

Arduino

Projects' Book



Level 1

Rana Dajani

Arduino Projects' Book - Level 1



Published by **LKD Educational Resources 2023**

Amman - Jordan

Tel: +962 6 5374141

Fax: +962 6 5516404

P.O.Box: 851346

Email: info@lkd.com.jo / info@lkdebooks.com

Websites: www.lkd.com.jo / www.lkdebooks.com

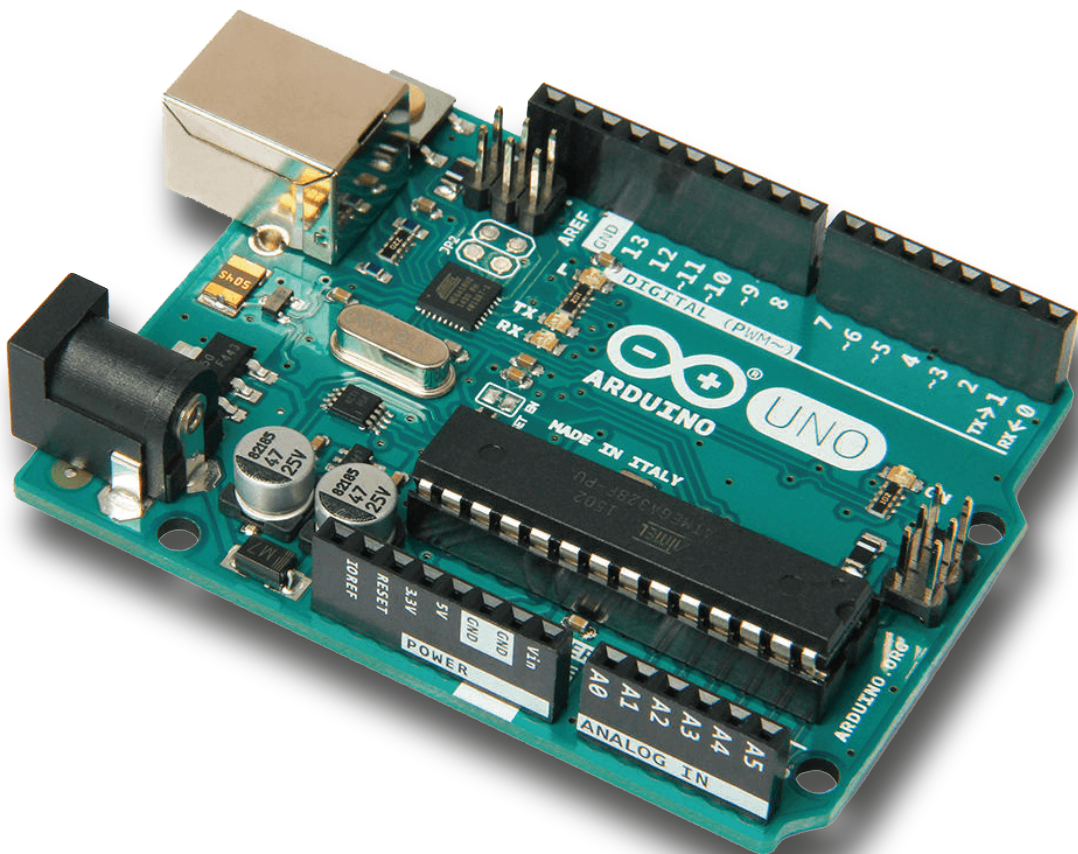
 **Author**

Rana Dajani

ISBN: 2023/9/4939



Arduino Projects' Book



Level 1

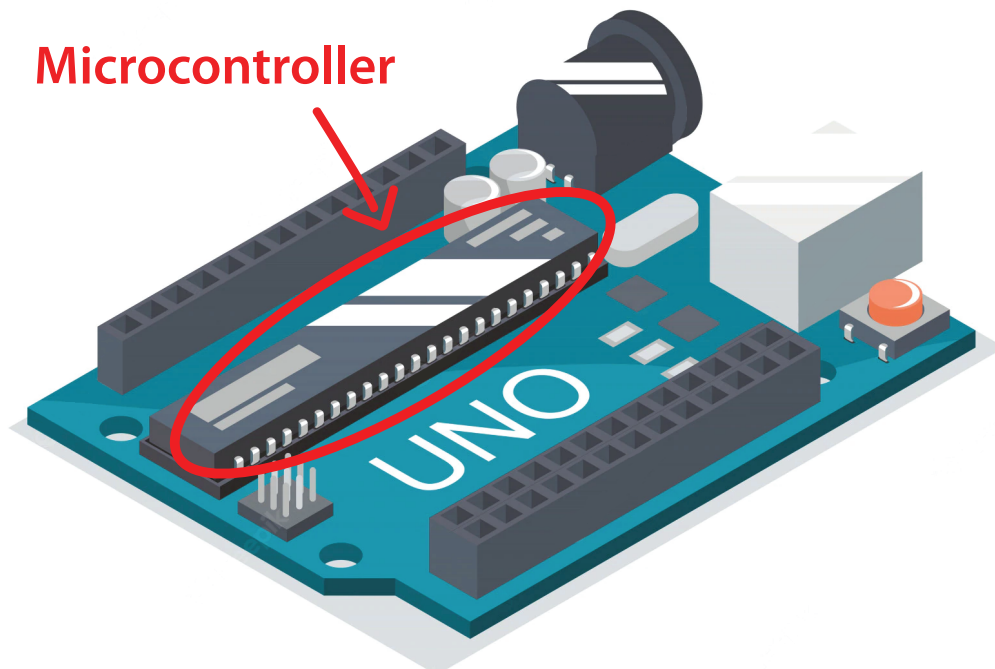
Rana Dajani

Introduction

Have you ever looked at a gadget and wondered how it works? Maybe it was a remote-controlled boat, a vending machine, or an electronic toy? Or have you wanted to create your own robot? Where and how do you start? The Arduino board can help you find some of the answers to the mysteries of electronics in a hands-on way.

Arduino system inspires you to create devices that can interact with the world around you. By using an almost unlimited range of input and output devices, sensors, indicators, displays, motors, and more, you can program the exact interactions required to create a functional device.

Over the years Arduino has been used as the “brain” in thousands of creative projects; by sending a set of instructions to the microcontroller on the board.



You are surrounded by dozens of microcontrollers every day: they are embedded in timers, thermostats, toys, remote controls, microwave ovens, and even some toothbrushes. They have been programmed to sense and control activity using sensors and actuators.

