

PROJETO ARDROBOTICA

PROGRAMAÇÃO COM ARDUINOS



Arduino

O Arduino é uma plataforma de prototipagem eletrónica open-source, baseada em princípios de flexibilidade e facilidade de utilização, para hardware e software.

Consiste numa placa microcontroladora, programável preparada para receber sinais de sensores e acionar atuadores.

A linguagem de programação é baseada em Wiring (semelhante a C/C++).



Arduino – História

Em 2005, Massimo Banzi e David Cuartielles desenvolveram o Arduino, um dispositivo programável, de fácil utilização, com o objetivo de poder ser utilizado em projetos interativos de arte e design.



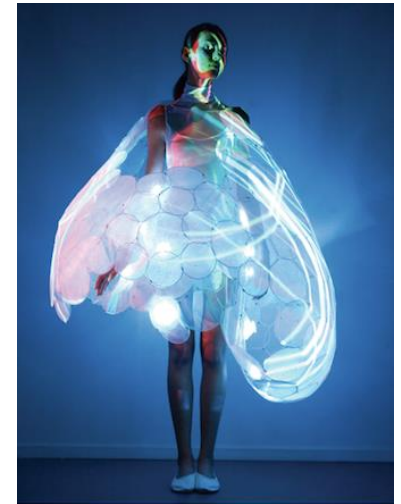
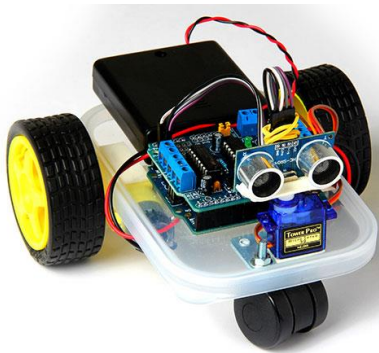
Projetaram então um dispositivo simples, de baixo custo, fácil de conectar a diversos componentes, como relés, motores ou sensores e fácil de programar.



Utilizaram um microcontrolador Atmega de 8 bits e projetaram uma placa com conexões de simples utilização, desenvolveram o firmware do bootloader para o microcontrolador e o ambiente de desenvolvimento integrado (IDE) – Software Arduino.

Um dos seus alunos, David Mellis, ficou responsável pelo desenvolvimento do software **Arduino**, baseado em Wiring.

Arduino – Aplicações



Arduino - Modelos mais comuns

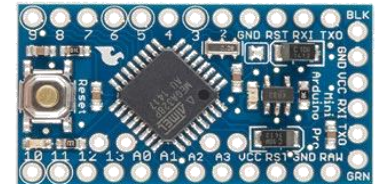
Arduino UNO



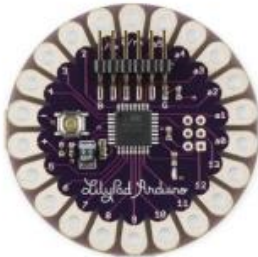
Arduino Mega



Arduino Mini



Lilypad



Arduino Nano



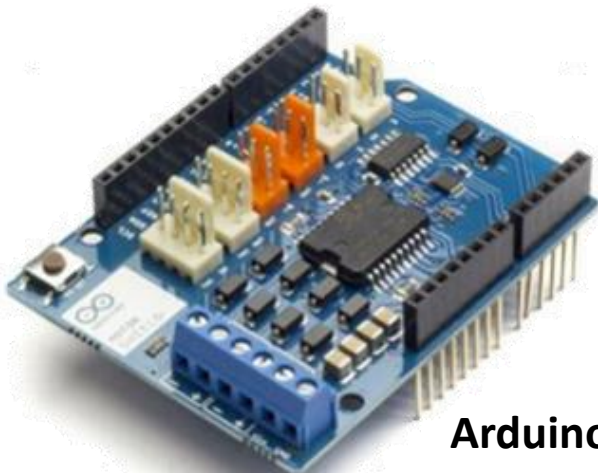
Shields (extensões) do Arduino



Arduino Ethernet Shield



Arduino GSM Shield



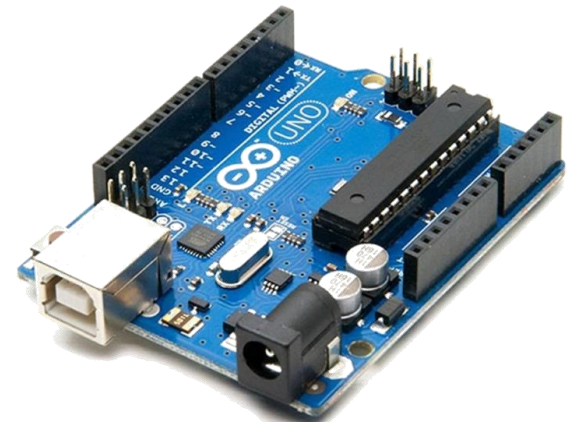
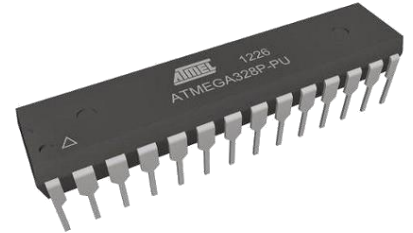
Arduino Motor Shield



Arduino Bluetooth Shield

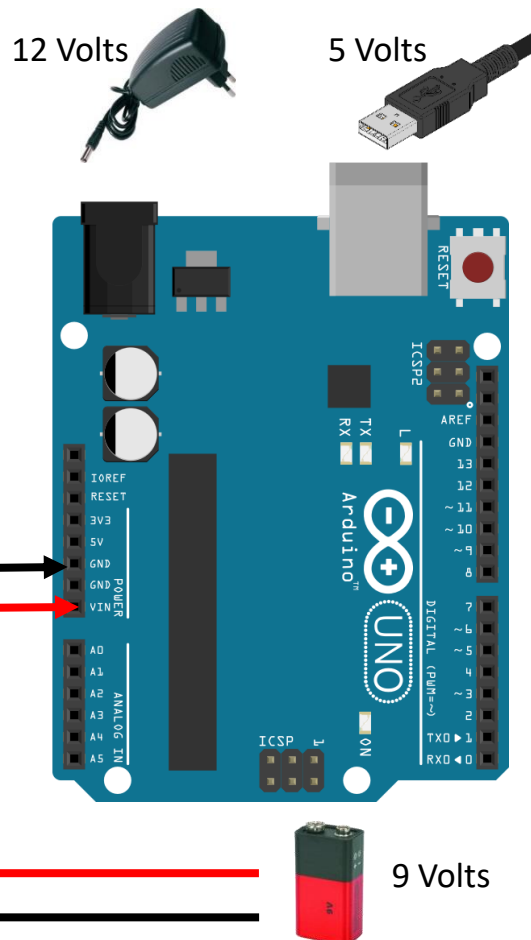
Arduino Uno – Características técnicas

- Micro controlador: ATmega328P
- Tensão de operação: 5V
- Tensão de entrada (recomendada): 7-12V
- Tensão de entrada (limite): 6-20V
- Pinos Digitais de E/S: 14 (6 pins PWM)
- Pinos Analógicos (Entrada): 6
- Corrente máxima por pino de E/S: 40mA
- Memória Flash: 32 kB (ATmega328), 0.5kB usado pelo bootloader
- SRAM: 2 kB
- EEPROM: 1 kB
- Velocidade de relógio: 16 MHz

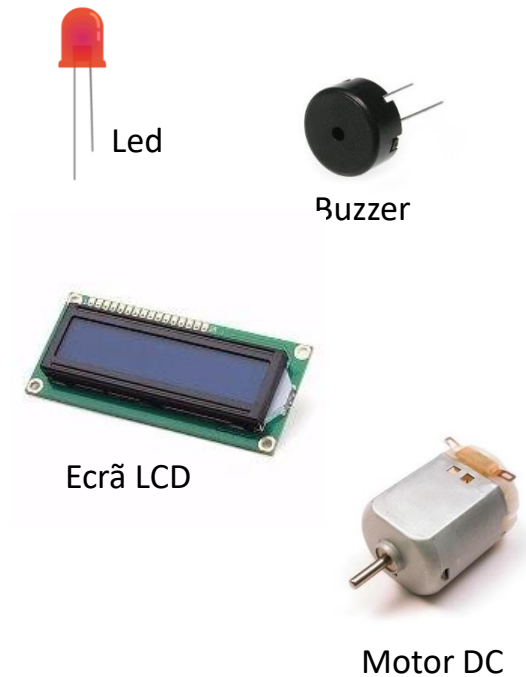


Componentes eletrônicos simples

Input (Ler)



Output (Escrever)



VIN - pino para alimentar a placa através de uma pilha/bateria externa

Explorando o Arduino UNO

3,3V - Fornece tensão de 3,3V para alimentação de shields e módulos externos.

