



INTRODUCTION TO ARDUINO

2021

Co-funded by the
Erasmus+ Programme
of the European Union



INTRODUCTION TO ARDUINO

Arduino is a small single-board computer based on microcontrollers. Arduino was first introduced in 2005. It is an open-source platform. Unlike the Raspberry Pi, the Arduino is not intended to be a full-fledged desktop computer. It is most often used to implement standalone interactive electronic circuits.

EXAMPLES OF MODELS

ARDUINO NANO

Arduino Nano is one of the smallest development boards. The development board includes an Atmel ATmega328 microcontroller as standard and 32KB flash memory. It has 14 input/output pins.

ARDUINO UNO

Arduino Uno is one of the most famous development boards. The development board includes an Atmel ATmega328P microcontroller as standard and 32KB flash memory. It has 14 input/output pins.

ARDUINO MEGA

Arduino Mega includes a more powerful Atmel ATmega2560 processor. It has 256 kB of memory and 54 input/output pins.

There are others – Arduino Due, Arduino Leonardo, Arduino Micro...

More information can be found at <https://store.arduino.cc/collections/boards>.

Quick overview

<i>Model</i>	<i>Microcontroller</i>	<i>Flash KiB</i>	<i>EEPROM KiB</i>	<i>SRAM KiB</i>	<i>Digital I/O pins</i>	<i>PWM channels</i>	<i>Analog inputs</i>
<i>Diecimila</i>	ATmega168	16	0.5	1	14	6	6
<i>Duemilanove</i>	ATmega168/328P	16/32	0.5/1	1/2	14	6	6
<i>Uno</i>	ATmega328P	32	1	2	14	6	6
<i>Due</i>	ATMEL SAM3U	256		50			16
<i>Mega</i>	ATmega1280	128	4	8	54	14	16
<i>Mega2560</i>	ATmega2560	256	4	8	54	14	16
<i>Leonardo</i>	ATmega32u4	32	1	2	14	6	12
<i>Fio</i>	ATmega328P	32	1	2	14	6	8

<i>Nano</i>	ATmega168 or ATmega328	16/32	0.5/1	1/2	14	6	8
<i>LilyPad</i>	ATmega168V or ATmega328V	16	0.5	1	14	6	6
<i>Yún</i>	Atheros AR9331 – 400MHz (ATmega32u4)	16 MiB (32KiB)	1	64 MiB DDR2 (1KiB)	20 (ATmega32u4)	7	12

STARTING WITH ARDUINO

For the first steps with Arduino, we need:

- ✓ software – Arduino IDE,
- ✓ Arduino development board,
- ✓ USB cable,
- ✓ computer or laptop,
- ✓ electronic components or sensors and wires.

DEVELOPMENT ENVIRONMENT

The program can be downloaded from websites for various types of operating systems.

<https://www.arduino.cc/en/software>

Downloads



Arduino IDE 1.8.16

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the [Getting Started](#) page for Installation instructions.

SOURCE CODE

Active development of the Arduino software is [hosted by GitHub](#). See the instructions for [building the code](#). Latest release source code archives are available [here](#). The archives are PGP-signed so they can be verified using [this](#) gpg key.

DOWNLOAD OPTIONS

Windows Win 7 and newer
Windows ZIP file

Windows app Win 8.1 or 10 

Linux 32 bits
Linux 64 bits
Linux ARM 32 bits
Linux ARM 64 bits

Mac OS X 10.10 or newer

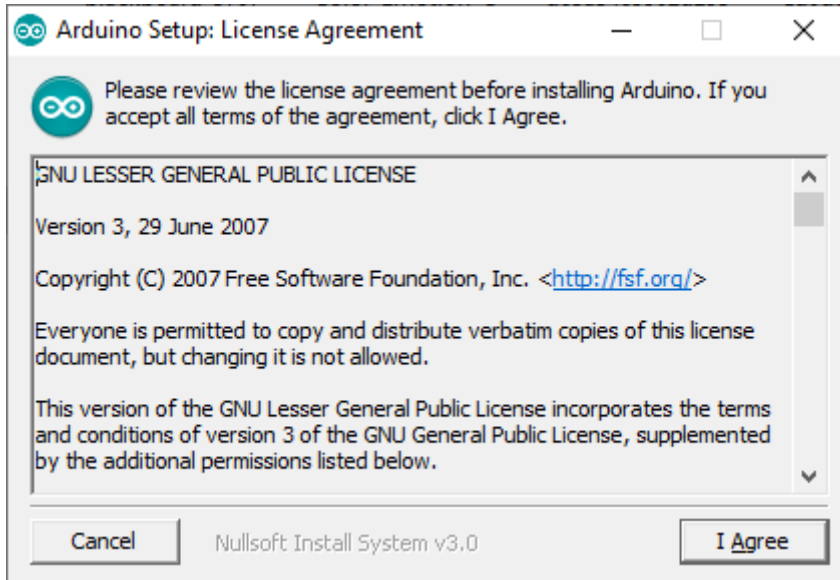
[Release Notes](#) [Checksums \(sha512\)](#)

Source of the screenshot: <https://www.arduino.cc/en/software>

INSTALLATION PROCEDURE

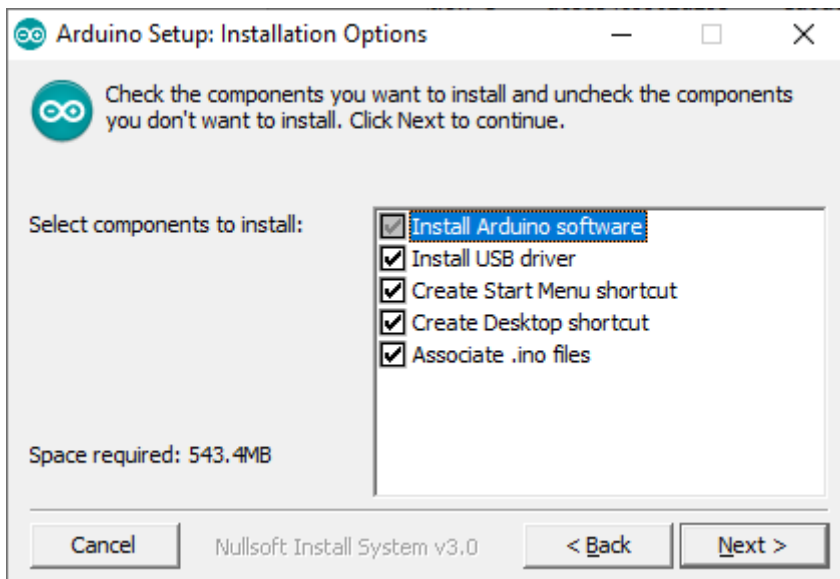
There are two ways to install Arduino IDE for Windows. Either the installation file runs or unpacks the ZIP archive.

We are going to use the first option – run the downloaded setup file.

