

# **LOM3258 Introdução à Eletrônica e Computação Física**

**Encontro inicial: Introdução à disciplina**

**Motivação**

**Trilha formativa**

**Perspectiva**

# Motivação

- Década 2010: “Boom” da eletrônica embarcada de baixo custo
- Arduino: projeto aberto de hardware/software
- Definição de Computação Física
- Final da década: Internet das Coisas (IoT)
- Computação na Nuvem (Cloud computing)
- Rápidas transformações aceleradas pela internet
- Pandemia covid-19: Transformação digital

# Trilha formativa

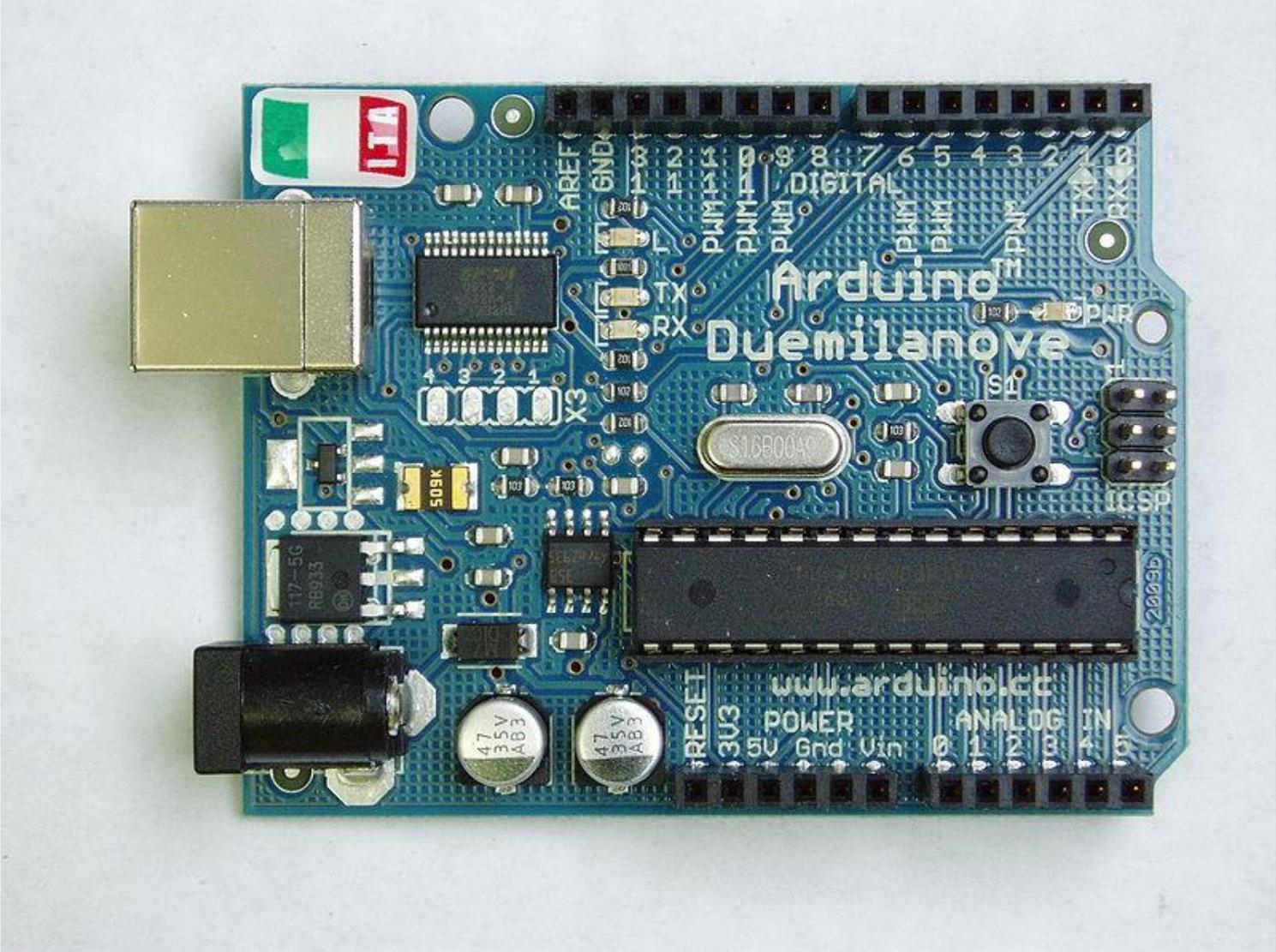
- Mentalidade de crescimento (*Growth Hacking*)
- CHA: Conhecimento-Habilidades-Atitude
- Cultura digital
- Protagonismo e liderança digital
- Autoridade percebida

