

Micro Controladores

Prática 2 - Semáforo de Carros e Pedestre com Display de 7 segmentos

Átila Camurça Alves

¹Instituto Federal do Ceará (IFCE)

Abstract. *TODO*

Resumo. *Esta prática tem como objetivo criar um semáforo de carros e pedestres, em que o pedestre pode solicitar sua passagem e um display de 7 segmentos mostra o tempo que ele tem para fazer a passagem. Imagine que este semáforo está em uma auto estrada de alta velocidade. Nesse cenário o sinal para os carros está sempre verde e para os pedestres está sempre vermelho. Quando um pedestre solicita a passagem através de um botão, o sinal para os carros sai do vermelho para o amarelo, logo em seguida para o verde, enquanto isso o sinal dos pedestres vai para o verde e um par de displays de 7 segmentos mostra o tempo que o pedestre tem para passar.*

1. Introdução

Esta prática tem como objetivo criar um semáforo de carros e pedestres, em que o pedestre pode solicitar sua passagem e um display de 7 segmentos mostra o tempo que ele tem para fazer a passagem. Imagine que este semáforo está em uma auto estrada de alta velocidade. Nesse cenário o sinal para os carros está sempre verde e para os pedestres está sempre vermelho. Quando um pedestre solicita a passagem através de um botão, o sinal para os carros sai do vermelho para o amarelo, logo em seguida para o vermelho, enquanto isso o sinal dos pedestres vai para o verde e um par de displays de 7 segmentos mostra o tempo que o pedestre tem para passar.

2. Material

- 5 LEDs
- 7 Resistores 390Ω
- 2 Displays de 7 segmentos
- 1 Cabo USB
- 1 Placa SanUSB
- 1 Protoboard

3. Iteração 1 - Semáforo simples

Inicialmente faremos um semáforo simples apenas dos carros, como mostra a figura a seguir:

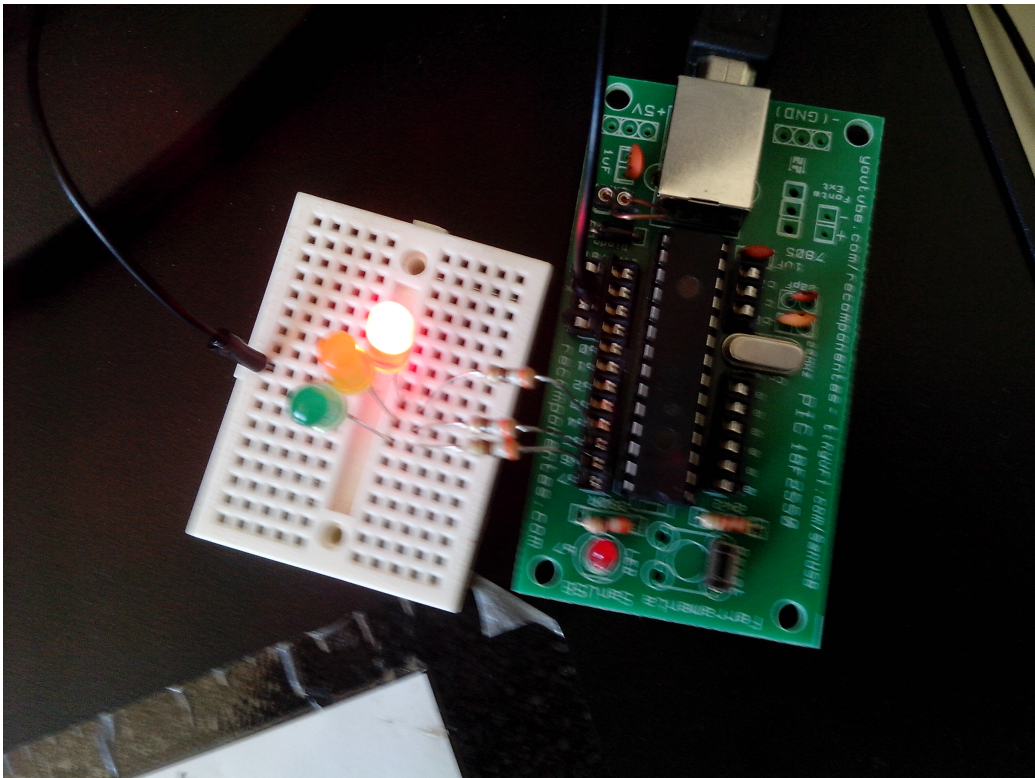


Figura 1. Semáforo Simples de Carros

Segue o código fonte da iteração:

```
#include "SanUSB48X.h"

void interrupt interrupcao() {}

void main(void) {
    clock_int_48MHz();
    while (1) {
        nivel_alto(pin_b7);
        tempo_ms(3000);
        nivel_baixo(pin_b7);

        nivel_alto(pin_b6);
        tempo_ms(1000);
        nivel_baixo(pin_b6);

        nivel_alto(pin_b5);
        tempo_ms(3000);
        nivel_baixo(pin_b5);
    }
}
```

4. Iteração 2 - Semáforo simples para carros e pedestres

Em seguida faremos um semáforo tanto para os carros quanto para os pedestres. O semáforo deve estar sempre verde para os carros e vermelho para os pedestres. Ao acionar um botão o semáforo entra num processo em que o sinal alterna por um tempo.

