



**OFICINA DE ROBÓTICA**  
INVENTAR E RECICLAR PARA EDUCAR  
oficinaderobotica.ufsc.br

# Oficina de Robótica

## Programação Básica em Arduino – Aula 3

Execução:

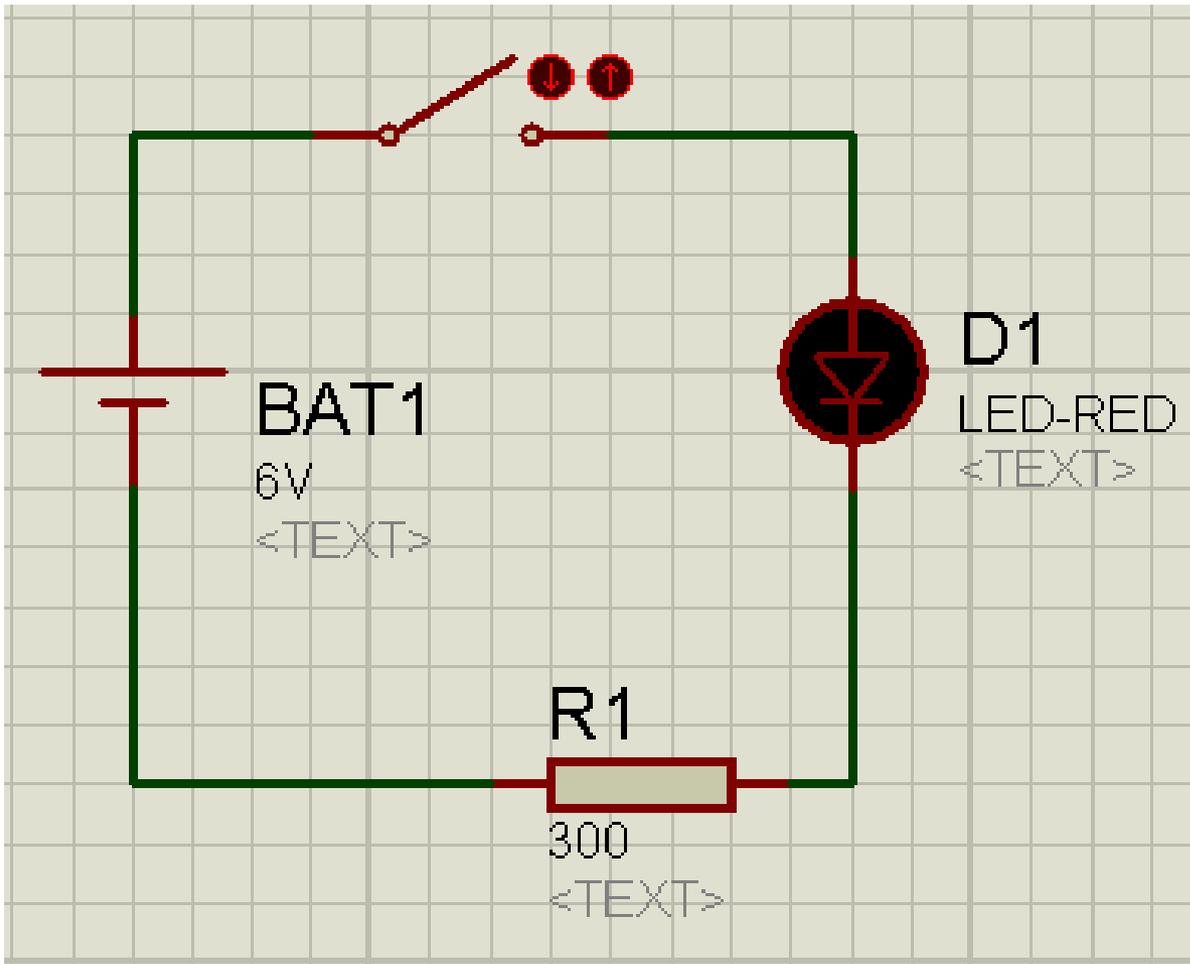


# Revisão

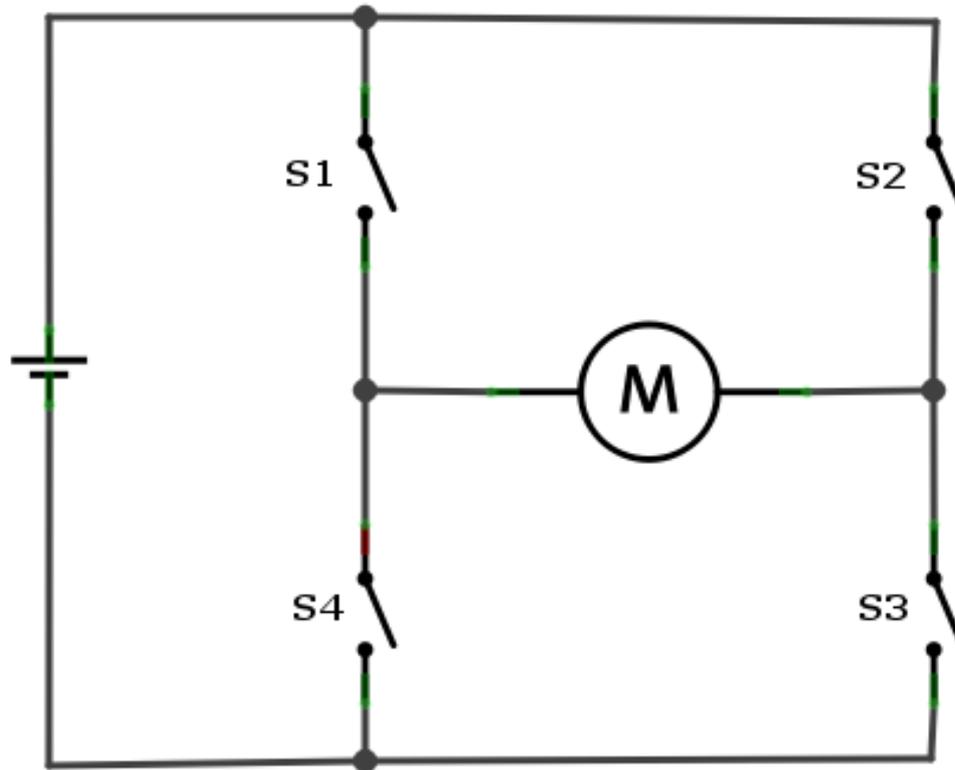


**LARM**

# Botão



# Ponte H



# Variáveis

▶ `int led = 13;`

led

