

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CURSO DE ENGENHARIA ELÉTRICA

DANIELLE SILVA DE OLIVEIRA ROCHA
MARIA EDUARDA LAU MAZON

**SEMÁFORO COM CONTROLE DE PEDESTRES UTILIZANDO
ARDUINO E CI 74HC08**

RELATÓRIO DO PROJETO FINAL DA DISCIPLINA DE ELETRÔNICA
DIGITAL

CURITIBA

2025

DANIELLE SILVA DE OLIVEIRA ROCHA
MARIA EDUARDA LAU MAZON

**SEMÁFORO COM CONTROLE DE PEDESTRES UTILIZANDO
ARDUINO E CI 74HC08**

Relatório do projeto final da disciplina de
Sistemas Digitais da Universidade Tecnológica
Federal do Paraná como requisito parcial para
obtenção da aprovação na disciplina no curso de
Engenharia Elétrica.

Prof. Adriano Ruseler.

CURITIBA

2025

SUMÁRIO

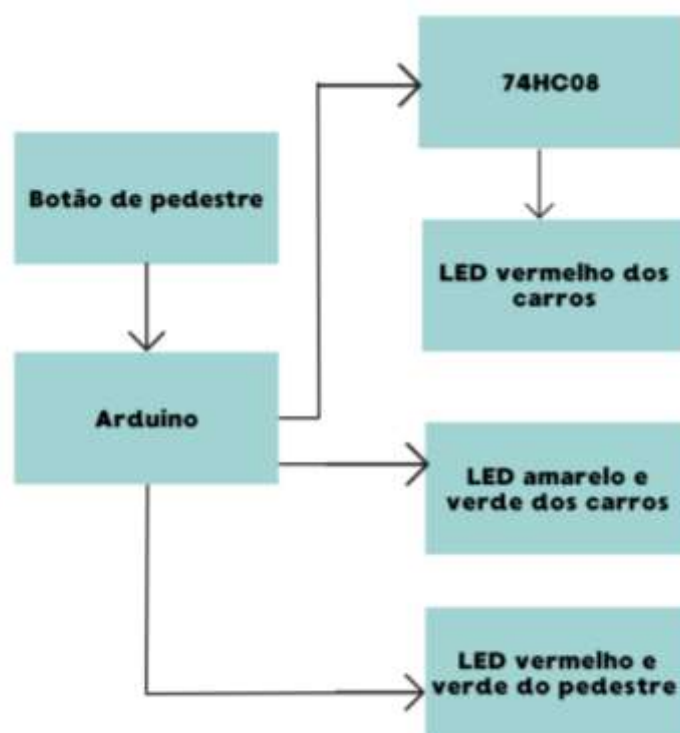
1	INTRODUÇÃO.....	3
2	DIAGRAMA EM BLOCOS DO PROJETO.....	3
3	DESCRIÇÃO DA SOLUÇÃO.....	3
4	LISTA DE MATERIAIS.....	4
5	CIRCUITO ESQUEMÁTICO DO PROJETO.....	4
6	MANUAL DE OPERAÇÃO.....	4
7	PRINCIPAIS DIFICULDADES ENCONTRADAS E SOLUÇÕES UTILIZADAS.	5
8	CONCLUSÕES.....	8
8.1	TRABALHOS FUTUROS.....	8
	REFERÊNCIAS.....	8

1 INTRODUÇÃO

O presente projeto tem como objetivo simular um sistema de semáforo com controle de passagem para pedestres, utilizando um microcontrolador Arduino Uno e o circuito integrado 74HC08. A proposta busca reproduzir, em pequena escala, a lógica de trânsito urbano, onde o pedestre pode solicitar a travessia através de um botão. A motivação deste projeto está na necessidade de integrar conceitos de eletrônica digital com controle lógico usando portas AND.

2 DIAGRAMA EM BLOCOS DO PROJETO

Figura 1 – Diagrama de Blocos.



Fonte: Autoria própria.

3 DESCRIÇÃO DA SOLUÇÃO

O sistema é composto por um Arduino Uno, que controla os LEDs dos semáforos com base no tempo e na leitura de um botão. O CI 74HC08 é utilizado como uma trava lógica apenas para os LEDs dos carros, garantindo que o LED só acenda quando duas condições forem verdadeiras (por exemplo, o Arduino autorizar e o circuito estar apto).

