

LAB 3

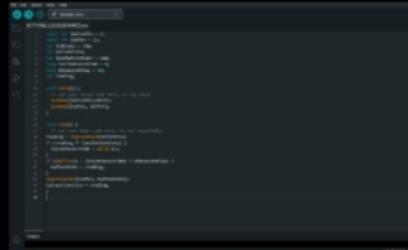
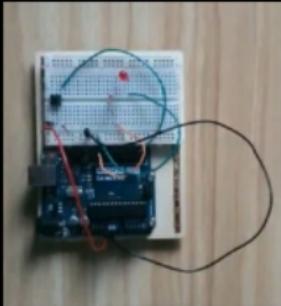
PWM, FOTORESISTORE
E POTENZIOMETRO

PWM, PHOTORESISTOR
AND POTENTIOMETER

DI KARROUM ABDERRAHIM, PARMA CHRISTIAN
E WARNAKULASOORIYA RICCARDO FERNANDO

Riassunto

IN QUESTA PRESENTAZIONE
SPIEGHEREMO COME
REALIZZARE VARI CIRCUITI
UTILIZZANDO IL
POTENZIOMETRO, IL
FOTORESISTORE E IL PWM,
UTILIZZANDO ARDUINO IDE PER
SCRIVERE I CODICI



```
void setup() {
  // put your setup code here, to run once:
  // initialize serial port:
  Serial.begin(9600);
}

void loop() {
  // put your main code here, to run repeatedly:
  // read value from potentiometer
  int value = analogRead(A0);
  // map it to 100-500
  value = map(value, 0, 1023, 100, 500);
  // print it to the serial monitor:
  Serial.println(value);
}
```

Summary

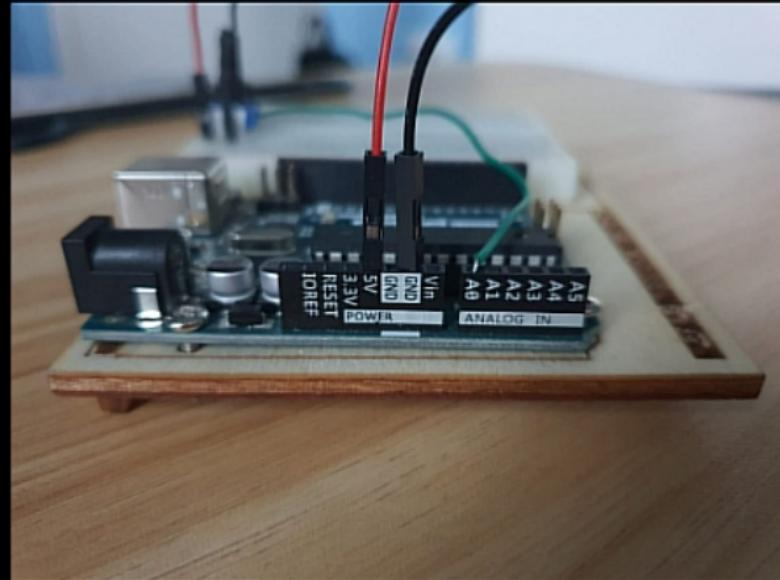
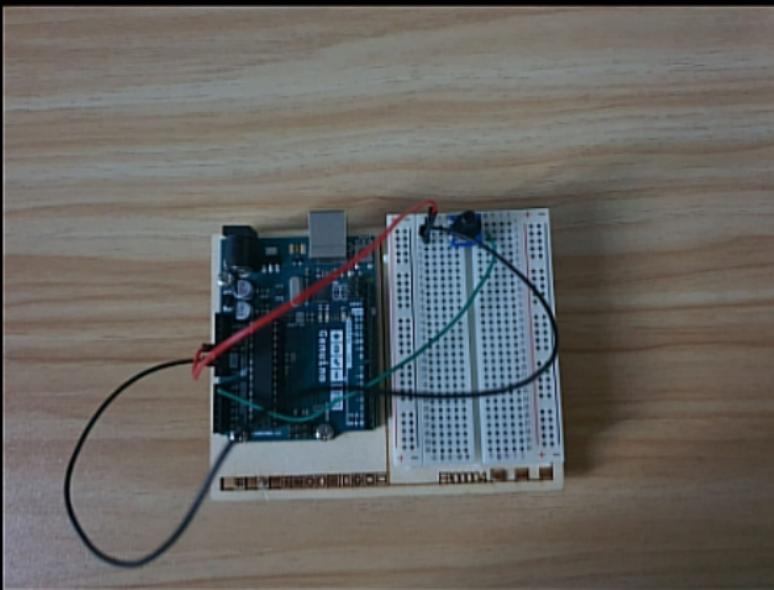
IN THIS PRESENTATION WE ARE
GOING TO EXPLAIN HOW TO
REALIZE VARIOUS ELECTRIC
CIRCUITS USING A
POTENTIOMETER, A
PHOTORESISTOR AND THE PWM,
UTILIZING ARDUINO IDE TO WRITE
THE CODES.

→ CONTENUTI / *CONTENTS*

- 01.** CIRCUITO 1 / *CIRCUIT 1*
- 02.** CIRCUITO 2 / *CIRCUIT 2*
- 03.** CIRCUITO 3 / *CIRCUIT 3*
- 04.** CIRCUITO 4 / *CIRCUIT 4*
- 05.** CIRCUITO 5 / *CIRCUIT 5*

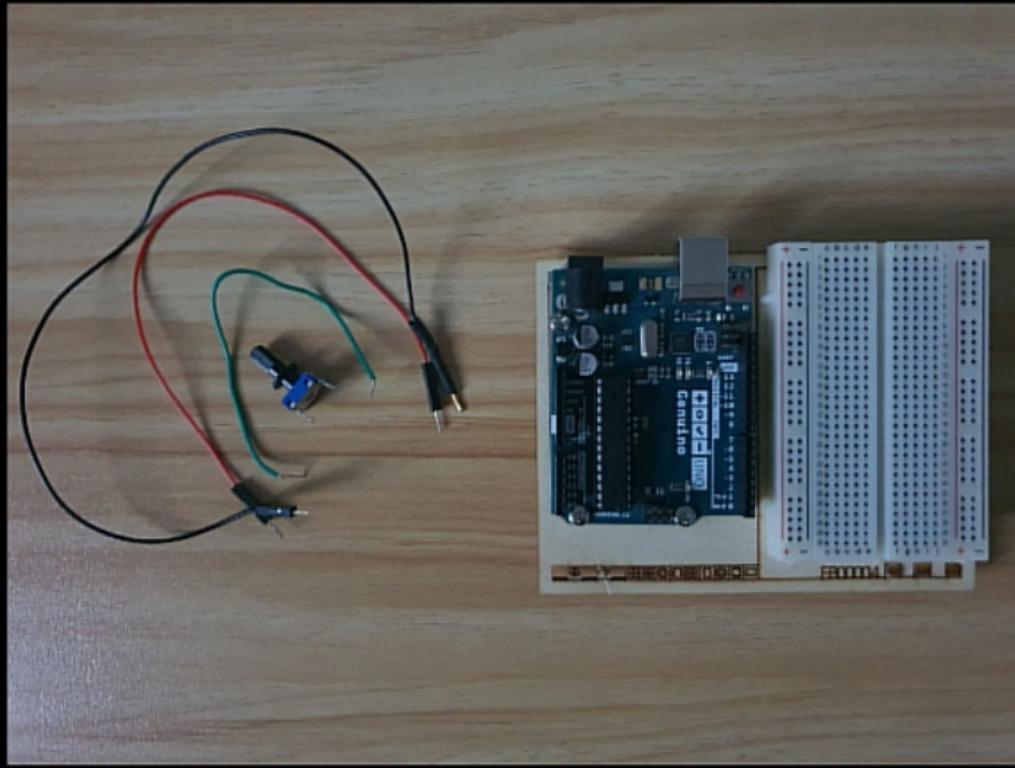
CIRCUITO 1 / CIRCUIT 1

In questo circuito vi spiegheremo come utilizzare Arduino IDE per programmare il potenziometro per leggerne il valore.



In this circuit we are going to explain to you how to use Arduino IDE in order to code the potentiometer to read its value.

CIRCUITO 1 / *CIRCUIT 1* **COMPONENTI / MATERIALS NEEDED**



Per realizzare questo circuito,
serviranno:

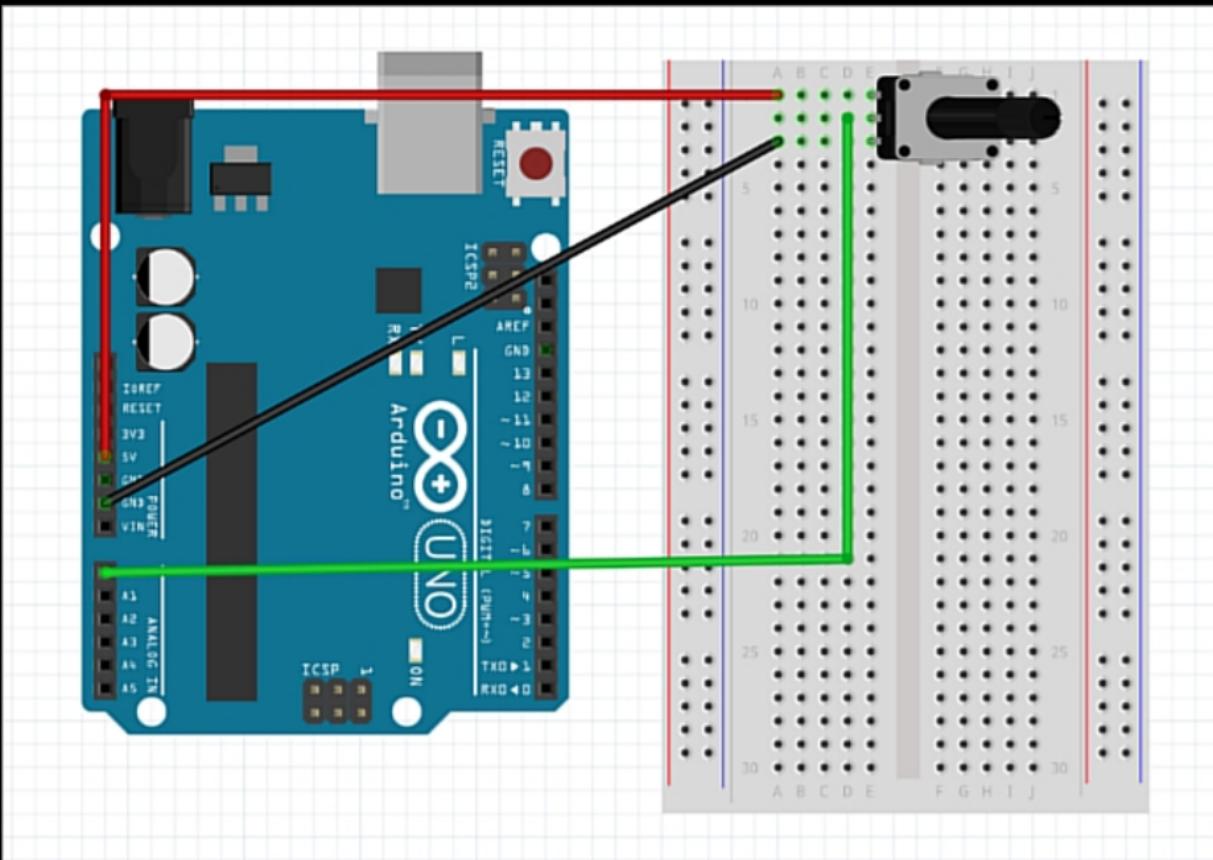
- Scheda Arduino Uno;
- Breadboard;
- 2x Cavi Jumper;
- Ponticello;
- Potenziometro;
- Arduino IDE software
(<https://www.arduino.cc/en/software>).

