

Foreword by Massimo Banzi, co-founder of Arduino

Arduino[®]

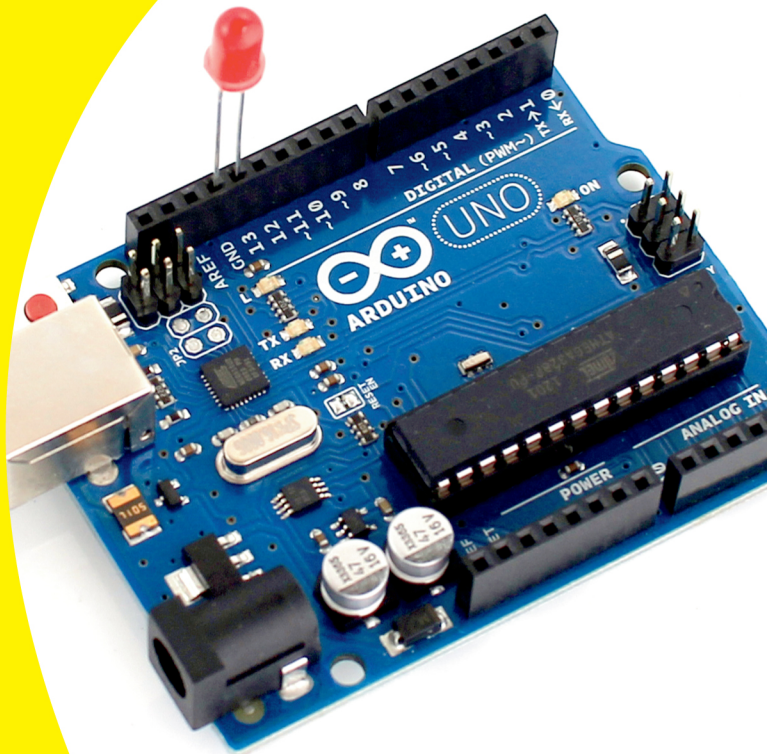
FOR
DUMMIES[®]

A Wiley Brand

Learn to:

- Build exciting interactive projects using Arduino
- Integrate inputs, outputs, and existing hardware into your projects
- Construct robust prototypes to get your project out into the real world
- Communicate between hardware and software using Arduino and Processing

John Nussey



Get More and Do More at Dummies.com®



Start with **FREE** Cheat Sheets

Cheat Sheets include

- Checklists
- Charts
- Common Instructions
- And Other Good Stuff!

To access the Cheat Sheet created specifically for this book, go to
www.dummies.com/cheatsheet/arduino

Get Smart at Dummies.com

Dummies.com makes your life easier with 1,000s of answers on everything from removing wallpaper to using the latest version of Windows.

Check out our

- Videos
- Illustrated Articles
- Step-by-Step Instructions

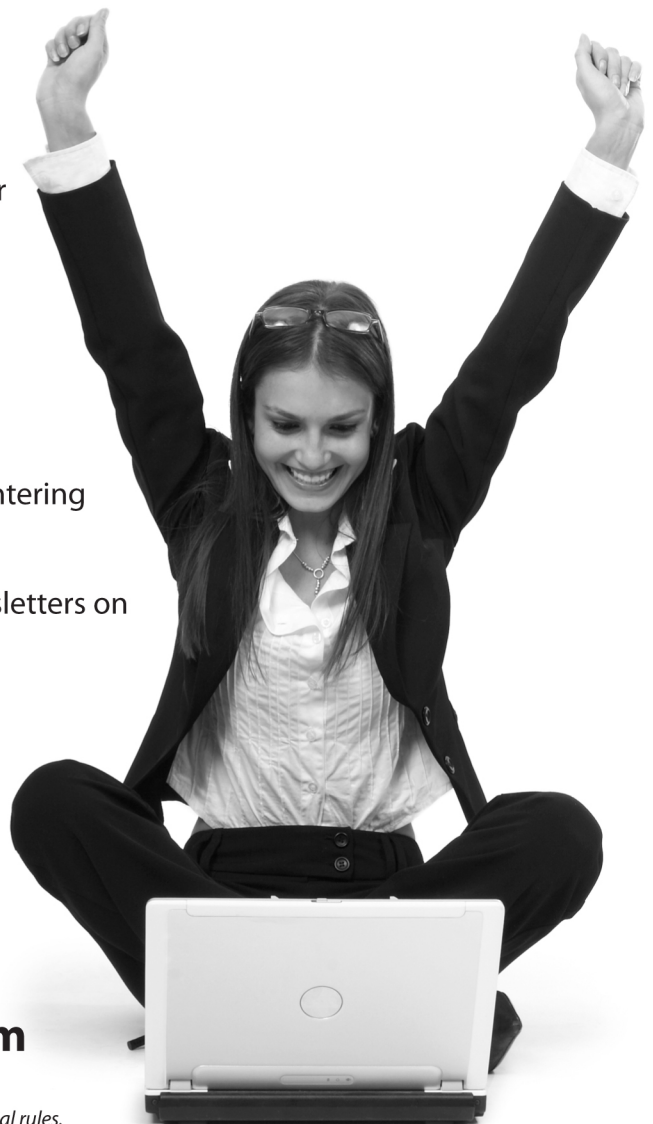
Plus, each month you can win valuable prizes by entering our Dummies.com sweepstakes. *