- 1. Sensors listen to the physical world (input devices).** They convert energy that you give off when you press buttons, or wave your arms, or shout, into electrical signals. There are many kinds of sensors (e.g. buttons and knobs are sensors that you touch with your fingers).
- 2. Actuators take action in the physical world (output devices).** They convert electrical energy back into physical energy, like light and heat and movement.
- 3. Microcontrollers listen to sensors and talk to actuators (brain).** They decide what to do based on a program that you write.

Microcontrollers and the electronics you attach to them are just the skeleton that make your projects responsive, but only you can make them beautiful.

Table of contents

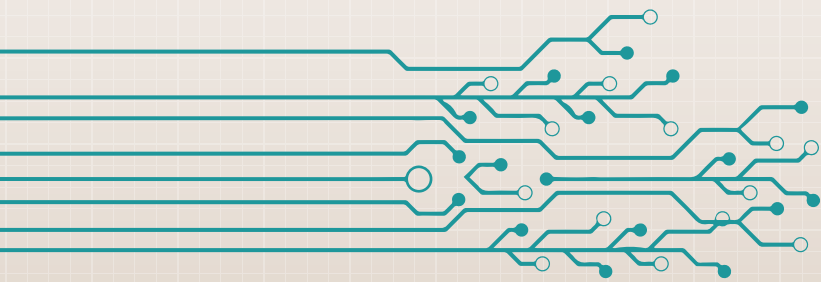
1	Getting Started	8
2	Blinking LED	20
3	Pattern Waves	28
4	Repeat and Resonate	36
5	Press On	44
6	Key Notes	54
7	Stopwatch	62
8	Quick Reflex	70
9	Counter Shake	78
10	Hourglass	88
11	Alarm System	98
12	Spin Range	102

13	LCD Messages	108
14	Crystal Luck	116
15	Pixel Dodge	124

Comprehensive Projects

16	Motion Count	136
17	Morse Code	140
18	Cyclone Spin	144
19	Whac-A-Mole	148
20	Lightsabers	152

Glossary



Lesson

Getting Started

1

🔧 Get to know your tools

Introduction to the microcontroller board...

