



## Synopsis

Clear, easy-to-follow examples show you how to program Arduino with ease! "Programming Arduino: Getting Started with Sketches" helps you understand the software side of Arduino and explains how to write well-crafted Sketches (the name given to Arduino programs) using the C language of Arduino. This practical guide offers an unintimidating, concise approach for non-programmers that will get you up and running right away. "Programming Arduino: Getting Started with Sketches" Explains basic concepts and syntax of C with simple language and clear examples designed for absolute beginners - no prior knowledge of programming is required. It leads you from basic through to advanced C programming concepts and features dozens of specific examples that illustrate concepts and can be used as-is or modified to suit your purposes.\* All code from the book is available for download.\* Helps you develop working Sketches quickly. Coverage includes: C Language Basics; Functions; Arrays, Strings; Input / Output; Standard Library Goodies; Storage; LCD Displays; Programming for the Web; Program Design; C++ and Library Writing

## Book Information

Paperback: 176 pages

Publisher: McGraw-Hill Education TAB; 1 edition (November 29, 2011)

Language: English

ISBN-10: 0071784225

ISBN-13: 978-0071784221

Product Dimensions: 5.4 x 0.4 x 8.4 inches

Shipping Weight: 5.6 ounces

Average Customer Review: 4.4 out of 5 stars [See all reviews](#) (510 customer reviews)

Best Sellers Rank: #29,235 in Books (See Top 100 in Books) #10 in [Books > Engineering & Transportation > Engineering > Electrical & Electronics > Electronics > Microelectronics](#) #11 in [Books > Science & Math > Experiments, Instruments & Measurement > Experiments & Projects](#) #12 in [Books > Computers & Technology > Programming > Languages & Tools > C & C++ > C](#)

## Customer Reviews

Having just started using the Arduino last summer, I can say that this is the ABSOLUTELY BEST book a person could get to become familiar with the Arduino. I've been an electronics hobbyist for years but mainly dealt with analog world with minimal interest in too much on the digital side. I did programming but never with a microprocessor until I got my Arduino last summer. Now I'm hooked. Unfortunately, the usual books for "helping" with Arduino give sketches and explain bits and pieces

without ever doing an overview of the system. However, this book is fantastic. Just got it today and have read over half of it; I've marked it up with pen to remind myself of crucial facts that I wish I had known a few months back. I programmed in Fortran (antique language) and BASIC before, but never learned the C programming language (the language of the Arduino). Thankfully, this book assumes no knowledge of C language and holds your hand through the process. It does a wonderful job explaining what each section of the programming does, it explains the hardware of the board, and it explains how the board communicates with the computer. At long last, instead of just copying and pasting someone else's program and hoping I can modify it to work for my own purposes, I can understand WHY certain things work the way they do. UPDATE: I've had this book for over five months, and I still maintain that you have GOT to have this book. I have used it so much that I've about worn it out. Yes, the info that is in the book can be found elsewhere, but he covers so much material so well in such logical places. If you're trying to understand how and why Arduino sketches work the way they do, BUY THIS BOOK! UPDATE 2: I STILL stand by my review of almost a year ago.

I've been using, and learning about, Arduino for a few months. I bought a few other books to help me get moving, and have not been very pleased with them. So, recently, I wanted to start trying to do some of the more difficult things with Arduino, like using data storage and Ethernet. I looked on various web sites for tutorials, and examples, and found that none of the ones I downloaded were very good examples. The Ethernet subject was probably the most difficult. Many example programs would work sometimes, and not other times. Some never worked at all. But, worst of all, some examples had so many errors that I gave up on trying to fix and recompile them, and was never able to load them to the Arduino. Then, about a week ago, I saw this book by Simon Monk, and decided to order it, and give it a try. I have not regretted that move at all. The two best working Ethernet programs that I have found both came out of THIS book. I am using them as templates for building the more complicated applications I wanted. Unlike the other Ethernet examples I've had, Simon's examples both worked on the first try. I only had to change the IP address and MAC address in the program to point to my own Arduino devices. He has obviously tested them thoroughly before publishing them. How I wish everyone was that conscientious! Then, as I started to use these programs as templates to build my applications, I found myself going back into Simon's book for more examples, and more explanations on other topics. I used more of his code examples as building blocks for my data storage, and other functionality. Again, the examples in this book are very simple and solid, and they WORK. That's exactly what you need to use as a framework to build

your own applications.

