

**FIAP GRADUAÇÃO**

# TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

Arquiteturas Disruptivas, IoT, Big Data e IA

Prof. Rafael Matsuyama

# I Roteiro

- ✓ Sensores Analógicos e Digitais
  - ✓ LDR
  - ✓ DHT-11
  - ✓ LEDs
  - ✓ Saídas Digitais
  - ✓ Saídas Analógicas

# Sensores Analógicos e Digitais

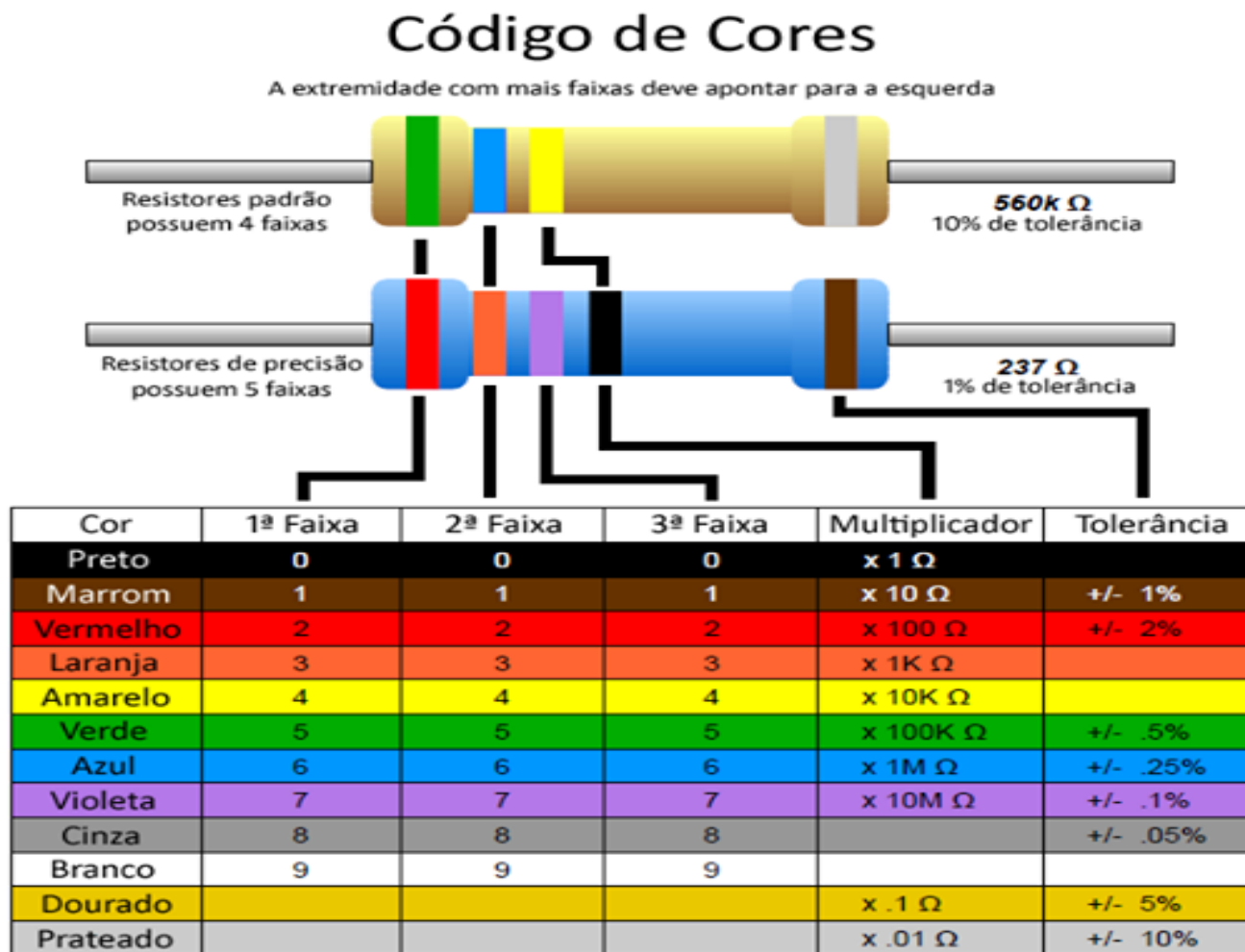
## Lendo o valor dos resistores (Pt. 1)

- Verificar através da tabela de cores o algarismo correspondente à cor do primeiro anel, que será o primeiro dígito do valor da resistência.
- Verificar através da tabela de cores o algarismo correspondente à cor do segundo anel, que será o segundo dígito do valor da resistência.
- Determinar o valor para multiplicar o número formado pelos itens 1 e 2 pela cor do anel multiplicador (3ª ou 4ª faixa).

