

Minicurso

Introdução à Plataforma

Arduino

MINISTRANTES:

ALCIMAR FRANCELINO DE MEDEIROS

NATHECIA DA CUNHA SANTOS



Quem somos?



- Alcimar Medeiros, tem 26 anos, graduando em licenciatura em informática pelo IFRN. Amante das tecnologias, Arduino, Raspberry Pi e jogador 2048 nas horas vagas.



- Nathecia Cunha, tem 18 anos, estudante de Informática no IFRN, Tec. Redes de Computadores pelo IMD e entusiasta na área maker, apaixonada por Arduino, Tecnologias e Unicórnios.

Sumário

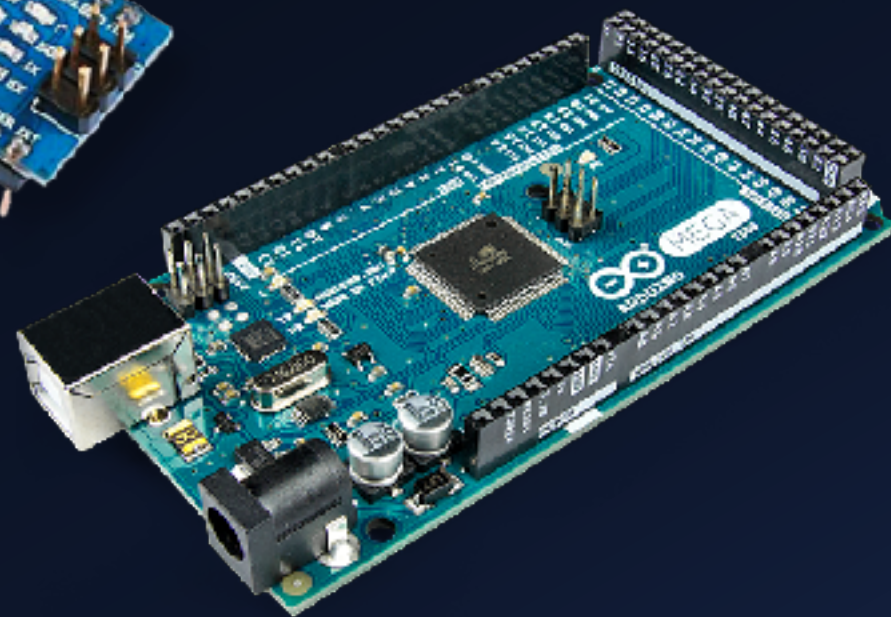
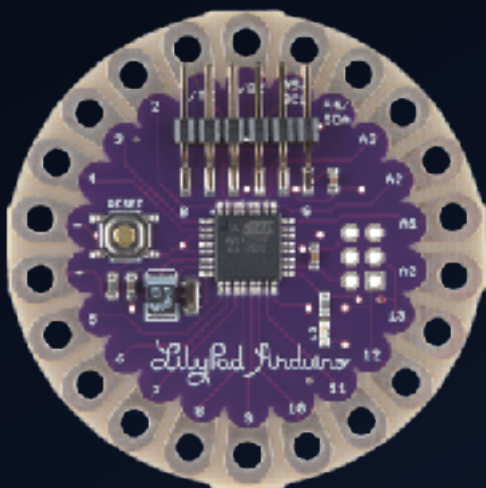
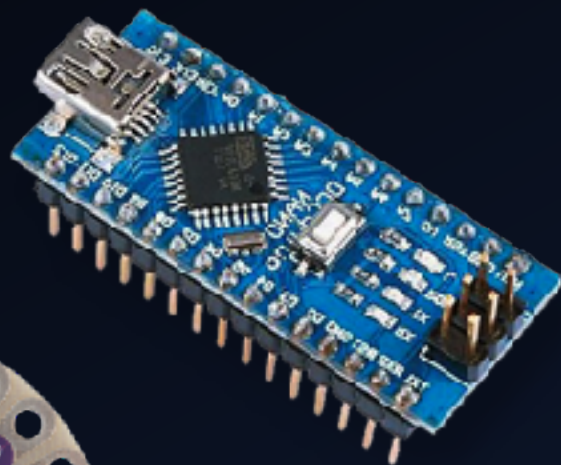
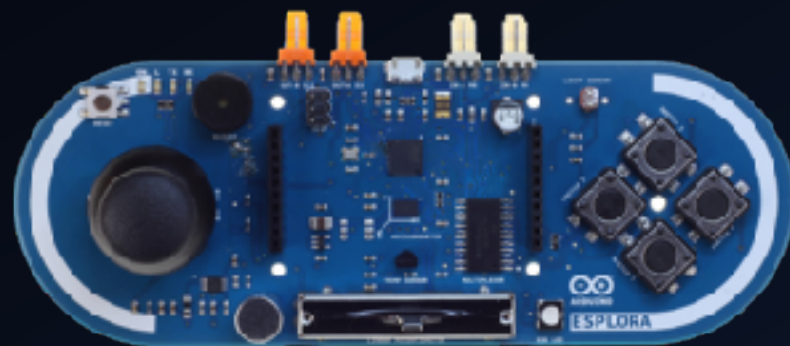
- Arduino
 - Introdução
 - Funcionamento
- Eletrônica Básica
 - Corrente
 - Tensão
 - Resistência
 - Circuito
- Ambiente de Desenvolvimento
 - Programação
- Exercícios práticos



A Final de Contas o Que é o Arduino?

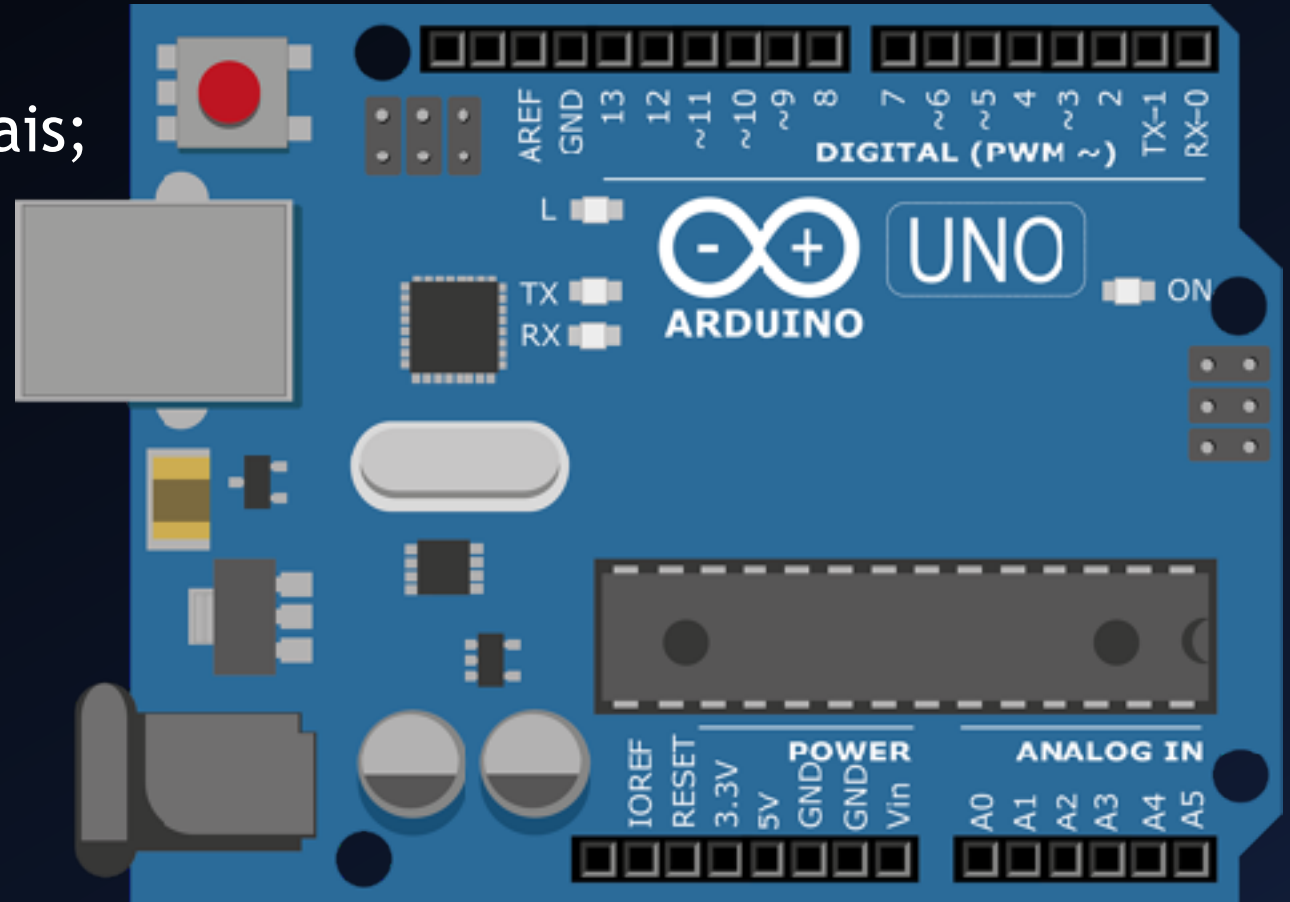
- Plataforma Open-Source de prototipagem eletrônica com hardware e software flexíveis e fáceis de usar.
- Dividido em dois componentes:
 - A Placa -> Hardware, Construir Projetos
 - IDE Arduino -> Software, Código.



