

Prižiganje lučk z Arduinom



Kaj bomo danes počele?

Spoznale:

- Arduino Nano
- Testno ploščico (Breadboard)
- LED lučke, upornike, žičke, foto senzorje
- Osnove programiranja v C++ (spremenljivke, funkcija, for zanka)

Koda:

- <https://github.com/22nds/lfu-arduino-basics>

Sestavni deli

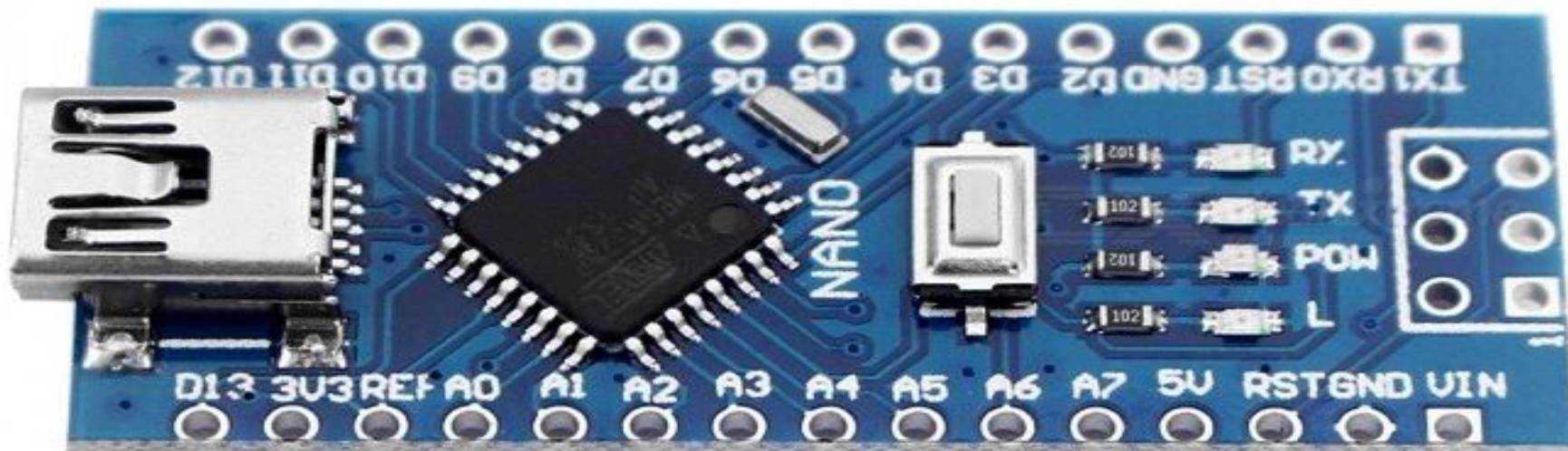
- 2 x LED
 - 1 x RGB LED
 - 3 x 220 Ohm upornik
 - 1 x 1k Ohm upornik
 - 2 x žičke
 - 1 x senzor svetlobe
 - Testna ploščica (Breadboard)
-

- USB kabel
- Računalnik
- Programska oprema (Arduino, Processing)

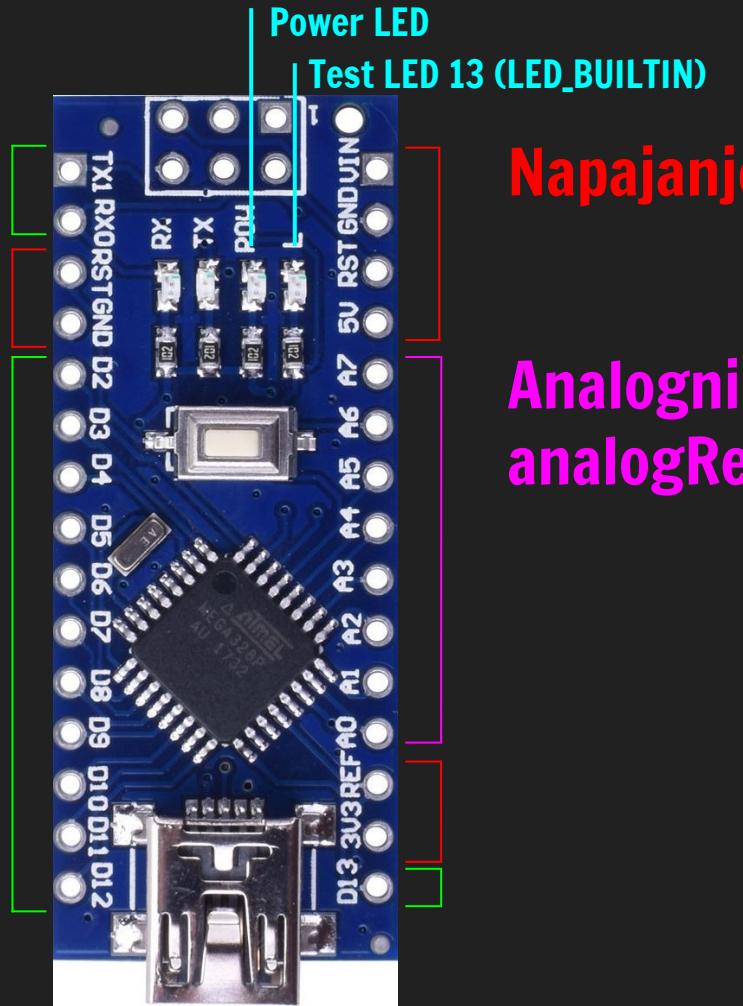
Arduino Nano

Arduino Uno ->

- Uradna stran: <https://store.arduino.cc/arduino-nano>
- Gonilniki: <http://sparks.gogo.co.nz/ch340.html>



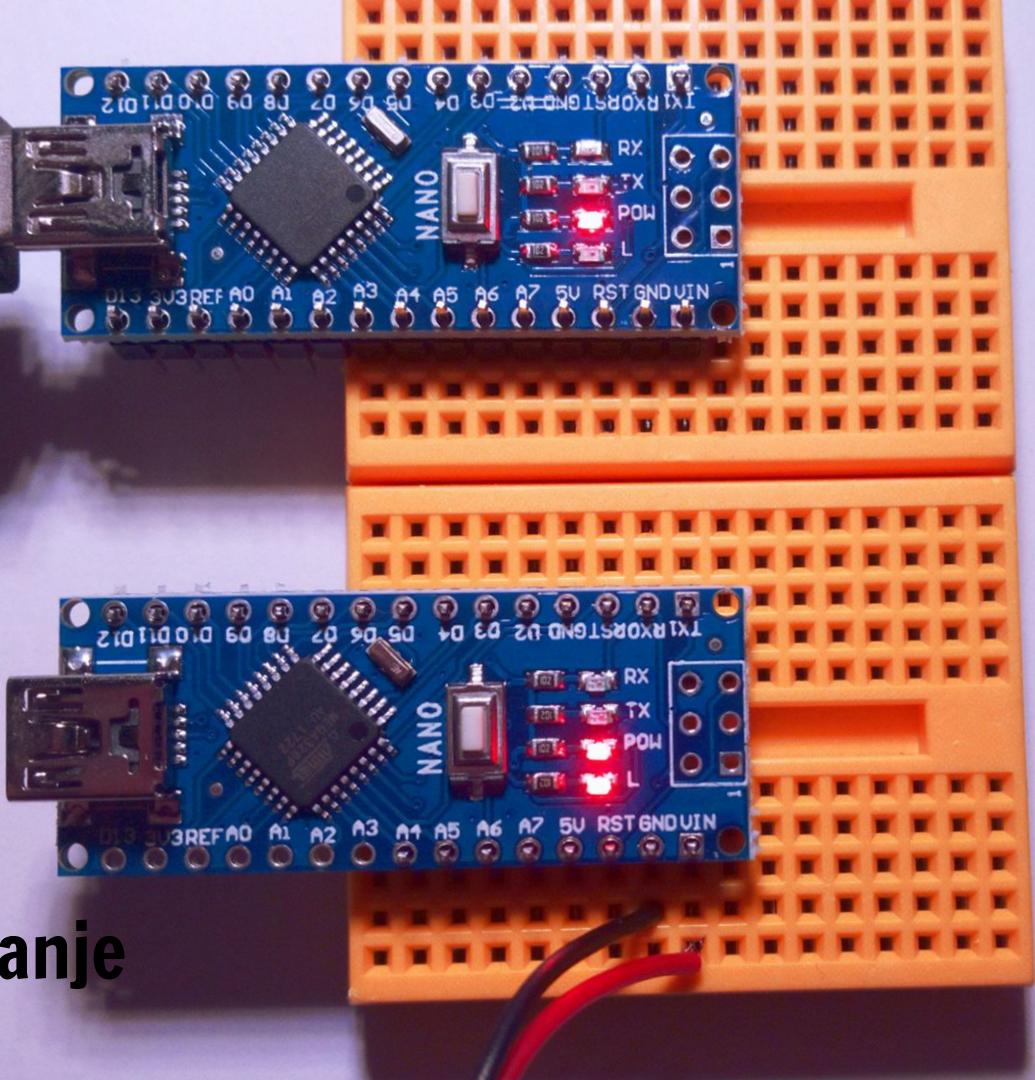
Digitalni pini
pinMode()
DigitalRead()
DigitalWrite()
AnalogWrite() - PWM
3, 5, 6, 9, 10, 11



Napajanje

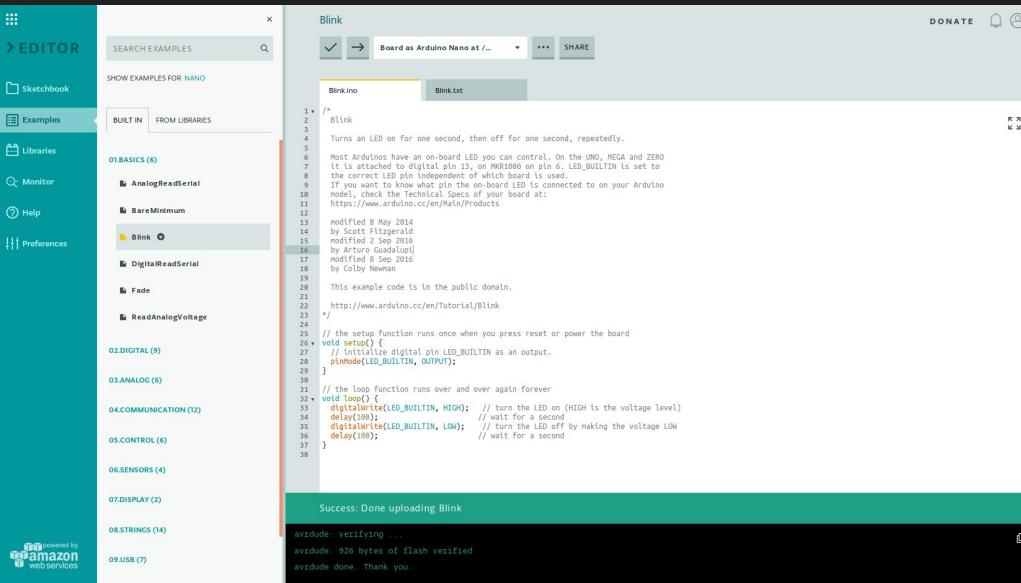
Analogni pini
analogRead()

Napajanje



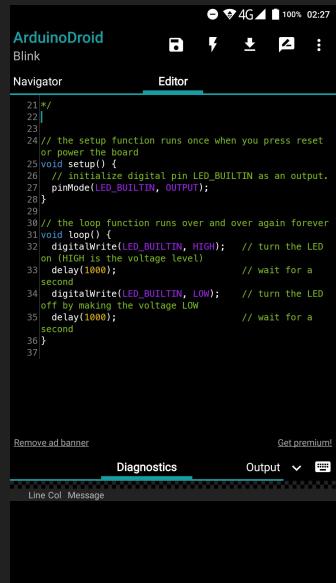
Programska oprema za Arduino

- Arduino Editor (online) <https://create.arduino.cc/editor/> ali
- Arduino IDE <https://www.arduino.cc/en/Main/Software> ali
- ArduinoDroid (Android): <https://play.google.com/store/apps/details?id=name.antonsmirnov.android.arduinodroid2>