*In order to realize this circuit, you will
need:*

- *Arduino Uno board;*
- *Breadboard;*
- *2x Jumper Cables;*
- *Cable;*
- *Potentiometer;*
- *Arduino IDE software*
(<https://www.arduino.cc/en/software>).

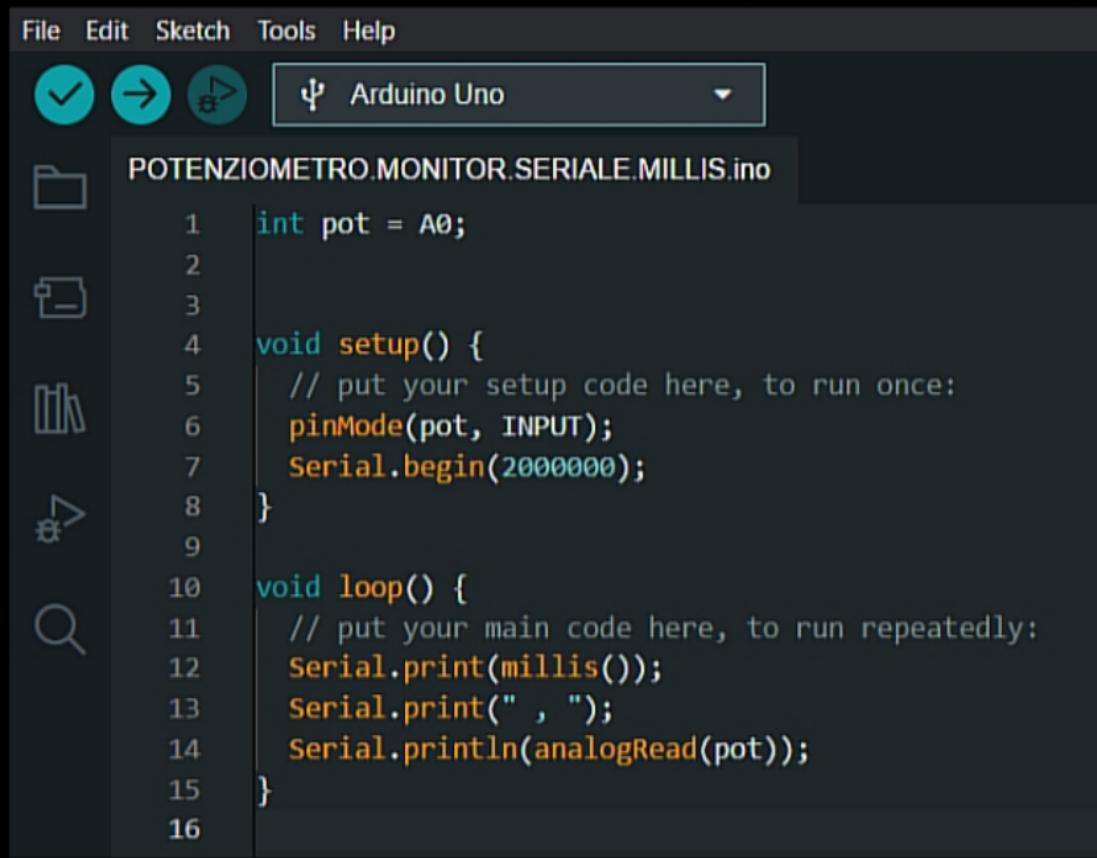
CIRCUITO 1 / CIRCUIT 1

SCHEMA FRITZING / FRITZING SCHEME



CODICE LETTURA PIN ANALOGICO CON MILLIS

CODE FOR THE READING OF THE ANALOG PIN WITH MILLIS



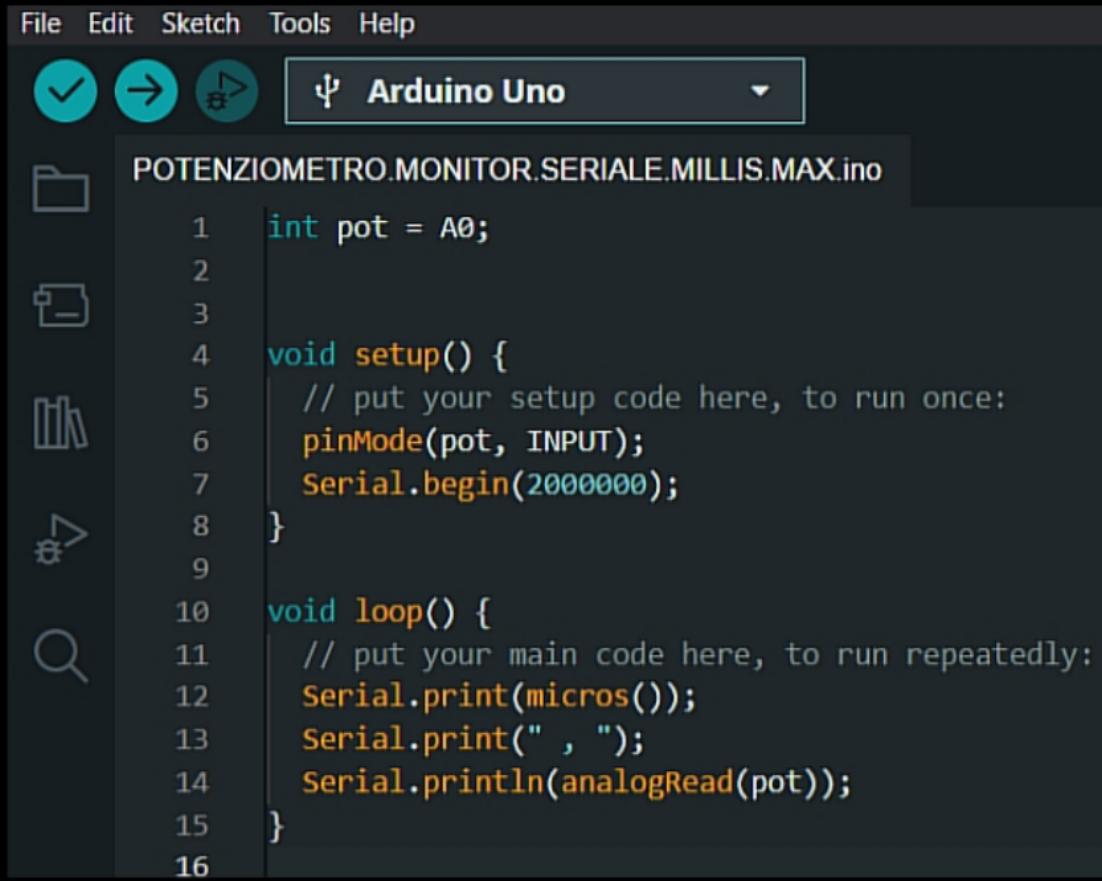
The screenshot shows the Arduino IDE interface with the following details:

- Menu Bar:** File, Edit, Sketch, Tools, Help.
- Tool Buttons:** Checkmark, Refresh, Upload, and a dropdown menu set to "Arduino Uno".
- Code Editor:** The file name is "POTENZIOMETRO.MONITOR.SERIALE.MILLIS.ino".
- Code Content:** The code reads an analog pin (A0) and prints the current time and analog value to the Serial monitor.

```
File Edit Sketch Tools Help
Arduino Uno
POTENZIOMETRO.MONITOR.SERIALE.MILLIS.ino
1 int pot = A0;
2
3
4 void setup() {
5     // put your setup code here, to run once:
6     pinMode(pot, INPUT);
7     Serial.begin(2000000);
8 }
9
10 void loop() {
11     // put your main code here, to run repeatedly:
12     Serial.print(millis());
13     Serial.print(" , ");
14     Serial.println(analogRead(pot));
15 }
16
```

CODICE LETTURA PIN ANALOGICO CON MICROS

CODE FOR THE READING OF THE ANALOG PIN WITH MICROS



The screenshot shows the Arduino IDE interface with the following details:

- Menu Bar:** File, Edit, Sketch, Tools, Help.
- Tool Buttons:** Checkmark, Refresh, Upload, Board Selection (set to Arduino Uno), and a dropdown menu.
- Code Area:** A file named "POTENZIOMETRO.MONITOR.SERIALE.MILLIS.MAX.ino" is open.
- Code Content:** The code reads an analog pin (A0) and prints the micros() value and the analogRead result to the Serial monitor.

```
File Edit Sketch Tools Help
    ✓   ➔   ⚡   Arduino Uno
POTENZIOMETRO.MONITOR.SERIALE.MILLIS.MAX.ino
1 int pot = A0;
2
3
4 void setup() {
5     // put your setup code here, to run once:
6     pinMode(pot, INPUT);
7     Serial.begin(2000000);
8 }
9
10 void loop() {
11     // put your main code here, to run repeatedly:
12     Serial.print(micros());
13     Serial.print(" , ");
14     Serial.println(analogRead(pot));
15 }
16
```

CODICE LETTURA VALORE PIN ANALOGICO

CODE FOR THE READING OF THE ANALOG PIN'S VALUE

The screenshot shows the Arduino IDE interface. At the top, there's a menu bar with File, Edit, Sketch, Tools, and Help. Below the menu is a toolbar with icons for upload (checkmark), upload with serial monitor (right arrow), and serial monitor (refresh). A dropdown menu shows "Arduino Uno". The central area is a code editor with the following content:

```
POTENZIOMETRO.MONITOR.SERIALE.VALORE.ino
1 int pot=A0;
2 int value;
3
4 void setup() {
5     // put your setup code here, to run once:
6     pinMode(pot, INPUT);
7     Serial.begin(9600);
8 }
9
10 void loop() {
11     // put your main code here, to run repeatedly:
12     value=analogRead(pot);
13     Serial.println(value);
14 }
15
```

