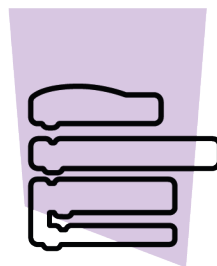
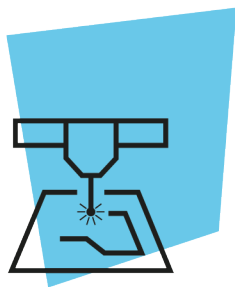
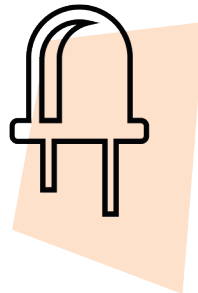
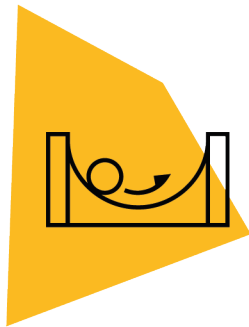
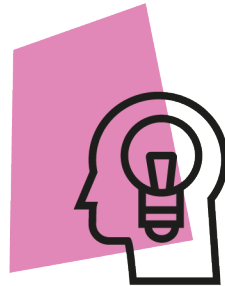


# ROBÓTICA

Módulo 1



## LED e Resistor

AULA 08

## **GOVERNADOR DO ESTADO DO PARANÁ**

Carlos Massa Ratinho Júnior

## **SECRETÁRIO DE ESTADO DA EDUCAÇÃO**

Renato Feder

## **DIRETOR DE TECNOLOGIA E INOVAÇÃO**

Andre Gustavo Souza Garbosa

## **COORDENADOR DE TECNOLOGIAS EDUCACIONAIS**

Marcelo Gasparin

### **Produção de Conteúdo**

Cleiton Rosa

Darice Alessandra Deckmann Zanardini

### **Revisão Textual**

Adilson Carlos Batista

### **Leitura Crítica e Normalização Bibliográfica**

Ricardo Hasper

### **Projeto Gráfico e Diagramação**

Edna do Rocio Becker

### **Ilustração**

Jocelin Vianna (Educa Play)

### **Fotografia**

Stella Maris Oliveira Ludwig (Educa Play)

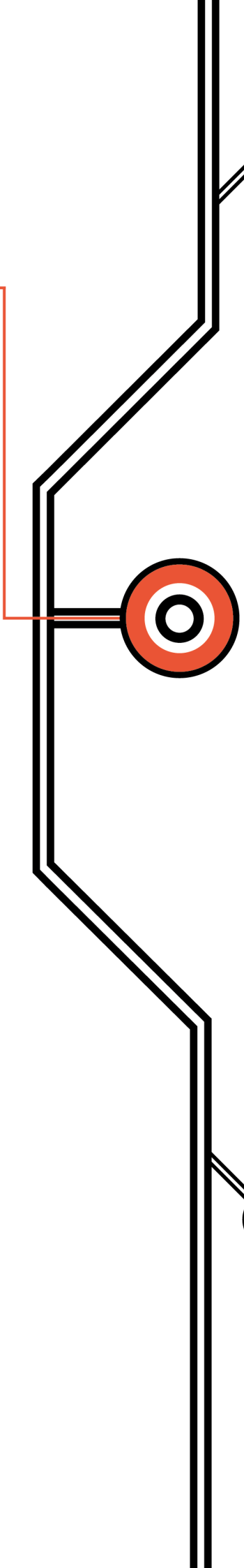
**2021**

Aula 01	Por Que Robótica?
Aula 02	Tensão, Corrente e Resistência
Aula 03	Kit de Robótica
Aula 04	Arduino Uno R3
Aula 05	Softwares Arduino IDE e mBlock
Aula 06	Portas Digitais
Aula 07	Circuito Elétrico
Aula 08	LED e Resistor
Aula 09	Semáforo [Carros]
Aula 10	Semáforo [Cruzamento Carros]
Aula 11	Semáforo [Pedestres]
Aula 12	Semáforo [Cruzamento Carros + Pedestres]
Aula 13	Push Button
Aula 14	Feedbacks + Inventário I
Aula 15	Semáforo [Carros + Pedestres com Botão]
Aula 16	Display 7 Segmentos
Aula 17	Fonte DC + Plug P4
Aula 18	Portas PWM
Aula 19	LED Fade-In
Aula 20	LED Fade-Out
Aula 21	Super Máquina 80's
Aula 22	Super Máquina 2008
Aula 23	Potenciômetro
Aula 24	Buzzer Passivo
Aula 25	LED RGB
Aula 26	Arco-Íris
Aula 27	Sensor LDR
Aula 28	Feedbacks + Inventário II
Aula 29	Sensor de Temperatura
Aula 30	Sensor de Obstáculo IR
Aula 31	Controle Motor DC
Aula 32	Kit Chassi 2WD Robô
Aula 33	Seguidor de Linha
Aula 34	Sensor de Distância
Aula 35	Sensor de Estacionamento
Aula 36	Display LCD 16x2
Aula 37	Trena Digital
Aula 38	Robô Sumô [Estrutura]
Aula 39	Robô Sumô [Programação + Treinamento I]
Aula 40	Robô Sumô [Programação + Treinamento II]
Aula 41	Disputa de Sumôs
Aula 42	Feedbacks + Inventário III

Aula 07  
Circuito Elétrico

# Aula 08 LED e Resistor

Aula 09  
Semáforo [Carros]



# Sumário

<b>Introdução</b>	<b>2</b>
<b>Objetivos desta Aula</b>	<b>2</b>
<b>Competências Gerais Previstas na BNCC</b>	<b>3</b>
<b>Habilidades do Século XXI a Serem Desenvolvidas</b>	<b>4</b>
<b>Lista de Materiais</b>	<b>4</b>
<b>Roteiro da Aula</b>	<b>5</b>
1. Contextualização	5
2. Montagem e Programação	14
3. Feedback e Finalização	23
<b>Videotutorial</b>	<b>24</b>
<b>Referências</b>	<b>25</b>



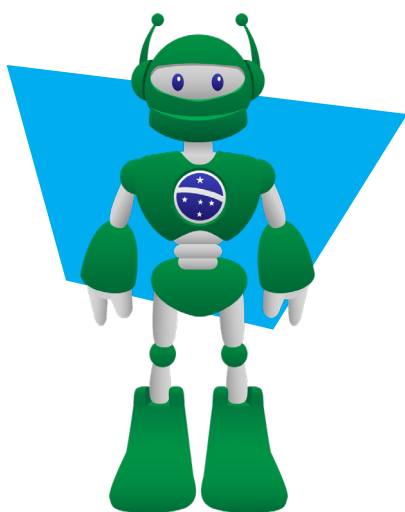
## Introdução

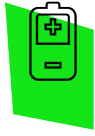
Nesta aula, aprenderemos mais sobre um componente corriqueiro em nosso cotidiano - o LED, algumas vezes “compreendido” - ou, talvez, “confundido” - como outro item elétrico. Também veremos como fazer uso dos Resistores, componente que mencionamos em aulas anteriores e é essencial para os circuitos elétricos. Vamos lá?



