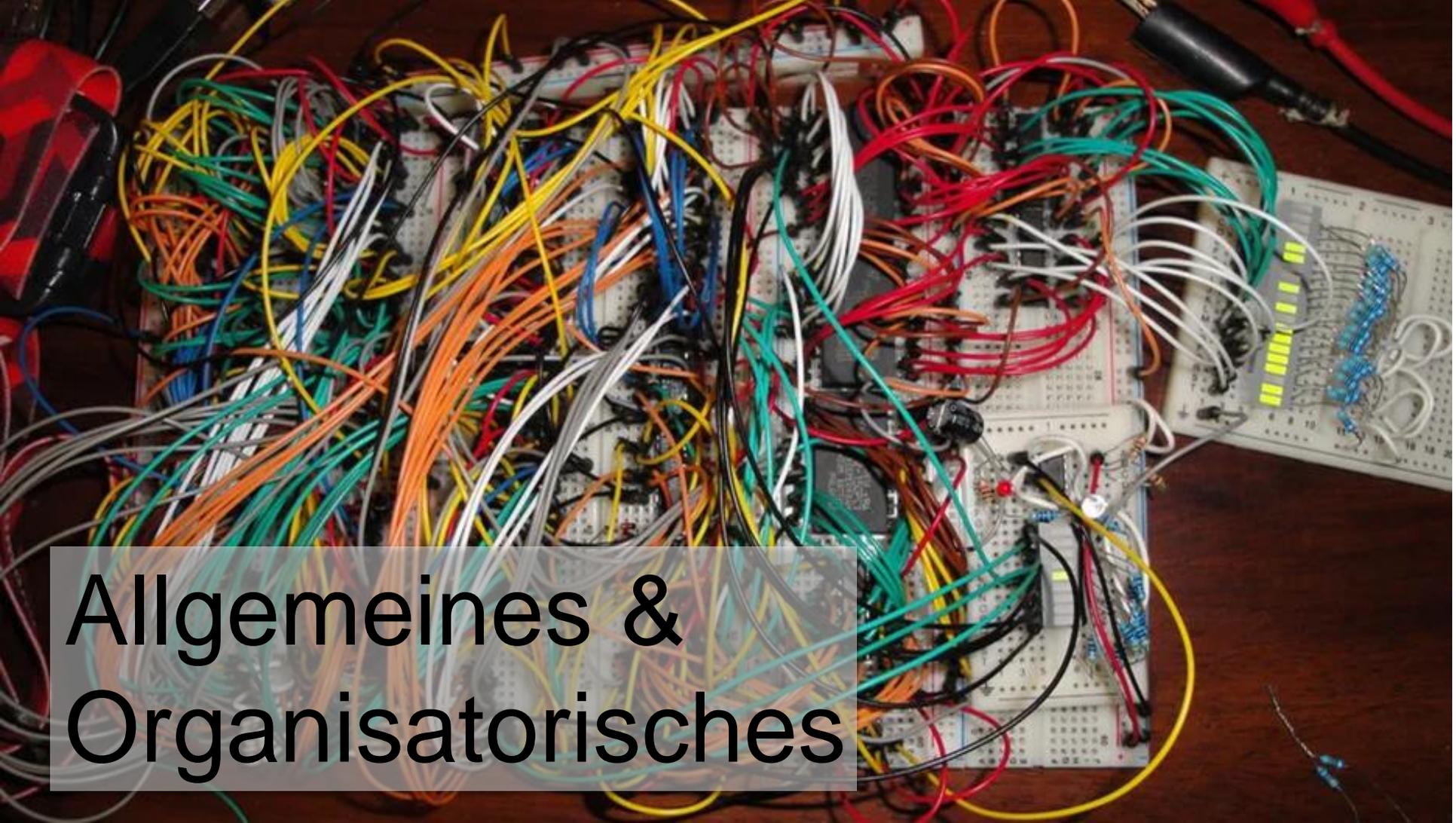


Übungen zur TI

1. Übung



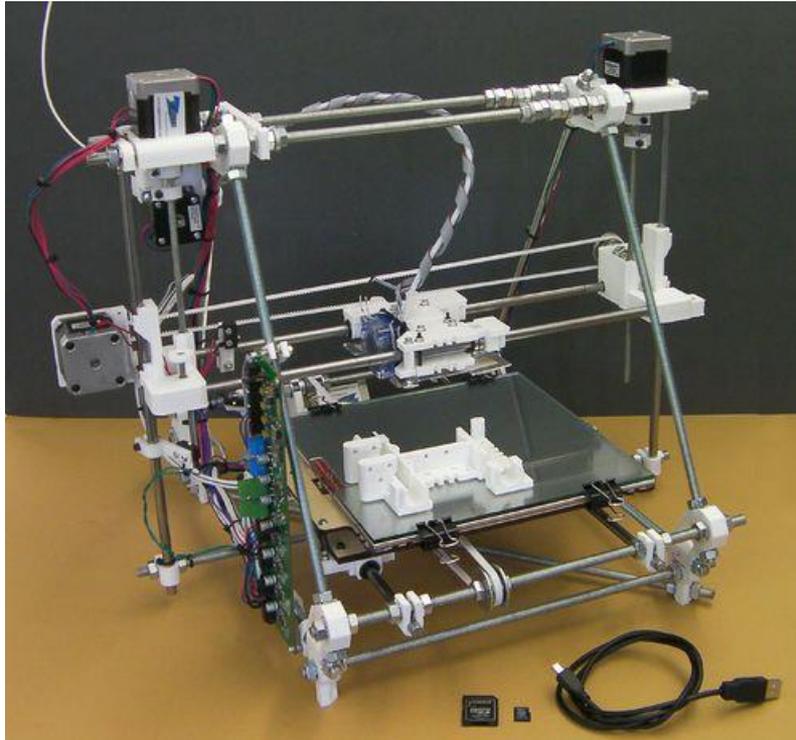
Allgemeines & Organisatorisches

Allgemeines & Organisatorisches

- Übungen müssen dokumentiert werden
- Nachweis muss erbracht werden (80%; Zulassungsvoraussetzung für die Prüfung)
- Einreichung der Nachweise erfolgt wöchentlich über TinkerCAD
- Prüfung erfolgt in Form einer Klausur
