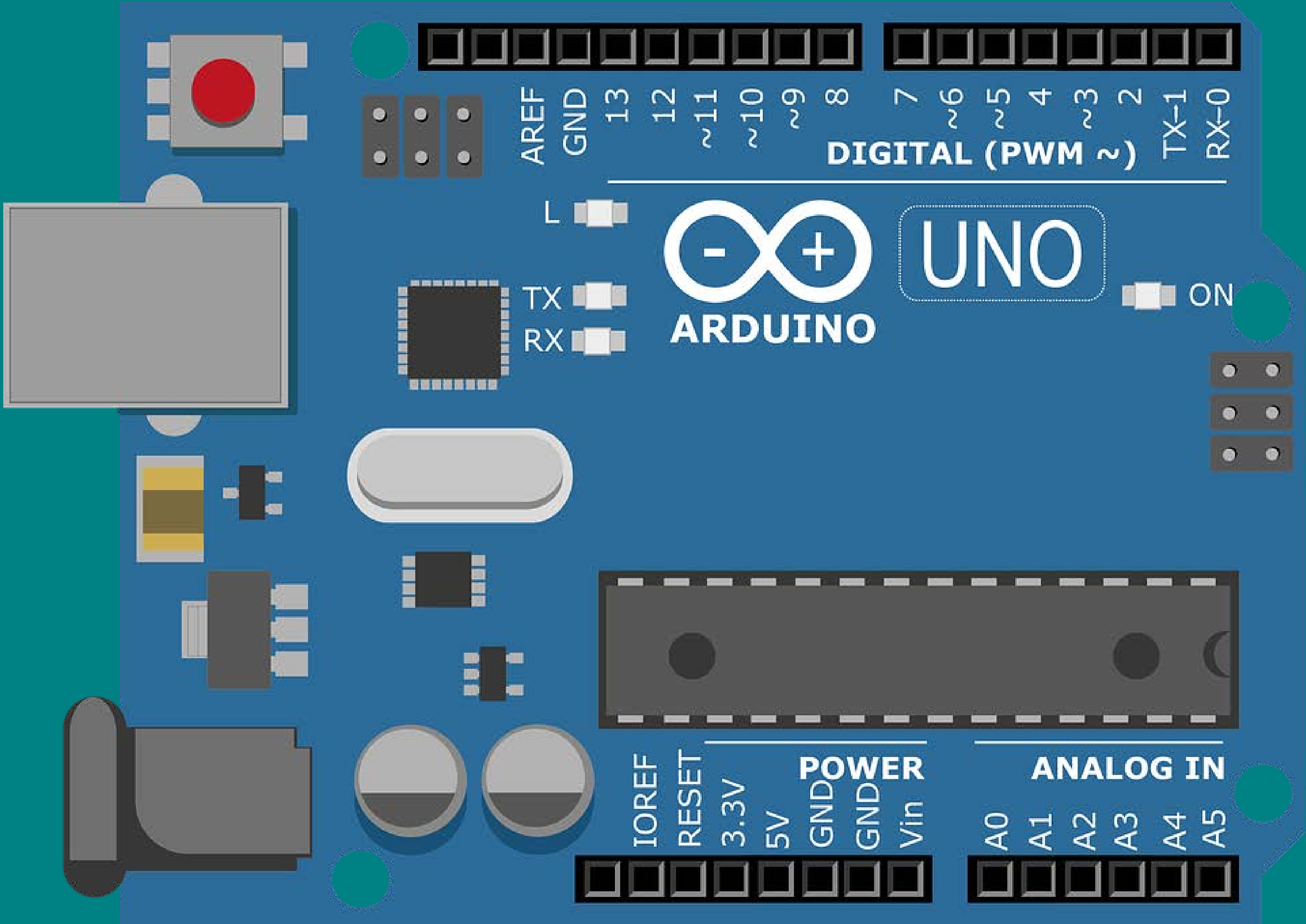


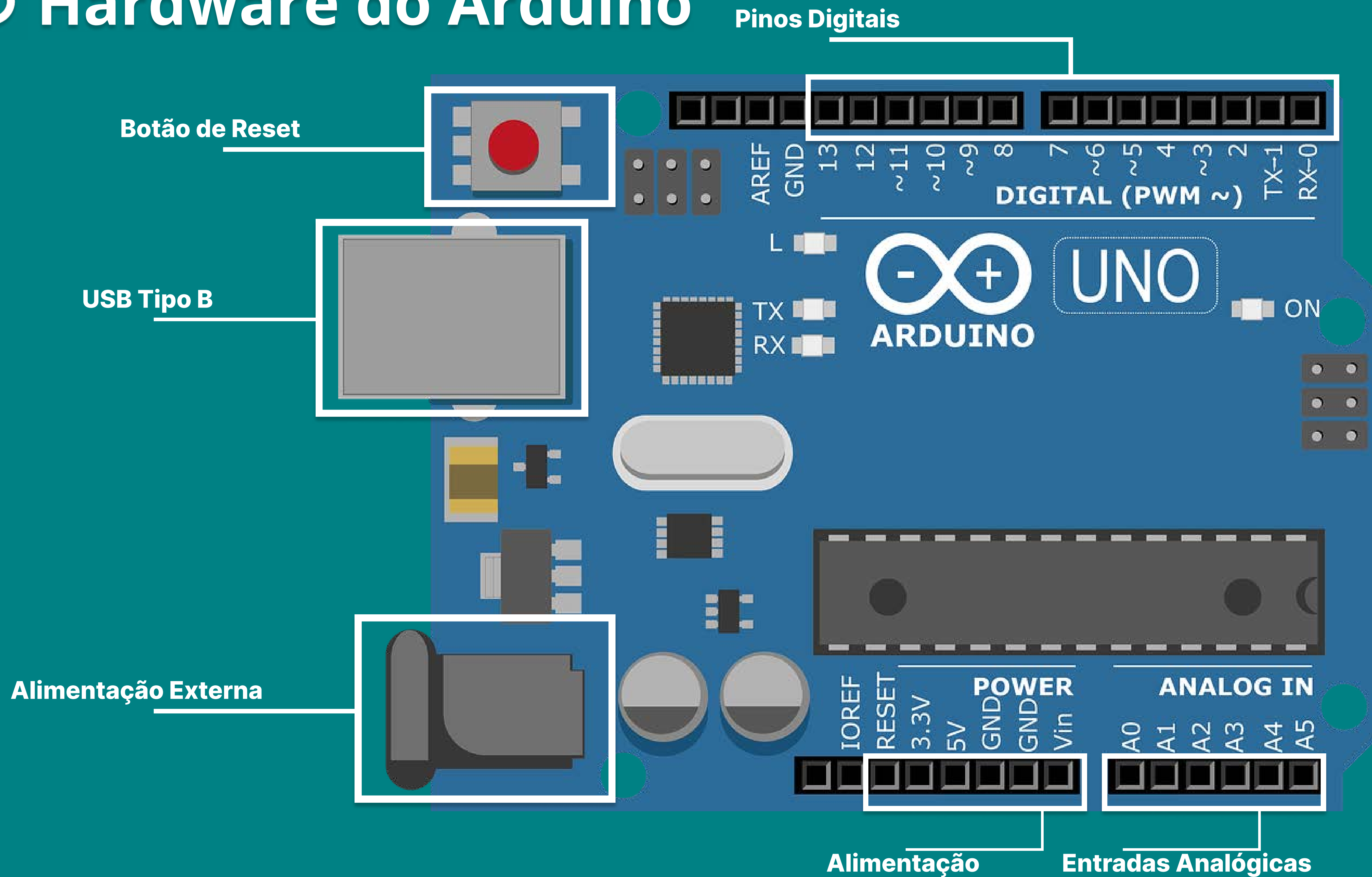
Introdução ao kit de desenvolvimento Arduino

O Que é o Arduino?

Hardware do Arduino



Hardware do Arduino



Programação do Arduino

A linguagem de programação do Arduino pode ser dividida em:

Estruturas

- **Preâmbulo**
- **Setup**
- **Loop**

Variáveis

- **Constantes:**
INPUT
OUTPUT
HIGH
LOW
- **Tipos de Dados**
int
float
char
boolean

Funções

- **E/S digitais**
pinMode()
digitalWrite()
digitalRead()
- **E/S analógicas**
analogWrite()
analogRead()
- **Temporizador**
delay()

Programação do Arduino

Preâmbulo

É o local empregado para a definição dos elementos que serão utilizados ao longo do restante do código.

Setup

A função *setup()* é utilizada para configurar o modo dos pinos que serão utilizados durante o código. Ela é executada somente uma vez quando a placa é iniciada.

Loop

A função *loop()* é definida logo após o término do *Setup*. Ela é executada repetidamente.

Programação do Arduino



```
int pinoLed = 10;
```

Preâmbulo

```
void setup( ) {
```

```
  pinMode(pinoLed, OUTPUT);
```

```
}
```

Setup

```
void loop( ) {
```

```
  digitalWrite(pinoLed, HIGH);
```

```
  delay(1000);
```

```
  digitalWrite(pinoLed, LOW);
```

```
  delay(1000);
```

```
}
```

Loop

Alguns elementos utilizados na linguagem:

Operadores

= (atribuição)
* (multiplicação)
+ (adição)
- (subtração)
/ (divisão)
% (resto)

Controle

```
if (condição) {  
  
}  
else {  
  
}
```

Repetição

```
while (condição) {  
  
}  
  
for (...) {  
  
}
```


Onde programar o Arduino?

IDE do Arduino

O Arduino possui o próprio IDE (Ambiente de Desenvolvimento Integrado).

Encontrado em:
[arduino.cc](https://www.arduino.cc)



```
sketch_mar26a
void setup() {
  // put your setup code here, to run once:
}

void loop() {
  // put your main code here, to run repeatedly:
}
```

Uma alternativa é utilizar o ***Tinkercad***, que é um aplicativo Web gratuito para projetos 3D, eletrônica e codificação.



