

INSTITUTO FEDERAL DE MATO GROSSO DO SUL
CAMPUS CAMPO GRANDE
PROGRAMA PÓS-GRADUAÇÃO EM EDUCAÇÃO PROFISSIONAL E
TECNOLÓGICA

ELETOPRATICANDO

ATIVIDADES PRÁTICAS
INTEGRANDO ELETRODINÂMICA E ARDUINO

MATERIAL DO ESTUDANTE



AUTOR:
HUGO EDUARDO PIMENTEL MOTTA SISCAR

ORIENTADOR:
DANTE ALIGHIERI ALVES DE MELLO

Siscar, Hugo Eduardo Pimentel Motta
S622e Eletropraticando: atividades práticas integrando eletrodinâmica e arduino: material do estudante / Hugo Eduardo Pimentel Motta Siscar, Dante Alighieri Alves de Mello. – Campo Grande-MS, 2022.
34 f. : il. color.

Produto educacional (Mestrado em Educação Profissional e Tecnológica) – Programa de Pós-Graduação em Educação Profissional e Tecnológica, Instituto Federal de Mato Grosso do Sul-IFMS, Campus Campo Grande, 2022.

Orientador: Prof. Dr. Dante Alighieri Alves de Mello.

Inclui referências.

Inclui ISBN 978-65-00-57670-2.

1. Produto educacional. 2. Eletrodinâmica. 3. Arduino. I. Mello, Dante Alighieri Alves de. II. Instituto Federal de Mato Grosso do Sul. Programa de Pós-Graduação em Educação Profissional e Tecnológica. III. Título.

CDD 23. ed. 537.6

APRESENTAÇÃO

Caro(a) estudante,

Este Caderno de Atividades é um material de apoio aos seus estudos.

A concepção deste material surgiu da elaboração de um Produto Educacional final da minha Dissertação de Mestrado, cujo título é: "Proposta de aplicação de atividades práticas on-line integrando Eletrodinâmica e Arduino". Assim, alguns conteúdos aqui apresentados são referentes à Eletrodinâmica e Arduino.

As atividades abordarão conceitos de corrente elétrica, resistores, potenciômetros, Light Dependent Resistor (LDR), Light Emitter Diode (LED), Arduino e linguagem de programação.

Você poderá clicar nos ícones presentes nas atividades para acessar alguns recursos.



Vídeos



Links

SUMÁRIO

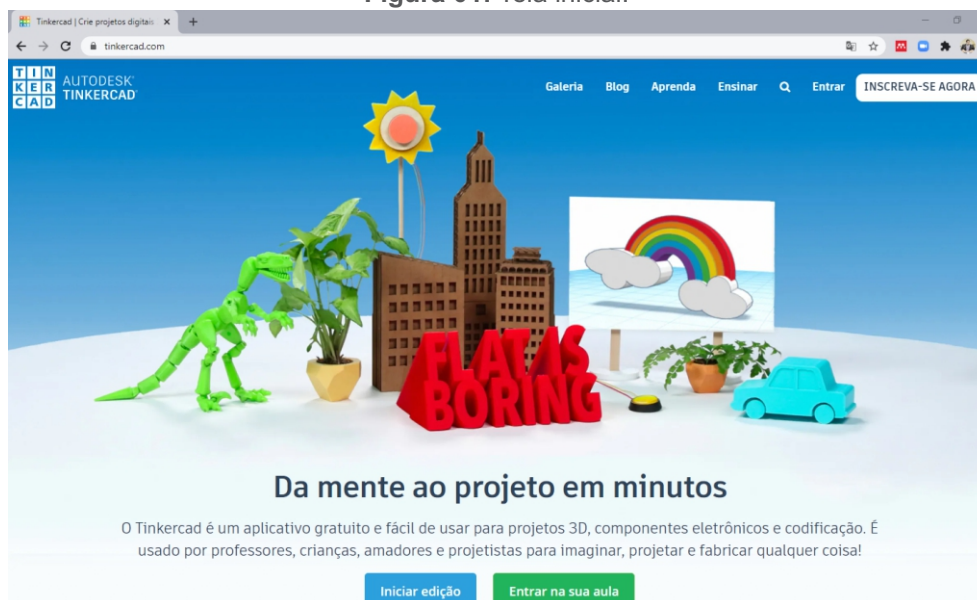
CONHECENDO O AMBIENTE VIRTUAL	04
CORRENTE ELÉTRICA	07
RESISTORES	08
ATIVIDADE 01 - Associação de resistores em série	12
ATIVIDADE 02 - Associação de resistores em paralelo	13
LED - <i>Light Emitter Diode</i>	14
ATIVIDADE 03 - Calculando o valor do resistor para ligar o LED.....	15
ARDUINO	16
ATIVIDADE 04 - Pisca LED com Arduino	24
ATIVIDADE 05 - Semáforo	25
POTENCIÔMETRO.....	27
ATIVIDADE 06 - Potenciômetro	28
ATIVIDADE 07 - Regulando o brilho do LED por meio do Potenciômetro.....	29
LDR - <i>Light Dependent Resistor</i>.....	30
ATIVIDADE 08 - Lendo o valor do LDR	31
ATIVIDADE 09 - Acionando o LED por meio do LDR	32
REFERÊNCIAS	33

CONHECENDO O AMBIENTE VIRTUAL

Acesse o link: <https://www.tinkercad.com/>

Clique em «Entrar»

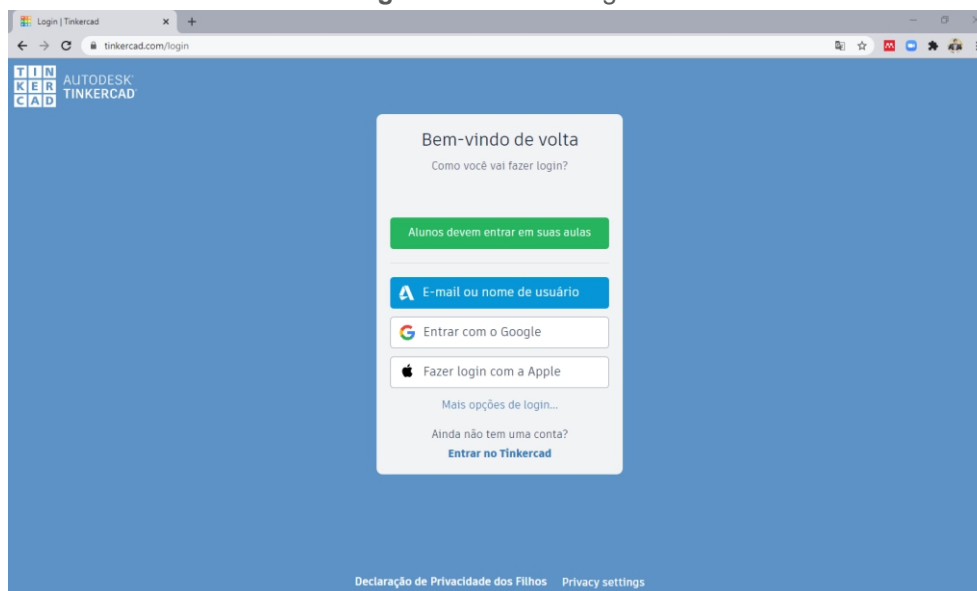
Figura 01: Tela inicial.



Fonte: Tinkercad.

Clique em «Alunos devem entrar em suas aulas»

Figura 02: Tela de login.



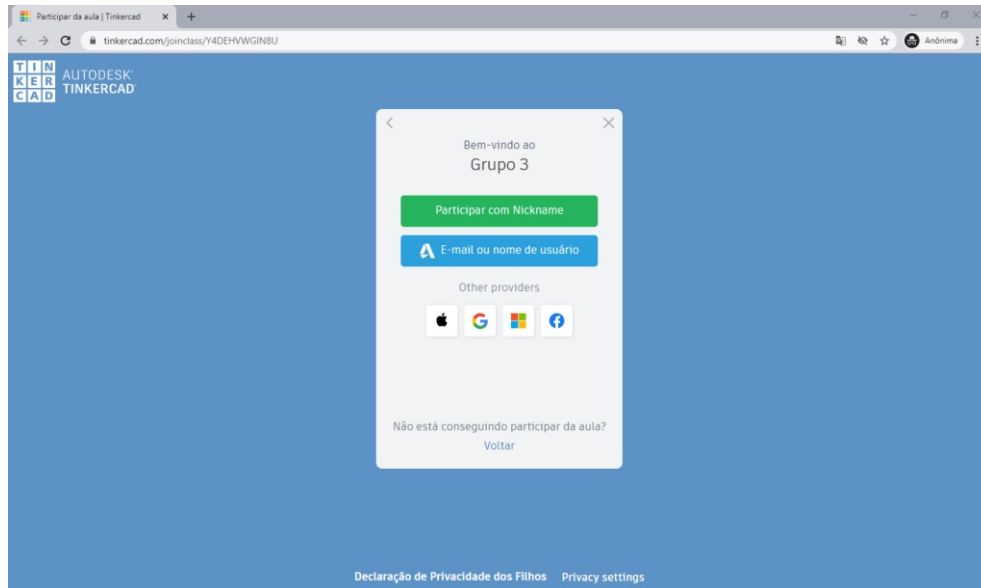
Fonte: Tinkercad.



CONHECENDO O AMBIENTE VIRTUAL

Clique em «Participar com Nickname» Obs.: Disponível pelo professor!