1. ATmega microcontroller

The 'brain' of your board

2. Analog Input

Pins that use `analogRead()`

3. GND and 5V Pins

Provide **power** (5V) and **ground/GND** (0V) to the circuit

4. Digital Pins

Pins that use `digitalWrite()`, `digitalRead()`, `analogWrite()` –only with PWM (~) symbol pins

5. Reset Button

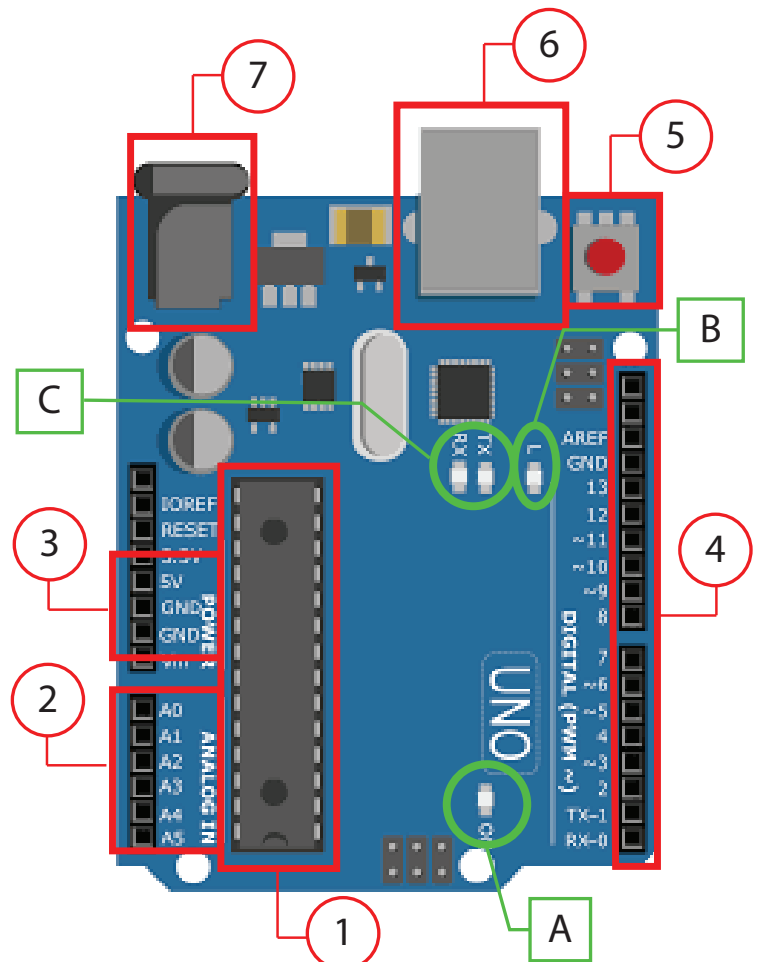
Reset the microcontroller

6. USB Port

Power the board, upload **sketches** (programs), communicate with your program

7. Power Connector

Power the board without plugging USB, using an external power source (e.g. 9V battery)



A. Power LED

Indicates the board is receiving power

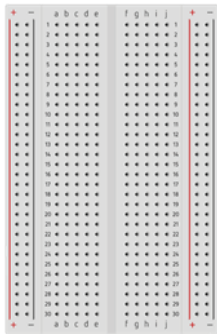
B. Pin 13 LED

Built-in LED actuator

C. TX and RX LEDs

Indicate communication between board and computer. Flickers during sketch upload and communication

Basic Circuit Components



1 Pcs

Breadboard

The breadboard is the primary place you will be building circuits. The horizontal and vertical rows (used for power and ground connections) of the breadboard carry electricity through thin metal connectors under the top layer of plastic. The middle row breaks the connection between the two sides of the board. The holes allow you to connect wires and components together without having to use soldering



100 Pcs

Jump wires

Used to connect components to each other on the breadboard



1 Pcs

USB Cable

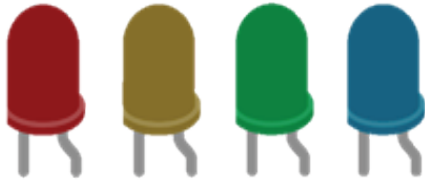
Allows you to connect your board to the computer for programming. It also provides power to the board



1 Pcs

Battery Snap

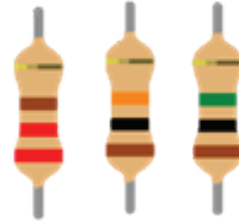
Used to connect a 9V battery to power leads that can be easily plugged into the breadboard or the Arduino



100 Pcs

LED

Light emitting diode that illuminates when electricity passes through in one direction