# Perspectiva

- Influenciadores digitais: líderes do século XXI

**arduino**

# Arduino Duemilanove



# Arduino Uno



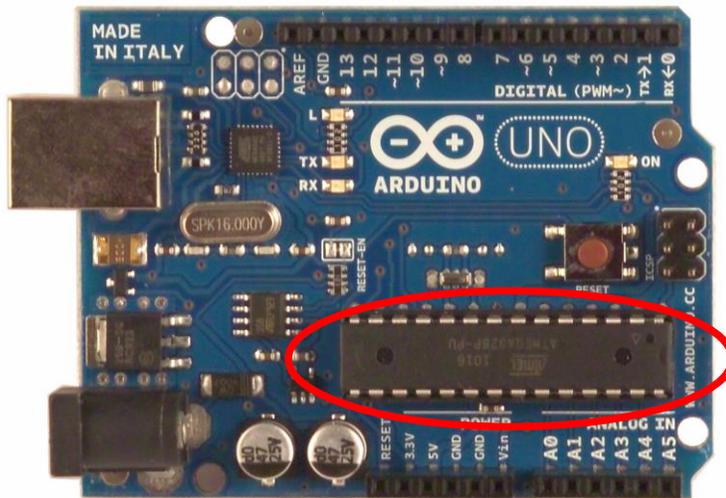
# Arduino Uno SMD



# O que é o Arduino?

▣ O Arduino é uma plataforma de prototipagem de eletrônicos de código aberto

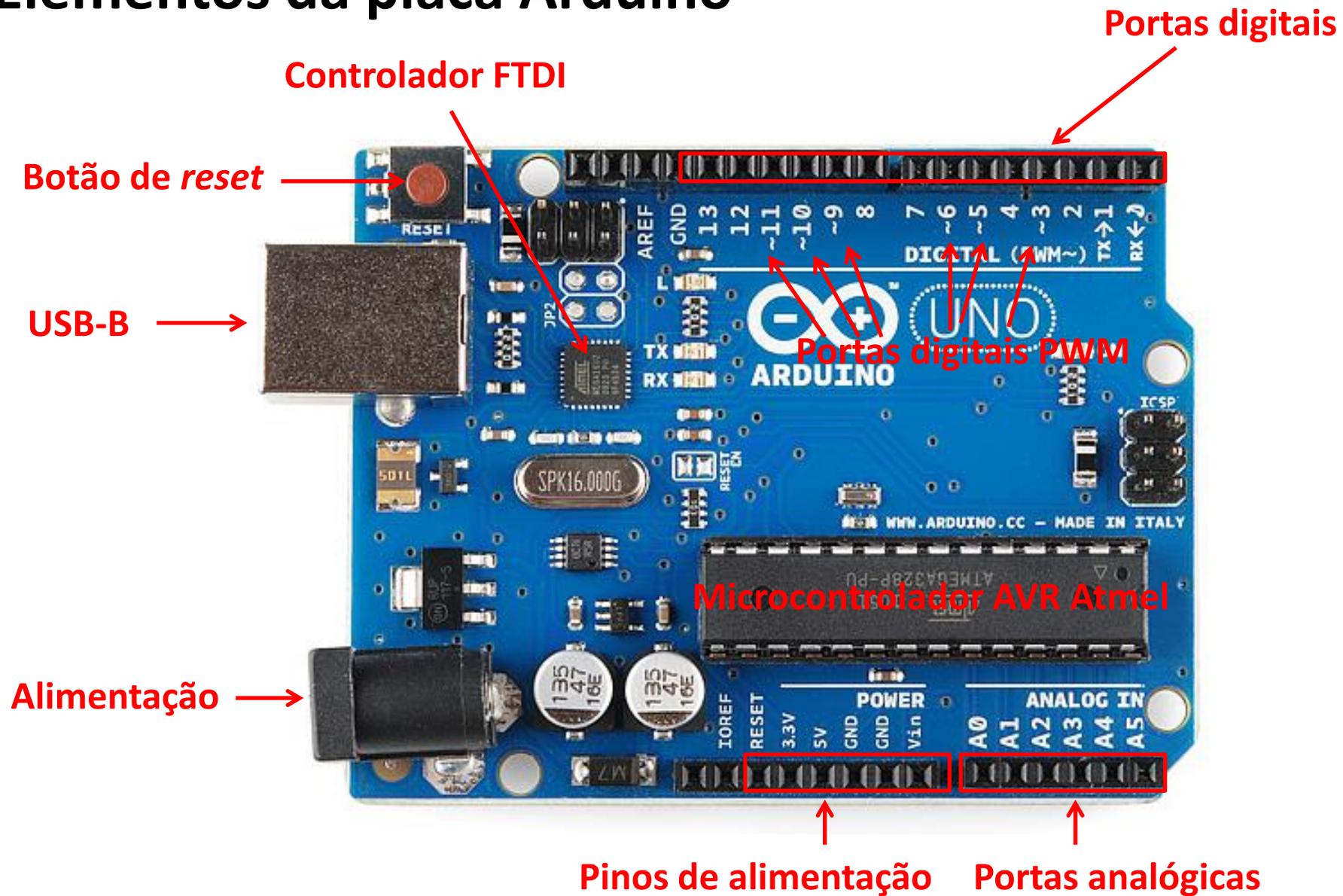
▣ Todos os diagramas e fontes de programação estão disponíveis sob licenças livres



Microcontrolador AVR  
Atmega328 da Atmel

O Arduino Uno é baseado no Atmega328 e contém pinos digitais de entrada e saída, entradas analógicas. A conexão USB é realizada por um chip separado

# Elementos da placa Arduino



# Especificações técnicas da placa Arduino



**RAM: 2K**  
**Flash: 32K**  
**Timers**  
**Serial (UART)**  
**I<sup>2</sup>C**  
**SPI**

# Cuidados

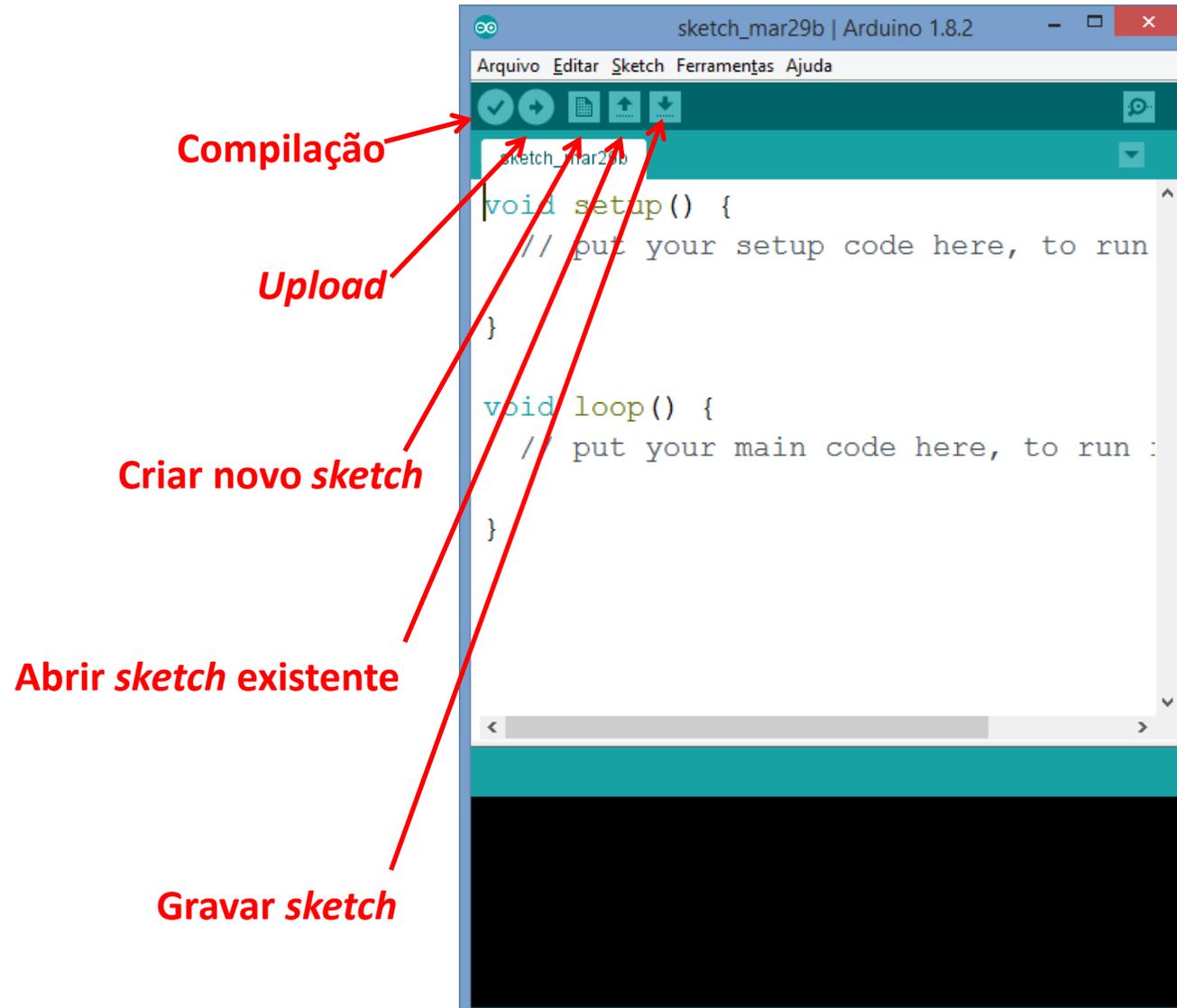
## Antes de começar:

- Limitações do Arduino:
  - Corrente máxima: 500 mA total, 50 mA por porta
- Certifique-se que seu circuito não requer mais corrente do que o Arduino pode oferecer
- Cuidado com curto-circuitos, pode queimar o microcontrolador
- Sempre desligue o Arduino quando for mexer no circuito eletrônico

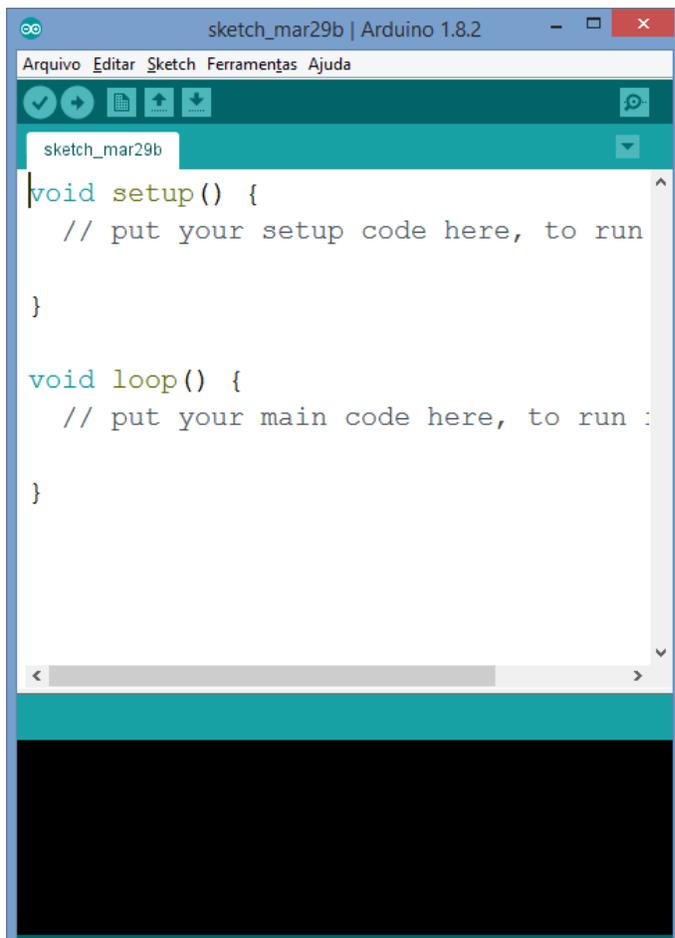
# Ambiente de programação: IDE *Wiring*



# Ambiente de programação: IDE *Wiring*



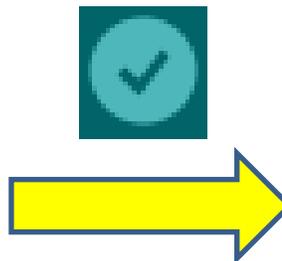
# Ambiente de programação: IDE Wiring



```
sketch_mar29b | Arduino 1.8.2
Arquivo Editar Sketch Ferramentas Ajuda
sketch_mar29b
void setup() {
  // put your setup code here, to run
}

void loop() {
  // put your main code here, to run
}
```

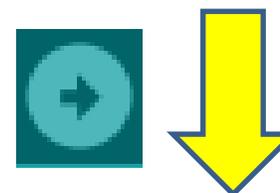
Código fonte (*sketch*)



