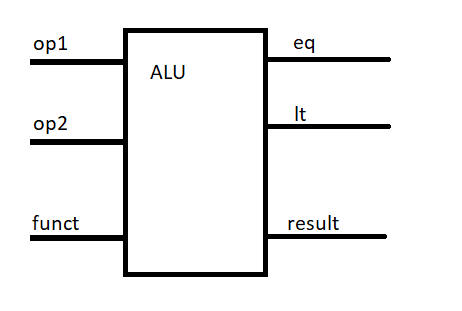
ALU

ll3374



## 1.1 Implementation

Signal list:

in op1: operand1

in op2: operand2

in funct: function selection

out eq: HIGH when op1 == op2, otherwise LOW

out lt: HIGH when op1 < op2, otherwise LOW

out result: function result

Function:

0: do result = op1 + op2

1: do result = op1 - op2

2: do result = op1 & op2

3: do result = op1 | op2

4: do result = !(op1 | op2)

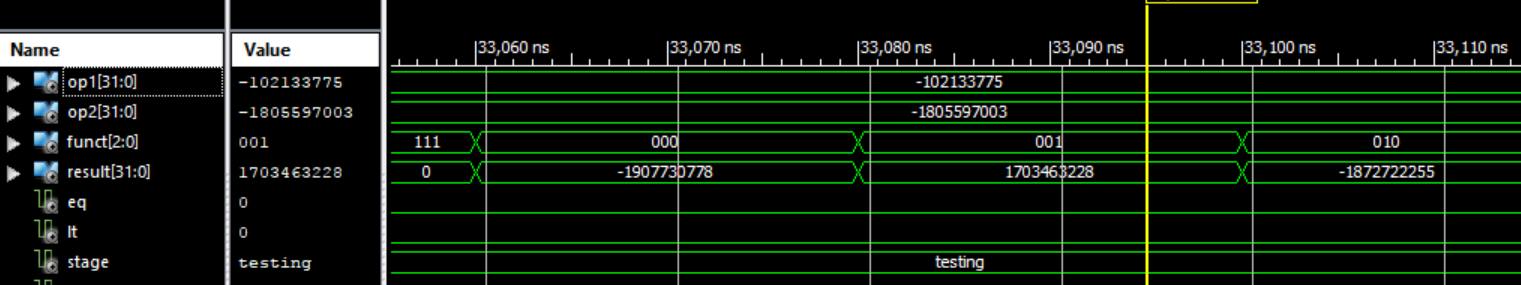
5: do result = op1 << op2

6: do result = op1 >> op2

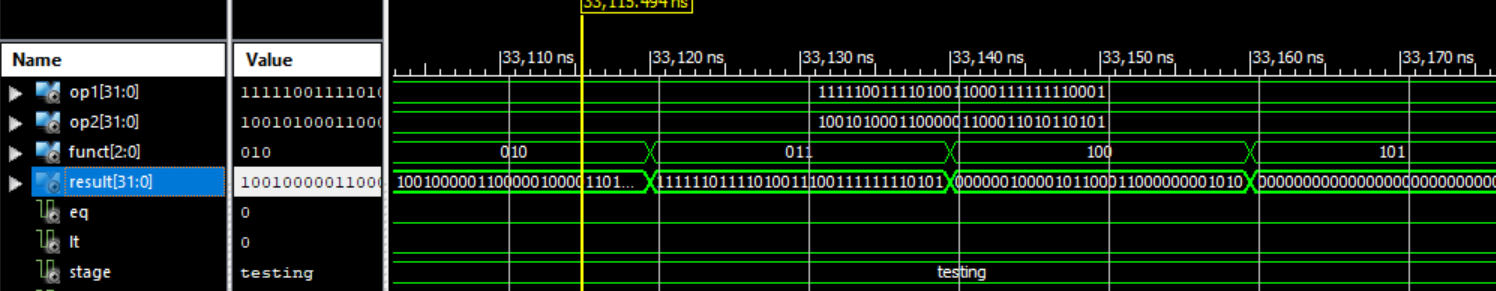
## 1.2 testbench

For testbench, I tested the ALU unit on 1000 random op1 and op2 for all the functions and used ‘assert’ statement to check the status automatically. The testbench would finish and show “1000 cases passed” only if all cases passed.

