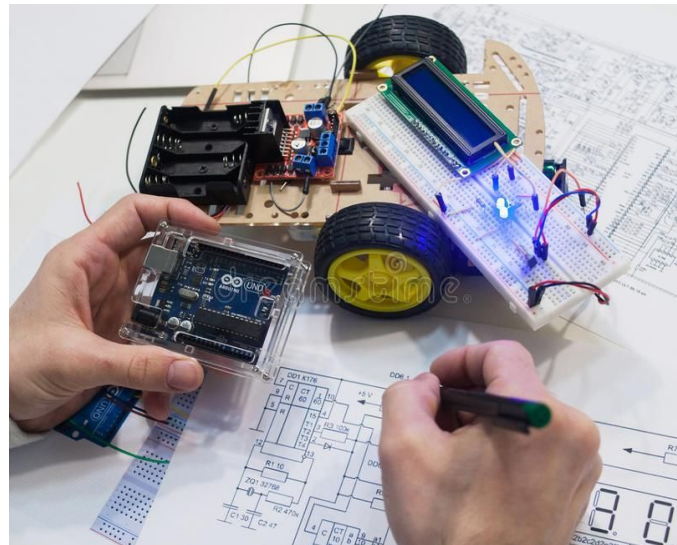
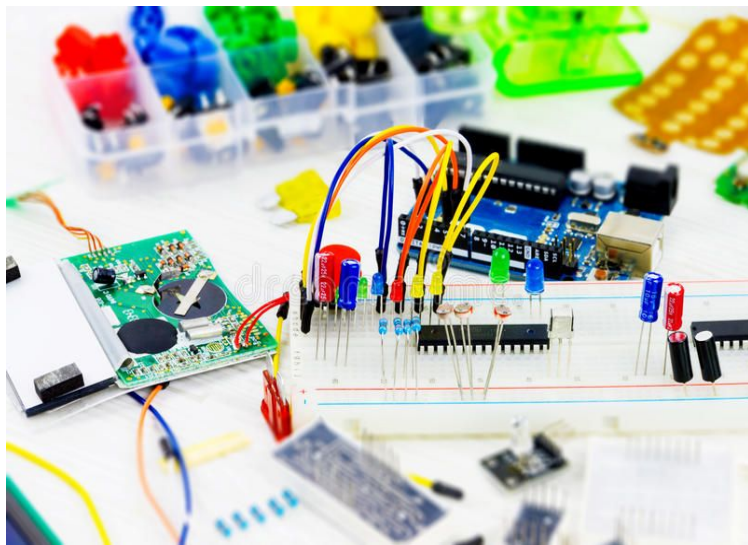


Laboratório de Introdução à Engenharia de Computação

Aula 8 - Introdução Microcontroladores



O que precisamos saber?

- Eletrônica básica
- Componentes eletrônicos
- Programação (C, C++)

Arduino / Esp32

Existem diversas placas de desenvolvimento que possui um microcontrolador integrado.

Exemplos: arduino, esp e outros.



Esp32

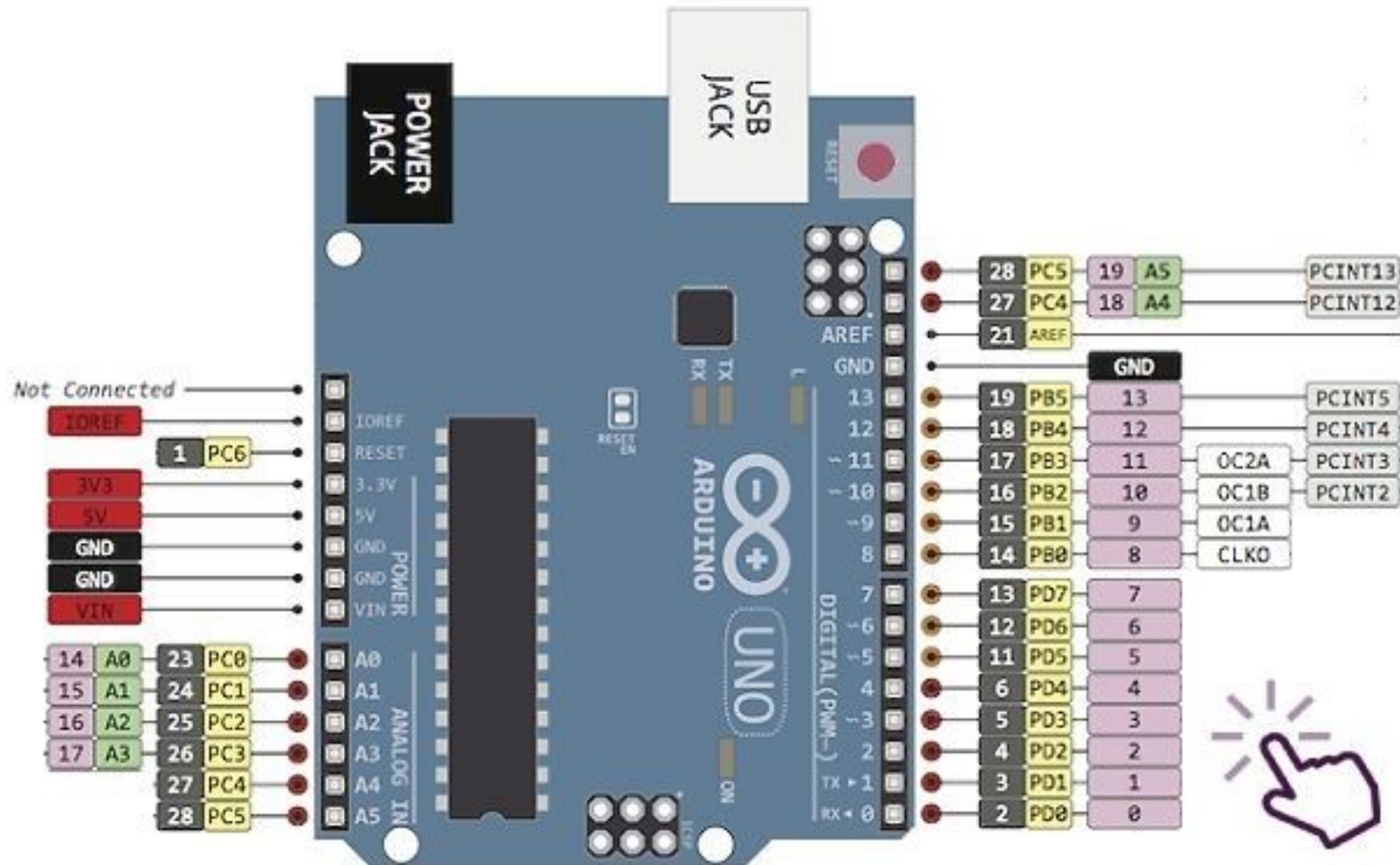
Existem diversas placas de desenvolvimento que possui um microcontrolador integrado.

