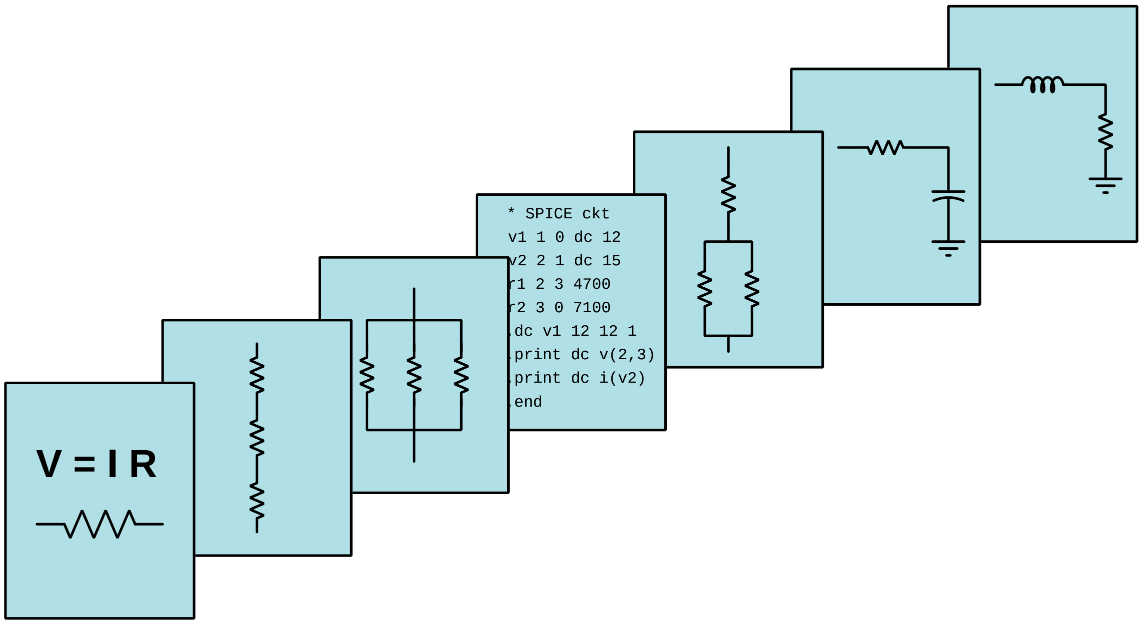
Modular Electronics Learning (ModEL) project



Vacuum Tubes

Last update = 12 July 2026

Contents

Chapter 1

Introduction

Chapter 2

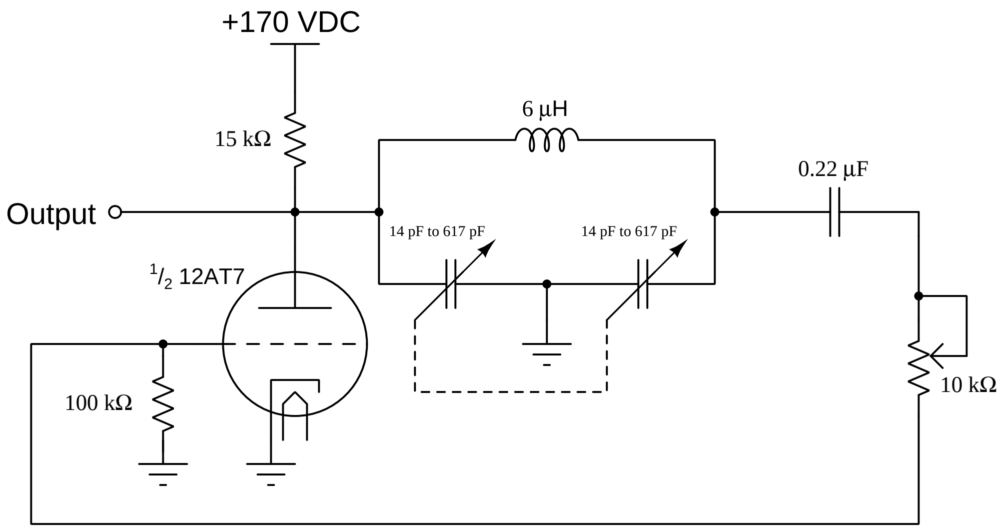
Case Tutorial

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: adjustable-frequency triode Colpitts oscillator

The following Colpitts oscillator circuit utilizes a model 12AT7 dual-triode vacuum tube as its active element. The LC network consists of a single inductor and two variable capacitors with a grounded center-point between the two capacitors. Only one of the two triodes contained within the glass vacuum envelope is being used for this circuit. The +170 VDC and 12 Volt filament power supply circuitry is omitted for simplicity:



The inductor is formed of 14 turns of 18 AWG enameled wire wound with a diameter of 2.25 inches over a length of 3 inches, yielding approximately 6 microHenrys of inductance. The ganged variable capacitors each have a range of 14 picoFarads to 617 picoFarads, salvaged from an old Hewlett-Packard model 651A test oscillator.

Given these component values, the low and high resonant frequencies may be predicted as follows. Note that because the capacitors are essentially connected in series with each other (neglecting any current passing through the ground connection between them), the total capacitance for the LC feedback network ranging from 7 picoFarads to 308.5 picoFarads, just one-half that of each capacitor individually, since capacitance diminishes in series:

$$f_{low} = \frac{1}{2\pi\sqrt{LC}} = \frac{1}{2\pi\sqrt{(6\ \mu\text{H})(308.5\ \text{pF})}} = 3.699\ \text{MHz}$$

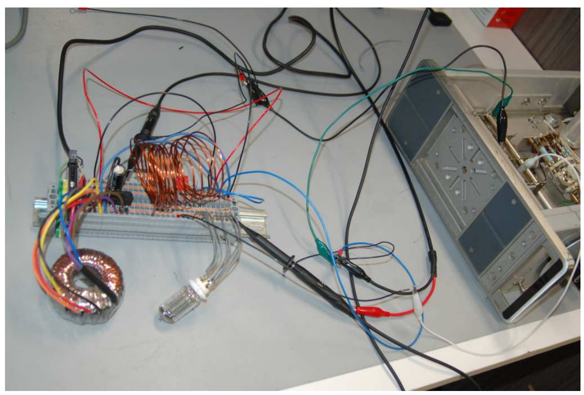
$$f_{high} = \frac{1}{2\pi\sqrt{LC}} = \frac{1}{2\pi\sqrt{(6\ \mu\text{H})(7\ \text{pF})}} = 24.56\ \text{MHz}$$

Actual measurements resulted in significantly lower frequency values than predicted, the low being 2.710 MHz and the high being only 8.130 MHz, owing to parasitic capacitances spread throughout the prototyped circuit.

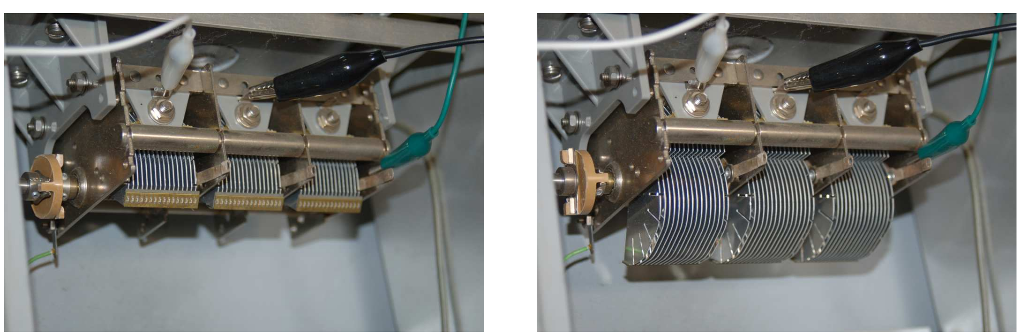
The 10 kΩ potentiometer wired as a rheostat provides adjustable feedback to maintain good oscillations. With too much feedback the waveform suffers from some distortion (strong 2nd and

3rd harmonics), and with too little feedback the oscillations die out completely. The 0.22 microFarad capacitor simply couples the AC portion of the LC network's signal around to the feedback network while blocking any DC bias, the 100 kiloOhm resistor centering the fed-back signal around ground potential so that the tube's grid electrode sees a potential swinging symmetrically positive and negative with respect to ground. Triode vacuum tubes are functionally similar to N-channel depletion-type MOSFETs, in that their unstimulated state is partially on, with a positive grid potential increasing plate current and a negative grid potential decreasing plate current.

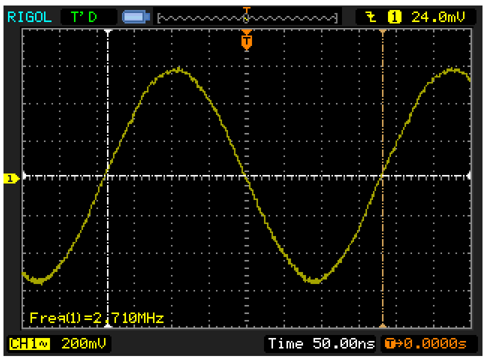
A set of DIN-rail-mounted terminal blocks served to interconnect all components, rather than using a solderless breadboard. The 170 Volt DC power supply was built using a 1:1 transformer to take 120 VAC (RMS) and rectify and filter it to DC. A 12 VAC winding on that same transformer provides power for both triodes' filaments (wired in series, 6 Volts AC per filament). The legacy Hewlett-Packard model 651A test oscillator is seen on the far right of the photograph, its variable capacitor disconnected from the original factory circuitry for use in this test:

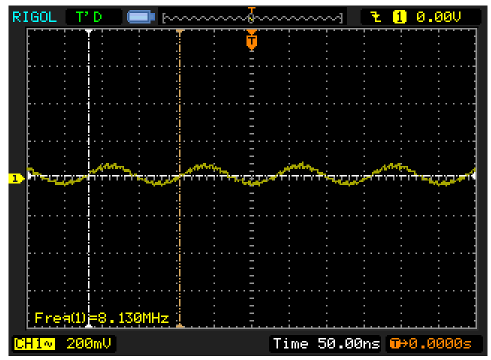


The variable capacitor assembly inside the legacy HP-651A actually consists of *three* ganged together on a common shaft, but only two are being used for this circuit. The metal frame of this assembly is electrically common to the shaft for all three capacitors, and is connected to the circuit's ground. The twin photographs below show near-maximum capacitance on left and near-minimum on right:



With the rheostat set at a constant 1 kΩ we see the peak-to-peak amplitude[1] of the oscillation vary considerably from lowest frequency to highest frequency:





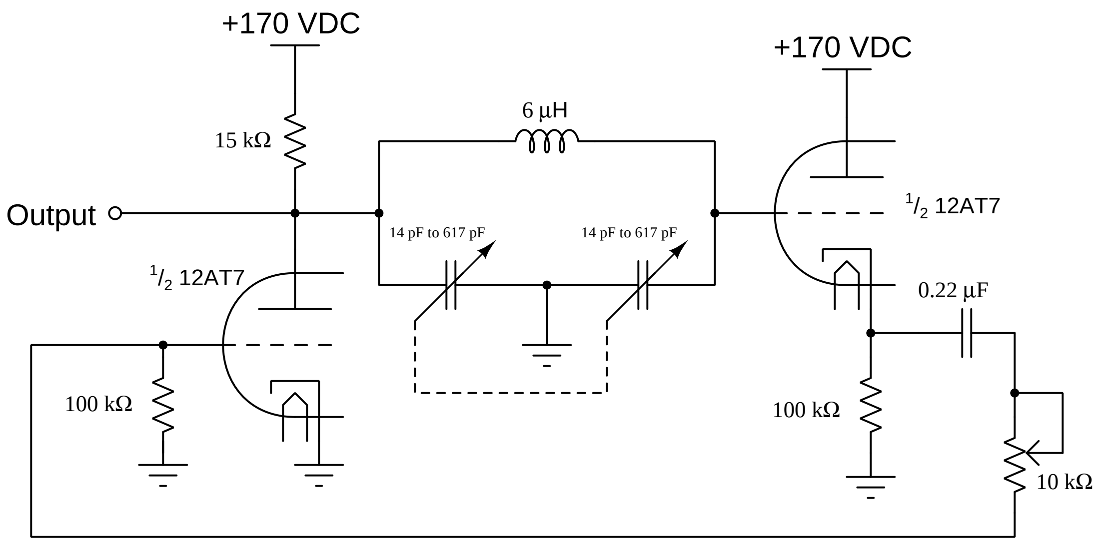
This variance in amplitude is the result of the LC resonant network impedance varying with frequency. With more capacitance (greater overlapping area between plates) to create lower-frequency oscillations, the inductor's reactance at resonance will be lower ($X_L = 2\pi f L$), and since the capacitors' combined reactance must equal X_L at resonance it means the entire LC network has a lower impedance. Decreased impedance in the LC network results in a stronger signal fed back to the triode tube's grid electrode and therefore a stronger output signal. Moving the capacitor plates away from each other to create less capacitance and thereby raise resonant frequency means both X_L and X_C increase, resulting in greater LC network impedance overall and less signal fed back to the triode for amplification, resulting in diminished signal amplitude.

