

Sua tarefa é projetar, implementar em VHDL e testar um circuito que efetue uma aproximação inteira de um filtro Gaussiano de duas dimensões.

Dada uma matriz X de inteiros positivos representados em 8 bits, seu circuito deve computar a matriz Y , segundo:

$$Y(i, j) = 1/2X(i, j) + 1/2\{ 1/4[X(i - 1, j) + X(i + 1, j) + X(i, j - 1) + X(i, j + 1)] \}$$

Seu circuito deve percorrer a matriz de entrada X com $L \times C$ pontos e produzir a matriz de saída Y com os pontos computados como indicado acima. O tamanho máximo dos lados da matriz de entrada é 1024. Os valores dos pontos nas bordas da matriz de entradas são fixos.

Para cada valor de Y são necessárias, no máximo, 5 operações de leitura de X , e uma escrita em Y . Passado um período de inicialização, sua implementação deve produzir um novo valor de Y a cada $N \leq 6$ ciclos do relógio. Para tanto, o circuito deve ser segmentado em tantos estágios quanto os necessários para efetuar todas as leituras, o cálculo da média, e a escrita em Y . O modelo do circuito do filtro deve ser um misto de estrutural e de RTL.

Especificação:

1. O trabalho pode ser efetuado em duplas;
2. o arquivo com os produtos deve ser nomeado `xx-yy.tgz` sendo `xx` e `yy` os *usernames* dos componentes do grupo, e todos os arquivos relevantes deverão estar abaixo do diretório `xx-yy`;
3. copie o arquivo `www.inf.ufpr.br/roberto/ci312/trab12pri.tgz` para sua área de trabalho. O arquivo `LEIAME` contém descrição dos conteúdos de `trab12pri.tgz`. Estude os *scripts* e código VHDL antes de usá-los;
4. PLÁGIO NÃO SERÁ TOLERADO. É interessante que os alunos conversem sobre o projeto mas cada grupo deve escrever seu próprio código.

Produtos:

1. Relatório em papel A4, com letras em 11 pontos, espaço simples, formatação simples, contendo os nomes dos componentes do grupo, e o código VHDL do seu projeto, que deverá estar todo contido no arquivo `filtro.vhd`.
2. arquivo enviado por e-mail para `roberto@inf.ufpr.br` contendo todos os arquivos fonte necessários para testar o projeto. Projetos com arquivos faltando e que não possam ser testados receberão nota zero. Todos os programas serão re-compilados antes de simulados na avaliação, e a matriz de testes será alterada para a verificação do trabalho;
3. presença dos membros do grupo na data e hora marcadas para a apresentação.

Sugestões:

1. Assegure-se de que entendeu a especificação antes de iniciar o projeto do circuito;
2. o que ocorre com as bordas da matriz? O que ocorre em caso de *overflow* nas somas?
3. O código do filtro deve ser escrito como um modelo estrutural, com elementos em RTL, porque se o modelo do filtro contiver código no estilo comportamental, será muito difícil obter o comportamento temporal de um circuito segmentado.

Histórico das Revisões:

26set primeira versão.