



Microcontrollers & Applications

Lecture 1.2: Programming and Testing Environments

Programming & Testing Environments



Arduino IDE + Board

WokWi

Tinker CAD

Fritzing

Proteus ISIS + Arduino IDE

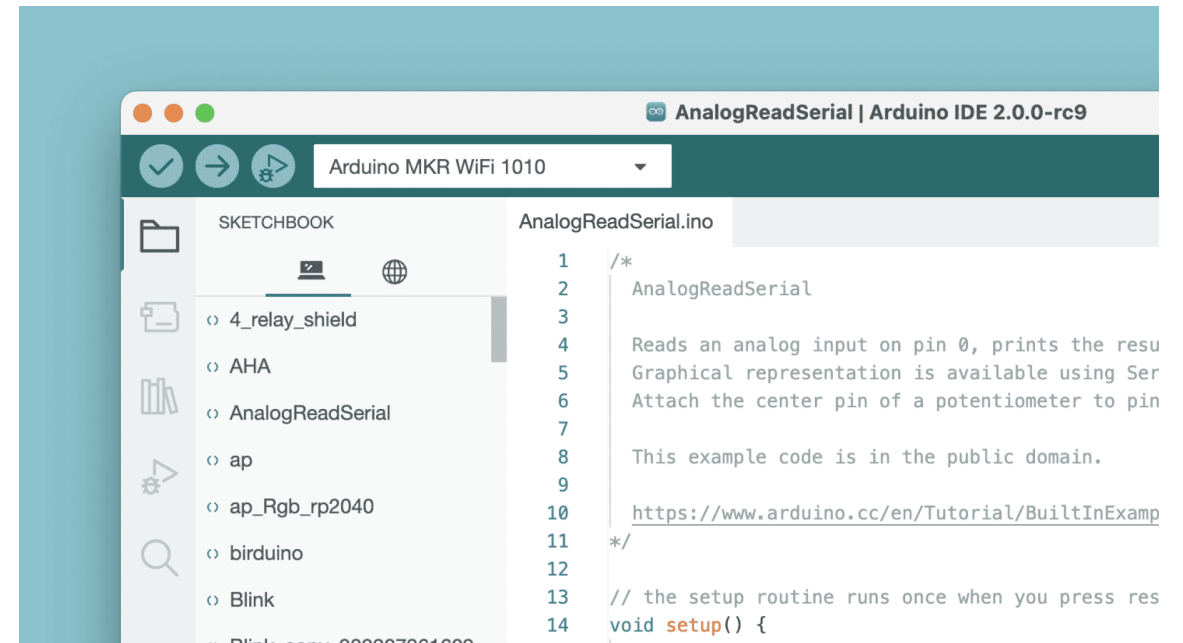
Arduino IDE Main Screen



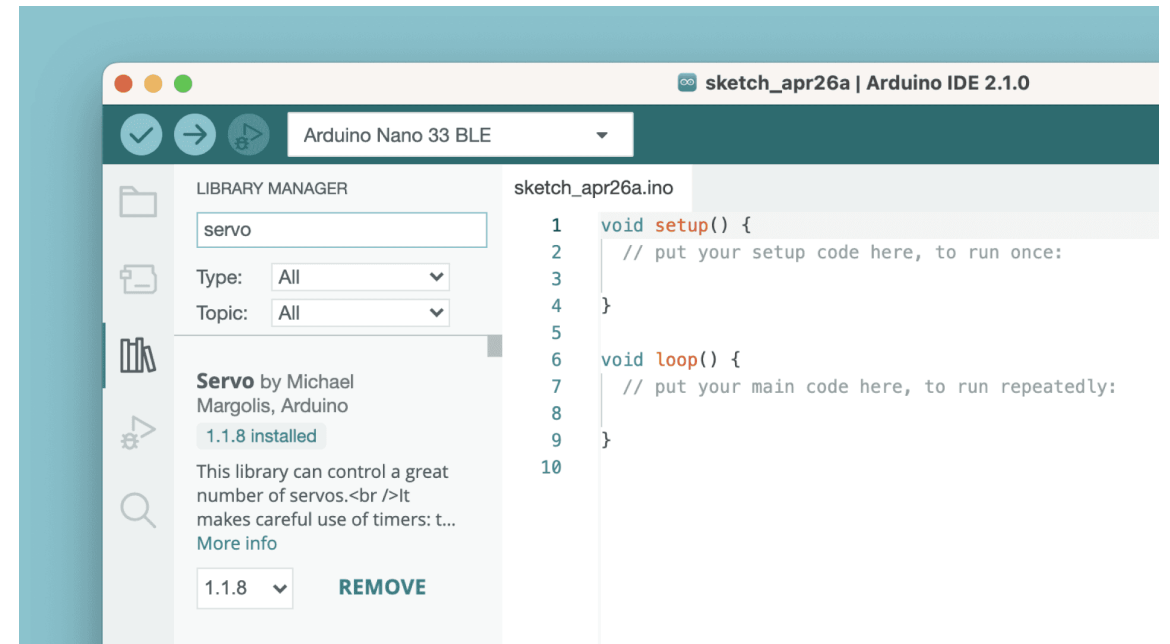
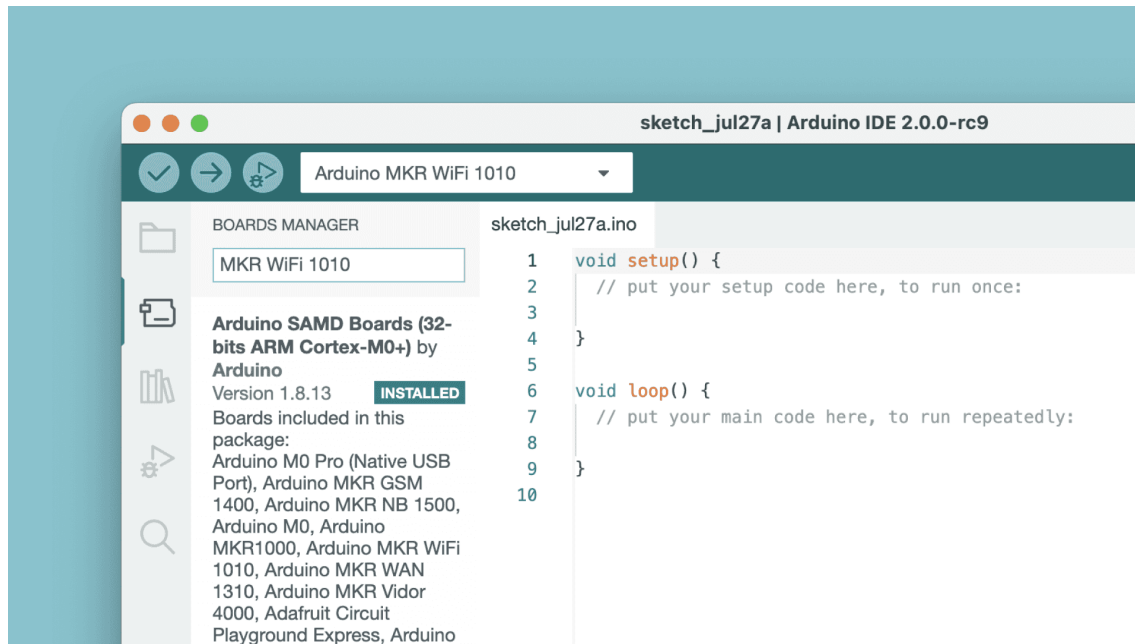
- **Verify / Upload** - compile and upload your code to your Arduino Board.
- **Select Board & Port** - detected Arduino boards automatically show up here, along with the port number.
- **Sketchbook** - here you will find all of your sketches locally stored on your computer. Additionally, you can sync with the Arduino Cloud, and also obtain your sketches from the online environment.
- **Boards Manager** - browse through Arduino & third-party packages that can be installed. For example, using a MKR WiFi 1010 board requires the Arduino SAMD Boards package installed.
- **Library Manager** - browse through thousands of Arduino libraries, made by Arduino & its community.
- **Debugger** - test and debug programs in real time.
- **Search** - search for keywords in your code.
- **Open Serial Monitor** - opens the Serial Monitor tool, as a new tab in the console.
- **Open Serial Plotter** - opens the Serial Plotter tool, as a new tab.