AUTODESK®
TINKERCAD®

**AUTODESK**
Tinkercad

Tinker ▾ Galeria Projetos Salas de aula Recursos ▾




Setel 2023 - Arduino

Pesquisar projetos...

Aulas

Projetos

Tutoriais

Coleções

 Criar coleção

Seus projetos

 Criar

Projetos 3D


Crie seu primeiro projeto 3D


Place It


View It


Move It

Circuitos

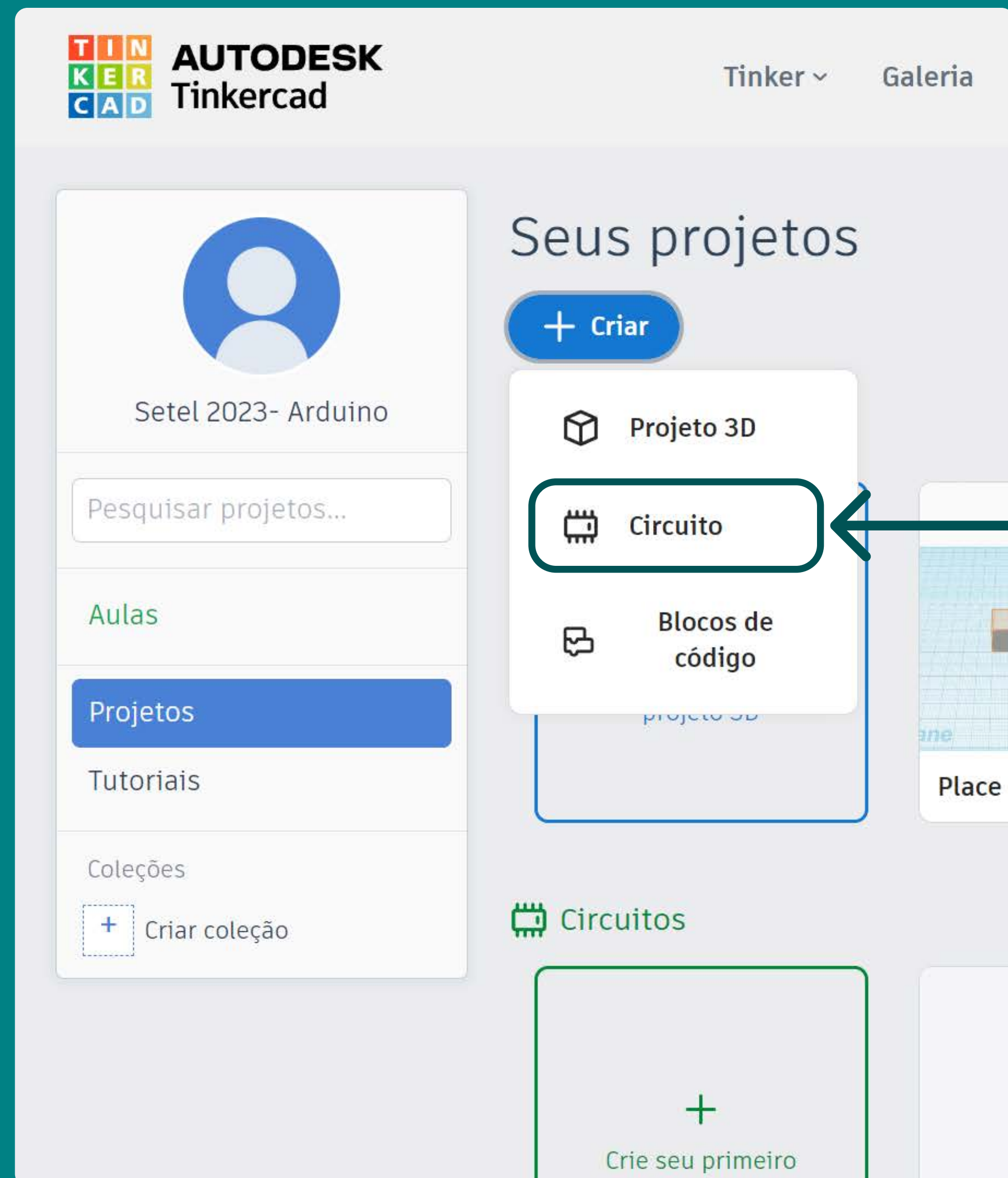
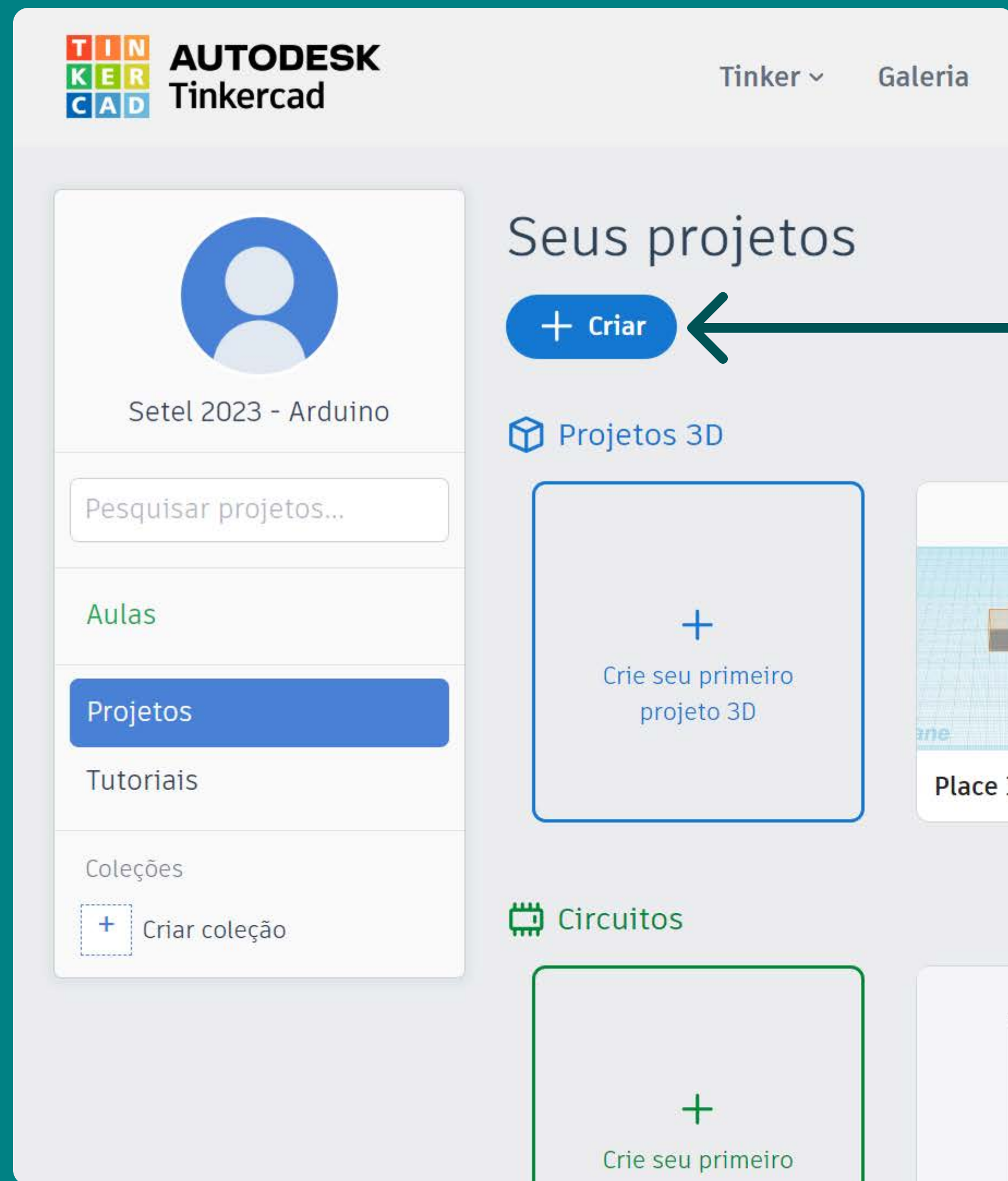

