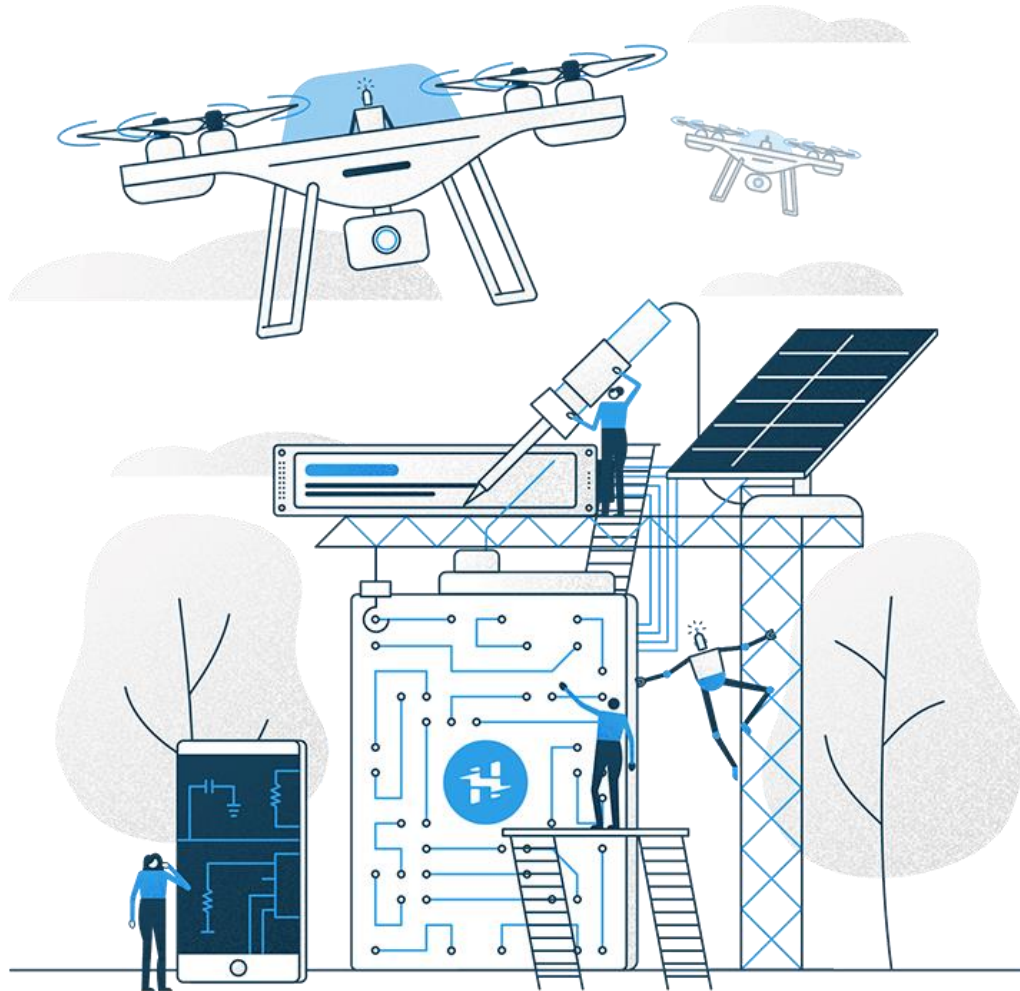


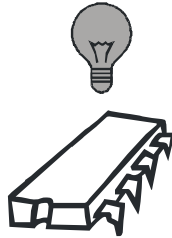
МІКРОПРОЦЕСОРНІ СИСТЕМИ УПРАВЛІННЯ



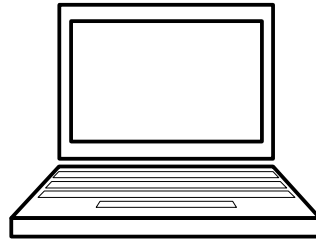
Lesson 3

Етапи розробки програмного забезпечення

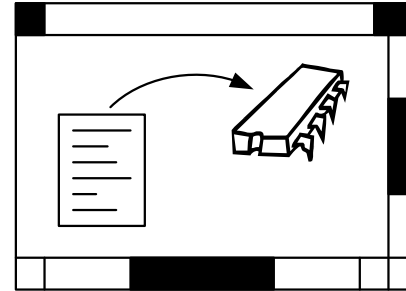
1. Чистий AVR нічого не робить



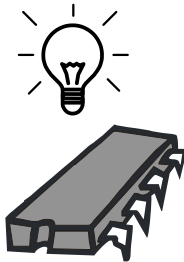
2. Пишемо програму на комп'ютері



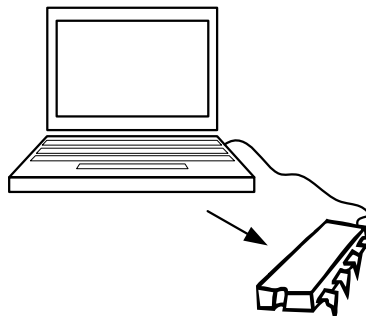
3. Програмування віртуального AVR в комп'ютері



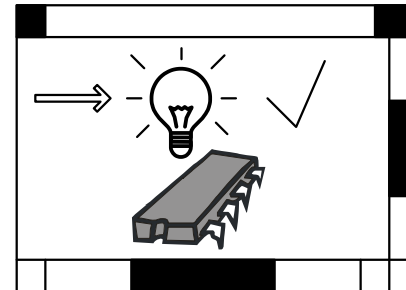
6. Перевіряємо програму в реальному пристрої



5. Програмування реального AVR



4. Перевіряємо програму на комп'ютері



Процес написання програм для МК AVR як і для будь-яких інших, складається з декількох етапів: підготовка вихідного тексту програми на якій-небудь мові програмування; компіляція програми; налагодження й тестування програми; остаточне програмування й підготовка до серійного виробництва.

Система команд і програмна модель AVR


The screenshot shows the Arduino.cc website's software page. The navigation bar includes 'PROFESSIONAL', 'EDUCATION', 'STORE', 'HARDWARE', 'SOFTWARE' (highlighted), 'DOCUMENTATION', 'COMMUNITY', 'BLOG', and 'ABOUT'. A search bar and 'SIGN IN' button are also present. The main content area features a 'Getting Started' card for the 'Arduino Web Editor' with a 'CODE ONLINE' button circled in red. Below this is a 'Downloads' section for 'Arduino IDE 1.8.13', which includes a description, installation instructions, and a list of 'DOWNLOAD OPTIONS' for Windows, Linux, and Mac OS X. A 'Help' button is located in the bottom right corner of the content area.

Arduino Web Editor

Start coding online and save your sketches in the cloud. The most up-to-date version of the IDE includes all libraries and also supports new Arduino boards.

