Programming Arduino with LabVIEW

This book covers several projects that you can build using LabVIEW and Arduino. You will learn how to use LabVIEW to control your Arduino projects simply by dragging and dropping blocks in LabVIEW.

The book starts with some basic projects that you will create in order to learn how to interface LabVIEW and Arduino. For example, you will learn how to control a motor from the LabVIEW interface. Then, the book dives into more complex projects, such as building a weather measurement station, making a simple alarm system, and controlling a mobile robot wirelessly via LabVIEW.

Going through the projects of this book will allow you to automate your Arduino projects without writing a single line of code, therefore creating complex projects in little time.

Who this book is written for
If you already have some experience with LabVIEW and want to apply your skills to control physical objects and make measurements using the Arduino sensor, this book is for you. Prior knowledge of Arduino and LabVIEW is essential to fully understand the projects detailed in this book.

What you will learn from this book
- Install LabVIEW and set it up to interface with Arduino
- Automate your Arduino projects with LabVIEW via a USB cable or XBee
- Control a servo motor and a smart power switch from LabVIEW
- Make a simple weather measurement station using Arduino and LabVIEW
- Build a simple wireless alarm system
- Maneuver an Arduino-based robot wirelessly via LabVIEW
- Collect feedback from the robot sensors using Arduino and LabVIEW

In this package, you will find:

- The author biography
- A preview chapter from the book, Chapter 1 'Welcome to LabVIEW and Arduino'
- A synopsis of the book’s content
- More information on Programming Arduino with LabVIEW

About the Authors

Marco Schwartz is an electrical engineer, entrepreneur, and blogger. He has a master's degree in electrical engineering and computer science from SUPELEC in France and a master's degree in micro engineering from the EPFL in Switzerland.

He has more than 5 years of experience working in the domain of electrical engineering. His interests gravitate around electronics, home automation, the Arduino and Raspberry Pi platforms, open source hardware projects, and 3D printing.

He also runs several websites on Arduino, including the http://www.openhomeautomation.net/ website, which is dedicated to building home automation systems using open source hardware.

He has written another book called Arduino Home Automation Projects, Packt Publishing, on home automation and Arduino and also published a book called Internet of Things with the Arduino, on how to build Internet-of-Things projects with Arduino.

Oliver Manickum has been working in the embedded development scene for almost 20 years. His favorite development platform is Arduino. He has delivered thousands of projects and is a big fan of ATMEAL and the Arduino platform. He currently writes high-performance games on mobile platforms; however, developing prototypes with Arduino is his main hobby.

He has also reviewed Netduino Home Automation Projects, Matt Cavanagh.

I would like to thank my wife, Nazia Osman, for her patience while I was building devices that would sometimes burn down parts of our house, over and over again.
Programming Arduino with LabVIEW

Arduino is a powerful electronics prototyping platform used by millions of people around the world to build amazing projects. Using Arduino, it is possible to easily connect sensors and physical objects to a microcontroller, without being an expert in electronics.

However, using Arduino still requires us to know how to write code in C/C++, which is not easy for everyone. This is where LabVIEW comes into play. LabVIEW is software used by many professionals and universities around the world, mainly to automate measurements without having to write a single line of code.

Thanks to a module called LINX, it is actually very easy to interface Arduino and LabVIEW. This means that we will be able to control Arduino projects without having to type a single line of code. The possibilities are endless, and in this book, we will focus on several exciting projects in order for you to discover the key features of the LabVIEW Arduino interface.

What This Book Covers

Chapter 1, Welcome to LabVIEW and Arduino, introduces you to the Arduino platform and the LabVIEW software.

Chapter 2, Getting Started with the LabVIEW Interface for Arduino, shows you how to install and use the LabVIEW interface for Arduino via the LINX module.

Chapter 3, Controlling a Motor from LabVIEW, explains how to make your first real project with Arduino and LabVIEW by controlling a DC motor from LabVIEW.

Chapter 4, A Simple Weather Station with Arduino and LabVIEW, talks about how to automate measurements from several sensors that are connected to the Arduino platform.

Chapter 5, Making an XBee Smart Power Switch, shows you how to make our own do-it-yourself (DIY) version of a smart wireless power switch. We will make a device that can control electrical devices, measure their current consumption, and control the whole power switch from LabVIEW.

Chapter 6, A Wireless Alarm System with LabVIEW, helps you connect motion sensors to an Arduino board and monitor their state remotely via LabVIEW to create a simple alarm system.

Chapter 7, A Remotely Controlled Mobile Robot, teaches you how to use everything you learned so far to control a small mobile robot from LabVIEW. You will be able to wirelessly move the robot and also continuously measure the distance in front of the robot.
National Instruments Corporation, NI, is a world leader when it comes to automated test equipment and virtual instrumentation software. LabVIEW is a product that they have developed, and it is being used in many labs throughout the world. LabVIEW, which stands for Laboratory Virtual Instrument Engineering Workbench, is programmed with a graphical language known as G; this is a dataflow programming language. LabVIEW is supported by Visual Package Manager (VIPM). VIPM contains all the tools and kits to enhance the LabVIEW product.