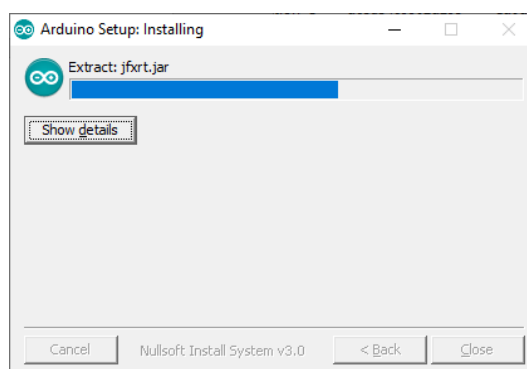
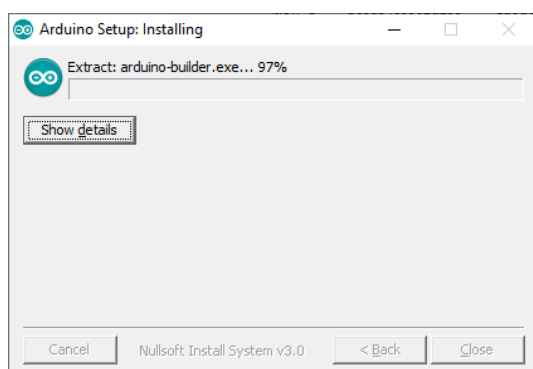
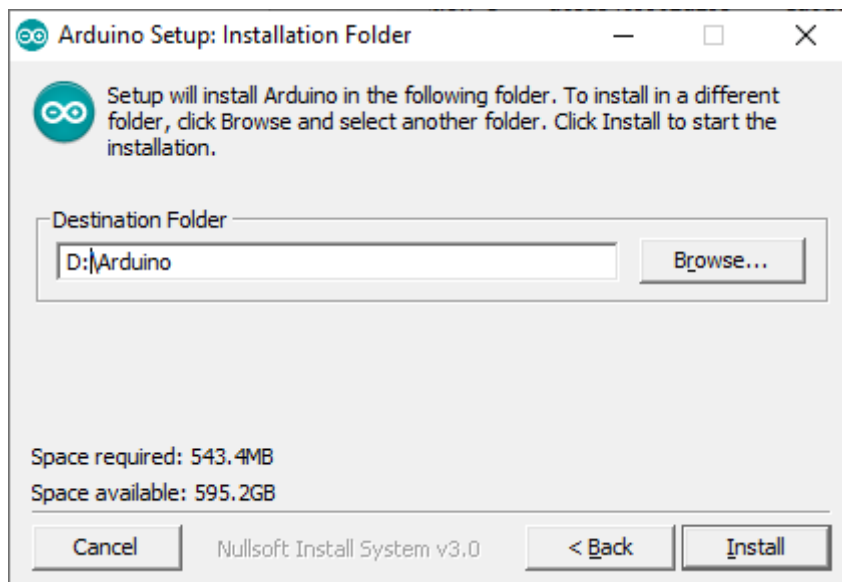


It is required to accept all terms of the License Agreement. Press the I agree button.

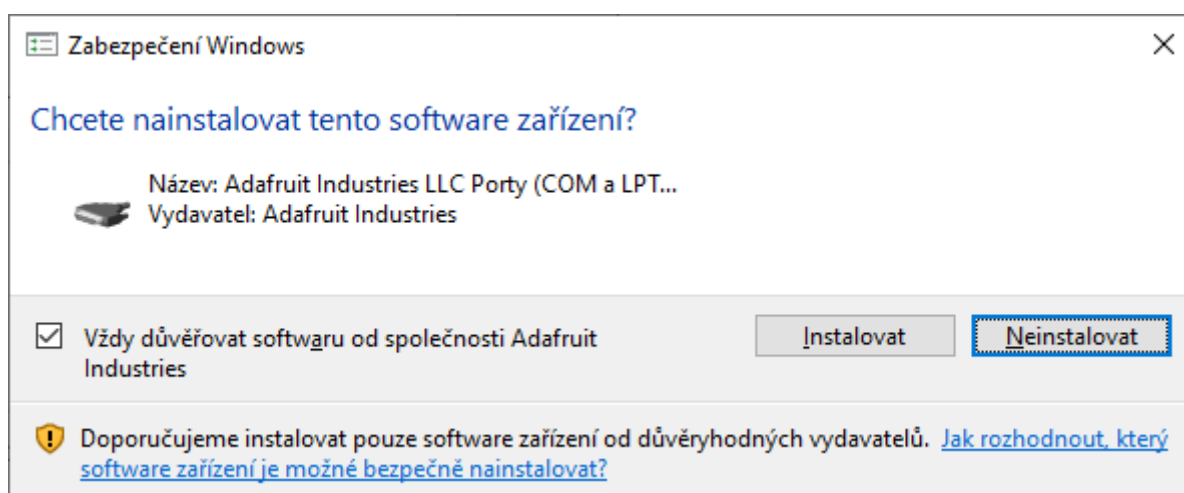
In the next window, leave all options checked and press the Next button.

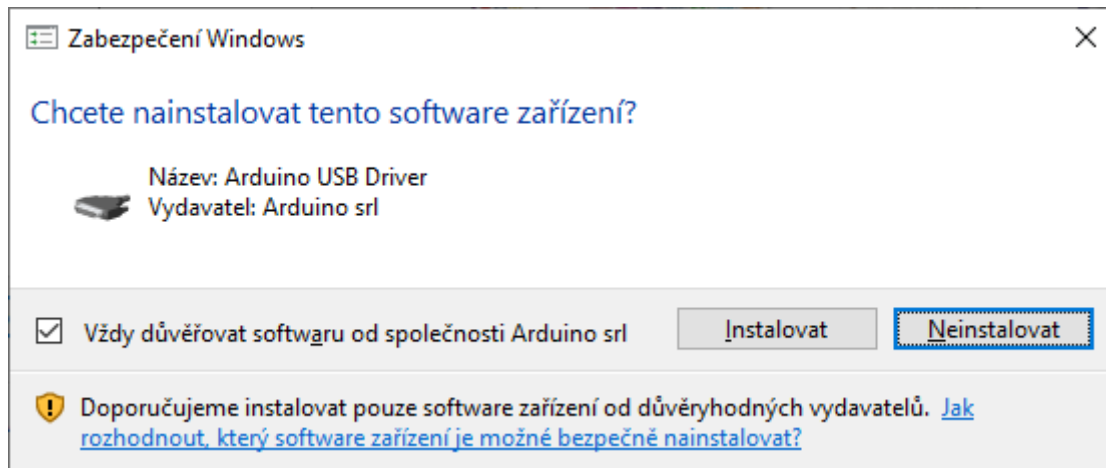


Select the folder where the program will be installed.

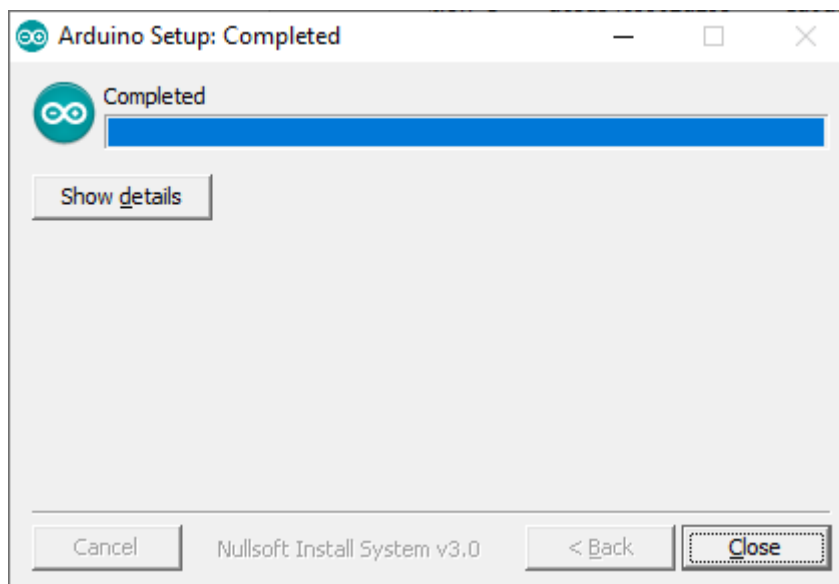


If drivers will be installed – we will enable the installation.

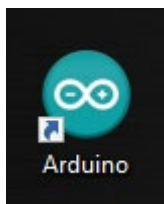




After installation, the message Completed appears.



The Arduino icon appears on the Windows desktop.