[CODE ONLINE](#) [GETTING STARTED](#)

Downloads

 **Arduino IDE 1.8.13**

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)

[? Help](#)

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

Середовище розробки Arduino

WHAT IS ARDUINO CREATE?

ARDUINO NEWS

- Arduino Explore IoT Kit vs. Oplà IoT Kit: a detailed comparison
- This Arduino-powered robot is like a Roomba with a paintbrush
- This DIY domino clock tells the time using three LED-lit tiles
- This automated table makes beer

Web Editor

IoT Cloud

Manager for Linux

Digital Store

INTEL®-BASED PLATFORMS RUNNING LINUX ARE SUPPORTED!

LEARN MORE

powered by **amazon** web services

CREATE

<https://create.arduino.cc/>

Web Editor

The screenshot displays the Arduino IDE Web Editor interface. On the left is a teal sidebar with navigation options: EDITOR, Sketchbook, Examples (highlighted with a red dashed box), Libraries, Monitor, Help, Preferences, and Features usage. The main workspace is divided into three sections. The top section contains a 'NEW SKETCH' button, a file icon, and an upload icon. The middle section features a search bar for the sketchbook, an ordering dropdown set to 'LAST MODIFIED', and a list of sketches including 'sketch_may4a'. The bottom section shows an upload icon and a message: 'Import your sketches to your online Sketchbook and access them from any device!'. On the right, a code editor window titled 'sketch_may4a' is open, showing the Arduino sketch code for 'sketch_may4a.ino'. The code includes a header for 'Arduino Uno' and the standard 'void setup()' and 'void loop()' functions. A red dashed box highlights the code editor area.