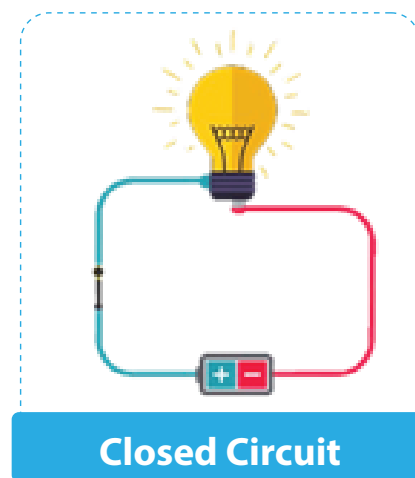
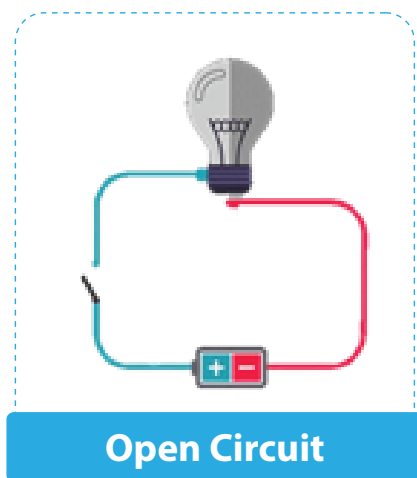
100 Pcs

Resistors

Resist the flow of electrical energy in a circuit, changing the voltage and current

Learning about circuits

A circuit is a complete path around which electricity can flow. It must include a source of electricity (power), such as a battery. Conductive materials, like metal wires, link the positive point of higher potential energy (often referred to as power or +ve) and negative ends of the battery (often referred to as GND or -ve), the point where potential energy is lowest in the circuit. In these type of circuits electricity only flows in one direction (direct current, or DC). In an open circuit, there is a break along the line, and the current (electric flow) stops.

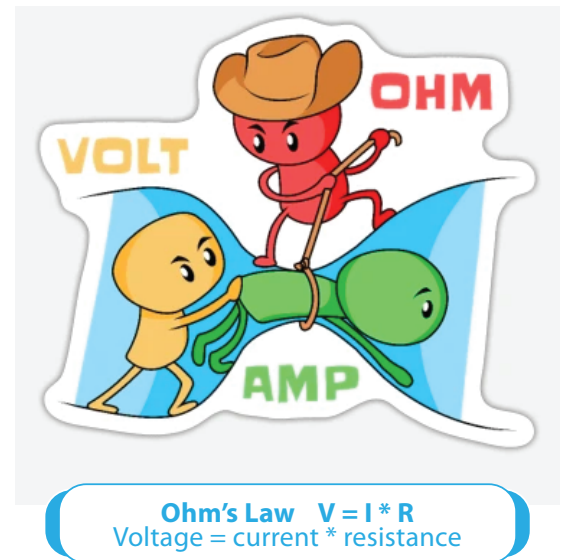


When electric current flows, it can be used by electrical appliances, such as light bulbs, that convert the electrical energy into other forms of energy. Things that convert other forms of energy into electrical energy are called **sensors**, and things that convert electrical energy into other forms of energy are called **actuators**.



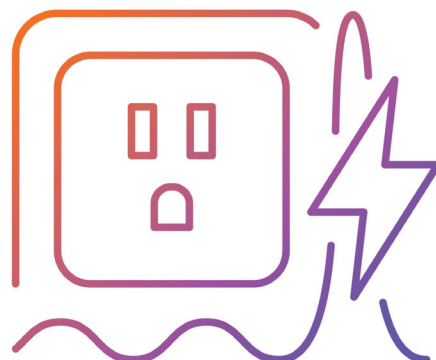
In order to talk about electrical circuits, you need to be familiar with the terms voltage, current, resistance and about Ohm's Law, which defines the relationship between all three.

- **Voltage** (measured in volts – V) is the force that pushes electrons through a circuit to produce electricity.
- **Current** (measured in amperes – A) is the act of the electrons flowing through the circuit (rate of flow).
- **Resistance** (measured in ohms – Ω) is a force that counteracts the flow of current.



✍ Note:

1. All the **electrical energy gets used up in a circuit** by the components in it. Each component converts some of the energy into another form of energy (light, heat, sound, etc.).
2. The **flow of current** at a specific point in a circuit will always be the same coming in and going out.
3. Electrical current will seek the path of least resistance to ground. Given two possible paths, more of the electrical current will go down the path with less resistance. If you have a connection that connects power and ground together with no resistance, you will cause a **short circuit**, and the current will try to follow that path. In a short circuit, the power source and wires convert the electrical energy into light and heat, usually as **sparks or an explosion**.



Software Setup

These software options will provide you a programming environment where you can write and experiment with your code.

1. Arduino IDE

In order to use the Arduino board, you need to program the microcontroller. You do this using the “Arduino IDE” (IDE = Integrated Development Environment) software, which allows you to write programs, which we call “sketches,” and upload them.

