

Universidade Federal de São Carlos

Departamento de Computação

Disciplina:

Laboratório de Arquitetura e Organização de Computadores 1

Relatório 2

Circuitos Reconfiguráveis: Projeto Sequencial e Hierarquia.

Daniel Chaves Macedo RA: 280844

Pedro Gabriel Artiga RA: 351180

Felipe Fiori Campos Martins RA: 316660

São Carlos, 2011.

Introdução

Utilizando o software Altera Quartus II, simular um projeto para testar o funcionamento do circuito Debounce, para tal a simulação de formas de ondas fornecida pelo Quatus II, além disso testaremos o funcionamento de barramentos e implementações em linguagem VHDL.

Objetivo

Este relatório tem como objetivo auxiliar a compreensão do funcionamentos de circuitos a partir da simulação de ondas, o conceito de barramentos, além de iniciar o modelamento de projetos em linguagem VHDL.

Outro objetivo importanté é reconhecer o efeito que partes mecânicas podem ter sobre o circuitos, especialmente os que são sensíveis a oscilações.

Experimento 1 - Contador crescente de 8 bits com clear assíncrono

Contadores são circuitos sequenciais que incrementam seu valor sempre que recebem um clock positivo ou saem do clock positivo. Para demonstrar o funcionamento deste nós criamos um circuito que captura o valor de um destes contadores e o exibe em dois displays de sete segmentos. Um destes contadores receberá os 4 bits mais significativos do contador enquanto o outro receberá os 4 bits menos significativos.

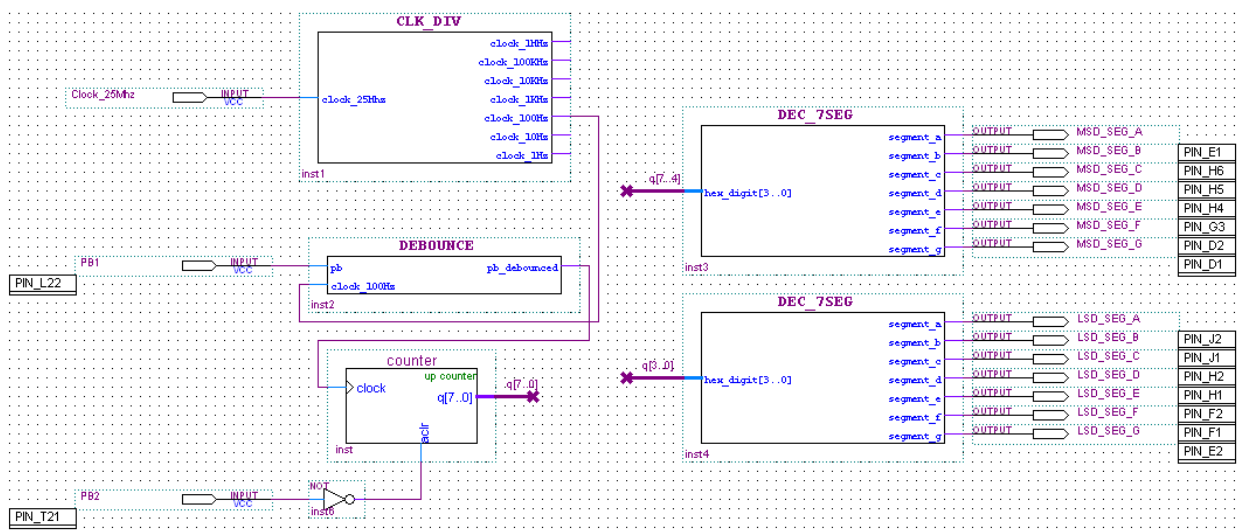


Figura 1: Contador de 8 bits ligado a 2 decodificadores de sete segmentos.

Como é possível perceber na figura acima, existe um circuito Debouncer entre o CLOCK DIV e o COUNTER, isso se deve ao fato de chaves mecânicas gerarem trepidações que podem causar problemas no circuito, afetando o resultado mostrado nos displays, desse modo devemos utilizar um circuito Debouncer entre eles, para “limpar” a saída da chave.

