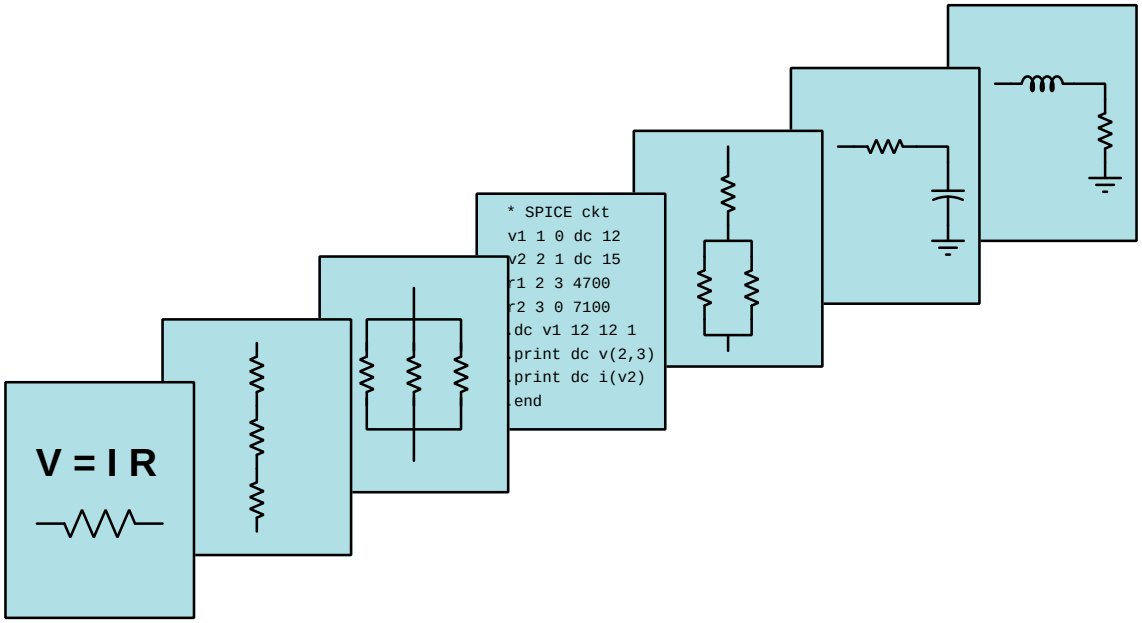
Modular Electronics Learning (ModEL) project



Digital Signal Integrity

© 2020-2025 by Tony R. Kuphaldt – under the terms and conditions of the Creative Commons Attribution 4.0 International Public License

Last update = 27 September 2025

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

Contents

1 Introduction **3**
- 1.1 Recommendations for students . . . 3
- 1.2 Challenging concepts related to digital signal integrity . . . 5
- 1.3 Recommendations for instructors . . . 7

2 Case Tutorial **9**
- 2.1 Oscilloscope, cable, and signal generator . . . 10
- 2.2 Demonstration of set-up time . . . 14

3 Tutorial **17**
- 3.1 Digital logic levels . . . 18
- 3.2 Pulse imperfections . . . 19
- 3.3 Consequences of distorted pulses . . . 23
- 3.4 Radiated interference . . . 26
- 3.5 Conducted interference . . . 28
- 3.6 Diagnostics . . . 31

4 Historical References **35**
- 4.1 Signal integrity in wireless versus cable-based telephony . . . 36

5 Derivations and Technical References **39**
- 5.1 TTL logic levels . . . 40
- 5.2 CMOS logic levels . . . 41
- 5.3 Digital pulse criteria . . . 42
- 5.4 Latch-up in integrated circuits . . . 46

6 Animations **57**
- 6.1 Animation of an open transmission line . . . 58

7 Questions **75**
- 7.1 Conceptual reasoning . . . 79
  - 7.1.1 Reading outline and reflections . . . 80
  - 7.1.2 Foundational concepts . . . 81
  - 7.1.3 Measuring rise and fall times . . . 84
  - 7.1.4 Transient oscillation . . . 85

7.1.5 The Soul of a New Machine . . . 86
7.1.6 Two crosstalk scenarios . . . 88
7.1.7 Simultaneous crosstalk . . . 89
7.2 Quantitative reasoning . . . 90
7.2.1 Miscellaneous physical constants . . . 91
7.2.2 Introduction to spreadsheets . . . 92
7.2.3 Measuring jitter using an eye diagram . . . 95
7.2.4 Evaluating set-up and hold times . . . 96
7.3 Diagnostic reasoning . . . 97
7.3.1 Testing a cable for magnetic crosstalk . . . 98
7.3.2 Pulse waveform oscillographs . . . 99

A Problem-Solving Strategies **101**

B Instructional philosophy **103**
B.1 First principles of learning . . . 104
B.2 Proven strategies for instructors . . . 105
B.3 Proven strategies for students . . . 107
B.4 Design of these learning modules . . . 108

C Tools used **111**

D Creative Commons License **115**

E References **123**

F Version history **125**

Index **126**

Chapter 1

Introduction

1.1 Recommendations for students

Digital signaling seems like it ought to be simple. After all, there's only "high" and "low", right? Analog signals, with their infinite resolution, may be corrupted by *any* amount of distortion or noise, but digital should be immune to those faults.

Or is it? Despite the fact that digital signals are defined by their limited number of states, they too may suffer from distortion and noise because the voltage and current pulses comprising digital "high" and "low" states are still fundamentally analog quantities. The same electro-magnetic principles applying to analog voltage and current signals apply to digital voltage and current signals as well, which means the exact same *mechanisms* of distortion and noise apply. The only difference is that digital signals get *interpreted* differently than analog, and this interpretational difference helps ignore certain types of distortion and noise.

Important concepts related to digital signal integrity include **logic states**, **logic levels**, **high** and **low** logic states, **noise margin**, **capacitance**, **inductance**, **rates of change**, the **Conservation of Energy**, **Fourier series** and the **frequency domain**, **harmonic frequencies**, **fundamental frequency**, **indeterminate** logic states, **propagation delay**, **Schmitt trigger** gates, **jitter**, **set-up** and **hold** times, electromagnetic **waves**, **shielded** cables, **twisted-pair** cables, **differential** signaling, **resonance**, **eye diagrams**, and **time-domain reflectometry**.

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

- How might an experiment be designed and conducted to demonstrate the effective shielding from electric fields? What hypothesis (i.e. prediction) might you pose for that experiment, and what result(s) would either support or disprove that hypothesis?
- How might an experiment be designed and conducted to demonstrate the effective shielding from magnetic fields? What hypothesis (i.e. prediction) might you pose for that experiment, and what result(s) would either support or disprove that hypothesis?
- How might an experiment be designed and conducted to measure the ESR of a capacitor intended for decoupling? What hypothesis (i.e. prediction) might you pose for that experiment,

and what result(s) would either support or disprove that hypothesis?

- What does the ideal digital pulse look like on an oscillograph?
- In what ways do real digital pulses deviate from the ideal?
- What does "integrity" mean for a digital voltage signal?
- Why is *noise margin* a function of both the transmitting and receiving circuitry within a digital system?
- How could we empirically determine (i.e. measure) the noise margin of a particular logic system?
- What is a *rate of change*?
- How do *rates of change* relate to the phenomena of capacitance and inductance?
- Why do parasitic effects arise between conductors? What are the fundamental physical principles at work?
- How does Fourier's Theorem apply to the study of signal integrity?
- What does it mean to say that a signal contains "harmonic" frequencies?
- What are some practical ways to minimize signal coupling between different conductors?
- What purpose does the "shield" conductor serve in a cable?
- What purpose do "twisted" pairs of conductors serve in a cable?
- How does differential signaling work to mitigate the effects of coupled noise?
- How do square-wave pulse signals relate to series of sinusoids?
- What does the phrase "nature abhors a perfect square wave" mean?
- What does the phrase "voltage uncertainty" mean?
- What are some parasitic electrical properties of printed circuit boards?
- How are "high" and "low" signal signals distinguished from one another?
- How does a Schmitt-trigger logic gate help improve digital signal integrity?
- What is a disadvantage of using a Schmitt-trigger logic instead of a standard logic gate?
- What is the importance of ensuring adequate set-up and hold times for digital circuits?
- How does the insertion of resistance into signal lines help mitigate ringing?
- What are some diagnostic tools useful for assessing digital signal integrity?
- How is an "eye diagram" created, and what does one tell us about signal integrity?

1.2 Challenging concepts related to digital signal integrity

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

- **Parasitic effects** – phenomena such as resistance, capacitance, and inductance don't just exist where we want them to, but in fact always exist everywhere they can. All conductors (except for superconductors) have some amount of resistance, all conductors separated by an insulating gap exhibit capacitance, and all conductors possess inductance by virtue of their ability to produce a magnetic field with the passage of current. Often these "parasitic" effects are small enough to ignore, but in some applications they may be severe.

- **Rates of change** – when learning the relationships between voltage and current for inductors and capacitors, one must think in terms of how fast a variable is changing. The amount of voltage induced across an inductor is proportional to how *quickly* the current through it changes, not how strong the current is. Likewise, the amount of current "through" a capacitor is proportional to how *quickly* the voltage across it changes. This is the first hurdle in calculus: to comprehend what a rate of change is, and it is far from obvious. A helpful strategy here is to relate electrical rates of change to other, more common, rates of change such as *speed* (i.e. the rate-of-change of physical position over time). Also helpful is to relate rates of change to the *slope* of a line or curve plotted on a graph, showing how *rise/fall over run* describes not only the phenomenon of a rate-of-change but also the appropriate units of measurement.

- **Fourier's Theorem** – while this is no doubt a non-intuitive concept, it is incredibly useful. Knowing that any waveshape whatsoever may be reproduced by summing together the right combination of sinusoidal waves may defy our intuition, but at least seeing it is possible to synthesize common non-sinusoidal waveshapes like square and triangle using nothing but sine and/or cosine waves helps prove one can get non-round waves from lots of round waves added together. The practical upshot of this is that is it possible to consider very complex waveshapes as being nothing more than a set of sine waves added together. Since sine waves are easy to analyze in the context of electric circuits, this means we have a way of simplifying what would otherwise be a dauntingly complex problem: analyzing how circuits respond to non-sinusoidal waveforms. Time spent with a simple oscilloscope and spectrum analyzer viewing signal generator waveforms is also helpful in grasping how time-domain and frequency-domain representations relate.

- **Logic level voltages** – in any positive-logic circuit, a "high" state is ideally any ground-referenced voltage at or near the positive power supply rail voltage, and a "low" state is ideally any ground-referenced voltage at or near zero. However, the presence of noise, interference, and loading effects means logic circuits must be able to tolerate imperfect "high" and "low" voltage levels in order to reliably function in real-world scenarios. Thus, logic gate "families" are designed to operate such that "high" signals are defined as any ground-referenced voltage falling within a standardized *range* up to and including the positive rail potential, and "low" signals are defined as falling within a standardized *range* down to and including zero. These voltage requirements must be honored especially in cases where we need to interface two or more different families of logic gate, or gates powered by differing supply voltages, in the same circuit.

- **Jitter** – this is when a repetitive pulse signal does not repeat at *exact* intervals of time, but sometimes leads or lags by some amount of timing error.

- **Single-ended versus Differential signals** – single-ended signals are also known as *ground-referenced* or *unbalanced* signals, since their voltage values are always measured with respect to ground. Differential signals which are also known as *balanced* signals, by contrast, are voltages measured between two non-grounded conductors. This is but one more example of how the inherently differential nature of voltage can be challenging to grasp, there being no such thing as voltage existing at any single point in any network.

1.3 Recommendations for instructors

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

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

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

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

- **Outcome** – Interpret oscilloscope-measured signal characteristics

  Assessment – Determine rise and fall times based on given oscillographs; e.g. pose problems in the form of the "Measuring rise and fall times" Conceptual Reasoning question.

  Assessment – Determine whether set-up and/or hold time requirements are met in a given oscillograph; e.g. pose problems in the form of the "Evaluating set-up and hold times" Quantitative Reasoning question.

- **Outcome** – Diagnose measured signal integrity

  Assessment – Identify signal integrity problems and possible causes from given oscillographs; e.g. pose problems in the form of the "Pulse waveform oscillographs" Diagnostic Reasoning question.

- **Outcome** – Independent research

  Assessment – Locate a digital storage oscilloscope manual and properly interpret some of the information contained in that document.

Chapter 2

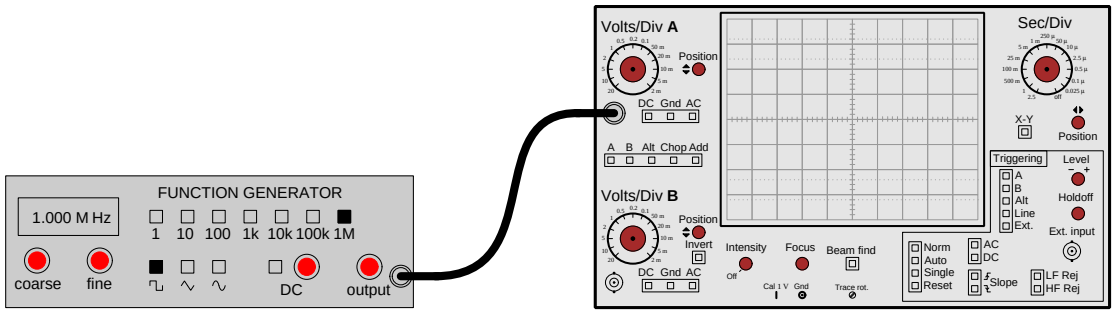
Case Tutorial

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.

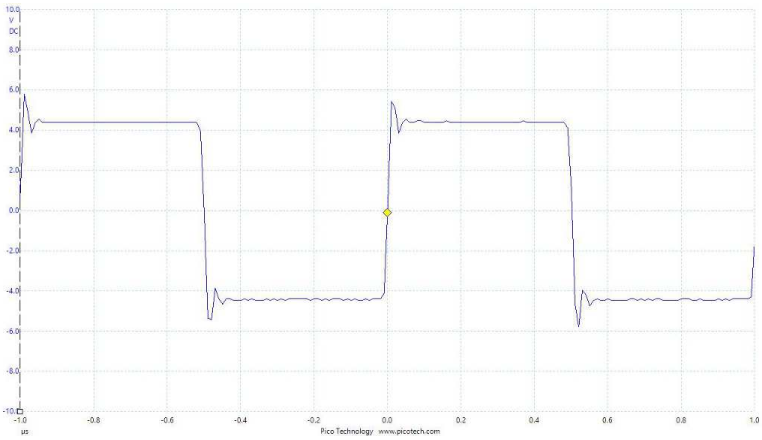
These examples also serve well as challenges following your reading of the other Tutorial(s) in this module – can you explain *why* the circuits behave as they do?

2.1 Oscilloscope, cable, and signal generator

In this experiment we connect the output of a signal generator directly to one of the input channels of an oscilloscope using a coaxial cable, and set the signal generator to output a square wave at approximately 1 MHz:



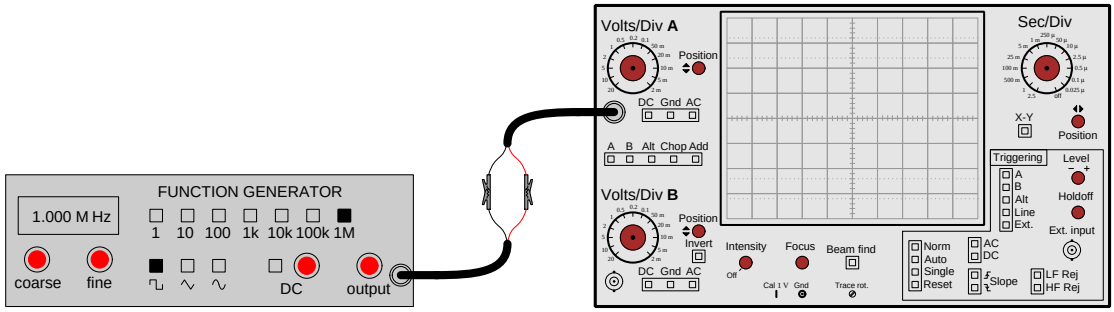
The resulting oscillograph[1] appears here:



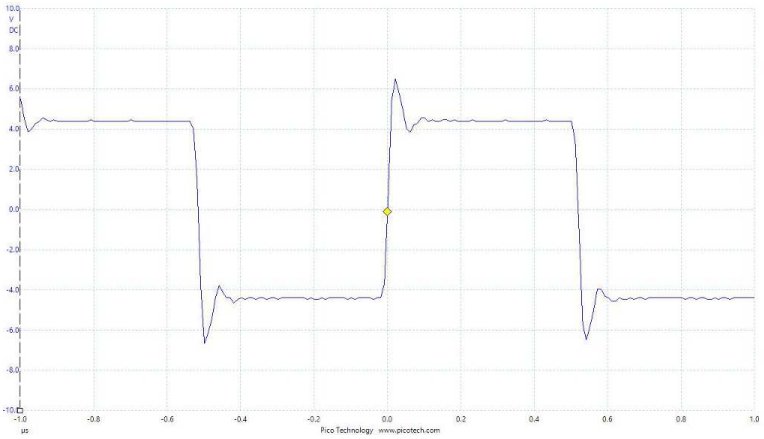
Note the various imperfections of this square wave.

[1] This was recorded using a Picoscope model 2204A

Next, the continuous coaxial cable was replaced with a pair of coaxial-to-alligator cables and the same square wave signal re-measured:

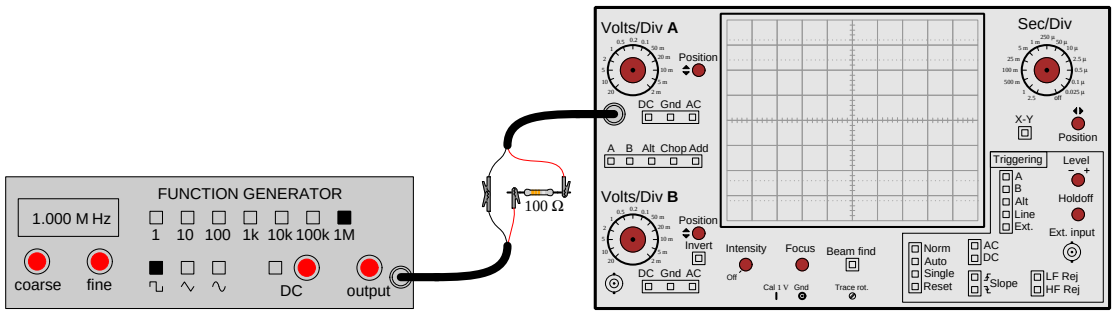


The resulting oscillograph appears here:

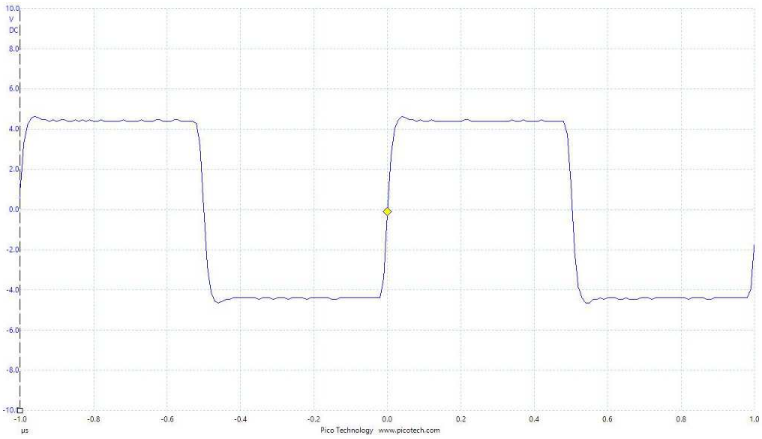


Note which of the imperfections worsened in this square wave from the use of different test cabling, and which remained unaffected.

Next, a 100 Ohm resistor was connected between the two red-wire leads of the coaxial-to-alligator cables, and the same square wave signal re-measured:

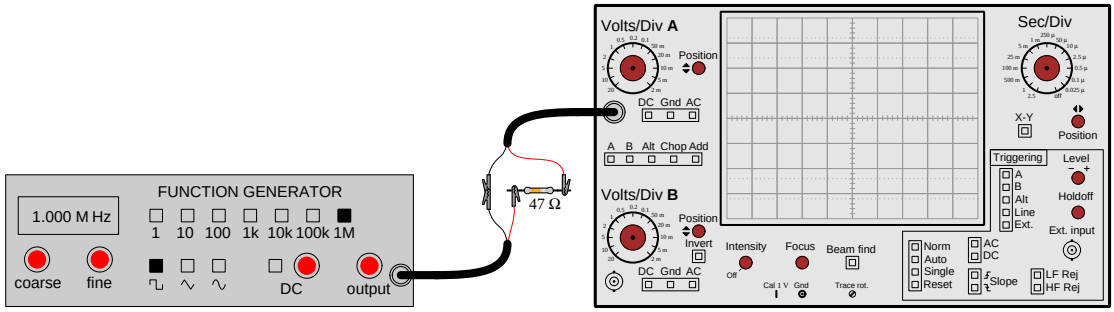


The resulting oscillograph appears here:

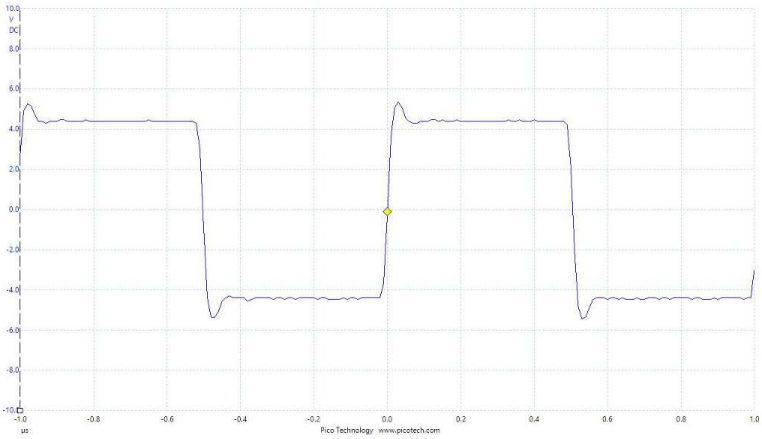


Note which of the imperfections improved in this square wave from the addition of the 100 Ohm resistor, and which remained unaffected.

Finally, a 47 Ohm resistor is substituted for the 100 Ohm resistor, and the same signal re-measured:



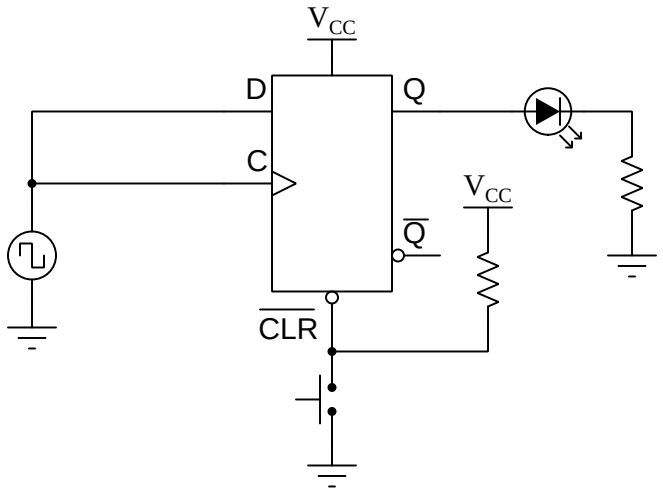
The resulting oscillograph appears here:



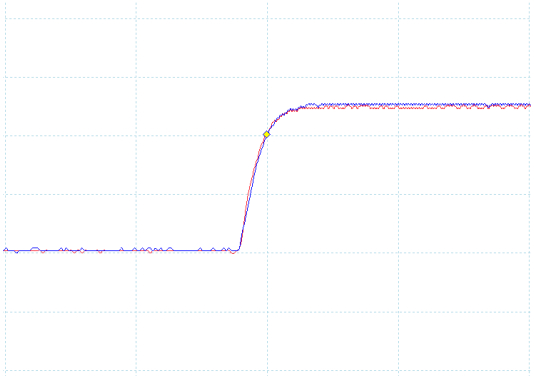
Note which of the imperfections improved in this square wave from the addition of the 100 Ohm resistor, and which remained unaffected.

