

CS 232 - Assignment 4

Design Rules:

- 1) For all questions, only structural VHDL code is permitted.*
- 2) Also for every question, the top-level entity description is given in the question itself. It must be used exactly as it is in your code without changing the entity name or any of the port names and the port names must be in the same order.*

- 1) Design a 4-bit unsigned ALU. 'a' and 'b' are the two 4-bit unsigned numbers given as input to the ALU. 'result' is the 8-bit unsigned number which is output after performing the operation. 'sel' is the 3-bit mode selection input. **This must be a structural design.**

entity FoutbitALU is

port (a, b: in std_logic_vector (3 downto 0);

sel: in std_logic_vector (2 downto 0);

result: out std_logic_vector (7 downto 0));

end entity;

Design details of each operation are below.

A. Adder (sel = 000) & Subtractor (sel = 001)

It is a 4-bit unsigned adder-subtractor. **This can only be designed using a carry-look-ahead adder circuit. It is a structural design and only basic gates (and, or, not) are allowed.** During this operation, the lower 4 bits of 'result' should represent the sum (for adder) and difference (for subtractor) and the 5th bit is the carry. The remaining upper 3 bits of 'result' are don't care.

B. Multiplier (sel = 010)

It is a 4-bit unsigned multiplier. **Use an array multiplier circuit. It is a structural design and only basic gates (and, or, not) are allowed.** During the multiplication operation, all the 8 bits of 'result' will represent the multiplication result.

C. Comparator (sel = 011)

It is a 4-bit unsigned comparator. **It is a structural design and only basic gates (and, or, not) are allowed.** The comparator should be able to tell whether 'a' is less than or greater than or equal to 'b'. During comparator operation, only the lower 3 bits of 'result' will be used. The remaining upper 5 bits are don't care. The comparator outputs are explained in detail below.

- a) $a < b \Rightarrow \text{result} = \text{xxxxx}001$
- b) $a > b \Rightarrow \text{result} = \text{xxxxx}100$
- c) $a = b \Rightarrow \text{result} = \text{xxxxx}010$

D. Bitwise NAND (sel = 100)

It is a 4-bit bitwise NAND operation. **It is a structural design and only basic gates (and, or, not) are allowed.** The lower 4 bits of 'result' should represent the result of bitwise NAND and the remaining upper 4 bits are don't care.

E. Bitwise NOR (sel = 101)

It is a 4-bit bitwise NOR operation. **It is a structural design and only basic gates (and, or, not) are allowed.** The lower 4 bits of 'result' should represent the result of bitwise NOR and the remaining upper 4 bits are don't care.

F. Bitwise XOR (sel = 110)

It is a 4-bit bitwise XOR operation. **It is a structural design and only basic gates (and, or, not) are allowed.** The lower 4 bits of 'result' should represent the result of bitwise XOR and the remaining upper 4 bits are don't care.

G. Bitwise XNOR (sel = 111)

It is a 4-bit bitwise XNOR operation. **It is a structural design and only basic gates (and, or, not) are allowed.** The lower 4 bits of 'result' should represent the result of bitwise XNOR and the remaining upper 4 bits are don't care.

Things required in Submission:

1. All VHDL files of top-level and sub-components (.vhd or .vhd1 files).
2. Screenshot of Waveforms (Few input patterns are sufficient).
3. A report explaining how you designed (pdf format).

Submission rules:

1. **Deadline:** Sunday (6th Feb 2022), 23:59 pm.
2. Create a folder named **lab4** (all small letters and no spaces).
3. Inside the above-mentioned folder create 1 more folder named **q1** (all small letters and no spaces).
4. Inside the folder, **q1** put all the .vhd or .vhd1 files of both top-level entity and sub-components. Also in the same folder put the related waveform screenshots and pdf report related question1.
5. Finally, before you submit the lab4 folder on moodle you will be zipping it. The zip file should have your id number as the name. For eg: if your id is 184070026 then your zip file will be 184070026.zip