Note: Installation on another operating system is eg. on <https://www.arduino.cc/en/guide/linux> or <https://ubuntu.com/tutorials/install-the-arduino-ide#1-overview>

CONNECTING THE DEVELOPMENT BOARD

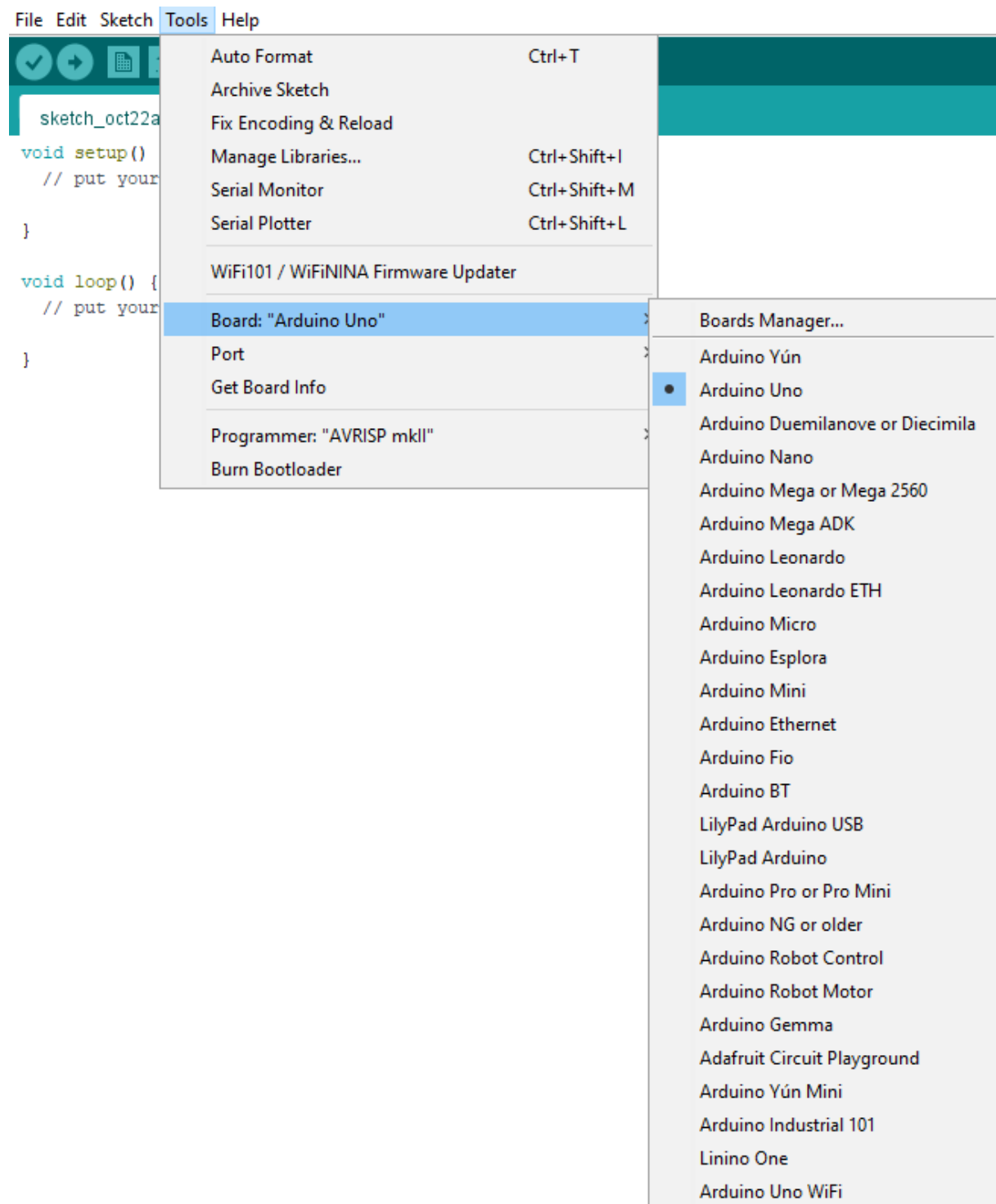
Connect the Arduino board to your computer using a USB cable. Development boards are usually powered from USB or from an external supply. When connected, the green LED should light up.

INSTALLING THE ARDUINO MEGA 2560 DRIVER

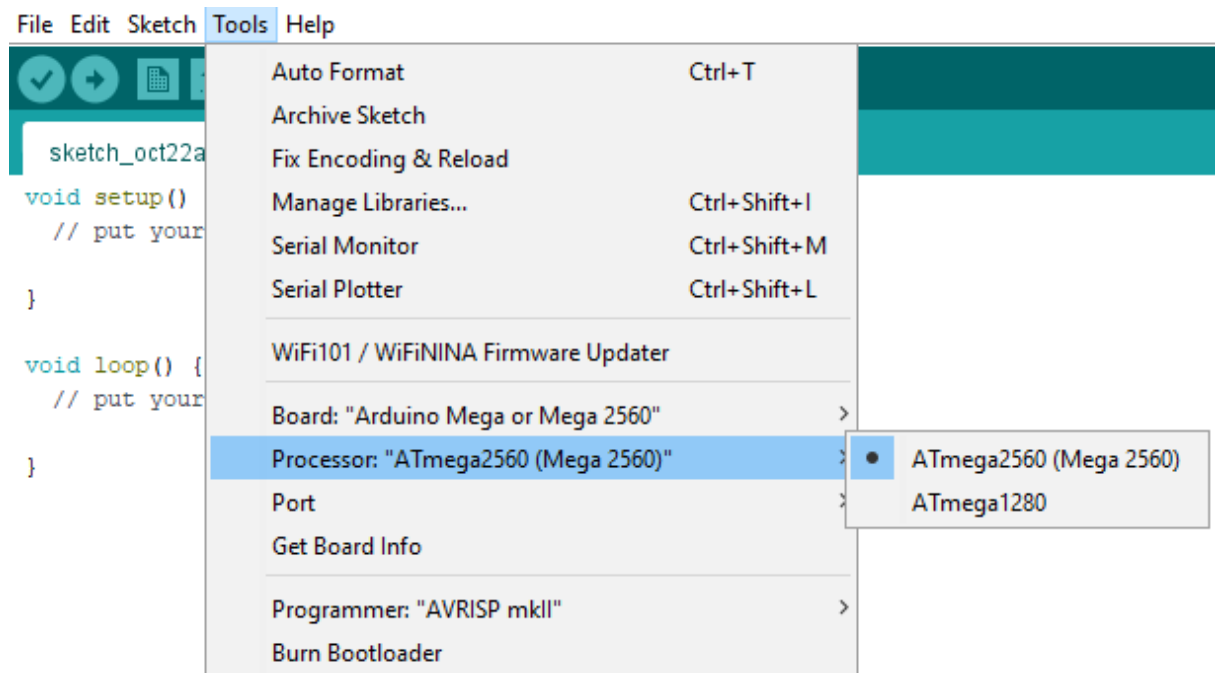
In Windows 10, driver installation should run automatically. Alternatively, you can check in the Device Manager menu – COM Ports entry.

LAUNCH THE ARDUINO IDE

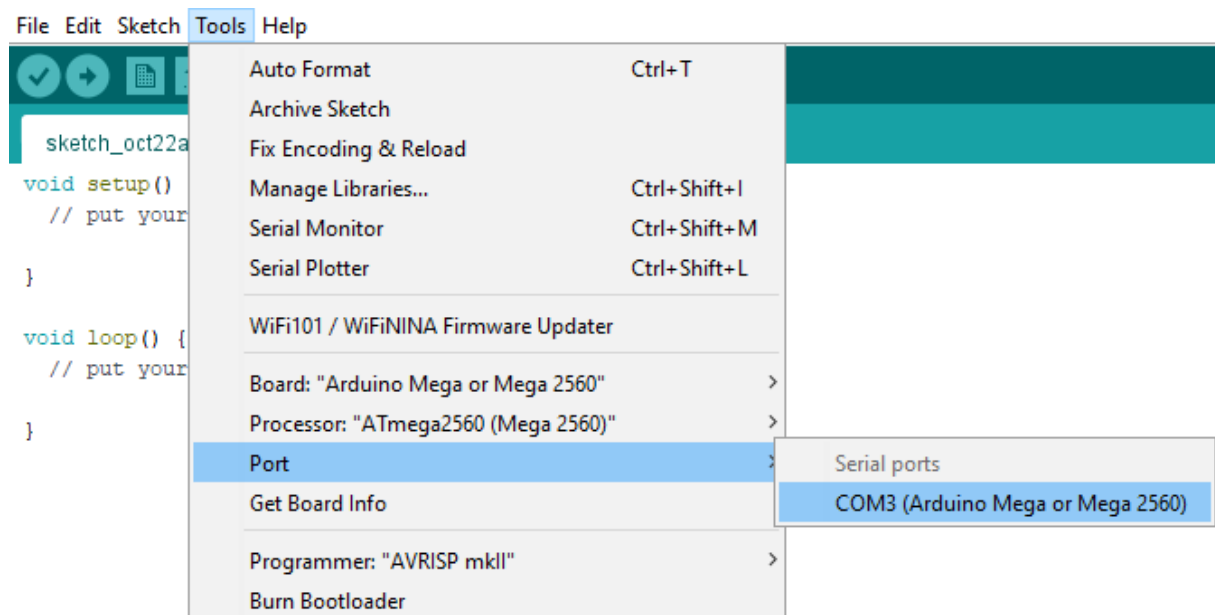
In the Tools > Board menu, select the board type.



