California State University, Northridge

Department of Electrical & Computer Engineering

ECE 526L

Lab 9 Report

By

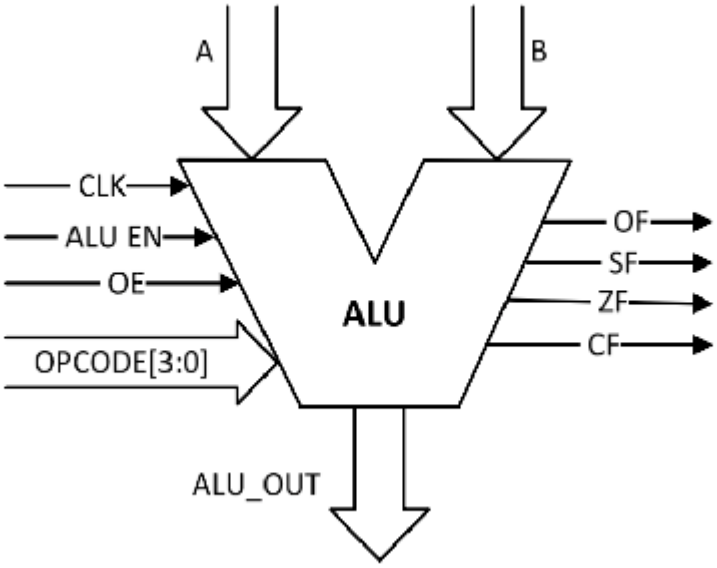
Avinash Damse

CSUN ID- 203131064

1: Introduction

The objective of this lab Model an arithmetic-logic unit using behavioral Verilog/System Verilog and Verify the outputs and functionalities of the Arithmetic Logic.

Here we have to use following diagram to build an ALU



2: Procedure

a. Part 1: Creating ALU Module

In this lab I have created a ALU module. Inside the module I have assigned “CLK, EN, OE, OPCODE, A, B” as input variables and “ALU_OUT” as output variable and CF, OF, SF, ZF as an output registers. Then I wrote the ALU logic for given arithmetic operations inside module. After completing the code I ended the module using and saved the file with name “ALU.v”.

b. Part 4: Creating ALU_tb Testbench

I have written the test bench for the **ALU** module. We require test bench just to make sure that the module we have created is working properly. Here, in this

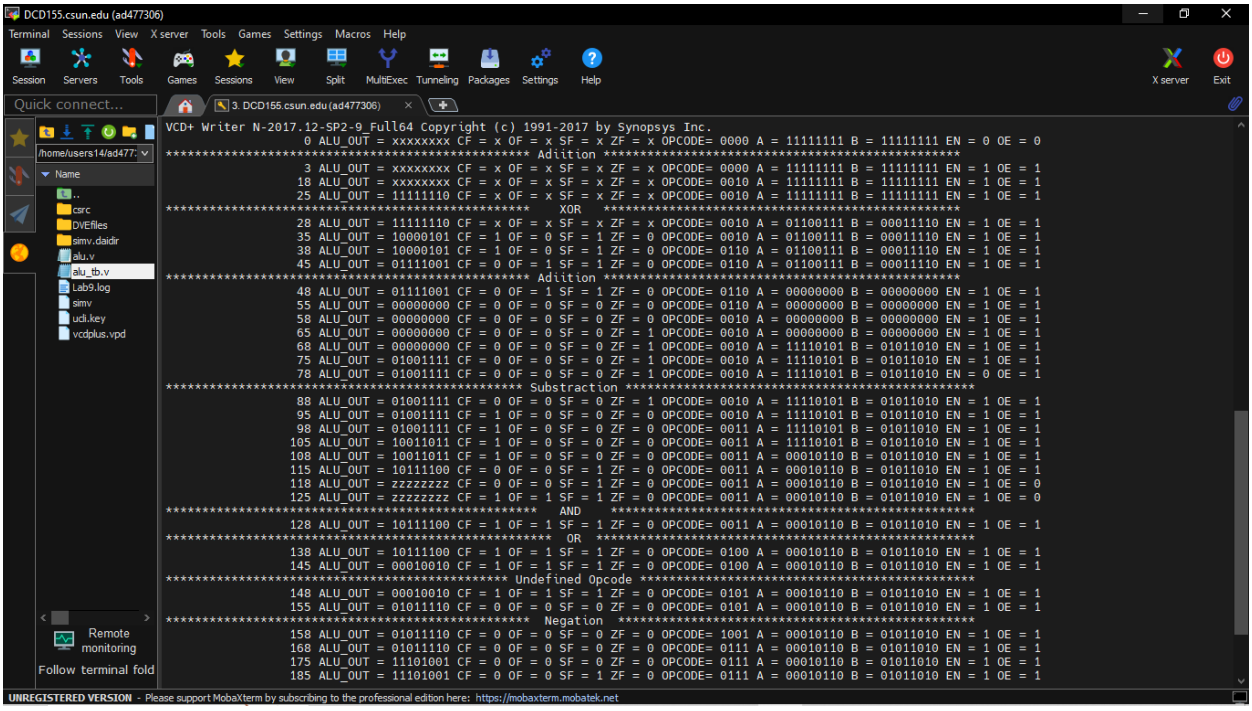
testbench I have used some inputs with EN and OE to test the functioning of ALU.

