Design for Electrical and Computer Engineers

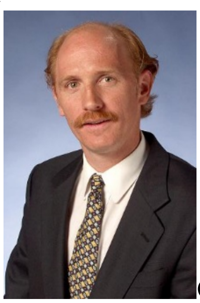
Theory, Concepts, and Practice

Ralph M. Ford and Christopher S. Coulston

This document was prepared with LaTeX.

0.1 About the Authors

Ralph Ford obtained his Ph.D. and M.S. degrees in Electrical Engineering from the University of Arizona in 1994 and 1989 respectively. He obtained his B.S. in Electrical Engineering from Clarkson University in 1987. He worked for the IBM Microelectronics Division in East Fishkill, NY from 1989-1991, where he developed machine vision systems to inspect electronic packaging modules for mainframe computers. Ralph also has experience working for IBM Data Systems and the Brookhaven National Laboratory. He joined the faculty at Penn State Erie, The Behrend College in 1994. Ralph has experience teaching electronics and software design, as well as teaching the capstone design course sequence in the electrical, computer, and software engineering programs. His research interests are in engineering design, image processing, machine vision, and signal processing. Ralph is currently Director of the School of Engineering at Penn State Behrend. He also serves as a program evaluator for ABET. He was awarded a Fulbright Scholarship to study at the Brno University of Technology in the Czech Republic in 2005.

Chris Coulston obtained his Ph.D. in Computer Science and Engineering from the Pennsylvania State University in 1999. He obtained his M.S. and B.S in Computer Engineering from the Pennsylvania State University in 1994 and 1992 respectively. Chris has industry experience working for IBM in Manassas, VA and Accu-Weather in State College, PA. He joined the faculty at Penn State Erie, The Behrend College in 1999. He has experience teaching design-oriented courses in digital systems, embedded systems, computer architecture, and database management systems.

Chris' research interests are in Steiner tree routing algorithms and artificial life. He is currently an Associate Professor of Electrical and Computer at Penn State Behrend and also serves as Chairperson of the program.

Contents

0.2 Preface

This book is written for undergraduate students and teachers engaged in electrical and computer engineering (ECE) design projects, primarily in the senior year. The objective of the text is to provide a treatment of the design process in ECE with a sound academic basis that is integrated with practical application. This combination is necessary in design projects because students are expected to apply their theoretical knowledge to bring useful systems to reality. This topical integration is reflected in the subtitle of the book: Theory, Concepts, and Practice. Fundamental theories are developed whenever possible, such as in the chapters on functional design decomposition, system behavior, and design for reliability. Many aspects of the design process are based upon time-tested concepts that represent the generalization of successful practices and experience. These concepts are embodied in processes presented in the book, for example, in the chapters on needs identification and requirements development. Regardless of the topic, the goal is to apply the material to practical problems and design projects. Overall, we believe that this text is unique in providing a comprehensive design treatment for ECE, something that is sorely missing in the field. We hope that it will fill an important need as capstone design projects continue to grow in importance in engineering education.

We have found that there are three important pieces to completing a successful design project. The first is an understanding of the design process, the second is an understanding of how to apply technical design tools, and the third is successful application of professional skills. Design teams that effectively synthesize all three tend to be far more successful than those that don't. The book is organized into three parts that support each of these areas.

The first part of the book, the *Design Process*, embodies the steps required to take an idea from concept to successful design. At first, many students consider the design process to be obvious. Yet it is clear that failure to understand and follow a structured design process often leads to problems in development, if not outright failure. The design process is a theme that is woven throughout the text; however, its main emphasis is placed in the first four chapters. Chapter 1 is an introduction to design processes in different ECE application domains. Chapter 2 provides guidance on how to select projects and assess the needs of the customer or user. Depending upon how the design experience is structured, both students and faculty may be faced with the task of selecting the project concept. Further, one of the important issues in the engineering design is to understand that systems are developed for use by an end-user, and if not designed to properly meet that need, they will likely fail. Chapter 3 explains how to develop the Requirements Specification along with methods for developing and documenting the requirements. Practical examples are provided to illustrate these methods and techniques. Chapter 4 presents concept generation and evaluation. A hallmark of design is that there are many potential solutions to the problem. Designers need to creatively explore the space of possible solutions and apply judgment to select the best one from the competing alternatives.

The second part of the book, *Design Tools*, presents important technical tools that ECE designers often draw upon. Chapter 5 emphasizes system engineering concepts including the well known functional decomposition design technique and applications in a number of ECE problem domains. Chapter 6 provides methods for describing system behavior, such as flowcharts, state diagrams, data flow diagrams and a brief overview of the Unified Modeling Language (UML). Chapter 7 covers important issues in testing and provides different viewpoints on testing throughout the development cycle. Chapter 8 addresses reliability theory in design, and reliability at both the component and system level is considered.

The third part of the book focuses on *Professional Skills*. Designing, building, and testing a system is a process that challenges the best teams, and requires good communication and project management skills. Chapter 9 provides guidance for effective teamwork. It provides an overview of pertinent research on teaming and distills it into a set of heuristics. Chapter 10 presents traditional elements of project planning, such as the work breakdown structure, network diagrams, and critical path estimation. It also addresses how to estimate manpower needs for a design project. Chapter 11 addresses ethical considerations in both system design and professional practice. Case studies for ECE scenarios are examined and analyzed using the IEEE (Institute of Electrical and Electronics Engineers) Code of Ethics as a basis. The book concludes with Chapter 12, which contains guidance for students preparing for oral presentations, often a part of capstone design projects.

Features of the Book

This book aims to guide students and faculty through the steps necessary for the successful execution of design projects. Some of the features are listed below.

- Each chapter provides a brief motivation for the material in the chapter followed by specific learning objectives.

- There are many examples throughout the book that demonstrate the application of the material.
- Each end-of-chapter problem has a different intention. Review problems demonstrate comprehension of the material in the chapter. Application problems require the solution of problems based upon the material learned in the chapter. Design problems are directly applicable to design projects and are usually tied in with the Project Application section.
- Nearly all chapters contain a Project Application section that describes how to apply the material to a design project.
- Some chapters contain a Guidance section that represents the author's advice on application of the material to a design project.
- Checklists are provided for helping students assess their work.
- There are many terms used in design whose meaning needs to be understood. The text contains a glossary with definitions of design terminology. The terms defined in the glossary (Appendix A) are indicated by ***italicized-bold*** highlighting in the text.
- All chapters conclude with a Summary and Further Reading section. The aim of the Further Reading portion is to provide pointers for those who want to delve deeper into the material presented.
- The book is structured to help programs demonstrate that they are meeting the ABET (accreditation board for engineering programs) accreditation criteria. It provides examples of how to address constraints and standards that must be considered in design projects. Furthermore, many of the professional skills topics, such as teamwork, ethics, and oral presentation ability, are directly related to the ABET Educational Outcomes. The requirements development methods presented in Chapter 3 are valuable tools for helping students perform on cross-functional teams where they must communicate with non-engineers.
- An instructor's manual is available that contains not only solutions, but guidance from the authors on teaching the material and managing student design teams. It is particularly important to provide advice to instructors since teaching design has unique challenges that are different than teaching engineering science oriented courses that most faculty are familiar with.

- PowerPoint^{TM} presentations are available for instructors through McGraw-Hill

- There are a number of complete case study student projects available in electronic form for download by both students and instructors and available at. These projects have been developed using the processes provided in this book.

How to Use this Book

There are several common models for teaching capstone design, and this book has the flexibility to serve different needs. Particularly, chapters from the Professional Skills section can be inserted as appropriate throughout the course. Recommended usage of the book for three different models of teaching a capstone design course is presented.

- **Model I.** This is a two-semester course sequence. In the first semester, students learn about design principles and start their capstone projects. This is the model that we follow. In the first semester the material in the book is covered in its entirety. The order of coverage is typically Chapters 1–3, 9, 4–6, 10–11, and 7–8. Chapter 9 (Teams and Teamwork) is covered immediately after the projects are identified and the teams are formed. Chapters 10 (Project Management) and 11 (Ethical and Legal Issues) are covered after the system design techniques in Chapters 5 and 6 are presented. Students are in a good position to create a project plan and address ethical issues in their designs after learning the more technical aspects of design. Chapter 12 (Oral Presentations) is assigned to students to read before their first oral presentation to the faculty. The course concludes with principles of testing and system reliability (Chapter 7 and 8). We assign a good number of end-of-chapter problems and have quizzes throughout the semester. By the end of the first semester, design teams are expected to have completed development of the requirements, the high-level or architectural design, and developed a project plan. In the second semester, student teams implement and test their designs under the guidance of a faculty advisor.

- **Model II**. This two-semester course sequence is similar to Model I with the difference being that the first semester is a lower credit course (often one credit) taught in a seminar format. In this model chapters can be selected to support the projects. Some of the core chapters for consideration are Chapters 1–5, which take the student from project selection to functional design, and Chapters 9–11 on teamwork, project management, and ethical issues. Other chapters could be covered at the instructor's discretion. The use of end-of-chapter problems would be limited, but the project application sections and example problems in the text would be useful in guiding students through their projects.

- **Model III**. This is a one-semester design sequence. Here, the book would be used to guide students through the design process. Chapters for consideration are 1–5 and 9–10, which provide the basics of design, teamwork, and project management. The project application sections and problems could be used as guidance for the project teams.

Acknowledgements

Undertaking this work has been a challenging experience and could not have been done without the support of many others. First, we thank our families for their support and patience. They have endured many hours and late evenings that we spent researching and writing. Melanie Ford is to be thanked for her diligent proofreading efforts. Bob Simoneau, the former Director of the Penn State Behrend School of Engineering, has been a great supporter of the book and has also lent his time in reading and providing comments. Our school has a strong design culture, and this book would not have happened without that emphasis; our faculty colleagues need to be recognized for developing that culture. Jana Goodrich and Rob Weissbach are two faculty members with whom we have collaborated on other courses and projects. They have influenced our thinking in this book, particularly in regard to project selection, requirements development, cost estimation, and teamwork. We must also recognize the great collaborative working environment that exists at Penn State Behrend, which has allowed this work to flourish. Our students have been patient in allowing us to experiment with different material in the class and on the projects. Examples of their work are included in the book and are greatly appreciated. John Wallberg contributed the disk drive diagnostics case study in Chapter 11 that we have found very useful for in-class discussions. John developed this while he was a student at MIT. Thanks to Anne Maloney for her copyediting of the manuscript. The following individuals at McGraw-Hill have been very supportive and we thank them for their efforts to make this book a reality – Carlise Stembridge, Julie Kehrwald, Darlene Schueller, Craig Marty, Kris Tibbetts, and Mike Hackett.

Finally, we would like to thank the external reviewers of the book for their thorough reviews and valuable ideas. They are Frederick C. Berry

(Rose-Hulman Institute of Technology), Mike Bright (Grove City College), Geoffrey Brooks (Florida State University Panama City Campus) Wils L. Cooley (West Virginia University), D. J. Godfrey (US Coast Guard Academy), and Michael Ruane (Boston University).

We hope that you find this book valuable, and that it motivates you to create great designs. We welcome your comments and input. Please feel free to email us.

Ralph M. Ford,

Chris S. Coulston,

Chapter 1

The Engineering Design Process

1.1 The Engineering Design Process

> **en-gi-neer (n)** *1. One versed in the design, construction, and use of machines. 2. One who employs the innovative and methodical application of scientific knowledge and technology to produce a device, system, or process, which is intended to satisfy human needs.* —American College Dictionary

Take a moment to read and analyze the key elements of the two definitions presented above. If you are an engineering student or practicing engineer, do you think that this definition applies to you? The first definition uses the terms *design* and *construction*. People like to think of themselves as designers. Why is that so? The answer may be in the combination of the term *construction,* and from the second definition, the idea of *innovation*. Applying innovation and creativity to produce something new is a wonderfully rewarding process. The great thing about being an engineer is that it allows you to be a creative designer. That is generally not the way the profession is viewed. What is the difference between engineering design and other types of design that are associated with creativity such as interior design, fashion design, or webpage design? The answer is supplied in the second definition which states *"...methodical application of scientific knowledge and technology...*" As an engineering student, you have studied a great deal of math, science, and fundamental technology, but probably have had limited exposure to creative and innovative design.

