

**UNIVERSIDADE DE SÃO PAULO  
ESCOLA DE ENGENHARIA DE SÃO CARLOS  
ENGENHARIA MECATRÔNICA**

Gustavo Fernandes Lodi - 10308801

**SEL0336 - Aplicação de Microprocessadores I (2021)**

**Prática 3 – Temporizador com Interrupção**

## 1 INTRODUÇÃO

O trabalho teve como finalidade a criação de um contador binário com incrementos de períodos de  $1ms$  por meio do uso do *timer* interno do microprocessador. O contador de 8 *bits* varia de 0 a 255.

## 2 ESPECIFICAÇÕES E RESULTADOS

- Implementar o contador binário de 8 *bits* e frequência de 1 kHz com display resultante na placa de LEDs;
- Programar o intervalo com o uso do timer interno do microprocessador.

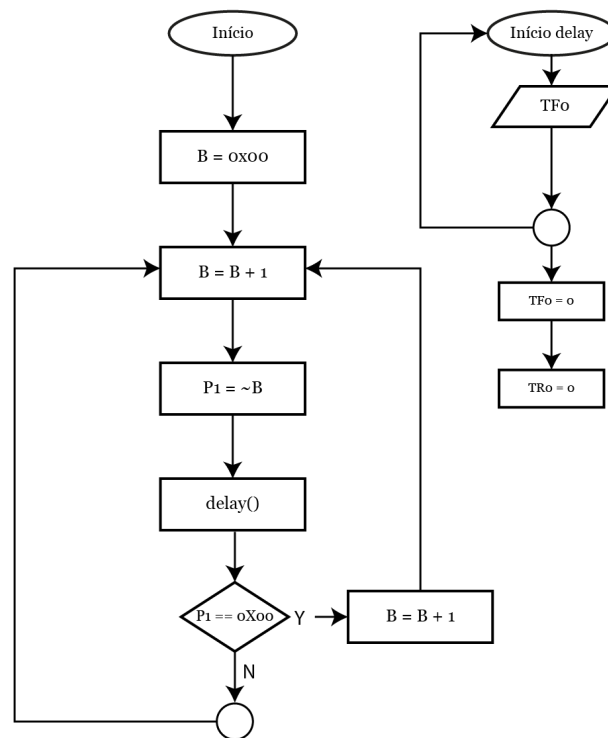


Figura 1 - Fluxograma Geral

Os sete LEDs foram conectados aos bits da porta P1. Para o código, foi utilizada a biblioteca do microprocessador at89s52 e os comentários foram descritos ao longo da aplicação. Como ilustrado no fluxograma acima, o contador recebe o valor 0x00 e, após isso, o programa entra em um loop, incrementando o valor do contador B em uma unidade. As ações que se seguem são a porta P1 recebe o valor *NOT* do contador B, uma vez que usa-se a configuração ânodo comum (LED aceso em nível lógico 0), em seguida há o intervalo de  $1ms$  e reinicia-se o loop. Importante destacar, a configuração do Timer 0 é feita por meio do cálculo do tempo requisitado. A especificação de tempo é  $1ms$ , então, obtém o valor a ser inserido no timer como demonstrado:

A frequência de *clock* é 12000MHz e são 12 pulsos por ciclo de máquina, portanto, o período é  $1/1000 = 1$  microssegundo. Para o intervalo de 1 milissegundo, têm-se que 1000 pulsos do microprocessador são necessários para o timer. O valor 1000 em binário é 3E8, que subtraído de FFFF, torna-se FC17, valor a ser inserido no timer.

Porém, as demais instruções presentes nas linhas de código somam um intervalo de 29 microssegundos, portanto, torna-se necessário a compensação no intervalo do timer desse valor, o qual torna-se 971 microssegundos, que em hexadecimal é igual a 3CB e, subtraído de FFFF, torna-se FC34. Este valor sendo o mais preciso.

A função auxiliar *configTimer()* seta o valor calculado no Timer0 e inicia seu funcionamento. O Timer0 apenas se encerra com a flag da variável TF0 acionada como 1. A prévia do funcionamento pode ser visualizada nas imagens a seguir.

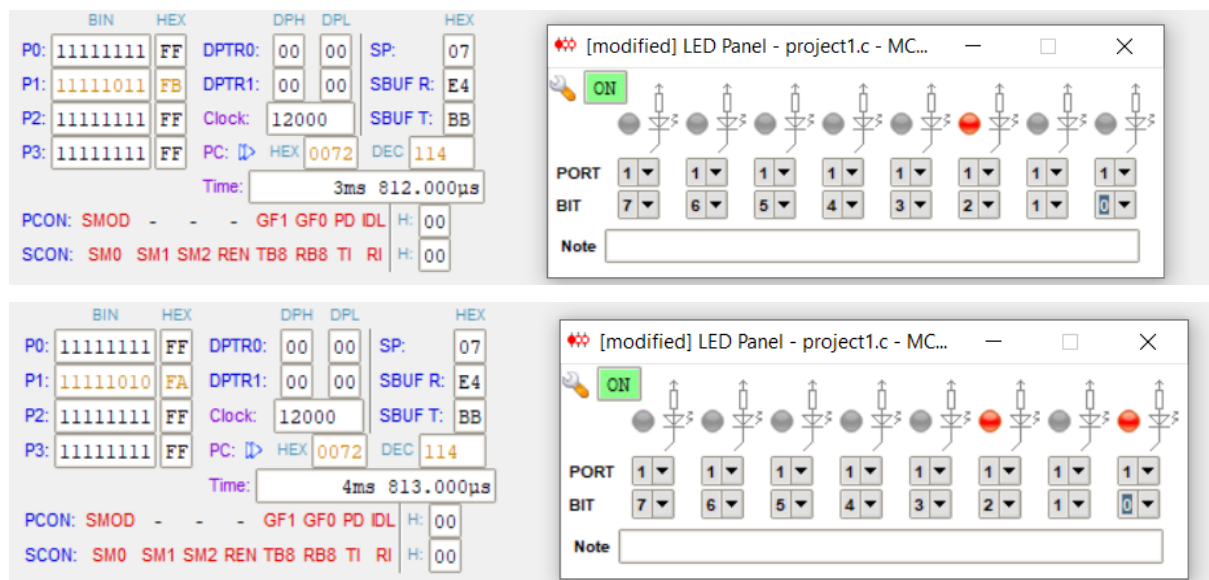


Figura 2 - Resultado da simulação

### 3 PINAGEM

LEDs	P1_7	P1_6	P1_5	P1_4	P1_3	P1_2	P1_1	P1_0
-	MSB	-	-	-	-	-	-	LSB

Tabela 1 - Pinagem da simulação

## **4 CONCLUSÃO**

A simulação atendeu aos requisitos do projeto e foi bem sucedida na realização da contagem binária de 8 bits pelos LEDs a uma frequência de 1kHz. Ademais, o timer interno mostrou-se mais preciso que estruturas de repetição da própria linguagem C, permitindo maior controle pelo usuário com o intervalo desejado.