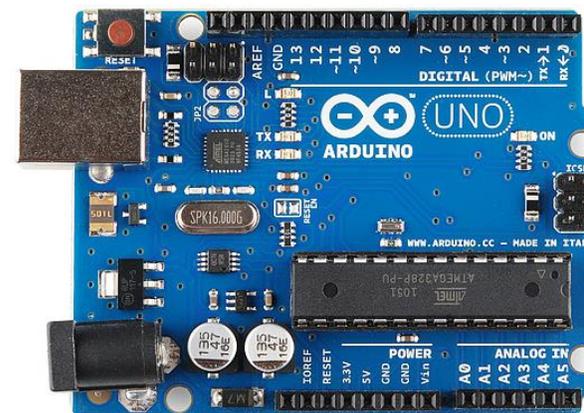
Compilação



Código binário



*Upload*



Arduino

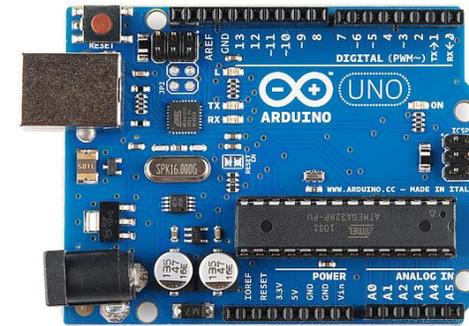
# Comunicação serial



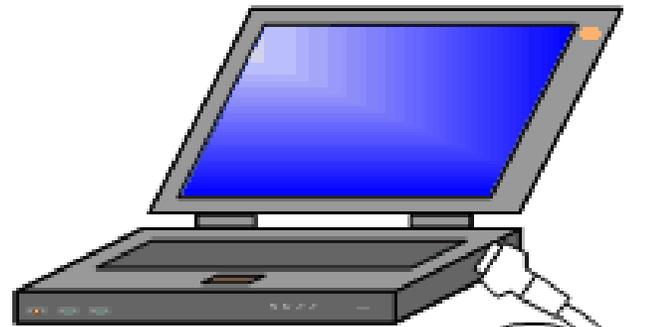
010010001101...



Porta USB

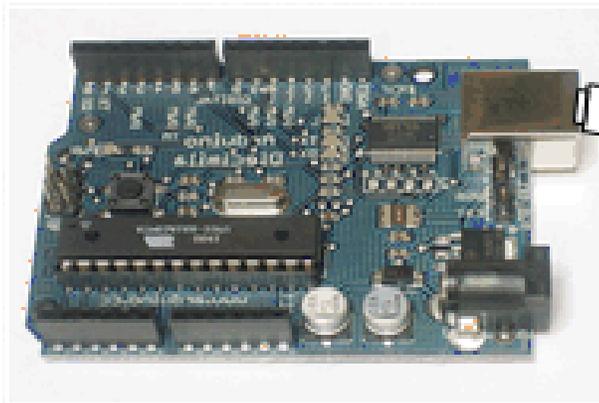


# Comunicação serial



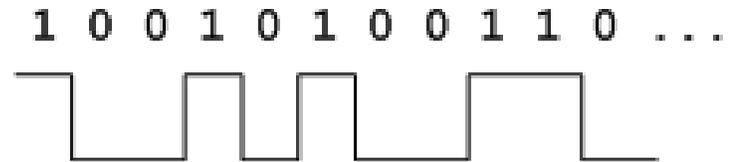
Notebook

Cabo USB



Arduino

A informação passa entre o Arduino e o computador por meio do cabo USB. A informação é transmitida como dígitos binários (bits) zeros ('0') e uns ('1').



# Comunicação serial

Permite receber e enviar informações entre o Arduino e um computador

- ▣ Deve ser configurado na função setup:
  - ▣ `Serial.begin(9600);` // Inicializa a porta serial para uma taxa de 9600 baud
- ▣ Dados são enviados para o PC pelo comando:
  - ▣ `Serial.println("Olá Arduino");`
- ▣ Na IDE utilize o monitor serial

# Meu primeiro *sketch* Arduino

```
/* olaArduino
   Primeiro sketch Arduino
*/

void setup() {
    Serial.begin(9600);
    Serial.println("Ola Arduino!");
}

void loop() {

}
```

# Gravando o *sketch* Arduino

The screenshot shows the Arduino IDE interface with the following elements:

- IDE Window:** Titled "aloArduino | Arduino 1.8.2". The menu bar includes "Arquivo", "Editar", "Sketch", "Ferramentas", and "Ajuda". The toolbar shows icons for check, undo, redo, upload, and download. The file name "aloArduino" is displayed in the top bar.
- Code Editor:** Contains the following C++ code:

```
/*  olaArduino.ino
   Meu primeiro sketch
*/

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

void loop() {
  Serial.println("Alo Arduino!");
}
```
- Serial Monitor:** Shows the status "Salvando..." and the output: "O sketch usa 1438 bytes (4%) de e" and "Variáveis globais usam 198 bytes".
- Save Dialog:** Titled "Salvar a pasta de sketches como...". The "Salvar em:" dropdown is set to "Aulas". The file list shows:

Nome	Data de modificaç...	Tipo	Tam
aloArduino	29/03/2017 09:46	Pasta de arquivos	
blinkLED13	29/03/2017 10:27	Pasta de arquivos	

The "Nome:" field contains "aloArduino" and the "Tipo:" dropdown is set to "Todos os Arquivos (\*.\*)". Buttons for "Salvar" and "Cancelar" are visible.
- Taskbar:** Shows the status "10" and "Arduino/Genuino Uno em COM4".