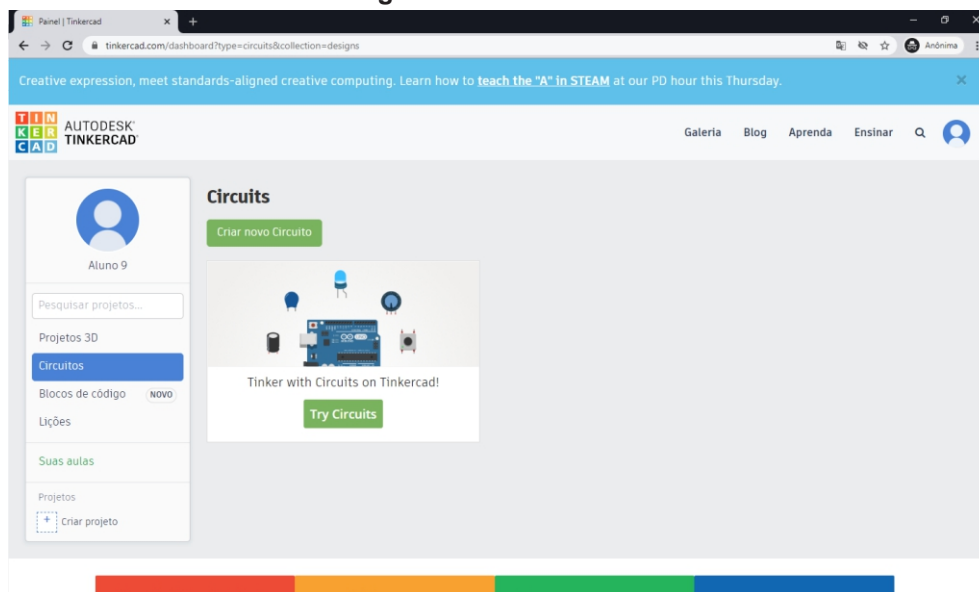
Figura 03: Nickname.



Fonte: Tinkercad.

Clique em «Circuitos» e depois clicar em «Criar novo circuito»

Figura 04: Circuito.

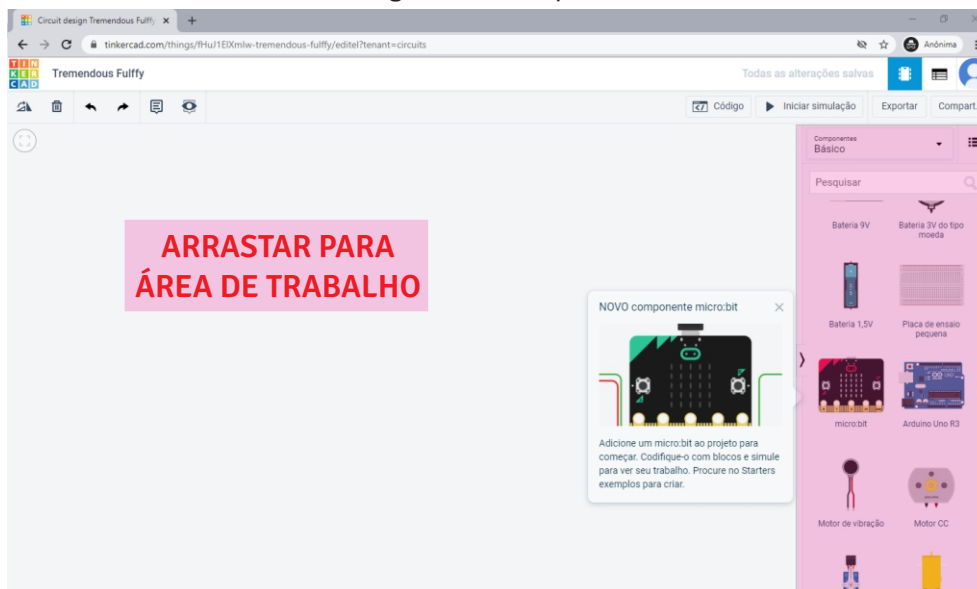


Fonte: Tinkercad.

CONHECENDO O AMBIENTE VIRTUAL

Clique nos componentes e arraste para área de trabalho.

Figura 05: Componentes.



Fonte: Tinkercad (com adaptações).

Clique em «Tremendous Fulffy» e renomeie a atividade. Ex.: «Atividade 01»

Figura 06: Salvar arquivo.



Fonte: Tinkercad (com adaptações).

CORRENTE ELÉTRICA

CONCEITOS

A corrente elétrica é definida como **movimento ordenado de elétrons livres** no interior de um condutor.

Os elétrons livres movimentam-se no interior de um condutor sempre do aumento de potencial, ou seja, do potencial elétrico menor para o maior.

A **intensidade média de corrente elétrica** i_m pode ser obtida pelo quociente entre a quantidade de carga elétrica Δq que atravessa uma seção reta do condutor e o respectivo intervalo de tempo Δt gasto para percorrê-lo.

$$i_m = \frac{\Delta q}{\Delta t}$$

A unidade de medida da intensidade da corrente elétrica é o ampère (A). Como a unidade de medida da quantidade de carga elétrica no Sistema Internacional (SI) é o coulomb (C) e a do tempo o segundo (s), temos:

$$1A = \frac{1C}{1s}$$

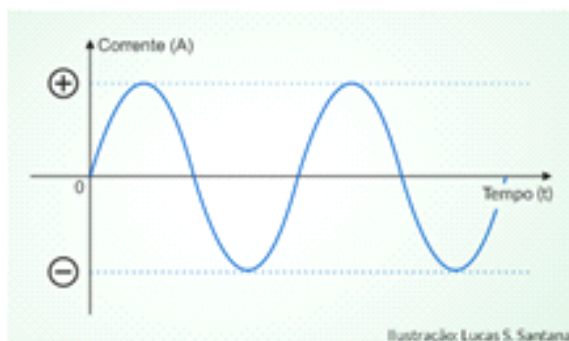


Figura 07 - Corrente alternada

Figura 08 - Corrente contínua



Fonte: <https://blog.bluesol.com.br/corrente-alternada-e-continua-diferencas/>

RESISTORES

CONCEITOS

Resistência elétrica é a capacidade de um objeto opor-se à passagem de corrente elétrica, quando submetido a uma diferença de potencial. A figura abaixo apresenta a simbologia elétrica do resistor no padrão europeu e americano, respectivamente.



Fonte: https://commons.wikimedia.org/wiki/File:Upor_simbol1.png

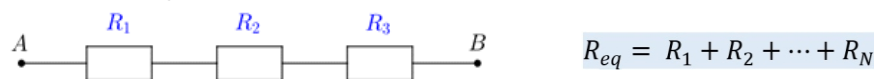
No Sistema Internacional (SI) a unidade de medida da resistência elétrica é o ohm (Ω). Sendo:

$$U = 1V \text{ (volt)} \quad e \quad R = \frac{U}{i}, \text{ temos } R = \frac{1V}{1A} = 1\Omega \text{ (ohm)}$$

$$i = 1A \text{ (ampère)}$$

Na associação em **série** todos os resistores devem ser percorridos pela mesma intensidade de corrente. Entretanto, as ddp's entre os extremos de cada um serão diferentes se os resistores associados forem diferentes. Abaixo, temos n resistores de resistências, respectivamente, R_1, R_2, \dots, R_n , associados em série:

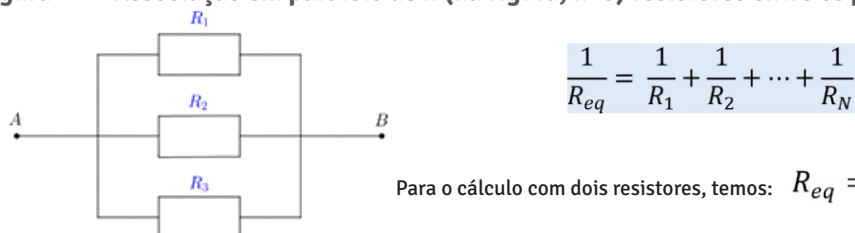
Figura 10 – Associação em série de n (na figura, $n=3$) resistores entre os pontos A e B.



Fonte: <https://noic.com.br/materiais-fisica/cursos/aula-5-6-associacoes-de-resistores/>

Na associação em **paralelo**, todos os resistores devem estar sob a mesma ddp. Entretanto as intensidades de corrente que atravessam cada resistor são diferentes, desde que os resistores associados não sejam iguais.

Figura 11 – Associação em paralelo de n (na figura, $n=3$) resistores entre os pontos A e B.

