



Programação básica com Arduino

Felipe Kühne
fkuhne@pucrs.br

Maio/2018

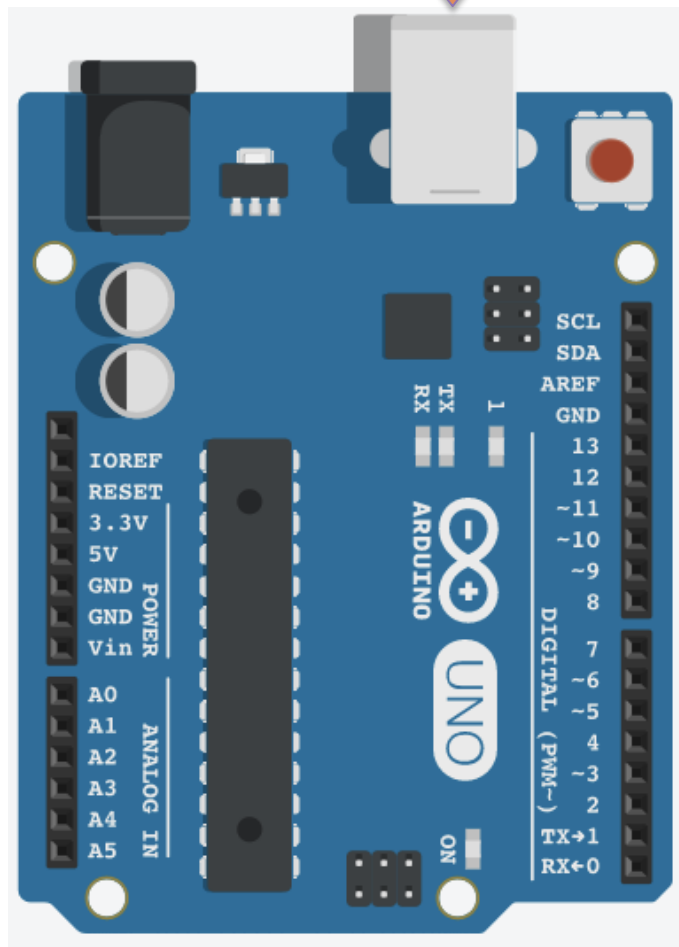


github.com/fkuhne/aday18



Nosso cenário

USB PC



```
Blink | Arduino 1.6.8

/*
 * 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
}
```

chipKIT DP32 on /dev/cu.usbmodem1411



TINKERCAD FOR...

GALERIA

COMMUNITY

APRENDA

ENSINAR



fkuhne

Pesquisar projetos...