If a Colpitts oscillator is designed to operate at just one frequency (i.e. fixed inductance and fixed capacitance) then such variations are irrelevant. However, when designing a Colpitts oscillator to have a wide frequency range one must anticipate having to adjust feedback gain (attenuation) to maintain the desired amplitude while adjusting frequency.

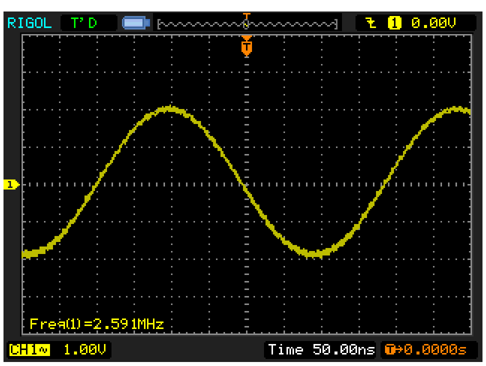
Careful inspection of the sinusoidal plots at low frequency versus high frequency also reveal a "cleaner" waveform at higher frequency. Note how symmetrical the positive and negative peaks of the waveform appear at 8.130 MHz, versus the 2.710 MHz waveform which has "fatter" positive peaks versus "narrower" negative peaks. The greater asymmetry of the lower-frequency waveform betrays even-numbered harmonic content. The "cleaner" (i.e. less harmonic content) of the higher-frequency waveform is due partly to the lower-amplitude signal keeping the triode tube in a more linear range of its operation, as well as the LC network having a greater quality factor (Q) value when both X_L and X_C are greater – the greater Q factor making it have a steeper rolloff rate as a band-pass filter.

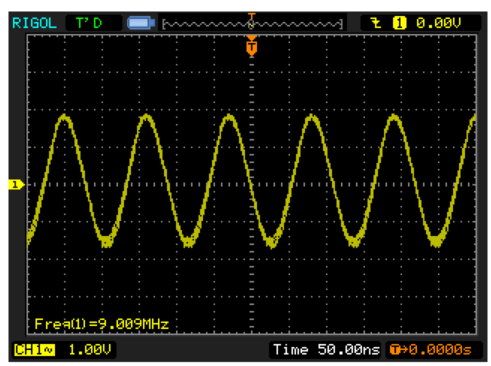
[1]In these oscilloscope measurements a 10:1 probe was used while the oscilloscope's channel setting was still set for 1:1, so the actual voltage is ten times greater than the vertical sensitivity would suggest. For example, in the left-hand oscillograph the sinusoidal waveform has a peak voltage of nearly 6 Volts, not 0.6 Volts as it might appear.

An improved version of this oscillator circuit appears below, making use of the second triode inside the same 12AT7 vacuum tube. This second triode is connected as a "voltage follower" between the LC network and the resistor-capacitor biasing network for the amplifying triode. The voltage-follower stage presents a more constant impedance to the RC biasing network and to the amplifying triode, resulting in a more stable oscillation magnitude over the full frequency range:



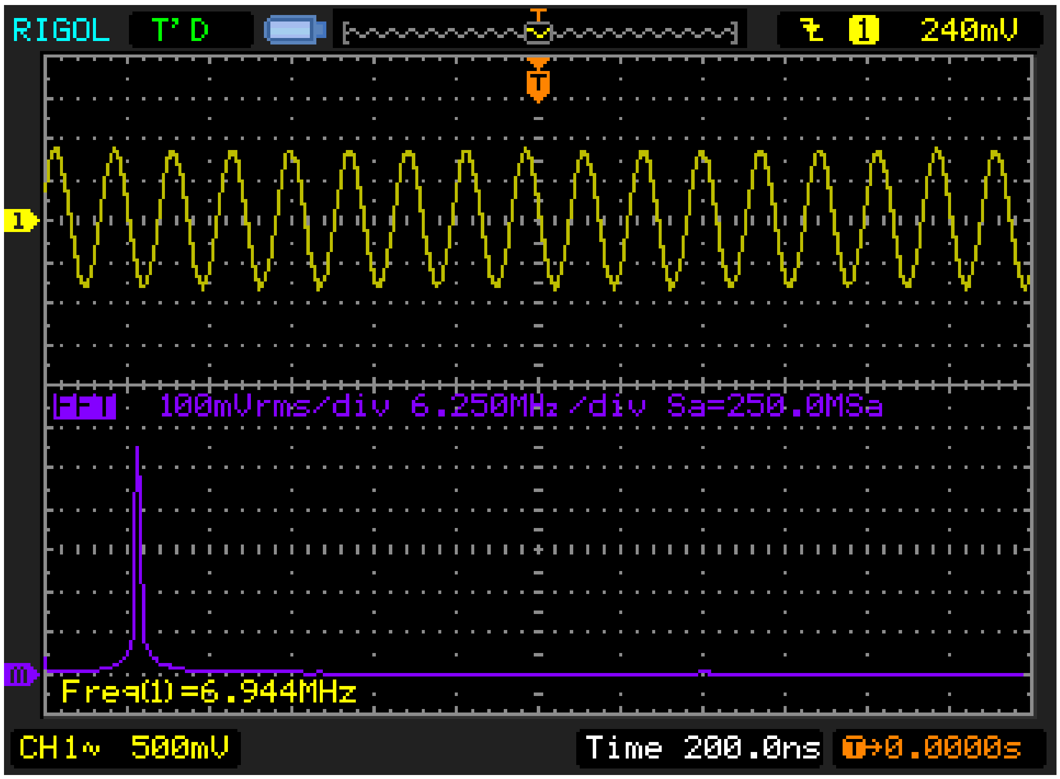
With the rheostat still set at a constant 1 kΩ we now see the peak-to-peak amplitude[2] of the oscillation vary much less than it did before from lowest frequency to highest frequency, and a significantly greater signal voltage in all cases (approximately 40 Volts peak-to-peak versus only about 6 Volts peak-to-peak in the former circuit):





[2]Again, I was using a 10:1 probe with the oscilloscope set for 1:1 which makes the amplitude look ten times smaller than it really is. Here, both waveforms are nearly 40 Volts peak-to-peak.

Adjusting the rheostat[3] to achieve the lowest harmonic content in the oscillation waveform yields extremely clean results:



Another interesting thing I learned about this oscillator is that it continues to perform quite well operating at reduced AC line voltage. Using a Variac to experimentally reduce AC line voltage from 120 Volts RMS to approximately 80 Volts RMS resulted in the tube filaments receiving 9 Volts AC RMS total (i.e. just 4.5 Volts RMS each in series) and the plate voltage being 107 Volts DC, yet the oscillations continued to be clean and robust. Operating at such reduced power levels means the tube runs at cooler temperatures than usual, potentially extending its operating life. At these operating parameters the tube could be fully grasped by the hand without discomfort from high temperature.

Frequency was quite stable as supply voltage varied, too. Unlike MOSFETs whose internal capacitances tend to vary with applied voltage, the inter-electrode capacitances of vacuum tubes are a function of electrode spacing which is more or less constant[4] and unaffected by supply voltage.

[3]For this measurement the rheostat was set to approximately 7 kΩ, resulting in a signal with less harmonic content but also smaller in peak-to-peak amplitude (about 16 Volts peak-to-peak instead of 40 Volts peak-to-peak).

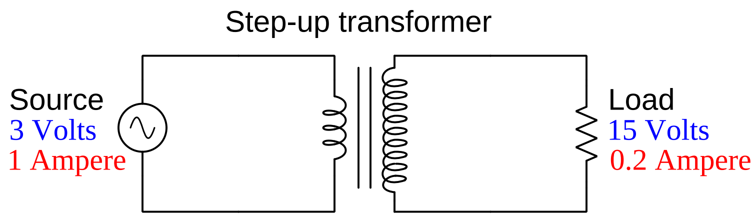
[4]These physical dimensions will change slightly as the tube warms from ambient to operating temperature, but such changes are very gradual which means for a manually-adjusted oscillator such as this any such drift may be easily corrected simply by turning the variable capacitor knob slightly.

Chapter 3

Tutorial

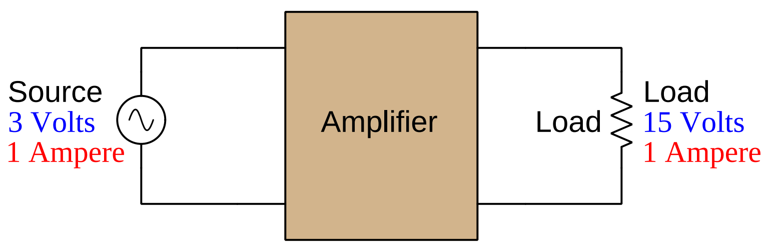
3.1 Amplifier concepts

An *amplifier* is any device allowing a small amount of energy to exert control over a larger amount of energy. It is easy to misunderstand this concept, and so we will begin by exploring what amplification is *not*. Consider a step-up transformer, where an AC source energizes a primary winding, producing a larger voltage at the secondary winding to energize a load:



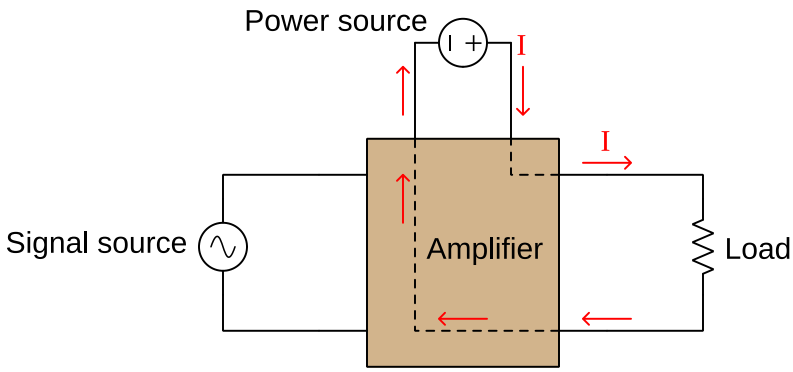
This transformer boosts voltage by a factor of five, but this magnification of voltage is *not* amplification because it comes at the expense of an equivalent reduction in current. Like a mechanical lever used to multiply either force or motion, multiplication of one always leads to division of the other by the same ratio so that energy in equals energy out (in this case, 3 Watts in = 3 Watts out).