Exemplos: arduino, esp e outros.

Uma dessas plataformas é a ESP32 e ESP8266

São processadores dual core, com Wifi (ESP8266) e Bluetooth e Wifi (ESP32)

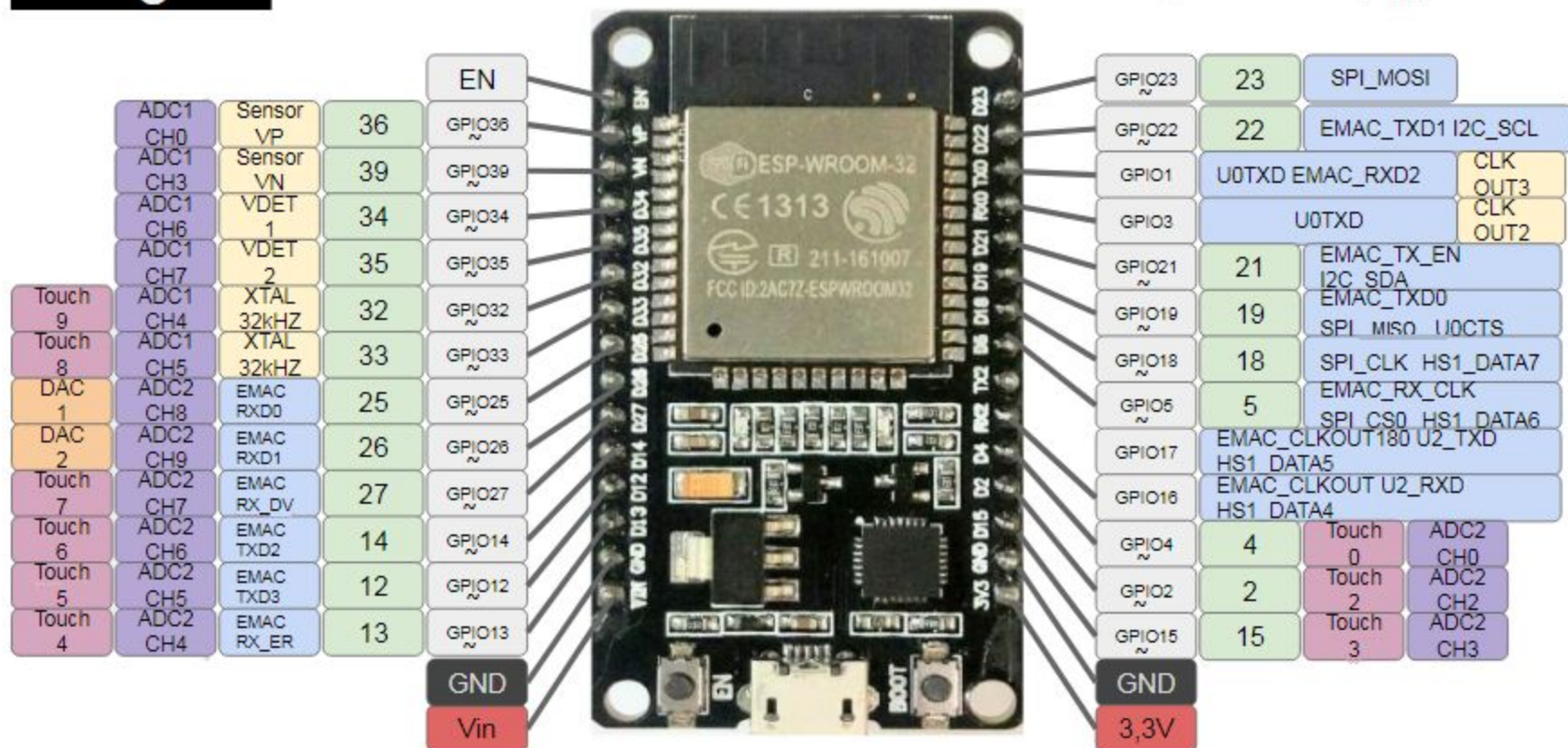




ESP32

Pinagem

- Touch
- Analógico
- Serial
- Pino
- Arduino
- Controle
- DCA
- ~ PWM



SOMENTE ao usar ESP32
Configurando o ambiente

Antes de tudo!

