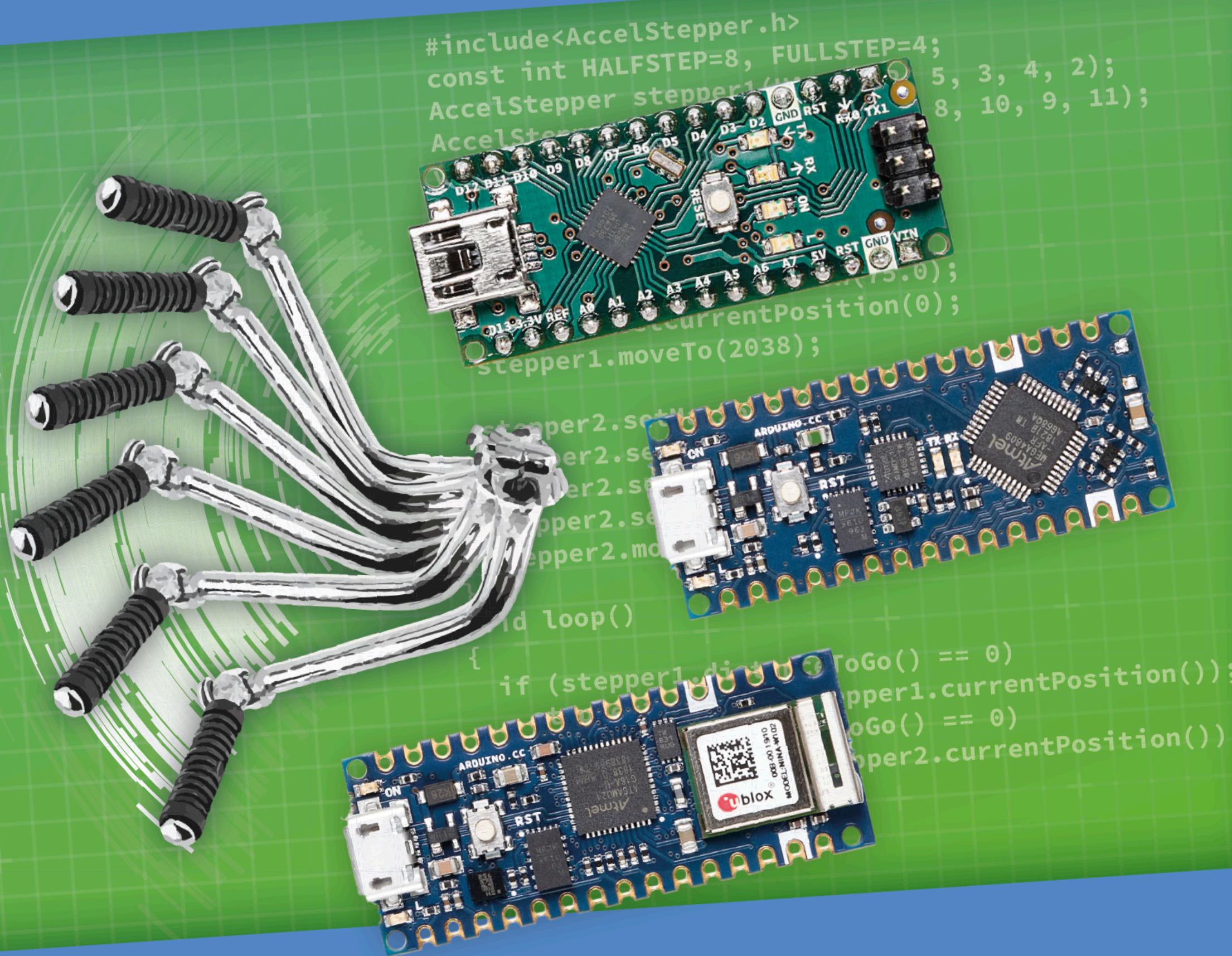


Kickstart to Arduino Nano

Get Cracking with the Arduino Nano V3, Nano Every, and Nano 33 IoT



Ashwin Pajankar

Kickstart to Arduino Nano

Get Cracking with the Arduino Nano V3, Nano Every, and Nano 33 IoT



Ashwin Pajankar

● This is an Elektor Publication. Elektor is the media brand of
Elektor International Media B.V.
PO Box 11, NL-6114-ZG Susteren, The Netherlands
Phone: +31 46 4389444

● All rights reserved. No part of this book may be reproduced in any material form, including photocopying, or storing in any medium by electronic means and whether or not transiently or incidentally to some other use of this publication, without the written permission of the copyright holder except in accordance with the provisions of the Copyright Designs and Patents Act 1988 or under the terms of a licence issued by the Copyright Licensing Agency Ltd., 90 Tottenham Court Road, London, England W1P 9HE. Applications for the copyright holder's permission to reproduce any part of the publication should be addressed to the publishers.

● **Declaration**

The Author and Publisher have used their best efforts in ensuring the correctness of the information contained in this book. They do not assume, and hereby disclaim, any liability to any party for any loss or damage caused by errors or omissions in this book, whether such errors or omissions result from negligence, accident, or any other cause.

