



## **Laboratório 2** **- ULA e FPULA -**

### **Objetivos:**

- Introduzir ao aluno a Linguagem de Descrição de Hardware Verilog;
- Familiarizar o aluno com a plataforma de desenvolvimento FPGA DE1-SoC e o software QUARTUS Prime da Intel;
- Desenvolver a capacidade de análise, síntese e caracterização de sistemas digitais usando HDL;

Dica: Para envio de um projeto do Quartus:

- i) Project/ Archive Project/Advanced e selecione também Programming output files.
- ii) Envie apenas os arquivos .qar gerados e o arquivo .pdf do relatório em um arquivo Grupox\_Lab2.zip

### **1) (0.0) Implementação de um driver para display de 7 segmentos Síncrono e Assíncrono**

(0.0) Faça o download do arquivo lab2.zip do Moodle, deszipa em um diretório do desktop que **NÃO** contenha espaço nem caracteres especiais no nome. O arquivo Tutorial\_Quartus\_Primev11.pdf contém instruções sobre como criar um projeto, usar em simulações e como sintetizar na DE1-SoC.

Abra o Quartus Prime e Project/Restore Archived Project e escolha o arquivo Decoder7.qar

1.1 (0.0) Para a versão assíncrona do decoder7.v, defina-o como o toplevel, verifique se todos os pinos (menos o Clock) estão colocados como pinos virtuais (Assignments/Assignment Editor) e compile:

- a) Indique os requerimentos físicos da implementação: i) número de elementos lógicos (ALMs), ii) número de registradores, iii) quantidade de bits de memória e iv) número de blocos DSP usados;
  - b) Veja o diagrama esquemático do circuito gerado pelo Tools/Netlist Viewers/RTL viewer;
  - c) Com o arquivo em forma de onda decoder7.vwf (University Program) realize a simulação funcional (verifique sempre se o Simulation settings estão com os parâmetros default e retire –novopt). Tire um print screen da tela para comparação;
  - d) Usando o Tool/Run Simulation Tool/RTL Simulation realize a simulação do decoder7\_tb;
  - e) Usando o TimeQuest (Tools/Timing Analyzer), gere a Timing Netlist e em seguida relatório Report Datasheet e indique os requerimentos temporais: i) o caminho de maior atraso, ii) maior tempo de atraso tpd.
- ⌘ Defina o arquivo TopDE.v como o toplevel, compile, usando o Tools/Programmer sintetize na DE1-SoC e filme o funcionamento para todos os valores de entrada.

1.2 (0.0) Para a versão síncrona do decoder7.v, defina-o como toplevel e compile:

- a) Indique os requerimentos físicos da implementação: i) número de elementos lógicos (ALMs), ii) número de registradores, iii) quantidade de bits de memória e iv) número de blocos DSP usados;
  - b) Veja o diagrama esquemático do circuito gerado pelo Tools/Netlist Viewers/RTL viewer;
  - c) Com o arquivo de forma de onda decoder7.vwf, realize a simulação funcional. Qual a diferença para o item 1.1)c)?
  - d) Usando o Tool/Run Simulation Tool/RTL Simulation realize a simulação do decoder7\_tb. Qual a diferença para o item 1.1)d)?
  - e) Usando o TimeQuest, defina o clock de 50MHz, indique os requerimentos temporais: i) tempos th, tco, tsu e slacks, ii) máxima frequência de clock utilizável, iii) se há algum requerimento não atendido. Aumente a frequência para 1000MHz e repita os itens i), ii) e iii).
- ⌘ Defina o arquivo TopDE.v como o toplevel, sintetize na DE1-SoC e filme o funcionamento para todos os valores de entrada. Notou alguma diferença? Porque?

