TECHNOLOGY IN ACTION™

Arduino Robotics

- STEP-BY-STEP INSTRUCTIONS FOR BOTH BASIC AND ADVANCED ROBOTS
- DISCOVER HOW TO MAKE YOUR OWN PCBS
- LEARN HOW TO WORK WITH DIFFERENT TYPES OF SENSORS, MOTORS, AND CONTROLLERS
- BUILD, TEST, AND SHOW OFF YOUR ROBOT COLLECTION!

John-David Warren,
Josh Adams, and Harald Molle

Arduino Robotics

Â Â Â

John-David Warren
Josh Adams
Harald Molle

Apress®

Arduino Robotics

Copyright © 2011 by John-David Warren, Josh Adams, and Harald Molle

All rights reserved. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the copyright owner and the publisher.

ISBN-13 (pbk): 978-1-4302-3183-7

ISBN-13 (electronic): 978-1-4302-3184-4

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

President and Publisher: Paul Manning
Lead Editor: Michelle Lowman and James Markham
Technical Reviewer: Guilherme Martins and Josh Adams
Editorial Board: Steve Anglin, Mark Beckner, Ewan Buckingham, Gary Cornell, Jonathan Gennick, Jonathan Hassell, Michelle Lowman, Matthew Moodie, Jeff Olson, Jeffrey Pepper, Frank Pohlmann, Douglas Pundick, Ben Renow-Clarke, Dominic Shakeshaft, Matt Wade, Tom Welsh
Coordinating Editor: Anita Castro
Copy Editor: Ginny Munroe and Mary Ann Fugate
Compositor: Bytheway Publishing Services
Indexer: BIM Indexing & Proofreading Services
Artist: April Milne
Cover Designer: Anna Ishchenko

Distributed to the book trade worldwide by Springer Science+Business Media, LLC., 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com.

For information on translations, please e-mail rights@apress.com, or visit www.apress.com.

Apress and friends of ED books may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Special Bulk Sales–eBook Licensing web page at www.apress.com/info/bulksales.

The information in this book is distributed on an “as is” basis, without warranty. Although every precaution has been taken in the preparation of this work, neither the author(s) nor Apress shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in this work.

The source code for this book is available to readers at www.apress.com. You will need to answer questions pertaining to this book in order to successfully download the code.

To my dad, Poppy, and Papa Bert for showing me that even ordinary men can be extraordinary.

John-David Warren

Contents at a Glance

Â About the Authors xix
Â About the Technical Reviewers xxi
Â Acknowledgments xxii
Â Introduction xxiv
Â Chapter 1: Introducing Oracle APEX 1
Â Chapter 1: The Basics 1
Â Chapter 2: Arduino for Robotics 51
Â Chapter 3: Let's Get Moving 83
Â Chapter 4: Linus the Line-Bot 119
Â Chapter 5: Wally the Wall-Bot 169
Â Chapter 6: Making PCBs 203
Â Chapter 7: The Bug-Bot 257
Â Chapter 8: Explorer-Bot 295
Â Chapter 9: RoboBoat 331
Â Chapter 10: Lawn-Bot 400 403
Â Chapter 11: The Seg-Bot 453
Â Chapter 12: The Battle-Bot 513
Â Chapter 13: Alternate Control 563
Â Index 581

Contents

Â **About the Authors** ..... **xix**
Â **About the Technical Reviewers** ..... **xxi**
Â **Acknowledgments** ..... **xxii**
Â **Introduction** ..... **xxiv**
Â **Chapter 1: The Basics** **1**
Electricity 2
Electrical Analogy 3
Electrical Basics 4
Circuits 6
Measuring Electricity 7
Multi-Meters 7
Measuring Voltage 8
Measuring Amperage 9
Measuring Capacitance 9
Measuring Resistance 10
Calculating Resistor Power Using Ohm's Law 11
Oscilloscope 12
Loads 13
Electrical Connections 14
Electronics 15
Semi-Conductors 16
Datasheets 16

Integrated Circuits 17
Through-Hole Components 18
Arduino Primer 20
Arduino Variants 20
Arduino IDE 23
The Sketch 24
Signals 26
Building Circuits 35
Circuit Design 35
Schematics 36
Prototyping 40
Building a Robot 46
Hardware 46
Materials 48
Work Area 48
Summary 49
Â Chapter 2: Arduino for Robotics 51
Interfacing Arduino 51
Relays 52
Transistors 58
Interfacing a Motor-Controller 65
User Control 68
Tethered (Wired) Control 68
Infrared Control (IR) 68
Radio Control Systems 69
Sensor Navigation 72
Contact Sensing 72
Distance and Reflection Sensing 73

Orientation (Positioning) 76
Non-Autonomous Sensors 79
Summary 81
Â Chapter 3: Let's Get Moving 83
Electric Motors 83
Brushed DC Motor (Permanent Magnet Type) 84
Brushless Motors 84
Stepper Motors 85
Gear Motors 86
Servo Motors 86
Linear Actuators 87
Calculating Power 87
Driving 88
Finding the Right Motor 88
The H-Bridge 89
Generating a Brake 90
Implementation 90
H-Bridge ICs 93
Changing PWM Frequencies 93
Back EMF 96
Current Sensing 96
Commercial H-Bridges (Motor-Controllers) 98
Batteries 100
Nickel Cadmium (NiCad) 100
Nickel Metal Hydride (NiMH) 101
Lithium Polymer (LiPo) 101
Lead-Acid 101
Charging 103

Materials 103
Wood 104
Metals 104
Nuts and Bolts 104
Plastics 105
Chain and Sprockets 105
Wheels 105
Summary 105
Â Chapter 4: Linus the Line-Bot 119
Parts List for Linus 120
How Linus Works 122
The Track 123
Building the IR Sensor Board 123
Modifying a Servo for Continuous Rotation 132
Method 1: Direct DC Drive with External Speed Controller 133
Method 2: Servo Pulse Drive with Internal Motor Driving Circuitry 135
Fitting the Drive wheels 137
Making a Frame 141
Making Connections 148
Installing Batteries 150
Install Power Switch 150
Loading the Code 150
Making the Track 156
Testing 157
Add-ons 158
LED Lights 158
Painting 160

Adding Speed Regulator (Potentiometer) 164
Summary 167
Â **Chapter 5: Wally the Wall-Bot 169**
How Wally Works 170
Parts List for Wally 172
The Motor-Controller 173
The High-Side Switches 174
The Low-Side Switches 174
Building the Circuit 176
Building the Frame 182
Installing the Sensors 187
Installing the Battery and Power Switch 190
Installing a Power Switch 191
The Code 192
Code Objectives 193
Summary 201
Â **Chapter 6: Making PCBs 203**
PCB Basics 203
What You Need to Get Started 204
Circuit Design 206
Searching for Open-Source Designs 207
Making Your Own Designs 208
Working with the Schematic Editor 211
Working with the Board Editor 216
Transferring the Design 226
Let's Make an Arduino Clone: the Jduino 227
Making the Transfer 228