All the programs given in the book are Copyright of the Author and Elektor International Media. These programs may only be used for educational purposes. Written permission from the Author or Elektor must be obtained before any of these programs can be used for commercial purposes.

● British Library Cataloguing in Publication Data

A catalogue record for this book is available from the British Library

● **ISBN 978-3-89576-509-4** Print
ISBN 978-3-89576-510-0 eBook

● © Copyright 2022: Elektor International Media B.V.
Editor: Jan Buiting
Prepress Production: D-Vision, Julian van den Berg

Elektor is part of EIM, the world's leading source of essential technical information and electronics products for pro engineers, electronics designers, and the companies seeking to engage them. Each day, our international team develops and delivers high-quality content - via a variety of media channels (including magazines, video, digital media, and social media) in several languages - relating to electronics design and DIY electronics. www.elektormagazine.com

Contents

Acknowledgements	9
Dedication	9
Preface	10
Chapter 1 • Introduction to the Arduino Platform and the Arduino Nano	11
The Microcontroller	11
Popular MCU Families	11
Arduino	12
The Arduino "Classic" Family	12
The Arduino "MKR" Family	12
The Arduino "Nano" Family	13
The Arduino "Pro" Family	13
Arduino is for Everyone	13
The Arduino Ecosystem	13
Arduino Software	14
Official Arduino Boards	14
Arduino Clones and Derivatives	14
Arduino Counterfeits	14
The Arduino Nano and Arduino Nano Every	14
Arduino Nano and Arduino Nano Every Pinouts	16
Powering the Nano and Nano Every MCU boards	19
The Arduino IDE	20
Installing the stable version of Arduino IDE	26
Installation of Arduino IDE on Raspberry Pi OS	30
Working with the Boards Manager	32
Working with the Arduino Nano Every	35
Working with the Arduino Nano	39
Summary	41
Chapter 2 • Playing with Electronics	42
Basics of Programming with Arduino IDE	42
Blink in the built-in LED	43

Working with the basic electronic components	45
Breadboards and power supplies.	46
Jumper cables	50
Light Emitting Diodes	51
Resistors	52
Pushbuttons	53
Improving the LED blink sketch with Functions	54
Building your first circuit on a breadboard	56
Circuits using Nano	58
Working with multiple LEDs	59
Adding a pushbutton to the circuit.	63
Working with RGB LEDs	65
Using Arduino Nano boards with expansion shields	67
Summary	70
Chapter 3 • Assorted Buses and the Analog Input	71
Parallel and Serial Data Transfer	71
Arduino Serial.	73
SPI and I ² C	75
Analog Input	75
Plotting multiple variables	80
Summary	81
Chapter 4 • Pulse Width Modulation and Driving Unipolar Stepper Motors with Digital I/O.	82
The Concept of Pulse Width Modulation	82
PWM with Arduino Nano.	84
Working with a Servo Motor	89
Working with the 28BYJ-48 Unipolar Stepper Motor and the ULN2003A Motor Driver. . .	92
Using a Custom Library for Stepper Motors	98
Working with Multiple Stepper Motors	101
Summary	102
Chapter 5 • Plotting Geometric Art on an External Display	103
The Ilitek 9225 Driver IC and the Display.	103

Programming the Display	104
Summary	143
Chapter 6 • Working with a Buzzer and a Sensor	144
Working with a Buzzer	144
Working with a Joystick	149
Working with DS18B20 Temperature Sensor.	151
Summary	153
Chapter 7 • Working with the Arduino Nano 33 IoT	154
Introduction to the Nano 33 IoT board.	154
Getting Started	157
Working with WiFININA library	159
A Telnet-based Group Chat Server	172
Pinging a Remote Server	174
A simple Web Client.	176
Working with a Real-Time Clock	177
Using the DS18B20 Temperature Sensor Jointly with the RTC.	179
Visualizing the Temperature graph with ThingSpeak	185
Programming the Built-in IMU.	190
Summary	191
Conclusion	191
Index	192

Acknowledgements

I wish to thank my friend Anuradha for encouraging me to author this book. I also want to express my heartfelt gratitude towards Ferdinand TeWalvaart, Jan Buiting, Alina Neacsu, and Shenja Panik from the Elektor team for guiding me through each phase of the publication process. This is my second book with Elektor International; Media and it is a great experience to perform with their publishing team. Finally, I want to show appreciation to everyone directly and indirectly associated with this project.

Dedication

This book is dedicated to the memory of Prof. Govindarajulu Regeti (July 9, 1945 — March 18, 2021)



Popularly known to everyone as RGR, Professor Govindarajulu obtained his B. Tech in Electrical and Electronic Engineering from JNTU Kakinada. He also earned his M. Tech and Ph. D. from IIT Kanpur. Prof. Govindarajulu was an early faculty member of IIIT Hyderabad and played a significant role in making IIIT Hyderabad the top-class institution it is today. He was by far the most loved and cheered for faculty member of the institute. He was full of energy to teach and full of old-fashioned charm. There is no doubt he was concerned with every student as an individual, taking care to know about as well as guide them. He has taught, guided, and mentored many batches of students at IIIT Hyderabad, including the author of this book.

