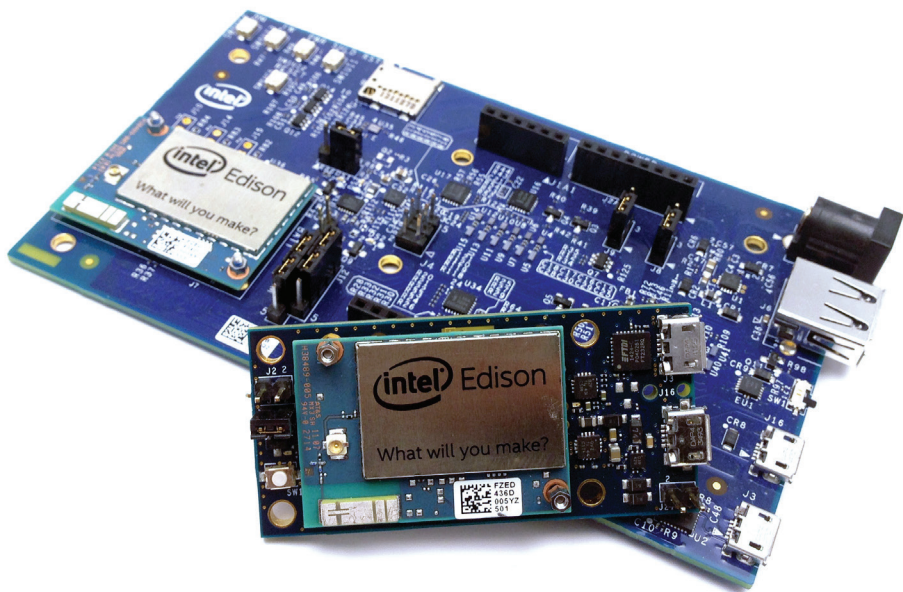


Make: Getting Started with Intel Edison



**Sensors, Actuators, Bluetooth, and WiFi
on the Tiny Atom-Powered Linux Module**
Stephanie Moyerman

Make: Getting Started with Intel Edison

Stephanie Moyerman



Make: Getting Started with Intel Edison

by Stephanie Moyerman

Copyright © 2016 Stephanie Moyerman. All rights reserved.

Printed in the United States of America.

Published by Maker Media, Inc., 1160 Battery Street East, Suite 125, San Francisco, CA 94111.

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

Editor: Roger Stewart

Production Editor: Nicole Shelby

Copyeditor: Gillian McGarvey

Proofreader: Amanda Kersey

Indexer: Ellen Troutman

Interior Designer: David Futato

Cover Designer: Brian Jepson

Illustrator: Rebecca Demarest

November 2015: First Edition

Revision History for the First Edition

2015-11-02: First Release

See <http://oreilly.com/catalog/errata.csp?isbn=9781457187599> for release details.

Make:, Maker Shed, and Maker Faire are registered trademarks of Maker Media, Inc. The Maker Media logo is a trademark of Maker Media, Inc. *Getting Started with Intel Edison* and related trade dress are trademarks of Maker Media, Inc.

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and Maker Media, Inc. was aware of a trademark claim, the designations have been printed in caps or initial caps.

While every precaution has been taken in the preparation of this book, the publisher and authors assume no responsibility for errors or omissions, or for damages resulting from the use of the information contained herein.