c. Part 3: execution.

Using “vcs -debug -full64 alu.v alu_tb.v” command I executed testbench file.

d. Part 4: Simulation

After an execution of all modules, I have run the command “simv” for simulation.



e. Part 5: Creating Log File

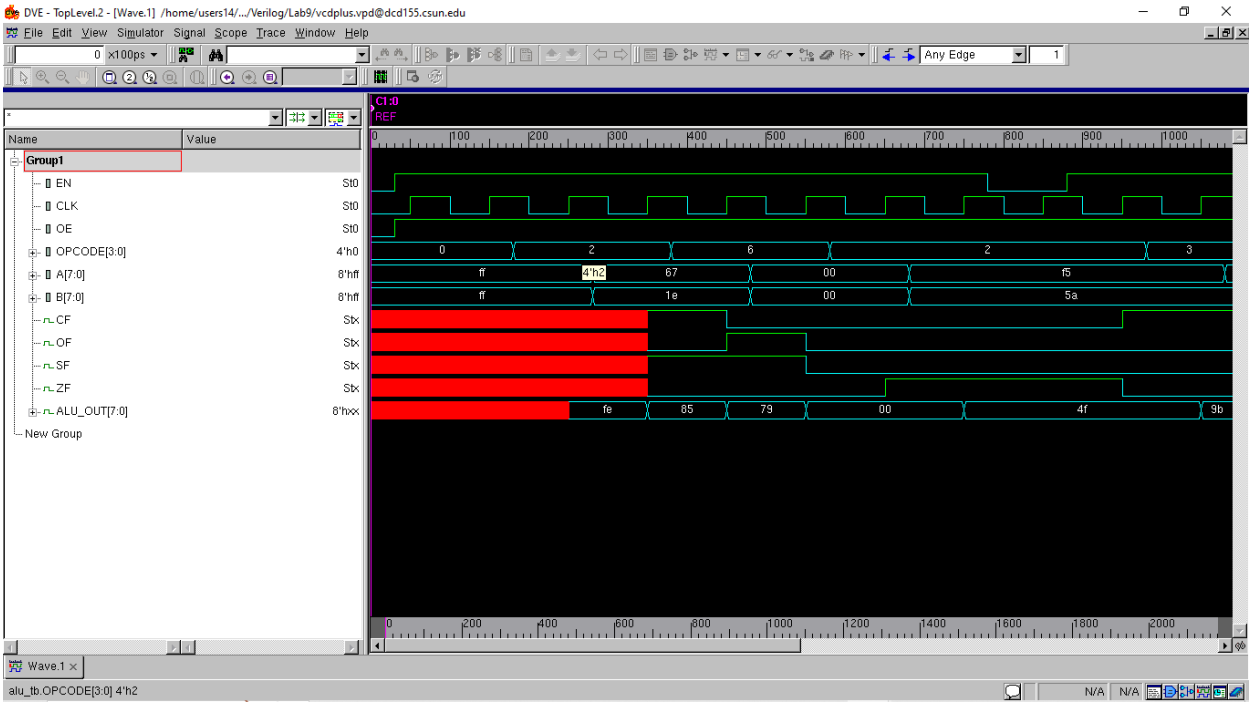
After running the simulation I created the log file for first stategy using the “simv -l Lab8_non_exh.log” command