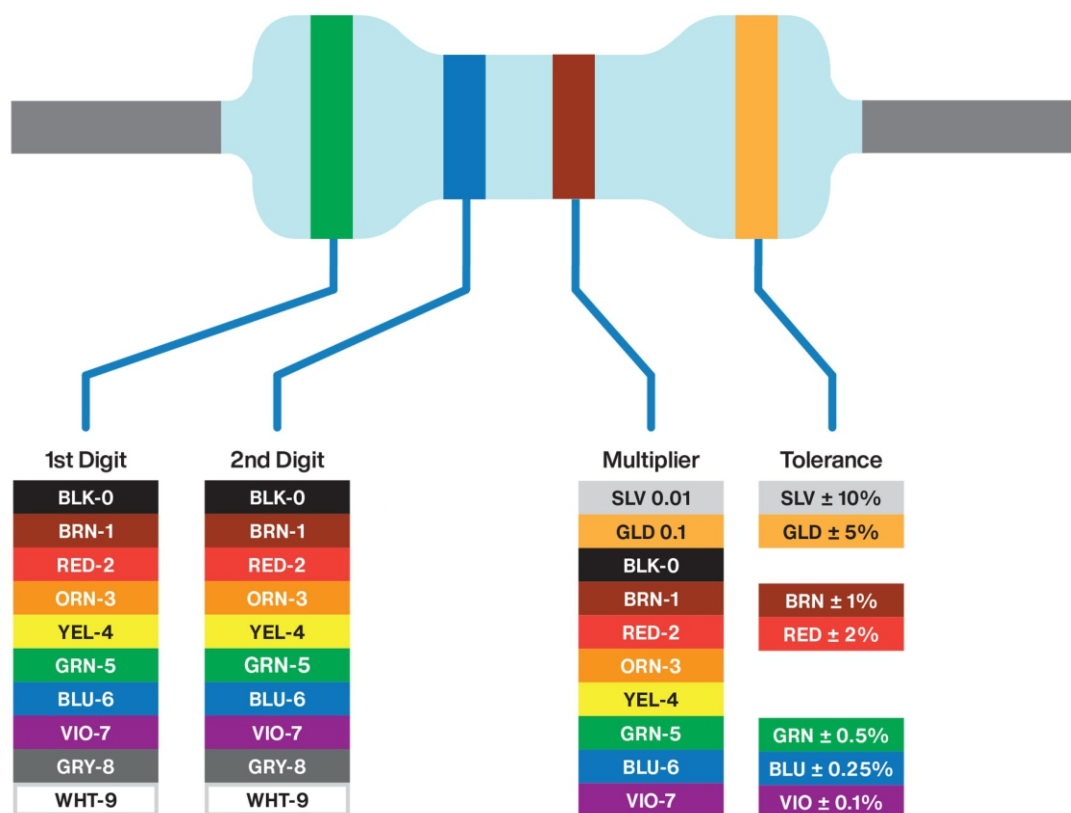


Para o cálculo com dois resistores, temos: $R_{eq} = \frac{(R_1 \cdot R_2)}{(R_1 + R_2)}$

Fonte: <https://noic.com.br/materiais-fisica/cursos/aula-5-6-associacoes-de-resistores/>

RESISTORES

CÓDIGO DE CORES



Fonte: [Adaptado] <https://br.mouser.com/technical-resources/conversion-calculators/resistor-color-code-calculator>

1ª cor: Verde - Primeiro algarismo = 5

2ª cor: Azul - Segundo algarismo = 6

3ª cor: Marrom - Valor do expoente da potência de base $10^1 = 10$

4ª cor: Dourada - Tolerância de 5%

Valor da resistência: 560 Ω .

O valor da resistência pode variar de 532 Ω a 588 Ω .



Calculadora gráfica on-line para código de cores de resistores



ASSOCIAÇÃO DE RESISTORES EM SÉRIE

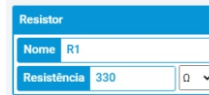
UTILIZANDO SIMULADOR VIRTUAL TINKERCAD



ADICIONAR O(S) RESISTOR(ES)

CLIQUE NO COMPONENTE E ARRASTE PARA ÁREA DE TRABALHO

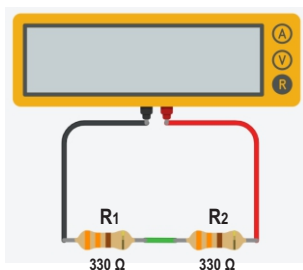
SELECIONAR O VALOR DO(S) RESISTOR(RES)



ADICIONAR O MULTÍMETRO

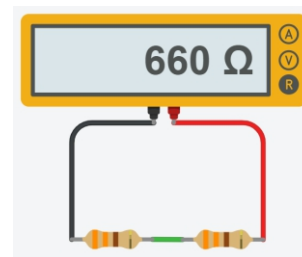
CLIQUE NO COMPONENTE E ARRASTE PARA ÁREA DE TRABALHO

ASSOCIAÇÃO EM SÉRIE



CLIQUE EM

Iniciar simulação



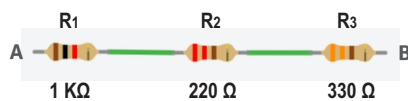
AO SELECIONAR O MULTÍMETRO ESCOLHA A OPÇÃO RESISTÊNCIA

Sabendo que:

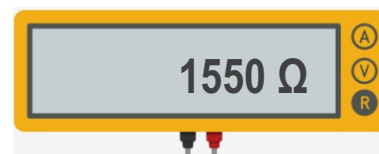
$$R_{eq} = R_1 + R_2 + R_3 + \dots + R_n$$

$$R_{eq} = 330 + 330 = 660 \Omega$$

SIMULE:

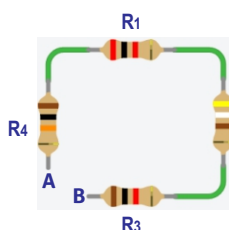


$R_{eq} =$



DETERMINE:

- O valor dos resistores observando o **CÓDIGO DE CORES**;
- O resistor equivalente do circuito.



$$R_1 = 2 \text{ K} \Omega$$

$$R_2 = 490 \Omega$$

$$R_3 = 1 \text{ K} \Omega$$

$$R_4 = 10 \text{ K} \Omega$$

$$R_{eq} = R_1 + R_2 + R_3 + R_4$$

$$R_{eq} = 13490 \Omega \text{ ou } 13,49 \text{ K}\Omega$$

ASSOCIAÇÃO DE RESISTORES EM PARALELO

UTILIZANDO SIMULADOR VIRTUAL TINKERCAD



ADICIONAR O(S) RESISTOR(ES)

CLIQUE NO COMPONENTE E ARRASTE PARA ÁREA DE TRABALHO

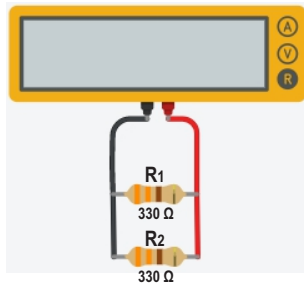
SELECIONAR O VALOR DO(S) RESISTOR(RES)



ADICIONAR O MULTÍMETRO

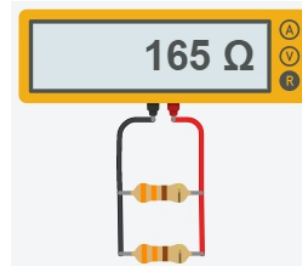
CLIQUE NO COMPONENTE E ARRASTE PARA ÁREA DE TRABALHO

ASSOCIAÇÃO EM PARALELO



CLIQUE EM

Iniciar simulação



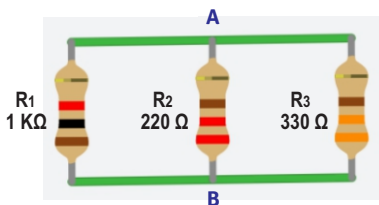
AO SELECIONAR O MULTÍMETRO ESCOLHA A OPÇÃO RESISTÊNCIA

Sabendo que:

$$\frac{1}{R_{eq}} = \frac{1}{R_1} + \frac{1}{R_2} + \dots + \frac{1}{R_n} \text{ ou } R_{eq} = \frac{(R_1 \cdot R_2)}{(R_1 + R_2)}$$

$$R_{eq} = \frac{(330 \cdot 330)}{(330 + 330)} = 165\Omega$$

SIMULE:



$R_{eq} =$



ATIVIDADE 01

ASSOCIAÇÃO DE RESISTORES EM SÉRIE

UTILIZANDO SIMULADOR VIRTUAL TINKERCAD, CALCULE:

1.  $\rightarrow R_{eq} = R_1 + R_2 = 330\Omega + 220\Omega = 550\Omega$

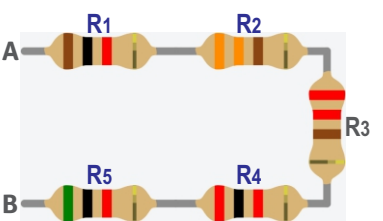
2.  $\rightarrow R_{eq} =$

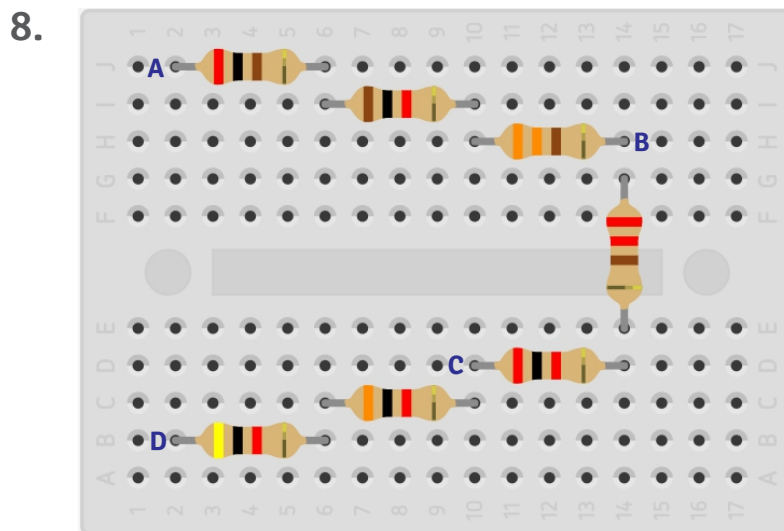
3.  $\rightarrow R_{eq} =$

4.  $\rightarrow R_{eq} =$

5.  $\rightarrow R_{eq} =$

6.  $\rightarrow R_{eq} =$

7.  $\rightarrow R_{eq} =$



Utilizando o recurso,



calcule:

A ~ B: _____

B ~ C: _____

C ~ D: _____

A ~ C: _____

B ~ D: _____

A ~ D: _____

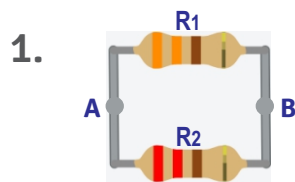


CONSULTAR TABELA
DE CÓDIGO DE CORES

ATIVIDADE 02

ASSOCIAÇÃO DE RESISTORES EM PARALELO

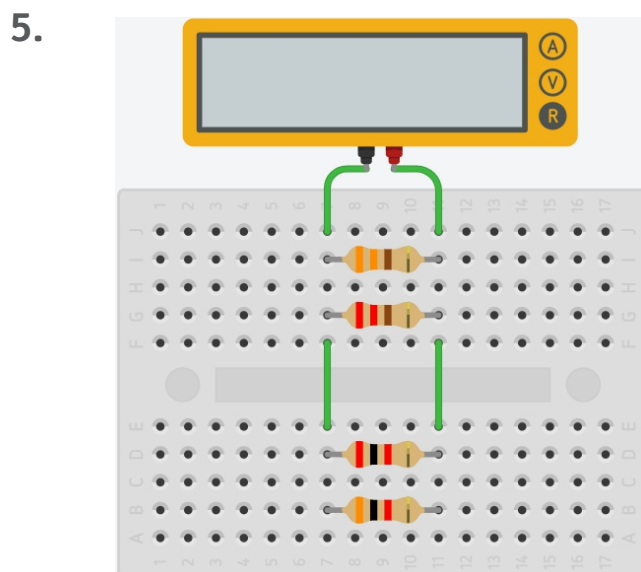
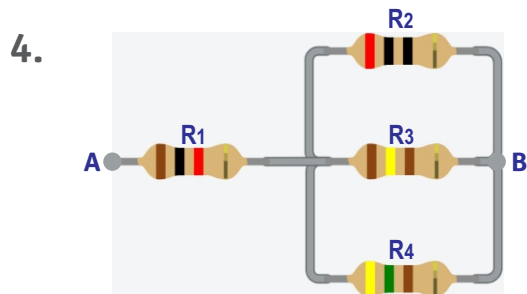
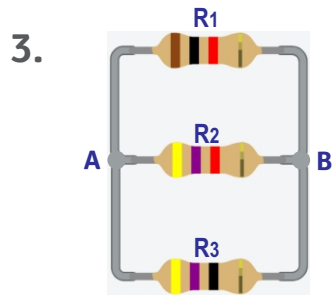
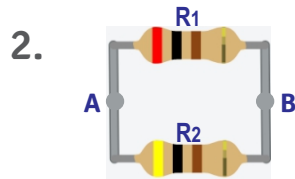
UTILIZANDO SIMULADOR VIRTUAL TINKERCAD



$$\frac{1}{R_{eq}} = \frac{1}{R_1} + \frac{1}{R_2} = \frac{1}{330} + \frac{1}{220} = 132\Omega$$

ou, no caso de cálculo com somente dois resistores:

$$R_{eq} = \frac{(R_1 \cdot R_2)}{(R_1 + R_2)} = \frac{(330 \cdot 220)}{(330 + 220)} = 132\Omega$$

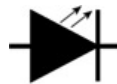


Light Emitter Diode - LED

CONCEITOS

Light Emitter Diode ou, em português, Diodo Emissor de Luz (LED) é um componente eletrônico semicondutor, mesma tecnologia utilizada nos chips dos computadores, que tem a propriedade de transformar energia elétrica em luz (SCOPACASA, 2004, p. 5). A função do diodo nos circuitos eletrônicos é permitir a passagem da corrente elétrica em apenas uma direção. Caso a corrente tente retornar o diodo impede que ela o faça (MCROBERTS, 2011, p.49). A Figura mostra a representação elétrica do LED.

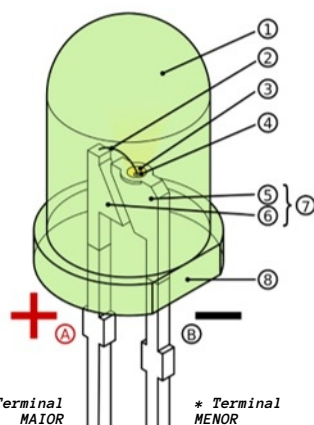
Figura 12 – Simbologia do LED.



Fonte: https://commons.wikimedia.org/wiki/File:Diodo_LED.png

Scopacasa (2004, p. 5) aponta que o LED é um componente do tipo bipolar, ou seja, tem um terminal chamado ânodo e outro chamado cátodo. Dependendo de como for polarizado, permite ou não a passagem da corrente elétrica e, conseqüentemente, a geração ou não de luz. Os componentes que compõem a estrutura do LED são relacionados na figura abaixo.

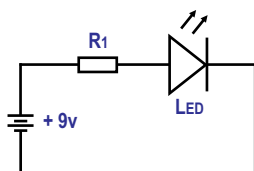
Figura 13 – Componentes de um LED.



Rótulo	Descrição
A	Ânodo
B	Cátodo
1	Lente Epóxi / Encapsulamento
2	Fio condutor
3	Cavidade reflexiva
4	Semicondutor
5	Bigorna
6	Plaquetária
7	Estrutura
8	Borda plana

* Quando ainda não tiveram seus terminais cortados. O terminal negativo também pode ser indicado pelo chanfrado conforme rótulo 8.

Fonte: [https://commons.wikimedia.org/wiki/File:LED,_5mm,_green_\(int\).svg](https://commons.wikimedia.org/wiki/File:LED,_5mm,_green_(int).svg)



$$R = (V_{IN} - V_{LED}) / I_{LED}$$

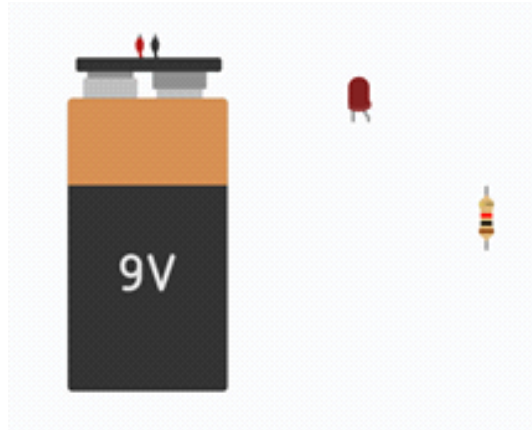
R: RESISTÊNCIA (Ω)
 V_{IN} : TENSÃO BATERIA (V)
 V_{LED} : TENSÃO DO LED (V)
 I_{LED} : CORRENTE DO LED (A)

ATIVIDADE 03

CALCULANDO O VALOR DO RESISTOR PARA LIGAR O LED

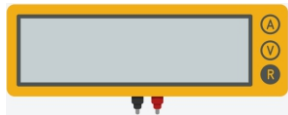
O OBJETIVO DESTA ATIVIDADE É LIGAR O LED.

PARA QUE ISTO ACONTEÇA VOCÊ DEVE DETERMINAR O VALOR DO RESISTOR E INTERLIGAR OS COMPONENTES.



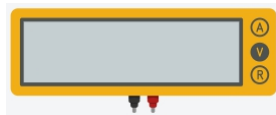
CONSULTAR TABELA
DE CÓDIGO DE CORES

A. QUAL O VALOR DO RESISTOR?



B. A SIMULAÇÃO ACONTECEU CORRETAMENTE NA PRIMEIRA TENTATIVA? CASO A RESPOSTA SEJA NEGATIVA, DESCREVER O QUE FOI FEITO PARA CORRIGIR O PROBLEMA.

C. COM O AUXÍLIO DO MULTÍMETRO DIGITAL, INFORME O VALOR DA TENSÃO NOS TERMINAIS DO RESISTOR.



D. INFORME TAMBÉM O VALOR DA TENSÃO NOS TERMINAIS DO LED.

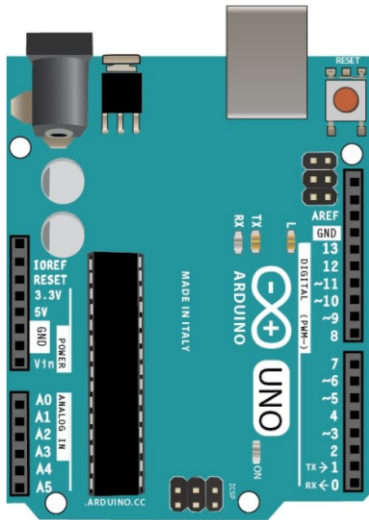


E. TESTE COM RESISTORES DE 10 Ω , 220 Ω e 10 M Ω . DESCREVA O QUE ACONTECE.

10 Ω -
 220 Ω -
 10 M Ω -

ARDUINO

CONCEITOS



Arduino é o que chamamos de uma plataforma de computação física embarcada, ou seja, um sistema que pode interagir com seu ambiente por meio de hardware e software (MCROBERTS, 2011, p. 20).

