

# TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

# DISRUPTIVE ARCHITECTURES: IOT, IOB E GENERATIVE AI

**Prof. André Tritiack de Farias**  
[profandre.farias@fiap.com.br](mailto:profandre.farias@fiap.com.br)

2023

# DISRUPTIVE ARCHITECTURES: IOT, IOB E GENERATIVE AI

## OBJETIVOS DE HOJE

- **Introdução aos Microcontroladores**
  - **Arduino UNO**
- **Pinos de Entrada e Saída (INPUT/OUTPUT – I/O)**
  - **Apresentação do Arduino IDE**

# O que é Arduino?

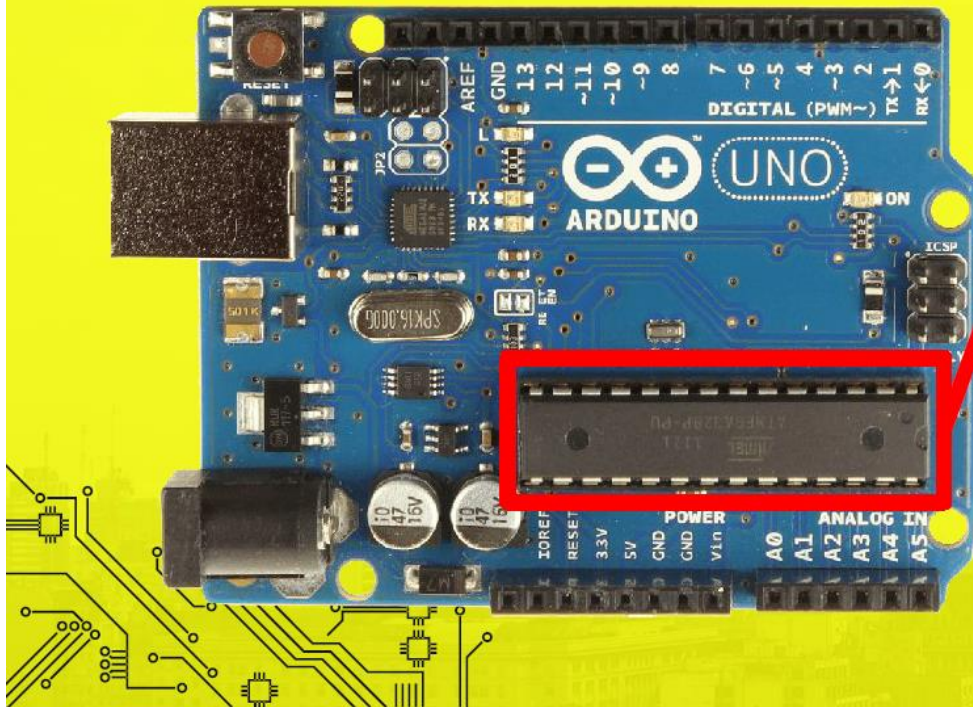
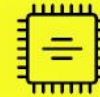
- Plataforma de prototipagem de hardware livre
- Possui microcontrolador AVR ATmega
- Arquitetura Harvard (modificada) de 8 bits (instruções)
- Versão UNO:
  - ◆ ATmega328p | 16 MHz | 1kB EEPROM | 2kB SRAM | 32 kB



# Componentes

## Microcontrolador

Processa os dados e executa a programação





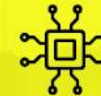
# Componentes



**Portas analógicas**  
(0 até 1023)

## Aplicações

→ Leitura de sensores analógicos



# Componentes



## Portas digitais

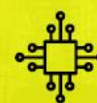
(0 ou 1)

PWM~ (0 a 255)

Serial (RX e TX)

## Aplicações

- Leitura de sensores digitais
- Acionamento de atuadores
- Comunicação





# Componentes

## Entrada jack

Serve como alimentação externa para a placa.





# Componentes

## Entrada USB



Utilizada para possibilitar a gravação do firmware no microcontrolador.

Também faz comunicação Serial (transferência de dados em série) e alimentação (5V).

# Componentes

Vin



Serve como alimentação externa para a placa.

Caso for conectada alguma fonte de energia (entrada jack ou USB) a Vin passará a fornecer a mesma tensão da fonte. Portanto, NUNCA conecte outra fonte de energia no Vin quando o Arduino estiver ligado.