An amplifier, then, must be capable of outputting more energy than it receives from its signal source. For example, a circuit boosting voltage while outputting the same current as it inputs would be a true amplifier:



How is this possible, though, without violating the Law of Energy Conservation?

The answer to this very good question is that amplifiers must *also* connect to an additional energy source providing the greater electrical power delivered to the load. The input signal merely controls the flow of electrical energy from source to load. Another way of stating this fact is to say that amplification is a form of *controlled conductivity*, where the input signal determines the degree of conductivity for some other energy source's output to flow to a load:



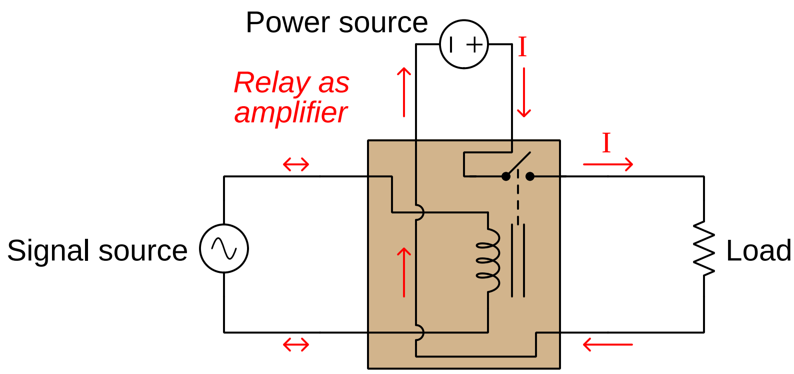
By the same definition, a lever used to magnify force is *not* an amplifying device because you never get more mechanical work out of the lever than what you put into it. However, a motor-driven crane is an amplifying device because a small amount of force and motion applied by a human operator to pushbuttons or levers on the crane *controls the flow of energy from the crane's motor to the load* to lift heavy objects over large distances. The specific component within the crane enabling the operator's weak hand motions to direct many kilowatts' worth of mechanical power to move the load could be a hydraulic valve, a clutch, or some other energy-controlling device. Its specific identity is not important, but its function is essential.

Terms used to distinguish energy-*controlling* devices from other devices, at least in the world of electric circuits, are *active* versus *passive*. Active devices are what make true amplification possible. Components such as resistors, inductors, capacitors, diodes, switches, fuses, and batteries are all passive: they may be sources, they may be loads, and they may alternate roles, or they may be neither, but they do not permit one electrical signal to *control conductivity* for another.

An important way to quantify amplification in any of its forms by a term called *gain*. Simply defined, gain is the ratio of controlled versus controlling quantities, usually symbolized by the variable A. As such, the value of A tells us how many times larger an amplifier's output signal is compared to its input signal. For example, the *power gain* of an amplifier is the ratio of its output signal power divided by the controlling input signal power: $A_P = \frac{P_{out}}{P_{in}}$. The voltage gain of an electronic amplifier is the ratio of output signal voltage to input signal voltage: $A_V = \frac{V_{out}}{V_{in}}$. The current gain of an electronic amplifier is the ratio of output signal current to input signal current: $A_I = \frac{I_{out}}{I_{in}}$. Similarly, we could rate the gain of a motorized crane in terms of force (i.e. the amount of lifting force output divided by the amount of force necessary to push the control lever): $A_F = \frac{F_{lift}}{F_{lever}}$. In every case, gain is a unitless quantity because it is always a ratio between quantities having identical units of measurement (e.g. Watts versus Watts, Volts versus Volts, etc.).

The first active circuit component was the *electromechanical relay* – a combination of solenoid coil and electrical switch. When a power source energizes the solenoid coil, its magnetic field attracts an iron armature which in turn actuates the switch. Relays may be considered active components because the amount of energy conveyed by the switch contact may be much greater than the amount of energy required to energize the coil.

An amplification circuit using a relay is shown in the following schematic diagram:



All energy delivered to the load originates from the power source, not the signal source. All the signal does is energize the relay's coil, to *control* energy sent to the load from the power source. The relay's switch contact then closes to allow the flow of electrical energy to the load, at whatever voltage and current levels the contact is able to handle, which is typically *thousands of times greater*[1] than the energy required to energize the coil.

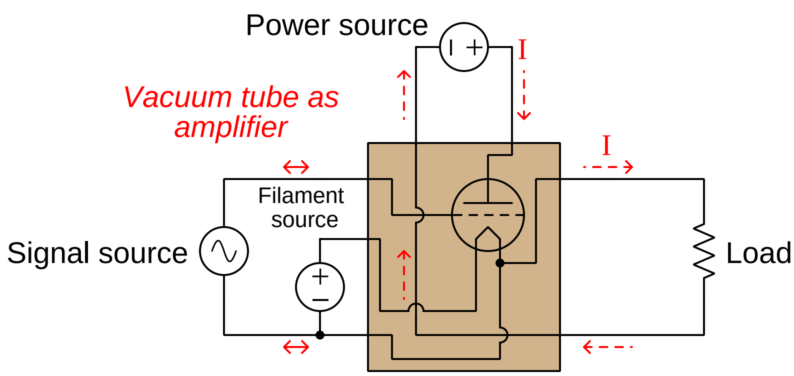
Electromechanical relays revolutionized electric circuits by enabling them to perform functions previously unimaginable. One of these functions is the ability of a fragile switch to exert control over a powerful load: the switch only needs to handle as much current when closed and voltage when open as is necessary to energize the relay coil, while the relay's much larger switch contacts handle the voltage and current requirements of the powerful load. *Telegraph* systems were one early application of relays, where they were used as "repeaters" to boost the power of a telegraph signal weakened by many kilometers of wire resistance, thereby extending the practical range of wired telegraphy. Another function permitted by relays is *automation*, where electrical systems become self-controlling. This is only possible where one electrical signal has the ability to control another, which is the very essence of relay function.

While electromechanical relays made possible such technologies as long-distance telegraph networks, automatic motor control systems, automatic voltage regulators for generators, automatic protection systems for electrical power networks, and even the first digital electric computers, relays do have some significant limitations. Relay contacts are either open or closed, and as such they cannot achieve any states in-between those two. As such, relays are helpless to amplify anything but discrete on-off signals. Amplification of other signal types, such as *analog* signals where information is encoded in the variable magnitude of the signal, is beyond the reach of electromechanical relays.

[1]Thus, we could say that a relay's *energy gain* is a unitless value ranging in the thousands.

The next innovation in active devices was the *vacuum tube* or *electron tube*, originally known as the *thermionic valve*. This was a derivative of the incandescent electric lamp, exploiting the fact that a glowing-hot metal surface emits free electrons into the space around it. Consisting of a glass envelope evacuated of air and containing a filament along with one or more metal grids and plates, vacuum tubes used the application of a small voltage between the grid and filament to control the flow of electrons passing from[2] the filament to the plate.

An amplification circuit using a vacuum tube is shown in the following schematic diagram:



Vacuum tubes once again revolutionized electric circuits by enabling them to perform functions previously unimaginable. In fact, this is the innovation that ushered in the era of *electronic* circuits, where the flow of electrons could be precisely controlled by a voltage signal. Unlike relays which are discrete in nature, vacuum tubes are capable of analog operation, gradually restricting the flow of electrons from full rated current to zero, and any level in-between. Vacuum tubes were also much *faster* than relays, allowing a time-varying signal to reproduce its same time-variance at the output. Note in the diagram shown above how the AC signal causes the DC power source to apply a *pulsed* DC signal to the load. It is as though the controlling signal causes an AC-like waveform to be "carved" out of the steady DC supply voltage.

The ability of a vacuum tube to amplify AC signals of arbitrary wave-shape led to the development of new forms of communications including *telephone*, *radio*, and *television*. Prior to vacuum tubes, the only practical form of long-distance electrical communication was the *telegraph*, using discrete on/off states to digitally represent letter and number characters. Vacuum tube electronic circuits also made possible *radar* to detect objects far out of human eyesight, and played a significant role in World War Two. Another significant military application of vacuum tubes was automatic gun direction systems, which when coupled to radar systems made possible automatic defense from aerial attacks.

For all their capabilities, vacuum tubes had significant limitations. Their requirement of a glowing-hot metal filament made them fairly energy-inefficient. They were physically fragile as well, owing to their glass envelope construction. Also, the high-vacuum condition required inside the glass envelope to permit electrons to freely flow between filament and plate was difficult to maintain. Over

[2]Vacuum tubes are one of the few applications where the *electron flow* convention of denoting current makes more sense than *conventional flow*. In the diagram shown here, the red current arrow still points in the direction of conventional flow, even though it is electrons flowing from the "pointed" filament structure to the flat plate structure inside the tube that comprises load current through the tube.

time, ambient air would leak inside the tube and render it useless. Finally, they were limited in current to milliAmperes, requiring high voltage to amplify signals of significant power.

These limitations spurred the invention of active devices based on electricity conducting through solid materials rather than through a vacuum. The most promising of these solid materials was a class of substances called *semiconductors*. A pure "semiconductor" may be defined as a material whose charge carriers inhabit energy states very nearly, but not quite, permitting charge carrier motion within the substance. This stands in contrast to conductors possessing an abundance of free charge carriers, and insulators where the charge carriers are fairly "locked" into place by a lack of available energy states.

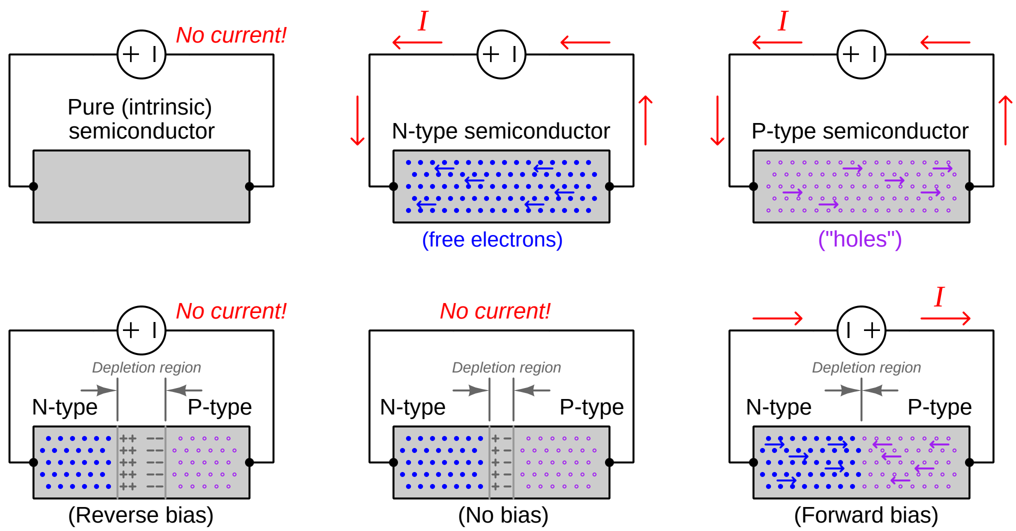
It is not as though a semiconductor is just a poor conductor, either. A conductor of poor quality contains mobile charge carriers, but its charge carriers collide with stationary atoms as they attempt to drift through the atomic lattice of that substance. These collisions convert the charge carriers' kinetic energy into heat, thus constituting electrical resistance. A semiconductor in its pure state, by contrast, contains no free charge carriers at room temperature – i.e. a pure semiconductor is an insulator – but just the right stimulus will *liberate* charge carriers, setting them free to drift through the material's volume. In other words, the phenomenon of *controllable conductivity* is what makes a semiconductor "semi"-conductive.