Arduino IDE: Install & Sketchbook

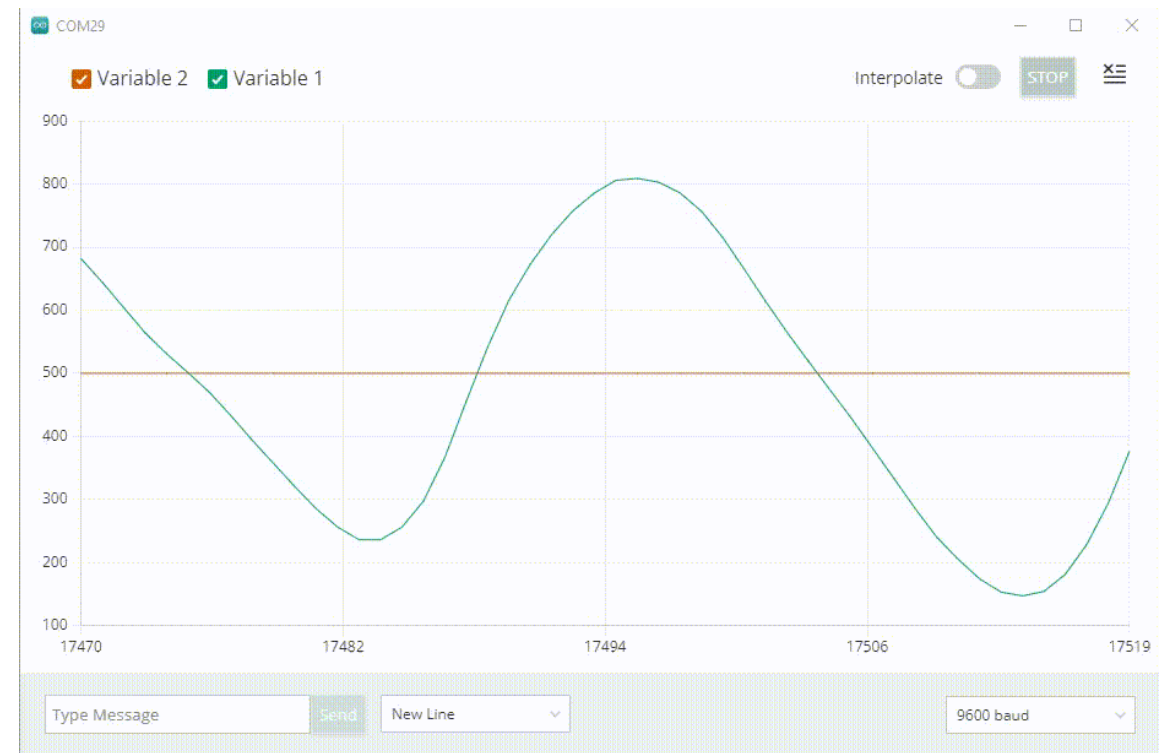
- Prepare the programming environment:
 - Download and install it from <https://www.arduino.cc/en/software>
 - Or use it online via <https://create.arduino.cc/editor>
- Code & run the program:
 - Connect the Arduino board via USB port
 - Choose the correct board and port
 - Code the program in Arduino IDE
 - Upload to compile and transfer program to the board
 - After completing the transfer, it starts running automatically, we hope 😊



