



# Conhecendo o Arduino

Capacitação em Arduino  
PET ELÉTRICA - UFPB



## Origem do Arduino

Contexto tecnológico e a necessidade do ensino de robótica.

01

## Características básicas do Arduino

Configurações da placa utilizada: alimentação, entrada e saída, etc.

02

## Pinos Digitais e Analógicos

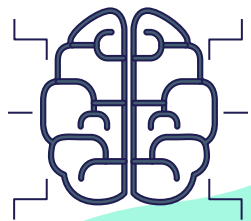
Entendendo a diferença entre grandeza analógica e grandeza digital

03

## Aplicações

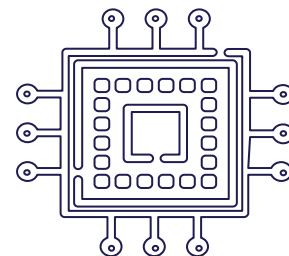
Demonstração de desenvolvimento de projetos utilizando Arduino.

04



# 01

## ORIGEM DO ARDUINO





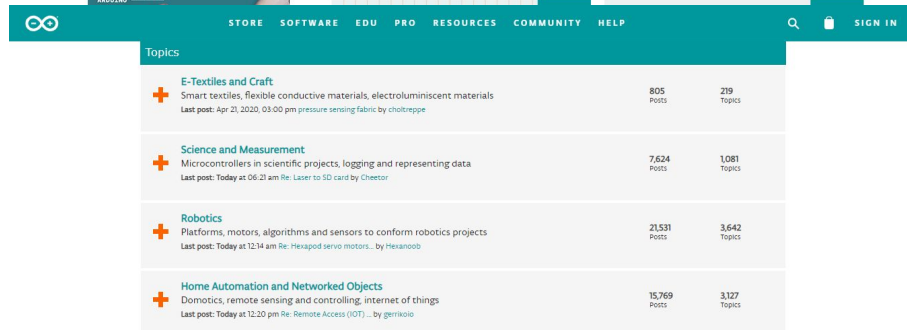
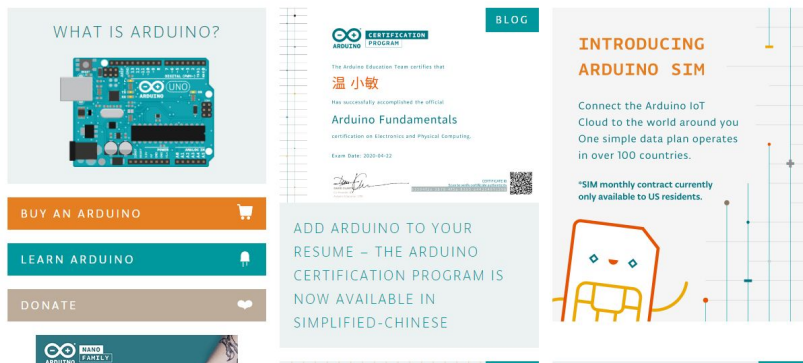
# Origem do Arduino

- ▶ A ideia de desenvolver o Arduino surgiu na Itália graças a um grupo de pesquisadores.
- ▶ Possibilidade de confeccionar um circuito específico e interativo com a aplicação desejada.
- ▶ Necessidade do ensino de robótica e programação para os mais jovens.
- ▶ Desenvolvimento e exercício da criatividade individual e coletiva.