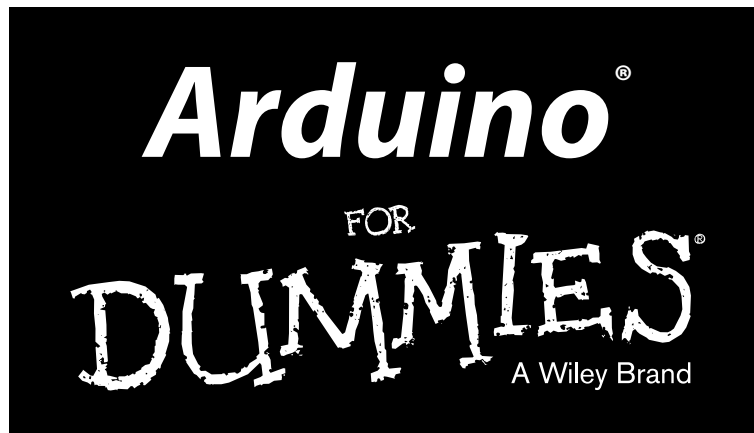
Want a weekly dose of Dummies? Sign up for Newsletters on

- Digital Photography
- Microsoft Windows & Office
- Personal Finance & Investing
- Health & Wellness
- Computing, iPods & Cell Phones
- eBay
- Internet
- Food, Home & Garden

Find out “HOW” at Dummies.com

*Sweepstakes not currently available in all countries; visit Dummies.com for official rules.





by John Nussey



Arduino® For Dummies®

Published by

John Wiley & Sons, Ltd

The Atrium

Southern Gate

Chichester

West Sussex

PO19 8SQ

England

Email (for orders and customer service enquires): cs-books@wiley.co.uk

Visit our home page on www.wiley.com

Copyright © 2013 John Wiley & Sons, Ltd, Chichester, West Sussex, England

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except under the terms of the Copyright, Designs and Patents Act 1988 or under the terms of a licence issued by the Copyright Licensing Agency Ltd., Saffron House, 6-10 Kirby Street, London EC1N 8TS, UK, without the permission in writing of the Publisher. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Ltd, The Atrium, Southern Gate, Chichester, West Sussex, PO19 8SQ, England, or emailed to permreq@wiley.co.uk, or faxed to (44) 1243 770620.

Trademarks: Wiley, For Dummies, the Dummies Man logo, A Reference for the Rest of Us!, The Dummies Way, Dummies Daily, The Fun and Easy Way, Dummies.com, Making Everything Easier, and related trade dress are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates in the United States and other countries, and may not be used without written permission. Arduino is a registered trademark of Arduino LLC. Arduino drawings and circuit diagrams used throughout the book are based on the Fritzing Arduino drawings. All other trademarks are the property of their respective owners. John Wiley & Sons, Ltd. is not associated with any product or vendor mentioned in this book.

LIMIT OF LIABILITY/DISCLAIMER OF WARRANTY: THE PUBLISHER, THE AUTHOR, AND ANYONE ELSE IN PREPARING THIS WORK MAKE NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE ACCURACY OR COMPLETENESS OF THE CONTENTS OF THIS WORK AND SPECIFICALLY DISCLAIM ALL WARRANTIES, INCLUDING WITHOUT LIMITATION WARRANTIES OF FITNESS FOR A PARTICULAR PURPOSE. NO WARRANTY MAY BE CREATED OR EXTENDED BY SALES OR PROMOTIONAL MATERIALS. THE ADVICE AND STRATEGIES CONTAINED HEREIN MAY NOT BE SUITABLE FOR EVERY SITUATION. THIS WORK IS SOLD WITH THE UNDERSTANDING THAT THE PUBLISHER IS NOT ENGAGED IN RENDERING LEGAL, ACCOUNTING, OR OTHER PROFESSIONAL SERVICES. IF PROFESSIONAL ASSISTANCE IS REQUIRED, THE SERVICES OF A COMPETENT PROFESSIONAL PERSON SHOULD BE SOUGHT. NEITHER THE PUBLISHER NOR THE AUTHOR SHALL BE LIABLE FOR DAMAGES ARISING HEREFROM. THE FACT THAT AN ORGANIZATION OR WEBSITE IS REFERRED TO IN THIS WORK AS A CITATION AND/OR A POTENTIAL SOURCE OF FURTHER INFORMATION DOES NOT MEAN THAT THE AUTHOR OR THE PUBLISHER ENDORSES THE INFORMATION THE ORGANIZATION OR WEBSITE MAY PROVIDE OR RECOMMENDATIONS IT MAY MAKE. FURTHER, READERS SHOULD BE AWARE THAT INTERNET WEBSITES LISTED IN THIS WORK MAY HAVE CHANGED OR DISAPPEARED BETWEEN WHEN THIS WORK WAS WRITTEN AND WHEN IT IS READ.

For general information on our other products and services, please contact our Customer Care Department within the U.S. at 877-762-2974, outside the U.S. at 317-572-3993, or fax 317-572-4002.

For technical support, please visit www.wiley.com/techsupport.

Wiley also publishes its books in a variety of electronic formats and by print-on-demand. Some content that appears in standard print versions of this book may not be available in other formats. For more information about Wiley products, visit us at www.wiley.com.

British Library Cataloguing in Publication Data: A catalogue record for this book is available from the British Library.

ISBN 978-1-118-44637-9 (paperback); ISBN 978-1-118-44643-0 (ebook); 978-1-118-44644-7 (ebook); 978-1-118-44642-3 (ebook)

Printed and bound in the United States by Bind-Rite

10 9 8 7 6 5 4 3 2



About the Author

John Nussey is a creative technologist based in London. His work involves using technology in new and interesting ways and covers many areas, including physical computing, creative coding, interaction design, and product prototyping.