Arduino IDE: Boards Manager & Library Manager



Arduino IDE: Serial Monitor & Serial Plotter



WokWi: <https://wokwi.com/>



Simulate IoT Projects in Your Browser

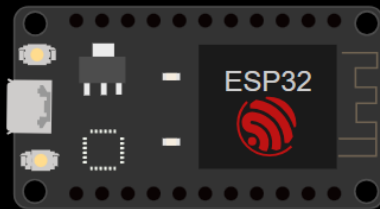
[Discord Community](#)

[LinkedIn Group](#)

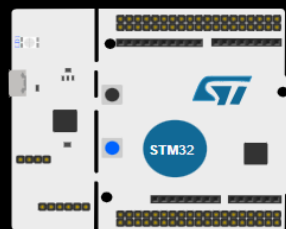
Simulate with Wokwi Online



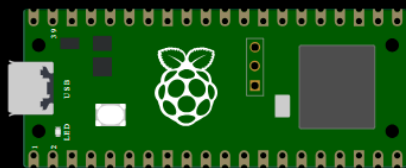
Arduino (Uno, Mega, Nano)



ESP32



STM32



Pi Pico

Online Arduino Simulator

A faster way to prototype Uno, Mega and Nano projects

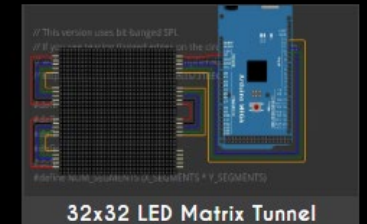
Featured projects



Simon Says Game



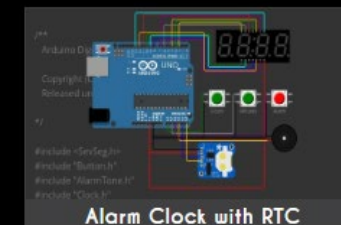
32 Servos Dancing



32x32 LED Matrix Tunnel



Mini Piano



Alarm Clock with RTC



Electronic Safe



Nano Pong



Touch LCD Breakout Game



Dino Game

WokWi: Coding & Simulation

WOKWI

SAVE

SHARE

mini-piano.ino by urish

Docs

mini-piano.ino

pitches.h

diagram.json

Library Manager

```
1  /**
2   * Mini piano for Arduino.
3   *
4   * You can control the colorful buttons with your keyboard:
5   * After starting the simulation, click anywhere in the diagram to focus it.
6   * Then press any key between 1 and 8 to play the piano (1 is the lowest note,
7   * 8 is the highest).
8   *
9   * Copyright (C) 2021, Uri Shaked. Released under the MIT License.
10  */
11
12  #include "pitches.h"
13
14  #define SPEAKER_PIN 8
15
16  const uint8_t buttonPins[] = { 12, 11, 10, 9, 7, 6, 5, 4 };
17  const int buttonTones[] = {
18    NOTE_C4, NOTE_D4, NOTE_E4, NOTE_F4,
19    NOTE_G4, NOTE_A4, NOTE_B4, NOTE_C5
20  };
21  const int numTones = sizeof(buttonPins) / sizeof(buttonPins[0]);
22
23  void setup() {
24    for (uint8_t i = 0; i < numTones; i++) {
25      pinMode(buttonPins[i], INPUT_PULLUP);
26    }
27    pinMode(SPEAKER_PIN, OUTPUT);
28  }
29
30  void loop() {
31    int pitch = 0;
32    for (uint8_t i = 0; i < numTones; i++) {
33      if (digitalRead(buttonPins[i]) == LOW) {
34        pitch = buttonTones[i];
35      }
36    }
37    if (pitch) {
38      tone(SPEAKER_PIN, pitch);
39    } else {
40      noTone(SPEAKER_PIN);
41    }
42  }
43  }
```