In a sense, the internal condition of a pure semiconductor is analogous to the internal condition of a vacuum tube: when cool, no free charge carriers exist within the vacuum space of the tube. When stimulated (e.g. heated), however, charge carriers jump off the heated filament and enter the vacuum where they are free to drift, a phenomenon known as *thermionic emission*. In fact, heat is one of the stimuli known to cause semiconductor materials to transition from insulator to conductor, just like a vacuum tube. Light[3] also liberates charge carriers within a pure semiconducting substance.

Through many years of research and experimentation it was discovered that electrical conductivity through a semiconductor could be controlled by carefully alloying ("doping") the semiconducting material with certain types of impurities and applying voltage across junctions of differently-doped semiconductor. The basis of this phenomenon is something called a *depletion region* that forms between differently-doped layers of semiconducting material. Semiconductors doped with impurities placing electrons at energy levels within reach of free motion within the material are called *N-type*. Semiconductors doped with impurities creating vacancies (called *holes*) for low-energy electrons to "hop" into and out of these vacancies from atom to atom are called *P-type*. Both N-type and P-type semiconductors are electrically conductive at room temperature and behave as resistances. Whenever an N-type and a P-type material meet, though, free electrons from the N-type rush to fill vacancies in the P-type, leaving a narrow region at the junction depleted of free charge carriers (i.e. insulating). This so-called *depletion region* may be expanded or collapsed by an applied voltage, making its conduction controllable by a small electrical signal: the key to amplification.

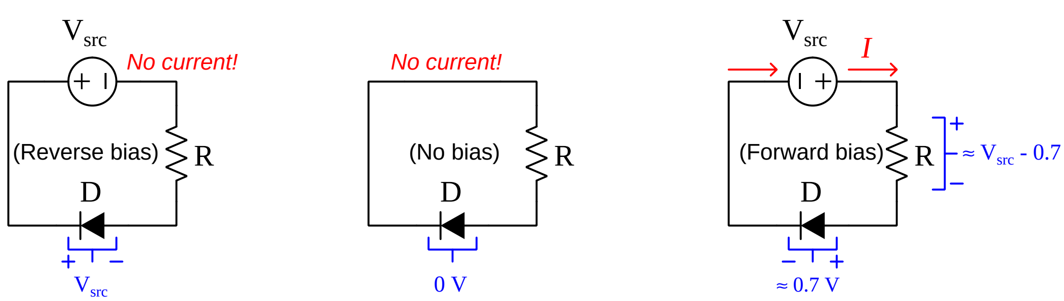
[3]Interestingly, light can also do the same in a vacuum tube, and this is known as the *photoelectric effect*. Not all colors of light will do this, but only those photonic frequencies high enough to meet the *work function* criterion, which is essentially the "escape velocity" energy needed to boost an electron free of a metal surface and into a surrounding vacuum.

A series of illustrations shows how electrical conduction through semiconductors works:



Any section of pure semiconductor, in the absence of heat and light, essentially acts as an insulator and conducts no current. Impurities added to the semiconductor making it either N-type or P-type permits charge carrier motion and thus the semiconductor becomes a conductor. At the junction point of N-type and P-type semiconductors a depletion region forms that is devoid of mobile charge carriers, much like a pure semiconductor, forming a narrow insulating region. However, this depletion region is elastic, and may be expanded or collapsed by the application of an external "bias" voltage.

A single junction formed of N-type and P-type semiconductor alloys (called a *PN junction*) is the basis for solid-state *diodes*. When unbiased, a diode is electrically "open", and remains that way until forward-biased with sufficient voltage. The amount of voltage required to fully collapse the depletion region and make it conductive once again is fairly small, approximately 0.5 to 0.7 Volts for a PN junction made of silicon. It is for this reason that approximate calculations made in semiconductor circuits usually assume a 0.5 to 0.7 Volt forward voltage drop in a conducting diode:



Just like the relay and the vacuum tube, the advent of "solid state" semiconductor electronic devices made possible technologies that were previously unimaginable. Solid-state diodes made conversion of AC power to DC simple, reliable, and power-efficient. *Transistors* which are the solid-state equivalent of vacuum tube amplifiers made possible portable radios and televisions as well as the computer technology you are no doubt using to read this very document!

3.2 Pros and cons of tubes

Vacuum tubes are all but an obsolete technology at the time of this writing (2026), so why bother studying their theory or application? The simple reason is that *obsolete technologies still work, and for some niche applications may still be optimal for the task.* Before diving into the details of vacuum tube technology, it is worthwhile contrasting some of the unique advantages vacuum tubes still enjoy over modern semiconductor devices:

- **Ease of manufacture** – all that is necessary to manufacture a vacuum tube is glass and suitable glass-blowing equipment, sheet metal and wires to make the tube's internal elements, and appropriate vacuum pumps for evacuating the glass envelopes of air. No exotic materials are necessary, nor is a clean room required as is the case with modern semiconductor manufacturing.

- **Thermal ruggedness** – vacuum tubes are able to operate at temperatures far in excess of the maximum allowed for silicon-based semiconductor devices, as they are nothing more than glass and metal. Excessive temperature may melt the internal metal components, fracture the glass envelope, and/or disrupt the metal-to-glass seals necessary to maintain a state of high vacuum inside the tube, but semiconductors stop working at temperatures far below these. The generally high operating temperature of vacuum tubes also makes them able to shed heat at higher rates with less surface area for cooling than semiconductors, the latter generally requiring large metal *heat sinks* drawing thermal energy away from the devices.

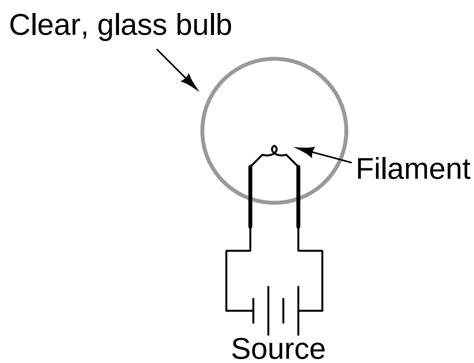
- **Electrical ruggedness** – vacuum tubes are entirely immune to damage from static electricity, from electromagnetic pulses, and other electrical stressors. Tubes have high overload capacities, able to tolerate over-current conditions for longer than most semiconductors owing to their higher operating temperatures and greater thermal mass.

- **Radiation ruggedness** – nuclear radiation has little to no effect on a vacuum tube, unlike semiconductors whose crystal lattice structures are damaged by radiation exposure. This characteristic is useful in close proximity to nuclear reactions, and also for travel through outer space where no atmosphere or planetary magnetic field shields the spacecraft from cosmic radiation.

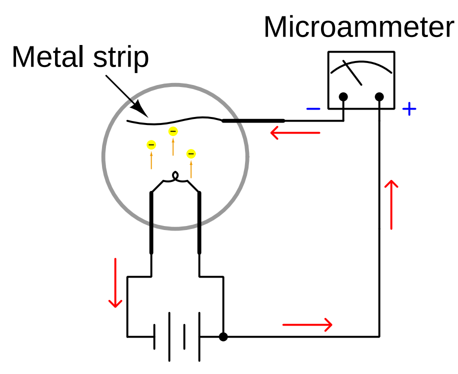
- **Design versatility** – a great multitude of possibilities exist for the internal arrangement of metallic elements in a vacuum tube, making a very wide variety of tube types and capabilities possible.

3.3 The Edison Effect

The prolific American inventor Thomas Edison is often credited with the invention of the incandescent lamp. More accurately, it could be said that Edison was the man who *perfected* the incandescent lamp. Edison's successful design of 1879 was actually preceded 77 years by the British scientist Sir Humphry Davy, who was the first to demonstrate the principle of using electric current to heat a thin strip of metal (called a "filament") to the point of incandescence (glowing white hot). Edison's success building a practical incandescent lamp was due to placing the carbonized-thread filament inside a clear glass bulb from which the air had been removed. In this vacuum, the filament could glow at white-hot temperatures without being consumed by combustion:



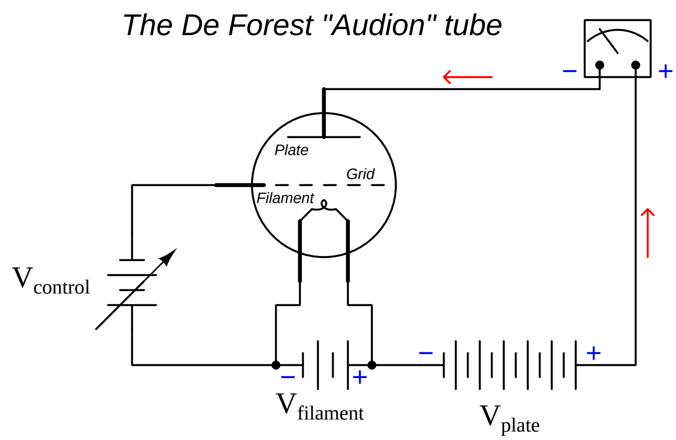
In the course of his experimentation (sometime around 1883), Edison placed a strip of metal inside of an evacuated (vacuum) glass bulb along with the filament. Between this metal strip and one of the filament connections he attached a sensitive ammeter. What he found was that electrons would flow through the meter whenever the filament was hot, but ceased when the filament cooled down:



Note how the electrons move inside the tube from the heated filament across the vacuum space to the metal strip, opposite the direction of the conventional-flow-notation current arrows.

3.4 Triode vacuum tubes

In 1906 the American inventor Lee De Forest started experimenting with the "Edison Effect" to see what more could be gained from the phenomenon. In doing so, he made a startling discovery: by placing a metal grid or screen between the glowing filament and the metal strip (which by now had taken the form of a *plate* for greater surface area and correspondingly higher tube current capabilities), the stream of electrons flowing from filament to plate could be regulated by the application of a small voltage between the metal screen and the filament:

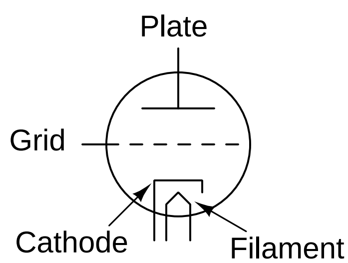


It wasn't just the amount of control voltage between grid and filament that varied the flow of electrons from filament to plate, it was the polarity as well. A negative potential applied to the grid with respect to the filament would choke off the natural flow of electrons, whereas a positive voltage would enhance the flow. Although there was some amount of current through the grid, it was very small – much smaller than the current through the plate.

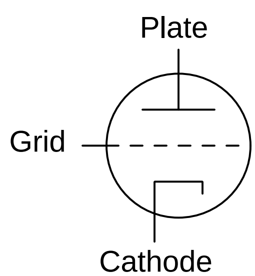
De Forest called his three-element vacuum tube the *Audion*, and its advent marked the beginning of the field of *electronics* by allowing one signal to control the flow of electrons without any moving components. Today we generally refer to three-element vacuum tubes as *triodes*. You'll note the "tri" refers to three while "di" refers to two, in regard to the number of elements within each tube (triode and diode, respectively).

Later developments in vacuum tube technology led to the refinement of the electron emitter: instead of using the filament wire directly as the emissive element, another metal strip called the *cathode* could be heated by the filament. This refinement was necessary in order to avoid some undesired effects of an incandescent filament as an electron emitter. First, a filament experiences a voltage drop along its length, as current overcomes the resistance of the filament material and dissipates heat energy. This resulted in a potential *gradient* from one end of the filament to the other, resulting in uneven electron emission from the filament's surface, as well as electron emission being affected by the magnitude and polarity of the energizing source. If using alternating current (AC) to energize the filament, this voltage gradient will alternate as well, thereby introducing AC "noise" into the rest of the tube circuit. Furthermore, the surface area of a thin filament wire is limited at best, and limited surface area on the electron emitting element tends to place a corresponding limit on the tube's current-carrying capacity. An indirectly-heated filament may be made of metal sheet with much greater surface area, and of a material having greater emissivity than the tungsten wire filaments must be made from.

