

# Curso de Arduino

Prof Neri Aldoir Neitzke





# O que é Arduino

É uma plataforma de computação física com um microcontrolador de placa única com suporte embutido de entrada e saída. Para facilitar o entendimento, podemos até dizer que Arduino é um computador. Nós podemos criar programas para ele que manipulem entradas e saídas. É algo incrível, e você poderá interagir com o meio ambiente e criar o que quiser, basta ter entusiasmo, criatividade e imaginação.





# Arduino para todos

É fantástico o fato de você poder criar um projeto envolvendo eletrônica sem precisar dominar eletrônica (não precisa ser engenheiro) nem precisar criar um circuito, isso abre portas a um novo mundo de pessoas criativas e com potencial para desenvolver projetos magníficos e baratos.

Não é preciso começar do zero.





# Arduino

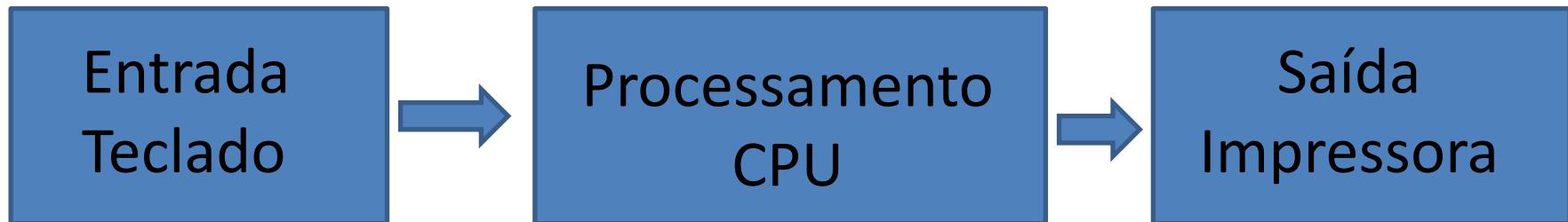
Palavra chave = brincar

Isso mesmo, você aprende  
brincando, testando, se  
deliciando.



# Para entender bem

Computador



Arduino





# O que podemos conectar e controlar

Com Arduino é possível conectar qualquer dispositivo que emita dados ou possa ser controlado. Entre eles são: LED's, Displays, interruptores, motores, sensores de Luz, sensores de Temperatura, sensores de Umidade, sensores de Pressão, sensores de distância, sensores para nível de álcool, receptores de GPS, módulos Ethernet etc.





# Surgimento

Surgiu na Itália em 2005 com o objetivo de criar um dispositivo prático e simples no controle de entrada e saída de dados.

Característica: Multi-Plataforma





# Interagir com o ambiente

Isso é uma das coisas mais fantásticas que o Arduino proporciona: interação com o meio ambiente. Mas como isso é possível ? O

Arduino (circuito eletrônico) consegue perceber o que acontece no seu ambiente através de sensores que convertem medidas do mundo real em sinais elétricos)





# Lixo Eletrônico (tralhas)

Motores, alto-falantes, fios, resistores, etc etc.. Tudo isso você pode aproveitar daquela impressora ou daquele gabinete velho de computador que você iria jogar fora e não sabe onde colocar. Aproveitem o lixo eletrônico, basta usar sua imaginação .....