The definition also contains the somewhat contradictory terms *innovative* and *methodical*. If there is an established and methodical way of employing a scientific principle or process, it does not seem to allow much room for creativity and innovation. The truth is that the two concepts are in competition with each other, but a good engineer realizes this and utilizes both effectively. The definition also indicates that engineers design to satisfy human needs, an important, yet often overlooked point. That means that when designing systems, it is necessary to determine the user's needs and the ethical application of the technology.

This book aims to help electrical and computer engineers become effective designers, to better understand professional practices, and to provide guidance for executing design projects. This chapter presents the processes by which designs are realized, the characteristics of successful engineers, and an overview of the book.

Learning Objectives

By the end of this chapter, the reader should:

- Understand what is meant by engineering design.
- Understand the phases of the engineering design process.
- Be familiar with the attributes of successful engineers.
- Understand the objectives of this book.

1.2 The Engineering Design Process

ABET (formerly known as Accreditation Board of Engineering and Technology) provides the following definition of engineering design [ABE03].

> Engineering design is the process of devising a system, component, or process to meet desired needs. It is a decision-making process (often iterative), in which the basic sciences, mathematics, and engineering sciences are applied to convert resources optimally to meet a stated objective. Among the fundamental elements of the design process are the establishment of objectives and criteria, synthesis, analysis, construction, testing, and evaluation.

The definition indicates that, in engineering design, different phases of the process have to be re-visited and the deliverables for each phase updated as necessary. Realistic problems are complex with many potential solutions; the goal is not to find just any solution, but the best one given the constraints and available resources. This requires the application of sound judgment, decision-making skills, and patience in constantly evaluating progress towards a solution. The definition identifies some common elements of the design process, such as establishment of criteria, synthesis, construction, and testing.

Design processes embody the steps required to take an idea from concept to realization of the final system, and are problem-solving methodologies that aim to develop a system that best meets the customer's need within given constraints. This is not all that different from some everyday processes, such as preparing dinner. Say you are hungry and need to eat dinner before you can go to see a movie that starts in one hour. The constraints are time, money, food, your tastes, and nutritional value if you are health-conscious. You brainstorm and come up with the options of making dinner at home, going to a restaurant, or buying something to eat at the theater. Based on these options, you then select the solution based on your evaluation of the best one. This is similar in philosophy to the stages of design processes where you have a problem to solve, constraints, and a number of potential solutions to select from.

A related term is known as the *product realization process.* The product realization process is broader in scope, including aspects such as entrepreneurship, market research, financial planning, product pricing, and market strategy. Many technologies have their own particular design processes that have evolved over time and have been found by practitioners in the field to be valuable. For example, different methodologies are applied in the design of integrated circuits (VLSI), embedded systems, and software systems, yet they all have some degree of commonality, such as requirements analysis, technical design, and system test. Design processes continue to evolve. One field in which this is particularly true is in software design due to the constantly changing nature of software and the special challenges that large software projects pose.

Cross [Cro00] identified two types of design processes—prescriptive and descriptive. As the name implies, ***prescriptive design processes*** set down an exact process, or systematic recipe, for realizing a system. Prescriptive design processes are often algorithmic in nature and expressed using flow charts with decision logic. An example of a prescriptive process is shown in Figure 1.1, which describes the front end of the design process where the problem and requirements are determined. A decision block is included where the requirements are examined to determine if they satisfy the needs of the problem. ***Descriptive processes*** are less formal, describing typical

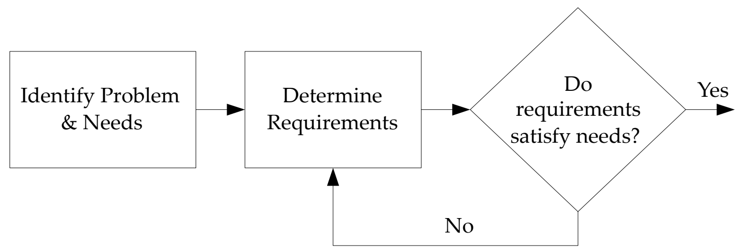


Figure 1.1: A prescriptive design process for problem identification and requirements selection.

activities involved in realizing designs with less emphasis on exact sequencing. The distinction between descriptive and prescriptive processes is not always clear, however, and some processes may be considered more strongly associated with one property than the other. Cross makes an important point in stating that design processes are sometimes viewed as common sense and thus ignored, resulting in failed products. Cross cites two good reasons to adhere to design processes: 1) they formalize thought processes to ensure good practices are followed, leading to better and more innovative solutions, and 2) they keep all members of the team synchronized in terms of understanding where they are in the design process.

A descriptive process that is widely applicable to design problems is shown in Figure 1.2. In a perfect world, the process starts with the identification of the problem, proceeds clockwise to research, followed the requirements phase, and so on until the system or device is delivered and goes into service (maintenance phase). This scenario is unrealistic, ignoring the iterative nature of design where the design team alternates between different phases as necessary. Consequently, links are inserted that allow transitions between all the different phases of the

design process. Of course, transitions between certain phases are unreasonable or very costly. It is virtually impossible to move directly from problem identification to system integration without developing a design concept first. It is much more likely for engineers to alternate between nearby phases in the process, such as problem identification, research, requirements specification, and concept generation. This does not mean that you can't move between phases that are not in close proximity in the model. For instance, the customer's needs may change while in the design phase, necessitating re-evaluation of the needs, correction of the requirements specification, and system redesign—all at a substantial cost in time and money. Studies have

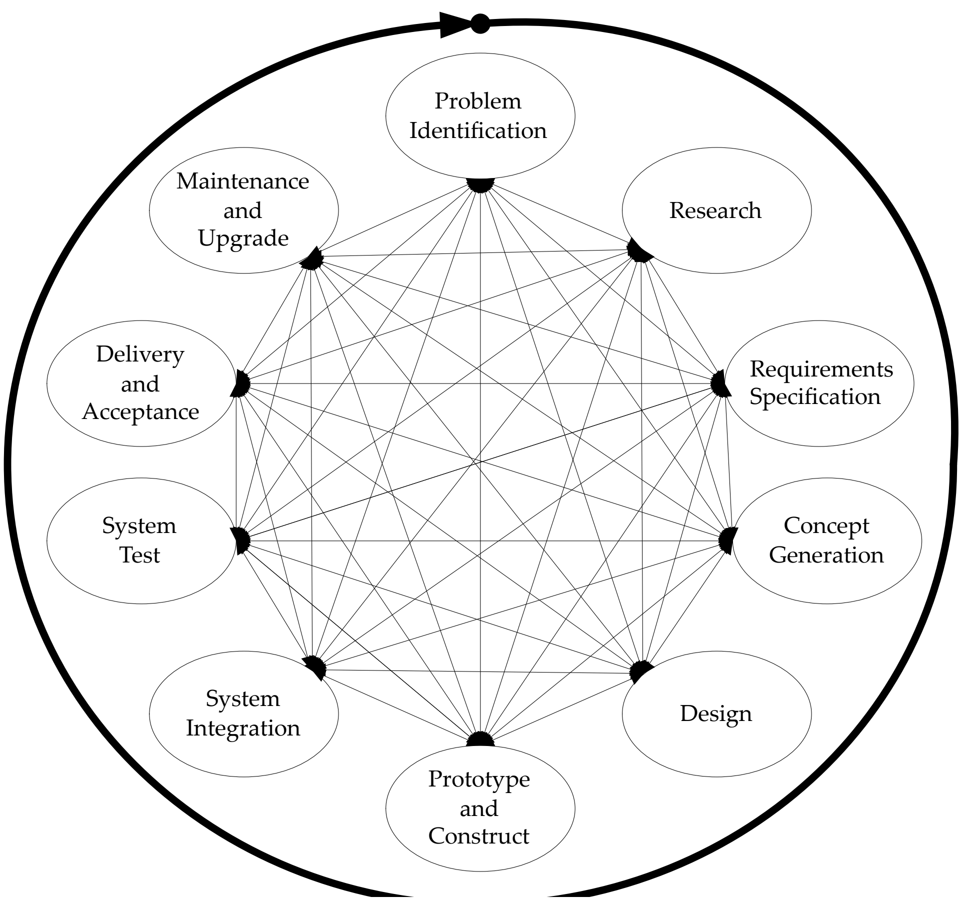


Figure 1.2: A descriptive overview of the design process.

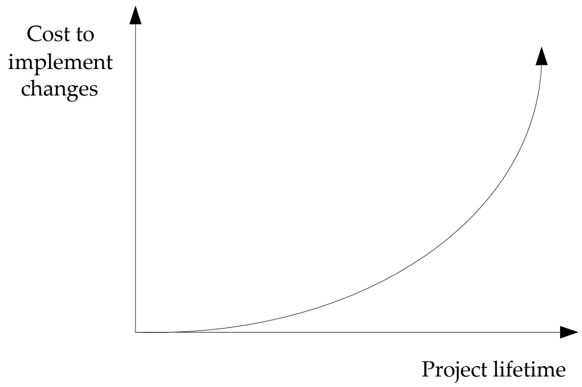


Figure 1.3: The cost to implement design changes increases exponentially with project lifetime.

shown that the cost required to correct errors or make changes increases exponentially as the project lifetime increases, as presented in Figure 1.3.

1.2.1 Elements of the Design Process

Nearly all the phases of the design process in Figure 1.2 are covered in this book, with the exception of the maintenance phase. The objective of the first phase, ***problem identification***, is to identify the problem and customer needs. This occurs in a variety of ways, from someone conceiving a new idea to a client coming to you with a problem to solve. In either case, it is important to determine the true needs for the product, device, or system (terms that are used interchangeably throughout the book and often referred to as systems). Failure to correctly identify the needs has negative ramifications for the entire process, typically resulting in costly redesigns, or even worse, abandonment of the project.

In the ***research phase*** the design team conducts research on the basic engineering and scientific principles, related technologies, and existing solutions. The objective is to become experts on the problem, save time and money by not re-inventing the wheel, and be positioned to develop new and innovative solutions.

The ***Requirements Specification*** articulates what the system must do for it to be successful and to be accepted by the customer. It is important to focus on what the system must do, as opposed to how the solution will be implemented. This is challenging since engineers tend to focus on solutions and propose implementations early in the process. This is not surprising since engineering education focuses on solving problems rather than specifying them. The requirements are the mission statement that guides the entire project, and if properly developed, provide flexibility for creativity and innovation in developing solutions.

In ***concept generation,*** many possible solutions to the problem are developed. The hallmark of design is that it is open-ended, meaning that there are multiple solutions to the problem and the objective is to develop the one that best meets the requirements and satisfies the constraints. In this phase, wild creativity is encouraged, but it is ultimately tempered with critical evaluation of the competing alternatives.

In the ***design phase,*** the team iteratively develops a technical solution, ultimately producing a detailed system design. Upon its completion, all major systems and subsystems are identified and described using an appropriate model that depends upon the particular technology being employed.

In the ***prototyping and construction phase,*** different elements of the system are constructed and tested. In rapid prototyping, the objective is to model some aspect of the system, demonstrating functionality to be employed in the final realization. Many prototypes are discarded or modified as the system evolves—the idea is to experiment, demonstrate proof-of-concept principles, and improve understanding. Prototypes may be used anywhere in the process—you may present the client with prototypes after the concept generation phase, or they may be utilized in the design phase to test a design idea, or as the final system is tested and developed.