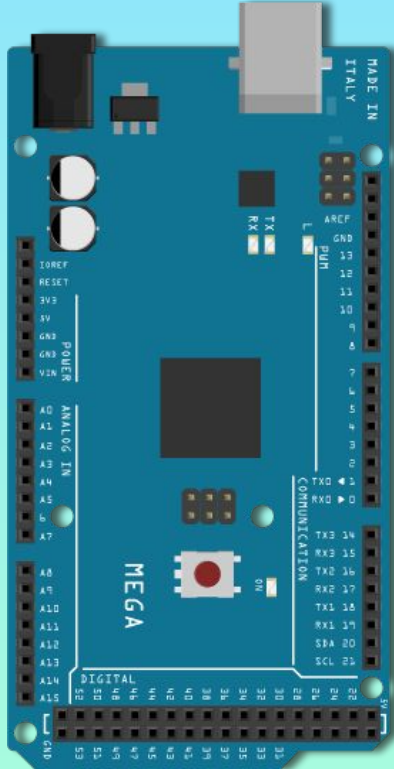
# Origem do Arduino



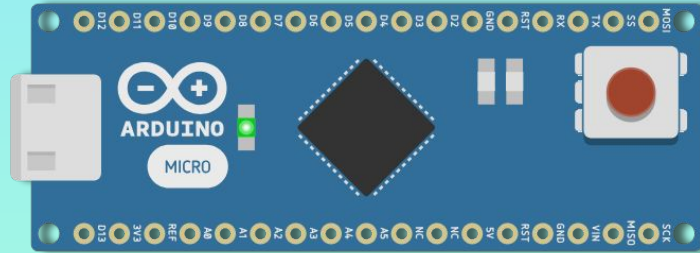
Para mais informações, no site oficial Arduino é disponibilizado diversos materiais, além de uma revista sobre o Arduino para melhor informar os interessados.



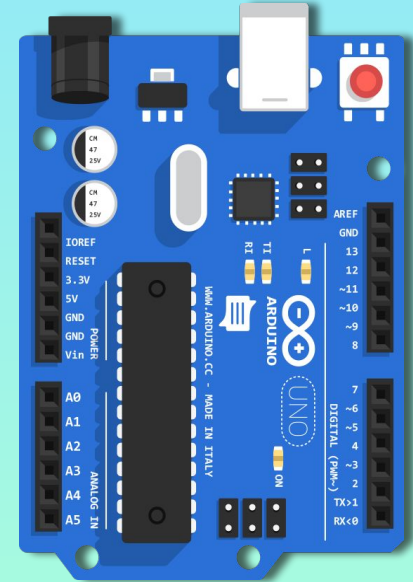
# Tipos de Arduino



Arduino MEGA



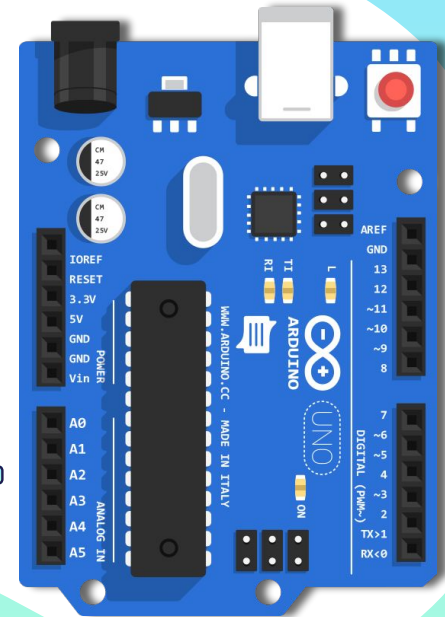
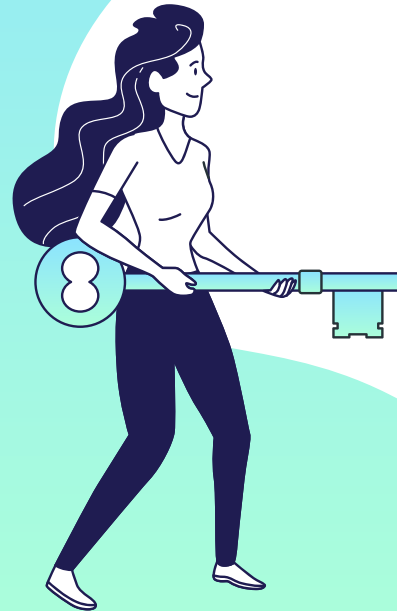
Arduino Nano