Crie seu primeiro

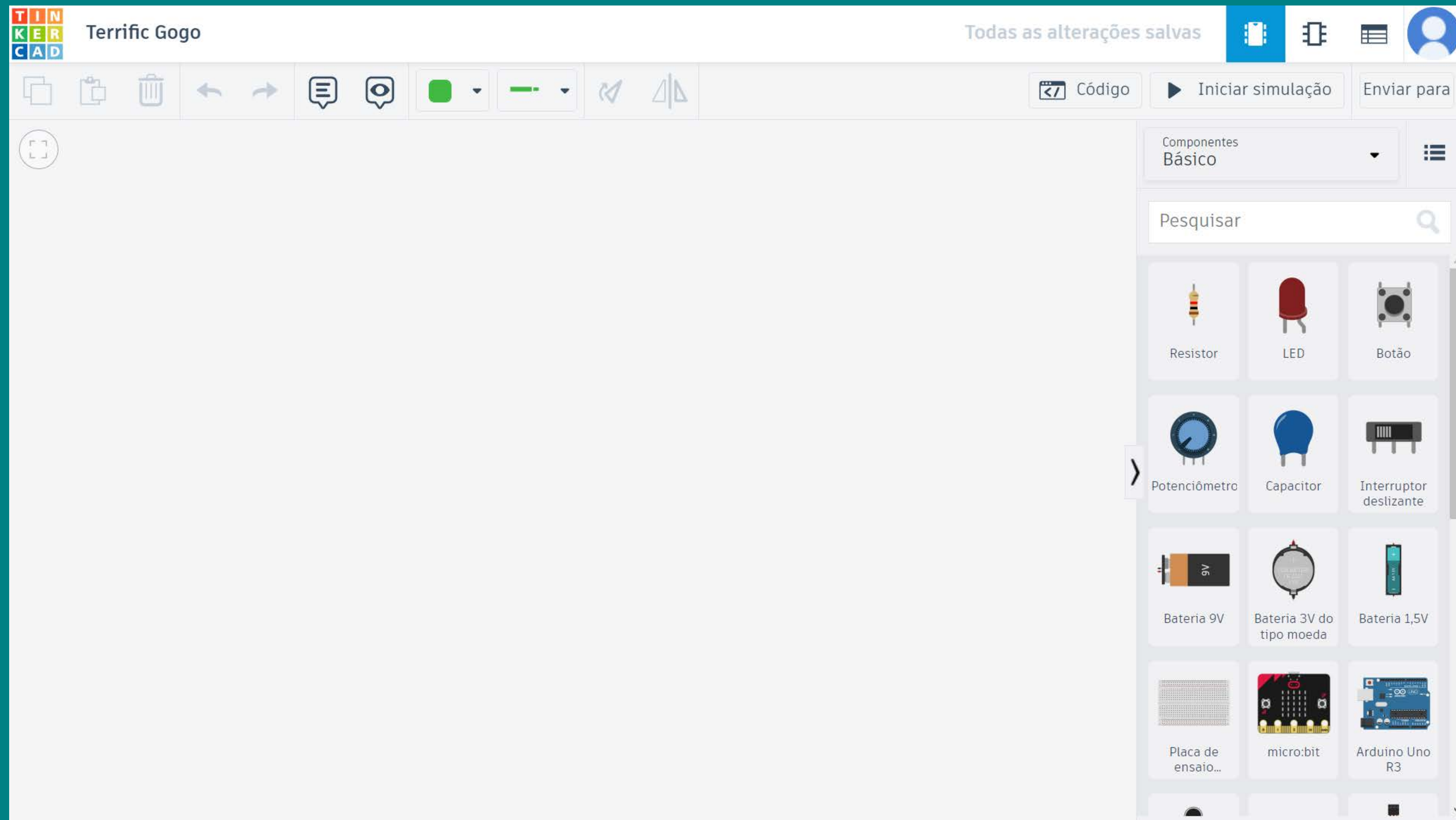






Novidades ▾

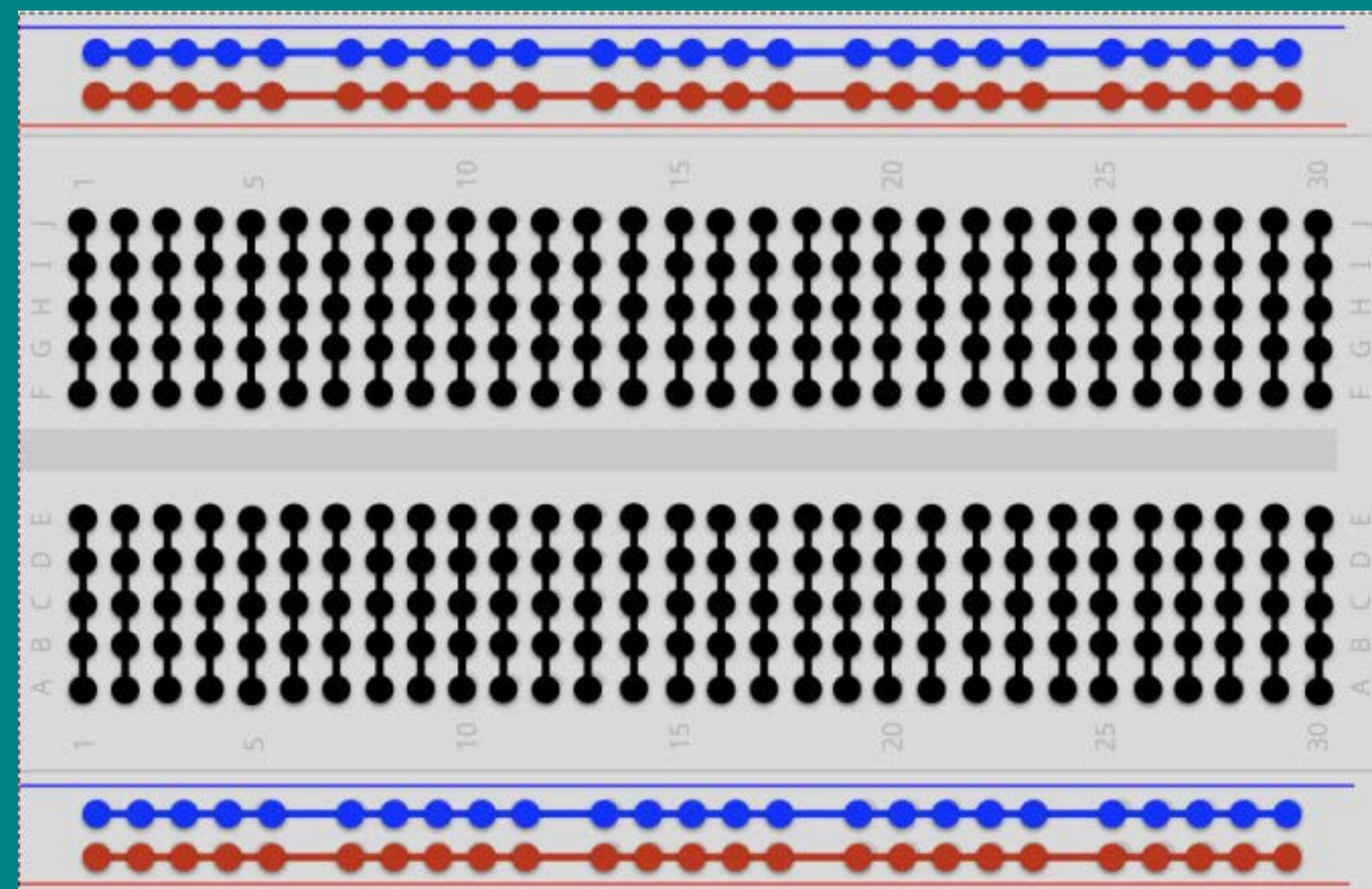
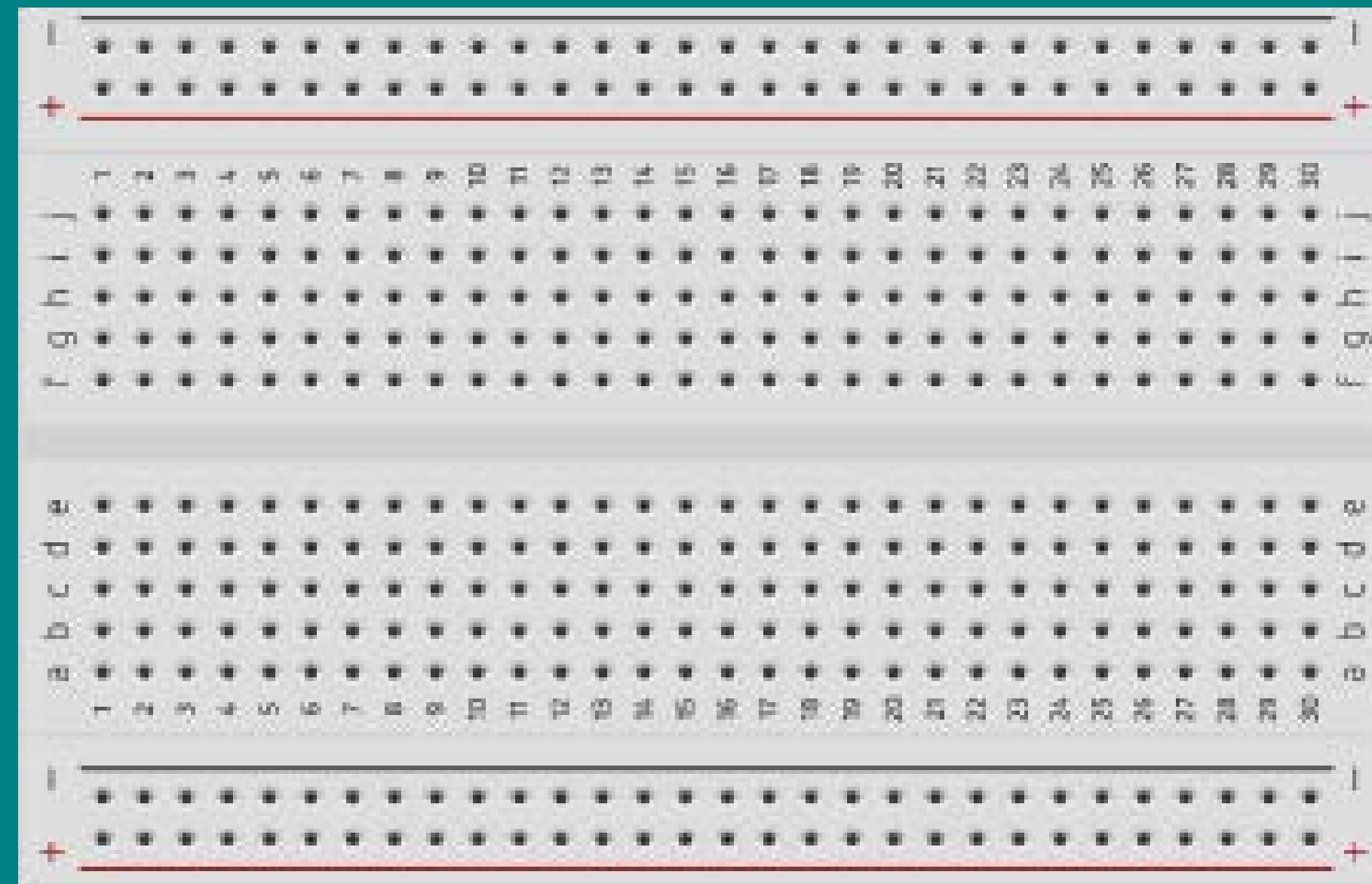




