



Minicurso de Arduino

Aula 02



UNIVERSIDADE
FEDERAL DO CEARÁ
CAMPUS QUIXADÁ



Lei de Ohm

- A Lei de Ohm descreve a relação entre **tensão (V)**, **corrente (I)** e **resistência (R)** em um circuito elétrico.
- A fórmula básica da Lei de Ohm é:

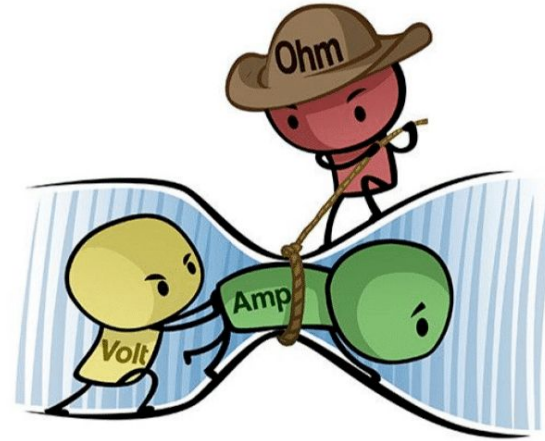
$$V = I \times R$$

- **V** é a tensão, medida em volts (V).
- **I** é a corrente, medida em amperes (A).
- **R** é a resistência, medida em ohms (Ω).



Lei de Ohm

- TENSÃO: Diferença de potencial entre dois pontos.
- CORRENTE: É o fluxo de elétrons em um condutor quando submetido a uma diferença de potencial.
- RESISTÊNCIA: É a obstrução do fluxo de elétrons.



Lei de Ohm

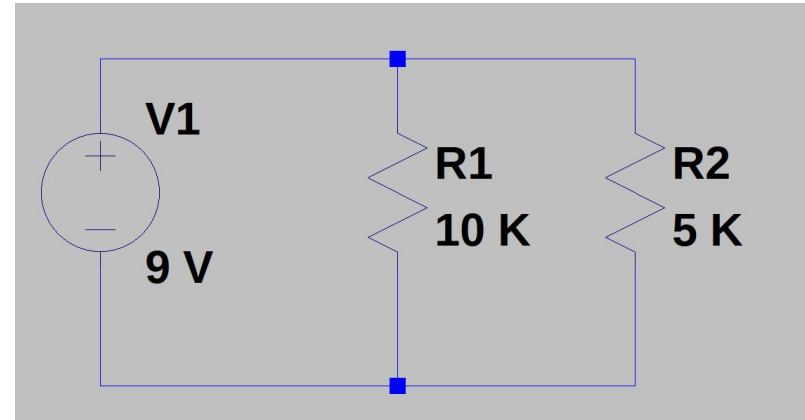
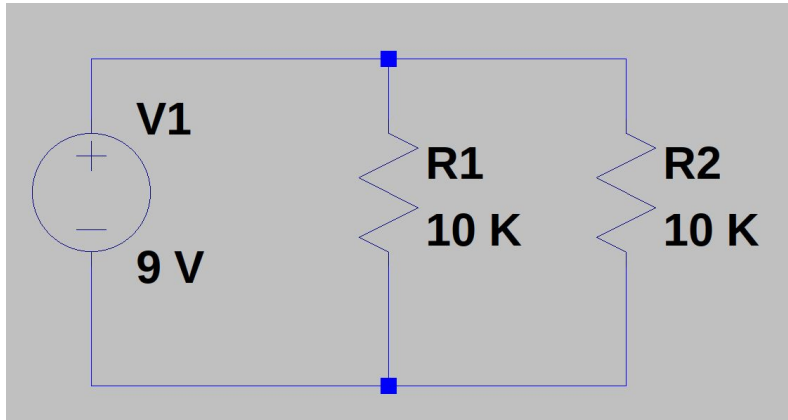


“A corrente que flui por um resistor é proporcional à tensão aplicada e inversamente proporcional ao valor de sua resistência”.

