Derek Barbosa, Peter Wallace
Professor Herbordt: EC 413
Oct 30th 2020

Lab 5: Pipeline Datapath

Datapath and Timing Diagrams

Description

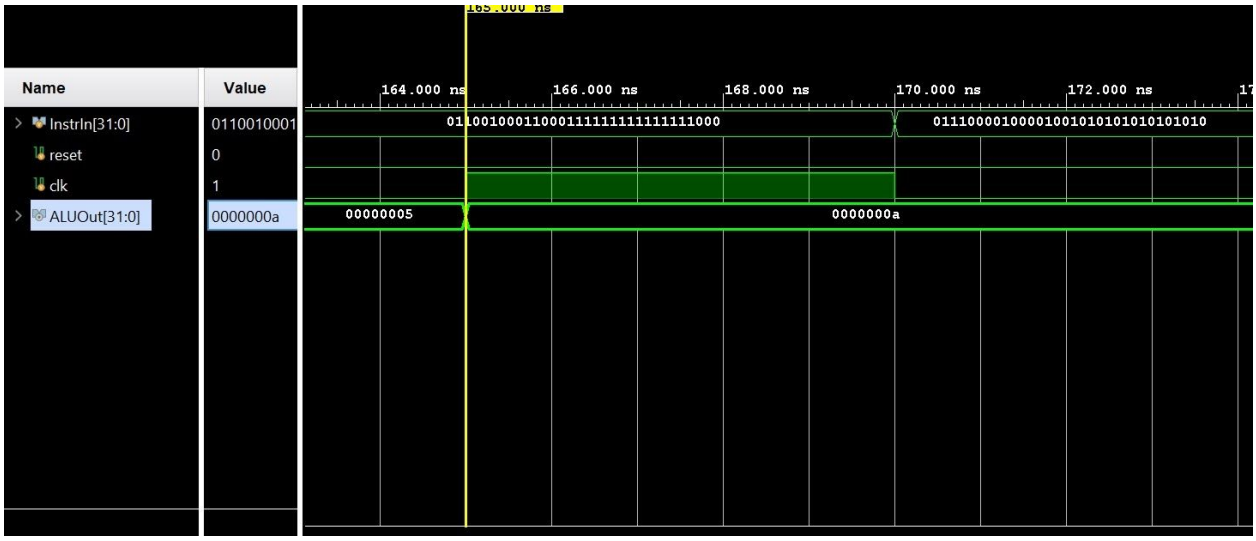
We built our own datapath from the ground up. We initialized the three stage registers and “piped” each output according to the diagram given (see Lab 5 Manual).

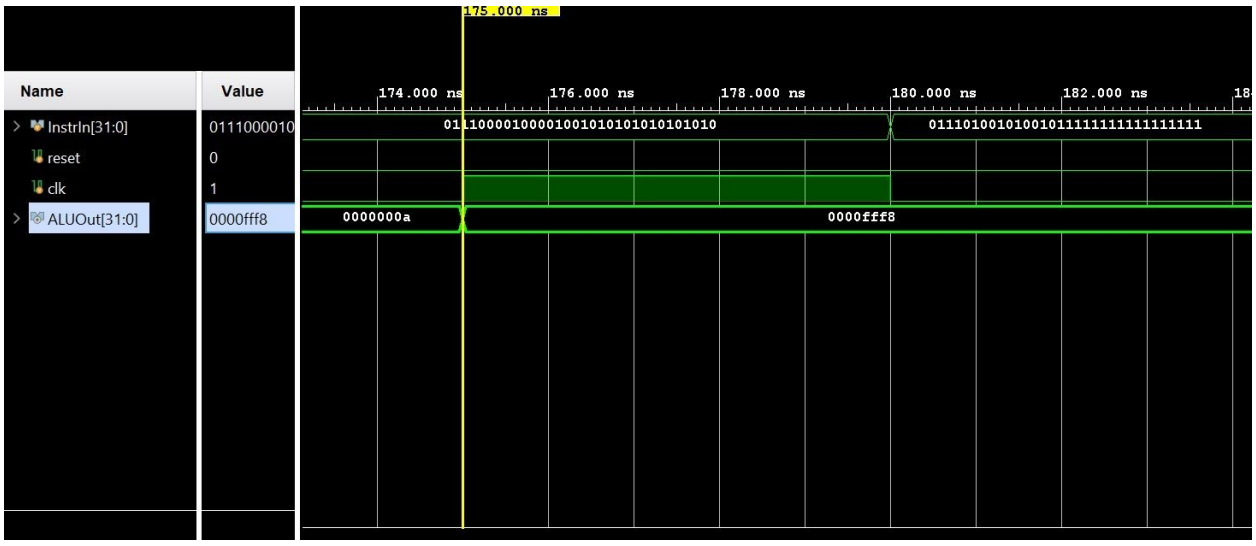
To test our datapath, we utilized the I-type and R-type instructions provided to us from the Lab Manual and other sample test benches (see the .tb file).