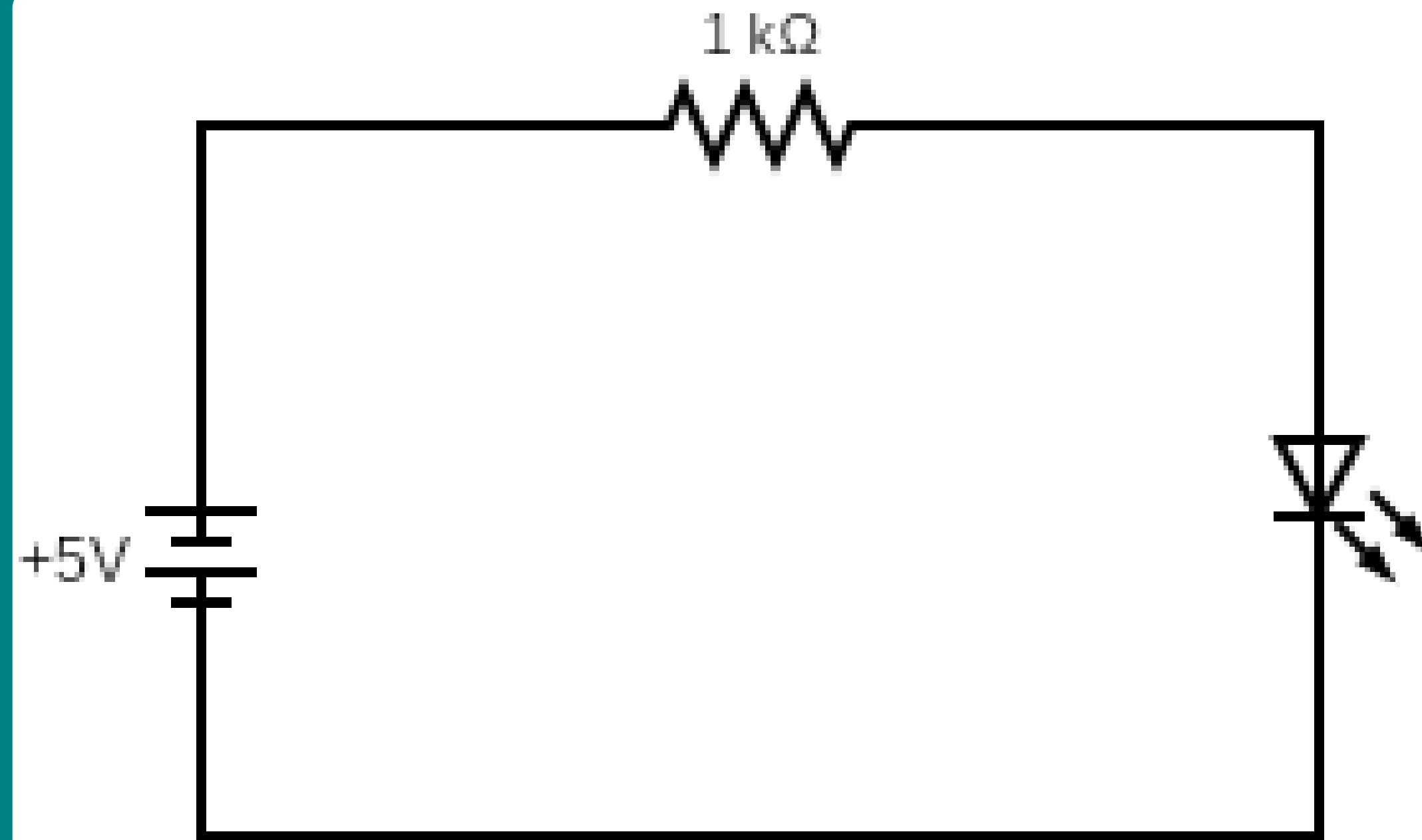
Como montar os circuitos?

É utilizado uma placa de prototipagem chamada protoboard

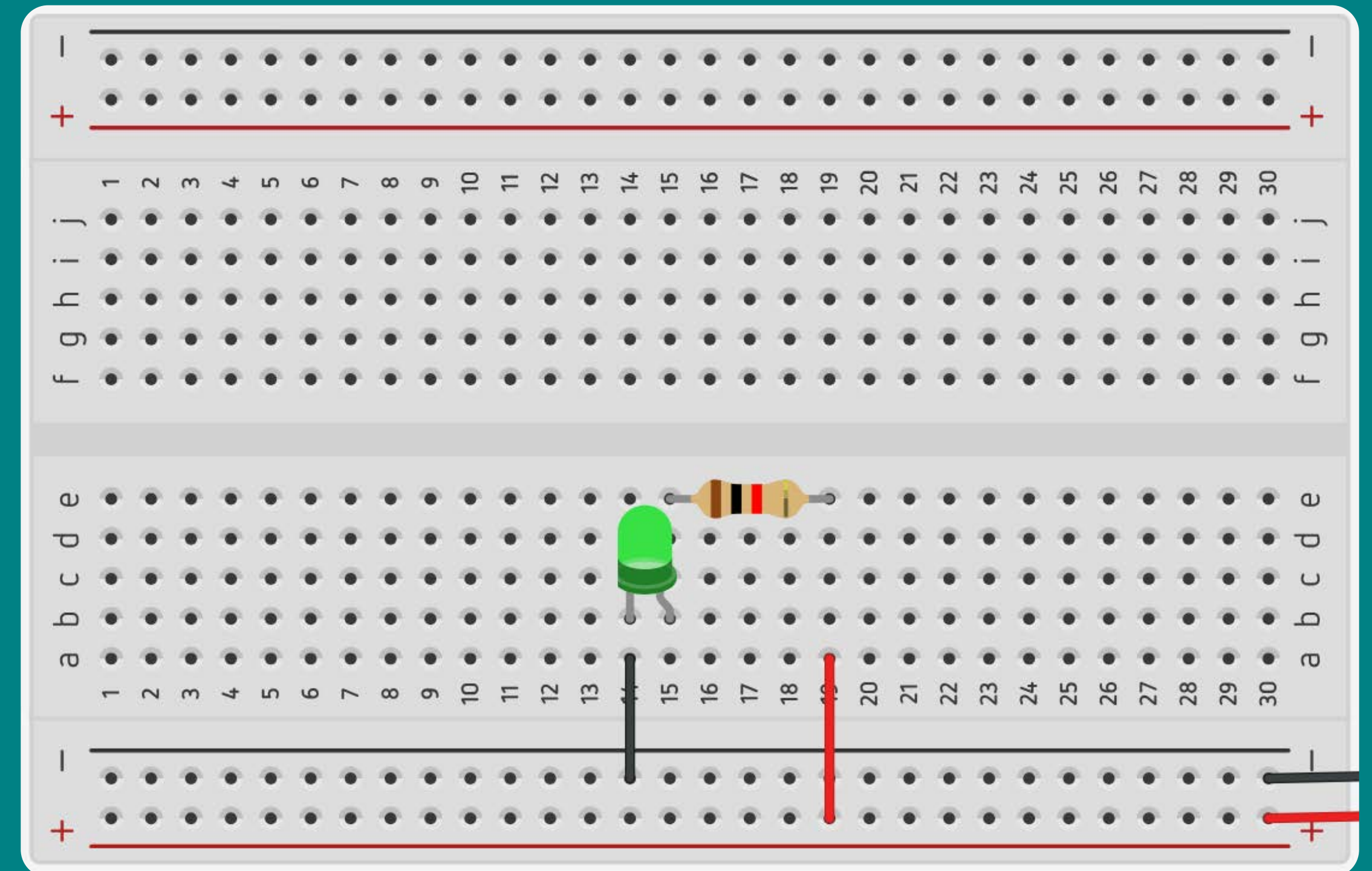
É uma matriz de contatos que possui furos e conexões condutoras.



Esquema elétrico:



Montagem:



Interação com LEDs

∞+ Interação com LEDs

O que são *LEDs*

Os LEDs (Light Emitting Diodes) são componentes semicondutores que possuem a capacidade de transformar energia em luz.

São componentes bipolares, ou seja, possuem um terminal anodo e o outro catodo.