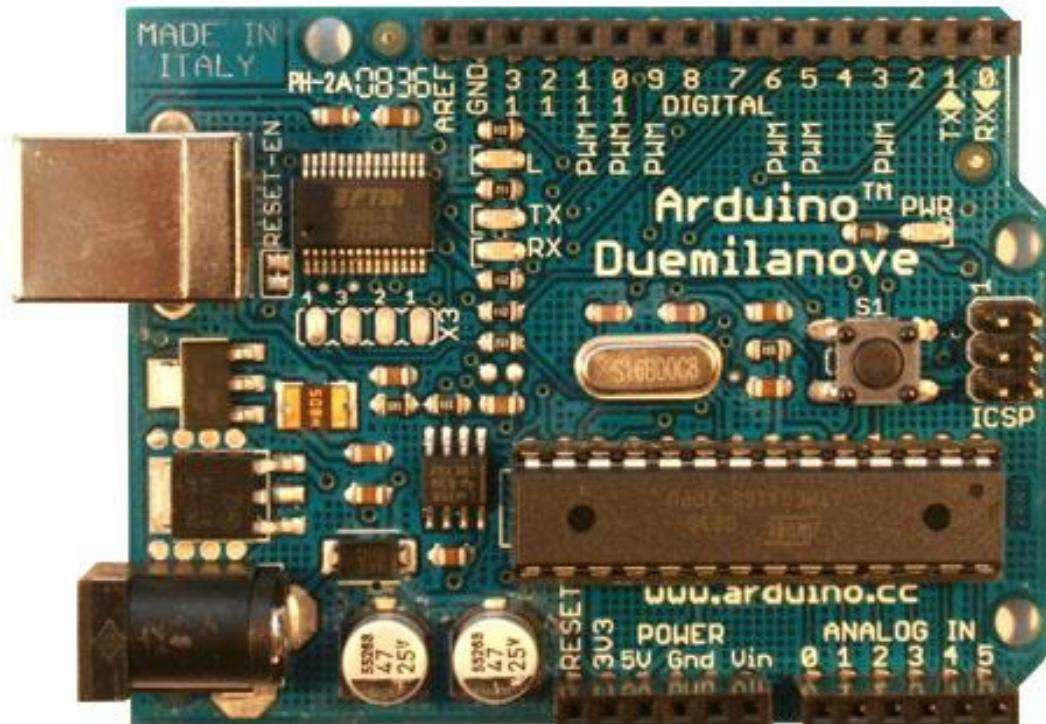
# Arduino: Hardware e Software de fonte aberta

O projeto, fontes e esquemas podem ser usados de forma livre. Você pode criar em casa sua própria placa Arduino baseado no diagrama de circuito do projeto original. Só não pode colocar o nome Arduino, mas pode ser Neriuno, Informaticonino, Nerizonino, videoaulasino ... etcino



# Placa Arduino Duemilanove

Versão Anterior, ainda muito usada



# Placa Arduino Uno R3

Mais recente e mais popular



Pinos ligados no soquete de Circuito Integrado

Usa um Chip Atmega8U2

Vantagens:

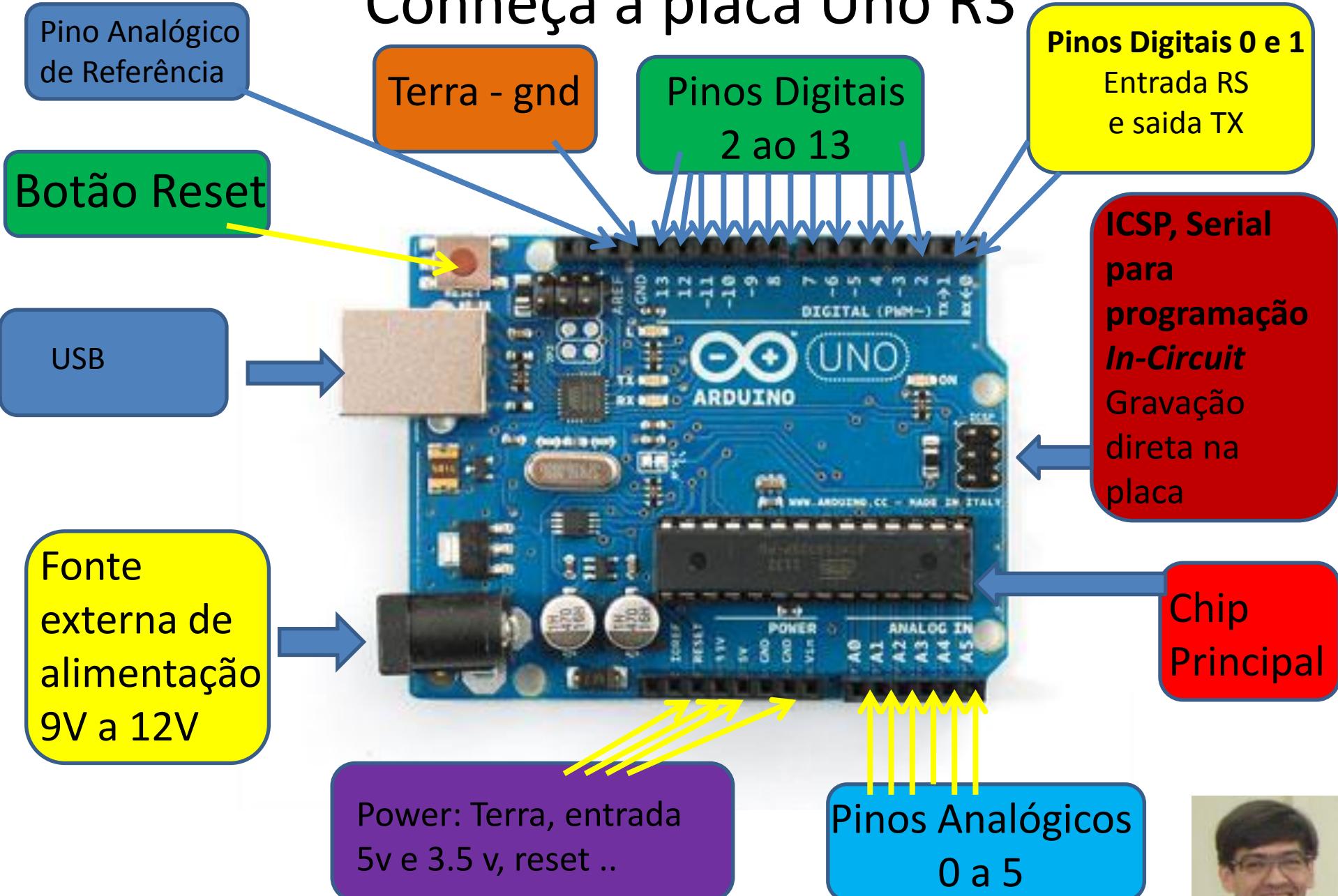
- mais barato
- Arduino é exibido no Pc como se fosse qualquer outro dispositivo
- Pode deixar permanente

Desvantagens:

- Difícil de ser clonado



# Conheça a placa Uno R3



# Placa Arduino Uno R3

Pinos Digitais  
02 à 13

**Pinos digitais:** usa-se para entradas e saídas gerais. Alguns comandos são: pinMode(), digitalRead(), e digitalWrite().



Pinos Analógicos  
01 à 05

**Pino 13:** Nele tem um LED integrado, você poderá programar para ligar (high) ou desligar (low). Nele já existe um resistor, ou seja, você poderá testar uma aplicação usando o pino 13.

**Pinos Analógicos:** Recebem valores analógicos (sensores) mas pode ser configurado e utilizado da mesma maneira que os pinos digitais.



# Primeiro Exemplo

```
const int ledPin = 13; //variável do tipo inteiro com valor 13
```

```
void setup() //função ou método que é executado toda vez
```

```
{
```

```
  pinMode(ledPin,OUTPUT);
```

```
}
```

```
void loop()
```

```
{
```

```
  digitalWrite(ledPin,HIGH); //acende a luz (envia corrente)
```

```
  delay(1000); //espera 2 segundos
```

```
  digitalWrite(ledPin,LOW); //apaga a luz (não envia corrente)
```

```
  delay(1000); //espera 2 segundos
```

```
}
```



# Linguagem

```
const int ledPin = 13; //variável do tipo inteiro com valor 13
```

```
void setup() //função ou método que é executado toda vez
```

```
{
```

```
  pinMode(ledPin,OUTPUT);
```

```
}
```

```
void loop()
```

```
{
```

```
  digitalWrite(ledPin,HIGH); //acende a luz (envia corrente)
```

```
  delay(1000); //espera 2 segundos
```

```
  digitalWrite(ledPin,LOW); //apaga a luz (não envia corrente)
```

