



CEFET – RJ / *Campus* Maria da Graça
Centro Federal de Educação Tecnológica
Celso Suckow da Fonseca – Rio de Janeiro



Prof. Cristiano Fuschilo

cristiano.fuschilo@cefet-rj.br

ROBÓTICA - 2022_1 T.OPTCSINF -
(SIOP0005)



Bacharelado em
Sistemas de
Informação



COORDENAÇÃO DE
**Automação
Industrial**
Ensino de Qualidade



computação física



computação física



- uso de computação e eletrônica [sensores e atuadores] na prototipação de objetos físicos para interação com seres humanos
- comportamento implementado por software
- utilização de microcontroladores



computação física



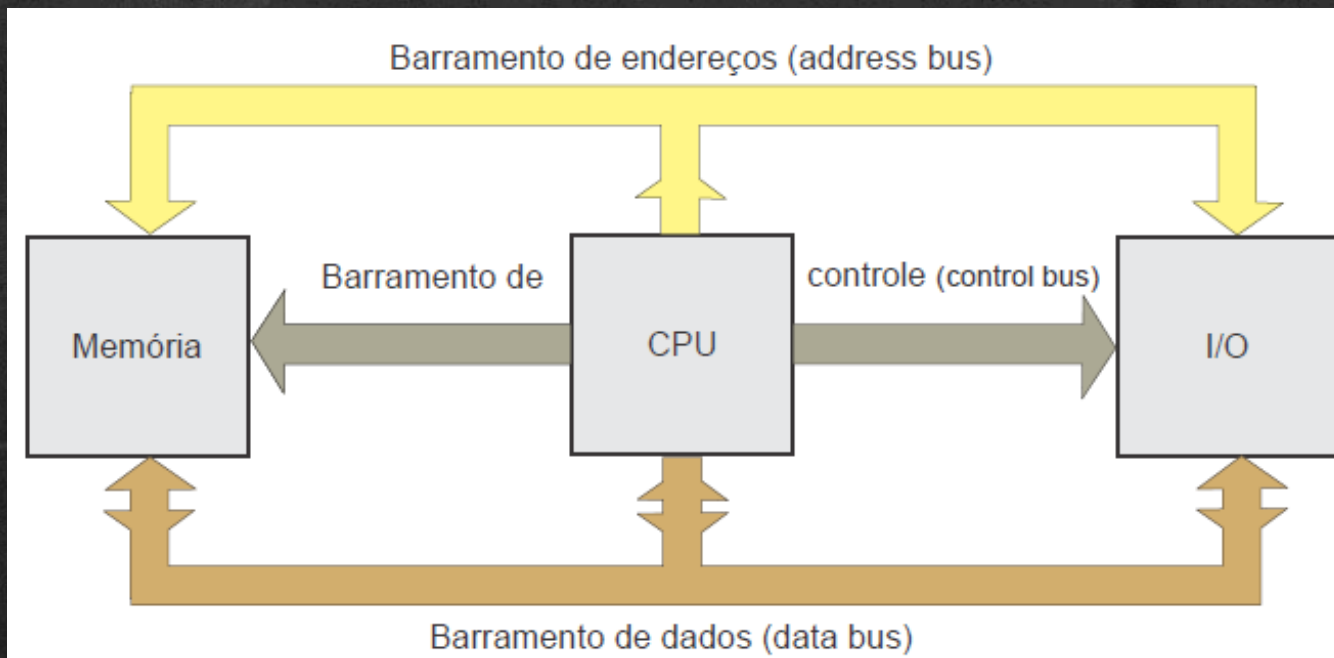
- o objetivo é interligar o mundo físico com o mundo virtual
- usar a computação e a interação com a tecnologia para o desenvolvimento das suas atividades
- meio para comunicação e interação entre pessoas



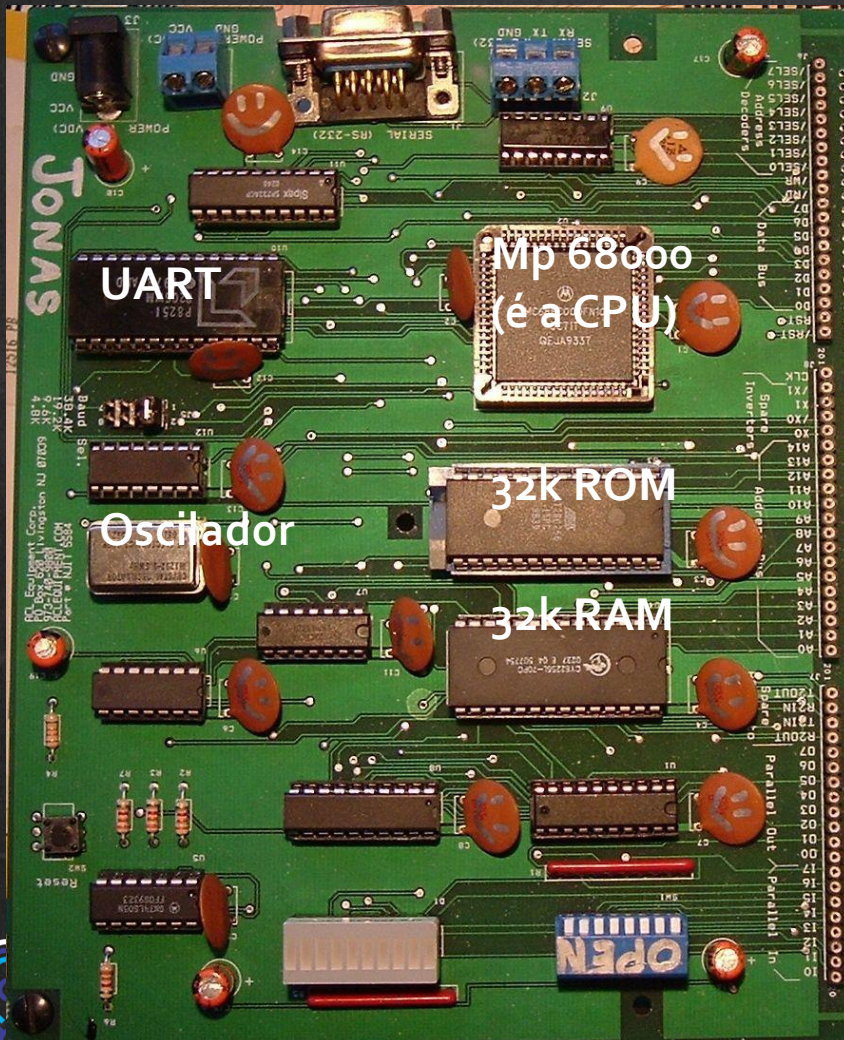
computador



- Um computador é um sistema digital que possui uma CPU (central processing unit), memória e portas de entrada e saída (I/O). Estes três módulos comunicam-se através dos barramentos de dados, endereços e controle.



Microprocessador



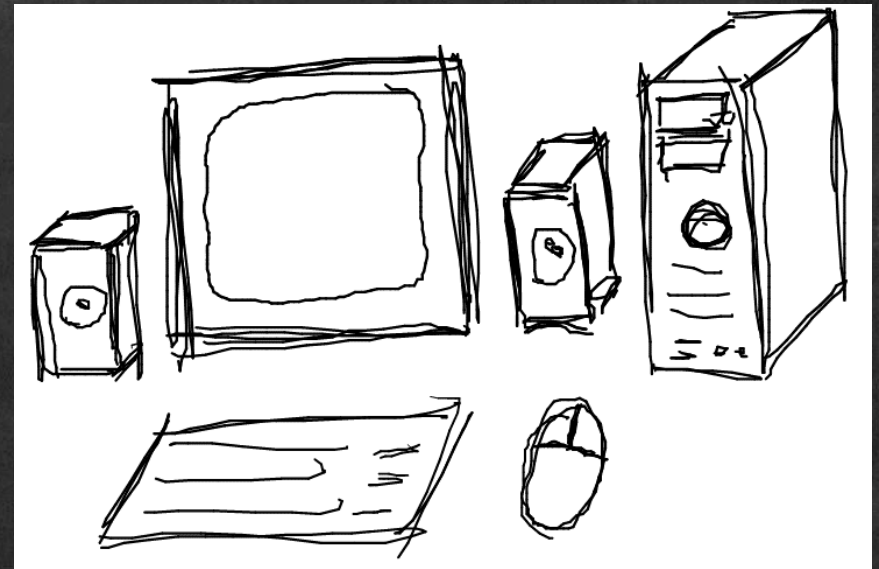
- Se construído com um Microprocessador, o computador do exemplo necessitaria CIs externos de RAM, ROM, UART e Oscilador.



Computação física Como vemos os computadores?



- teclado
- mouse
- monitor
- CPU



caixas de som

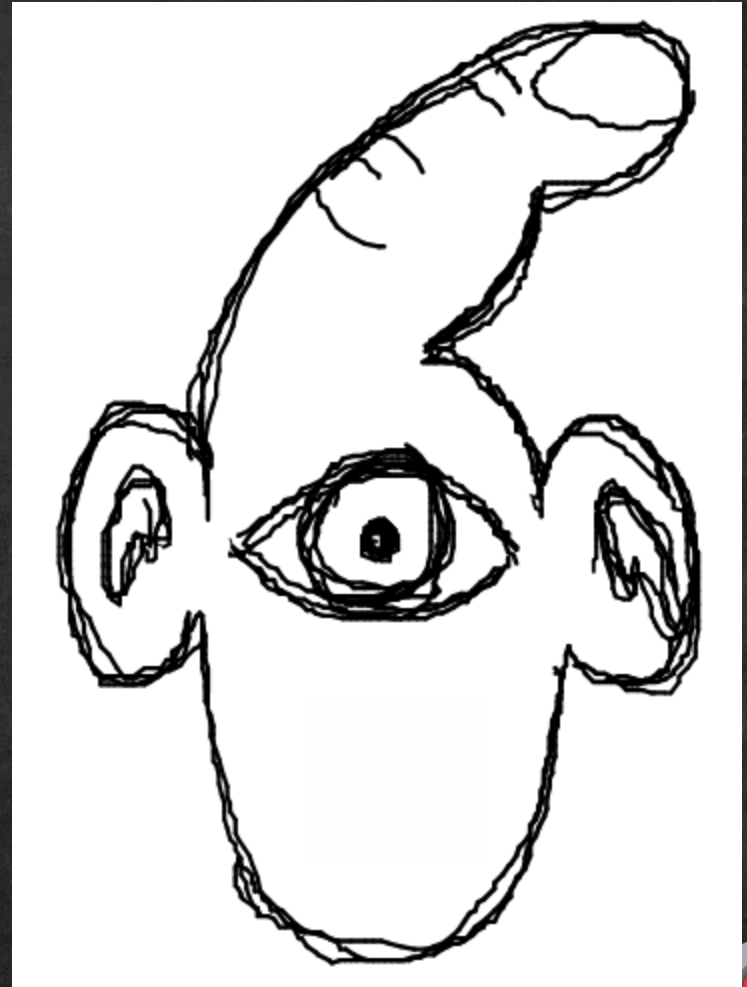


Computação física

Como os computadores nos veem?



- dedos
[teclado/mouse]
- olho
[monitor]
- duas orelhas
[caixas de som]
- reflexo das entradas e saídas
do computador



computação física



- “mudar a forma que os computadores nos veem
mudará como eles interagem conosco”
- Tom Igoe – Physical Computing



computação física



- “mudar a forma que os computadores nos veem mudará como eles interagem conosco”
- ...através de elementos físicos de interação mais adequados às interfaces humanas





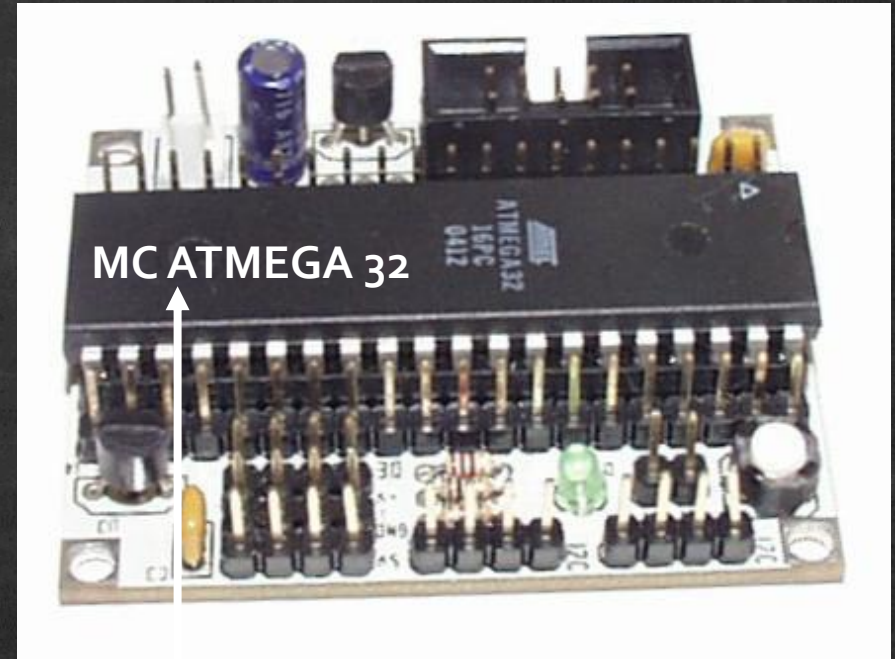
plataforma Arduino



Microcontrolador (Mc)



- Um Mc possui RAM, ROM, UART e Oscilador embutidos.
- Por isso, um Mc também é chamado de single-chip computer. CPU
- Há muitas opções de recursos embutidos, também chamados de periféricos embutidos ou periféricos integrados.



