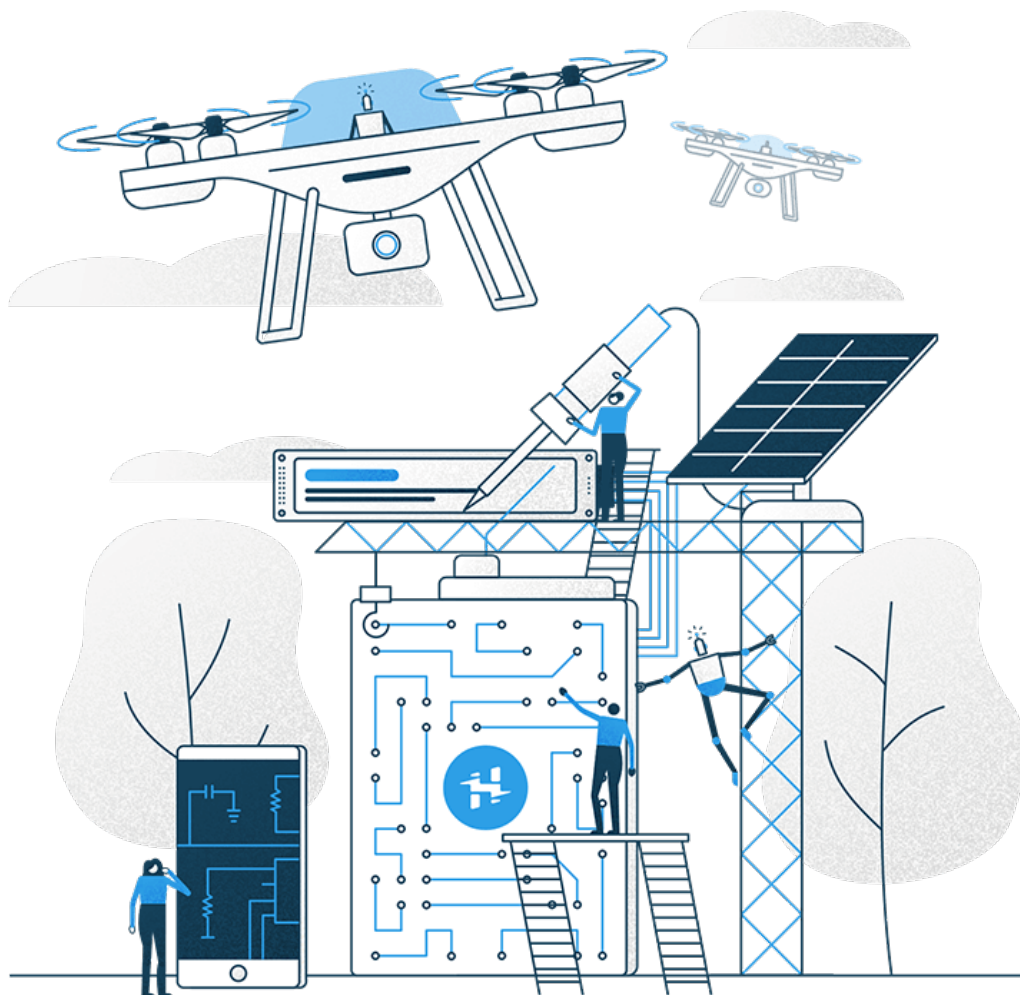


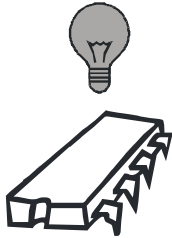
# МІКРОПРОЦЕСОРНІ СИСТЕМИ УПРАВЛІННЯ



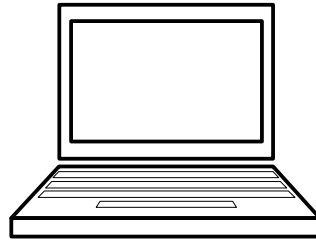
Lesson 2

# Етапи розробки програмного забезпечення

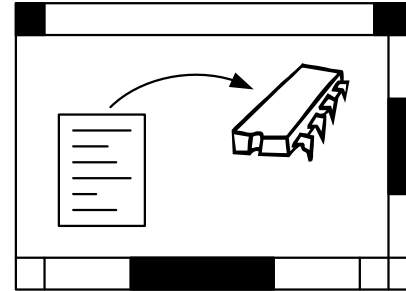
1. Чистий AVR нічого не робить



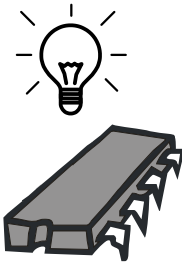
2. Пишемо програму на комп'ютері



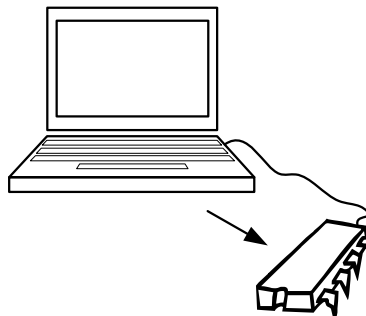
3. Програмування віртуального AVR в комп'ютері



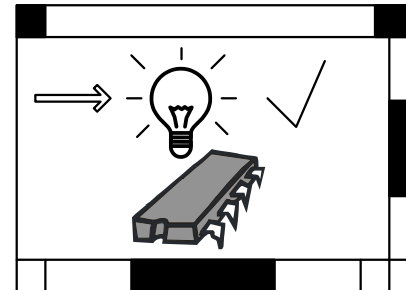
6. Перевіряємо програму в реальному пристрої



5. Програмування реального AVR



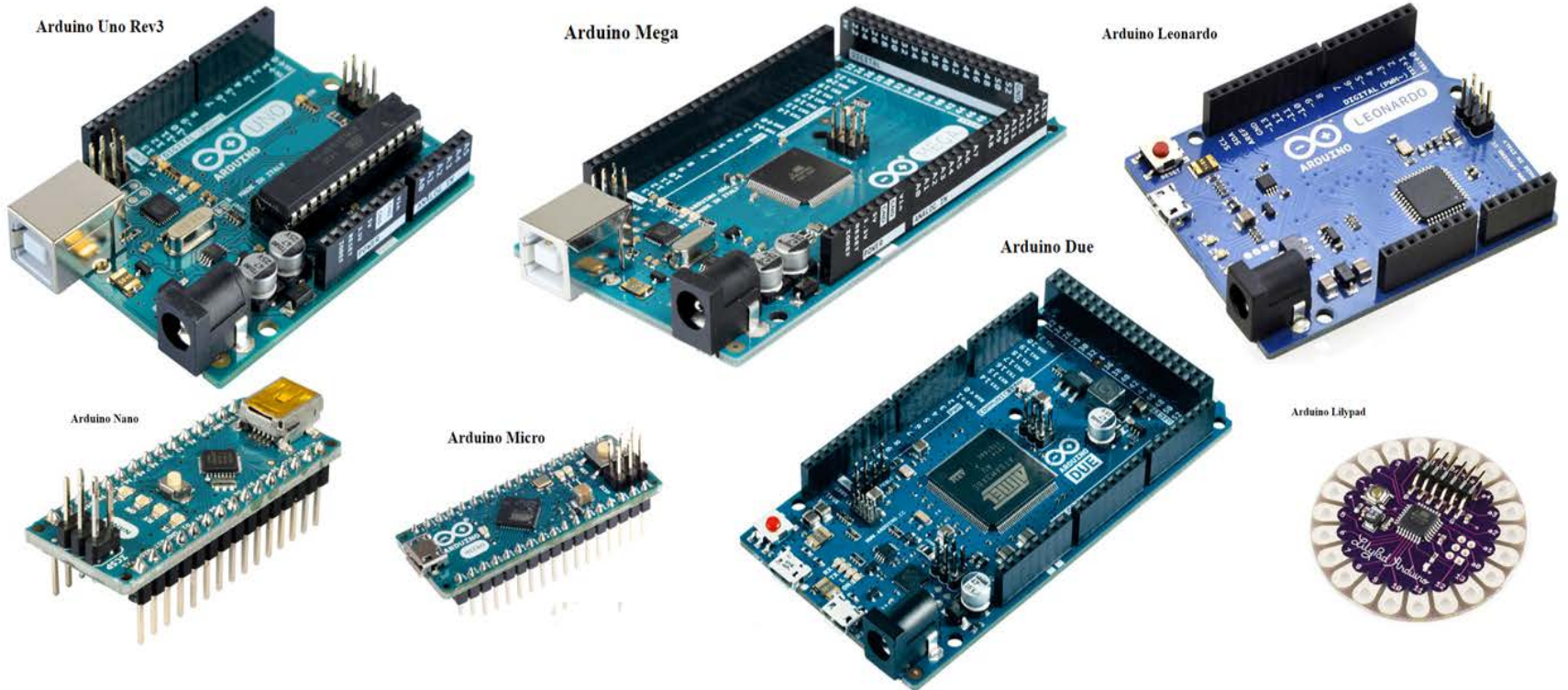
4. Перевіряємо програму на комп'ютері



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.

