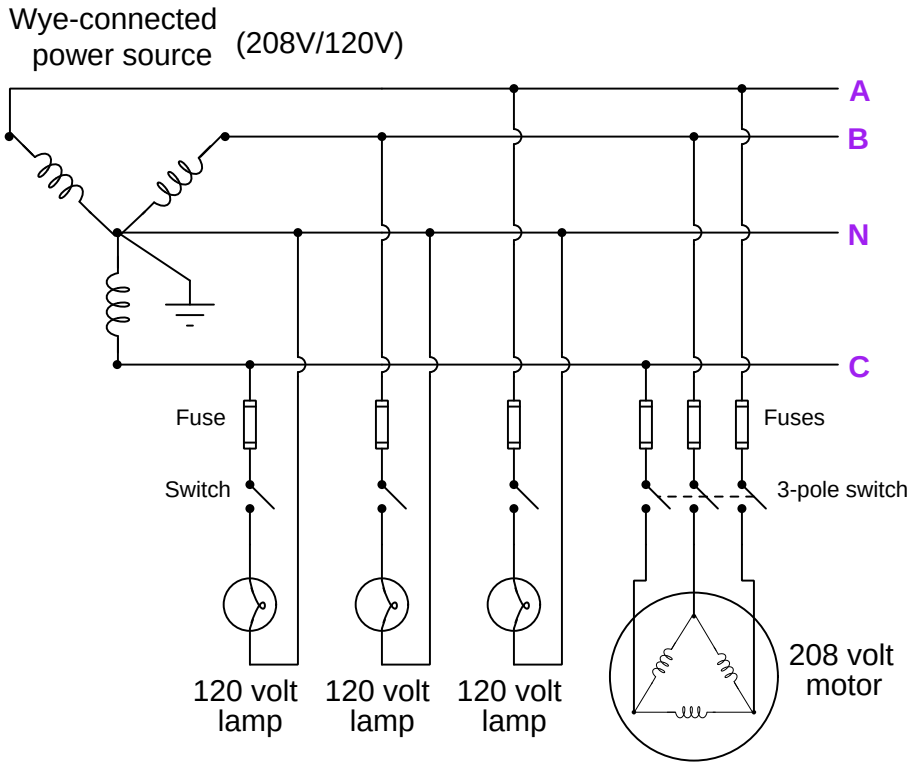
In Wye-connected systems, the natural point to ground is the center of the "Y", like this:



The three "hot" terminals of the source are typically labeled "A", "B", and "C", while the grounded point is referred to as the "Neutral" (N). The voltage measured between any two of the "hot" terminals (A to B, B to C, or A to C) will be $\sqrt{3}$ times more than the voltage measured between any "hot" terminal and the neutral (A to N, B to N, or C to N). Common voltages for 4-wire Wye-connected systems include 208/120 and 480/277.

[8] *Common-mode voltage* is the algebraic average of voltage with respect to ground shared among multiple points.

The existence of dual voltage levels in a center-grounded Wye system enables the use of loads with different voltage ratings. For example, in a 208/120 Wye system, a three-phase motor with windings rated for 208 Volts would draw power from the three "hot" conductors directly, while lamps rated for 120 Volts would connect between any "hot" conductor and the neutral:



A good practice in such systems is to equally spread the 120 Volt loads among the three phases, so that (ideally) the phase loading on the source will be nicely balanced when all loads are operating. If the loading is perfectly balanced, in fact, the neutral conductor will carry no current at the point where it connects to the center of the Wye.

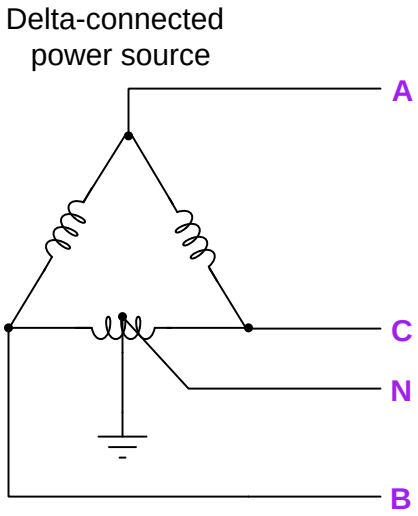
Note the use of fuses on all the "hot" load conductors, but no fuses on any "neutral" (grounded) conductor. The reason for this is so that in the event of a fuse blowing, only the hot conductor(s) will be disconnected from the load, leaving the neutral conductor connected and thereby maintaining the lowest possible potential at the load relative to Earth ground for safety.

Note also how only the source has an Earth ground connection, and none of the loads. This is important, as multiple connections to Earth ground between source and load invites *ground loop* currents which are problematic for many reasons[9]. Also, grounding at the source ensures all loads

[9]The Earth is a relatively poor conductor of electricity, and so any current passing from one grounding point to another will develop a voltage drop between those points, just like current passing through a resistor, which would make those points no longer equipotential with each other. Also, ground currents can, over time, cause corrosion of the metal grounding rod as it chemically interacts with ions in the soil, and this will increase its contact resistance to make it a poorer ground connection. Finally, ground loops invite electrical noise, as they constitute a physically

powered by that source have a reliable ground reference; if the entire system relied on one load for its grounding and that load ever disconnected from the system for any reason, the grounding would be lost.

In Delta-connected systems, there is no "natural" point to connect to Earth ground as there is in Wye-connected systems. The most common grounding configuration for a Delta-connected source is shown here, sometimes called the "high-leg" connection. Here, one of the source's phase coils is center-tapped to provide a connection point for Earth ground:



This configuration yields *three different* voltages available to power loads. If the phase voltage of the Delta-connected source is 240 Volts, the three available voltages are:

- 240 Volts (between A-B, B-C, and A-C)
- 120 Volts (between B-N and C-N)
- 208 Volts (between A-N)[10]

A disadvantage of this configuration is that the lower-voltage loads cannot be balanced among all three phase coils of the source as they can in Wye-connected systems. Any single-phase loads (those connected between any one "hot" conductor and the neutral conductor) inevitably place more burden on the B-C phase coil than the other two phase coils. However, this imbalance is often negligible in commercial settings where three-phase loads (e.g. motors) are larger than single-phase loads (e.g. receptacles and lighting).

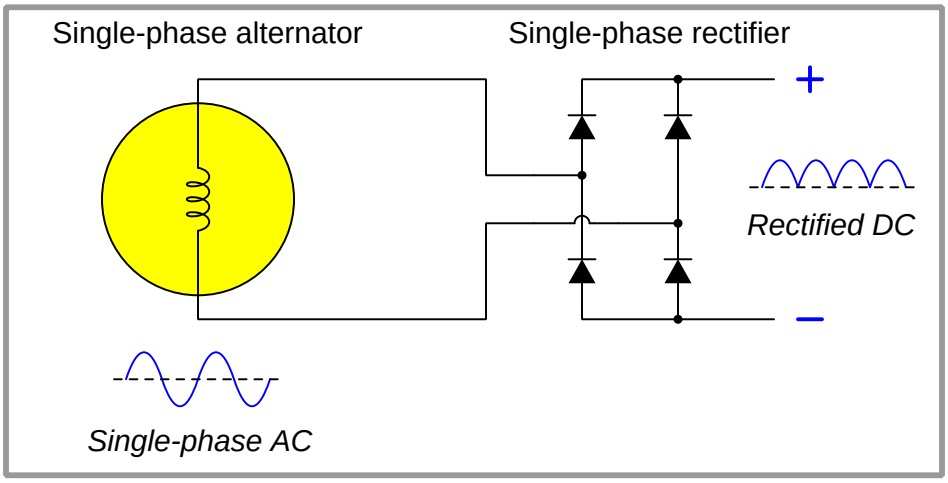
Another interesting characteristic of this grounding scheme is that any Wye-connected load powered by such a "high-leg" Delta source is guaranteed to have a center-node potential that is *not* the same as Earth's. This may be a source of confusion for anyone accustomed to measuring phase-to-ground voltages on a Wye-connected network.

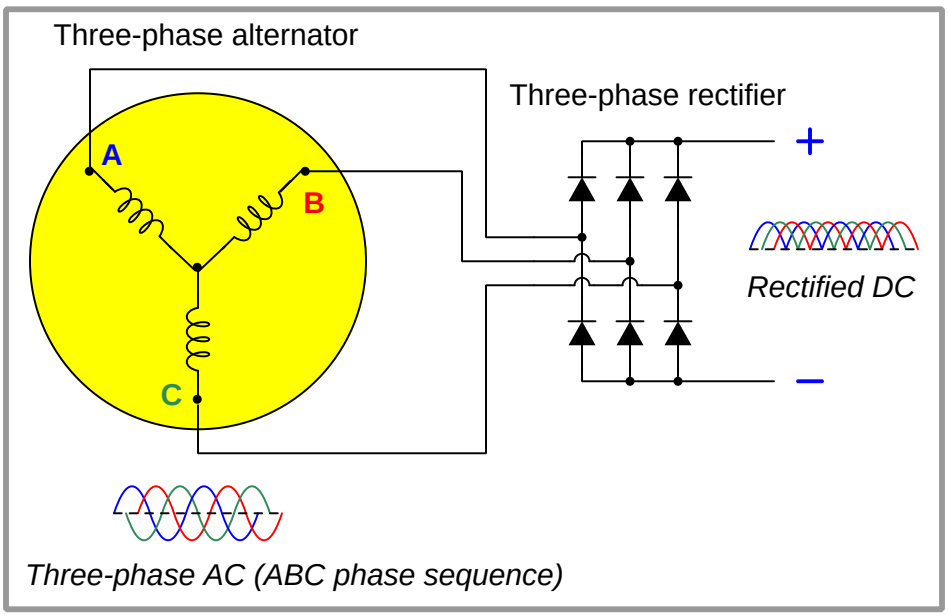
large circuit which may experience induced currents from magnetic fields perpendicular to that loop. This noise may compromise the integrity of electronic signals with connections to the power circuit.

[10] A colorful term for this odd voltage is *bastard voltage*.

3.4 Three-phase rectification

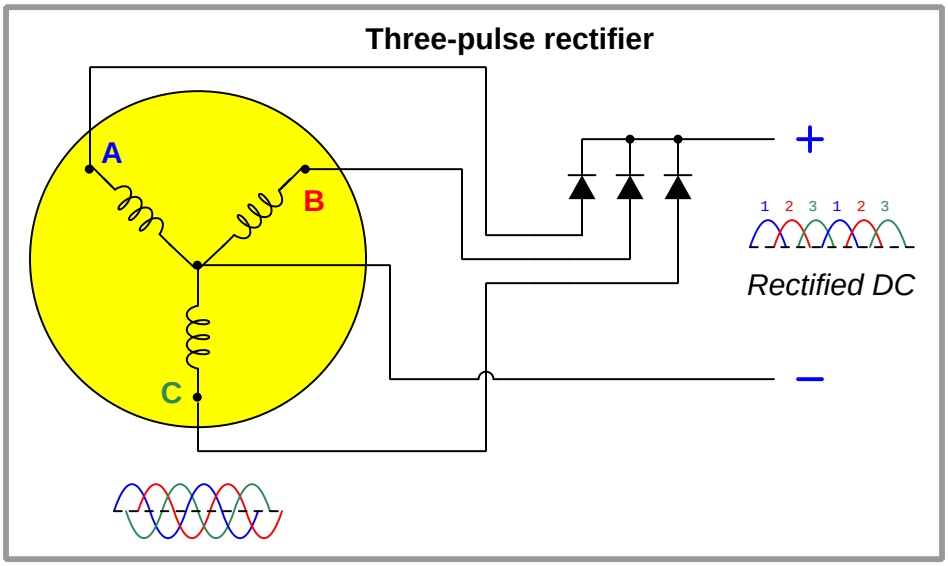
Another advantage of three-phase AC power is in applications where the AC must be rectified into DC using semiconductor *diodes*, which may be thought of as *one-way valves* for electricity: allowing current to flow in one direction but blocking current from flowing in the other:

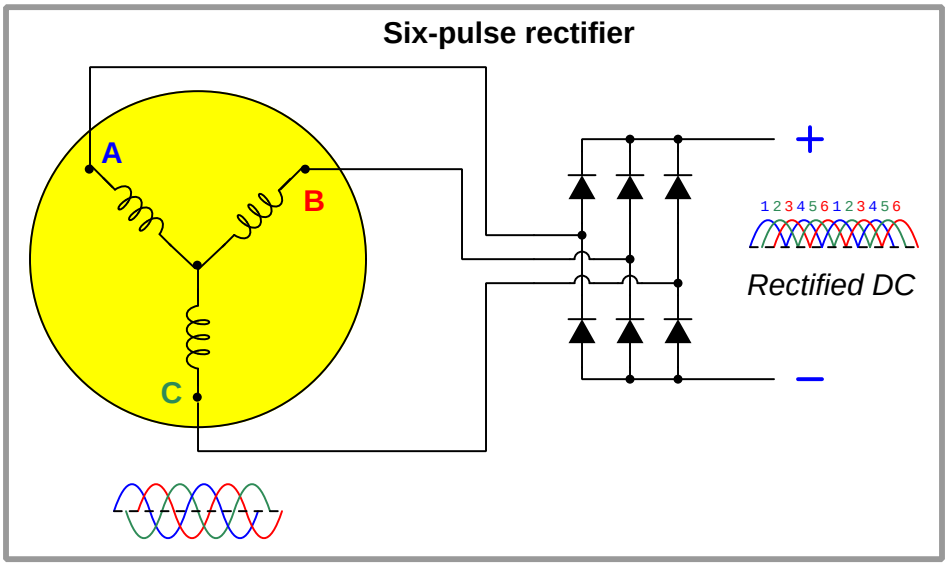




Rectified three-phase AC is "smoother" rectified single-phase AC, with less ripple voltage to create noise that may couple to nearby electronic circuits, because the phase-shifted currents overlap each other. This is why all automotive alternators are three-phase rather than single-phase machines.

Both of the rectifier circuits previously shown are *full-wave* in nature, making full use of each half-cycle of the AC. A three-phase, full-wave rectifier is also known as a *six-pulse rectifier* because for each full cycle of the AC waveform there are six distinct pulses output by the rectifier network: two pulses per phase multiplied by three phases. However, half-wave rectification of three-phase AC is also possible, and such a rectifier is known as a *three-pulse rectifier*:





3.5 Harmonics in three-phase networks

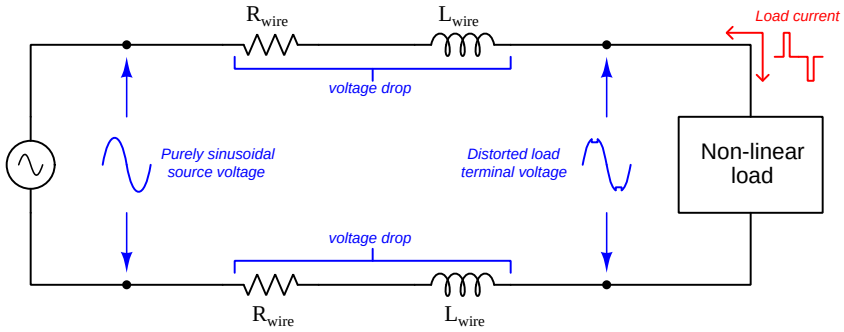
Any distortion of an otherwise ideal sinusoidal oscillation is mathematically equivalent to a series of perfectly sinusoidal oscillations of differing frequency summed together. This is broadly referred to as *Fourier's Theorem*:

Fourier's Theorem

> **Any wave-shape, no matter how complicated, is equivalent to a sum of purely sinusoidal waveforms.**

If the wave-shape in question is *periodic* (i.e. repeating the exact same pattern over and over again) then these equivalent sinusoids will have frequencies limited to integer-multiples of the base waveform's frequency, each of these integer-multiple values being known as a *harmonic frequency*. For example, if a non-sinusoidal waveform repeats its shape every 20 milliseconds, its fundamental frequency will be 50 Hertz and any harmonic frequencies contained therein are limited to 100 Hz (second harmonic), 150 Hz (third harmonic), and/or 200 Hz (fourth harmonic), etc.

AC generators naturally produce sinusoidal voltages, and so if an AC generator powers one or more *linear*[11] loads containing resistance, capacitance, and/or inductance we would expect the current(s) in that circuit to be equally sinusoidal in wave-shape. However, some components such as semiconductors (e.g. diodes and transistors) are non-linear which means current will not flow in direct and constant proportion to applied voltage. This means non-linear loads energized by even the most perfectly sinusoidal voltage will draw non-sinusoidal currents (i.e. the current will consist of multiple harmonics). As these harmonic currents pass through linear circuit elements such as resistance and reactance[12], they produce voltage drops with the same frequencies, and this in turn causes measured voltages in most AC circuits to inevitably contain harmonics as well:

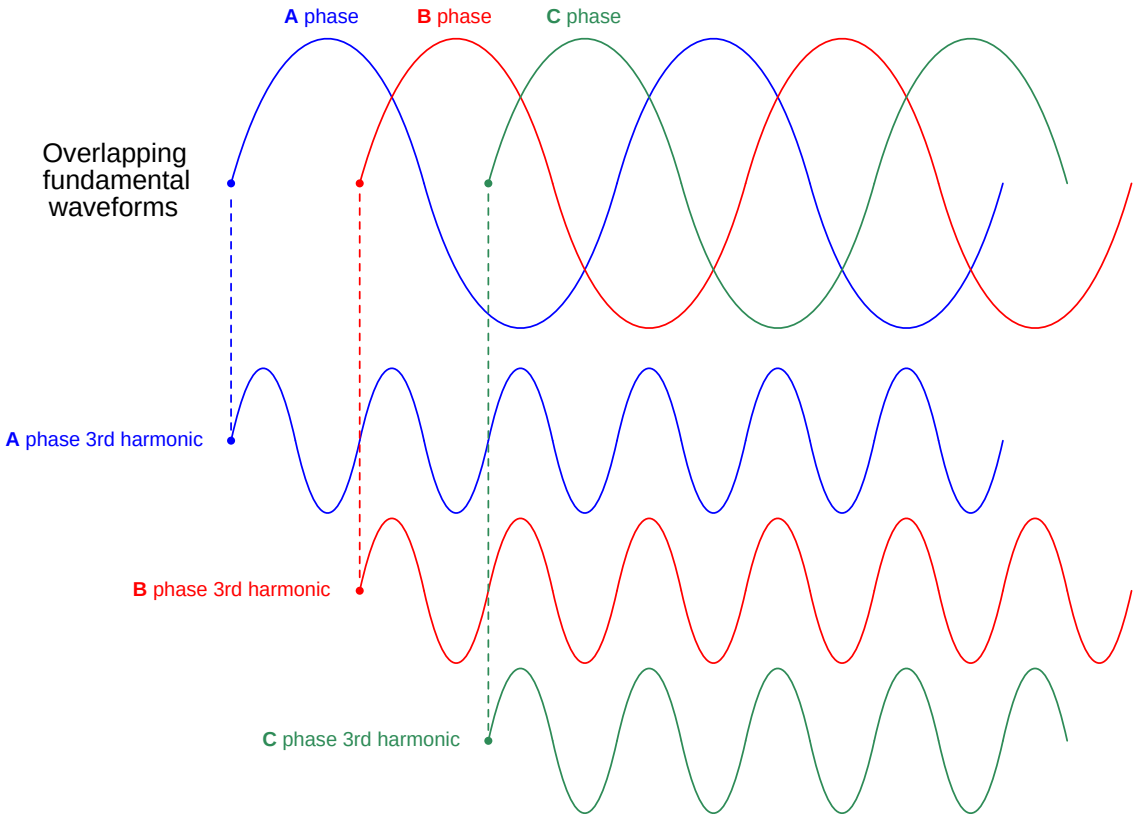


Given the popularity of semiconductor-based switching loads such as switch-mode power supply circuits and electronic motor speed controls, harmonic currents and voltages are nearly impossible to avoid in modern AC power circuits.

[11] A "linear" electrical component is one with a constant proportion of voltage to current. Both resistance (R) and reactance (X_C and X_L) meet this criterion, making resistance, capacitance, and inductance all linear in nature.

[12] Examples include ohmic resistance of metallic conductors and closed switch contacts, as well as reactive parasitics such as conductor inductance and leakage inductance within transformers.

In a three-phase AC power circuit, the three phases in question are shifted 120 degrees from each other. However, the *harmonics* of those three phases may not exhibit the same amount of relative phase shift because their periods differ from that of the fundamental. This is clearest to see in the case of *third* harmonics, where the harmonic frequency is precisely three times higher than the fundamental. If we examine the following illustration showing the three phase-shifted fundamental waveforms of a three-phase AC circuit along with the third-harmonic waveforms for each of the fundamentals, we see that the third-harmonic waveforms are actually in-phase with each other:



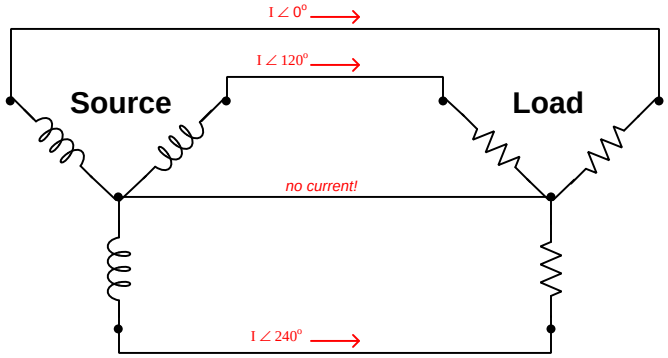
Mathematically, this is because $3 \times 120^o = 360^o$, a phase-shift of 360 degrees being equivalent to no phase shift at all. Note how the B-phase third harmonic starts at exactly the same moment in time when the A-phase third harmonic completes its first whole cycle, because the time interval represented by 120^o of phase shift separating A-phase from B-phase is the same time interval as one complete cycle (360^o) of separation for waves oscillating three times as often.

This zero-phase-shift phenomenon holds true not just for the third harmonic, but also for any harmonic with a number that is *any integer multiple of three*. Therefore we also see 6th harmonic waveforms being exactly in-phase with each other in a three-phase circuit, as well as 9th harmonics, 12th harmonics, 15th harmonics, 18th harmonics, etc. The general term for these specific harmonics is *triplen*. However, in practice the third harmonic is usually the only significant triplen harmonic

frequency we need be concerned about, since even-numbered harmonics are almost non-existent[13] in most power circuits, and the remaining odd harmonics (9th, 15th, etc.) are generally much weaker in magnitude than the third.

Triplen harmonics result in strange effects within three-phase networks because they add together in places where the fundamentals cancel. Consider a perfectly balanced four-wire Wye source/load network that is also perfectly free of harmonics:

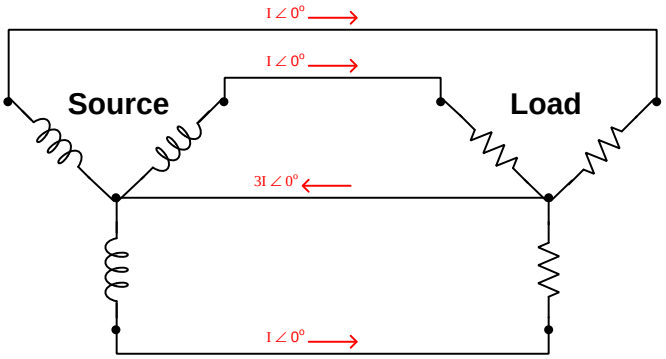
Balanced three-phase network with no harmonics



In this circuit, the center ("neutral") conductor carries no current whatsoever, because the three phase currents with their 120^o mutual phase-shifts perfectly cancel at the center nodes of the Wye source and the Wye load. In fact, the neutral conductor is unnecessary and could even be eliminated!

Compare this happy state of affairs with the next network harboring triplen harmonic currents:

Balanced triplen harmonics in three-phase network

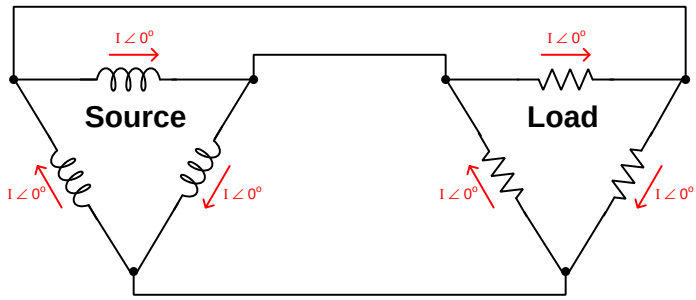


Here, the "neutral" conductor carries the sum total of all triplen harmonic currents in the three lines because those triplen harmonic currents are all in-phase with each other.

[13] If the waveform distortion created by non-linear loads is symmetrical above and below the zero line, only odd harmonics result. Most non-linear loads produce current waves having this symmetry, which means even-numbered harmonics are usually weak enough to ignore.

Triplen currents in Delta networks have no neutral conductor to sum at, and so they *circulate* through the phase elements of each Delta source and load:

Balanced triplen harmonics in three-phase network



Such currents, when circulating through the Delta source, perform no useful work and indeed contribute to heating of the source by uselessly dissipating power in the source's internal resistance.

Non-triplen harmonics also result in strange effects when present in three-phase AC power networks. We will consider only odd harmonics here because even-numbered harmonics are usually of insignificant magnitude in AC power systems[14].

Fifth harmonics, by definition, have frequency values five times greater than the fundamental. In a three-phase power system where the fundamental waveforms are phase-shifted by 120^o this means the relative phase-shifts of the fifth harmonics will be five times larger: $5 \times 120^o = 600^o$ and $5 \times 240^o = 1200^o$. As we saw with triplen harmonics, we need to understand what these harmonic phase-shifts mean in relation to whole cycles (i.e. multiples of 360^o)[15]. In the case of fifth harmonics, $5 \times 120^o = (1 \times 360^o) + 240^o$ and $5 \times 240^o = (3 \times 360^o) + 120^o$. This means for a three-phase power network with an A-B-C phase sequence (i.e. A $\angle$ 240^o, B $\angle$ 120^o, and C $\angle$ 0^o) the effective phase rotation of the fifth harmonics will be *backwards* (i.e. A_5 $\angle$ 120^o, B_5 $\angle$ 240^o, and C_5 $\angle$ 0^o, which creates a B-A-C or C-B-A phase sequence).

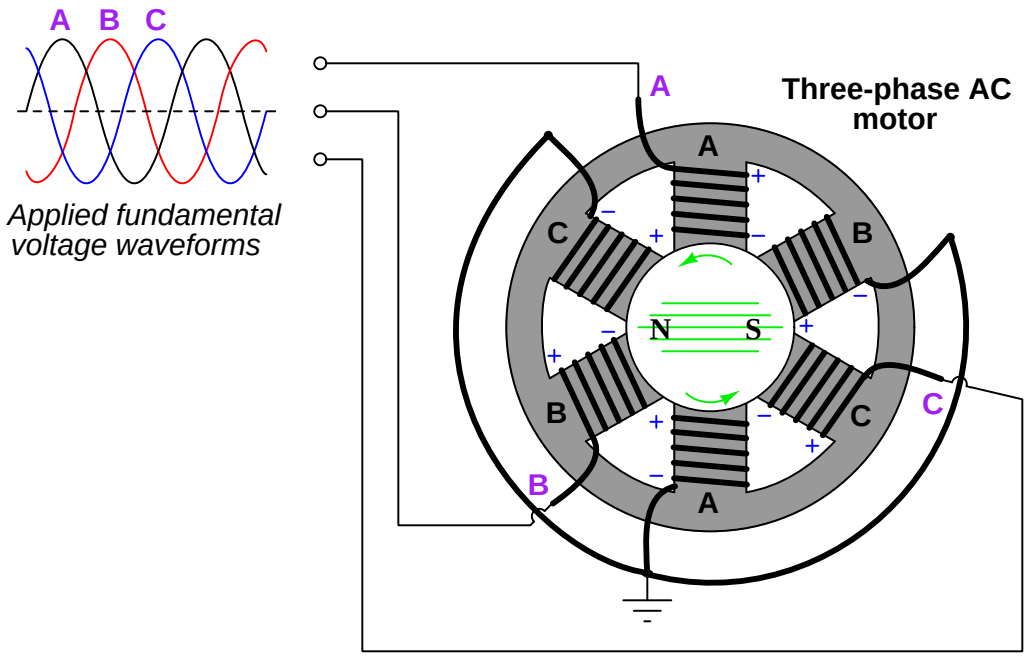
Seventh harmonics, by definition, have frequency values seven times greater than the fundamental. In a three-phase power system where the fundamental waveforms are phase-shifted by 120^o this means the relative phase-shifts of the seventh harmonics will be seven times larger: $7 \times 120^o = 840^o = (2 \times 360^o) + 120^o$ and $7 \times 240^o = 1680^o = (4 \times 360^o) + 240^o$. This means for a three-phase power network with an A-B-C phase sequence (i.e. A $\angle$ 240^o, B $\angle$ 120^o, and C $\angle$ 0^o) the effective phase rotation of the seventh harmonics will also be A-B-C (i.e. A_7 $\angle$ 240^o, B_7 $\angle$ 120^o, and C_7 $\angle$ 0^o) and therefore rotate in the same direction.

To summarize, third harmonics (and, by extension, all triplen harmonics) are all in phase and exhibit no "rotation" at all; fifth harmonics rotate against the direction of the fundamental; seventh harmonics rotate with the direction of the fundamental.

[14] Even-numbered harmonics are only present when the wave-shape is distorted *asymmetrically* about the horizontal axis, and non-linear loads will cause this only if the "fire" (i.e. draw current) more in one-half cycle than in the other. This is usually not the case, where loads typically draw current equally during both half-cycles.

[15] An arithmetic operation useful for this purpose is *modulo* which is usually symbolized by the "percent" symbol (%) in computer programming. For example, to compute the practical phase shift of the fifth harmonic of the 120^o-shift fundamental using the Python programming language, the calculation would be entered as `(5 * 120) % 360`.

Phase rotation profoundly affects polyphase AC motors, whose very operating principle is that of a rotating magnetic field created by energizing sets of stator windings with out-of-phase currents. In the illustration below we see a 6-pole, 3-phase AC induction motor energized by a 3-phase AC source having a phase rotation of A-B-C, creating a rotating magnetic field spinning counter-clockwise:



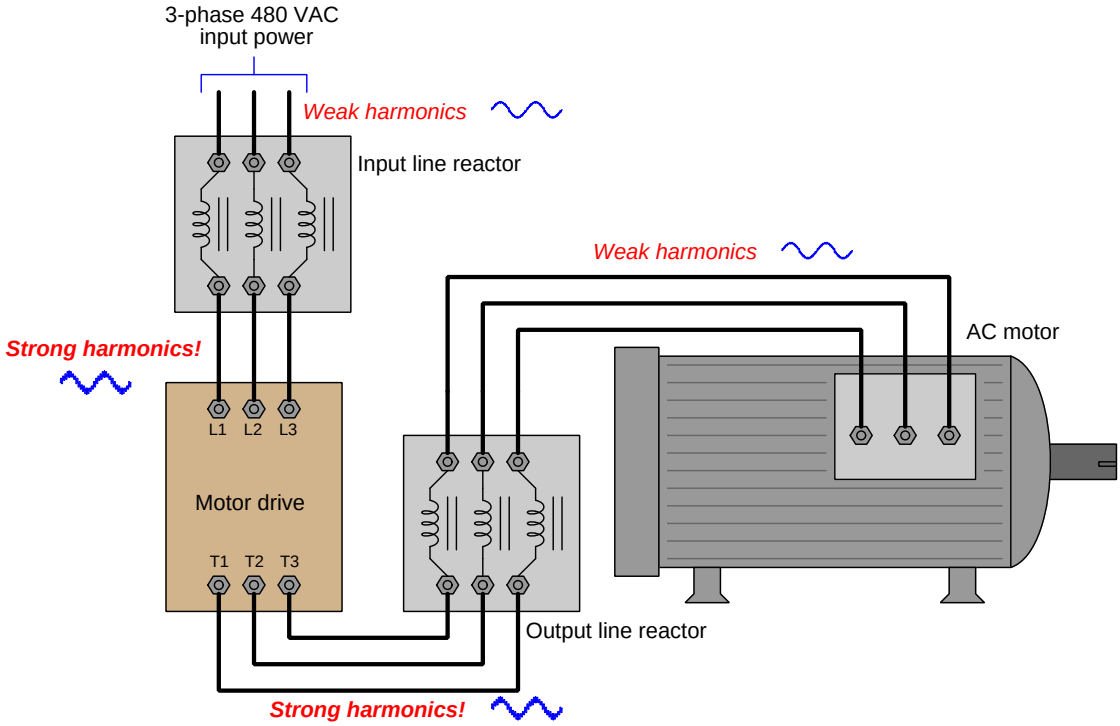
If the energizing voltage is totally free of all harmonics (i.e. perfectly sinusoidal), then this counter-clockwise rotating magnetic field will be the *only* magnetic field acting upon the rotor to make it spin. However, the presence of fifth harmonics in the energizing current will create a superimposed rotating magnetic field spinning *five times faster in the wrong direction!* Any seventh harmonics in the motor's energizing current create yet another superimposed magnetic field rotating in the correct direction (same as the fundamental) but seven times faster than the fundamental.

Clearly, fifth harmonics are counter-productive in the powering of AC motors because they create a magnetic field rotating in the wrong direction which tends to rob kinetic energy from the rotor, but even third and seventh harmonics are troublesome in that their higher frequencies result in greater effective conductor resistance due to the *skin effect* as well as higher dissipative losses resulting from more eddy currents forming within the iron of the motor's stator and rotor. Simply put, *all harmonics rob AC electric motors of efficiency.*

Direction of phase rotation for harmonics in a polyphase AC network may be categorized as either *positive-sequence* (i.e. rotating *with* the fundamental), *negative-sequence* (i.e. rotating *against* the fundamental), or *zero-sequence* (i.e. not rotating at all), and these different sequences follow a definite pattern as we ascend in harmonic number. Consider the following table showing the phase sequences of each harmonic (where n is the harmonic number, 1 being the fundamental) within a three-phase system having a fundamental rotation of A-B-C, using the *modulo* (%) operation to compute the remaining phase angle for each harmonic after whole divisions of 360 degrees:

n	**A phase angle**	**B phase angle**	**C phase angle**	**Sequence**
1	240	120	0	A-B-C (Positive)
3	$(3 \times 240)\%360 = 0$	$(3 \times 120)\%360 = 0$	$(3 \times 0)\%360 = 0$	C-C-C (Zero)
5	$(5 \times 240)\%360 = 120$	$(5 \times 120)\%360 = 240$	$(5 \times 0)\%360 = 0$	B-A-C (Negative)
7	$(7 \times 240)\%360 = 240$	$(7 \times 120)\%360 = 120$	$(7 \times 0)\%360 = 0$	A-B-C (Positive)
9	$(9 \times 240)\%360 = 0$	$(9 \times 120)\%360 = 0$	$(9 \times 0)\%360 = 0$	C-C-C (Zero)
11	$(11 \times 240)\%360 = 120$	$(11 \times 120)\%360 = 240$	$(11 \times 0)\%360 = 0$	B-A-C (Negative)
13	$(13 \times 240)\%360 = 240$	$(13 \times 120)\%360 = 120$	$(13 \times 0)\%360 = 0$	A-B-C (Positive)
15	$(15 \times 240)\%360 = 0$	$(15 \times 120)\%360 = 0$	$(15 \times 0)\%360 = 0$	C-C-C (Zero)
17	$(17 \times 240)\%360 = 120$	$(17 \times 120)\%360 = 240$	$(17 \times 0)\%360 = 0$	B-A-C (Negative)

If harmonics cannot be mitigated at their source (e.g. by replacing non-linear circuit components with linear components), then they may be minimized by adding *filter networks* to the circuit. A good example of this is strategy is seen in the following illustration where an electronic power-control device called a *variable frequency drive* or *VFD* is used to control the speed of a three-phase AC motor. VFDs work on the principle of transistors rapidly turning on and off to synthesize polyphase AC waveforms at the necessary frequency to make the motor spin at the desired speed, but that on/off switching necessarily creates strong harmonics on both the source side and the load (motor) side of the VFD circuit, both of which may be mostly confined to the VFD itself by means of inductive line filters:



These line filters are particularly effective at impeding triplen harmonic currents because the mutual inductance of their windings is far greater for three in-phase currents than their self-inductance is for line currents that are out-of-phase with each other. Triplen currents reaching a three-phase load, being in-phase with each other, must find some return path back to the source other than the three line conductors they arrived through, and if no fourth (neutral) conductor exists they will find a path to Earth ground through parasitic capacitance(s) within the load. Without line filtering, triplen ground currents may become severe enough to be detected by circuit-protection instruments as an actual *ground fault*[16], causing power to be automatically shut off to the equipment!

[16] I first experienced this as a teacher, having my students learn VFD technology by using small VFDs powered by conventional 120 VAC line power. Our lab room had ground fault current interrupting power receptacles, and we discovered that these would often "trip" when we attempted to control the motor's speed using the VFD! The only

practical solutions we found to this problem was to either (1) plug the VFD into a non-GFCI receptacle or (2) install a line filter blocking these triplen harmonics.

Chapter 4

Historical References

This chapter is where you will find references to historical texts and technologies related to the module's topic.