The screenshot shows the Arduino IDE 1.8.5 interface. The title bar says 'Blink | Arduino 1.8.5'. The code is identical to the one in the online editor:

```
1 // 
2 // Blink
3 //
4 // Turns an LED on for one second, then off for one second, repeatedly.
5 // Most Arduinos have an on-board LED you can control. On the UNO, MEGA or ZERO
6 // it is attached to digital pin 13, on MKR1000 on pin 6, LED_BUILTIN is set to
7 // that value. You can change this pin in the Sketch Preferences ("Digital pins used for
8 // the on-board LED") if you want to know what pin the on-board LED is connected to on your Arduino
9 // model, check the Technical Specs of your board at:
10 // https://www.arduino.cc/en/Main/Products
11 //
12 // modified 8 May 2014
13 // by Scott Fitzgerald
14 // modified 2 Sep 2016
15 // by Arturo Guadalupi
16 // modified 8 Sep 2016
17 // by Colby Newman
18 //
19 // This example code is in the public domain.
20 //
21 // http://www.arduino.cc/en/Tutorial/Blink
22 //
23 // the setup function runs once when you press reset or power the board
24 // void setup() {
25 //   // initialize digital pin LED_BUILTIN as an output.
26 //   pinMode(LED_BUILTIN, OUTPUT);
27 // }
28 //
29 // the loop function runs over and over again forever
30 // void loop() {
31 //   // turn the LED on (HIGH is the voltage level)
32 //   digitalWrite(LED_BUILTIN, HIGH);    // wait for a second
33 //   delay(1000);
34 //   // turn the LED off by making the voltage LOW
35 //   digitalWrite(LED_BUILTIN, LOW);    // wait for a second
36 //   delay(1000);
37 // }
```



Spletni urejevalnik

LOGIN

USERNAME

PASSWORD

[Forgot your username/password?](#)

LOGIN

CREATE A NEW ACCOUNT

**SIGN UP****SIGN UP**

USERNAME *

PASSWORD (min 8 characters) *

CONFIRM PASSWORD *

EMAIL *

I'm not a robot 
reCAPTCHA Privacy Terms

Subscribe to the Arduino Newsletter

CREATE ACCOUNT

ALREADY HAVE AN ACCOUNT?

LOGIN

JOIN THE ARDUINO COMMUNITY!

- MAKE NEW PROJECTS WITH ARDUINO CREATE
- PURCHASE ON THE ARDUINO STORE
- ASK A QUESTION ON THE ARDUINO FORUM

SIGN UP

Ifu

lezemuniverza@gmail.com

I'm not a robot 
reCAPTCHA Privacy Terms

Subscribe to the Arduino Newsletter

CREATE ACCOUNT

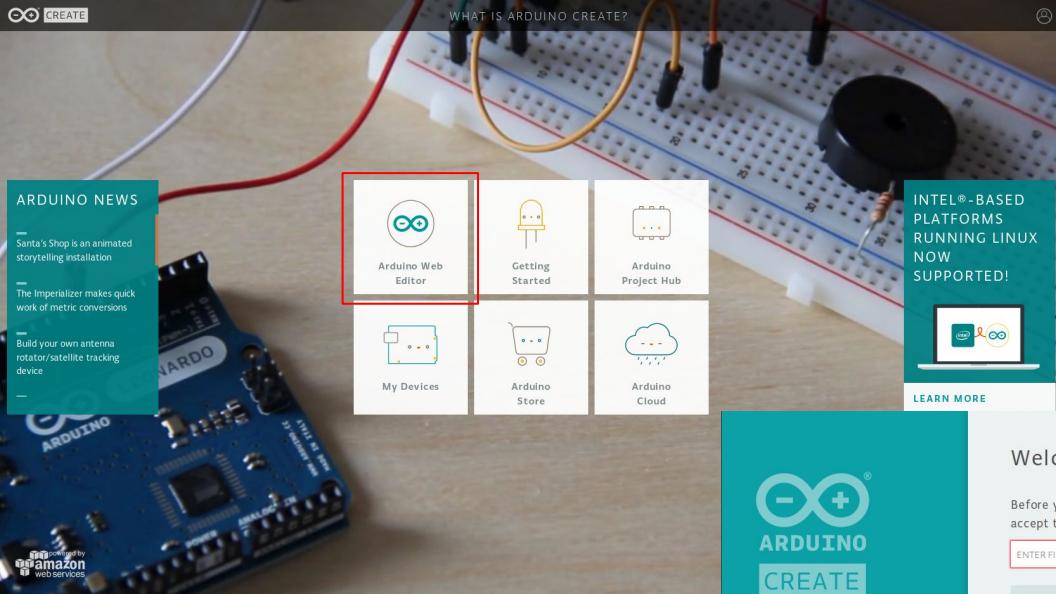
ALREADY HAVE AN ACCOUNT?

LOGIN

JOIN THE ARDUINO COMMUNITY!

- MAKE NEW PROJECTS WITH ARDUINO CREATE
- PURCHASE ON THE ARDUINO STORE
- ASK A QUESTION ON THE ARDUINO FORUM





Welcome to the Arduino Web Editor

Before you can start using the Arduino Web Editor we need you to provide us with some informations, and accept the following Terms & Condition:

ENTER FIRST NAME AND LAST NAME

Select Country

BCMI Labs SA. Create Terms of Service

Last updated: May 20, 2016

Please read these Terms of Service carefully before using the create.arduino.cc website.

1. Acceptance of Terms.

By using the Arduino Create Web site (the "Site") in any way, including using, transmitting, downloading, or uploading any of the services or functionality (the "Service") made available or enabled via the Site by BCMI Labs SA, or merely browsing the Site, you agree to these Terms of Service. You may not use the Service, or accept these Terms of Service, if (a) you are not of legal age to form a binding contract with BCMI Labs SA; or (b) you are prohibited by law from receiving or using the Service. If you are entering into these Terms of Service on behalf of a company or other legal entity, you represent that you have the authority to bind such entity to these Terms of Service, in which case "you" or "your" shall refer to such entity. BCMI Labs SA makes the Service available only if you have registered for an account. The Service may also be subject to a subscription or other agreement, posted guidelines, rules or terms of service ("Additional Terms"). If there is any conflict



Sign up to the Arduino Create Newsletter

DECLINE

AGREE



EDITOR

Sketchbook

Examples

Libraries

Monitor

Help

Preferences



NEW SKETCH

x

+



sketch_decl0a



→ -- Select Board or Port --



SHARE

SEARCH SKETCHBOOK



ORDER BY LAST MODIFIED

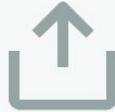
sketch_decl0a

sketch_decl0a.ino

ReadMe.adoc



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



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





EDITOR

NEW SKETCH



sketch_dec10a



-- Select Board or Port --



SHARE

Sketchbook

Examples

Libraries

Monitor

Help

Preferences



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

SEARCH SKETCHBOOK



ORDER BY LAST MODIFIED

sketch_dec10a

sketch_dec10a.ino

ReadMe.adoc

```
1 /*
2 */
3 */
4
5 void setup() {
6
7 }
```



NOTE: You can also import your whole Sketchbook!

Import all your sketches and libraries at once:

1. Compress the Arduino folder that contains all your sketches and custom libraries in .Zip format
2. Press 'IMPORT', a file system window will be opened
3. Select the file Arduino.zip
4. You are good to go!

Remember that you can import libraries just in .Zip, while sketches can be in .Zip, .Ino or .Pde

Do not show me again

CANCEL

IMPORT





EDITOR

Sketchbook

Examples

Libraries

Monitor

Help

Preferences

NEW SKETCH



sketch_decl0a



-- Select Board or Port --



SHARE

sketch_decl0a.ino

ReadMe.adoc



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

SEARCH SKETCHBOOK



ORDER BY NAME



01_blinking_LED

03_two_LEDs

04_RGB_LED

05_RGB_LED_for_loop

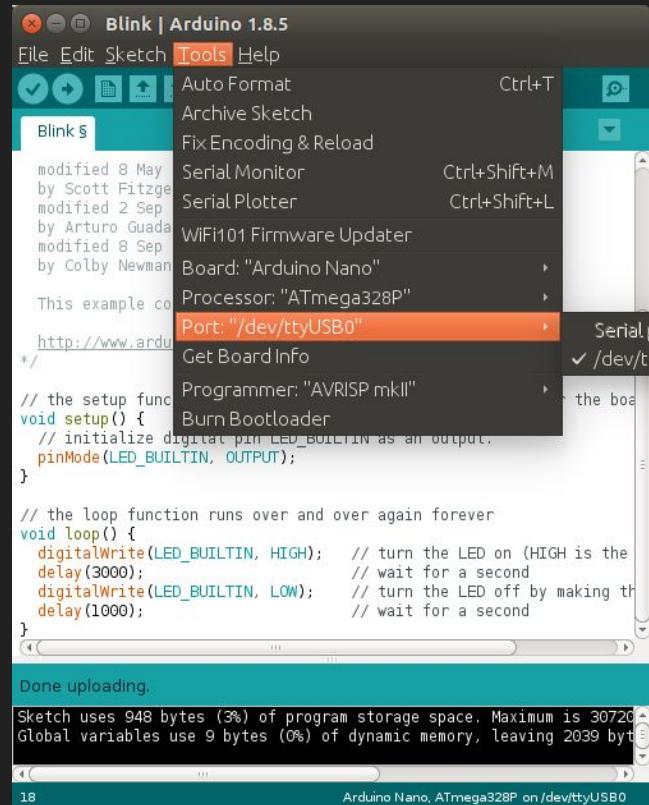
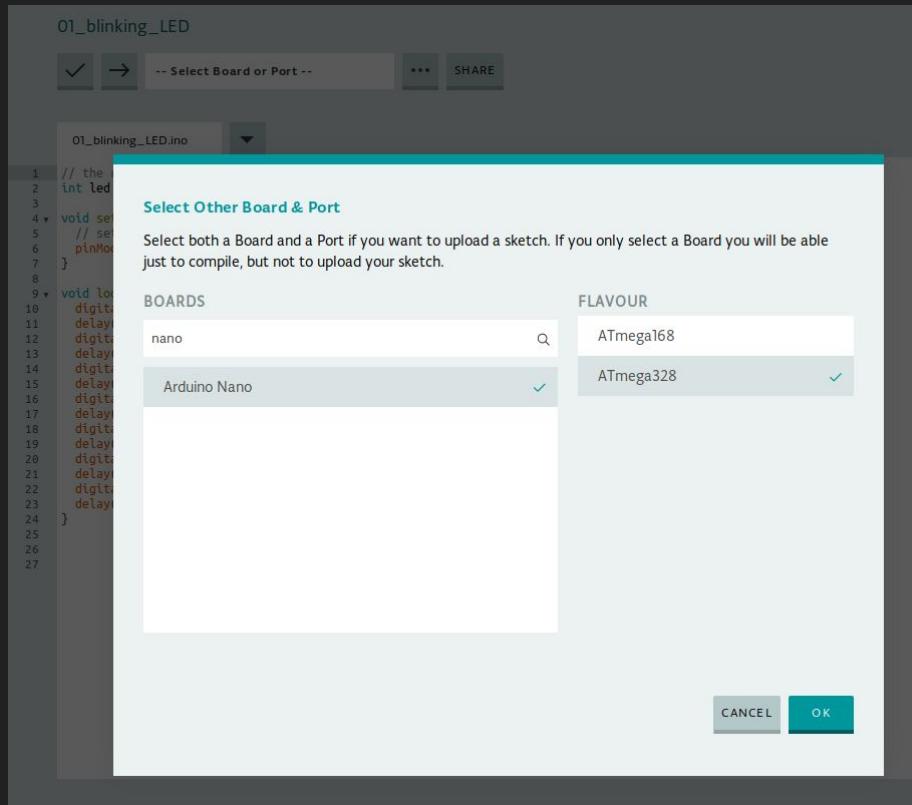
06_photo_resistor