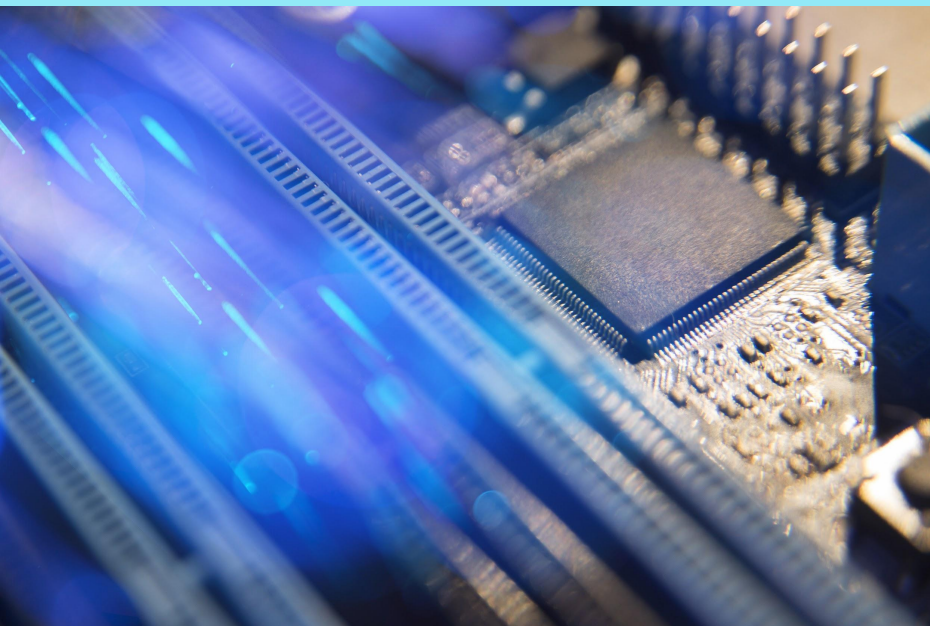


Arduino Uno R3

# O Arduino Uno R3

Primeiramente, é fundamental entendermos o que compõe o Hardware como forma de conhecermos o poder dessa ferramenta de prototipagem.



