

Organização de Computadores I
Documentação do Trabalho Prático

Christian Vieira
João Pedro Pinto Coelho
Marcus Oliveira
Paula Viriato

1 Atividade 1

Todas as operações que a ULA realiza são realizadas pra cada entrada recebida, porém é selecionado o resultado que a variável *op* escolhe.

A ULA é capaz de executar as operações de soma, subtração, *and* lógico, e outras. No entanto, nesse caso serão executadas apenas soma, *and* e *set on less than*.

2 Atividade 2

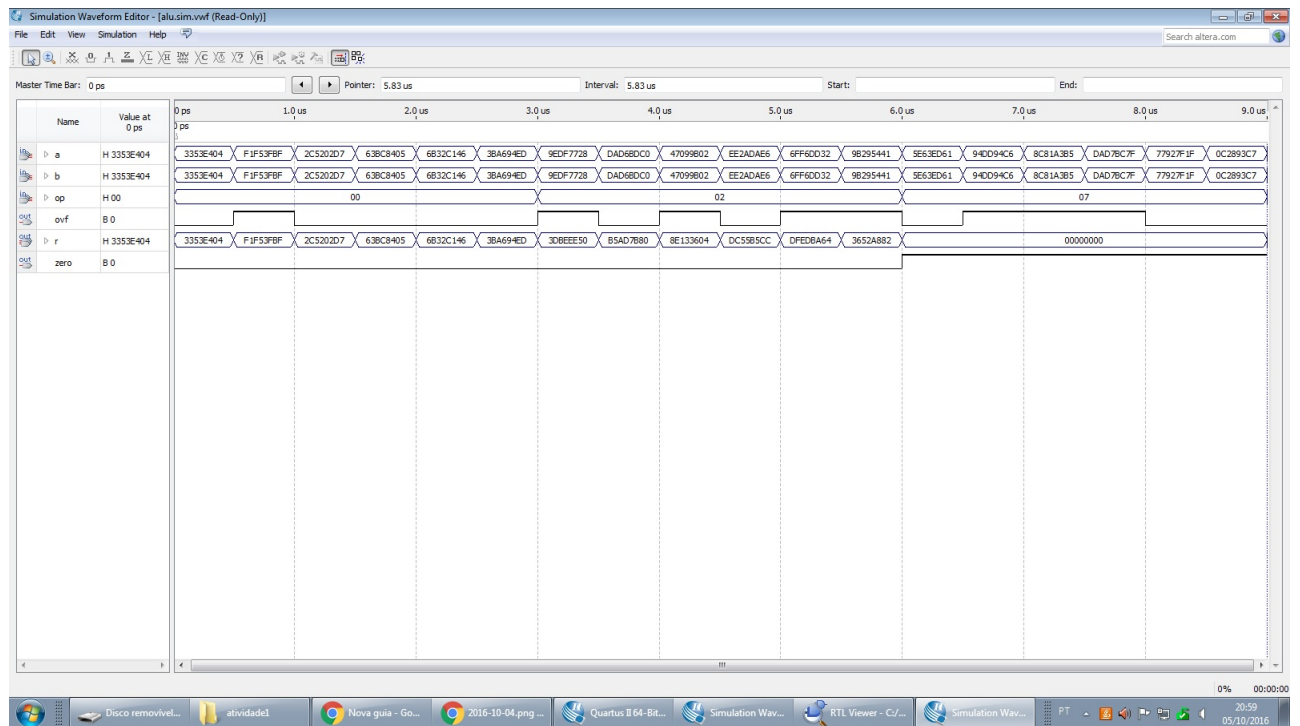
A simulação da ALU foi formulada da seguinte maneira:

Para cada operação implementada no exercício anterior, foram testadas as quatro possíveis entradas dos inputs A e B, conforme a tabela a seguir:

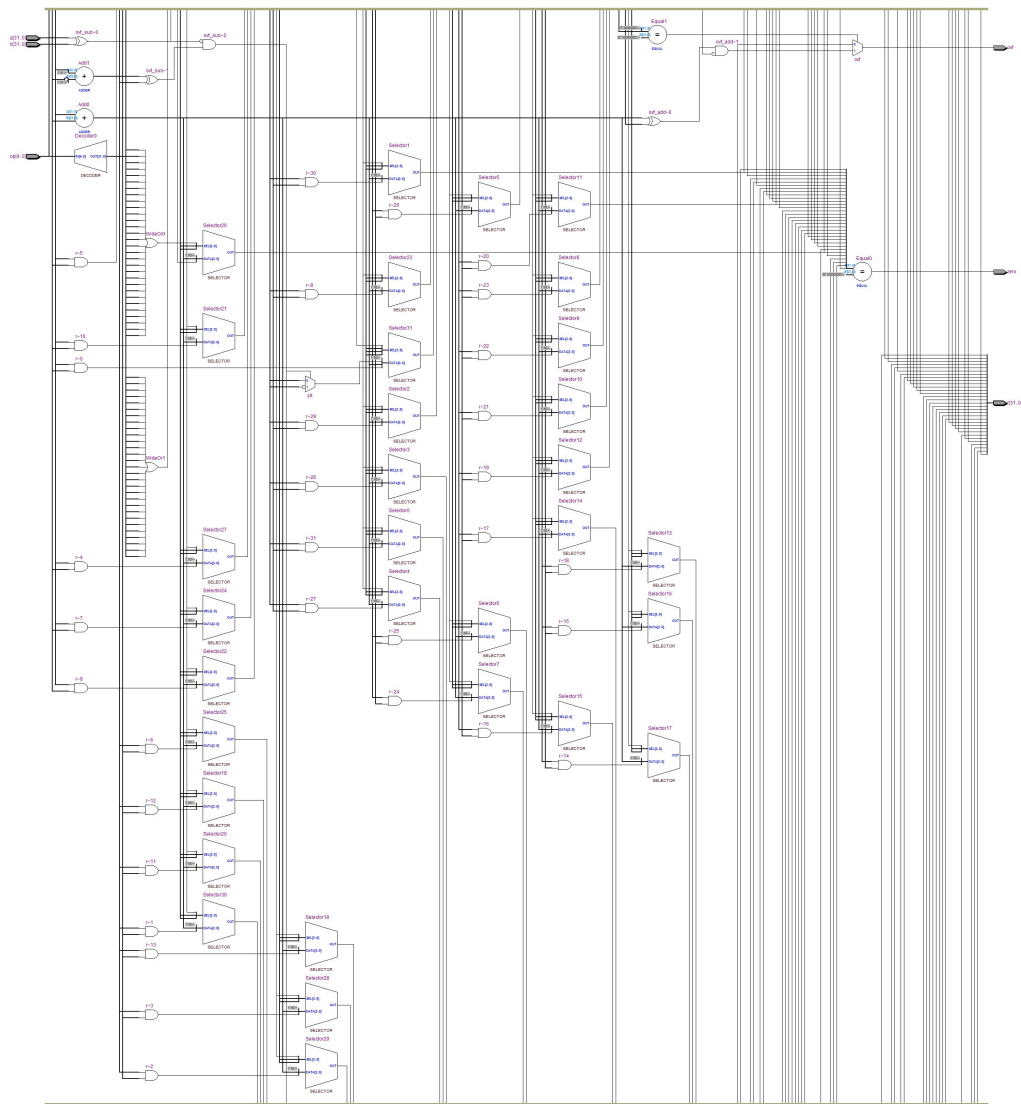
Tabela 1 My caption

	Input		Output		
	a	b	ovf	r	zero
And	3353E404	3353E404	0	3353E404	0
	F1F53FBF	F1F53FBF	1	F1F53FBF	0
	2C5202D7	2C5202D7	0	2C5202D7	0
	63BC8405	63BC8405	0	63BC8405	0
	6B32C146	6B32C146	0	6B32C146	0
	3BA694ED	3BA694ED	0	3BA694ED	0
Soma	9EDF7728	9EDF7728	1	3DBEEE50	0
	DAD6BDC0	DAD6BDC0	0	B5AD7B80	0
	47099B02	47099B02	1	8E133604	0
	EE2ADAE6	EE2ADAE6	0	DC55B5CC	0
	6FF6DD32	6FF6DD32	1	DFEDBA64	0
	9B295441	9B295441	1	3652A882	0
Set on less than	5E63ED61	5E63ED61	0	00000000	1
	94DD94C6	94DD94C6	1	00000000	1
	8C81A3B5	8C81A3B5	1	00000000	1
	DAD7BC7F	DAD7BC7F	1	00000000	1
	77927F1F	77927F1F	0	00000000	1
	0C2893C7	0C2893C7	0	00000000	1

Notamos que os dados esperados correspondem com os dados obtidos através da simulação, conforme as formas de onda a seguir:



Desta forma, concluímos que a ALU funciona corretamente.



Através do RTL View, presente no Quartus II, fomos capazes de gerar o circuito lógico da ULA, conforme a imagem acima.