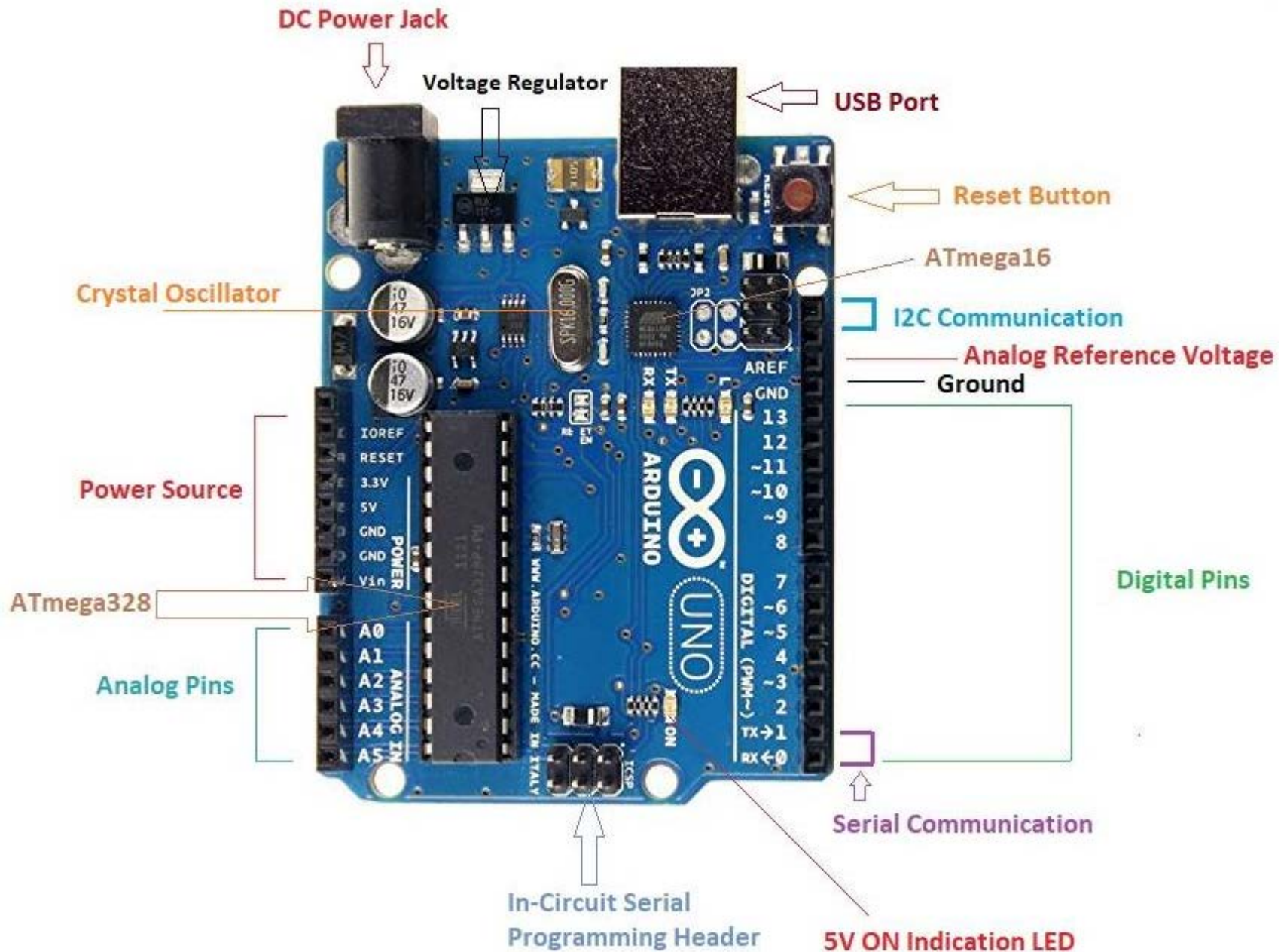
# Arduino

UNO, Mega, Nano, Micro, Due, Leonardo, LilyPad



Візуально **Arduino** являє собою невелику друковану плату з встановленим МК, кварцовим резонатором, стабілізатором живлення, конвертором USB-UART та уніфікованої «гребінкою» контактів, які підводять сигнали до цифрових та аналогових портів.