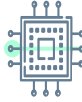


# 02

## CARACTERÍSTICAS BÁSICAS DO ARDUINO

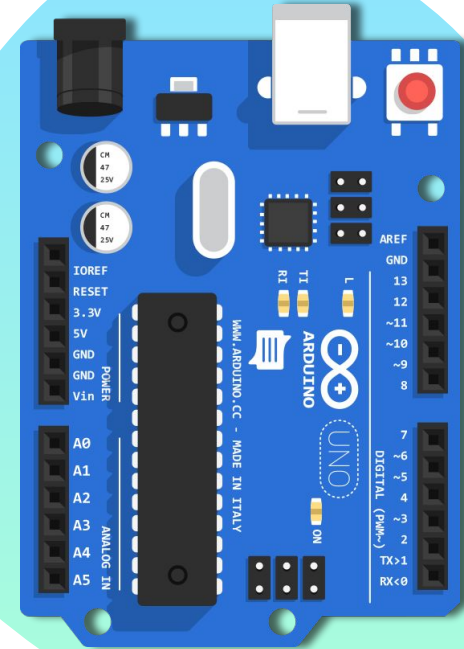


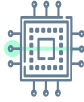




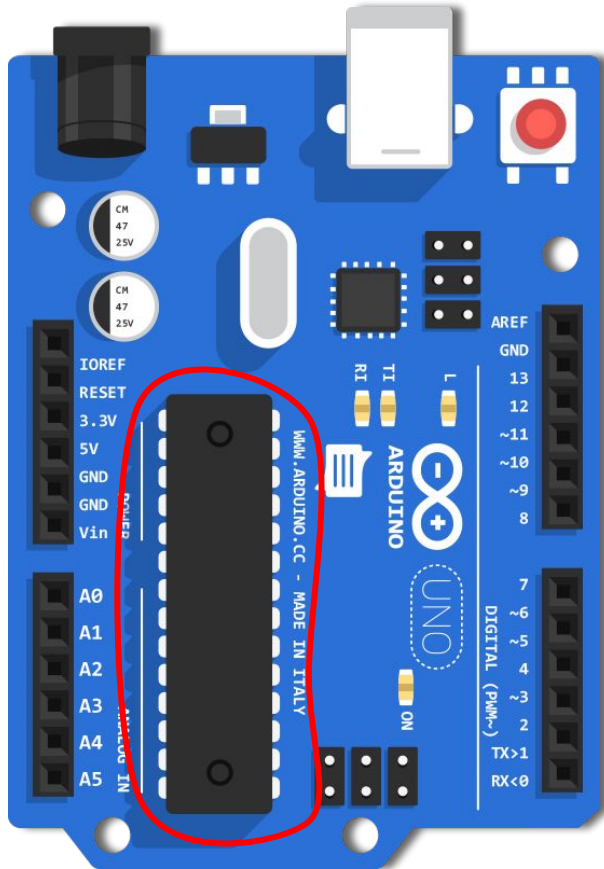
# Características Básicas do Arduino

Microcontrolador;  
Tensão Operacional;  
Tensão de Alimentação;  
Corrente Contínua;  
Memória;  
Frequência de Clock;  
Pinos de Entrada Analógica;  
Pinos de I/O Digitais;





# Microcontrolador

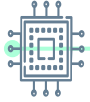


## Tipos

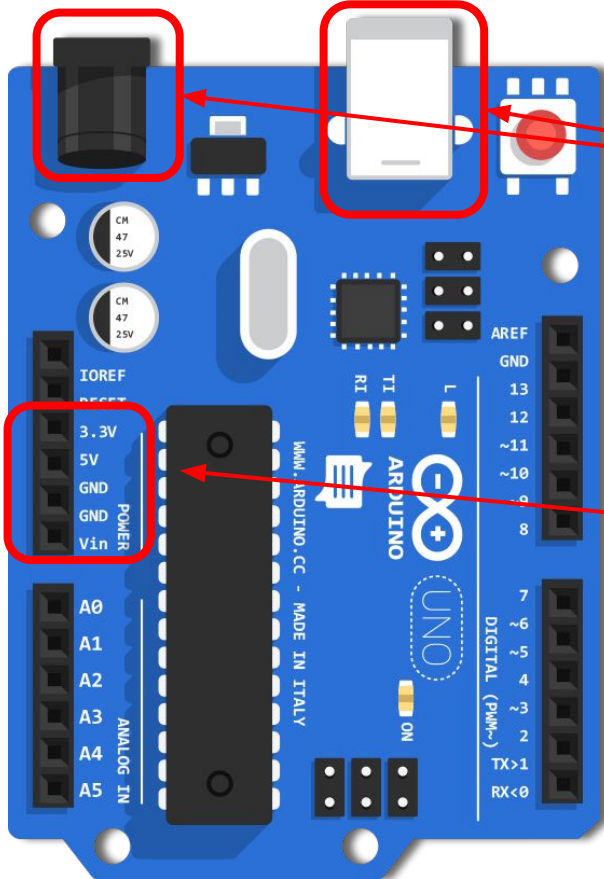
ATmega328 ou ATmega168.

## Atmega328

Possui memória flash, EEPROM e SRAM. Recebe e compila as informações do arduino.



# Alimentação



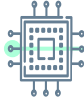
## Tipos

Conexão USB ou Fonte de Alimentação Externa.