Indirectly-heated cathodes usually take the form of a thin metal cylinder fitting snugly over a compact filament made of twisted wire. The cathode cylinder would be heated by the filament wire enough to freely emit electrons, without the undesirable side effects of actually carrying the heating current as the filament wire had to. The tube symbol for a triode with an indirectly-heated cathode is as follows:



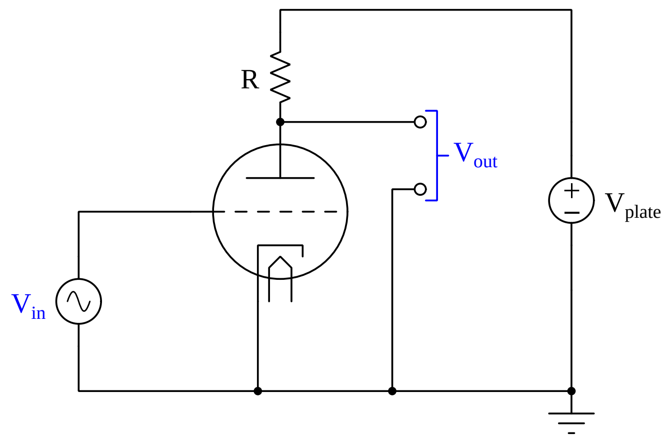
Since the filament is necessary for all but a few types of vacuum tubes, it is often omitted in the symbol for simplicity, or it may be included in the drawing but with no power connections drawn to it:



As the schematic symbol shows, a triode tube's grid necessarily lies between its cathode and its plate. Early experimental tubes placed the grid roughly half-way between the cathode (then, usually just a bare filament) and plate, but this is not the most effective location for the grid to exert control over electron passage from cathode to plate. Bear in mind that these emitted electrons are motivated by force as a result of any electric field between the tube electrodes: making the cathode negative and the plate positive creates an electric field between those electrodes accelerating electrons from cathode to plate. Any controlling voltage connected between the grid and cathode creates a second electric field, and electrons at the cathode's surface experience the superposition of both electric fields. This is why a negative potential at the grid relative to the cathode inhibits electron emission, by acting to cancel out the plate's electric field felt at the cathode surface. Since electric field strength is simply voltage over distance, measured in units of *Volts per meter*, the closer the grid may be located to the cathode's surface, the greater the grid-to-cathode electric field strength will be for any amount of controlling voltage, and therefore the greater effect on plate current that controlling voltage will have. This is why later tube designs placed the grid wires much closer to the cathode than to the plate.

A simple triode circuit appears below to illustrate its basic operation as an amplifier:

Elementary triode amplifier circuit



As the input signal (V_{in}) varies in magnitude and polarity, plate current varies with it: as V_{in} becomes more negative, I_{plate} decreases; as V_{in} becomes more positive, I_{plate} increases. These variations in plate current cause voltage across the resistor to vary as well in accordance with Ohm's Law ($V = IR$). Given a fixed DC plate voltage source, by Kirchhoff's Voltage Law the voltage measured between plate and ground (V_{out}) must vary to complement the resistor's voltage drop. Thus, as V_{in} increases in a positive direction, V_{out} decreases toward zero.

3.5 Tetrode vacuum tubes

3.6 Beam power and pentode vacuum tubes

3.7 Vacuum tube performance parameters

3.8 Vacuum tube characteristic curves

3.9 Display tubes

3.10 Special RF vacuum tubes

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]`

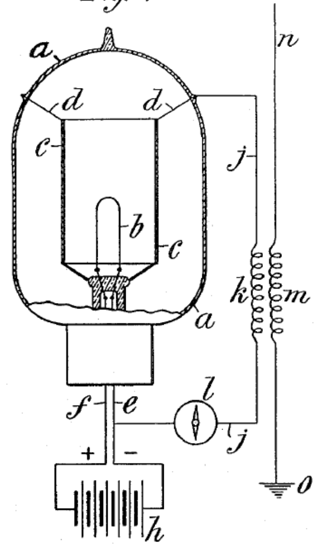
Furthermore, grasping the historical context of technological discoveries is important for understanding how science intersects with culture and civilization, which is ever important because new discoveries and new applications of existing discoveries will always continue to impact our lives. One will 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 Fleming's rectifier tube patent

John Ambrose Fleming, a British professor of electrical engineering and prolific author of electrical textbooks, was the first to put the Edison Effect to practical use in rectifying AC signals into DC. His US patent (number 803684, application 19 April 1905, granted 7 November 1905, "Instrument For Converting Alternating Electric Currents Into Continuous Currents") describes his invention in some detail, with particular application to the rectification of radio-frequency AC signals in order to aid in their detection by instruments responding best to DC.

> I have discovered that if two conductors are inclosed in a vessel in which a good vacuum is made, one being heated to a high temperature, the space between the hot and cold conductors possesses a unilateral electric conductivity, and negative electricity can pass from the hot conductor to the cold conductor, but not in the reverse direction. As the hot conductor should be heated to a very high temperature – say near to the melting-point of platinum, (1,700 centigrade) – it should be of carbon, preferably in the form of a filament such as is used in any ordinary incandescent electric lamp. The cold conductor may be of many materials; but I prefer a bright metal, such as platinum or aluminium or else carbon. The two conductors are inclosed in a glass bulb similar to that of an incandescent lamp, and I generally heat the carbon filament to a high state of incandescence by a continuous electric current, the electrical connection to the filament and the cold conductor being made by platinum wires, sealed air-tight through the glass.
>
> Figure 1 is a full-size sectional elevation of an instrument constructed according to this invention, the electrical connections being shown diagrammatically. This figure illustrates the application of the invention to wireless telegraphy. Figs. 2 and 3 show modifications to smaller scales:

Fig.1.



In Fig. 1, *a* is a glass bulb, and *b* is a carbon filament like the carbon filament of an incandescent lamp, suitable, say, for taking a current of six to eight volts and two to four amperes. *c* is a cylinder of aluminium, open at the top and bottom, which surrounds but does not touch the filament. The cylinder *c* is suspended and steadied by platinum wires *d*, and the ends of the filament *b* are connected to platinum wires connected to the leads *e* and *f*. The platinum wires are sealed through the glass in the ordinary manner.

As a very high vacuum should be obtained in the bulb *a* and as a considerable quantity of air is occluded in the conductors, these should be heated when the bulb is being exhausted. The filament *b* can be conveniently heated by passing a current through it, while the cylinder *e* can be heated by surrounding the bulb *a* with a resistance-coil through which a current is passed, the whole being inclosed in a box lined with asbestos or the like. When, as hereinafter described, the cylinder *c* is replaced by any form of conductor which can be heated by passing a current through it, this method is usually more convenient than that just described.

The carbon filament is made highly incandescent in the usual way by a continuous electric current produced by the battery *h*, the negative pole of which is connected to the wire *e* and the positive to the wire *f*. The wires *d* and *e* are connected together by a wire *j* which completes the circuit through the secondary winding *k* of an induction-coil (such as is ordinarily used in wireless telegraphy) and a galvanometer *l*.

m is the primary winding of the induction coil, having one end connected, as is usual, to an aerial wire *n* and the other to earth *o*. The arrangement described above operates as an electric valve and permits negative electricity to flow from the hot carbon *b* to the metal cylinder *c*, but not in the reverse direction, so that the alternations induced in the coil *k* by the Hertzian waves received by the aerial wire *n* are rectified or transformed into a more or less continuous current capable of actuating the galvanometer *l* by which the signals can be read. . .

4.2 De Forest's triode tube circuit patent

Lee De Forest patented a technique in 1908 for receiving radio telegraphy signals utilizing one of his "Audion" tubes as the active electronic component. The US patent numbered 879532 ("Space Telegraphy", application 29 January 1907, granted 18 February 1908) describes how the circuit worked. Unfortunately, this patent did not elaborate on the functionality of the tube itself, which is a shame being De Forest's earliest patent. Below I will focus on De Forest's explanation of the triode (Audion) circuit shown in Figure 1, as the circuit of Figure 2 utilizes a different (non-triode) form of Audion tube:

> In the drawings, Figure 1 represents in diagram a wireless telegraph receiving system comprising an osciliation detector constructed and connected in accordance with the present invention and Fig. 2 represents a space telegraph receiving system having a modified form of oscillation detector connected therein in a manner which constitutes one of the subjects matter of said invention.

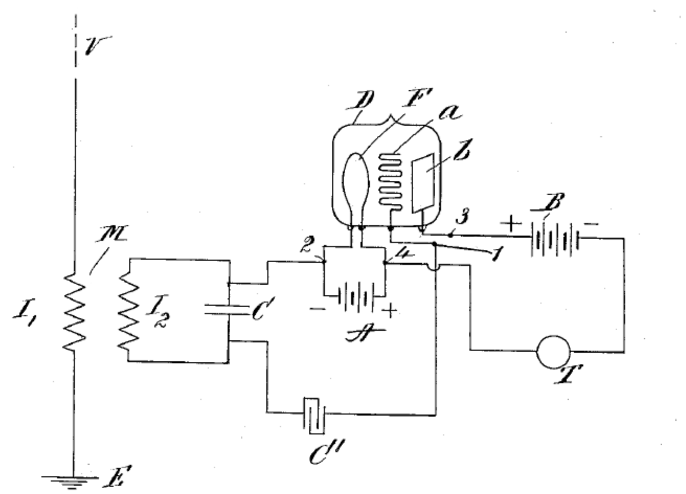


Fig-1-

> V I_1 E is an elevated conductor system including the elevated conductor V, earth connection E, and primary I_1 of the transformer M, the secondary I_2, of which forms part of the tuned receiving circuit I_2 C. It will be understood of course that the aforesaid tuned receiving circuit may be associated with the elevated conductor system in any suitable manner.
>
> D represents an evacuated vessel, preferably of glass, having sealed therein three conducting members, F, a and b, in Fig.1 and F, a' and b in Fig. 2. The conducting member or electrode F is shown as consisting of a filament, preferably of metal, which is connected in series with the battery A or other source of electrical current of sufficient strength to heat said filament, preferably to incandescence. The conducting member b,

which may be a plate of platinum, has one end brought out to the terminal *3*. Interposed between the members F and b is a grid-shaped member a, which may be formed of platinum wire, and which has one end brought out to to the terminal *1*. The local receiving circuit, which includes the battery B, or other suitable source of electromotive force, and the signal indicating device T, which may be a telephone receiver, has its terminals connected to the plate b and filament F at the points 3 and 4 respectively. The means for conveying the oscillations to be detected to the oscillation-detector, are the conductors which connect the filament F and grid a to the tuned receiving circuit and, as shown, said conductors pass from the terminals *2* and *1* to the armatures of the condenser C.