## Objetivos desta Aula

- Conhecer os conceitos de LED e Resistor;
- Reconhecer a Resistência Elétrica de um Resistor;
- Saber como definir os Resistores a serem utilizados em circuitos com LEDs;
- Programar o acendimento de LEDs em três projetos com Resistores.





## Competências Gerais Previstas na BNCC

**[CG02]** - Exercitar a curiosidade intelectual e recorrer à abordagem própria das ciências, incluindo a investigação, a reflexão, a análise crítica, a imaginação e a criatividade, para investigar causas, elaborar e testar hipóteses, formular e resolver problemas e criar soluções (inclusive tecnológicas) com base nos conhecimentos das diferentes áreas.

**[CG04]** - Utilizar diferentes linguagens – verbal (oral ou visual-motora, como Libras, e escrita), corporal, visual, sonora e digital –, bem como conhecimentos das linguagens artística, matemática e científica, para se expressar e partilhar informações, experiências, ideias e sentimentos em diferentes contextos e produzir sentidos que levem ao entendimento mútuo.

**[CG05]** - Compreender, utilizar e criar tecnologias digitais de informação e comunicação de forma crítica, significativa, reflexiva e ética nas diversas práticas sociais (incluindo as escolares) para se comunicar, acessar e disseminar informações, produzir conhecimentos, resolver problemas e exercer protagonismo e autoria na vida pessoal e coletiva.

**[CG09]** - Exercitar a empatia, o diálogo, a resolução de conflitos e a cooperação, fazendo-se respeitar e promovendo o respeito ao outro e aos direitos humanos, com acolhimento e valorização da diversidade de indivíduos e de grupos sociais, seus saberes, identidades, culturas e potencialidades, sem preconceitos de qualquer natureza.

**[CG10]** - Agir pessoal e coletivamente com autonomia, responsabilidade, flexibilidade, resiliência e determinação, tomando decisões com base em princípios éticos, democráticos, inclusivos, sustentáveis e solidários.



## Habilidades do Século XXI a Serem Desenvolvidas

- Pensamento crítico;
- Afinidade digital;
- Resiliência;
- Resolução de problemas;
- Colaboração;
- Comunicação.



## Lista de Materiais

- 01 Placa Arduino Uno R3;
- 01 Cabo USB;
- 01 Placa Protoboard;
- 03 LEDs;
- 04 Jumpers Macho-Macho;
- 03 Resistores 220 Ohms;
- Resistores Variados;
- 01 Notebook;
- Software mBlock ou Arduino IDE.



## Roteiro da Aula

### 1. Contextualização (15min):

“Lâmpada de LED” - talvez você já tenha ouvido esta expressão... Mas será que ela está correta? Quando falamos em LED, o que vêm à sua mente? Cartões de Natal que piscam, indicadores de equipamentos eletrônicos ligados, “lâmpadas” com consumo energético econômico? Procure pensar, com seus colegas, outros exemplos de utilização de LEDs.

Além de pensarmos nestas utilizações, precisaremos também compreender o que é um LED. Talvez você já tenha percebido, pelas questões que levantamos acima, que não se trata de uma “lâmpada”. Do que se trata, então?

É comum a associação do LED a uma lâmpada, devido à luz emitida, porém o LED nada mais é que um diodo emissor de luz (em inglês, Light Emitting Diodo - por isso a sigla LED).



### Para Saber Mais...

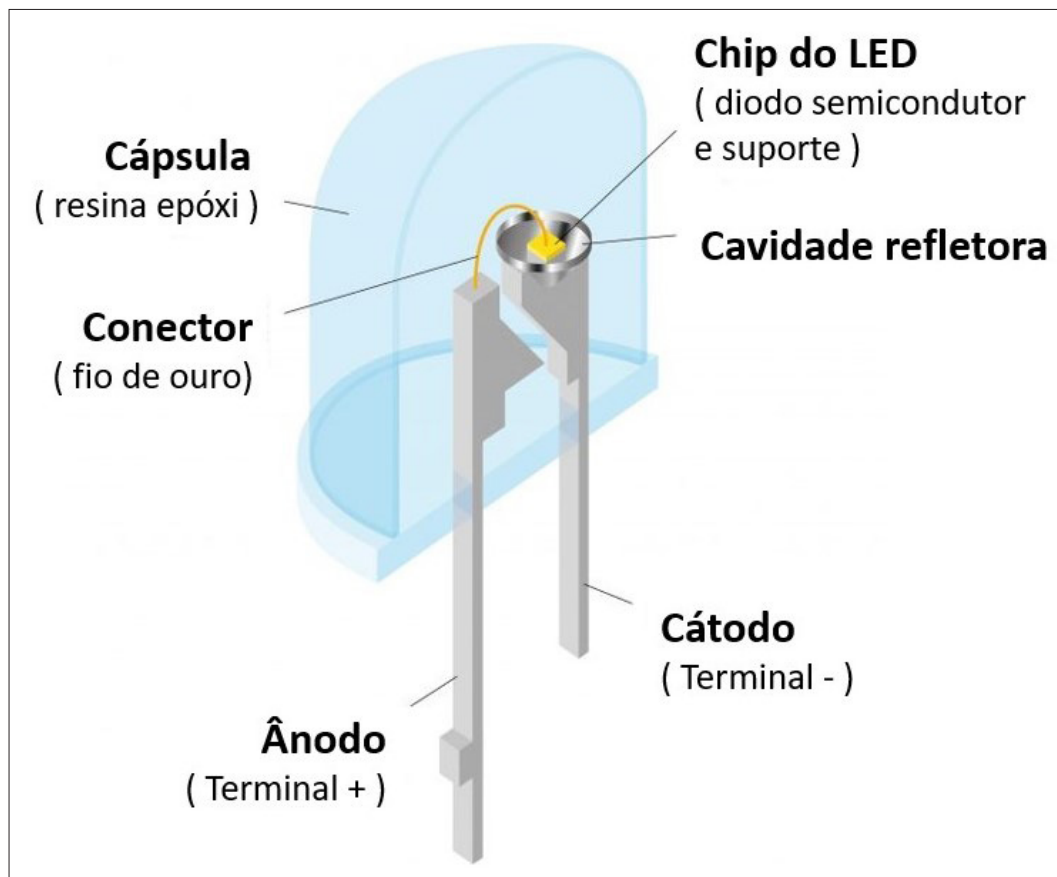
#### O que é um diodo?

É um componente eletrônico, com baixa corrente elétrica e, geralmente, feito de materiais semicondutores, o qual, através de suas polaridades, permite que a corrente siga em apenas um sentido.