(insofern die Zulassungsvoraussetzungen erfüllt sind)

Populäre Arduino Projekte

- 3D printers (reprap.org)



Populäre Arduino Projekte

- Dronen ([satshacopter-250X](#))



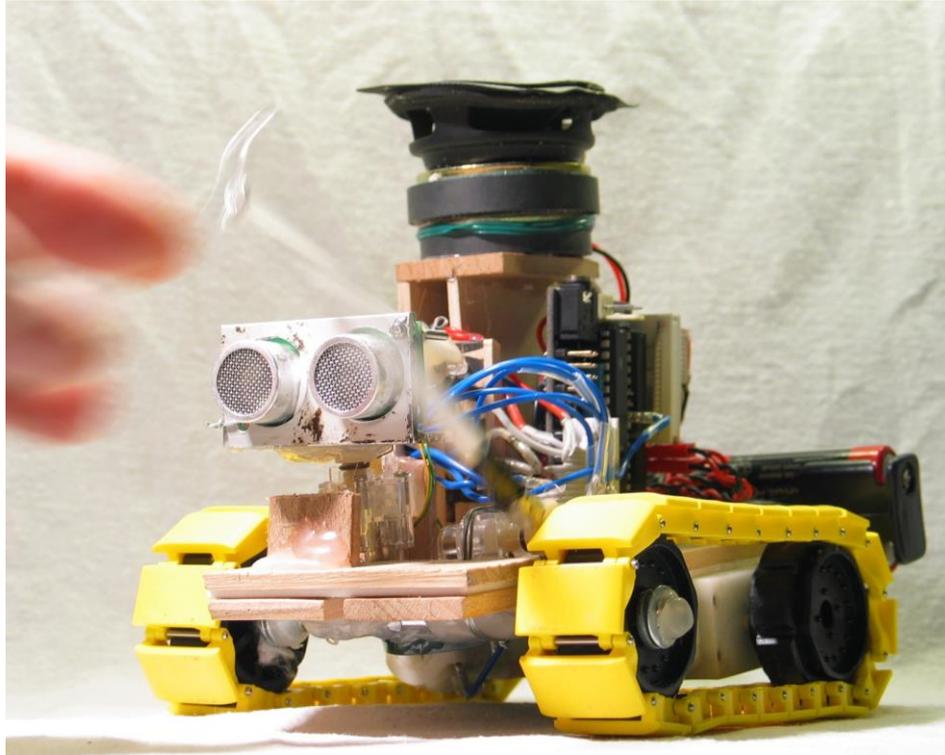
Populäre Arduino Projekte

[Punch Activated Arm Flamethrowers](#)



Populäre Arduino Projekte

- [Yellow Drum Machine by Frits Lyneborg](#)



Populäre Arduino Projekte

- [OpenCat](#), [Petoï Bittle](#)