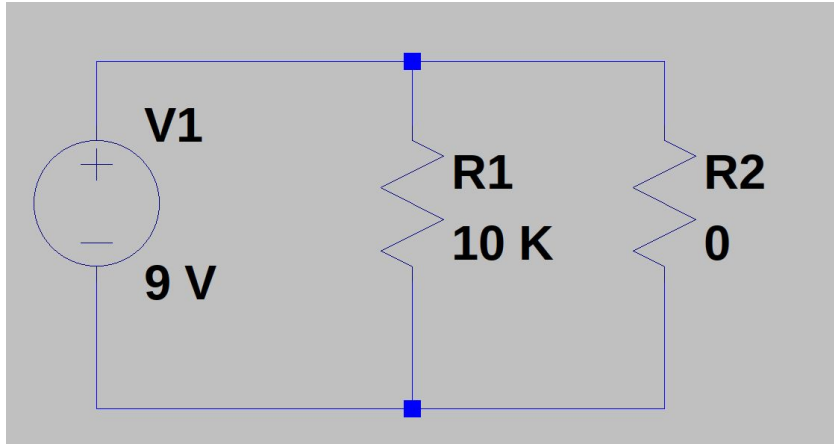
$$I = V/R$$



Curto Circuito



Curto Circuito





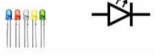


















Curto Circuito

- Curto-circuito é quando a corrente passa diretamente entre dois pontos de potencial diferente sem resistência suficiente, causando superaquecimento e riscos de dano.
- Queimar um LED geralmente ocorre quando ele é submetido a uma corrente ou tensão maior do que ele pode suportar.



Componentes Básicos

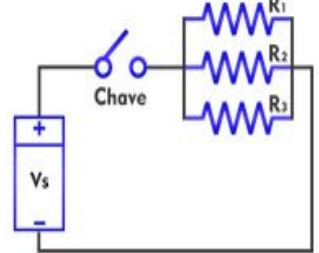
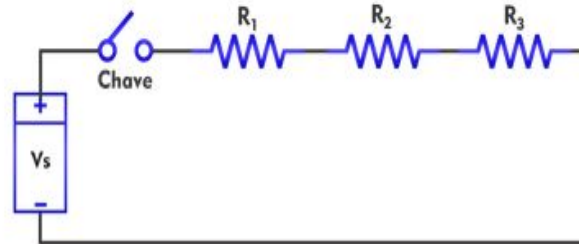
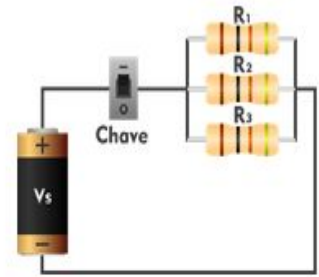
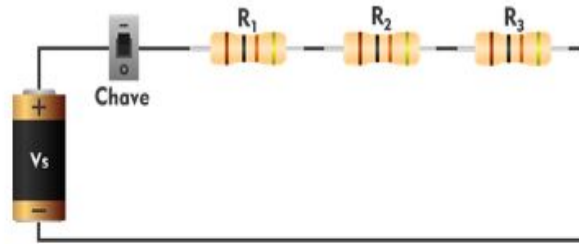
Resistor 	Lâmpada 	Varistor 
Capacitor Eletrolítico 	LED 	Termistor 
Capacitor 	Zener 	Diodo 
Fonte AC 	Alto-Falante 	Indutor 
Fonte DC 	Potenciometro 	Motor 
Transformador 	Diac 	Interruptor 
Fusível 	Triac 	Transistor NPN PNP 



Circuitos Série e Paralelo

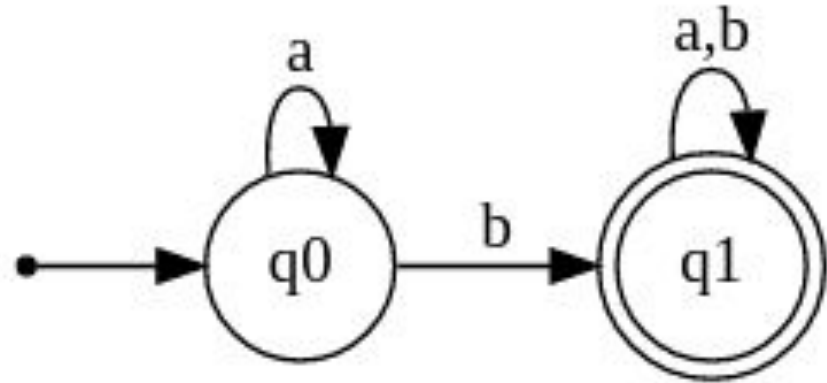
$$R_{eq} = R_1 + R_2 + R_3 + \dots + R_n$$