2.2 Demonstration of set-up time

If we build the following circuit, we will find that the LED never energizes, despite the fact that the data input (D) goes "high" every time the clock input (C) sees a positive (rising) edge:

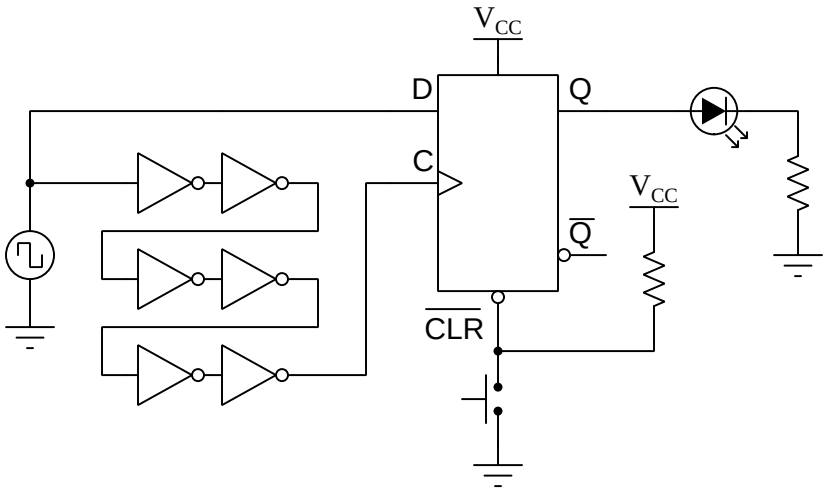


Looking at an oscilloscope's view of the common D and C signals, we see identical pulse edges because those two inputs are electrically common to each other. The blue trace shows the signal at the D input while the red trace shows the signal at the C input:

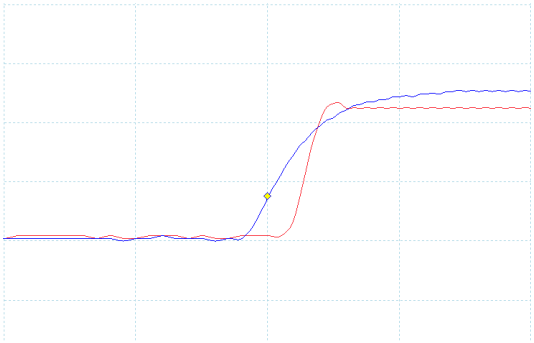


While it is tempting to think that the flip-flop ought to "clock in" the pulse signal's "high" state through to the Q output and turn on the LED because the D input goes high just in time for the rising edge of the clock pulse, simultaneous arrival is actually too late. What the flip-flop requires for reliable operation is for the logic state at the D input to be well-established *prior* to the C input receiving its rising-edge pulse. In other words, the desired state at the D input needs to be "set up" in advance of the clock or else it will miss its opportunity to be sampled by the flip-flop.

If we modify this circuit to intentionally delay the arrival of the clock pulse to the flip-flop's C input, the Q output will now go "high" and energize the LED. A simple way of creating a digital time-delay is to pass the clock signal through a set of logic gates, the propagation delay of each gate accumulating to make a larger delay the more gates the signal must pass through. Here we are using six inverter gates, the even number of gates being significant so that the C input does not assume the opposite state as the D input:



An examination of the signals at the D and C inputs shows us this new time delay, and serves to illustrate the importance of *set-up time* for the flip-flop. Once again, the data (D) input's signal is shown in blue while the clock (C) input's signal is shown in red:



With the six inverter gates delaying the clock pulse's arrival to the C input by about 50 nanoseconds[2], the "high" data state arrives at the D input 50 ns prior to the clock pulse's arrival. This advance time for the D input to "set up" in a high state gives the flip-flop ample opportunity to "clock in" that high state to the Q output. In the previous (non-delayed) example, there was zero set-up time for the high state, which means the state that always got "clocked in" to the Q output

[2]The oscilloscope's timebase in this second screenshot was set for 0.2 μs per division, and the amount of delay here is approximately one-quarter of a division. One-quarter of 0.2 μs is 0.05 μs or 50 ns.

was the D input's *low* state which was present at the D input long before the clock pulse arrived. The particular flip-flop used for this experiment was a 74LS74 which has a specified *minimum set-up time* of 20 nanoseconds.

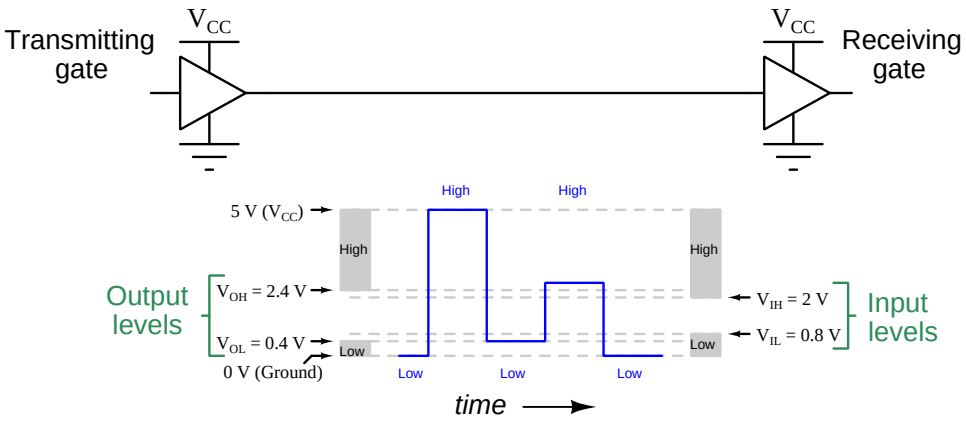
Chapter 3

Tutorial

3.1 Digital logic levels

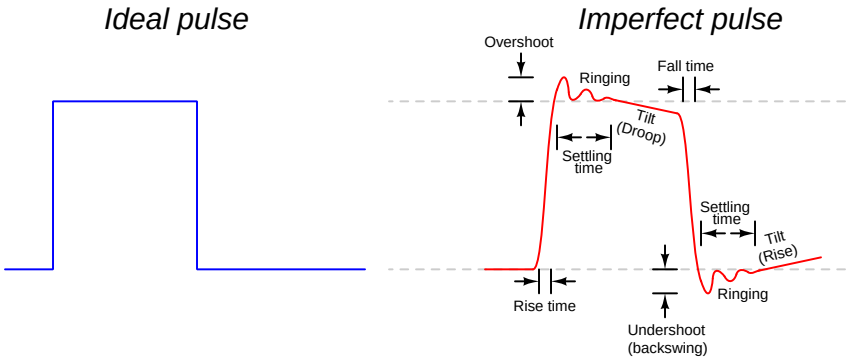
Digital logic gate circuits use voltage-based signals to represent and convey data. For example, the classic 5400/7400 series "TTL" logic family operates on a 5 Volt DC power supply, with gates interpreting an input signal voltage 2 Volts or greater as a "high" logic state and 0.8 Volts or less as a "low" logic state. The classic 4000 series "CMOS" logic family operating on the same 5 Volt supply voltage interprets any input signal 3.5 Volts or greater as a "high" state and 1.5 Volts or less as a "low" state. The newer 3.3 Volt version of TTL (called "Low-Voltage TTL, or LVTTL) respects the same logic levels as traditional 5-Volt TTL ($\geq$ 2 Volts = "high" and $\leq$ 0.8 Volts = "low") for reasons of compatibility.

Standardized threshold voltages defining "high" and "low" states makes digital signaling possible in a world where signal voltages may assume any value along a continuous (analog) range. Conservative design of logic circuits ensures *transmitted* voltage levels always surpass the minimum expectations for *received* signals, the difference between the transmitted and received thresholds called the *noise margin*. The following illustration shows a digital gate transmitting a low-high-low-high-low pulse sequence to a receiving gate, where the data is properly interpreted even though the voltage levels for "high" and "low" states vary from ideal (full rail) to barely acceptable:



3.2 Pulse imperfections

Despite the remarkable tolerance for imperfection we see with digital signaling, it is still possible for a pulse signal to become so distorted that a receiving gate fails to interpret it properly. It is a fact that *nature abhors a perfect square wave*, and so we should expect real pulses to deviate from the ideal square-edged profile shown in elementary timing diagrams. *Real* square wave signals often exhibit multiple imperfections[1], some of these so common they actually have names to describe them:



Additionally, pulse signals may have other signals superimposed on them such as DC bias, random noise, and AC power line frequencies. The causes of poor digital signal integrity are many:

- **Source problems**

  e.g. Slew rate limits of drivers

  e.g. Mechanical switch contact "bounce"

- **Parasitic effects**

  Stray inductance (L) along conductor length

  Stray capacitance (C) between conductors

  Reflected signals on improperly terminated transmission lines

- **External interference**

  "Crosstalk" from mutual inductance (L_M) between parallel conductors

  "Crosstalk" from mutual capacitance (C_M) between parallel conductors

  "Bounce" resulting from impedance of conductors sharing different load currents (e.g. +V and Ground busses)

[1] Quite often you will encounter pulse waveforms with some of these imperfections, but not all of them. For example, overshoot and undershoot without ringing, or ringing without droop.

We may apply both the *time-domain* perspective and the *frequency-domain* perspective to help us understand how these various forms of pulse distortion occur, though often one perspective proves more intuitive than another depending on the particular distortion.

From the time-domain perspective, we may explain phenomenon such as rise/fall time and ringing by appealing to well-known electrical network principles such as the "Ohm's Law" formulae for capacitance and inductance ($I = C\frac{dV}{dt}$ and $V = L\frac{dI}{dt}$, respectively), inverse-exponential growth and decay ($e^{-\frac{t}{\tau}}$, where τ is the "time-constant" equal to RC and $\frac{L}{R}$), and resonance ($f_r = \frac{1}{2\pi\sqrt{LC}}$). All of these principles stem from the ability of capacitance to store energy within an electric field and/or inductance to store energy within a magnetic field. Since both capacitance and inductance naturally exist in any array of conductors, this energy storage – and its unwanted effects – are to some degree inevitable. For example, the transistor output stage of a logic gate is necessarily limited in its ability to source and sink current (I), which means for any amount of capacitance (C) between digital wires or traces on a printed circuit board (PCB) the rate-of-rise for a voltage signal ($\frac{dV}{dt}$) will also be limited, helping to explain the rise and fall times of pulses. Furthermore, when resistance of conductors combines with stray capacitance and inductance, we see growth and decay curves also limiting rise and fall times, and contributing to "tilt" (droop and rise). Ringing is clearly a form of oscillation, resulting from the back-and-forth exchange of energy (i.e. resonance) between capacitance and inductance. In summary, waves distort due to the Law of Energy Conservation and the time required for energy to transfer from one form or location to another.

From the frequency-domain perspective, we may explain pulse distortion in terms of filtering, coupling, and device nonlinearities. Recall that any wave-shape is mathematically equivalent to an infinite series of sinusoids summed together, those sinusoids having particular amplitudes and frequencies unique to that wave-shape. For a square wave the *Fourier series* consists solely of odd harmonics[2], each one diminishing in amplitude by $\frac{1}{n}$ where n is the harmonic number and ω is the fundamental frequency:

$$\textbf{Square wave} = \sin\omega t + \frac{1}{3}\sin 3\omega t + \frac{1}{5}\sin 5\omega t + \frac{1}{7}\sin 7\omega t + \cdots + \frac{1}{n}\sin n\omega t$$

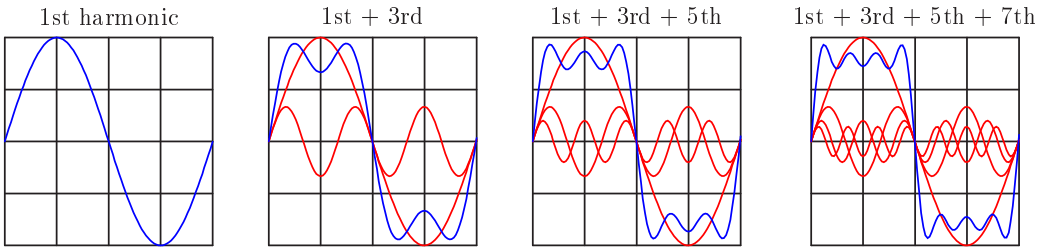
A *perfect* square wave requires an *infinite* number of these harmonics in order to achieve a square profile from sinusoidal (rounded) waves, which means attenuation or amplification or addition of new frequencies will result in the wave-shape no longer being square. Therefore, in order for a perfect square-edged digital pulse to propagate through a digital system untainted, the entire system would have to be *perfectly linear* in its characteristics, *perfectly shielded* from external signals, and have *infinite bandwidth* (i.e. no filtering of signal frequencies anywhere). We know this is impossible, as no transistor exhibits perfectly linear signal amplification, no signal path is perfectly shielded, and the parasitic effects of resistance, capacitance, and inductance along signal pathways and inside gates always conspire to filter some frequencies from others. In other words, component nonlinearities introduce new harmonics, parasitic effects attenuate others, and coupling with other signal sources introduces new frequencies (which may not even be harmonics of the original signal!). This explains why *nature abhors a perfect square wave*[3].

[2] A *harmonic* is a sinusoid having a frequency equal to some integer multiple of the wave's fundamental. For example, a 2 kHz square wave will have a fundamental frequency (i.e. 1st harmonic) of 2 kHz, a third harmonic of 6 kHz, a fifth harmonic of 10 kHz, etc.

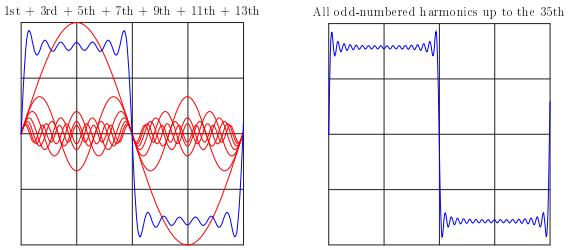
[3] To be fair, nature abhors the perfect preservation of *any* signal's wave-shape for the exact same reasons. We

At low frequencies these effects go unnoticed because they are so transitory. A 5 kHz clock pulse signal with rise and fall times of a few microseconds would still appear to be quite "square" because its entire period is 200 microseconds. However, at 5 MHz (still considered "slow" for modern digital circuitry) a single microsecond would exceed the entire period of the desired square wave!

Looking at a square wave from the perspective of its frequency spectrum (harmonics), we find that more than a few harmonics are necessary to make the wave reasonably square. Consider the following summation of sine waves following the Fourier series for a square wave ($\sin \omega t + \frac{1}{3} \sin 3\omega t + \frac{1}{5} \sin 5\omega t + \frac{1}{7} \sin 7\omega t + \cdots + \frac{1}{n} \sin n\omega t$), adding one harmonic (red) at a time to form a composite wave (blue). With each additional harmonic the composite wave looks more and more "square" but still exhibits slow rise and fall times – and also a lot of overshoot and ringing – even as far as the seventh harmonic:



Continuing the series up to the thirteenth and thirty-fifth harmonics, respectively, we see a continued approach to a true square-wave shape but still imperfections linger:



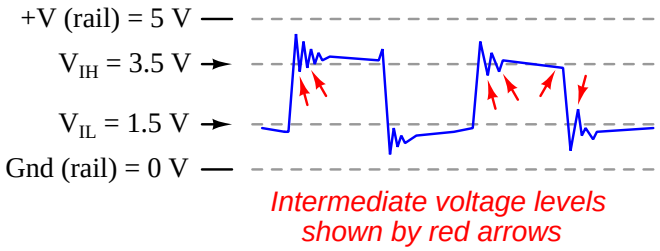
The lesson is clear: fast rise/fall times and flat-top profile defining a high-integrity digital pulse requires a large number of harmonics, and this means frequencies *much* higher than the fundamental. For example, the 35th harmonic of a 5 MHz square wave is 175 MHz. At that frequency a mere 1 nano-Henry of parasitic inductance along the length of a circuit board trace exhibits more than an

pick on square waves simply because they consist of an infinite series of harmonics and because their shape is easily identified in a time-domain display. In fact, this makes square waves well-suited to the oscillographic *testing* of electrical networks. Viewed in the time domain, a square wave's shape is very distinctive, and therefore any distortion – whether by the addition of new frequencies or the attenuation of existing harmonics, from the fundamental all the way to infinity – becomes easy to spot on an oscillograph. By contrast, subtle distortion of a sinusoid is much more difficult to identify by visual inspection in the time domain, because the curve of any given wave may not deviate in an obvious manner from the shape of a pure sine wave. Interestingly, when testing an electrical network for distortion using a spectrum analyzer, the sine wave becomes the ideal test waveform because a pure sine wave has a very distinctive spectral appearance: just one "peak" at the fundamental frequency and nothing more. Any deviation from the sinusoid shape results in additional peaks on the spectrum display, making distortion easy to spot in the frequency domain.

Ohm of series reactance ($X_L = 2\pi fL$), and a mere 1 pico-Farad of parasitic capacitance between circuit board traces offers less than 1000 Ohms of shunt (parallel) reactance ($X_C = \frac{1}{2\pi fC}$).

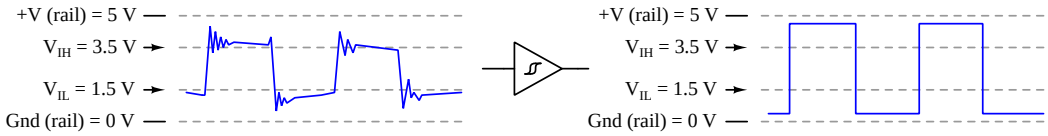
3.3 Consequences of distorted pulses

Whatever their cause(s), pulse signal distortions may cause problems in digital systems if sufficiently severe. Perhaps the most obvious effect of "analog" distortions on the interpretation of digital signals by receiving gates is incorrect logic state determinations resulting from fluctuations in signal voltage, referred to as *voltage uncertainty*. Consider this example of a highly distorted pulse signal overlaid on logic level thresholds for 5-Volt CMOS gates:



Digital gate circuits are designed to reliably interpret any voltage equal to or greater than V_{IH} as a logical "high" state, and any voltage equal to or less than V_{IL} as a logical "low", but voltages lying between the V_{IH} and V_{IL} thresholds are *indeterminate*. This means the gate may or may not function as designed with the input voltage at such a marginal level. In the example waveform shown, the receiving gate may behave erratically at all the points marked by red arrows.

Gates equipped with *Schmitt trigger* inputs fare better than ordinary logic gates when faced with marginal voltage levels because they tend to "latch" in the last-sensed logic state and will not switch unless and until the input signal crosses past the *other* state threshold voltage. In fact, Schmitt trigger buffers may be used to "clean up" corrupted pulse signals prior to the inputs of other digital logic elements requiring signals with high integrity, as shown in the following diagram:



However, this tolerance of marginal voltage levels comes at the price of speed: Schmitt trigger gates tend to propagate signals much more slowly than normal gates. This price is especially costly within high-speed digital circuits where time delays degrade performance. So it is with high-speed digital logic: the high speeds exacerbate pulse distortion effects, and what would otherwise be a practical solution to the problem of signal distortion (Schmitt triggers) unfortunately slows down the signal flow which makes high speed difficult to achieve.

One way to mitigate the *ringing* we see in the previous distorted pulse signals – which as we know is caused by inductive-capacitive resonance – is to intentionally insert resistance in series with the conductors. This added resistance dissipates energy stored in the inductance and capacitance rather than letting it freely exchange back and forth[4]. This usually leads to reduced ringing, reduced

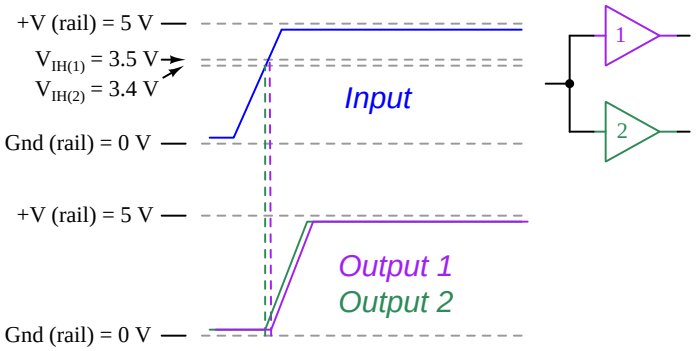
[4]This is essentially the same principle as a *shock absorber* added to a vehicle suspension system to dampen

over/undershoot, and shortened settling time. However, added signal line resistance tends to slow down the rise and fall times of the pulse which of course just degrades the signal in a different way.

A less-obvious problem caused by poor digital signal integrity is *inconsistent timing*, also known as *jitter* or *timing uncertainty*. Many causes of timing uncertainty (i.e. "jitter") exist. One potential source of jitter is instability in the clock pulse source, usually a crystal-based oscillator circuit. Another source of jitter is found in combinational logic circuits where the signal pathway switches between chains of gates having different total propagation delay times.

Even in the absence of unstable clock generators and varying network propagation times, we may experience timing inconsistencies within a digital system simply based on the fact that real pulses have finite rise and fall times. Since the interpretation of the next logic state at a gate's input terminal depends on that signal's voltage traversing a range of values deemed "indeterminate" to eventually cross a new threshold value, slow rise/fall times effectively add delay to the signal. Moreover, this added delay varies from gate to gate because each gate's effective high/low threshold values vary with manufacture as well as with operating conditions (e.g. supply voltage, temperature, etc.).

An illustration helps make this concept clearer. Consider the case of two logic gates, each receiving the same pulse signal, but having slightly different V_{IH} threshold values:



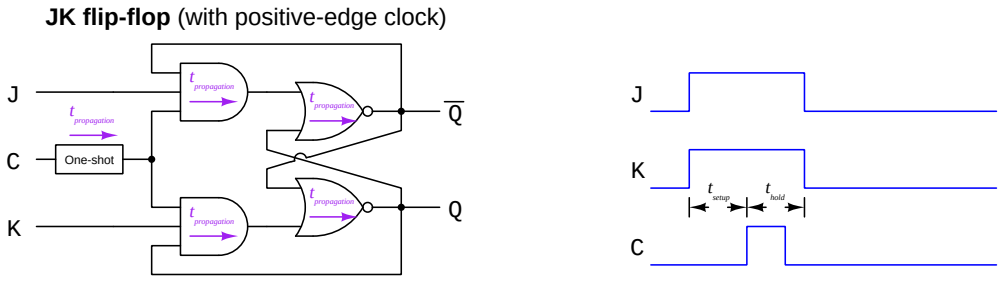
There is the obvious delay between the input signal's initial rise and each gates' output signal rise, but more than that there is a slight timing delay *between* the output signals of the two gates based solely on one having a slightly lower V_{IH} threshold than the other. Both the larger delay and the delay between the two gates' outputs will vary as a function of the input signal's rise time. No such timing discrepancy would exist had the pulse been a perfect square-wave.

To give just one example of how slow rise or fall times may adversely affect digital circuit operation, it is noteworthy that digital *counter* ICs always list maximum pulse rise/fall times for their clock signals. If one attempts to drive the clock input of a counter circuit using a pulse waveform that fails to rise or fall quickly enough, the counter may respond unpredictably[5].

oscillations resulting from driving on an uneven surface: the shock absorber acts like a mechanical *resistor* to the vehicle's mass (*parasitic inductance*) and spring elasticity (*parasitic capacitance*). Without a shock absorber, the vehicle will continue to undulate long after striking a bump in the road.

[5] For example, the CD4029 CMOS presettable up/down counter IC is a four-bit synchronous counter with BCD and binary counting capability, having an advertised maximum rise/fall time of 15 microseconds. I once worked on a prototype digital system using these counters where the clock pulse driving them had a rise time of *100* microseconds. The result was that the counters would often "double-count" from 7 to 9 (skipping 8), and sometimes "double-count"

One of the ways in which timing uncertainties may cause problems in high-speed digital systems is when the early or late arrival of a pulse creates a condition of insufficient *set-up time* and/or *hold time* for clock-driven functions such as flip-flops. Set-up time is the minimum amount of time a logic circuit's input states must remain stable *before* the next clock pulse arrives[6], while hold time is the minimum amount of time those states must remain stable *after* the clock pulse occurs. Set-up and hold time requirements stem from the propagation delays of the logic gates within flip-flops and related digital circuits, necessary to ensure the input signals and clock signal don't "miss" each other:



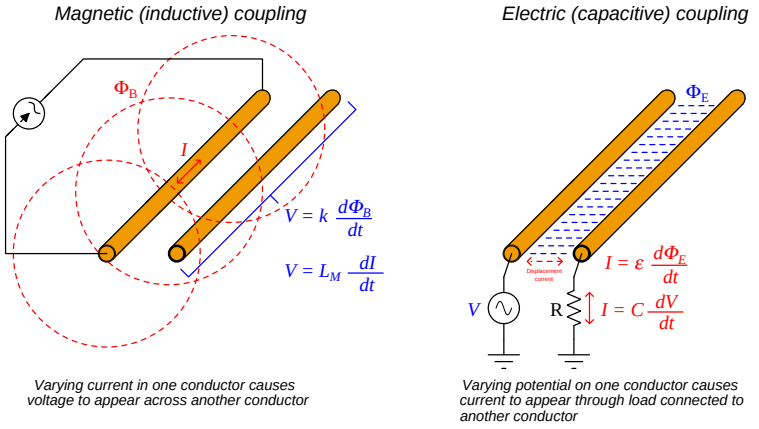
between 9 and 1 in decade mode (skipping 0). If the rise time grew slower the counters would often skip other count values as well, but in what appeared to be a completely random fashion. The problem was fixed by inserting a Schmitt-trigger inverter gate between the slow-rise clocking signal and the counters' clock input lines to hasten the pulse waveform's edges.