In some cases, we also have to select a processor.



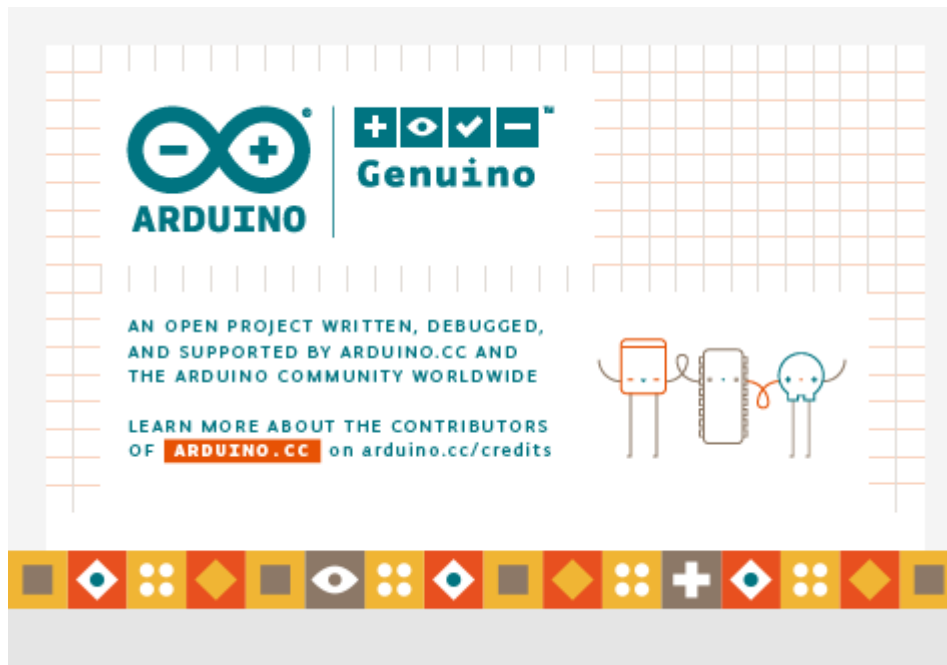
The next step is to select a port in Tools > Port.



TESTING THE CONNECTION

We will create a short program for testing. The goal will be to blink the BUILTIN LED on the development board.

We will run the program.

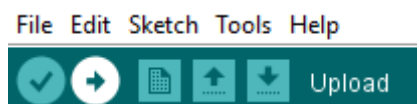


File > Save — filename: led

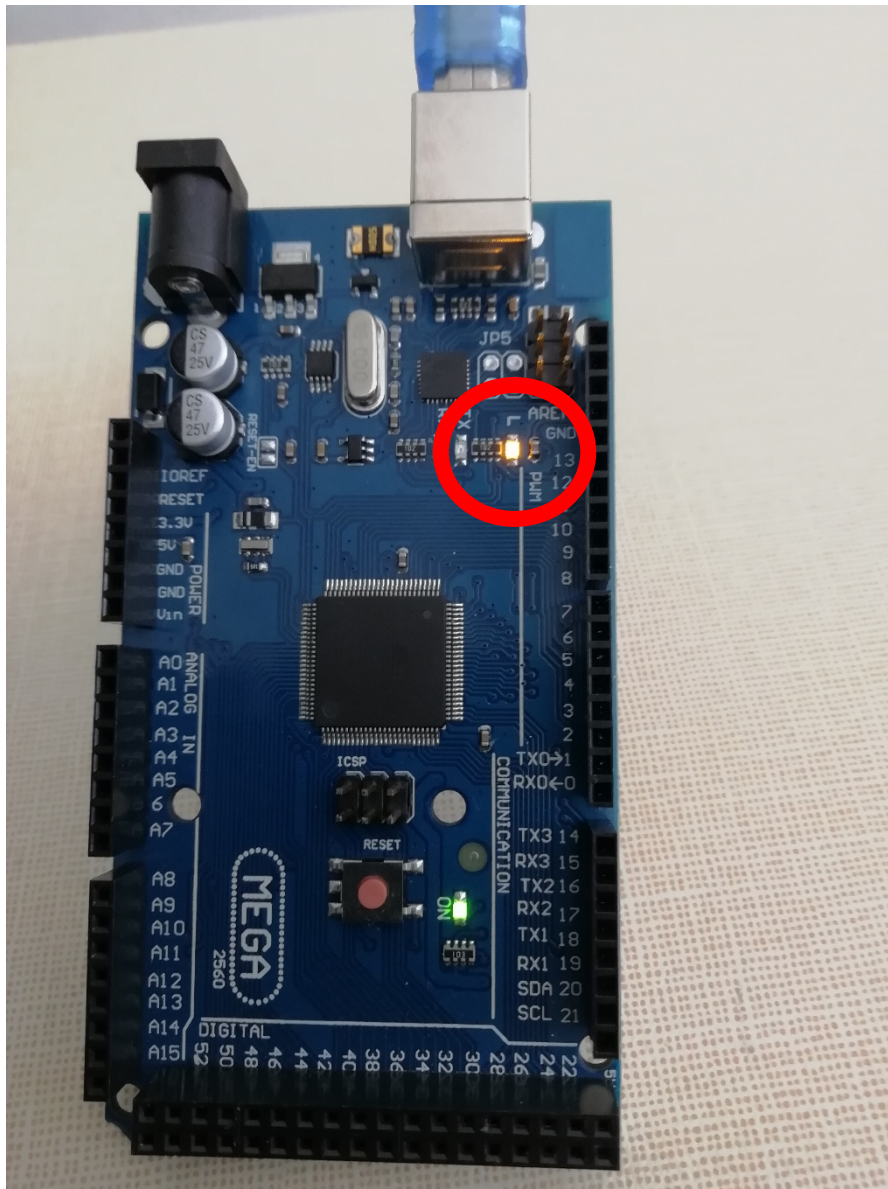
```
led | Arduino 1.8.15
File Edit Sketch Tools Help
void setup() {
  pinMode(LED_BUILTIN, OUTPUT);
}

void loop() {
  digitalWrite(LED_BUILTIN, HIGH);
  delay(250);
  digitalWrite(LED_BUILTIN, LOW);
  delay(250);
}
```

We will upload the program to the development board.



