

# Interfaces físicas para dispositivos móveis

Felipe Navarro Balbino Alves

# Interface homem máquina

Como nós enxergamos um computador?



O conjunto

**Monitor Mouse Teclado**

Nos permite identificar facilmente um computador

# Interface homem máquina

Como o computador nos enxergaria?



**Dedos**

[teclado/mouse]

**Olho**

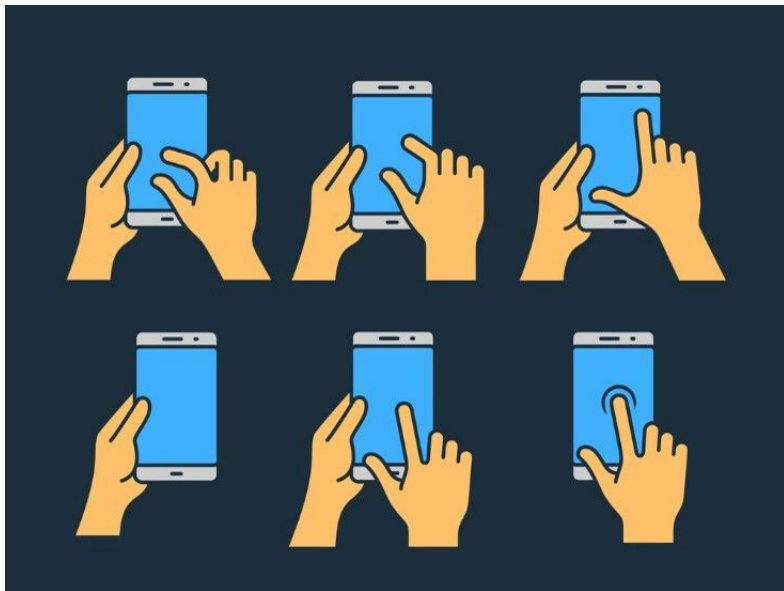
[monitor]

**Duas orelhas**

[caixa de som]

# Interface homem máquina

No entanto, os computadores começaram a mudar, e, conseqüentemente, a forma como interagimos com os mesmos



# Interface homem máquina

O surgimento de telas touch screen não se restringiu aos smartphones....



Ainda assim, a forma como interagimos com nossos dispositivos ainda pode evoluir bastante...

# Computação física

“Mudar a forma como os computadores nos veem mudará como eles interagem conosco”

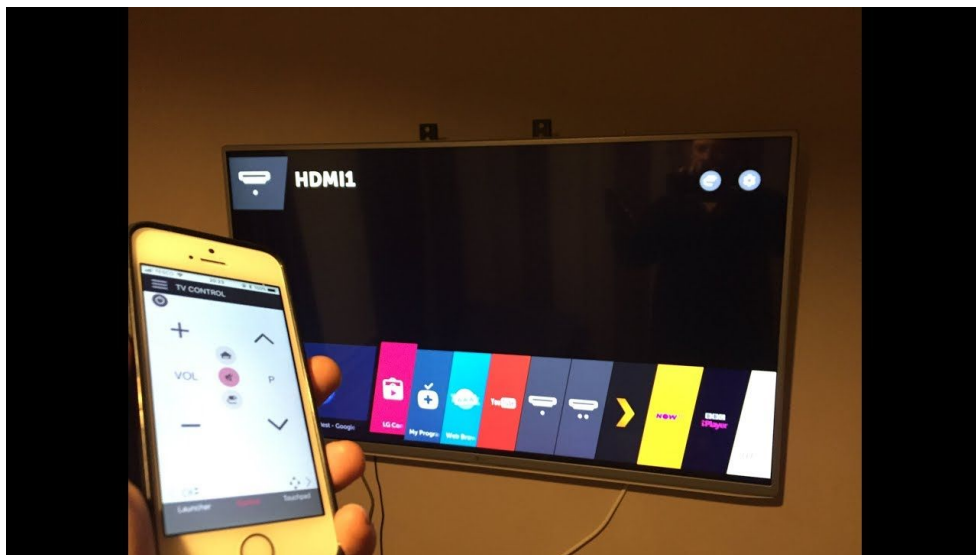
Tom Igoe - Physical Computing

**Xbox accessible  
controller**



# Computação física

A computação física também pode mudar a forma como interagimos com o próprio ambiente



# A abordagem deste curso

Ao longo deste curso, empregaremos conhecimentos de computação e eletrônica para desenvolvermos aplicações de computação física

