INTRODUCTION TO ENGINEERING - THINKING LIKE AN ENGINEER



Jonathan Compton
Antelope Valley College

LibreTexts™

Introduction to Engineering

Thinking Like an Engineer

Disciplined analytical thinking, technical problem solving, design, and professional practice

Dr. Jonathan Compton, Ph.D.

Antelope Valley College

This text is disseminated via the Open Education Resource (OER) LibreTexts Project (https://LibreTexts.org) and like the thousands of other texts available within this powerful platform, it is freely available for reading, printing, and "consuming."

The LibreTexts mission is to bring together students, faculty, and scholars in a collaborative effort to provide an accessible, and comprehensive platform that empowers our community to develop, curate, adapt, and adopt openly licensed resources and technologies; through these efforts we can reduce the financial burden born from traditional educational resource costs, ensuring education is more accessible for students and communities worldwide.

Most, but not all, pages in the library have licenses that may allow individuals to make changes, save, and print this book. Carefully consult the applicable license(s) before pursuing such effects. Instructors can adopt existing LibreTexts texts or Remix them to quickly build course-specific resources to meet the needs of their students. Unlike traditional textbooks, LibreTexts' web based origins allow powerful integration of advanced features and new technologies to support learning.



LibreTexts is the adaptable, user-friendly non-profit open education resource platform that educators trust for creating, customizing, and sharing accessible, interactive textbooks, adaptive homework, and ancillary materials. We collaborate with individuals and organizations to champion open education initiatives, support institutional publishing programs, drive curriculum development projects, and more.

The LibreTexts libraries are Powered by NICE CXone Expert and was supported by the Department of Education Open Textbook Pilot Project, the California Education Learning Lab, the UC Davis Office of the Provost, the UC Davis Library, the California State University Affordable Learning Solutions Program, and Merlot. This material is based upon work supported by the National Science Foundation under Grant No. 1246120, 1525057, and 1413739.

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation nor the US Department of Education.

Have questions or comments? For information about adoptions or adaptions contact info@LibreTexts.org or visit our main website at https://LibreTexts.org.

The LibreTexts name, logo, and book covers are trademarked by LibreTexts, Inc. (a 501(c)(3) not-for-profit organization) and protected against unauthorized use.

This text was compiled on 08/26/2026

TABLE OF CONTENTS

Licensing

Preface

1: What is Engineering?

1.1: The Engineer You Don't Notice

1.2: A Working Definition

1.3: Engineering vs Science vs Technology

1.4: What Engineers Actually Do

1.5: The Engineering Mindset

1.6: What Engineers Actually Do

1.7: Major Disciplines at a Glance

1.8: Ethics Is Not a Separate Chapter

1.9: What This Course Will Build

1.10: Summary

1.11: Quick Check

2: Engineering Disciplines and Careers

2.1: The Question Most Students Ask

2.2: The Major Engineering Disciplines

2.3: Most Real Problems Cross Discipline Boundaries

2.4: Engineering by the Numbers

2.5: Functional Roles

2.6: Choosing a Discipline

2.7: One Obligation That Every Discipline Shares

2.8: Summary

2.9: Quick Check

3: Academic Pathways and Transfer Strategy

3.1: Engineering Is a Structured Academic Path

3.2: The Mathematics Backbone

3.3: Physics as Applied Mathematics

3.4: Typical Lower-Division Engineering Path (Community College)

3.5: Degree Levels and Transfer Destinations

3.6: Transfer Strategy

3.7: Academic Risk Factors

3.8: Treating Your Academic Plan Like an Engineering Problem

3.9: Summary

3.10: Quick Check

4: Engineering Success Systems

4.1: It Is Not About Talent

4.2: Understanding Your Real Workload

4.3: How to Actually Study Engineering

4.4: Understanding Your Errors

4.5: Goals, Humility, and the Long Game

4.6: Engineering Teamwork

4.7: Sustaining Yourself Over the Long Haul

4.8: Summary

4.9: End-of-Chapter Exercise

5: Units and Dimensional Analysis

5.1: Why Units Matter

5.2: Quantities vs. Numbers

5.3: SI System — Base and Derived Units

5.4: Unit Conversion Method

5.5: SI Prefixes

5.6: Dimensional Analysis

5.7: Units in Computational Models

5.8: Order-of-Magnitude Reasoning

5.9: From Units to Everything That Follows

5.10: Summary

5.11: End-of-Chapter Problem Set

6: Structured Engineering Problem Solving

6.1: Why Structure Matters

6.2: Full Workflow Worked Example First

6.3: Step 1 — Define

6.4: Step 2 — Model

6.5: Step 3 — Analyze

6.6: Step 4 — Interpret + Sensitivity

6.7: Step 5 — Communicate

6.8: Five Common Failure Patterns

6.9: The Workflow Across Disciplines

6.10: Structure and Professional Accountability

6.11: Summary

6.12: End-of-Chapter Problem Set

7: The Engineering Design Process

7.1: Analysis vs Design

7.2: What Makes a Design Problem

7.3: Seven-Step Design Framework

7.4: Define the Design Objective

7.5: Identify Constraints
7.6: Generate Alternatives
7.7: Model and Analyze
7.8: Evaluate Trade-offs
7.9: Iterate and Refine
7.10: Communicate
7.11: Design, Ethics, and Safety Factors
7.12: Summary
7.13: Quick Check

8: Spreadsheet Modeling

8.1: From Calculator to Model
8.2: Structuring an Engineering Spreadsheet
8.3: Absolute vs Relative References
8.4: Parameter Sweeps
8.5: Graphical Visualization
8.6: Conditional Formatting and IF Logic
8.7: Six Common Spreadsheet Errors
8.8: Spreadsheets as Design Exploration Tools
8.9: When Spreadsheets Are Not Enough
8.10: Summary
8.11: End-of-Chapter Problem Set

9: Data Visualization for Engineers

9.1: Why Graphs Are an Engineering Tool
9.2: Chart Type Selection
9.3: Anatomy of an Engineering Graph
9.4: Linear vs Nonlinear Trends
9.5: Misleading Visualizations
9.6: Logarithmic Scales
9.7: Multi-Curve Comparison
9.8: Visualization and Interpretation
9.9: Advanced Chart Features
9.10: Graph Checklist — Quick Reference
9.11: Summary
9.12: Quick Check

10: MATLAB for Engineering Computation

10.1: Why Engineers Program
10.2: MATLAB as Structured Engineering
10.3: Script Basics
10.4: Four-Step Build
10.5: Trigonometry in MATLAB

10.6: Conditional Logic
10.7: Logical Indexing
10.8: Programming Project- Cable Car in MATLAB
10.9: Choosing the Right Tool
10.10: Summary
10.11: End-of-Chapter Problem Set

11: Circuit Fundamentals

11.1: Circuits as Energy Systems
11.2: Physical Meaning
11.3: Ohm's Law
11.4: Power
11.5: Schematic Symbols
11.6: Series Circuits
11.7: Parallel Circuits
11.8: Series vs Parallel Comparison
11.9: Kirchhoff's Laws
11.10: Power in Networks
11.11: Mixed Circuits
11.12: LED Design
11.13: Circuit Simulator
11.14: Summary
11.15: End-of-Chapter Problem Set

12: Parametric CAD and 3D Modeling

12.1: CAD Is Not Drawing
12.2: Parametric Modeling Core Concepts
12.3: From Sketch to Solid
12.4: The Sketch as a Mathematical Definition
12.5: The Fusion Project
12.6: Modeling Workflow
12.7: Five Common Modeling Errors
12.8: CAD in the Engineering Landscape
12.9: Summary
12.10: End-of-Chapter Problems

13: Engineering Communication

13.1: Communication Is Part of the Deliverable
13.2: Three Modes of Communication
13.3: The Technical Memo
13.4: Complete Memo Example
13.5: Quantitative Precision
13.6: Graphs and Tables

13.7: Technical Presentations
13.8: TEGO Case Study
13.9: Professional Email
13.10: Communicating Uncertainty and Limitations
13.11: Summary
13.12: End-of-Chapter Applied Exercises

14: Engineering Ethics

14.1: Ethics Is Not an Add-On
14.2: NCEES Model Rules
14.3: Ethics HW Preview Table
14.4: Safety Factors
14.5: Modeling Assumptions
14.6: Eight-Step Framework
14.7: Ethics HW Scenario 1
14.8: Ethics HW Scenario 2
14.9: Ethics HW Scenario 3
14.10: Academic Integrity
14.11: Citicorp Case Study
14.12: Environmental Responsibility
14.13: Summary
14.14: Quick Check

15: Integrated Engineering Design Challenge

15.1: This Is Not a New Topic
15.2: Challenge Specifications
15.3: Define- Units, Loads, What "Best" Means
15.4: Model- Tension, Compression, Straws
15.5: Mass Budget
15.6: Parameter Sweep
15.7: Alternatives and Decision Matrix
15.8: Iterate
15.9: Bridge Report
15.10: Course Map
15.11: Summary
15.12: End-of-Chapter Problem Set

16: Appendices

16.1: Appendix A — Standard Symbols and Units
16.2: Appendix B — Unit Conversion Reference Table
16.3: Appendix C — MATLAB Quick Reference
16.4: Appendix D — Spreadsheet Quick Reference (Excel)
16.5: Appendix E — Technical Memo Template

Index
Glossary
Detailed Licensing
Detailed Licensing

This page titled Introduction to Engineering - Thinking Like an Engineer is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by Jonathan Compton.

Licensing

A detailed breakdown of this resource's licensing can be found in ***Back Matter/Detailed Licensing***.

This page titled Licensing is shared under a license and was authored, remixed, and/or curated by .

Preface

Preface

Introduction to Engineering: Thinking Like an Engineer

Dr. Jonathan Compton — Antelope Valley College

You woke up this morning because of engineers.

The alarm that woke you runs firmware written by a software engineer. The building you are sitting in was designed by a structural engineer and wired by an electrical engineer. The phone in your pocket required aerospace engineers to launch the satellites that give it a signal, materials engineers to develop the glass on its screen, and manufacturing engineers to build it at scale. The road you took to get here was designed by civil engineers. The power grid that lit this room was built by power engineers. The water you drank this morning passed through infrastructure that took a generation of engineers to design, build, and maintain.

You are already living inside engineering. This course is about learning to think the way the people who built it think.

What This Book Is About

This book teaches engineering reasoning — the structured, disciplined, evidence-based way of approaching problems that is common to every engineering discipline, from aerospace to circuits to software to structures.

The spine of the book is a five-step workflow: **Define → Model → Analyze → Interpret → Communicate.** You will see this framework for the first time in Chapter 6. You will use it in every chapter and every project after that — when you build a spreadsheet model, write a MATLAB script, analyze a circuit, model a part in Fusion 360, present a solution to a non-technical audience, and evaluate an ethical dilemma. The workflow does not change. The domain does.

By the time you reach the Bridge Challenge in Chapter 15, you will have applied every tool in this book — unit conversion, structured problem solving, spreadsheet modeling, data visualization, MATLAB, circuit analysis, parametric CAD, engineering communication, and ethics — simultaneously, to a single physical problem made of plastic straws and tape. That is what integration means. That is what engineering does.

What You Will Build

Introduction to Engineering is not a lecture-only course. The projects are where the learning happens:

- You will build a parametric spreadsheet model of a cable car's velocity, using piecewise equations and conditional formatting,

and write a technical memo explaining what you found.

- You will automate that same model in MATLAB and compare the two approaches in writing.
- You will simulate and analyze electrical circuits using a browser-based circuit simulator, validate Ohm's Law experimentally, and design a current-limiting resistor for an LED.
- You will model nine geometric solids in Fusion 360 — from a 10 mm cube to a sphere to a cube with a hole — verify each model's mass against a hand calculation, and produce an ASME engineering drawing.
- You will present a technical solution to a manufacturing quality problem to a non-technical management audience.
- You will analyze three real ethical scenarios using the NCEES Model Rules of Professional Conduct and produce written arguments for each.
- You will design and build a bridge from plastic straws, evaluate candidate geometries using a decision matrix, calculate your mass budget, and test your bridge to failure.

These projects are not additions to the course. They are the course. The chapters prepare you for them.

A Note on Who This Book Is Written For

Engineering students at community colleges look different from the national image of "an engineering student." Some of you are working 20 or more hours a week. Some of you are the first person in your family to attend college. Some of you are returning after time away. Many of you are not sure yet whether you belong here.

You do.

The technical skills in this book — unit conversion, structured calculation, spreadsheet modeling, programming, circuit analysis, CAD, communication, ethics — are learnable. They are not innate. The engineers who built the world you woke up in were first-year students who did not yet know these things. They learned them, one problem set and one project at a time.

So will you.

How to Use This Book

Each chapter is written to be read *before* the material is needed, not during or after. The chapters are short for a reason — they introduce concepts clearly and leave the deep practice to the projects and problem sets.

The **Worked Examples** show you what a complete solution looks like before you attempt the problems. The **Watch Out** callouts identify the errors that most commonly produce wrong answers without any warning. The **Connection** boxes tell you explicitly where each concept appears in the course projects, so you understand why you are learning something before you need it.

The **Appendices** at the back of the book are reference tools: standard symbols and units, conversion factors, a MATLAB quick

reference, a spreadsheet quick reference, and a blank technical memo template.

If you are ever unsure what a term means, check the **Glossary**. Every key term is defined there with the chapter in which it is introduced.

A Note on Ethics

Engineering ethics appears throughout this book — not only in Chapter 14. The Challenger O-ring case is in Chapter 9, because misleading graphs are an engineering ethics problem. The safety factor discussion is in Chapter 7, because building with inadequate margin is an engineering ethics problem. The limitation statements in Chapter 13 are an engineering ethics practice, because omitting a known constraint from a report is misleading communication.

This is intentional. Engineering ethics is not a module at the end of a curriculum. It is the discipline of making technical decisions responsibly. It shows up every time you decide whether to check your answer, whether to disclose an assumption, whether to report what actually happened. The habits you build in this course are the same habits that govern professional practice.

Acknowledgments

I wrote this book because I believe engineering education should be accessible — not just financially, but intellectually. Every student who completes this course should leave with a clear, confident understanding of how engineers think and what engineers do. That is the standard I wrote to.

This book grew out of the ENGR 110 curriculum developed and refined at Antelope Valley College. I especially want to acknowledge Karl Major, whose work developing course curriculum, project structure, instructional materials, and classroom activities helped shape much of the foundation from which this textbook was built. Many of the themes in this book — engineering thinking, applied problem solving, modeling, design, communication, and hands-on project work — reflect curriculum developed through that shared instructional effort.

I also acknowledge the use of artificial intelligence tools during the development of this textbook. AI was used to help organize material, transform class notes and course resources into readable textbook prose, improve formatting consistency, draft captions and alt text, and revise HTML for publication in LibreTexts. These tools were used as drafting and production aids, not as independent sources of authority. The final organization, technical content, examples, instructional framing, and editorial decisions were reviewed, revised, and approved by the author.

Dr. Jonathan Compton
Antelope Valley College
2026

This page titled Preface is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

1: WHAT IS ENGINEERING?

What Is Engineering?

Engineering is mostly invisible. When it works, you don't notice it. That invisibility is not an accident — it is the goal.

Learning Objectives

By the end of this chapter, you will be able to:

- Define engineering as a disciplined way of thinking, not a job description.
- Distinguish engineering from science and from technology.
- Describe how engineered systems shape daily life and why public trust matters.
- Trace, at a high level, how engineering knowledge has built on itself across history.
- Identify the core habits of the engineering mindset.
- Recognize that ethics is embedded in engineering practice, not added on top.

1.1: The Engineer You Don't Notice
1.2: A Working Definition
1.3: Engineering vs Science vs Technology
1.4: What Engineers Actually Do
1.5: The Engineering Mindset
1.6: What Engineers Actually Do
1.7: Major Disciplines at a Glance
1.8: Ethics Is Not a Separate Chapter
1.9: What This Course Will Build
1.10: Summary
1.11: Quick Check

This page titled 1: What is Engineering? is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

1.1: The Engineer You Don't Notice

1.1 The Engineer You Don't Notice

You woke up this morning because of engineers.

The alarm that went off — someone designed the oscillator circuit inside it. The phone it came from connected through a cellular protocol that a team of communications engineers specified down to the microsecond. The water that came out of your shower at a usable temperature passed through a municipal treatment system sized by civil and environmental engineers, pumped through pipes rated for a specific pressure by mechanical engineers, and heated by a unit whose combustion efficiency was optimized by thermal engineers. The coffee you may or may not have had passed through a machine whose heating element was designed to fail *safely* if left on overnight.

You probably noticed none of this. That is the point.

Engineering is mostly invisible. When it works, you don't think about it. You only notice engineering when it fails — when a bridge closes, when a phone battery catches fire, when a website goes down on a holiday weekend.

That invisibility is not an accident. It is the goal. An engineer's job, done well, is to make something complicated behave simply and safely enough that you can ignore it.

So when people say "I want to be an engineer because I like building things," they are not wrong — but they are describing the tip of the iceberg. Building things is what gets photographed. The real work is figuring out *how* to build them so that they keep working, under pressure, for years, for strangers who will never know your name.



Figure 1.1.1: Everyday morning activities depend on many engineered systems that usually remain unnoticed, including phone electronics, wireless communication networks, water treatment and distribution, thermal control, transportation systems, data

centers, bridges, batteries, and the electric power grid. (Copyright; Jonathan Compton, AI-assisted original figure, CC BY-NC 4.0).

This page titled 1.1: The Engineer You Don't Notice is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

1.2: A Working Definition

1.2 A Working Definition

Let's put a stake in the ground.

> **Engineering** is the disciplined application of mathematics, science, and design to create systems that serve human needs under real-world constraints.

Every word in that sentence is earning its place. Let's unpack it.

Disciplined. Engineering is not guesswork, intuition, or inspiration — though all three show up in the process. It is a structured way of making decisions that other engineers can check, reproduce, and trust.

Mathematics and science. Engineering borrows its rules from physics, chemistry, biology, and mathematics. An engineer cannot overrule conservation of energy or Newton's second law. The laws of nature are the non-negotiable boundary conditions of every engineering problem.

Design. This is the creative part. Given a problem and a set of constraints, there is almost never one correct answer. There are better and worse answers, and the engineer's judgment is what separates them.

Systems. Engineers rarely build *things* in isolation. They build systems — a bridge is not just a beam, it is a load path; a phone is not just a screen, it is a stack of hardware, firmware, operating system, and apps. Thinking in systems is a habit you will develop throughout this course.

Human needs. Engineering is purposeful. It exists because people need things — shelter, transportation, communication, clean water, medicine. The need comes first; the technology follows.

Real-world constraints. This is where engineering becomes hard. Every project has limits: budget, schedule, materials, regulations, the laws of physics themselves. A design that ignores its constraints is not a solution — it is a wish.

This page titled 1.2: A Working Definition is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

1.3: Engineering vs Science vs Technology

1.3 Engineering Is Not Science (and Not Technology)

Students often blur these three words together. They are related, but they are not the same thing. Keeping them straight will help you understand where your work fits.

Comparison of science, engineering, and technology by purpose, output, measure of success, and example.

	Science	Engineering	Technology
Core question	*Why does this happen?*	*How can we use this safely and reliably?*	*What can this do?*
Primary output	Knowledge, laws, theories	Systems, designs, analyses	Products, tools, capabilities
Success looks like	An accurate explanation	A working solution within constraints	A useful artifact people adopt
Example	A physicist measures how steel deforms under load	A structural engineer uses that data to design a bridge	The bridge, once built, is a piece of transportation technology

Neil Armstrong — a mechanical and aeronautical engineer before he was an astronaut — put it simply:

> *"Science is a quest for truth for its own sake… Engineering turns those explanations and understandings into new or improved machines, technologies, or processes."*

Engineering *depends* on science. Every equation you will use in this course was discovered by scientists. But engineering's job is different: engineers are judged by whether their designs work, not by whether they discovered something new.

> **Real-World Example — The Mars Climate Orbiter**
>
> In 1999, NASA lost a $327 million spacecraft called the Mars Climate Orbiter. It did not crash because of bad science. Gravity worked exactly as Newton said it would. It crashed because one engineering team provided thrust data in pound-seconds (US units) while another team's navigation software expected newton-seconds (SI units). The spacecraft flew too close to Mars and was destroyed in the upper atmosphere.

> Science was not the failure. Engineering discipline was. Every number was correct inside its own unit system — but engineering requires that the *entire system* be consistent. You will study units in Chapter 5 for exactly this reason.

This page titled 1.3: Engineering vs Science vs Technology is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

1.4: What Engineers Actually Do

1.4 What Engineers Actually Do All Day

If you shadow a working engineer, you will not find them "doing math" for eight hours. The job is more varied — and, honestly, more interesting — than most students expect.

A typical engineer's week includes:

- **Defining problems.** Turning a vague request ("we need the battery to last longer") into a specific, measurable problem ("increase runtime from 8 hours to 12 hours without exceeding the current mass budget").
- **Modeling.** Building a simplified representation of a real system — equations, spreadsheets, CAD models, simulations — that lets you predict behavior before building anything.
- **Analyzing.** Running the model, checking results, comparing options.
- **Designing.** Choosing component values, geometry, materials, or algorithms that satisfy the constraints.
- **Testing.** Prototyping, measuring, comparing real behavior to predicted behavior.
- **Documenting.** Writing memos, calculations, and specifications so that someone else — a colleague, a regulator, a future engineer — can understand and trust the work.
- **Communicating.** Explaining technical decisions to non-engineers: managers, clients, the public.

Notice that writing and communication are on that list. Engineers who cannot explain their work — in plain language, in writing — end up having their ideas overruled by people who *can*.

> 🔗 **Connection**
>
> The **Career Preparation Project** in this course will walk you through building a resume, identifying internships, and interviewing real engineers about their day-to-day work. If you've ever wondered "but what is the job *actually* like?" — that project is where you find out.

This page titled 1.4: What Engineers Actually Do is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

1.5: The Engineering Mindset

1.5 The Engineering Mindset

Technical knowledge will come from your calculus, physics, and engineering courses over the next few years. But there is a *way of thinking* that transfers across every discipline — from civil to electrical to software — and it is worth naming early, because you will start practicing it this semester.

The engineering mindset has five recurring habits.

1. Define before you solve. Before touching an equation, an engineer asks: *What exactly is the problem? What is given? What is unknown? What does success look like?* Students who skip this step spend hours solving the wrong problem. You will see this framework again in Chapter 6.

2. Model, then analyze. Real systems are messy. An engineer builds a simpler representation — an idealized version of reality — and analyzes *that*. The art is choosing which details matter and which can be ignored without breaking the answer.

3. Check your units. This sounds mundane. It is not. Most expensive engineering failures trace back to a unit mistake, a sign error, or a decimal in the wrong place. Professional engineers are obsessive about dimensional consistency because it is one of the cheapest forms of error-catching ever invented.

4. Expect to iterate. The first design is almost never the last. Engineers expect to build, test, fail, learn, and revise — usually multiple times. "Failure" in this sense is not embarrassment; it is data.

5. Think about who is affected. Every engineering decision has consequences for people who never signed off on it: future users, neighbors, workers, the environment. Professional engineers build the habit of asking *who else is in this decision?* before they finalize it.

> **⚠ Watch Out — "I'm just not a math person"**
>
> Students sometimes arrive in engineering thinking their only hurdle is the math. It's not. The math is learnable, and you will learn it. The harder shift is learning to *think* like an engineer — to be comfortable with messy problems, partial information, and answers that are "good enough" rather than "provably correct." Students who master the mindset usually find the math follows. Students who only grind on math, without building the mindset, tend to hit a wall in upper-division courses where problems stop having single correct answers.

This page titled 1.5: The Engineering Mindset is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

1.6: What Engineers Actually Do

1.6 A Brief History of Engineering

Engineering is older than the word itself. Every civilization that survived a winter, crossed a river, or outlasted its neighbors did so because someone figured out how to build things that worked. What follows is not a complete history — that would fill its own book. It is a tour of eight eras, chosen to show how engineering knowledge builds on itself. Every tool you will use in this course descends from something in this story.

Ancient Era (before ~500 BCE) — Engineering as Applied Intuition

The earliest engineering had no textbooks and no formal training. It was passed down through apprenticeship, trial, and error. But it was unmistakably engineering: structured problem-solving under constraints of material, labor, and time.

The Great Pyramid of Giza, completed around 2560 BCE, is made of roughly 2.3 million limestone blocks, each weighing 2.5 to 15 tons. Building it required solving transportation, quarrying, leveling, and scheduling problems at a scale no one had attempted before. The Egyptians did this without iron tools, pulleys as we know them, or written mathematics in the modern sense. They did it with ramps, copper chisels, organized labor, and a remarkable understanding of geometry — the base of the Great Pyramid is level to within about 2 centimeters across more than 230 meters on a side.

Similar stories unfolded across the ancient world:

- **Mesopotamian ziggurats** and early **canal systems** moved water across long distances, making large-scale agriculture possible in arid regions.
- **Chinese hydraulic works**, including early flood-control systems along the Yellow River, coordinated labor across whole provinces.
- **Minoan plumbing** in Crete — with clay pipes, drainage systems, and even flushing toilets — existed around 1800 BCE.
- **Indus Valley cities** like Mohenjo-Daro had grid-planned streets and centralized drainage roughly 4,500 years ago.

The common thread: engineering existed long before "engineers" did. The knowledge was craft knowledge, held by builders, stonemasons, and water-managers, and it was effective enough to build cities that still stand.



Figure 1.6.1: The Pyramids of Giza consist of the Great Pyramid of Giza (known as the Great Pyramid and the Pyramid of Cheops or Khufu -- left), the somewhat smaller Pyramid of Khafre (or Chephren) a few hundred meters to the south-west (right), and the relatively modest-sized Pyramid of Menkaure (or Mykerinos) a few hundred meters further south-west (not in photo).. (CC BY-SA 3.0, Ad Meskens, Wikimedia Commons)

Classical Engineering (~500 BCE – 500 CE) — Writing It Down

The Greeks and Romans did something new: they wrote engineering down.

Greek engineers, drawing on the mathematics of Euclid and Archimedes, began to treat mechanical problems as problems of *principles* rather than just recipes. Archimedes' work on levers, buoyancy, and the screw pump gave mechanical advantage a theoretical basis. Hero of Alexandria documented dozens of machines — steam-powered novelties, surveying instruments, automatic doors — in writings that survived for centuries.

Roman engineers took these ideas to industrial scale. Rome built:

- A road network of roughly 400,000 kilometers across Europe, North Africa, and the Middle East.
- Aqueducts that delivered clean water across distances of more than 50 kilometers, dropping only a few meters per kilometer in elevation — a feat of surveying and construction that depended on careful leveling.
- Concrete (specifically pozzolanic concrete) that hardens underwater and has survived 2,000 years of exposure, outperforming much modern concrete.
- The Colosseum, aqueduct arches, and the Pantheon's 43-meter unreinforced concrete dome — a span not exceeded for 1,300 years.

Most importantly, around 15 BCE, the Roman architect-engineer Vitruvius compiled *De Architectura*, a ten-volume work covering building, machinery, water systems, and materials. It was the first comprehensive engineering textbook, and it remained a primary reference for 1,500 years. The key idea of this era: **engineering knowledge could be written down, taught, and**

improved by those who came after.

Medieval and Islamic Golden Age (~500 – 1400 CE) — Knowledge Preserved and Advanced

When the Western Roman Empire fell, a common narrative says engineering stagnated. That's only partially true — and mostly only in Western Europe.

In the Islamic world, from roughly 800 to 1200 CE, engineering flourished. Baghdad's House of Wisdom translated Greek and Roman texts into Arabic, preserving Vitruvius and others that might otherwise have been lost. Islamic engineers extended that knowledge:

- **Al-Jazari's** 13th-century *Book of Knowledge of Ingenious Mechanical Devices* described water clocks, pumps, and automated mechanisms with precision drawings — the ancestor of the modern engineering drawing.
- **Windmills** were developed in Persia for grinding grain and pumping water.
- **Sophisticated irrigation systems** (qanats) moved groundwater across dry landscapes.
- **Surveying, cartography, and navigation** advanced to support trade and pilgrimage.

In East Asia, Chinese engineers independently developed cast iron production, the magnetic compass, block printing, gunpowder, and the mechanical clock.

Western Europe wasn't idle either. Gothic cathedrals — Notre-Dame (begun 1163), Chartres, Cologne — are stunning structural engineering: flying buttresses transfer roof loads outward along visible, calculable load paths, freeing the walls to hold enormous stained-glass windows. The builders didn't have formal structural theory. But they had 300 years of accumulated craft experience, and they could feel when a design was at its limit.

The key idea of this era: **engineering knowledge was never monolithic.** When one region's institutions faltered, others preserved and advanced what was known. Modern engineering inherits from all of these threads.

Renaissance (~1400 – 1700 CE) — Engineering Meets Science

The Renaissance did not invent engineering, but it did something important: it began to formally connect engineering practice to mathematical theory.

Three developments drove this:

1. **The printing press** (Gutenberg, ~1440) made technical knowledge cheap to copy and widely distributable. Vitruvius was printed in 1486. Within a century, engineering manuals circulated across Europe.
2. **Polymath practitioners** — builders, military engineers, and inventors who were also artists and mathematicians — blurred the line between craft and science. Leonardo da Vinci's notebooks contain sketches of bridges, cranes, flying machines, and parachutes, analyzed with the same care he gave to anatomy and painting.
3. **The scientific revolution** provided new tools. Galileo Galilei applied mathematics to the strength of beams in 1638, arguing

that structural behavior could be *predicted* from material properties and geometry. Isaac Newton's *Principia* (1687) gave engineers the mathematical machinery to analyze motion, force, and gravitation.

This was a pivotal shift. For the first time, engineers could ask *what will happen?* before building, and get a quantitative answer. Early analytical tools — beam theory, statics, basic fluid mechanics — were crude by modern standards, but they were the seeds of what we now call engineering analysis.

The key idea of this era: **engineering began to marry to formal mathematics.** Trial and error would never go away, but it was no longer the only tool.

The Birth of the Profession (1700s) — Engineering Gets a Name

In 1771, the British engineer John Smeaton coined the term **civil engineer** — to distinguish his work on canals, bridges, and harbors from the work of military engineers, who had organized themselves as a profession much earlier.

Around this formalization, a set of distinct shifts took place:

- **Formal engineering schools opened.** France's École des Ponts et Chaussées dated to 1747; the École Polytechnique followed in 1794. These institutions trained engineers through a rigorous sequence of mathematics, mechanics, and drawing.
- **Professional societies emerged.** The Institution of Civil Engineers was founded in London in 1818 — the first professional engineering society in the world.
- **Engineering became credentialed.** Before this era, an engineer was someone who built things well enough that people paid them. After it, an engineer was someone with formal training, recognized by peers.
- **Engineering became a career with identity.** The word "engineer" shifted from describing a role to describing a professional.

The key idea of this era: **modern engineering as a profession is only about 250 years old.** Every engineering society, every accredited degree, every PE license descends from this moment.

The Industrial Revolution (1800s) — Specialization

The 1800s changed what engineers could build, and in doing so, they changed what engineering *was*.

Steam power — made practical by Watt's improvements to the Newcomen engine in the 1760s — scaled up into locomotives, steamships, and factory power systems. Iron and, later, steel (the Bessemer process, 1856) made possible structures that previous centuries couldn't have imagined: railway bridges, skyscrapers, ocean liners. Electricity moved from scientific curiosity to industrial utility, thanks to work by Faraday, Maxwell, Edison, and Tesla. Industrial chemistry — Haber, Bessemer, Solvay — transformed agriculture, metallurgy, and manufacturing.

No single engineer could master all of this. So the profession specialized.

- **Mechanical engineering** emerged around the steam engine and industrial machinery. The American Society of Mechanical Engineers (ASME) was founded in 1880.
- **Electrical engineering** emerged as electrical distribution, motors, and telegraphy became industries. The American Institute of Electrical Engineers (AIEE) was founded in 1884.
- **Chemical engineering** emerged as continuous-process chemistry — refineries, large-scale synthesis — required people who understood both chemistry and process design. MIT established the first chemical engineering degree program in 1888.
- **Mining, metallurgical, and industrial engineering** followed, each tied to a specific economic need.

The key idea of this era: **when one discipline became too broad for one person, disciplines specialized.** You are about to see that pattern repeat.



Figure 1.6.2: Forth Bridge. (CC BY-SA 2.0, Graeme Yuill, Wikimedia Commons)

The 20th Century (1900s) — The Discipline Explosion

Every major technology of the 20th century spawned a new engineering discipline to handle its complexity.

- **1903:** The Wright brothers' powered flight begins aerospace engineering. The first university aerospace degree is offered at MIT in 1914.
- **1930s–40s:** Rocketry, radar, and nuclear technology drive massive wartime engineering efforts — most visibly the Manhattan Project, which applied engineering management techniques at a scale no civilian project had seen.
- **1940s–50s:** The invention of the transistor (1947) and the integrated circuit (1958) launches electronics and eventually computer engineering as distinct disciplines.
- **1950s–60s:** Software engineering begins to emerge as something separate from computer science — programs

have become large enough that building them requires its own engineering discipline. The term "software engineering" is coined at a NATO conference in 1968.
- **1960s:** The Apollo program applies systems engineering on an unprecedented scale: a stack of over 400,000 people and hundreds of thousands of requirements, all traceable through a formal management system.
- **1970s–80s:** Biomedical engineering matures — dialysis, pacemakers, imaging, prosthetics — as engineering methods cross into medicine.
- **1980s–90s:** Environmental engineering and sustainability-focused engineering grow in response to pollution, regulation, and emerging climate concerns.

By 2000, the engineering profession had fractured into dozens of recognized disciplines and hundreds of sub-specialties. And yet every one of them still traced back to the same five-step way of thinking you are about to learn: define, model, analyze, iterate, communicate.

The key idea of this era: **every major 20th-century technology triggered a new engineering discipline to handle its complexity.** The pattern of specialization set in the 1800s accelerated dramatically.

The 21st Century (2000s – today) — New Frontiers

The pattern continues. Just since the year 2000:

- **Data engineering and machine learning engineering** have emerged as distinct roles at most large technology companies. They apply classical engineering discipline to training pipelines, feature stores, and model deployment.
- **Cybersecurity engineering** has evolved from a sub-specialty of IT into its own field, with its own principles of defense in depth, threat modeling, and incident response.
- **Robotics engineering** now spans mechanical, electrical, computer, and software disciplines — a fully interdisciplinary field with its own graduate programs.
- **Synthetic biology and tissue engineering** apply design methodology to living systems, blurring the engineering-biology boundary in ways that raise genuinely new ethical questions.
- **Quantum engineering** is beginning to turn quantum physics into usable devices — sensors first, then possibly computers.

None of these existed as distinct disciplines when your professors started college. Some of them will look obvious and established by the time *you* graduate. And new disciplines that don't yet have names will emerge during your career.

The Through-Line

Step back from the timeline for a moment. Every era inherited tools and knowledge from the one before it.

A civil engineer designing a bridge today stands on about 4,600 years of accumulated knowledge: Egyptian quarrying and surveying, Roman arch and concrete theory, Islamic hydraulics, Gothic load-path reasoning, Renaissance mathematics, 18th-

century strength-of-materials theory, 19th-century structural analysis, 20th-century finite element analysis, and 21st-century parametric design software. Every one of those layers is still in use somewhere in the design process.

That is what separates a profession from a collection of tricks. A profession is what happens when knowledge accumulates, gets taught, gets refined, and gets passed on. When you finish this degree, you will inherit all of it. And when you solve a problem no one has solved before, your solution — if you document it well — becomes part of what the next engineer inherits.

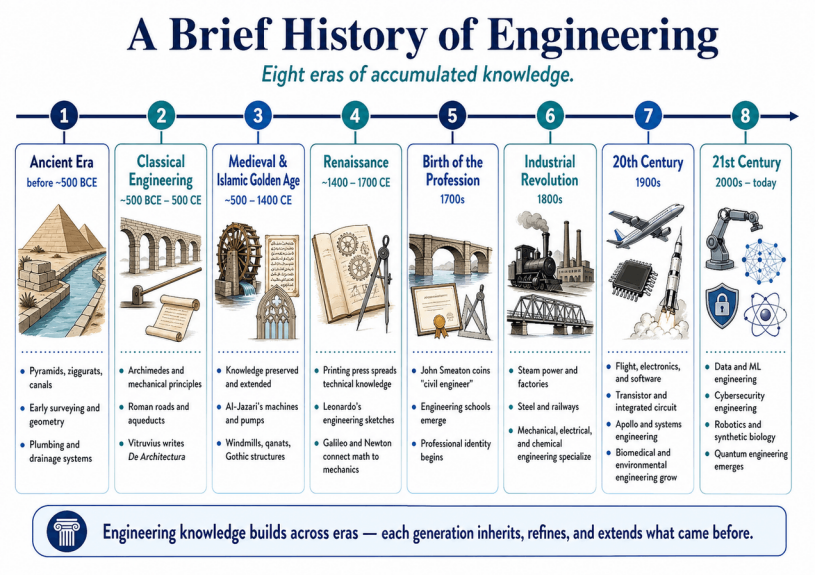


Figure 1.6.3: A brief history of engineering, organized as eight eras of accumulated knowledge from ancient engineering through twenty-first-century emerging fields. The timeline emphasizes that each generation of engineers inherits, refines, and extends the tools, methods, and systems developed before it. (Copyright; Jonathan Compton, AI-assisted original figure, CC BY-NC 4.0)

🔗 Connection

The **Historical Research** homework assignment asks you to investigate one specific engineer, engineering milestone, or project from this timeline in depth. As you read this section, notice which eras or ideas feel most interesting — that's a good place to start your research.

This page titled 1.6: What Engineers Actually Do is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

1.7: Major Disciplines at a Glance

1.7 The Major Disciplines at a Glance

You will not choose a discipline this semester — and you shouldn't feel pressure to. But it helps to know what the landscape looks like. This is a preview; Chapter 2 goes deeper into each.

Overview of major engineering disciplines, their core focus, and common products or systems.

Discipline	Core Focus	Typical Products/Systems
Civil	Infrastructure and the built environment	Roads, bridges, buildings, water systems
Mechanical	Motion, force, energy, heat	Engines, HVAC, robots, vehicles
Electrical	Electricity, electronics, power systems	Power grids, circuits, motors
Computer	Hardware-software integration, computing systems	Processors, embedded systems, architecture
Software	Design of software systems	Applications, operating systems, web platforms
Chemical	Materials transformation, processes	Pharmaceuticals, plastics, fuels, food
Aerospace	Flight and space systems	Aircraft, spacecraft, propulsion
Biomedical	Engineering applied to biology and medicine	Prosthetics, medical devices, imaging
Industrial	Systems, processes, optimization	Manufacturing lines, logistics, human factors
Environmental	Engineering applied to natural systems	Pollution control, sustainability, remediation

Every one of these disciplines shares the same core mindset you are learning in this course. The specialization comes later; the habits of thought come first.

🔗 Connection

The **Discipline Research** homework assignment asks you to pick one discipline from this table and investigate it in depth — what courses you'd take, what the job market looks like, what specific problems engineers in that field work on. You do not need to commit to that discipline. You just need to explore one seriously.

This page titled 1.7: Major Disciplines at a Glance is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

1.8: Ethics Is Not a Separate Chapter

1.8 Ethics Is Not a Separate Chapter

We will spend all of Chapter 14 on engineering ethics. But ethics is not something you bolt on at the end — it is woven into every technical decision you make.

Consider: an engineer who designs a bridge is not just solving a structural problem. They are implicitly deciding how much margin of safety is enough. They are choosing which failure modes to guard against and which to accept. They are making assumptions about maintenance, inspection, and future use. Every one of those decisions is partly ethical, even when the calculation itself is technical.

This is why engineers hold what is called a position of **public trust**. Most people will never check your calculations. They assume the bridge is safe because someone qualified did the work. When engineers cut corners — whether to save cost, hit a schedule, or avoid an uncomfortable conversation — the people who get hurt are almost always people who had no way of knowing the risk.

Ethics begins now, in this course, with small habits:

- Doing your own work.
- Stating your assumptions clearly.
- Checking your units.
- Admitting when you do not understand something.
- Revising work when you find an error, instead of hiding it.

Those habits are not "academic integrity." They are the beginnings of **professional identity**.

> **Ethics Check**
>
> Think of a time you used a product that failed in a way that could have hurt you — a rickety ladder, a chair that collapsed, a phone charger that sparked. Someone designed that product. Someone decided it was "good enough" to ship. What would you want that engineer to have considered, knowing you would eventually be the user?
>
> Hold onto your answer. We will return to it in Chapter 14.

This page titled 1.8: Ethics Is Not a Separate Chapter is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

1.9: What This Course Will Build

1.9 What This Course Will Build

ENGR 110 is not a survey of every engineering topic — that would be impossible in one semester, and most of it would be forgotten by next year anyway. Instead, this course is designed to build the **foundation habits** that carry across every discipline you might eventually choose.

By the end of the semester, you will have practiced:

- **Structured problem solving.** A repeatable five-step process (Define → Model → Analyze → Interpret → Communicate) that applies to nearly every engineering problem you will ever see.
- **Dimensional discipline.** Treating units as first-class objects and catching errors before they propagate.
- **Computational competence.** Using spreadsheets and a preview of MATLAB to model systems and explore "what if" questions.
- **Design reasoning.** Recognizing trade-offs, constraints, and the iterative nature of real design.
- **Technical communication.** Writing memos, documenting calculations, and explaining decisions clearly.
- **Professional ethics.** Recognizing that engineering authority carries responsibility.

These habits will show up in every engineering course you take after this one. Master them now, and the upper-division work will feel like applying tools you already trust. Skip them, and you will find yourself rebuilding them under pressure in courses that assumed you already had them.

> **Key Takeaway**
>
> Engineering is a disciplined way of solving real-world problems under constraint. It borrows its rules from science, uses design to navigate trade-offs, and builds systems that people depend on — often without realizing it. The technical skills you will learn in college matter, but the **mindset** — define carefully, model honestly, check your units, iterate, and consider who is affected — is what separates engineers from everyone else.

This page titled 1.9: What This Course Will Build is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

1.10: Summary

Summary

- Engineering is the disciplined application of math, science, and design to create systems that serve human needs under real-world constraints.
- Science asks *why*; engineering asks *how to do it safely and reliably*; technology is the resulting artifact.
- Engineers spend their time defining problems, modeling, analyzing, designing, testing, documenting, and communicating — not just "doing math."
- Engineering knowledge has accumulated across eight identifiable eras, from ancient applied intuition to the modern discipline explosion. Every tool we use today descends from something earlier.
- The engineering mindset has five habits: define before solving, model then analyze, check units, expect to iterate, and think about who is affected.
- Ethics is not separate from engineering — it is embedded in every technical decision, and it begins with the habits you build in this course.

This page titled 1.10: Summary is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

1.11: Quick Check

Quick Check

1. In your own words, explain the difference between engineering and science. Give one example of each.
2. A civil engineer and an electrical engineer both solve problems. What do their approaches have in common? Where might they differ?
3. The definition of engineering in Section 1.2 includes the phrase "real-world constraints." Why is that phrase essential? What happens to a design that ignores it?
4. Pick one engineered system you used today that you had not previously thought of as engineered. Describe the constraints its designer probably had to balance.
5. Why do engineers hold a "position of public trust"? Give one example, real or hypothetical, where that trust could be violated.
6. Choose any two eras from §1.6. Describe one specific way that engineering knowledge from the earlier era made the later era possible.
7. Of the five engineering mindset habits in Section 1.5, which do you think will be hardest for you personally? Why?
8. *[Stretch]* Suppose you are an engineer who discovers a small error in a calculation after the design has been approved. Fixing it will delay the project by two weeks. Not fixing it *probably* won't matter. What factors should go into your decision?

Answers - click to expand

1. Engineering and science are related, but they ask different kinds of questions. Science tries to explain how the natural world works. Engineering uses scientific knowledge, mathematics, materials, tools, and judgment to design solutions that work under real conditions.

   Example: A scientist might study how heat moves through a material. An engineer might use that knowledge to design insulation for a building, a heat sink for a computer, or a thermal protection system for a spacecraft.

2. A civil engineer and an electrical engineer both define problems, use models, make calculations, evaluate constraints, consider safety, and communicate results. Their work is similar because both are applying structured problem-solving to real-world systems.

   They differ in the systems and tools they use. A civil engineer may focus on structures, transportation,

water systems, or foundations, using loads, materials, soil properties, and codes. An electrical engineer may focus on circuits, power, signals, electronics, or communication systems, using voltage, current, resistance, frequency, and control logic.

3. The phrase "real-world constraints" is essential because engineering is not just about finding a theoretically correct answer. A design must work within limits such as cost, time, safety, materials, manufacturing methods, environmental impact, laws, codes, maintainability, and user needs.

   A design that ignores constraints may be impossible to build, too expensive, unsafe, unreliable, illegal, or unusable. In engineering, a solution that works only on paper is not enough.

4. Answers will vary. One possible example is a shower system. It may seem ordinary, but it depends on engineered water treatment, pumps, pipes, valves, drainage, pressure regulation, temperature control, and a water heater.

   The designers likely had to balance constraints such as water pressure, safety, temperature limits, corrosion resistance, cost, energy use, building codes, ease of repair, and reliability over many years of use.

5. Engineers hold a position of public trust because people rely on engineering work without being able to personally verify it. Most people cannot inspect the structural calculations for a bridge, the wiring inside a building, the software in a medical device, or the safety margins in an aircraft.

   Example: An engineer violates public trust if they approve a design they know has not been properly checked, hide a safety concern to avoid delaying a project, or sign off on calculations they did not review. Even if no failure occurs immediately, the public has been placed at risk.

6. Answers will vary. One possible connection is between the Classical Engineering era and the Renaissance. Greek and Roman engineers preserved and wrote down ideas about geometry, mechanics, materials, roads, aqueducts, and structures. Renaissance engineers later built on that written knowledge and connected it more formally to mathematics and scientific reasoning.

   Another possible connection is between the Industrial Revolution and the 20th century. The Industrial Revolution developed steam power, steel production, railways, electricity, and specialized engineering disciplines. Those developments made later technologies such as aircraft, automobiles, electronics, power grids, and large-scale manufacturing possible.

7. Answers will vary. A strong answer should identify one mindset habit, explain why it may be difficult, and connect it to a specific behavior the student can improve.

   Example: A student might say that checking assumptions will be hardest because they often want to start calculating immediately. To improve, they could make a habit of writing down knowns, unknowns, units, constraints, and assumptions before solving a problem.

8. The engineer should not dismiss the error simply because it *probably* will not matter. The decision should consider the size and type of error, whether it affects safety, code compliance, performance, cost, reliability, public welfare, environmental impact, or the assumptions used elsewhere in the design.

The engineer should also consider who needs to be informed, whether the approved design is already being built, whether the error affects other calculations, and whether a licensed engineer or supervisor must review the issue. A two-week delay may be inconvenient, but inconvenience does not outweigh safety, honesty, or professional responsibility.

A responsible course of action would be to document the error, evaluate its impact, communicate it to the appropriate supervisor or design authority, and recommend correction if the error affects safety, performance, compliance, or public trust.

This page titled 1.11: Quick Check is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

2: ENGINEERING DISCIPLINES AND CAREERS

Engineering Disciplines and Careers

Engineering disciplines are really *problem domains*. Pick yours based on the problems you want to solve — not on a single metric like salary — and remember you don't have to pick today.

Learning Objectives

By the end of this chapter, you will be able to:

- Describe the major engineering disciplines in terms of the **problems they solve** rather than just the titles they hold.
- Recognize that most modern engineering problems cross discipline boundaries.
- Interpret engineering salary and employment data critically, rather than at face value.
- Identify the functional roles engineers play in industry (design, test, R&D, management, etc.).
- Approach discipline selection as an ongoing exploration, not a one-time decision.

2.1: The Question Most Students Ask
2.2: The Major Engineering Disciplines
2.3: Most Real Problems Cross Discipline Boundaries
2.4: Engineering by the Numbers
2.5: Functional Roles
2.6: Choosing a Discipline
2.7: One Obligation That Every Discipline Shares
2.8: Summary
2.9: Quick Check

This page titled 2: Engineering Disciplines and Careers is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

2.1: The Question Most Students Ask

2.1 The Question Most Students Ask (and a Better One)

A very common first-semester question is:

> *"What kind of engineer should I be?"*

It is an honest question. It is also the wrong one to start with.

A better first question is:

> *"What kind of problems do I want to spend my career solving?"*

Engineering disciplines are really **problem domains**. Each discipline developed historically around a class of problems that was important enough — and hard enough — that specialists emerged to handle it. Civil engineering grew around infrastructure. Electrical engineering grew around electrical power and signals. Chemical engineering grew around industrial-scale material transformation. And so on.

Once you frame the question in terms of problems rather than titles, the discipline landscape becomes much easier to navigate. You don't have to figure out which tribe you belong to; you just have to figure out what kinds of problems grab your attention.

Some honest reassurance before we start:

- You do not need to pick a discipline this semester.
- Your first and second year coursework (calculus, physics, programming, basic design) is shared across nearly every engineering major.
- Plenty of successful engineers change disciplines during their degree — or work in a different discipline than the one their diploma names.

What matters now is *awareness* of what exists. Specialization can come later.

This page titled 2.1: The Question Most Students Ask is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

2.2: The Major Engineering Disciplines

2.2 The Major Engineering Disciplines

There are more than a dozen recognized engineering disciplines, and they overlap in meaningful ways. Below are short profiles of the ones you are most likely to encounter in an undergraduate engineering program. Think of this as a map, not a menu.

Each profile answers four questions: *What problems does this discipline solve? What tools and principles do engineers in this field use? Where do they work? What makes it distinctive?*

Mechanical Engineering (ME)

Problems solved: Anything involving motion, force, energy, or heat. Mechanical engineers design the moving parts — and the not-so-moving parts — of the physical world.

Core tools: Newton's laws, thermodynamics, strength of materials, fluid mechanics, CAD software, finite element analysis.

Where they work: Automotive, aerospace, HVAC, manufacturing, robotics, consumer products, energy.

What's distinctive: Mechanical is one of the broadest disciplines. An ME background lets you work on engines, roller coasters, medical devices, or wind turbines — sometimes in the same career. This breadth is why it is consistently one of the largest engineering disciplines by enrollment.

Professional society: American Society of Mechanical Engineers (ASME, founded 1880).

Electrical Engineering (EE)

Problems solved: Anything involving electricity — from power generation and distribution at the grid scale, down to signals inside microchips.

Core tools: Circuit analysis, electromagnetics, signal processing, control theory, Ohm's law, Kirchhoff's laws.

Where they work: Utilities, semiconductor companies, aerospace, consumer electronics, telecommunications, renewable energy.

What's distinctive: EEs work with quantities you cannot see directly — voltage, current, electromagnetic fields — but can measure, model, and control precisely. This requires comfort with mathematical abstraction. Circuits in particular reward careful, structured analysis, which is why Chapter 11 spends real time on them.

Professional society: Institute of Electrical and Electronics Engineers (IEEE, founded 1963 from predecessor organizations dating to 1884).

Civil Engineering (CE)

Problems solved: Infrastructure — the built environment that societies depend on. Buildings, bridges, roads, water and wastewater systems, dams, airports, tunnels.

Core tools: Structural analysis, materials (concrete, steel, soil), hydraulics, surveying, safety factors.

Where they work: Consulting firms, government agencies (city/state/federal), construction companies, utilities.

What's distinctive: Public safety is front and center. A failed consumer product is a recall; a failed bridge is a tragedy. Civil engineers typically pursue a **Professional Engineer (PE)** license because many civil engineering deliverables legally require a PE's stamp. This is less common in other disciplines.

Professional society: American Society of Civil Engineers (ASCE, founded 1852 — the oldest engineering society in the United States).

Computer Engineering (CpE)

Problems solved: The integration of hardware and software. Computer engineers design the processors, embedded systems, and computing architectures that software runs on.

Core tools: Digital logic, computer architecture, programming, embedded systems, hardware description languages (VHDL, Verilog).

Where they work: Semiconductor manufacturers, consumer electronics, automotive (increasingly), aerospace, defense, any industry building custom computing hardware.

What's distinctive: CpE sits at the boundary of electrical engineering and computer science. If you want to know how the chip inside your phone actually works — not just what the operating system does on top of it — this is the discipline.

Professional society: IEEE Computer Society.

Software Engineering / Computer Science (SWE/CS)

Problems solved: The design, construction, and maintenance of software systems — from the apps on your phone to the distributed systems running entire industries.

Core tools: Programming languages, algorithms, data structures, software design principles, testing frameworks, version control.

Where they work: Essentially every industry now. Technology companies, finance, healthcare, government, media, research.

What's distinctive: Software is the youngest engineering discipline and the one whose job titles and tools evolve fastest. "Computer Science" is usually the academic degree; "software engineering" is usually the job. The distinction matters less than it used to.

Note for this course: If you are on the computer/software track at AVC, you will take ENGR 110 alongside CS 110. The engineering habits you learn here — define before you solve, check your units, iterate — all apply to software, even though the "units" become things like response times and memory.

Chemical Engineering (ChemE)

Problems solved: Transforming raw materials into useful products at scale. Chemical engineers design the processes that turn crude oil into gasoline, bauxite into aluminum, corn into ethanol, and molecules into pharmaceuticals.

Core tools: Thermodynamics, reaction kinetics, heat and mass transfer, process control, safety analysis.

Where they work: Pharmaceuticals, petrochemicals, food and beverage, materials manufacturing, semiconductors, energy.

What's distinctive: Scale is the defining challenge. A reaction that works in a lab flask may behave completely differently in a 10,000-gallon reactor. Chemical engineers specialize in the math and process design that let chemistry operate continuously, safely, and economically at industrial scale.

Professional society: American Institute of Chemical Engineers (AIChE, founded 1908).

Aerospace Engineering (AE)

Problems solved: Flight — both inside and outside the atmosphere. Aircraft, spacecraft, propulsion systems, satellites, missiles.

Core tools: Aerodynamics, propulsion, structural analysis under extreme loads, flight control systems, orbital mechanics.

Where they work: Commercial aviation (Boeing, Airbus), defense (Lockheed Martin, Northrop Grumman), space (NASA, SpaceX, Blue Origin), supplier companies.

What's distinctive: Aerospace operates under some of the tightest weight, safety, and performance constraints in engineering. A part that is "good enough" in automotive might be completely unacceptable in aerospace. AE is often closely related to mechanical engineering — many schools offer a combined ME/AE degree.

Professional society: American Institute of Aeronautics and Astronautics (AIAA, founded 1963).

Biomedical Engineering (BME)

Problems solved: Applying engineering methods to medicine and biology. Prosthetics, medical imaging, implantable devices, drug delivery systems, diagnostic equipment.

Core tools: Physiology, materials science, electronics, signal processing, mechanics, regulatory knowledge.

Where they work: Medical device companies, hospitals, pharmaceutical companies, research institutions.

What's distinctive: BME is genuinely interdisciplinary — you need to be comfortable switching between engineering constraints and biological ones. It is also one of the younger disciplines, and the one most likely to evolve dramatically over your career as fields like tissue engineering and neural engineering mature.

Professional society: Biomedical Engineering Society (BMES, founded 1968).

Industrial Engineering (IE)

Problems solved: Optimizing systems and processes. Industrial engineers are usually not designing a physical object — they are designing the *flow* around it. Manufacturing lines, supply chains, warehouses, logistics networks, even hospitals.

Core tools: Statistics, operations research, optimization, simulation, human factors, lean manufacturing.

Where they work: Manufacturing, logistics (Amazon, UPS), healthcare systems, consulting, tech companies (supply chain roles).

What's distinctive: IE answers "how do we do this more efficiently, safely, or reliably?" rather than "how do we build this?" It is engineering applied to complexity and coordination, which is why IE is increasingly valuable in software companies (site reliability engineering, operations research) and healthcare.

Professional society: Institute of Industrial and Systems Engineers (IISE, founded 1948).

Environmental Engineering (EnvE)

Problems solved: The interaction between engineered systems and natural ones. Water and air quality, waste management, pollution control, remediation of contaminated sites, sustainable infrastructure.

Core tools: Chemistry, biology, fluid mechanics, regulatory frameworks (EPA, state agencies), modeling of natural systems.

Where they work: Consulting firms, government agencies, industrial companies with environmental compliance needs, non-profits.

What's distinctive: Environmental engineering evolved from civil and chemical engineering and still overlaps with both. It is distinctively tied to regulation and public policy — many environmental engineering decisions are driven by what the law requires.

Professional society: American Academy of Environmental Engineers and Scientists (AAEES, founded 1955).

Materials Engineering (MatE) / Materials Science

Problems solved: Understanding and designing the substances other engineers build with. Metals, polymers, ceramics, composites, semiconductors, biomaterials.

Core tools: Solid-state chemistry, crystallography, mechanical testing, microscopy, processing methods.

Where they work: Semiconductor companies, aerospace (lightweight alloys, thermal protection), biomedical (implants), automotive (batteries, composites), research labs.

What's distinctive: Almost every other discipline *uses* materials. Materials engineers *design* them. The field sits right at the boundary between engineering and chemistry/physics, and it often requires a graduate degree to do the most interesting work.

Other disciplines you will encounter

- **Nuclear engineering** — design and safe operation of nuclear reactors and related systems.
- **Petroleum engineering** — extraction and processing of oil and natural gas.
- **Agricultural / bioresource engineering** — farming equipment, food systems, agricultural water management.
- **Manufacturing engineering** — closely related to mechanical and industrial, focused on how things are made at scale.
- **Mining / geological engineering** — extraction of mineral resources, tunneling, underground work.

Each of these is smaller than the disciplines above but offers real careers. You will investigate many of these in the Discipline Research homework.

🔗 Connection

The **Discipline Research** homework asks you to identify five California universities offering each of these programs and two current research areas for each. This is deliberately a lot of disciplines — the point is not to memorize them, but to get exposure to the breadth of choices available before you commit to one.

Later in the semester, the **PowerPoint Presentation Project** asks you to go deep on one discipline you are assigned: the professional societies, the accreditation, the companies, the products, the research. The presentation project is where "which discipline do I care about" starts to get a real answer.

This page titled 2.2: The Major Engineering Disciplines is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

2.3: Most Real Problems Cross Discipline Boundaries

2.3 Most Real Problems Cross Discipline Boundaries

A trap first-year students sometimes fall into is believing that a chemical engineer only works on chemical problems, a mechanical engineer only works on mechanical ones, and so on. In practice, almost every modern engineering problem is **multidisciplinary**. Look at a few examples:

- **An electric vehicle** needs mechanical engineering (chassis, drivetrain, thermal management), electrical engineering (motor, power electronics, wiring), computer engineering (the ECU and embedded software), chemical engineering (battery chemistry), materials engineering (battery materials, lightweight structure), industrial engineering (assembly line), and environmental engineering (recyclability, supply chain).
- **A modern hospital** requires civil engineering (building), mechanical engineering (HVAC, gas systems), electrical engineering (power, backup systems), biomedical engineering (imaging, devices), software engineering (electronic health records), and industrial engineering (patient flow, scheduling).
- **A smartphone** requires computer engineering (SoC design), electrical engineering (RF, power management), mechanical engineering (enclosure, thermal design), materials engineering (display, battery, housing), software engineering (OS and apps), and industrial engineering (manufacturing lines).

The takeaway: **your specialty is where you go deep, but you will almost always work with engineers from other disciplines.** The habits this course teaches — structured problem solving, clear communication, dimensional discipline, ethical reasoning — are the habits that let you collaborate across disciplines without losing rigor.

Real-World Example — The Apollo Program

The Apollo program in the 1960s is often cited as one of the most successful multidisciplinary engineering efforts in history. Putting astronauts on the Moon required:

- **Aerospace engineers** to design the rockets and spacecraft.
- **Mechanical and materials engineers** for heat shields, structures, and mechanisms.
- **Electrical engineers** for power and communications.
- **Computer engineers** for the Apollo Guidance Computer — a groundbreaking embedded system.
- **Chemical engineers** for fuels and life-support chemistry.
- **Industrial engineers** for the massive production and logistics effort.
- **Biomedical engineers** for spacesuits and astronaut health monitoring.

Over 400,000 people contributed. No one person held the full design in their head. What held it together was disciplined *engineering process* — requirements, specifications, reviews, documentation — not any one person's genius. That discipline is what you are starting to build this semester.

This page titled 2.3: Most Real Problems Cross Discipline Boundaries is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

2.4: Engineering by the Numbers

2.4 Engineering by the Numbers — Reading the Data Critically

Engineering is, on average, a well-compensated profession. The U.S. Bureau of Labor Statistics reported that in May 2024, the median annual wage across all engineering occupations was about **$97,310**, compared to **$49,500** for all U.S. occupations combined. Employment in engineering is projected to grow faster than the overall labor market through 2034, with roughly 186,000 openings per year, on average.

Broken down by discipline, the picture varies quite a bit:

Median annual wages and projected growth outlook by engineering discipline.

Discipline	Median Annual Wage (May 2024)	2024–2034 Growth Outlook
Petroleum	$141,280	Slower than average (1%)
Aerospace	$134,830	Faster than average (6%)
Nuclear	$127,520	Decline (–1%)
Chemical	$121,860	About as fast as average (3%)
Electrical	$111,910	Much faster than average (7%)
Bioengineering / Biomedical	$106,950	Faster than average (5%)
Environmental	$104,170	About as fast as average (4%)
Mechanical	$102,320	Much faster than average (9%)
Industrial	$101,140	Much faster than average (11%)
Civil	$99,590	Faster than average (5%)
All engineers (combined)	**$97,310**	**Faster than average**
All U.S. occupations (reference)	*$49,500*	—

Data source: U.S. Bureau of Labor Statistics, Occupational Employment and Wage Statistics, May 2024, and Employment Projections 2024–2034.

The figures below show two different ways to read the same career data. The first figure compares **median annual wages** by discipline. The second figure compares **field size** by showing the number of engineers employed in each discipline. Both views matter: a discipline can have a high median wage but a relatively small job market.

Median Wage Comparison

This graph compares median annual wages across engineering disciplines and includes the all-U.S.-occupations median as a reference point.

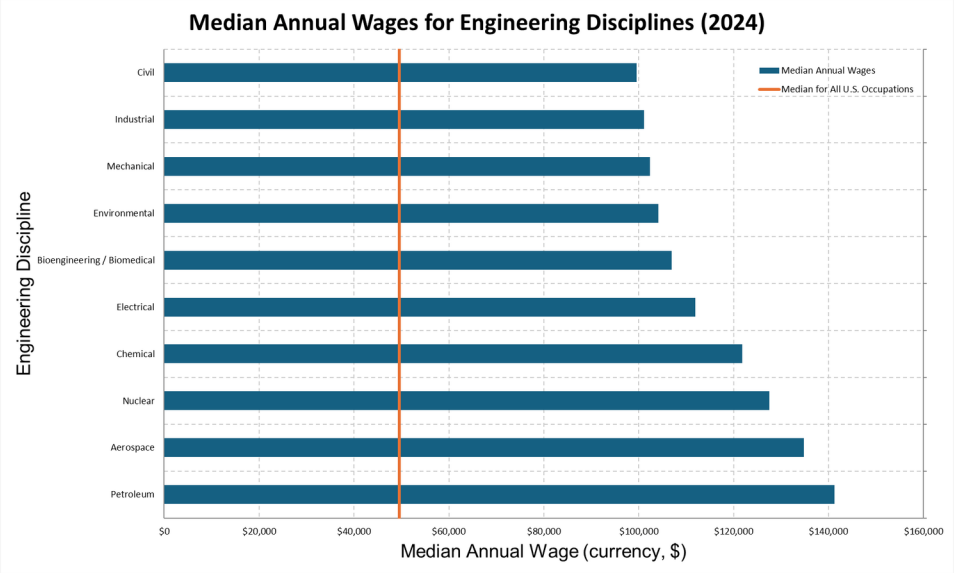


Figure 2.4.1: Median annual wage by engineering discipline, compared with the median for all engineers combined and all U.S. occupations. The graph shows that engineering disciplines generally have higher median wages than the all-occupation reference, but wage alone does not describe job availability or career stability. (Data source: U.S. Bureau of Labor Statistics, May 2024; Copyright; Jonathan Compton, CC BY-NC 4.0).

Discipline Size Comparison

This graph shows how many engineers work in each discipline. Larger fields usually provide more openings, more geographic flexibility, and more resilience to downturns in a single industry sector.

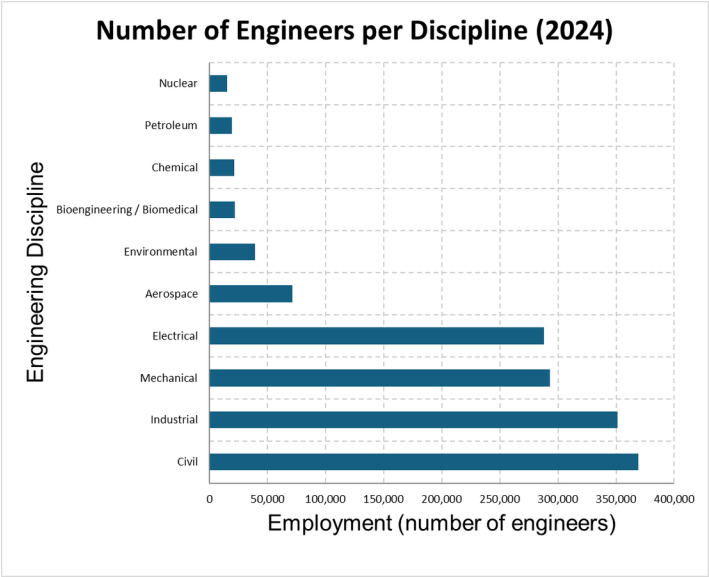


Figure 2.4.2: Approximate number of engineers employed by discipline in 2024. Field size is an important complement to salary data because a high-paying discipline may employ relatively few engineers nationally, while larger disciplines may offer more total opportunities. (Data source: U.S. Bureau of Labor Statistics, Employment Projections 2024–2034; Copyright; Jonathan Compton, CC BY-NC 4.0).

Why the highest-paying discipline is not automatically the best choice

A student looking at that table might reasonably think: *pick petroleum, it pays the most.* That instinct deserves some engineering reasoning.

A few things that first table doesn't tell you:

Field size matters. Petroleum engineering employs roughly 15,000 engineers nationally. Civil engineering employs over 370,000. If you study petroleum engineering and the industry contracts — as it has during several historical downturns — the job market gets crowded quickly. Large disciplines are more resilient to industry-specific shocks.

