SAE2019

Semana Acadêmica das Engenharias
ULBRA Canoas 2019

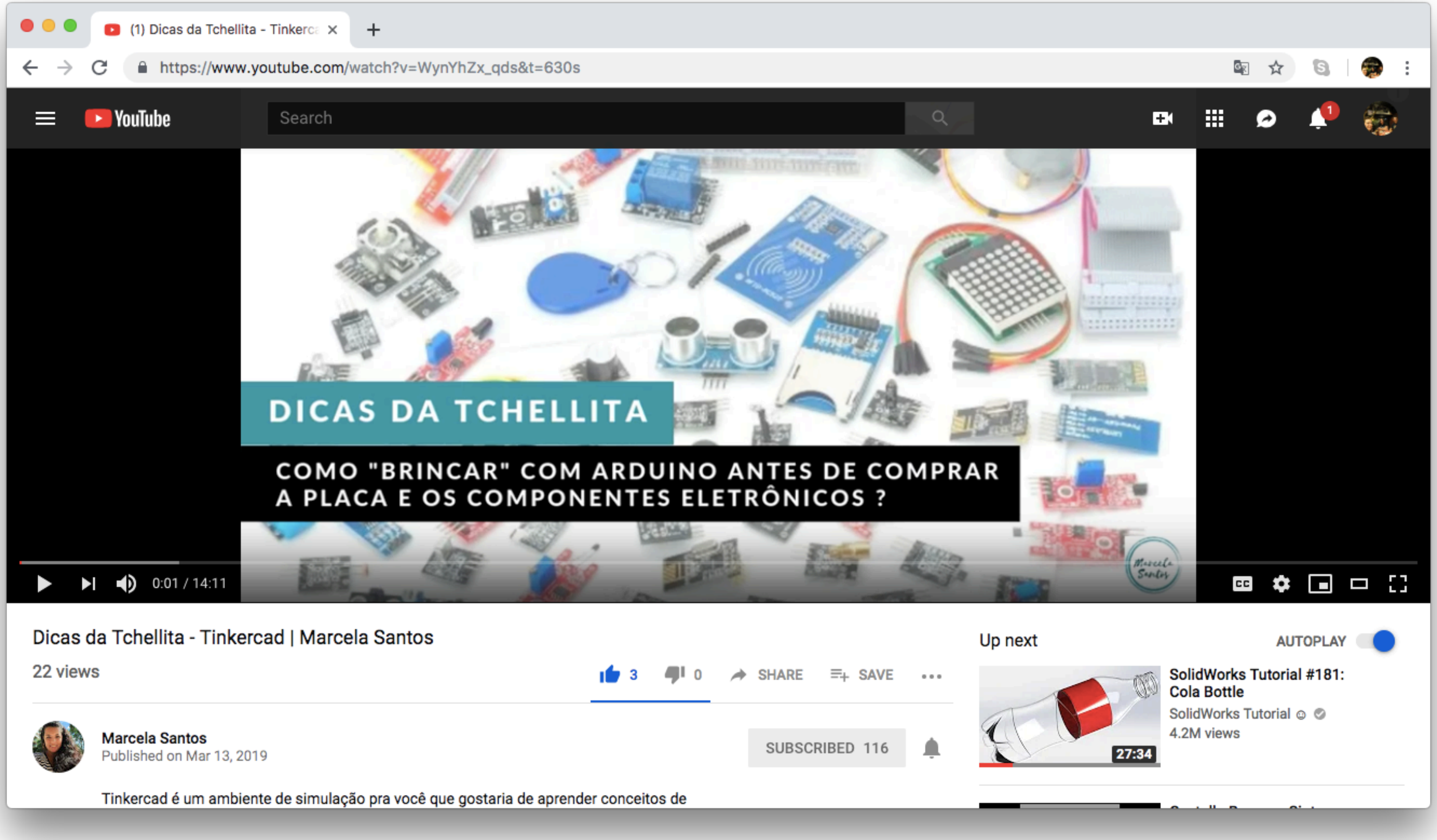
Simulação de circuitos com Tinkercad

Felipe Kühne
fkuhne@gmail.com

Maio/2019

github.com/fkuhne/sae2019

https://youtu.be/WynYhZx_qds



(1) Dicas da Tchellita - Tinkerc

https://www.youtube.com/watch?v=WynYhZx_qds&t=630s

DICAS DA TCHELLITA

COMO "BRINCAR" COM ARDUINO ANTES DE COMPRAR
A PLACA E OS COMPONENTES ELETRÔNICOS ?

Dicas da Tchellita - Tinkercad | Marcela Santos

22 views

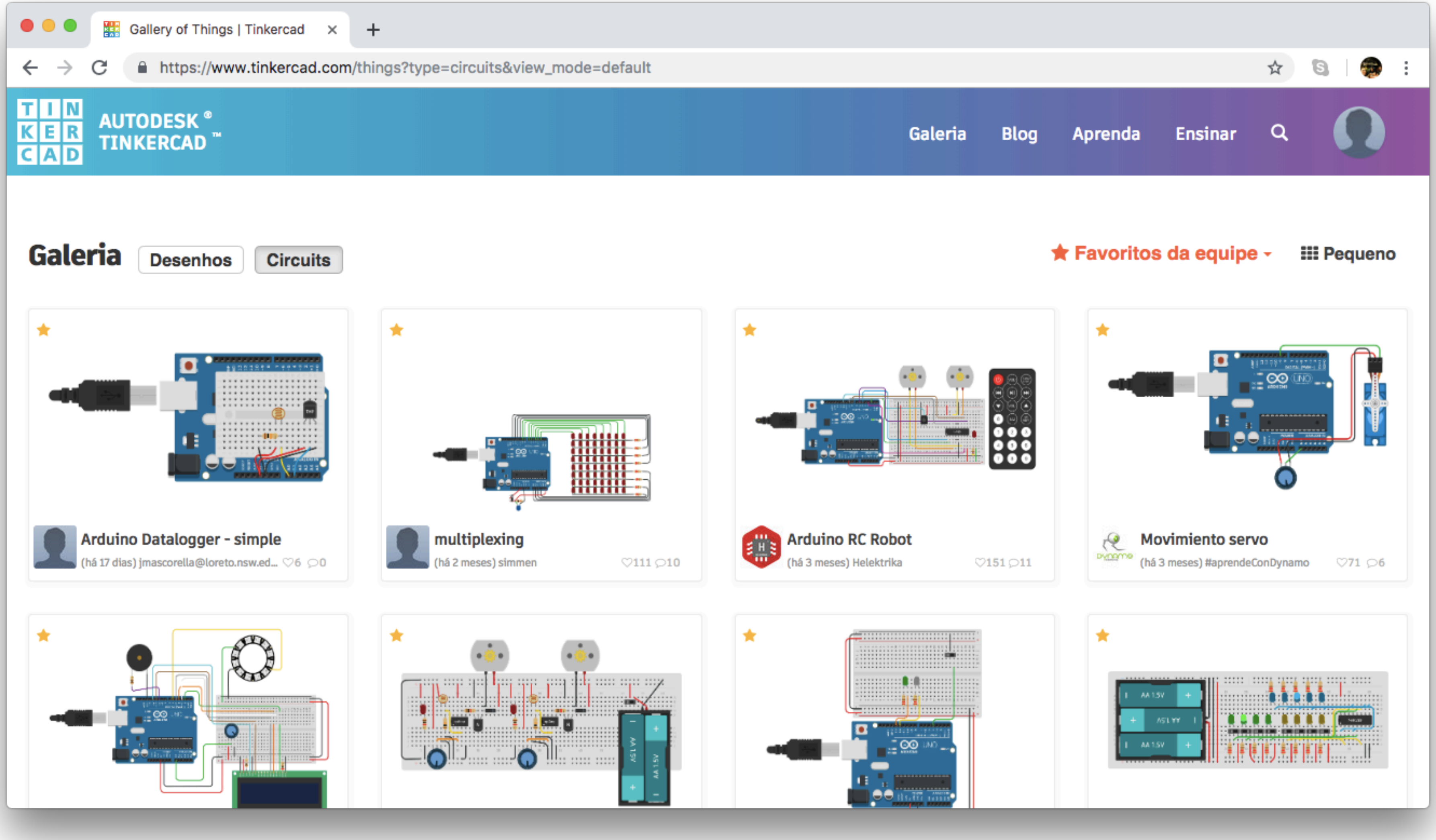
Marcela Santos
Published on Mar 13, 2019

Tinkercad é um ambiente de simulação pra você que gostaria de aprender conceitos de

Up next

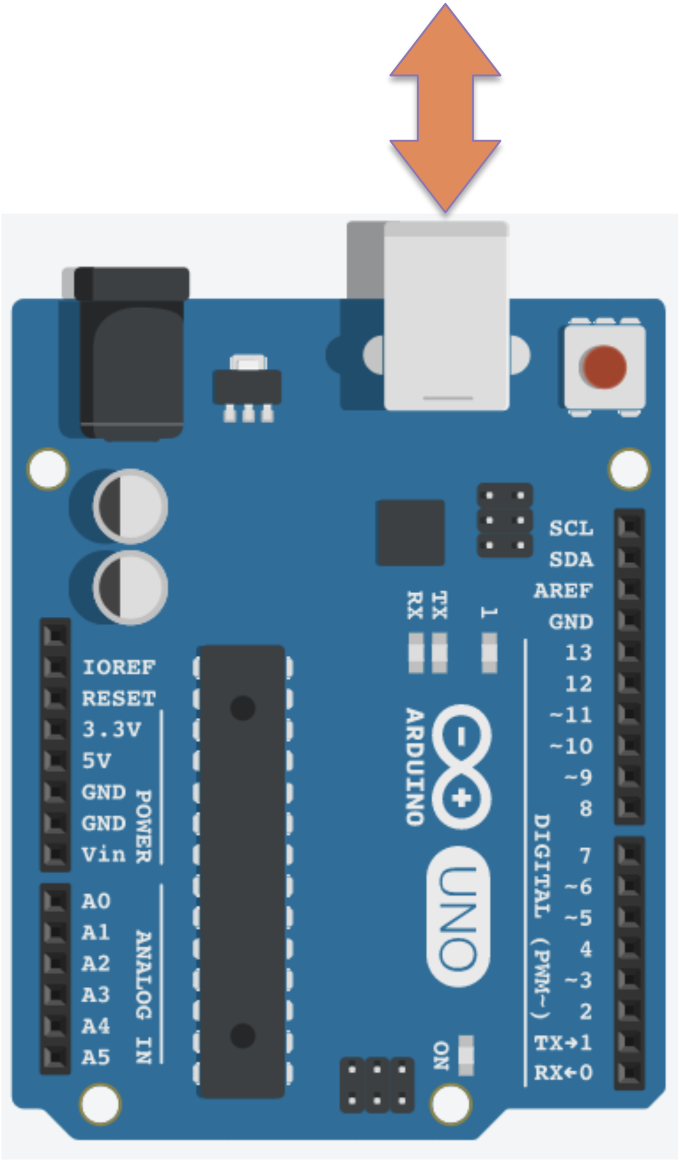
SolidWorks Tutorial #181: Cola Bottle
SolidWorks Tutorial
4.2M views