During his career, he has worked on many varied projects with clients such as ARUP, the BBC, the Museum of Science and Industry, the National Maritime Museum, Nokia, and the Southbank Centre.

He is a proud advocate of the Arduino prototyping platform and has taught the craft of interaction design and prototyping to people of all ages, competencies, and abilities at a variety of establishments, including Goldsmiths College, the Bartlett School of Architecture, the Royal College of Art, and OneDotZero.

Dedication

To Avril, the love of my life (and only person I trust with a soldering iron), for providing encouragement when I was writing and distraction when I wished I wasn't; to Roanne and Oliver for our stimulating literary chats; to Craig for helping me bridge the trans-Atlantic gap; to all the guys and girls at Kin for still feigning interest and asking "How's the book going?" after 6 months; and to Alexandra for all the guidance that led me to such an enjoyable and inspiring career.

Author's Acknowledgments

John would like to thank the folks at Wiley, especially Craig Smith for always being upbeat and his gentle reminders, and Susan Christophersen for her hard work and support throughout.

Many thanks to Andy Huntington for his excellent technical editing and occasional humorous comments that helped me through the long nights.

A big thanks to all my friends, family, and loved ones for their encouragement and enthusiasm. I love making things, and I hope that this book inspires you to make things of your own and find the same enjoyment I have.

Special thanks to Thomas Harding, Andrew Tune and Brian Jones for their vigilance. Your comments and corrections are invaluable to this book and aspiring Arduin-ists all over the world. I salute you!

Publisher's Acknowledgments

We're proud of this book; please send us your comments at <http://dummies.custhelp.com>. For other comments, please contact our Customer Care Department within the U.S. at 877-762-2974, outside the U.S. at 317-572-3993, or fax 317-572-4002.

Some of the people who helped bring this book to market include the following:

Acquisitions, Editorial

Project and Copy Editor:

Susan Christophersen

Executive Commissioning Editor: Craig Smith

Development Editors: Susan Christophersen,
Susannah Gardner

Technical Editor: Andy Huntington

Editorial Manager: Jodi Jensen

Senior Project Editor: Sara Schlaer

Editorial Assistant: Leslie Saxman

Cover Photo: © John Nussey

Cartoons: Rich Tennant (www.the5thwave.com)

Marketing

Associate Marketing Director: Louise Breinholt

Marketing Manager: Lorna Mein

Senior Marketing Executive: Kate Parrett

Composition Services

Senior Project Coordinator: Kristie Rees

Layout and Graphics: Melanee Habig,
Joyce Haughey, Andrea Hornberger,
Christin Swinford

Proofreader: Wordsmith Editorial

Indexer: Potomac Indexing, LLC

UK Tech Publishing

Michelle Leete, VP Consumer and Technology Publishing Director

Martin Tribe, Associate Director—Book Content Management

Chris Webb, Associate Publisher

Publishing and Editorial for Technology Dummies

Richard Swadley, Vice President and Executive Group Publisher

Andy Cummings, Vice President and Publisher

Mary Bednarek, Executive Acquisitions Director

Mary C. Corder, Editorial Director

Publishing for Consumer Dummies

Kathleen Nebenhaus, Vice President and Executive Publisher

Composition Services

Debbie Stailey, Director of Composition Services

Contents at a Glance

Foreword	xvii
Introduction	1
Part I: Getting to Know Arduino	5
Chapter 1: What Is Arduino and Where Did It Come From?	7
Chapter 2: Finding Your Board and Your Way Around It	17
Chapter 3: Downloading and Installing Arduino	33
Chapter 4: Blinking an LED	41
Part II: Getting Physical with Arduino	61
Chapter 5: Tools of the Trade	63
Chapter 6: A Primer on Electricity and Circuitry	75
Chapter 7: Basic Sketches: Inputs, Outputs, and Communication	91
Chapter 8: More Basic Sketches: Motion and Sound	123
Part III: Building on the Basics	161
Chapter 9: Learning by Example	163
Chapter 10: Soldering On	179
Chapter 11: Getting Clever with Code	209
Chapter 12: Common Sense with Common Sensors	241
Part IV: Unlocking Your Arduino's Potential	287
Chapter 13: Becoming a Specialist with Shields and Libraries	289
Chapter 14: Sensing More Inputs and Controlling More Outputs	315
Chapter 15: Multiplying Your Outputs with I ² C	339
Part V: Sussing Out Software	357
Chapter 16: Getting to Know Processing	359
Chapter 17: Processing the Physical World	359

<i>Part VI: The Part of Tens</i>	<i>407</i>
Chapter 18: Ten Places to Learn More about Arduino	409
Chapter 19: Ten Great Shops to Know	413
Chapter 20: Ten Places to Find Parts and Components.....	417
<i>Index</i>	<i>421</i>
 <i>Bonus Chapter: Hacking Other Hardware On the</i>	
<i>Companion Website at www.dummies.com/go/arduinoofd</i>	

Table of Contents

Foreword.....	<i>xvii</i>
----------------------	--------------------

Introduction	1
---------------------------	----------

About This Book	1
Foolish Assumptions	2
How This Book Is Organized	2
Part I: Getting to Know Arduino	3
Part II: Getting Physical with Arduino	3
Part III: Building on the Basics	3
Part IV: Unlocking Your Arduino's Potential	3
Part V: Sussing Out Software	3
Part VI: The Part of Tens	4
Icons Used In This Book	4
Where to Go from Here	4

