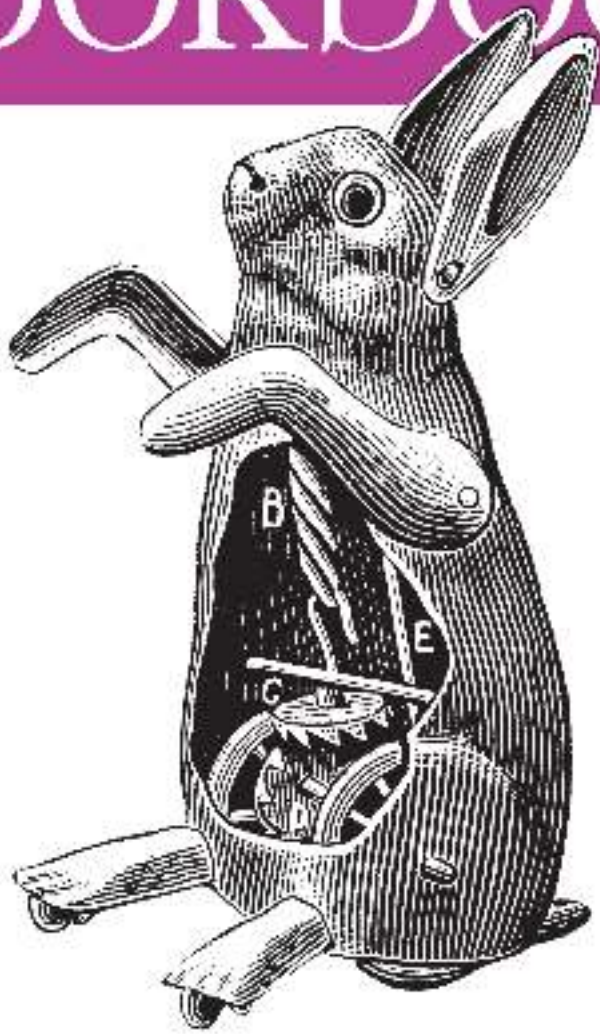
Recipes to Begin, Expand, and Enhance Your Projects

Arduino Cookbook



O'REILLY®

Michael Margolis

Arduino Cookbook

Create your own toys, remote controllers, alarms, detectors, robots, and many other projects with the Arduino device. This simple microcontroller board lets artists and designers build a variety of amazing objects and prototypes that interact with the physical world. With this cookbook, you can dive right in and experiment with more than a hundred tips and techniques, no matter what your skill level is.

The recipes in this book provide solutions for most common problems and questions Arduino users have, including everything from programming fundamentals to working with sensors, motors, lights, and sound, and communicating over wired and wireless networks. You'll find the examples and advice you need to begin, expand, and enhance your projects right away.

- Get to know the Arduino development environment
- Understand the core elements of the Arduino programming language
- Use common output devices for light, motion, and sound
- Interact with almost any device that has a remote control
- Learn techniques for handling time delays and time measurement
- Transfer digital information from sensors to the Arduino device
- Create complex projects that incorporate shields and external modules
- Use and modify existing Arduino libraries, and learn how to create your own

Previous programming experience is recommended.

"Michael Margolis's comprehensive set of recipes is a fine gift to the burgeoning Arduino community. Whatever your background or skill, the Cookbook provides solutions for that project you're wrestling with today and fuel for imagining what you'll build tomorrow. I doubt it will ever leave my workbench table."

—Mikal Hart
Arduino Uno Advisory Team

Michael Margolis, a technologist in the field of real-time computing, has expertise in developing and delivering hardware and software for interacting with the environment. Formerly the Chief Technical Officer with Avaya, he has more than 30 years of experience with Sony, Microsoft, and Lucent/Bell Labs.

US $44.99 CAN $51.99
ISBN: 978-0-596-80247-9
9 780596 802479 54499

Twitter: @oreillymedia
facebook.com/oreilly

O'REILLY®
oreilly.com

Arduino Cookbook

Arduino Cookbook

Michael Margolis

O'REILLY®
Beijing • Cambridge • Farnham • Köln • Sebastopol • Tokyo

Arduino Cookbook
by Michael Margolis

Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

O'Reilly books may be purchased for educational, business, or sales promotional use. Online editions are also available for most titles (*http://my.safaribooksonline.com*). For more information, contact our corporate/institutional sales department: (800) 998-9938 or *corporate@oreilly.com*.

