

# ROBOT CLASS



# Sumário

## 1.0 Guia Didático De Montagem - SIGAT

1.1 Guia Didático De Montagem - SIGAT .....	01
---------------------------------------------	----

## 2.0 Conheça Os Componentes Necessários Para A Criação De Um Semáforo Utilizando Arduíno

2.1 Material Necessário .....	02
2.2 Arduíno Uno .....	03
2.3 Protoboard e Led's .....	04
2.4 Resistor(es), Jumpers e Ferramenta de Simulação ...	05

## 3.0 Montagem Do SIGAT

3.1 Conexões .....	06
3.2 Código .....	07
3.3 Explicação das Funções .....	08
3.4 Semáforo em cruzamento de rua.....	09 e 10

## 4.0 SIGAT:

4.1 SIGAT.....	10
----------------	----

## 5.0 Referências Bibliográficas

5.1 Referências do projeto .....	11
----------------------------------	----

## Guia Didático De Montagem - SIGAT

Este guia é destinado a entusiastas da tecnologia e profissionais de engenharia, desenvolvedores e corporações no ramo da tecnologia que buscam implementar uma solução de sinalização de trânsito na cidade de Pereiro, que enfrenta desafios com os controles de tráfego adequados.

O objetivo do SIGAT (Sistema Inteligente de Gestão Automática de Trânsito) é criar um semáforo inteligente que seja aplicável às necessidades específicas da cidade, levando em consideração a segurança dos pedestres e a eficiência do trânsito. Utilizaremos o Arduino como a ferramenta principal para construir este sistema, aproveitando sua simplicidade e flexibilidade para desenvolvedores de todos os níveis.

Neste guia, você aprenderá como montar um semáforo básico, e o uso adequado do sistema em cruzamentos de ruas, com o objetivo de demonstrar o funcionamento de semáforos nas ruas de Pereiro.

No Robot Class, vamos começar com os conceitos básicos e, gradualmente, avançar para projetos mais complexos. Este guia é projetado para ser interativo e prático, permitindo que você aplique os conhecimentos de robótica e programação que você adquiriu em um contexto real e significativo.

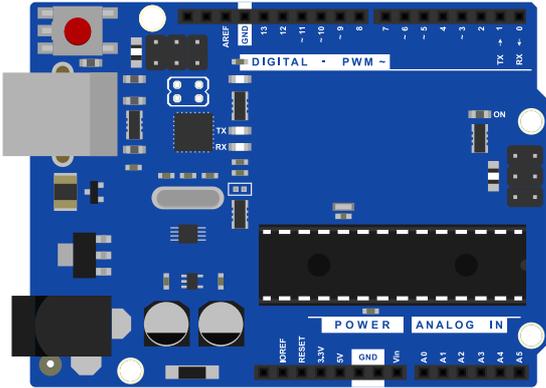
# Conheça Os Componentes Necessários Para a Criação De Um Semáforo Utilizando Arduíno

## Material Necessário:

<b>MATERIAL</b>	<b>QUANTIDADE</b>
<b>Arduino Uno</b>	<b>1</b>
<b>Protoboard (placa de ensaio)</b>	<b>1</b>
<b>Led Vermelha</b>	<b>2</b>
<b>Led Amarela</b>	<b>2</b>
<b>Led Verde</b>	<b>2</b>
<b>Resistor(es)</b>	<b>6</b>
<b>Jumpers</b>	<b>8</b>

# Conheça Os Componentes Necessários Para A Criação De Um Semáforo Utilizando Arduíno

- **Arduíno Uno:**



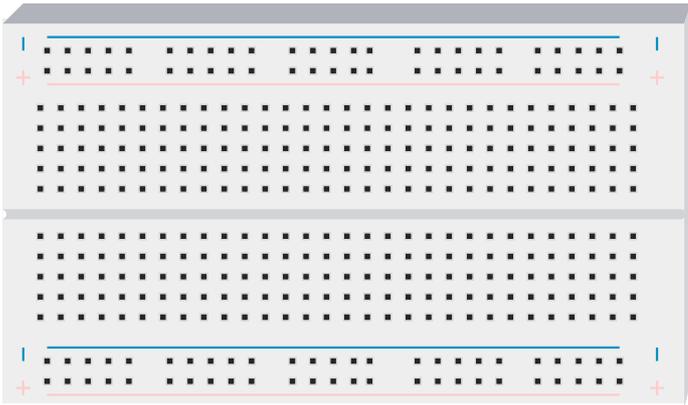
O Arduino Uno é uma placa de microcontrolador muito popular e amplamente utilizada, ideal para iniciantes em eletrônica e programação. Ele é baseado no microcontrolador ATmega328P.