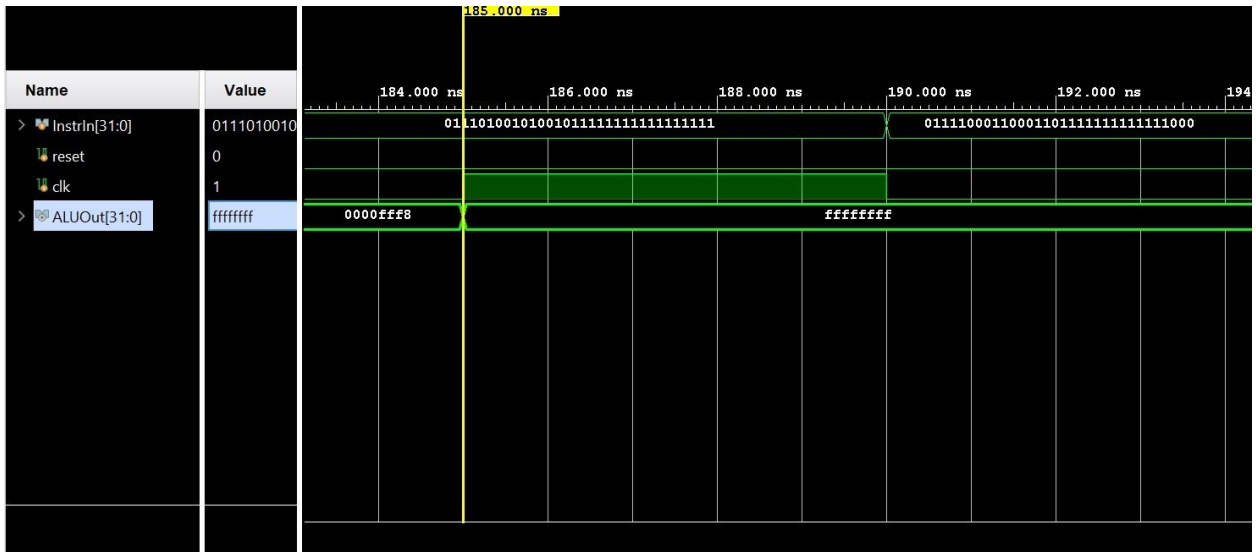
```
#30
InstrIn= 32'b011010_00000_00000_0000000000000101;              // I, add r0 with 00000005 =>   r0 = 00000005

#10
InstrIn= 32'b011010_00001_00001_0000000000001010;              // I, add r1 with 0000000A =>   r1 = 0000000A

#10
InstrIn= 32'b011010_00010_00010_1111111111111000;              // I, add r2 with 0000FFF8 =>   r2 = 0000FFF8

#10
InstrIn= 32'b011001_00011_00011_1111111111111000;              // I, not r3                                  =>   r3 = FFFFFFFF^M
^M
#10
InstrIn= 32'b011100_00100_00100_1010101010101010;              // I, or r4 with 0000AAAA  =>   r4 = 00000AAAA^M
^M
#10
InstrIn= 32'b011101_00101_00101_1111111111111111;              // I, and r5 with 0000FFFF =>   r5 = 00000000^M
^M
#10
InstrIn= 32'b011110_00110_00110_1111111111111000;              // I, slt r6 with 0000FFFF8 =>  r6 = 00000001^M

---- R type ---
#10
InstrIn= 32'b010001_00111_00001_00000_00000000000;             // R, not r1(0000000A)                  =>  r7 = FFFFFFF5

#10
InstrIn= 32'b010010_01000_00001_00010_00000000000;             // R, add r1(0000000A) with r2(0000FFF8) =>  r8 = 00010002^M
^M
#10
InstrIn= 32'b010010_01001_00001_00011_00000000000;             // R, add r1(0000000A) with r3(FFFFFFFF) =>  r9 = 00000009^M
^M
#10
InstrIn= 32'b010010_01010_00001_00100_00000000000;             // R, add r1(0000000A) with r4(0000AAAA) =>  r10 = 0000AAB4^M
^M
#10
InstrIn= 32'b010010_01011_00001_00101_00000000000;             // R, add r1(0000000A) with r5(00000000) =>  r11 = 0000000A  ^M
^M
#10
InstrIn= 32'b010010_01100_00001_00110_00000000000;             // R, add r1(0000000A) wtih r6(00000001) =>  r12 = 0000000B  ^M
```