[6] An analogy for set-up time is intentionally arriving to a bus or train stop a few minutes in advance of the bus's or train's expected arrival, to ensure you will not miss your ride.

3.4 Radiated interference

So far we have only explored problems resulting from a signal's natural loss of integrity as it moves from one point to another in a circuit. Another mechanism of signal degradation is *interference* from signals traveling through neighboring conductors, through the interaction of electric and/or magnetic fields between those conductors. This problem is especially prevalent in densely-populated circuit boards and within multi-conductor cables where the conductors lie in close proximity to one another, but it may manifest between any conductors where the radiated fields are sufficiently strong.

Signals conveyed by one conductor (the "aggressor") may unintentionally link to a nearby conductor (the "victim") in two distinct ways: by magnetic and/or by electric "coupling". The magnetic field generated by one current-carrying conductor may "link" to an adjacent conductor and induce a voltage there. Similarly, the electric field generated by one energized conductor may "link" to an adjacent conductor and induce a current through any available connection to ground. The principles include Faraday's Law of Electromagnetic Induction ($V = N\frac{d\Phi}{dt}$) as well as the "Ohm's Law" formulae for inductance and capacitance ($V = L\frac{dI}{dt}$ and $I = C\frac{dV}{dt}$, respectively):



As the formulae suggest, the degree of signal *crosstalk* between conductors varies as a direct function of rates-of-change of both voltage and current. This fact makes high-speed digital signaling challenging because as data rates increase, so must the rates at which voltages and currents rise and fall.

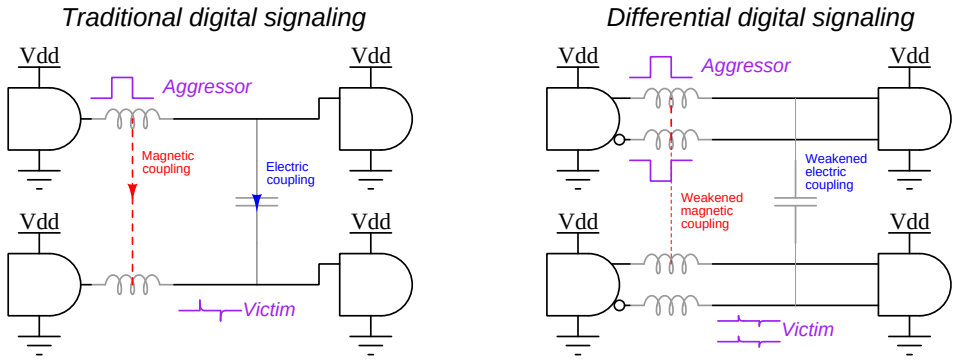
Not only does conductor-to-conductor crosstalk worsen as rise/fall shorten, but more energy radiates away from the circuit in the form of *electromagnetic waves* (i.e. radio emissions) as $\frac{dV}{dt}$ and $\frac{dI}{dt}$ increase. Not only may these emissions induce noise in more distant parts of the digital system, but they may become strong enough to induce noise in *other systems* located some distance from the offending circuit. Electromagnetic interference (EMI) may be mitigated by encasing the high-speed circuit in metal foil and/or by using a suitable metal enclosure rather than plastic, but these measures add cost and complexity to the final assembly.

Guidelines published by the Federal Communications Commission (FCC) in the United States place limits on such *radiated emissions*, and any manufacturer of electronic devices must test their products for compliance against these standards.

One mitigation technique for radiated interference is to run aggressor and victim conductors *perpendicular* to each other rather than parallel. Perpendicular crossing not only minimizes electric-field coupling by minimizing the area of conductor overlap, but it also minimizes magnetic-field coupling by setting the aggressor's magnetic field parallel to the victim conductor where it cannot induce a voltage. This strategy may be impractical on many printed circuit boards where multi-bit digital busses must run between shared components, and it is flatly impossible for multi-conductor cables where the conductors must run parallel to each other. Mitigation techniques developed for cables include *shielding* individual conductors or pairs of conductors by wrapping them with grounded wire mesh or metal foil to prevent those wires' electric fields from spreading to other wires, and *twisting* wire pairs so that any external magnetic field's inductive effect will tend to cancel out within each of the twisted pairs. Furthermore, multi-pair cables such as Cat-5 use unique twist rates for each wire pair, so as to help avoid magnetic coupling from one pair to another within the same cable.

One clever solution to both magnetic and electric crosstalk is the use of *differential signaling*. Traditional digital signaling uses ground-referenced voltages on single conductors, but differential signaling uses dedicated pairs of conductors for each digital channel, with special logic gates having differential input and output terminal pairs.

The advantages of differential signaling extend to both the victim and the aggressor. From the victim perspective, conductor pairs in close proximity experience practically the same magnetic and electric field strengths from neighboring conductors, and therefore the amount of interference induced on each conductor is about the same. However, since the digital data is represented by a *difference* in voltage between those conductors, having the same amount of noise induced on each conductor makes the noise cancel itself out at the receiving gate. From the point of view of the aggressor, differential lines radiate less field strength because the magnetic fields produced by parallel conductor pairs cancel each other (i.e. currents flowing at the same intensity but in opposite directions) and the electric fields exist mostly just between those conductors. For both reasons the amount of crosstalk is far less for differential signal lines than for traditional (ground-referenced) signal lines:

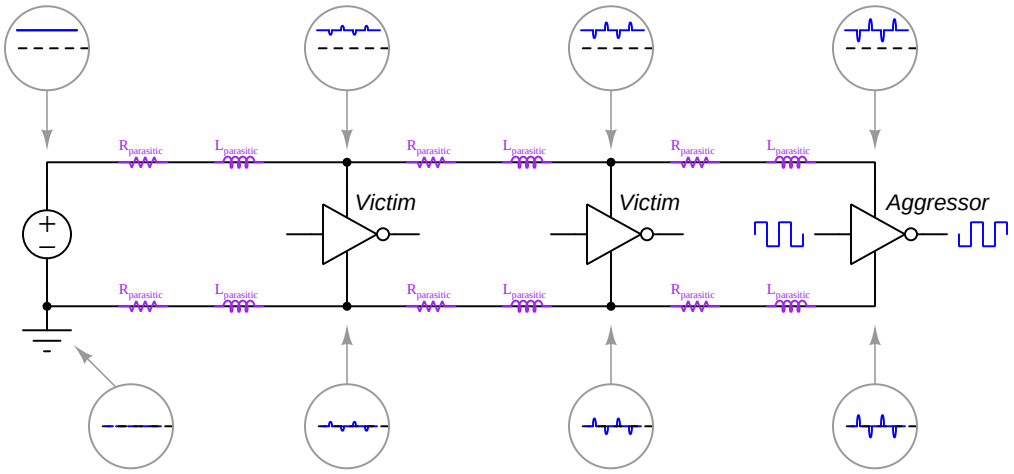


An obvious disadvantage of differential signaling is that it requires special digital gates equipped with differential inputs and outputs, and of course it also doubles the necessary number of gate terminals and signal conductors in the circuit.

3.5 Conducted interference

Another form of component-to-component interference occurs when multiple devices such as digital logic gates receive their DC power from the same set of power supply conductors, whether those conductors are discrete wires or traces on a printed circuit board. Since lengths of metallic wire or PCB traces exhibit parasitic resistance and parasitic inductance, these properties will cause voltage to be dropped along those conductors' lengths as the various devices draw current through them. Of these two parasitic effects, inductance is generally the more significant. When the devices' DC supply current pulses on and off at high frequency, the conductors' inductance presents a reactance proportional to signal frequency ($X_L = 2\pi fL$), and at the very high frequencies typical for modern digital circuits the reactance of a typical PCB trace or even short length of wire may be sufficient to cause enough voltage "sag" at the device's DC power terminals to compromise its performance.

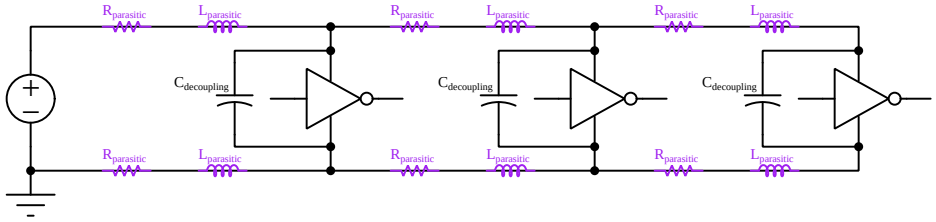
In the following illustration we see a simplified schematic diagram with three digital logic gates being fed DC power from the same two conductors. The gate farthest from the DC source is pulsing at a high frequency, and this causes the electrical potentials of both the +V and Ground conductors to *bounce* with each rising and falling edge of the digital pulse signal. Circled oscillographs show those "rail" voltages with reference to the DC source's ground point, the gate closest to the DC source receiving the most "stable" power because it has the least amount of parasitic resistance and inductance between it and the source:



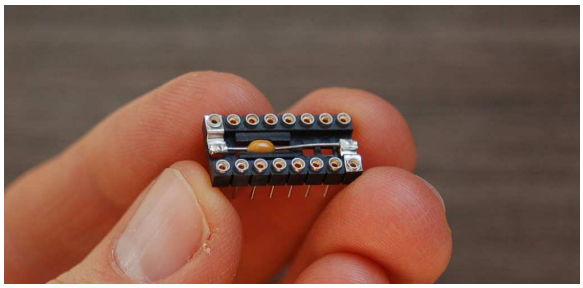
A common term for this phenomenon is *ground bounce* because it makes the electrical potential of the ground terminal on each victim device "bounce" with every pulse of current through the ground rail conductor. The same phenomenon on the +V power supply rail is sometimes referred to as V_{DD} *bounce* because "V_{DD}" is the common label for the positive rail feeding CMOS logic gates. Often you will encounter "ground bounce" used as a general term for both.

An interesting feature of ground bounce is that the aggressor may also be a victim of its own actions, due to the "bounce" effect being worst at its power supply terminals.

Ground bounce may be mitigated in several different ways: (1) placing loads with the greatest current rates-of-change ($\frac{dI}{dt}$) closest to the DC power supply, (2) minimizing the parasitic inductance along shared power supply conductors, (3) eliminating all shared power conductors by giving each load its own pair of conductors back to the power source terminals, and/or (4) adding *decoupling* capacitance as close to the load terminals as possible. Conductor inductance is best minimized simply by keeping the power supply conductors as short as possible. "Decoupling" consists of connecting capacitors directly in parallel with every device's DC power terminals in an effort to stabilize DC voltage. Decoupling is a good practice in any system using integrated circuits (ICs), whether digital or analog, and it is best to use capacitors with low equivalent series resistance (ESR) so they will be able to source high levels of transient current if necessary.



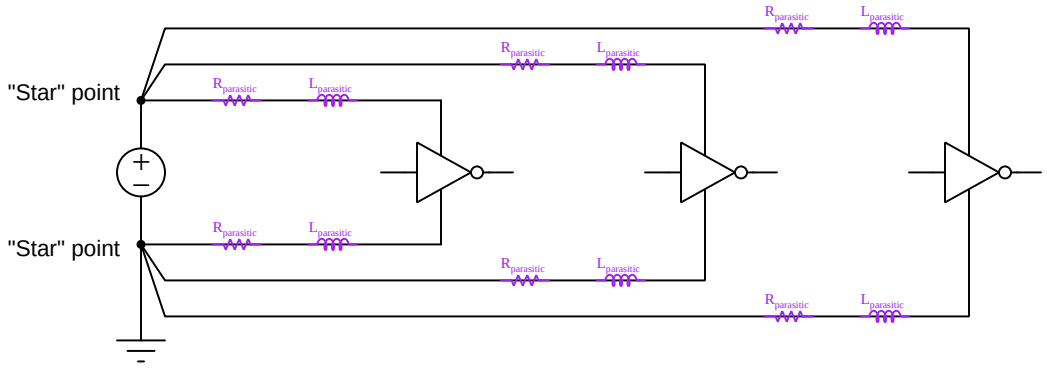
In this next photograph we see a convenient strategy for decoupling logic circuits: a 16-pin IC socket with a built-in decoupling capacitor connected between pins 16 and 8 (typically V_{DD} and V_{SS}, respectively, for a 16-pin digital logic IC). Keeping the capacitor's leads as short as possible minimizes parasitic inductance there, maximizing its ability to source current to the IC when needed:



For very high-frequency printed circuit boards a good practice for DC power distribution is to devote two or more of the layers in a multi-layer board as *power planes*: sheets of copper sandwiched between layers of insulating board material acting as "busses" for each rail of the DC power source. Not only does this make DC power easy for each device to access (simply install a "via" which is an inter-layer connector between the desired power plane layer and the device power terminal) but the natural capacitance existing between power planes separated by board insulation contributes to the decoupling offered by discrete capacitors installed adjacent to each device.

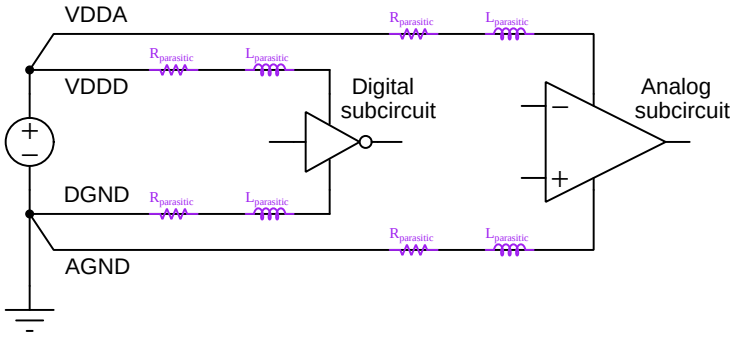
Providing each load with its own dedicated power conductors is commonly known as *star-point grounding*, which is a misnomer because it applies equally well to non-ground power conductors as it does to ground. An example is shown below:

Star-point power distribution



With each load having its own power supply conductors, there is much less[7] potential for conducted interference between the loads than when power conductors are shared. In mixed-signal circuits where both digital and analog signals coexist on the same printed circuit board(s), implementing star-point grounding means separate *digital* and *analog* power supply rails as shown below:

Digital versus analog power rails



VDDD simply means V_{DD} for the digital load(s), while VDDA means V_{DD} for the analog load(s). Similarly, DGND means digital ground while AGND means analog ground.

[7] The common power source still has its own internal (Thévenin) impedance which cannot be entirely eliminated, which is why star-point power distribution cannot entirely eliminate conducted interference between loads.

3.6 Diagnostics

Diagnostic tools for testing signal integrity include the following:

- Logic analyzers
- Oscilloscopes

  Mixed-signal oscilloscopes (MSOs)

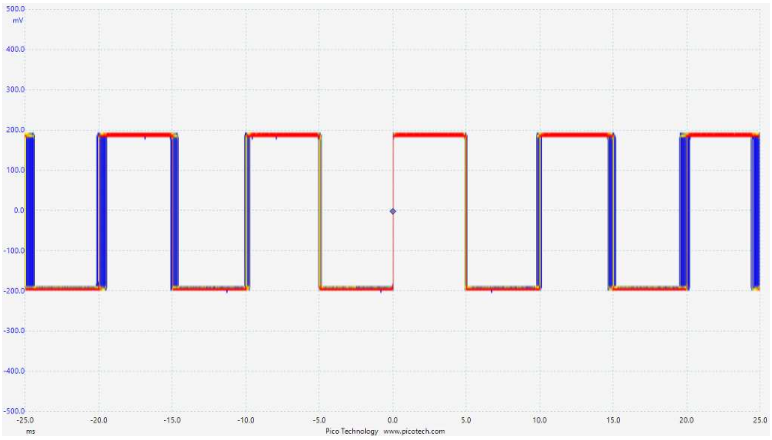
  Digital phosphor oscilloscopes (DPOs)
- Time-delay reflectometers (TDRs)

Logic analyzers display digital signal timing diagrams, much like an oscilloscope displays analog signal voltages in the time domain, except that logic analyzers interpret the signals only as discrete states (high and low) and they usually display *several* simultaneous traces compared to the few shown by most oscilloscopes.

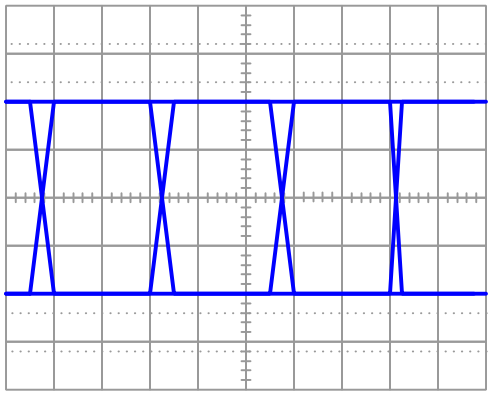
Mixed-signal oscilloscopes blend the capabilities of standard oscilloscopes with logic analyzers, allowing the user to examine digital signals two different ways simultaneously on the same screen. As with logic analyzers, the number of digital input channels is usually quite large, in order to facilitate simultaneous monitoring of multiple bit-lines within a parallel digital bus. Triggering is usually available on either analog or digital channels, allowing the user to synchronize the oscilloscope's display to any arbitrary reference waveform being measured.

Digital phosphor oscilloscopes are special variations of digital oscilloscopes designed to mimic the behavior of old analog CRT-based oscilloscopes where the relative brightness of the trace represented its frequency of re-occurrence, useful when measuring a repetitive pulse signal to see if successive pulses drift either in amplitude, timing, or both. Legacy analog oscilloscopes with phosphor-based luminescent displays would naturally show a brighter trace where the waveform retraces the same path multiple times, and show a dimmer trace where the waveform only occasionally follows that path. Digital phosphor oscilloscopes mimic that legacy functionality and actually do a step further by *colorizing* the frequency of re-occurrence so that the most frequently retraced paths show as a completely different color compared to seldom-traced paths.

This colorized-occurrence feature is so popular that even some entry-level digital oscilloscopes now offer it. An example is this screenshot from a Picoscope model 2204A where the colorizing function is called *persistence mode*. Red indicates a frequently-traced path, while blue indicates a seldom-traced path. The following screenshot was generated using this model of oscilloscope set to persistence mode, measuring a square wave with a shifting frequency. Note how the waveform is crisp and red near the center of the screen where the triggering cursor is located, but shows more blue regions in the rising and falling edges (shifting position from the trigger point due to the waveform's period changing from sweep to sweep):

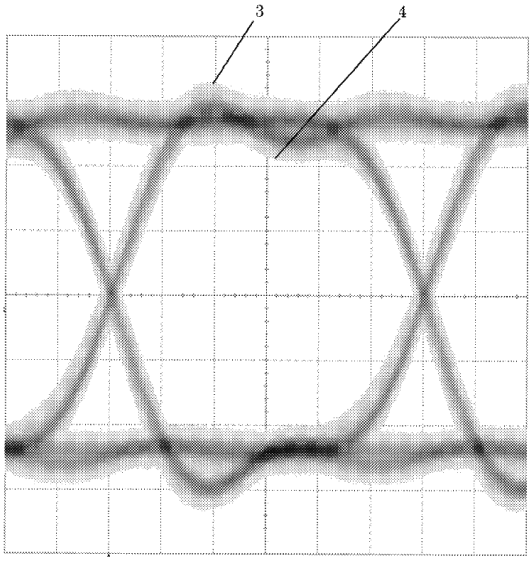


Digital signals in high-speed circuits appear as pulse sequences of high (1) and low (0) states when displayed in the time domain. An alternating sequence of 1 and 0 states would be a simple square wave, but real digital data typically doesn't repeat such simple patterns over and over. Instead, the pattern of high and low states often appears random without any knowledge or context of what those pulses represent in the system at the time of measurement. As such, an oscilloscope configured to repeatedly display a serial digital pulse stream will show something that looks like this:



These are clearly square (or trapezoidal) wave pulses that overlap enough times so that only the high-to-low and low-to-high edges are distinct. At first, this sort of display looks useless because

we really cannot discern individual high or low logic states due to the overlapping of repeated traces. However, when displayed on a digital phosphor oscilloscope, variations in amplitude and timing create what is commonly known as an *eye diagram* which is a good visual representation of uncertainty and therefore of signal integrity. The following image of a greyscale digital phosphor oscillograph was taken from Martin Miller's patent "Noise Analysis To Reveal Jitter And Crosstalk's Effect On Signal Integrity"[8] as part of his explanation of "prior art" (i.e. conventional state-of-the-art) technology for signal integrity analysis:



As with any oscillograph of a digital pulse signal, we can easily identify such imperfections as ringing, overshoot, undershoot, settling time, rise time, and fall time from this image. The added benefit granted by the digital phosphor technology is being able to discern the degree of variation in amplitude and time for successive pulses. For example, the labels 3 and 4 in Miller's image show the highest and lowest amplitude points captured for the "high" portion of the pulse signal over a collection of 5 million sampled pulses, representing a peak-to-peak variance of more than one division on the oscilloscope's vertical axis. Similarly, one could quantify the amount of jitter in this signal stream by measuring the *width* of any rising or falling edge. Identifying the amplitudes and times of these "outlier" signals allows the user to determine worst-case limits for signal integrity in this system[9]. The larger the area inside the "eye" (the "hole"), the greater the signal's integrity.

[8] Miller, Martin T. *US Patent 9,843,402*, "Noise Analysis To Reveal Jitter And Crosstalk's Effect On Signal Integrity", application 4 October 2016, patent granted 12 December 2017. This particular eye diagram results from *5 million* overlaid pulses, the re-occurrence frequency represented by the density of the grey pixels.

[9] A helpful analogy is to picture a city intersection on a snowy day. As vehicles navigate through the intersection, their tires leave tracks in the snow which remain long after each vehicle passes through. After much traffic has passed through the intersection, the culmination of all tracks indicate where most vehicles have driven as well as how far from center-lane the farthest outlying vehicle path was.

Time-domain reflectometers are a completely different diagnostic tool for signal-integrity troubleshooting. Unlike logic analyzers and oscilloscopes which passively measure signals, a TDR actively outputs pulses and then measures the reflected energy of those pulses to determine the high-speed characteristics of the signal networks. With the fast clock speeds and short rise/fall times of modern digital pulse signals, many data conductors in digital systems are long enough to function as transmission lines[10], and most of these are left unterminated in order to avoid excessive signal energy losses. However, this means some of the signal energy will reflect off the far end of the line rather than be absorbed entirely by the receiving gate(s), and if these reflected signals are strong enough and have sufficient duration they will interfere with new data transmitted in succession.

[10] In other words, the frequency of the pulse signal is high enough that the signal does not travel down the full length of the line to arrive at its destination before the next data bit is transmitted. If not properly terminated (with a resistance equal in value to the *surge impedance* of the line), some or all of that pulse energy will "echo" off the receiving end of the line and return back toward the source.

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 Signal integrity in wireless versus cable-based telephony

The problem of signal integrity and signal degradation is not limited to digital circuitry, nor is it particularly novel. From the earliest days of electrical-based communication, researchers, engineers, and technicians all dealt with various forms of this problem.

John Ambrose Fleming was an early researcher and prolific writer on various electrical and early electronic topics, including communication systems. In his 1910 book *The Principles of Electric Wave Telegraphy and Telephony* Fleming discusses the limitations of communicating speech signals over long distances via wire cable, compared to transmitting them through space[1]. Fleming comments here on the problem of audio signal degradation through telephone cables, and contrasts it against the use of wireless (i.e. radio) to communicate human speech:

> Although, therefore, wireless telephony has not attained the range reached by wireless telegraphy, nevertheless there is evidence that it has been conducted over distances of 200 miles or so with considerable success. One feature of importance in connection with radiotelephony is that there is no distortion of the wave form with distance. In ordinary telephony with wires, this distortion, which is specially marked in the case of circuits having large capacity, such as submarine cables, and the distorting power of the circuit impose a very serious limit upon the range of telephony. Briefly speaking, the reason for this is as follows:
>
> If we have a conductor with resistance R per unit of length, capacity C per unit of length, inductance L per unit of length, and dielectric conductance K per unit of length, then from the fundamental equations for the propagation of electrical disturbance along such a circuit (see Chap. IV., 1), it can be shown that the velocity of propagation of a periodic electric disturbance along the cable is a function of the frequency, the greater the frequency the less being the velocity of propagation. When articulate speech is being made against a telephone diaphragm, we have already seen that the wave form representing that articulate sound is a complex single valued curve, which in accordance with the theorem of Fourier can be analyzed into the sum of a number of simple constituent sine curves of different amplitudes differing in phase. These different harmonic constituents are propagated through the cable with different velocities and attenuated at different rates, so that when they are synthesized at the other end by the receiving telephone and the ear, the wave form which is reproduced is not an exact copy of the sound originally transmitted. In other words, we may say that the received sound is `[page 863]`
>
> a caricature of the sound made in the transmitting microphone. If the distortion is not too great, the ear at the receiving telephone is able to reconstruct or guess the sound at the other end creating it, just as we can recognize the original in a caricature of the human face if the caricaturing is confined within certain limits. Or to use another simile, in ordinary handwriting probably no single letter is quite correctly formed. If, however,

[1] In Fleming's time it was widely believed that empty space was not empty at all, but in fact was filled with a mysterious substance called *aether*. The reasoning was that since electromagnetic radiation possessed wave-like characteristics, these waves required a substance to travel through just as waves in the ocean require water as their medium. The aether theory of electromagnetic waves was later disproved, but for the context of this discussion its accuracy is irrelevant.