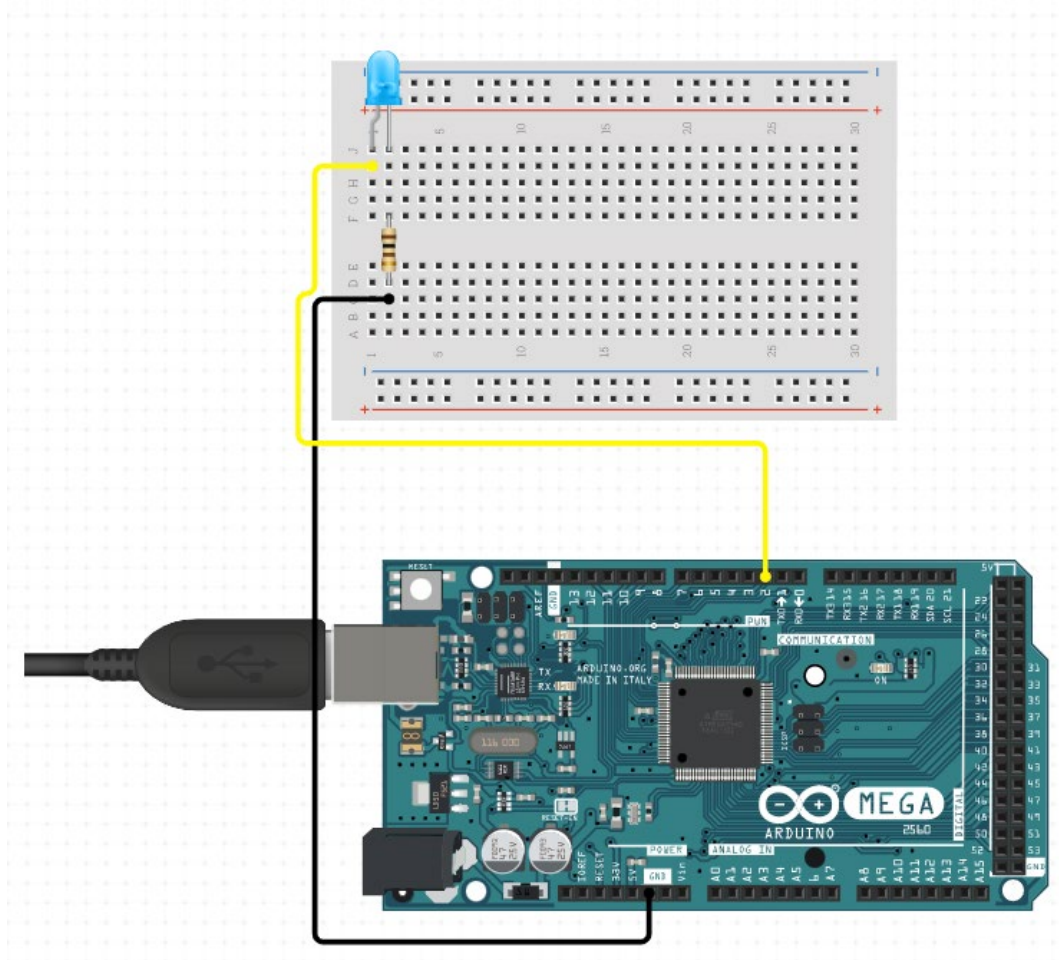
After a successful upload, the program will automatically start. The LED starts blinking.



THE SECOND PROGRAM – EXTERNAL LED BLINK

Required components: Arduino development board, LED, ballast resistor 100 Ω , jumper wires.

We will make the wiring according to the diagram below.



Initialize the digital PIN 2 as the output.

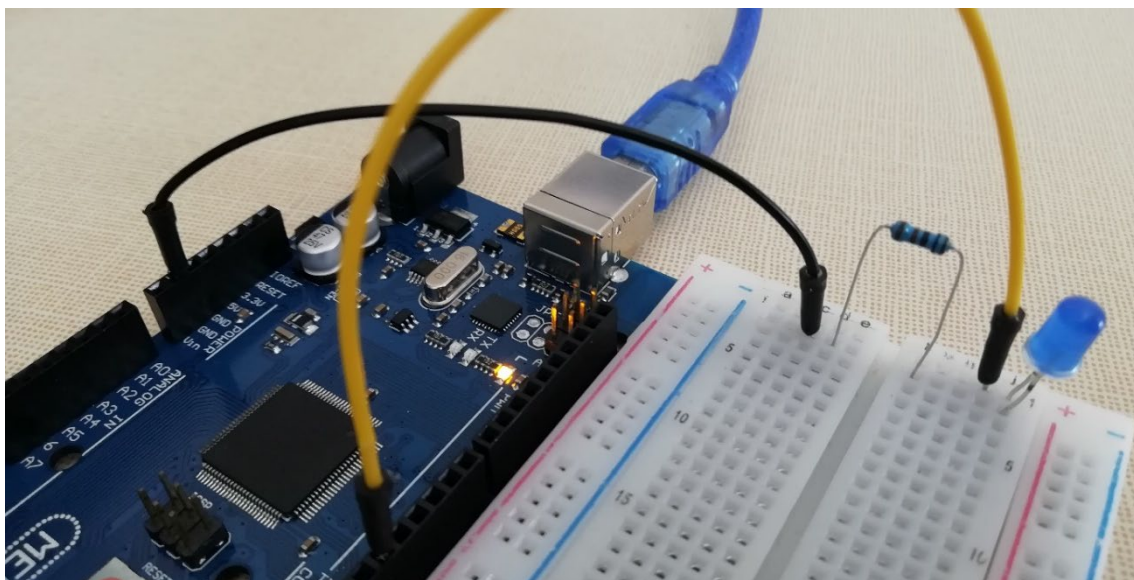
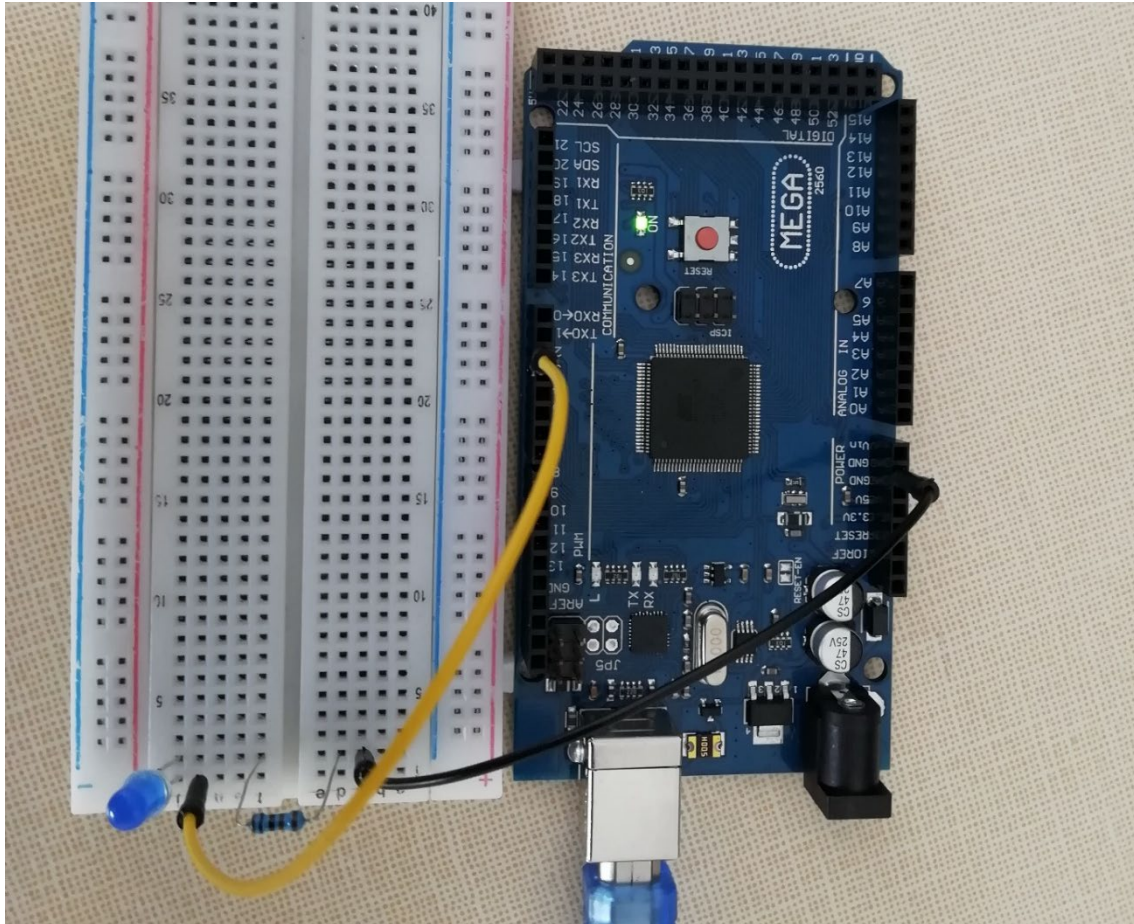
```
void setup(){
    pinMode(2, OUTPUT);
}
```