Editors: Simon St. Laurent and Brian Jepson
Production Editor: Teresa Elsey
Copyeditor: Audrey Doyle
Proofreader: Teresa Elsey
Indexer: Lucie Haskins
Cover Designer: Karen Montgomery
Interior Designer: David Futato
Illustrator: Robert Romano

Printing History:
March 2011: First Edition.

ISBN: 978-0-596-80247-9

[LSI]

1299267108

Table of Contents

Preface

This book was written by Michael Margolis with Nick Weldin to help you explore the amazing things you can do with Arduino.

Arduino is a family of microcontrollers (tiny computers) and a software creation environment that makes it easy for you to create programs (called *sketches*) that can interact with the physical world. Things you make with Arduino can sense and respond to touch, sound, position, heat, and light. This type of technology, often referred to as *physical computing*, is used in all kinds of things, from the iPhone to automobile electronics systems. Arduino makes it possible for anyone—even people with no programming or electronics experience—to use this rich and complex technology.

Who This Book Is For

Unlike in most technical cookbooks, experience with software and hardware is not assumed. This book is aimed at a broad range of readers interested in using computer technology to interact with the environment. It is for people who want to quickly find the solution to hardware and software problems.

You may have no programming experience—perhaps you have a great idea for an interactive project but don't have the skills to develop it. This book will help you learn what you need to know to write code that works, using examples that cover the kinds of tasks you want to perform.

If you have some programming experience but are new to Arduino, the book will help you become productive quickly by demonstrating how to implement specific Arduino capabilities for your project.

People already using Arduino should find the content helpful for quickly learning new techniques, which are explained using practical examples. This will help you to embark on more complex projects by showing how to solve problems and use capabilities that may be new to you.

Experienced C/C++ programmers will find examples of how to use the low-level AVR resources (interrupts, timers, I2C, Ethernet, etc.) to build applications using the Arduino environment.

How This Book Is Organized

The book contains information that covers the broad range of the Arduino's capabilities, from basic concepts and common tasks to advanced technology. Each technique is explained in a recipe that shows you how to implement a specific capability. You do not need to read the content in sequence.

Chapter 1, *Getting Started*, introduces the Arduino environment and provides help on getting the Arduino development environment and hardware installed and working.

The next couple of chapters introduce Arduino software development. Chapter 2, *Making the Sketch Do Your Bidding*, covers essential software concepts and tasks, and Chapter 3, *Using Mathematical Operators*, shows how to make use of the most common mathematical functions.

Chapter 4, *Serial Communications*, describes how to get Arduino to connect and communicate with your computer and other devices. Serial is the most common method for Arduino input and output, and this capability is used in many of the recipes throughout the book.

Chapter 5, *Simple Digital and Analog Input*, introduces a range of basic techniques for reading digital and analog signals. Chapter 6, *Getting Input from Sensors*, builds on this with recipes that explain how to use devices that enable Arduino to sense touch, sound, position, heat, and light.

Chapter 7, *Visual Output*, covers controlling light. Recipes cover switching on one or many LEDs and controlling brightness and color. This chapter explains how you can drive bar graphs and numeric LED displays, as well as create patterns and animations with LED arrays. In addition, the chapter provides a general introduction to digital and analog output for those who are new to this.

Chapter 8, *Physical Output*, explains how you can make things move by controlling motors with Arduino. A wide range of motor types are covered: solenoids, servo motors, DC motors, and stepper motors.

Chapter 9, *Audio Output*, shows how to generate sound with Arduino through an output device such as a speaker. It covers playing simple tones and melodies and playing WAV files and MIDI.

Chapter 10, *Remotely Controlling External Devices*, describes techniques that can be used to interact with almost any device that uses some form of remote controller, including TV, audio equipment, cameras, garage doors, appliances, and toys. It builds on techniques used in previous chapters for connecting Arduino to devices and modules.