Readers may wonder why historical references might be included in any modern lesson on a subject. Why dwell on old ideas and obsolete technologies? One answer to this question is that the initial discoveries and early applications of scientific principles typically present those principles in forms that are unusually easy to grasp. Anyone who first discovers a new principle must necessarily do so from a perspective of ignorance (i.e. if you truly *discover* something yourself, it means you must have come to that discovery with no prior knowledge of it and no hints from others knowledgeable in it), and in so doing the discoverer lacks any hindsight or advantage that might have otherwise come from a more advanced perspective. Thus, discoverers are forced to think and express themselves in less-advanced terms, and this often makes their explanations more readily accessible to others who, like the discoverer, comes to this idea with no prior knowledge. Furthermore, early discoverers often faced the daunting challenge of explaining their new and complex ideas to a naturally skeptical scientific community, and this pressure incentivized clear and compelling communication. As James Clerk Maxwell eloquently stated in the Preface to his book *A Treatise on Electricity and Magnetism* written in 1873,

> It is of great advantage to the student of any subject to read the original memoirs on that subject, for science is always most completely assimilated when it is in its nascent state . . . `[page xi]`

Furthermore, grasping the historical context of technological discoveries is important for understanding how science intersects with culture and civilization, which is ever important because new discoveries and new applications of existing discoveries will always continue to impact our lives. One will often find themselves impressed by the ingenuity of previous generations, and by the high degree of refinement to which now-obsolete technologies were once raised. There is much to learn and much inspiration to be drawn from the technological past, and to the inquisitive mind these historical references are treasures waiting to be (re)-discovered.

4.1 Prototype electrical power transmission system

An interesting description of a prototype three-phase electric power system installed in Frankfort, Germany is found on page 454 of Volume 1 of *Cassier's Magazine*, which was an engineering periodical published from 1891 to the early 1900's out of London, England. The article in which this quote is found is called "The Transmission of Power" written by Carl Hering, describing the innovative features of a three-phase electric power system, particularly with reference to electric motors:

> Time will not permit a detailed discussion of this three-phase system here; it will suffice to say that it consists of three simple alternating currents, – one in each wire, – the waves or impulses in the three wires following each other in succession similarly to the relative motions of the three cranks of a three-cylinder steam engine, or the pistons of a three-plunger pump. Such a set of currents produces a *rotating* magnetic field, while the simple alternating current produces merely an *oscillating* field. The rotating field furthermore has a definite direction of rotation. It is this feature of a rotating magnetic field as distinguished from an oscillating one which renders it particularly well adapted to run motors. The ordinary simple alternating current motor may be said to be analogous to a steam-engine with one crank, having its slide-valve operated by an independent rapidly running engine ; it has a dead point from which it will not start, but after it is up to speed it will run very steadily. The rotating field motor, on the other hand, is analogous to a three-crank engine operating its own valves. There is no dead point, and it can start itself, even with full load. [page 454]

On this same page Hering begins to describe the lack of a neutral, or "return", wire in the three-phase power system:

> An interesting feature about this system is that although the three currents have different phases, they need no separate return leads, because the nature of the currents is such that each acts as a return current for the other two in multiple arc. The three circuits are therefore merely connected together to [page 454]
>
> a common terminal, after they have passed through the dynamos, transformers, and lamps. This terminal we will call the neutral point. This neutral point was grounded throughout the system, in the low-tension as well as in the high-tension circuits, and at both ends of the line. No current, however, will flow through the ground when the plant is working normally. [page 455]

Next, the author describes the transmission line connecting the power plant to the load:

> The line consists of three bare, hard-drawn, copper wires, 4 millimeters in diameter (slightly less than a No. 6 B. & S. wire), having a resistance of about 2 Ohms per mile. The distance being 108.6 miles, the resistance of each wire is about 217 Ohms, which, with 5 amperes, makes a loss of about 1000 volts on each of the three lines. The lines were run on poles with one cross-arm 16 feet above the ground, the wires being one meter from another. There were 3227 poles, at 200 feet spans : each pole had a skull and crossbones painted on it to indicate the sure fate of anyone who could not keep his hands off. There were 3000 large porcelain insulators, 8 inches in diameter, made in two

parts and with three oil grooves, and 6500 small ones, made in one piece and with one oil groove. The only apparatus connected with the high-tension circuit was the fuse wire and a short-circuiting device to replace the switches for opening the circuit. The fuses for each of the three lines consisted of two copper wires in parallel, each .15 millimeters in diameter (between Nos. 34 and 35 B. & S. gage) and 2 meters in length. They were stretched between two poles outside of the building. They were exploded (one can hardly apply the term "blown") several times to test their effectiveness. Even to an electrician it seems strange to think of several hundred horse-power being transmitted through a few wires about the size of a hair! The short-circuiting devices consisted of pieces of bare wire shaped like a V inverted, suspended directly over the line wires. These were placed at very frequent intervals along the line. In case of any danger or interruption, or in case the line must be handled, this wire is lowered by a string, and drops on the line wires, thereby short-cutting the line, which blows the fuse and opens the circuit. This is a very cheap and effective substitute for a switch, which would be dangerous to handle in a 30,000 volt circuit ; furthermore there is absolutely no possibility of a current being on the line when this device is down. [page 455]

Chapter 5

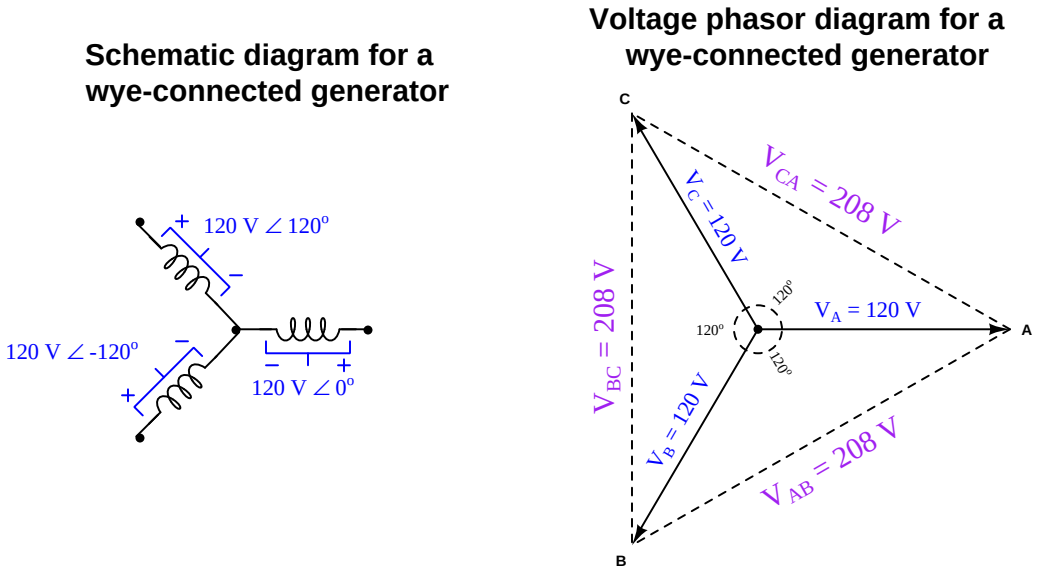
Derivations and Technical References

This chapter is where you will find mathematical derivations too detailed to include in the tutorial, and/or tables and other technical reference material.

5.1 Derivation of $\sqrt{3}$ factor

The mathematical ratio of unequal line to phase quantities in a balanced three-phase network is always $\sqrt{3}$, and this will be demonstrated for a Wye-connected network:

The following *phasor diagram* shows line and phase voltages in a Wye-connected three-phase source. Solid phasors express the phase voltage for each of the three stator winding pairs in a three-phase generator, while dashed lines express the line voltage between any two of the three output terminals on the generator. The direction of each vector (each line segment tipped by an arrowhead) expresses its phase shift in time, as the sine wave voltages produced by each of the three phase windings will be shifted apart from each other by 120 degrees. Here, each phase winding in the alternator happens to produce 120 VAC, resulting in a line voltage equal to the trigonometric sum of the 120 VAC phasors, 208 VAC:



A simple way to calculate the side lengths of these non-right triangles is to use the *Law of Sines*, which simply states the ratio between the length of a triangle's side and the sine of its opposite angle is constant, for *any* triangle:

$$\frac{\sin a}{A} = \frac{\sin b}{B} = \frac{\sin c}{C}$$

Applying this to the solution of the voltage between phases A and B (120 Volts each, individually):

$$\frac{\sin 120^o}{V_{AB}} = \frac{\sin 30^o}{V_A} = \frac{\sin 30^o}{V_B}$$

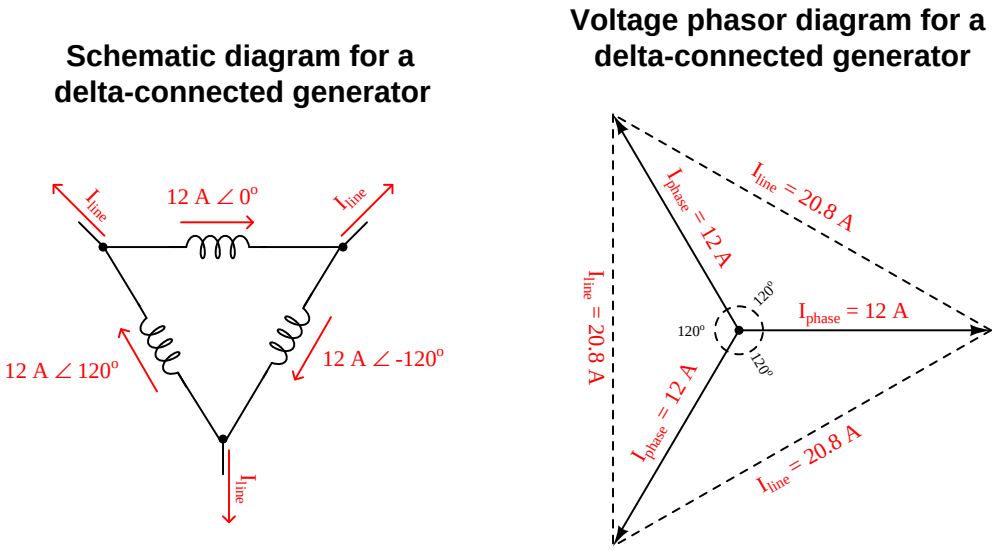
$$\frac{\sin 120^o}{V_{AB}} = \frac{\sin 30^o}{120 \text{ Volts}}$$

$$V_{AB} = (120 \text{ Volts}) \left(\frac{\sin 120^o}{\sin 30^o} \right)$$

$$V_{AB} = 207.84 \text{ Volts} \approx 208 \text{ Volts}$$

The ratio $\frac{\sin 120^o}{\sin 30^o}$ is equal to the square-root of three ($\sqrt{3}$), and with a symmetrical phasor diagram (i.e. a balanced three-phase network) this factor will be true as the ratio between any line voltage to phase voltage in a Wye network.

This same factor applies to balanced Delta-connected networks as the ratio between line current and phase current, and for precisely the same reason. Consider the following diagrams for a Delta-connected AC generator outputting 12 Amperes through each phase winding (to a load that is not shown):



Applying the Law of Sines to the solution of currents at each node:

$$\frac{\sin 120^o}{I_{line}} = \frac{\sin 30^o}{I_{phase}}$$

$$\frac{\sin 120^o}{I_{line}} = \frac{\sin 30^o}{12 \text{ Amperes}}$$

$$I_{line} = (12 \text{ Amperes}) \left(\frac{\sin 120^o}{\sin 30^o} \right)$$

$$V_{AB} = 20.784 \text{ Amperes} \approx 20.8 \text{ Amperes}$$

The ratio $\frac{\sin 120^o}{\sin 30^o}$ is equal to the square-root of three ($\sqrt{3}$), and with a symmetrical phasor diagram (i.e. a balanced three-phase network) this factor will be true as the ratio between any line current to phase current in a Delta network.

5.2 Electrical safety

A subject of extreme importance to all electrical practitioners is electrical *safety*, with Ohm's and Joule's Laws being excellent starting points for a discussion on that topic. Here we examine the human body as an electrical *load*: electrical charge carriers passing through the resistance of the body from an external source relinquish some of their energy in the same way charge carriers lose energy passing through any other resistance. The rate of energy dissipation (i.e. *power*) through the body's resistance is predicted by Joule's Law, $P = I^2R$. The total amount of energy delivered to a body by an electric current is a function of that power dissipation rate multiplied by the amount of time current flowed[1].

Electrical energy poses two distinctly different threats to any living body: the first threat is forced activation of the body's nervous system by electric current passing through nerve cells, and the second threat is burning from the thermal power dissipated in flesh and bone. Both threats are direct functions of the amount of energy delivered to the body, with the first effect (called *electric shock*) beginning at lower levels of current than the second effect.

Electric shock – not to be confused with the general condition of *circulatory shock* characterized by reduced blood circulation in the body – first manifests as a tingling sensation, then as pain with greater electric current intensity. At a certain threshold value, the current will be sufficient to override voluntary muscle control. At higher levels of current, breathing will become difficult or may cease due to paralysis of the diaphragm muscles within the chest. At even higher levels of current, the heart (itself a muscle of the body) will either fall into an arrhythmic beat pattern or cease beating altogether. *All of these effects will occur at current levels significantly less than one Ampere.*

Some of the most detailed data we possess on the effects of electric shock come from the research of University of California Berkeley Professor Charles Dalziel, who in the year 1961 published a report entitled "Deleterious Effects of Electric Shock". Dalziel performed electric shock experiments on human volunteers, subjecting both males and females to varying degrees of electric current, both direct (DC) and alternating (AC), for the purpose of determining thresholds of sensation, pain, and loss of muscular control.

Table II of Dalziel's' report (shown on page 24) is partially[2] reproduced in the following table. The headings "M" and "F" refer to male and female subjects, respectively. Tests conducted using direct current[3] are labeled "DC" while tests conducted using alternating current[4] are labeled with

[1] Putting units of measurement to this concept, the amount of energy in *Joules* is equal to average power in *Joules per second* multiplied by time in *seconds*, with the unit of "seconds" canceling out. For brief exposures to electricity, such as lightning strikes, the most important measurement with regard to safety is the total energy delivered to the body. The same is true for deliberate applications of electricity to the body, for example cardiac defibrillators, where the machine's setting is calibrated in Joules of energy delivered per impulse.

[2] The original Table II contained a column of data representing thresholds for women at 10 kHz alternating current, but these were estimations and not actual data. Extrapolating from the other data points where women tended to exhibit the same effects as men at approximately $\frac{2}{3}$ the current, Dalziel writes, *"Tests on women were not made on frequencies other than 60 cycles, but if it is assumed that the response for women would be similar, values for women can be estimated at two-thirds of the corresponding value for men."* Readers should note that I have taken editorial liberties with the description of bodily effects, for no reason other than formatting.

[3] Direct current, or DC, refers to a continuous flow of electric charge carriers in one direction only.

[4] Alternating current, or AC, refers to an electric current that periodically switches direction, the period of that

frequency values expressed in the unit of Hertz (Hz) or cycles per second. All data points are expressed in *milliAmperes* (mA), one milliAmpere being $\frac{1}{1000}$ of an Ampere:

Bodily effect	**DC, M**	**DC, F**	**60 Hz, M**	**60 Hz, F**	**10 kHz, M**
Slight sensation felt on hand	1 mA	0.6 mA	0.4 mA	0.3 mA	7 mA
Median perception threshold	5.2 mA	3.5 mA	1.1 mA	0.7 mA	12 mA
Shock, with no loss of muscular control	9 mA	6 mA	1.8 mA	1.2 mA	17 mA
Pain, with 50% of subjects losing muscular control	62 mA	41 mA	9 mA	6 mA	55 mA
Pain, labored breathing, 99.5% of subjects losing muscular control	90 mA	60 mA	23 mA	15 mA	94 mA

For rather obvious reasons no human tests were conducted to the point of cardiac fibrillation. Dalziel's report does, however, provide data collected on a variety of animals (pigs, sheep, calves, dogs, cats, guinea pigs, rabbits) which were anesthetized and then administered large amounts of electric current until their hearts malfunctioned. From this admittedly limited data, Dalziel extrapolated the values to obtain 500 mA ($\frac{1}{2}$ Ampere) of direct current and 100 mA ($\frac{1}{10}$ Ampere) of alternating current as thresholds for "possible" human heart fibrillation following a three-second electric shock.

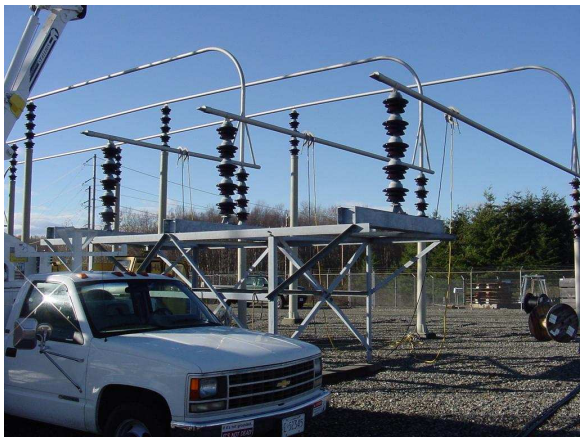
All gruesome details aside, the lesson to be learned here is very plain: *very little electric current is necessary to induce painful and even life-threatening effects on the human body!* These danger thresholds are all substantially less than the amount of current most power conductors are rated to handle, and less than the ratings of fuses and circuit breakers designed to protect conductors from overheating.

switching measured in cycles per second or Hertz. In North America, the standard AC grid power frequency is 60 cycles per second, or 60 Hz. The second AC frequency used in Dalziel's experiments is 10 kHz, which is 10 *kilo*-Hertz, or 10000 cycles per second.

The first line of defense against electrical shock is to place as much electrical resistance between your body and the circuit's conductors as is practical, as a means of impeding the flow of electric current to and through your body. Turning "off" any disconnecting switches between the circuit and its energy source is a simple means to do this, essentially inserting an air gap between the circuit and its normal source of power. This allows all points within the circuit to achieve an equipotential state, which may then be made equipotential to your body by connection to Earth ground (where you are standing). If there is no voltage present (i.e. no difference in the potential energy levels of electric charge carriers at different points), then there should be no possibility of dissipating electrical energy into your body.

Once all electrical energy sources have been disconnected from the circuit you intend to work on, an additional safety measure is to *bond* that circuit's power conductors to Earth ground. This step forces the power conductors to be *electrically common* with Earth, and therefore guarantees a condition of equipotentiality with the Earth. Line workers who install and maintain electric power line conductors do this as a standard part of their operating procedure: attaching temporary grounding cables between the power conductors and Earth after opening all disconnect switches normally connecting those lines to electrical sources. This extra step of bonding the power conductors ensures no stray sources[5] of electrical energy may pose a threat.

The following photograph shows a work site at a 230 kV (230,000 Volt!) electrical substation, where electricians are busy performing maintenance work on a high-voltage component. In addition to opening large switches (called *disconnects*) to isolate this new component from any source of voltage, they have *bonded* the high-voltage conductors to each other and to Earth ground by means of temporary wire cables. The cables on this work site happen to be yellow in color, and may be seen hanging down from C-shaped clamps attached to three horizontal metal tubes called *busbars* which serve as conductors for electricity in this substation:



[5] Examples include electrostatic or magnetic "coupling" with adjacent energized power lines, nearby lightning strikes, etc.

Obviously, such measures are quite necessary on high-voltage systems such as substation busbars – there simply is no safe way to work on energized conductors at this voltage level. However, in lower-voltage circuits it is often necessary to take electrical measurements and make certain adjustments while the circuit is in an energized state. If the circuit in question cannot be "killed" by disconnection of its power source and therefore must be worked on "live", the next best protective measure is to layer insulating material on your body where contact might otherwise be made to permit an electric current through it. This means wearing insulating gloves and shoes, at minimum. The principle behind this technique is Ohm's Law: for any given amount of voltage (V), current (I) will be inversely proportional to the total resistance (R) of the circuit pathway. Layering electrically insulating material over your body's possible points of contact (e.g. hands, feet) increases the total resistance of the circuit pathway, and therefore minimizes the amount of current that may flow in the event of physical contact between two points where a substantial voltage exists.

Lastly, in order to minimize the risk of electric current passing through one's chest (where the heart and diaphragm muscles are located), a wise habit when working on energized circuits is to place one hand in a pocket so that only one hand is in use. This will not only prevent arm-to-arm passage of electric current, but it also minimizes the number of potential points of contact with bare skin. Electrical practitioners commonly refer to this as the *One-Hand Rule*. Ideally, the best hand to place in a pocket is the left hand, because this is the side of the body where the heart is most vulnerable.

It is worth noting that the danger from electric shock is best quantified in terms of current, not voltage, since it is electric current that activates nerve cells. The amount of current passing through a victim's body from an applied voltage is a function of Ohm's Law ($I = \frac{V}{R}$), and since resistance (R) varies greatly with skin dryness and layering provided by shoes and clothing, it is difficult to predict how much voltage poses a shock hazard. A generally accepted threshold of danger is 30 Volts, but this assumes direct contact with dry skin. Moist skin, perspiration, cuts or punctures, and other factors reducing body resistance may greatly reduce the voltage threshold for shock hazard! Another factor is the general health of the victim prior to receiving the electric shock. A preexisting cardiac condition will likely predispose that individual to harm resulting from an electric shock.

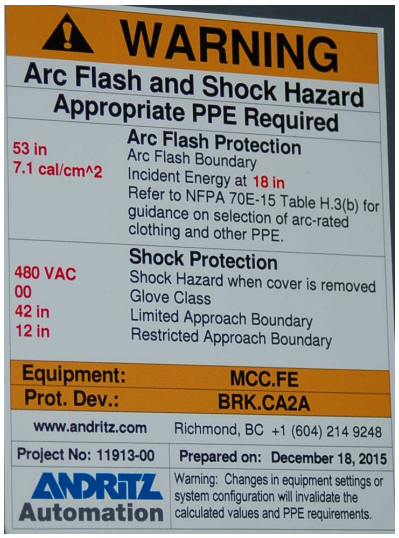
Burns produced by electricity passing through the body may manifest on the skin, at the point of contact with an electrical conductor (such as a wire), or in severe cases may extend to internal organs. Comparing internal flesh with skin, dry human skin tends to exhibit much greater levels of electrical resistance than the internal organs which are wet. This is why electricity causes skin-surface burns before causing internal organ burns: for any given amount of electric current passing through different resistances, power dissipated by that current will be greatest wherever the electrical resistance is greatest. Mathematically stated, P is maximized where R is greatest, given any value of (I), in accordance with Joule's Law (I^2R).

Another mechanism of electrically-caused burns is *arc flash*: the heating of air by the passage of electric current through it (rather than through the body). Under normal conditions air is an extremely good insulator of electricity, with no free charge carriers available to sustain an electric current. However, when sufficient voltage causes the electrons in air molecules to separate from their respective atoms, the negatively-charged electrons and positively-charged ions constitute charge carriers, and will form an electric current called either a *spark* or an *arc*. This current heats the air molecules by dissipating power as described by Joule's Law ($P = I^2R$), with I being the magnitude of current traveling through the ionized air and R being the resistance of the arc path.

The amount of resistance exhibited by a high-temperature arc is surprisingly low, typically less than one Ohm across the entire length of the arc. With such low resistance, Ohm's Law predicts relatively high current values for even modest voltages ($I = \frac{V}{R}$), resulting in high power levels. Even if the amount of energy released by each charge carrier moving through the arc is small, the fact that a great many charge carriers are moving through the arc each second means that the total amount of energy dissipated may be phenomenally large. This is why arcs forming in high-voltage electric power systems may reach temperatures of tens of thousands of degrees[6]!

In the United States of America, a widely respected standard document for electrical hazards and protection is the National Fire Protection Association (NFPA) standard 70E. This document rates both electric shock and arc flash hazards for electric power circuits based on voltage and current capabilities, as well as specifies best practices for protection against those hazards.

An example of NFPA 70E standards applied to an industrial installation is the following pair of photographs showing warning labels affixed to metal-clad electrical switchgear (i.e. metal cabinets housing large circuit breakers). Each label cites both arc flash and electric shock hazards, including *boundary* distances within which greater hazards exist:



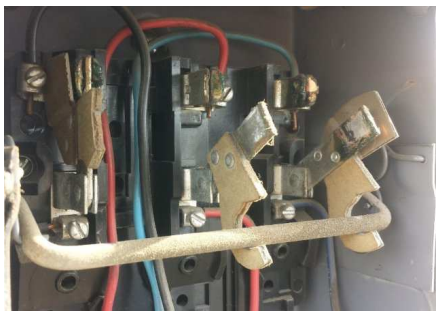
The first line of defense against arc flash is the same as for electric shock: *de-energize the circuit* so there will be no electrical energy present to harm you. The procedure for de-energizing includes placing a warning *tag* as well as a secure *lock* on any main disconnecting switches or circuit breakers to ensure power does not get accidently applied to the circuit while people are in harm's way. This is referred to in industry as *lock-out, tag-out*, or *LOTO*.

[6]The NFPA 70E electrical safety standard (Informative Annex K) cites temperatures as high as 35,000 degrees Fahrenheit in arc flash events, and states that such events are lethal at distances up to 10 feet (3.05 meters). It is worth noting that electric arc temperatures are limited only by the rate of power dissipated in the arc. Unlike chemically-driven combustion events, where temperature is limited by the rate at which the various chemical reactants are able to combine, no such limiting factor exists with electric arcs: the more power dissipated in the arc, the hotter it will become. These temperatures involved with electric power faults can be so high that they *vaporize* the metal wires!

Generally accepted lock-out, tag-out procedures for electrical equipment include (but are not limited to) the following steps:

1. Turn equipment off using regular on/off switch to interrupt load current
2. Open all disconnect switches and/or circuit breakers feeding electrical power to the equipment
3. Visually confirm (if possible) that *all* poles of the disconnecting switch(es) are open
4. Attempt to re-start equipment as your first confirmation that power is indeed shut off
5. Attach locks and tags to prevent disconnecting switches from being re-closed; if no lockable disconnect means exists, you must open and tag *two* or more redundant disconnects!
6. Confirm proper operation of voltage meter against a known source
7. Test for dangerous voltage between all relevant conductors, especially those you could bodily contact, including between pairs of phase conductors in a polyphase system as well as between each phase conductor and earth ground
8. Confirm (again) proper operation of voltage meter against a known source

Such a lengthy procedure may seem paranoid, but there is good reason for all these steps. Remember that *all safety standards and procedures are written in blood:* in other words, every one of them exists because either someone got hurt or came close to getting hurt by not taking appropriate precautions. One example of the latter is the following photograph generously shared by Jim Lyon[7] who discovered a failed three-phase disconnect switch during a routine LOTO procedure:



As you can see in this photograph, one of the linkages between the disconnecting switch pole and the actuating lever has broken, allowing one phase to remain closed while the other two opened as they should. Such a failure would neither be apparent to the operator of that switch nor by anyone's external inspection of it, but would be revealed by proper voltage testing (phase-to-ground) on that faulted phase conductor. Had Mr. Lyon simply trusted the switch handle's "off" position as proof of a zero-energy state, he could have been electrocuted touching the un-opened conductor "downstream" of the switch.

[7] Mr. Lyon granted me permission to include his safety-related photographs in my open-source writings during an exchange on the professional social-media website `LinkedIn`.

Another important note with regard to electrical safety and LOTO is the proper function of your test equipment. Multimeters and test leads are not just tools for precision measurement, but are life-saving indicators of dangerous voltage. I have personally witnessed multiple instances of failed multimeter test leads, where one or more leads failed open such that the meter could not reliably register the presence of voltage. This is why good LOTO procedure always includes verification of the meter both *before and after* checking for dangerous voltages at the system conductors.

In cases where de-energization is not possible or not practical, special "arc-flash rated" clothing may be worn to protect your skin against the high temperatures of arc flash should an arc flash occur. Arc flash suits cover all skin surfaces, and are rated according to the number of calories[8] of heat the fabric may sustain without disintegrating. The following photograph shows a pair of arc flash suits hanging on a wall ready for electricians to use while working on circuit breakers at an electric power generating station:

The blue-colored hood covers the worker's head and neck, while the grey-colored jumpsuit covers the rest of the worker's body.

[8] A *calorie* is simply another unit of energy measurement. The unit-conversion equivalence is 4.187 Joules per calorie.

The hazards of electrical arcs are not limited to bodily burns. Given sufficient arc power, the explosive expansion of hot air and the shrapnel created by disintegrating hardware represents its own unique hazard, known as *arc blast*. As an *electrically-driven* explosion[9], arc blast is limited only by the available power of the fault, and can in fact be more violent than a chemical explosion. No suit can ensure safety against arc blast, and so the only reasonable precaution is maintaining a safe distance beyond the blast radius.

[9]The concussive effects of an arc blast originate from the rapid expansion of air and vaporized metal, producing intense sound waves and blast pressures. Extremely bright light, as well as high temperatures caused by convection of super-heated air and by radiation of infrared light from the arc are capable of creating third-degree burns on unprotected skin.

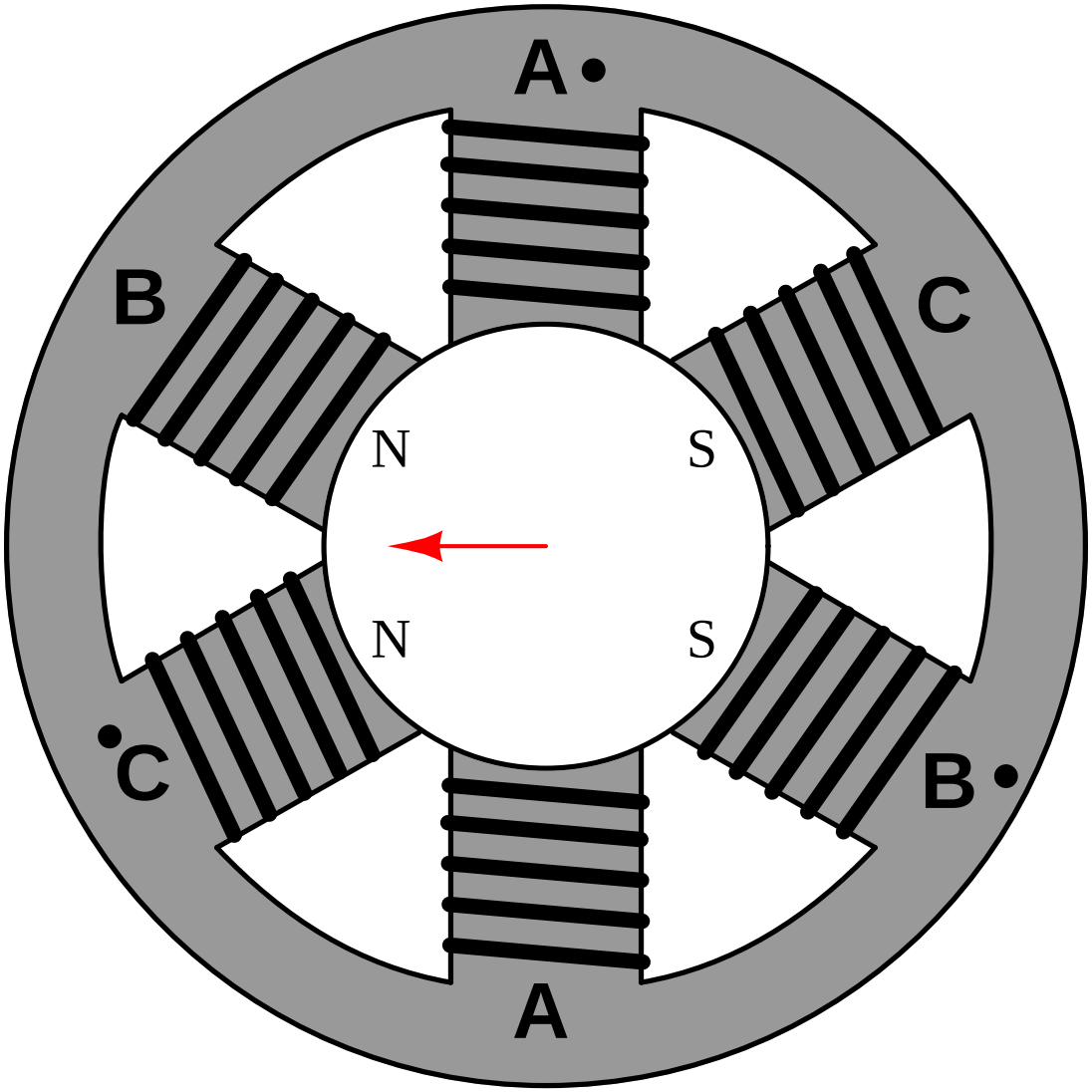
Chapter 6

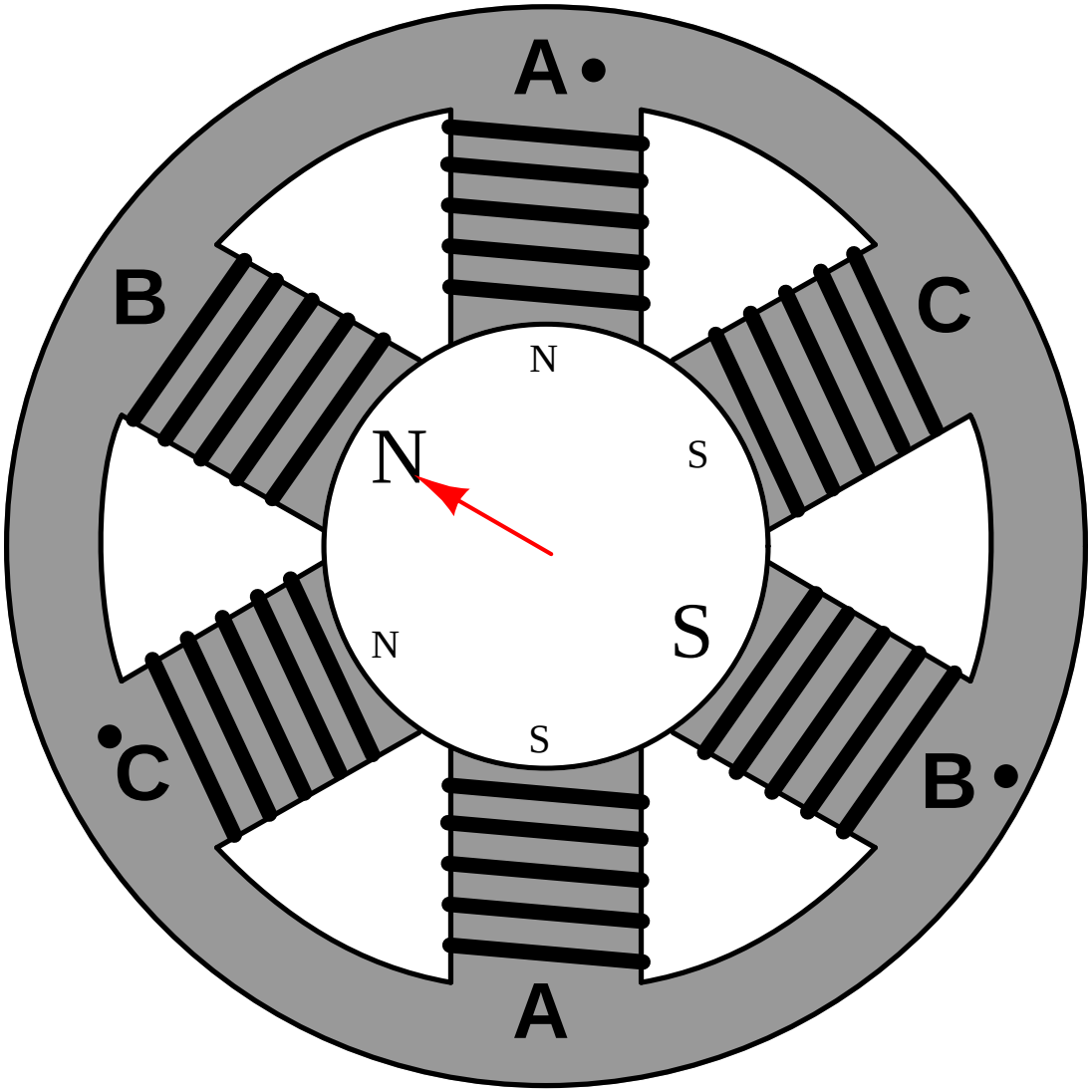
Animations

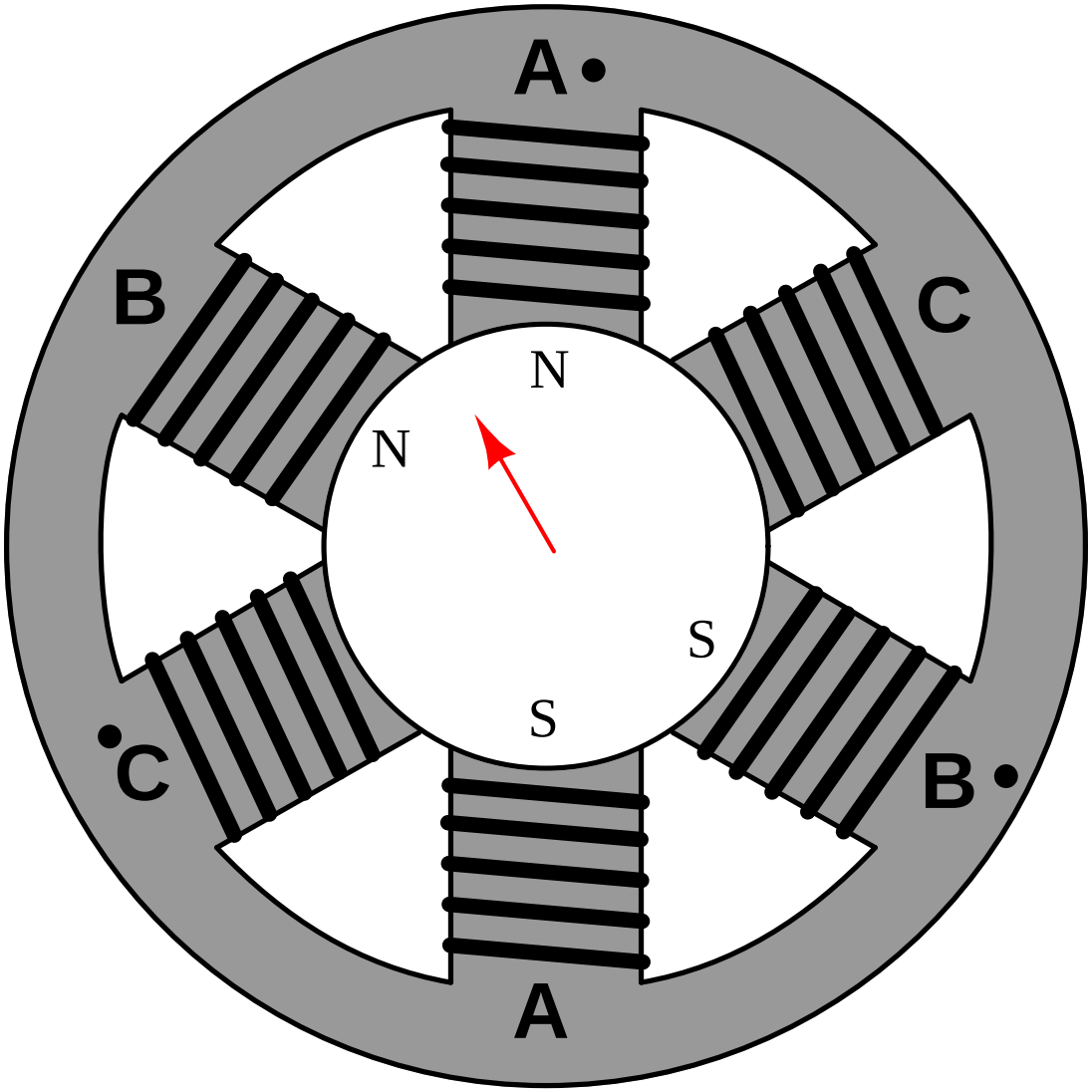
Some concepts are much easier to grasp when seen in *action*. A simple yet effective form of animation suitable to an electronic document such as this is a "flip-book" animation where a set of pages in the document show successive frames of a simple animation. Such "flip-book" animations are designed to be viewed by paging forward (and/or back) with the document-reading software application, watching it frame-by-frame. Unlike video which may be difficult to pause at certain moments, "flip-book" animations lend themselves very well to individual frame viewing.