Download Arduino IDE

Preferências

Configurações Rede

Local do Sketchbook:

C:\Users\Teixeira\Documents\Arduino

Navegador

Idioma do editor: Português (Brasil) (Portuguese (Brazil)) (requer reinicialização do Arduino)

Tamanho da fonte do editor: 18

Escala de interface: ☒ Automático 100 % (requer reinicialização do Arduino)

Theme: Default theme (requer reinicialização do Arduino)

Mostrar mensagens de saída durante: ☐ compilação ☐ carregar

Avisos do compilador: Nenhum

☐ Mostrar números de linhas

☐ Habilitar Dobramento de Código

☒ Verificar código depois de carregar

☐ Usar editor externo

☒ Núcleo compilado em cache agressivamente

☒ Checar atualizações ao iniciar

☒ Atualizar arquivos de sketch para nova extensão ao salvar (.pde -> .ino)

☒ Salve ao verificar ou carregar

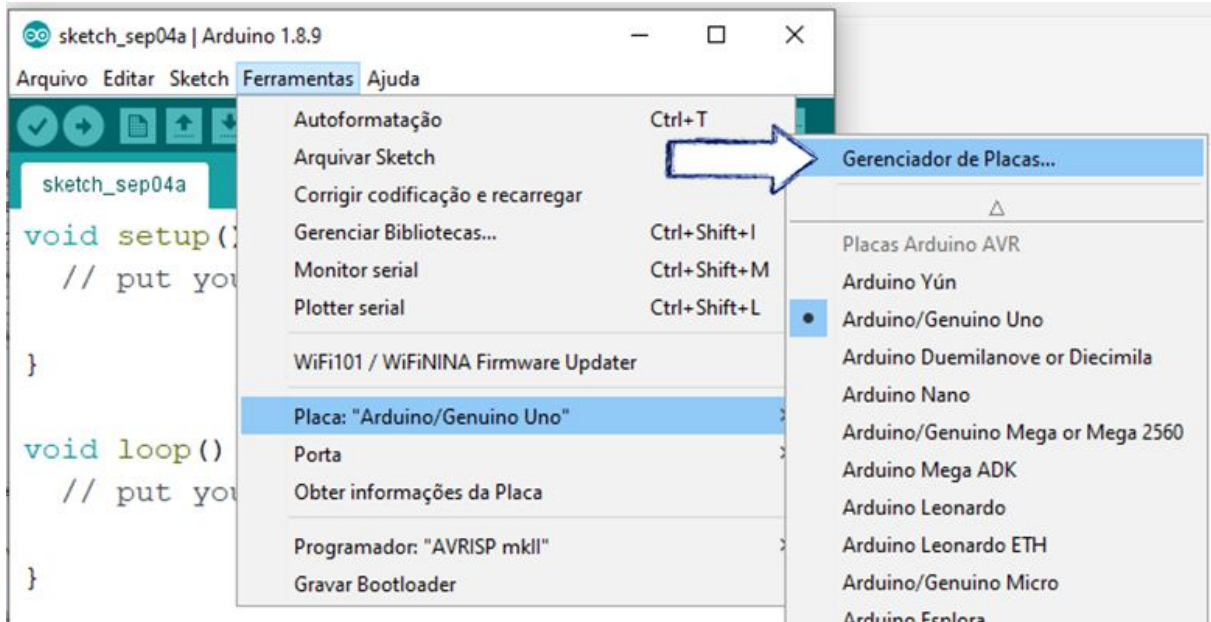
URLs Adicionais para Gerenciadores de Placas: https://dl.espressif.com/dl/package_esp32_index.json

Mais preferências podem ser editadas diretamente no arquivo

C:\Users\Teixeira\AppData\Local\Arduino15\preferences.txt

(editar apenas quando o Arduino não estiver em execução)

OK Cancelar



sketch_sep09a | Arduino 1.8.9

Arquivo Editar Sketch Ferramentas Ajuda

Autoformatação Ctrl+T

Arquivar Sketch

Corrigir codificação e recarregar

Gerenciar Bibliotecas... Ctrl+Shift+I

Monitor serial Ctrl+Shift+M

Plotter serial Ctrl+Shift+L

WiFi101 / Wi-FiNINA Firmware Updater

Placa: "ESP32 Dev Module"

Upload Speed: "921600"

CPU Frequency: "240MHz (WiFi/BT)"

Flash Frequency: "80MHz"

Flash Mode: "QIO"

Flash Size: "4MB (32Mb)"

Partition Scheme: "Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)"

Core Debug Level: "Nenhum"

PSRAM: "Disabled"

Porta

Obter informações da Placa

Programador: "AVRISP mkII"

Gravar Bootloader

Gerenciador de Placas...