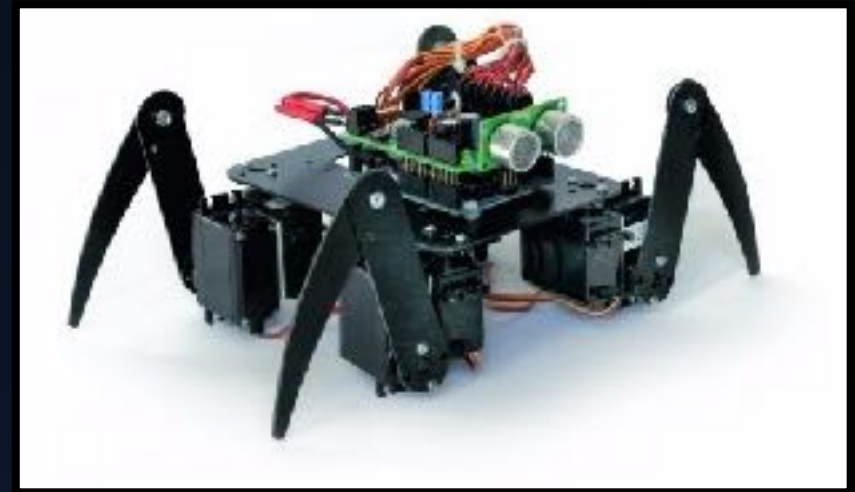
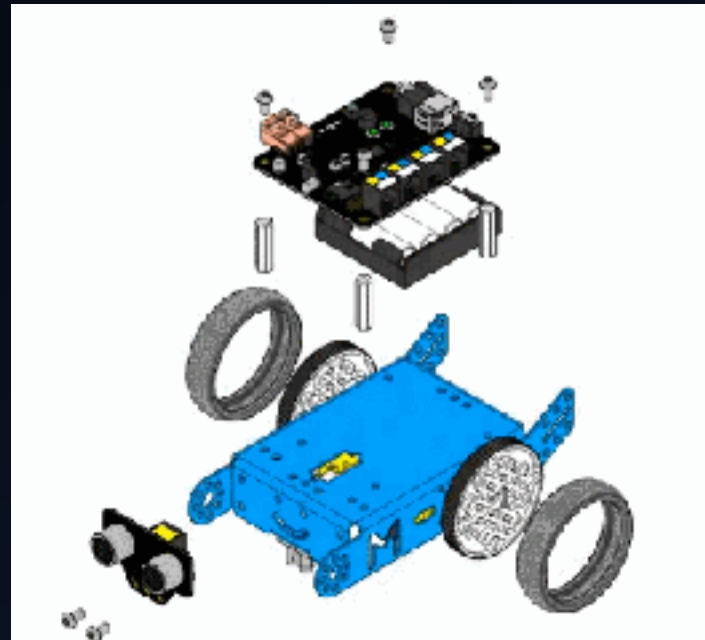
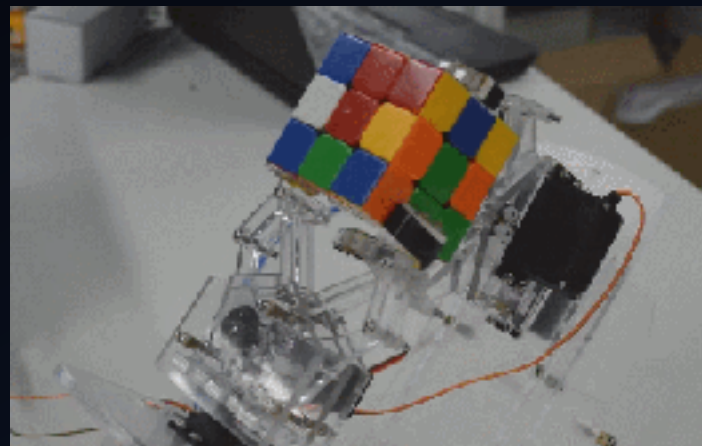


Funcionamento:

- Portas:
 - 14 Entradas e Saídas Digitais;
 - 6 Entradas Analógicas;
- Alimentação:
 - VCC 5 Volts;
 - GND Terra;
- Botão Reset
- 32KB Para Programas;
- Comunicação USB;



PROJETOS



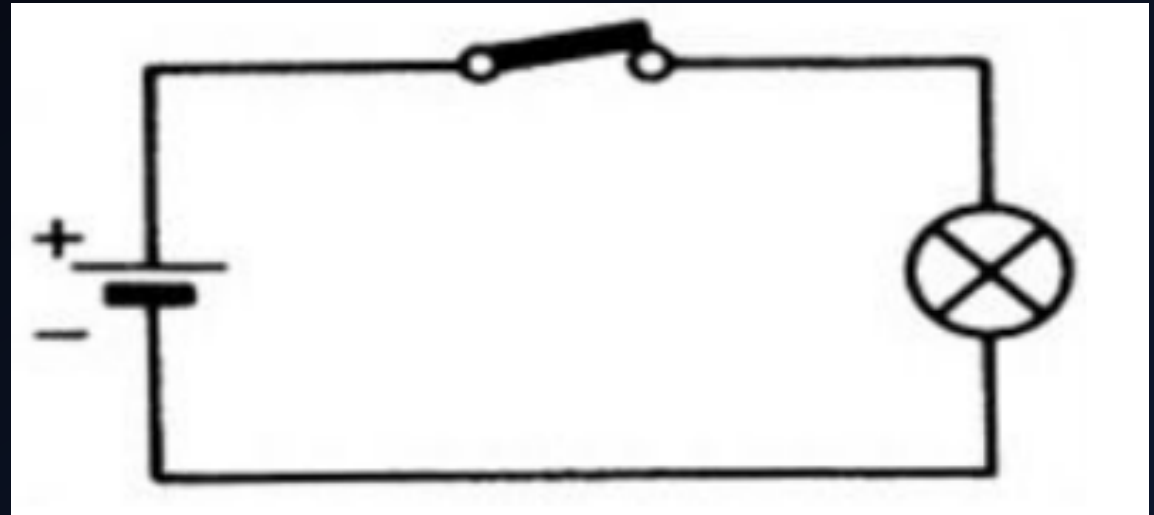
Eletrônica Básica:

- Corrente (amp)
- Tensão (volt)
- Resistência (ohm)

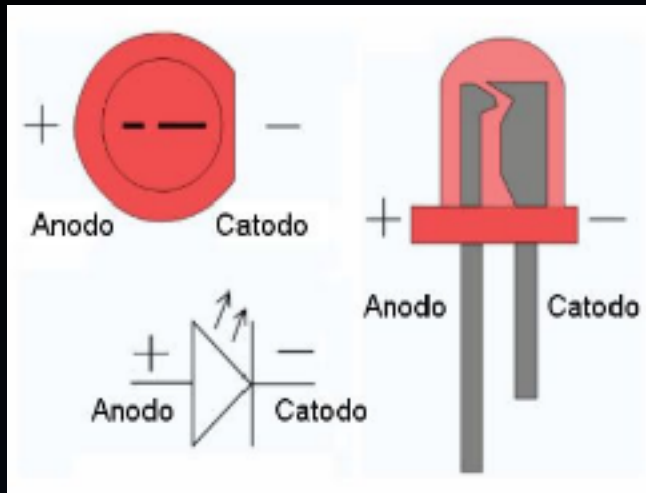


Eletrônica Básica:

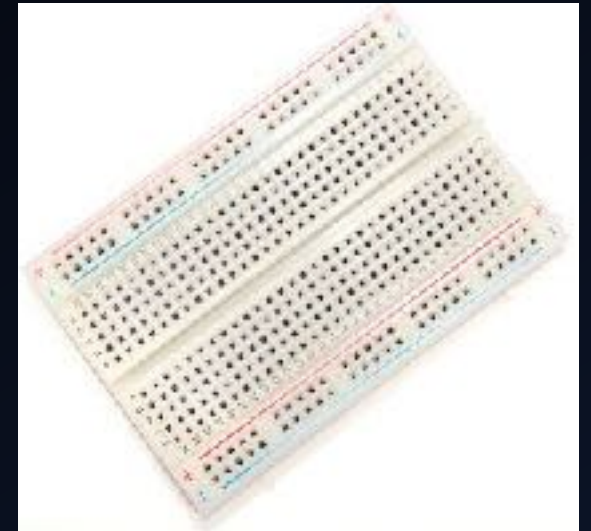
- Circuito Aberto;
- Circuito fechado;



Componentes:



LED - Light Emitting Diode:
Emite luz quando se passa
corrente.



Protoboard:

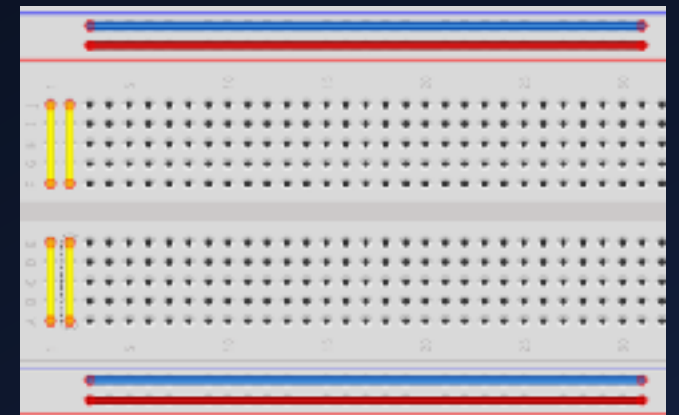
Uma placa de tamanho variado, com
diversos furos interligados para
prototipagem rápida. Os furos podem
conectar componentes em série ou
paralelo.



Resistor:
Limita a passagem de corrente pelo circuito,
impedindo um curto.