Desenhos

Circuits

Lições

Projetos

+ Criar projeto

Tweets

Seguir

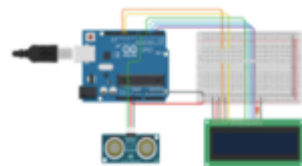


Tinkercad
@tinkercad

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

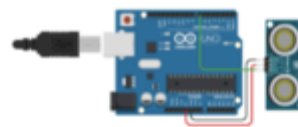
Circuits

Create new Circuit



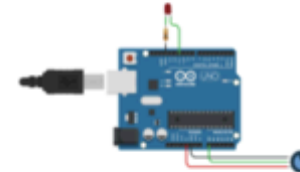
LCD e sensor ultrassônico

há 2 minutos
Público



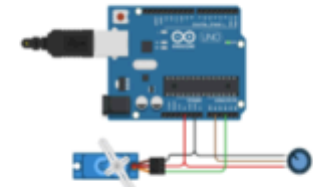
Sensor ultrassônico

há 21 minutos
Público



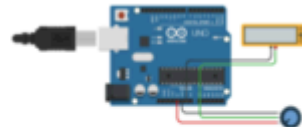
Entrada e saída analógicas

há 44 minutos
Público



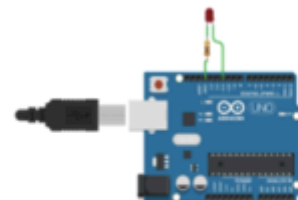
Potenciômetro e servo

há uma hora
Público



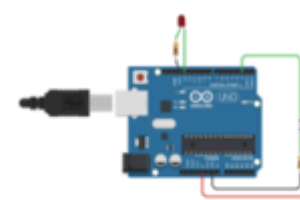
Entrada analógica

há 2 horas



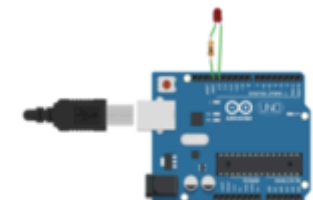
Blink Analógico

há 2 horas



Botão e led

há 3 horas



Blink

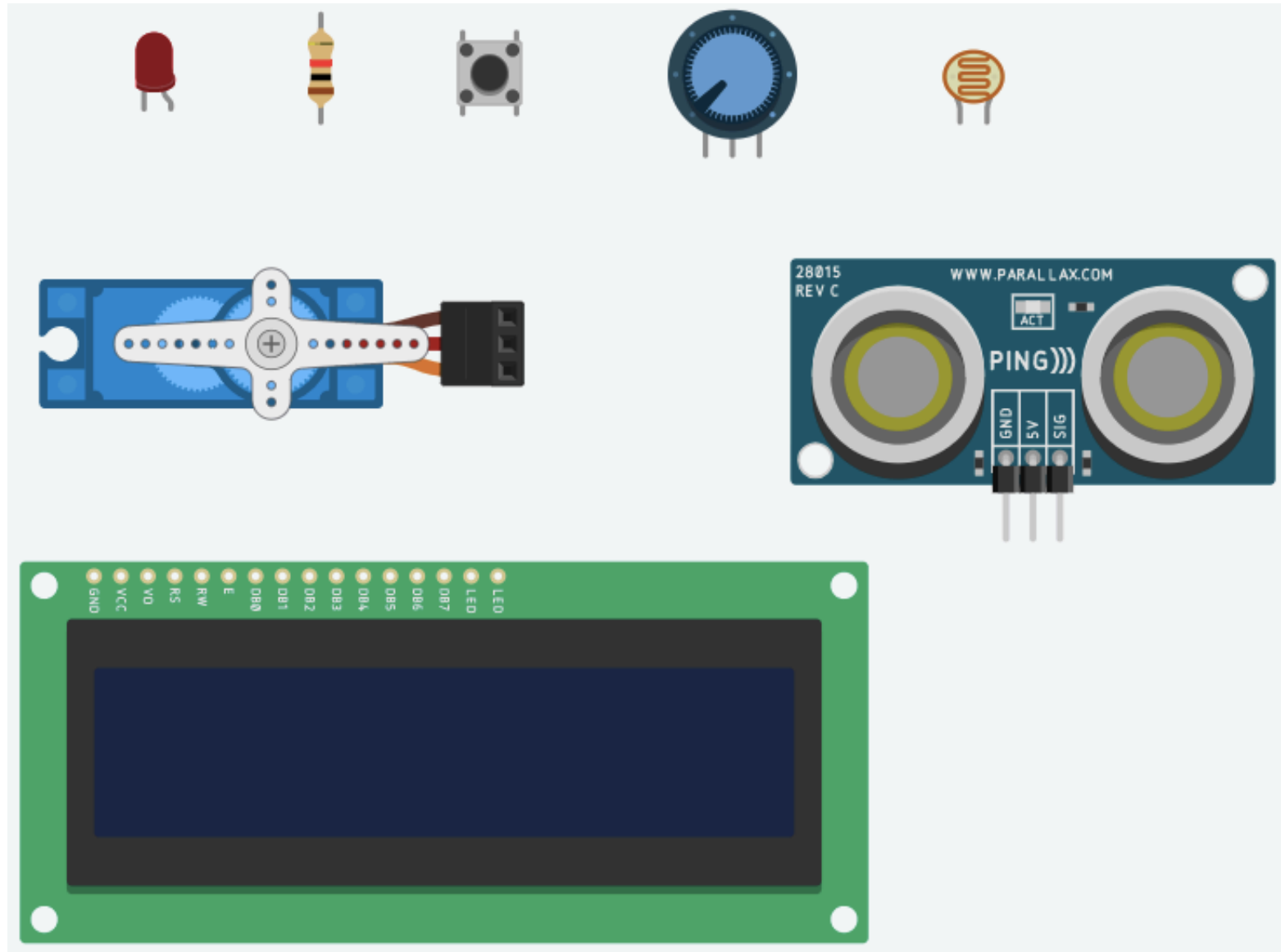
há 3 horas

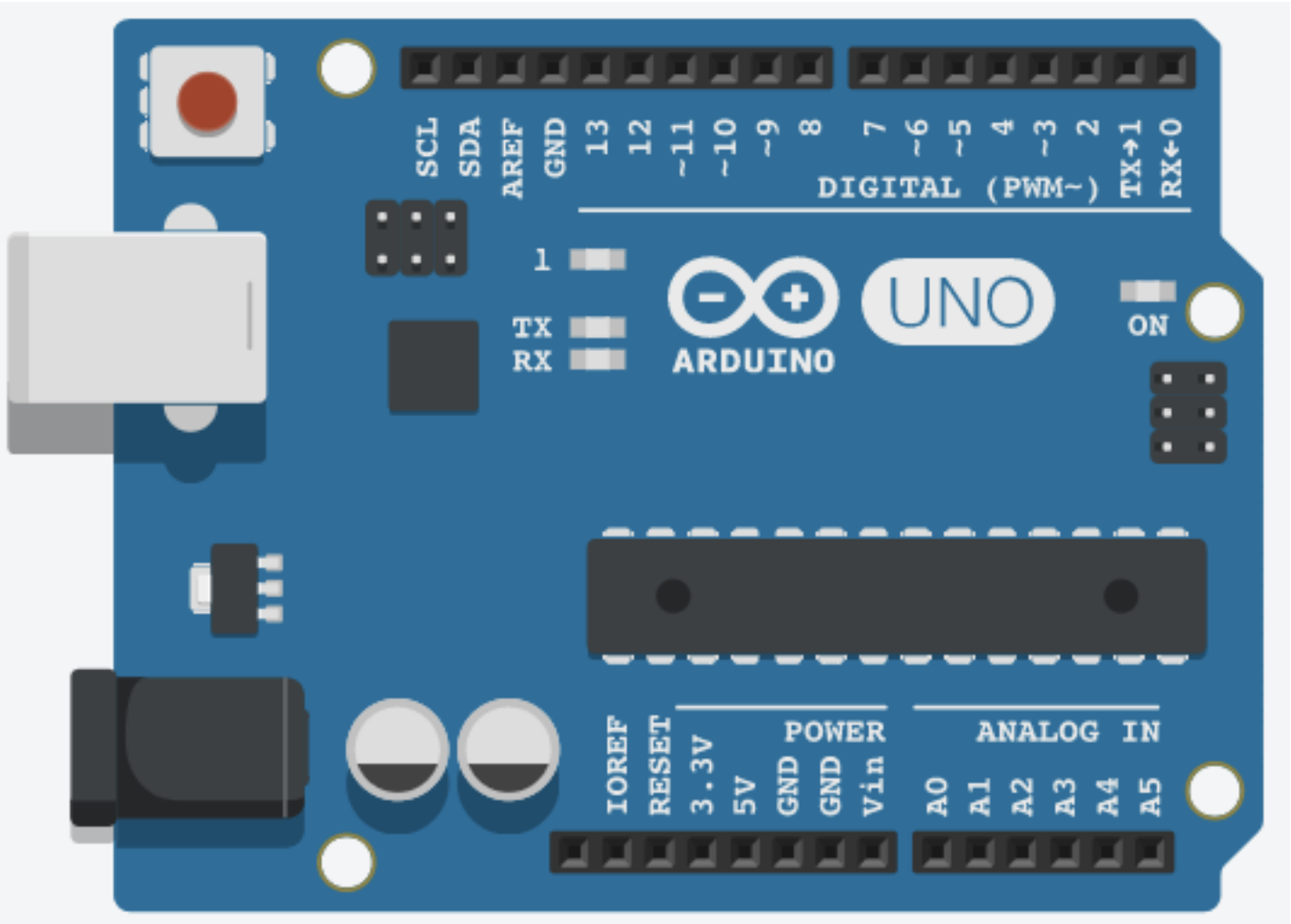


Funções básicas

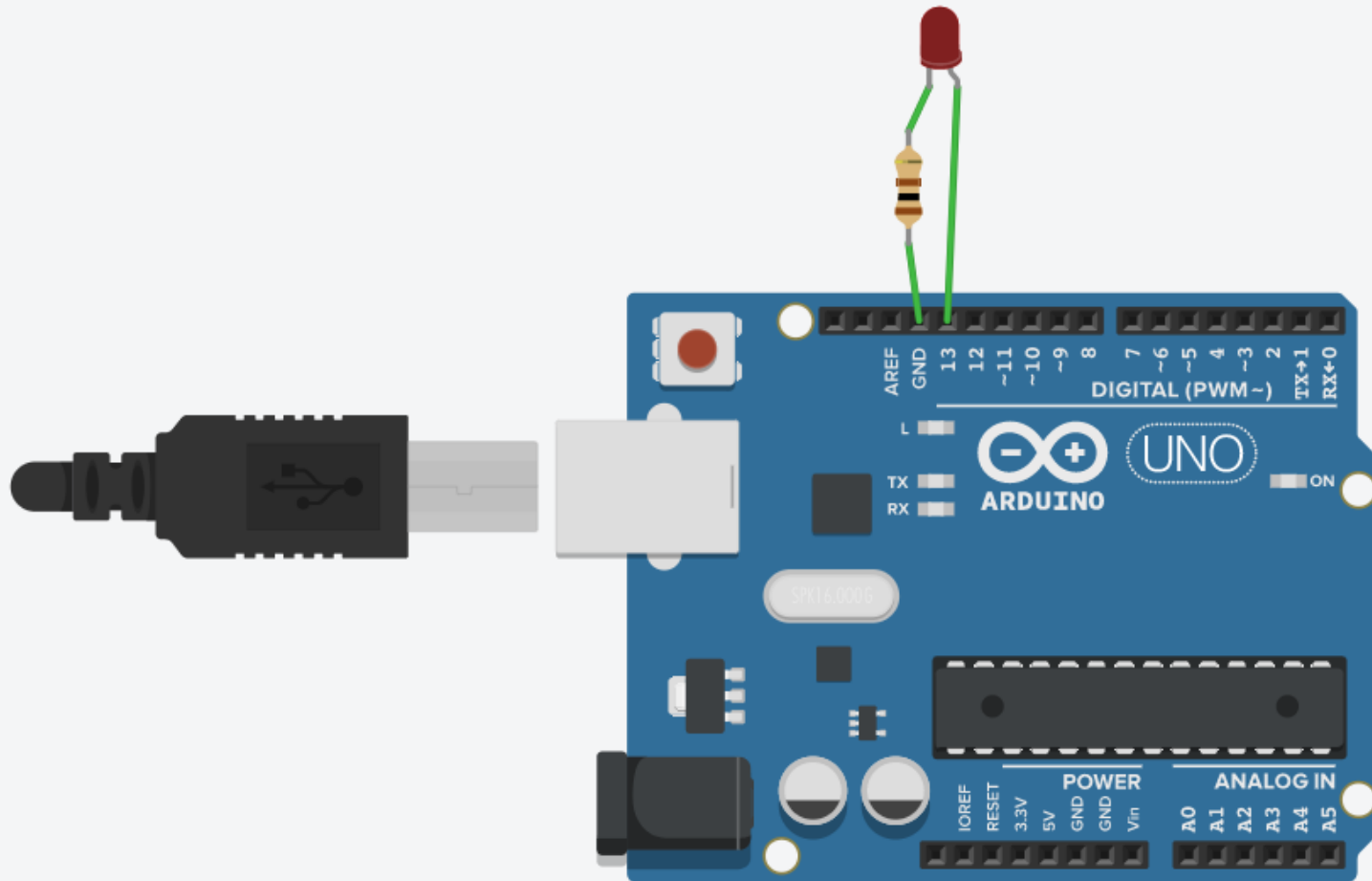
- `pinMode()`
- `digitalWrite()`
- `digitalRead()`
- `delay()`
- `analogWrite()`
- `analogRead()`
- Classe `Serial`
- Outras libs...