During ***system integration,*** all of the subsystems are brought together to produce a complete working system. This phase is challenging and time-consuming since many different pieces of the design must be interfaced, and the team must work closely to make it all work. Care taken in the design phase to clearly communicate the functionality and interfaces between subsystems aids in system integration. System integration is closely tied to the ***test phase***, where the overall system is tested to demonstrate that it meets the requirements.

Ultimately the system is *delivered* to the customer where it is likely that they will test it using a mutually agreed upon process. Development does not necessarily end when the system goes into service, as it will likely enter the ***maintenance phase*** where it is maintained, upgraded to add new functionality, or where design problems are corrected. Following and understanding the design process improves the probability of successful system development. The process is flexible, and the designer needs to transition between different phases in order to bring the system to realization. Design is an iterative process—you may not fully understand everything necessary

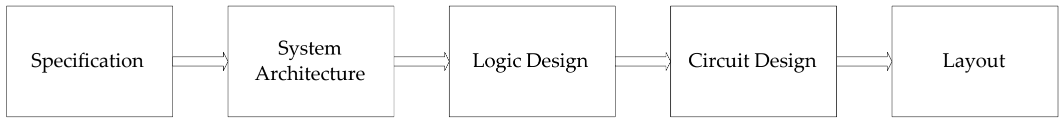


Figure 1.4: A process for integrated circuit (VLSI) design [Wol02].

in any given phase and have to revisit different steps as the system evolves. That is not a license for not trying to develop the best design you can on the first attempt—by all means do so—but realize that flexibility and a willingness to change the design are necessary.

1.2.2 Technology Specific Design Processes

Different application domains have developed specialized processes for technology-specific design. One such example is VLSI (Very Large Scale Integration) design. A typical VLSI design process is shown in Figure 1.4 [Wol02]. In this model the system specification is used to develop the system architecture. The system architecture is composed of the major functional units that constitute an integrated circuit. Each functional unit is then designed at the gate logic level, which is subsequently designed at the circuit (transistor) level, and finally the circuit elements are laid out on the silicon chip. This is an excellent demonstration of the divide-and-conquer approach to design, where a complex system is broken down into lower levels of abstraction and each of these is further broken down until the design objectives are met.

Next, consider the design process for embedded computer systems shown in Figure 1.5. Embedded systems are combined hardware/software systems embedded into a larger system to perform dedicated application specific operations. Embedded systems are employed in automobiles, DVD players, and digital cameras to name a few applications. Performance issues dominate embedded applications, and the designer needs to partition tasks between software and hardware to achieve optimum performance. This design process is somewhat prescriptive with phases for requirements gathering, specifications, and architectural design. The process reflects the unique nature of embedded systems with separate software and hardware design blocks, married together by the interface design.

The field of software engineering is one in which the development of different design process models is still under considerable flux today. This is due to the complex nature of software and the failure of computer scientists and engineers to effectively develop high-quality software systems. There

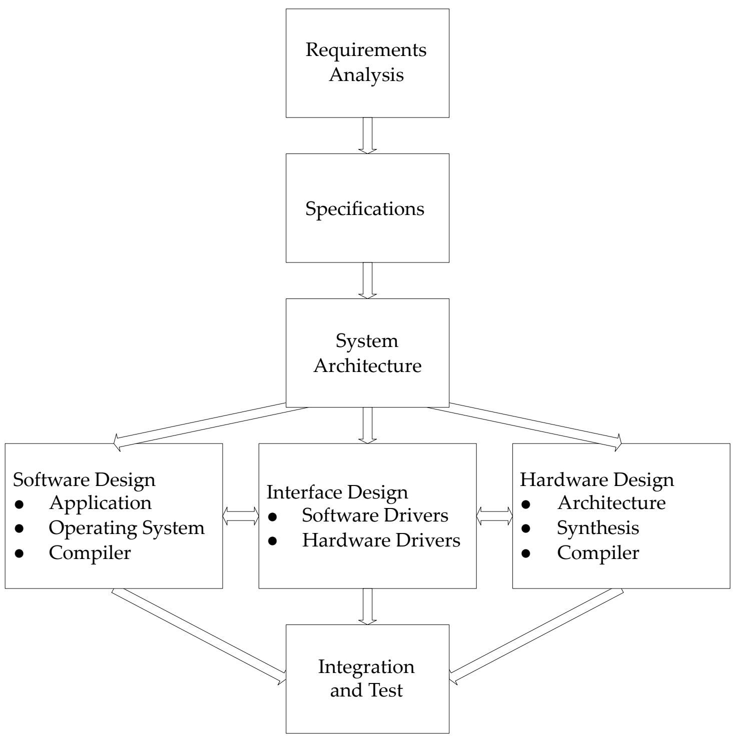


Figure 1.5: An embedded system design process [Ern97].

are many reasons why this is so. The sheer size of software programs may easily exceed one million lines of code written by many different software developers. One small mistake in those millions of lines of code can cause the system to fail. Another difficulty is in designing for upgrade and reuse of software. What if the needs change after the millions of lines of code are developed and one of the fundamental structures or objects needs to be upgraded?

The *waterfall model* shown in Figure 1.6 is one of the first proposed and most well-known software design processes. This is a prescriptive model since the development proceeds linearly from the first step where the user's needs are analyzed through the phases of specification development, design, test, and maintenance. This works for well-defined and moderately complex software applications, but fails as complexity grows due to the inability to move between phases. A more flexible and descriptive software design process is known as the *spiral model*, which is a cyclical process where phases are revisited as necessary [Som01]. *Extreme Programming* is a more recent and controversial software development process, where relatively small teams of software developers rapidly develop software following some strict rules. Both the spiral model and Extreme Programming are examined in more detail in the end of chapter problems.

1.3 The World-Class Engineer

The ability to effectively design is important for engineers, requiring strong technical skills and an understanding of the design process. Yet, this ability in itself is not enough to become an effective practicing engineer. The Pennsylvania State University Leonhard Center for the Advancement of Engineering Education, in consultation with a number of industries, developed a description of what is referred to as a "World-Class Engineer" [Leo95]. Shown in Table 1.1, the description identifies the characteristics of successful engineers, and contains six major elements: 1) Aware of the World, 2) Solidly Grounded, 3) Technically Broad, 4) Effective in Group Operations, 5) Versatile, and 6) Customer-Oriented. The description recognizes that engineers must be effective in group operations, since the majority of projects are carried out in teams. Not only that, many projects span multiple technical disciplines and are executed in multifunctional organizations that have diverse groups such as marketing, finance, human resources, technical support, and service. It also recognizes that an engineer must be versatile, innovative, understand ethical principles, and be customer-oriented, impor-

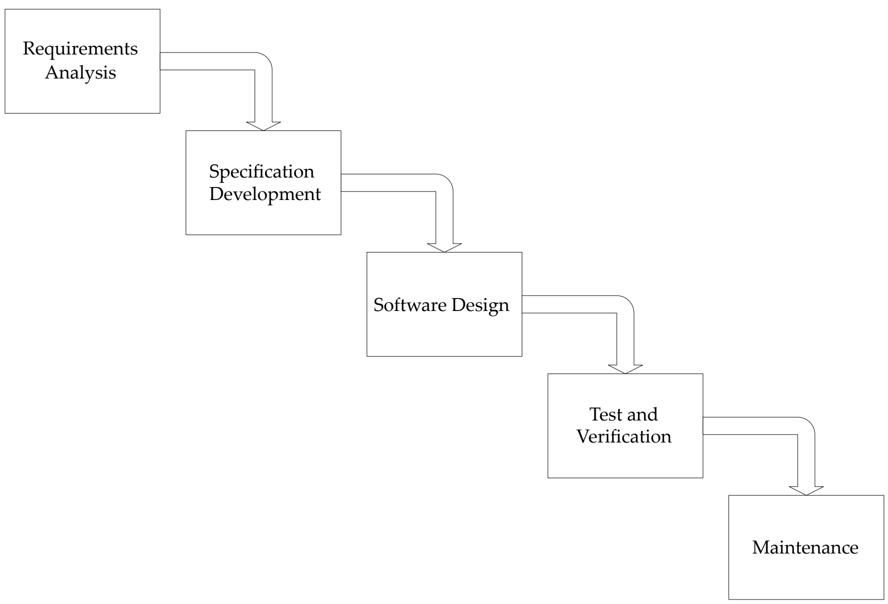


Figure 1.6: Waterfall software development process. In this model, development proceeds linearly from requirements analysis, through each subsequent phase, terminating with maintenance.

tant themes that are stressed throughout this book.

1.4 Book Overview

Consider the digital camera, the cellular phone, and the space shuttle, all complex systems that integrate a variety of technologies. A digital camera is the synthesis of an embedded electronics system, optics, a mechanical lens assembly, and the camera package itself. The embedded electronics contain an imaging sensor, a digital display, digital interface circuitry, flash memory storage, system control software, and the user interface. The challenges of integrating the components of such a system and having it record and transfer huge amounts of image data, within an acceptable timeframe, are immense. Cellular phones are another good example of a complex system that represents a technology that has shrunk in size, but increased tremendously in functionality at the same time. They encompass digital data communications, antenna design, encryption for secure data transmission, a user interface display, and Internet connectivity. At the other end of the spectrum are large-scale space and military systems, such as the space shuttle. Despite the two shuttle accidents, the safety and reliability requirements of the space shuttle are incredibly high. Realizing such a system is accomplished by a tremendous number of people from many disciplines working for different organizations. All three of these technologies were developed by large teams that encompass multiple disciplines. The processes and practices employed in their development represent application of the fundamentals that this book hopes to cover. While you won't be building complete space shuttles by the end of this design course, you can expect to apply design principles that allow you to design and integrate a relatively complex system, maybe even a part of the space shuttle.

Table 1.1: The World-Class Engineer (Copyright the Leonhard Center for the Advancement of Engineering Education, The Pennsylvania State University. Reprinted by permission.)

I) Aware of the World

- sensitive to cultural differences, environmental concerns, and ethical principles
- alert to market opportunities (both high- and low-tech)
- cognizant of competitive talents, work ethic, and motivation

II) Solidly Grounded

- thoroughly trained in the fundamentals of a selected engineering discipline
- has a historical perspective and remains aware of advances in science that can impact engineering
- realizes that knowledge doubles at breakneck speed and is prepared to continue learning throughout a career

III) Technically Broad

- understands that real-life problems are multidisciplinary
- thinks broadly, seeing an issue in a rich context of various alternatives, probabilities, etc., rather than a narro quest to find a single answer
- is conversant in several disciplines
- is trained in systems modeling and the identification of critical elements. Understands the need to design exper ments to verify or extend analysis, as well as meet specification requirements
- is psychologically prepared to embrace any field necessary to solve the problem at hand

IV) Effective in Group Operations

- cooperative in an organization of individuals working toward a common creative goal that is often multidisciplinar and multifunctional in nature
- effective in written and oral communication
- willing to seek and use expert advice
- cognizant of the value of time and the need to make efficient use of the time in all phases of an endeavor
- understanding and respectful of the many facets of business operation -- general management, marketing, financ law, human resources, manufacturing, service, and especially quality

V) Versatile

- innovative in the development of products and services
- sees engineering as applicable to problem solving in general
- considers applying engineering beyond the typical employment focus of engineering graduates in the manufacturin industries, to the much broader economy (financial services, health care, transportation, etc.) where engineerin skills could make a dramatic improvement in the productivity of those segments of the economy that employ 8 percent of the U.S. population

VI) Customer Oriented

- realizes that finding and satisfying customers is the only guarantee of business success
- understands that products and services must excel in the test of cost-effectiveness in the global marketplace

The intention is to teach the application of design principles to computer

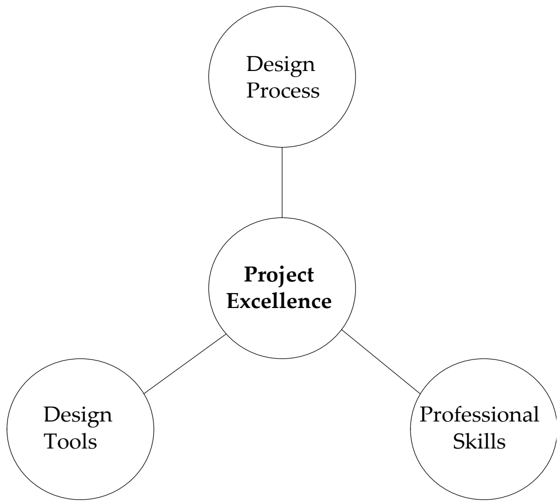


Figure 1.7: The guiding philosophy of this book. In order to achieve success in executing engineering and design projects, it takes an understanding of the design process, strong technical design tools, and professional skills.

and electrical engineers and to help prepare students for a professional career. The majority of engineering education is devoted to math, science, engineering science, and problem-solving. They are important topics required to enter this highly technical field. However, it is clear that there are other aspects beyond this that are equally important for success, including an understanding of system design, innovation, ethical principles, teamwork, and strong communication skills.