Fios/Jumpers:
Conectam dois pontos no
circuito

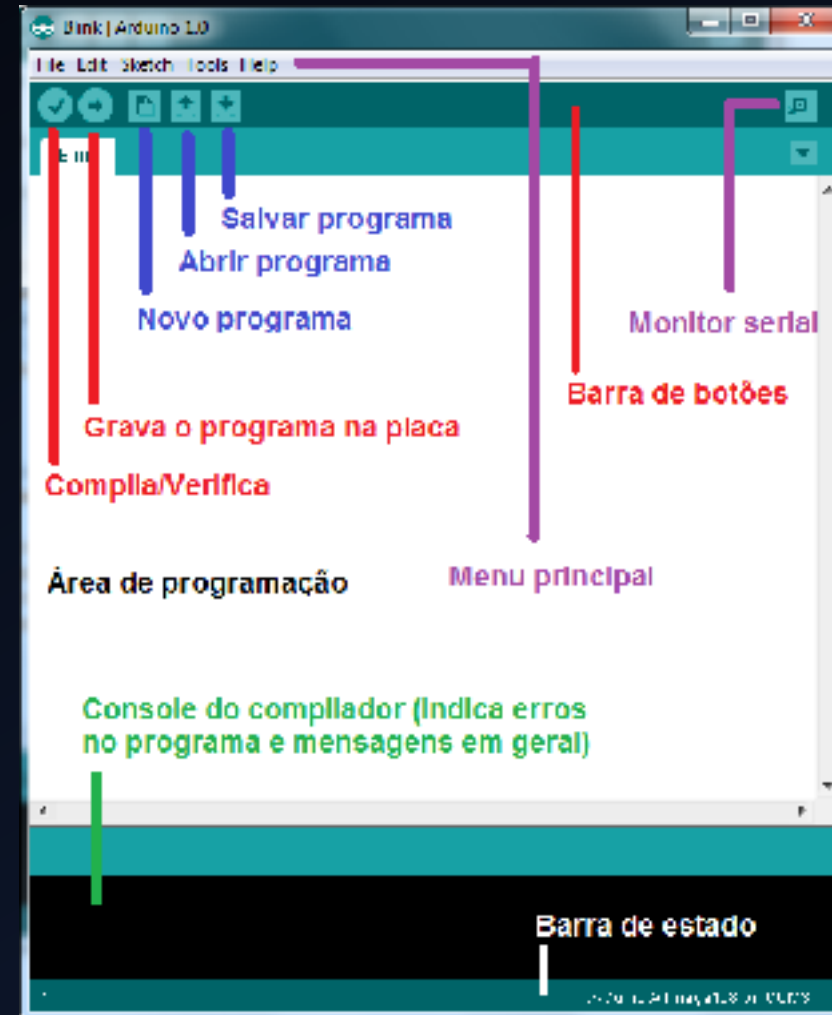


IDE - INTEGRATED DEVELOPMENT
ENVIRONMENT



Ambiente de Desenvolvimento:

- Ambiente de programação para arduino.
- INTERFACE:
 - Verificar
 - Carregar
 - Novo
 - Abrir
 - Salvar
- FUNÇÕES:
 - Void Setup;
 - Void Loop;



Comandos:

tipos:

int
boolean
char
unsigned int
string
void
float

Comparadores

&&
||
==
!=
!
<
>
<=
>=

Funções:

Serial.begin(9600)
pinMode(porta, definição)
digitalWrite(porta, valor)
digitalRead(porta)
analogWrite(porta, valor)
analogRead(porta)
delay(milisegundos)
Serial.println("texto")

Estruturas Condicionais / loops

if()
else()
for()

Operadores

+
-
.
/

Não tenho Arduino, como irei praticar?





AUTODESK®
123D® CIRCUITS

<https://circuits.io/>

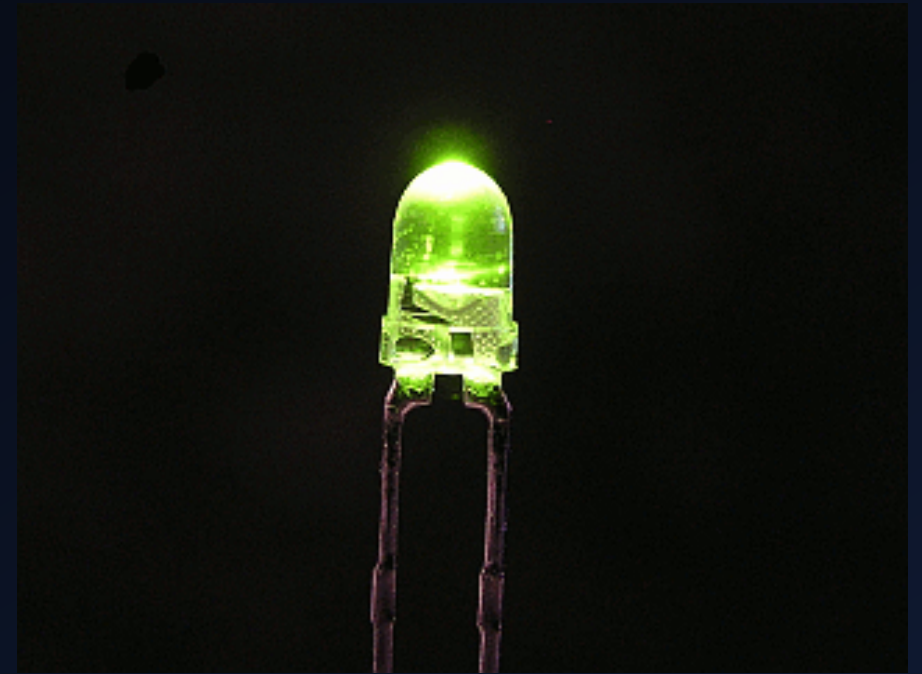


Vamos começar ?