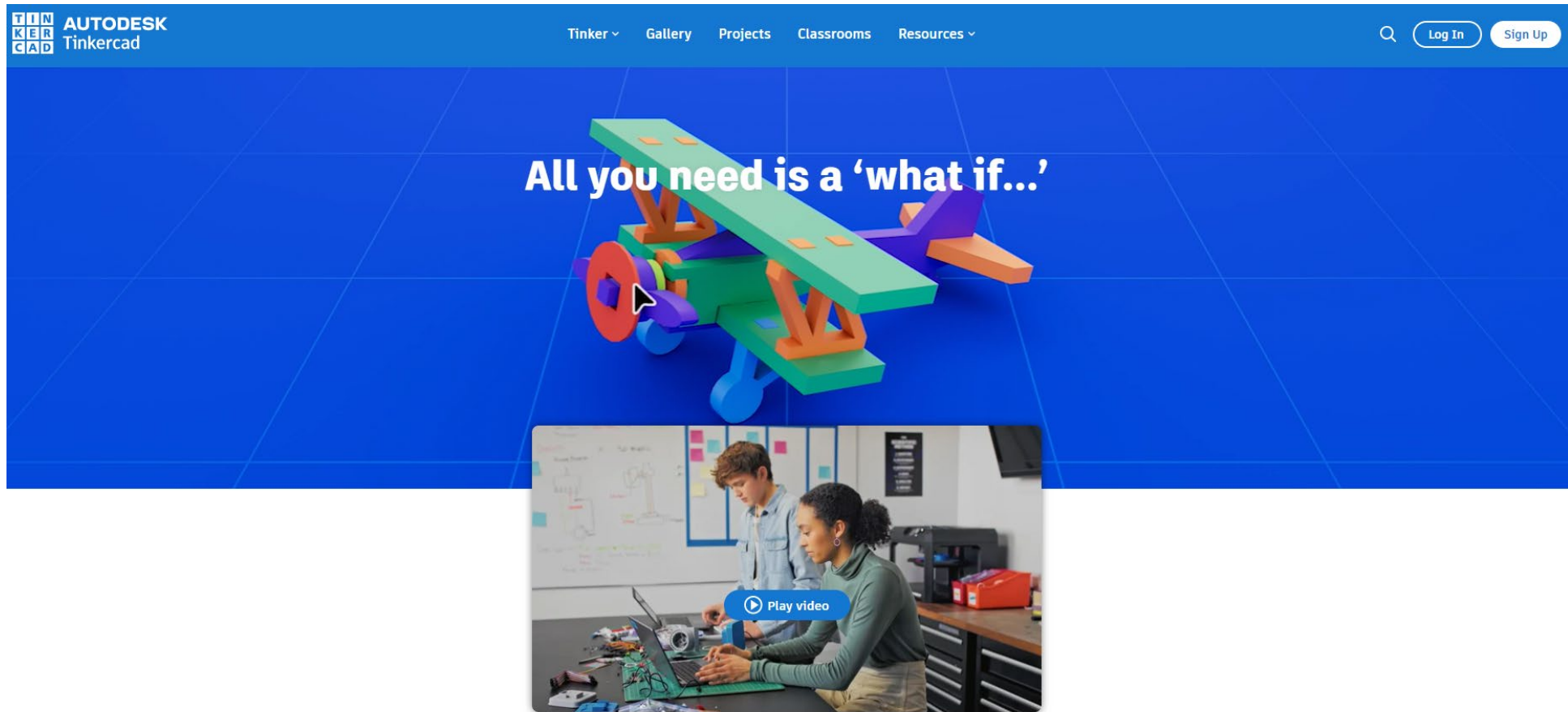
Simulation

▶

+


⋮

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




Tinkercad is a free web app for 3D design, electronics, and coding, trusted by over 50 million people around the world.

TinkerCAD: Main Screen

 **AUTODESK**
Tinkercad

Tinker ▾ Gallery Projects Classrooms Resources ▾


Yalcin Isler

Search designs...

Classes

Designs

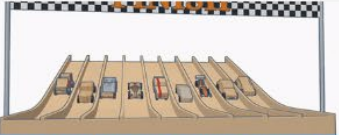
Tutorials

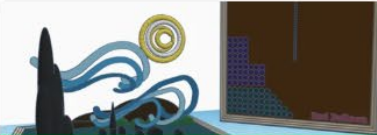
Challenges


Collections

+ Create collection

What's new


See which car won the Tinkercad Pinewood Derby Challenge! »


Take the new challenge to Reimagine Classic Art in 3D. »


Check out all of the fun designs in the gallery that you can Copy and Tinker! »

Your Classes

Teaching Archived Co-teaching Enrolled

☐ Create new class Actions ▾ Create date ▾

<input type="radio"/> Mavişehir LH	0 students	Created Jan 8, 2021	...
<input type="radio"/> Mavişehir 9-A	0 students	Created Jan 8, 2021	...
<input type="radio"/> Mavişehir 10-A	0 students	Created Jan 8, 2021	...
<input type="radio"/> Buca 10-A	0 students	Created Jan 8, 2021	...