Etching....236
Measuring the Solution....237
Etching: Method 1....238
Etching: Method 2....239
Removing the Toner....242
Drilling....245
Soldering....246
Building the Arduino Clone....247
Building the BJT H-Bridge....250
Testing....253
Summary....256
Â **Chapter 7: The Bug-Bot....257**
Reading a Switch with Arduino....258
How the Bug-bot Works....259
Antennae Sensors....259
Bumper Sensors....260
Parts List for the Bug-bot....260
The Motors....261
Modifying the Servo Motors....261
Controlling the Servo Motors....264
Converting the Pulse Value to Degrees....265
Mounting the Wheels to the Servos....266
Building the Frame....267
Marking the PlexiGlass....267
Cutting the PlexiGlass....269
Mounting the Motors....271
Mounting the Caster Wheels....272

Mounting the Arduino 274
Installing the Battery 275
Making the Sensors 277
The Front Antennae Sensors 277
The Rear Bump Sensors 278
Making Wire Connections 281
Loading the Code 283
Creating a Delay 283
Variables 284
The Code 285
Making a Top Hat 292
Summary 294
Â **Chapter 8: Explorer-Bot 295**
How the Explorer-Bot Works 296
R/C Control 296
Powerful Motors 297
Current Sensing 297
Video-Enabled 297
Xbee-Enabled 297
Parts List for the Explorer-Bot 298
Building the Frame 300
Specs 300
Adding Battery Bracket 301
Cut Top Frame Brackets 303
Cut Top Frame Braces 303
Cut and Bend Main Frame Piece 304
Add Crossbar and Mount Caster Wheel 306

Building the Motor Controller 308
Current Sensing and Limiting 309
H-bridge Design 309
Setting Up the Arduino 313
Connecting the H-Bridges 314
Setting Up Xbee 315
Testing the Xbees 317
Adding a Camera 318
Pan and Tilt 319
Make First Bracket 319
Make Second Bracket 321
Loading the Code 321
Summary 329
Â **Chapter 9: RoboBoat 331**
Some Words in Advance 332
Parts List for the RoboBoat 332
Polystyrene Foam 335
Epoxy Resin 336
Gloves 336
Fiberglass Fabric 336
Glue 338
Styrofoam Cutter and Razor Knife 338
Miscellaneous 338
The RoboBoat Design 338
Assembling the Boat 340
The Templates 340
Gluing the Templates on the EPS/XPS Board 342

Cutting Out the Segments 343
Gluing the Segments Together 346
Inserting the Foam Anchors 347
The Coating 348
Applying the Finish 350
The Fins 350
Painting 351
The Deck 351
Completing the Assembly 352
The Propulsion Assembly 356
The Baseplate 357
The Pivot 358
The Tube 359
The Rudder Horns 360
The Motor 360
The Servo 361
The Pushrods 361
Electronics 361
The Heart of the System: The ArduPilot PCB 362
The GPS Module 363
The Electronic Speed Controller (ESC) 364
The Motor 365
The Rudder Servo 365
The Battery Pack 365
Assembling the Electronics 365
Software and Mission Planning 368
GPS Receivers 368
The Software 369

Installing the Software 387
Mission Planning 392
Putting It All Together 396
Integrating the System 397
Ships Ahoy! 399
Troubleshooting 400
The Thrust of the Motor/Propeller Is Not Sufficient 400
The Motor Does Not Start 401
Summary 402
Â Chapter 10: Lawn-Bot 400 403
How the Lawn-bot 400 Works 404
Lawn Mower Deck 405
High-Capacity Batteries 406
Steel Frame 406
Dump-Bucket 406
Pneumatic Tires 406
Headlights 406
Failsafe 407
Tools and Parts List 407
Lawnmower 407
Parts Lists 407
The Wheels 408
Front Caster Wheels 409
Rear Drive Wheels 409
Installing Sprockets 410
The Frame 412
The Drive-Train 420

Motor Mount Brackets 420
Installing the Chain 424
The Motor Controller 427
Buying a Motor Controller 427
Cooling Fans 429
Motor Controller feedback 429
The Arduino 430
Securing Connections for a Bumpy Ride 431
The Failsafe 435
R/C Toggle Switch 436
Power Relays 439
Avoiding an R/C Failsafe 440
Making Connections 441
The Code 442
Adding Cosmetics and Accessories 447
Painting 447
Headlights 448
Dump Bucket 449
Lawnmower Kill-Switch 451
Summary 451
Â Chapter 11: The Seg-Bot 453
How the Seg-Bot Works 455
Inertial Measurement Unit 455
Steering and Gain 455
Engage Switch 455
Parts List for the Seg-Bot 456
Selecting the Right Sensors 457

3.3v Power 459
Accelerometer 459
Gyroscope 461
Gyroscope Versus Accelerometer Summary 463
Filtering the Angle 463
Making the IMU Adapter Board 464
Selecting the Motors 465
Electric Brake Removal 467
Motor Mounting Position 471
Selecting the Motor Controller 472
SoftwareSerial Library 473
Sabertooth Simplified Serial 474
The Batteries 475
Sealed Lead-Acid 475
Charging 476
12v Supply 476
The Frame 477
Frame Design 478
Building the Frame 479
Inputs 483
Steering 483
Gain 483
Engage Switch 483
Level-Start 483
Mounting the Inputs to the Frame 484
Installing the Electronics 488
Soldering the Inputs 491
Wiring the Connections 492

Reviewing the Code 493
The sample_accel() Function 494
The sample_gyro() Function 494
Checking the Angle Readings 496
The calculate_angle() Function 496
The read_pots() Function 497
The auto_level() Function 498
The update_motor_speed() Function 499
The time_stamp() Function 501
The serial_print_stuff() Function 502
The Full Code 503
Testing 509
Summary 511
References 511
Â **Chapter 12: The Battle-Bot 513**
Robotic Combat Is Born 515
Battle-Bot Rules and Regulations 515
No Price Limit! 516
Parts List for the Battle-Bot 517
Input Control 520
Fly Sky CT-6: The $32, 5-ch*, 2.4GHz Radio Alternative 520
The Electronics 523
The Arduino 523
The Motor Controllers 524
The Frame 525
To Buy or Build? 526
Modifying the Wheels 527

Building the Frame 528
The Drive Train 532
Gearing 533
Chain Tensioning Nut 534
Batteries 538
Securing Electronics 540
Protecting Your Brains 541
Making Connections 542
The Code 545
Armor 550
Weapons 552
Additional Information 561
Summary 562
Â Chapter 13: Alternate Control 563
Using Processing to Decode the Signals 563
Parts List for Alternate Control 564
Selecting the Input 565
Processing Prerequisites 566
Following Protocol 566
Examining the Processing Sketch 567
Explanation 570
Testing Processing 572
Examining the Arduino Sketch 574
Summary 579
Â Index 581

About the Authors

J-D is an electronics hobbyist, builder, and relentless tinkerer. As a child, he took apart everything he owned to figure out how it worked. Since then he has built many different projects ranging from an electric fishing pole to a remote-controlled lawn mower, which was featured on the cover of *MAKE* magazine in April 2010. Having worked as a builder doing carpentry, plumbing, and electrical work for eight years, his knowledge is founded on real-world experience rather than text-book recitation.

In addition to building robots and remote-controlled toys, he enjoys automating everyday tasks, blinking LEDs, designing and etching PCBs, and lots of random things in between. Much of his time has been spent researching, building, and testing various motor-controllers to make his bots move. As a self-proclaimed "poor man's roboticist," he will always try to find the cheapest way to do something—usually by building it himself.

J-D graduated from the University of Alabama at Birmingham with a degree in business management. He currently lives in Birmingham, Alabama with his beautiful wife, Melissa, and their growing flock of animals.