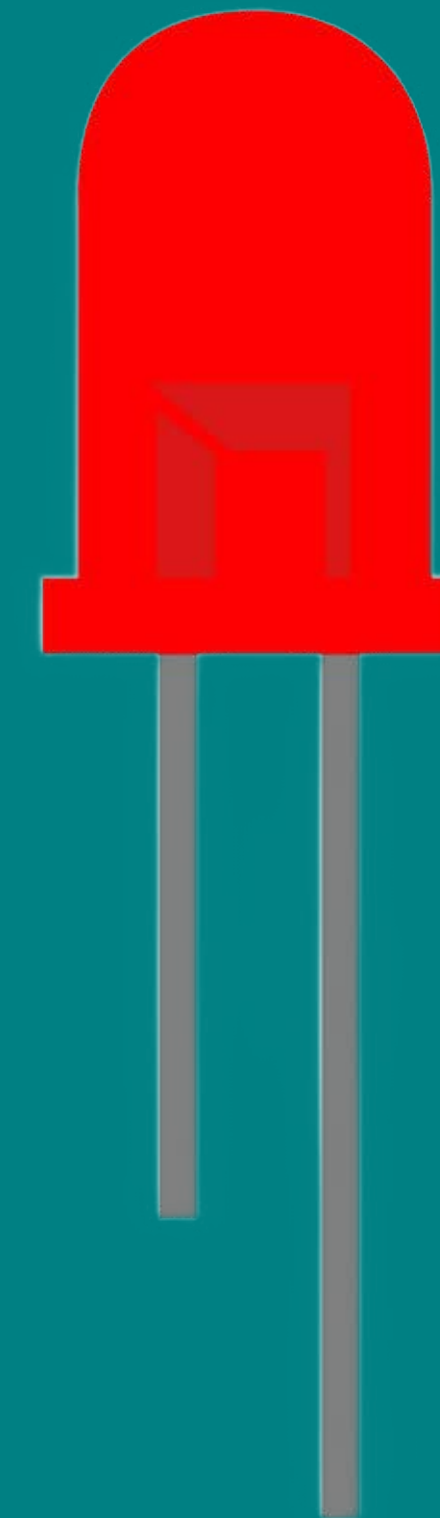


Figura: LED Vermelho

⊖⊕ Interação com LEDs

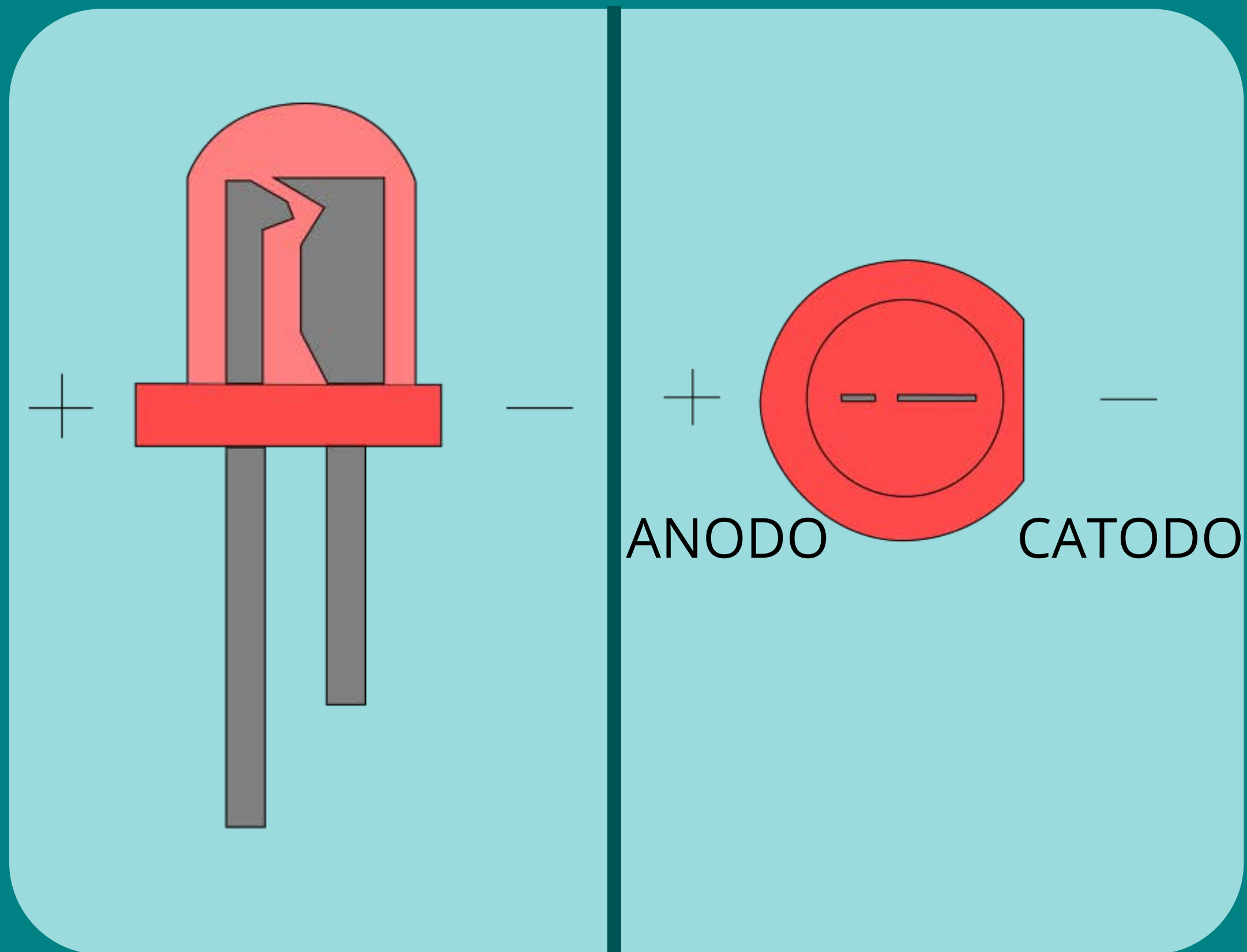


Figura: Polaridade do LED

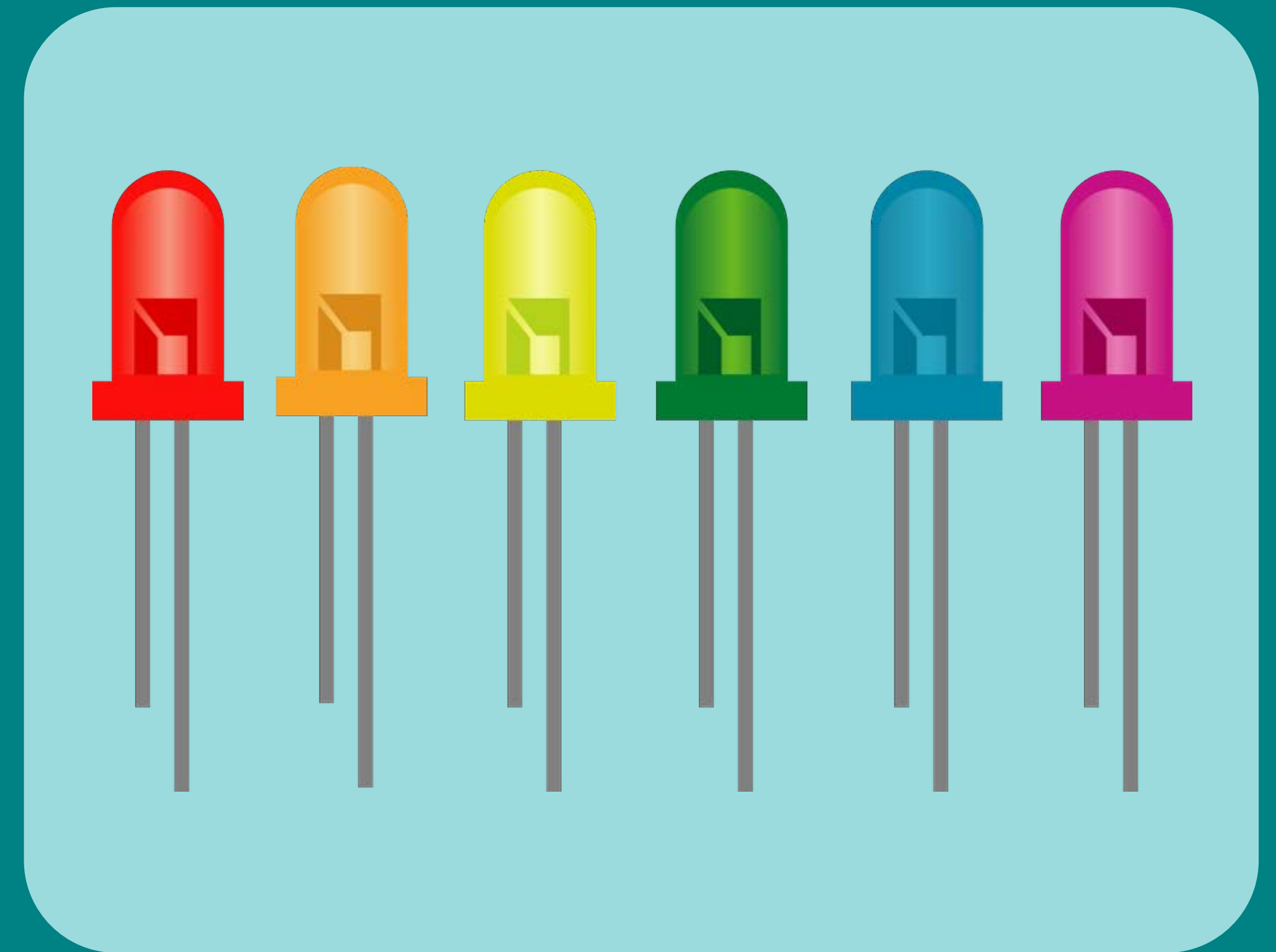


Figura: LED com cores variadas

∞+ Interação com LEDs

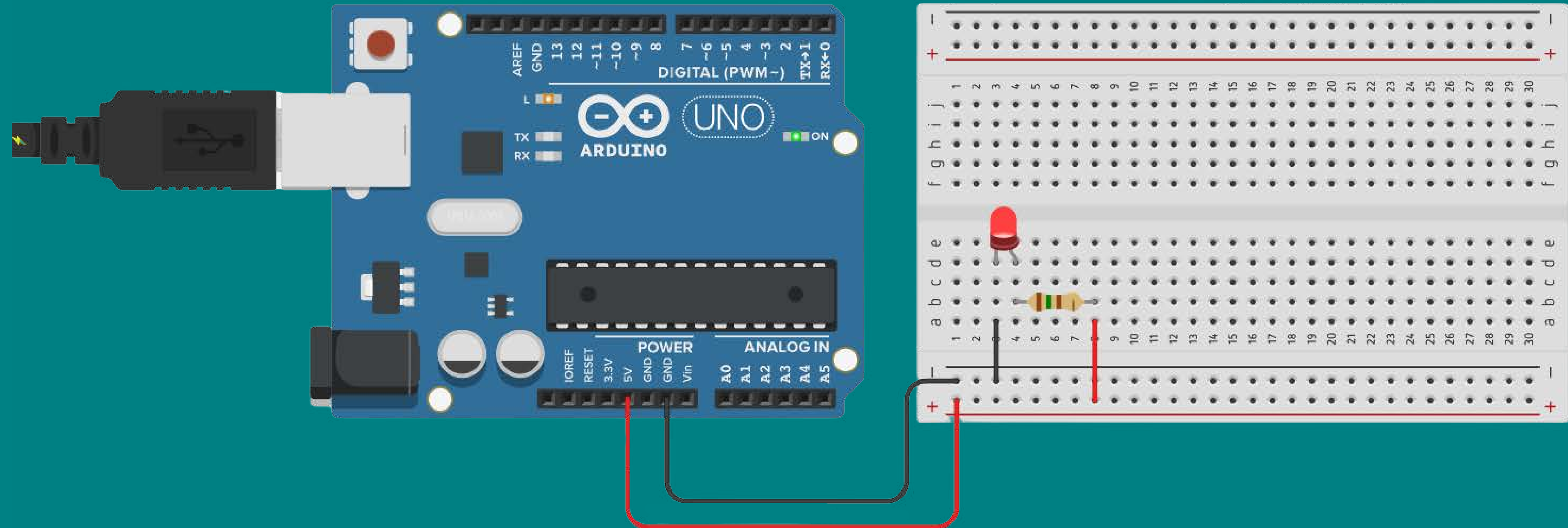


Figura: Circuito LED Ativo

Interação com LEDs

pinMode(pino, modo)