> the departure from perfect correctness is not too great, practice enables us to guess the word which is signified. The ear is therefore in this way able to recognize as a certain articulate sound, a sound which is not precisely like it. If, however, the distortion of the wave form has proceeded beyond certain limits, the ear is no longer able to recognize the origin of the sound heard. Hence a serious limit is imposed upon telephony through the ordinary submarine cable.

Note how Fleming explains the dominant cause of speech distortion through telephone cables, in terms of a speech signal's frequency composition (i.e. being comprised of a multitude of sinusoidal waves of differing frequency and amplitude). Wire cables do not convey all frequencies with equal efficiency, and so the result at the receiving end of the cable is a waveform with a different spectral composition than what was generated at the transmitting end. This is analogous to square-wave clock pulses in a digital circuit losing integrity as they encounter parasitic resistance, inductance, and capacitance along the way. What began as clear speech will be heard as muddled sounds at the receiving end of the telephone cable, and what began as a "clean" square wave will become tainted with ringing, overshoot, undershoot, tilt, and other defects upon receipt at the input(s) of the destination gate(s).

Interestingly, Fleming describes a technique to mitigate some of the signal degradation by installing additional inductance at various locations along a telephone line:

> It can be shown from the theory of such a cable, that if the relation between the four constant quantities R, L, C, and K is such that if $\frac{R}{L} = \frac{K}{C}$ then the cable will possess no distortion. In most ordinary cables the inductance is too small to fulfil this relation, but by the addition of inductance coils inserted at certain regular intervals in the cable, Pupin has been able to improve considerably telephonic speech through cables and long aerial lines. Such a cable is called a *loaded* cable. In the case, however, of wave transmission through the aether there is no distortion, because electromagnetic waves of all wave lengths travel through the aether at precisely the same speed, namely with the velocity of light. Accordingly, whatever may be the wave form of the wave which leaves the transmitting antenna, it will always preserve that same wave form although it may be attenuated or weakened by the diffusion of the energy over a large area. [page 864]

Expressed in more modern terms, radio communication enjoys much greater *bandwidth* than cable-based communication.

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 TTL logic levels

Logic gates need to reliably communicate with each other, the voltage levels output by the "driving" gate being compatible with the input voltage levels of the "receiving" gate. These specifications are given as follows:

- V_{OH} = minimum voltage output by a gate in the "high" (1) state
- V_{OL} = maximum voltage output by a gate in the "low" (0) state
- V_{IH} = minimum voltage received by a gate to be interpreted as a "high" (1) state
- V_{IL} = maximum voltage received by a gate to be interpreted as a "low" (0) state

Classic "LS" family of TTL gate circuits using bipolar transistors operated on a 5 Volt DC power supply, outputting at least 2.4 Volts in the "high" state and 0.4 Volts in the "low" state. The same logic family would accept any input voltage signal greater than 2.0 Volts as a "high" and less than 0.8 Volts as a "low". The amount of *noise margin* is the difference between the guaranteed output voltage levels and the acceptable input voltage levels. Represented in table form for 5-Volt TTL gates:

Logic state	**Guaranteed output**	**Acceptable input**	**Noise margin**
High	$V_{OH} = 2.4$ Volts min.	$V_{IH} = 2.0$ Volts min.	$2.4 - 2.0 = 0.4$ Volts
Low	$V_{OL} = 0.4$ Volts max.	$V_{IL} = 0.8$ Volts max.	$0.8 - 0.4 = 0.4$ Volts

If an ordinary digital logic gate receives an input voltage signal lying somewhere between the threshold values of V_{IH} and V_{IL}, its output state will be unpredictable. The gate may "assume" either a high or a low state, or worse yet may try to process that signal in an analog fashion, generating an output voltage level below V_{OH} and above V_{OL}, thus propagating the problem to the next logic gate(s). For this reason it is very important to design logic circuits to avoid voltage levels below V_{IH} and above V_{IL}.

Classic TTL logic IC part numbers begin with either 54 (military-grade) or 74 (commercial grade), but not all 54- or 74- series ICs are of traditional TTL (bipolar transistor) design. For example, the "HC" series of 54- and 74-numbered ICs utilize MOSFETs rather than bipolar transistors, but are otherwise designed to be pin-for-pin compatible with classic TTL logic ICs. These "high-speed CMOS" ICs are designed to function nearly identically to their bipolar-transistor counterparts, but with much lower current requirements and a slightly wider power supply range (2 to 6 Volts typical). Like classic 4000-series CMOS logic ICs, this means the minimum and maximum acceptable voltage levels for 54HC- or 74HC- series ICs "high" and "low" signals vary with supply voltage. By contrast, since classic TTL digital logic ICs are constrained to a very narrow range of DC power supply voltage (typically 4.75 to 5.25 Volts) their acceptable voltage levels for "high" and "low" signals is correspondingly fixed.

5.2 CMOS logic levels

Logic gates need to reliably communicate with each other, the voltage levels output by the "driving" gate being compatible with the input voltage levels of the "receiving" gate. These specifications are given as follows:

- V_{OH} = minimum voltage output by a gate in the "high" (1) state
- V_{OL} = maximum voltage output by a gate in the "low" (0) state
- V_{IH} = minimum voltage received by a gate to be interpreted as a "high" (1) state
- V_{IL} = maximum voltage received by a gate to be interpreted as a "low" (0) state

Classic "CD" family of CMOS gate circuits using complementary MOSFET transistors operating on a 5 Volt DC power supply has much wider greater noise margins than "LS" series TTL, outputting at least 4.44 Volts in the "high" state and 0.5 Volts in the "low" state, while accepting any input voltage signal greater than 3.5 Volts as a "high" and less than 1.5 Volts as a "low". Represented in table form for 5-Volt CMOS gates:

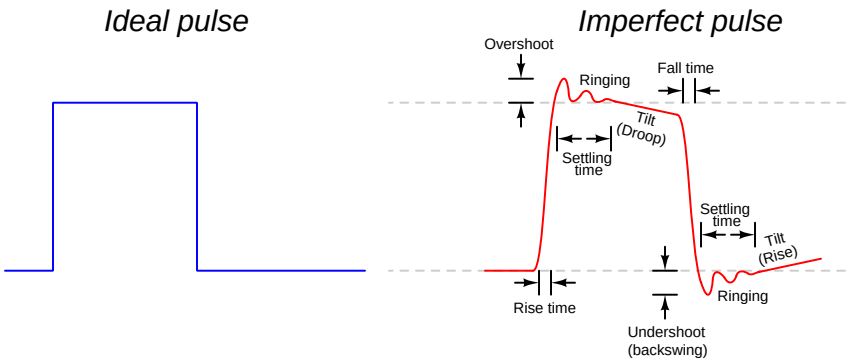
Logic state	**Guaranteed output**	**Acceptable input**	**Noise margin**
High	$V_{OH} = 4.44$ Volts min.	$V_{IH} = 3.5$ Volts min.	$4.44 - 3.5 = 0.94$ Volts
Low	$V_{OL} = 0.5$ Volts max.	$V_{IL} = 1.5$ Volts max.	$1.5 - 0.5 = 1.0$ Volt

CD-family CMOS logic was not limited to a 5 Volt DC power supply as was the LS (TTL) bipolar family of logic gates. Typical DC power supply limits ranged from 3 Volts to 18 Volts, with acceptable input voltage levels varying as a function of power supply voltage. Output voltage levels for a CD-family logic gate also followed power supply voltage, typically within 0.5 Volts of the power supply rails.

If an ordinary digital logic gate receives an input voltage signal lying somewhere between the threshold values of V_{IH} and V_{IL}, its output state will be unpredictable. The gate may "assume" either a high or a low state, or worse yet may try to process that signal in an analog fashion, generating an output voltage level below V_{OH} and above V_{OL}, thus propagating the problem to the next logic gate(s). For this reason it is very important to design logic circuits to avoid voltage levels below V_{IH} and above V_{IL}.

5.3 Digital pulse criteria

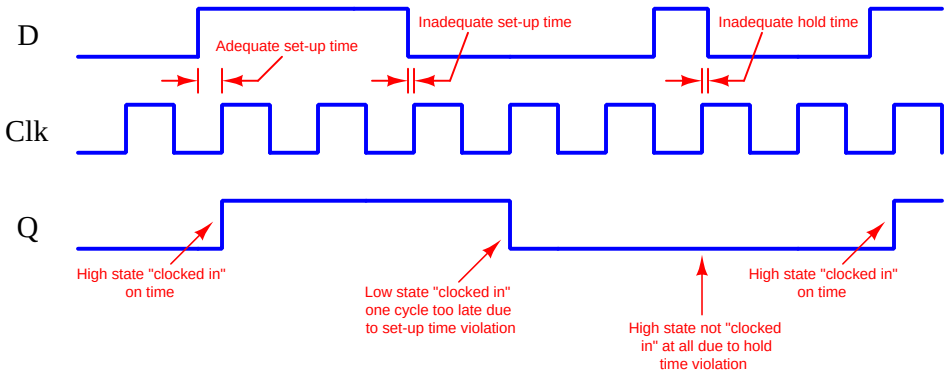
Ideal "pulse" signals have infinitely-steep rise and fall times, level "high" and "low" states well within the specified voltage ranges, and no other imperfections or artifacts. Real pulses always deviate from ideal, often in multiple ways:



Rise and fall times are strongly influenced by parasitic capacitance existing on the signal line with reference to ground, as well as the current-sourcing and current-sinking capability of the logic gate or other device generating the pulse signal. The capacitive "Ohm's Law" formula $I = C\frac{dV}{dt}$ predicts how much current will be necessary to create a linear rate-of-rise or rate-of-fall of voltage for a given capacitance. Note that to achieve infinitely steep rise or fall times an *infinite* amount of current would be necessary to charge/discharge whatever capacitance happens to exist on that signal line!

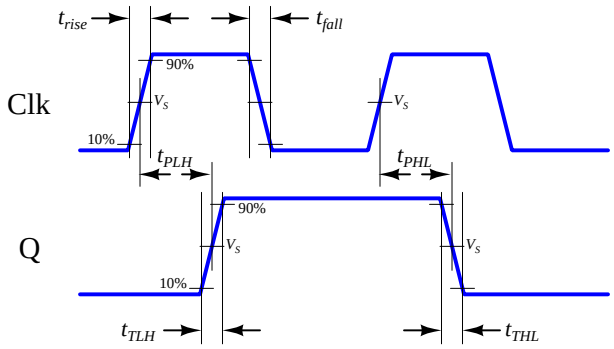
Ringing is caused by *resonance* occurring between parasitic capacitance and parasitic inductance, those two phenomena naturally exchanging energy back and forth with each other to produce the oscillatory waves we call "ringing". Overshoot and undershoot are also the result stored energy within these parasitic L and C circuit properties, and since parasitic capacitance and parasitic inductance can never be fully eliminated from any real circuit it means the effects of over/undershoot and ringing is likewise unavoidable. If you don't see these effects in your pulses, you just aren't viewing them at a fast enough time scale!

Clock-synchronized digital logic circuits such as counters, shift registers, and microprocessors require their input signals to be at stable states immediately before and immediately after the clock pulse arrives. For example, the following timing diagram shows input and output states for a D-type flip-flop circuit (positive-edge triggered), showing the effects of some signal timing violations:



Datasheets for digital circuits often provide timing diagrams showing criteria related to pulse signal timing and logic states. These diagrams don't typically show ideal square-edged pulses, but rather *trapezoidal* pulse profiles intended to exaggerate realistic features such as rise and fall times, propagation delays, and minimum set-up/hold times. Such diagrams usually confuse students who are accustomed to seeing square-edged pulses in their textbook timing diagrams. This technical reference will show some typical timing diagrams and explain what they represent.

For example, consider this timing diagram for a positive-edge-triggered JK flip-flop having both its J and K inputs tied high so as to maintain the circuit in its "toggle" mode. As such we would expect its output (Q) to change state with every rising edge of the clock pulse:



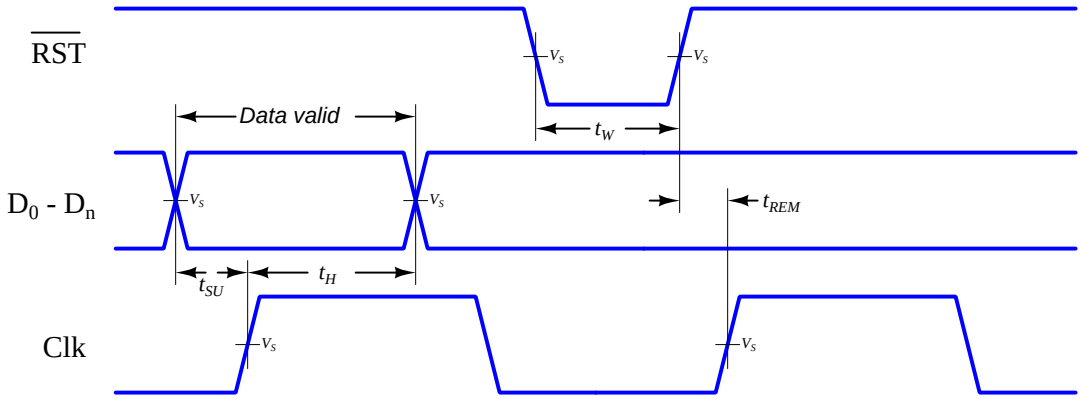
Each of the labels found in this diagram is defined as follows:

- t_{rise} = Rise time of input signal, typically measured from 10% of signal amplitude to 90% of signal amplitude
- t_{fall} = Fall time of input signal, typically measured from 90% of signal amplitude to 10% of signal amplitude
- t_{TLH} = Low-to-High transition time of output signal, typically measured from 10% of signal amplitude to 90% of signal amplitude (the same concept as rise time, but applied to the output signal instead of the input signal)
- t_{THL} = High-to-Low transition time of output signal, typically measured from 90% of signal amplitude to 10% of signal amplitude (the same concept as fall time, but applied to the output signal instead of the input signal)
- t_{PLH} = Propagation delay time of output signal when switching from low to high
- t_{PHL} = Propagation delay time of output signal when switching from high to low
- V_S = Switching threshold voltage, typically defined as 50% of signal amplitude

This timing diagram shows how a digital logic circuit reacts to a single input signal, in this case the clock pulse. Although this example happens to be for a JK flip-flop in toggle mode, the same type of timing diagram with its exaggerated rise/fall times and propagation delays could be applied to any digital logic gate whose output state depended solely on the state of a single input.

For synchronous digital logic circuits where input signals must coordinate with the clock pulse signal in order to be properly accepted by the circuit, we typically find timing diagrams comparing these input states to each other, often without showing the output(s) at all. Instead of showing us how the digital logic circuit will react to an input signal, this sort of timing diagram shows what the digital logic circuit *expects* of its multiple input signals.

The example is shown here for a positive-edge-triggered D register[1] having multiple data lines (D_0 through D_n), one asynchronous[2] reset line ($\overline{RST}$), and one clock input. The arbitrary logic levels of the multiple data lines are shown as a pair of complementary-state pulse waveforms, the only relevant features being the *timing* of the data and not the particular voltage levels of the data signals:



Labels shown in this diagram refer to *minimum* time durations the logic circuit requires for reliable operation:

- t_{SU} = Minimum set-up time before the arrival of the next clock pulse
- t_H = Minimum hold time following the last clock pulse
- t_W = Minimum width (duration) of the asynchronous reset pulse
- t_{REM} = Minimum removal time before the arrival of the next clock pulse

Violations of any of these minimum times may result in unexpected behavior from the logic circuit, and is an all-too-common cause of spurious errors in high-speed digital circuit designs. The assessment of digital pulse signals with regard to reliable circuit operation is generally known as *digital signal integrity*.

[1] In this case, a "D register" is synonymous with multiple D-type flip-flops sharing a common clock input, passing data through from each D input to each corresponding Q output synchronously with each clock pulse.

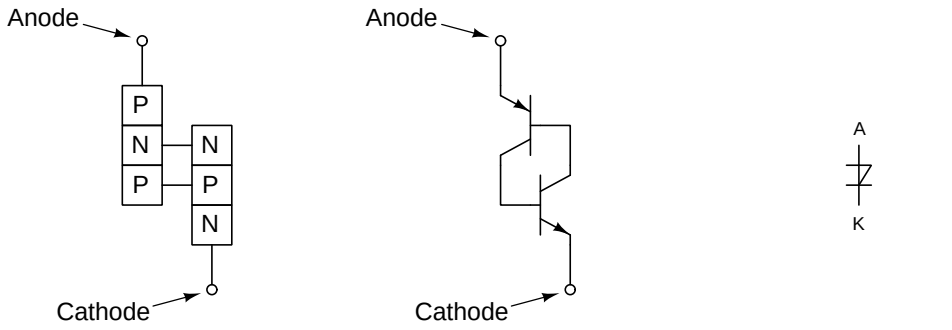
[2] To review, a *synchronous* input depends on a clock pulse while an *asynchronous* input is able to affect the circuit independent of the clock pulse.

5.4 Latch-up in integrated circuits

An often bewildering phenomenon for students of electronics is something called *latch-up*, known to plague certain forms of integrated circuit. "Latch-up" describes a condition where the IC suddenly ceases to function properly, and instead draws an unreasonably high amount of current from the DC power source, sometimes to the point of catastrophic failure. What is more, this sudden and unexpected high-current draw refuses to stop unless and until the DC power source is interrupted.

Close examination of the afflicted IC's internal schematic diagram is unhelpful, as no apparent cause for latch-up will ever be evident upon inspection. This is because latch-up is a phenomenon rooted in *parasitic* transistor structures within the IC itself resulting from the monolithic form of the circuit (i.e. being fabricated on a single wafer of semiconducting material). In fact, if the IC were constructed from a large collection of discrete components on a printed circuit board, latch-up could never occur!

In order to understand what latch-up is and why it occurs, we must first review a class of semiconductor devices called *thyristors*. A "thyristor" by definition is any semiconductor device manifesting *hysteresis*: the tendency to remain on once triggered on and to remain off once turned off. Thyristors include devices such as DIACs, TRIACs, silicon-controlled rectifiers (SCRs), Shockley diodes, and unijunction transistors. Among these various thyristor devices, the most elementary is the Shockley diode, illustrated below:



Crude diagram *Equivalent schematic* *Schematic symbol*

If we examine the equivalent Shockley diode schematic consisting of two interconnected bipolar junction transistors, one NPN and the other PNP, we find that it should not conduct current between the anode and cathode terminals. The only way for one of these transistors to ever turn on is to first initiate base current through it, but the interconnection with the other transistor makes base current for one contingent on collector current of the other, which cannot occur unless and until that other transistor's base is also activated by current. In other words, we have a "Catch-22" scenario where the NPN transistor cannot turn on until the PNP transistor turns on, and vice-versa. Therefore, Shockley diodes – and by extension all thyristors – naturally assume an "off" state and tend to maintain that state even as external voltage is applied across the two end-terminals.

If, however, we imagine *something* turning on one of these transistors, the resulting current would immediately turn on the other transistor, which in turn would hold the first transistor in its "on" state after the initiating event passed. Therefore, Shockley diodes – and by extension all

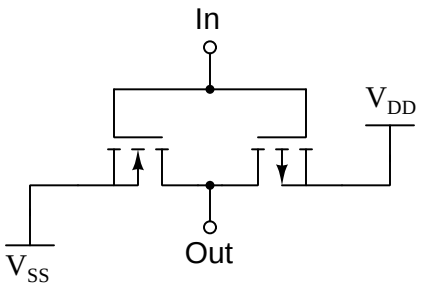
thyristors – tend to remain in their "on" states once triggered. Turning off a thyristor necessitates some external interruption of current. That "something" initially triggering the thyristor could be excessive applied voltage resulting in BJT breakdown, an extremely rapid rate-of-voltage change applied across the thyristor resulting in displacement current "through" one of the transistor's otherwise non-conducting depletion regions, excessive device temperature, a photon of light striking the depletion region and liberating an electron-hole pair, etc. Thyristors designed as controllable devices (e.g. SCRs, TRIACs, unijunction transistors) provide a third terminal to apply a triggering current.

Monolithic integrated circuit construction – where the entire circuit is built on a single wafer of semiconducting material – often results in parasitic PN junctions, and sometimes even parasitic NPN and PNP junction sets. If parasitic NPN and PNP junctions happen to coexist on the silicon die to form the equivalent of a Shockley diode thyristor structure, we will find that circuit susceptible to the unfortunate phenomenon of latch-up. These parasitic thyristors exist within many different types of integrated circuits, from logic gates to amplifiers to comparators to timers such as the popular model 555. They are guaranteed to exist within integrated circuits using complementary pairs of N-channel and P-channel MOSFETs as final "push-pull" output driver stages.

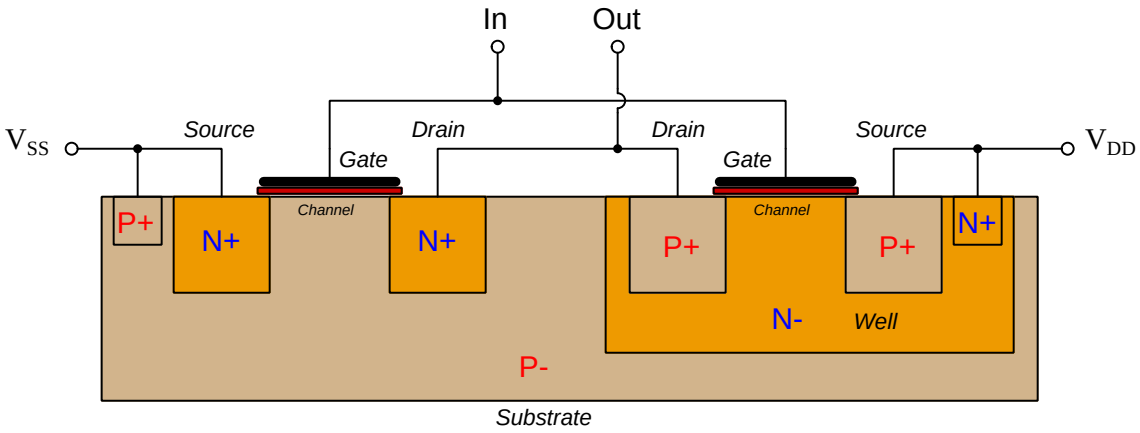
Sometimes the potential for latch-up is communicated in the form of a warning within IC datasheets, but other times not. Inconsistent and incomplete documentation, added to the fact that an examination of an IC's internal schematic does not necessarily reveal the presence of parasitic thyristors, makes latch-up risk a difficult thing to predict.