$$\frac{1}{R_{eq}} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} + \dots + \frac{1}{R_n}$$



Autômatos

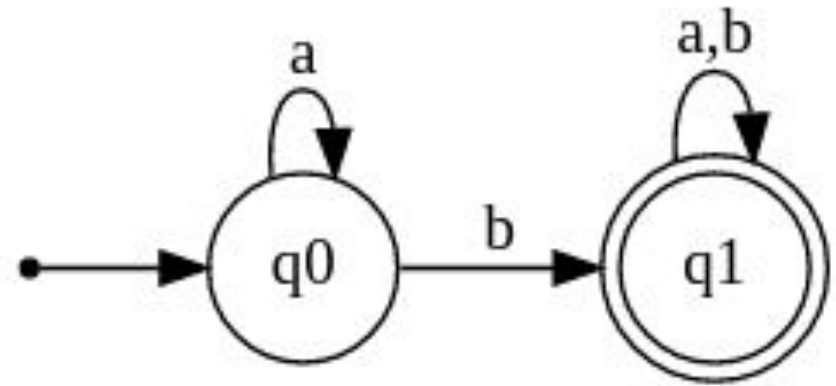
- Autômatos Finitos:
 - Estados
 - Transições
 - Entrada



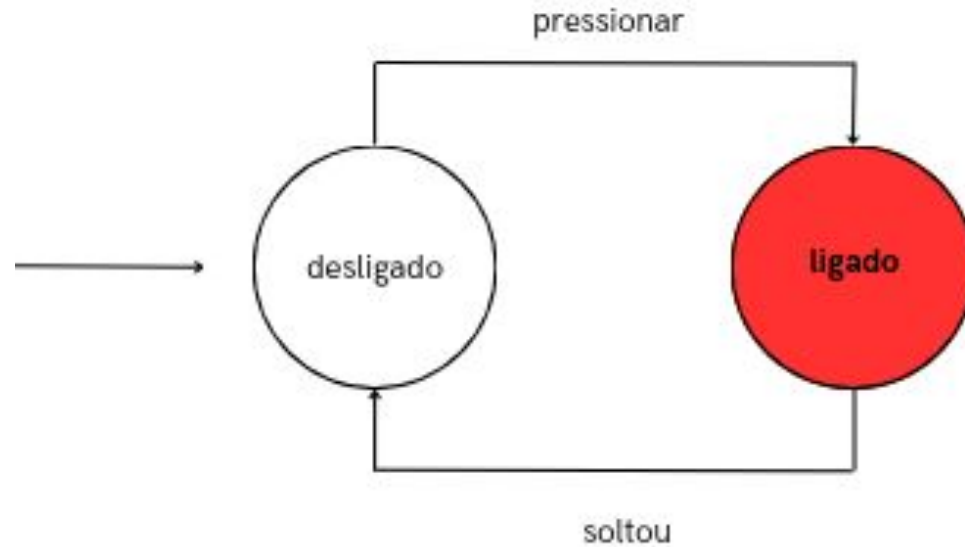
Autômatos

$$A = (Q, \Sigma, \delta, q_0, F)$$

1. Q = é um conjunto finito de símbolos de estado.
2. Σ = é um conjunto finito de símbolos de alfabeto.
3. $\delta: Q \times \Sigma \rightarrow Q$ = é a função de transição.
4. $q_0 \in Q$ = é o estado inicial.
5. $F \subseteq Q$ = é o conjunto de estados de aceitação.