O circuito foi projetado no programa Altera Quartus, utilizamos no projeto um clock de 100 Hz. É importante citar que os nomes dos pinos foram mantidos conforme os especificados na placa.

Após terminado o projeto, carregamos o mesmo na placa DE1 através de sua interface USB, possibilitando o uso das chaves não filtradas como entradas.

Os testes demonstraram que o circuito funciona como o esperado, sendo que toda vez que a chave mecânica é pressionada o contador é incrementado em apenas um.

Experimento 2 - Relógio

Contadores são circuitos sequenciais que incrementam ou decrementam o valor composto destes circuitos a cada pulso de clock.

O componente básico de um contador é o circuito Flip-Flop. No caso desta implementação o modelo de Flip-Flop escolhido foi o tipo T.

O Flip-Flop tipo T funciona invertendo o valor armazenado nele a cada ciclo de clock caso a entrada seja alta. Caso a entrada seja baixa o valor atual é mantido.

Nos contadores assíncronos o clock do primeiro Flip-Flop é ativado por um controle do usuário ou de forma automática de frequência constante, enquanto o clock de todos os outros Flip-Flops são ativados pela saída do Flip-Flop representando o bit imediatamente menos significativo. As entradas de todos os Flip-Flops (neste caso do tipo T) são então ligadas ao VCC para garantir que a inversão de estado ocorra a cada pulso de clock recebido.

Cada contador representa um dígito exibido no relógio. Quando aquele dígito atinge o valor máximo que ele deve atingir aquele contador é zerado e o próximo contador recebe um clock indicando que deve ser incrementado.

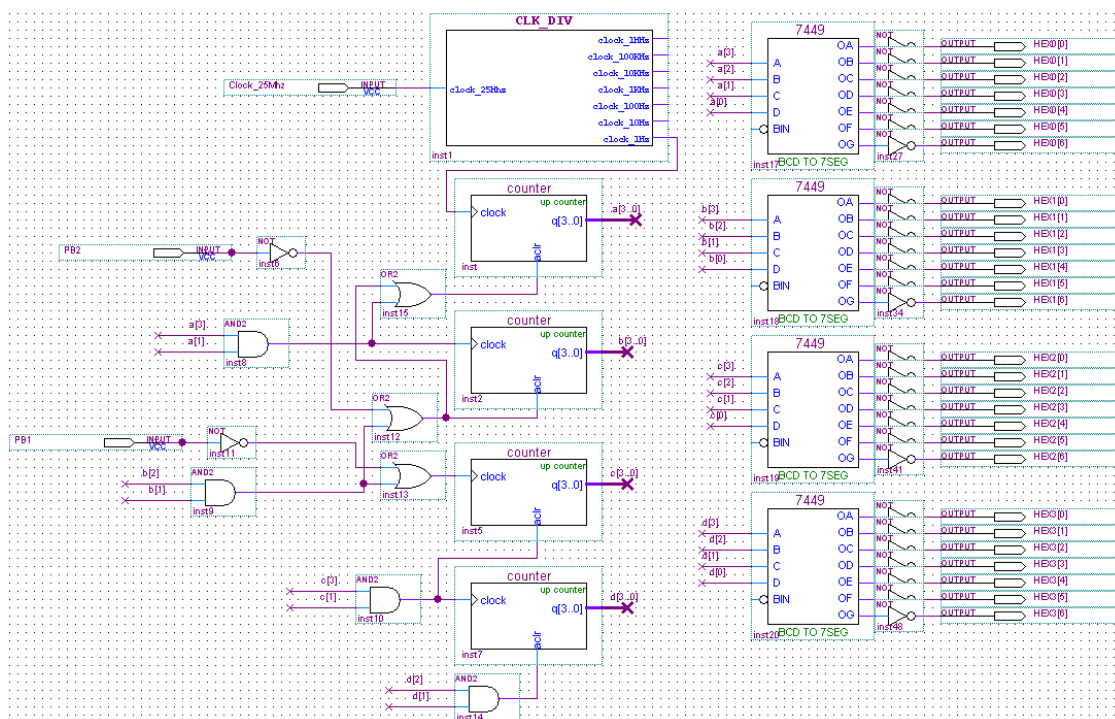


Figura 2: Relógio implementado com quatro contadores de quatro bits

Para mostrar a contagem progressiva tudo que é necessário é ligar a saída de cada Flip-Flop a algum

método de exibição ou transmissão na ordem correta de significância dos bits.