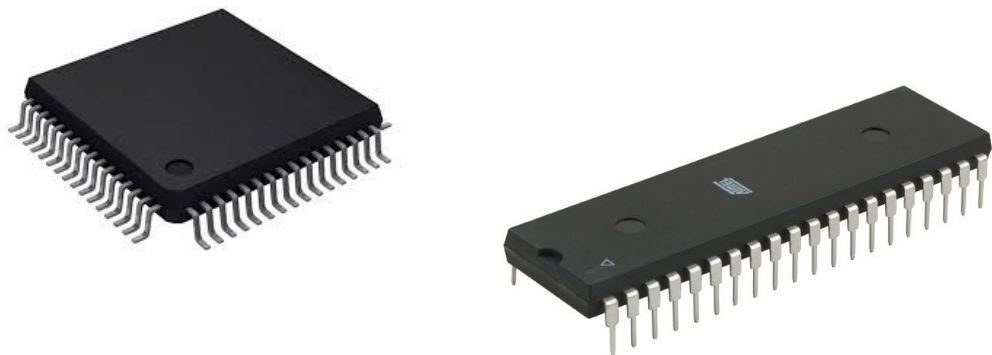
**Computação:** Comportamento definido por software

**Eletrônica:** Uso de microcontroladores e circuitos em geral

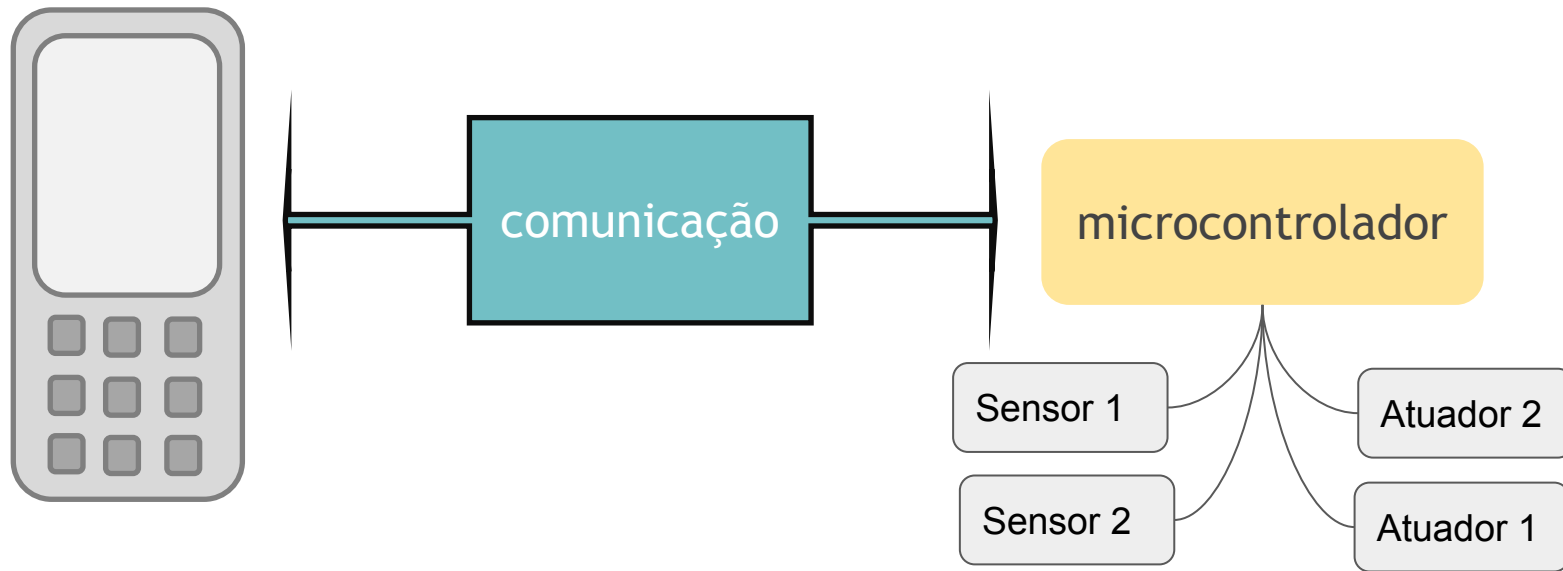


# O que são microcontroladores (MCU) ?

Microcontroladores são pequenos computadores em circuito integrado com um **número reduzido de instruções** além de um conjunto de **periféricos** (ex: Memória, interfaces de comunicação, etc...). São utilizados em sistemas computacionais com **uso específico**.



