CODING PROJECT-1

CSN-221

Submitted by:

Gr12- Ritik Kumar (17114063) , Prateek Mali (17114059)
Gr87-Harshit Maurya (17114037) , Mularam Choudhary (17114049)

Calculator using Verilog:

This Coding Project is based on the given Problem:

"Design a calculator using Verilog (modular programming in hardware description language or HDL) with GUI."

Description :-

The verilog code have been designed using Xilinx ISE design suit and simulated in ISim simulator.

There are 3 main verilog files included in the .zip file, which are used to process the input and give the desired output.Along with them, there are files required for executing and rendering the GUI interface. The 3 verilog files are:

1. ALU.v

This is the central verilog file which receives the user input via a testbench, and processes a task based upon the control (*OPcode*) received. It also have a *SevenSegmentEncoder* task to convert the bcd output to the required 7-segment code. After this the output is returned via an *o* variable.
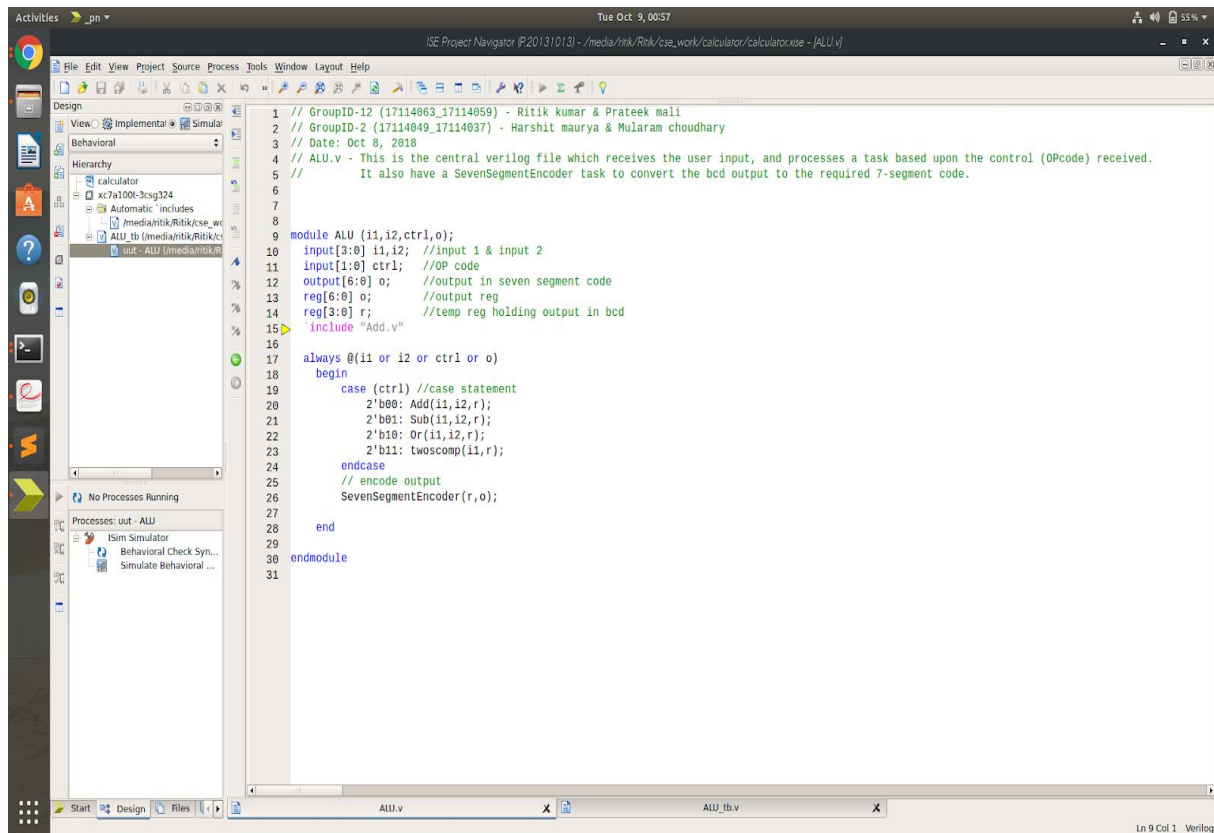
ALU.v file

