

# Usage and applications of an Arduino

[ufrjnautilus.com/post/uso-e-aplicações-do-arduino](https://ufrjnautilus.com/post/uso-e-aplicações-do-arduino)

14 aprile 2021

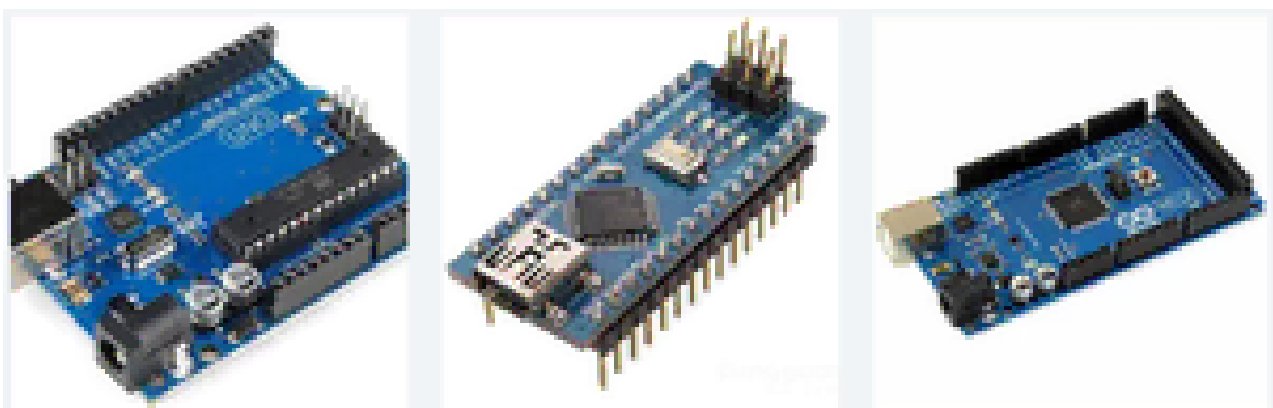
Not so long ago, in order to build and test any circuit it was necessary to design it from the ground up, so any changes in the project required lots of work and research.

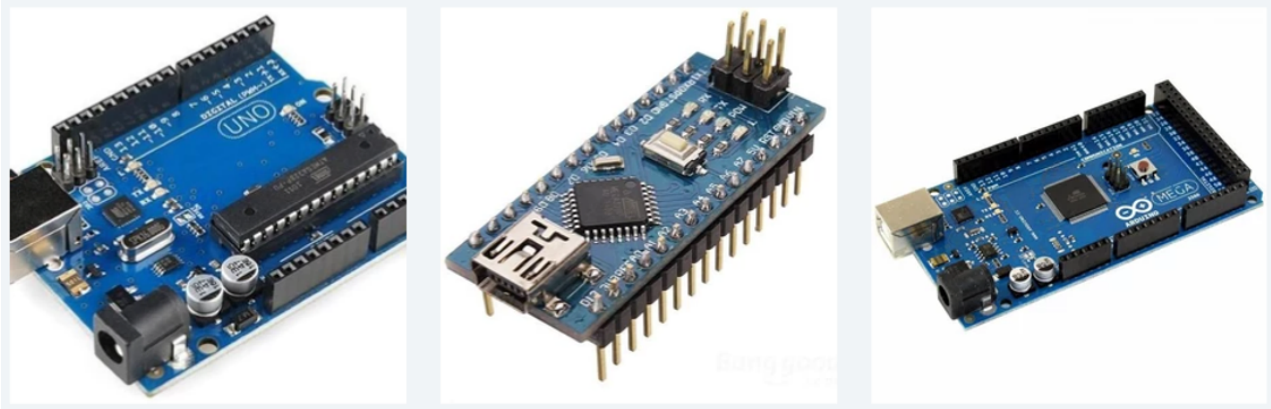
The popularization of microcontrollers made it possible to program a whole new range of circuits using software. The circuits could then perform different functions by simply programming them. However, manipulating microcontrollers was still difficult, it wasn't trivial to tinker with microcontrollers, and that is when the need arose to create something similar to what we now call the Arduino.

## What is Arduino?

Arduino is an open-source electronics platform based on free and easy-to-use hardware and software. It is practically a mini computer without the peripherals that fits in the palm of your hand and can perform various tasks, from turning an LED on and off to more complex things like a robotic hand that repeats the user's movements. It uses a microcontroller that can be programmed by a simple and very friendly IDE (Integrated Development Environment) that uses a language based on C/C++.