Growth matters more than current wage. Industrial engineering is projected to grow 11% over the next decade. Petroleum, 1%. That growth differential compounds over a 40-year career. A discipline that's growing fast tends to have more opportunity, more mobility, and more upward salary pressure.

Regional variation is huge. A civil engineer in San Francisco earns substantially more than a civil engineer in rural Oklahoma — but the San Francisco engineer also pays dramatically more for housing. National medians hide these swings.

Industry sector matters as much as discipline. A mechanical engineer at a semiconductor company and a mechanical engineer at a small-town HVAC contractor have the same degree but very different compensation trajectories. Industry choice affects pay as much as discipline choice.

Salary alone should not drive the decision. Over a long career, you will be much more productive — and much happier — working on problems you find interesting. That matters more than a 10% starting-salary delta, which tends to shrink as you gain experience.

⚠ Watch Out — Reading Salary Data in Isolation

Three common mistakes when interpreting engineering salary data:

1. **Confusing median with entry-level.** The median is the middle of the *whole career distribution*. Entry-level salaries are typically 30–50% lower. A discipline with a $120,000 median probably starts new graduates around $70,000–$80,000.
2. **Ignoring cost of living.** A $90,000 salary in Lancaster, CA buys a different lifestyle than a $90,000 salary in Manhattan. National numbers do not reflect regional reality.
3. **Assuming the data will stay the same.** These tables reflect 2024. By the time you graduate in four years, the rankings will have shifted — probably not dramatically, but enough to matter.

Engineering reasoning applies to career decisions, not just technical ones: *what is the data, what does it actually say, and what is it missing?*

This page titled 2.4: Engineering by the Numbers is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

2.5: Functional Roles

2.5 What Engineers Actually Do (Functional Roles)

Your discipline (mechanical, civil, electrical, etc.) tells you *what problems you solve*. Your role tells you *what you do about them*. The same mechanical engineer might be in very different roles at different points in their career:

- **Design engineer** — generates the actual designs. Heavy use of CAD, analysis tools, and design reviews.
- **Test engineer** — builds and runs experiments to verify that designs actually work as predicted. Heavy use of instrumentation, data analysis, and statistics.
- **Research and development (R&D)** — pushes the boundaries of what is possible. Often requires or benefits from a graduate degree.
- **Manufacturing / production engineer** — makes sure designs can actually be built, at scale, reliably, and economically.
- **Quality / reliability engineer** — monitors whether the designs continue to perform as expected, and catches failures before they become disasters.
- **Project / program manager** — coordinates teams, schedules, budgets, and deliverables across engineers and other stakeholders.
- **Systems engineer** — owns the big picture — how subsystems fit together and whether overall system requirements are met.
- **Applications / field engineer** — works with customers who are using the product, often in-person. Part engineering, part technical sales.
- **Regulatory / compliance engineer** — ensures designs meet legal and industry standards (especially important in biomedical, aerospace, nuclear, and environmental).

New engineers typically start in design, test, or manufacturing roles and branch out from there. A career that begins with six years of design work and then transitions into project management is a very common arc. So is a career that stays technical (moving into more senior design or R&D roles) rather than moving toward management.

There is no single "correct" path, and you do not have to know your path now. But it's worth knowing the roles exist, because they shape what a typical day looks like.

This page titled 2.5: Functional Roles is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

2.6: Choosing a Discipline

2.6 Choosing a Discipline (Without Committing Too Early)

If you are reading this chapter in your first semester, the best thing you can do is:

1. **Take foundational coursework seriously.** Calculus, physics, and an introductory programming course are required in essentially every engineering major. Strong performance in these keeps every door open. Weak performance closes doors you haven't even thought about yet.
2. **Notice what interests you.** As you move through this chapter, the Discipline Research HW, and the Presentation Project, notice which topics you find yourself curious about after the assignment is over. That is real data about your interests.
3. **Talk to actual engineers.** The Career Preparation Project explicitly asks you to do this. It is one of the highest-value exercises in the course, precisely because engineering is so different from the outside than from the inside.
4. **Don't commit prematurely.** Changing from one engineering major to another in year two is easy. Changing from engineering to a non-engineering major in year two is also easy. Changing from a non-engineering path *to* engineering in year two is harder, because you will be behind on calculus and physics.
5. **Accept that you may not know.** Many successful engineers pick their discipline somewhat arbitrarily at first, then find over time what they actually enjoy. That is fine. The first choice is a starting point, not a lifetime sentence.

The students who struggle most are usually not the ones who pick the "wrong" discipline — they are the ones who neglect the foundations while agonizing over a decision that does not need to be made yet.

> 🔗 **Connection**
>
> The **Career Preparation Project** will ask you to build a resume, research potential internships, and interview a working engineer. That interview is often the moment when a student's abstract picture of "what engineers do" turns into something concrete and personal. It's worth treating the interview as exploration, not just as an assignment — ask the questions that would actually help you figure out whether this is a path you want.

This page titled 2.6: Choosing a Discipline is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

2.7: One Obligation That Every Discipline Shares

2.7 One Obligation That Every Discipline Shares

Disciplines differ. Tools differ. Work environments differ. One thing does not differ: **every engineering discipline shares the same primary obligation to the safety, health, and welfare of the public.**

You will see this principle written into every engineering code of ethics — ASME's, IEEE's, ASCE's, AIChE's, all of them. The specific hazards differ by discipline. A structural engineer worries about different failure modes than a software engineer. But the core obligation is the same: the public trusts engineers to do careful, honest work, because the public cannot do it themselves.

This is why the habits you are building in this course — checking your units, stating your assumptions, documenting your reasoning, revising your work when you find errors — are not just academic. They are the habits that make public trust warranted.

Chapter 14 will come back to this in detail. For now, recognize that whatever discipline you eventually pick, the ethical core is the same.

Key Takeaway

Engineering disciplines are best understood as **problem domains**, not as job titles or tribes. The mechanical engineer, the software engineer, and the civil engineer share more than they differ — all of them define problems, model systems, analyze rigorously, iterate on designs, and carry responsibility to the public. Pick your discipline based on the *problems* you want to solve, not on a single metric like salary. And do not rush the decision — your first two years of coursework are largely shared across disciplines anyway.

This page titled 2.7: One Obligation That Every Discipline Shares is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

2.8: Summary

Summary

- Engineering disciplines are organized around classes of problems, not around job titles.
- Mechanical, electrical, civil, computer, software, chemical, aerospace, biomedical, industrial, environmental, and materials are among the major disciplines; others (nuclear, petroleum, agricultural, manufacturing) exist in more specialized contexts.
- Most real engineering problems are multidisciplinary. Learning to collaborate across disciplines matters as much as mastering your own.
- Median engineering wages in May 2024 ranged from about $99,000 (civil) to $141,000 (petroleum), with all disciplines well above the U.S. median of $49,500. Growth rates and field sizes vary substantially.
- Functional roles (design, test, R&D, manufacturing, project management, systems, field) describe *what you do* within a discipline. Most engineers move through several roles over a career.
- All disciplines share the same primary ethical obligation: the safety, health, and welfare of the public.

This page titled 2.8: Summary is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

2.9: Quick Check

Quick Check

1. Pick any engineered system you interact with daily. Name at least three engineering disciplines that contributed to its design.
2. A student says, "I want to be an engineer but I don't know what kind." What are two better questions they could start asking themselves? Why?
3. Looking at the salary table in §2.4, explain why a student might rationally choose mechanical engineering over petroleum engineering despite the ~$40,000/year difference in median wage.
4. Name two engineering disciplines that evolved historically out of older ones. Why did the split make sense?
5. Describe one functional role from §2.5 and explain why that role might appeal to someone who likes hands-on work. Then pick a role that might appeal to someone who prefers big-picture coordination.
6. All engineering disciplines share an obligation to public welfare. Give one example, real or hypothetical, of how that obligation might apply differently in civil engineering than in software engineering.
7. *[Stretch]* Suppose your current top two discipline choices are, say, mechanical and computer engineering. Design a first-year and second-year course plan that would keep *both* options open. What foundational courses would you prioritize?

Answers — click to expand

1. Answers will vary. One possible example is a smartphone. Electrical engineers contributed to the circuits, antennas, sensors, battery systems, and power management. Computer engineers and software engineers contributed to the processor architecture, operating system, apps, security, and communication protocols. Materials engineers contributed to the glass, metals, polymers, semiconductors, and battery materials. Mechanical engineers contributed to the enclosure, thermal management, durability, and manufacturability.
2. Two better questions are:
    1. What kinds of problems do I want to work on? This shifts the decision from job title to purpose: transportation, energy, robotics, medicine, electronics, infrastructure, software, aerospace, manufacturing, or environmental systems.
    2. What kinds of tools and work environments do I enjoy? Some students enjoy coding, some like physical systems, some like lab testing, some like design modeling, and some like field work or project coordination.

These questions are better because engineering disciplines are not just labels. They represent different problem domains, tools, materials, constraints, and work settings.

3. A student might rationally choose mechanical engineering over petroleum engineering because salary is only one factor. Mechanical engineering is a broad discipline connected to many industries, including aerospace, automotive, energy, robotics, HVAC, manufacturing, biomedical devices, and consumer products. That breadth can create more flexibility if one industry slows down.

   Petroleum engineering may have a higher median wage, but it is a smaller and more industry-specific field. A smaller field can mean fewer total openings, more geographic concentration, and more exposure to changes in energy markets. Mechanical engineering may offer a better fit for the student's interests, more transfer options, more job-market flexibility, and a wider range of long-term career paths.

4. One example is aerospace engineering, which evolved partly from mechanical engineering as flight, propulsion, aerodynamics, and spacecraft systems became complex enough to require specialized study. The split made sense because aircraft and spacecraft involve specialized constraints such as lift, drag, stability, propulsion, lightweight structures, and extreme operating environments.

   Another example is computer engineering, which grew from electrical engineering and computer science as digital hardware, microprocessors, embedded systems, and hardware-software integration became central technologies. The split made sense because modern computing systems require engineers who understand both circuits and software.

5. Answers will vary depending on the roles listed in §2.5. A role that may appeal to someone who likes hands-on work is a test engineer or manufacturing engineer. These roles often involve prototypes, equipment, measurements, troubleshooting, production processes, and real physical systems.

   A role that may appeal to someone who prefers big-picture coordination is a systems engineer or project engineer. These roles focus on how parts of a system fit together, how requirements are managed, how teams coordinate, and how technical decisions affect the overall project.

6. In civil engineering, the obligation to public welfare might involve designing a bridge, building, road, or water system with adequate safety factors, code compliance, drainage, seismic resistance, and long-term durability. If a civil engineer ignores a structural or environmental risk, the public could be physically harmed.

   In software engineering, the obligation to public welfare might involve protecting user data, preventing security vulnerabilities, avoiding biased or unsafe automated decisions, and ensuring reliability in critical systems. If software controls medical devices, vehicles, financial systems, or public infrastructure, a software failure can also create real public harm, even though the work may not look physical at first.

7. Answers will vary. For a student trying to keep mechanical engineering and computer engineering open, the priority should be foundational courses that count toward both or support both.

   A reasonable first-year plan would prioritize Calculus I and II, calculus-based Physics I, an introductory engineering course, and an introductory programming course. If available, the student should also complete

general education requirements that transfer broadly.

A reasonable second-year plan would prioritize Calculus III, Differential Equations, calculus-based Physics II, circuit analysis, and a second programming or computer science course. To keep mechanical engineering open, the student should consider statics, materials science, and CAD/engineering graphics. To keep computer engineering open, the student should consider digital logic, discrete structures, linear algebra, and additional programming.

The key is to verify transfer requirements early using a counselor and articulation resources, because the best plan depends on the target university.

This page titled 2.9: Quick Check is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

3: ACADEMIC PATHWAYS AND TRANSFER STRATEGY

Academic Pathways and Transfer Strategy

Engineering coursework is sequential, not flexible. The students who succeed at transfer are the ones who map the path early, protect the math sequence, and ask for help before falling behind.

Learning Objectives

By the end of this chapter, you will be able to:

- Explain why engineering coursework is sequential rather than flexible.
- Describe the mathematics and physics backbone that all engineering disciplines share.
- Read and use the AVC engineering track tables to plan your own course sequence.
- Distinguish between the associate, bachelor's, master's, and doctoral degrees and what each enables.
- Apply the engineering problem-solving framework to your own academic plan.
- Identify the most common academic risk factors in engineering programs and how to mitigate them.

3.1: Engineering Is a Structured Academic Path
3.2: The Mathematics Backbone
3.3: Physics as Applied Mathematics
3.4: Typical Lower-Division Engineering Path (Community College)
3.5: Degree Levels and Transfer Destinations
3.6: Transfer Strategy
3.7: Academic Risk Factors
3.8: Treating Your Academic Plan Like an Engineering Problem
3.9: Summary
3.10: Quick Check

This page titled 3: Academic Pathways and Transfer Strategy is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

3.1: Engineering Is a Structured Academic Path

3.1 Engineering Is a Structured Academic Path

Here is something that surprises many first-semester students: you cannot take engineering courses in whatever order you want.

Engineering education is **cumulative by design**. Every course assumes you have genuinely mastered what came before it — not just passed it, but internalized it well enough to use it automatically while learning something new.

The chain of dependency runs something like this:

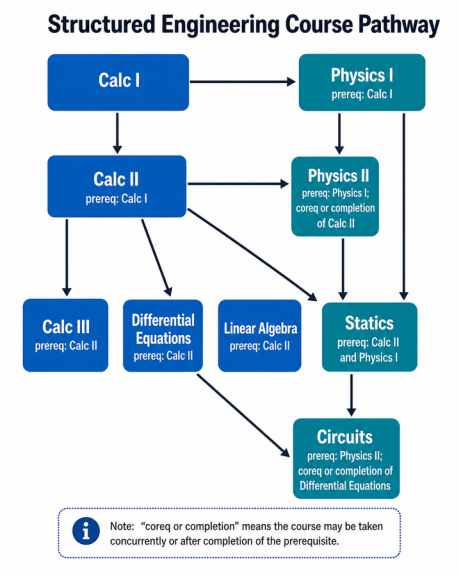


Figure 3.1.1: Structured engineering course pathway showing prerequisite relationships among foundational math, physics, and engineering courses. The sequence begins with Calculus I, continues to Calculus II, and then branches to Calculus III, Differential Equations, Linear Algebra, and Physics II. Physics I requires Calculus I; Physics II requires Physics I and co-requisite or completion of Calculus II; Statics requires Calculus II and Physics I; and Circuits requires Physics II and co-requisite or completion of Differential Equations. (Copyright; Jonathan Compton, AI-assisted original figure, CC BY-NC 4.0).

This is not bureaucratic gatekeeping. It is a genuine reflection of how the subject matter is structured. You cannot analyze a circuit without understanding derivatives and integrals. You cannot understand how a bridge carries load without understanding Newton's laws. You cannot write a useful simulation without knowing what differential equation you are actually solving.

The practical implication is significant: **if you fall behind in the sequence, delays compound.** A missed semester of calculus does not just delay calculus — it can cascade into delayed physics, delayed engineering courses, delayed transfer, and a longer overall timeline. On the flip side, students who build strong foundations early often find that upper-division coursework feels *manageable* rather than overwhelming, because the same tools just keep getting applied in new contexts.

Think of your academic plan the same way an engineer thinks about a project timeline: identify the dependencies, protect the

critical path, and deal with disruptions early before they propagate.

This page titled 3.1: Engineering Is a Structured Academic Path is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

3.2: The Mathematics Backbone

3.2 The Mathematics Backbone

Every engineering discipline rests on the same mathematical foundation. The math courses are not general education filler — they are the primary language you will use to do engineering.

At AVC, the math backbone consists of:

AVC mathematics courses that form the foundation for engineering coursework.

Course	AVC Course	Units	What it gives you
Calculus I	MATH 150	5	Derivatives, limits, basic integration — rates
Calculus II	MATH 160	4	Integration techniques, series — energy, wor
Calculus III	MATH 250	4	Multivariable calculus — fields, surfaces, 3D
Differential Equations	MATH 230	4	Dynamic systems — oscillations, circuits, hea
Linear Algebra	MATH 220	4	Matrix methods, systems of equations — con

What each course actually does for your engineering:

- **Derivatives** let you describe how fast something changes — velocity, acceleration, heat flow rate, current through a capacitor.
- **Integrals** let you accumulate quantities — total work done, total charge stored, total area under a load curve.
- **Differential equations** let you model systems that *evolve over time* — a vibrating structure, a charging battery, a control loop.
- **Linear algebra** underlies numerical methods, circuit analysis, structural analysis, and machine learning.

Students sometimes experience difficulty in calculus not because calculus is abstractly hard, but because **algebra and trigonometry foundations are shaky**. If you find yourself making consistent sign errors or getting lost in algebraic manipulation, that is the real problem — and it is fixable with targeted practice before or during Calculus I.

Mathematics in engineering is performance-based, not reading-based. You build fluency by *solving problems*, not by reading solutions. The students who do well are the ones who treat each homework set as a practice session, not a box to check.

⚠ Watch Out — "I'll just push through calculus"

"Pushing through" a math course with a C when your algebra is shaky is one of the most common and costly mistakes in engineering education. A C in Calculus I means you retained about half the material. You will meet that other half again in Calculus II, Physics I, Differential Equations, and every engineering course that follows. A semester spent strengthening algebra before or during Calc I is not a delay — it is an investment that pays back many times over.

This page titled 3.2: The Mathematics Backbone is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

3.3: Physics as Applied Mathematics

3.3 Physics as Applied Mathematics

Physics is where the math you learned in calculus becomes *physical* — connected to real things like forces, velocities, charges, and fields. For many students, physics is the moment when calculus stops feeling like abstract symbol-shuffling and starts feeling like it actually describes how the world works.

At AVC, you will take two semesters of physics:

AVC physics courses required for the engineering pathway.

Course	AVC Course	Units	Content
Physics I — Mechanics	PHYS 110	4	Forces, motion, ener
Physics II — Electricity & Magnetism	PHYS 120	4	Electric fields, magn

Physics I covers Newton's Laws, equilibrium, kinematics, work and energy, momentum — all the building blocks for statics, dynamics, and structural analysis. Physics II introduces the electrical concepts that will run through everything from Chapter 11 of this book to the full Circuit Analysis course (ENGR 230).

How to study physics productively:

Physics rewards the same five-step framework introduced in Chapter 1:

1. **Define** — what system are you analyzing? What is given? What is unknown?
2. **Model** — draw a free-body diagram (Physics I) or circuit schematic (Physics II). The picture matters.
3. **Analyze** — apply the governing equation (Newton's 2nd Law, Ohm's Law, Kirchhoff's Laws).
4. **Interpret** — does the answer make physical sense? Check magnitude and direction.
5. **Communicate** — show your work with units at every step.

Students who treat physics as formula memorization — picking equations off a list and plugging in numbers — usually struggle in upper-division engineering, where problems are less structured and require genuine understanding of *why* a particular model applies. Students who practice the five-step process consistently develop the engineering judgment that carries forward.

Physics also has a hard prerequisite structure: Physics I requires Calculus I to be in progress or completed; Physics II requires Calculus II to be at minimum concurrent.

This page titled 3.3: Physics as Applied Mathematics is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

3.4: Typical Lower-Division Engineering Path (Community College)

3.4 The AVC Engineering Tracks

Antelope Valley College has structured three distinct engineering tracks, each targeting a different cluster of 4-year disciplines. All three share the same general core — the calculus and physics sequence above — but diverge in their discipline-specific courses.

General Core (all tracks)

Every AVC engineering student completes the following regardless of track:

Table 3.4.1: General mathematics and physics core required for all AVC engineering tracks.

Course	Units
MATH 150, Calculus I	5
MATH 160, Calculus II	4
MATH 250, Calculus III — Multivariable	4
MATH 230, Ordinary Differential Equations	4
PHYS 110, Physics — Mechanics	4
PHYS 120, Physics — E&M	4
Total General Core	**25**

These 25 units form the shared backbone. A student who completes this core has done the first two years of math and physics that any accredited engineering program will require — regardless of where they transfer.

Track 1 — Mechanical and Aerospace Engineering

This track prepares students for transfer into ME, AE, or closely related programs.

Table 3.4.2: Track-specific courses for mechanical and aerospace engineering.

Course	Units
ENGR 110, Introduction to Engineering	3
ENGR 125, Programming and Problem-Solving in MATLAB	4
ENGR 130, Materials Science	4
ENGR 140, Engineering 3D Graphics	3
ENGR 210, Statics	4
CHEM 110	5
Track-Specific Total	**23**

Combined with general core: ~48 units of transferable engineering coursework.

Key connections

ENGR 210 (Statics) is the first explicitly engineering-mechanics course and builds directly on Physics I. ENGR 130 connects to why materials matter for design (Chapter 12 of this book). ENGR 140 is direct preparation for parametric CAD.

Track 2 — Electrical Engineering

This track prepares students for transfer into EE or electronics-focused programs.

Table 3.4.3: Track-specific courses for electrical engineering.

Course	Units
ENGR 110, Introduction to Engineering	3
ENGR 125, Programming and Problem-Solving in MATLAB	4
ENGR 185, Digital Logic	4
ENGR 230, Circuit Analysis	4
CS 120, 121, or 122 (pick language based on 4-year target)	3

Course	Units
PHYS 211, Optics and Modern Physics	5
Track-Specific Total	**23**

Combined with general core: ~48 units of transferable engineering coursework.

Key connections

ENGR 230 (Circuit Analysis) is the course this textbook's Chapter 11 previews. ENGR 185 (Digital Logic) introduces the hardware foundation that computer engineering builds on.

Track 3 — Computer Engineering

This track prepares students for transfer into CpE, CS, or software-focused programs.

Table 3.4.4: Track-specific courses for computer engineering.

Course	Units
ENGR 110 / CS 110 (paired)	3
ENGR 230, Circuit Analysis	4
MATH 220, Linear Algebra	4
CS 140, Computer Architecture	3
CS 150, Discrete Structures	4
CS 120/130, 121/131, or 122/132 (pick language sequence)	6
Track-Specific Total	**25**

Combined with general core: ~50 units of transferable engineering coursework.

Key connections

CS 150 (Discrete Structures) is the mathematical foundation of algorithms and digital systems. CS 140 (Computer Architecture) bridges hardware and software. Linear Algebra is essentially required for anything involving graphics, ML, or systems analysis.

Suggested Electives (any track)

Table 3.4.5: Suggested electives that strengthen preparation for any engineering track.

Course	Units
MATH 220, Linear Algebra	4
CS 122, Programming and Algorithms in Python	3
ENGR 140, Engineering 3D Graphics	3

These electives increase competitiveness at transfer and are relevant across disciplines.

> **🔗 Connection**
>
> The course numbers and units here are current as of this publication. Requirements change — always verify your track requirements with an AVC counselor, and use **ASSIST.org** to confirm how your AVC coursework will articulate to your target 4-year institution. ASSIST is the official California articulation database and the authoritative source for transfer planning.

This page titled 3.4: Typical Lower-Division Engineering Path (Community College) is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

3.5: Degree Levels and Transfer Destinations

3.5 Where You Can Go: Degree Levels and Transfer Destinations

The Degree Hierarchy

Associate Degree (A.S. or A.S.-T)

The associate degree you are working toward at AVC is not a terminal engineering credential — it does not, by itself, qualify you for professional engineering practice. What it *is* is the most effective transfer preparation available. Students who use the two years at community college intentionally arrive at their 4-year institution in excellent position.

The A.S.-T (Associate in Science for Transfer) guarantees admission to the CSU system with junior standing, though not necessarily to a specific campus or program. If you are targeting a CSU, ask your counselor about the A.S.-T in Engineering.

Bachelor's Degree (B.S. in Engineering)

The bachelor's is the **minimum professional credential** for engineering practice. Most entry-level engineering roles require it, and it is the minimum for pursuing an Engineer-in-Training (EIT) credential — the first step toward full Professional Engineer (PE) licensure. A bachelor's program typically takes four years from the start of college, meaning two years at AVC plus two more at a 4-year institution if you transfer on schedule.

Master's Degree (M.S.)

The master's is not required for most engineering jobs, but it is increasingly common in competitive sectors — semiconductor engineering, aerospace, robotics, biomedical devices. It adds one to two years beyond the bachelor's and allows deeper specialization. Many engineers pursue it while working or immediately after the bachelor's; some employers sponsor it.

Doctorate (Ph.D.)

The Ph.D. is a research degree: 4–6 years of original research after the bachelor's. It is appropriate for those who want to lead research groups, teach at the university level, or work in advanced R&D. It is not required — or suited to — most engineering careers.

Transfer Destinations from AVC

UC System — The University of California campuses (UCLA, UC Berkeley, UC Davis, UC San Diego, UC Santa Barbara, UC Irvine, and others) offer highly competitive engineering programs. UC transfer is merit-based and competitive; required GPA varies by campus and major.

CSU System — The California State University campuses (Cal Poly Pomona, Cal Poly SLO, CSUN, CSULB, CSUF, and others) offer accredited ABET engineering programs. Cal Poly campuses have strong engineering reputations.

CSUB at AVC (Continuation Degree) — California State University, Bakersfield has established continuation degree programs directly at the AVC campus:

- **Computer Science (B.S.)** — available Fall 2025
- **Electrical Engineering (B.S.)** — available Fall 2026
- **Mechanical Engineering (B.S.)** — available Fall 2027

These are completion programs: you start at AVC, finish your lower-division work, then complete your upper-division coursework at the CSUB program on the AVC campus. This eliminates the need to relocate for a 4-year degree — a significant advantage for students with family, work, or community ties in the Antelope Valley.

Private institutions — USC, Cal Tech, LMU, and others also accept transfers. Admission standards and financial aid structures vary significantly.

Real-World Example — Why Transfer Planning Is an Engineering Problem

Consider two students, both starting ENGR 110 in the same semester.

Student A completes Calculus I and II in Year 1, takes Physics I concurrently with Calc II, and enters Year 2 with a clean prerequisite path. By the end of Year 2, all general core and most track-specific courses are done. Transfer happens on schedule.

Student B delays Calculus II by one semester — maybe to lighten the load, maybe because of a scheduling conflict. Physics I gets pushed. ENGR 210 (Statics) gets pushed. By the end of Year 2, Student B has a gap in the prerequisite chain and needs a third semester at AVC to complete what would have been a two-year program.

The difference is not talent. It is sequencing. Engineering rewards the student who treats their academic plan like a project schedule: identify the critical path, protect it, and flag delays early enough to recover.

This page titled 3.5: Degree Levels and Transfer Destinations is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

3.6: Transfer Strategy

3.6 Transfer Strategy — How to Actually Do This

Transfer success is not something that happens to you. It is something you build, deliberately, starting in your first semester. Here is a practical framework:

Semester 1: Map the path. Identify your target discipline (even a rough idea is enough to choose a track). Meet with an AVC engineering counselor — once, in writing. Look up your target institutions on ASSIST.org and verify which AVC courses transfer as credit. Identify any prerequisite gaps in math and address them immediately.

Every semester: Protect the math sequence. The single most damaging thing a student can do to their transfer timeline is break the math sequence. Calculus I → II → III → Differential Equations should ideally run as four consecutive semesters. A one-semester gap typically causes measurable skill decay. If you must pause, plan a focused review before returning.

GPA: What programs actually evaluate. Transfer GPA requirements vary by institution. What most engineering programs emphasize is your **technical course GPA** — specifically your performance in calculus and physics. A 3.5 overall GPA with a C in Calculus II will look worse to an engineering program than a 3.2 with an A in Calculus II. A rough benchmark: competitive UC engineering transfer typically requires a 3.5+ technical GPA; CSU programs vary by campus.

Talk to Counselors. Ensure that your educational plan is mapped out and that it aligns with your goals of transfer institutions. Counselors can help determine courses that you may need specific to your university choice.

Use ASSIST.org. Before registering for any course, verify on ASSIST that it counts toward your target degree. Discovering a course doesn't transfer *after* you take it is a preventable waste of time and money.

Apply earlier than you think you need to. UC and most CSU applications are due in the fall for transfer the following fall. That means working on your application in October–November of the year *before* you want to transfer. The timeline arrives faster than most students expect.

> 🔗 **Connection**
>
> The **Career Preparation Project** in this course asks you to research specific companies and job descriptions. That exercise pairs naturally with transfer planning: the 4-year institutions that feed into specific industries — aerospace companies in the Antelope Valley, semiconductor firms in Silicon Valley, defense contractors — will be visible once you start looking at where engineers in your target field actually went to school.

This page titled 3.6: Transfer Strategy is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

3.7: Academic Risk Factors

3.7 Academic Risk Factors — and What To Do About Them

Most students who struggle in engineering programs do not struggle because the material is beyond their capability. They struggle because of **patterns** — study habits, course load decisions, and how they respond to early difficulty — that are entirely fixable.

- **Risk: Weak algebra going into Calculus.** Counter: Work through an algebra review (Khan Academy, ALEKS) during the summer before Calc I. Target your specific gaps — factoring, rational expressions, trig identities.
- **Risk: Treating math as memorization.** Counter: After each lecture, close your notes and rework the example problems from scratch. If you cannot reconstruct the steps without looking, you have not learned it yet.
- **Risk: Falling behind and waiting until the end of the semester.** Counter: Seek help at the first sign of confusion. Office hours are dramatically underused. Tutoring centers work better when used weekly, not in crisis mode before exams.
- **Risk: Overloading the first semester.** Counter: MATH 150 (5 units) + PHYS 110 (4 units) + ENGR 110 (3 units) + one GE course is already a full-time load. Better to go 15 units with strong grades than 18 with mediocre ones.
- **Risk: Social isolation when things get hard.** Counter: Form a study group early in the semester. Engineering students who study collaboratively consistently outperform those who study alone. You are not competing against your classmates.
- **Risk: Ignoring GE courses.** Counter: Technical GPA matters most, but a D or F in English will still hurt your cumulative GPA. Keep GE courses from becoming surprise problems.

⚠ Watch Out — The "I'll Catch Up Next Exam" Trap

Engineering courses are cumulative within themselves, not just across the sequence. Every exam in Circuits builds on the previous one; every problem set in Statics assumes mastery of the concepts from the week before. Bombing an early exam and planning to "make up the points" later is a strategy that rarely works. The more effective response is immediate: identify exactly what you did not understand, go to office hours with specific questions, and rework the problems before the next unit begins.

This page titled 3.7: Academic Risk Factors is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

3.8: Treating Your Academic Plan Like an Engineering Problem

3.8 Treating Your Academic Plan Like an Engineering Problem

The five-step framework introduced in Chapter 1 applies directly to academic planning. Academic planning is a design problem with constraints, dependencies, and real consequences.

- **Define.** What is your goal? "I want to transfer to Cal Poly Pomona's Mechanical Engineering program by Fall 2027 with a 3.4 GPA" is a plan. "I want to be an engineer" is not.
- **Model.** Use the AVC track tables, ASSIST, and your target institution's requirements to map every course semester by semester, including prerequisites.
- **Analyze.** Where are the constraints? The math sequence is almost certainly the critical path. Are there courses offered only in fall or only in spring? What GPA do you need to maintain?
- **Iterate.** Life changes. When something shifts — a course grade lower than expected, a scheduling conflict — revisit the model and adjust early, not at the end of the semester.
- **Communicate.** Share your plan with an AVC counselor at least once per year. They will catch errors you will not catch yourself, especially around articulation requirements that have changed.

Key Takeaway

Engineering education is sequential, cumulative, and dependency-driven. The students who succeed at transfer are not the ones who work the hardest at the end — they are the ones who map the path early, protect the math sequence, maintain consistent technical performance, and ask for help before falling behind. Treat your academic plan as an engineering problem: define your goal, model the path, identify the constraints, and iterate when something changes.

This page titled 3.8: Treating Your Academic Plan Like an Engineering Problem is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

3.9: Summary

Summary

- Engineering coursework is sequential by design — each course depends on real mastery of what came before it.
- The math backbone — Calculus I through Differential Equations — is the same for all disciplines. At AVC: MATH 150, 160, 250, 230 plus PHYS 110 and 120 (25 units total).
- AVC offers three engineering tracks: Mechanical/Aerospace, Electrical, and Computer Engineering, each adding ~23–25 track-specific units on top of the shared core.
- Degree levels range from the associate (transfer preparation) through bachelor's (minimum professional credential), master's (specialization), and doctorate (research).
- Transfer destinations include the UC system, CSU system, and the CSUB at AVC continuation degree program in ME, EE, and CS.
- Transfer success requires early planning, uninterrupted math sequencing, protecting technical GPA, using ASSIST.org, and seeking help before falling behind.

This page titled 3.9: Summary is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

3.10: Quick Check

Quick Check

1. Explain in your own words why you cannot take Dynamics before Statics. What kind of knowledge does Statics develop that Dynamics assumes?
2. Pick one AVC track from §3.4 and identify two courses in it that connect directly to topics covered in this textbook.
3. A student is in their second semester and realizes their Calculus I grade was a C+. Their plan had Physics I starting next semester. What should they do, and why?
4. What is ASSIST.org and why should every AVC transfer student use it before registering for courses?
5. What is the difference between the A.S.-T and a bachelor's degree in terms of what each enables professionally?
6. Apply the five-step planning framework from §3.8 to your own situation. Write out your current answers to Define and Model — even if incomplete.
7. *[Stretch]* Design a four-semester course plan for one of the AVC engineering tracks that completes the general core and all track-specific courses. Identify the critical path and note any courses that can only be taken in a specific semester.

Answers - click to expand

1. You cannot take Dynamics before Statics because Dynamics assumes that you already know how to model forces correctly. Statics develops the foundation for drawing free-body diagrams, identifying supports and reactions, writing force and moment equations, and understanding equilibrium.

   Dynamics builds on those same ideas but adds motion, acceleration, and time. Without Statics, a student may be trying to solve motion problems before they can reliably identify the forces causing the motion.

2. Answers will vary depending on the AVC track selected. For example, in a Mechanical Engineering track, *Statics* connects directly to the textbook's work on units, forces, modeling, and structured problem solving. A CAD or engineering graphics course connects directly to the textbook chapter on parametric CAD and 3D modeling.

   In an Electrical Engineering track, a circuit analysis course connects to the circuit fundamentals chapter, while MATLAB or programming coursework connects to the MATLAB and computational modeling material.

3. The student should not simply continue with the original plan without checking whether the C+ creates

a problem. They should meet with an engineering counselor, review prerequisite rules, and check the expectations of their target transfer universities.

Even if the C+ technically allows them to move forward, Physics I depends heavily on Calculus I skills. The student should honestly evaluate whether their calculus foundation is strong enough. Possible next steps include retaking Calculus I, delaying Physics I, taking a lighter schedule, or completing a structured calculus review before physics begins.

4. ASSIST.org is the official California transfer articulation database. It shows how courses at a California community college, such as AVC, transfer to CSU and UC campuses.

   Every AVC transfer student should use ASSIST.org before registering because it helps confirm which courses count for lower-division major preparation, general education, and transfer requirements. This reduces the risk of taking courses that do not apply to the student's intended university or major.

5. The A.S.-T is an associate degree designed to support transfer preparation. It can help students move efficiently toward a CSU pathway and show that they have completed an organized lower-division program.

   A bachelor's degree is the professional entry-level degree for most engineering positions. It is also the degree that usually supports later engineering licensure pathways, graduate study, and full participation in professional engineering work.

6. Answers will vary. A strong Define response should identify the student's intended engineering discipline, target transfer institutions, degree goal, and major constraints. These constraints may include math placement, course availability, work hours, transportation, family responsibilities, financial aid, and time to transfer.

   A strong Model response should begin turning that goal into a semester-by-semester plan. The model should include prerequisite chains, course sequencing, unit load, critical courses such as calculus and physics, and any track-specific requirements. The model does not have to be perfect, but it should make the student's assumptions visible.

7. Answers will vary depending on the selected AVC engineering track. A strong four-semester plan should include the general engineering core, calculus sequence, physics sequence, chemistry if required, programming or MATLAB if required, and all track-specific courses.

   The critical path will usually include courses such as Calculus I, Calculus II, Calculus III, Differential Equations, Physics I, and Physics II because these courses unlock later engineering coursework. The plan should also identify courses that may only be offered in fall or spring and explain how those scheduling limits affect the student's path.

This page titled 3.10: Quick Check is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

4: ENGINEERING SUCCESS SYSTEMS

Engineering Success in Practice

Academic performance in engineering is not about talent. It is about systems — how you study, how you respond to errors, how you recover, and how you work with others.

Learning Objectives

By the end of this chapter, you will be able to:

- Explain why engineering academic performance is a function of habits and systems, not talent.
- Estimate realistic weekly workload requirements and build a schedule around them.
- Apply deliberate practice and the feedback loop to technical skill development.
- Recognize and classify common student error types to accelerate correction.
- Describe how teamwork functions in engineering and what makes it work well.
- Identify the personal habits — goal-setting, humility, recovery — that sustain performance over a full academic career.

4.1: It Is Not About Talent
4.2: Understanding Your Real Workload
4.3: How to Actually Study Engineering
4.4: Understanding Your Errors
4.5: Goals, Humility, and the Long Game
4.6: Engineering Teamwork
4.7: Sustaining Yourself Over the Long Haul
4.8: Summary
4.9: End-of-Chapter Exercise

This page titled 4: Engineering Success Systems is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

4.1: It Is Not About Talent

4.1 It Is Not About Talent

Let's get the most common misconception out of the way early.

Students who succeed in engineering are not, by and large, more naturally gifted at mathematics than students who struggle. The research on this is quite consistent: at the introductory level, prior preparation and study habits predict performance far better than any measure of "natural ability." The students who tend to thrive treat engineering like a sport — something you get better at through deliberate, consistent practice, not through occasional bursts of effort around deadlines.

This matters because the wrong belief about talent leads to the wrong strategy when things get hard. If you believe performance is fixed ("I'm just not a math person"), then a difficult exam becomes evidence that you don't belong. If you believe performance is built ("I need to practice this type of problem more consistently"), then the same difficult exam becomes a signal about what to work on next.

The engineering framing: **your academic performance is a system.** Systems can be analyzed, improved, and optimized. That is what this chapter is about.

To be successful in engineering school, you need to be humble, organized, positive, goal-oriented, and perseverant. None of those are things you are born with. They are all things you practice.

> **Real-World Example — The Confidence Trap**
>
> There is a well-known pattern in learning: beginners often overestimate how much they understand, because they do not yet know enough to recognize what they are missing. Experts, by contrast, tend to underestimate themselves, because they are acutely aware of how much remains unknown.
>
> The lesson is not that confidence is bad. It is that genuine competence requires first recognizing the gap between what you can repeat and what you can actually use under pressure. Every student who has studied their notes for hours but then blanked on an exam has experienced this gap firsthand. The fix is not more reading — it is more doing.

This page titled 4.1: It Is Not About Talent is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

4.2: Understanding Your Real Workload

4.2 Understanding Your Real Workload

One of the most reliable ways to undermine your first semester is to underestimate how much time engineering actually takes. So let's do the math.

A standard benchmark is **2–3 hours of outside work per credit unit per week**. For engineering courses, the upper end is more realistic. Here is what a typical engineering semester looks like:

Estimated weekly study time for a typical 15-unit engineering semester.

Course	Units	Est. Study Hours/Week
Calculus (MATH 150 or 160)	4–5	8–10
Physics (PHYS 110 or 120)	4	8–10
Programming / MATLAB	3–4	6–8
Engineering or GE course	3	4–6
Total	**~15 units**	**~26–34 hrs study**

Add 15 hours of in-class time, and a 15-unit engineering semester requires **roughly 40–50 hours per week** of combined class and study time. Here is what remains in a typical week:

$$168 \text{ total hours} - 56 \text{ (sleep, 8 hrs/night)} - 45 \text{ (class + study)} = 67 \text{ remaining hours}$$

Those 67 hours cover eating, commuting, errands, family, a job, social time, and recovery. For many AVC students who work or have family responsibilities, that number is smaller. A few practical rules:

- **Protect the morning.** Cognitive work is best done when you are fresh. Schedule hard study early if you have the choice.
- **Count commute time honestly.** A 45-minute commute each way is 7.5 hours per week not available for studying.
- **If you work more than 20 hours/week, reduce your course load.** A 12-unit semester with strong grades is worth

more than a 15-unit semester with mediocre ones.

⚠ Watch Out — "I'll work harder during finals"

Engineering courses are cumulative within each semester. Material from Week 3 appears on every exam after Week 3. A student who falls behind in calculus in October and plans to catch up in December is hoping, not planning. Steady weekly effort with early course corrections is the strategy that works.

This page titled 4.2: Understanding Your Real Workload is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

4.3: How to Actually Study Engineering

4.3 How to Actually Study Engineering

Most students arrive at college with a study strategy that worked in high school: read the notes, highlight things, re-read before the test. In engineering, that strategy produces students who *recognize* correct solutions without being able to *generate* them.

The distinction matters enormously on engineering exams, where the problem you are given is almost never identical to any example you have seen. You need to be able to construct a solution, not recall one.

Active problem solving is the core practice:

1. After lecture, close your notes and attempt the assigned problems from scratch — without looking at examples first. If you cannot start, open your notes briefly to reorient, then close them and attempt again.
2. When you find an error, don't just fix it. **Classify it** — see §4.4. Understanding *why* you got something wrong is more valuable than knowing the correct answer.
3. Rework problems you got wrong from the beginning, without referencing the solution. If you can reconstruct it correctly, you have learned it. If you can't, you haven't.

Focused sessions beat marathon sessions. Research on learning consistently shows that 45–90 minute focused blocks with short breaks outperform 3–4 hour sessions with degrading attention. Engineering work has high cognitive load — you are tracking units, managing algebra, and reasoning through physical logic simultaneously. That depletes quickly when attention wanders.

Studying while watching TV, half-checking a phone, or in a noisy environment measurably increases errors and decreases retention. This is not a moral judgment — it is a cognitive one.

Use a weekly rhythm, not last-minute surges:

Weekly study rhythm for engineering courses.

Timing	Activity
Same day as lecture	Preview the next topic (15 min); review today's notes while still fresh
Within 24 hours	Attempt assigned problems; flag the ones you cannot start

Timing	Activity
Midweek	Office hours with specific questions — not "I don't get it" but "I tried this and got stuck here"
Weekend	Cumulative review; rework any problems that were wrong earlier in the week

This page titled 4.3: How to Actually Study Engineering is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

4.4: Understanding Your Errors

4.4 Understanding Your Errors

When you get a problem wrong, the most valuable thing you can do is not look at the answer — it is figure out *which kind* of wrong you were. Engineering errors fall into five categories, and each has a different fix:

Common engineering error types, how they appear, and how to correct them.

Error Type	What it looks like	Fix
Conceptual	Wrong governing equation chosen	Review the underlying principle; understand when each equation applies
Structural	Skipped Define or Model step; jumped straight to calculating	Practice the full five-step process; always draw diagrams and define variables before calculating
Algebraic	Correct setup, arithmetic or symbol error in execution	Slow down; write every step explicitly; check intermediate results before continuing
Unit	Mixed unit systems, wrong prefix, forgot to convert	Write units alongside every quantity at every step — not just at the end
Interpretation	Calculation correct but answer not evaluated for reasonableness	Build the habit of asking "does this number make sense?" before writing it down

Without classification, students often study *more* of what they already understand while the actual source of their errors goes unaddressed. Classifying errors makes studying targeted rather than indiscriminate.

✓ Worked Example — Classifying an Error

A student solves $V = IR$ and gets $I = 0.5\ \text{A}$, but the correct answer is $I = 5\ \text{A}$. What type of error is it?

- If R was entered as 2 Ω instead of 0.2 Ω → **structural error** (wrong input value).
- If Ohm's Law was misapplied entirely → **conceptual error**.
- If the student used 20 mA rather than 0.020 A in the calculation → **unit error**.

Each diagnosis leads to a different study action. Without the diagnosis, the student might just redo more Ohm's Law problems, when the actual problem was a unit conversion habit.

⚠ Watch Out — "I knew how to do it, I just made a careless mistake"

"Careless mistakes" are usually not random. When the same type of careless mistake recurs, it is a pattern — and patterns have causes. Misread signs almost always mean rushing. Dropped units almost always mean units were not written throughout. Wrong formula almost always means the conceptual picture was not drawn first. Treating recurring errors as bad luck prevents you from fixing them.

This page titled 4.4: Understanding Your Errors is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

4.5: Goals, Humility, and the Long Game

4.5 Goals, Humility, and the Long Game

Engineering is a long road. You are in the first semester of a path that leads to a bachelor's degree in roughly four years, and then a career that may span forty. The habits you build now matter disproportionately — not because first-semester grades follow you forever, but because first-semester habits tend to.

Set goals at multiple timescales. Short-term goals (finish Calc I with a B+, meet with a counselor by Week 3, form a study group) give you something concrete to execute this week. Long-term goals (transfer to Cal Poly, work in aerospace, start a company) give you a reason to keep executing when the semester gets hard. Goals without timescales are wishes; goals with timescales are plans.

Stay humble about what you know — and what you don't. One of the most reliable warning signs in engineering students is premature confidence: the feeling, usually after understanding one or two key concepts, that you "get it" and don't need to keep practicing. The better you actually understand a topic in engineering, the more clearly you see how much remains to learn. That is not discouraging — it is accurate. And accurate self-assessment is the foundation of knowing where to direct your effort.

Humility in practice looks like:

- Going to office hours even when you passed the last exam, because you know there were things you were lucky on.
- Redoing problems you got right to make sure you can reproduce the reasoning, not just the answer.
- Listening when your study group explains a solution differently than you would — even when your approach also works.
- Asking "does this make sense?" before moving on, rather than stopping once you have a number.

Perseverance is a skill, not a personality trait. "I'm just not the type to push through when things get hard" is a story, not a fact. Perseverance is a practiced response to difficulty — and the engineering path will contain difficulty. The students who finish are not the ones who never struggle; they are the ones who have developed a reliable response to struggle that involves working through it rather than around it.

> 🔗 **Connection**
>
> The **Personal Success and Creative Thinking** homework asks you to write out your short-term, long-term, and career goals, build a weekly timetable, and honestly identify the barriers that prevent you from sticking to it. That exercise puts this section into practice — actually writing out your goals and schedule is the first step toward following through on them.

This page titled 4.5: Goals, Humility, and the Long Game is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

4.6: Engineering Teamwork

4.6 Engineering Teamwork

Nearly every engineering program and nearly every engineering job involves working in teams. This course will almost certainly involve team-based projects. It is worth thinking carefully about what makes engineering teams work — and what makes them fail.

Everyone is different — and that is the point. No two people learn or work in exactly the same way. Some work best through direct hands-on experience; others through analysis and theory first. Teams that include both types of learners tend to be more robust than teams of people who all approach problems the same way. The person with mechanical intuition and the person who thinks in systems often catch each other's blind spots.

What makes engineering teams work:

- **Defined roles.** Agree explicitly on who does what and when deliverables are due before any project starts. Ambiguity about responsibility is the most common cause of dropped work.
- **Honest communication about progress.** The most dangerous team member is the one who says "I've got it" and hasn't started. Problems caught early can be fixed; problems caught at the deadline usually cannot.
- **Constructive conflict.** Good engineering teams disagree about approaches. That is healthy — it surfaces assumptions and exposes weak reasoning.
- **Showing up.** Teams that meet regularly, even briefly, outperform teams that coordinate only by text. You catch problems faster and build shared context.

What makes engineering teams fail:

- One person carries the load while others free-ride.
- Team members avoid conflict by accepting poor work rather than giving honest feedback.
- Roles are undefined, so some work is duplicated and other work is missed entirely.
- The team only communicates when a deadline is approaching.

In professional engineering, team failures are everyone's failures. The bridge does not stand or fall based on who did their part — it stands or falls as a system. Taking shared ownership of team outcomes, not just your piece, is a professional skill worth building now.

Ethics Check

If a team member is consistently not contributing, what are your obligations — to the project, to them, and to yourself? At what point is it appropriate to escalate to an instructor? And when, if ever, is it acceptable to put someone's name on work they did not do? These situations come up in student teams and in professional ones. It is worth thinking through your principles before you are in the middle of one.

This page titled 4.6: Engineering Teamwork is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

4.7: Sustaining Yourself Over the Long Haul

4.7 Sustaining Yourself Over the Long Haul

Engineering education is a multi-year commitment. The habits that get you through a single hard week are different from the ones that sustain you across four years.

Recovery is not optional. Sleep, exercise, and social connection are inputs to cognitive performance, not rewards you earn after the coursework is done. The cognitive science here is unambiguous: sleep deprivation degrades working memory, error detection, and problem-solving ability. A student who studies for six hours on five hours of sleep will typically retain less than one who studies for four hours on eight. When workload feels unsustainable, the productive response is not "work harder" — it is to diagnose the system: What specifically is taking longer than expected? Is it a content gap, a time management issue, or a genuine overload?

Build a support system early. Professors, a study group, counselors, tutoring services — these resources exist and work best when you engage them before crisis, not during it. Asking for help when you are ahead is engineering thinking. Waiting until you are behind and hoping no one notices is not.

You belong here. Engineering programs serve students from a wide range of backgrounds — students who work full-time, who are first-generation college students, who are returning after time away, who came from schools with limited advanced coursework. None of those things are disqualifiers. They are circumstances. The habits in this chapter work regardless of your starting point. What matters is consistent practice, not where you started.

> **Key Takeaway**
>
> Academic success in engineering is not about talent — it is about system design. Estimate your workload realistically, protect focused study time, practice actively, classify your errors, set goals at multiple timescales, and build a support structure around you. The students who succeed are not the ones who never struggle. They are the ones who have built reliable systems for what to do when they do.

This page titled 4.7: Sustaining Yourself Over the Long Haul is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

4.8: Summary

Summary

- Engineering academic performance is a function of habits and systems, not innate talent.
- A 15-unit engineering semester requires approximately 40–50 hours per week of combined class time and study. Build your schedule around this reality.
- Effective study is active — solving problems from scratch, classifying errors, reworking incorrect problems — not passive reading or re-highlighting.
- Engineering errors fall into five types: conceptual, structural, algebraic, unit, and interpretation. Each has a distinct fix.
- Goals at multiple timescales — short-term, long-term, career — provide both direction and motivation for consistent effort.
- Humility and honest self-assessment are engineering competencies: knowing what you do not know is the first step to addressing it.
- Teamwork is integral to engineering. Effective teams define roles clearly, communicate honestly, and take shared ownership of outcomes.
- Sustainable performance requires recovery — sleep, support systems, and realistic workload management.

This page titled 4.8: Summary is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

4.9: End-of-Chapter Exercise

End-of-Chapter Exercise

This exercise puts the chapter into practice. Work through each step as a planning document for your own semester.

1. **Define your goals.** Write one short-term goal (achievable this semester), one long-term goal (1–3 years), and one career goal. Be specific enough that you can tell whether you achieved it.
2. **Model your week.** Create a table mapping a typical week by hour block. Include class time, commute, work, family obligations, and realistic sleep. How many hours are actually available for studying?
3. **Analyze the constraints.** Compare your available hours to the estimates in §4.2. Where is the gap, if any? Which of your current courses is highest risk given your schedule?
4. **Identify your error patterns.** Think back to the last technical course where you lost points on a test or homework. Which error type from §4.4 best describes most of your mistakes?
5. **Design your response.** Given your constraints and error patterns, write two or three concrete changes you will make to your study approach this semester.

Answer example - click to expand

1. **Define your goals.**

   Short-term goal: Earn at least a B in Calculus I this semester by completing every homework assignment, attending office hours at least twice before each exam, and correcting missed quiz problems within one week.

   Long-term goal: Complete Calculus I, Calculus II, Physics I, and Introduction to Engineering within the next three semesters so I stay on track for transfer preparation.

   Career goal: Transfer into a mechanical engineering program and eventually work on mechanical design, robotics, manufacturing systems, or another field where I can design and test physical products.

2. **Model your week.**

Example weekly time budget for a student planning an engineering semester.

Activity	Hours per week	Notes
Class time	15	Calculus, engineering, English, a
Work	20	Part-time job
Commute	6	Driving to campus and work
Family and household responsibilities	10	Meals, errands, chores, and famil
Sleep	56	8 hours per night
Meals, hygiene, and basic daily routines	14	About 2 hours per day
Total committed time	**121**	Out of 168 hours per week
Available study/flex time	**47**	Time available for homework, stu

In this example, about 47 hours per week remain for studying and flexible time. That does not mean all 47 hours are easy to use; some of that time will be fragmented, so planned study blocks are still necessary.

3. **Analyze the constraints.**

   If the student is taking 15 units, a rough planning estimate is that they may need about 30 hours per week outside of class for reading, homework, problem practice, labs, and exam preparation. With 47 hours available, the schedule is possible, but not forgiving.

   The highest-risk course is Calculus I because it is both time-intensive and part of the engineering critical path. If the student falls behind in Calculus I, later courses such as Calculus II, Physics I, and engineering analysis may also be delayed. The second risk is work schedule variability; if work hours increase during exam weeks, study time may disappear quickly.

4. **Identify your error patterns.**

   In the last technical course, most lost points came from setup and process errors rather than from not knowing the topic at all. Common mistakes included skipping units, starting calculations before identifying knowns and unknowns, not checking whether the answer was reasonable, and making algebra mistakes under time pressure.

   The main error pattern is therefore **weak problem setup**. The student understands many procedures after seeing an example, but does not always slow down enough to structure the problem independently.

5. **Design your response.**

Based on the constraints and error patterns, the student will make three changes:

1. **Use a fixed problem-solving format.** For every technical homework problem, write knowns, unknowns, diagram or model, equation, substitution with units, result, and reasonableness check.
2. **Schedule calculus practice before homework is due.** Instead of doing homework only the night before, reserve three 90-minute calculus blocks each week for problem practice and review.
3. **Use feedback faster.** After every quiz or exam, redo missed problems within one week and write one sentence explaining the cause of each error.

The purpose is not just to spend more time studying. The purpose is to make study time more structured, reduce repeated mistakes, and protect the courses that control the engineering sequence.

This page titled 4.9: End-of-Chapter Exercise is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

5: UNITS AND DIMENSIONAL ANALYSIS

Units and Dimensional Analysis

A quantity is a number *and* a unit — both are required. Unit errors do not produce obvious failures; they produce answers that look almost right until the moment they matter most.

Learning Objectives

By the end of this chapter, you will be able to:

- Distinguish between a number and a physical quantity.
- Identify the seven SI base units and construct derived units from them.
- Apply dimensional analysis as an algebraic technique to convert units and verify equations.
- Perform multi-step unit conversion chains with complete cancellation.
- Interpret engineering scale using SI prefixes from pico to tera.
- Detect and correct unit errors in computational models.
- Apply order-of-magnitude reasoning to evaluate whether a result is physically plausible.

5.1: Why Units Matter
5.2: Quantities vs. Numbers
5.3: SI System — Base and Derived Units
5.4: Unit Conversion Method
5.5: SI Prefixes
5.6: Dimensional Analysis
5.7: Units in Computational Models
5.8: Order-of-Magnitude Reasoning
5.9: From Units to Everything That Follows
5.10: Summary
5.11: End-of-Chapter Problem Set

This page titled 5: Units and Dimensional Analysis is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

5.1: Why Units Matter

5.1 Why Units Matter More Than You Think

Here is a quick question: What is $F = 120$?

Is it 120 newtons? 120 pound-force? 120 kilograms? 120 watts? You cannot answer that from the number alone. The unit is not a label attached after the math is done — it is part of the physical meaning of the quantity. A number without a unit is incomplete in exactly the same way a price without a currency is incomplete.

Engineers work with physical quantities, not abstract numbers. Every measurement, every calculation input, every result must carry its unit explicitly. This discipline is not optional, and the reason is not merely convention — it is that **unit errors have caused real, expensive, and sometimes fatal failures in engineering history.**

⛐ Real-World Example — Mars Climate Orbiter (Full Case)

In September 1999, NASA's Mars Climate Orbiter approached Mars after a 286-day journey. It was supposed to enter a stable orbit at roughly 150 km altitude. Instead, it passed within 57 km of the surface and was destroyed by atmospheric heating.

The cause: One engineering team provided thruster impulse data in **pound-force seconds (lbf·s)**; the other team's navigation software expected **newton-seconds (N·s)**. Neither team caught the discrepancy because both numbers looked plausible in isolation. Over 286 days, small per-burn errors accumulated into a trajectory that missed the intended orbit entry by over 100 km.

The cost: $327 million. The mission was lost completely.

The lesson: Unit errors do not cause dramatic, obvious failures. They produce numbers that look *almost* right — until the moment they matter most. The investigation board's primary recommendation was stronger unit-checking protocols throughout the design process.

Source: NASA Mars Climate Orbiter Mishap Investigation Board Phase I Report, November 1999.

This page titled 5.1: Why Units Matter is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

5.2: Quantities vs. Numbers

5.2 Quantities vs. Numbers

Every engineering quantity has two parts:

$$\text{Quantity} = \text{Magnitude} \times \text{Unit}$$

Units participate in arithmetic just like numbers. When you add, subtract, multiply, or divide quantities, the units go along for the ride:

Addition requires identical units

$$25\ \text{m} + 10\ \text{m} = 35\ \text{m} \quad \checkmark$$

$$25\ \text{m} + 10\ \text{s} = \text{undefined} \quad \times$$

Multiplication creates new units

$$5\ \text{kg} \times 2\frac{\text{m}}{\text{s}^2} = 10\ \frac{\text{kg} \cdot \text{m}}{\text{s}^2} = 10\ \text{N}$$

Division cancels units

$$\frac{120\ \text{m}}{4\ \text{s}} = 30\ \frac{\text{m}}{\text{s}}$$

This algebraic behavior is the foundation of dimensional analysis.

This page titled 5.2: Quantities vs. Numbers is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

5.3: SI System — Base and Derived Units

5.3 The SI System — Base and Derived Units

Base Units

SI base quantities, base units, and unit symbols.

Quantity	SI Base Unit	Symbol
Length	meter	m
Mass	kilogram	kg
Time	second	s
Electric current	ampere	A
Temperature	kelvin	K
Amount of substance	mole	mol
Luminous intensity	candela	cd

For most engineering work at this level, you will primarily use meters, kilograms, seconds, and amperes. The others become important in thermodynamics, chemistry, and photonics.

Derived Units

All other units are derived by combining base units. The derivations encode the physics of the relationship.

Common SI derived units and their equivalent base-unit forms.

Quantity	Symbol	In Base Units
Force	N (newton)	$\mathrm{kg} \cdot \mathrm{m} \cdot \mathrm{s}^{-2}$
Energy / Work	J (joule)	$\mathrm{kg} \cdot \mathrm{m}^{2} \cdot \mathrm{s}^{-2}$

Quantity	Symbol	In Base Units
Power	W (watt)	$\mathrm{kg \cdot m^2 \cdot s^{-3}}$
Pressure / Stress	Pa (pascal)	$\mathrm{kg \cdot m^{-1} \cdot s^{-2}}$
Electrical resistance	Ω (ohm)	V/A

ADAPT 5.3.1

This interactive element is not available in this format. Visit the link or scan the QR code to access it online.

View this Interactive Content online

This page titled 5.3: SI System — Base and Derived Units is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

5.4: Unit Conversion Method

5.4 Unit Conversion — The Conversion Factor Method

Converting units is one of the most common tasks in engineering. The reliable method uses **conversion factors written as fractions equal to 1**, multiplied through the original quantity so that unwanted units cancel.

Since $1\ \text{ft} = 0.3048\ \text{m}$, the fraction $\dfrac{0.3048\ \text{m}}{1\ \text{ft}} = 1$. Multiplying by 1 preserves the physical quantity while changing the unit.

✓ Worked Example 5.1 — Force Conversion

Problem: Convert 45 lb_f to newtons.

Solution:

$$45\ \cancel{\text{lb}_\text{f}} \times \frac{4.448\ \text{N}}{1\ \cancel{\text{lb}_\text{f}}} = 200.2\ \text{N}$$

Result: $F = 200\ \text{N}$ (3 significant figures)

Check: 1 lbf ≈ 4.4 N, so 45 lbf ≈ 200 N. ✓

✓ Worked Example 5.2 — Multi-Step Velocity Conversion

Problem: Convert 72 km/h to m/s.

Strategy: Convert km → m, then h → s.

$$72\ \frac{\cancel{\text{km}}}{\cancel{\text{h}}} \times \frac{1000\ \text{m}}{1\ \cancel{\text{km}}} \times \frac{1\ \cancel{\text{h}}}{3600\ \text{s}} = \frac{72{,}000}{3600}\frac{\text{m}}{\text{s}} = 20\ \frac{\text{m}}{\text{s}}$$

Result: $v = 20\ \text{m/s}$

Check: 72 km/h ≈ 45 mph. 20 m/s ≈ 66 ft/s. Reasonable highway speed. ✓

✓ Worked Example 5.3 — Volume Conversion

Problem: Convert 0.45 m³ to cm³.

Key step — cube the linear conversion:

$$1\ \text{m}^3 = (100\ \text{cm})^3 = 10^6\ \text{cm}^3$$

$$0.45\ \cancel{\text{m}^3} \times \frac{10^6\ \text{cm}^3}{1\ \cancel{\text{m}^3}} = 450{,}000\ \text{cm}^3$$

Common error: Using 100 instead of 10^6 gives 45 cm^3 — off by a factor of 10,000. Always cube (or square) the conversion factor when converting cubed (or squared) units.ADAPT 5.4.1

ADAPT 5.4.1

This interactive element is not available in this format. Visit the link or scan the QR code to access it online.

View this Interactive Content online

ADAPT 5.4.2

This interactive element is not available in this format. Visit the link or scan the QR code to access it online.

View this Interactive Content online

ADAPT 5.4.3

This interactive element is not available in this format. Visit the link or scan the QR code to access it online.

View this Interactive Content online

This page titled 5.4: Unit Conversion Method is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

5.5: SI Prefixes

5.5 SI Prefixes and Scale Awareness

Common SI prefixes, symbols, scale factors, and engineering examples.

Prefix	Symbol	Factor	Example in engineering
pico	p	10^{-12}	1 pF capacitor
nano	n	10^{-9}	1 nm chip feature size
micro	μ	10^{-6}	1 μA sensor current
milli	m	10^{-3}	1 mV sensor output
(base)	—	10^{0}	1 m, 1 A, 1 kg
kilo	k	10^{3}	1 kΩ resistor
mega	M	10^{6}	1 MPa material stress
giga	G	10^{9}	1 GHz processor speed
tera	T	10^{12}	1 TWh energy statistics

⚠ Watch Out — milli vs. Mega

m (lowercase) = milli = 10^{-3}

M (uppercase) = mega = 10^{6}

The difference between mA (milliamp) and MA (megaamp) is a factor of 10^{9}. One is a current you might see in a sensor circuit. The other would vaporize any conductor on Earth. Check your cases every time.

✓ Worked Example 5.4 — Prefix Conversion

A resistor is labeled $4.7\ \mathrm{k\Omega}$. A capacitor is labeled $22\ \mu\mathrm{F}$. Express each in base units.

$$4.7\ \text{k}\Omega = 4.7 \times 10^{3}\ \Omega = 4{,}700\ \Omega$$

$$22\ \mu\text{F} = 22 \times 10^{-6}\ \text{F} = 2.2 \times 10^{-5}\ \text{F}$$

Why this matters: In computational models, enter 4700 and 0.0000220 — not 4.7 and 22. Entering prefixed values directly is one of the most common sources of factor-of-1000 errors in student calculations.

ADAPT 5.5.1

This interactive element is not available in this format. Visit the link or scan the QR code to access it online.

View this Interactive Content online

ADAPT 5.5.2

This interactive element is not available in this format. Visit the link or scan the QR code to access it online.

View this Interactive Content online

This page titled 5.5: SI Prefixes is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

5.6: Dimensional Analysis

5.6 Dimensional Analysis — Verifying Equations

Dimensional analysis checks whether an equation is physically correct. The rule is simple: **both sides of any valid equation must have identical dimensions.**

✓ Worked Example 5.5 — Verifying $\sigma = F/A$

Left side:

$$[\sigma] = \text{Pa} = \frac{\text{N}}{\text{m}^2} = \frac{\text{kg} \cdot \text{m/s}^2}{\text{m}^2} = \frac{\text{kg}}{\text{m} \cdot \text{s}^2}$$

Right side:

$$\left[\frac{F}{A}\right] = \frac{\text{N}}{\text{m}^2} = \frac{\text{kg}}{\text{m} \cdot \text{s}^2}$$

Conclusion: Both sides reduce to $\frac{\text{kg}}{\text{m} \cdot \text{s}^2}$. Dimensionally consistent. ✓

✓ Worked Example 5.6 — Verifying $P = V^2/R$

Right side (using $[V]$ = V, $[R]$ = Ω = V/A, $[P]$ = W = V·A):

$$\left[\frac{V^2}{R}\right] = \frac{\text{V}^2}{\text{V/A}} = \text{V}^2 \times \frac{\text{A}}{\text{V}} = \text{V} \cdot \text{A} = \text{W}$$

Conclusion: Both sides are watts. Dimensionally consistent. ✓

Practice — Spot the Invalid Equation

For each equation, determine whether it is dimensionally valid. If not, identify what is wrong.

a) $F = mv$ (F in N, m in kg, v in m/s)

b) $P = VI$ (P in W, V in V, I in A)

c) $V = I + R$ (Ohm's Law written incorrectly)

d) $T = mg\theta$ (T in N, m in kg, g in m/s², θ in radians)

Answers - click to expand

a. $F = mv$ **is dimensionally invalid.**

$$[F] = \mathrm{N} = \mathrm{kg} \cdot \mathrm{m/s}^2$$

The right side is mass times velocity:

$$[mv] = \mathrm{kg} \cdot \mathrm{m/s}$$

The right side has dimensions of momentum, not force. To produce force, the equation would need acceleration, not velocity: $F = ma$.

b. $P = VI$ **is dimensionally valid.**

$$[VI] = \mathrm{V} \cdot \mathrm{A} = \mathrm{W}$$

Both sides have units of watts, so the equation is dimensionally consistent.

c. $V = I + R$ **is dimensionally invalid.**

$$[V] = \mathrm{V}$$

The right side attempts to add current and resistance:

$$[I + R] = \mathrm{A} + \Omega$$

You cannot add quantities with different dimensions. Ohm's Law should multiply current and resistance: $V = IR$.

d. $T = mg\theta$ **is dimensionally valid, but only because radians are dimensionless.**

$$[mg\theta] = \text{kg} \cdot \text{m/s}^2 \cdot 1 = \text{N}$$

Since θ in radians is dimensionless, both sides have units of newtons. However, dimensional consistency does not prove the equation is physically correct. An equation can pass the dimensional check and still be the wrong model.