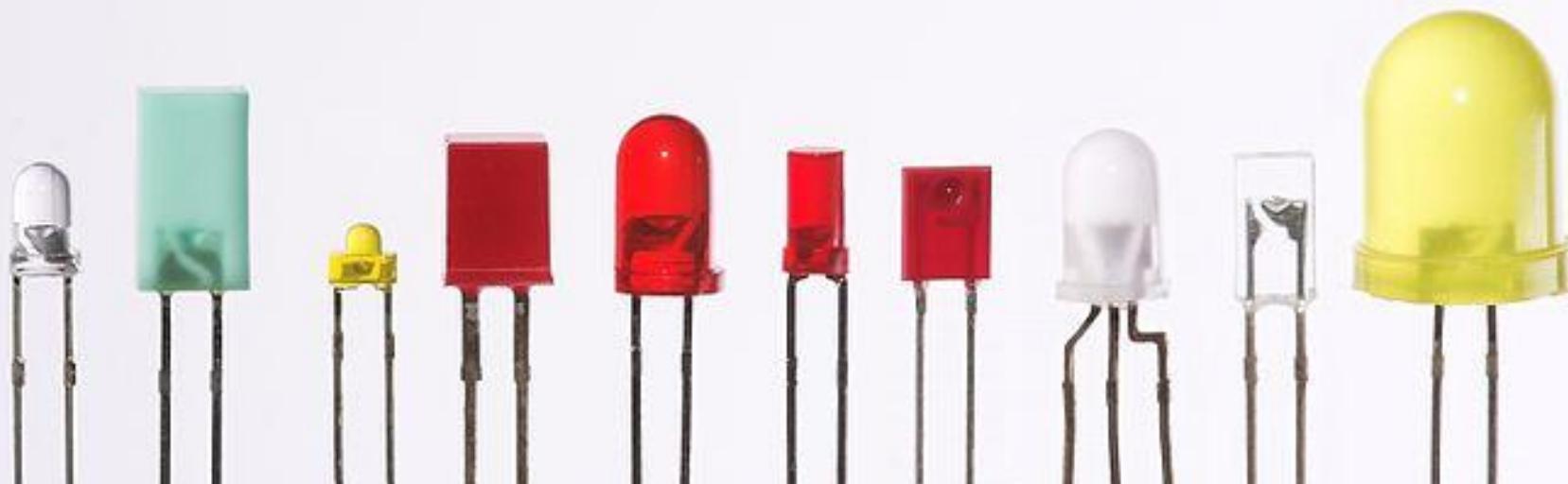
```
  delay(1000); //espera 2 segundos
```

```
}
```



# LED - Light Emitting Diode

Diodo Emissor de Luz, usado muito para sinalizadores, carros, semáforo, painéis (avião), informações de ligado, desligado, processando (piscando)



# LED - Light Emitting Diode

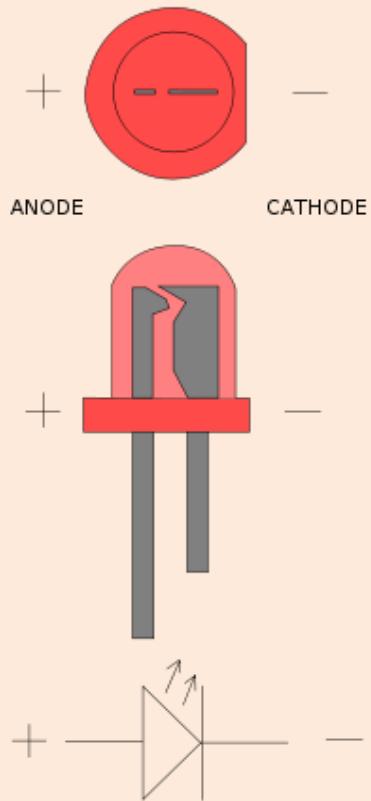


LED



# LED - Light Emitting Diode

Para funcionar deverá estar polarizado (Negativo e positivo)



# LED – E a voltagem ?

A maioria dos LED'S encontrados possui uma tensão direta entre 1,5 e 3,5 V, variando conforme suas características , por exemplo: os de cores vermelhas apresentam tensão direta menor do que os verdes e Amarelos.



