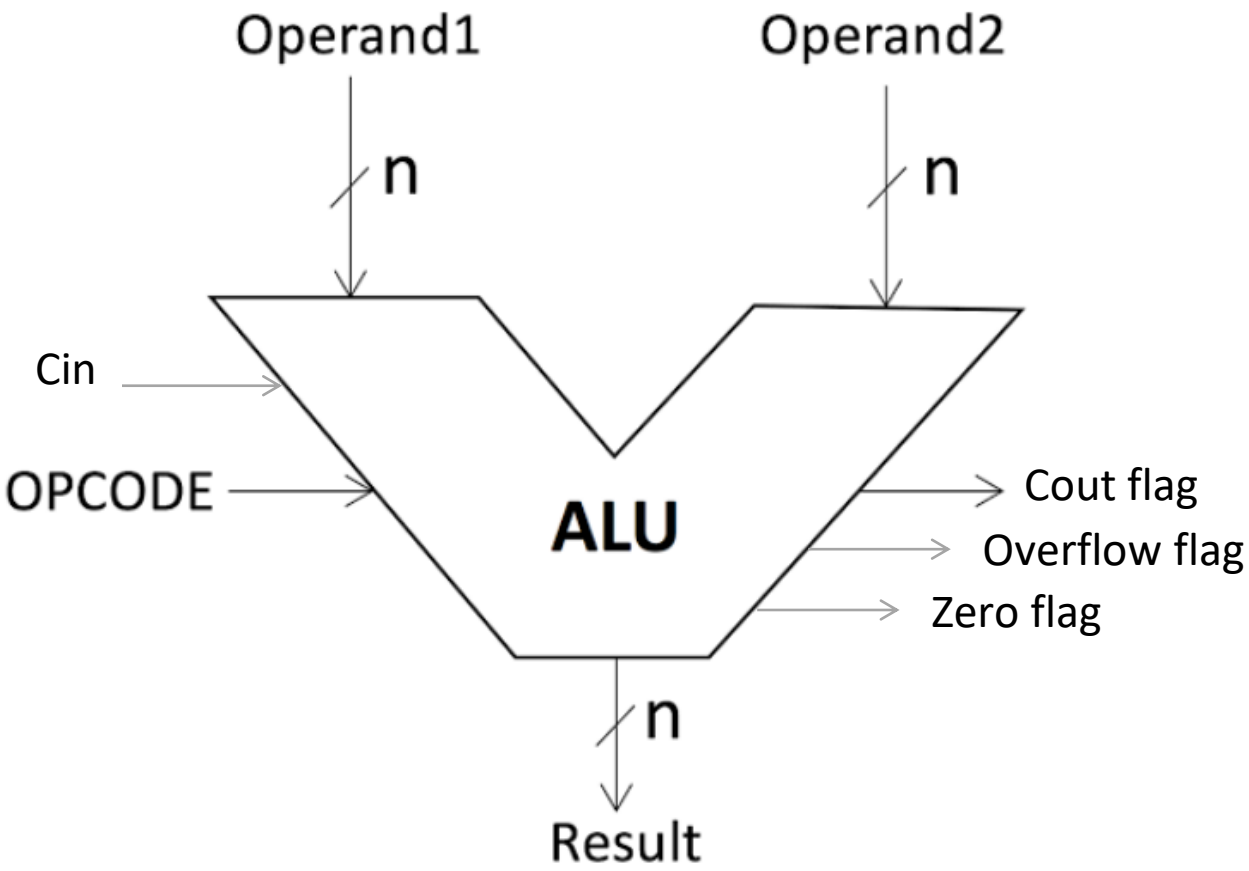
[Introduction to Computer Architecture'21] Lab Project: Phase 1

Implement n-bit full ALU. *(10 points)*

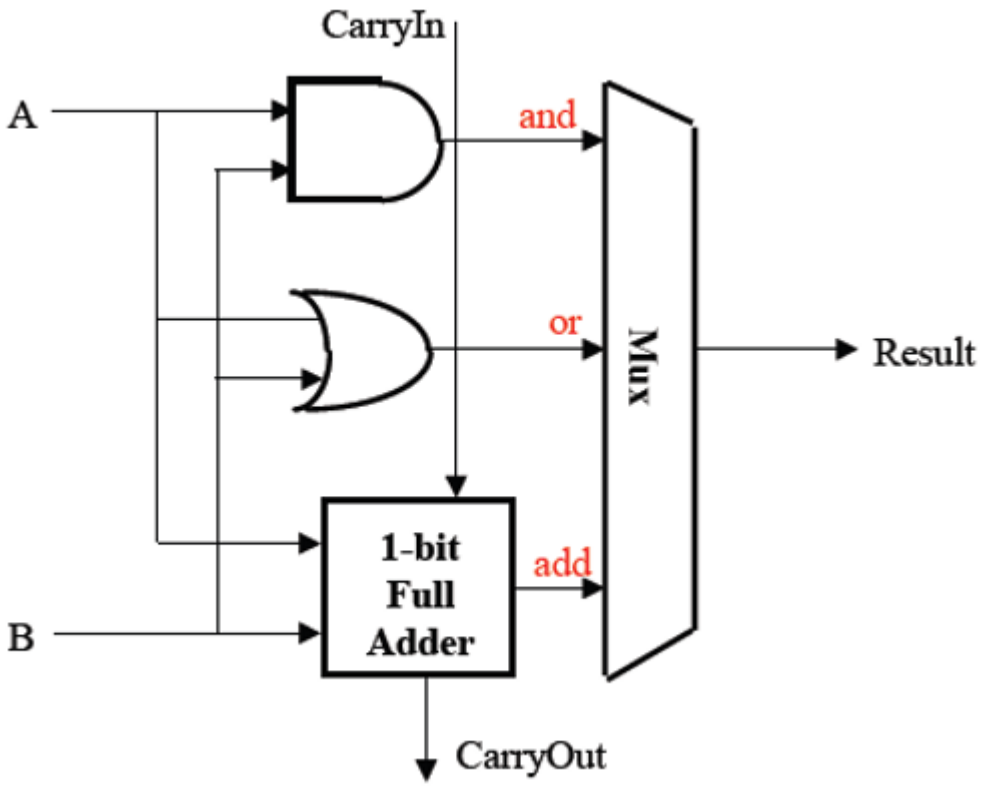
Description: The ALU performs simple addition, subtraction, multiplication, division, and logic operations, such as OR and AND. The memory stores the program's instructions and data.