# Arquitetura das soluções desenvolvidas



# Arquitetura das soluções desenvolvidas

Conectados ao microcontrolador estão

**Sensores:** Responsáveis por perceber o ambiente e alterações no mesmo



Presença



Luz



Som

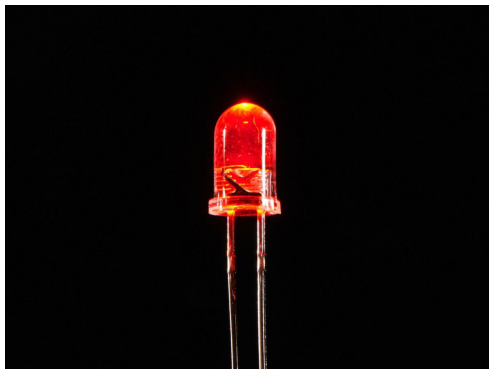


Temperatura /  
Umidade

# Arquitetura das soluções desenvolvidas

Conectados ao microcontrolador estão

**Atuadores:** Responsáveis por criar alterações no ambiente



Luz



Movimento



Som

# Arquitetura das soluções desenvolvidas

## Smartphone

Será utilizado o sistema Android



# Arquitetura das soluções desenvolvidas

## Microcontrolador

Utilizaremos neste curso a plataforma **Arduino**



# A plataforma arduino

É uma das plataformas de prototipagem mais conhecidas da atualidade. Possui **software e hardware livres** e **baixo custo**. O principal objetivo destas plataformas é facilitar o desenvolvimento de microcontroladores através do uso de uma **única placa** e um pacote de software para desenvolvimento

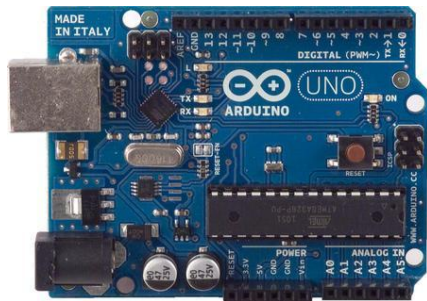


# A plataforma arduino

**Microcontrolador:** Atmel

**Linguagem de programação:** Wiring (subconjunto do processing, baseado em C/C++)

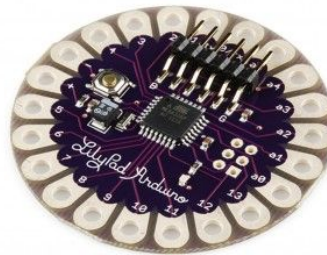
**Diversos tipos:**



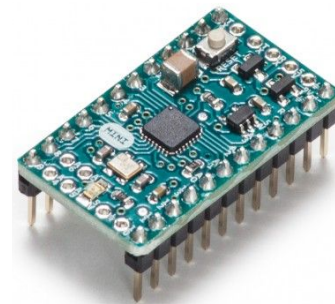
Arduino Uno



Arduino Mega



Lilypad



Arduino mini



# A plataforma arduino

## **Portas:**

14 entradas / saídas digitais

6 entradas analógicas

## **Memória:**

RAM (1k)

Flash (programa): 16k - 2k (bootloader)

**Clock (velocidade de processamento) : 16 Mhz**

# A plataforma arduino

## Utilizando o IDE do Arduino:

Instale o software correspondente ao seu sistema

<https://www.arduino.cc/en/main/software>

No caso do Windows, utilize o **Instalador** para que os drivers sejam instalados automaticamente