▲

Arduino Leonardo ETH

Arduino/Genuino Micro

Arduino Esplora

Arduino Mini

Arduino Ethernet

Arduino Fio

Arduino BT

LilyPad Arduino USB

LilyPad Arduino

Arduino Pro or Pro Mini

Arduino NG or older

Arduino Robot Control

Arduino Robot Motor

Arduino Gemma

Adafruit Circuit Playground

Arduino Yún Mini

Arduino Industrial 101

Linino One

Arduino Uno WiFi

ESP32 Arduino

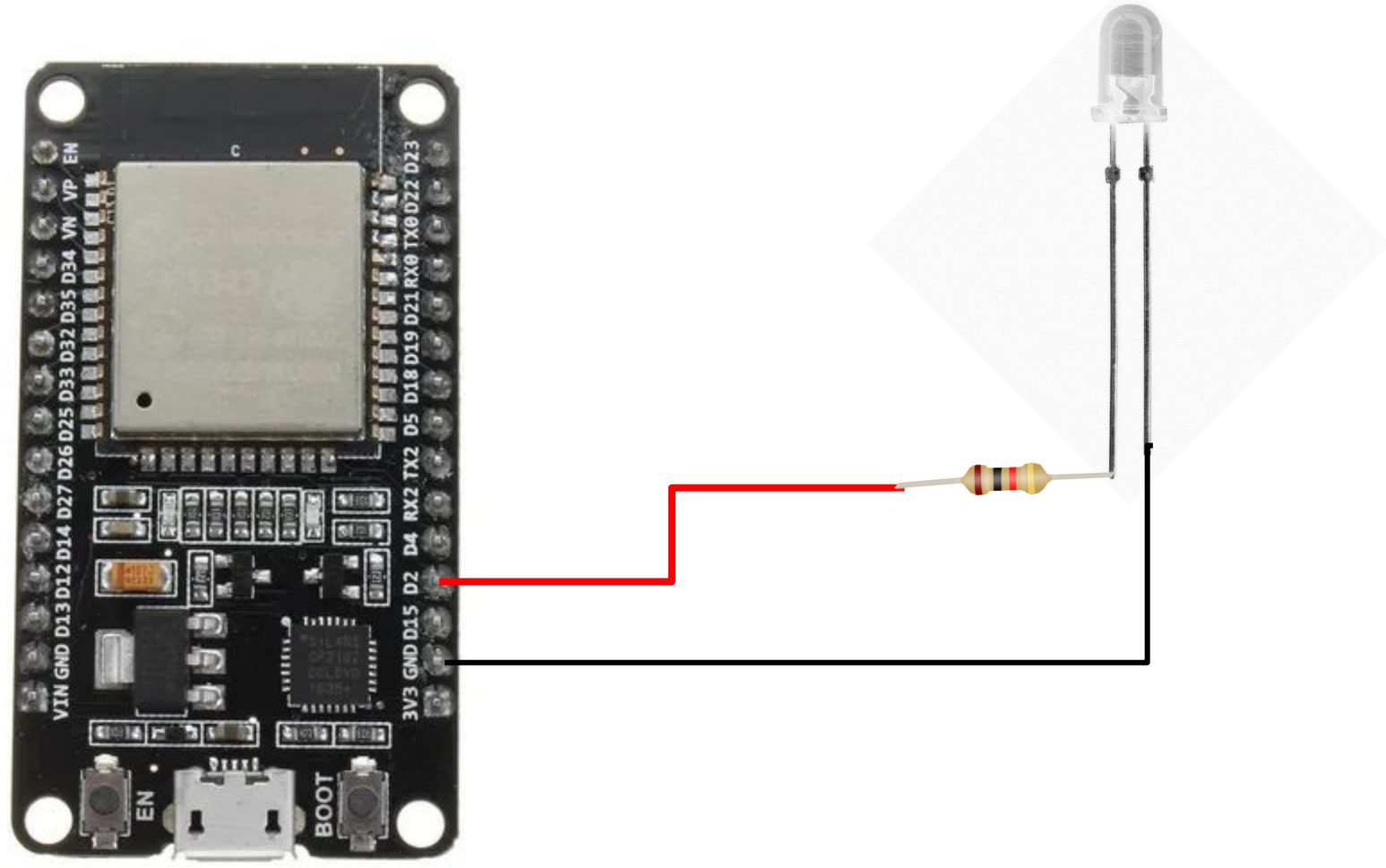