# Executando o *sketch* Arduino



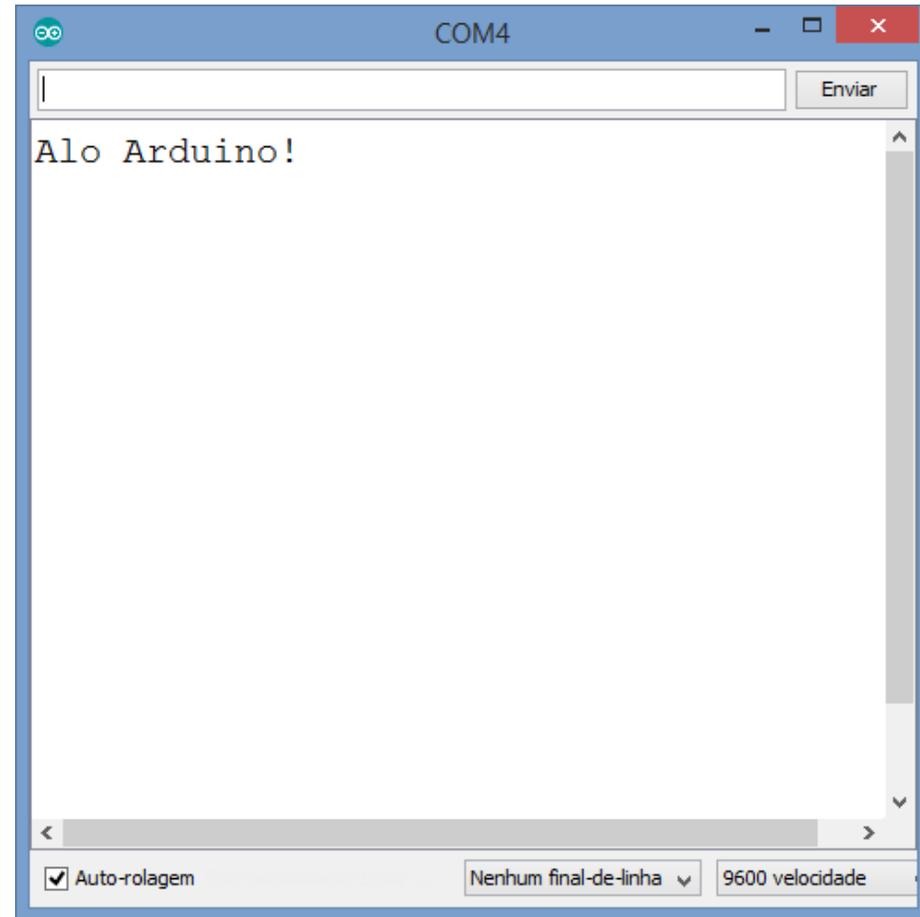
The screenshot shows the Arduino IDE editor window titled 'aloArduino | Arduino 1.8.2'. The menu bar includes 'Arquivo', 'Editar', 'Sketch', 'Ferramentas', and 'Ajuda'. The toolbar contains icons for saving, undo, redo, and uploading. The sketch name 'aloArduino \$' is shown in the top left. The code in the editor is as follows:

```
/*  olaArduino.ino
   Meu primeiro sketch
 */

void setup() {
  Serial.begin(9600);
  Serial.println("Alo Arduino!");
}

void loop() {
}
```

Below the code editor, a status bar indicates 'Carregado.' and provides memory usage information: 'O sketch usa 1438 bytes (4%) de espaço' and 'Variáveis globais usam 198 bytes (9%) d'. At the bottom, it shows '9' and 'Arduino/Genuino Uno em COM4'.



The screenshot shows the serial monitor window titled 'COM4'. It features a text input field at the top with an 'Enviar' button. The main area displays the output 'Alo Arduino!'. At the bottom, there are settings for 'Auto-rolagem' (checked), 'Nenhum final-de-linha', and '9600 velocidade'.

**Janela do monitor serial**

# Executando a instrução `Serial.println` no *loop*

```
/* olaArduino.ino
   Primeiro sketch Arduino
*/

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

void loop() {
    Serial.println("Ola Arduino!");
}
```

# Executando o *sketch* Arduino



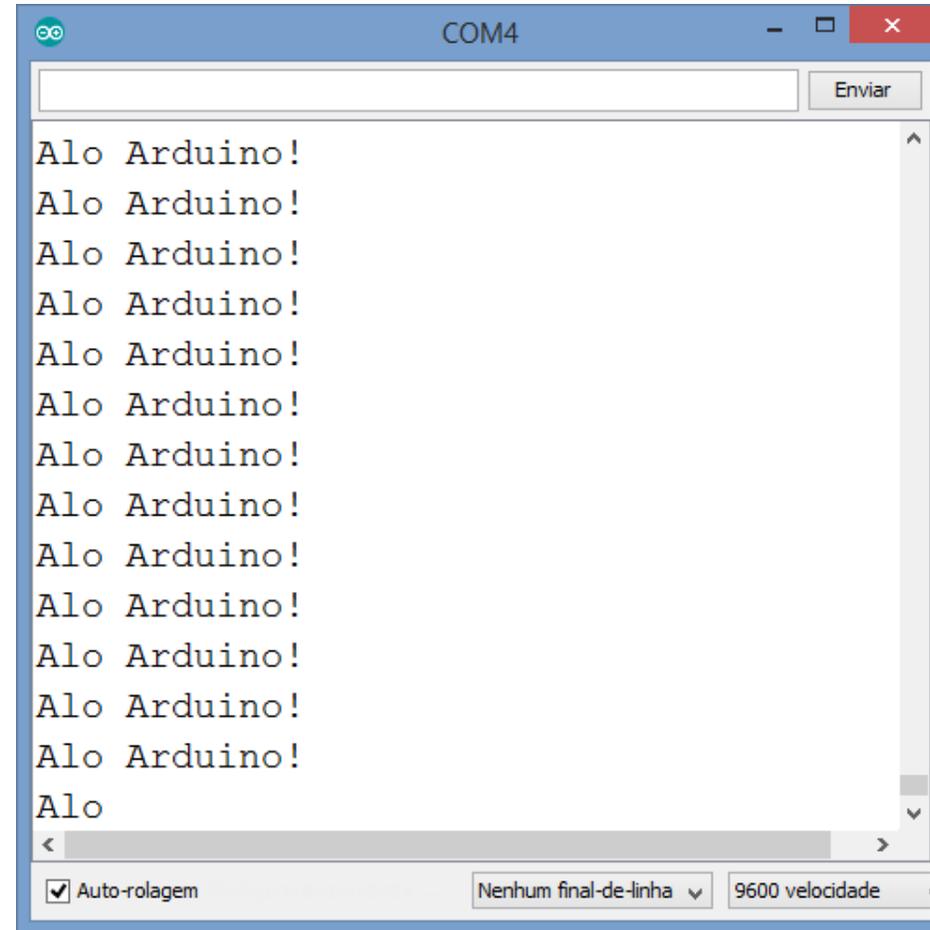
The screenshot shows the Arduino IDE interface. The main window is titled "aloArduino | Arduino 1.8.2". The menu bar includes "Arquivo", "Editar", "Sketch", "Ferramentas", and "Ajuda". The toolbar contains icons for saving, running, uploading, and downloading. The sketch editor shows the following code:

```
/* olaArduino.ino
   Meu primeiro sketch
*/

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

void loop() {
  Serial.println("Alo Arduino!");
}
```

Below the editor, a status bar indicates "Carregado." and provides memory usage information: "O sketch usa 1438 bytes (4%) de espaço" and "Variáveis globais usam 198 bytes (9%) d". The bottom status bar shows "10" and "Arduino/Genuino Uno em COM4".