## Lendo o valor dos resistores (Pt. 2)

- Verificar a porcentagem de tolerância do valor nominal da resistência pela cor do último anel.
- Ex: cores marrom (1), preto (0), laranja (x1k) e dourado: 10k ohms de resistência com tolerância de 5%

## Lendo o valor dos resistores (Pt. 3)



Fonte: <http://eletronsdadepressao.blogspot.com.br/2015/01/codigo-de-cores-de-resistores.html>

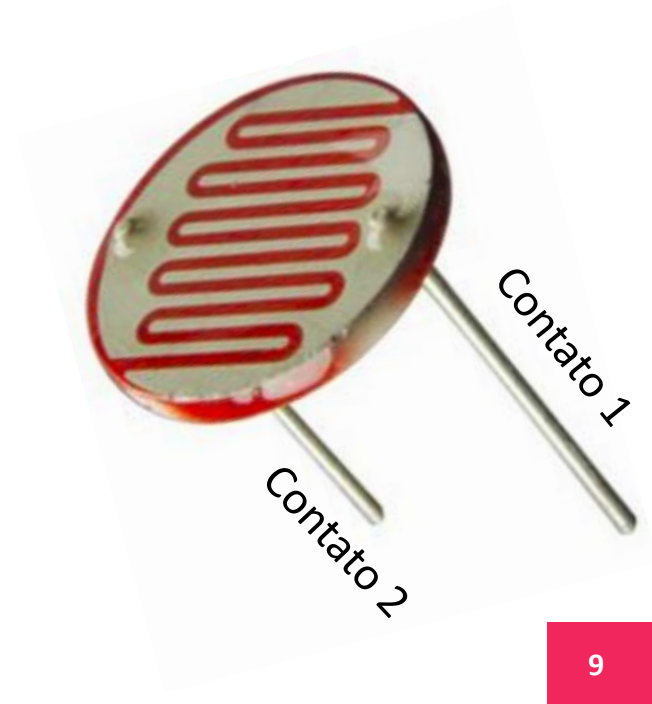
## ■ Sensor de Luminosidade (LDR) (Pt. 1)

- Nosso primeiro sensor analógico será o LDR (Light Dependent Resistor), um sensor de luminosidade ambiente.
- Ele possui dois contatos elétricos planos separados por uma trilha de sulfeto de cádmio.
  - Quando há incidência de luz, essa substância permite a condução de corrente, porém apresentando uma certa resistência elétrica.
  - Quanto maior essa incidência de luz, menor será essa resistência, e portanto, caso haja uma tensão aplicada, maior será a corrente elétrica ali passando.



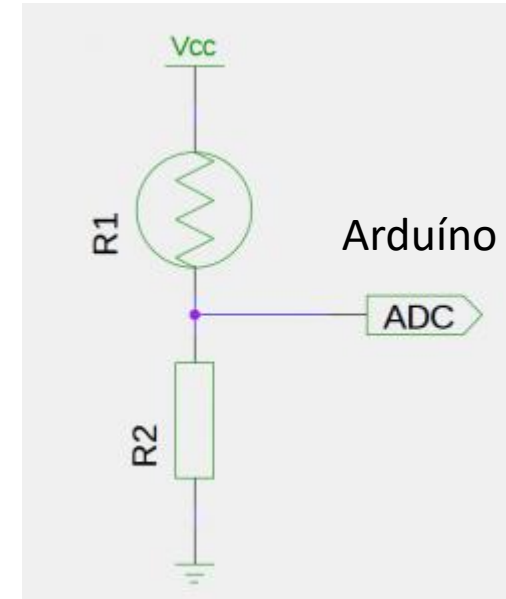
## Sensor de Luminosidade (LDR) (Pt. 2)

- O LDR funciona então como um resistor cuja resistência varia de acordo com a incidência de luz.
  - Para medir a luz, precisamos medir essa resistência.
  - Para tanto, aplicamos uma diferença de potencial e medimos a corrente que passa.