CODICE LETTURA VOLTAGGIO PIN ANALOGICO

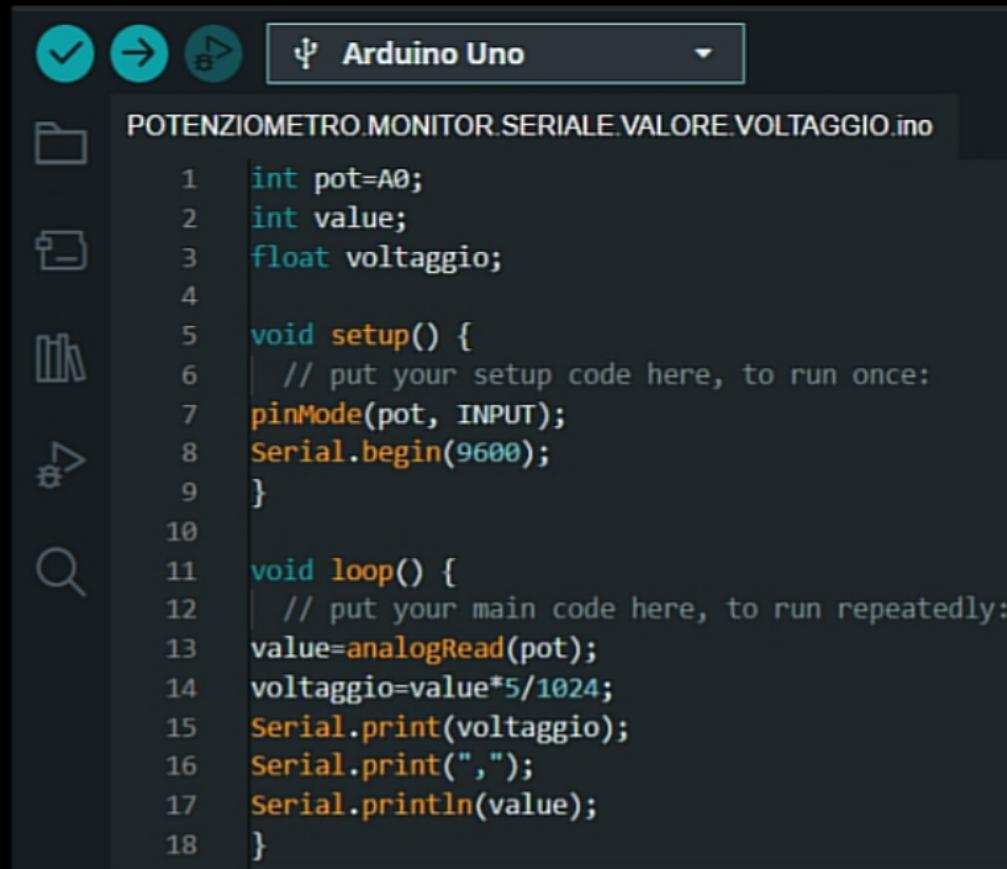
CODE FOR THE READING OF THE ANALOG PIN'S VOLTAGE

The screenshot shows the Arduino IDE interface. At the top, there is a menu bar with 'File', 'Edit', 'Sketch', 'Tools', and 'Help'. To the right of the menu is a dropdown menu set to 'Arduino Uno'. Below the menu bar, there are several icons: a checkmark, a right-pointing arrow, a play button, and a dropdown arrow. The main area displays a file named 'POTENZIOMETRO.MONITOR.SERIALE.VOLTAGGIO.ino'. The code in the editor is:

```
1 int pot=A0;
2 int value;
3 float voltaggio;
4
5 void setup() {
6     // put your setup code here, to run once:
7     pinMode(pot, INPUT);
8     Serial.begin(9600);
9 }
10
11 void loop() {
12     // put your main code here, to run repeatedly:
13     value=analogRead(pot);
14     voltaggio=value*5/1024;
15     Serial.println(voltaggio);
16 }
17
```

CODICE LETTURA VOLTAGGIO E VALORE PIN ANALOGICO

CODE FOR THE READING OF THE ANALOG PIN'S VOLTAGE AND VALUE



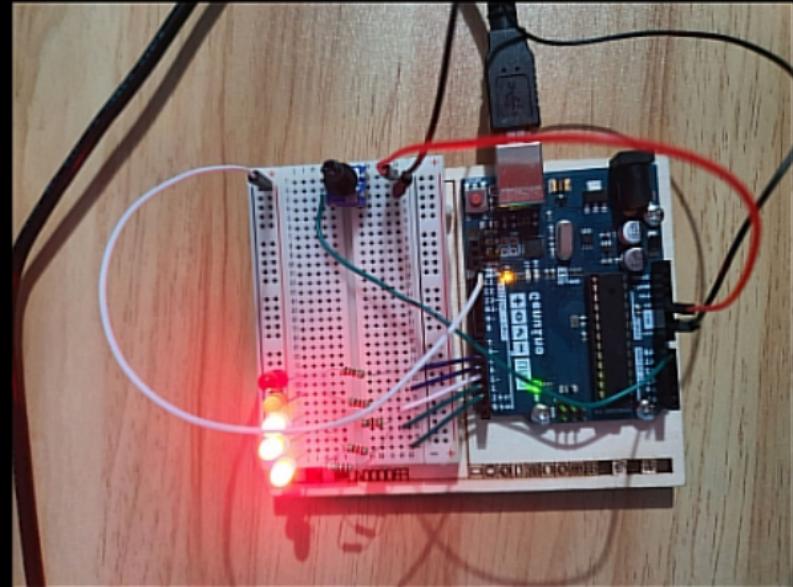
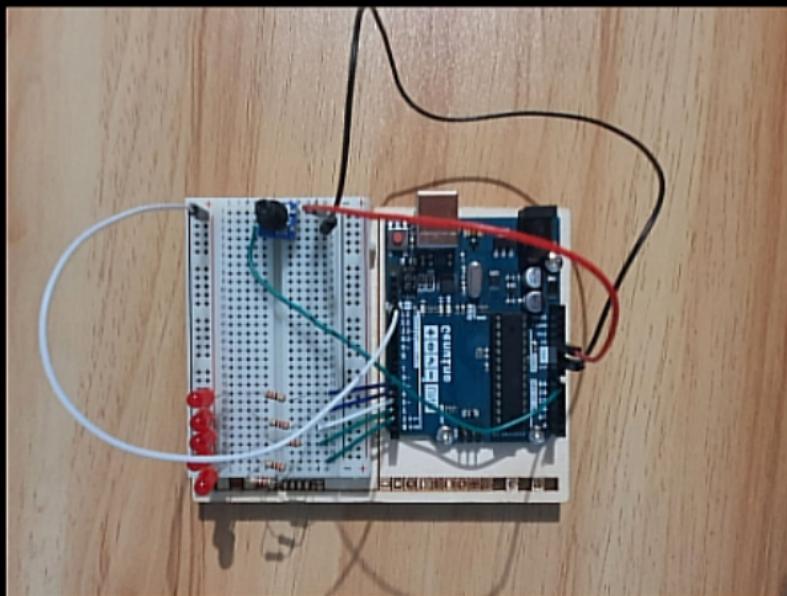
The screenshot shows the Arduino IDE interface with the following details:

- Top Bar:** Contains icons for file operations (checkmark, arrow, circular arrow), a gear icon, and the text "Arduino Uno".
- File Explorer:** On the left, it shows a folder icon and a file named "POTENZIOMETRO.MONITOR.SERIALE.VALORE.VOLTAGGIO.ino".
- Code Editor:** The main area displays the following C++ code:

```
1 int pot=A0;
2 int value;
3 float voltaggio;
4
5 void setup() {
6     // put your setup code here, to run once:
7     pinMode(pot, INPUT);
8     Serial.begin(9600);
9 }
10
11 void loop() {
12     // put your main code here, to run repeatedly:
13     value=analogRead(pot);
14     voltaggio=value*5/1024;
15     Serial.print(voltaggio);
16     Serial.print(",");
17     Serial.println(value);
18 }
```

CIRCUITO 2 / CIRCUIT 2

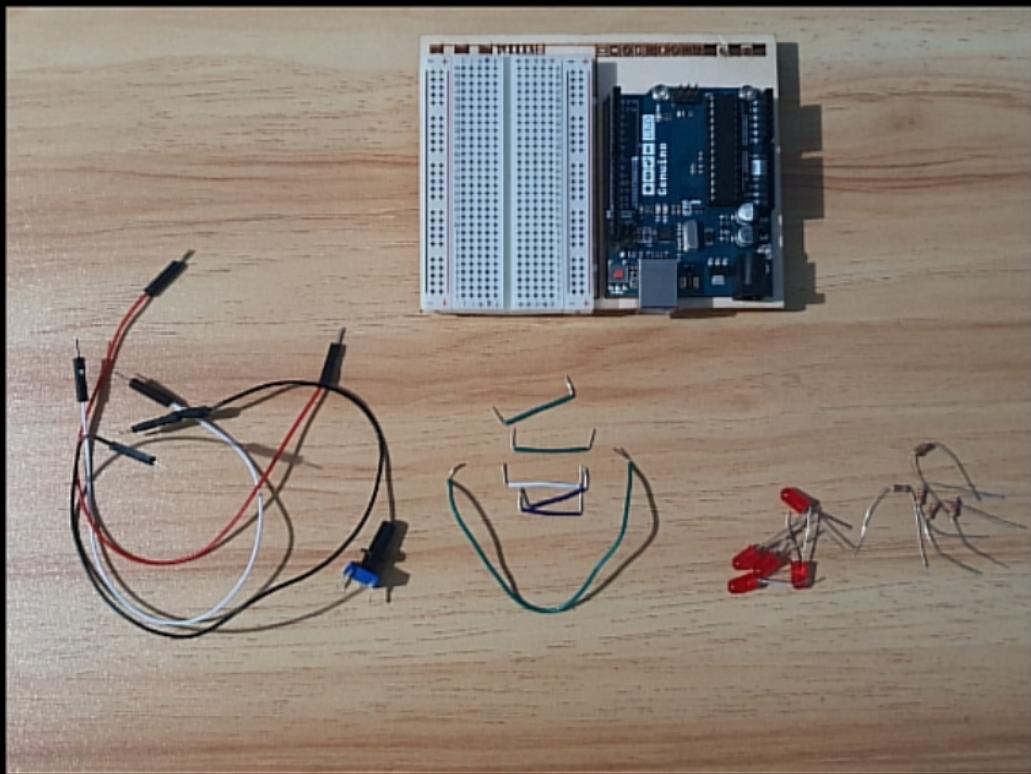
In questo circuito vi spiegheremo come utilizzare Arduino IDE per programmare il potenziometro in modo tale da accendere gradualmente una striscia di diodi LED.