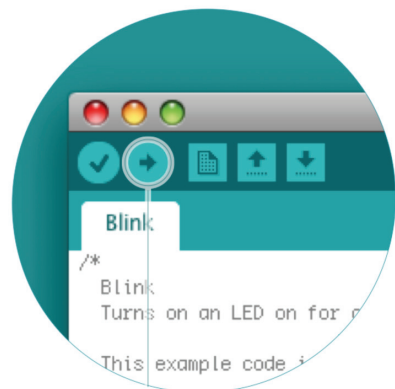
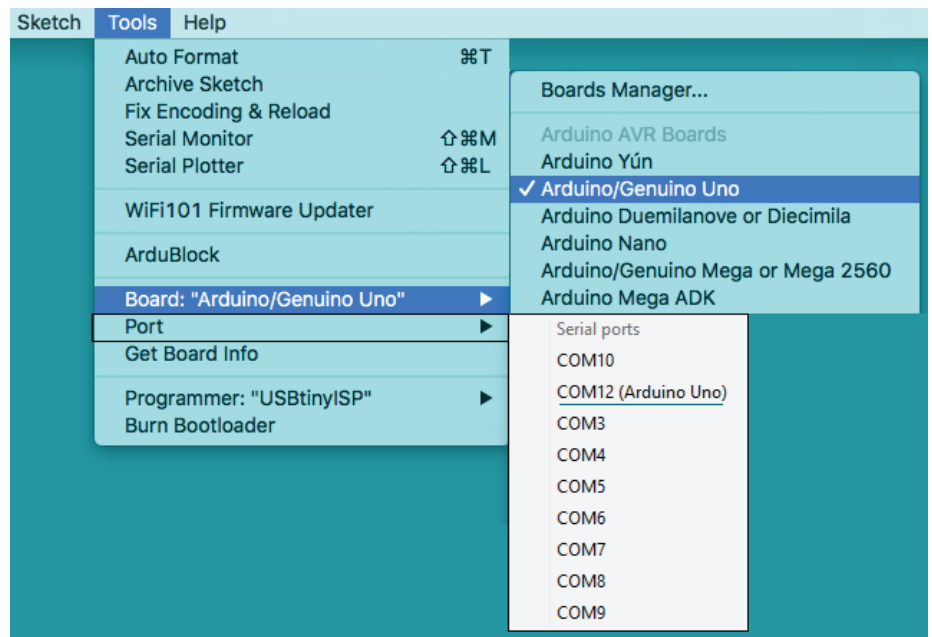
You can download the latest version of the IDE, and find the installation instructions, from: arduino.cc/download



After you have installed the Arduino IDE, it is time to set up your Arduino board.

You will tell Arduino IDE which board you are using. There are a lot of options! The one in your kit is an Arduino Uno.

Next choose the serial port your Arduino is connected to from the TOOLS > SERIAL PORT menu.



This is likely to be the COM port with the highest number. There is no harm in guessing wrong, and if it doesn't work, try the next one. To find out, you can disconnect your Arduino board and re-open the menu; the entry that disappears should be the Arduino board. Reconnect and select that serial port.

Finally, you will upload the Arduino sketch to the board!