Josh Adams is a software developer with over ten years of professional experience building production-quality software and managing projects. He built a Tesla coil for a high-school science project that shot >27" bolts of lightning. Josh is Isotope Eleven's lead architect, and is responsible for overseeing architectural decisions and translating customer requirements into working software. Josh graduated from the University of Alabama at Birmingham (UAB) with Bachelor of Science degrees in both mathematics and philosophy. When he's not working, Josh enjoys spending time with his family.

Harald Molle has been a computer engineer since 1984. He started his career by becoming a researcher at a university in the southwest of Germany before cofounding an embedded systems company. Harald is also an expert scuba diver, a passion he is trying to combine with his work by developing a GPS-controlled robot to survey lakes. He is happily married to Jacqueline, who knows that an interest in robotics requires substantial amounts of time. And he owns a cat.

About the Technical Reviewers

Josh Adams is a software developer with over ten years of professional experience building production-quality software and managing projects. He built a Tesla coil for a high-school science project that shot >27" bolts of lightning. Josh is Isotope Eleven's lead architect, and is responsible for overseeing architectural decisions and translating customer requirements into working software. Josh graduated from the University of Alabama at Birmingham (UAB) with Bachelor of Science degrees in both mathematics and philosophy. When he's not working, Josh enjoys spending time with his family.

Guilherme Martins was born in Lisbon in 1977. He has always been interested in many forms of art and started early to experiment with various mediums—photography, video, drawing, and painting. Guilherme has worked for several design studios and advertising agencies in Lisbon since 2000 and has been freelancing visual projects, usually related to motion graphics, visual effects, and web design. In 2007 he started to collaborate with the choreographer Rui Horta on visual content to be projected on stage during dance, theater, and opera performances.

As a tinkerer and inventor, he has a particular interest in experimenting with robotics and electronics in order to create innovative interactive experiences. Most of his research and professional work is online at http://guilhermemartins.net.

Acknowledgments

Co-authors:
I would like to thank my friend and fellow hacker, Josh Adams, for his assistance in several of my projects as well as his constant support and ideas, and for introducing me to the Arduino in the first place. He also wrote a chapter in this book on alternate control (Chapter 13), using your PC with a game-pad and wireless serial link to control a large robot (Arduino + Processing). Josh makes my projects more awesome with his mad coding skills and by keeping up with all the latest hacks. Thanks for all of your help—you are a true friend.

I would also like to thank a fellow Arduino hacker, Harald Molle, for contributing his time and the details of his complex project, the GPS-guided RoboBoat (Chapter 9). Harald graciously took time out of his schedule to write a chapter that I was having issues with. I made three (unsuccessful) boat hulls before finding Harald's project on the Arduino forums and discovering that he had already been through the same pitfalls that I was encountering. He concocted a brilliant design for a catamaran-style boat that is easy to build and holds a straight line in the water. Upon realizing that he knew far more about this project than I could hope to learn, I was thrilled that he was willing to share his experience.

Tech editors:
A special thanks to Guilherme Martins and Josh Adams for their time reviewing the book—your suggestions and feedback were greatly appreciated.

Editors:
A big thanks to Michelle Lowman for giving me the opportunity to write this book, Anita Castro for being patient despite the extra time needed to complete the book, James Markham, Frank Pohlmann, and Dominic Shakeshaft for their help and guidance, and the rest of the Apress team that helped make this book happen—writing your first book is not easy.

Family:
I would like to thank my wife, Melissa, for being so understanding and helpful during this process. I could not have written this book without your support—I love you! Lastly, I would like to thank my family for their support and prayers during this project.

The Arduino community:
The Arduino community was the single largest source of inspiration for the various projects in this book. When I first started learning about the Arduino and physical computing (back in 2008), I had never touched a microcontroller or programmed a computer before, and I was welcomed with open arms. Random people gave me code examples, project advice, ideas for additional features, and, most of all, support. In a society where everyone is trying to make a buck, it is nice to be a part of something awesome that you know you can participate in without spending any money (other than the inexpensive Arduino board itself).

I would like to call out a few specific robotics enthusiasts, makers, and electronics gurus that have contributed to my learning and consequently this book: thanks to Massimo Banzi, Tom Igoe, and the rest of the Arduino development team, Limor Fried (Ladyada.net), Nathan Seidle (SparkFun.com), David Cood (Robotroom.com), and Jordi Munoz and Chris Anderson (DIYdrones.com) for their excellent Arduino tutorials, projects, and contributions to the open source community. There are a few people who were personally helpful with specific questions, projects, and parts: Larry Barello, John Dingley, Shane Colton, and Bob Blick.

Thanks to my common-thread friends that listened to my ranting ideas and nonsense, checked out my prototypes, offered suggestions, and even supplied me with parts to test when I could not afford my own: Josh Adams, Anthony DiGiovanni, and Laird Foret.

Though I took many photos for this book of various parts and procedures, there were some that did not turn out as well as I hoped, and for those I had help. The following photos originated from my various parts suppliers, and I would like to thank them for their help:

SparkFun.com: 1-10, 1-21, 1-30, 2-22, 2-24, 2-26, 2-27, 2-28, 2-29, 3-20

Digikey.com: 1-15, 1-16, 1-17

PartsForScooters.com: 10-4, 12-8, 12-18

DimensionEngineering.com: 10-20, 11-14

Adafruit.com: 2-16

Pololu.com: 10-22

Electronics-Tutorials.ws: 1-3

DannyG.com: 1-9

RobotMarketplace.com: 12-7

DFRobots.com: 5-11

HarborFreightTools.com: 10-3

John-David Warren

My wife, Kristen, who has been so patient with me fiddling with robots; my son, Matthew, who just wants to play with them; and my daughter, Gracie, who I'll meet in person real soon now.

Josh Adams

Thanks to Chris Anderson and Jordi Munoz, from DIYdrones.com, for their excellent pioneering work in building autopilots, Jean Margail from `http://water.resist.free.fr` for his plans of the catamaran hulls, Matthias Wolbert for making a 3D model out of these plans, and Robert Herrmann for his troubleshooting advice. Lastly, I dedicate this to my wife, Jacqueline—she knows why.

Harald Molle

Introduction

This book was written for anyone interested in learning more about the Arduino and robotics in general. Though some projects are geared toward college students and adults, several early chapters cover robotics projects suitable for middle-school to high-school students. I will not, however, place an age restriction on the material in this book, since I have seen some absolutely awesome projects created by makers both young and old.

Prerequisites

Ultimately, you will need to be able to use some basic power tools, hand tools, a voltage meter, and soldering iron. Do not worry if you are not yet experienced in these areas, as your first experience will get you well on your way (you have to start somewhere)! Just like riding a bike, you will get better at it the more you do it.

If you are an experienced robot builder, you will likely be able to improve upon some of my methods. If, however, you are a beginner, you might end up with a few extra holes drilled in the wrong spot, a wheel that is not mounted perfectly straight, or a downright ugly robot. Do not worry about trying to complete every step perfectly the first time; do your best the first time around and then go back and improve upon it later. It is better to have an imperfect robot that you can work on than no robot at all because you were too afraid to try!