3 Atividade 3

Foram acrescentadas à ULA implementada na atividade 1 as funções *or*, *nor*, *sub* e *lui*. Como é demonstrado na seção seguinte, verificamos o funcionamento correto das instruções acrescentadas.

4 Atividade 4

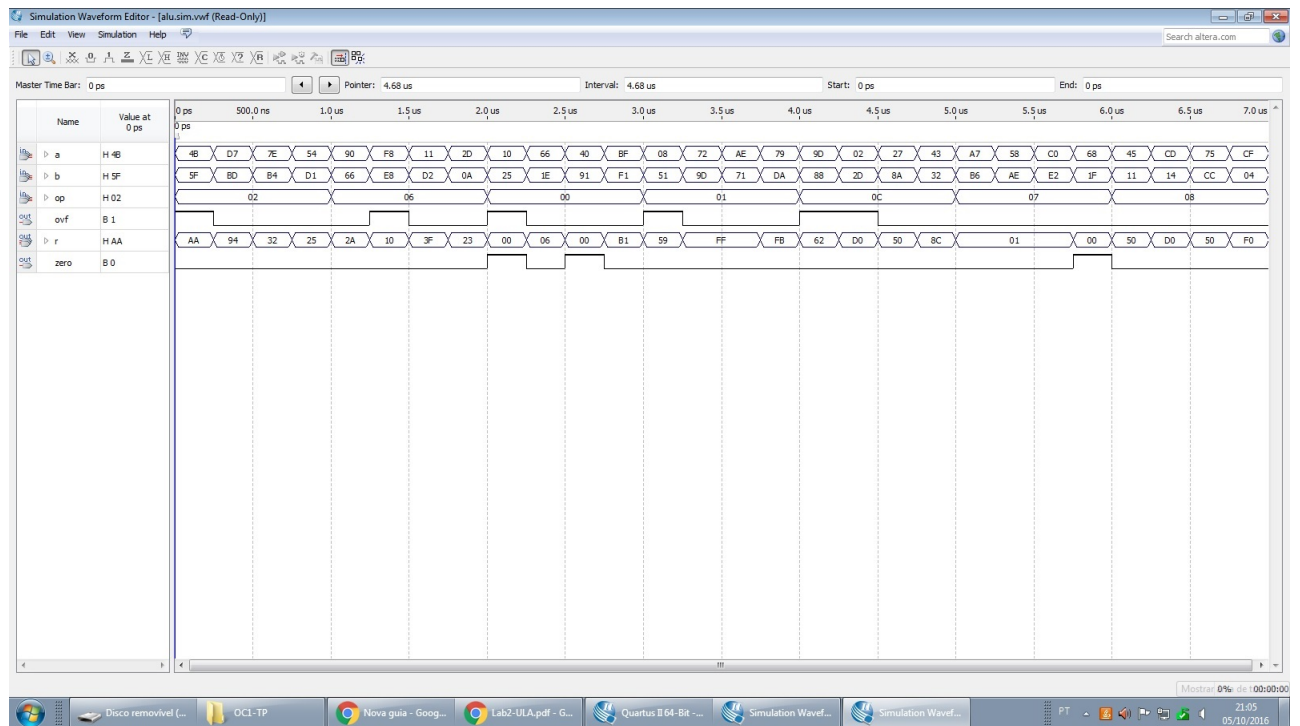
A simulação da ALU foi executada da mesma forma que a atividade 2:

Para cada operação implementada nos exercícios 1 e 3, foram testadas as quatro possíveis entradas dos inputs A e B, conforme a tabela a seguir:

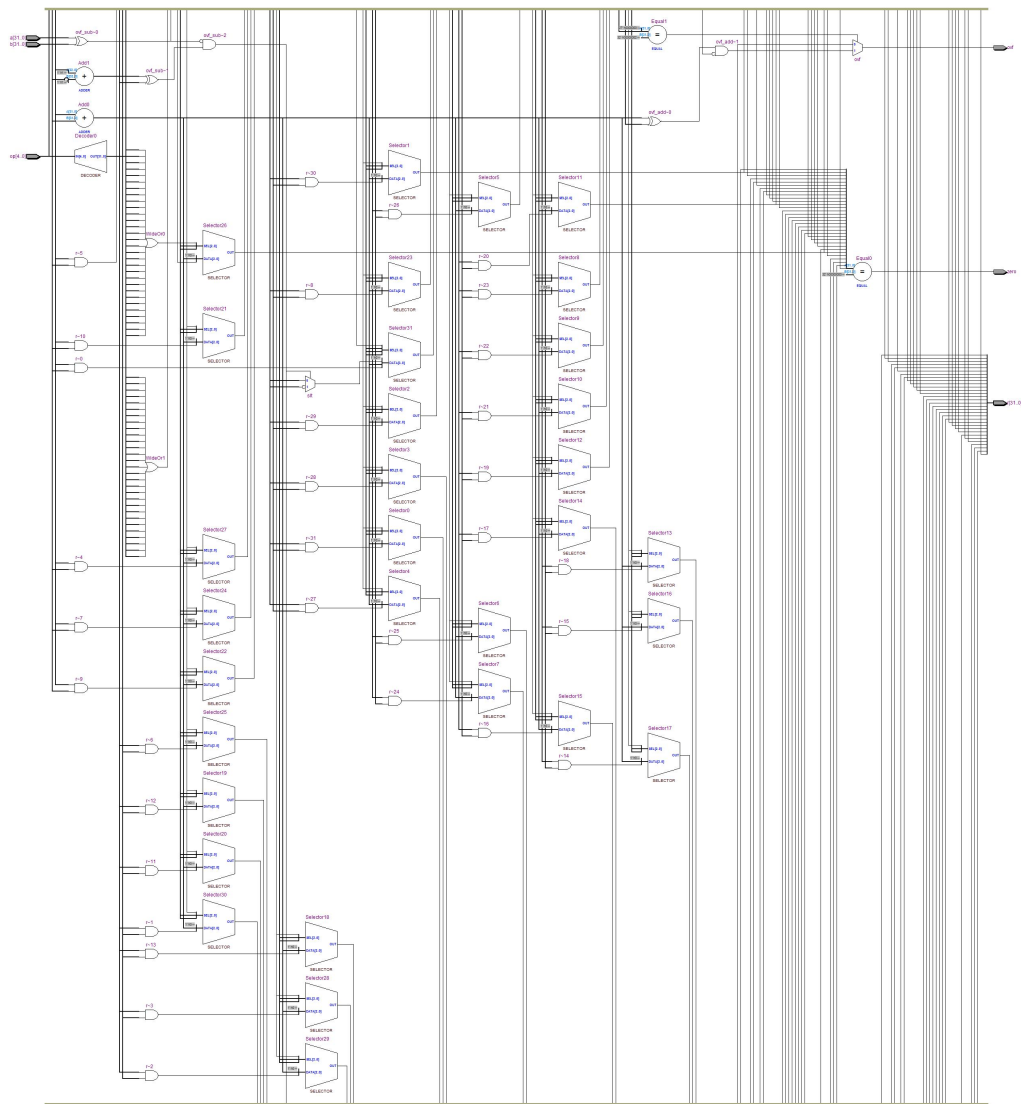
Tabela 2 My caption

	Input			Output		
	a (Hexadecimal)	b (Hexadecimal)	op (Hexadecimal)	ovf	r (Hexadecimal)	zero
Soma	4B	5F	02	1	AA	0
	D7	BD	02	0	94	0
	7E	B4	02	0	32	0
	54	D1	02	0	25	0
Subtração	90	66	06	0	2A	0
	F8	E8	06	1	10	0
	11	D2	06	0	3F	0
	2D	0A	06	0	23	0
And	10	25	00	1	00	1
	66	1E	00	0	06	0
	40	91	00	0	00	1
	BF	F1	00	0	B1	0
Or	08	51	01	1	59	0
	72	9D	01	0	FF	0
	AE	71	01	0	FF	0
	79	DA	01	0	FB	0
Nor	9D	88	0C	1	62	0
	02	2D	0C	1	D0	0
	27	8A	0C	0	50	0
	43	32	0C	0	8C	0
Set on Less	A7	B6	07	0	01	0
	58	AE	07	0	01	0
	C0	E2	07	0	01	0
	68	1F	07	0	00	1
Load Upper immediate	45	11	08	0	50	0
	CD	14	08	0	D0	0
	75	CC	08	0	50	0
	CF	04	08	0	F0	0

Notamos que os dados esperados correspondem com os dados obtidos através da simulação, conforme as formas de onda a seguir:



Desta forma, concluímos que a ALU funciona corretamente.



Através do RTL View, presente no Quartus II, fomos capazes de gerar o circuito lógico da ULA, conforme a imagem acima.

5 Conclusão

Para o desenvolvimento da ALU foram seguidas as regras e programas utilizados previamente em sala de aula. A ALU está funcionando perfeitamente conforme os testes executados.