• ESP32 Dev Module

ESP32 Wrover Module

USINAINFO

ELETRÔNICA & ROBÓTICA

Controlando um led

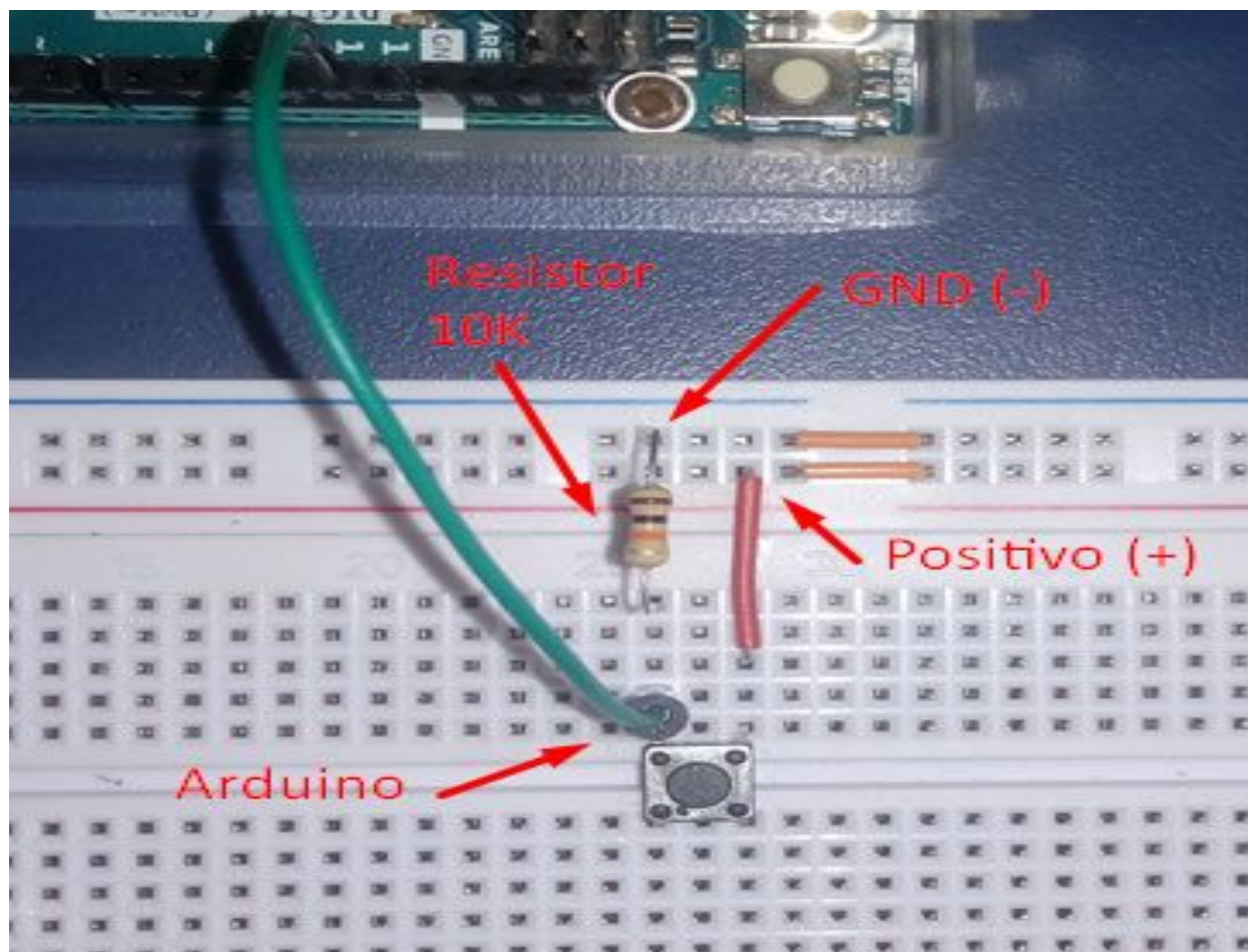


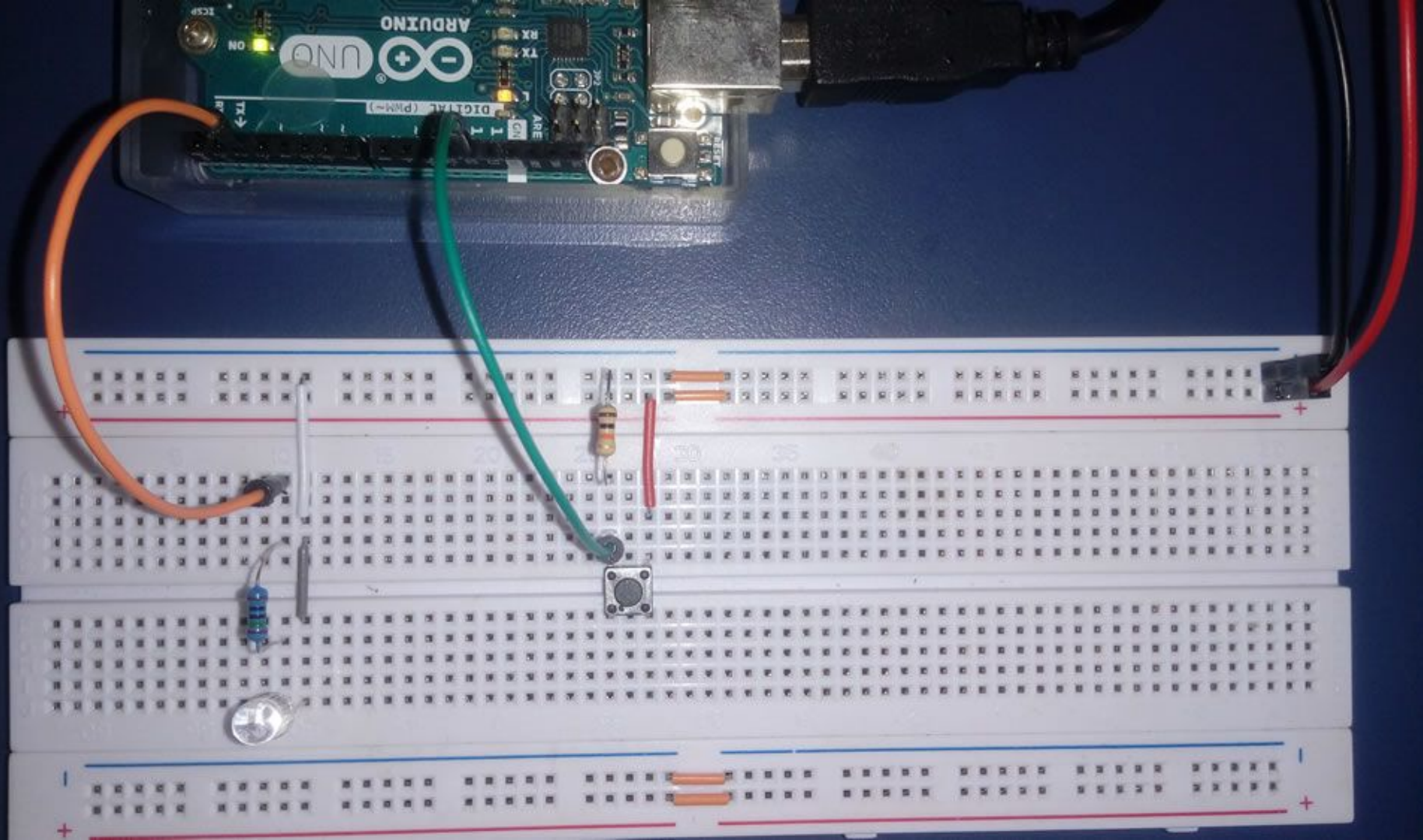
```
const int led1 = 2;
```

```
void setup() {  
    pinMode(led1, OUTPUT);  
}
```

```
void loop() {  
    digitalWrite(led1, 1);  
    delay(1000);  
    digitalWrite(led1, 0);  
    delay(1000);  
}
```