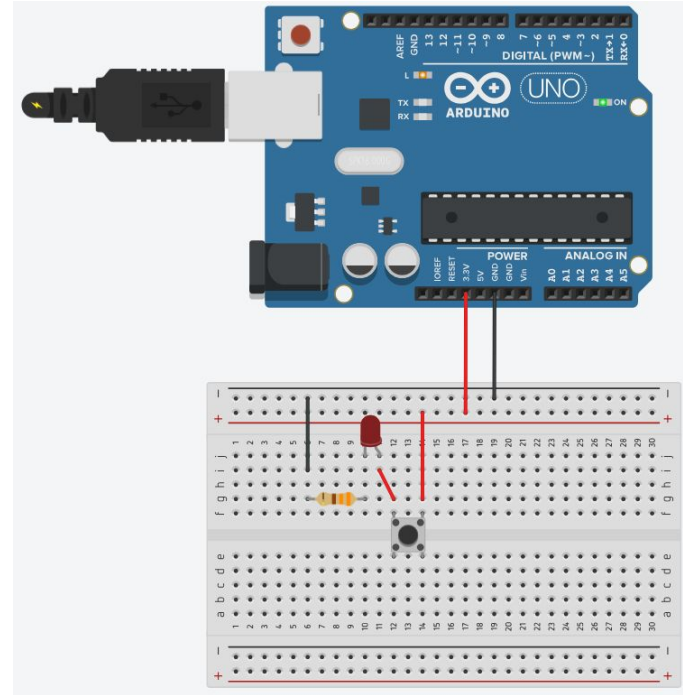


Autômatos



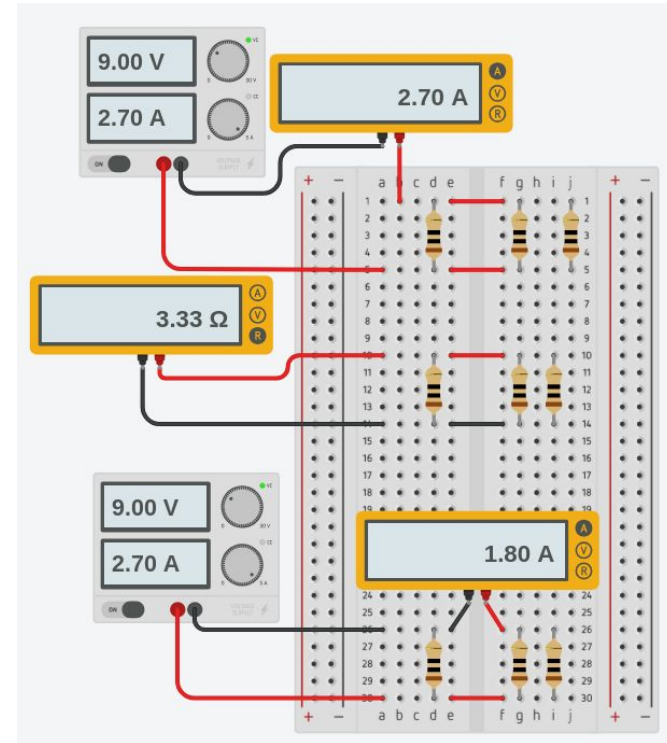
Prática Tinkercad

- Componentes:
 - Led
 - Resistor
 - Botão



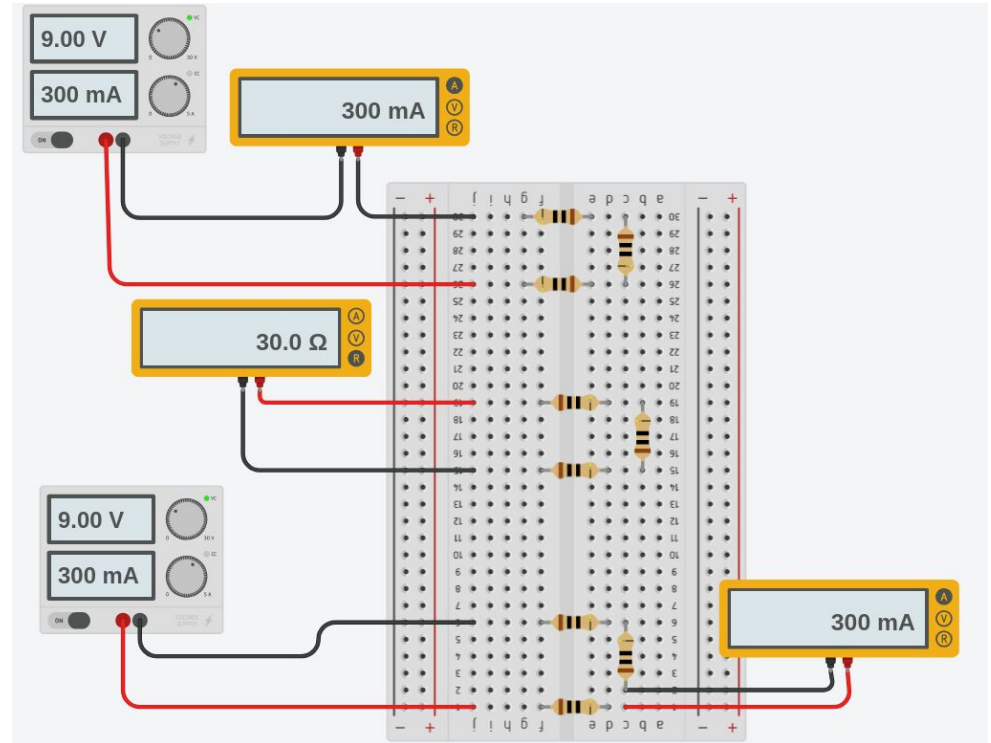
Prática

- Componentes:
 - Fonte
 - Multímetro
 - Resistor



Prática

- Componentes:
 - Fonte
 - Multímetro
 - Resistor



Revisão de FUP

- Constantes:
 - **const int x = 10;**
 - **#define x 100**
 - **True/False.**
 - **HIGH/LOW.**
 - **INPUT/OUTPUT**
- Comentário:
 - **// Este é um comentário de linha**
 - **/*Este é um comentário permite mais de uma linha */**



Revisão de FUP

- Tipo de Variáveis:
 - As variáveis são lugares na memória principal que servem para armazenar dados.
 - São acessadas por meio de um identificador único.
 - O seu valor pode ser alterado ao longo da execução do programa.
 - A variável só pode armazenar um valor a cada instante.
 - Obedecendo a regra: o primeiro caractere do nome de uma variável deve, obrigatoriamente, ser uma letra e não pode ter caracteres especiais ou palavras reservadas.

Tipo de dados	RAM	Intervalo numérico