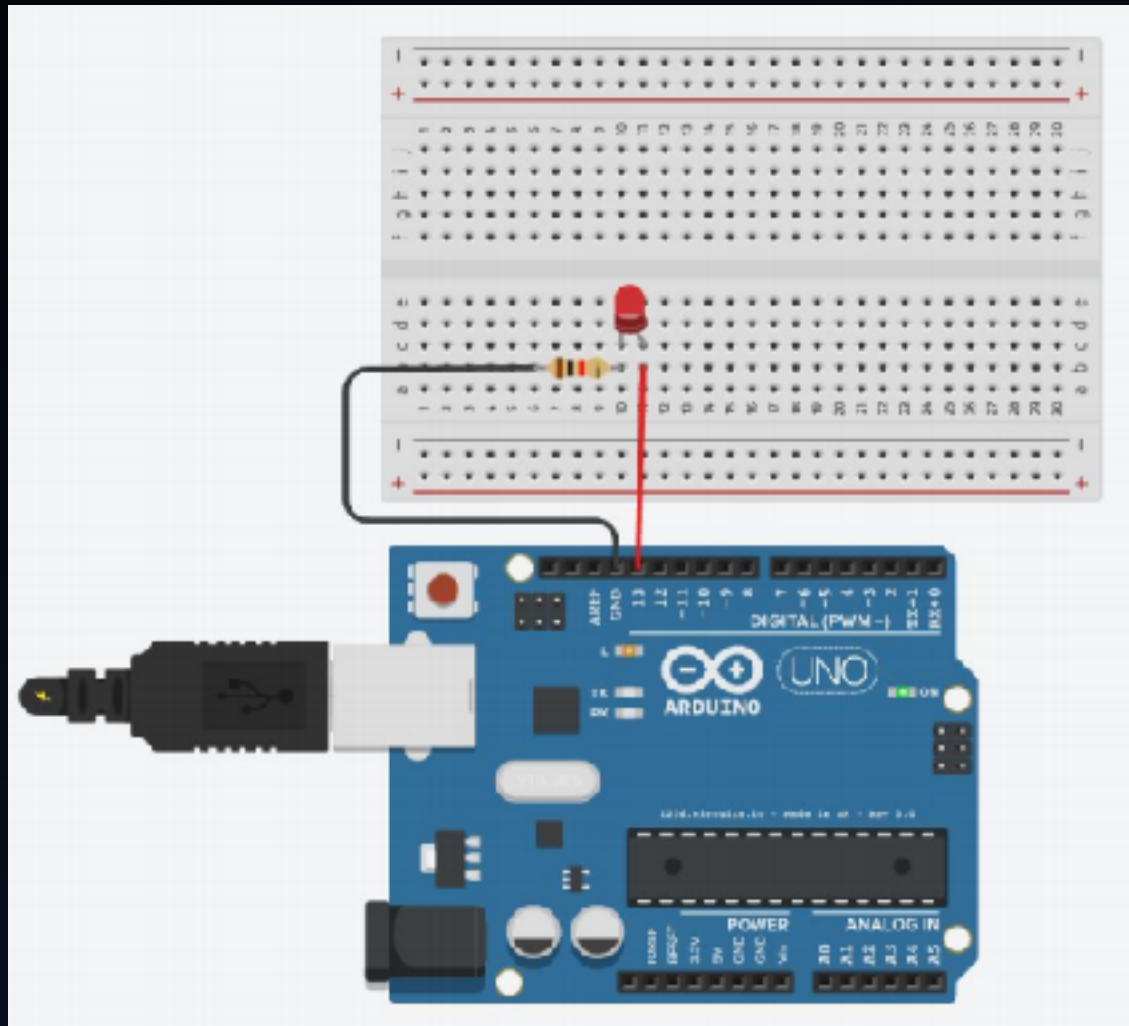


Exercício 1 - BLINK

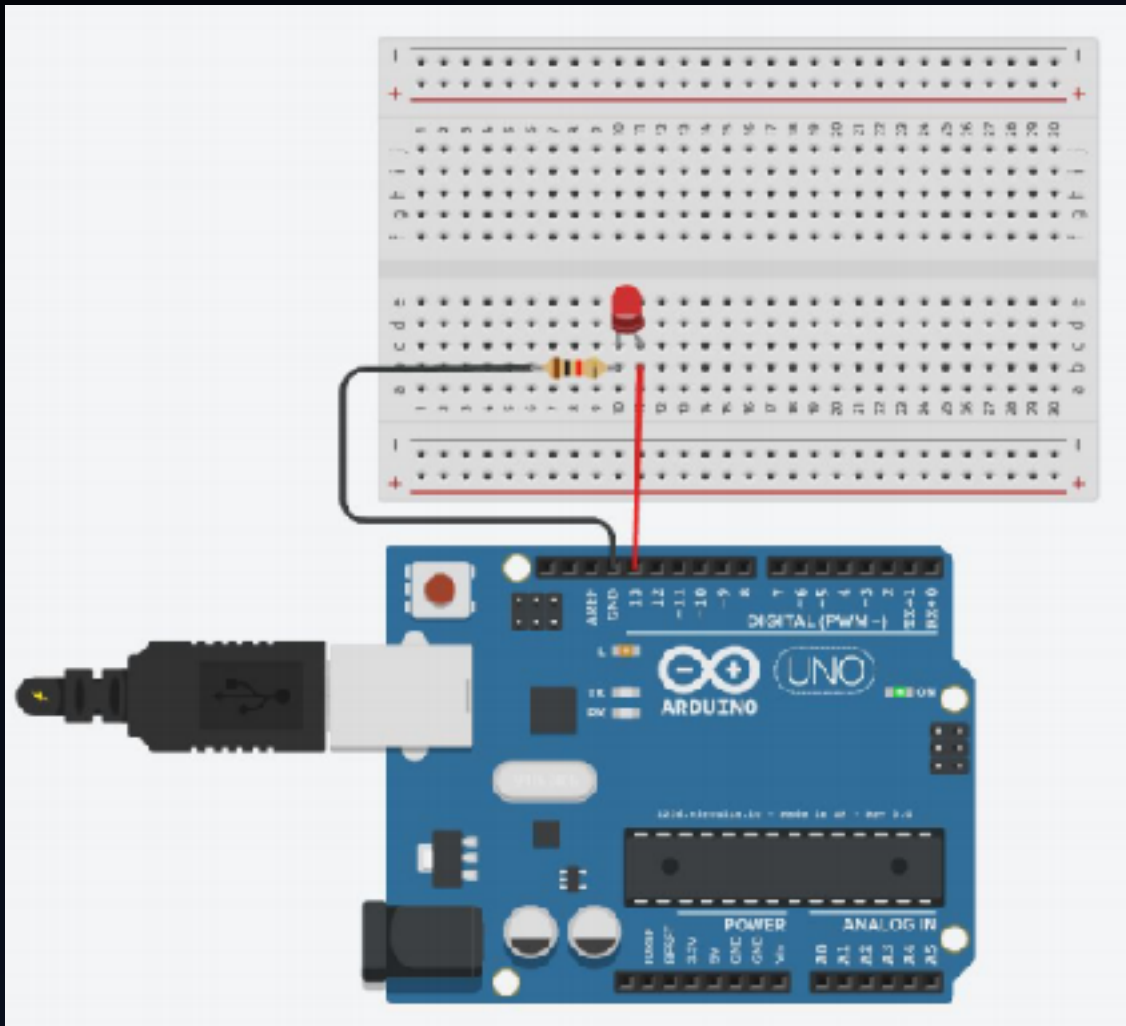
- Objetivo: Fazer o LED piscar através da oscilação de sinal digital.
- Materiais:
 - LED
 - Protoboard
 - Jumpers
 - Resistor 220 ohm
 - Arduino



Exercício 1 - BLINK



Exercício 1 - BLINK

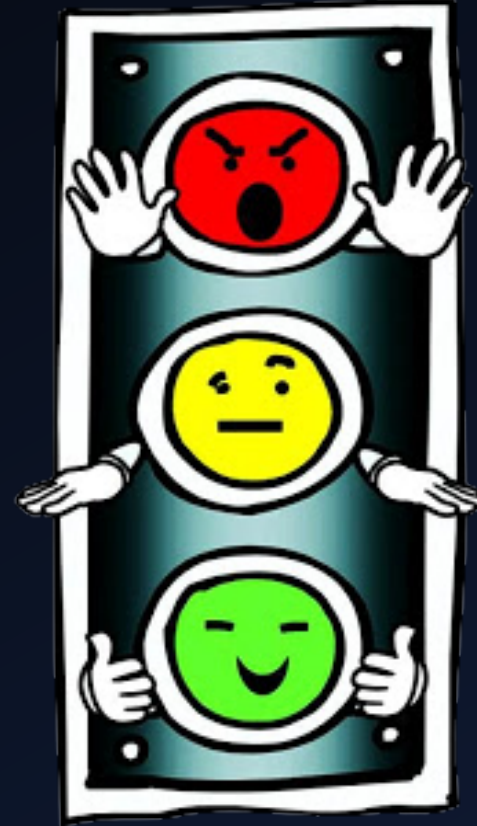


```
void setup()
{
    pinMode(13, OUTPUT);
}
void loop()
{
    digitalWrite(13, HIGH);
    delay(1000);

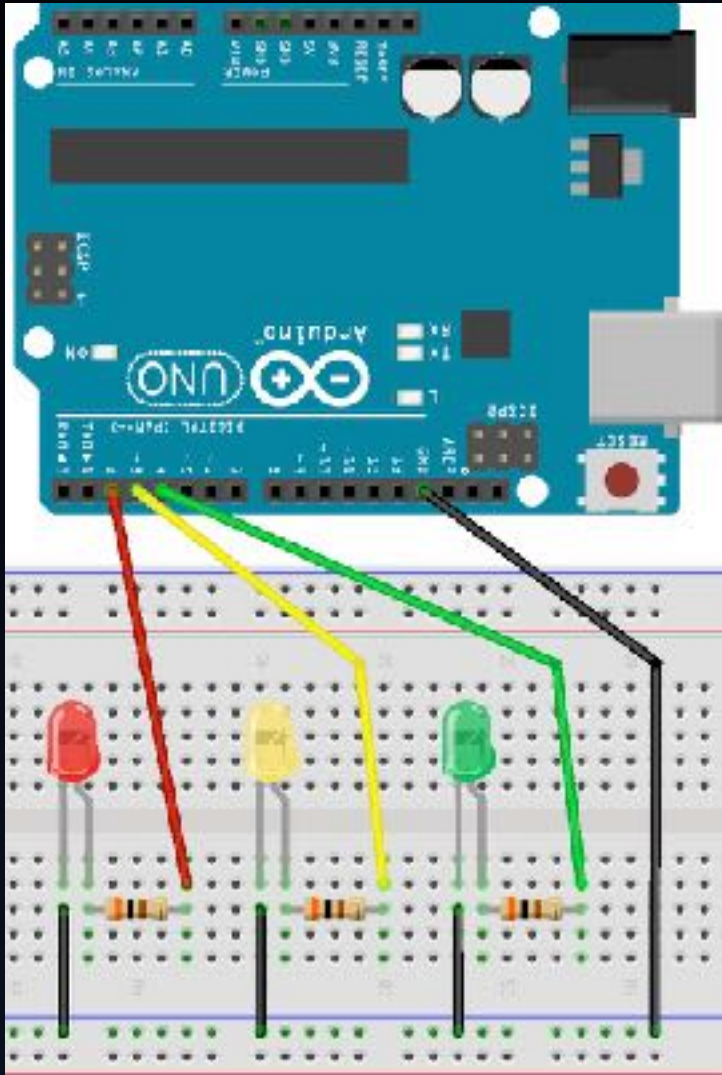
    digitalWrite(13, LOW);
    delay(1000);
}
```

Exercício 2 - Semáforo

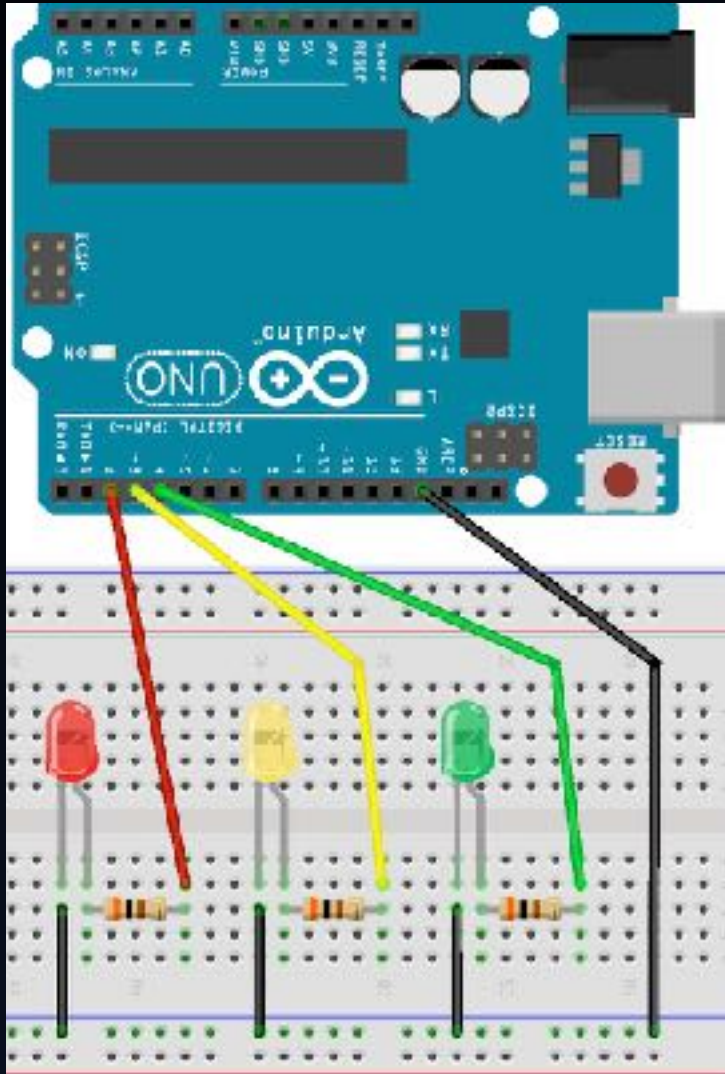
- Objetivo: Replique o código de 1 único LED para 3 LEDs. Monte na sua protoboard as cores dos LEDs conforme um SEMÁFORO de trânsito.
- Materiais:
 - 3 LED's
 - Protoboard
 - Jumpers
 - 3 Resistores 220 ohm
 - Arduino