2. ALU_tb.v

This is a *testbench* file which sits on the top of ALU.v file and it receives the user input via an '*input.txt*' file and writes the corresponding output in to the '*output.txt*' file.
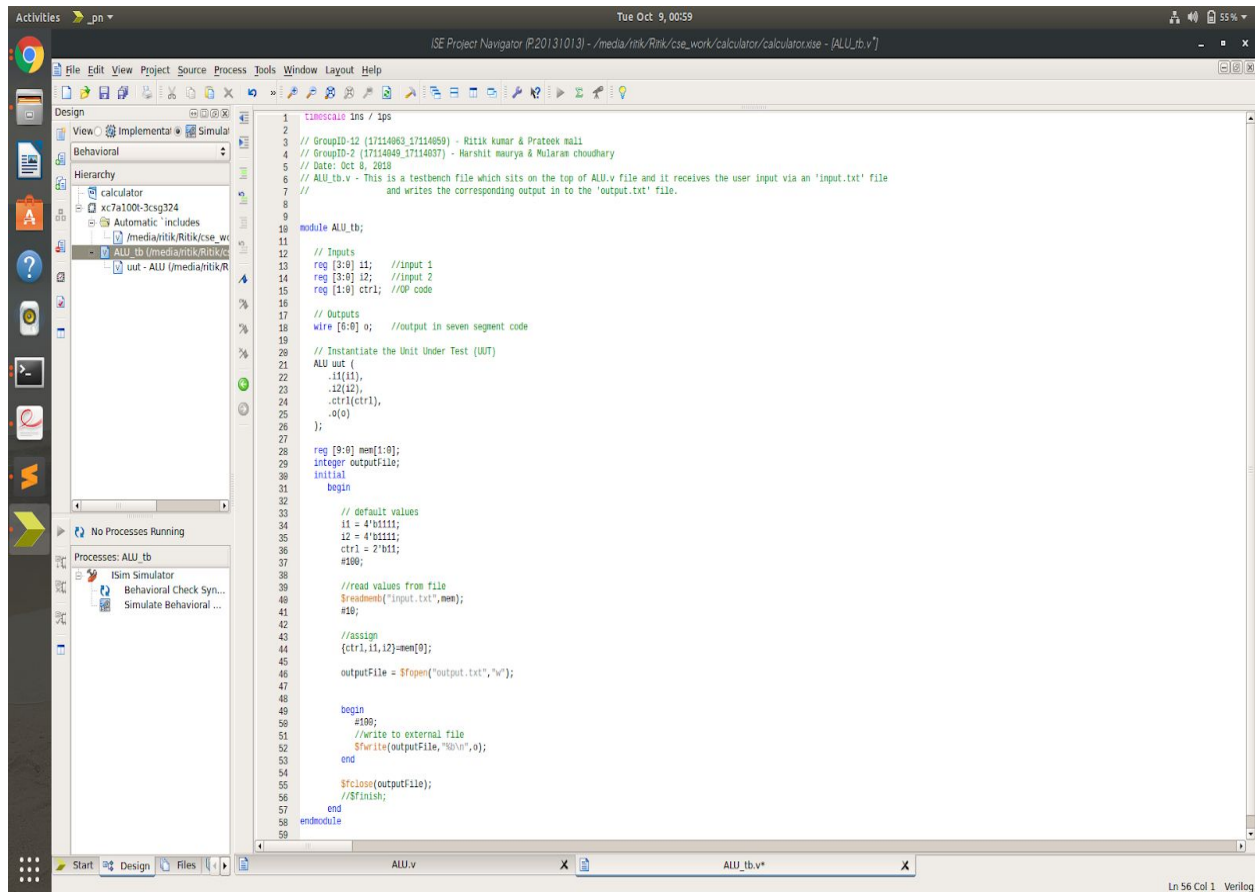
ALU_tb.v file