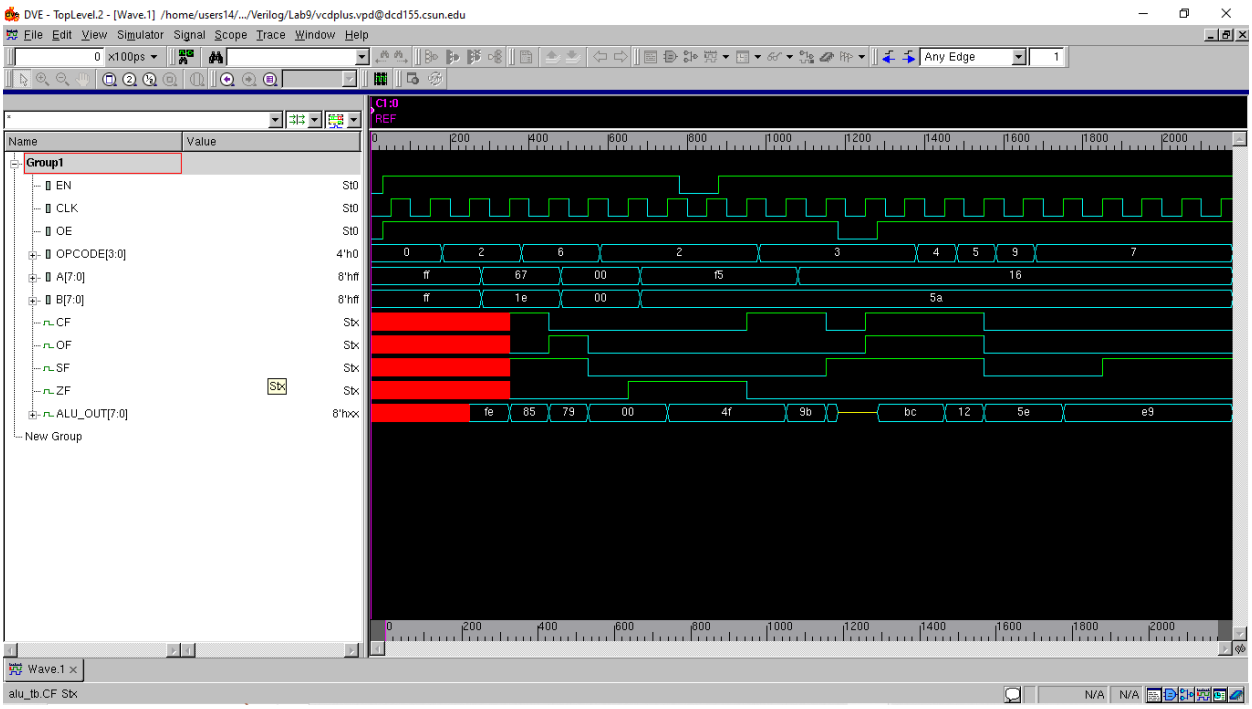
```
Command: /home/users14/ad477306/Verilog/Lab9/./simv -l Lab9.log
Chronologic VCS simulator copyright 1991-2017
Contains Synopsys proprietary information.
Compiler version N-2017.12-SP2-9_Full64; Runtime version N-2017.12-SP2-9_Full64;  May  3 23:04
2023
VCD+ Writer N-2017.12-SP2-9_Full64 Copyright (c) 1991-2017 by Synopsys Inc.
                0 ALU_OUT = xxxxxxxx CF = x OF = x SF = x ZF = x OPCODE= 0000 A = 11111111 B =
11111111 EN = 0 OE = 0
************************************************** Adiition **************************************************
                3 ALU_OUT = xxxxxxxx CF = x OF = x SF = x ZF = x OPCODE= 0000 A = 11111111 B =
11111111 EN = 1 OE = 1
               18 ALU_OUT = xxxxxxxx CF = x OF = x SF = x ZF = x OPCODE= 0010 A = 11111111 B =
11111111 EN = 1 OE = 1
               25 ALU_OUT = 11111110 CF = x OF = x SF = x ZF = x OPCODE= 0010 A = 11111111 B =
11111111 EN = 1 OE = 1
**************************************************    XOR   **************************************************
               28 ALU_OUT = 11111110 CF = x OF = x SF = x ZF = x OPCODE= 0010 A = 01100111 B =
00011110 EN = 1 OE = 1
               35 ALU_OUT = 10000101 CF = 1 OF = 0 SF = 1 ZF = 0 OPCODE= 0010 A = 01100111 B =
00011110 EN = 1 OE = 1
               38 ALU_OUT = 10000101 CF = 1 OF = 0 SF = 1 ZF = 0 OPCODE= 0110 A = 01100111 B =
00011110 EN = 1 OE = 1
               45 ALU_OUT = 01111001 CF = 0 OF = 1 SF = 1 ZF = 0 OPCODE= 0110 A = 01100111 B =
00011110 EN = 1 OE = 1