We will want the LED to blink. That means it will be on for 1 second and then off for 1 second.

```
void loop() {
    digitalWrite(2, HIGH);
    delay(1000);
    digitalWrite(2, LOW);
    delay(1000);
}
```

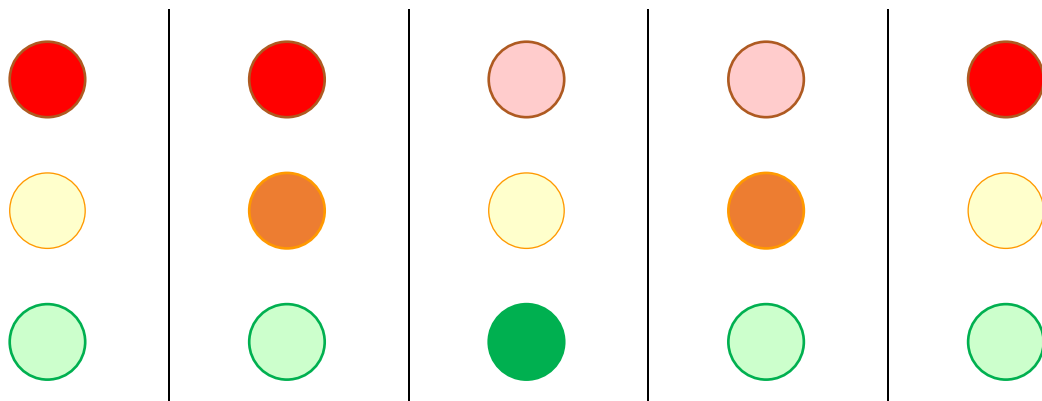

This code writes a logical 1 (HIGH) to the PIN 2. It leaves it there for 1 second (1000 milliseconds). Then it writes a logical 0 (LOW) to the same pin, which is left there again for 1 second. It is repeated in the infinite loop.

The digitalWrite function has two parameters. The first is the pin to be controlled and the second is the logical value.