<i>Part 1: Getting to Know Arduino</i>	<i>5</i>
---	-----------------

Chapter 1: What Is Arduino and Where Did It Come From?	7
---	----------

Where Did Arduino Come From?	8
Learning by Doing	11
Patching	11
Hacking	12
Circuit bending	13
Electronics	14
Inputs	15
Outputs	15
Open Source	15

Chapter 2: Finding Your Board and Your Way Around It.	17
---	-----------

Getting to Know the Arduino Uno R3	18
The Brains: ATmega328 microcontroller chip	19
Header sockets	20
Digital pins	21
Analog in pins	21
What about analog out?	22
Power pins	22
USB socket	22
External power jack	22
Reset button	24



Discovering Other Arduino Boards.....	24
Official Arduino boards.....	24
Contributed (Approved) Arduinos.....	26
Shopping for Arduino.....	27
Official Arduino Store.....	28
Distributors in the United Kingdom.....	28
Distributors in the United States.....	28
Amazon.....	28
Electronics distributors.....	29
Kitted Out: Starting with a Beginner's Kit.....	29
Preparing a Workspace.....	32

Chapter 3: Downloading and Installing Arduino 33

Installing Arduino.....	33
Installing Arduino for Windows.....	34
Installing Arduino for Mac OS X.....	37
Installing Arduino for Linux.....	39
Surveying the Arduino Environment.....	39

Chapter 4: Blinking an LED 41

Working with Your First Arduino Sketch.....	41
Finding the Blink Sketch.....	42
Identifying your board.....	43
Configuring the software.....	45
Uploading the sketch.....	47
Congratulate yourself!.....	49
What just happened?.....	50
Looking Closer at the Sketch.....	50
Comments.....	51
Declarations.....	52
Variables.....	52
Functions.....	53
Setup.....	54
Loop.....	56
Blinking Brighter.....	57
Tweaking the Sketch.....	59

Part 11: Getting Physical with Arduino 61

Chapter 5: Tools of the Trade 63

Finding the Right Tools for the Job.....	63
Breadboard.....	64
Jump wires.....	66
Needle-nose pliers.....	67
Multimeter.....	68

Using the Multimeter to Measure Voltage, Current, and Resistance	70
Measuring voltage (in volts) in a circuit	70
Measuring current (in amps) in a circuit	71
Measuring resistance (in ohms) of a resistor	72
Measuring resistance (in ohms) of a variable resistor	72
Checking the continuity (in beeps) of your circuit	73
Chapter 6: A Primer on Electricity and Circuitry	75
Understanding Electricity.....	75
Using Equations to Build Your Circuits	77
Ohm's Law	77
Calculating power	80
Joule's Law.....	80
Working with Circuit Diagrams.....	82
A simple circuit diagram.....	82
Using a circuit diagram with an Arduino	84
Color Coding	85
Datasheets.....	86
Resistor Color Charts.....	87
Chapter 7: Basic Sketches: Inputs, Outputs, and Communication . .	91
Uploading a Sketch.....	91
Using Pulse Width Modulation (PWM)	92
The LED Fade Sketch.....	93
Understanding the fade sketch	97
Tweaking the fade sketch	98
The Button Sketch	100
Understanding the Button sketch.....	104
Tweaking the Button sketch.....	105
The AnalogInput Sketch.....	106
Understanding the AnalogInput sketch	110
Tweaking the AnalogInput sketch	111
Talking Serial.....	112
The DigitalReadSerial Sketch	112
Understanding the DigitalReadSerial sketch.....	115
The AnalogInOutSerial Sketch.....	116
Understanding the AnalogInOutSerial sketch	120
Chapter 8: More Basic Sketches: Motion and Sound	123
Working with Electric Motors	123
Discovering Diodes.....	125
Spinning a DC Motor	125
The Motor sketch.....	126
Understanding the Motor sketch.....	129
Changing the Speed of Your Motor	130
The MotorSpeed sketch	130
Understanding the MotorSpeed sketch	131

Controlling the Speed of Your Motor.....	132
The MotorControl sketch	132
Understanding the MotorControl Sketch	135
Tweaking the MotorControl sketch.....	135
Getting to Know Servo Motors.....	136
Creating Sweeping Movements.....	137
The Sweep sketch	137
Understanding the Sweep sketch	140
Controlling Your Servo	142
The Knob sketch	142
Understanding the Knob sketch	145
Making Noises	146
Piezo buzzer	146
The toneMelody sketch.....	147
Understanding the sketch.....	153
Making an Instrument	156
The PitchFollower sketch	156
Understanding the sketch.....	159

Part III: Building on the Basics 161

Chapter 9: Learning by Example163

Skube.....	163
How it works.....	164
Further reading	165
Chorus	165
How it works.....	166
Further reading	167
Push Snowboarding.....	167
How it works.....	168
Further reading	169
Baker Tweet.....	169
How it works.....	170
Further reading	171
The National Maritime Museum's Compass Lounge and Compass Card...171	
How it works.....	172
Further reading	174
The Good Night Lamp.....	174
How it works.....	175
Further reading	175
Little Printer	175
How it works.....	176
Further reading	177
Flap to Freedom.....	177
How it works.....	178
Further reading	178