13

# Condição – if e else

```
int botao = 12;
int led = 13;

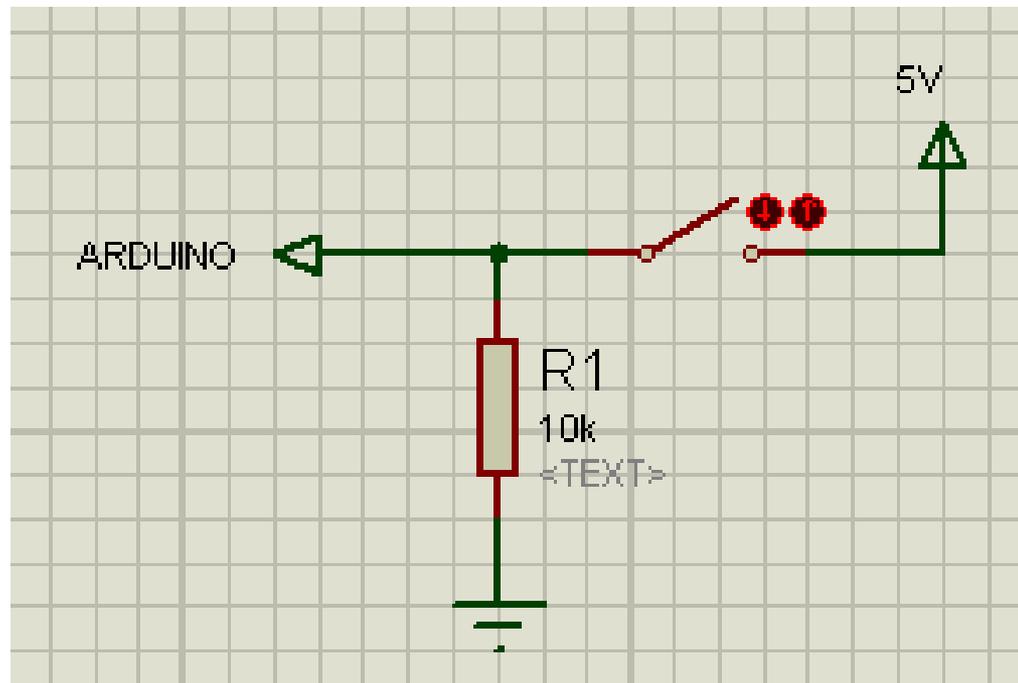
void setup()
{
  pinMode(led, OUTPUT);
  pinMode(botao, INPUT);
}

void loop()
{
  if(digitalRead(botao) == 1)
  {
    digitalWrite(led, HIGH);
    delay(200);
  }
  else
  {
    digitalWrite(led, LOW);
    delay(200);
  }
}
```