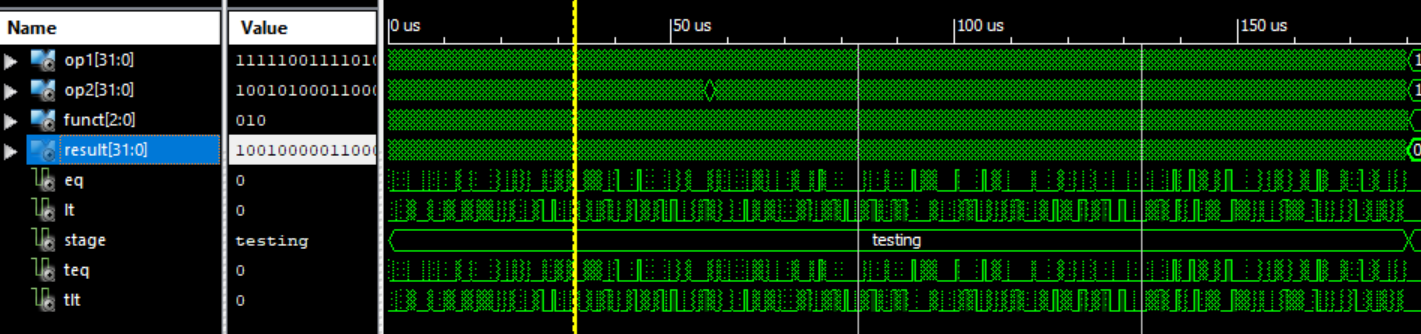
## 1.3 Function simulation:



1.3.1 case “+” and “-” showed on decimal

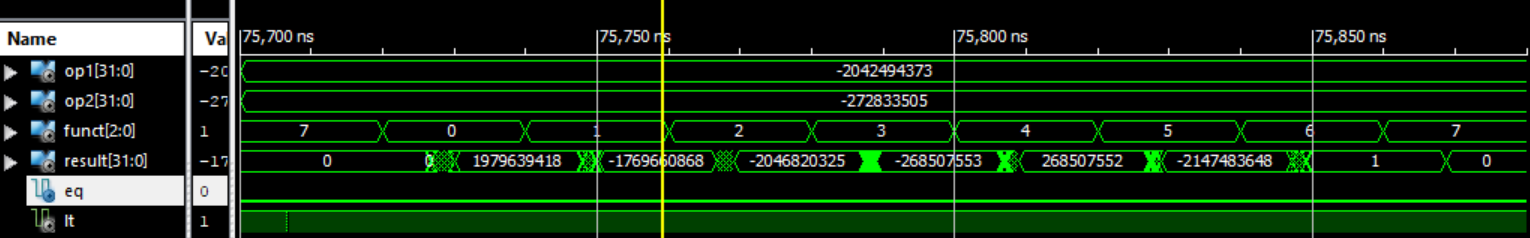


1.3.2 bit operation case showed on bits

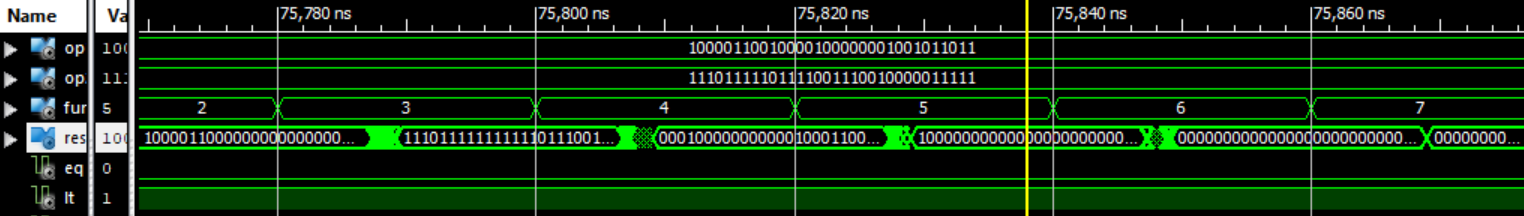


1.3.3 1 An overview. All cases passed

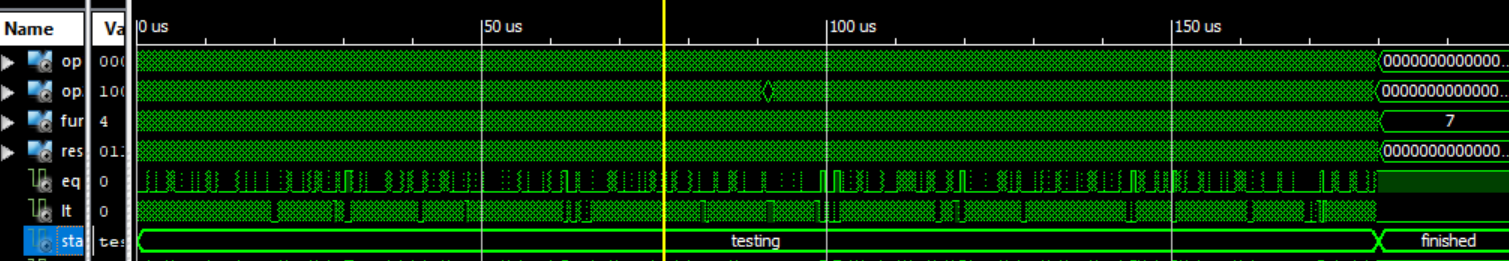
## 1.4 Timing simulation:



1.4.1 case “+” and “-” showed on decimal



1.4.2 bit operation case showed on bits



1.4.3 Overview. All cases passed

## 1.5 Timing analysis

|  |  |
| --- | --- |
| Critical path delay | 10.4 ns |
| Highest frequency | 96.2 MHz |