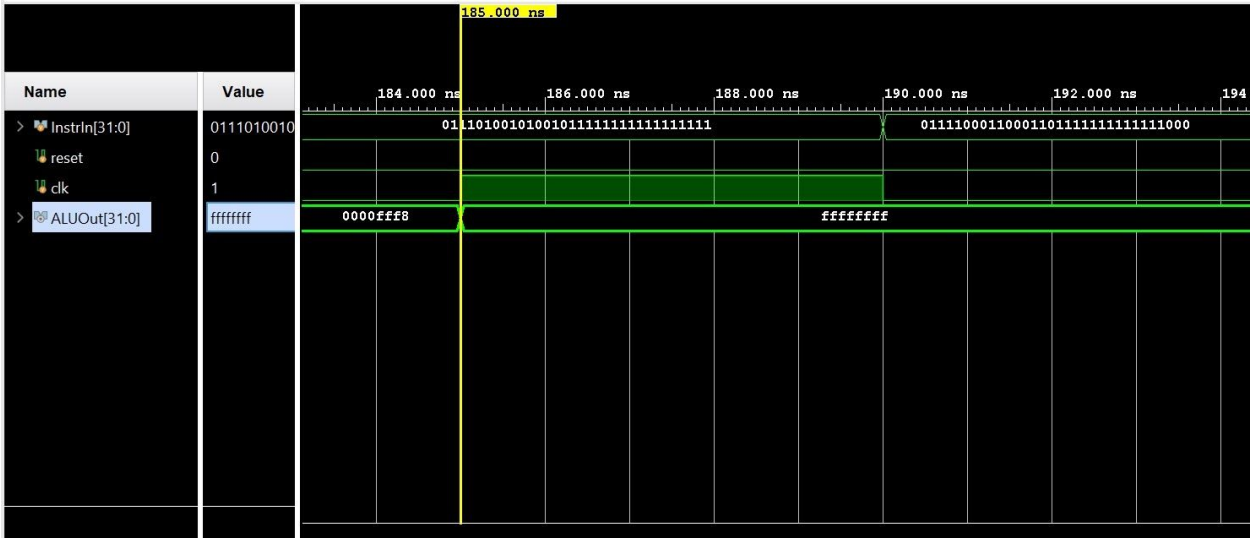
Some sample instructions

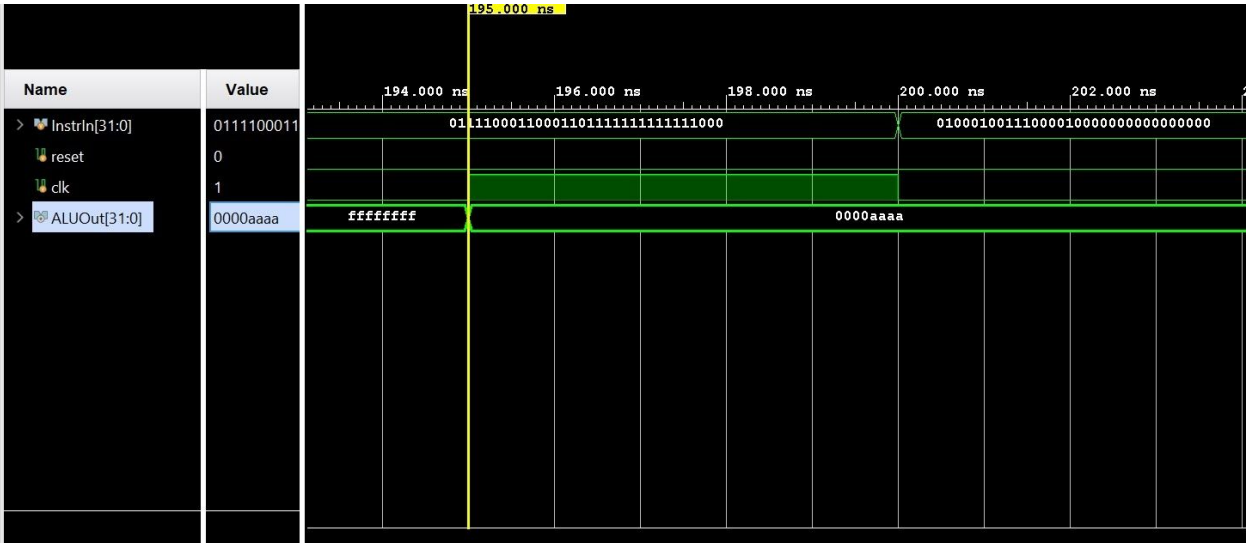
Waveforms

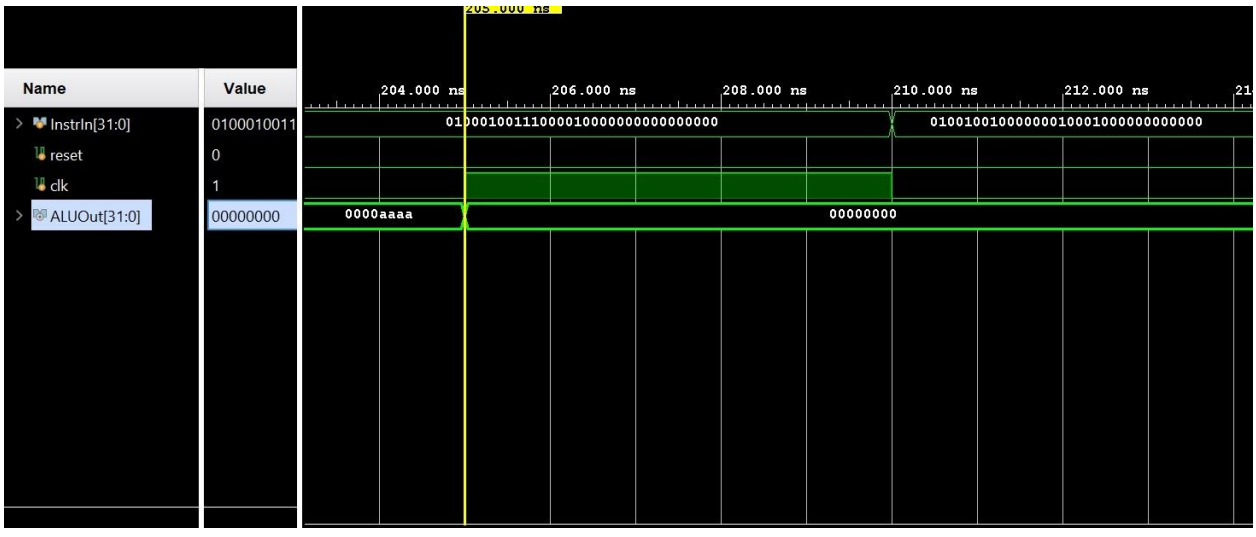

Name
Value
InstrIn[31:0]
0110010001
reset
0
clk
1
ALUOut[31:0]
0000000a
165.000 ns
164.000 ns
166.000 ns
168.000 ns
170.000 ns
172.000 ns
00000005
0000000a