A emissão de luz pelo LED depende da passagem de corrente elétrica pelos seus terminais (ânodo e cátodo), conforme sua polarização. Na aula anterior, **Aula 7 - Circuito Elétrico**, conhecemos as características de um circuito elétrico e esse será o princípio que adotaremos para “acender” o LED. Então, antes de o ligarmos, analise a estrutura física do LED e perceba as características deste componente:



Figura 01 - Detalhe da estrutura física do LED

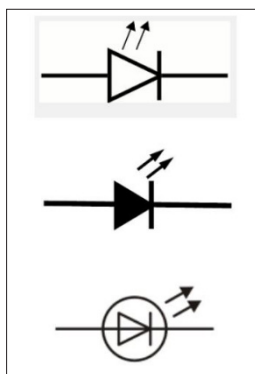


Fonte: Lichtfragen (adaptado)

Como podemos perceber, a luz emitida pelo LED é oriunda da sua composição interna, não da sua cápsula, conforme imagem acima. Por isso, você pode encontrar LEDs, por exemplo, amarelos ou vermelhos, em cápsulas transparentes, uma vez que a emissão das cores depende dos elementos químicos utilizados em seu interior e do comprimento de onda do espectro visível gerado quando os elétrons passam por este semicondutor. Além dos LEDs de espectro visível, há também LEDs de espectros infravermelho e ultravioleta.

Em diagramas esquemáticos de circuitos, como os vistos na **Aula 07 - Circuito Elétrico**, utilizamos os seguintes símbolos para a representação dos LEDs:

Figura 02 - Representação gráfica de LED



Observe que a representação gráfica dos LEDs utiliza a mesma simbologia para diodos, com o acréscimo das setas para simbolizar a emissão de luz.



### Para Saber Mais...

Encontrado, atualmente, em várias cores, o primeiro LED “colorido” a surgir foi o vermelho, em 1962, e depois o verde. Dez anos depois, surgiu o LED amarelo. O LED azul foi um grande desafio, e seu surgimento data os anos de 1990.

Os cientistas japoneses Isamu Akasaki, Hiroshi Amano e Shuji Nakamura receberam, em 2014, o Prêmio Nobel da Física pelo desenvolvimento do LED azul a partir da transformação do gás nitreto de gálio (GaN) em um cristal.

A partir do LED azul, desenvolveu-se fontes de luz branca com grande potencial e baixo consumo energético, contribuindo, no campo da iluminação, como um recurso econômico de energia ao uso das, até então, habituais lâmpadas incandescentes e fluorescentes.

Caso sua escola disponha de baterias 3 V, que tal experimentar já o acendimento do LED? Relembre os conceitos de circuito que aprendemos na **Aula 07 - Circuito Elétrico** e acenda um LED

utilizando apenas uma bateria 3 V. Se a sua escola não dispuser destas baterias, que tal utilizar o próprio Arduino, sem programá-lo, para acender o LED? Na **Aula 06 - Portas Digitais**, você viu como fazer o blink no próprio Arduino. Nesta aula, você fará o blink utilizando o LED e a porta 3.3 V do Arduino!

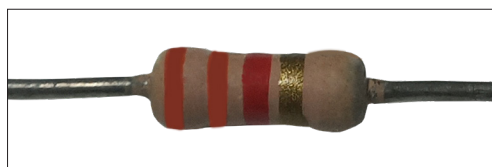
Para a programação do LED com Arduino, utilizaremos uma porta digital. Nas próximas aulas, você aprenderá como programar, utilizando outros comandos, o LED com os efeitos fade-in e fade-out e seu controle por potenciômetro.

O LED, assim como outros componentes que conheceremos no decorrer de nossas aulas, precisa, ao ser inserido em um circuito elétrico, da utilização de um Resistor, o qual limita o fluxo da corrente que está passando pelo circuito elétrico. Portanto, a associação do Resistor ao LED visa ao aproveitamento máximo do brilho do LED sem danos ao componente por diminuir, no circuito, a corrente fornecida pelo Arduino quando utilizamos a porta de 5 V (relembre alguns conceitos da física e da eletrônica na **Aula 02 - Corrente, Tensão e Resistência** e na **Aula 07 - Circuito Elétrico**).



### Para Saber Mais...

Figura 03 - Resistor axial



Fonte: SEED/DTI/CTE


Nesta aula, abordamos o Resistor axial, modelo mais utilizado em circuitos elétricos. Na sequência das aulas de robótica - Módulo 1 e no Módulo 2, você conhecerá outros componentes eletrônicos, como o potenciômetro e o joystick, que são modelos de Resistores flexíveis por possibilitarem variação da resistência.

Ao longo das nossas aulas, você perceberá que o Resistor é um componente que frequentemente será utilizado nos projetos – não apenas associados aos LEDs, mas também a outros componentes eletrônicos – e, por existirem Resistores com variados valores de resistência, precisamos aprender a identificar cada tipo. Para isso, pode ser utilizado um multímetro ou a tabela de cores que conheceremos a seguir.

Há um padrão internacional de marcação que determina os valores do Resistor através de linhas coloridas pintadas em seu corpo cerâmico. Os Resistores mais utilizados possuem quatro ou cinco linhas e na figura abaixo temos a representação desses e dos demais, com três e seis linhas, com os valores atribuídos a cada linha do Resistor (por isso, é importante sabermos como calcular a resistência ao montarmos nossos projetos).

Como você pode encontrar Resistores com quantidades variadas de linhas coloridas, e precisa reconhecer o que cada linha representa, faça a leitura do Resistor considerando o seguinte esquema de cores:

Figura 04 - Tabela de cores para cálculo de Resistores



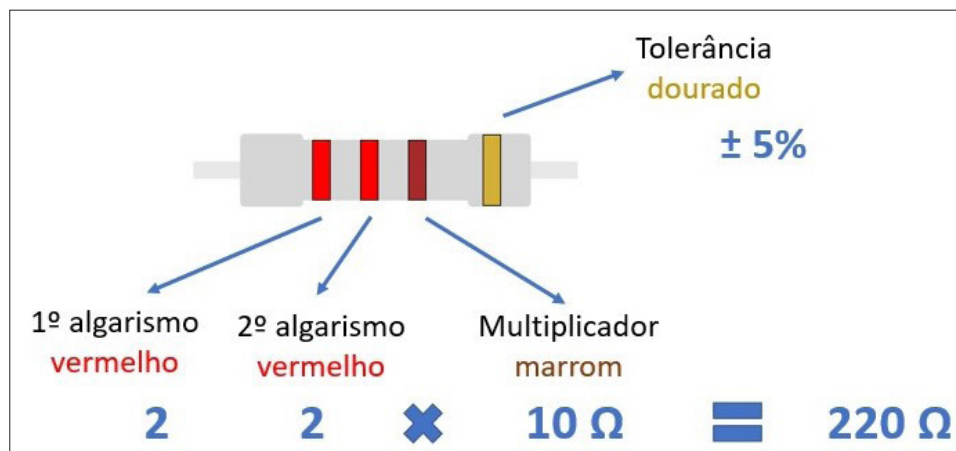
1º algarismo	2º algarismo	3º algarismo	multiplicador	tolerância	temperatura
0	0	0	1 Ω		
1	1	1	10 Ω	± 1%	100 PPM/°C
2	2	2	100 Ω	± 2%	50 PPM/°C
3	3	3	1000 Ω		15 PPM/°C
4	4	4	10000 Ω		25 PPM/°C
5	5	5	100000 Ω	± 0,5%	
6	6	6	1000000 Ω	± 0,25%	10 PPM/°C
7	7	7		± 0,10%	5 PPM/°C
8	8	8		± 0,05%	
9	9	9			
			0,1 Ω dourado	± 5% dourado	
			0,01 Ω prateado	± 10% prateado	

Fonte: SEED/DTI/CTE

Para saber a posição correta do Resistor, visando à leitura das cores, verifique um espaçamento maior entre duas linhas – este espaçamento maior entre linhas ficará à sua direita, como na imagem acima, e as linhas à esquerda são os algarismos e multiplicador. Caso o Resistor possua o total de três linhas, ele não tem a linha de tolerância ou margem de erro – neste caso, considere uma tolerância média de 20% e a linha do multiplicador estará totalmente à direita. Já os Resistores com seis linhas coloridas possuem um coeficiente de temperatura que indica a variação do valor do Resistor conforme a mudança de temperatura.

