Texto

Departamento de Engenharia Elétrica e de Computação

# SEL 455–Lab. de Sistemas Digitais

Profa. Luiza Maria Romeiro Codá

**“Cronômetro”**

1. **Objetivos:**

* Utilização do programa QUARTUSII da empresa ALTERA no projeto, simulação e teste de circuitos seqüenciais em dispositivos “HCPLDs”

**2. Material utilizado:**

* Dispositivo Programável de Alta Complexidade HCPLD do tipo FPGA Cyclone IV da Altera
* Módulo de desenvolvimento Mercúrio IV – Macnica DWH
* Multímetro
* Osciloscópio

**3. Procedimento Experimental:**

Utilizando o dispositivo FPGA EP4CE30F23C7 da família Cyclone IV-E da placa mercúrio IV da macnica, projete, simule e teste os seguintes circuitos(Observação: O nome e pinagem dos pinos desse circuito estão listados na Tabela 2)

* 1. Projetar e programar um circuito que transforme a freqüência de relógio(CLOCK\_50MHz) da placa de testes para uma freqüência de 10 Hz (frequência da placa mercúrio é 50MHz), como mostra a Figura 1. Verifique a saída (10Hz) no LED\_B. Para isso importe o arquivo de pinagem da placa mercúrio, que se encontra no moodle, e compile novamente

, se = 50MHz Módulo = 5000000

Criar um contador módulo 5000000 que representado em binário é apresenta (10011000100101101000000)b , o qual apresenta 23 bits, ou seja, o contador a ser implementado deve ter 23 Flip Flops, e a saída do último FF apresenta a frequência de 10Hz.

Utilize o projeto lpm\_counter para implementar um contador de 23 bits e módulo 5000000 e nomeie as saídas como f[22..0]. A saída f[22] é um sinal de 10hz.

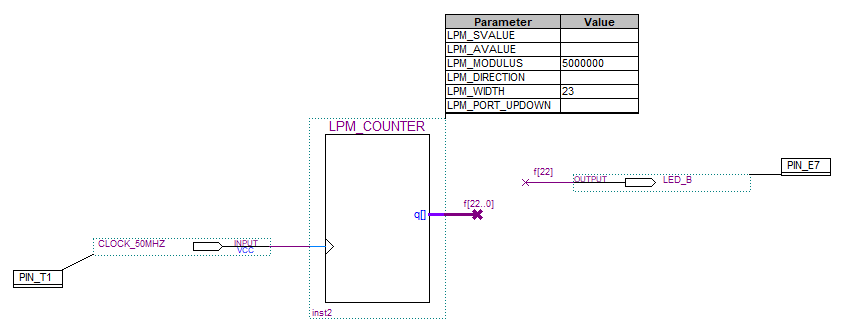


Figura 1 divisor de frequência

* 1. Visualização de 10Hz na matriz de LEDs:

Sabendo que um LED da matriz de LEDS ( Figura 2) acende quando a Coluna e a linha específica estiverem em nível ‘0’, projete um circuito que deve acender todos os LEDS da mesma coluna, e ir acendendo a coluna subsequente ( da direita para a esquerda de (C[0] e por último a coluna 4 (C[4] ) até a matriz ficar totalmente acesa. Então, deve ir apagando os LEDS da última coluna até a primeira e recomeçar a acender a cada 10Hz, como mostra a Figura 3.

Utilize o componente lpm\_shift\_reg , que é um registrador de deslocamento, para funcionar como entrada serial (shiftin). Crie esse registrador com 5 saídas que serão ligadas às colunas da matriz (LEDM\_C[4..0]). A entrada clock é a frequência de 10Hz. O circuito do acionamento dos LEDs da matriz é mostrado na figura 4. As linhas da matriz devem ser todas mantidas no nível baixo, Passe para o dispositivo e teste o funcionamento.