07_photo_resistor_LED

08_photo_resistor

sketch_decl0a

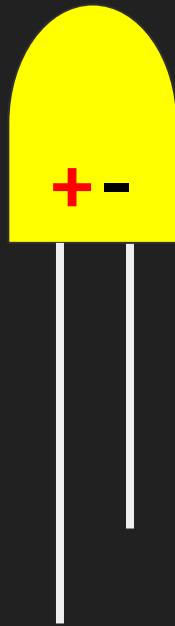
Nastavitve za Arduino v Arduino IDE



Testna ploščica



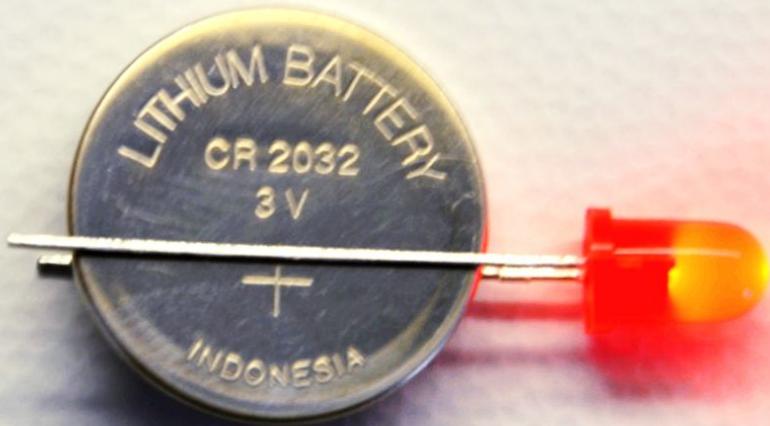
LED lučke



LED



RGB LED



LITHIUM BATTERY
CR 2032
3V



INDONESIA

Uporniki - moč in računanje upora

Upornik se upira električnemu toku in mu preprečuje, da bi nemoteno tekel skozenj.

$$\text{Napetost (V)} = \text{Tok(I)} * \text{Upor(R)}$$

Upor: 220 Ohmov

Napetost: 5 Voltov

Tok: 23 mA (mili amperov)

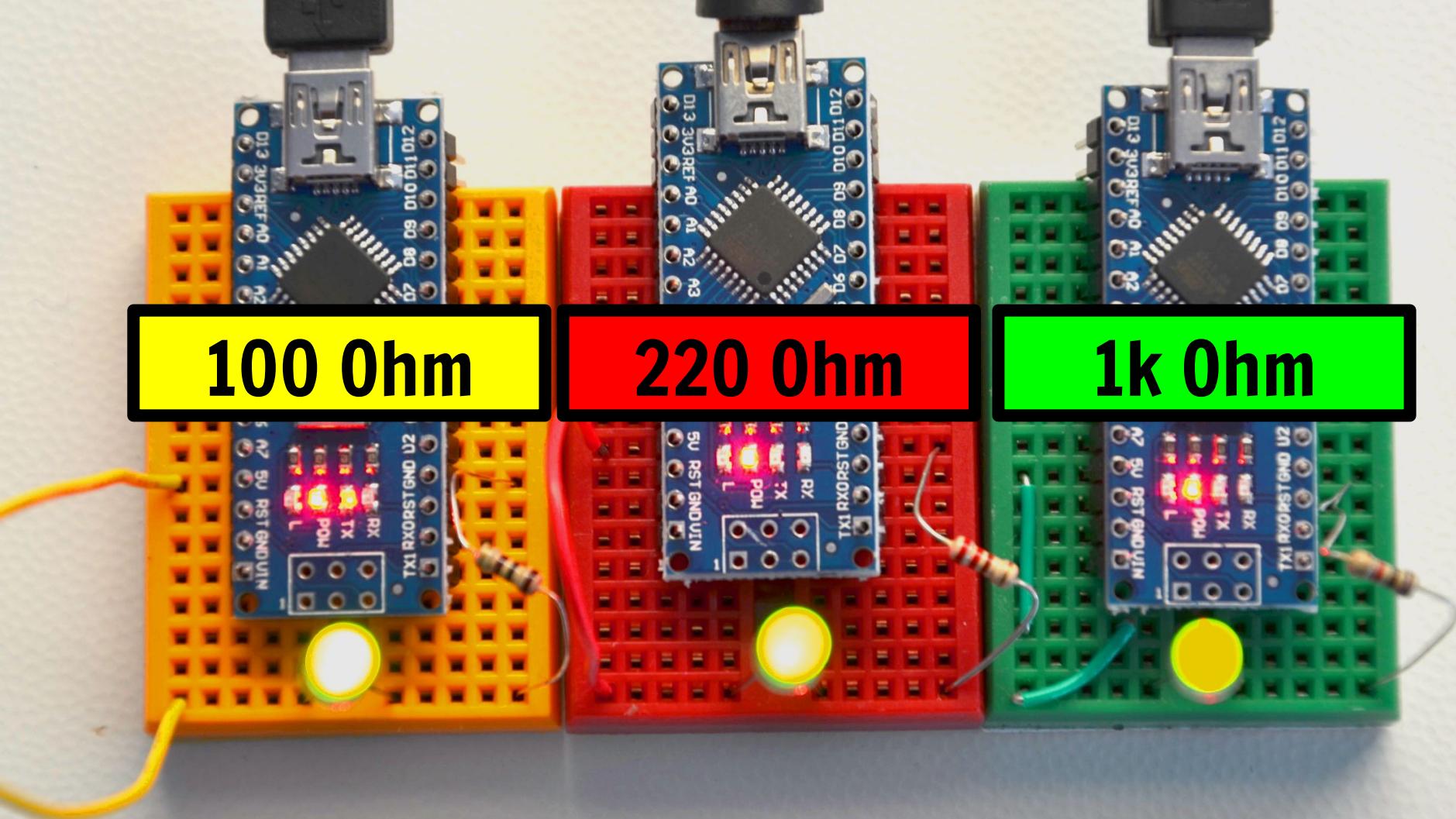
$$5V = 23 \text{ mA} * 220 \text{ Oh}$$

$$5V = 0.0227A * 220 \text{ Oh}$$

100 Ohm

220 Ohm

1k Ohm

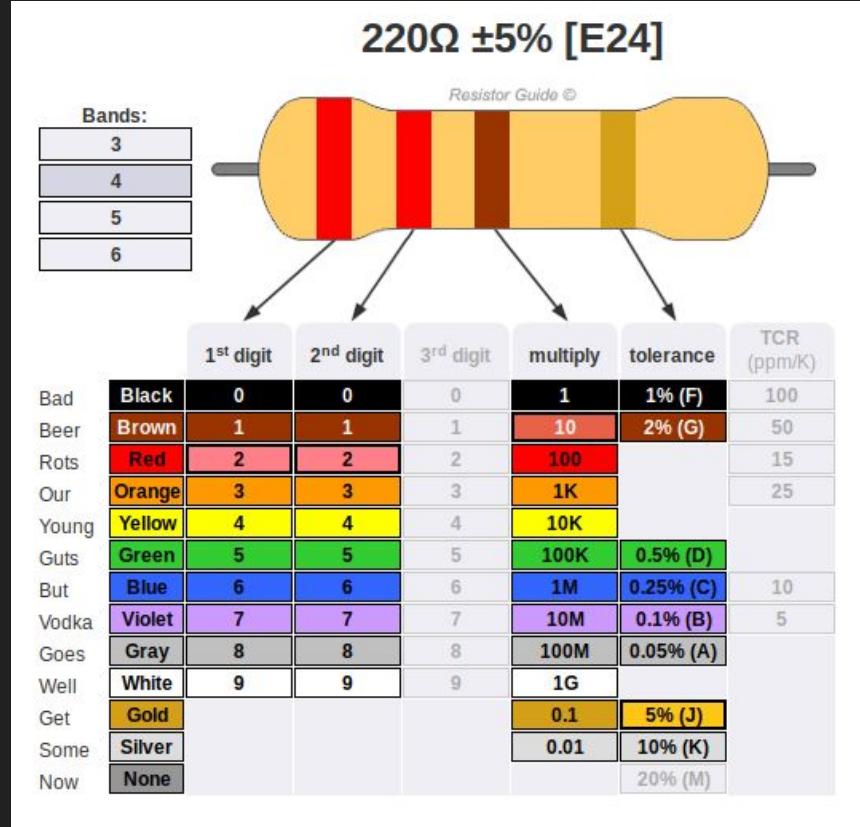


$100\Omega \pm 5\% [E24]$

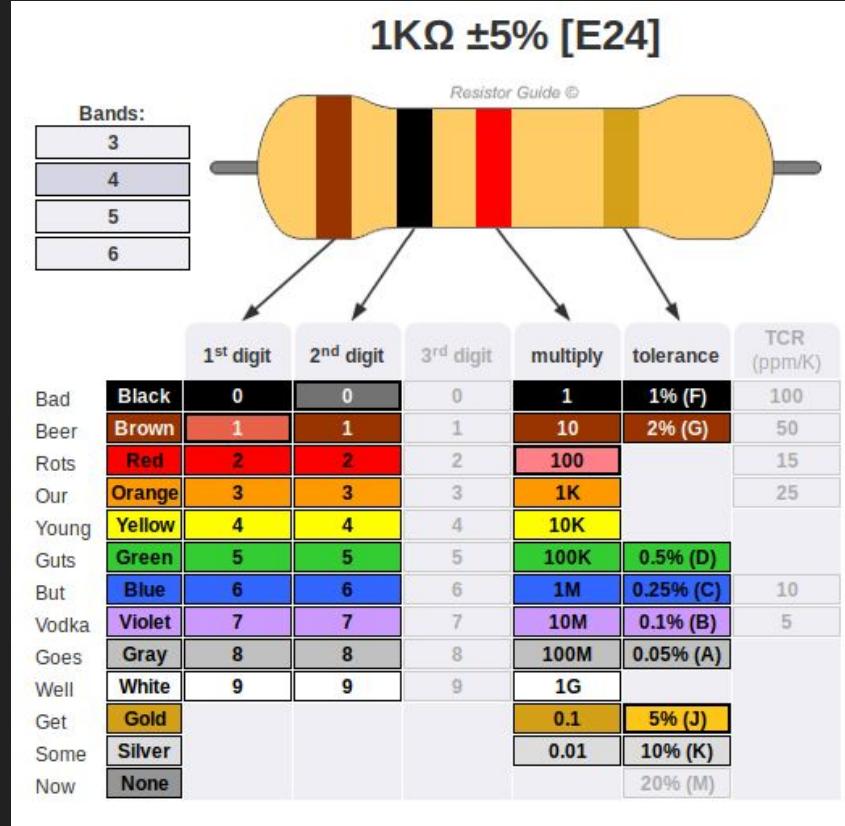
Resistor Guide ©

Bands:	1 st digit	2 nd digit	3 rd digit	multiply	tolerance	TCR (ppm/K)	
Bad	Black	0	0	0	1	1% (F)	100
Beer	Brown	1	1	1	10	2% (G)	50
Rots	Red	2	2	2	100		15
Our	Orange	3	3	3	1K		25
Young	Yellow	4	4	4	10K		
Guts	Green	5	5	5	100K	0.5% (D)	
But	Blue	6	6	6	1M	0.25% (C)	10
Vodka	Violet	7	7	7	10M	0.1% (B)	5
Goes	Gray	8	8	8	100M	0.05% (A)	
Well	White	9	9	9	1G		
Get	Gold				0.1	5% (J)	
Some	Silver				0.01	10% (K)	
Now	None					20% (M)	