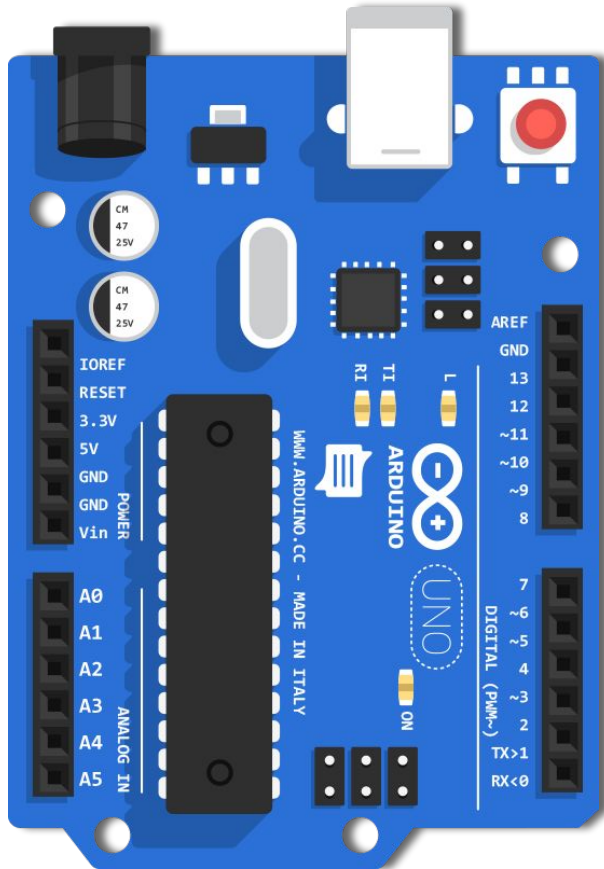
## Pinos de Alimentação

$V_{in}$ : Alimentação recebida pela Placa

5V, 3.3V e GND.



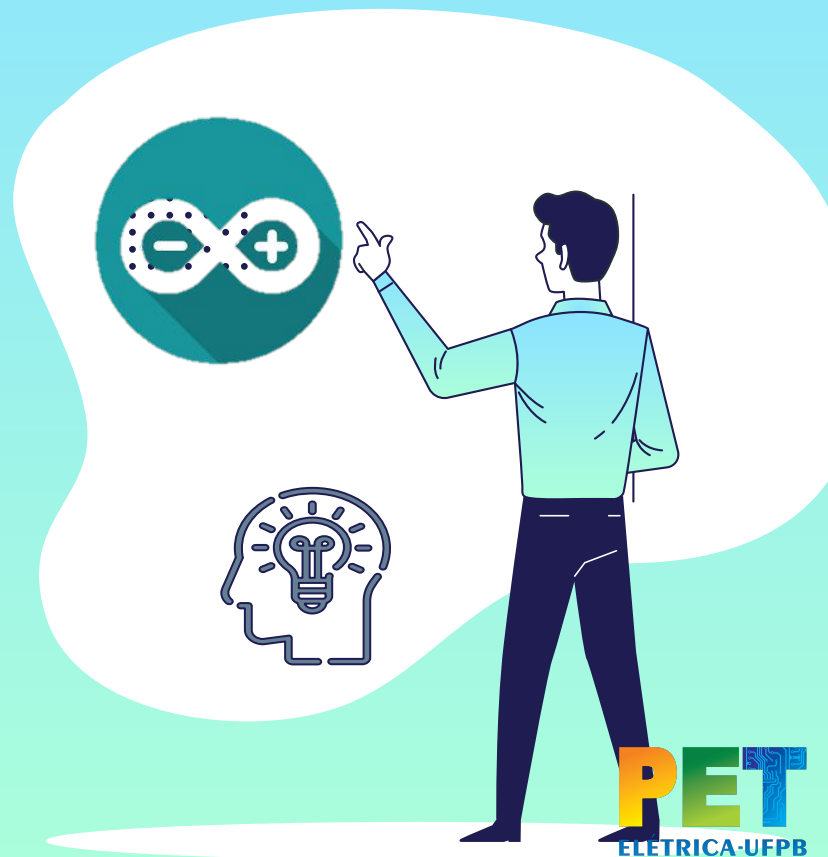
## Outras Características

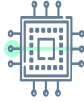


Tensão Operacional	5 V
Tensão de Alimentação	7 - 12 V
Corrente Contínua	40mA / 50mA
Memória	32KB
Frequência de Clock	16MHz