# LED – E a Corrente?

Normalmente os LED'S apresentam uma corrente máxima direta entre 40mA e 50 mA (mas com apenas 5 mA, por exemplo a luminosidade já é bem perceptível), nunca ultrapasse o valor máximo pois uma corrente maior do que o LED pode suportar, irá sobreaquecer e poderá queimá-lo (eu já queimei muitos). A luminosidade do LED é diretamente proporcional à corrente do mesmo, portanto, quanto maior a corrente, mais intensa a luz é emitida (mas sempre com tal corrente dentro do máximo permitido). Embora a corrente máxima ( supondo em torno de 40 mA ), ocasione luminosidade também máxima, isso não quer dizer que o LED não acenda com correntes menores. O componente começa a emitir luz com corrente de uns poucos milliampères.



# Resistor



# Resistor

Provoca resistência a uma corrente elétrica, diminuindo voltagem.

O valor da resistência é medido em OHM e seu símbolo é o Ômega Grego

 $\Omega$ 

$\Omega$ ,  $k\Omega$  ou  $M\Omega$  comparando com a informática Byte, Kilobyte, Megabyte

1 ohm = 1 byte

1.000 ohm = 1 k  $\rightarrow$  1.000 Bytes = 1 k

1.000.000 ohm = 1.000 k  $\Omega$  = 1 M  $\Omega$   $\rightarrow$  1.000.000 Bytes = 1.000 K = 1 Megabyte



# Resistor - Cálculo

$$R = (VF - VL) / CML$$

R = Resistor Necessário

VF = Voltagem Fornecida (Pilha, Bateria, USB)

VL = Voltagem do Led

CML = Corrente Máxima do Led

Exemplo.: Para um LED de 5mm e que necessita 2V e trabalha com corrente de 35 milliampères, qual é o resistor necessário ?



# Resistor - Cálculo

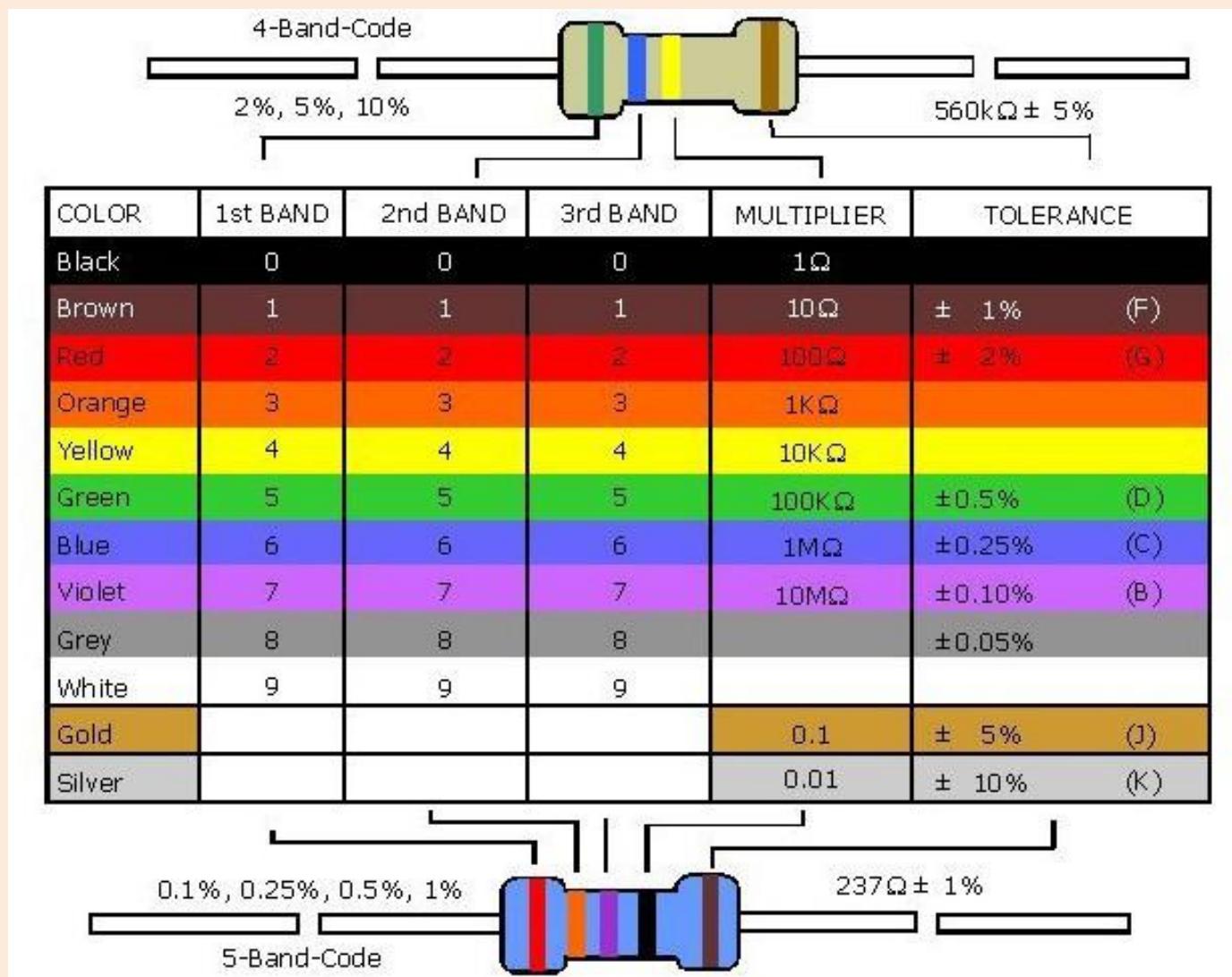
Exemplo.: Para um LED de 5mm e que necessita 2V e trabalha com corrente de 35 milliampères, qual é o resistor necessário ? Vamos aplicar a fórmula:

$$R = (VF - VL) / CML$$

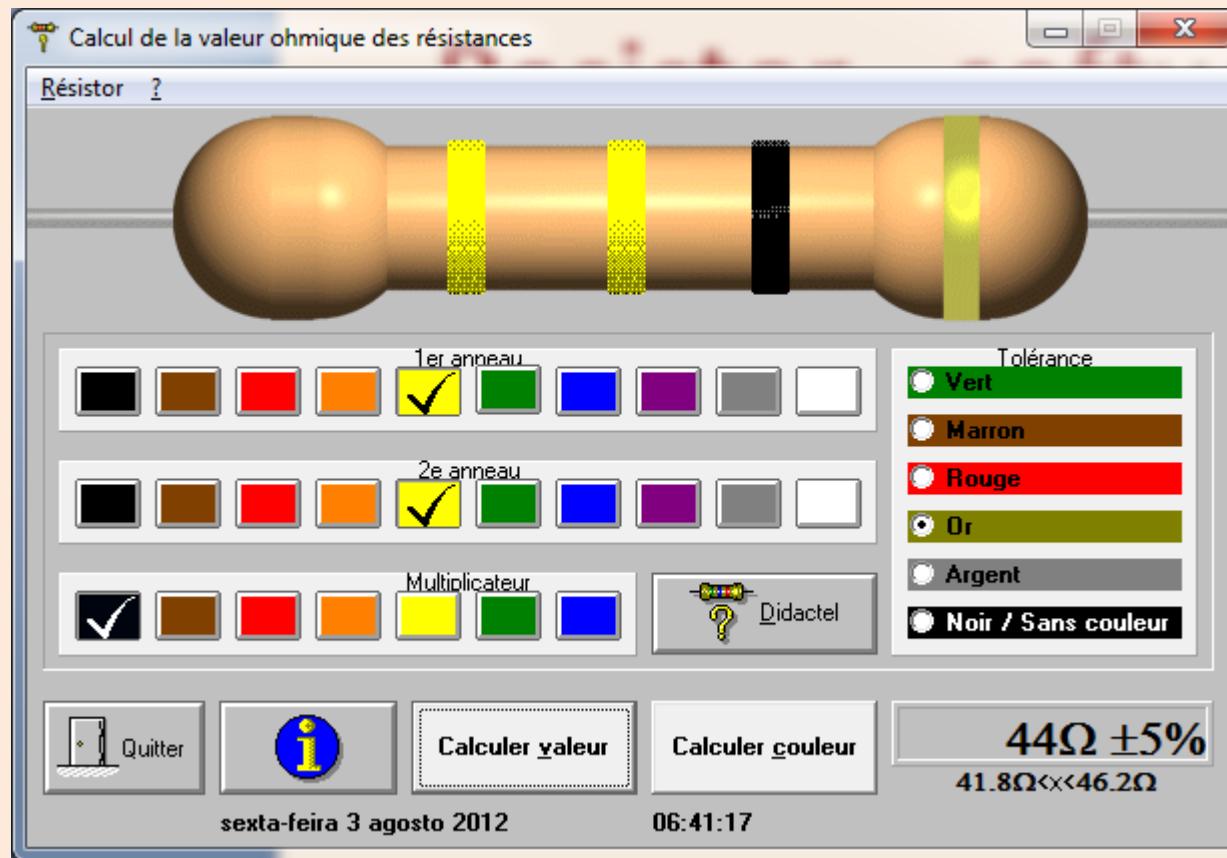
$$R = (5 - 2) / 0.035 = 85 \text{ ohm} \text{ (mais próximo é o 100 ohm)}$$