Controlando um led e um botão





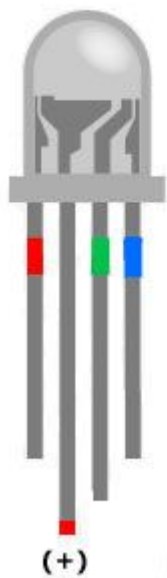
```
const int led = 2;
const int botao = 4;
int estadoBotao;

void setup() {
    pinMode(led, OUTPUT);
    pinMode(botao, INPUT);
}

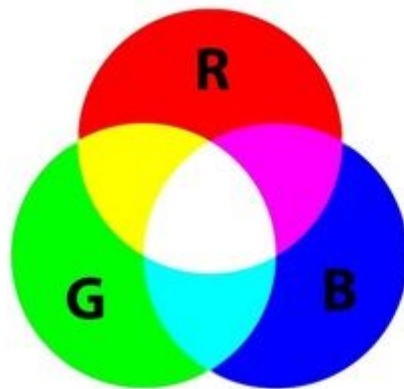
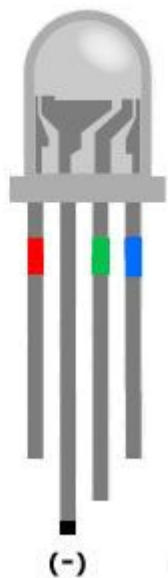
void loop() {
    estadoBotao = digitalRead(botao);
    if (estadoBotao == 1) {
        digitalWrite(led, 1);
    }
    if (estadoBotao == 0) {
        digitalWrite(led, 0);
    }
}
```