ADAPT 5.6.1

This interactive element is not available in this format. Visit the link or scan the QR code to access it online.

View this Interactive Content online

ADAPT 5.6.2

This interactive element is not available in this format. Visit the link or scan the QR code to access it online.

View this Interactive Content online

This page titled 5.6: Dimensional Analysis is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

5.7: Units in Computational Models

5.7 Units in Computational Models

Spreadsheets and programming environments process numbers, not units. When you type `R = 4.7`, the software has no idea whether that is 4.7 Ω, 4.7 kΩ, or 4.7 MΩ. If your intent and your entry disagree, the calculation runs without error — and produces a wrong answer.

✓ Worked Example 5.7 — Hidden Unit Error in a Model

Problem: A circuit has a 24 V source and a 4.7 kΩ resistor. Find the current.

Correct approach

$$R = 4.7\ \text{k}\Omega = 4700\ \Omega \qquad I = \frac{24\ \text{V}}{4700\ \Omega} = 0.00511\ \text{A} = 5.11\ \text{mA}$$

Common error — entering kilohms directly

$$I = \frac{24}{4.7} = 5.11\ (\text{but this is actually A, not mA — a factor of 1000 off})$$

In a real circuit, 5.11 A through a component rated for milliamps would destroy it. The model runs without complaint; the engineer fails silently.

Modeling checklist — apply before running any calculation

- Convert all inputs to SI base units (Ω not kΩ, Pa not MPa, m not mm).
- Label every input variable with its unit in a comment or adjacent cell.
- After solving, check the unit of the result — does it match what you expect?
- Apply the magnitude sanity check — does the number make physical sense?

This page titled 5.7: Units in Computational Models is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

5.8: Order-of-Magnitude Reasoning

5.8 Order-of-Magnitude Reasoning

Unit discipline catches errors in kind — "this should be newtons, not kilograms." Order-of-magnitude reasoning catches errors in scale — "this number is dimensionally correct but suspiciously large." Both checks are part of professional engineering judgment.

Typical order-of-magnitude ranges for common engineering quantities.

Quantity	Typical range
Structural force (small component)	$10\ \text{N}$ to $10\ \text{kN}$
Atmospheric pressure	$\approx 101\ \text{kPa} = 1.01 \times 10^5\ \text{Pa}$
Electrical current (signal circuits)	μA to mA
Electrical current (power circuits)	$1\ \text{A}$ to $100\ \text{A}$
Resistor values (typical)	$10\ \ \Omega$ to $1\ \ \text{M}\Omega$
Structural steel stress (typical)	$100\ \text{MPa}$ to $500\ \text{MPa}$

Ethics Check

Professional engineers sign and stamp their calculations. When an engineer certifies a calculation, they are asserting that units are consistent, dimensions are valid, magnitudes are reasonable, and assumptions are transparent. Unit negligence is not a minor academic error — in a professional context, it is grounds for liability and, depending on consequences, for loss of licensure. The habits you build now are the habits that make certification meaningful.

ADAPT 5.8.1

This interactive element is not available in this format. Visit the link or scan the QR code to access it online.

View this Interactive Content online

This page titled 5.8: Order-of-Magnitude Reasoning is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

5.9: From Units to Everything That Follows

5.9 From Units to Everything That Follows

This chapter is first in Part II for a specific reason: every technical chapter that follows depends on it.

- **Chapter 6 (Problem Solving)** — the structured solution format requires units at every step.
- **Chapter 8 (Spreadsheet Modeling)** — all model inputs must be in base units before the spreadsheet processes them.
- **Chapter 10 (MATLAB)** — MATLAB performs arithmetic on numbers; unit consistency is entirely the engineer's responsibility.
- **Chapter 11 (Circuits)** — volts, ohms, and amperes mix with prefixes constantly; the kΩ error from §5.7 is one of the most common circuit calculation mistakes.
- **Chapter 12 (CAD)** — a model built in inches interpreted as millimeters produces geometry that is 25.4× too large.

Key Takeaway

A quantity is a number **and** a unit — both are required. Units behave algebraically: they multiply, divide, and cancel just like variables. The conversion factor method converts units systematically. Dimensional analysis verifies equations. SI prefix errors are a frequent source of factor-of-1000 mistakes that computational tools will not catch. Building the habit of unit discipline now prevents errors that are embarrassing in class and catastrophic in practice.

This page titled 5.9: From Units to Everything That Follows is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

5.10: Summary

Summary

- Every physical quantity is a magnitude multiplied by a unit. Numbers without units are physically incomplete.
- SI defines seven base units; all other engineering units are derived through multiplication and division.
- Unit conversion uses conversion factors as fractions equal to 1. Squared and cubed units require squaring and cubing the conversion factor.
- SI prefixes span from pico (10^{-12}) to tera (10^{12}). Lowercase m (milli) and uppercase M (mega) differ by 10^{9}.
- Dimensional analysis checks equation validity: both sides must reduce to the same base units.
- Computational tools do not track units. Convert all inputs to base units before entry; verify the output unit and magnitude.
- Order-of-magnitude reasoning provides a second layer of error detection beyond dimensional correctness.

This page titled 5.10: Summary is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

5.11: End-of-Chapter Problem Set

End-of-Chapter Problem Set

Unit Conversions

1. Convert 150 ft to meters.
2. Convert 120 lb_f to newtons.
3. Convert 35 mph to m/s.
4. Convert 0.0025 m³ to liters. *(Note:* $1\ \text{L} = 10^{-3}\ \text{m}^3$*)*
5. Convert 85 kPa to Pa.
6. Convert 3.2 GPa to MPa.
7. Convert 450 µA to mA.
8. Convert 330 km/h to ft/s. *(Multi-step: use both metric and imperial conversions.)*

Dimensional Verification

9. Verify the dimensional consistency of $F = ma$.
10. Verify the dimensional consistency of $P = I^2R$.
11. Determine whether $v = at^2$ is dimensionally valid. If not, what is wrong?
12. Determine whether $E = mv$ is dimensionally valid. If not, what is the correct form?

Applied Problems

13. A 12 V source is connected to a 330 Ω resistor. Calculate the current in mA and the power in mW.
14. A student calculates a cable tension of 850 kg. Identify the error type from Chapter 4's error taxonomy and state the correct unit.
15. A 24 V source is connected to a 4.7 kΩ resistor in a spreadsheet model. The student enters $R = 4.7$. Calculate the current the model will compute, compare it to the correct current, and state the factor of error.
16. A pressure sensor reports 2.3×10^{-2} GPa. Convert this to kPa.
17. *[Challenge]* A container holds 0.75 m³ of water. Density of water = 1000 kg/m³. Calculate the mass in kg and weight in N and lb_f. Show all unit cancellations explicitly.

Answer Key - click to expand

Unit Conversions

1. $150\ \text{ft} \times \dfrac{0.3048\ \text{m}}{1\ \text{ft}} = 45.72\ \text{m}$
2. $120\ \text{lb}_\text{f} \times \dfrac{4.448\ \text{N}}{1\ \text{lb}_\text{f}} = 533.8\ \text{N}$
3. $35\ \text{mph} \times \dfrac{1609.34\ \text{m}}{1\ \text{mi}} \times \dfrac{1\ \text{hr}}{3600\ \text{s}} = 15.65\ \text{m/s}$
4. $0.0025\ \text{m}^3 \times \dfrac{1\ \text{L}}{10^{-3}\ \text{m}^3} = 2.5\ \text{L}$
5. $85\ \text{kPa} \times \dfrac{1000\ \text{Pa}}{1\ \text{kPa}} = 85{,}000\ \text{Pa}$
6. $3.2\ \text{GPa} \times \dfrac{1000\ \text{MPa}}{1\ \text{GPa}} = 3200\ \text{MPa}$
7. $450\ \mu\text{A} \times \dfrac{1\ \text{mA}}{1000\ \mu\text{A}} = 0.45\ \text{mA}$
8. $330\dfrac{\text{km}}{\text{h}} \times \dfrac{1000\ \text{m}}{1\ \text{km}} \times \dfrac{3.28084\ \text{ft}}{1\ \text{m}} \times \dfrac{1\ \text{h}}{3600\ \text{s}} = 300.7\dfrac{\text{ft}}{\text{s}}$

Dimensional Verification

9. $F = ma$ is dimensionally consistent. Since $[m][a] = M \cdot \dfrac{L}{T^2} = \dfrac{ML}{T^2}$, and force has dimensions $\dfrac{ML}{T^2}$, the equation is valid.
10. $P = I^2R$ is dimensionally consistent. Since $R = \dfrac{V}{I}$, then $I^2R = I^2\left(\dfrac{V}{I}\right) = IV$. Current times voltage gives power, so the result is watts.
11. $v = at^2$ is not dimensionally valid. The right side has dimensions $[a][t^2] = \dfrac{L}{T^2} \cdot T^2 = L$, which is length, not velocity. A valid constant-acceleration form is $v = at$, assuming initial velocity is zero.
12. $E = mv$ is not dimensionally valid for energy. The right side has dimensions $[m][v] = M \cdot \dfrac{L}{T} = \dfrac{ML}{T}$, which is momentum, not energy. The kinetic energy form is $E = \dfrac{1}{2}mv^2$.

Applied Problems

13. **Current:** $I = \dfrac{V}{R} = \dfrac{12\ \text{V}}{330\ \Omega} = 0.03636\ \text{A} = 36.4\ \text{mA}.$

    Power: $P = VI = (12)(0.03636) = 0.436\ \text{W} = 436\ \text{mW}.$

14. This is a unit or dimensional error. Tension is a force, so the correct SI unit is newtons, N, not kilograms.

    If $850\ \text{kg}$ represents an equivalent mass, the corresponding weight force would be $W = mg = (850\ \text{kg})\left(9.81\dfrac{\text{m}}{\text{s}^2}\right) = 8338.5\ \text{N} \approx 8.34\ \text{kN}.$

15. The spreadsheet model computes $I_\text{model} = \dfrac{24\ \text{V}}{4.7\ \Omega} = 5.11\ \text{A}.$

    The correct resistance is $4.7\ \text{k}\Omega = 4700\ \Omega$, so the correct current is $I_\text{correct} = \dfrac{24\ \text{V}}{4700\ \Omega} = 0.00511\ \text{A} = 5.11\ \text{mA}.$

The model current is 1000 times too large.

16. $2.3 \times 10^{-2}\ \text{GPa} \times \dfrac{10^6\ \text{kPa}}{1\ \text{GPa}} = 2.3 \times 10^4\ \text{kPa} = 23{,}000\ \text{kPa}.$
17. **Mass:** $m = 0.75\ \text{m}^3 \times \dfrac{1000\ \text{kg}}{1\ \text{m}^3} = 750\ \text{kg}.$

    Weight in newtons: $W = 750\ \text{kg} \times 9.81 \dfrac{\text{m}}{\text{s}^2} = 7357.5\ \text{N}.$

    Weight in pounds-force: $7357.5\ \text{N} \times \dfrac{1\ \text{lb}_\text{f}}{4.448\ \text{N}} = 1654\ \text{lb}_\text{f}.$

This page titled 5.11: End-of-Chapter Problem Set is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

6.1: Why Structure Matters

6.1 Why Structure Matters

Here is how most students first approach engineering problems:

1. Read the question.
2. Find a formula that seems relevant.
3. Plug in numbers.
4. Write down the result.

That process occasionally produces correct answers on simple exercises. It almost never produces defensible engineering work.

Real engineering problems are embedded in physical context, constrained by real-world limits, and built on assumptions that must be stated and justified. A number that appears at the end of a calculation means very little unless you know what physical system it represents, what equation governs it, what assumptions made it tractable, and whether the result is physically reasonable.

Throughout this book, we use the following five-step workflow:

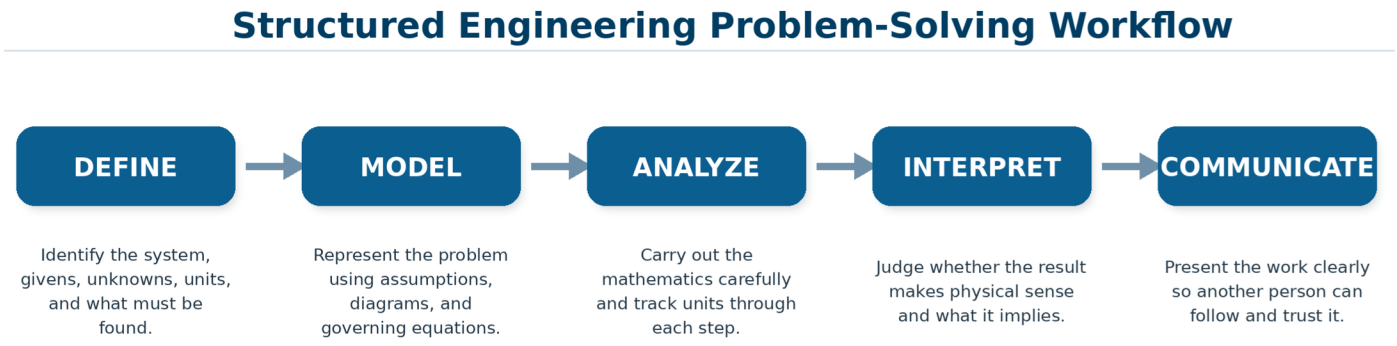


Figure 6.1.1: The engineering problem-solving workflow. Each step builds on the previous one; skipping any step increases error risk. (Copyright; Jonathan Compton, AI-assisted original figure, CC BY-NC 4.0)

This workflow is not a rigid script. It is a professional habit — a way of organizing thinking so that problems are solved correctly and solutions are transparent to others. You will use it in every technical chapter in this book.

This page titled 6.1: Why Structure Matters is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

6.2: Full Workflow Worked Example First

6.2 The Full Workflow — Worked Example First

Before dissecting each step, it helps to see the whole process in one place. Here is a complete five-step solution. This same problem will reappear in Chapter 8 (spreadsheet modeling) and Chapter 10 (MATLAB), where you will explore how the result changes across a range of angles.

✓ Worked Example 6.1 — Suspended Load, Full Workflow

Scenario: A 1000 kg mass is suspended from two identical cables. Each cable makes an angle of 30° with the horizontal. Determine the tension in each cable.

Step 1 — Define

Given values, assumptions, and unknowns for the suspended-load example.

Symbol	Quantity	Value	Units
m	Mass of suspended load	1000	kg
g	Gravitational acceleration	9.81	m/s²
θ	Cable angle from horizontal	30	°
T	Tension in each cable	*unknown*	N

Assumptions: static equilibrium; symmetric loading (both cables identical); massless cables.

Step 2 — Model

Vertical equilibrium: the two upward cable force components must balance the downward weight.

$$2T\sin\theta = mg$$

$$T = \frac{mg}{2\sin\theta}$$

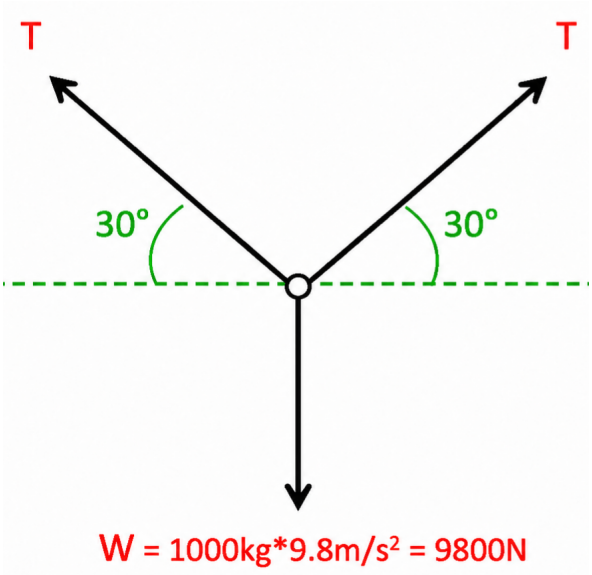


Figure 6.2.1: A free body diagram of a 1000 kg mass suspended from two identical cables. Each cable, T, makes an angle of 30° with the horizontal. (Copyright; Jonathan Compton, AI-assisted original figure, CC BY-NC 4.0).

Step 3 — Analyze

$$T = \frac{(1000\ \text{kg})(9.81\ \text{m/s}^2)}{2\sin(30°)} = \frac{9810\ \text{N}}{2 \times 0.500} = \frac{9810\ \text{N}}{1.000} = 9810\ \text{N}$$

Step 4 — Interpret

The tension is 9810 N ≈ **9.81 kN** — roughly equal to the full weight of the 1000 kg load. This makes sense: at 30°, each cable's force is mostly horizontal, so it must exert nearly twice the vertical force a straight-up cable would need. If the angle were 90°, each cable would carry exactly half the weight (4905 N). The shallower the angle, the higher the tension.

Design implication: cables must be rated for at least 9810 N plus a safety factor (typically 3–5× for structural applications).

Step 5 — Communicate

> *"Tension in each cable is **9810 N** (9.81 kN), assuming static equilibrium and symmetric loading at θ = 30°. This result is sensitive to cable angle — shallower angles increase tension significantly. Cables should be specified with a safety factor applied to this value."*

This page titled 6.2: Full Workflow Worked Example First is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

6.3: Step 1 — Define

6.3 Step 1 — Define the Problem Clearly

The Define step is the most commonly skipped — and the most important. A complete Define step includes:

- **Known quantities** with values and units
- **Unknown quantities** with symbols and expected units
- **Governing constraints** (equilibrium, conservation laws, circuit laws)
- **Explicit assumptions** — anything you are simplifying or ignoring
- A **free-body diagram or schematic** for physical problems

The diagram is underused by students and consistently present in professional engineering work. Drawing the physical situation before writing equations forces spatial thinking about the problem and catches sign errors, missing forces, and incorrect geometry before they propagate.

> ⚠ **Watch Out — "I'll just set up the equation"**
>
> "Setting up the equation" without a Define step is usually where unit errors and sign errors enter. If you cannot write down every known quantity with its unit and every unknown with its expected unit *before* touching an equation, you do not yet have a full mental model of the problem. The equation will not save you — it will amplify the gap.

Practice 6.1 — Write the Define Step Only

A 12 V battery is connected to a 330 Ω resistor. Determine the current and power. Write *only* the Define step — known quantities, unknowns with expected units, and two assumptions. No equations yet.

Compare these two approaches

Comparison of weak and complete Define steps for a simple circuit problem.

Weak define	Complete define
`V = 12, R = 330`	V = 12 V (source voltage), R = 330 Ω (resistance), I = ? A, P = ? W *Assumptions: ideal resistor (constant R), DC source, steady-state.*

The complete version assigns units at the point of definition. This takes 30 extra seconds and prevents the majority of common errors.

This page titled 6.3: Step 1 — Define is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

6.4: Step 2 — Model

6.4 Step 2 — Model the Physical System

Modeling is where physics becomes mathematics. You select the governing equation(s), write them symbolically, and arrange them to solve for the unknown. The key word is **select**: the art of modeling is choosing the right equation because you understand why it applies — not just what it looks like.

For the cable tension problem:

$$\sum F_y = 0 \quad \Rightarrow \quad 2T\sin\theta = mg \quad \Rightarrow \quad T = \frac{mg}{2\sin\theta}$$

For a circuit power problem:

$$I = \frac{V}{R} \qquad P = VI = \frac{V^2}{R}$$

In both cases, equations are written symbolically before numbers appear. This allows dimensional consistency checking (Chapter 5) before substitution and makes the structure of the problem visible.

✓ Worked Example 6.2 — Circuit Power, Full Workflow

Scenario

A 9 V battery is connected to a 2.2 kΩ resistor. Determine the current and power dissipated. Evaluate whether a standard 0.25 W resistor is safe.

Define

Given values and unknowns for the circuit power example.

Symbol	Quantity	Value	Units
V	Source voltage	9	V
R	Resistance	2.2 kΩ → 2200	Ω

Symbol	Quantity	Value	Units
I	Current	*unknown*	A
P	Power dissipated	*unknown*	W

Assumption: ideal resistor, DC source, steady-state operation.

Model

$$I = \frac{V}{R} \qquad P = VI$$

Analyze

$$I = \frac{9\ \text{V}}{2200\ \Omega} = 0.00409\ \text{A} = 4.09\ \text{mA}$$

$$P = (9\ \text{V})(0.00409\ \text{A}) = 0.0368\ \text{W} = 36.8\ \text{mW}$$

Interpret

$P = 36.8\ \text{mW}$ is well below the 0.25 W (250 mW) rating. The resistor is operating at approximately 15% of rated capacity — safe, with substantial margin.

Communicate

> *"With a 9 V source and 2.2 kΩ resistor, current is 4.09 mA and power dissipation is 36.8 mW. A 0.25 W resistor is safe (operating at ~15% of rating)."*

This page titled 6.4: Step 2 — Model is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

6.5: Step 3 — Analyze

6.5 Step 3 — Analyze Carefully

Analysis is where numbers enter. This is the step students rush, and the one where most arithmetic errors occur. Four habits that prevent them:

- **Convert before substituting.** Unit conversions belong in the Define step or at the top of Analyze — not buried mid-calculation.
- **Carry units through every step.** Every intermediate result needs its unit. If a unit disappears without canceling, something went wrong.
- **Write every intermediate step.** Mental arithmetic is unreliable under exam pressure. Write it out.
- **Do not round intermediate results.** Round only the final answer. Premature rounding accumulates error through multi-step calculations.

Practice 6.2 — Analysis Step

Using the Define and Model from Practice 6.1 (12 V battery, 330 Ω resistor), carry out the analysis step completely. Show unit cancellation for each calculation.

This page titled 6.5: Step 3 — Analyze is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

6.6: Step 4 — Interpret + Sensitivity

6.6 Step 4 — Interpret the Result

Interpretation is where engineering separates from mathematics. A student who computes $T = 9810$ and writes nothing else has done arithmetic. Interpretation asks three questions after every calculation:

1. **Is the unit correct?** If you expected newtons and got kilograms, find the error before proceeding.
2. **Is the magnitude physically reasonable?** A cable tension of 1.5×10^8 N for a 1000 kg mass is dimensionally possible but physically absurd.
3. **What does the result tell you about the system?** Is the design safe? Is an alternative worth exploring? Is the answer sensitive to a key assumption?

Sensitivity Reasoning — How Do Outputs Change When Inputs Vary?

One of the most useful interpretive skills is understanding how a result changes when an input changes — without recalculating numerically. Consider the cable tension equation:

$$T = \frac{mg}{2\sin\theta}$$

Ask: what happens to T if θ decreases?

- $\sin\theta$ decreases.
- $\sin\theta$ is in the **denominator**.
- Dividing by a smaller number gives a **larger** result.
- Therefore: tension **increases** as angle decreases.

But the relationship is not linear. Near $\theta = 90\,°$, $\sin\theta$ changes slowly ($\sin 90\,° = 1.000$, $\sin 80\,° = 0.985$). Near $\theta = 10\,°$, $\sin\theta$ changes rapidly ($\sin 15\,° = 0.259$, $\sin 10\,° = 0.174$). A 5° change near $\theta = 10\,°$ causes a much larger change in tension than the same 5° change near $\theta = 90\,°$.

This explains why engineers worry about shallow cable angles — small errors at low angles produce dramatically larger load changes than the same errors at steep angles. You do not need calculus to see this; just look at the equation structure and reason about it.

▶ **Try It — Sensitivity Reasoning**

For $T = \frac{mg}{2\sin\theta}$, answer without calculating:

1. If the mass doubles, what happens to T? Is the relationship linear?
2. If θ increases from 30° to 60°, does T increase or decrease? By approximately how much?
3. At what angle would the tension become theoretically infinite? What does that tell you about engineering constraints on cable installations?

Answer - click to expand

1. **If the mass doubles, T doubles.**

   The mass m is in the numerator of $T = \frac{mg}{2\sin\theta}$. If m is multiplied by 2, the entire numerator is multiplied by 2, so the tension is also multiplied by 2.

   The relationship between mass and tension is **linear** when g and θ are held constant.

2. **If θ increases from 30° to 60°, T decreases.**

   The angle appears inside $\sin\theta$ in the denominator. Since $\sin(30\,°) = 0.5$ and $\sin(60\,°) \approx 0.866$, the denominator gets larger when the angle increases from 30° to 60°.

   Because the denominator increases, the tension decreases. The new tension is approximately $0.5/0.866 \approx 0.577$ of the original tension. In other words, the tension at 60° is about **58% of the tension at 30°**, or about a **42% decrease**.

   For the 1000 kg load example, this would reduce the tension from about $9810\ \mathrm{N}$ to about $5660\ \mathrm{N}$.

3. **The tension becomes theoretically infinite when $\theta = 0\,°$.**

   At $\theta = 0\,°$, the cable is horizontal and $\sin(0\,°) = 0$. The equation becomes $T = \frac{mg}{0}$, which is undefined and represents an infinite theoretical tension.

   Physically, this means a perfectly horizontal cable cannot support a downward load using vertical force components, because the vertical component of the cable tension is zero. In real engineering design, this tells us that cable systems must avoid shallow angles and must specify minimum allowable cable angles, adequate anchor strength, and appropriate safety factors.

This page titled 6.6: Step 4 — Interpret + Sensitivity is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

6.7: Step 5 — Communicate

6.7 Step 5 — Communicate Clearly

Engineering communication has two properties that distinguish it from simply writing down a number:

It is complete. A complete engineering answer includes the numerical result, its unit, the key assumptions, and a statement of what the result means physically or from a design standpoint.

It is auditable. A professional engineering document must be checkable by someone who was not in the room. This means your Define step shows all variables, your Model shows the governing equation symbolically before substitution, your Analyze shows every intermediate calculation, and your Communicate explains the meaning.

Comparison of unstructured and structured engineering communication.

Unstructured	Structured
"9810"	"Tension in each cable is 9810 N (9.81 kN), assuming static equilibrium and symmetric loading at θ = 30°. The result would increase significantly at shallower angles — cables should be specified with a safety factor of at least 3× this value."

Only the structured version is defensible. Only the structured version can be checked by another engineer, used to specify the correct cable, or cited as the basis for a design decision. Chapter 13 (Engineering Communication) develops this further. For now, the standard to aim for: could another engineer reproduce your conclusion from your written work alone?

This page titled 6.7: Step 5 — Communicate is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

6.8: Five Common Failure Patterns

6.8 Five Common Failure Patterns

Most errors in engineering problem solving trace back to one of five patterns. Recognizing yours is the first step to fixing it.

Pattern 1 — Formula hunting. The student reads the problem, finds an equation that looks related, substitutes numbers, and records the result. Failure mode: the equation may not apply, may require unstated assumptions, or may be dimensionally correct but physically wrong. Fix: complete the Define step before selecting any equation.

Pattern 2 — Unit neglect. Unit conversions are skipped or units are not written alongside quantities. Failure mode: calculations produce wrong answers silently — the kΩ-as-Ω error from Chapter 5. Fix: units at every step, convert before substituting.

Pattern 3 — Skipping the model step. The student jumps from Define directly to numbers without writing the governing equation symbolically. Failure mode: cannot check dimensional consistency; cannot apply sensitivity reasoning. Fix: always write the symbolic equation before any substitution.

Pattern 4 — Stopping at the number. The student computes a numerical result and considers the problem done. Failure mode: the answer may be numerically correct but physically unreasonable or unsafe. Fix: always ask "what does this number mean?" before moving on.

Pattern 5 — Precision without accuracy. The spreadsheet or calculator returns eight decimal places and the student records all of them, equating precision with correctness. Failure mode: a wrong equation run to eight decimal places produces a wrong answer with eight decimal places of false confidence. Fix: precision is earned through correct modeling, not more significant figures.

This page titled 6.8: Five Common Failure Patterns is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

6.9: The Workflow Across Disciplines

6.9 The Workflow Across Disciplines

The five-step workflow transfers across every engineering domain. The physical system changes; the governing equations change; the structure does not.

How the five-step engineering workflow applies across different engineering domains.

Domain	Define includes	Model uses	Interpret checks
Structural (cable, beam)	Mass, geometry, angle	Equilibrium, FBD	Magnitude, safety factor
Circuit analysis	Voltage, resistance, current	Ohm's law, Kirchhoff's	Power rating, margin
Spreadsheet model	Parameter table	Algebraic formula in cells	Graph shape, sensitivity
CAD/geometry	Dimensions, material	Volume / mass formulas	Physical plausibility

> **🔗 Connection**
>
> Every quantitative chapter in this book uses the **Define → Model → Analyze → Interpret → Communicate** format for worked examples. When you see a Worked Example callout in Chapters 8, 10, 11, or 12, the five steps will be labeled explicitly. Developing fluency now means the technical content in those chapters can be absorbed more quickly — the structure will be familiar, and the only new thing will be the domain.

This page titled 6.9: The Workflow Across Disciplines is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

6.10: Structure and Professional Accountability

6.10 Structure and Professional Accountability

When an engineer's calculation is reviewed — by a colleague, a client, a regulator, or an accident investigation board — what they are checking is not just whether the final answer is correct. They are checking whether the reasoning is transparent, the assumptions are stated, the units are consistent, and the result was evaluated for physical reasonableness.

A calculation that jumps from problem statement to number, with no visible Define or Model step, cannot be audited. It cannot be trusted. It cannot be signed and stamped by a licensed engineer.

Structure is not bureaucratic overhead. It is the mechanism by which engineering maintains accountability to the public it serves.

> **Key Takeaway**
>
> The five-step workflow — Define, Model, Analyze, Interpret, Communicate — is the core professional habit of engineering problem solving. It prevents the most common errors, makes calculations auditable, and separates engineering work from arithmetic. You will use it in every technical chapter in this book. Mastering it here means the structure becomes automatic, freeing your attention for the physical content of each new domain.

This page titled 6.10: Structure and Professional Accountability is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

6.11: Summary

Summary

- The five-step engineering workflow — **Define** → **Model** → **Analyze** → **Interpret** → **Communicate** — is the professional method for solving quantitative engineering problems.
- Define: all known and unknown quantities with units, governing constraints, and explicit assumptions. Draw a diagram.
- Model: write the governing equation symbolically before any substitution. Check dimensional consistency.
- Analyze: convert units before substituting; carry units through every step; write every intermediate result; round only the final answer.
- Interpret: verify the unit, evaluate physical reasonableness, and extract engineering meaning including safety implications.
- Communicate: state the number, unit, key assumptions, and what the result means. Make it auditable.
- Five failure patterns — formula hunting, unit neglect, skipping the model, stopping at the number, precision without accuracy — are all prevented by the five-step structure.
- The workflow transfers across structural, circuit, computational, and CAD problems and appears in every worked example in this book.

This page titled 6.11: Summary is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

6.12: End-of-Chapter Problem Set

End-of-Chapter Problem Set

Apply the complete five-step workflow to each problem. Show the Define table, symbolic Model equation, Analyze steps with units, Interpret paragraph, and Communicate sentence.

1. A 750 kg load is supported symmetrically by two cables at 25° from horizontal. Compute tension in each cable.
2. A 15 V source is connected to a 470 Ω resistor. Compute current and power. Is a standard 0.25 W resistor safe?
3. A 12 V source is connected to a 1.2 kΩ resistor. Compute current in mA and power in mW.
4. For $T = \frac{mg}{2\sin\theta}$, determine what happens as $\theta \to 0$ °. Explain the physical implication for cable installation.
5. A resistor rated at 0.5 W is connected to a 9 V battery. What is the minimum safe resistance? Show the full workflow.
6. If the cable angle in Problem 1 decreases from 25° to 15°, does tension increase by roughly 50% or more than 100%? Compute both values and explain the nonlinearity.
7. A 24 V source is connected to a 4.7 kΩ resistor. Compute current and power. Interpret safety if the resistor rating is 0.125 W.
8. *[Challenge]* A bracket supports a hanging sign weighing 200 N via a horizontal strut and a cable at 45°. Using equilibrium $\sum F_x = 0$, $\sum F_y = 0$, find the tension in the cable and the compressive force in the strut. Apply the full workflow.

Answers - click to expand

1. **750 kg load supported by two cables at 25°**

   Model: For symmetric loading, the vertical components of the two cable tensions support the weight: $2T\sin\theta = mg$. Solving for tension gives $T = \frac{mg}{2\sin\theta}$.

   Analyze: $T = \frac{(750\ \text{kg})(9.81\ \text{m/s}^2)}{2\sin(25°)} = \frac{7357.5\ \text{N}}{0.845} \approx 8705\ \text{N}.$

   Interpret: Each cable carries about $8.71\ \text{kN}$, which is greater than half the load weight because each cable is angled. Only the vertical components support the load.

   Communicate: The required tension is approximately $8.71\ \text{kN}$ **per cable**, before applying any safety factor.

2. **15 V source with 470 Ω resistor**

Model: Ohm's Law gives $I = \frac{V}{R}$. Resistor power can be found from $P = VI$ or $P = \frac{V^2}{R}$.

Analyze: $I = \frac{15\ \text{V}}{470\ \Omega} = 0.0319\ \text{A} = 31.9\ \text{mA}$. Power is $P = \frac{(15\ \text{V})^2}{470\ \Omega} = 0.479\ \text{W}$.

Interpret: The resistor dissipates about $0.479\ \text{W}$, which is greater than $0.25\ \text{W}$. A standard quarter-watt resistor is not safe for this condition.

Communicate: The current is approximately $31.9\ \text{mA}$ and the power is approximately $0.479\ \text{W}$; use a resistor rated above this value, such as $0.5\ \text{W}$ or higher with appropriate margin.

3. **12 V source with 1.2 kΩ resistor**

   Model: Use $I = \frac{V}{R}$ and $P = VI$.

   Analyze: $I = \frac{12\ \text{V}}{1200\ \Omega} = 0.010\ \text{A} = 10\ \text{mA}$. Power is $P = (12\ \text{V})(0.010\ \text{A}) = 0.120\ \text{W} = 120\ \text{mW}$.

   Interpret: The current is modest and the resistor dissipates $120\ \text{mW}$. This would be below a $0.25\ \text{W}$ rating, though margin should still be considered.

   Communicate: The circuit current is $10\ \text{mA}$ and the resistor power is $120\ \text{mW}$.

4. **Behavior of $T = \frac{mg}{2\sin\theta}$ as $\theta \to 0\ °$**

   Model: The cable tension is $T = \frac{mg}{2\sin\theta}$.

   Analyze: As $\theta \to 0\ °$, $\sin\theta \to 0$. Therefore, the denominator $2\sin\theta$ approaches zero, so T grows without bound.

   Interpret: A perfectly horizontal cable cannot support a vertical load through vertical force components because the vertical component of tension becomes zero. The equation predicts theoretically infinite tension, which signals an impossible or unsafe physical design.

   Communicate: Cable systems must avoid very shallow angles; designers should specify minimum cable angles, adequate anchor strength, and appropriate safety factors.

5. **Minimum safe resistance for a 0.5 W resistor on a 9 V battery**

   Model: Use $P = \frac{V^2}{R}$. To keep power at or below the rating, solve for resistance: $R_{\text{min}} = \frac{V^2}{P_{\text{max}}}$.

   Analyze: $R_{\text{min}} = \frac{(9\ \text{V})^2}{0.5\ \text{W}} = \frac{81}{0.5} = 162\ \Omega$.

   Interpret: A resistor smaller than $162\ \Omega$ would dissipate more than $0.5\ \text{W}$ and exceed the rating. In practice, a designer would choose a standard value above this minimum and include margin.

   Communicate: The minimum safe resistance is $162\ \Omega$; a practical design should use a higher standard

resistance or a higher power rating.

6. **Cable angle decreases from 25° to 15°**

   Model: Use $T = \dfrac{mg}{2\sin\theta}$.

   Analyze: At $25\ °$, $T_{25} = \dfrac{7357.5\ \text{N}}{2\sin(25°)} \approx 8705\ \text{N}$. At $15\ °$, $T_{15} = \dfrac{7357.5\ \text{N}}{2\sin(15°)} \approx 14215\ \text{N}$.

   The percent increase is $\dfrac{14215 - 8705}{8705} \times 100\% \approx 63\%$.

   Interpret: The increase is closer to 50% than to more than 100%, but it is still a large increase. The relationship is nonlinear because $\sin\theta$ is in the denominator. As the angle becomes shallow, small changes in angle cause increasingly large changes in tension.

   Communicate: Reducing the cable angle from $25\ °$ to $15\ °$ increases the tension from about $8.71\ \text{kN}$ to about $14.2\ \text{kN}$, an increase of about **63%**.

7. **24 V source with 4.7 kΩ resistor**

   Model: Use $I = \dfrac{V}{R}$ and $P = \dfrac{V^2}{R}$.

   Analyze: $I = \dfrac{24\ \text{V}}{4700\ \Omega} = 0.00511\ \text{A} = 5.11\ \text{mA}$. Power is $P = \dfrac{(24\ \text{V})^2}{4700\ \Omega} = 0.1226\ \text{W}$.

   Interpret: The calculated power is slightly below $0.125\ \text{W}$, so it is technically under the rating. However, the margin is extremely small. Component tolerance, supply variation, temperature, and continuous operation could push the resistor beyond its safe rating.

   Communicate: The current is approximately $5.11\ \text{mA}$ and the power is approximately $0.123\ \text{W}$; a $0.125\ \text{W}$ resistor is too close to the limit for a robust design.

8. **Challenge: bracket with 200 N hanging sign, horizontal strut, and 45° cable**

   Model: At the joint, use equilibrium. Vertical equilibrium gives $\sum F_y = 0$, so $T\sin(45\ °) - 200\ \text{N} = 0$. Horizontal equilibrium gives $\sum F_x = 0$, so $F_s - T\cos(45\ °) = 0$, assuming the strut provides the opposing horizontal force.

   Analyze: From vertical equilibrium, $T = \dfrac{200\ \text{N}}{\sin(45°)} = \dfrac{200\ \text{N}}{0.707} \approx 283\ \text{N}$. From horizontal equilibrium, $F_s = T\cos(45\ °) = (283\ \text{N})(0.707) \approx 200\ \text{N}$.

   Interpret: The cable tension is larger than the sign weight because only the vertical component of the cable tension supports the sign. The horizontal component of the cable tension must be balanced by the strut, so the strut carries a compressive force of about $200\ \text{N}$.

   Communicate: The cable tension is approximately $283\ \text{N}$, and the horizontal strut carries approximately $200\ \text{N}$ in compression.

This page titled 6.12: End-of-Chapter Problem Set is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

7: THE ENGINEERING DESIGN PROCESS

The Engineering Design Process

Design begins with incomplete information and ends with a justified decision. It is not analysis with extra steps — it is a fundamentally different kind of engineering work.

Learning Objectives

By the end of this chapter, you will be able to:

- Distinguish between analysis problems and design problems.
- Apply a seven-step engineering design framework from objective definition through communication.
- Write a strong design objective that includes function, performance target, and evaluation metric.
- Identify and classify hard versus soft constraints in a design problem.
- Generate multiple design alternatives and evaluate them using a decision matrix.
- Explain why iteration is a feature of design, not a symptom of failure.
- Connect design decisions to safety factors and ethical responsibility.

7.1: Analysis vs Design
7.2: What Makes a Design Problem
7.3: Seven-Step Design Framework
7.4: Define the Design Objective
7.5: Identify Constraints
7.6: Generate Alternatives
7.7: Model and Analyze
7.8: Evaluate Trade-offs
7.9: Iterate and Refine
7.10: Communicate
7.11: Design, Ethics, and Safety Factors
7.12: Summary
7.13: Quick Check

This page titled 7: The Engineering Design Process is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

7.1: Analysis vs Design

7.1 Analysis vs. Design — Two Different Kinds of Work

Chapter 6 built your problem-solving workflow for **analysis** problems — problems where you are given a system and asked to compute something about it. Analysis tells you what a system *will do* under specific conditions.

Most real engineering work is **design**: deciding what a system *should be* so that it accomplishes a goal safely, affordably, and reliably.

Comparison of analysis and design work in engineering.

	Analysis	Design
Starting point	A defined system	A need or requirement
The unknown	A calculated quantity	A configuration, geometry, or set of values
Answer type	A specific number	A defensible solution (often one of several valid ones)
Example	Compute tension in a cable at 30°	Choose cable angle and cross-section so tension stays below 12 kN with SF ≥ 3

A design problem is recognizable by at least one of these features: you must choose variables (not just compute them); constraints exist that cannot be violated; trade-offs exist; multiple valid solutions exist.

⚠ Real-World Example — When Design Process Fails

Tacoma Narrows Bridge (1940): The bridge was optimized for aesthetics and material cost, with an elegant slender deck. The design process failed to adequately analyze the aerodynamic behavior of that deck profile in wind. The bridge oscillated into resonance and collapsed four months after opening. Wind analysis was an afterthought, not a constraint.

Hyatt Regency walkway collapse (1981): A connection detail was changed during construction without re-analysis of the load path. The change seemed equivalent — it was not. 114 people were killed. A proper design process requires that any design change be re-analyzed from first principles.

Both failures trace to the same lesson: design process shortcuts have consequences. The rigor you apply to defining objectives, checking constraints, and iterating on safety factors is the mechanism that prevents these outcomes.

This page titled 7.1: Analysis vs Design is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

7.2: What Makes a Design Problem

7.2 What Makes a Design Problem

Examples distinguishing analysis tasks from design tasks.

Task	Type	Why
Compute the current through a 330 Ω resistor at 9 V	**Analysis**	System fully defined; one answer exists
Choose a resistor so the circuit stays below 0.25 W	**Design**	You choose the variable; constraints and criteria apply
Compute cable tension at 30° for a 1000 kg load	**Analysis**	System fully defined; one answer exists
Choose cable angle and cross-section for a 5000 N load with SF ≥ 3	**Design**	You choose variables; multiple valid answers; trade-offs exist

A useful test: if the problem has a unique numerical answer, it is probably analysis. If you need to *select* a value, *choose* from alternatives, or *justify* a decision against competing criteria — it is design.

This page titled 7.2: What Makes a Design Problem is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

7.3: Seven-Step Design Framework

7.3 A Practical Engineering Design Framework

For this course, we use a seven-step design framework:

> **1. Define Objective → 2. Identify Constraints → 3. Generate Alternatives → 4. Model & Analyze → 5. Evaluate Trade-offs → 6. Iterate & Refine → 7. Communicate**

This is not a straight line. Engineers loop. A constraint in Step 2 may force you back to redefine the objective in Step 1. An analysis failure in Step 4 pushes you back to Step 3. A trade-off comparison in Step 5 may reveal that your evaluation metric in Step 1 needs adjustment.

This looping is not a design flaw — it is how design is supposed to work. Iteration is the mechanism by which designs improve.

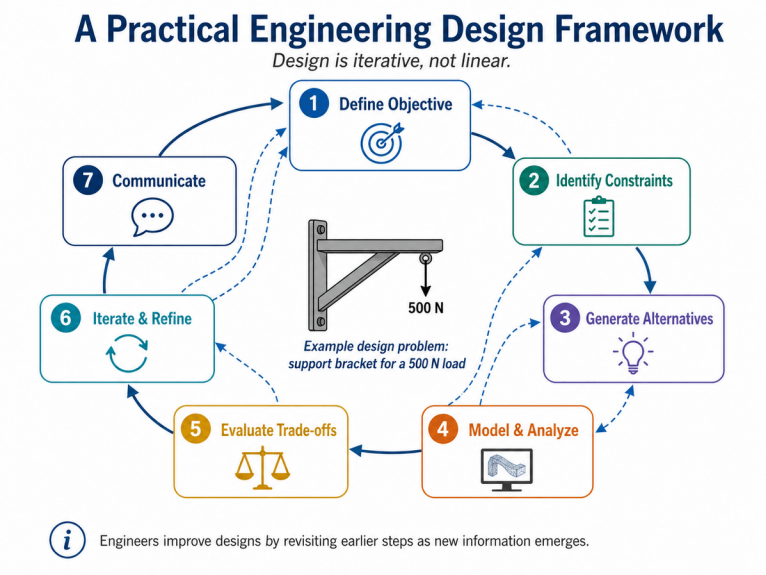


Figure 7.3.1: A practical engineering design framework showing seven major steps: define the objective, identify constraints, generate alternatives, model and analyze, evaluate trade-offs, iterate and refine, and communicate. The feedback arrows emphasize that engineering design is iterative rather than linear. (Copyright; Jonathan Compton, AI-assisted original figure, CC BY-NC 4.0)

Chapter thread: To make each step concrete, this chapter works through a single design problem from start to finish — designing a support bracket for a 500 N load. This is a design *thinking* exercise, not a full structural analysis project.

This page titled 7.3: Seven-Step Design Framework is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

7.4: Define the Design Objective

7.4 Step 1 — Define the Design Objective

Vague objectives produce weak designs. "Design a bridge" is not an objective — it is a topic. A strong design objective has four components:

Components of a strong design objective.

Component	What it specifies	Example (bracket thread)
Function	What the design must *do*	Support a vertical load
Performance target	How well it must do it (quantitative)	Support 500 N
Context	Operating conditions, material	Static loading, steel
Evaluation metric	What "best" means among acceptable solutions	Minimize mass

Combining these: *"Design a steel bracket that supports a 500 N vertical load with safety factor ≥ 2.0, while minimizing mass."*

Without the evaluation metric, you cannot compare two designs — you have no basis for saying one is better than another. "Both are safe" is not enough; an engineer should be able to say which is preferred and why.

Practice 7.1 — Strengthen a Design Objective

Turn this into a strong design objective: *"Design a circuit that powers an LED."* Include: supply voltage, maximum LED power, what you are optimizing, and at least one hard constraint.

This page titled 7.4: Define the Design Objective is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

7.5: Identify Constraints

7.5 Step 2 — Identify Constraints and Requirements

Constraints define the space within which acceptable designs live. If you ignore them, you are not doing engineering.

Hard and soft constraints in engineering design.

Type	Definition	Examples
Hard constraint	Must never be violated	SF ≥ 2; power rating not exceeded; bridge must span 50 cm
Soft constraint	A preference — violation may be acceptable with justification	Minimize mass; prefer standard component values; stay under cost target

Hard constraints come first. Every design candidate must satisfy all hard constraints before it is even considered for evaluation.

For the bracket thread, the constraints yield a design space. Given $F = 500\ \text{N}$, $\sigma_{\text{allow}} = 80\ \text{MPa}$, SF ≥ 2:

$$\sigma_{\text{work}} = \frac{\sigma_{\text{allow}}}{\text{SF}} = \frac{80\ \text{MPa}}{2} = 40\ \text{MPa}$$

$$A_{\text{min}} = \frac{F}{\sigma_{\text{work}}} = \frac{500\ \text{N}}{40 \times 10^{6}\ \text{Pa}} = 1.25 \times 10^{-5}\ \text{m}^2 = 12.5\ \text{mm}^2$$

Any bracket with cross-sectional area below 12.5 mm² violates the hard constraint and is eliminated before any other evaluation.

✓ Worked Example 7.1 — Constraint Calculation

Problem: A rod supports 2000 N. Allowable stress = 120 MPa. Required SF = 3. Find minimum area.

$$\sigma_{\text{work}} = \frac{120\ \text{MPa}}{3} = 40\ \text{MPa}$$

$$A_{\text{min}} = \frac{2000\ \text{N}}{40 \times 10^{6}\ \text{Pa}} = 5.0 \times 10^{-5}\ \text{m}^2 = 50\ \text{mm}^2$$

Interpretation: Any cross-section below 50 mm² is automatically eliminated. A 5 mm × 10 mm section (50 mm²) exactly meets the minimum; a 6 mm × 10 mm section (60 mm²) provides additional margin.

Practice 7.2 — Write a Constraint Statement

A resistor is rated at 0.25 W. Write a constraint statement using $P = V^2/R$ that specifies the minimum resistance for a 9 V source.

This page titled 7.5: Identify Constraints is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

7.6: Generate Alternatives

7.6 Step 3 — Generate Alternatives (Not Just One Idea)

The most common novice design mistake is selecting the first idea that seems to work and "making it work." Professional engineers generate multiple alternatives because the first idea is rarely optimal, alternative designs reveal trade-offs, and creativity improves when you know the constraints.

A practical rule: **generate at least three distinct alternatives** before moving to analysis. "Distinct" means meaningfully different — not three versions of the same shape with slightly different dimensions.

For the bracket thread, three alternatives:

Design alternatives for the bracket example.

Alternative	Cross-section	Area (mm²)	Description
A	Solid square rod, 5 × 5 mm	25	Simple, easy to manufacture
B	Hollow rect. tube, 6 × 6 mm, 1 mm wall	20	Less material, slightly more complex
C	L-bracket with gusset	~18	More complex; better stiffness profile

All three exceed the 12.5 mm² minimum. Now analysis will quantify which performs best against the mass-minimization objective.

This page titled 7.6: Generate Alternatives is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

7.7: Model and Analyze

7.7 Step 4 — Model and Analyze Alternatives

This is where Chapter 6's structured workflow becomes a design tool. You apply **Define** → **Model** → **Analyze** → **Interpret** to each alternative, quantifying the behavior you care about.

For the bracket, the two key quantities are stress (constraint check) and mass (optimization target):

$$\sigma = \frac{F}{A} \qquad m = \rho \cdot A \cdot L$$

For a fixed bracket length, mass is proportional to area — so minimum mass corresponds to the smallest area that still meets the stress constraint.

Stress and safety factor analysis for bracket design alternatives.

Alternative	Area (mm²)	Stress (MPa)	Actual SF	Passes SF ≥ 2?
A	25	20	4.0	**Yes**
B	20	25	3.2	**Yes**
C	18	27.8	2.88	**Yes**

All three pass the hard constraint. Now trade-off evaluation determines which is preferred.

This page titled 7.7: Model and Analyze is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

7.8: Evaluate Trade-offs

7.8 Step 5 — Evaluate Trade-offs

Trade-offs are the core of design. You rarely improve everything simultaneously: thicker bracket → more strength but more mass; lower resistance → more current but more power; shallower cable angle → lower structure but higher tension.

A **decision matrix** is the standard engineering tool for comparing alternatives against multiple criteria. It makes evaluation explicit, transparent, and reproducible — rather than based on gut feeling.

Building a Decision Matrix

Steps: (1) List the criteria that matter. (2) Assign weights that sum to 100%. (3) Score each alternative on each criterion (1 = worst, 5 = best). (4) Compute weighted total = Σ(weight × score). (5) The highest weighted total wins — subject to re-examining the weights.

For the bracket:

Weighted decision matrix for bracket design alternatives.

Criterion	Weight	Alt A	Alt B	Alt C
Low mass	40%	2 → 0.80	4 → 1.60	5 → 2.00
Manufacturing simplicity	30%	5 → 1.50	3 → 0.90	2 → 0.60
Safety margin	30%	5 → 1.50	4 → 1.20	3 → 0.90
Weighted Total	**100%**	**3.80**	**3.70**	**3.50**

With these weights, Alternative A wins — but notice: if mass reduction were weighted 60% instead of 40%, Alternative B or C would win. The decision matrix makes this sensitivity visible. That is why the weights must be set deliberately, not arbitrarily.

> **⚠ Watch Out — "The safe choice" is not automatically the right choice**
>
> A common student instinct is to pick the design with the highest safety factor, reasoning that "more safe is always better." But over-engineering has real costs: mass, material, cost, size. In aerospace, unnecessary mass is itself a safety problem — it reduces fuel margin and payload capacity. The design goal is to meet the safety requirement with an appropriate margin, not to maximize it.

This page titled 7.8: Evaluate Trade-offs is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

7.9: Iterate and Refine

7.9 Step 6 — Iterate and Refine

Iteration is not redoing work because you made a mistake. It is a feature of design: refining toward the constraint rather than stopping at the first acceptable solution.

Iteration typically occurs because: (1) the first design violates a hard constraint; (2) the design is safe but inefficient — actual stress far below working limit means excess material; (3) a better alternative is discovered during analysis.

Example iteration sequence for the bracket (starting from Alternative A, 25 mm²):

Iteration sequence for refining the bracket design toward the safety constraint.

Iteration	Area (mm²)	Stress (MPa)	SF	Action
Start	25	20	4.0	Overdesigned — reduce area
Iter 1	16	31.25	2.56	Better — still has margin, continue
Iter 2	13	38.5	2.08	Close to constraint — try smaller
Iter 3	12	41.7	1.92	**Violates SF ≥ 2 — step back**
Final	**13**	**38.5**	**2.08**	**Minimum mass while meeting constraint**

Students often stop when a design "works." Engineers continue until the design is both safe and efficient. A bracket with SF = 4.0 where SF = 2.0 is required used twice as much material as necessary.

This page titled 7.9: Iterate and Refine is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

7.10: Communicate

7.10 Step 7 — Communicate the Design

A complete design communication includes: objective, constraints, alternatives considered, analysis supporting each, decision basis, final recommendation with justification, and limitations.

✓ Worked Example 7.2 — Short Design Recommendation

> *"A solid square rod cross-section of 13 mm² (approximately 3.6 mm × 3.6 mm) is recommended. This selection satisfies the 500 N load requirement with a safety factor of 2.08 — above the minimum of 2.0 — while minimizing material use. Manufacturing simplicity was also a factor: solid square rod stock is widely available and requires no special tooling.*
>
> *Three alternatives were evaluated; the hollow tube offered slightly lower mass but lower safety margin; the L-bracket was eliminated due to manufacturing cost.*
>
> *Limitation: this analysis treats the load as purely axial. Real bracket geometry introduces bending, and a more detailed analysis would be required before committing to final dimensions."*

🔗 Connection — Bridge Challenge Project

The Bridge Challenge asks you to design a bridge using plastic straws and tape that spans 50 cm, stays under 10 cm wide and 80 grams, and supports as much load as possible. Every step of this chapter applies directly:

- **Step 1 (Objective):** Maximize load capacity while satisfying span, width, and mass constraints.
- **Step 2 (Constraints):** 50 cm span, ≤10 cm width, ≤80 g, ≥10 lbs supported — all hard constraints. Minimizing straws is a soft constraint.
- **Step 3 (Alternatives):** What bridge types will you consider — beam, truss, arch? Generate at least three configurations before building.
- **Step 5 (Trade-offs):** More straws = more strength, but more mass. Where is the right balance?
- **Step 6 (Iterate):** Your first design will not be optimal. Prototype, test, adjust.
- **Step 7 (Communicate):** The project report asks you to explain your design decisions, how the bridge behaved under load, and what you would change.

This page titled 7.10: Communicate is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

7.11: Design, Ethics, and Safety Factors

7.11 Design, Ethics, and Safety Factors

Design decisions carry ethical weight. The failures in §7.1 were not acts of malice — they were the cumulative result of individually defensible choices that were collectively insufficient. Aerodynamics was not in scope for Tacoma Narrows. A connection change seemed equivalent. Neither team was trying to build a dangerous structure.

Ethical engineering design means:

- **State your assumptions explicitly.** Every model simplifies. That simplification is a decision and must be visible to anyone reviewing the design.
- **Apply safety factors appropriate to consequence.** A prototype for internal testing can tolerate lower margins than a public-facing structure — but that decision must be deliberate and documented.
- **Never compromise a hard constraint for convenience or cost.** Safety factors are not design variables to negotiate away under pressure.
- **Document limitations honestly.** "This analysis does not capture bending" is not a weakness — it is professional integrity.

⚠ Ethics Check

You are designing a bracket for a mezzanine floor. Your analysis shows it meets the required safety factor of 2.0 exactly. Your manager asks whether you could use a slightly thinner cross-section to reduce material cost by 15% — which would put the safety factor at 1.85.

What factors should go into your decision? What would a responsible engineer do? What would you document?

Key Takeaway

Design is not analysis with extra steps. It begins with incomplete information and ends with a justified decision — not a computed answer. The seven-step framework (Define Objective → Identify Constraints → Generate Alternatives → Model & Analyze → Evaluate Trade-offs → Iterate → Communicate) applies across all engineering disciplines. The decision matrix makes trade-off evaluation explicit. Iteration is how designs improve. Every design decision carries ethical weight — safety factors and stated assumptions are not negotiable under pressure.

This page titled 7.11: Design, Ethics, and Safety Factors is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

7.12: Summary

Summary

- Design answers "what should we build?" while analysis answers "what happens if we build this?" Design involves choosing variables, satisfying constraints, and navigating trade-offs.
- A strong design objective includes function, quantitative performance target, context, and evaluation metric.
- Hard constraints must never be violated; soft constraints are preferences that can be traded against other goals.
- Generate at least three distinct alternatives before selecting and analyzing.
- A decision matrix makes trade-off evaluation explicit: weight criteria, score alternatives, compute weighted totals.
- Iterate toward the constraint — stop when safe and efficient, not just when safe.
- Communicate with objective, constraints, alternatives, analysis, decision basis, and limitations.
- Safety factors and stated assumptions are not negotiable under schedule or cost pressure.

This page titled 7.12: Summary is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

7.13: Quick Check

Quick Check

1. Classify each as analysis or design and explain why: (a) Compute current through 470 Ω at 12 V. (b) Choose a resistor so a 9 V circuit stays below 0.25 W. (c) Compute cable tension at 45° for 2000 kg. (d) Select cable angle and cross-section for 3000 N load with SF ≥ 2.5.
2. Write a strong design objective for an LED circuit at 9 V with 0.06 W maximum LED power. Include the evaluation metric.
3. A rod supports 8000 N. Allowable stress = 160 MPa. Required SF = 2.5. Find minimum area in mm².
4. Using $P = V^2/R$, find the minimum resistance so power stays below 0.25 W for a 12 V source.
5. Given two alternatives: A has 20 mm² (SF = 3.2, simple to manufacture) and B has 15 mm² (SF = 2.4, harder to manufacture). Weight: mass minimization 50%, manufacturing simplicity 25%, safety margin 25%. Build a decision matrix and determine which alternative wins.
6. Explain why iteration is a feature of design rather than a sign of failure. Give one example of a productive iteration cycle.
7. A bracket analysis gives actual stress of 8 MPa against a working limit of 40 MPa. What does this suggest, and what action would you take?
8. *[Bridge Challenge prep]* List the hard constraints for the Bridge Challenge. For each, explain what design choices it rules out immediately.

Answers - click to expand

1. **Classify each as analysis or design.**
   a. **Analysis:** Computing current through a known $470\ \Omega$ resistor at $12\ \text{V}$ uses known inputs to find an output. The component has already been selected.
   b. **Design:** Choosing a resistor so a $9\ \text{V}$ circuit stays below $0.25\ \text{W}$ requires selecting a component that satisfies a constraint.
   c. **Analysis:** Computing cable tension at $45\ °$ for a $2000\ \text{kg}$ load uses a known load and geometry to calculate the resulting tension.
   d. **Design:** Selecting cable angle and cross-section for a $3000\ \text{N}$ load with $SF \geq 2.5$ requires choosing design variables that satisfy strength and safety constraints.
2. **Strong design objective:** Design a resistor-protected LED circuit powered by a $9\ \text{V}$ source so that the LED power remains at or below $0.06\ \text{W}$ during normal operation while still producing acceptable brightness.

   Evaluation metric: A suitable metric is LED power margin, defined by how far the calculated LED power is below the $0.06\ \text{W}$ limit. A second useful metric could be circuit current or brightness if LED specifications are provided.
3. **Minimum rod area:**

   Since $1\ \text{MPa} = 1\ \text{N/mm}^2$, the minimum area is $A = \dfrac{F}{\sigma_{\text{design}}} = \dfrac{8000\ \text{N}}{64\ \text{N/mm}^2} = 125\ \text{mm}^2$.

Answer The rod needs a minimum cross-sectional area of $125\ \text{mm}^2$.

4. **Minimum resistance for $P \leq 0.25\ \text{W}$:**

   For $V = 12\ \text{V}$ and $P_{\max} = 0.25\ \text{W}$, $R_{\min} = \dfrac{(12\ \text{V})^2}{0.25\ \text{W}} = \dfrac{144}{0.25} = 576\ \Omega$.

   Answer The resistance must be at least $576\ \Omega$. In practice, choose the next higher standard resistor value to provide margin.

5. **Decision matrix:** One reasonable scoring method is to score each category from 1 to 5, where 5 is best. Since lower mass is better, the smaller area receives the higher mass score. Since higher safety factor is better, the larger safety factor receives the higher safety score.

Alternative	Mass score (50%)	Manufacturing score (25%)	Safety score (25%)	Weighted total
A: $20\ \text{mm}^2$, $SF = 3.2$, simple	3	5	5	$3(0.50) + 5(0.25) + 5(0.25) = 4.00$
B: $15\ \text{mm}^2$, $SF = 2.4$, harder	5	2	2	$5(0.50) + 2(0.25) + 2(0.25) = 3.50$

   Answer Alternative A wins with a weighted score of 4.00 compared with 3.50 for Alternative B. More importantly, B has $SF = 2.4$, which may fail if the design requires $SF \geq 2.5$. If $SF \geq 2.5$ is a hard constraint, B should be rejected before scoring.

6. **Iteration is a feature of design because design problems have constraints, trade-offs, and incomplete information.** A first design rarely satisfies every requirement perfectly. Iteration allows engineers to test an idea, identify weaknesses, revise the design, and improve performance.

   Example: A bridge prototype may be strong enough but too heavy. The team can revise the truss layout, remove unnecessary members, retest the load capacity, and compare the new strength-to-weight ratio. That loop is not failure; it is how the design improves.

7. **Bracket stress interpretation:** If the actual stress is $8\ \text{MPa}$ and the working limit is $40\ \text{MPa}$, then the bracket is using only $8/40 = 0.20$, or 20%, of the allowed stress.

   This suggests the bracket may be overdesigned for strength. That is not automatically bad, but it may mean the design is heavier, more expensive, or larger than necessary.

   Action: I would check whether other constraints control the design, such as deflection, buckling, fatigue, impact loading, manufacturability, or required safety factor. If strength is the only concern, I would iterate by reducing material or changing geometry, then rerun the analysis.

8. **Bridge Challenge prep:** Answers will depend on the exact rules provided for the course version of the Bridge Challenge. A strong answer should identify hard constraints and explain what each one rules out.

   Common hard constraints might include:

   - **Span requirement:** The bridge must cross a specified gap, so designs that are too short are immediately ruled out.
   - **Load requirement:** The bridge must support a required load, so designs that cannot carry the test load are ruled out even if they are lightweight.
   - **Material limit:** Only approved materials may be used, so metal, glue types, fasteners, or reinforcement

methods outside the rules are not allowed.
- **Mass or material quantity limit:** The bridge cannot exceed a maximum weight or material amount, so simply adding more material everywhere is not an acceptable strategy.
- **Dimensional envelope:** The bridge must fit within height, width, or clearance limits, so overly tall, wide, or obstructive designs are ruled out.
- **Connection rules:** If joint types or adhesive locations are restricted, designs that depend on prohibited connections must be revised.
- **Testing interface:** The bridge must accept the loading fixture, so designs that leave no place to apply the load are ruled out.

Answer Hard constraints define the boundary of acceptable designs. Any design that violates a hard constraint should be rejected before comparing performance metrics such as strength-to-weight ratio, cost, or ease of construction.

This page titled 7.13: Quick Check is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

8: SPREADSHEET MODELING

Spreadsheet Modeling

A calculator gives you one answer. A spreadsheet lets you understand behavior across a range — which is what engineering design actually asks.

Learning Objectives

By the end of this chapter, you will be able to:

- Describe why spreadsheets are engineering modeling tools, not just calculators.
- Structure a spreadsheet into inputs, model equations, and outputs with proper labeling.
- Use absolute and relative cell references correctly in parameter sweeps.
- Convert angles from degrees to radians within formulas.
- Perform a parameter sweep and generate an XY scatter plot with fully labeled axes.
- Interpret a graph's shape as engineering information, not just a visual.
- Identify and fix six common spreadsheet modeling errors.
- Apply conditional formatting and IF logic for piecewise engineering models.

8.1: From Calculator to Model
8.2: Structuring an Engineering Spreadsheet
8.3: Absolute vs Relative References
8.4: Parameter Sweeps
8.5: Graphical Visualization
8.6: Conditional Formatting and IF Logic
8.7: Six Common Spreadsheet Errors
8.8: Spreadsheets as Design Exploration Tools
8.9: When Spreadsheets Are Not Enough
8.10: Summary
8.11: End-of-Chapter Problem Set

This page titled 8: Spreadsheet Modeling is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

8.1: From Calculator to Model

8.1 From Calculator to Model

In Chapters 5 and 6, you computed single numerical answers. For the cable tension problem:

$$T = \frac{mg}{2\sin\theta}$$

Engineering design, however, almost never asks about *one* angle. It asks: What happens to tension as the cable angle varies from 10° to 80°? At what voltage does this resistor exceed its power rating? What combination of geometry gives the lightest bracket that still meets the stress constraint?

These are questions about **behavior across a range**. A spreadsheet is an engineering **modeling environment** where:

- Parameters are defined once and changed in one place
- Equations are implemented once and applied across a range automatically
- Results are visualized immediately as graphs
- Design alternatives can be compared side by side

A calculator gives you one answer. A spreadsheet lets you understand behavior. That difference is what makes spreadsheets a core engineering tool — not a substitute for a calculator, but a fundamentally different kind of instrument.

This page titled 8.1: From Calculator to Model is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

8.2: Structuring an Engineering Spreadsheet

8.2 Structuring an Engineering Spreadsheet

A disciplined engineering spreadsheet has three clearly separated regions:

Three-region structure for a disciplined engineering spreadsheet.

Region	What goes here	Why it matters
Inputs	Named parameters with values and units	Change once, updates everywhere; auditable
Model equations	Formula cells only — no hardcoded numbers	Transparent, checkable by others
Outputs / results	Labeled computed values and graphs	Clear interpretation, defensible conclusions

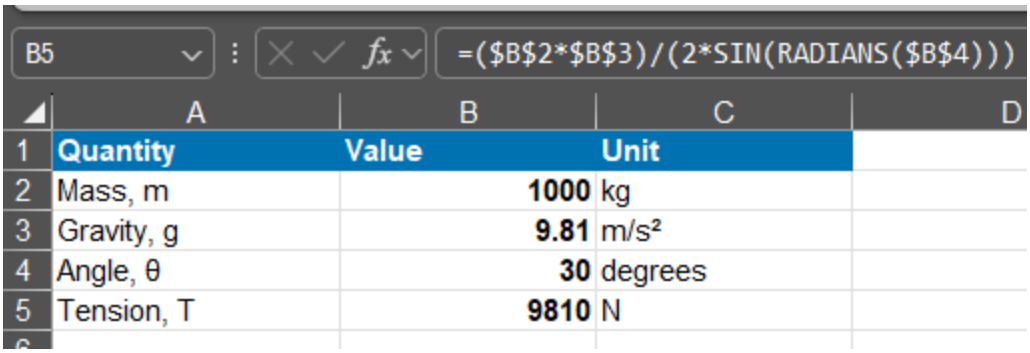
Every input cell must have a plain-language label, the value, and the unit. The canonical inputs block for the cable tension model:

```
B2: 1000 ← Mass, m (kg)
B3: 9.81 ← Gravity, g (m/s²)
B4: 30 ← Angle, θ (degrees — single-case only)
```

The model cell references these inputs using absolute references and converts the angle:

```
B5: =($B$2*$B$3)/(2*SIN(RADIANS($B$4)))
```

Notice: `$B$2` locks to the mass cell; `RADIANS($B$4)` converts degrees to radians before passing to `SIN()`. **Never hardcode a constant inside a formula.** Writing `=1000*9.81/...` hides the input; change the mass and nothing updates.