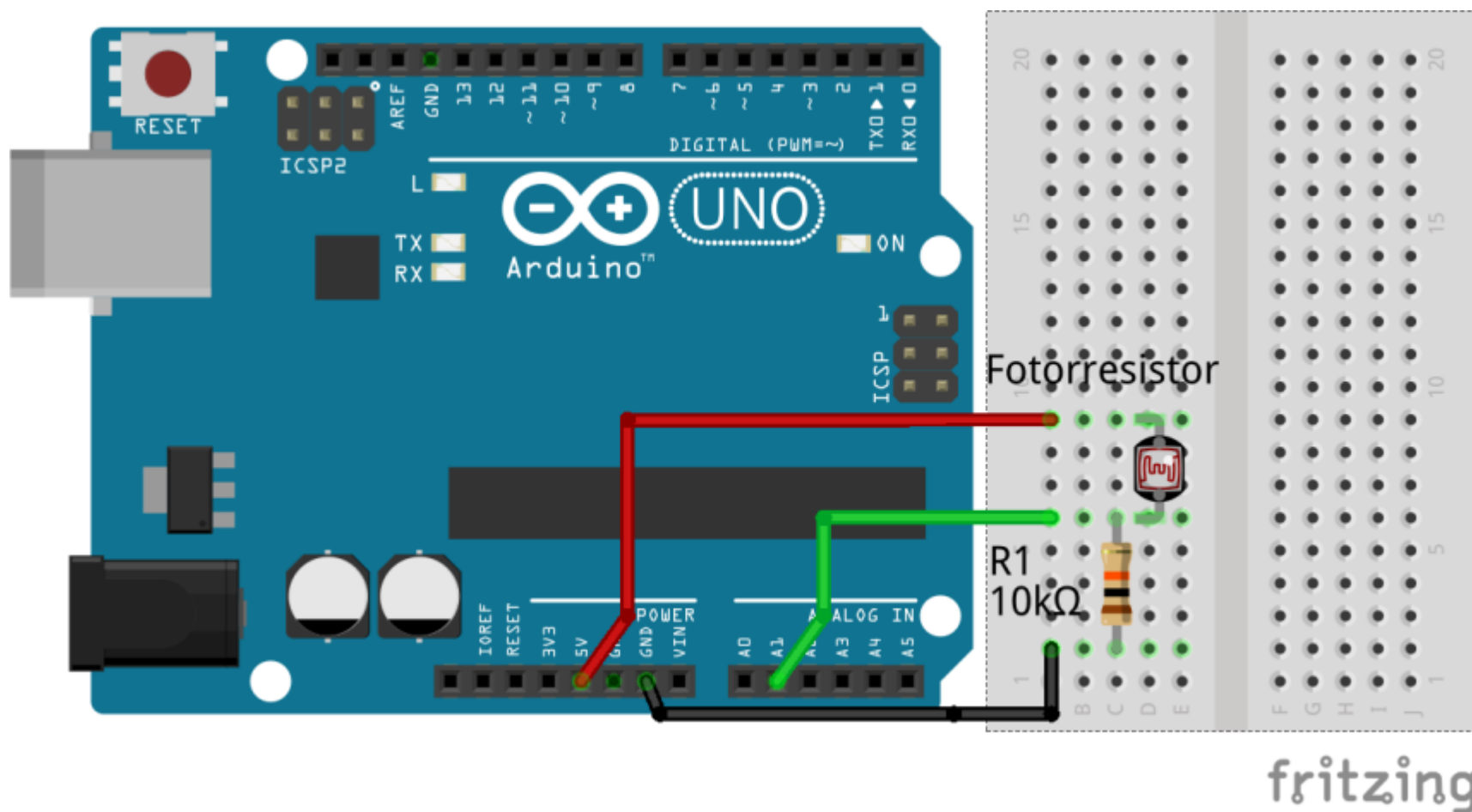


## Medindo a luminosidade com o LDR

- Usamos um resistor (R2) ligado em série com o LDR (R1), aplicando uma tensão constante nas extremidades livres.
- A resistência equivalente dos dois componentes é a soma das resistências de ambos, e a corrente que passa nos dois componentes é a mesma.
- Como a resistência do resistor é constante, a tensão elétrica entre as suas extremidades será proporcional à corrente, que por sua vez é proporcional à luminosidade ambiente.