Arduino was born at the Ivrea Institute for Interaction Design as an easy tool for rapid prototyping, aimed at students without a background in electronics and programming, so that anyone could develop technology. Nowadays, it is very popular all over the world having several models with different particularities.

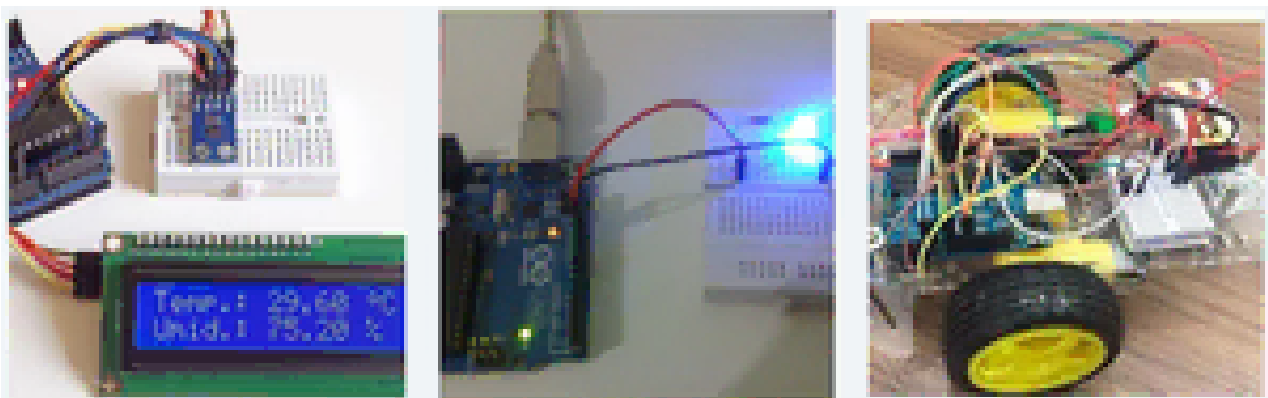




Arduino Uno, Nano and Mega  
**Why use Arduino?**

Inexpensive, functional and easy to program, with an IDE that runs on several operating systems. Arduino has free hardware and software, which allows the user (students, hobbyists, engineers and anyone who wants to learn) to customize and modify the Arduino from a base system.

With a few lines of code it is possible to turn a led on and off, obtain a measurement from a distance or temperature sensor, and show it on an LCD display, for example. Besides these simple applications, you can use it for several other purposes, such as automating your garage door, lighting your house, among other possibilities.





Display LCD, Led and Line Follower  
**You don't need to buy a board!**

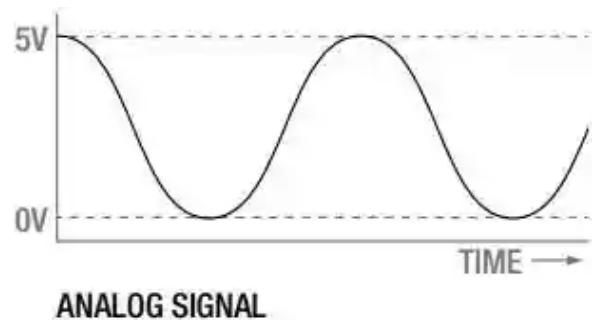
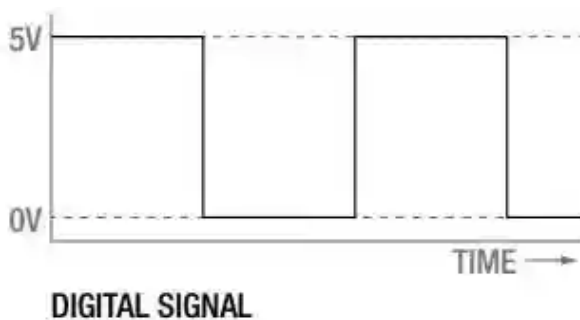
Yes, that's what you read, it is not necessary to spend a dime to learn how to work with the arduino. Nowadays there are some great sites/simulators where you can assemble the circuit (some with a lot of complexity) and compile the code to see how everything would look like in real life. Below are some of these sites/simulators:

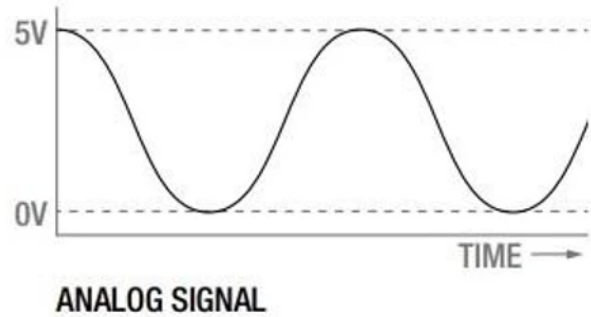
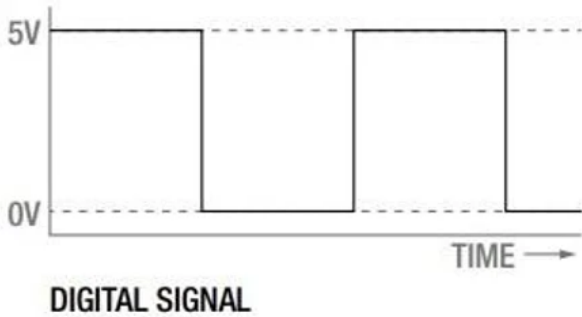
<https://www.tinkercad.com/>

After registering, just click on "Circuits" and have fun. In addition, we also have Virtual BreadBox, however, I can not give an opinion on it, because I never used it, but it is another option that can be very cool and useful.

### About the pins

In an arduino we have digital signal pins (PWM or not) and analog signal pins. But what is each type of pin?





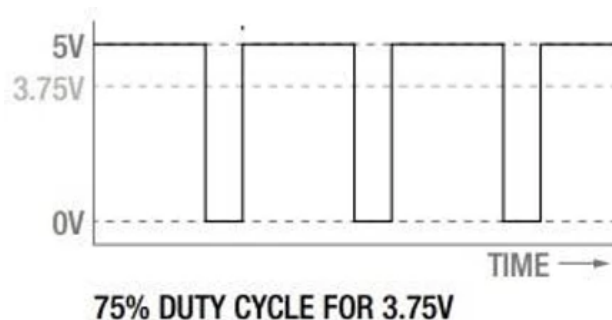
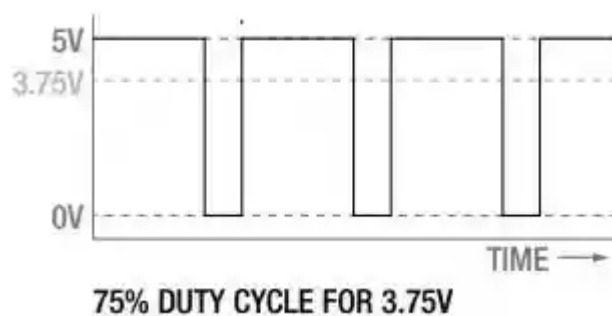
Digital signal and Analog signal.

### Digital Signal Pins

Initially let's talk about the digital signal pins, which are basically a "switch", that is, it works as on or off (in the case of the arduino, this means that it either delivers nothing, 0V, or it delivers its highest value, +5V) and can work as output and as input.

Some digital pins (3,5,6,9,10,11 in the case of the arduino uno) can be programmed to be used as analog outputs using PWM (Pulse Width Modulation). What is PWM anyway?

Let's imagine a simple on and off switch, when on it delivers all the voltage and power to the load, but when off it delivers nothing. Now imagine that you control the time that the switch stays on or off. If we make it stay on for 75% of the time and the rest of the time off, this is the same as saying that for 75% of the time we have current and voltage. So the average power applied to the load is the average voltage itself, i.e. 75%, so the longer the switch is on, the higher the average power delivered to the load.



Signal PWM 75%

In this way we are able to use a certain voltage that is contained between 0V and + 5V through a digital pin.