Let's begin our exploration of these parasitic structures by closely examining one of the most basic building-blocks of CMOS (Complementary Metal-Oxide Semiconductor) circuitry, the so-called *inverter*. A CMOS inverter network is nothing more than a P-channel enhancement-type MOSFET connected to an N-channel enhancement-type MOSFET, as shown below both in schematic form as well as semiconductor wafer cross-section. This complementary MOSFET pair acts as a "push-pull" transistor amplifier, capable of either sourcing current to a load or sinking current from one:

Schematic of an integrated CMOS inverter

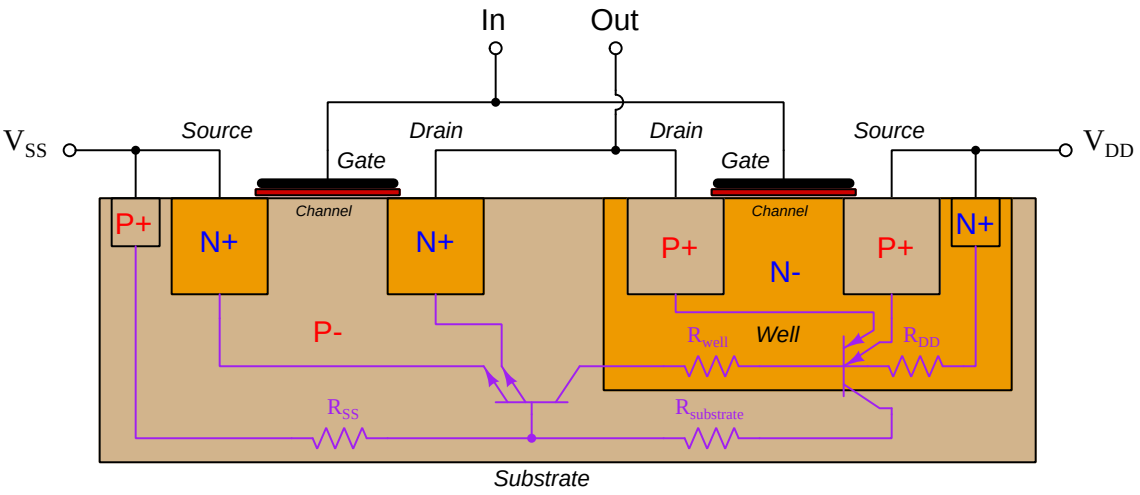


Cross-section of an integrated CMOS inverter

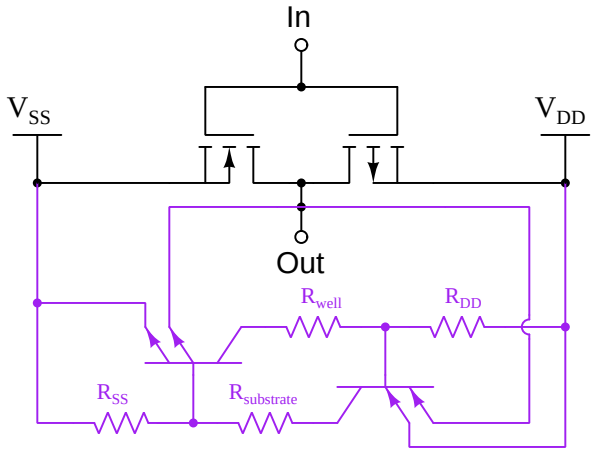


Each of the doped semiconductor layers in the cross-section are noted with their respective dopant concentration levels, a "+" meaning a high concentration of dopant and "−" meaning a low concentration of dopant. Strongly-doped P-type and N-type layers are shown as P+ and N+, respectively. Lightly-doped P-type and N-type layers are shown as P− and N−, respectively. Note how the source and drain regions are all heavily-doped in order to provide a rich reservoir of majority charge carriers to fill in each of the channels when prompted by a sufficient gate-to-substrate electric field.

Now we will include within this cross-sectional illustration some the parasitic transistors[3] and resistances formed by the P and N type doped regions:



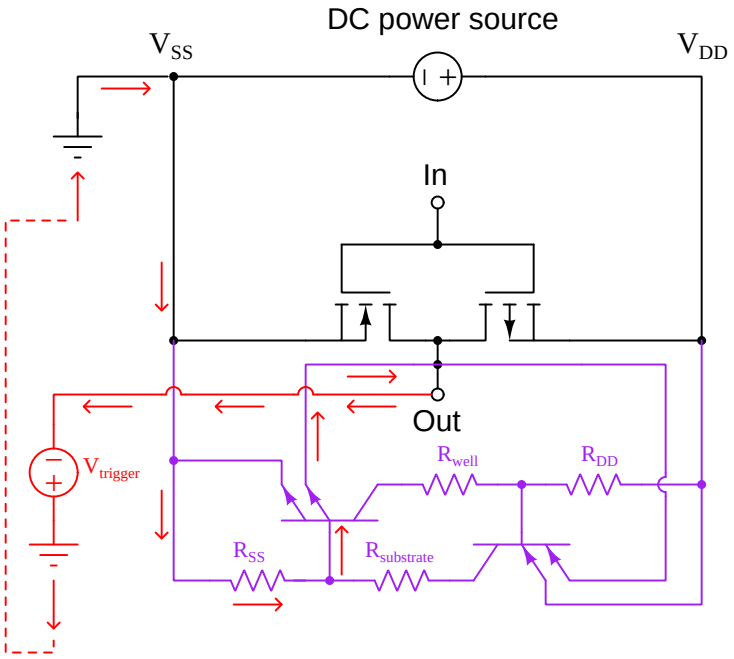
Including these parasitic components into the CMOS inverter schematic diagram:



Now it should be clear that a Shockley diode structure exists within the CMOS inverter's integrated circuit. Furthermore, with each bipolar transistor having two emitters, one each connected to the inverter's output terminal, that output terminal becomes a pathway to externally trigger this thyristor structure and thereby throw the IC into a state of latch-up.

[3] It is helpful to recall that the emitter region of any bipolar junction transistor is more heavily doped than the collector, in order to provide an abundance of charge carriers to inject into the base region as minority carriers. This is why the emitters of the PNP are P+ doped and the collector P− doped, and why the emitters of the NPN are N+ doped and the collector N− doped. Multiple emitters in a BJT simply mean more than one location from which charge carriers may be injected into the base and thereby render the transistor conductive between emitter and collector.

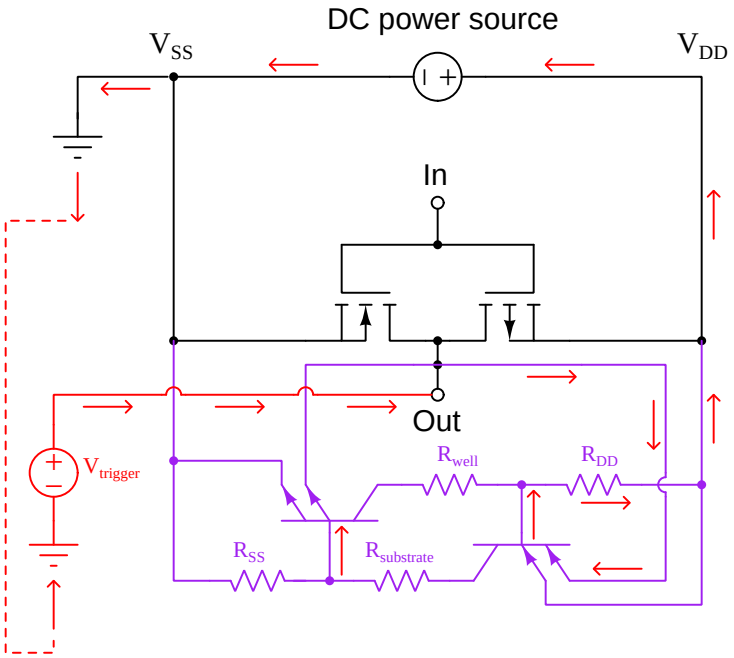
Triggering a latch-up condition is as simple as something external to the inverter IC driving its output terminal below V_{SS} potential or above V_{DD} potential. In the next example we will see a below-V_{SS} potential initiating latch-up:



In this scenario, external source $V_{trigger}$ forced the inverter's output terminal to a potential lower than ground (V_{SS}), and this in turn prompts a current through one of the emitters of the parasitic NPN transistor, injecting electrons into its P-type base region and turning it on. Once the NPN transistor turns on, its collector current permits base current through the PNP transistor, turning it on as well, and thus the two parasitic transistors latch each other into conductive states.

Once latched on, the parasitic thyristor creates a V_{DD}-to-V_{SS} short-circuit. Excessive current now passes through the DC power source and through the parasitic resistances of R_{well} and $R_{substrate}$, possibly damaging the entire integrated circuit by excessive heating. Only an interruption of the DC power source will halt the latch-up condition.

Next, we will see an above-V_{DD} potential initiating latch-up:



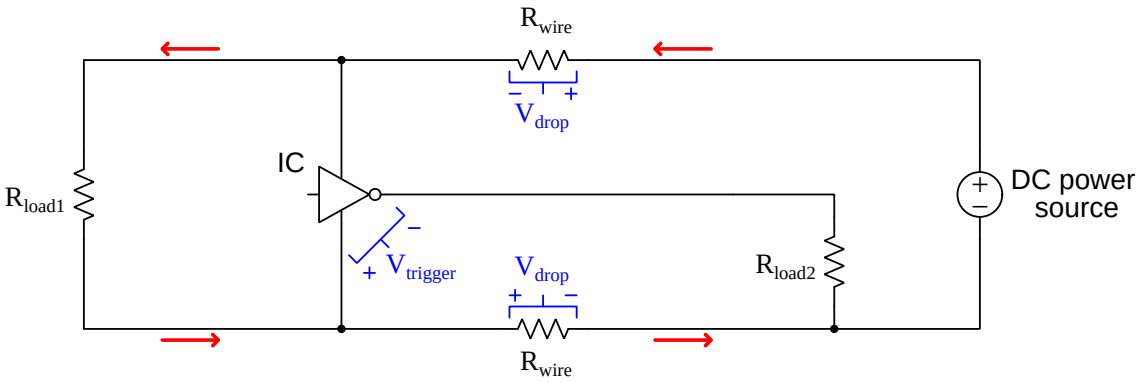
In this scenario, external source $V_{trigger}$ forced the inverter's output terminal to a potential higher than that of the DC source voltage (V_{DD}), and this in turn prompts a current through one of the emitters of the parasitic PNP transistor, injecting holes into its N-type base region and turning it on. Once the PNP transistor turns on, its collector current permits base current through the NPN transistor, turning it on as well, and thus the thyristor formed by the two parasitic transistors becomes latched in its "on" state.

Again, once the parasitic thyristor is latched in its "on" state it will create a V_{DD}-to-V_{SS} short-circuit. Excessive current will now pass through the DC power source and through the parasitic resistances of R_{well} and $R_{substrate}$, possibly damaging the entire integrated circuit by excessive heating. Only an interruption of the DC power source will abort the latch-up condition.

Clearly, one of the causes of latch-up is some external electrical source attempting to force the output terminal either above V_{DD} potential or below V_{SS} potential, and as such we should strive to avoid such possibilities in all our circuit designs.

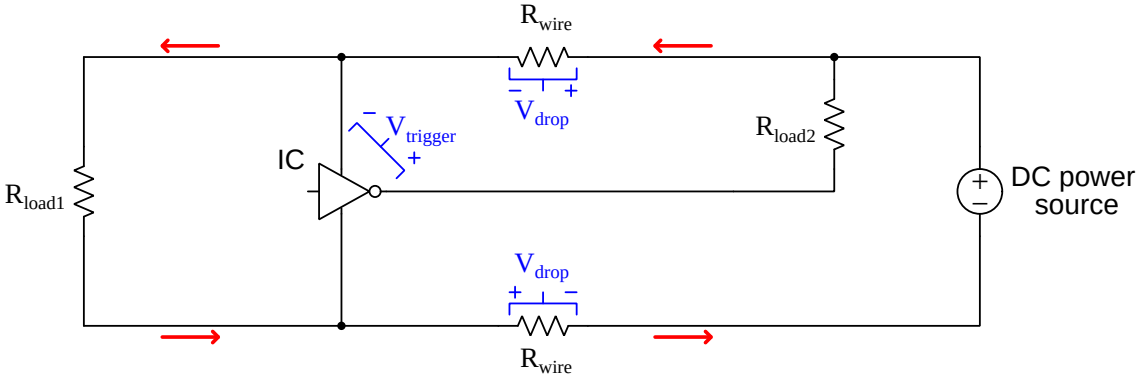
Another potential cause of latch-up is if the load driven by an IC output is too inductive. Inductance stores energy in magnetic fields, and the amount of energy is a function of the current through the inductance. If an inductive load is energized by the output of an IC and then the IC switches state to de-energize that load, the load inductance will become a temporary current source and try to force current through the IC's output, often generating surprisingly high voltages in the process. Such voltage transients may easily trigger latch-up in the IC.

Another possible cause of latch-up is power supply rail instability. That is, if something is causing an IC's positive and/or negative rail potentials to vary, and that same IC connects to some other device with higher positive or lower negative power supply rails, this may constitute a triggering mechanism for latch-up. Consider the following example:



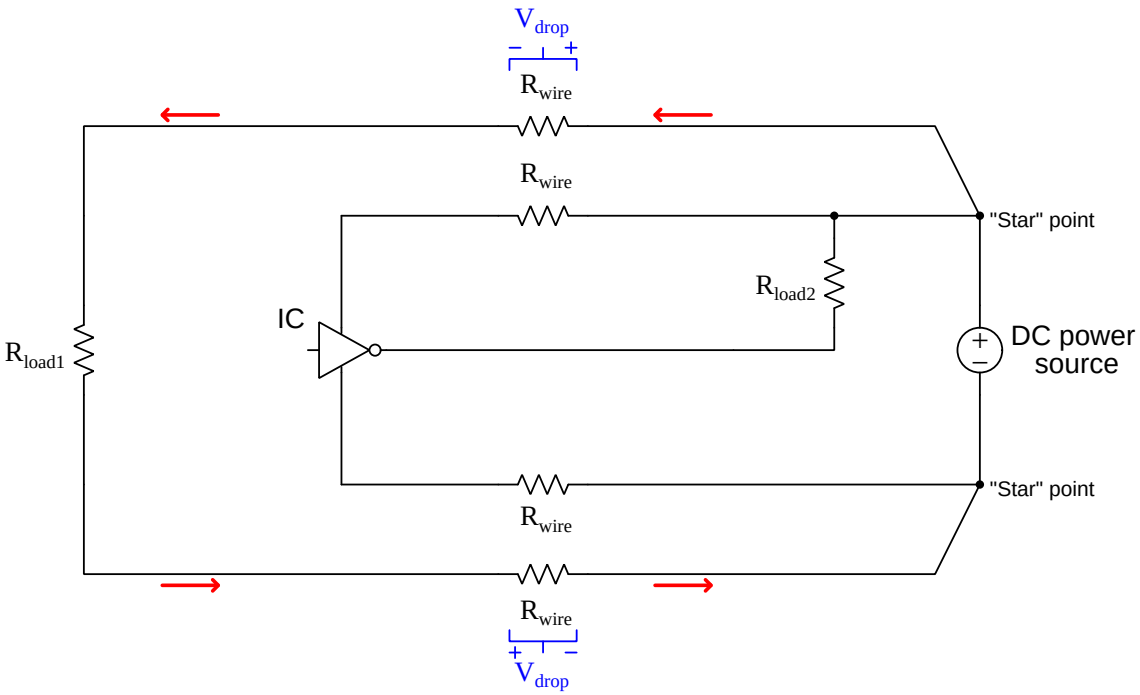
Here we see some voltage dropped along the conductors carrying current to a heavy load (R_{load1}), the effect of these voltage drops being to suppress the potential of the IC's positive rail terminal slightly below that of the power source's positive terminal, and to elevate the potential of IC's negative rail terminal slightly above that of the power source's negative terminal. In this case, with the IC's output connected to a second load (R_{load2}), we see that the voltage drop along the negative power conductor has the effect of making the IC's output terminal fall to a potential slightly less than its negative power terminal. As we have seen previously, this may be enough to trigger latch-up in that IC.

If R_{load2} happened to be connected to the positive conductor rather than the negative, we would similarly risk latch-up due to the load imposing a potential on the IC's output greater than that of its positive power terminal:



In either scenario the wiring voltage drop caused by the high current demand of R_{load1} risks latch-up in the IC by skewing its power-supply terminal potentials compared to the load it connects to. A

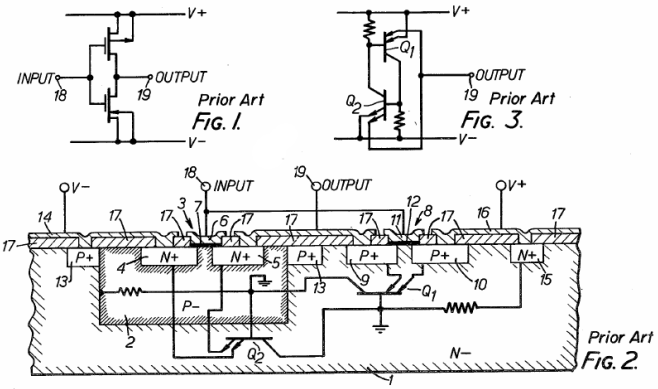
strategy for avoiding this problem is to provide high-current loads with their own dedicated power conductors, and not connect them to the same power conductors that sensitive ICs use for power delivery. This strategy is often referred to as *star-point grounding* or star-point power distribution:

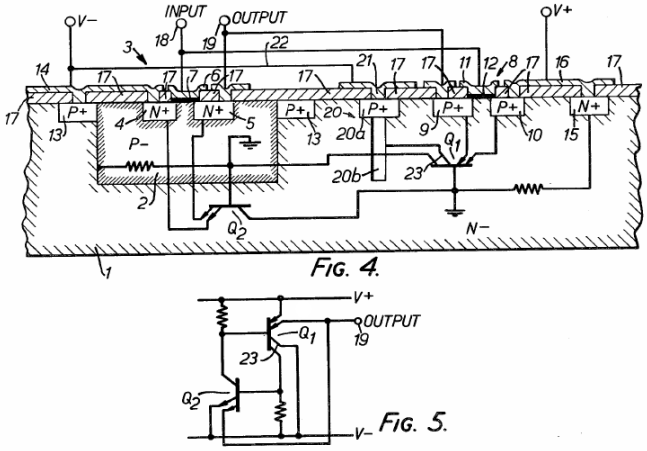


Star-point power connections do not eliminate voltage drop due to high load current, but they do isolate those voltage drops from affecting *other* loads such as ICs sensitive to latch-up.

Given the fact that Complementary MOSFET transistor pairs are so prevalent in modern integrated circuits (e.g. *all* CMOS digital logic) we must find mitigations for latch-up or else risk self-destruction of a great many ICs any time an output pin is subjected to even the briefest transient voltage outside of the DC power supply's rail potentials.

One such mitigation was invented by Alastair Stevenson of the Hughes Aircraft Company in 1978, consisting of an extra layer of implanted material in the semiconductor structure serving as a new collector of charge carriers. His patent ("CMOS Monolithic Integrated Circuit", US patent 4,173,767 filed 5 May 1978 and granted 6 November 1979) contrasts the "prior art" of CMOS inverter construction[4] (figures 1 through 3) against his innovation shown in figures 4 and 5:





In Stevenson's design we see the addition of a heavily-doped P+ region (20a in figure 4) near the surface of the semiconductor die connecting with a deeper region of lightly-doped P− material (20b in figure 4) inset into the N− substrate. This P− region serves to collect holes injected as minority

[4]Stevenson's prior art example happens to use an N-type substrate rather than a P-type substrate as I showed earlier, thus necessitating a P-type well surrounding the N-channel MOSFET rather than the N-type well surrounding the P-channel MOSFET in my illustrations.

carriers into the N− substrate that might otherwise travel to the P− well region of the N-channel MOSFET, essentially adding a second collector to the PNP parasitic transistor. The purpose for the P+ region interfacing the metal surface conductor to the deeper P− inset is to avoid "rectifying" contact between metal and semiconductor surfaces, the heavy doping of that P+ region (20a) serving instead to form an "ohmic" contact with the metal. This additional collector region serves to divert current away from the parasitic NPN transistor that would otherwise turn it on and ensure latch-up once the PNP transistor was triggered by an offending potential at the output terminal.

Stevenson describes how such current-collecting channels inserted into the substrate need only be placed between complementary MOSFETs at device input or output terminals, not between every pair of complementary MOSFETs, due to the fact that only those connecting to device pins are susceptible to the triggering that cause latch-up:

> It will be appreciated that the circuit arrangement shown in FIG. 4 represents only one small portion of the integrated circuit which will in fact include many more complementary MOS transistors. However, channels such as 20 need not necessarily be included between all adjacent complementary transistors in the circuit, and preferably channels such as 20 are only included between complementary transistors that constitute input or output stages of the circuit since it is only these transistors which will be subject to spurious voltage fluctuations which could trigger a bipolar conductive latch up in the circuit.

A final note on the topic of latch-up is that this term is unfortunately mis-used to describe a phenomenon plaguing certain comparators and operational amplifiers. In this other meaning of the term "latch-up", the comparator or opamp output terminal assumes an incorrect potential because one or more of the input terminal potentials has approached or exceeded a power supply rail potential. This kind of "latch-up" is sometimes referred to as *pseudo latch-up* because unlike the thyristor-based latch-up we've been exploring here, pseudo latch-up immediately ceases as soon as the offending input potential returns to normal. In the "real" form of latch-up, only interruption of DC power will cause the latching action to cease.