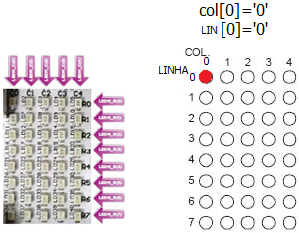


Figura 2 Matriz de LEDs

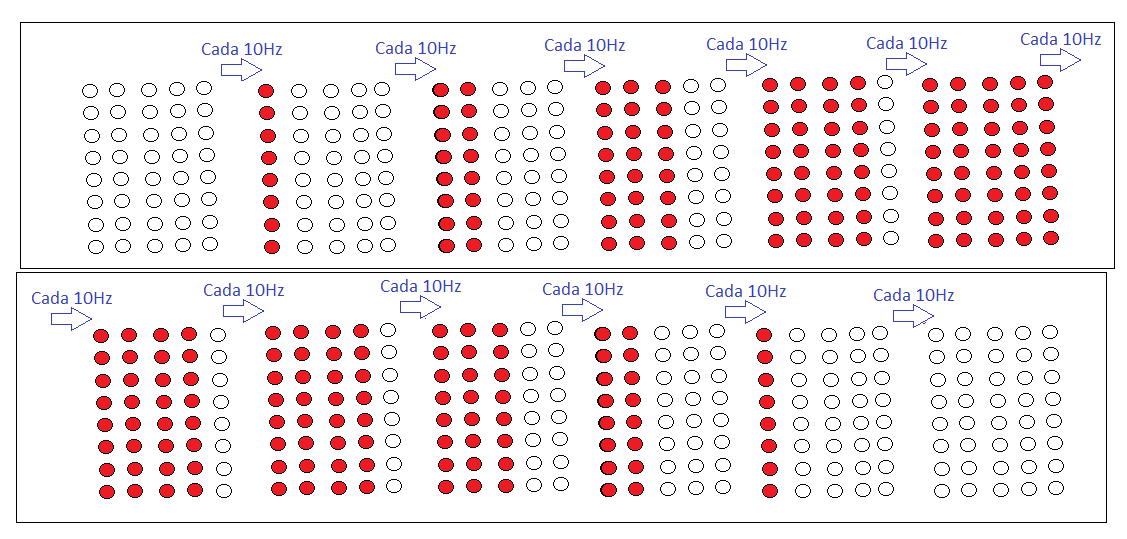


Figura 3 LEDS de cada coluna da Matriz a cada 10Hz

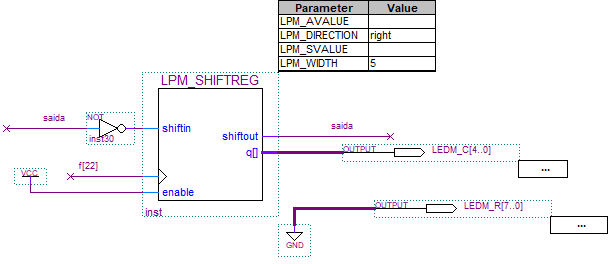


Figura 4 Circuito de visualização do 10Hz na matriz de LEDs

* 1. Implementação do cronômetro:

Projete e implemente um cronômetro que conte 60 segundos. Mostrar a contagem nos displays de 7 segmentos da placa.

Utilize o projeto lpm\_counter para fazer um contador módulo 10 para gerar a unidade do cronômetro, onde o clock deste contador é uma frequência de 1Hz. Utilizar um componente lpm\_counter para dividir a frequência de 10Hz da saída f[22] por 10, como mostra a Figura 5.

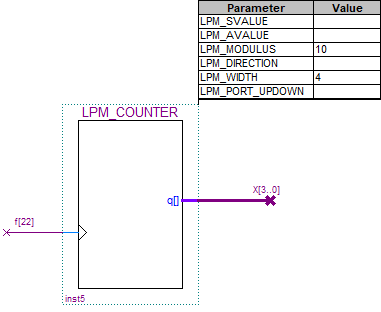


Figura 5 Divisor para obter frequência de 1Hz

Para implementar o contador da unidade, utilize o projeto lpm\_counter para fazer um contador módulo 10. Use como clock a frequência de 1Hz.

Ligar às saídas desse contador um decodificador BCD para display de 7 segmentos catodo comum, CI 7448 (Figura 6)

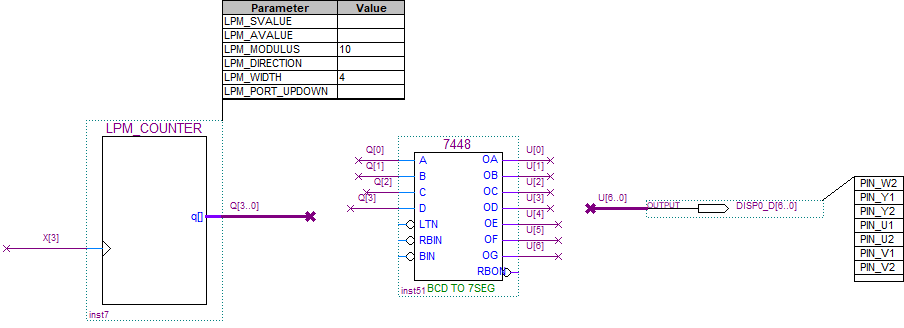


Figura 6 Contador da unidade com decodificador BCD para display de 7 segmentos

Para implementar o contador da dezena, utilize o projeto lpm\_counter para fazer um contador módulo 6 para gerar a dezena do cronômetro, onde o clock deste contador é saída mais significativa invertida do contador módulo 10 (pois o lpm\_counter é sensível à subida do clock e é preciso que seja sensível a descida para mudar o valor qdo o bit mais significativo passa de 0 para 1 , de (101)b para (000)b. Às suas saídas ligue a um decodificador BCD para display de 7 segmentos catodo comum, CI 7448 (Figura 7).