CPU / ROM / RAM /
UART / Oscilador



Arduino e similares



- “Computação física (physical computing): conjunto de ferramentas e métodos para construir sistemas digitais capazes sensorar e atuar no mundo físico, mais efetivamente que um computador convencional.”
- “O Arduino é uma plataforma open-source de computação física baseada em uma placa simples com um microcontrolador e um ambiente de desenvolvimento para escrever programas para a placa.”
- “Exemplos de outras plataformas: Parallax Basic Stamp, Netmedia's BX-24, Phidgets, MIT's Handyboard, Texas Instrument's Energia.”
- “Estas plataformas transformam todos os detalhes complicados da programação de microcontroladores em um pacote fácil de usar.”



plataforma arduino - hardware



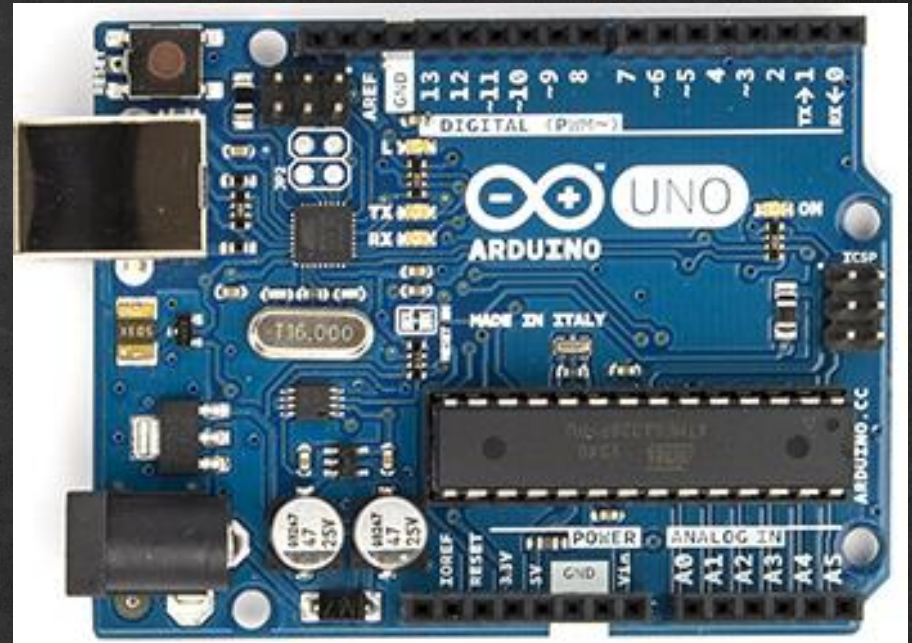
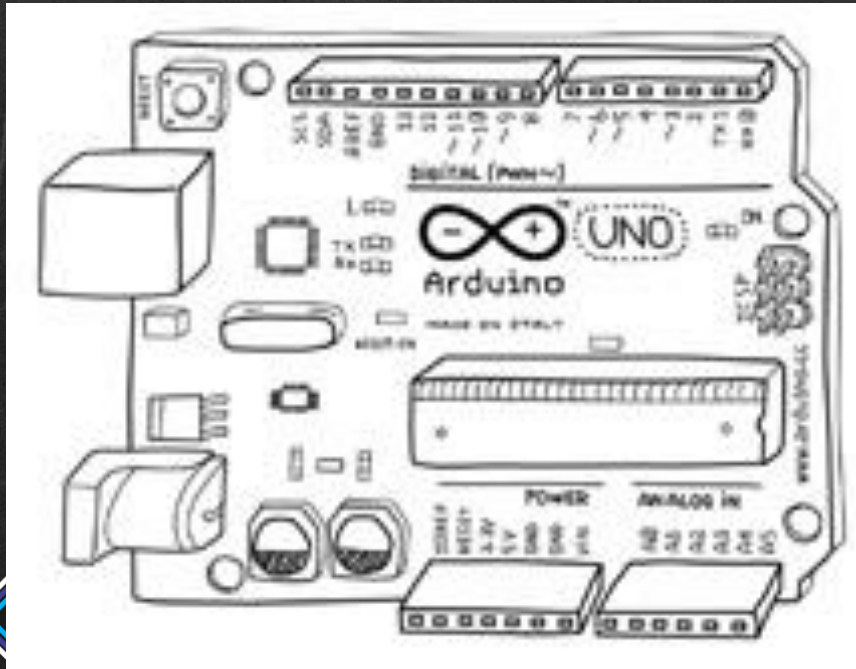
ENTRY LEVEL	ARDUINO UNO / ARDUINO PRO / ARDUINO PRO MINI / ARDUINO MICRO / ARDUINO NANO / ARDUINO STARTER KIT / ARDUINO BASIC KIT / ARDUINO MOTOR SHIELD
ENHANCED FEATURES	ARDUINO MEGA / ARDUINO ZERO / ARDUINO DUE / ARDUINO PROTO SHIELD
INTERNET OF THINGS	ARDUINOYÚN / ARDUINO ETHERNET SHIELD / ARDUINO GSM SHIELD / ARDUINO WIFI SHIELD 101
WEARABLE	ARDUINO GEMMA / ARDUINO LILYPAD / ARDUINO LILYPAD SIMPLE / ARDUINO LILYPAD USB
3D PRINTING	MATERIA 101



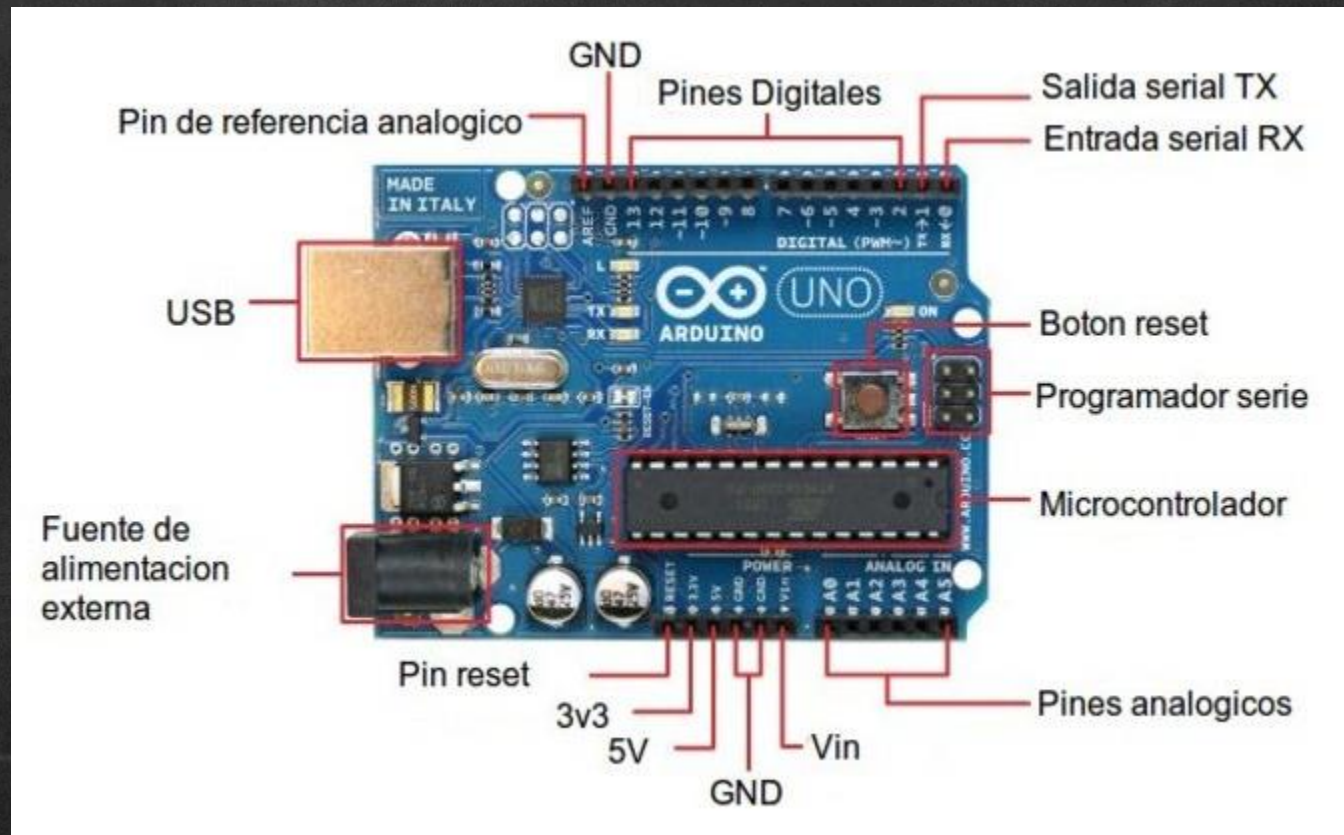
nossa escolha



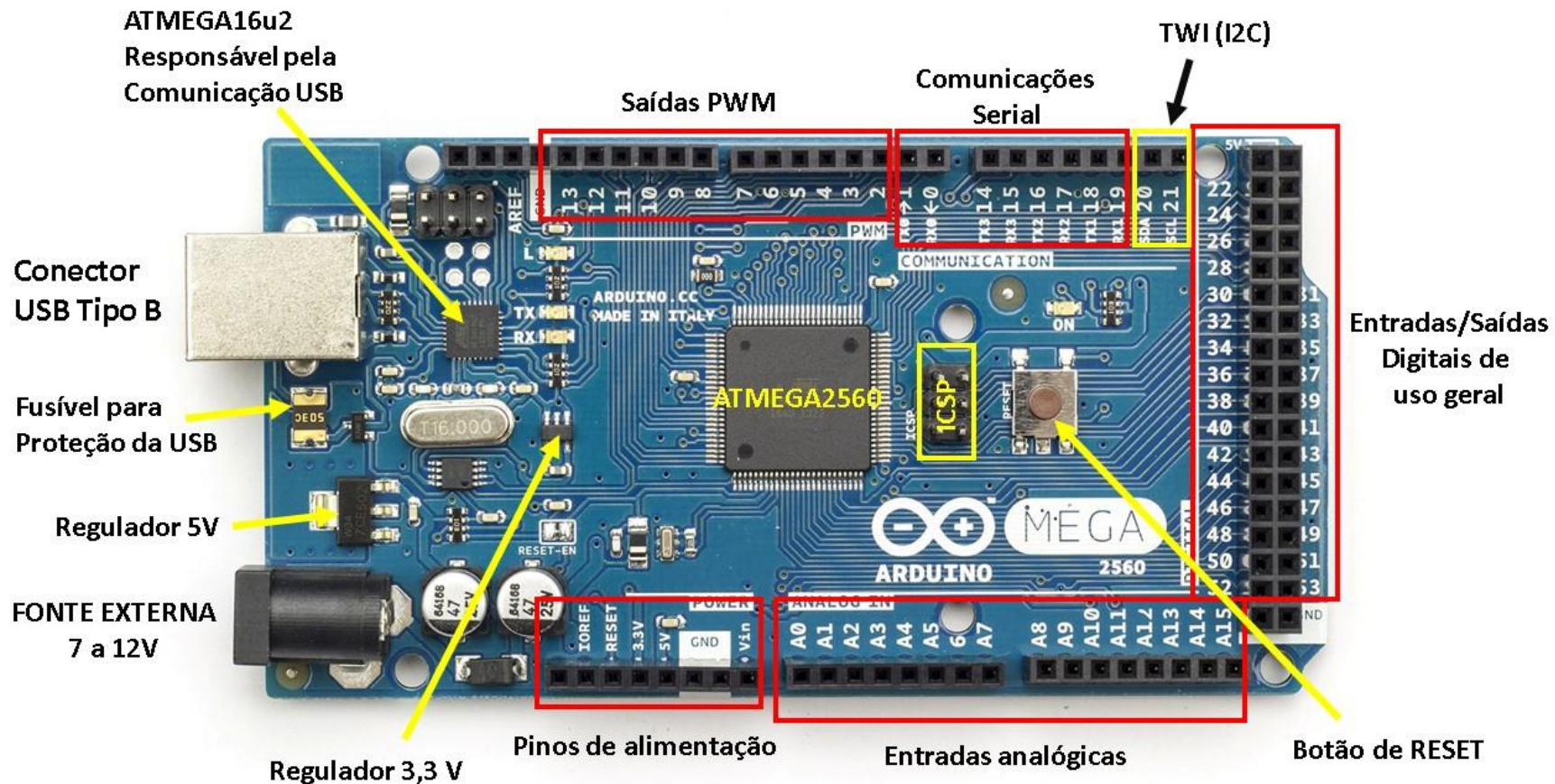
- Arduino UNO



arduino UNO



arduino MEGA – hardware opção ao UNO



plataforma arduino - instalação



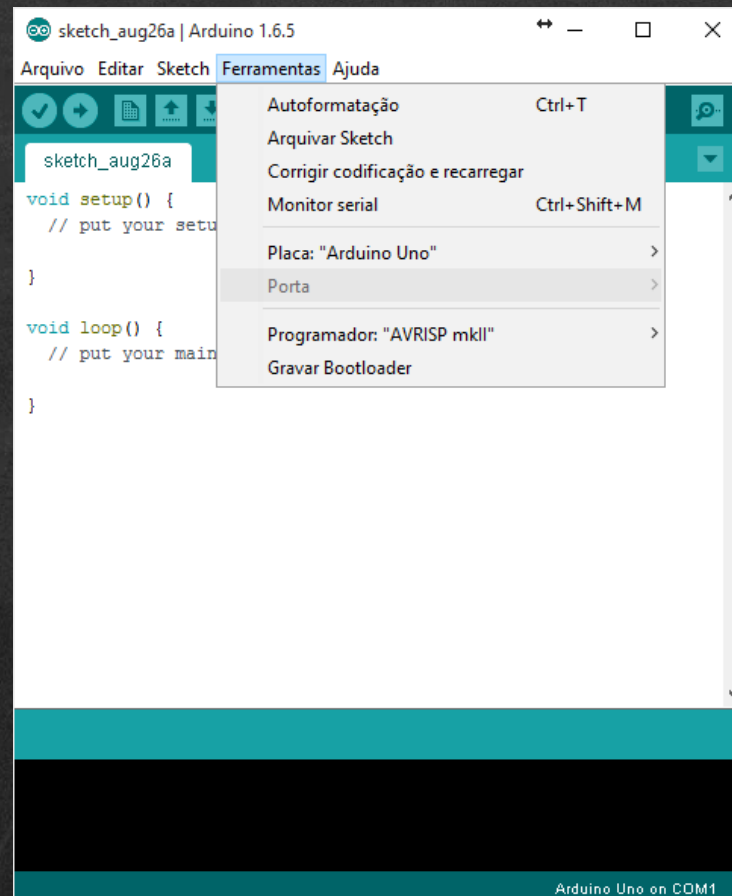
- Driver
 - windows: FTDI Serial USB
 - linux: não precisa instalar nada :-)
- Software - existem 2 versões:
 - descompactar e executar
 - Instalação (instala os drivers automaticamente)