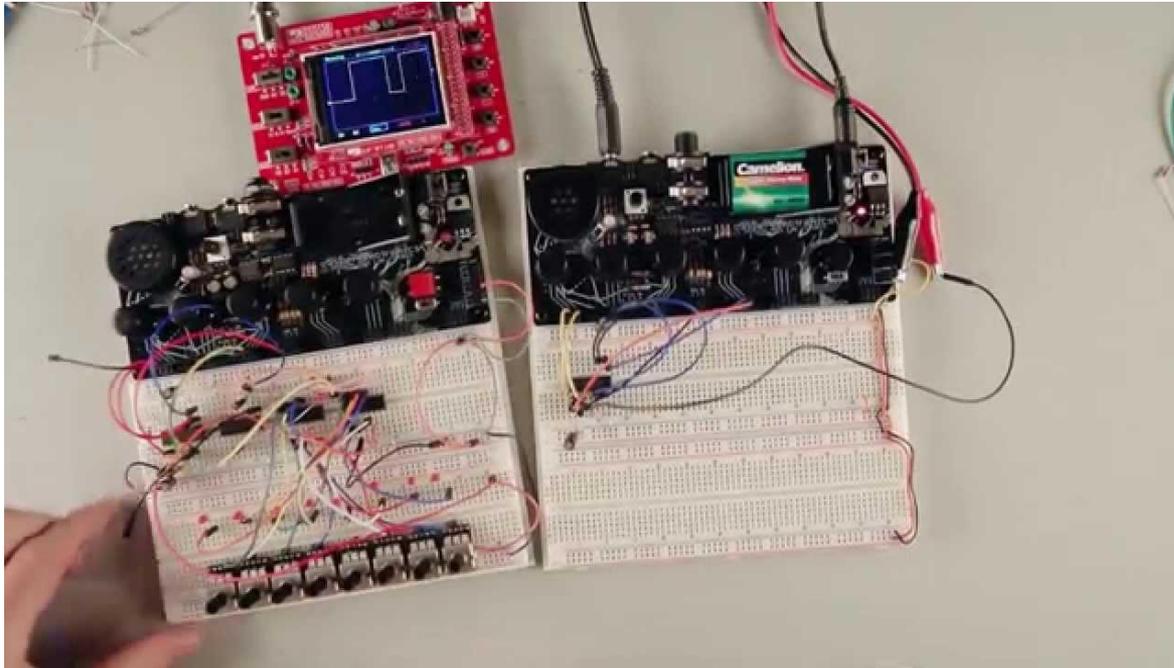
Populäre Arduino Projekte

- <https://www.ottodiy.com>



Populäre Arduino Projekte

- [Casper Electronics DIY synth building](#)



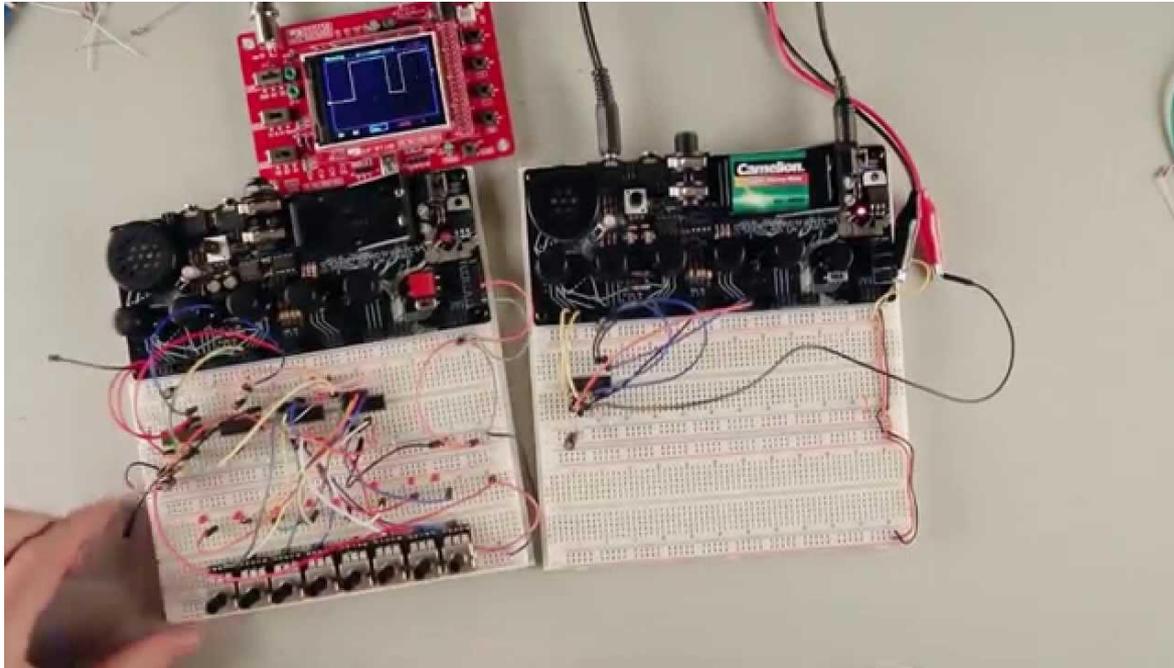
Populäre Arduino Projekte

- <https://smartcitizen.me/>



Populäre Arduino Projekte

- [Casper Electronics DIY synth building](#)



Blink



Youtube: Infinity Dodecahedron

<https://jonathanindetzki.gitlab.io/digital-fabrication/#/tasks/final>

Opensource Frameworks

- [Openframeworks](#)
- [Cinder](#)
- [Processing](#)
- [PureData](#)
- [Vvvv](#)



Arduino “Kit”

Allgemeines & Organisatorisches

- Übungen finden in TinkerCAD statt
- <https://www.tinkercad.com/joinclass/U371LVL594QZ>

Material Kits for Technischenformatik

Menge		Name	Datasheet	Kommentar	Anzahl
1		Arduino Nano	Datasheet	link	1
1		Arduino USB cable	-		1
1		Breadboard Big	Datasheet	link	2
20		Jumper cable MM	-	link	20
20		Jumper cable MF	-	link	20
20		Jumper cable FF	-	link	20
		220Ohm Widerstände (rot - rot - braun)			
		10kOhm Widerstände (braun - schwarz - orange)			

	18650 Li-ion		link	
	18650 Batteriehalter		link	
	Lipo Charger USB mini		link	
	Input			
	LDR			1*
	Potentiometer 10k	Datasheet	link	1
	Buttons for the breadboard			2
	Output			
	LEDs through hole		link	2
	RGB LED		link	
	Speaker/buzzer	Datasheet	link	1

	Transistor BC547		link	1
	DC motor		link	1
	Servo motor small/big			1

https://www.tinkercad.com

→ Schaltkreise → Neuen Schaltkreis erstellen



Klassen Katalog **Blog** Lernen Lehren Q



feadi

Entwürfe durchsuchen...