In conclusion, this book is intended to provide fun projects for those interested in the Arduino. If you are working on one of these projects and you aren't having fun, you're doing it wrong. If you get stuck on a project, please ask for help—nobody wants you to be frustrated, but learning something new can sometimes make you want to drive your head through a wall...don't do that. Just keep with it, and you will *eventually* figure out your problem. I have created a Google web site to host the files for each project and provide a place to ask questions and get help:
`https://sites.google.com/site/arduinorobotics/`

If you would like to try some other Arduino projects, dealing with various types of sensors, LEDs, home automation, and various other projects, you might consider the following Arduino books from Apress:

Practical Arduino by Jonathan Oxer and Hugh Blemings (2009)

Beginning Arduino by Michael McRoberts (2010)

John-David Warren

CHAPTER 1

The Basics

The Arduino microcontroller (Figure 1-1) is like a little command center that is awaiting your orders. With a few lines of code, you can make your Arduino turn a light on or off, read a sensor value and display it on your computer screen, or even use it to build a homemade circuit to repair a broken kitchen appliance. Because of the versatility of the Arduino and the massive support available from the online community of Arduino users, it has attracted a new breed of electronics hobbyists who have never before touched a microcontroller, let alone programmed one.

Figure 1-1. *An Arduino Duemilanove microcontroller*

The basic idea of the Arduino is to create an atmosphere where *anyone* who is interested can participate and contribute with little upfront cost. A basic Arduino board can be found online for around $20, and all of the software needed to program the Arduino is open-source (free to use and modify). You need only a computer and a standard USB cable. In addition to being inexpensive, the creators of

J.-D. Warren et al., *Arduino Robotics*
© John-David Warren, Josh Adams, and Harald Molle 2011

Arduino came up with an easy-to-learn programming language (derived from C++) that incorporates various complex programming functions into simple commands that are much easier for a beginner to learn.

This book integrates some basic robot-building techniques with the simplicity of the Arduino to create bots that you can modify and improve with a clear understanding of your work. This book is not intended to simply "show" you how to build a bot, but rather to educate the beginning robot builder and hopefully inspire creativity so that you can design, build, and modify your own robots.

One unavoidable obstacle that most people encounter when building a robot is cost. Obviously we can spend thousands of dollars adding top-of-the-line parts and expensive commercial products, but most hobby builders have neither the time nor the money to build such a robot. With that in mind, this book takes every opportunity to show you how to build a part from scratch—or as inexpensively as possible to get the job done. If any of these methods seem too involved, do not worry because there are substitute parts listed for you to purchase.

Please understand that each project in this book requires multiple tries before working—some of them even take weeks of "debugging." I can tell you from experience that when you are persistent, you will eventually solve your problem—and this will make the experience that much more rewarding. Figuring out why a robot is not working often requires a lot of troubleshooting. Troubleshooting requires understanding each step in the process from start to finish, and inspecting each step for errors. The more you tinker with something, the better you will understand it.

Lastly, do not be discouraged if some of the information in this book appears to be over your head. We try to assume that you are new to robotics and programming, and we focus on providing a *practical working knowledge* of the parts and code used in each project, rather than loading you down with electronics theory and complicated instructions. It is best to take a positive "I can do it" attitude before you start—this will be your greatest tool.

To better understand what is happening inside an Arduino, we should first discuss electricity and other basics in general (i.e., electronics and circuits). Although levels found in your Arduino (+5 DCV) are relatively harmless, if you don't know how electricity works you won't know at what point it becomes dangerous. As it turns out, the projects covered in this book do not use electrical levels high enough to conduct through your body, but electricity should still be handled with caution.

Electricity

Electricity is nothing more than harnessed heat. This heat can be used to do a variety of different things like lighting up a lightbulb, spinning a motor, or simply heating a room. If electricity can transfer through an object easily, it is called a "conductor" (like copper wire). Every conductor has an internal resistance to the electricity that keeps it from transferring 100% of the power. Even a copper wire has some resistance that slows the flow of electricity, thereby generating heat. Conductors also have a maximum amount of power that they can transfer before "overheating" (if the conductor is a copper wire, that means melting). With regard to electricity, total power can also be referred to as total heat. This is why you might see a lightbulb or microwave that has its heat rating in watts. A watt is not only a measurement of heat, but of electrical power.

Some electrical devices (like the Arduino) consume little electricity therefore producing little heat, so no attention is given to heat dissipation. Other devices are made specifically to transfer large amounts of electricity (like a motor-controller) and must use metal heat-sinks or fans to aid in removing heat from the device. In either case, it is helpful to be able to determine the amount of heat that an electrical device produces so we know how to properly handle it.

Electrical Analogy

Electricity is not usually seen (except maybe in a lightning storm), so it is difficult to understand what is happening inside of a wire when you turn on a lamp or kitchen appliance. For ease of illustration, consider an electrical system to be a tank of water with an outlet pipe at the bottom (see Figure 1-2).

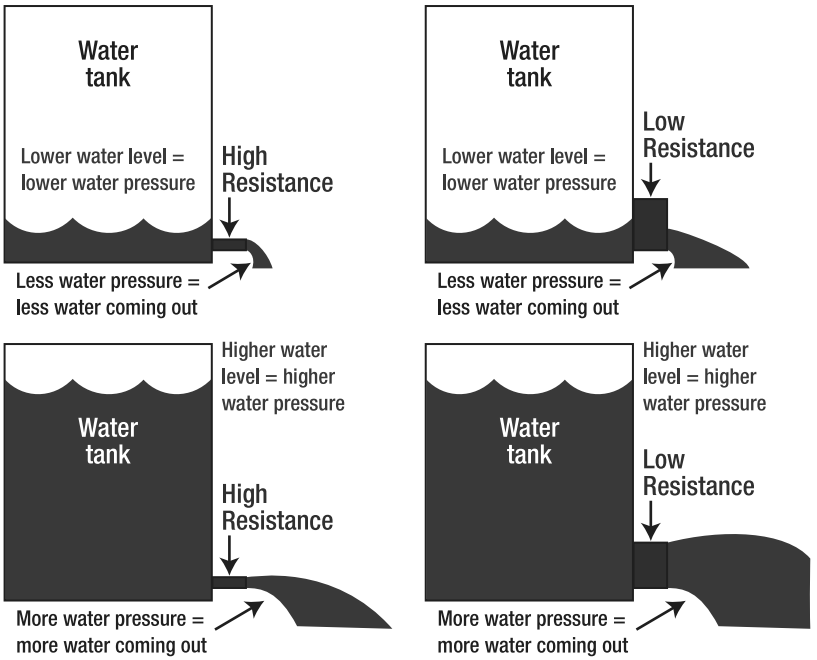

Figure 1-2. *An analogous electrical system*

The four images illustrate how resistance and pressure affect the water output from the tank. A higher resistance yields less water output, whereas a higher pressure yields more water output. You can also see that as the resistance is lowered, much more water is allowed to exit the tank, even with a lower pressure.

The more water that is in the tank, the faster (higher pressure) it pushes the water through the outlet pipe. If there were no outlet pipe, the tank of water would simply be a reservoir. The fact that there is an outlet pipe at the bottom of the tank enables water to exit, but only at a rate determined by the size of the pipe. The size of the outlet pipe determines the resistance to the water leaving the tank—so increasing or decreasing the size of the outlet pipe inversely increases or decreases the resistance to the water leaving the tank (i.e., smaller pipe = more resistance = less water exiting the tank).

Both the level (or pressure) of the water and the resistance (or size of the outlet pipe) can be measured, and using these measurements, you can calculate the amount of water exiting the tank at a given point in time. The difference in the water analogy and electricity flow is that the electricity must complete its path back to the source before it can be used.