EDITOR

Sketchbook

Examples

Libraries

Monitor

Help

Preferences

Features usage

NEW SKETCH

SEARCH SKETCHBOOK

ORDERING BY LAST MODIFIED

sketch_may4a

Import your sketches to your online Sketchbook and access them from any device!

sketch_may4a

Arduino Uno

SHARE

```
sketch_may4a.ino
1 /*
2
3 */
4
5 void setup() {
6     .....
7 }
8
9 void loop() {
10    .....
11 }
12
```

Web Editor

The screenshot displays the Arduino IDE Web Editor interface. On the left is a teal sidebar with navigation options: EDITOR, Sketchbook, Examples (selected), Libraries, Monitor, Help, Preferences, and Features usage. The main area is divided into three panes. The top-left pane shows a search bar for examples and a list of categories: 01.BASICS (6), 02.DIGITAL (9), 03.ANALOG (6), 04.COMMUNICATION (12), 05.CONTROL (6), 06.SENSORS (4), and 07.DISPLAY (2). Under the BASICS category, the 'Blink' example is highlighted with a red dashed box. The top-right pane shows the selected 'Blink' example with a dropdown menu. The main editor pane displays the C++ code for the Blink sketch, which includes a setup function to initialize the LED pin and a loop function that toggles the LED on and off with 1000ms delays. The bottom pane is a terminal window showing the output of the verification process, which concludes with 'Success: Done verifying Blink'.

```
26 void setup() {
27   // initialize digital pin LED_BUI
28   pinMode(LED_BUILTIN, OUTPUT);
29 }
30
31 // the loop function runs over and
32 void loop() {
33   digitalWrite(LED_BUILTIN, HIGH);
34   delay(1000);
35   digitalWrite(LED_BUILTIN, LOW);
36   delay(1000);
37 }
```

Success: Done verifying Blink

```
runtime.tools.bossac.path=/home/builder/.arduino15/packages/industruino/tools/bossac/1.7.0-
industruino -prefs runtime.tools.nrf5x-cl-
tools.path=/home/builder/.arduino15/packages/arduino/tools/nrf5x-cl-tools/9.3.1 -prefs
runtime.tools.xtensa-lx106-elf-
gcc.path=/home/builder/.arduino15/packages/esp8266/tools/xtensa-lx106-elf-gcc/2.5.0-3-
```

Arduino IDE

Test_Matrix_2 | Arduino 1.8.5

Файл Правка Скетч Инструменты Помощь

- Новый Ctrl+N
- Открыть... Ctrl+O
- Открыть недавние
- Папка со скетчами
- Примеры
- Закрыть Ctrl+W
- Сохранить Ctrl+S
- Сохранить как... Ctrl+Shift+S
- Настройки страницы Ctrl+Shift+P
- Печать Ctrl+P
- Настройки Ctrl+Comma
- Выход Ctrl+Q

- Control.h binary.h
- h"
- Встроенные Примеры
- 01.Basics
- 02.Digital
- 03.Analog
- 04.Communication
- 05.Control
- 06.Sensors
- 07.Display
- 08.Strings
- 09.USB
- 10.StarterKit_BasicKit
- 11.ArduinoISP

Примеры для любой платы

- Adafruit Circuit Playground
- Bridge
- Esplora
- Ethernet
- Firmata
- GSM
- LiquidCrystal
- Robot Control
- Robot Motor
- SD
- Servo
- SpacebrewYun
- Stepper
- Temboo
- ВЫБЫТ

Примеры для Arduino Nano

- EEPROM
- SoftwareSerial

- AnalogReadSerial
- BareMinimum
- Blink
- DigitalReadSerial
- Fade
- ReadAnalogVoltage