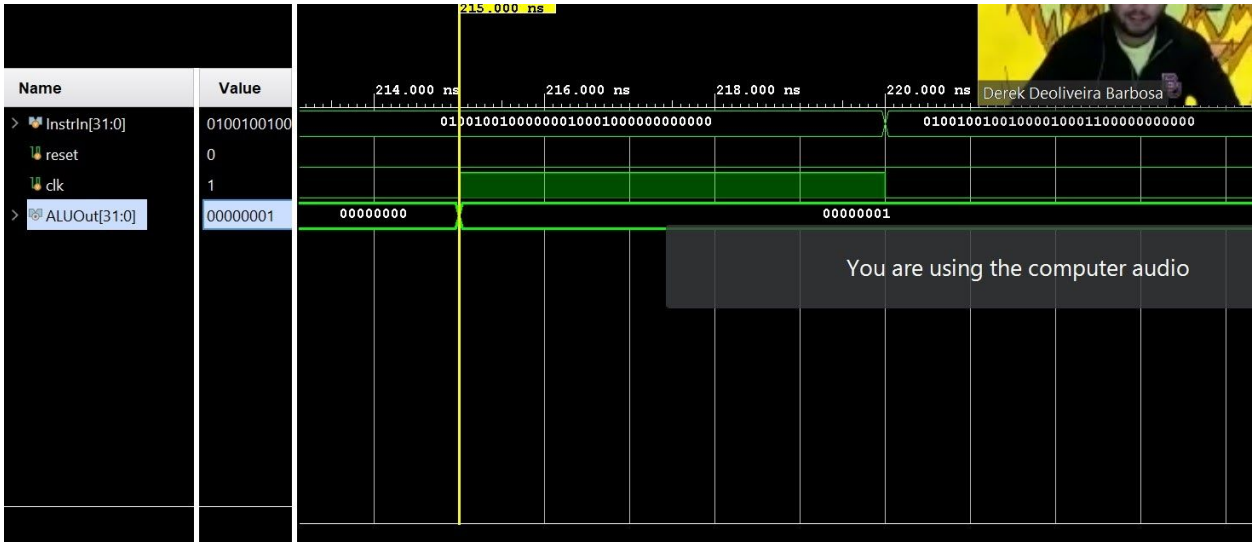

Name
Value
InstrIn[31:0]
0111000010
reset
0
clk
1
ALUOut[31:0]
0000fff8
175.000 ns
174.000 ns
176.000 ns
178.000 ns
180.000 ns
182.000 ns
0000000a
0000fff8

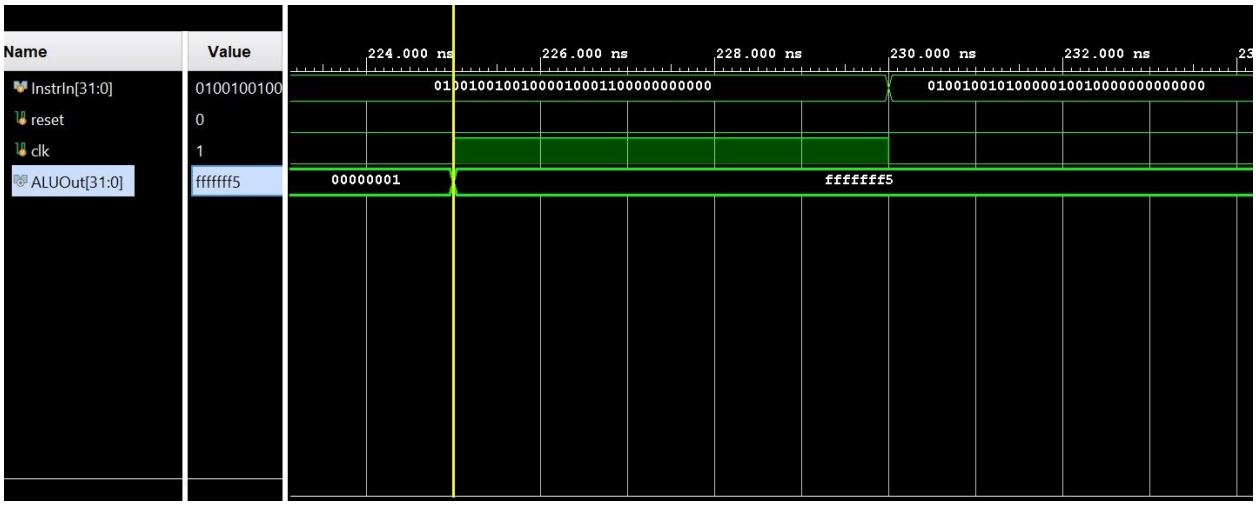

Name
Value
InstrIn[31:0]
0111010010
reset
0
clk
1
ALUOut[31:0]
ffffffff
185.000 ns
184.000 ns
186.000 ns
188.000 ns
190.000 ns
192.000 ns
0000fff8
ffffffff