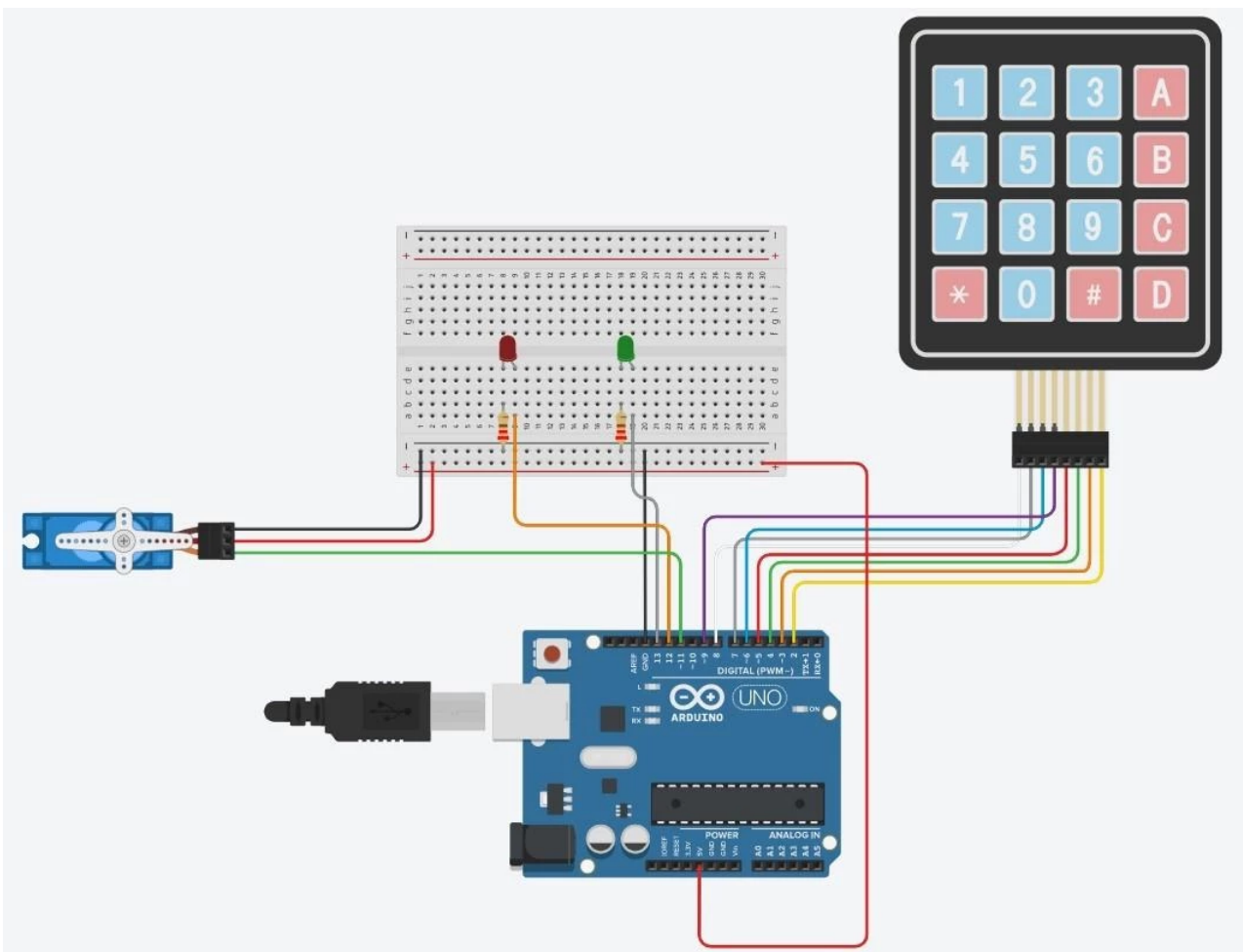
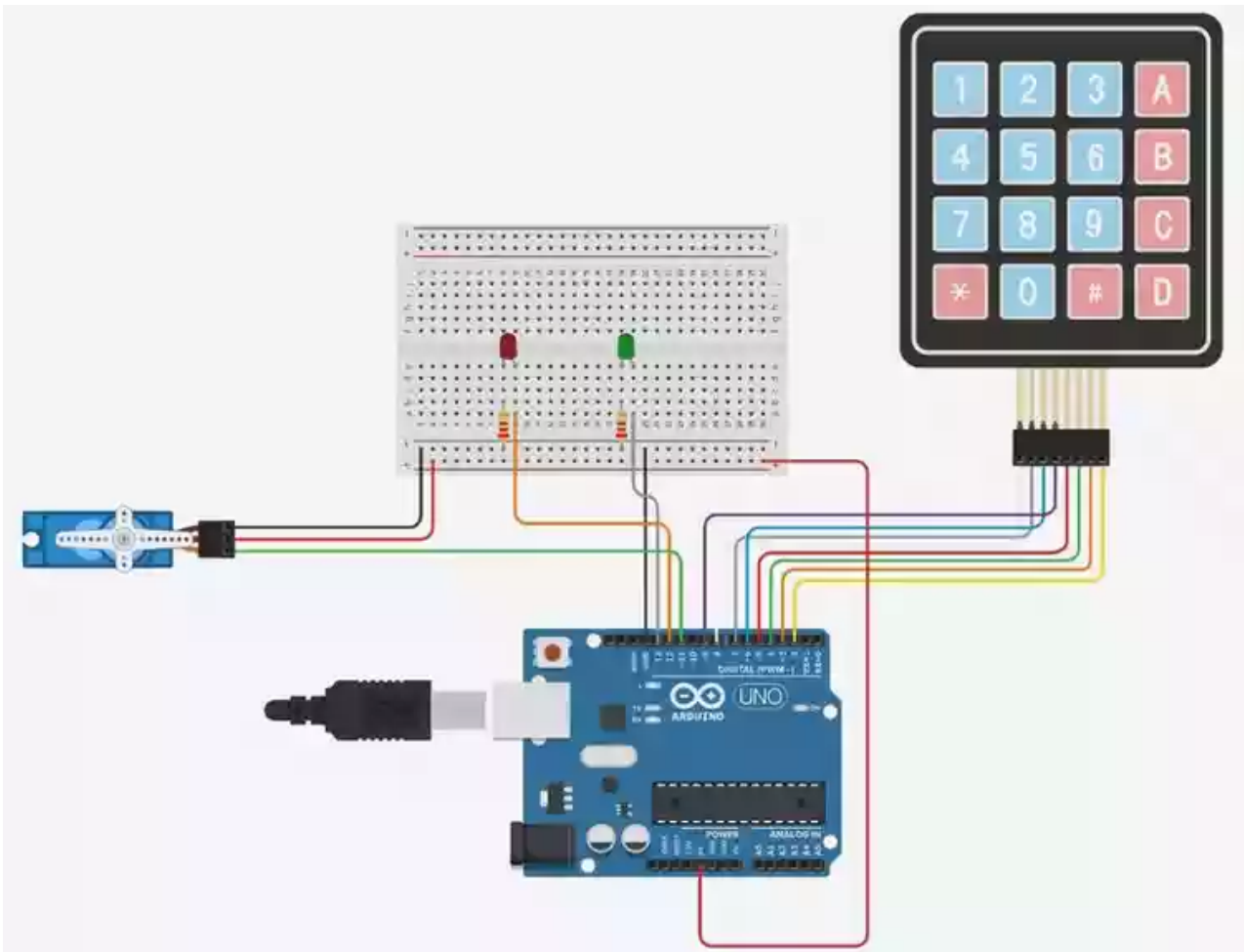
## **Analog Signal Pins**

As the figure above shows, the analog signal pins are the pins that deliver any value in the given range (here in this case from 0V to +5V), i.e., we can use a voltage of +5V as well as +3.8V. This type of pin is ideal for reading values about temperature, pressure, and humidity, since these are magnitudes that vary continuously.

## **Automating your gate lock**

This is an example of something very cool that can be done using one arduino uno, 2 LEDs, 2 220 ohm resistors, a membrane keyboard, and a servo motor (here representing the gate lock). The system works as follows:

As soon as the correct password is entered, the red led will turn off and the green led will light up at the same time the servo motor will rotate 90° (as if it were your gate lock).



Gate Circuit

In the link below you can access the schematic and the code:

[https://www.tinkercad.com/things/3VYSuWYKX5c-glorious-jaiks-robo/editel?sharecode=s-T1oX32cQSyIi6D9Ms19BXMpmKe1a1\\_DbONXIQXXnY](https://www.tinkercad.com/things/3VYSuWYKX5c-glorious-jaiks-robo/editel?sharecode=s-T1oX32cQSyIi6D9Ms19BXMpmKe1a1_DbONXIQXXnY)

**Whiten by Iago Oliveira.**