In this circuit we are going to explain to you how to use Arduino IDE in order to code the potentiometer to gradually light up a series of LED diodes.

CIRCUITO 2 / CIRCUIT 2

COMPONENTI / MATERIALS NEEDED



Per realizzare questo circuito,
serviranno:

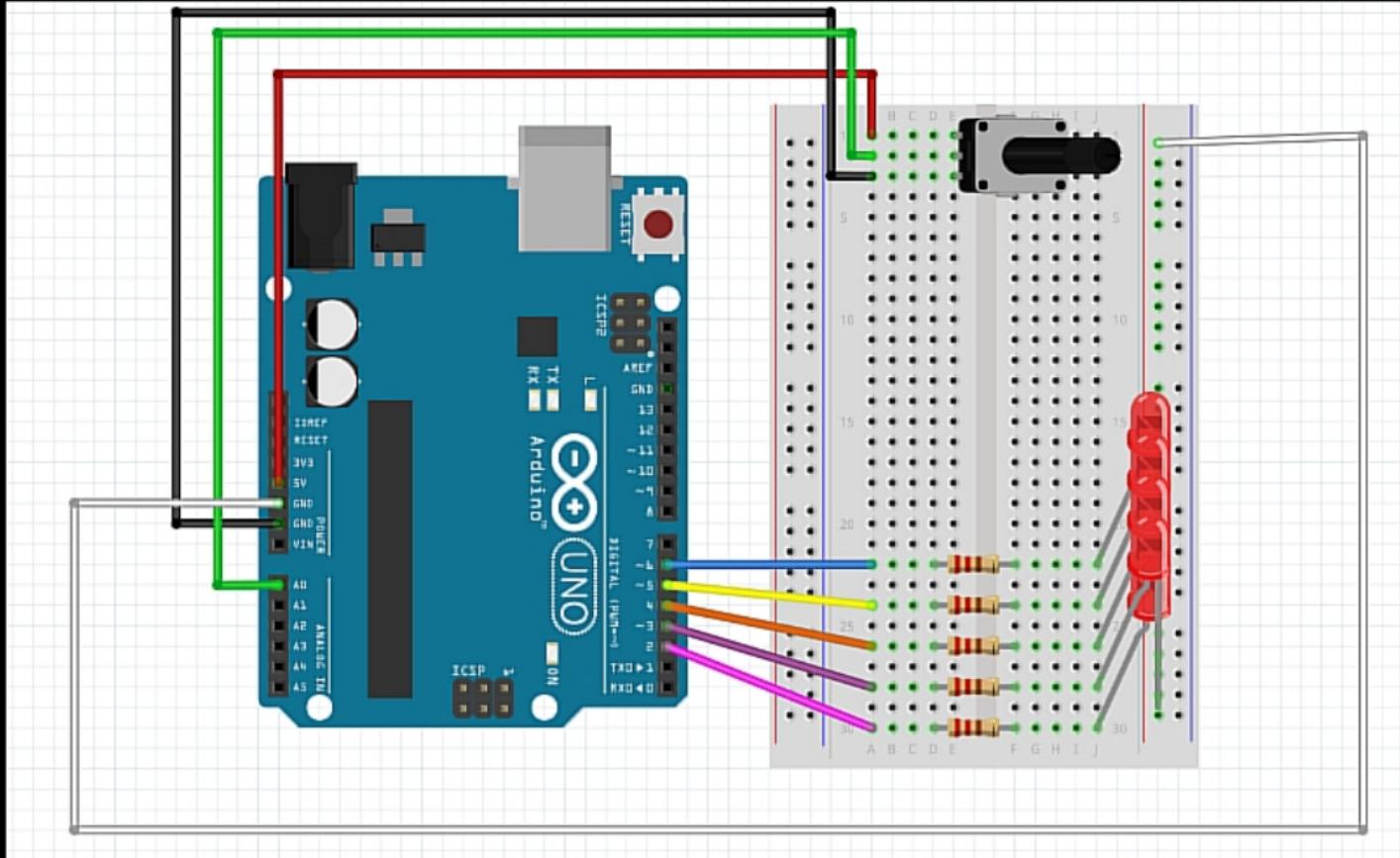
- Scheda Arduino Uno;
- Breadboard;
- 3x Cavi Jumper;
- 6x Ponticelli;
- 5x diodi LED;
- 5x Resistenze da 220Ω ;
- Potenziometro;
- Arduino IDE software
(<https://www.arduino.cc/en/software>).

*In order to realize this circuit, you will
need:*

- Arduino Uno board;
- Breadboard;
- 3x Jumper Cables;
- 6x Cables;
- 5x LED diodes;
- 5x 220Ω Resistors;
- Potentiometer;
- Arduino IDE software
(<https://www.arduino.cc/en/software>).

CIRCUITO 2 / CIRCUIT 2

SCHEMA FRITZING / FRITZING SCHEME



CODICE LETTURA PIN ANALOGICO

CODE FOR THE READING OF THE ANALOG PIN

File Edit Sketch Tools Help



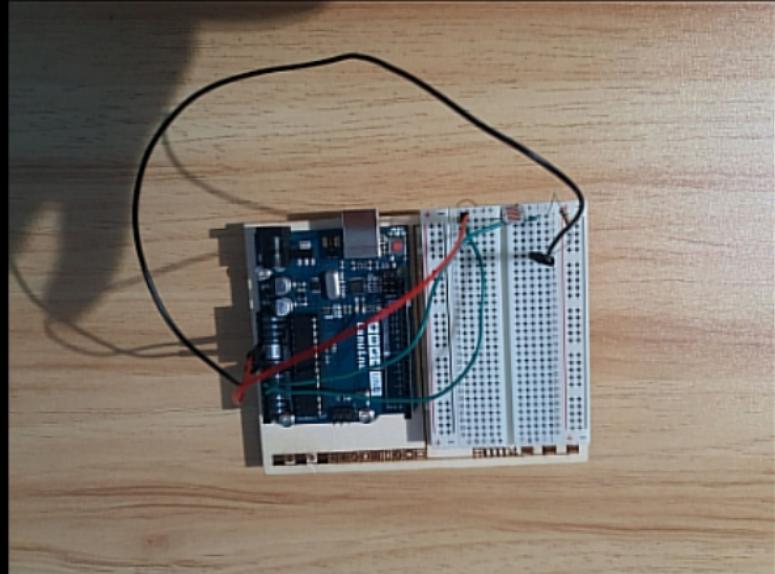
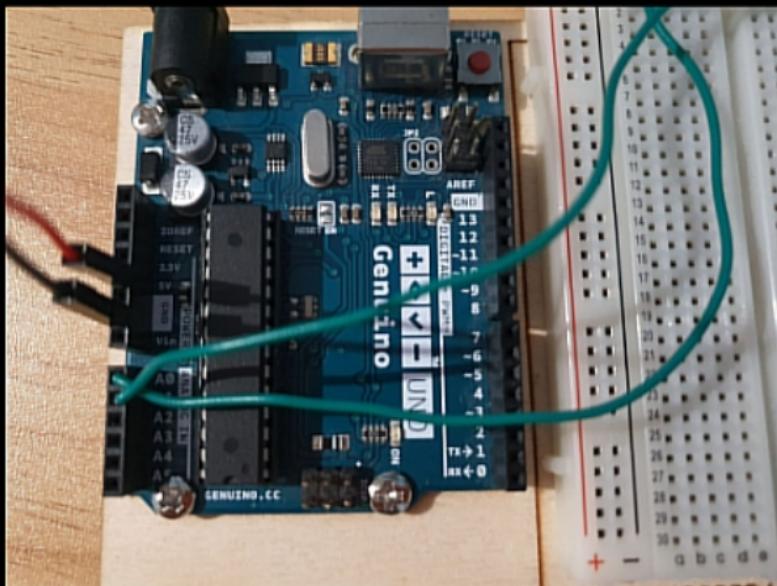
Arduino Uno

POTENZIOMETROACCENDELED.ino

```
1 const int analogPin = A0; // the pin that the potentiometer is attached to
2 const int ledCount = 5; // the number of LEDs in the bar graph
3 int ledPins[] = {2, 3, 4, 5, 6}; // an array of pin numbers to which LEDs are attached
4
5
6 void setup() {
7 // ciclo for, thisted è una variabile muta, parte da zero e aumenta di 1 finché non arriva al n° totale
8 for (int thisted = 0; thisted < ledCount; thisted++) {
9 pinMode (ledPins [thisted], OUTPUT);
10 }
11 }
12
13
14 void loop() {
15 // sensorReading legge il sensore
16 int sensorReading = analogRead (analogPin) ;
17 // map è una funzione che prende il valore del sensorReading e lo trasforma da 0 a ledCount: sensorReading che inizialmente va da 0 a 1023 poi cambia da 0 a ledCount
18 int ledLevel = map (sensorReading, 0, 1023, 0, ledCount);
19 // ciclo for
20 for (int thisted = 0; thisted < ledCount; thisted++) {
21 // per ogni giro di ciclo for thisted ha un unico valore
22 if (thisted < ledLevel) [
23 digitalWrite (ledPins [thisted], HIGH);
24 ]else [ // con digitalWrite si accendono i LED
25 digitalWrite (ledPins[thisted], LOW);
26 ]
27 }
28 }
29 // andiamo a leggere un segnale analogico che in uscita da un numero da 0 a 1023, poi mappo i valori da 0 a 1023 fino a renderli da 0 a 5.
30 // quando giro il potenziometro i led restano accesi, ma se lo giro dall'altro lato si spengono grazie al comando LOW, perché il livello dei led diminuisce
31 // lo stesso codice si potrebbe utilizzare con un sensore di livello, ad esempio quello del gas
32 }
```

CIRCUITO 3 / CIRCUIT 3

In questo circuito vi spiegheremo come utilizzare Arduino IDE per leggere il valore di una resistenza variabile, come il fotoresistore.



In this circuit we are going to explain to you how to use Arduino IDE in order to see the value of a variable resistor like a photoresistor.