Blink | Arduino 1.8.5

Файл Правка Скетч Инструменты Помощь

- ✓
-
- 📄
- ↑
- ↓

Blink

```
23 */
24
25 // the setup function runs once when you press reset or power the board
26 void setup() {
27   // initialize digital pin LED_BUILTIN as an output.
28   pinMode(LED_BUILTIN, OUTPUT);
29 }
30
31 // the loop function runs over and over again forever
32 void loop() {
33   digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage
34   delay(1000); // wait for a second
35   digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the volta
36   delay(1000); // wait for a second
37 }
```

Особливості мови Arduino

Оператори

[setup \(\)](#)

[loop \(\)](#)

Керуючі оператори

[if](#)

[if ... else](#)

[for](#)

[switch case](#)

[while](#)

[do ... while](#)

[break](#)

[continue](#)

[return](#)

[goto](#)

Логічні оператори

[&&](#) (І)

[||](#) (АБО)

[!](#) (НЕ)

Синтаксис

[;](#) (крапка з комою)

[{ }](#) (Фігурні дужки)

[//](#) (одностроковий коментар)

[/ * */](#) (Многостроковий коментар)

[#define](#)

[#include](#)

Арифметичні оператори

[=](#) (Оператор присвоювання)

[+](#) (Додавання)

[-](#) (Віднімання)

[*](#) (Множення)

[/](#) (Ділення)

[%](#) (Залишок від ділення)

Оператори порівняння

[==](#) (дорівнює)

[!=](#) (Не дорівнює)

[<](#) (Менше ніж)

[>](#) (Більше ніж)

[<=](#) (Менше або дорівнює)

[>=](#) (Більше або дорівнює)

Бітові оператори

[&](#) (Побітове І)

[|](#) (побітове АБО)

[^](#) (Побітове XOR або виключає АБО)

[~](#) (Побітове НЕ)

[<<](#) (побітовий зрушення вліво)

[>>](#) (побітовий зрушення вправо)

Особливості мови Arduino

Типи даних

[void](#)

<https://doc.arduino.ua/ru/prog/>

[boolean](#)

<https://alexgyver.ru/lessons/arduino-reference/>

[char](#)

<https://arduinoplus.ru/coding-arduino/>

[unsigned char](#)

[byte](#)

[int](#)

(2-байтове) значення у діапазоні від -32768 до 32767

[unsigned int](#)

(2-байтове) значення у діапазоні від 0 до 65535

[word](#)

[long](#)

32 біта (4 байта) у діапазоні від -2 147 483 648 до 2 147 483 647

[unsigned long](#)

32 біта (4 байта) у діапазоні від 0 до 4 294 967 295

[short](#)

(2-байтове) значення у діапазоні від -32768 до 32767

[float](#)

32 біта (4 байта) у діапазоні від $-3.4028235E+38$ до $3.4028235E + 38$

[double](#)