# Arduino UNO



# Arduino UNO

DC Power Jack



USB Port

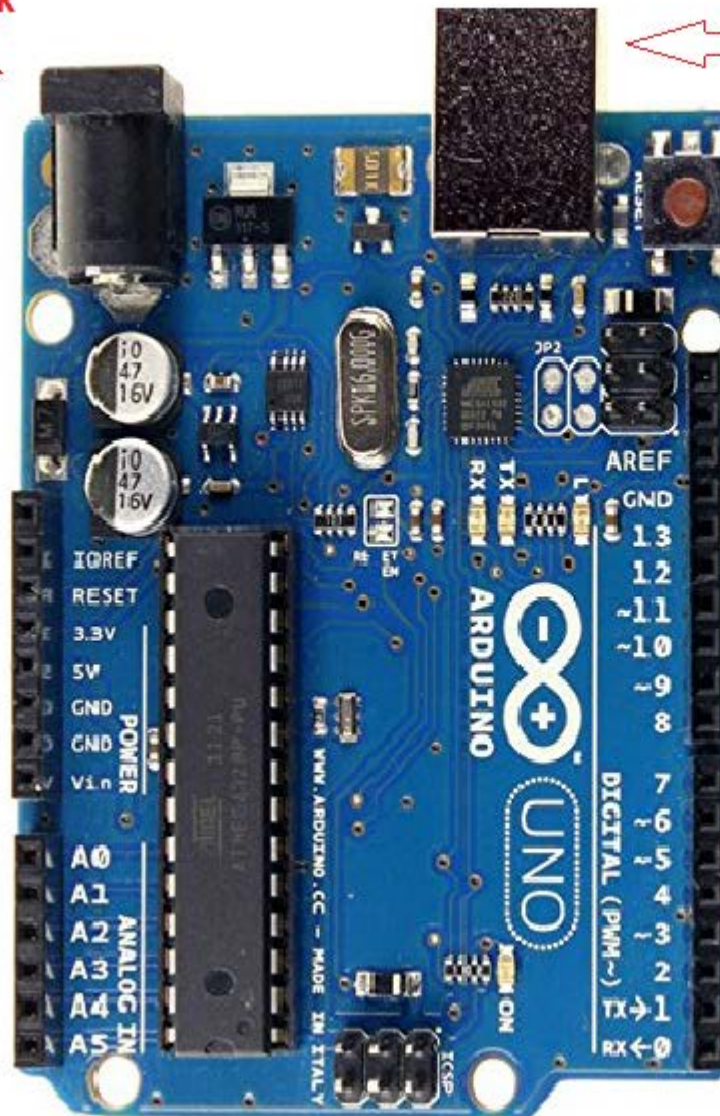


Reset Button



No Connection
5 V
Reset Input
3.3 V
5 V
Ground
Ground
Vin 7-12 V

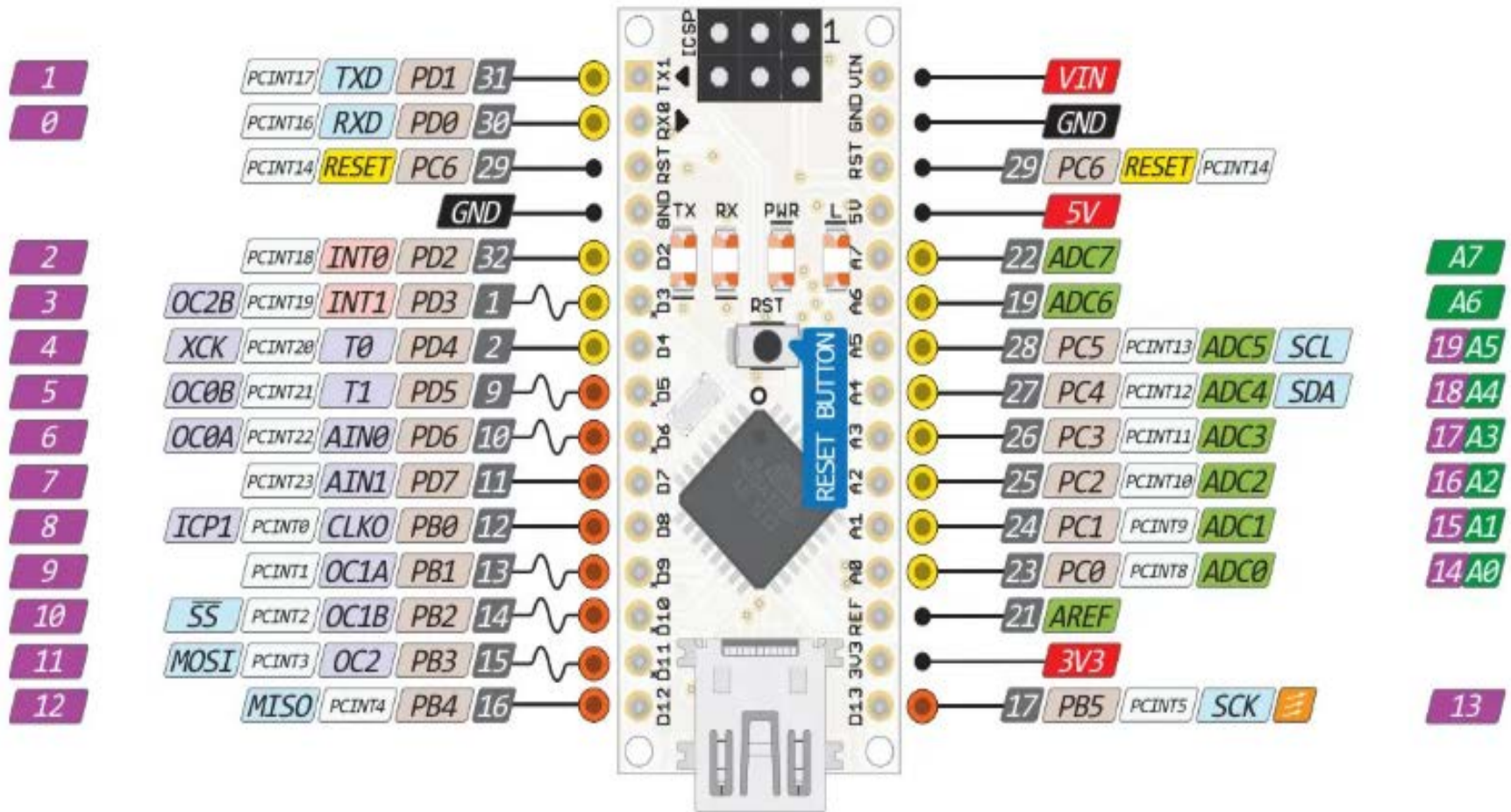
I2C/SDA	Analog Pin 0	A0
I2C/SCL	Analog Pin 1	A1
	Analog Pin 2	A2
	Analog Pin 3	A3
	Analog Pin 4	A4
	Analog Pin 5	A5



I2C/SCL	Serial Clock
I2C/SDA	Serial Data
Analog Reference Voltage	
Ground	

13	Digital Pin13	SPI/SCK	
12	Digital Pin12	SPI/MISO	
11	Digital Pin11	SPI/MOSI	PWM
10	Digital Pin10	SPI/SS	PWM
9	Digital Pin9		PWM
8	Digital Pin8		
7	Digital Pin7		
6	Digital Pin6	PWM	
5	Digital Pin5	PWM	
4	Digital Pin4		
3	Digital Pin3	Ext Int 1	PWM
2	Digital Pin2	Ext Int 0	
1	Digital Pin1	Serial Port TXD	
0	Digital Pin0	Serial Port RXD	

# Arduino Nano



# Середовище розробки Arduino

arduino.cc/en/software

Вхідні - sovmsvom... Входящие http://ai2.appinvent... Комісія "Радіотехні... Онлайн переводч... Repl.it - PossibleSc... Online C++ Compil... Online 3D Viewer

PROFESSIONAL EDUCATION STORE

Search on Arduino.cc

SIGN IN

HARDWARE SOFTWARE DOCUMENTATION COMMUNITY BLOG ABOUT

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 [Get](#)

**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/>

# Arduino Web Arduino

The screenshot displays the Arduino Web IDE interface. On the left is a teal sidebar with the title '> EDITOR'. Below the title are several menu items: 'Sketchbook' (with a folder icon), 'Examples' (with a list icon and highlighted by a red dashed box), 'Libraries' (with a book icon), 'Monitor' (with a magnifying glass icon), 'Help' (with a question mark icon), 'Preferences' (with a list icon), and 'Features usage' (with a circular arrow icon). The main area is divided into two panels. The left panel, titled 'sketch\_may4a', contains a 'NEW SKETCH' button, a search bar labeled 'SEARCH SKETCHBOOK', and a list of sketches ordered by 'LAST MODIFIED'. A sketch named 'sketch\_may4a' is listed. Below this is a large upward-pointing arrow icon and the text: 'Import your sketches to your online Sketchbook and access them from any device!'. The right panel, also titled 'sketch\_may4a', shows a code editor with a dropdown menu set to 'Arduino Uno' and a 'SHARE' button. Below this is a file explorer showing 'sketch\_may4a.ino' (selected) and 'ReadMe.adoc'. The code editor displays the following C++ code:

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

# Arduino Web Arduino

The screenshot displays the Arduino Web IDE 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 left pane shows a search bar and a list of examples for the UNO board, categorized by section (01.BASICS, 02.DIGITAL, etc.). The 'Blink' example is highlighted with a red dashed box. The middle pane shows the 'Blink.ino' file selected, with a description of the example. The right pane displays the C++ code for the Blink example. At the bottom, a green status bar indicates 'Success: Done verifying Blink', and a terminal window shows the compilation output.

**EDITOR**

Sketchbook

Examples

Libraries

Monitor

Help

Preferences

Features usage

powered by amazon web services

SEARCH EXAMPLES

SHOWING EXAMPLES FOR UNO

BUILT IN FROM LIBRARIES

01.BASICS (6)

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

02.DIGITAL (9)

03.ANALOG (6)

04.COMMUNICATION (12)

05.CONTROL (6)

06.SENSORS (4)

07.DISPLAY (2)

Blink

✓ → ✗ Arduino Uno

Blink.ino

9 If you want to know what  
10 model, check the Technica  
11 <https://www.arduino.cc/en/>  
12  
13 modified 8 May 2014  
14 by Scott Fitzgerald  
15 modified 2 Sep 2016  
16 by Arturo Guadalupi  
17 modified 8 Sep 2016  
18 by Colby Newman  
19  
20 This example code is in t  
21  
22 <http://www.arduino.cc/en/>  
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 level)  
34 delay(1000); // wait for a second  
35 digitalWrite(LED\_BUILTIN, LOW); // turn the LED off by making the voltage LOW  
36 delay(1000); // wait for a second  
37 }  
38

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 Плавка Скетч Инструменты Помощь

Новый Ctrl+N  
Открыть... Ctrl+O  
Открыть недавние  
Папка со скетчами

Примеры  
Заккрыть Ctrl+W  
Сохранить Ctrl+S  
Сохранить как... Ctrl+Shift+S

Настройки страницы Ctrl+Shift+P  
Печать Ctrl+P

Настройки Ctrl+Comma

Выход Ctrl+Q

> dControl.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  
ВЫВЫТ

Примеры для Arduino Nano

EEPROM  
SoftwareSerial

AnalogReadSerial  
BareMinimum  
Blink  
DigitalReadSerial  
Fade  
ReadAnalogVoltage

Blink | Arduino 1.8.5

Файл Плавка 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](#)

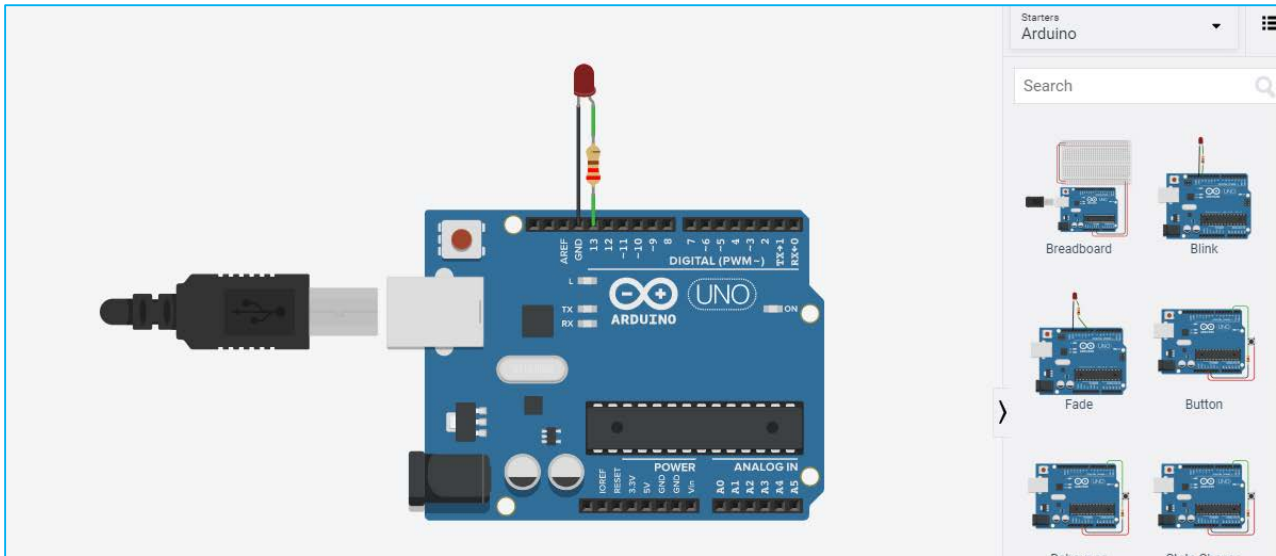
32 біта (4 байта) у діапазоні від  $-3.4028235E+38$  до  $3.4028235E + 38$