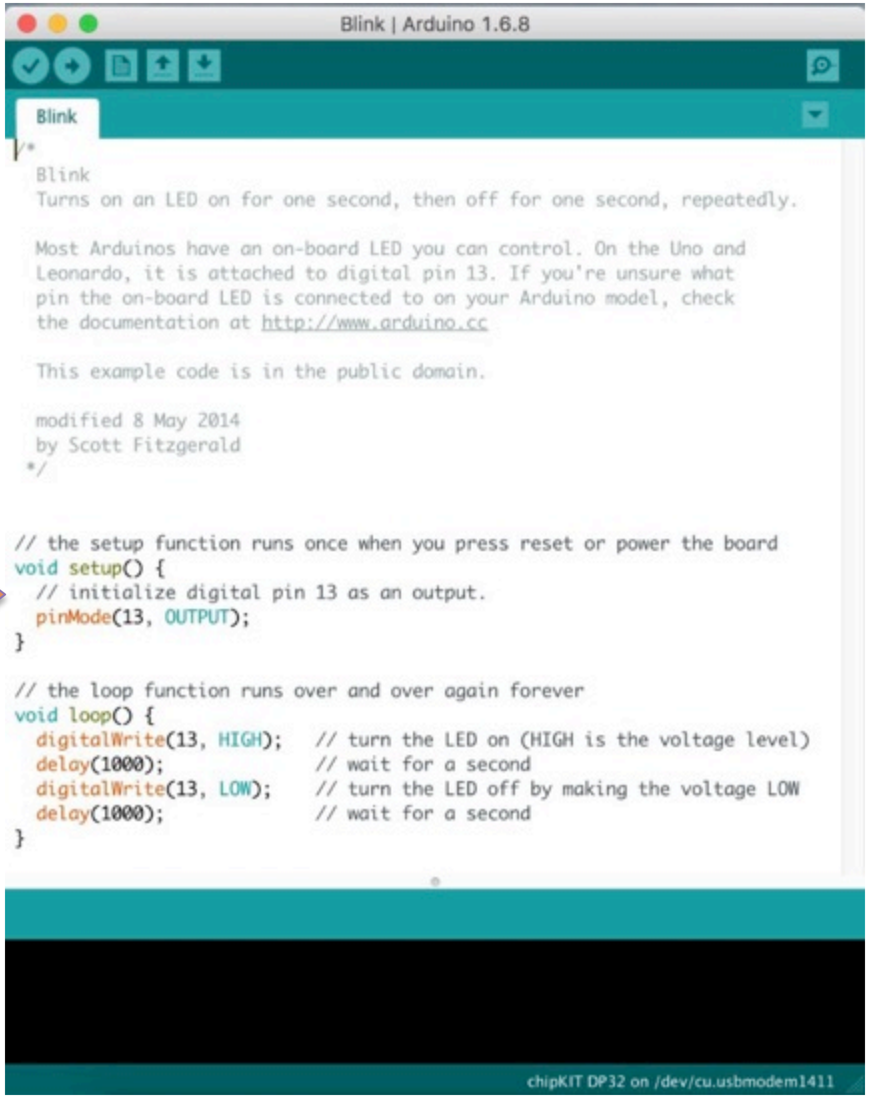
https://www.tinkercad.com


Gallery of Things | Tinkercad
https://www.tinkercad.com/things?type=circuits&view_mode=default
TIN KER CAD
AUTODESK®
TINKERCAD™
Galeria
Blog
Aprenda
Ensinar
Galeria
Desenhos
Circuits
Favoritos da equipe
Pequeno
Arduino Datalogger - simple
(há 17 dias) jmascorella@loreto.nsw.ed... ♡6 ○0
multiplexing
(há 2 meses) simmen ♡111 ○10
Arduino RC Robot
(há 3 meses) Helektrika ♡151 ○11
Movimiento servo
(há 3 meses) #aprendeConDynamo ♡71 ○6

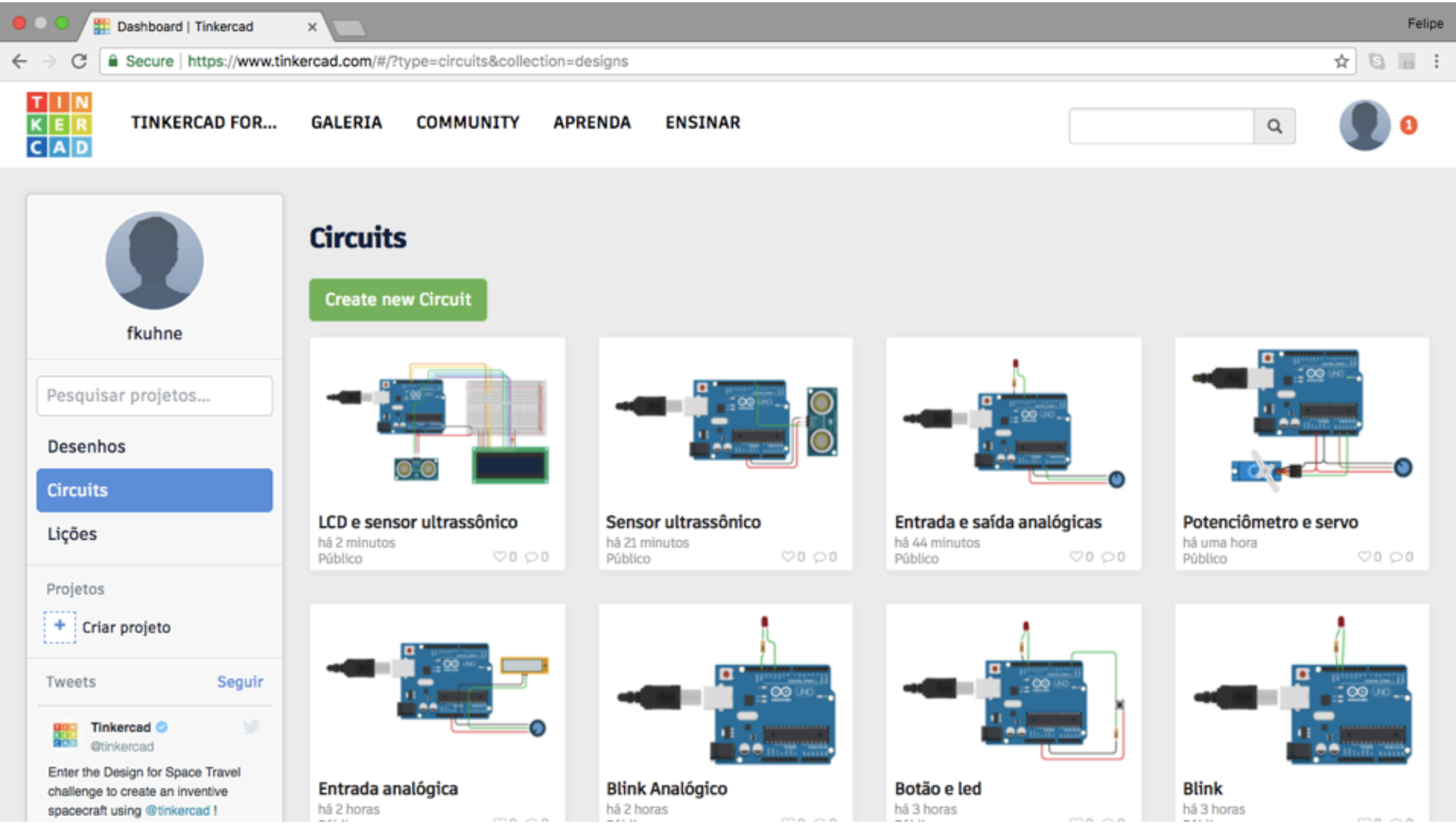
Nosso cenário

USB PC





```
/*
  Blink
  Turns on an LED on for one second, then off for one second, repeatedly.

  Most Arduinos have an on-board LED you can control. On the Uno and
  Leonardo, it is attached to digital pin 13. If you're unsure what
  pin the on-board LED is connected to on your Arduino model, check
  the documentation at http://www.arduino.cc

  This example code is in the public domain.

  modified 8 May 2014
  by Scott Fitzgerald
 */


// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin 13 as an output.
  pinMode(13, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(13, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);              // wait for a second
  digitalWrite(13, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);              // wait for a second
}
```



Enter the Design for Space Travel challenge to create an inventive spacecraft using @tinkercad !