Unassigned Students
Students who have not been assigned to a Class

0 students

TinkerCAD: Designs → Circuits

Your designs

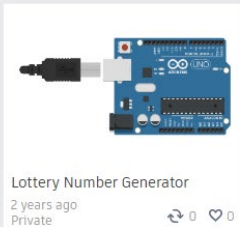
+ Create

3D Designs



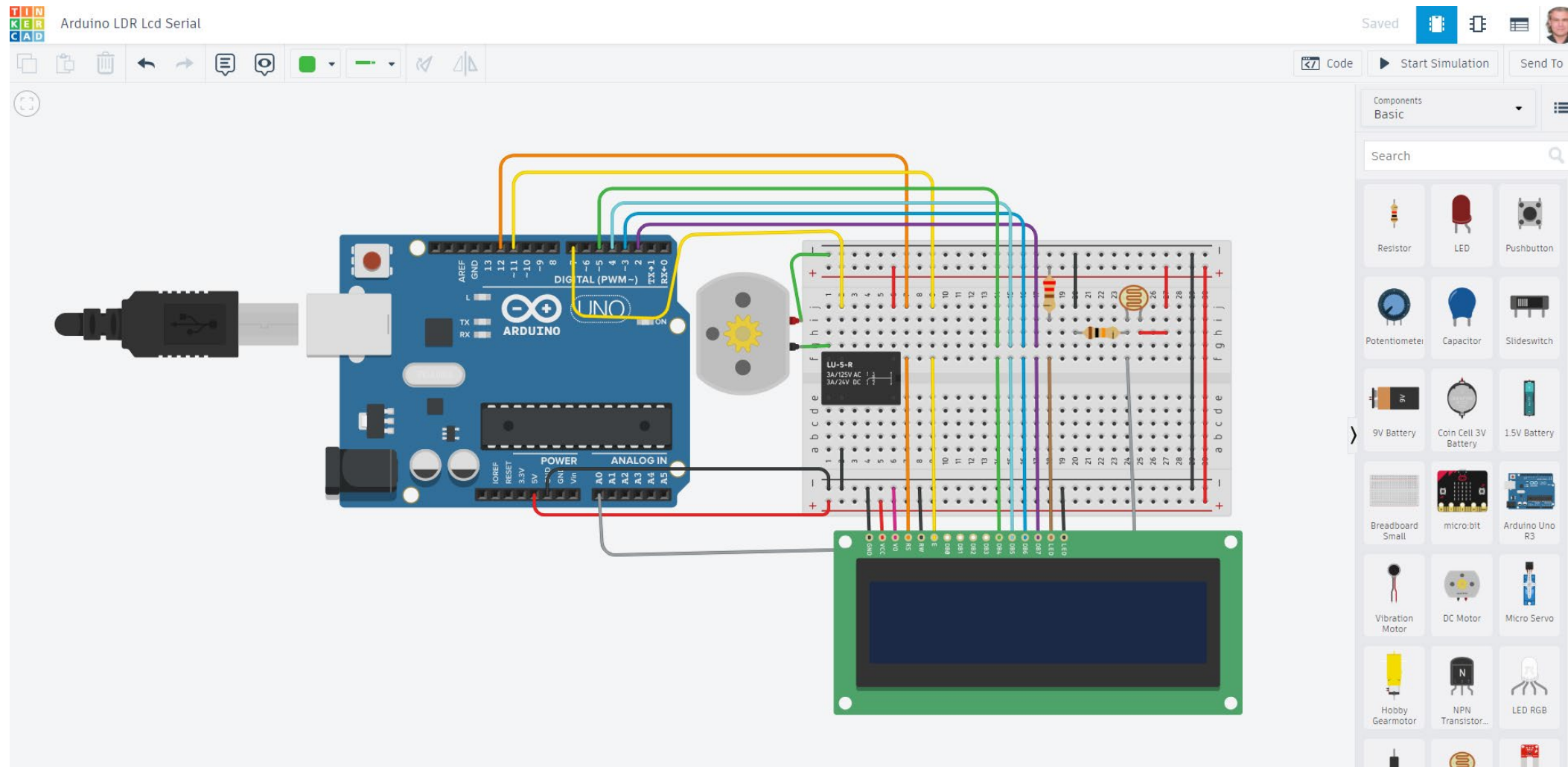
Circuits

View 26 more >

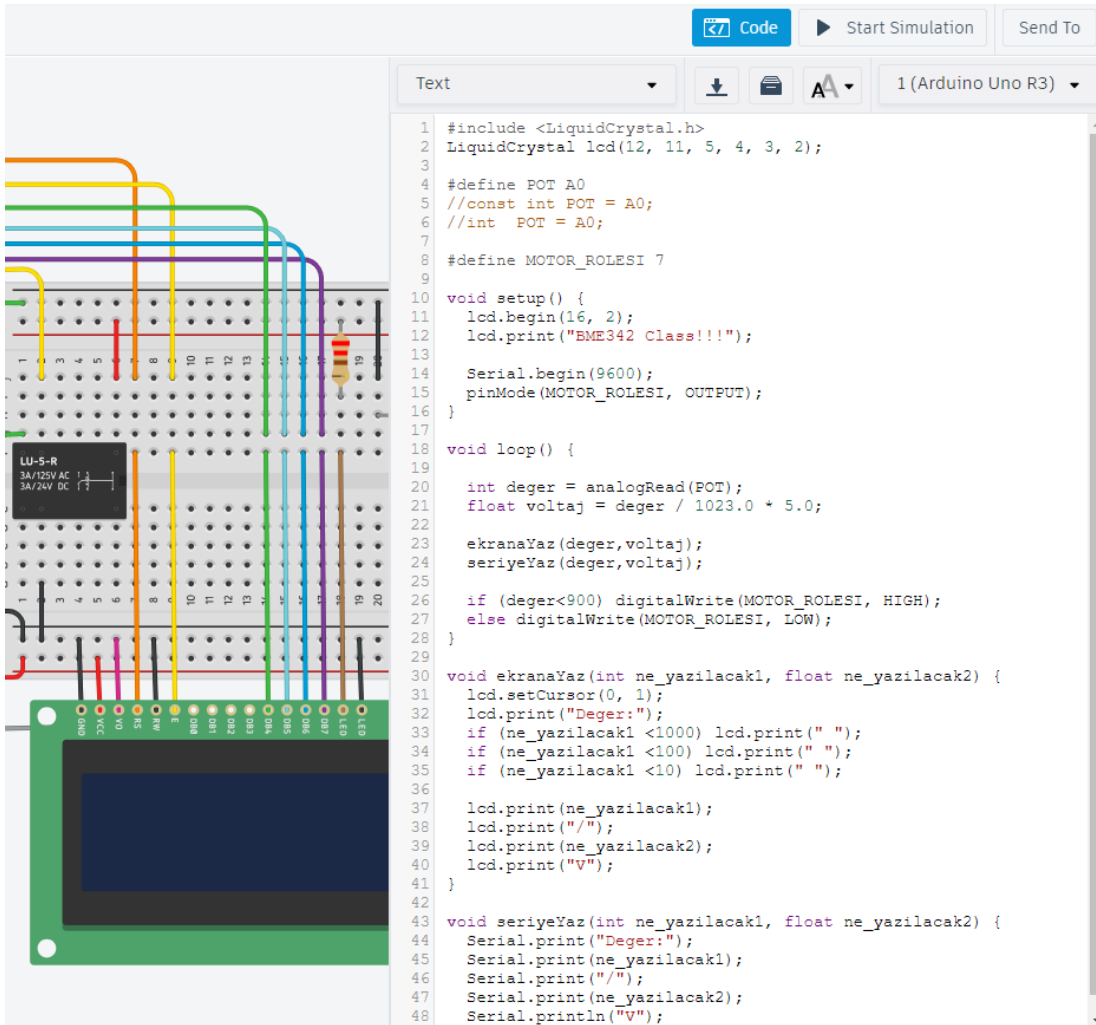


- Classes
- Designs
 - 3D Designs
 - **Circuits**
 - Codeblocks
- Tutorials
- Challenges
- Collections