<http://www.resistorguide.com/resistor-color-code-calculator/>



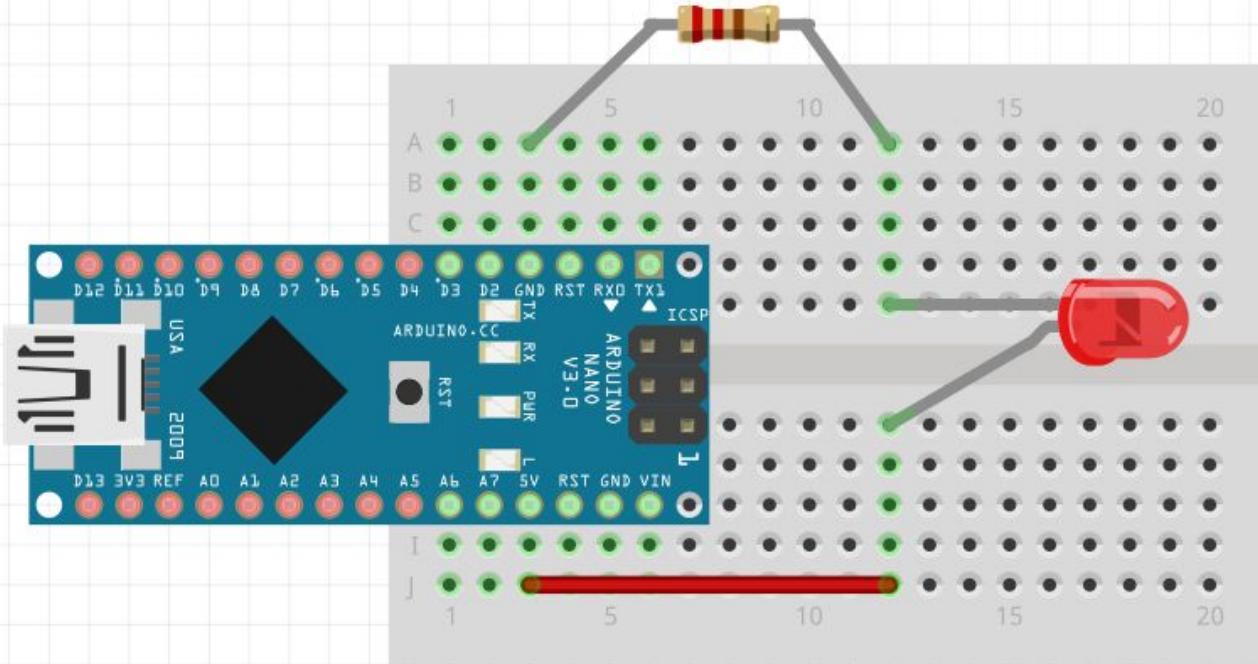
<http://www.resistorguide.com/resistor-color-code-calculator/>



<http://www.resistorguide.com/resistor-color-code-calculator/>

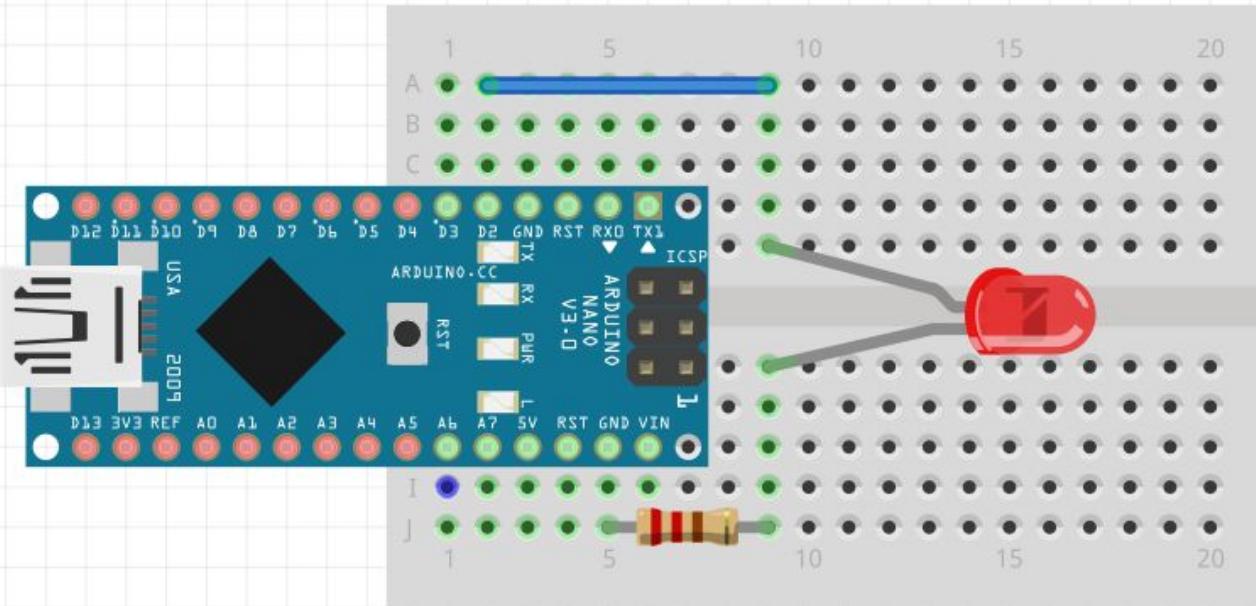
[00]

Lučka brez programa



[01]

Utricanje lučke



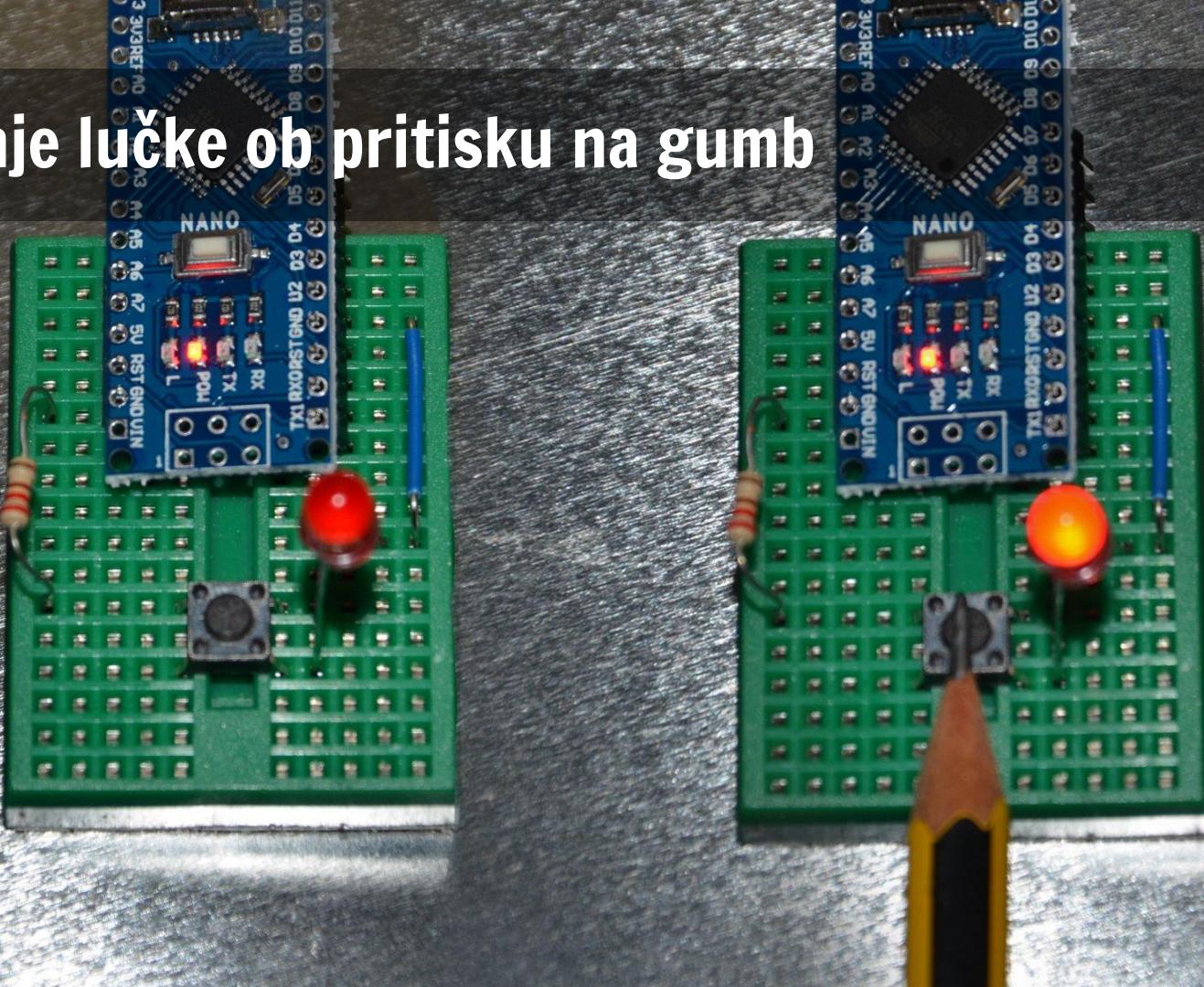
```
int LED = 2;

void setup() {
    pinMode(LED, OUTPUT);
}

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

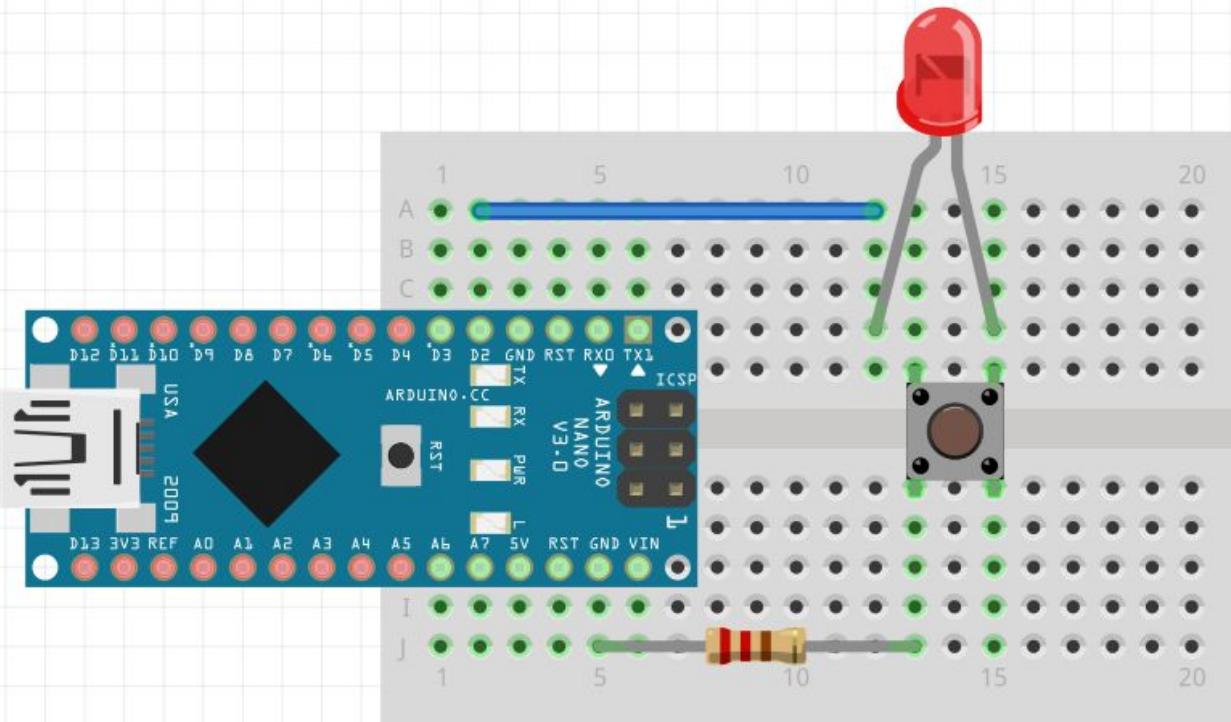
Utripanje lučke ob pritisku na gumb

[02]



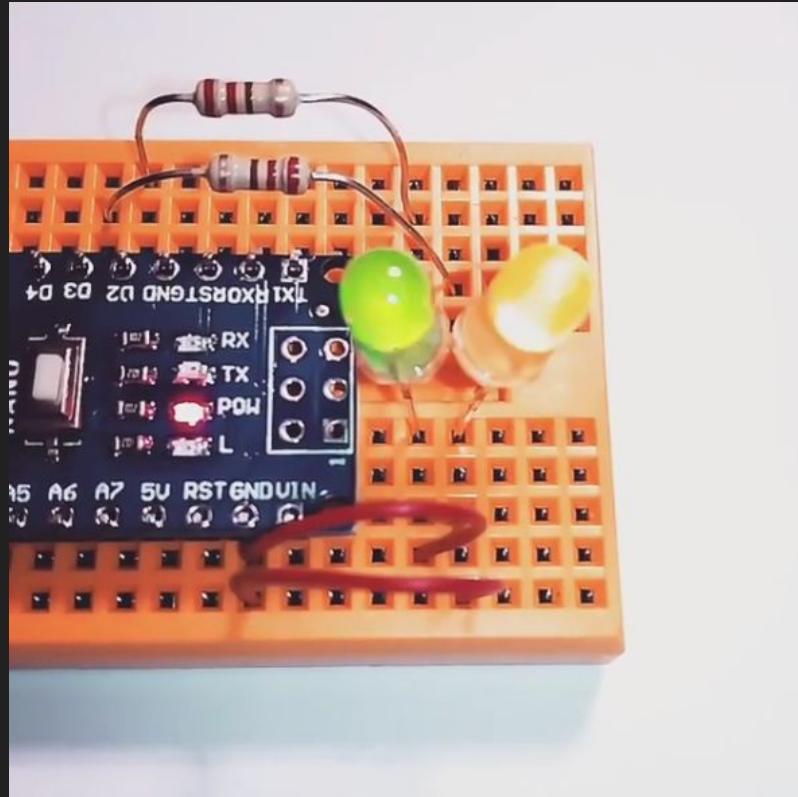
[02]

Utripanje lučke ob pritisku na gumb

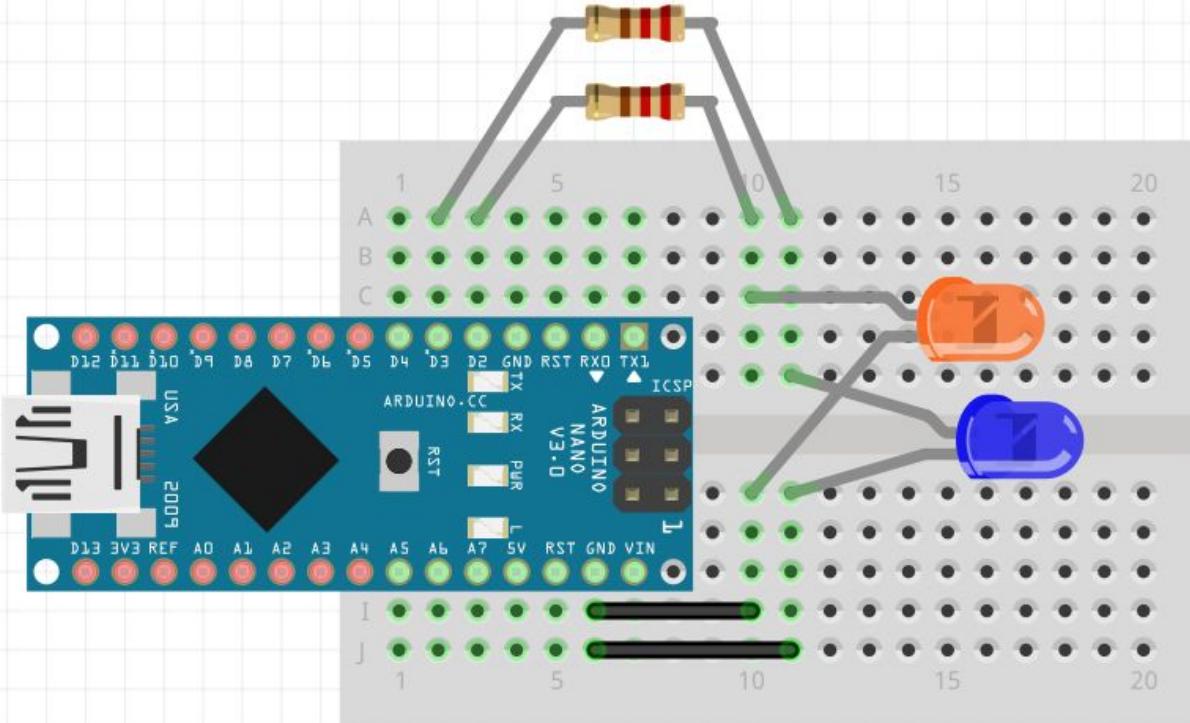


[03]

Izmenično utripanje LED lučk



Izmenično utripanje