Electrical Basics

Notice that a higher water pressure yields a higher water output (keeping resistance the same). The same is true with the electrical equivalent of pressure, called "voltage" (V), which represents the potential energy that can be found in an electrical system. A higher system voltage has more energy to drive the components in the system. The amount of "resistance"(R) found in a system impedes (slow) the flow of electricity, just as the resistance caused by the outlet pipe slows the flow of water from the tank. This means that as the resistance increases, the voltage (pressure) must also increase to maintain the same amount of output power. The amount of electrical charge (in coulombs) that is passed through an electrical system each second is called the "amperage" (I) or "current," and can be calculated using the voltage, resistance, and Ohm's law. A "watt" (P) is a measure of electrical power that is calculated by multiplying the voltage times the amperage. In this chapter, we further discuss voltage, resistance, and amperage. First, let's look at the relationship among them, Ohm's law.

According to Wikipedia (*Source:* **http://en.wikipedia.org/wiki/Ohm's_law**), Ohm's law states that the current through a conductor between two points is directly proportional to the potential difference or voltage across the two points, and inversely proportional to the resistance between them.

There is a simple relationship among voltage, resistance, and amperage (current) that can be calculated mathematically. Given any two of the variables and Ohm's law, you can calculate the third. A watt is a measure of electrical power—it is related to Ohm's law because it can also be calculated using the same variables. See the formulas in Figure 1-3 where V = voltage, R = resistance, I = amperage, and P = watts.

Â **Note** The pie chart in Figure 3-1 is used courtesy of **www.electronics-tutorials.ws**. If you are interested in learning more about electronics, you should definitely visit this website —it has some helpful illustrations and descriptions.

The different views of Ohm's law include the following:

$V = I * R$

$I = V / R$

$R = V / I$

Use the following formulas to calculate total power:

$P = V * I$

$P = I^2 * R$

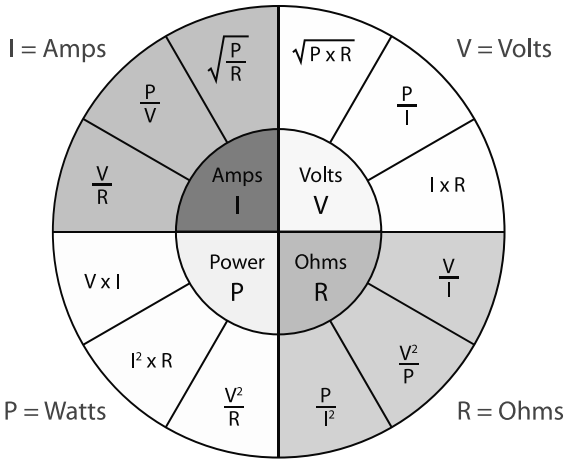

Figure 1-3. *Ohm's law to calculate power*

There are several other terms that you might come across when working on an electrical system; we discuss a few here. As you might know, an electrical system usually has a "power" wire and a "common" wire to complete the circuit. Depending on what you are reading, these two sides can be called different things. To help avoid the confusion that I experienced when I was learning, Table 1-1 provides a quick comparison of the various names for the positive and negative ends of an electrical system.

Table 1-1. *Common Names That Refer to the Positive and Negative Ends of an Electrical System*

Voltage Bias	Polarized Terminal	Electrical Current Flow	Schematic Label	Common Name
Positive	Anode	Source	VCC	Power
Negative	Cathode	Sink	VSS	Ground (GND)

We discussed Ohm's law and the common measurements that are used to describe the various properties of electrical current flow. Table 1-2 provides a list of standard electrical units and their symbols. These are used in every subsequent chapter of this book, so it is a good idea to get familiar with them.

Table 1-2. *Common Electrical Measurement Terms with Their Symbols*

Measurement	Unit	Symbol
Voltage (energy)	Volt	V or E
Amperage (current)	Ampere (amp)	I or A
Resistance	Ohm	R or Ý

Measurement	Unit	Symbol
Power (electrical heat)	Watt	P or W
Capacitance	Farad	F
Frequency	Hertz	Hz

Let's now talk more about the different parts of an electrical system.

Circuits

The starting point of the electricity in a system is called the "source" and usually refers to the positive battery lead or power supply. The electricity flows from the source, through the system, and to the sink, which is usually the negative battery terminal or ground wire (GND). For electricity to flow, the circuit must be "closed," which means that the electrical current can get back to its starting point.

The term "ground" comes from the practice of connecting the return path of an AC circuit, directly into the ground outside using a copper rod. You might notice that most electrical meters also have a ground rod nearby that is clamped to a wire leading into the fuse-box. This ground wire gives the returning electrical current a path to exit the system. Even though the DC equivalent of GND is the negative battery terminal, we still call it GND.

Â **Note** the actual electron-flow of electrical current travels from negative to positive, but unless you are a physicist, that is not relevant here. For learning purposes, we assume the conventional electron-flow theory, which suggests that electrical current flows from Positive (+) ----> Negative (-) in a system.

An electrical system is called a "circuit," and can be simple like a string of Christmas lights plugged into a power outlet or very intricate like the motherboard in your PC. Now consider that in a circuit, the electricity flows only if something is there to complete the circuit, called a "load" (see Figure 1-6). In general, the load in a circuit is the device you intend to provide with electricity. This can be a lightbulb, electric motor, heater coil, loud speaker, computer CPU, or any other device that the circuit is intended to power.

There are three general types of circuits: open-circuit, closed-circuit, and short-circuit. Basically, an open-circuit is one that is turned off, a closed-circuit is one that is turned on, and a short-circuit is one that needs repair (unless you used a fuse). This is because a short-circuit implies that the electricity has found a path that bypasses the load and connects the positive battery terminal to the negative battery terminal. This is always bad and usually results in sparks and a cloud of smoke, with the occasional loud popping sound.

In Figure 1-4, the lightbulb is the load in this circuit and the switch on the left determines whether the circuit is open or closed. The image on the left shows an open-circuit with no electricity flowing through the load, whereas the image on the right shows a closed-circuit supplying power to the load.

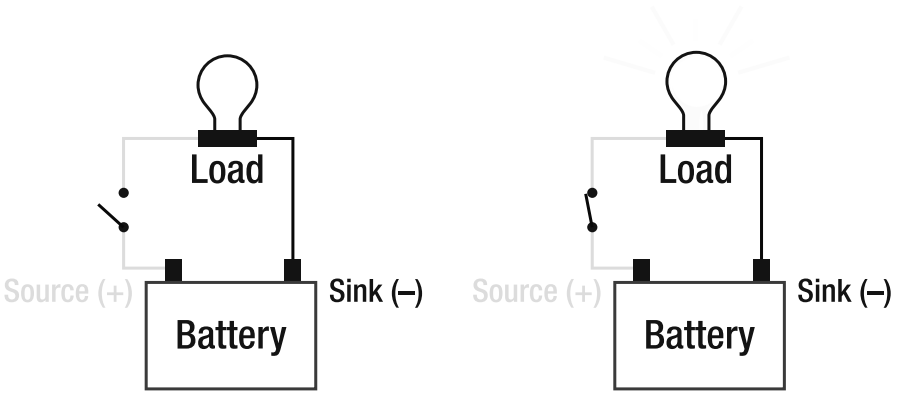

Figure 1-4. *Open- and closed-circuits*

Measuring Electricity

Without a way to measure electrical signals, we would be flying blind—luckily, there is a device called a "multi-meter" that is inexpensive and can easily measure voltage, resistance, and small levels of current.