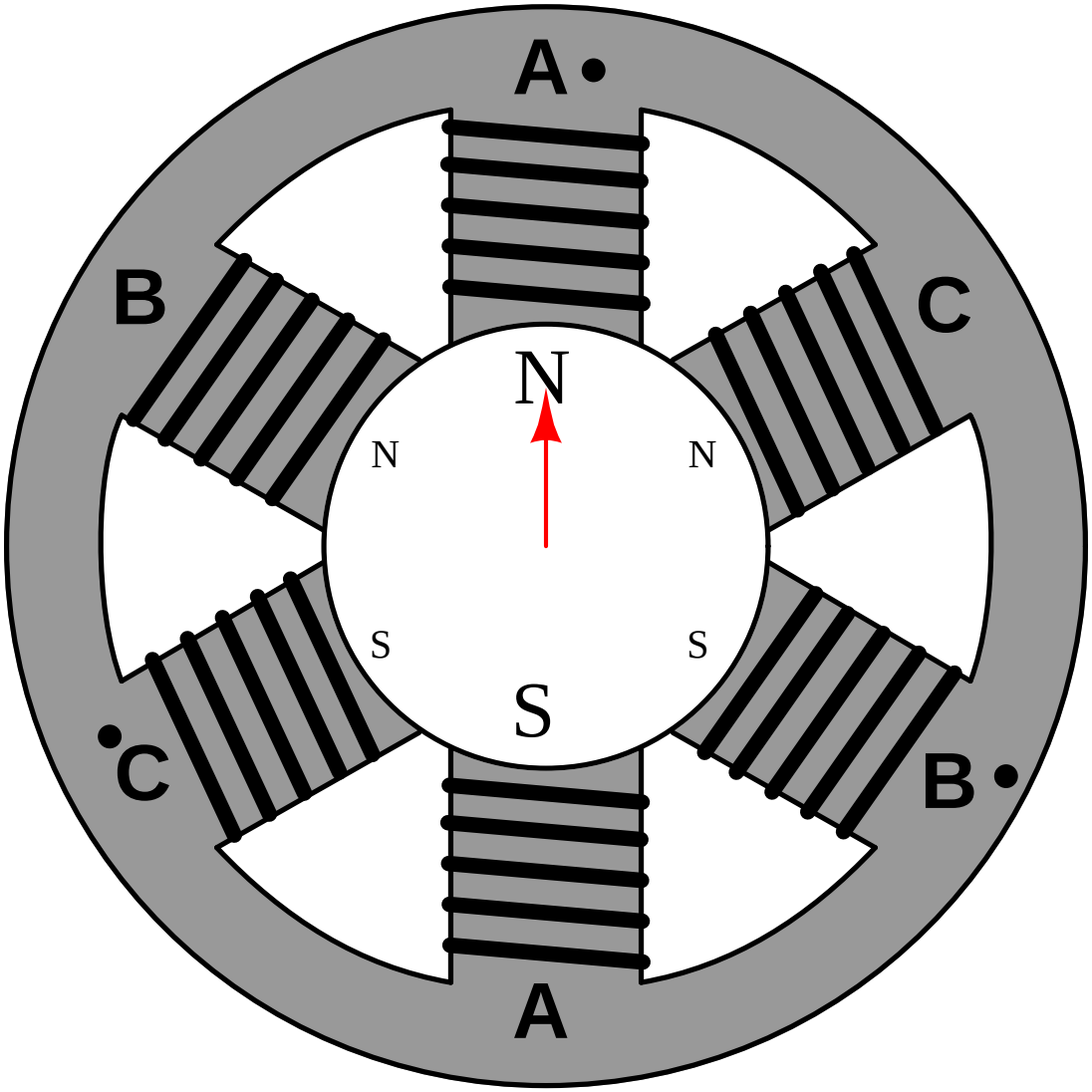
6.1 Rotating magnetic field animated

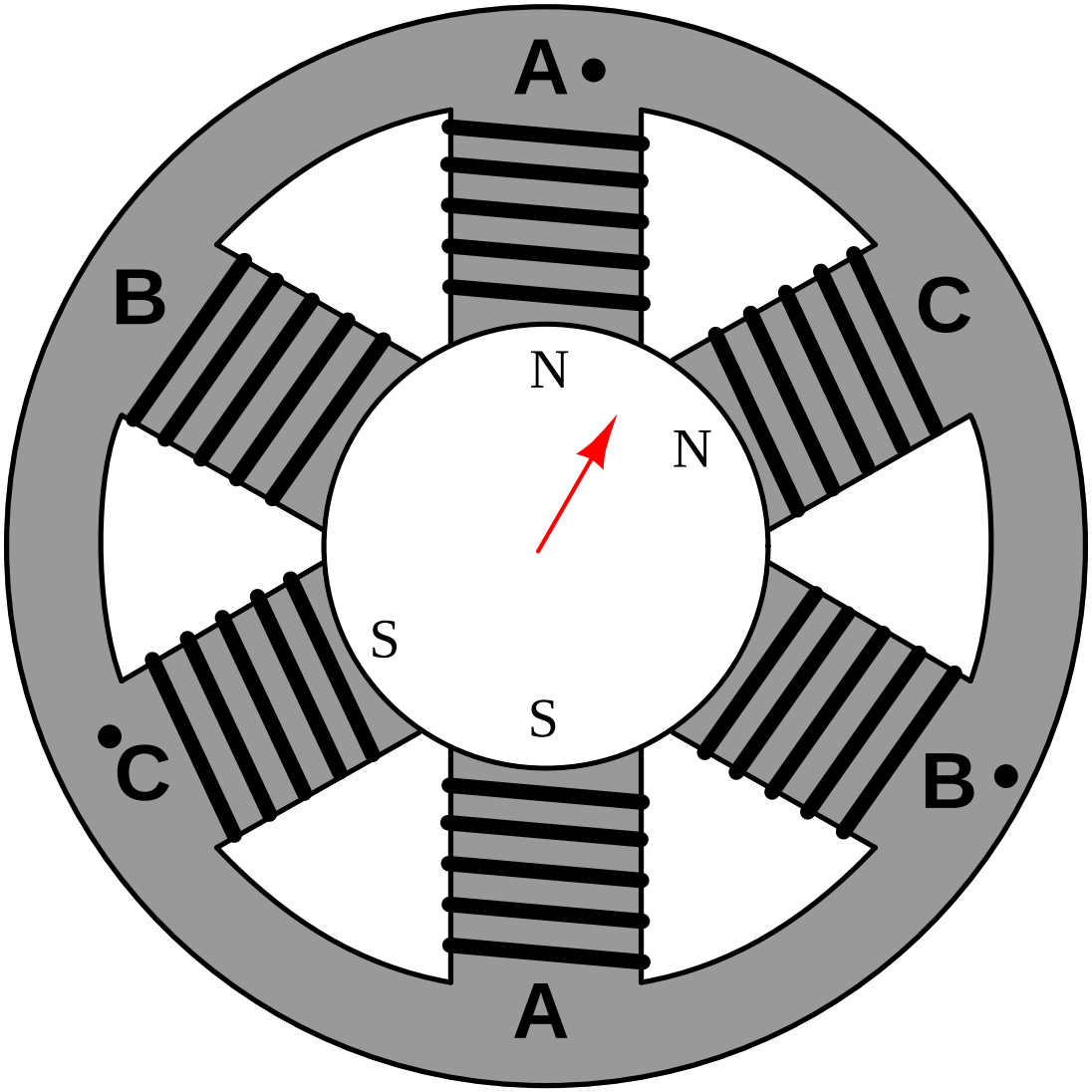
The following animation shows how the "rotating" magnetic field of a three-phase AC induction motor is produced by the interaction of three stator winding sets energized with different phases (A, B, and C) of a three-phase AC power source. A red arrow shows the direction of the *resultant* magnetic field created by the interaction of the three winding sets.