Os LEDs dos pedestres são controlados diretamente pelo Arduino. O sistema dá prioridade ao fluxo de carros, mas ao detectar o botão pressionado, interrompe o fluxo e libera a passagem dos pedestres.

4 LISTA DE MATERIAIS

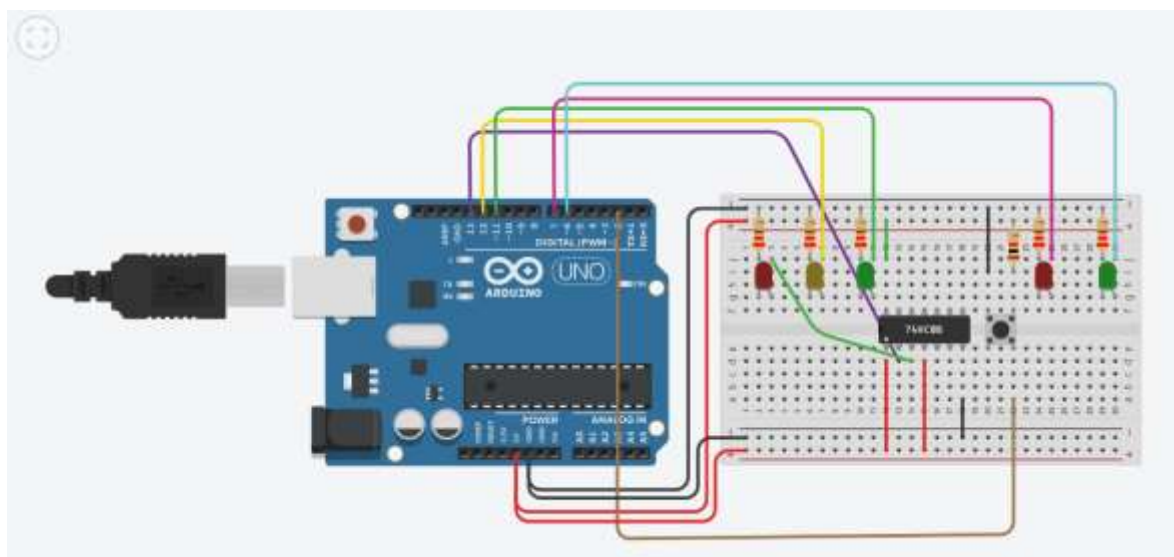
Tabela 1 – Lista de materiais.

Item	Quantidade	Fonte
Arduino Uno R3	1	Loja “So Tudo Componentes Eletrônicos”
LED vermelho	2	Loja “So Tudo Componentes Eletrônicos”
LED amarelo	1	Loja “So Tudo Componentes Eletrônicos”
LED verde	2	Loja “So Tudo Componentes Eletrônicos”
Resistores 220 ohms	5	Loja “So Tudo Componentes Eletrônicos”
Resistor 1k ohm	1	Loja “So Tudo Componentes Eletrônicos”
Botão	1	Loja “So Tudo Componentes Eletrônicos”
CI 74HC08 (Portas AND)	1	Loja “So Tudo Componentes Eletrônicos”
Protoboard + jumpers	Diversos	Loja “So Tudo Componentes Eletrônicos”

Fonte: Autoria própria.

5 CIRCUITO ESQUEMÁTICO DO PROJETO

Imagem 2 – Circuito esquemático Tinkercad.



Fonte: Autoria própria usando o Tinkercad.

6 MANUAL DE OPERAÇÃO

- Ao ligar o sistema, o sinal está verde para os carros e vermelho para os pedestres.
- Quando o botão é pressionado, o sistema espera o momento seguro, aciona o LED amarelo do carro, depois o vermelho, e libera o verde para o pedestre.
- Após 5 segundos, o sistema volta ao estado inicial automaticamente.

6.1 LINK PARA O VÍDEO DEMONSTRATIVO NO YOUTUBE

O vídeo explicando o presente projeto, bem como apresentando as simulações do circuito no Tinkercad e o funcionamento do circuito físico está nesse link: <https://youtu.be/vrCpsXgOvYw>.

Bem como, o circuito montado no tinkercad se encontra nesse link: <https://www.tinkercad.com/things/1PyKWJpS5tw/editel?sharecode=NOI8mIW0Vtc4UgdFi8wUsgWJ5FGRibKn0wvT2ZpgmZ8>.