[string](#) - масив символів

[String](#) - об'єкт

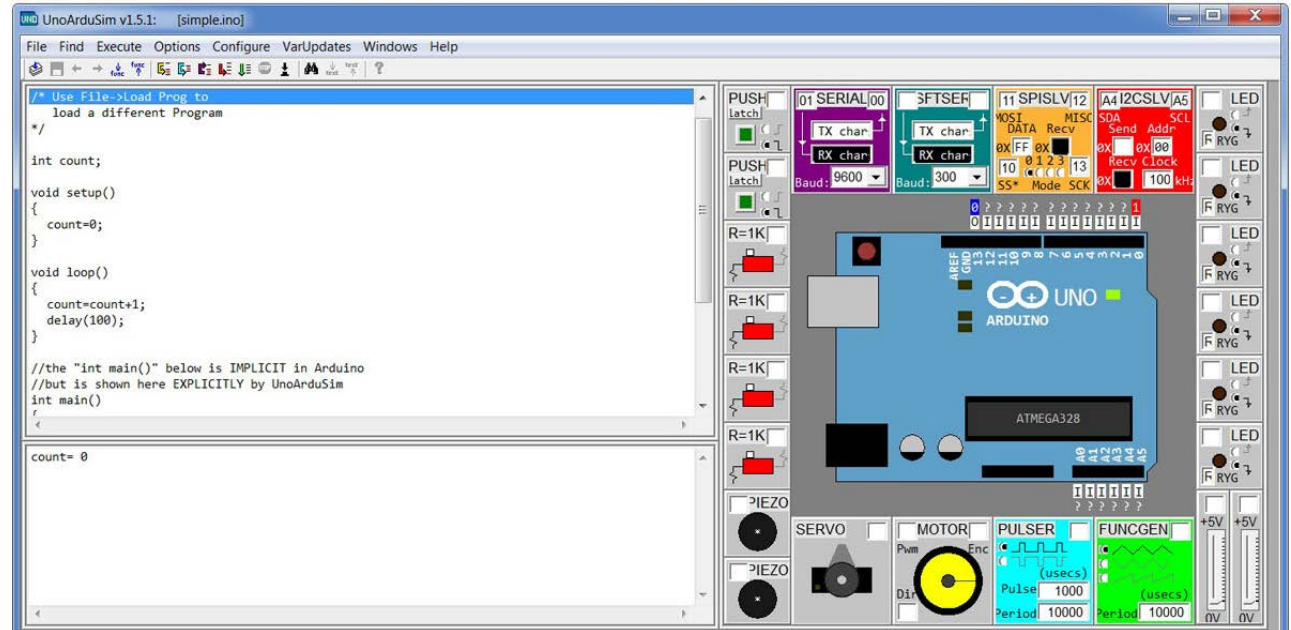
[масиви](#)

# Симуляція роботи Arduino



TinkerCad

UnoArduSim



<https://kolotushkin.com/indexsoft.php>

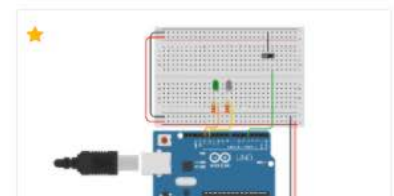
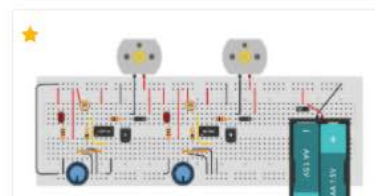
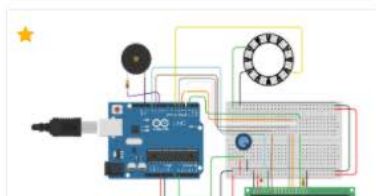
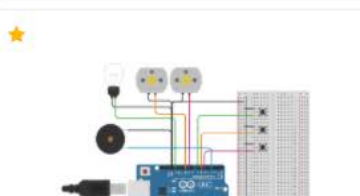
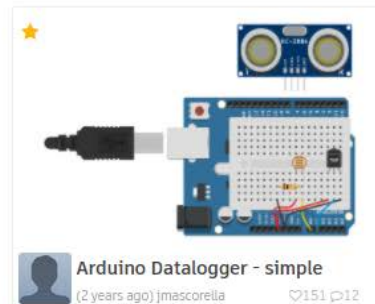
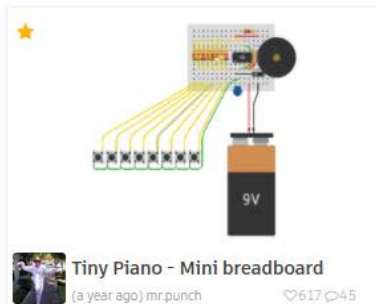
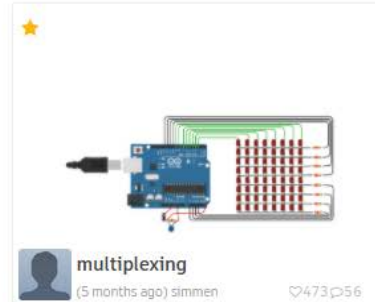
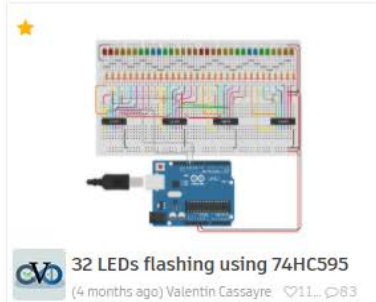
# Arduino. TinkerCad



AUTODESK®  
TINKERCAD™

[Gallery](#)[Blog](#)[Learn](#)[Teach](#)

## Gallery

[Designs](#)[Circuits](#)[★ Staff favorites ▾](#)[Small](#)