5V - Fornece tensão de 5 V para alimentação de shields e circuitos externos.

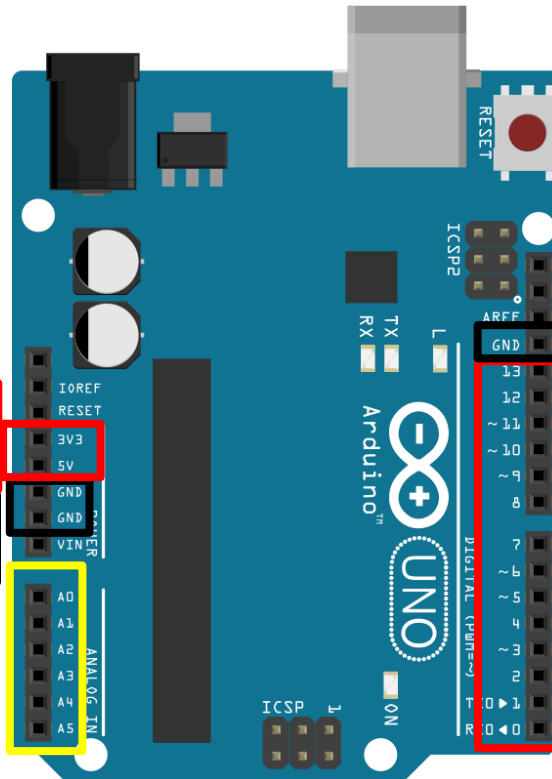
3,3 Volts

5 Volts

GROUND - 0 Volts (GND)

Pinos Analógicos:

A0 – A5 (Input)



GND - pinos de referência, terra.

Pinos Digitais:

0 – 13 (Input / Output)

~3
~5
~6

Exemplo:

PWM

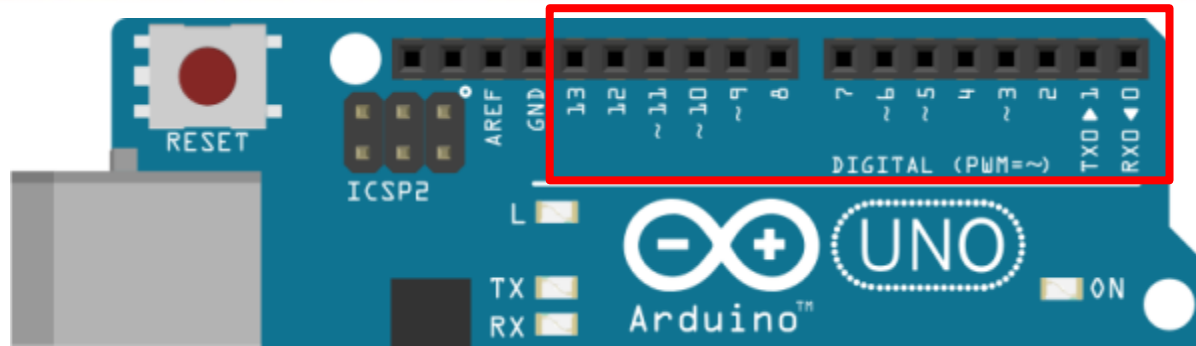
~9
~10
~11



Para variar a velocidade de um motor DC



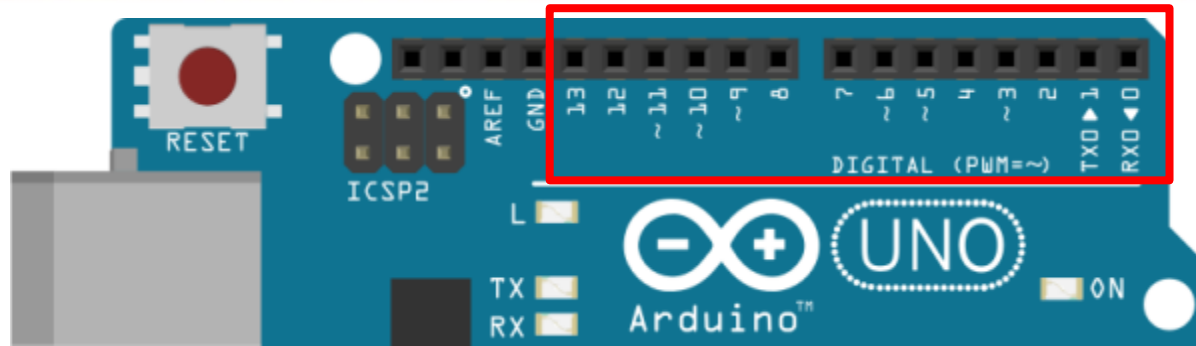
Explorando o Arduino UNO



Pinos digitais operam a **5V**, onde cada pino pode fornecer ou receber uma corrente máxima de **40 mA**. Cada pino possui uma resistência de **pull-up interno (20-50 kOhms)** que pode ser ativada por software.

Digitais: 0 a 13 podem ser usados como entradas/saídas digitais através das função **digitalWrite()** e **digitalRead()** como input ou output através da função **pinMode()**;

Explorando o Arduino UNO

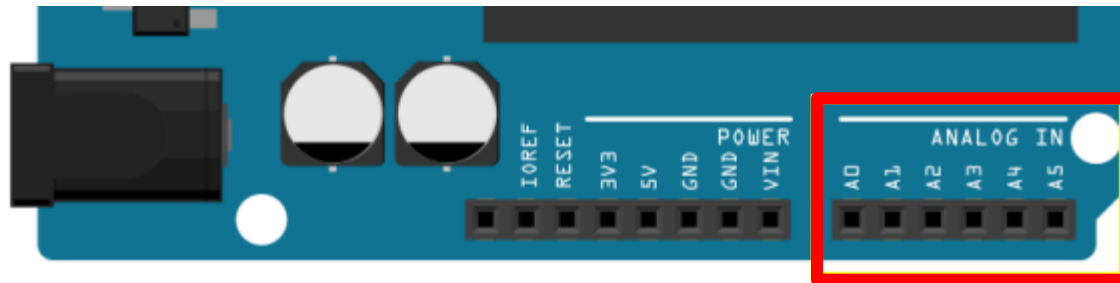


PWM: 3,5,6,9,10 e 11 podem ser usados como saídas PWM de 8 bits através da função `analogWrite()` – Output Analógico;

Serial: 0 (RX) and 1 (TX). Usados para receber (RX) e transmitir (TX);