The book is divided into three parts: I–Design Process, II–Design Tools, and III–Professional Skills. This is shown in Figure 1.7 as three separate, but related components that play a key role in achieving Project Excellence—the ability to complete a project, in an ethical manner that meets the customer's need, satisfies the constraints, and is clearly communicated to all involved. The chapters are decoupled as much as possible so that the reader can move between chapters as necessary. In Part I, the emphasis is on understanding and gaining experience in the different phases of the design process. The reader is guided through the steps of project identification, research, specification development, creative concept generation, and critical evaluation of competing solutions. Part II addresses topics that are often employed in design, including functional decomposition, description of system behavior, reliability, and testing. Part III addresses professional skills, including teamwork principles, project planning, ethics in design and the profession, and oral communication skills.

Here are a few thoughts to conclude the chapter and get started on the path to great designs. You are embarking on what will likely be a fun, challenging, sometimes frustrating, and ultimately rewarding journey. The systems that engineers work with continue to become increasingly complex and multi-disciplinary in nature. The example problems presented in this book come from the fields of analog electronics, digital electronics, electrical systems theory, and software systems. These four areas comprise a significant problem-domain common to the education of most electrical and computer engineers. Finally, consider the quote below by Robert Hayes on the importance of design.

Fifteen years ago, companies competed on price. Today it's quality. Tomorrow it's design.—Robert H. Hayes, Harvard Business School, 1991.

What is this saying? Well, it is clear that the world continues to move to a more knowledge-based society, where individuals and companies compete on the strength of their intellectual capital and ability to produce new and innovative products. That is what design is all about. It is not saying that price and quality are unimportant, they certainly are; in fact quality and reliability in design are part of this book. It is that quality and price are a given, and successful products will be distinguished by their design characteristics. The implication is that design will play a larger role in the development and success of products. The future Hayes predicted is now. Design is what distinguishes between products that are seen as commodities and those that are truly unique and profitable.

1.5 Summary and Further Reading

Engineering design is an iterative process in which the design team employs creativity and technical knowledge to develop a solution that best meets the end-users' needs within the constraints applied to the problem. There is no single design process that can be applied to all situations and technologies, but there are many common elements shared, regardless of the technology under consideration. In order to successfully bring designs to fruition, it takes a combination of design tools, professional skills, and a clear understanding of the process needed to complete designs. The objective of this book is to develop your proficiency in these areas so that you may become an effective engineer and achieve excellence in design projects.

<u>Engineering Design Methods</u> by Nigel Cross [Cro00] presents the differ-

ences between descriptive and prescriptive design processes, and covers a wide array of processes in more detail. It also discusses the cognitive characteristics of effective designers. There are many good books on software engineering process development methods. Software Engineering by Ian Sommerville [Som01] discusses the different software design process models, such as the waterfall and spiral models. This is also true of many modern software engineering texts. The original reference to the waterfall model is by Royce [Roy70]. The Art of Innovation by Michael Kelley [Kel01] describes the activities of well-known design company IDEO and is a highly readable description of their design practices. The ABC *Nightline* news program also produced an interesting segment on IDEO [ABC01] that can be purchased at the ABC website. The Circle of Innovation by Tom Peters [Pet97] is another popular book that provides his perspective on current trends in business and the importance of design.

Chapter 2

Project Selection and Needs Identification

For every problem there is a solution that is simple, neat, and wrong.—H.L. Mencken

Traditionally, companies have organized resources based on functions such as accounting, engineering, finance, manufacturing, and marketing. It is often more effective to organize around projects that are of significant value and align resources to meet the needs of the project. This means that traditional departments and middle management are being de-emphasized and the role of projects is growing. Capstone design projects provide a great opportunity to gain experience in the management and execution of a project. One of the first and most important decisions encountered is selecting a project to pursue.

The objective of this chapter is to provide pragmatic guidance in the project selection phase. A description of design and engineering projects is presented, followed by advice on how projects can be selected by engineering students who wish to put design principles into practice. The chapter addresses how to identify the needs of the end-user and provides guidance for conducting background research. All of this information is brought together in a Problem Statement that identifies the needs, the goals of the project, and research on the technology.

Learning Objectives

By the end of this chapter, the reader should:

- Have an understanding of the types of projects that electrical and computer engineers undertake.

- Understand and be able to apply criteria for project selection.
- Know how to determine, document, and rank end-user needs.
- Be aware of resources available for conducting research surveys.
- Have selected a project concept and developed a Problem Statement.

2.1 Engineering Design Projects

This section provides a classification of design and describes some of the types of projects undertaken by practicing engineers and those tackled in student projects. In reality, most projects don't fit neatly into the categories presented, but are some combination of them. The objective of a design project is to create a new ***artifact*** (system, component, or process) to meet a given need. Within the design domain there are different types of designs that are classified broadly into three categories of Creative, Routine, and Variant designs [Cro00].

Creative designs represent new and innovative products. An example of a creative design is the Palm Pilot Personal Digital Assistant (PDA). While the idea for the PDA had been around for awhile, earlier attempts at developing the technology, notably the Apple Newton, were unsuccessful. This was primarily due to unreliable handwriting recognition that frustrated the user. However, Palm Computing had the creative idea to develop a simplified handwriting language, Graffiti, which eliminated the need for natural handwriting recognition. The Palm Pilot is a great example of a creative design—it is simple (four basic functions), fits in your pocket, and is easy to use. This innovation spawned a huge hand-held computing industry.

Variant designs are variations of existing designs, where the intent is to improve performance or add features to an existing system. Many engineering projects fall into this category. For example, the objective may be to increase accuracy or system throughput.

Routine designs represent the design of devices for which theory and practice are well-developed. Examples are DC power supplies, analog and digital filters, and basic digital components such as adders and comparators. Routine designs are often components of more complex creative and variant designs.

Within these three categories of design, there are many different types of projects. *Systems engineering and systems integration projects* represent the synthesis of many subsystems into a larger system. They may be creative or variant designs, but have unique challenges since they are typically large and involve many people and technologies. Adherence to good design processes is important for their success. Engineers are often engaged in *systems test*, where the objective is to ensure that a system meets stated requirements and the needs of the user. Examples include the testing of systems for use in space and military environments.

The objective in *experimental design projects* is to design experimental procedures and apparatus for determining the characteristics of a system. For example, an engineering team may test a system under a variety of operating conditions. Example 2.3 presented later in the chapter is such a project, where the objective was to design a series of experiments to test the feasibility of gigabit Ethernet technology in a military environment. The test explored the impact of environmental factors such as temperature and vibration, and further used this data to estimate the operating lifetime of the Ethernet board. Upon completion of this project, the team made recommendations as to the allowable operating ranges of the technology.

The objective in *analysis projects* is to analyze some aspect of an existing system to improve or correct it. For example, a system or process may be failing in the field and the source of the failure unknown. Tools such as the Failure Mode Effects and Analysis technique may be applied in this situation to identify the sources of failure. In t*echnology evaluation projects*, technologies are assessed to determine if they can be used in a given application. This may be to determine if the technology can improve an existing system, or to characterize its operating performance.

The objective of a *research project* is to perform research or experiments with the goal of discovering or creating a new technology. The fundamental difference between this and other types of projects is that the ultimate outcomes are unknown. Most engineering research falls under the category of *applied research*. This refers to the creation of new technology or systems based on existing technology and theory developed from fundamental research. *Fundamental research* emphasizes the discovery of new scientific principles without necessarily having an intended application. Fundamental research is very valuable, but not typically a part of design projects.

2.2 Sources of Project Ideas

Depending upon your situation, you may have the opportunity to identify and select your project. The list below provides some places to search for project ideas:

- *Industry sponsored projects.* Many companies will sponsor projects and are happy to do so, particularly if you have worked for them on an internship.
- *Engineers without Borders (www.ewb-usa.org).* This organization sponsors student projects to improve the quality of life in developing countries.
- *www.FreeRandD.com*. This is a clearinghouse for businesses and students teams to collaborate on projects. It allows businesses to post capstone project ideas for students to work on, while students can post resumes and project interests.
- *Your campus and local community.* In our school, a number of student teams have identified novel projects by asking other departments on campus for ideas. They have also been successful in approaching local community organizations for ideas, such as museums and research institutes.
- *Brainstorm.* Get together with a group of your peers and brainstorm on project ideas. You will be surprised at how many project ideas you can develop in a good brainstorming session (see Chapter 4). Do not only consider project ideas, but also brainstorm to identify problems that need solutions.

2.3 Project Feasibility and Selection Criteria

This section provides questions to consider when examining the feasibility of a project. George H. Heilmeier (an electrical engineer who has held positions as Chief Technology Officer of Texas Instruments, Director of the Defense Advanced Research Projects Agency, and CEO of Bellcore) developed a set of questions to answer when starting a new project [Sha94]. Heilmeier argued that all projects must be tied to the goals of the organization, and applied this by asking the following questions:

- What are you trying to do? Articulate your goals using absolutely no jargon.
- How is it done today, and what are the limitations of current practice?
- What is new in your approach, and why do you think it will be successful?

- Who cares? If you are successful, what difference will it make?
- What are the risks and payoffs?
- How much will it cost? How long will it take?
- What are the midterm and final exams to check for success?

Heilmeier credits successful completion of projects that he managed to answering these questions up-front and adhering to disciplined project management processes.

A second perspective is offered from an organizational project management viewpoint [Gra02] that provides the following criteria for project selection:

- *The project must be tied to the mission and vision of the organization.* Believe it or not, organizations often spend resources fruitlessly on projects that don't meet this criterion. To be fair, there is always risk associated with a project and it is sometimes hard to judge exactly how well a project meets this criterion. For engineers who are new to an organization, it is hard to judge a project's importance relative to the mission and goals, but if you find yourself in this situation, do not be afraid to ask some questions. Novices ask basic questions that are often overlooked by those who are highly experienced or intimately involved in a project.
- *Must have payback.* An economic analysis should be done to estimate if the project will make a profit. Much of this is outside the scope of this text, requiring marketing and financial analyses. Chapter 10 covers the basics of project cost estimation that will help in trying to answer this question.
- *Should have selection criteria.* Sound criteria for selecting among competing projects should be employed. The example at the end of this section demonstrates the application of criteria in project selection.
- *Objectives of the Project should be SMART: Specific, Measurable, Assignable, Realistic, Time-Related.* Chapter 3 addresses how to determine project requirements that are Specific and Measurable. Assignable, Realistic, and Time-Related all refer to project management aspects that are covered in Chapter 10. The objective is to develop tasks that are assigned to groups or individuals and realistically can be completed in the given timeframe.

