

# Laboratório 2

## - ULA e FPULA –

### GRUPO 6

Dayanne Fernandes da Cunha, 13/0107191

Lucas Mafra Chagas, 12/0126443

Marcelo Giordano Martins Costa de Oliveira, 12/0037301

Lucas Junior Ribas, 16/0052289

Caio Nunes de Alencar Osório, 16/0115132

Diego Vaz Fernandes, 16/0117925

<sup>1</sup>Dep. Ciência da Computação – Universidade de Brasília (UnB)  
CiC 116394 - OAC - Turma A

## 1. Objetivos

- Introduzir ao aluno a Linguagem de Descrição de *Hardware Verilog*;
- Familiarizar o aluno com a plataforma de desenvolvimento *FPGA DE2* da *Altera* e o *software QUARTUS-II*;
- Desenvolver a capacidade de análise e síntese de sistemas digitais usando *HDL*.

## 2. Ferramentas

- FPGA DE2 da Altera
- QUARTUS-II
- Verilog
- HDL

## 3. Exercícios

Todos os códigos escritos neste laboratório podem ser encontrados no repositório <https://github.com/Dayof/OAC172> do *GitHub*.

### 3.1. Exercício 1. Implementação de um *driver* para *display* de 7 segmentos

Conforme descrito no arquivo *QuartusIiv3.txt* e *Set.txt*, um novo projeto foi criado no diretório *Lab2*, denominado *Display*.

Para as versões síncrona e assíncrona foram geradas as simulações temporais (Figura 1 e Figura 3) e funcionais (Figura 2 e Figura 4).

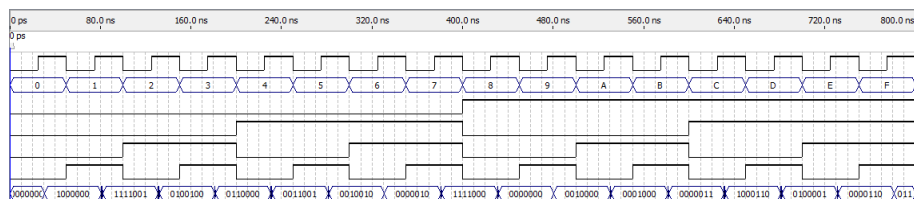


Figure 1. Simulação síncrona temporal do *decoder7*.

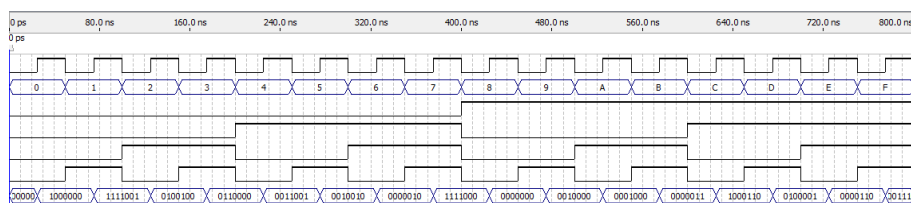


Figure 2. Simulação síncrona funcional do *decoder7*.

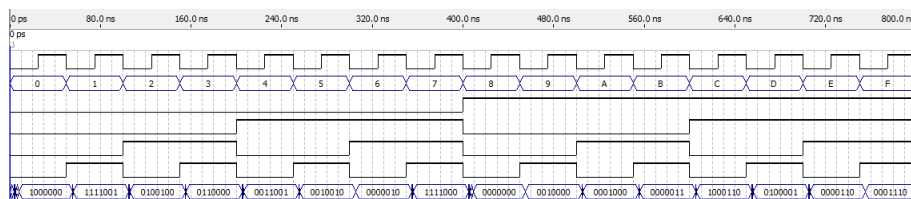


Figure 3. Simulação assíncrona temporal do *decoder7*.