3. Add.v

This is the verilog file which executes all the tasks required by ALU.v file. This consists of 5 tasks:
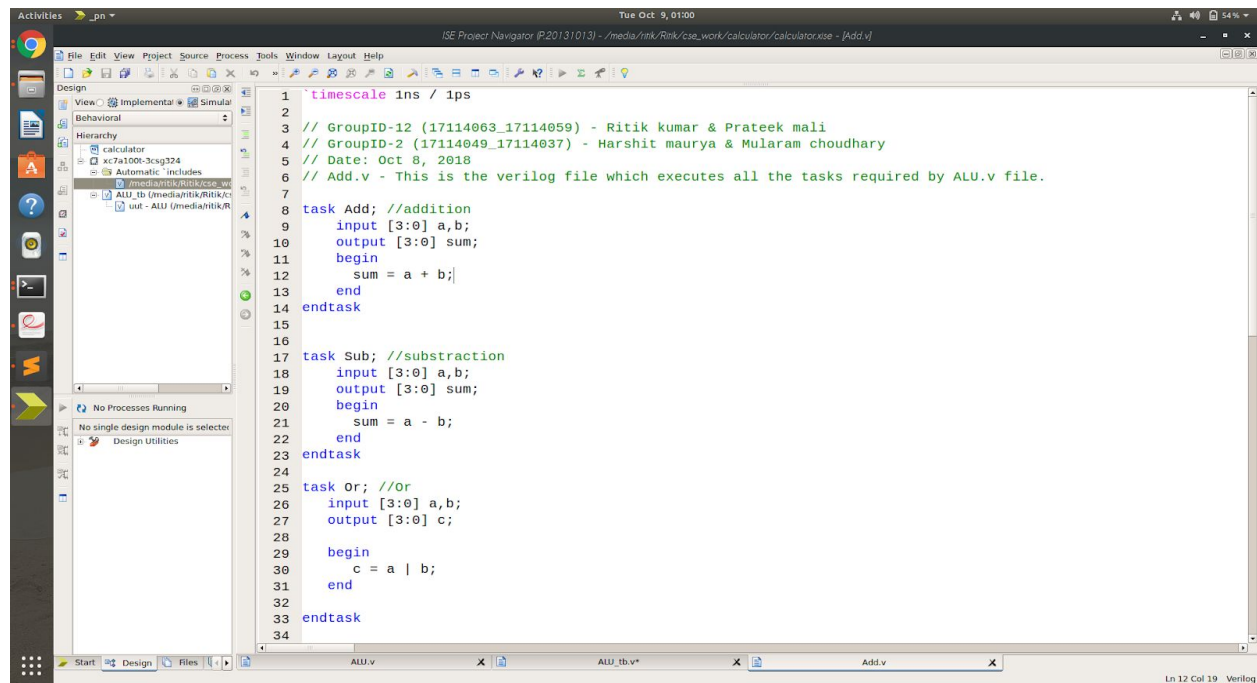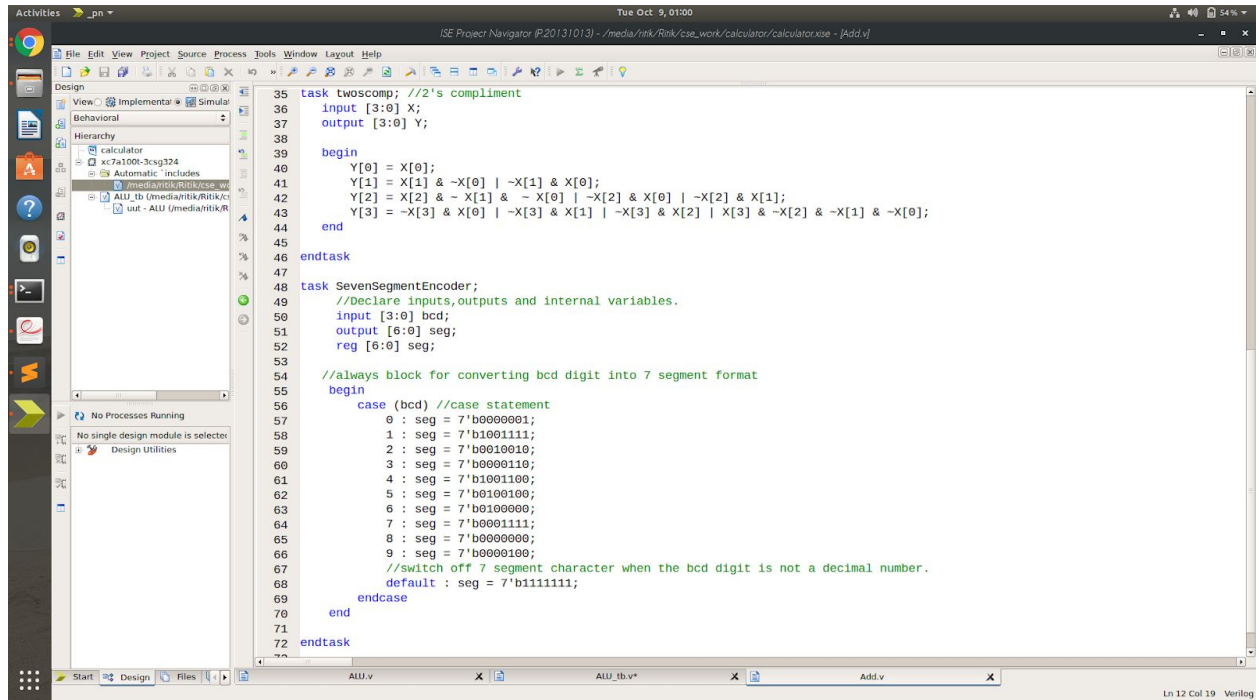
I. Add