Exercício 2 - Semáforo



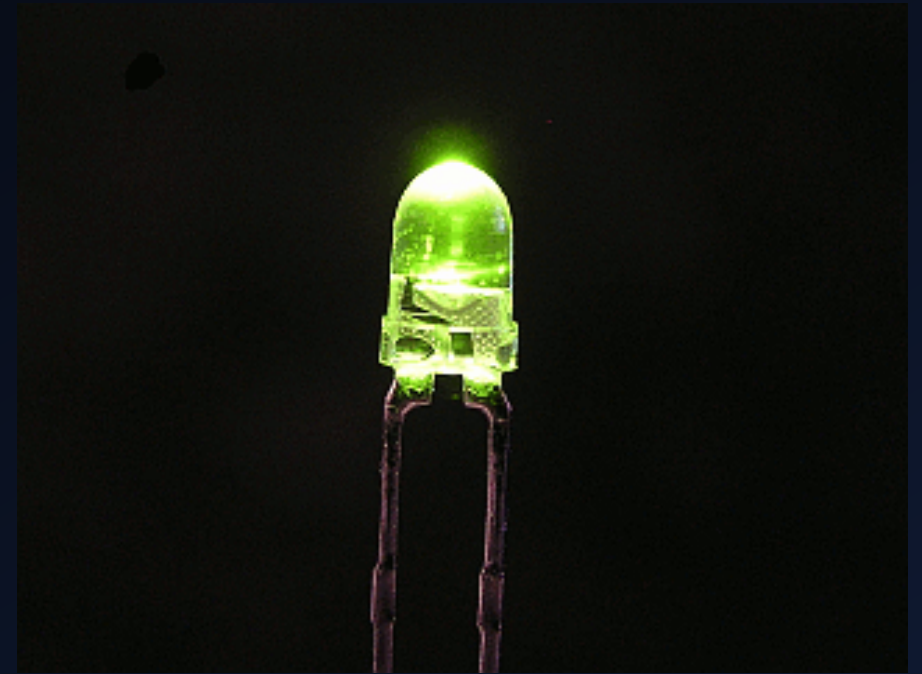
Exercício 2 - Semáforo



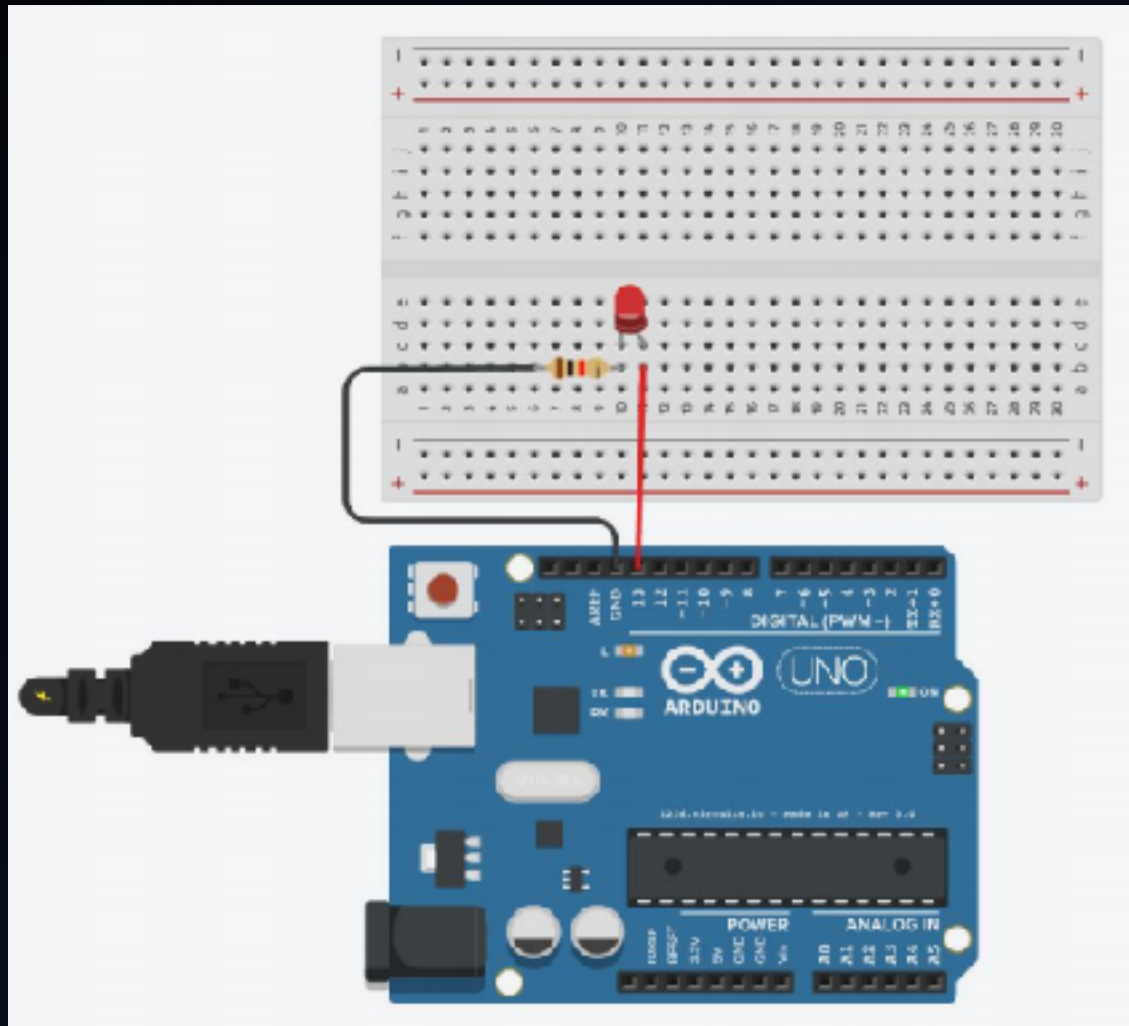
```
void setup() {  
  pinMode (2, OUTPUT); // verde  
  pinMode (3, OUTPUT); // amarelo  
  pinMode (4, OUTPUT); // vermelho  
}  
  
void loop() {  
  digitalWrite (2,HIGH); // verde  
  delay (4000);  
  digitalWrite (2,LOW);  
  digitalWrite (3,HIGH); // amarelo  
  delay (2000);  
  digitalWrite (3,LOW);  
  digitalWrite (4,HIGH); //vermelho  
  delay (4000);  
  digitalWrite (4,LOW);  
  delay (2000);  
}
```