Chapter 11, *Using Displays*, covers interfacing text and graphical LCD displays. The chapter shows how you can connect these devices to display text, scroll or highlight words, and create special symbols and characters.

Chapter 12, *Using Time and Dates*, covers built-in Arduino time-related functions and introduces many additional techniques for handling time delays, time measurement, and real-world times and dates.

Chapter 13, *Communicating Using I2C and SPI*, covers the Inter-Integrated Circuit (I2C) and Serial Peripheral Interface (SPI) standards. These standards provide simple ways for digital information to be transferred between sensors and Arduino. This chapter shows how to use I2C and SPI to connect to common devices. It also shows how to connect two or more Arduino boards, using I2C for multiboard applications.

Chapter 14, *Wireless Communication*, covers wireless communication with XBee. This chapter provides examples ranging from simple wireless serial port replacements to mesh networks connecting multiple boards to multiple sensors.

Chapter 15, *Ethernet and Networking*, describes the many ways you can use Arduino with the Internet. It has examples that demonstrate how to build and use web clients and servers and shows how to use the most common Internet communication protocols with Arduino.

Arduino software libraries are a standard way of adding functionality to the Arduino environment. Chapter 16, *Using, Modifying, and Creating Libraries*, explains how to use and modify software libraries. It also provides guidance on how to create your own libraries.

Chapter 17, *Advanced Coding and Memory Handling*, covers advanced programming techniques, and the topics here are more technical than the other recipes in this book because they cover things that are usually concealed by the friendly Arduino wrapper. The techniques in this chapter can be used to make a sketch more efficient—they can help improve performance and reduce the code size of your sketches.

Chapter 18, *Using the Controller Chip Hardware*, shows how to access and use hardware functions that are not fully exposed through the documented Arduino language. It covers low-level usage of the hardware input/output registers, timers, and interrupts.

Appendix A, *Electronic Components*, provides an overview of the components used throughout the book.

Appendix B, *Using Schematic Diagrams and Data Sheets*, explains how to use schematic diagrams and data sheets.

Appendix C, *Building and Connecting the Circuit*, provides a brief introduction to using a breadboard, connecting and using external power supplies and batteries, and using capacitors for decoupling.

Appendix D, *Tips on Troubleshooting Software Problems*, provides tips on fixing compile and runtime problems.

Appendix E, *Tips on Troubleshooting Hardware Problems*, covers problems with electronic circuits.

Appendix F, *Digital and Analog Pins*, provides tables indicating functionality provided by the pins on standard Arduino boards.

Appendix G, *ASCII and Extended Character Sets*, provides tables showing ASCII characters.

What Was Left Out

There isn't room in this book to cover electronics theory and practice, although guidance is provided for building the circuits used in the recipes. For more detail, readers may want to refer to material that is widely available on the Internet or to books such as the following:

- *Make: Electronics* by Charles Platt (O'Reilly)
- *Getting Started in Electronics* by Forrest Mims (Master Publishing)
- *Physical Computing* by Tom Igoe (Cengage)
- *Practical Electronics for Inventors* by Paul Scherz (McGraw-Hill)

This cookbook explains how to write code to accomplish specific tasks, but it is not an introduction to programming. Relevant programming concepts are briefly explained, but there is insufficient room to cover the details. If you want to learn more about programming, you may want to refer to the Internet or to one of the following books:

- *Practical C Programming* by Steve Oualline (O'Reilly)
- *A Book on C* by Al Kelley and Ira Pohl (Addison-Wesley)

My favorite, although not really a beginner's book, is the book I used to learn C programming:

- *The C Programming Language* by Brian W. Kernighan and Dennis M. Ritchie (Prentice Hall)

Code Style (About the Code)

The code used throughout this book has been tailored to clearly illustrate the topic covered in each recipe. As a consequence, some common coding shortcuts have been avoided, particularly in the early chapters. Experienced C programmers often use rich but terse expressions that are efficient but can be a little difficult for beginners to read. For example, the early chapters increment variables using explicit expressions that are easy for nonprogrammers to read:

```
result = result + 1; // increment the count
```

Rather than the following, commonly used by experienced programmers, that does the same thing:

```
result++;  // increment using the post increment operator
```

Feel free to substitute your preferred style. Beginners should be reassured that there is no benefit in performance or code size in using the terse form.