🔗 Arduino Uno R3 Clone Driver Fix

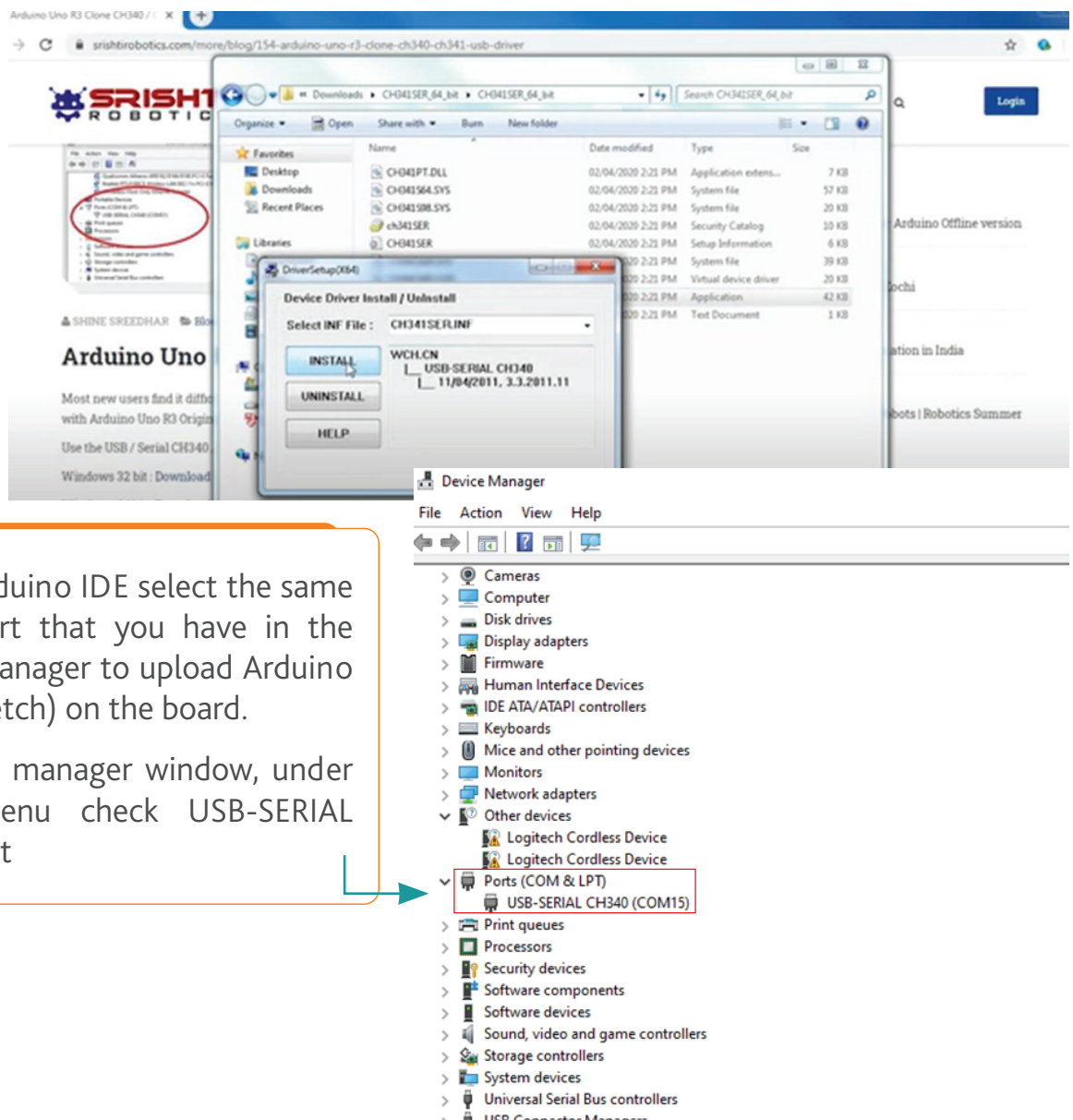
For the Arduino IDE software to recognize an Arduino clone microcontroller you need to download and install an Arduino Uno R3 clone USB Driver.

<https://www.srishtirobotics.com/more/blog/154-arduino-uno-r3-clone-ch340-ch341-usb-driver>

*Download the serial driver suitable to your device

Connect the Arduino board to your computer using the USB cable.

If installation is successful, the new USB driver will be visible in the Devices Manager of Windows.