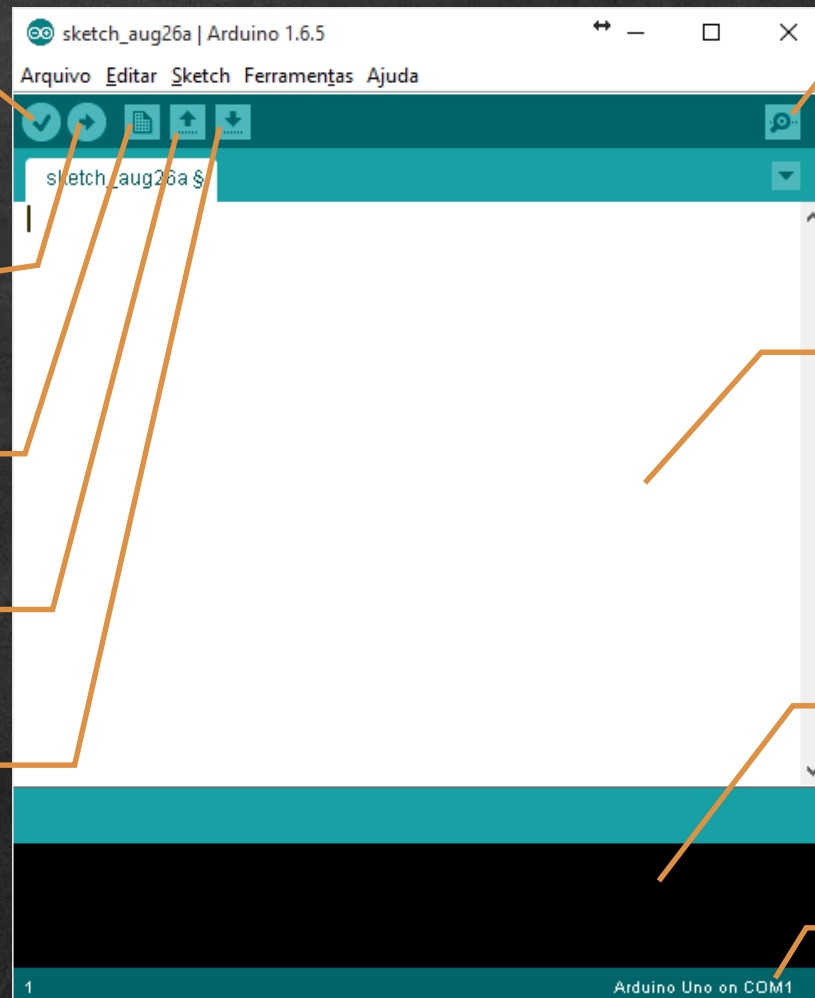
plataforma arduino – ide



- Seleccionando a placa e a porta serial



plataforma arduino – ide



Compilar
(Verif. programa)

enviar programa
para placa

novO

abrir

salvar

exibir
"Monitor
Serial"

área de código

área de status

exibir serial



plataforma arduino – estrutura do sketch (programa)



The screenshot shows the Arduino IDE window titled 'sketch_aug27a | Arduino 1.6.5'. The menu bar includes 'Arquivo', 'Editar', 'Sketch', 'Ferramentas', and 'Ajuda'. The toolbar contains icons for checking, running, serial monitor, upload, and download. The sketch editor shows the following code:

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

The status bar at the bottom right indicates 'Arduino Uno on COM1'.



plataforma arduino – linguagem



- linguagem baseada em C (mas bem mais fácil... será?)
- comandos básicos
 - pinMode() – define um pino com entrada ou saída
 - digitalWrite() – liga ou desliga uma saída digital
 - delay() – “espera” um determinado tempo



plataforma arduino – linguagem



- exemplos

- pinMode(num_do_pino, OUTPUT);
- digitalWrite(num_do_pino, valor);
valor é LOW ou HIGH (0 ou 1, 0V ou 5V)
- delay(milisegundos);



plataforma arduino – linguagem



- constantes

- LOW | HIGH – indica nível baixo (0V) e alto (5V) nos pinos
- INPUT | OUTPUT – define se um pino vai ser pino de entrada ou de saída



atividade prática!



- fazer o programa hello arduino, que pisca um led
- use o pino 13 de saída digital, a placa já possui um led ligado a ele :-)



plataforma arduino – Blink



The screenshot shows the Arduino IDE window titled "Blink | Arduino 1.6.5". The menu bar includes "Arquivo", "Editar", "Sketch", "Ferramentas", and "Ajuda". The toolbar contains icons for checking, running, serial monitor, upload, and download. The sketch name "Blink\$" is displayed in the top bar. The code in the editor is as follows:

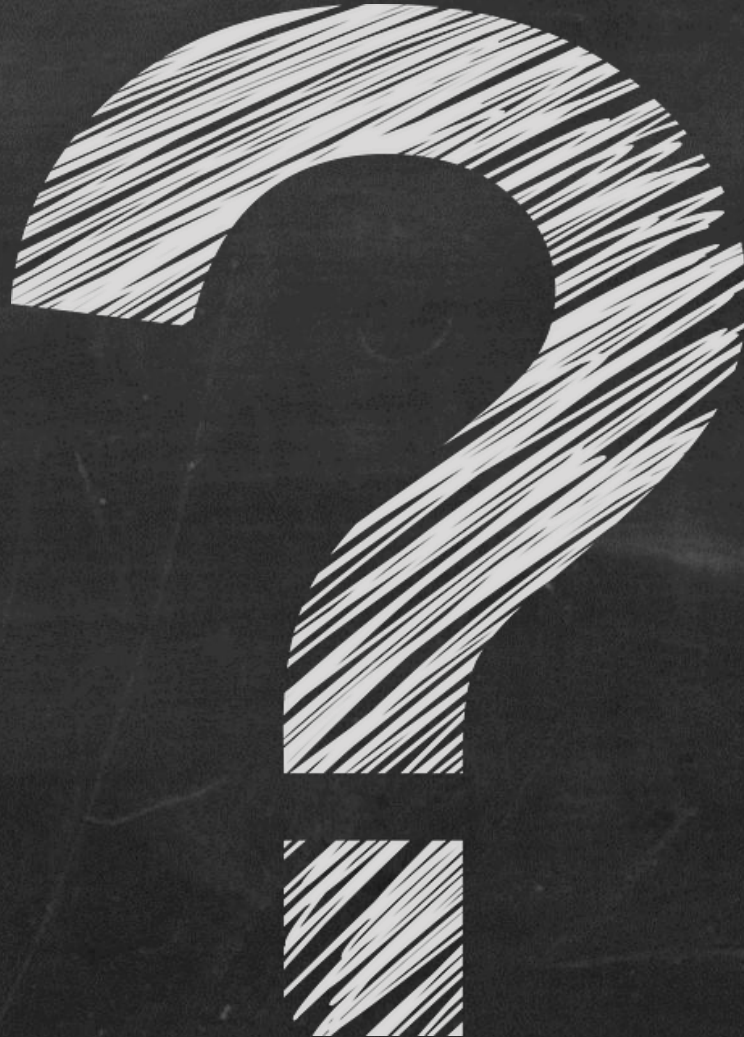
```
#define LED 13
#define tempo 1000

// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin 13 as an output.
  pinMode(LED, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED, HIGH);  // turn the LED on (HIGH is the voltage level)
  delay(tempo);             // wait for a second
  digitalWrite(13, LOW);    // turn the LED off by making the voltage LOW
  delay(tempo);             // wait for a second
}
```



Perguntas



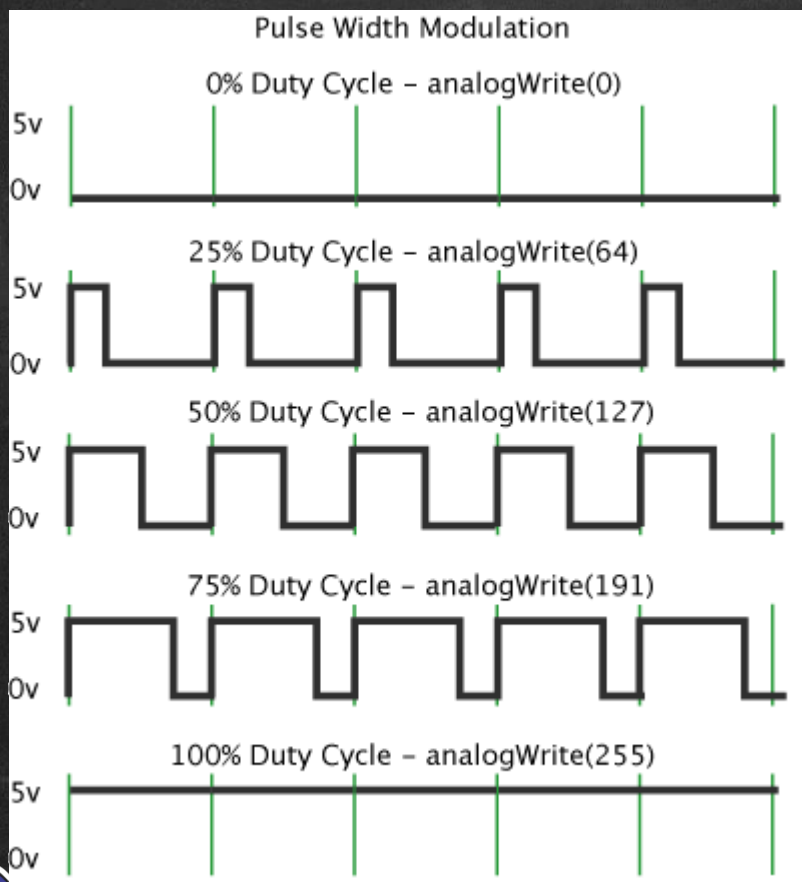
plataforma arduino – linguagem



- comandos básicos
 - analogWrite() – escreve um valor analógico no pino
 - analogWrite(num_pino, valor);
valor entre 0 e 255




eletrônica – modulação PWM



- PWM
PULSE WIDTH MODULATION, ou MODULAÇÃO POR LARGURA DE PULSO, é uma forma de controle de tensão por recorte onde os tiristores ou transistores de potência são ligados ou bloqueados de modo a obter na saída o valor de tensão desejada.
- a função `analogWrite()` escreve “pulsos” muito rápidos no pino digital (só funciona nos pinos marcados com PWM).
- o valor a ser escrito representa o tempo que o pulso fica em nível alto e varia de 0 a 255.
- quanto mais tempo o pulso permanecer em nível alto, maior é a “tensão média” da saída



plataforma arduino – Blink PWM



The screenshot shows the Arduino IDE window titled "Blink | Arduino 1.6.5". The menu bar includes "Arquivo", "Editar", "Sketch", "Ferramentas", and "Ajuda". The toolbar contains icons for checking, running, serial monitor, upload, and download. The file name "Blink\$" is shown in the tab. The code is as follows:

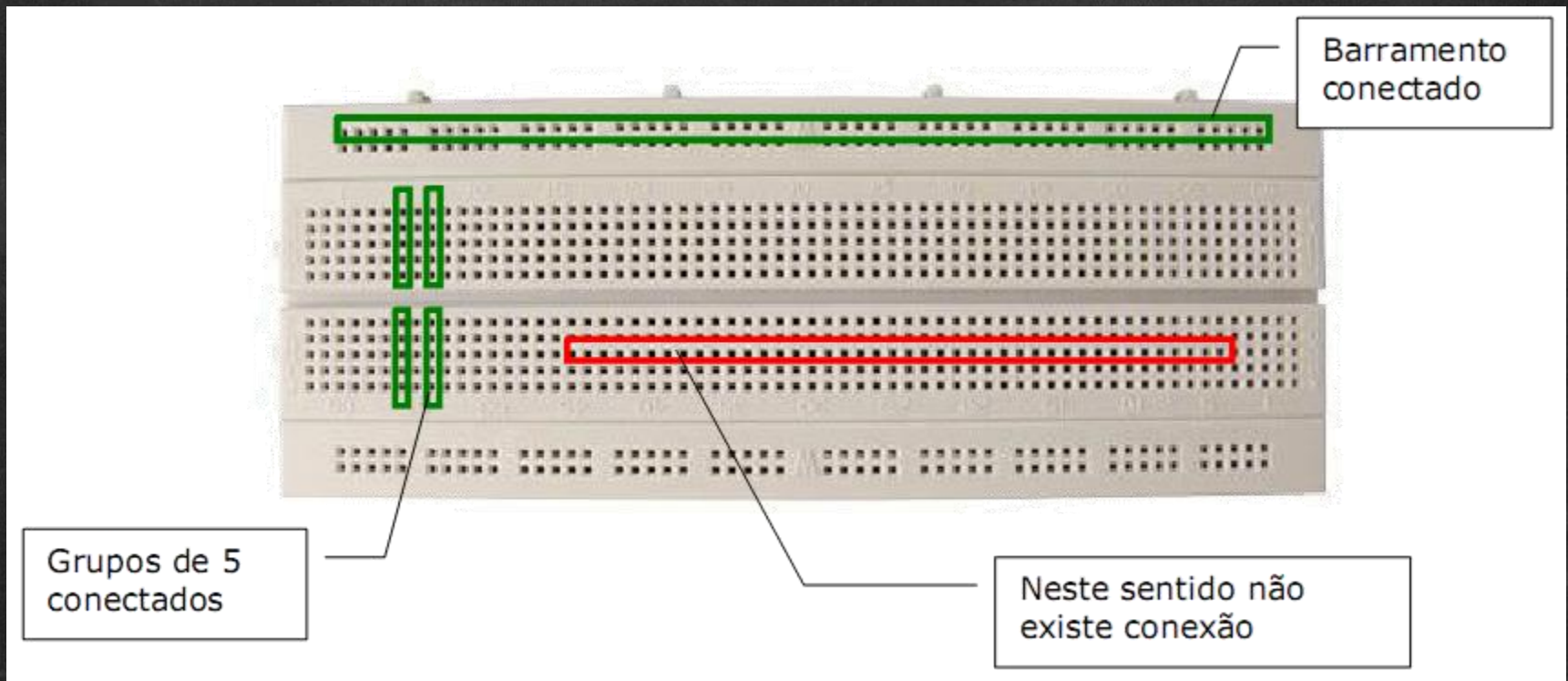
```
#define LED 13
#define tempo 1000

// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin 13 as an output.
  pinMode(LED, OUTPUT); int i=0;
}

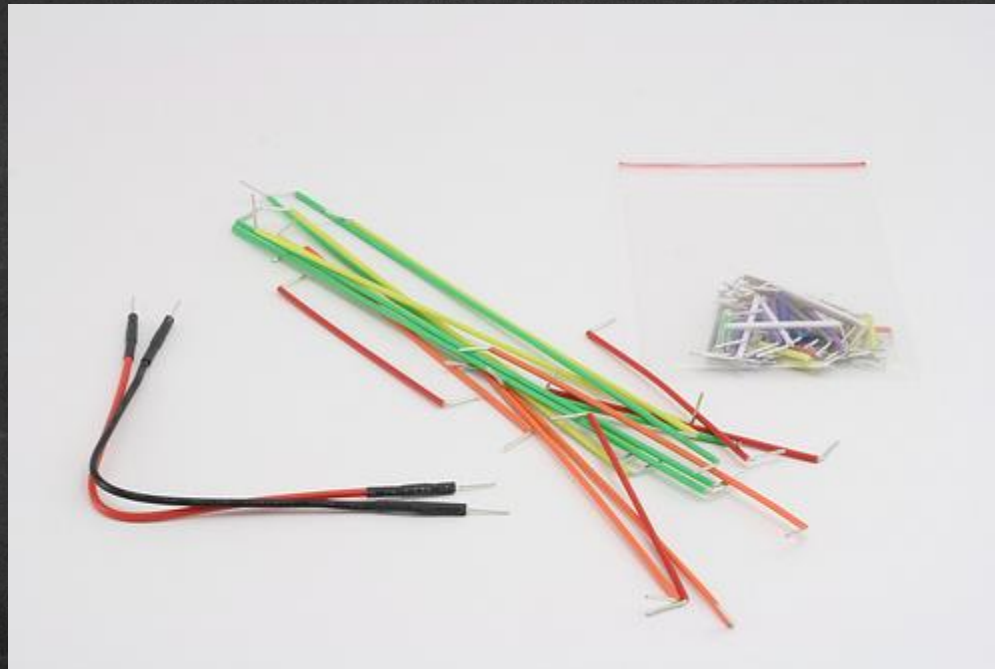
// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED, HIGH); // turn the LED on (HIGH is the voltage level)
                           // wait for a second
  if (i>255)
    digitalWrite(LED, LOW); // turn the LED off by making the voltage LOW
                           // wait for a second
  i = 0;
  else
    i++;
}
```