```
int led_one = 2;
int led_two = 3;

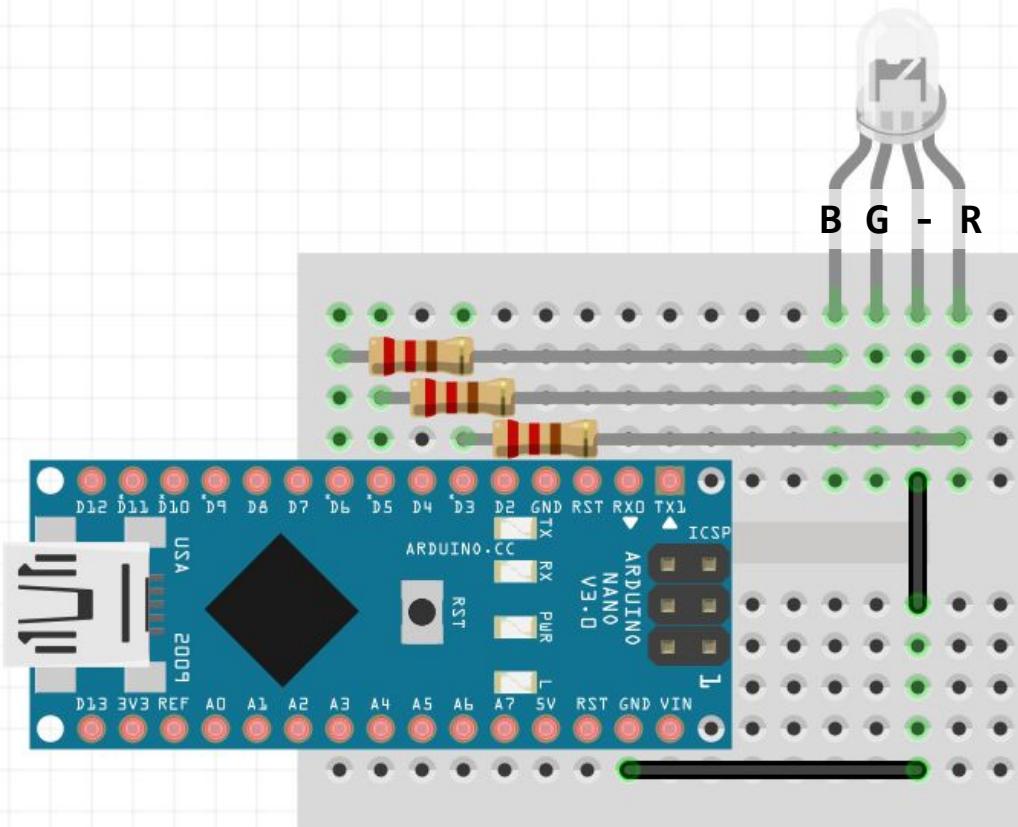
void setup() {
    // set up LED as OUTPUT
    pinMode(led_one, OUTPUT);
    pinMode(led_two, OUTPUT);
}

void loop() {
    digitalWrite(led_one, HIGH);
    digitalWrite(led_two, LOW);
    delay(500); // wait 0.5 second
    digitalWrite(led_one, LOW);
    digitalWrite(led_two, HIGH);
    delay(500); // wait 0.5 second
}
```

[04]
[05]

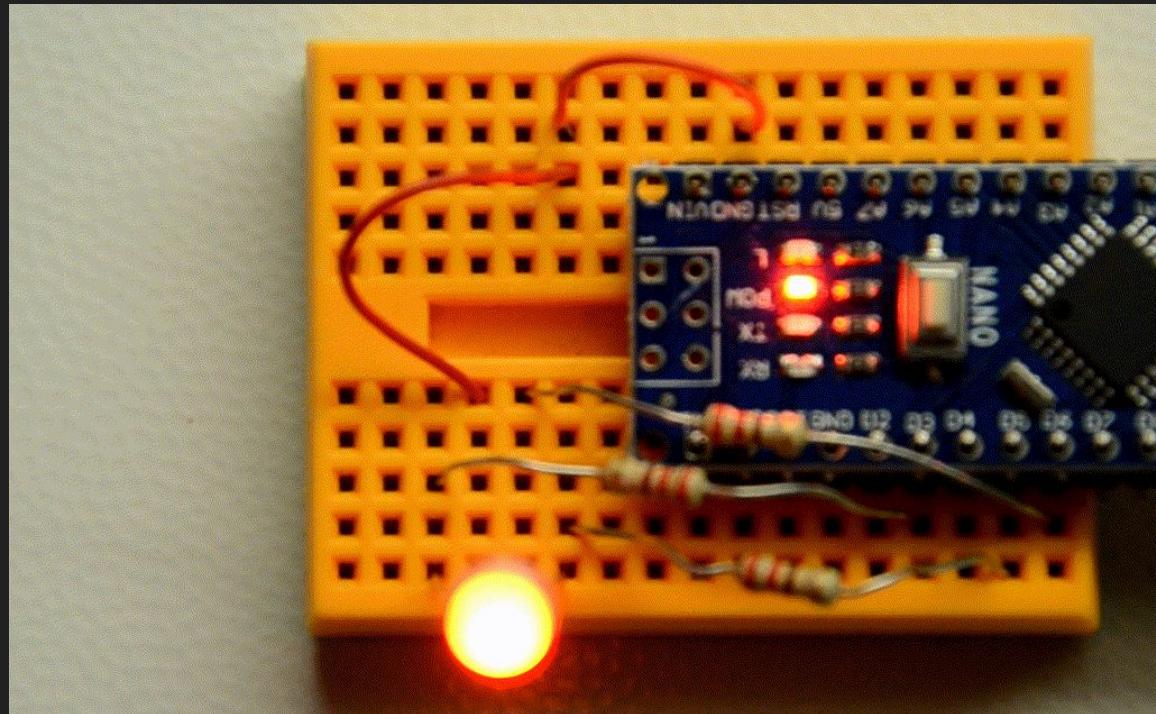
RGB lučka

PWM pins *



[04]

RGB lučka menja barve



```
int redPin = 3;
int greenPin = 6;
int bluePin = 5;

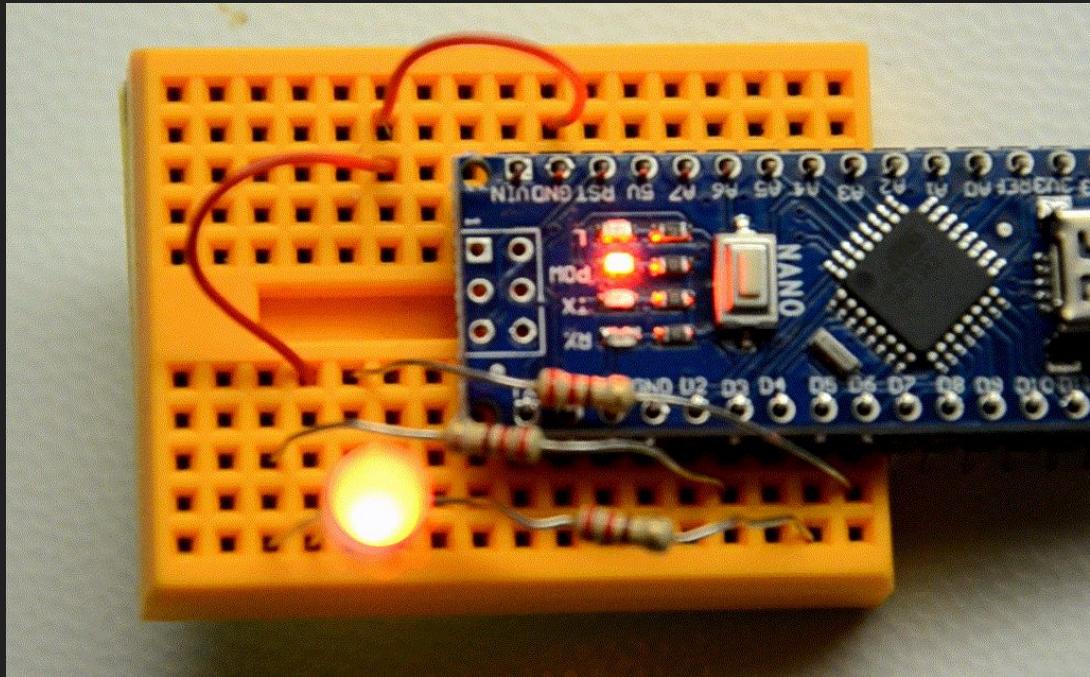
void setup() {
    pinMode(redPin, OUTPUT);
    pinMode(greenPin, OUTPUT);
    pinMode(bluePin, OUTPUT);
}

void loop()
{
    setColor(255, 0, 0); // red
    delay(3000);
    setColor(0, 255, 0); // green
    delay(2000);
    setColor(0, 0, 255); // blue
    delay(1000);
}

void setColor(int red, int green, int blue)
{
    analogWrite(redPin, red);
    analogWrite(greenPin, green);
    analogWrite(bluePin, blue);
}
```

[05]

RGB lučka pulzajoče barve