Componentes básicos





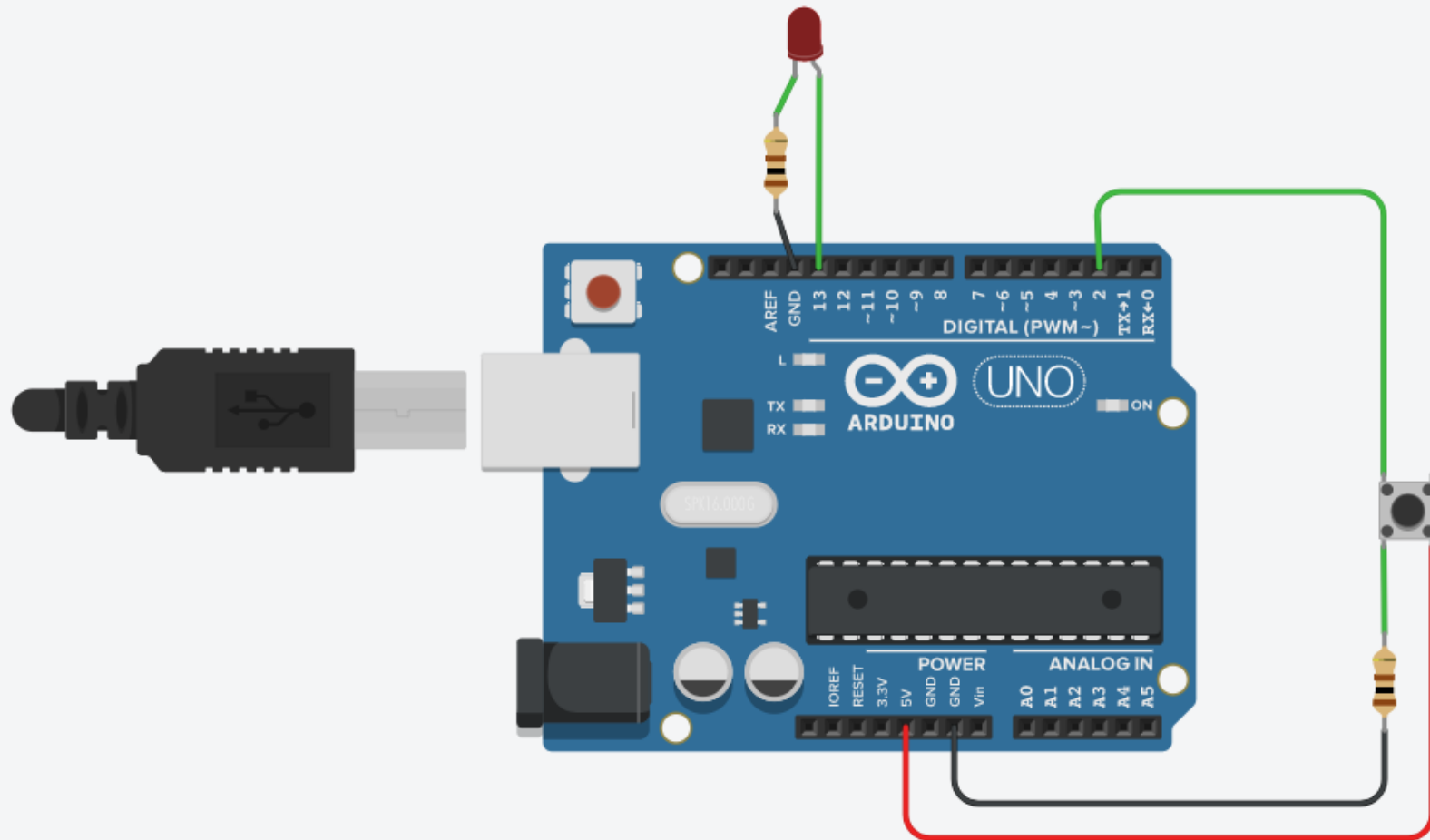
Blink



Blink

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

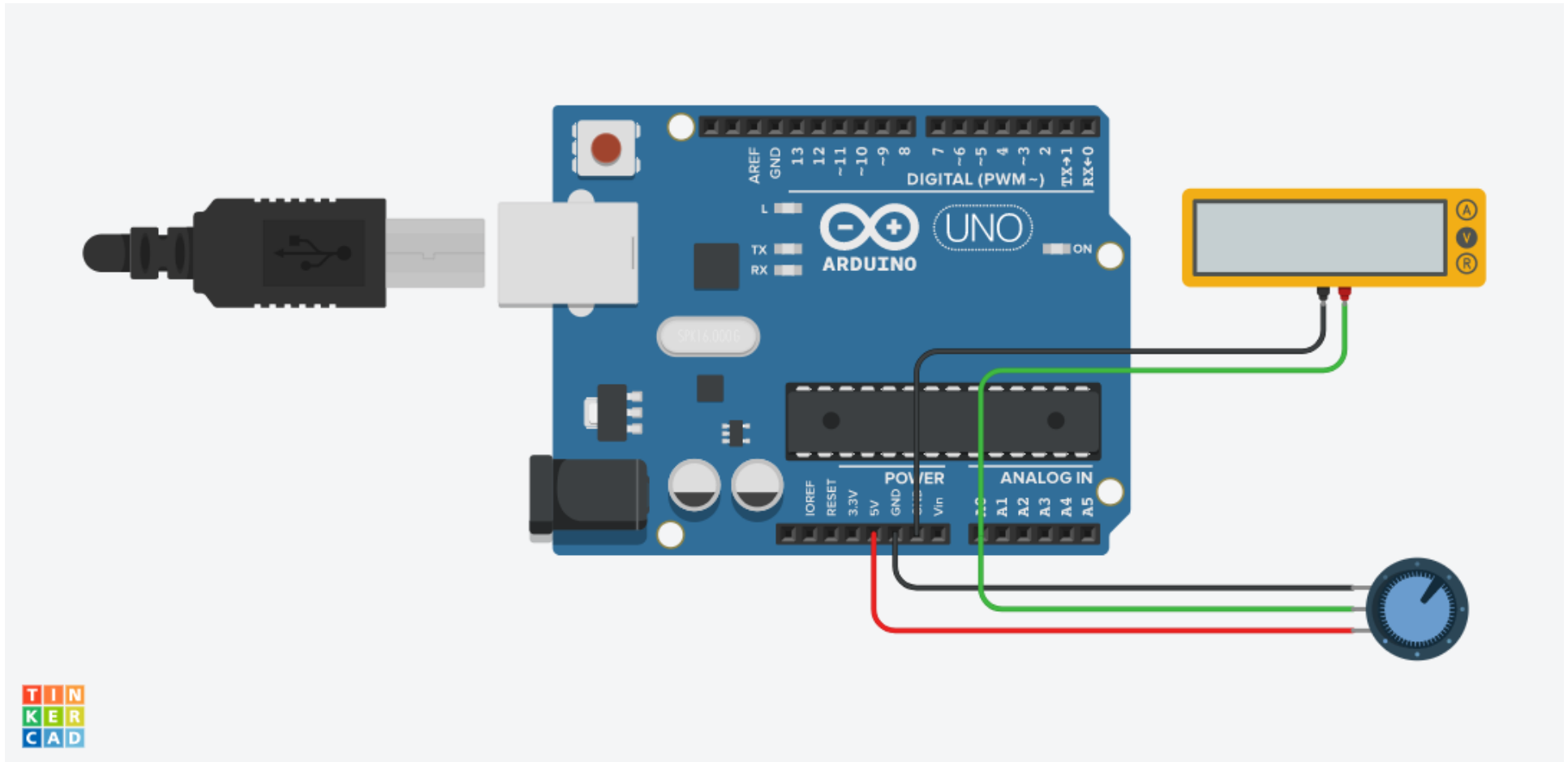
Botão e LED



Botão e LED

```
1 void setup()  
2 {  
3   pinMode(2, INPUT);  
4   pinMode(13, OUTPUT);  
5 }  
6  
7 void loop()  
8 {  
9   bool state = digitalRead(2);  