Circuits

LCD e sensor ultrassônico
há 2 minutos
Público ♡0 ◯0

Sensor ultrassônico
há 21 minutos
Público ♡0 ◯0

Entrada e saída analógicas
há 44 minutos
Público ♡0 ◯0

Potenciômetro e servo
há uma hora
Público ♡0 ◯0

Entrada analógica
há 2 horas

Blink Analógico
há 2 horas

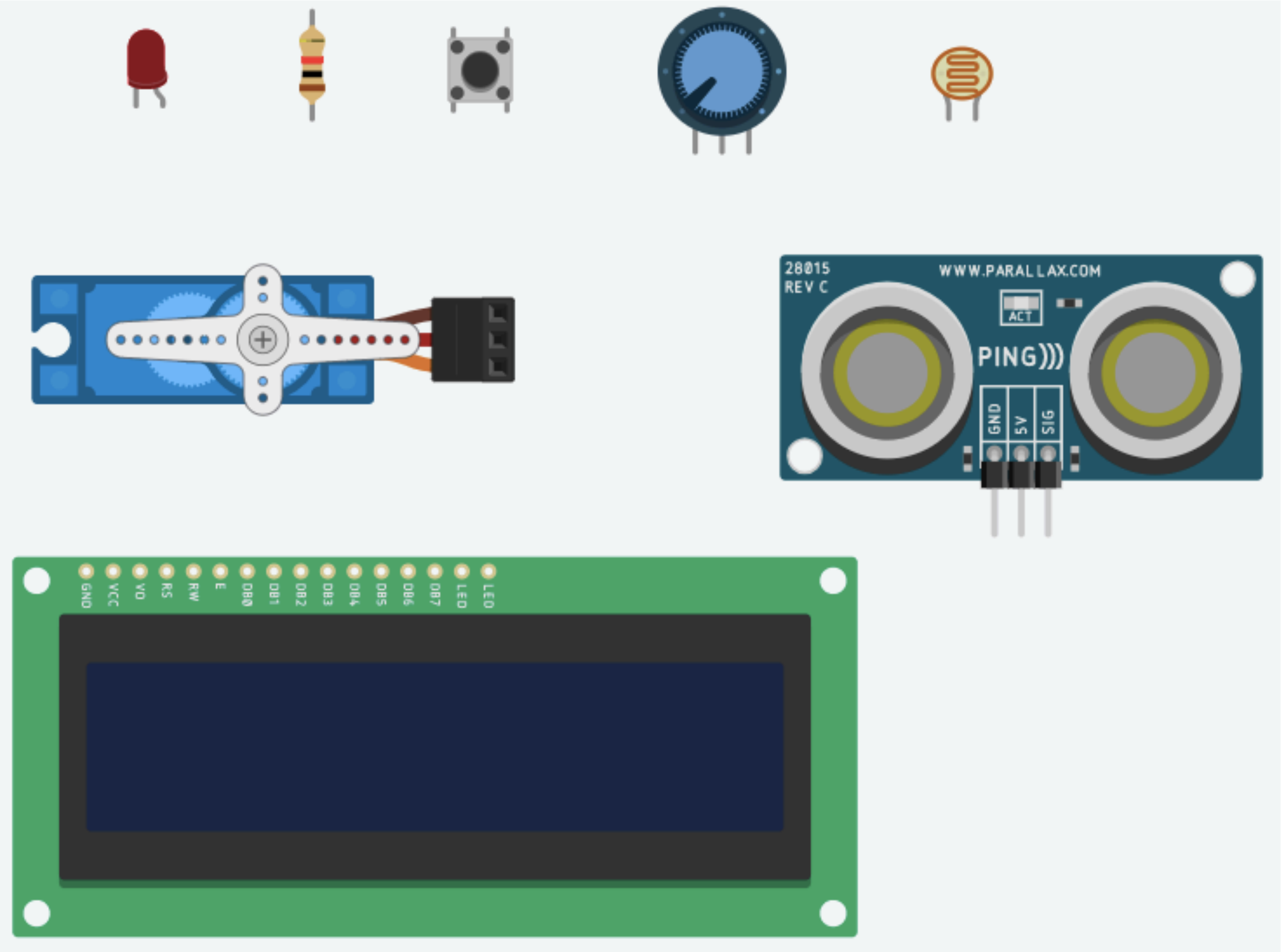
Botão e led
há 3 horas

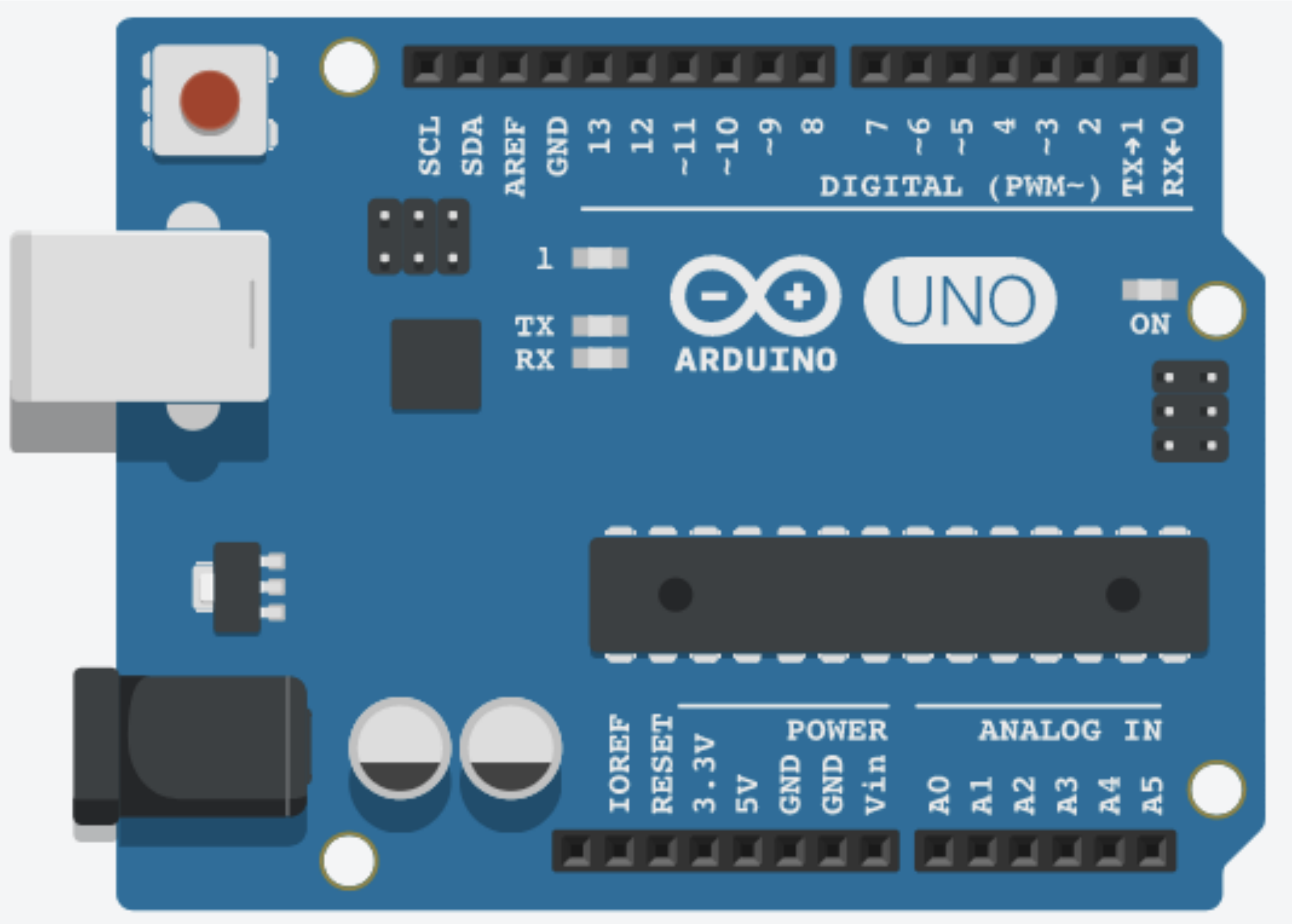
Blink
há 3 horas

Funções básicas

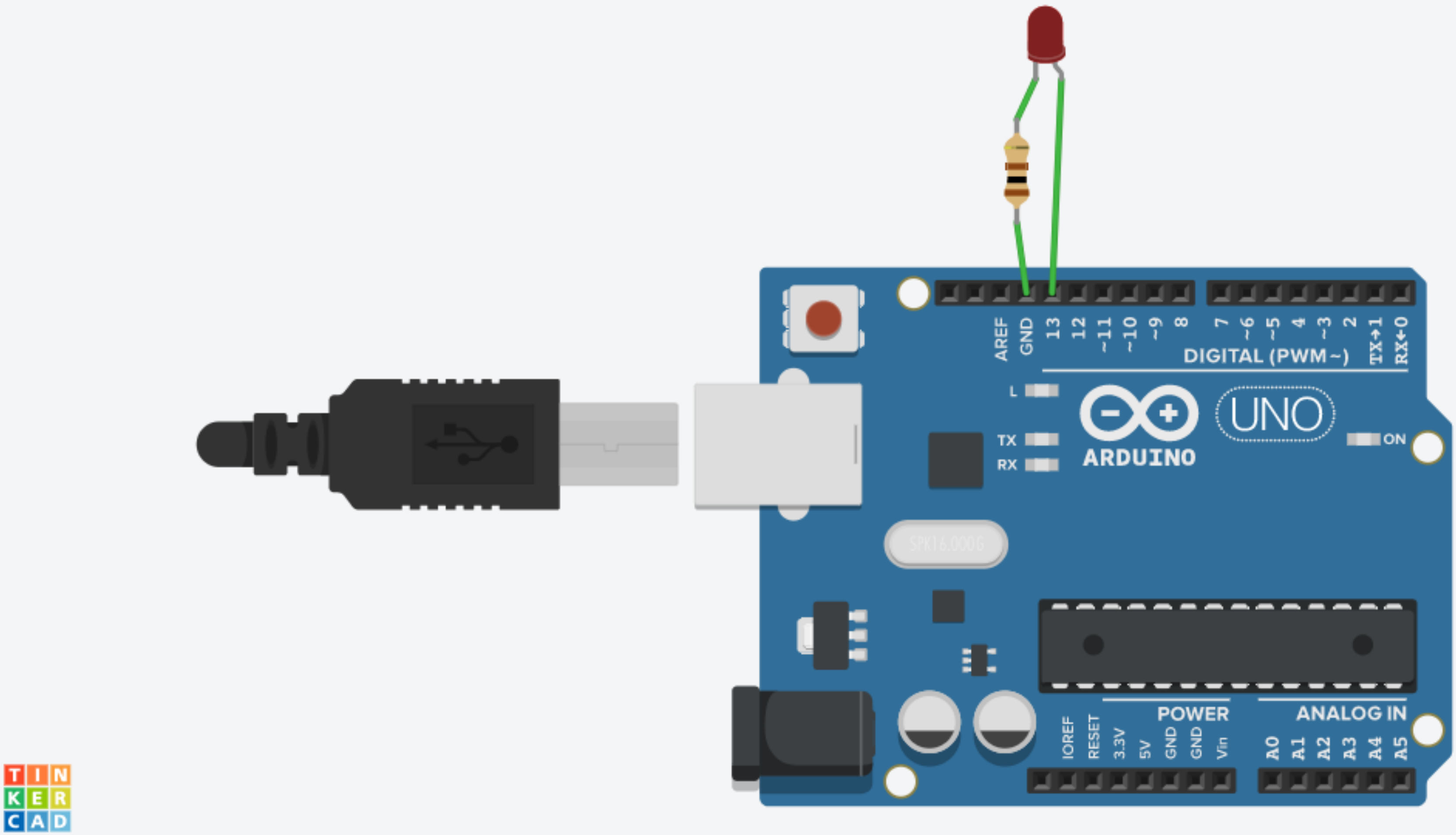
- pinMode()
- digitalWrite()
- digitalRead()
- delay()
- analogWrite()
- analogRead()
- Classe Serial
- Outras libs...

Componentes básicos

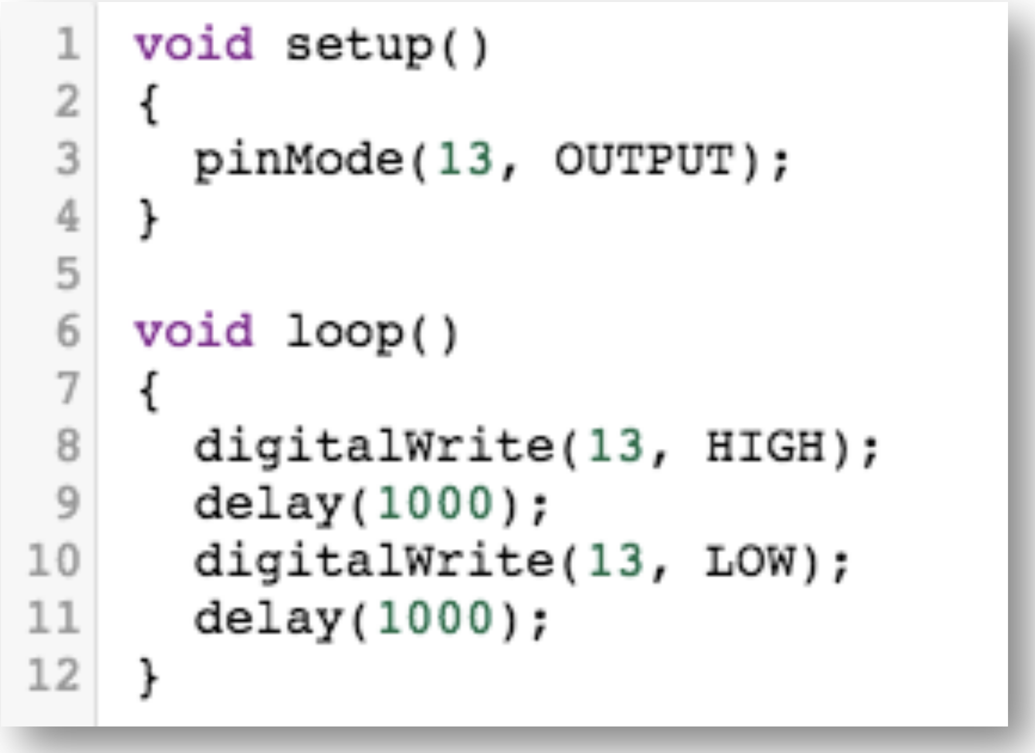



SCL
SDA
AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX→1
RX←0
DIGITAL (PWM~)
1
TX
RX
ARDUINO
UNO
ON
IOREF
RESET
3.3V
5V
GND
GND
Vin
POWER
ANALOG IN
A0
A1
A2
A3
A4
A5

Blink

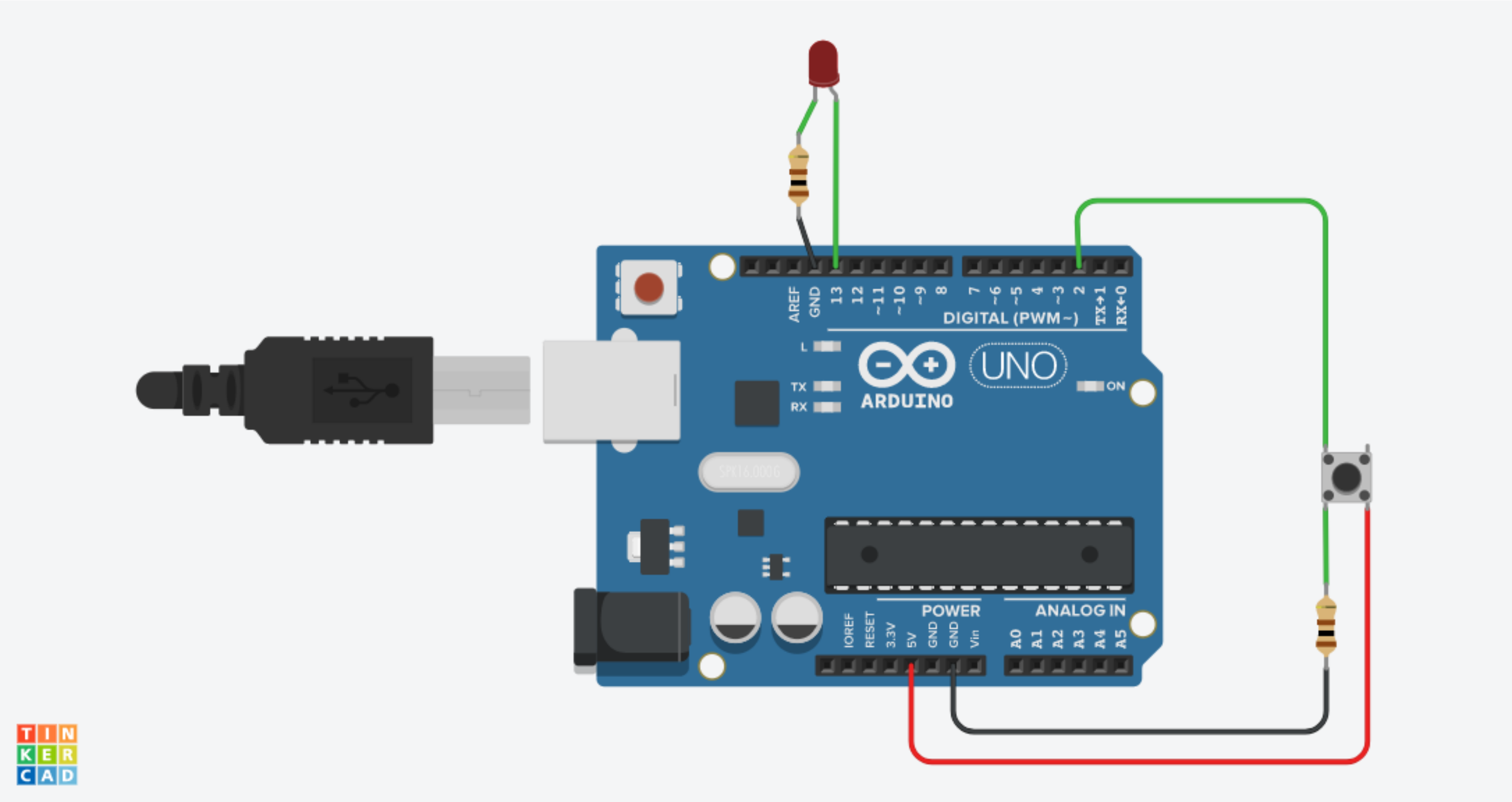

AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX→1
RX←0
DIGITAL (PWM~)
L
TX
RX
ARDUINO
UNO
ON
POWER
ANALOG IN
IOREF
RESET
3.3V
5V
GND
GND
Vin
A0
A1
A2
A3
A4
A5
TIN
KER
CAD