## Esquema Elétrico do Exercício 02



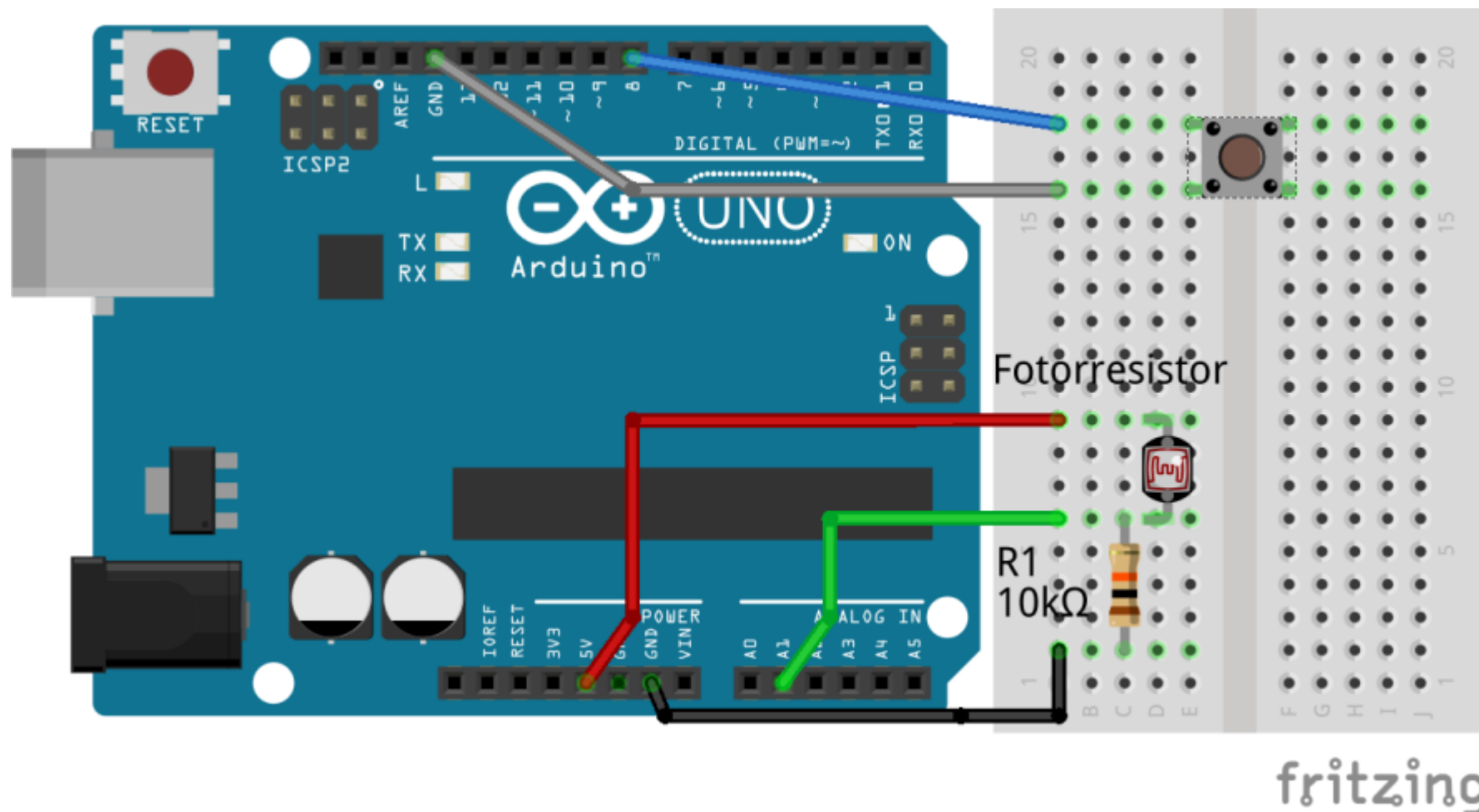
## Código do Exercício 02

```
int sensor = A1; // Pino analógico em que o sensor  
está conectado
```

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

```
void loop(){  
    // Lendo o valor do sensor.  
    int valorSensor = analogRead(sensor);  
    // Exibindo o valor do sensor no serial monitor.  
    Serial.println(valorSensor);  
    delay(500);  
}
```

## Esquema Elétrico do Exercício 03



## Código do Exercício 03

- Ligue um terminal do botão na porta 8, e a outra no GND (ver figura logo mais).
- Execute o programa a seguir e acompanhe pelo monitor serial.
- Experimente apertar o botão e ver a saída.

```
int inPin = 8; //entrada digital na porta 8
int val = 0;
void setup() {
    Serial.begin(9600);
    pinMode(inPin, INPUT_PULLUP); // porta 8 vira entrada
}
void loop() {
    val = digitalRead(inPin);    // read the input pin
    Serial.println(val);
    delay(2000);
}
```

## Saídas Digitais (Pt. 1)

- As saídas D0 a D13 permitem a leitura ou escrita de valores lógicos. Por isso são chamadas de GPIO (General-Purpose Input and Output).
- As portas D0 (RX0) e D1 (TX0) são destinadas para a comunicação serial – via cabo USB ou comunicação com shields, por exemplo.
- São configuradas como saídas digitais na função `setup()` através da chamada:
  - `pinMode(numero, OUTPUT);`

## Saídas Digitais (Pt. 2)

- Podem acionar dispositivos que necessitem de uma informação digital (0 ou 1) para seu acionamento, por exemplo:
  - Luz LED.
  - Relê para acionamento de equipamentos elétricos.
  - Dispositivo de acionamento de motores (drives).
- Na saída porta D13 há um LED conectado, que acende assim que a porta esta acionada.
- Vamos executar o exemplo: `File >> Examples >> 01.Basic >> Blink`
  - `digitalWrite(HIGH | LOW) ;` aciona ou desliga a saída digital