The following example demonstrates how to apply a project selection model using a method known as the Analytical Hierarchy Process. AHP is a decision making method that is described in Appendix B and is utilized frequently throughout the text – **the reader should read Appendix B prior to proceeding with this example**.

Example 2.1 A project selection model for capstone design.

Assume that you are part of a capstone design team that has the opportunity to select their project from competing project ideas. The steps in making a decision using AHP are to select the criteria that drive the decision, determine relative weights of the criteria, rate the alternatives (in this case project concepts) against the criteria, to compute a weighted score for each of the alternatives, and then review the decision.

Step 1: Determine the selection criteria

To select the criteria, assume that the team brainstorms to determine the following criteria that interest the team members:

A – Match to team skills

B – Technical complexity

C – Creativity

D – Market potential

E – Industry sponsorship

Step 2: Determine the criteria weightings

Assume the team applies the method of pairwise comparison to determine the weights as shown in Appendix B. In order to do so, the team systematically compares each criterion to all others using the following scale of relative importance:

1 = equal, 3 = moderate, 5 = strong, 7 = very strong, 9 = extreme.

Again, details of pairwise comparison are outlined in Appendix B and the results are below.

This is an important step and one often overlooked – the team has identified what is important to it in project selection. It is clear that match to the team skills (criterion A) is most important, by a large margin, followed by market potential.

Step 3: Identify and rate alternatives relative to the criteria

Assume that the team identifies three potential projects ideas: 1 – IEEE sponsored robot competition, 2 – Industry sponsored project to design a new test protocol, and 3 – Design of an item-finder device to help people locate lost items. Furthermore, the team goes through the process of rating

Criteria	A	B	C	D	E	Weight
A	1	5	5	3	3	0.52
B	1/5	1	3	1/3	1/3	0.12
C	1/5	1/3	1	1	3	0.09
D	1/3	3	1	1	5	0.18
E	1/3	3	1/3	1/5	1	0.09

Table 2.1: Weighting for selection criteria.

Selection	Weights	Alternatives		
		Project 1	Project 2	Project 3
A (Match to skills)	0.52	0.40	0.20	0.40
B (Technical Complexity)	0.12	0.40	0.30	0.30
C (Creativity)	0.09	0.45	0.20	0.35
D (Market potential)	0.18	0.05	0.35	0.60
E (Industry sponsorship)	0.09	0.00	1.0	0.00
Score		0.31	0.31	0.38

Table 2.2: Weighted score for the selection criteria.

each project relative to the criteria as outlined in Appendix B. These ratings are reflected in the decision matrix in the next step.

Step 4: Compute scores for the alternatives

The decision matrix below is constructed and the scores for the alternatives determined.

Step 5: Review the decision

Project 3 (item finder) is rated the highest among the three choices based upon the weights determined by the team members. It is a good match to the team skills, but also matches their desire to solve a problem with good market potential. The remaining two projects are rated about equal.

2.4 Needs Identification

Often a customer, client, or supervisor comes to you with a problem to solve and you must determine the needs or requirements for the solution to the problem. In other words, determine the *voice of the customer*. This seems like a simple statement—ask the customer what they want and you are done, right?

As an illustration, let's say a client comes to you with the following request—*The traffic at the front of campus is too congested. I would like you to design a new traffic lane for northbound traffic exiting at the intersection at the front of the college.* So you design this new lane and have it added to the intersection. However, you find out three months later that the traffic congestion has decreased a little bit, but it is still a significant problem. So what went wrong? Clearly you did what was asked of you, but the problem was not solved, meaning that you were solving the wrong problem. The real problem was to improve the flow of traffic at the entrance. In this case, the client gave both the problem and the solution all in one statement. That is fine if a careful feasibility study was done and it was known that the additional traffic lane would alleviate the problem, but that was not the case here. This hypothetical situation is not so far fetched and happens in practice via neglect to do the up-front research or because underlying assumptions change. The point is that the correct problem should be identified and solved.

It would be better if the client had simply asked to improve the traffic flow, providing the opportunity to analyze the situation and develop different design options. Some questions to be asked in this situation are: *How much additional traffic is there? At what times does this happen? Where is the traffic coming from? What is an acceptable waiting time at the intersection?* It may be that several new lanes are needed, or perhaps the sequencing of the traffic signals is wrong, or maybe a new entrance could be added for less cost and improved traffic flow.

The lesson is that customers often come with the problems and solution all wrapped up together. When this is done, the ***design space***, the space of all possible solutions to the problem, is unnecessarily limited. Be ready to tactfully challenge the assumptions and ask questions to get to the root of the problem. Ask clarifying questions, analyze, pick apart the request, and focus on the problem, not the solution.

Researchers and practitioners have examined the problem of eliciting needs, and it is an important pre-requisite for developing good engineering requirements specifications. Ulrich and Eppinger [Ulr03] proposed a process for obtaining the *voice of the customer* using the following five steps: 1) Gather raw data from users; 2) Interpret raw data in terms of needs; 3) Organize needs into a hierarchy; 4) Determine the relative importance of the needs; and 5) Review the outcomes and the process. Each of these steps is described in the following sections.

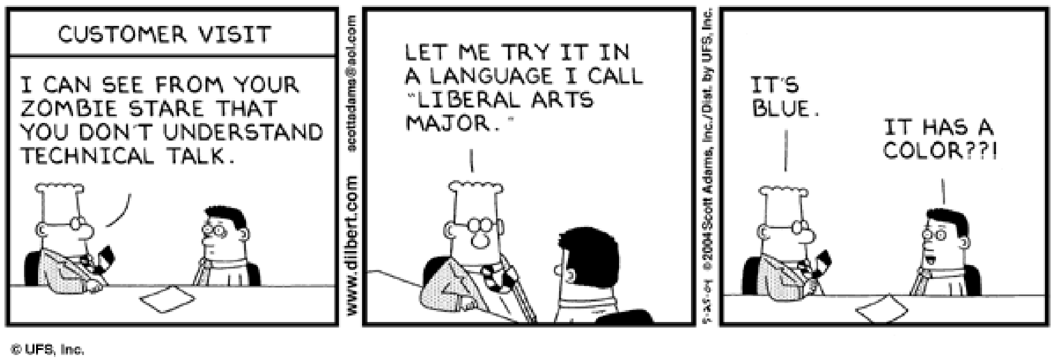


Figure 2.1: The difficulties of communicating with the customer. (Dilbert © United Feature Syndicate. Reprinted by permission.)

Step 1: Gather Raw Data from Users

DILBERT® by Scott Adams

This is often accomplished via interviews with supervisors, key users, or people from the client organization. In cases where new products are being developed, focus groups are often employed. The advantage of interviews and focus groups is that they provide the opportunity for dialogue with the user where new ideas, concepts, and needs may emerge. Another option is direct observation, where the team goes out and examines the system in use and develops concepts for improving it. IDEO Corporation is an innovative and successful company that designs new products and systems. They rely heavily on direct observation as a technique for successfully developing innovative products [Kel01]. For example, IDEO was asked by a client to develop a new medical instrument for balloon angioplasty used in hospital operating rooms. A critical requirement from the user was that only one hand could be used to operate the device because the technician's other hand had to be free during the procedure. From direct observation, the IDEO design team found that even though the current system was designed for one-hand use, it was impractical, and the technicians actually used both hands. IDEO designed and developed a two-handed pump that not only worked better than the one handed pump, but was quieter, easier to read, and had increased precision. This is another example of the customer specifying the solution as part of the problem statement.

Ulrich and Eppinger provide the following questions to ask during an interview:

- When and why do you use this type of product (system)?
- Walk us through a typical session using the product.
- What do you like about the existing products?
- What do you dislike about the existing products?
- What issues do you consider when purchasing the product?
- What improvements would you make to the product?

Step 2: Interpret the Raw Data in Terms of Needs

In this step the raw data is translated into customer needs. The needs are expressed in terms of what the system must do (a requirement) as opposed to how it is done. Statements of the customer's needs are known as ***marketing requirements*** or ***marketing specifications***. For example, "*The system should have high-quality audio*" is a need or marketing requirement from the customer regarding performance, but says nothing about how it will be achieved. Marketing requirements are short sentences that describe the need in the language of the customer. They typically do not have a numerical target and are described as a state of being for the system. Other examples of marketing requirements are, "*The system should be easy-to-use,*" and "*The system should be able survive a drop from the runner's height.*"

Step 3: Organize Needs into a Hierarchy

The marketing requirements are organized into a hierarchy of needs arranged from the most general to the most specific in successive levels of detail as required by the problem. It is organized by functional similarity, not as hierarchy of importance (that is the next step). This hierarchy is referred to as an ***objective tree***. An example objective tree for a portable audio device intended for use by runners is shown in Figure 2.2. The three high-level objectives determined were high-quality audio, portable, and easy-to-use. Each of these is further sub-divided into the characteristics that support the higher level. For example, portability is divided into the needs of lightweight, small, ergonomic, and the ability to operate in the environment. The environmental need is further expanded into needs that support it.

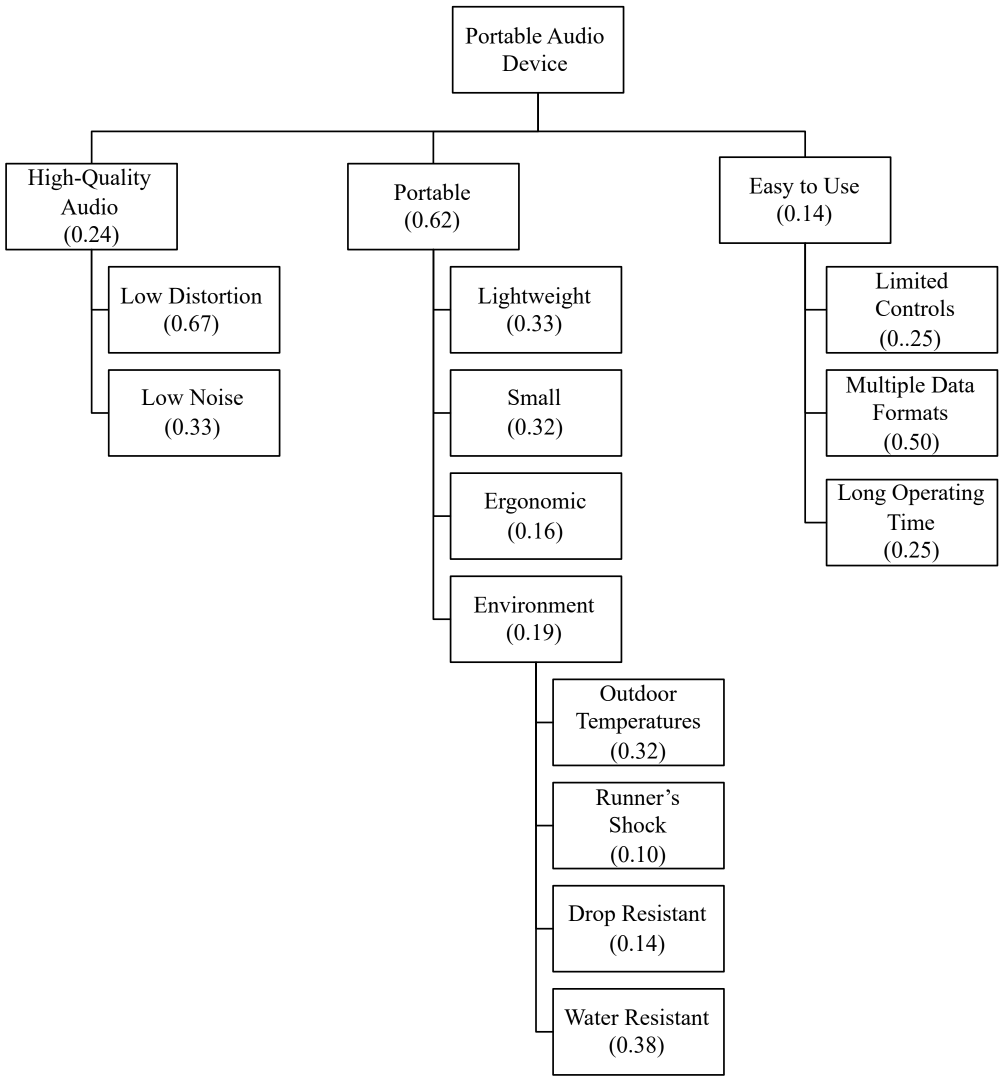


Figure 2.2: Objective tree for a portable audio device to be used by runners. The weights reflect the relative importance of needs at each level in the hierarchy as determined in Step 4 of the process.