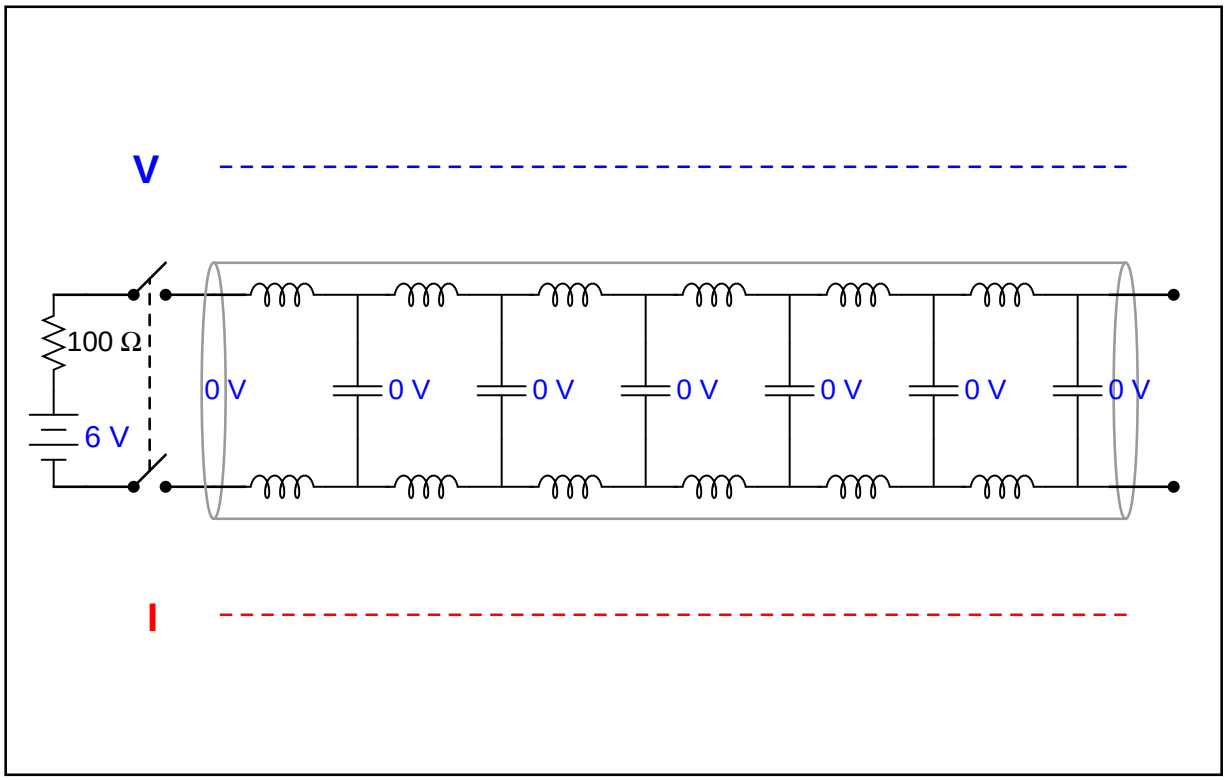
Chapter 6

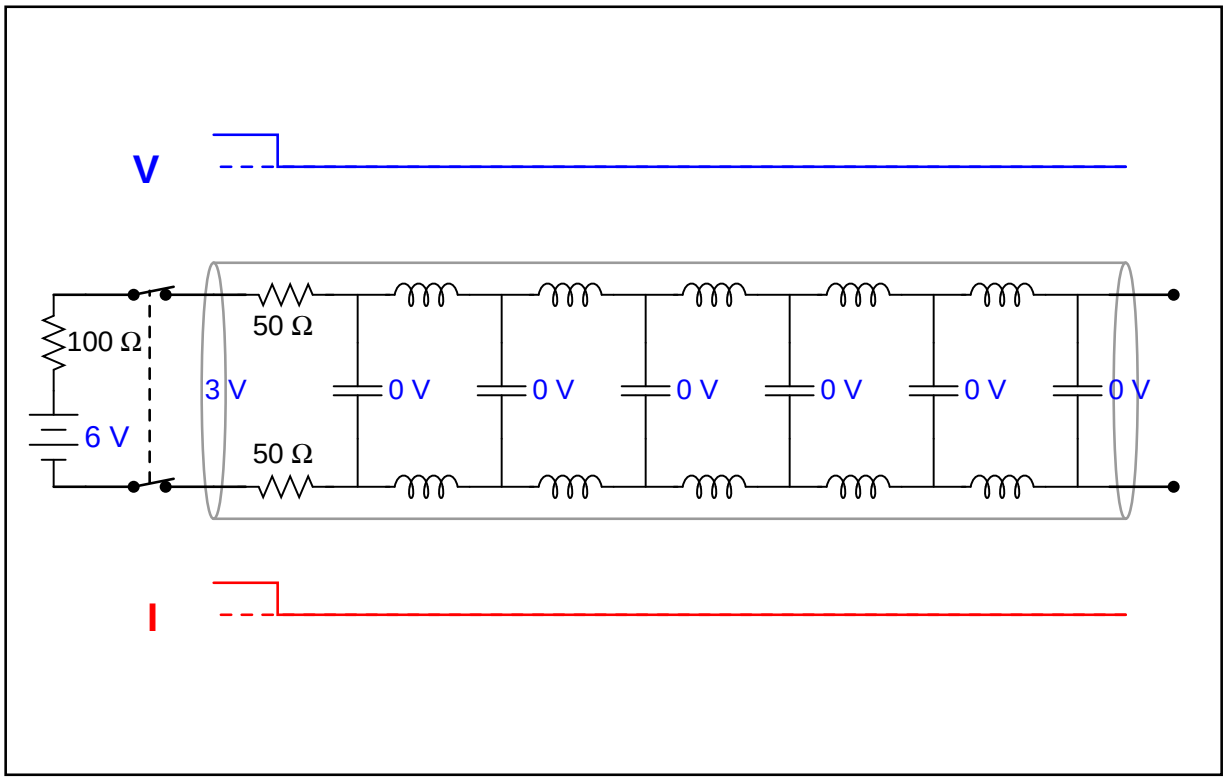
Animations

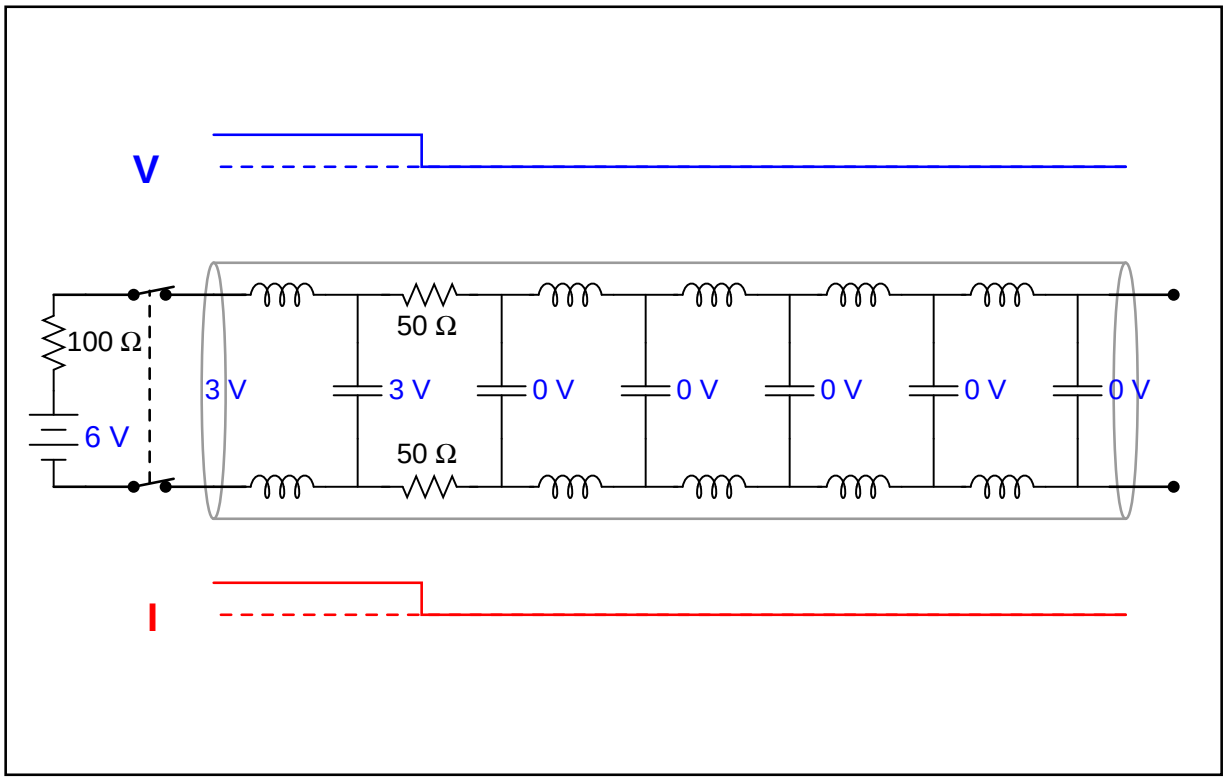
Some concepts are much easier to grasp when seen in *action*. A simple yet effective form of animation suitable to an electronic document such as this is a "flip-book" animation where a set of pages in the document show successive frames of a simple animation. Such "flip-book" animations are designed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

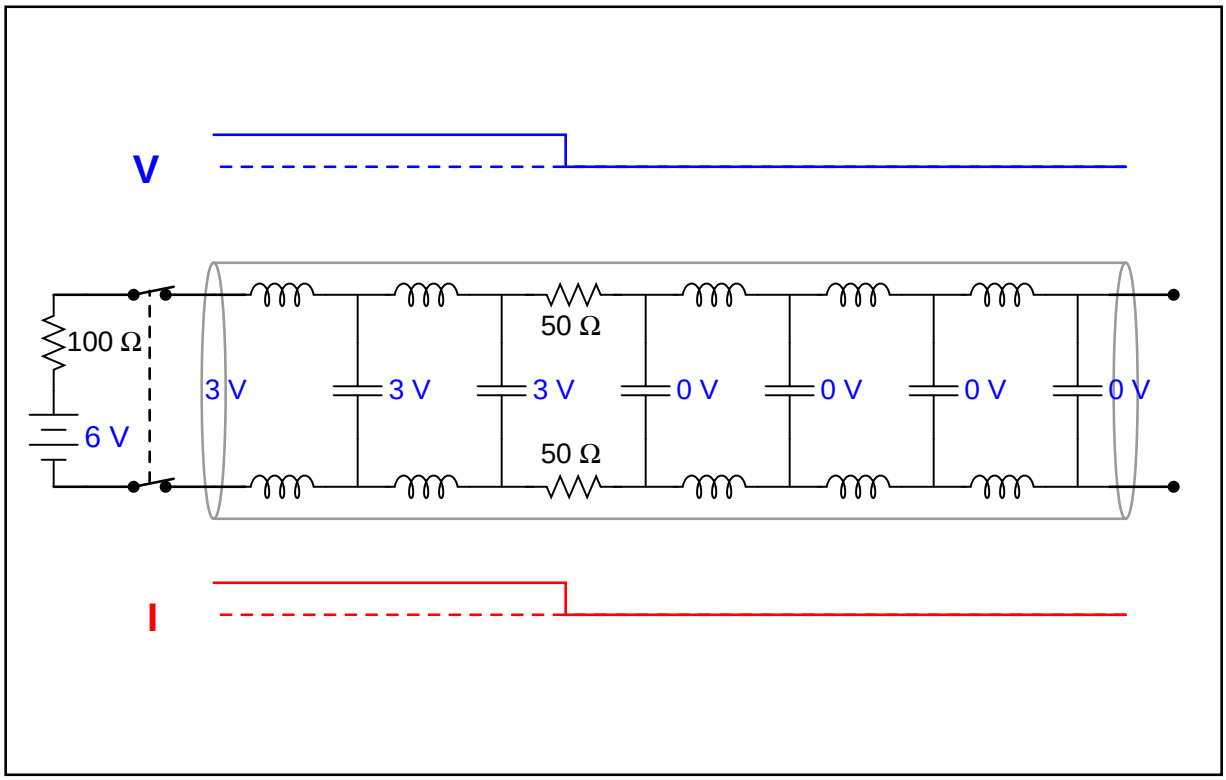
6.1 Animation of an open transmission line

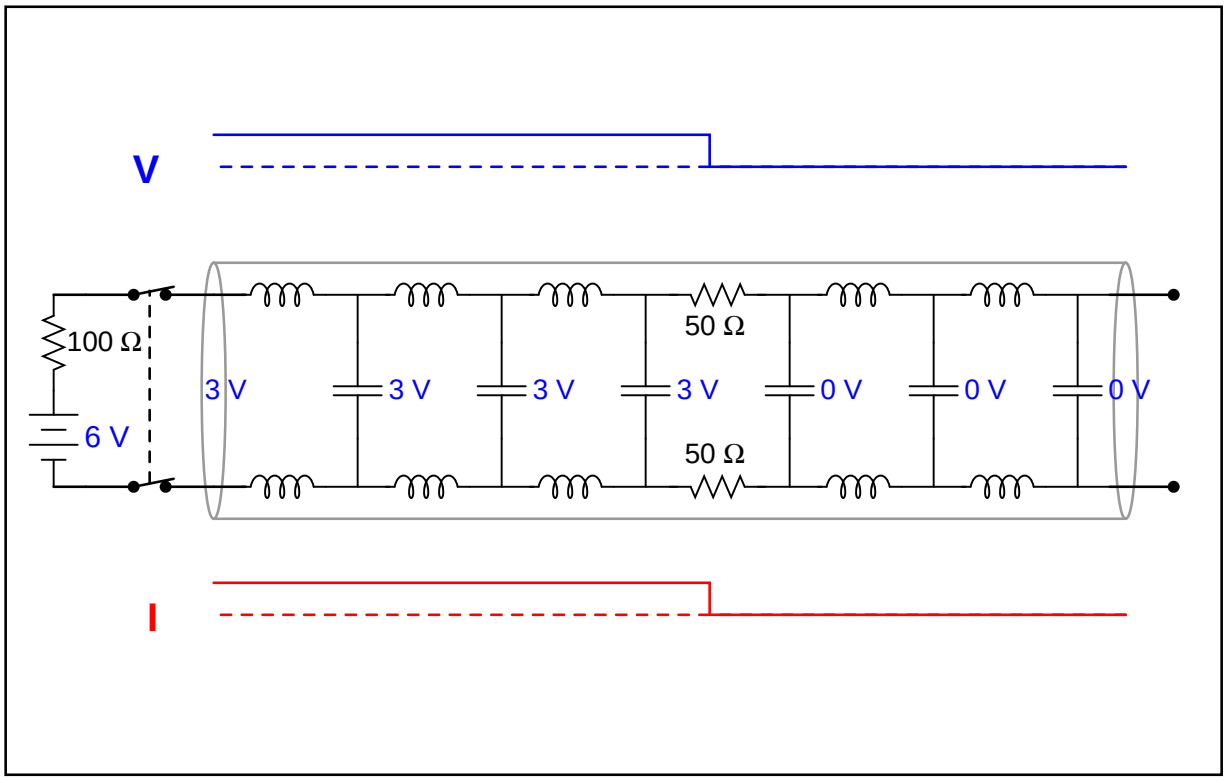
The following animation shows the progression of voltage and current down the length of a 100 Ohm transmission line while energized by a 6 Volt DC source, that line being terminated by an open-circuit.