A função **pinMode()** configura o *pino* especificado para funcionar como entrada (**INPUT**) ou saída (**OUTPUT**).

```
void setup() {  
    pinMode(10, OUTPUT);  
}
```

Figura: Exemplo pinMode()

digitalWrite(pino, valor)

A função **digitalWrite()** aciona o valor 1 (**HIGH**) ou 0 (**LOW**) em um pino digital.

```
void loop() {  
    digitalWrite(10, HIGH);  
}
```

Figura: Exemplo digitalWrite()

∞+ Interação com LEDs

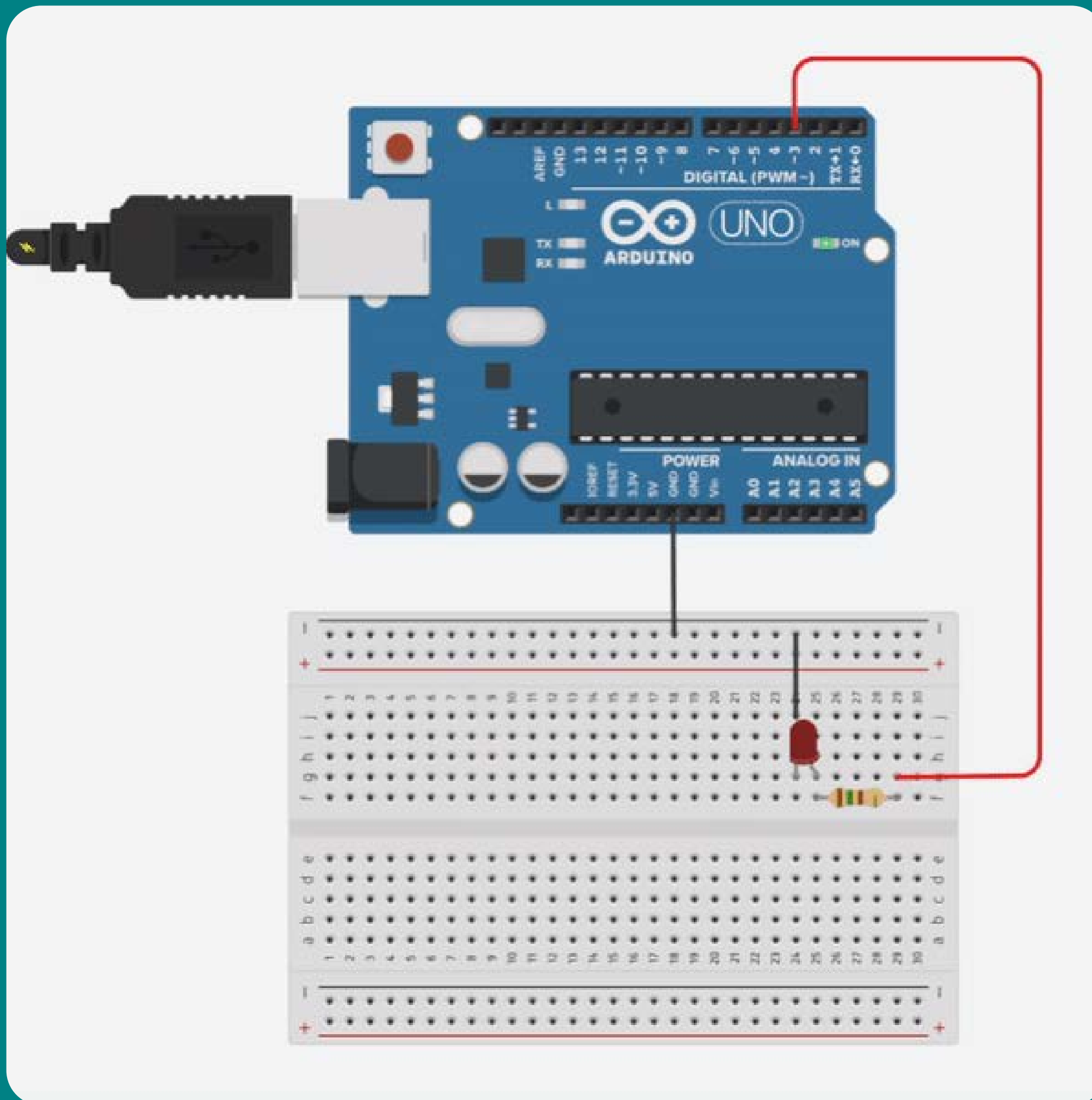


Figura: Circuito LED Piscando

```
int pinoLed = 3;

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

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

Figura: Código LED Piscando

Como fazer um “semáforo” usando LEDs?



∞+ Interação com LEDs

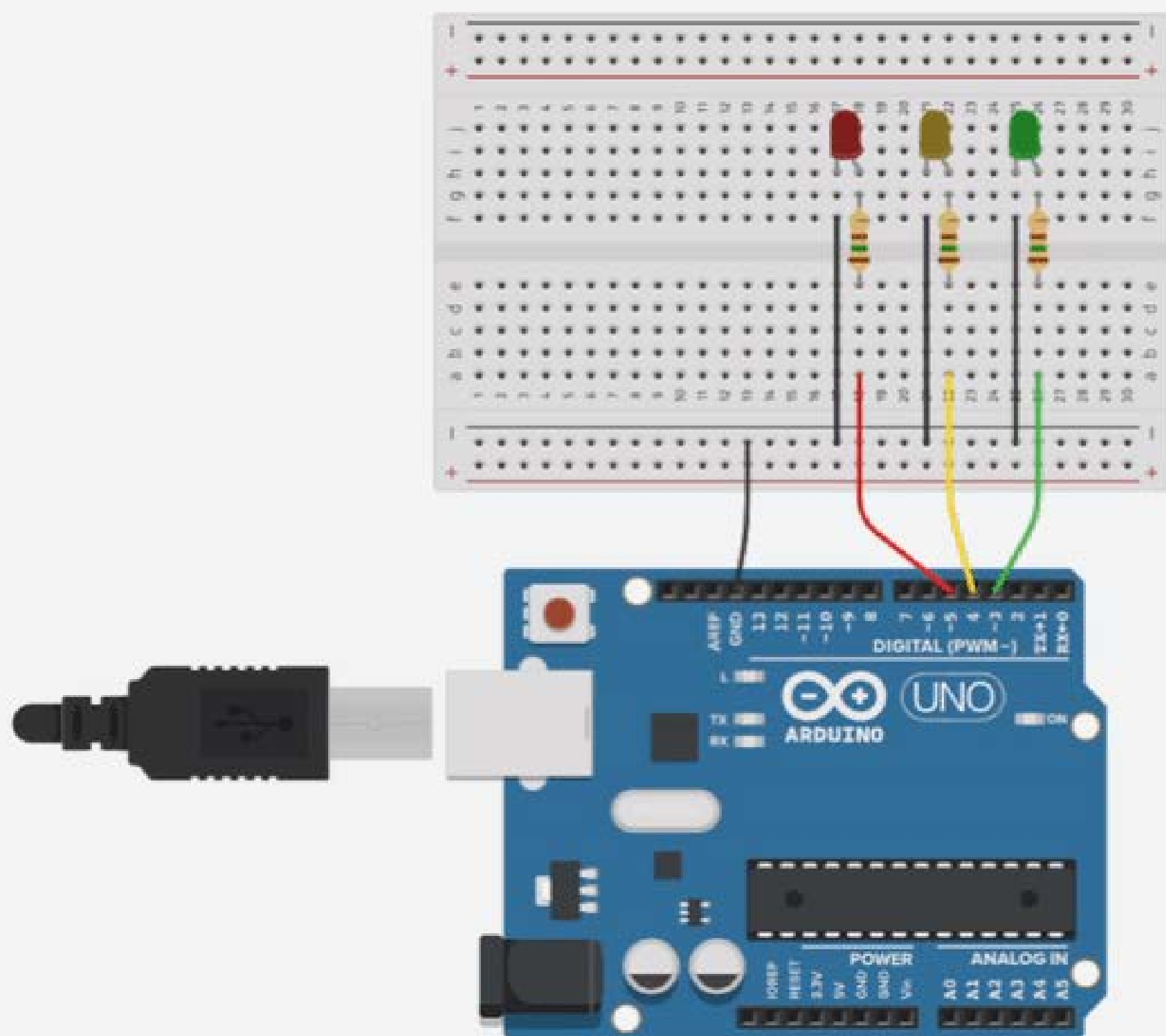


Figura: Circuito LED Semáforo