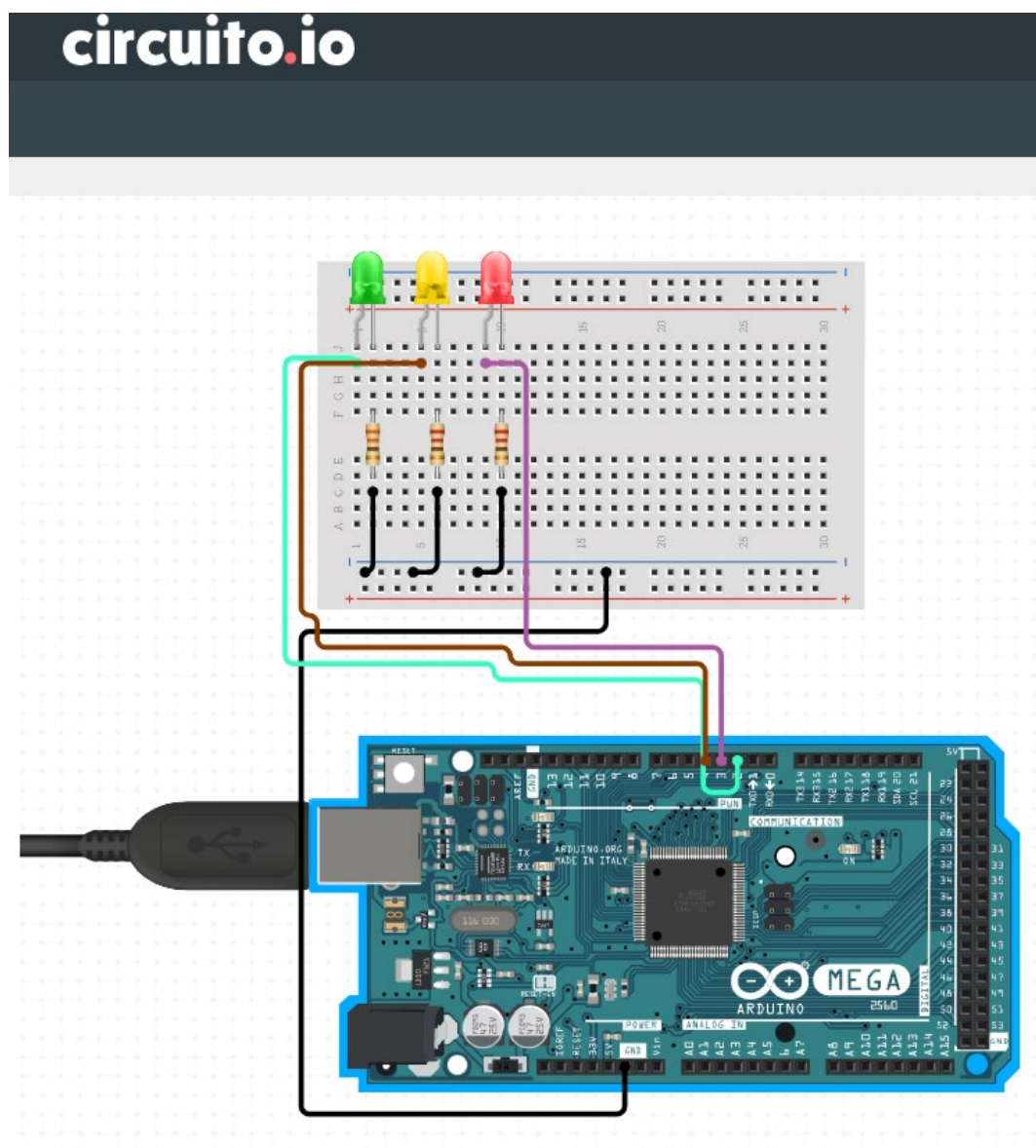


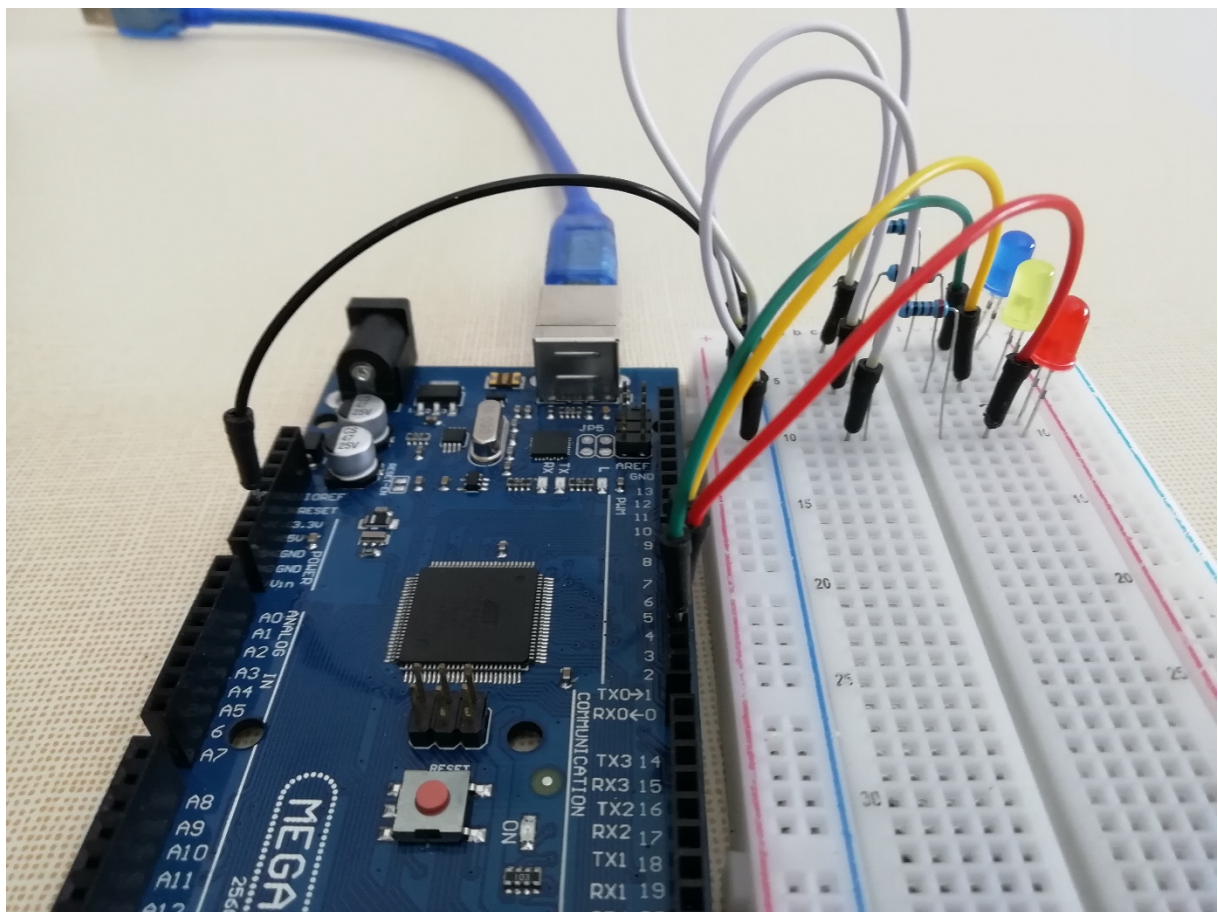
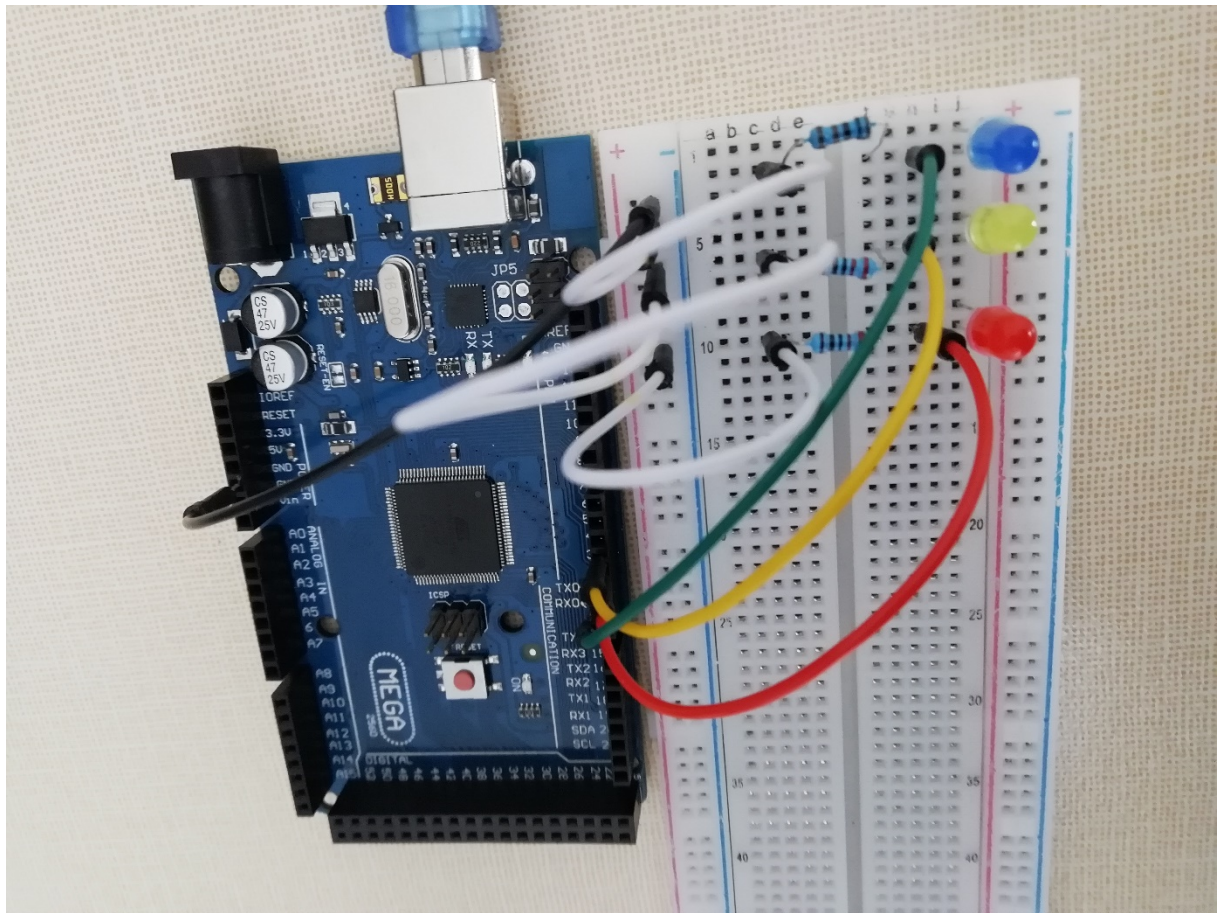
TRAFFIC LIGHTS

We will try to simulate traffic lights. The states of the lights are shown on the image below.



Wiring schema:





Code:

```
const int LED_RED = 3;
const int LED_YELLOW = 4;
const int LED_GREEN = 2;

void setup()
{
  pinMode(LED_RED, OUTPUT);
  pinMode(LED_YELLOW, OUTPUT);
  pinMode(LED_GREEN, OUTPUT);
}

void loop()
{
  digitalWrite(LED_RED, HIGH);
  digitalWrite(LED_YELLOW, LOW);
  digitalWrite(LED_GREEN, LOW);
  delay(2000);

  digitalWrite(LED_YELLOW, HIGH);
  delay(1000);

  digitalWrite(LED_RED, LOW);
  digitalWrite(LED_YELLOW, LOW);
  digitalWrite(LED_GREEN, HIGH);
  delay(4000);

  digitalWrite(LED_YELLOW, HIGH);
  digitalWrite(LED_GREEN, LOW);
  delay(1000);
}
```