************************************************** Adiition **************************************************
               48 ALU_OUT = 01111001 CF = 0 OF = 1 SF = 1 ZF = 0 OPCODE= 0110 A = 00000000 B =
00000000 EN = 1 OE = 1
               55 ALU_OUT = 00000000 CF = 0 OF = 0 SF = 0 ZF = 0 OPCODE= 0110 A = 00000000 B =
00000000 EN = 1 OE = 1
               58 ALU_OUT = 00000000 CF = 0 OF = 0 SF = 0 ZF = 0 OPCODE= 0010 A = 00000000 B =
00000000 EN = 1 OE = 1
               65 ALU_OUT = 00000000 CF = 0 OF = 0 SF = 0 ZF = 1 OPCODE= 0010 A = 00000000 B =
00000000 EN = 1 OE = 1
               68 ALU_OUT = 00000000 CF = 0 OF = 0 SF = 0 ZF = 1 OPCODE= 0010 A = 11110101 B =
01011010 EN = 1 OE = 1
               75 ALU_OUT = 01001111 CF = 0 OF = 0 SF = 0 ZF = 1 OPCODE= 0010 A = 11110101 B =
01011010 EN = 1 OE = 1
               78 ALU_OUT = 01001111 CF = 0 OF = 0 SF = 0 ZF = 1 OPCODE= 0010 A = 11110101 B =
01011010 EN = 0 OE = 1
************************************************** Substraction **************************************************
               88 ALU_OUT = 01001111 CF = 0 OF = 0 SF = 0 ZF = 1 OPCODE= 0010 A = 11110101 B =
01011010 EN = 1 OE = 1
               95 ALU_OUT = 01001111 CF = 1 OF = 0 SF = 0 ZF = 0 OPCODE= 0010 A = 11110101 B =
01011010 EN = 1 OE = 1
               98 ALU_OUT = 01001111 CF = 1 OF = 0 SF = 0 ZF = 0 OPCODE= 0011 A = 11110101 B =
01011010 EN = 1 OE = 1
              105 ALU_OUT = 10011011 CF = 1 OF = 0 SF = 0 ZF = 0 OPCODE= 0011 A = 11110101 B =
01011010 EN = 1 OE = 1
```