# Componentes

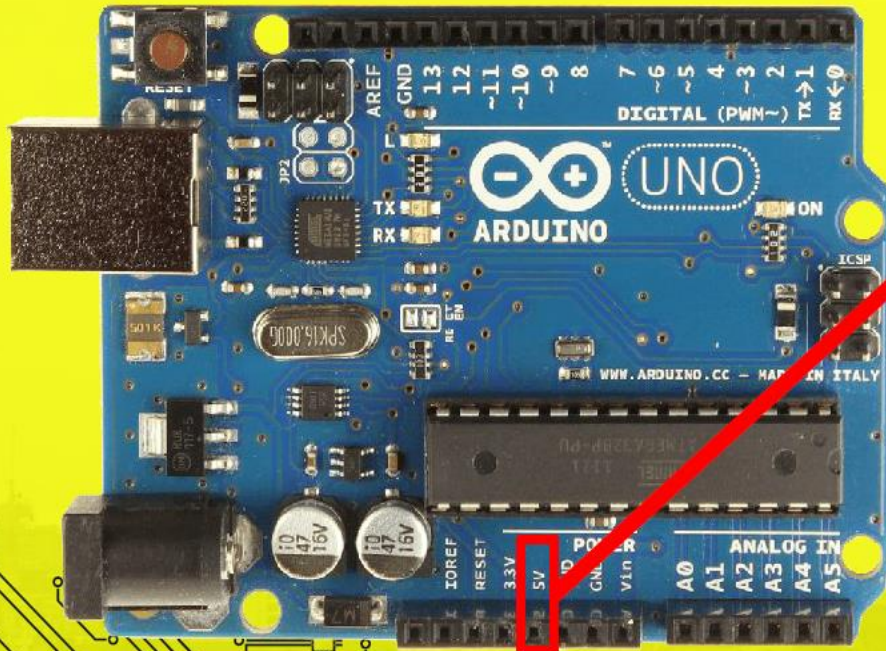


**GND**

Pontos de aterramento ("terra")  
da placa.



# Componentes



**5V**

Fornece a tensão de 5V  
(corrente contínua) para  
circuitos externos.

Ideal para alimentação de  
sensores e atuadores de baixa  
tensão.

# Componentes



**3.3V**

Fornece a tensão de 3.3V  
(corrente contínua) para  
circuitos externos.

Ideal para alimentação de  
sensores e atuadores de baixa  
tensão.



# Componentes

**AREF**



Utilizado para mudar a escala de leitura analógica dos pinos (o padrão é de 0V até 5V, mas é possível diminuir esses valores).

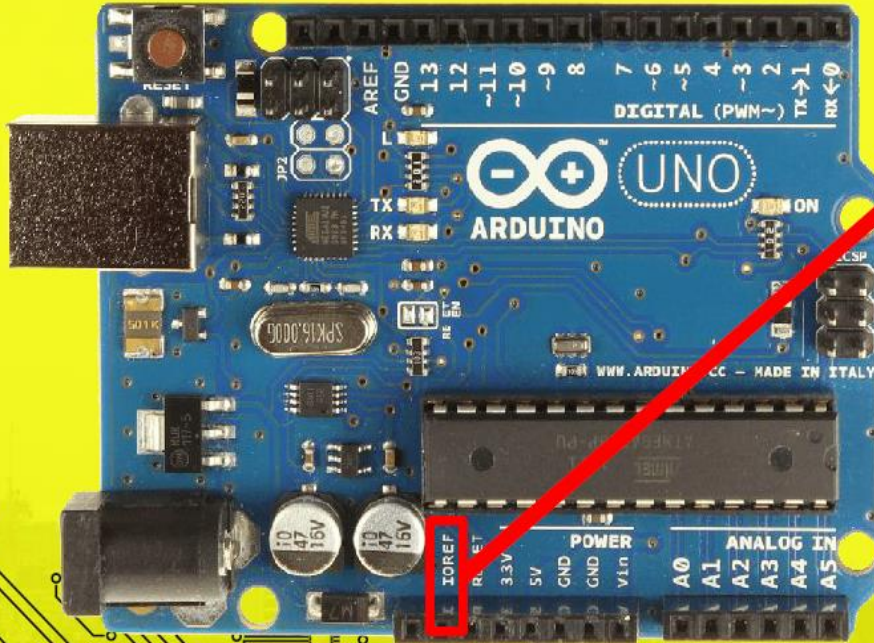
Não coloque tensões menores que 0V ou maiores que 5V, caso contrário poderá danificar o microcontrolador.