The screenshot shows the Arduino Serial Monitor window titled "COM4". The window contains a text area with the following output:

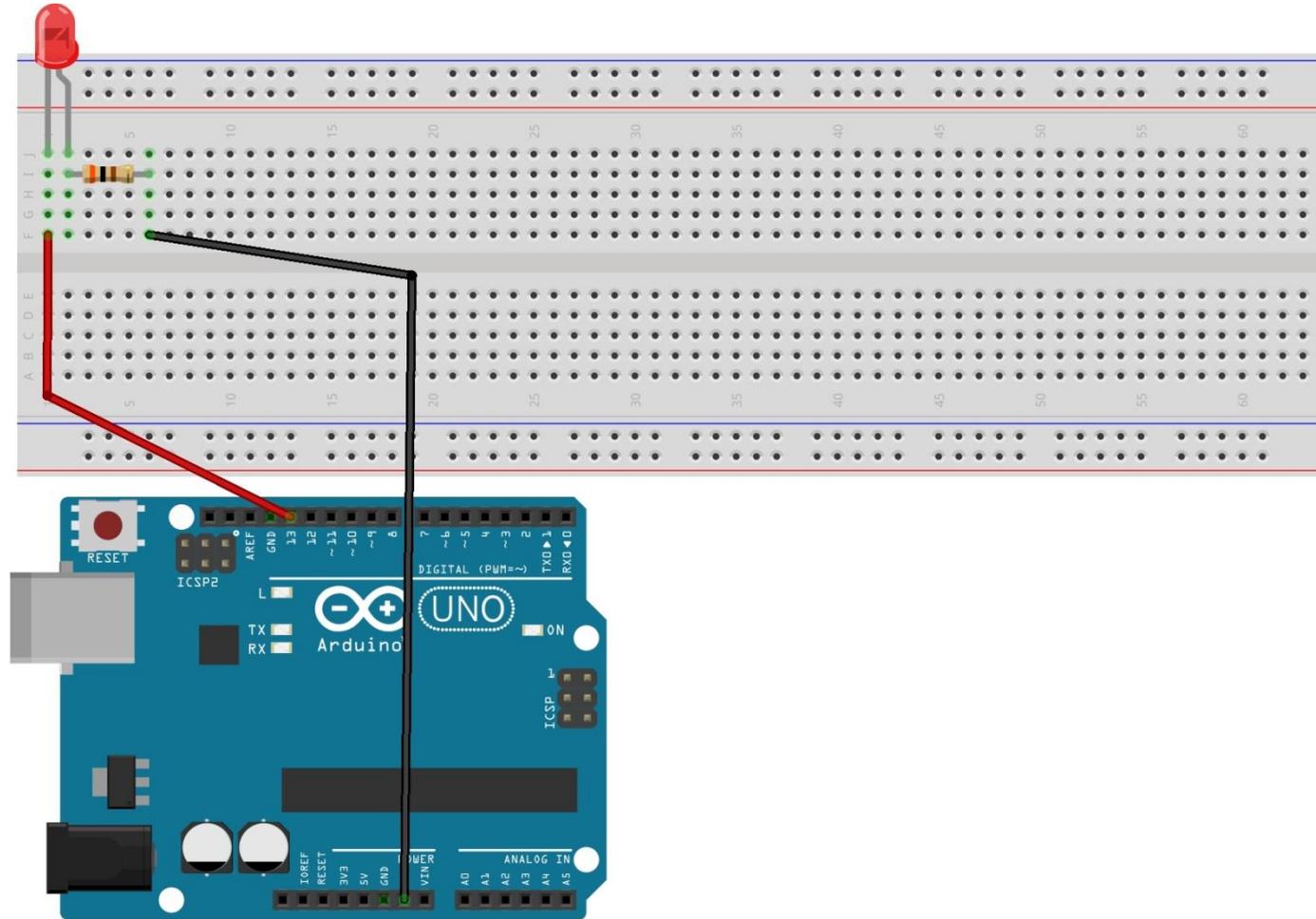
```
Alo Arduino!
Alo
```

At the bottom of the window, there are controls for "Auto-rolagem" (checked), "Nenhum final-de-linha" (dropdown), and "9600 velocidade" (dropdown). An "Enviar" button is located at the top right of the text area.