## 2) (5.5) Unidade Lógico Aritmética de Inteiros:

Abra o Quartus Prime e Project/Restore Archived Project e escolha o arquivo ULA.qar

- a) (1.0) Para a ULA de inteiros fornecida, analise sua descrição Verilog e descreva suas funções e escreva a tabela de seus códigos para cada operação. Defina o arquivo ALU.v como toplevel, compile e visualize o circuito sintetizado com o Tools/Netlist Viewers/RTL viewer.
- a) (1.0) Modifique os valores do arquivo de forma de onda ULA.vwf e verifique cada operação implementada. Confira com o RTL Simulation. Dica: Escolha valores de entrada que sejam representativos (comuns) e valores que possam gerar resultados singulares (overflow, divisão por zero).
- b) (1.0) Indique os requisitos físicos da implementação da ULA total e para cada operação separadamente: i) Número de Elementos Lógicos (ALMs), ii) Número de Registradores e iii) Quantidade de bits de memória e iv) Número de blocos DSP usados. v) Somando todos os ALMs do item i) compare com o número de ALMs da ULA total e explique a diferença. vi) Indique quais são e analise o impacto das funções com maiores circuitos no tamanho da ULA. Dica: Defina manualmente o sinal de controle iControl no arquivo ULA.v, pois o Quartus otimiza o projeto retirando as partes não utilizadas.
- c) (1.0) Usando o TimeQuest, indique os requerimentos temporais para a ULA total e para cada operação separadamente: i) o caminho de maior atraso, ii) maior tempo de atraso tpd. iii) Indique quais são e analise o impacto das funções mais demoradas no tpd.
- d) (1.0) Defina o arquivo TopDE.v como o toplevel, sintetize na DE1-SoC e filme comprovando o correto funcionamento de acordo com os valores testados no item b).
- e) (0.5) Verifique o quanto seu grupo consegue otimizar esta ULA em termos de requerimentos físicos e/ou temporais (sem retirar nenhuma funcionalidade!).

## 3) (5.5) Unidade Aritmética de Ponto Flutuante:

Abra o Quartus Prime e Project/Restore Archived Project e escolha o arquivo FPALU.qar

- b) (1.0) Para a FPULA fornecida, analise sua descrição Verilog e descreva suas funções e escreva a tabela de seus códigos para cada operação. Defina o arquivo FPALU.v como toplevel, compile e visualize o circuito sintetizado com o Tools/Netlist Viewers/RTL viewer.
- c) (1.0) Modifique os valores do arquivo de forma de onda FPULA.vwf e verifique cada operação implementada. Confira com o RTL Simulation. Dica: Escolha valores de entrada que sejam representativos (comuns) e também valores que gerem resultados singulares (divisão por zero, overflow, underflow, NaN).
- d) (1.0) Indique os requisitos físicos da implementação da FPULA total e para cada operação separadamente: i) Número de Elementos Lógicos (ALMs), ii) Número de Registradores e iii) Quantidade de bits de memória e iv) Número de blocos DSP usados, v) Somando todos os ALMs do item i) compare com o número de ALMs da ULA total e explique a diferença. vi) Indique quais são e analise o impacto das funções com maiores circuitos no tamanho da ULA. Dica: Defina manualmente o icontrol pois o Quartus otimiza o projeto retirando os elementos não utilizados.
- e) (1.0) Usando o TimeQuest, defina um clock de 50MHz, indique os requerimentos temporais para a ULA total e para cada operação separadamente: i) número de ciclos necessários à execução de cada operação, ii) tempos th, tco, tsu e slacks, iii) máxima frequência de clock utilizável, e iv) se há algum requerimento não atendido. Indique quais são e analise o impacto das funções mais demoradas na máxima frequência utilizável. Aumente a frequência para 200MHz e repita os itens i), ii), iii) e iv). Analise os resultados obtidos.
- f) (1.0) Defina o arquivo TopDE.v como o toplevel, sintetize na DE1-SoC e filme comprovando o correto funcionamento de acordo com os valores de entrada testados no item b).
- g) (0.5) Verifique o quanto seu grupo consegue otimizar esta FPULA em termos de requerimentos físicos e/ou temporais (sem retirar nenhuma funcionalidade!).