Fonte: <https://commons.wikimedia.org/wiki/File:ArduinoUNO.png>

MODELOS



Arduino Uno



Arduino Leonardo



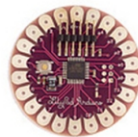
Arduino Ethernet



Arduino Pro



Arduino Mega 2560



Arduino LilyPad



Arduino BT



Arduino Nano



Arduino Mega ADK



Arduino Fio



USB/Serial Light Adapter



Arduino Mini

Fonte: <http://ardufo.blogspot.com/2012/10/modelos-do-arduino.html>

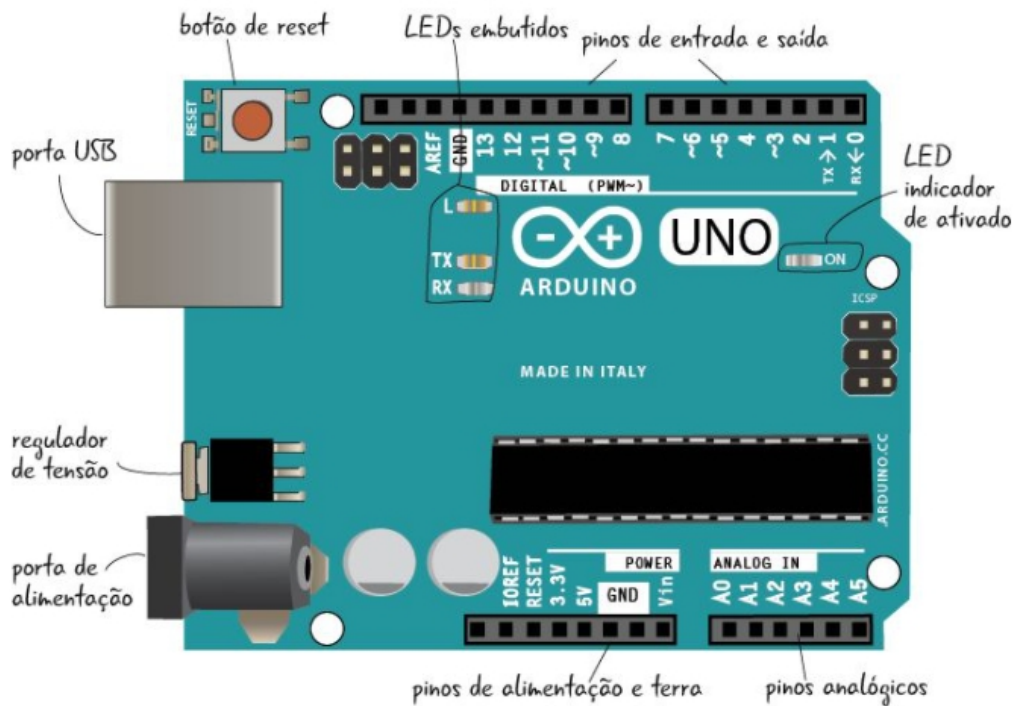
ARDUINO

ESPECIFICAÇÕES TÉCNICAS

Microcontrolador	ATmega328P
Tensão operacional	5V
Tensão de entrada (recomendado)	7-12V
Tensão de entrada (limite)	6-20V
Pinos de E / S digitais	14 (dos quais 6 fornecem saída PWM)
Pinos de E / S digital PWM	6
Pinos de entrada analógica	6
Corrente DC por pino de I / O	40 mA
Corrente DC para pino de 3,3 V	50 mA
Memória flash	32 KB (ATmega328P) dos quais 0,5 KB usados pelo bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Velocidade do relógio	16 Mhz
LED_BUILTIN	13
Comprimento	68,6 mm
Largura	53,4 mm
Peso	25 g

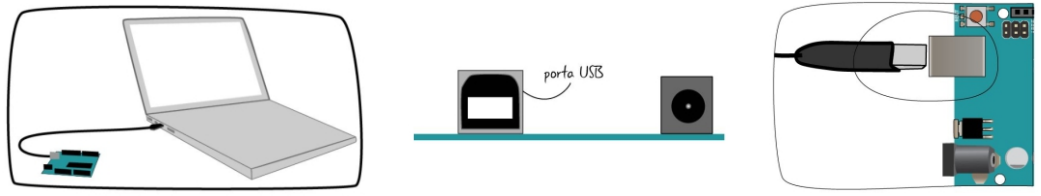
Fonte: <https://datasheet.octopart.com/A000066-Arduino-datasheet-38879526.pdf>

ARDUINO UNO



ARDUINO

ARDUINO - ALIMENTAÇÃO



Fonte: (CULKIN E HAGAN, 2018, p. 32 e 33).



Fonte: (CULKIN E HAGAN, 2018, p. 35 e 36).

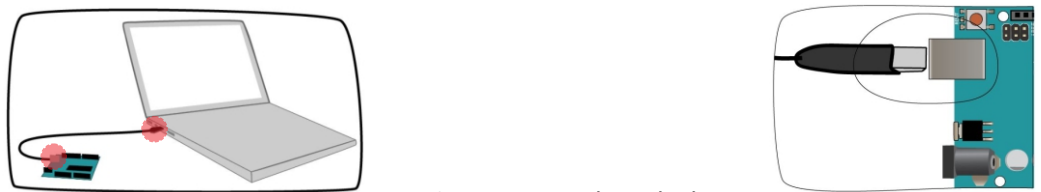


Figura: Forma de onda da corrente contínua e símbolo DC.



Fonte: (CULKIN e HAGAN, 2018, p. 149).



Figura: Forma de onda da corrente alternada e símbolo AC.

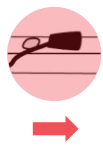
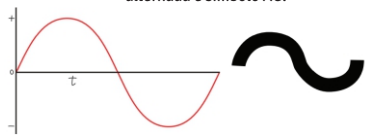


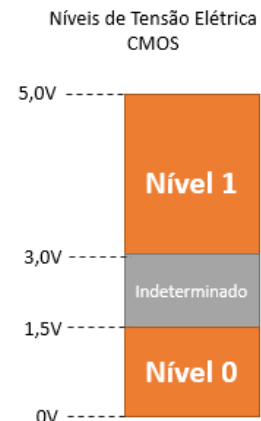
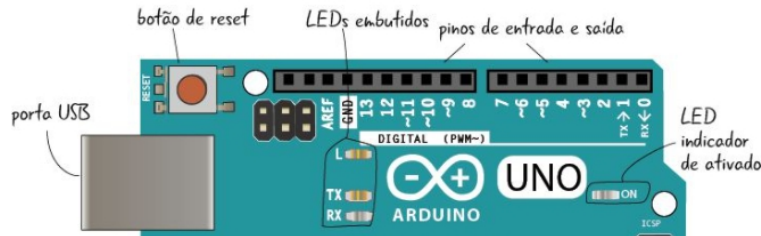
Figura: Forma de onda da corrente contínua e símbolo DC.



Fonte: (CULKIN e HAGAN, 2018, p. 149).

ARDUINO

ARDUINO UNO - ENTRADAS E SAÍDAS DIGITAIS



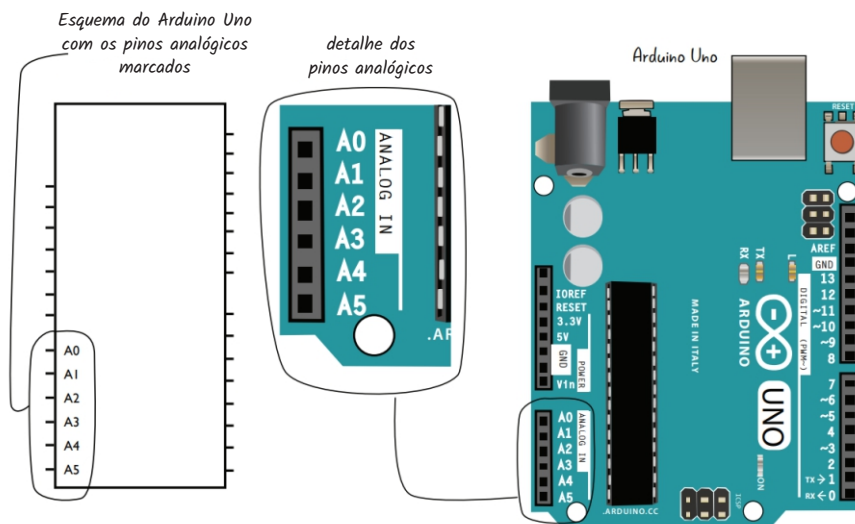
As entradas digitais só podem assumir dois estados:
 Nível lógico **alto** - HIGH: 5 volts.
 Nível lógico **baixo** - LOW: 0 volts.



Fonte:
<https://portal.vidadesilicio.com.br/entrada-s-e-saidas-digitais/>

Fonte:
<http://www.bosontreinamentos.com.br/electronica/electronica-digital/o-que-sao-portas-logicas/>

ARDUINO UNO - PINOS DE ENTRADA ANALÓGICOS



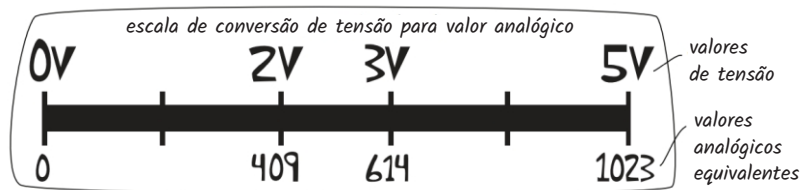
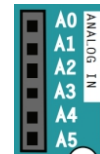
Fonte: (CULKIN E HAGAN, 2018, p. 243).

ARDUINO

ARDUINO UNO - CONVERSÃO ANALÓGICO DIGITAL

CONVERSÃO AD DE 10 Bits

O sinal de entrada é transformado em valores entre 0 e 1023.



000000000 (0) - 0 V
 000000001 (1) - 5 mV
 000000010 (2) - 10 mV

Fonte: (CULKIN E HAGAN, 2018, p. 258).

....

00110011001 (409) - 2 V

....

1001100110 (614) - 3 V

....

1111111111 (1023) - 5 V

Nome da função	O que ela faz	Argumentos da função	Intervalo de valores
digitalRead()	Lê o valor de um pino de entrada digital	O número do pino a ser lido	Lê os valores 1 ou 0 do pino
digitalWrite()	Escreve um valor para um pino de saída	O número do pino a ser escrito e o valor	Escreve 1 ou 0 no pino
analogRead()	Lê o valor de um pino de entrada analógica	O número do pino a ser lido	Lê um número inteiro entre 0 e 1023
analogWrite()	Escreve um valor para um pino de saída com PWM	O número do pino a ser escrito e o valor	Escreve um inteiro entre 0 e 255 no pino, o que resulta em um valor de tensão entre 0 e 5 volts

Fonte: (CULKIN E HAGAN, 2018, p. 263).