Preface

I have been working with the Arduino since 2016 and I believe it to be an excellent platform not only to learn electronics but also to deploy real-life production systems. Using Arduino boards, I have produced and deployed many real-world applications such as IoT enabled electrical infrastructures at home and workplaces, web-enabled weather monitoring systems, and a robot for welcoming guests in the reception area of my former employer.

I always wished to write a step-by-step and detailed book on one of the most prominent family of microcontroller boards by Arduino, the "Nano". In this book, I have covered three members of this family: Arduino Nano V3, Arduino Nano Every, and Arduino Nano 33 IoT, in happy unison with the Arduino IDE 1.x and Arduino IDE 2 RC5 for the purpose of demonstrating the C and C++ code. I have written the book in a step-by-step and detailed way to explain concepts, build required circuits, and finally write the code. The book covers more than 60 examples to address a variety of major concepts related to the world of electronics and the Arduino Ecosystem in particular.

This book required me to find and digest loads of documentation, attend online tutorials, write code, and post on online forums when I was stuck with some problem. This result accounts for hundreds of manhours of work I spent to finish the project. I have referred many online sources for code and images. I also mentioned those sources with all the licenses whenever and wherever I borrowed and modified the source codes and images.

I finished writing the draft of the book when India (my current resident country) and the world is gradually recovering from the COVID-19 pandemic. This project and the guidance provided by the Elektor team gave me a great sense of purpose and motivation to continue exploring the world of knowledge on Arduino. I hope all the readers will like and enjoy this book as much as I enjoyed writing it. Happy learning and exploring!

Reader Notice. An archive file (.zip) comprising the software examples and Fritzing-style circuit diagrams discussed in the book may be downloaded free of charge from the book's product and resources page on www.elektor.com (search for: book title and author).

Chapter 1 • Introduction to the Arduino Platform and the Arduino Nano

Welcome to the very first chapter. In case you have not read the Table of Contents and the Preface, I strongly recommend that you do so before commencing with this chapter. Let's begin the journey to the amazing world of the Arduino Platform and its Ecosystem. This is an introductory chapter, so you will be learning a lot of concepts needed to build sufficient background to begin with Arduino. This chapter addresses the following topics:

- the microcontroller
- Arduino
- Arduino is for Everyone
- the Arduino Ecosystem
- the Arduino Nano and Arduino Nano Every
- powering Nano and Nano Every microcontroller boards
- the Arduino IDE
- working with the Boards Manager
- working with the Arduino Nano Every
- working with the Arduino Nano

After reading this chapter, you are all set to explore the Arduino Platform and Ecosystem.

The Microcontroller

A microcontroller is usually a tiny, complete computer system on a single integrated circuit (abbreviated as IC). An MCU (MicroController Unit) can have one or more processors, memory, and programmable Input/Output (IO) within a single IC package. In most cases, there are two types of memory included on the chip. The first one is Programmable memory, usually EEPROM (Electrically Erasable Programmable Read-Only Memory), OTP (One Time Programmable) ROM, ferroelectric RAM, or NOR flash. Also, MCUs also have a small amount of RAM (Random Access Memory) where programs are loaded and executed. In short, an MCU is a microprocessor with Program Memory, RAM, and Programmable IO on a single chip. The main difference between microprocessors and MCUs is in their applications. Microprocessors are general-purpose processors used in desktop computing and general-purpose computing. MCUs are usually used in embedded systems and tasked to performing some sort of physical action. For example, MCUs can be employed in printers, vehicle dashboards, or engine controls in cars and aircraft.

Popular MCU Families

Based on their architectures, MCUs can be divided into various families. Here's a list of some of the most common families:

- PIC (Peripheral Interface Controller)
- ARM (Advanced RISC Machines)
- AVR (Alf and Vegard's RISC Processor)
- MSP (Mixed Signal Processor)
- Intel 8051 and derivatives

Arduino

Arduino (<https://www.arduino.cc>) is an open-source MCU platform, meaning it has open-source and free software and open-source hardware. It usually combines an MCU on a single PCB with other features. Arduino is a popular family of MCU based MCU-based boards. Earlier, the Arduino group had also produced a series of Linux-based computers. However, by now they seem to have discontinued them and are solely focusing on MCUs. The Arduino family has a lot of members, and they can be categorized into three subfamilies as follows:

The Arduino "Classic" Family

This family has the most popular, the oldest, and the most successful boards, including (April 2022):

- Arduino UNO Rev3
- Arduino Mega2560 Rev3
- Arduino Leonardo
- Arduino UNO Mini Limited Edition
- Arduino Due
- Arduino Micro
- Arduino Zero
- Arduino UNO Wi-Fi Rev2

You can find them at the URL:

<https://www.arduino.cc/en/hardware#classic-family>

The Arduino "MKR" Family

MKR is a family of Arduino MCU boards that combine an MCU with advanced communications such as Wi-Fi, LoRa, Bluetooth, and Sigfox. They are based on the MCUs that are members of Microchip's SAM D21 Family. These are the members of MKR family:

- Arduino Nano 33 IoT
- Arduino Nano RP2040 Connect
- Arduino Nano BLE Sense
- Arduino Nano 33 BLE
- Arduino Nano Every
- Arduino Nano

You can read more about the MKR family at

<https://www.arduino.cc/en/hardware#mkr-family>

You can also read more about the SAM D21 family at

https://content.arduino.cc/assets/mkr-microchip_samd21_family_full_datasheet-ds40001882d.pdf

The Arduino "Nano" Family

The "Nano" family has a very small physical footprint, and its members range from the basic Arduino Nano Every to advanced boards such as Nano RP2040 Connect.

- Arduino Nano 33 IoT
- Arduino Nano RP2040 Connect
- Arduino Nano BLE Sense
- Arduino Nano 33 BLE
- Arduino Nano Every
- Arduino Nano

We will discuss the **Arduino Nano** and **Arduino Nano Every** boards in detail throughout this book.

You can read more about the Nano family at the URL:

<https://www.arduino.cc/en/hardware#mkr-family>

The Arduino "Pro" Family

This family provides a complete IoT platform and industry-ready solutions. The members include (but are not limited to):

- Portenta H7
- Portenta X8
- Portenta Machine Control

You can read more about the Pro family at the URL

<https://www.arduino.cc/pro/hardware>

Arduino is for Everyone

Anyone can use an Arduino thanks to the following features:

- inexpensive
- cross-platform
- simple, clear programming environment
- open-source and extendable software
- open-source and extendable hardware

You will explore these features in detail in this book in the upcoming chapters.

The Arduino Ecosystem

As I have mentioned in the earlier section, "Arduino" is actually a family of open-source and free software and open-source hardware. Consequently, both software and hardware are open-source and can be extended. There is a mature community of makers and professionals using these tools, as well as many open-source libraries available for various third-party

hardware components that can be interfaced with Arduino. Even you can develop your own library once you are comfortable with the coding of the Arduino platform. Let's dive deeper into the ecosystem of Arduino and explore all the components.

Arduino Software

The Arduino organization has published a free and open-source software and IDE known as the **Arduino IDE**, a desktop application for Linux, Windows, and macOS. It comes with many libraries and example programs for learning the programming "on" the Arduino platform. It supports C and C++ syntax tailor-made for the Arduino platform. All the members of Arduino family can be programmed with this IDE. It is freely available for download and use. Also, it is open-source. Many people have created their own libraries and made them available for everyone.

Official Arduino Boards

All the boards produced, marketed, and sold under the brand **Arduino** are official boards. They have all the Arduino markings. You can procure them from the Arduino eStore located at <https://store.arduino.cc>. You can find the list of the appointed global distributors at <https://store-usa.arduino.cc/pages/distributors>.

Arduino Clones and Derivatives

Since the hardware is open-source, all the schematics for creating the PCBs (Printed Circuit Boards) are available online for free at the Arduino pages for the respective board. Consequently, you can create your own boards and sell them. The boards that emulate the functionality of the original and official Arduino boards are known as **Clones**. They are not marketed under the Arduino brand, their makers promoting them under their own brand(s). Similarly, the Arduino design-based boards that have additional functionality and are promoted under brand names other than Arduino are known as **Derivatives**. The line that distinguishes the clones from derivatives is vague at best.

Arduino Counterfeits

These are based on the official boards and bear the Arduino logo, except they are not manufactured, marketed, or sold by Arduino. These products are known as fakes or counterfeits. You can guesstimate their status from the quality of components and build. Do not buy and encourage counterfeits. They are illegal and unethical products.

The Arduino Nano and Arduino Nano Every

As I have mentioned in an earlier section, **Arduino Nano** is a family of boards within the larger ecosystem. The Nano members are marked by a tiny physical footprint. This book explores the two members of this family, the **Arduino Nano** and the **Arduino Nano Every**, in detail. So, let's get started.

The Arduino Nano is the elementary and oldest member of the Arduino Nano family. The current revision is **Arduino Nano 3.x**. The Arduino Nano Every is a pin-to-pin compatible upgrade for Arduino Nano with a faster processor and more memory. You can use both indiscriminately in your projects. Let's compare the specifications of both boards.