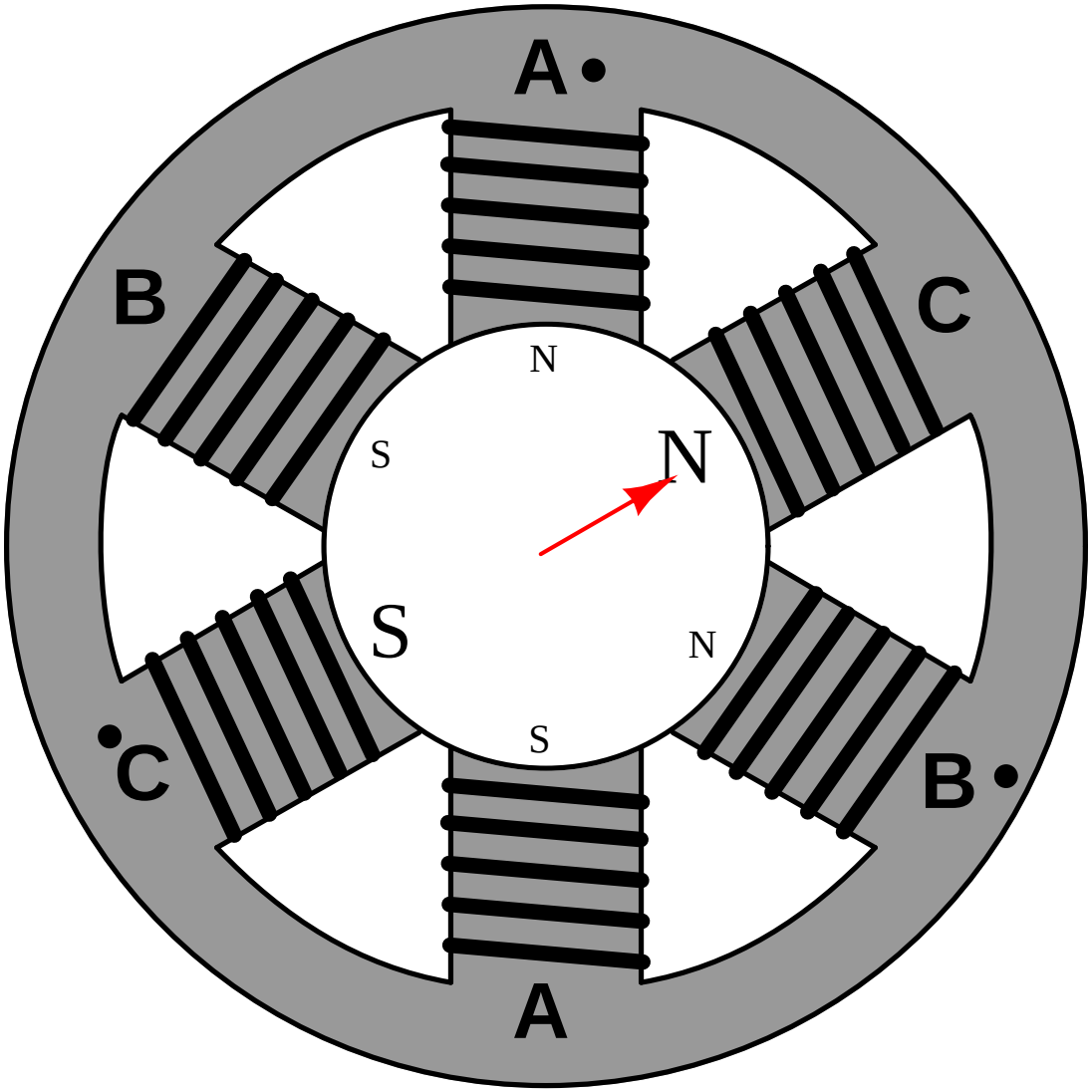

A
B
C
N
S
N
S
C
B
A

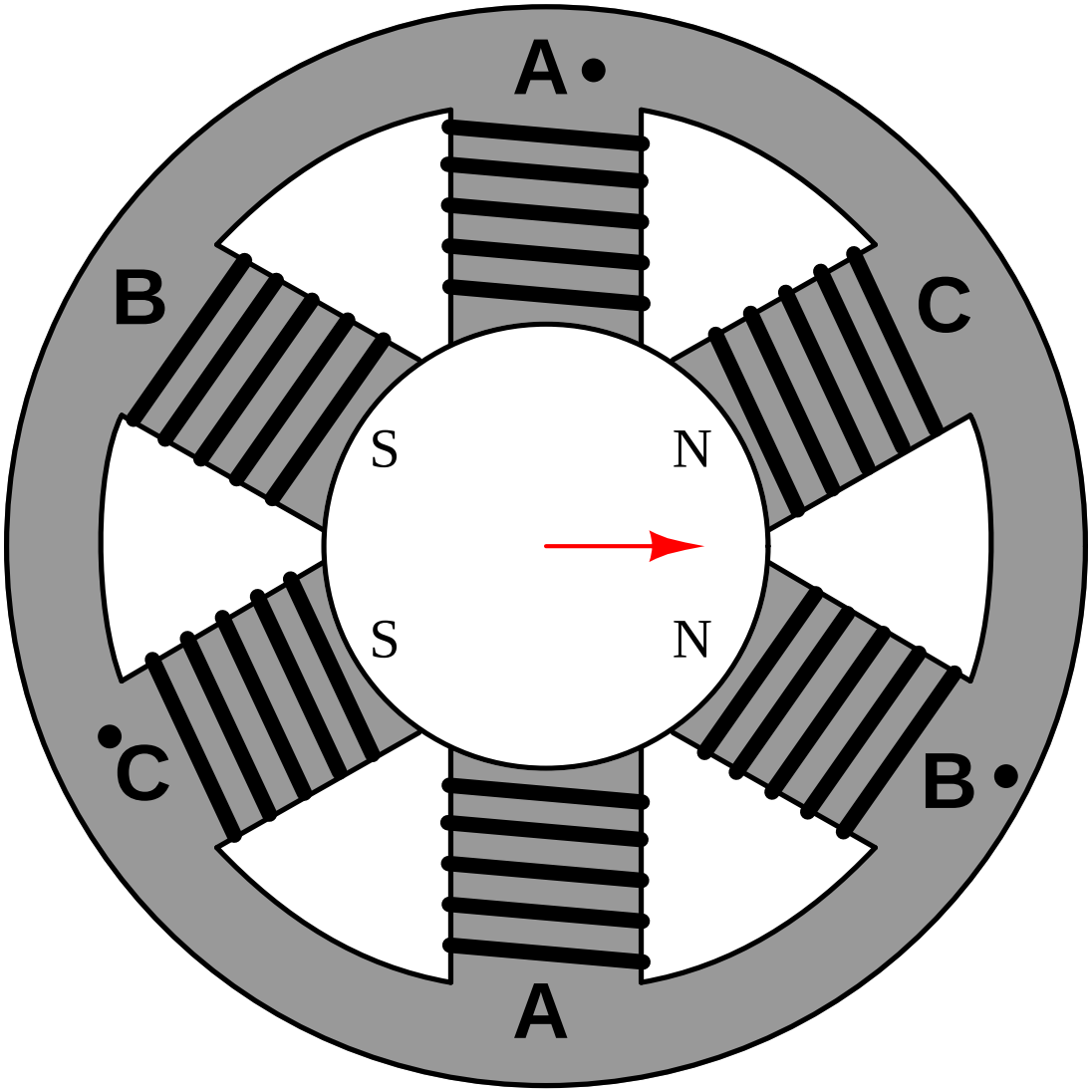

A
B
C
N
S
N
S
N
S
C
B
A

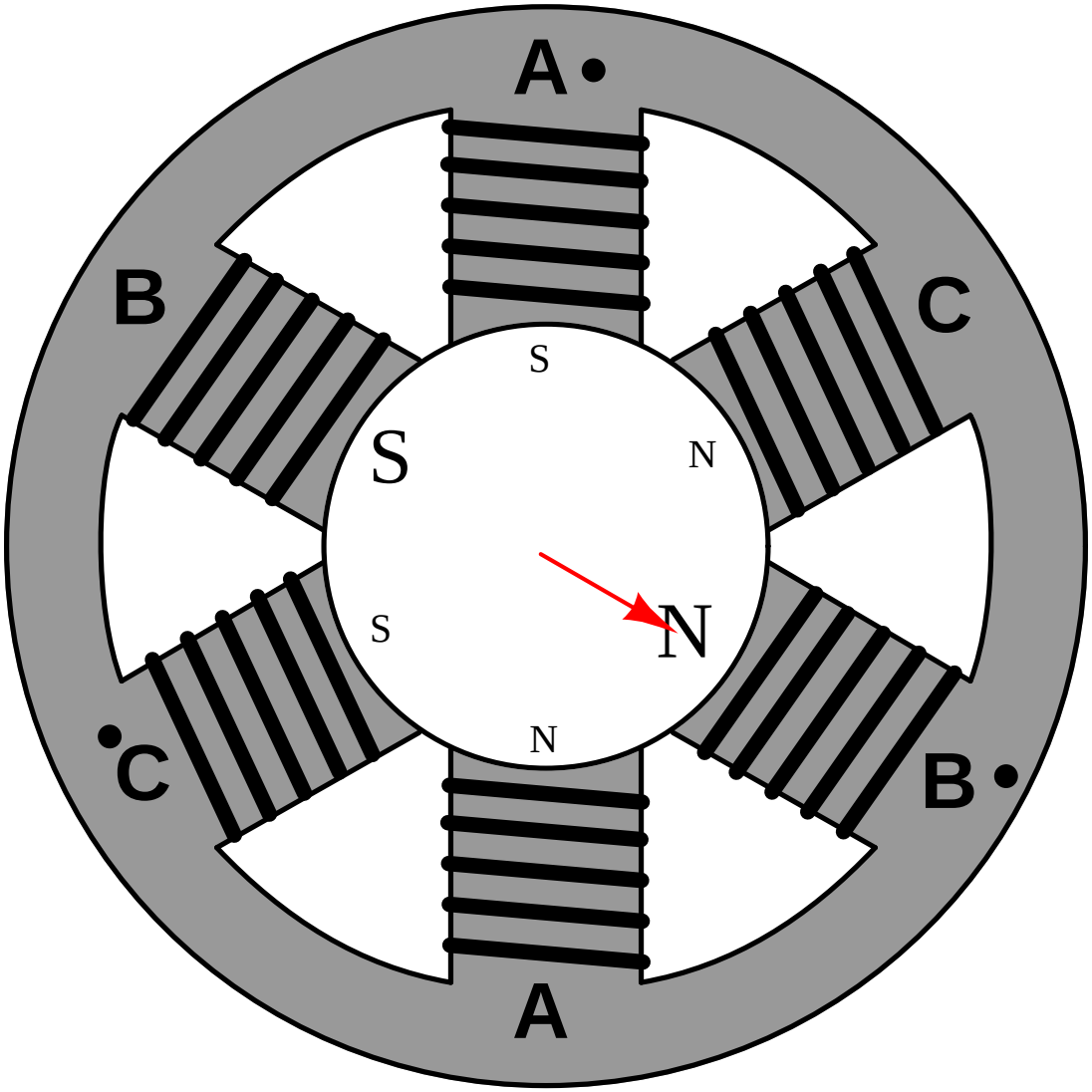

A
B
C
N
N
S
S
C
B
A

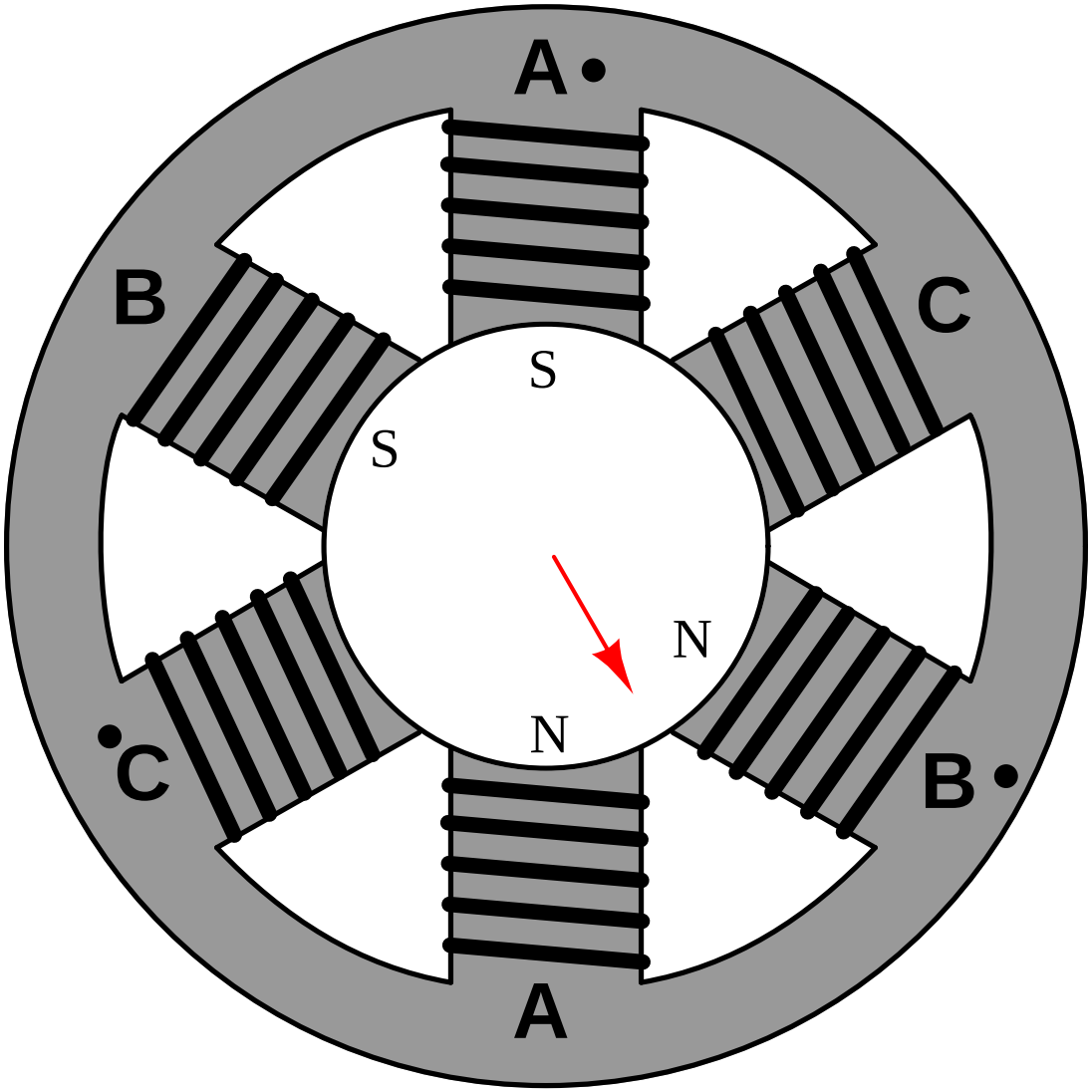

A
B
C
N
N
N
S
S
S
C
B
A

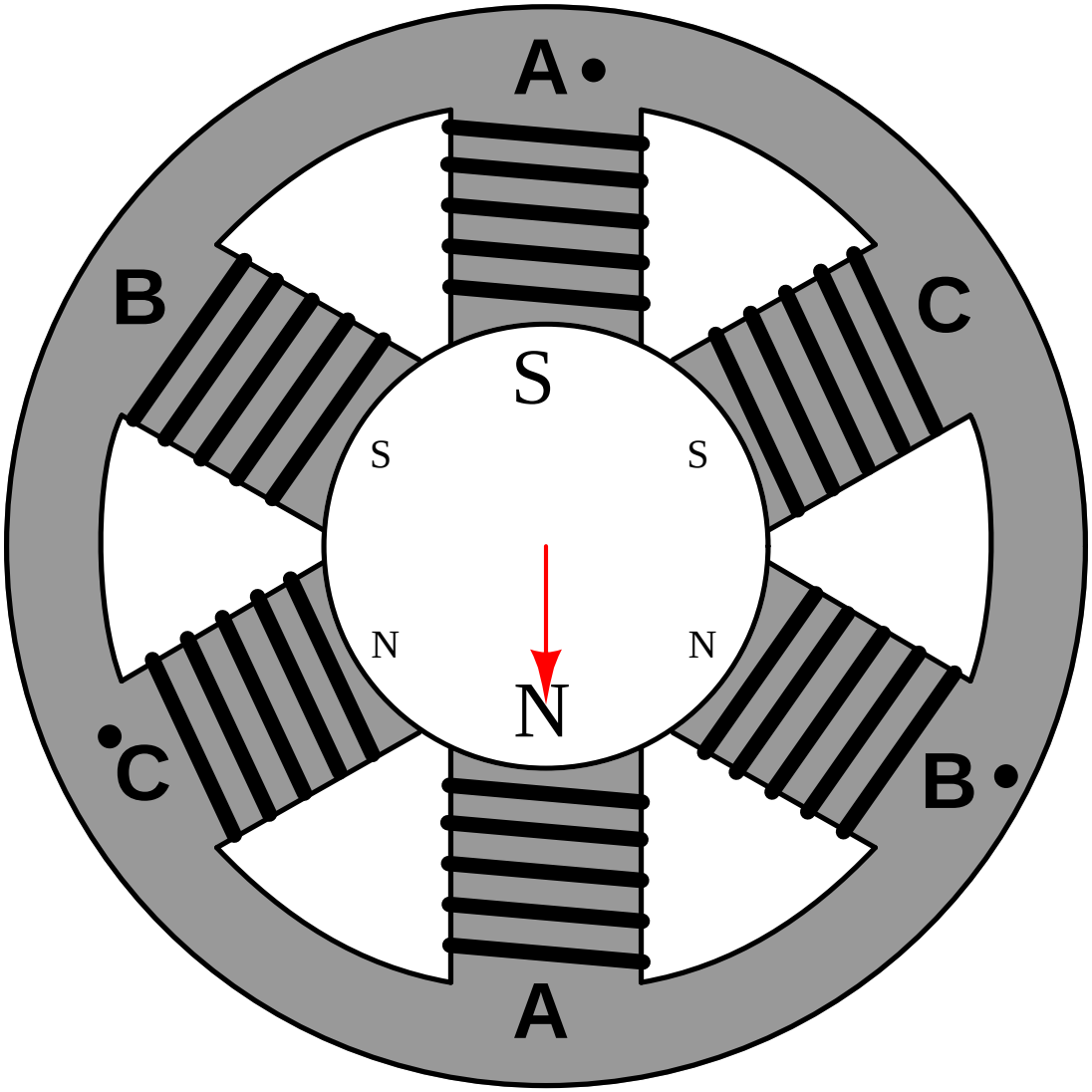

A
B
C
N
N
S
S
C
B
A

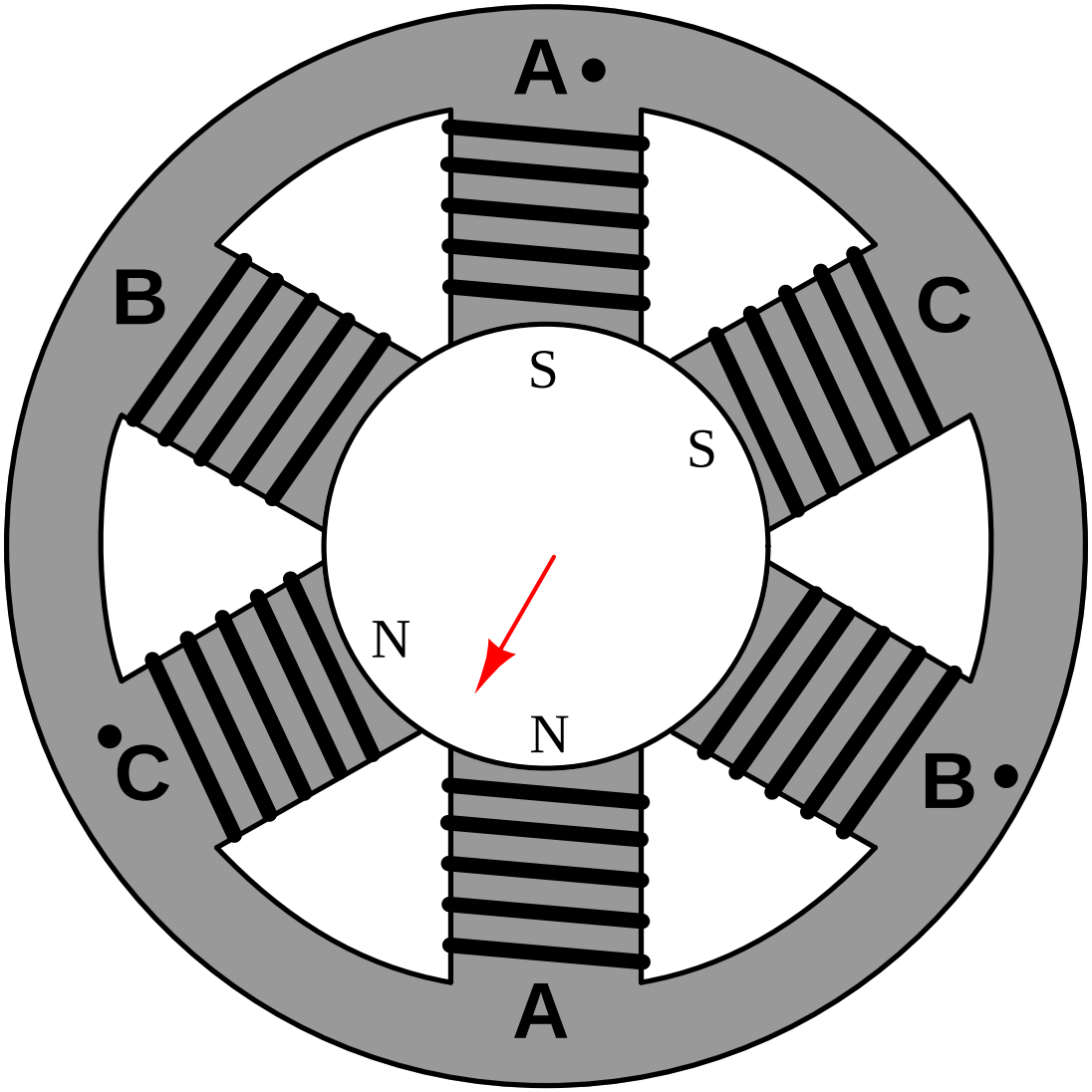

A
B
C
N
S
N
S
N
S
C
B
A

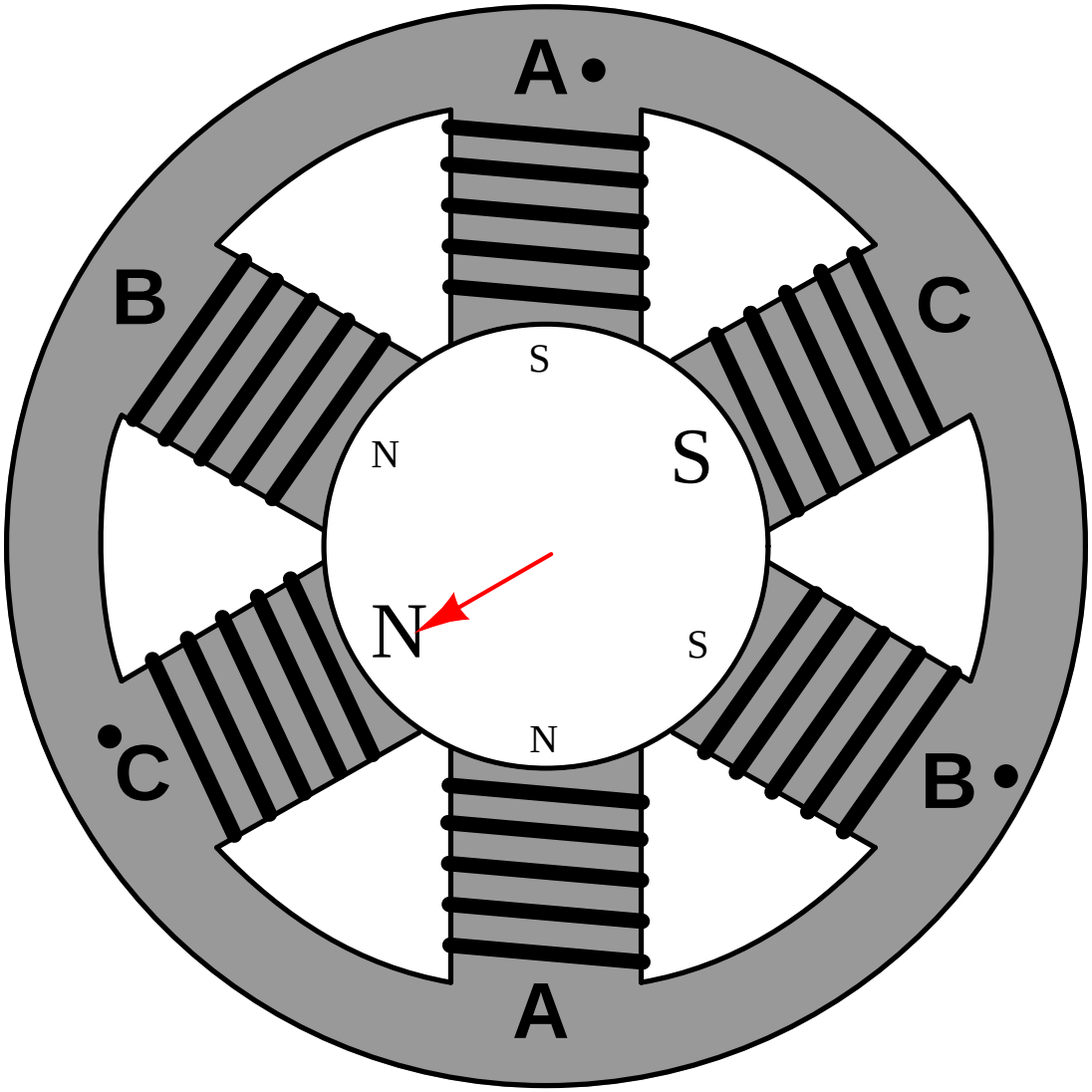

A
B
C
S
N
S
N
C
B
A

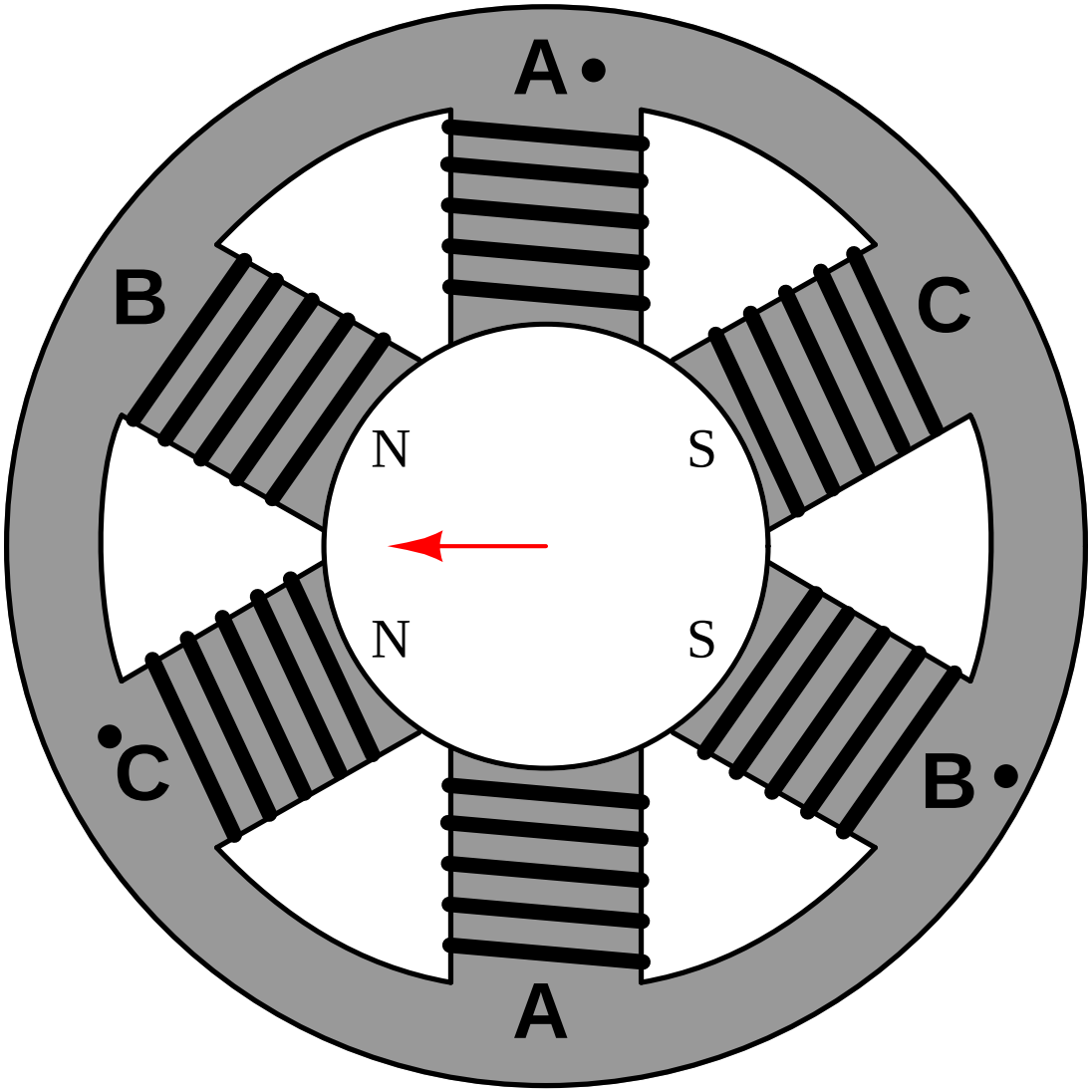

A
B
C
S
N
S
N
S
N
C
B
A

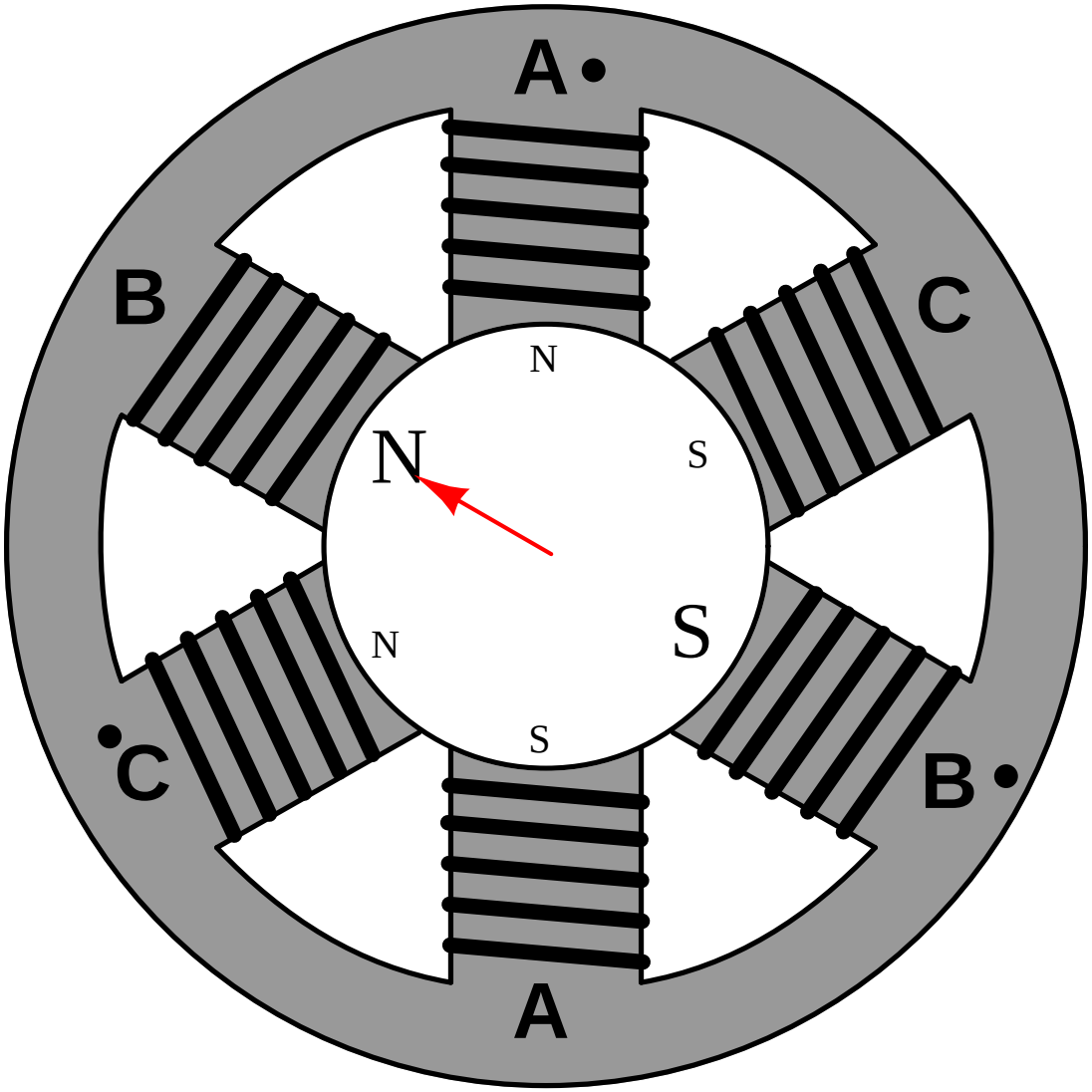

A
B
C
S
S
N
N
C
B
A

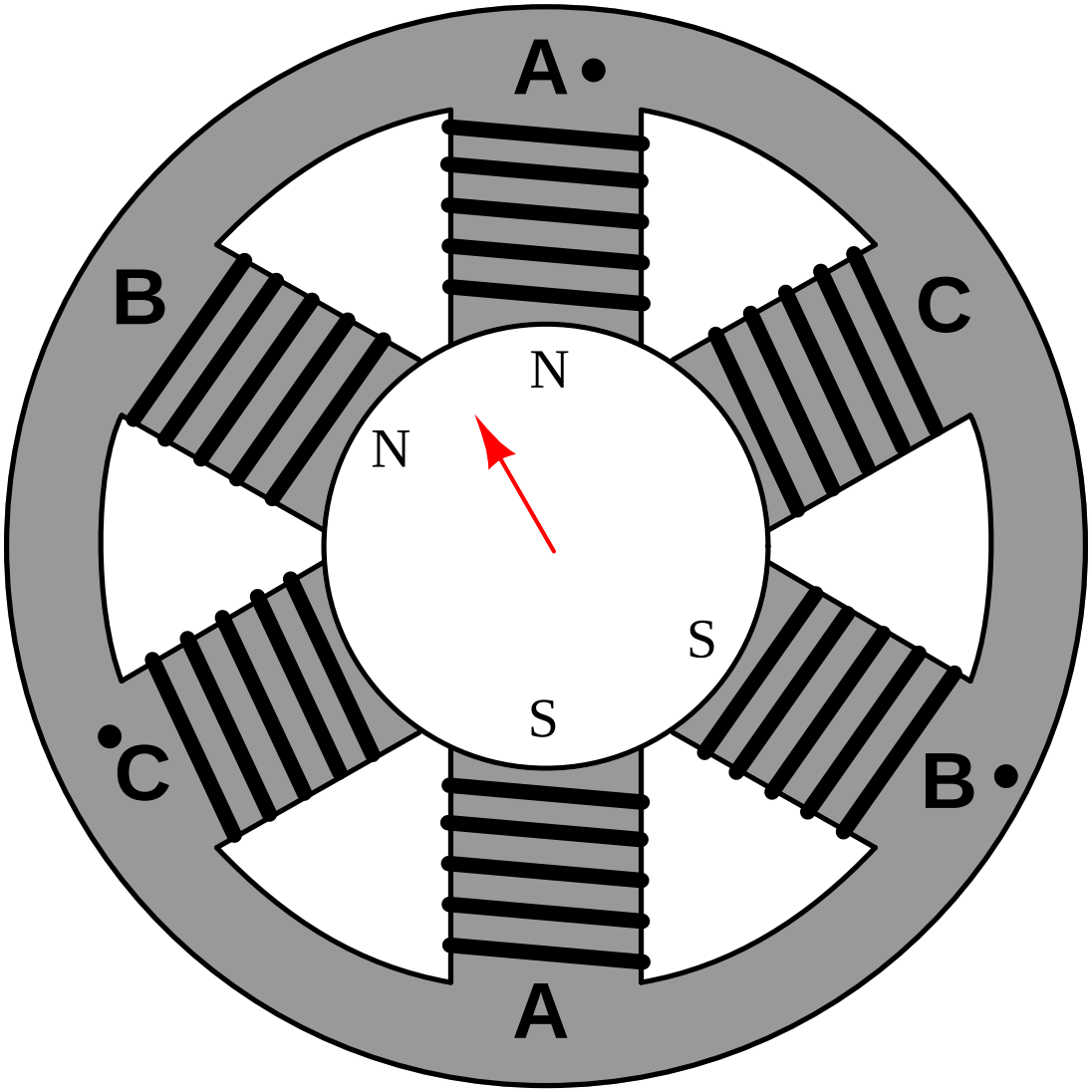

A
B
C
S
S
S
N
N
N
C
B
A

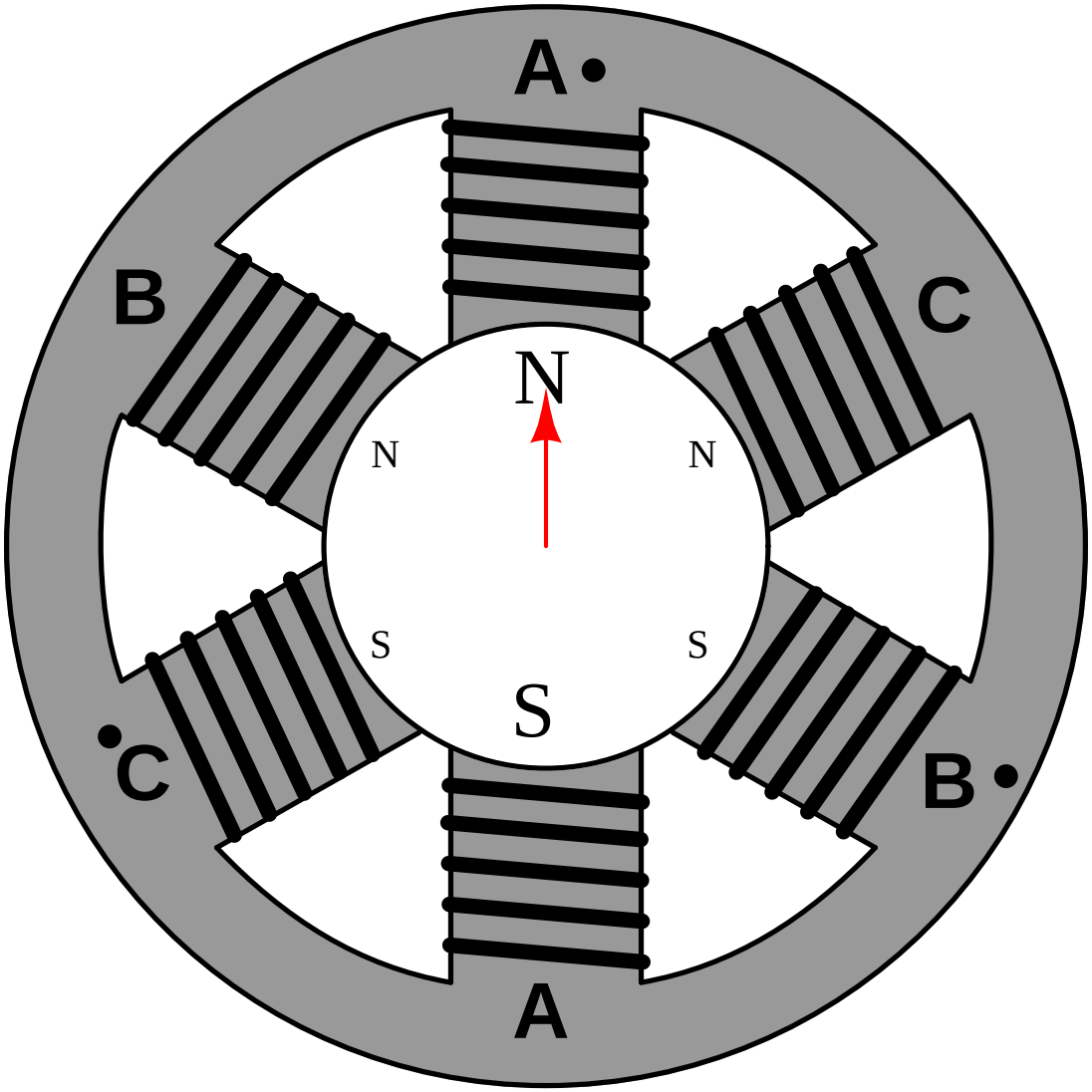

A
B
C
S
S
N
N
C
B
A

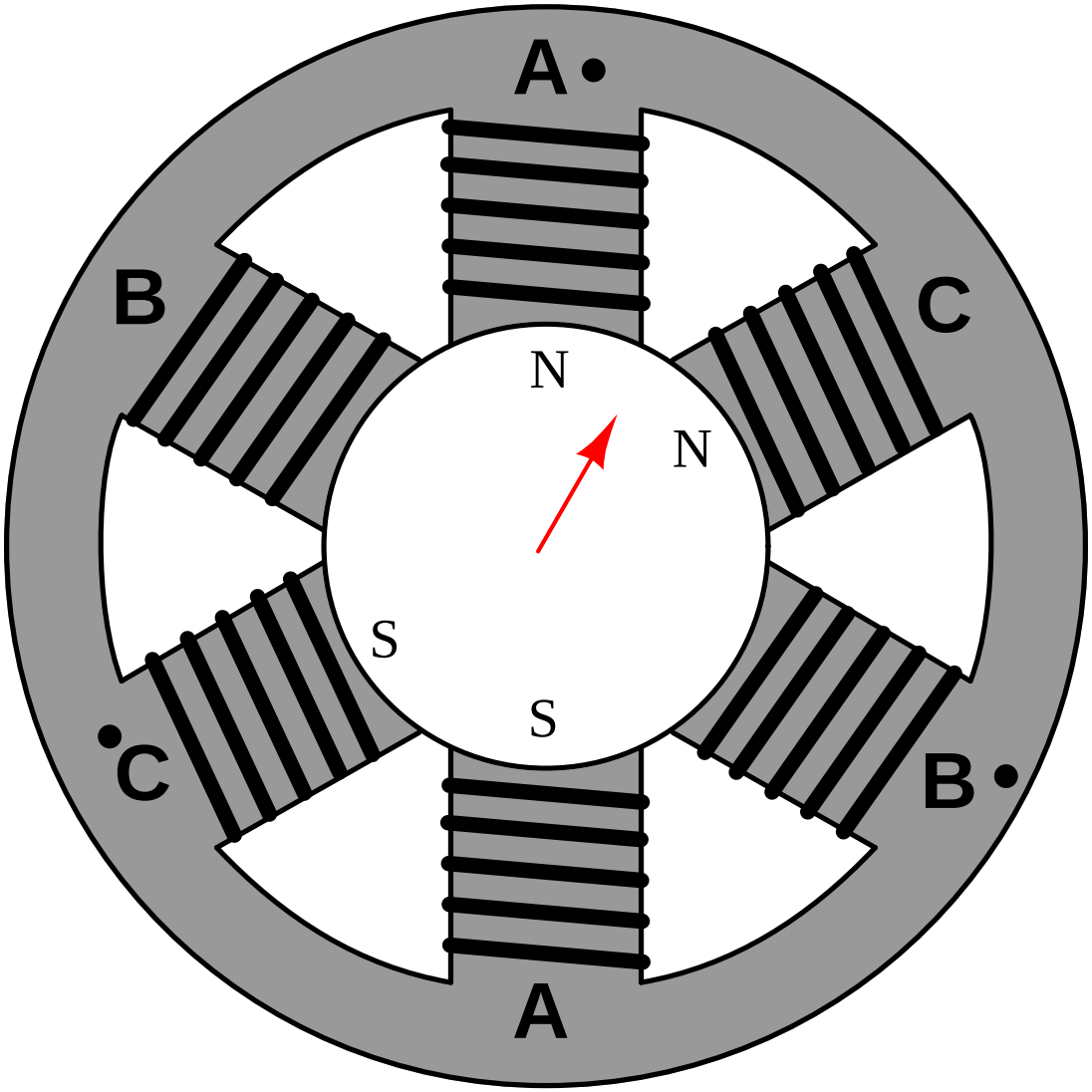

A
B
C
S
N
S
N
S
N
C
B
A

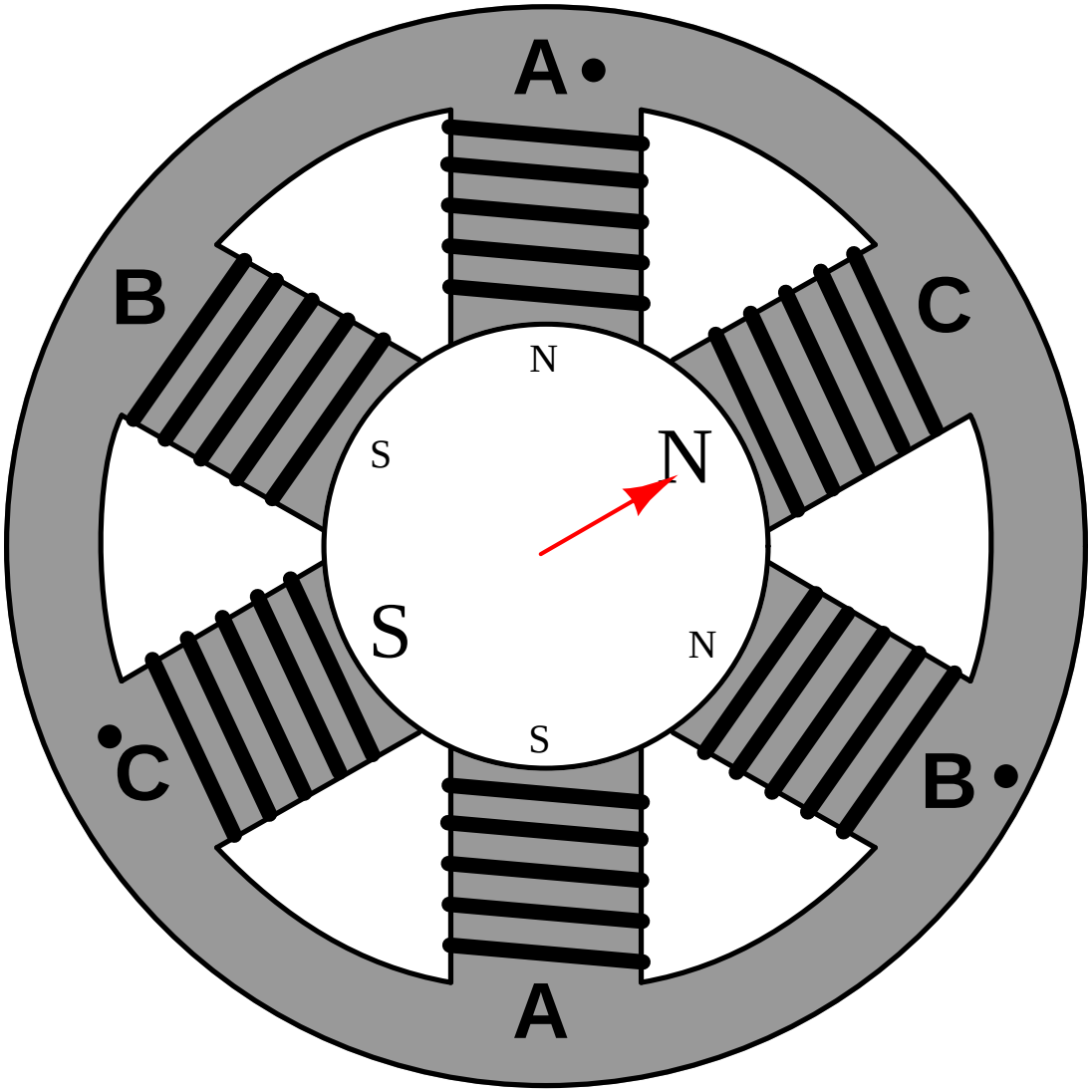

A
B
C
N
S
N
S
C
B
A

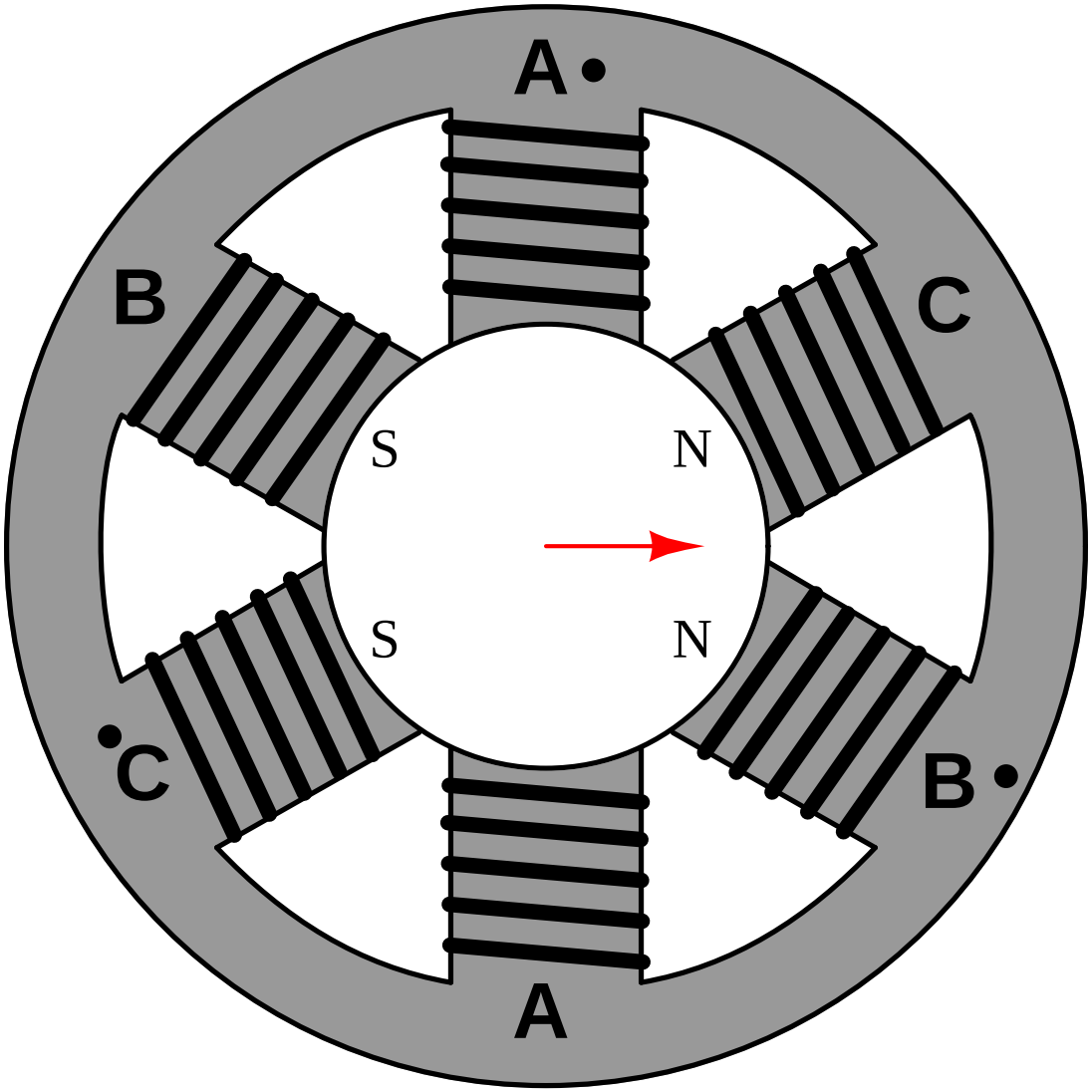

A
B
C
N
S
N
S
N
S
C
B
A

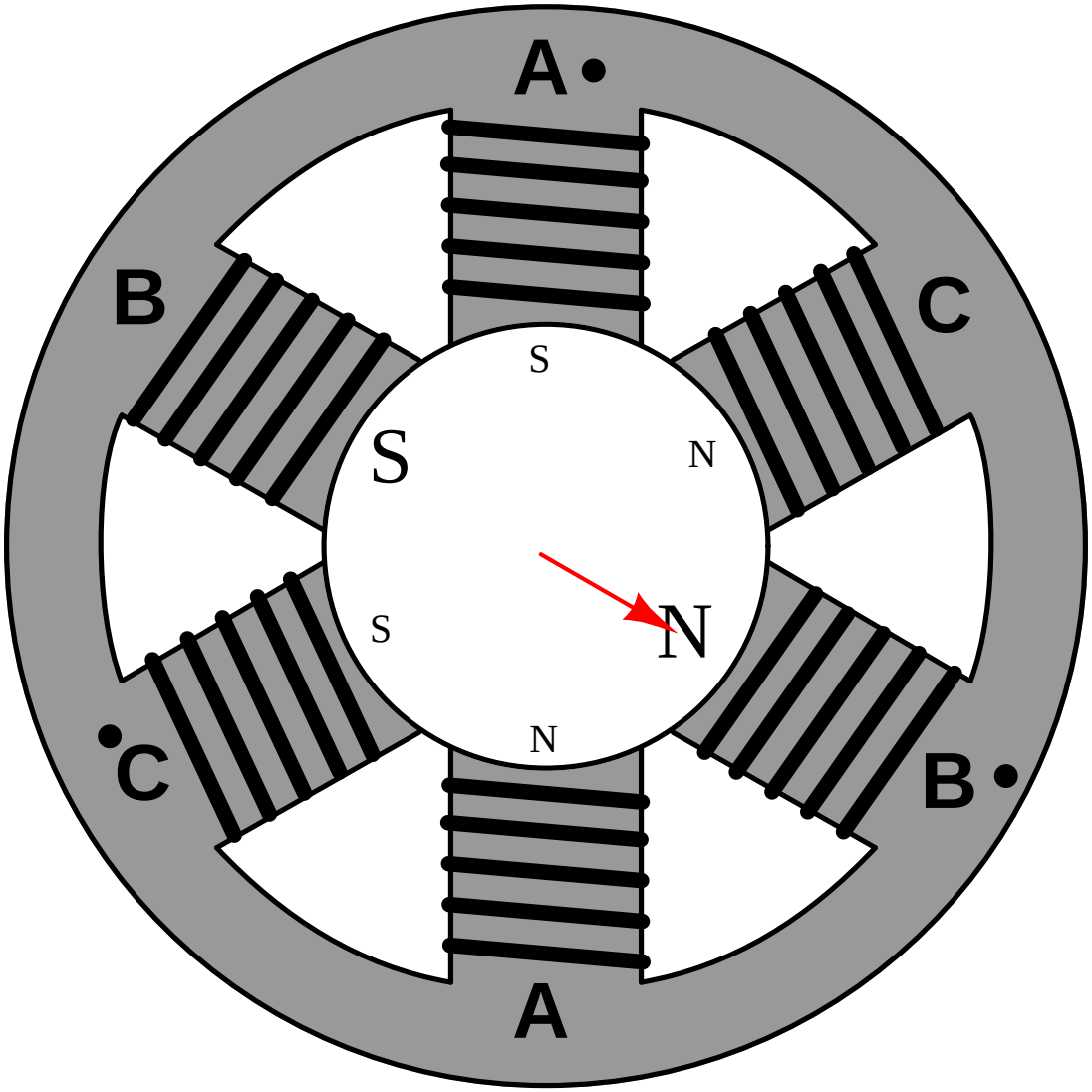

A
B
C
N
N
S
S
C
B
A

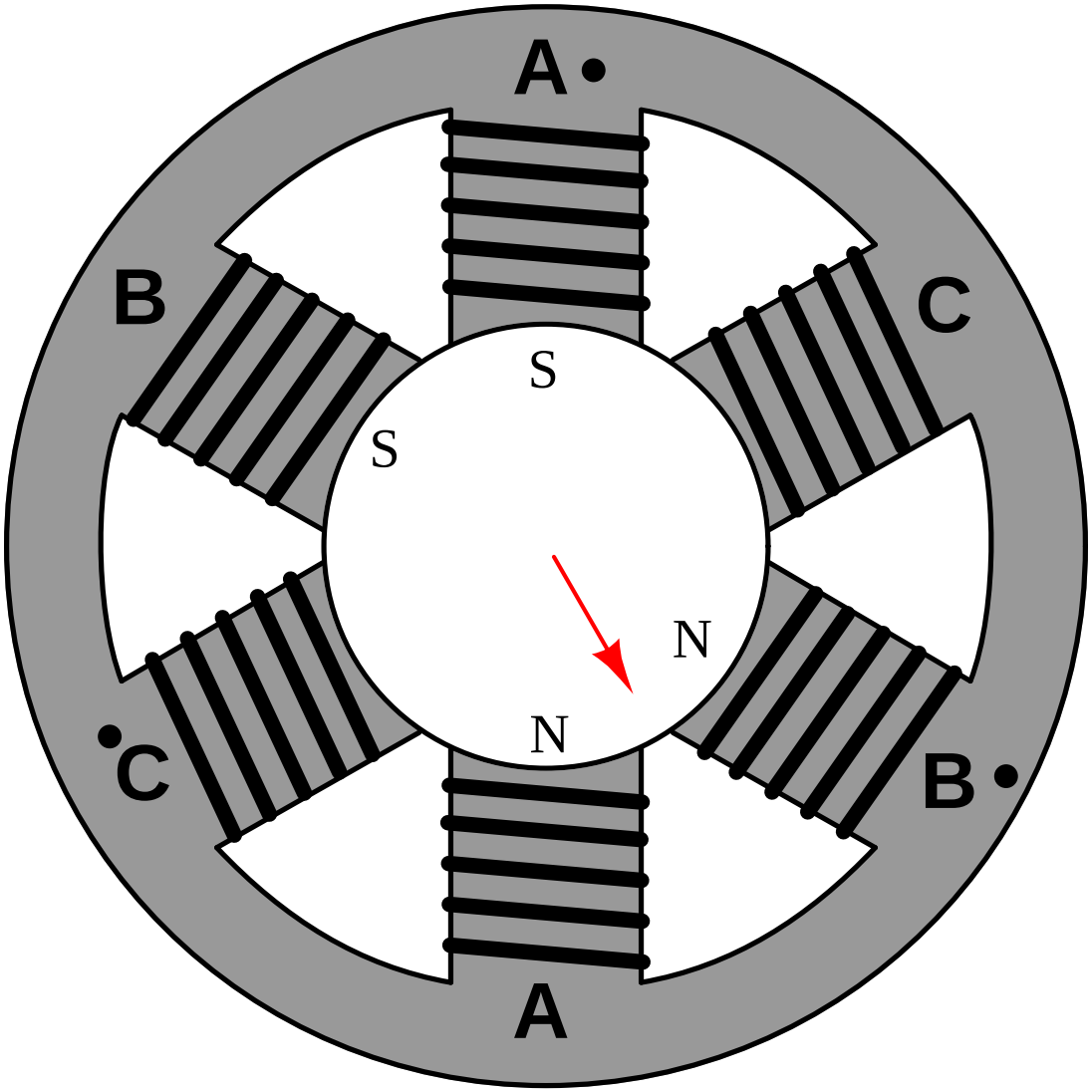

A
B
C
N
N
N
S
S
S
C
B
A

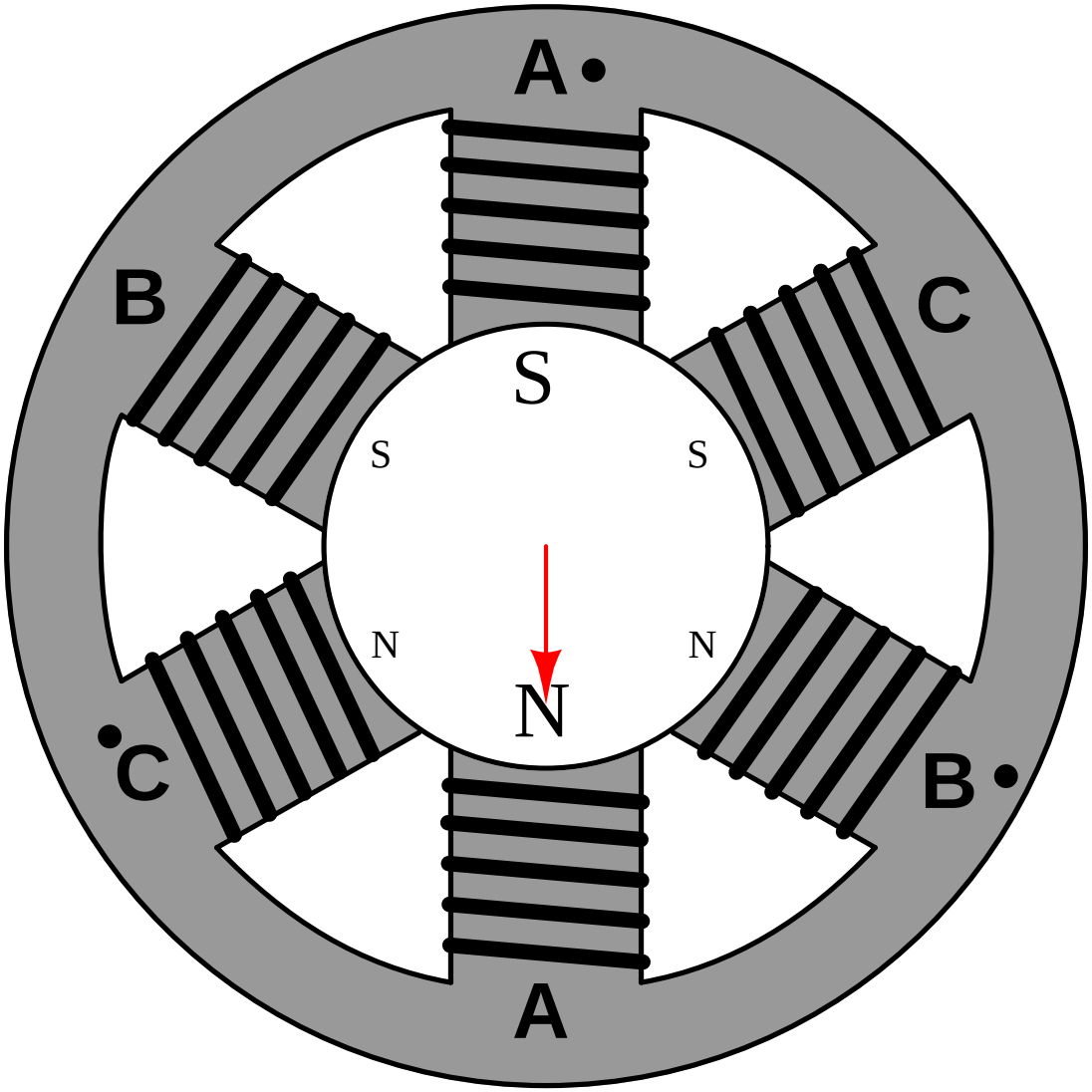

A
B
C
N
N
S
S
C
B
A

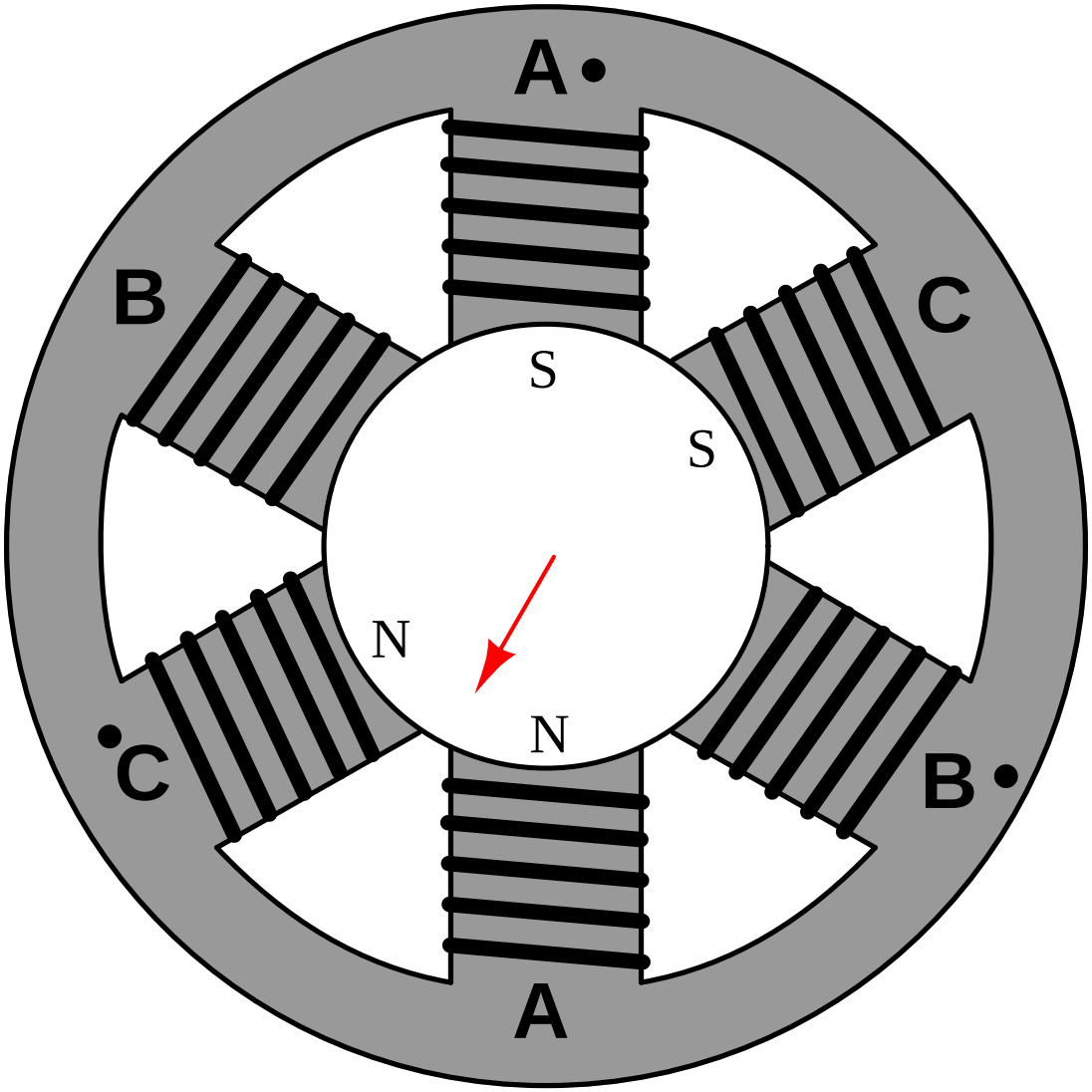

A
B
C
N
S
N
S
N
S
C
B
A

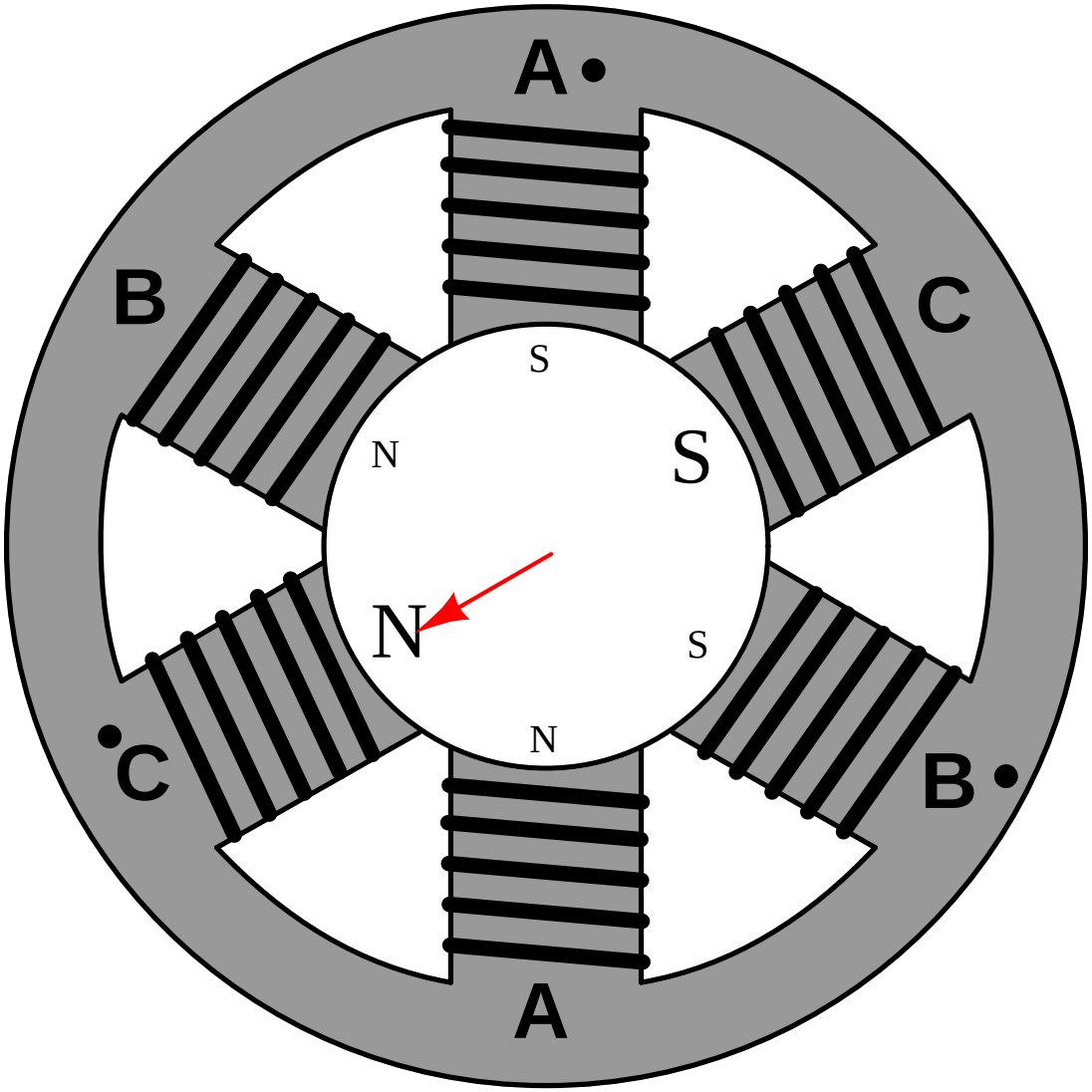

A
B
C
S
N
S
N
C
B
A


A
B
C
S
N
S
N
S
N
C
B
A


A
B
C
S
S
N
N
C
B
A


A
B
C
S
S
S
N
N
N
C
B
A


A
B
C
S
S
N
N
C
B
A


A
B
C
S
N
S
N
S
N
C
B
A

Chapter 7

Programming References

A powerful tool for mathematical modeling is text-based *computer programming*. This is where you type coded commands in text form which the computer is able to interpret. Many different text-based languages exist for this purpose, but we will focus here on just two of them, *C++* and *Python*.