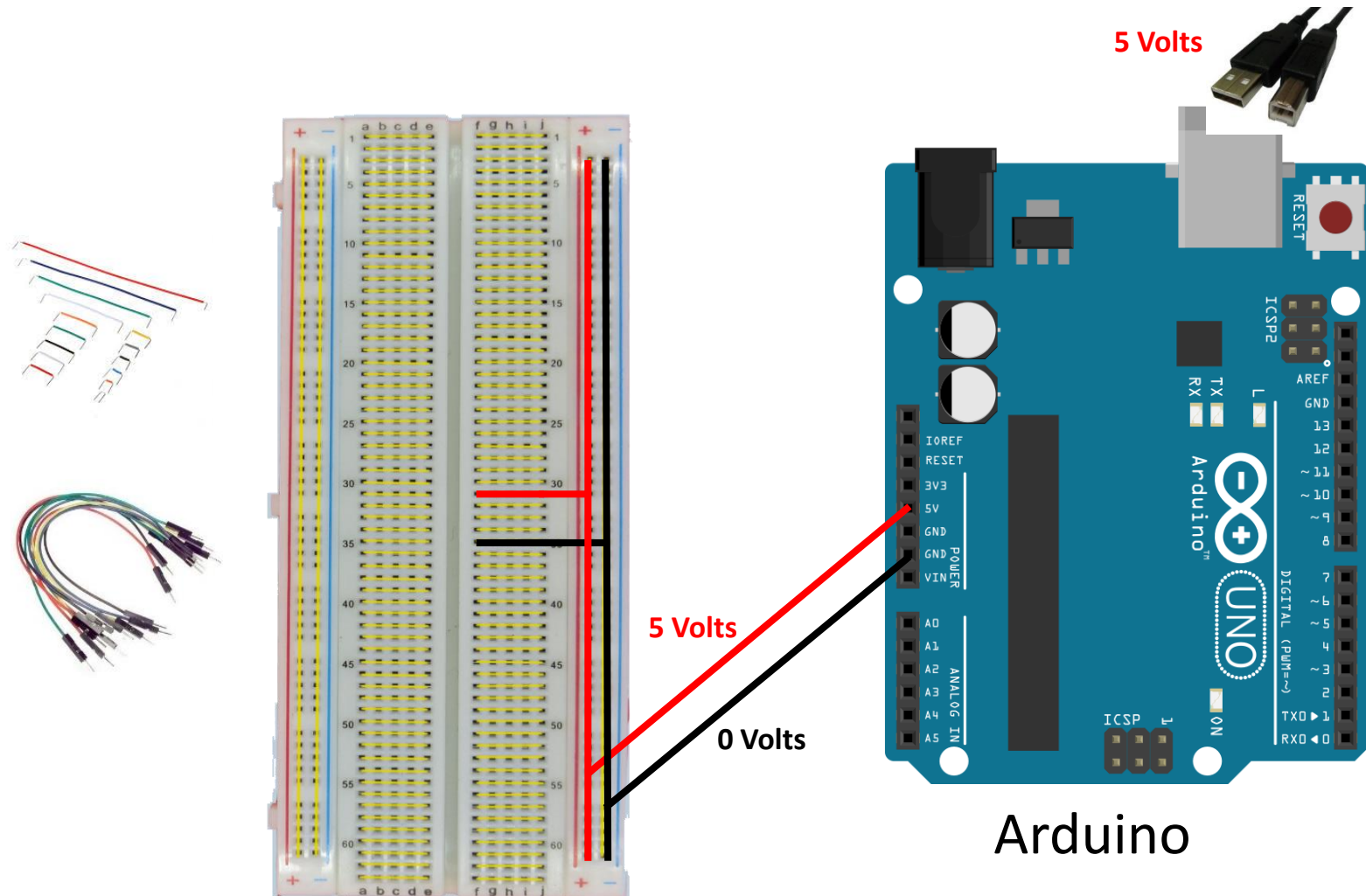
LED 13: Existe um LED embutido no pin digital 13.

Explorando o Arduino UNO

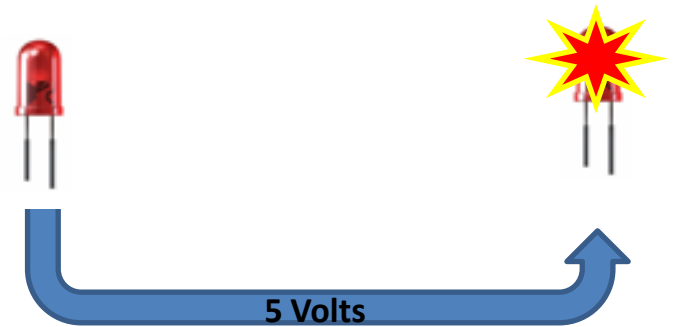
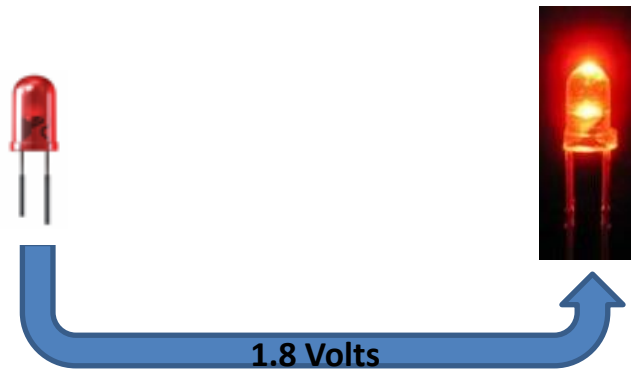


Analógicas: 0 a 5 podem ser usados como entradas analógicas através da função **`analogRead()`**, cada pin disponibiliza 10bits (ou seja valores de **0 a 1023**). Por defeito debitam entre **0 e 5 volts**;

Placa de ensaio (Breadboard)



Led – Díodo Emissor de Luz



Led – Díodo Emissor de Luz

Resistência – qual utilizar?

- Tensão de saída de uma porta digital do Arduino = 5v
- Tensão de funcionamento do led vermelho = 1,8v
- Corrente máxima do led = 0,02 A (20mA)

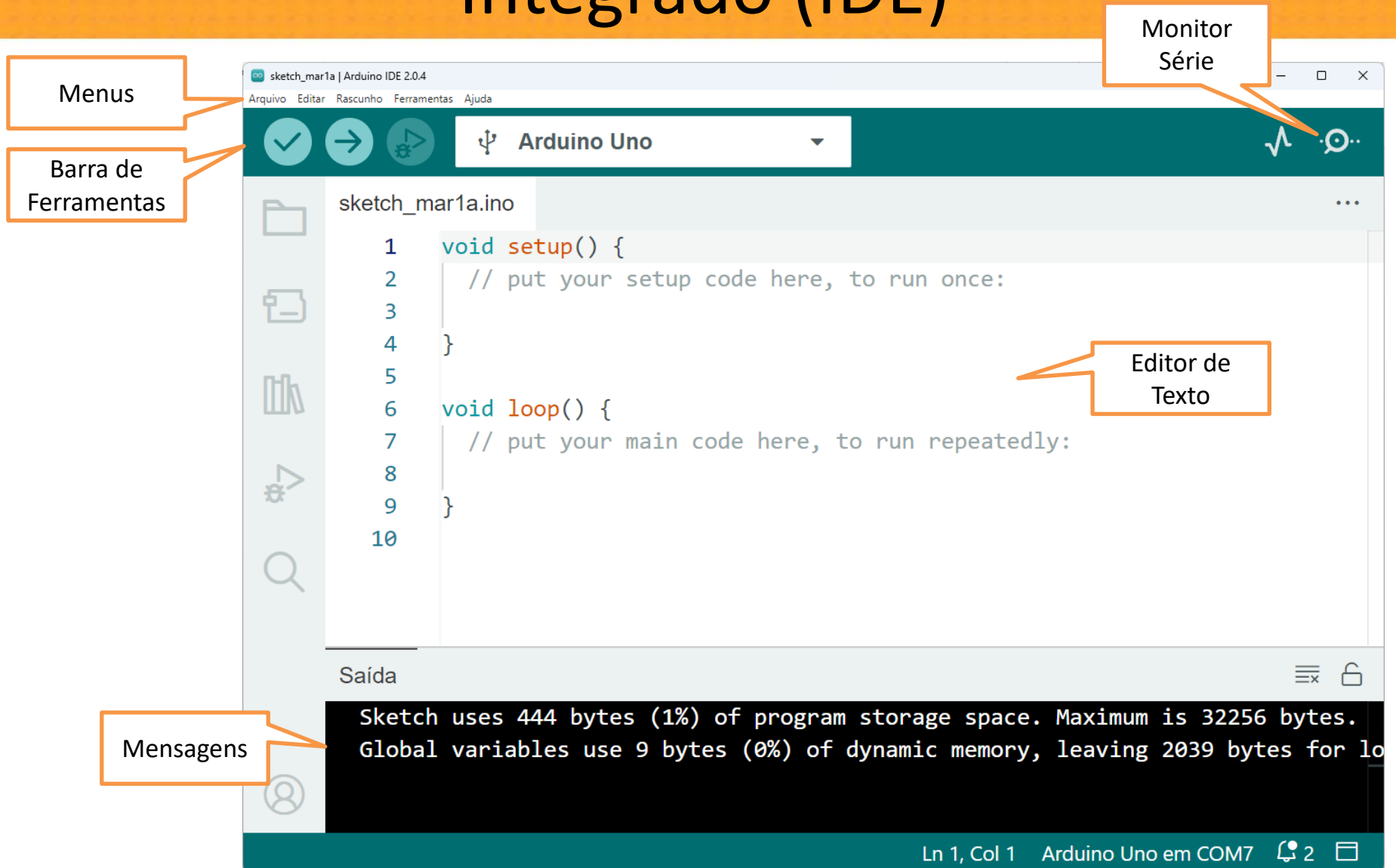
Lei de Ohm:
$$R = \frac{V}{I} = \frac{5V - 1,8V}{0,02} = 160\Omega$$



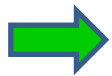
Não se deve utilizar uma resistência MENOR do que o valor recomendado, pois o led pode queimar.

Caso se utilize um resistência MUITO elevada, o led não acenderá.