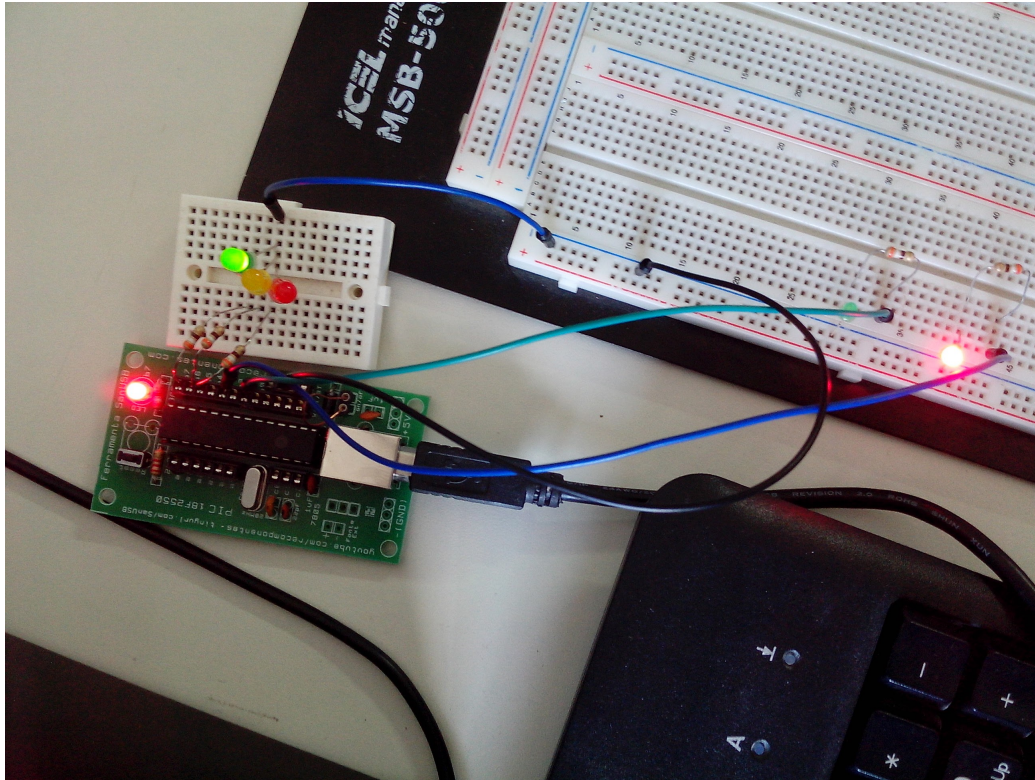


Figura 2. Semáforo simples para carros e pedestres

Segue o código fonte da iteração:

```
#include "SanUSB48X.h"
void interrupt interrupcao() {}

#define verde_carro pin_b7
#define amarelo_carro pin_b6
#define vermelho_carro pin_b5

#define vermelho_pedestre pin_b3
#define verde_pedestre pin_b1

int flag_pedestre = 0, i;

void tempo(int tempo) {
    for (i = 0; i < tempo; i += 100) {
        tempo_ms(100);
        if (! entrada_pin_e3) {
```

```

        flag_pedestre = 1;
    }
}

void main(void) {
    clock_int_48MHz();
    while (1) {
        nivel_alto(vermelho_pedestre);
        nivel_baixo(vermelho_carro);

        nivel_alto(verde_carro);
        tempo(10000);
        nivel_baixo(verde_carro);

        if (flag_pedestre) {
            nivel_alto(amarelo_carro);
            tempo_ms(1000);
            nivel_baixo(amarelo_carro);

            nivel_alto(vermelho_carro);
            nivel_baixo(vermelho_pedestre);

            nivel_alto(verde_pedestre);
            tempo_ms(5000);
            nivel_baixo(verde_pedestre);
            flag_pedestre = 0;
        }
    }
}

```


5. Iteração 3 - Display de 7 segmentos com contagem regressiva

Agora que temos um semáforo precisaremos usar um display de 7 segmentos que irá mostrar a contagem regressiva para que o sinal verde do pedestre vá para vermelho. Mas antes vamos fazer apenas que o display conte de zero a nove indefinidamente.

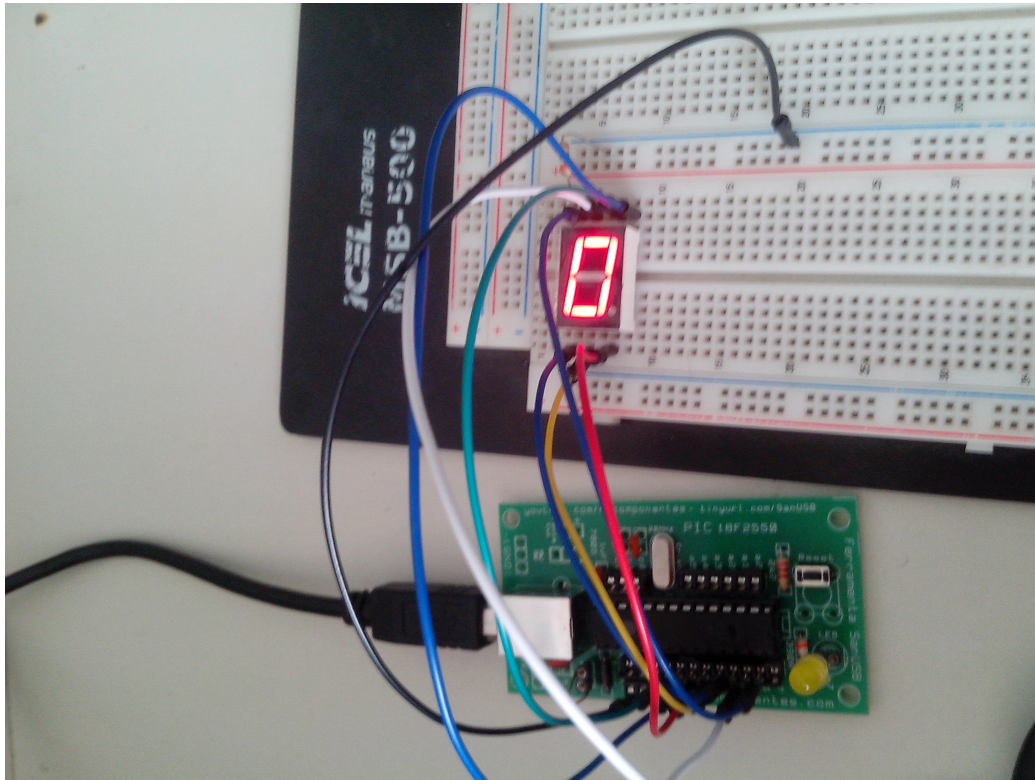


Figura 3. Display de 7 segmentos com contagem regressiva

Referências

[Grupo SanUSB 2011] Aplicações Práticas de Eletrônica e Microcontroladores em Sistemas Computacionais, https://www.dropbox.com/s/0e8r2sh94x9enof/2%20-%20Apostila_MPLABX.pdf?dl=0, 27 07 2016.