4.3 Telephony amplifier triodes

Early telephone systems designed, built, and maintained by the Bell Telephone company used vacuum triode tubes as electronic amplifying elements. An article in Volume 1 of the *Bell System Technical Journal* (circa 1922) by James Pilliod revealed schematic diagrams of the tube-based amplifier circuits used to boost (or "repeat") the power of weak telephony signals. The schematic diagram shown below (on the left-hand wide) and a corresponding photograph of a vacuum tube (on the right-hand wide) comes from that article:

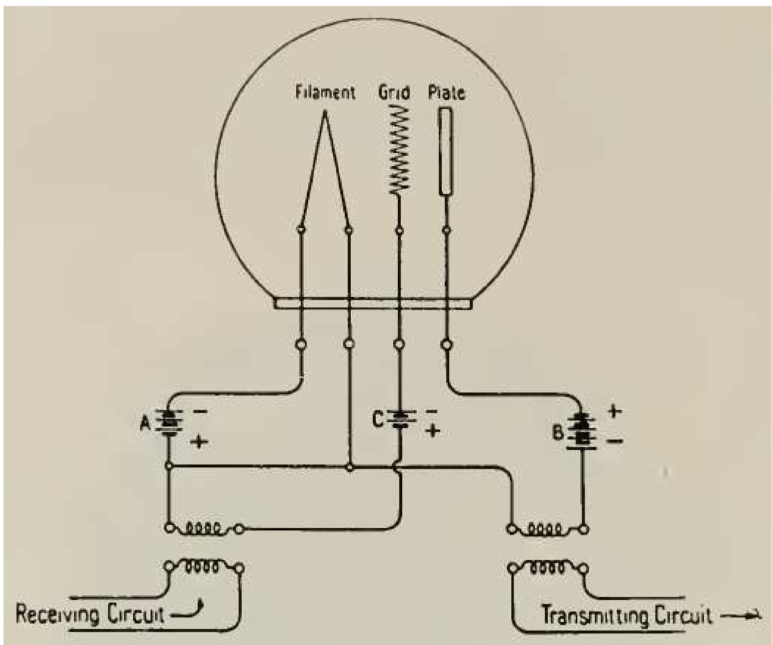


Fig. 21—Vacuum-Tube Repeater Element

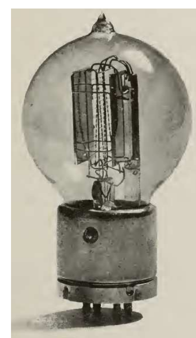


Fig. 20—Vacuum Tube

Interestingly, the + and − polarity symbols shown in this diagram are correct, but the battery component symbols are mis-drawn (where the long-line sides should all be + and the short-line sides should all be −). Battery *A* provides excitation to the vacuum tube's filament, which must be maintained at a high temperature in order to emit free electrons into the vacuum space between the filament and plate. The plate is maintained at a positive polarity (compared to the filament) by battery *B* to attract negatively-charged electrons toward it through the vacuum. The grid is a metal screen placed between the filament and plate, with any voltage between it and the filament regulating the flow of electrons through the spaces in-between the grid wires from the filament to the plate.

Note how this triode tube uses a *directly-heated cathode* where the filament wire itself functions as the electron-emitting cathode surface. Modern vacuum tubes use indirectly-heated cathodes where the cathode is a metal surface in close proximity to the filament, thermally coupled to the filament but electrically insulated from it. Indirectly-heated cathodes provide more surface area for electron emission and exhibit more stable surface temperatures when the filament is energized by AC rather than DC as is the case in this circa-1922 tube.

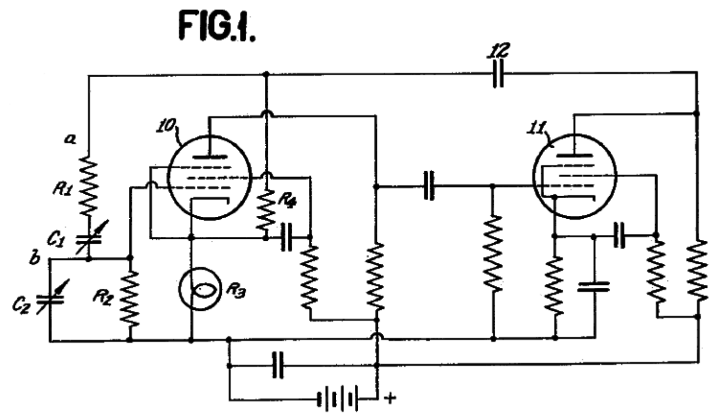
This amplifier receives its AC input signal through the left-hand signal transformer. That transformer's secondary voltage appears between the grid and filament, modulating[1] the flow of electrons from filament to plate. As the AC input signal varies over time, so does the plate's current. This time-varying current passes through the primary winding of the right-hand signal transformer, inducing an AC voltage at the secondary winding which constitutes the amplified ("repeated") signal.

Biasing for the vacuum tube comes from battery C, offsetting the grid's potential to be more negative than it would be otherwise. This is necessary because vacuum tubes behave much like N-channel junction field-effect transistors (JFETs) where a larger range of control exists on the depletion side than the enhancement side of the device's characteristic curve. Without some negative biasing, this amplifier would likely distort the AC signal, resulting in "rounding" or even "clipping" of the waveform with each positive peak of the input signal.

[1] A negatively-charged grid (with reference to the filament) repels negatively-charged electrons and thereby restricts their motion toward the plate. A positively-charged grid attracts electrons to accelerate them toward the plate.

4.4 William Hewlett's oscillator

One of the founders of the famous electronics manufacturer *Hewlett-Packard*, William Hewlett, filed for a US patent in early July of 1939 for a reliable sine-wave oscillator. This patent (number 2,268,872) was granted on the 6th of January 1942, and it shows a Wein-bridge design using vacuum tubes (*pentodes*, to be precise) as amplifying elements:



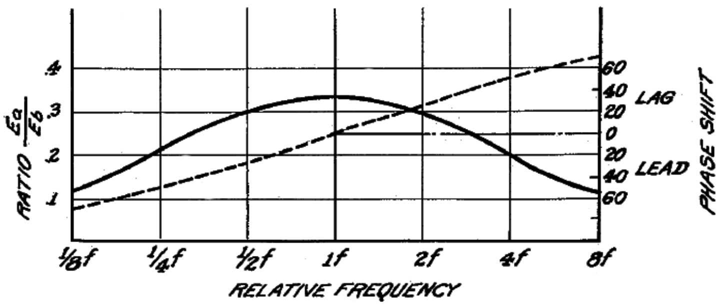
The text of Hewlett's patent begins as such on page 2 of the patent document:

> My invention relates to variable frequency oscillation generators and more particularly to oscillation generators wherein the frequency is determined by means of resistance capacity networks.
>
> Oscillation generators capable of tuning over a wide range of frequencies are useful for many purposes, for example, as test oscillation generators in place of beat frequency generators. In general such oscillation generators have been provided with a variably tuned resonant circuit to maintain the oscillations at a frequency to which the tuned circuit is adjusted.
>
> According to my invention, I provide an oscillation generator with a variable resistance capacity network forming a positive feedback path for the purpose of determining the frequency of oscillation.
>
> According to another feature of my invention an oscillation generator with resistance capacity frequency determining arrangements and with a negative feedback circuit for reducing distortion caused in the amplifier circuit, is provided.
>
> According to a further feature of my invention I provide an automatic amplitude control arrangement in a negative feedback path of an oscillation generator to maintain the amplitude of the generated oscillations constant despite changes in the supply voltages for the tubes. `[page 2]`

Hewlett continues, on the same page, making reference to the illustrations:

> Other advantages and features of my invention will be apparent from the particular description thereof made in connection with the accompanying drawing, in which
>
> Fig.1 shows a diagrammatic circuit of the preferred form of an oscillation generator in accordance with my invention, and
>
> Fig. 2 shows curves for the purpose of illustrating the operation of the circuit.
>
> In Fig. 1 is shown an oscillation generator comprising two amplifying tubes 10 and 11, coupled together by resistance capacity means in a known manner. The voltage supply sources and the cathode heating supply are omitted for simplifying the circuit. The anode of tube is coupled to the input grid of tube and the anode of tube is then connected to a circuit which provides a feedback path through a coupling condenser 12 and series connected resistance R_1 and variable capacity C_1, and a second circuit comprising R_2 and variable condenser C_2, to the input of tube 10. The circuit including R_1, C_1, R_2, C_2, provides a regenerative or positive coupling between the output and input circuits of the amplifiers 10 and 11, so that feedback occurs sufficient to maintain oscillations in the circuit. By properly proportioning the value of the resistance capacity networks R_1, C_1, R_2, C_2, the desired frequency may be obtained. If R_1, and C_1, respectively, equal R_2, and C_2, the ratio of the voltage E_b at point b to the voltage at point a varies with frequency in a manner similar to a resonance curve. In Fig. 2 this curve is shown in solid lines with voltage ratios of E_a and E_b plotted as ordinates against the relative frequency. At the maximum of this curve, the frequency
>
> $$f_o = \frac{1}{2\pi\sqrt{R_1, R_2, C_1, C_2}}$$
>
> and the voltages at points a and b have the same phase. Accordingly, oscillations tend to occur at the frequency f_o for this circuit. `[page 2]`

FIG. 2.



In Fig. 2 the broken line curve indicates the phase shifts as ordinates with the change in relative frequency. It can be seen that at zero phase shift, the maximum of the E_a, E_b ratio occurs. If the various condensers and resistances are not made equal as they were for plotting this curve, the maximum would occur at some other point and a symmetrical curve would not be obtained. However, the principles of my invention apply to the circuit regardless of equality of these elements. In practical construction, however, it is generally desirable to make the separate elements equal.

As shown in Fig. 1, frequency shift in the oscillation generator is accomplished by adjusting variable capacities C_1 and C_2. It is clear, however, that since the resistances as well as the capacities serve to control the frequency of the oscillator, either the resistances alone or both the resistances and capacities may be adjustable for varying the tuning.

For the type of oscillators illustrated in Fig. 1, it is necessary in order to obtain satisfactory operation that the amplifiers comprising tubes 10 and 11 have phase shift independent of changes in supply, etc., and furthermore, there should be provided some means for controlling the amplitude of the oscillations so that they do not exceed the range over which tubes 10 and 11 will operate as class A amplifiers. A constant amplifier phase shift is necessary to insure a constant frequency, because the phase angle of the transfer impedance of the resistance capacity network from point *a* to point *b* varies only slowly with frequency. Accordingly, a small change in amplifier phase shift such as might be produced by a variation in voltage supply requires a comparatively large change in the frequency of opera- `[page 2]`

tion to produce a compensating phase shift in the resistance capacity Coupling System. `[page 3]`

After explaining the basic operation of the oscillator circuit, Hewlett begins to describe some of the more advanced features of his design. He begins by explaining how his circuit is stabilized against variations such as tube parameters, temperature, and supply voltage (using negative feedback):