TinkerCAD: Circuit Design Example

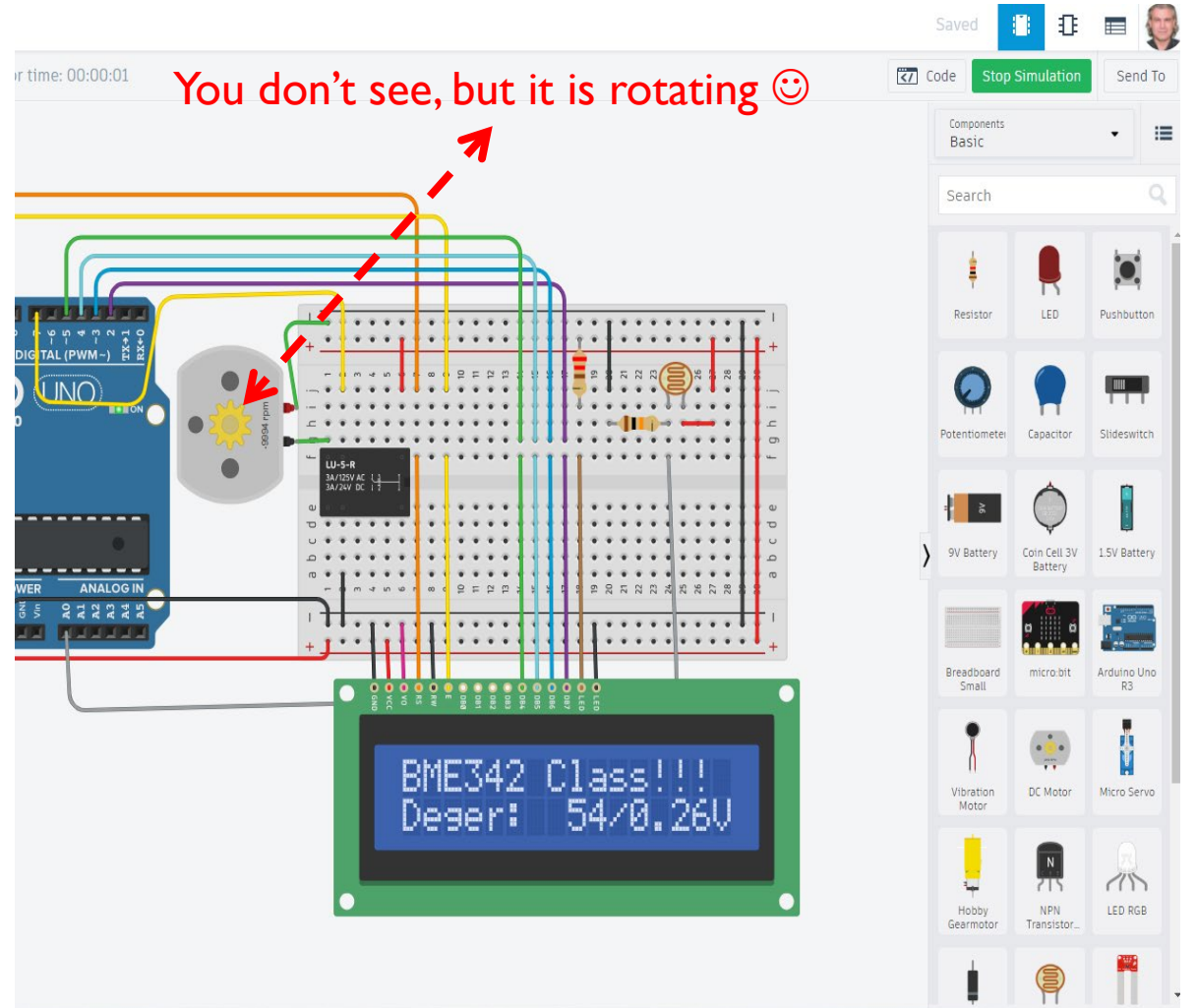


TinkerCAD: Code & Simulation



The screenshot shows the TinkerCAD interface with a breadboard circuit and the Arduino code. The breadboard contains an Arduino Uno R3, a BME342 sensor, a potentiometer, and a motor. The code is as follows:

```
1 #include <LiquidCrystal.h>
2 LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
3
4 #define POT A0
5 //const int POT = A0;
6 //int POT = A0;
7
8 #define MOTOR_ROLES1 7
9
10 void setup() {
11   lcd.begin(16, 2);
12   lcd.print("BME342 Class!!!");
13
14   Serial.begin(9600);
15   pinMode(MOTOR_ROLES1, OUTPUT);
16 }
17
18 void loop() {
19
20   int deger = analogRead(POT);
21   float voltaj = deger / 1023.0 * 5.0;
22
23   ekranaYaz(deger, voltaj);
24   seriyeYaz(deger, voltaj);
25
26   if (deger < 900) digitalWrite(MOTOR_ROLES1, HIGH);
27   else digitalWrite(MOTOR_ROLES1, LOW);
28 }
29
30 void ekranaYaz(int ne_yazilacak1, float ne_yazilacak2) {
31   lcd.setCursor(0, 1);
32   lcd.print("Deger:");
33   if (ne_yazilacak1 < 1000) lcd.print(" ");
34   if (ne_yazilacak1 < 100) lcd.print(" ");
35   if (ne_yazilacak1 < 10) lcd.print(" ");
36
37   lcd.print(ne_yazilacak1);
38   lcd.print("/");
39   lcd.print(ne_yazilacak2);
40   lcd.print("V");
41 }
42
43 void seriyeYaz(int ne_yazilacak1, float ne_yazilacak2) {
44   Serial.print("Deger:");
45   Serial.print(ne_yazilacak1);
46   Serial.print("/");
47   Serial.print(ne_yazilacak2);
48   Serial.println("V");
```



The screenshot shows the TinkerCAD interface with a breadboard circuit. The breadboard contains an Arduino Uno R3, a BME342 sensor, a potentiometer, and a motor. The motor is connected to the Arduino's digital output pin 7. The potentiometer is connected to the Arduino's analog input pin A0. The BME342 sensor is connected to the Arduino's I2C pins (SCL to 5, SDA to 4). The LCD is connected to the Arduino's I2C pins (SCL to 12, SDA to 11). The motor is shown rotating, indicated by a red arrow and the text "You don't see, but it is rotating 😊". The LCD displays "BME342 Class!!!" and "Deger: 54/0.26V".