```
int redPin = 3;
int greenPin = 5;
int bluePin = 6;
int i;

void setup() {
    // set up OUTPUTS
    pinMode(redPin, OUTPUT);
    pinMode(greenPin, OUTPUT);
    pinMode(bluePin, OUTPUT);
}

void loop()
{
    for (i=0; i<=255; i++) {
        analogWrite(redPin, i);
        analogWrite(greenPin, 0);
        analogWrite(bluePin, 0);
        delay(5);
    }
}
```

Senzorji in serial port



The image shows the Arduino IDE interface with a sketch named "06_photo_resistor". The code reads an analog value from pin A7 and prints it to the serial port:

```
int sensorPin = A7;  
void setup() {  
  Serial.begin(9600);  
}  
void loop() {  
  int sensorValue = analogRead(sensorPin);  
  Serial.println(sensorValue);  
}
```

A breadboard in the foreground has a red LED connected to digital pin 13. A photoresistor is connected between digital pin 2 and ground. A 10k pull-down resistor is also connected between pin 2 and ground.

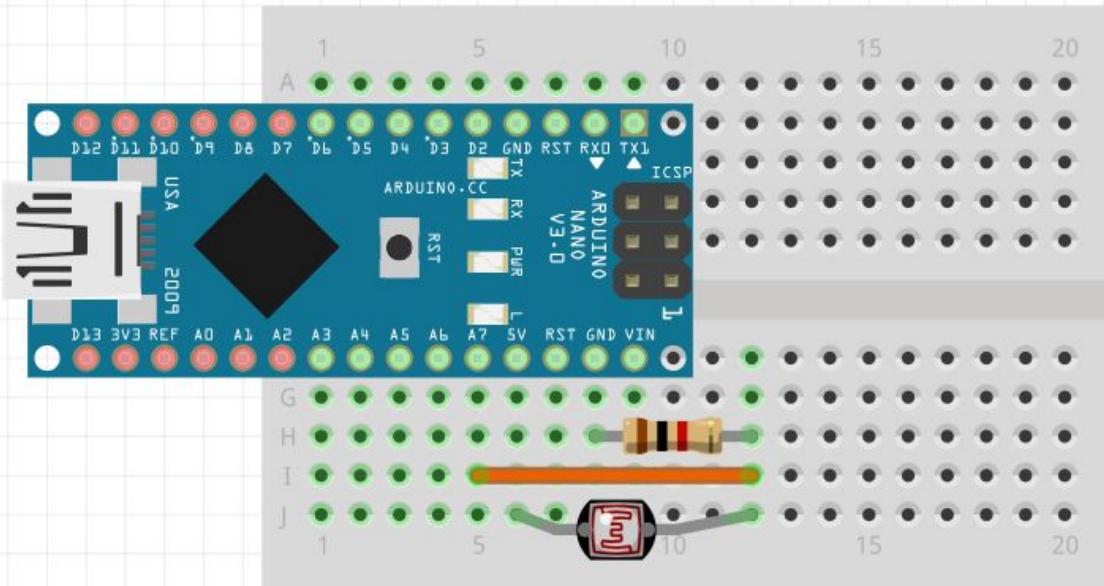


The serial monitor window shows a continuous stream of data, with the "Autoscroll" checkbox checked. The data consists of the value read from the photoresistor, which fluctuates between 882 and 902. The baud rate is set to 9600.

882
883
889
899
892
894
896
898
899
899
900
901
902
902
902

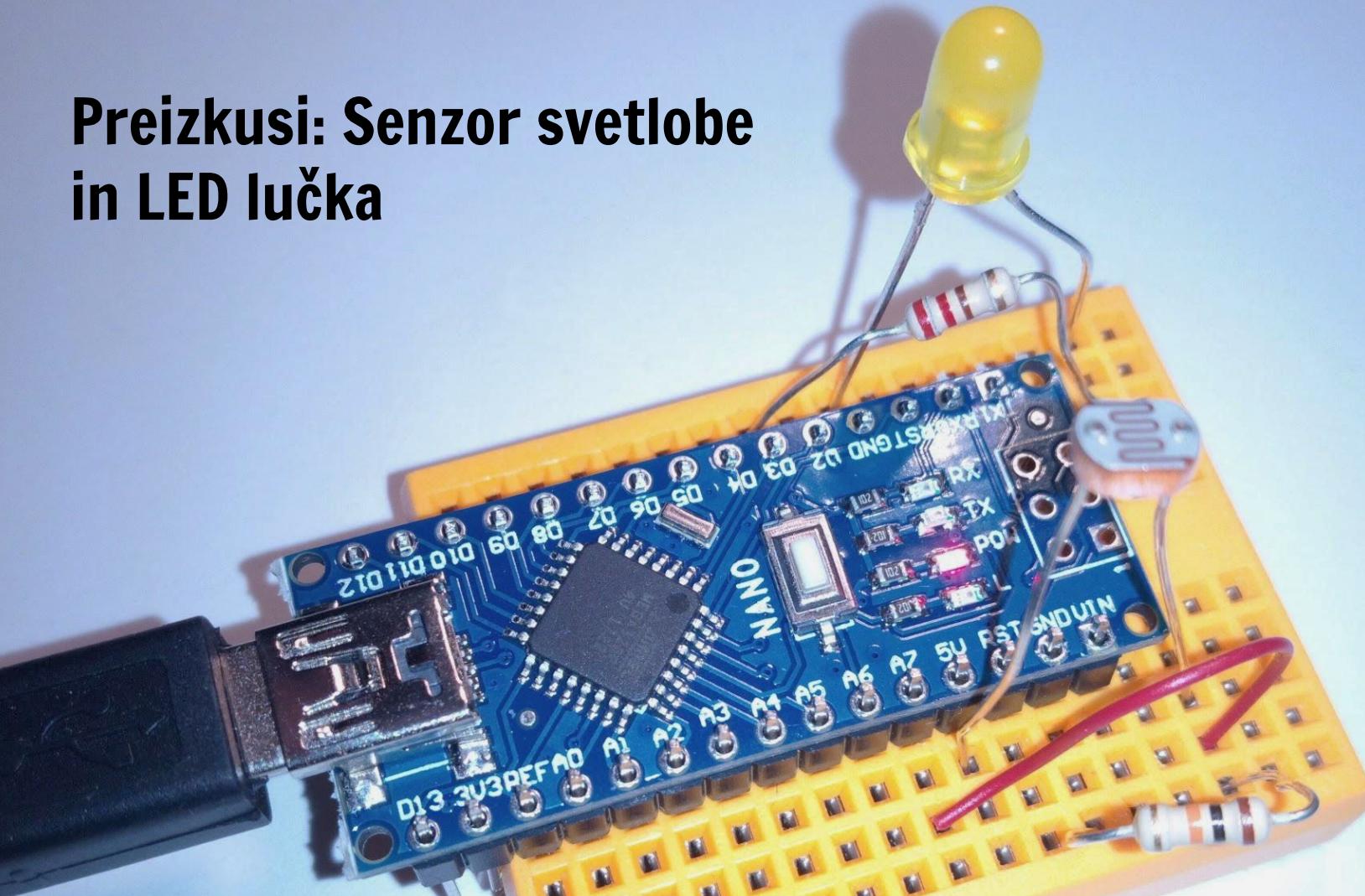
[06]

Foto-upornik & Serial port



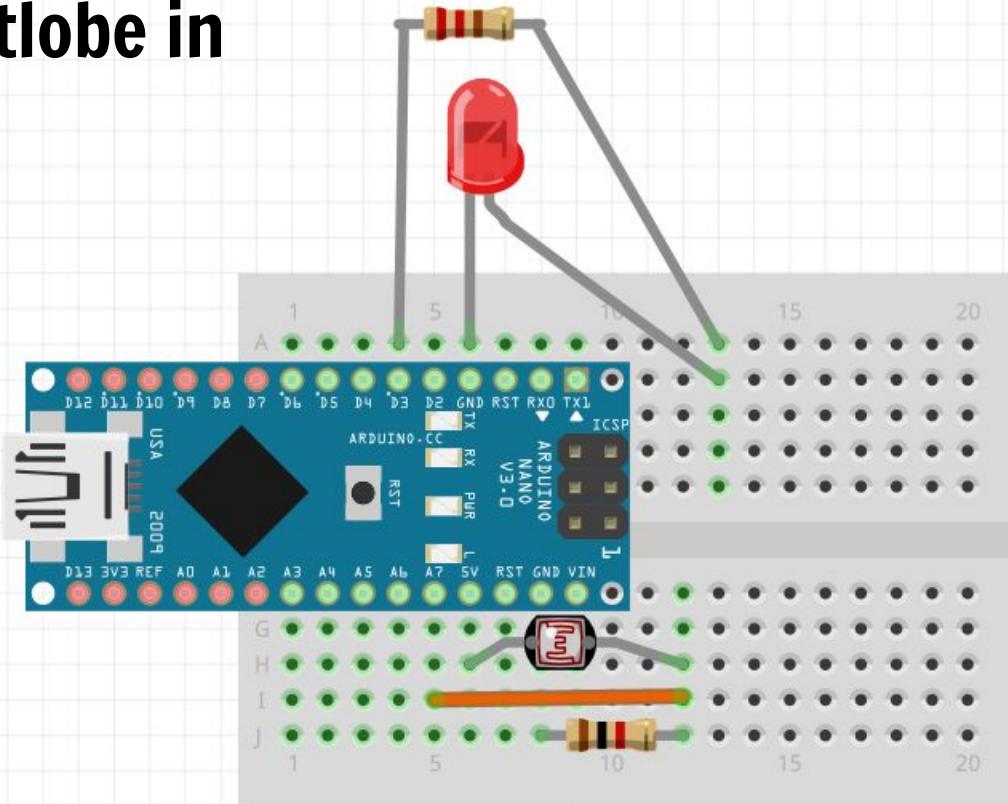
[07]

Preizkusi: Senzor svetlobe in LED lučka



Senzor svetlobe in LED lučka

[07]