void keyword	N/A	N/A
boolean	1 byte	0 a 1 (false ou true)
byte	1 byte	0 a 255
char	1 byte	-128 a 127
unsigned char	1 byte	0 a 255
int	2 bytes	-32.768 a 32.767
unsigned int	2 bytes	0 a 65.535
word	2 bytes	0 a 65.535
long	4 bytes	-2.147.483.648 a 2.147.483.647
unsigned long	4 bytes	0 a 4.294.967.295
float	4 bytes	-3,4028235E+38 a 3,4028235E+38
double	4 bytes	-3,4028235E+38 a 3,4028235E+38
string	1 byte + x	Sequência de caracteres
array	1 byte + x	Coleção de variáveis



Revisão de FUP

- Tipo de operadores:
 - Aritméticos:

Símbolo	Significados
-	Subtração
+	Adição
*	Multiplicação
/	Divisão
%	Resto da divisão (módulo)

- Relacionais:

Operação	Operador	Exemplo
Igualdade	==	$x == y$
Diferença	!=	$x != y$
Maior	>	$x > y$
Menor	<	$x < y$
Maior ou igual	>=	$x >= y$
Menor ou igual	<=	$x <= y$



Revisão de FUP

- Tipo de operadores:
 - Lógicos:
 - Compostos:

Operação	Operador
E	&
Ou	
Negação	!
Ou Exclusivo	^

- ++ (incremento)
- -- (decremento)
- += (adição com atribuição)
- -= (subtração com atribuição)
- *= (multiplicação com atribuição)
- /= (divisão com atribuição)