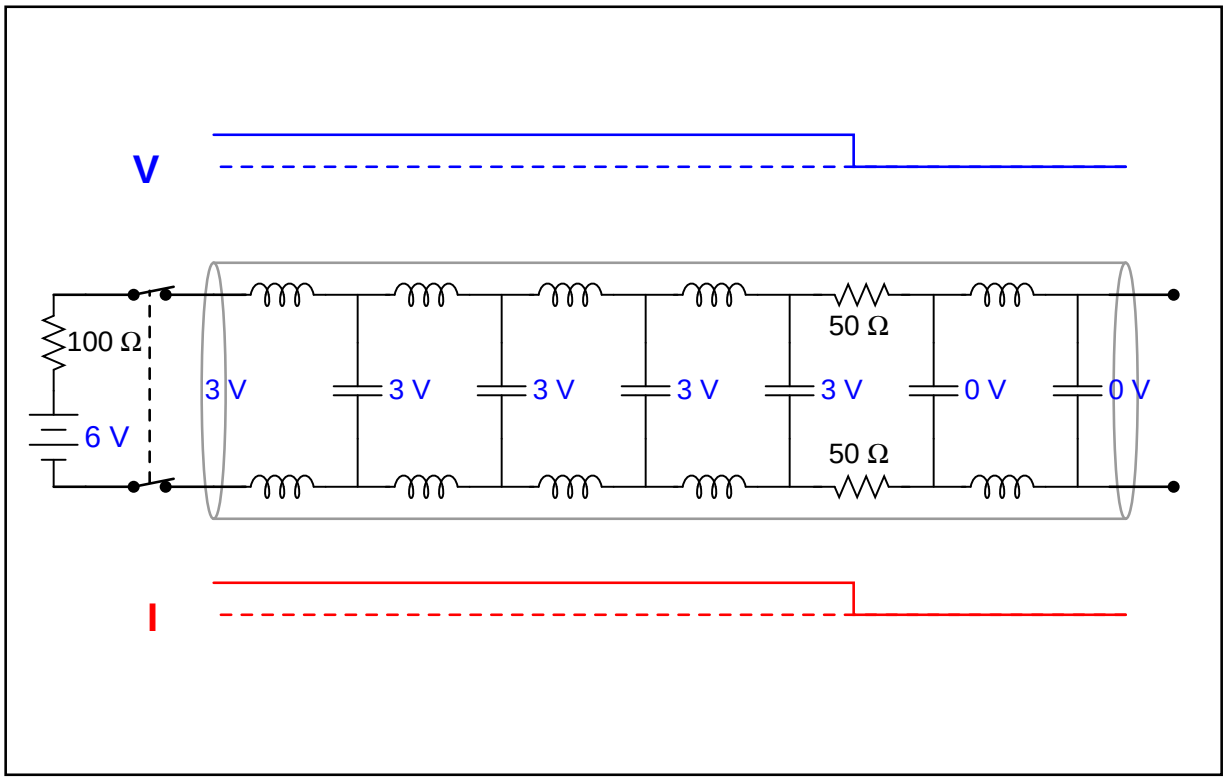

V
100 Ω
6 V
0 V
0 V
0 V
0 V
0 V
0 V
0 V
I

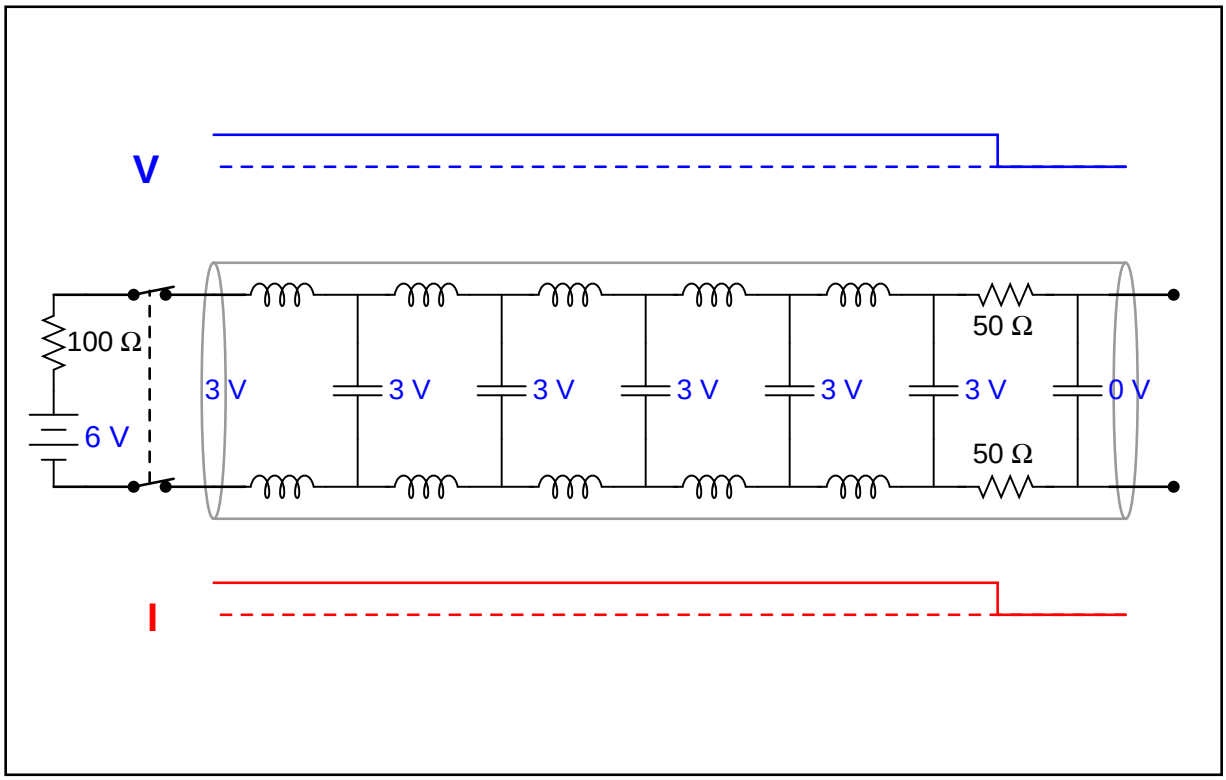

V
100 Ω
6 V
50 Ω
3 V
0 V
0 V
0 V
0 V
0 V
0 V
50 Ω
I

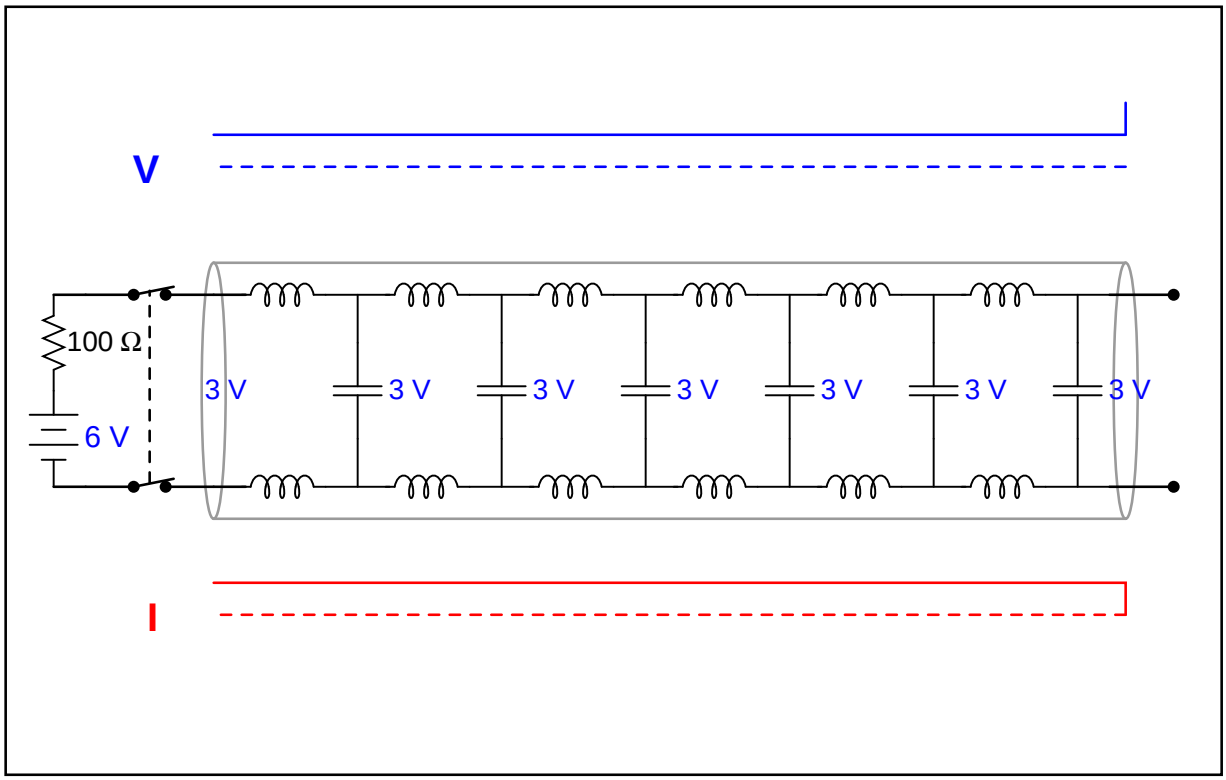

V
100 Ω
6 V
50 Ω
3 V
3 V
0 V
0 V
0 V
0 V
0 V
50 Ω
I

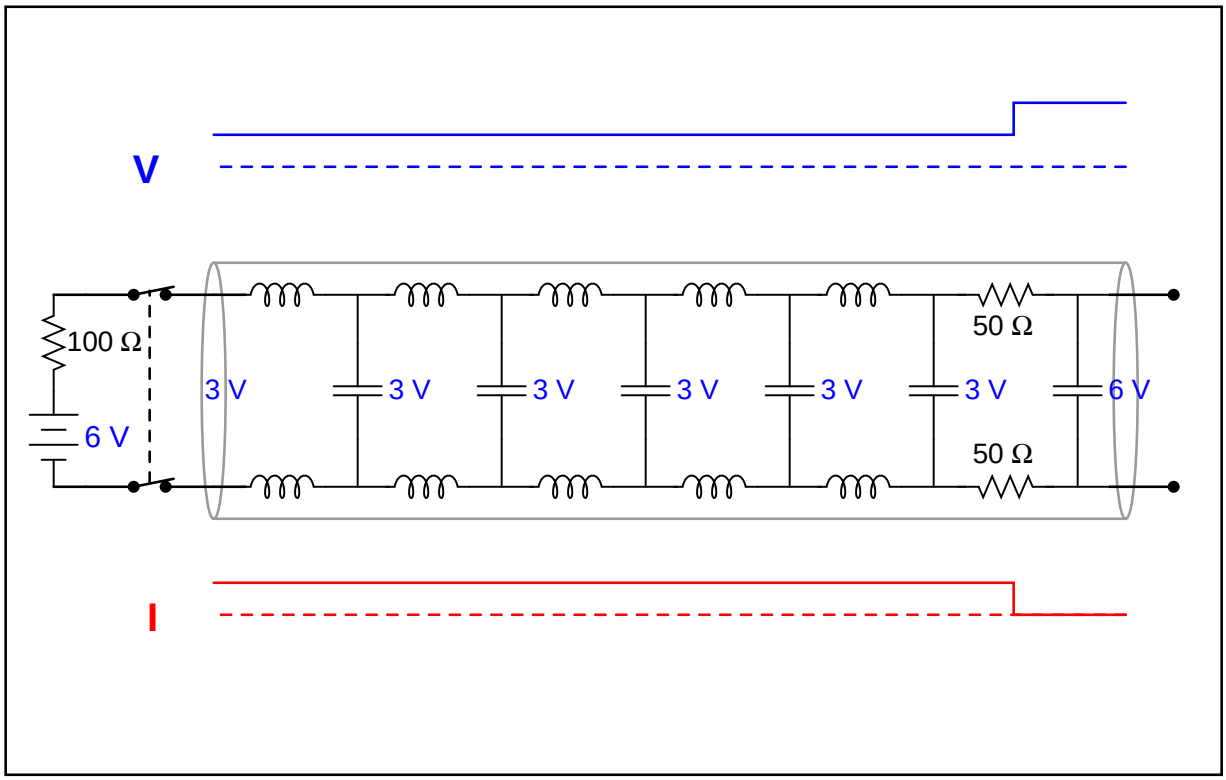

V
100 Ω
6 V
3 V
3 V
3 V
50 Ω
50 Ω
0 V
0 V
0 V
0 V
I

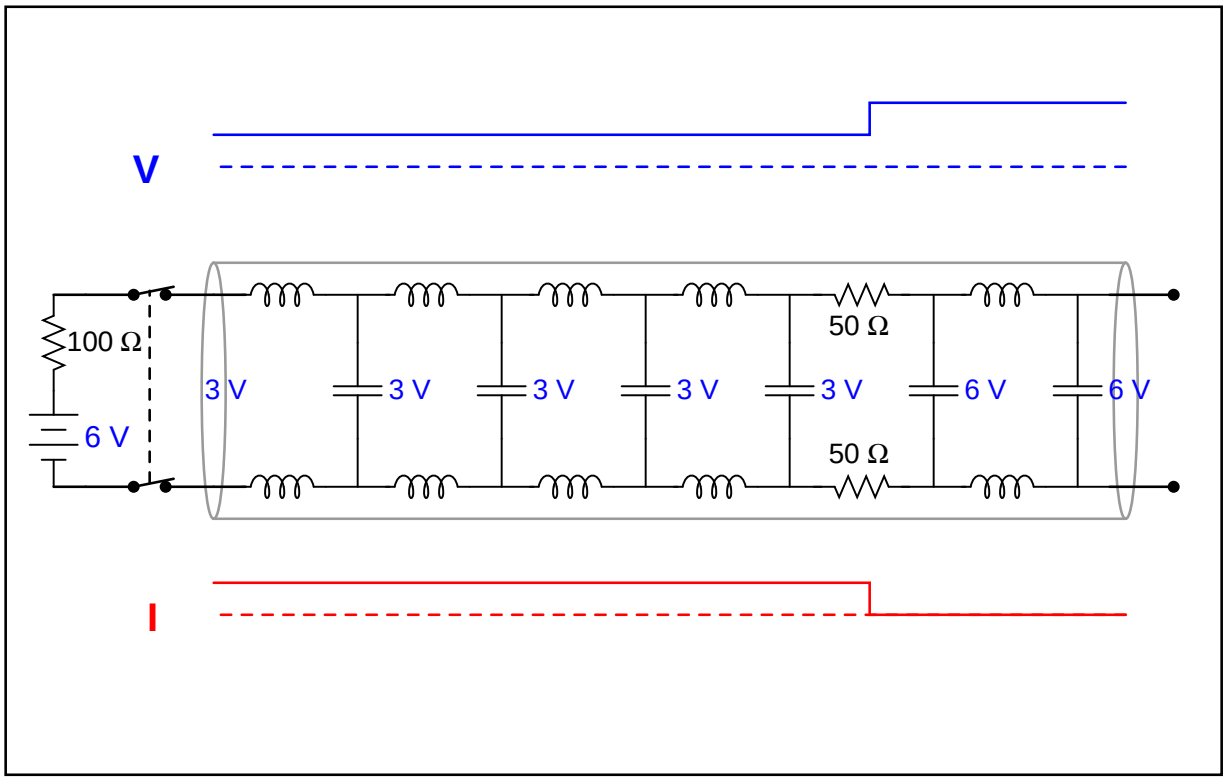

V
100 Ω
6 V
3 V
3 V
3 V
3 V
50 Ω
50 Ω
0 V
0 V
0 V
I

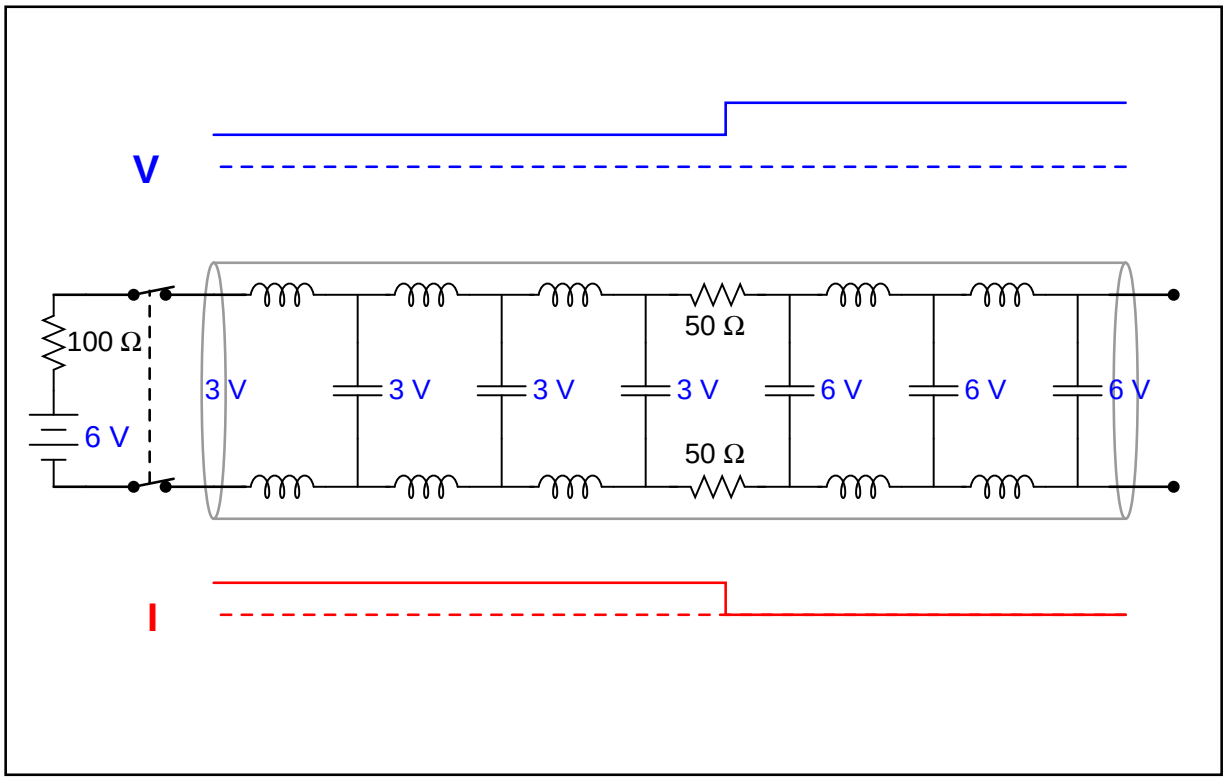

V
100 Ω
6 V
3 V
3 V
3 V
3 V
3 V
50 Ω
0 V
0 V
50 Ω
I

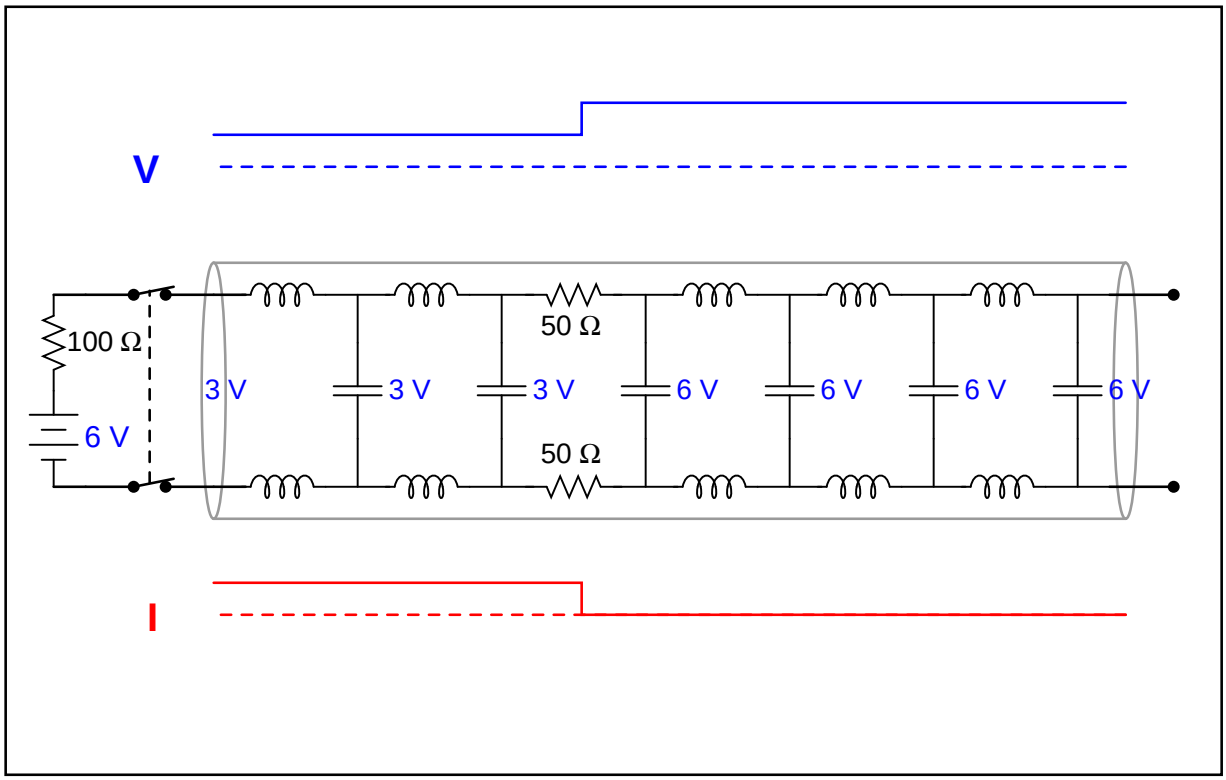

V
100 Ω
6 V
3 V
3 V
3 V
3 V
3 V
3 V
0 V
50 Ω
50 Ω
I

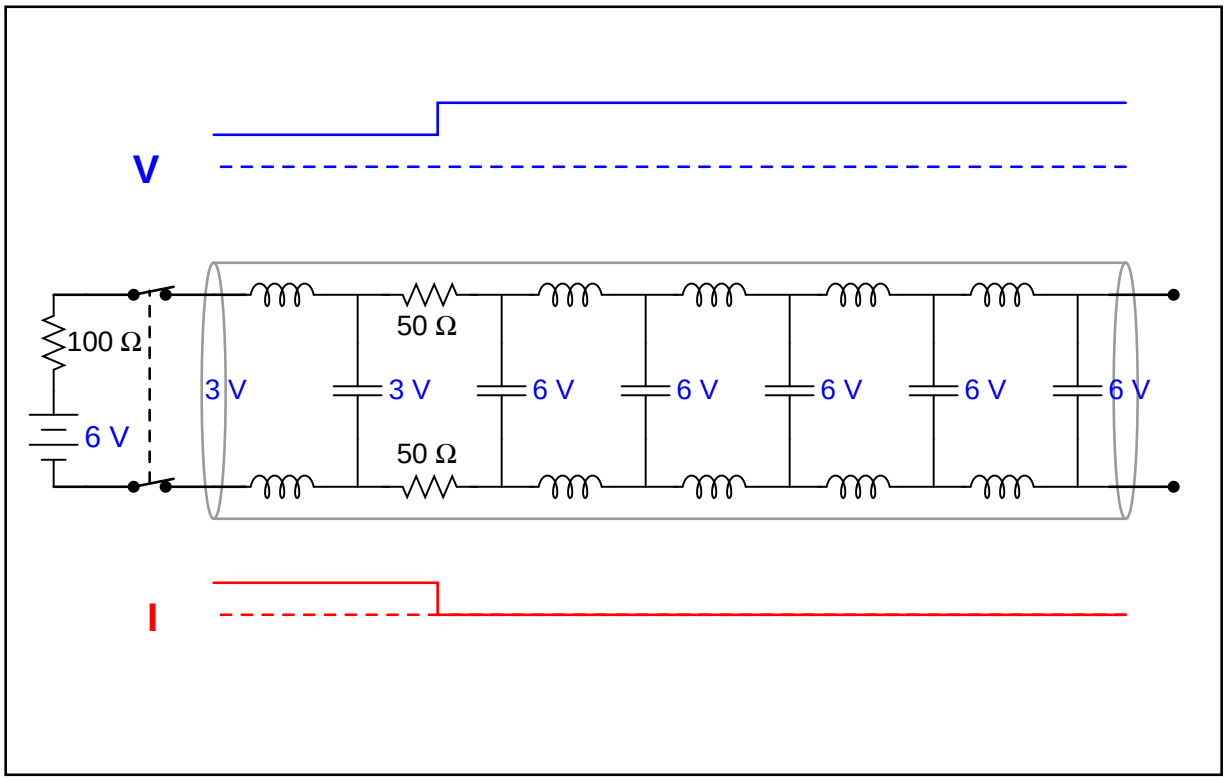

V
100 Ω
6 V
3 V
3 V
3 V
3 V
3 V
3 V
3 V
I

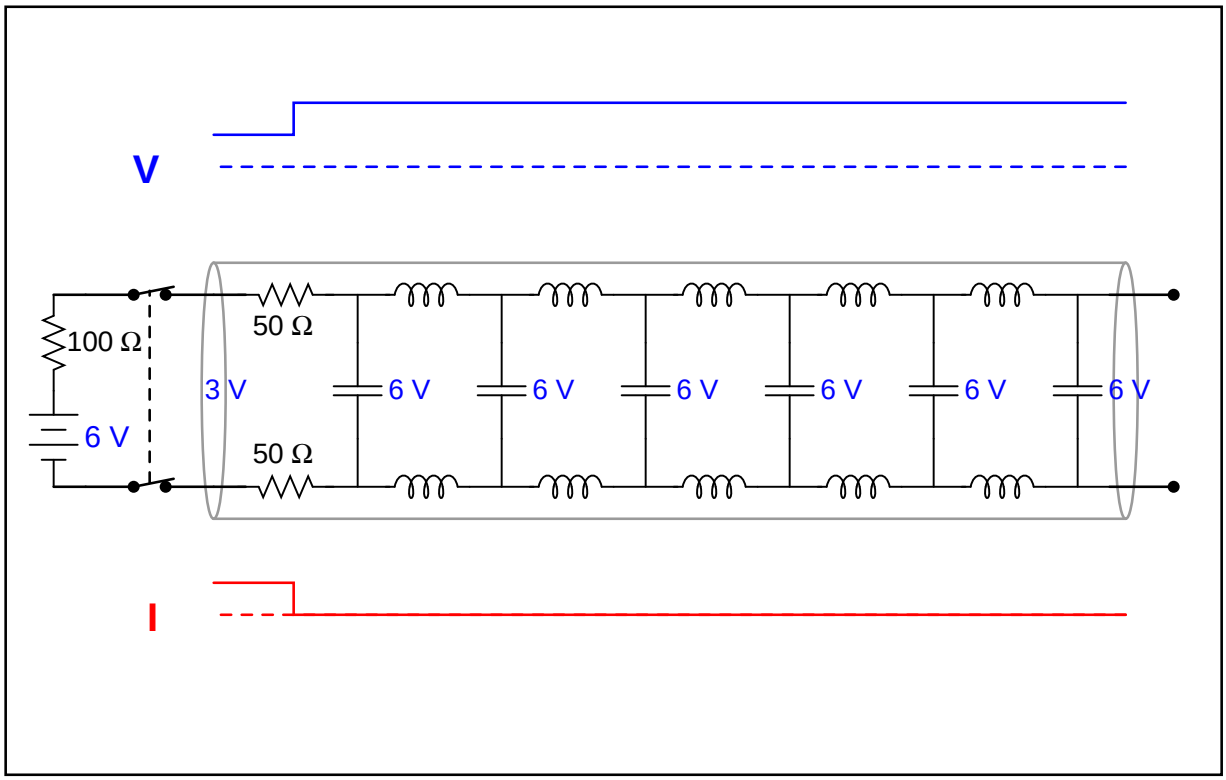

V
100 Ω
6 V
3 V
3 V
3 V
3 V
3 V
3 V
6 V
50 Ω
50 Ω
I

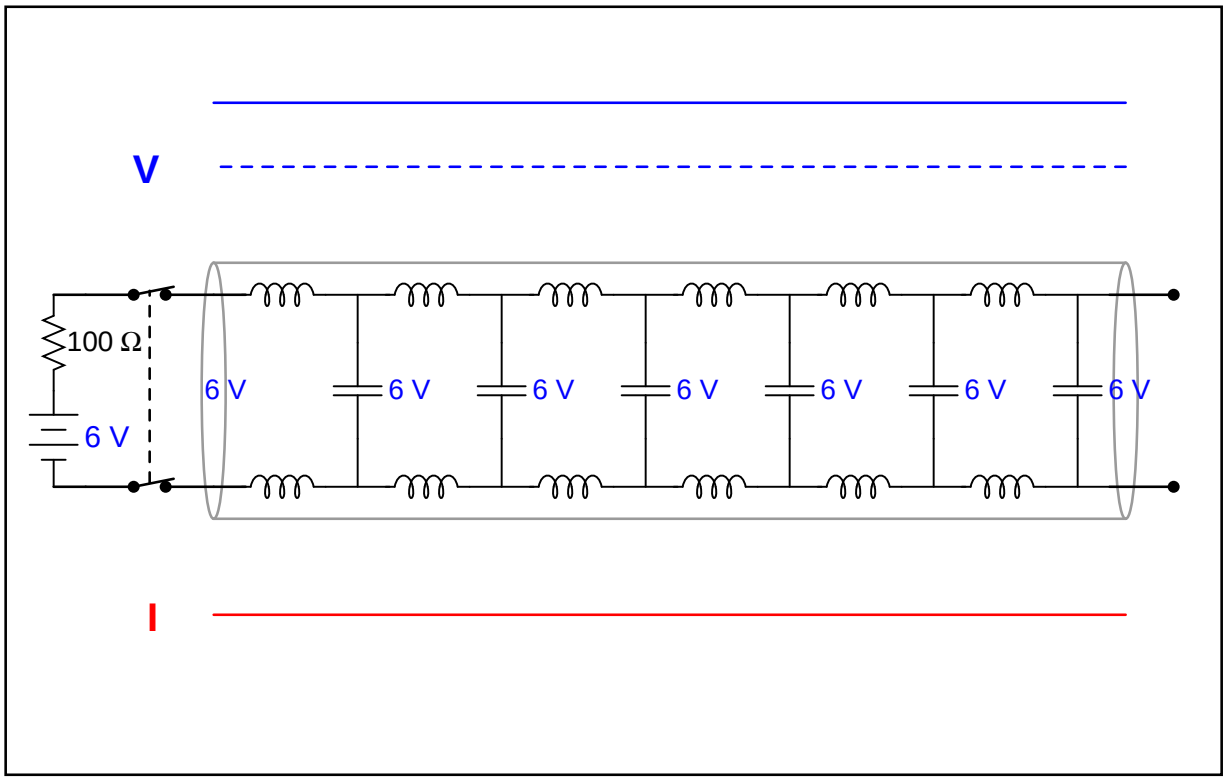

V
100 Ω
6 V
3 V
3 V
3 V
3 V
3 V
50 Ω
6 V
6 V
50 Ω
I


V
100 Ω
6 V
3 V
3 V
3 V
3 V
50 Ω
50 Ω
6 V
6 V
6 V
I


V
100 Ω
6 V
3 V
3 V
3 V
50 Ω
6 V
6 V
6 V
6 V
50 Ω
I


V
100 Ω
6 V
50 Ω
3 V
3 V
6 V
6 V
6 V
6 V
6 V
50 Ω
I


V
100 Ω
6 V
50 Ω
3 V
6 V
6 V
6 V
6 V
6 V
6 V
50 Ω
I


V
100 Ω
6 V
6 V
6 V
6 V
6 V
6 V
6 V
6 V
I

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:

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

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

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

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

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

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

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

Logic gate

Logic state

TTL

CMOS logic

Frequency

Electric field

Magnetic field

Capacitance

Inductance

Parasitic effect

Transmission line

Electromagnetic induction

Faraday's Law of electromagnetic induction

Mutual induction

Time domain

Frequency domain

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

Fourier series

Fundamental frequency

Harmonic frequency

Linearity

Bandwidth

Resonance

Hysteresis

Propagation delay

Clocking

Set-up time

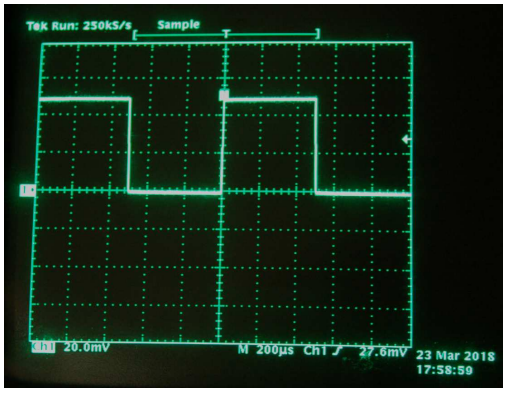
Hold time

Electromagnetic wave

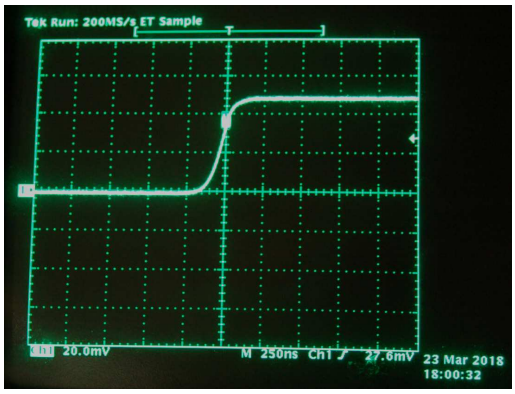
Differential voltage signal

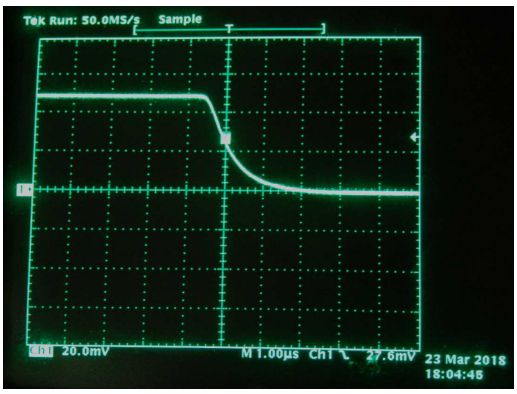
7.1.3 Measuring rise and fall times

The following photograph shows a digital oscilloscope's display of a 1 kHz square wave:



Next are two displays of the exact same square wave signal, but with the oscilloscope configured differently:





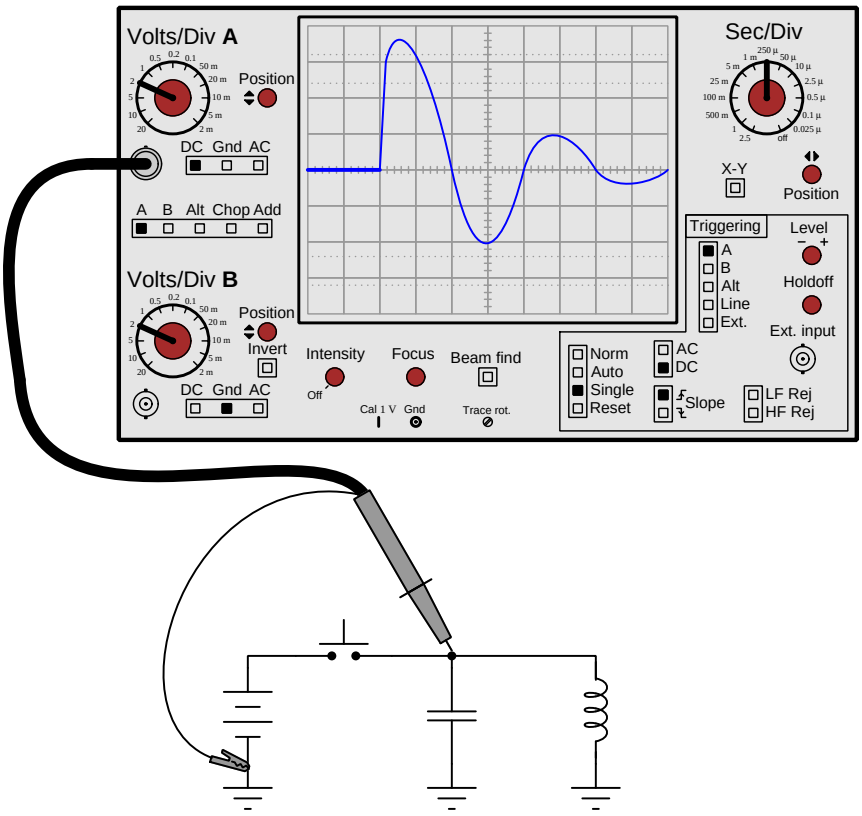
Identify why this same signal looks quite different in the lower two photographs compared to the upper photograph, and why the upper image may be misleading.

Challenges

- Identify the peak amplitude of this square wave voltage signal, assuming the use of a 10:1 oscilloscope probe.
- Which is faster in this waveform, the rising edge or the falling edge?

7.1.4 Transient oscillation

If an oscilloscope is set up for "single-sweep" triggering and connected to a DC-excited resonant circuit such as the one shown in the following schematic, the resulting oscillation will last just a short time (after momentarily pressing and releasing the pushbutton switch):



Explain why the oscillations die out, rather than go on forever.

Challenges

- Why does a pendulum, following an initial push, eventually stop swinging?
- Could this circuit be modified so as to permit continual oscillations instead of damped oscillations?

7.1.5 The Soul of a New Machine

Tracy Kidder's 1981 Pulitzer Prize-winning book *The Soul of a New Machine* chronicles the effort by electrical engineers working at the Data General Corporation to design and build a 32-bit minicomputer, in a time before personal computers became popular. An obsolete term now, *minicomputer* refers to a computer intended for small business usage, more powerful than a personal computer of that era but far less expensive than a "mainframe" of the type used by universities and large corporations. Minicomputers did not use microprocessors, but used medium-scale integrated circuits (ICs) soldered into large printed circuit boards to form basic processing modules.

In chapter 10 of Kidder's book, entitled "The Case of the Missing NAND Gate", he writes about a particularly challenging period of the new computer's development where engineers labored for days to identify and remedy some design flaws in the system. One of the remedies involved adding a new NAND gate to one of the circuit designs, which explains the chapter's title.