978-1-457-18759-9

[LSI]

Contents

Preface	vii
1/Introduction to Edison	1
Tour of Breakout Boards.....	2
The Arduino Breakout Board.....	3
Other Breakout Options.....	6
Setup and Configuration.....	11
Install.....	13
Mac and Windows.....	13
Connecting.....	16
Windows.....	16
Mac.....	19
Linux.....	19
Logging In.....	20
Configuring Edison and Getting Online.....	20
Troubleshooting.....	23
Going Further.....	24
2/Introduction to Linux	25
What Is Linux?.....	25
The Edison Filesystem.....	26
Basic Linux Commands.....	29
Accounts, Permissions, and Ownership.....	35
Scripting and More Advanced Linux Commands.....	38
The Internet.....	41
Installing Packages in Yocto.....	44
Text Editors.....	46
Going Further.....	47
3/Introduction to Arduino	49
What is Arduino.....	49
Materials List.....	50
The Arduino IDE.....	51

Installing.....	51
Navigating the IDE.....	52
Sketches and Functions.....	55
Digital Output with Blink.....	57
Going Further with Blink.....	59
The Blink Circuit.....	61
Digital Input: Adding a Button.....	68
The Serial Console.....	70
Toggling the LED.....	73
Analog Output.....	75
Analog Input.....	78
I2C Accelerometer.....	82
SPI Screen.....	87
Linux, C++, and the Arduino IDE.....	92
Troubleshooting.....	95
Going Further.....	95
4/Programming in Python.....	97
Introduction.....	97
“Hello, World” in Python.....	98
Functions and Loops.....	101
Installing Libraries.....	102
Blink in Python.....	103
Scripting at Bootup.....	105
Button-Controlled Blink.....	106
Bluetooth-Controlled LED.....	106
Bluetooth Pairing.....	107
Exchanging Information.....	109
The Potentiometer.....	112
I2C Accelerometer.....	113
Installing the Dependencies.....	115
Using the MMA Library.....	115
SPI Screen.....	118
BLE Beacon.....	120
Edison Side.....	120
Smartphone Side.....	122
Going Further.....	123
5/Teach Edison to See.....	125
Introduction.....	125
Materials List.....	126

OpenCV.....	127
Extracting Colored Objects.....	127
Viewing Images.....	128
Face Detection.....	130
Webcam.....	133
Snapping Photos.....	134
Recording Video.....	135
Streaming Video.....	137
Processed Streaming Video.....	140
Going Further.....	141
6/Exploring Sound.....	143
Introduction.....	143
Materials List.....	143
Connecting a Headset.....	144
Playing and Recording Sounds.....	145
Makeshift MP3 Player.....	146
Recording Audio with Python.....	148
Basic Recording.....	148
Thresholding.....	150
Speech Recognition.....	151
Controlling Devices.....	153
Going Further.....	156
7/Conclusions.....	157
Linux Flavors.....	157
Programming Languages.....	159
Node.js.....	159
C and C+.....	160
The Intel XDK IoT Edition.....	162
Shutdown Now.....	163
A/Materials.....	165
Glossary.....	169
Index.....	177

Preface

Intel Edison is ... well, it's hard to say. Intel Edison is so many different things. If someone told you about Edison 10 years ago, you would have thought that person was utterly insane.

At its core, Intel Edison is a very powerful and cheap little computer. Priced at around \$50 USD, this dual-core Atom processor is comparable to an entry-level Windows 8 tablet. At only a small fraction of the cost, it contains all the software amenities of modern-day computing. The system contains 1 GB of memory and 4 GB of storage to handle heavy computational tasks and data logging.

Intel Edison is a communications device. It comes integrated with Bluetooth and WiFi capabilities and preinstalled software to run both.

Intel Edison is a hardware development platform. It contains an Intel Quark microcontroller that allows you to program and control connected devices and circuits. It's also Arduino-compatible, meaning the vast majority of shields, code examples, and libraries that have been specifically designed for Arduino will work as is with Edison, too.

Finally, Intel Edison is an embedded device, designed for the *Internet of Things (IoT)* and wearable technology. Even though Intel Edison is a computer, communications device, and hardware development platform, it weighs only eight grams, is approximately the size of a postage stamp, and can run in low power modes.