Fritzing: <https://fritzing.org/>

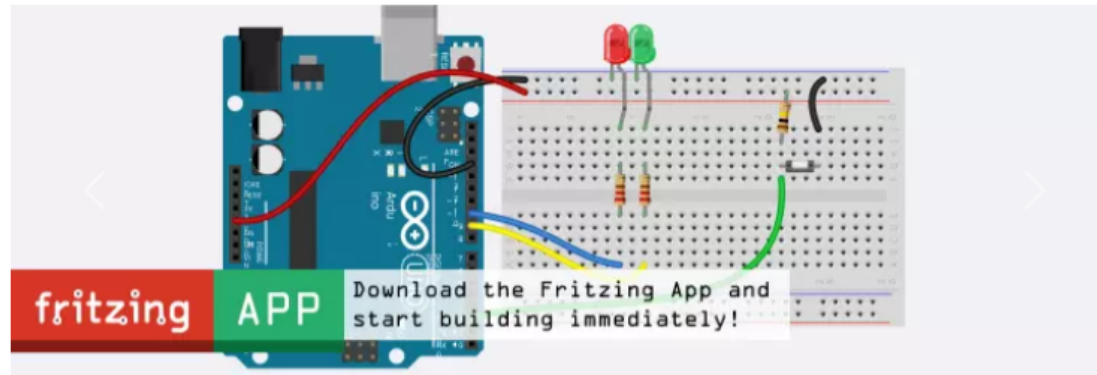
fritzing electronics
made easy

[Projects](#) [Parts](#) [Download](#) [Learning](#) [Services](#) [Contribute](#)

[FORUM](#)

[FAB](#)

[SIGN UP](#) [LOGIN](#)



Fritzing is an open-source hardware initiative that makes electronics accessible as a creative material for anyone. We offer a software tool, a community website and services in the spirit of Processing and Arduino, fostering a creative ecosystem that allows users to document their prototypes, share them with others, teach electronics in a classroom, and layout and manufacture professional PCBs.

Download and Start

Download our latest version, 1.0.2 of Fritzing, released on January 02, 2024 and start right away. Fritzing is available for Windows, macOS, and Linux.

Produce your own board

With Fritzing Fab you can easily and inexpensively turn your circuit into a real, custom-made PCB. Try it out now!

Participate

[GO](#)

[FAQ](#) [ABOUT](#) [CONTACT](#)

Blog

[Fritzing 1.0.2 released](#)
January 02, 2024

[Fritzing 1.0.1 released](#)
September 06, 2023

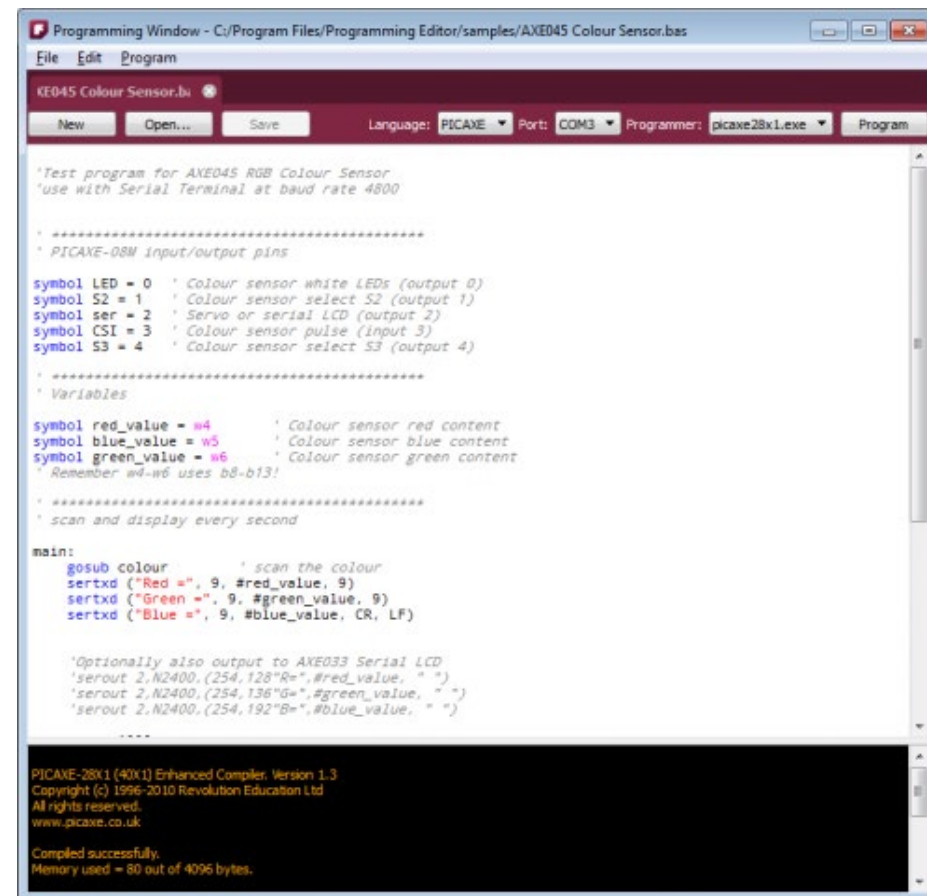
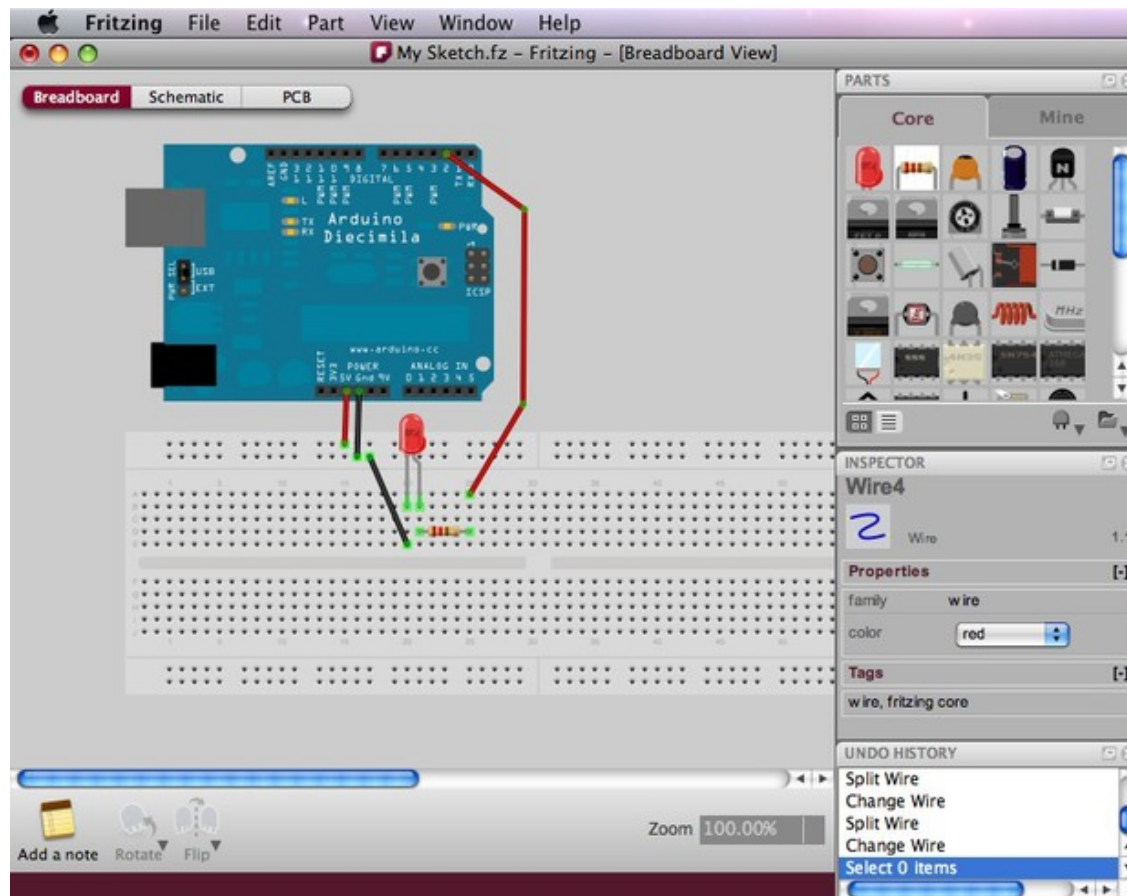
[Fritzing 1.0.0 released](#)
June 15, 2023