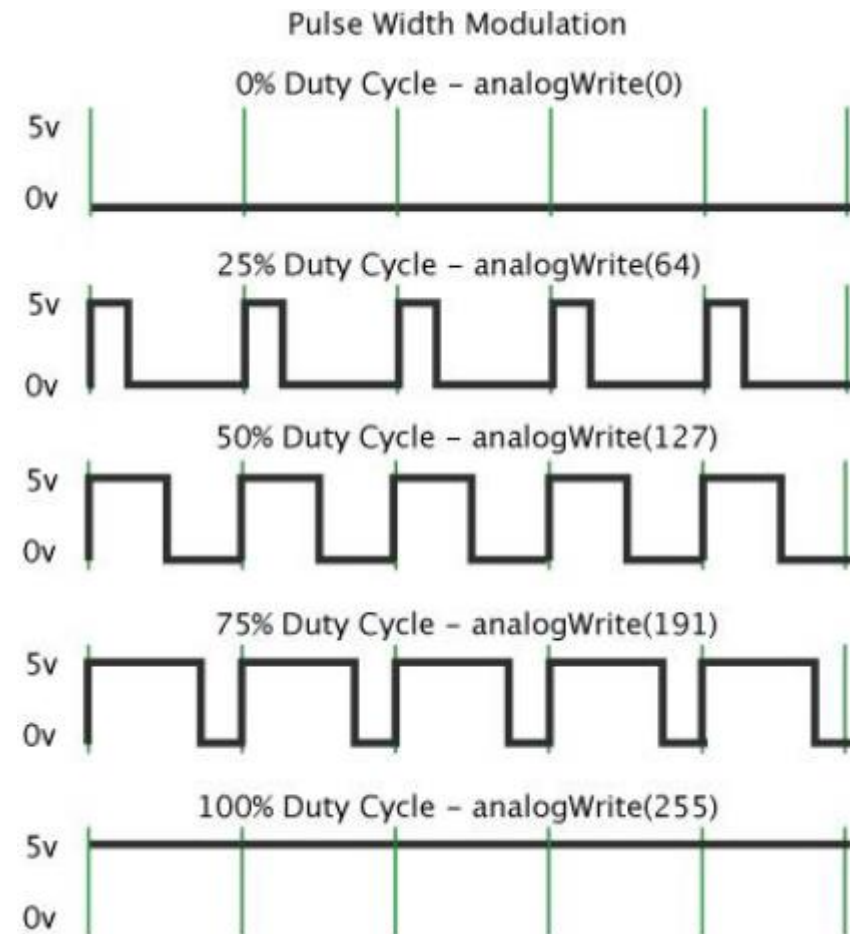


## Código do Exercício 04 – Piscar LED na Porta D13

```
// Pin 13 has an LED connected on most Arduino boards.
int led = 13;
// the setup routine runs once when you press reset:
void setup() {
    // initialize the digital pin as an output.
    pinMode(led, OUTPUT);
}
// the loop routine runs over and over again forever:
void loop() {
    // turn the LED on (HIGH is the voltage level)
    digitalWrite(led, HIGH);
    delay(1000);                // wait for a second
    // turn the LED off by making the voltage LOW
    digitalWrite(led, LOW);
    delay(1000);                // wait for a second
}
```

## Saídas Analógicas - PWM (Pt. 1)

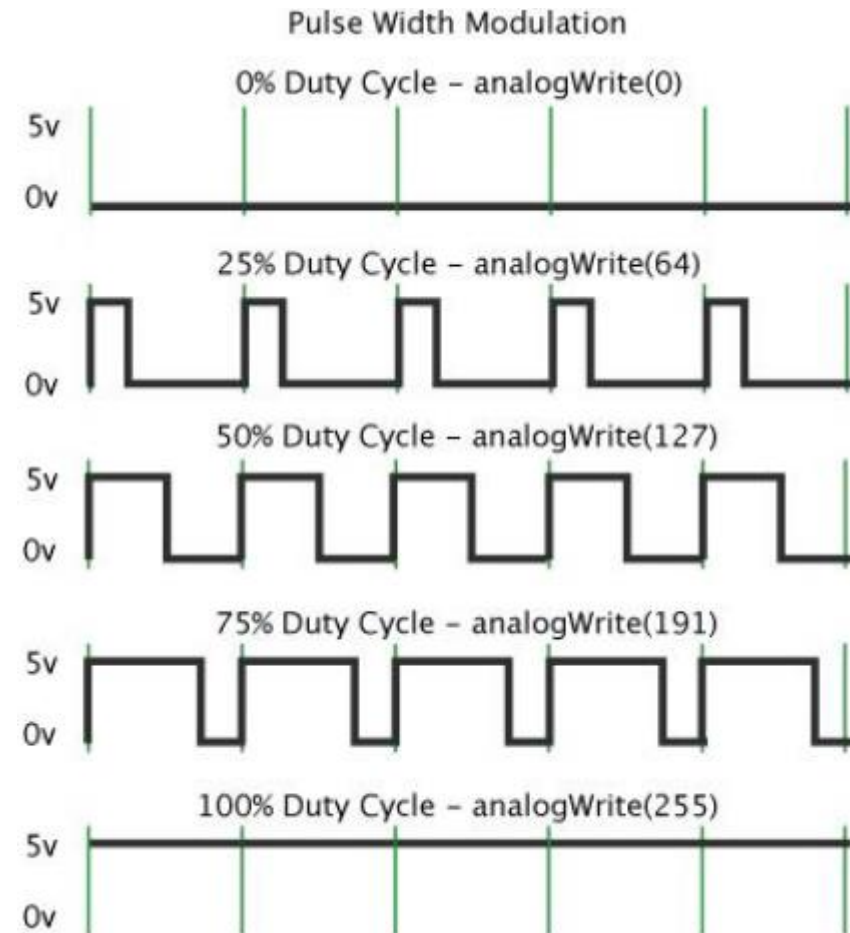
- As portas digitais (GPIO) marcadas com um ~ (til) possuem a capacidade de servirem de saída do tipo PWM – Pulse-Width Modulation.
- Uma saída PWM automaticamente alterna períodos em que ela está ligada (HIGH) e em que está desligada (LOW). A proporção do tempo em que a saída está ligada em relação ao tempo total do ciclo é chamad de Duty Cycle, e varia de 0 a 100%.