I am experienced with microcontroller systems but new to Arduino. Here's what I think of this book:

**THE GOOD:**+ For a beginner it's very readable and isn't "here's a bunch of theory and then we do fun stuff". The book is very much hands-on oriented, giving just enough explanation related to the task at hand (which is both good & bad, more on this below)+ Book is thin and small, will not cause "I have to read and understand all that?" anxiety for beginners

**THE BAD:-** The use of timer interrupts is not discussed at all in this book. I found this disappointing as timer interrupts are a common way to handle "background" tasks (this is where you can run code at specific time intervals). One could argue that this is not a beginner topic but since pin interrupt-on-change interrupts ARE discussed I think timer interrupts should have been as well.

**UPDATE:** The follow-up book *Programming Arduino Next Steps: Going Further with Sketches* (Tab) covers timer interrupts and other fun stuff.- There are many issues that could be expanded on, even a couple of sentences would be helpful. Examples include:Arduino maximum pin current sink/source & applied voltage limitations. This is briefly mentioned in Chapter 1, but the author just mentions to be careful to not exceed the 40mA current limitation. He should have also mentioned that you can damage the chip this way (also if you exceed max voltage to an I/O pin). This should have been repeated in Chapter 6 which deals with digital interfacing.

[Download to continue reading...](#)

Arduino: Complete Beginners Guide For Arduino - Everything You Need To Know To Get Started (Arduino 101, Arduino Mastery) Programming Arduino: Getting Started with Sketches, Second Edition (Tab) Programming Arduino Getting Started with Sketches Programming Raspberry Pi 3: Getting Started With Python (Programming Raspberry Pi 3, Raspberry Pi 3 User Guide, Python Programming, Raspberry Pi 3 with Python Programming) Programming #8:C Programming Success in a Day & Android Programming In a Day! (C Programming, C++programming, C++ programming language, Android , Android Programming, Android Games) Programming #57: C++ Programming Professional Made Easy & Android Programming in a Day (C++ Programming, C++ Language, C++for beginners, C++, Programming ... Programming, Android, C, C Programming) Arduino: The Ultimate QuickStart Guide - From Beginner to Expert (Arduino, Arduino for Beginners) Getting Started Making Metal Jewelry (Getting Started series) Getting Started with Geese (Getting Started with... Book 4) Programming Arduino Next Steps: Going Further with Sketches Programming #45: Python Programming Professional Made Easy & Android Programming In a Day! (Python Programming, Python Language, Python for beginners, ... Programming Languages, Android

Programming) ESP8266: Programming NodeMCU Using Arduino IDE - Get Started With ESP8266: (Internet Of Things, IOT, Projects In Internet Of Things, Internet Of Things for Beginners, NodeMCU Programming, ESP8266) Getting Started with Arduino: The Open Source Electronics Prototyping Platform (Make) Getting Started with Sensors: Measure the World with Electronics, Arduino, and Raspberry Pi Programming: Computer Programming for Beginners: Learn the Basics of Java, SQL & C++ - 3. Edition (Coding, C Programming, Java Programming, SQL Programming, JavaScript, Python, PHP) Raspberry Pi 2: Raspberry Pi 2 Programming Made Easy (Raspberry Pi, Android Programming, Programming, Linux, Unix, C Programming, C+ Programming) Android: Programming in a Day! The Power Guide for Beginners In Android App Programming (Android, Android Programming, App Development, Android App Development, ... App Programming, Rails, Ruby Programming) DOS: Programming Success in a Day: Beginners guide to fast, easy and efficient learning of DOS programming (DOS, ADA, Programming, DOS Programming, ADA ... LINUX, RPG, ADA Programming, Android, JAVA) ASP.NET: Programming success in a day: Beginners guide to fast, easy and efficient learning of ASP.NET programming (ASP.NET, ASP.NET Programming, ASP.NET ... ADA, Web Programming, Programming) C#: Programming Success in a Day: Beginners guide to fast, easy and efficient learning of C# programming (C#, C# Programming, C++ Programming, C++, C, C Programming, C# Language, C# Guide, C# Coding)

[Dmca](#)