eletrônica – protoboard



eletrônica – jumpers



eletrônica – resistores



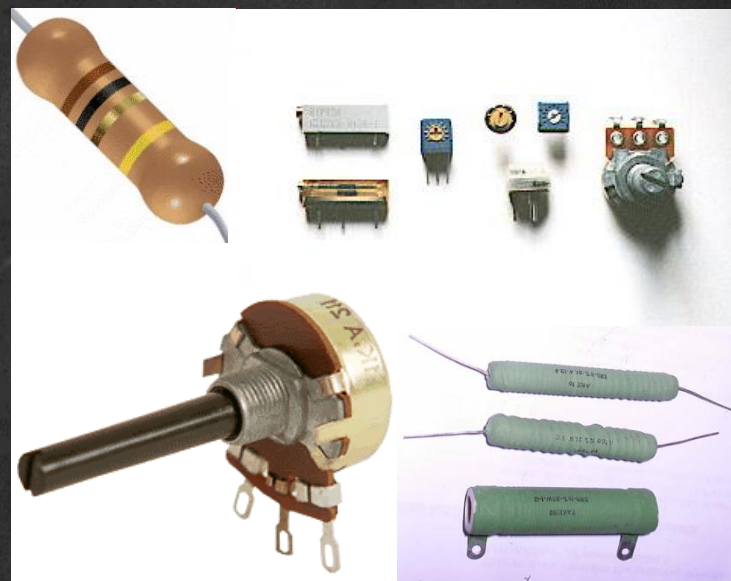
- oferecem resistência à passagem da corrente elétrica

- resistência:

- fixo
- variável

- tipos:

- carvão [carbono]
- filme
- fio transformam energia elétrica em energia térmica [pode ser usado como atuador]



eletrônica – resistores



- valores expressos em ohms
 - o corpo dos resistores possui um código de cores para identificar o valor

Diagram of a 4-Band-Code resistor with a central body and four colored bands (Green, Blue, Yellow, Brown). The tolerance is indicated as 2%, 5%, or 10%.

4-Band-Code

2%, 5%, 10%

560k Ω \pm 5%

COLOR	1st BAND	2nd BAND	3rd BAND	MULTIPLIER	TOLERANCE
Black	0	0	0	1 Ω	
Brown	1	1	1	10 Ω	\pm 1% (F)
Red	2	2	2	100 Ω	\pm 2% (G)
Orange	3	3	3	1K Ω	
Yellow	4	4	4	10K Ω	
Green	5	5	5	100K Ω	\pm 0.5% (D)
Blue	6	6	6	1M Ω	\pm 0.25% (C)
Violet	7	7	7	10M Ω	\pm 0.10% (B)
Grey	8	8	8		\pm 0.05%
White	9	9	9		
Gold				0.1	\pm 5% (J)
Silver				0.01	\pm 10% (K)



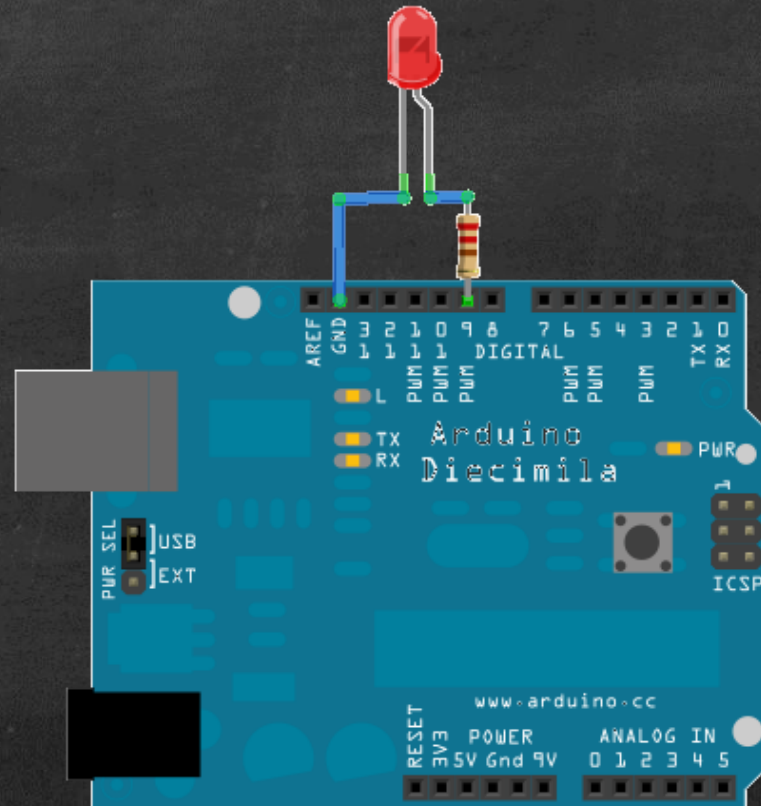
+ prática!



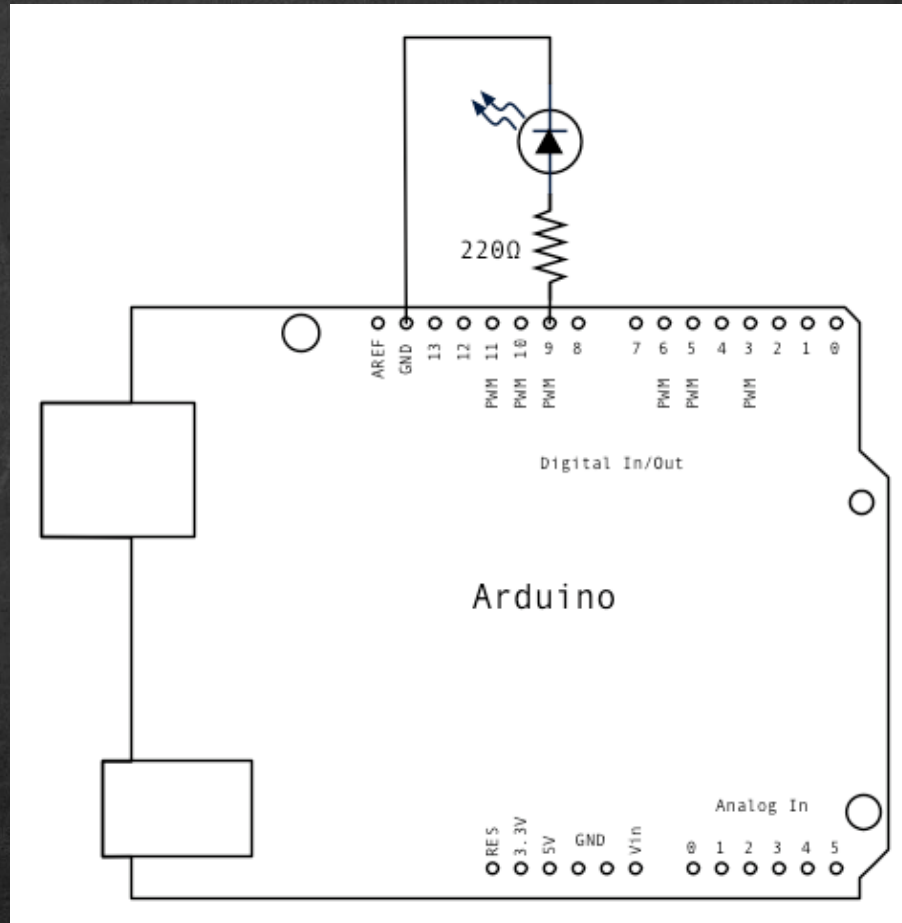
- modificar o programa BLINK para acender o led com efeito de “fading” (acender gradativamente)
- dica: use `analogWrite()` em vez de `digitalWrite()`, variando os valores escritos, de 0 a 255



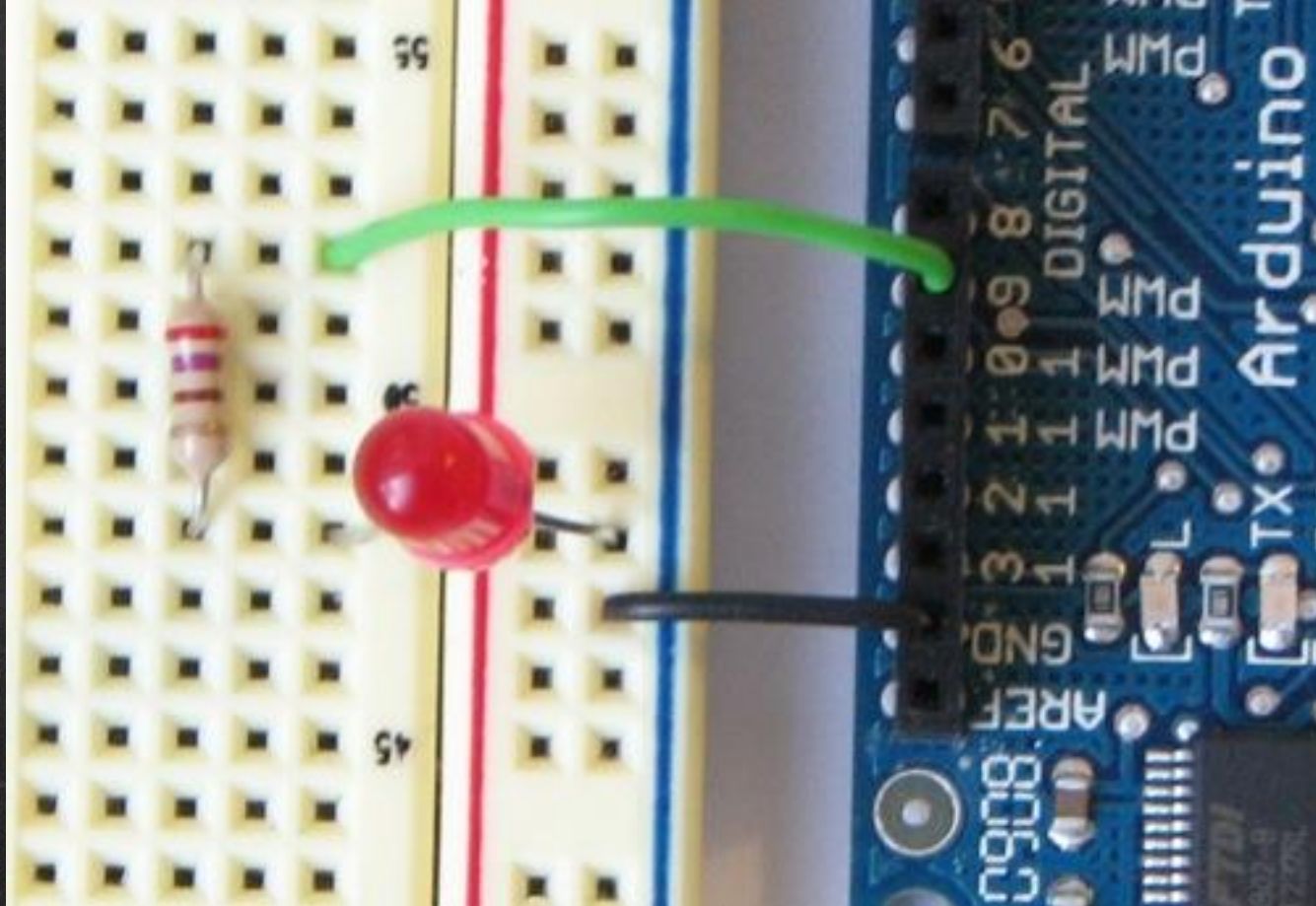
Circuito



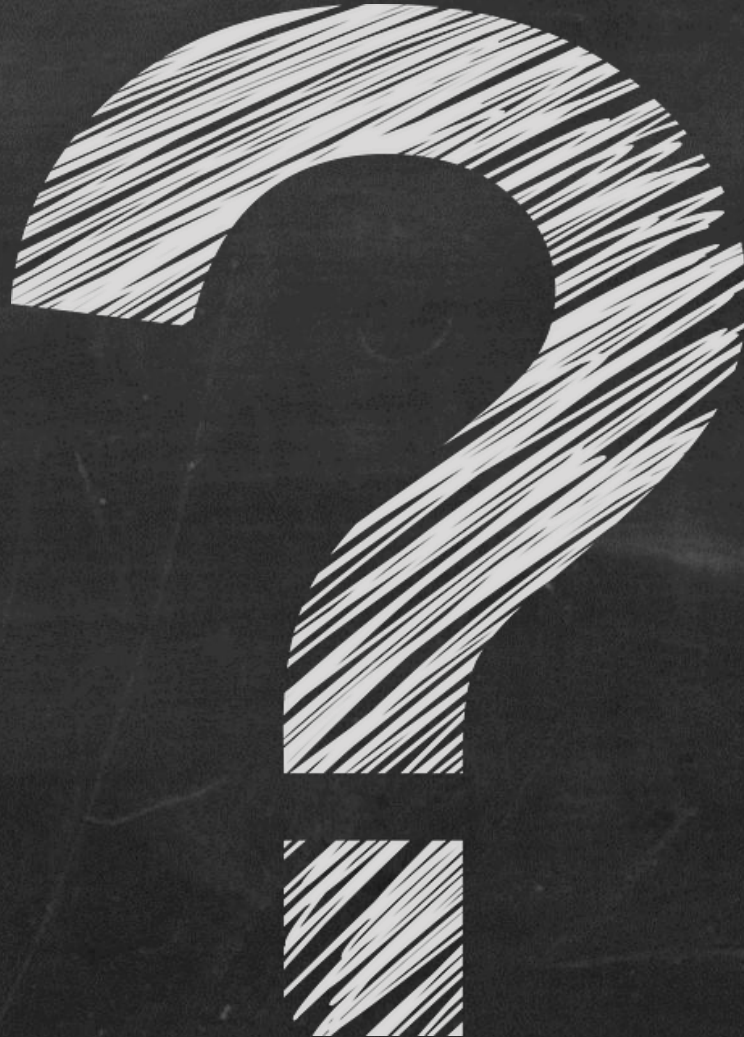
esquemático



protoboard



perguntas





sensores



sensores – chave (switch/button)