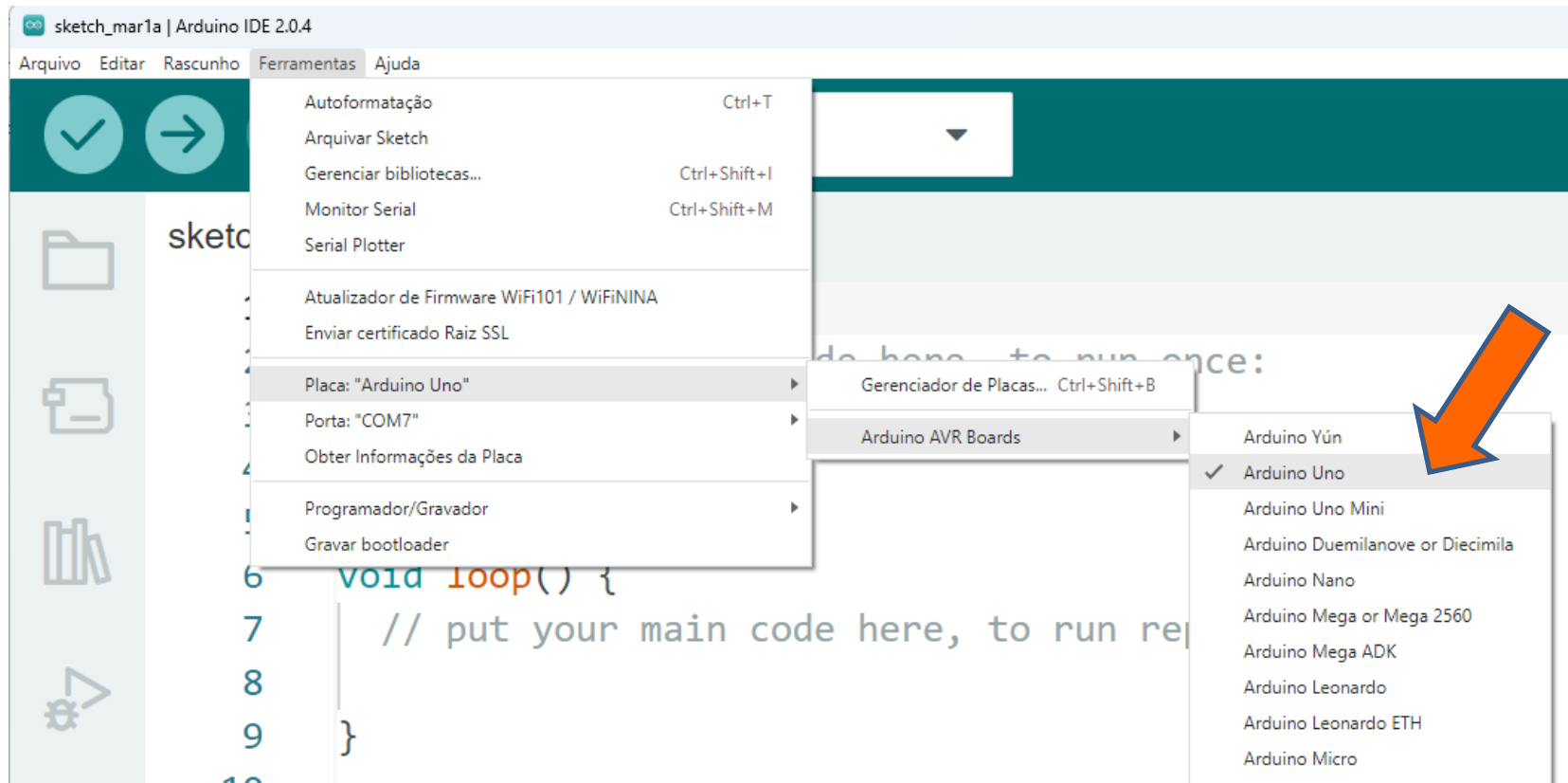
Ambiente de Desenvolvimento Integrado (IDE)



Ambiente de Desenvolvimento Integrado (IDE)

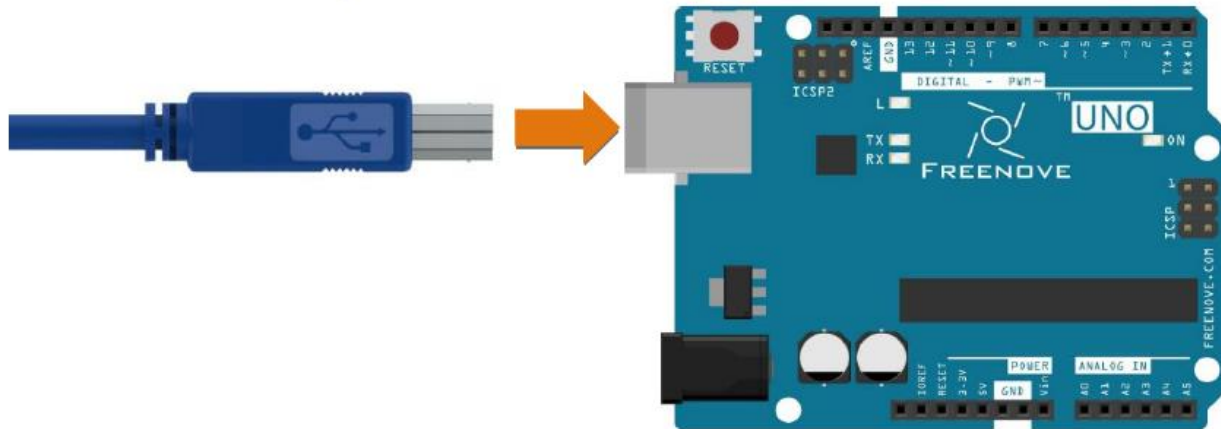


1º - Selecionar a placa

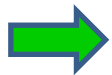


Ambiente de Desenvolvimento Integrado (IDE)

➡ 2º - Ligar o cabo USB ao Arduino após verificação das ligações



Ambiente de Desenvolvimento Integrado (IDE)



3º - Selecionar a porta

Verificar código

Enviar código

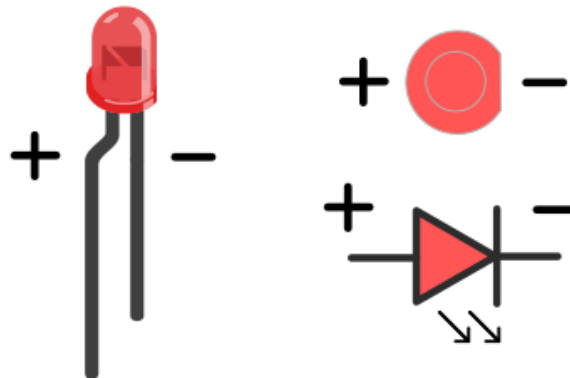
Arquivo Editar Rascunho Ferramentas Ajuda

- Autoformatação Ctrl+T
- Arquivar Sketch
- Gerenciar bibliotecas... Ctrl+Shift+I
- Monitor Serial Ctrl+Shift+M
- Serial Plotter
- Atualizador de Firmware WiFi101 / WiFinina
- Enviar certificado Raiz SSL
- Placa: "Arduino Uno"
- Porta: "COM7"
 - Serial ports
 - COM1
 - ✓ COM7
- Obter Informações da Placa
- Programador/Gravador
- Gravar bootloader

```
6 void loop() {  
7     // put your main code here, to run repeatedly:  
8  
9 }  
10
```

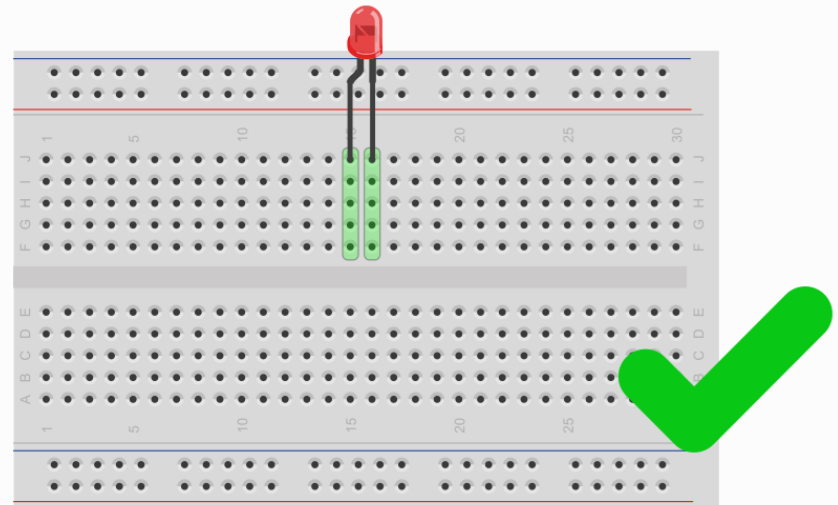
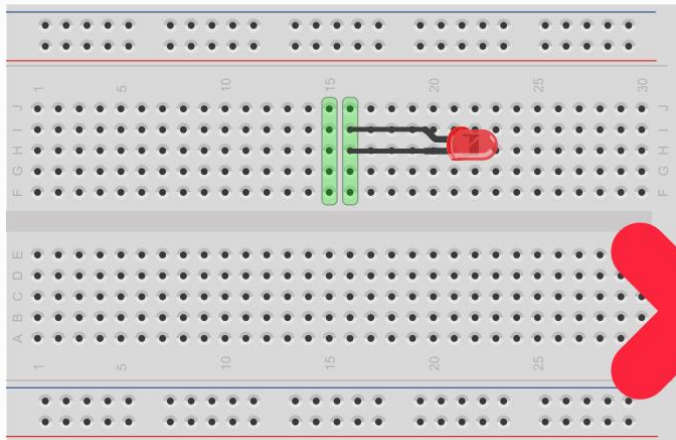
Atividade 1 – Acender um led