10  digitalWrite(13, state);  
11 }
```

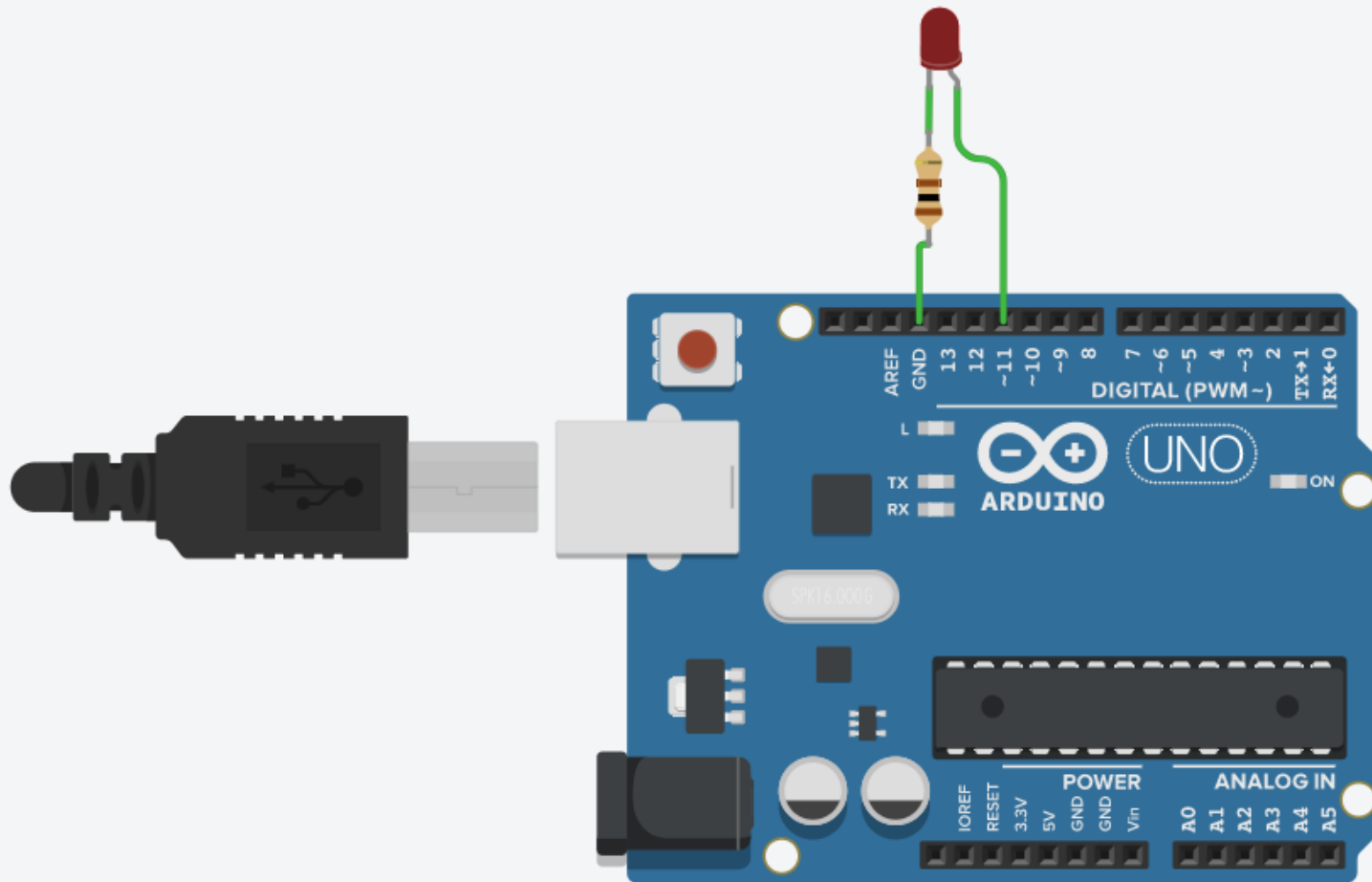
Entrada analógica



Entrada analógica

```
1 void setup()  
2 {  
3   Serial.begin(9600);  
4   pinMode(A0, INPUT);  
5 }  
6  
7 void loop()  
8 {  
9   int analogInput = analogRead(A0);  
10  Serial.println(analogInput);  
11 }
```