Blink

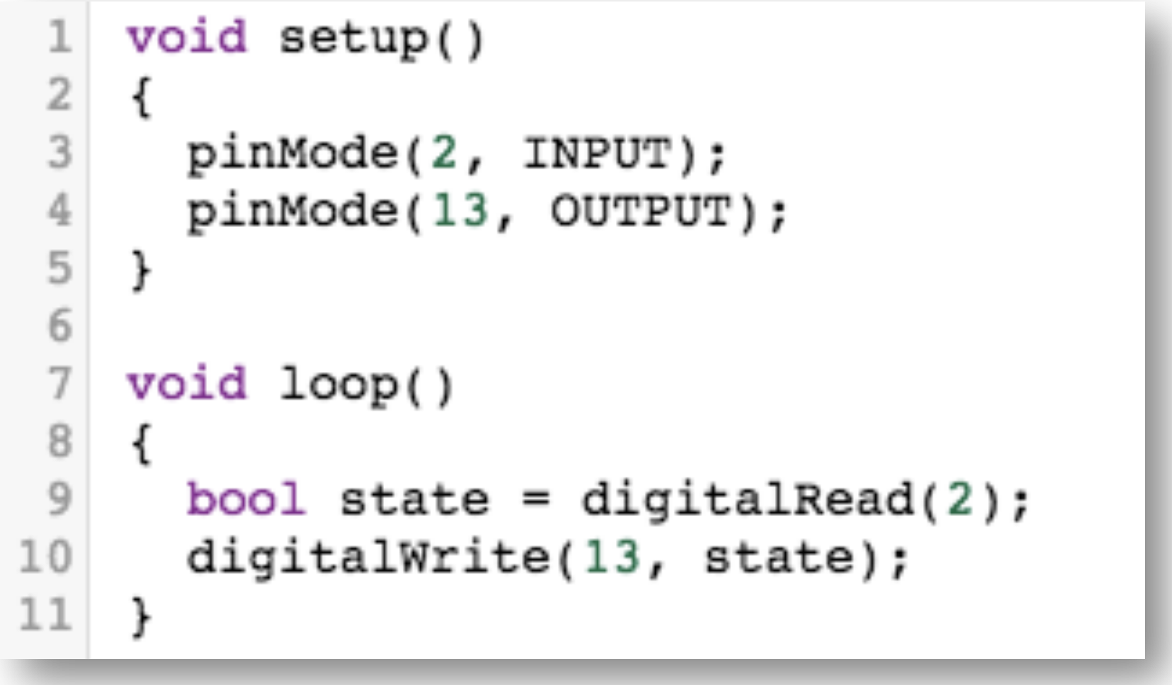


```
void setup()
{
  pinMode(13, OUTPUT);
}

void loop()
{
  digitalWrite(13, HIGH);
  delay(1000);
  digitalWrite(13, LOW);
  delay(1000);
}
```

Botão e LED

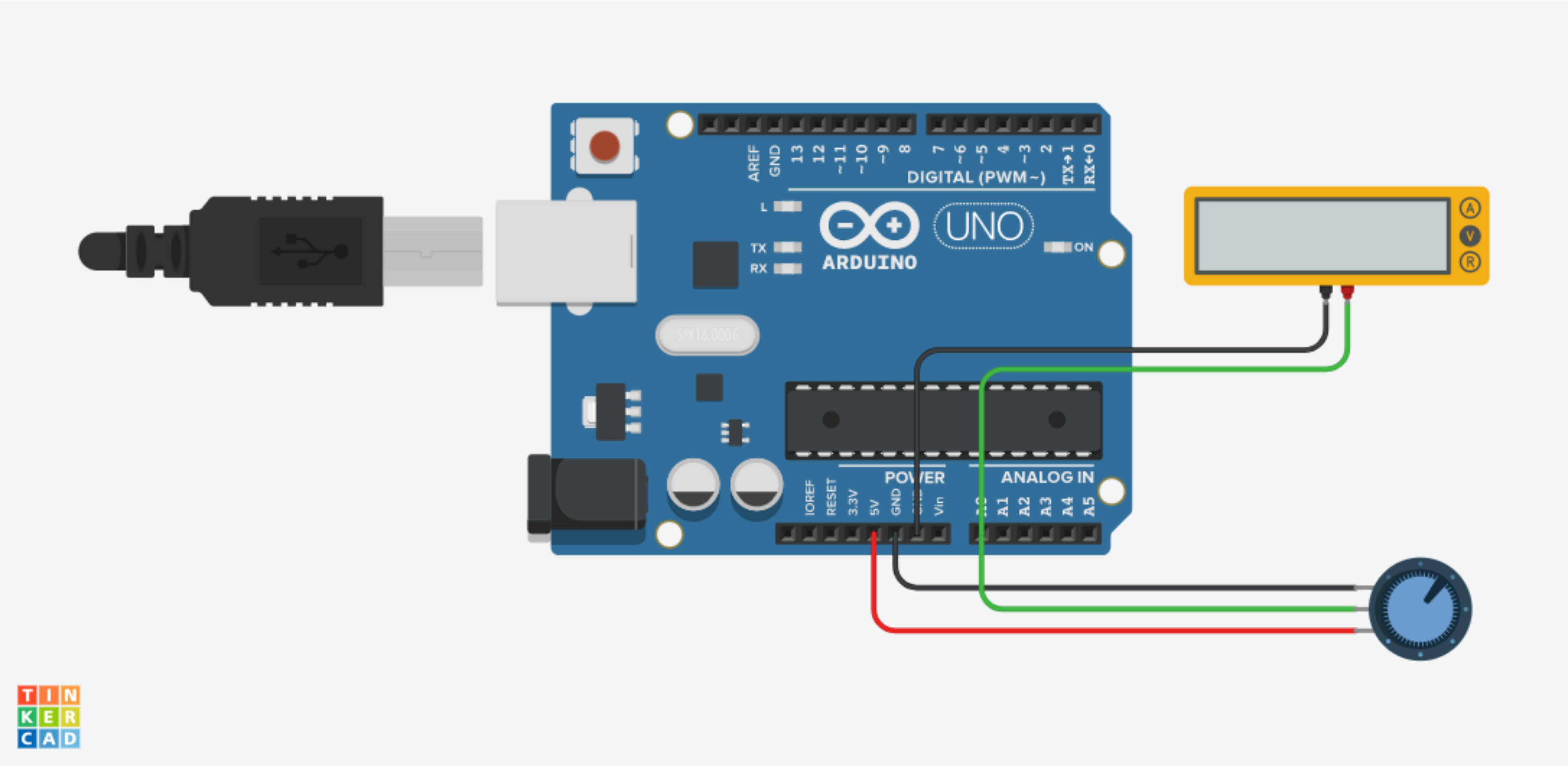

AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX→1
RX←0
DIGITAL (PWM~)
L
TX
RX
ARDUINO
UNO
ON
IOREF
RESET
3.3V
5V
GND
GND
Vin
POWER
ANALOG IN
A0
A1
A2
A3
A4
A5
TINKERCAD

Botão e LED

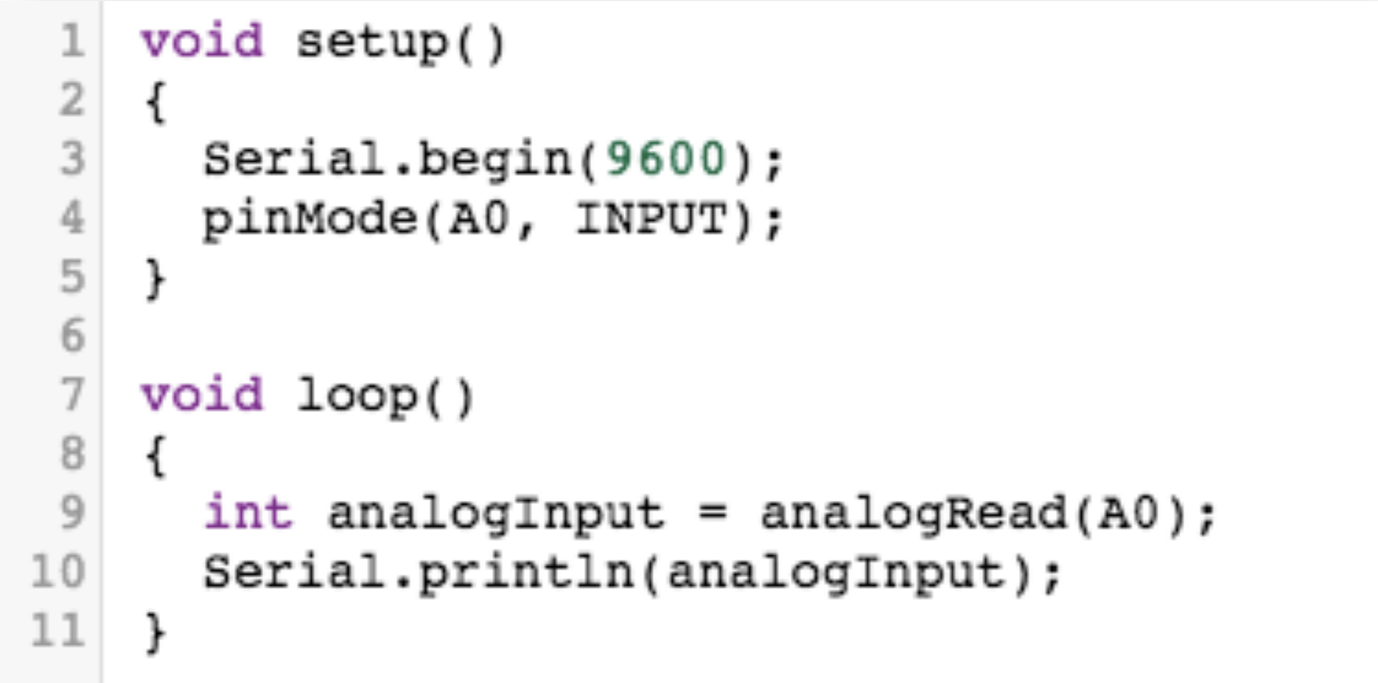


```
void setup()
{
  pinMode(2, INPUT);
  pinMode(13, OUTPUT);
}

void loop()
{
  bool state = digitalRead(2);
  digitalWrite(13, state);
}
```