```
int pinoLedVermelho = 5;
int pinoLedAmarelo = 4;
int pinoLedVerde = 3;

void setup(){
  pinMode(pinoLedVermelho, OUTPUT);
  pinMode(pinoLedVerde, OUTPUT);
  pinMode(pinoLedAmarelo, OUTPUT);
}

void loop(){
  digitalWrite(pinoLedVermelho, HIGH);
  delay(3000);
  digitalWrite(pinoLedVermelho, LOW);

  digitalWrite(pinoLedAmarelo, HIGH);
  delay(1000);
  digitalWrite(pinoLedAmarelo, LOW);

  digitalWrite(pinoLedVerde, HIGH);
  delay(2000);
  digitalWrite(pinoLedVerde, LOW);
}
```

Figura: Código LED Semáforo

Interação com LED RGB

∞+ Interação com LED RGB

O que é o *LED RGB*

Assim como os LEDs tradicionais, o LED RGB emite luz por meio da passagem de corrente elétrica, mas ele apresenta a peculiaridade de ser composto por três LEDs de cores diferentes, sendo elas: vermelho (*Red*), verde (*Green*) e azul (*Blue*), surgindo daí o nome RGB.

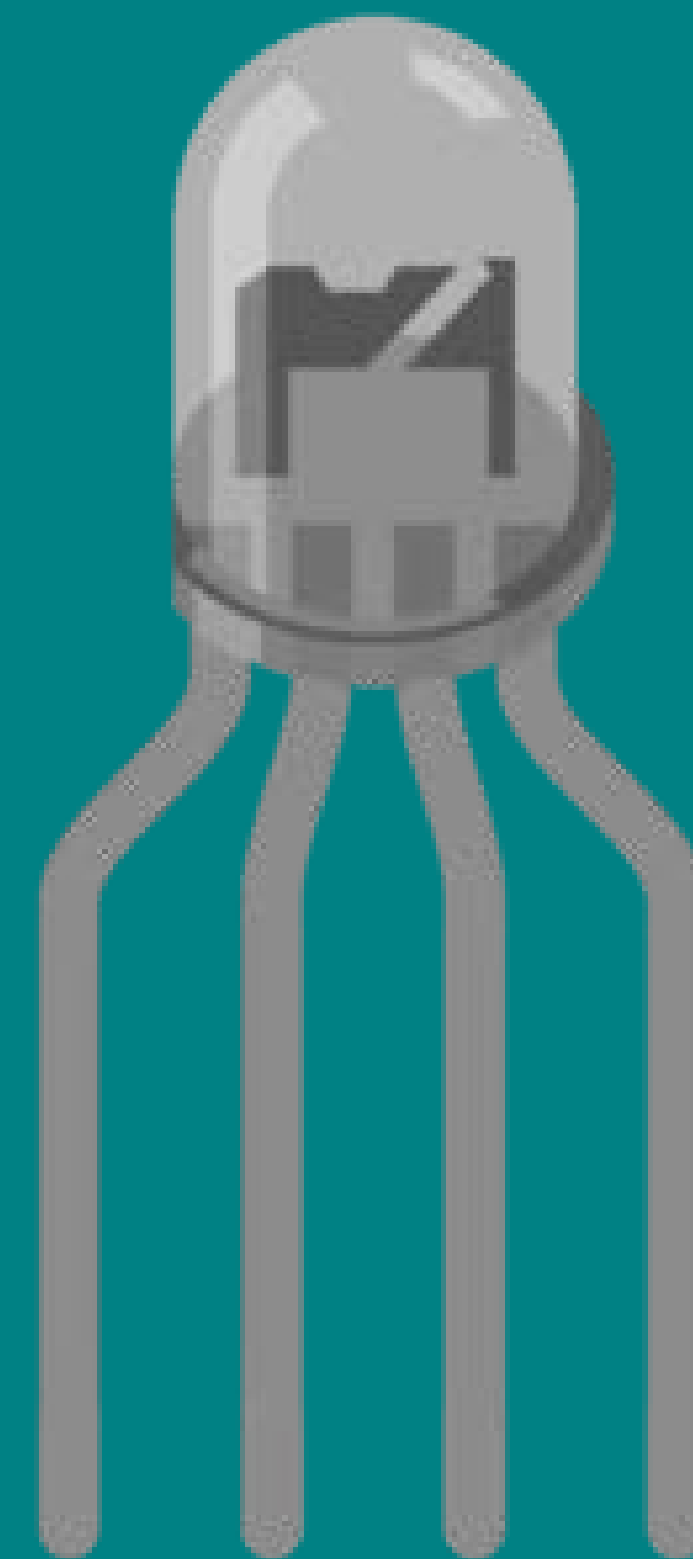


Figura: LED RGB

⊖⊕ Interação com LED RGB

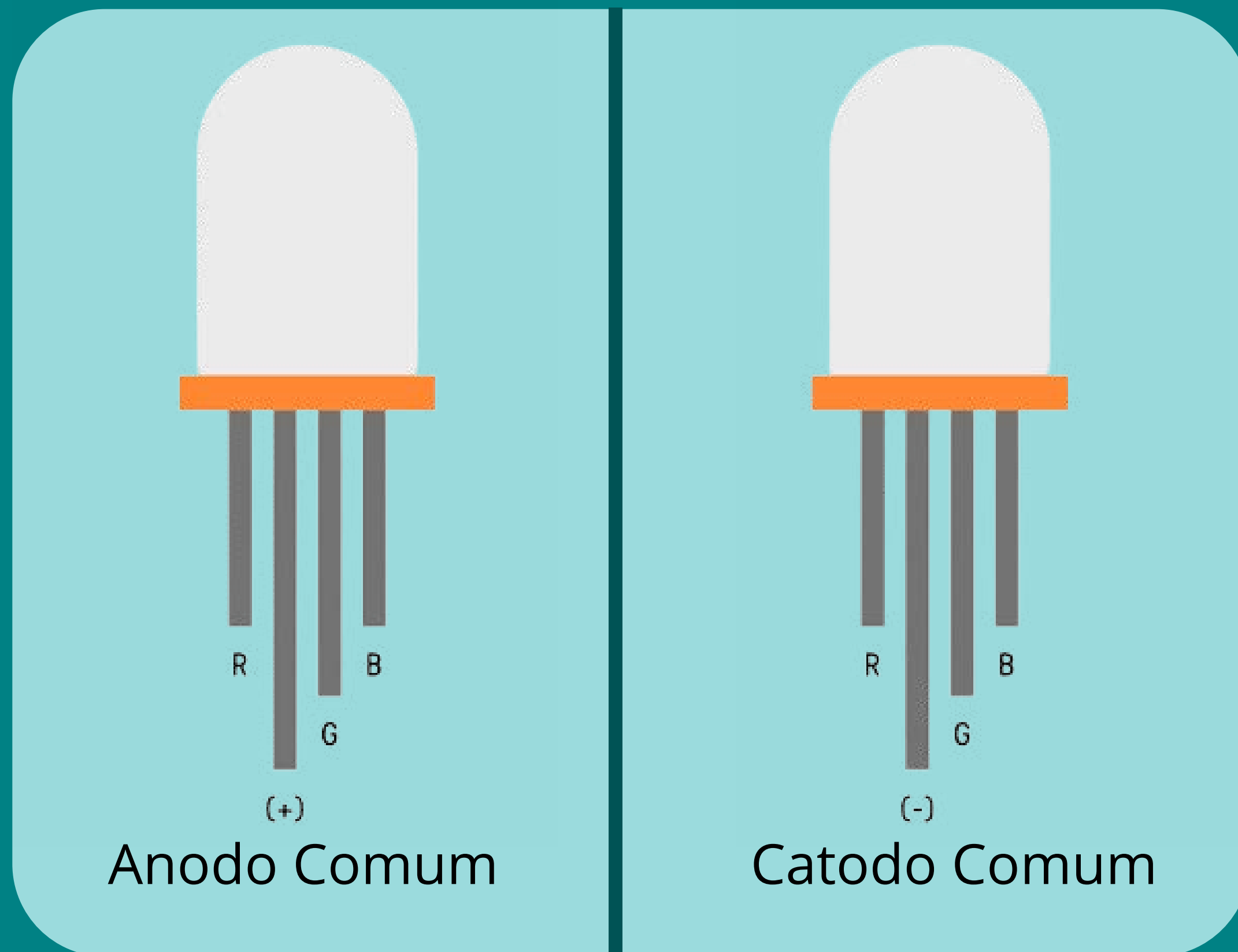


Figura: Polaridade do LED RGB

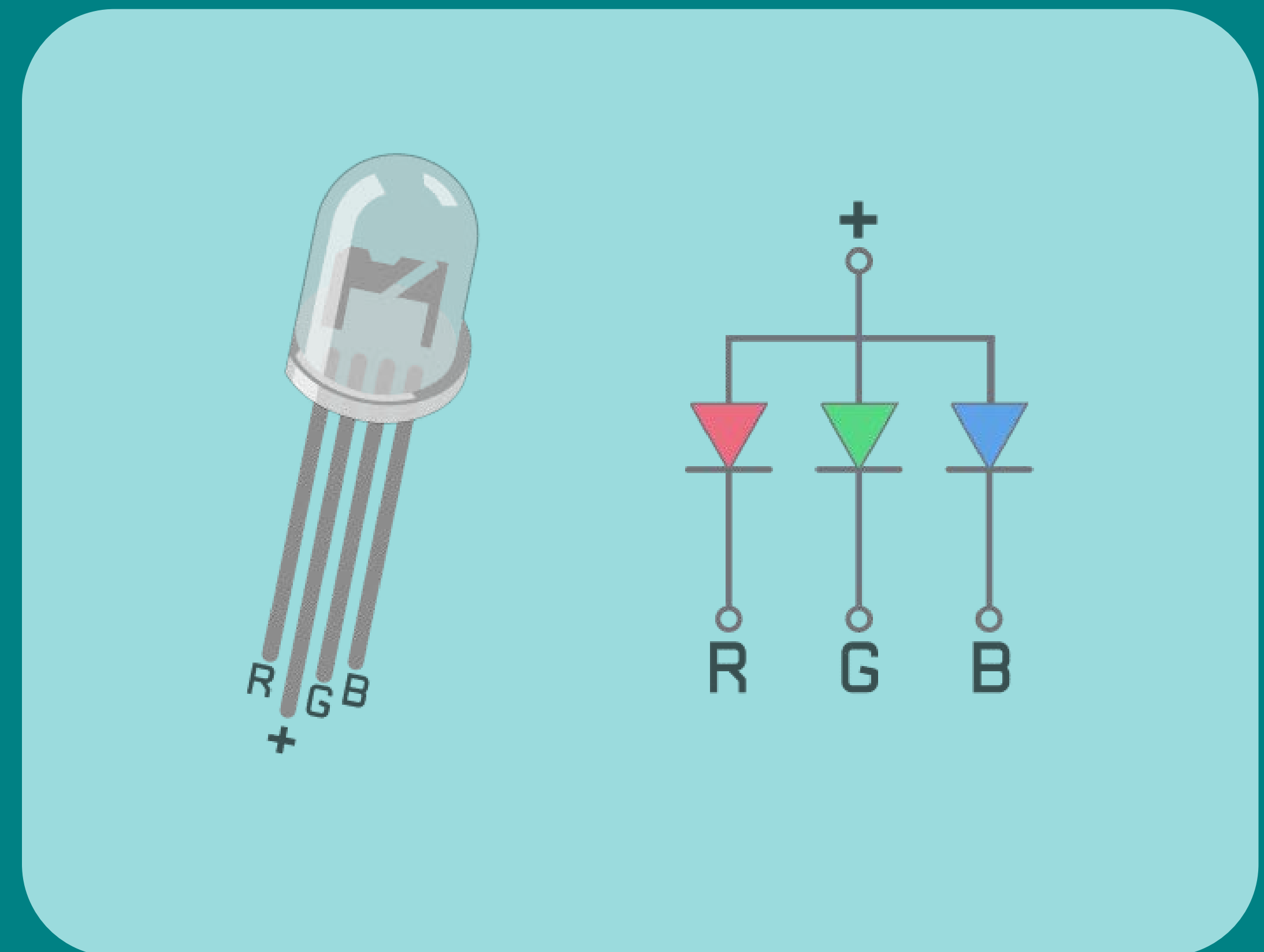


Figura: Desenho elétrico do LED RGB

Interação com LED RGB

Como usar o LED RGB?

É necessário utilizar os pinos PWM do Arduino, esses pinos são marcados com um (~) na placa.

PWM (Pulse Width Modulation)

Modulação por largura de pulso, ou PWM, é uma técnica para obter resultados analógicos via portas digitais. Quando repetido esse padrão rápido o suficiente com um LED, por exemplo, o resultado é como se o sinal fosse uma tensão constante entre 0 e Vcc controlando o brilho do LED.

Interação com LED RGB

analogWrite(pino, valor)

A função **analogWrite()** aciona uma onda PWM em um pino. Ela recebe como parâmetros o *pino* e o *valor*. Esse valor vai de 0 (sempre desligado) a 255 (sempre ligado).

Essa função só atua nos pinos que suportam PWM.