Compile e sintetize no FPGA para verificar o funcionamento dos displays.

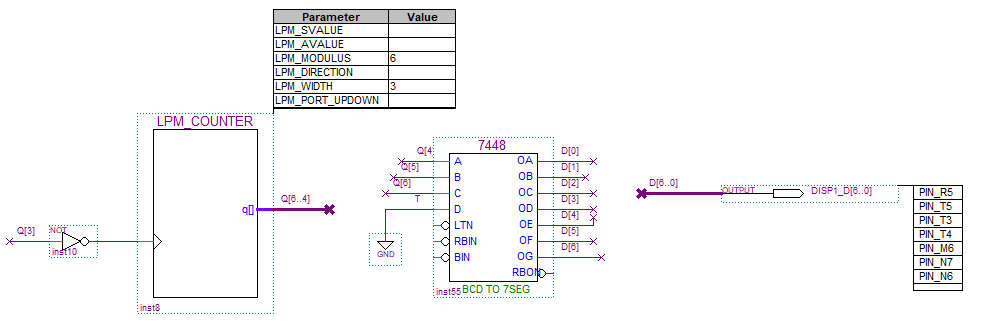


Figura 7 Circuito da Dezena do cronômetro

* 1. Zerar a contagem:

Utilize uma chave push button ( botão (KEY[0] da placa) para zerar a contagem. Para implementa-lo modifique os contadores módulo 10 e Módulo 6 do item 3.3 para incluir uma entrada clear assíncrona (aclear ) na qual deve ser ligada a chave KEY[0] ( mostrada na Figura 8), O circuito modificado para zerar é mostrado na Figura 9.

Uma imagem contendo Diagrama

Descrição gerada automaticamente

Figura 8 Chaves do Módulo Mercúrio IV

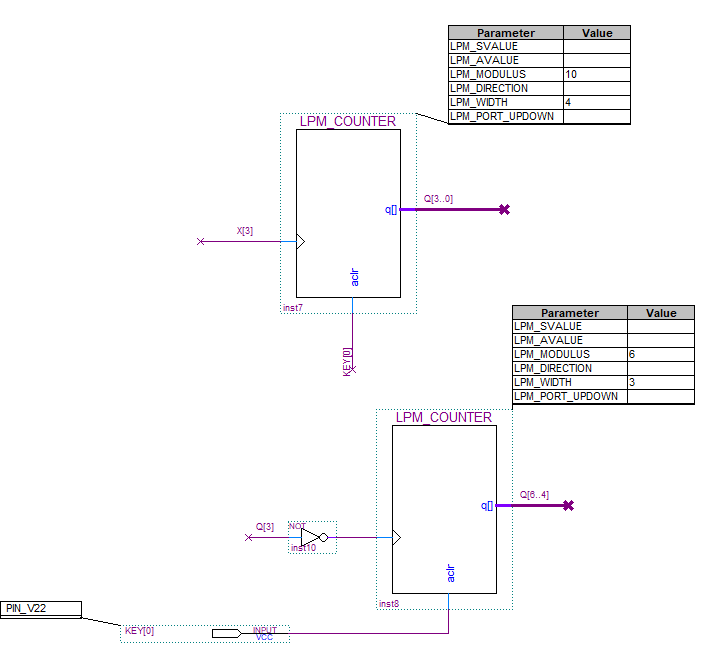
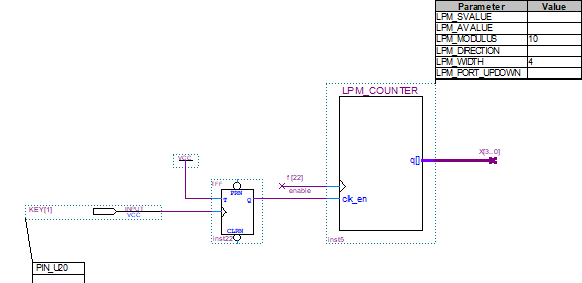


Figura 9 inclusão de botão para zerar ( chave KEY[0]) nos contadores do cronômetro

3.5 Parar a contagem:

Utilizar uma chave Push Button (KEY[1] (mostrada na Figura 8) para iniciar e/ou parar a contagem. Dessa forma, modifique o circuito do divisor (utilizado para obter a frequência de 1Hz), incluindo uma entrada clock enable. Essa entrada deve ser conectada à saída Q de um Flip Flop tipo T no modo Toggle, cujo clock é a chave de parar KEY[1], como mostra a Figura 10

.