- interrompe a passagem da corrente elétrica
- liga/desliga o circuito
- sensor de toque



plataforma arduino – linguagem



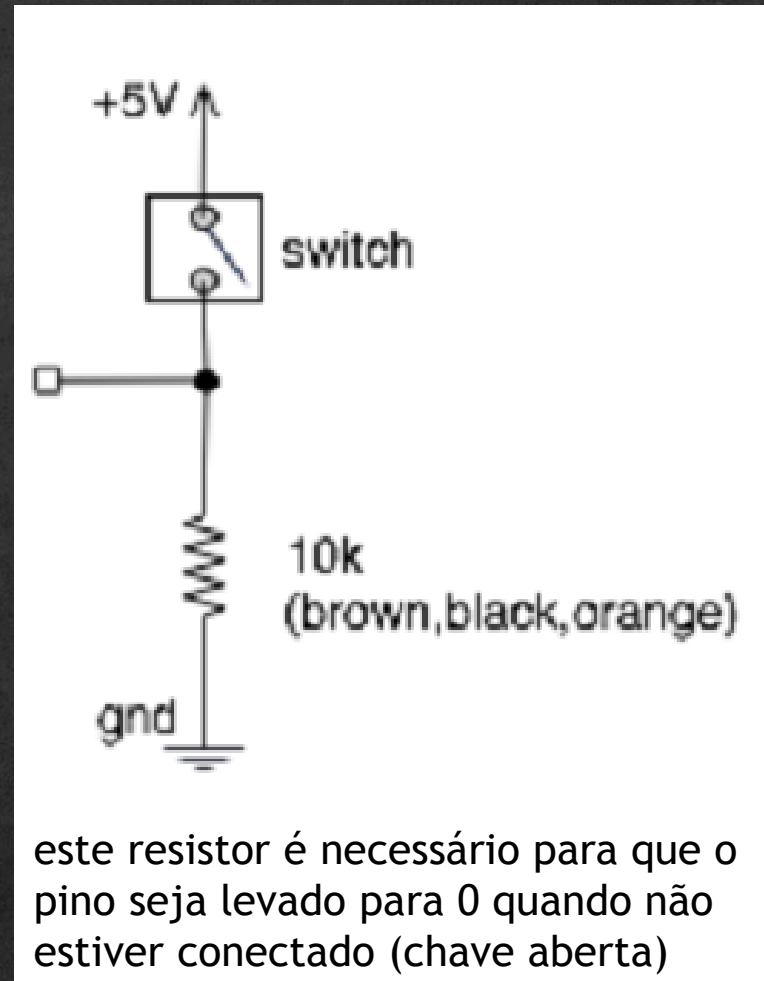
- Comandos
digitalRead() – le um pino de entrada
- Exemplo:
int chave;
chave = digitalRead(num_do_pino);



sensores



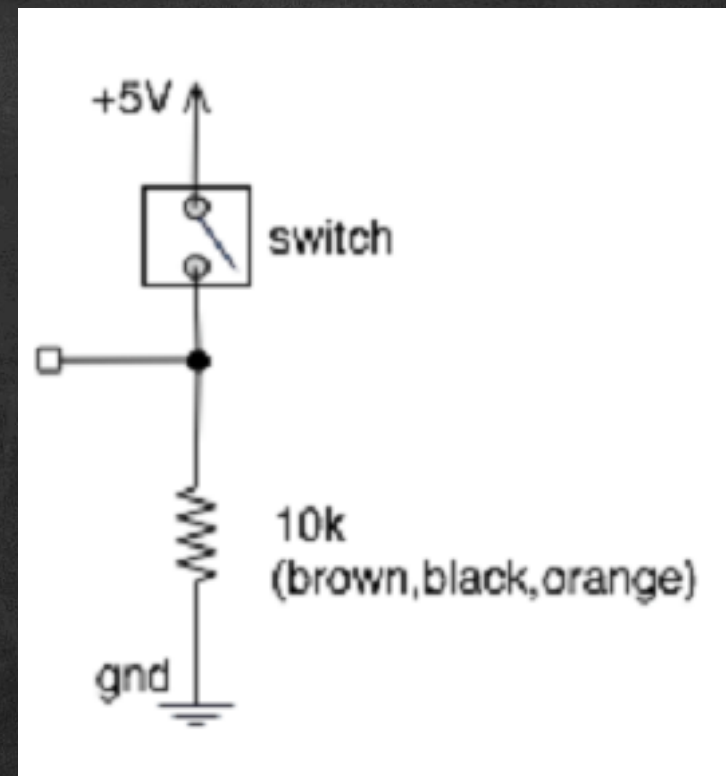
- arduino lê tensões de entrada (e não valores 0 e 1)
- 5 volts == HIGH (1)
- 0 volts == LOW (0)
- sem conexão em um pino, a entrada flutua entre 0 e 5 volts (HIGH e LOW)



sensores - prática



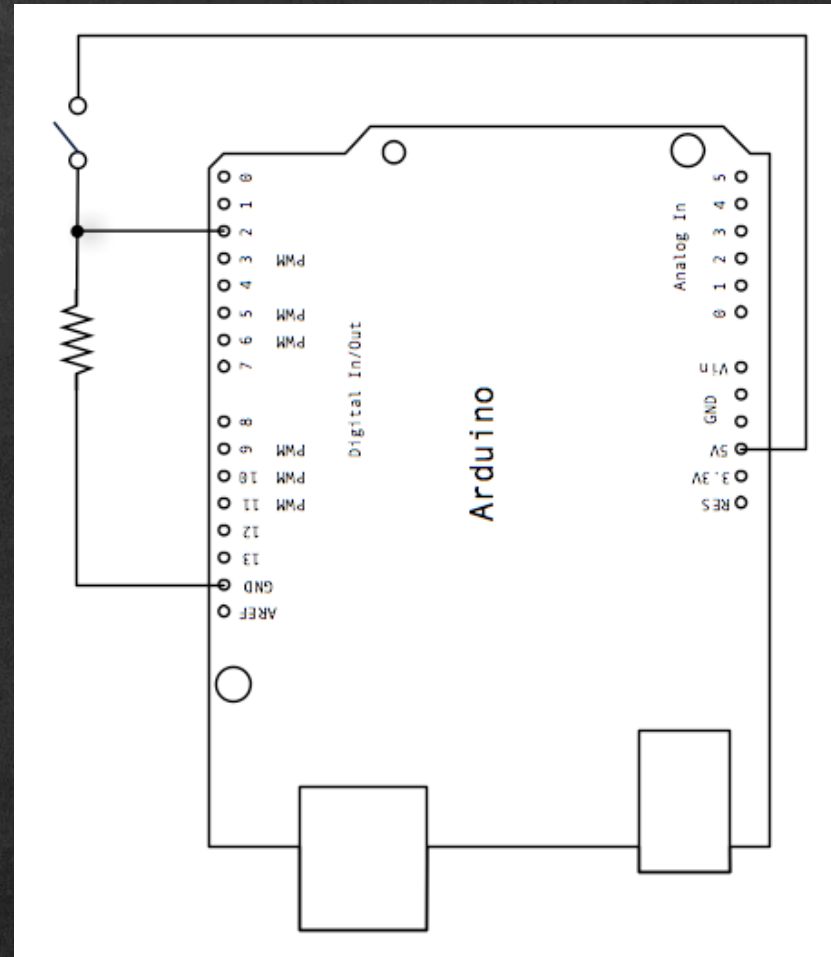
- fazer o circuito e o programa para acender o led 13 de acordo com sinal de entrada do pino 2



sensores - prática



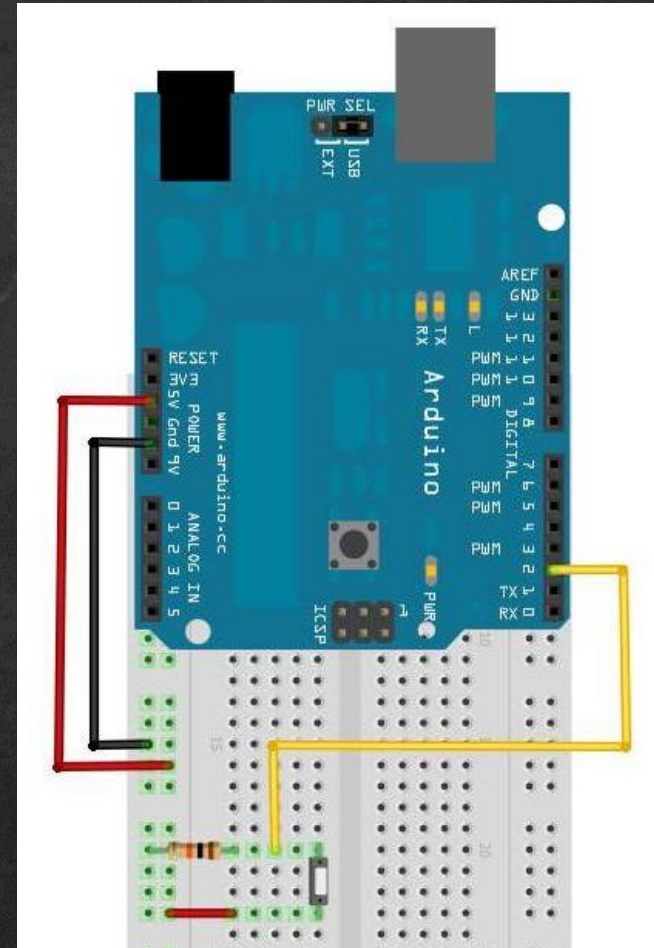
- esquemático



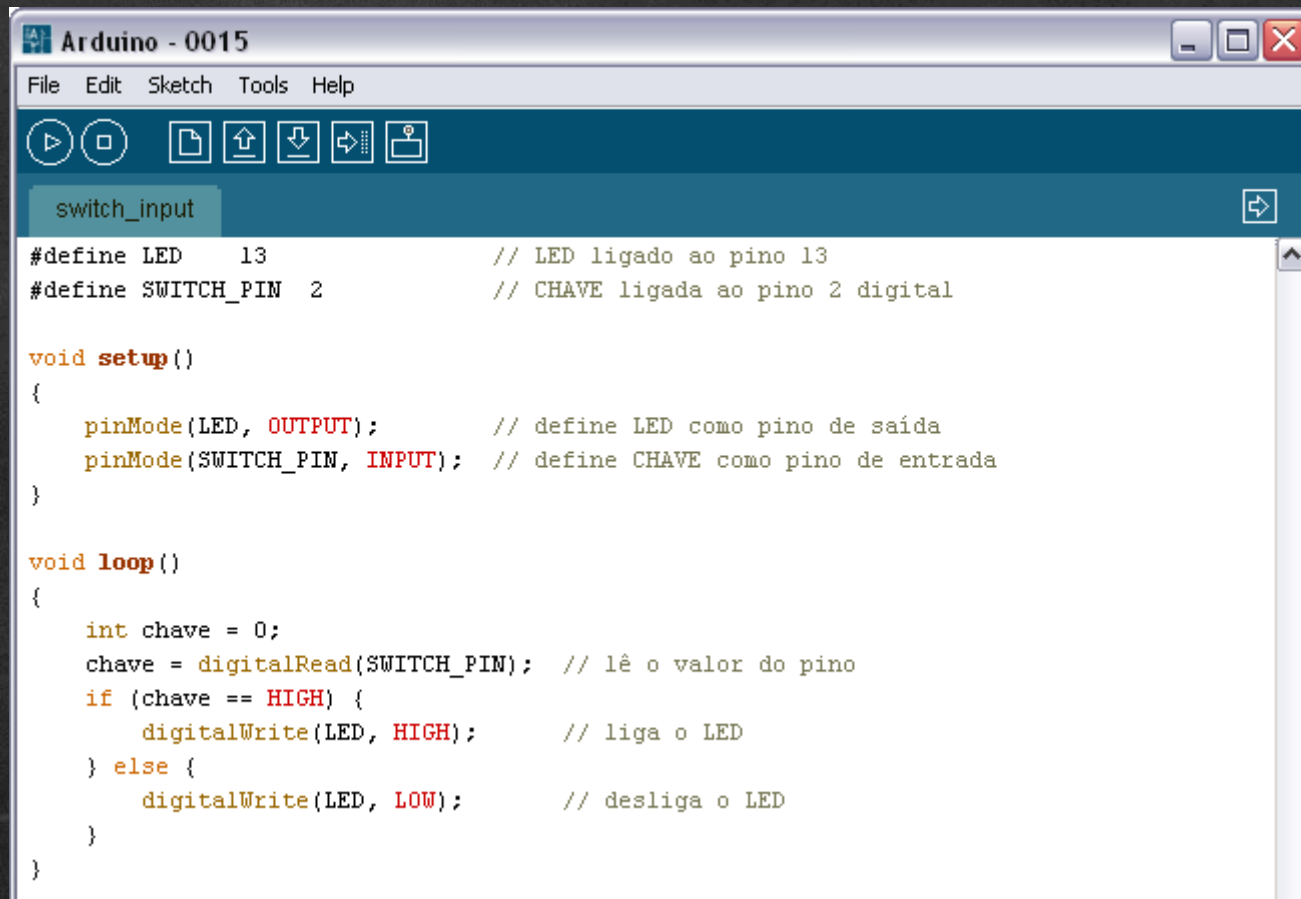
sensores - prática



- protoboard



sensores - prática



```
#define LED 13 // LED ligado ao pino 13
#define SWITCH_PIN 2 // CHAVE ligada ao pino 2 digital

void setup()
{
  pinMode(LED, OUTPUT); // define LED como pino de saída
  pinMode(SWITCH_PIN, INPUT); // define CHAVE como pino de entrada
}

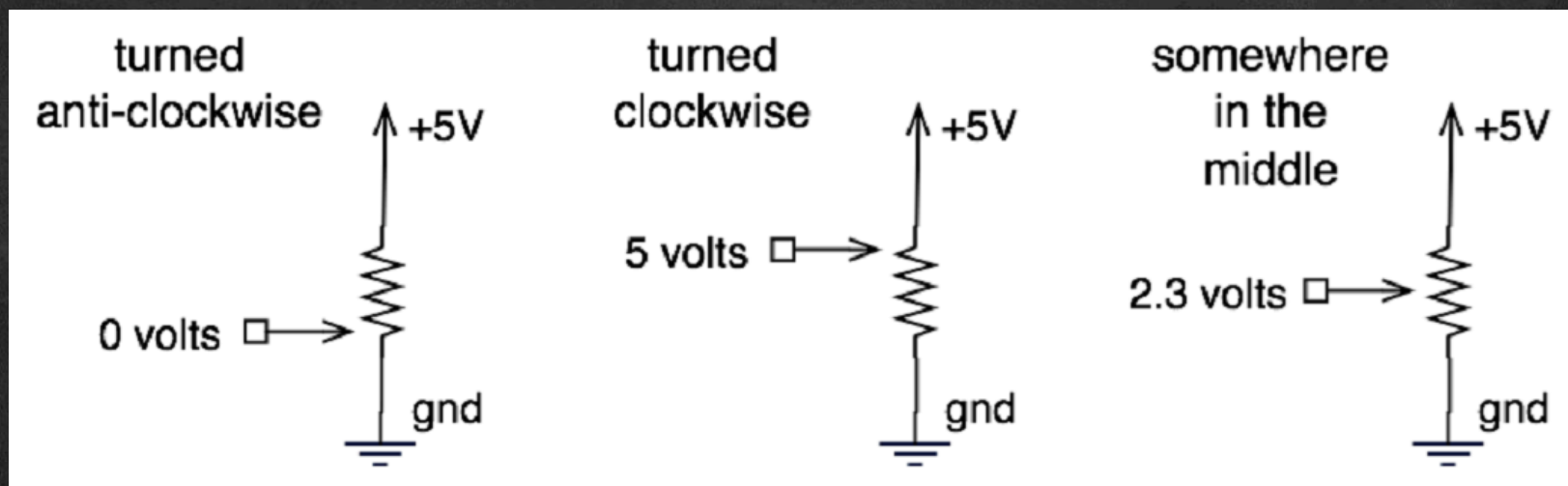
void loop()
{
  int chave = 0;
  chave = digitalRead(SWITCH_PIN); // lê o valor do pino
  if (chave == HIGH) {
    digitalWrite(LED, HIGH); // liga o LED
  } else {
    digitalWrite(LED, LOW); // desliga o LED
  }
}
```



eletrônica – resistores Variáveis



- Como funciona um resistor variável?