Como vimos na figura acima, cada linha colorida informa um valor que, quando associado aos demais, determina a resistência em Ohms. No exemplo abaixo, temos a identificação de um Resistor de  $220 \Omega$ , indicado para utilização com LEDs, a partir da leitura de suas linhas coloridas.

Figura 05 - Cálculo de Resistor  $220 \Omega$



Fonte: SEED/DTI/CTE

No exemplo acima, a primeira e a segunda linha indicam os algarismos; a terceira linha, o multiplicador, ou seja, por quanto você precisa multiplicar seus algarismos. Portanto, no exemplo acima temos  $2 \times 2 \times 10 \Omega$ , o que nos dá o valor de  $220 \Omega$ . A quarta faixa nos indica a tolerância – então o valor deste Resistor pode variar de  $209 \Omega$  a  $231 \Omega$ .

No início da realização de projetos com Resistores, pode ser difícil identificar, nas linhas coloridas, quais cores se referem aos

algarismos e quais se referem ao multiplicador. Pesquisando na Internet, você encontra calculadoras on-line que podem auxiliar este processo - e se tranquilize, logo você “pegará” o jeito e identificará com mais facilidade, utilizando a tabela de cores, o valor dos diversos Resistores presentes em seu kit de robótica.

Por não possuir polaridade, os Resistores podem ser utilizados tanto em corrente contínua quanto em corrente alternada e, no momento de utilização do Resistor no circuito elétrico, a posição em que ele é colocado não influencia sua utilização. Por isso, o foco está no uso adequado do Resistor conforme a tensão presente no circuito.

Como vimos na **Aula 02 - Corrente, Tensão e Resistência**, a Lei de Ohm é utilizada para definir a resistência a ser aplicada no circuito elétrico. Ao lembrarmos os conteúdos desta aula e da **Aula 07 - Circuito Elétrico**, temos que a corrente elétrica é o movimento dos elétrons por um meio condutor - e quanto mais elétrons percorrerem esta corrente, maior sua intensidade. No caso do Arduino, placa de prototipagem eletrônica sobre a qual faremos nossos projetos, sua amperagem é de 40mA e portas com tensão de 3.3 e 5 V. Então, para cálculo da resistência a ser utilizada não apenas com o LED, mas também com outros componentes eletrônicos, utilize a Primeira Lei de Ohm, a qual permite cálculos da corrente, da resistência e da tensão.

Para o cálculo do Resistor adequado a um LED, adota-se a Lei de Ohm, na qual Resistência = Tensão / Corrente, porém, no circuito, é preciso definir *tensão da fonte - tensão do LED*.

Neste caso,

$$R = \frac{V_{cc} - V_{LED}}{Corrente_{LED}}$$

Outro ponto a saber, para definir qual Resistor é adequado ao LED, se o LED é difuso, ou seja, com a cápsula na cor a ser emitida, ou alto brilho, quando sua cápsula é transparente (independentemente da cor a ser emitida).

Geralmente, os LEDs alto brilho (com cápsula transparente) utilizam corrente entre 10mA e 20mA, enquanto os LEDs difuso (com cápsula colorida) utilizam corrente entre 5mA e 10mA. Para maior precisão, pode-se consultar, junto aos fabricantes de componentes eletrônicos, o *datasheet*, ou seja, as especificações técnicas dos

componentes (tensão, corrente e demais informações). Caso não seja possível a consulta ao *datasheet*, pode-se utilizar uma tabela padrão, como a disponível no quadro abaixo:

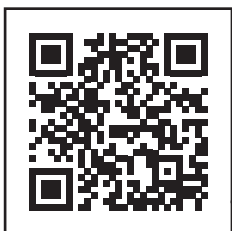
Quadro 01 - Valores habituais de LEDs 5mm alto brilho

LED ALTO BRILHO	TENSÃO EM VOLTS (V)	CORRENTE EM MILIAMPERES (mA)
Vermelho	1,8 V – 2,0 V	10 mA - 20 mA
Laranja	1,8 V – 2,0 V	10 mA - 20 mA
Amarelo	1,8 V – 2,0 V	10 mA - 20 mA
Verde	2,0 V – 2,5 V	10 mA - 20 mA
Azul	2,5 V – 3,0 V	20 mA
Branco	2,5 V – 3,0 V	10 mA - 20 mA



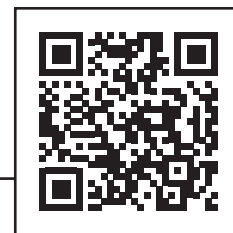
### Para Saber Mais...

Conheça ferramentas on-line que te auxiliarão na escolha de resistores conforme o projeto a ser desenvolvido:



[Resistor Color Code Calculator](#)

[Calculador de LED](#)

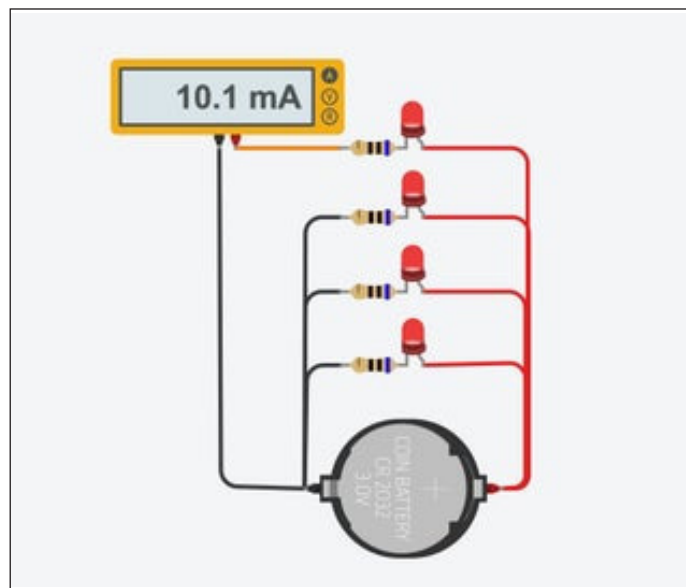




## Treine Mais!

A plataforma de prototipagem Tinkercad possui um espaço de aprendizagem. Na lição **Series and Parallel Circuits**, disponível em inglês, é possível aprender sobre dois tipos fundamentais de circuitos – circuitos em série e paralelos – e descobrir como aplicá-los para o acendimento de vários LEDs de maneira mais eficiente.

