

LABORATÓRIO DE SISTEMAS DIGITAIS

EXPERIMENTO: VHDL 3

DEE – CT – UFRN

Prof.: Bruno de Melo Pinheiro

Objetivo

Projetar, elaborar o código e implementar a máquina de estados descrita abaixo, usando Linguagem de Descrição de Hardware (VHDL), e realizar a simulação do mesmo, usando o FPGA disponível em sala.

Descrição da atividade

Implemente um projeto em VHDL, que represente um sistema digital composto pelos seguintes componentes, e contendo o seguinte comportamento:

Componente 1 (Unidade Lógica Aritmética – ALU)

Esse componente deve ser um bloco contendo diversos outros blocos internos, os quais devem desempenhar uma função lógica ou aritmética. As opções de blocos disponíveis estão listadas abaixo. Cada aluno deve escolher no mínimo 4 blocos para compor sua ALU. Todos os blocos devem considerar entradas e saídas de 4 bits para realizar as suas operações. Os blocos lógicos devem fazer suas operações bit a bit, com saída apresentada em LEDs que representem os bits de saída, enquanto os blocos aritméticos devem ter sua saída apresentada em displays de 7 segmentos, para verificação do resultado no formato decimal. Por fim, a escolha da saída mostrada pela ALU deve ser feita através do componente 2 (máquina de estados).

- Somador
- Subtrator
- Incrementador/Decrementador
- Multiplicação
- Divisão
- Deslocador à direita
- Deslocador à esquerda
- Rotação
- XOR
- NAND

Componente 2 (Máquina de Estados – Controlador)

Esse componente deve conter um estado **inicial** para zerar quaisquer valores em LEDs ou displays e seguir para um estado de **espera**. Nesse estado de espera o usuário deve interagir com a máquina para escolher a operação que será realizada. Use sua criatividade para implementar uma forma de fazer essa etapa ser o mais interativo possível. Após escolher a operação da ALU, o usuário deve confirmar a escolha, fazendo a máquina seguir para o terceiro estado. Nesse estado o usuário deve escolher o valor da **primeira variável** de entrada ($A - 4$ bits), através de botões *deep switch*, e confirmar seu valor antes de partir para o estado seguinte. No quarto estado o usuário deve usar os mesmos botões para escolher o valor da **segunda variável** ($B - 4$ bits). Finalmente, após essa confirmação, a máquina deve seguir para um último estado, onde deve ser mostrado o **resultado da operação**. Qualquer confirmação nesse estado deve retornar para o estado de espera.

Esse componente, como o próprio nome diz, deve apenas controlar as operações, as quais são realizadas na ALU (Componente 1). Para realizar isso de forma mais simples, faça com que a saída desse bloco seja exatamente o sinal de seleção do MUX contido dentro da ALU, além de um conjunto de LEDs para identificar o estado em que a máquina se encontra e também uma lógica que permita o funcionamento correto das saídas da ALU apenas no estado final.

LABORATÓRIO DE SISTEMAS DIGITAIS

EXPERIMENTO: VHDL 3

DEE – CT – UFRN

Prof.: Bruno de Melo Pinheiro

Orientações e dicas extras

Construa inicialmente a máquina de estados com todos os estados propostos em um arquivo de código *.vhd. Lembre-se de criar as saídas para mostrar através de LEDs o estado em que a máquina se encontra. Não esqueça também de reservar uma saída para selecionar a operação que o usuário deseja fazer. Essa saída da seleção será conectada a ALU (Componente 1) através do comando de PORT MAP, no código final.

A seguir, construa cada um dos blocos internos da ALU, considerando sempre que suas entradas são A e B, ambas de 4 bits. Para as operações que dependem apenas de uma entrada, considere apenas a entrada A. Cada bloco será um arquivo simples *.vhd. Crie agora a ALU propriamente dita, a qual fará a declaração de cada um desses componentes e a conexão interna entre eles, através do comando PORT MAP.

Finalmente, crie um último arquivo *.vhd, que será simplesmente o código que vai chamar o Componente 1 (ALU) e o Componente 2 (Controlador) e vai conectá-los entre si, usando novamente o comando PORT MAP.

OBS: Todas as orientações dadas para esse projeto são dicas para uma forma mais intuitiva de fazê-lo. Caso o aluno tenha ideias diferentes de fazer algum dos procedimentos de forma mais interativa, criativa ou robusta, também poderá fazer dessa forma. A única orientação que não pode ser feita de forma diferente é a base do projeto, a qual consiste em **construir os blocos separadamente, e interliga-los usando a declaração COMPONENT e o comando PORT MAP**.

Melhorias no projeto serão tratadas como pontos extras para essa atividade específica.

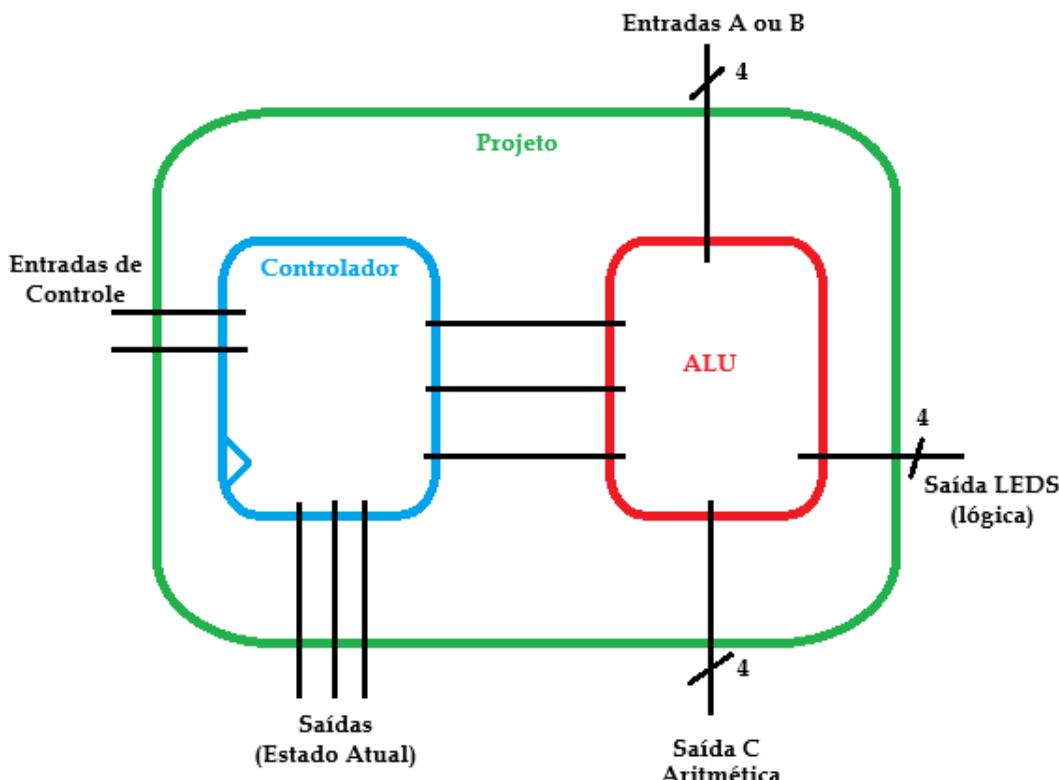


Figura 1 - Esquema básico dos blocos do projeto.