A placa Arduino Uno possui 14 pinos de entrada/saída digitais, 6 entradas analógicas, um ressonador cerâmico de 16 MHz, uma conexão USB, um conector de alimentação, um cabeçalho ICSP e um botão de reset.

Para programar o Arduino Uno, você precisará do Arduino IDE, que permite escrever e carregar código na placa usando a linguagem de programação Arduino.

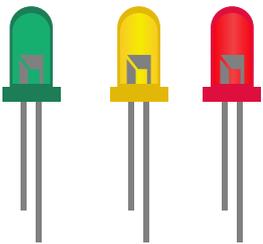
# Conheça Os Componentes Necessários Para A Criação De Um Semáforo Utilizando Arduíno

- **Protoboard**



A protoboard, também conhecida como breadboard, é uma plataforma de construção usada para criar protótipos “semi-permanentes” de circuitos eletrônicos sem a necessidade de solda. A protoboard é ideal para para a construção de diferentes projetos, porém não são ideais para circuitos de alta velocidade, alta corrente ou alta frequência.

- **Led's**



Os LEDs (Light Emitting Diodes) são componentes eletrônicos que emitem luz quando uma corrente elétrica passa por eles. Neste projeto, nosso sistema depende e é constituído por esses LEDs, que por sua vez são das cores: verde, amarelo e vermelho, simulando assim o sistema de sinalização de trânsito.

# Conheça Os Componentes Necessários Para A Criação De Um Semáforo Utilizando Arduíno

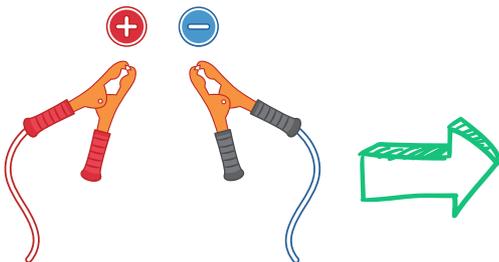
- **Resistor(es)**



Resistores são componentes eletrônicos que são usados para limitar a corrente elétrica que passa através deles. Eles são essenciais em circuitos eletrônicos para controlar a quantidade de corrente que passa para evitar danos aos componentes do circuito.

Os resistores são componentes fundamentais de muitos projetos eletrônicos, incluindo aqueles que envolvem LEDs.

- **Jumpers**



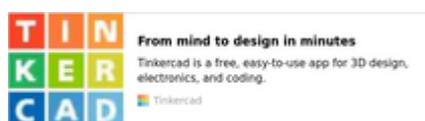
Jumpers são pequenos dispositivos usados para criar conexões físicas entre dois pontos específicos em uma placa de circuito. Eles são feitos de material condutor, como metal, para garantir a transmissão eficiente de sinais elétricos.

Os jumpers permitem modificações fáceis. Diferentemente de componentes soldados, os jumpers podem ser ajustados ou reposicionados sem causar danos à placa de circuito.

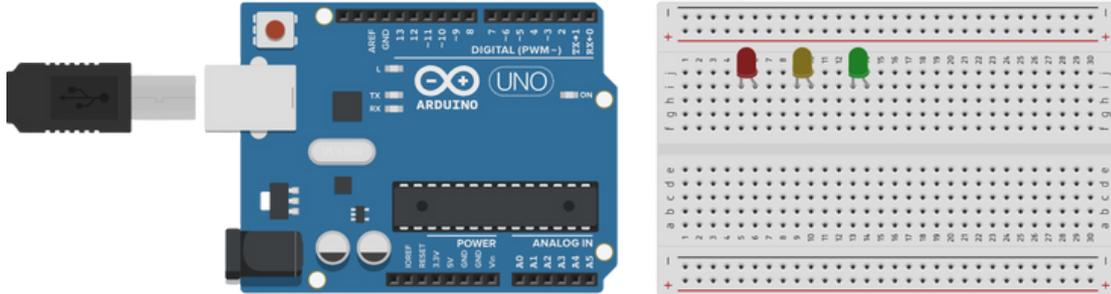
- **Ferramenta de Simulação:**

O **Tinkercad**, é uma plataforma que permite aos usuários criar e testar projetos eletrônicos sem a necessidade de componentes físicos. Ele é especialmente útil para projetos Arduino, pois oferece uma simulação virtual do ambiente de programação e hardware do Arduino.

