

## EXPERIÊNCIA # 01 – PISCA LED

**Introdução:** O LED (*Light Emitting Diode*) é um diodo (componente eletrônico que deixa passar a corrente em apenas um sentido) que emite luz quando energizado (Vide Figura 1.1). Os LEDs apresentam muitas vantagens sobre as fontes de luz incandescentes como um consumo menor de energia, maior tempo de vida, menor tamanho, grande durabilidade e confiabilidade. O LED tem uma polaridade correta de ligação ao circuito eletrônico, ou seja se conectá-lo invertido não funcionará adequadamente. Revise os desenhos para verificar a correspondência do negativo e do positivo. Os LEDs são especialmente utilizados em produtos de microeletrônica como sinalizadores de avisos. Também são muito usados em painéis, cortinas e pistas de LED. Podem ser encontrados em tamanho maior, como em alguns modelos de semáforos ou displays.



**Figura 1.1** – Símbolo e apresentação física de um LED (Fonte: [http://rsandas.com/P2\\_Session\\_4-5.html](http://rsandas.com/P2_Session_4-5.html)).

Algumas informações para iniciar o experimento:

1. A saída da placa arduino é de 5V;
2. Cada cor de LED tem uma tensão de saída. Neste caso utilizaremos o LED vermelho de 1,6V;
3. Corrente aproximada deste LED aproximadamente 20mA (ou 0,020 Ampères).

Sendo a saída do arduino 5V e a tensão do LED 1,6V faz-se necessária a introdução de um resistor em série para que a tensão sobre o LED não seja maior do que suportável por ele. Assim, aplica-se a **Lei de Ohm**:

$$V = R \times I$$

Tensão Elétrica = Resistência Elétrica x Corrente Elétrica

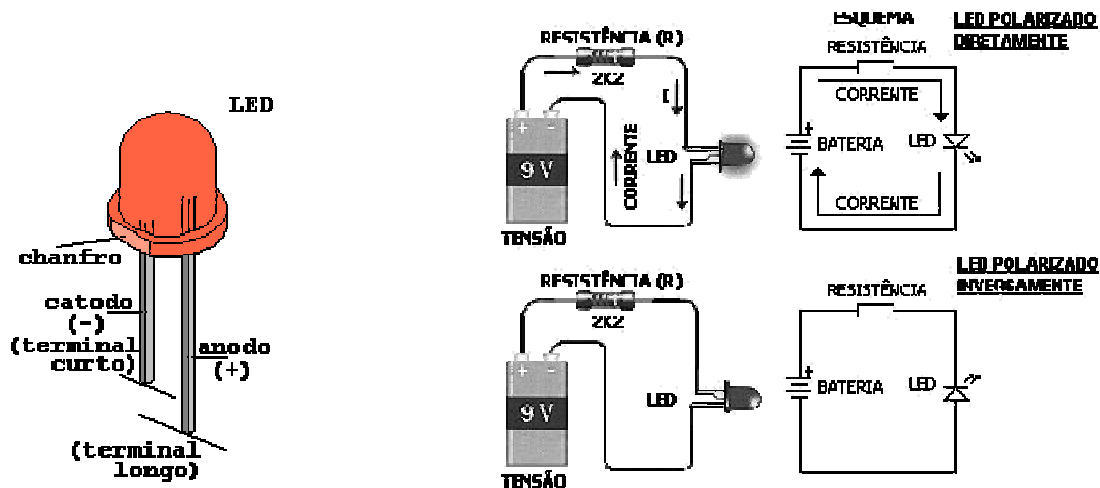
No nosso caso,  $i_{LED} \leq 20mA = 0,020A$ . Assim, aplicando a Lei de Ohm a um Resistor em série com o LED, teremos (vide Figura 1.2):

$$V_{CC(ARDUINO)} = R_{SÉRIE} \times i_{LED} + V_{LED}$$

$$+5V = R_{SÉRIE} \times 0,02A + 1,6V$$

$$R_{SÉRIE} = (5V - 1,6V) / 0,02A = 3,4V / 0,02A = 170\Omega$$

Então, devemos utilizar um resistor de pelo menos 170  $\Omega$  (Ohms) para limitar a corrente no LED a no máximo 20mA (ou 0,02 Ampères). Repare que a corrente no LED diminuirá à medida que o resistor em série for maior que 170  $\Omega$ . Por isso, para não prejudicar a emissão de luz, não se deve usar um resistor maior que 1k $\Omega$ .



**Figura 1.2** - Símbolo, diagrama e esquema de polarização de um LED. Repare que apenas a polarização direta permite o funcionamento correto do LED.

### Objetivos:

- Verificar o funcionamento de um diodo emissor de luz (LED);
- Utilizar portas digitais e temporização no Arduino.