# Entrada e saída digital

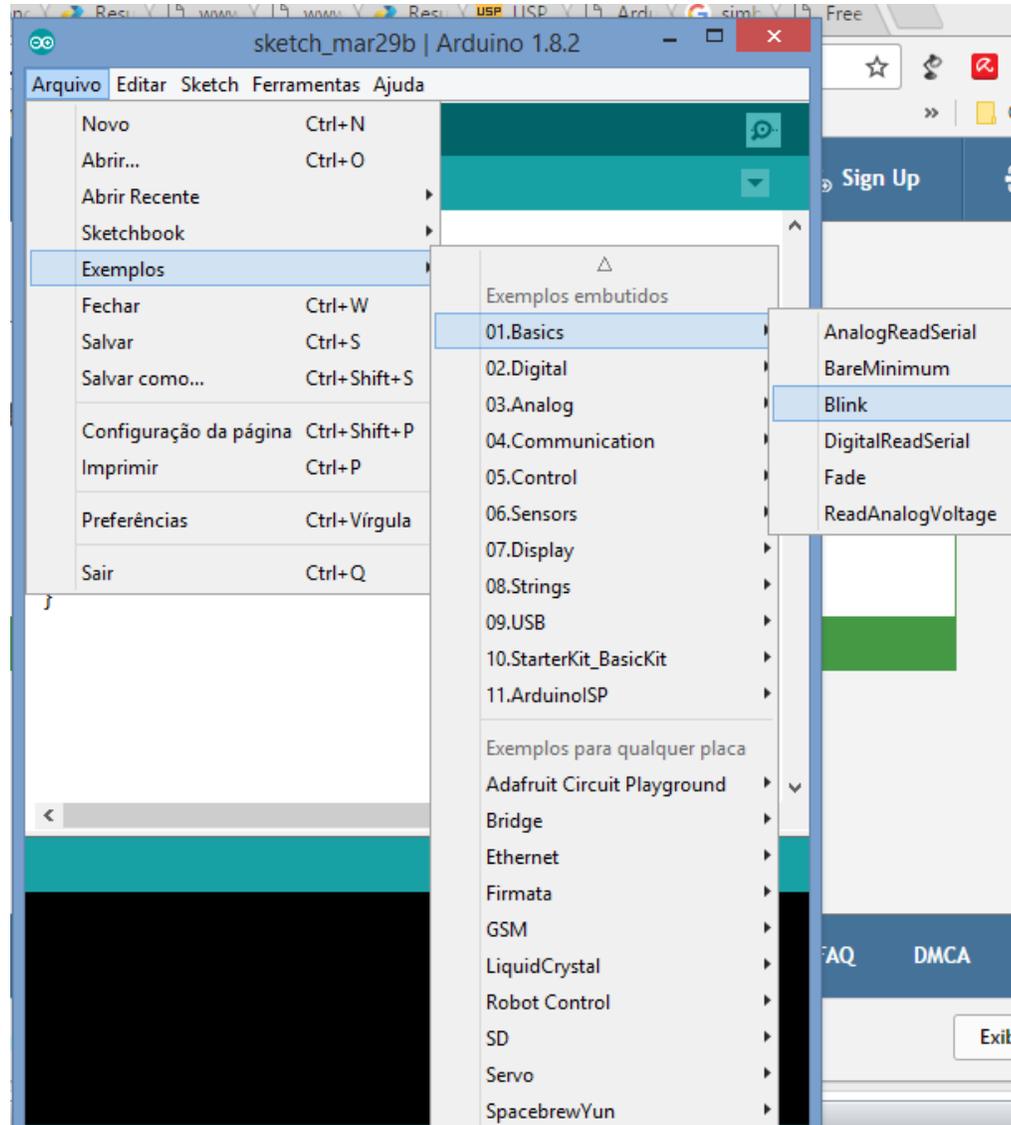
- ▣ Entradas e saídas digitais são portas programáveis para leitura ou "gravação" de um sinal digital (0 ou 1 – ligado ou desligado – zero ou 5 volts, HIGH e LOW)
  - ▣ Entrada: botão pressionado, porta aberta...
  - ▣ Saída: ligar e desligar lâmpadas, motores... (interruptor)

# Montagem do LED com resistor





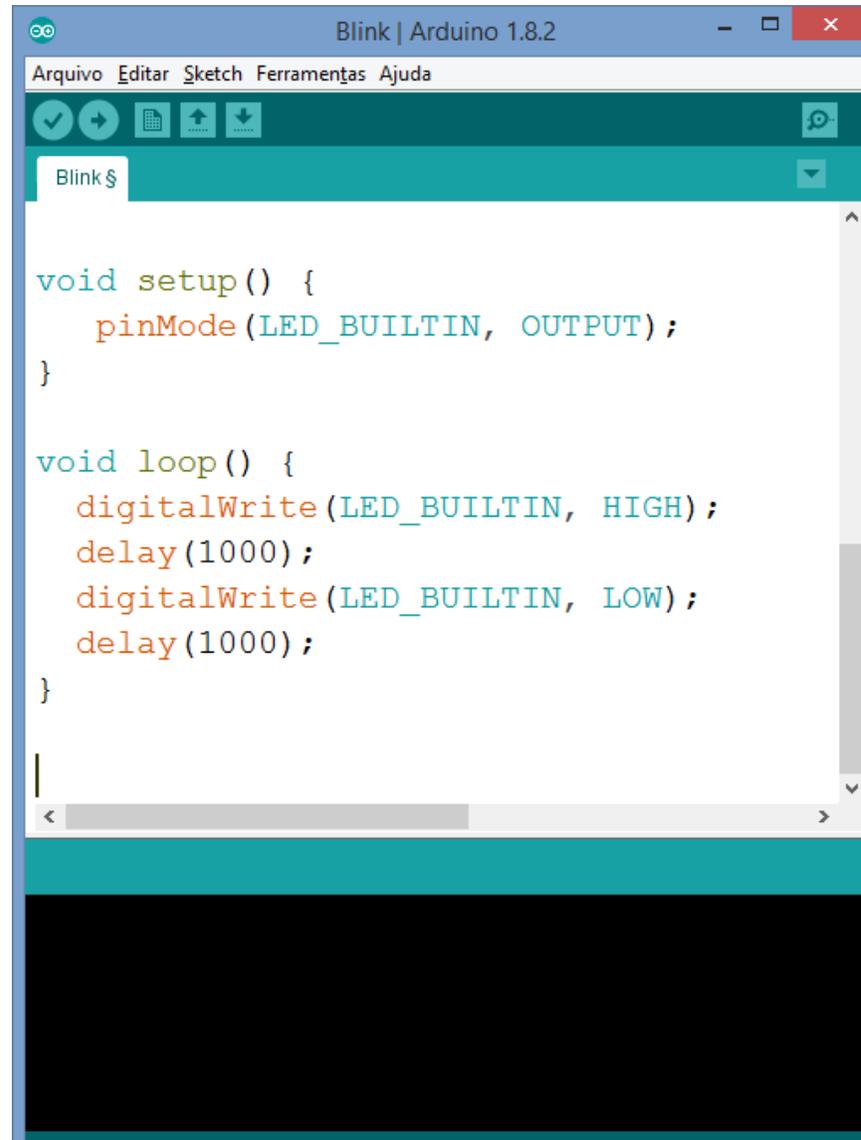
# Abrindo a janela de exemplos do IDE Wiring



# Sketch blink (exemplo do Arduino)

```
/*  
  Blink  
  Turns on an LED on for one second, then off for one second, repeatedly.  
  
  This example code is in the public domain.  
  
  modified 8 May 2014  
  by Scott Fitzgerald  
  
  modified 2 Sep 2016  
  by Arturo Guadalupi  
  
  modified 8 Sep 2016  
  by Colby Newman  
*/  
  
// the setup function runs once when you press reset or power the board  
void setup() {  
  // initialize digital pin LED_BUILTIN as an output.  
  pinMode(LED_BUILTIN, OUTPUT);  
}  
  
// the loop function runs over and over again forever  
void loop() {  
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)  
  delay(1000); // wait for a second  
  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW  
  delay(1000); // wait for a second  
}
```

# Executando o *sketch* Arduino

A screenshot of the Arduino IDE interface. The window title is "Blink | Arduino 1.8.2". The menu bar includes "Arquivo", "Editar", "Sketch", "Ferramentas", and "Ajuda". The toolbar contains icons for checkmark, right arrow, grid, upload, download, and search. The tab bar shows "Blink\$". The main editor area contains the following C++ code:

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

The bottom of the window is a dark teal bar, and the bottom-most part is a black area, likely representing the serial monitor or a terminal window.