Incluir eliminador de ruido

Figura 10 Inclusão do botão para parar a contagem (chave KEY[1]).

* 1. **Eliminador de ruído PB** : Caso o ruído mecânico da chave PB (chave de parada) cause mal funcionamento no circuito, implementar um eliminador de ruído de chave PB, mostrado na Figura 11, e ligar a chave PB à ele. O clock deve ter 100Hz, e deve ser obtido implementando um divisor de frequência a partir da frequência da placa (50MHz). Utilize o projeto LPM\_COUNTER para criar a a frequência de 100Hz e os projetos DFF para implementar o Flip flop D, o projeto AND2 para a porta and. O projeto do eliminador no software é mostrado na Figura 12.

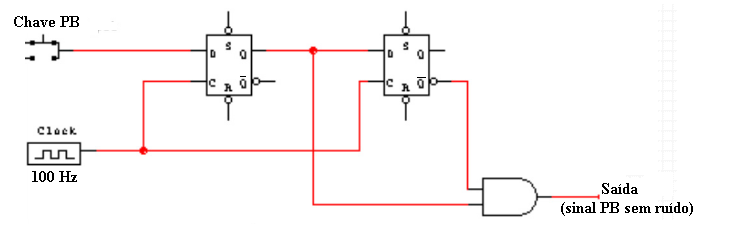


Figura 11 circuito eliminador de ruído de chave PB

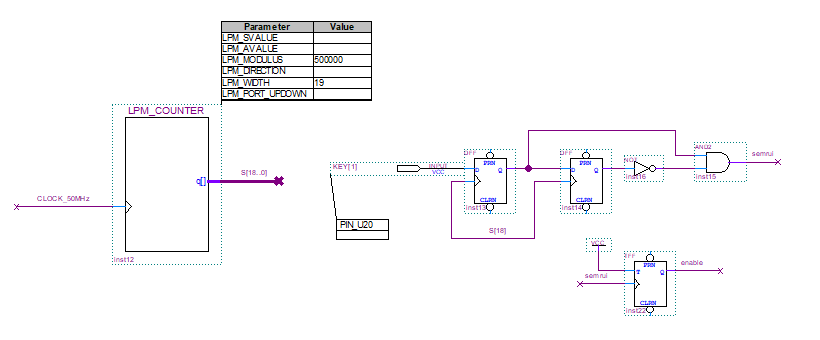
****

Figura 12 Implementação do eliminador de ruído PB

A saída enable da Figura 12 deve ser ligada na entrada “enable” do registrador de deslocamento e na entrada clk\_en do divisor de 1Hz para parar os LEDs da matriz e a cronômetro, como mostra a figura 13

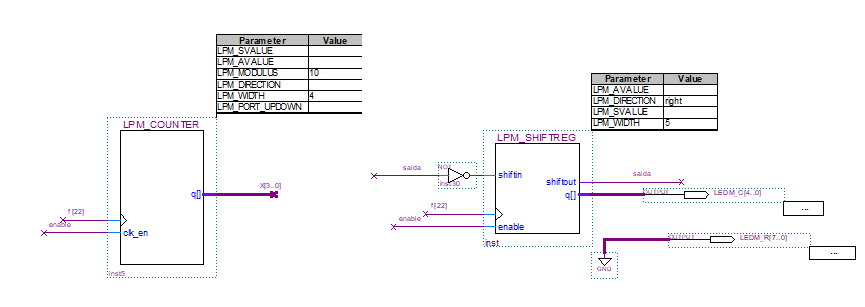


Figura 13 Divisor e circuito da matriz de LEDS com a entrada enable para parar

Compile e sintetize no FPGA.

**4.Bibliografia:**

* Site da ALTERA
* Fregni, E. & Saraiva, A.M., “ Engenharia do Projeto Lógico Digital”, Ed. Edgard Blücher Ltda.
* Tocci, J. R. , “Sistemas Digitais- Princípios e Aplicações