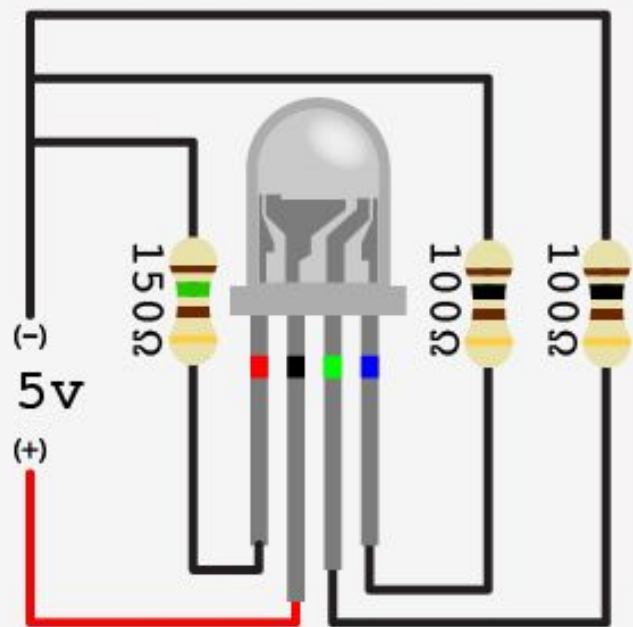
Controlando um led RGB

Ânodo comum

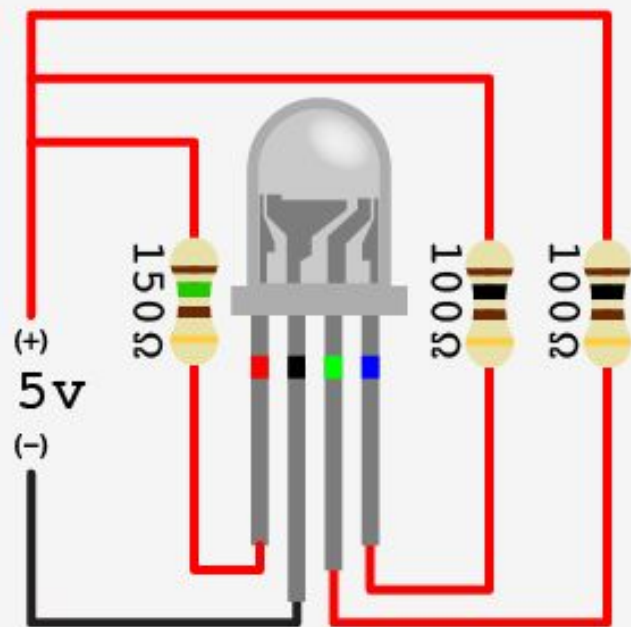


Cátodo comum

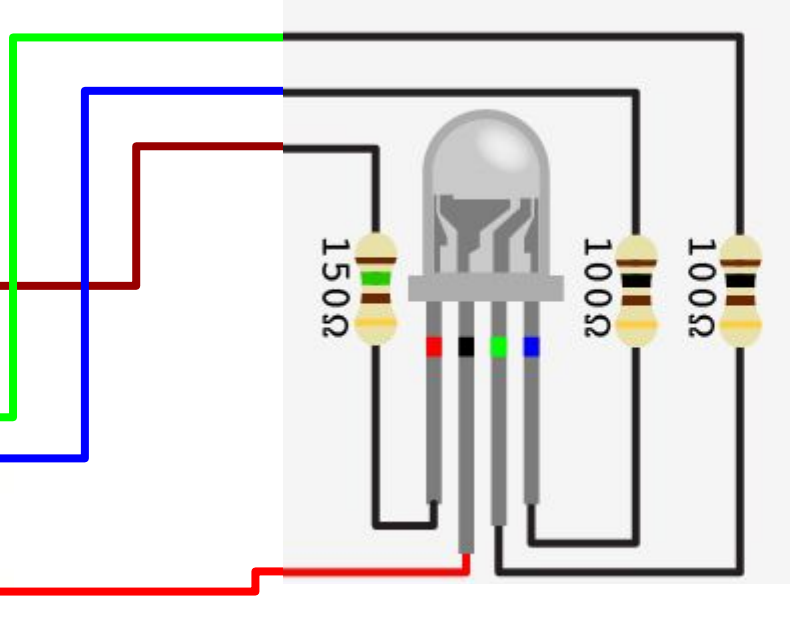
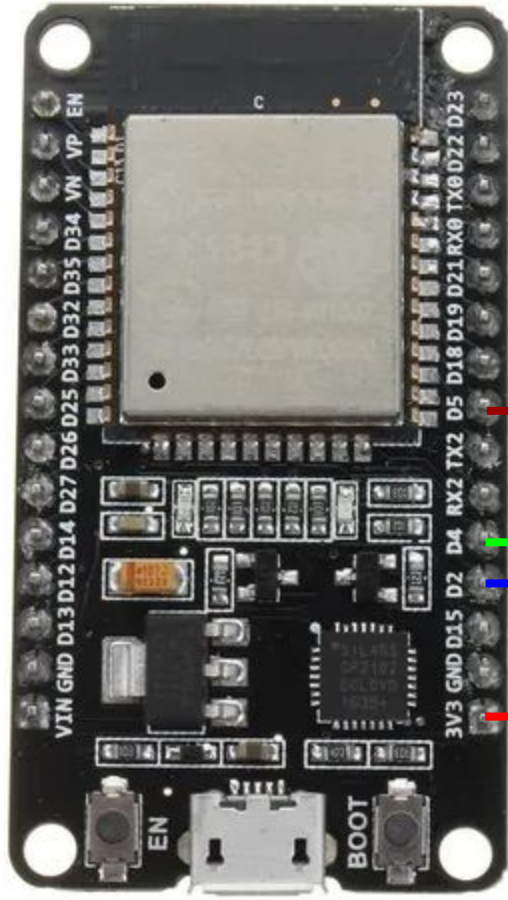