## Saídas Analógicas - PWM (Pt. 2)

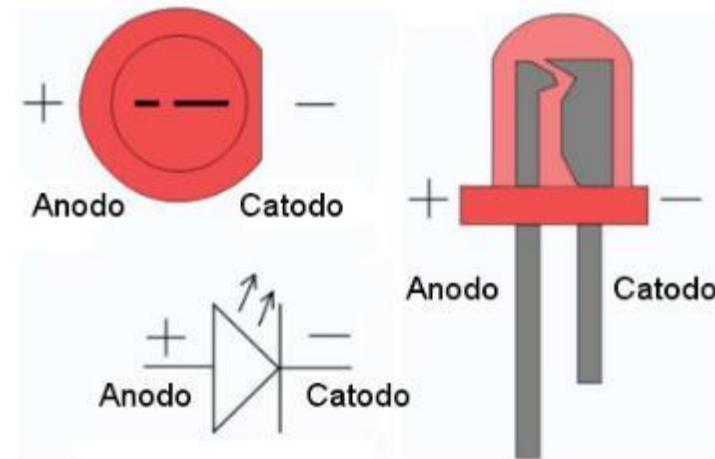
- Exemplos: motor de passo, iluminação LED ou acionador dimerizável de lâmpada, LED RGB (seleciona a cor do LED).
- Para escrever um valor de 0 a 255 na saída PWM, esta deve estar configurada como uma saída digital, sendo que a escrita se dá da forma:

```
analogWrite(numero, valor);
```