	A	B	C	D
1	**Quantity**	**Value**	**Unit**	
2	Mass, m	**1000**	kg	
3	Gravity, g	**9.81**	m/s²	
4	Angle, θ	**30**	degrees	
5	Tension, T	**9810**	N	

Figure 8.2.1: Excel single-case cable tension model using labeled inputs, units, absolute cell references, and degree-to-radian conversion. The formula in cell B5 computes the cable tension $T = mg/(2\sin\theta)$ using mass, gravity, and angle values from the input cells. (Copyright; Jonathan Compton, CC BY-NC 4.0)

This page titled 8.2: Structuring an Engineering Spreadsheet is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

8.3: Absolute vs Relative References

8.3 Absolute vs. Relative References

Relative reference (B3): shifts when the formula is copied. Move one row down, and B3 becomes B4.

Absolute reference ($B$3): locked. Copy the formula anywhere and it still points to B3.

The rule for parameter sweeps:

> **Constants use absolute references. The swept variable uses a relative reference.**

In a tension vs. angle sweep, mass and gravity are constants in specific cells. The angle is the swept variable occupying a column. The correct formula placed in the tension column:

```
=($B$1*$B$2)/(2*SIN(RADIANS(A10)))
```

$B$1 and $B$2 are locked (constants). A10 is relative (shifts to A11, A12, etc. as the formula copies down). Copy this formula down 14 rows and you instantly have tension at 15 different angles.

> **⚠ Watch Out — Forgetting RADIANS()**
>
> SIN(30) does not compute sin(30°). It computes sin(30 radians) ≈ −0.988. The correct result for sin(30°) is 0.500.
>
> The error produces **no warning**. The spreadsheet runs silently and produces a completely wrong answer. Always wrap trigonometric arguments: SIN(RADIANS(A10)), COS(RADIANS(A10)), TAN(RADIANS(A10)).

This page titled 8.3: Absolute vs Relative References is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

8.4: Parameter Sweeps

8.4 Parameter Sweeps

A parameter sweep systematically varies one input and computes the corresponding output at each value. The sweep occupies two adjacent columns — the parameter values and the computed outputs.

For a tension vs. angle sweep from 10° to 80° in 5° increments:

Spreadsheet layout for a cable tension parameter sweep.

Column A — Angle (°)	Column B — Tension
10	`=($B$1*$B$2)/(2*SIN(RADIANS(A10)))`
15	← *copy formula down*
…	← *copy formula down*
80	← *copy formula down*

✓ Worked Example 8.1 — Cable Tension Parameter Sweep

Setup: Inputs: B1 = 1000 (kg), B2 = 9.81 (m/s²). Angle column A10:A24 = 10, 15, 20, …, 80. Formula in B10 copied down to B24.

Selected results:

Selected cable tension results from the angle parameter sweep.

Angle (°)	Tension (N)
10	28,129
20	14,327
30	9,810
45	6,937
60	5,659
80	4,977

Interpretation: Tension at 10° is 28,129 N — nearly 3× the tension at 30° and almost 6× the tension at 80°. This confirms

the nonlinear sensitivity: shallow angles dramatically increase cable load. A design using this cable must specify the minimum acceptable angle.

This page titled 8.4: Parameter Sweeps is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

8.5: Graphical Visualization

8.5 Graphical Visualization

A table of numbers is a result. A graph of those numbers is *understanding*.

For engineering parameter sweeps, always use an **XY scatter plot** — not a bar chart or line chart. Every engineering graph must have:

1. **Title** — what system is being modeled
2. **X-axis label with units** — "Angle (degrees)", not just "Angle"
3. **Y-axis label with units** — "Tension (N)", not just "Tension"
4. **Appropriate scale** — y-axis should not start at an arbitrary value
5. **No decorative formatting** — no 3D effects, unnecessary fills, or distracting styles

A graph without labeled axes is not an engineering graph. It is a picture of a curve.

The two graphs below use the same cable tension data. The first graph shows professional engineering graph formatting; the second graph shows common choices that make a graph less useful for engineering communication.

✓ Good Engineering Graph

This graph uses a descriptive title, labeled axes with units, an appropriate zero-based scale, and an XY scatter format for continuous numerical data.

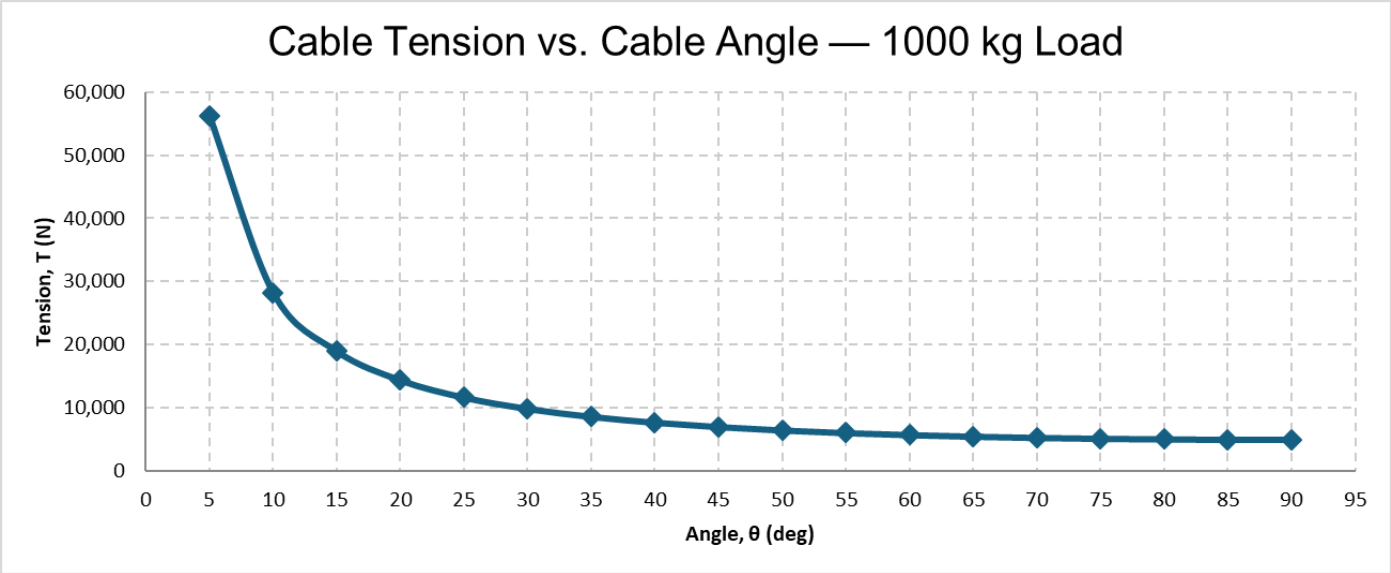


Figure 8.5.1: Properly formatted engineering graph of cable tension versus cable angle for a $1000\ \text{kg}$ load. The graph uses a descriptive title, labeled axes with units, an XY scatter format, and a zero-based y-axis. (Copyright; Jonathan Compton, CC BY-NC 4.0).

Poor Graphing Choices

This graph shows common mistakes: a generic title, missing axis units, a bar chart for continuous data, and formatting that makes the relationship harder to interpret.

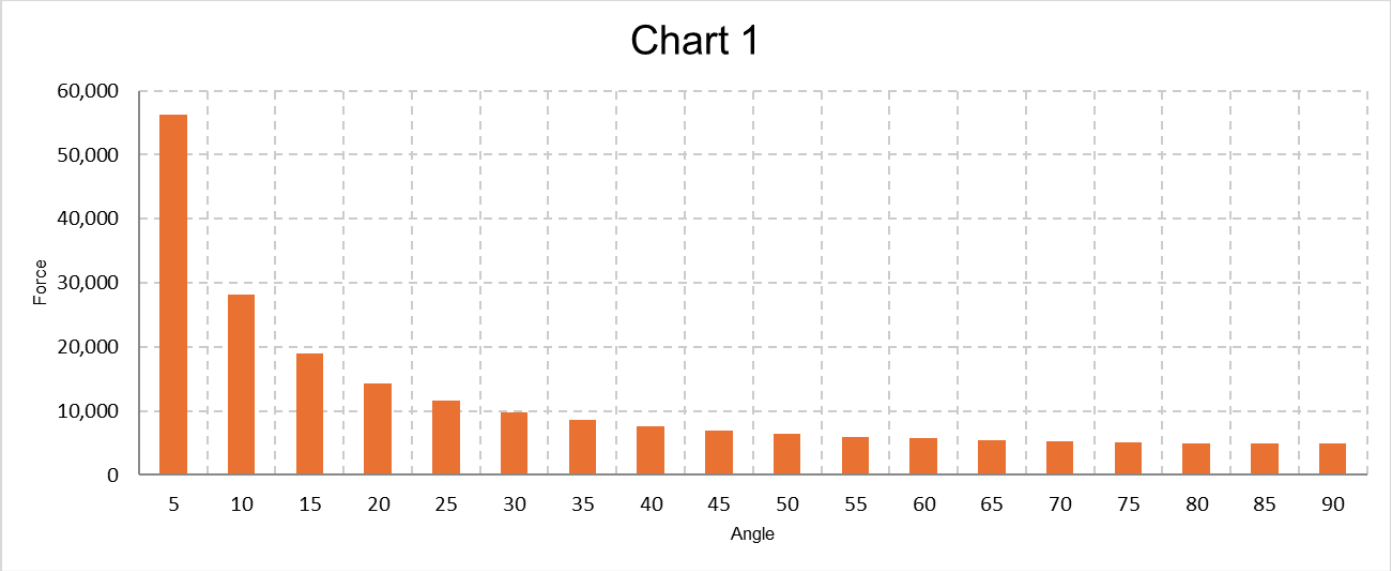


Figure 8.5.2: Poorly formatted version of the same cable tension data. The graph uses a generic title, omits units from the axis labels, and displays continuous angle data as a bar chart rather than an XY scatter plot. (Copyright; Jonathan Compton, CC BY-NC 4.0).

What the Graph Tells You

When you look at an engineering graph, extract engineering information:

- **Where is the curve steepest?** High sensitivity — small input changes produce large output changes. For tension vs. angle, this is the 10°–30° region.
- **Where does it flatten?** Low sensitivity — more design flexibility here.
- **Where does the output exceed a constraint?** Draw a horizontal line at the rating limit and see where the curve crosses it.

✓ Worked Example 8.2 — Interpreting a Power vs. Voltage Graph

Scenario: A 330 Ω resistor rated at 0.25 W. Sweep voltage 1 V to 12 V. At what voltage does power exceed the rating?

Formula: `=A10^2/$B$1` (R in B1 = 330)

Key results

Selected resistor power results from the voltage parameter sweep.

V (V)	P (W)	
5	0.076	Safe
9	0.245	Safe
9.1	**0.251**	⚠ **Exceeds 0.25 W**
12	0.436	Unsafe

Analytical crossover point:

$$V_{\max} = \sqrt{P_{\max} \cdot R} = \sqrt{0.25 \times 330} = \sqrt{82.5} \approx 9.08 \text{ V}$$

Conclusion: Safe operating range is below 9.08 V. If the source can reach 12 V, a higher-rated resistor (≥0.5 W) or larger resistance is needed.

This page titled 8.5: Graphical Visualization is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

8.6: Conditional Formatting and IF Logic

8.6 Conditional Formatting and IF Logic

Engineering models frequently involve piecewise behavior — different equations apply in different regions. Spreadsheets handle this through the `IF` function and conditional formatting.

The IF Function

```
=IF(condition, value_if_true, value_if_false)

=IFS(condition1, result1, condition2, result2, ..., TRUE, default)
```

For multiple conditions, `IFS` is cleaner and less error-prone than nested `IF` statements.

Example — safety flag for a resistor

```
=IF(C10 > 0.25, "UNSAFE", "SAFE")
```

If the power in C10 exceeds 0.25 W, the cell displays "UNSAFE." Otherwise "SAFE." This provides an automatic safety check that updates whenever inputs change.

Conditional Formatting

Conditional formatting changes cell color based on value — purely visual, no effect on calculations. It provides an immediate visual map of the data without requiring the reader to scan numbers row by row.

> **🔗 Connection — Excel Cable Car Project**
>
> The Excel Cable Car Project is the primary application of this chapter. A cable car moves along a 500-foot cable. Its velocity follows different equations depending on proximity to each tower:
>
> - Within 30 feet of a tower (30 ft included): $v = 2.425 + 0.00175 \cdot d^2$ ft/sec, where d is distance from the nearest tower
> - More than 30 feet from a tower: $v = 2.425 + 0.12 \cdot d - 0.00025 \cdot d^2$ ft/sec, where d is distance from the nearest tower
>
> The project requires you to:
>
> - Compute position and nearest-tower distance for 0 to 500 feet in 5-foot increments — using a single formula for all

cases

- Apply the correct velocity equation using `IF` or `IFS` — single formula for all cases
- Apply conditional formatting (blue/green/yellow/red) based on velocity percentile
- Generate a scatter plot of velocity vs. position with labeled axes
- Write a one-page technical memo answering specific engineering questions about the motion profile

This maps directly to the computational problem-solving model: problem statement → hand example → algorithm development → test → verify accuracy.

This page titled 8.6: Conditional Formatting and IF Logic is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

8.7: Six Common Spreadsheet Errors

8.7 Six Common Spreadsheet Modeling Errors

Spreadsheets silently amplify both good practice and bad. These six errors produce wrong answers without warning:

1. **Hardcoded constants in formulas.** Writing `=1000*9.81/...` instead of referencing input cells means changing mass requires hunting through every formula.
2. **Using degrees without `RADIANS()`.** `SIN(30)` ≠ sin(30°). Silent, wrong, no error message.
3. **Relative references where absolute are needed.** Copying a formula with `B1` (not `$B$1`) shifts the reference and breaks the sweep silently.
4. **Unlabeled inputs.** `B1: 1000` — kilograms? Pounds? Grams? Unit-unlabeled inputs create ambiguity that propagates through the model.
5. **Graphs without labeled axes.** A curve without axis labels communicates nothing engineeringly useful. It is not an engineering graph.
6. **Truncated y-axis.** Starting a y-axis at 8,000 when data ranges 8,000–12,000 makes a 50% variation look like a 500% variation. This is misleading communication — even if unintentional.

⚠ Ethics Check — Misleading Graphs

A student presents a graph of cable tension vs. angle that appears to show tension increasing by roughly 10× across the design range. A reviewer realizes the y-axis starts at 8,000 N rather than zero, compressing what is actually a 2.5× increase into a visual that looks like an order of magnitude.

Is this an error or dishonesty? Does intent matter? What is the engineer's responsibility when presenting data to a client, a regulator, or the public? Axis truncation is one of the most common forms of misleading engineering communication — and it is entirely the model builder's responsibility to catch and correct in their own work.

This page titled 8.7: Six Common Spreadsheet Errors is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

8.8: Spreadsheets as Design Exploration Tools

8.8 Spreadsheets as Design Exploration Tools

Compare how design decisions look with and without a spreadsheet:

Comparison of single-calculation and spreadsheet parametric design approaches.

Single calculation approach	Spreadsheet parametric approach
Compute tension at θ = 30° Result: 9810 N Select cable for 10 kN rating	Compute tension at θ = 10° through 80° Observe: 28,000 N (10°) to 5,000 N (80°) Identify: 12 kN limit is satisfied only above ~24° Decision: minimum angle 25°, cable rated for 15 kN

The parametric approach answers not just "does this design work?" but "what are its limits?" and "how close are we?" That is design-level reasoning.

This page titled 8.8: Spreadsheets as Design Exploration Tools is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

8.9: When Spreadsheets Are Not Enough

8.9 When Spreadsheets Are Not Enough

Spreadsheets are excellent for algebraic relationships, single-variable parameter sweeps, design alternative comparisons, and basic visualization. They start to struggle with:

- **Large matrix systems** — circuit analysis with many nodes, structural analysis with many members
- **Iterative algorithms** — optimization loops, numerical root-finding
- **Complex conditional logic** — more than a few nested IFs becomes unreadable and error-prone
- **Automation and reproducibility** — a spreadsheet run is difficult to reproduce exactly with different input data

Chapter 10 introduces MATLAB, which handles these tasks cleanly. Think of spreadsheets as the entry point into computational modeling — they make the concepts concrete — and MATLAB as the more powerful tool you graduate to as problems grow.

Key Takeaway

A spreadsheet is an engineering modeling environment, not just a calculator. The three-region structure (inputs / equations / outputs), absolute references for constants, `RADIANS()` for angle conversion, and labeled axes on all graphs are the habits that separate reliable models from ones that silently produce wrong answers. A parameter sweep turns "what is the answer?" into "how does the answer behave?" — which is what design actually asks.

This page titled 8.9: When Spreadsheets Are Not Enough is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

8.10: Summary

Summary

- Spreadsheets enable parametric modeling — define equations once, explore behavior across a range.
- Three-region structure: inputs (labeled with units) / model equations (no hardcoded constants) / outputs.
- Absolute references (`$B$1`) lock constants; relative references (`A10`) sweep with the formula.
- `RADIANS()` must wrap every angle argument in trigonometric functions. Missing it produces silent errors.
- XY scatter plots require a title, labeled axes with units, and an appropriate scale. Graph shape carries engineering information about sensitivity and constraints.
- IF/IFS enables piecewise models; conditional formatting provides visual summaries.
- Six common errors (hardcoded constants, missing RADIANS, wrong reference types, unlabeled units, unlabeled axes, truncated y-axis) produce wrong or misleading results silently.
- Parametric modeling turns analysis into design exploration: it reveals behavioral ranges, constraint boundaries, and sensitivity.

This page titled 8.10: Summary is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

8.11: End-of-Chapter Problem Set

End-of-Chapter Problem Set

Structure and Setup

1. Write the Excel formula for current $I = V/R$ where V is in cell B1 and R is in B2. Use proper absolute references.
2. Write the Excel formula for cable tension where mass is in B1, g is in B2, and angle is in the current row's column A. Use proper reference types.
3. Identify all errors in this formula: `=1000*9.81/(2*SIN(30))`

Parameter Sweeps

4. Set up a tension vs. angle sweep from 10° to 80° in 5° increments for a 1000 kg load. Report tensions at 15°, 30°, 45°, and 75°. Generate a labeled scatter plot.
5. Set up a power vs. voltage sweep from 1 V to 20 V for R = 470 Ω. Generate a labeled scatter plot. At what voltage does power exceed 0.5 W?
6. A 9 V supply connects through a variable resistor swept from 100 Ω to 2000 Ω in 100 Ω steps. Compute current and power. Plot power vs. resistance. Interpret the shape.

Graph Interpretation and Conditional Logic

7. Describe three pieces of engineering information you would extract from a tension vs. angle graph. For each, state what design decision it would inform.
8. A student's power graph has the y-axis starting at 0.20 W with data ranging 0.20–0.25 W. What problem does this create? What should be done?
9. Write an Excel IF formula that flags a resistor as "UNSAFE" if power in cell C10 exceeds 0.25 W, and "SAFE" otherwise.
10. *[Cable Car Project prep]* A cable car is 20 feet from the nearest tower. Using $v = 0.3d^{0.65}$, compute velocity by hand. Then write the Excel formula that computes it with d in cell D10.

Answer Key - click to expand

Structure and Setup

1. **Formula:** `=$B$1/$B$2`
2. **Formula:** `=($B$1*$B$2)/(2*SIN(RADIANS(A10)))`
3. **Errors:**

- The formula hardcodes 1000, 9.81, and 30 instead of referencing input cells.
- Excel evaluates `SIN(30)` as 30 radians, not 30 degrees.
- The angle should be written as `SIN(RADIANS(30))`, or better, should reference a cell containing the angle.
- The formula cannot be copied correctly for a parameter sweep because the angle is fixed.

A better single-case version would be `=($B$1*$B$2)/(2*SIN(RADIANS($B$3)))`, assuming mass, gravity, and angle are in B1, B2, and B3.

Parameter Sweeps

4. **Setup:** Put the angles $10, 15, 20, \ldots, 80$ in column A. Put $m = 1000$ in B1 and $g = 9.81$ in B2. In the first tension cell, use `=($B$1*$B$2)/(2*SIN(RADIANS(A10)))`, then copy down.

   Selected results: At $15\ °$, $T \approx 18{,}952\ \text{N}$. At $30\ °$, $T = 9810\ \text{N}$. At $45\ °$, $T \approx 6937\ \text{N}$. At $75\ °$, $T \approx 5078\ \text{N}$.

   Graph: Use an XY scatter plot. Label the x-axis “Cable Angle (degrees)” and the y-axis “Tension (N).” A clear title is “Cable Tension vs. Cable Angle — 1000 kg Load.”

5. **Setup:** Put $R = 470$ in B1. Put voltage values $1, 2, 3, \ldots, 20$ in column A. Use $P = V^2/R$. In the first power cell, use `=A10^2/$B$1`, then copy down.

   The crossover voltage for $0.5\ \text{W}$ is $V = \sqrt{PR} = \sqrt{(0.5)(470)} \approx 15.33\ \text{V}$.

   Answer Power exceeds $0.5\ \text{W}$ above about **15.3 V**. In a 1 V increment table, the first listed voltage above the limit is **16 V**.

   Graph: Use an XY scatter plot. Label the x-axis “Voltage (V)” and the y-axis “Power (W).” A clear title is “Power vs. Voltage — 470 Ω Resistor.”

6. **Setup:** Put $V = 9$ in B1. Put resistance values $100, 200, 300, \ldots, 2000$ in column A. Use $I = V/R$ and $P = V^2/R$.

   Formulas: Current formula: `=$B$1/A10`. Power formula: `=$B$1^2/A10`.

   At $R = 100\ \Omega$, $I = 0.09\ \text{A} = 90\ \text{mA}$ and $P = 0.81\ \text{W}$. At $R = 2000\ \Omega$, $I = 0.0045\ \text{A} = 4.5\ \text{mA}$ and $P = 0.0405\ \text{W}$.

   Interpretation: Power decreases as resistance increases. The curve is nonlinear because resistance is in the denominator of $P = V^2/R$. At low resistance, small resistance changes cause large power changes. At high resistance, the curve flattens.

Graph Interpretation and Conditional Logic

7. **Three useful pieces of engineering information:**
   - **Maximum tension region:** Low cable angles produce the highest tensions. This informs the minimum allowable cable angle.
   - **Sensitivity region:** The graph is steepest at small angles. This informs how tightly the installation angle must be controlled.
   - **Rating crossover:** A horizontal cable-rating line shows where the design becomes unsafe. This informs cable selection and safety factor decisions.
8. A y-axis starting at $0.20\ \text{W}$ can visually exaggerate small differences because the graph removes the zero baseline. A change from 0.20 to $0.25\ \text{W}$ may look much larger than it really is.

   For introductory engineering communication, the y-axis should usually start at zero unless there is a clear reason to truncate it. If a truncated axis is used, it should be explicitly marked or explained.

9. **Formula:** `=IF(C10>0.25,"UNSAFE","SAFE")`
10. **Hand calculation:** $v = 0.3(20)^{0.65} \approx 2.10$.

    Excel formula: `=0.3*D10^0.65`

    If the coefficient and exponent are stored in input cells, use absolute references. For example, if 0.3 is in B1 and 0.65 is in B2, use `=$B$1*D10^$B$2`.

This page titled 8.11: End-of-Chapter Problem Set is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

9: DATA VISUALIZATION FOR ENGINEERS

Data Visualization for Engineers

Graphs are engineering analysis tools, not report decorations. The shape, scale, and range of a graph determine whether it reveals or conceals the behavior that matters for a decision.

Learning Objectives

- Explain why graphs communicate behavior more effectively than tables.
- Select the appropriate chart type for engineering data.
- Construct a properly labeled engineering graph meeting all five professional requirements.
- Distinguish linear from nonlinear trends by graph shape and explain engineering implications.
- Identify at least four ways a graph can mislead a viewer.
- Apply logarithmic scales appropriately and interpret them correctly.
- Use multi-curve graphs to compare design alternatives.
- Add and interpret trendlines (including forced-zero-intercept and extended forecast) and error bars in Excel.
- Select the correct chart type for categorical data (pie, column) and explain when each is appropriate versus when it is not.

9.1: Why Graphs Are an Engineering Tool
9.2: Chart Type Selection
9.3: Anatomy of an Engineering Graph
9.4: Linear vs Nonlinear Trends
9.5: Misleading Visualizations
9.6: Logarithmic Scales
9.7: Multi-Curve Comparison
9.8: Visualization and Interpretation
9.9: Advanced Chart Features
9.10: Graph Checklist — Quick Reference
9.11: Summary
9.12: Quick Check

This page titled 9: Data Visualization for Engineers is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

9.1: Why Graphs Are an Engineering Tool

9.1 Why Graphs Are an Engineering Tool

In Chapter 8 you built parameter sweeps — tables of 15, 20, 50 rows of numbers. Looking at those numbers directly to find where power exceeds a rating, or where tension becomes dangerous, requires scanning every row and comparing mentally.

A graph makes that information visible in a single glance. But graphs are not just convenient summaries — they reveal things tables cannot:

- **Shape** — whether a relationship is linear, quadratic, inverse, or something else
- **Sensitivity** — where small input changes produce large output changes
- **Constraint crossings** — exactly where the system enters an unsafe or infeasible region
- **Comparative behavior** — how two designs differ across a full range, not just at one point

This is why engineers draw graphs before making decisions, not after. The graph is part of the analysis.

This page titled 9.1: Why Graphs Are an Engineering Tool is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

9.2: Chart Type Selection

9.2 Choosing the Right Chart Type

Appropriate and inappropriate uses of common chart types.

Chart type	When to use	When NOT to use
XY scatter plot	Both axes continuous numeric (voltage vs current, angle vs tension)	Categorical comparisons
Bar chart	Comparing discrete categories (Material A vs B at one condition)	Continuous parameter sweeps — bars misrepresent continuity
Pie chart	Proportional shares of a categorical whole (e.g., percentage of graduates by discipline)	Any continuous or numerical relationship — the slices cannot show rate of change
Column / bar chart	Comparing discrete named categories (top 5 schools by enrollment, material strengths)	Continuous parameter sweeps — bars visually imply that the gaps between categories are meaningless, which misrepresents the math
3D charts	Rarely — only when a third numeric axis genuinely adds information	2D data presented in 3D — perspective distortion makes values impossible to compare accurately

For the parameter sweeps in Chapters 8 and 10, always use **XY scatter plots**. Both axes are continuous numeric quantities and the relationship is mathematical. Pie and column charts have their place — but only for *categorical* data where you are counting or comparing named groups, not modeling a physical relationship.

This page titled 9.2: Chart Type Selection is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

9.3: Anatomy of an Engineering Graph

9.3 The Anatomy of an Engineering Graph

Every professional engineering graph must meet five requirements:

1. **Title** — what system is modeled and what relationship is shown.

   Good: "Cable Tension vs. Cable Angle — 1000 kg Load" Insufficient: "Chart 1"
2. **X-axis label with units.**

   Good: "Cable Angle (degrees)" Insufficient: "Angle" or "X"
3. **Y-axis label with units.**

   Good: "Tension (N)" Insufficient: "Force" or "Y"
4. **Appropriate scale** — y-axis starts at zero unless there is a stated, defensible reason not to. Scale neither compresses nor exaggerates the variation of interest.
5. **Clean formatting** — no 3D effects, no decorative fills. Light gridlines if any. Data is the visual.

The two graphs below use the same cable tension data. The first graph shows professional engineering graph formatting; the second graph shows common choices that make a graph less useful for engineering communication.

✓ Good Engineering Graph

This graph uses a descriptive title, labeled axes with units, an appropriate zero-based scale, and an XY scatter format for continuous numerical data.

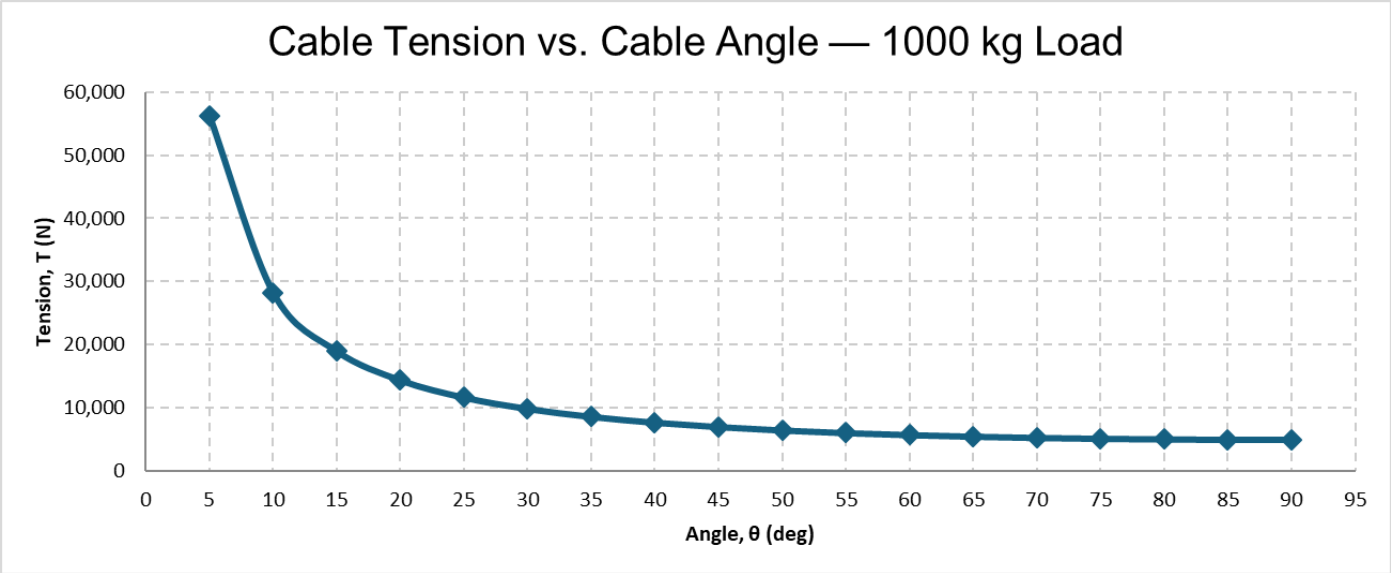


Figure 9.3.1: Properly formatted engineering graph of cable tension versus cable angle for a $1000\ \mathrm{kg}$ load. The graph uses a descriptive title, labeled axes with units, an XY scatter format, and a zero-based y-axis. (Copyright; Jonathan Compton, CC BY-NC 4.0).

Poor Graphing Choices

This graph shows common mistakes: a generic title, missing axis units, a bar chart for continuous data, and formatting that makes the relationship harder to interpret.

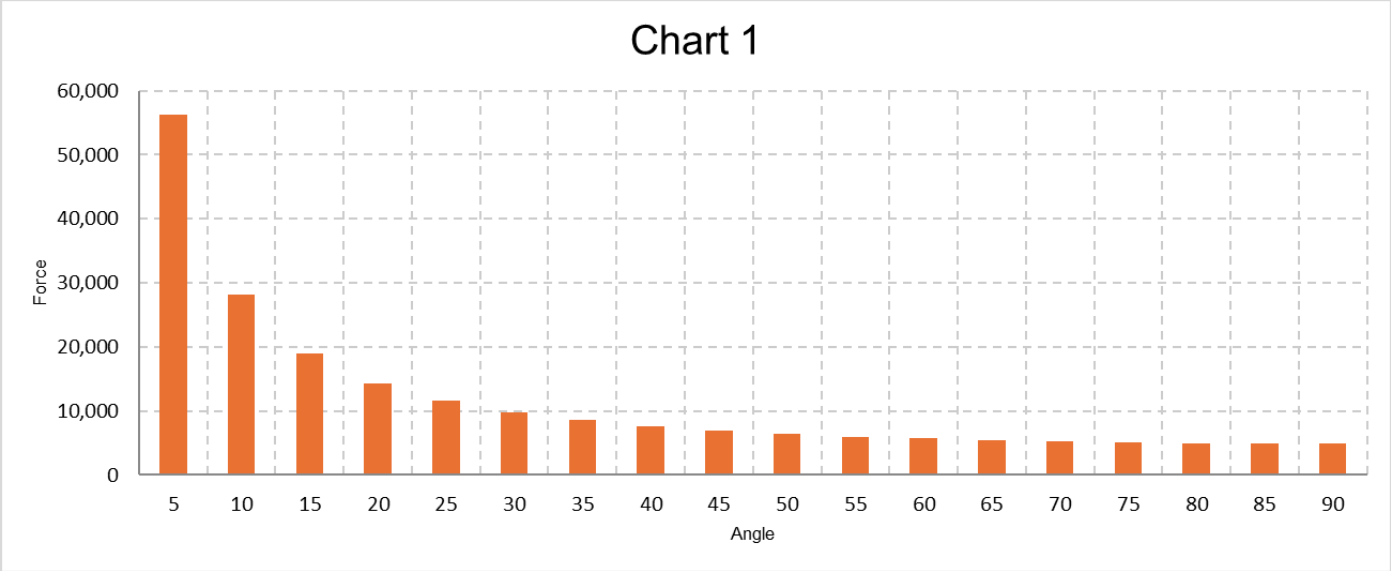


Figure 9.3.2: Poorly formatted version of the same cable tension data. The graph uses a generic title, omits units from the axis labels, and displays continuous angle data as a bar chart rather than an XY scatter plot. (Copyright; Jonathan Compton, CC BY-NC 4.0).

This page titled 9.3: Anatomy of an Engineering Graph is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

9.4: Linear vs Nonlinear Trends

9.4 Linear vs. Nonlinear Trends

Graph shape communicates how a system behaves as you push it.

Common graph shapes, example equations, and engineering implications.

Shape	Example equation	Engineering implication
Straight line	$I = V/R$ (constant R)	Constant rate of change. Easy to extrapolate. Doubling V doubles I.
Upward parabola	$P = V^2/R$	Accelerating increase. Doubling V quadruples P. Safe at 6 V, near limit at 9 V, destroyed at 12 V.
Hyperbola (sharp drop then flat)	$T = mg/(2\sin\theta)$	High sensitivity at low angles, low sensitivity at high angles. Small errors near 10° are far more dangerous than near 80°.

Understanding graph shape is about knowing which region of operation requires the most care — where the margin erodes fastest.

▶ Try It — Shape Recognition

Match each equation to its graph shape and state one engineering consequence:

1. $P = V^2/R$ (V varies, R fixed)
2. $I = V/R$ (V varies, R fixed)
3. $T = mg/(2\sin\theta)$ (θ varies, m fixed)
4. $R_{\text{total}} = R_1 + R_2$ (R_1 varies, R_2 fixed)

Answers - click to expand

1. $P = V^2/R$**: upward parabola.**

   With R fixed, power increases with the square of voltage. Doubling V makes P four times larger. The engineering consequence is that electrical power and heat can rise much faster than expected, so a resistor that is safe at a lower voltage may exceed its power rating at a slightly higher voltage.

2. $I = V/R$**: straight line.**

   With R fixed, current is directly proportional to voltage. Doubling V doubles I. The engineering consequence is that the relationship is predictable and easy to extrapolate, but current limits still need to be checked as voltage increases.

3. $T = mg/(2\sin\theta)$**: nonlinear hyperbola-like curve.**

   As θ decreases, $\sin\theta$ decreases, and because it is in the denominator, T increases. The curve becomes especially steep at small angles. The engineering consequence is that shallow cable angles are dangerous: small installation errors near low angles can cause large increases in cable tension.

4. $R_{\text{total}} = R_1 + R_2$**: straight line.**

   With R_2 fixed, total resistance increases one-for-one as R_1 increases. The engineering consequence is that series resistance is simple to predict: adding more resistance directly increases the total resistance by the same amount.

This page titled 9.4: Linear vs Nonlinear Trends is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

9.5: Misleading Visualizations

9.5 Misleading Visualizations — A Case Study

⚠ Real-World Example — The Challenger O-Ring Data

On January 27, 1986, the night before the Space Shuttle Challenger launch, engineers at Morton Thiokol presented data to NASA managers about O-ring erosion and launch temperature. Their concern: the forecast launch temperature of 26°F was far colder than any previous launch, and they believed cold increased O-ring failure risk.

The data supported their concern — but the graphs they used nearly concealed it. Engineers had plotted *only the flights that showed O-ring damage*, omitting flights with no damage. When you exclude the undamaged cases, there is no visible temperature trend — you have removed the comparison. A plot of *all* flights shows the relationship clearly: damage was concentrated in cold launches.

NASA managers, seeing graphs that showed no clear trend, asked engineers to prove cold caused failure. Engineers could not make their case from the data as presented. The launch proceeded. Challenger was destroyed 73 seconds after liftoff due to O-ring failure in the cold.

The lesson: Incomplete data, cherry-picked cases, or poorly constructed graphs can lead experienced engineers to wrong conclusions — with fatal consequences. Plotting all relevant data honestly is not just good practice. It is an ethical obligation.

Four Ways Graphs Mislead

1. **Truncated y-axis.** Starting at a non-zero value compresses or exaggerates variation. A 5% data change can appear to be a 500% change. Start at zero unless there is a stated, defensible reason.
2. **Cherry-picked data range.** Plotting only the "good" part of the operating range hides constraint violations that appear elsewhere. Plot the full relevant range.
3. **Wrong chart type.** A bar chart of a continuous power sweep misrepresents the continuous nature of the relationship. Match chart type to data structure.
4. **Missing units on axes.** A graph of "Force" vs. "Angle" with no units cannot be verified, reproduced, or trusted. Axis labels without units are incomplete.

This page titled 9.5: Misleading Visualizations is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

9.6: Logarithmic Scales

9.6 Logarithmic Scales

Linear axes work well when data spans a modest range. When data spans multiple orders of magnitude — which happens constantly in electrical engineering — logarithmic scales become necessary.

On a log scale, each decade (factor of 10) takes the same physical space. Values are spaced as: 0.001, 0.01, 0.1, 1, 10, 100 ... rather than 0, 1, 2, 3, 4 ...

Engineering situations where logarithmic scales improve graph interpretation.

Situation	Why log scale helps
Frequency response plots (Bode plots)	Frequency ranges from ~1 Hz to ~1 MHz — linear scale crushes the low end
Component value selection	Capacitors span pF (10^{-12}) to mF (10^{-3}) — 9 orders of magnitude
Signal amplitudes in dB	Decibels are a logarithmic unit by definition
Reaction rates, material fatigue cycles	Quantities that span many powers of 10

A common mistake: plotting data spanning multiple orders of magnitude on a linear scale, then concluding that small values are negligible because they appear flat near the axis. On a log scale, those same values may show important structure.

The two graphs below use the same RC low-pass frequency response data. The first graph uses a linear frequency axis, which compresses the low-frequency region. The second graph uses a logarithmic frequency axis, which spreads each decade evenly and makes the full response easier to interpret.

Linear Axis — Low-Frequency Detail Compressed

The linear frequency axis places most of the visual space at high frequencies. When the data spans several orders of magnitude, important low-frequency behavior is crowded near the left edge.

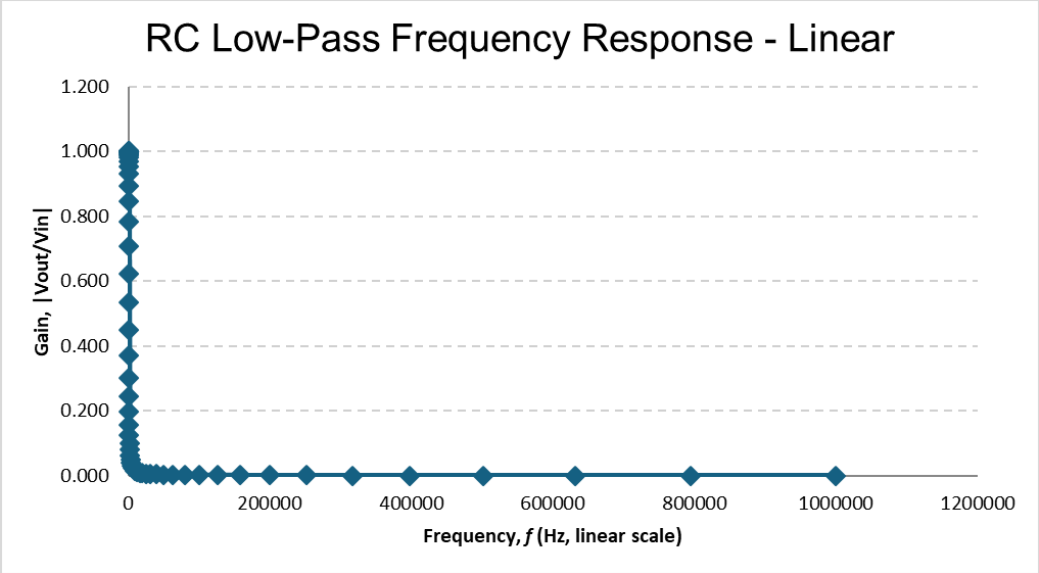


Figure 9.6.1: RC low-pass frequency response plotted with a linear frequency axis. Because the frequency range spans several orders of magnitude, the low-frequency portion of the response is compressed near the left side of the graph. (Copyright; Jonathan Compton, CC BY-NC 4.0).

Logarithmic Axis — Full Structure Visible

The logarithmic frequency axis gives equal visual spacing to each decade. This makes the transition region and overall trend easier to read across the full frequency range.

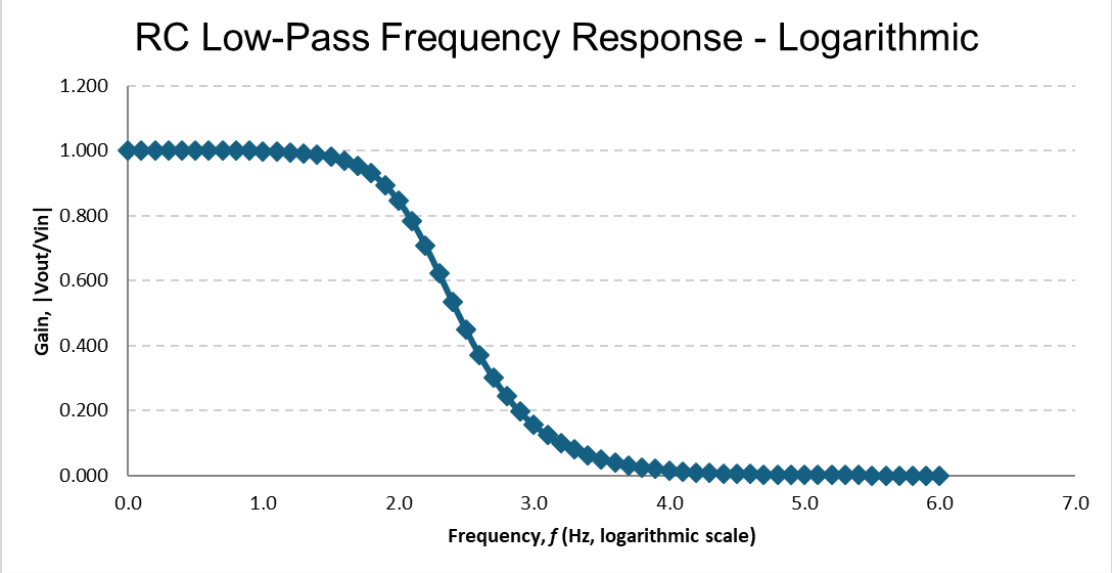


Figure 9.6.2: The same RC low-pass frequency response plotted with a logarithmic frequency axis. Equal spacing by decade makes the full response visible, including the transition region and high-frequency roll-off. (Copyright; Jonathan Compton, CC BY-NC 4.0).

This page titled 9.6: Logarithmic Scales is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

9.7: Multi-Curve Comparison

9.7 Multi-Curve Comparison Graphs

Plotting multiple curves on the same axes makes design comparisons visible. This directly supports Step 5 (trade-off evaluation) of Chapter 7's design framework.

Using $P = V^2/R$ across voltage 1–15 V for three resistance values:

Power values for three resistor choices across selected source voltages.

V (V)	P, R=220 Ω (W)	P, R=330 Ω (W)	P, R=470 Ω (W)
5	0.114	0.076	0.053
7	0.223	0.148	0.104
7.4	**0.249** ⚠	0.166	0.117
9	0.368	0.245	0.172
9.1	0.376	**0.251** ⚠	0.176
10.8	0.530	0.353	**0.248** ⚠

⚠ *marks the approximate crossover point where each resistor exceeds the 0.25 W rating.*

On a single graph with a dashed line at P = 0.25 W, you immediately read: R = 220 Ω exceeds the rating at ~7.4 V; R = 330 Ω at ~9.1 V; R = 470 Ω at ~10.8 V. If the source voltage can reach 12 V, all three exceed the rating. The 470 Ω has the most margin.

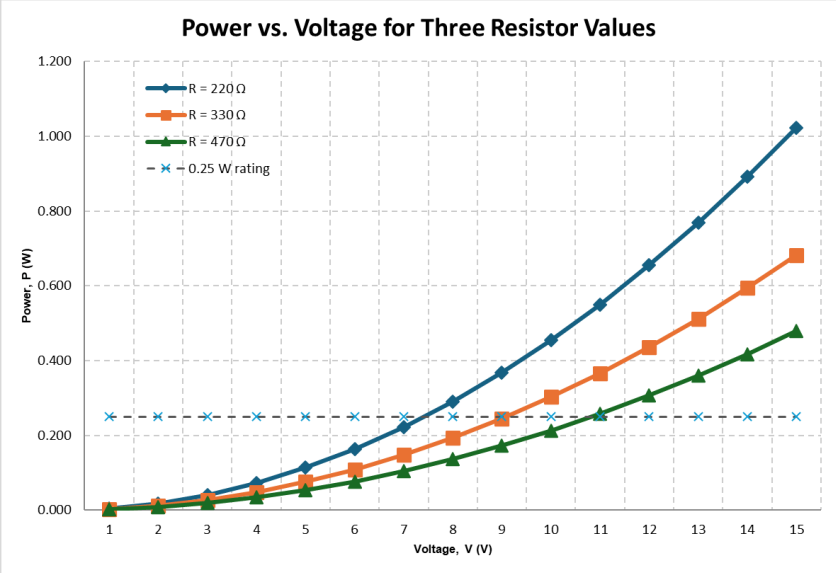


Figure 9.7.1: Power versus voltage for three resistor values calculated using $P = V^2/R$. The dashed horizontal line marks the 0.25 W resistor power rating, showing that lower resistance values exceed the rating at lower voltages. (Copyright; Jonathan Compton, CC BY-NC 4.0).

This page titled 9.7: Multi-Curve Comparison is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

9.8: Visualization and Interpretation

9.8 Visualization and Interpretation — Completing the Loop

A graph without an interpretation paragraph is not complete engineering work. For every graph you produce, write at least two or three sentences answering:

- What does the curve shape tell you about the relationship?
- Where is the region of highest sensitivity?
- Does the output stay within constraints across the full range? If not, where does it violate?
- What design decision does this graph support?

Example interpretation for the cable tension sweep

> *"Tension decreases nonlinearly with increasing cable angle. The curve is steepest between 10° and 25°, where a 5° change produces a tension difference of approximately 5,000 N. Above 45°, the curve flattens significantly. For a cable rated at 12,000 N, the safe operating range begins at approximately 24°. Cable installations should specify a minimum angle of at least 25° with margin."*

That paragraph turns a graph into engineering judgment.

Key Takeaway

A graph is an engineering analysis tool, not a report decoration. Chart type, axis scale, and data range determine whether a graph reveals or conceals the behavior that matters. Nonlinear shapes carry specific engineering implications. Log scales are necessary when data spans multiple orders of magnitude. Multi-curve graphs make design comparisons visible. Every graph requires an interpretation paragraph — the graph is the evidence; the interpretation is the conclusion.

This page titled 9.8: Visualization and Interpretation is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

9.9: Advanced Chart Features

9.9 Advanced Chart Features — Trendlines, Error Bars, and Categorical Charts

Once a basic scatter plot is correctly constructed, three additional features become useful in engineering and scientific reporting: **trendlines**, **error bars**, and **categorical charts** for reporting data. Each has a specific purpose — and misusing them is as problematic as missing axis labels.

Trendlines — Fitting a Model to Data

A trendline fits a mathematical model (linear, polynomial, exponential, power) to a set of data points. It is useful when you want to:

- **Confirm a theoretical relationship.** If theory says deflection vs. load should be linear, a linear trendline through the data either confirms or contradicts that assumption.
- **Extract a slope or equation for use in calculations.** Displaying the trendline equation gives you a formula you can apply beyond the measured data range.
- **Forecast beyond measured data.** Extending a trendline past the last data point is a forecast — useful but requires caution, because models only hold within the range where the underlying physics applies.

Trendline settings, appropriate uses, and common mistakes.

Trendline setting	When to use it	Common mistake
Force intercept = 0	When theory requires the relationship to pass through the origin (zero input → zero output), e.g., beam deflection vs. load, Ohm's law	Forcing the intercept when theory does not require it — this distorts the fit and hides real offset errors
Display equation on chart	When the slope value will be used in subsequent calculations or reported as a result	Leaving the equation in scientific notation (e.g., 2.34E+03) — always reformat to a readable number with appropriate decimal places
Forecast / extend trendline	When predicting future values based on historical trend (e.g., forecasting graduation rates in a future year)	Treating a forecast as a certainty — trendlines extrapolate the past trend, they do not guarantee the future

Error Bars — Showing Measurement Uncertainty

Every physical measurement contains uncertainty. A data point plotted without error bars says implicitly: "this value is exact." Error bars make measurement uncertainty visible.

In Excel, error bars can be added as a fixed value, a percentage of each data point, or a custom range. For engineering lab data, a fixed value (e.g., ±0.1 mm) is common when the measurement instrument has a known resolution limit.

> **Why error bars matter for engineering credibility**
>
> A trendline that passes cleanly through all error bars supports the theoretical model. A data point whose error bar does not overlap the trendline signals a potential outlier or measurement problem worth investigating. Error bars turn a graph into a quantitative argument, not just a picture.

> **Excel tip — adding only vertical error bars**
>
> When you add error bars in Excel for the first time, both horizontal and vertical bars are inserted by default. For most engineering measurements (where the x-axis values are set precisely), only vertical (y-axis) error bars are appropriate. Click on the horizontal bars and delete them separately.

Pie Charts and Column Charts — When Categorical Data Is the Point

The rule in §9.2 stands: for continuous engineering parameter relationships, XY scatter is the only appropriate choice. But engineers also work with *categorical data* — counts, enrollments, proportions by named group — and here pie and column charts earn their place.

Appropriate uses and formatting rules for pie and column charts.

Chart type	Appropriate for	Key formatting rules
Pie chart	Showing each category's share of a total — useful when there are ≤ 6 categories and the proportional story matters (e.g., percentage of PhD recipients by demographic)	Label directly on slices (category + percentage); avoid legends when possible; no 3D; rotate the chart so the largest slice starts at 12 o'clock for readability
Column chart	Comparing magnitudes across named categories (e.g., top 5 universities by BS degrees awarded)	Sort bars by value (largest to smallest) so ranking is immediately visible; label axes with units; y-axis starts at zero; avoid 3D effects and gap-filling colors

The unifying principle: always ask *what story is this chart supposed to tell?* Pie charts tell a proportional story. Column charts tell a ranking or magnitude story. Scatter plots tell a relationship story. Choosing the wrong chart type obscures the story — even when the data is correct.

▶ Try It — Which Chart?

For each dataset below, identify the correct chart type and explain why:

1. Beam deflection (mm) measured at 10 different loads (N)
2. Percentage of AVC engineering graduates who chose each of six disciplines last year
3. Number of BS engineering degrees awarded at the top 5 U.S. universities
4. Cable tension (N) calculated at cable angles from 10° to 80°
5. Forecasted graduation numbers for Mechanical, Civil, and Electrical Engineering through 2028

Answers - click to expand

1. **Beam deflection measured at 10 different loads: XY scatter plot.**

   Load is a continuous numerical input and deflection is a continuous numerical output. An XY scatter plot correctly shows the relationship between the two variables. A trendline may also be appropriate if the goal is to test whether deflection is approximately linear with load.

2. **Percentage of AVC engineering graduates by discipline: pie chart or column chart.**

   A pie chart is appropriate if the goal is to show each discipline's share of the total graduating group, especially because there are six categories. A column chart would also work if the goal is to compare disciplines more directly. The key is that the data are categorical, not continuous.

3. **Number of BS engineering degrees awarded at the top 5 U.S. universities: column chart.**

   The universities are named categories, and the data compare counts across those categories. A column chart should be sorted from largest to smallest so the ranking is immediately visible. The y-axis should start at zero and should be labeled with the unit, such as “Degrees Awarded.”

4. **Cable tension calculated at cable angles from 10° to 80°: XY scatter plot.**

   Cable angle is a continuous numerical input and tension is a continuous numerical output. An XY scatter plot preserves the numerical spacing of the angle values and shows the nonlinear relationship between angle and tension. A bar or column chart would incorrectly treat the angle values like categories.

5. **Forecasted graduation numbers for Mechanical, Civil, and Electrical Engineering through 2028: line chart or XY scatter plot with trendlines.**

   Because the data are organized over time and compare multiple disciplines, a multi-series line chart is appropriate for showing trends by year. If the year values are treated as numerical x-values and the goal is to fit or extend trends, an XY scatter plot with trendlines may also be appropriate. The forecast should be clearly labeled as a projection, not as confirmed future data.

This page titled 9.9: Advanced Chart Features is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

9.10: Graph Checklist — Quick Reference

Graph Checklist — Quick Reference

Quick-reference checklist for creating engineering graphs.

Item	✓
Chart type: XY scatter (not bar, pie, or 3D)	
Title present	
X-axis labeled with units	
Y-axis labeled with units	
Y-axis starts at zero (or stated reason if not)	
No 3D effects or decorative fills	
Constraint line drawn if applicable	
Interpretation paragraph written	
If trendline used: intercept forced only when theory requires it; equation reformatted (no scientific notation); forecast extension noted as estimate	
If error bars used: fixed value matches instrument resolution; horizontal bars removed if x-values are set precisely	
If pie/column chart: data is categorical (not a parameter sweep); bars sorted by value; labels direct on chart; y-axis starts at zero	

This page titled 9.10: Graph Checklist — Quick Reference is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

9.11: Summary

Summary

- Graphs reveal shape, sensitivity, constraint crossings, and comparative behavior — things tables cannot show efficiently.
- Use XY scatter plots for continuous parameter sweeps. Avoid bar charts, pie charts, and 3D effects.
- Five requirements: title, x-axis label with units, y-axis label with units, appropriate scale, clean formatting.
- Graph shape communicates relationship type: straight line (linear), parabola (quadratic), hyperbola (inverse). Each has specific engineering implications about where margins erode fastest.
- Misleading graphs (truncated axes, cherry-picked data, wrong chart type, missing units) can lead to wrong engineering decisions — the Challenger case demonstrates the most severe consequence.
- Logarithmic scales are appropriate when data spans multiple orders of magnitude.
- Multi-curve graphs support design comparison by overlaying alternatives against the same axes and constraint lines.
- Every graph requires an interpretation paragraph: shape, sensitivity region, constraint status, design implication.
- Trendlines fit a model to data. Forcing the intercept through zero is only correct when theory requires it. Displayed equations must be reformatted out of scientific notation. Extended trendlines are forecasts — useful but not guarantees.
- Error bars make measurement uncertainty visible. A trendline that passes through all error bars supports the theoretical model; a point whose bar doesn't overlap the trendline warrants investigation.
- Pie and column charts are appropriate only for categorical data (proportional shares, ranked counts). They are never correct for continuous parameter sweeps. The chart type must match the story the data is telling.

This page titled 9.11: Summary is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

9.12: Quick Check

Quick Check

1. You have 30 power values at different voltages. Why is a graph more useful than the table for identifying the safe operating voltage?
2. What chart type is correct for a cable tension parameter sweep? What is wrong with using a bar chart?
3. List the five requirements for a professional engineering graph. For each, state what happens if it is missing.
4. A student plots tension vs. angle and sees a curve that drops sharply from 10° to 30° then levels off. What does this shape tell them, and where should they focus safety margin?
5. A power vs. voltage graph has the y-axis starting at 0.22 W with all data between 0.22 W and 0.25 W. What problem does this create? How should it be fixed?
6. Why are logarithmic scales used for frequency response plots in electrical engineering?
7. Plot power vs. voltage for R = 220 Ω and R = 470 Ω on the same graph (1–15 V). Add a dashed line at P = 0.25 W. For each resistor, identify the maximum safe voltage.
8. *[Challenger]* The engineers omitted flights with no O-ring damage from their charts. What type of misleading visualization error does this represent? How would plotting all flights have changed the picture?
9. You fit a linear trendline to beam deflection vs. load data and the equation shows `y = 2.34E-03x`. What is wrong with leaving it in this format, and how do you fix it in Excel?
10. A classmate forces a trendline intercept to zero for a dataset where the y-intercept is clearly not zero. What is wrong with this, and when *is* forcing through zero the correct choice?
11. You are charting the percentage of engineering bachelor's degrees awarded by discipline. A classmate uses an XY scatter plot. What is wrong with that choice and what should be used instead?

Answers - click to expand

1. A graph is more useful because it shows the pattern immediately. With 30 power values, a table requires the reader to scan row by row to find where the power crosses the safe limit. A graph can show the curve and the rating line at the same time, making the safe operating voltage visible at the intersection.
2. The correct chart type is an **XY scatter plot**. Cable angle is a continuous numerical input, and tension is a continuous numerical output. An XY scatter plot preserves the numerical spacing between angles.

   A bar chart is wrong because it treats each angle value as a separate category instead of a numerical x-value. That can hide or distort the continuous relationship between angle and tension.

3. **Five requirements for a professional engineering graph:**
   1. **Descriptive title:** Without it, the viewer does not know what system or relationship is being shown.
   2. **X-axis label with units:** Without it, the input variable and its units are unclear.
   3. **Y-axis label with units:** Without it, the output quantity and its units are unclear.
   4. **Appropriate scale:** Without it, the graph may exaggerate, compress, or misrepresent the relationship.
   5. **Clean formatting:** Without it, decorative effects can distract from the data or make the graph harder to read.
4. The curve tells the student that cable tension is highly sensitive to angle at low angles. From 10° to 30°, small changes in angle cause large changes in tension. At higher angles, the curve levels off, meaning the same

angle change has a smaller effect.

The student should focus safety margin on the low-angle region, especially near 10° to 30°. That is where installation error, cable stretch, or geometry changes can produce the largest increase in tension.

5. Starting the y-axis at $0.22\ \mathrm{W}$ exaggerates the visual difference between values. A small change from $0.22\ \mathrm{W}$ to $0.25\ \mathrm{W}$ can appear much larger than it really is.

   For a basic engineering graph, the y-axis should normally start at zero. If a truncated axis is truly necessary, the graph should clearly indicate that the axis is truncated and explain why.

6. Logarithmic scales are used because frequency response data often spans many orders of magnitude, such as $1\ \mathrm{Hz}$ to $1\ \mathrm{MHz}$. On a linear scale, the low-frequency region gets compressed near the axis and important behavior can become invisible.

   On a log scale, each factor-of-10 interval takes the same physical space, so the full frequency range can be interpreted more clearly.

7. Use $P = V^2/R$ for each resistor and plot both curves from 1 to $15\ \mathrm{V}$. Add a horizontal dashed line at $P = 0.25\ \mathrm{W}$.

   The maximum safe voltage occurs where $P = 0.25\ \mathrm{W}$, so $V_{\max} = \sqrt{PR}$.

   For $R = 220\ \Omega$, $V_{\max} = \sqrt{(0.25)(220)} = \sqrt{55} \approx 7.42\ \mathrm{V}$.

   For $R = 470\ \Omega$, $V_{\max} = \sqrt{(0.25)(470)} = \sqrt{117.5} \approx 10.84\ \mathrm{V}$.

   Answer The $220\ \Omega$ resistor is safe up to about $7.4\ \mathrm{V}$. The $470\ \Omega$ resistor is safe up to about $10.8\ \mathrm{V}$, assuming a $0.25\ \mathrm{W}$ limit.

8. This is a **selection bias** or **omitted-data** visualization error. By leaving out flights with no O-ring damage, the chart only showed damaged cases and removed the comparison group needed to understand the relationship between temperature and damage.

   Plotting all flights would have shown both damaged and undamaged launches across the temperature range. That would have made the low-temperature risk pattern more visible and would have helped viewers judge whether cold launches were associated with higher O-ring damage.

9. The equation `y = 2.34E-03x` is mathematically valid, but it is not reader-friendly for most introductory engineering reports. Many readers will not quickly interpret 2.34×10^{-3}, and the slope may be used incorrectly if the units are not clear.

   In Excel, fix it by formatting the trendline label. Right-click the trendline equation, choose the formatting options for the label, and change the number format from scientific notation to a readable decimal format, such as $y = 0.00234x$. Also make sure the graph or caption identifies the slope units.

10. Forcing the intercept to zero is wrong when the data clearly have a real nonzero intercept. It can distort the slope, hide calibration offsets, and make the model look more theoretically clean than the measurements actually support.

    Forcing through zero is correct when the underlying physics requires zero output at zero input. Examples include ideal Ohm's Law data where $V = IR$, or beam deflection versus load when zero load should produce zero deflection after proper zeroing. The key rule is that the choice must come from the physical model, not from a desire to make the graph look cleaner.

11. An XY scatter plot is the wrong choice because engineering disciplines are categories, not continuous numerical x-values. A scatter plot implies a quantitative relationship along the x-axis, which does not exist for named disciplines.

For percentages by discipline, use a **pie chart** if the goal is to show each discipline's share of the total. Use a **column chart** if the goal is to compare discipline percentages more directly.

This page titled 9.12: Quick Check is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

10: MATLAB FOR ENGINEERING COMPUTATION

Introduction to MATLAB

MATLAB and spreadsheets are complementary tools. MATLAB excels at piecewise logic, large datasets, reproducible automation, and embedded safety constraints. The engineering workflow is identical — only the syntax changes.

Learning Objectives

- Describe when MATLAB is more appropriate than a spreadsheet, and vice versa.
- Write a structured MATLAB script with the standard header, comment blocks, and section format.
- Define scalar and vector variables with correct syntax and units documentation.
- Apply element-wise operators (`.*`, `./`, `.^`) to compute engineering equations across a vector.
- Use `sind()` for degree-based trigonometry without manual radian conversion.
- Generate a properly labeled engineering plot using `plot()`, `xlabel()`, `ylabel()`, `title()`, and `grid on`.
- Apply `if` statements and logical indexing to enforce safety constraints and filter design results.
- Recognize the connection between MATLAB scripting and the five-step engineering workflow.

10.1: Why Engineers Program
10.2: MATLAB as Structured Engineering
10.3: Script Basics
10.4: Four-Step Build
10.5: Trigonometry in MATLAB
10.6: Conditional Logic
10.7: Logical Indexing
10.8: Programming Project- Cable Car in MATLAB
10.9: Choosing the Right Tool
10.10: Summary
10.11: End-of-Chapter Problem Set

This page titled 10: MATLAB for Engineering Computation is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

10.1: Why Engineers Program

Why Engineers Program

MATLAB is introduced not as a separate coding topic, but as a tool for expressing engineering models clearly, scaling them efficiently, and evaluating them reproducibly.

By this point in the book, you can already:

- convert units correctly
- structure engineering solutions
- build spreadsheet models
- visualize nonlinear behavior

So why introduce MATLAB?

Because engineering systems grow.

Spreadsheets are excellent for structured tabular exploration. They work well when the model is relatively compact, the formulas remain visible, and the number of cases is manageable.

But as models expand, spreadsheets become:

Why Larger Models Outgrow Spreadsheets

- hard to audit
- difficult to scale
- error-prone with repeated copying
- limited in automation and logic

MATLAB allows engineers to express mathematical models more directly and automate repeated evaluation without relying on long chains of copied cells.

That matters because engineering computation is not just about getting answers. It is about building models that can be checked, reused, modified, and extended as the problem becomes more complex.

Programming is not simply “learning to code.”

It is writing engineering logic in a reproducible form.

Key Takeaway

Engineers program because larger models require automation, scalability, and reproducibility. MATLAB makes it possible to express engineering logic clearly and evaluate complex systems more efficiently than spreadsheets alone.

This page titled 10.1: Why Engineers Program is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

10.2: MATLAB as Structured Engineering

10.2 MATLAB as Structured Engineering

A well-written MATLAB script mirrors the engineering workflow you already know:

How MATLAB script sections align with the engineering workflow.

Script section	Engineering step	Example
Header comment block	Identity + purpose	`% Cable Tension Model`
Inputs section (%%)	Define	`m = 1000; % kg`
Model section	Model	`T = (m*g)./(2*sind(theta));`
Output section	Analyze + Interpret	`plot(theta, T)`

If a script does not reflect this structure, it is poorly engineered — the same standard as a spreadsheet where inputs are buried in formulas, or a handwritten calculation that jumps from problem to answer without a Define step.

This page titled 10.2: MATLAB as Structured Engineering is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

10.3: Script Basics

10.3 Script Basics — Structure and Syntax

The Standard Header

Every MATLAB script in this course starts with three cleanup commands and a comment block:

clear clc close all % Your Name % ENGR 110 Programming Project % Date

`clear` removes all workspace variables. `clc` clears the command window. `close all` closes open figure windows. Together they ensure every run starts from a clean state — a reproducible result. A script that behaves differently depending on what you ran before is not an engineering tool.

Variables, Comments, and Semicolons

m = 1000; % mass, kg g = 9.81; % gravitational acceleration, m/s² theta = 30; % cable angle, degrees

The semicolon suppresses output to the command window. Without it, every assignment prints its value. Units belong in comments, not variable names — write `m = 1000; % kg`, not `m_kg = 1000`.