[Simulating Circuits with Fritzing](#)
June 27, 2022

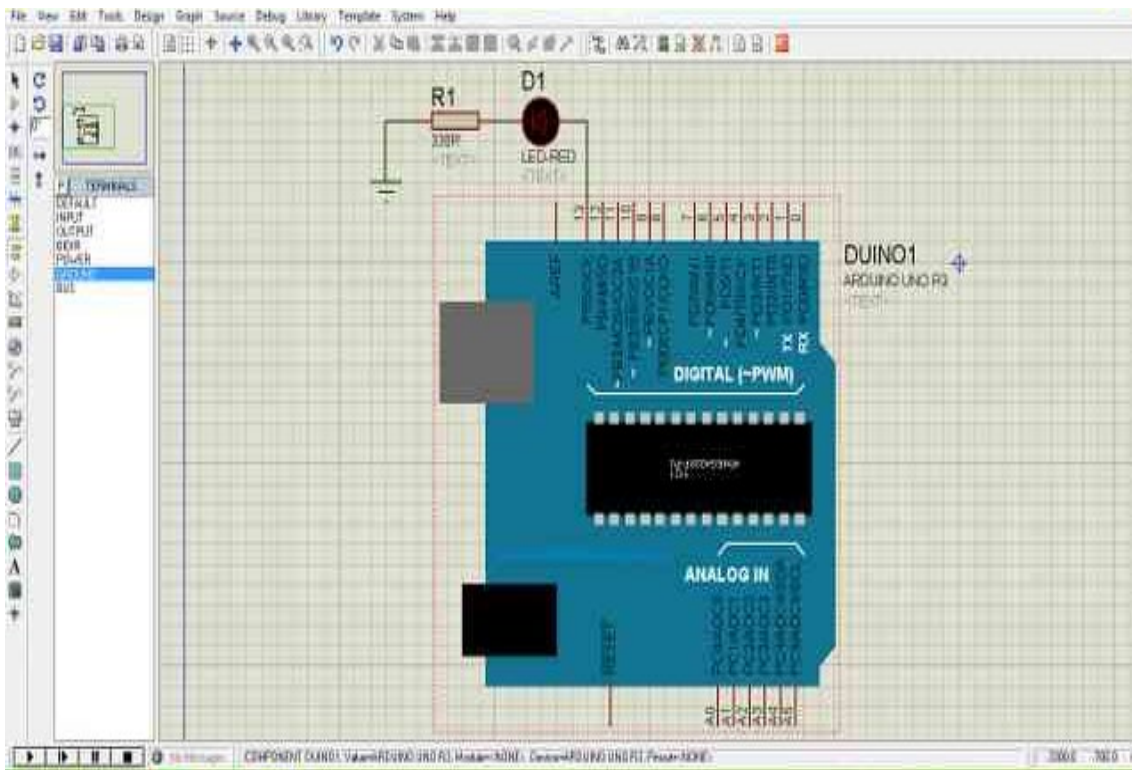
[Fritzing 0.9.10 released](#)
May 22, 2022

[More posts...](#)

Fritzing: Design and Coding



PROTEUS ISIS



- Commercial only !
- A bit complicated.
- More professional.
- How to? :
 - Make selected the options of «Compilation» and «Upload» in the section of File→Preferences in Arduino IDE.
 - Code and compile your program in Arduino IDE.
 - Double click Arduino symbol in Proteus ISIS.
 - Write compiled HEX file full folder and filename into the «Program File» field in Proteus ISIS. Then, click the OK button.
 - Now, you can run the simulation in the main screen in Proteus ISIS.



Thanks for
listening 😊

YALÇIN İŞLER

Assoc. Prof.