Exercício 3 - FADE

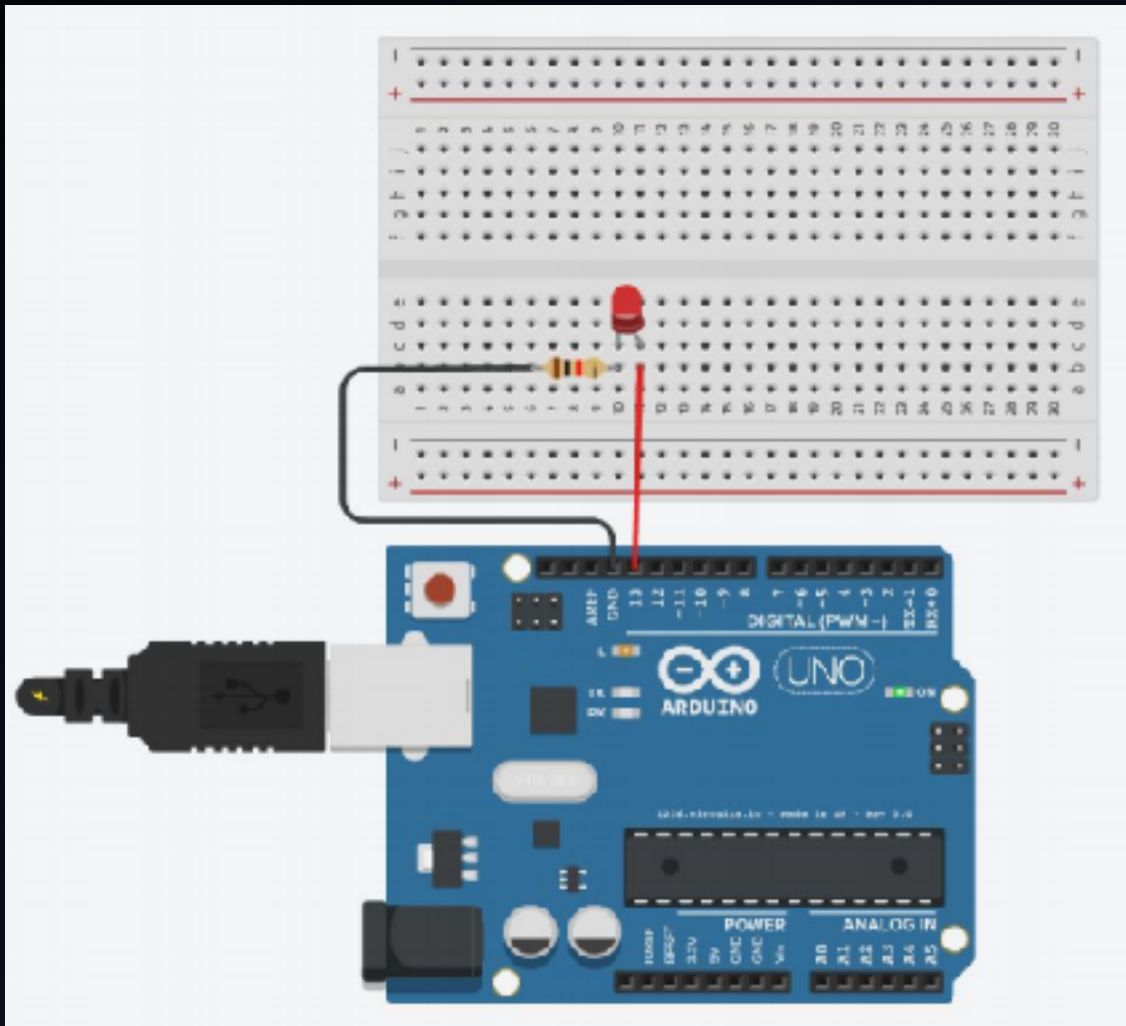
- Objetivo: Fazer o LED piscar através da oscilação de sinal digital.
- Materiais:
 - LED
 - Protoboard
 - Jumpers
 - Resistor 220 ohm
 - Arduino



Exercício 3 - FADE



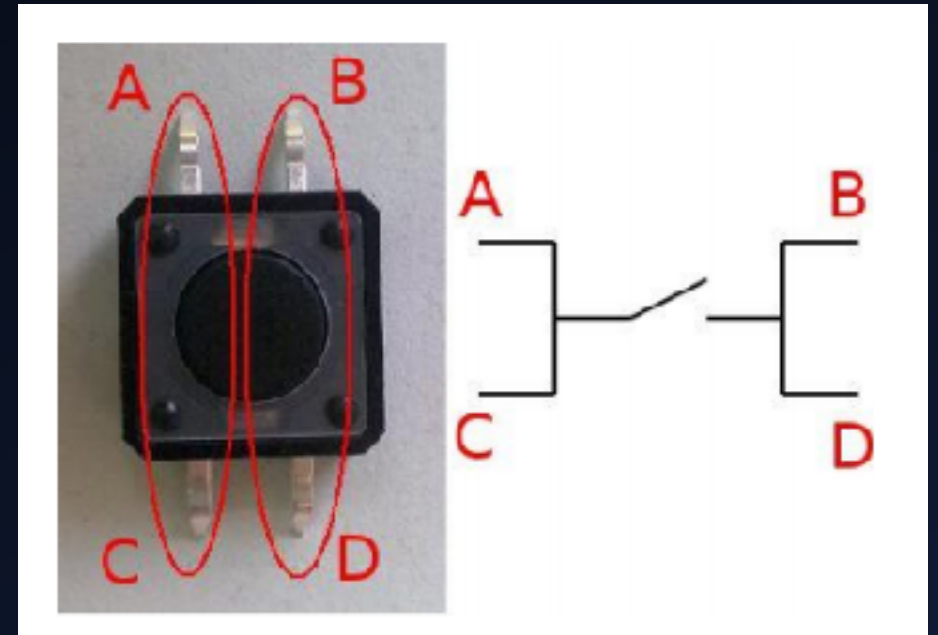
Exercício 3 - FADE



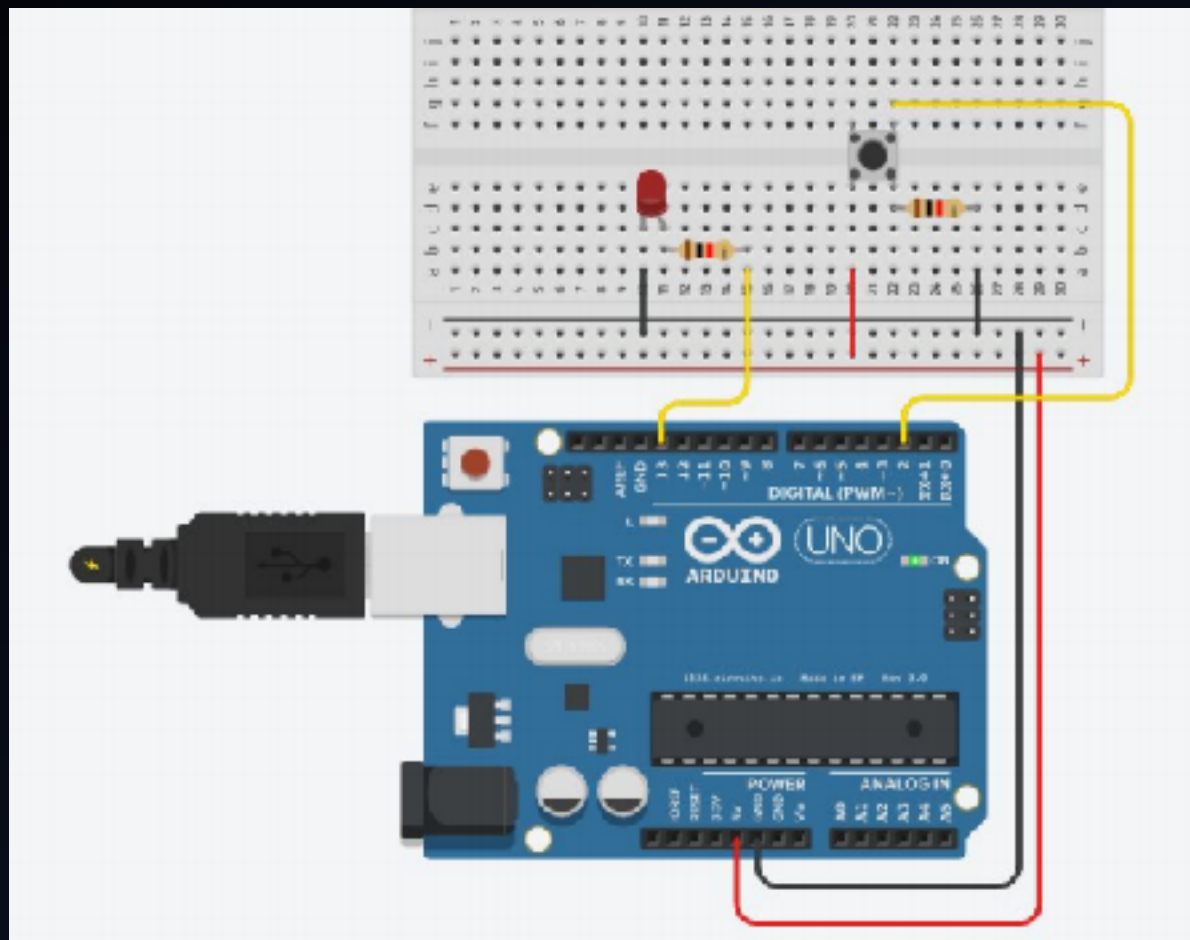
```
1 int led = 13;
2 void setup() {
3 }
4 void loop() {
5     for(int i = 0; i <= 255; i+=5){
6         analogWrite(led, i);
7         delay(30);
8     }
9     for(int i = 255; i >= 0; i-=5){
10        analogWrite(led, i);
11        delay(30);
12    }
13 }
```