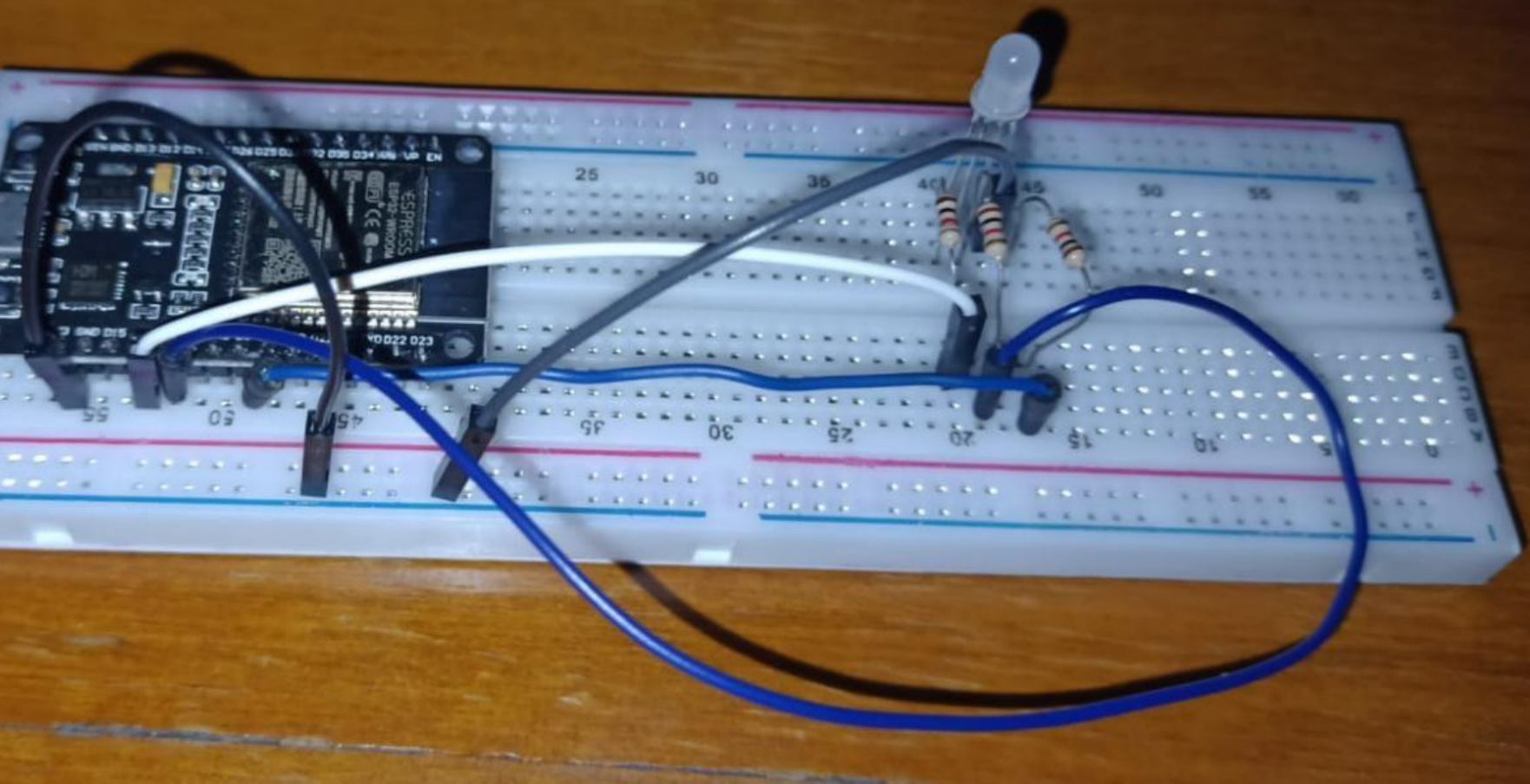


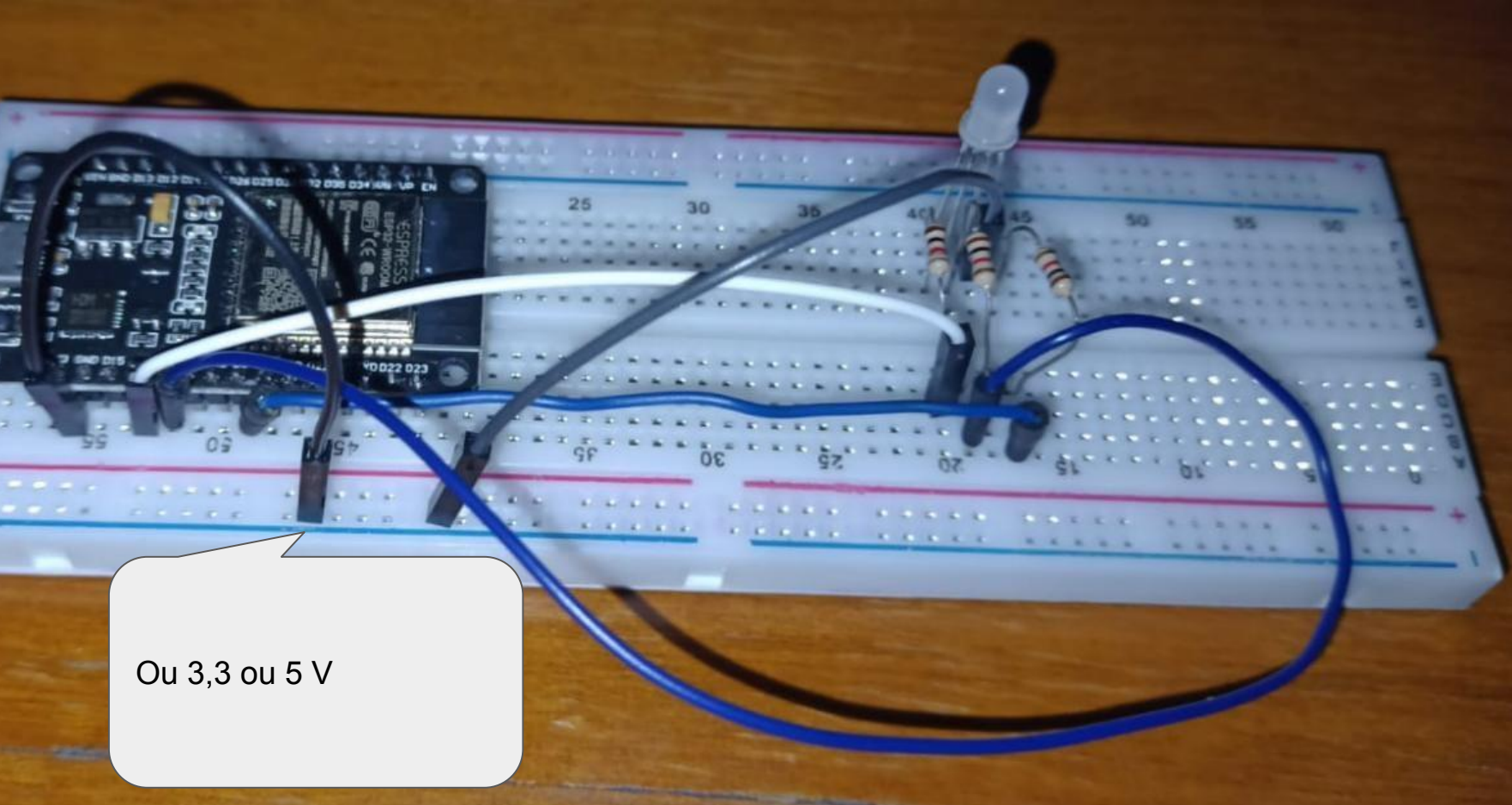
Common Anode



Common Cathode







Ou 3,3 ou 5 V


```
const int azul = 2;
const int verde = 4;
const int vermelho = 5;

void setup() {
  pinMode(verde, OUTPUT);
  pinMode(azul, OUTPUT);
  pinMode(vermelho, OUTPUT);
}
```

```
void loop() {
  digitalWrite(vermelho, 0);
  digitalWrite(azul, 1);
  digitalWrite(verde, 1);
  delay(2000);

  digitalWrite(vermelho, 1);
  digitalWrite(azul, 0);
  digitalWrite(verde, 1);
  delay(2000);

  digitalWrite(vermelho, 1);
  digitalWrite(azul, 1);
  digitalWrite(verde, 0);
  delay(2000);
}
```