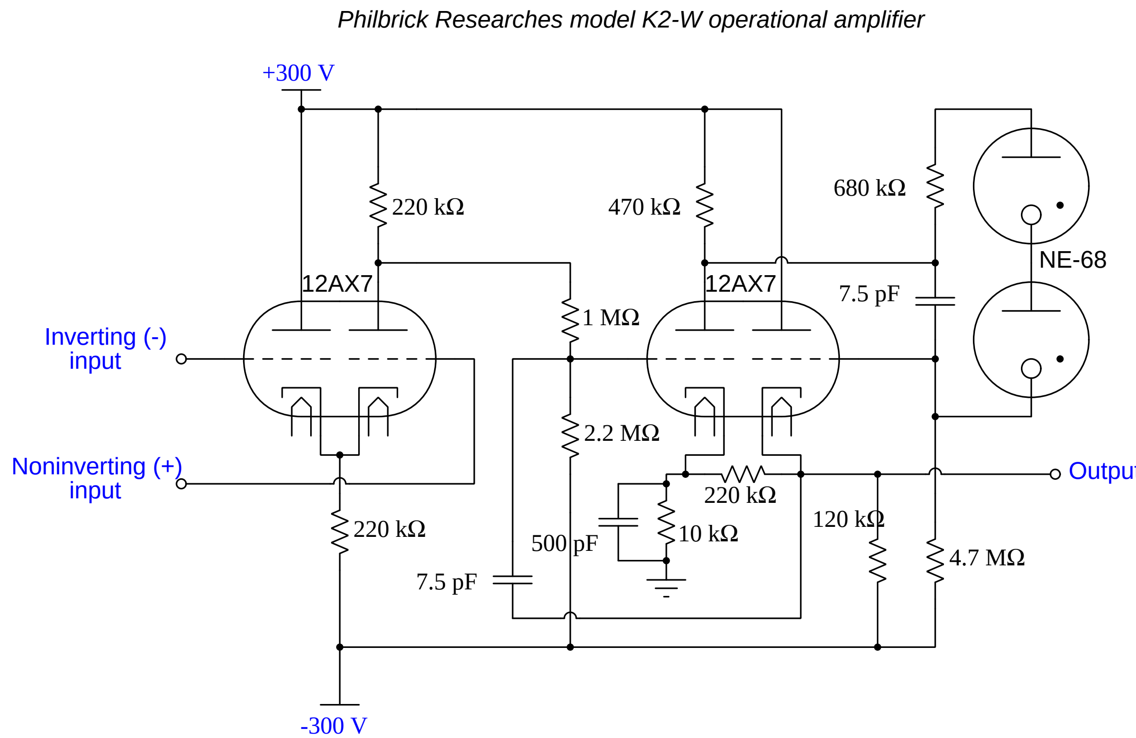
> In the circuit of Fig. 1, I provide a negative feedback arrangement for stabilizing the amplifier. This feedback is made from the output of amplifier 11 over coupling condenser 12 and the resistance R_3, R_4 to the cathode of tube 10. The potential drop across the resistance produces a negative feedback to the input of amplifier 10. This negative feedback operates in a manner known per se to reduce the amplification of the system and at the same time to stabilize the amplifier gain in magnitude and phase with respect to variations occurring within the amplifier, such as tube changes, temperature variations and changes caused by fluctuation in the voltage supplies to the electrodes of the amplifier system. `[page 3]`

Next, he explains what is perhaps the most innovative feature of his oscillator's design, a method for automatically controlling gain for the purpose of reducing waveform distortion:

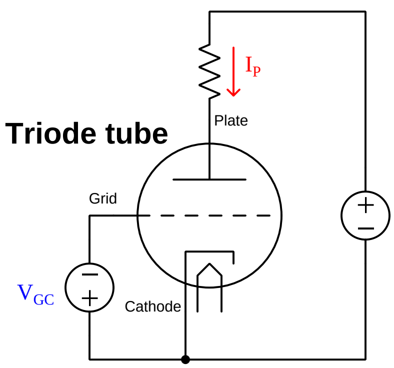
> Amplitude control to prevent the oscillations from building up to such a large value that distortion occurs, is obtained according to my invention by non-linear action in the amplifier circuit. In order to produce this non-linear variation, I provide for resistance R_3 a small incandescent lamp, or similar device in which the resistance increases rapidly with increased current flow, the lamp being heated by the plate current of the tubes 10 and 1l or by an auxiliary means, so such a temperature that its resistance will vary rapidly with a small change in current. Thus, when the oscillation amplitudes tend to increase, the temperature of the lamp R_3 increases with a resulting increase in resistance thereby causing a greater negative feedback, thus reducing the amplification. Similarly, as the oscillations decrease in amplitude, the current through the lamp is reduced permitting the lamp to cool with an accompanying decrease in resistance and reduction of the negative feedback, thus increasing the amplitude of the generated oscillations. As a result the system operates at substantially a constant amplitude which is preselected to be below the value at which grid current flows. As a result no distortion of the wave form takes place.
>
> While I have described the amplitude control system in connection with my particular resistance capacity tuned oscillation generator, it is clear that this feature of my invention is similarly applicable to various types of oscillation generators, wherein frequency stabilization is obtained by means of negative feedback. Since the lamp itself responds relatively slowly to current changes it tends to average out the voltage peaks and does not cut off the peaks of the generated waves, maintaining the amplitude constant without producing distortions in wave form. `[page 3]`

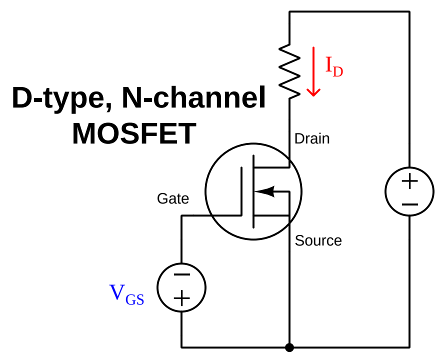
4.5 Philbrick Researches model K2-W operational amplifier

George A. Philbrick Researches, Incorporated, was a manufacturer of analog electronic computing systems founded in 1946. One of their products was a compact (for its time) operational amplifier based on two model 12AX7 triode vacuum tubes called the *K2-W*, a schematic diagram of which is shown here:

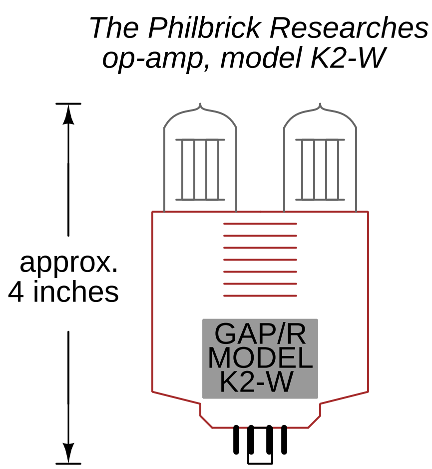


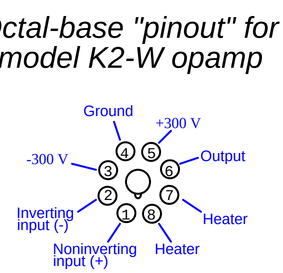
The vacuum tubes used in this circuit design are *dual triode tubes*, model 12AX7. Each tube contained two separate triode elements, each one behaving similar to a D-type N-channel MOSFET where conduction is controlled by an applied voltage between two of the device's terminals.





The model K2-W operational amplifier housed the two vacuum tubes along with all passive components shown in the schematic diagram neatly inside of a plastic case, with an eight-pin (octal) plug in the bottom. This octal-base pattern was the same used for some large power vacuum tubes as well as for electromechanical relays, and in fact this same octal pin pattern survives to this day for relays. Illustrations of a typical K2-W module and its pin assignments are shown below:





This amplifier unit was released for individual sale in 1952, for the purpose of allowing customers to build their own analog computing and signal-processing systems using general-purpose operational amplifier modules. A professor of electrical engineering at MIT, Henry M. Paynter, wrote a 28-page pamphlet titled *Applications Manual for Philbrick Octal Plug-In Computing Amplifiers* just a few years later in 1956, and with the publication of this document Philbrick Researches saw individual opamp sales exceed sales for their complete analog computing systems.

Technical specifications for the K2-W were as follows, taken from the manufacturer's datasheet:

- $A_V(OL)$ = 15,000 open-loop voltage gain
- I_{supply} = 4.5 mA DC at ± 300 VDC
- Heater power = 6.3 Volts (AC or DC) at 0.6 Amperes
- R_{input} = at least 100 MΩ
- Input voltage range = ± 50 Volts
- Output voltage range = ± 50 Volts
- Input bias current = less than 0.1 μA
- Output load current = ± 1 mA driving 50 kΩ load resistance

For comparison, technical specifications for the legacy model 741 (solid-state) operational amplifier introduced in 1968 show dramatic improvements in most metrics:

- $A_V(OL)$ = 50,000 to 200,000 open-loop voltage gain
- I_{supply} = 1.7 mA DC at ± 15 VDC
- (no heaters required!)
- R_{input} = 300 kΩ to 2 MΩ
- Input voltage range = ± 13 Volts
- Output voltage range = ± 14 Volts
- Input bias current = less than 0.1 μA
- Output load current = ± 25 mA driving a short-circuit

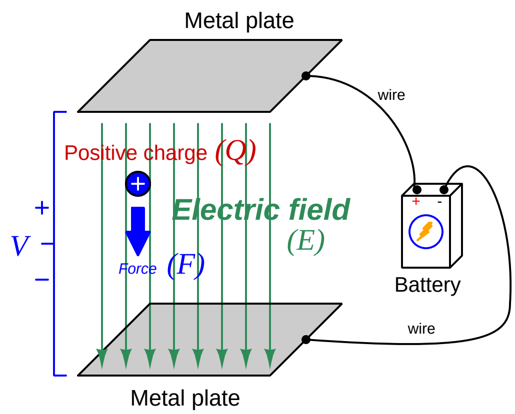
Chapter 5

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 Electric field quantities

A useful definition of electric field (E) is in terms of the force (F) exerted on an electric charge (Q) influenced by that field:



$$\vec{F} = Q\vec{E}$$

Where,
$\vec{F}$ = Force exerted on the charge (Newtons)
Q = Charge quantity (Coulombs[1])
$\vec{E}$ = Electric field (Newtons per Coulomb)

The small "arrow" symbols above the variables for force and electric field in the equation denote those variables as *vector quantities*, having both magnitude and direction. Charge is a *scalar quantity* having only magnitude but no direction, and as a scalar quantity when multiplied by the electric field vector it simply magnifies the magnitude but does not alter the direction. Therefore, the force and electric field vectors always point in the same direction.

Alternatively electric field may be defined in terms of the voltage between the end-points and the distance separating them, in which case we may express the electric field in units of *Volts per meter* as an alternative to *Newtons per Coulomb*:

$$\vec{E} = \frac{V}{\vec{d}}$$

This measurement of electric field strength is very important for quantifying the *breakdown* of electrical insulators: the point at which the electric field becomes so powerful that otherwise immobile charges within the insulating substance are torn free to constitute a current and that substance is no longer an insulator. For rating the dielectric strength of insulating materials, we often see electric fields expressed in units of *kiloVolts per millimeter* rather than Volts per meter just to make the numerical quantities easier to manage (1 kV/mm = 1 *million* V/m).

[1] One Coulomb of electric charge is equal to 6.2415×10^{18} electrons.

The vector arrows shown in the previous illustration representing the electric field between two metal plates actually represent electric *flux* (Φ_E). The electric field (E) is related to electric flux by area (A), the field being a measurement of how densely-packed those flux lines are per unit area:

$$\vec{E} = \frac{\Phi_E}{\vec{A}}$$

Where,
$\vec{E}$ = Electric field, or electric flux density (Newtons per Coulomb)
Φ_E = Electric flux (Newton-meter squared per Coulomb)
$\vec{A}$ = Area over which flux is distributed (square meters)

The mere presence of an unbalanced electric charge at any point in space is sufficient to generate lines of electric flux, the total magnitude of that flux predicted by the following equation:

$$\Phi_E = \frac{Q}{\epsilon}$$

Where,
Φ_E = Electric flux (Newton-meter squared per Coulomb)
Q = Charge quantity (Coulombs)
ϵ = Electric permittivity of the surrounding space (Coulombs squared per Newton-meter squared, approximately 8.85×10^{-12} for empty space)

By convention, these flux vectors point *away* from positive charges and point *toward* negative charges, their direction indicating force exerted on any positively-charged particle influenced by that field[2]. As the equation states, the amount of flux depends on how much charge exists at each particle as well as the permittivity of the surrounding space:

For example, identical charges suspended in a vacuum versus in a substance such as oil will have different amounts of flux associated with them as a result of oil and vacuum having different permittivity values. Perfectly empty space has the least amount of permittivity, which means anything else (gas, liquid, or solid matter) has greater ϵ which acts to diminish the amount of electric flux surrounding any charged particle. Superconducting materials have infinite permittivity, which means they forbid the existence of any electric field inside their bulk.