	High-Quality Audio	Portable	Easy-to-Use	Weight
High-Quality Audio	1	1/3	2	0.24
Portable	3	1	4	0.62
Easy-to-Use	1/2	1/4	1	0.14

Table 2.3: Pairwise comparison matrix for ranking the highest-level needs of the portable audio device. This comparison should be carried out for all levels of the objective tree.

Step 4: Determine the Relative Importance of the Needs

The relative importance of the needs is determined based upon the user needs. As we saw in Example 2.1 and as presented in Appendix B, the pairwise comparison is a good technique for determining relative importance and weighting of needs. In pairwise comparison, all needs are systematically compared to all other needs at the same level in the hierarchy. An example pairwise comparison table for this problem is shown in Table 2.3 with the resulting weights for each need indicated. This shows that portability is the most important need, followed by audio quality and ease-of-use. The weights are also reflected in the objective tree in Figure 2.2. In addition, the needs at each sublevel in the hierarchy are compared, the results of which are reflected in Figure 2.2. The rankings are used in later chapters to compare design alternatives.

Step 5: Review the Outcomes and the Process

The design process and all of its sub-processes are methods for making good decisions, and this technique for needs identification is no different. There is a certain amount of subjectivity and judgment that goes into it; the end result should be reviewed to determine if it makes sense. The objective is to challenge assumptions, fully identify the problem, and make informed decisions.

The three outcomes of this process are the marketing requirements that identify the needs, an objective tree that provides a hierarchical representation of the needs, and a ranking of the relative importance of needs. This process may seem as though it does not apply to student design projects, but in reality it does. The questions in this chapter are certainly candidates to ask when working on company-sponsored projects. If it is not a company sponsored project, the user needs should still be considered. For example, questions can be asked of friends and co-workers who are potential users of the system, focus groups can be formed, surveys administered, and Internet bulletin boards and discussion groups employed to gather this information.

2.5 The Research Survey

It is important to conduct a thorough research survey while defining the project concept. Failure to do so may translate into time and money spent reinventing the wheel, while not taking full advantage of existing components, knowledge, practices, and technology. During the research phase, competing systems and technologies are identified, and based upon them the project concept refined, or in some cases abandoned. The character and strategy of the research survey is driven by the nature of the project. In general, the objective is to develop an understanding of the underlying scientific principles and demonstrate a familiarity with the state-of-the-art in the particular field. Some questions to be answered in the research survey are:

- What is the basic theory behind the concept?
- How is it currently being done?
- What are the limitations of current designs or technology?
- What are the similarities and differences between your concept and existing technologies?
- Are there existing or patented technologies that may be relevant to the design? If so, what are they and why are they relevant?

Internet Searching

The Internet is a powerful, fast, and readily accessible source for conducting research. There are many excellent search engines for locating web resources, but understand that it is important to go beyond the well-known search engines and beyond the Internet in the survey.

One of the risks, and also one of the wonderful things, about the Internet is that virtually anybody can post information. It is important to analyze websites to ensure that they are reliable and credible. There are resources available that provide pointers on how to evaluate this credibility [Mci02, Sch98], and a little common sense goes a long way. One of the important things to look for is authorship—the author should be clearly identified and any affiliations listed. Carefully determine whether the information is subjective opinion or possibly a commercial for a product. Credible sites should provide references to original sources of material. Another step is to verify website content in print media or other reliable sources.

There are thousands of search engines available, making the task of selecting one challenging. Also, there are different types of search engines: text (search for the text or keywords; subject heading or full-page text search), indexed (information categorized into directories), meta-search (engines that search other engines), and natural language processing (allowing natural language queries). A listing of search engines to try are www.altavista.com, www.AskJeeves.com, www.google.com, www.kartoo.com, and www.yahoo.com. The Librarian's Index to the Internet, www.lii.org is a collection selected and evaluated by librarians, and according to their website, it is *a "well-organized point of access for reliable, trustworthy, librarian-selected Internet resources.*" This information may change rapidly and represents current information at the time of publication.

Electrical and Computer Engineering Resources

Realize that the major search engines will not find all information available on the Internet. There are many websites with specialized search capabilities related to electrical and computer engineering design.

- EE Product Center, www.EEProductCenter.com. A website for locating electronic components and their manufacturers. It provides links to product datasheets and application notes. It has a keyword search engine and a tree structure search for finding components. For example, you can start with Op Amps and delve into sub-categories such as Precision and High-Speed.

- Circuit Cellar, www.CircuitCellar.com. This companion website for the magazine is a great reference for designers. It emphasizes embedded systems and electronics projects with many tutorial articles and project ideas.

- Datasheet Catalog, www.DatasheetCatalog.com. A datasheet source for electronic components and semiconductors.

- Dr. Dobbs, www.ddj.com. The magazine and companion website are a resource for software developers that includes tips and tutorials.

- EE Times, www.EETimes.com. Industry newspaper for electrical engineering field with information on current technology developments.

- Electronic Design Magazine, www.EDNmag.com. This is free magazine for electrical design engineers that provides information on the latest products. The website has a number of categorized technical resources and a design ideas section.

- ON Semiconductor, www.OnSemi.com. ON Semiconductor is a supplier of semiconductors for a wide range of applications, with a particular emphasis on power management. The website has a searchable database of over 15,000 components, and provides guidelines for component selection based on different applications.

- The Thomas Register, www.ThomasRegister.com. This is a source for finding companies and products in North America. It allows searches for parts and equipment that may be used in a design project. It provides profiles of companies that meet the search criteria and describes the products they make.

  In addition, most manufacturers of electronic components have websites providing product datasheets and application notes for their products. Application notes demonstrate how to use components in real applications. Examples are Dallas Semiconductor, Fairchild Semiconductor, Motorola, and Texas Instruments.

Government Resources

- US Bureau of Labor Statistics, `http://stats.bls.gov`. This has valuable information on consumer spending information, allowing one to determine things such as how much people spend and what they spend it on. It also profiles specific industries and forecasts employment in different industry sectors.

- US Government Official WebPortal, www.FirstGov.gov. This is an entrance to all US government web resources.

- US Patent Office, www.uspto.gov. A searchable database of all patents back to 1790. Full text searches are available back to 1976 and full images back to 1790. It has information on the basics of patents, trademarks, and copyrights.

Journal and Conference Papers

The search should include journal and conference papers if technically detailed information on the latest theory or applications is needed.

- ACM (Association for Computing Machinery) Digital Library, www.acm.org. Provides abstracts (full text for subscribers) for ACM journals and conference proceedings.
- Compendex, www.engineeringvillage2.org. This provides indices to journal and conference papers in a broad scope of engineering fields, referencing material back to 1970.
- IEEE (Institute of Electrical and Electronics Engineers) Xplore Electronic Library, www.ieee.org. Provides abstracts (full text for subscribers) to all IEEE journals, transactions, magazines, and conference subscriptions published since 1988. Abstracts for all IEEE standards are publicly available.

2.6 Needs and Objectives Statements

Two parts of the Problem Statement are the needs and objectives statements. The *needs statement* identifies and motivates the need for the project and should:

- Briefly and clearly state the need being addressed.
- Not provide a solution to the problem.
- Provide supporting information collected as outlined in Section 2.4.
- Provide any supporting statistics and anecdotes that support the need.
- Describe current limitations.
- Describe supporting processes that are needed to understand the need. This is particularly important in industry-sponsored projects having specific needs that may not be clear to the average person.

The *objectives statement* typically ranges from one or two sentences to one or two paragraphs in and should:

- Summarize what is being proposed to meet the need.
- Provide some preliminary design objectives (detailed requirements are developed later).

- Provide a preliminary description of the technical solution, avoiding a detailed description of the implementation. Often the input and output behavior of the system are described. The complete solution is not usually posed until after the engineering requirements are fully determined.

Example needs and objectives statements are provided in Examples 2.2–2.4.

Example 2.2 iPod Hands-Free Device Needs and Objectives. *Abstracted from the iPod Hands-Free Device Design Report by Al-Busaidi, Bellavia, and Roseborough [Alb06].*

Need: According to AppleInsider, approximately 10.3 million people owned iPods at the end of 2004 and many of the owners used them while operating their automobiles. The National Highway Traffic Safety Administration estimates that driver distraction is a contributing cause of 20 to 30 percent of all motor vehicle crashes – or 1.2 million accidents per year. One research study has estimated that driver inattention may cause as many as 10,000 deaths each year and approximately $40 billion in damages. iPods can present a distraction to drivers that is similar to cell phones in that the driver's attention is divided between controlling the steering wheel, watching the road, and navigating controls on the iPod. A system is needed to allow users to navigate among the music selections of their iPod without distracting their attention from the road.

Objective: The objective of this project is to design and prototype a device that will make the iPod safer to use while driving an automobile, by allowing hands-free control of the iPod. The device will interact with the user using spoken English commands. The user will be able to issue simple voice commands to the device to control the operation of the iPod. In turn, the device will communicate information verbally, such as song titles that are displayed on the iPod screen, to the user.

Example 2.3 Experimental Design Problem Needs and Objectives. *Abstracted from the Intel Pro 1000XF Server Testing Design Report by Esek, Hunt, and Lewis. [Ese03].*

Need: Our industry sponsor is investigating the performance of commercial grade gigabit Ethernet fiber optic equipment for computer data communications in a military environment. The proposed system will utilize an Intel Pro1000 XF server card. This is a harsh operating environment and its effects on the performance and lifetime of the equipment are unknown. The client wishes to understand how the military environment affects the optical power margin of the Intel Pro 1000 XF card and associated connectors and cabling.

Objective: The goal of this project is to design the experimental equipment and test procedures to determine the effects of temperature variations and vibration on the optical power margin and the operating lifespan of the system.

Example 2.4 Portable Aerial Surveillance Needs and Objectives. *Abstracted from the PASS Design Report by Andre, Kolb, and Thaler [And06].*

Need: Emergencies happen all across the world, all of the time. There are nearly 2,000,000 reported fires in the United States every year, and over 90 tactical activations of Pennsylvania's Special Emergency Response Team which handles barricaded suspects and hostage situations. There have been over 100 documented riots in the United States in the past century, with the Los Angeles Riot alone causing $1 billion in damage. Having an aerial view of these situations would be a great benefit to the emergency workers on the ground. For example, police may have to monitor a large crowd or a hostage situation where aerial surveillance would allow them to observe the situation from a safe distance and use the footage as evidence in court. Firefighters could use aerial surveillance to examine fire damaged buildings and search for victims through the windows of high-rise buildings. In large cities, emergency organizations often employ helicopters for aerial surveillance. However, in smaller rural towns, helicopters either take too long to reach the scene from a nearby city or they are too expensive to afford. The least expensive two-seat helicopters cost over $400,000, while new helicopters cost well over a million dollars with average operating costs of $400-$1000 per hour. There is a need for a low cost aerial device that can provide emergency workers with overhead surveillance of emergency situations.