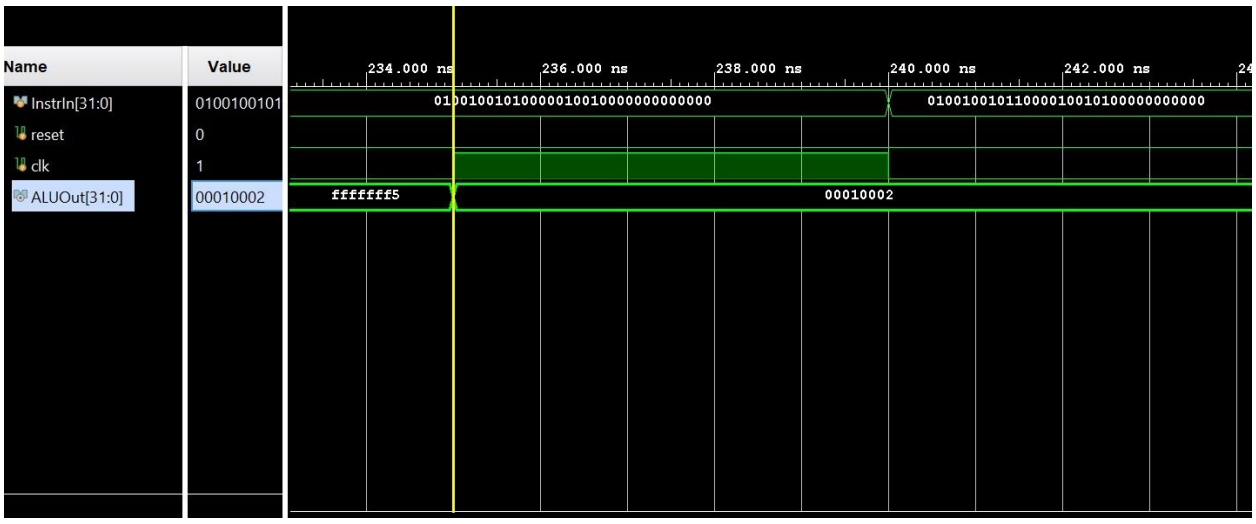

Name
Value
InstrIn[31:0]
0111010010
reset
0
clk
1
ALUOut[31:0]
ffffffff
185.000 ns
184.000 ns
186.000 ns
188.000 ns
190.000 ns
192.000 ns
0000fff8
ffffffff

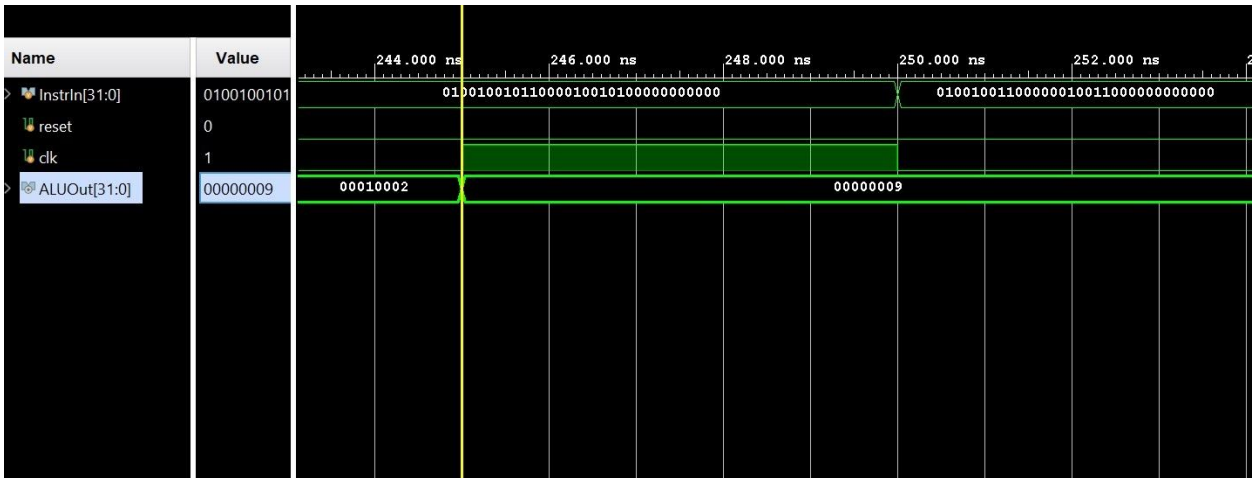

Name
Value
InstrIn[31:0]
0111100011
reset
0
clk
1
ALUOut[31:0]
0000aaaa
195.000 ns
194.000 ns
196.000 ns
198.000 ns
200.000 ns
202.000 ns
ffffffff
0000aaaa

