

---

## PREFACE

---

**T**wo years ago, I took my six-year-old son to a “robot race” up in the Rockies near Boulder. It was held in the community center of a small mountain town. Nevertheless, it was packed with about 100 enthusiastic people and many interesting exhibits. The central event was to be a timed race along a prescribed course. Several school-aged kids had entered plastic robots clearly built from parts from the same toy manufacturer. The racecourse was a plastic mat approximately 15 feet on each side. The robots had to follow a one-inch-wide, serpentine black line on the mat from beginning to end. The winner would be the robot finishing with the fastest time.

I watched the kids tuning up their robots on the racecourse before the race. Each robot had a sensor on each side that could detect the black line. If the robot moved forward and started to cross the line, the electronics would correct the steering and move the robot back on course.

It was clear the kids were all having trouble. None of the robots could follow the course from beginning to end. They would invariably lurch too far over the black racecourse line and get lost, spinning in useless circles. Legions of adult advisors huddled with the kids, making all sorts of changes, yet nobody was making progress. To me, the answer was obvious and I wanted to help.

Off in the corner, a bit cowed and unsure of himself, was the youngest competitor. Let's call him Sam. He may have been 13 and was there with his mom. They, too, were making changes without good results. I approached Sam's mom, discretely asked permission to help, and joined their team. Without going into the theory, I explained that the robots were all too fast and powerful for their own control systems. I recommended slowing down Sam's robot by adding more weight at the back end. We finally decided to build a sled for the robot to drag and set about finding the materials. With the race deadline approaching, Sam himself came up with the solution. With a quick glance to ask permission, he grabbed his mom's handheld camera and slipped the wrist strap over a post on the rear of the robot. We confirmed the robot could still move slowly down the racecourse line towing the camera. Sam took the batteries out of the camera until it was near the right weight. All too soon, race time came and we had to halt our experiment.

One after another, the older competitors' robots raced down the course only to stray off the black line and be disqualified. A couple of the robots did finish after wandering around lost and wasting a good deal of time. Eventually, the time came for Sam to race his robot. He placed his robot on the starting line, plopped his mom's camera down behind it, confidently put the wrist strap on the rear bumper, and pushed the start button with a bit of ceremony. As Sam's robot left the starting line, it lurched forward, tugging the camera behind it. The crowd started to buzz and I watched the highly amused advisors talking among themselves. It was clear some of them understood what was going on.

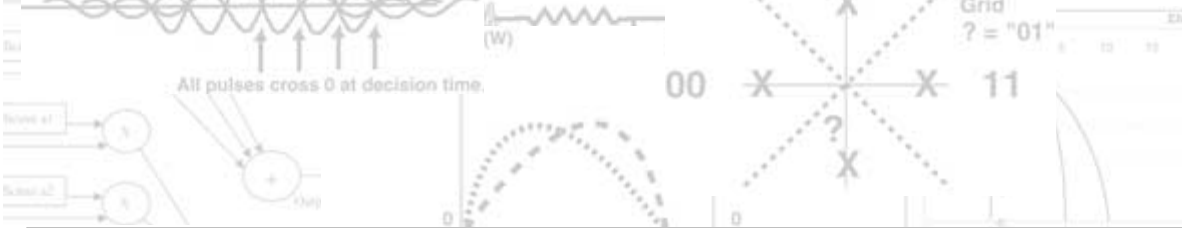
To make a long story short, Sam's robot reliably chugged around the racecourse and he won. The look on his face alone was worth the effort. Sam's nominal reward was a kit for a bigger robot, but I think he walked away with much more than that.

After the race, Sam was eager to know how I knew the solution. I took Sam aside and gave him a glimpse of the college-level mathematics and graphs that were behind his victory. My intention was to stimulate his curiosity and point him in the direction that would lead him to further accomplishments. I went home feeling wonderful, proud for myself, and happy for Sam.

After all, everyone seeks direction in life. We experience a feeling of comfort when we discover that our problems are definitive, comprehensible, and tractable. To build a successful robot, it takes a disciplined approach. Many pitfalls are possible, but they are not inevitable. The subjects you will have to master are many and difficult, but not incomprehensible.

To be clear, it is not the intention of my book to teach you how to build a robot. Others can find the nuts and bolts better than I, but if you want to come away enriched with the seminal knowledge of the academic and professional disciplines necessary to be successful in the field, then this book is for you. Each major discipline is the subject of a separate chapter. Each chapter will cover the basics but will also lead you to theory and reasoning that can capture the imagination. For each discipline, legions of engineers and professors spend their entire careers sweating the details.

Sam, if you're out there, I hope one of them is you.



---

# INTRODUCTION

---

**T**he boundless energy of youth often must give way to the laws of physics. All too often I've seen bright ideas flounder for a lack of fundamental knowledge. If this book can foster the development of the art, if it can encourage and educate the robotic community, if it can provide the missing ingredients—the secret sauce—then I did my job right. If you have a sense that a robot is more than wires and wheels, then this book is for you.

Math rules physics, and physics rules robots. This book sheds light on the math and physics behind a robot design, and does so in an accessible way. The text was written for all ages, from high school through college and beyond. The math used in the book includes algebra, calculus, and differential equations. For readers unacquainted with these subjects, I made sure the text “returns to Earth” often. Nobody should be left behind. The laws of physics and math are evident in everyday life, and several examples are given in this book. Throughout the history of science and technology, the path to great discoveries has almost always started with the observation of simple events. Newton's apple, Einstein's empty room in space, and Shannon's word games are clear examples. Proceeding from an intuitive, personal understanding of the basic laws of

physics and math, you can take your understanding further. Using this knowledge, you can predict the behavior of your robot in advance. As problems crop up, you'll have the basic knowledge to move effectively toward solutions.