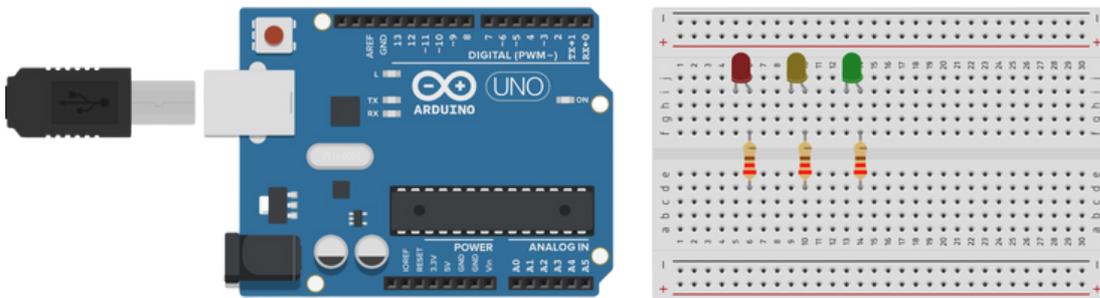
**Link do Tinkercad**



## Montagem o SIGAT :

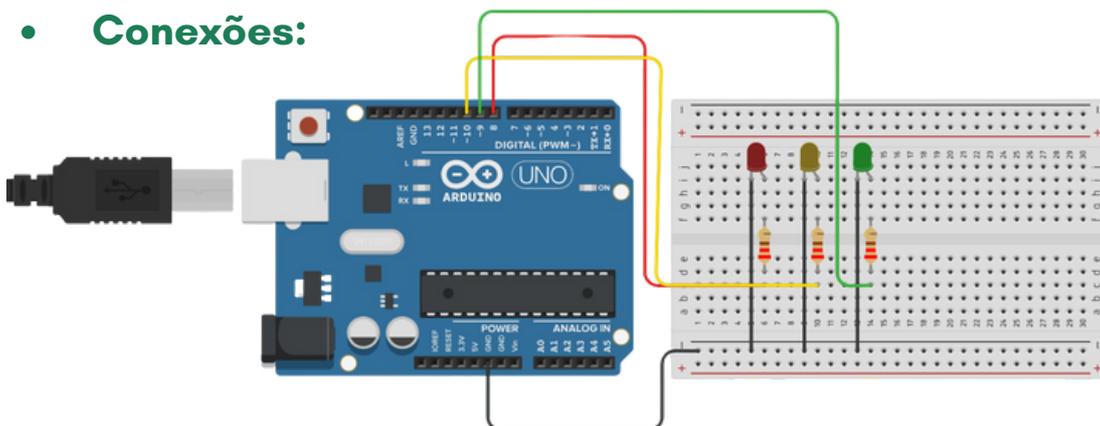


- Primeiro colocamos a placa Arduino uno (que é responsável por conectar os componentes), em seguida adicionamos a protoboard, adicionamos 3 leds com as seguintes cores: **amarelo**, **vermelho** e **verde**.
- Polaridade dos LEDs: A polaridade dos LEDs é crucial para o seu funcionamento. O anodo é o terminal positivo, e o catodo é o terminal negativo. Para que o LED funcione corretamente, o lado positivo (anodo) deve ser conectado à fonte de energia (no caso, um pino do Arduino), e o lado negativo (catodo) deve ser conectado ao GND (terra) da protoboard.