# Sketch blink modificado

```
/*  
  Blink  
  
  modified 8 May 2014  
  by Scott Fitzgerald  
  
  modified 2 Sep 2016  
  by Arturo Guadalupi  
  
  modified 8 Sep 2016  
  by Colby Newman  
*/  
  
void setup() {  
  pinMode(13, OUTPUT);  
}  

```

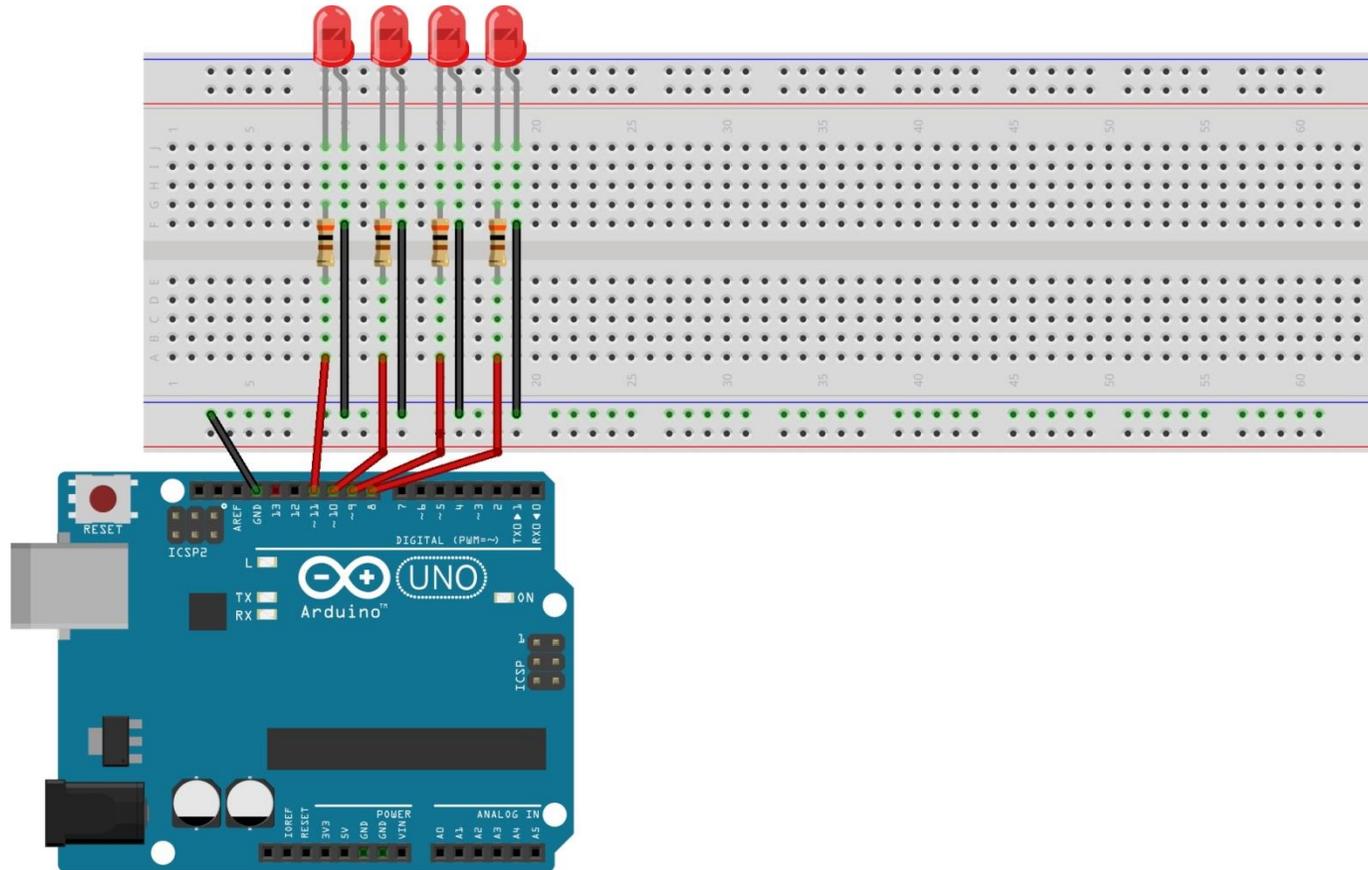
# Sketch blink com variável LED

```
/*  
  Blink  
  
  modified 8 May 2014  
  by Scott Fitzgerald  
  
  modified 2 Sep 2016  
  by Arturo Guadalupi  
  
  modified 8 Sep 2016  
  by Colby Newman  
*/  
  
int LED = 13;  
  
void setup() {  

```

# Prática de programação do Arduino

Montar um circuito de quatro LEDs com acendimento sequencial, usando as portas digitais 8, 9, 10 e 11



# Sketch do LED sequencial

```
/*
  LEDsequencial
  Liga e desliga quatro LEDs
  em sequencia
  */

int pinLED1 = 8;
int pinLED2 = 9;
int pinLED3 = 10;
int pinLED4 = 11;
int atraso = 30;

void setup() {
  pinMode(pinLED1, OUTPUT);
  pinMode(pinLED2, OUTPUT);
  pinMode(pinLED3, OUTPUT);
  pinMode(pinLED4, OUTPUT);
}

void loop() {
  digitalWrite(pinLED1, HIGH);
  delay(atraso);
  digitalWrite(pinLED1, LOW);
  delay(atraso);
  digitalWrite(pinLED2, HIGH);
  delay(atraso);
  digitalWrite(pinLED2, LOW);
  delay(atraso);
  digitalWrite(pinLED3, HIGH);
  delay(atraso);
  digitalWrite(pinLED3, LOW);
  delay(atraso);
  digitalWrite(pinLED4, HIGH);
  delay(atraso);
  digitalWrite(pinLED4, LOW);
  delay(atraso);
}
```