

## LABORATÓRIO DE CIRCUITOS DIGITAIS

7º Experimento: **Unidade Lógico-Aritmética em SystemVerilog**

UFERSA – Campus Pau dos Ferros – DETEC

Prof.: Pedro Thiago Valério de Souza



**OBJETIVO** Anderson Carlos da Silva, Marcos Marília, Fomaisa Andreide

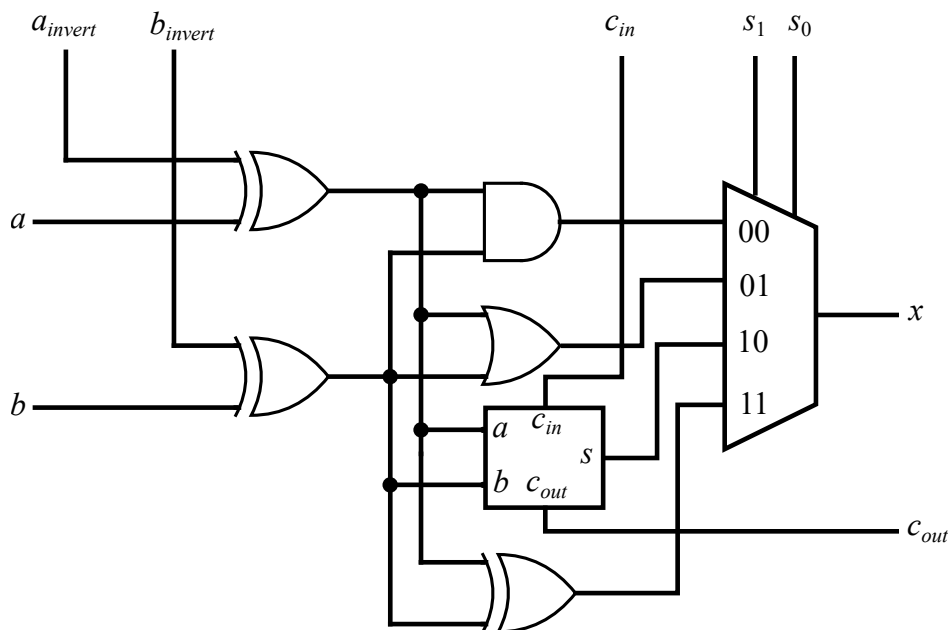
Descrever uma unidade lógico-aritmética em SystemVerilog.

### COMPONENTES

- FPGA Cyclone DE2-115.

### PROCEDIMENTO PRÁTICO

A Figura abaixo apresenta uma unidade lógico-aritmética elementar de apenas um 1 bit. Ela é composta por um somador completo, um multiplexador 4×1 e um conjunto de portas lógicas adicionais.



- Sabendo que as expressões lógicas de um somador completo são:

$$s = a \oplus b \oplus c_{in}$$

$$c_{out} = ab + ac_{in} + bc_{in}$$

sendo  $a$  e  $b$  os dois bits a serem somados,  $c_{in}$  a entrada de carry-in,  $s$  a saída de soma e  $c_{out}$  a saída de carry-out, implemente o bloco de somador completo em SystemVerilog.

- Sabendo que a expressão lógica de saída de um multiplexador 4×1 é dada por:

$$x = \overline{s_1} \overline{s_0} i_0 + \overline{s_1} s_0 i_1 + s_1 \overline{s_0} i_2 + s_1 s_0 i_3$$

sendo  $s_1$  e  $s_0$  os bits de seleção e  $i_0$ ,  $i_1$ ,  $i_2$  e  $i_3$  as entradas de dados, implemente o bloco de multiplexador em SystemVerilog.

- Implemente a unidade lógico-aritmética proposta em SystemVerilog, utilizando os blocos projetados nos itens (1) e (2), em conjunto com as portas lógicas necessárias. *Importante:* O nome do módulo deve ser alu1b.

4. Utilizando o ModelSim, realize a simulação utilizando o arquivo *testbench* entitulado *tb\_alu1b sv* e anote os resultados encontrados na tabela abaixo.

<i>a</i>	<i>b</i>	<i>a<sub>invert</sub></i>	<i>b<sub>invert</sub></i>	<i>c<sub>in</sub></i>	<i>s<sub>1</sub></i>	<i>s<sub>0</sub></i>	<i>x</i>	<i>c<sub>out</sub></i>
1	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	1	1
1	0	0	0	0	0	1	1	0
0	0	0	0	0	0	1	0	0
1	0	0	0	0	1	0	1	0
1	1	0	0	0	1	0	0	1
1	1	0	0	1	1	0	1	1
1	1	0	1	1	1	0	0	1
1	0	0	0	0	1	1	1	0
1	1	0	0	0	1	1	0	1

5. Implemente o projeto do item (3) na FPGA DE2-115 (referência: EP4CE115F29C7). Utilize, como *pin planner* as configurações apresentadas na tabela abaixo:

Entrada/Saída	Recurso na FPGA	No. Pino da FPGA
✓ <i>a</i>	SW[0]	PIN_AB28
✓ <i>b</i>	SW[1]	PIN_AC28
✓ <i>a<sub>invert</sub></i>	SW[2]	PIN_AC27
✓ <i>b<sub>invert</sub></i>	SW[3]	PIN_AD27
✓ <i>s<sub>0</sub></i>	SW[5]	PIN_AC26
✓ <i>s<sub>1</sub></i>	SW[6]	PIN_AD26
✓ <i>c<sub>in</sub></i>	SW[7]	PIN_AB26
✓ <i>x</i>	LEDG[0]	PIN_E21
✓ <i>c<sub>out</sub></i>	LEDG[1]	PIN_E22

Aplique, como entradas, os mesmos valores apresentados na Tabela abaixo, anotando os valores de saída.

<i>a</i>	<i>b</i>	<i>a<sub>invert</sub></i>	<i>b<sub>invert</sub></i>	<i>c<sub>in</sub></i>	<i>s<sub>1</sub></i>	<i>s<sub>0</sub></i>	<i>x</i>	<i>c<sub>out</sub></i>
1	0	0	0	0	0	0	0	0 ✓
1	1	0	0	0	0	0	1	1 ✓
1	0	0	0	0	0	1	1	0 ✓
0	0	0	0	0	0	1	0	0 ✓
1	0	0	0	0	1	0	1	0 ✓
1	1	0	0	0	1	0	0	1 ✓
1	1	0	0	1	1	0	1	1 ✓
1	1	0	1	1	1	0	0	1 ✓
1	0	0	0	0	1	1	1	0 ✓
1	1	0	0	0	1	1	0	1 ✓