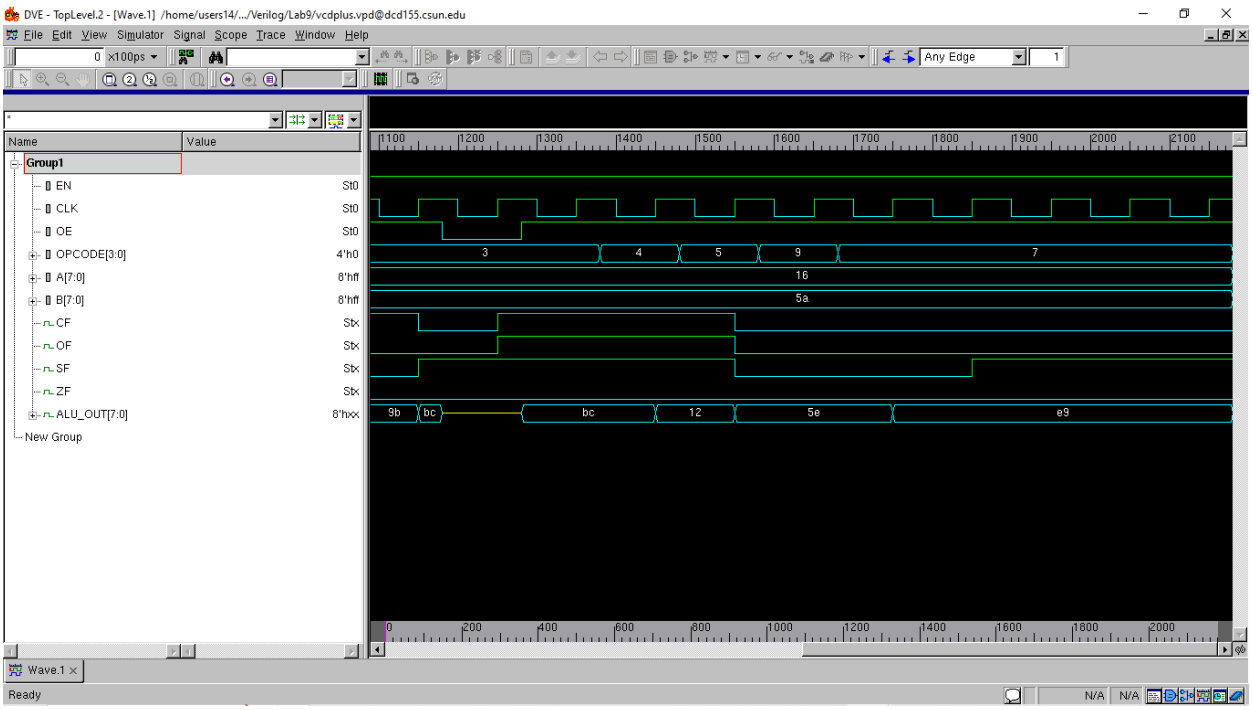
```
            108 ALU_OUT = 10011011 CF = 1 OF = 0 SF = 0 ZF = 0 OPCODE= 0011 A = 00010110 B =
01011010 EN = 1 OE = 1
            115 ALU_OUT = 10111100 CF = 0 OF = 0 SF = 1 ZF = 0 OPCODE= 0011 A = 00010110 B =
01011010 EN = 1 OE = 1
            118 ALU_OUT = zzzzzzzz CF = 0 OF = 0 SF = 1 ZF = 0 OPCODE= 0011 A = 00010110 B =
01011010 EN = 1 OE = 0
            125 ALU_OUT = zzzzzzzz CF = 1 OF = 1 SF = 1 ZF = 0 OPCODE= 0011 A = 00010110 B =
01011010 EN = 1 OE = 0
************************************************** AND **************************************************
            128 ALU_OUT = 10111100 CF = 1 OF = 1 SF = 1 ZF = 0 OPCODE= 0011 A = 00010110 B =
01011010 EN = 1 OE = 1
************************************************** OR **************************************************
            138 ALU_OUT = 10111100 CF = 1 OF = 1 SF = 1 ZF = 0 OPCODE= 0100 A = 00010110 B =
01011010 EN = 1 OE = 1
            145 ALU_OUT = 00010010 CF = 1 OF = 1 SF = 1 ZF = 0 OPCODE= 0100 A = 00010110 B =
01011010 EN = 1 OE = 1
********************************************** Undefined Opcode **********************************************
            148 ALU_OUT = 00010010 CF = 1 OF = 1 SF = 1 ZF = 0 OPCODE= 0101 A = 00010110 B =
01011010 EN = 1 OE = 1
            155 ALU_OUT = 01011110 CF = 0 OF = 0 SF = 0 ZF = 0 OPCODE= 0101 A = 00010110 B =
01011010 EN = 1 OE = 1
************************************************** Negation **************************************************
            158 ALU_OUT = 01011110 CF = 0 OF = 0 SF = 0 ZF = 0 OPCODE= 1001 A = 00010110 B =
01011010 EN = 1 OE = 1
            168 ALU_OUT = 01011110 CF = 0 OF = 0 SF = 0 ZF = 0 OPCODE= 0111 A = 00010110 B =
01011010 EN = 1 OE = 1
            175 ALU_OUT = 11101001 CF = 0 OF = 0 SF = 0 ZF = 0 OPCODE= 0111 A = 00010110 B =
01011010 EN = 1 OE = 1
            185 ALU_OUT = 11101001 CF = 0 OF = 0 SF = 1 ZF = 0 OPCODE= 0111 A = 00010110 B =
01011010 EN = 1 OE = 1
$finish called from file "alu_tb.v", line 63.
$finish at simulation time                2180
         V C S   S i m u l a t i o n   R e p o r t
Time: 218000 ps
CPU Time:      0.250 seconds;       Data structure size:   0.0Mb
Wed May  3 23:04:27 2023
```