7 CÓDIGO DE PROGRAMAÇÃO DO PROJETO DO ARDUINO

No Quadro 1 é apresentado o código de programação desenvolvido em linguagem C neste projeto final com os respectivos comentários.

Quadro 1 – Código do projeto final.

```
// Pinos
const int vermelhoCarro = 13;
const int amareloCarro = 12;
const int verdeCarro = 11;
const int vermelhoPed = 7;
const int verdePed = 6;
const int botaoPed = 2;

bool pedestreSolicitou = false;

void setup() {
  pinMode(vermelhoCarro, OUTPUT);
  pinMode(amareloCarro, OUTPUT);
```

```
pinMode(verdeCarro, OUTPUT);
pinMode(vermelhoPed, OUTPUT);
pinMode(verdePed, OUTPUT);
pinMode(botaoPed, INPUT); // Assumindo botão ligado ao GND com pull-up externo

// Estado inicial
digitalWrite(vermelhoCarro, LOW);
digitalWrite(amareloCarro, LOW);
digitalWrite(verdeCarro, HIGH);

digitalWrite(vermelhoPed, HIGH);
digitalWrite(verdePed, LOW);
}

// Função de delay com verificação de botão
void delayComVerificacao(int tempoTotal) {
  for (int i = 0; i < tempoTotal; i += 100) {
    if (digitalRead(botaoPed) == LOW) {
      pedestreSolicitou = true;
    }
    delay(100);
  }
}

void loop() {
  // Fase verde do carro
  digitalWrite(verdeCarro, HIGH);
  digitalWrite(amareloCarro, LOW);
  digitalWrite(vermelhoCarro, LOW);
  delayComVerificacao(3000);

  // Se o botão foi pressionado, interrompe e inicia travessia
  if (pedestreSolicitou) {
```

```
digitalWrite(verdeCarro, LOW);
digitalWrite(amareloCarro, HIGH);
delay(2000);

digitalWrite(amareloCarro, LOW);
digitalWrite(vermelhoCarro, HIGH);

digitalWrite(vermelhoPed, LOW);
digitalWrite(verdePed, HIGH);
delay(5000);

digitalWrite(verdePed, LOW);
digitalWrite(vermelhoPed, HIGH);
delay(1000);

digitalWrite(vermelhoCarro, LOW);
digitalWrite(verdeCarro, HIGH);

pedestreSolicitou = false;
return; // Volta ao começo do loop
}

// Continua o ciclo normal
digitalWrite(verdeCarro, LOW);
digitalWrite(amareloCarro, HIGH);
delayComVerificacao(2000);

digitalWrite(amareloCarro, LOW);
digitalWrite(vermelhoCarro, HIGH);
delayComVerificacao(2000);

digitalWrite(vermelhoCarro, LOW);
}
```


Fonte: Autoria própria.

8 SIMULAÇÃO DO PROJETO

O projeto foi simulado utilizando a plataforma Tinkercad. Durante a simulação, foi possível validar o funcionamento dos LEDs, o tempo de resposta ao botão, a lógica do semáforo e o funcionamento do CI 74HC08 nas saídas dos LEDs dos carros.

9 PRINCIPAIS DIFICULDADES ENCONTRADAS E SOLUÇÕES UTILIZADAS

- Compreensão do funcionamento do CI 74HC08, que foi resolvida com testes estudo sobre portas lógicas AND.
- Organização do circuito na protoboard para evitar curto-circuitos e confusão nos cabos.
- Ajuste do delay do código para garantir uma travessia segura ao pedestre.

10 CONCLUSÕES

O projeto atingiu seus objetivos ao integrar a lógica digital com um sistema prático de semáforo. O uso do CI 74HC08 acrescentou uma camada de lógica por hardware que complementa a lógica por software do Arduino.

10.1 TRABALHOS FUTUROS

Para trabalhos futuros, propõe-se:

- Implementar sinalização sonora para pedestres com deficiência visual;
- Usar sensores para contar veículos e otimizar o tempo do semáforo;
- Testar em protótipos físicos com trânsito real.

REFERÊNCIAS

Datasheet CI 74HC08. Texas Instruments.

Tinkercad Circuits. Disponível em: <https://www.tinkercad.com/>

Documentação oficial do Arduino: <https://www.arduino.cc/>