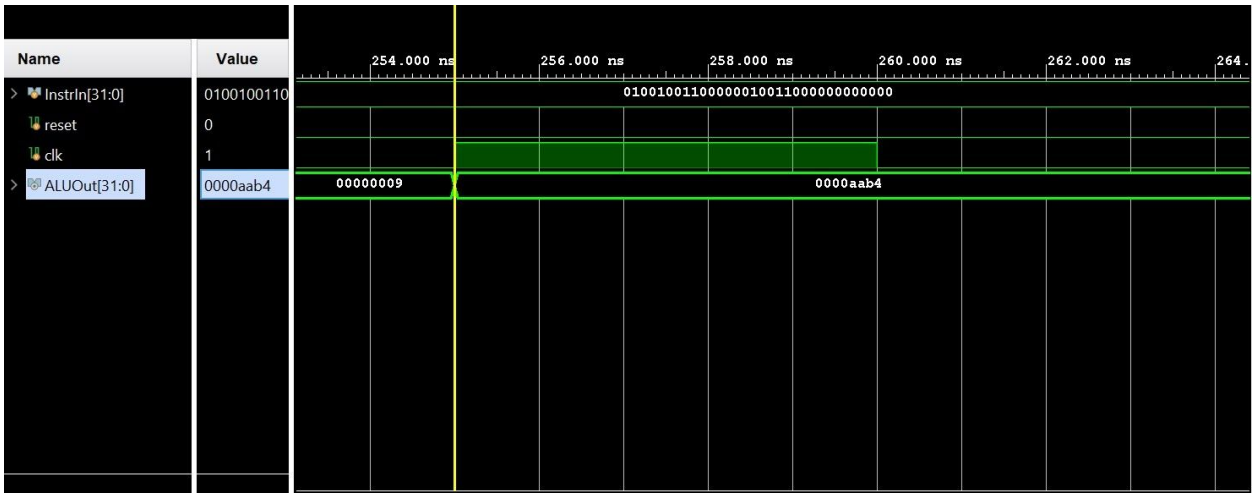

Name
Value
InstrIn[31:0]
0100010011
reset
0
clk
1
ALUOut[31:0]
00000000
204.000 ns
206.000 ns
208.000 ns
210.000 ns
212.000 ns
0000aaaa
00000000

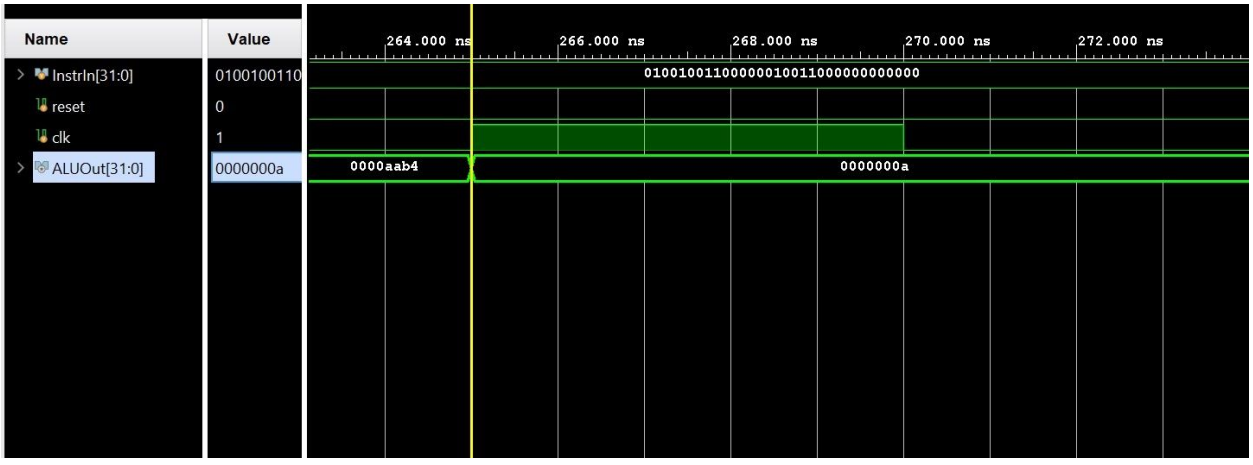

Name
Value
InstrIn[31:0]
0100100100
reset
0
clk
1
ALUOut[31:0]
00000001
215.000 ns
214.000 ns
216.000 ns
218.000 ns
220.000 ns
Derek Deoliveira Barbosa
00000000
00000001
You are using the computer audio