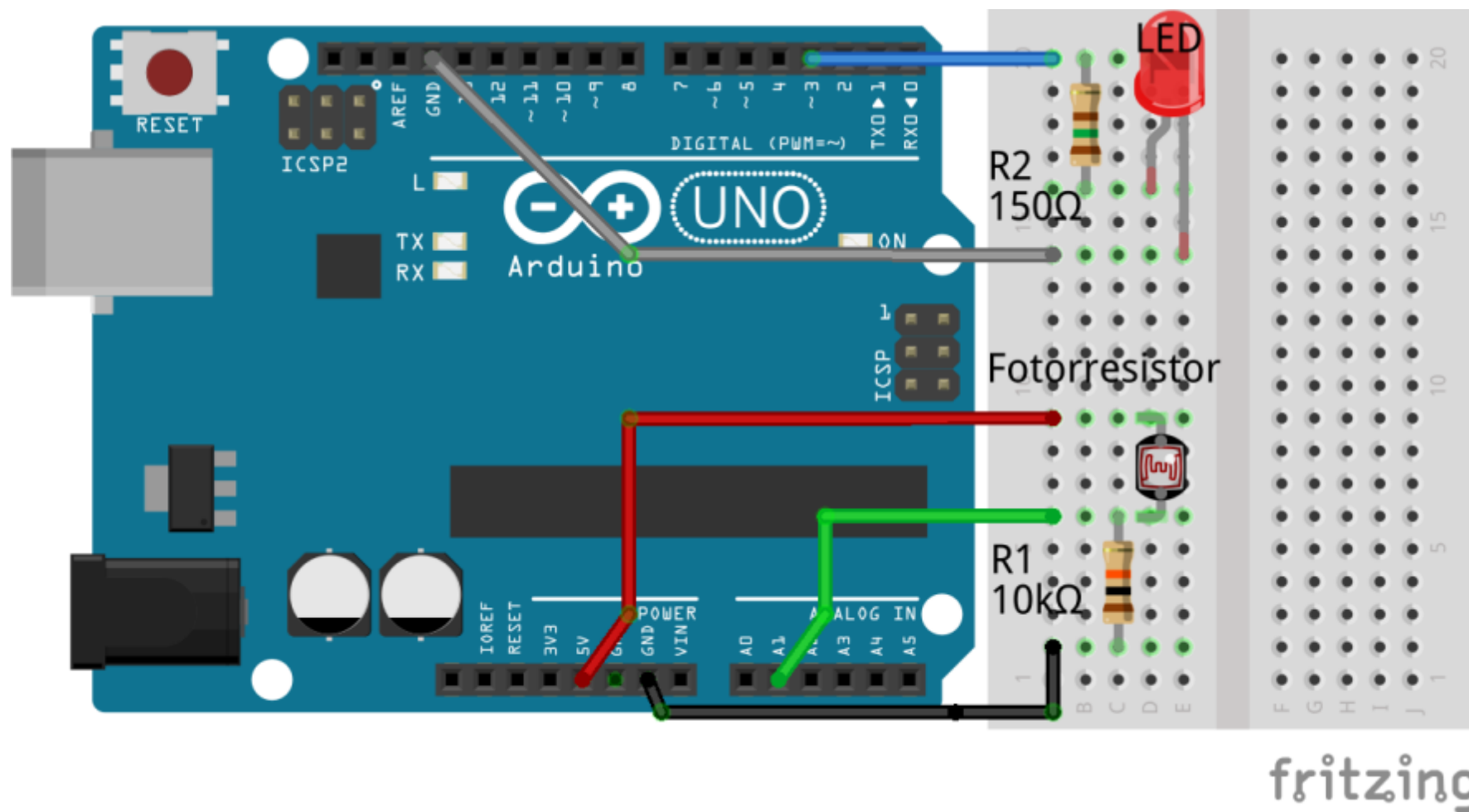


## LEDs – Instalação

- Deve-se sempre interpor um resistor de 100 a 150 ohms em série ao LED, para evitar uma sobretensão no mesmo.
  - O resistor pode estar interposto tanto ao catodo quanto ao anodo, indistintamente.
- O LED possui polaridade. Seus terminais são:
  - Anodo: correspondente ao polo positivo, deve ser ligado à porta controladora.
  - Catodo: correspondente ao polo negativo, deve ser ligado ao GND (0V).



## Esquema Elétrico do Exercício 05



## Código do Exercício 05 – LED proporcional ao sensor (Pt. 1)

```
int sensor = 1; //Pino analógico em que o sensor está
conectado
int led = 3; // Pino em que o led está conectado

void setup(){
    Serial.begin(9600);
    pinMode(led,OUTPUT);
}
```