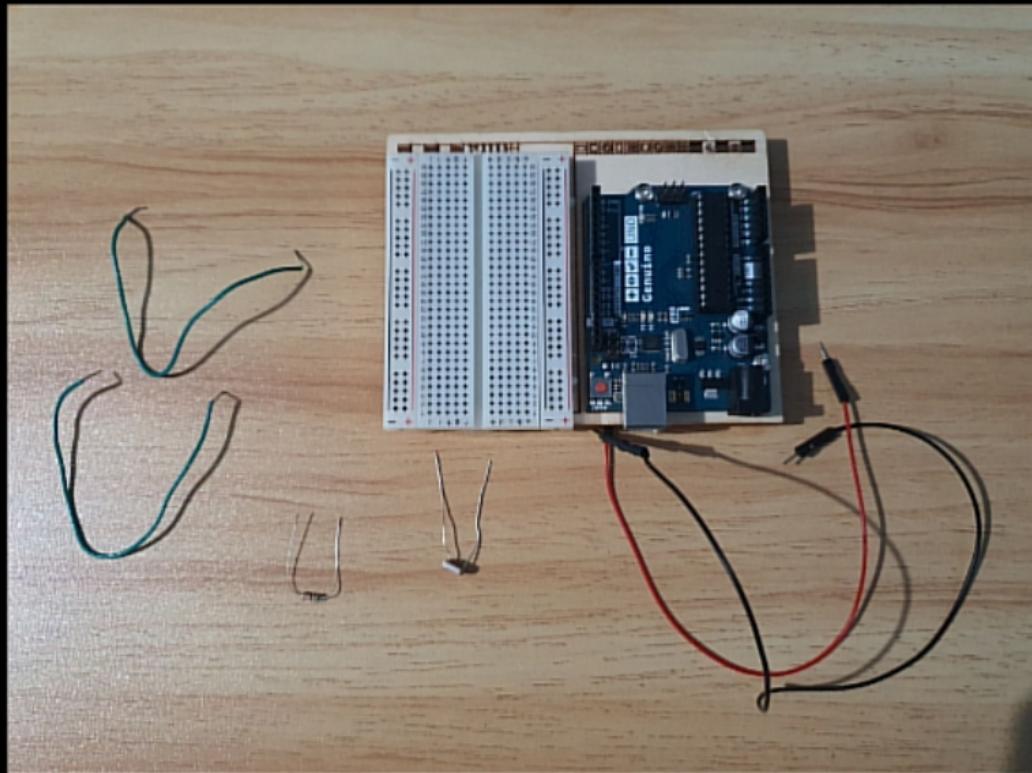
CIRCUITO 3 / *CIRCUIT 3* **COMPONENTI / MATERIALS NEEDED**

Per realizzare questo circuito,
serviranno:

- Scheda Arduino Uno;
- Breadboard;
- 2x Cavi Jumper;
- 2x Ponticelli;
- Fotoresistore;
- Resistenza da $1\text{k}\Omega$;
- Arduino IDE software
(<https://www.arduino.cc/en/software>).

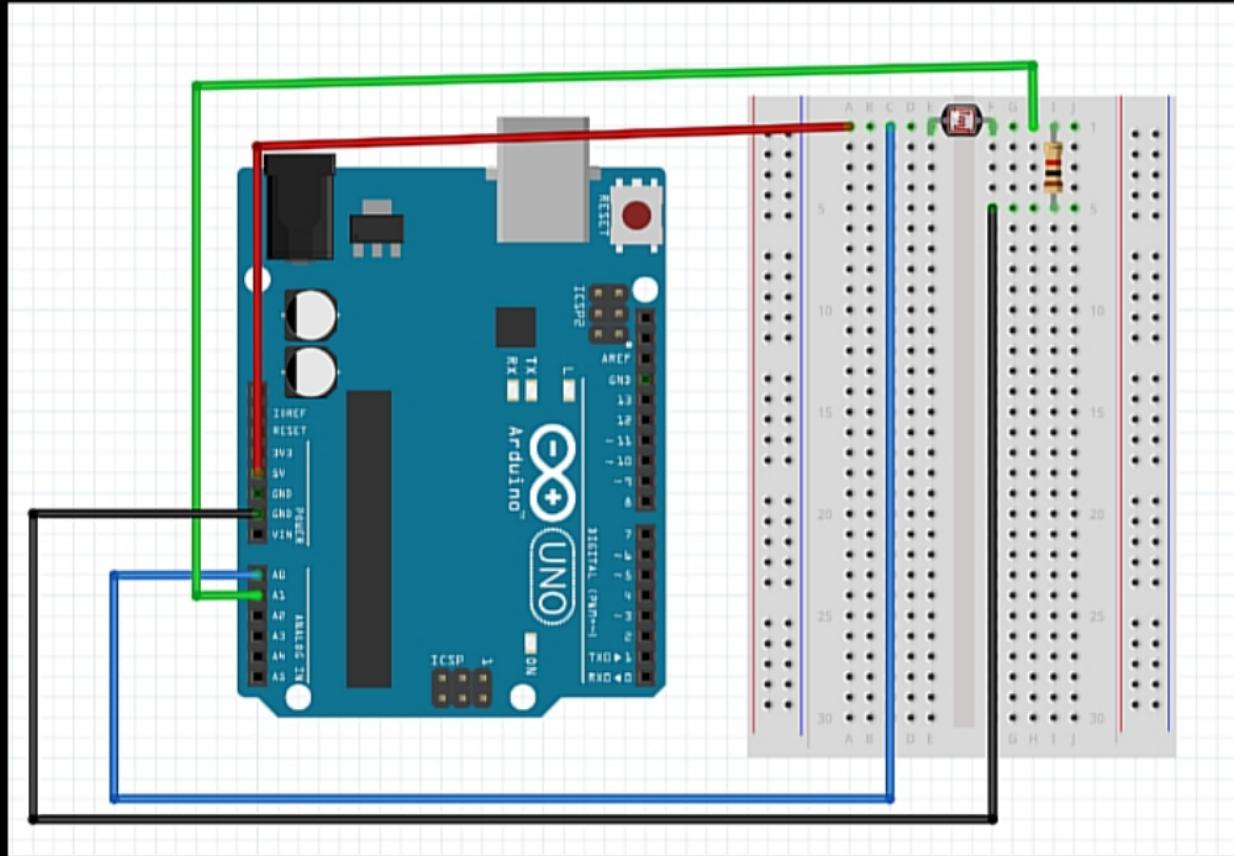
*In order to realize this circuit, you will
need:*

- Arduino Uno board;
- Breadboard;
- 2x Jumper Cables;
- 2x Cables;
- Photoresistor;
- $1\text{k}\Omega$ Resistor;
- Arduino IDE software
(<https://www.arduino.cc/en/software>).