Objective: The objective of this project is to design a device that will provide emergency workers with a live aerial view of a situation at a cost that small municipalities can afford. The device will deploy rapidly and record and log video. The camera will also include pan and zoom functionality to make identification of victims and suspects easier.

2.7 Project Application: The Problem Statement

A format for a Problem Statement that integrates the elements of this chapter is as follows:

- *Need.* A statement that identifies the needs of the project.

- *Objective.* Describes the concept proposed to meet the needs identified.
- *Background.* A summary of the research survey on the relevant technologies and systems. The objective is to provide an introduction answer the questions posed in Section 2.4. The length and content of this section varies depending upon the project.
- *Marketing Requirements.* Short statements describe the user needs.
- *Objective Tree.* A hierarchical representation of the needs based on functional similarity with the relative weights of the needs identified.

2.8 Summary and Further Reading

This chapter addressed the types of projects that are often undertaken by engineers and provides guidance in terms of questions to ask when selecting a project. The success of design projects depends upon adequately determining the user's needs and desires for the system. A process developed by Ulrich and Eppinger for needs elicitation was presented. The three outcomes of this process are: 1) marketing requirements identifying the customer needs, 2) an objective tree that hierarchically represents the needs, and 3) a ranking of the relative importance of the needs. It is important to conduct research on the concept and related technologies, and pointers for conducting the research survey were provided. Finally, a format for a Problem Statement was presented that summarizes the needs, objectives, and research survey for a design project.

The works by Griffin and Hauser [Gri93] and Ulrich and Eppinger [Ulr03] are readable and more detailed discussions on how to obtain the voice of the customer. There are also other design books available that address how to identify needs and develop objective trees [Cro00]. Cagan and Vogel [Cag02] have proposed a process for product development known as iNPD (integrated new product development) and provide methods for navigating what they refer to as "the fuzzy front end of project definition."

2.9 Problems

1. In your own words, describe the differences between creative, variant, and routine designs.

2. List three guidelines that should be employed when selecting a project.

3. Assume a customer comes to you with the following request—*Design a mechanical arm to pick apples from a tree.* What are the assumptions in this statement? Rewrite the request to eliminate the assumptions. (This problem was originally posed by Edward DeBono [Deb70]).

4. Assume a customer comes to you with the following request—*Design an RS-232 networked personal computer measurement system to transmit voltage measurements from a remote location to a central server.* What are the assumptions this statement? Develop a list of questions that you might ask the customer to further clarify the problem statement.

5. Describe what is meant by a marketing requirement.

6. What is the purpose of an objective tree and how is it developed?

7. The needs for a garage door opener have been determined to be: safety, speed, security, reliability, and noise. Create a pairwise comparison to determine the relative weights of the needs. Apply your judgment in making the relative comparisons.

8. Consider the design of an everyday consumer device such as computer printer, digital camera, electric screwdriver, or electric toothbrush. Determine the customer needs for the device selected. The deliverables should be: 1) marketing requirements, 2) an objective tree, and 3) a ranking of the customer needs using pairwise comparison.

9. **Project Application.** Select criteria to be applied for selecting a project concept as shown in Example 2.1 then brainstorm and search to generate project concepts. Rank the top three to five concepts against the criteria as presented in Example 2.1.

10. **Project Application.** Determine the needs for the project selected. The result should be list of marketing requirements, an objective tree, and a ranking of the needs.

11. **Project Application.** Conduct a research survey for your project using the guidance presented in Section 2.4. The result should be a report summarizing the results of the survey.

12. **Project Application.** Develop a Problem Statement for your project concept as outlined in Section 2.7. Apply the processes presented in the chapter as appropriate.

Appendix A Glossary

Term	Definition
acceptance test	An acceptance test verifies that the system meets the ***Requirements Specification*** and stipulates the conditions under which the customer will accept the system (Chapter 7).
activity on node	A form of a ***network diagram*** used in a project plan. In the Activity on Node (AON) form, activities are represented by nodes and the dependencies by arrows (Chapter 10).
activity	An activity is a combination of a ***task*** and its associated ***deliverables*** that is part of a project plan (Chapter 10).
activity view	The activity view is part of the ***Unified Modeling Language***. It is characterized by an activity diagram; its ***intention*** is to describe the sequencing of processes required to complete a task (Chapter 6).
Analytical Hierarchy Process (AHP)	A decision-making process that combines both quantitative and qualitative inputs. It is characterized by weighted criteria against which the decision is made, a numeric ranking of alternatives, and computation of a numerical score for each alternative (Appendix B and Chapters 2 and 4).
artifact	System, component, or process that is the end-result of a design (Chapter 2).
automated script test	An automated script test is a sequence of commands given to a unit under test. For example, a test may consist of a sequence of inputs that are provided to the unit, where the outputs for each input are then verified against pre-specified values (Chapter 7).
baseline requirements	The original set of requirements that are developed for a system (Chapter 3).
black box test	A test that is performed without any knowledge of internal workings of the unit under test (Chapter 7).

Term	Definition
bottom-up design	An approach to system design where the designer starts with basic components and synthesizes them to achieve the design objectives. This is contrasted to ***top-down*** design (Chapter 5).
Bohrbug	Bohrbugs are reliable ***bugs***, in which the error is always in the same place. This is analogous to the electrons in the Bohr atomic model which assume a definite orbit (Chapter 7).
brainstorming	A freeform approach to concept generation that is often done in groups. This process employs five basic rules: 1) no criticism of ideas, 2) wild ideas are encouraged, 3) quantity is stressed over quality, 4) build upon the ideas of others, and 5) all ideas are recorded (Chapter 4).
Brainwriting	A variation of ***brainstorming*** where a group of people systematically generate ideas and write them down. Ideas are then passed to other team members who must build upon them.
break-even point	The break-even point is the point where the number of units sold is such that there is no profit or loss. It is determined from the total costs and revenue (Chapter 10).
bug	A problem or error in a system that causes it to operate incorrectly (Chapter 7).
cardinality ratio	The cardinality ratio describes the multiplicity of the entities in a relationship. It is applied to ***entity relationship diagrams*** and Unified Modeling Language ***static view diagrams*** (Chapter 6).
class	Classes are used in object-oriented system design. A class defines the attributes and methods (functions) of an ***object*** (Chapter 6).
cohesion	Refers to how focused a module is—highly cohesive systems do one or a few things very well. Also see ***coupling*** (Chapter 5).
component design specification	See ***subsystem design specification*** (Chapter 3).
concept fan	A graphical tree representation of design decisions and potential solutions to a problem. Also see ***concept table*** (Chapters 1 and 4).
concept generation	A phase in the ***design process*** where many potential solutions to solve the problem are identified (Chapter 1).
concept table	A tool for generating concepts to solve a problem. It allows systematic examination of different combinations, arrangements, and substitutions of different elements for a system. Also see ***concept fan*** (Chapter 4).

Term	Definition
conditional rule-based ethics	An ethics system in which there are certain conditions under which an individual can break a rule. This is generally because it is believed that the moral good of the situation outweighs the rule. Also see ***rule-based ethics*** (Chapter 11).
constraint	A special type of requirement that encapsulates a design decision imposed by the environment or a stakeholder. Constraints often violate the abstractness property of engineering requirements (Chapter 3).
controllability	A principle that applies to testing. Controllability is the ability to set any node of the system to a prescribed value (Chapter 7).
copyright	Copyrights protect published works such as books, articles, music, and software. A copyright means that others cannot distribute copyrighted material without permission of the owner (Chapter 11).
coupling	Modules are coupled if they depend upon each other in some way to operate properly. Coupling is the extent to which modules or subsystems are connected. See also ***cohesion*** (Chapter 5).
creative design	A formal categorization of design projects. Creative designs represent new and innovative designs (Chapter 2).
critical path	The path with the longest duration in a project plan. It represents the minimum time required to complete the project (Chapter 10).
cross-functional team	Cross-functional teams are those that are composed of people from different organizational functions, such as engineering, marketing, and manufacturing. Also see ***multi-disciplinary team*** (Chapter 9).
data dictionary	A dictionary of data contained in a ***data flow diagram***. It contains specific information on the data flows and is defined using a formal language (Chapter 6).
data flow diagram	The ***intention*** of a data flow diagram (DFD) is to model the processing and flow of data inside a system (Chapter 6).
decision matrix	A matrix that is used to evaluate and rank concepts. It integrates both the user-needs and the technical merits of different concepts (Chapter 4).
derating	A decrease in the maximum amount of power that can be dissipated by a device. The amount of derating is based upon operating conditions, notably increases in temperature (Chapter 8).
deliverable	Deliverables are entities that are delivered to the project based upon completion of ***tasks.*** Also see ***activity*** (Chapter 10).

Term	Definition
descriptive design process	Describes typical activities involved in realizing designs with less emphasis on exact sequencing than a ***prescriptive design process*** (Chapter 1).
design architecture	The main (Level 1) organization and interconnection of modules in a system (Chapter 5).
design phase	Phase in the ***design process*** where the technical solution is developed, ultimately producing a detailed system design. Upon its completion, all major systems and subsystems are identified and described using an appropriate model (Chapter 1).
design process	The steps required to take an idea from concept to realization of the final system. It is a problem-solving methodology that aims to develop a system that best meets the customer's need within given constraints (Chapter 1).
design space	The space, or collection, of all possible solutions to a design problem (Chapter 2).
detailed design	A phase in the technical design where the problem can be decomposed no further and the identification of elements such as circuit components, logic gates, or software code takes place (Chapter 5).
engineering requirement	A requirement of the system that applies to the technical aspects of the design. An engineering requirement should be abstract, unambiguous, verifiable, traceable, and realistic (Chapter 3).
entity relationship diagram (ERD)	An ERD is used to model database systems. The ***intention*** of an ERD is to catalog a set of related objects (entities), their attributes, and the relationships between them (Chapter 6).
entity relationship matrix	A matrix that is used to identify relationships between entities in a database system (Chapter 6).
ethics	Philosophy that studies ***morality***, the nature of good and bad, and choices to be made (Chapter 11).
event	An event is an occurrence at a specific time and place that needs to be remembered and taken into consideration in the system design (Chapter 6).
event table	A table that is used to store information about ***events*** in the system. It includes information regarding the event trigger, the source of the event, and process triggered by the event (Chapter 6).
failure function	The failure function, $F(t)$, is a mathematical function that provides the probability that a device has failed at time t (Chapter 8).

Term	Definition
failure rate	The failure rate, $\lambda(t)$, for a device is the expected number of device failures that will occur per unit time (Chapter 8).
fixed costs	Fixed costs are those that are constant regardless of the number of units produced and cannot be directly charged to a process or activity (Chapter 10).
float	The amount of ***slippage*** that an activity in a project plan can experience without it becoming part of a new ***critical path*** (Chapter 10).
flowchart	A modeling diagram whose intention is to visually describe a process or algorithm, including its steps and control (Chapter 6).
functional decomposition	A design technique in which a system is designed by determining its overall functionality and then iteratively decomposing it into component subsystems, each with its own functionality (Chapter 5).
functional requirement	A ***subsystem design specification*** that describes the inputs, outputs, and functionality of a system or component (Chapters 3 and 5).
Gantt chart	Gantt charts are a bar graph representation of a project plan where the activities are shown on a timeline (Chapter 10).
Heisenbugs	Heisenbugs are ***bugs*** that are not always reproducible with the same input. This is analogous to the Heisenberg Uncertainty Principle, in which the position of an electron is uncertain (Chapter 7).
high-performance team	A team that significantly outperforms all similar teams. Part of the Katzenbach and Smith team model (Chapter 9).
integration test	An integration test is performed after the units of a system have been constructed and tested. The integration test verifies the operation of the integrated system behavior (Chapter 7).
intention	The intention of a model is the target behavior that it aims to describe (Chapter 6).
interaction view	The interaction view is part of the ***Unified Modeling Language***. Its ***intention*** is to show the interaction between objects. It is characterized by collaboration and sequence diagrams (Chapter 6).
key attribute	An attribute for an entity in a database system that uniquely identifies an instance of the entity (Chapter 6).

