

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

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

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*.

Ferramentas

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

Exercícios

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

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).

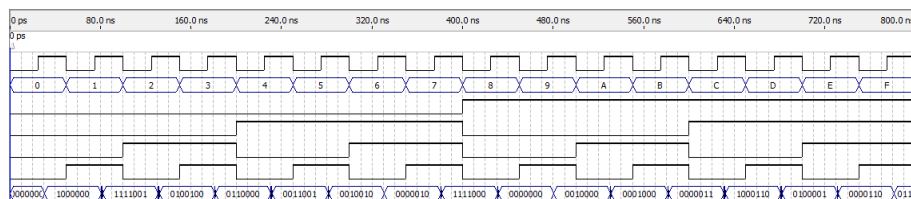


Figure 1. Simulação síncrona temporal do *decoder7*.

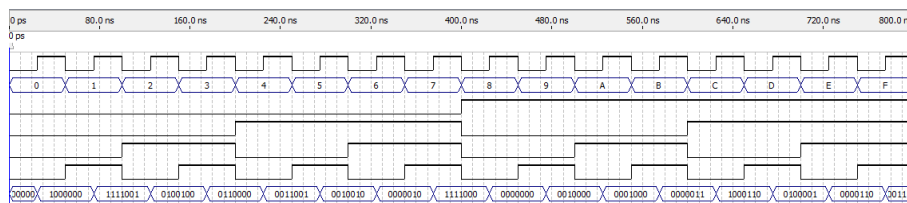


Figure 2. Simulação síncrona funcional do *decoder7*.

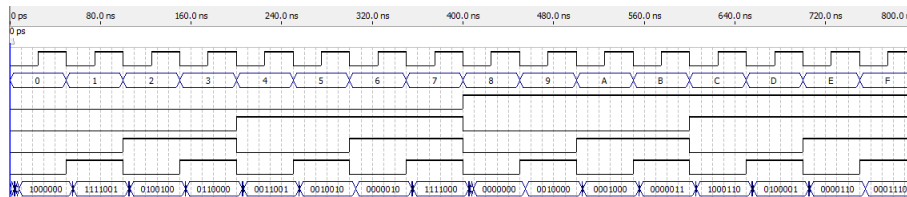


Figure 3. Simulação assíncrona temporal do *decoder7*.

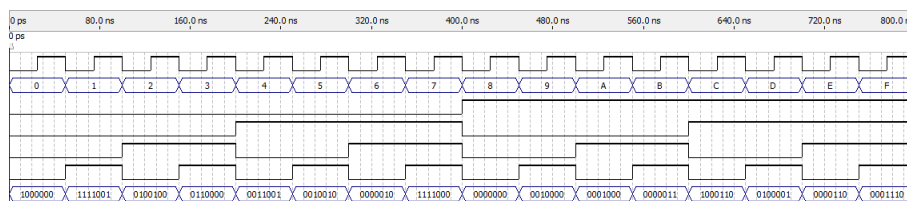


Figure 4. Simulação assíncrona funcional do *decoder7*.

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

Exercício 2. Unidade Lógica Aritmética de Inteiros

Operações

Requisitos físicos

Para a ULA de inteiros foi levantado os requisitos físicos de cada operação e da ULA total como podemos ver na Tabela 1 e Tabela 2. Todos estes dados foram encontrados utilizando o seguinte procedimento:

- Foi aberto o projeto da *ULA* no *Quartus II 64-Bit*;
- No arquivo *ALU.v* para testar a ULA completa foi preciso comentar as linhas `"wire [4:0] iControlSignal"` e `"assign iControlSignal=OPMSUB"` e descomentar a linha `"input [4:0] iControlSignal"`. No arquivo *ALU.v* para testar cada operação foi preciso descomentar as linhas `"wire [4:0] iControlSignal"` e `"assign iControlSignal=OPMSUB"` e comentar a linha `"input [4:0] iControlSignal"`. A troca de operação avaliada foi feita substituindo o nome da operação na variável *iControlSignal* (e.g. `assign iControlSignal=OPSLI`);
- Ao trocar a operação desejada foi compilado o projeto;

- Com a nova aba (*Compilation Report - ULA*) aberta, no menu *Flow Summary* foi possível achar informações da quantidade total de elementos lógicos usado naquela operação;
- No menu *TimeQuest Timing Analyzer > Multicorner Datasheet Report Summary* foram encontrados valores dos maiores / menores tempos de atraso para concluir a operação. Estes tempos são medidos desde o ato de inserir o dado na entrada (*iA* e/ou *iB*) e resultar em algo na saída (*oALUresult*). Alguns resultados assíncronos eram aparentes na aba *RR* (medição ao subir a borda inicial até a subida da borda final), outros na *RF* (medição ao subir a borda inicial até a descida da borda final) [Altera]. Para operações síncronas era possível captar os resultados na aba *Rise*;
 - Para operações puramente assíncronas o maior tempo foi encontrado no menu *Propagation Delay*. Para operações também síncronas tiveram estes dados aparentes no menu *Clock to Output Times*;
 - Para operações puramente assíncronas o menor tempo foi encontrado no menu *Minimum Propagation Delay*. Para operações também síncronas tiveram estes dados aparentes no menu *Minimum Clock to Output Times*.
- A frequência máxima de *clock* utilizável foi gerada a partir do cálculo $F_{MAX} = 1/T$, sendo T o maior tempo de atraso da operação. Esse T tem que ser o pior caso de tempo ocorrido pois precisa ser suficiente para concluir toda a operação em qualquer caso.