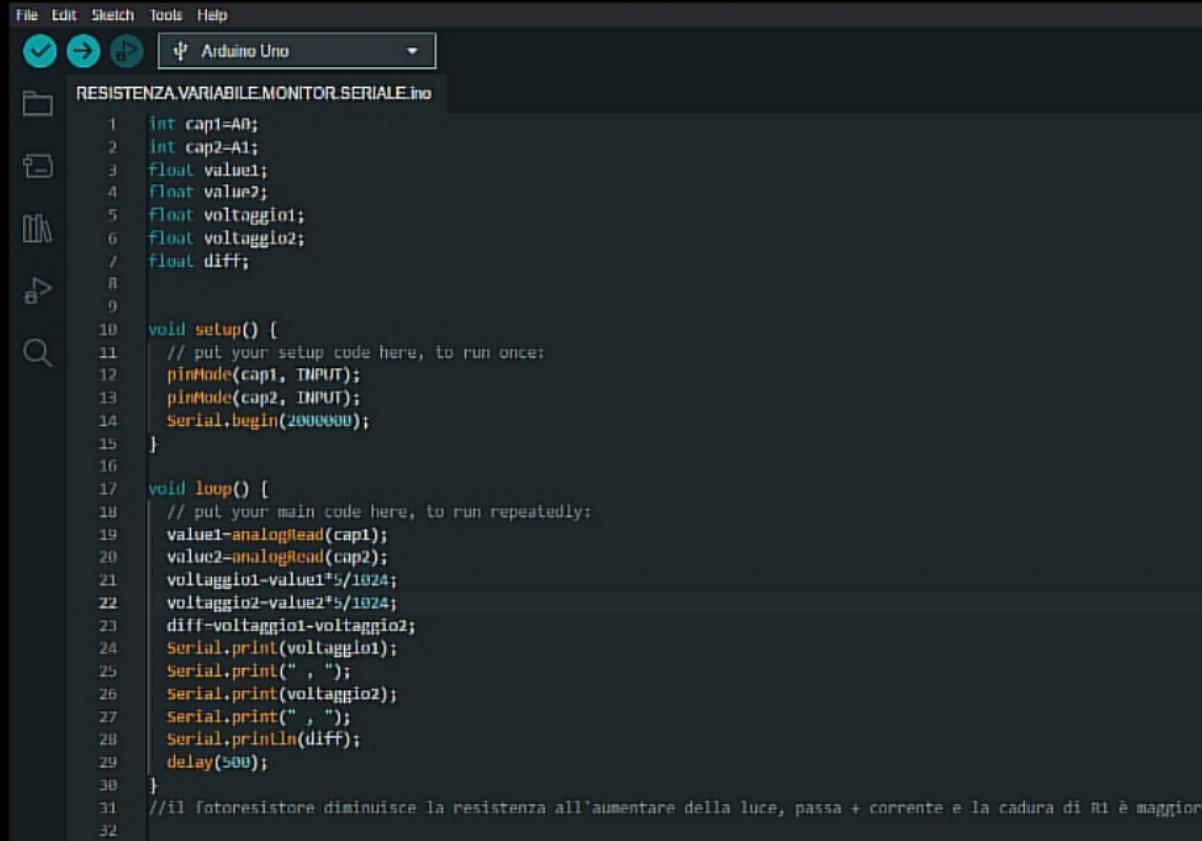
CIRCUITO 3 / CIRCUIT 3

SCHEMA FRITZING / FRITZING SCHEME



CODICE LETTURA PIN ANALOGICI

CODE FOR THE READING OF THE ANALOG PINS



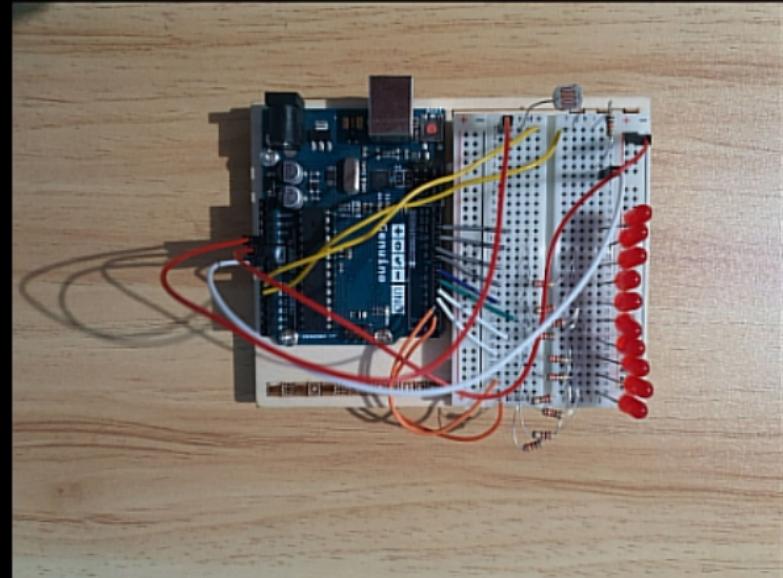
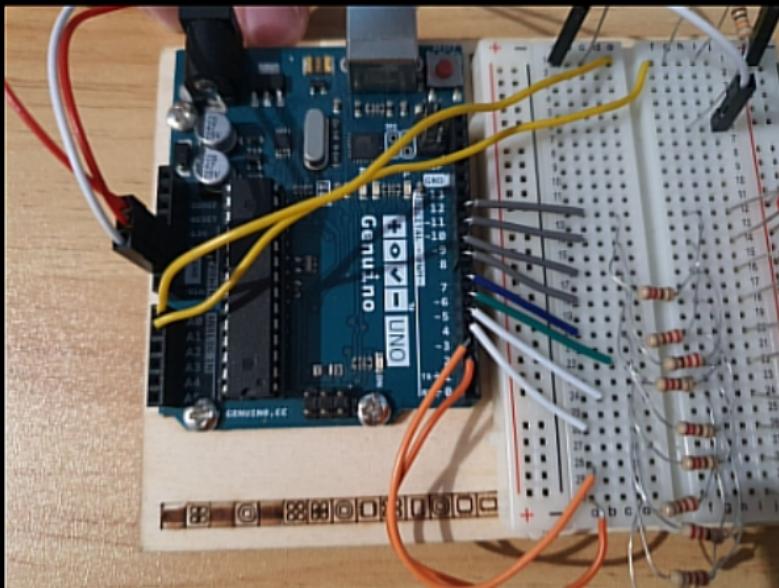
The screenshot shows the Arduino IDE interface with the following details:

- Menu Bar:** File, Edit, Sketch, Tools, Help.
- Tool Buttons:** Checkmark, Refresh, Upload, and a dropdown menu set to "Arduino Uno".
- Code Editor:** The file name is "RESISTENZA,VARIABLE,MONITOR,SERIALE.ino".
- Code Content:** The code reads two analog pins (A0 and A1), calculates their voltage, and prints them to the serial monitor. It also calculates the difference between the voltages and prints it. A note at the bottom explains that increasing light intensity decreases the resistance, which increases current and makes R1 larger.

```
1 int cap1=A0;
2 int cap2=A1;
3 float value1;
4 float value2;
5 float voltaggio1;
6 float voltaggio2;
7 float diff;
8
9
10 void setup() {
11     // put your setup code here, to run once:
12     pinMode(cap1, INPUT);
13     pinMode(cap2, INPUT);
14     Serial.begin(2000000);
15 }
16
17 void loop() {
18     // put your main code here, to run repeatedly:
19     value1=analogRead(cap1);
20     value2=analogRead(cap2);
21     voltaggio1=value1*5/1024;
22     voltaggio2=value2*5/1024;
23     diff=voltaggio1-voltaggio2;
24     Serial.print(voltaggio1);
25     Serial.print(" , ");
26     Serial.print(voltaggio2);
27     Serial.print(" , ");
28     Serial.println(diff);
29     delay(500);
30 }
31 //il fotoresistore diminuisce la resistenza all'aumentare della luce, passa + corrente e la caduta di R1 è maggiore
```

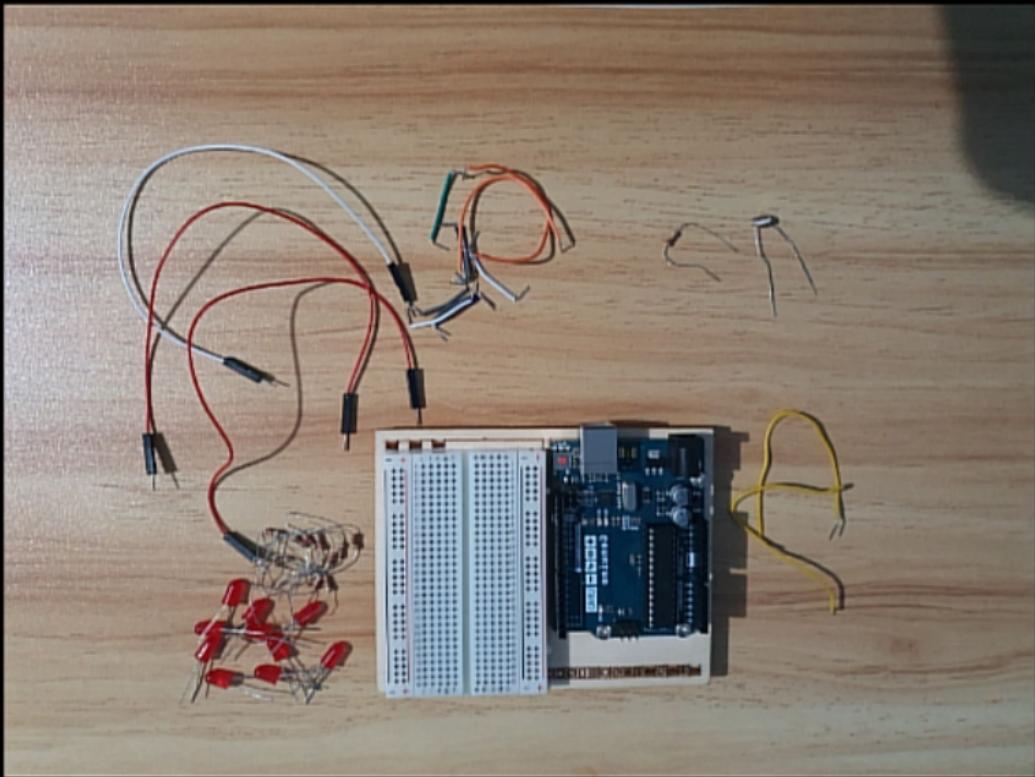
CIRCUITO 4 / CIRCUIT 4

In questo circuito vi spiegheremo come utilizzare Arduino IDE per programmare il fotoresistore in modo tale da accendere gradualmente una striscia di diodi LED.



In this circuit we are going to explain to you how to use Arduino IDE in order to code the photoresistor to gradually light some LEDs.

CIRCUITO 4 / CIRCUIT 4



COMPONENTI / MATERIALS NEEDED

Per realizzare questo circuito,
serviranno:

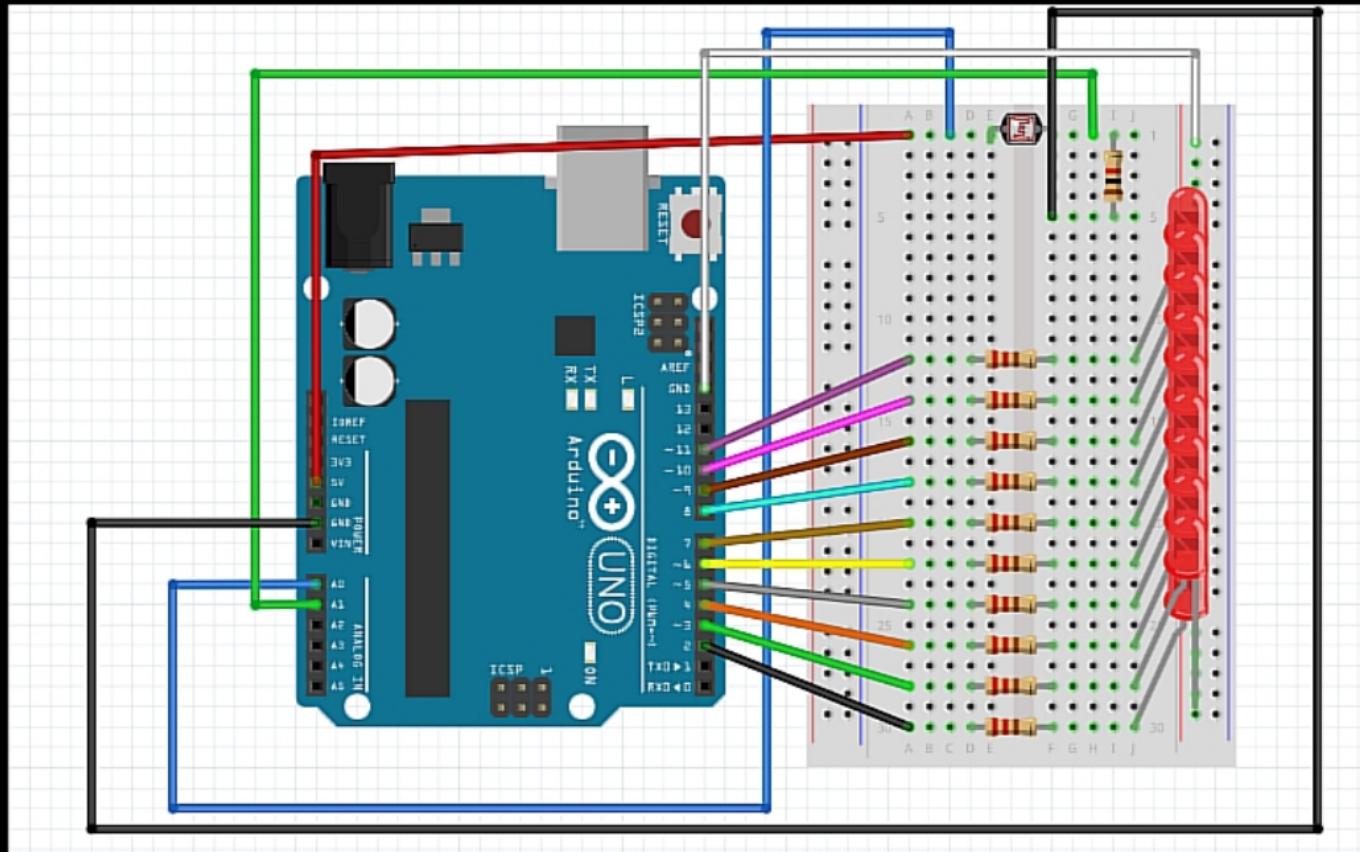
- Scheda Arduino Uno;
- Breadboard;
- 3x Cavi Jumper;
- 12x Ponticelli;
- Fotoresistore;
- Resistenza da $1\text{k}\Omega$;
- 10x Diodi LED rossi;
- 10x Resistenze da 220Ω ;
- Arduino IDE software
(<https://www.arduino.cc/en/software>).