- O led tem polaridade, só emite luz se corretamente ligado.
- Certifique-se que a linha da placa de ensaio onde está ligada a perna mais longa do led (ânodo) está ligada a um pino digital do Arduino.
- A perna mais curta (cátodo) do led deve ser ligada à resistência e daí o circuito fecha no terminal GND (terra).




Atividade 1 – Acender um led


- Os dois terminais do LED não podem ficar na mesma fila da placa de ensaio. Este princípio de montagem aplica-se a outros componentes.



Atividade 1 – Acender um led

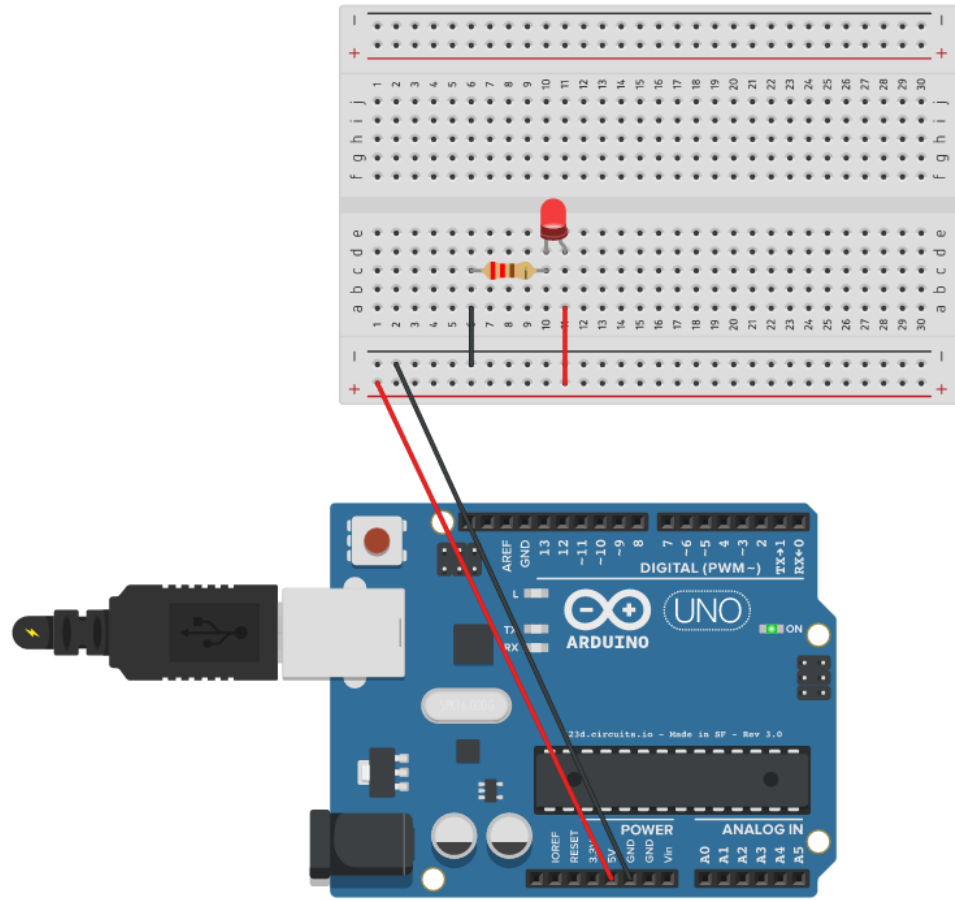
Material necessário:

Led  (Output Digital)

Resistência 

Fios 

Atenção:



Programação - Funções

setup()

Esta função é executada quando o Arduino é ligado. Serve para definir configurações iniciais de portas, comunicações, etc...

loop()

Esta função entra em execução logo após a função setup. O código que está dentro desta função é executado de forma repetitiva, indefinidamente.

delay(parâmetro)

Esta função permite colocar um intervalo em milissegundos entre uma instrução e outra. Recebe como parâmetro um valor inteiro

Programação - Funções

`pinMode(parâmetro 1, parâmetro 2)`

Configura uma porta digital, podendo ela ser de entrada(INPUT) ou saída (OUTPUT). Recebe no parâmetro 1 o número da porta digital, e no parâmetro 2 é definido se a porta é de entrada ou saída.

`digitalWrite(parâmetro 1,parâmetro 2)`

Escreve na porta digital selecionada. Recebe no parâmetro 1 o número da porta digital, e no parâmetro 2 recebe um valor: ligado(HIGH – 5V) ou desligado(LOW- 0V)

Programação - Funções

`analogWrite(parâmetro 1, parâmetro2)`

Escreve numa porta digital PWM. Recebe no parâmetro 1 o número da porta digital PWM e no parâmetro2 recebe um valor entre 0 e 255.

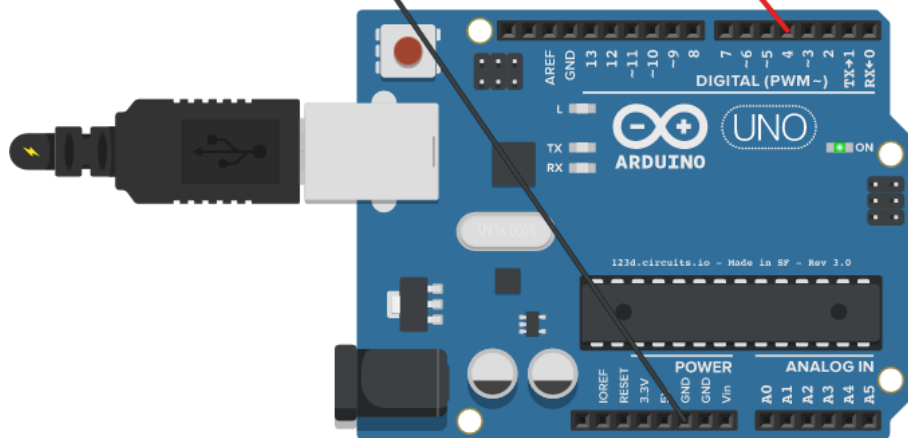
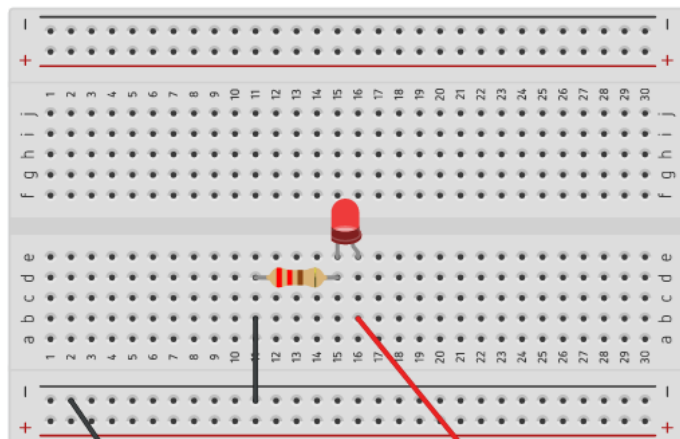
`digitalRead(parâmetro)`

Efetua leituras na porta digital selecionada. Recebe apenas um parâmetro com o número da porta digital. Valores recebidos: 0 ou 1


`analogRead(parâmetro)`


Efetua leituras na porta analógica selecionada. Recebe apenas um parâmetro com o número da porta analógica. Valores recebidos: Entre 0 e 1023

Atividade 2 – Led a piscar (Blink)



Material necessário:

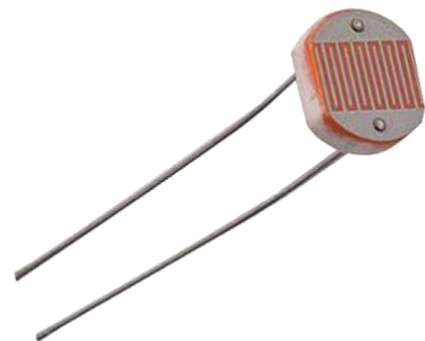
Led  (Output Digital)

Resistência 220 Ω 

Fios 

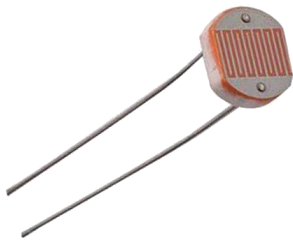
Sensor de luz LDR

- As foto-resistências LDR (Light-Dependent Resistor) representam um tipo de resistências cujo valor resistivo varia em função da luz, o que as torna especialmente úteis em sistemas automáticos.
- Devido às suas características poderão funcionar como sensores de luminosidade.
- A foto-resistência LDR é composta essencialmente por Sulfeto de cádmio (CdS).