```
int sensorPin = A7;  
int led = 3;  
int input;  
int output;  
  
void setup() {  
    Serial.begin(9600);  
}  
  
void loop()  
{  
    input = analogRead(sensorPin);  
    output = input / 4;  
    delay(1000);  
    analogWrite(led, output);  
    Serial.print( input );  
    Serial.print( " - " );  
    Serial.println( output );  
}
```

Processing Demo

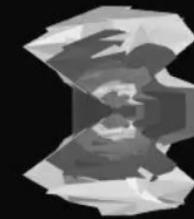
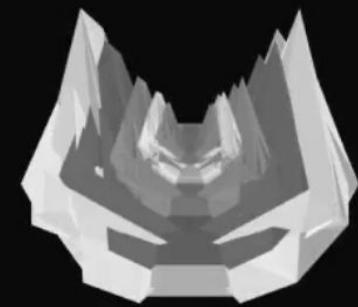
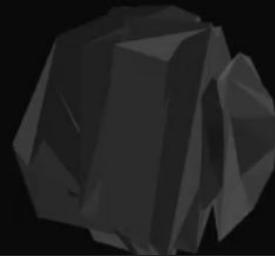
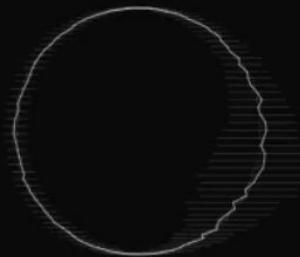
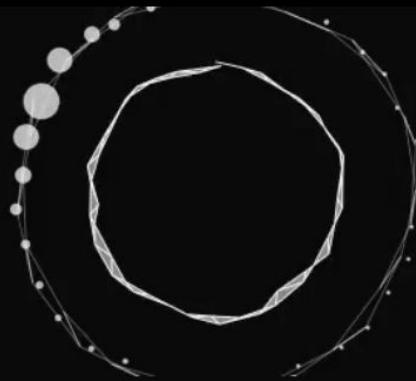
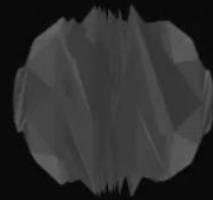
Processing

[Cover](#)[Download](#)[Donate](#)[Exhibition](#)[Reference](#)[Libraries](#)[Tools](#)[Environment](#)[Tutorials](#)[Examples](#)[Books](#)

3.3.6 (4 September 2017)

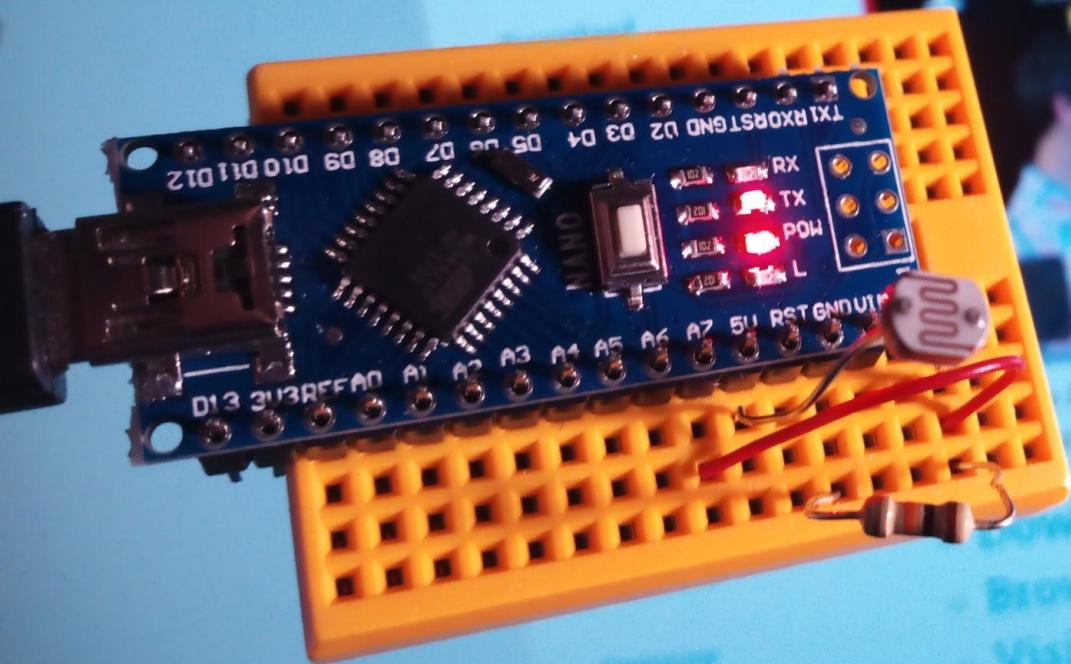
[Windows 64-bit](#)[Windows 32-bit](#)[Linux 64-bit](#)[Linux 32-bit](#)[Mac OS X](#)[Linux ARMv6hf](#)[» Github](#)Read about the [changes in 3.0](#). The [list of revisions](#) covers the differences

<https://processing.org/download/>



[08]

Processing

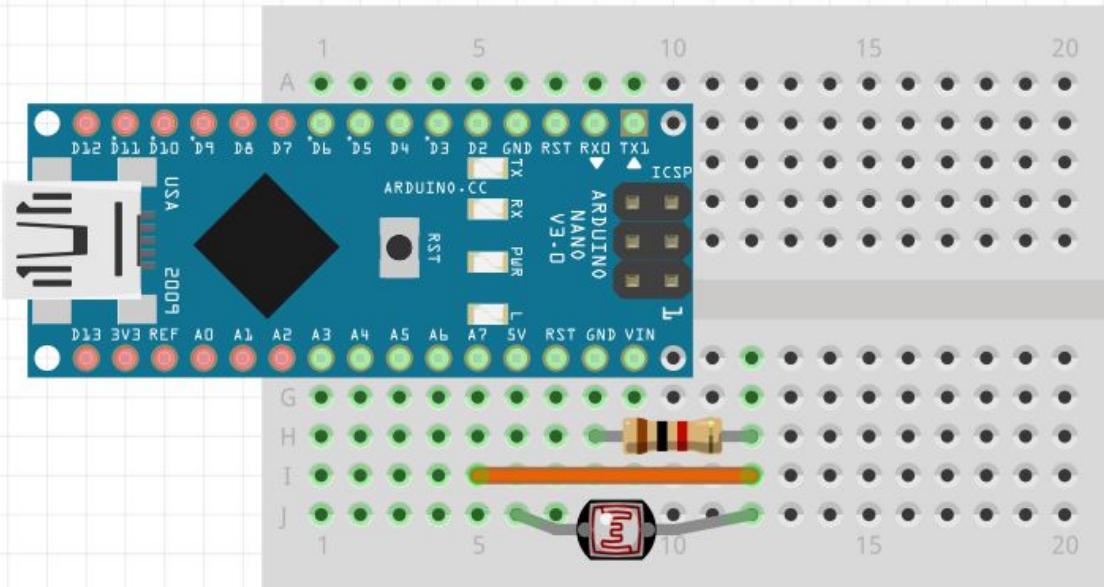


Processing is a programming language and environment for�
learning how to code within the context of the visual arts, design, and
technology. It is used by millions of people around the world to
explore art, science, and engineering through computation and
visual expression.

Download Processing
Browse Tutorials
Visit the Reference

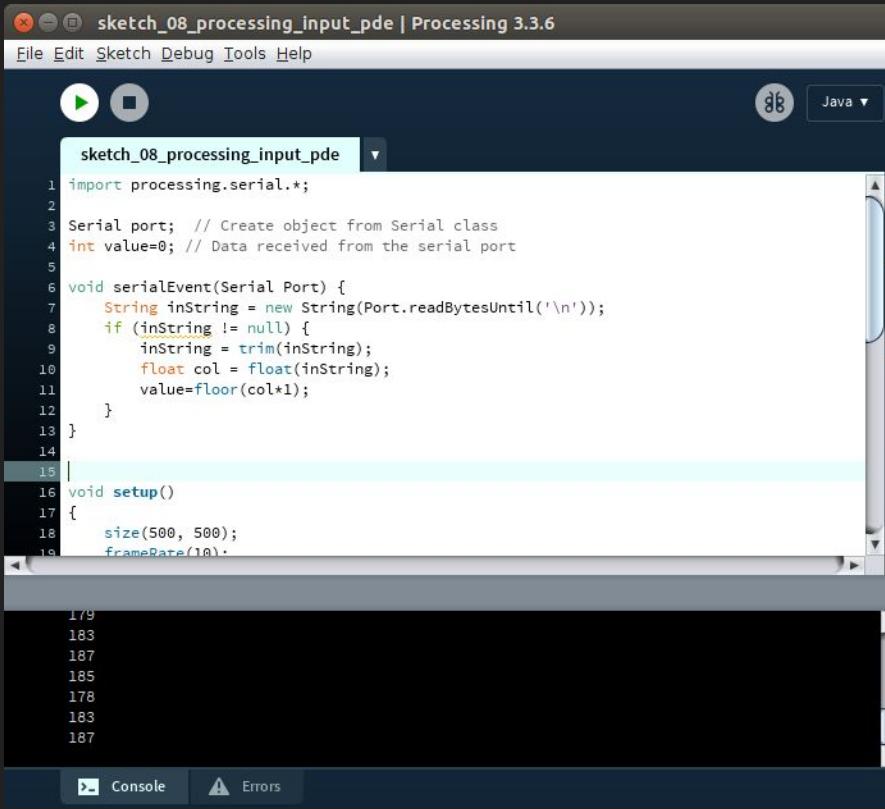
[08]

Foto-upornik & Serial port



```
int sensorPin = A7;  
int input;  
int output;  
  
void setup() {  
    Serial.begin(9600);  
}  
  
void loop()  
{  
    input = analogRead(sensorPin);  
    output = input / 4;  
    delay(10);  
    Serial.println( output );  
}
```

Processing sketch



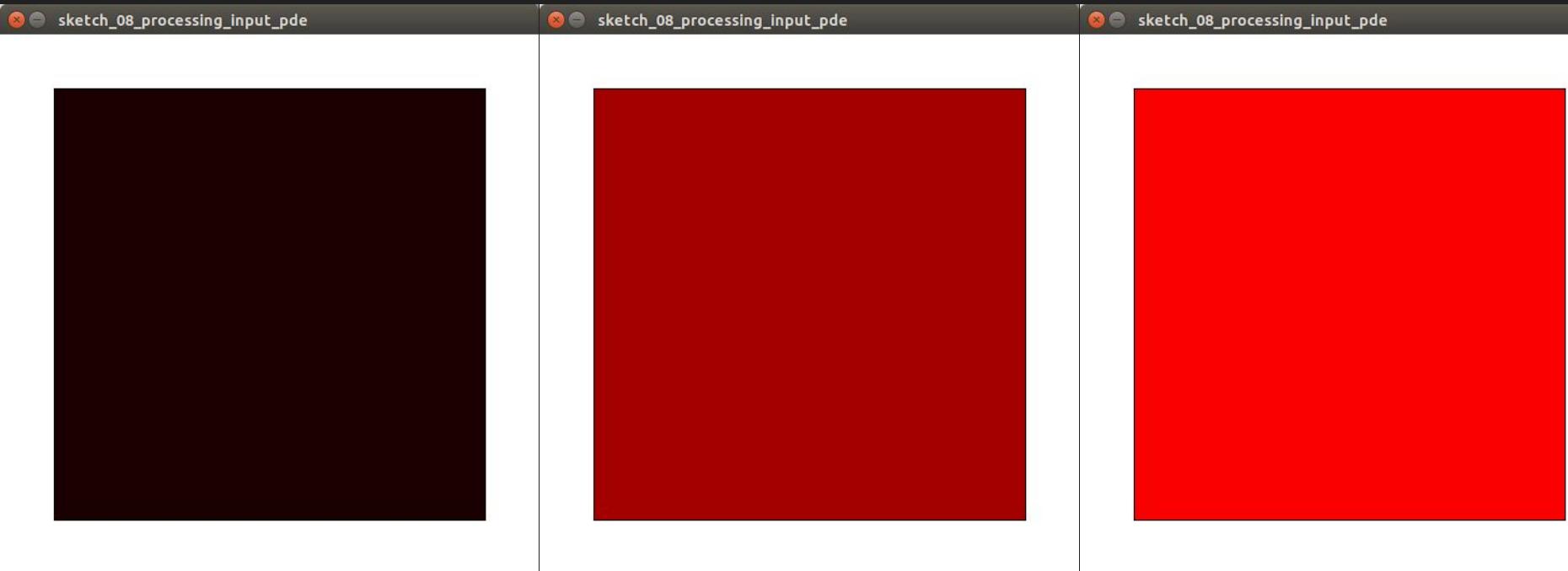
The screenshot shows the Processing 3.3.6 IDE interface. The title bar reads "sketch_08_processing_input_pde | Processing 3.3.6". The menu bar includes File, Edit, Sketch, Debug, Tools, and Help. Below the menu is a toolbar with a play button, a square button, and a Java dropdown set to "Java". The main area displays the sketch code. The code imports the Serial class and defines a variable value. It contains a serialEvent function to handle incoming data from a serial port and a setup function to initialize the window size and frame rate.