*In order to realize this circuit, you will
need:*

- Arduino Uno board;
- Breadboard;
- 3x Jumper Cables;
- 12x Cables;
- Photoresistor;
- $1\text{k}\Omega$ Resistor;
- 10x Red LED diodes;
- 10x 220Ω Resistors;
- Arduino IDE software
(<https://www.arduino.cc/en/software>).

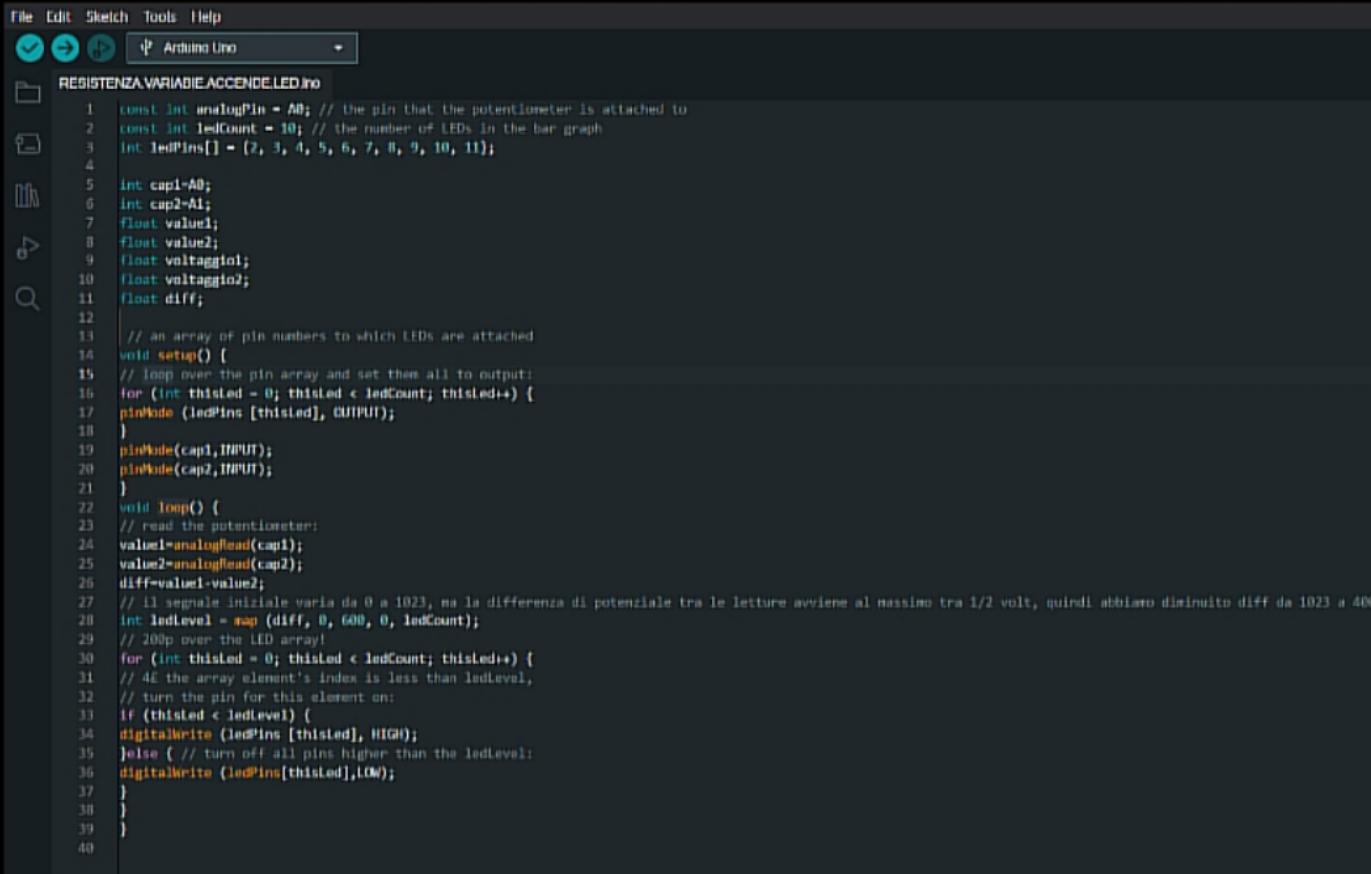
CIRCUITO 4 / CIRCUIT 4

SCHEMA FRITZING / FRITZING SCHEME



CODICE LETTURA PIN ANALOGICI

CODE FOR THE READING OF THE ANALOG PINS



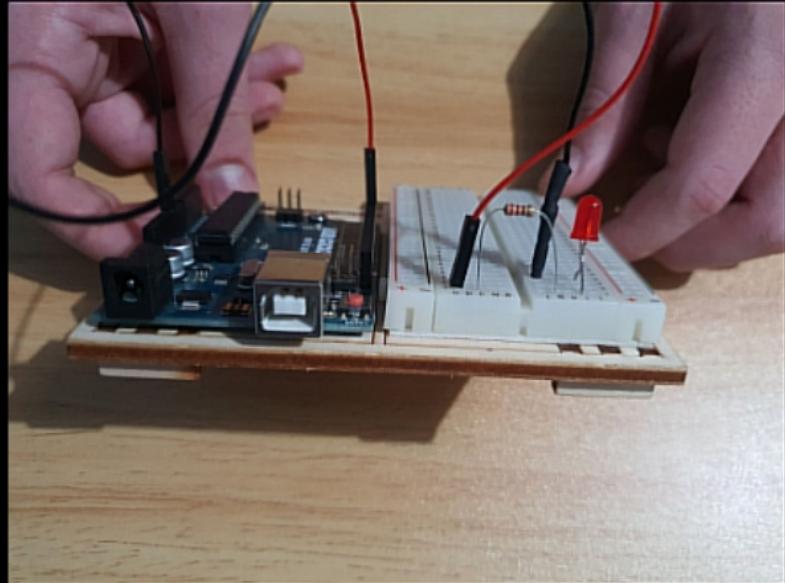
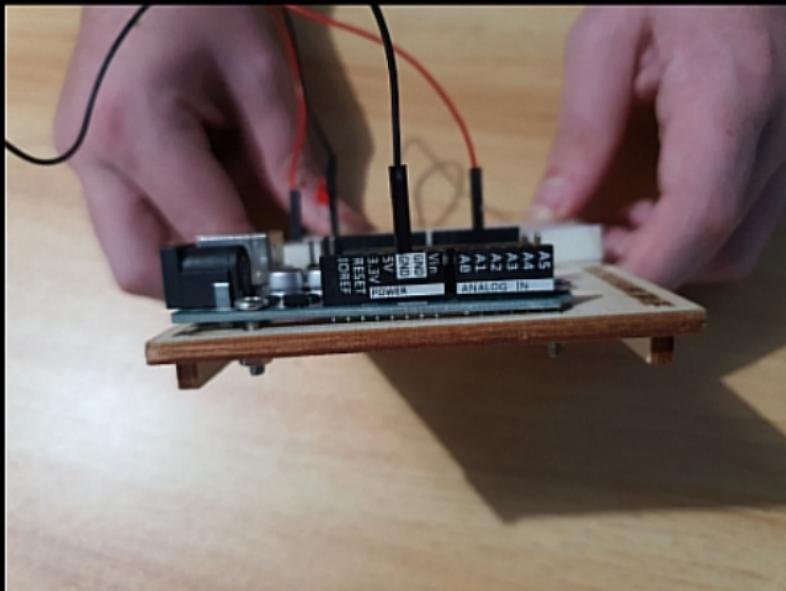
The screenshot shows the Arduino IDE interface with the following details:

- Menu Bar:** File, Edit, Sketch, Tools, Help.
- Toolbar:** Includes icons for Open, Save, Print, and Search.
- Sketch Name:** RESISTENZA VARIABILE ACCENDE LED.ino
- Code Area:** Displays the C++ code for reading analog pins. The code initializes pins A0 and A1 as inputs, and pins 2 through 11 as outputs. It reads values from both capacitive sensors and calculates the difference. Then, it iterates through the LED array, turning on LEDs based on the difference value. The code uses the map function to scale the difference from 0 to 1023 to a range of 0 to 600, which is then used to index into the ledPins array to turn on the appropriate LEDs.

```
1 const int analogPin = A0; // the pin that the potentiometer is attached to
2 const int ledCount = 10; // the number of LEDs in the bar graph
3 int ledPins[] = {2, 3, 4, 5, 6, 7, 8, 9, 10, 11};
4
5 int cap1=A0;
6 int cap2=A1;
7 float value1;
8 float value2;
9 float voltaggio1;
10 float voltaggio2;
11 float diff;
12
13 // an array of pin numbers to which LEDs are attached
14 void setup() {
15 // loop over the pin array and set them all to output:
16 for (int thisled = 0; thisled < ledCount; thisled++) {
17 pinMode (ledPins [thisled], OUTPUT);
18 }
19 pinMode(cap1,INPUT);
20 pinMode(cap2,INPUT);
21 }
22 void loop() {
23 // read the potentiometer:
24 value1=analogRead(cap1);
25 value2=analogRead(cap2);
26 diff=value1-value2;
27 // il segnale iniziale varia da 0 a 1023, ma la differenza di potenziale tra le letture avviene al massimo tra 1/2 volt, quindi abbiamo diminuito diff da 1023 a 400
28 int ledlevel = map (diff, 0, 600, 0, ledCount);
29 // loop over the LED array
30 for (int thisled = 0; thisled < ledCount; thisled++) {
31 // if the array element's index is less than ledLevel,
32 // turn the pin for this element on:
33 if (thisled < ledLevel) {
34 digitalWrite (ledPins [thisled], HIGH);
35 } else { // turn off all pins higher than the ledLevel:
36 digitalWrite (ledPins[thisled],LOW);
37 }
38 }
39 }
```

CIRCUITO 5 / CIRCUIT 5

In questo circuito vi spiegheremo come utilizzare Arduino IDE per programmare i pin PWM di Arduino in modo tale da far lampeggiare un diodo LED facendo passare la luminosità da gradi diversi.



In this circuit we are going to explain to you how to use Arduino IDE in order to code the PWM pins to make an LED diode blink by fading its light.