Entrada analógica

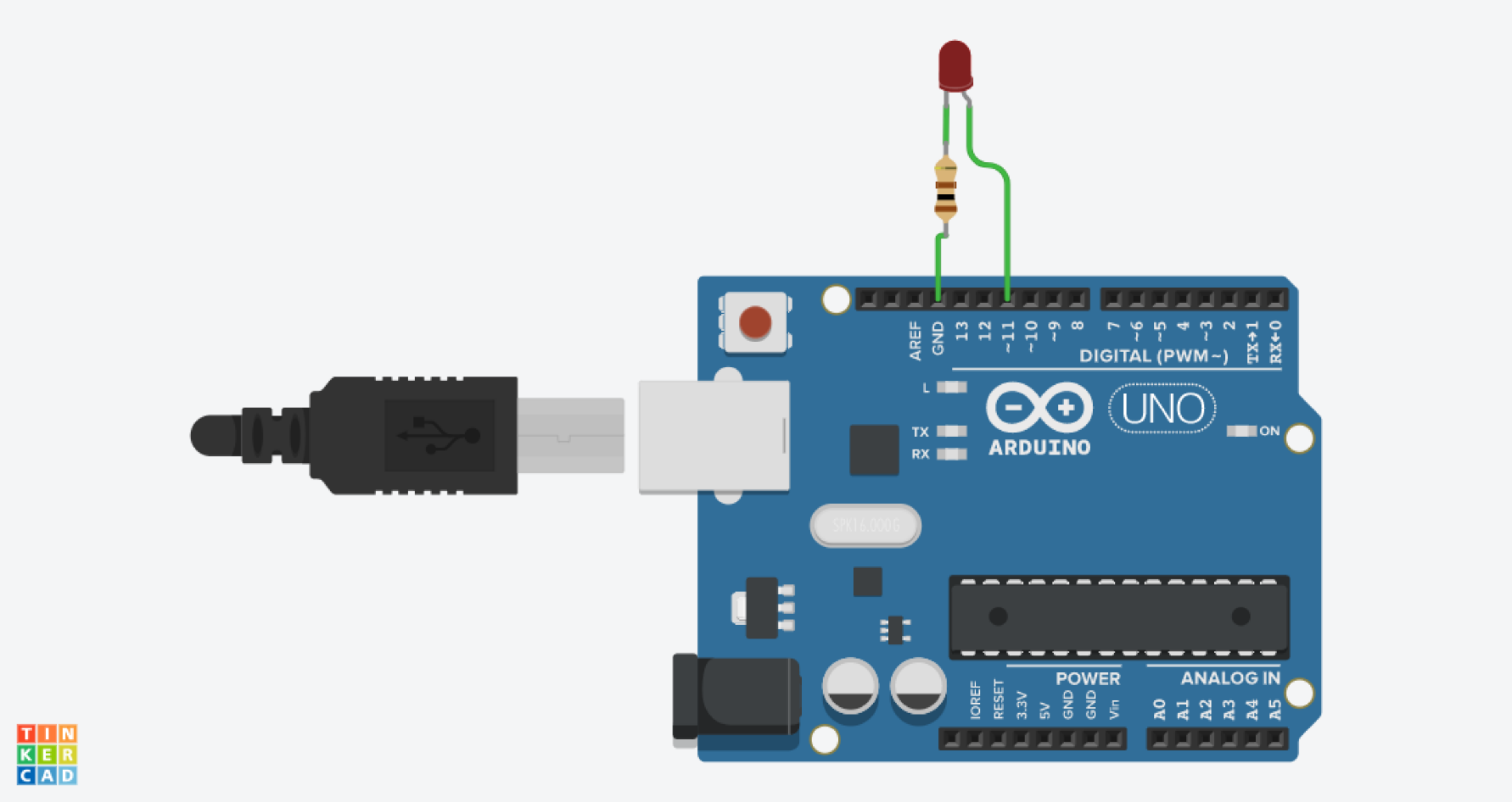


Entrada analógica

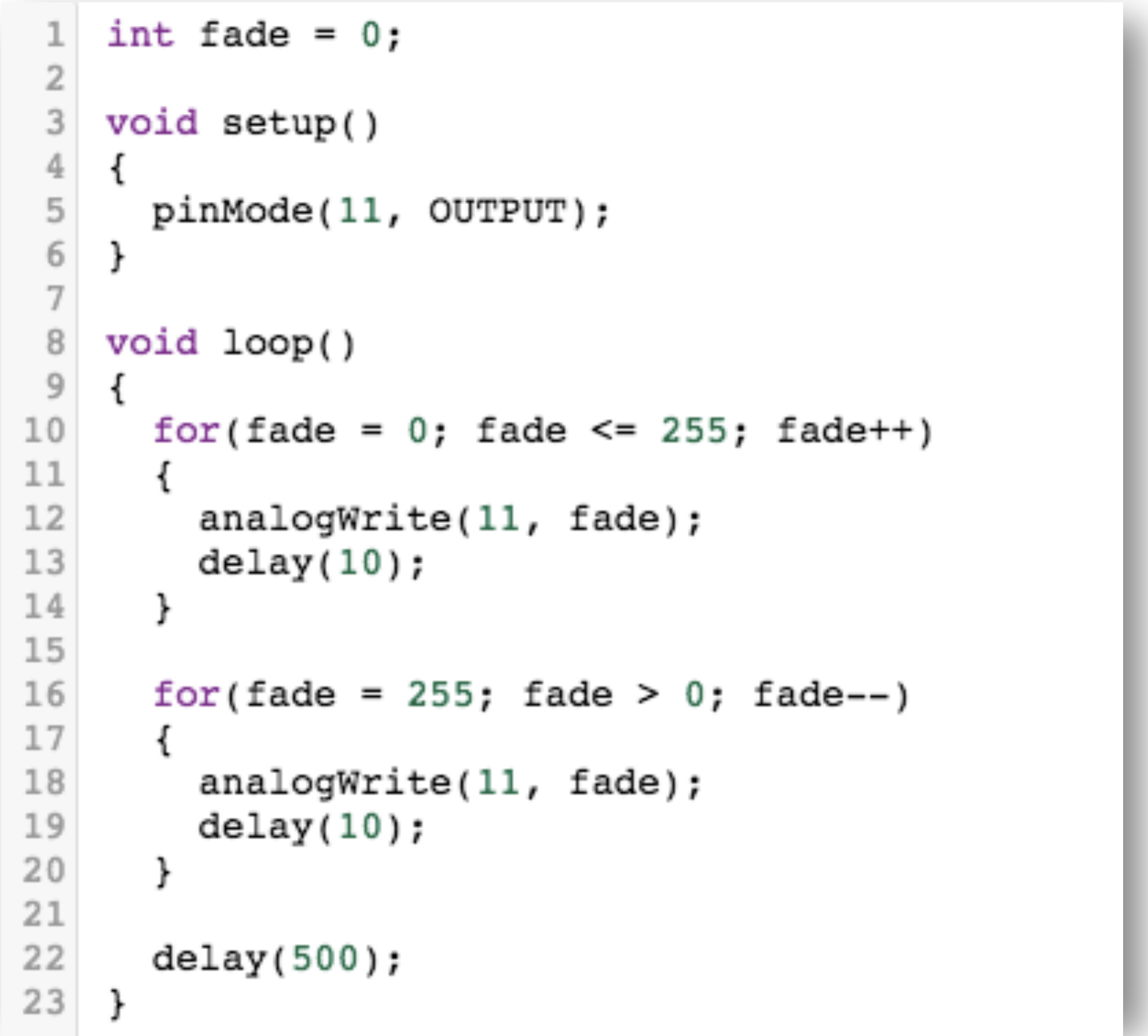


```
void setup()
{
  Serial.begin(9600);
  pinMode(A0, INPUT);
}

void loop()
{
  int analogInput = analogRead(A0);
  Serial.println(analogInput);
}
```

Saída analógica

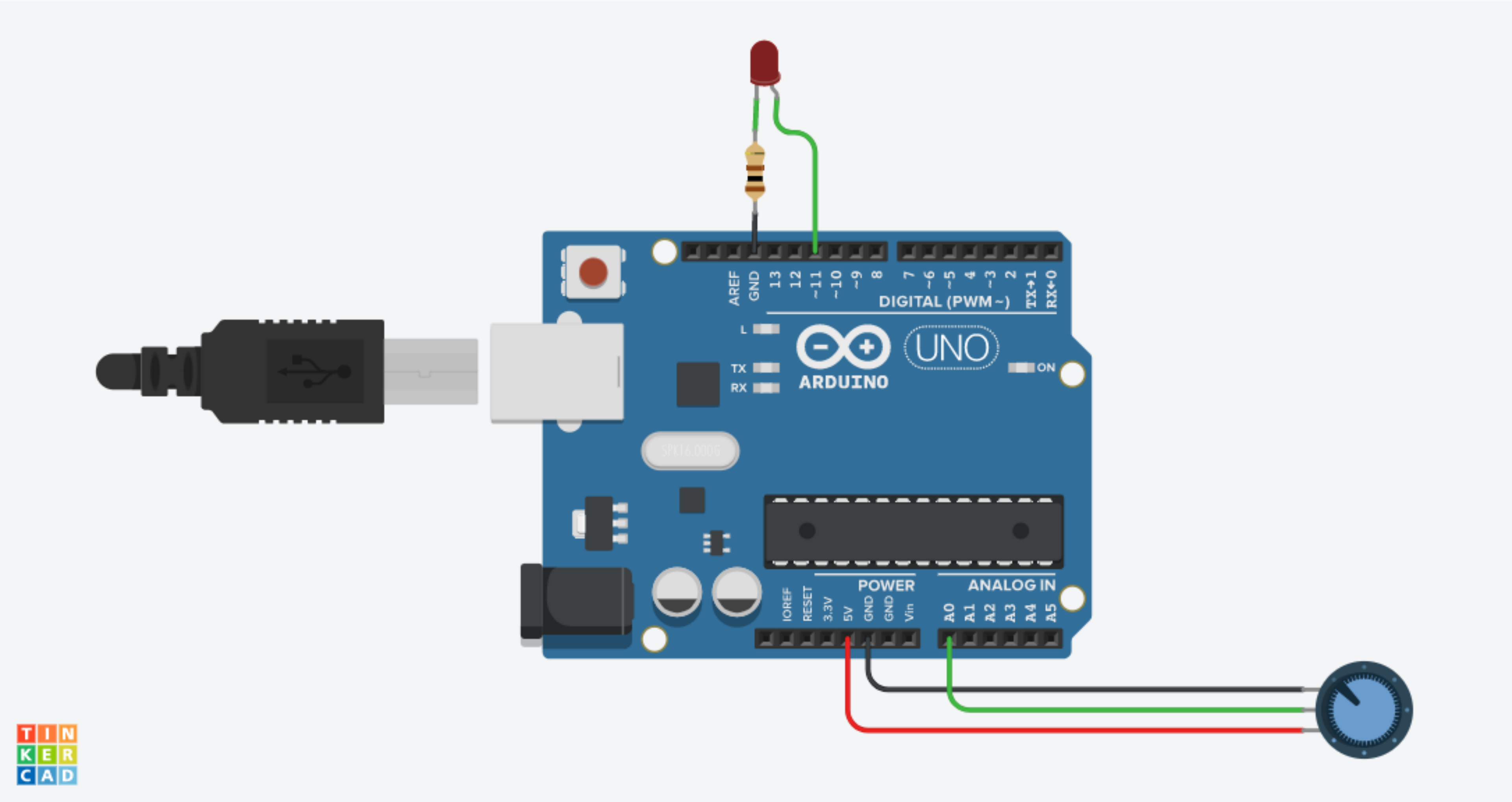


Saída analógica

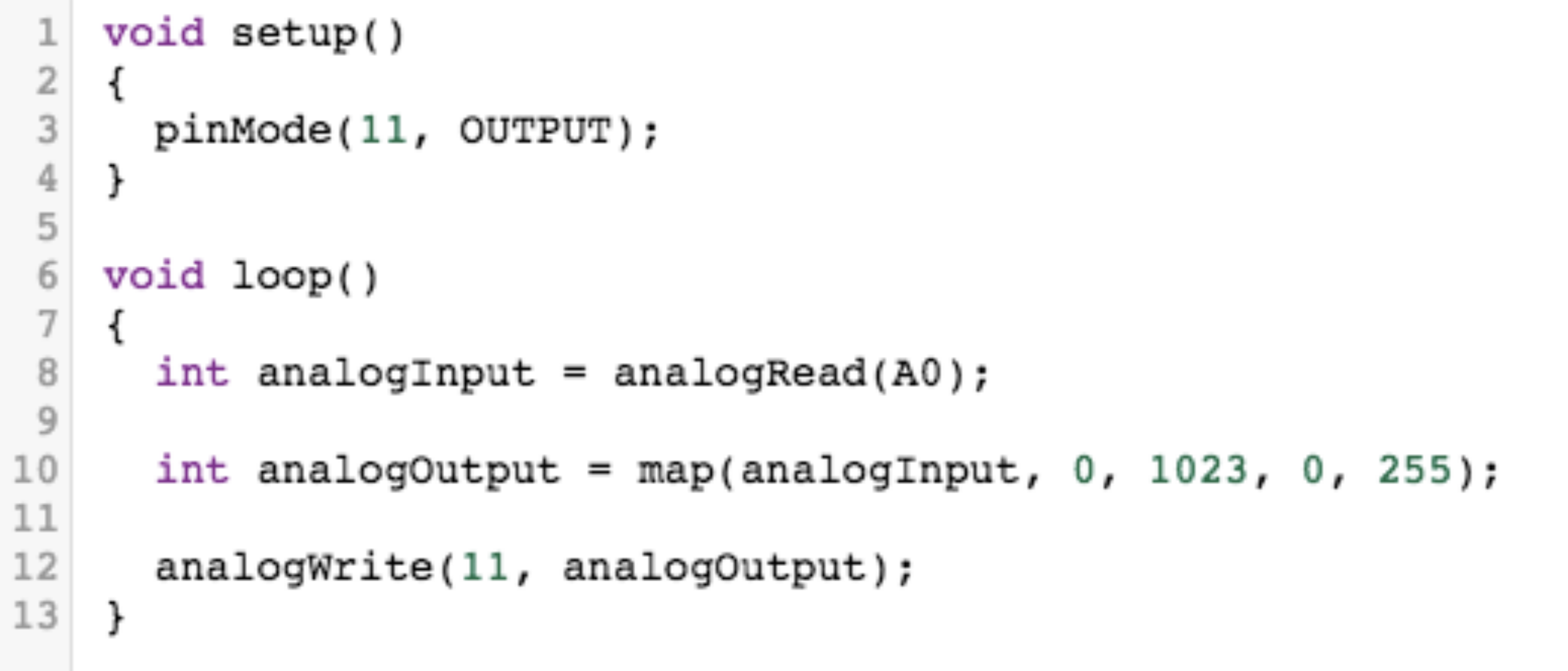


```
int fade = 0;

void setup()
{
  pinMode(11, OUTPUT);
}

void loop()
{
  for(fade = 0; fade <= 255; fade++)
  {
    analogWrite(11, fade);
    delay(10);
  }

  for(fade = 255; fade > 0; fade--)
  {
    analogWrite(11, fade);
    delay(10);
  }

  delay(500);
}
```