Some programming expressions are so common that they are used in their terse form. For example, the loop expressions are written as follows:

```
for(int i=0; i < 4; i++)
```

This is equivalent to the following:

```
int i;
for(i=0; i < 4; i = i+1)
```

See Chapter 2 for more details on these and other expressions used throughout the book.

Good programming practice involves ensuring that values used are valid (garbage in equals garbage out) by checking them before using them in calculations. However, to keep the code focused on the recipe topic, very little error-checking code has been included.

Arduino Platform Release Notes

The code has been tested using Arduino releases from version 0018 through version 0020. This book was written before Arduino v1.0 was finalized, and although almost all of the examples should still work, small changes required for running with v1.0 will be published on the site for the book:

http://www.oreilly.com/catalog/9780596802479/

There's also a link to errata there. Errata give readers a way to let us know about typos, errors, and other problems with the book. Errata will be visible on the page immediately, and we'll confirm them after checking them out. O'Reilly can also fix errata in future printings of the book and on Safari, making for a better reader experience pretty quickly.

If you have problems making examples work, check the web link to see if the code has been updated. If that doesn't fix the problem, see Appendix D, which covers troubleshooting software problems. The Arduino forum is a good place to post a question if you need more help: *http://www.arduino.cc*.

We hope to keep this book updated for future Arduino versions, and we will also incorporate suggestions and complaints into future editions.

If you like—or don't like—this book, by all means, please let people know. Amazon reviews are one popular way to share your happiness (or lack of happiness), or you can leave reviews at the site for the book.

Conventions Used in This Book

The following font conventions are used in this book:

Italic
: Indicates pathnames, filenames, and program names; Internet addresses, such as domain names and URLs; and new items where they are defined

`Constant width`
: Indicates command lines and options that should be typed verbatim; names and keywords in programs, including method names, variable names, and class names; and HTML element tags

`Constant width bold`
: Indicates emphasis in program code lines

`Constant width italic`
: Indicates text that should be replaced with user-supplied values

This icon signifies a tip, suggestion, or general note.

This icon indicates a warning or caution.

Using Code Examples

This book is here to help you make things with Arduino. In general, you may use the code in this book in your programs and documentation. You do not need to contact us for permission unless you're reproducing a significant portion of the code. For example, writing a program that uses several chunks of code from this book does not require permission. Selling or distributing a CD-ROM of examples from this book *does* require permission. Answering a question by citing this book and quoting example code does not require permission. Incorporating a significant amount of example code from this book into your product's documentation *does* require permission.

We appreciate, but do not require, attribution. An attribution usually includes the title, author, publisher, and ISBN. For example: "*Arduino Cookbook* by Michael Margolis with Nick Weldin (O'Reilly). Copyright 2011 Michael Margolis and Nicholas Weldin, 9780596802479."

If you feel your use of code examples falls outside fair use or the permission given here, feel free to contact us at *permissions@oreilly.com*.

Safari® Books Online

Safari Books Online is an on-demand digital library that lets you easily search over 7,500 technology and creative reference books and videos to find the answers you need quickly.

With a subscription, you can read any page and watch any video from our library online. Read books on your cell phone and mobile devices. Access new titles before they are available for print, and get exclusive access to manuscripts in development and post feedback for the authors. Copy and paste code samples, organize your favorites, download chapters, bookmark key sections, create notes, print out pages, and benefit from tons of other time-saving features.

O'Reilly Media has uploaded this book to the Safari Books Online service. To have full digital access to this book and others on similar topics from O'Reilly and other publishers, sign up for free at *http://my.safaribooksonline.com*.