Chapter 10: Soldering On179

Understanding Soldering	179
Gathering What You Need for Soldering	180
Creating a workspace	180
Choosing a soldering iron	181
Solder	185
Third hand (helping hand)	186
Adhesive putty	187
Wire cutters	188
Wire strippers	188
Needle-nosed pliers	189
Multimeter	189
Solder sucker	190
Solder wick	190
Equipment wire	191
Staying Safe while Soldering	192
Handling your soldering iron	192
Keeping your eyes protected	193
Working in a ventilated environment	193
Cleaning your iron	193
Don't eat the solder!	193
Assembling a Shield	194
Laying out all the pieces of the circuit	195
Assembly	196
Header pins	196
Acquiring Your Soldering Technique	197
Building Your Circuit	201
Knowing your circuit	201
Laying out your circuit	202
Preparing your wire	202
Soldering your circuit	203
Cleaning up	203
Testing your shield	205
Packaging Your Project	205
Enclosures	205
Wiring	206
Securing the board and other elements	207

Chapter 11: Getting Clever with Code209

Blinking Better	209
Setting up the BlinkWithoutDelay sketch	211
Understanding the BlinkWithoutDelay sketch	214
Taking the Bounce Out of Your Button	216
Setting up the Debounce sketch	216
Understanding the Debounce sketch	219
Making a Better Button	221
Setting up the StateChangeDetection sketch	221
Understanding the StateChangeDetection sketch	225

Smoothing Your Sensors	227
Setting up the Smoothing sketch	228
Understanding the Smoothing sketch.....	231
Calibrating Your Inputs	233
Setting up the Calibration sketch	233
Understanding the Calibration sketch	237

Chapter 12: Common Sense with Common Sensors241

Making Buttons Easier	242
Implementing the DigitalInputPullup sketch.....	243
Understanding the DigitalInputPullup sketch.....	246
Exploring Piezo Sensors	247
Implementing the Knock sketch	248
Understanding the Knock sketch.....	251
Utilizing Pressure, Force, and Load Sensors.....	252
Implementing the toneKeyboard sketch.....	254
Understanding the toneKeyboard sketch.....	257
Sensing with Style.....	258
Implementing the CapPinSketch sketch	261
Understanding the CapPinSketch sketch.....	264
Tripping Along with Lasers	267
Implementing the AnalogInOutSerial sketch.....	268
Understanding the AnalogInOutSerial sketch	271
Detecting Movement	271
Implementing the DigitalReadSerial sketch.....	273
Understanding the DigitalReadSerial sketch	276
Measuring Distance	277
Implementing the MaxSonar sketch	278
Understanding the MaxSonar sketch	281
Testing, Testing . . . Can Anybody Hear This?	282
Implementing the AnalogInOutSerial sketch	283
Understanding the AnalogInOutSerial sketch	286

Part IV: Unlocking Your Arduino's Potential 287

Chapter 13: Becoming a Specialist with Shields and Libraries . . . 289

Looking at Shields	289
Considering combinations.....	290
Reviewing the field.....	291
Staying current.....	308
Browsing the Libraries.....	309
Reviewing the standard libraries	309
Installing additional libraries	311
Obtaining contributed libraries	313

Chapter 14: Sensing More Inputs and Controlling More Outputs . . . 315

Controlling Multiple LEDs	315
Implementing the AnalogWriteMega sketch	318
Understanding the AnalogWriteMega Sketch	322
Tweaking the AnalogWriteMega sketch.....	324
Controlling Lots of LEDs by Shifting Out.....	327
Implementing the shiftOutCode, Hello World sketch	329
Understanding the shiftOutCode, Hello World sketch.....	333
Tweaking the shiftOutCode, Hello World sketch.....	334
Doing more with the same circuit	337

Chapter 15: Multiplying Your Outputs with I²C 339

What Is I ² C?.....	339
Assembling the I ² C PWM/Servo Driver	341
Using the I ² C PWM/Servo Driver.....	343
Understanding the I ² C PWM/Servo Driver Sketch.....	350
Buying Servo Motors.....	353
Other Uses for I ² C.....	355

Part V: Sussing Out Software..... 357

Chapter 16: Getting to Know Processing 359

Looking Under the Hood.....	360
Installing Processing.....	362
Taking a look at Processing.....	364
Trying Your First Processing Sketch.....	365
Drawing shapes	368
Changing color and opacity.....	372
Playing with interaction	374

Chapter 17: Processing the Physical World 359

Making a Virtual Button	359
Setting up the Arduino code.....	360
Setting up the Processing code.....	362
Understanding the Processing PhysicalPixel sketch	364
Understanding the Arduino Physical Pixel sketch	367
Drawing a Graph	369
Setting up the Arduino code.....	371
Setting up the Processing code.....	372
Understanding the Arduino Graph sketch.....	374
Understanding the Processing Graph sketch.....	374
Sending Multiple Signals.....	378
Setting up the Arduino code.....	380
Setting up the Processing code.....	381
Understanding the Arduino SerialCallResponse sketch	383
Understanding the Processing SerialCallResponse sketch	385

Part VI: The Part of Tens 407**Chapter 18: Ten Places to Learn More about Arduino.409**

Arduino Blog	409
Hack a Day.....	409
SparkFun.....	410
MAKE	410
Adafruit.....	410
Bldr.....	410
Instructables	411
YouTube	411
Hackerspaces	411
Forum	411
Friends, Colleagues, and Workshops	412

Chapter 19: Ten Great Shops to Know413

Shops in the United Kingdom	413
SK Pang.....	413
Technobots.....	414
Proto-PIC	414
Oomlout	414
RoboSavvy	414
Active Robots	415
Shops around the World.....	415
Adafruit (U.S.).....	415
Arduino Store (Italy).....	415
Seeed Studio (China)	416
SparkFun (U.S.).....	416

Chapter 20: Ten Places to Find Parts and Components417

RS Components (World).....	417
Farnell (World).....	417
Rapid (World)	418
Digi-Key (World)	418
eBay (World)	418
Maplin (U.K.)	418
RadioShack (U.S.)	419
Ultraleds (U.K.)	419
EnvironmentalLights.com (U.S.).....	419
Skip/Dumpster Diving (World)	419

Index 421

**Bonus Chapter: Hacking Other Hardware On the
Companion Website at www.dummies.com/go/arduino4d**

Foreword

The moment a *For Dummies* book comes out, it's definitely a milestone in the history of a product.