- Três resistores foram acionados com a carga de 220 Ohms cada

### • Conexões:



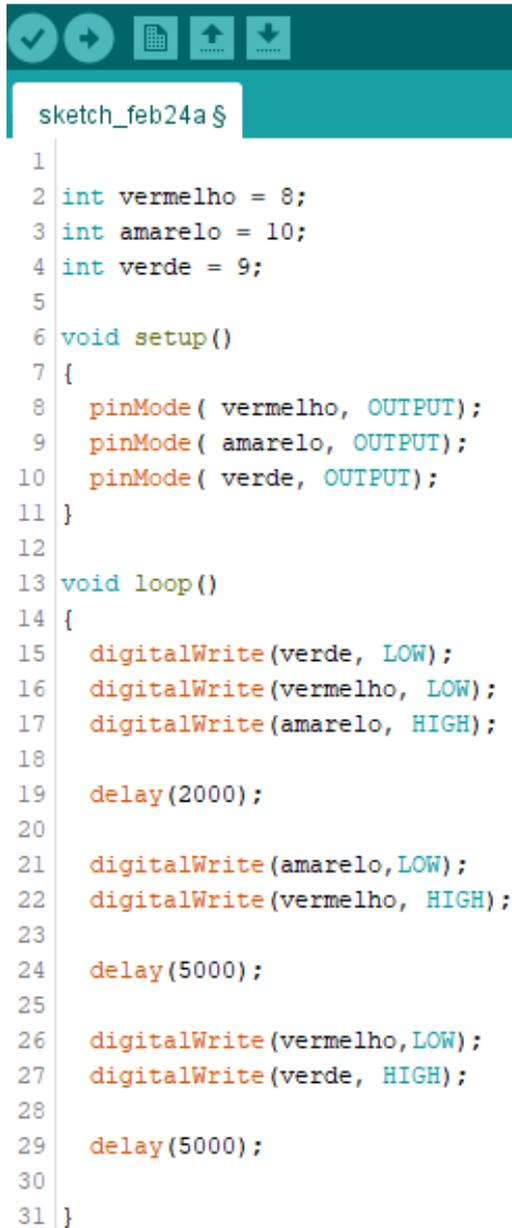
# Montagem do SIGAT :

- **Código:**

```

1
2 int vermelho = 8;
3 int amarelo = 10;
4 int verde = 9;
5
6 void setup()
7 {
8   pinMode( vermelho, OUTPUT);
9   pinMode( amarelo, OUTPUT);
10  pinMode( verde, OUTPUT);
11 }
12
13 void loop()
14 {
15   digitalWrite(verde, LOW);
16   digitalWrite(vermelho, LOW);
17   digitalWrite(amarelo, HIGH);
18
19   delay(2000);
20
21   digitalWrite(amarelo,LOW);
22   digitalWrite(vermelho, HIGH);
23
24   delay(5000);
25   |
26   digitalWrite(vermelho,LOW);
27   digitalWrite(verde, HIGH);
28
29   delay(5000);
30
31 }

```



```

sketch_feb24a §
1
2 int vermelho = 8;
3 int amarelo = 10;
4 int verde = 9;
5
6 void setup()
7 {
8   pinMode( vermelho, OUTPUT);
9   pinMode( amarelo, OUTPUT);
10  pinMode( verde, OUTPUT);
11 }
12
13 void loop()
14 {
15   digitalWrite(verde, LOW);
16   digitalWrite(vermelho, LOW);
17   digitalWrite(amarelo, HIGH);
18
19   delay(2000);
20
21   digitalWrite(amarelo,LOW);
22   digitalWrite(vermelho, HIGH);
23
24   delay(5000);
25
26   digitalWrite(vermelho,LOW);
27   digitalWrite(verde, HIGH);
28
29   delay(5000);
30
31 }

```

## Montagem do SIGAT :

- **Explicação das Funções:**

01. A função `"int vermelho = 8;"` Declara uma variável inteira chamada vermelho e atribui a ela o valor 8. Este valor representa o número do pino digital ao qual um LED vermelho está conectado. Assim como `"int amarelo = 10"` declara que variável amarelo está no pino digital 10. O mesmo sucede na variável verde.

02. `void setup()` - Inicia a função `setup()`, que é executada uma única vez quando o Arduino é inicializado.

03. `pinMode( vermelho, OUTPUT);` - Configura o pino digital definido pela variável vermelho como saída (OUTPUT). Isso significa que o Arduino pode enviar sinais para este pino para controlar o LED. O mesmo acontece com os comandos `"pinMode( amarelo, OUTPUT);"` e `"pinMode( verde, OUTPUT);"`