# 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



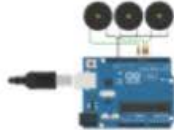
Servo



Tone Keyboard



Tone Melody



Tone Multiple



Tone Pitch Buzzer



Ultrasonic Range Finder



Motor



LCD

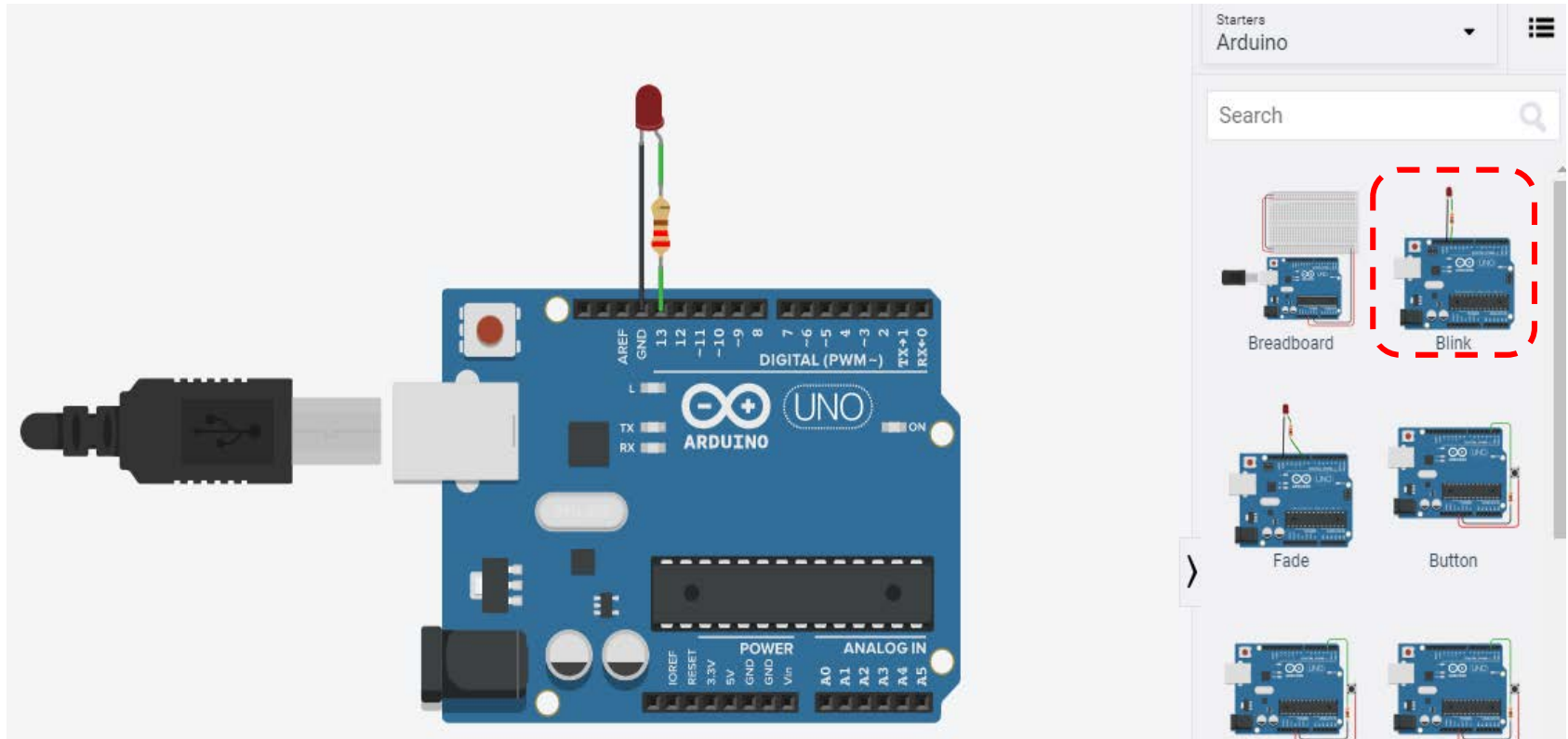


Analog to Digital Out

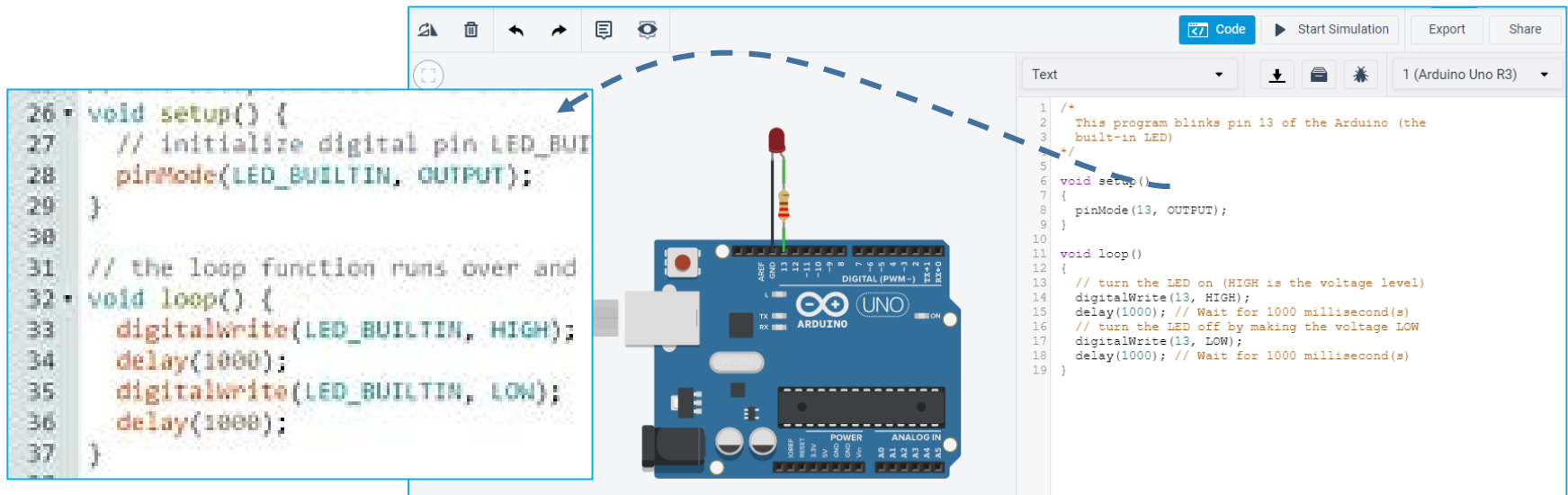
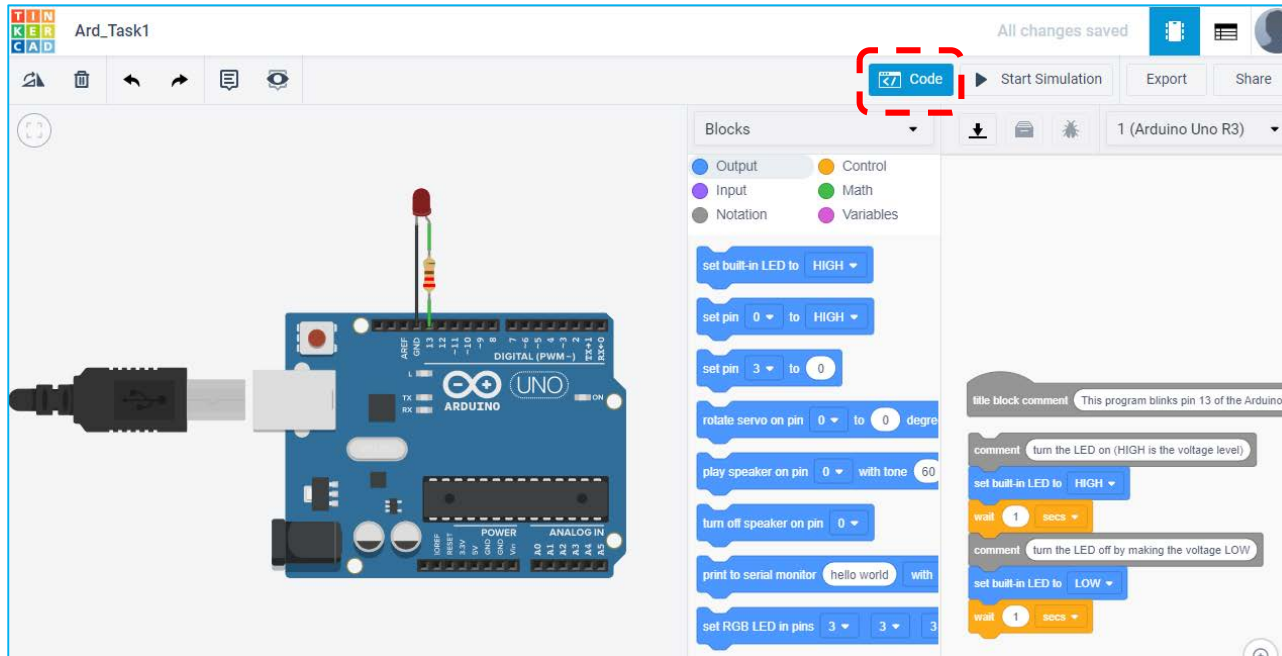