3D-Entwürfe

Schaltkreise

Codeblöcke

NEU

Lektionen

Deine Klassen

Projekte

Project 1

Projekt erstellen



[Tinkercad Lesson Plans](#)

Tinkercad lesson plans are ready to use online or in the classroom. Discover curriculum developed in partnership with teachers. [Learn more](#)

Meine zuletzt verwendeten Entwürfe

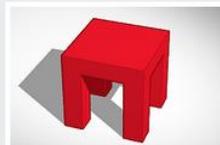
Neuen Entwurf erstellen

Select



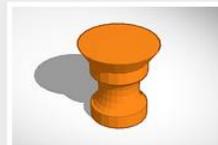
Powerful Wolt

vor 4 Monaten Privat



Surprising Tumelo-Bru...

vor 4 Monaten Privat



Grand Borwo-Densor

vor 4 Monaten Privat



Brave Trug-Hango

vor 4 Monaten Privat



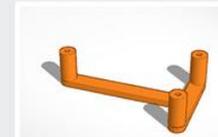
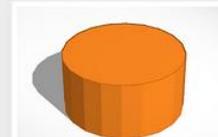
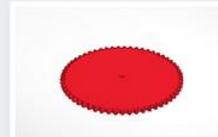
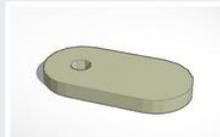
Neat Wolt

vor 4 Monaten Privat



Brilliant Blad

vor 4 Monaten Privat

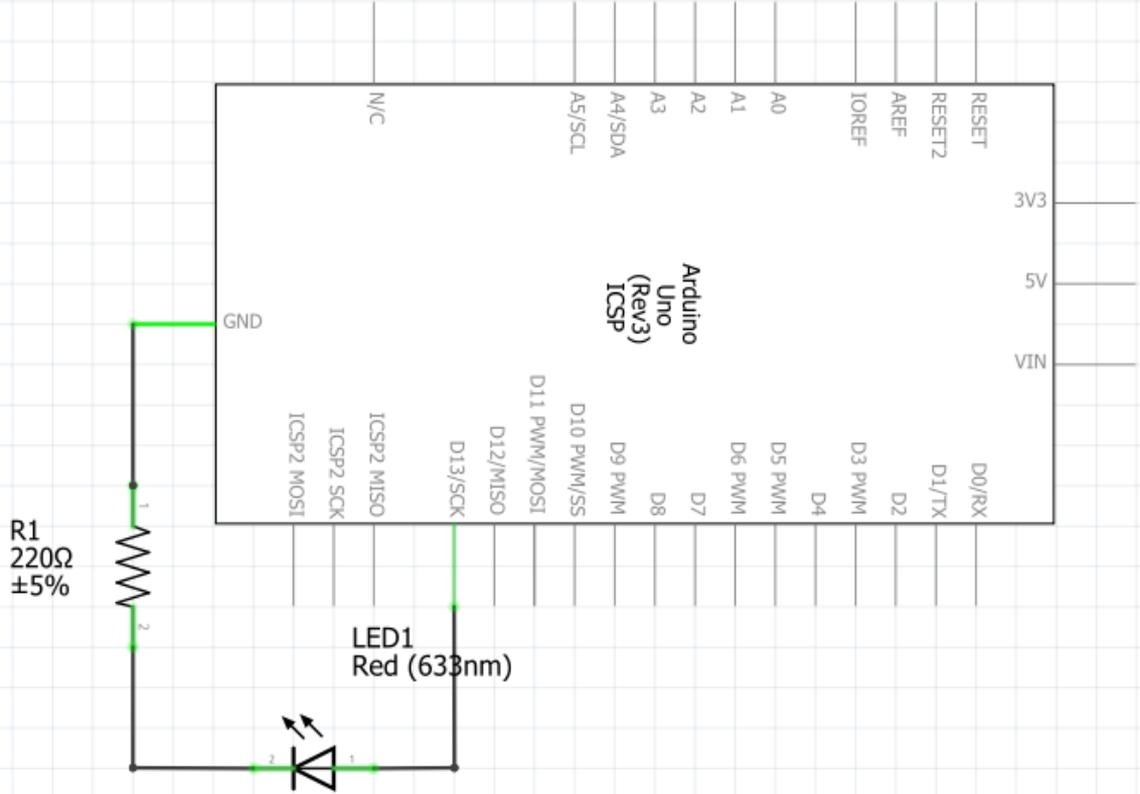


<https://www.tinkercad.com>

→ Circuits → Neuen Schaltkreis erstellen

The screenshot displays the Tinkercad web interface. At the top left, there is a navigation bar with the text "Getting Started Start Simulating" and a "Start Simulating" button. The main workspace shows a simulated circuit consisting of an Arduino Uno board connected to a breadboard. A USB cable is plugged into the Arduino's Type-B port. The breadboard has a red wire connected to the positive terminal and a black wire connected to the negative terminal. On the right side, there is a component library panel titled "Komponenten Einfach" with a search bar and icons for various components: Widerstand (Resistor), LED, Drucktaster (Push Button), Potentiometer, Kondensator (Capacitor), and Schiebeshalter (Slide Switch). The top right corner shows a "Gespeichert" (Saved) status and a user profile icon.

