

Computação Reconfigurável

Aula prática 2

Resumo:

1. Parte da VHDL apresentada na aula teórica 2.
2. Simulação comportamental em Vivado utilizando VHDL.
3. Síntese de circuitos em Vivado a partir de descrição em VHDL.

Objetivos

Realização de um conjunto de exercícios práticos introdutórios para:

- Simulação comportamental de circuitos a partir de descrição em VHDL;
- Utilização de processos combinatórios e sequenciais;
- Descrição de projetos utilizando VHDL comportamental, estrutural e misto;
- Desenvolvimento, síntese, implementação e verificação de circuitos simples em FPGA.

Materiais adicionais úteis:

- WebPack de Vivado: <http://www.xilinx.com/products/design-tools/vivado.html> .
- Xilinx Design Constraints File XDC: [elearning](http://www.xilinx.com/learn/learning).
- Manual de utilização da placa Nexys-4 : <http://www.digilentinc.com/Products/Detail.cfm?NavPath=2,400,1184&Prod=NEXYS4>.

Problemas para resolver

Resolver exercícios da aula anterior (aula prática 1).

Exercício 2.1. Descreva em VHDL comportamental o decodificador de valores binários (4 bits) para 7 segmentos de um display disponível na placa Nexys-4 (pode escolher qualquer um de 8 displays disponíveis):

- a) Usar construção **when ... else** na arquitetura;
- b) Usar construção **with ... select** na arquitetura;
- c) Usar um processo e a construção **case** dentro do processo.
- d) Usar um processo e a construção **if...elsif...else...end if** dentro do processo.
- e) Usar uma constante (ver aula 2 teórica).
- f) Fazer simulação comportamental de qualquer um dos projetos dos pontos a)...e).

Exercício 2.2. Descreva em VHDL estrutural (ou misto) e faça ligação entre saídas de contador (ver exercício 1.6 da aula anterior) e entradas do decodificador (ver exercício 2.1). Use para o contador a frequência de 1 Hz (implemente um divisor de frequência 100 MHz → 1 Hz).

Exercício 2.3. Implemente um registo de deslocamento de 16 bits. Entradas são a) relógio com frequência 1 Hz (mais ou menos); b) *clock enable* para permitir relógio na entrada (botão btnL); c) *reset* para fazer reset (i.e. todas as saídas são iguais a zero) (botão btnC); d) SW_0, \dots, SW_{15} entradas paralelas; e) *set* para gravar no registo entradas paralelas (botão btnR); f) *left* para fazer *shift* dois bits à esquerda com frequência de relógio (botão

btnU='0'); g) *right* para fazer *shift* dois bits à direita com frequência de relógio (botão btnU='1'). Mostrar o conteúdo do registo nos LEDs (16 LEDs totalmente). Fazer simulação comportamental do circuito.

Exercício 2.4. Implemente um circuito que permite encontrar o valor máximo e mínimo de 4 vetores binários (de 4 bits cada). Use 16 interruptores para 4 vetores binários.

Exercício 2.5. Verifique operadores VHDL **mod**, **abs**, e **rem** (use operandos de interruptores e mostre os resultados em LEDs/displays). **Sugestão:** altere o projeto da aula teórica 2 que usa operadores aritméticos. Considere duas possibilidades: a) uso de um processo combinatório; b) sem processos.