Ii. Sub

Iii.Or

iv.Two'scomp

V. Seven Segment Encoder

The names here are self explanatory and sufficient comments have been added wherever required.

Each task has input and output variables and reg (if required). They have begin and end statements between which the execution occurs.
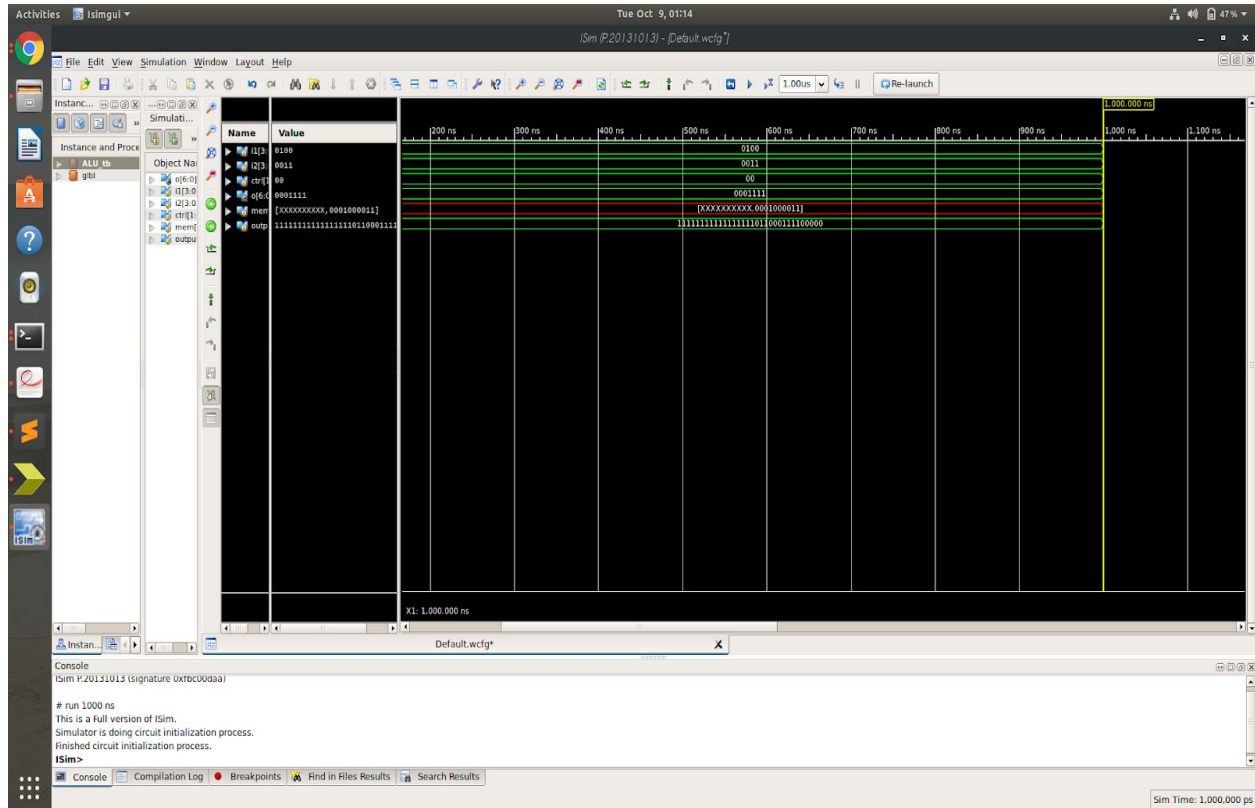
ISE Project Navigator (P.20131013) - /media/ritik/Ritik/cse_work/calculator/calculator.xise - [Add.v]