Figura 06 - LEDs e Resistores conectados a uma fonte de tensão



Fonte: Tinkercad



[Lição Series and Parallel Circuits](#)



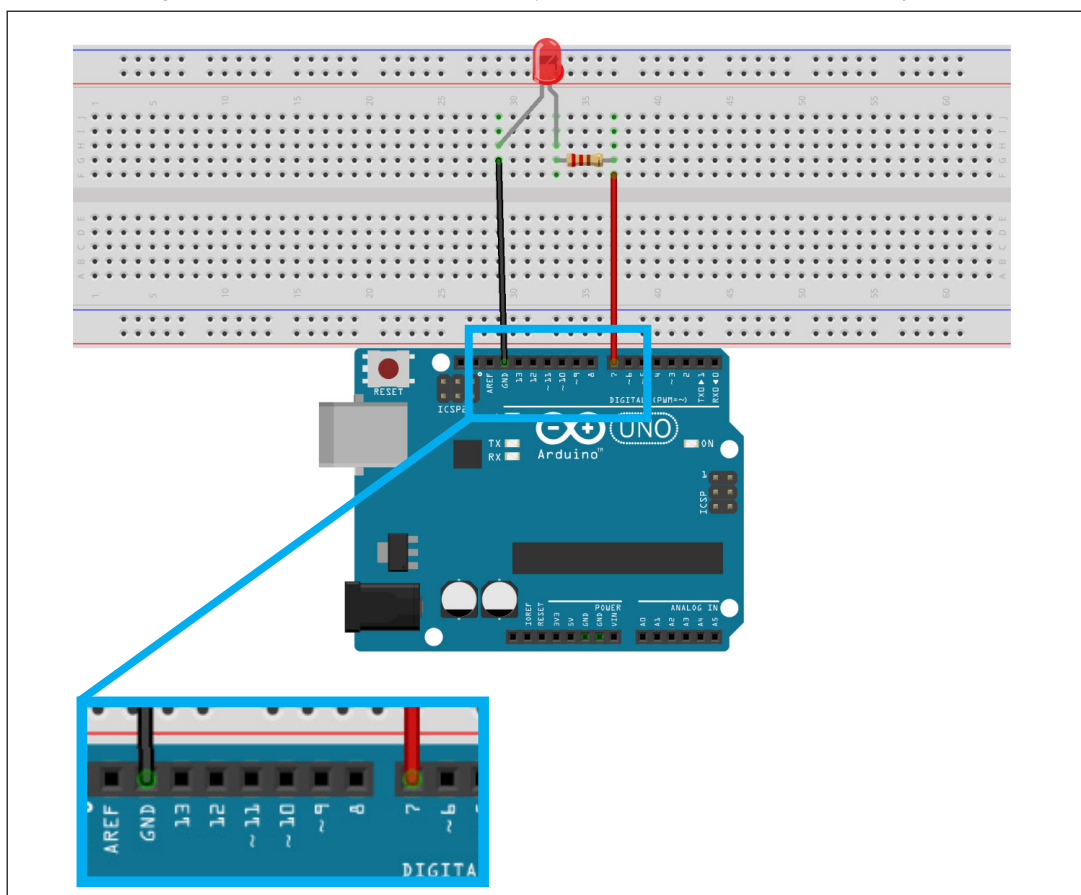
## 2. Montagem e Programação (60min):

Esta aula apresenta três projetos com LEDs e Resistores, demonstrando as possibilidades de conexão destes componentes. Nossa sugestão é que o professor organize os alunos para que cada grupo escolha um dos projetos a executar, compartilhando com os colegas, ao final da aula, as experiências na realização do projeto escolhido.

### Projeto 1 - Ligando um LED com Resistor

Na Protoboard, encaixe o LED e, no seu terminal positivo (terminal mais longo), conecte o Resistor. No terminal negativo do LED (terminal mais curto), conecte 1 Jumper até a porta GND do Arduino e, no terminal da direita do Resistor, conecte outro Jumper até a porta digital 7 do Arduino, conforme indicado na figura 07.

