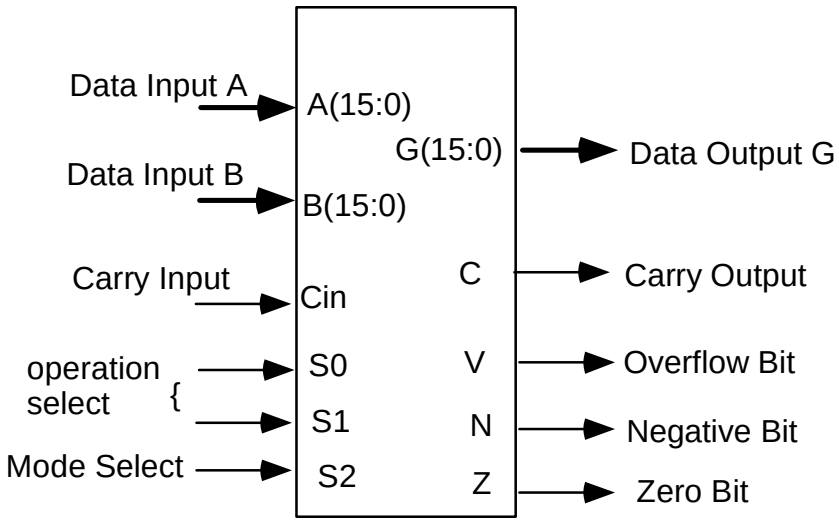
Lab 8:16 Bit Arithmetic/Logic Unit Design

Objective: Design, synthesize, and simulate a 16 bit ALU with specified operation.

16-bit ALU Specification:



Function Table for ALU

Operation Select					
S2	S1	S0	Cin	Operation	Function
0	0	0	0	G=A	Transfer A
0	0	0	1	G=A+1	Increment A
0	0	1	0	G=A+B	Addition
0	0	1	1	G=A+B+1	Add with carry input of 1
0	1	0	0	G=A+ B’	A plus 1’s complement of B
0	1	0	1	G=A-B	Subtraction
0	1	1	0	G=A-1	Decrement A
0	1	1	1	G=A	Transfer A
1	0	0	0	G=A^B	AND
1	0	1	1	G=A∨B	OR
1	1	0	0	G=A⊕B	Xor
1	1	1	1	G=A’	Not(1’s complement)

Hints for overflow dectection bit V:

Considering the full adder of the most significant bit, the overflow bit could be derived by xoring the carry-out bits of the leftmost two adders, C_n and C_{n-1}.

$$V=C_n \text{ xor } C_{n-1}$$

TASKS:

(a) Write the vhdl entity "alu" and architecture that model the above 16-bit alu
(b) Write the testbench "alu_test". Simulate your vhdl model
(c) Synthesize your vhdl model, i.e. generate schematic.