Intel Galileo Blueprints

The Intel Galileo board was designed to add the power of an Intel processor to the simplicity of the Arduino platform. Intel Galileo gives you the freedom to create a wide range of DIY projects. Intel Galileo Blueprints will be a detailed guide that covers several projects based on the Intel Galileo board, exploiting the full potential of the board.

You will first go through how to set up the development environment for the Galileo board. Next, you will connect different kinds of sensors to the Galileo board, and learn how to use the SD card reader of the board. You will then connect actuators to the Galileo board, like a relay and a servomotor, and write simple software to control these components. Later, you will access the Galileo board remotely in order to monitor the measurements done by the board and send the measured data to a Twitter feed at regular intervals. Finally, you will move on to more advanced topics, such as building a complete home automation system, building a mobile robot controlled by the Intel Galileo board and computer vision applications such as face recognition.

Who this book is written for

If you are an experienced developer using classic Arduino boards and would like to extend your knowledge to the Intel Galileo board and polish your project building skills, this book is for you.

What you will learn from this book

- Monitor data remotely using the onboard Ethernet connection
- Control outputs using the Galileo board and control the board remotely
- Store and plot monitored data in the cloud
- Access your Intel Galileo projects from anywhere in the world
- Automate your garden and monitor it from the cloud
- Create a whole home automation system using the Galileo board as the hub
- Build a mobile robot based on your Galileo board


Marco Schwartz

Discover the true potential of the Intel Galileo board for building exciting projects in various domains such as home automation and robotics

$ 44.99 US
£ 29.99 UK

Prices do not include local sales tax or VAT

Community Experience Distilled

Free Sample
In this package, you will find:

- The author biography
- A preview chapter from the book, Chapter 1 'Setting Up the Galileo Board and the Development Environment'
- A synopsis of the book’s content
- More information on Intel Galileo Blueprints
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 EPFL in Switzerland.

Marco 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 runs several websites around Arduino, including the Open Home Automation website, which is dedicated to building home automation systems using open source hardware.

He is the author of two other books, namely Home Automation with Arduino and Internet of Things with the Arduino Yun, both by Packt Publishing.
Preface

The Intel Galileo board is an amazing development board for all your DIY electronic projects. The board combines the power of an Intel processor with the simplicity of the Arduino platform. This makes it the perfect board for all sorts of projects, especially projects requiring complex interactions with cloud-based services, making it the ideal platform for Internet of Things applications.

In this book, we will start from simple projects that can be done with most Arduino boards. However, even at this point, we will use the advanced features of the Galileo board.

Later, we will use the Galileo board for more complex applications in fields such as the Internet of Things, home automation, and robotics.

What this book covers

Chapter 1, Setting Up the Galileo Board and the Development Environment, demonstrates how to completely set up the development environment to build and use all the projects that you will find in this book.

Chapter 2, Creating a Weather Measurement and Data Logging Station, covers how to use the inputs of the Intel Galileo board. As an example, we will make a simple weather measurement station that will log data on an SD card.

Chapter 3, Controlling Outputs Using the Galileo Board, covers how to control different devices that can be connected to the Galileo board, such as a servomotor.

Chapter 4, Monitoring Data Remotely, teaches you how to use the Ethernet port of the Galileo board and create a measurement station that can be accessed from your local network.
Chapter 5, *Interacting with Web APIs*, covers how to connect the Galileo board to the Internet and interact with Web APIs to add more functionalities to the board.

Chapter 6, *Internet of Things with Intel Galileo*, covers using the Galileo board to create applications in the very exciting field of Internet of Things.

Chapter 7, *Controlling Your Galileo Projects from Anywhere*, teaches you how to control your Galileo projects from any web browser, wherever you are in the world.

Chapter 8, *Displaying the Number of Unread Gmail E-mails on an LCD Screen*, lets you use what you learned so far and build an application to display the number of unread e-mails you have in your Gmail inbox on an external LCD screen.