Entrada e saída analógicas

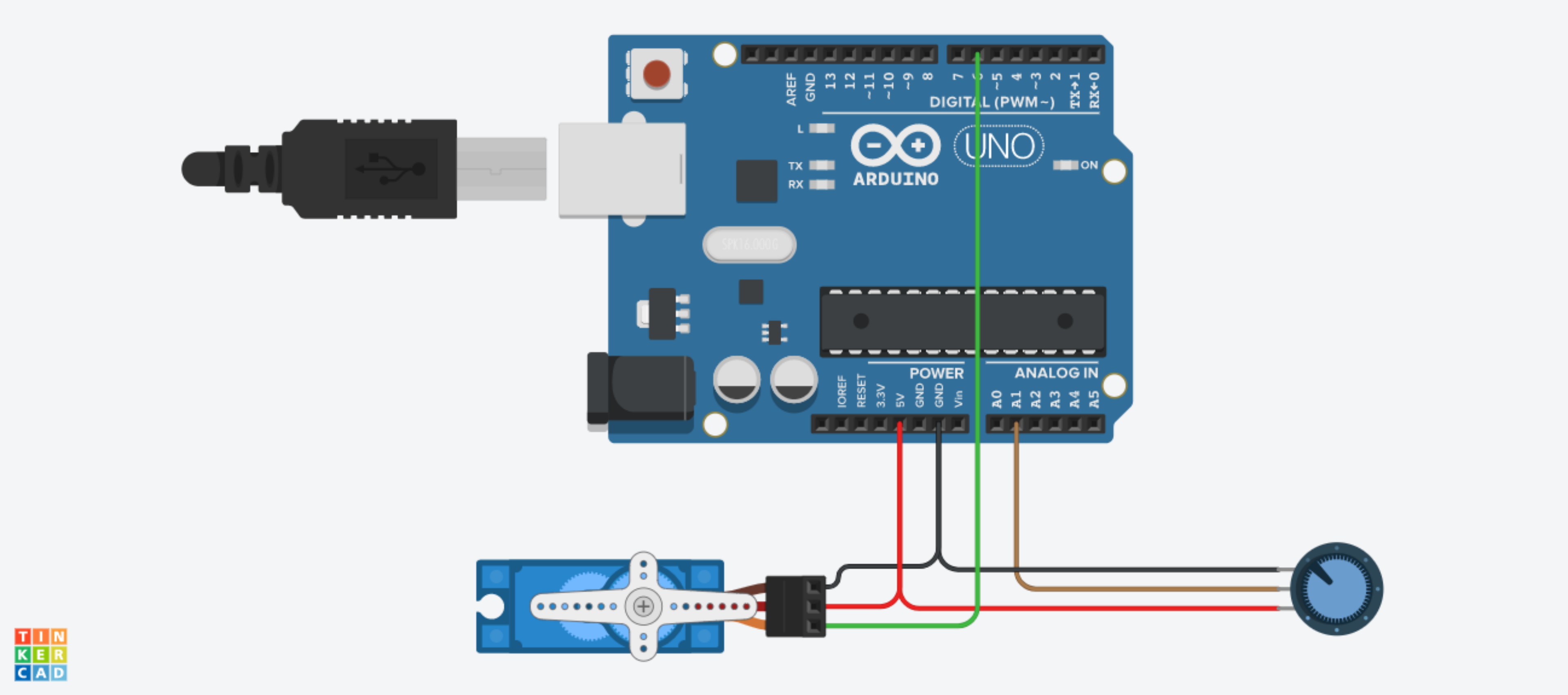


Entrada e saída analógicas

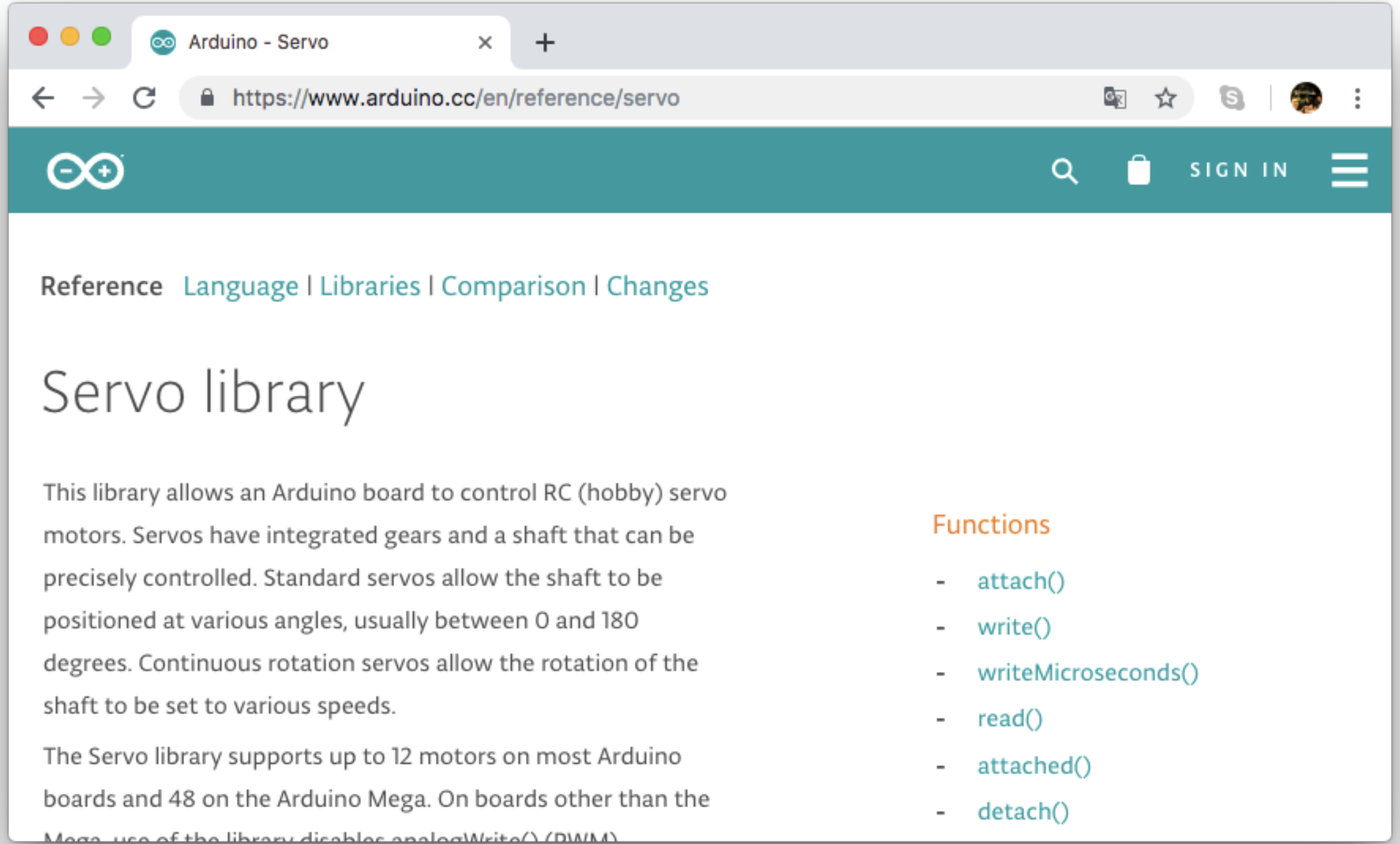


```
void setup()
{
  pinMode(11, OUTPUT);
}

void loop()
{
  int analogInput = analogRead(A0);

  int analogOutput = map(analogInput, 0, 1023, 0, 255);

  analogWrite(11, analogOutput);
}
```

Potenciômetro e Servo



https://www.arduino.cc/en/reference/servo


Arduino - Servo
https://www.arduino.cc/en/reference/servo
SIGN IN
Reference Language | Libraries | Comparison | Changes

Servo library

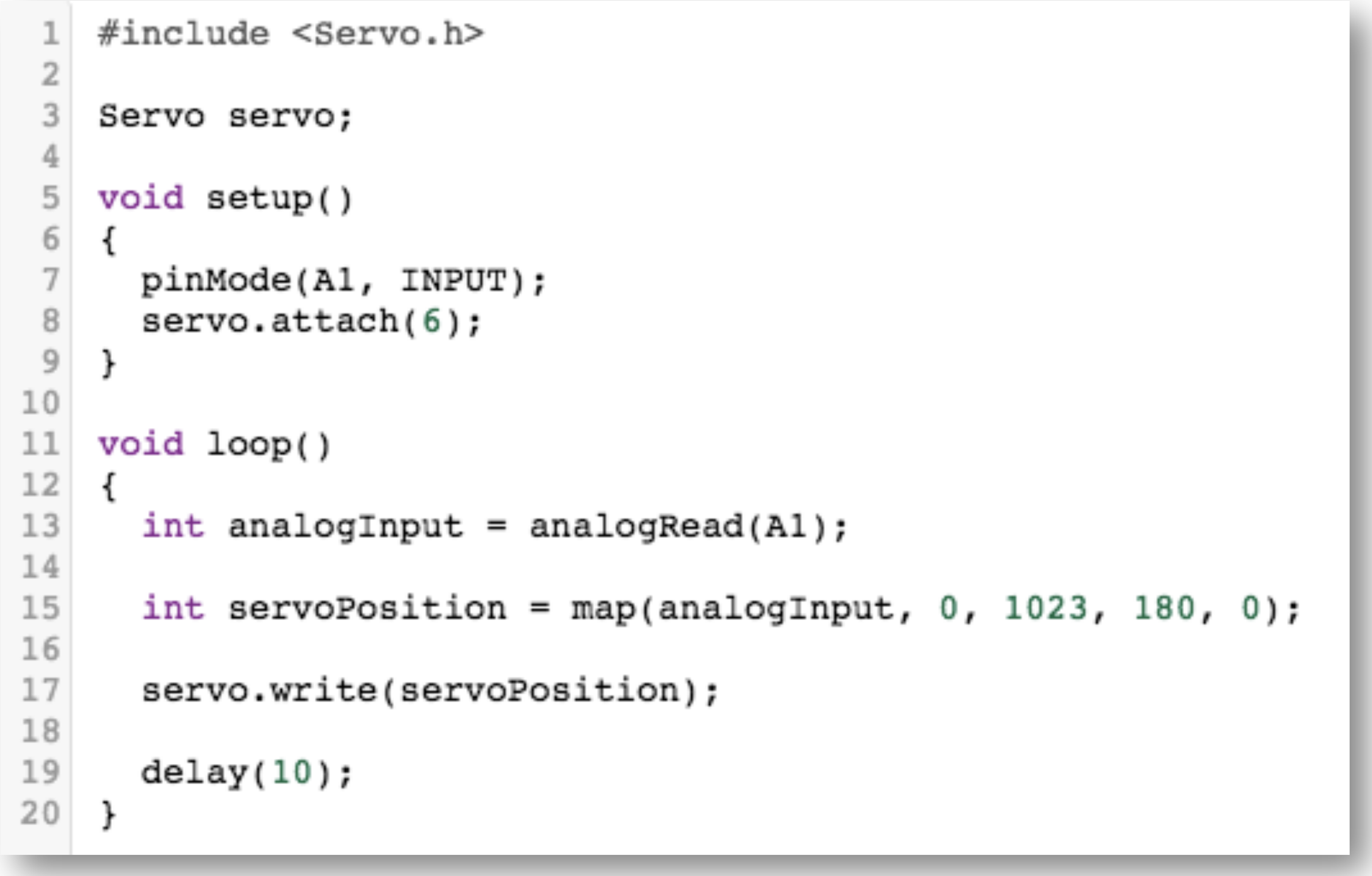
This library allows an Arduino board to control RC (hobby) servo motors. Servos have integrated gears and a shaft that can be precisely controlled. Standard servos allow the shaft to be positioned at various angles, usually between 0 and 180 degrees. Continuous rotation servos allow the rotation of the shaft to be set to various speeds.