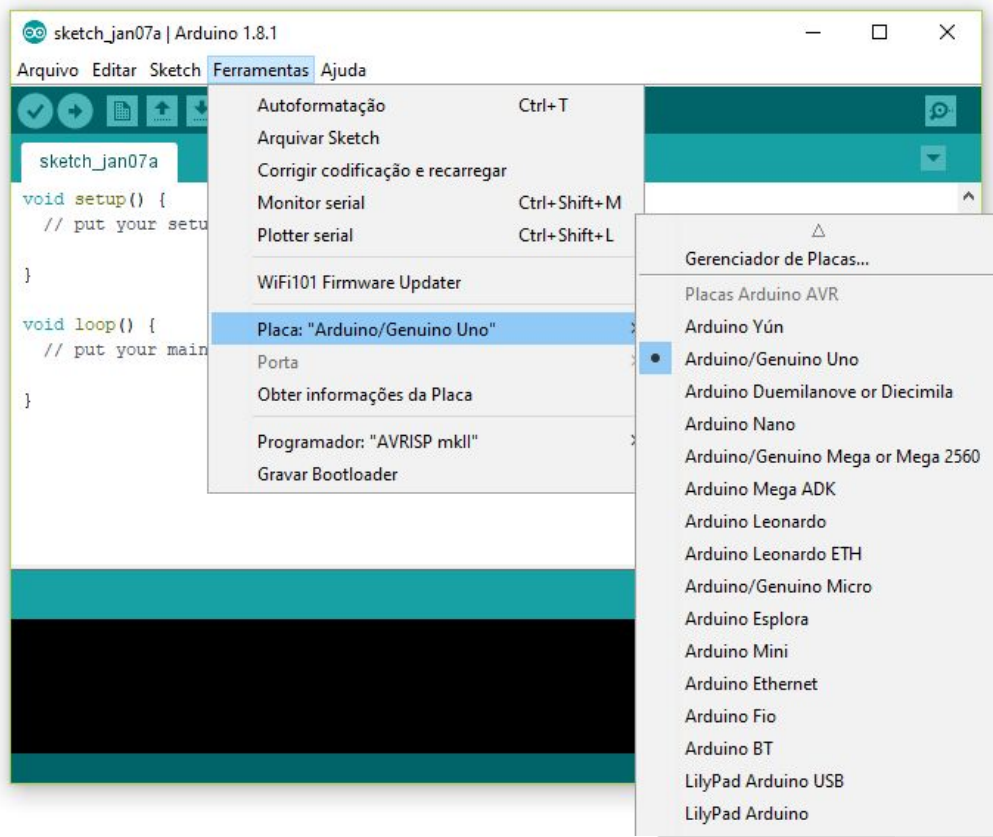
Conecte o Arduino no computador

# A plataforma arduino

Utilizando o IDE do Arduino:

Selecionando a placa e

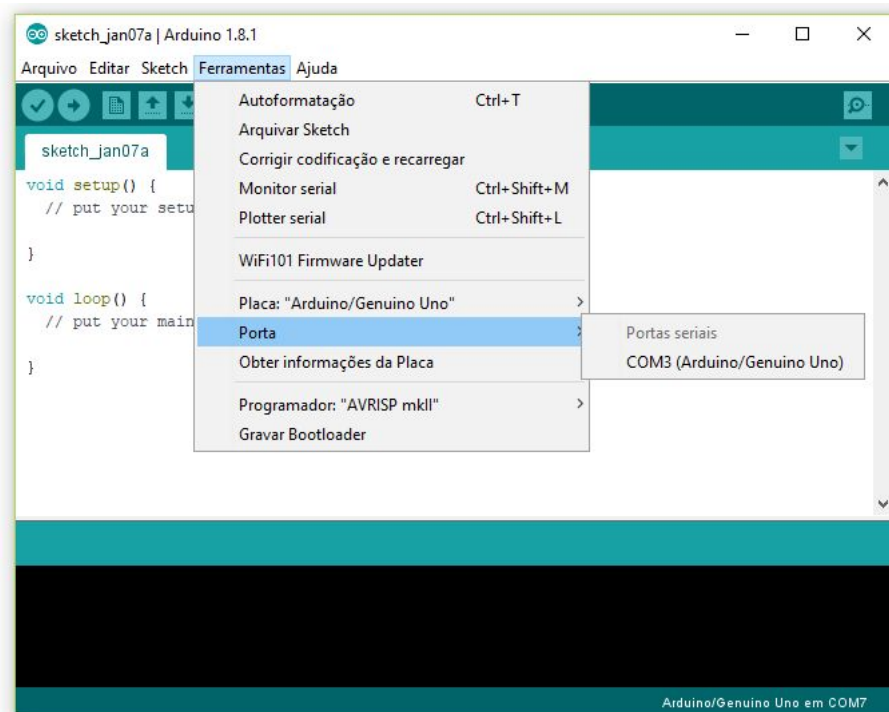
a porta serial



# A plataforma arduino

## Utilizando o IDE do Arduino:

Selecionando a placa e  
a porta serial



# A plataforma arduino

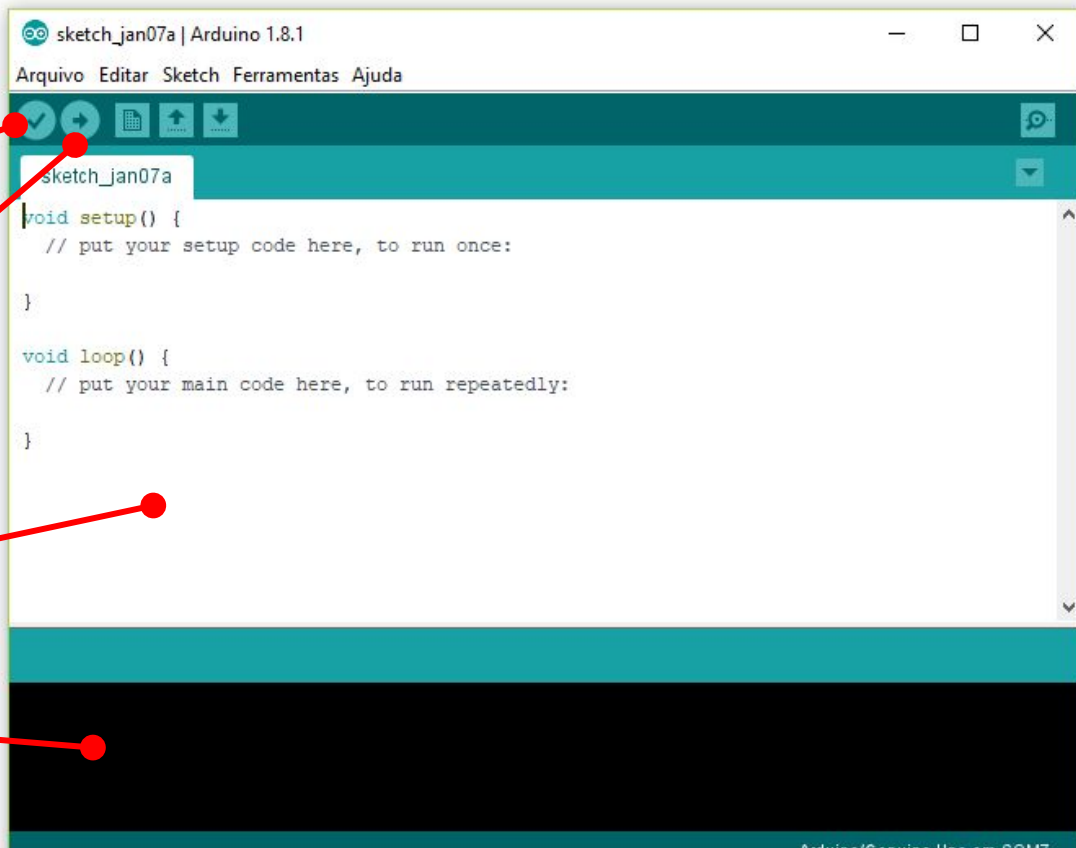
## Utilizando o IDE do Arduino:

Compilar

Fazer  
upload  
na placa

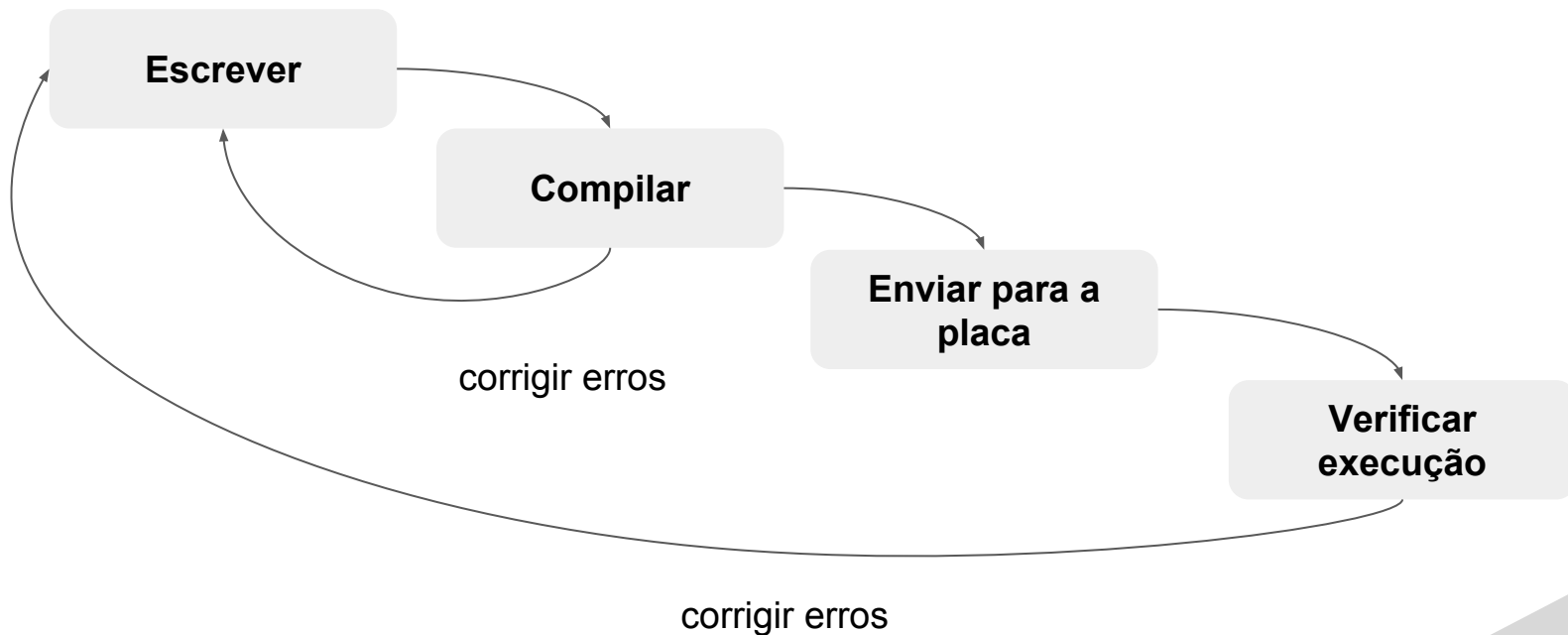
Área do código

Status



# A plataforma arduino

## Workflow arduino

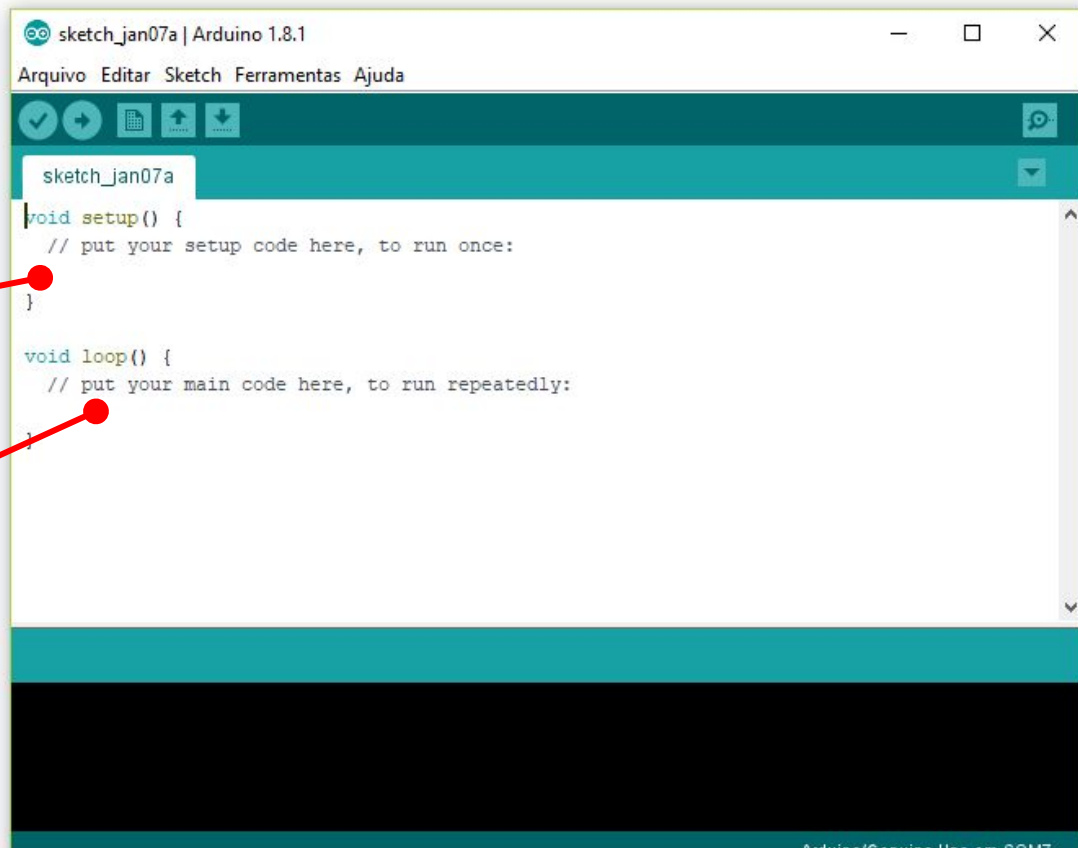


# A plataforma arduino

## Estrutura do código

Código executado apenas uma vez para configurar os periféricos

Loop principal -  
Leitura de sensores,  
comunicação de  
dados, etc...