Sensor de luz LDR

- O cádmio reage à luz, deixando que os seus eletrões se movam livremente, permitindo assim a passagem da corrente elétrica.
- Quanto maior for a luminosidade, menor será a resistência elétrica.
- Quanto menor for a luminosidade, maior será a resistência elétrica.



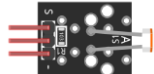
Quando existe muita luminosidade, os valores obtidos pelo LDR serão mais elevados




Luz de Presença

Atividade 3 – Sensor de luz LDR

Material necessário:

1 Sensor de luz  (Input Analógico)

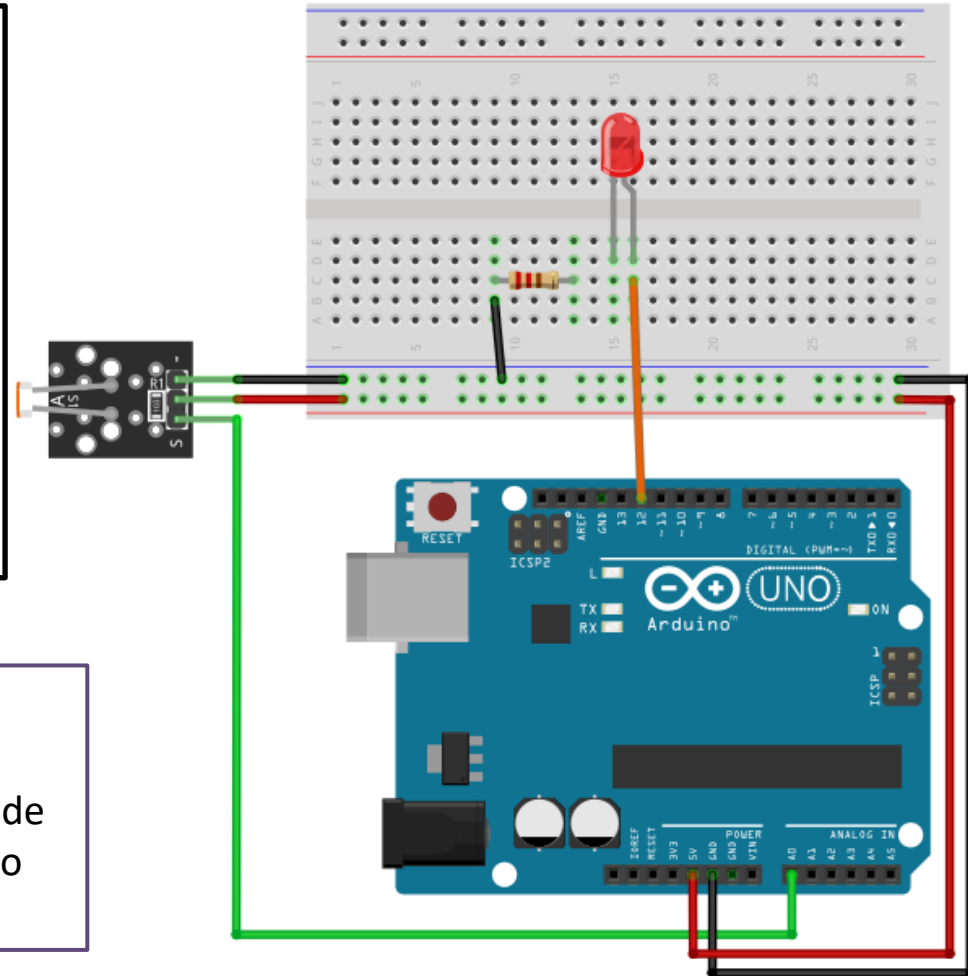
1 Led  (Output Digital)

1 Resistência 220Ω 

Fios 

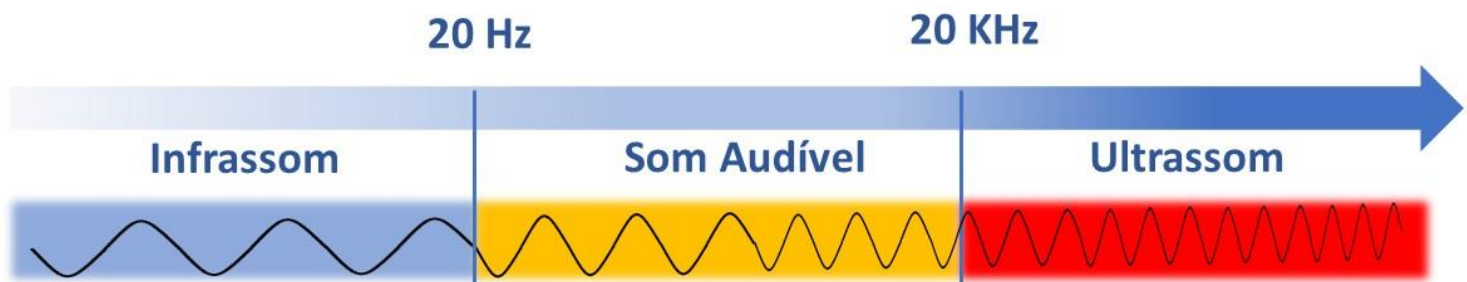
Nota:

Nesta atividade pretende-se simular uma luz de presença. O led deverá acender quando o sensor de luz LDR registrar valores abaixo de um determinado limiar.



Sensores Ultrassónicos

- Os sensores ultrassónicos possuem um emissor (trigger) e um recetor (echo) de ondas sonoras.
- O seu funcionamento baseia-se na emissão de uma onda sonora de alta frequência (40khz), impercetível ao ouvido humano.
- Ao ser emitida a onda sonora, é acionada uma espécie de relógio de alta precisão, que mede o tempo entre a colisão da onda sonora com um obstáculo e a reflexão de volta ao recetor do sensor.



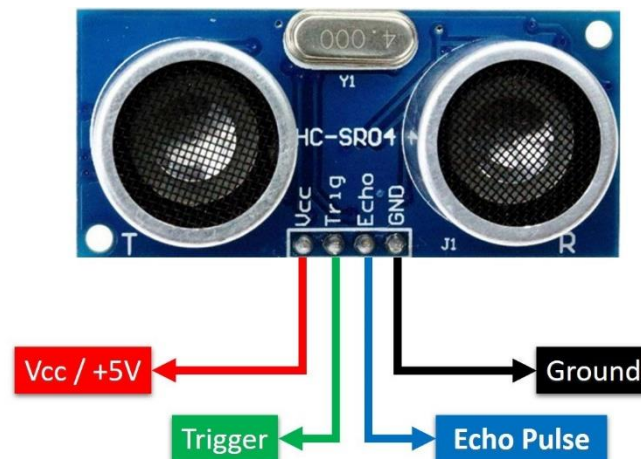
Sensores Ultrassônicos

- Através do valor da velocidade do som e tendo em conta o tempo que o sinal demorou a colidir com o obstáculo e retornar, é possível, determinar qual a distância percorrida entre sensor e obstáculo.