While on the subject, it's worth taking a moment to discuss what Edison is *not*. Although Edison is a standalone computer

running a full Linux operating system, it's meant to be the brains of your connected and wearable devices—not a laptop or desktop in the traditional sense. For this reason, Edison does not interface with a display, keyboard, or mouse on its own; you connect to Edison through a host computer and load your software directly on the module. Edison is not a great choice if you're looking to build a home media console or old-school arcade game (though both could still be done, with a lot of work), but it's a great candidate for a personal assistant, smart watch, automated robot, smart-home controller, or basically any other electronic system you can imagine. What will you make?



Total Cost

The Edison module alone costs around \$50, and if you add the accompanying components to complete every exercise in this book, the total cost is around \$200. For a complete materials list, see [Appendix A](#).

What You Can Do with It

As you can see, Intel Edison is a versatile platform that can be used for a great many applications:

Learn about computers

Edison is a great and inexpensive way to learn more about computers at the lowest level. With Edison, you can learn Linux, configure your operating system, learn about kernels and builds, and install and program drivers for your accessories. You can learn about WiFi and Bluetooth networking by programming Edison to be a dynamic wireless hotspot or Bluetooth beacon. [Chapter 2](#) provides an introduction to Linux and the specific operating system (OS) on Edison. [Chapter 3](#) and [Chapter 4](#) touch on Bluetooth commands.

Learn to program

Edison is an amazing tool for learning to program; it comes preloaded with many different compilers and interpreters,

and installing more is a breeze. From [Chapter 4](#) onward, this book focuses mainly on programming in Python, but Edison also supports Java, node.js, C, C++, and many more languages. Even Arduino programming is expanded with Edison. Standard Arduino programs consist of compiled C++ based on the *avrlibc* library, but Edison exposes the Arduino IDE to the full C++ standard programming environment. With these additional libraries, you can make system calls and tap into the power of Linux within your Arduino sketches, making Arduino for Edison a powerful tool. You'll see how in [Chapter 3](#).

Make

You can use Edison to program your electronics projects. Edison and its accompanying ecosystem support the same inputs and outputs (I/Os) as the Arduino Uno and integrate seamlessly with almost any existing Arduino project (more on this in [Chapter 1](#) and [Chapter 3](#)). With its size, connectivity, and computing power, you can build elaborate and computationally heavy systems that interface directly with the web or store data on board. In [Chapter 3](#), you'll create a range of electronic circuits powered by Edison and program them using the Arduino IDE. Because programming electronics on Edison is not limited to the Arduino IDE, you recreate these same projects in [Chapter 4](#) by programming them in Python. After that, you'll interface some of these electronics with other computational tasks, leveraging the full power of Edison.

Go to market

Edison was created specifically to lower the barrier to entry for makers hoping to take their ideas and prototypes all the way to product. Edison interfaces with a variety of breakout boards (discussed in [Chapter 1](#)), allowing you to rapidly prototype on one while resting assured that your final design will work on another. In this way, you can develop and program without worrying about the final form factor; Edison will handle this for you.

What's Been Done with It

Sometimes it's hard to decide what to do with such an interesting new device, especially one with such a range of options. Although Edison has been, at the time of this writing, in production for under a year, a wealth of amazing projects have already been created and you can look to for inspiration:

- [Intel's project gallery](#)
- [Hackster's gallery](#)
- [Intel's Make It Wearable finalists](#)
- [Hackaday.io](#)

Who This Book Is For

This book is an introductory tutorial for Intel Edison. It is meant to showcase the versatility of the product and therefore spans a wide range of topics, from Linux to hardware interfaces to Python programming. You don't need experience in any of these topics to get started, just a little curiosity and a desire to build new things. The only thing you'll need are some basic computer skills: the ability to move and manipulate files, search within your computer, and install software.

The aim of this book is to help you get started designing, building, and programming end-to-end systems with Edison. This book is in no way a comprehensive guide to systems engineering or computer science—there are many resources available should you want to delve deeper into either of those topics. Instead, this book is meant to inspire you, so that you can take your ideas and concepts very quickly to reality.