## Código do Exercício 05 – LED proporcional ao sensor (Pt. 2)

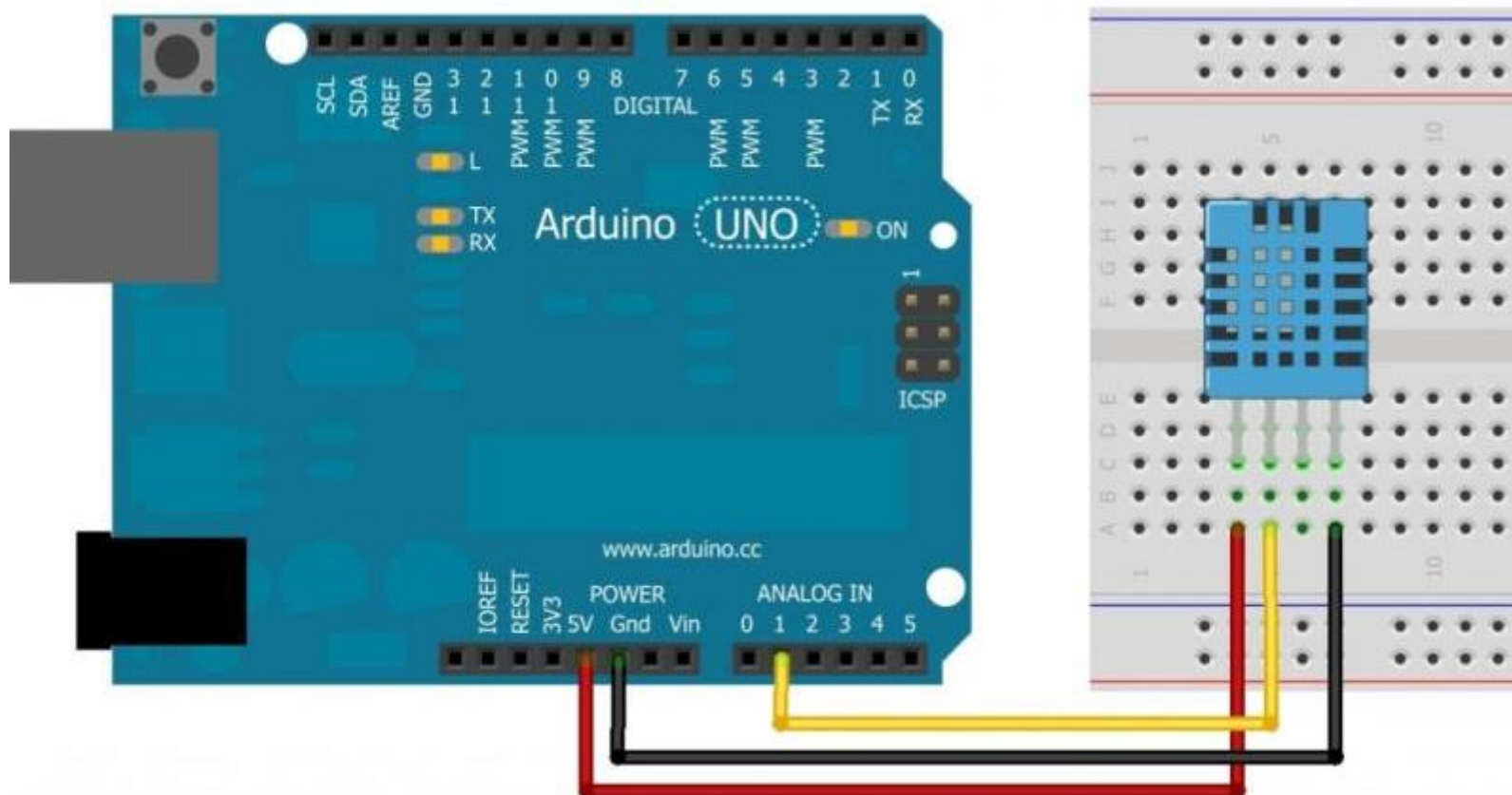
```
void loop() {  
    //Lendo o valor do sensor.  
    int valorSensor = analogRead(sensor);  
    //Exibindo o valor do sensor no serial monitor.  
    Serial.println(valorSensor);  
    //Acionando o LED: quanto menos luz externa, mais  
    forte o LED  
  
    //map(valor, deEscalaAnt, ateEscalaAnt, deNovaEscala, ateNovaEscala)  
    analogWrite(led, map(valorSensor, 0, 1023, 255, 0));  
    delay(50);  
}
```

## Sensor DHT-11

- O modo mais simples de se trabalhar com bibliotecas no Arduino é através de arquivos compactados
- Elas possuem definição de classes e podem conter exemplos de utilização
- Instalar o arquivo DHT.zip através de  
`Sketch >> Import Libraries >> Add Library...`
- O sensor DHT11 fornece tanto temperatura quanto umidade do ar instantaneamente e de forma muito fácil.
- Isso se deve à biblioteca que cuida de todas as funções necessárias, restando para nós apenas acessar aos dados.



## Esquema Elétrico do Exercício 06



## Código do Exercício 06 – Ler o sensor DHT-11 (Pt. 1)

```
#include "DHT.h"
#define DHTPIN A1 // pino que estamos conectado
#define DHTTYPE DHT11 // DHT 11
DHT dht(DHTPIN, DHTTYPE); //Instanciação do objeto do
sensor
void setup() {
    Serial.begin(9600);
    dht.begin();
}
```

## Código do Exercício 06 – Ler o sensor DHT-11 (Pt. 2)

```
void loop() {  
    // A leitura da temperatura e umidade pode levar  
    250ms!  
    float h = dht.readHumidity();//Valor da umidade  
    float t = dht.readTemperature(); //Valor da  
    temperatura  
    if (isnan(t) || isnan(h)) {  
        Serial.println("Erro ao ler do DHT");  
    } else {  
        Serial.print("Umidade: ");  
        Serial.print(h); Serial.print(" %\t");  
        Serial.print("Temperatura: ");  
        Serial.print(t); Serial.println(" °C");  
    }  
}
```

## REFERÊNCIAS



1. [http://www.telecom.uff.br/pet/petws/downloads/tutoriais/arduino/Tut\\_Arduino.pdf](http://www.telecom.uff.br/pet/petws/downloads/tutoriais/arduino/Tut_Arduino.pdf)
2. <http://arduino.cc/en/Reference/HomePage>
3. <https://www.arduino.cc/en/Reference/StringObject>
4. <http://json.org>

Copyright © 2018-2019 Prof. Rafael Matsuyama / Prof. Antônio Selvatici

Todos direitos reservados. Reprodução ou divulgação total ou parcial deste documento é expressamente proibido sem o consentimento formal, por escrito, do Professor (autor).