Exercício 4 - Botão

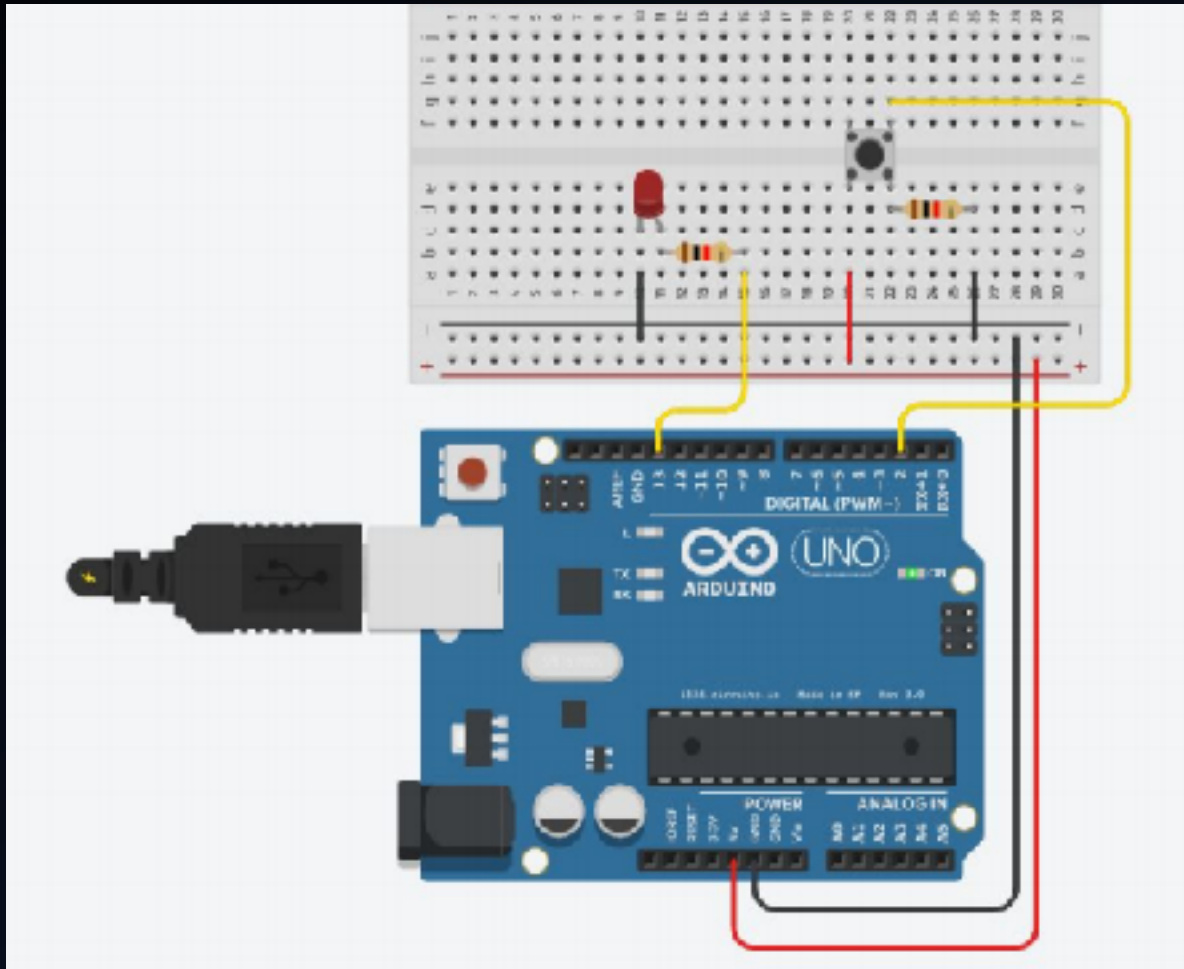
- Objetivo: Ligar/Desligar um LED ao apertar o botão.
- Materiais:
 - Protoboard
 - Jumpers
 - Resistor 220 ohm
 - Arduino
 - Push-button
- Push-button: um simples componente abre-fecha circuito



Exercício 4 - Botão



Exercício 4 - Botão



```
const int ledPin = 13;
const int Botao = 2;
int EstadoBotao = 0;

void setup()
{
    pinMode(ledPin, OUTPUT);
    pinMode(Botao, INPUT);
}

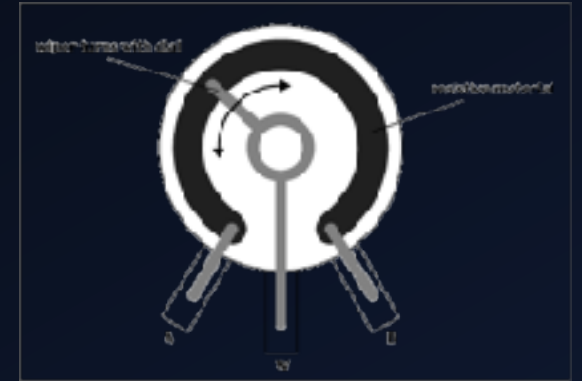
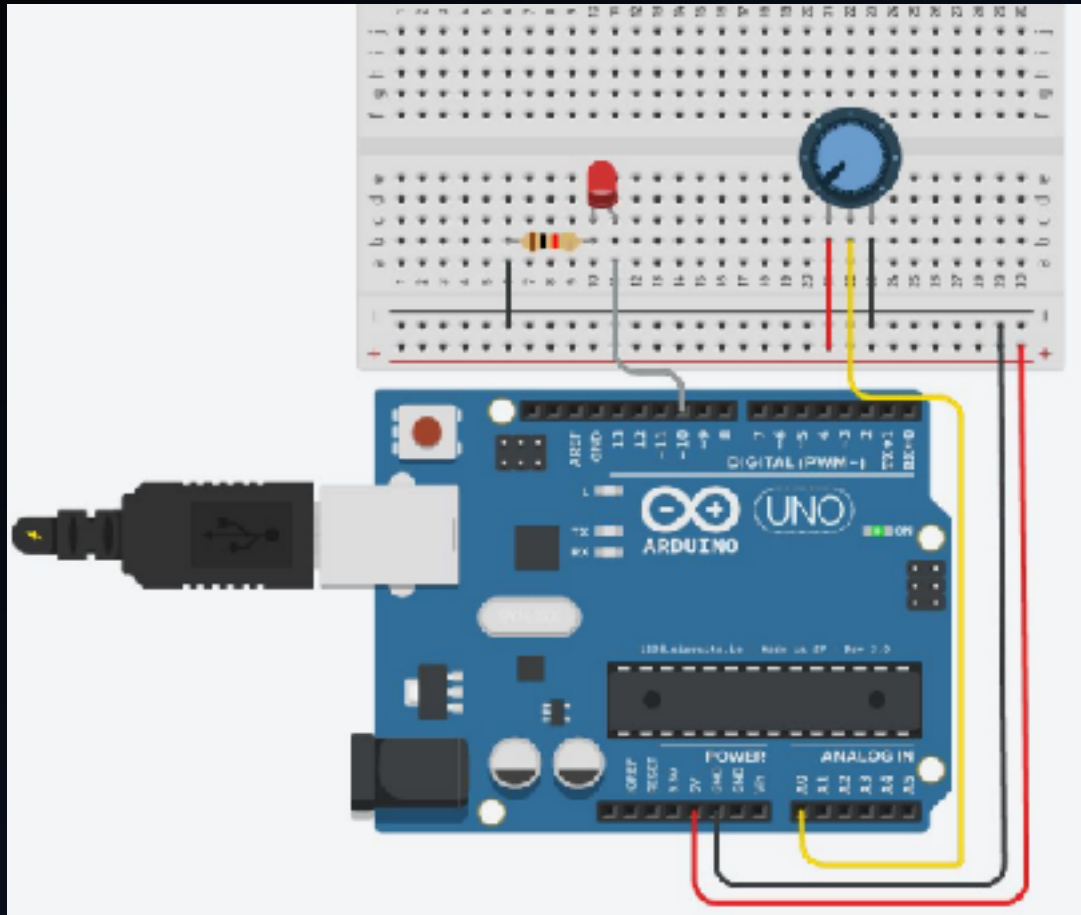
void loop()
{
    EstadoBotao = digitalRead(Botao);
    if (EstadoBotao == HIGH){
        digitalWrite(ledPin, HIGH);
    }
    else{
        digitalWrite(ledPin, LOW);
    }
}
```

Exercício 5 - Intensidade de um LED

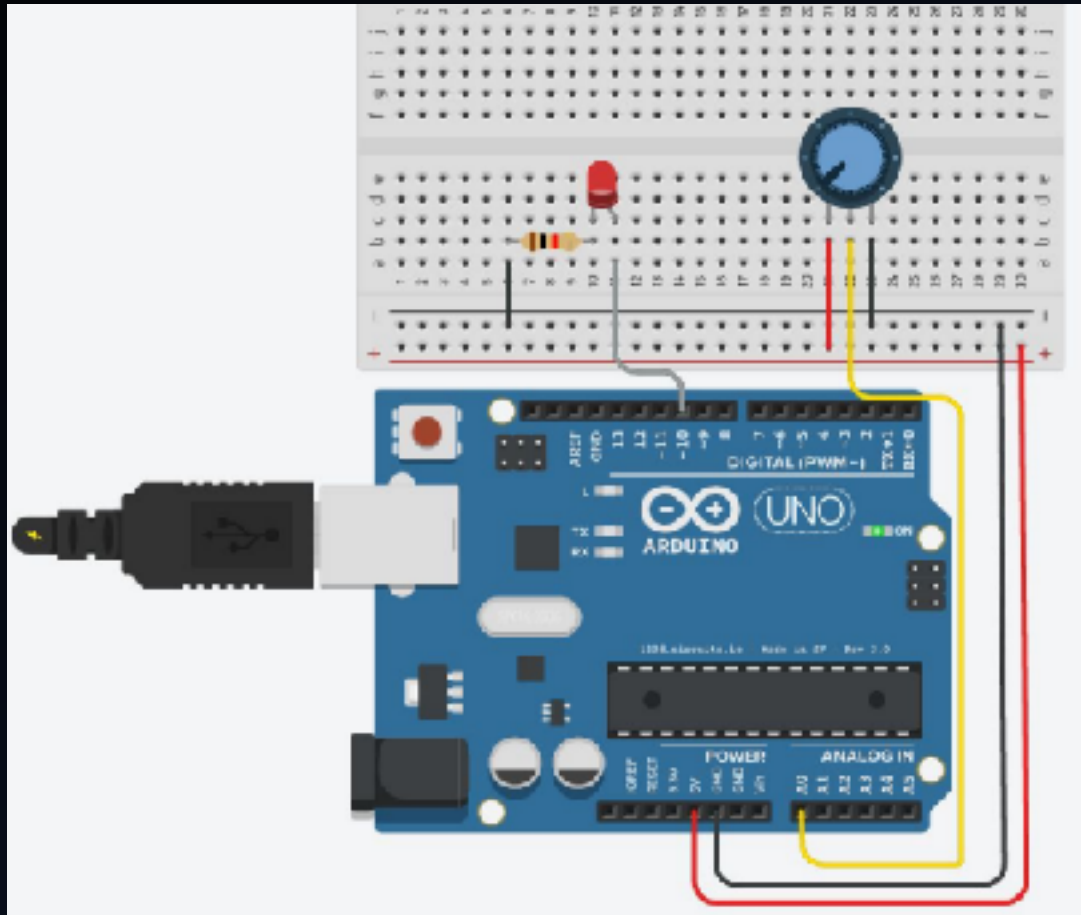
- Objetivos: Montar um circuito capaz de modificar a intensidade da luz através da porta PWM.
- Materiais:
 - LED
 - Protoboard
 - Jumpers
 - Resistor 220 ohm
 - Arduino
 - Potenciômetro
- Potenciômetros: são resistores que possuem resistência ajustável, ao girar o eixo central. Nesse exemplo, usaremos o potenciômetro como regulador de tensão sob o LED.