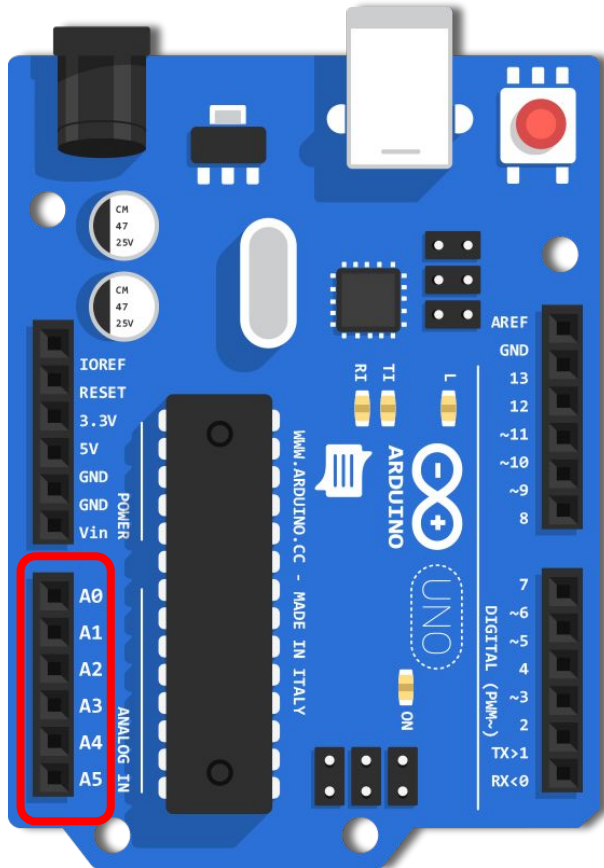
# 03

## Pinos Digitais e Analógicos





# Pinos de Entrada Analógica

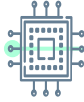


## Quais são?

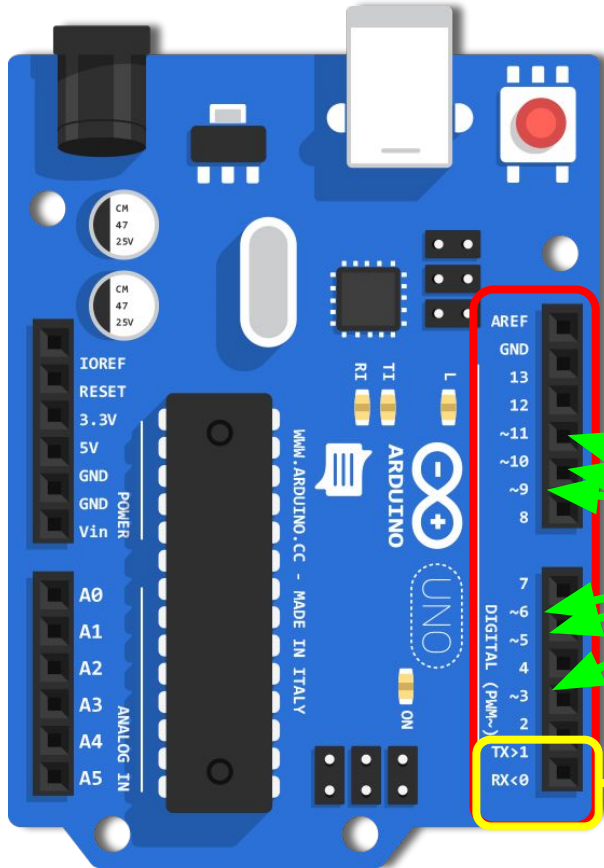
São 6 pinos de entrada analógica que são nomeadas de A0 a A5.

## Valores

Podem ler qualquer valor com determinada precisão (10 bits), porém, não podem enviar valores.



# Pinos de Entrada/Saída Digitais



## Binárias Comuns

14 portas digitais,  
numeradas de 0 a 13.