		Arduino Nano	Arduino Nano Every
Microcontroller		ATmega328	ATmega4809
USB connector		Mini-B USB	Micro USB
Pins	Built-in LED pins	13	13
	Digital I/O pins	14	14
	Analog input pins	8	8
	PWM pins	6	5
Serial Communications	UART	RX/TX	RX/TX
	I2C	A4 (SDA), A5 (SCL)	A4 (SDA), A5 (SCL)
	SPI	D11 (COPI), D12 (CIPO), D13 (SCK). Use any GPIO for Chip Select (CS).	D11 (COPI), D12 (CIPO), D13 (SCK). Use any GPIO for Chip Select (CS).
Power	I/O Voltage	5 V	5 V
	Input Voltage	7 V to 12 V	7 V to 18 V
	DC Current per I/O Pin	20 mA	15 mA
Clock Speed		16 Megahertz	16 Megahertz
Memory	SRAM	2 KB	6 KB
	Flash	32 KB	48 KB
	EEPROM	1 KB	256 bytes
Physical Dimensions	Weight	5 grams	5 grams
	Width	18 mm	18 mm
	Length	45 mm	45 mm

You can check the official pages at arduino.cc for more information on Nano, see

<https://store-usa.arduino.cc/products/arduino-nano>

<https://docs.arduino.cc/hardware/nano>

You can download ATmega328 Datasheet from the following URL:

<http://ww1.microchip.com/downloads/en/DeviceDoc/ATmega48A-PA-88A-PA-168A-PA-328-P-DS-DS40002061A.pdf>

You can check the official pages at arduino.cc for more information on the Nano Every, here:

<https://store-usa.arduino.cc/products/arduino-nano-every>

<https://docs.arduino.cc/hardware/nano-every>

You can download the ATmega4809 Datasheet from the following URL:

https://content.arduino.cc/assets/Nano-Every_processor-48-pin-Data-Sheet-megaAVR-0-series-DS40002016B.pdf

After checking the above URLs, you must have realized that the Arduino Nano Every is better than the Arduino Nano. It is even cheaper, so, for all your projects I would recommend using an Arduino Nano Every.

Arduino Nano and Arduino Nano Every Pinouts

Let's study the pinouts of Arduino Nano and Arduino Nano Every. The diagrams shown in Figures 1-1, 1-2, and 1-3 are reproduced from

https://content.arduino.cc/assets/Pinout-NANO_latest.pdf

and

https://content.arduino.cc/assets/Pinout-NANOEvery_latest.pdf.

They are shared under creative commons license which can be found at <https://creativecommons.org/licenses/by-sa/4.0/deed.en>. I have modified them for using here in this chapter.

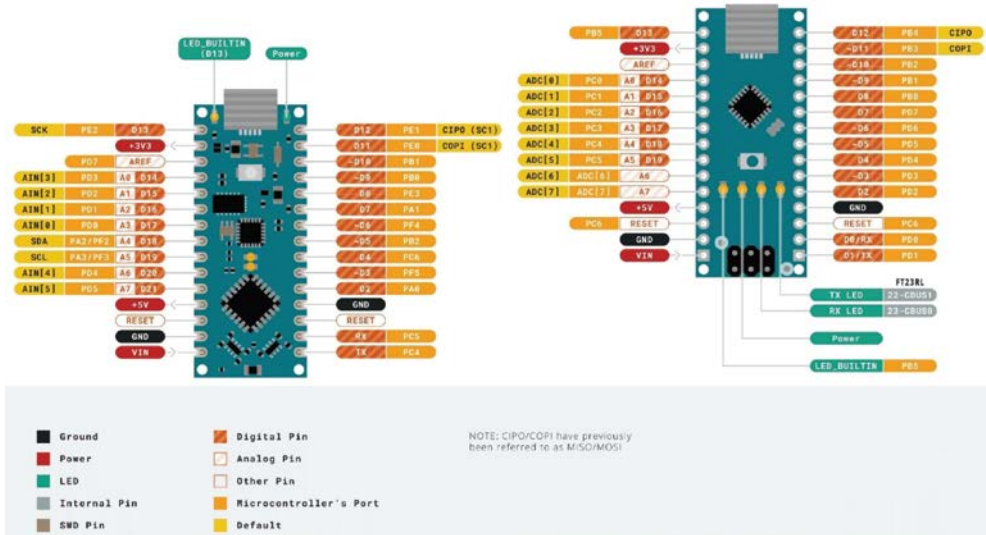


Figure 1-1: Pinouts of Arduino Nano Every (left) and Arduino Nano (right).

In Figure 1-1, at the bottom, the legend explains the meaning of the colors. You need to focus on Digital Pins, Analog Pins, Ground, Power, and LED pins in this diagram.

In Figure 1-2, please focus on the Serial Communications pins allocated to I²C (SDA and SCL), SPI (SS, MOSI/COPI, MISO/CIPO, SCK), and UART (RX and TX),