Conventions Used in This Book

The following typographical conventions are used in this book:

Italic

Indicates new terms, URLs, email addresses, filenames, and file extensions.

Constant width

Used for program listings, as well as within paragraphs to refer to program elements such as variable or function names, databases, data types, environment variables, statements, and keywords.

Constant width bold

Shows commands or other text that should be typed literally by the user.

Constant width *italic*

Shows text that should be replaced with user-supplied values or by values determined by context.



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



This element indicates a warning or caution.


Using Code Examples

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

We appreciate, but do not require, attribution. An attribution usually includes the title, author, publisher, and ISBN. For example: *Make: Getting Started with Intel Edison* by Stephanie Moyerman (Maker Media). Copyright 2016, 978-1-4571-8759-9.

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

Safari® Books Online

 **Safari**® *Safari Books Online* is an on-demand digital library that delivers expert [content](#) in both book and video form from the world's leading authors in technology and business.

Technology professionals, software developers, web designers, and business and creative professionals use Safari Books Online as their primary resource for research, problem solving, learning, and certification training.

Safari Books Online offers a range of [plans and pricing](#) for [enterprise](#), [government](#), [education](#), and individuals.

Members have access to thousands of books, training videos, and prepublication manuscripts in one fully searchable database from publishers like O'Reilly Media, Prentice Hall Professional, Addison-Wesley Professional, Microsoft Press, Sams, Que, Peachpit Press, Focal Press, Cisco Press, John Wiley & Sons, Syngress, Morgan Kaufmann, IBM Redbooks, Packt, Adobe Press, FT Press, Apress, Manning, New Riders, McGraw-Hill, Jones & Bartlett, Course Technology, and hundreds [more](#). For more information about Safari Books Online, please visit us [online](#).

How to Contact Us

Please address comments and questions concerning this book to the publisher:

Make:
1160 Battery Street East, Suite 125
San Francisco, CA 94111
877-306-6253 (in the United States or Canada)
707-639-1355 (international or local)

Make: unites, inspires, informs, and entertains a growing community of resourceful people who undertake amazing projects in their backyards, basements, and garages. Make: celebrates your right to tweak, hack, and bend any technology to your will. The Make: audience continues to be a growing culture and community that believes in bettering ourselves, our environment, our educational system—our entire world. This is much more than an audience, it's a worldwide movement that Make: is leading—we call it the Maker Movement.

For more information about Make:, visit us online:

Make: magazine: <http://makezine.com/magazine>
Maker Faire: <http://makerfaire.com>
Makezine.com: <http://makezine.com>
Maker Shed: <http://makershed.com>

We have a web page for this book, where we list errata, examples, and any additional information. You can access this page at: <http://bit.ly/gsw-intel-edison>.

To comment or ask technical questions about this book, send email to bookquestions@oreilly.com.

Acknowledgements

There are a great many people I want to acknowledge for helping me, either directly or indirectly, with this book. First, the Intel Edison team, for supplying me with enumerable Edisons to play with over the past 12 months. Special thanks to Ed Ross, Jim

Chase, and Jay Melican for putting up with me during that time (and hopefully moving forward, too).

Second, to my bosses, Jeff Ota and Lakshman Krishnamurthy, who give me the time and freedom to play with Edison as a part of my actual job. To the remainder of the team, who challenge me every day to build something cool and do something with my life—thank you guys, too.

Super special thanks to my technical editors, Jason Wright (two thank yous!) and Esther Kim, for just being amazing people and friends and putting up with the many edits that this book needed. To Jonathan and Julija for playing the role of my non-technical editors and working through all the examples in the book (and also putting up with my needed edits).

Finally, thanks to my parents for letting me be a very curious child and indulging my need to take everything apart. And to my beautiful wife Kelsey, for tolerating my endless string of electronic and coding projects and my tendency to leave half-finished projects all over our otherwise clean house. I love you.