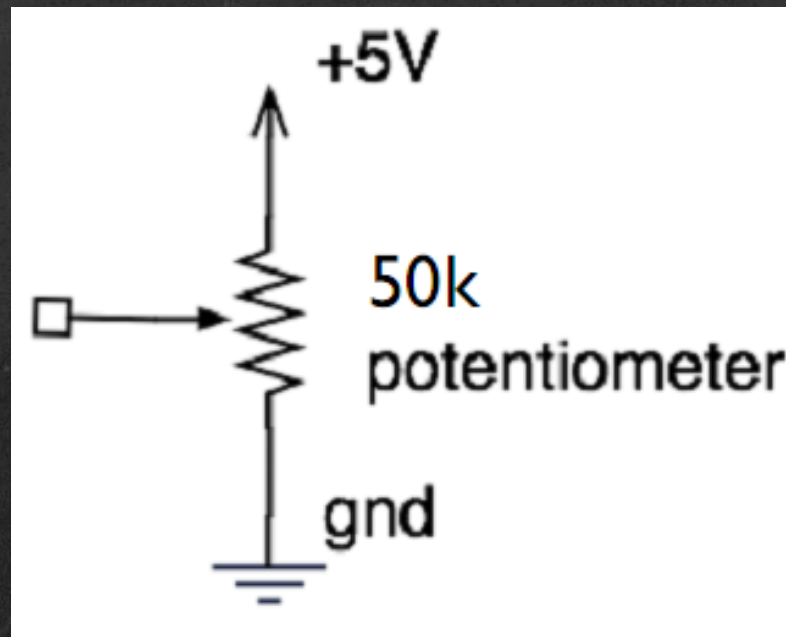
- no arduino, o valor da tensão é transformado em um valor digital entre 0 e 1023



sensores analógicos – prática



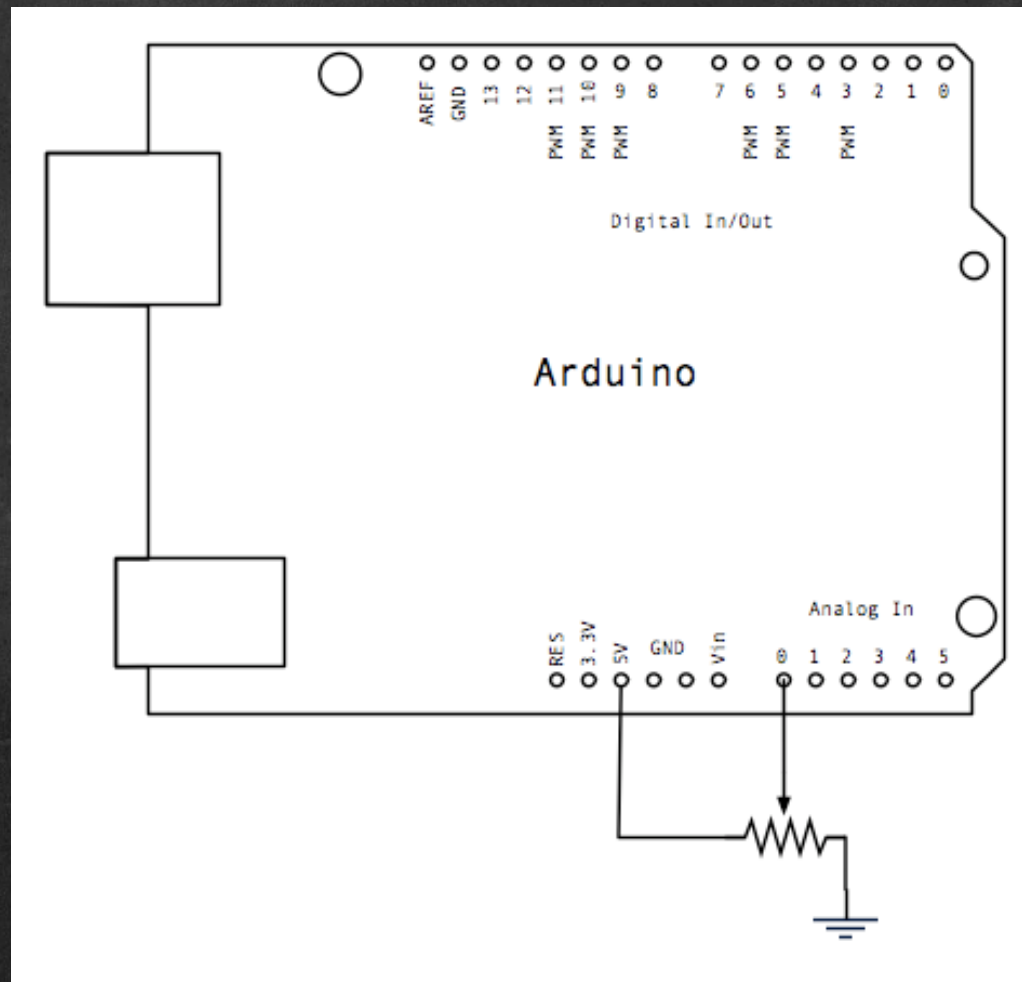
- ler o valor do resistor variável e ligar um LED se esse valor passar de um determinado limite.



sensores analógicos – prática



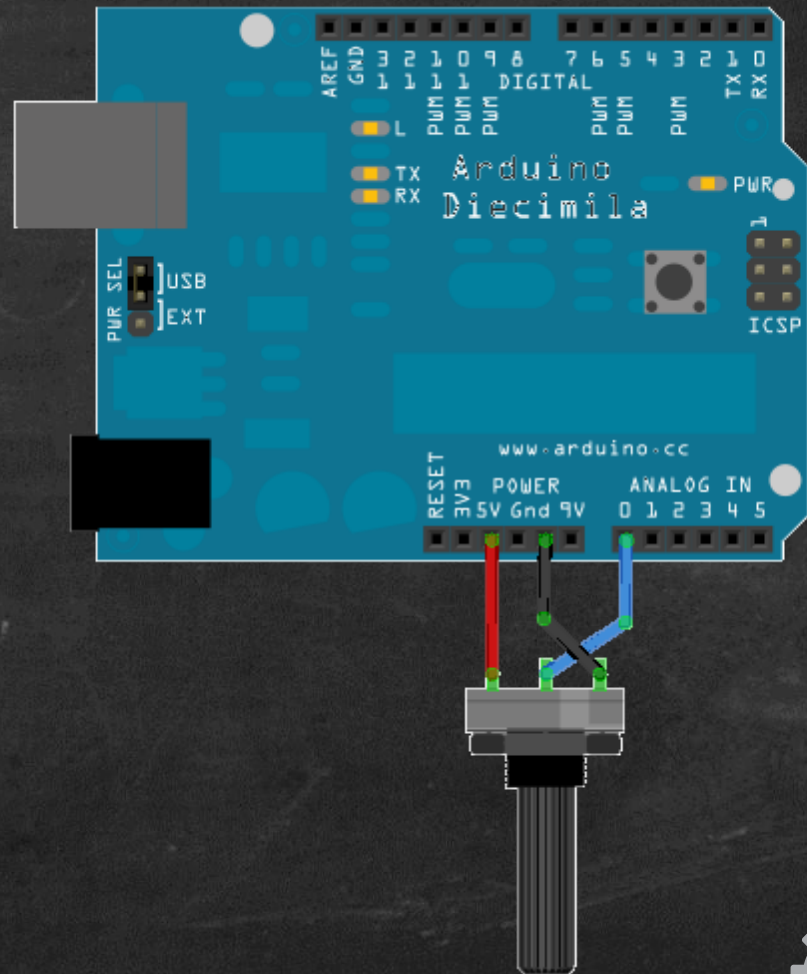
- esquemático



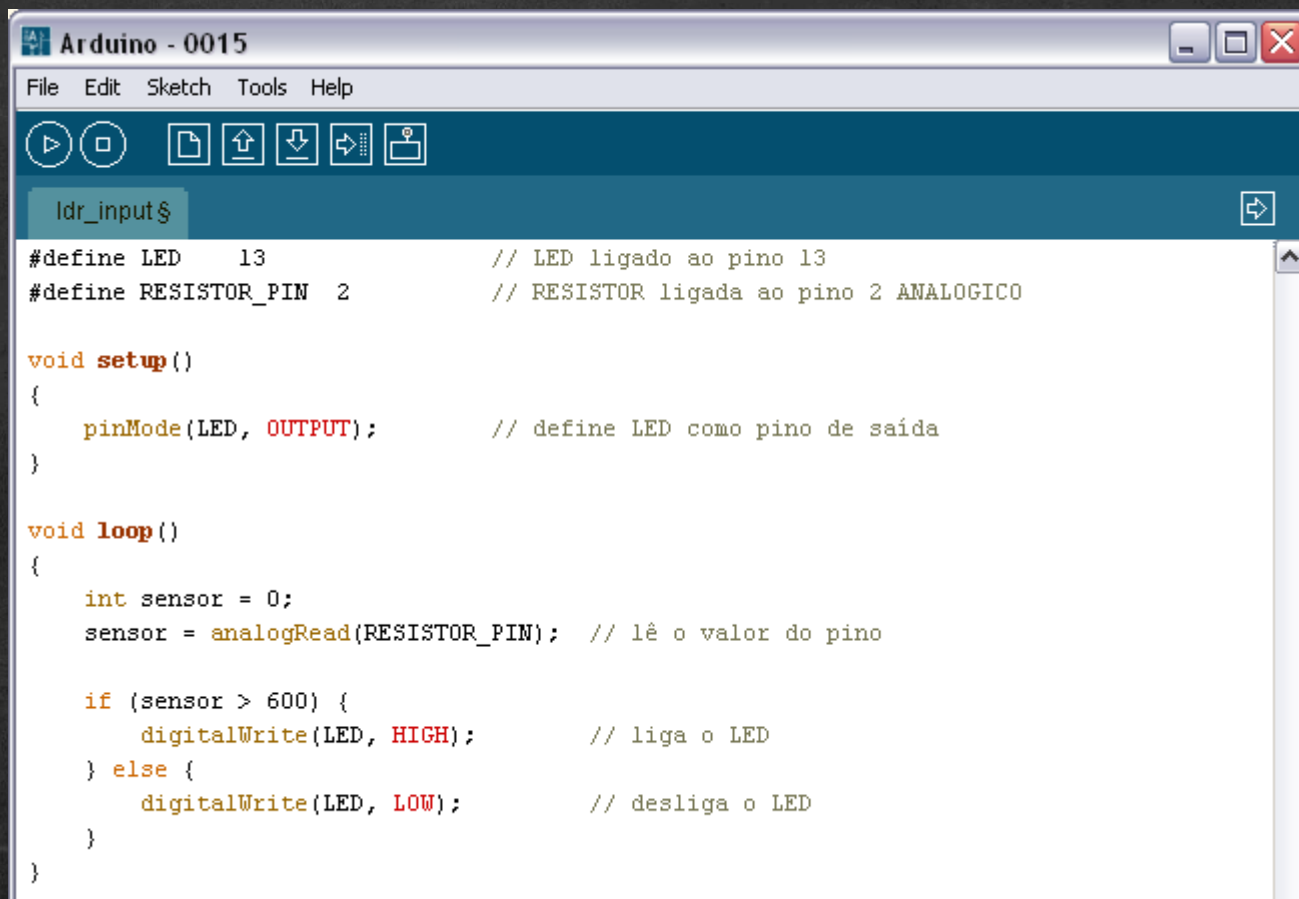
sensores analógicos – prática



- circuito



entrada analógica – prática



```
Arduino - 0015
File Edit Sketch Tools Help

ldr_input$

#define LED 13 // LED ligado ao pino 13
#define RESISTOR_PIN 2 // RESISTOR ligada ao pino 2 ANALOGICO

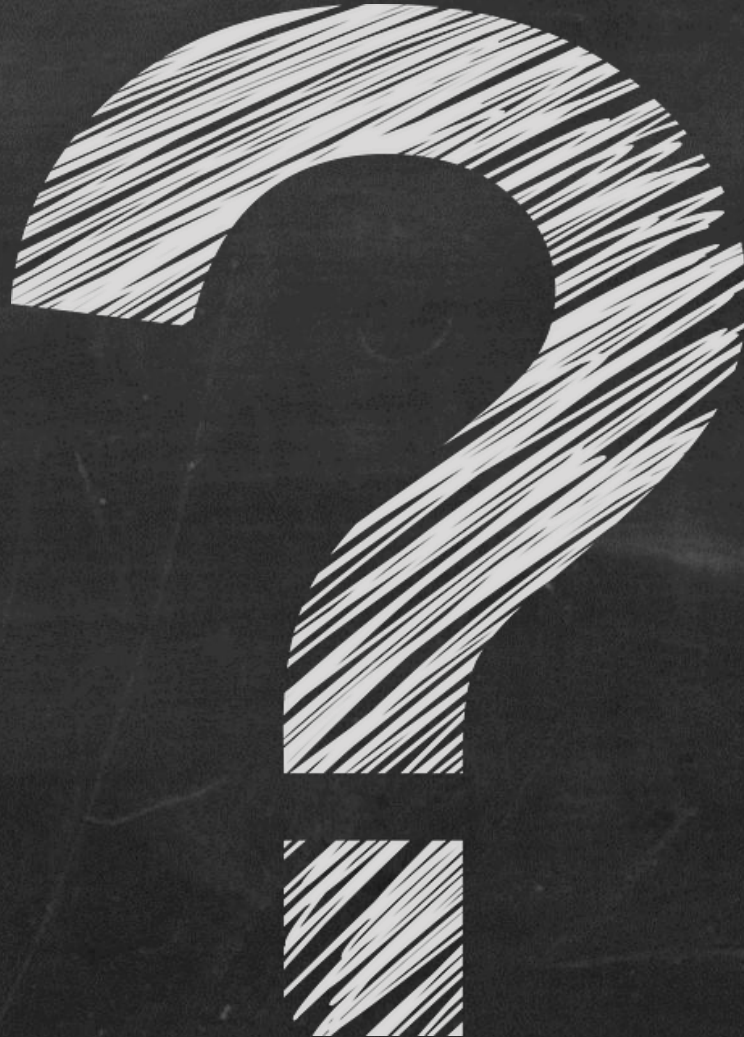
void setup()
{
  pinMode(LED, OUTPUT); // define LED como pino de saída
}

void loop()
{
  int sensor = 0;
  sensor = analogRead(RESISTOR_PIN); // lê o valor do pino

  if (sensor > 600) {
    digitalWrite(LED, HIGH); // liga o LED
  } else {
    digitalWrite(LED, LOW); // desliga o LED
  }
}
```