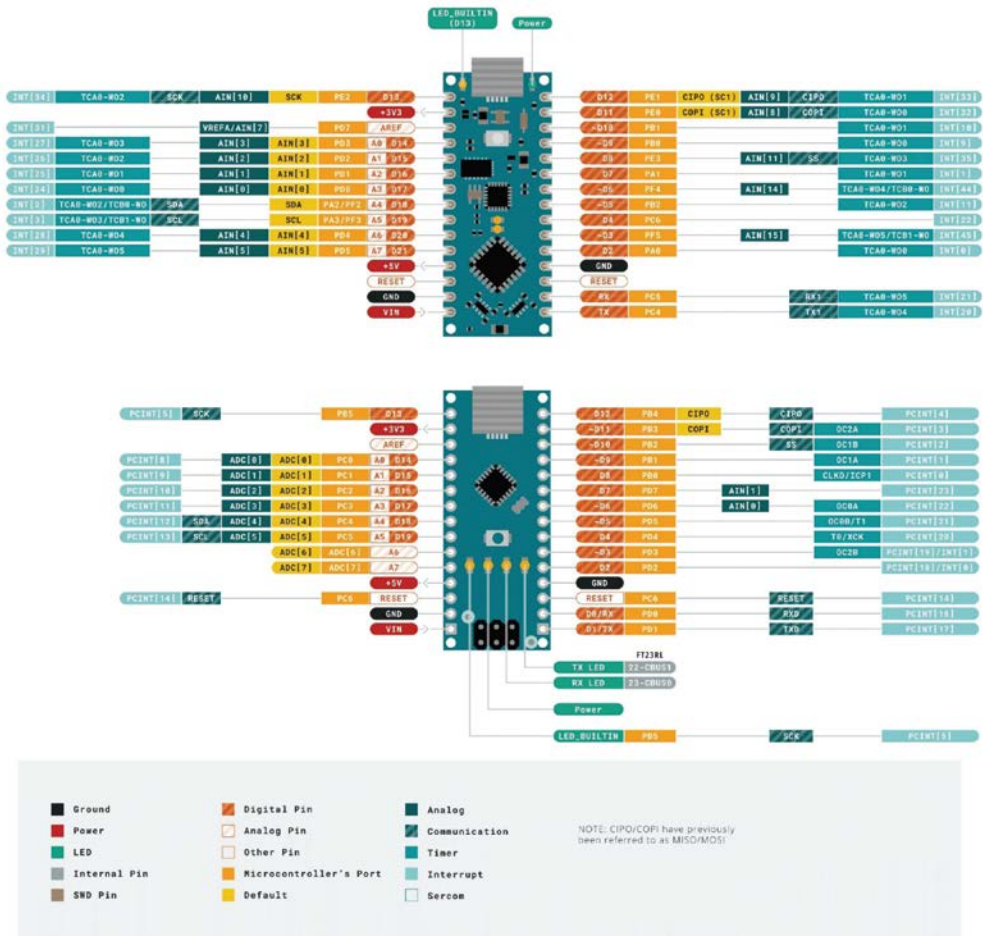


Figure 1-2: Serial communications and interrupt pin description.

In Figure 1-2, the top section shows the Nano Every pin functions, and the middle section, those of the Arduino Nano.