Figura 07 - Conexões dos componentes eletrônicos do Projeto 1

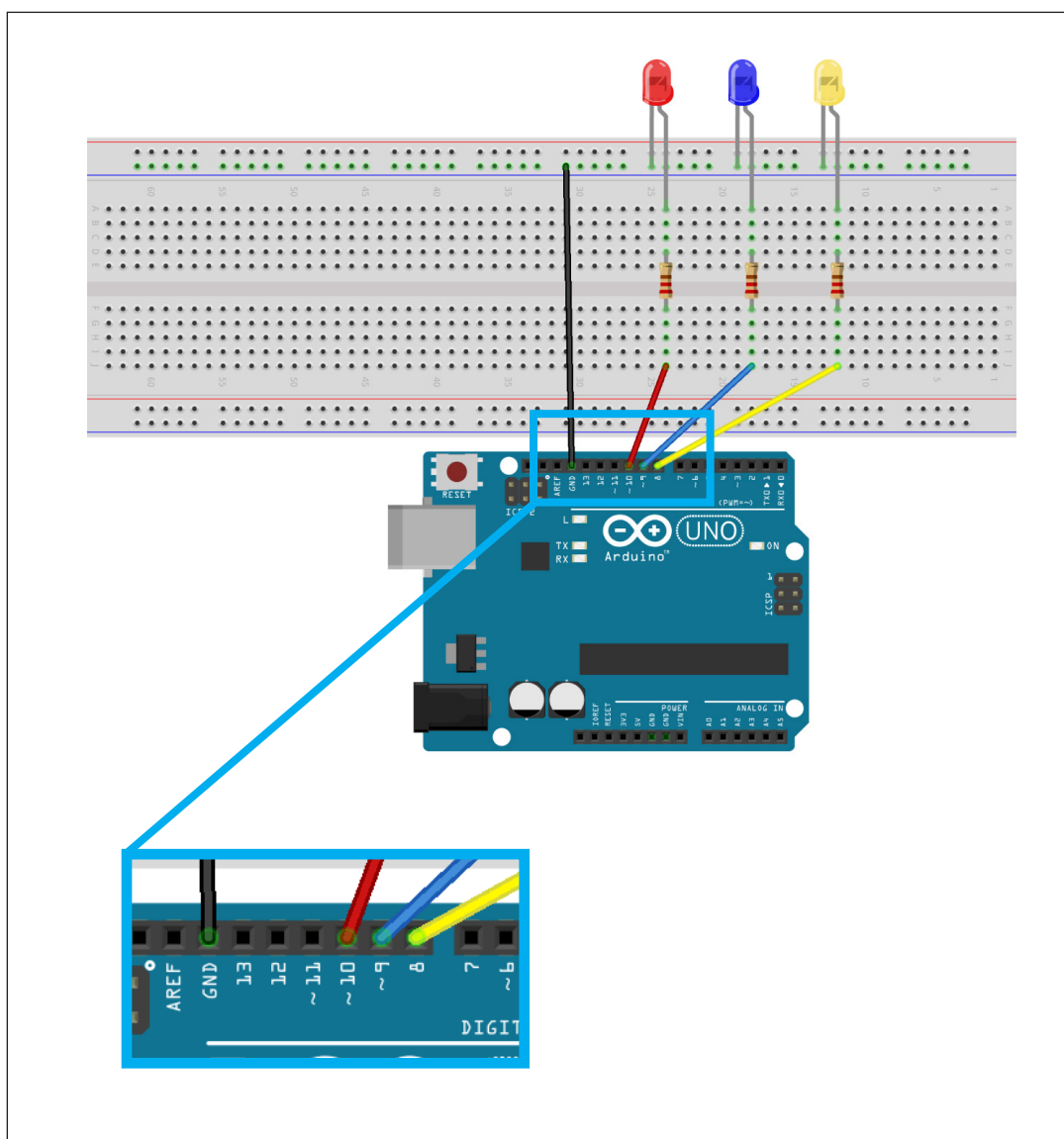


Fonte: Fritzing

## Projeto 2 - Ligando três LEDs com três Resistores

Na Protoboard, encaixe os 3 LEDs da seguinte maneira: terminais negativos (terminais mais curtos) na linha lateral azul da Protoboard e terminais positivos nas colunas verticais centrais da Protoboard. Conecte 3 Resistores, um para cada terminal positivo dos LEDs, até as colunas verticais inferiores da Protoboard. Conecte 3 Jumpers nos terminais inferiores de cada Resistor até as três portas digitais 8, 9 e 10 do Arduino. Por fim, conecte 1 Jumper entre a porta GND do Arduino e a linha lateral azul da Protoboard, conforme indicado na figura 08.

Figura 08 - Conexões dos componentes eletrônicos do Projeto 2

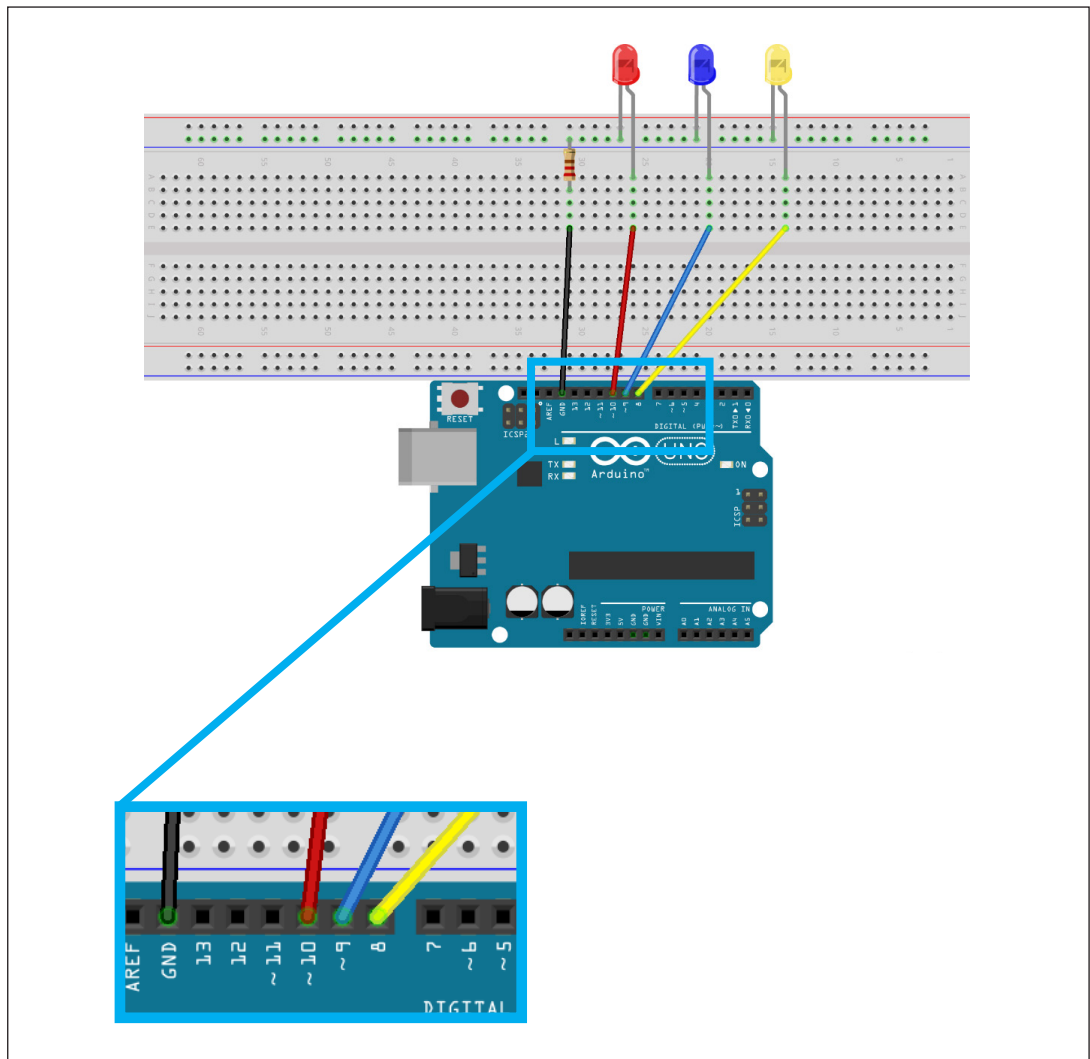


Fonte: Fritzing

### Projeto 3 - Ligando três LEDs com um Resistor

Na Protoboard, encaixe os 3 LEDs da seguinte maneira: terminais negativos (terminais mais curtos) na linha lateral azul da Protoboard e terminais positivos nas colunas verticais centrais da Protoboard. Conecte 1 Resistor entre a linha lateral azul e uma coluna superior central da Protoboard. Conecte 3 Jumpers nos terminais positivos de cada LED (terminal mais longo) até as três portas digitais 8, 9 e 10 do Arduino. Por fim, conecte 1 Jumper entre a porta GND do Arduino e o terminal inferior do Resistor, conforme indicado na figura 09.

Figura 09 - Conexões dos componentes eletrônicos do Projeto 3



Fonte: Fritzing



## Agora, vamos programar!

Como vimos acima, esta aula apresenta três projetos utilizando LEDs e Resistores. Abaixo, apresentamos as programações para cada projeto - então, fique atento para utilizar a programação destinada ao projeto que você e seus colegas escolheram.

Com os componentes eletrônicos montados, vamos programar, por codificação e por blocos, os nossos projetos.

### i. Linguagem de programação por código

Conecte a placa Arduino ao computador, através de um cabo USB, para que ocorra a comunicação entre a placa microcontroladora e o software Arduino IDE.

No software IDE, crie um sketch e lembre-se de selecionar a porta que o computador atribuiu ao Arduino; então, escreva ou copie e cole o código-fonte de programação correspondente ao projeto que você montou na Protoboard, conforme apresentado no quadro 2, 3 e 4.