Main module should be called "**ALU**"



ALU Functional Specifications:

ALU OPCODE	Function
0000	AND
0001	OR
0010	ADD
0110	SUB



1-bit ALU Process

Testbench [*Main*] should look as follows:

```
module Main();
    reg [31:0] data1;
    reg [31:0] data2;
    reg [3:0] aluop;
    reg cin;
    wire [31:0] dataout;
    wire cflag;
    wire zflag;
    wire oflag;
```

Delivery Method: on Moodle [*only one in each team can submit the team project folder*]

Grading Schema:

Code structure	2 (points)
WAVE window	2 (points)
RTL Design	2 (points)
Test 1	0.25 (point)
Test 2	0.25 (point)
Test 3	0.25 (point)
Test 4	1 (point)
Test 5	0.25 (point)
Test 6	1 (point)
Test 7	1 (point)
Total	**10 (points)**

Deadline: Friday November 19, 2021 @ 11:59 PM

Nile University

[Introduction to Computer Architecture'21] Lab Project: Phase 1

Deliverables:

You should submit the following file(s) on Moodle:

1. ZIP file for Lab Project Phase 1 (File name must be of the following format: **CA21_LP1_Student1ID_ Student2ID_ Student3ID.zip**) contains the following file(s):
    a. Each module you're created [.v].
    b. Screenshot of the WAVE window [7 tests].
    c. ALU Schematic [RTL Design].
    d. Document file [.pdf] contains your work in each module.
    e. Text file contains the team's info. (*Team ID, Team Names and Team IDs*).

upload the below files on Moodle:

Rules:

1. Each team consists of ***3 students*** at most.
2. Any submissions after the deadline are not acceptable.
3. Important Plagiarism Notice:
    a. Deliverables based on other students' solutions lead to rejection of BOTH deliverables.
    b. Examples of plagiarism (but not limited to) copying (partial) code from other students, open-source software (or Internet in general), tutors, etc.

Search before ask

Verilog + ZYBO Z7 board Help:

- Check this link for Verilog syntax: https://www.nandland.com/verilog/tutorials/index.html
- Check this link for ZYBO Z7 board info.: https://digilent.com/reference/programmable-logic/zybo/start

If you need any help regarding anything about the course, ask:

- Engr. Ahmad M. Abdel-Hafeez: akassem@nu.edu.eg
- Engr. Mohammad Rady: mrady@nu.edu.eg

Test before submitting:

Test #1:

Type	Name	Value
input	data1	11000000000000000000000000000000
input	data2	10100000000000000000000000000000
input	aluop	0000
ouput	dataout	10000000000000000000000000000000
output	zflag	0

Test #2:

Type	Name	Value
input	data1	11000000000000000000000000000000
input	data2	10100000000000000000000000000000
input	aluop	0001
ouput	dataout	11100000000000000000000000000000
output	zflag	0

Test #3:

Type	Name	Value
input	data1	01110000000000000000000000000000
input	data2	01100000000000000000000000000000
input	aluop	0010
ouput	dataout	11010000000000000000000000000000
output	zflag	0
output	oflag	1
output	cflag	0

Test #4:

Type	Name	Value
input	data1	11110000000000000000000000000000
input	data2	00010000000000000000000000000000
input	aluop	0010
ouput	dataout	00000000000000000000000000000000
output	zflag	1
output	oflag	0
output	cflag	1

Test #5:

Type	Name	Value
input	data1	00000000000000000000000000000111
input	data2	00000000000000000000000000000110
input	aluop	0110
input	cin	1
ouput	dataout	00000000000000000000000000000001
output	zflag	0
output	oflag	0
output	cflag	1

Test #6:

Type	Name	Value
input	data1	00000000000000000000000000000110
input	data2	00000000000000000000000000000111
input	aluop	0110
input	cin	1
ouput	dataout	11111111111111111111111111111111
output	zflag	0
output	oflag	0
output	cflag	0

Test #7:

Type	Name	Value
input	data1	11111111111111111111111111111000
input	data2	00000000000000000000000000000001
input	aluop	0110
input	cin	1
ouput	dataout	11111111111111111111111111110111
output	zflag	0
output	oflag	1
output	cflag	1