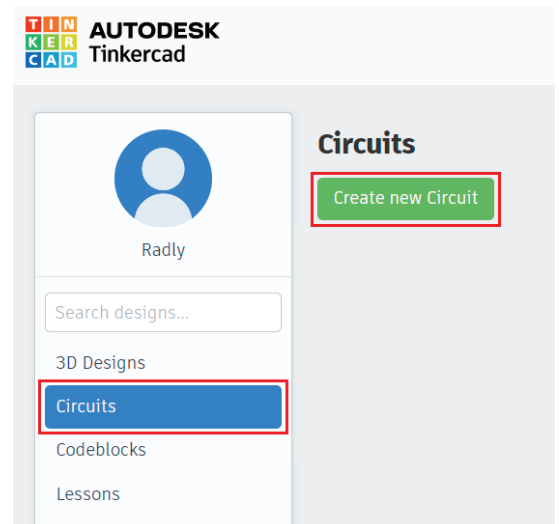


2. Tinker CAD (online open source)

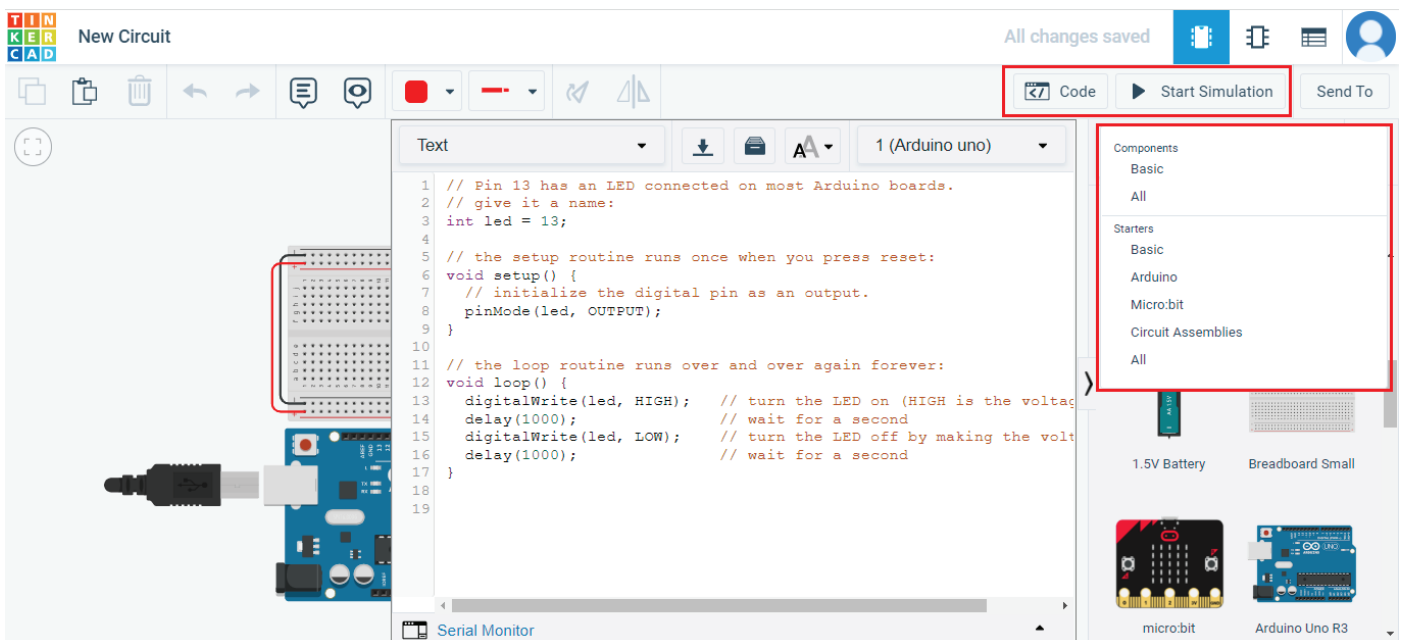
Tinkercad Circuits allows students to use an interactive circuit editor, where they can explore, connect, and code virtual projects with a bottomless toolbox of simulated components.



You need to create an account and sign into Tinkercad, where you will find a dashboard of your recent designs. To see a view of your Circuit designs, simply click the Circuits link in the left menu and you can create something new by clicking the *Create New Circuit button*.



Tinkercad allows you to visualize what a program, you write in the “Code Editor”, does using the simulated Arduino Board in the Workplane.



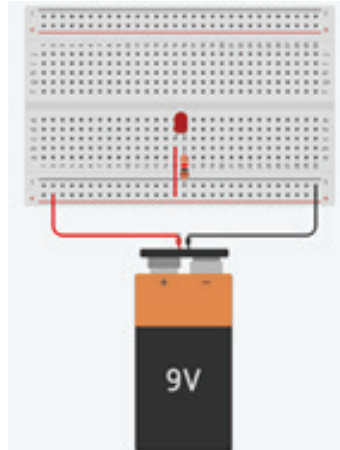
While you are writing an Arduino sketch, sometimes you make a mistake and the program will not be compiled (saved) and uploaded. This happens to professional programmers all the time! The error message can tell you when a problem occurs and how to fix it.

🔌 First circuit – Let there be light!

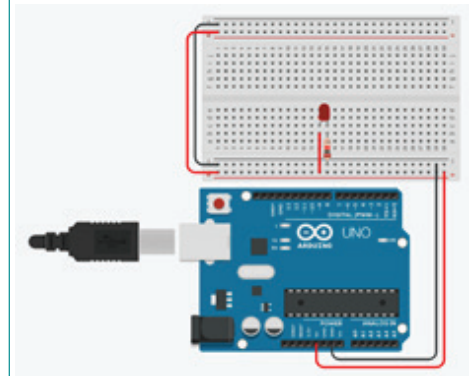
You can light up an LED in three ways:



The LED can be connected straight to a 9V power source.



