

# Implementação de semáforo para carros e pedestres com LEDs e um botão

Elias de Almeida Sombra Neto<sup>1</sup>

<sup>1</sup>Instituto Federal de Educação, Ciência e Tecnologia do Ceará (IFCE) - Campus Maracanaú  
Av. Parque Central, 1315 - Distrito Industrial I, Maracanaú-CE, Brasil

elias.almeida09@aluno.ifce.edu.br

**Abstract.** *This paper describes the implementation of a traffic light system for cars and pedestrians using an ESP32 microcontroller. The system allows the pedestrian green signal to be activated via a button, while the car red signal is turned on. After a transition cycle, the traffic lights return to their initial configuration. The assembly was performed on a protoboard using LEDs and resistors to simulate the light signals. The code was developed in C++ using Visual Studio Code with the PlatformIO extension. Tests demonstrated the correct functioning of the system, validating the efficiency of the circuit and the applied programming.*

**Resumo.** *Este artigo descreve a implementação de um sistema de semáforo para carros e pedestres utilizando um microcontrolador ESP32. O sistema permite que o sinal verde para pedestres seja ativado por meio de um botão, enquanto o sinal vermelho para carros é acionado. Após um ciclo de transição, os semáforos retornam à configuração inicial. A montagem foi realizada em uma protoboard, utilizando LEDs e resistores para simular os sinais luminosos. O código foi desenvolvido em C++ utilizando o Visual Studio Code com a extensão PlatformIO. Os testes demonstraram o correto funcionamento do sistema, validando a eficácia do circuito e da programação aplicada.*

## 1. Introdução

Este artigo descreve uma atividade prática realizada no Laboratório de Eletroeletrônica e Sistema Embarcados (LAESE) durante a disciplina de Microcontroladores no Instituto Federal do Ceará (IFCE).

O objetivo da atividade é acionar o sinal verde para pedestres através de um botão, enquanto o sinal vermelho para carros é ativado. Inicialmente, o sinal verde para carros e o sinal vermelho para pedestres permanecem ligados. Ao pressionar o botão, o verde para carros é desligado e o vermelho é ligado. Em paralelo a isso, o vermelho para pedestres é desligado e o verde é ligado. Após 4 segundos, o sinal vermelho de carros desliga e o amarelo começa a piscar 3 vezes juntamente ao verde de pedestres para indicar a troca nos semáforos. Depois disso, o amarelo de carros se apaga com o verde de pedestres, e então o verde de carros acende em conjunto com o vermelho de pedestres.

## 2. Materiais utilizados

Os materiais para a construção do circuito incluem:

- 1 ESP32 30 pinos;
- 1 cabo micro USB;
- 1 protoboard;
- 2 LEDs vermelhos;
- 1 LED amarelo;
- 2 LEDs verdes;
- 1 botão;
- 6 resistores;
- 7 cabos jumper macho-fêmea;
- 1 cabo jumper macho-macho.

### 3. Montagem do circuito

Para realizar a montagem do circuito, é preciso seguir os seguintes passos:

- Posicionar o botão e os 5 LEDs na região central da protoboard.
- Conectar uma extremidade do jumper macho-fêmea em um GPIO do ESP32 e o outro na mesma coluna do pino positivo de cada LED.
- Conectar uma extremidade do jumper macho-fêmea em um GPIO do ESP32 e o outro em uma coluna na qual um pino do botão está inserido.
- Conectar pino GND do ESP32 em um furo positivo da região de alimentação na protoboard através de um cabo jumper macho-fêmea.
- Conectar as duas regiões de alimentação (esquerda e direita) da protoboard por meio de um jumper macho-macho, se necessário.
- Conectar os resistores em paralelo com o botão e os LEDs. Uma extremidade deve ficar em um furo positivo da região de alimentação e a outra deve ficar na mesma coluna do botão e dos LEDs na região central da protoboard.