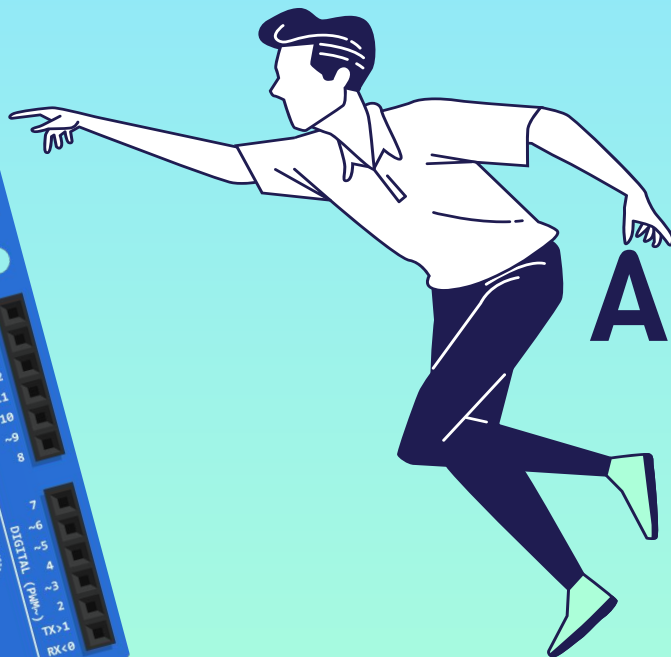
## PWM

Das 14 portas digitais, 6  
delas são PWM (Pulse Width  
Modulation).

Através das portas 0 e 1, RX  
e TX, conseguimos realizar  
comunicação serial.

04

# APLICAÇÕES





# Aplicações

Arduino é uma plataforma composta por dois componentes, a placa e a IDE Arduino.

Dentre as vantagens da utilização de Arduino, destaca-se:

- Fácil Manuseio;
- Hardware Livre e Placa Única;
- Consideravelmente de Baixo Custo;
- Compatibilidade;
- Simplifica Processos;
- Existência de Fóruns de Discussão;
- Ambiente de Programação Simples.