Throughout the book, I've also thrown in experience gained from 32 years of engineering design. I can't be there when you build your robot, but I can put tools in your belt and pass on such wisdom as we both can sit still for.

Originally, I started this project for the fame, fortune, and groupies. As the chapters rolled out, I got my true rewards. I relearned the basic technologies to better explain them. I dug into the larger questions lurking behind the equations and technology. And as the book developed, I found an outlet for other thoughts I've had for quite some time. I hope my philosophical asides prove entertaining.

The book is divided into chapters that deal with monolithic subjects like computer hardware, computer software, *digital signal processing* (DSP), communications, power, and control systems. It is my hope that readers will find these individual subjects compelling enough to pursue them further. In each chapter, I've included URLs for web sites that explain the technologies in more depth. The Web can be a great place to obtain a continuing education.

Chapter 1 covers project management. More robots bite the dust for a lack of management discipline than any other reason. Building robots is much like going into battle. You can do great damage coming straight out of the gate and swinging swords, but it takes planning to make sure only the enemy gets cut. The chapter outlines how to approach a robot project from the outset. It includes development process flowcharts, checklists, and lots of tips. Robots are not built; they are born. With forethought and preparation, the process can be much less painful. And lest we forget, the project depends on people. Motivation and management, of self and others, are required for success.

Chapter 2 covers control systems. This is a complex field with a language of its own and many disciplines. If someone were to gather data about why robot projects fail, I'm guessing mechanics and power problems would come first. Control system problems would be right up there, too. The chapter discusses control system architecture; distributed and centralized control systems are compared and contrasted. Most robots have centralized systems and use open-loop and closed-loop control methods. The text outlines the basic behavior of a second order-control system, a good model for the behavior of many robotic systems. The text explains the math needed to understand and control system behavior. Specific examples of ways to design and correct such a control system are also given. Last of all, I've thrown in all the tricks of the trade that I know.

Chapter 3 covers computer hardware. I've outlined many of the reasons for using a computer in a robot and ways to accelerate the design process. Several computer architectures are listed, including analog, general-purpose digital, DSP, neural networks, and parallel processors. I've outlined the basic architecture of general-purpose digital

microprocessors and commented on the applicability of various computer options. Just as the lack of planning can ruin a robot project, so too can the wrong choice of microprocessor. The last part of the chapter has a large checklist that can help you through the process of selecting a computer.

Chapter 4 covers reliability, safety, and compliance. The first section defines reliability and provides methods for predicting and measuring it. The chapter also includes a list of components to be wary of and some advice about using them. In the safety section is a list of dangers that can sneak up on even the most experienced designers, and it also offers advice about managing risks. The compliance and testing section covers environmental considerations, emissions, and many tips for forestalling problems.

Chapter 5 covers the early stage of the design process, the *high-level design* (HLD). The text covers where to start, what to consider first, and how to make the design gel early. Although every robotic project will be different, I wanted the chapter to document how I would go about designing a robot. I closed my eyes, gave myself a phantom team of engineers, and wrote down what I'd do. Let me know if you'd do it differently.

Chapter 6 covers power and energy. First, I discuss how to determine the robot's energy requirements. It outlines a series of considerations that should be taken into account in the selection and use of an energy source, with a specific concentration on batteries.

Chapter 7 covers energy and software control systems, with an emphasis on energy management. It includes a list of specific actions to take in the design of an energy-efficient robot. I mentioned many considerations that should be kept in mind during the selection and design of robotic software. The chapter outlines a coordinated approach to the selection of a processor, a battery, a power supply, operating software, and application software. Included are many software techniques that have proven successful, including a discussion of braking methods.

Chapter 8 covers DSP and the chapter starts with an example of DSP processing that is familiar to all of us. This leads to the two basic theorems of DSP. Specific examples illustrate the need for both learning and using the theorems. The chapter includes different methods of constructing a classic DSP control system. I've included rules of thumb for picking components, methods for programming them, and ways to test them.

Chapter 9 covers communication, which is vital to the effectiveness and power of people, and robots are not far behind in this need. The chapter starts with the definition of communication, the concept of noise, and Shannon's theorem for the capacity of a noisy communications link. I discuss baseband transmission, the basic techniques for sending pulses down a wire, and the common baseband communication links, including the Ethernet. The chapter outlines the reasons for modulated communication and some of the methods for doing so. The emphasis is on the transmission of digital data and the control of errors in a noisy communication channel. I've explained several methods of encoding the data that make modern wireless communication possible. The chapter lists and explains many of the standard tools used by communication engineers,

## **XIV INTRODUCTION**

---

including coding, multiuser access, security, and compression. Lastly, I've described a few of the most popular communication protocols that can be used in a robot project.

Chapter 10 covers motors. Engineers classify motors by the type of power they consume. AC and DC motors (including stepping motors) are discussed along with the different internal structures that make them work. The advantages and disadvantages of each type are presented as well.

Chapter 11 covers mechanics and covers the selection and the relevant properties of materials. Many robots have mechanical problems, so several design tips are included. In addition, short sections are dedicated to static and dynamic calculations.