04. `void loop()` - Inicia a função `loop()`, que é executada repetidamente

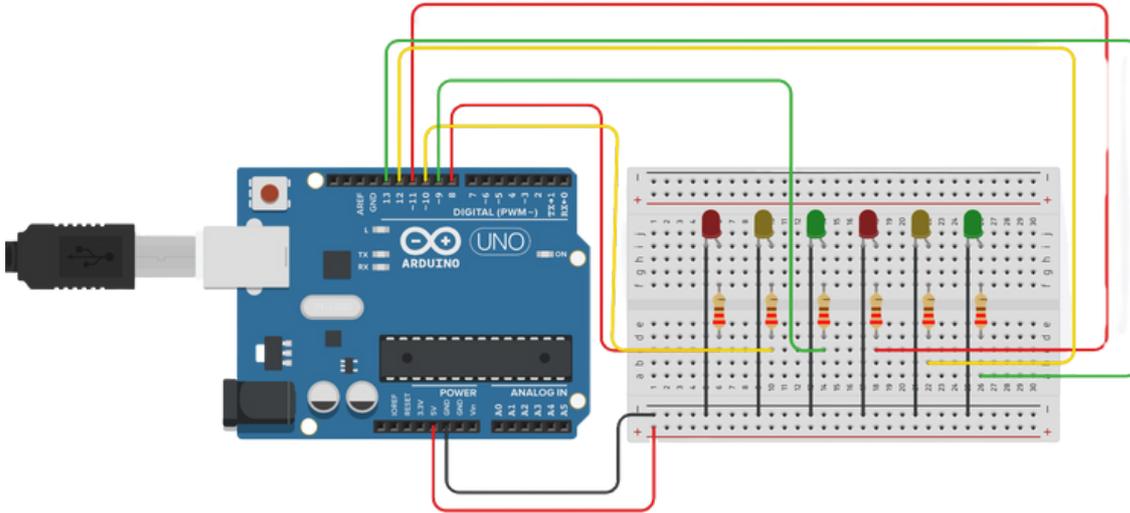
05. LOW - "Desliga" o led definido;

06. HIGH - "liga" o led definido (em 3.3V ou 5V);

07. `delay()` - pausa a execução do programa por um período de tempo especificado em milissegundos.

## Montagem Do SIGAT :

- **Semaforo em cruzamento de ruas:**



- A lógica é simples, só precisamos automatizar/harmonizar ambos os semáforos. Para isso precisamos aumentar o nosso numero de variáveis inteiras declaradas;



```
int vermelho = 8;
int amarelo = 10;
int verde = 9;
int vermelho2 = 11;
int amarelo2 = 12;
int verde2 = 13;
```

- Portanto declaramos também outros pinos digitais de saída



```
void setup()
{
  pinMode( vermelho, OUTPUT);
  pinMode( amarelo, OUTPUT);
  pinMode( verde, OUTPUT);
  pinMode( vermelho2, OUTPUT);
  pinMode( amarelo2, OUTPUT);
  pinMode( verde2, OUTPUT);
}
```

- Começaremos com os ambos LEDs amarelos ligados, posteriormente ligaremos o LED vermelho do semaforo1 e LED verde do semaforo2



```
void loop()
{
  digitalWrite(verde, LOW);
  digitalWrite(vermelho, LOW);
  digitalWrite(amarelo, HIGH);
  digitalWrite(verde2, LOW);
  digitalWrite(vermelho2, LOW);
  digitalWrite(amarelo2, HIGH);

  delay(2000);

  digitalWrite(amarelo,LOW);
  digitalWrite(vermelho, HIGH);
  digitalWrite(amarelo2,LOW);
  digitalWrite(verde2, HIGH);

  delay(5000);
```