Sections

%% marks a section. Sections let you run one block at a time (Ctrl+Enter) and document the structure:

%% Define Inputs m = 1000; % kg g = 9.81; % m/s² theta = 30; % degrees %% Model T = (m*g) / (2*sind(theta)); %% Output fprintf('Cable tension = %.2f N\n', T);

This page titled 10.3: Script Basics is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

10.4: Four-Step Build

10.4 From Single Calculation to Parameter Sweep — A Four-Step Build

Step 1 — Single Calculation

clear; clc; close all; % Cable Tension — Single Case | m=1000 kg, theta=30° %% Inputs m = 1000; % kg g = 9.81; % m/s² theta = 30; % degrees %% Model T = (m*g) / (2*sind(theta)); %% Output fprintf('Cable tension at %.0f degrees = %.2f N\n', theta, T);

Output: `Cable tension at 30 degrees = 9810.00 N` — matches Worked Example 6.1. But this only tells us about one angle.

Step 2 — Define a Vector

Replace the single angle with a range using the colon operator `start:step:end`:

theta = 10:5:80; % angles from 10° to 80° in 5° steps → [10, 15, 20, ..., 80]

Step 3 — Element-Wise Operations

Apply the formula to every element of the vector using the dot operator:

T = (m*g) ./ (2*sind(theta)); % ./ divides each element individually

> **⚠ Watch Out — The Dot Matters**
>
> `T = (m*g) / (2*sind(theta))` — matrix division. Will fail or produce a wrong scalar.
>
> `T = (m*g) ./ (2*sind(theta))` — element-wise division. Correct.
>
> Always use `./ .* .^` when operating on vectors. MATLAB sometimes errors on the wrong version, but sometimes silently gives a wrong answer.

Step 4 — Plot

plot(theta, T) xlabel('Angle (degrees)') ylabel('Tension (N)') title('Cable Tension vs. Angle — 1000 kg Load') grid on

Complete Script

clear; clc; close all; % Cable Tension Model — Parameter Sweep % m = 1000 kg, theta swept 10° to 80° %% Inputs m = 1000; % kg g = 9.81; % m/s² theta = 10:5:80; % degrees %% Model T = (m*g) ./ (2*sind(theta)); %% Plot figure(1) plot(theta, T, 'bo-') xlabel('Angle (degrees)') ylabel('Tension (N)') title('Cable Tension vs. Angle — 1000 kg Load') grid on

This page titled 10.4: Four-Step Build is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

10.5: Trigonometry in MATLAB

10.5 Trigonometry in MATLAB

MATLAB trigonometric functions for radian and degree inputs.

Function	Input	Use when
`sin(x), cos(x), tan(x)`	Radians	You have already converted to radians
`sind(x), cosd(x), tand(x)`	Degrees	Angles are in degrees — the more common case

`sin(30)` = sin(30 radians) ≈ −0.988. `sind(30)` = sin(30°) = 0.500.

Use `sind()`, `cosd()`, `tand()` whenever your angles are in degrees. The `d` suffix stands for "degrees." This is the most common trigonometry error in new MATLAB users.

This page titled 10.5: Trigonometry in MATLAB is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

10.6: Conditional Logic

10.6 Conditional Logic and Safety Checking

```
if condition

    % code to run if true
else

    % code to run if false
end
```

For automatic safety checking:

```
V = 0:1:20; R = 330;
P = (V.^2) / R;
if max(P) > 0.25

    fprintf('WARNING: Power %.3f W exceeds 0.25 W rating.\n', max(P))
else

    disp('All operating points within safe limit.')
end
```

The engineer does not need to scan numbers — the model enforces the constraint automatically.

> ✓ **Worked Example 10.1 — Power Safety Check with Plot**
>
> 330 Ω resistor rated 0.25 W — check safety across 0–20 V and plot with constraint line:

```
clear;
clc;
close all;
% Power Safety Check — R=330Ω, rating=0.25W
%% Inputs
R = 330; % ohms
P_max = 0.25; % W (resistor rating)
V = 0:1:20; % V
%% Model
P = (V.^2) / R;
%% Safety Check
if max(P) > P_max

    fprintf('WARNING: Max power = %.3f W at V = %.0f V\n', max(P), V(P==max(P)
else

    disp('All points within safe limit.')
end
%% Plot
figure(1)
plot(V, P, 'b-')
hold on
yline(P_max, 'r--', '0.25 W limit')
xlabel('Voltage (V)')
ylabel('Power (W)')
title('Power vs. Voltage — 330 Ω Resistor')
grid on
```

`yline()` draws a horizontal reference line — the rated limit becomes a visible constraint on the plot.

This page titled 10.6: Conditional Logic is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

10.7 Logical Indexing and Design Filtering

Logical indexing extracts only the values satisfying a condition — turning a model from a reporter into a design tool:

```
safe_V = V(P < 0.25); % voltages where power is safe
max_safe_V = max(safe_V); % highest safe voltage
fprintf('Max safe voltage: %.1f V\n', max_safe_V)
```

Instead of "what is power at 12 V?" this asks "what voltages are acceptable?" — the engineering design question.

For the cable tension model:

```
T_limit = 12000; % N (cable rating)
safe_angles = theta(T < T_limit); % angles where tension is OK
min_safe_angle = min(safe_angles); % tightest safe angle
fprintf('Minimum safe angle: %.1f degrees\n', min_safe_angle)
```

This directly answers the design question from Chapter 7: given a cable rating, what is the minimum allowable angle?

This page titled 10.7: Logical Indexing is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

10.8: Programming Project- Cable Car in MATLAB

10.8 The Programming Project — Cable Car in MATLAB

The Programming Project recreates the Excel Cable Car problem in MATLAB. Here is the MATLAB structure for the core cable car calculation.

```
clear;

clc;

close all;
% ENGR 110 Programming Project – Cable Car Velocity Model
% [Your Name] – [Date]
```

Distance vector

```
%% Cable Car Setup
distFromT1 = 0:5:500; % ft, distance from Tower 1 (101 values)
```

Nearest tower — string array and logical indexing

```
%% Nearest Tower
nearestTower = strings(length(distFromT1), 1);
nearestTower(distFromT1 < 250) = "1";
nearestTower(distFromT1 > 250) = "2";
nearestTower(distFromT1 == 250) = "MidPoint";
```

Distance from nearest tower

```
%% Distance from Nearest Tower
distNearT = zeros(length(distFromT1), 1);
distNearT(distFromT1 <= 250) = distFromT1(distFromT1 <= 250);
distNearT(distFromT1 > 250) = 500 - distFromT1(distFromT1 > 250);
```

Velocity — piecewise logical indexing

```
%% Velocity
velocity = zeros(length(distFromT1), 1);
% Near-tower equation (d ≤ 30 ft)
velocity(distNearT <= 30) = 2.425 + 0.00175 * distNearT(distNearT <= 30).^2;
% Mid-cable equation (d > 30 ft)
velocity(distNearT > 30) = 0.625 + 0.12*distNearT(distNearT > 30) - 0.00025*distN
```

Plot

```
%% Plot
figure(3)
plot(distFromT1, velocity, 'go')
xlabel('Distance from Tower 1 (ft)')
ylabel('Velocity (ft/s)')
title('Cable Car Velocity vs. Position')
xlim([0 500])
ylim([0 16])
grid on
```

Notice what logical indexing does here: the piecewise velocity equations are applied in two clean passes, each targeting exactly the right elements. The Excel version required a single IF formula handling all cases simultaneously — which was readable for two conditions but would become unwieldy for four or five. MATLAB's approach scales cleanly.

> 🔗 **Connection — Programming Project**
>
> The Programming Project asks you to complete Parts 1–3 of the walkthrough (data manipulation, formula trials, cable car model), produce the velocity vs. position plot with labeled axes, and write a technical memo comparing Excel and MATLAB for the same problem.
>
> The memo is an engineering comparison, not just a description. What does each tool do well? Where did you encounter limitations? Which would you use for a scaled-up version of this problem? Answering those questions draws directly on the comparison table in §10.1.
>
> Submission: `.m` script file + PDF memo to Canvas.

This page titled 10.8: Programming Project- Cable Car in MATLAB is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

10.9: Choosing the Right Tool

10.9 Choosing the Right Tool

When to use spreadsheets versus MATLAB for engineering models.

Use a spreadsheet when...	Use MATLAB when...
The model is simple (one equation, one sweep)	The model has piecewise logic or multiple interacting equations
The audience includes non-programmers who need to inspect inputs	You need to reproduce the run exactly with different data
You want visual formatting (colored cells, formatted tables)	The dataset is large (hundreds or thousands of rows)
A design selection comparison will be reviewed by others	You need embedded safety constraints that automatically flag violations

Neither tool is universally better. The engineers who are most effective are the ones who know which tool to reach for and why.

Key Takeaway

MATLAB complements spreadsheets. It excels at piecewise logic, large datasets, reproducible automation, and embedded safety constraints. The engineering structure — inputs, model, output, interpretation — is identical across both tools. Element-wise operators (`./`, `.*`, `.^`) and degree-based trig (`sind`, `cosd`) are the two most common syntax errors in new MATLAB users. Get those right and the rest follows from the engineering logic you already know.

This page titled 10.9: Choosing the Right Tool is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

10.10: Summary

Summary

- MATLAB complements spreadsheets — better for piecewise logic, large datasets, reproducibility, and embedded safety checks.
- Every script starts with `clear; clc; close all;` and a comment header. Sections (`%%`) organize the script into Define / Model / Output blocks.
- Semicolons suppress output. Comments (`%`) document units and intent. Units belong in comments, not variable names.
- The colon operator creates vectors: `theta = 10:5:80` creates 15 values from 10° to 80°.
- Element-wise operators (`./`, `.*`, `.^`) apply equations to every element. The dot is required.
- `sind()`, `cosd()`, `tand()` accept degrees. `sin()` requires radians. Use the `d` versions.
- Plots require `xlabel()`, `ylabel()`, `title()`, `grid on`. Same requirements as Chapter 9.
- Logical indexing (`V(P < 0.25)`) filters vectors to extract values satisfying a condition.

This page titled 10.10: Summary is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

10.11: End-of-Chapter Problem Set

End-of-Chapter Problem Set

Script Writing

1. Write a complete MATLAB script (with header) that computes cable tension at 45° for a 750 kg load and prints the result using `fprintf`.
2. Modify Problem 1 to sweep angle from 5° to 85° in 2° increments. Plot tension vs. angle with fully labeled axes and `grid on`.
3. Identify all errors in this line: `T = (m*g) / (2*sin(theta))` — assume `theta` is a vector in degrees.

Vectors and Operators

4. Define a voltage vector from 0 to 15 V in 0.5 V increments. Compute power for R = 220 Ω. Plot power vs. voltage with labeled axes.
5. For Problem 4, add an `if` statement that prints a warning if any power exceeds 0.5 W, showing the maximum value and the voltage at which it occurs.
6. Use logical indexing to find and print the maximum voltage that keeps power below 0.5 W for R = 220 Ω.

Design Applications

7. A cable is rated for 10,000 N. Using the tension model with m = 1000 kg, find and print the minimum safe cable angle using logical indexing.
8. Plot power vs. voltage for R = 100 Ω, 220 Ω, and 470 Ω on the same axes using `hold on`. Add a dashed line at P = 0.25 W using `yline`. Label axes and include a legend.
9. *[Cable Car prep]* Write MATLAB code that creates `distFromT1 = 0:5:500` and uses logical indexing to populate `distNearT`. Verify that `distNearT(51)` equals 250 (the midpoint).

Answers - click to expand

Script Writing

1. **Complete MATLAB script:**

```
%% Problem 1 - Cable Tension at One Angle
% ENGR 110
% Computes the tension in each cable for a symmetric suspended load.
clear; clc; close all;
% Given values
m = 750; % mass, kg
g = 9.81; % gravitational acceleration, m/s^2
theta = 45; % cable angle from horizontal, degrees
% Model
T = (m*g)/(2*sind(theta)); % tension in each cable, N
% Output
fprintf('Cable tension at %.1f degrees = %.1f N\n', theta, T);
fprintf('Cable tension = %.2f kN\n', T/1000);
```

   The expected result is $T \approx 5202.5\ \mathrm{N}$, or about $5.20\ \mathrm{kN}$ per cable.

2. **Angle sweep and plot:**

```
%% Problem 2 - Cable Tension Angle Sweep
% ENGR 110
% Sweeps cable angle and plots tension for a symmetric suspended load.
clear; clc; close all;
% Given values
m = 750; % mass, kg
g = 9.81; % gravitational acceleration, m/s^2
theta = 5:2:85; % cable angle from horizontal, degrees
% Model
T = (m*g)./(2*sind(theta)); % tension in each cable, N
% Plot
figure;
plot(theta, T, 'o-', 'LineWidth', 1.5);
grid on;
xlabel('Cable Angle, \theta (degrees)');
ylabel('Tension, T (N)');
title('Cable Tension vs. Cable Angle - 750 kg Load');
```

   The key changes are using a vector for `theta`, using `sind` for degrees, and using element-wise division `./` so MATLAB computes one tension value for each angle.

3. **Errors in `T = (m*g) / (2*sin(theta))`:**

- `theta` is in degrees, but `sin()` expects radians.
- Use `sind(theta)` for degrees, or convert first using `sin(deg2rad(theta))`.
- Because `theta` is a vector, the division must be element-wise: use `./`, not `/`.

Corrected line:

```
T = (m*g)./(2*sind(theta));
```

Vectors and Operators

4. **Power vs. voltage for $R = 220\ \Omega$:**

```
%% Problem 4 - Power vs Voltage
% Computes resistor power for a voltage sweep.
clear; clc; close all;
V = 0:0.5:15; % voltage, V
R = 220; % resistance, ohms
P = V.^2/R; % power, W
figure;
plot(V, P, 'o-', 'LineWidth', 1.5);
grid on;
xlabel('Voltage, V (V)');
ylabel('Power, P (W)');
title('Power vs. Voltage - 220 Ohm Resistor');
```

The operator `.^` is required because `V` is a vector and each voltage value must be squared separately.

5. **Add an `if` statement warning:**

```
%% Problem 5 - Power Warning
% Adds a warning if resistor power exceeds 0.5 W.
clear; clc; close all;
V = 0:0.5:15; % voltage, V
R = 220; % resistance, ohms
P = V.^2/R; % power, W
[maxP, idxMax] = max(P);
V_at_maxP = V(idxMax);
if any(P > 0.5)
fprintf('WARNING: Power exceeds 0.5 W.\n');
fprintf('Maximum power = %.3f W at V = %.1f V\n', maxP, V_at_maxP);
else
fprintf('Power remains below 0.5 W for all voltages tested.\n');
end
figure;
plot(V, P, 'o-', 'LineWidth', 1.5);
grid on;
xlabel('Voltage, V (V)');
ylabel('Power, P (W)');
title('Power vs. Voltage - 220 Ohm Resistor');
```

The function `any(P > 0.5)` checks whether at least one value in the power vector exceeds the limit.

6. **Logical indexing for maximum safe voltage:**

```
%% Problem 6 - Maximum Safe Voltage
% Uses logical indexing to find the maximum voltage below the power limit
clear; clc; close all;
V = 0:0.5:15; % voltage, V
R = 220; % resistance, ohms
P = V.^2/R; % power, W
safe = P <= 0.5;
V_safe_max = max(V(safe));
P_at_safe_max = P(V == V_safe_max);
fprintf('Maximum safe voltage = %.1f V\n', V_safe_max);
fprintf('Power at that voltage = %.3f W\n', P_at_safe_max);
```

For the 0.5 V increment sweep, the maximum listed safe voltage is $10.0\ \text{V}$, since $P = 10^2/220 \approx 0.455\ \text{W}$. The exact crossover is $V = \sqrt{(0.5)(220)} \approx 10.49\ \text{V}$.

Design Applications

7. **Minimum safe cable angle for a 10,000 N cable rating:**

```
%% Problem 7 - Minimum Safe Cable Angle
% Finds the minimum angle where cable tension is within rating.
clear; clc; close all;
m = 1000; % mass, kg
g = 9.81; % gravitational acceleration, m/s^2
rating = 10000; % cable rating, N
theta = 1:0.1:89; % angle sweep, degrees
T = (m*g)./(2*sind(theta));
safe = T <= rating;
theta_min_safe = min(theta(safe));
T_at_min_safe = T(theta == theta_min_safe);
fprintf('Minimum safe cable angle = %.1f degrees\n', theta_min_safe);
fprintf('Tension at this angle = %.1f N\n', T_at_min_safe);
```

The exact boundary is found from $\sin\theta = mg/(2T)$. With $m = 1000$ kg and $T = 10000$ N, $\theta \approx 29.4$ °. With the 0.1° sweep above, the minimum safe listed angle is about 29.4 °.

8. **Power curves for three resistors:**

```
%% Problem 8 - Multi-Curve Power Plot
% Plots power vs voltage for three resistor values.
clear; clc; close all;
V = 0:0.5:15; % voltage, V
R_values = [100 220 470]; % resistance, ohms
figure;
hold on;
for k = 1:length(R_values)
R = R_values(k);
P = V.^2/R;
plot(V, P, 'LineWidth', 1.5);
end
yline(0.25, '--', '0.25 W limit', 'LineWidth', 1.2);
hold off;
grid on;
xlabel('Voltage, V (V)');
ylabel('Power, P (W)');
title('Power vs. Voltage for Three Resistor Values');
legend('R = 100 \Omega', 'R = 220 \Omega', 'R = 470 \Omega', '0.25 W limi
```

This graph makes it clear that lower resistance produces higher power at the same voltage. The $100\ \Omega$ resistor exceeds the 0.25 W limit first, followed by $220\ \Omega$, then $470\ \Omega$.

9. **Cable Car prep using logical indexing:**

```
%% Problem 9 - Cable Car Distance from Nearest Tower
% Creates distance from tower 1 and calculates distance to nearest tower.
clear; clc; close all;
distFromT1 = 0:5:500; % distance from tower 1, ft
spanLength = 500; % distance between towers, ft
distNearT = zeros(size(distFromT1));
% First half of span: nearest tower is tower 1
leftHalf = distFromT1 <= spanLength/2;
distNearT(leftHalf) = distFromT1(leftHalf);
% Second half of span: nearest tower is tower 2
rightHalf = distFromT1 > spanLength/2;
distNearT(rightHalf) = spanLength - distFromT1(rightHalf);
% Verification
fprintf('distFromT1(51) = %.0f ft\n', distFromT1(51));
fprintf('distNearT(51) = %.0f ft\n', distNearT(51));
```

Because `distFromT1 = 0:5:500`, element 51 corresponds to 250 ft. The nearest tower distance at the midpoint is also 250 ft, so `distNearT(51)` should equal 250.

This page titled 10.11: End-of-Chapter Problem Set is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

11: CIRCUIT FUNDAMENTALS

Circuit Fundamentals

Circuits are energy systems governed by two conservation laws: charge (KCL) and energy (KVL). Every circuit analysis technique traces back to these two principles.

Learning Objectives

- Describe the physical meaning of voltage, current, and resistance using definitions and analogy.
- Apply Ohm's Law to compute unknown quantities in a single-resistor circuit.
- Calculate power dissipation and evaluate component safety.
- Analyze series circuits: equivalent resistance, current, and voltage across each component.
- Analyze parallel circuits: equivalent resistance, branch currents, and total current.
- Apply Kirchhoff's Current Law and Voltage Law conceptually and numerically.
- Solve a simple LED current-limiting resistor design problem.
- Identify the purpose of Falstad's Circuit Simulator and what the Circuit Design Project requires.

11.1: Circuits as Energy Systems
11.2: Physical Meaning
11.3: Ohm’s Law
11.4: Power
11.5: Schematic Symbols
11.6: Series Circuits
11.7: Parallel Circuits
11.8: Series vs Parallel Comparison
11.9: Kirchhoff's Laws
11.10: Power in Networks
11.11: Mixed Circuits
11.12: LED Design
11.13: Circuit Simulator
11.14: Summary
11.15: End-of-Chapter Problem Set

This page titled 11: Circuit Fundamentals is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

11.1: Circuits as Energy Systems

11.1 Circuits as Energy Systems

An electrical circuit is a controlled path for energy transfer — from a source to a load, through a closed conductive loop. Every circuit has three fundamental elements:

- **Source** — provides electrical energy (battery, power supply, generator)
- **Path** — a closed conductive loop that allows charge to flow
- **Load** — consumes the energy (resistor, LED, motor, lightbulb)

The four quantities used to track energy in electrical systems:

Fundamental electrical quantities used to describe circuit behavior.

Quantity	Symbol	Unit	What it describes
Voltage	V	Volts (V)	Energy per unit charge — the "pressure" that drives current
Current	I	Amperes (A)	Rate of charge flow — coulombs per second
Resistance	R	Ohms (Ω)	Opposition to current flow — converts electrical energy to heat
Power	P	Watts (W)	Rate of energy transfer — joules per second

Two conservation principles govern every circuit: **charge conservation** (KCL) — charge cannot accumulate at a node; and **energy conservation** (KVL) — energy supplied equals energy dissipated around any closed loop.

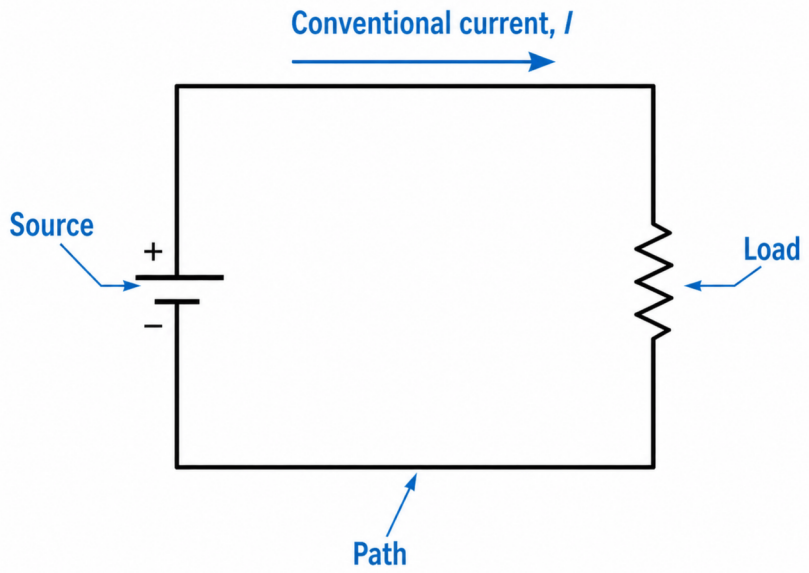


Figure 11.1.1: A simple closed circuit showing a battery as the source, wires as the conductive path, and a resistor as the load. Conventional current, I, flows from the positive terminal of the source through the load and returns to the negative terminal. (Copyright; Jonathan Compton, AI-assisted original figure, CC BY-NC 4.0)

This page titled 11.1: Circuits as Energy Systems is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

11.2: Physical Meaning

11.2 Voltage, Current, and Resistance — Physical Meaning

Voltage

Voltage is electric potential difference — energy per unit charge between two points:

$$[V] = \frac{\text{joules}}{\text{coulomb}} = \frac{\text{J}}{\text{C}}$$

The most useful analogy is **water pressure**. Voltage is like the pressure difference between two points in a pipe system. Higher voltage = stronger driving force. Voltage is always measured *between two points* — "the voltage at point A" requires a reference (typically ground at 0 V).

Current

Current is the rate of charge flow:

$$[I] = \frac{\text{C}}{\text{s}} = \text{A}$$

In the water analogy: current is the flow rate (gallons per minute). Current only flows in a **closed loop** — break the loop and current stops everywhere instantly.

Resistance

Resistance opposes current and converts electrical energy to heat:

$$[R] = \frac{\text{V}}{\text{A}} = \Omega$$

In the water analogy: resistance is the narrowness or roughness of the pipe. Important: resistance does not "use up" voltage — it converts electrical energy to thermal energy. The voltage across a resistor represents energy transferred to heat per unit charge.

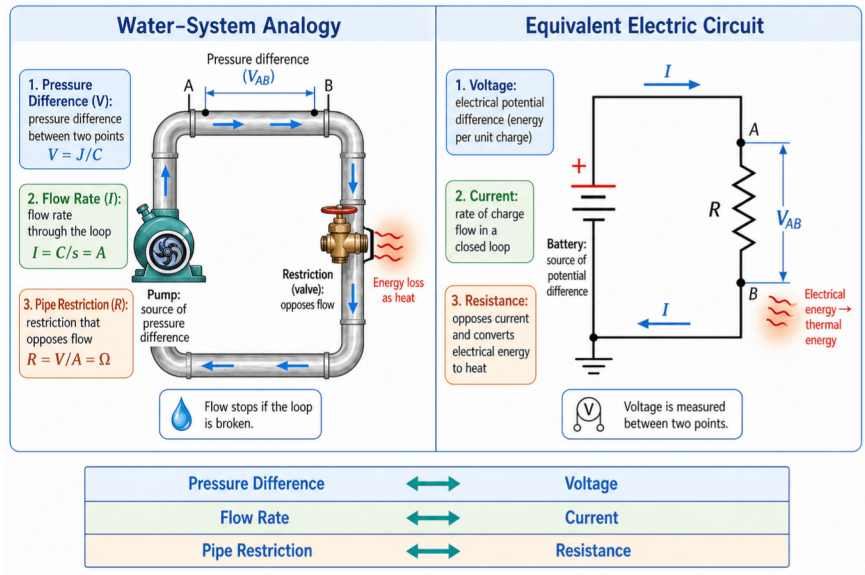


Figure 11.2.1: Water-system analogy for voltage, current, and resistance. Pressure difference corresponds to voltage, flow rate corresponds to current, and pipe restriction corresponds to resistance; in both systems, energy is transferred or lost as heat across the resisting element. (Copyright; Jonathan Compton, AI-assisted original figure, CC BY-NC 4.0)

This page titled 11.2: Physical Meaning is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

11.3: Ohm's Law

11.3 Ohm's Law

$$V = IR \qquad I = \frac{V}{R} \qquad R = \frac{V}{I}$$

For fixed R: doubling V doubles I (linear). For fixed V: doubling R halves I (inverse). Ohm's Law is linear — but power introduces nonlinearity (§11.4).

✓ Worked Example 11.1 — Basic Ohm's Law

9 V source, 220 Ω resistor. Find the current.

$$I = \frac{V}{R} = \frac{9\ \text{V}}{220\ \Omega} = 0.0409\ \text{A} = 40.9\ \text{mA}$$

Interpret: 40.9 mA is typical for small electronics — an LED or signal circuit. Unit check: V/Ω = A ✓

▶ Try It — Ohm's Law Practice

Compute current for each case. Show units at every step.

a) $V = 12\ \text{V}$, $R = 330\ \Omega$ b) $V = 5\ \text{V}$, $R = 1000\ \Omega$ c) $V = 3.3\ \text{V}$, $R = 470\ \Omega$

Answers - click to expand

a. $I = V/R = 12\ \text{V}/330\ \Omega = 0.0364\ \text{A} = 36.4\ \text{mA}$

Unit check: $\text{V}/\Omega = \text{A}$.

b. $I = V/R = 5\ \text{V}/1000\ \Omega = 0.00500\ \text{A} = 5.00\ \text{mA}$

Unit check: $\mathrm{V}/\Omega = \mathrm{A}$.

c. $I = V/R = 3.3\ \mathrm{V}/470\ \Omega = 0.00702\ \mathrm{A} = 7.02\ \mathrm{mA}$

Unit check: $\mathrm{V}/\Omega = \mathrm{A}$.

This page titled 11.3: Ohm's Law is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

11.4: Power

11.4 Electrical Power

$$P = VI = I^2R = \frac{V^2}{R}$$

The critical fact: power grows with the *square* of voltage. Doubling voltage **quadruples** power. A component safe at 6 V may be at its limit at 9 V and destroyed at 12 V. Power is always the first safety check in circuit design.

✓ Worked Example 11.2 — Power Safety Check

12 V source, 330 Ω resistor rated at 0.25 W. Safe?

$$P = \frac{V^2}{R} = \frac{(12)^2}{330} = \frac{144}{330} = 0.436 \text{ W}$$

Interpret: 0.436 W exceeds the 0.25 W rating. The resistor will overheat and fail. Options: higher-rated resistor (≥0.5 W), larger resistance, or lower supply voltage.

⚠ Watch Out — Check Every Component Individually

In multi-component circuits, total power and branch power are different. Always compute and check power for *each* component. Students consistently check the safe-looking resistor and miss the one at risk.

This page titled 11.4: Power is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

11.5: Schematic Symbols

11.5 Reading Circuit Schematics

Circuit schematics use standard symbols. The symbols used in the Circuit Design Project are:

Common circuit schematic symbols and their functions.

Component	Symbol description	What it does
DC Voltage Source	Long/short parallel lines (+ and –)	Provides constant voltage
Resistor	Zigzag line (US) or rectangle (IEC)	Opposes current, dissipates energy as heat
Ammeter	Circle with A	Measures current — placed **in series**
Voltmeter	Circle with V	Measures voltage — placed **in parallel** across a component
Ground	Line to downward triangle	Reference point (0 V)
LED	Triangle with lines (directional)	Light-emitting diode — only conducts one direction
Switch (SPST)	Line with a gap and lever	Opens or closes the current path

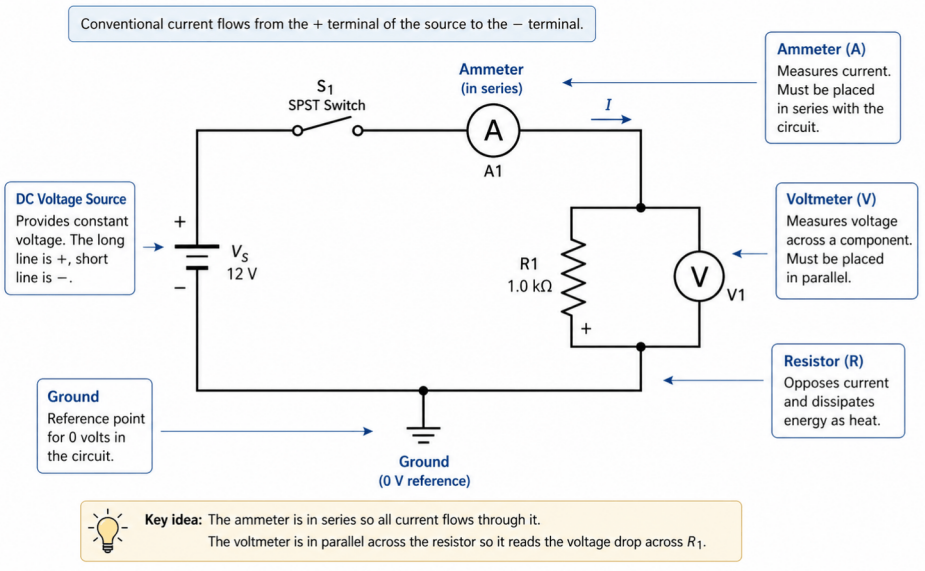


Figure 11.5.1: Labeled circuit schematic showing a DC voltage source, SPST switch, ammeter, resistor, voltmeter, and ground reference. The ammeter is placed in series so all circuit current flows through it, while the voltmeter is placed in parallel across the resistor to measure the voltage drop. (Copyright; Jonathan Compton, AI-assisted original figure, CC BY-NC 4.0)

This page titled 11.5: Schematic Symbols is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

11.6: Series Circuits

11.6 Series Circuits

In a series circuit, all components share a single current path. The same current flows through every element.

$$R_{\text{eq}} = R_1 + R_2 + \cdots \qquad I_{\text{total}} = \frac{V_s}{R_{\text{eq}}} \qquad V_k = I \cdot R_k$$

Voltage divider: $V_k = V_s \cdot \dfrac{R_k}{R_{\text{eq}}}$ — voltage distributes proportionally to resistance. Higher R → larger drop.

✓ Worked Example 11.3 — Series Circuit

$R_1 = 220\ \Omega$, $R_2 = 330\ \Omega$, $V_s = 12\ \text{V}$

$$R_{\text{eq}} = 220 + 330 = 550\ \Omega$$

$$I = \frac{12}{550} = 21.82\ \text{mA}$$

$$V_1 = (0.02182)(220) = 4.80\ \text{V} \qquad V_2 = (0.02182)(330) = 7.20\ \text{V}$$

KVL check: $4.80 + 7.20 = 12.00\ \text{V}$ ✓

Power: $P_1 = (0.02182)^2(220) = 0.105\ \text{W}$, $P_2 = (0.02182)^2(330) = 0.157\ \text{W}$ — both below 0.25 W rating ✓

This page titled 11.6: Series Circuits is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

11.7: Parallel Circuits

11.7 Parallel Circuits

In a parallel circuit, all components connect across the same two nodes. Every branch sees the same voltage.

$$\frac{1}{R_{\text{eq}}} = \frac{1}{R_1} + \frac{1}{R_2} + \cdots \qquad I_k = \frac{V_s}{R_k} \qquad I_{\text{total}} = I_1 + I_2 + \cdots$$

Adding resistors in parallel *decreases* equivalent resistance — more paths, less total opposition. This is why plugging more appliances into a household circuit increases the load on the supply.

✓ Worked Example 11.4 — Parallel Circuit

$R_1 = 220\ \Omega$, $R_2 = 330\ \Omega$, $V_s = 9\ \text{V}$

$$\frac{1}{R_{\text{eq}}} = \frac{1}{220} + \frac{1}{330} = \frac{5}{660} \quad \Rightarrow \quad R_{\text{eq}} = 132\ \Omega$$

$$I_1 = \frac{9}{220} = 40.9\ \text{mA} \qquad I_2 = \frac{9}{330} = 27.3\ \text{mA} \qquad I_{\text{total}} = 68.2\ \text{mA}$$

KCL check: $40.9 + 27.3 = 68.2\ \text{mA}$ ✓

This page titled 11.7: Parallel Circuits is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

11.8: Series vs Parallel Comparison

11.8 Series vs. Parallel — Comparison

Comparison of series and parallel circuit behavior.

Property	Series	Parallel
Current	**Same** everywhere	**Splits** among branches
Voltage	**Splits** across elements	**Same** across all branches
Equivalent R	**Increases** (sum)	**Decreases** (reciprocal sum)
Adding more resistors	Reduces total current	Increases total current
One element fails	Breaks entire circuit	Other branches still function

The last row explains why household circuits are wired in parallel: if one lamp fails in a series string, every lamp goes dark.

This page titled 11.8: Series vs Parallel Comparison is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

11.9: Kirchhoff's Laws

11.9 Kirchhoff's Laws

Kirchhoff's Current Law (KCL)

> *The sum of currents entering any node equals the sum leaving.*
>
> $$\sum I_{\text{in}} = \sum I_{\text{out}}$$

This is conservation of charge. KCL is how you verify parallel circuit calculations — if branch currents don't add to the total, you have an error.

Kirchhoff's Voltage Law (KVL)

> *The sum of voltages around any closed loop equals zero.*
>
> $$\sum V = 0 \qquad (\text{any closed loop})$$

This is conservation of energy. KVL is how you verify series circuit calculations — and how ENGR 230 solves complex multi-loop circuits.

This page titled 11.9: Kirchhoff's Laws is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

11.10: Power in Networks

11.10 Power in Multi-Component Circuits

Total power supplied equals total power dissipated:

$$P_{\text{source}} = \sum P_{\text{loads}}$$

For Worked Example 11.3: $P_{\text{source}} = 12 \times 0.02182 = 0.262\ \text{W}$. $P_1 + P_2 = 0.105 + 0.157 = 0.262\ \text{W}$ ✓

Key rule: in series, the higher-resistance element dissipates more power (larger voltage drop). In parallel, the lower-resistance element dissipates more power (larger current). Check the right component.

This page titled 11.10: Power in Networks is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

11.11: Mixed Circuits

11.11 Mixed Circuits

Analysis strategy: systematic reduction. Identify the simplest subgroup (usually a parallel pair), combine it, redraw, and repeat until one equivalent resistance remains. Then work backwards.

✓ Worked Example 11.5 — Mixed Circuit

$R_1 = 100\ \Omega$ in series with $R_2 = 220\ \Omega \| R_3 = 330\ \Omega$. Source = 9 V.

$$R_{23} = \frac{220 \times 330}{220 + 330} = \frac{72600}{550} = 132\ \Omega$$

$$R_{\text{eq}} = 100 + 132 = 232\ \Omega \quad \Rightarrow \quad I_{\text{total}} = \frac{9}{232} = 38.8\ \text{mA}$$

$$V_{23} = 0.0388 \times 132 = 5.12\ \text{V}$$

$$I_2 = \frac{5.12}{220} = 23.3\ \text{mA} \qquad I_3 = \frac{5.12}{330} = 15.5\ \text{mA}$$

KCL check: $23.3 + 15.5 = 38.8\ \text{mA}$ ✓

This page titled 11.11: Mixed Circuits is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

11.12: LED Design

11.12 Circuit Design — LED Current Limiting

An LED requires a specific forward current (typically 10–30 mA) and has a forward voltage drop (V_F ≈ 1.8–3.3 V). Without a series resistor, the LED draws excessive current and burns out. The resistor's job: drop the excess supply voltage at the target current.

✓ Worked Example 11.6 — LED Resistor Design

Red LED: $V_F = 2.0\ \text{V}$, $I_F = 20\ \text{mA}$. Supply = 9 V. Find the current-limiting resistor.

Given values and unknown resistor value for the LED current-limiting design.

Symbol	Quantity	Value	Units
V_s	Supply	9	V
V_F	LED forward voltage	2.0	V
I_F	Target current	20 mA = 0.020	A
R	Resistor value	*unknown*	Ω

$$V_R = V_s - V_F = 9 - 2.0 = 7.0\ \text{V}$$

$$R = \frac{V_R}{I_F} = \frac{7.0\ \text{V}}{0.020\ \text{A}} = 350\ \Omega$$

Standard value: Round up to **390 Ω** (next E12 value above 350 Ω). Always round up to keep current at or below target.

Power check: $P = I_F^2 \cdot R = (0.020)^2 \times 390 = 0.156\ \text{W}$ — below 0.25 W rating ✓

Communicate: A 390 Ω, 0.25 W resistor limits LED current to ~17.9 mA. The LED will be slightly below maximum brightness but operating safely within rating.

This page titled 11.12: LED Design is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

11.13: Circuit Simulator

11.13 Circuit Simulation — Falstad Circuit Simulator

The Circuit Design Project uses **Falstad's Circuit Simulator** at falstad.com/circuit — a free browser-based tool that builds and simulates circuits without hardware.

It displays current flow visually as animated dots (proportional to current magnitude) and allows voltmeters and ammeters to update in real time as you change component values. Right-click any component to edit its value and immediately observe the effect.

🔗 Connection — Circuit Design Project

The Circuit Design Project uses Falstad's Circuit Simulator to walk through eight tasks that directly apply this chapter:

- **Task 1:** Validate $V = IR$ — current source, sweep resistance, plot V vs R in Excel
- **Task 2:** Validate $I = V/R$ — voltage source, sweep resistance, plot I vs 1/R in Excel
- **Tasks 3–5:** Combine resistors in series, parallel, and mixed configurations into equivalent circuits
- **Task 6:** Verify KCL numerically at a multi-branch node
- **Task 7:** Lamp circuit — observe power behavior as voltage changes
- **Task 8:** LED circuit with switches — observe current changes as branches are added or removed

Submission steps: (1) Complete all tasks, inserting circuit screenshots and Excel charts into the workbook document. (2) **Delete the first 3 pages** of the workbook (the instruction pages) — keep only the Activity section with your work. (3) Add your name at the top of the Activity section. (4) Save as PDF and submit to Canvas. Tasks 5, 7, and 8 are weighted 1.5×.

Key Takeaway

Circuits are energy systems governed by conservation of charge (KCL) and energy (KVL). Voltage drives current through resistance via Ohm's Law; power grows with the square of voltage — always check component ratings. Series topologies add resistance and divide voltage; parallel topologies reduce resistance and divide current. The structured analysis workflow — identify topology, compute equivalent resistance, find total current, then branch values, then power — applies to circuits of any complexity.

This page titled 11.13: Circuit Simulator is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

11.14: Summary

Summary

- A circuit moves energy from source to load through a closed loop, governed by KCL and KVL.
- Ohm's Law: $V = IR$. Power: $P = VI = I^2R = V^2/R$. Power grows quadratically with voltage.
- Series: current same everywhere; voltage divides; $R_{\text{eq}} = \sum R_k$.
- Parallel: voltage same across all branches; current divides; $1/R_{\text{eq}} = \sum 1/R_k$.
- KCL: currents in = currents out at any node. KVL: voltages sum to zero around any closed loop.
- Mixed circuits: systematically reduce parallel pairs, then combine with series elements.
- LED design: $R = (V_s - V_F)/I_F$; round up to next standard value; verify power rating.
- Falstad Circuit Simulator (falstad.com/circuit) is used in the Circuit Design Project for Tasks 1–8.

This page titled 11.14: Summary is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

11.15: End-of-Chapter Problem Set

End-of-Chapter Problem Set

Ohm's Law and Power

1. Compute current and power for a 9 V source, 470 Ω resistor. Is a 0.25 W resistor safe?
2. 12 V source, 680 Ω resistor. Compute current in mA and power in mW.
3. At what voltage does a 330 Ω resistor reach its 0.25 W power rating?

Series Circuits

4. Two resistors (150 Ω, 300 Ω) in series on 12 V. Find R_{eq}, current, voltage across each, power in each. Verify KVL.
5. Three resistors (100 Ω, 220 Ω, 470 Ω) in series on 9 V. Find voltage across each and verify they sum to 9 V.

Parallel Circuits

6. Two resistors (220 Ω, 470 Ω) in parallel on 12 V. Find R_{eq}, branch currents, total current. Verify KCL.
7. For Problem 6, compute power in each branch. Which dissipates more, and why?

Mixed Circuits and Design

8. $R_1 = 100\ \Omega$ in series with ($R_2 = 220\ \Omega \parallel R_3 = 330\ \Omega$) on 9 V. Find total resistance, total current, voltage across the parallel pair, and branch currents. Verify KCL.
9. Design a current-limiting resistor for a green LED ($V_F = 3.2\ \text{V}$, $I_F = 15\ \text{mA}$) from a 5 V USB supply. Find ideal value, select nearest standard (100, 120, 150, 180, 220 Ω), and check power.
10. A 12 V source powers two LEDs in series (each $V_F = 2.0\ \text{V}$, both requiring 20 mA). Design the current-limiting resistor and check the power rating required.
11. *[Challenge]* Explain why KVL holds in Problem 4 and KCL holds in Problem 6. Connect each to the physical conservation law it represents.

Answer Key - click to expand

Ohm's Law and Power

1. **9 V source, 470 Ω resistor:**

$$P = VI = (9\ \text{V})(0.0191\ \text{A}) = 0.172\ \text{W}$$

Interpretation: A $0.25\ \text{W}$ resistor is safe because $0.172\ \text{W} < 0.25\ \text{W}$. However, the resistor is using about 69% of its rating, so a higher rating may be preferred if heat or continuous operation is a concern.

2. **12 V source, 680 Ω resistor:**

   $$P = VI = (12\ \text{V})(0.0176\ \text{A}) = 0.212\ \text{W} = 212\ \text{mW}$$

   Answer The current is approximately $17.6\ \text{mA}$, and the power is approximately $212\ \text{mW}$.

3. **Voltage where a 330 Ω resistor reaches 0.25 W:**

   $$V = \sqrt{(0.25\ \text{W})(330\ \Omega)} = \sqrt{82.5} = 9.08\ \text{V}$$

   Answer A $330\ \Omega$ resistor reaches $0.25\ \text{W}$ at approximately $9.08\ \text{V}$.

Series Circuits

4. **150 Ω and 300 Ω in series on 12 V:**

   $$I = V/R_{\text{eq}} = 12\ \text{V}/450\ \Omega = 0.0267\ \text{A} = 26.7\ \text{mA}$$

   $$V_1 = IR_1 = (0.0267\ \text{A})(150\ \Omega) = 4.00\ \text{V}$$

   $$V_2 = IR_2 = (0.0267\ \text{A})(300\ \Omega) = 8.00\ \text{V}$$

   $$P_1 = I^2R_1 = (0.0267\ \text{A})^2(150\ \Omega) = 0.107\ \text{W}$$

   $$P_2 = I^2R_2 = (0.0267\ \text{A})^2(300\ \Omega) = 0.213\ \text{W}$$

   KVL check: $V_1 + V_2 = 4.00\ \text{V} + 8.00\ \text{V} = 12.00\ \text{V}$. This matches the source voltage.

5. **100 Ω, 220 Ω, and 470 Ω in series on 9 V:**

   $$I = V/R_{\text{eq}} = 9\ \text{V}/790\ \Omega = 0.0114\ \text{A} = 11.4\ \text{mA}$$

   $$V_1 = IR_1 = (0.0114\ \text{A})(100\ \Omega) = 1.14\ \text{V}$$

   $$V_2 = IR_2 = (0.0114\ \text{A})(220\ \Omega) = 2.51\ \text{V}$$

   $$V_3 = IR_3 = (0.0114\ \text{A})(470\ \Omega) = 5.35\ \text{V}$$

   KVL check: $1.14\ \text{V} + 2.51\ \text{V} + 5.35\ \text{V} = 9.00\ \text{V}$, allowing for rounding.

Parallel Circuits

6. **220 Ω and 470 Ω in parallel on 12 V:**

   $$R_{\text{eq}} = 149.9\ \Omega$$

   In a parallel circuit, each branch has the same voltage, so each resistor has $12\ \text{V}$ across it.

   $$I_{220} = V/R_{220} = 12\ \text{V}/220\ \Omega = 0.0545\ \text{A} = 54.5\ \text{mA}$$

   $$I_{470} = V/R_{470} = 12\ \text{V}/470\ \Omega = 0.0255\ \text{A} = 25.5\ \text{mA}$$

$$I_{\text{total}} = I_{220} + I_{470} = 54.5\ \text{mA} + 25.5\ \text{mA} = 80.0\ \text{mA}$$

KCL check: The current entering the parallel node equals the sum of the branch currents: $I_{\text{total}} = I_{220} + I_{470}$.

7. **Power in each branch:**

   $$P_{470} = V^2/R_{470} = (12\ \text{V})^2/470\ \Omega = 0.306\ \text{W}$$

   Interpretation: The $220\ \Omega$ resistor dissipates more power because both resistors have the same voltage across them, and $P = V^2/R$. With fixed voltage, the smaller resistance dissipates more power.

Mixed Circuits and Design

8. $R_1 = 100\ \Omega$ **in series with** $R_2 = 220\ \Omega$ **parallel** $R_3 = 330\ \Omega$**, on 9 V:**

   $$R_{23} = (R_2R_3)/(R_2 + R_3) = (220\ \Omega)(330\ \Omega)/(220\ \Omega + 330\ \Omega) = 132\ \Omega$$

   Total resistance:

   $$R_{\text{total}} = R_1 + R_{23} = 100\ \Omega + 132\ \Omega = 232\ \Omega$$

   Total current:

   $$I_{\text{total}} = V/R_{\text{total}} = 9\ \text{V}/232\ \Omega = 0.0388\ \text{A} = 38.8\ \text{mA}$$

   Voltage across R_1:

   $$V_1 = I_{\text{total}}R_1 = (0.0388\ \text{A})(100\ \Omega) = 3.88\ \text{V}$$

   Voltage across the parallel pair:

   $$V_{23} = 9\ \text{V} - 3.88\ \text{V} = 5.12\ \text{V}$$

   Branch currents:

   $$I_2 = V_{23}/R_2 = 5.12\ \text{V}/220\ \Omega = 0.0233\ \text{A} = 23.3\ \text{mA}$$

   $$I_3 = V_{23}/R_3 = 5.12\ \text{V}/330\ \Omega = 0.0155\ \text{A} = 15.5\ \text{mA}$$

   KCL check: $I_2 + I_3 = 23.3\ \text{mA} + 15.5\ \text{mA} = 38.8\ \text{mA}$, which matches I_{total}, allowing for rounding.

9. **Green LED current-limiting resistor:**

   The resistor must drop the remaining voltage:

   $$V_R = V_s - V_F = 5\ \text{V} - 3.2\ \text{V} = 1.8\ \text{V}$$

   Ideal resistor:

   $$R = V_R/I_F = 1.8\ \text{V}/0.015\ \text{A} = 120\ \Omega$$

   From the available standard values, select $120\ \Omega$.

   Power in the resistor:

   $$P_R = I^2R = (0.015\ \text{A})^2(120\ \Omega) = 0.027\ \text{W} = 27\ \text{mW}$$

Answer Use a $120\ \Omega$ resistor. A standard $0.25\ \mathrm{W}$ resistor is more than sufficient.

10. **Two LEDs in series from a 12 V source:**

    $V_{\mathrm{LED,total}} = 2.0\ \mathrm{V} + 2.0\ \mathrm{V} = 4.0\ \mathrm{V}$

    The resistor must drop:

    $V_R = 12\ \mathrm{V} - 4.0\ \mathrm{V} = 8.0\ \mathrm{V}$

    Required current is $I = 20\ \mathrm{mA} = 0.020\ \mathrm{A}$.

    $R = V_R / I = 8.0\ \mathrm{V} / 0.020\ \mathrm{A} = 400\ \Omega$

    Power in the resistor:

    $P_R = VI = (8.0\ \mathrm{V})(0.020\ \mathrm{A}) = 0.160\ \mathrm{W}$

    Answer The ideal current-limiting resistor is $400\ \Omega$. The resistor dissipates $0.160\ \mathrm{W}$, so a $0.25\ \mathrm{W}$ resistor is technically above the calculated value, but a $0.5\ \mathrm{W}$ resistor would provide better thermal margin.

11. **Challenge: KVL and KCL as conservation laws:**

    KCL in Problem 6: Kirchhoff's Current Law holds because charge is conserved at a node. Current cannot disappear or accumulate at the junction in steady-state circuit operation. The total current entering the node equals the sum of the current through the $220\ \Omega$ branch and the current through the $470\ \Omega$ branch.

    Physical meaning: KVL represents conservation of energy. KCL represents conservation of charge.

This page titled 11.15: End-of-Chapter Problem Set is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

12: PARAMETRIC CAD AND 3D MODELING

Parametric CAD and 3D Modeling

Parametric CAD is rule-based geometry, not drawing. Constraints enforce relationships; parameters control magnitudes; design intent encodes how the model responds to change.

Learning Objectives

- Distinguish parametric CAD from 2D drafting and explain why the distinction matters.
- Define design intent and explain why it must be encoded in the model, not just assumed.
- Describe the role of geometric constraints and dimensions in a fully defined sketch.
- Identify the primary solid modeling operations (extrude, cut, revolve) and what each produces.
- Calculate mass from geometry using density, volume, and unit conversions — and verify against Fusion's mass properties tool.
- Apply the structured modeling workflow to a new part.
- Describe what an ASME engineering drawing communicates and why it is the standard output format.
- Identify the nine deliverables for the Fusion Project and what the ASME drawing of Part 9 must contain.

12.1: CAD Is Not Drawing
12.2: Parametric Modeling Core Concepts
12.3: From Sketch to Solid
12.4: The Sketch as a Mathematical Definition
12.5: The Fusion Project
12.6: Modeling Workflow
12.7: Five Common Modeling Errors
12.8: CAD in the Engineering Landscape
12.9: Summary
12.10: End-of-Chapter Problems

This page titled 12: Parametric CAD and 3D Modeling is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

12.1: CAD Is Not Drawing

12.1 CAD Is Not Drawing

This distinction matters enough to state first.

Drawing — whether on paper or in 2D CAD like AutoCAD — creates a static visual representation. A 2D drawing of a bracket is a picture of a bracket. If you change the width, you redraw the affected lines. The drawing has no knowledge of relationships between dimensions.

Parametric CAD creates a mathematical model of geometry. Dimensions are parameters that control shape. Constraints are geometric rules the software enforces. When you change a parameter, the model recomputes. A hole constrained to remain 10 mm from an edge will still be 10 mm from the edge after you change the plate width — because the relationship was encoded in the model.

Differences between static 2D drawing and parametric CAD modeling.

Property	2D Drawing	Parametric CAD
What it creates	A static visual	A mathematical model
Dimensions	Labels on a picture	Parameters that drive geometry
When you change a size	Redraw affected lines	Model recomputes automatically
Relationships encoded	None (visual only)	Constraints enforce rules
Mass properties	Not available	Computed from geometry

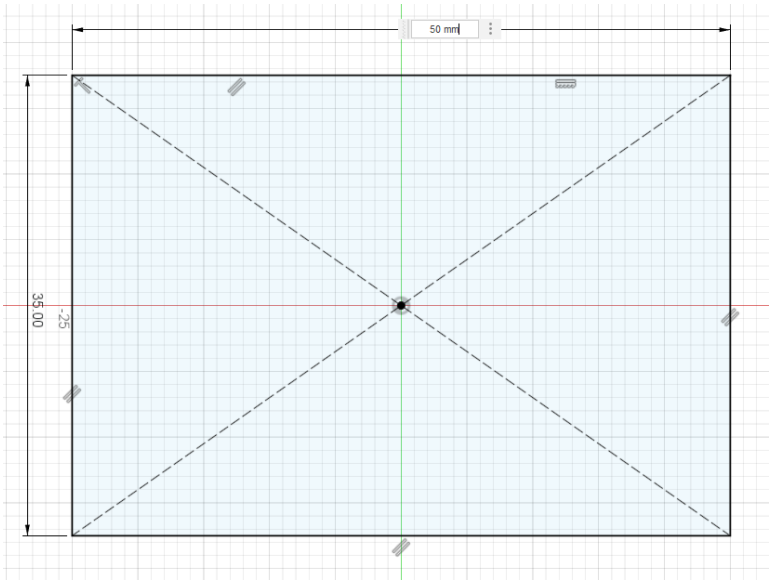


Figure 12.1.1: Fusion sketch of a fully defined rectangular profile using parametric dimensions and geometric constraints. The 50 mm horizontal dimension and 35 mm vertical dimension define the overall sketch size, while the constraint indicators and construction diagonals show how the rectangle is controlled relative to the sketch origin. (Copyright; Jonathan Compton, CC BY-NC 4.0).

This page titled 12.1: CAD Is Not Drawing is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

12.2: Parametric Modeling Core Concepts

12.2 Parametric Modeling — Core Concepts

Parameters

A parameter is a controlled variable that drives geometry. In Fusion 360, every dimension you add to a sketch is a parameter. Parameters can be named and related (`hole_offset = 0.1 * width`), so changing one dimension cascades correctly through the entire model.

This is directly analogous to the spreadsheet structure from Chapter 8: just as a spreadsheet's model cells reference input cells rather than hardcoded numbers, a parametric model's geometry references parameters rather than fixed values. The same discipline applies — hardcode a dimension and the model becomes brittle; reference a parameter and the model becomes adaptable.

Geometric Constraints

Constraints are geometric rules enforced by the software. A dimension controls *size*; a constraint controls *relationship*.

Common geometric constraints used in parametric CAD sketches.

Constraint	What it enforces
Coincident	Two points share the same location
Horizontal / Vertical	A line is exactly horizontal or vertical
Parallel	Two lines run in the same direction
Perpendicular	Two lines meet at exactly 90°
Equal	Two lengths or radii are identical
Symmetric	Two features mirror across a centerline
Tangent	A line and arc are tangent at their meeting point
Fixed	A point or line does not move

Fully constrained vs. under-constrained: A fully constrained sketch has exactly enough constraints and dimensions — no geometry can drift. Fusion shows under-constrained geometry in blue; fully constrained geometry in black. An under-constrained sketch is like starting Analyze without completing Define.

Design Intent

Design intent is the engineering logic embedded in a model — how should the model behave when parameters change?

Example: A mounting plate with four corner bolt holes, required to be 10 mm from each edge.

Without design intent: Place holes visually ~10 mm from edges. Change plate width → holes stay where drawn → relationship broken.

With design intent: Constrain hole centers at a 10 mm dimension from each edge. Change plate width → holes reposition automatically. The model enforces the functional requirement.

This page titled 12.2: Parametric Modeling Core Concepts is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

12.3: From Sketch to Solid

12.3 From Sketch to Solid — Modeling Operations

Solid models are built by applying operations to 2D sketches. The feature tree records every operation in order, making the model auditable and revisable.

Common solid modeling operations used to build Fusion parts.

Operation	What it does	Fusion Project use
Extrude	Pushes a 2D profile along a direction to add material	Parts 1–7 (most primitives)
Cut	Extrusion that removes material	Part 9 (the hole)
Revolve	Rotates a profile around an axis	Part 2 (cylinder), alternative for Part 8 (sphere)
Sphere (direct)	Create → Sphere command	Part 8

Each operation records in the feature tree. You can go back and edit any previous feature — but later features depend on earlier ones, so deleting an early sketch may break downstream geometry. A well-ordered feature tree is as important as well-constrained sketches.

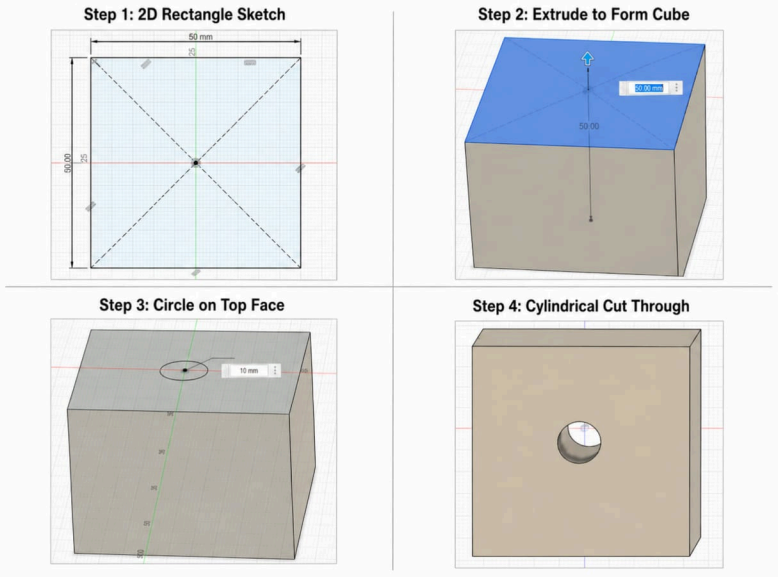


Figure 12.3.1: Four-step Fusion workflow for building Part 9, a cube with a through-hole. The sequence shows a 2D rectangle

sketch, extrusion into a cube, creation of a circle on the top face, and a cylindrical cut through the solid. (Copyright; Jonathan Compton, AI-assisted original figure, CC BY-NC 4.0)

This page titled 12.3: From Sketch to Solid is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

12.4: The Sketch as a Mathematical Definition

12.4 Mass and Geometry — Calculating and Verifying

One of the most direct engineering uses of a CAD model is extracting mass properties:

$$m = \rho \cdot V$$

For steel: $\rho \approx 7850 \text{ kg/m}^3 = 7.85 \times 10^{-3} \text{ g/mm}^3$. The Fusion Project uses steel as the default material with mm as the working unit.

Workflow — calculate first, then verify

1. Calculate expected mass by hand using geometry volume formulas.
2. Build the model in Fusion 360.
3. Check Physical Properties (Inspect → Physical Properties).
4. If hand calculation and Fusion agree → high confidence the model is correct.
5. If they disagree → find the error in the calculation or the model.

✓ Worked Example 12.1 — Mass of the Fusion Project Cylinder (Part 2)

Cylinder: radius = 15 mm, length = 40 mm.

$$V = \pi r^2 L = \pi(15)^2(40) = 28{,}274 \text{ mm}^3$$

$$m = 7.85 \times 10^{-3} \times 28{,}274 = 221.95 \text{ g}$$

Fusion result: 221.954 g ✓

Agreement to 4 significant figures confirms the model is correctly dimensioned.

✓ Worked Example 12.2 — Mass of the Cube with Hole (Part 9)

5 cm × 5 cm × 5 cm cube with a 1 cm diameter hole through the front face. Convert to mm: 50 mm cube, 5 mm radius hole.

$$V_{\text{cube}} = 50^3 = 125{,}000 \text{ mm}^3$$

$$V_{\text{hole}} = \pi(5)^2(50) = 3{,}927 \text{ mm}^3$$

$$V_{\text{net}} = 125{,}000 - 3{,}927 = 121{,}073 \text{ mm}^3$$

$$m = 7.85 \times 10^{-3} \times 121{,}073 = 950.4 \text{ g}$$

Fusion result: 950.423 g ✓

This page titled 12.4: The Sketch as a Mathematical Definition is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

12.5: The Fusion Project

12.5 The Fusion Project — Nine Parts and One Drawing

Part List with Expected Masses (Steel Default)

Fusion project part list with key dimensions and expected steel masses.

Part	Description	Key dimensions	Expected mass
1	Cube	10 × 10 × 10 mm	7.85 g
2	Cylinder	r = 15 mm, L = 40 mm	221.954 g
3	Elliptic Cylinder	1 in × 0.75 in × 4 in	303.097 g
4	Pentagon Prism	28 mm **inscribed** circle, 30 mm long	109.747 g
5	Hexagon Prism	26 mm **circumscribed** circle, 4 mm long	18.383 g
6	Large Cube	10 × 10 × 10 in	1.286×10^5 g
7	Triangular Prism	20 mm base, 15 mm height, 30 mm long	35.325 g
8	Sphere	70 mm diameter	1409.816 g
9	**Cube with Hole**	5 cm × 5 cm × 5 cm, 1 cm diameter hole	**950.423 g**

Note on Parts 3 and 6 (inch dimensions): Keep the default Fusion unit in mm, then type the unit explicitly when dimensioning (e.g., `1 in`). If you forget and type just `1`, Fusion interprets this as 1 mm — the part will have the wrong proportions and a dramatically different mass.

Note on Parts 4 and 5 (inscribed vs. circumscribed): The Fusion polygon tool lets you specify which type — selecting the wrong one produces the wrong geometry and the wrong mass. Inscribed = circle touches side midpoints. Circumscribed = circle passes through vertices.

The Drawing — Part 9

In addition to nine `.f3d` model files, the project requires one ASME standard engineering drawing of Part 9 (the cube with hole), exported as PDF.

Required drawing elements for the Part 9 engineering drawing.

Drawing element	What it shows / contains
Third-angle projection views	Front view, top view (above front), right-side view (right of front) — US/ASME standard
Isometric view	3D pictorial view from a diagonal angle — helps reader visualize geometry
Dimensions	Cube side length, hole diameter — all information needed to manufacture without the model file
Title block	Project: *Fusion 360 Project* | Title: *Cube with Hole* | scale, standard info

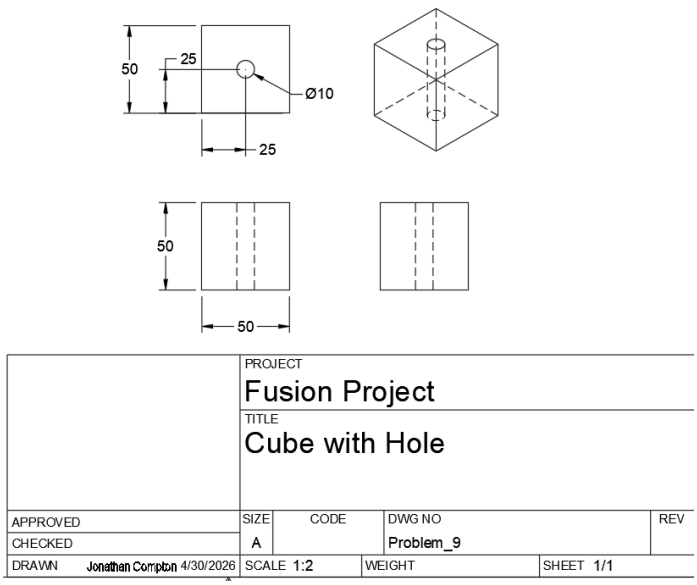


Figure 12.5.1: ASME-style drawing for Part 9, a 50 mm cube with a centered Ø10 mm through-hole. The drawing includes orthographic views, an isometric reference view, hidden lines showing the hole through the cube, centerline/center-location dimensions, and a completed title block. (Copyright; Jonathan Compton, CC BY-NC 4.0).

This page titled 12.5: The Fusion Project is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

12.6: Modeling Workflow

12.6 Modeling Workflow — Applying the Engineering Framework

The same structured thinking from Chapter 6 applies to CAD modeling:

- **Define.** Before opening Fusion: identify the required geometry, constraints, units, and material. Compute expected mass by hand.
- **Sketch.** Select the correct plane. Apply constraints first (relationships before magnitudes), then dimensions. Confirm the sketch turns black (fully constrained) before extruding.
- **Feature.** Extrude, cut, or revolve as needed. Check the feature tree is logical and clean.
- **Verify.** Check Physical Properties. Compare to hand calculation. If they disagree, find the discrepancy before proceeding to the next part.
- **Iterate.** If mass is wrong, check: unit settings (mm default?), dimension entries (did you type units?), polygon type (inscribed vs. circumscribed?), sketch constraints (fully constrained?).

This page titled 12.6: Modeling Workflow is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

12.7: Five Common Modeling Errors

12.7 Five Common Modeling Errors

1. **Under-constrained sketches.** Blue geometry in Fusion = under-constrained. Extruding an under-constrained sketch creates a model that can shift unexpectedly. Fix: add the missing constraints or dimensions.
2. **Wrong polygon type.** For Parts 4 and 5, using inscribed where circumscribed (or vice versa) is specified produces the wrong geometry. Mass difference is large enough to fail verification. Fix: read the spec carefully before selecting in the polygon tool.
3. **Missing unit specification for inch dimensions.** Typing `1` when Fusion expects mm gives 1 mm, not 1 inch. Fix: always type the unit explicitly for non-default units: `1 in`, `4 in`.
4. **Hardcoded positions without constraints.** Placing geometry by dragging rather than by dimension constraint makes the position fragile under revision. Fix: use dimension constraints for all position-critical geometry.
5. **Wrong sketch plane.** Orientation doesn't affect mass — but for the drawing (Part 9), the front face matters for correct projection views. Fix: look at the part shape before selecting the base sketch plane.

⚠ Watch Out — Units in Fusion

Fusion stores all geometry internally in a single unit system. When mixing mm and inch dimensions (Parts 3 and 6), you must type the unit explicitly for non-default dimensions. If you forget, the dimension is interpreted in mm, and the part will have wrong proportions.

The expected mass values in §12.5 are your verification check. If your mass is off by a factor of about 25 (linear conversion) or ~16,000 (cubic conversion: 1 in^3 = 16,387 mm^3), a unit specification error is almost certainly the cause.