perguntas





protocolos de comunicação



comunicação serial – RS232



- chip ATMEGA 168 só tem interface serial, não tem USB
- nossa placa arduino possui um chip que converte Serial para USB
- usamos o mesmo cabo USB pra enviar dados pro PC via serial



comunicação serial – RS232



- o arduino possui uma biblioteca que implementa comunicação serial
- `Serial.begin();`
- `Serial.print();`
- `Serial.read();`



comunicação serial – RS232



- Leds
 - TX: dados enviados para o PC
 - RX: dados recebidos do PC



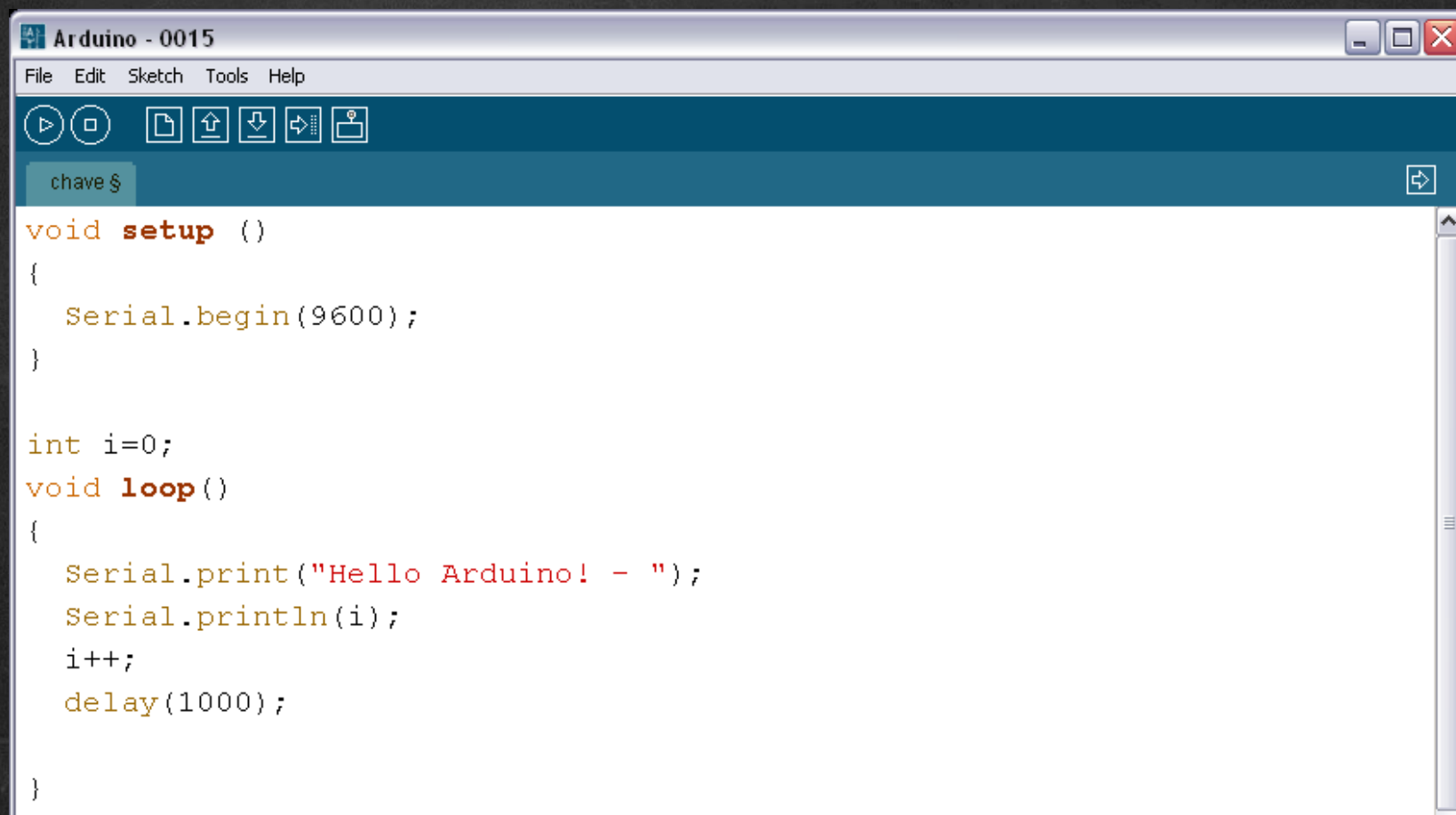
comunicação serial - prática



- “Hello Arduino” via serial



comunicação serial - prática



The screenshot shows the Arduino IDE window titled "Arduino - 0015". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". The toolbar contains icons for running, stopping, saving, opening, and uploading. The sketch name "chave\$" is displayed in the top bar. The code in the editor is as follows:

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

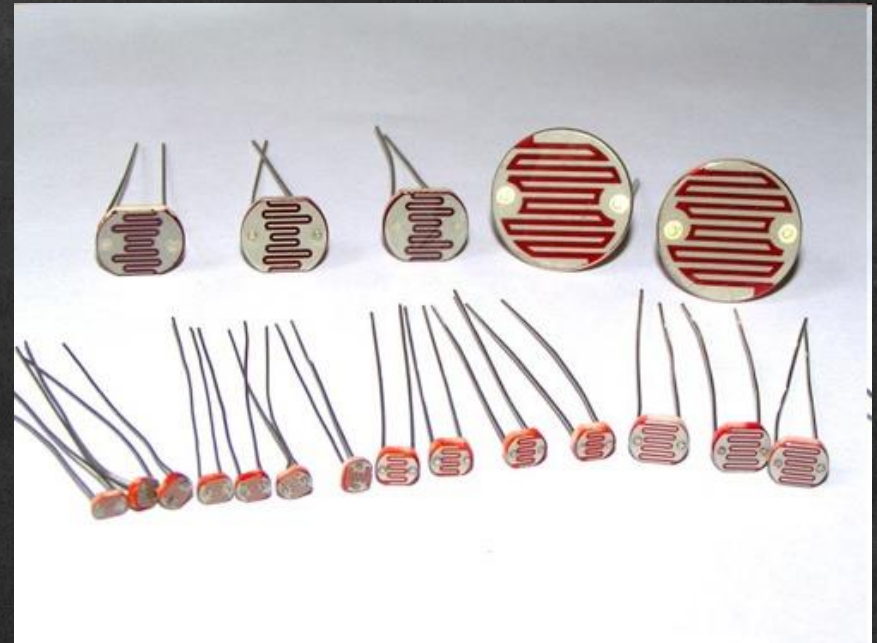
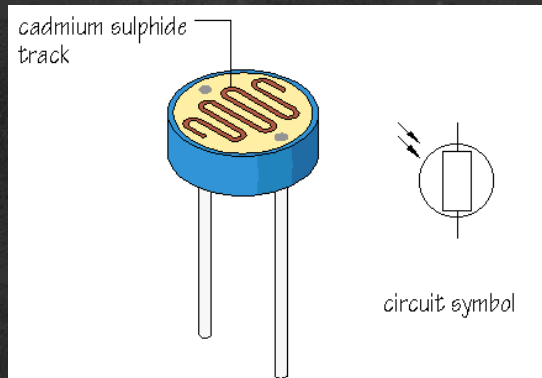
int i=0;
void loop()
{
  Serial.print("Hello Arduino! - ");
  Serial.println(i);
  i++;
  delay(1000);
}
```



eletrônica – resistores LDR



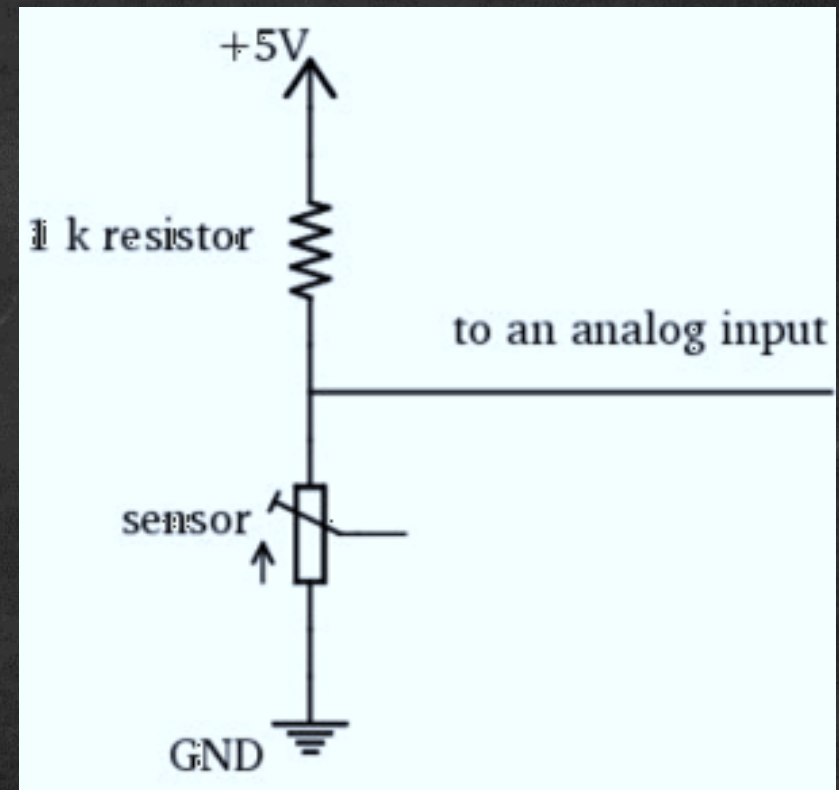
- resistor variável sensível à luz



eletrônica – resistores LDR



- resistor variável sensível à luz
- circuito para arduino
- porque o resistor de 1k?
- para limitar a corrente se o LDR assumir valores muito baixos



comunicação serial - prática



- ler valores do LDR e enviar via serial



comunicação serial - prática



```
sketch_aug28a | Arduino 1.6.5
Arquivo Editar Sketch Ferramentas Ajuda

sketch_aug28a$

//Pino analógico em que o sensor está conectado.
int sensor = 0;

//variável usada para ler o valor do sensor em tempo real.
int valorSensor = 0;

//Método setup, executado uma vez ao ligar o Arduino.
void setup(){
  //Ativando o serial monitor que exibirá os valores lidos no sensor.
  Serial.begin(9600);
}

//Método loop, executado enquanto o Arduino estiver ligado.
void loop(){
  //Lendo o valor do sensor.
  int valorSensor = analogRead(sensor);

  //Exibindo o valor do sensor no serial monitor.
  Serial.println(valorSensor);
  delay(500);
}

Compilação terminada.