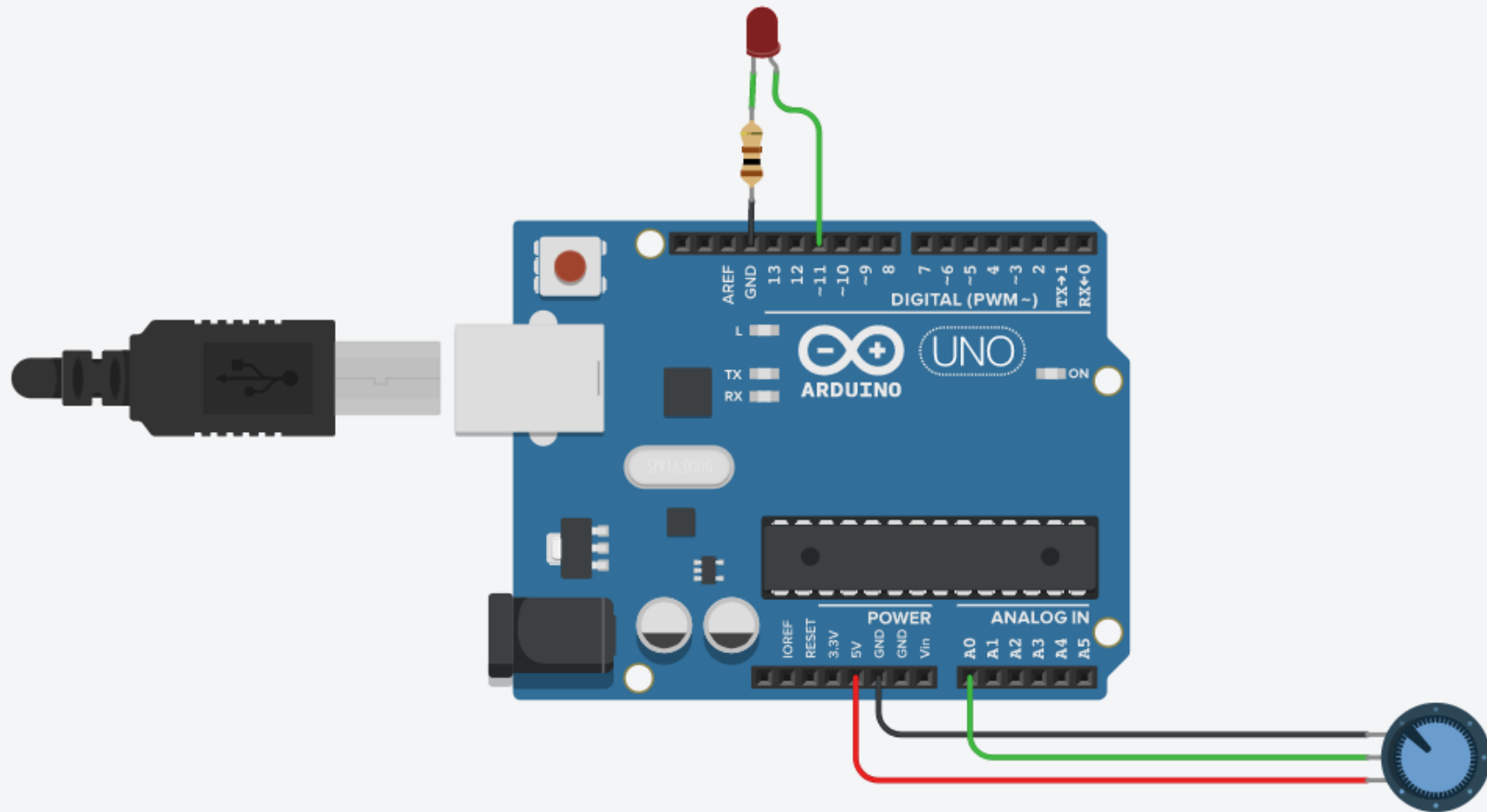
Saída analógica



Saída analógica

```
1  int fade = 0;
2
3  void setup()
4  {
5      pinMode(11, OUTPUT);
6  }
7
8  void loop()
9  {
10     for(fade = 0; fade <= 255; fade++)
11     {
12         analogWrite(11, fade);
13         delay(10);
14     }
15
16     for(fade = 255; fade > 0; fade--)
17     {
18         analogWrite(11, fade);
19         delay(10);
20     }
21
22     delay(500);
23 }
```