Chapter 9, *Automated Remote Gardening with Intel Galileo*, covers building another application based on the Galileo board—a complete management system for garden irrigation. You will also be able to monitor it from anywhere in the world.

Chapter 10, *Building a Complete Home Automation System*, lets you use what you learned so far in this book to build a project in an exciting field—home automation. We will see how to use the Galileo board as the hub of a home automation system.

Chapter 11, *Building a Mobile Robot Controlled by the Intel Galileo Board*, demonstrates the use of the Galileo board as the "brain" of a mobile robot.

Chapter 12, *Controlling the Galileo Board from the Web in Real Time Using MQTT*, will let you discover the MQTT protocol that we will use to control the board in real time from a web browser.
Setting Up the Galileo Board and the Development Environment

*Intel Galileo Blueprints* is for Arduino and electronics hobbyists who want to bring their electronic Do It Yourself (DIY) projects to the next level, using an Intel-based Arduino board—the Intel Galileo.

This book will teach you how to develop the Galileo software and how to connect the sensors for the board. It will be your guide on how to integrate the board into an Internet of Things framework. Indeed, many of the projects you will find in this book will be about how to connect your Galileo board to web services and monitor it remotely.

It will teach you how to create applications involving mobile robot control, home automation, remote data monitoring, and much more. This book will help you in the first steps of your Galileo projects and it will lead you closer to your mission of making great electronic creations for the world.

In this chapter, you will learn:

- Introduction to Arduino
- The Intel Galileo board
- Setting up the development environment
What is Arduino?

Arduino is an open-source single-board microcontroller, which is used in building electronics projects. It can be connected to sensors, LEDs, motors, and other devices to create an interactive display, analysis kits, and anything else electronics-based that you can think of.

Arduino has been popular among students, hobbyists, and even professionals ever since it was first introduced in 2005.

The first Arduino started as a project for the students of the Interaction Design Institute Ivrea. A hardware thesis was contributed for Hernando Barragan's (a Columbian student) wiring design. When the wiring platform was completed, a team of researchers and developers worked on the thesis to create a lighter and less expensive prototype to be available to the open source community. The five-man team that created the prototype Arduino board was led by Massimo Banzi.

Types of Arduino boards

Before the Intel-based boards, the Arduino platform was composed of either an 8-bit Atmel AVR microcontroller or a 32-bit Atmel ARM. The latest Arduino models have 6 analog inputs and 14 digital I/O pins with a USB interface.

However, as with all developing technologies, Arduino boards have evolved a lot. Here are the Arduino versions that are commercially available along with their basic features:

<table>
<thead>
<tr>
<th>Name</th>
<th>Release date</th>
<th>Processor</th>
<th>I/O</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Processor</td>
<td>Digital I/O (pins)</td>
</tr>
<tr>
<td>Arduino Zero</td>
<td>May 15, 2014</td>
<td>ATSAMD21G18A(Cortex-M0+)</td>
<td>14</td>
</tr>
<tr>
<td>Intel Galileo</td>
<td>October 17, 2015</td>
<td>Intel® Quark SoC X1000 Application Processor</td>
<td>14</td>
</tr>
<tr>
<td>Arduino Yún</td>
<td>September 10, 2013</td>
<td>Atmega32U4</td>
<td>14</td>
</tr>
<tr>
<td>Arduino Esplora</td>
<td>December 10, 2012</td>
<td>Atheros AR9331</td>
<td></td>
</tr>
<tr>
<td>Arduino Micro</td>
<td>November 8, 2012</td>
<td>ATmega32U4</td>
<td>20</td>
</tr>
<tr>
<td>Name</td>
<td>Release date</td>
<td>Processor</td>
<td>Digital I/O (pins)</td>
</tr>
<tr>
<td>------------------</td>
<td>--------------</td>
<td>-------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>Arduino Due</td>
<td>October 22, 2012</td>
<td>ATSAM3X8E(Cortex-M3)</td>
<td>54</td>
</tr>
<tr>
<td>Arduino Leonardo</td>
<td>July 23, 2012</td>
<td>Atmega32U4</td>
<td>14</td>
</tr>
<tr>
<td>Arduino Ethernet</td>
<td>July 13, 2011</td>
<td>ATmega328</td>
<td>14</td>
</tr>
<tr>
<td>Arduino Uno</td>
<td>September 24, 2010</td>
<td>ATmega328P</td>
<td>14</td>
</tr>
<tr>
<td>Arduino Mega2560</td>
<td>September 24, 2010</td>
<td>ATmega2560</td>
<td>54</td>
</tr>
<tr>
<td>Arduino Fio</td>
<td>March 18, 2010</td>
<td>ATmega328P</td>
<td>14</td>
</tr>
<tr>
<td>Arduino (Pro) Mini</td>
<td>August 23, 2008</td>
<td>ATmega168(Pro uses ATmega328)</td>
<td>14</td>
</tr>
<tr>
<td>LilyPad Arduino</td>
<td>October 17, 2007</td>
<td>ATmega168V or ATmega328V</td>
<td>14</td>
</tr>
<tr>
<td>Arduino Pro</td>
<td></td>
<td>ATmega168 or ATmega328</td>
<td>14</td>
</tr>
</tbody>
</table>

Types of Arduino Boards

Each of these board versions differ in voltage, processor frequency, dimensions, flash memory, Electrically Erasable Programmable Read-Only Memory (EEPROM), and Static Random Access Memory (SRAM).

Among these Arduino versions, the Galileo Board is the first board to be based on Intel x86 architecture.

The Intel Galileo board

Galileo is based on 32-bit Quark SoC x1000, Intel's first ever product from the Intel Quark family, which features low power small-core products. The board is hardware compatible with Arduino's expansion cards (called shields) and is software compatible with Arduino's Integrated software Development Environment (IDE). It can run on Linux, Mac OS X, and Microsoft Windows.
Currently, there are two versions of Intel Galileo—Gen 1 and Gen 2. The difference between the two lies in the **General-Purpose Input/Output (GPIO)** pins, the latter having improved performance and compatibility with Arduino shields and accessories.

Here is a picture of the Intel Galileo Gen 2 board:

![Intel Galileo Gen 2 board](image)

You will notice that Intel Galileo has 3.3V and 5V power supply ports, 14 digital pins, 6 analog pins, In-Circuit Serial Programming (ICSP) header, and Universal Asynchronous Receiver/Transmitter (UART) ports. Galileo has similar basic port locations as the Arduino Uno R3, but it features components that stretch its capabilities beyond the usual Arduino shield system. You can find a microSD slot, RS-232 serial port, USB port, a full-size mini-PC Express slot, Joint Test Action Group (JTAG) header, an 8-byte NOR flash, and a 100Mb Ethernet port.

Here is a short description of the Intel Galileo facility features:

- The default operating voltage of Galileo is 3.3V, but you can modify the jumper on the board to translate the voltage to 5V. The 5V option allows you to be able to use 5V Uno shields. Ensure that you check the state of this jumper before starting any project using the board.
- The 400MHz 32-bit Intel microprocessor is simple to program, can support **Advanced Configuration and Power Interface (ACPI)** compatible CPU sleep states and runs on an integrated **Real-Time Clock (RTC)**.
Among the 14 digital I/O ports, you can utilize 6 as Pulse Width Modulation (PWM) outputs. PWM is used to control the power sent to electrical devices: for example, to modulate the intensity of a LED or the speed of a DC motor. The analog pins have 12-bit resolution via an AD7298 A-to-D converter.

The 6-pin ICSP header is situated on the same location as the other Arduino boards, so you can easily plug existing shields. The ICSP supports SPI communication.