READING ANALOG VALUE OF PIN

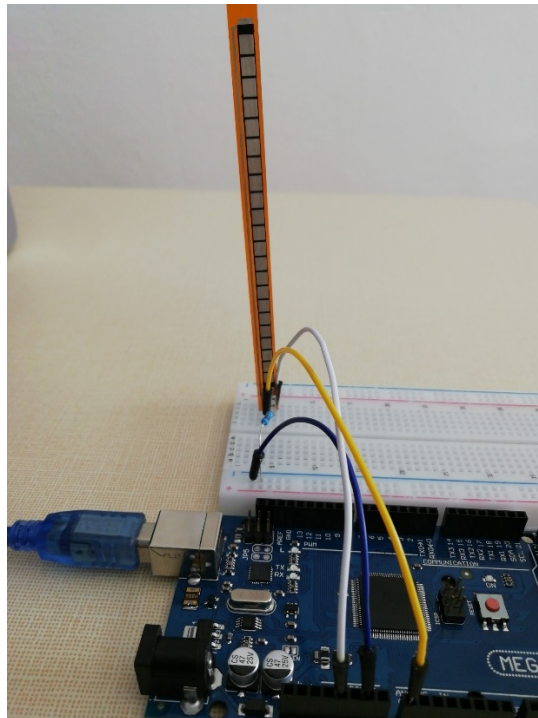
```
const int PIN_FLEX_SENSOR = 10;
int flexSensorState = 0;

void setup() {
    // Begin serial communication so we can print it out
    Serial.begin(9600);
}

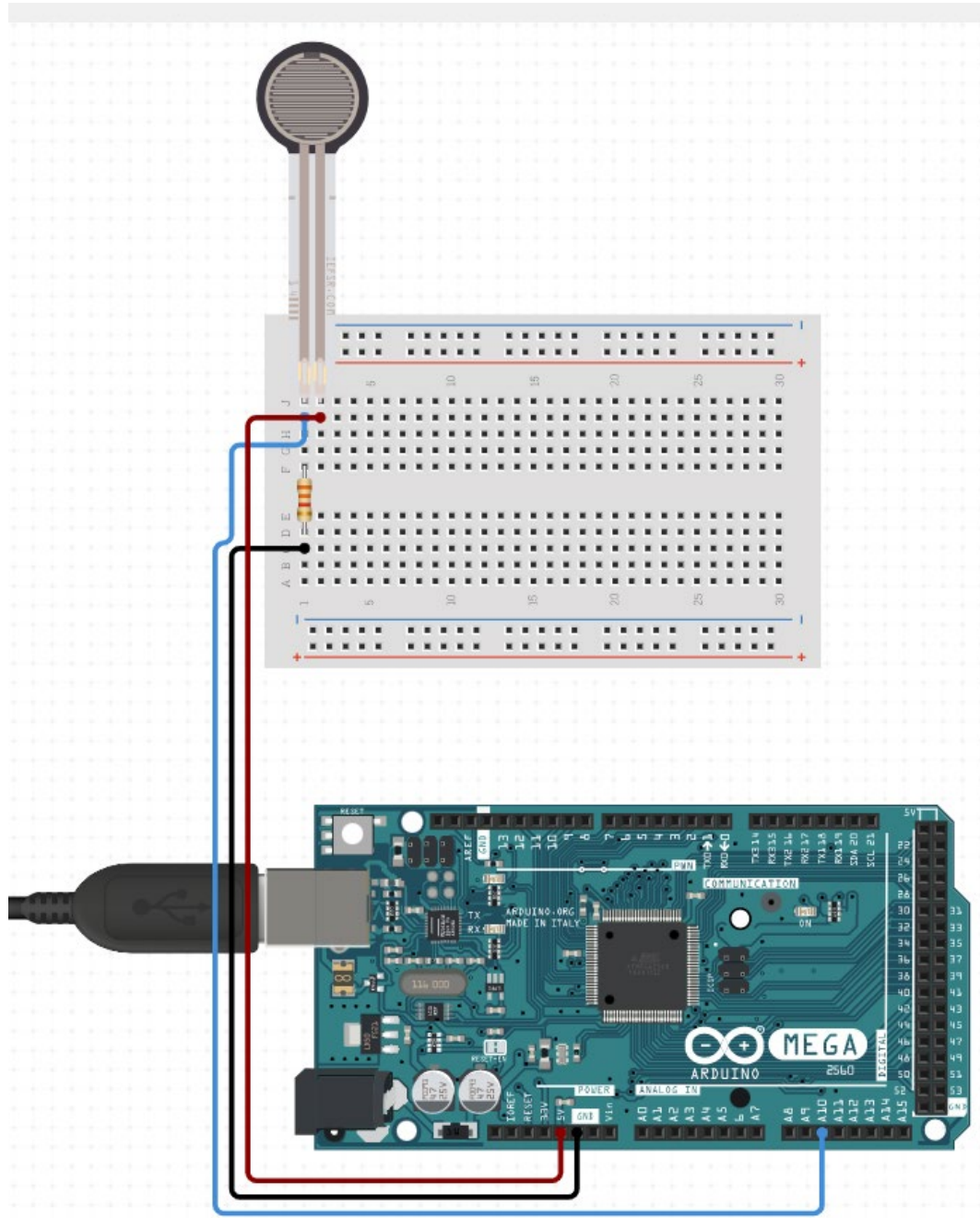
void loop()
{
    // Read analog pin of flex sensor to the flexSensorState variable
    flexSensorState = analogRead(PIN_FLEX_SENSOR);

    // Print value of flexSensorState variable to the serial monitor
    Serial.println(flexSensorState);

    // Wait 50 milliseconds
    delay(50);
}
```

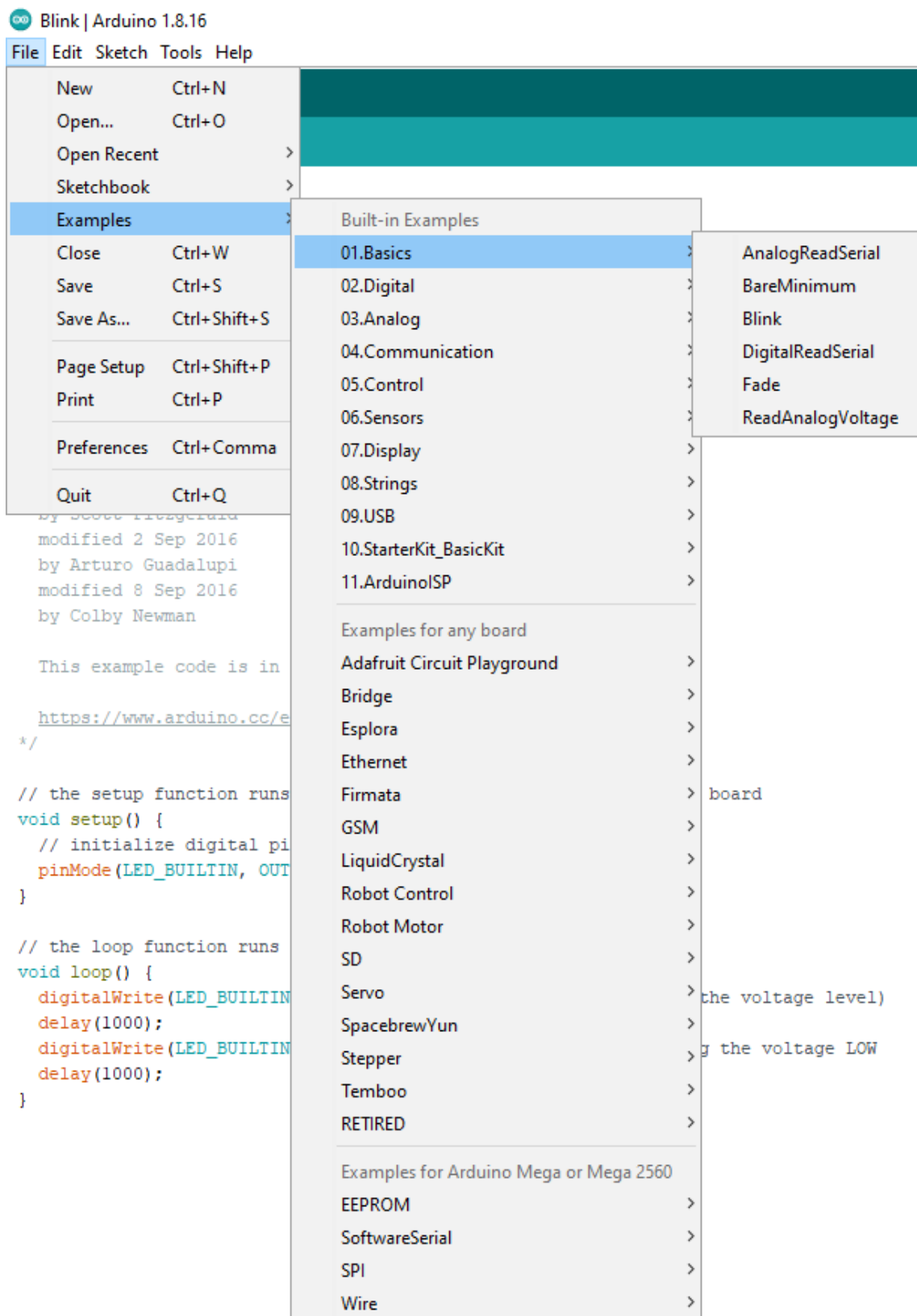


In this example is demonstrated reading values from analog pins. It uses the `analogRead` method that reads value from the specified analog pin. Arduino boards contain a multichannel, 10-bit analog to digital converter. This means that it will map input voltages between 0 and the operating voltage (for example 5 V) into integer values between 0 and 1023.



MORE EXAMPLES

Arduino IDE has built-in examples for the basic situations.



ONLINE SIMULATION

There are many online simulators that can be used to learn programming of Arduino without the physical board.

One of the best simulators is Tinkercad Circuits from Autodesk company.

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



Getting Started