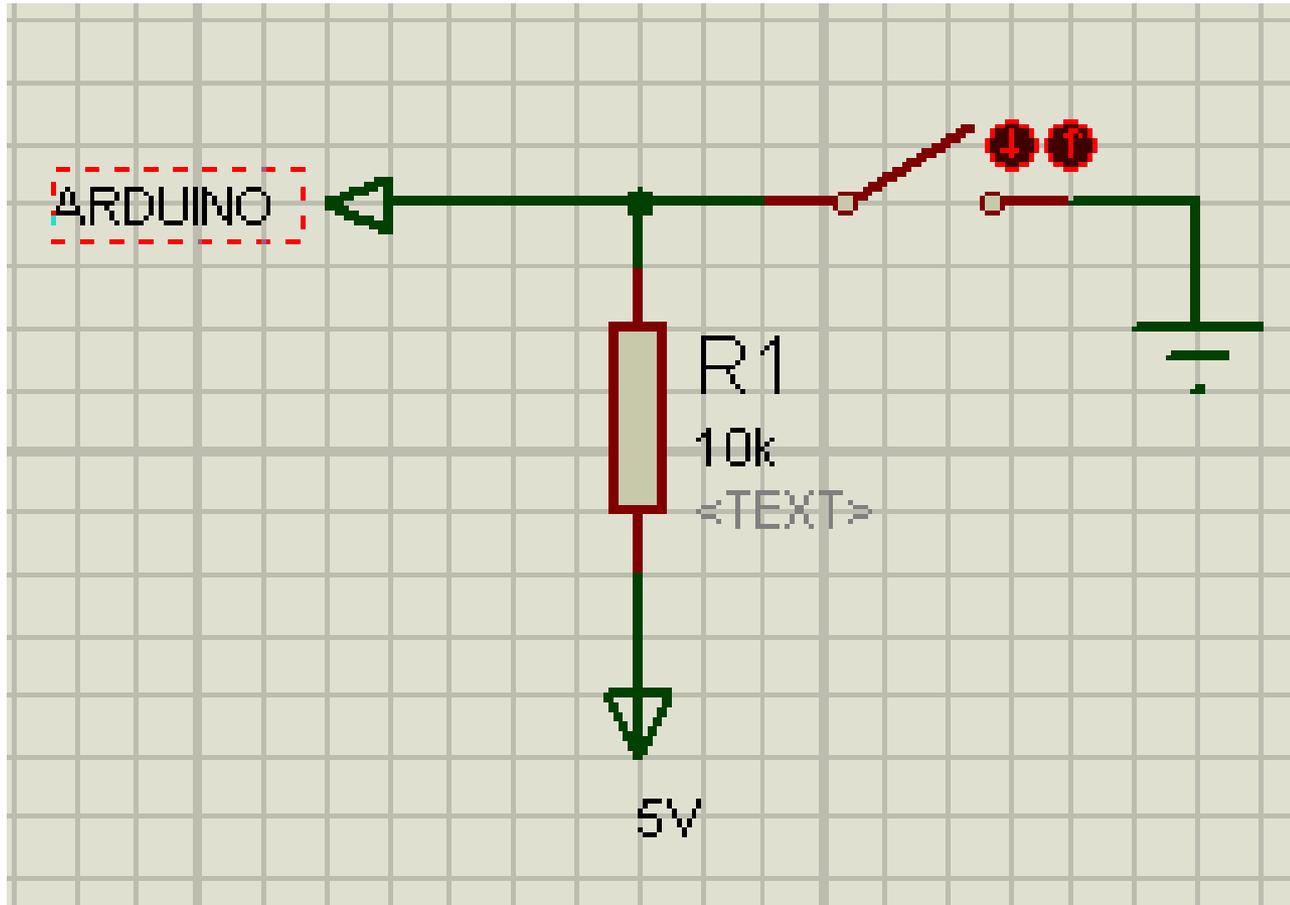


**LARM**

# Esquemático Pulldown

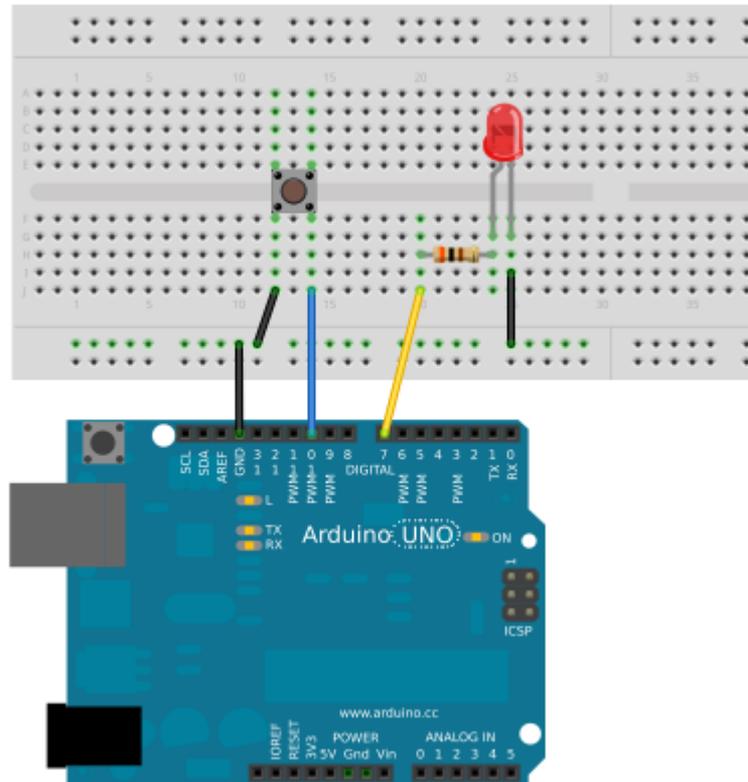


# Esquemático PULLUP



# Entrada Digital de Dados

- ▶ Ativando o *pull-up* de uma porta digital
  - Ligação na protoboard



# Código PULLUP

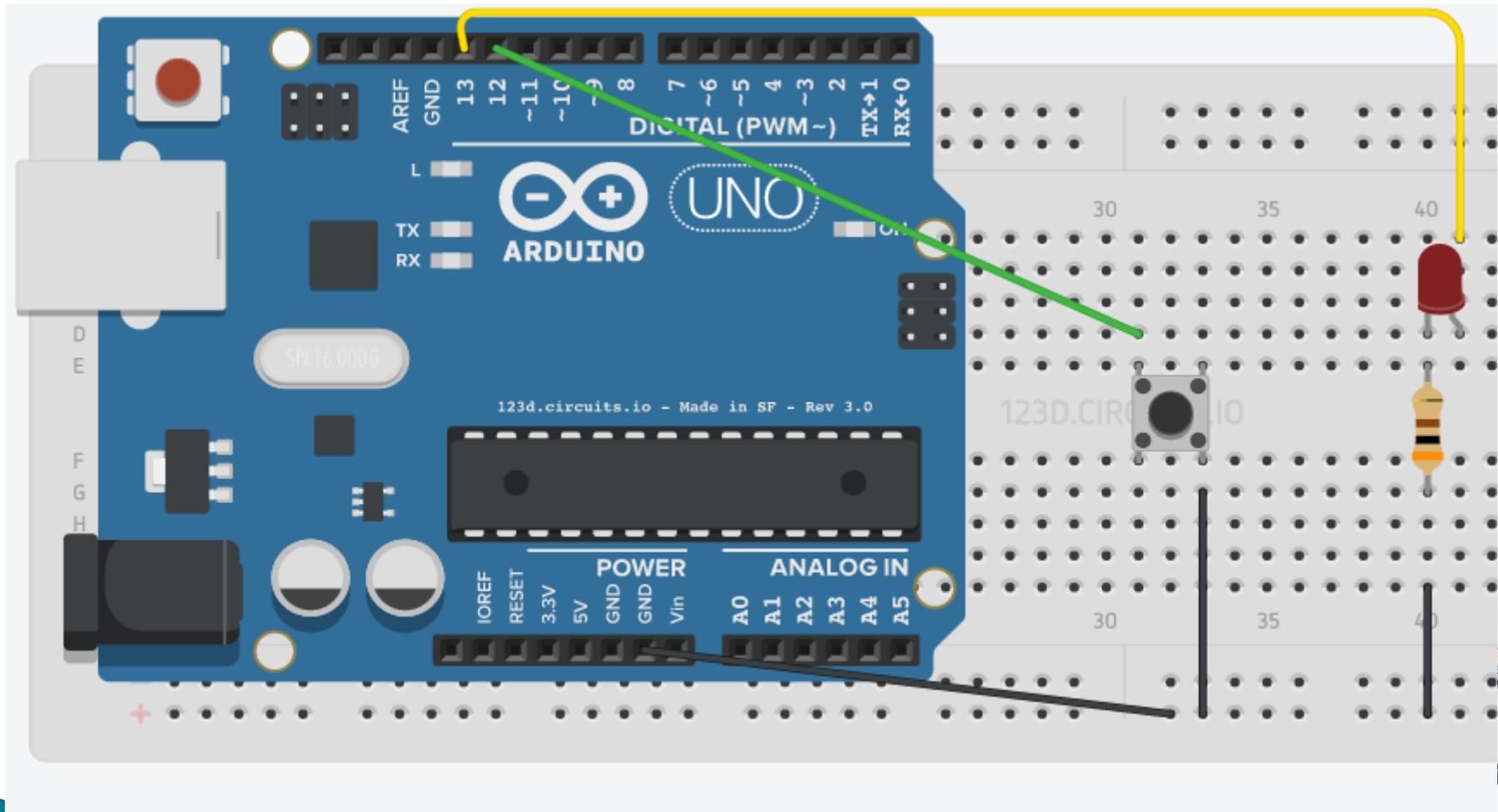
```
int botao = 12;
int led = 13;
|
void setup()
{
  pinMode(led, OUTPUT);
  pinMode(botao, INPUT_PULLUP);
}

void loop()
{
  if(digitalRead(botao) == 0)
  {
    digitalWrite(led, HIGH);
    delay(200);
  }
  else
  {
    digitalWrite(led, LOW);
    delay(200);
  }
}
```



LARM

# Exemplo PULLUP



# Exercícios

- ▶ Semáforo para pedestres. Um semaforo comum que está sempre no verde. Ele terá mais 2 LEDs para o pedestre e um botão. Quando um pedestre pressionar o botão, o semáforo deve esperar 500 ms e depois deve passar do amarelo para o vermelho. O semáforo de pedestre deve ficar verde, contar 2 segundos e depois voltar a vermelho. Quando ficar vermelho, o semáforo da estrada deve apagar o vermelho e acender o verde.

# Exercícios

- ▶ Desenvolver um jogo para perguntas e respostas:
- ▶ 2 botões – 1 para cada jogador
- ▶ 2 LEDs (vermelho e verde) – 1 pra cada jogador
- ▶ Os jogadores devem pressionar o botão quando permitido;
- ▶ O jogador que pressionar primeiro deve ter seu LED acesso por 5 segundos e o LED do adversário não pode acender junto.