The screenshot shows the Arduino IDE interface with a sketch named 'sketch\_jan07a'. The code is as follows:

```

void setup() {
  // put your setup code here, to run once:
}

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

Two red lines with circular endpoints point from the text boxes on the left to the code. The first line points to the `void setup()` block, and the second line points to the `void loop()` block.

# A plataforma arduino

## Alguns comandos introdutórios

**pinMode(*numero\_pino*, *modo*)** - configura um pino como entrada (INPUT) ou saída (OUTPUT);

**digitalWrite(*numero\_pino*, *valor*)** - desliga (LOW) ou liga (HIGH) uma saída digital;

**digitalRead(*numero\_pino*)** - lê uma entrada digital;

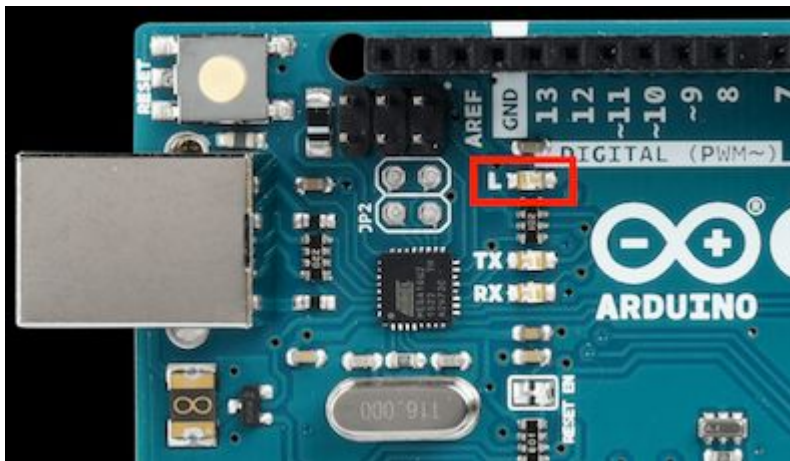
**delay(*milissegundos*)** - espera um determinado tempo em milissegundos;



# A plataforma arduino

## Atividade prática 1- Acender e apagar um LED

Vamos utilizar o LED built-in na placa - ligado ao pino 13 no Arduino Uno



Apresentação  
de código

# A plataforma arduino

**Atividade prática 2** - Faça o Led piscar periodicamente

**Atividade prática 3** - Faça o Led acender em meia potência

**NOW IT'S  
*YOUR* TURN.**

# A plataforma arduino

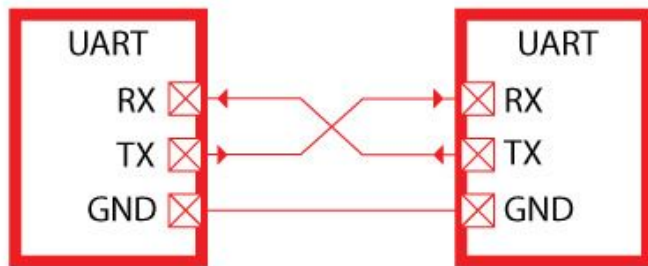
## Pacote serial

O microcontrolador Atmega não possui comunicação USB. Em vez disso, ele possui uma interface serial UART (Universal Asynchronous Receiver / Transmitter):

# A plataforma arduino

## Comunicação serial

O microcontrolador Atmega não possui comunicação USB. Em vez disso, ele possui uma interface serial UART (Universal Asynchronous Receiver / Transmitter):



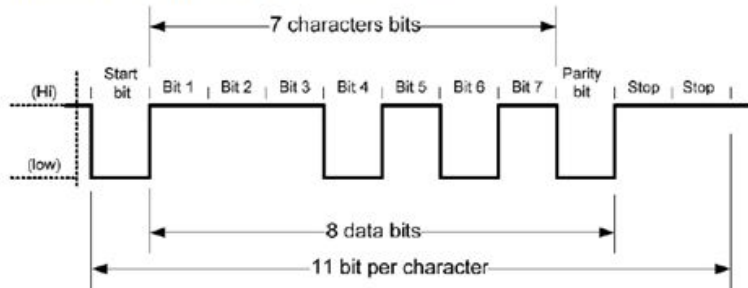