ARDUINO

NOÇÕES BÁSICAS DE PROGRAMAÇÃO EM C++

A linguagem do Arduino é baseada em C/C++. Conforme a Documentação de Referência da Linguagem Arduino (ARDUINO.CC, 2021), a linguagem de programação do Arduino pode ser dividida em três partes principais: estruturas, valores (variáveis e constantes) e funções.

ALGUNS ELEMENTOS DE SINTAXE

Quadro – Operadores aritméticos.

Operação	Descrição
+	adição
-	subtração
*	multiplicação
/	divisão
%	resto da divisão

Fonte: Autoria própria.

Operadores aritméticos são usados para desenvolver operações matemáticas. No Quadro 4 apresentamos uma lista resumida de operações aritméticas do C/C++.

Quadro – Operadores de comparação.

Operação	Descrição
==	igual
!=	diferente
<	menor
>	maior
<=	menor igual
>=	maior igual

Fonte: Autoria própria.

Operadores relacionais, ou de comparação, realizam comparações entre as variáveis.

Quadro – Operadores booleanos.

Operação	Descrição
&&	e
	ou
!	não

Fonte: Autoria própria.

ARDUINO

NOÇÕES BÁSICAS DE PROGRAMAÇÃO EM C++

As instruções de controle, ou condicionais, controlam o fluxo do seu código. A instrução *if* (*se*) é um exemplo de uma estrutura de controle, cujo propósito é verificar se determinada condição foi ou não atingida. Caso a condição seja verdadeira, a instrução executa o código dentro de seu bloco de código (MCROBERTS, 2011, p.62). Se utilizarmos a instrução *if* junto com *else* (*senão*), no caso da sentença *if* for falsa, serão executados os comandos entre as chaves do *else*.

```

1  if (condição1) {
2      comando a;
3  }
4  else if (condição2) {
5      comando b;
6  }
7  else {
8      comando c;
9  }
```

Podemos analisar o código acima da seguinte forma: se (*if*) a determinada condição1 for satisfeita, execute os comandos entre chaves (*comando a*), caso contrário se (*else if*), ou seja, se a condição1 não for satisfeita, verifica a condição2. Se a condição2 for satisfeita, executa o comando *b*, caso contrário será executado o comando *c*.

ESTRUTURAS DE REPETIÇÃO

A estrutura de repetição permite que uma sequência de instruções seja repetida quantas vezes o programador desejar.

A função `loop()` já faz algo semelhante.

Os principais comandos de repetição são o `for()` e o `while()`.

ARDUINO

VARIÁVEIS

McRoberts (2011, pg. 42) define uma variável como um local que podemos armazenar dados. Na linguagem C++ o nome da variável deve iniciar com uma letra e o restante pode ser formado por letras, números e underscores (underline).

Quadro – Tipos de variáveis.		
Tipo de variável	Descrição	Exemplo de código
byte	Um número inteiro de 8 bits com um intervalo de 0 a 255.	<code>byte myVariable = 215;</code>
int	Abreviação de número inteiro; um número inteiro de 16 bits com um intervalo 32.767 a -32.768.	<code>int myVariable = 32.000;</code>
long	Um número inteiro logo armazenado como um valor de 32 bits com um intervalo de 2.147.483.647 a -2.147.483.648.	<code>long myVariable = 320.000</code>
float	Um número que tem um ponto decimal, armazenado como um valor de 32 bits com um intervalo de 3,4028325E + 38 a -3,4028325E + 38	<code>float myVariable = 3,14;</code>
unsigned int	Um número inteiro sem quaisquer valores negativos, armazenado como um valor de 16 bits com um intervalo de 0 a 65.545.	<code>unsigned int myVariable = 45.000;</code>
unsigned long	Um número inteiro sem quaisquer valores negativos, armazenado como um valor de 32 bits com um intervalo de 0 a 4.294.967.296.	<code>unsigned long myVariable = 569.124;</code>
word	Um número sem sinal, armazenado como um valor de 16 bits com um intervalo de 0 a 65.545 (o mesmo que um número inteiro sem sinal).	<code>word myVariable = 53.000;</code>
boolean	Uma variável booleana que detém um dos dois valores: verdadeiro ou falso.	<code>boolean myVariable = true;</code>
char	Um único caractere ASCII, armazenado como um número de 8 bits, ou byte. Isso permite que os caracteres possuam funções aritméticas executadas neles. O caractere 'c' é armazenado como o número 99, então você pode adicionar 2 a ele e obter 'e'.	<code>char myVariable = 'c';</code> <code>char myVariable = 99;</code>

Fonte: EVANS, NOBLE e HOCHENBAUM, 2013, p. 398 – 399.

FUNÇÕES

Em linguagens de programação, funções são um conjunto de instruções que podem ser chamadas em qualquer parte do programa. Com esses conjuntos de instruções, a função pode ou retornar um valor ou executar uma determinada tarefa.

O Arduino possui algumas funções predefinidas: `pinMode()`, `digitalWrite()`, `delay()`, entre outras. Também podem ser criadas pelo programador, geralmente antes da função `setup()`.

ATIVIDADE 04

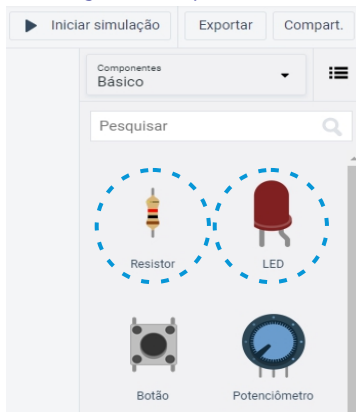
PISCA LED COM ARDUINO

Objetivo:
Acender um **LED** por um segundo e depois apagar por um segundo, repetidamente.

UTILIZANDO SIMULADOR VIRTUAL TINKERCAD

1º
PASSO

Figura 1: Componentes.



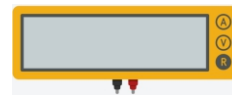
Fonte: Elaborado pelo autor.

ADICIONAR OS COMPONENTES

CLIQUE NOS COMPONENTES E ARRASTE PARA ÁREA DE TRABALHO

- ARDUINO
- LED
- RESISTOR

SABENDO QUE A TENSÃO DE SAÍDA NO PINO 13 DO ARDUINO É DE 5V, DETERMINE O VALOR DO RESISTOR.



2º
PASSO

INTERLIGAR OS COMPONENTES

Figura 2: Layout.

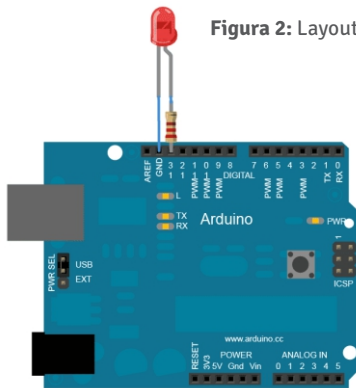
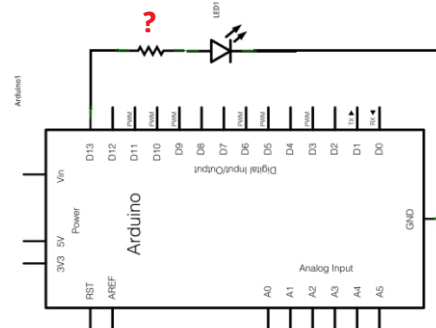


Figura 3: Esquemático.



Fonte: <https://www.arduino.cc/en/Tutorial/BuiltInExamples/Blink>

3º
PASSO

CÓDIGO

CLICAR ➔

Código ▶ Iniciar simulação Exportar Compart.

```
// a função de configuração é executada uma vez quando você pressiona reset ou
liga a placa
void setup () {
  // inicializa o pino digital LED_BUILTIN como uma saída.
  pinMode (LED_BUILTIN, OUTPUT);
}
// a função de loop é executada repetidamente para sempre
void loop () {
  digitalWrite (LED_BUILTIN, HIGH); // liga o LED (ALTO é o nível de tensão)
  delay(1000); // espere um segundo
  digitalWrite (LED_BUILTIN, LOW); // desligue o LED tornando a tensão BAIXA
  delay(1000); // espere um segundo
}
```

4º
PASSO

INICIAR SIMULAÇÃO

AUMENTE/DIMINUA O DELAY E OBSERVE O QUE ACONTECE!

ATIVIDADE 05

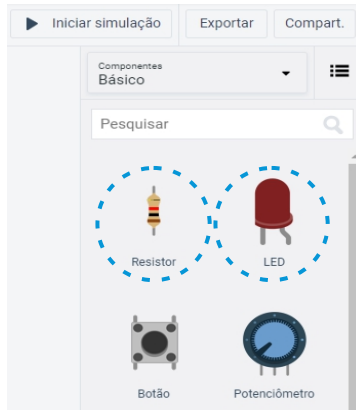
SEMÁFORO

Objetivo:
Simular o funcionamento de um semáforo.

UTILIZANDO SIMULADOR VIRTUAL TINKERCAD

1º
PASSO

Figura 1: Componentes.



Fonte: Elaborado pelo autor.