Programming embedded computers used to be a very difficult task, reserved only to experienced engineers willing to master the obscure assembly language. In recent years, however, many platforms have tried to make this task simpler and more accessible to everyday people. Arduino is one of the latest attempts at making technology less scary and more creative.

With John, this book's author, we watched this creative tool being adopted by designers and artists in London, making its way into many memorable projects. Now Arduino has escaped the lab of Arts & Design and spread like a virus, becoming the tool of choice for all kinds of people who have great ideas they want to realize.

I'm really glad that John decided to write this book, because he's an early user of the Arduino platform from back in the days when it was still quite experimental. Having taught Arduino classes for many years, he has the ability to introduce the subject to all audiences.

Any newcomer to Arduino will, with the right tools and teaching — such as those found in this book — show true genius in no time.

Massimo Banzi

Introduction

Arduino is a tool, a community, and a way of thinking that is affecting how we use and understand technology. It has rekindled a love and understanding for electronics for many people, including myself, who felt that electronics was something that they had left behind at school.

Arduino is tiny circuit board that has huge potential. It can be used to blink a Morse-code signal using a single LED or to control every light in a building, depending on how far you take it. Its capabilities are limited only by your imagination.

Arduino is also providing a new, practical approach to technical education, lowering the entry level for those wanting to use electronics to complete small projects and, I hope, encouraging you to read further to take on big ones.

A huge and ever-growing community of Arduino-ists has emerged — users and developers who learn from each other and contribute to the open source philosophy by sharing the details of their projects. Arduino-ists and their supporters with their open source attitude are responsible for the huge popularity of Arduino.

Arduino is more than just a “bit of kit”; it’s a tool. A piece of technology that makes understanding and using today’s technology easier.

So if the prospect of understanding the limitless possibilities of technology doesn’t sound interesting to you, please put this book down and back away.

Otherwise, read on!

About This Book

This is a technical book, but it’s not for technical people only. Arduino is designed to be usable by anyone, whether they’re technical, creative, crafty, or just curious. All you need is an open mind or a problem to fix and you’ll soon find ways that using Arduino can benefit you.

Arduino has rekindled my love of electronics and opened many avenues for my career. I wrote this book to share that experience. When I first went to an Arduino workshop, I had no experience in programming and could only vaguely remember which end of a soldering iron to hold (don’t worry, I cover

soldering, too). Now the mainstay of my work involves building interactive installations, prototyping products, and generally finding new ways to play with technology using Arduino.

I think it is an excellent platform that lowers the entry level into electronics and coding, allowing people who may not have had the attention span or interest at school to dive straight into the areas that interest them and explore them from there.

Foolish Assumptions

This book assumes nothing about your technical knowledge. Arduino is an easy-to-use platform for learning about electronics and programming. It is for people from all walks of life, whether you're a designer, an artist, or a hobbyist.

It can also be a great platform for people who are already technical. Maybe you've done a bit of coding but want to bring your projects into the physical world in some way, or maybe you've worked with electronics and want to see what Arduino can bring to the table.

But whoever you are, you'll find that Arduino has great potential. It's really up to you to decide what to make of it.

This book starts on the most basic level to get you started with using and understanding Arduino. At times throughout the book, I may refer to a number of very technical things that will, like anything, take time to understand. I guide you through all the basics and then on to more advanced activities.

Much of what is in this book is based on my learning and teaching experiences. I learned all about Arduino from scratch, but have always found that the best way to learn is in practice, by making your own projects. The key is to learn the basics that I cover in this book and then build on that knowledge by thinking about how you can apply it to solve problems, create things, or just entertain yourself.

How This Book Is Organized

Arduino For Dummies is organized in a way that allows you to jump around the book as you like. If you've dabbled in Arduino before, you might want to skip to the later chapters, or if you've forgotten some of the basics, consider starting at the beginning.

Part I: Getting to Know Arduino

In Part I, I introduce you to Arduino, outlining a variety of other practices and circumstances that created a need for Arduino and that have influenced its development. Then I look at Arduino in more detail, both as a physical board and software environment, and I walk you through uploading your first sketch.

Part II: Getting Physical with Arduino

In this part, you find out how to do some basic prototyping using breadboards and other components to give your Arduino more reach into the physical world. Using just a few simple components, you can explore a variety of applications for Arduino and form a base on which you can build your own projects. The chapters in this part cover a variety of inputs and outputs, including light, motion, and sound that you can build on and combine to form your own projects.

Part III: Building on the Basics

After you have covered the basics, you'll be itching to do more. In Part III, I tell you about some real-world projects and how they work. You find out how to solder your own circuit board to get your project out into the world for others to see. You also learn how to choose the correct sensor for the job and how to use code to fine-tune or change the behavior of your circuits.

Part IV: Unlocking Your Arduino's Potential

This part pushes the possibilities of your Arduino project further. You learn about using shields to add specific functionality to your Arduino, using hardware and techniques to allow you project to grow, and hacking existing hardware. You also find out how to communicate with Processing, Arduino's sister project, to combine open source hardware with software.