Blink



Blink

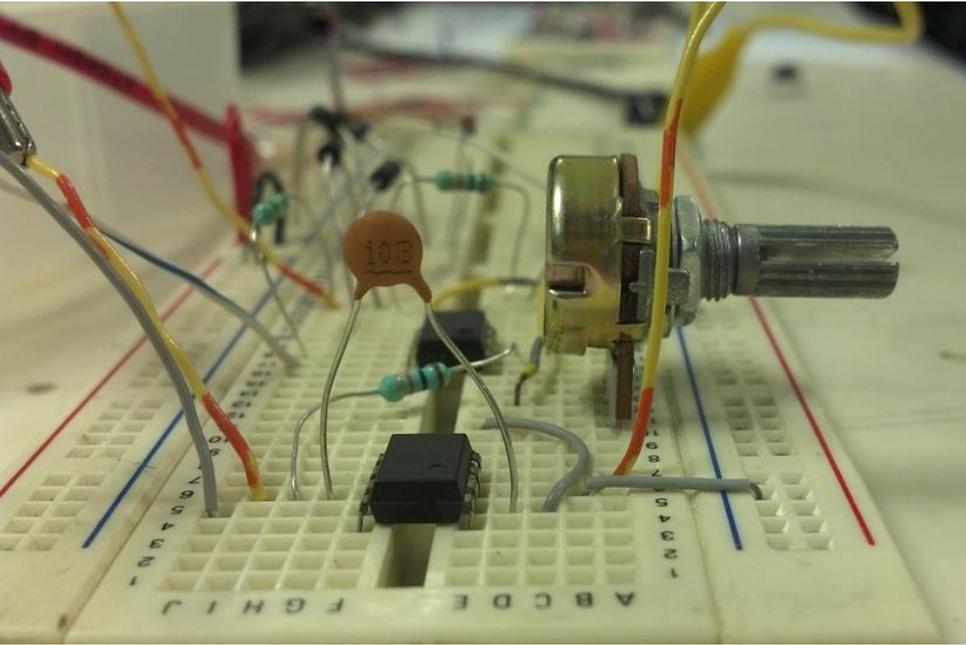
The image shows a digital simulation of an Arduino Uno R3 board connected to a breadboard. A red LED is connected to digital pin 13 and ground. The code in the IDE is as follows:

```
1 void setup()
2 {
3   pinMode(13, OUTPUT);
4 }
5
6 void loop()
7 {
8   digitalWrite(13, HIGH);
9   delay(1000); // Wait for 1000 millisecond(s)
10  digitalWrite(13, LOW);
11  delay(1000); // Wait for 1000 millisecond(s)
12 }
```

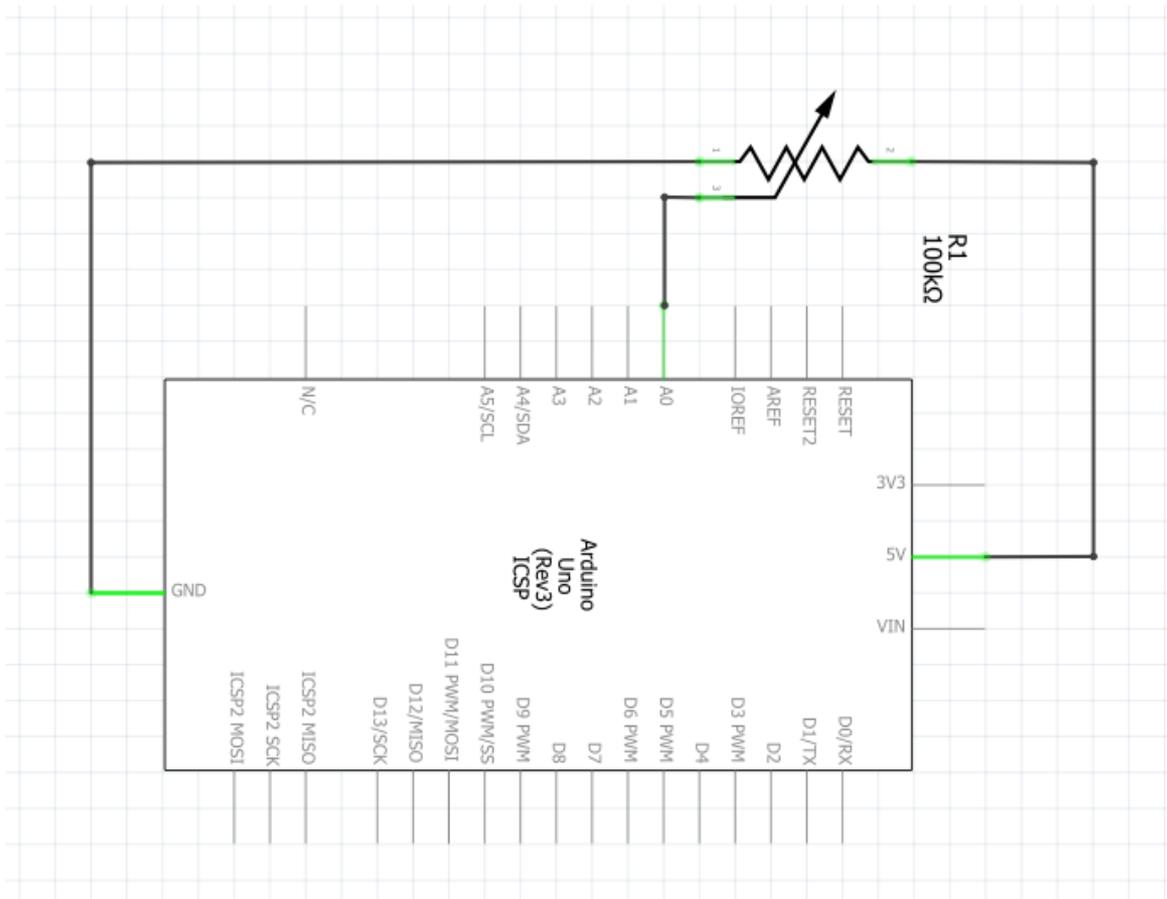
The interface includes a toolbar at the top with icons for file operations and simulation controls, and a 'Code' button. The breadboard shows the LED connected to digital pin 13 and ground.