	Elementos lógicos	Menor atraso (ns)	Maior atraso (ns)	Frequência máxima de <i>clock</i> utilizável (MHz)
ULA	6686	4,788	26,648	37,526
OPAND	43	5,495	9,810	101,937
OPOR	43	5,490	9,811	101,926
OPADD	44	5,081	14,134	70,751
OPMFHI	0	0	0	0
OPSL	170	6,000	15,048	66,454
OPMFLO	0	0	0	0
OPSUB	44	5,119	13,909	71,896
OPSLT	32	5,341	13,470	74,239
OPSGT	32	5,341	13,470	74,239
OP SRL	170	6,065	16,137	61,969
OPSRA	174	4,908	15,736	63,549
OPXOR	43	4,802	8,511	117,495
OPSLTU	32	5,341	13,470	74,239
OPNOR	43	4,822	9,811	101,926
OPLUI	5	4,546	8,330	120,048
OPSLLV	170	5,397	15,048	66,454
OPSRV	174	4,908	15,736	63,549
OP SRLV	170	5,445	16,137	61,969
OPMULT	53	3,914	8,885	112,549
OPDIV	1266	4,051	9,446	105,865
OPDEBUG	11	4,549	8,255	121,139

Table 1. Requisitos físicos da ULA total e de cada operação. Informações das operações assíncronas.

	Elementos lógicos	Menor atraso (ns)	Maior atraso (ns)	Frequência máxima de <i>clock</i> utilizável (MHz)
ULA	6686	6,804	14,672	68,157
OPMULT	53	?	?	?
OPDIV	1266	?	?	?
OPMULTU	?	?	?	?
OPDIVU	?	?	?	?
OPMTHI	?	?	?	?
OPMTLO	?	?	?	?
OPMADD	?	?	?	?
OPMADDU	?	?	?	?
OPMSUB	?	?	?	?
OPMSUBU	?	?	?	?

Table 2. Requisitos físicos da ULA total e de cada operação. Informações das operações síncronas.

Funcionamento

O projeto da *ULA* de inteiros foi sintetizado utilizando a interface *TopDE.v* na placa *DE2-70*.

As funções testadas foram:

Operação	Vídeo	Operação	Vídeo
AND	AND	OR	OR
ADD	ADD	MFHI	MFHI
SLL	SLL	MFLO	MFLO
SUB	SUB	SLT	SLT
SRL	SRL	SRA	SRA
XOR	XOR	SLTU	SLTU
NOR	NOR	MULT	MULT
DIV	DIV	LUI	LUI
SLLV	SLLV	SRAV	SRAV
SRLV	SRLV	MULTU	MULTU
DIVU	DIVU	MTHI	MTHI
MLTO	MLTO	SGT	SGT
MADD	MADD	MADDU	MADDU
MSUB	MSUB	MSUBU	MSUBU

Table 3. Funcionamento das operações da ULA.

Exercício 3. Unidade Aritmética de Ponto Flutuante

Operações

Requisitos físicos

Para a *ULA* de ponto flutuante foi levantado os requisitos físicos de cada operação e da *FPULA* total como podemos ver na Tabela 4. Todos estes dados foram encontrados utilizando o seguinte procedimento:

- Foi aberto o projeto da *FPULA* no *Quartus II 64-Bit*;
- No arquivo *FPALU.v* para testar a *FPULA* completa foi preciso comentar as linhas `"wire [3:0] icontrol"` e `"assign icontrol=OPSQRT"` e descomentar a linha `"input [3:0] icontrol"`. No arquivo *FPALU.v* para testar cada operação foi preciso descomentar as linhas `"wire [3:0] icontrol"` e `"assign icontrol=OPSQRT"` e comentar a linha `"input [3:0] icontrol"`. A troca de operação avaliada foi feita substituindo o nome da operação na variável *icontrol* (e.g. `assign icontrol=OPSQRT`);
- Ao trocar a operação desejada foi compilado o projeto;
- Com a nova aba (*Compilation Report - FPULA*) aberta, no menu *Flow Summary* foi possível achar informações da quantidade total de elementos lógicos usado naquela operação;
- No menu *TimeQuest Timing Analyzer > Multicorner Datasheet Report Summary* ; *Minimum Propagation Delay* foram encontrados os números de ciclos mínimos

da operação avaliada. Estes tempos são medidos desde o ato de inserir o dado na entrada (*idataa* e/ou *idatab*) e resultar em algo na saída (*oCompResult*). Os resultados eram aparentes na aba *RR* (medição ao subir a borda inicial até a subida da borda final), outros na *RF* (medição ao subir a borda inicial até a descida da borda final) [Altera];

- A frequência máxima de *clock* utilizável foi gerada a partir do cálculo $F_{MAX} = 1/T$, sendo T o número de ciclos mínimo da operação.

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

Table 4. Requisitos físicos da *FPULA* total e de cada operação.

Funcionamento

O projeto da *ULA* de pontos flutuantes foi sintetizado utilizando a interface *TopDE.v* na placa *DE2-70*.

As funções testadas foram:

Operação	Vídeo	Operação	Vídeo
ADDS	ADDS	SUBS	SUBS
MULS	MULS	DIVS	DIVS
SQRT	SQRT	ABS	ABS
NEG	NEG	CEQ	CEQ
CLT	CLT	CLE	CLE
CVTSW	CVTSW	CVTWS	CVTWS

Table 5. Funcionamento das operações da *ULA*.

References

- [Altera] Altera. Multicorner timing. http://quartushelp.altera.com/15.0/mergedProjects/report/rpt/rpt_file_multicorner_timing.htm. [Online; acessado 4-Outubro-2017].