File Edit View Project Source Process Tools Window Layout Help

```verilog
35  task twoscomp; //2's compliment
36      input [3:0] X;
37      output [3:0] Y;
38
39  begin
40      Y[0] = X[0];
41      Y[1] = X[1] & ~X[0] | ~X[1] & X[0];
42      Y[2] = X[2] & ~ X[1] & ~ X[0] | ~X[2] & X[0] | ~X[2] & X[1];
43      Y[3] = ~X[3] & X[0] | ~X[3] & X[1] | ~X[3] & X[2] | X[3] & ~X[2] & ~X[1] & ~X[0];
44      end
45
46  endtask
47
48  task SevenSegmentEncoder;
49      //Declare inputs,outputs and internal variables.
50      input [3:0] bcd;
51      output [6:0] seg;
52      reg [6:0] seg;
53
54  //always block for converting bcd digit into 7 segment format
55    begin
56      case (bcd) //case statement
57          0 : seg = 7'b0000001;
58          1 : seg = 7'b1001111;
59          2 : seg = 7'b0010010;
60          3 : seg = 7'b0000110;
61          4 : seg = 7'b1001100;
62          5 : seg = 7'b0100100;
63          6 : seg = 7'b0100000;
64          7 : seg = 7'b0001111;
65          8 : seg = 7'b0000000;
66          9 : seg = 7'b0000100;
67          //switch off 7 segment character when the bcd digit is not a decimal number.
68          default : seg = 7'b1111111;
69      endcase
70    end
71
72  endtask
```

Hierarchy

- calculator
  - xc7a100t-3csg324
    - Automatic `includes
      - /media/ritik/Ritik/cse_w
    - ALU_tb (/media/ritik/Ritik/c
      - uut - ALU (/media/ritik/R