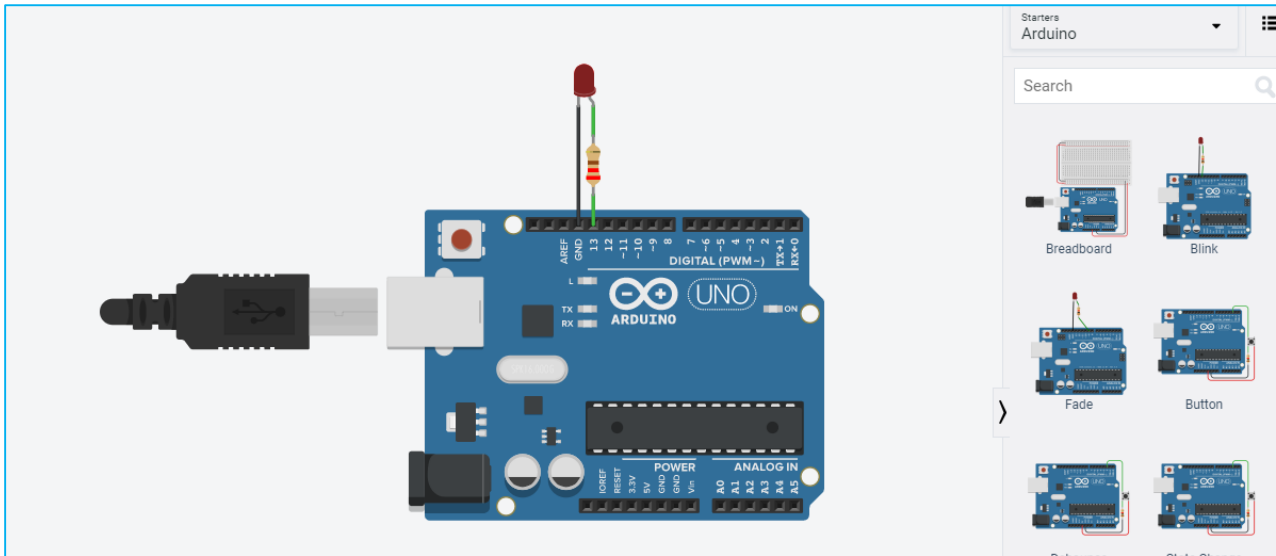
32 біта (4 байта) у діапазоні від $-3.4028235E+38$ до $3.4028235E + 38$

[string](#) - масив символів

[String](#) - об'єкт

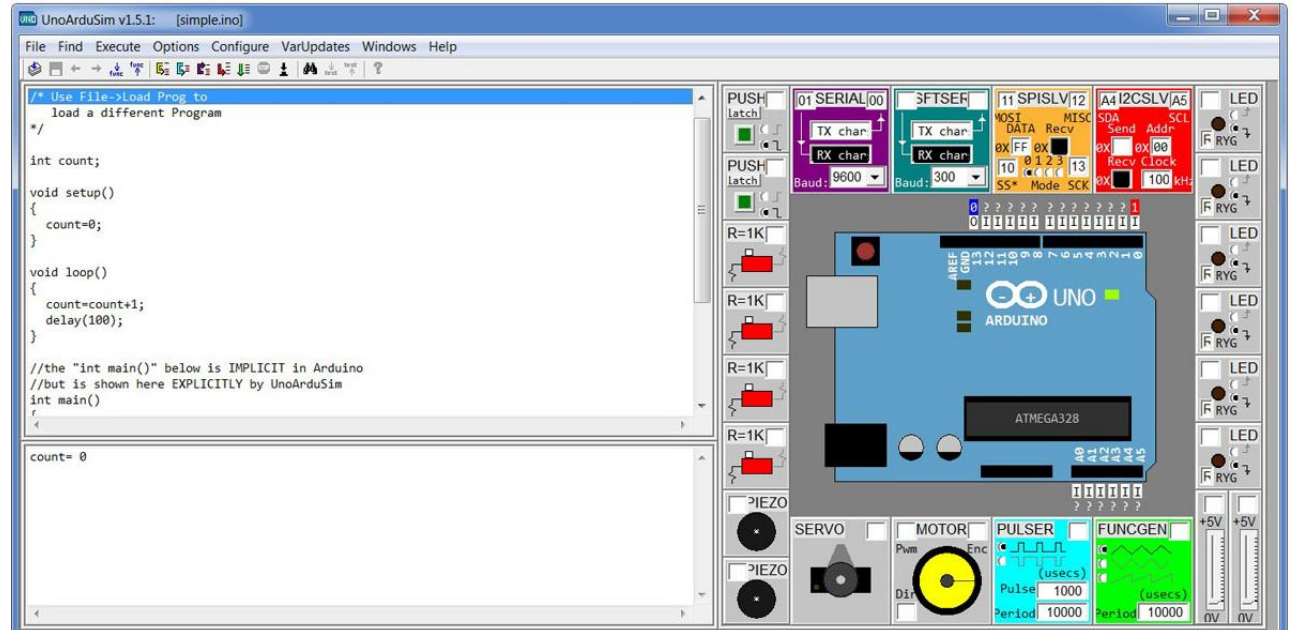
[масиви](#)