Multi-Meters

There are different types of multi-meters that have varying features, but all we need is a basic meter that can measure voltage levels up to about 50DCV.

A typical multi-meter can measure the voltage level of a signal and the resistance of a component or load. Because you can calculate the amperage given the voltage and resistance, this is really all you need to do basic circuit testing. Although the full-featured digital multi-meter in Figure 1-5 (left) is priced around $50, you can usually find a simple analog multi-meter (right) that measures both voltage and resistance for under $10. Both meters will do basic testing and although the digital meter is nicer, I actually like to keep a cheap analog meter around to measure resistance, because you can see the intensity of the signal by how fast the needle moves to its value.

Figure 1-5. *The Extech MN16a digital multi-meter (left) measures AC and DC voltages, resistance, continuity, diode test, capacitance, frequency, temperature, and up to 10 amps of current. An inexpensive analog multi-meter purchased at my local hardware store (right) measures DC and AC voltages, resistance (1k ohm), and up to 150mA (0.15A) of current. Either work to diagnose an Arduino and most other circuits—but you definitely need one.*

The standard multi-meter has two insulated test-probes that plug into its base, and are used to contact the electrical device being tested. If you test the voltage of a circuit or battery, you should place the red probe (connected to the multi-meter "V, Ý, A" port) on the positive battery supply, and the black probe (connected to the multi-meter "COM" port) on the negative battery supply or GND.

Measuring Voltage

Voltage is measured as either Alternating Current (AC), which is the type found in your home electrical outlets, or Direct Current (DC), which is found in batteries. Your multi-meter needs to be set accordingly to read the correct voltage type. Some multi-meters also have a range that you need to set before testing a voltage. The analog multi-meter in Figure 1-5 (right) is set to 10DCV, effectively setting the needle range from 0-10VDC.

Trying to read a voltage that is much higher than the selected range can result in a blown fuse, so you should always use a voltage range that is higher than the voltage you test. If you are unsure what voltage level you are testing, select the highest range setting (300VDC on this multi-meter) to get a better idea. The digital multi-meter in Figure 1-7 (left) has DC and AC voltage settings, but the range is automatically detected and the exact voltage number appears on the screen—just be sure not to exceed the maximum voltage ratings stated in the multi-meter owner's manual.

The voltage level of an electrical signal also determines whether or not it is capable of using your body as a conductor. The exact voltage level that passes through the human body is probably different depending on the size of the person (moisture levels, thickness of skin, etc.), but I can verify that accidentally touching a 120v AC wall outlet (phase wire) while standing on the ground produces quite a muscle convulsion, even if wearing rubber-soled shoes.

Â **Caution** Voltage levels above 40v can be harmful to humans or pets. Always remember to disconnect the power source when working on your circuits and use insulated tools (with rubber grips) to test circuits. You don't want to end up in a hospital bed!

Measuring Amperage

Most multi-meters have a feature to measure small amounts of amperage (250mA or less) of either AC or DC. The digital multi-meter in Figure 1-5 (left) can measure up to 10 amps of current for a few seconds at a time whereas the less featured meter can measure up to 150mA of current only. To measure large amounts of current (over 10A), you either need a current-sensor, ammeter, or voltage clamp, depending on the application.

This unit of measure depends on the operating voltage and resistance of the circuit. As the operating voltage decreases (batteries discharge) or the resistance fluctuates, the amperage draw also changes. On a large robot that is constantly moving, the amperage draw changes every time the robot drives over a rock or up a slight incline. This is because DC motors consume more amperage when presented with more resistance. An LED flashlight on the other hand, consumes a steady amount of current (about 20-100mA per LED) until the batteries run dead.

You might have noticed that batteries are rated in Amp/Hours (AH) to reflect the amount of electrical current they can supply and for how long. This loosely means that a battery rated for 6v and 12AH can supply a 6v lamp with 1 ampere of current for 12 hours or the same 6v lamp with 12 amperes for 1 hour. You might also notice that smaller batteries (like the common AA) are rated in milliamp/hours (mAH). Thus a 2200mAH battery has a rating equal to 2.2AH.

Measuring Capacitance

Capacitance is the measure of electrical charge that can be stored in a device, measured in Farads—but 1 Farad is a huge amount of capacitance, so you will notice that most of the projects use capacitors with values in the microfarad (uF) range. A *capacitor* is an electrical device that can hold (store) electrical charge and supply it to other components in the circuit as needed. Though it might sound like a battery, a capacitor can be completely drained and recharged multiple times each second—the amount of capacitance determines how fast the capacitor can be drained and recharged.

Some multi-meters can measure the amount of capacitance that is present between two points in a circuit (or the value of a capacitor), like the Extech MN16a in Figure 1-5. Most multi-meters do not measure capacitance, because it is not usually of great importance in most circuits. Being able to test capacitance can be helpful when trying to achieve specific values or testing a capacitor, but generally you will not need this feature on your multi-meter.

Â **Caution** Larger capacitors can hold a significant charge for long periods of time, and touching the leads of a charged capacitor can cause electrical shock. Capacitors found in CRT computer monitors or televisions, motor-start capacitors, and even the small capacitors found in disposable cameras can provide a shock that leave your arm tingling for several minutes and even burn your skin. It is a good idea to "short" the leads of a capacitor together with an insulated screwdriver to discharge any stored current before attempting to handle it.

Measuring Resistance

Resistance is measured in ohms and tells us how well a conductor transfers electricity. Current flow and resistance are inversely related. As resistance increases, current flow decreases. Thus, a conductor with lower resistance transfers more electricity than one with higher resistance. Every conductor has *some* resistance—some materials have such a high resistance to current flow, they are called "insulators" meaning that they will not transfer electricity. When electricity is resisted while passing through a conductor, it turns into heat; for this reason, we use conductors with the lowest resistance possible to avoid generating heat.

A *resistor* is an electrical device that has a known resistance value in ohms and is used to limit the amount of current that can flow through it (see Figure 1-6).

Figure 1-6. *Three resistors: 1/4 watt surface mount resistor (left), 1/8 watt through-hole resistor (center), and 1/4 watt through-hole resistor (right)*

Notice that the 1/4 watt surface mount resistor (left) is much smaller than the equivalent ¼ watt through-hole resistor (right), even though it dissipates the same amount of power. I typically use 1/8 watt through-hole resistors as they are small but still easy to work with.

You can use a resistor in-line with a component to limit the amount of electrical current delivered to the device, in order to ensure it stays within a safe operating range.

The number on the chip resistor designates its resistance value in ohms, while the color-coded stripes on the through-hole resistors designate their resistance value. If you want to manually check the resistance of a component, use your multi-meter on the Ohm (Ý) setting – polarity does not matter, unless you measure the resistance of a diode or transistor.

I use a neat web page that enables you to enter the colors of each band on a resistor, and it tells you the resistance value in ohms (see Figure 1-7). It is helpful for quick reference while prototyping or identifying a loose resistor's value. Visit **http://www.dannyg.com/examples/res2/resistor.htm**.

Image used with permission from Danny Goodman.