f. Part 6: Seeing the waveform.

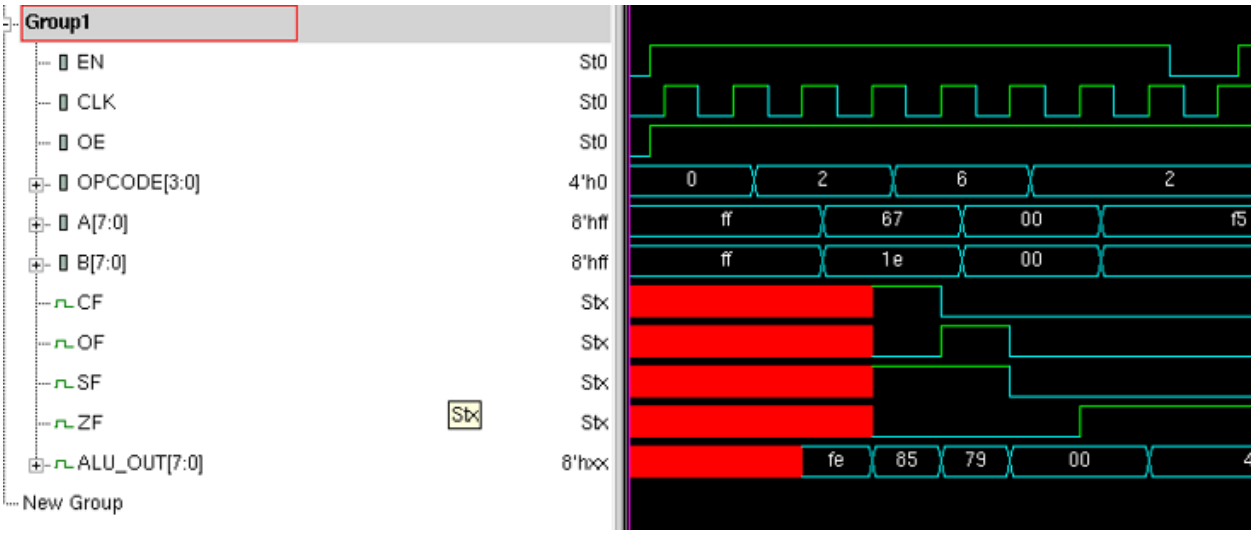
After creating the log file I opened the DVE using “dve -full64 &” command to see the waveforms.