CIRCUITO 5 /

CIRCUIT 5

COMPONENTI /

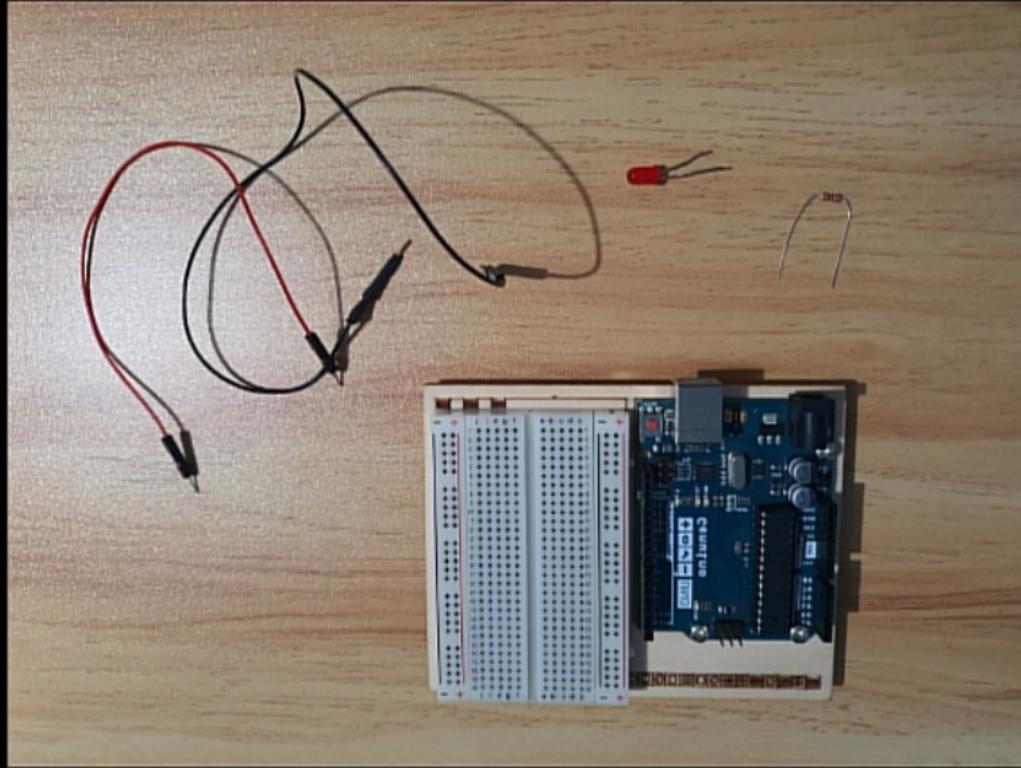
MATERIALS NEEDED

Per realizzare questo circuito,
serviranno:

- Scheda Arduino Uno;
- Breadboard;
- 2x Cavi Jumper;
- Diodo LED rosso;
- Resistenza da 220Ω ;
- Arduino IDE software
(<https://www.arduino.cc/en/software>).

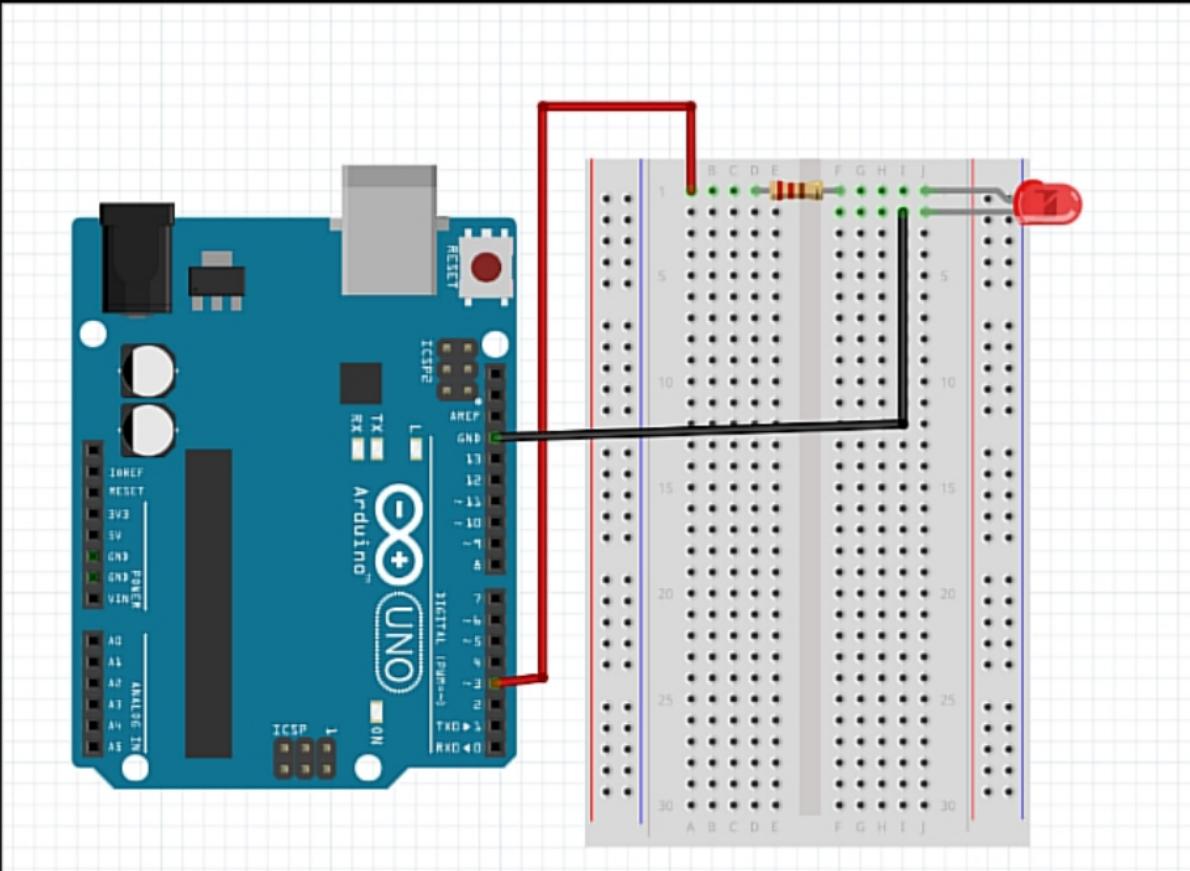
*In order to realize this circuit, you will
need:*

- Arduino Uno board;
- Breadboard;
- 2x Jumper Cables;
- Red LED diode;
- 220Ω Resistor;
- Arduino IDE software
(<https://www.arduino.cc/en/software>).



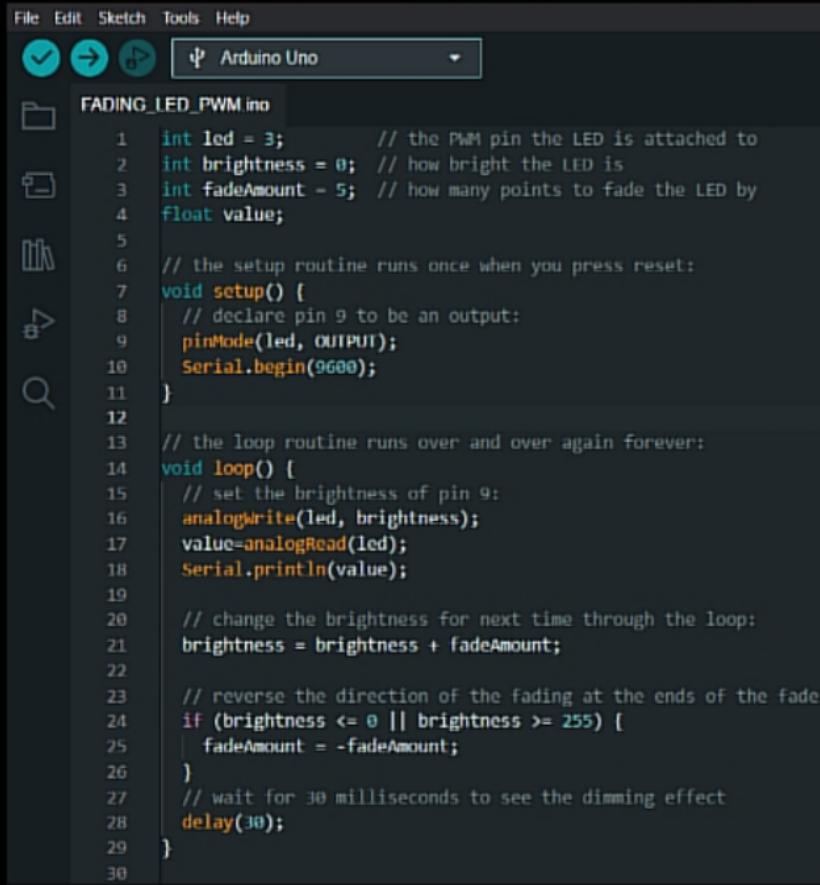
CIRCUITO 5 / CIRCUIT 5

SCHEMA FRITZING / FRITZING SCHEME



CODICE LETTURA PIN ANALOGICI

CODE FOR THE READING OF THE ANALOG PINS



The screenshot shows the Arduino IDE interface with the following details:

- File menu:** File, Edit, Sketch, Tools, Help.
- Tools menu:** A dropdown menu showing "Arduino Uno".
- Sketch:** FADING_LED_PWM.ino
- Code:** The code listed below is displayed in the main editor area.

```
1 int led = 3;           // the PWM pin the LED is attached to
2 int brightness = 0;    // how bright the LED is
3 int fadeAmount = 5;   // how many points to fade the LED by
4 float value;
5
6 // the setup routine runs once when you press reset:
7 void setup() {
8     // declare pin 9 to be an output:
9     pinMode(led, OUTPUT);
10    Serial.begin(9600);
11 }
12
13 // the loop routine runs over and over again forever:
14 void loop() {
15     // set the brightness of pin 9:
16     analogWrite(led, brightness);
17     value=analogRead(led);
18     Serial.println(value);
19
20     // change the brightness for next time through the loop:
21     brightness = brightness + fadeAmount;
22
23     // reverse the direction of the fading at the ends of the fade:
24     if (brightness <= 0 || brightness >= 255) {
25         fadeAmount = -fadeAmount;
26     }
27     // wait for 30 milliseconds to see the dimming effect
28     delay(30);
29 }
```

**GRAZIE PER
L'ATTENZIONE!**

LINK CANALE YT (EN LINK YT CHANNEL):

<https://www.youtube.com/channel/UCIXRpnzghK-d6ahtxNOkVqA>

LINK CODICI GITHUB (EN LINK CODES ON GITHUB):

https://github.com/ISTITUTOFREUDSIMBALADD/ISTITUTOFREUDSIMBALADD/blob/main/CODICI_CODES.docx