You can study the meaning of additional pins from Figure 1-3.,

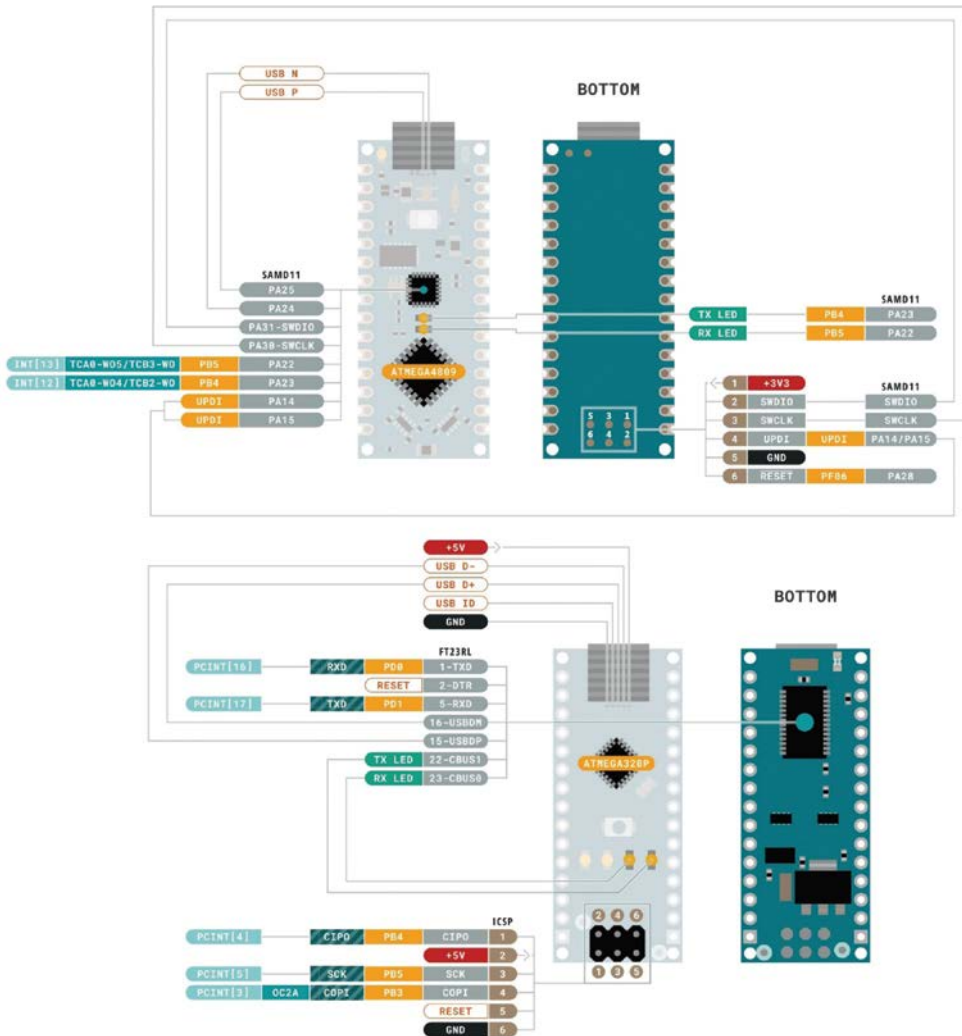


Figure 1-3: Bottom views of the Arduino Nano Every and Arduino Nano boards.

Let's fathom the meanings of the pins shown and described in the diagrams. The pin descriptions of the Nano and Nano Every are almost identical with a few exceptions. Let's see that in detail.

Digital Pins – The pins from D0 through D13 are 14 digital input output pins in all. They operate at 5 volts logic swing Each pin can provide or receive a maximum current of 40 mA. Each pin also has an internal pull-up resistor with a value of 20-50 k-ohms which is disconnected by default.

A few digital pins have additional functions as follows:

UART – These are D0 (**RX**) and D1 (**TX**), used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the pins of an onboard FTDI USB-to-TTL Serial converter chip.

PWM – In the Nano, pins D3, D5, D6, D9, D10, and D11 provide an 8-bit PWM output. In the Nano Every, there are only five pins allocated to PWM: D3, D5, D6, D9, and D10.

SPI – SPI communication is provided by D10 (**SS**), D11 (**MOSI/COPI**), D12 (**MISO/CIPO**), and D13 (**SCK**) on both boards.

On both boards, the on-board LED is connected to D13.

The Nano and Nano Every each have 8 analog inputs (A0 through A7). Each pin provides 10 bits of resolution that enables $10^8 = 1024$ different values for input. The default range for the operational voltage for these pins is 0 V to 5 V. You can change the upper range from 5 V to your desired value by programming.

In the Nano, pins A0 to A5 can also be configured as digital pins D14 through D19.

In the Nano Every, pins A0 to A7 can also be configured as digital pins D14 through D21.

I²C – This functionality is provided by the pins labelled A4/D18 (**SDA**) and A5/D19 (**SCL**). They support I²C using the **Two Wire Interface (TWI)** and use the Wire library.

These are all the important pins on both MCU boards. There are more pins and more meanings to the pins. However, I will not be covering most of those concepts, as they are really out of the scope of this book.

Powering the Nano and Nano Every MCU boards

You can power a Nano with built-in USB with a Mini B type cable and a Nano Every with a USB Micro B type cable as shown in Figure 1-4.