This page titled 12.7: Five Common Modeling Errors is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

12.8: CAD in the Engineering Landscape

12.8 CAD in the Engineering Landscape

Parametric CAD sits at the intersection of design (Chapter 7) and the physical world. It is a design analysis tool:

- **Volume and mass** come directly from model geometry — as you verified in §12.4.
- **Stress analysis (FEA)** can be applied to find regions of high stress — directly connected to the allowable stress calculations from Chapter 7.
- **Parametric sensitivity** — changing a wall thickness by 1 mm immediately shows the mass and volume change. That is the same design-iteration thinking as Chapter 7's iterative bracket refinement.
- **3D printing** reads STL files exported from the CAD model. Errors in the model become errors in the physical part.
- **Manufacturing drawings** (ASME standard, Part 9) are the formal output that machinists use. An ambiguous drawing wastes material and fabrication time.

🔗 Connection — Fusion Project

Submit to Canvas:

- **Nine model files:** `P1_LastName_FirstInitial.f3d` through `P9_LastName_FirstInitial.f3d`
- **One drawing PDF:** `D1_LastName_FirstInitial.pdf` — ASME drawing of Part 9
- **Canvas comment:** list mass in grams for each part in the format `Part 1: x.xx g`

Use the expected mass values in §12.5 as your verification check after modeling each part.

Access Fusion 360 free for education at: autodesk.com/education/free-software/featured — use your AVC Gmail address for account creation.

Key Takeaway

Parametric CAD is rule-based geometry, not drawing. Constraints enforce relationships; parameters control magnitudes; design intent encodes how the model should respond to change. The Fusion modeling workflow — Define, Sketch with constraints first then dimensions, Feature, Verify mass against hand calculation, Iterate when verification fails — maps directly to the engineering problem-solving workflow you already know. The expected mass values for the nine Fusion Project parts are your ground truth for verification.

This page titled 12.8: CAD in the Engineering Landscape is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

12.9: Summary

Summary

- Parametric CAD encodes engineering logic — constraints enforce relationships; dimensions control size. Design intent determines how the model responds to revision.
- A fully constrained sketch has no remaining degrees of freedom. Fusion shows under-constrained geometry in blue; fully constrained in black.
- Primary operations: extrude (adds material), cut (removes material), revolve (rotates profile). The feature tree records operations in order.
- Mass: $m = \rho V$. For steel: $\rho \approx 7.85 \times 10^{-3}\ \mathrm{g/mm^3}$. Calculate by hand first; verify against Fusion Physical Properties.
- The Fusion Project: nine parts (cube through cube-with-hole) plus one ASME drawing of Part 9.
- Five common errors: under-constrained sketch, wrong polygon type, missing unit specification, hardcoded positions, wrong sketch plane.
- ASME engineering drawings: third-angle projection (US standard), isometric view, dimensions, title block — the formal communication to fabricators.

This page titled 12.9: Summary is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

12.10: End-of-Chapter Problems

End-of-Chapter Problems

1. Explain in your own words why a parametric CAD model is more useful than a 2D drawing for a part that will be revised.
2. A steel rectangular plate (120 × 80 × 6 mm) has four bolt holes (5 mm diameter, 8 mm from each edge) through the thickness. Calculate expected mass by hand. Show all steps including unit consistency.
3. Which constraint type ensures two holes remain the same diameter when one is resized? Explain.
4. A student models the Fusion Project cylinder (r = 15 mm, L = 40 mm) and gets 2219.5 g instead of 221.95 g. What is the most likely cause? How would you fix it?
5. What is the difference between an inscribed and a circumscribed circle for a polygon? For the hexagon in Part 5 (26 mm circumscribed), calculate the apothem (center to flat side distance) and explain why using the wrong type changes the mass.
6. Describe the three orthographic views required in an ASME third-angle projection drawing of a rectangular solid. Where does each appear on the sheet?
7. A model has a slot length constrained to equal $2 \times$ the bolt diameter. Bolt diameter is currently 8 mm. If it changes to 10 mm, what happens to the slot length? What design intent does this encode?
8. *[Challenge]* Verify the expected mass of the Fusion Project Part 6 (10 in × 10 in × 10 in cube, steel) by hand. Show the conversion from inches to mm explicitly.

Answer Key - click to expand

1. **Parametric CAD vs. 2D drawing:**

   For example, if a hole is defined as centered on a face and located 8 mm from an edge, changing the plate size can preserve that design intent. A 2D drawing records the current shape, but a parametric model records both the shape and the logic used to create it.

2. **Steel rectangular plate with four holes:**

   Plate volume before holes:

   $V_{\text{plate}} = (120\ \text{mm})(80\ \text{mm})(6\ \text{mm}) = 57{,}600\ \text{mm}^3$

   Volume removed by one hole:

   The hole diameter is $5\ \text{mm}$, so $r = 2.5\ \text{mm}$.

   $V_{\text{hole}} = \pi r^2 h = \pi(2.5\ \text{mm})^2(6\ \text{mm}) = 117.8\ \text{mm}^3$

   Volume removed by four holes:

   $V_{\text{removed}} = 4(117.8\ \text{mm}^3) = 471.2\ \text{mm}^3$

Net volume:

$V_{\text{net}} = 57{,}600\ \text{mm}^3 - 471.2\ \text{mm}^3 = 57{,}128.8\ \text{mm}^3$

Mass:

$m = \rho V = (0.00785\ \text{g/mm}^3)(57{,}128.8\ \text{mm}^3) = 448.5\ \text{g}$

Answer The expected mass is approximately $448\ \text{g}$.

3. **Constraint type for equal hole diameters:**

   If one hole diameter is changed, the other hole diameter updates automatically. This preserves design intent because the model records that the holes are supposed to match, not just that they happened to have the same number when first drawn.

4. **Cylinder mass is off by a factor of 10:**

   $V = \pi r^2 L = \pi(15\ \text{mm})^2(40\ \text{mm}) = 28{,}274\ \text{mm}^3 = 28.274\ \text{cm}^3$

   $m = (7.85\ \text{g/cm}^3)(28.274\ \text{cm}^3) \approx 221.95\ \text{g}$

   If the student gets $2219.5\ \text{g}$, the result is exactly 10 times too large. The most likely cause is a unit or material-density error, such as entering the density as $78.5\ \text{g/cm}^3$ instead of $7.85\ \text{g/cm}^3$, or using an incorrect unit conversion.

   Fix: Check the document units, confirm the model dimensions are in millimeters, verify the assigned material is steel, and confirm the density is approximately $7.85\ \text{g/cm}^3$ or $0.00785\ \text{g/mm}^3$.

5. **Inscribed vs. circumscribed circle for a polygon:**

   An **inscribed circle** fits inside the polygon and touches the flat sides. For a hexagon, this circle touches the middle of each side.

   For a regular hexagon with a $26\ \text{mm}$ circumscribed diameter:

   $R = 26\ \text{mm}/2 = 13\ \text{mm}$

   The apothem is the distance from the center to a flat side:

   $a = R\cos(30\ °) = (13\ \text{mm})\cos(30\ °) = 11.26\ \text{mm}$

   Answer The apothem is approximately $11.3\ \text{mm}$.

   Using the wrong circle type changes the size of the hexagon. If a student uses a 26 mm inscribed circle instead of a 26 mm circumscribed circle, the distance across flats becomes 26 mm instead of about 22.5 mm. That makes the hexagon larger, which increases volume and mass.

6. **ASME third-angle orthographic views:**
   - **Front view:** The main view. It should show the most descriptive shape of the part.
   - **Top view:** Placed directly above the front view in third-angle projection.
   - **Right-side view:** Placed directly to the right of the front view in third-angle projection.

   For a rectangular solid, the front view usually shows width and height, the top view shows width and depth, and the right-side view shows depth and height. Hidden lines may be used when internal features, such as holes, are not directly visible in a given view.

7. **Slot length tied to bolt diameter:**

$$\text{slot length} = 2 \times \text{bolt diameter}$$

If the bolt diameter changes from $8\ \text{mm}$ to $10\ \text{mm}$, then:

$$\text{slot length} = 2(10\ \text{mm}) = 20\ \text{mm}$$

Answer The slot length changes from $16\ \text{mm}$ to $20\ \text{mm}$.

This encodes the design intent that the slot should always scale with the bolt size. The slot is not just a fixed length; it is defined as a functional clearance feature based on the bolt diameter.

8. **Challenge: 10 in × 10 in × 10 in steel cube:**

$$1\ \text{in} = 25.4\ \text{mm}$$

$$10\ \text{in} = 10(25.4\ \text{mm}) = 254\ \text{mm}$$

Volume:

$$V = (254\ \text{mm})(254\ \text{mm})(254\ \text{mm}) = 16{,}387{,}064\ \text{mm}^3$$

Convert to cubic centimeters:

$$1\ \text{cm}^3 = 1000\ \text{mm}^3$$

$$V = 16{,}387{,}064\ \text{mm}^3/1000 = 16{,}387.064\ \text{cm}^3$$

Mass:

$$m = \rho V = (7.85\ \text{g/cm}^3)(16{,}387.064\ \text{cm}^3) = 128{,}638\ \text{g}$$

Convert to kilograms:

$$128{,}638\ \text{g} = 128.6\ \text{kg}$$

Answer The expected mass of the steel cube is approximately $128.6\ \text{kg}$.

This page titled 12.10: End-of-Chapter Problems is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

13: ENGINEERING COMMUNICATION

Engineering Communication

Undocumented engineering is unfinished engineering. Communication is not the final step that can be abbreviated — it is part of the deliverable.

Learning Objectives

- Explain why communication is an integral part of the engineering deliverable, not a step that can be abbreviated.
- Structure a technical engineering memo using the standard six-section format.
- Write with quantitative precision: correct units, appropriate significant figures, and objective rather than vague language.
- Select between graphs and tables appropriately and explain the purpose of each.
- Construct a technical presentation for a non-technical audience.
- Apply the communication framework to the TEGO case study.
- Identify the five most common errors that undermine an engineering document's credibility.

13.1: Communication Is Part of the Deliverable
13.2: Three Modes of Communication
13.3: The Technical Memo
13.4: Complete Memo Example
13.5: Quantitative Precision
13.6: Graphs and Tables
13.7: Technical Presentations
13.8: TEGO Case Study
13.9: Professional Email
13.10: Communicating Uncertainty and Limitations
13.11: Summary
13.12: End-of-Chapter Applied Exercises

This page titled 13: Engineering Communication is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

13.1: Communication Is Part of the Deliverable

13.1 Communication Is Part of the Engineering Deliverable

A bridge is built from drawings. A circuit is built from schematics. A manufacturing line is built from specifications.

Engineering decisions exist in two domains simultaneously — in the physical world and in documentation. The physical artifact may be exactly right, but if the documentation is absent or misleading, the engineering work is not complete. Other engineers cannot check it, build on it, or defend it. Future engineers cannot maintain it.

This is why the five-step workflow ends with *Communicate* — not because writing is the final chore, but because undocumented engineering is unfinished engineering. Every project in this course requires a communication deliverable for this reason: a Cable Car memo, a MATLAB comparison memo, a TEGO presentation, a PowerPoint on an engineering discipline, a design selection brief.

> **The auditable standard:** could a competent engineer who was not involved in the work understand what was done, verify the calculations, evaluate the assumptions, and reach the same conclusion — from your document alone? If not, the communication is incomplete.

This page titled 13.1: Communication Is Part of the Deliverable is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

13.2: Three Modes of Communication

13.2 Three Modes of Engineering Communication in This Course

Communication modes used for engineering work in this course.

Mode	Purpose	This course uses it for
Technical memo	Document an analysis; present a finding; recommend an action	Cable Car memo; Programming Project comparison memo
Technical presentation	Communicate decisions to an audience, often non-technical	TEGO case study; PowerPoint Project; Design Selection brief
Professional email / brief	Efficient, context-appropriate exchange	Any course-related communication with instructors or colleagues

This page titled 13.2: Three Modes of Communication is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

13.3: The Technical Memo

13.3 The Technical Memo — Structure and Standards

```
ENGINEERING MEMORANDUM

To: [Recipient name / title]
From: [Your name]
Date: [Date]
Subject: [Specific topic — one line]
```

A professional engineering memo has six sections:

Required sections of a professional engineering memo.

Section	What it contains	Common failure
1. Purpose	What system, what question, what constraints	"This memo is about the circuit" — no scope
2. Background	Physical system, parameters, governing relationships, **assumptions**	Assumptions missing or buried later
3. Methodology	Equations used, tools used, parameter ranges, simplifications	"I used Excel" — not replicable
4. Results	Key numbers, labeled graphs, summary tables	Graphs without axis labels or units
5. Analysis	Interpretation: is the result reasonable, what does it mean, are thresholds exceeded?	Just restates results without meaning
6. Conclusion	Specific decision, justification, limitations, next steps	"The results look okay" — not a conclusion

Weak vs. strong writing, section by section

Examples of weak and strong technical memo writing.

Section	Weak	Strong
Purpose	"This memo is about the cable car project."	"This memo presents the cable car velocity model, evaluates peak velocity, identifies regime transition points, and compares Excel and MATLAB implementations."
Methodology	"I used Excel to calculate velocities."	"Velocity was computed using two piecewise equations in 5-ft increments from 0 to 500 ft. Excel IF/IFS functions applied the appropriate equation at each step."
Analysis	"Velocity increases near the middle."	"Peak velocity of 10.5 ft/s occurs at d = 250 ft, 47% above the near-tower minimum of 7.2 ft/s, consistent with the parabolic mid-cable model."
Conclusion	"The power is 0.436 W."	"The 330 Ω, 0.25 W resistor is unsafe at 12 V — dissipation of 0.436 W exceeds the rating by 74%. A 0.5 W or higher rated component is required."

⚠ Watch Out — The Five Most Common Memo Errors

1. **Vague Purpose** — "This memo discusses the circuit" is not a scope.
2. **Missing assumptions** — buried in Methodology or absent entirely.
3. **Results without analysis** — presenting 0.436 W without stating it exceeds the 0.25 W rating by 74%.
4. **Unlabeled graphs** — a graph without axis labels and units is not an engineering figure.
5. **Conclusion that restates results** — "The power is 0.436 W" is a result. A conclusion states what that means and what should be done.

This page titled 13.3: The Technical Memo is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

13.4: Complete Memo Example

13.4 A Complete Memo Example — Annotated

Here is a complete six-section memo addressing the resistor safety question from Chapter 11. Notice what it accomplishes in under one page.

ENGINEERING MEMORANDUM

To: ENGR 110 Instructor

From: [Student Name]

Date: [Date]

Subject: Resistor Power Rating Evaluation — 12 V Source, 330 Ω Resistor

---

1. Purpose

This memo evaluates whether a 330 Ω resistor rated at 0.25 W is safe for use in a 12 V DC circuit. The analysis determines actual power dissipation, compares it to the rated limit, and provides a component recommendation.

2. Background

The circuit consists of a 12 V DC source in series with a single 330 Ω resistor rated at 0.25 W maximum continuous power. *Assumptions: ideal resistor (constant resistance), DC source, steady-state conditions only.*

3. Methodology

Power dissipation was computed using $P = V^2/R$, derived from Ohm's Law and the power relationship $P = VI$. No parameter sweep was required; the operating condition is a single fixed voltage.

4. Results

$$P = \frac{V^2}{R} = \frac{(12\ \text{V})^2}{330\ \Omega} = \frac{144}{330} = 0.436\ \text{W}$$

5. Analysis

The computed dissipation of 0.436 W exceeds the 0.25 W rating by 74%. Operating at this condition would cause the resistor to overheat and potentially fail. The current drawn ($I = 12/330 = 36.4$ mA) is reasonable for signal circuits, but the power constraint is the binding limit.

6. Conclusion and Recommendation

The 330 Ω, 0.25 W resistor is **not safe** at 12 V. A minimum rating of 0.5 W is required. Alternatively, resistance could be increased to 620 Ω (next standard value above 576 Ω) to stay below 0.25 W. *Limitation: this analysis assumes steady-state DC; transient or AC conditions require additional analysis.*

This page titled 13.4: Complete Memo Example is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

13.5: Quantitative Precision

13.5 Writing with Quantitative Precision

Engineering writing has one standard that distinguishes it from other technical writing: **every significant claim must be supported by a number with a unit.**

Weak and strong examples of quantitative engineering writing.

Weak (unverifiable)	Strong (verifiable)
"The results seem high."	"Maximum tension is 21,500 N, exceeding the 18,000 N allowable by 19%."
"The resistor gets hot."	"Dissipation is 0.436 W, 74% above the 0.25 W rating."
"Velocity increases near the middle."	"Velocity peaks at 10.5 ft/s at d = 250 ft, 47% higher than the near-tower minimum of 7.2 ft/s."
"The mass looks right."	"Calculated mass is 221.95 g; Fusion shows 221.954 g — agreement to 4 significant figures."
"Safety factor is adequate."	"Actual SF = 2.08, exceeding the required SF = 2.0 by 4%."

Significant figures: report results to appropriate precision — not more, not fewer. Reporting $I = 0.03$ A implies one significant figure. $I = 0.030$ A correctly communicates two. $I = 0.030909090909$ A implies false precision. **Units always:** a number without a unit in an engineering document is an error.

This page titled 13.5: Quantitative Precision is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

13.6: Graphs and Tables

13.6 Graphs and Tables — Choosing the Right Tool

When to use tables versus graphs in engineering communication.

Use a table when...	Use a graph when...
Exact numerical values are what the reader needs	Trend shape is the key information
Comparing discrete options side by side	Showing continuous behavior across a range
Precision matters more than pattern	Pattern matters more than precision; threshold crossings visible

Never present the same data in both a table and a graph unless there is a specific reason (e.g., a graph shows the trend; a table gives the precise crossover value). Every graph in a memo must meet Chapter 9's five requirements.

This page titled 13.6: Graphs and Tables is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

13.7: Technical Presentations

13.7 Technical Presentations

The second major communication mode in this course is the technical presentation — used for the TEGO case study, Engineering Design Selection brief, and PowerPoint Presentation Project.

Presentations and memos serve different purposes. A memo is read at the reader's own pace and can be checked. A presentation is delivered under time constraints — the argument must land in real time.

The Five Core Rules for Engineering Presentations

1. **Slides carry visuals; the speaker carries the argument.** Do not put paragraphs on slides. Slides hold graphs, diagrams, tables, images. Your voice explains what the evidence means.
2. **Structure the argument before designing the slides.** What is your claim? What evidence supports it? What objection would a skeptic raise? What is the recommended action? Write that paragraph first — then design slides around the evidence.
3. **Match vocabulary to the audience.** The TEGO presentation goes to business managers. The PowerPoint Project goes to first-semester engineering students. Both are non-specialists in whatever you are presenting.
4. **Cite all sources.** Every data point, salary figure, technical specification, or company fact needs a source. The reference slide is required.
5. **Practice the narration.** Slides with excellent visuals and an unprepared speaker are less effective than adequate slides with a well-prepared speaker.

This page titled 13.7: Technical Presentations is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

13.8: TEGO Case Study

13.8 The TEGO Case Study — Applied Communication

The TEGO case study is a deliberate exercise in a skill engineers use constantly: communicating a solution to decision-makers who need to act on it but are not equipped to evaluate the technical details themselves.

The scenario: TEGO is a small toy company that has received 300 complaints about missing pieces in its most popular set. All sets are hand-packed — no automated equipment in the workflow. You are the engineer tasked with diagnosing the problem and presenting solutions to management (business majors, not engineers).

This is a design problem (Chapter 7) communicated through an engineering presentation. Your five slides:

Required TEGO case study slides and their engineering design parallels.

Slide	Required content	Engineering parallel
1	Problem identification and consequences — what is happening and why does it matter? Quantify where possible.	Ch 7 Step 1: Define the design objective
2	Variables affecting the problem — what factors cause pieces to be missed? Worker fatigue? Piece similarity? Station design?	Ch 7 Step 2: Identify constraints
3	Possible solutions — at least three meaningfully different approaches (vision system, weight verification, process redesign, QC staff...)	Ch 7 Step 3: Generate alternatives
4	Solution requirements and costs — capital cost, training, maintenance, throughput impact for your preferred solution(s)	Ch 7 Step 5: Evaluate trade-offs
5	Testing and optimization — how would you verify the solution works? What metric, and what constitutes success?	Ch 7 Step 6/7: Iterate and communicate

The audience is business managers. Your slides should communicate clearly without requiring technical background. Engineering details belong in speaker notes — but the conclusion and recommendation must be immediately understandable.

> 🔗 **Connection — TEGO Case Study and PowerPoint Presentation Project**

TEGO Case Study (HW 04): minimum 5 slides, submitted as PDF. One slide per bullet above. Audience: business managers.

PowerPoint Presentation Project: minimum 10 content slides + title + reference = 12 total. Narration audio for each slide (~8–10 minutes). Topic: your assigned engineering discipline. Required content: abbreviation and professional society; typical academics and certifications (sourced from 2 universities); typical and non-typical employers and job descriptions (at least 4 companies); actual products and research topics; your personal thoughts. Use AVC's free Microsoft 365 account for PowerPoint. Filename: `Discipline_LastName_FirstName`.

This page titled 13.8: TEGO Case Study is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

13.9: Professional Email

13.9 Professional Email

Engineering email has conventions that differ from casual texting. They exist because professional email is a permanent record that can be forwarded, cited, and retrieved.

Subject: ENGR 110-2: Homework 2 Problem 2 Question

Hello Dr. Compton,

I am a student in your ENGR 110 Section 2. I have a question about Problem 2 on Homework 2. I believe the problem is asking [specific interpretation], but I am not certain how to apply [specific concept] to the given conditions. Can you confirm whether my approach of [specific method] is on the right track?

Thank you,

Your Name

The specific email — "I tried X, got Y, I think the issue might be Z" — is much more useful to answer than "I'm confused." The conventions serve the communication, not the other way around.

- **Subject line:** specific. "ENGR 110-2: Question about HW 2 Problem 2" — not just "Question."
- **Salutation:** use the recipient's title. "Hello Dr. Compton" — not "Hey."
- **Context immediately:** state your course and section. Do not assume they recognize your email address.
- **Specific question:** state what you tried, what happened, and your hypothesis. Not just "I don't get it."
- **Closing:** "Thank you" and your full name.

This page titled 13.9: Professional Email is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

13.10: Communicating Uncertainty and Limitations

13.10 Communicating Uncertainty and Limitations

Stating limitations is not a weakness — it is professional integrity. A limitation that is stated can be planned around. A limitation that is concealed can cause someone to rely on your result in a situation where it does not apply.

The format for a limitation statement:

> *"This analysis assumes [stated condition]. Results may not apply if [condition is violated]. Additional analysis would be required to [extend scope]."*

Examples from this book's running problems:

- "This analysis treats the cable load as purely axial. Real bracket geometry introduces bending, which would require a more detailed finite element analysis."
- "The cable tension model assumes static equilibrium. Dynamic loads (vibration, impact) are not captured."
- "Power dissipation was computed for steady-state DC. Transient or AC conditions are outside the scope of this analysis."

Ethics Check

Omitting a known limitation from an engineering document is not a minor omission — it is misleading communication. If a structural analysis report does not state that it assumed static loading, and the structure subsequently experiences dynamic loads, the engineer who wrote that report has misrepresented what was analyzed. This is part of what the Communicate step in the engineering workflow means: accurately representing what you know, how you know it, and what you do not know.

Key Takeaway

Engineering communication is part of the deliverable. A technical memo communicates through six sections: Purpose, Background (with explicit assumptions), Methodology, Results, Analysis, Conclusion. Every significant claim needs a number with a unit. Presentations carry visual evidence on slides and argument in the speaker's voice. State limitations explicitly. The auditable standard: could a competent engineer not involved in the work verify your reasoning from your document alone?

This page titled 13.10: Communicating Uncertainty and Limitations is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

13.11: Summary

Summary

- Undocumented engineering is unfinished engineering. Communication is integral to the deliverable.
- Three modes in this course: technical memos, technical presentations, professional written communication.
- A technical memo has six sections: Purpose, Background (with assumptions), Methodology, Results, Analysis, Conclusion and Recommendation.
- Write with quantitative precision: every significant claim needs a number with a unit. Vague language is unverifiable language.
- Use graphs for trends; tables for precise values and discrete comparisons.
- Presentations: slides carry visual evidence, speaker carries the argument. No paragraphs. Match vocabulary to audience. Cite sources.
- State limitations explicitly — an omitted limitation is misleading communication.
- The auditable standard applies to everything a professional engineer produces.

This page titled 13.11: Summary is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

13.12: End-of-Chapter Applied Exercises

End-of-Chapter Applied Exercises

1. **Rewrite for precision.** Convert each weak statement: (a) "The tension is too high." (b) "Power dissipation seemed okay." (c) "The mass came out about right." (d) "The velocity increases in the middle."
2. **Memo practice.** Draft a complete six-section memo for: 9 V source, 220 Ω resistor, 0.25 W rating. Is it safe? The analysis takes one calculation; the memo takes six paragraphs.
3. **Graph introduction practice.** For the cable tension graph you built in Chapter 8, write two sentences introducing it in a Results section: one citing the figure, one stating the key result.
4. **Conclusion rewrite.** A classmate's Conclusion reads: "In conclusion, the computed power is 0.436 W and the current is 36.4 mA." Identify two problems and rewrite it.
5. **TEGO prep.** Before doing the TEGO homework, write a one-paragraph Purpose section as you would for a memo. Then list three candidate solutions and the primary trade-off for each.
6. **Limitation practice.** For each: (a) cable tension, static loading assumed; (b) resistor power, steady-state DC; (c) bracket stress using $\sigma = F/A$ (axial only) — write a one-sentence limitation statement.
7. **Professional email.** Write a professional email to your instructor asking for clarification on the TEGO case study assignment. Apply the conventions from §13.9.

Answer Examples - click to expand

1. **Rewrite for precision:**
   a. **Weak:** “The tension is too high.”
   Improved: “The computed cable tension is $14.2\ \text{kN}$, which exceeds the $10.0\ \text{kN}$ cable rating by $4.2\ \text{kN}$, so the cable angle or cable size must be changed.”

   b. **Weak:** “Power dissipation seemed okay.”
   Improved: “The resistor dissipates $0.172\ \text{W}$, which is below the $0.25\ \text{W}$ rating; however, the design uses about 69% of the rating, so thermal margin should be considered for continuous operation.”

   c. **Weak:** “The mass came out about right.”
   Improved: “The calculated mass is $448\ \text{g}$, which is within 2% of the CAD mass estimate, indicating that the material density and unit conversions are consistent.”

   d. **Weak:** “The velocity increases in the middle.”
   Improved: “The cable car velocity increases as the car approaches midspan, reaching its maximum value near the midpoint where the distance from the nearest tower is greatest.”
2. **Memo practice example:**

   To: ENGR 110 Instructor
   From: Student Name
   Date: [Insert Date]
   Subject: Power Check for a $220\ \Omega$ Resistor on a $9\ \text{V}$ Source

Purpose: The purpose of this memo is to determine whether a $220\ \Omega$ resistor connected to a $9\ \text{V}$ source can safely use a standard $0.25\ \text{W}$ power rating.

Given Information: The circuit uses a $9\ \text{V}$ DC source and a $220\ \Omega$ resistor. The resistor power rating is $0.25\ \text{W}$. The relevant circuit relationships are Ohm's Law, $I = V/R$, and resistor power, $P = V^2/R$.

Analysis: The current is $I = 9\ \text{V}/220\ \Omega = 0.0409\ \text{A}$, or $40.9\ \text{mA}$. The resistor power is $P = V^2/R = (9\ \text{V})^2/220\ \Omega = 0.368\ \text{W}$.

Results: The resistor dissipates $0.368\ \text{W}$, which is greater than the $0.25\ \text{W}$ rating. The resistor would operate at approximately 147% of its rated power.

Conclusion: The $0.25\ \text{W}$ resistor is not safe for this circuit. A higher power rating, such as $0.5\ \text{W}$ or greater, should be selected, or the resistance should be increased to reduce current and power dissipation.

Recommendation: Use a resistor rated above $0.368\ \text{W}$, preferably with additional thermal margin. A $0.5\ \text{W}$ resistor is the minimum practical improvement, but a larger rating may be appropriate for continuous operation or enclosed spaces.

3. **Graph introduction practice:**
4. **Conclusion rewrite:**
   - The conclusion only repeats numerical results; it does not interpret whether the design is acceptable.
   - It does not compare the computed power to a rating, limit, or design requirement.

   Improved conclusion: The resistor dissipates $0.436\ \text{W}$, which exceeds a standard $0.25\ \text{W}$ rating and would not be safe for continuous operation. The circuit should use a resistor with a higher power rating, such as $0.5\ \text{W}$ or greater, or the resistance should be increased to reduce power dissipation.

5. **TEGO prep example:**

   Candidate solutions and primary trade-offs:

   - **Solution A: Simple low-cost design.** Primary trade-off: easiest to build and least expensive, but may provide less performance margin.
   - **Solution B: Higher-performance design.** Primary trade-off: better performance or reliability, but likely requires more material, cost, or assembly time.
   - **Solution C: Balanced design.** Primary trade-off: may not be the best in any single category, but could provide the strongest overall compromise among cost, performance, and manufacturability.

6. **Limitation statements:**
   a. **Cable tension:** This analysis assumes static loading only and does not account for dynamic effects such as swinging, shock loading, vibration, or sudden load application.
   b. **Resistor power:** This calculation assumes steady-state DC operation and does not account for transient behavior, temperature rise, ventilation, or changes in resistance with heating.
   c. **Bracket stress:** The equation $\sigma = F/A$ assumes uniform axial loading and does not account for bending, shear, stress concentrations, buckling, or connection effects.

7. **Professional email example:**

   Subject: Clarification on TEGO Case Study Assignment

   Dear Professor [Last Name],

   I am working on the TEGO case study assignment and would like to clarify one requirement before I continue. For the candidate solutions section, should we compare three fully developed design alternatives, or is it acceptable to compare one main design with two smaller variations?

I want to make sure my memo follows the assignment expectations and uses the correct level of detail. Thank you for your time and help.

Respectfully,

[Student Name]

This page titled 13.12: End-of-Chapter Applied Exercises is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

14: ENGINEERING ETHICS

Engineering Ethics

Engineering ethics is not a philosophical add-on. It is the discipline of making technical decisions responsibly — and it shows up in margin decisions, disclosure decisions, and the habits formed in coursework long before graduation.

Learning Objectives

- State the primary ethical obligation of a professional engineer and identify the NCEES Model Rules as the governing framework.
- Explain why safety factors are an ethical acknowledgment of uncertainty, not just engineering conservatism.
- Apply the eight-step ethical decision-making framework to engineering scenarios.
- Analyze the three Ethics HW scenarios using NCEES principles.
- Describe the Citicorp Building case and identify the ethical principles LeMessurier demonstrated.
- Distinguish between legal compliance and ethical responsibility.
- Connect academic integrity to professional ethics.

14.1: Ethics Is Not an Add-On
14.2: NCEES Model Rules
14.3: Ethics HW Preview Table
14.4: Safety Factors
14.5: Modeling Assumptions
14.6: Eight-Step Framework
14.7: Ethics HW Scenario 1
14.8: Ethics HW Scenario 2
14.9: Ethics HW Scenario 3
14.10: Academic Integrity
14.11: Citicorp Case Study
14.12: Environmental Responsibility
14.13: Summary
14.14: Quick Check

This page titled 14: Engineering Ethics is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

14.1: Ethics Is Not an Add-On

14.1 Ethics Is Not an Add-On to Engineering

Every chapter in this book has embedded ethical content. The Challenger O-ring data in Chapter 9 was a communication ethics problem. The safety factor in Chapter 7 was an uncertainty and public welfare problem. The resistor over-rating in Chapter 11 was a small-margin problem. The limitation statements in Chapter 13 were a disclosure obligation.

These were not special ethics topics. They were normal engineering decisions that happened to have ethical content — which is to say, they were representative of what engineering actually involves.

Engineering ethics is not a philosophical subject studied separately from technical work. It is the discipline of making technical decisions responsibly. An engineer who calculates correctly but omits an unfavorable data point has an ethics problem. An engineer who meets code minimums but leaves a knowingly inadequate margin has an ethics problem. An engineer who approves subordinate work without reviewing it has an ethics problem.

This page titled 14.1: Ethics Is Not an Add-On is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

14.2: NCEES Model Rules

14.2 The Governing Framework — NCEES Model Rules

The **National Council of Examiners for Engineering and Surveying (NCEES) Model Rules of Professional Conduct** provides the ethical codes used in this course and serves as the basis for professional ethics standards across most US jurisdictions.

The NCEES Model Rules establish three purposes for engineering ethics codes:

1. **Harm prevention** — causing harm to people is ethically wrong.
2. **Public welfare promotion** — engineering exists to serve the public, not just clients.
3. **Maintaining integrity and practice standards** — the profession's credibility depends on engineers holding each other to a consistent standard.

The primary obligation:

> *"Engineers shall hold paramount the **safety, health, and welfare of the public**."*

"Hold paramount" means this obligation is not weighed against other considerations — it overrides profit, employer pressure, project deadlines, and personal advancement.

This page titled 14.2: NCEES Model Rules is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

14.3: Ethics HW Preview Table

14.3 The Ethics HW Scenarios — A Preview

The Ethics HW asks you to analyze three scenarios using the NCEES framework. This chapter scaffolds the reasoning for each:

Ethics homework scenarios and the NCEES principles they raise.

Scenario	Core ethical tension	NCEES principle at stake
1 — SolidWorks competence	Misrepresenting capability to obtain a position	Honesty and truthfulness in professional representations
2 — Rubber-stamping supervisor	Approving subordinate work without adequate review	Supervisory responsibility; sealing work you have not verified
3 — Factory safety shortcuts	Workers bypassing mandated safety features for efficiency	Safety regulation compliance; obligation to report unsafe conditions

This page titled 14.3: Ethics HW Preview Table is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

14.4: Safety Factors

14.4 Safety Factors as Ethical Design

$$SF = \frac{\text{Allowable value}}{\text{Actual value}}$$

Designing at SF = 1.0 assumes perfect material, perfect modeling, perfect loading, and zero variability. None of those assumptions are physically realistic. A safety factor is an honest acknowledgment that our models and materials are imperfect, and that the people who rely on engineered systems deserve a margin that accounts for what we do not know.

Reducing a safety factor to save cost is therefore not just a technical decision — it is an ethical one. It transfers the risk of uncertainty onto the public without their knowledge or consent.

✓ Mini Case Study 14.1 — Circuit Overload

A 330 Ω resistor dissipating 0.26 W with a 0.25 W rating. Options: (A) Approve — difference is small. (B) Replace with 0.5 W. (C) Lower voltage. (D) Ignore — usage is short.

Ethical analysis: The rating is exceeded. Whether the excess is "small" depends on unstated assumptions about temperature, tolerance, and usage duration — precisely the kind of hidden assumptions that belong in a memo's Background section, not in a silent approval decision.

Professional action: Option B or C. Engineering ethics is often about small decisions made consistently, not dramatic confrontations.

✓ Mini Case Study 14.2 — Structural Margin Reduction

A project is behind schedule. An engineer is asked to reduce SF from 2.0 to 1.2 to cut material cost.

Ethical analysis: The original SF = 2.0 covered material variability, load estimation uncertainty, model simplification, and fatigue. Reducing to SF = 1.2 eliminates most of that margin without addressing any of the underlying uncertainties. The engineer has no obligation to approve this change — and has an obligation to document the request and the specific uncertainties the original SF covered. If the reduction is escalated over the engineer's objection, that objection should be recorded in writing.

This page titled 14.4: Safety Factors is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

14.5: Modeling Assumptions

14.5 Modeling Assumptions and Transparency

Every model in this book made assumptions: static loading, ideal resistors, axial-only stress, nominal geometry. Assumptions are necessary — models cannot be complete. The ethical obligation is not to eliminate assumptions but to state them visibly.

An assumption that is documented can be checked, challenged, and planned around. An assumption that is hidden allows a model to appear more rigorous than it is — and allows it to be used in contexts where it does not apply, with the engineer who hid the scope limitation bearing responsibility for how it is misused.

This is why the limitation statements from Chapter 13 are ethical requirements, not stylistic suggestions.

This page titled 14.5: Modeling Assumptions is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

14.6: Eight-Step Framework

14.6 The Eight-Step Ethical Decision-Making Framework

Eight-step framework for ethical engineering decision-making.

Step	Action
1	**Identify stakeholders** — who is affected, beyond the client?
2	**Identify safety risks** — what could go wrong, and who is exposed?
3	**Identify applicable regulations** — what codes, standards, or laws apply?
4	**Clarify assumptions** — what is the analysis based on, and are those assumptions stated?
5	**Evaluate margin and uncertainty** — how much design margin exists, and is it adequate?
6	**Consult codes of ethics** — what do the NCEES Rules, professional society codes, or employer codes say?
7	**Seek peer or supervisory input** — consult a colleague or supervisor. Document that consultation.
8	**Document your reasoning** — whatever decision you reach, record how you reached it and why.

Structured ethical reasoning mirrors structured technical reasoning. An engineer who applies the five-step problem-solving framework to calculations and intuition to ethics is applying discipline unevenly.

This page titled 14.6: Eight-Step Framework is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

14.7: Ethics HW Scenario 1

14.7 Ethics HW Scenario 1 — Competence and Honesty

Scenario: Your company urgently needs a new manager. You are invited to apply, but a requirement is SolidWorks competence. You do not currently have that competence — you are enrolled in a CAD course but will not complete it before the application deadline. The application asks whether you have the necessary competence.

The tension: You are otherwise well-qualified. The competence gap is temporary. Claiming competence might get you the position and you might learn quickly enough.

NCEES analysis: The Model Rules require engineers to be "objective and truthful in professional reports, statements, or testimony" and to "avoid deceptive acts." Stating you have SolidWorks competence when you do not is a deceptive act — regardless of your intentions, the timeline, or your other qualifications. Professional credibility depends on engineers meaning what they state.

The honest path: Disclose that you are developing the competence and expect to have it by a specified date. This is accurate, professional, and gives the employer the information they need. An employer who hires based on accurate information cannot fairly blame you if the timeline slips; one who hires based on a false claim has been deceived.

This page titled 14.7: Ethics HW Scenario 1 is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

14.8: Ethics HW Scenario 2

14.8 Ethics HW Scenario 2 — Supervisory Responsibility

Scenario: As a supervisor on a complex design project, you are aware that NCEES rules allow engineers to coordinate a project provided each component is signed or sealed by the responsible engineer. Overworked, you have developed a habit of rubber-stamping subordinates' work after only a superficial read-through — no testing or independent checking. Your justification: the rule says nothing explicit about supervisory checking, and you trust your engineers.

The tension: The letter of the NCEES rule might technically be satisfied. You are overworked. You trust your team. The project is moving.

NCEES analysis: The rule permitting coordination does not eliminate the supervisory engineer's responsibility for the project's overall safety and integrity. Approving without adequate review allows errors to accumulate undetected — and errors in complex projects are often systemic. The NCEES code is a floor, not a ceiling. "I followed the rule" and "I fulfilled my ethical obligation" are not the same statement.

The honest path: Conduct adequate review — which may mean escalating a workload concern rather than silently doing inadequate work — or formally reassign supervisory responsibility to someone with capacity to perform it properly.

This page titled 14.8: Ethics HW Scenario 2 is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

14.9: Ethics HW Scenario 3

14.9 Ethics HW Scenario 3 — Safety Regulations and Professional Judgment

Scenario: Newly hired as a production engineer, you find workers routinely bypassing government-mandated machine safety features because following them slows production. The workers believe their shortcuts are still "adequately safe." Do you insist on immediate compliance, or does the workers' case justify looking the other way?

The tension: The workers have an efficiency argument and no history of incidents. As a new hire, confronting established practice is uncomfortable.

NCEES analysis: Government-mandated safety regulations are a legal floor. Bypassing them creates liability exposure for the company and for you as the engineer who observed and did not act. More importantly, the workers' self-assessment that their shortcuts are "adequately safe" is exactly the kind of undocumented assumption ethics requires you to challenge — the safety features were designed based on formal failure mode analysis of these specific machines. Individual workers are not in a position to override that analysis based on operational preference.

The non-option: ignoring the violations is not a professional option. If a worker is injured at a machine whose safety features you knew had been bypassed and you did not act, you have serious legal and ethical exposure — regardless of how long the practice preceded your arrival.

A more effective approach than simply demanding compliance: (1) document what you observed immediately; (2) understand the ergonomic concern; (3) escalate to management; (4) work toward solutions that achieve safety compliance and operational practicality. This produces durable compliance rather than compliance theater.

This page titled 14.9: Ethics HW Scenario 3 is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

14.10: Academic Integrity

14.10 Academic Integrity as Professional Foundation

Engineering ethics does not begin at graduation. It begins with the habits formed in coursework.

Academic misconduct — copying homework, sharing code without attribution, falsifying data, submitting group work as individual work — does more than violate course policy. It forms habits. The student who copies a homework problem because checking online is faster than working it out is practicing: *when the shortcut is available and detection risk is low, take the shortcut.*

That habit does not stop at graduation. It becomes the engineer who approves a design without checking it, or who copies a calculation from a previous project without verifying it applies to the current one. The three Ethics HW scenarios are the professional-scale version of the same shortcuts: claim temporary competence you don't yet have; trust your team without verifying; ignore a safety violation that hasn't caused problems yet.

Professional ethics is built from academic integrity — not because they are legally connected, but because they are psychologically and practically the same discipline.

This page titled 14.10: Academic Integrity is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

14.11: Citicorp Case Study

14.11 Case Study — William LeMessurier and the Citicorp Building (1978)

Background

In the late 1970s, structural engineer William LeMessurier designed the Citicorp Center skyscraper in Midtown Manhattan. The building had an unusual structure: supported from the midpoints of each face (not the corners), elevated on four massive nine-story stilts to accommodate a church at the base of one corner. A tuned mass damper at the top counteracted wind sway.

The structural system used a chevron (V-shaped) bracing pattern, analyzed under perpendicular wind loads hitting each face.

The Problem

After construction was complete, a civil engineering student contacted LeMessurier with a question about the design. LeMessurier did not dismiss the inquiry. He re-examined the original structural analysis and discovered something it had not fully addressed: **quartering winds** — winds arriving at 45° to the building face.

Because of the chevron bracing geometry, quartering winds produced higher stress in certain connections than perpendicular winds did. And those connections had been modified during construction: the contractor, for cost reasons, had substituted bolted joints for the welded joints originally specified. Bolted joints in this configuration were not adequate for worst-case quartering wind loads.

The analysis showed that under the worst-case quartering wind storm — with a return period of approximately 16 years in New York City — the building could fail. Not cosmetically. Catastrophically.

What LeMessurier Did

1. Re-ran the full analysis to confirm the findings before acting.
2. Immediately notified the building owner (Citicorp) and the New York City building department.
3. Organized a remediation plan — welding steel plates over the bolted joints throughout the building.
4. Remediation was conducted at night to avoid alarming occupants while ensuring safety was addressed.
5. Documented everything.

No collapse occurred. The remediation was completed. Years later, LeMessurier gave public lectures about the case, treating it as a lesson in engineering responsibility rather than a career embarrassment.

Real-World Example — Citicorp: Ethical Dimensions

Ethical dimensions of the Citicorp Building case study.

Ethical dimension	How it appeared in the case
Public welfare priority	Safety measures implemented despite serious cost and reputational risk
Transparency	Owner and city regulators informed promptly — no concealment
Assumption disclosure	The original design had not analyzed quartering winds; LeMessurier identified and disclosed this gap
Risk evaluation	Recognized failure potential under an off-design load case with non-negligible probability (~16-year return period)
Iterative responsibility	Continued analysis beyond original design scope when new information emerged
Professional courage	Reported a problem he could conceivably have hidden; the public welfare obligation overrode the personal cost

Discussion Questions

1. **Responsibility beyond compliance.** The original design met applicable codes. The quartering wind scenario was not explicitly required. How does this case extend engineering responsibility beyond "meeting standards"? What is the difference between legal compliance and ethical safety?
2. **Uncertainty and low-probability risk.** The worst-case quartering wind had a return period of ~16 years — a fairly common event for a building expected to stand decades. Why is analyzing low-probability but high-consequence scenarios an ethical obligation, not just a technical nicety?
3. **Transparency versus reputation.** LeMessurier's disclosure risked professional embarrassment and litigation. Concealment was practically available. Why is transparency ethically required even when personal consequences follow?
4. **Communication under pressure.** The remediation was conducted at night without public announcement. Was this

ethically defensible? Where is the line between responsible communication management and concealment?

5. **Lessons for you.** If you discovered a latent safety issue in a design already submitted and approved — in your own work — what steps would you be ethically obligated to take, and in what order?

This page titled 14.11: Citicorp Case Study is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

14.12: Environmental Responsibility

14.12 Environmental and Societal Responsibility

The NCEES framework's invocation of "public welfare" is intentionally broad. Engineering decisions affect communities near facilities, future generations who will use infrastructure, environmental systems affected by manufacturing and operation, and people who have no voice in decisions that affect them.

This dimension of engineering ethics has become more prominent as the consequences of large-scale engineering decisions — on climate, resource systems, and infrastructure equity — have become clearer. The obligation is the same as the more immediate safety obligations above: the engineer's responsibility is to the welfare of people affected by the work, whether or not they are present when the decision is made.

Key Takeaway

Engineering ethics is the discipline of making technical decisions responsibly — recognizing uncertainty, disclosing assumptions, maintaining safety margin, and prioritizing public welfare over convenience. The NCEES Model Rules establish the primary obligation: hold paramount the safety, health, and welfare of the public. The Citicorp case shows what that obligation looks like under pressure: disciplined re-analysis, voluntary disclosure, and remediation — carried out by an engineer who placed the obligation above his own reputation. The habits formed in academic work are the same habits that govern professional practice.

This page titled 14.12: Environmental Responsibility is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

14.13: Summary

Summary

- Engineering ethics is operational — it appears in margin decisions, disclosure decisions, approval decisions, and academic habits.
- The NCEES Model Rules establish the primary obligation: hold paramount the safety, health, and welfare of the public. This overrides employer pressure, cost, and schedule.
- Safety factors are an ethical acknowledgment of uncertainty. Reducing them without additional analysis transfers risk onto the public.
- Assumptions must be documented. A hidden assumption is misleading communication.
- The eight-step ethical decision-making framework mirrors the engineering problem-solving workflow.
- The three Ethics HW scenarios test honesty, supervisory responsibility, and safety compliance — each analyzed against NCEES principles.
- The Citicorp case demonstrates professional courage: voluntary disclosure, remediation at cost, and public accountability when concealment was available.
- Academic integrity and professional ethics are the same discipline at different scales.

This page titled 14.13: Summary is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

14.14: Quick Check

Quick Check

1. State the primary ethical obligation of an engineer under the NCEES Model Rules.
2. A safety factor is reduced from 2.0 to 1.5 to save cost without additional analysis. Identify the ethical problems.
3. An engineer's model assumes linear elastic behavior but the material will yield in service. The assumption is not stated in the report. What is the ethical problem?
4. Apply the eight-step framework to: a manager asks you to approve a design by end of day; you have not had time to fully review it.
5. In the SolidWorks scenario: is there any version in which claiming current competence is ethically permissible? Explain.
6. What is the difference between meeting the letter of an NCEES rule and fulfilling the ethical obligation behind it? Use the rubber-stamping scenario.
7. In the Citicorp case, LeMessurier discovered the problem himself. Would the ethical calculus change if a subordinate had discovered it and brought it to him?
8. Explain how copying a homework calculation connects to the supervisory ethics scenario in §14.8.

Answer Examples - click to expand

1. **Primary ethical obligation:** An engineer's primary ethical obligation is to hold public health, safety, and welfare paramount. Client, employer, schedule, and cost pressures are important, but they do not override the engineer's duty to protect the public.
2. **Safety factor reduction:** Reducing a safety factor from 2.0 to 1.5 is not automatically unethical, but doing so only to save cost and without additional analysis is ethically problematic.

   The main concerns are that the engineer may be allowing cost to override safety, making an unsupported technical judgment, failing to document the basis for the change, and exposing users or the public to increased risk without justification. A reduced safety factor would need to be supported by calculations, testing, applicable standards, and professional review.

3. **Unstated model assumption:** The ethical problem is that the report is misleading. If the analysis assumes linear elastic behavior but the material will yield in service, then the model does not represent the actual operating condition.

   Failing to state that limitation prevents reviewers, clients, or decision-makers from understanding the risk. The ethical issue is not just a technical modeling error; it is incomplete and potentially deceptive communication.

4. **Eight-step framework example:**
   1. **State the problem:** A manager wants approval by the end of the day, but the design has not been fully reviewed.
   2. **Identify stakeholders:** The engineer, manager, employer, client, end users, maintenance personnel, and the public.
   3. **Identify obligations:** The engineer must protect public safety, be truthful about the review status, and

work within their competence.
4. **Gather facts:** Determine what has been reviewed, what remains unchecked, what failure modes are possible, and whether any codes or safety-critical requirements apply.
5. **Generate options:** Refuse to approve, approve only the reviewed portions, request more time, escalate the concern, or issue a conditional status memo instead of approval.
6. **Evaluate consequences:** Approving too early could create safety risk and professional liability. Delaying may affect schedule but preserves technical integrity.
7. **Choose and justify action:** Do not approve the design as complete. Communicate that the review is incomplete and identify the remaining checks needed.
8. **Document and follow up:** Send a written response documenting what was reviewed, what was not reviewed, and what is required before approval.

5. **SolidWorks competence:** Claiming current competence is ethically permissible only if the person can truthfully perform the required SolidWorks work at the expected professional level for that task.

   For example, it may be acceptable to say, "I have prior SolidWorks experience and can become current with review and practice before taking responsibility for production work." It would not be acceptable to imply current professional competence if the person has not used the software recently and cannot yet complete the required tasks reliably.

6. **Letter of the rule vs. ethical obligation:** Meeting the letter of a rule means doing the minimum required by the written language. Fulfilling the ethical obligation means acting in the spirit of the rule: protecting the public, maintaining technical honesty, and taking responsibility for engineering judgment.

   In a rubber-stamping scenario, an engineer might try to claim they technically "reviewed" a document before signing it. But if the review was superficial and the engineer did not actually verify the work, then the ethical obligation has not been met. The issue is not just whether a signature was allowed; it is whether the engineer had sufficient knowledge and control to take responsibility for the design.

7. **Citicorp subordinate scenario:** The ethical calculus would not fundamentally change if a subordinate discovered the problem and brought it to LeMessurier. Once the issue was known, the obligation to investigate, disclose, and correct the risk would still exist.

   However, the situation would add supervisory responsibility. A senior engineer would need to take the subordinate's concern seriously, verify the analysis, protect the subordinate from retaliation, and ensure the concern reached the appropriate decision-makers. Ignoring or suppressing the subordinate's finding would make the ethical failure worse.

8. **Homework copying and supervisory ethics:** Copying a homework calculation trains the same bad habit that causes supervisory ethics failures: accepting work without understanding or verifying it.

   In school, copying may seem like a shortcut. In professional practice, relying on someone else's calculation without checking it can become rubber-stamping. Engineers are responsible for the work they submit, approve, or sign. The habit of verifying assumptions, units, methods, and results begins with student work.

This page titled 14.14: Quick Check is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

15: INTEGRATED ENGINEERING DESIGN CHALLENGE

The Bridge Challenge: An Integrated Design Case

Everything in this chapter is a tool you have already built. The Bridge Challenge is the integration test — the first time all of those tools apply to a single physical problem at the same time.

Learning Objectives

- Identify and apply every step of the engineering design framework to a physical build challenge.
- Convert the Bridge Challenge load requirement from pounds to Newtons and explain why unit discipline matters.
- Classify the Bridge Challenge specifications as hard or soft constraints and state what happens if each is violated.
- Explain qualitatively how tension and compression arise in a loaded bridge and identify which members experience each.
- Estimate a mass budget for a straw bridge and evaluate design alternatives against the 80 g constraint.
- Set up a spreadsheet parameter sweep relating straw count to estimated load capacity.
- Complete the Bridge Challenge report using the scientific report format.
- Map the tools used in the Bridge Challenge to the chapters in which each was introduced.

15.1: This Is Not a New Topic
15.2: Challenge Specifications
15.3: Define- Units, Loads, What "Best" Means
15.4: Model- Tension, Compression, Straws
15.5: Mass Budget
15.6: Parameter Sweep
15.7: Alternatives and Decision Matrix
15.8: Iterate
15.9: Bridge Report
15.10: Course Map
15.11: Summary
15.12: End-of-Chapter Problem Set

This page titled 15: Integrated Engineering Design Challenge is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

15.1: This Is Not a New Topic

15.1 This Is Not a New Topic

Everything in this chapter is a tool you have already built. The Bridge Challenge is the integration test — the first time all of those tools apply to a single physical problem at the same time.

The chapter follows the engineering design framework from Chapter 7. Each section takes one step of that framework and applies it concretely to the Bridge Challenge. The goal is not to tell you exactly how to build the bridge. The goal is to show you *how to think about it* the way an engineer would, before the first straw is cut or bent.

This page titled 15.1: This Is Not a New Topic is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

15.2: Challenge Specifications

15.2 The Challenge Specifications

List the rules explicitly in the Define step format from Chapter 6 before any analysis begins:

Bridge challenge specifications and hard constraints.

Symbol	Quantity	Value	Units	Type
L	Span	50	cm	**Hard**
w	Maximum width	<10	cm	**Hard**
m_{bridge}	Maximum bridge mass	80	g	**Hard**
F_{min}	Minimum load capacity	10	lbs	**Hard**
—	Materials	Jumbo straws + tape only		**Hard**
—	PVC support location	Center of span		**Hard (test condition)**

Every legitimate design must satisfy all six simultaneously. A bridge holding 15 lbs but massing 95 g fails. A bridge massing 60 g but spanning only 40 cm fails. The hard constraints define the feasible design space; everything else is performance competition.

This page titled 15.2: Challenge Specifications is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

15.3: Define- Units, Loads, What "Best" Means

15.3 Step 1 — Define: Units, Loads, and What "Best" Means

Convert the Load Requirement

The minimum load is stated in pounds. Every calculation in this course has used SI units. Chapter 5 applies immediately:

$$10 \text{ lb} \times \frac{4.448 \text{ N}}{1 \text{ lb}} = 44.5 \text{ N}$$

Including the testing apparatus (≈0.5 kg → ≈4.9 N), the total minimum structural load is approximately **49 N**.

The record load — 24.25 lbs — converts to:

$$24.25 \times 4.448 = 107.8 \text{ N}$$

The record tells you something real is achievable: the constraint floor is 44.5 N; the record is 107.8 N. Your problem is to get as high as possible within the mass and geometry constraints.

Defining an Efficiency Metric

A useful single metric is the **load-to-mass ratio**:

$$\text{Efficiency} = \frac{F_{\text{max load}}}{m_{\text{bridge}}} \quad \left(\frac{\text{lb}}{\text{g}}\right)$$

Defining this metric before building means you have an objective basis for comparing designs — exactly what the design framework requires.

Hard vs. Soft Constraints

Hard and soft constraints for the bridge challenge.

Constraint	Type	What happens if violated
Span = 50 cm	**Hard**	Bridge does not reach both supports — automatically disqualified

Constraint	Type	What happens if violated
Width ≤ 10 cm	**Hard**	Disqualified
Mass ≤ 80 g	**Hard**	Disqualified — weighed before testing
Load ≥ 10 lbs	**Hard**	Challenge requirement not met
Load maximized	**Soft**	Not a rule — it's what you're competing on
Mass minimized	**Soft**	Lower mass with same load = higher efficiency

This page titled 15.3: Define- Units, Loads, What "Best" Means is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

15.4: Model- Tension, Compression, Straws

15.4 Step 2 — Model: Tension, Compression, and What Straws Can Do

Where the Forces Go

When a load is applied to the center of a bridge, the members on the **bottom** are in **tension** (being pulled apart) and the members on the **top** are in **compression** (being pushed together).

For straws specifically: tension members are reasonably strong. Compression members are vulnerable to **buckling** — a sudden sideways collapse that happens at loads well below the material yield stress. A **bundle** of straws taped or glued together resists buckling much better than a single straw, because the increased cross-section raises the buckling load dramatically.

Simplified Axial Stress Model

From Chapter 7: $\sigma = F/A$, so $F_{\max} = \sigma_{\text{allow}} \times A$. For a jumbo straw with outer diameter ≈ 8 mm and wall thickness ≈ 0.25 mm:

$$A_{\text{wall}} = \pi \times d_{\text{outer}} \times t \approx \pi \times 8 \times 0.25 \approx 6.3 \text{ mm}^2$$

For a bundle of n straws: $A_{\text{bundle}} = n \times 6.3 \text{ mm}^2$. This model ignores buckling and bending — see the Watch Out below.

⚠ Watch Out — Model Limitations

1. **Buckling is ignored.** Real compression members fail by buckling at loads well below the axial stress limit. Bundling and keeping compression members short both help.
2. **Bending is ignored.** The PVC support creates bending in the deck and primary members, adding stress beyond what the axial model predicts.
3. **Straw properties are estimated.** Your actual straws may have different dimensions. Measure them before computing.

Despite these limitations, the model gives a useful basis for comparing configurations — which is what models are for.

This page titled 15.4: Model- Tension, Compression, Straws is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

15.5: Mass Budget

15.5 Step 3 — Mass Budget and the 80 g Constraint

Working conservatively at 2.5 g per straw and 3 g of tape total:

$$n_{\max} = \frac{80 - 3}{2.5} = \frac{77}{2.5} \approx 30 \text{ straws}$$

This matches the "approximately 30 straws" in the challenge document — not coincidence. At maximum straw count, you are right at the mass limit. Every straw used in one location is unavailable elsewhere. That is the fundamental trade-off.

Example mass budget allocation

Example mass budget allocation for a straw bridge design.

Component	Straws	Mass (g)
Top chord (compression)	4	10
Bottom chord (tension)	4	10
Diagonal bracing	8	20
Deck (load distribution)	6	15
Tape	—	3
Total	**22 straws**	**58 g**

This hypothetical design uses 22 of 30 straws and masses 58 g, leaving 22 g (≈8–9 straws) unused. Does adding them increase load capacity enough to justify the mass? That is what the parameter sweep in §15.6 explores.

This page titled 15.5: Mass Budget is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

15.6: Parameter Sweep

15.6 Step 4 — Analyze: Parameter Sweep for Load vs. Straw Count

Spreadsheet setup — Chapter 8 structure

```
Inputs block:
A1: Wall thickness (mm) B1: 0.25
A2: Outer diameter (mm) B2: 8
A3: Single straw wall area (mm²) B3: 6.3
A4: Allowable stress estimate (MPa) B4: 50

Sweep table (row 6 onward):
C1: n D6: Bundle Area (mm²) E1: Est. Force (N) F1: Est. Force (lbs)

C2: 1 D7: =C7*$B$3 E2: =D7*$B$4 F2: =E7/4.448
C3: 2 D8: =C8*$B$3 E3: =D8*$B$4 F3: =E8/4.448
... (continue through C11 for n = 10)
```

Row 5 is left blank as a visual separator between the inputs block and the sweep table — consistent with the modeling discipline introduced in Chapter 8. Column A holds labels, column B holds constant values. Columns C through F contain the sweep table, with relative references in column C and absolute references to the inputs block in columns D and E.

Illustrative results based on estimated straw dimensions

Illustrative load estimates for straw bundle size in the bridge challenge.

Straws (n)	Bundle Area (mm^2)	Est. Force (N)*	Est. Force (lbs)*
1	6.3	315	70.8
2	12.6	630	141.6
4	25.2	1260	283.2
6	37.8	1890	424.8

** These values are artificially high because buckling is not modeled. Use them for relative comparisons between configurations, not as absolute*

predictions.

The graph shape is **linear** — each additional straw contributes equally to estimated capacity. This means there is no geometric diminishing returns in the axial model. The practical limits come from geometry: not all straws can be aligned with the primary load path, and the mass constraint caps the total count at approximately 30.

🔗 Connection — Chapter Tools Used So Far

At this point in the Bridge Challenge analysis, you have directly applied:

- **Chapter 5** — unit conversion: lbs → N, mm → m, mm^2 → m^2
- **Chapter 6** — structured problem solving: Define table with symbols, quantities, values, units
- **Chapter 7** — design framework: constraint classification, design objective, parameter comparison
- **Chapter 8** — spreadsheet modeling: inputs block, parameter sweep, XY scatter plot

This page titled 15.6: Parameter Sweep is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

15.7: Alternatives and Decision Matrix

15.7 Step 5 — Generate Alternatives and Evaluate Trade-offs

Three Geometry Families

Flat truss (beam bridge): Top and bottom horizontal chords connected by diagonal or vertical web members. Bottom chord in tension; top chord in compression; diagonals carry mixed loads. The most intuitive design for most students.

Warren truss: Alternating diagonal members with no verticals. Load paths are clean and distributed among more members than a simple beam. Generally more efficient by mass for the same load.

Suspension / inverted arch: Deck hung below tension cables. Tension-dominated geometry is advantageous for straws. The challenge span-to-support geometry may limit how much sag is achievable, but this family rewards creative exploration.

Decision Matrix

Decision matrix comparing bridge geometry alternatives.

Criterion	Weight	Flat beam	Warren truss	Suspension
Load capacity potential	40%	2	4	3
Mass efficiency	25%	3	4	3
Build complexity (lower = better)	20%	4	3	2
Meets geometry constraints	15%	4	4	3
Weighted total		**2.85**	**3.75 ✓**	**2.75**

The matrix makes the decision basis explicit and auditable. The Warren truss scores highest for *this particular weighting*. If your team values build simplicity more highly (limited build time), recompute with different weights — the matrix is the tool, not the verdict.

This page titled 15.7: Alternatives and Decision Matrix is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

15.8: Iterate

15.8 Step 6 — Iterate: What Changes Between Design 1 and Design 2

Design iteration is not starting over. It is updating one decision based on what analysis or testing reveals.

- **Cycle 1 — Mass overage:** Design estimates 88 g. Identify lowest-contributing member, remove it, recheck mass and capacity.
- **Cycle 2 — Weak joint:** Prototype fails at a diagonal-to-chord joint. Reinforce that joint, or redesign geometry to reduce stress concentration there.
- **Cycle 3 — Load concentration:** The PVC pipe applies a concentrated center load. If the deck fails locally before the main structure, add deck members spanning the width directly below the load point.

Each cycle follows: observe → hypothesize → change one thing → re-evaluate. Same pattern as the safety factor iteration in Chapter 7 and the spreadsheet parameter sweep in Chapter 8.

This page titled 15.8: Iterate is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

15.9: Bridge Report

15.9 Step 7 — Communicate: The Bridge Report

The Bridge Challenge report uses **scientific report format**, which differs from the technical memo:

Comparison of technical memo and scientific report sections.

Section	Technical memo	Scientific report
Opening	Purpose	Abstract
Context	Background	Introduction
How it was done	Methodology	Materials and Methods
What happened	Results	Results
What it means	Analysis	Discussion
What to do	Conclusion & Recommendation	Conclusion

The scientific report is appropriate here because you are reporting *what you did and observed*. The Bridge Challenge is an experiment: you predicted behavior, built a structure, applied load, and observed what happened.

What to include in each section: Title page (team, date, section). Abstract (challenge, design approach, load achieved, outcome — 3–5 sentences). Introduction (why this design, what you expected, the mass budget and decision matrix from §§15.5–15.7). Materials and Methods (straw count, tape used, construction approach, diagram or photo). Results (actual failure load in lbs and N, final bridge mass, efficiency ratio). Discussion/Conclusion (which members stretched, which compressed, where failure occurred, what you would change, whether more straws would help and where).

> **Ethics Check — Reporting Actual Results**
>
> If your bridge fails at 8 lbs — below the 10 lb minimum — report 8 lbs. Not a rounded-up approximation. Not an ambiguous description of the failure mode.
>
> The record of 24.25 lbs exists because everyone reports actual results. More broadly: engineering credibility depends on accurate failure reporting. The Mars Climate Orbiter, the Challenger O-ring data, the Citicorp analysis — each case produced an engineering lesson because someone reported what actually happened. Accurate failure reporting is how the engineering profession learns, and it is how your design skills improve.

This page titled 15.9: Bridge Report is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

15.10: Course Map

15.10 Course Map — Every Tool in One Challenge

Course concepts applied throughout the Bridge Challenge.

Chapter	Tool / Concept	Where it appeared in the Bridge Challenge
5	Unit conversion	lbs → N for load; g → kg for mass; mm → m for stress
6	Structured problem solving	Define table (§15.2); model equations (§15.4); interpret results (§15.8)
7	Engineering design framework	Objective definition; hard/soft constraints; alternatives; decision matrix; iteration
8	Spreadsheet modeling	Inputs block; straw count parameter sweep; XY scatter plot
9	Data visualization	Load vs. straw count graph; failure load in Results with labeled axes and units
10	MATLAB (optional)	Automate the parameter sweep; logical indexing to flag mass-budget violations
12	CAD (optional)	Model bridge geometry in Fusion; compute member volume and mass to verify mass budget
13	Engineering communication	Scientific report format; quantitative language throughout all sections
14	Engineering ethics	Accurate failure reporting; safety factor as ethical design choice

Key Takeaway

The Bridge Challenge is not separate from the textbook. It is the textbook in physical form. Unit conversion, constraint classification, mass budget calculation, parameter sweep, decision matrix, iteration, scientific report, accurate failure reporting — every one of these is a tool this course gave you. The challenge asks: can you apply them all at once, to a real problem, with physical consequences? That is what engineering does.

This page titled 15.10: Course Map is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

15.11: Summary

Summary

- Bridge Challenge specs define a constrained design problem: four hard constraints and two soft optimization objectives.
- 10 lbs ≈ 44.5 N; record 24.25 lbs ≈ 107.8 N. Unit discipline (Ch 5) is required before any comparison.
- Bottom chord is in tension; top chord is in compression. Compression members are more vulnerable to buckling; bundled straws help.
- Mass budget: ≈30 straws at the 80 g limit. Every straw used in one location is unavailable elsewhere.
- Straw-count parameter sweep shows linear scaling in the axial model. The plot reveals the trend; model limitations must be stated.
- Decision matrix (Ch 7) compares flat beam, Warren truss, suspension against weighted criteria. The matrix is auditable; the weights are your team's choice.
- Scientific report format (not technical memo) is appropriate for experimental work: Title, Abstract, Introduction, Materials and Methods, Results, Discussion/Conclusion.
- Accurate failure reporting is an engineering ethics obligation.