#### Quadro 02 - Código-fonte da programação do Projeto 1: Ligando um LED com Resistor na linguagem do Arduino (Wiring)

```
/* Projeto 1: Ligando um LED com Resistor */

int Pin_led = 7; /* Define o controle do LED através do pino
7. */

void setup() {
  pinMode(Pin_led, OUTPUT); /* Define o pino de controle do
LED como SAÍDA. */
}

void loop() {
  digitalWrite(Pin_led, HIGH); /* Energiza o pino 7 (liga o
LED). */
  delay(1000); /* Aguarda 1 segundo. */
  digitalWrite(Pin_led, LOW); /* Desenergiza o pino 7
(desliga o LED). */
  delay(1000); /* Aguarda 1 segundo. */
}
```

**Quadro 03 - Código-fonte da programação do Projeto 2: Ligando três LEDs com três Resistores na linguagem do Arduino (Wiring)**

```
/* Projeto 2: Ligando três LEDs com três Resistores */

int Pin_led1 = 8; /* Define o controle do LED 1 através do
pino 8. */
int Pin_led2 = 9; /* Define o controle do LED 2 através do
pino 9. */
int Pin_led3 = 10; /* Define o controle do LED 3 através do
pino 10. */

void setup() {
  pinMode(Pin_led1, OUTPUT); /* Define o pino de controle do
LED 1 como SAÍDA. */
  pinMode(Pin_led2, OUTPUT); /* Define o pino de controle do
LED 2 como SAÍDA. */
  pinMode(Pin_led3, OUTPUT); /* Define o pino de controle do
LED 3 como SAÍDA. */
}

void loop() {
  digitalWrite(Pin_led1, HIGH); /* Energiza o pino 8 (liga o
LED 1). */
  digitalWrite(Pin_led2, HIGH); /* Energiza o pino 9 (liga o
LED 2). */
  digitalWrite(Pin_led3, HIGH); /* Energiza o pino 10 (liga o
LED 3). */
  delay(1000); /* Aguarda 1 segundo. */
  digitalWrite(Pin_led1, LOW); /* Desenergiza o pino 8
(desliga o LED 1). */
  digitalWrite(Pin_led2, LOW); /* Desenergiza o pino 9
(desliga o LED 2). */
  digitalWrite(Pin_led3, LOW); /* Desenergiza o pino 10
(desliga o LED 3). */
  delay(1000); /* Aguarda 1 segundo. */
}
```

#### Quadro 04 - Código-fonte da programação do Projeto 3: Ligando três LEDs com um Resistor na linguagem do Arduino (Wiring)

```
/* Projeto 3: Ligando três LEDs com um Resistor */

int Pin_led1 = 8; /* Define o controle do LED 1 através do
pino 8. */
int Pin_led2 = 9; /* Define o controle do LED 2 através do
pino 9. */
int Pin_led3 = 10; /* Define o controle do LED 3 através do
pino 10. */

void setup() {
  pinMode(Pin_led1, OUTPUT); /* Define o pino de controle do
LED 1 como SAÍDA. */
  pinMode(Pin_led2, OUTPUT); /* Define o pino de controle do
LED 2 como SAÍDA. */
  pinMode(Pin_led3, OUTPUT); /* Define o pino de controle do
LED 3 como SAÍDA. */
}

void loop() {
  digitalWrite(Pin_led1, HIGH); /* Energiza o pino 8 (liga o
LED 1). */
  delay(200); /* Aguarda 0,2 s. */
  digitalWrite(Pin_led1, LOW); /* Desenergiza o pino 8
(desliga o LED 1). */
  digitalWrite(Pin_led2, HIGH); /* Energiza o pino 9 (liga o
LED 2). */
  delay(200); /* Aguarda 0,2 de segundo. */
  digitalWrite(Pin_led2, LOW); /* Desenergiza o pino 9
(desliga o LED 2). */
  digitalWrite(Pin_led3, HIGH); /* Energiza o pino 10 (liga o
LED 3). */
  delay(200); /* Aguarda 0,2 de segundo. */
  digitalWrite(Pin_led3, LOW); /* Desenergiza o pino 10
(desliga o LED 3). */
  delay(200); /* Aguarda 0,2 de segundo. */
}
```

A seguir, compile o programa pressionando o botão **Verificar** para examinar se não há erros de sintaxe. Estando o código correto, o próximo passo é realizar a transferência do programa para o Arduino. Pressione o botão **Carregar**, para realizar upload do programa para o Arduino.

Após a transferência do programa para o Arduino, o(s) LED(s) piscará(ão) conforme as instruções descritas no código-fonte do projeto selecionado.

### i. Linguagem de programação por blocos

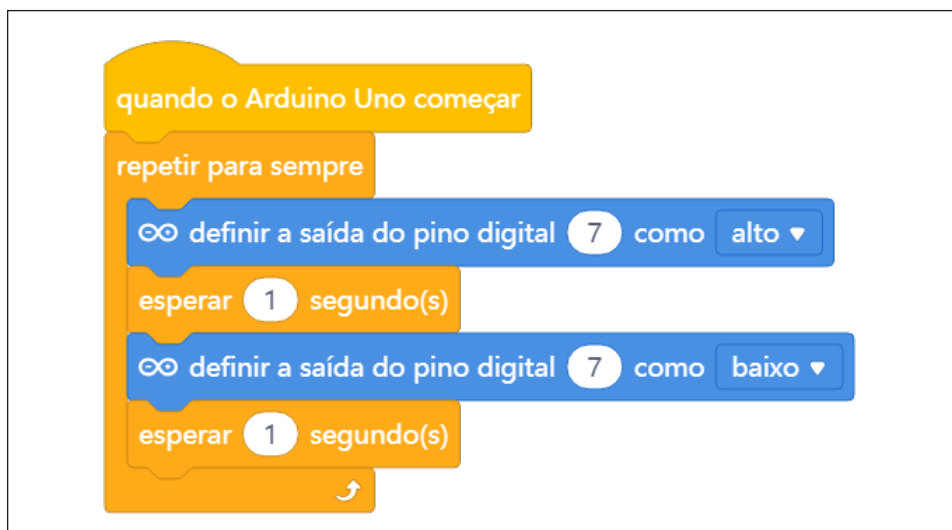
Outra forma de programar o(s) LED(s) é por meio da linguagem de programação que utiliza blocos de funções prontas, os quais representam comandos de programação. Para isso, vamos utilizar o software mBlock.

Para conectar o mBlock ao Arduino, você deve clicar no ícone **Adicionar**, localizado no campo **Dispositivos**, e selecionar o Arduino na biblioteca de dispositivos do mBlock, clicando, na sequência, no botão **OK**.

Uma vez selecionado, o Arduino Uno é visualizado no campo **Dispositivos** do mBlock e já é possível iniciar a programação por blocos.

Monte os blocos, arrastando e soltando, de acordo com a programação de cada projeto, como mostram as figuras 10, 11 e 12 (atente para selecionar a programação correspondente ao projeto que você escolheu e montou na Protoboard).