Part V: Sussing Out Software

If you work through the book to this part, you should have a good understanding of how you can use electronics and hardware in your own projects. In this part, you learn how to combine this knowledge of the physical world with the digital world of software. I introduce you to a few open source

programming environments and then more specifically to Processing, which is a digital sketchbook that you can use for a huge variety of applications to enhance your Arduino project.

Part VI: The Part of Tens

The Part of Tens is a *For Dummies* standard that breaks down useful information into groups of ten bite-sized chunks. This part covers where to learn more about Arduino, where to shop for Arduino-specific parts, and where to shop for electronics in general.

Icons Used in This Book



Arduino For Dummies uses icons to highlight important points for you. Keep an eye out for these:

This icon highlights a bit of helpful information. That info may be a technique to help you complete a project more easily or the answer to common problems.



Arduinos aren't dangerous on their own; indeed, they're made to be extremely safe and easy to use. But if they are used in a circuit without proper planning as well as care and attention, they can damage your circuit, your computer, and yourself. When you see a Warning icon, please take special note.



There are often points that must be considered before proceeding with a task. I use Remember icons to remind you of such points.



Some information is more technical than others and is not for the faint hearted. The joy of Arduino is that you don't need to fully understand the technical details immediately. You can skip anything that's marked with this icon if it's more complicated than you want to deal with at the moment; you can always return to it when you're ready.

Where to Go from Here

If you're uncertain about where to start, I suggest the beginning. By the end of Chapter 2, you'll have acquired a simple understanding of Arduino and will know where you can get a kit to continue learning.

If you've used Arduino before, you may want to jump straight to Chapter 4 to cover the basics again, or head straight to the area that interests you.

Part I

Getting to Know Arduino

The 5th Wave

By Rich Tennant



“So I guess you forgot to tell me to strip out the components before drilling for blowholes.”

In this part . . .

So what is an Arduino, anyway? In the chapters ahead, you find out all about this little blue circuit board, how it came into being, and what it can be used for. After a brief introduction, I talk you through all the things you need to get started with Arduino and where to get them. Next, you learn how to wield the awesome power of an LED, blinking it on command with a few simple lines of code.

Chapter 1

What Is Arduino and Where Did It Come From?

In This Chapter

- ▶ Discovering Arduino
 - ▶ Learning where Arduino came from and why it's so important
 - ▶ Introducing the basic principles
-

Arduino is made up of both hardware and software.

The Arduino board is a printed circuit board (PCB) that is specifically designed to use a microcontroller chip as well as other input and outputs. It also has many other electronic components that are needed for the microcontroller to function or to extend its capabilities.

Microcontrollers are small computers contained within a single, integrated circuit or computer chip, and they are an excellent way to program and control electronics. Many devices, referred to as microcontroller boards, have a microcontroller chip and other useful connectors and components that allow a user to attach inputs and outputs. Some examples of devices with microcontroller boards are the Wiring board, the PIC, and the Basic Stamp.

You write code in the Arduino software to tell the microcontroller what to do. For example, by writing a line of code, you can tell an LED to blink on and off. If you connect a pushbutton and add another line of code, you can tell the LED to turn on only when the button is pressed. Next, you may want to tell the LED to blink only when the pushbutton is held down. In this way, you can quickly build a behavior for a system that would be difficult to achieve without a microcontroller.

Similarly to a conventional computer, an Arduino can perform a multitude of functions, but it's not much use on its own. It requires other inputs or outputs to make it useful. These inputs and outputs allow a computer to sense objects in the world and to affect the world.

Before you move forward, it might help you to understand a bit of the history of Arduino.

Where Did Arduino Come From?

Arduino started its life in Italy, at Interaction Design Institute Ivrea (IDII), a graduate school for interaction design. This is a specific school of design education that focuses on how people interact with digital products, systems, and environments and how they in turn influence us.

The term *interaction design* was coined by Bill Verplank and Bill Moggridge in the mid-1980s. The sketch in Figure 1-1 by Verplank illustrates the basic premise of interaction design. This diagram is an excellent illustration of how the process of interaction works: If you do something, you feel a change, and from that you can know something about the world.

Although it is a general principle, interaction design more commonly refers to how we interact with conventional computers by using peripherals, such as mice, keyboards, and touchscreens, to navigate a digital environment that is graphically displayed on a screen.

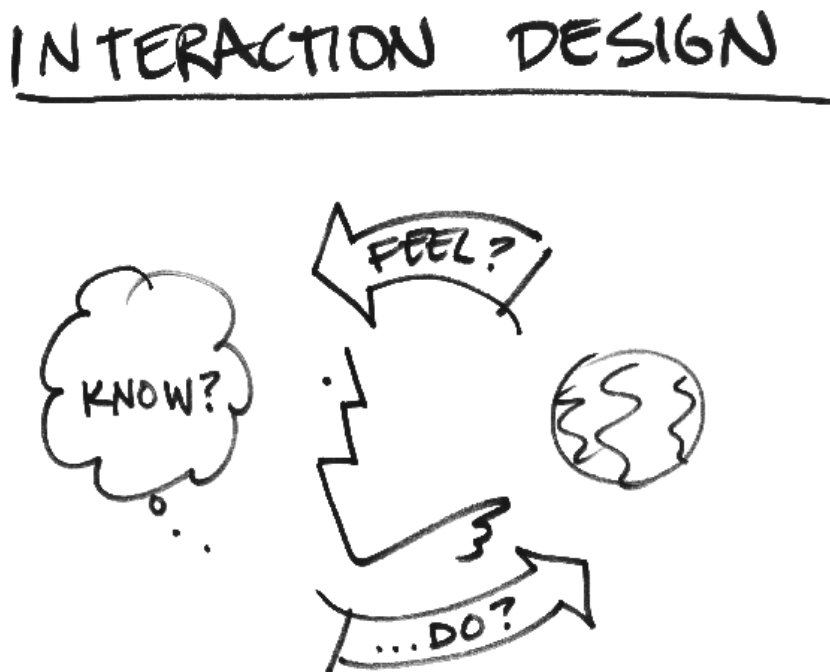


Figure 1-1:
The principle of
interaction
design, illus-
trated by Bill
Verplank.