ADICIONAR OS COMPONENTES

CLIQUE NOS COMPONENTES E ARRASTE PARA ÁREA DE TRABALHO

- 1 ARDUINO
- 3 LED's
- 3 RESISTORES

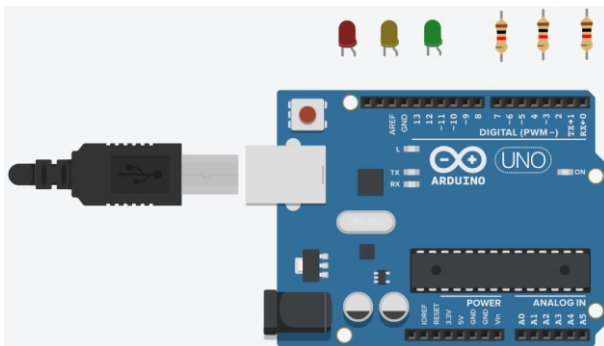
SABENDO QUE A TENSÃO DE SAÍDA NOS PINOS DO ARDUINO É DE 5V, DETERMINE O VALOR DE CADA RESISTOR.



2º
PASSO

INTERLIGAR OS COMPONENTES

Figura 2: Layout.



Fonte: Elaborado pelo autor.

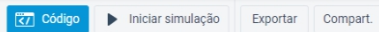
DESENVOLVA O
ESQUEMÁTICO

Enviar no formato de imagem
no Google Chat

3º
PASSO

CODIFICAR

CLICAR →



Para desenvolver o código, considere os seguintes parâmetros:

Semáforo Vermelho: 2 segundos
Semáforo Amarelo: 1 segundo
Semáforo Verde: 2 segundos



4º
PASSO

INICIAR SIMULAÇÃO

Observar o acionamento dos LEDs e os respectivos intervalos de tempo.

5º
PASSO

DESENVOLVA O DIAGRAMA DE TEMPO DA TENSÃO (V) x TEMPO (s)

ATIVIDADE 05 SEMÁFORO

Diagrama de Tempo - Semáforo Vermelho

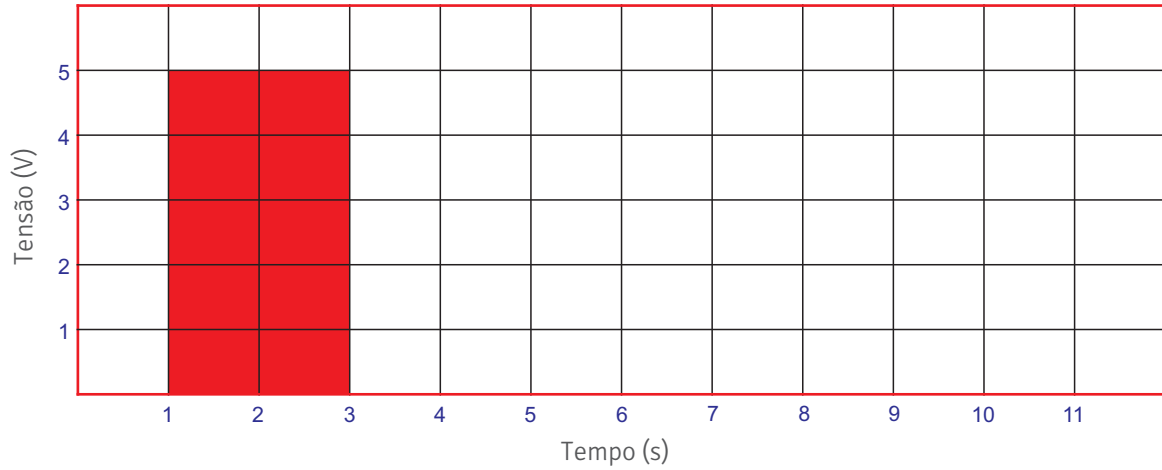


Diagrama de Tempo - Semáforo Amarelo

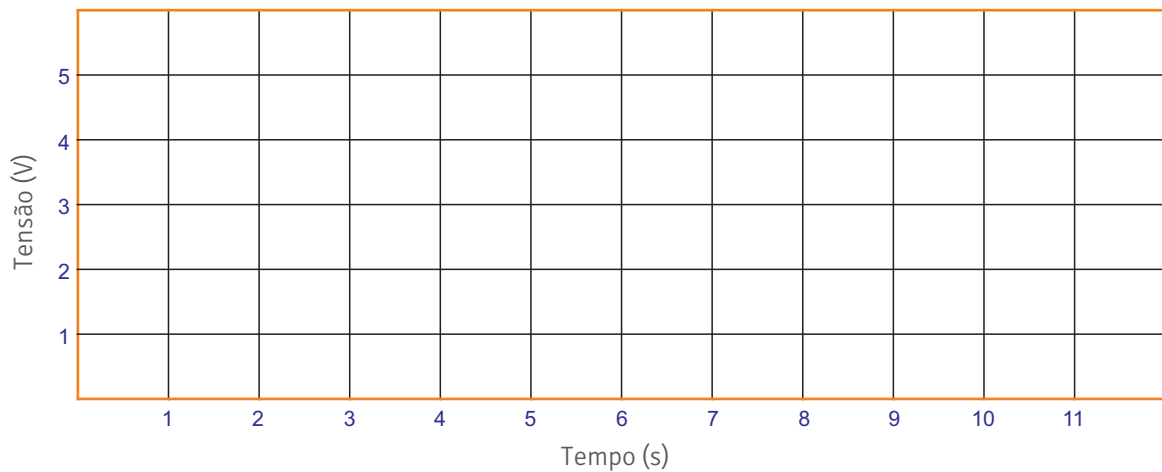
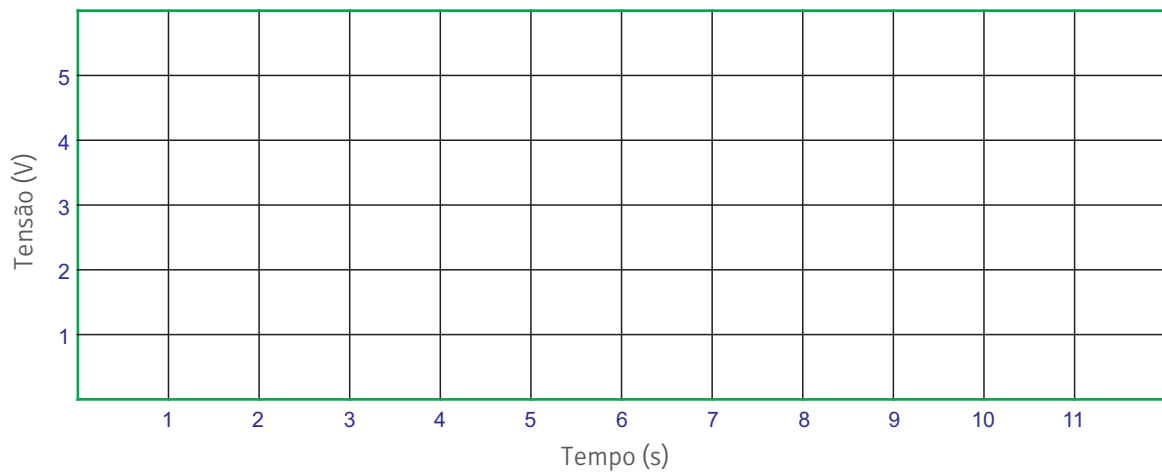


Diagrama de Tempo - Semáforo Verde

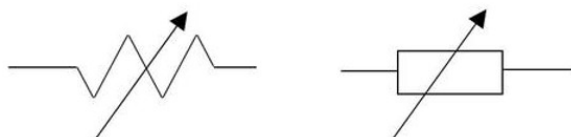


POTENCIÔMETRO

CONCEITOS

Potenciômetros, ou pots, permitem que você ajuste a resistência continuamente. Pots são dispositivos de três terminais, o que significa que proporcionam três locais para se conectar ao mundo externo. Entre os terminais externos existe uma resistência fixa - o valor máximo do pot. A figura abaixo apresenta a simbologia do potenciômetro no padrão americano e europeu.

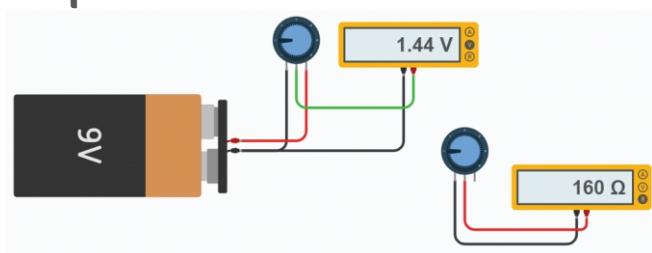
Figura 14 – Símbolo do potenciômetro.



Fonte: https://commons.wikimedia.org/wiki/File:Potenciometre_simbol.jpg

Entre o terminal central e a extremidade de qualquer terminal final, a quantidade de resistência varia dependendo da posição de um eixo rotativo ou outro mecanismo de controle na parte externa do potenciômetro.

Pratique!



Qual o valor da intensidade de corrente elétrica do circuito?

Posicione o potenciômetro de acordo com os valores abaixo e informe a tensão correspondente.

160 Ω - 1.44V

500 Ω -

840 Ω -

ATIVIDADE 06

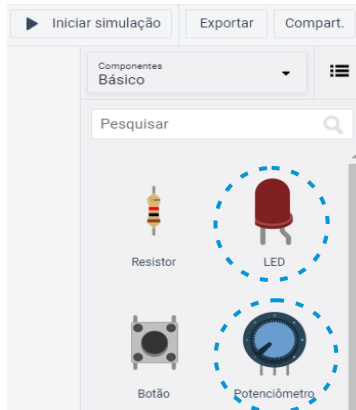
POTENCIÔMETRO

Objetivo:
Realizar leitura do potenciômetro por meio da porta analógica do Arduino.

UTILIZANDO SIMULADOR VIRTUAL TINKERCAD

1º
PASSO

Figura 1: Componentes.