Serial-Monitor

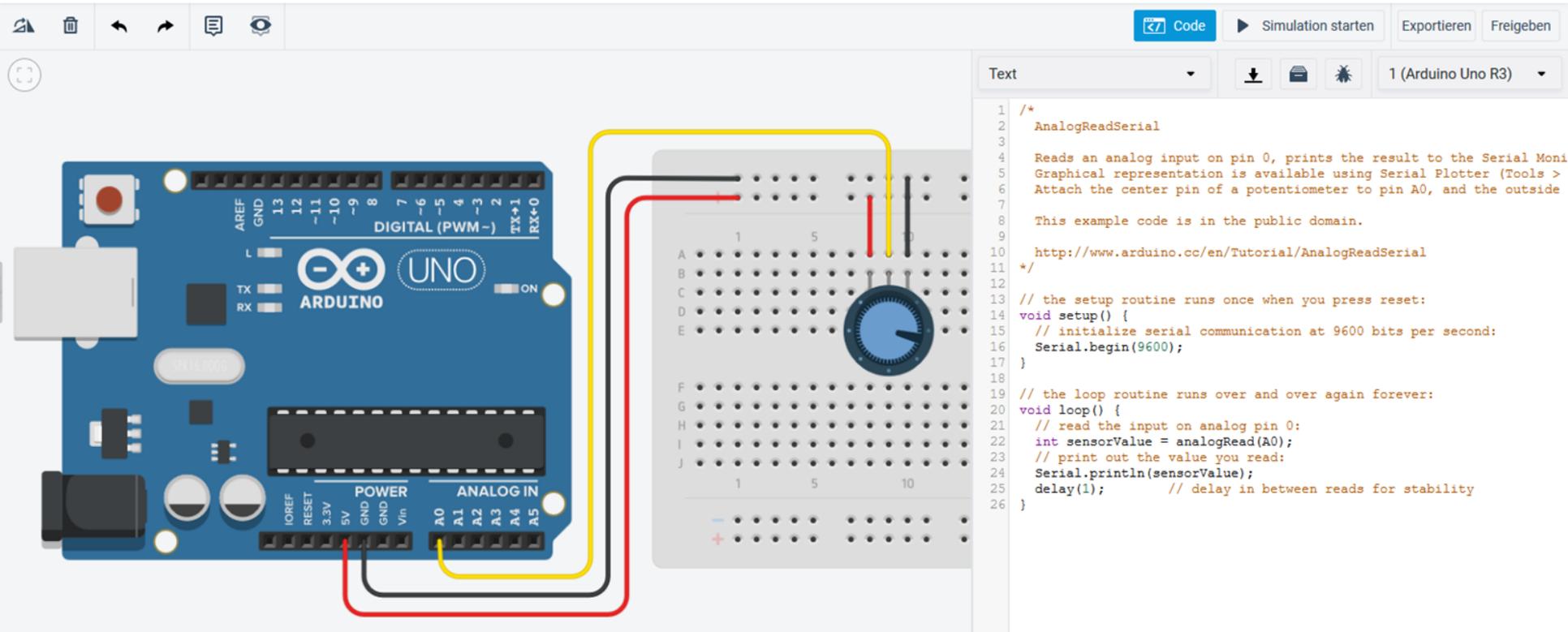
AnalogReadSerial



AnalogReadSerial



AnalogReadSerial

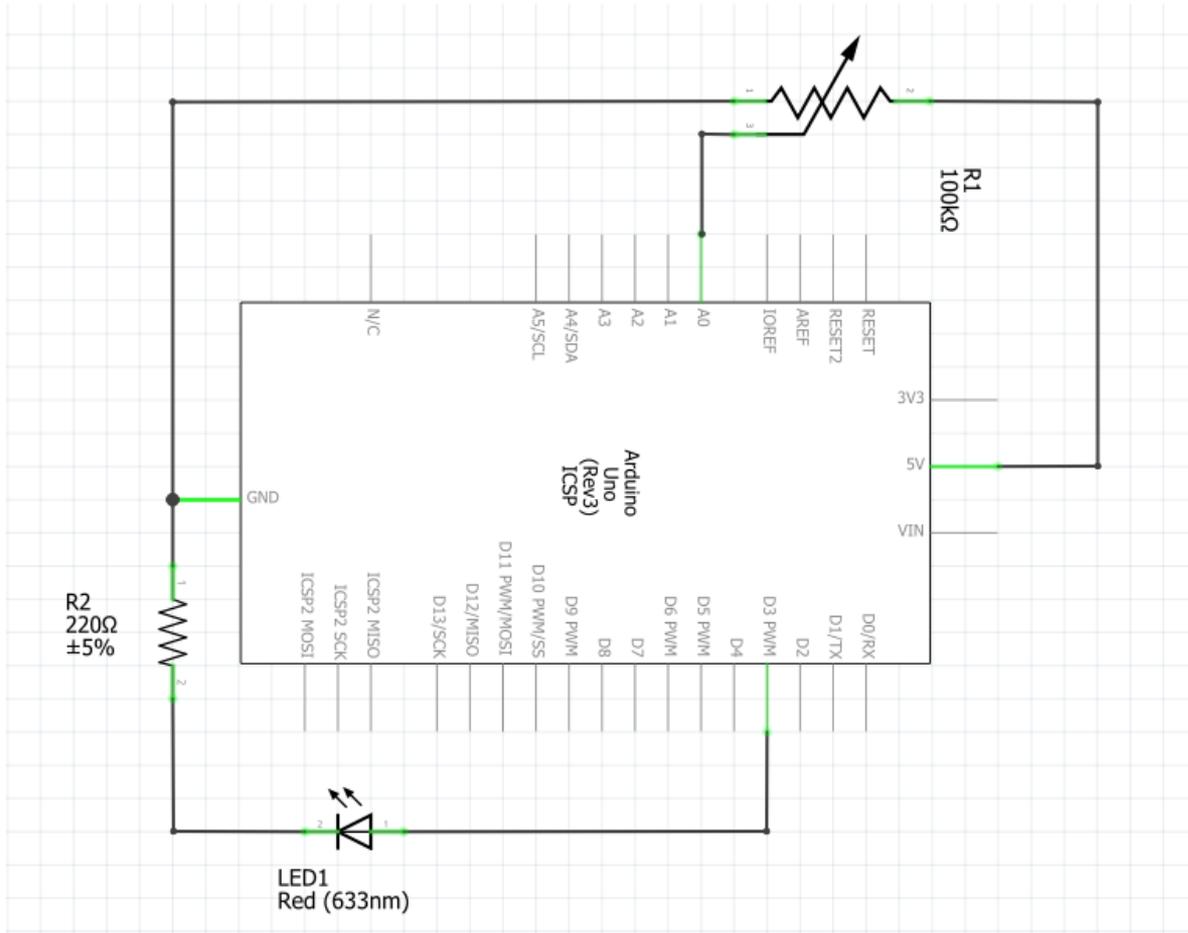


The image shows a screenshot of the Arduino IDE interface. On the left, a blue Arduino Uno board is connected to a breadboard. A potentiometer is mounted on the breadboard. Wires connect the potentiometer's center pin to Arduino pin A0, and its outer pins to ground and the 5V supply. The IDE window on the right displays the following C++ code:

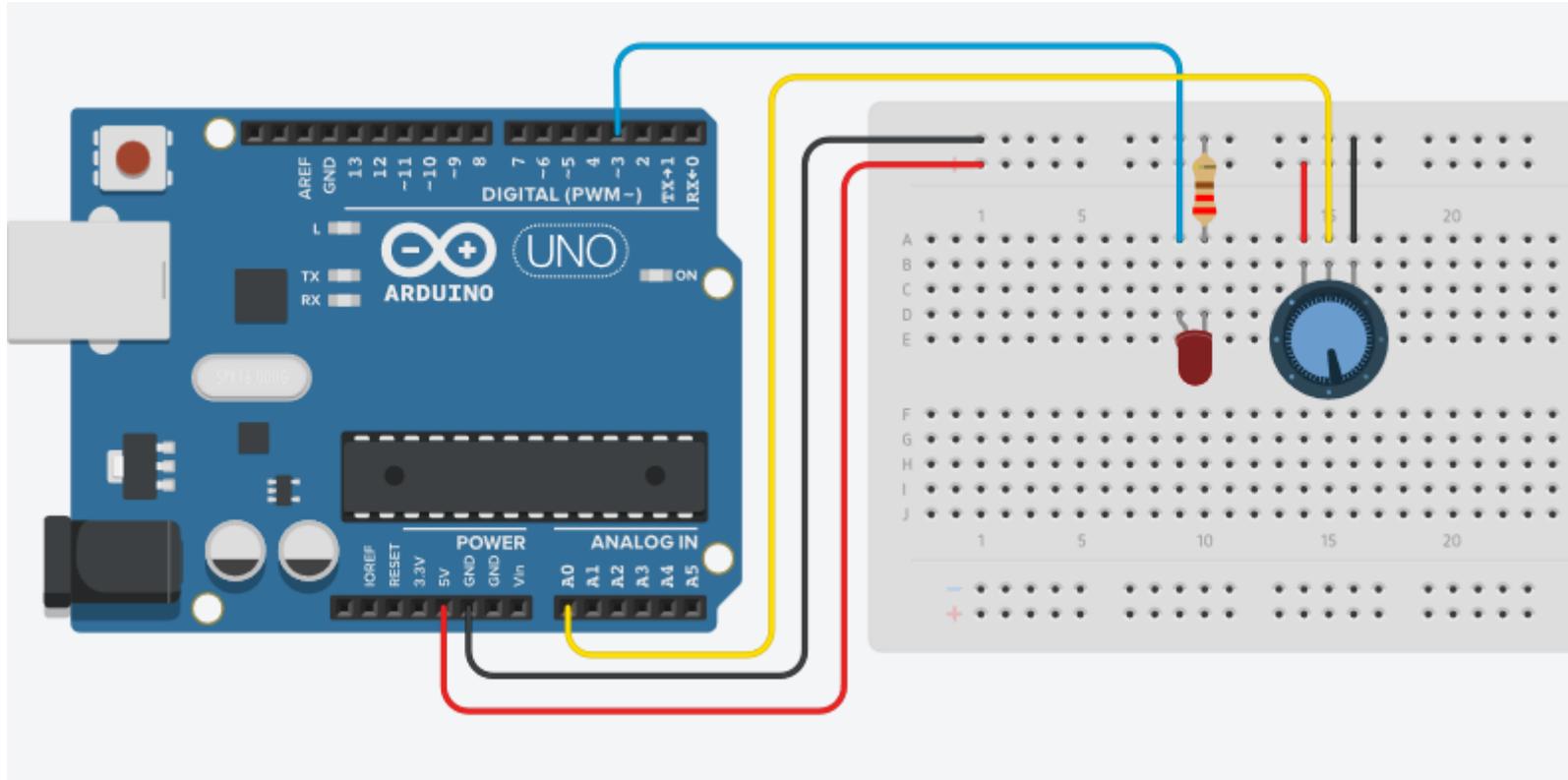
```
1 /*
2  AnalogReadSerial
3
4  Reads an analog input on pin 0, prints the result to the Serial Monitor.
5  Graphical representation is available using Serial Plotter (Tools >
6  Attach the center pin of a potentiometer to pin A0, and the outside
7
8  This example code is in the public domain.
9
10 http://www.arduino.cc/en/Tutorial/AnalogReadSerial
11 */
12
13 // the setup routine runs once when you press reset:
14 void setup() {
15   // initialize serial communication at 9600 bits per second:
16   Serial.begin(9600);
17 }
18
19 // the loop routine runs over and over again forever:
20 void loop() {
21   // read the input on analog pin 0:
22   int sensorValue = analogRead(A0);
23   // print out the value you read:
24   Serial.println(sensorValue);
25   delay(1);        // delay in between reads for stability
26 }
```

Sie können den Quellcode auf dieser Seite finden:
<https://www.arduino.cc/en/Tutorial/AnalogReadSerial>

AnalogInOutSerial



AnalogInOutSerial



Sie können den Quellcode auf dieser Seite finden:
<https://www.arduino.cc/en/Tutorial/AnalogInOutSerial>

AnalogInOutSerial

Über Potentiometer einstellbare Blinkgeschwindigkeit

```
1 |const int analogInPin = A0; // Analog input pin that the potentiometer is attached to
2 |const int analogOutPin = 3; // Analog output pin that the LED is attached to
3 |
4 |int sensorValue = 0; // value read from the pot
5 |int outputValue = 0; // value output to the PWM (analog out)
6 |
7 |void setup() {
8 |    // initialize serial communications at 9600 bps:
9 |    Serial.begin(9600);
10|}
11|
12|void loop() {
13|    // read the analog in value:
14|    sensorValue = analogRead(analogInPin);
15|    // change the blink delay
16|    digitalWrite(3, HIGH); // turn the LED on (HIGH is the voltage level)
17|    delay(sensorValue); // wait for a second
18|    digitalWrite(3, LOW); // turn the LED off by making the voltage LOW
19|    delay(sensorValue); // wait for a second
20|
21|    // print the results to the Serial Monitor:
22|    Serial.print("sensor = ");
23|    Serial.println(sensorValue);
24|    // wait 2 milliseconds before the next loop for the analog-to-digital
25|    // converter to settle after the last reading:
26|    delay(2);
27|}
```

Herausforderung:

zwei LEDs

mit

zwei Potentiometern

die

mit *unabhängig einstellbarer* Geschwindigkeit

blinken.

2ledBlink

Vielen Dank fürs Mitmachen!

Bis nächste/übernächste Woche!