```
sketch_08_processing_input_pde
import processing.serial.*;
Serial port; // Create object from Serial class
int value=0; // Data received from the serial port
void serialEvent(Serial Port) {
    String inString = new String(Port.readBytesUntil('\n'));
    if (inString != null) {
        inString = trim(inString);
        float col = float(inString);
        value=floor(col*1);
    }
}
void setup()
{
    size(500, 500);
    frameRate(10);
}
```

The bottom console and errors panes are empty.

[08]

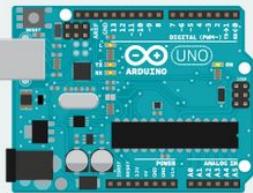
Processing - Resultat



Povezave



WHAT IS ARDUINO?

[BUY AN ARDUINO](#)[LEARN ARDUINO](#)[DONATE](#)[BLOG](#)

THE IMPERIALIZER MAKES
QUICK WORK OF METRIC
CONVERSIONS



REDEFINING THE
LEARNING EXPERIENCE
ONE CLASSROOM
AT A TIME

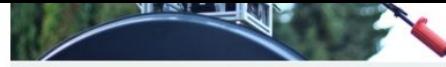
[BLOG](#)

ARDUINO MKR ZERO,
THE POWER OF THE
ZERO IN A SMALLER



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

and access cool tutorials!





PROJECT HUB

MY DASHBOARD

NEW PROJECT

SEARCH PROJECTS



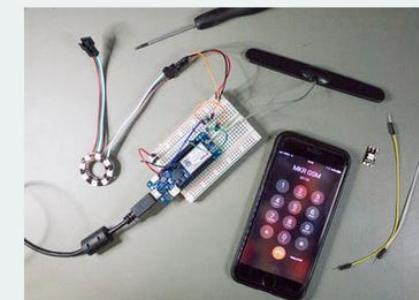
All products ▾

All categories ▾

Trending ▾

Any difficulty ▾

Any type ▾



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

77 VIEWS 0 COMMENTS 1 RESPECT

1,916 VIEWS 0 COMMENTS 4 RESPECTS

1,596 VIEWS 0 COMMENTS 7 RESPECTS

The screenshot shows the Arduino IDE's online editor interface. On the left, a sidebar lists categories like Sketchbook, Examples, Libraries, Monitor, Help, and Preferences. The Examples section is selected, showing examples categorized by type: 01.BASICS (6), 02.DIGITAL (9), 03.ANALOG (6), 04.COMMUNICATION (12), 05.CONTROL (6), and 06.SENSORS (4). The 'Blink' example under '01.BASICS' is selected and displayed in the main code editor. The code is as follows:

```
1  /*
2   * Blink
3   *
4   * Turns an LED on for one second, then off for one second, repeatedly.
5   *
6   * Most Arduinos have an on-board LED you can control. On the UNO, MEGA and ZERO
7   * it is attached to digital pin 13, on MKR1000 on pin 6. LED_BUILTIN is set to
8   * the correct LED pin independent of which board is used.
9   * If you want to know what pin the on-board LED is connected to on your Arduino
10  model, check the Technical Specs of your board at:
11  https://www.arduino.cc/en/Main/Products
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 the public domain.
21
22  http://www.arduino.cc/en/Tutorial/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 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
```

The top right of the editor shows a yellow bar with the message "No Plugin Connection. Uploading is disabled until you reconnect." and a "HELP" button. The title bar says "Blink".

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



Simulator time: 00:00:15

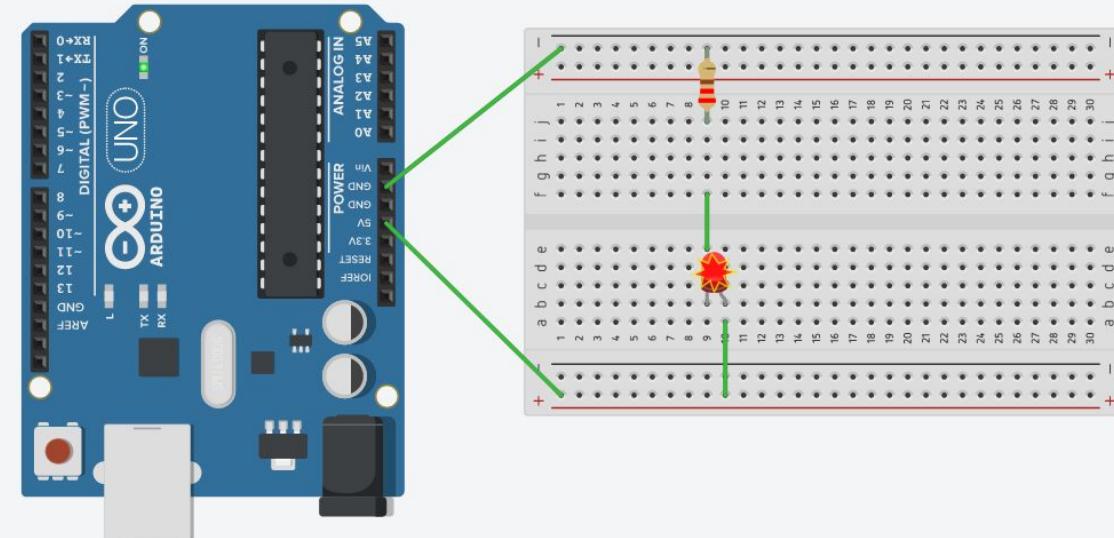
Code Editor

Components

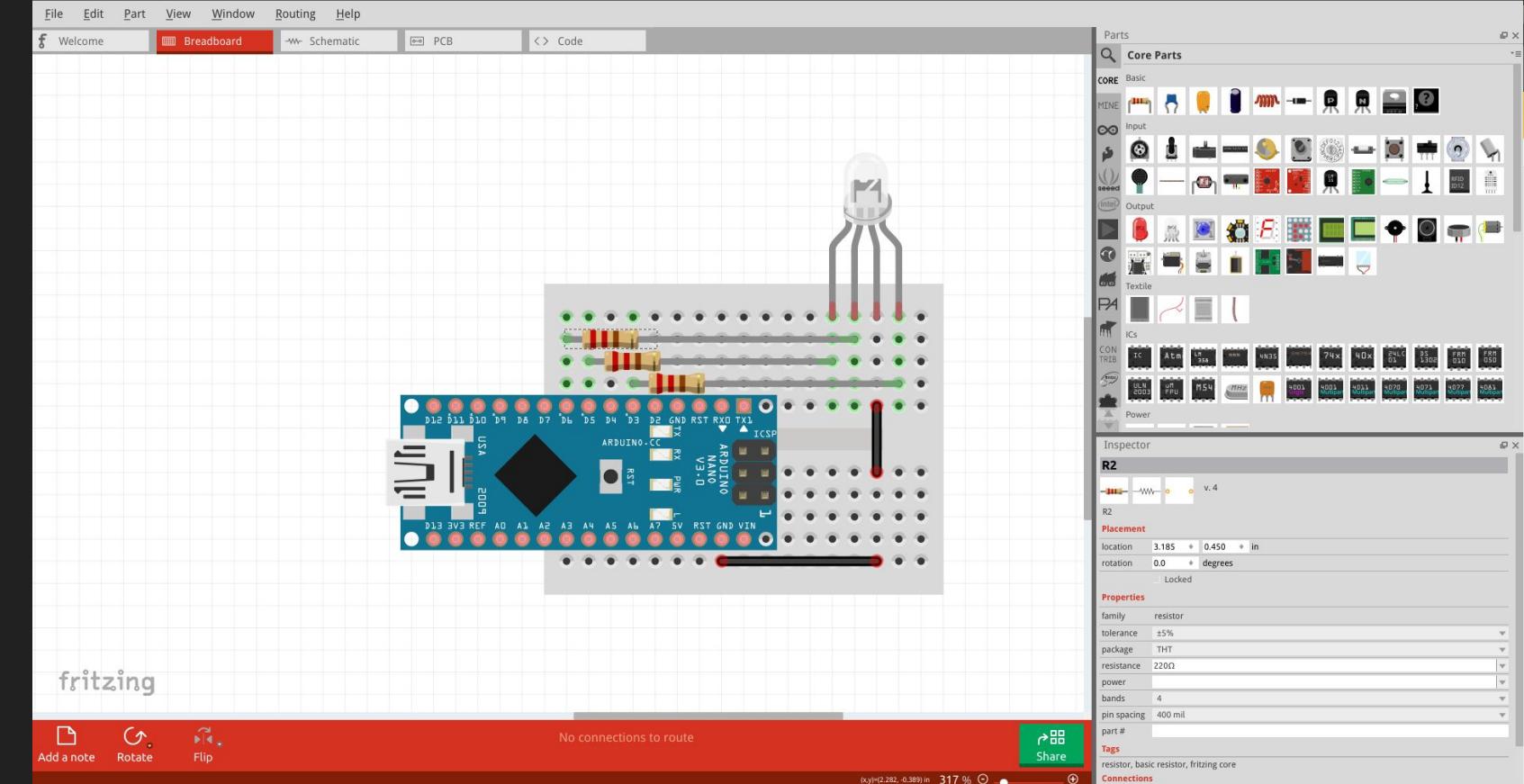
Stop Simulation

Export

Share



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



<http://fritzing.org/download/>

Išči po celotni trgovini...

[AKCIJA](#) [NOVO](#) [ARDUINO/GENUINO](#) [SBC](#) [SPARKFUN](#) [I/O](#) [MAKERBEAM](#) [TCT](#) [3D](#) [DRONI](#) [OSTALO](#)

SMAKSHOP : GENUINO/ARDUINO, LILYPAD, MAKERBEAM, 3D (TISKALNIKI, ABS, PLA), SPARKFUN



AKCIJA



NAROČI SE NA OBJAVE

* potrebna polja
email *

Ime in priimek *

NAROČI SE

Način dostave po pošti (Pošta Slovenije d.o.o.).

Strošek dostave je 3,55 EUR in brezplačna dostava pri naročilih nad 99,00 EUR.

Način dostave osebni prevzem na lokaciji podjetja.

Predhodno se je potrebno dogovoriti za termin prevzema.

Podpora kupcem(med 8:30-15:30, Pon - Pet)

elektronska pošta
tel: 041 3344 86

<https://smakshop.si>

149,99 € 129,99 €

299,99 € 249,99 €



All Categories

Arduino



Cart(0)

Wish

ALL DEPARTMENTS ▾

New ▾

\$0.99

Top Sellers

Extreme Deals

Community ▾

MVP 24 hrs

Staff Picks

DX » "arduino"

Category

Hobbies & Toys(3)

- R/C Toys(3)
- Other Accessories(1)
- Repair Parts and Tools(1)
- R/C Tanks(1)

Cell Phones&Accessories(1)

- Gadgets(1)
- NFC/Smart Control(1)

Automobiles & Motorcycles(2)

- Gadgets & Auto Parts(2)
- Other Gadgets(2)

Electrical & Tools(1191)

- Arduino & SCM Supplies(1155)
- Raspberry Pi(21)
- Other Accessories(104)
- Boards & Shields(294)
- Sensors(274)

arduino(1198)



Promotion Products



<http://www.dx.com/slarduino>

» Boards & Shields

» Boards & Shields

» Boards & Shields

» Displays



LANGUAGE

FUNCTIONS

VARIABLES

STRUCTURE

LIBRARIES

GLOSSARY

The Arduino Reference text is licensed under a Creative Commons Attribution-Share Alike 3.0 License.

Find anything that can be improved? [Suggest corrections and new documentation via GitHub](#).

Doubts on how to use Github? Learn everything you need to know in this [tutorial](#).

Language Reference

Arduino programming language can be divided in three main parts: structure, values (variables and constants), and functions.

FUNCTIONS

For controlling the Arduino board and performing computations.

Digital I/O

[`digitalRead\(\)`](#)[`digitalWrite\(\)`](#)[`pinMode\(\)`](#)

Analog I/O

[`analogRead\(\)`](#)[`analogReference\(\)`](#)[`analogWrite\(\)`](#)

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

Advanced I/O

[`noTone\(\)`](#)[`pulseIn\(\)`](#)