[2]Conversely, the flux vectors point exactly opposite the direction of force applied to any *negatively*-charged particle within that field. This makes sense of the aphorism that *like charges repel and opposite charges attract*. If you consider the two charges shown in this illustration, the positive charge will be pulled in the direction of the flux vectors pointing toward the negative charge, as the negative charge will also be pulled opposite the direction of the flux vectors pointing away from the positive charge (i.e. the negative charge will be pulled *toward* the positive charge) – thus the positive and negative charges feel mutual attraction.

Chapter 6

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*.

6.1 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.

6.2 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 ($441.717\ \Omega \angle -25.102^o$). 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.

6.3 Modeling ??? using C++

Chapter 7

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?

7.1 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.

7.1.1 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:

$\boxed{\surd}$ 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.

$\boxed{\surd}$ 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.

$\boxed{\surd}$ 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.

$\boxed{\surd}$ 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.

$\boxed{\surd}$ 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.

$\boxed{\surd}$ Specifically identify any points you found CONFUSING. The reason for doing this is to help diagnose misconceptions and overcome barriers to learning.

7.1.2 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.

Energy

Conservation of Energy

Simplification as a problem-solving strategy

Thought experiments as a problem-solving strategy

Limiting cases as a problem-solving strategy

Annotating diagrams as a problem-solving strategy

Interpreting intermediate results as a problem-solving strategy

Graphing as a problem-solving strategy

Converting a qualitative problem into a quantitative problem

Converting a quantitative problem into a qualitative problem

Working "backwards" to validate calculated results

Reductio ad absurdum

Re-drawing schematics as a problem-solving strategy

Cut-and-try problem-solving strategy

Algebraic substitution

???

7.1.3 First conceptual question

Challenges

- ???.
- ???.
- ???.

7.1.4 Second conceptual question

Challenges

- ???.
- ???.
- ???.

7.1.5 Applying foundational concepts to ???

Identify which foundational concept(s) apply to each of the declarations shown below regarding the following circuit. If a declaration is true, then identify it as such and note which concept supports that declaration; if a declaration is false, then identify it as such and note which concept is violated by that declaration:

(Under development)

- ???
- ???
- ???
- ???

Here is a list of foundational concepts for your reference: **Conservation of Energy**, **Conservation of Electric Charge**, **behavior of sources vs. loads**, **Ohm's Law**, **Joule's Law**, **effects of open faults**, **effect of shorted faults**, **properties of series networks**, **properties of parallel networks**, **Kirchhoff's Voltage Law**, **Kirchhoff's Current Law**. More than one of these concepts may apply to a declaration, and some concepts may not apply to any listed declaration at all. Also, feel free to include foundational concepts not listed here.

Challenges

- ???.
- ???.
- ???.

7.1.6 Explaining the meaning of calculations

Below is a quantitative problem where all the calculations have been performed for you, but all variable labels, units, and other identifying data are unrevealed. *Assign proper meaning* to each of the numerical values, identify the correct unit of measurement for each value as well as any appropriate metric prefix(es), explain the significance of each value by describing where it "fits" into the circuit being analyzed, and identify the general principle employed at each step:

Schematic diagram of the ??? circuit:

(Under development)

Calculations performed in order from first to last:

1. $x + y = z$
2. $x + y = z$
3. $x + y = z$
4. $x + y = z$
5. $x + y = z$
6. $x + y = z$

Challenges

- ???.
- ???.
- ???.

7.1.7 Explaining the meaning of code

Shown below is a schematic diagram for a ??? circuit, and after that a source-code listing of a computer program written in the ??? language simulating that circuit. Explain the purpose of each line of code relating to the circuit being simulated, identify the correct unit of measurement for each computed value, and identify all foundational concepts of electric circuits (e.g. Ohm's Law, Kirchhoff's Laws, etc.) employed in the program:

Schematic diagram of the ??? circuit:

(Under development)

Code listing:

```
#include <stdio.h>

int main (void)
{

  return 0;
}
```

Challenges

- ???.
- ???.
- ???.

7.2 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.

7.2.1 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-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.

7.2.2 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.*

7.2.3 First quantitative problem

Challenges

- ???.
- ???.
- ???.

7.2.4 Second quantitative problem

Challenges

- ???.
- ???.
- ???.

7.2.5 ??? simulation program

Write a text-based computer program (e.g. C, C++, Python) to calculate ???

Challenges

- ???.
- ???.
- ???.

7.3 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.

7.3.1 First diagnostic scenario

Challenges

- ???.
- ???.
- ???.

7.3.2 Second diagnostic scenario

Challenges

- ???.
- ???.
- ???.

Chapter 8

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!*

8.1 Recommended practices

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

8.1.1 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.

8.1.2 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.

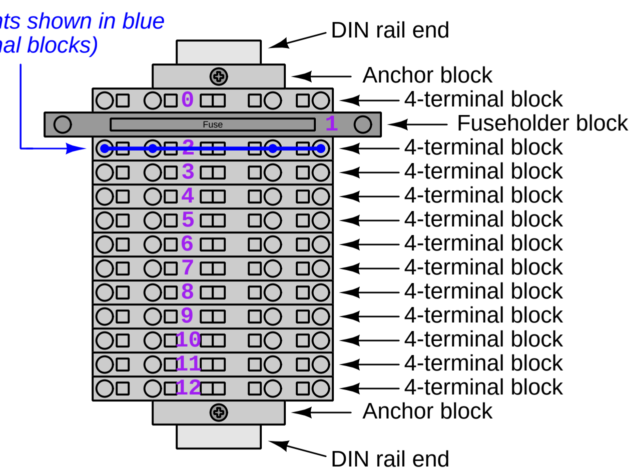
[4]An example of a necessary resistor value much less than 1 kΩ is a *shunt resistor* used to produce a small voltage drop for the purpose of sensing current in a circuit. Such shunt resistors must be low-value in order not to impose an undue load on the rest of the circuit. An example of a necessary 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.

8.1.3 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.

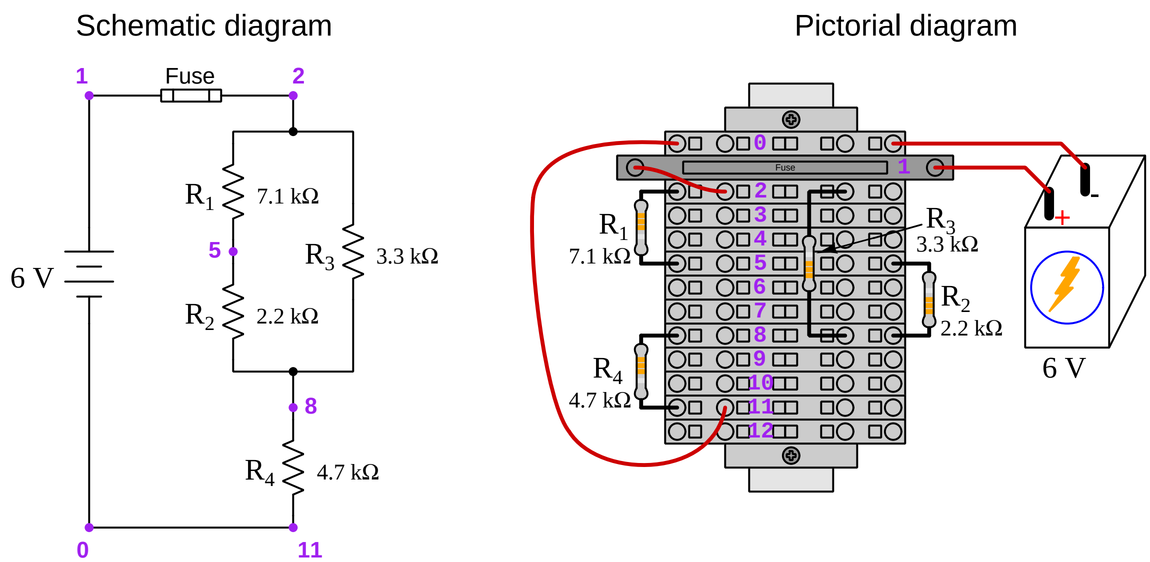
[6]Solderless breadboard are preferable for complicated electronic circuits with multiple integrated "chip" components, but for simpler circuits I find terminal blocks much more practical. An alternative to solderless breadboards for "chip" circuits is to solder chip sockets onto a PCB and then use wires 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
```

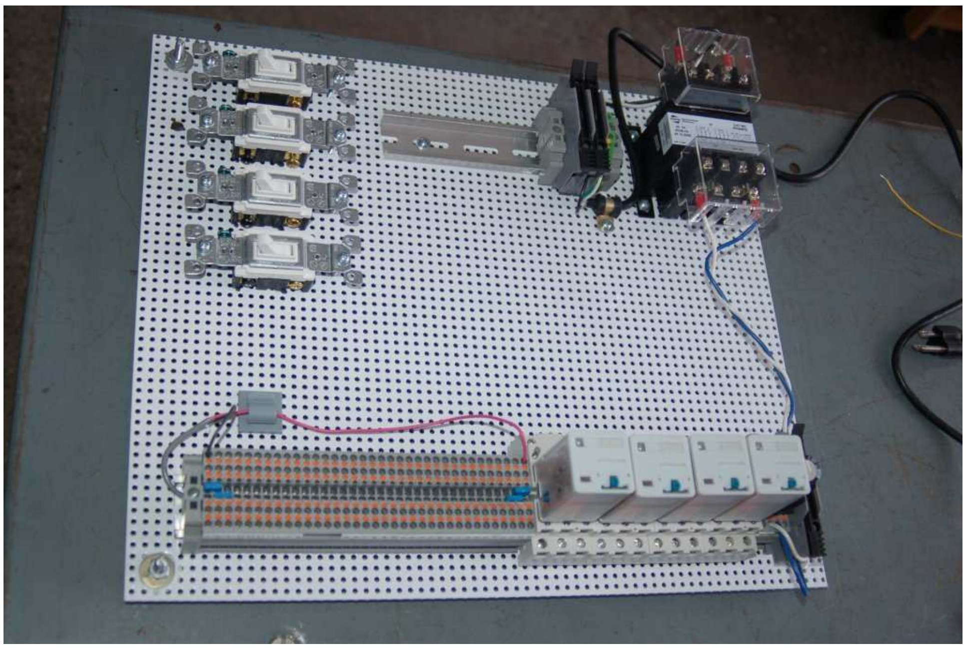
Note the use of two resistances `rfuse` and `rwire` to describe fuse resistance between terminals 1 and 2 and "jumper" wire resistance between terminals 0 and 11, respectively. Being resistances, SPICE requires a resistance value for each, and here we see they have both been set to an arbitrarily low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

8.1.4 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.

8.1.5 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.

8.2 Experiment: (first experiment)

Conduct an experiment to . . .

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!
- ???.
- ???.

8.3 Project: (first project)

This is a description of the project!

PROJECT CHECKLIST:

- Prior to construction:
  - ☑ Prototype diagram(s) and description of project scope.
  - ☑ Risk assessment/mitigation plan.
  - ☑ Timeline and action plan.
- During construction:
  - ☑ Safe work habits (e.g. no contact made with energized circuit at any time).
  - ☑ Correct equipment usage according to manufacturer's recommendations.
  - ☑ Timeline and action plan amended as necessary.
  - ☑ Maintain the originally-planned project scope (i.e. avoid adding features!).
- After completion:
  - ☑ All functions tested against original plan.
  - ☑ Full, accurate, and appropriate documentation of all project details.
  - ☑ Complete bill of materials.
  - ☑ Written summary of lessons learned.

Challenges

- ???.
- ???.
- ???.

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.

Appendix E

References

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.

2-12 July 2026 – document first created, added content bit by bit. A significant portion of the Tutorial was created by borrowing and editing text and images from an old version of my *Lessons In Electric Circuits* volume III book published before anyone else contributed to that chapter ("Electron Tubes").

Index