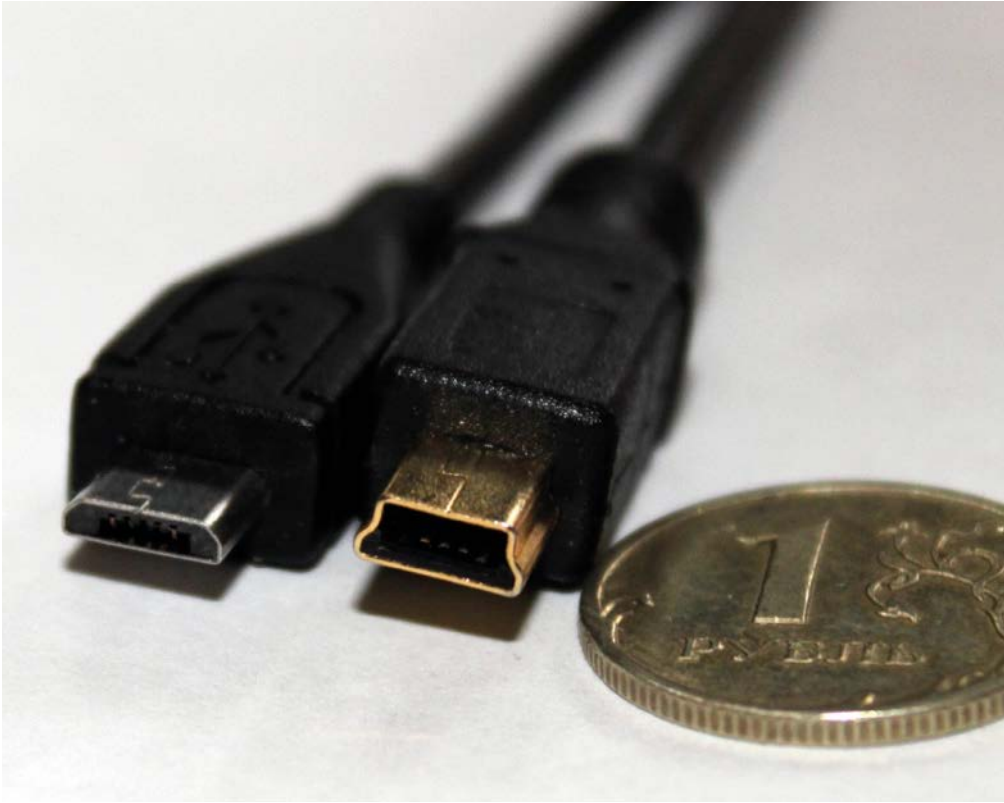


Figure 1-4: USB Micro B and USB Mini B cables. Image provided by [Andrej A. Antonov](#) under the license <https://creativecommons.org/licenses/by-sa/3.0/deed.en>

You can power these boards by plugging one end of the USB cable to the MCU board and the other end to a 5-V battery pack, power supply, or the USB port of a computer. You can also power the boards through their VIN pins, but I do not recommend it yet to the beginning readers. Powering through the USB is the best and hassle-free way to power your Arduino.

The Arduino IDE

Arduino IDE is an open-source and free **Integrated Development Environment** for developing the code for Arduino and other MCU boards. You can download the latest version from <https://www.arduino.cc/en/software>. At the time of authoring this book, the latest version is 1.8.19. However, I am going to try the Release Candidate for version 2.0. This is going to be the new major release of the IDE with a lot of improvements. You can download it by clicking the **MSI Installer** option for Windows as shown in Figure 1-5.

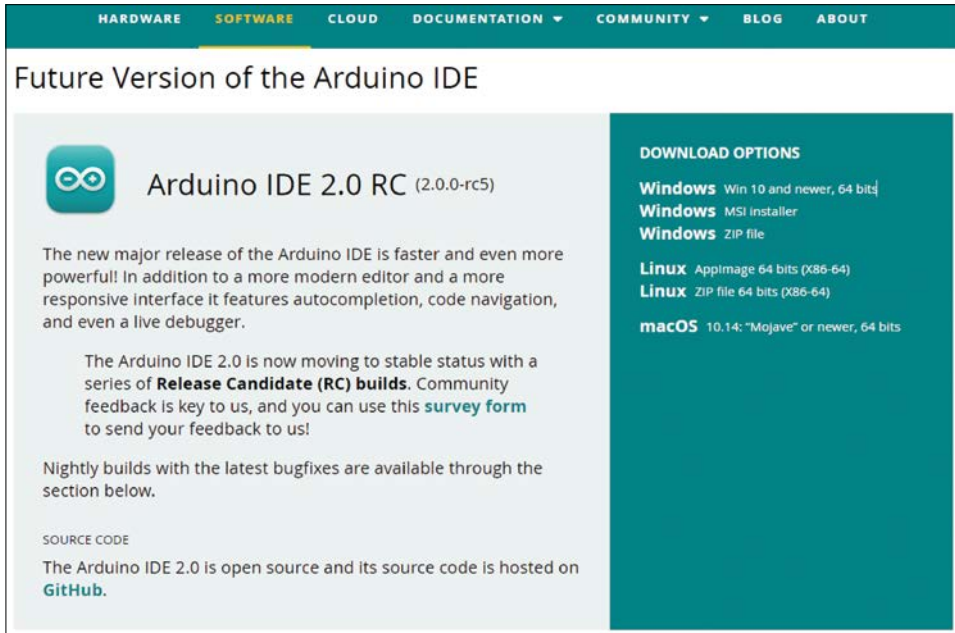


Figure 1-5: Arduino IDE 2.0 Release Candidate (RC).

Options for macOS and Linux distributions are also available. Once you click an appropriate option, the appropriate installation package for your platform will be downloaded. I have a 64-bit Windows computer and consequently downloaded the file **arduino-ide_2.0.0-rc5_Windows_64bit.msi**. **rc5** i.e., **Release Candidate 5**. You can see the file at the bottom bar of your **Chrome browser**, as well as under the **Downloads** option in the browser. You can find the physical file in the **Downloads** directory of your user in the OS. Launch (run) the file and it will show a window like in Figure 1-6.

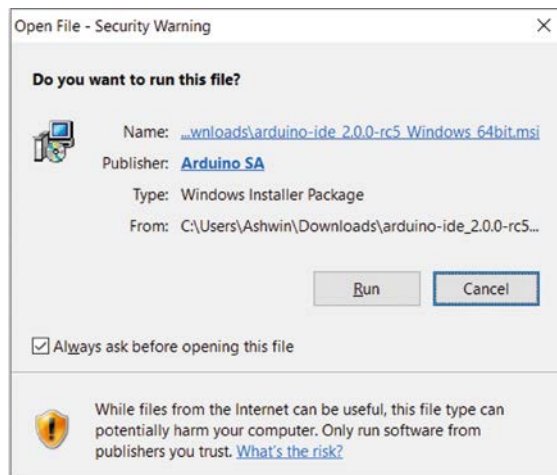


Figure 1-6: Installation window.