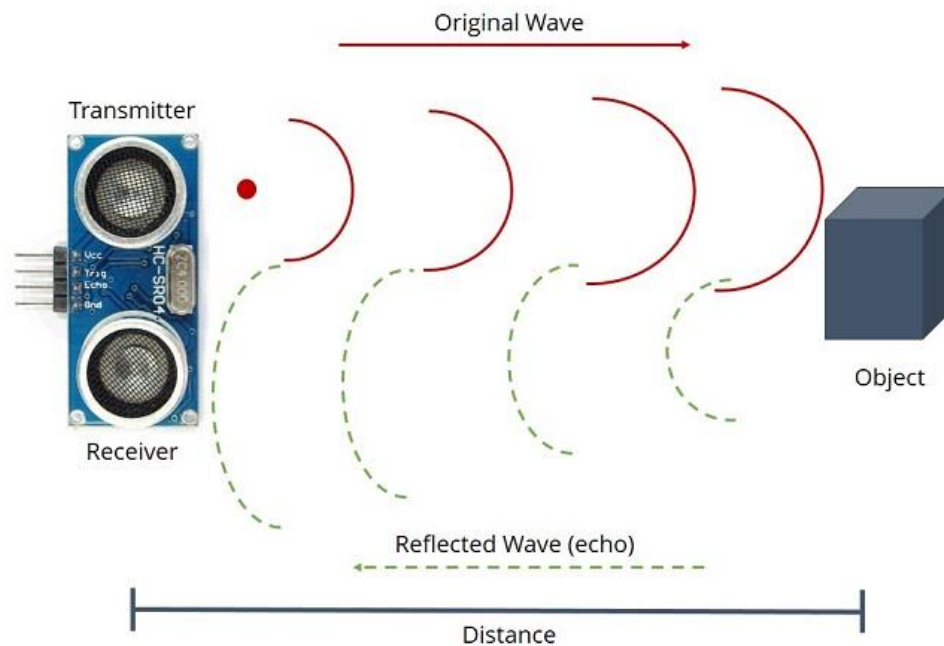
HC-SR04

- O módulo HC-SR04 é um sensor de distância composto por um emissor e um recetor, com capacidade de medir distâncias de 2cm até 4m, com precisão aproximada de 3mm.
- Este sensor emite sinais ultrassónicos que refletem no objeto a ser atingido e retornam ao sensor, indicando a distância do alvo.



HC-SR04

- A velocidade do sinal ultrassônico emitida pelo sensor corresponde à velocidade do som, aproximadamente 340 m/s.
- Se o sensor estiver a uma distância x do objeto, o sinal percorrerá 2 vezes a distância em relação ao objeto.

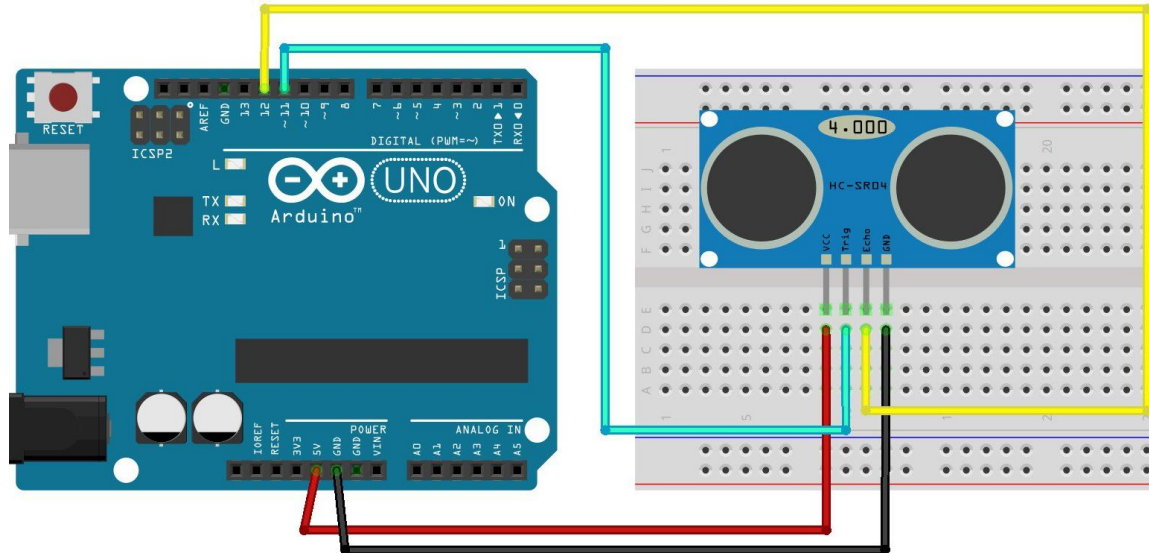


Atividade 4 – Sensor Ultrassónico

Material necessário:

1 Sensor Ultrassónico
(Input/Output Digital)

Fios



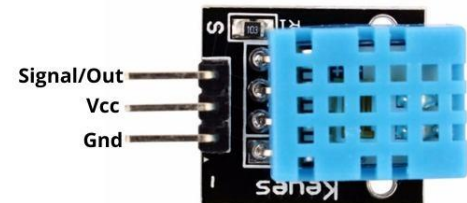
Nota:

Nesta atividade pretende-se construir um circuito que permita determinar a distância de um determinado objeto através de um sinal ultrassónico.

Sensor de Humidade e Temp. DHT11

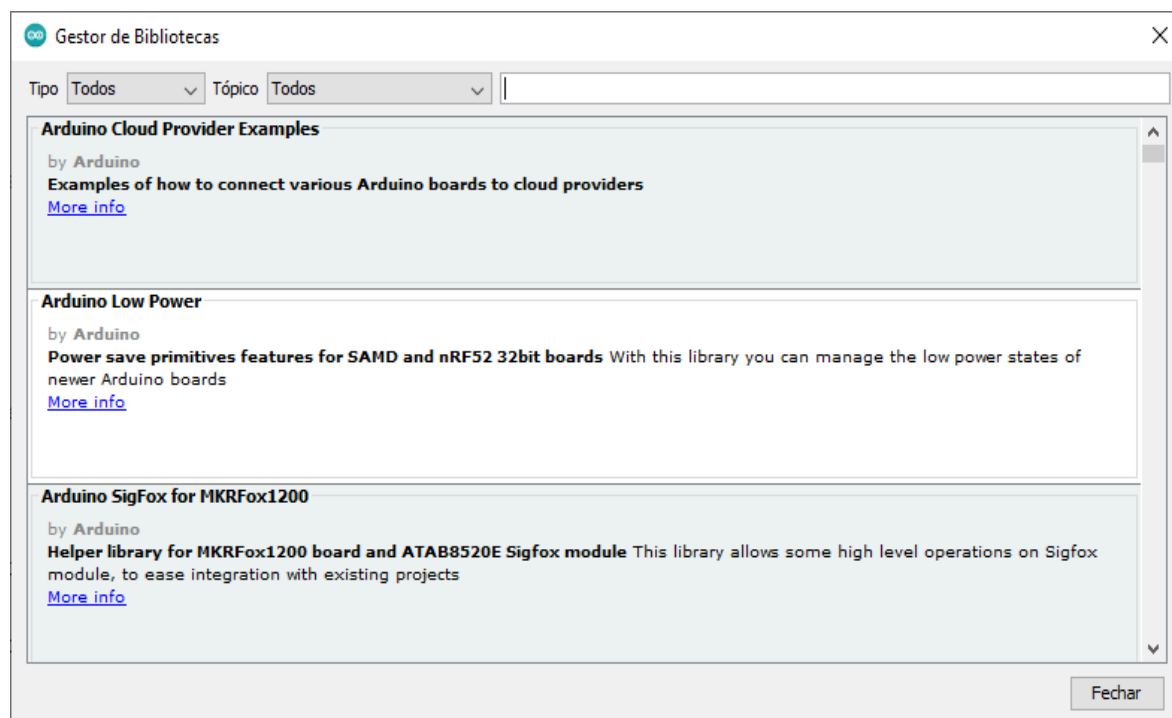
KY-015 (Opcional)

- O DHT11 KY-015 é um sensor de humidade e temperatura que permite fazer leituras de temperaturas entre 0 a 50° Celsius e humidade entre 20 a 95%.
- Transmissão de sinal: até 20m
- Amplitude de medição de humidade: 20 – 95% RH
- Incerteza de medição de humidade: + / -5%
- Amplitude de medição de temperatura: 0 – 50°C
- Incerteza de medição de temperatura: + / -2°C
- Tensão Operacional: 3.3 – 5V
- Sinal de saída: digital
- Dimensões do PCB: 32 x 14mm
- Peso: 8g



Sensor de Humidade e Temp. DHT11 KY-015 (Opcional)

- Para facilitar a programação do sensor DHT11 KY-015 deve instalar uma biblioteca designada por DHT Sensor Library da Adafruit, através do Gestor de Bibliotecas.