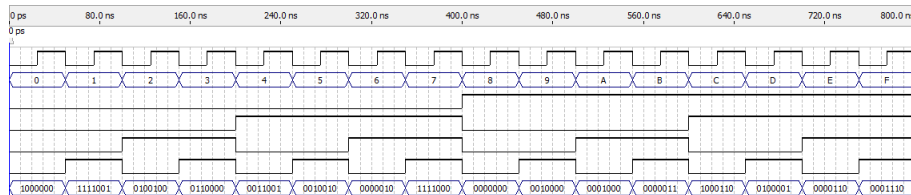


Figure 4. Simulação assíncrona funcional do *decoder7*.

Os requisitos físicos do *decoder7* do *driver* para *display* de 7 segmentos foram analisados nas versões síncrona e assíncrona como é possível ver na Tabela 1.

	Elementos lógicos	Maior atraso (ns)	Frequência máxima de operação (MHz)
Síncrono	7	?	?
Assíncrono	7	9.761	?

Table 1. Requisitos físicos do *display* de 7 segmentos assíncrono e síncrono.

O arquivo de interface *TopDE.v* foi incluso no projeto, sintetizado e testado como é mostrado no link <https://youtu.be/wGKjze5PkcU>.

### 3.2. Exercício 2. Unidade Lógica Aritmética de Inteiros

#### 3.2.1. ULA MIPS32

#### 3.2.2. Operações

#### 3.2.3. Requisitos físicos

	Elementos lógicos	Tempo de atraso (ns)	Frequência máxima de <i>clock</i> utilizável (MHz)
ULA	?	?	?
OPAND	?	?	?
OPOR	?	?	?
OPADD	?	?	?
OPMFHI	?	?	?
OPSL	?	?	?
OPMFLO	?	?	?
OPSUB	?	?	?
OPSLT	?	?	?
OPSGT	?	?	?
OPSR	?	?	?
OPSR	?	?	?
OPXOR	?	?	?
OPSLTU	?	?	?
OPNOR	?	?	?
OPLUI	?	?	?
OPSLLV	?	?	?
OPSRV	?	?	?
OPSRV	?	?	?
OPMULT	?	?	?
OPDIV	?	?	?
OPDEBUG	?	?	?
OPMULTU	?	?	?
OPDIVU	?	?	?
OPMTHI	?	?	?
OPMTLO	?	?	?
OPMADD	?	?	?
OPMADDU	?	?	?
OPMSUB	?	?	?
OPMSUBU	?	?	?

Table 2. Requisitos físicos da *ULA* total e de cada operação.

### 3.2.4. Funcionamento

O projeto da *ULA* de inteiros foi sintetizado utilizando a interface *TopDE.v* na placa *DE2-70* e seu funcionamento pode ser visto através do *link* ?.

## 3.3. Exercício 3. Unidade Aritmética de Ponto Flutuante

### 3.3.1. FPULA MIPS

### 3.3.2. Operações

### 3.3.3. Requisitos físicos

	Elementos lógicos	Número de ciclos mínimo da operação (ns)	Frequência máxima de <i>clock</i> utilizável (MHz)
<b>FPULA</b>	?	?	?
<b>OPADDS</b>	?	?	?
<b>OPSUBS</b>	?	?	?
<b>OPMULS</b>	?	?	?
<b>OPDIVS</b>	?	?	?
<b>OPSQRT</b>	?	?	?
<b>OPABS</b>	?	?	?
<b>OPNEG</b>	?	?	?
<b>OPCEQ</b>	?	?	?
<b>OPCLT</b>	?	?	?
<b>OPCLE</b>	?	?	?
<b>OPCVTSW</b>	?	?	?
<b>OPCVTWS</b>	?	?	?

Table 3. Requisitos físicos da *FPULA* total e de cada operação.

### 3.3.4. Funcionamento

O projeto da *ULA* de ponto flutuante foi sintetizado utilizando a interface *TopDE.v* na placa *DE2-70* e seu funcionamento pode ser visto através do *link* ?.