Calibration

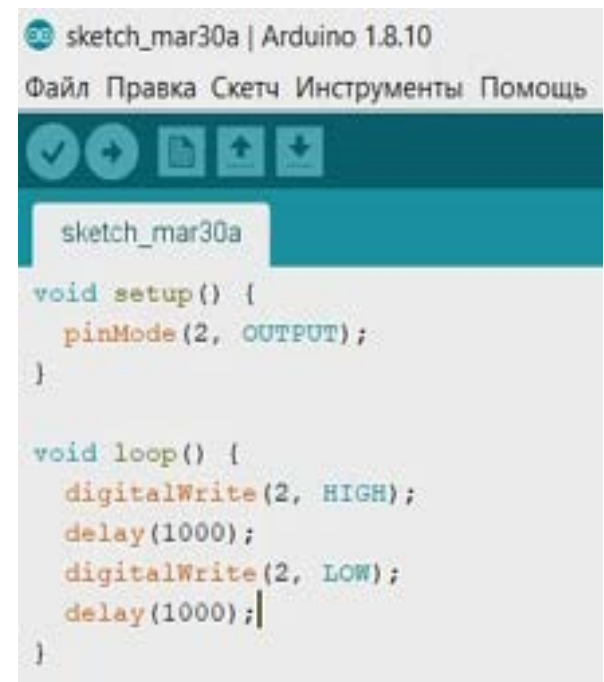
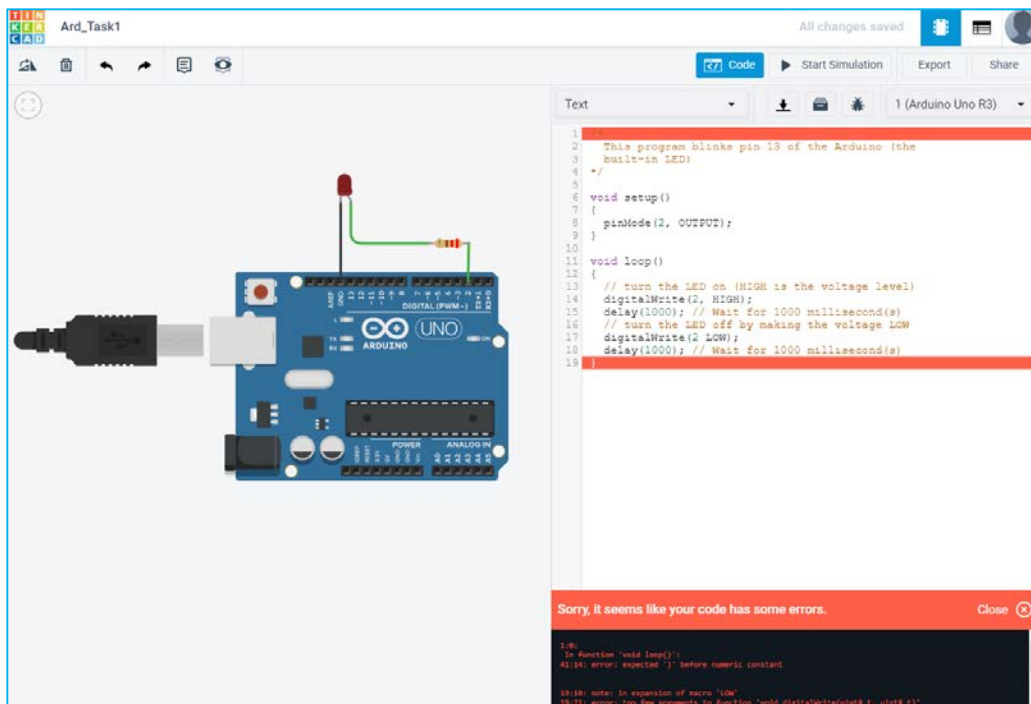
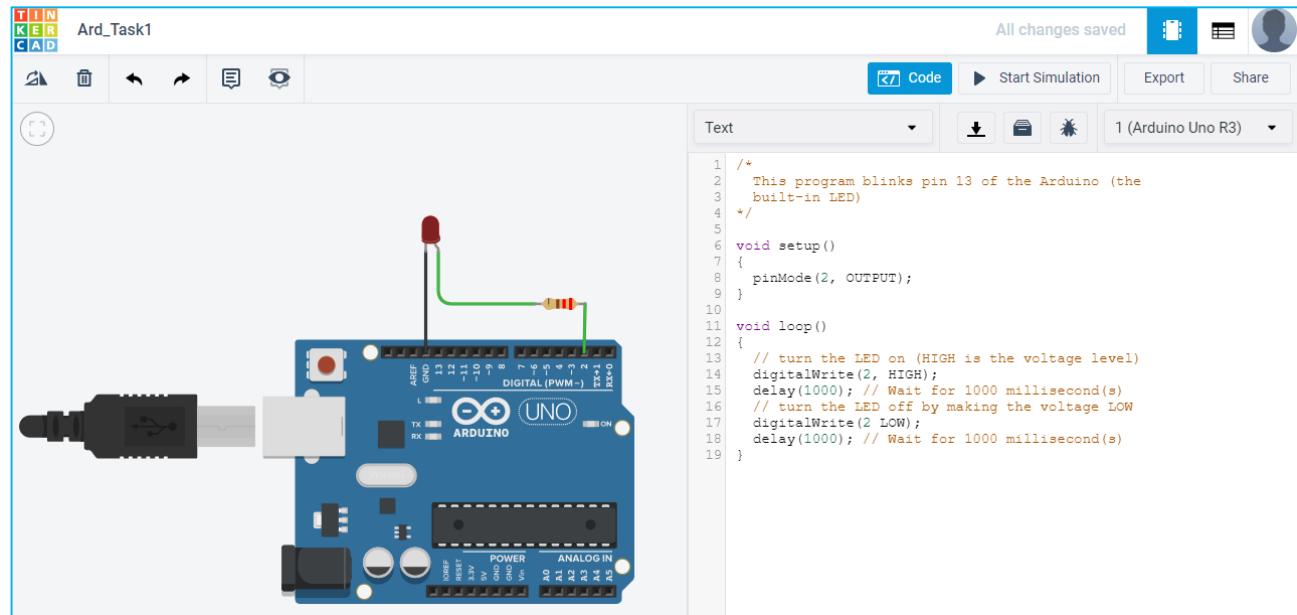
# Arduino. TinkerCad