# Componentes

**IOREF**

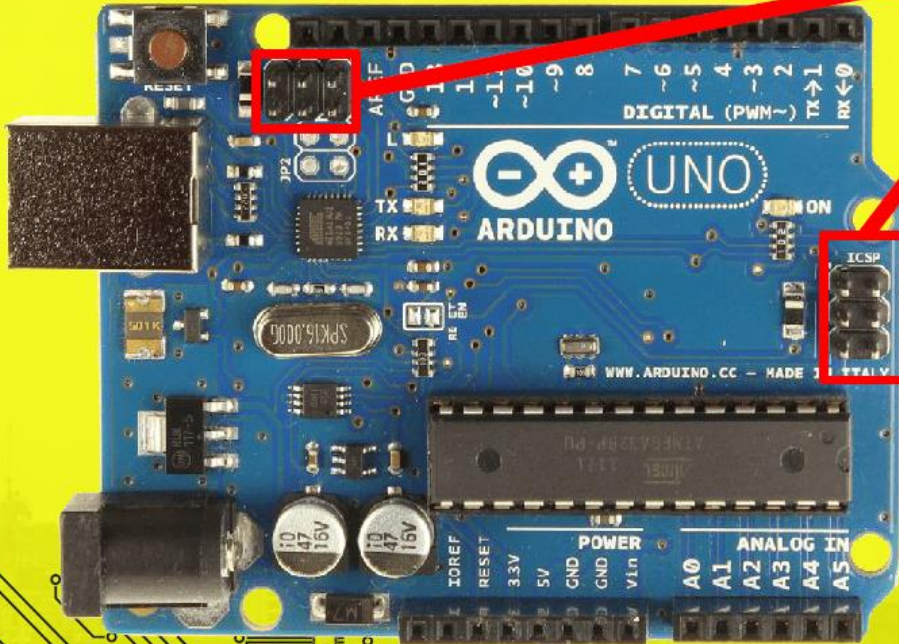
Fornece a tensão de operação  
que as portas de entrada e saída  
estão operando.



# Componentes

## ICSP headers

Possibilita comunicação através do protocolo SPI e também é uma alternativa para gravar firmwares no Arduino

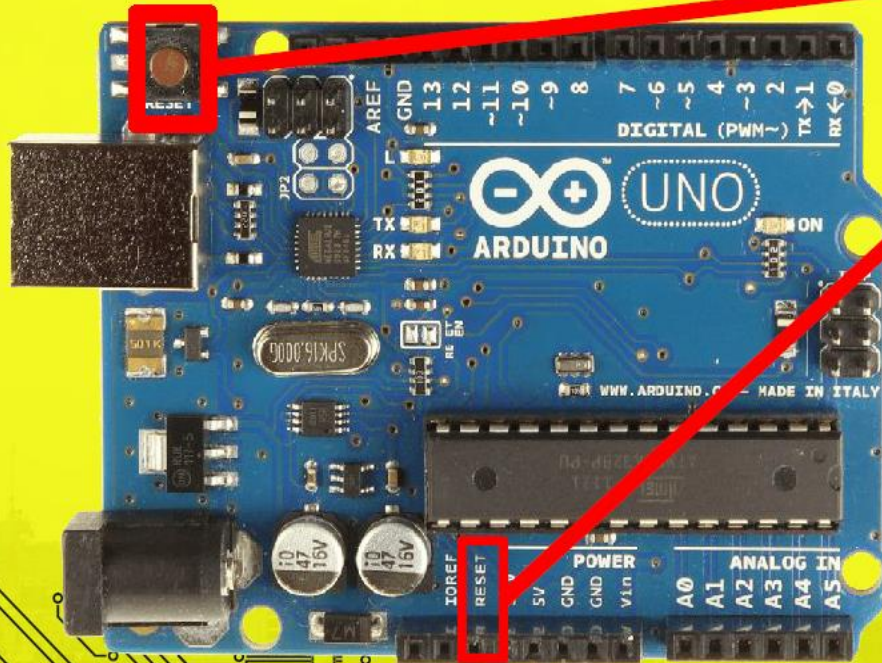




# Componentes

**Reset**

Serve para reiniciar a placa







# Comandos Principais

para programação do firmware em Wiring

<https://www.arduino.cc/reference/pt/>



# pinMode ( **pin**, **type** )

→ Configura uma porta como **entrada** ou **saída** de corrente elétrica

- ◆ **pin**: número da porta
- ◆ **type**: INPUT (entrada) ou OUTPUT (saída)
- ◆ Exemplo: pinMode(13, OUTPUT);

<https://www.arduino.cc/reference/pt/>



**digitalWrite(*pin*, *action*)**

→ **Liga** ou **desliga** uma porta digital.

- ◆ **pin:** número da porta que deseja alterar o estado
- ◆ **action:** HIGH ou 1 (ligado), LOW ou 0 (desligado)
- ◆ Exemplo: digitalWrite(13, HIGH);

<https://www.arduino.cc/reference/pt/>







`delay( time )`

→ **Aguarda** alguns milissegundos antes de iniciar a próxima instrução (linha) do código.

- ◆ **time**: tempo em milissegundos
- ◆ Exemplo: `delay(1000);`

<https://www.arduino.cc/reference/pt/>