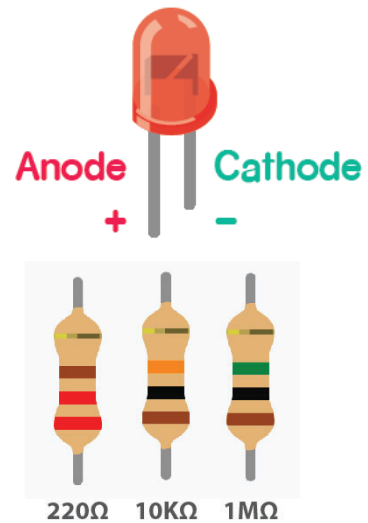
The LED can be connected to the breadboard (with its two legs in different rows of the board) which is powered by a 9V power source

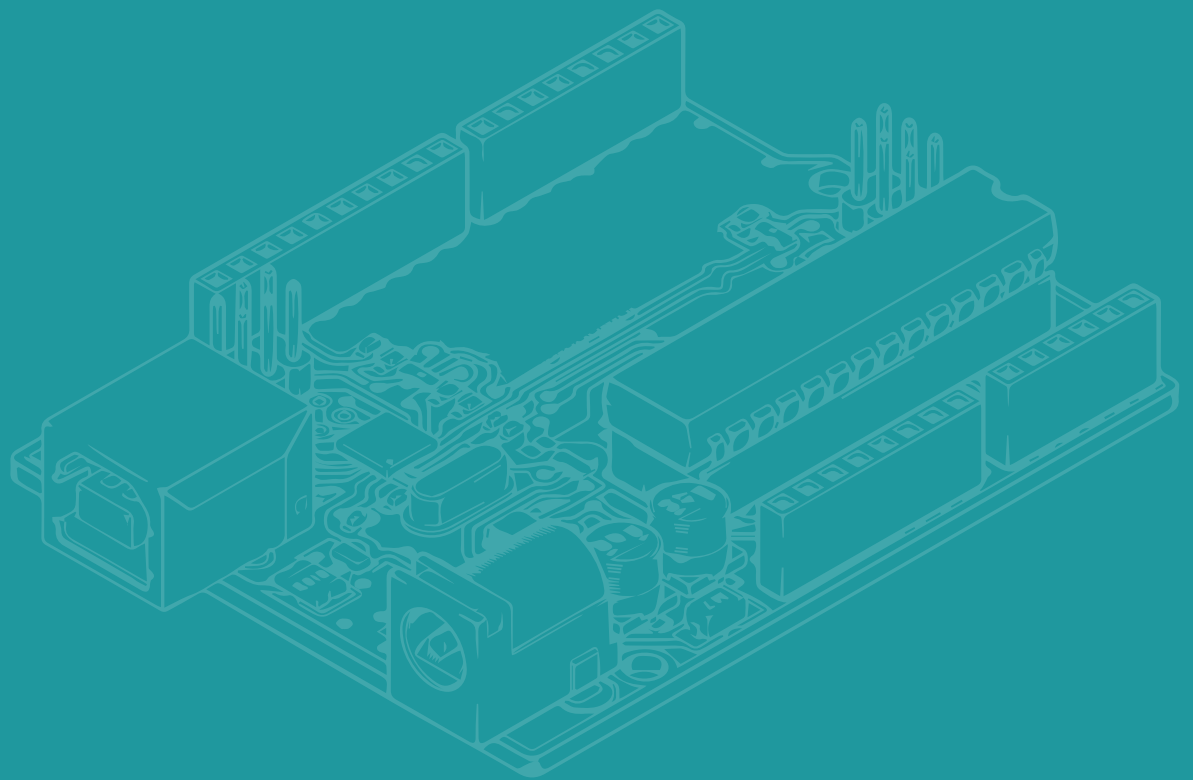


The LED can be connected to the breadboard (with its two legs in different rows of the board) which is powered by the Arduino board through the USB power source

Notes:

- The longer leg of the LED, called the **anode**, is supposed to connect to the positive (+ve) power source and the shorter leg, the **cathode**, should connect to the ground negative (-ve) source.
- In all the 3 ways, a 220Ω **resistor** (the colored stripes identify the resistor's value) was added to the circuit to regulate the voltage flow through the LED, because a high voltage would cause the LED to pop!
- The resistor can go before or after the LED, and still protect it. The current that flows out of a battery is always equal to the current that flows back into the battery.
- The breadboard is powered through the horizontal rows "*voltage rails*", at the top and bottom of the breadboard, by connecting the red row to the positive (+ve) side of the power source (5V pin on the Arduino board) and the black row to the negative (-ve) side of the power source (GND pin on the Arduino board).
- When building your circuits for the future projects be sure to replicate the reference images of the lessons precisely.
- Be sure you are wiring the components to their correct polarity, which side is connected to the (+ve) end and which is connected to the (-ve).
- It is helpful to keep your wire colours consistent (red for power, black for ground) throughout your circuits.





ISBN: 2023/9/4939

ARDUINO