Fonte: Elaborado pelo autor.

ADICIONAR OS COMPONENTES

CLIQUE NOS COMPONENTES E ARRASTE PARA ÁREA DE TRABALHO

- ARDUINO
- POTENCIÔMETRO 10 KΩ

2º
PASSO

INTERLIGAR OS COMPONENTES

Figura 2: Layout.

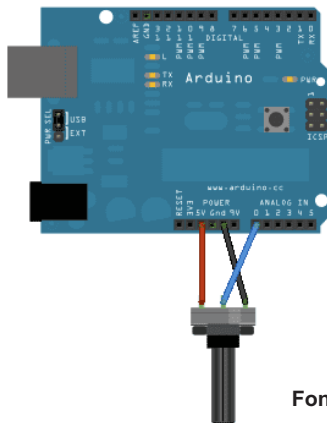
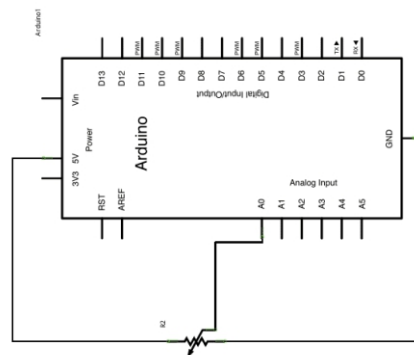


Figura 3: Esquemático.



Fonte: <https://www.arduino.cc/en/Tutorial/BuiltInExamples/AnalogReadSerial>

3º
PASSO

CÓDIGO

CLICAR →

Código ▶ Iniciar simulação Exportar Compart.

```
// a rotina de configuração é executada uma vez quando você pressiona reset:
void setup() {
  // initialize serial communication at 9600 bits per second:
  Serial.begin(9600);
}
//a rotina de loop é executada indefinidamente:
void loop() {
  // leia a entrada no pino analógico 0:
  int sensorValue = analogRead(A0);
  // imprime o valor que você leu:
  Serial.println(sensorValue);
  delay(1); // atraso entre as leituras para estabilidade
}

```

4º
PASSO

INICIAR SIMULAÇÃO

VERIFICAR SERIAL MONITOR

ATIVIDADE 07

REGULANDO O BRILHO DO LED POR MEIO DO POTENCIÔMETRO

Objetivo:
Realizar o controle de brilho do LED por meio da variação da potenciômetro.

UTILIZANDO SIMULADOR VIRTUAL TINKERCAD

1º
PASSO



Fonte: Elaborado pelo autor.

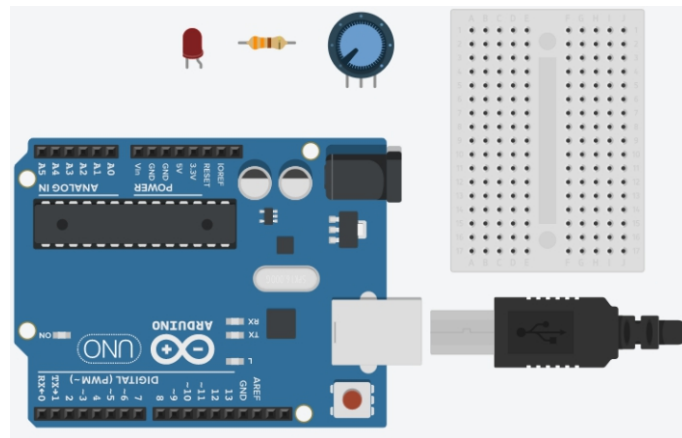
ADICIONAR OS COMPONENTES

CLIQUE NOS COMPONENTES E ARRASTE PARA ÁREA DE TRABALHO

- ARDUINO
- LED
- RESISTOR
- POTENCIÔMETRO 10 K Ω
- PROTOBOARD (Placa de Testes)

2º
PASSO

INTERLIGAR OS COMPONENTES



3º
PASSO

CODIFICAR

CLICAR → [Código](#) ▶ Iniciar simulação ▶ Exportar ▶ Compart.

4º
PASSO

INICIAR SIMULAÇÃO

Variar o potenciômetro e observar o que acontece com o LED.

Light Dependent Resistor - LDR

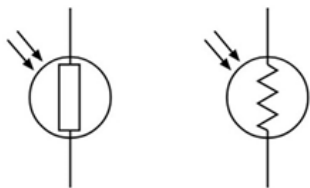
CONCEITOS

São dispositivos cuja resistência varia com a intensidade da luz que incide sobre eles.



Fonte:
<https://commons.wikimedia.org/wiki/File:LDR-gs-2012.jpg>

A resistência diminui com o aumento da intensidade da luz.



Fonte:
https://www.usinainfo.com.br/resistor/ldr-sensor-de-luminosidade-5mm-2982.html?search_query=ldr&results=19

Conforme a incidência de luz diminui sobre o LDR, a resistência aumenta.

ATIVIDADE 08

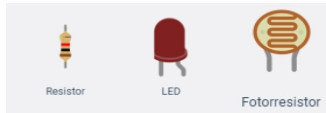
LENDO O VALOR DE UM LDR

Objetivo:
Realizar leitura do potenciômetro por meio da porta analógica do Arduino.

UTILIZANDO SIMULADOR VIRTUAL TINKERCAD

1º
PASSO

Figura 1: Componentes.



Fonte: Elaborado pelo autor.

ADICIONAR OS COMPONENTES

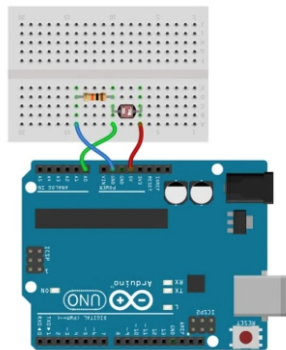
CLIQUE NOS COMPONENTES E ARRASTE PARA ÁREA DE TRABALHO

- ARDUINO
- RESISTOR 10 KΩ
- LDR ou Fotoresistor

2º
PASSO

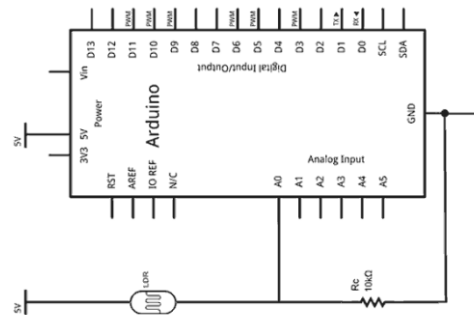
INTERLIGAR OS COMPONENTES

Figura 2: Layout.



Fonte:
<https://create.arduino.cc/projecthub/krivanja/working-with-light-dependent-resistor-ldr-1ded4f>

Figura 3: Esquemático.



Fonte: <https://www.luisllamas.es/medir-nivel-luz-con-arduino-y-fotoresistencia-ldr/>

3º
PASSO

CÓDIGO

CLICAR → Código ▶ Iniciar simulação Exportar Compart.

```
const int LDR = A0;
int input_val = 0;

void setup()
{
  Serial.begin(9600);
}

void loop()
{
  input_val = analogRead(LDR);
  Serial.print("LDR Value is: ");
  Serial.println(input_val);
  delay(1000);
}
```

4º
PASSO

INICIAR SIMULAÇÃO

Observar o que acontece no Serial Monitor quando muda a incidência de luz no ambiente.

ATIVIDADE 09

ACIONANDO UM LED POR MEIO DO LDR

Objetivo:
Ligar / Desligar o LED conforme a intensidade luminosa do ambiente.

UTILIZANDO SIMULADOR VIRTUAL TINKERCAD

1º
PASSO

Figura 1: Componentes.



Fonte: Elaborado pelo autor.

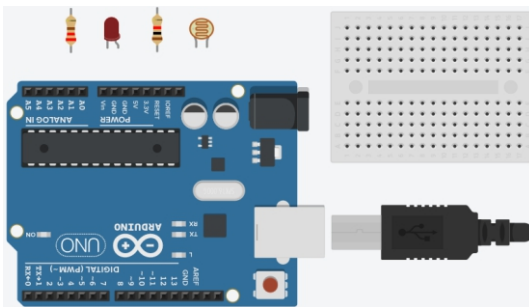
ADICIONAR OS COMPONENTES

CLIQUE NOS COMPONENTES E ARRASTE PARA ÁREA DE TRABALHO

- ARDUINO
- LED
- 2 RESISTORES
- LDR
- PROTOBOARD (Placa de Testes)

2º
PASSO

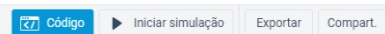
INTERLIGAR OS COMPONENTES



3º
PASSO

CODIFICAR

CLICAR →



Para desenvolver o código, considere os seguintes parâmetros:

*Led ligado quando houver pouca luz no ambiente;
Led desligado quando houver muita luz no ambiente.*

4º
PASSO

INICIAR SIMULAÇÃO

Observar o que acontece no Serial Monitor e no comportamento do LED quando muda a incidência de luz no ambiente.

REFERÊNCIAS

Arduino, 2021. <https://www.arduino.cc/en/Guide/Introduction>

CULKIN, J.; HAGAN, E. **Aprenda eletrônica com Arduino: um guia ilustrado de eletrônica para iniciantes**. São Paulo: Novatec Editora, 2018.

EVANS, M.; NOBLE, J.; HOCHENBAUM, J. **Arduino em Ação**. São Paulo: Novatec Editora, 2013.

MCROBERTS, M. **Arduino Básico**. 1 ed. São Paulo: Novatec, 2011.

SCOPACASA, V. A. Introdução à tecnologia de LED. Revista Lume Arquitetura LA PRO, São Paulo. ed. 01. p. 5-11, nov. 2004.