# Tabela Verdade

- ▶ A tabela verdade nos permite verificar a funcionalidade do código;
- ▶ Com ela é possível testar o estado das variáveis e definir se o código está funcionando corretamente;

# Exemplo

```
void setup() {
    int led = 13;
    int botao = 10;
    pinMode(13, OUTPUT);
    pinMode(12, OUTPUT);
    pinMode(11, OUTPUT);

    pinMode(botao, INPUT_PULLUP);
}

void loop() {

    digitalWrite(led, HIGH);
    delay(500);
    digitalWrite(led, LOW);
    delay(500);

    if(digitalRead(botao)==0)
    {
        led = led - 1;
        delay(250);
    }
    if(led < 11)
    {
        led = 13;
    }
}
```

# Tabela Verdade

led	Botao
13	1
12	0
12	1
11	0
11	11
10	0
13	1

# Exercício

- ▶ Do código que será mostrado a seguir:
- ▶ Diga quantos ciclos o loop irá fazer para que todos os 3 leds estejam acessos.
- ▶ Diga qual o valor final das variáveis quando isso ocorrer

# Exercício

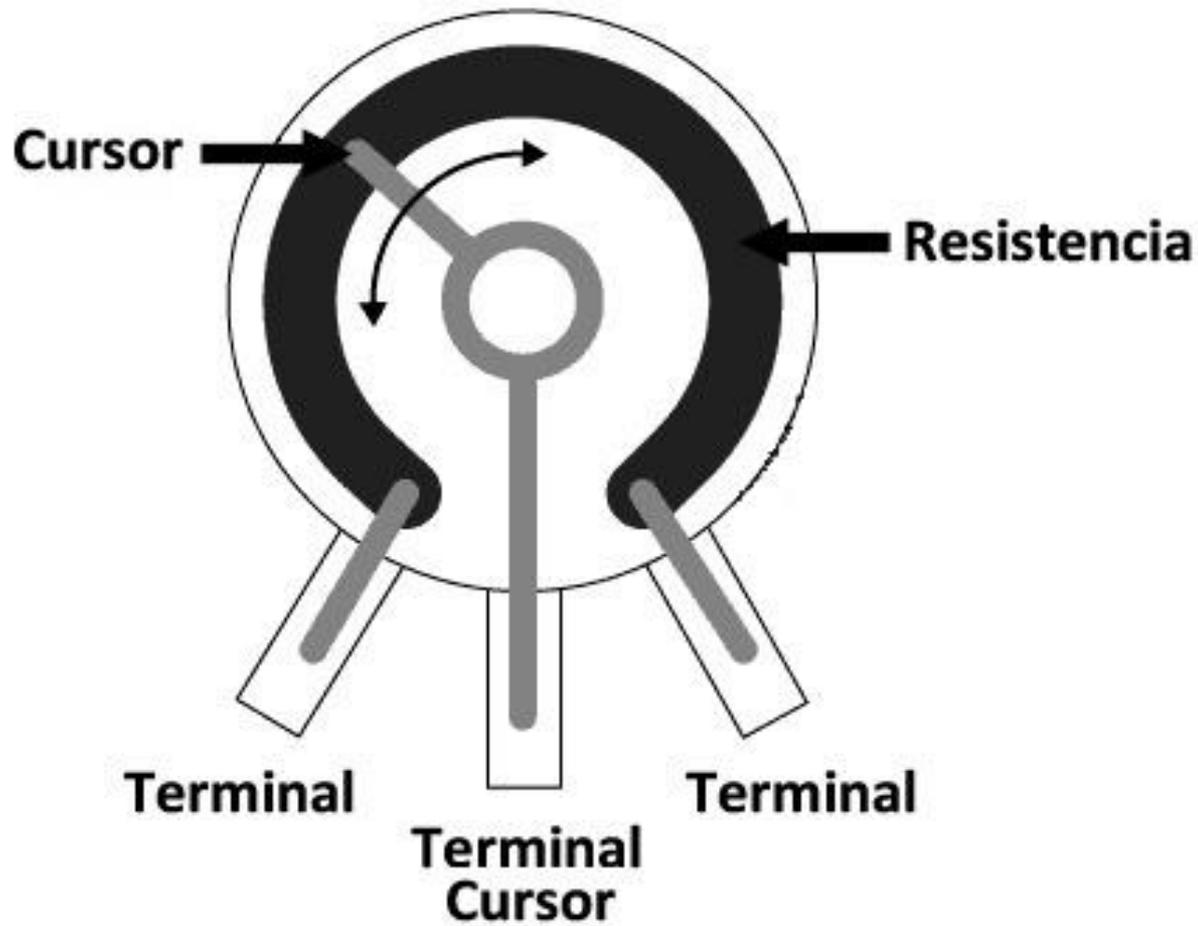
```
void setup() {  
    int led = 13;  
    int contador = 0;  
  
    pinMode (13, OUTPUT);  
    pinMode (12, OUTPUT);  
    pinMode (11, OUTPUT);  
  
}  
  
void loop() {  
    contador = contador + 1;  
    if(contador == 2)  
    {  
        led = led - 1;  
        contador = 0;  
    }  
    digitalWrite (led, HIGH);  
}
```