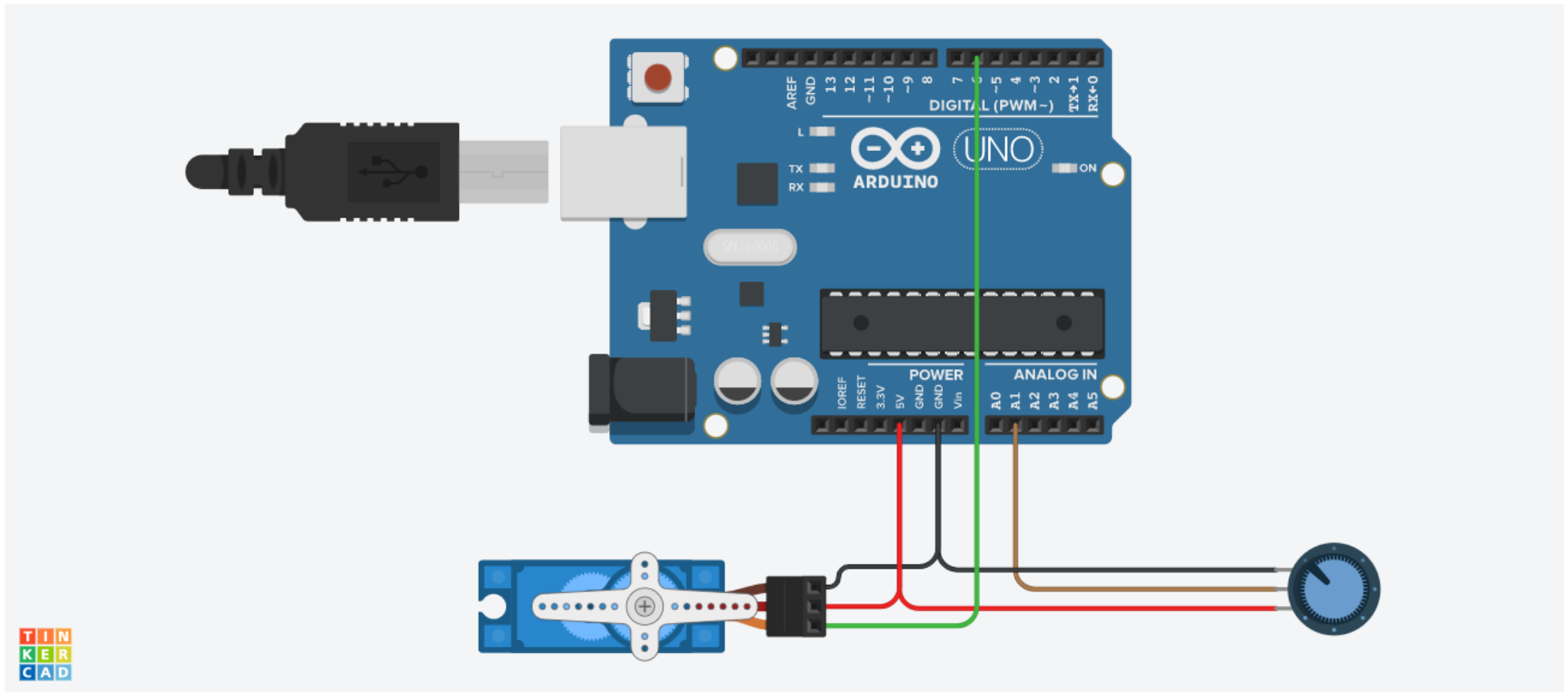
Entrada e saída analógicas



Entrada e saída analógicas

```
1 void setup()
2 {
3   pinMode(11, OUTPUT);
4 }
5
6 void loop()
7 {
8   int analogInput = analogRead(A0);
9
10  int analogOutput = map(analogInput, 0, 1023, 0, 255);
11
12  analogWrite(11, analogOutput);
13 }
```

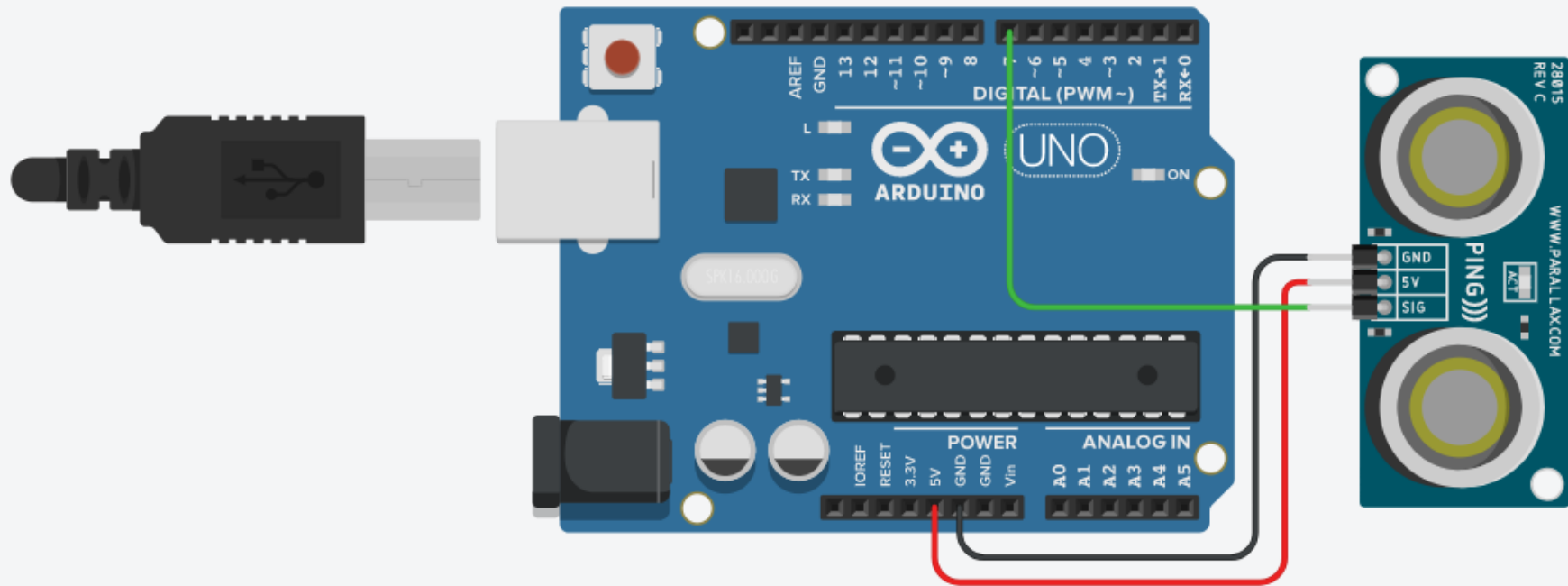
Potenciômetro e Servo



Potenciômetro e Servo

```
1  #include <Servo.h>
2
3  Servo servo;
4
5  void setup()
6  {
7      pinMode(A1, INPUT);
8      servo.attach(6);
9  }
10
11 void loop()
12 {
13     int analogInput = analogRead(A1);
14
15     int servoPosition = map(analogInput, 0, 1023, 180, 0);
16
17     servo.write(servoPosition);
18
19     delay(10);
20 }
```