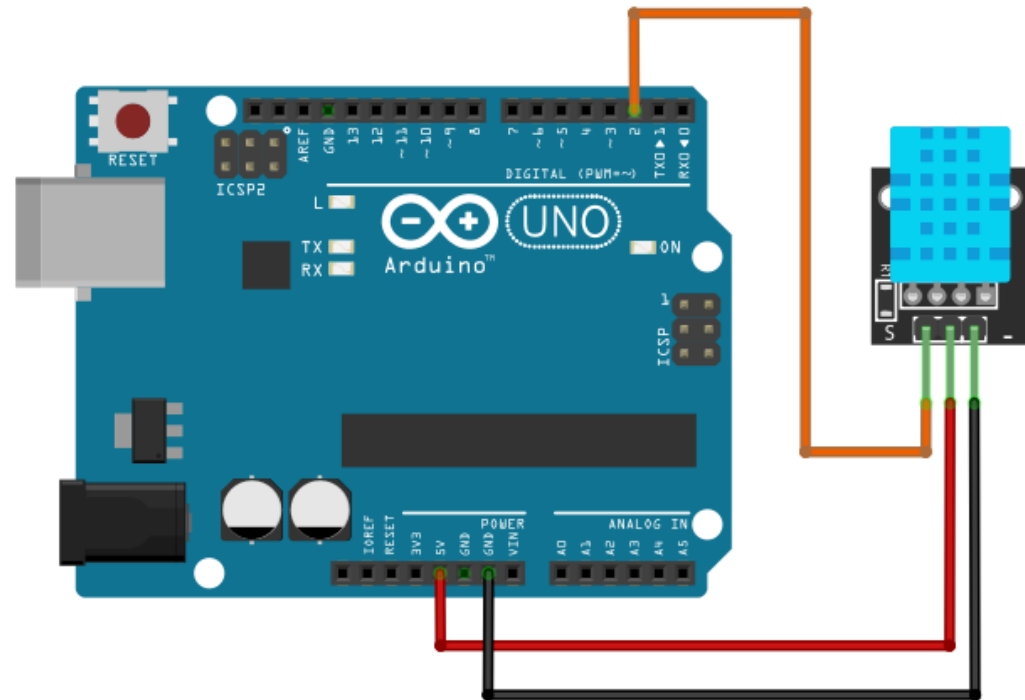
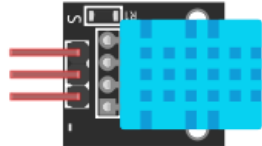


Atividade 5 - Sensor de Temperatura

Material necessário:

1 Sensor Temperatura (Input Digital)

Fios



Nota:

Nesta atividade pretende-se construir um circuito que permita medir os valores de humidade e temperatura ambiente de uma sala, utilizando um sensor de temperatura DHT11. Os valores deverão ser apresentados numa folha de cálculo.

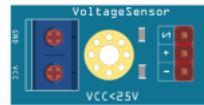
Atividade 6 - Sensor de Tensão

Material necessário:

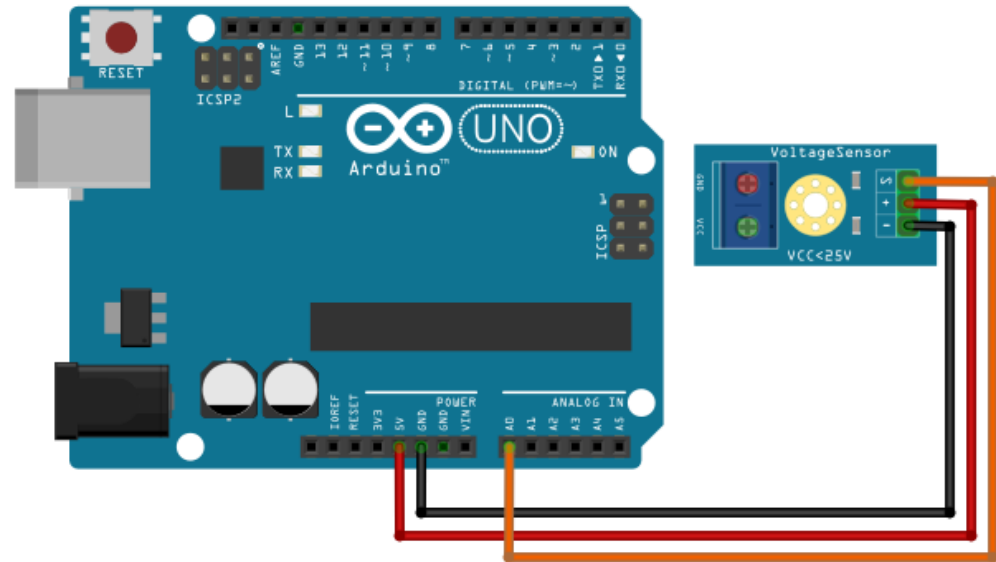
1 Pilha 9V



1 Sensor de Tensão (Input Analógico)



Fios



Nota:

Nesta atividade pretende-se construir um circuito que permita medir os valores de tensão de uma fonte de energia ou de um circuito

Atividade 6 - Sensor Capacitivo

Material necessário:

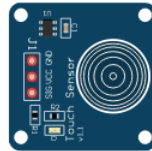
1 led (Output Digital)



1 Resistência 220 Ω



1 Sensor Capacitivo (Input Digital)

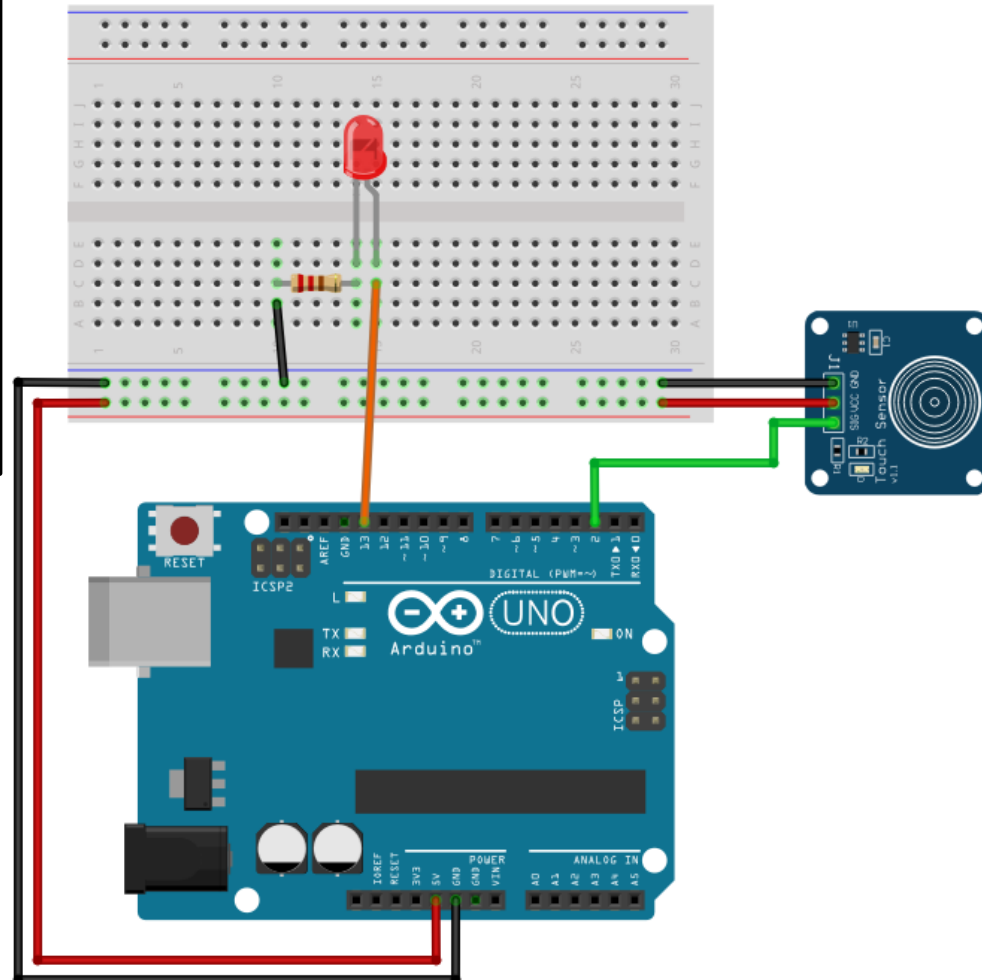


Fios



Nota:

Nesta atividade pretende-se construir um circuito que acender um led sempre que o sensor capacitivo for pressionado



Fim



The End