# Resistor



# Resistor - softwares



# Motores

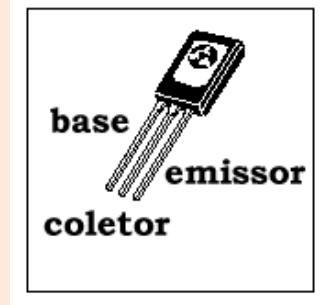
Motor de Corrente Contínua (motor CC)

Velocidade variável e você pode controlar a velocidade





# Transistor

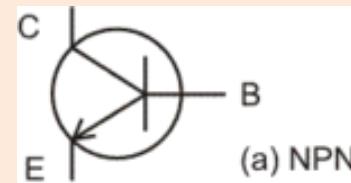


Considerado o principal responsável pela revolução da [eletrônica](#) na [década de 1960](#)

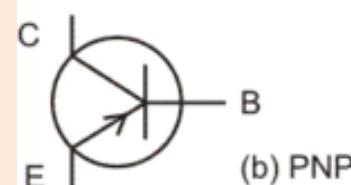
O transistor é considerado uma das maiores invenções da história moderna, tendo tornado possível a revolução dos [computadores](#) e equipamentos eletrônicos.

Amplificador de corrente (sinal elétrico mais forte ou mais fraco) podendo então  
Regular a Tensão, Amplificador de sinal (exemplo do Microfone, aparelhos auditivos)  
Chave eletrônica..

Resumindo: você poderá controlar componentes que  
Exigem mais tensão ou corrente que o Arduino  
Consegue prover.



(a) NPN

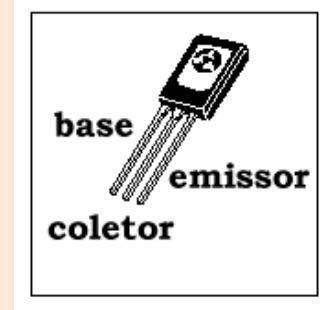


(b) PNP





# Transistor

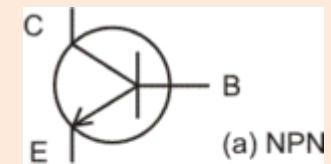


Eu estou usando o TIP 120

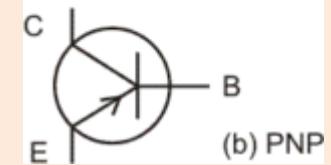
**Base:** é a parte que controla a passagem da corrente

**Coletor:** é nele que “entra” a corrente a ser controlada.

**Emissor:** é onde sai a corrente que foi controlada.



(a) NPN



(b) PNP



# Diodo

Retificação de corrente