Figure 1-7. *This screen-shot shows the web application designed by Danny Goodman. I have this web page bookmarked in my web browser and use it often to check unfamiliar resistor color codes.*

Calculating Resistor Power Using Ohm's Law

Remember that any time resistance is present in a circuit, heat will be generated, so it is always a good idea to calculate how much heat will be passed through a resistor (depending on the load) in order to select a resistor with a sufficient power rating. Resistors are not only rated in ohms, but also by how much power they can dissipate (get rid of) without failing. Common power ratings are 1/8 watt, ¼ watt, ½ watt, and so on, where larger watt values are typically larger resistors unless using surface mount components (see Figure 1-5).

To calculate the power dissipated in a resistor, you need to know the circuit voltage and the resistor value in ohms. First, we need to use Ohm's law to determine the current that will pass through the resistor. Then we can use the resistance and amperage to calculate the total heat that can be dissipated by the resistor in watts.

For example, if we have a 1000 ohm resistor (1kilo-Ohm) and a 12v power supply, how much amperage will be allowed to pass through the resistor? And what should the minimum power rating be for the resistor?

First we calculate the amperage through the resistor using Ohm's law:

$V = I * R$

$I = V / R$

$I = 12v / 1000 ohm$

$I = 0.012$ amps or 12 milliamps

Now we use the amperage to calculate the total power (heat):

$P = I^2 * R$

$P = (0.012 amps * 0.012 amps) * 1000 ohms$

$P = 0.144$ watts

The total power is calculated as 0.144 watts, which means we should use a resistor with a power rating greater than .144w. Because common resistor values are usually 1/8w (0.125w), 1/4w (0.25w), 1/2w (0.5w), and so on, we should use a resistor with a power rating of at least 1/4w (a common size) and still safely dissipate 0.144w of power. Using a 1/2w resistor will not hurt anything if you can fit the larger size into the circuit–it will simply transfer heat more easily than a 1/4w resistor with the same resistance value.

Now you should be able to figure out if your resistors have an appropriate power rating for your application. Let's talk about the different types of load components.

Oscilloscope

Although the multi-meter is great for measuring the voltage, resistance, and amperage, it is sometimes helpful to be able to see exactly what is going on in an electrical signal. There is another device that is designed to analyze electrical signals, called an "oscilloscope." The oscilloscope can detect repeated patterns or oscillations in an electrical signal, and display the wave-form of the signal on the screen of the device. It is effectively a microscope for electrical signals. These machines have been expensive ($500-$5000) until recently—some hobby grade oscilloscopes have entered the market for under $100.

The open-source DSO Nano (see Figure 1-8) digital oscilloscope built by Seeedstudio.com and also sold (in the United States) through Sparkfun.com (part #TOL-10244). I have had this oscilloscope for about a year and use it frequently because it is easy to use and about the size/weight of a cell-phone, all for about $89. It contains a rechargeable lithium battery and can be charged through a mini USB cable. It also has a memory card slot available for storing readings to view later on a PC.

Figure 1-8. *The DSO Nano from SeeedStudio.com (and sold through Sparkfun.com) is an excellent choice for an inexpensive ($89), but full-featured, digital pocket oscilloscope.*

Although an oscilloscope is an invaluable tool to have when diagnosing electronic signals, it is not necessary to have for the projects in this book. You can get by with readings from a simple multi-meter. There are also other budget oscilloscope options available, including a DIY kit from Sparkfun.com for around $60 (part #KIT-09484).

Loads

The "load" in a circuit refers to a device in the circuit that uses the electricity. There are many different examples of a load from a DC motor to an LED or a heater coil, and each will create a different reaction in the circuit. For instance, a heater coil (found in a hair dryer or space heater) is simply a coiled resistive wire made from a metal that can become glowing red when it is hot, but it does not melt. Whereas an electric motor uses electricity to energize an electro-magnetic field around a coil of wire, causing the motor shaft to physically move. There are two types of loads on which we focus: inductive and resistive.

Inductive Loads

If you apply power to a device and it creates moving energy, it is likely an inductive load–this includes motors, relays, and solenoids. Inductive loads create an electro-magnetic field when energized and usually take some time to deenergize after the power is disconnected. When the power is disconnected using a switch, the magnetic field collapses and dumps the remaining current back to the power terminals. This phenomenon is called Back-EMF (Electro-Motive Force) and it can damage the switching components in a circuit if they are not protected by rectifying diodes.

Resistive Loads

A resistive load uses electrical current to produce light or some other form of heat, rather than mechanical movement. This includes LEDs, heater elements, lightbulb filaments, welding machines, soldering irons, and many others. Resistive loads use a constant amount of electricity because their load is not affected by external influence.

Electrical Connections

When building an electrical circuit, you should determine the desired operating voltage before selecting components with which to build the circuit. Although lowering AC voltage levels requires the use of a transformer, specific DC voltage levels can be achieved by using different wiring methods to connect several individual battery packs. There are two different types of electrical connections: series and parallel.

Series Connections

To arrange a circuit in “series” means to place the devices in-line with or through one another. We often use a series connection with batteries to achieve a higher voltage. To demonstrate this circuit, we use two 6v 10-Ah batteries with the positive (+) terminal of the first battery connected to the negative (-) terminal of the second. The only open terminals now are the negative (-) terminal of the first and the positive (+) terminal of the second, which will produce a difference of 12v.

When two batteries are arranged in a series circuit (see Figure 1-9), the voltage is doubled but the Amp/Hour capacity stays the same. Thus the two 6v 10AH batteries work together to produce a single 12v 10AH battery pack. This technique can be helpful to reach specific voltage levels.

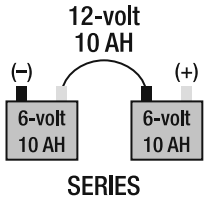

Figure 1-9. *Two batteries arranged in a series circuit produce twice the voltage but the same Amp/Hour capacity.*

Parallel Connections

To arrange a circuit in “parallel” means to place all common terminals together. This means that all the positive terminals are connected together and all the negative terminals are connected together. If we place the two 6v 10AH batteries from the previous example into a parallel circuit (see Figure 1-10), the voltage will stay the same but the Amp/Hour capacity will double resulting in a single 6v 20AH battery pack.

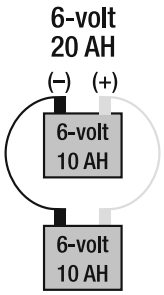

Figure 1-10. *Two batteries arranged in a parallel circuit produce the same voltage but with twice the Amp/Hour capacity.*

Series and Parallel Connection

It is also perfectly acceptable to arrange several battery packs in both series and parallel at the same time, in order to achieve a specific voltage and Amp/Hour rating (see Figure 1-11). Notice that there are two sets of 6V, 10AH batteries arranged in series to produce 12V, and then the two series packs are arranged in parallel to produce the same voltage, but with 20AH capacity.

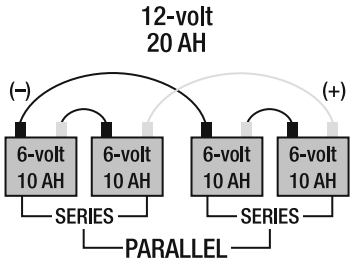

Figure 1-11. *By making two sets of series connections and placing them in parallel, you can create a 12v battery pack with 20AH of current capacity using four 6v 10AH battery packs.*

When building a battery pack, it is important to use batteries of the same voltage and AH capacity to build larger cells. This means that you should not pair a 12v battery with a 6v battery to achieve 18v. Instead use three 6v batteries with the same capacity to achieve 18v and avoid uneven charging/discharging.