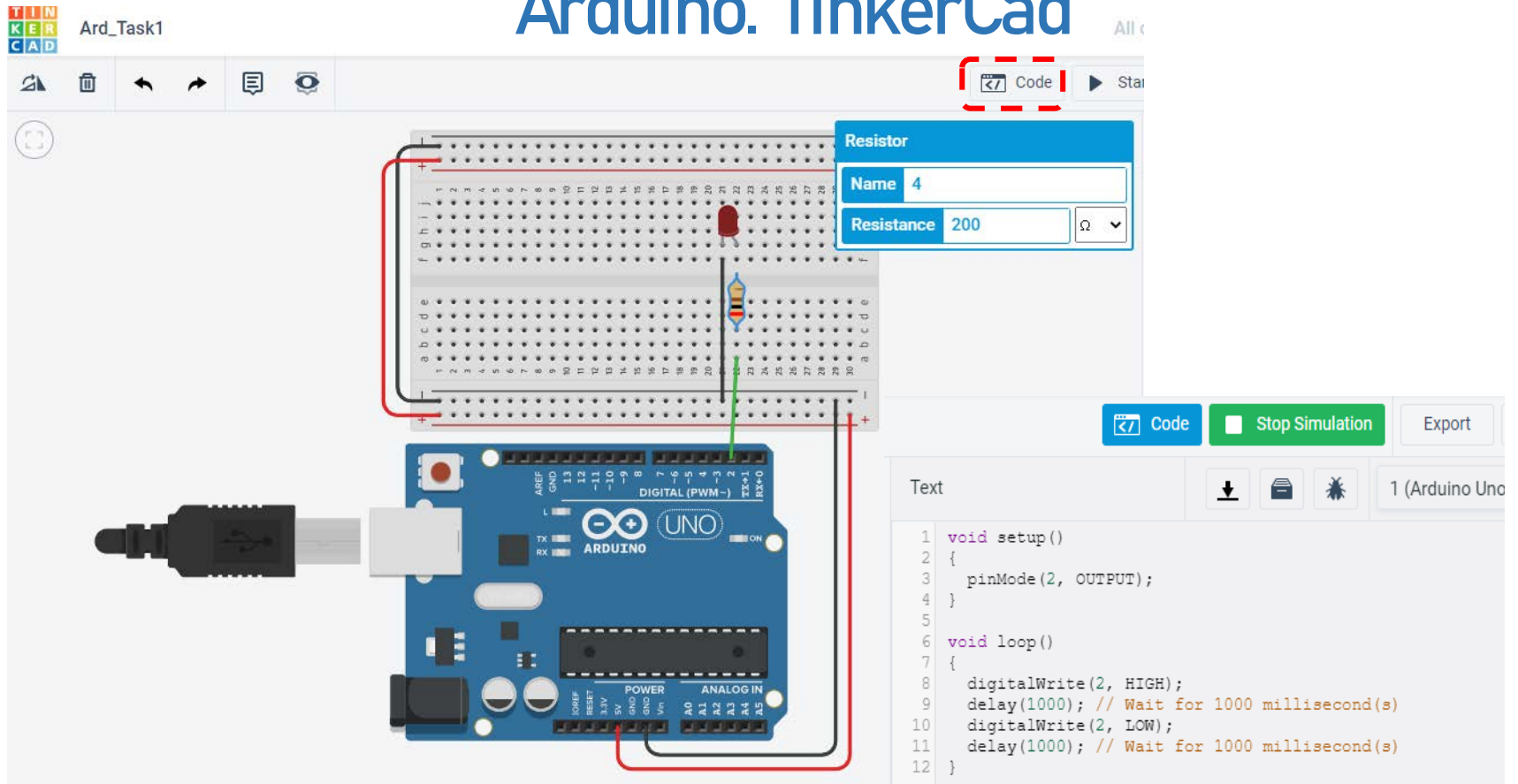
# Arduino. TinkerCad



# Arduino. TinkerCad



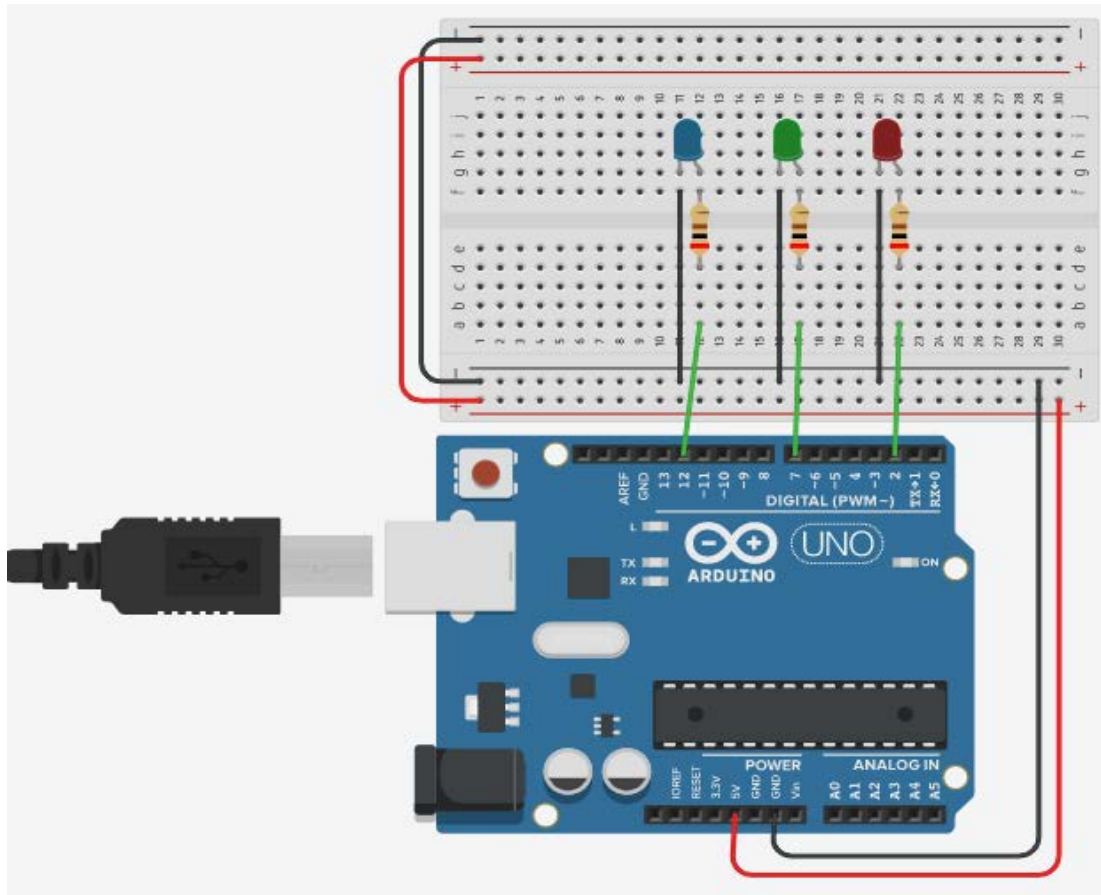
# Arduino. TinkerCad



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

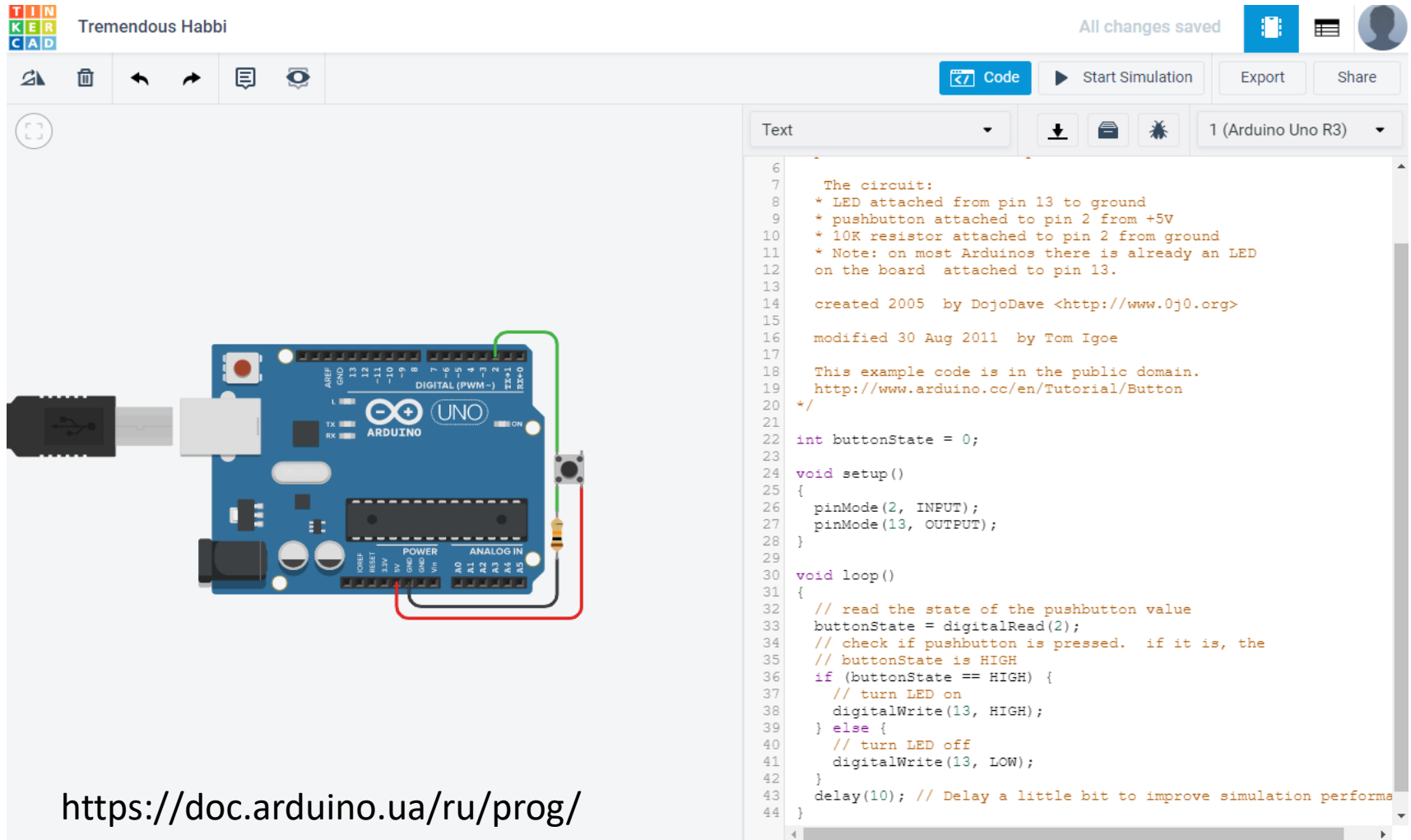
void loop() {
  digitalWrite(2, HIGH);
  delay(1000);
  digitalWrite(2, LOW);
  delay(1000);
}
```

# Arduino. TinkerCad



```
1  int yd=2;  
2  int gd=7;  
3  int bd=12;  
4  
5  
6  void setup()  
7  {  
8      pinMode(yd, OUTPUT);  
9      pinMode(gd, OUTPUT);  
10     pinMode(bd, OUTPUT);  
11 }  
12  
13 void loop()  
14 {  
15     digitalWrite(yd, HIGH);  
16     digitalWrite(gd, LOW);  
17     digitalWrite(bd, LOW);  
18     delay(1000);  
19  
20     digitalWrite(gd, HIGH);  
21     digitalWrite(yd, LOW);  
22     digitalWrite(bd, LOW);  
23     delay(1000);  
24  
25     digitalWrite(bd, HIGH);  
26     digitalWrite(gd, LOW);  
27     digitalWrite(yd, LOW);  
28     delay(1000);  
29  
30 }
```

# Arduino. TinkerCad



The screenshot displays the TinkerCad web interface. On the left, a 3D model of an Arduino Uno R3 is shown with a USB Type-C cable connected to its left side. A pushbutton is connected to the digital pins: one leg to pin 2 and the other to ground. An LED is connected to pin 13 and ground. On the right, the code editor shows the following C++ code:

```
6
7
8   The circuit:
9   * LED attached from pin 13 to ground
10  * pushbutton attached to pin 2 from +5V
11  * 10K resistor attached to pin 2 from ground
12  * Note: on most Arduinos there is already an LED
13    on the board attached to pin 13.
14
15  created 2005 by DojoDave <http://www.0j0.org>
16
17  modified 30 Aug 2011 by Tom Igoe
18
19  This example code is in the public domain.
20  http://www.arduino.cc/en/Tutorial/Button
21  */
22
23  int buttonState = 0;
24
25  void setup()
26  {
27    pinMode(2, INPUT);
28    pinMode(13, OUTPUT);
29  }
30
31  void loop()
32  {
33    // read the state of the pushbutton value
34    buttonState = digitalRead(2);
35    // check if pushbutton is pressed. if it is, the
36    // buttonState is HIGH
37    if (buttonState == HIGH) {
38      // turn LED on
39      digitalWrite(13, HIGH);
40    } else {
41      // turn LED off
42      digitalWrite(13, LOW);
43    }
44    delay(10); // Delay a little bit to improve simulation performance
```

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

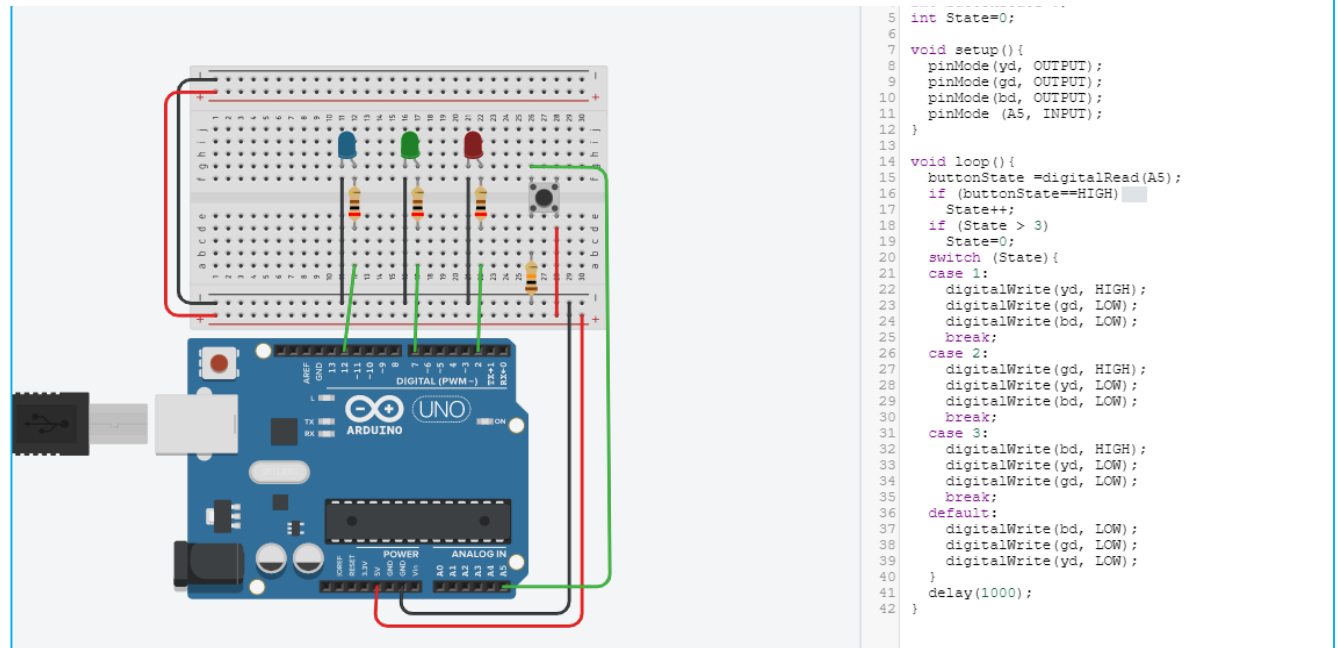
<https://alexgyver.ru/lessons/arduino-reference/>

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

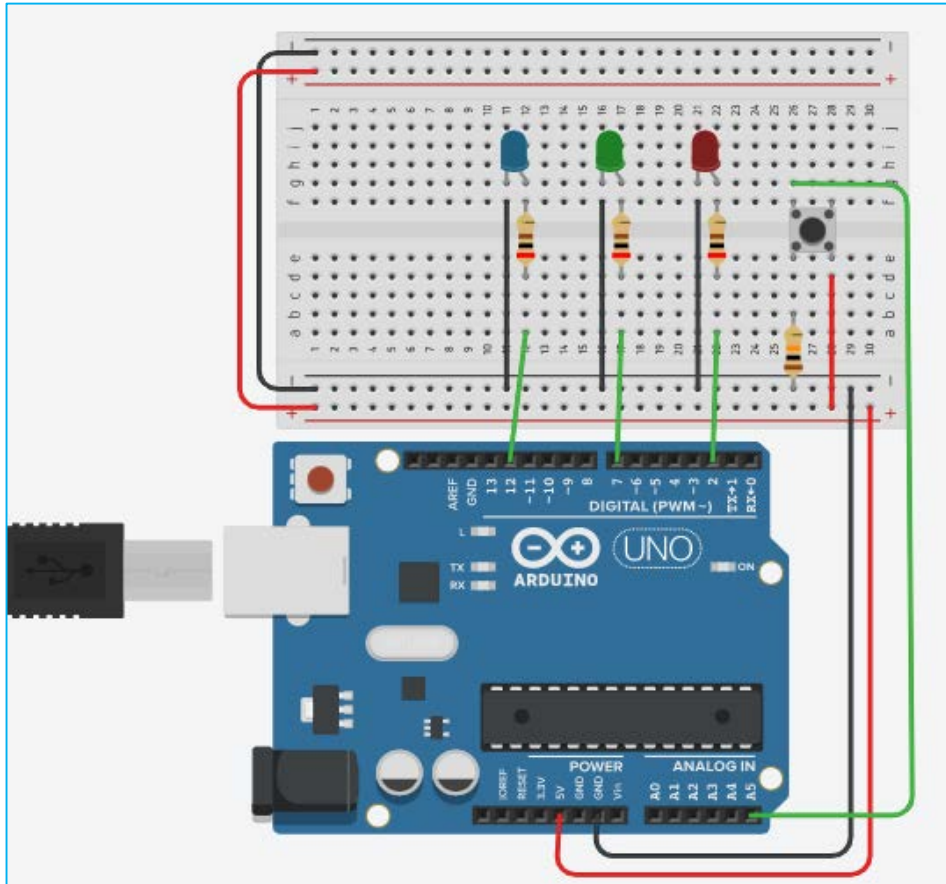
# Arduino. TinkerCad

```
switch (var) {  
  case 1:  
    //виконується, коли var дорівнює 1  
    break;  
  case 2:  
    // виконується, коли var дорівнює 2  
    break;  
  default:  
    // виконується, коли не вибрана жодна з альтернатив  
    // default необов'язковий  
}
```

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



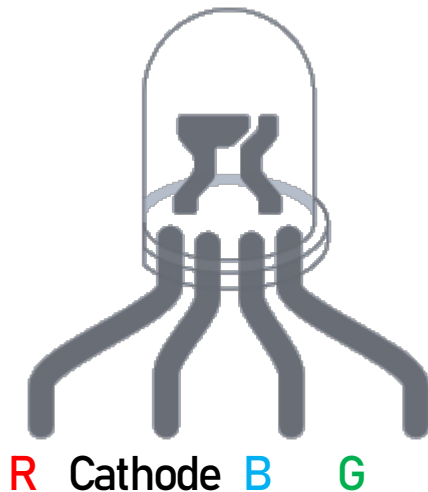
# Arduino. TinkerCad



```
1  int yd=2;
2  int gd=7;
3  int bd=12;
4  int buttonState=0;
5  int State=0;
6
7  void setup() {
8      pinMode(yd, OUTPUT);
9      pinMode(gd, OUTPUT);
10     pinMode(bd, OUTPUT);
11     pinMode(A5, INPUT);
12 }
13
14 void loop() {
15     buttonState =digitalRead(A5);
16     if (buttonState==HIGH)
17         State++;
18     if (State > 3)
19         State=0;
20     switch (State){
21     case 1:
22         digitalWrite(yd, HIGH);
23         digitalWrite(gd, LOW);
24         digitalWrite(bd, LOW);
25         break;
26     case 2:
27         digitalWrite(gd, HIGH);
28         digitalWrite(yd, LOW);
29         digitalWrite(bd, LOW);
30         break;
31     case 3:
32         digitalWrite(bd, HIGH);
33         digitalWrite(yd, LOW);
34         digitalWrite(gd, LOW);
35         break;
36     default:
37         digitalWrite(bd, LOW);
38         digitalWrite(gd, LOW);
39         digitalWrite(yd, LOW);
40     }
41     delay(1000);
42 }
```

# Домашнє завдання

Замінити 3 світлодіода одним RGB світлодіодом.  
Кнопкою передбачити 8 режимів роботи



**RGB**

```
000 // black (Led off)
001 // green
010 // blue
011 // cyan
100 // red
101 // yellow
110 // purple
111 // white
```