This page titled 15.11: Summary is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

15.12: End-of-Chapter Problem Set

End-of-Chapter Problem Set

1. Convert 10 lbs and 24.25 lbs to Newtons. Show all unit conversion steps.
2. A bridge masses 63 g and holds 18.5 lbs. Compute efficiency (lbs/g). A second bridge masses 71 g and holds 21 lbs. Which is more efficient? Which held more absolute load?
3. Classify each as hard constraint or soft objective: (a) 50 cm span, (b) maximum load achieved, (c) mass under 80 g, (d) fewest straws used.
4. A straw has outer diameter 8 mm and wall thickness 0.25 mm. Compute wall area in mm². For a bundle of 6 straws with allowable stress 45 MPa, compute estimated axial capacity in N and lbs.
5. **Spreadsheet exercise.** Build the parameter sweep from §15.6 for straw count 1 to 12. Plot straw count vs. estimated load in lbs. Write a two-sentence interpretation and one model limitation.
6. Explain why compression members are more likely to fail by buckling than tension members are by fracture for the same area. What design strategy delays buckling?
7. In the decision matrix in §15.7, the weight on build complexity is 20%. If a team increases this weight to 40% (reducing load capacity to 25% and mass efficiency to 20%, geometry stays 15%), recompute weighted totals. Which design wins?
8. *[Post-challenge]* Your bridge failed at a diagonal-to-chord joint. Apply one iteration cycle: state the change, explain why it addresses the failure mode, and estimate the mass impact.
9. Using the parameter sweep model, write two paragraphs addressing: (a) would adding straws make the bridge stronger? and (b) under what conditions would adding straws NOT help?
10. Map these activities to their source chapters: (a) converting 10 lbs to N, (b) classifying constraints, (c) building the straw-count sweep, (d) plotting the sweep, (e) using the decision matrix, (f) writing the scientific report.

Answer Examples — click to expand

1. **Convert pounds-force to newtons.** Use $1\ \text{lb}_\text{f} = 4.44822\ \text{N}$.

   $$10\ \text{lb}_\text{f} \times \frac{4.44822\ \text{N}}{1\ \text{lb}_\text{f}} = 44.4822\ \text{N} \approx 44.5\ \text{N}$$

   $$24.25\ \text{lb}_\text{f} \times \frac{4.44822\ \text{N}}{1\ \text{lb}_\text{f}} = 107.869\ \text{N} \approx 108\ \text{N}$$

2. **Compute bridge efficiency.**

   Bridge B: $\frac{21\ \text{lb}}{71\ \text{g}} = 0.2958\ \text{lb/g}$

   Conclusion: Bridge B is slightly more efficient because $0.2958\ \text{lb/g} > 0.2937\ \text{lb/g}$. Bridge B also held the greater absolute load because $21\ \text{lb} > 18.5\ \text{lb}$.

3. **Hard constraints vs. soft objectives:**

Item	Classification	Reason
50 cm span	Hard constraint	The bridge must span the required distance.
Maximum load achieved	Soft objective	Higher is better, but it is optimized rather than simply pass/fail.
Mass under 80 g	Hard constraint	A bridge above the limit does not meet the rules.
Fewest straws used	Soft objective	Using fewer straws may improve efficiency, but it is a design preference unless stated as a rule.

4. **Estimate straw wall area and bundle capacity.**

   $A_{\text{wall}} = \pi(8\ \text{mm})(0.25\ \text{mm}) = 6.28\ \text{mm}^2$

   For 6 straws:

   $A_{\text{bundle}} = 6(6.28\ \text{mm}^2) = 37.7\ \text{mm}^2$

   Since $1\ \text{MPa} = 1\ \text{N/mm}^2$:

   $F_{\max} = \sigma_{\text{allow}} A = (45\ \text{N/mm}^2)(37.7\ \text{mm}^2) = 1696\ \text{N}$

   Convert to pounds-force:

   $$1696\ \text{N} \times \frac{1\ \text{lb}_\text{f}}{4.44822\ \text{N}} = 381\ \text{lb}_\text{f}$$

   Important interpretation: This is an ideal axial stress estimate. It will likely overpredict real compression capacity because it ignores buckling, joints, bending, imperfect construction, and load distribution.

5. **Spreadsheet sweep answer example.** Using the same model from Problem 4, one straw has an estimated axial capacity of:

   $F_1 = (45\ \text{N/mm}^2)(6.28\ \text{mm}^2) = 282.7\ \text{N}$

   $$282.7\ \text{N} \times \frac{1\ \text{lb}_\text{f}}{4.44822\ \text{N}} = 63.6\ \text{lb}_\text{f}$$

   Therefore, the ideal model predicts:

   $F_{\text{bundle}} \approx 63.6n\ \text{lb}_\text{f}$, where n is the number of straws.

Straw Count	Estimated Load (lb)
1	63.6
2	127.1
3	190.7
4	254.2

Straw Count	Estimated Load (lb)
5	317.8
6	381.3
7	444.9
8	508.4
9	572.0
10	635.5
11	699.1
12	762.6

Interpretation example: The ideal axial model predicts a linear increase in estimated load as straw count increases. Each added straw contributes about $63.6\ \text{lb}$ of ideal axial capacity, so the graph should be a straight line.

Model limitation: The model assumes all straws carry load equally in pure axial tension or compression. Real bridges often fail first by buckling, bending, joint separation, or uneven load sharing.

6. **Compression vs. tension failure:** Tension members fail when the material is pulled apart and the tensile stress exceeds the material capacity. Compression members can fail earlier because they may buckle sideways before the material itself crushes or yields.

   Design strategies that delay buckling: Make compression members shorter, brace them laterally, bundle straws together, improve joint alignment, and use triangulated geometry so members carry mostly axial loads instead of bending loads.

7. **Decision matrix recomputation:** Use the same raw scores from §15.7, but replace the original weights with the revised weights:

   Load capacity: 25%

   Mass efficiency: 20%

   Geometry: 15%

   Build complexity: 40%

   For each design:

   $$\text{Weighted Total} = 0.25(\text{Load Score}) + 0.20(\text{Mass Efficiency Score}) + 0.15(\text{Geometry Score}) + 0.40(\text{Build Complexity Score})$$

   Interpretation: Increasing build complexity to 40% favors the design that is easiest and most reliable to construct. If the original winning design had high load capacity but was difficult to build, it may lose under this revised weighting. The winning design is the one with the highest recomputed weighted total.

   Instructor note: Students should use the actual raw scores from the §15.7 decision matrix to complete the final numerical comparison.

8. **Iteration example:** If the bridge failed at a diagonal-to-chord joint, one design change would be to increase

the overlap length at that joint and add a small gusset or wrap of tape around the connection.

This addresses the failure mode because the joint, not the straw body, was the weak point. Increasing bonded area spreads the force over a larger region and reduces local peeling or tearing. The mass impact would likely be small, perhaps $2\ \mathrm{g}$ to $5\ \mathrm{g}$, depending on how much tape or glue is added. The trade-off is that the bridge becomes slightly heavier, but the load capacity may increase significantly if the joint was the controlling failure point.

9. **(a) Would adding straws make the bridge stronger?** In the simplified parameter sweep model, yes. Adding straws increases total cross-sectional area, and the axial load capacity is modeled as $F_{\max} = \sigma_{\text{allow}} A$. If the allowable stress stays constant and each straw shares load equally, then adding straws increases the predicted capacity linearly.

   (b) When would adding straws not help? Adding straws will not help if the bridge is not failing because of axial stress in the straw walls. For example, if failure occurs by joint separation, buckling, bending, twisting, poor alignment, or load concentration at the support, then more straw area may not fix the actual failure mode. Additional straws can also add mass, make construction more difficult, and create imperfect joints, so the efficiency may decrease even if the bridge becomes stronger in absolute load.

10. **Source chapter mapping:**

Activity	Source Chapter	Reason
Converting 10 lbs to N	Chapter 5	Units and dimensional analysis
Classifying constraints	Chapter 7	Engineering design process and constraints
Building the straw-count sweep	Chapter 8	Spreadsheet modeling and parameter sweeps
Plotting the sweep	Chapter 9	Engineering graphs and visualization
Using the decision matrix	Chapter 7	Comparing design alternatives using weighted criteria
Writing the scientific report	Chapter 13	Engineering communication and technical reporting

This page titled 15.12: End-of-Chapter Problem Set is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

16: APPENDICES

16.1: Appendix A — Standard Symbols and Units
16.2: Appendix B — Unit Conversion Reference Table
16.3: Appendix C — MATLAB Quick Reference
16.4: Appendix D — Spreadsheet Quick Reference (Excel)
16.5: Appendix E — Technical Memo Template

This page titled 16: Appendices is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

16.1: Appendix A — Standard Symbols and Units

Appendix A — Standard Symbols and Units

This appendix defines the standard symbols and units used throughout the text. Unless otherwise stated, SI units are used.

A.1 Mechanical and Structural Quantities

Standard symbols and SI units for mechanical and structural quantities.

Symbol	Quantity	SI Unit	Notes
m	Mass	kg	
g	Gravitational acceleration	m/s²	Standard: 9.81 m/s²
F	Force	N	
T	Tension	N	Force in a cable or member
W	Weight	N	$W = mg$
L	Length / span	m	
A	Cross-sectional area	m²	Often expressed in mm²
V_{vol}	Volume	m³	Often in mm³ or cm³
ρ	Density	kg/m³	Steel: 7850 kg/m³ = 7.85×10⁻³ g/mm³
σ	Normal stress	Pa (N/m²)	Often expressed in MPa
SF	Safety factor	dimensionless	$SF = \sigma_{\text{allow}}/\sigma_{\text{actual}}$
θ	Angle	degrees (°)	Convert to radians for trig functions

A.2 Electrical Quantities

Standard symbols and SI units for electrical quantities.

Symbol	Quantity	SI Unit	Notes
V	Voltage	V (volts)	Potential difference
I	Current	A (amperes)	Rate of charge flow
R	Resistance	Ω (ohms)	$\Omega = \mathrm{V/A}$
R_{eq}	Equivalent resistance	Ω	Series or parallel combination
P	Power	W (watts)	$P = VI = I^2R = V^2/R$
V_F	LED forward voltage	V	Typically 1.8–3.3 V depending on color

A.3 Computational Conventions

Common computational notation used in Excel and MATLAB.

Notation	Meaning	Tool
`$B$1`	Absolute cell reference — does not shift when copied	Excel
`B1`	Relative cell reference — shifts when copied	Excel
`RADIANS(x)`	Convert degrees to radians — wrap all trig arguments	Excel
`./ .* .^`	Element-wise division, multiplication, power	MATLAB
`sind(x)`	Sine of x in degrees — preferred for this course	MATLAB

A.4 SI Prefixes

Common SI prefixes and factors used in engineering calculations.

Prefix	Symbol	Factor	Common use
Mega	M	10^{6}	Stress (MPa)
kilo	k	10^{3}	Resistance (kΩ), mass (kg)
milli	m	10^{-3}	Current (mA), length (mm)
micro	μ	10^{-6}	Capacitance (μF)
nano	n	10^{-9}	Transistor dimensions

This page titled 16.1: Appendix A — Standard Symbols and Units is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

16.2: Appendix B — Unit Conversion Reference Table

Appendix B — Unit Conversion Reference

Convert to consistent units **before** substituting into equations. The most common errors: forgetting to convert mm² to m² (factor of 10^{-6}), mixing lbf and N in stress calculations, and working with mA instead of A in power calculations.

B.1 Length

Common length conversion factors.

From	To	× by
mm	m	0.001
cm	m	0.01
in	m	0.0254
ft	m	0.3048
in	mm	25.4

B.2 Mass

Common mass conversion factors.

From	To	× by
g	kg	0.001
lb (mass)	kg	0.4536
kg	g	1000

B.3 Force

Common force conversion factors.

From	To	× by
lb (force)	N	4.448
N	lb (force)	0.2248

B.4 Area

Common area conversion factors.

From	To	× by
mm^2	m^2	10^{-6}
in^2	mm^2	645.2

B.5 Stress and Pressure

Common stress and pressure conversion factors.

From		To	× by
Pa		MPa	10^{-6}
psi		MPa	0.006895
ksi		MPa	6.895

B.6 Electrical Prefixes

Common electrical prefix conversions.

Prefix	Example	Equivalent
kilo (k)	5 kΩ	5000 Ω
milli (m)	20 mA	0.020 A
micro (μ)	100 μF	0.0001 F

This page titled 16.2: Appendix B — Unit Conversion Reference Table is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

16.3: Appendix C — MATLAB Quick Reference

Appendix C — MATLAB Quick Reference

C.1 Standard Script Header

```
clear
clc
close all

% Your Name
% ENGR 110 — Script Description
% Date
```

C.2 Variables, Comments, Sections

```
m = 1000;          % mass, kg          ← semicolon suppresses output
g = 9.81;          % m/s²
theta = 30;        % degrees

%% Inputs          ← %% marks a section (Ctrl+Enter runs one section)

T = (m*g) / (2*sind(theta));    % cable tension, N
fprintf('Tension = %.2f N\n', T);
```

C.3 Vectors and Parameter Sweeps

```
theta = 10:5:80;               % [10, 15, 20, ..., 80]
T = (m*g) ./ (2*sind(theta));  % ./ = element-wise — REQUIRED for vectors
P = V.^2 / R;                  % .^ = element-wise power
```

C.4 Trigonometry

MATLAB trigonometric functions for degree and radian inputs.

Function	Input	Use when
`sind(x)`	degrees	**Preferred for this course** — no conversion needed
`cosd(x)`, `tand(x)`	degrees	Same — use d-suffix for degree inputs
`sin(x)`	radians	Only when angle is already in radians. `sin(30)` ≈ −0.988 — not sin(30°)

C.5 Plotting

```
figure(1)
plot(theta, T, 'bo-')         % blue circles, connected line
xlabel('Angle (degrees)')
ylabel('Tension (N)')
title('Cable Tension vs. Angle')
grid on

hold on                        % add a second curve to same axes
plot(theta, T2, 'rs--')
legend('Design A', 'Design B')

yline(0.25, 'r--', '0.25 W limit')  % horizontal constraint line
```

C.6 Conditional Logic and Logical Indexing

```
if max(T) > 12000
    fprintf('WARNING: Tension %.0f N exceeds limit.\n', max(T))
else
    disp('All angles within safe tension limit.')
end

safe_angles = theta(T < 12000);       % extract values satisfying condition
min_safe = min(safe_angles);
fprintf('Min safe angle: %.1f degrees\n', min_safe)
```

C.7 Common MATLAB Mistakes

Common MATLAB mistakes, symptoms, and fixes.

Mistake	Symptom	Fix
Forgetting `;`	Command window floods	Add `;` after assignments
`/` instead of `./`	Wrong scalar result or error	Use `./` for element-wise division
`sin()` instead of `sind()`	Silent wrong answer	Use `sind()` for degree inputs
Hard-coded constants	Model breaks when inputs change	Assign named variables; reference them
Unlabeled plots	Graph conveys nothing	Always `xlabel`, `ylabel`, `title`

This page titled 16.3: Appendix C — MATLAB Quick Reference is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

16.4: Appendix D — Spreadsheet Quick Reference (Excel)

Appendix D — Spreadsheet Quick Reference (Excel / Microsoft 365)

D.1 Three-Region Structure

Three-region structure for disciplined spreadsheet models.

Region	Contents	Rule
Inputs	Named parameters with values and units	One value per cell; label and unit in adjacent cell
Model equations	Formula cells only	Reference input cells — no hardcoded numbers
Outputs / results	Computed values, graphs	Labeled; every graph meets the five requirements

D.2 Parameter Sweep Layout

A1: Mass, m (kg) B1: 1000 A2: Gravity, g (m/s²) B2: 9.81 A10: 10 B10: =($B$1*$B$2)/(2*SIN(RADIANS(A10))) A11: 15 B11: ← copy B10 downward ... ← $B$1,$B$2 stay fixed; A10 shifts to A11,A12,... A24: 80

D.3 Key Functions

Key Excel functions used in engineering spreadsheet models.

Function	Syntax	Use
`RADIANS()`	`=RADIANS(angle)`	**Required** before SIN/COS/TAN — angles must be in radians
`IF()`	`=IF(cond, true, false)`	Piecewise conditional — two branches
`IFS()`	`=IFS(c1,v1,c2,v2,...,TRUE,def)`	Multiple conditions — cleaner than nested IF

D.4 Graph Requirements Checklist

Excel graph requirements for engineering communication.

Requirement	How to apply in Excel
Chart type: XY scatter	Insert → Charts → Scatter (not Line, not Bar)
Title	Chart Design → Add Chart Element → Chart Title
X-axis label with units	Axis Titles → Primary Horizontal
Y-axis label with units	Axis Titles → Primary Vertical
Y-axis starts at zero	Format Axis → Minimum = 0 (unless stated reason)
No 3D effects	Avoid all 3D chart subtypes

This page titled 16.4: Appendix D — Spreadsheet Quick Reference (Excel) is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

16.5: Appendix E — Technical Memo Template

Appendix E — Technical Memo Template (Blank)

For the fully annotated example with all sections completed, see **Chapter 13 §13.4**.

ENGINEERING MEMORANDUM

To: _____

From: _____

Date: _____

Subject: _____

---

1. Purpose *(What system? What question? What constraints? 2–4 sentences.)*

---

2. Background *(Physical system, parameters, governing relationships, and all assumptions — stated explicitly here.)*

---

3. Methodology *(Equations used, tools, parameter ranges, simplifications. Another engineer must be able to replicate from this section.)*

---

4. Results *(Key numerical values, labeled graphs, summary tables. Reference every figure: "As shown in Figure 1...")*

---

5. Analysis *(Interpret results: physically reasonable? Thresholds exceeded? Trade-offs? Numbers → engineering judgment.)*

6. Conclusion and Recommendation *(Decision, justification, limitations, next steps. "The power is 0.436 W" is a Result. "The 330 Ω resistor is unsafe and must be replaced" is a Conclusion.)*

This page titled 16.5: Appendix E — Technical Memo Template is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by .

INDEX

A

absolute reference
8.3: Absolute vs Relative References

absolute references
8: Spreadsheet Modeling
8.10: Summary
8.7: Six Common Spreadsheet Errors

academic
3.1: Engineering Is a Structured Academic Path
3.7: Academic Risk Factors

academic integrity
14.10: Academic Integrity

academic misconduct
14.10: Academic Integrity

academic pathways
3: Academic Pathways and Transfer Strategy

academic performance
2.6: Choosing a Discipline
4: Engineering Success Systems
4.8: Summary

academic planning
3.8: Treating Your Academic Plan Like an Engineering Problem

academic success
3.6: Transfer Strategy
3.8: Treating Your Academic Plan Like an Engineering Problem
4.1: It Is Not About Talent
4.2: Understanding Your Real Workload
4.5: Goals, Humility, and the Long Game
4.7: Sustaining Yourself Over the Long Haul
4.9: End-of-Chapter Exercise

accelerating increase
9.4: Linear vs Nonlinear Trends

accurate information
14.7: Ethics HW Scenario 1

active learning
4.3: How to Actually Study Engineering
4.8: Summary

addition
5.2: Quantities vs. Numbers

advanced chart features
9.9: Advanced Chart Features

aeronautical engineering
1.3: Engineering vs Science vs Technology

aerospace
1.3: Engineering vs Science vs Technology
1.7: Major Disciplines at a Glance
2.8: Summary

aerospace engineering
1.6: What Engineers Actually Do
2.2: The Major Engineering Disciplines
3.4: Typical Lower-Division Engineering Path (Community College)
7.8: Evaluate Trade-offs

aerospaceengineering
3.9: Summary

algebra
3.1: Engineering Is a Structured Academic Path
3.2: The Mathematics Backbone
3.7: Academic Risk Factors

algebraic error
4.4: Understanding Your Errors

algebraic relationships
8.9: When Spreadsheets Are Not Enough

algorithm development
8.6: Conditional Formatting and IF Logic

allocation
15.5: Mass Budget

alternatives
7.10: Communicate
7.12: Summary
7.7: Model and Analyze
7.8: Evaluate Trade-offs

ammeter
11.5: Schematic Symbols

Amount of Substance
5.3: SI System — Base and Derived Units

analysis
1.10: Summary
1.4: What Engineers Actually Do
10.2: MATLAB as Structured Engineering
12.8: CAD in the Engineering Landscape
13.11: Summary
13.3: The Technical Memo
13.4: Complete Memo Example
15.9: Bridge Report
16.5: Appendix E — Technical Memo Template
2.7: One Obligation That Every Discipline Shares
3.8: Treating Your Academic Plan Like an Engineering Problem
6: Structured Engineering Problem Solving
6.1: Why Structure Matters
6.2: Full Workflow Worked Example First
6.5: Step 3 — Analyze
7.1: Analysis vs Design
7.10: Communicate
7.11: Design, Ethics, and Safety Factors
7.12: Summary
7.13: Quick Check
7.2: What Makes a Design Problem
7.5: Identify Constraints
7.7: Model and Analyze
7.9: Iterate and Refine
8.1: From Calculator to Model

analysis assumptions
13.10: Communicating Uncertainty and Limitations

analysis documentation
13.2: Three Modes of Communication

analysis methods
7.6: Generate Alternatives

analysis problems
7: The Engineering Design Process

analysis strategy
11.11: Mixed Circuits

analysis tool
9.8: Visualization and Interpretation

analysis tools
9.1: Why Graphs Are an Engineering Tool

Analyze
6.10: Structure and Professional Accountability
6.9: The Workflow Across Disciplines

anatomy
9.3: Anatomy of an Engineering Graph

angle
8.1: From Calculator to Model
8.3: Absolute vs Relative References
8.4: Parameter Sweeps

angle sweep
10.11: End-of-Chapter Problem Set

angles
10.4: Four-Step Build
10.5: Trigonometry in MATLAB
6.6: Step 4 — Interpret + Sensitivity

appendix
16: Appendices
16.1: Appendix A — Standard Symbols and Units
16.5: Appendix E — Technical Memo Template

appliances
11.7: Parallel Circuits

application timeline
3.6: Transfer Strategy

applications engineering
2.5: Functional Roles

applied communication
13.8: TEGO Case Study

applied exercises
13.12: End-of-Chapter Applied Exercises

applied problems
5.11: End-of-Chapter Problem Set

area
16.2: Appendix B — Unit Conversion Reference Table

arithmetic
5.2: Quantities vs. Numbers
6.5: Step 3 — Analyze

ASME drawings
12.9: Summary

ASME engineering drawing
12: Parametric CAD and 3D Modeling

ASME standard
12.5: The Fusion Project

associate degree
3: Academic Pathways and Transfer Strategy
3.5: Degree Levels and Transfer Destinations

assumptions
14.5: Modeling Assumptions
5.8: Order-of-Magnitude Reasoning
6.1: Why Structure Matters
7.11: Design, Ethics, and Safety Factors

assumptions in engineering
14.13: Summary

audience
10.9: Choosing the Right Tool
13.7: Technical Presentations

audience communication
13.2: Three Modes of Communication

audience presentation
13.8: TEGO Case Study

auditable documents
6.7: Step 5 — Communicate

auditing
6.10: Structure and Professional Accountability

automation
10: MATLAB for Engineering Computation
10.1: Why Engineers Program

AVC
3.4: Typical Lower-Division Engineering Path (Community College)
AVC counseling
3.6: Transfer Strategy
awareness
2.1: The Question Most Students Ask
axial capacity
15.12: End-of-Chapter Problem Set
axial stress
15.4: Model- Tension, Compression, Straws
axis labels
16.4: Appendix D — Spreadsheet Quick Reference (Excel)
9.10: Graph Checklist — Quick Reference
axis scale
9.8: Visualization and Interpretation

B

bachelor's degree
3: Academic Pathways and Transfer Strategy
3.10: Quick Check
3.5: Degree Levels and Transfer Destinations
background
16.5: Appendix E — Technical Memo Template
bar
9.2: Chart Type Selection
bar chart
9.12: Quick Check
base units
5.10: Summary
5.3: SI System — Base and Derived Units
biomedical
1.7: Major Disciplines at a Glance
2.8: Summary
biomedical engineering
1.6: What Engineers Actually Do
2.2: The Major Engineering Disciplines
2.3: Most Real Problems Cross Discipline Boundaries
Bode plots
9.6: Logarithmic Scales
bridge
15.1: This Is Not a New Topic
15.4: Model- Tension, Compression, Straws
bridge challenge
15: Integrated Engineering Design Challenge
15.11: Summary
bridge design
15.2: Challenge Specifications
7.10: Communicate
brightness
11.12: LED Design
buckling
15.11: Summary
15.12: End-of-Chapter Problem Set
15.4: Model- Tension, Compression, Straws
build complexity
15.7: Alternatives and Decision Matrix
building safety
14.11: Citicorp Case Study
business managers
13.8: TEGO Case Study

C

cable car project
8.6: Conditional Formatting and IF Logic
cable rating
8.8: Spreadsheets as Design Exploration Tools
cable tension
10.11: End-of-Chapter Problem Set
10.4: Four-Step Build
10.7: Logical Indexing
6.6: Step 4 — Interpret + Sensitivity
8.4: Parameter Sweeps
9.12: Quick Check
cable tension graph
13.12: End-of-Chapter Applied Exercises
cable tension model
10.2: MATLAB as Structured Engineering
cables
6.2: Full Workflow Worked Example First
CAD
12.1: CAD Is Not Drawing
12.4: The Sketch as a Mathematical Definition
12.6: Modeling Workflow
12.8: CAD in the Engineering Landscape
15.10: Course Map
5.9: From Units to Everything That Follows
6.9: The Workflow Across Disciplines
CAD course
14.7: Ethics HW Scenario 1
calculation
12.4: The Sketch as a Mathematical Definition
7.5: Identify Constraints
8.8: Spreadsheets as Design Exploration Tools
calculations
5.1: Why Units Matter
5.8: Order-of-Magnitude Reasoning
6.5: Step 3 — Analyze
6.6: Step 4 — Interpret + Sensitivity
8.2: Structuring an Engineering Spreadsheet
calculator
8.1: From Calculator to Model
calculus
2.6: Choosing a Discipline
3.1: Engineering Is a Structured Academic Path
3.10: Quick Check
3.2: The Mathematics Backbone
3.3: Physics as Applied Mathematics
3.4: Typical Lower-Division Engineering Path (Community College)
3.7: Academic Risk Factors
3.9: Summary
4.2: Understanding Your Real Workload
capacitors
9.6: Logarithmic Scales
capacity evaluation
15.8: Iterate
career
2.1: The Question Most Students Ask
career decisions
2.4: Engineering by the Numbers
career growth
2.4: Engineering by the Numbers
career preparation
1.4: What Engineers Actually Do
2.6: Choosing a Discipline
careers
2: Engineering Disciplines and Careers
case study
13.8: TEGO Case Study
14.11: Citicorp Case Study
9.5: Misleading Visualizations
categorical
9.2: Chart Type Selection
categorical data
9.11: Summary
challenge
15.1: This Is Not a New Topic
challenge specifications
15.2: Challenge Specifications
Challenger
9.5: Misleading Visualizations
charge
11: Circuit Fundamentals
charge flow
11.2: Physical Meaning
chart
9.2: Chart Type Selection
chart type
16.4: Appendix D — Spreadsheet Quick Reference (Excel)
9.10: Graph Checklist — Quick Reference
9.8: Visualization and Interpretation
chart type selection
9: Data Visualization for Engineers
checkable data
6.7: Step 5 — Communicate
chemical
1.7: Major Disciplines at a Glance
2.8: Summary
chemical engineering
1.6: What Engineers Actually Do
2.1: The Question Most Students Ask
2.2: The Major Engineering Disciplines
2.3: Most Real Problems Cross Discipline Boundaries
chevron bracing
14.11: Citicorp Case Study
choosing
10.9: Choosing the Right Tool
9.2: Chart Type Selection
choosing a discipline
2.6: Choosing a Discipline
circuit
11: Circuit Fundamentals
11.13: Circuit Simulator
11.14: Summary
11.3: Ohm's Law
13.4: Complete Memo Example
circuit analysis
11.11: Mixed Circuits
5.7: Units in Computational Models
6.9: The Workflow Across Disciplines
8.9: When Spreadsheets Are Not Enough
circuit design
11.12: LED Design
11.4: Power
11.5: Schematic Symbols
11.8: Series vs Parallel Comparison
6.11: Summary
7.2: What Makes a Design Problem
circuit power
6.4: Step 2 — Model
circuit schematics
11.5: Schematic Symbols
circuits
11.1: Circuits as Energy Systems
11.7: Parallel Circuits
5.9: From Units to Everything That Follows
7.13: Quick Check
citations
13.7: Technical Presentations

Citicorp Building
14.11: Citicorp Case Study
Citicorp Building case
14: Engineering Ethics
civil
1.7: Major Disciplines at a Glance
2.8: Summary
civil engineering
1.1: The Engineer You Don't Notice
1.11: Quick Check
1.5: The Engineering Mindset
2.1: The Question Most Students Ask
2.2: The Major Engineering Disciplines
2.9: Quick Check
claims
13.5: Quantitative Precision
climate impact
14.12: Environmental Responsibility
closing
13.9: Professional Email
codes of ethics
14.6: Eight-Step Framework
cognitive load
4.3: How to Actually Study Engineering
cognitive performance
4.7: Sustaining Yourself Over the Long Haul
collaboration
4.6: Engineering Teamwork
commands
10.3: Script Basics
comments
10.3: Script Basics
16.3: Appendix C — MATLAB Quick Reference
common errors
8.10: Summary
communicating uncertainty
13.10: Communicating Uncertainty and Limitations
Communication
1.4: What Engineers Actually Do
13: Engineering Communication
13.1: Communication Is Part of the Deliverable
13.11: Summary
13.7: Technical Presentations
13.9: Professional Email
4.6: Engineering Teamwork
6: Structured Engineering Problem Solving
6.1: Why Structure Matters
9: Data Visualization for Engineers
communication ethics
14.1: Ethics Is Not an Add-On
communication in engineering
6.11: Summary
communication skills
13.8: TEGO Case Study
communication tools
13.6: Graphs and Tables
communications
1.1: The Engineer You Don't Notice
comparative behavior
9.1: Why Graphs Are an Engineering Tool
comparison
10.9: Choosing the Right Tool
11.8: Series vs Parallel Comparison
13.6: Graphs and Tables
9.2: Chart Type Selection
9.7: Multi-Curve Comparison
compensation
2.4: Engineering by the Numbers
competence
14.3: Ethics HW Preview Table
14.7: Ethics HW Scenario 1
complete answer
6.7: Step 5 — Communicate
compliance
14.3: Ethics HW Preview Table
14.9: Ethics HW Scenario 3
component value selection
9.6: Logarithmic Scales
compression
15: Integrated Engineering Design Challenge
15.11: Summary
15.4: Model- Tension, Compression, Straws
computation
10.1: Why Engineers Program
computational
16.1: Appendix A — Standard Symbols and Units
computational competence
1.9: What This Course Will Build
computational models
5: Units and Dimensional Analysis
5.7: Units in Computational Models
computed output
8.4: Parameter Sweeps
computer
1.7: Major Disciplines at a Glance
2.8: Summary
computer engineering
2.2: The Major Engineering Disciplines
2.3: Most Real Problems Cross Discipline Boundaries
3.4: Typical Lower-Division Engineering Path (Community College)
computer science
3.5: Degree Levels and Transfer Destinations
computerengineering
3.9: Summary
conceptual error
4.4: Understanding Your Errors
conclusion
13.11: Summary
13.3: The Technical Memo
13.4: Complete Memo Example
15.9: Bridge Report
16.5: Appendix E — Technical Memo Template
9.8: Visualization and Interpretation
conclusion revision
13.12: End-of-Chapter Applied Exercises
conditional formatting
8: Spreadsheet Modeling
8.6: Conditional Formatting and IF Logic
conditional logic
10.6: Conditional Logic
confidence
4.1: It Is Not About Talent
conflict resolution
4.6: Engineering Teamwork
consequences
1.5: The Engineering Mindset
7.11: Design, Ethics, and Safety Factors
conservation laws
6.3: Step 1 — Define
conservation of charge
11.9: Kirchhoff's Laws
conservation of energy
11.9: Kirchhoff's Laws
conservation principles
11.1: Circuits as Energy Systems
constant rate of change
9.4: Linear vs Nonlinear Trends
constants
8.3: Absolute vs Relative References
constrained sketch
12.9: Summary
constraint
15.12: End-of-Chapter Problem Set
constraint crossings
9.1: Why Graphs Are an Engineering Tool
constraint type
12.10: End-of-Chapter Problems
constraints
12.1: CAD Is Not Drawing
12.6: Modeling Workflow
15.2: Challenge Specifications
15.5: Mass Budget
3.8: Treating Your Academic Plan Like an Engineering Problem
4.9: End-of-Chapter Exercise
7: The Engineering Design Process
7.1: Analysis vs Design
7.10: Communicate
7.12: Summary
7.13: Quick Check
7.2: What Makes a Design Problem
7.5: Identify Constraints
7.7: Model and Analyze
7.9: Iterate and Refine
construction
15.1: This Is Not a New Topic
context
13.9: Professional Email
6.1: Why Structure Matters
continuous
9.2: Chart Type Selection
conversion
15.12: End-of-Chapter Problem Set
16.2: Appendix B — Unit Conversion Reference Table
conversion factor
5.4: Unit Conversion Method
conversion factor method
5.9: From Units to Everything That Follows
conversion factors
5.10: Summary
copying homework
14.10: Academic Integrity
cosine
10.5: Trigonometry in MATLAB
course
3.7: Academic Risk Factors
course communication
13.2: Three Modes of Communication
course load
4.2: Understanding Your Real Workload
course map
15.10: Course Map
course mapping
3.6: Transfer Strategy
course modeling
3.8: Treating Your Academic Plan Like an Engineering Problem

course plan
2.9: Quick Check
course planning
3.10: Quick Check
coursework
3.9: Summary
coursework habits
14.10: Academic Integrity
creativity
1.2: A Working Definition
creativity in design
7.6: Generate Alternatives
credibility
14.7: Ethics HW Scenario 1
criteria
7.2: What Makes a Design Problem
7.8: Evaluate Trade-offs
cumulative review
4.3: How to Actually Study Engineering
current
11: Circuit Fundamentals
11.1: Circuits as Energy Systems
11.10: Power in Networks
11.13: Circuit Simulator
11.15: End-of-Chapter Problem Set
11.2: Physical Meaning
11.3: Ohm's Law
11.4: Power
11.6: Series Circuits
11.7: Parallel Circuits
11.8: Series vs Parallel Comparison
5.11: End-of-Chapter Problem Set
6.12: End-of-Chapter Problem Set
6.3: Step 1 — Define
6.4: Step 2 — Model
8.11: End-of-Chapter Problem Set
current calculation
5.7: Units in Computational Models
current competence
14.14: Quick Check
current limiting
11.12: LED Design
Current Limiting Resistor
11.15: End-of-Chapter Problem Set
cut
12.3: From Sketch to Solid
12.9: Summary
cybersecurity engineering
1.6: What Engineers Actually Do

D

data
10.8: Programming Project- Cable Car in MATLAB
13.5: Quantitative Precision
9.3: Anatomy of an Engineering Graph
data analysis
10: MATLAB for Engineering Computation
10.10: Summary
10.7: Logical Indexing
2.4: Engineering by the Numbers
8.11: End-of-Chapter Problem Set
8.6: Conditional Formatting and IF Logic
9.9: Advanced Chart Features
data engineering
1.6: What Engineers Actually Do
data integrity
9.5: Misleading Visualizations
data interpretation
13.6: Graphs and Tables
8.5: Graphical Visualization
9: Data Visualization for Engineers
9.11: Summary
data presentation
13.6: Graphs and Tables
8.7: Six Common Spreadsheet Errors
9.10: Graph Checklist — Quick Reference
data range
9.8: Visualization and Interpretation
data representation
8.5: Graphical Visualization
data visualization
15.10: Course Map
9: Data Visualization for Engineers
9.1: Why Graphs Are an Engineering Tool
9.10: Graph Checklist — Quick Reference
9.11: Summary
9.5: Misleading Visualizations
9.9: Advanced Chart Features
decibels
9.6: Logarithmic Scales
decision documentation
14.6: Eight-Step Framework
decision making
1.2: A Working Definition
1.8: Ethics Is Not a Separate Chapter
14.1: Ethics Is Not an Add-On
14.4: Safety Factors
14.8: Ethics HW Scenario 2
2.7: One Obligation That Every Discipline Shares
9.1: Why Graphs Are an Engineering Tool
decision matrix
15: Integrated Engineering Design Challenge
7: The Engineering Design Process
7.12: Summary
7.8: Evaluate Trade-offs
define
6.10: Structure and Professional Accountability
6.3: Step 1 — Define
define objective
7.3: Seven-Step Design Framework
define step
15.2: Challenge Specifications
definition
1: What is Engineering?
degree
3.9: Summary
degree hierarchy
3.5: Degree Levels and Transfer Destinations
degrees
10.5: Trigonometry in MATLAB
deliberate practice
4: Engineering Success Systems
deliverables
13.1: Communication Is Part of the Deliverable
derivatives
3.2: The Mathematics Backbone
derived units
5: Units and Dimensional Analysis
5.10: Summary
5.3: SI System — Base and Derived Units
design
1.1: The Engineer You Don't Notice
1.10: Summary
1.11: Quick Check
1.2: A Working Definition
1.3: Engineering vs Science vs Technology
1.4: What Engineers Actually Do
10.9: Choosing the Right Tool
12.8: CAD in the Engineering Landscape
13.1: Communication Is Part of the Deliverable
15.1: This Is Not a New Topic
15.5: Mass Budget
2: Engineering Disciplines and Careers
2.5: Functional Roles
2.7: One Obligation That Every Discipline Shares
7.1: Analysis vs Design
7.11: Design, Ethics, and Safety Factors
7.12: Summary
7.13: Quick Check
7.5: Identify Constraints
7.7: Model and Analyze
7.8: Evaluate Trade-offs
7.9: Iterate and Refine
8.1: From Calculator to Model
9.7: Multi-Curve Comparison
design alternatives
15: Integrated Engineering Design Challenge
7: The Engineering Design Process
7.6: Generate Alternatives
8.9: When Spreadsheets Are Not Enough
design applications
10.11: End-of-Chapter Problem Set
design approach
15.9: Bridge Report
design approval
14.14: Quick Check
design communication
7.10: Communicate
design components
7.4: Define the Design Objective
design considerations
8.4: Parameter Sweeps
design constraints
7.4: Define the Design Objective
design criteria
7.4: Define the Design Objective
design decision
9.8: Visualization and Interpretation
design decisions
7.10: Communicate
8.11: End-of-Chapter Problem Set
8.8: Spreadsheets as Design Exploration Tools
design exploration
8.10: Summary
8.8: Spreadsheets as Design Exploration Tools
design filtering
10.7: Logical Indexing
design flexibility
8.5: Graphical Visualization
design framework
15.6: Parameter Sweep
7: The Engineering Design Process
7.3: Seven-Step Design Framework
design intent
12: Parametric CAD and 3D Modeling
12.10: End-of-Chapter Problems
12.2: Parametric Modeling Core Concepts
12.9: Summary
design iteration
15.8: Iterate

design limits
8.8: Spreadsheets as Design Exploration Tools
design metrics
15.3: Define- Units, Loads, What "Best" Means
design objective
7: The Engineering Design Process
7.4: Define the Design Objective
design optimization
15.8: Iterate
design problem
13.8: TEGO Case Study
15.11: Summary
7.2: What Makes a Design Problem
design problems
7: The Engineering Design Process
design process
7.4: Define the Design Objective
design reasoning
8.8: Spreadsheets as Design Exploration Tools
design requirements
12.2: Parametric Modeling Core Concepts
design rules
15.2: Challenge Specifications
design safety
6.6: Step 4 — Interpret + Sensitivity
design standpoint
6.7: Step 5 — Communicate
design strategy
15.12: End-of-Chapter Problem Set
design thinking
7.3: Seven-Step Design Framework
design updates
15.8: Iterate
diagnosing errors
4.4: Understanding Your Errors
differential equations
3.2: The Mathematics Backbone
differential_equations
3.1: Engineering Is a Structured Academic Path
differentialequations
3.9: Summary
dimension constraints
12.7: Five Common Modeling Errors
dimensional analysis
5: Units and Dimensional Analysis
5.10: Summary
5.2: Quantities vs. Numbers
5.4: Unit Conversion Method
5.6: Dimensional Analysis
5.9: From Units to Everything That Follows
dimensional consistency
1.5: The Engineering Mindset
5.6: Dimensional Analysis
6.11: Summary
6.8: Five Common Failure Patterns
dimensional validity
5.6: Dimensional Analysis
dimensional verification
5.11: End-of-Chapter Problem Set
dimensions
12.1: CAD Is Not Drawing
12.2: Parametric Modeling Core Concepts
12.4: The Sketch as a Mathematical Definition
12.5: The Fusion Project
12.6: Modeling Workflow
16.1: Appendix A — Standard Symbols and Units
disciplines
2: Engineering Disciplines and Careers
2.1: The Question Most Students Ask
6.9: The Workflow Across Disciplines
disclosure
14.1: Ethics Is Not an Add-On
division
5.2: Quantities vs. Numbers
doctoral degree
3: Academic Pathways and Transfer Strategy
doctorate
3.5: Degree Levels and Transfer Destinations
documentation
1.10: Summary
1.4: What Engineers Actually Do
13.1: Communication Is Part of the Deliverable
7.11: Design, Ethics, and Safety Factors
drafting standards
12.5: The Fusion Project
drawing
12.1: CAD Is Not Drawing
drawing orientation
12.7: Five Common Modeling Errors
drawings
12.8: CAD in the Engineering Landscape
13.1: Communication Is Part of the Deliverable
driving force
11.2: Physical Meaning
dynamic loads
13.10: Communicating Uncertainty and Limitations
dynamics
3.10: Quick Check

E

editing
12.3: From Sketch to Solid
education
1.3: Engineering vs Science vs Technology
12.8: CAD in the Engineering Landscape
2.7: One Obligation That Every Discipline Shares
3.1: Engineering Is a Structured Academic Path
3.10: Quick Check
6.1: Why Structure Matters
educational objectives
10: MATLAB for Engineering Computation
efficiency
15.12: End-of-Chapter Problem Set
15.3: Define- Units, Loads, What "Best" Means
7.9: Iterate and Refine
Electric Current
5.3: SI System — Base and Derived Units
electric potential
11.2: Physical Meaning
electrical
1.7: Major Disciplines at a Glance
16.1: Appendix A — Standard Symbols and Units
16.2: Appendix B — Unit Conversion Reference Table
2.8: Summary
electrical circuit
11.1: Circuits as Energy Systems
6.3: Step 1 — Define
electrical circuits
11.10: Power in Networks
11.11: Mixed Circuits
11.2: Physical Meaning
11.9: Kirchhoff's Laws
electrical components
11.4: Power
electrical current
5.8: Order-of-Magnitude Reasoning
electrical energy
11.2: Physical Meaning
electrical engineering
1.11: Quick Check
1.5: The Engineering Mindset
1.6: What Engineers Actually Do
10.11: End-of-Chapter Problem Set
11.1: Circuits as Energy Systems
11.15: End-of-Chapter Problem Set
11.6: Series Circuits
11.7: Parallel Circuits
11.8: Series vs Parallel Comparison
2.1: The Question Most Students Ask
2.2: The Major Engineering Disciplines
2.3: Most Real Problems Cross Discipline Boundaries
3.4: Typical Lower-Division Engineering Path (Community College)
3.5: Degree Levels and Transfer Destinations
5.7: Units in Computational Models
6.4: Step 2 — Model
9.6: Logarithmic Scales
electrical power
11.4: Power
electrical resistance
5.11: End-of-Chapter Problem Set
electricalengineering
3.9: Summary
electricity
3.3: Physics as Applied Mathematics
electronics
11.12: LED Design
11.3: Ohm's Law
email conventions
13.9: Professional Email
email etiquette
13.9: Professional Email
email structure
13.9: Professional Email
employment
2: Engineering Disciplines and Careers
2.4: Engineering by the Numbers
energy
11: Circuit Fundamentals
11.14: Summary
3.3: Physics as Applied Mathematics
energy per unit charge
11.2: Physical Meaning
energy systems
11.1: Circuits as Energy Systems
energy transfer
11.1: Circuits as Energy Systems
engineered systems
1: What is Engineering?

engineering
- 1: What is Engineering?
- 1.1: The Engineer You Don't Notice
- 1.10: Summary
- 1.11: Quick Check
- 1.2: A Working Definition
- 1.3: Engineering vs Science vs Technology
- 1.4: What Engineers Actually Do
- 1.7: Major Disciplines at a Glance
- 1.8: Ethics Is Not a Separate Chapter
- 10: MATLAB for Engineering Computation
- 10.1: Why Engineers Program
- 10.10: Summary
- 10.3: Script Basics
- 10.4: Four-Step Build
- 10.8: Programming Project- Cable Car in MATLAB
- 12.6: Modeling Workflow
- 12.8: CAD in the Engineering Landscape
- 13: Engineering Communication
- 13.1: Communication Is Part of the Deliverable
- 13.11: Summary
- 13.3: The Technical Memo
- 13.4: Complete Memo Example
- 13.5: Quantitative Precision
- 13.7: Technical Presentations
- 14.1: Ethics Is Not an Add-On
- 14.6: Eight-Step Framework
- 14.8: Ethics HW Scenario 2
- 14.9: Ethics HW Scenario 3
- 15.1: This Is Not a New Topic
- 15.5: Mass Budget
- 16.5: Appendix E — Technical Memo Template
- 2: Engineering Disciplines and Careers
- 2.1: The Question Most Students Ask
- 2.2: The Major Engineering Disciplines
- 2.3: Most Real Problems Cross Discipline Boundaries
- 2.4: Engineering by the Numbers
- 2.5: Functional Roles
- 2.7: One Obligation That Every Discipline Shares
- 2.8: Summary
- 2.9: Quick Check
- 3.1: Engineering Is a Structured Academic Path
- 3.10: Quick Check
- 3.2: The Mathematics Backbone
- 3.7: Academic Risk Factors
- 3.9: Summary
- 4: Engineering Success Systems
- 4.1: It Is Not About Talent
- 4.2: Understanding Your Real Workload
- 4.5: Goals, Humility, and the Long Game
- 4.6: Engineering Teamwork
- 4.8: Summary
- 5.2: Quantities vs. Numbers
- 5.4: Unit Conversion Method
- 5.8: Order-of-Magnitude Reasoning
- 6.1: Why Structure Matters
- 6.2: Full Workflow Worked Example First
- 6.3: Step 1 — Define
- 6.6: Step 4 — Interpret + Sensitivity
- 6.9: The Workflow Across Disciplines
- 7.1: Analysis vs Design
- 7.11: Design, Ethics, and Safety Factors
- 7.13: Quick Check
- 7.5: Identify Constraints
- 7.7: Model and Analyze
- 7.8: Evaluate Trade-offs
- 7.9: Iterate and Refine
- 8.1: From Calculator to Model
- 8.11: End-of-Chapter Problem Set
- 8.2: Structuring an Engineering Spreadsheet
- 8.5: Graphical Visualization
- 9: Data Visualization for Engineers
- 9.1: Why Graphs Are an Engineering Tool
- 9.3: Anatomy of an Engineering Graph
- 9.8: Visualization and Interpretation

engineering analysis
- 10.6: Conditional Logic
- 6.11: Summary
- 8: Spreadsheet Modeling

engineering calculations
- 6.10: Structure and Professional Accountability

engineering codes
- 2.7: One Obligation That Every Discipline Shares

engineering communication
- 13.10: Communicating Uncertainty and Limitations
- 13.12: End-of-Chapter Applied Exercises
- 13.2: Three Modes of Communication
- 13.3: The Technical Memo
- 15.10: Course Map
- 6.7: Step 5 — Communicate

engineering coursework
- 3: Academic Pathways and Transfer Strategy
- 3.4: Typical Lower-Division Engineering Path (Community College)

engineering data
- 9: Data Visualization for Engineers

engineering decisions
- 1.5: The Engineering Mindset
- 14.12: Environmental Responsibility

engineering design
- 10.7: Logical Indexing
- 13.7: Technical Presentations
- 15: Integrated Engineering Design Challenge
- 15.10: Course Map
- 15.2: Challenge Specifications
- 15.6: Parameter Sweep
- 15.7: Alternatives and Decision Matrix
- 7.3: Seven-Step Design Framework
- 7.4: Define the Design Objective
- 7.6: Generate Alternatives
- 8: Spreadsheet Modeling
- 8.8: Spreadsheets as Design Exploration Tools

engineering design process
- 15.8: Iterate
- 7: The Engineering Design Process

engineering drawing
- 12.5: The Fusion Project

engineering education
- 1.9: What This Course Will Build
- 3.8: Treating Your Academic Plan Like an Engineering Problem
- 4.6: Engineering Teamwork
- 4.7: Sustaining Yourself Over the Long Haul

engineering errors
- 4.4: Understanding Your Errors
- 5.1: Why Units Matter
- 6.8: Five Common Failure Patterns

engineering ethics
- 1.8: Ethics Is Not a Separate Chapter
- 14: Engineering Ethics
- 14.10: Academic Integrity
- 14.11: Citicorp Case Study
- 14.12: Environmental Responsibility
- 14.13: Summary
- 14.14: Quick Check
- 14.2: NCEES Model Rules
- 14.4: Safety Factors
- 14.5: Modeling Assumptions
- 14.7: Ethics HW Scenario 1
- 15.10: Course Map
- 15.11: Summary
- 8.7: Six Common Spreadsheet Errors

engineering history
- 5.1: Why Units Matter

engineering logic
- 12.2: Parametric Modeling Core Concepts
- 12.9: Summary

engineering majors
- 2.6: Choosing a Discipline

engineering mindset
- 1: What is Engineering?
- 1.11: Quick Check
- 1.5: The Engineering Mindset
- 1.9: What This Course Will Build

engineering models
- 8.6: Conditional Formatting and IF Logic

engineering practices
- 14.10: Academic Integrity
- 5.9: From Units to Everything That Follows

engineering presentation
- 13.8: TEGO Case Study

engineering principles
- 15.3: Define- Units, Loads, What "Best" Means
- 6: Structured Engineering Problem Solving
- 7.6: Generate Alternatives

engineering problem
- 3.8: Treating Your Academic Plan Like an Engineering Problem

engineering reporting
- 9.9: Advanced Chart Features

engineering responsibilities
- 14.4: Safety Factors

engineering scale
- 5: Units and Dimensional Analysis

engineering standards
- 14.2: NCEES Model Rules

engineering thinking
- 4.7: Sustaining Yourself Over the Long Haul

engineering tools
- 8: Spreadsheet Modeling

engineering tracks
- 3.4: Typical Lower-Division Engineering Path (Community College)

engineering transfer
- 3.6: Transfer Strategy

engineering workflow
- 10: MATLAB for Engineering Computation
- 10.2: MATLAB as Structured Engineering
- 6: Structured Engineering Problem Solving
- 6.11: Summary

engineering_analysis
- 3.1: Engineering Is a Structured Academic Path

engineers
- 1.4: What Engineers Actually Do
- 9.5: Misleading Visualizations

ENGR 110
- 1.9: What This Course Will Build

environmental engineering
- 1.1: The Engineer You Don't Notice
- 1.6: What Engineers Actually Do
- 2.3: Most Real Problems Cross Discipline Boundaries

environmental responsibility
- 14: Engineering Ethics
- 14.12: Environmental Responsibility

equation matching
- 9.4: Linear vs Nonlinear Trends

equations
5.6: Dimensional Analysis
6.3: Step 1 — Define
6.4: Step 2 — Model
8.1: From Calculator to Model
8.2: Structuring an Engineering Spreadsheet
equivalent resistance
11.11: Mixed Circuits
11.6: Series Circuits
11.7: Parallel Circuits
11.8: Series vs Parallel Comparison
error analysis
4.3: How to Actually Study Engineering
error bars
9: Data Visualization for Engineers
9.10: Graph Checklist — Quick Reference
9.11: Summary
9.9: Advanced Chart Features
error detection
5.10: Summary
error patterns
4.9: End-of-Chapter Exercise
error types
4.4: Understanding Your Errors
errors
5.7: Units in Computational Models
6.5: Step 3 — Analyze
estimated force
15.6: Parameter Sweep
ethical calculus
14.14: Quick Check
ethical codes
14.2: NCEES Model Rules
ethical considerations
1.11: Quick Check
ethical decision making
14.6: Eight-Step Framework
ethical design
14.4: Safety Factors
ethical obligation
14: Engineering Ethics
14.14: Quick Check
ethical obligations
14.5: Modeling Assumptions
ethical principles
14: Engineering Ethics
ethical problems
14.14: Quick Check
ethical responsibility
14: Engineering Ethics
ethics
1: What is Engineering?
1.8: Ethics Is Not a Separate Chapter
14.1: Ethics Is Not an Add-On
14.3: Ethics HW Preview Table
14.7: Ethics HW Scenario 1
14.8: Ethics HW Scenario 2
14.9: Ethics HW Scenario 3
2.7: One Obligation That Every Discipline Shares
7.11: Design, Ethics, and Safety Factors
evaluation
10.1: Why Engineers Program
13.1: Communication Is Part of the Deliverable
13.4: Complete Memo Example
7.12: Summary
7.7: Model and Analyze
7.8: Evaluate Trade-offs
9.7: Multi-Curve Comparison
evaluation metric
7.4: Define the Design Objective
evaluation metrics
7: The Engineering Design Process
exam
3.7: Academic Risk Factors
excel
16: Appendices
16.4: Appendix D — Spreadsheet Quick Reference (Excel)
8.11: End-of-Chapter Problem Set
8.6: Conditional Formatting and IF Logic
exercise
4.9: End-of-Chapter Exercise
experiment
15.9: Bridge Report
extrapolation
9.4: Linear vs Nonlinear Trends
extrude
12.3: From Sketch to Solid
12.9: Summary

F

failure patterns
6.8: Five Common Failure Patterns
falsifying data
14.10: Academic Integrity
Falstad
11.13: Circuit Simulator
FEA
12.8: CAD in the Engineering Landscape
feature tree
12.3: From Sketch to Solid
12.9: Summary
feedback loop
4: Engineering Success Systems
figures
13.5: Quantitative Precision
finite element analysis
13.10: Communicating Uncertainty and Limitations
fixed resistance
11.3: Ohm's Law
flat truss
15.7: Alternatives and Decision Matrix
focused sessions
4.3: How to Actually Study Engineering
force
15.4: Model- Tension, Compression, Straws
16.2: Appendix B — Unit Conversion Reference Table
5.11: End-of-Chapter Problem Set
forces
3.3: Physics as Applied Mathematics
Forecasting
9.9: Advanced Chart Features
formula
10.8: Programming Project- Cable Car in MATLAB
8.3: Absolute vs Relative References
formula hunting
6.8: Five Common Failure Patterns
formulas
8.11: End-of-Chapter Problem Set
8.2: Structuring an Engineering Spreadsheet
forward current
11.12: LED Design
forward voltage
11.12: LED Design
foundational coursework
2.6: Choosing a Discipline
framework
15.1: This Is Not a New Topic
frequency response
9.6: Logarithmic Scales
frequency response plots
9.12: Quick Check
functional roles
2: Engineering Disciplines and Careers
functions
10.5: Trigonometry in MATLAB
Fundamentals
11: Circuit Fundamentals
fusion
12.6: Modeling Workflow
12.7: Five Common Modeling Errors
Fusion 360
12.2: Parametric Modeling Core Concepts
12.5: The Fusion Project
fusion project
12: Parametric CAD and 3D Modeling
12.5: The Fusion Project

G

generate alternatives
7.3: Seven-Step Design Framework
geometric constraints
12: Parametric CAD and 3D Modeling
12.2: Parametric Modeling Core Concepts
geometry
12.1: CAD Is Not Drawing
12.3: From Sketch to Solid
12.4: The Sketch as a Mathematical Definition
12.6: Modeling Workflow
12.7: Five Common Modeling Errors
15.4: Model- Tension, Compression, Straws
15.7: Alternatives and Decision Matrix
giga
5.5: SI Prefixes
goal setting
3.8: Treating Your Academic Plan Like an Engineering Problem
4.7: Sustaining Yourself Over the Long Haul
goals
4.5: Goals, Humility, and the Long Game
4.9: End-of-Chapter Exercise
GPA
3.7: Academic Risk Factors
GPA management
3.8: Treating Your Academic Plan Like an Engineering Problem
GPA requirements
3.6: Transfer Strategy
graph
9.3: Anatomy of an Engineering Graph
9.8: Visualization and Interpretation
graph checklist
9.10: Graph Checklist — Quick Reference
graph construction
9: Data Visualization for Engineers
graph shape
9.4: Linear vs Nonlinear Trends

graphical visualization
8: Spreadsheet Modeling
8.5: Graphical Visualization
graphs
13: Engineering Communication
13.6: Graphs and Tables
8.2: Structuring an Engineering Spreadsheet
9: Data Visualization for Engineers
9.1: Why Graphs Are an Engineering Tool
9.11: Summary
9.7: Multi-Curve Comparison
ground
11.5: Schematic Symbols
group work
14.10: Academic Integrity

H

habits
3.7: Academic Risk Factors
6.5: Step 3 — Analyze
hard constraint
7.5: Identify Constraints
hard constraints
15: Integrated Engineering Design Challenge
15.11: Summary
15.3: Define- Units, Loads, What "Best" Means
hardcoded constants
8.7: Six Common Spreadsheet Errors
health
14.2: NCEES Model Rules
high sensitivity
9.4: Linear vs Nonlinear Trends
honesty
14.3: Ethics HW Preview Table
14.7: Ethics HW Scenario 1
household circuit
11.7: Parallel Circuits
household circuits
11.8: Series vs Parallel Comparison
humility
4: Engineering Success Systems
4.5: Goals, Humility, and the Long Game

I

ideal resistors
14.5: Modeling Assumptions
identify constraints
7.3: Seven-Step Design Framework
IF function
8.6: Conditional Formatting and IF Logic
IF logic
8: Spreadsheet Modeling
IFS function
8.6: Conditional Formatting and IF Logic
industrial
1.7: Major Disciplines at a Glance
2.8: Summary
industrial engineering
2.2: The Major Engineering Disciplines
2.3: Most Real Problems Cross Discipline Boundaries
industry
2: Engineering Disciplines and Careers
infrastructure
2.1: The Question Most Students Ask
infrastructure equity
14.12: Environmental Responsibility
innovation
1.1: The Engineer You Don't Notice
input
10.3: Script Basics
input variation
6.6: Step 4 — Interpret + Sensitivity
8.4: Parameter Sweeps
inputs
16.3: Appendix C — MATLAB Quick Reference
16.4: Appendix D — Spreadsheet Quick Reference (Excel)
inputs section
10.2: MATLAB as Structured Engineering
integrals
3.2: The Mathematics Backbone
integration
15.1: This Is Not a New Topic
integrity
1.8: Ethics Is Not a Separate Chapter
14.8: Ethics HW Scenario 2
integrity standards
14.2: NCEES Model Rules
interdisciplinary
2.9: Quick Check
intermediate results
6.5: Step 3 — Analyze
internships
2.6: Choosing a Discipline
Interpret
6.10: Structure and Professional Accountability
6.9: The Workflow Across Disciplines
interpretation
10.2: MATLAB as Structured Engineering
6: Structured Engineering Problem Solving
6.6: Step 4 — Interpret + Sensitivity
9.8: Visualization and Interpretation
interpretation error
4.4: Understanding Your Errors
interviews
2.6: Choosing a Discipline
invalid equations
5.6: Dimensional Analysis
isometric view
12.5: The Fusion Project
iteration
15: Integrated Engineering Design Challenge
7: The Engineering Design Process
7.9: Iterate and Refine
iterative algorithms
8.9: When Spreadsheets Are Not Enough
iterative design
1.5: The Engineering Mindset
1.9: What This Course Will Build

J

job market
2.4: Engineering by the Numbers

K

KCL
11.11: Mixed Circuits
11.13: Circuit Simulator
11.14: Summary
11.15: End-of-Chapter Problem Set
11.7: Parallel Circuits
11.9: Kirchhoff's Laws
key functions
16.4: Appendix D — Spreadsheet Quick Reference (Excel)
kilo
5.5: SI Prefixes
Kirchhoff's Current Law
11.9: Kirchhoff's Laws
Kirchhoff's Laws
11.9: Kirchhoff's Laws
Kirchhoff's Voltage Law
11.6: Series Circuits
11.9: Kirchhoff's Laws
KVL
11.13: Circuit Simulator
11.14: Summary
11.15: End-of-Chapter Problem Set
11.9: Kirchhoff's Laws

L

labeled axes
8.7: Six Common Spreadsheet Errors
labels
9.10: Graph Checklist — Quick Reference
large datasets
10.10: Summary
launch
9.5: Misleading Visualizations
Learning
3.10: Quick Check
learning from mistakes
4.4: Understanding Your Errors
learning objectives
4: Engineering Success Systems
learning strategies
4.1: It Is Not About Talent
LED
11.12: LED Design
11.5: Schematic Symbols
7.13: Quick Check
legal compliance
14: Engineering Ethics
length
16.2: Appendix B — Unit Conversion Reference Table
5.3: SI System — Base and Derived Units
lifelong learning
4.5: Goals, Humility, and the Long Game
limitation statements
13.12: End-of-Chapter Applied Exercises
limitations
13.10: Communicating Uncertainty and Limitations
linear algebra
3.2: The Mathematics Backbone
linear axes
9.6: Logarithmic Scales
linear elastic behavior
14.14: Quick Check
linear equations
5.6: Dimensional Analysis
linear trends
9.4: Linear vs Nonlinear Trends
load
11.14: Summary
15.12: End-of-Chapter Problem Set
15.4: Model- Tension, Compression, Straws

load analysis
15.6: Parameter Sweep
7.2: What Makes a Design Problem
load capacity
15.2: Challenge Specifications
15.5: Mass Budget
15.7: Alternatives and Decision Matrix
7.10: Communicate
load estimation
14.4: Safety Factors
load requirement
15: Integrated Engineering Design Challenge
load to mass ratio
15.3: Define- Units, Loads, What "Best" Means
loads
15.3: Define- Units, Loads, What "Best" Means
logarithmic scales
9.6: Logarithmic Scales
logic
10.1: Why Engineers Program
logical indexing
10.11: End-of-Chapter Problem Set
10.7: Logical Indexing
long game
4.5: Goals, Humility, and the Long Game
low sensitivity
9.4: Linear vs Nonlinear Trends
Luminous Intensity
5.3: SI System — Base and Derived Units

M

machine learning engineering
1.6: What Engineers Actually Do
magnetism
3.3: Physics as Applied Mathematics
magnitude
5.10: Summary
5.2: Quantities vs. Numbers
major disciplines
1: What is Engineering?
manipulation
10.8: Programming Project- Cable Car in MATLAB
manufacturing
12.8: CAD in the Engineering Landscape
2.5: Functional Roles
7.13: Quick Check
7.6: Generate Alternatives
manufacturing practices
14.9: Ethics HW Scenario 3
margin of safety
9.4: Linear vs Nonlinear Trends
Mars Climate Orbiter
5.1: Why Units Matter
mass
12.4: The Sketch as a Mathematical Definition
12.6: Modeling Workflow
16.2: Appendix B — Unit Conversion Reference Table
5.11: End-of-Chapter Problem Set
5.3: SI System — Base and Derived Units
mass budget
15.11: Summary
15.5: Mass Budget
mass calculation
12: Parametric CAD and 3D Modeling
12.10: End-of-Chapter Problems
12.9: Summary
mass efficiency
15.7: Alternatives and Decision Matrix
mass limit
15.2: Challenge Specifications
mass minimization
7.6: Generate Alternatives
mass properties
12.5: The Fusion Project
mass verification
12.7: Five Common Modeling Errors
master's degree
3: Academic Pathways and Transfer Strategy
3.5: Degree Levels and Transfer Destinations
material
12.6: Modeling Workflow
material restrictions
15.2: Challenge Specifications
material variability
14.4: Safety Factors
materials
1.7: Major Disciplines at a Glance
15.5: Mass Budget
7.9: Iterate and Refine
materials and methods
15.9: Bridge Report
materials engineering
2.3: Most Real Problems Cross Discipline Boundaries
materials science
3.4: Typical Lower-Division Engineering Path (Community College)
math sequence
3: Academic Pathways and Transfer Strategy
mathematical model
12.1: CAD Is Not Drawing
mathematics
1.10: Summary
1.2: A Working Definition
10.10: Summary
3.2: The Mathematics Backbone
3.3: Physics as Applied Mathematics
3.9: Summary
6.4: Step 2 — Model
MATLAB
10: MATLAB for Engineering Computation
10.1: Why Engineers Program
10.10: Summary
10.11: End-of-Chapter Problem Set
10.2: MATLAB as Structured Engineering
10.3: Script Basics
10.4: Four-Step Build
10.5: Trigonometry in MATLAB
10.6: Conditional Logic
10.8: Programming Project- Cable Car in MATLAB
10.9: Choosing the Right Tool
15.10: Course Map
16: Appendices
16.3: Appendix C — MATLAB Quick Reference
5.9: From Units to Everything That Follows
maximum voltage
10.7: Logical Indexing
measurement
5.1: Why Units Matter
5.5: SI Prefixes
measurement uncertainty
9.11: Summary
9.9: Advanced Chart Features
measurements
5.4: Unit Conversion Method
5.7: Units in Computational Models
mechanical
1.7: Major Disciplines at a Glance
16.1: Appendix A — Standard Symbols and Units
2.8: Summary
2.9: Quick Check
mechanical engineering
1.1: The Engineer You Don't Notice
1.3: Engineering vs Science vs Technology
15.6: Parameter Sweep
2.2: The Major Engineering Disciplines
2.3: Most Real Problems Cross Discipline Boundaries
2.5: Functional Roles
3.4: Typical Lower-Division Engineering Path (Community College)
3.5: Degree Levels and Transfer Destinations
mechanicalengineering
3.9: Summary
mechanics
3.3: Physics as Applied Mathematics
median wage
2.4: Engineering by the Numbers
mega
5.5: SI Prefixes
memo
13: Engineering Communication
memo drafting
13.12: End-of-Chapter Applied Exercises
memorandum
16.5: Appendix E — Technical Memo Template
memorandums
13.7: Technical Presentations
mental arithmetic
6.5: Step 3 — Analyze
meta
6.2: Full Workflow Worked Example First
methodology
13.11: Summary
13.3: The Technical Memo
13.4: Complete Memo Example
15.1: This Is Not a New Topic
15.9: Bridge Report
16.5: Appendix E — Technical Memo Template
metric system
5.4: Unit Conversion Method
micro
5.5: SI Prefixes
Microsoft 365
16.4: Appendix D — Spreadsheet Quick Reference (Excel)
milli
5.5: SI Prefixes
mindset
4.1: It Is Not About Talent
minimum angle
10.7: Logical Indexing
misleading graphs
8.7: Six Common Spreadsheet Errors
9.11: Summary
misleading visualizations
9.5: Misleading Visualizations

misrepresentation
14.3: Ethics HW Preview Table
mixed circuits
11.11: Mixed Circuits
mixing units
12.7: Five Common Modeling Errors
model
10.9: Choosing the Right Tool
12.4: The Sketch as a Mathematical Definition
6.10: Structure and Professional Accountability
8.2: Structuring an Engineering Spreadsheet
model adaptation
12.2: Parametric Modeling Core Concepts
model analysis
7.3: Seven-Step Design Framework
model equations
16.4: Appendix D — Spreadsheet Quick Reference (Excel)
8.10: Summary
model recomputes
12.1: CAD Is Not Drawing
Model Rules
14.2: NCEES Model Rules
model section
10.2: MATLAB as Structured Engineering
model step
6.8: Five Common Failure Patterns
modeling
1.10: Summary
1.4: What Engineers Actually Do
12.3: From Sketch to Solid
12.6: Modeling Workflow
12.8: CAD in the Engineering Landscape
14.5: Modeling Assumptions
6: Structured Engineering Problem Solving
6.1: Why Structure Matters
6.2: Full Workflow Worked Example First
6.4: Step 2 — Model
7.7: Model and Analyze
8.1: From Calculator to Model
modeling errors
12.7: Five Common Modeling Errors
8.7: Six Common Spreadsheet Errors
modeling workflow
12: Parametric CAD and 3D Modeling
models
10.1: Why Engineers Program
momentum
3.3: Physics as Applied Mathematics
motion
3.3: Physics as Applied Mathematics
motivation
4.8: Summary
multidisciplinary
2.3: Most Real Problems Cross Discipline Boundaries
multiplication
5.2: Quantities vs. Numbers