The Servo library supports up to 12 motors on most Arduino boards and 48 on the Arduino Mega. On boards other than the Mega, use of the library disables analogWrite() (PWM)

Functions

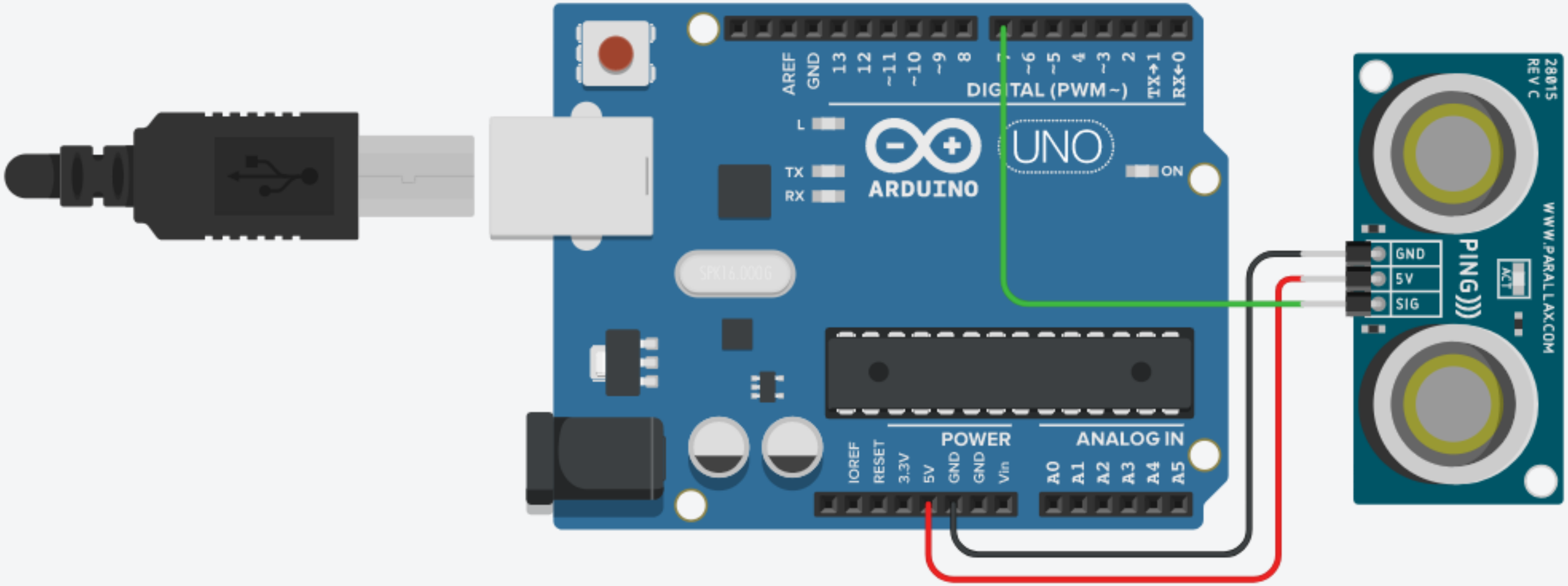
- attach()
- write()
- writeMicroseconds()
- read()
- attached()
- detach()

Potenciômetro e Servo

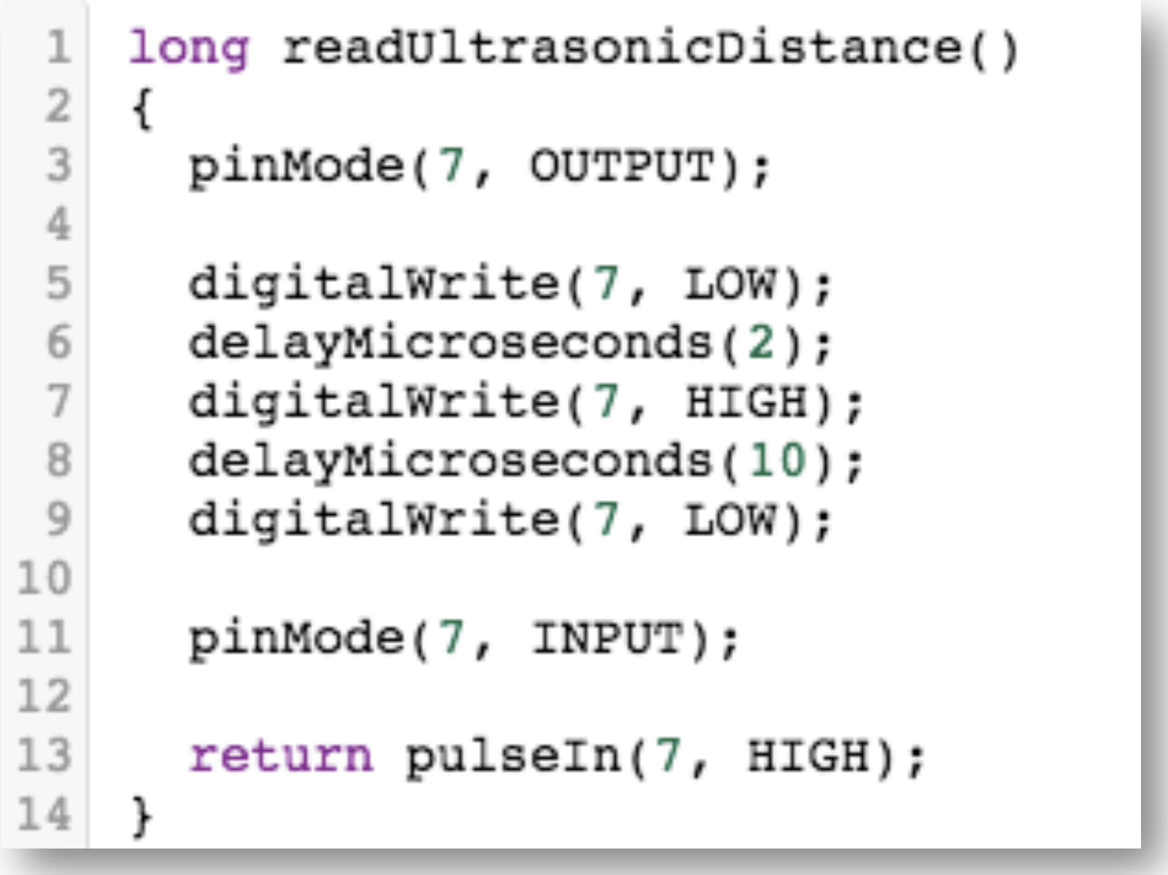


```
#include <Servo.h>

Servo servo;

void setup()
{
  pinMode(A1, INPUT);
  servo.attach(6);
}

void loop()
{
  int analogInput = analogRead(A1);

  int servoPosition = map(analogInput, 0, 1023, 180, 0);

  servo.write(servoPosition);

  delay(10);
}
```