Figura 10  
Programação em blocos para o Projeto 1: Ligando um LED com Resistor



Fonte: site mBlock oficial

Figura 11 - Programação em blocos para o Projeto 2:  
Ligando três LEDs com três Resistores

```
quando o Arduino Uno começar
repetir para sempre
  definir a saída do pino digital 8 como alto
  definir a saída do pino digital 9 como alto
  definir a saída do pino digital 10 como alto
  esperar 1 segundo(s)
  definir a saída do pino digital 8 como baixo
  definir a saída do pino digital 9 como baixo
  definir a saída do pino digital 10 como baixo
  esperar 1 segundo(s)
```

Fonte: site mBlock oficial

Figura 12 - Programação em blocos para o Projeto 3:  
Ligando três LEDs com um Resistor

```
quando o Arduino Uno começar
repetir para sempre
  definir a saída do pino digital 8 como alto
  esperar 0.2 segundo(s)
  definir a saída do pino digital 8 como baixo
  definir a saída do pino digital 9 como baixo
  esperar 0.2 segundo(s)
  definir a saída do pino digital 9 como alto
  definir a saída do pino digital 10 como alto
  esperar 0.2 segundo(s)
  definir a saída do pino digital 10 como baixo
```

Fonte: site mBlock oficial



Assim que os blocos estiverem montados, clique no botão **Conectar** para iniciar a comunicação entre o software mBlock com a placa de Arduino Uno. Ao clicar sobre o botão **Conectar**, aparecerá um Tooltip solicitando a confirmação da conexão entre os dois dispositivos.

Uma vez realizada a conexão entre os dispositivos, será ativado, na interface do mBlock, o botão **Upload**. Ao clicar neste botão, o software irá verificar se não há erros na estrutura do programa e, então, compilará para enviar o programa à placa Arduino.

Com a transferência do código para o dispositivo Arduino Uno, o(s) LED(s) piscará(ão) conforme as instruções organizadas nas montagens dos blocos para cada projeto.



### Desafio:

i. Que tal programar dois ou mais LEDs para piscarem alternadamente? Assim, quando um LED estiver aceso, os demais deverão estar apagados. Para isso, defina o pinMode de cada LED e adicione os delays e o status pela função digitalWrite.



### E se... ?

i. O LED estiver com as “pernas” cortadas, estando ambas do mesmo tamanho, como identificar sua polaridade? Caso o LED esteja com as hastes do mesmo tamanho, analise a cápsula plástica: o lado que possui a estrutura reta, na borda, é cátodo (polo negativo). O lado que possui a estrutura curva é ânodo (positivo).

ii. Você não tiver a resistência indicada? Utilize, em seu projeto, um Resistor com valor mais alto, nunca mais baixo. No caso do LED, um valor maior influenciará na intensidade da cor do LED.

iii. O projeto não funcionar, se atente a alguns dos possíveis erros:

**1.** Verifique se os Jumpers estão nos pinos certos, se estão na mesma coluna dos terminais dos componentes, fazendo assim as conexões;

**2.** Verifique se os LEDs estão conectados corretamente e não de modo invertido;

**3.** Confira se os Resistores utilizados estão de acordo com o projeto.

**4.** Verifique se a programação está adequada a cada porta digital.

### 3. Feedback e Finalização (15min):

**a.** Confira, compartilhando seu projeto com os demais colegas, se o objetivo foi alcançado.

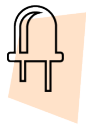
**b.** Analise seu projeto desenvolvido e os que os colegas desenvolveram, de modo a atender aos requisitos para acendimento de LED e utilização de Resistores.

**c.** Reflita se as seguintes situações ocorreram:

**i.** Colaboração e Cooperação: você e os membros de sua equipe interagiram entre si, compartilhando ideias que promoveram a aprendizagem e o desenvolvimento deste projeto?

**ii.** Pensamento Crítico e Resolução de Problemas: você conseguiu identificar os problemas, analisar informações e tomar decisões de modo a contribuir para o projeto desenvolvido?

**d.** Reúna todos os componentes utilizados nesta aula e os organize novamente, junto aos demais, no kit de robótica.



## Videotutorial

Com o intuito de auxiliar na montagem e na programação desta aula, apresentamos um videotutorial, disponível em:



<https://rebrand.ly/a8robotica>

Acesse, também, pelo QRCode:





## Referências

ARDUINO. Site oficial. **Ambiente de Programação do Arduino**. Disponível em: <https://create.arduino.cc/editor>. Acesso em: 15 out. 2021.

ARDUINO. Site oficial. **Downloads**. Disponível em: [www.arduino.cc/en/Main/Software](http://www.arduino.cc/en/Main/Software). Acesso em: 15 out. 2021.

BRASIL. Ministério da Educação. **Base Nacional Comum Curricular**. Brasília, 2018. Disponível em: [http://basenacionalcomum.mec.gov.br/images/BNCC\\_EI\\_EF\\_110518\\_versaofinal\\_site.pdf](http://basenacionalcomum.mec.gov.br/images/BNCC_EI_EF_110518_versaofinal_site.pdf). Acesso em: 08 out. 2021.

MAKEBLOCK. mBlock. **Download mBlock**. Disponível em: <https://mblock.makeblock.com/en-us/download/>. Acesso em: 15 out. 2021.

MAKEBLOCK. mBlock. **Programação em blocos**. Disponível em: <https://ide.mblock.cc/>. Acesso em: 26 out. 2021.

SQUIDS. Arduino. **LEDs** (Tabela). Disponível em: <http://www.squids.com.br/arduino/index.php/hardware/componentes-eletronicos/63-led-5mm>. Acesso em: 10 out. 2021.

TINKERCAD. Autodesk. **Circuitos**. Disponível em: [www.tinkercad.com/things?type=circuits&view\\_mode=default](http://www.tinkercad.com/things?type=circuits&view_mode=default). Acesso em: 30 set. 2021.

DIRETORIA DE TECNOLOGIAS E INOVAÇÃO (DTI)  
COORDENAÇÃO DE TECNOLOGIAS EDUCACIONAIS (CTE)

**EQUIPE ROBÓTICA PARANÁ**

Adilson Carlos Batista  
Cleiton Rosa  
Darice Alessandra Deckmann Zanardini  
Edna do Rocio Becker  
Marcelo Gasparin  
Michelle dos Santos  
Ricardo Hasper  
Simone Sinara de Souza

Os materiais, aulas e projetos da “Robótica Paraná”, foram produzidos pela Coordenação de Tecnologias Educacionais (CTE), da Diretoria de Tecnologia e Inovação (DTI), da Secretaria de Estado da Educação e do Esporte do Paraná (Seed), com o objetivo de subsidiar as práticas docentes com os estudantes por meio da Robótica.

Este material foi produzido para uso didático-pedagógico exclusivo em sala de aula.



Este trabalho está licenciado com uma Licença  
Creative Commons – CC BY-NC-SA  
[Atribuição - NãoComercial - Compartilhavel 4.0](https://creativecommons.org/licenses/by-nc-sa/4.0/)