Na contagem regressiva tudo que é necessário é negar cada saída dos Flip-Flops e exibí-las na ordem invertida de significância.

Na Figura 2 é possível notar a aplicação de todos os conceitos descritos para implementar o diagrama esquemático de um contador assíncrono regressivo.

Experimento 3 - Memória ROM

Memória ROM é uma aplicação da memória tradicional, com a diferença de que ela não pode ser sobre-escrita com novas informações nos sistemas onde ela é utilizada.

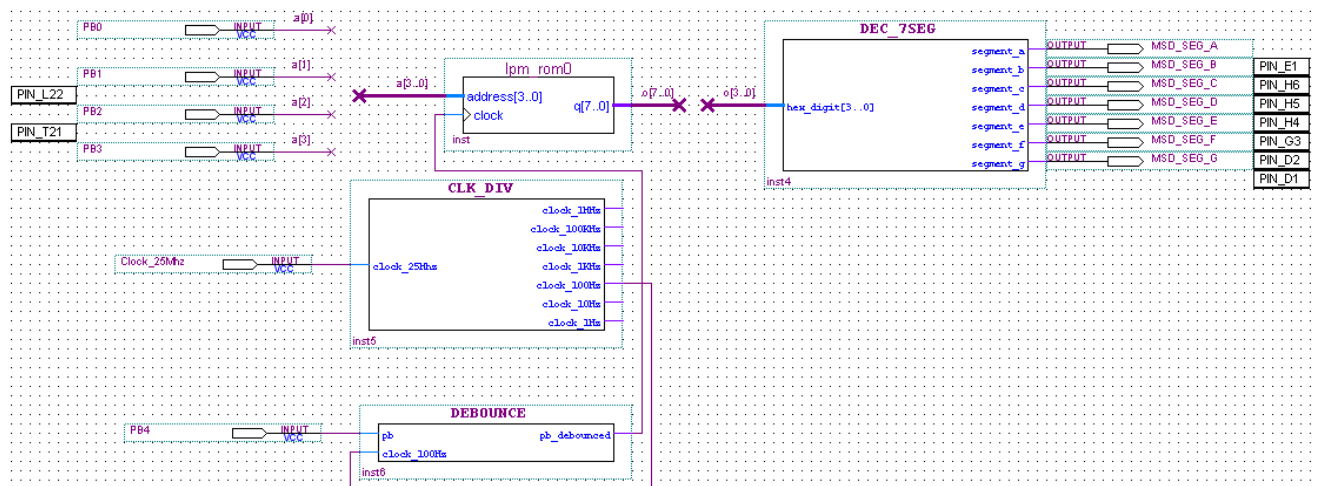


Figura 3. Implementação de memória ROM

Neste caso o Altera exige que a memória seja inicializada através de um arquivo de inicialização de memória. No circuito acima você escolhe qual posição deseja acessar através de quatro chaves mecânicas e através do clock filtrado a memória ROM exibe na sua saída a informação armazenada na posição informada. Neste exemplo a informação é decodificada e exibida em um display de sete segmentos.

Conclusão

A função que gera um simulador de funções, são de grande ajuda para analisar o funcionamento dos circuitos utilizados no projeto, além disso pudemos perceber as variações que uma chave mecânica sem filtro pode causar em um circuito sensível e a utilidade de um circuito Debouncer.

Outra questão interessante é a utilidade da linguagem VHDL, que facilita a esquematização de circuitos de grande porte e complexidade.