```
void loop(){  
    analogWrite(pinoLed, 0)  
    analogWrite(pinoLed, 128);  
    analogWrite(pinoLed, 255)  
}
```

Figura: Exemplo analogWrite()

∞+ Interação com LED RGB

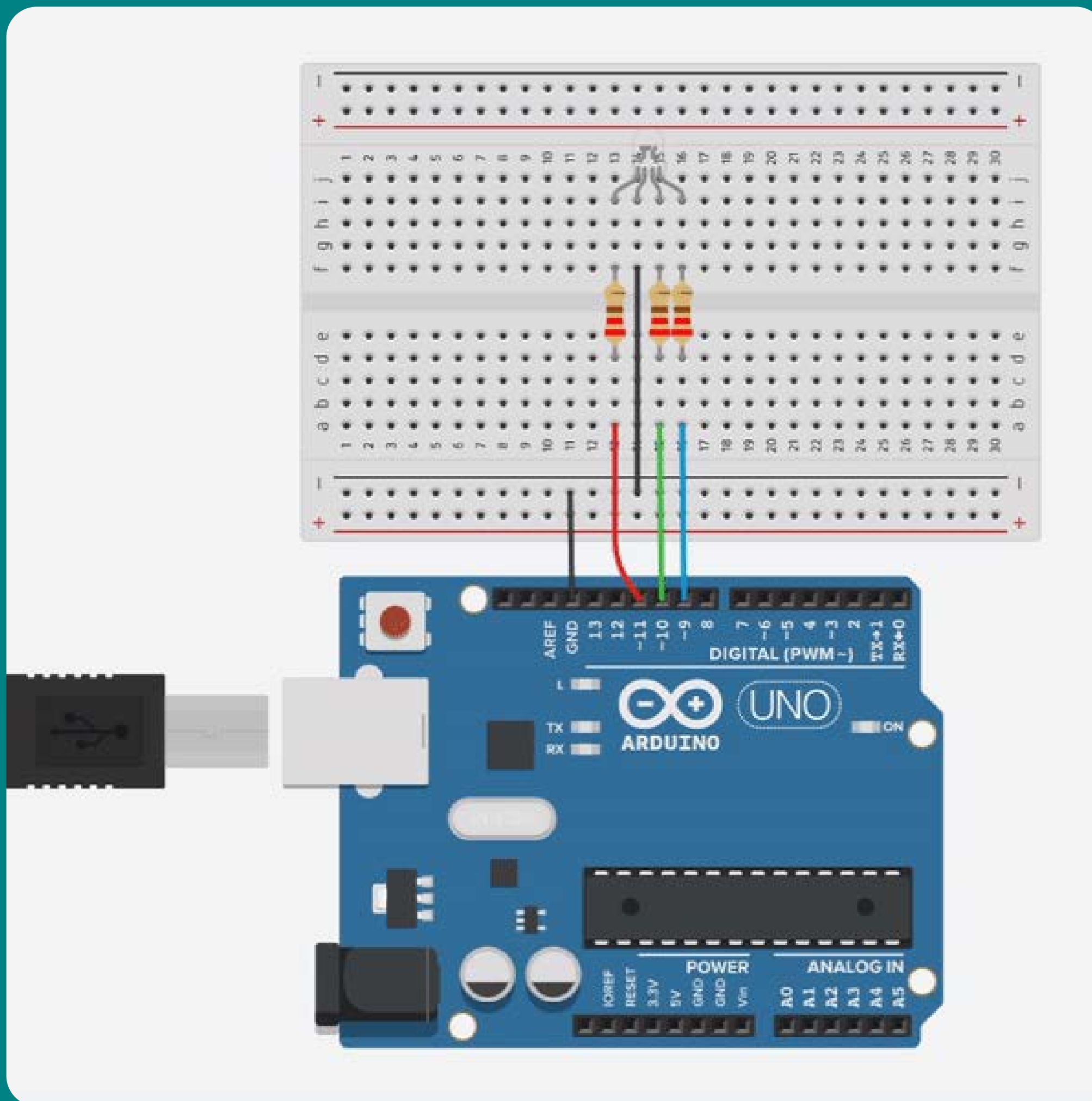


Figura: Circuito LED RGB



```
int pinoR = 11;
int pinoG = 10;
int pinoB = 9;

void setup(){
  pinMode(pinoR, OUTPUT);
  pinMode(pinoG, OUTPUT);
  pinMode(pinoB, OUTPUT);
}

void loop(){
  analogWrite(pinoR, 255);
  analogWrite(pinoG, 0);
  analogWrite(pinoB, 0);
  delay(1500);

  analogWrite(pinoR, 0);
  analogWrite(pinoG, 255);
  analogWrite(pinoB, 0);
  delay(1500);

  analogWrite(pinoR, 0);
  analogWrite(pinoG, 0);
  analogWrite(pinoB, 255);
  delay(1500);
}
```

Figura: Código LED RGB

Interação com LED RGB

Funções

Uma função é um bloco de código que executa alguma operação.

Opcionalmente, uma função pode definir parâmetros de entrada que permitem que os chamadores passem argumentos para a função. Uma função também pode retornar um valor como saída.



```
void setColor(int red, int green, int blue){  
  
    analogWrite(pinoR, red);  
    analogWrite(pinoG, green);  
    analogWrite(pinoB, blue);  
  
}
```

Figura: Exemplo de uma função

∞+ Interação com LED RGB

```
int pinoR = 11;
int pinoG = 10;
int pinoB = 9;

void setup(){
  pinMode(pinoR, OUTPUT);
  pinMode(pinoG, OUTPUT);
  pinMode(pinoB, OUTPUT);
}

void loop(){
  analogWrite(pinoR, 255);
  analogWrite(pinoG, 0);
  analogWrite(pinoB, 0);
  delay(1500);

  analogWrite(pinoR, 0);
  analogWrite(pinoG, 255);
  analogWrite(pinoB, 0);
  delay(1500);

  analogWrite(pinoR, 0);
  analogWrite(pinoG, 0);
  analogWrite(pinoB, 255);
  delay(1500);
}
```

Figura: Código LED RGB

```
int pinoR = 11;
int pinoG = 10;
int pinoB = 9;

void setup(){
  pinMode(pinoR, OUTPUT);
  pinMode(pinoG, OUTPUT);
  pinMode(pinoB, OUTPUT);
}

void loop(){
  setColor(255, 0, 0);
  delay(1500);

  setColor(0, 255, 0);
  delay(1500);

  setColor(0, 0, 255);
  delay(1500);
}

void setColor(int red, int green, int blue){
  analogWrite(pinoR, red);
  analogWrite(pinoG, green);
  analogWrite(pinoB, blue);
}
```

Figura: Código LED RGB simplificado

∞+ Interação com LED RGB

Faça uma
sequência de 5
cores variadas,
usando a função
setColor().

```
void setColor(int red, int green, int blue){  
  
    analogWrite(pinoR, red);  
    analogWrite(pinoG, green);  
    analogWrite(pinoB, blue);  
  
}
```