How to Contact Us

We have tested and verified the information in this book to the best of our ability, but you may find that features have changed (or even that we have made a few mistakes!). Please let us know about any errors you find, as well as your suggestions for future editions, by writing to:

O'Reilly Media, Inc.
1005 Gravenstein Highway North
Sebastopol, CA 95472
800-998-9938 (in the United States or Canada)
707-829-0515 (international/local)
707-829-0104 (fax)

We have a web page for this book, where we list errata, examples, and any additional information. You can access this page at:

http://www.oreilly.com/catalog/9780596802479

To comment or ask technical questions about this book, send email to:

bookquestions@oreilly.com

For more information about our books, courses, conferences, and news, see our website at *http://www.oreilly.com*.

Find us on Facebook: *http://facebook.com/oreilly*

Follow us on Twitter: *http://twitter.com/oreillymedia*

Watch us on YouTube: *http://www.youtube.com/oreillymedia*

Acknowledgments

Nick Weldin's contribution was invaluable for the completion of this book. It was 90 percent written when Nick came on board—and without his skill and enthusiasm, it would still be 90 percent written. His hands-on experience running Arduino workshops for all levels of users enabled us to make the advice in this book practical for our broad range of readers. Thank you, Nick, for your knowledge and genial collaborative nature.

Simon St. Laurent was the editor at O'Reilly who first expressed interest in this book. And in the end, he is the man who pulled it together. His support and encouragement kept us inspired as we sifted our way through the volumes of material necessary to do the subject justice.

Brian Jepson helped me get started with the writing of this book. His vast knowledge of things Arduino and his concern and expertise for communicating about technology in plain English set a high standard. He was an ideal guiding hand for shaping the book and making technology readily accessible for readers. We also have Brian to thank for the XBee content in Chapter 14.

Audrey Doyle worked tirelessly to stamp out typos and grammatical errors in the manuscript and untangle some of the more convoluted expressions.

Philip Lindsay collaborated on Chapter 15, and his combination of deep technical knowledge and clear understanding of the needs of nontechnical people was essential in making the complex subject of Ethernet accessible.

Mikal Hart wrote recipes covering GPS and software serial. Mikal was the natural choice for this—not only because he wrote the libraries, but also because he is a fluent communicator, an Arduino enthusiast, and a pleasure to collaborate with.

Arduino is possible because of the creativity of the core Arduino development team: Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, and David Mellis. On behalf of all Arduino users, I wish to express our appreciation for their efforts in making this fascinating technology simple and their generosity in making it free.

Special thanks to Alexandra Deschamps-Sonsino, CEO of Tinker London, whose workshops provided important understanding of the needs of users. Thanks also to Peter Knight, who has provided all kinds of clever Arduino solutions as well as the basis of a number of recipes in this book.

On behalf of everyone who has downloaded user-contributed Arduino libraries, I would like to thank the authors who have generously shared their knowledge.

The availability of a wide range of hardware is a large part of what makes Arduino exciting—thanks to the suppliers for stocking and supporting a broad range of great devices. The following were helpful in providing hardware used in the book: SparkFun, Maker Store, Gravitech, and NKC Electronics. Other suppliers that have been helpful

include Modern Device, Liquidware, Adafruit, Makerbot Industries, Mindkits, Oomlout, and SK Pang.

Nick would like to thank Alexandra and Peter at Tinker London, as well as Brock Craft, and especially Daniel Soltis for all the workshops we have done together.

Nick would also like to thank everyone who has assisted at workshops, and participants who asked a "silly" question, as there are no silly questions. Many of those have led to clarifications and corrections in this book.

Nick's final thanks go to his family, Jeanie, Emily, and Finn, who agreed to let him do this over their summer holiday, and of course, much longer after that than they originally thought, and to his parents, Frank and Eva, for bringing him up to take things apart.

Last but not least, I express thanks to the following people:

Joshua Noble for introducing me to O'Reilly. His book, *Programming Interactivity*, is highly recommended for those interested in broadening their knowledge in interactive computing.

Robert Lacy-Thompson for offering advice early on with the book.

Mark Margolis for his support and help as a sounding board in the book's conception and development.

I thank my parents for helping me to see that the creative arts and technology were not distinctive entities and that, when combined, they can lead to extraordinary results.

And finally, this book would not have been started or finished without the support of my wife, Barbara Faden. My grateful appreciation to her for keeping me motivated and for her careful reading and contributions to the manuscript.