7.1 Programming in C++

One of the more popular text-based computer programming languages is called *C++*. This is a *compiled* language, which means you must create a plain-text file containing C++ code using a program called a *text editor*, then execute a software application called a *compiler* to translate your "source code" into instructions directly understandable to the computer. Here is an example of "source code" for a very simple C++ program intended to perform some basic arithmetic operations and print the results to the computer's console:

```
#include <iostream>
using namespace std;

int main (void)
{
  float x, y;

  x = 200;
  y = -560.5;

  cout << "This simple program performs basic arithmetic on" << endl;
  cout << "the two numbers " << x << " and " << y << " and then" << endl;
  cout << "displays the results on the computer's console." << endl;

  cout << endl;

  cout << "Sum = " << x + y << endl;
  cout << "Difference = " << x - y << endl;
  cout << "Product = " << x * y << endl;
  cout << "Quotient of " << x / y << endl;

  return 0;
}
```

Computer languages such as C++ are designed to make sense when read by human programmers. The general order of execution is left-to-right, top-to-bottom just the same as reading any text document written in English. Blank lines, indentation, and other "whitespace" is largely irrelevant in C++ code, and is included only to make the code more pleasing[1] to view.

[1] Although not included in this example, *comments* preceded by double-forward slash characters (`//`) may be added to source code as well to provide explanations of what the code is supposed to do, for the benefit of anyone reading it. The compiler application will ignore all comments.

Let's examine the C++ source code to explain what it means:

- `#include <iostream>` and `using namespace std;` are set-up instructions to the compiler giving it some context in which to interpret your code. The code specific to your task is located between the brace symbols ({ and }, often referred to as "curly-braces").

- `int main (void)` labels the "Main" function for the computer: the instructions within this function (lying between the { and } symbols) it will be commanded to execute. Every complete C++ program contains a `main` function at minimum, and often additional functions as well, but the `main` function is where execution always begins. The `int` declares this function will return an *integer* number value when complete, which helps to explain the purpose of the `return 0;` statement at the end of the `main` function: providing a numerical value of zero at the program's completion as promised by `int`. This returned value is rather incidental to our purpose here, but it is fairly standard practice in C++ programming.

- Grouping symbols such as (parentheses) and {braces} abound in C, C++, and other languages (e.g. Java). Parentheses typically group data to be processed by a function, called *arguments* to that function. Braces surround lines of executable code belonging to a particular function.

- The `float` declaration reserves places in the computer's memory for two *floating-point* variables, in this case the variables' names being `x` and `y`. In most text-based programming languages, variables may be named by single letters or by combinations of letters (e.g. `xyz` would be a single variable).

- The next two lines assign numerical values to the two variables. Note how each line terminates with a semicolon character (`;`) and how this pattern holds true for most of the lines in this program. In C++ semicolons are analogous to periods at the ends of English sentences. This demarcation of each line's end is necessary because C++ ignores whitespace on the page and doesn't "know" otherwise where one line ends and another begins.

- All the other instructions take the form of a `cout` command which prints characters to the "standard output" stream of the computer, which in this case will be text displayed on the console. The double-less-than symbols (`<<`) show data being sent *toward* the `cout` command. Note how verbatim text is enclosed in quotation marks, while variables such as `x` or mathematical expressions such as `x - y` are not enclosed in quotations because we want the computer to display the numerical values represented, not the literal text.

- Standard arithmetic operations (add, subtract, multiply, divide) are represented as `+`, `-`, `*`, and `/`, respectively.

- The `endl` found at the end of every `cout` statement marks the end of a line of text printed to the computer's console display. If not for these `endl` inclusions, the displayed text would resemble a run-on sentence rather than a paragraph. Note the `cout << endl;` line, which does nothing but create a blank line on the screen, for no reason other than esthetics.

After saving this *source code* text to a file with its own name (e.g. `myprogram.cpp`), you would then *compile* the source code into an *executable* file which the computer may then run. If you are using a console-based compiler such as *GCC* (very popular within variants of the Unix operating system[2], such as Linux and Apple's OS X), you would type the following command and press the Enter key:

```
g++ -o myprogram.exe myprogram.cpp
```

This command instructs the `GCC` compiler to take your source code (`myprogram.cpp`) and create with it an executable file named `myprogram.exe`. Simply typing `./myprogram.exe` at the command-line will then execute your program:

```
./myprogram.exe
```

If you are using a graphic-based C++ development system such as Microsoft Visual Studio[3], you may simply create a new console application "project" using this software, then paste or type your code into the example template appearing in the editor window, and finally run your application to test its output.

As this program runs, it displays the following text to the console:

```
This simple program performs basic arithmetic on
the two numbers 200 and -560.5 and then
displays the results on the computer's console.

Sum = -360.5
Difference = 760.5
Product = -112100
Quotient of -0.356824
```

As crude as this example program is, it serves the purpose of showing how easy it is to write and execute simple programs in a computer using the C++ language. As you encounter C++ example programs (shown as source code) in any of these modules, feel free to directly copy-and-paste the source code text into a text editor's screen, then follow the rest of the instructions given here (i.e. save to a file, compile, and finally run your program). You will find that it is generally easier to

[2] A very functional option for users of Microsoft Windows is called *Cygwin*, which provides a Unix-like console environment complete with all the customary utility applications such as GCC!

[3] Using Microsoft Visual Studio community version 2017 at the time of this writing to test this example, here are the steps I needed to follow in order to successfully compile and run a simple program such as this: (1) Start up Visual Studio and select the option to create a New Project; (2) Select the Windows Console Application template, as this will perform necessary set-up steps to generate a console-based program which will save you time and effort as well as avoid simple errors of omission; (3) When the editing screen appears, type or paste the C++ code within the `main()` function provided in the template, deleting the "Hello World" `cout` line that came with the template; (4) Type or paste any preprocessor directives (e.g. `#include` statements, `namespace` statements) necessary for your code that did not come with the template; (5) Lastly, under the Debug drop-down menu choose either Start Debugging (F5 hot-key) or Start Without Debugging (Ctrl-F5 hotkeys) to compile ("Build") and run your new program. Upon execution a console window will appear showing the output of your program.

learn computer programming by closely examining others' example programs and modifying them than it is to write your own programs starting from a blank screen.

7.2 Programming in Python

Another text-based computer programming language called *Python* allows you to type instructions at a terminal prompt and receive immediate results without having to compile that code. This is because Python is an *interpreted* language: a software application called an *interpreter* reads your source code, translates it into computer-understandable instructions, and then executes those instructions in one step.

The following shows what happens on my personal computer when I start up the Python interpreter on my personal computer, by typing `python3`[4] and pressing the Enter key:

```
Python 3.7.2 (default, Feb 19 2019, 18:15:18)
[GCC 4.1.2] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

The `>>>` symbols represent the prompt within the Python interpreter "shell", signifying readiness to accept Python commands entered by the user.

Shown here is an example of the same arithmetic operations performed on the same quantities, using a Python interpreter. All lines shown preceded by the `>>>` prompt are entries typed by the human programmer, and all lines shown without the `>>>` prompt are responses from the Python interpreter software:

```
>>> x = 200
>>> y = -560.5
>>> x + y
-360.5
>>> x - y
760.5
>>> x * y
-112100.0
>>> x / y
-0.35682426404995538
>>> quit()
```

[4] Using version 3 of Python, which is the latest at the time of this writing.

More advanced mathematical functions are accessible in Python by first entering the line `from math import *` which "imports" these functions from Python's math *library* (with functions identical to those available for the C programming language, and included on any computer with Python installed). Some examples show some of these functions in use, demonstrating how the Python interpreter may be used as a scientific calculator:

```
>>> from math import *
>>> sin(30.0)
-0.98803162409286183
>>> sin(radians(30.0))
0.49999999999999994
>>> pow(2.0, 5.0)
32.0
>>> log10(10000.0)
4.0
>>> e
2.7182818284590451
>>> pi
3.1415926535897931
>>> log(pow(e,6.0))
6.0
>>> asin(0.7071068)
0.78539819000368838
>>> degrees(asin(0.7071068))
45.000001524425265
>>> quit()
```

Note how trigonometric functions assume angles expressed in *radians* rather than *degrees*, and how Python provides convenient functions for translating between the two. Logarithms assume a base of e unless otherwise stated (e.g. the `log10` function for common logarithms).

The interpreted (versus compiled) nature of Python, as well as its relatively simple syntax, makes it a good choice as a person's first programming language. For complex applications, interpreted languages such as Python execute slower than compiled languages such as C++, but for the very simple examples used in these learning modules speed is not a concern.

Another Python math library is `cmath`, giving Python the ability to perform arithmetic on complex numbers. This is very useful for AC circuit analysis using *phasors*[5] as shown in the following example. Here we see Python's interpreter used as a scientific calculator to show series and parallel impedances of a resistor, capacitor, and inductor in a 60 Hz AC circuit:

```
>>> from math import *
>>> from cmath import *
>>> r = complex(400,0)
>>> f = 60.0
>>> xc = 1/(2 * pi * f * 4.7e-6)
>>> zc = complex(0,-xc)
>>> xl = 2 * pi * f * 1.0
>>> zl = complex(0,xl)
>>> r + zc + zl
(400-187.38811239154882j)
>>> 1/(1/r + 1/zc + 1/zl)
(355.837695813625+125.35793777619385j)
>>> polar(r + zc + zl)
(441.717448903332, -0.4381072059213295)
>>> abs(r + zc + zl)
441.717448903332
>>> phase(r + zc + zl)
-0.4381072059213295
>>> degrees(phase(r + zc + zl))
-25.10169387356105
```

When entering a value in rectangular form, we use the `complex()` function where the arguments are the real and imaginary quantities, respectively. If we had opted to enter the impedance values in polar form, we would have used the `rect()` function where the first argument is the magnitude and the second argument is the angle in radians. For example, we could have set the capacitor's impedance (`zc`) as $X_C \angle -90^o$ with the command `zc = rect(xc,radians(-90))` rather than with the command `zc = complex(0,-xc)` and it would have worked the same.

Note how Python defaults to rectangular form for complex quantities. Here we defined a 400 Ohm resistance as a complex value in rectangular form (400 +j0 Ω), then computed capacitive and inductive reactances at 60 Hz and defined each of those as complex (phasor) values ($0 - jX_c$ Ω and $0 + jX_l$ Ω, respectively). After that we computed total impedance in series, then total impedance in parallel. Polar-form representation was then shown for the series impedance (441.717 Ω $\angle -25.102^o$). Note the use of different functions to show the polar-form series impedance value: `polar()` takes the complex quantity and returns its polar magnitude and phase angle in *radians*; `abs()` returns just the polar magnitude; `phase()` returns just the polar angle, once again in radians. To find the polar phase angle in degrees, we nest the `degrees()` and `phase()` functions together.

The utility of Python's interpreter environment as a scientific calculator should be clear from these examples. Not only does it offer a powerful array of mathematical functions, but also unlimited

[5] A "phasor" is a voltage, current, or impedance represented as a complex number, either in rectangular or polar form.

assignment of variables as well as a convenient text record[6] of all calculations performed which may be easily copied and pasted into a text document for archival.

It is also possible to save a set of Python commands to a text file using a text editor application, and then instruct the Python interpreter to execute it at once rather than having to type it line-by-line in the interpreter's shell. For example, consider the following Python program, saved under the filename `myprogram.py`:

```
x = 200
y = -560.5

print("Sum")
print(x + y)

print("Difference")
print(x - y)

print("Product")
print(x * y)

print("Quotient")
print(x / y)
```

As with C++, the interpreter will read this source code from left-to-right, top-to-bottom, just the same as you or I would read a document written in English. Interestingly, whitespace *is* significant in the Python language (unlike C++), but this simple example program makes no use of that.

To execute this Python program, I would need to type `python myprogram.py` and then press the Enter key at my computer console's prompt, at which point it would display the following result:

```
Sum
-360.5
Difference
760.5
Product
-112100.0
Quotient
-0.35682426405
```

As you can see, syntax within the Python programming language is simpler than C++, which is one reason why it is often a preferred language for beginning programmers.

[6] Like many command-line computing environments, Python's interpreter supports "up-arrow" recall of previous entries. This allows quick recall of previously typed commands for editing and re-evaluation.

If you are interested in learning more about computer programming in *any* language, you will find a wide variety of books and free tutorials available on those subjects. Otherwise, feel free to learn by the examples presented in these modules.

7.3 Modeling Wye and Delta networks using C++

The following program written in the C++ programming language calculates parameters in a three-phase circuit, where the source may be either Wye or Delta, and the load may also be (independently) Wye or Delta:

```
#include <iostream>
#include <cmath>
using namespace std;

int main (void)
{
  bool source_wye, load_wye;
  float Vsrc_phase, V_line, Vld_phase, Isrc_phase, I_line, Ild_phase, Rld_phase;

  cout << "Enter source phase voltage: ";
  cin >> Vsrc_phase;
  cout << "Choose source type, Wye (1) or Delta (0): ";
  cin >> source_wye;

  cout << "Enter load phase resistance: ";
  cin >> Rld_phase;
  cout << "Choose load type, Wye (1) or Delta (0): ";
  cin >> load_wye;

  if (source_wye == true)
    V_line = sqrt(3) * Vsrc_phase;
  else
    V_line = Vsrc_phase;

  if (load_wye == true)
    Vld_phase = V_line / sqrt(3);
  else
    Vld_phase = V_line;

  Ild_phase = Vld_phase / Rld_phase;

  if (load_wye == true)
    I_line = Ild_phase;
  else
    I_line = sqrt(3) * Ild_phase;

  if (source_wye == true)
    Isrc_phase = I_line;
```

```
  else
    Isrc_phase = I_line / sqrt(3);

  cout << "Source phase voltage = " << Vsrc_phase << " Volts" << endl;
  cout << "Source phase current = " << Isrc_phase << " Amperes" << endl;
  cout << "Line voltage = " << V_line << " Volts" << endl;
  cout << "Line current = " << I_line << " Amperes" << endl;
  cout << "Load phase voltage = " << Vld_phase << " Volts" << endl;
  cout << "Load phase current = " << Ild_phase << " Amperes" << endl;
  cout << "Total power = " << V_line * I_line * sqrt(3) << " Watts" << endl;

  return 0;
}
```

Let's analyze how this program works, exploring the following programming principles along the way:

- Order of execution
- Preprocessor directives, namespaces
- The `main` function: return values, arguments
- Delimiter characters (e.g. { } ;)
- Variable types (`float`, `bool`), names, and declarations
- Printing text output (`cout`, `<<`, `endl`)
- Accepting user input (`cin`, `>>`)
- Conditional statements (`if`, `else`)
- Comparison (`==`)
- Basic arithmetic (`*`, `/`)
- Arithmetic functions (`sqrt`)
- Boolean states (`true`, `false`)

The first thing we should recognize is that the sequence of execution for this code is in the same order you would read an English document: left to right and top to bottom. This is very common in text-based computer programming languages.

The `#include` and `namespace` directives instruct the compiler how to interpret later instructions such as `cout` and `sqrt`. The `main` function encapsulates all the code we wish to run (enclosed

between a matching pair of "curly-brace" symbols: { }), and presents it to the compiler software as the first point of execution for the compiled program.

The first two lines within the `main` function declare all the variables this program will need for its mathematical calculations. In C and C++ all variables used in a program must be explicitly declared by type and by name before they are used in any way. This ensures the computer knows how much memory to reserve for each variable, and how to access each one. Here we have two Boolean-type variables capable only of being true or false, 1 or 0; also we have seven floating-point variables suitable for representing "real" numbers.

Note how variable names in C/C++ may be any combination of letters, and are case-sensitive. Spaces are not allowed in variable names because these are interpreted by the compiler software to be delimiting symbols, separating different objects. This is why underscore characters (_) are used in some of these variable names, to put some visual separation between parts of the variable name while maintaining an unbroken word for that name.

All `cout` lines print text to the computer's console for viewing by the user. All `cin` lines accept user-typed input. Like the variable declaration lines, each of these is terminated by a semicolon character (`;`) to signify the end of each complete line of code. This is necessary, like the period symbol at the end of a sentence, to tell the C++ compiler software where each line of code stops and the next begins, because in C and C++ whitespace in the source code is ignored.

Much of this program consists of `if` and `else` statements, the purpose of these being to check condition of the Boolean variables (true or false) and use these conditions to calculate phase quantities from line quantities (or vice-versa). Immediately following each `if` and `else` statement is a single line of code containing one or more arithmetic instructions, which is what will be executed if the condition is true. If the condition stated within the parentheses of an `if` statement is true, then the line following that `if` statement is executed and the `else` line (and its following statement) are ignored. If the condition for an `if` statement is false, the computer skips ahead to the line following the `else` statement and executes that (only).

Note the use of *double-equals* symbols (`==`) in the `if` statement conditional lines. In C and C++, the double-equals symbol means *check for equality* and returns a result that is either true or false. Compare this with the use of the *single-equals* symbols in all the calculation lines. In C and C++, a single equals symbol means the variable on the left will be *set equal to* the quantity on the right.

Finally, at the end of the program we have a set of `cout` lines instructing the computer to print text labels and variable values to the console for viewing. Anything enclosed in parentheses is printed verbatim, while anything in a `cout` line not enclosed in parentheses will have its value printed to the console rather than the literal letters of its name. Note also how `endl` is a special control word, instructing the computer to terminate that line of printed text and begin a new line.

Here is an example screenshot of this program running:

```
Enter source phase voltage: 120
Choose source type, Wye (1) or Delta (0): 1
Enter load phase resistance: 100
Choose load type, Wye (1) or Delta (0): 0
Source phase voltage = 120 Volts
Source phase current = 3.6 Amperes
Line voltage = 207.846 Volts
Line current = 3.6 Amperes
Load phase voltage = 207.846 Volts
Load phase current = 2.07846 Amperes
Total power = 1296 Watts
```

An interesting simplification of the conditional (`if`) code is to eliminate the equality check, since the variables whose states we are checking are already Boolean in nature and therefore can only be either true or false (1 or 0). Conditional statements such as `if` evaluate the check function within their parentheses and then act on the result of that check either being true or false. However, since the variable's we're checking are *already* either true or false, we may simplify the `if` statements in the following manner, as shown in this portion of the source code listing:

```
if (source_wye)
  V_line = sqrt(3) * Vsrc_phase;
else
  V_line = Vsrc_phase;

if (load_wye)
  Vld_phase = V_line / sqrt(3);
else
  Vld_phase = V_line;

Ild_phase = Vld_phase / Rld_phase;

if (load_wye)
  I_line = Ild_phase;
else
  I_line = sqrt(3) * Ild_phase;