Exercício 5 - Intensidade de um LED



Exercício 5 - Intensidade de um LED



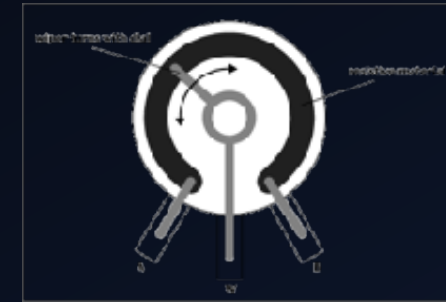
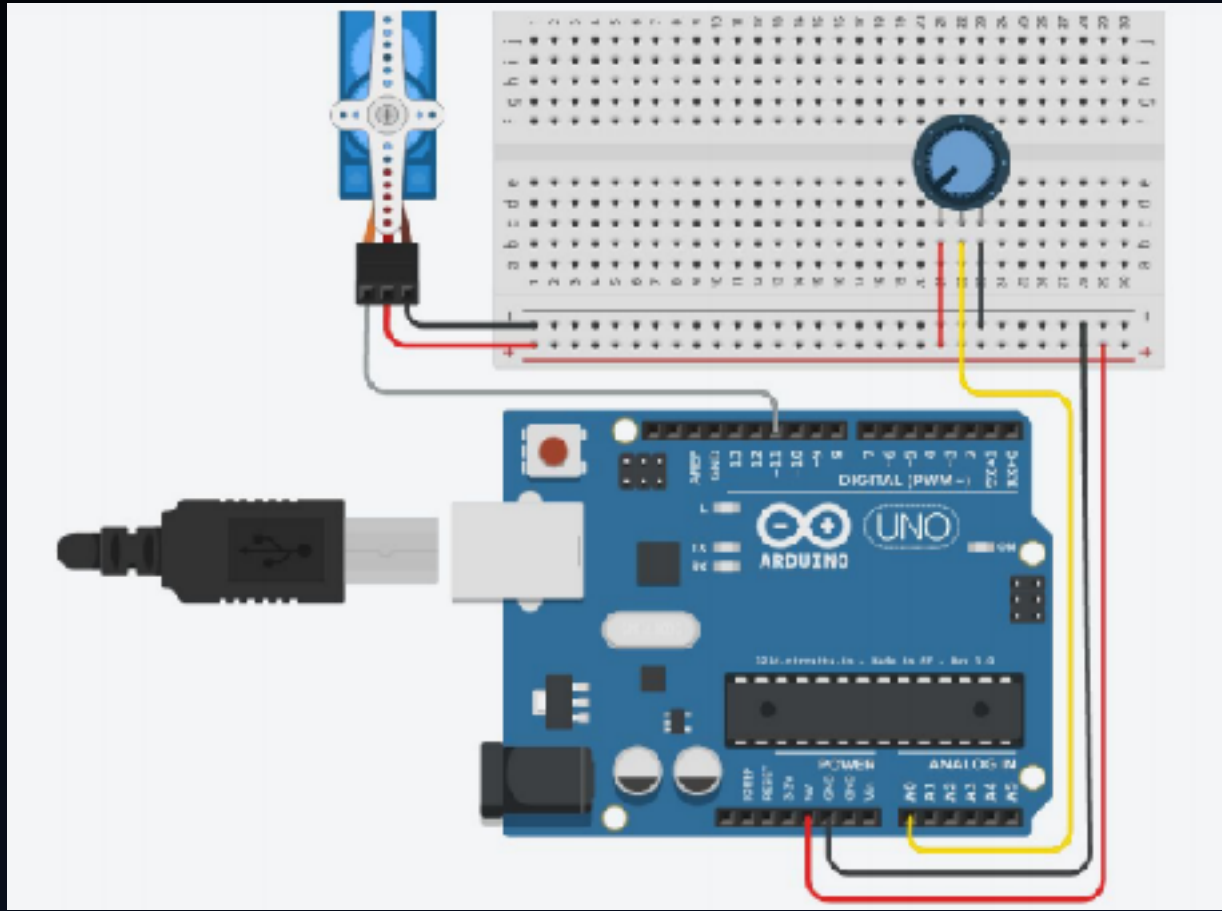
```
#define potPin 0
#define ledPin 10
int valPot = 0;
void setup()
{
    pinMode(ledPin,OUTPUT);
}
void loop()
{
    valPot = analogRead(potPin);
    valPot = map(valPot,0,1023,0,255);
    analogWrite(ledPin,valPot);
}
```

Exercício 6 - Servo motor

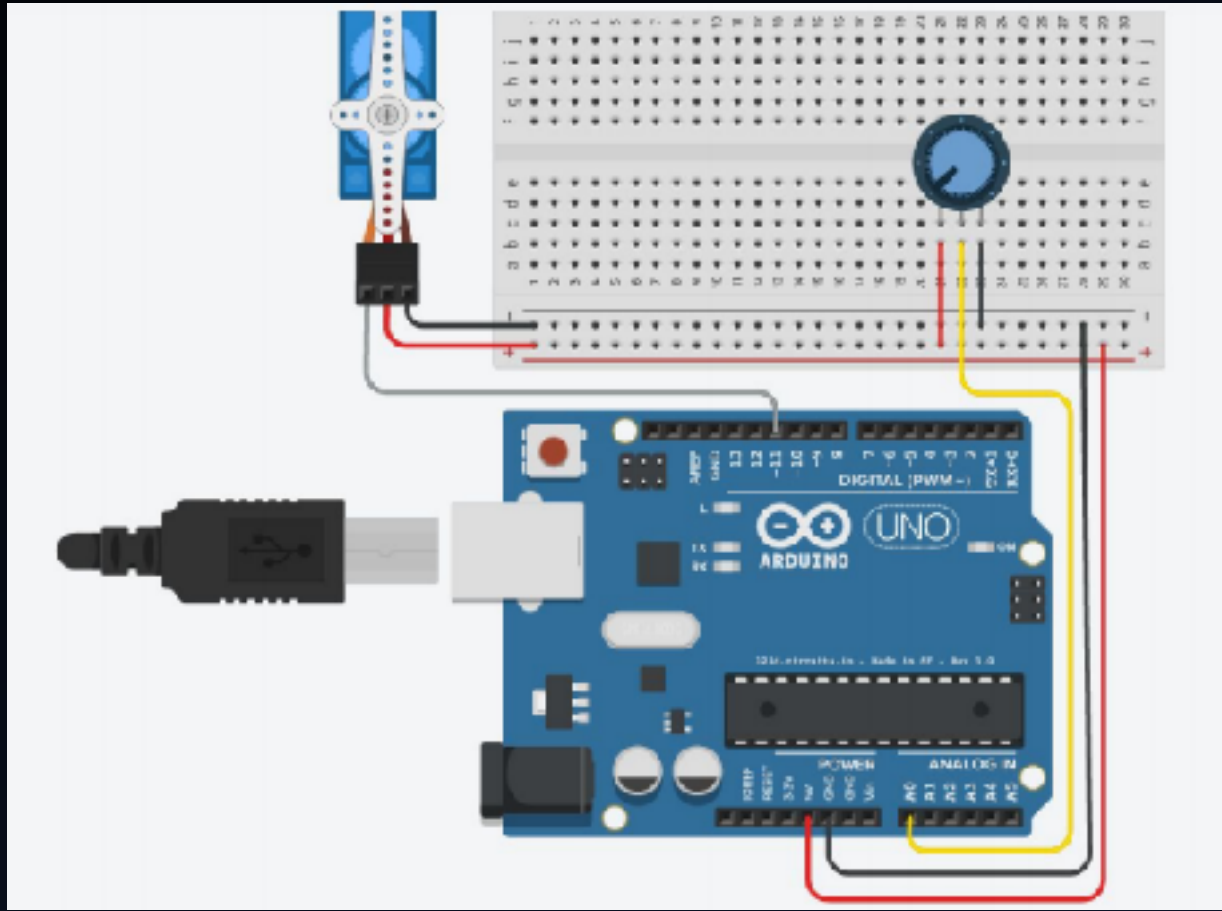
- Objetivo: Mover o servomotor ao rotacionar o eixo do potenciômetro.
- Materiais:
 - Potenciômetro
 - Protoboard
 - Jumpers
 - Arduino
 - Servomotor
- Servomotor: um componente, eletromecânico, que apresenta movimento proporcional a um comando. O eixo dos servo motores possui a liberdade de cerca de 180° graus.



Exercício 6 - Servo motor



Exercício 6 - Servo motor



```
#include <Servo.h>
Servo myservo;

int potpin = 0;
int val;

void setup()
{
  myservo.attach(11);
}

void loop()
{
  val = analogRead(potpin);
  val = map(val, 0, 1023, 0, 179);
  myservo.write(val);
  delay(250);
}
```



CONTATO



alcimarbmx



alcimar_bmx



alcimar_bmx



nathecia



nathecia



natheciacunha

OBRIGADO!