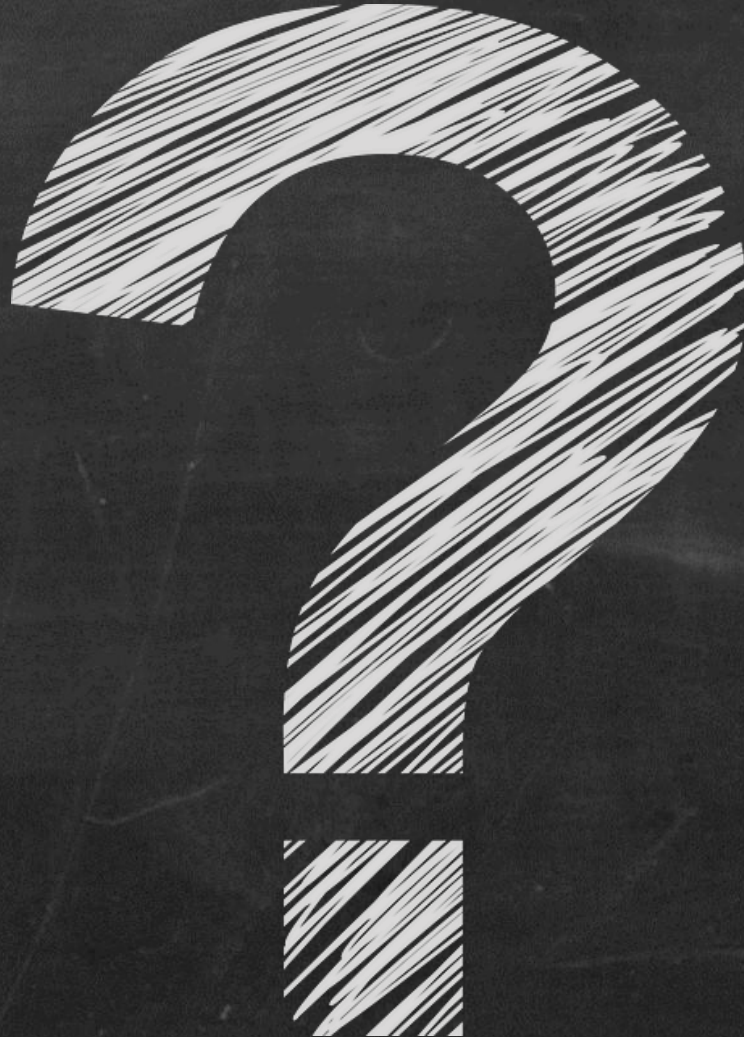
Variáveis globais usam 202 bytes (9%) de memória dinâmica, deixando 1.846 bytes para variáveis locais. O máximo são 2.048 bytes.
```

20

Arduino Uno on COM1



perguntas





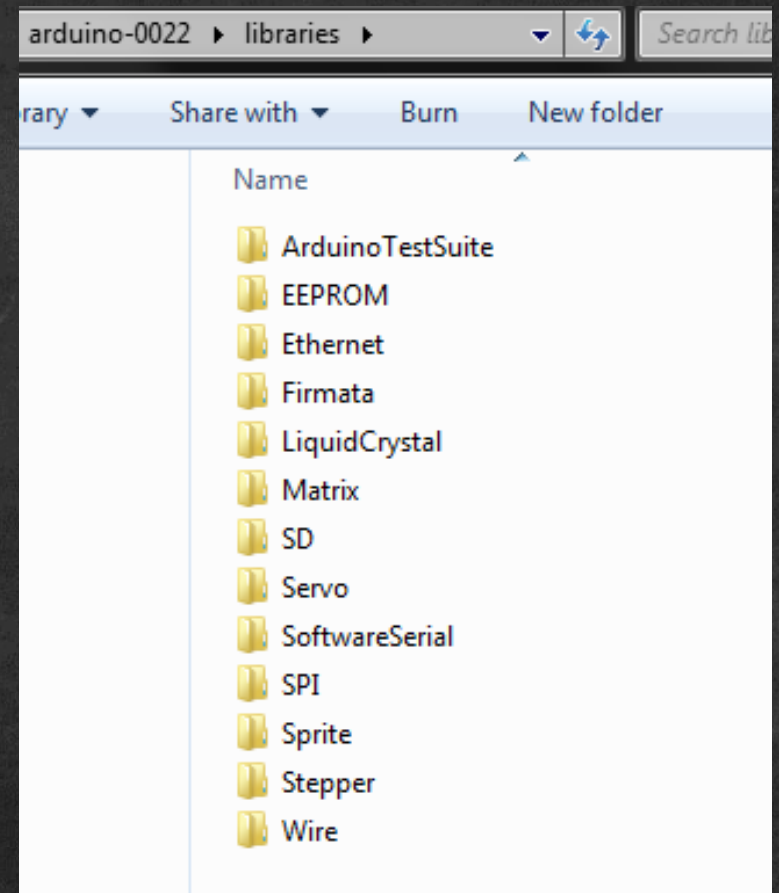
bibliotecas do arduino



bibliotecas do arduino



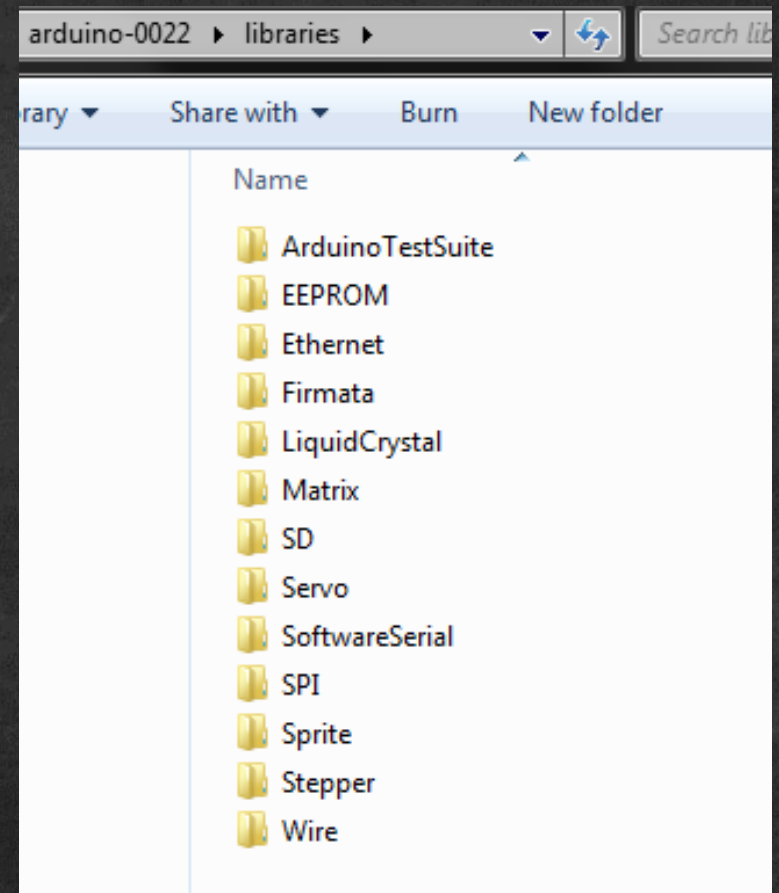
- é possível estender a plataforma Arduino com adição de componentes de código, para controlar sensores e atuadores específicos.
- estes componentes são chamados de bibliotecas (libraries)



bibliotecas do arduino



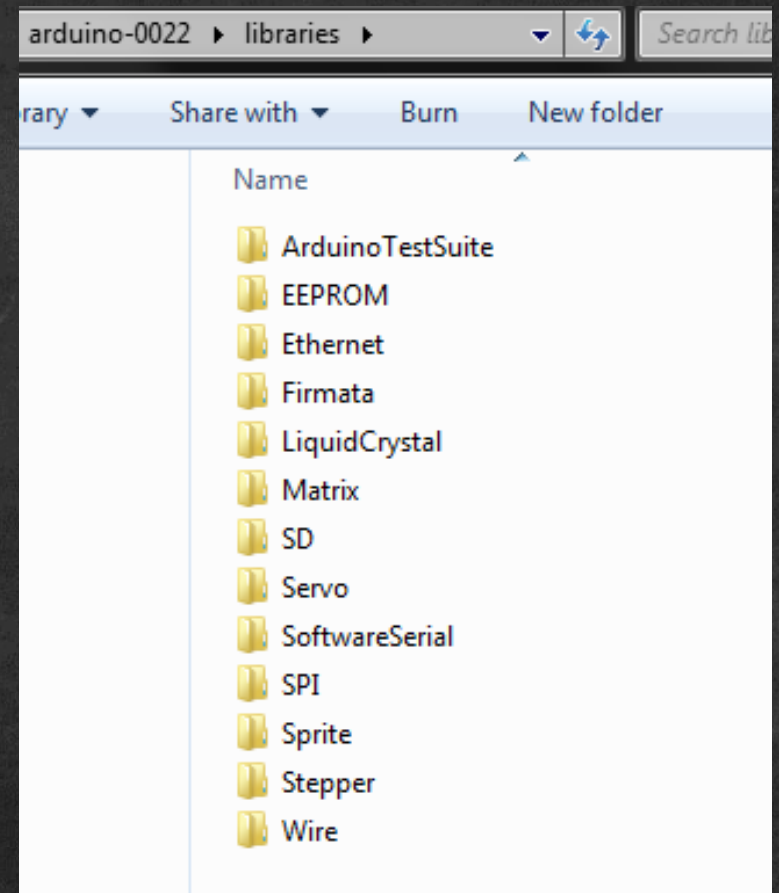
- as bibliotecas são geralmente disponibilizadas como um zip que deve ser descompactado dentro da pasta libraries do Arduino.



bibliotecas do arduino



- após reiniciar o Arduino, a biblioteca estará disponível no menu Sketch->Import Library
- a maioria das bibliotecas para o Arduino pode ser encontrada em <http://arduino.cc/en/Reference/Libraries>



arduino - referencias



- Lista dos comandos da linguagem em:
- <http://arduino.cc/en/Reference/HomePage>
- Lista dos tutoriais em:
- <http://www.arduino.cc/en/Tutorial/HomePage>





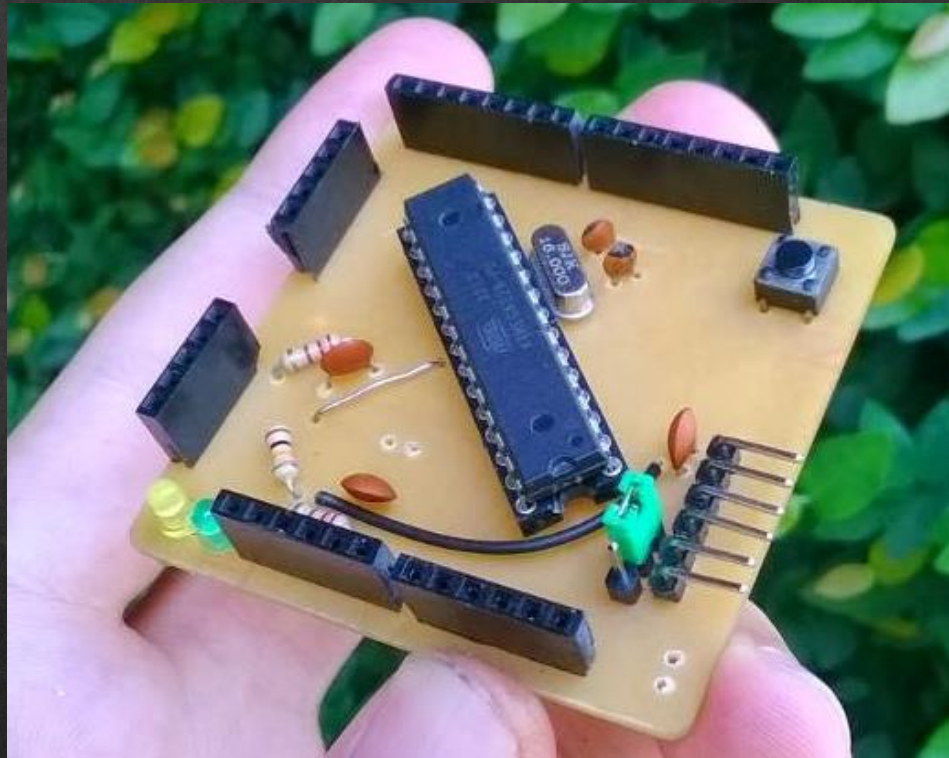
O CEFETINO



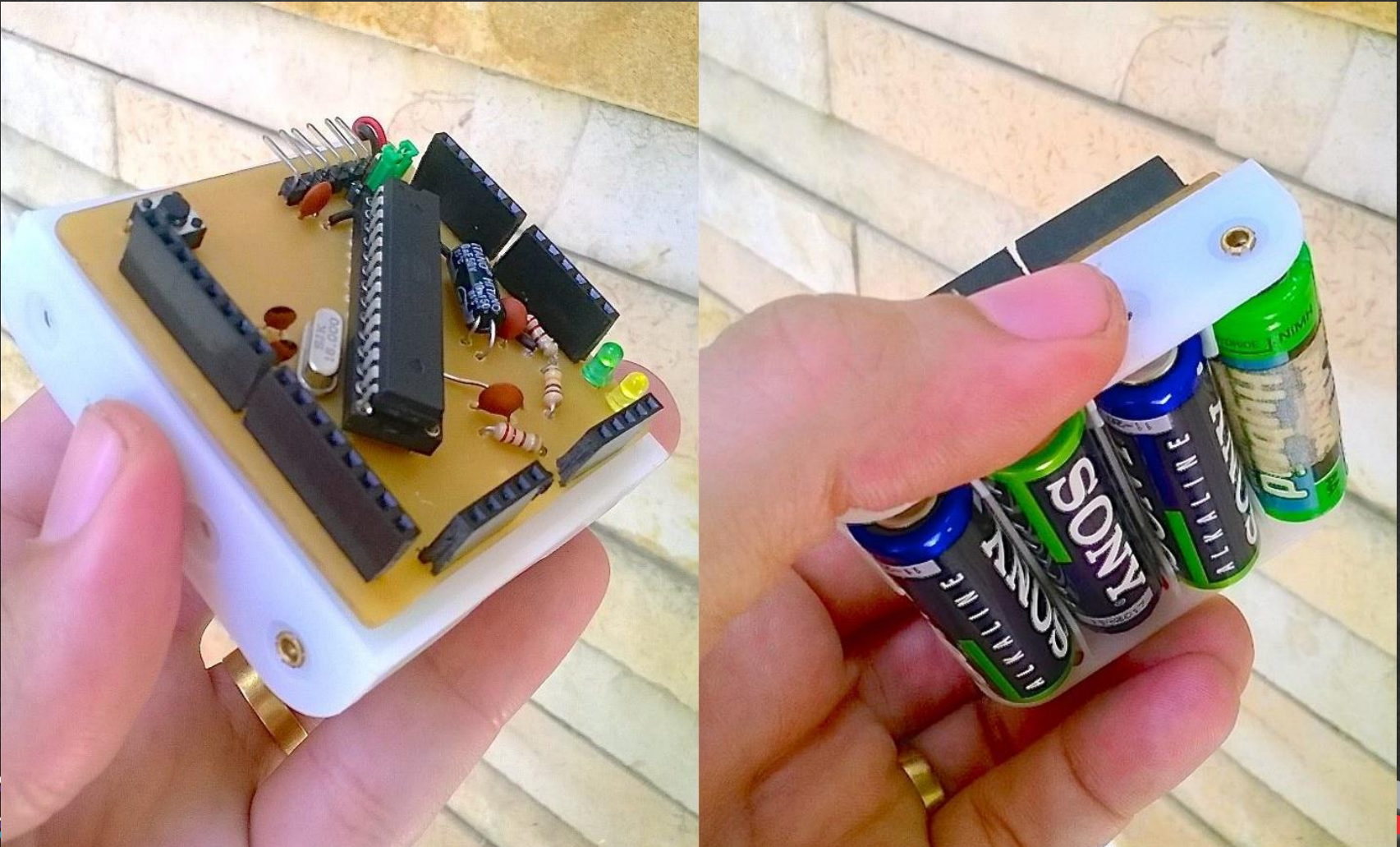
Exemplificando com o Marminino



- Maiores detalhes em:
- <http://blog.fazedores.com/marminino-o-arduino-cearense/>



Alimentação



Listagem dos componentes



Especificação	qtd	\$ +/-
Placa virgem 10cm x10cm (tam real 5,5cm x 5,5cm)	1	R\$ 2,00
Processador ATMEGA 328P-PU	1	R\$ 13,00
Soquete Estampado para Circuito Integrado 28 Pinos SLIM	1	R\$ 1,00
Barra de Pino 1x40 MCI Fêmea	2	R\$ 1,50
Barra de Pino 1x40 15mm Macho	1	R\$ 0,70
Conector tipo Barril P ₄	1	R\$ 0,10
Resistor 10k 1/4W (marrom, preto e laranja)	1	R\$ 0,25
Resistor 1k 1/4W (marrom, preto e vermelho)	2	R\$ 0,25
Led 3mm ou 5mm (cores variadas)	2	R\$ 0,15
Cristal 16Mhz	1	R\$ 0,50
Botão Chave Táctil 6x6x6mm 2 terminais	1	R\$ 0,15
Capacitor Cerâmico 22pF (pico faraday) 50V	2	R\$ 0,10
Capacitor Eletrolitico 0.1uF 50V	3	R\$ 0,15
Capacitor Eletrolitico 100uF 50V	1	R\$ 0,15
Regulador de Tensão 7805		

projeto das trilhas