if (source_wye)
  Isrc_phase = I_line;
else
  Isrc_phase = I_line / sqrt(3);
```

For example, `if (source_wye)` is logically equivalent to `if (source_wye == true)`.

Chapter 8

Questions

This learning module, along with all others in the ModEL collection, is designed to be used in an inverted instructional environment where students independently read[1] the tutorials and attempt to answer questions on their own *prior* to the instructor's interaction with them. In place of lecture[2], the instructor engages with students in Socratic-style dialogue, probing and challenging their understanding of the subject matter through inquiry.

Answers are not provided for questions within this chapter, and this is by design. Solved problems may be found in the Tutorial and Derivation chapters, instead. The goal here is *independence*, and this requires students to be challenged in ways where others cannot think for them. Remember that you always have the tools of *experimentation* and *computer simulation* (e.g. SPICE) to explore concepts!

The following lists contain ideas for Socratic-style questions and challenges. Upon inspection, one will notice a strong theme of *metacognition* within these statements: they are designed to foster a regular habit of examining one's own thoughts as a means toward clearer thinking. As such these sample questions are useful both for instructor-led discussions as well as for self-study.

[1] Technical reading is an essential academic skill for any technical practitioner to possess for the simple reason that the most comprehensive, accurate, and useful information to be found for developing technical competence is in textual form. Technical careers in general are characterized by the need for continuous learning to remain current with standards and technology, and therefore any technical practitioner who cannot read well is handicapped in their professional development. An excellent resource for educators on improving students' reading prowess through intentional effort and strategy is the book textitReading For Understanding – How Reading Apprenticeship Improves Disciplinary Learning in Secondary and College Classrooms by Ruth Schoenbach, Cynthia Greenleaf, and Lynn Murphy.

[2] Lecture is popular as a teaching method because it is easy to implement: any reasonably articulate subject matter expert can talk to students, even with little preparation. However, it is also quite problematic. A good lecture always makes complicated concepts seem easier than they are, which is bad for students because it instills a false sense of confidence in their own understanding; reading and re-articulation requires more cognitive effort and serves to verify comprehension. A culture of teaching-by-lecture fosters a debilitating dependence upon direct personal instruction, whereas the challenges of modern life demand independent and critical thought made possible only by gathering information and perspectives from afar. Information presented in a lecture is ephemeral, easily lost to failures of memory and dictation; text is forever, and may be referenced at any time.

General challenges following tutorial reading

- Summarize as much of the text as you can in one paragraph of your own words. A helpful strategy is to explain ideas as you would for an intelligent child: as simple as you can without compromising too much accuracy.
- Simplify a particular section of the text, for example a paragraph or even a single sentence, so as to capture the same fundamental idea in fewer words.
- Where did the text make the most sense to you? What was it about the text's presentation that made it clear?
- Identify where it might be easy for someone to misunderstand the text, and explain why you think it could be confusing.
- Identify any new concept(s) presented in the text, and explain in your own words.
- Identify any familiar concept(s) such as physical laws or principles applied or referenced in the text.
- Devise a proof of concept experiment demonstrating an important principle, physical law, or technical innovation represented in the text.
- Devise an experiment to disprove a plausible misconception.
- Did the text reveal any misconceptions you might have harbored? If so, describe the misconception(s) and the reason(s) why you now know them to be incorrect.
- Describe any useful problem-solving strategies applied in the text.
- Devise a question of your own to challenge a reader's comprehension of the text.

General follow-up challenges for assigned problems

- Identify where any fundamental laws or principles apply to the solution of this problem, especially before applying any mathematical techniques.
- Devise a thought experiment to explore the characteristics of the problem scenario, applying known laws and principles to mentally model its behavior.
- Describe in detail your own strategy for solving this problem. How did you identify and organized the given information? Did you sketch any diagrams to help frame the problem?
- Is there more than one way to solve this problem? Which method seems best to you?
- Show the work you did in solving this problem, even if the solution is incomplete or incorrect.
- What would you say was the most challenging part of this problem, and why was it so?
- Was any important information missing from the problem which you had to research or recall?
- Was there any extraneous information presented within this problem? If so, what was it and why did it not matter?
- Examine someone else's solution to identify where they applied fundamental laws or principles.
- Simplify the problem from its given form and show how to solve this simpler version of it. Examples include eliminating certain variables or conditions, altering values to simpler (usually whole) numbers, applying a limiting case (i.e. altering a variable to some extreme or ultimate value).
- For quantitative problems, identify the real-world meaning of all intermediate calculations: their units of measurement, where they fit into the scenario at hand. Annotate any diagrams or illustrations with these calculated values.
- For quantitative problems, try approaching it qualitatively instead, thinking in terms of "increase" and "decrease" rather than definite values.
- For qualitative problems, try approaching it quantitatively instead, proposing simple numerical values for the variables.
- Were there any assumptions you made while solving this problem? Would your solution change if one of those assumptions were altered?
- Identify where it would be easy for someone to go astray in attempting to solve this problem.
- Formulate your own problem based on what you learned solving this one.

General follow-up challenges for experiments or projects

- In what way(s) was this experiment or project easy to complete?
- Identify some of the challenges you faced in completing this experiment or project.

- Show how thorough documentation assisted in the completion of this experiment or project.
- Which fundamental laws or principles are key to this system's function?
- Identify any way(s) in which one might obtain false or otherwise misleading measurements from test equipment in this system.
- What will happen if (component X) fails (open/shorted/etc.)?
- What would have to occur to make this system unsafe?

8.1 Conceptual reasoning

These questions are designed to stimulate your analytic and synthetic thinking[3]. In a Socratic discussion with your instructor, the goal is for these questions to prompt an extended dialogue where assumptions are revealed, conclusions are tested, and understanding is sharpened. Your instructor may also pose additional questions based on those assigned, in order to further probe and refine your conceptual understanding.

Questions that follow are presented to challenge and probe your understanding of various concepts presented in the tutorial. These questions are intended to serve as a guide for the Socratic dialogue between yourself and the instructor. Your instructor's task is to ensure you have a sound grasp of these concepts, and the questions contained in this document are merely a means to this end. Your instructor may, at his or her discretion, alter or substitute questions for the benefit of tailoring the discussion to each student's needs. The only absolute requirement is that each student is challenged and assessed at a level equal to or greater than that represented by the documented questions.

It is far more important that you convey your *reasoning* than it is to simply convey a correct answer. For this reason, you should refrain from researching other information sources to answer questions. What matters here is that *you* are doing the thinking. If the answer is incorrect, your instructor will work with you to correct it through proper reasoning. A correct answer without an adequate explanation of how you derived that answer is unacceptable, as it does not aid the learning or assessment process.

You will note a conspicuous lack of answers given for these conceptual questions. Unlike standard textbooks where answers to every other question are given somewhere toward the back of the book, here in these learning modules students must rely on other means to check their work. The best way by far is to debate the answers with fellow students and also with the instructor during the Socratic dialogue sessions intended to be used with these learning modules. Reasoning through challenging questions with other people is an excellent tool for developing strong reasoning skills.

Another means of checking your conceptual answers, where applicable, is to use circuit simulation software to explore the effects of changes made to circuits. For example, if one of these conceptual questions challenges you to predict the effects of altering some component parameter in a circuit, you may check the validity of your work by simulating that same parameter change within software and seeing if the results agree.

[3] *Analytical* thinking involves the "disassembly" of an idea into its constituent parts, analogous to dissection. *Synthetic* thinking involves the "assembly" of a new idea comprised of multiple concepts, analogous to construction. Both activities are high-level cognitive skills, extremely important for effective problem-solving, necessitating frequent challenge and regular practice to fully develop.

8.1.1 Reading outline and reflections

"Reading maketh a full man; conference a ready man; and writing an exact man" – Francis Bacon

Francis Bacon's advice is a blueprint for effective education: reading provides the learner with knowledge, writing focuses the learner's thoughts, and critical dialogue equips the learner to confidently communicate and apply their learning. Independent acquisition and application of knowledge is a powerful skill, well worth the effort to cultivate. To this end, students should read these educational resources closely, journal their own reflections on the reading, and discuss in detail their findings with classmates and instructor(s). You should be able to do all of the following after reading any instructional text:

☑ Briefly SUMMARIZE THE TEXT in the form of a journal entry documenting your learning as you progress through the course of study. Share this summary in dialogue with your classmates and instructor. Journaling is an excellent self-test of thorough reading because you cannot clearly express what you have not read or did not comprehend.

☑ Demonstrate ACTIVE READING STRATEGIES, including verbalizing your impressions as you read, simplifying long passages to convey the same ideas using fewer words, annotating text and illustrations with your own interpretations, working through mathematical examples shown in the text, cross-referencing passages with relevant illustrations and/or other passages, identifying problem-solving strategies applied by the author, etc. Technical reading is a special case of problem-solving, and so these strategies work precisely because they help solve any problem: paying attention to your own thoughts (metacognition), eliminating unnecessary complexities, identifying what makes sense, paying close attention to details, drawing connections between separated facts, and noting the successful strategies of others.

☑ Identify IMPORTANT THEMES, especially GENERAL LAWS and PRINCIPLES, expounded in the text and express them in the simplest of terms as though you were teaching an intelligent child. This emphasizes connections between related topics and develops your ability to communicate complex ideas to anyone.

☑ Form YOUR OWN QUESTIONS based on the reading, and then pose them to your instructor and classmates for their consideration. Anticipate both correct and incorrect answers, the incorrect answer(s) assuming one or more plausible misconceptions. This helps you view the subject from different perspectives to grasp it more fully.

☑ Devise EXPERIMENTS to test claims presented in the reading, or to disprove misconceptions. Predict possible outcomes of these experiments, and evaluate their meanings: what result(s) would confirm, and what would constitute disproof? Running mental simulations and evaluating results is essential to scientific and diagnostic reasoning.

☑ Specifically identify any points you found CONFUSING. The reason for doing this is to help diagnose misconceptions and overcome barriers to learning.

8.1.2 Foundational concepts

Correct analysis and diagnosis of electric circuits begins with a proper understanding of some basic concepts. The following is a list of some important concepts referenced in this module's full tutorial. Define each of them in your own words, and be prepared to illustrate each of these concepts with a description of a practical example and/or a live demonstration.

Kirchhoff's Voltage Law

Kirchhoff's Current Law

Grounding

Polyphase

Alternator

Phasor diagram

Phase sequence

Wye connection

Delta connection

Phase versus line

Wye voltages and currents

Delta voltages and currents

Advantages of polyphase electrical systems

Power in a three-phase system

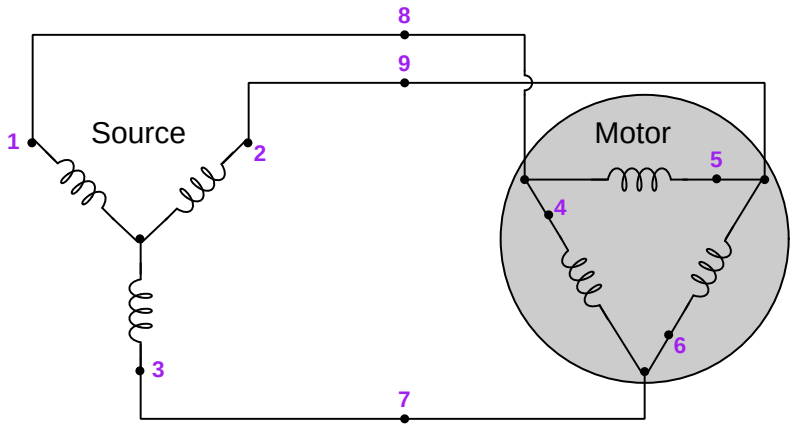
Wye grounding

Delta grounding

Sinusoidal decomposition (i.e. Fourier's Theorem)

8.1.3 Similar voltages and currents

Identify points within this circuit *absolutely guaranteed* to share the same current, whether or not the source or load happen to be balanced:



Next, identify point-pairs within this circuit absolutely guaranteed to share the same voltage between them, whether or not the source or load happen to be balanced.

In each case, identify the fundamental electrical principle that guarantees these equalities.

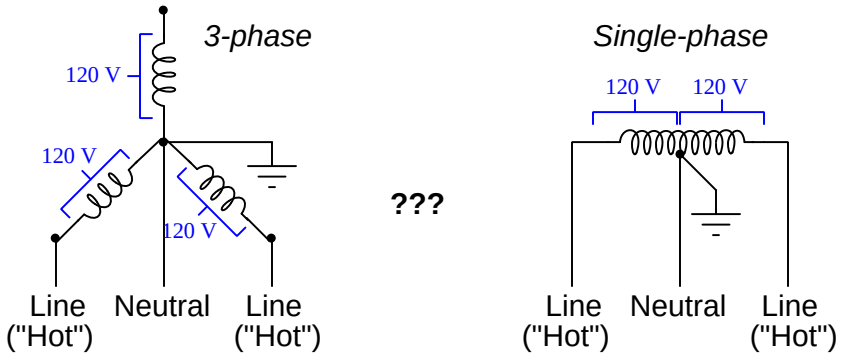
Challenges

- What conditions would have to be met in order for $I_4 = I_5 = I_6$?
- What conditions would have to be met in order for $V_{1,2} = V_{2,3} = V_{3,1}$?

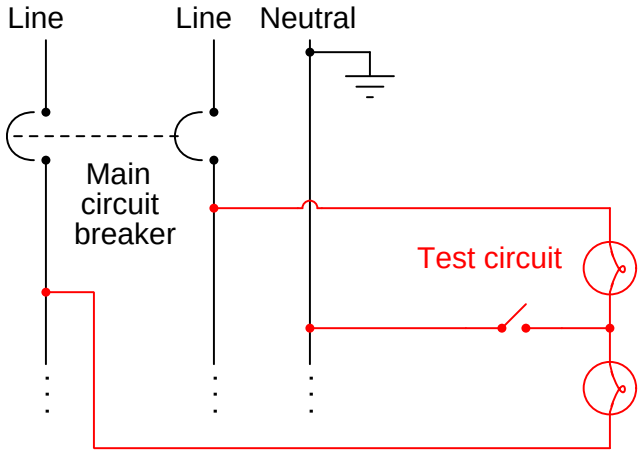
8.1.4 Testing for single-phase or three-phase

Suppose you are preparing to repair some power wiring inside of a home, and are wondering whether or not the home is supplied with 3-phase or single-phase power. You do not have a voltmeter available to measure voltage, but you do have plenty of electric lamps, switches, wires, and other standard residential wiring components available for use.

The two possibilities for this home's power source are shown here, the coils representing secondary windings of the utility power transformer:



An experienced electrician suggests you build the following circuit to test whether or not the home's power is supplied by a 3-phase source or a single-phase source:



The electrician tells you to open and close the switch, and observe the brightness of the lamps. This will indicate whether or not the system is 3-phase.

Explain how this circuit works. What sort of lamp behavior would indicate a 3-phase source? What sort of lamp behavior would indicate a single-phase source?

Is this test circuit capable of indicating the *phase rotation* of the source, assuming it is three-phase?

Challenges

- Assuming you did possess an AC voltmeter at this work site, describe how you could use that meter to tell whether this was a polyphase or single-phase power system.
- If the utility transformer were pole-mounted and visible from ground level, what details might you look for to determine whether this was a polyphase or single-phase power system?
- Explain why the "Hot" conductors are called by that name.

8.2 Quantitative reasoning

These questions are designed to stimulate your computational thinking. In a Socratic discussion with your instructor, the goal is for these questions to reveal your mathematical approach(es) to problem-solving so that good technique and sound reasoning may be reinforced. Your instructor may also pose additional questions based on those assigned, in order to observe your problem-solving firsthand.

Mental arithmetic and estimations are strongly encouraged for all calculations, because without these abilities you will be unable to readily detect errors caused by calculator misuse (e.g. keystroke errors).

You will note a conspicuous lack of answers given for these quantitative questions. Unlike standard textbooks where answers to every other question are given somewhere toward the back of the book, here in these learning modules students must rely on other means to check their work. My advice is to use circuit simulation software such as SPICE to check the correctness of quantitative answers. Refer to those learning modules within this collection focusing on SPICE to see worked examples which you may use directly as practice problems for your own study, and/or as templates you may modify to run your own analyses and generate your own practice problems.

Completely worked example problems found in the Tutorial may also serve as "test cases[4]" for gaining proficiency in the use of circuit simulation software, and then once that proficiency is gained you will never need to rely[5] on an answer key!

[4] In other words, set up the circuit simulation software to analyze the same circuit examples found in the Tutorial. If the simulated results match the answers shown in the Tutorial, it confirms the simulation has properly run. If the simulated results disagree with the Tutorial's answers, something has been set up incorrectly in the simulation software. Using every Tutorial as practice in this way will quickly develop proficiency in the use of circuit simulation software.

[5] This approach is perfectly in keeping with the instructional philosophy of these learning modules: *teaching students to be self-sufficient thinkers*. Answer keys can be useful, but it is even more useful to your long-term success to have a set of tools on hand for checking your own work, because once you have left school and are on your own, there will no longer be "answer keys" available for the problems you will have to solve.

8.2.1 Miscellaneous physical constants

Note: constants shown in **bold** type are *exact*, not approximations. Values inside of parentheses show one standard deviation (σ) of uncertainty in the final digits: for example, the magnetic permeability of free space value given as $1.25663706212(19) \times 10^{-6}$ H/m represents a center value (i.e. the location parameter) of $1.25663706212 \times 10^{-6}$ Henrys per meter with one standard deviation of uncertainty equal to $0.0000000000019 \times 10^{-6}$ Henrys per meter.

Avogadro's number (N_A) = **6.02214076** $\times 10^{23}$ **per mole** (mol^{-1})

Boltzmann's constant (k) = **1.380649** $\times 10^{-23}$ **Joules per Kelvin** (J/K)

Electronic charge (e) = **1.602176634** $\times 10^{-19}$ **Coulomb** (C)

Faraday constant (F) = **96,485.33212...** $\times 10^{4}$ **Coulombs per mole** (C/mol)

Magnetic permeability of free space (μ_0) = $1.25663706212(19) \times 10^{-6}$ Henrys per meter (H/m)

Electric permittivity of free space (ϵ_0) = $8.8541878128(13) \times 10^{-12}$ Farads per meter (F/m)

Characteristic impedance of free space (Z_0) = 376.730313668(57) Ohms (Ω)

Gravitational constant (G) = $6.67430(15) \times 10^{-11}$ cubic meters per kilogram-seconds squared ($m^3/kg\text{-}s^2$)

Molar gas constant (R) = **8.314462618... Joules per mole-Kelvin** (J/mol-K) = 0.08205746(14) liters-atmospheres per mole-Kelvin

Planck constant (h) = **6.62607015** $\times 10^{-34}$ **joule-seconds** (J-s)

Stefan-Boltzmann constant (σ) = **5.670374419...** $\times 10^{-8}$ **Watts per square meter-Kelvin**^{4} ($W/m^2 \cdot K^4$)

Speed of light in a vacuum (c) = **299,792,458 meters per second** (m/s) = 186282.4 miles per second (mi/s)

Note: All constants taken from NIST data "Fundamental Physical Constants – Complete Listing", from `http://physics.nist.gov/constants`, National Institute of Standards and Technology (NIST), 2018 CODATA Adjustment.

8.2.2 Introduction to spreadsheets

A powerful computational tool you are encouraged to use in your work is a *spreadsheet*. Available on most personal computers (e.g. Microsoft Excel), *spreadsheet* software performs numerical calculations based on number values and formulae entered into cells of a grid. This grid is typically arranged as lettered columns and numbered rows, with each cell of the grid identified by its column/row coordinates (e.g. cell `B3`, cell `A8`). Each cell may contain a string of text, a number value, or a mathematical formula. The spreadsheet automatically updates the results of all mathematical formulae whenever the entered number values are changed. This means it is possible to set up a spreadsheet to perform a series of calculations on entered data, and those calculations will be re-done by the computer any time the data points are edited in any way.

For example, the following spreadsheet calculates average speed based on entered values of distance traveled and time elapsed:

	A	B	C	D
1	Distance traveled	46.9	Kilometers	
2	Time elapsed	1.18	Hours	
3	Average speed	= B1 / B2	km/h	
4				
5				

Text labels contained in cells `A1` through `A3` and cells `C1` through `C3` exist solely for readability and are not involved in any calculations. Cell `B1` contains a sample distance value while cell `B2` contains a sample time value. The formula for computing speed is contained in cell `B3`. Note how this formula begins with an "equals" symbol (`=`), references the values for distance and speed by lettered column and numbered row coordinates (`B1` and `B2`), and uses a forward slash symbol for division (`/`). The coordinates `B1` and `B2` function as *variables*[6] would in an algebraic formula.

When this spreadsheet is executed, the numerical value `39.74576` will appear in cell `B3` rather than the formula `= B1 / B2`, because 39.74576 is the computed speed value given 46.9 kilometers traveled over a period of 1.18 hours. If a different numerical value for distance is entered into cell `B1` or a different value for time is entered into cell `B2`, cell `B3`'s value will automatically update. All you need to do is set up the given values and any formulae into the spreadsheet, and the computer will do all the calculations for you.

Cell `B3` may be referenced by other formulae in the spreadsheet if desired, since it is a variable just like the given values contained in `B1` and `B2`. This means it is possible to set up an entire chain of calculations, one dependent on the result of another, in order to arrive at a final value. The arrangement of the given data and formulae need not follow any pattern on the grid, which means you may place them anywhere.

[6] Spreadsheets may also provide means to attach text labels to cells for use as variable names (Microsoft Excel simply calls these labels "names"), but for simple spreadsheets such as those shown here it's usually easier just to use the standard coordinate naming for each cell.

Common[7] arithmetic operations available for your use in a spreadsheet include the following:

- Addition (`+`)
- Subtraction (`-`)
- Multiplication (`*`)
- Division (`/`)
- Powers (`^`)
- Square roots (`sqrt()`)
- Logarithms (`ln()` , `log10()`)

Parentheses may be used to ensure[8] proper order of operations within a complex formula. Consider this example of a spreadsheet implementing the *quadratic formula*, used to solve for roots of a polynomial expression in the form of $ax^2 + bx + c$:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

	A	B
1	x_1	= (-B4 + sqrt((B4^2) - (4*B3*B5))) / (2*B3)
2	x_2	= (-B4 - sqrt((B4^2) - (4*B3*B5))) / (2*B3)
3	a =	9
4	b =	5
5	c =	-2

This example is configured to compute roots[9] of the polynomial $9x^2 + 5x - 2$ because the values of 9, 5, and -2 have been inserted into cells `B3`, `B4`, and `B5`, respectively. Once this spreadsheet has been built, though, it may be used to calculate the roots of *any* second-degree polynomial expression simply by entering the new a, b, and c coefficients into cells `B3` through `B5`. The numerical values appearing in cells `B1` and `B2` will be automatically updated by the computer immediately following any changes made to the coefficients.

[7] Modern spreadsheet software offers a bewildering array of mathematical functions you may use in your computations. I recommend you consult the documentation for your particular spreadsheet for information on operations other than those listed here.

[8] Spreadsheet programs, like text-based programming languages, are designed to follow standard order of operations by default. However, my personal preference is to use parentheses even where strictly unnecessary just to make it clear to any other person viewing the formula what the intended order of operations is.

[9] Reviewing some algebra here, a *root* is a value for x that yields an overall value of zero for the polynomial. For this polynomial ($9x^2 + 5x - 2$) the two roots happen to be $x = 0.269381$ and $x = -0.82494$, with these values displayed in cells `B1` and `B2`, respectively upon execution of the spreadsheet.

Alternatively, one could break up the long quadratic formula into smaller pieces like this:

$$y = \sqrt{b^2 - 4ac} \qquad z = 2a$$

$$x = \frac{-b \pm y}{z}$$

	A	B	C
1	x_1	= (-B4 + C1) / C2	= sqrt((B4^2) - (4*B3*B5))
2	x_2	= (-B4 - C1) / C2	= 2*B3
3	a =	9	
4	b =	5	
5	c =	-2	

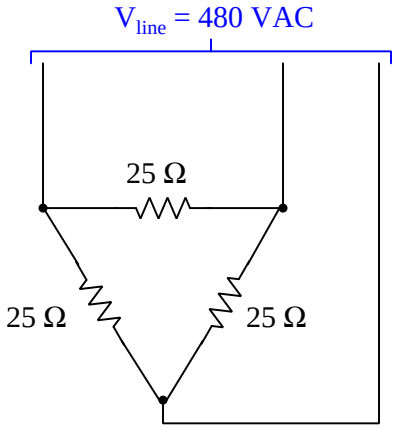
Note how the square-root term (y) is calculated in cell C1, and the denominator term (z) in cell C2. This makes the two final formulae (in cells B1 and B2) simpler to interpret. The positioning of all these cells on the grid is completely arbitrary[10] – all that matters is that they properly reference each other in the formulae.

Spreadsheets are particularly useful for situations where the same set of calculations representing a circuit or other system must be repeated for different initial conditions. The power of a spreadsheet is that it automates what would otherwise be a tedious set of calculations. One specific application of this is to simulate the effects of various components within a circuit failing with abnormal values (e.g. a shorted resistor simulated by making its value nearly zero; an open resistor simulated by making its value extremely large). Another application is analyzing the behavior of a circuit design given new components that are out of specification, and/or aging components experiencing drift over time.

[10] My personal preference is to locate all the "given" data in the upper-left cells of the spreadsheet grid (each data point flanked by a sensible name in the cell to the left and units of measurement in the cell to the right as illustrated in the first distance/time spreadsheet example), sometimes coloring them in order to clearly distinguish which cells contain entered data versus which cells contain computed results from formulae. I like to place all formulae in cells below the given data, and try to arrange them in logical order so that anyone examining my spreadsheet will be able to figure out *how* I constructed a solution. This is a general principle I believe all computer programmers should follow: *document and arrange your code to make it easy for other people to learn from it.*

8.2.3 Total power dissipation

The line voltage to this three-phase load is 480 Volts. How much power (total) is dissipated by the load? How much current is there in each line supplying the load?



P_{total} =

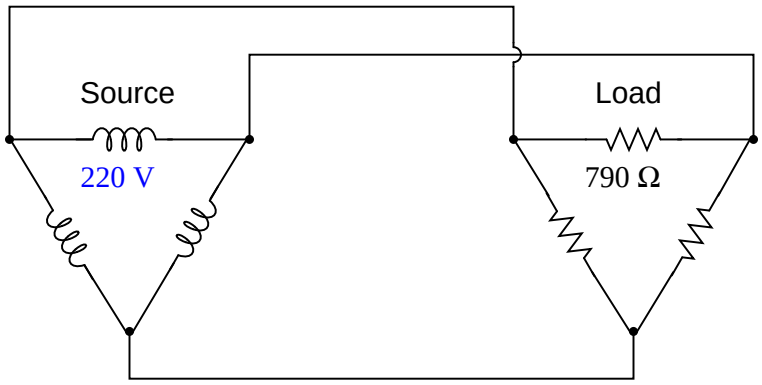
I_{line} =

Challenges

- Suppose the load were unbalanced instead of balanced as it is. How could we calculate total power in this case?

8.2.4 Balanced Delta source and Delta load

Calculate all voltages and currents in this three-phase circuit where a three-phase generator delivers electrical energy to a resistive load:



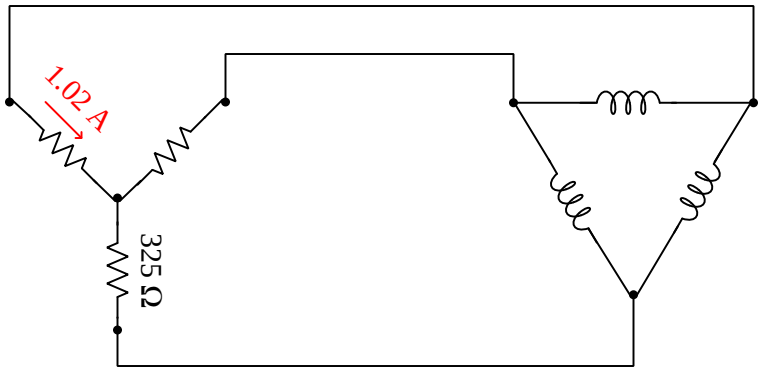
- V_{line} =
- I_{line} =
- V_{phase} (source) =
- I_{phase} (source) =
- V_{phase} (load) =
- I_{phase} (load) =
- P_{total} =

Challenges

- A common challenge for students new to three-phase circuit calculations is remembering when phase and line quantities differ, versus when they are the same. Explain how we may draw correct conclusions based on the properties of *series* versus *parallel* networks.

8.2.5 Balanced Delta source and Wye load

Calculate all voltages and currents in this three-phase circuit where a three-phase generator delivers electrical energy to a resistive load:



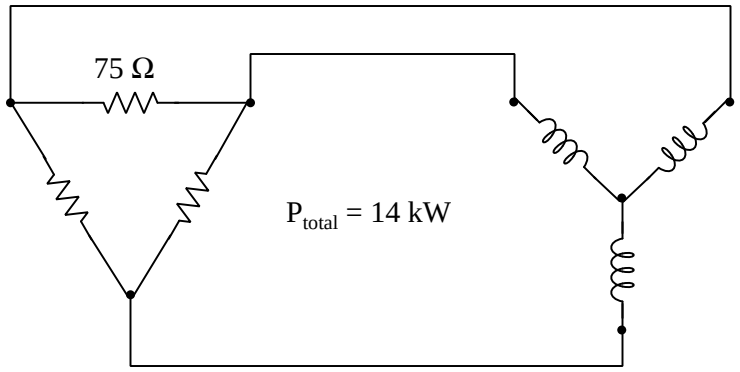
- V_{phase} (source) =
- I_{phase} (source) =
- V_{phase} (load) =
- I_{phase} (load) =
- V_{line} =
- I_{line} =
- P_{total} =

Challenges

- A common challenge for students new to three-phase circuit calculations is remembering when phase and line quantities differ, versus when they are the same. Explain how we may draw correct conclusions based on the properties of *series* versus *parallel* networks.

8.2.6 Balanced Wye source and Delta load

Calculate all voltages and currents in this three-phase circuit where a three-phase generator delivers electrical energy to a resistive load:



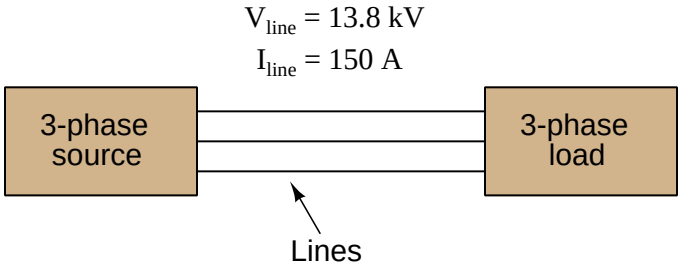
- V_{phase} (source) =
- I_{phase} (source) =
- V_{phase} (load) =
- I_{phase} (load) =
- V_{line} =
- I_{line} =

Challenges

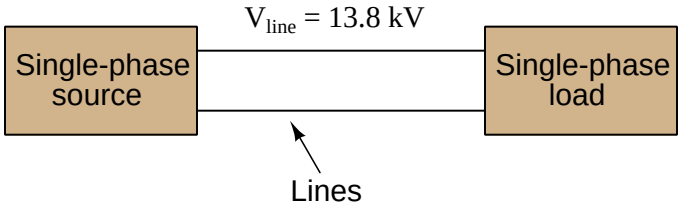
- A common challenge for students new to three-phase circuit calculations is remembering when phase and line quantities differ, versus when they are the same. Explain how we may draw correct conclusions based on the properties of *series* versus *parallel* networks.

8.2.7 Three-phase versus single-phase power transmission

A balanced, three-phase power system has a line voltage of 13.8 kV and a line current of 150 Amperes. How much power is being delivered to the resistive load?



A 13.8 kV *single-phase* system could be designed to provide the same amount of power to a load, but it would require heavier-gauge (more expensive!) conductors. Determine the extra percentage of expense in wire cost (based on the weight of the wires) resulting from the use of single-phase instead of three-phase.



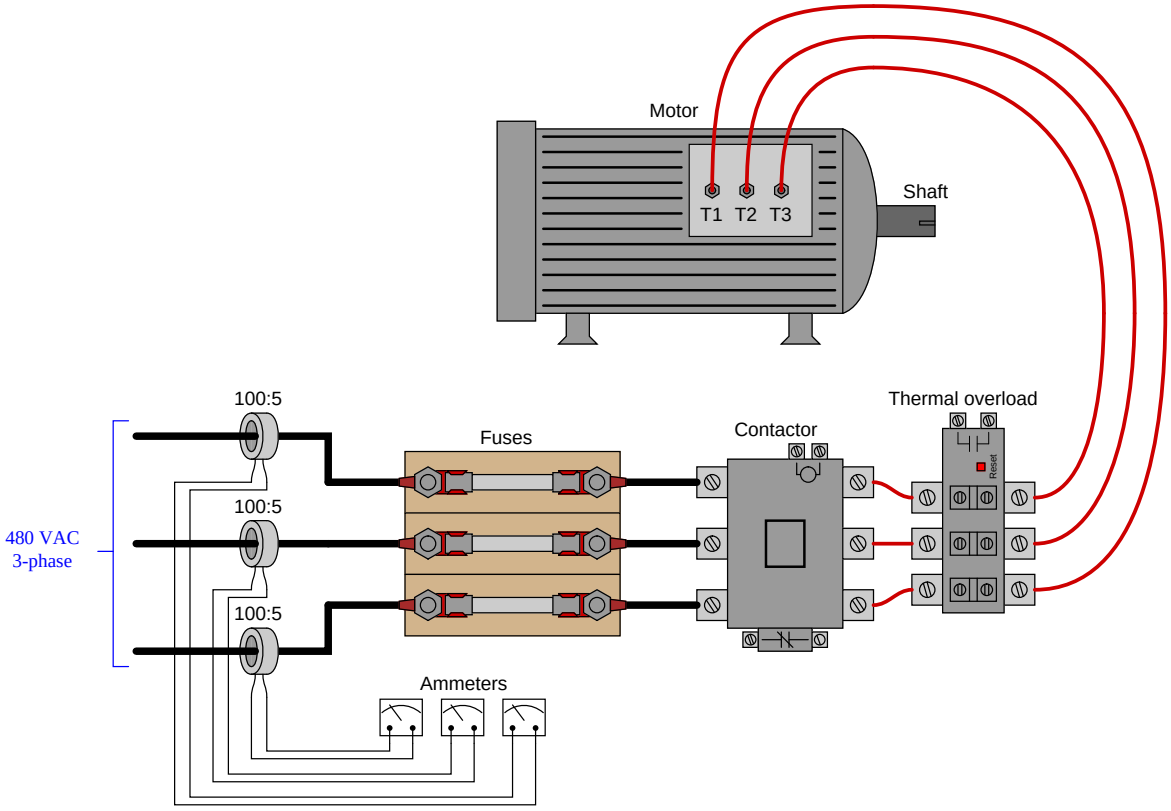
Also, explain why a three-phase power system has a *phase sequence* or *phase rotation* but a single-phase power system does not.

Challenges

- If three-phase requires less conductor metal than single-phase, would four-phase be even better?
- What would a *two-phase* system look like, and would it have a phase rotation?

8.2.8 Three-phase electric motor with CT current measurement

Suppose the current through each of the ammeters is 2.81 Amperes, and the ratio of each current transformer is 100:5. Calculate the horsepower output of this AC motor, assuming a power factor of 1 and an efficiency of 88%:

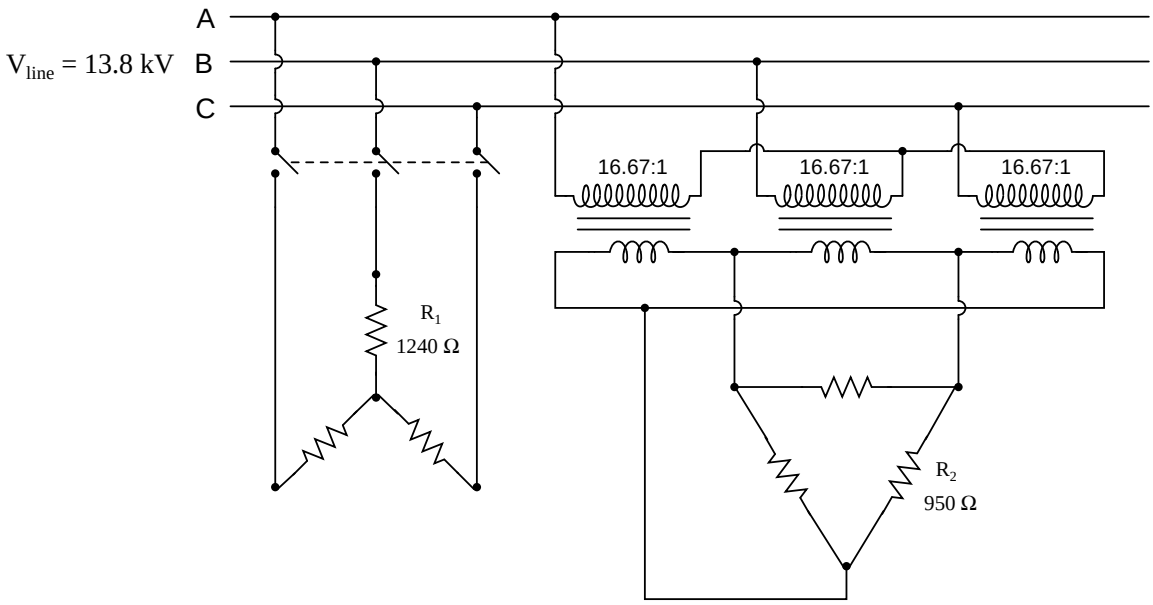


Challenges

- Why do you suppose CTs are being used to aid in current measurement, as opposed to directly-connected ammeters?

8.2.9 Direct-coupled and transformer-coupled loads

Calculate the operating current through each of the load resistances shown in this circuit (assuming each three-phase load is balanced), as well as the power dissipated by each load:

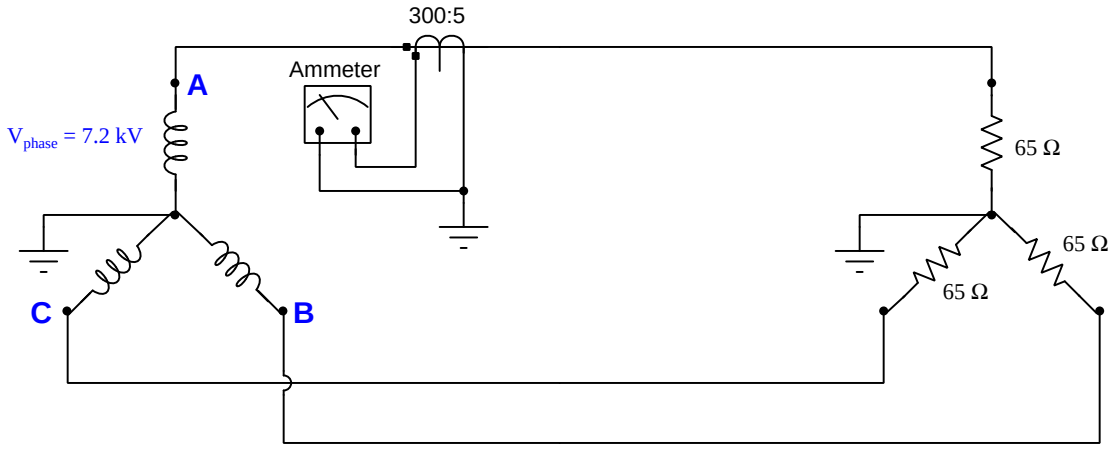


Challenges

- If a resistor in either load happens to fail open, will that fault affect current in any other resistor within that load?
- If a resistor in either load happens to fail open, will that fault affect current in any other resistor within the *other* load?

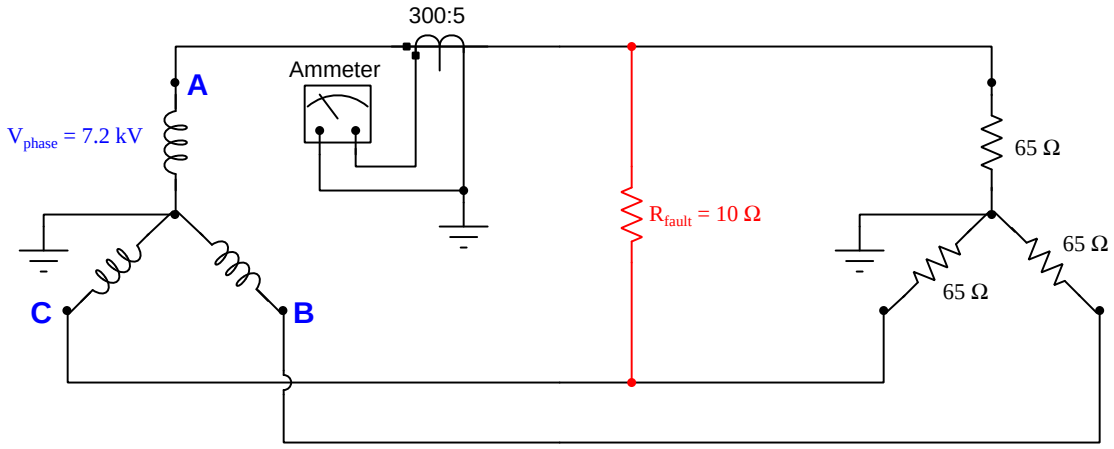
8.2.10 Wye source and Wye load with and without line fault

Calculate the amount of current output to the ammeter by the current transformer (CT) under normal load conditions, assuming a balanced three-phase source and a balanced three-phase load:



$I_{ammeter}$ = _____

Now, re-calculate the ammeter's current supposing a tree branch falls across lines A and C, causing a low-resistance fault:



$I_{ammeter}$ = _____

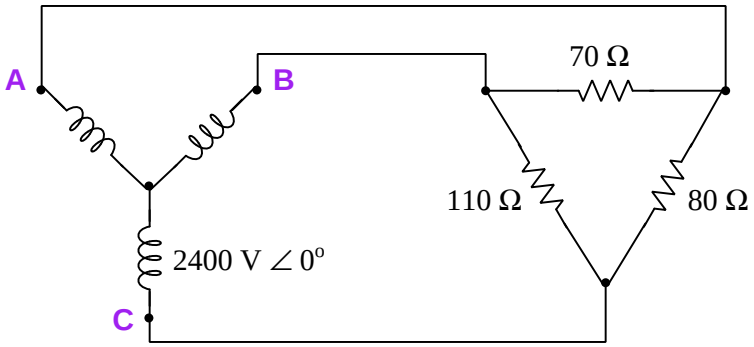
Hint: you will need to consider phase angles for the fault current calculation! Feel free to assume an ABC phase rotation, where $V_A = 7200$ V $\angle 0^o$ and $V_B = 7200$ V $\angle -120^o$ and $V_C = 7200$ V $\angle 120^o$.

Challenges

- Will the load phase currents be affected by this fault?

8.2.11 Balanced Wye source with unbalanced Delta load

Calculate the amount of current passing through each of the line conductors as this balanced Wye source powers an unbalanced Delta load:



CBA phase rotation

$I_A =$

$I_B =$

$I_C =$

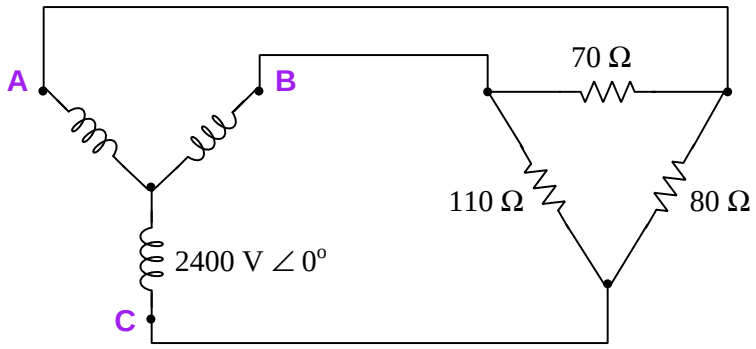
Also, express voltage V_{BC} (voltage and phase angle as sensed by an oscilloscope with probe touching B and reference touching C) as a phasor quantity.

Challenges

- Does the imbalance of this circuit affect voltages, currents, or both.
- Identify some of the fundamental principles of electric circuits (e.g. laws, properties) applicable to the solution of this problem.
- Is an unbalanced load realistic? Why or why not?

8.2.12 Balanced Wye source with unbalanced Delta load and line fault

Calculate the amount of current passing through each of the line conductors as this balanced Wye source powers an unbalanced Delta load with a fault between line conductors B and C, the fault itself equivalent to a 3 Ohm resistance in series with a 1000 μF capacitance:



CBA phase rotation

Note that the fault is not shown in this diagram, and is left for you to insert.

$I_A =$

$I_B =$

$I_C =$

Challenges

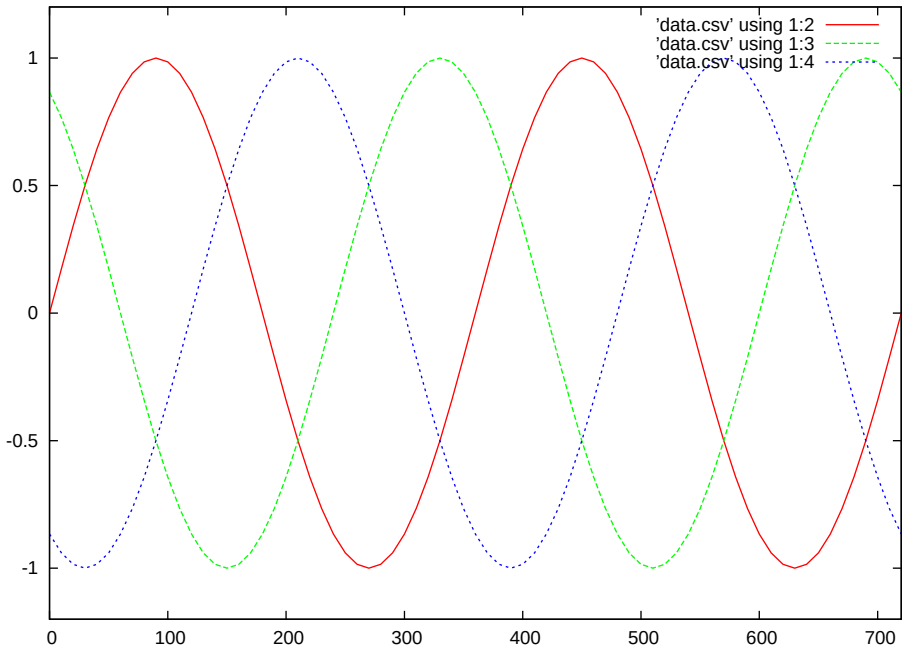
- How does the presence of this fault affect the phasor diagram for line currents?

8.2.13 Three-phase simulation program

The following computer program written in the C language simulates AC waveforms in a three-phase system:

```
#include <stdio.h>
#include <math.h>

float sinecalc(float);

int main (void)
{
  float angle;
  int n;

  printf("Angle , Phase A , Phase B , Phase C");

  for (angle = 0 ; angle <= 720 ; angle = angle + 10.0)
  {
    printf("\n%f", angle);

    for (n = 0 ; n <= 2 ; ++n)
    {
      printf(" , %f", sinecalc(angle + (n * 120)));
    }
  }

  return 0;
}

float sinecalc (float input)
{
  return sin(input * M_PI / 180);
}
```

This program outputs text in a comma-separated-variable (CSV) format suitable for plotting with spreadsheet or other math visualization software, a plot shown on the following page.



Answer the following questions about the program and its corresponding plotted output:

- What purpose is served by the `sinecalc()` function?
- Which waveform in the plot represents the "A" phase, the "B" phase, and the "C" phase?
- How are the three different phase angles of these waveforms implemented in the code?
- Are both `for()` loops strictly necessary in this program?
- What would happen if the `10.0` constant value were set to `30.0` instead?
- How could the code be modified to plot waveforms with RMS values of 277 Volts each?

Challenges

- Do these plotted waveforms have a specific frequency? Why or why not?

8.3 Diagnostic reasoning

These questions are designed to stimulate your deductive and inductive thinking, where you must apply general principles to specific scenarios (deductive) and also derive conclusions about the failed circuit from specific details (inductive). In a Socratic discussion with your instructor, the goal is for these questions to reinforce your recall and use of general circuit principles and also challenge your ability to integrate multiple symptoms into a sensible explanation of what's wrong in a circuit. Your instructor may also pose additional questions based on those assigned, in order to further challenge and sharpen your diagnostic abilities.

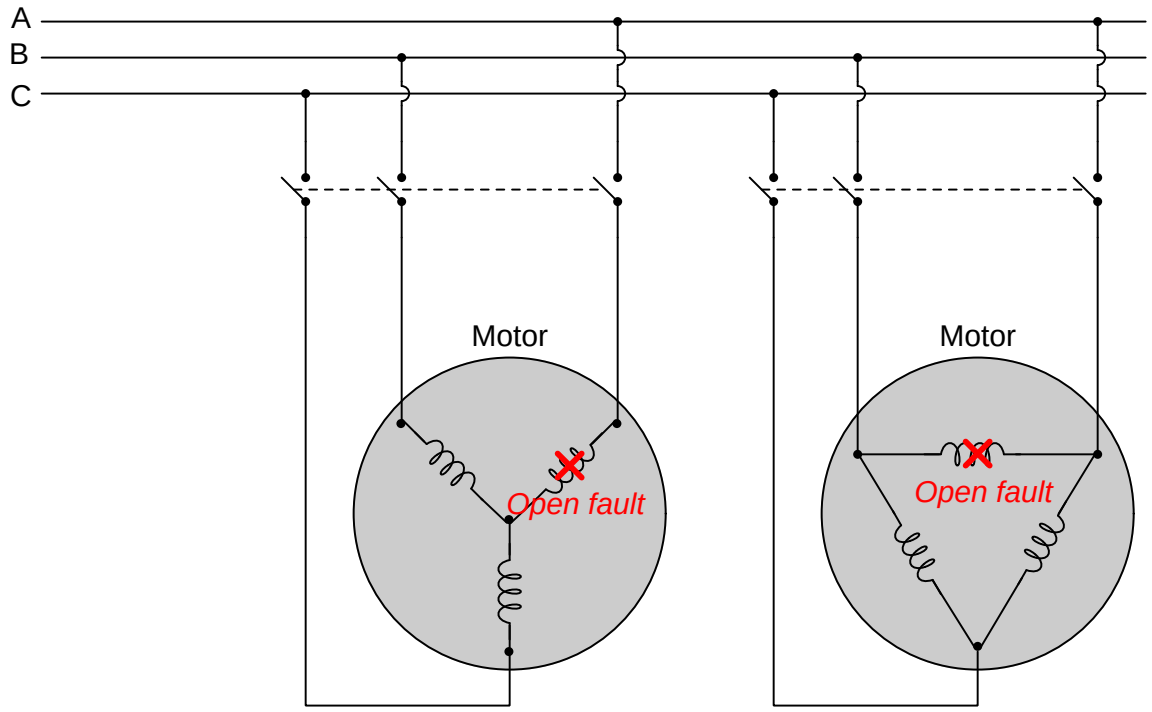
As always, your goal is to fully *explain* your analysis of each problem. Simply obtaining a correct answer is not good enough – you must also demonstrate sound reasoning in order to successfully complete the assignment. Your instructor's responsibility is to probe and challenge your understanding of the relevant principles and analytical processes in order to ensure you have a strong foundation upon which to build further understanding.

You will note a conspicuous lack of answers given for these diagnostic questions. Unlike standard textbooks where answers to every other question are given somewhere toward the back of the book, here in these learning modules students must rely on other means to check their work. The best way by far is to debate the answers with fellow students and also with the instructor during the Socratic dialogue sessions intended to be used with these learning modules. Reasoning through challenging questions with other people is an excellent tool for developing strong reasoning skills.

Another means of checking your diagnostic answers, where applicable, is to use circuit simulation software to explore the effects of faults placed in circuits. For example, if one of these diagnostic questions requires that you predict the effect of an open or a short in a circuit, you may check the validity of your work by simulating that same fault (substituting a very high resistance in place of that component for an open, and substituting a very low resistance for a short) within software and seeing if the results agree.

8.3.1 Winding faults within three-phase motors

Three-phase AC induction motors respond differently to the loss (i.e. "open" fault) of one phase winding, depending on whether they are internally wye- or delta-connected:



Which of these two motor designs will fare better in the event of an open fault in one phase winding, and why?

Challenges

- A good problem-solving technique is to insert quantitative values into a problem which has none. Explain how you might apply such a technique to this problem, and describe how the results could be helpful.

Chapter 9

Projects and Experiments

The following project and experiment descriptions outline things you can build to help you understand circuits. With any real-world project or experiment there exists the potential for physical harm. *Electricity can be very dangerous in certain circumstances, and you should follow proper safety precautions at all times!*

9.1 Recommended practices

This section outlines some recommended practices for all circuits you design and construct.

9.1.1 Safety first!

Electricity, when passed through the human body, causes uncomfortable sensations and in large enough measures[1] will cause muscles to involuntarily contract. The overriding of your nervous system by the passage of electrical current through your body is particularly dangerous in regard to your heart, which is a vital muscle. Very large amounts of current can produce serious internal burns in addition to all the other effects.

Cardio-pulmonary resuscitation (CPR) is the standard first-aid for any victim of electrical shock. This is a very good skill to acquire if you intend to work with others on dangerous electrical circuits. You should never perform tests or work on such circuits unless someone else is present who is proficient in CPR.

As a general rule, any voltage in excess of 30 Volts poses a definitive electric shock hazard, because beyond this level human skin does not have enough resistance to safely limit current through the body. "Live" work of any kind with circuits over 30 volts should be avoided, and if unavoidable should only be done using electrically insulated tools and other protective equipment (e.g. insulating shoes and gloves). If you are unsure of the hazards, or feel unsafe at any time, stop all work and distance yourself from the circuit!

A policy I strongly recommend for students learning about electricity is to *never come into electrical contact*[2] *with an energized conductor, no matter what the circuit's voltage*[3] *level!* Enforcing this policy may seem ridiculous when the circuit in question is powered by a single battery smaller than the palm of your hand, but it is precisely this instilled habit which will save a person from bodily harm when working with more dangerous circuits. Experience has taught me that students who learn early on to be careless with safe circuits have a tendency to be careless later with dangerous circuits!

In addition to the electrical hazards of shock and burns, the construction of projects and running of experiments often poses other hazards such as working with hand and power tools, potential

[1] Professor Charles Dalziel published a research paper in 1961 called "The Deleterious Effects of Electric Shock" detailing the results of electric shock experiments with both human and animal subjects. The threshold of perception for human subjects holding a conductor in their hand was in the range of 1 milliampere of current (less than this for alternating current, and generally less for female subjects than for male). Loss of muscular control was exhibited by half of Dalziel's subjects at less than 10 milliamperes alternating current. Extreme pain, difficulty breathing, and loss of all muscular control occurred for over 99% of his subjects at direct currents less than 100 milliamperes and alternating currents less than 30 milliamperes. In summary, it doesn't require much electric current to induce painful and even life-threatening effects in the human body! Your first and best protection against electric shock is maintaining an insulating barrier between your body and the circuit in question, such that current from that circuit will be unable to flow through your body.

[2] By "electrical contact" I mean either directly touching an energized conductor with any part of your body, or indirectly touching it through a conductive tool. The only physical contact you should ever make with an energized conductor is via an electrically insulated tool, for example a screwdriver with an electrically insulated handle, or an insulated test probe for some instrument.

[3] Another reason for consistently enforcing this policy, even on low-voltage circuits, is due to the dangers that even some low-voltage circuits harbor. A single 12 Volt automobile battery, for example, can cause a surprising amount of damage if short-circuited simply due to the high current levels (i.e. very low internal resistance) it is capable of, even though the voltage level is too low to cause a shock through the skin. Mechanics wearing metal rings, for example, are at risk from severe burns if their rings happen to short-circuit such a battery! Furthermore, even when working on circuits that are simply too low-power (low voltage and low current) to cause any bodily harm, touching them while energized can pose a threat to the circuit components themselves. In summary, it's generally wise (and *always* a good habit to build) to "power down" *any* circuit before making contact between it and your body.

contact with high temperatures, potential chemical exposure, etc. You should never proceed with a project or experiment if you are unaware of proper tool use or lack basic protective measures (e.g. personal protective equipment such as safety glasses) against such hazards.

Some other safety-related practices should be followed as well:

- All power conductors extending outward from the project must be *firmly* strain-relieved (e.g. "cord grips" used on line power cords), so that an accidental tug or drop will not compromise circuit integrity.
- All electrical connections must be sound and appropriately made (e.g. soldered wire joints rather than twisted-and-taped; terminal blocks rather than solderless breadboards for high-current or high-voltage circuits). Use "touch-safe" terminal connections with recessed metal parts to minimize risk of accidental contact.
- Always provide overcurrent protection in any circuit you build. *Always.* This may be in the form of a fuse, a circuit breaker, and/or an electronically current-limited power supply.
- Always ensure circuit conductors are rated for more current than the overcurrent protection limit. *Always.* A fuse does no good if the wire or printed circuit board trace will "blow" before it does!
- Always bond metal enclosures to Earth ground for any line-powered circuit. *Always.* Ensuring an equipotential state between the enclosure and Earth by making the enclosure electrically common with Earth ground ensures no electric shock can occur simply by one's body bridging between the Earth and the enclosure.
- Avoid building a high-energy circuit when a low-energy circuit will suffice. For example, I always recommend beginning students power their first DC resistor circuits using small batteries rather than with line-powered DC power supplies. The intrinsic energy limitations of a dry-cell battery make accidents highly unlikely.
- Use line power receptacles that are GFCI (Ground Fault Current Interrupting) to help avoid electric shock from making accidental contact with a "hot" line conductor.
- Always wear eye protection when working with tools or live systems having the potential to eject material into the air. Examples of such activities include soldering, drilling, grinding, cutting, wire stripping, working on or near energized circuits, etc.
- Always use a step-stool or stepladder to reach high places. Never stand on something not designed to support a human load.
- When in doubt, *ask an expert.* If anything even seems remotely unsafe to you, do not proceed without consulting a trusted person fully knowledgeable in electrical safety.

9.1.2 Other helpful tips

Experience has shown the following practices to be very helpful, especially when students make their own component selections, to ensure the circuits will be well-behaved:

- Avoid resistor values less than 1 kΩ or greater than 100 kΩ, unless such values are definitely necessary[4]. Resistances below 1 kΩ may draw excessive current if directly connected to a voltage source of significant magnitude, and may also complicate the task of accurately measuring current since any ammeter's non-zero resistance inserted in series with a low-value circuit resistor will significantly alter the total resistance and thereby skew the measurement. Resistances above 100 kΩ may complicate the task of measuring voltage since any voltmeter's finite resistance connected in parallel with a high-value circuit resistor will significantly alter the total resistance and thereby skew the measurement. Similarly, AC circuit impedance values should be between 1 kΩ and 100 kΩ, and for all the same reasons.

- Ensure all electrical connections are low-resistance and physically rugged. For this reason, one should avoid *compression splices* (e.g. "butt" connectors), solderless breadboards[5], and wires that are simply twisted together.

- Build your circuit with **testing** in mind. For example, provide convenient connection points for test equipment (e.g. multimeters, oscilloscopes, signal generators, logic probes).

- Design permanent projects with **maintenance** in mind. The more convenient you make maintenance tasks, the more likely they will get done.

- **Always document and save your work**. Circuits lacking schematic diagrams are more difficult to troubleshoot than documented circuits. Similarly, circuit construction is simpler when a schematic diagram precedes construction. Experimental results are easier to interpret when comprehensively recorded. Consider modern videorecording technology for this purpose where appropriate.

- **Record your steps** when troubleshooting. **Talk to yourself** when solving problems. These simple steps clarify thought and simplify identification of errors.

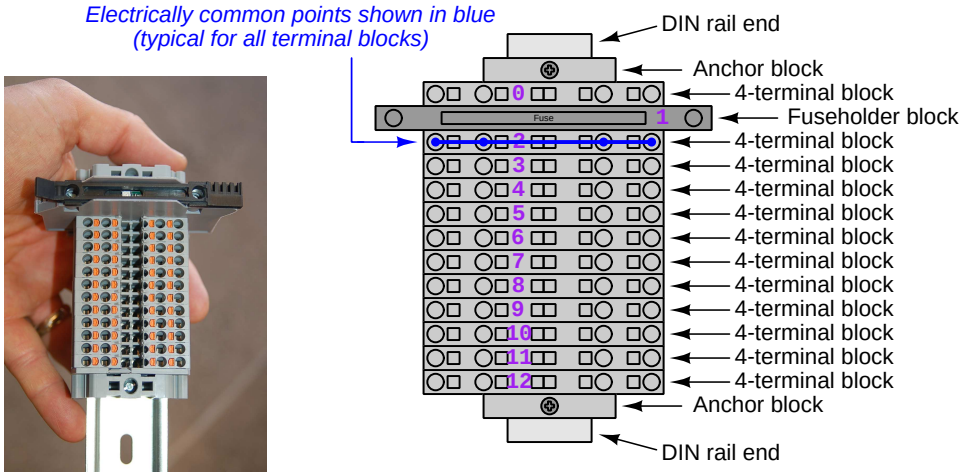
[4] An example of a necessary resistor value much less than 1 kΩ is a *shunt resistor* used to produce a small voltage drop for the purpose of sensing current in a circuit. Such shunt resistors must be low-value in order not to impose an undue load on the rest of the circuit. An example of a necessary resistor value much greater than 100 kΩ is an electrostatic *drain resistor* used to dissipate stored electric charges from body capacitance for the sake of preventing damage to sensitive semiconductor components, while also preventing a path for current that could be dangerous to the person (i.e. shock).

[5] Admittedly, solderless breadboards are very useful for constructing complex electronic circuits with many components, especially DIP-style integrated circuits (ICs), but they tend to give trouble with connection integrity after frequent use. An alternative for projects using low counts of ICs is to solder IC sockets into prototype printed circuit boards (PCBs) and run wires from the soldered pins of the IC sockets to terminal blocks where reliable temporary connections may be made.

9.1.3 Terminal blocks for circuit construction

Terminal blocks are the standard means for making electric circuit connections in industrial systems, and are also quite useful for student-built experiments, providing a robust alternative to solderless breadboards[6]. Terminal blocks provide highly reliable connections rated for significant voltage and current magnitudes, and they force the builder to think carefully about component placement which is an important mental practice. Terminal blocks designed to mount on standard 35 mm DIN rail[7] come in many types and capabilities.

Every student of electricity should build their own terminal block array consisting of several terminal blocks where each block has at least 4 connection points all electrically common to each other[8] and at least one terminal block that is a fuse holder for overcurrent protection. A pair of anchoring blocks hold all terminal blocks securely on the DIN rail, preventing them from sliding off the rail. Each of the terminals should bear a unique number starting from 0. An example is shown in the following photograph and illustration:



Screwless terminal blocks (using internal spring clips to clamp wire and component lead ends) are preferred over screw-based terminal blocks, as they reduce assembly and disassembly time, and also minimize repetitive wrist stress from twisting screwdrivers. Some screwless terminal blocks require the use of a special tool to release the spring clip, while others such as the Phoenix Contact model 3211019 blocks shown in the photograph provide buttons[9] for this task which may be pressed using the tip of any suitable tool such as a small screwdriver or multimeter test probe tip.

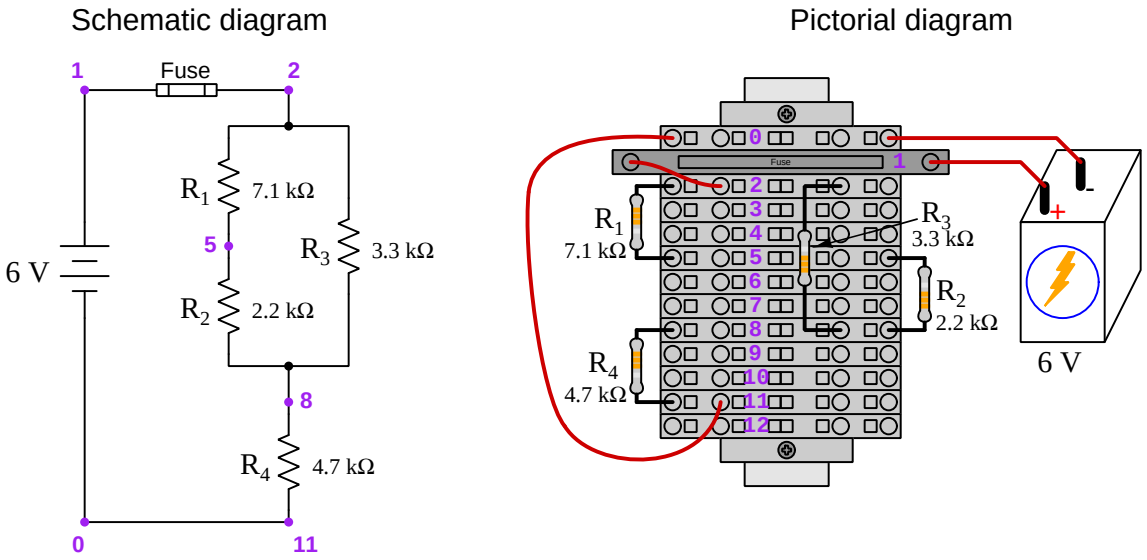
[6] Solderless breadboard are preferable for complicated electronic circuits with multiple integrated "chip" components, but for simpler circuits I find terminal blocks much more practical. An alternative to solderless breadboards for "chip" circuits is to solder chip sockets onto a PCB and then use wires to connect the socket pins to terminal blocks. This also accommodates *surface-mount* components, which solderless breadboards do not.

[7] DIN rail is a metal rail designed to serve as a mounting point for a wide range of electrical and electronic devices such as terminal blocks, fuses, circuit breakers, relay sockets, power supplies, data acquisition hardware, etc.

[8] Sometimes referred to as *equipotential*, *same-potential*, or *potential distribution* terminal blocks.

[9] The small orange-colored squares seen in the above photograph are buttons for this purpose, and may be actuated by pressing with any tool of suitable size.

The following example shows how such a terminal block array might be used to construct a series-parallel resistor circuit consisting of four resistors and a battery:



Numbering on the terminal blocks provides a very natural translation to SPICE[10] netlists, where component connections are identified by terminal number:

```
* Series-parallel resistor circuit
v1 1 0 dc 6
r1 2 5 7100
r2 5 8 2200
r3 2 8 3300
r4 8 11 4700
rfuse 1 2 0.01
rwire 0 11 0.01
.op
.end
```

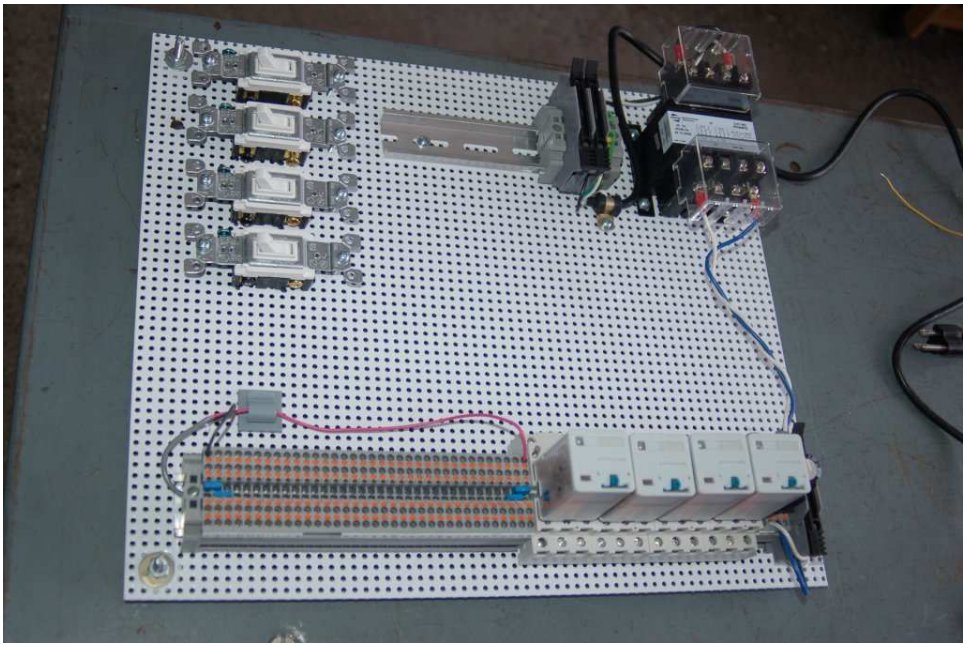
Note the use of two resistances `rfuse` and `rwire` to describe fuse resistance between terminals 1 and 2 and "jumper" wire resistance between terminals 0 and 11, respectively. Being resistances, SPICE requires a resistance value for each, and here we see they have both been set to an arbitrarily low value of 0.01 Ohm appropriate to their real-life resistance values.

Listing all components and wires along with their numbered terminals happens to be a useful documentation method for any circuit built on terminal blocks, independent of SPICE. Such a

[10] SPICE is computer software designed to analyze electrical and electronic circuits. Circuits are described for the computer in the form of *netlists* which are text files listing each component type, connection node numbers, and component values.

"wiring sequence" may be thought of as a *non-graphical description* of an electric circuit, and is exceptionally easy to follow.

An example of a more elaborate terminal block array is shown in the following photograph, with terminal blocks and "ice-cube" style electromechanical relays mounted to DIN rail, which is turn mounted to a perforated subpanel[11]. This "terminal block board" hosts an array of thirty five undedicated terminal block sections, four SPDT toggle switches, four DPDT "ice-cube" relays, a step-down control power transformer, bridge rectifier and filtering capacitor, and several fuses for overcurrent protection:



Four plastic-bottomed "feet" support the subpanel above the benchtop surface, and an unused section of DIN rail stands ready to accept other components. Safety features include electrical bonding of the AC line power cord's ground to the metal subpanel (and all metal DIN rails), mechanical strain relief for the power cord to isolate any cord tension from wire connections, clear plastic finger guards covering the transformer's screw terminals, as well as fused overcurrent protection for the 120 Volt AC line power and the transformer's 12 Volt AC output. The perforated holes happen to be on $\frac{1}{4}$ inch centers with a diameter suitable for tapping with 6-32 machine screw threads, their presence making it very easy to attach other sections of DIN rail, printed circuit boards, or specialized electrical components directly to the grounded metal subpanel. Such a "terminal block board" is an inexpensive[12] yet highly flexible means to construct physically robust circuits using industrial wiring practices.

[11] An electrical *subpanel* is a thin metal plate intended for mounting inside an electrical enclosure. Components are attached to the subpanel, and the subpanel in turn bolts inside the enclosure. Subpanels allow circuit construction outside the confines of the enclosure, which speeds assembly. In this particular usage there is no enclosure, as the subpanel is intended to be used as an open platform for the convenient construction of circuits on a benchtop by students. In essence, this is a modern version of the traditional *breadboard* which was literally a wooden board such as might be used for cutting loaves of bread, but which early electrical and electronic hobbyists used as platforms for the construction of circuits.

[12] At the time of this writing (2019) the cost to build this board is approximately $250 US dollars.

9.1.4 Conducting experiments

An *experiment* is an exploratory act, a test performed for the purpose of assessing some proposition or principle. Experiments are the foundation of the *scientific method*, a process by which careful observation helps guard against errors of speculation. All good experiments begin with an *hypothesis*, defined by the American Heritage Dictionary of the English Language as:

> An assertion subject to verification or proof, as (a) A proposition stated as a basis for argument or reasoning. (b) A premise from which a conclusion is drawn. (c) A conjecture that accounts, within a theory or ideational framework, for a set of facts and that can be used as a basis for further investigation.

Stated plainly, an hypothesis is an *educated guess* about cause and effect. The correctness of this initial guess matters little, because any well-designed experiment will reveal the truth of the matter. In fact, *incorrect* hypotheses are often the most valuable because the experiments they engender lead us to surprising discoveries. One of the beautiful aspects of science is that it is more focused on the process of *learning* than about the status of *being correct*[13]. In order for an hypothesis to be valid, it must be testable[14], which means it must be a claim possible to refute given the right data. Hypotheses impossible to critique are useless.

Once an hypothesis has been formulated, an experiment must be designed to test that hypothesis. A well-designed experiment requires careful regulation of all relevant variables, both for personal safety and for prompting the hypothesized results. If the effects of one particular variable are to be tested, the experiment must be run multiple times with different values of (only) that particular variable. The experiment set up with the "baseline" variable set is called the *control*, while the experiment set up with different value(s) is called the *test* or *experimental*.

For some hypotheses a viable alternative to a physical experiment is a *computer-simulated experiment* or even a *thought experiment*. Simulations performed on a computer test the hypothesis against the physical laws encoded within the computer simulation software, and are particularly useful for students learning new principles for which simulation software is readily available[15].

[13] Science is more about clarifying our view of the universe through a systematic process of error detection than it is about proving oneself to be right. Some scient*ists* may happen to have large egos – and this may have more to do with the ways in which large-scale scientific research is *funded* than anything else – but scientific *method* itself is devoid of ego, and if embraced as a practical philosophy is quite an effective stimulant for humility. Within the education system, scientific method is particularly valuable for helping students break free of the crippling fear of *being wrong*. So much emphasis is placed in formal education on assessing correct retention of facts that many students are fearful of saying or doing anything that might be perceived as a mistake, and of course making mistakes (i.e. having one's hypotheses disproven by experiment) is an indispensable tool for learning. Introducing science in the classroom – *real* science characterized by individuals forming actual hypotheses and testing those hypotheses by experiment – helps students become self-directed learners.

[14] This is the principle of *falsifiability*: that a scientific statement has value only insofar as it is liable to disproof given the requisite experimental evidence. Any claim that is unfalsifiable – that is, a claim which can *never* be disproven by any evidence whatsoever – could be completely wrong and we could never know it.

[15] A very pertinent example of this is learning how to analyze electric circuits using simulation software such as SPICE. A typical experimental cycle would proceed as follows: (1) Find or invent a circuit to analyze; (2) Apply your analytical knowledge to that circuit, predicting all voltages, currents, powers, etc. relevant to the concepts you are striving to master; (3) Run a simulation on that circuit, collecting "data" from the computer when complete; (4) Evaluate whether or not your hypotheses (i.e. predicted voltages, currents, etc.) agree with the computer-generated results; (5) If so, your analyses are (provisionally) correct – if not, examine your analyses and the computer simulation again to determine the source of error; (6) Repeat this process as many times as necessary until you achieve mastery.

Thought experiments are useful for detecting inconsistencies within your own understanding of some subject, rather than testing your understanding against physical reality.

Here are some general guidelines for conducting experiments:

- The clearer and more specific the hypothesis, the better. Vague or unfalsifiable hypotheses are useless because they will fit *any* experimental results, and therefore the experiment cannot teach you anything about the hypothesis.
- Collect as much data (i.e. information, measurements, sensory experiences) generated by an experiment as is practical. This includes the time and date of the experiment, too!
- *Never* discard or modify data gathered from an experiment. If you have reason to believe the data is unreliable, write notes to that effect, but never throw away data just because you think it is untrustworthy. It is quite possible that even "bad" data holds useful information, and that someone else may be able to uncover its value even if you do not.
- Prioritize *quantitative* data over *qualitative* data wherever practical. Quantitative data is more specific than qualitative, less prone to subjective interpretation on the part of the experimenter, and amenable to an arsenal of analytical methods (e.g. statistics).
- Guard against your own bias(es) by making your experimental results available to others. This allows other people to scrutinize your experimental design and collected data, for the purpose of detecting and correcting errors you may have missed. Document your experiment such that others may independently replicate it.
- Always be looking for sources of error. No physical measurement is perfect, and so it is impossible to achieve *exact* values for any variable. Quantify the amount of uncertainty (i.e. the "tolerance" of errors) whenever possible, and be sure your hypothesis does not depend on precision better than this!
- Always remember that scientific confirmation is provisional – no number of "successful" experiments will prove an hypothesis true for all time, but a single experiment can disprove it. Put into simpler terms, *truth is elusive but error is within reach.*
- Remember that scientific method is about *learning*, first and foremost. An unfortunate consequence of scientific triumph in modern society is that science is often viewed by non-practitioners as an unerring source of truth, when in fact science is an ongoing process of challenging existing ideas to probe for errors and oversights. This is why it is perfectly acceptable to have a failed hypothesis, and why the only truly failed experiment is one where nothing was learned.

The following is an example of a well-planned and executed experiment, in this case a physical experiment demonstrating Ohm's Law.

```
Planning Time/Date = 09:30 on 12 February 2019

HYPOTHESIS: the current through any resistor should be exactly proportional
to the voltage impressed across it.

PROCEDURE: connect a resistor rated 1 k Ohm and 1/4 Watt to a variable-voltage
DC power supply.  Use an ammeter in series to measure resistor current and
a voltmeter in parallel to measure resistor voltage.

RISKS AND MITIGATION: excessive power dissipation may harm the resistor and/
or pose a burn hazard, while excessive voltage poses an electric shock hazard.
30 Volts is a safe maximum voltage for laboratory practices, and according to
Joule's Law a 1000 Ohm resistor will dissipate 0.25 Watts at 15.81 Volts
(P = V^2 / R), so I will remain below 15 Volts just to be safe.

Experiment Time/Date = 10:15 on 12 February 2019

DATA COLLECTED:
  (Voltage)    (Current)         (Voltage)   (Current)
   0.000 V   =  0.000 mA          8.100    =  7.812 mA
   2.700 V   =  2.603 mA          10.00 V  =  9.643 mA
   5.400 V   =  5.206 mA          14.00 V  =  13.49 mA

Analysis Time/Date = 10:57 on 12 February 2019

ANALYSIS: current definitely increases with voltage, and although I expected
exactly one milliAmpere per Volt the actual current was usually less than
that.  The voltage/current ratios ranged from a low of 1036.87 (at 8.1 Volts)
to a high of 1037.81 (at 14 Volts), but this represents a variance of only
-0.0365% to +0.0541% from the average, indicating a very consistent
proportionality -- results consistent with Ohm's Law.

ERROR SOURCES: one major source of error is the resistor's value itself.  I
did not measure it, but simply assumed color bands of brown-black-red meant
exactly 1000 Ohms.  Based on the data I think the true resistance is closer
to 1037 Ohms.  Another possible explanation is multimeter calibration error.
However, neither explains the small positive and negative variances from the
average.  This might be due to electrical noise, a good test being to repeat
the same experiment to see if the variances are the same or different.  Noise
should generate slightly different results every time.
```

The following is an example of a well-planned and executed *virtual* experiment, in this case demonstrating Ohm's Law using a computer (SPICE) simulation.

```
Planning Time/Date = 12:32 on 14 February 2019

HYPOTHESIS: for any given resistor, the current through that resistor should be
exactly proportional to the voltage impressed across it.

PROCEDURE: write a SPICE netlist with a single DC voltage source and single
1000 Ohm resistor, then use NGSPICE version 26 to perform a "sweep" analysis
from 0 Volts to 25 Volts in 5 Volt increments.

    * SPICE circuit
    v1 1 0 dc
    r1 1 0 1000
    .dc v1 0 25 5
    .print dc v(1) i(v1)
    .end

RISKS AND MITIGATION: none.

DATA COLLECTED:
       DC transfer characteristic  Thu Feb 14 13:05:08  2019
    ------------------------------------------------------------
    Index   v-sweep         v(1)            v1#branch
    ------------------------------------------------------------
    0       0.000000e+00    0.000000e+00    0.000000e+00
    1       5.000000e+00    5.000000e+00    -5.00000e-03
    2       1.000000e+01    1.000000e+01    -1.00000e-02
    3       1.500000e+01    1.500000e+01    -1.50000e-02
    4       2.000000e+01    2.000000e+01    -2.00000e-02
    5       2.500000e+01    2.500000e+01    -2.50000e-02

Analysis Time/Date = 13:06 on 14 February 2019

ANALYSIS: perfect agreement between data and hypothesis -- current is precisely
1/1000 of the applied voltage for all values.  Anything other than perfect
agreement would have probably meant my netlist was incorrect.  The negative
current values surprised me, but it seems this is just how SPICE interprets
normal current through a DC voltage source.