Симуляція роботи Arduino



TinkerCad


UnoArduSim




Arduino. TinkerCad

Gallery Designs Circuits

★ Staff favorites Small

★ 


 **Falling rock game**
(8 days ago) mr.punch ♡30 🗨️8


★ 

 **32 LEDs flashing using 74HC595**
(4 months ago) Valentin Cassayre ♡11 🗨️83

★ 

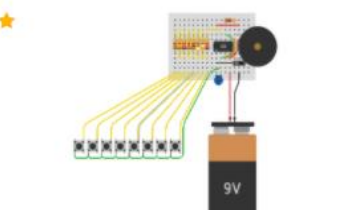
 **multiplexing**
(5 months ago) simmen ♡473 🗨️56


★ 

 **Line-Following Robot (LM393)**
(6 months ago) mr.punch ♡3 🗨️28

★ 

 **Arduino RC Robot**
(9 months ago) Helektrika ♡681 🗨️54

★ 

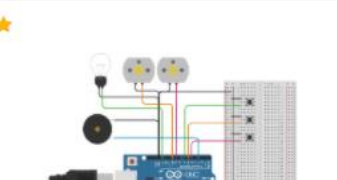
 **Tiny Piano - Mini breadboard**
(a year ago) mr.punch ♡617 🗨️45

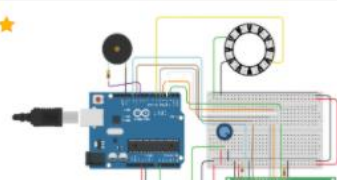
★ 

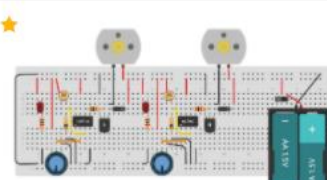
 **Arduino Datalogger - simple**
(2 years ago) jmscorella ♡151 🗨️12