Além disso, é necessário reservar os seguintes pinos do ESP32:

- GPIO 13 para o LED vermelho de carros;
- GPIO 12 para o LED amarelo de carros;
- GPIO 14 para o LED verde de carros;
- GPIO 27 para o LED vermelho de pedestres;
- GPIO 26 para o LED verde de pedestres;
- GPIO 25 para o botão que aciona o circuito;

### 4. Implementação do código

O código responsável por acionar ambos semáforos por meio do botão foi desenvolvido com a linguagem C++ no editor de código Visual Studio Code juntamente com sua extensão PlatformIO.

```

1 #include <Arduino.h>
2
3 #define VERMELHO_CARRO 13
4 #define AMARELO_CARRO 12
5 #define VERDE_CARRO 14
6 #define VERMELHO_PEDESTRE 27
7 #define VERDE_PEDESTRE 26

```

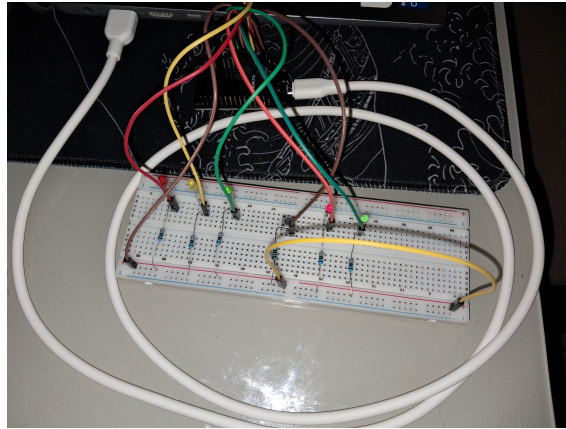
```

8  #define BOTAO 25
9
10 void setup() {
11     pinMode(VERMELHO_CARRO, OUTPUT);
12     pinMode(AMARELO_CARRO, OUTPUT);
13     pinMode(VERDE_CARRO, OUTPUT);
14     pinMode(VERMELHO_PEDESTRE, OUTPUT);
15     pinMode(VERDE_PEDESTRE, OUTPUT);
16     pinMode(BOTAO, INPUT_PULLUP);
17     Serial.begin(9600);
18 }
19
20 void loop() {
21     int leituraBotao = digitalRead(BOTAO);
22     digitalWrite(VERDE_CARRO, HIGH);
23     digitalWrite(VERMELHO_PEDESTRE, HIGH);
24     if (leituraBotao == LOW) {
25         digitalWrite(VERDE_CARRO, LOW);
26         digitalWrite(VERMELHO_CARRO, HIGH);
27         delay(500);
28         digitalWrite(VERMELHO_PEDESTRE, LOW);
29         digitalWrite(VERDE_PEDESTRE, HIGH);
30         delay(4000);
31         digitalWrite(VERMELHO_CARRO, LOW);
32         for (int i = 0; i < 3; i++) {
33             digitalWrite(AMARELO_CARRO, HIGH);
34             delay(1000);
35             digitalWrite(AMARELO_CARRO, LOW);
36             digitalWrite(VERDE_PEDESTRE, LOW);
37             delay(1000);
38             digitalWrite(VERDE_PEDESTRE, HIGH);
39         }
40         digitalWrite(VERDE_CARRO, HIGH);
41         digitalWrite(VERDE_PEDESTRE, LOW);
42         digitalWrite(VERMELHO_PEDESTRE, HIGH);
43     }
44 }

```

## 5. Resultados

A montagem e implementação de código do botão para acionar os semáforos de carro e pedestre foi um sucesso. A seguir, pode-se visualizar a forma do circuito no final da prática, assim como uma demonstração do seu funcionamento.



**Figure 1. Circuito que ativa o sinal verde para pedestres e o sinal vermelho para carros através de um botão.**

O vídeo que demonstra o circuito de ambos semáforos em funcionamento está disponível em: [Resultado da atividade prática 02](#)

## **6. Conclusão**

Essa atividade prática demonstrou como implementar um circuito acionado por meio de um botão. Ao pressionar, ele realiza a transição entre cores dos LEDs que simulam o semáforo de carros e de pedestres. Para isso foi utilizado um microcontrolador ESP32, uma protoboard, alguns resistores e cabos jumper. Dessa forma, foi possível concluir a atividade e atingir o objetivo proposto.