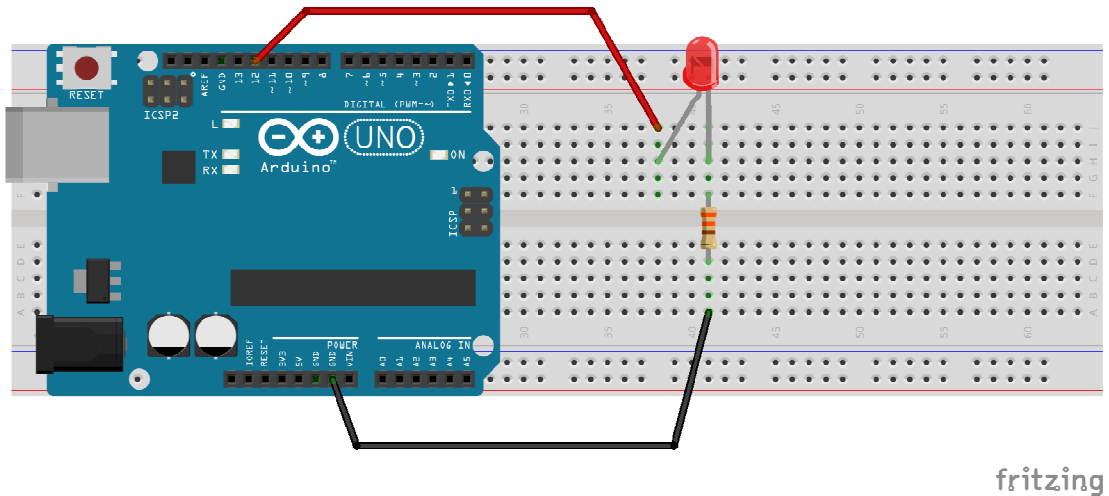
### Material Necessário:

- 01 LED 5mm;
- 01 resistor de 1kΩ;
- 01 resistor de 220Ω ou 330Ω;
- 01 Placa Arduino UNO REV 03;
- fios (jumpers) diversos.

### Procedimentos Experimentais:

- 1) Faça o acendimento de um LED vermelho, conforme o esquema mostrado na polarização direta da Figura 1.2. Para isso:
  - a) conecte a placa Arduino na entrada USB do computador;
  - b) conecte o pólo positivo do LED na saída +5V (Vcc) do Arduino;
  - c) conecte o pólo negativo do LED a um terminal do resistor de 220Ω (ou 330Ω);
  - d) conecte o outro terminal do resistor ao GND (Terra) do Arduino e observe o acendimento do LED;
  - e) Repita os itens a) – d), mas usando um resistor de 1kΩ. Verifique em qual montagem o LED vai acender mais intensamente. Você sabe por que isso acontece?

2) Monte o circuito mostrado na Figura 1.3. Lembre-se de verificar a correta polarização do diodo.



**Figura 1.3** – Circuito utilizado na experiência de “Pisca LED” (Cortesia: <http://fritzing.org>).

3) Na interface de programação de *Sketches* do Arduino, insira e compile o programa a seguir com a placa conectada ao seu computador através da porta USB.

Procure interpretar cada linha de comando do programa, tentando entender como o ARDUINO responde a esses comandos.

```
// EXP 01: PISCA_LED COM TEMPORIZAÇÃO

void setup()
{
    pinMode(12,OUTPUT); //define pino 12 como saída digital
}

void loop()
{
    //deixa pino 12 em nível alto (+5V) durante 1000ms (=1s)
    digitalWrite(12,1);
    delay(1000);

    //deixa pino 12 em nível baixo (0V) durante 1000ms (=1s)
    digitalWrite(12,0);
    delay(1000);
}
```

Programa 1.1 – Código em linguagem C utilizado na experiência de “Pisca LED”. As linhas iniciadas por “//” não são lidas pelo compilador, e por isso são utilizadas como comentários.

4) Altere o tempo de funcionamento do LED criando uma variável “t” para a temporização.

```
// EXP 02: PISCA_LED TEMPORIZADO COM VARIÁVEL t

int t = 1000; //cria variável t = 1000ms = 1s

void setup()
{
    pinMode(12,OUTPUT); //define pino 12 como saída digital
}

void loop()
{
    //deixa pino 12 em nível alto (+5V) durante 1000ms (=1s)
    digitalWrite(12,1);
    delay(t);

    //deixa pino 12 em nível baixo (0V) durante 1000ms (=1s)
    digitalWrite(12,0);
    delay(t);
}
```

Programa 1.2 – Código em linguagem C utilizado na experiência de “Pisca LED”. As linhas iniciadas por “//” não são lidas pelo compilador, e por isso são utilizadas como comentários.

5) Altere o tempo de *delay* para 2 segundos com LED ligado e 0,5 segundo para LED desligado.

6) Inclua um novo LED que funcione na saída número 13 da placa do Arduino (sem retirar o anterior). Faça um novo programa em que os 2 LEDs funcionem alternadamente, de modo que quando o primeiro LED se apagar, o novo LED imediatamente se acenda e vice-versa. Cada um permanecerá ligado por 1 segundo.

7) Faça um novo programa alterando o tempo de acendimento de cada LED, de tal modo que o primeiro esteja ligado durante 0,3 segundo; o segundo LED durante 1 segundo; e que haja um tempo de 0,5 segundo de *delay* antes de acender o LED seguinte.

**Para o Diário de Bordo:**

a) Escreva suas conclusões sobre o LED.

b) Explique para que serve a criação da variável “t” no programa 1.2.