Sensor ultrassônico

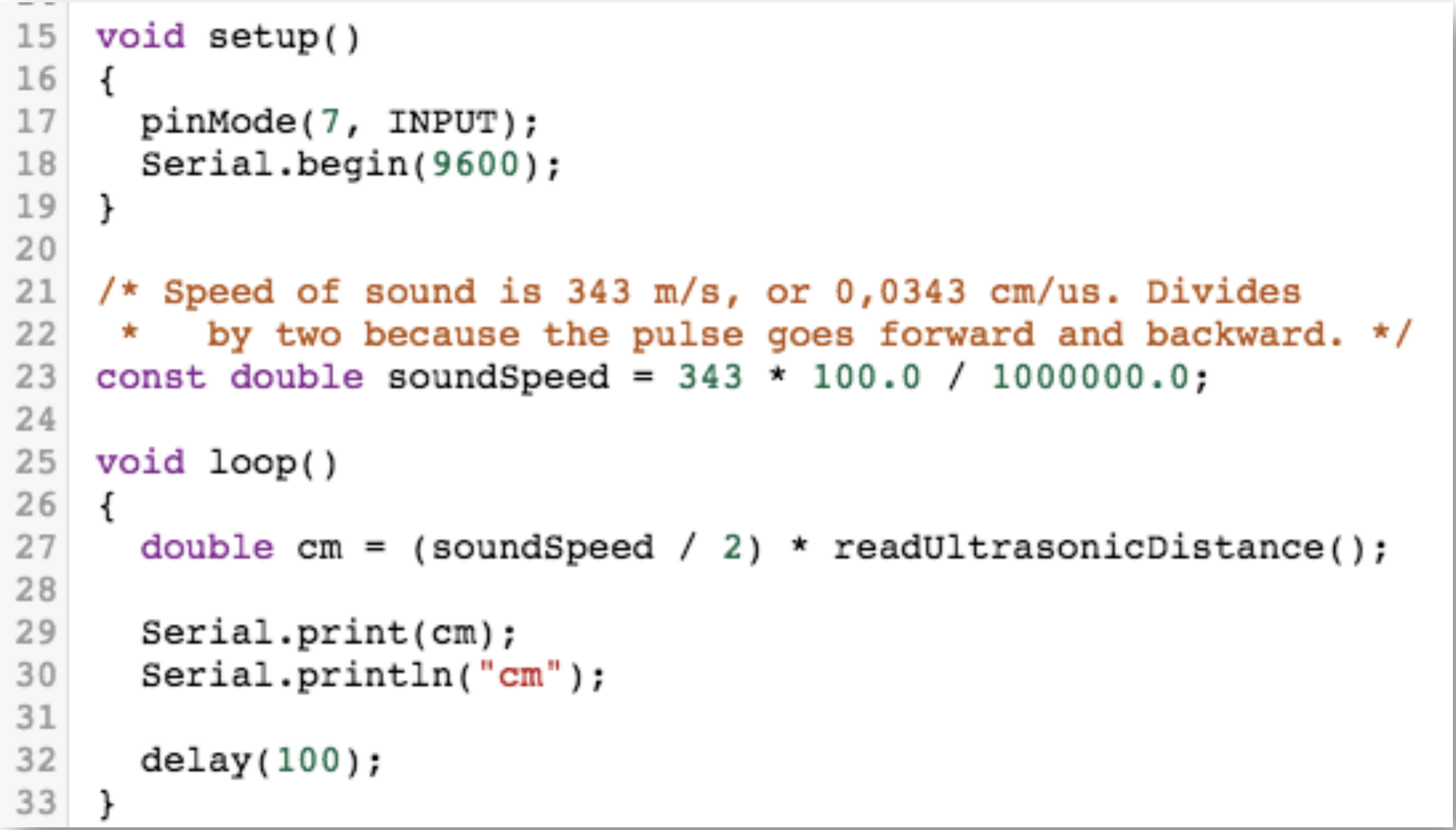


Sensor ultrassônico



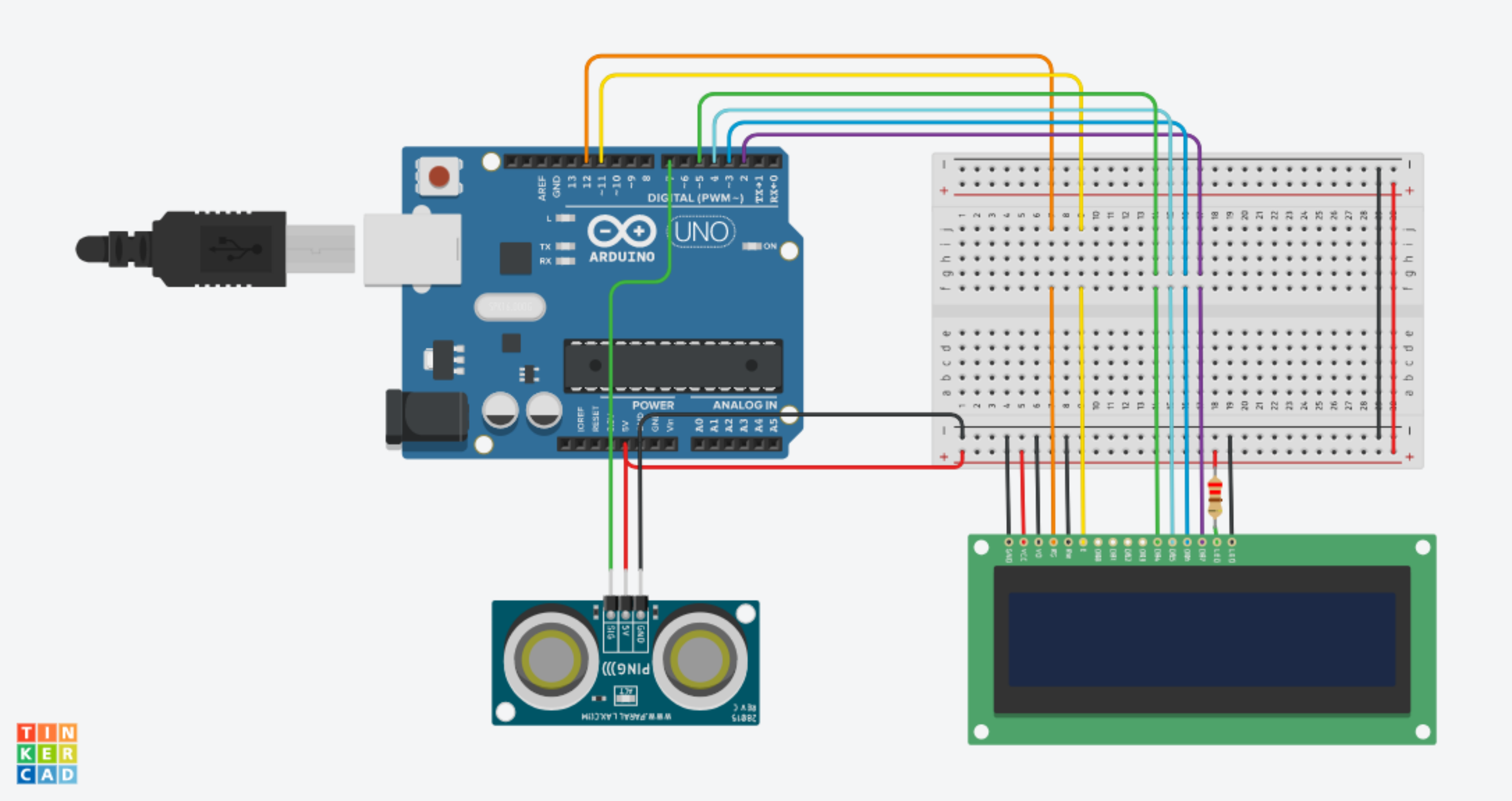
```
long readUltrasonicDistance()
{
  pinMode(7, OUTPUT);

  digitalWrite(7, LOW);
  delayMicroseconds(2);
  digitalWrite(7, HIGH);
  delayMicroseconds(10);
  digitalWrite(7, LOW);

  pinMode(7, INPUT);

  return pulseIn(7, HIGH);
}
```

Sensor ultrassônico

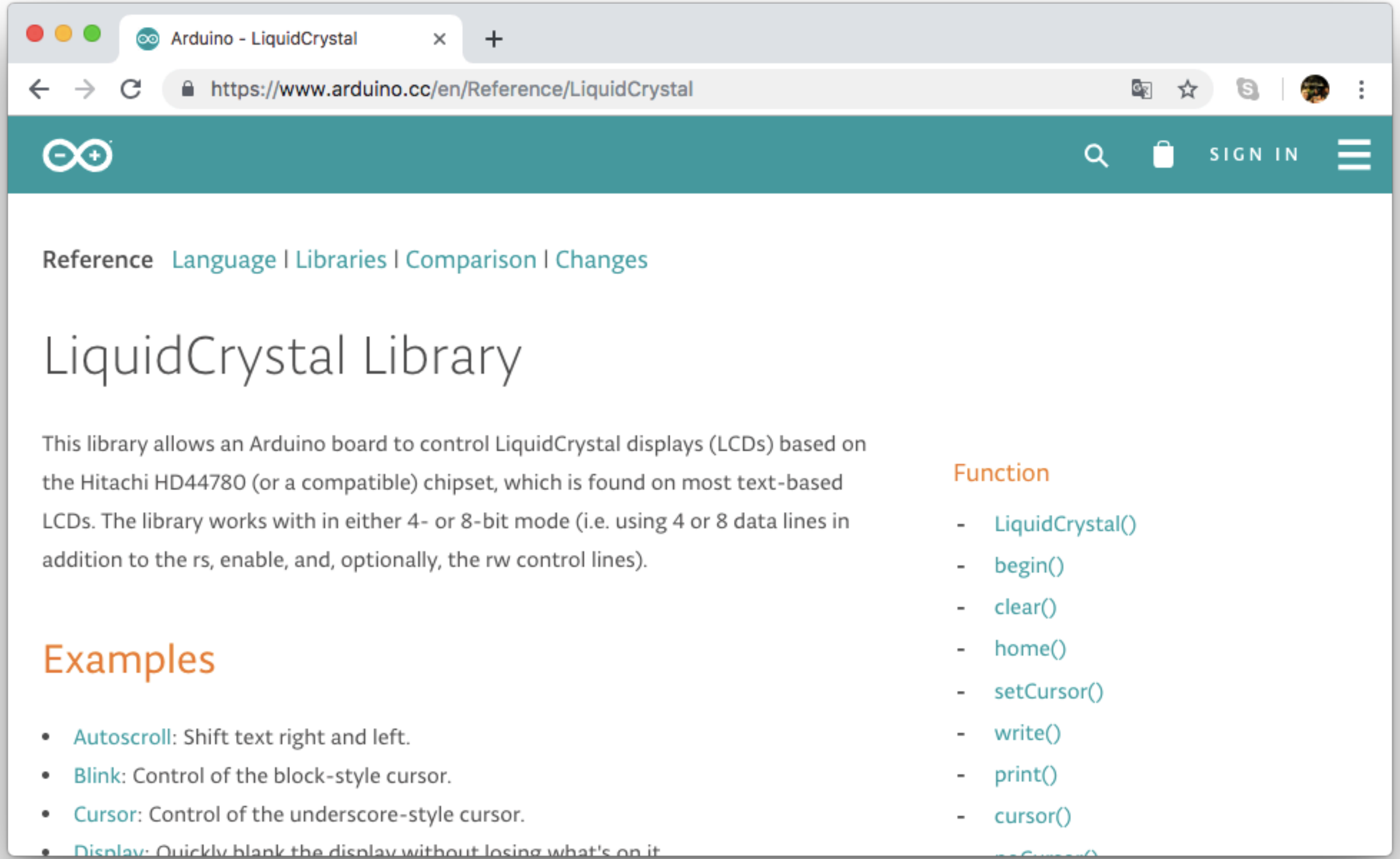


```
void setup()
{
  pinMode(7, INPUT);
  Serial.begin(9600);
}

/* Speed of sound is 343 m/s, or 0,0343 cm/us. Divides
 *   by two because the pulse goes forward and backward. */
const double soundSpeed = 343 * 100.0 / 1000000.0;

void loop()
{
  double cm = (soundSpeed / 2) * readUltrasonicDistance();

  Serial.print(cm);
  Serial.println("cm");

  delay(100);
}
```

Sensor ultrassônico e LCD



https://www.arduino.cc/en/Reference/LiquidCrystal



Reference Language | Libraries | Comparison | Changes

LiquidCrystal Library

This library allows an Arduino board to control LiquidCrystal displays (LCDs) based on the Hitachi HD44780 (or a compatible) chipset, which is found on most text-based LCDs. The library works with in either 4- or 8-bit mode (i.e. using 4 or 8 data lines in addition to the rs, enable, and, optionally, the rw control lines).

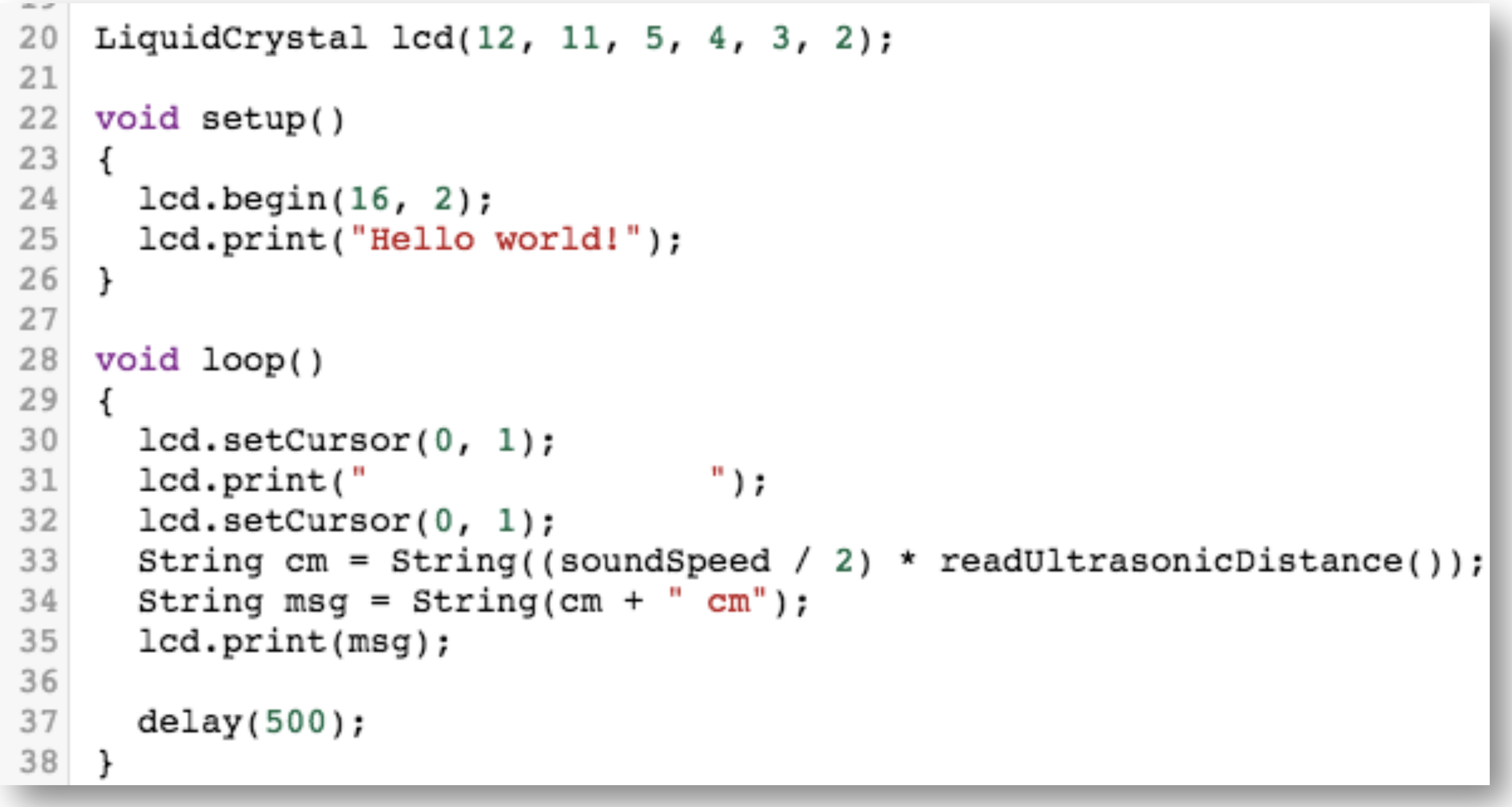
Examples

- Autoscroll: Shift text right and left.
- Blink: Control of the block-style cursor.
- Cursor: Control of the underscore-style cursor.
- Display: Quickly blank the display without losing what's on it.

Function

- LiquidCrystal()
- begin()
- clear()
- home()
- setCursor()
- write()
- print()
- cursor()
- noCursor()

Sensor ultrassônico e LCD



```
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup()
{
  lcd.begin(16, 2);
  lcd.print("Hello world!");
}

void loop()
{
  lcd.setCursor(0, 1);
  lcd.print("                ");
  lcd.setCursor(0, 1);
  String cm = String((soundSpeed / 2) * readUltrasonicDistance());
  String msg = String(cm + " cm");
  lcd.print(msg);

  delay(500);
}
```

Referências

- https://www.arduino.cc
- https://www.arduino.cc/reference/en
- https://www.tinkercad.com
- https://youtu.be/WynYhZx_qds
- https://www.arduino.cc/en/Tutorial/Knob
- http://www.arduino.cc/en/Tutorial/Ping
- http://www.arduino.cc/en/Tutorial/LiquidCrystal

Obrigado!

Felipe Kühne

fkuhne@gmail.com

github.com/fkuhne/sae2019