The final problem (or "bug" as it was called) described in this chapter resided in a circuit board called the Instruction Processor, or IP. Two engineers, Jim Veres and Ken Holberger, are trying to figure out why one of the prototype computers (named Gollum) occasionally fails a diagnostic test program even after the inclusion of the new NAND gate, their "debugging" efforts described in the following paragraph:

> Veres takes out the NAND gate. He runs the failing program. The machine still commits the new failure. So it's not the NAND gate that did it. Greatly relieved, smiling now, Holberger points out that they have the IP board out on the "extender." The board is hooked up to Gollum but it's sitting in a frame of its own, outside the main frame of the machine. This is standard debugging practice, but boards aren't made with extenders in mind, and in some cases a perfectly good board won't work while out on the extender. With nimble fingers, they put the IP back in its proper place among the other boards, and the failure no longer occurs.

Here is a photograph of a "bus extender board" for a simple computer (not the Data General "Gollum" computer!). As you can see, the printed circuit board's wiring consists of nothing more than straight copper traces from plug (left) to socket (right):

Although Kidder never explains why placing the IP board on the extender caused the fault, we may hypothesize based on our knowledge of digital signal integrity. Identify some plausible reasons

why a circuit board such as the IP in this prototype minicomputer might fail to properly function when connected to the rest of the computer via an "extender" which usually consists of another printed circuit board with long parallel traces acting as a sort of "extension cable" for all the power and digital signal conductors.

Challenges

- Modern computers utilize far denser integrated circuits than were available at the time Kidder wrote his book. Do you think the denser packaging of circuitry helps or hurts with regard to digital signal integrity?
- Identify where on the extender card (shown in the photograph) the designer of this card added copper tracing to act as electrostatic shielding between the signal traces.

7.1.6 Two crosstalk scenarios

Two graphical images from US Patent 7,562,323, "System, Method and Computer Program Product for Handling Small Aggressors in Signal Integrity Analysis", by Xiaoliang Bai and Igor Keller show examples of crosstalk between logic gates:

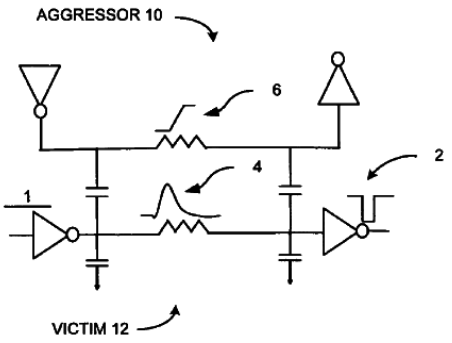


Fig. 1A

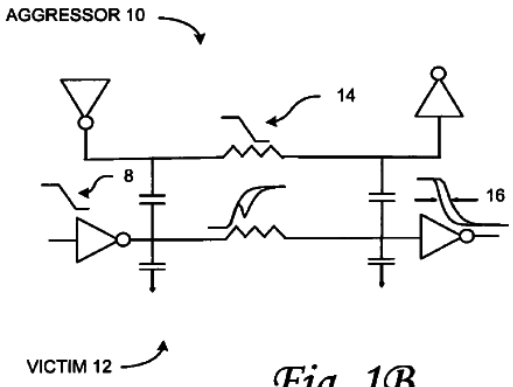


Fig. 1B

Some explanatory text contained in this patent helps illuminate these diagrams:

> Thus, the crosstalk can have two major effects in modern digital circuits: (i) it can cause a strong glitch on a logically static signal, which may lead to functional fault if the glitch is sampled, and (ii) it can impact arrival time of a signal transition by speeding it up or slowing it down. The importance of the crosstalk effects has grown significantly with recent technology advances. For example, crosstalk delay has become a more prominent element of timing analysis during integrated circuit design due to factors such as, (i) increase in coupling-to-total capacitance ratio, (ii) decrease in Supply Voltage resulting in a reduction of gate overdrive (iii) shortening of clock period causing transition waveforms to play a bigger role, and (iv) tighter timing margins requiring more accurate timing analysis and less overestimation of delay. `[page 18]`

Explain what is happening in each diagram, with as much detail as possible.

Specifically, identify where *jitter* occurs in the circuit, and explain its cause.

Challenges

- Identify test equipment useful for diagnosing the problems outlined in these illustrations.

7.1.7 Simultaneous crosstalk

A graphical image from US Patent 7,562,323, "System, Method and Computer Program Product for Handling Small Aggressors in Signal Integrity Analysis", by Xiaoliang Bai and Igor Keller shows a set of logic gates interfering with one another:

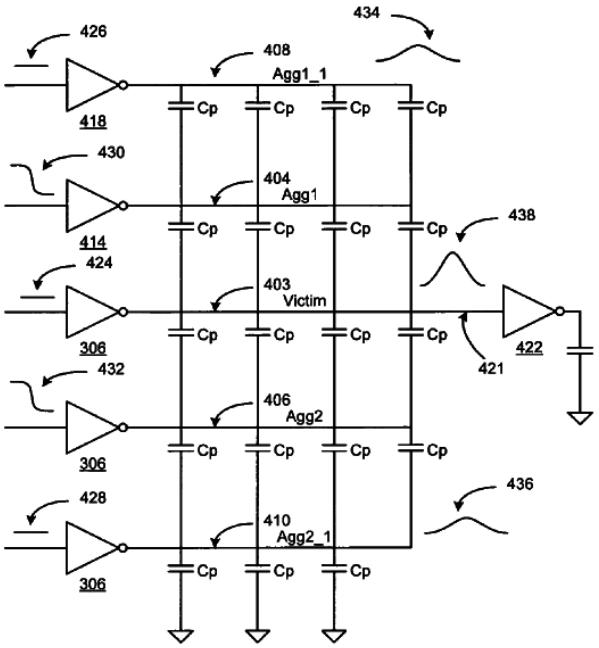


Fig. 8

Identify which line(s) cast interference onto which other line(s) in this diagram, and explain the nature of that interference.

Why are signals 434 and 436 smaller in amplitude than signal 438?

Identify the error(s) potentially caused by this interference.

Identify a means to reduce the interference and thereby avoid the error(s).

Challenges

- Do "victim" lines influence "aggressor" lines, or is it only the other way around?
- Sketch the waveshape you would expect to find on line 404.

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 ($m^3/kg\text{-}s^2$)

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

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

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

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

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

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	

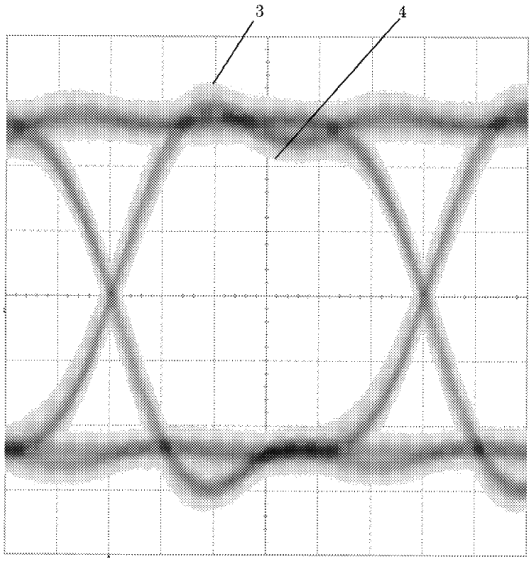
Note how the square-root term (y) is calculated in cell `C1`, and the denominator term (z) in cell `C2`. This makes the two final formulae (in cells `B1` and `B2`) simpler to interpret. The positioning of all these cells on the grid is completely arbitrary[10] – all that matters is that they properly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

Measure the amount of total jitter in this eye diagram, assuming the oscilloscope's timebase is set for 0.5 ns per division:

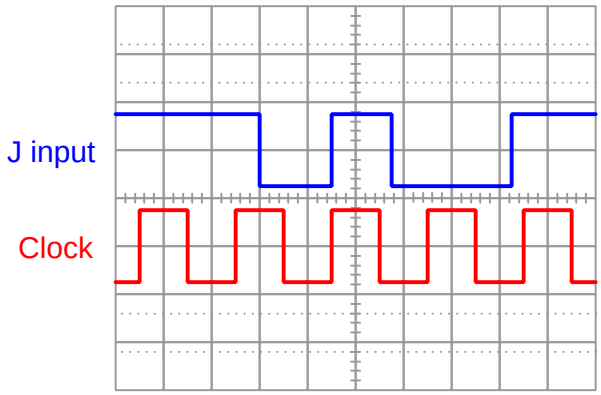


Challenges

- Calculate the frequency of this data stream, assuming alternating 1 and 0 states.
- What do the labels 3 and 4 point to in this eye diagram?

7.2.4 Evaluating set-up and hold times

Suppose a digital oscilloscope captures the following waveforms, one for the J input of a JK flip-flop, and the other for the clock signal (positive-edge):



If the flip-flop under test has an advertised minimum set-up time requirement of 100 ns and hold time requirement of 3 ns, does the J input signal satisfy these conditions? If not, identify *where* it fails.

Challenges

- If any set-up or hold time violations exist in this oscillograph, re-sketch the J input waveform to allow for enough time.

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.

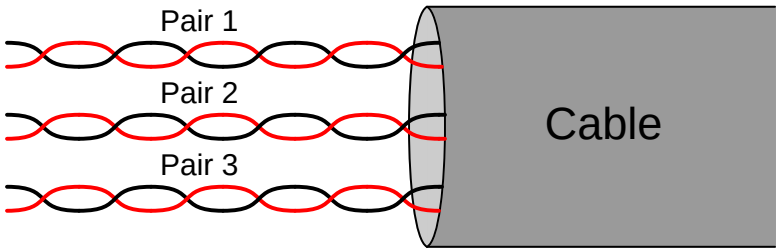
As always, your goal is to fully *explain* your analysis of each problem. Simply obtaining a correct answer is not good enough – you must also demonstrate sound reasoning in order to successfully complete the assignment. Your instructor's responsibility is to probe and challenge your understanding of 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 Testing a cable for magnetic crosstalk

Suppose a long length of multi-pair signal cable is suspected of exhibiting *magnetic crosstalk*, where current traveling through one wire pair magnetically induces electrical noise on an adjacent wire pair. The cable has been removed from service and is wrapped up on a spool awaiting testing. Looking at one end of this cable, you see this:

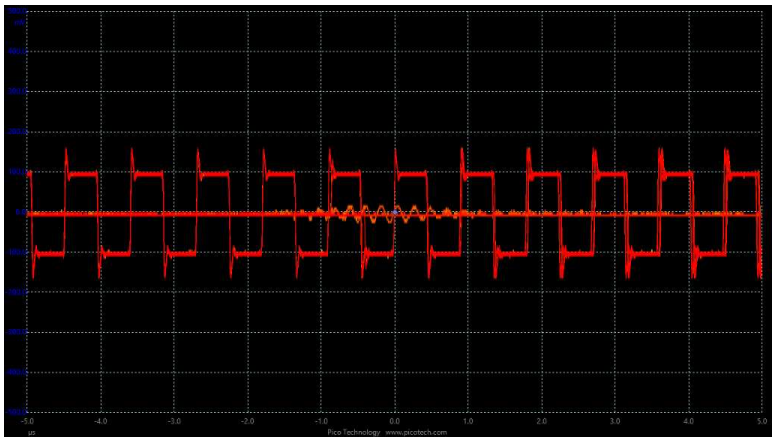


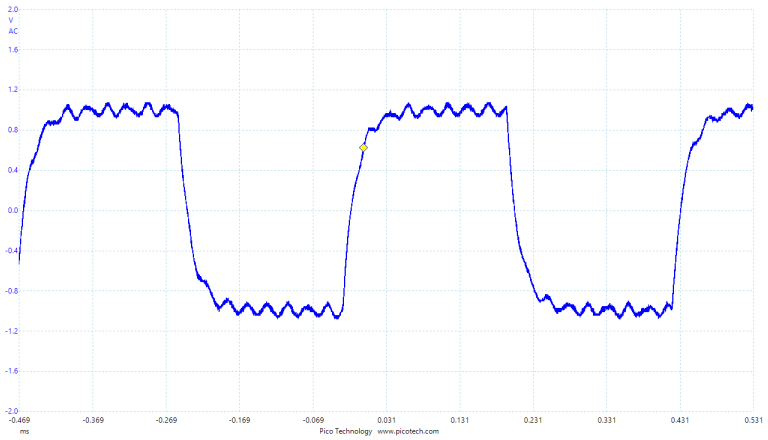
Challenges

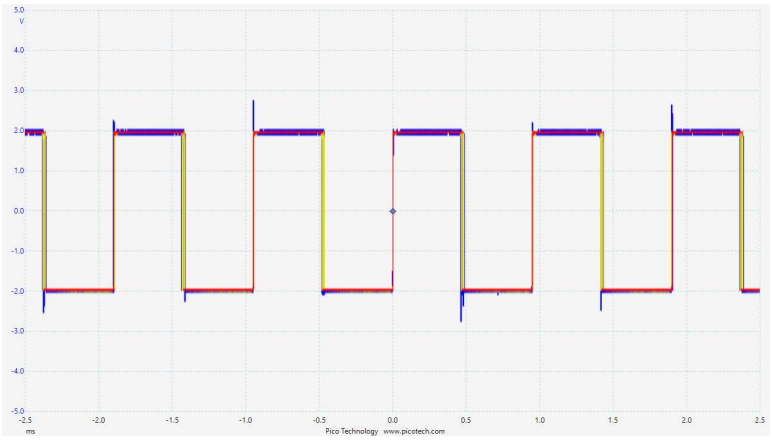
- Describe how you could test this same cable for *electrostatic* crosstalk.

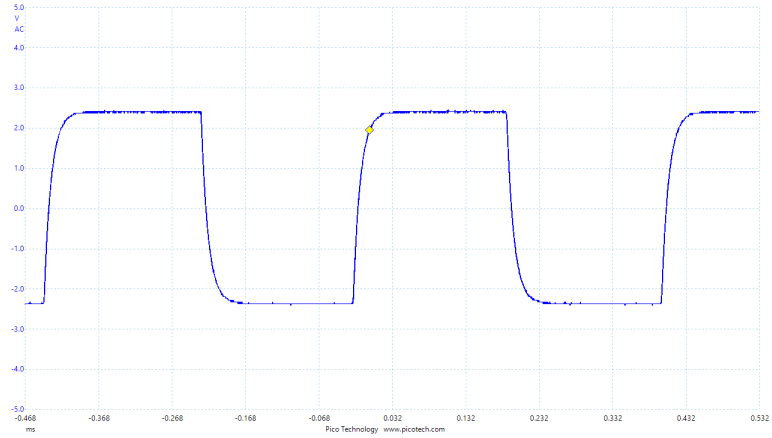
7.3.2 Pulse waveform oscillographs

Identify the major problem(s) you see in each of the oscillographs, quantifying as much of the signal degradation as you can, while also identifying possible causes:









Challenges

- Explain how you might be able to distinguish a case of external interference versus signal degradation unrelated to crosstalk.

Appendix A

Problem-Solving Strategies

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

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

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

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

Appendix B

Instructional philosophy

B.1 First principles of learning

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

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

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

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

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

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

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

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

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

B.2 Proven strategies for instructors

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

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

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

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

B.3 Proven strategies for students

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

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

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

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

B.4 Design of these learning modules

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

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

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

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

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

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

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

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

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

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

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

To high standards of education,

Tony R. Kuphaldt

Appendix C

Tools used

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

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

The `GNU/Linux` computer operating system

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

Bram Moolenaar's `Vim` text editor

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

Donald Knuth's TEX typesetting system

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

Leslie Lamport's LATEX extensions to TEX

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

Tim Edwards' `Xcircuit` drafting program

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

`Gimp` graphic image manipulation program

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

`SPICE` circuit simulation program

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

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

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

`gnuplot` mathematical visualization software

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

`Python` programming language

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

Appendix D

Creative Commons License

Creative Commons Attribution 4.0 International Public License

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

Section 1 – Definitions.

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

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

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

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

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

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

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

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

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

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

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

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

Section 2 – Scope.

a. License grant.

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

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

B. produce, reproduce, and Share Adapted Material.

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

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

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

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

5. Downstream recipients.

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

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

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

b. Other rights.

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

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

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

Section 3 – License Conditions.

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

a. Attribution.

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

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

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

ii. a copyright notice;

iii. a notice that refers to this Public License;

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

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

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

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

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

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

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

Section 4 – Sui Generis Database Rights.

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

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

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

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

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

Section 5 – Disclaimer of Warranties and Limitation of Liability.

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

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

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

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

Section 6 – Term and Termination.

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

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

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

2. upon express reinstatement by the Licensor.

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

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

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

Section 7 – Other Terms and Conditions.

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

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

Section 8 – Interpretation.

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

be made without permission under this Public License.

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

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

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

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

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

Appendix E

References

Bai, Xiaoliang and Keller, Igor, *US Patent 7,562,323*, "System, Method and Computer Program Product for Handling Small Aggressors in Signal Integrity Analysis", application 4 May 2006, patent granted 14 July 2009.

Fleming, John Ambrose, *The Principles of Electric Wave Telegraphy and Telephony*, Second Edition, Longmans, Green, and Co., London, 1910.

"Fundamentals of Signal Integrity" primer, document 54W-22137-1, Tektronix, 2008.

Kidder, Tracy, *The Soul of a New Machine*, Random House, Inc., New York, NY, 1981.

"Logic Guide – Logic Products", Texas Instruments, Dallas, TX, 2017.

Miller, Martin T. *US Patent 9,843,402*, "Noise Analysis To Reveal Jitter And Crosstalk's Effect On Signal Integrity", application 4 October 2016, patent granted 12 December 2017.

"Understanding Data Eye Diagram Methodology for Analyzing High Speed Digital Signals" Application Note AND9075/D revision 1, ON Semiconductor, Semiconductor Components Industries LLC, Denver, CO, June 2015.

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.

23-28 September 2025 – re-organized the "Interference" and "Mitigation techniques" sections of the Tutorial chapter to be more topical. Also, added more instructor notes.

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

7 December 2023 – added comments in the Tutorial chapter regarding problems caused by slow pulse rise times.

8 May 2023 – added some Challenge questions.

4 May 2023 – minor edits to the Introduction chapter and to the Tutorial.

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

1 January 2022 – added comment to the "Mitigation techniques" section of the Tutorial about placing loads with the greatest rates of current change closest to the DC power supply.

13 December 2021 – added photograph of a bus extender board to the "The Soul of a New Machine" Conceptual Reasoning question. Also fixed a minor typographical error in an instructor comment.

11-13 October 2021 – added photograph of IC socket with built-in decoupling capacitor, and added content to the Introduction chapter. Also added a new Case Tutorial section showing an experiment (courtesy of Luke Jones) done with a D-type flip-flop to demonstrate the importance of set-up time, as well as new index entries.

10 May 2021 – commented out or deleted empty chapters.

8 March 2021 – added a page discussing ground bounce to the Tutorial, as well as some instructor notes.

4 March 2021 – minor edits to Introduction and Tutorial, including change made to image_3088 making the noise margins for high and low dimensionally correct.

12 October 2020 – corrected an error in the instructor notes for the "Measuring jitter using an eye diagram" question.

8 October 2020 – minor additions to the Introduction, and minor edits to the Tutorial.

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

1 July 2020 – edited image_3101.eps to be more specific on electric and magnetic flux symbols.

7 March 2020 – added Technical Reference section on digital pulse criteria.

27 February 2020 – added Foundational Concepts to the Conceptual Reasoning section.

23-26 February 2020 – continued writing the Tutorial.

22 February 2020 – added content to the Case Tutorial chapter.

21 February 2020 – began writing the Simplified Tutorial.

20 February 2020 – document first created.

Index

V_{DD} bounce, 28
"Ohm's Law" for a capacitor, 26
"Ohm's Law" for an inductor, 26

Adding quantities to a qualitative problem, 102
Aggressor, 26
Annotating diagrams, 101
Asynchronous, 45

Bounce, ground, 28
Bug, 86

Capacitance, parasitic, 42
Capacitive reactance, 22
Capacitor, decoupling, 29
Checking for exceptions, 102
Checking your work, 102
CMOS, 18, 41
Code, computer, 111
Concentration, doping, 48
Contact, ohmic, 55
Contact, rectifying, 55
Crosstalk, 26

Data General Corporation, 86
Debugging, 86
Decoupling, 29
Decoupling capacitor, 29
Differential signaling, 27
Digital phosphor oscilloscope, 31
Digital signal integrity, 45
Dimensional analysis, 101
DPO, 31

Edwards, Tim, 112
Electromagnetic induction, 26
Electromagnetic radiation, 26
Emissions, radiated, 26
Equivalent series resistance, 29
ESR, 29
Eye diagram, 33

Fall time, 44
Family, logic gate, 18, 40, 41
Faraday's Law of Electromagnetic Induction, 26
FCC, 26
Federal Communications Commission, 26
Filter, 20
Fourier series, 20
Frequency domain, 20

Graph values to solve a problem, 102
Greenleaf, Cynthia, 75
Ground bounce, 28

Harmonic, 20
Holberger, Ken, 86
Hold time, 25, 43
How to teach with these modules, 109
Hwang, Andrew D., 113

Identify given data, 101
Identify relevant principles, 101
Inductance, parasitic, 42
Induction, electromagnetic, 26
Inductive reactance, 22, 28
Integrity, signal, 45
Intermediate results, 101
Inverse exponential functions, 20
Inverted instruction, 109
Inverter gate, 48

Jitter, 24

Kidder, Tracy, 86
Knuth, Donald, 112

Lamport, Leslie, 112
Latch-up, 46
Latch-up, pseudo, 55
Limiting cases, 102
Linear system, 20
Logic analyzer, 31
Logic family gate, 18
Logic gate family, 40, 41
Logic states, 18
LVTTL, 18

Margin, noise, 40, 41
Maxwell, James Clerk, 35
Metacognition, 80
Miller, Martin, 33
Minicomputer, 86
Mixed-signal oscilloscope, 31
Moolenaar, Bram, 111
MSO, 31
Murphy, Lynn, 75

Noise margin, 18, 40, 41
Nonlinear system, 20

Ohmic contact, 55
Open-source, 111
Oscilloscope, digital phosphor, 31
Oscilloscope, mixed-signal, 31

Parasitic capacitance, 42
Parasitic effect, 20, 22, 24, 28, 37
Parasitic inductance, 42
Parasitic transistor, 46
Positive edge triggering, 44
Power plane, PCB, 29
Power supply rail, 18, 41
Problem-solving: annotate diagrams, 101
Problem-solving: check for exceptions, 102
Problem-solving: checking work, 102
Problem-solving: dimensional analysis, 101
Problem-solving: graph values, 102
Problem-solving: identify given data, 101
Problem-solving: identify relevant principles, 101
Problem-solving: interpret intermediate results, 101
Problem-solving: limiting cases, 102
Problem-solving: qualitative to quantitative, 102
Problem-solving: quantitative to qualitative, 102
Problem-solving: reductio ad absurdum, 102
Problem-solving: simplify the system, 101
Problem-solving: thought experiment, 101
Problem-solving: track units of measurement, 101
Problem-solving: visually represent the system, 101
Problem-solving: work in reverse, 102
Propagation delay, 15, 24, 25, 44
Pseudo latch-up, 55
Push-pull amplifier, 48

Qualitatively approaching a quantitative problem, 102

Radiated emissions, 26
Rail, power supply, 18, 41
Reactance, 22, 28
Reading Apprenticeship, 75
Rectifying contact, 55
Reductio ad absurdum, 102, 108, 109
Register, 45
Resonance, 20, 42
Rise time, 44

Schmitt trigger, 23
Schoenbach, Ruth, 75
Scientific method, 80
Set-up time, 14, 25, 43
Shielding, 20
Shunt, 22
Signal integrity, 45
Simplifying a system, 101
Sinking, 20, 48
Socrates, 108
Socratic dialogue, 109
Sourcing, 20, 48
SPICE, 75
Stallman, Richard, 111
Star grounding, 30, 53
Star-point power distribution, 30, 53
Synchronous, 45

The Soul of a New Machine, 86

Thought experiment, 101
Thyristor, 46
Time constant, 20
Time domain, 20
Time, hold, 25
Time, set-up, 14, 25
Timing uncertainty, 24
Toggle mode, 44
Torvalds, Linus, 111
Transistor, parasitic, 46
Transition time, 44
Transmission line, 19, 34
TTL, 18, 40

Uncertainty, timing, 24
Uncertainty, voltage, 23
Units of measurement, 101

Veres, Jim, 86
Via, PCB, 29
Victim, 26
Visualizing a system, 101
Voltage uncertainty, 23

Work in reverse to solve a problem, 102
WYSIWYG, 111, 112