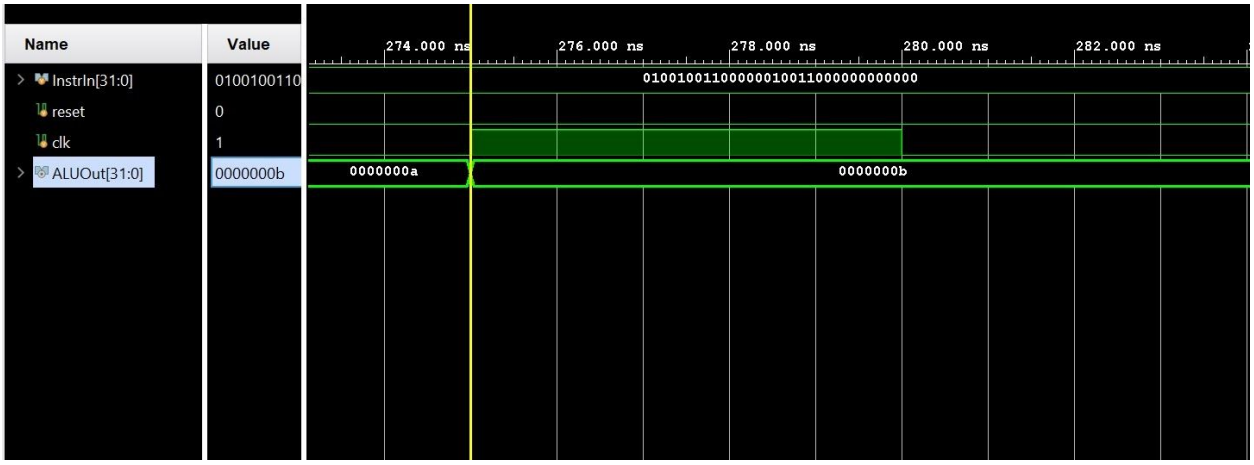

Name
Value
InstrIn[31:0]
0100100100
reset
0
clk
1
ALUOut[31:0]
fffffff5
224.000 ns
226.000 ns
228.000 ns
230.000 ns
232.000 ns
00000001
fffffff5

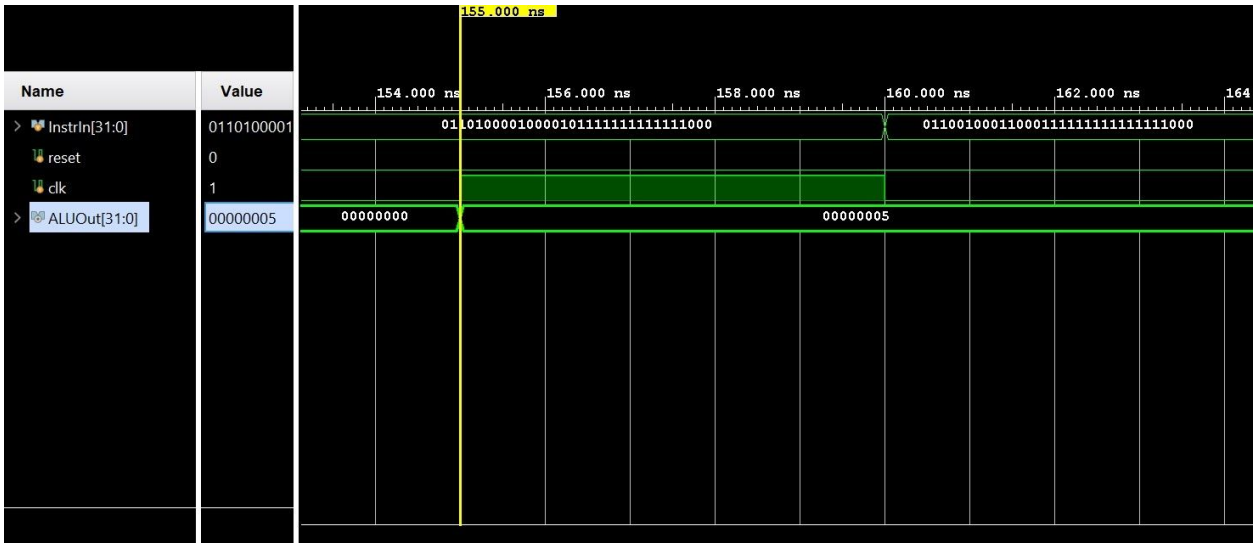

Name
Value
InstrIn[31:0]
0100100101
reset
0
clk
1
ALUOut[31:0]
00010002
234.000 ns
236.000 ns
238.000 ns
240.000 ns
242.000 ns
fffffff5
00010002


Name
Value
InstrIn[31:0]
0100100101
reset
0
clk
1
ALUOut[31:0]
00000009
244.000 ns
246.000 ns
248.000 ns
250.000 ns
252.000 ns
00010002
00000009


Name
Value
InstrIn[31:0]
0100100110
reset
0
clk
1
ALUOut[31:0]
0000aab4
254.000 ns
256.000 ns
258.000 ns
260.000 ns
262.000 ns
00000009
0000aab4


Name
Value
InstrIn[31:0]
0100100110
reset
0
clk
1
ALUOut[31:0]
0000000a
264.000 ns
266.000 ns
268.000 ns
270.000 ns
272.000 ns
0000aab4
0000000a


Name
Value
InstrIn[31:0]
0100100110
reset
0
clk
1
ALUOut[31:0]
0000000b
274.000 ns
276.000 ns
278.000 ns
280.000 ns
282.000 ns
0100100110000001001100000000000
0000000a
0000000b


Name
Value
InstrIn[31:0]
0110100001
reset
0
clk
1
ALUOut[31:0]
00000005
155.000 ns
154.000 ns
156.000 ns
158.000 ns
160.000 ns
162.000 ns
0110100001000010111111111111000
01100100011000111111111111111000
00000000
00000005