Electronics

The field of electronics deals with controlling the flow of electrical current through a circuit, specifically using the electronic switch. Prior to the invention of the electronic switch, electrical circuits were turned on and off using mechanical switches, which requires mechanical motion (i.e., your hand moving the switch up or down) to connect or disconnect the circuit. Although mechanical switches are perfectly acceptable and even preferred for some applications, they are limited to how fast they can be switched due to the physical motion that must occur during the switching process. Even an electro-mechanical

switch (called a relay) does not qualify as an electronic device, because it uses electricity to generate a mechanical motion used to activate the switch.

The electronic switch forgoes the mechanical switching action by using an electrical reaction within the device, thus there are no moving parts. Without a physical movement, these devices can be switched extremely fast and with much greater reliability. The substances that these switches are made from conduct electricity only under certain circumstances—usually a specific voltage or current level must be present at the input and output of the device to open or close it. When the device is turned on, it conducts electricity with a specified amount of resistance. When the device is turned off, it does not conduct electricity and instead acts as an insulator. This type of electronic component is called a "semi-conductor" because it can become a conductor or insulator depending on the electrical conditions.

Semi-Conductors

The use of semi-conductors in place of mechanical switches is what makes a circuit "electronic," because they enable electrical signals to be switched at extremely high speeds, which is not possible with mechanical circuits. There are many different semi-conductors, and we discuss a few important types that are used in most of our circuits.

- *Diode*: Like a one-way valve for electrical current, this device enables only electrical current to pass through it in one direction–extremely useful by itself, but also the basis for all solid state electronics.
- *Light Emitting Diode (LED)*: This type of diode emits a small amount of light when electrical current passes through it.
- *Light Dependent Resistor (LDR)*: This type of semi-conductor has a changing resistance, depending on the amount of light present.
- *Bipolar Junction Transistor (BJT*: This is a current-driven electronic switch used for its fast switching properties.
- *Metal-Oxide Semiconductor Field-Effect Transistor (moset)*: This is a voltage-driven electronic switch used for its fast switching properties, low resistance, and capability to be operated in a parallel circuit. These are the basis for most power amplifier circuits.

These devices all have multiple layers of positively and negatively charged silicon attached to a chip with conductive metal leads exposed for soldering into the circuit. Some transistors and mosfets have built-in diodes to protect them from reverse voltages and Back-EMF, so it is always a good idea to review the datasheet of the part you are using.

Datasheets

Each device should have its own datasheet that can be obtained from the manufacturer–usually by downloading from its website. The datasheet has all of the important electrical information about the device. The upper limits, usually called "Absolute Maximum Ratings," show you at what point the device will fail (see Figure 1-12). The lower limits (if applicable) tell you at what level the device will no longer respond to inputs–these usually will not hurt the device, it just won't work.

Figure 1-12. *Here you can see the first page of a sample datasheet from Fairchild Semiconductor for the popular 2n2222 NPN transistor switch. First it shows the available packages and pin-configurations, and then a brief listing of the absolute maximum ratings.*

There is also a section called "Electrical Characteristics" that tells you at what level the device operates properly. This usually shows the exact voltage or current level that will turn the device on or off. These ratings are helpful in determining what other component values (i.e., resistors and capacitors) should be selected or whether the device will work for the intended purpose.

The datasheet usually tells you far more than you know what to do with, ending with graphs and package dimensions. Some datasheets even have circuit layout recommendations and suggest ways to interface the component with a micro-controller. For popular or commonly used component parts, you can also check the manufacturer's website for additional documents that further describe how to use the component–these are called "application notes," and can be insightful.

Integrated Circuits

Some semi-conductors include multiple components housed on the same chip, which are called Integrated Circuits (IC). An Integrated Circuit can contain thousands of transistors, diodes, resistors, and logic gates on a tiny chip (see Figure 1-13). These components are available in the larger "through-hole" packages and newer versions are being made on super-small "surface mount" chips.

Figure 1-13. *Here you can see an 8-pin Dual Inline Package (DIP) IC (left), and a 16-pin DIP IC (right). The Arduino's Atmega168/328 is a 28-pin DIP IC (14 pins on each side).*

Packages

We use different types of semi-conductors in various packages. The component package refers to the physical shape, size, and pin-configuration in which it is available. Different packages allow for various heat dissipation depending on the semi-conductor. If you are going for high power, larger cases usually dissipate heat better. For low power circuits, it is usually desirable to be as compact as possible, so smaller package sizes might be of interest. The most common packages that we use are the TO-92 and the T0-220 (see Figure 1-14), which house anything from temperature sensors to transistors to diodes.

Figure 1-14. *The smaller TO-92 IC package (left) is used for low-power voltage regulators, signal transistors, and sensor ICs. The larger TO-220 package (right) is used for higher power voltage regulators, power Mosfet switches, and high-power diodes.*

The TO-92 is a smaller package that is usually used for low-power transistor switches and sensors. The TO-220 packaged is commonly used for high-powered applications and is the basis for most power Mosfet transistors, capable of handling close to 75 amperes before the metal leads on the chip will fail. The TO-220 package also has a built-in metal tab used to help dissipate more heat from the package, and allowing a heat sink to be attached if needed.

Through-Hole Components

Throughout this book, we look for the easiest way to build and modify our projects. Usually that means using parts that can be replaced easily if needed and also using parts that are large enough for a beginner to feel comfortable soldering into place.

With respect to semi-conductor components, the term "through-hole" refers to any component whose leads are fed through holes drilled in the PCB and soldered to a copper "pad" on the bottom of the board. These parts are typically large enough to easily solder to a PCB, even for a beginner. Many through-hole components have pins that are much longer than needed, so it is recommended to solder

the component in place and finish by snipping the excess from the bottom of each pin to avoid any short-circuits on the under-side of the PCB.

IC Sockets

An “IC socket” is a plastic base that has metal contacts, which are intended to be soldered to the PCB (see Figure 1-15). The IC is then inserted into the socket after soldering is complete, alleviating the risk of overheating the IC during the soldering process. This is also helpful if something were to go wrong in the circuit, which causes the IC to fail. It is easily replaced without the need for additional soldering. We use IC sockets anytime we are able to for these reasons.

Figure 1-15. *An IC socket used to solder onto a PCB, in order to place the actual IC into once the circuit is built. These sockets are usually less than $1 each, so I try to use them whenever possible.*

Surface-Mount Components (SMT or SMD)

With the technological leaps that manufacturers have made in recent years, smaller has become better. This has led to decreasing the size of components and ICs so that they can create smaller devices that do the same thing as their larger counterparts.

Although these devices are internally the same, their lead pins are much smaller and might be a bit frustrating for a beginner when trying to solder them to a PCB (see Figure 1-7 (left) for a surface-mount resistor). The main difference between these and through-hole components is that they are soldered to the top of the PCB and no holes need to be drilled in the PCB. They also typically sit close to the PCB and require little room to mount them, making them desirable for space-saving applications.

Some surface mount parts have exposed terminals that are able to be soldered by normal means, but, others have their terminals exposed only on the underside of the chip, which requires that they are soldered in an surface mount reflow oven. Although a make-shift reflow oven can be emulated using a toaster-oven, we attempt to stay away from surface mount parts in the circuits we build in this book to avoid the added difficulty present with SMD parts.