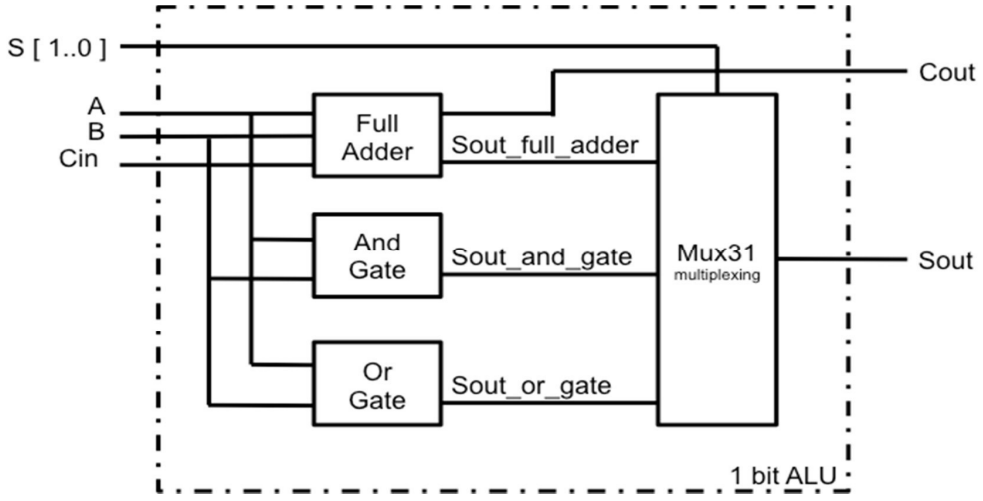
Lab 4 – Implementation of a 16-bit ALU

Before the lab

In this lab you will learn to implement an arithmetic logic unit or ALU. Your lab time will be more meaningful and effective if you enter the lab with the understanding of how an ALU (add/subtract/and/or) works.

Assignment

For this lab your first goal is to create a 1-bit ALU based on the full adder you have built on Lab3. Our ALU will have addition, subtraction, Logical AND and Logical OR functionality. Once finished, you will use the 1-bit ALU as a module to create a 16 bit ALU. Finally, you have to test the ALU by writing a test bench that goes through each case presented at the end of this lab.



The functionality of the ALU is determined by its "select" signal (S[1..0] in above picture). The mapping from select to functionality should be as follows.

S1	S0	Function
0	0	Add
0	1	Sub
1	0	And
1	1	Or

How to do subtraction:

4 – 3 = 1

4 + (–3) = 1

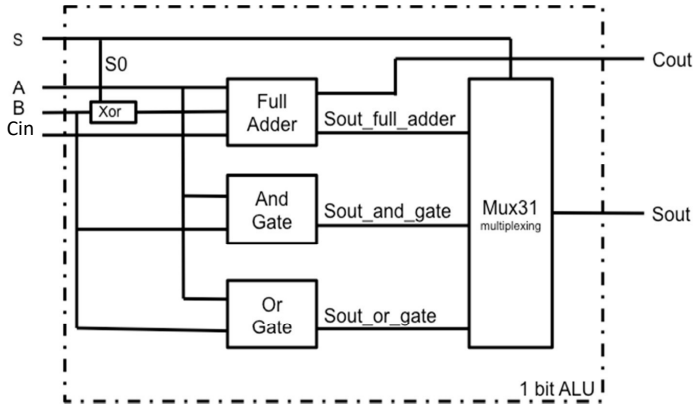
0011

1100

1101

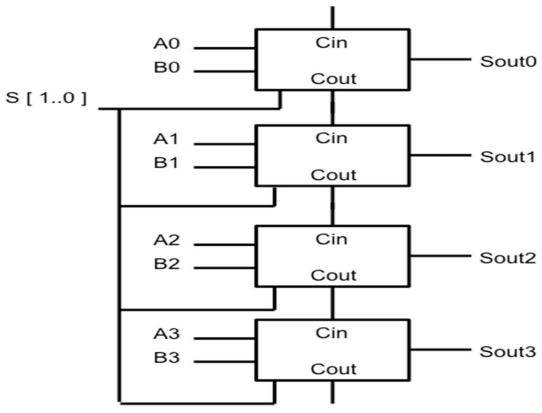
0100 + 1101 = 0001

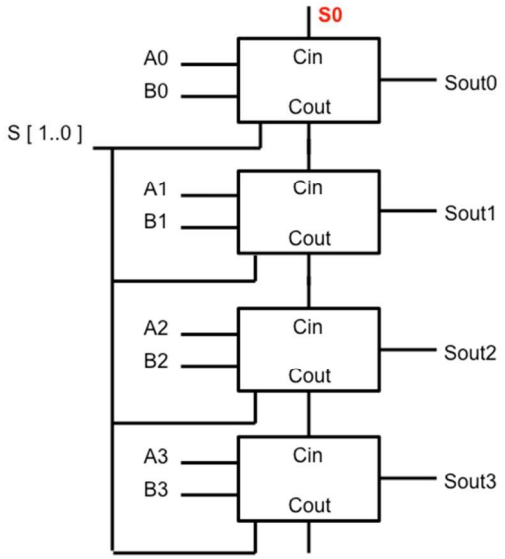
To do subtraction; add a XOR logic to input B and S0 before sending it to the full_adder.



Codes for 1-bit AND, OR, for 4-to-1 MUX is provided. Also a template for the ALU and its testbench is available. Import all files to your project (using "Add New Source"). Look at the *ALU.vhd* code first. It includes 4 *components* – full_adder, and_gate, or_gate, and MUX31. You have to declare appropriate signals and fill out the PORT MAP sections. Think it like connecting wires in a circuit with multiple ICs.

Once you have built and tested your 1-bit ALU, then you should create a 16 bit ALU by writing appropriate port mapping (using the same concepts you applied to create 1-bit ALU).





Complete your testbench to test your implementation with the values in the table below and write the result in your report!

S1	S0	A	B	Sout	Cout
0	0	150	260		
0	0	-25	65		
0	0	2	2		
0	1	550	320		
0	1	25	60		
0	1	-2	16		
1	0	64	256		
1	0	2000	3		
1	0	5342	986		
1	1	12	7		
1	1	0	2345		
1	1	-1	5		

For Lab report

Your lab report should include the following –

- Screenshots of your VHDL codes
- Screenshots of the simulator
- A table with your results