No Processes Running

No single design module is selected
  Design Utilities

Start   Design   Files   ALU.v   ALU_tb.v*   Add.v   Ln 12 Col 19 Verilog

Add.v file

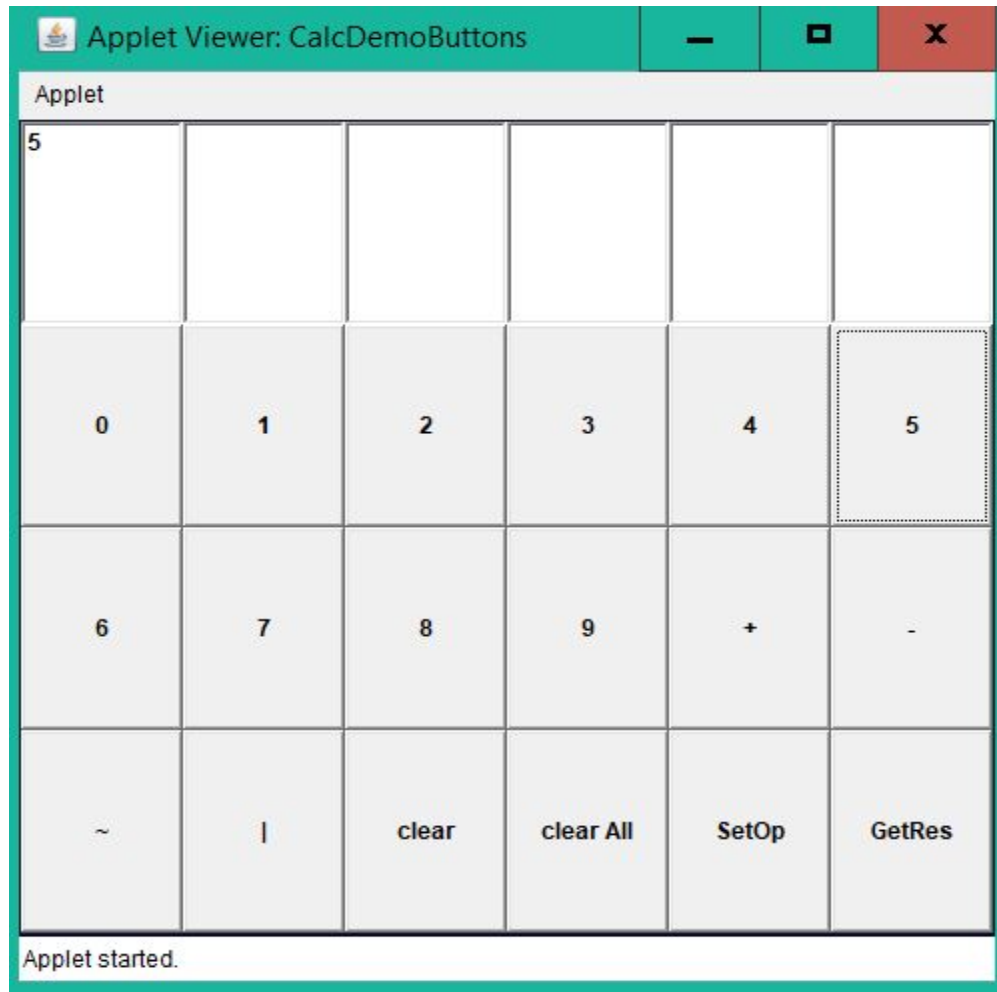# SCREENSHOT OF WAVEFORMS

ADD (waveforms)

SUB (waveforms)
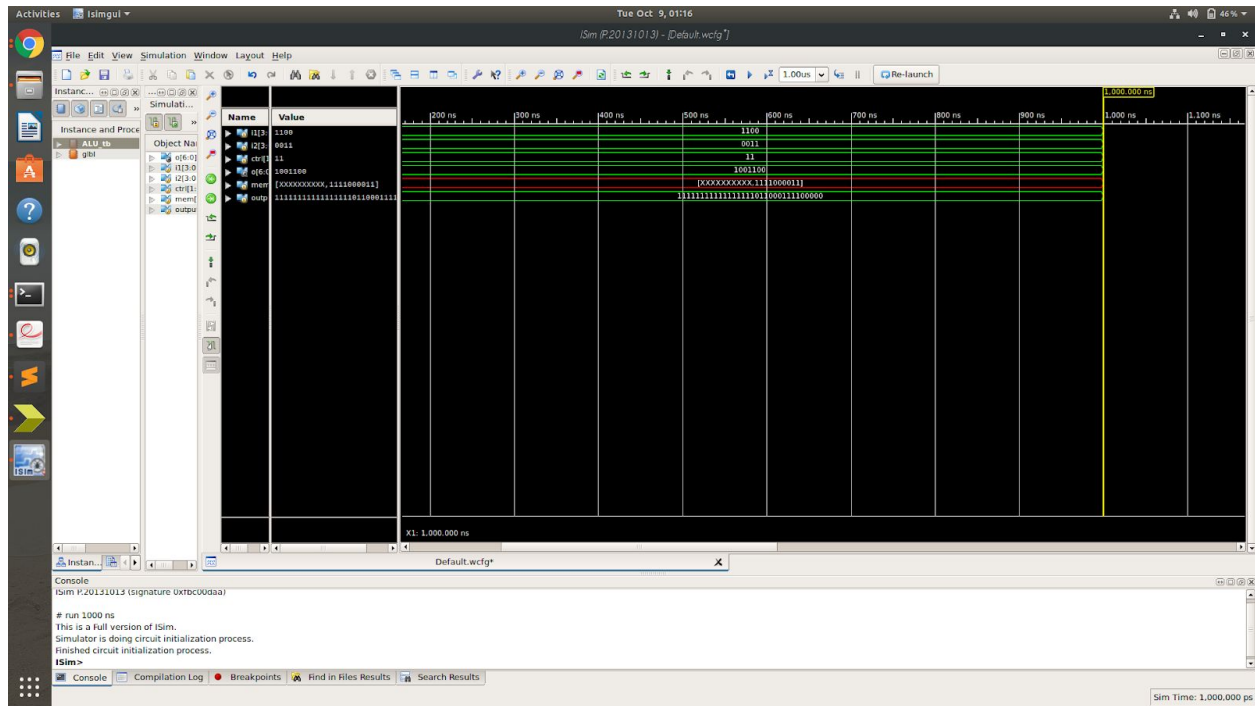
OR (waveforms)

Two's complement(waveforms)

CALCULATOR GUI

How to run GUI Applet:

1. Navigate to the root folder.
2. `javac CalcDemoButtons.java`.
3. `appletviewer Applet.html`.
4. Enter the numbers and operations as we do in normal calculators.
5. Press SetOp.
6. Head over to the Xilinx program and simulate it.
7. This will generate output in the file `Output.txt`.

8. Press GetRes button on the calculator to get decimal output in GUI.