## Montagem Do SIGAT :

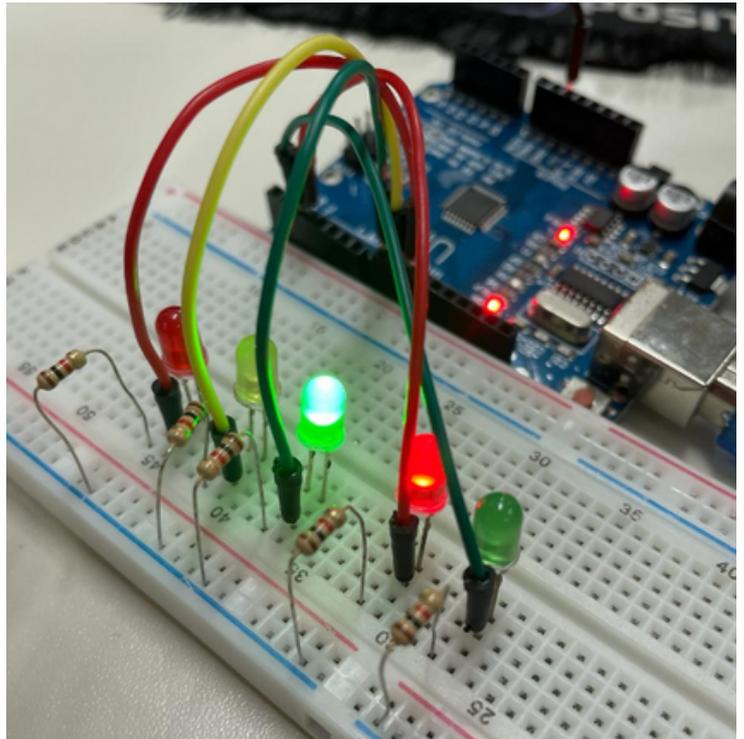
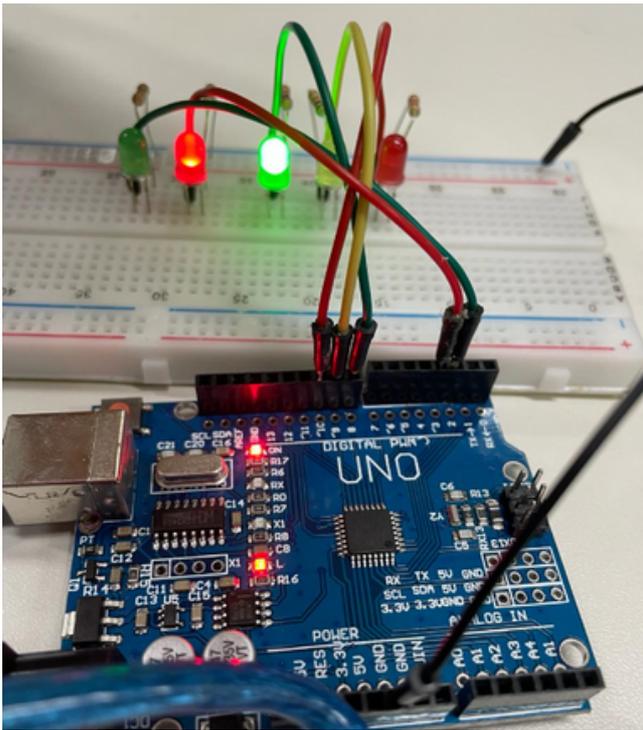
- **Semáforo em cruzamento de ruas:**
- Depois faremos o inverso, acendendo o LED verde do semaforo1 e ligando o LED vermelho do semaforo2;



```
digitalWrite(vermelho,LOW);
digitalWrite(verde, HIGH);
digitalWrite(verde2,LOW);
digitalWrite(vermelho2, HIGH);

delay(5000);
}
```

## SIGAT:



## Referências Bibliográficas:

Arduíno. Disponível em:

<https://www.arduino.cc/reference/tr/language/functions/digital-io/digitalwrite/>

Arduíno. Disponível em:

<https://www.arduino.cc/reference/en/language/functions/digital-io/pinmode/>

Arduíno. Disponível em:

<https://www.arduino.cc/reference/en/language/functions/time/delay/>

Examineja. Disponível em:

<https://examineja.com.br/o-que-e-o-gnd-na-eletronica/>

Maker Hero. Disponível em:

<https://www.makerhero.com/blog/como-criar-um-semaforo-com-arduino/>

Tutorials Point. Disponível em:

[https://www.tutorialspoint.com/arduino/arduino\\_delay\\_function.htm#:~:text=The%20way%20the%20delay\(\),when%20it%20encounters%20this%20function.](https://www.tutorialspoint.com/arduino/arduino_delay_function.htm#:~:text=The%20way%20the%20delay(),when%20it%20encounters%20this%20function.)

Vida de Silício. Disponível em:

<https://www.vidadesilicio.com.br/categoria/prototipagem-e-ferramentas/protoboard-e-jumpers/#:~:text=Os%20jumpers%20s%C3%A3o%20cabos%20ou,eletricidade%20ao%20longo%20do%20mesmo.>

Prodemge. Disponível em:

[https://www.prodemge.gov.br/inova2019/pdf/arquivos/handson\\_arduino2/arduino3.pdf](https://www.prodemge.gov.br/inova2019/pdf/arquivos/handson_arduino2/arduino3.pdf)