Alarme\$

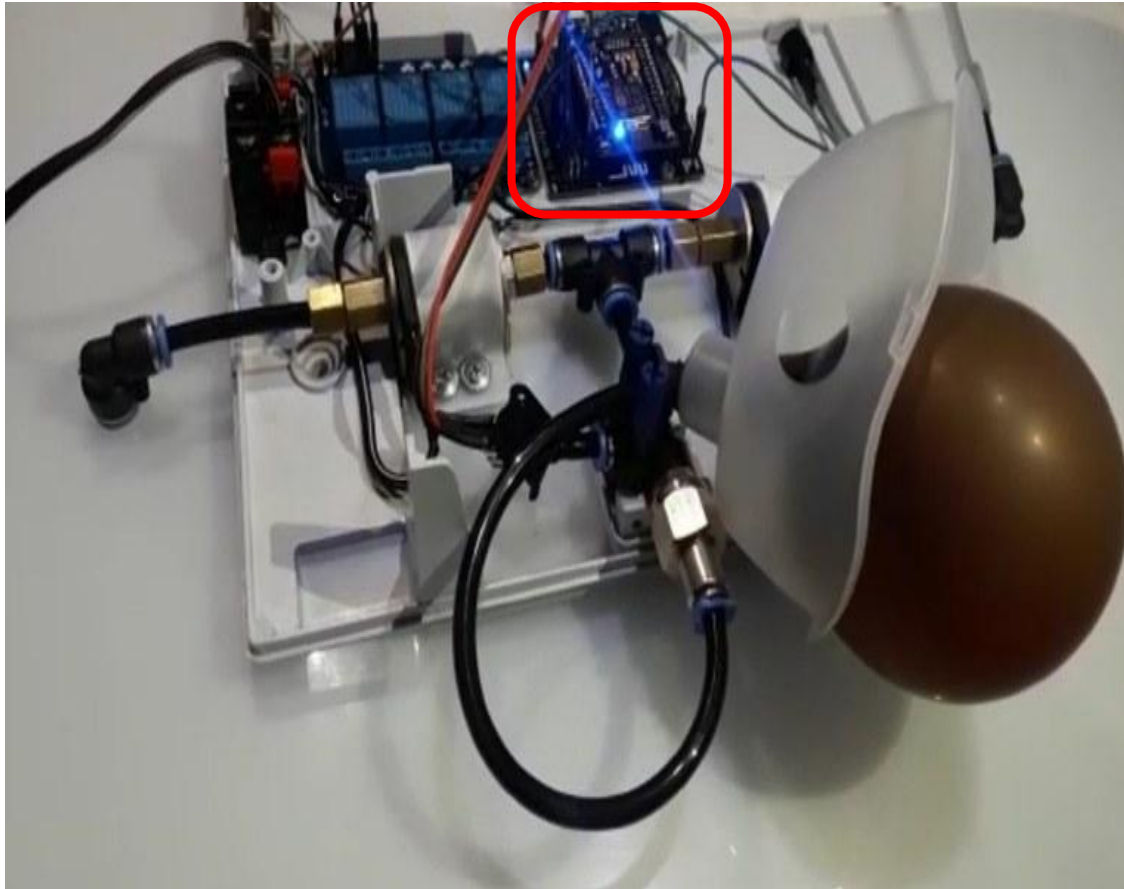
```
1 const int sensor = 2;
2 const int LEDam = 13;
3 const int LEDvm= 12;
4 const int som= 11;
5 int u=0;
6 int sensor_valor = 0;
7
8 void setup() {
9     pinMode(LEDam, OUTPUT);
10    pinMode(LEDvm, OUTPUT);
11    pinMode(som, OUTPUT);
12    Serial.begin(9600);
13 }
14 void loop() {
15     sensor_valor = analogRead(sensor);
16     Serial.print(sensor_valor);
17     Serial.print("\n");
18     if (sensor_valor>1000) {
19         u=1;
20     }if (u==1){
21         tone(som, 2000);
22         digitalWrite(LEDam, HIGH);
23         delay(100);
24         digitalWrite(LEDam, LOW);
25         digitalWrite(LEDvm, HIGH);
26         delay(100);
27         digitalWrite(LEDvm, LOW);
28     }Serial.flush();
29 }
```

Compilação terminada.

O sketch usa 3080 bytes (9%) de espaço de armazenamento para programas. O máximo são 32256 bytes.  
Variáveis globais usam 209 bytes (10%) de memória dinâmica, deixando 1839 bytes para variáveis locais. O máximo são 2048 bytes.



## Respirador Pulmonar da UFPB



“O equipamento também é de rápida montagem e programação, sendo possível operá-lo em 60 segundos. Outro detalhe é que ele não é um respirador de emergência, podendo ser usado indefinidamente, ou seja, um substituto aos convencionais comercializados atualmente.”

Reportagem disponível no site da UFPB notícias:

<https://www.ufpb.br/ufpb/contents/noticias/respirador-pulmonar-da-ufpb-tem-licenca-liberada-para-producao-por-empresas>

# Obrigado!

Como forma de atender a todas as dúvidas, segue abaixo o e-mail dos envolvidos no desenvolvimento desse vídeo:

- miguel.ferreira@cear.ufpb.br
- maria.silva@cear.ufpb.br
- josman.rodriques@cear.ufpb.br
- gabriela.cavalcanti@cear.ufpb.br
- pedro.morais@cear.ufpb.br
- vitor.franca@cear.ufpb.br
- rubens.lima@cear.ufpb.br
- lucas.fernandes@cear.ufpb.br

## Créditos:

- Modelo de Apresentação por Slidego
- Ícones por Flaticon
- Infográficos e Imagens por Freepik
- Conteúdo por Maria Heloisa e Miguel Marques