N

nano
5.5: SI Prefixes
NASA
1.3: Engineering vs Science vs Technology
5.1: Why Units Matter
9.5: Misleading Visualizations
NCEES
14.2: NCEES Model Rules
14.3: Ethics HW Preview Table
14.7: Ethics HW Scenario 1
NCEES Model Rules
14: Engineering Ethics
14.14: Quick Check
Newtons
15.12: End-of-Chapter Problem Set
nominal geometry
14.5: Modeling Assumptions
nonlinear sensitivity
8.4: Parameter Sweeps
nonlinear trends
9.4: Linear vs Nonlinear Trends
numbers
5.2: Quantities vs. Numbers
numeric
9.2: Chart Type Selection
numerical methods
3.2: The Mathematics Backbone
numerical result
6.8: Five Common Failure Patterns
numerical values
13.6: Graphs and Tables

O

objective
15.12: End-of-Chapter Problem Set
Objectives
7.12: Summary
Ohm's law
11: Circuit Fundamentals
11.13: Circuit Simulator
11.14: Summary
11.15: End-of-Chapter Problem Set
11.3: Ohm's Law
5.7: Units in Computational Models
operational safety
14.9: Ethics HW Scenario 3
operations
12.3: From Sketch to Solid
optimization
15.3: Define- Units, Loads, What "Best" Means
15.5: Mass Budget
7.4: Define the Design Objective
7.6: Generate Alternatives
7.7: Model and Analyze
7.8: Evaluate Trade-offs
7.9: Iterate and Refine
8.9: When Spreadsheets Are Not Enough
optimization objectives
15.11: Summary
organization
4.1: It Is Not About Talent
orthographic views
12.10: End-of-Chapter Problems
output
10.3: Script Basics
output change
6.6: Step 4 — Interpret + Sensitivity
output section
10.2: MATLAB as Structured Engineering
outputs
16.4: Appendix D — Spreadsheet Quick Reference (Excel)
8.10: Summary
8.2: Structuring an Engineering Spreadsheet
overheating
11.4: Power

P

parallel
11.14: Summary
11.8: Series vs Parallel Comparison
Parallel Circuits
11.10: Power in Networks
11.15: End-of-Chapter Problem Set
11.7: Parallel Circuits
11.9: Kirchhoff's Laws
parallel resistors
11.11: Mixed Circuits
parameter sweep
10.4: Four-Step Build
15.6: Parameter Sweep
8.4: Parameter Sweeps
9.12: Quick Check
parameter sweeps
8: Spreadsheet Modeling
8.3: Absolute vs Relative References
8.9: When Spreadsheets Are Not Enough
9.1: Why Graphs Are an Engineering Tool
parameters
12.2: Parametric Modeling Core Concepts
8.1: From Calculator to Model
8.2: Structuring an Engineering Spreadsheet
parametric approach
8.8: Spreadsheets as Design Exploration Tools
parametric CAD
12: Parametric CAD and 3D Modeling
12.1: CAD Is Not Drawing
12.10: End-of-Chapter Problems
12.9: Summary
parametric modeling
12.2: Parametric Modeling Core Concepts
8.10: Summary
part description
12.5: The Fusion Project
peer input
14.6: Eight-Step Framework
performance
7.12: Summary
performance competition
15.2: Challenge Specifications
performance metrics
15.9: Bridge Report
performance target
7.4: Define the Design Objective
perseverance
4.5: Goals, Humility, and the Long Game
personal growth
4.5: Goals, Humility, and the Long Game
petroleum
2.9: Quick Check
physical quantities
5.1: Why Units Matter
physical quantity
5: Units and Dimensional Analysis
5.10: Summary

physics
10.4: Four-Step Build
11.4: Power
2.6: Choosing a Discipline
3: Academic Pathways and Transfer Strategy
3.1: Engineering Is a Structured Academic Path
3.3: Physics as Applied Mathematics
3.4: Typical Lower-Division Engineering Path (Community College)
4.2: Understanding Your Real Workload
5.11: End-of-Chapter Problem Set
5.4: Unit Conversion Method
5.6: Dimensional Analysis
6.4: Step 2 — Model
6.6: Step 4 — Interpret + Sensitivity
physics principles
5.6: Dimensional Analysis
physics textbook
11.2: Physical Meaning
pico
5.5: SI Prefixes
pie
9.2: Chart Type Selection
piecewise logic
10.10: Summary
planning
4.9: End-of-Chapter Exercise
plot
10.4: Four-Step Build
10.8: Programming Project- Cable Car in MATLAB
plot analysis
9.6: Logarithmic Scales
plotting
10.10: Summary
10.11: End-of-Chapter Problem Set
16.3: Appendix C — MATLAB Quick Reference
9.7: Multi-Curve Comparison
polygon types
12.7: Five Common Modeling Errors
power
10.6: Conditional Logic
11: Circuit Fundamentals
11.1: Circuits as Energy Systems
11.10: Power in Networks
11.13: Circuit Simulator
11.14: Summary
11.15: End-of-Chapter Problem Set
11.3: Ohm's Law
13.4: Complete Memo Example
5.11: End-of-Chapter Problem Set
6.12: End-of-Chapter Problem Set
6.4: Step 2 — Model
7.13: Quick Check
8.11: End-of-Chapter Problem Set
9.7: Multi-Curve Comparison
power calculation
10.11: End-of-Chapter Problem Set
11.4: Power
11.6: Series Circuits
power dissipation
11.10: Power in Networks
power rating
11.12: LED Design
power safety check
10.6: Conditional Logic
power values
9.12: Quick Check
power vs voltage
8.5: Graphical Visualization
9.12: Quick Check
practice
13.7: Technical Presentations
4.1: It Is Not About Talent
6.5: Step 3 — Analyze
practice problems
11.3: Ohm's Law
precision
13.5: Quantitative Precision
13.6: Graphs and Tables
precision accuracy
6.8: Five Common Failure Patterns
precision rewriting
13.12: End-of-Chapter Applied Exercises
preparation
4.1: It Is Not About Talent
prerequisite courses
3.6: Transfer Strategy
prerequisites
3.1: Engineering Is a Structured Academic Path
3.10: Quick Check
presentations
13: Engineering Communication
pressure
16.2: Appendix B — Unit Conversion Reference Table
5.11: End-of-Chapter Problem Set
problem definition
6.3: Step 1 — Define
problem identification
13.8: TEGO Case Study
problem set
6.12: End-of-Chapter Problem Set
problem solving
1.11: Quick Check
1.2: A Working Definition
1.3: Engineering vs Science vs Technology
1.4: What Engineers Actually Do
1.5: The Engineering Mindset
15.6: Parameter Sweep
2: Engineering Disciplines and Careers
2.7: One Obligation That Every Discipline Shares
4.3: How to Actually Study Engineering
4.6: Engineering Teamwork
4.8: Summary
5.4: Unit Conversion Method
5.9: From Units to Everything That Follows
6: Structured Engineering Problem Solving
6.10: Structure and Professional Accountability
6.8: Five Common Failure Patterns
8.11: End-of-Chapter Problem Set
problems
12.10: End-of-Chapter Problems
professional
9.3: Anatomy of an Engineering Graph
professional accountability
6.10: Structure and Professional Accountability
Professional Conduct
14.2: NCEES Model Rules
professional email
13: Engineering Communication
13.12: End-of-Chapter Applied Exercises
13.2: Three Modes of Communication
13.9: Professional Email
professional engineer
14: Engineering Ethics
professional engineering
6.7: Step 5 — Communicate
professional engineering graph
9.12: Quick Check
professional ethics
1.9: What This Course Will Build
14.10: Academic Integrity
14.6: Eight-Step Framework
professional identity
1.8: Ethics Is Not a Separate Chapter
professional integrity
13.10: Communicating Uncertainty and Limitations
14.7: Ethics HW Scenario 1
professional judgment
14.9: Ethics HW Scenario 3
5.8: Order-of-Magnitude Reasoning
professional representations
14.3: Ethics HW Preview Table
professional responsibility
14.13: Summary
professional skills
4.6: Engineering Teamwork
professional standards
14.1: Ethics Is Not an Add-On
14.8: Ethics HW Scenario 2
professional writing
13.11: Summary
13.3: The Technical Memo
project
10.8: Programming Project- Cable Car in MATLAB
project management
1.11: Quick Check
1.2: A Working Definition
13.1: Communication Is Part of the Deliverable
14.8: Ethics HW Scenario 2
2.5: Functional Roles
4.6: Engineering Teamwork
project report
7.10: Communicate
properties
12.4: The Sketch as a Mathematical Definition
prototype
7.11: Design, Ethics, and Safety Factors
prototype testing
15.8: Iterate
public accountability
14.13: Summary
public health
14.12: Environmental Responsibility
public safety
14.13: Summary
14.4: Safety Factors
2.7: One Obligation That Every Discipline Shares
public trust
1: What is Engineering?
1.8: Ethics Is Not a Separate Chapter
public welfare
14.1: Ethics Is Not an Add-On
14.12: Environmental Responsibility
14.2: NCEES Model Rules

Q

quality
2.5: Functional Roles
quality control
13: Engineering Communication

quantitative
13.5: Quantitative Precision
6.9: The Workflow Across Disciplines
quantitative analysis
6: Structured Engineering Problem Solving
quantitative engineering
6.11: Summary
quantitative precision
13: Engineering Communication
quantities
16.1: Appendix A — Standard Symbols and Units
5.2: Quantities vs. Numbers
7.7: Model and Analyze
quick reference
16: Appendices
16.4: Appendix D — Spreadsheet Quick Reference (Excel)
9.10: Graph Checklist — Quick Reference

R

radians
10.5: Trigonometry in MATLAB
rating
9.7: Multi-Curve Comparison
reasoning
5.8: Order-of-Magnitude Reasoning
recommendation
13.11: Summary
recommendations
7.10: Communicate
redesign strategies
15.8: Iterate
reference
16.2: Appendix B — Unit Conversion Reference Table
reference table
16: Appendices
refinement
7.9: Iterate and Refine
regulations
14.6: Eight-Step Framework
regulatory compliance
2.5: Functional Roles
relationships
9.1: Why Graphs Are an Engineering Tool
relative reference
8.3: Absolute vs Relative References
relative references
8: Spreadsheet Modeling
8.10: Summary
8.7: Six Common Spreadsheet Errors
reliability
1.1: The Engineer You Don't Notice
7.1: Analysis vs Design
reproducibility
10.1: Why Engineers Program
10.10: Summary
requirements
7.5: Identify Constraints
research and development
2: Engineering Disciplines and Careers
resistance
11: Circuit Fundamentals
11.1: Circuits as Energy Systems
11.10: Power in Networks
11.11: Mixed Circuits
11.13: Circuit Simulator
11.2: Physical Meaning
11.3: Ohm's Law
11.4: Power
11.6: Series Circuits
6.12: End-of-Chapter Problem Set
6.4: Step 2 — Model
9.7: Multi-Curve Comparison
resistor
11.5: Schematic Symbols
13.4: Complete Memo Example
6.12: End-of-Chapter Problem Set
6.3: Step 1 — Define
7.2: What Makes a Design Problem
resistor design
11.12: LED Design
resistor rating
10.6: Conditional Logic
resistor values
5.8: Order-of-Magnitude Reasoning
resistors
11.7: Parallel Circuits
11.8: Series vs Parallel Comparison
7.13: Quick Check
resizing
12.10: End-of-Chapter Problems
response
4.9: End-of-Chapter Exercise
responsibility
14.1: Ethics Is Not an Add-On
14.8: Ethics HW Scenario 2
results
13.11: Summary
13.3: The Technical Memo
13.4: Complete Memo Example
15.9: Bridge Report
16.5: Appendix E — Technical Memo Template
8.2: Structuring an Engineering Spreadsheet
retention
4.3: How to Actually Study Engineering
review
14.8: Ethics HW Scenario 2
revolve
12.3: From Sketch to Solid
12.9: Summary
risk assessment
14.13: Summary
14.9: Ethics HW Scenario 3
risk evaluation
14.11: Citicorp Case Study
risk management
1.8: Ethics Is Not a Separate Chapter
14.4: Safety Factors
2.7: One Obligation That Every Discipline Shares
rounding
6.5: Step 3 — Analyze

S

safe operating range
8.5: Graphical Visualization
safe operating voltage
9.12: Quick Check
safe voltage
10.7: Logical Indexing
safety
1.1: The Engineer You Don't Notice
1.8: Ethics Is Not a Separate Chapter
13.4: Complete Memo Example
14.1: Ethics Is Not an Add-On
14.2: NCEES Model Rules
14.8: Ethics HW Scenario 2
5.1: Why Units Matter
6.12: End-of-Chapter Problem Set
7.1: Analysis vs Design
7.11: Design, Ethics, and Safety Factors
7.12: Summary
7.9: Iterate and Refine
8.11: End-of-Chapter Problem Set
safety checking
10.6: Conditional Logic
safety compliance
14.9: Ethics HW Scenario 3
safety factor
14.14: Quick Check
15.8: Iterate
6.7: Step 5 — Communicate
7.10: Communicate
7.5: Identify Constraints
safety factors
14.13: Summary
14.4: Safety Factors
safety implications
6.11: Summary
safety margin
14.4: Safety Factors
7.13: Quick Check
9.12: Quick Check
safety obligations
14.12: Environmental Responsibility
safety regulation
14.3: Ethics HW Preview Table
safety regulations
14.9: Ethics HW Scenario 3
safety risks
14.6: Eight-Step Framework
salary
2: Engineering Disciplines and Careers
2.4: Engineering by the Numbers
2.9: Quick Check
salutation
13.9: Professional Email
scalability
10.1: Why Engineers Program
scale
9.3: Anatomy of an Engineering Graph
scatter
9.2: Chart Type Selection
schematic
6.3: Step 1 — Define
science
1: What is Engineering?
1.10: Summary
1.11: Quick Check
1.2: A Working Definition
1.3: Engineering vs Science vs Technology
scientific report
15.9: Bridge Report
scope limitation
14.5: Modeling Assumptions

script
10.3: Script Basics
10.8: Programming Project- Cable Car in MATLAB
scripting
10.10: Summary
10.2: MATLAB as Structured Engineering
scripts
10: MATLAB for Engineering Computation
semester planning
4.2: Understanding Your Real Workload
sensitivity analysis
8.5: Graphical Visualization
9.1: Why Graphs Are an Engineering Tool
sensitivity reasoning
6.6: Step 4 — Interpret + Sensitivity
6.8: Five Common Failure Patterns
sequential coursework
3: Academic Pathways and Transfer Strategy
series
11.14: Summary
11.8: Series vs Parallel Comparison
Series Circuits
11.10: Power in Networks
11.15: End-of-Chapter Problem Set
11.6: Series Circuits
11.9: Kirchhoff's Laws
series resistors
11.11: Mixed Circuits
sharing code
14.10: Academic Integrity
SI
16.1: Appendix A — Standard Symbols and Units
5.10: Summary
SI base units
5: Units and Dimensional Analysis
si prefixes
5: Units and Dimensional Analysis
5.5: SI Prefixes
SI System
5.3: SI System — Base and Derived Units
SI units
5.7: Units in Computational Models
signal amplitudes
9.6: Logarithmic Scales
significant
13.5: Quantitative Precision
significant figures
5.4: Unit Conversion Method
simulation
11.13: Circuit Simulator
12.8: CAD in the Engineering Landscape
simulations
10.6: Conditional Logic
sine
10.5: Trigonometry in MATLAB
single calculation
10.4: Four-Step Build
sketch definition
12: Parametric CAD and 3D Modeling
sketching
12.2: Parametric Modeling Core Concepts
skyscraper design
14.11: Citicorp Case Study
slides
13.7: Technical Presentations
societal responsibility
14.12: Environmental Responsibility
soft constraint
7.5: Identify Constraints
soft constraints
15: Integrated Engineering Design Challenge
15.3: Define- Units, Loads, What "Best" Means
software
1.7: Major Disciplines at a Glance
2.8: Summary
software engineering
1.5: The Engineering Mindset
1.6: What Engineers Actually Do
2.2: The Major Engineering Disciplines
2.9: Quick Check
solid modeling
12: Parametric CAD and 3D Modeling
solid models
12.3: From Sketch to Solid
SolidWorks
14.7: Ethics HW Scenario 1
solutions
7.1: Analysis vs Design
specialization
2.1: The Question Most Students Ask
specific question
13.9: Professional Email
specifications
13.1: Communication Is Part of the Deliverable
spreadsheet
10.9: Choosing the Right Tool
16: Appendices
16.4: Appendix D — Spreadsheet Quick Reference (Excel)
8.1: From Calculator to Model
8.2: Structuring an Engineering Spreadsheet
8.3: Absolute vs Relative References
spreadsheet errors
8.7: Six Common Spreadsheet Errors
spreadsheet model
6.9: The Workflow Across Disciplines
spreadsheet modeling
15.10: Course Map
15.6: Parameter Sweep
5.9: From Units to Everything That Follows
8: Spreadsheet Modeling
spreadsheets
8.10: Summary
8.8: Spreadsheets as Design Exploration Tools
8.9: When Spreadsheets Are Not Enough
Stakeholders
14.6: Eight-Step Framework
standard symbols
11.5: Schematic Symbols
16: Appendices
standards
13.3: The Technical Memo
static equilibrium
6.2: Full Workflow Worked Example First
6.7: Step 5 — Communicate
static loading
13.10: Communicating Uncertainty and Limitations
14.5: Modeling Assumptions
static visual
12.1: CAD Is Not Drawing
statics
3.10: Quick Check
statistical analysis
9.9: Advanced Chart Features
steel
12.4: The Sketch as a Mathematical Definition
STEM fields
3.6: Transfer Strategy
stress
16.2: Appendix B — Unit Conversion Reference Table
stress analysis
15.4: Model- Tension, Compression, Straws
structural
16.1: Appendix A — Standard Symbols and Units
6.9: The Workflow Across Disciplines
structural analysis
13.10: Communicating Uncertainty and Limitations
15.6: Parameter Sweep
15.7: Alternatives and Decision Matrix
8.9: When Spreadsheets Are Not Enough
structural design
15.5: Mass Budget
7.6: Generate Alternatives
structural engineering
14.11: Citicorp Case Study
15.4: Model- Tension, Compression, Straws
6.11: Summary
7.4: Define the Design Objective
structural error
4.4: Understanding Your Errors
structural force
5.8: Order-of-Magnitude Reasoning
structural load
15.3: Define- Units, Loads, What "Best" Means
structural steel stress
5.8: Order-of-Magnitude Reasoning
structure
13.3: The Technical Memo
6.1: Why Structure Matters
6.10: Structure and Professional Accountability
structured communication
6.7: Step 5 — Communicate
structured engineering
10.2: MATLAB as Structured Engineering
6: Structured Engineering Problem Solving
structured problem solving
1.9: What This Course Will Build
15.10: Course Map
structured thinking
6: Structured Engineering Problem Solving
student mistakes
4.4: Understanding Your Errors
student strategies
4.2: Understanding Your Real Workload
student support
4.7: Sustaining Yourself Over the Long Haul
study
3.7: Academic Risk Factors
4.9: End-of-Chapter Exercise
study habits
4.1: It Is Not About Talent
4.7: Sustaining Yourself Over the Long Haul
4.8: Summary
study hours
4.2: Understanding Your Real Workload
study skills
4: Engineering Success Systems

study strategies
4.3: How to Actually Study Engineering

study techniques
4.3: How to Actually Study Engineering

subject line
13.9: Professional Email

supervision
14.8: Ethics HW Scenario 2

supervisory responsibility
14.3: Ethics HW Preview Table

supply voltage
11.12: LED Design

suspended load
6.2: Full Workflow Worked Example First

suspension bridge
15.7: Alternatives and Decision Matrix

sustainable performance
4.8: Summary

swept variable
8.3: Absolute vs Relative References

switch
11.5: Schematic Symbols

symbols
16.1: Appendix A — Standard Symbols and Units

symmetric loading
6.7: Step 5 — Communicate

syntax
10.3: Script Basics

systematic approach
1.9: What This Course Will Build

systems
1.10: Summary
1.2: A Working Definition
11: Circuit Fundamentals
7.1: Analysis vs Design

systems engineering
1.6: What Engineers Actually Do
2.5: Functional Roles

T

tables
13: Engineering Communication
13.6: Graphs and Tables

tangent
10.5: Trigonometry in MATLAB

team roles
4.6: Engineering Teamwork

teamwork
4: Engineering Success Systems
4.6: Engineering Teamwork
4.8: Summary

technical
13.5: Quantitative Precision
16.5: Appendix E — Technical Memo Template

technical communication
1.9: What This Course Will Build

technical decisions
1.8: Ethics Is Not a Separate Chapter

technical GPA
3.6: Transfer Strategy

technical knowledge
1.5: The Engineering Mindset

technical memo
13.10: Communicating Uncertainty and Limitations
13.2: Three Modes of Communication
15.9: Bridge Report

technical memo template
16: Appendices

technical memos
13.11: Summary

technical presentation
13.2: Three Modes of Communication

technical presentations
13.11: Summary
13.7: Technical Presentations

technical skills
4: Engineering Success Systems

technical writing
13: Engineering Communication
13.12: End-of-Chapter Applied Exercises
13.3: The Technical Memo

technological development
1.11: Quick Check

technology
1: What is Engineering?
1.1: The Engineer You Don't Notice
1.10: Summary
1.2: A Working Definition
1.3: Engineering vs Science vs Technology

TEGO
13.8: TEGO Case Study

TEGO homework
13.12: End-of-Chapter Applied Exercises

temperature
5.3: SI System — Base and Derived Units
9.5: Misleading Visualizations

template
16.5: Appendix E — Technical Memo Template

tension
15: Integrated Engineering Design Challenge
15.11: Summary
15.12: End-of-Chapter Problem Set
15.4: Model- Tension, Compression, Straws
6.12: End-of-Chapter Problem Set
6.2: Full Workflow Worked Example First
8.1: From Calculator to Model
8.3: Absolute vs Relative References
8.4: Parameter Sweeps
8.8: Spreadsheets as Design Exploration Tools

tension calculation
8.4: Parameter Sweeps

tension limits
10.7: Logical Indexing

tension vs angle
8.5: Graphical Visualization
9.12: Quick Check

tera
5.5: SI Prefixes

test
2.5: Functional Roles

testing
1.10: Summary
1.4: What Engineers Actually Do

testing analysis
15.8: Iterate

Textbook
6.2: Full Workflow Worked Example First

theoretical model
9.9: Advanced Chart Features

time
5.3: SI System — Base and Derived Units

time management
4.2: Understanding Your Real Workload
4.3: How to Actually Study Engineering
4.5: Goals, Humility, and the Long Game
4.7: Sustaining Yourself Over the Long Haul
4.8: Summary
4.9: End-of-Chapter Exercise

title
9.3: Anatomy of an Engineering Graph

tool
10.9: Choosing the Right Tool

tool selection
10: MATLAB for Engineering Computation

tools
15.1: This Is Not a New Topic

total power
11.10: Power in Networks

transfer destinations
3.5: Degree Levels and Transfer Destinations

transfer process
3.8: Treating Your Academic Plan Like an Engineering Problem

transfer strategy
3: Academic Pathways and Transfer Strategy
3.6: Transfer Strategy

transfer students
3.10: Quick Check

transparency
14.5: Modeling Assumptions

transparency in communication
14.13: Summary

trend analysis
13.6: Graphs and Tables

trendline
9.10: Graph Checklist — Quick Reference

trendlines
9.11: Summary
9.9: Advanced Chart Features

trends
9: Data Visualization for Engineers

trials
10.8: Programming Project- Cable Car in MATLAB

trigonometric functions
8.10: Summary
8.3: Absolute vs Relative References

trigonometry
10.5: Trigonometry in MATLAB
3.1: Engineering Is a Structured Academic Path
3.2: The Mathematics Backbone

truss
15.7: Alternatives and Decision Matrix

truthfulness
14.3: Ethics HW Preview Table

tutoring
3.7: Academic Risk Factors

tutoring services
4.7: Sustaining Yourself Over the Long Haul

type
9.2: Chart Type Selection

U

uncertainty
14.1: Ethics Is Not an Add-On

unique answer
7.2: What Makes a Design Problem
unit
5.10: Summary
unit consistency
12.10: End-of-Chapter Problems
5.9: From Units to Everything That Follows
unit conversion
15.10: Course Map
15.6: Parameter Sweep
16: Appendices
5: Units and Dimensional Analysis
5.10: Summary
5.4: Unit Conversion Method
6.11: Summary
unit conversions
5.11: End-of-Chapter Problem Set
6.5: Step 3 — Analyze
unit error
4.4: Understanding Your Errors
unit errors
5: Units and Dimensional Analysis
unit neglect
6.8: Five Common Failure Patterns
unit specification
12.7: Five Common Modeling Errors
units
11.3: Ohm's Law
15.3: Define- Units, Loads, What "Best" Means
16: Appendices
16.1: Appendix A — Standard Symbols and Units
16.2: Appendix B — Unit Conversion Reference Table
5: Units and Dimensional Analysis
5.1: Why Units Matter
5.2: Quantities vs. Numbers
5.4: Unit Conversion Method
5.7: Units in Computational Models
5.9: From Units to Everything That Follows
unlabeled inputs
8.7: Six Common Spreadsheet Errors

v

variables
10.3: Script Basics
16.3: Appendix C — MATLAB Quick Reference
7.2: What Makes a Design Problem
vector
10.4: Four-Step Build
vectors
16.3: Appendix C — MATLAB Quick Reference
velocity equations
8.6: Conditional Formatting and IF Logic
verification
12.4: The Sketch as a Mathematical Definition
13.5: Quantitative Precision
verifying equations
5.6: Dimensional Analysis
visual
10.9: Choosing the Right Tool
visual communication
9.11: Summary
visual data representation
8.6: Conditional Formatting and IF Logic
visualization
13.6: Graphs and Tables
8.9: When Spreadsheets Are Not Enough
9.3: Anatomy of an Engineering Graph
9.8: Visualization and Interpretation
visuals
13.7: Technical Presentations
voltage
10.6: Conditional Logic
11: Circuit Fundamentals
11.1: Circuits as Energy Systems
11.13: Circuit Simulator
11.15: End-of-Chapter Problem Set
11.2: Physical Meaning
11.3: Ohm's Law
11.4: Power
11.6: Series Circuits
11.7: Parallel Circuits
11.8: Series vs Parallel Comparison
6.12: End-of-Chapter Problem Set
6.4: Step 2 — Model
8.11: End-of-Chapter Problem Set
9.7: Multi-Curve Comparison
voltage acceptance
10.7: Logical Indexing
voltage divider
11.6: Series Circuits
voltage drop
11.10: Power in Networks
voltage vector
10.11: End-of-Chapter Problem Set
voltmeter
11.5: Schematic Symbols
volume
12.4: The Sketch as a Mathematical Definition

w

wages
2.9: Quick Check
Warren truss
15.11: Summary
15.7: Alternatives and Decision Matrix
weights
7.8: Evaluate Trade-offs
William LeMessurier
14.11: Citicorp Case Study
wind loads
14.11: Citicorp Case Study
worked example
11.11: Mixed Circuits
11.4: Power
11.6: Series Circuits
8.4: Parameter Sweeps
worked examples
5.6: Dimensional Analysis
worker safety
14.9: Ethics HW Scenario 3
workflow
12.6: Modeling Workflow
13.1: Communication Is Part of the Deliverable
6.1: Why Structure Matters
6.2: Full Workflow Worked Example First
6.9: The Workflow Across Disciplines
7.7: Model and Analyze
workload
4.2: Understanding Your Real Workload
writing
13.5: Quantitative Precision

x

XY scatter
9.10: Graph Checklist — Quick Reference
XY scatter plot
8.5: Graphical Visualization
XY scatter plots
8.10: Summary
9.11: Summary

This page titled Index is shared under a license and was authored, remixed, and/or curated by .

GLOSSARY

A

Absolute Cell Reference | A spreadsheet reference (e.g., `$B$1`) that does not change when copied. Used to keep constants fixed during parameter sweeps.

Allowable Stress | The maximum stress a component is permitted to experience: $\sigma_{\text{allow}} = \sigma_{\text{yield}}/SF$.

Ampere (A) | The SI unit of electric current. One ampere equals one coulomb of charge per second.

Assumption | A stated simplification accepted for modeling purposes. Must be documented in the Background section of any technical memo.

Auditable | The standard that an engineering document or model can be reviewed and verified by another engineer not involved in creating it.

B

Buckling | The sudden lateral collapse of a compression member at loads well below material yield stress. Bundling members and keeping them short both increase buckling resistance.

C

CAD (Computer-Aided Design) | Software that creates precise digital geometry models. Parametric CAD (Fusion 360, SolidWorks) defines geometry through dimensions and constraints so the model updates when parameters change.

Conditional Formatting | An Excel feature that changes cell appearance based on its value. Used in the Cable Car Project to provide a visual velocity map.

Constraint | A limitation or requirement on a design. Hard constraints cannot be violated. Soft constraints are objectives to optimize within the hard constraint boundaries.

Current (I) | The rate of electric charge flow, measured in amperes. $1\ \text{A} = 1\ \text{C/s}$.

D

Decision Matrix | A tool for comparing design alternatives against weighted criteria. Makes the basis of a design choice explicit and auditable.

Design Intent | The engineering logic embedded in a parametric CAD model that determines how geometry behaves when parameters change.

Dimensional Analysis | Verifying equations and converting units by treating units algebraically. Units must cancel to produce the correct unit for the result.

E

Element-Wise Operators | MATLAB operators (`./ .* .^`) that apply arithmetic to each element of a vector individually rather than performing matrix operations.

Equivalent Resistance | A single resistance representing a combination: $R_{\text{eq}} = \sum R_k$ (series); $1/R_{\text{eq}} = \sum 1/R_k$ (parallel).

Ethics (Engineering) | The professional obligation to hold paramount the safety, health, and welfare of the public in all engineering decisions. Governed by the NCEES Model Rules of Professional Conduct.

Extrude | A CAD operation that extends a 2D sketch profile into a 3D solid body.

F

Feature Tree | The sequential record of all modeling operations in a parametric CAD system. Can be edited, but later features that depend on earlier ones may be affected.

Fully Constrained Sketch | A CAD sketch with no remaining degrees of freedom — shown in black in Fusion 360. Under-constrained sketches (blue) can drift unexpectedly.

I

IF / IFS Function | Excel functions for piecewise models. `=IF(condition, true_value, false_value)`. IFS handles multiple conditions more cleanly than nested IF.

Iteration | The process of systematically refining a design toward the objective while staying within constraints. Not redoing work because of errors.

K

Kirchhoff's Current Law (KCL) | Currents entering a node equal currents leaving. An expression of conservation of charge.

Kirchhoff's Voltage Law (KVL) | Voltages around any closed loop sum to zero. An expression of conservation of energy.

L

Limitation Statement | An explicit statement of when an analysis is valid and when it is not. Omitting a known limitation is misleading communication.

Logical Indexing | A MATLAB technique for extracting vector elements satisfying a condition: `V(P < 0.25)` returns only voltages where power is safe.

N

NCEES Model Rules | The National Council of Examiners for Engineering and Surveying Model Rules of Professional Conduct. Primary obligation: hold paramount the safety, health, and welfare of the public.

Nonlinear Relationship | A relationship where the rate of change is not constant. $P = V^2/R$ is quadratic: doubling voltage quadruples power.

O

Ohm's Law | The linear relationship $V = IR$ between voltage, current, and resistance.

P

Parallel Circuit | Circuit topology where all components connect between the same two nodes. Voltage same across all branches; current divides. $1/R_{\text{eq}} = \sum 1/R_k$.

Parameter Sweep | Systematic variation of one input across a range, with output computed at each value. The primary technique for understanding system behavior across an operating range.

Parametric Modeling | A CAD approach where geometry is defined through adjustable dimensions and constraints, updating automatically when parameters change.

Power (P) | Rate of energy transfer in watts. $P = VI = I^2R = V^2/R$. Grows with the square of voltage — the primary safety check in circuit design.

R

RADIANS() | The Excel function that converts degrees to radians. Must wrap all trig arguments: `=SIN(RADIANS(A10))`. Forgetting produces a silent wrong answer.

Relative Cell Reference | A spreadsheet reference (e.g., `B3`) that shifts when copied. Used for the swept variable in a parameter sweep.

S

Safety Factor (SF) | Ratio of allowable capacity to actual load: $SF = \sigma_{\text{allow}}/\sigma_{\text{actual}}$. An ethical acknowledgment of irreducible uncertainty.

Scientific Report Format | The standard format for reporting experimental work: Title, Abstract, Introduction, Materials and Methods, Results, Discussion, Conclusion. Used for the Bridge Challenge report.

Series Circuit | Circuit topology where all components share one current path. Current same everywhere; voltage divides. $R_{\text{eq}} = \sum R_k$.

sind() / cosd() / tand() | MATLAB functions accepting angles in degrees directly. Preferred over `sin()`/`cos()`/`tan()` when angles are in degrees. The `d` suffix stands for "degrees."

Stress (σ) | Force per unit area: $\sigma = F/A$. Measured in pascals (Pa) or megapascals (MPa).

Structured Problem Solving | The engineering workflow: **Define** → **Model** → **Analyze** → **Interpret** → **Communicate**. Applies universally across calculations, models, scripts, circuits, and documents.

T

Technical Memo | A structured engineering communication document with six sections: Purpose, Background, Methodology, Results, Analysis, Conclusion and Recommendation.

Tension | A pulling force in a structural member. Members in tension are being stretched. In bridge structures, the bottom chord is typically in tension under a center load.

U

Unit Discipline | The consistent and correct use of units throughout all modeling and analysis. Converting to consistent units before substitution is the first line of defense against silent errors.

V

Voltage (V) | Electric potential difference measured in volts. $1\ \mathrm{V} = 1\ \mathrm{J/C}$. Drives current through a circuit.

This page titled Glossary is shared under a license and was authored, remixed, and/or curated by .

DETAILED LICENSING

Overview

Title: Introduction to Engineering - Thinking Like an Engineer

Webpages: 205

Applicable Restrictions: Noncommercial

All licenses found:

- CC BY-NC 4.0: 95.1% (195 pages)
- Undeclared: 4.4% (9 pages)
- Other: 0.5% (1 page)

By Page

- Introduction to Engineering - Thinking Like an Engineer - *CC BY-NC 4.0*
  - Front Matter - *Undeclared*
    - TitlePage - *Undeclared*
    - InfoPage - *Other*
    - Table of Contents - *Undeclared*
    - Licensing - *Undeclared*
    - Preface - *CC BY-NC 4.0*
  - 1: What is Engineering? - *CC BY-NC 4.0*
    - 1.1: The Engineer You Don't Notice - *CC BY-NC 4.0*
    - 1.2: A Working Definition - *CC BY-NC 4.0*
    - 1.3: Engineering vs Science vs Technology - *CC BY-NC 4.0*
    - 1.4: What Engineers Actually Do - *CC BY-NC 4.0*
    - 1.5: The Engineering Mindset - *CC BY-NC 4.0*
    - 1.6: What Engineers Actually Do - *CC BY-NC 4.0*
    - 1.7: Major Disciplines at a Glance - *CC BY-NC 4.0*
    - 1.8: Ethics Is Not a Separate Chapter - *CC BY-NC 4.0*
    - 1.9: What This Course Will Build - *CC BY-NC 4.0*
    - 1.10: Summary - *CC BY-NC 4.0*
    - 1.11: Quick Check - *CC BY-NC 4.0*
  - 2: Engineering Disciplines and Careers - *CC BY-NC 4.0*
    - 2.1: The Question Most Students Ask - *CC BY-NC 4.0*
    - 2.2: The Major Engineering Disciplines - *CC BY-NC 4.0*
    - 2.3: Most Real Problems Cross Discipline Boundaries - *CC BY-NC 4.0*
    - 2.4: Engineering by the Numbers - *CC BY-NC 4.0*
    - 2.5: Functional Roles - *CC BY-NC 4.0*
    - 2.6: Choosing a Discipline - *CC BY-NC 4.0*
    - 2.7: One Obligation That Every Discipline Shares - *CC BY-NC 4.0*
    - 2.8: Summary - *CC BY-NC 4.0*
    - 2.9: Quick Check - *CC BY-NC 4.0*
  - 3: Academic Pathways and Transfer Strategy - *CC BY-NC 4.0*
    - 3.1: Engineering Is a Structured Academic Path - *CC BY-NC 4.0*
    - 3.2: The Mathematics Backbone - *CC BY-NC 4.0*
    - 3.3: Physics as Applied Mathematics - *CC BY-NC 4.0*
    - 3.4: Typical Lower-Division Engineering Path (Community College) - *CC BY-NC 4.0*
    - 3.5: Degree Levels and Transfer Destinations - *CC BY-NC 4.0*
    - 3.6: Transfer Strategy - *CC BY-NC 4.0*
    - 3.7: Academic Risk Factors - *CC BY-NC 4.0*
    - 3.8: Treating Your Academic Plan Like an Engineering Problem - *CC BY-NC 4.0*
    - 3.9: Summary - *CC BY-NC 4.0*
    - 3.10: Quick Check - *CC BY-NC 4.0*
  - 4: Engineering Success Systems - *CC BY-NC 4.0*
    - 4.1: It Is Not About Talent - *CC BY-NC 4.0*
    - 4.2: Understanding Your Real Workload - *CC BY-NC 4.0*
    - 4.3: How to Actually Study Engineering - *CC BY-NC 4.0*
    - 4.4: Understanding Your Errors - *CC BY-NC 4.0*
    - 4.5: Goals, Humility, and the Long Game - *CC BY-NC 4.0*
    - 4.6: Engineering Teamwork - *CC BY-NC 4.0*
    - 4.7: Sustaining Yourself Over the Long Haul - *CC BY-NC 4.0*
    - 4.8: Summary - *CC BY-NC 4.0*
    - 4.9: End-of-Chapter Exercise - *CC BY-NC 4.0*
  - 5: Units and Dimensional Analysis - *CC BY-NC 4.0*
    - 5.1: Why Units Matter - *CC BY-NC 4.0*

- 5.2: Quantities vs. Numbers - *CC BY-NC 4.0*
- 5.3: SI System — Base and Derived Units - *CC BY-NC 4.0*
- 5.4: Unit Conversion Method - *CC BY-NC 4.0*
- 5.5: SI Prefixes - *CC BY-NC 4.0*
- 5.6: Dimensional Analysis - *CC BY-NC 4.0*
- 5.7: Units in Computational Models - *CC BY-NC 4.0*
- 5.8: Order-of-Magnitude Reasoning - *CC BY-NC 4.0*
- 5.9: From Units to Everything That Follows - *CC BY-NC 4.0*
- 5.10: Summary - *CC BY-NC 4.0*
- 5.11: End-of-Chapter Problem Set - *CC BY-NC 4.0*

◦ 6: Structured Engineering Problem Solving - *CC BY-NC 4.0*
- 6.1: Why Structure Matters - *CC BY-NC 4.0*
- 6.2: Full Workflow Worked Example First - *CC BY-NC 4.0*
- 6.3: Step 1 — Define - *CC BY-NC 4.0*
- 6.4: Step 2 — Model - *CC BY-NC 4.0*
- 6.5: Step 3 — Analyze - *CC BY-NC 4.0*
- 6.6: Step 4 — Interpret + Sensitivity - *CC BY-NC 4.0*
- 6.7: Step 5 — Communicate - *CC BY-NC 4.0*
- 6.8: Five Common Failure Patterns - *CC BY-NC 4.0*
- 6.9: The Workflow Across Disciplines - *CC BY-NC 4.0*
- 6.10: Structure and Professional Accountability - *CC BY-NC 4.0*
- 6.11: Summary - *CC BY-NC 4.0*
- 6.12: End-of-Chapter Problem Set - *CC BY-NC 4.0*

◦ 7: The Engineering Design Process - *CC BY-NC 4.0*
- 7.1: Analysis vs Design - *CC BY-NC 4.0*
- 7.2: What Makes a Design Problem - *CC BY-NC 4.0*
- 7.3: Seven-Step Design Framework - *CC BY-NC 4.0*
- 7.4: Define the Design Objective - *CC BY-NC 4.0*
- 7.5: Identify Constraints - *CC BY-NC 4.0*
- 7.6: Generate Alternatives - *CC BY-NC 4.0*
- 7.7: Model and Analyze - *CC BY-NC 4.0*
- 7.8: Evaluate Trade-offs - *CC BY-NC 4.0*
- 7.9: Iterate and Refine - *CC BY-NC 4.0*
- 7.10: Communicate - *CC BY-NC 4.0*
- 7.11: Design, Ethics, and Safety Factors - *CC BY-NC 4.0*
- 7.12: Summary - *CC BY-NC 4.0*
- 7.13: Quick Check - *CC BY-NC 4.0*

◦ 8: Spreadsheet Modeling - *CC BY-NC 4.0*
- 8.1: From Calculator to Model - *CC BY-NC 4.0*
- 8.2: Structuring an Engineering Spreadsheet - *CC BY-NC 4.0*
- 8.3: Absolute vs Relative References - *CC BY-NC 4.0*
- 8.4: Parameter Sweeps - *CC BY-NC 4.0*
- 8.5: Graphical Visualization - *CC BY-NC 4.0*
- 8.6: Conditional Formatting and IF Logic - *CC BY-NC 4.0*
- 8.7: Six Common Spreadsheet Errors - *CC BY-NC 4.0*
- 8.8: Spreadsheets as Design Exploration Tools - *CC BY-NC 4.0*
- 8.9: When Spreadsheets Are Not Enough - *CC BY-NC 4.0*
- 8.10: Summary - *CC BY-NC 4.0*
- 8.11: End-of-Chapter Problem Set - *CC BY-NC 4.0*

◦ 9: Data Visualization for Engineers - *CC BY-NC 4.0*
- 9.1: Why Graphs Are an Engineering Tool - *CC BY-NC 4.0*
- 9.2: Chart Type Selection - *CC BY-NC 4.0*
- 9.3: Anatomy of an Engineering Graph - *CC BY-NC 4.0*
- 9.4: Linear vs Nonlinear Trends - *CC BY-NC 4.0*
- 9.5: Misleading Visualizations - *CC BY-NC 4.0*
- 9.6: Logarithmic Scales - *CC BY-NC 4.0*
- 9.7: Multi-Curve Comparison - *CC BY-NC 4.0*
- 9.8: Visualization and Interpretation - *CC BY-NC 4.0*
- 9.9: Advanced Chart Features - *CC BY-NC 4.0*
- 9.10: Graph Checklist — Quick Reference - *CC BY-NC 4.0*
- 9.11: Summary - *CC BY-NC 4.0*
- 9.12: Quick Check - *CC BY-NC 4.0*

◦ 10: MATLAB for Engineering Computation - *CC BY-NC 4.0*
- 10.1: Why Engineers Program - *CC BY-NC 4.0*
- 10.2: MATLAB as Structured Engineering - *CC BY-NC 4.0*
- 10.3: Script Basics - *CC BY-NC 4.0*
- 10.4: Four-Step Build - *CC BY-NC 4.0*
- 10.5: Trigonometry in MATLAB - *CC BY-NC 4.0*
- 10.6: Conditional Logic - *CC BY-NC 4.0*
- 10.7: Logical Indexing - *CC BY-NC 4.0*
- 10.8: Programming Project- Cable Car in MATLAB - *CC BY-NC 4.0*
- 10.9: Choosing the Right Tool - *CC BY-NC 4.0*
- 10.10: Summary - *CC BY-NC 4.0*
- 10.11: End-of-Chapter Problem Set - *CC BY-NC 4.0*

◦ 11: Circuit Fundamentals - *CC BY-NC 4.0*
- 11.1: Circuits as Energy Systems - *CC BY-NC 4.0*
- 11.2: Physical Meaning - *CC BY-NC 4.0*
- 11.3: Ohm's Law - *CC BY-NC 4.0*
- 11.4: Power - *CC BY-NC 4.0*
- 11.5: Schematic Symbols - *CC BY-NC 4.0*
- 11.6: Series Circuits - *CC BY-NC 4.0*
- 11.7: Parallel Circuits - *CC BY-NC 4.0*
- 11.8: Series vs Parallel Comparison - *CC BY-NC 4.0*
- 11.9: Kirchhoff's Laws - *CC BY-NC 4.0*

- 11.10: Power in Networks - *CC BY-NC 4.0*
- 11.11: Mixed Circuits - *CC BY-NC 4.0*
- 11.12: LED Design - *CC BY-NC 4.0*
- 11.13: Circuit Simulator - *CC BY-NC 4.0*
- 11.14: Summary - *CC BY-NC 4.0*
- 11.15: End-of-Chapter Problem Set - *CC BY-NC 4.0*

◦ 12: Parametric CAD and 3D Modeling - *CC BY-NC 4.0*
- 12.1: CAD Is Not Drawing - *CC BY-NC 4.0*
- 12.2: Parametric Modeling Core Concepts - *CC BY-NC 4.0*
- 12.3: From Sketch to Solid - *CC BY-NC 4.0*
- 12.4: The Sketch as a Mathematical Definition - *CC BY-NC 4.0*
- 12.5: The Fusion Project - *CC BY-NC 4.0*
- 12.6: Modeling Workflow - *CC BY-NC 4.0*
- 12.7: Five Common Modeling Errors - *CC BY-NC 4.0*
- 12.8: CAD in the Engineering Landscape - *CC BY-NC 4.0*
- 12.9: Summary - *CC BY-NC 4.0*
- 12.10: End-of-Chapter Problems - *CC BY-NC 4.0*

◦ 13: Engineering Communication - *CC BY-NC 4.0*
- 13.1: Communication Is Part of the Deliverable - *CC BY-NC 4.0*
- 13.2: Three Modes of Communication - *CC BY-NC 4.0*
- 13.3: The Technical Memo - *CC BY-NC 4.0*
- 13.4: Complete Memo Example - *CC BY-NC 4.0*
- 13.5: Quantitative Precision - *CC BY-NC 4.0*
- 13.6: Graphs and Tables - *CC BY-NC 4.0*
- 13.7: Technical Presentations - *CC BY-NC 4.0*
- 13.8: TEGO Case Study - *CC BY-NC 4.0*
- 13.9: Professional Email - *CC BY-NC 4.0*
- 13.10: Communicating Uncertainty and Limitations - *CC BY-NC 4.0*
- 13.11: Summary - *CC BY-NC 4.0*
- 13.12: End-of-Chapter Applied Exercises - *CC BY-NC 4.0*

◦ 14: Engineering Ethics - *CC BY-NC 4.0*
- 14.1: Ethics Is Not an Add-On - *CC BY-NC 4.0*
- 14.2: NCEES Model Rules - *CC BY-NC 4.0*
- 14.3: Ethics HW Preview Table - *CC BY-NC 4.0*
- 14.4: Safety Factors - *CC BY-NC 4.0*
- 14.5: Modeling Assumptions - *CC BY-NC 4.0*
- 14.6: Eight-Step Framework - *CC BY-NC 4.0*
- 14.7: Ethics HW Scenario 1 - *CC BY-NC 4.0*
- 14.8: Ethics HW Scenario 2 - *CC BY-NC 4.0*
- 14.9: Ethics HW Scenario 3 - *CC BY-NC 4.0*
- 14.10: Academic Integrity - *CC BY-NC 4.0*
- 14.11: Citicorp Case Study - *CC BY-NC 4.0*
- 14.12: Environmental Responsibility - *CC BY-NC 4.0*
- 14.13: Summary - *CC BY-NC 4.0*
- 14.14: Quick Check - *CC BY-NC 4.0*

◦ 15: Integrated Engineering Design Challenge - *CC BY-NC 4.0*
- 15.1: This Is Not a New Topic - *CC BY-NC 4.0*
- 15.2: Challenge Specifications - *CC BY-NC 4.0*
- 15.3: Define- Units, Loads, What "Best" Means - *CC BY-NC 4.0*
- 15.4: Model- Tension, Compression, Straws - *CC BY-NC 4.0*
- 15.5: Mass Budget - *CC BY-NC 4.0*
- 15.6: Parameter Sweep - *CC BY-NC 4.0*
- 15.7: Alternatives and Decision Matrix - *CC BY-NC 4.0*
- 15.8: Iterate - *CC BY-NC 4.0*
- 15.9: Bridge Report - *CC BY-NC 4.0*
- 15.10: Course Map - *CC BY-NC 4.0*
- 15.11: Summary - *CC BY-NC 4.0*
- 15.12: End-of-Chapter Problem Set - *CC BY-NC 4.0*

◦ 16: Appendices - *CC BY-NC 4.0*
- 16.1: Appendix A — Standard Symbols and Units - *CC BY-NC 4.0*
- 16.2: Appendix B — Unit Conversion Reference Table - *CC BY-NC 4.0*
- 16.3: Appendix C — MATLAB Quick Reference - *CC BY-NC 4.0*
- 16.4: Appendix D — Spreadsheet Quick Reference (Excel) - *CC BY-NC 4.0*
- 16.5: Appendix E — Technical Memo Template - *CC BY-NC 4.0*

◦ Back Matter - *Undeclared*
- Index - *Undeclared*
- Glossary - *Undeclared*
- Detailed Licensing - *Undeclared*
- Detailed Licensing - *Undeclared*

This page titled Introduction to Engineering - Thinking Like an Engineer is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by Jonathan Compton.

DETAILED LICENSING

Overview

Title: Introduction to Engineering - Thinking Like an Engineer

Webpages: 205

Applicable Restrictions: Noncommercial

All licenses found:

- CC BY-NC 4.0: 95.1% (195 pages)
- Undeclared: 4.4% (9 pages)
- Other: 0.5% (1 page)

By Page

- Introduction to Engineering - Thinking Like an Engineer - *CC BY-NC 4.0*
  - Front Matter - *Undeclared*
    - TitlePage - *Undeclared*
    - InfoPage - *Other*
    - Table of Contents - *Undeclared*
    - Licensing - *Undeclared*
    - Preface - *CC BY-NC 4.0*
  - 1: What is Engineering? - *CC BY-NC 4.0*
    - 1.1: The Engineer You Don't Notice - *CC BY-NC 4.0*
    - 1.2: A Working Definition - *CC BY-NC 4.0*
    - 1.3: Engineering vs Science vs Technology - *CC BY-NC 4.0*
    - 1.4: What Engineers Actually Do - *CC BY-NC 4.0*
    - 1.5: The Engineering Mindset - *CC BY-NC 4.0*
    - 1.6: What Engineers Actually Do - *CC BY-NC 4.0*
    - 1.7: Major Disciplines at a Glance - *CC BY-NC 4.0*
    - 1.8: Ethics Is Not a Separate Chapter - *CC BY-NC 4.0*
    - 1.9: What This Course Will Build - *CC BY-NC 4.0*
    - 1.10: Summary - *CC BY-NC 4.0*
    - 1.11: Quick Check - *CC BY-NC 4.0*
  - 2: Engineering Disciplines and Careers - *CC BY-NC 4.0*
    - 2.1: The Question Most Students Ask - *CC BY-NC 4.0*
    - 2.2: The Major Engineering Disciplines - *CC BY-NC 4.0*
    - 2.3: Most Real Problems Cross Discipline Boundaries - *CC BY-NC 4.0*
    - 2.4: Engineering by the Numbers - *CC BY-NC 4.0*
    - 2.5: Functional Roles - *CC BY-NC 4.0*
    - 2.6: Choosing a Discipline - *CC BY-NC 4.0*
    - 2.7: One Obligation That Every Discipline Shares - *CC BY-NC 4.0*
    - 2.8: Summary - *CC BY-NC 4.0*
    - 2.9: Quick Check - *CC BY-NC 4.0*
  - 3: Academic Pathways and Transfer Strategy - *CC BY-NC 4.0*
    - 3.1: Engineering Is a Structured Academic Path - *CC BY-NC 4.0*
    - 3.2: The Mathematics Backbone - *CC BY-NC 4.0*
    - 3.3: Physics as Applied Mathematics - *CC BY-NC 4.0*
    - 3.4: Typical Lower-Division Engineering Path (Community College) - *CC BY-NC 4.0*
    - 3.5: Degree Levels and Transfer Destinations - *CC BY-NC 4.0*
    - 3.6: Transfer Strategy - *CC BY-NC 4.0*
    - 3.7: Academic Risk Factors - *CC BY-NC 4.0*
    - 3.8: Treating Your Academic Plan Like an Engineering Problem - *CC BY-NC 4.0*
    - 3.9: Summary - *CC BY-NC 4.0*
    - 3.10: Quick Check - *CC BY-NC 4.0*
  - 4: Engineering Success Systems - *CC BY-NC 4.0*
    - 4.1: It Is Not About Talent - *CC BY-NC 4.0*
    - 4.2: Understanding Your Real Workload - *CC BY-NC 4.0*
    - 4.3: How to Actually Study Engineering - *CC BY-NC 4.0*
    - 4.4: Understanding Your Errors - *CC BY-NC 4.0*
    - 4.5: Goals, Humility, and the Long Game - *CC BY-NC 4.0*
    - 4.6: Engineering Teamwork - *CC BY-NC 4.0*
    - 4.7: Sustaining Yourself Over the Long Haul - *CC BY-NC 4.0*
    - 4.8: Summary - *CC BY-NC 4.0*
    - 4.9: End-of-Chapter Exercise - *CC BY-NC 4.0*
  - 5: Units and Dimensional Analysis - *CC BY-NC 4.0*
    - 5.1: Why Units Matter - *CC BY-NC 4.0*

- 5.2: Quantities vs. Numbers - *CC BY-NC 4.0*
- 5.3: SI System — Base and Derived Units - *CC BY-NC 4.0*
- 5.4: Unit Conversion Method - *CC BY-NC 4.0*
- 5.5: SI Prefixes - *CC BY-NC 4.0*
- 5.6: Dimensional Analysis - *CC BY-NC 4.0*
- 5.7: Units in Computational Models - *CC BY-NC 4.0*
- 5.8: Order-of-Magnitude Reasoning - *CC BY-NC 4.0*
- 5.9: From Units to Everything That Follows - *CC BY-NC 4.0*
- 5.10: Summary - *CC BY-NC 4.0*
- 5.11: End-of-Chapter Problem Set - *CC BY-NC 4.0*

◦ 6: Structured Engineering Problem Solving - *CC BY-NC 4.0*
- 6.1: Why Structure Matters - *CC BY-NC 4.0*
- 6.2: Full Workflow Worked Example First - *CC BY-NC 4.0*
- 6.3: Step 1 — Define - *CC BY-NC 4.0*
- 6.4: Step 2 — Model - *CC BY-NC 4.0*
- 6.5: Step 3 — Analyze - *CC BY-NC 4.0*
- 6.6: Step 4 — Interpret + Sensitivity - *CC BY-NC 4.0*
- 6.7: Step 5 — Communicate - *CC BY-NC 4.0*
- 6.8: Five Common Failure Patterns - *CC BY-NC 4.0*
- 6.9: The Workflow Across Disciplines - *CC BY-NC 4.0*
- 6.10: Structure and Professional Accountability - *CC BY-NC 4.0*
- 6.11: Summary - *CC BY-NC 4.0*
- 6.12: End-of-Chapter Problem Set - *CC BY-NC 4.0*

◦ 7: The Engineering Design Process - *CC BY-NC 4.0*
- 7.1: Analysis vs Design - *CC BY-NC 4.0*
- 7.2: What Makes a Design Problem - *CC BY-NC 4.0*
- 7.3: Seven-Step Design Framework - *CC BY-NC 4.0*
- 7.4: Define the Design Objective - *CC BY-NC 4.0*
- 7.5: Identify Constraints - *CC BY-NC 4.0*
- 7.6: Generate Alternatives - *CC BY-NC 4.0*
- 7.7: Model and Analyze - *CC BY-NC 4.0*
- 7.8: Evaluate Trade-offs - *CC BY-NC 4.0*
- 7.9: Iterate and Refine - *CC BY-NC 4.0*
- 7.10: Communicate - *CC BY-NC 4.0*
- 7.11: Design, Ethics, and Safety Factors - *CC BY-NC 4.0*
- 7.12: Summary - *CC BY-NC 4.0*
- 7.13: Quick Check - *CC BY-NC 4.0*

◦ 8: Spreadsheet Modeling - *CC BY-NC 4.0*
- 8.1: From Calculator to Model - *CC BY-NC 4.0*
- 8.2: Structuring an Engineering Spreadsheet - *CC BY-NC 4.0*
- 8.3: Absolute vs Relative References - *CC BY-NC 4.0*
- 8.4: Parameter Sweeps - *CC BY-NC 4.0*
- 8.5: Graphical Visualization - *CC BY-NC 4.0*
- 8.6: Conditional Formatting and IF Logic - *CC BY-NC 4.0*
- 8.7: Six Common Spreadsheet Errors - *CC BY-NC 4.0*
- 8.8: Spreadsheets as Design Exploration Tools - *CC BY-NC 4.0*
- 8.9: When Spreadsheets Are Not Enough - *CC BY-NC 4.0*
- 8.10: Summary - *CC BY-NC 4.0*
- 8.11: End-of-Chapter Problem Set - *CC BY-NC 4.0*

◦ 9: Data Visualization for Engineers - *CC BY-NC 4.0*
- 9.1: Why Graphs Are an Engineering Tool - *CC BY-NC 4.0*
- 9.2: Chart Type Selection - *CC BY-NC 4.0*
- 9.3: Anatomy of an Engineering Graph - *CC BY-NC 4.0*
- 9.4: Linear vs Nonlinear Trends - *CC BY-NC 4.0*
- 9.5: Misleading Visualizations - *CC BY-NC 4.0*
- 9.6: Logarithmic Scales - *CC BY-NC 4.0*
- 9.7: Multi-Curve Comparison - *CC BY-NC 4.0*
- 9.8: Visualization and Interpretation - *CC BY-NC 4.0*
- 9.9: Advanced Chart Features - *CC BY-NC 4.0*
- 9.10: Graph Checklist — Quick Reference - *CC BY-NC 4.0*
- 9.11: Summary - *CC BY-NC 4.0*
- 9.12: Quick Check - *CC BY-NC 4.0*

◦ 10: MATLAB for Engineering Computation - *CC BY-NC 4.0*
- 10.1: Why Engineers Program - *CC BY-NC 4.0*
- 10.2: MATLAB as Structured Engineering - *CC BY-NC 4.0*
- 10.3: Script Basics - *CC BY-NC 4.0*
- 10.4: Four-Step Build - *CC BY-NC 4.0*
- 10.5: Trigonometry in MATLAB - *CC BY-NC 4.0*
- 10.6: Conditional Logic - *CC BY-NC 4.0*
- 10.7: Logical Indexing - *CC BY-NC 4.0*
- 10.8: Programming Project- Cable Car in MATLAB - *CC BY-NC 4.0*
- 10.9: Choosing the Right Tool - *CC BY-NC 4.0*
- 10.10: Summary - *CC BY-NC 4.0*
- 10.11: End-of-Chapter Problem Set - *CC BY-NC 4.0*

◦ 11: Circuit Fundamentals - *CC BY-NC 4.0*
- 11.1: Circuits as Energy Systems - *CC BY-NC 4.0*
- 11.2: Physical Meaning - *CC BY-NC 4.0*
- 11.3: Ohm's Law - *CC BY-NC 4.0*
- 11.4: Power - *CC BY-NC 4.0*
- 11.5: Schematic Symbols - *CC BY-NC 4.0*
- 11.6: Series Circuits - *CC BY-NC 4.0*
- 11.7: Parallel Circuits - *CC BY-NC 4.0*
- 11.8: Series vs Parallel Comparison - *CC BY-NC 4.0*
- 11.9: Kirchhoff's Laws - *CC BY-NC 4.0*

- 11.10: Power in Networks - *CC BY-NC 4.0*
- 11.11: Mixed Circuits - *CC BY-NC 4.0*
- 11.12: LED Design - *CC BY-NC 4.0*
- 11.13: Circuit Simulator - *CC BY-NC 4.0*
- 11.14: Summary - *CC BY-NC 4.0*
- 11.15: End-of-Chapter Problem Set - *CC BY-NC 4.0*

◦ 12: Parametric CAD and 3D Modeling - *CC BY-NC 4.0*
- 12.1: CAD Is Not Drawing - *CC BY-NC 4.0*
- 12.2: Parametric Modeling Core Concepts - *CC BY-NC 4.0*
- 12.3: From Sketch to Solid - *CC BY-NC 4.0*
- 12.4: The Sketch as a Mathematical Definition - *CC BY-NC 4.0*
- 12.5: The Fusion Project - *CC BY-NC 4.0*
- 12.6: Modeling Workflow - *CC BY-NC 4.0*
- 12.7: Five Common Modeling Errors - *CC BY-NC 4.0*
- 12.8: CAD in the Engineering Landscape - *CC BY-NC 4.0*
- 12.9: Summary - *CC BY-NC 4.0*
- 12.10: End-of-Chapter Problems - *CC BY-NC 4.0*

◦ 13: Engineering Communication - *CC BY-NC 4.0*
- 13.1: Communication Is Part of the Deliverable - *CC BY-NC 4.0*
- 13.2: Three Modes of Communication - *CC BY-NC 4.0*
- 13.3: The Technical Memo - *CC BY-NC 4.0*
- 13.4: Complete Memo Example - *CC BY-NC 4.0*
- 13.5: Quantitative Precision - *CC BY-NC 4.0*
- 13.6: Graphs and Tables - *CC BY-NC 4.0*
- 13.7: Technical Presentations - *CC BY-NC 4.0*
- 13.8: TEGO Case Study - *CC BY-NC 4.0*
- 13.9: Professional Email - *CC BY-NC 4.0*
- 13.10: Communicating Uncertainty and Limitations - *CC BY-NC 4.0*
- 13.11: Summary - *CC BY-NC 4.0*
- 13.12: End-of-Chapter Applied Exercises - *CC BY-NC 4.0*

◦ 14: Engineering Ethics - *CC BY-NC 4.0*
- 14.1: Ethics Is Not an Add-On - *CC BY-NC 4.0*
- 14.2: NCEES Model Rules - *CC BY-NC 4.0*
- 14.3: Ethics HW Preview Table - *CC BY-NC 4.0*
- 14.4: Safety Factors - *CC BY-NC 4.0*
- 14.5: Modeling Assumptions - *CC BY-NC 4.0*
- 14.6: Eight-Step Framework - *CC BY-NC 4.0*
- 14.7: Ethics HW Scenario 1 - *CC BY-NC 4.0*
- 14.8: Ethics HW Scenario 2 - *CC BY-NC 4.0*
- 14.9: Ethics HW Scenario 3 - *CC BY-NC 4.0*
- 14.10: Academic Integrity - *CC BY-NC 4.0*
- 14.11: Citicorp Case Study - *CC BY-NC 4.0*
- 14.12: Environmental Responsibility - *CC BY-NC 4.0*
- 14.13: Summary - *CC BY-NC 4.0*
- 14.14: Quick Check - *CC BY-NC 4.0*

◦ 15: Integrated Engineering Design Challenge - *CC BY-NC 4.0*
- 15.1: This Is Not a New Topic - *CC BY-NC 4.0*
- 15.2: Challenge Specifications - *CC BY-NC 4.0*
- 15.3: Define- Units, Loads, What "Best" Means - *CC BY-NC 4.0*
- 15.4: Model- Tension, Compression, Straws - *CC BY-NC 4.0*
- 15.5: Mass Budget - *CC BY-NC 4.0*
- 15.6: Parameter Sweep - *CC BY-NC 4.0*
- 15.7: Alternatives and Decision Matrix - *CC BY-NC 4.0*
- 15.8: Iterate - *CC BY-NC 4.0*
- 15.9: Bridge Report - *CC BY-NC 4.0*
- 15.10: Course Map - *CC BY-NC 4.0*
- 15.11: Summary - *CC BY-NC 4.0*
- 15.12: End-of-Chapter Problem Set - *CC BY-NC 4.0*

◦ 16: Appendices - *CC BY-NC 4.0*
- 16.1: Appendix A — Standard Symbols and Units - *CC BY-NC 4.0*
- 16.2: Appendix B — Unit Conversion Reference Table - *CC BY-NC 4.0*
- 16.3: Appendix C — MATLAB Quick Reference - *CC BY-NC 4.0*
- 16.4: Appendix D — Spreadsheet Quick Reference (Excel) - *CC BY-NC 4.0*
- 16.5: Appendix E — Technical Memo Template - *CC BY-NC 4.0*

◦ Back Matter - *Undeclared*
- Index - *Undeclared*
- Glossary - *Undeclared*
- Detailed Licensing - *Undeclared*
- Detailed Licensing - *Undeclared*

This page titled Introduction to Engineering - Thinking Like an Engineer is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by Jonathan Compton.