Courtesy of Bill Verplank

There is another avenue, referred to as physical computing, which is about extending the range of these computer programs, software, or systems. Through electronics, computers can sense more about the world and have a physical impact on the world themselves.

Both of these areas — interaction design and physical computing — require prototypes to fully understand and explore the interactions, which presented a hurdle for nontechnical design students.

In 2001, a project called Processing that was started by Casey Reas and Benjamin Fry aimed to get nonprogrammers into programming by making it quick and easy to produce onscreen visualizations and graphics. The project gave the user a digital sketchbook on which to try ideas and experiment with a very small investment of time. This project in turn inspired a similar project for experimenting in the physical world.

Building on the same principles as Processing, in 2003 Hernando Barragán started developing a microcontroller board called Wiring. This board was the predecessor to Arduino.

In common with the Processing project, the Wiring project also aimed to involve artists, designers, and other nontechnical people, but Wiring was designed to get people into electronics rather than programming. The Wiring board (shown in Figure 1-2) was less expensive than some other microcontrollers, such as the PIC and the Basic Stamp, but it was still a sizable investment for students to make.

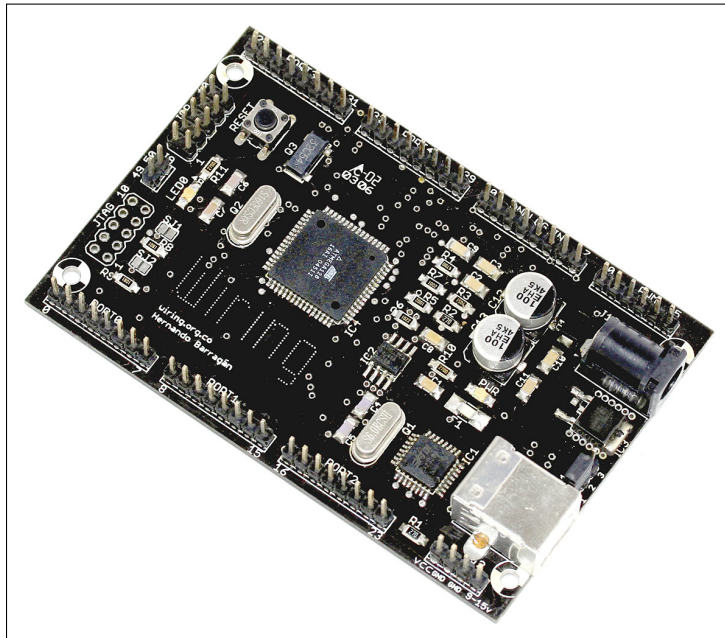


Figure 1-2:
An early
Wiring
board.

In 2005, the Arduino project began in response to the need for affordable and easy-to-use devices for Interaction Design students to use in their projects. It is said that Massimo Banzi and David Cuartielles named the project after Arduin of Ivrea, an Italian king, but I've heard from reliable sources that it also happens to be the name of the local pub near the university, which may have been of more significance to the project.

The Arduino project drew from many of the experiences of both Wiring and Processing. For example, an obvious influence from Processing is the *graphic user interface* (GUI) that is used in the Arduino software. This GUI was initially “borrowed” from Processing, and even though it still looks similar, it has since been refined to be more specific to Arduino. I cover the Arduino interface in more depth in Chapter 4.

Arduino also kept the naming convention from Processing, naming its programs *sketches*. In the same way that Processing gives people a digital sketchbook to create and test programs quickly, Arduino gives people a way to sketch out their hardware ideas as well. Throughout this book, I show many sketches that allow your Arduino to perform a huge variety of tasks. By using and editing the example sketches in this book, you can quickly build up your understanding of how they work and will be writing your own in no time. Each sketch is followed with a line-by-line explanation of how it works to ensure that no stone is left unturned.

The Arduino board, shown in Figure 1-3, was made to be more robust and forgiving than Wiring or other earlier microcontrollers. It was not uncommon for students and professions, especially those from a design or arts background, to break their microcontroller within minutes of using it, simply by getting the wires the wrong way around. This fragility was a huge problem, not only financially but also for the success of the boards outside technical circles.

It is also possible to change the microcontroller chip on an Arduino, so if it is damaged, you can just replace the chip rather than the whole board.

Another important difference between Arduino and other microcontroller boards is the cost. In 2006, another popular microcontroller, the Basic Stamp, cost nearly four times as much (<http://blog.makezine.com/2006/09/25/arduino-the-basic-stamp-k/>) as an Arduino, and even today, a Wiring board still costs nearly double the price of an Arduino.

In one of my first Arduino workshops, I was told that the price was intended to be affordable for students. The price of a nice meal and a glass of wine at that time was about 30 euros, so if you had a project deadline, you could choose to skip a nice meal that week and make your project instead.

The range of Arduino boards on the market is a lot bigger than it was back in 2006. In Chapter 2, you learn about just a few of the most useful Arduino and Arduino-compatible boards and how they differ to provide you with a variety