Sensor ultrassônico



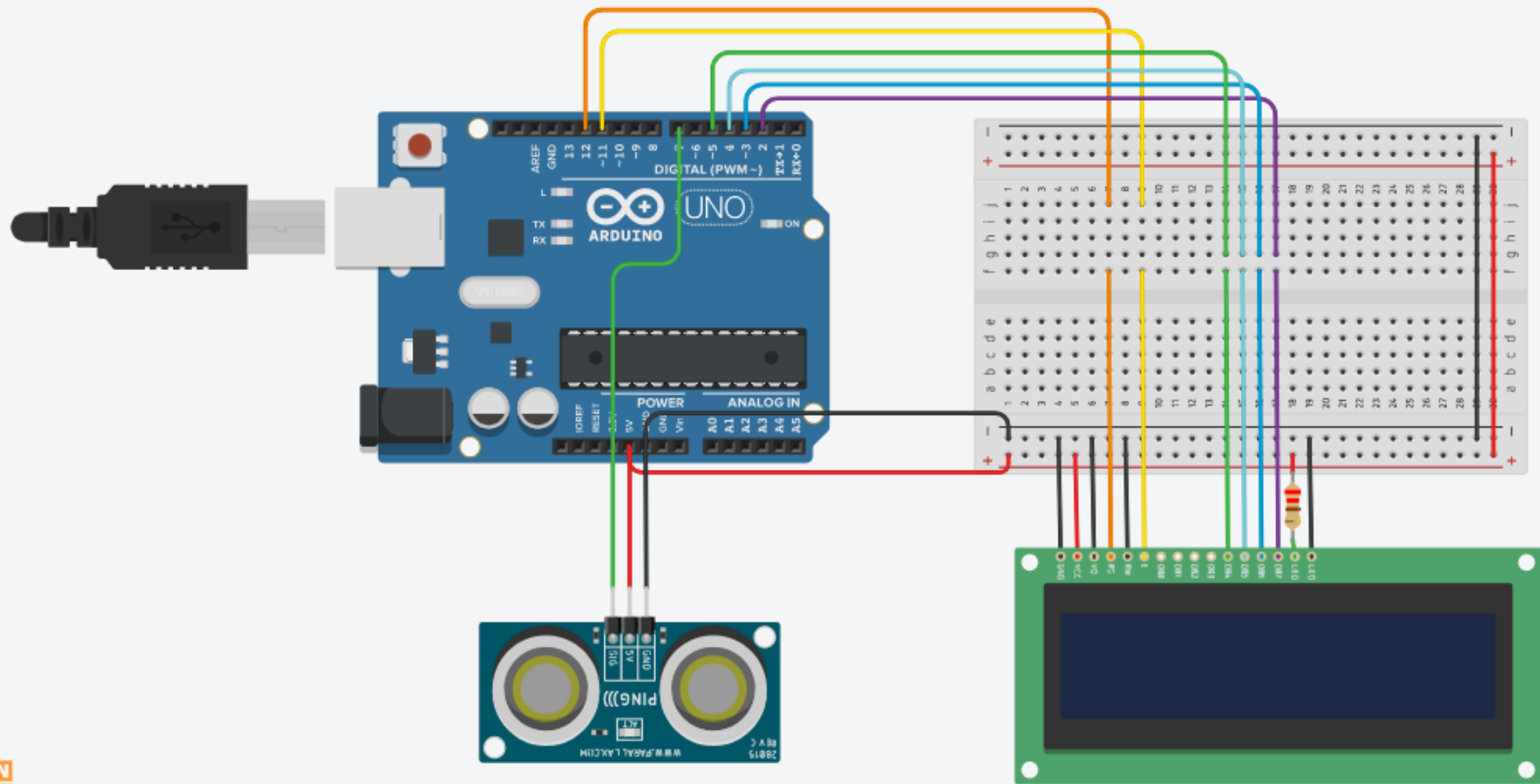
Sensor ultrassônico

```
1  long readUltrasonicDistance()  
2  {  
3      pinMode(7, OUTPUT);  
4  
5      digitalWrite(7, LOW);  
6      delayMicroseconds(2);  
7      digitalWrite(7, HIGH);  
8      delayMicroseconds(10);  
9      digitalWrite(7, LOW);  
10  
11     pinMode(7, INPUT);  
12  
13     return pulseIn(7, HIGH);  
14 }
```

Sensor ultrassônico

```
15 void setup()  
16 {  
17     pinMode(7, INPUT);  
18     Serial.begin(9600);  
19 }  
20  
21 /* Speed of sound is 343 m/s, or 0,0343 cm/us. Divides  
22    * by two because the pulse goes forward and backward. */  
23 const double soundSpeed = 343 * 100.0 / 1000000.0;  
24  
25 void loop()  
26 {  
27     double cm = (soundSpeed / 2) * readUltrasonicDistance();  
28  
29     Serial.print(cm);  
30     Serial.println("cm");  
31  
32     delay(100);  
33 }
```

Sensor ultrassônico e LCD



Sensor ultrassônico e LCD

```
20 LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
21
22 void setup()
23 {
24     lcd.begin(16, 2);
25     lcd.print("Hello world!");
26 }
27
28 void loop()
29 {
30     lcd.setCursor(0, 1);
31     lcd.print("                ");
32     lcd.setCursor(0, 1);
33     String cm = String((soundSpeed / 2) * readUltrasonicDistance());
34     String msg = String(cm + " cm");
35     lcd.print(msg);
36
37     delay(500);
38 }
```


Referências

- <https://www.arduino.cc>
- <https://www.arduino.cc/reference/en>
- <https://www.tinkercad.com>
- <https://www.arduino.cc/en/Tutorial/Knob>
- <http://www.arduino.cc/en/Tutorial/Ping>
- <http://www.arduino.cc/en/Tutorial/LiquidCrystal>

github.com/fkuhne/aday18