Arduino is a single-board microcontroller. The hardware consists of an open source hardware board that is designed around the Atmel AVR Microcontroller. The intention of Arduino was to make the application of interactive components or environments more accessible. Arduinos are programmed via an integrated development environment (IDE) and run on any platform that supports Java. An Arduino program is written in either C or C++ and is programmed using its own IDE.

Welcome to programming Arduino with LabVIEW. During the course of this book, we will take you through working with Arduino through NI’s LabVIEW product. The following are what you will need:

- A Windows or Mac-based machine
- Arduino (Uno preferred)
- LabVIEW 13 for students (or any other LabVIEW 13 distribution)

We will work with Servos, LEDs, and Potentiometers in both analog and digital configurations.
Welcome to LabVIEW and Arduino

What makes Arduino ideal for LabVIEW

The Arduino community is extremely vast with thousands and even hundreds of thousands of projects that can be found using simple searches on Google. Integrating LabVIEW with Arduino makes prototyping even simpler using the GUI environment of LabVIEW with the Arduino platform.

Officially, LabVIEW will work with the Uno and Mega 2560; however, you should be able to run it on other Arduino platforms such as the Nano. Building your own Uno board is just as simple as linking up the Arduino to LabVIEW. For detailed instructions on how to build your own Arduino Uno, check out the following URL: http://www.instructables.com/id/Build-Your-Own-Arduino/.

Significance of using LabVIEW

LabVIEW is a graphical programming language built for engineers and scientists. With over 20 years of development behind it, it is a mature development tool that makes automation a pleasure.

The graphical system design takes out the complexity of learning C or C++, which is the native language of Arduino, and lets the user focus on getting the prototype complete.

LabVIEW significantly reduces the learning curve of development, because graphical representations are more intuitive design notations than text-based code. Tools can be accessed easily through interactive palettes, dialogs, menus, and many function blocks known as virtual instruments (VIs). You can drag-and-drop these VIs onto the Block Diagram to define the behavior of your application. This point-and-click approach shortens the time it takes to get from the initial setup to a final solution.

Skills required to use LabVIEW and Arduino

With LabVIEW primarily being designed for and targeted at scientists and engineers, it has not excluded itself from being used by hobbyists. Users who have zero programming skills have been able to take entire projects to completion by just following the intuitive process of dragging controls onto the diagram and setting it up to automate.

We have designed this book to be completely intuitive, using parts that can be easily found at your local electronic store.

To get additional support when using LabVIEW with Arduino, have a look at their forum at https://decibel.ni.com/content.
Downloading LabVIEW

To download or purchase LabVIEW, head out to http://www.ni.com/trylabview/. LabVIEW can also be purchased with an Arduino Uno bundle from SparkFun. At the time of writing this book, the URL for this bundle is https://www.sparkfun.com/products/11225.

If you did not download LabVIEW, do so now. To try LabVIEW without purchasing it, click on Launch LabVIEW.

To install the product, click on all the default options. Note that the Arduino plugin is not found in the initial install of LabVIEW.

Once LabVIEW is installed, launch the Visual Package Manager.

The VIPM will now launch. The VIPM application will look like this:
Welcome to LabVIEW and Arduino

The VIPM will start downloading references to the package bundles into its repository. The status bar is located at the bottom of the application; when the references are downloaded, the status bar will switch to Ready.

Ready ...

**Downloading the Arduino IDE**


This book covers the Windows versions of LabVIEW and Arduino; however, the Mac versions will work just as well.

Click on **Windows Installer** to download the Windows version of the Arduino IDE.

At the time of writing this book, the current version of Arduino IDE is 1.5.8.

To install the product, click on all the default options.

Once the Arduino IDE is installed, click on the shortcut shown here to launch the application:
The Arduino IDE will launch with the following screen:

```java
void setup() {
  // put your setup code here, to run once:
}

void loop() {
  // put your main code here, to run repeatedly:
}
```

Now that the default settings for each of the applications are set up and launched, we are ready to start programming in each application.
Summary
In this chapter, you learned more about LabVIEW and Arduino. We also installed all the software that we need to get LabVIEW and the Arduino IDE up and running. In the next chapter, we will get the Arduino package for LabVIEW installed and upload a basic sketch to the Arduino board.
Where to buy this book

You can buy Programming Arduino with LabVIEW from the Packt Publishing website. Alternatively, you can buy the book from Amazon, BN.com, Computer Manuals and most internet book retailers. Click here for ordering and shipping details.