# A plataforma arduino

## Comunicação serial

Uma frame de informação serial possui um start bit (baixo lógico), um ou nenhum bit de paridade e um ou mais stop bits (alto lógico)



Figure 14-3. Character ASCII "W" serial transmission.



# A plataforma arduino

## Comunicação serial

É importante definir o **Baudrate** da comunicação, que é a taxa de bits por segundo.

Valores comumente usados:

9600

115200

etc...

# A plataforma arduino

## Comunicação USB no Arduino

Para que o Arduino se comunique via USB com o computador, existe um chip que converte a comunicação serial para uma comunicação USB (chip FTDI). Esta comunicação é usada para a gravação do código do Arduino e comunicação em geral com o computador



# A plataforma arduino

## Comunicação USB no Arduino

Para utilizar o módulo serial, utilizaremos as seguintes funções:

**Serial.begin(*baudrate*)** - inicia o módulo serial com um baudrate específico

**Serial.print(*string*)** - Envia dados pela serial

**Serial.println(*string*)** - Envia dados pela serial com uma quebra de linha no final

**Serial.read()** - Lê um caractere

**Serial.available()** - Verifica se existem dados a serem lidos



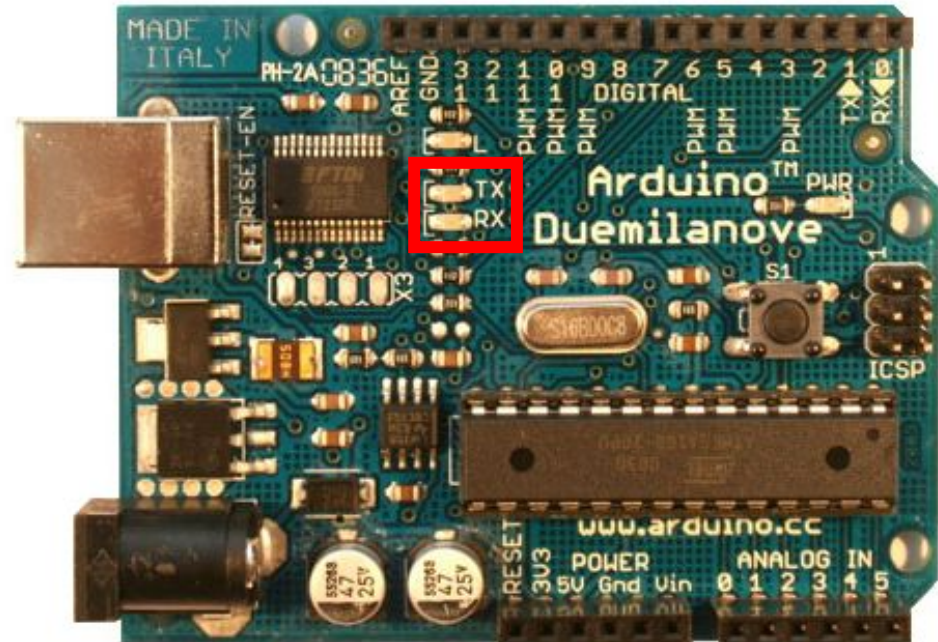
# A plataforma arduino

## Comunicação USB no Arduino

Leds

**Tx:** Dados enviados para o PC

**Rx:** Dados enviados pelo PC



# A plataforma arduino

## Atividades práticas

**Atividade prática 4** - Enviar dados para o PC e observar no monitor serial

**Atividade prática 5** - Enviar dados do PC para o arduino

# A plataforma arduino

**Atividade prática 6** - Faça o Led acender ou apagar a partir de um comando da serial

**NOW IT'S  
*YOUR* TURN.**