# Potenciômetro

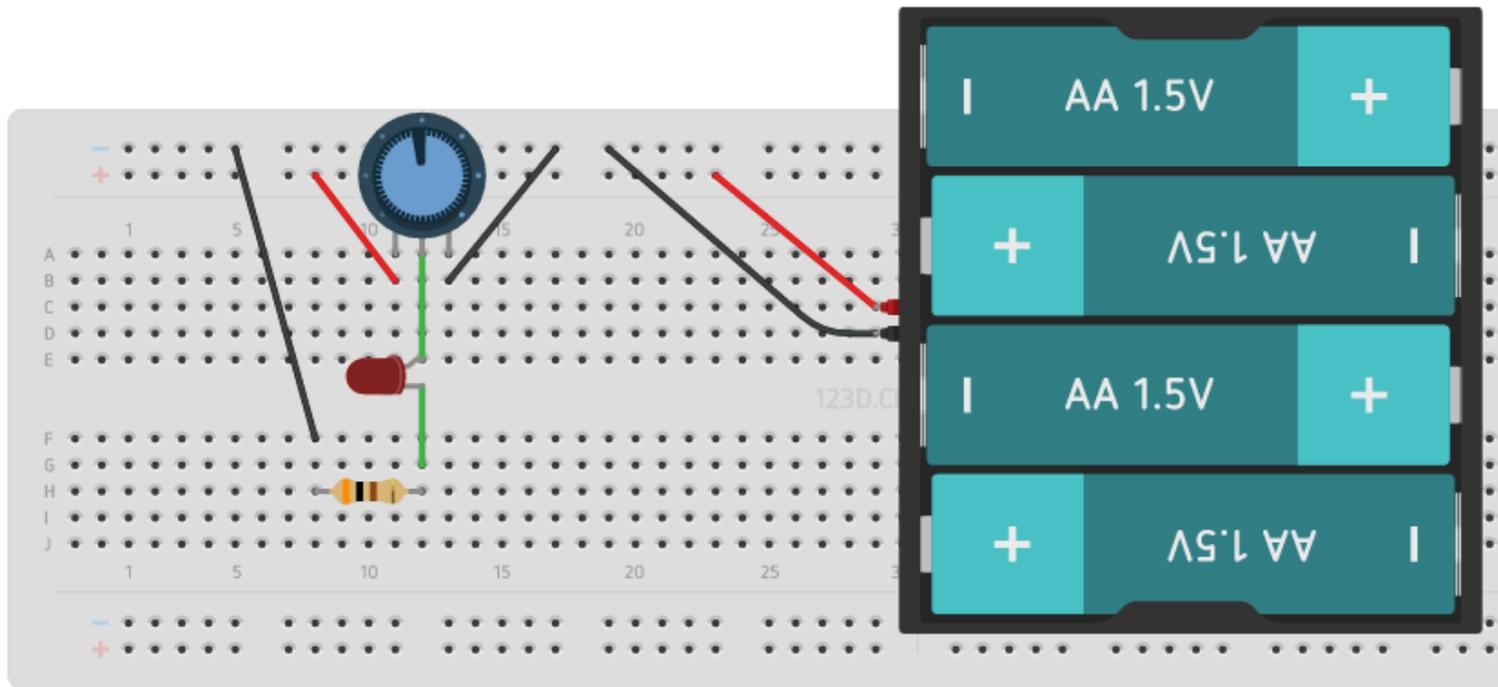
- ▶ Resistor variável



# Potenciômetro



# Potenciômetro



# Buzzer



# Buzzer