DVE - TopLevel.2 - [Wave.1] /home/users14/.../Verilog/Lab9/vcdplus.vpd@dcd155.csun.edu
File Edit View Simulator Signal Scope Trace Window Help
Any Edge
Name
Value
Group1
EN
CLK
OE
OPCODE[3:0]
A[7:0]
B[7:0]
CF
OF
SF
ZF
ALU_OUT[7:0]
New Group
Wave.1
alu_tb.OPCODE[3:0] 4'h2

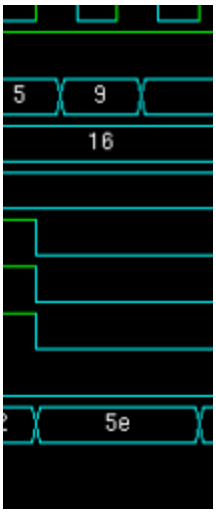

DVE - TopLevel.2 - [Wave.1] /home/users14/.../Verilog/Lab9/vcdplus.vpd@dcd155.csun.edu
File Edit View Simulator Signal Scope Trace Window Help
Any Edge
Name
Value
Group1
EN
CLK
OE
OPCODE[3:0]
A[7:0]
B[7:0]
CF
OF
SF
ZF
ALU_OUT[7:0]
New Group
Wave.1
alu_tb.CF Stx



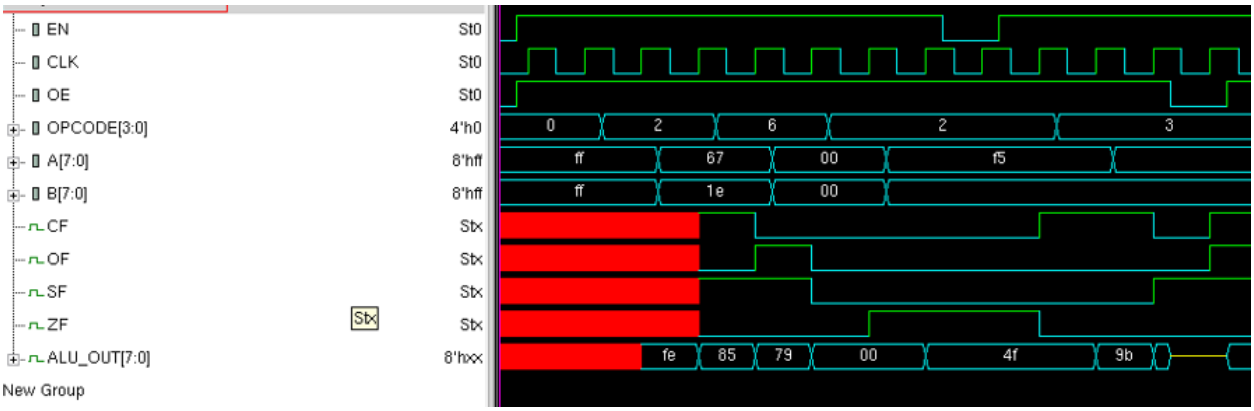
TESTING :



Here you can see that OPCODE is 2 which is 0010 in binary for which A= 67 and B=1e (in HEX) and their addition is ALU_OUT= 85 (in HEX).



Here 9 (4’b1001) which is undefined OPCODE for which ALU_OUT remains ineffective.



Here for OE “0” the ALU_OUT is “zzzzzzzz” you can check it in the simulation.

Conclusion:

The constructed arithmetic-logic unit in this experiment was verified of its functionalities. These include addition, subtraction, bitwise XOR, AND, OR, and negation. The outputs demonstrated that the carry flag was set for addition operations producing a carry-out, as well as for subtraction operations wherein input A is less than B. Bitwise operations that were performed functioned as expected also. The outputs were also shown to hold their values during a disabled state of the unit, and during an invalid opcode input.

I hereby attest that this lab report is entirely my own work. I have not copied either code or text from anyone, nor have I allowed or will I allow anyone to copy my work.

Name (printed) Avinash Damse

Name(signed)  Date : 4-May-2023