Term	Definition
lateral thinking	A thought process that attempts to identify creative solutions to a problem. It is not concerned with developing the solution for the problem, or right or wrong solutions. It encourages jumping around between ideas. It is contrasted to ***vertical thinking*** (Chapter 4).
liable	Required to pay monetary damages according to law (Chapter 11).
marketing requirement (specifications)	A statement that describe the needs of the customer or end-user of a system. They are typically stated in language that the customer would use (Chapters 2 and 3).
maintenance phase	Phase in the ***design process*** where the system is maintained, upgraded to add new functionality, or design problems are corrected (Chapter 1).
matrix test	A matrix test is a test that is suited to cases where the inputs submitted are structurally the same and differ only in their values (Chapter 7).
mean time to failure	The mean time to failure (MTTF) is a mathematical quantity which answers the question, "*On average how long does it take for a device to fail?*" (Chapter 8).
module	A block, or subsystem, in a design that performs a function (Chapter 5).
morals	The ***principles*** of right and wrong and the decisions that derive from those principles (Chapter 11).
multi-disciplinary team	In general, a multi-disciplinary team is one in which the members have complementary skills and the team may have representation from multiple technical disciplines. Also see ***cross-functional team*** (Chapter 9).
negligence	Failure to exercise caution, which in the case of design could be in not following reasonable standards and rules that apply to the situation (Chapter 11).
network diagram	A network diagram is a directed graph representation of the activities and dependencies between them for a project (Chapter 10).
Nominal Group Technique (NGT)	A formal approach to brainstorming and meeting facilitation. In NGT, each team member silently generates ideas that are reported out in a round-robin fashion so that all members have an opportunity to present their ideas. Concepts are selected by a multi-voting scheme with each member casting a predetermined number of votes for the ideas. The ideas are then ranked and discussed (Chapters 4 and 9).

Term	Definition
non-disclosure agreement	An agreement that prevents the signer from disseminating information about a company's products, services, and trade secrets (Chapter 11).
object	Objects represent both data (attributes) and the methods (functions) that can act upon data. An object represents a particular instance of a ***class***, which defines the attributes and methods (Chapter 6).
object type	Characteristic of a model used in design. The object type is capable of encapsulating the actual components used to construct the system (Chapter 6).
objective tree	A hierarchical tree representation of the customer's needs. The branches of the tree are organized based upon functional similarity of the needs (Chapter 2).
observability	This principle applies to testing. Observability is the ability to observe any node of a system (Chapter 7).
over-specificity	This refers to applying targets for ***engineering requirements*** that go beyond what is necessary for the system. Over-specificity limits the size of the ***design space*** (Chapter 3).
pairwise comparison	A method of systematically comparing all customer needs against each other. A comparison matrix is used for the comparison and the output is a scoring of each of the needs (Appendix B, Chapter 2, and Chapter 4).
parallel system	A system that contains multiple modules performing the same function where a single module would suffice. The overall system functions correctly when any one of the submodules is functioning (Chapter 8).
patent	A patent is a legal device for protecting a design or invention. If a patent is held for a technology, others cannot use it without permission of the owner (Chapter 11).
path-complete coverage	Path-complete coverage is where every possible ***processing path*** is tested (Chapter 7).
performance requirement	A particular type of ***engineering requirement*** that specifies performance related measures (Chapter 3).
physical view	The physical view is part of the ***Unified Modeling Language***. Its ***intention*** is to demonstrate the physical components of a system and how the logical views map to them. It is characterized by a component and deployment diagram (Chapter 6).

Term	Definition
potential team	A team where the sum effort of the team equals that of the individuals working in isolation. Part of the Katzenbach and Smith team model (Chapter 9).
prescriptive design process	An exact process, or systematic recipe, for realizing a system. Prescriptive design processes are often algorithmic in nature and expressed using flowcharts with decision logic (Chapter 1).
principle	Fundamental rules or beliefs that govern behavior, such as the Golden Rule (Chapter 11).
problem identification	The first phase in the design process where the problem is identified, the customer needs identified, and the project feasibility determined (Chapter 1).
processing path	A processing path is a sequence of consecutive instructions or states encountered while performing a computation. They are used to develop test cases (Chapter 7).
prototyping and construction phase	Phase in the ***design process*** in which different elements of the system are constructed and tested. The objective is to model some aspect of the system, demonstrating functionality to be employed in the final realization (Chapter 1).
pseudo-team	An under-performing team where the sum effort of the team is below that of the individuals working in isolation. Part of the Katzenbach and Smith team model (Chapter 9).
Pugh Concept Selection	A technique for comparing design concepts to the user needs. It is an iterative process where concepts are scored relative to the needs. Each concept is combined, improved, or removed from consideration in each iteration of the process (Chapter 4).
real team	A team where the sum effort of the team exceeds that of the individuals working in isolation. Part of the Katzenbach and Smith team model (Chapter 9).
redundancy	A design has redundancy if it contains multiple modules performing the same function where a single module would suffice. Redundancy is used to increase ***reliability*** (Chapter 8).
reliability	Reliability, $R(t)$, is the probability that a device is functioning properly (has not failed) at time t (Chapter 8).
research phase	Phase in the ***design process*** where research on the basic engineering and scientific principles, related technologies, and existing solutions for the problem are explored (Chapter 1).

Term	Definition
Requirements Specification	A collection of engineering and marketing requirements that a system must satisfy in order for it to meet the needs of the customer or end-user. Alternate terms that are used for the Requirements Specification are the *Product Design Specification* and the *Systems Requirements Specification* (Chapter 1 and 3).
reverse-engineering	Process where a device or process is taken apart to understand how it works (Chapter 11).
routine design	A formal categorization of design projects. They represent the design of artifacts for which theory and practice are well-developed (Chapter 2).
rule-based ethics	Rule-based ethics are based upon a set of rules that can be applied to make decisions. In the strictest form, they are considered to be absolute in terms of governing behavior (Chapter 11).
satisfice	Satisfice means that a solution may meet the design requirements, but not be the optimal solution (Chapter 11).
series system	A system in which the failure of a single component (or subsystem) leads to failure of the overall system (Chapter 8).
situational ethics	Situational ethics are where decisions are made based on whether they produce the highest good for the person (Chapter 11).
slippage	Refers to an activity in a project plan taking longer than its planned time to complete. See also ***critical path*** and ***float*** (Chapter 10).
standards	A standard or established way of doing things. Standards ensure that products work together, from home plumbing fixtures to the modules in a modern computer. They ensure the health and safety of products (Chapter 3).
state	The state of a system represents the net effect of all the previous inputs to the system. Since the state characterizes the history of previous inputs, it is often synonymous with the word memory (Chapter 6).
state diagram (machine)	Diagram used to describe systems with memory. It consists of states and transitions between states (Chapter 6).
static view	The static view is part of the ***Unified Modeling Language***. The ***intention*** of the static view is to show the classes in a system and their relationships. The static view is characterized by a class diagram (Chapter 6).
step-by-step test	A step-by-step test case is a prescription for generating a test and checking the results. It is most effective when the test consists of a complex sequence of steps (Chapter 7).

Term	Definition
strengths and weakness analysis	A technique for the evaluation of potential solutions to a design problem where the strengths and weaknesses are identified (Chapter 4).
structure charts	Specialized block diagrams for visualizing functional software designs. They employ input, output, transform, coordinate, and composite modules (Chapter 5).
strict liability	A form of ***liability*** that focuses only on the product itself—if the product contains a defect that caused harm, the manufacturer is liable (Chapter 11).
stub	A stub is a device that is used to simulate a subcomponent of a system during testing. Stubs simulate inputs or monitor outputs from the unit under test (Chapter 7).
subsystem design specification	Engineering requirements for subsystems that are constituents of a larger, more complex system (Chapter 3).
system integration	Phase in the ***design process*** where all of the subsystems are brought together to produce a complete working system (Chapter 1).
task	Tasks are actions that accomplish a job as part of a project plan. Also see ***activity*** and ***deliverable*** (Chapter 10).
Team Process Guidelines	Guidelines developed by a team that govern their behavior and identify expectations for performance (Chapter 9).
technical specification	A list of the technical details for a given system, such as operating voltages, processor architecture, and types of memory. The technical specification is fundamentally different from a requirement in that it indicates what was achieved in the end versus what a system needs to achieve from the outset. (Chapter 3).
test coverage	Test coverage is the extent to which the test cases cover all possible ***processing paths*** (Chapter 7).
test phase	Phase in the design process where the system is tested to demonstrate that it meets the requirements (Chapters 1 and 7).
testable	A design is testable when a failure of a component or subsystem can be quickly located. A testable design is easier to debug, manufacture, and service in the field (Chapter 7).
top-down design	An approach to design in which the designer has an overall vision of what the final system must do, and the problem is partitioned into components, or subsystems that work together to achieve the overall goal. Then each subsystem is successively refined and partitioned as necessary. This is contrasted to ***bottom-up*** design (Chapter 5).

Term	Definition
tort	The basis for which a lawsuit is brought forth (Chapter 11).
trade secret	An approach to protecting intellectual property where the information is held secretly, without ***patent*** protection, so that a competitor cannot access it (Chapter 11).
under-specificity	This refers to a state of the ***Requirements Specification***. When it is under-specified, requirements do not meet the needs of the user and/or embody all of the requirements needed to implement the system (Chapter 3).
Unified Modeling Language (UML)	A modeling language that captures the best practices of object-oriented system design. It encompasses six different system views that can be used to model electrical and computer systems (Chapter 6).
unit test	A unit test is a test of the functionality of a system module in isolation. It establishes that a subsystem performs a single unit of functionality to some specification (Chapter 7).
use-case view	The use-case view is part of the ***Unified Modeling Language***. Its ***intention*** is to capture the overall behavior of the system from the user's point of view and to describe cases in which the system will be used (Chapter 6).
utilitarian ethics	In utilitarian ethics, decisions are made based upon the decision that brings about the highest good for all, relative to all other decisions (Chapter 11).
validation	The process of determining whether the requirements meet the needs of the user (Chapter 3).
value	A value is something that a person or group believes to be valuable or worthwhile. Also see ***principles*** and ***morals*** (Chapter 11).
variable costs	Variable costs vary depending upon the process or items being produced, and fluctuate directly with the number of units produced (Chapter 10).
variant design	A formal categorization of design projects. They represent the design of existing systems, where the intent is to improve performance or add features (Chapter 2).
verifiable	Refers to a property of an engineering requirement. It means that there should be a way to measure or demonstrate that the requirement is met in the final system realization (Chapter 3).
vertical thinking	A linear, or sequential, thought process that proceeds logically towards the solution of a problem. It seeks to eliminate incorrect solutions. It is contrasted to ***lateral thinking*** (Chapter 4).

Term	Definition
whistleblower	A person who goes outside of their company or organization to report an ethical or safety problem (Chapter 11).
white box test	White box tests are those that are conducted with knowledge of the internal working of the unit under test (Chapter 7).
work breakdown structure	The work breakdown structure (WBS) is a hierarchical breakdown of the tasks and deliverables that need to be completed in order to accomplish a project (Chapter 10).
working group	A group of individuals working in isolation, who come together occasionally to share information. Part of the Katzenbach and Smith team model (Chapter 9).

Index