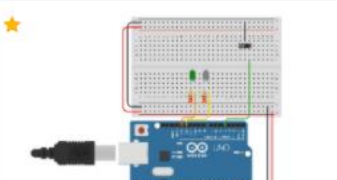
★ 

 **Movimiento servo**
(2 years ago) #aprendeConDynamo ♡254 🗨️16

★ 

★ 

★ 

★ 

Arduino. TinkerCad Starters

Basic Components All Components **Starters**

Search



Arduino



Blink



Fade



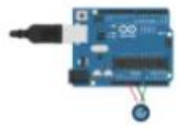
Button



Debounce



State Change Detection



Analog Input



Digital Read Serial



Analog Read Serial



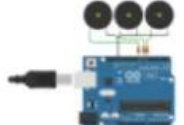
Servo



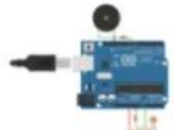
Tone Keyboard



Tone Melody



Tone Multiple



Tone Pitch Follows



Ultrasonic Range Finder



Motor



LED

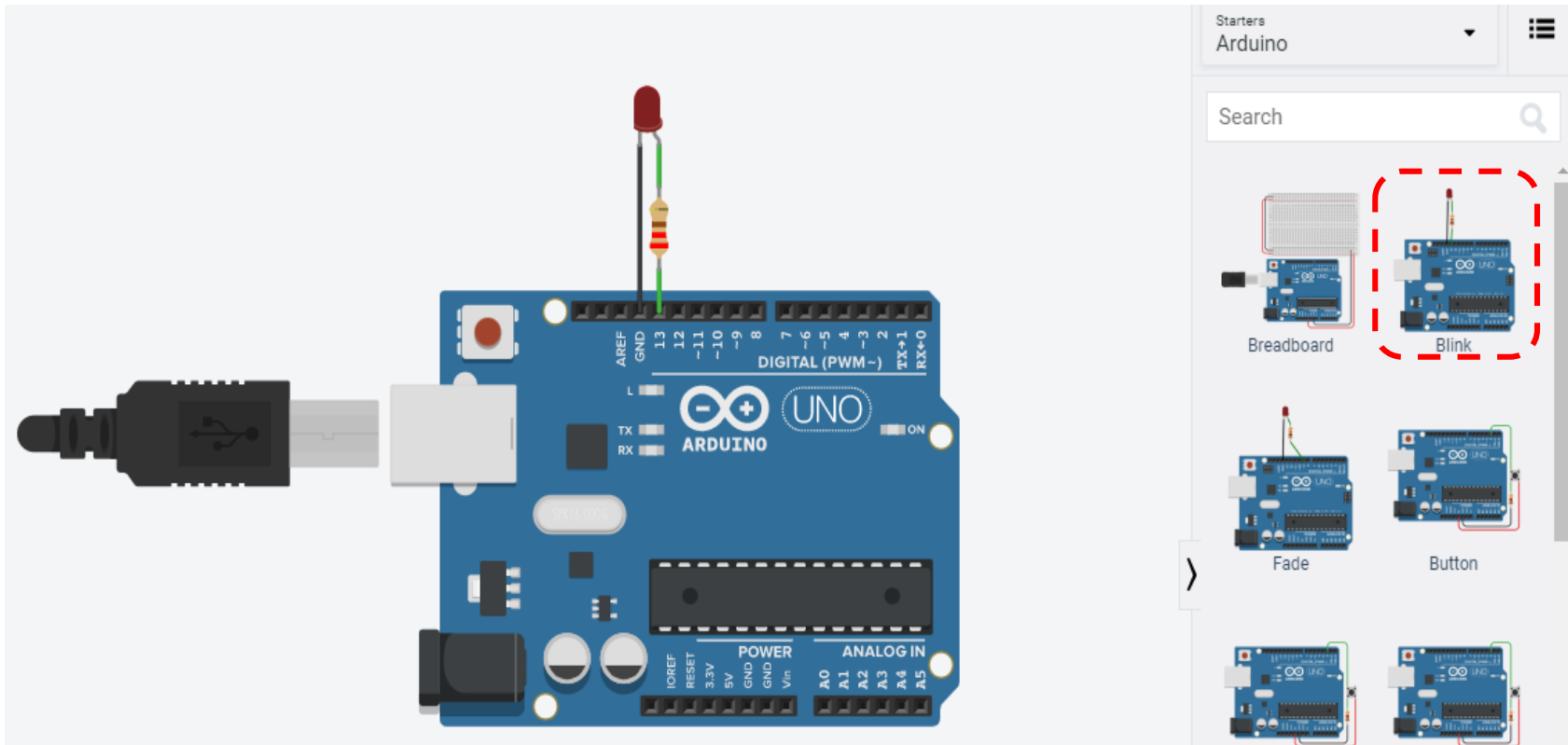


Analog to Digital Out



Calibration

Arduino. TinkerCad



Arduino. TinkerCad

Code

```
set built-in LED to HIGH
set pin 0 to HIGH
set pin 3 to 0
rotate servo on pin 0 to 0 degree
play speaker on pin 0 with tone 60
turn off speaker on pin 0
print to serial monitor hello world with
set RGB LED in pins 3 3 3
```

wait 1 secs

comment turn the LED on (HIGH is the voltage level)

comment turn the LED off by making the voltage LOW

```
26 void setup() {
27   // initialize digital pin LED_BUILTIN
28   pinMode(LED_BUILTIN, OUTPUT);
29 }
30
31 // the loop function runs over and
32 void loop() {
33   digitalWrite(LED_BUILTIN, HIGH);
34   delay(1000);
35   digitalWrite(LED_BUILTIN, LOW);
36   delay(1000);
37 }
```

```
1 /*
2  * This program blinks pin 13 of the Arduino (the
3  * built-in LED)
4  */
5
6 void setup()
7 {
8   pinMode(13, OUTPUT);
9 }
10
11 void loop()
12 {
13   // turn the LED on (HIGH is the voltage level)
14   digitalWrite(13, HIGH);
15   delay(1000); // Wait for 1000 millisecond(s)
16   // turn the LED off by making the voltage LOW
17   digitalWrite(13, LOW);
18   delay(1000); // Wait for 1000 millisecond(s)
19 }
```