The Serial Peripheral Interface (SPI) port's default frequency is 4MHz to directly support the Arduino Uno shields, but you can program it to 25MHz. Note that the Galileo SPI acts as a master and cannot be a slave to other SPIs. It can be a slave to the USB client connector, though.

The Peripheral Component Interconnect (PCI) Express slot is compliant with PCIe 2.0 and works with PCIe cards with an optional converter plate. Any standard module can be connected to provide Bluetooth, Wi-Fi, or cellular connection.

The USB 2.0 port can support up to 128 USB endpoint devices.

The 10-pin Standard JTAG header is present for debugging.

UART TTL is available for serial communication, while a second UART is available via a 3.5mm jack for RS-232 support.

The USB Device port supports serial (CDC) USB communication.

Mice, keyboards, and other peripherals can be connected through the USB Host port.

The Ethernet RJ45 connector enables wired network connections.

Access to the micro SD is through the SD library.

Use of TWI/I2C bus is simplified through the Arduino wire library.

Aside from the improved shield compatibility of Galileo Gen 2, it has upgraded the input power range from 7V to 15V instead of just 5V, upgraded resolution of digital pins with PWM, and provides optional 12V PoE support.

In this book, we will use Galileo Gen 2.

Now that you are familiar with the parts and features of Galileo, you will learn how to set up the development environment.
Setting up the development environment

Note that you should not use the same power supply for Galileo Board Gen 1 and Gen 2. Gen 1 is only rated at 3.3V to 5V, while Gen 2 boards are powered up with a voltage between 7V and 15V. Using the Gen 2 power supply on Gen 1 boards will cause permanent damage to the Gen 1 boards.

Let's get started with setting up the development environment. You need to perform the following steps:

1. First, connect the Galileo DC jack to power up your board. You will see the LEDs light up. Then, connect your Galileo board to your computer through the USB client port. Look at the picture here for the connection reference:

![Galileo Board Connection](image)

2. Next, download the Galileo development environment. It is a custom version of the usual Arduino IDE developed by Intel. You may download the IDE at the following link:

   https://communities.intel.com/docs/DOC-22226

3. Install the downloaded software in your computer and launch the application.
4. Before proceeding, ensure that the latest firmware is installed. To update the firmware, follow these steps:

   1. Navigate to **Help | Firmware Update**. You will need to wait for a bit, as this takes several minutes.

   2. Now, load the Blink example in the IDE by navigating to **Examples | Basics | Blink**.

5. The Blink code will load and you will end up with the editor window shown in the following screenshot:

6. Click on **Upload** to load the program.

7. The LED will blink on the board.

That's how easy navigation is on the Galileo interface!
Setting Up the Galileo Board and the Development Environment

If your LED light does not blink, check the following:

1. Confirm that you have the latest Galileo IDE loaded.
2. Ensure that the latest firmware is installed.
3. Check the version of the board that has been selected by navigating to the board in **Tools** | **Board**.
4. Ensure that there are no loose cable connections.

**Summary**

In this chapter, you learned what the Arduino platform is, how it came to be, what it is for, where it is used, and its commercially available versions. Most of all, we found out about one of the latest boards of the Arduino platform and the first to use the Intel microprocessor: the Intel Galileo board.

We know that there are two current versions of Intel Galileo: Gen 1 and Gen 2. We will focus on Galileo Gen 2 in this book, but Gen 1 will work as well for all the projects of the book. You also learned how to set up the development environment for Intel Galileo and how to update its firmware.

We are just starting to explore the first ever Arduino board to run on an Intel microprocessor and we are just barely scratching the surface. You have so much more to discover and learn!

In the following chapters, you will learn more of Galileo's basic and core functions. You will study structure, variables, and functions of Gen 2. We will explore digital, analog, and communication applications by creating our own DIY projects. We'll also start with the weather measurement and data logging station project in the next chapter.
Where to buy this book

You can buy Intel Galileo Blueprints 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.