ERROR SOURCES: none.
```

As gratuitous as it may seem to perform experiments on a physical law as well-established as Ohm's Law, even the examples listed previously demonstrate opportunity for real learning. In the physical experiment example, the student should identify and explain why their data does not perfectly agree with the hypothesis, and this leads them naturally to consider sources of error. In the computer-simulated experiment, the student is struck by SPICE's convention of denoting regular current through a DC voltage source as being *negative* in sign, and this is also useful knowledge for future simulations. Scientific experiments are most interesting when things *do not* go as planned!

Aside from verifying well-established physical laws, simple experiments are extremely useful as educational tools for a wide range of purposes, including:

- Component familiarization (e.g. *Which terminals of this switch connect to the NO versus NC contacts?*)
- System testing (e.g. *How heavy of a load can my AC-DC power supply source before the semiconductor components reach their thermal limits?*)
- Learning programming languages (e.g. *Let's try to set up an "up" counter function in this PLC!*)

Above all, the priority here is to inculcate the habit of hypothesizing, running experiments, and analyzing the results. This experimental cycle not only serves as an excellent method for self-directed learning, but it also works exceptionally well for troubleshooting faults in complex systems, and for these reasons should be a part of every technician's and every engineer's education.

9.1.5 Constructing projects

Designing, constructing, and testing projects is a very effective means of practical education. Within a formal educational setting, projects are generally chosen (or at least vetted) by an instructor to ensure they may be reasonably completed within the allotted time of a course or program of study, and that they sufficiently challenge the student to learn certain important principles. In a self-directed environment, projects are just as useful as a learning tool but there is some risk of unwittingly choosing a project beyond one's abilities, which can lead to frustration.

Here are some general guidelines for managing projects:

- Define your goal(s) before beginning a project: what do you wish to achieve in building it? What, exactly, should the completed project *do*?
- Analyze your project prior to construction. Document it in appropriate forms (e.g. schematic diagrams), predict its functionality, anticipate all associated risks. In other words, *plan ahead*.
- Set a reasonable budget for your project, and stay within it.
- Identify any deadlines, and set reasonable goals to meet those deadlines.
- Beware of *scope creep*: the tendency to modify the project's goals before it is complete.
- Document your progress! An easy way to do this is to use photography or videography: take photos and/or videos of your project as it progresses. Document failures as well as successes, because both are equally valuable from the perspective of learning.

9.2 Experiment: SPICE modeling of a three-phase AC circuit

SPICE is a powerful computer-based simulation tool for circuit analysis, and if mastered provides a way for students to create their own practice problems. In this experiment, we will do just that: create a three-phase AC circuit practice problem, and use SPICE to give us the answer key!

Your hypothesis may simply be the predicted values for voltage and/or current in any portions of the three-phase network. Feel free to take any three-phase AC circuit found in this learning module – especially one with an accompanying SPICE netlist – modify it, and use this as the basis of your experiment. The *SPICE Modeling of Power Circuits* learning module is another resource for example circuits and SPICE netlists, which you may sample and modify.

EXPERIMENT CHECKLIST:

- Prior to experimentation:
  - ☑ Write an hypothesis (i.e. a detailed description of what you expect will happen) unambiguous enough that it could be disproven given the right data.
  - ☑ Write a procedure to test the hypothesis, complete with adequate controls and documentation (e.g. schematic diagrams, programming code).
  - ☑ Identify any risks (e.g. shock hazard, component damage) and write a mitigation plan based on best practices and component ratings.
- During experimentation:
  - ☑ Safe practices followed at all times (e.g. no contact with energized circuit).
  - ☑ Correct equipment usage according to manufacturer's recommendations.
  - ☑ All data collected, ideally quantitative with full precision (i.e. no rounding).
- After each experimental run:
  - ☑ If the results fail to match the hypothesis, identify the error(s), correct the hypothesis and/or revise the procedure, and re-run the experiment.
  - ☑ Identify any uncontrolled sources of error in the experiment.
- After all experimental re-runs:
  - ☑ Save all data for future reference.
  - ☑ Write an analysis of experimental results and lessons learned.

Challenges

- Science is an *iterative* process, and for this reason is never complete. Following the results of your experiment, what would you propose for your *next* hypothesis and *next* experimental procedure? Hint: if your experiment produced any unexpected results, exploring those unexpected results is often a very good basis for the next experiment!
- It is quite likely you may need to insert some extraneous components in your simulated circuit in order to placate some of SPICE's idiosyncrasies. Identify some of these, and explain the compensating measures you must take to satisfy SPICE.

Appendix A

Problem-Solving Strategies

The ability to solve complex problems is arguably one of the most valuable skills one can possess, and this skill is particularly important in any science-based discipline.

- Study principles, not procedures. Don't be satisfied with merely knowing how to compute solutions – learn *why* those solutions work.
- Identify what it is you need to solve, identify all relevant data, identify all units of measurement, identify any general principles or formulae linking the given information to the solution, and then identify any "missing pieces" to a solution. Annotate all diagrams with this data.
- Sketch a diagram to help visualize the problem. When building a real system, always devise a plan for that system and analyze its function *before* constructing it.
- Follow the units of measurement and meaning of every calculation. If you are ever performing mathematical calculations as part of a problem-solving procedure, and you find yourself unable to apply each and every intermediate result to some aspect of the problem, it means you don't understand what you are doing. Properly done, every mathematical result should have practical meaning for the problem, and not just be an abstract number. You should be able to identify the proper units of measurement for each and every calculated result, and show where that result fits into the problem.
- Perform "thought experiments" to explore the effects of different conditions for theoretical problems. When troubleshooting real systems, perform *diagnostic tests* rather than visually inspecting for faults, the best diagnostic test being the one giving you the most information about the nature and/or location of the fault with the fewest steps.
- Simplify the problem until the solution becomes obvious, and then use that obvious case as a model to follow in solving the more complex version of the problem.
- Check for exceptions to see if your solution is incorrect or incomplete. A good solution will work for *all* known conditions and criteria. A good example of this is the process of testing scientific hypotheses: the task of a scientist is not to find support for a new idea, but rather to *challenge* that new idea to see if it holds up under a battery of tests. The philosophical

principle of *reductio ad absurdum* (i.e. disproving a general idea by finding a specific case where it fails) is useful here.

- Work "backward" from a hypothetical solution to a new set of given conditions.
- Add quantities to problems that are qualitative in nature, because sometimes a little math helps illuminate the scenario.
- Sketch graphs illustrating how variables relate to each other. These may be quantitative (i.e. with realistic number values) or qualitative (i.e. simply showing increases and decreases).
- Treat quantitative problems as qualitative in order to discern the relative magnitudes and/or directions of change of the relevant variables. For example, try determining what happens if a certain variable were to increase or decrease before attempting to precisely calculate quantities: how will each of the dependent variables respond, by increasing, decreasing, or remaining the same as before?
- Consider limiting cases. This works especially well for qualitative problems where you need to determine which direction a variable will change. Take the given condition and magnify that condition to an extreme degree as a way of simplifying the direction of the system's response.
- Check your work. This means regularly testing your conclusions to see if they make sense. This does *not* mean repeating the same steps originally used to obtain the conclusion(s), but rather to use some other means to check validity. Simply repeating procedures often leads to *repeating the same errors* if any were made, which is why alternative paths are better.

Appendix B

Instructional philosophy

B.1 First principles of learning

- **Anyone can learn anything** given appropriate time, effort, resources, challenges, encouragement, and expectations. Dedicating time and investing effort are the student's responsibility; providing resources, challenges, and encouragement are the teacher's responsibility; high expectations are a responsibility shared by both student and teacher.

- **Transfer is not automatic.** The human mind has a natural tendency to compartmentalize information, which means the process of taking knowledge learned in one context and applying it to another usually does not come easy and therefore should never be taken for granted.

- **Learning is iterative.** The human mind rarely learns anything perfectly on the first attempt. Anticipate mistakes and plan for multiple tries to achieve full understanding, using the lessons of those mistakes as feedback to guide future attempts.

- **Information is absorbed, but understanding is created.** Facts and procedures may be memorized easily enough by repeated exposure, but the ability to reliably apply principles to novel scenarios only comes through intense personal effort. This effort is fundamentally creative in nature: casting new concepts in one's own words, running mental experiments to perceive patterns and trends, etc. These acts of making knowledge "one's own" need not be perfect in order to be effective, as the value lies in the activity and not necessarily the finished product. Depth of understanding and permanence of memory are both directly related to mental effort during the learning process – i.e. *easy learning is shallow and fleeting.*

- **Education trumps training.** There is no such thing as an entirely isolated subject, as all fields of knowledge are connected. Training is narrowly-focused and task-oriented. Education is broad-based and principle-oriented. When preparing for a life-long technical career, education beats training every time.

- **Character matters.** Poor habits are more destructive than deficits of knowledge or skill. This is especially true in collective endeavors, where a team's ability to function depends on trust between its members. Simply put, no one wants an untrustworthy person on their team. An essential component of education then, is character development.

- **People learn to be responsible by bearing responsibility.** An irresponsible person is someone who has never *had* to be responsible for anything that mattered enough to them. Just as anyone can learn anything, anyone can become responsible if the personal cost of irresponsibility becomes high enough.

- **What gets measured, gets done.** Accurate and relevant assessment of learning is key to ensuring all students learn. Therefore, it is imperative to measure what matters.

- **Failure is nothing to fear.** Every human being fails, and fails in multiple ways at multiple times. Eventual success only happens when we don't stop trying.

B.2 Proven strategies for instructors

- Assume every student is capable of learning anything they desire given the proper conditions. Treat them as capable adults by granting real responsibility and avoiding artificial incentives such as merit or demerit points.
- Create a consistent culture of high expectations across the entire program of study. Demonstrate and encourage patience, persistence, and a healthy sense of self-skepticism. Anticipate and de-stigmatize error. Teach respect for the capabilities of others as well as respect for one's own fallibility.
- Replace lecture with "inverted" instruction, where students first encounter new concepts through reading and then spend class time in Socratic dialogue with the instructor exploring those concepts and solving problems individually. There is a world of difference between observing someone solve a problem versus actually solving a problem yourself, and so the point of this form of instruction is to place students in a position where they *cannot* passively observe.
- Require students to read extensively, write about what they learn, and dialogue with you and their peers to sharpen their understanding. Apply Francis Bacon's advice that "reading maketh a full man; conference a ready man; and writing an exact man". These are complementary activities helping students expand their confidence and abilities.
- Use artificial intelligence (AI) to challenge student understanding rather than merely provide information. Find productive ways for AI to critique students' clarity of thought and of expression, for example by employing AI as a Socratic-style interlocutor or as a reviewer of students' journals. Properly applied, AI has the ability to expand student access to critical review well outside the bounds of their instructor's reach.
- Build frequent and rapid feedback into the learning process so that students know at all times how well they are learning, to identify problems early and fix them before they grow. Model the intellectual habit of self-assessing and self-correcting your own understanding (i.e. a cognitive *feedback loop*), encouraging students to do the same.
- Use "mastery" as the standard for every assessment, which means the exam or experiment or project must be done with 100% competence in order to pass. Provide students with multiple opportunity for re-tries (different versions of the assessment every time).
- Require students to devise their own hypotheses and procedures on all experiments, so that the process is truly a scientific one. Have students assess their proposed experimental procedures for risk and devise mitigations for those risks. Let nothing be pre-designed about students' experiments other than a stated task (i.e. what principle the experiment shall test) at the start and a set of demonstrable knowledge and skill objectives at the end.
- Have students build as much of their lab equipment as possible: building power sources, building test assemblies[1], and building complete working systems (no kits!). In order to provide

[1] In the program I teach, every student builds their own "Development Board" consisting of a metal chassis with DIN rail, terminal blocks, and an AC-DC power supply of their own making which functions as a portable lab environment they can use at school as well as take home.

this same "ground-up" experience for every new student, this means either previous students take their creations with them, or the systems get disassembled in preparation for the new students, or the systems grow and evolve with each new student group.

- Incorporate external accountability for you and for your students, continuously improving the curriculum and your instructional methods based on proven results. Have students regularly network with active professionals through participation in advisory committee meetings, service projects, tours, jobshadows, internships, etc. Practical suggestions include requiring students to design and build projects for external clients (e.g. community groups, businesses, different departments within the institution), and also requiring students attend all technical advisory committee meetings and to dialogue with the industry representatives at those meetings.
- Repeatedly explore difficult-to-learn concepts across multiple courses, so that students have multiple opportunities to build their understanding.
- Relate all new concepts, whenever possible, to previous concepts and to relevant physical laws. Challenge each and every student, every day, to *reason* from concept to concept and to explain the logical connections between. Challenge students to verify their conclusions by multiple approaches (e.g. double-checking their work using different methods). Ask *"Why?"* often.
- Maintain detailed records on each student's performance and share these records privately with them. These records should include academic performance as well as professionally relevant behavioral tendencies.
- Hold mandatory "check-in" meetings between all program faculty and each new student during their first term. Offer these to all other students as an option, except for any students continuing to manifest unprofessional behaviors, poor academic performance, or who have some other need for a face-to-face meeting with faculty.
- Address problems while they are small, before they grow larger. This is equally true for tutoring technical concepts as it is for helping students build professional habits.
- Build rigorous quality control into the curriculum to ensure every student masters every important concept, and that the mastery is retained over time. This includes (1) review questions added to every exam to re-assess knowledge taught in previous terms, (2) cumulative exams at the end of every term to re-assess all important concepts back to the very beginning of the program, and (3) review assessments in practical (hands-on) coursework to ensure critically-important skills were indeed taught and are still retained. What you will find by doing this is that it actually boosts retention of students by ensuring that important knowledge gets taught and is retained over long spans of time. In the absence of such quality control, student learning and retention tends to be spotty and this contributes to drop-out and failure rates later in their education.
- Finally, *never rush learning.* Education is not a race. Give your students ample time to digest complex ideas, as you continually remind yourself of just how long it took you to achieve mastery! Long-term retention and the consistently correct application of concepts are always the result of *focused effort over long periods of time* which means there are no shortcuts to learning.

B.3 Proven strategies for students

The single most important piece of advice I have for any student of any subject is to take responsibility for your own development in all areas of life including mental development. Expecting others in your life to entirely guide your own development is a recipe for disappointment. This is just as true for students enrolled in formal learning institutions as it is for auto-didacts pursuing learning entirely on their own. Learning to think in new ways is key to being able to gainfully use information, to make informed decisions about your life, and to best serve those you care about. With this in mind, I offer the following advice to students:

- **Approach all learning as valuable**. No matter what course you take, no matter who you learn from, no matter the subject, there is something useful in every learning experience. If you don't see the value of every new experience, you are not looking closely enough!
- **Continually challenge yourself**. Let other people take shortcuts and find easy answers to easy problems. The purpose of education is to stretch your mind, in order to shape it into a more powerful tool. This doesn't come by taking the path of least resistance. An excellent analogy for an empowering education is productive physical exercise: becoming stronger, more flexible, and more persistent only comes through intense personal effort.
- **Master the use of language**. This includes reading extensively, writing every day, listening closely, and speaking articulately. To a great extent language channels and empowers thought, so the better you are at wielding language the better you will be at grasping abstract concepts and articulating them not only for your benefit but for others as well.
- **Do not limit yourself to the resources given to you**. Read books that are not on the reading list. Run experiments that aren't assigned to you. Form study groups outside of class. Take an entrepreneurial approach to your own education, as though it were a business you were building for your future benefit.
- **Express and share what you learn**. Take every opportunity to teach what you have learned to others, as this will not only help them but will also strengthen your own understanding[2].
- Realize that **no one can give you understanding**, just as no one can give you physical fitness. These both must be *built*.
- **Above all, recognize that learning is hard work, and that a certain level of frustration is unavoidable**. There are times when you will struggle to grasp some of these concepts, and that struggle is a natural thing. Take heart that it will yield with persistent and varied[3] effort, and never give up! That concepts don't immediately come to you is not a sign of something wrong, but rather of something right: that you have found a worthy challenge!

[2] On a personal note, I was surprised to learn just how much my own understanding of electronics and related subjects was strengthened by becoming a teacher. When you are tasked every day with helping other people grasp complex topics, it catalyzes your own learning by giving you powerful incentives to study, to articulate your thoughts, and to reflect deeply on the process of learning.

[3] As the old saying goes, "Insanity is trying the same thing over and over again, expecting different results." If you find yourself stumped by something in the text, you should attempt a different approach. Alter the thought experiment, change the mathematical parameters, do whatever you can to see the problem in a slightly different light, and then the solution will often present itself more readily.

B.4 Design of these learning modules

"The unexamined circuit is not worth energizing" – Socrates (if he had taught electricity)

These learning modules, although useful for self-study, were designed to be used in a formal learning environment where a subject-matter expert challenges students to digest the content and exercise their critical thinking abilities in the answering of questions and in the construction and testing of working circuits. Every effort has been made to embed the following instructional and assessment philosophies within:

- The first goal of education is to enhance clear and independent thought, in order that every student reach their fullest potential in a highly complex and inter-dependent world. Robust reasoning is *always* more important than particulars of any subject matter, because its application is universal.
- Literacy is fundamental to independent learning and thought because text continues to be the most efficient way to communicate complex ideas over space and time. Those who cannot read with ease are limited in their ability to acquire knowledge and perspective.
- Articulate communication is fundamental to work that is complex and interdisciplinary.
- Faulty assumptions and poor reasoning are best corrected through challenge, not presentation. The rhetorical technique of *reductio ad absurdum* (disproving an assertion by exposing an absurdity) works well to discipline student's minds, not only to correct the problem at hand but also to learn how to detect and correct future errors.
- Important principles should be repeatedly explored and widely applied throughout a course of study, not only to reinforce their importance and help ensure their mastery, but also to showcase the interconnectedness and utility of knowledge.

These learning modules were expressly designed to be used in an "inverted" teaching environment[4] where students first read the introductory and tutorial chapters on their own, then individually attempt to answer the questions and construct working circuits according to the experiment and project guidelines. The instructor never lectures, but instead meets regularly with each individual student to review their progress, answer questions, identify misconceptions, and challenge the student to new depths of understanding through further questioning. Regular meetings between instructor and student should resemble a Socratic[5] dialogue, where questions serve as scalpels to dissect topics and expose assumptions. The student passes each module only after consistently demonstrating their ability to logically analyze and correctly apply all major concepts in each question or project/experiment. The instructor must be vigilant in probing each student's understanding to ensure they are truly *reasoning* and not just *memorizing*. This is why "Challenge" points appear throughout, as prompts for students to think deeper about topics and as starting points for instructor queries. Sometimes these challenge points require additional knowledge that hasn't been covered in the series to answer in full. This is okay, as the major purpose of the Challenges is to stimulate analysis and synthesis on the part of each student.

The instructor must possess enough mastery of the subject matter and awareness of students' reasoning to generate their own follow-up questions to practically any student response. Even completely correct answers given by the student should be challenged by the instructor for the purpose of having students practice articulating their thoughts and defending their reasoning. Conceptual errors committed by the student should be exposed and corrected not by direct instruction, but rather by reducing the errors to an absurdity[6] through well-chosen questions and thought experiments posed by the instructor. Becoming proficient at this style of instruction requires time and dedication, but the positive effects on critical thinking for both student and instructor are spectacular.

An inspection of these learning modules reveals certain unique characteristics. One of these is a bias toward thorough explanations in the tutorial chapters. Without a live instructor to explain concepts and applications to students, the text itself must fulfill this role. This philosophy results in lengthier explanations than what you might typically find in a textbook, each step of the reasoning process fully explained, including footnotes addressing common questions and concerns students raise while learning these concepts. Each tutorial seeks to not only explain each major concept in sufficient detail, but also to explain the logic of each concept and how each may be developed

[4] In a traditional teaching environment, students first encounter new information via *lecture* from an expert, and then independently apply that information via *homework*. In an "inverted" course of study, students first encounter new information via *homework*, and then independently apply that information under the scrutiny of an expert. The expert's role in lecture is to simply *explain*, but the expert's role in an inverted session is to *challenge*, *critique*, and if necessary *explain* where gaps in understanding still exist.

[5] Socrates is a figure in ancient Greek philosophy famous for his unflinching style of questioning. Although he authored no texts, he appears as a character in Plato's many writings. The essence of Socratic philosophy is to leave no question unexamined and no point of view unchallenged. While purists may argue a topic such as electric circuits is too narrow for a true Socratic-style dialogue, I would argue that the essential thought processes involved with scientific reasoning on *any* topic are not far removed from the Socratic ideal, and that students of electricity and electronics would do very well to challenge assumptions, pose thought experiments, identify fallacies, and otherwise employ the arsenal of critical thinking skills modeled by Socrates.

[6] This rhetorical technique is known by the Latin phrase *reductio ad absurdum*. The concept is to expose errors by counter-example, since only one solid counter-example is necessary to disprove a universal claim. As an example of this, consider the common misconception among beginning students of electricity that voltage cannot exist without current. One way to apply *reductio ad absurdum* to this statement is to ask how much current passes through a fully-charged battery connected to nothing (i.e. a clear example of voltage existing without current).

from "first principles". Again, this reflects the goal of developing clear and independent thought in students' minds, by showing how clear and logical thought was used to forge each concept. Students benefit from witnessing a model of clear thinking in action, and these tutorials strive to be just that.

Another feature of these learning modules is that they do not treat topics in isolation. Rather, important concepts are introduced early in the series, and appear repeatedly as stepping-stones toward other concepts in subsequent modules. This helps to avoid the "compartmentalization" of knowledge, demonstrating the inter-connectedness of concepts and simultaneously reinforcing them. Each module is fairly complete in itself, reserving the beginning of its tutorial to a review of foundational concepts.

To high standards of education,

Tony R. Kuphaldt

Appendix C

Tools used

I am indebted to the developers of many open-source software applications in the creation of these learning modules. The following is a list of these applications with some commentary on each.

You will notice a theme common to many of these applications: a bias toward *code*. Although I am by no means an expert programmer in any computer language, I understand and appreciate the flexibility offered by code-based applications where the user (you) enters commands into a plain ASCII text file, which the software then reads and processes to create the final output. Code-based computer applications are by their very nature *extensible*, while WYSIWYG (What You See Is What You Get) applications are generally limited to whatever user interface the developer makes for you.

The `GNU/Linux` computer operating system

> There is so much to be said about Linus Torvalds' `Linux` and Richard Stallman's `GNU` project. First, to credit just these two individuals is to fail to do justice to the *mob* of passionate volunteers who contributed to make this amazing software a reality. I first learned of `Linux` back in 1996, and have been using this operating system on my personal computers almost exclusively since then. It is *free*, it is completely *configurable*, and it permits the continued use of highly efficient `Unix` applications and scripting languages (e.g. shell scripts, Makefiles, `sed`, `awk`) developed over many decades. `Linux` not only provided me with a powerful computing platform, but its open design served to inspire my life's work of creating open-source educational resources.

Bram Moolenaar's `Vim` text editor

> Writing code for any code-based computer application requires a *text editor*, which may be thought of as a word processor strictly limited to outputting plain-ASCII text files. Many good text editors exist, and one's choice of text editor seems to be a deeply personal matter within the programming world. I prefer `Vim` because it operates very similarly to `vi` which is ubiquitous on `Unix`/`Linux` operating systems, and because it may be entirely operated via keyboard (i.e. no mouse required) which makes it fast to use.

Donald Knuth's TEX typesetting system

Developed in the late 1970's and early 1980's by computer scientist extraordinaire Donald Knuth to typeset his multi-volume magnum opus *The Art of Computer Programming*, this software allows the production of formatted text for screen-viewing or paper printing, all by writing plain-text code to describe how the formatted text is supposed to appear. TEX is not just a markup language for documents, but it is also a Turing-complete programming language in and of itself, allowing useful algorithms to be created to control the production of documents. Simply put, TEX *is a programmer's approach to word processing*. Since TEX is controlled by code written in a plain-text file, this means anyone may read that plain-text file to see exactly how the document was created. This openness afforded by the code-based nature of TEX makes it relatively easy to learn how other people have created their own TEX documents. By contrast, examining a beautiful document created in a conventional WYSIWYG word processor such as Microsoft `Word` suggests nothing to the reader about *how* that document was created, or what the user might do to create something similar. As Mr. Knuth himself once quipped, conventional word processing applications should be called WYSIAYG (What You See Is *All* You Get).

Leslie Lamport's LATEX extensions to TEX

Like all true programming languages, TEX is inherently extensible. So, years after the release of TEX to the public, Leslie Lamport decided to create a massive extension allowing easier compilation of book-length documents. The result was LATEX, which is the markup language used to create all ModEL module documents. You could say that TEX is to LATEX as `C` is to `C++`. This means it is permissible to use any and all TEX commands within LATEX source code, and it all still works. Some of the features offered by LATEX that would be challenging to implement in TEX include automatic index and table-of-content creation.

Tim Edwards' `Xcircuit` drafting program

This wonderful program is what I use to create all the schematic diagrams and illustrations (but not photographic images or mathematical plots) throughout the ModEL project. It natively outputs PostScript format which is a true vector graphic format (this is why the images do not pixellate when you zoom in for a closer view), and it is so simple to use that I have never had to read the manual! Object libraries are easy to create for `Xcircuit`, being plain-text files using PostScript programming conventions. Over the years I have collected a large set of object libraries useful for drawing electrical and electronic schematics, pictorial diagrams, and other technical illustrations.

`Gimp` graphic image manipulation program

Essentially an open-source clone of Adobe's `PhotoShop`, I use `Gimp` to resize, crop, and convert file formats for all of the photographic images appearing in the ModEL modules. Although `Gimp` does offer its own scripting language (called `Script-Fu`), I have never had occasion to use it. Thus, my utilization of `Gimp` to merely crop, resize, and convert graphic images is akin to using a sword to slice bread.

`SPICE` circuit simulation program

`SPICE` is to circuit analysis as TEX is to document creation: it is a form of markup language designed to describe a certain object to be processed in plain-ASCII text. When the plain-text "source file" is compiled by the software, it outputs the final result. More modern circuit analysis tools certainly exist, but I prefer `SPICE` for the following reasons: it is *free*, it is *fast*, it is *reliable*, and it is a fantastic tool for *teaching* students of electricity and electronics how to write simple code. I happen to use rather old versions of `SPICE`, version 2g6 being my "go to" application when I only require text-based output. `NGSPICE` (version 26), which is based on Berkeley `SPICE` version 3f5, is used when I require graphical output for such things as time-domain waveforms and Bode plots. In all `SPICE` example netlists I strive to use coding conventions compatible with all `SPICE` versions.

Andrew D. Hwang's `ePiX` mathematical visualization programming library

This amazing project is a `C++` library you may link to any `C/C++` code for the purpose of generating PostScript graphic images of mathematical functions. As a completely free and open-source project, it does all the plotting I would otherwise use a Computer Algebra System (CAS) such as `Mathematica` or `Maple` to do. It should be said that `ePiX` is *not* a Computer Algebra System like `Mathematica` or `Maple`, but merely a mathematical *visualization* tool. In other words, it won't determine integrals for you (you'll have to implement that in your own `C/C++` code!), but it can graph the results, and it does so beautifully. What I really admire about `ePiX` is that it is a `C++` programming library, which means it builds on the existing power and toolset available with that programming language. Mr. Hwang could have probably developed his own stand-alone application for mathematical plotting, but by creating a `C++` library to do the same thing he accomplished something much greater.

`gnuplot` mathematical visualization software

Another open-source tool for mathematical visualization is `gnuplot`. Interestingly, this tool is *not* part of Richard Stallman's GNU project, its name being a coincidence. For this reason the authors prefer "gnu" *not* be capitalized at all to avoid confusion. This is a much "lighter-weight" alternative to a spreadsheet for plotting tabular data, and the fact that it easily outputs directly to an X11 console or a file in a number of different graphical formats (including PostScript) is very helpful. I typically set my `gnuplot` output format to default (X11 on my Linux PC) for quick viewing while I'm developing a visualization, then switch to PostScript file export once the visual is ready to include in the document(s) I'm writing. As with my use of `Gimp` to do rudimentary image editing, my use of `gnuplot` only scratches the surface of its capabilities, but the important points are that it's *free* and that it *works well*.

`Python` programming language

Both Python and C++ find extensive use in these modules as instructional aids and exercises, but I'm listing Python here as a *tool* for myself because I use it almost daily as a *calculator*. If you open a Python interpreter console and type `from math import *` you can type mathematical expressions and have it return results just as you would on a hand calculator. Complex-number (i.e. *phasor*) arithmetic is similarly supported if you include the complex-math library (`from cmath import *`). Examples of this are shown in the Programming References chapter (if included) in each module. Of course, being a fully-featured programming language, Python also supports conditionals, loops, and other structures useful for calculation of quantities. Also, running in a console environment where all entries and returned values show as text in a chronologically-ordered list makes it easy to copy-and-paste those calculations to document exactly how they were performed.

Appendix D

Creative Commons License

Creative Commons Attribution 4.0 International Public License

By exercising the Licensed Rights (defined below), You accept and agree to be bound by the terms and conditions of this Creative Commons Attribution 4.0 International Public License ("Public License"). To the extent this Public License may be interpreted as a contract, You are granted the Licensed Rights in consideration of Your acceptance of these terms and conditions, and the Licensor grants You such rights in consideration of benefits the Licensor receives from making the Licensed Material available under these terms and conditions.

Section 1 – Definitions.

a. **Adapted Material** means material subject to Copyright and Similar Rights that is derived from or based upon the Licensed Material and in which the Licensed Material is translated, altered, arranged, transformed, or otherwise modified in a manner requiring permission under the Copyright and Similar Rights held by the Licensor. For purposes of this Public License, where the Licensed Material is a musical work, performance, or sound recording, Adapted Material is always produced where the Licensed Material is synched in timed relation with a moving image.

b. **Adapter's License** means the license You apply to Your Copyright and Similar Rights in Your contributions to Adapted Material in accordance with the terms and conditions of this Public License.

c. **Copyright and Similar Rights** means copyright and/or similar rights closely related to copyright including, without limitation, performance, broadcast, sound recording, and Sui Generis Database Rights, without regard to how the rights are labeled or categorized. For purposes of this Public License, the rights specified in Section 2(b)(1)-(2) are not Copyright and Similar Rights.

d. **Effective Technological Measures** means those measures that, in the absence of proper authority, may not be circumvented under laws fulfilling obligations under Article 11 of the WIPO Copyright Treaty adopted on December 20, 1996, and/or similar international agreements.

e. **Exceptions and Limitations** means fair use, fair dealing, and/or any other exception or

limitation to Copyright and Similar Rights that applies to Your use of the Licensed Material.

f. **Licensed Material** means the artistic or literary work, database, or other material to which the Licensor applied this Public License.

g. **Licensed Rights** means the rights granted to You subject to the terms and conditions of this Public License, which are limited to all Copyright and Similar Rights that apply to Your use of the Licensed Material and that the Licensor has authority to license.

h. **Licensor** means the individual(s) or entity(ies) granting rights under this Public License.

i. **Share** means to provide material to the public by any means or process that requires permission under the Licensed Rights, such as reproduction, public display, public performance, distribution, dissemination, communication, or importation, and to make material available to the public including in ways that members of the public may access the material from a place and at a time individually chosen by them.

j. **Sui Generis Database Rights** means rights other than copyright resulting from Directive 96/9/EC of the European Parliament and of the Council of 11 March 1996 on the legal protection of databases, as amended and/or succeeded, as well as other essentially equivalent rights anywhere in the world.

k. **You** means the individual or entity exercising the Licensed Rights under this Public License. **Your** has a corresponding meaning.

Section 2 – Scope.

a. License grant.

1. Subject to the terms and conditions of this Public License, the Licensor hereby grants You a worldwide, royalty-free, non-sublicensable, non-exclusive, irrevocable license to exercise the Licensed Rights in the Licensed Material to:

A. reproduce and Share the Licensed Material, in whole or in part; and

B. produce, reproduce, and Share Adapted Material.

2. Exceptions and Limitations. For the avoidance of doubt, where Exceptions and Limitations apply to Your use, this Public License does not apply, and You do not need to comply with its terms and conditions.

3. Term. The term of this Public License is specified in Section 6(a).

4. Media and formats; technical modifications allowed. The Licensor authorizes You to exercise the Licensed Rights in all media and formats whether now known or hereafter created, and to make technical modifications necessary to do so. The Licensor waives and/or agrees not to assert any right or authority to forbid You from making technical modifications necessary to exercise the Licensed Rights, including technical modifications necessary to circumvent Effective Technological Measures.

For purposes of this Public License, simply making modifications authorized by this Section 2(a)(4) never produces Adapted Material.

5. Downstream recipients.

A. Offer from the Licensor – Licensed Material. Every recipient of the Licensed Material automatically receives an offer from the Licensor to exercise the Licensed Rights under the terms and conditions of this Public License.

B. No downstream restrictions. You may not offer or impose any additional or different terms or conditions on, or apply any Effective Technological Measures to, the Licensed Material if doing so restricts exercise of the Licensed Rights by any recipient of the Licensed Material.

6. No endorsement. Nothing in this Public License constitutes or may be construed as permission to assert or imply that You are, or that Your use of the Licensed Material is, connected with, or sponsored, endorsed, or granted official status by, the Licensor or others designated to receive attribution as provided in Section 3(a)(1)(A)(i).

b. Other rights.

1. Moral rights, such as the right of integrity, are not licensed under this Public License, nor are publicity, privacy, and/or other similar personality rights; however, to the extent possible, the Licensor waives and/or agrees not to assert any such rights held by the Licensor to the limited extent necessary to allow You to exercise the Licensed Rights, but not otherwise.

2. Patent and trademark rights are not licensed under this Public License.

3. To the extent possible, the Licensor waives any right to collect royalties from You for the exercise of the Licensed Rights, whether directly or through a collecting society under any voluntary or waivable statutory or compulsory licensing scheme. In all other cases the Licensor expressly reserves any right to collect such royalties.

Section 3 – License Conditions.

Your exercise of the Licensed Rights is expressly made subject to the following conditions.

a. Attribution.

1. If You Share the Licensed Material (including in modified form), You must:

A. retain the following if it is supplied by the Licensor with the Licensed Material:

i. identification of the creator(s) of the Licensed Material and any others designated to receive attribution, in any reasonable manner requested by the Licensor (including by pseudonym if designated);

ii. a copyright notice;

iii. a notice that refers to this Public License;

iv. a notice that refers to the disclaimer of warranties;

v. a URI or hyperlink to the Licensed Material to the extent reasonably practicable;

B. indicate if You modified the Licensed Material and retain an indication of any previous modifications; and

C. indicate the Licensed Material is licensed under this Public License, and include the text of, or the URI or hyperlink to, this Public License.

2. You may satisfy the conditions in Section 3(a)(1) in any reasonable manner based on the medium, means, and context in which You Share the Licensed Material. For example, it may be reasonable to satisfy the conditions by providing a URI or hyperlink to a resource that includes the required information.

3. If requested by the Licensor, You must remove any of the information required by Section 3(a)(1)(A) to the extent reasonably practicable.

4. If You Share Adapted Material You produce, the Adapter's License You apply must not prevent recipients of the Adapted Material from complying with this Public License.

Section 4 – Sui Generis Database Rights.

Where the Licensed Rights include Sui Generis Database Rights that apply to Your use of the Licensed Material:

a. for the avoidance of doubt, Section 2(a)(1) grants You the right to extract, reuse, reproduce, and Share all or a substantial portion of the contents of the database;

b. if You include all or a substantial portion of the database contents in a database in which You have Sui Generis Database Rights, then the database in which You have Sui Generis Database Rights (but not its individual contents) is Adapted Material; and

c. You must comply with the conditions in Section 3(a) if You Share all or a substantial portion of the contents of the database.

For the avoidance of doubt, this Section 4 supplements and does not replace Your obligations under this Public License where the Licensed Rights include other Copyright and Similar Rights.

Section 5 – Disclaimer of Warranties and Limitation of Liability.

a. Unless otherwise separately undertaken by the Licensor, to the extent possible, the Licensor offers the Licensed Material as-is and as-available, and makes no representations or warranties of any kind concerning the Licensed Material, whether express, implied, statutory, or other. This includes, without limitation, warranties of title, merchantability, fitness for a particular purpose, non-infringement, absence of latent or other defects, accuracy, or the presence or absence of errors,

whether or not known or discoverable. Where disclaimers of warranties are not allowed in full or in part, this disclaimer may not apply to You.

b. To the extent possible, in no event will the Licensor be liable to You on any legal theory (including, without limitation, negligence) or otherwise for any direct, special, indirect, incidental, consequential, punitive, exemplary, or other losses, costs, expenses, or damages arising out of this Public License or use of the Licensed Material, even if the Licensor has been advised of the possibility of such losses, costs, expenses, or damages. Where a limitation of liability is not allowed in full or in part, this limitation may not apply to You.

c. The disclaimer of warranties and limitation of liability provided above shall be interpreted in a manner that, to the extent possible, most closely approximates an absolute disclaimer and waiver of all liability.

Section 6 – Term and Termination.

a. This Public License applies for the term of the Copyright and Similar Rights licensed here. However, if You fail to comply with this Public License, then Your rights under this Public License terminate automatically.

b. Where Your right to use the Licensed Material has terminated under Section 6(a), it reinstates:

1. automatically as of the date the violation is cured, provided it is cured within 30 days of Your discovery of the violation; or

2. upon express reinstatement by the Licensor.

For the avoidance of doubt, this Section 6(b) does not affect any right the Licensor may have to seek remedies for Your violations of this Public License.

c. For the avoidance of doubt, the Licensor may also offer the Licensed Material under separate terms or conditions or stop distributing the Licensed Material at any time; however, doing so will not terminate this Public License.

d. Sections 1, 5, 6, 7, and 8 survive termination of this Public License.

Section 7 – Other Terms and Conditions.

a. The Licensor shall not be bound by any additional or different terms or conditions communicated by You unless expressly agreed.

b. Any arrangements, understandings, or agreements regarding the Licensed Material not stated herein are separate from and independent of the terms and conditions of this Public License.

Section 8 – Interpretation.

a. For the avoidance of doubt, this Public License does not, and shall not be interpreted to, reduce, limit, restrict, or impose conditions on any use of the Licensed Material that could lawfully

be made without permission under this Public License.

b. To the extent possible, if any provision of this Public License is deemed unenforceable, it shall be automatically reformed to the minimum extent necessary to make it enforceable. If the provision cannot be reformed, it shall be severed from this Public License without affecting the enforceability of the remaining terms and conditions.

c. No term or condition of this Public License will be waived and no failure to comply consented to unless expressly agreed to by the Licensor.

d. Nothing in this Public License constitutes or may be interpreted as a limitation upon, or waiver of, any privileges and immunities that apply to the Licensor or You, including from the legal processes of any jurisdiction or authority.

Creative Commons is not a party to its public licenses. Notwithstanding, Creative Commons may elect to apply one of its public licenses to material it publishes and in those instances will be considered the "Licensor." Except for the limited purpose of indicating that material is shared under a Creative Commons public license or as otherwise permitted by the Creative Commons policies published at `creativecommons.org/policies`, Creative Commons does not authorize the use of the trademark "Creative Commons" or any other trademark or logo of Creative Commons without its prior written consent including, without limitation, in connection with any unauthorized modifications to any of its public licenses or any other arrangements, understandings, or agreements concerning use of licensed material. For the avoidance of doubt, this paragraph does not form part of the public licenses.

Creative Commons may be contacted at `creativecommons.org`.

Appendix E

References

Dalziel, Charles F., *Deleterious Effects of Electric Shock*, University of California, Berkeley, CA, October 1961.

Hering, Carl, "The Transmission of Power", *Cassier's Magazine*, Volume 1, pp. 449-456, The Cassier Magazine Company, New York, April 1892.

NFPA 70E Standard for Electrical Safety in the Workplace, 2015 Edition, NFPA, Quincy, MA, 2016.

Appendix F

Version history

This is a list showing all significant additions, corrections, and other edits made to this learning module. Each entry is referenced by calendar date in reverse chronological order (newest version first), which appears on the front cover of every learning module for easy reference. Any contributors to this open-source document are listed here as well.

11 March 2026 – typographical errors corrections courtesy of Wyatt Gleason. One error also identified by Wyatt was showing 746:1 HP to W rather than W to HP!!

8-9 January 2025 – added a Tutorial section on harmonics within three-phase AC circuits, and also corrected some minor typographical errors.

27 September 2024 – divided the Introduction chapter into sections, one with recommendations for students, one with a listing of challenging concepts, and one with recommendations for instructors.

23 October 2023 – added a Case Tutorial section showing an RC phase-rotation testing network.

12 October 2023 – typo correction ("wye-wound versus delta-wound" instead of "wye-wound versus delta-would") courtesy of David Mitchell.

21 June 2023 – added Quantitative Reasoning question related to computer programming (simulating three-phase circuits).

8-9 March 2023 – divided the Tutorial into sections, and elaborated on three-phase rectifier circuits. Also edited some instructor notes.

28 November 2022 – placed questions at the top of the itemized list in the Introduction chapter prompting students to devise experiments related to the tutorial content.

8 May 2021 – commented out or deleted empty chapters.

23 April 2021 – added a new Case Tutorial section showing an inexpensive motor-generator set for producing low-voltage three-phase AC.

18 March 2021 – corrected multiple instances of "volts" that should have been capitalized "Volts".

11 March 2021 – added phase rotation note to illustrations in Tutorial, and also completed the quote from Hering's article.

9 March 2021 – added a Case Tutorial chapter with three-phase circuit examples, and also made some additions to instructor notes.

10 November 2020 – minor edits to Tutorial.

14 October 2020 – significantly edited the Introduction chapter to make it more suitable as a pre-study guide and to provide cues useful to instructors leading "inverted" teaching sessions.

20 April 2020 – added clarification that grounding is done only at the source, and never at the load, in a power circuit. Also, capitalized all instances of the word "Earth".

13-14 April 2020 – added animation of the rotating magnetic field for a three-phase AC induction motor, as well as some minor edits to graphics in one of the Conceptual Reasoning problems.

5 January 2020 – added bullet-list of relevant programming principles to the Programming References section.

4 January 2020 – added Programming References chapter, with section on simulating Wye/Delta circuits.

17 December 2019 – added footnote defining common-mode voltage.

25 April 2019 – edited Tutorial to reference Kirchhoff's Laws more completely.

30 March 2019 – added more questions, and one experiment.

28 March 2019 – added questions, as well as a comment about Watts-Horsepower conversion to the Tutorial.

10 March 2019 – completed the Foundational Concepts list.

1 January 2019 – added a Technical Reference section on electrical safety.

5 November 2018 – retitled Historical References section(s) so as to not be redundant to the "Historical References" chapter.

September 2018 – separated "Derivations and Technical References" chapter into two chapters, one for technical references and the other for "Historical References".

August 2018 – corrected error in citation of Carl Hering's article on an early three-phase electric power system. I originally described this system as being at Niagara Falls, when it was actually in Frankfort (Germany). Also, corrected References chapter, which was not showing up as its own chapter (from being commented out). Finally, added content to the Introduction.

July 2018 – document first created.

Index

AC induction motor, 25
Adding quantities to a qualitative problem, 146
Alternating current, 52
Annotating diagrams, 145
Arc, 54
Arc blast, 58
Arc flash, 54
Arc flash boundary, 55
Arc flash suit, 57

Bastard voltage, 31
Bonding, 53
Boundary, arc flash, 55
Boundary, limited approach, 55
Boundary, restricted approach, 55
Breadboard, solderless, 132, 133
Breadboard, traditional, 136
Busbar, 53

C++, 86
Calorie, 57
Cardio-Pulmonary Resuscitation, 130
Checking for exceptions, 146
Checking your work, 146
Circuit breaker, 52
Code, computer, 155
Common-mode voltage, 29
Compiler, C++, 86
Computer programming, 85
Conservation of Energy, 27
CPR, 130

Dalziel, Charles, 51, 130
Delta, 21
Dimensional analysis, 145
DIN rail, 133
Diode, 33
DIP, 132
Direct current, 52
Disconnect switch, 53

Eddy current, 38
Edwards, Tim, 156
Electric shock, 51, 130
Electrically common points, 53, 131
Enclosure, electrical, 136
Energy transfer, 25
Equipotential, 53
Equipotential points, 131, 133
Experiment, 137
Experimental guidelines, 138

Filter, 40
Fourier's Theorem, 34
Frequency, 52
Fuse, 52

Graph values to solve a problem, 146
Greenleaf, Cynthia, 99
Ground loop, 31

Harmonic frequency, 34
Hertz, 52
Hot, 29
How to teach with these modules, 153
Hwang, Andrew D., 157

IC, 132
Identify given data, 145
Identify relevant principles, 145
Induction motor, 25
Intermediate results, 145
Interpreter, Python, 90
Inverted instruction, 153
Ionized air, 54

Java, 87

Kirchhoff's Current Law, 24
Kirchhoff's Voltage Law, 24
Knuth, Donald, 156

Lamport, Leslie, 156
Law of sines, 48
Limited approach boundary, 55
Limiting cases, 146
Line current, 23
Line voltage, 23
Linear component, 34
Load, 51
Lock-out, tag-out, 55
LOTO, 55

Maxwell, James Clerk, 43
Metacognition, 104
Modulo, 37, 39
Moolenaar, Bram, 155
Motor, 25
Murphy, Lynn, 99

National Fire Protection Association, 55
Nervous system, 51
Neutral, 29
NFPA, 55
NFPA 70E, 55
Nonlinear component, 34

One-Hand Rule, 54
Open-source, 155

Phase current, 23
Phase rotation, 19
Phase sequence, 19
Phase voltage, 23
Phasor diagram, 48
Polyphase, 18
Polyphase versus single-phase, 25
Potential distribution, 133
Problem-solving: annotate diagrams, 145
Problem-solving: check for exceptions, 146
Problem-solving: checking work, 146
Problem-solving: dimensional analysis, 145
Problem-solving: graph values, 146
Problem-solving: identify given data, 145
Problem-solving: identify relevant principles, 145
Problem-solving: interpret intermediate results, 145
Problem-solving: limiting cases, 146
Problem-solving: qualitative to quantitative, 146
Problem-solving: quantitative to qualitative, 146
Problem-solving: reductio ad absurdum, 146
Problem-solving: simplify the system, 145
Problem-solving: thought experiment, 138, 145
Problem-solving: track units of measurement, 145
Problem-solving: visually represent the system, 145
Problem-solving: work in reverse, 146
Programming, computer, 85
Project management guidelines, 141
Python, 90

Qualitatively approaching a quantitative problem, 146

Reading Apprenticeship, 99
Reductio ad absurdum, 146, 152, 153
Restricted approach boundary, 55

Safety disconnect switch, 53
Safety, electrical, 130
Schoenbach, Ruth, 99
Scientific method, 104, 137
Scope creep, 141
Semiconductor, 33
Shock, electric, 51
Shunt resistor, 132
Simplifying a system, 145
Single-phase versus polyphase, 25
Skin effect, 38
Socrates, 152
Socratic dialogue, 153
Solderless breadboard, 132, 133
Source code, 86
Spark, 54
SPICE, 99, 138
SPICE netlist, 134
Stallman, Richard, 155
Star, 21

Stator winding, 18
Subpanel, 136
Surface mount, 133
Switch, disconnect, 53

Terminal block, 131–134, 136
Thought experiment, 138, 145
Torvalds, Linus, 155
Transformer, 22
Triplen harmonics, 36

Units of measurement, 145

Visualizing a system, 145
Voltage, common-mode, 29

Whitespace, C++, 86, 87
Whitespace, Python, 93
Wiring sequence, 135
Work in reverse to solve a problem, 146
Wye, 21
WYSIWYG, 155, 156