EE533 Network Processor Design & Programming

Lab #2: Designing of 32-bit ALU using Xilinx ISE 10.1

Instructor: Prof. Young Cho, PhD

Name: Archit Sethi

University of Southern California

Los Angeles, CA 90007

GITHUB LINK to my REPOSITORY

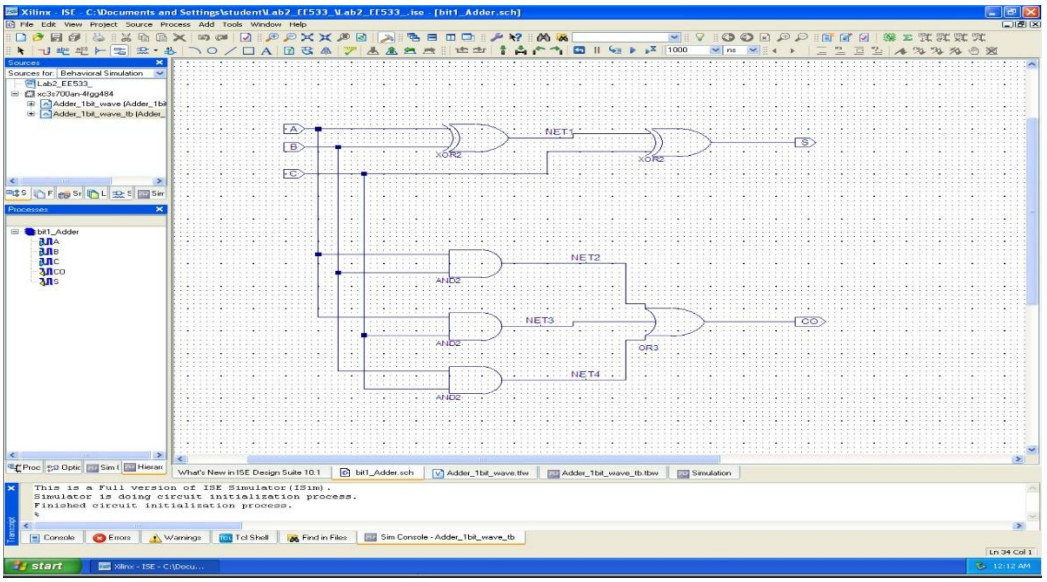
Aarch0811/EE533/LAB2/

Archit's GITHUB: LINK

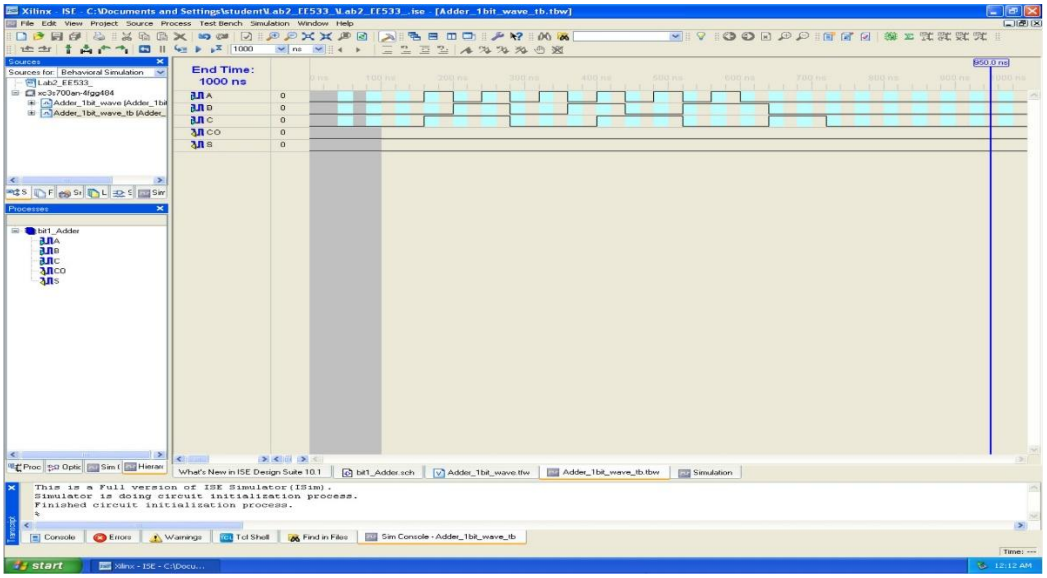
1. Designing and Simulating of an 8-bit Synchronous Adder
    a. 1-bit Adder

Implemented the 1-bit Adder using the basic gates from the Xilinx Library ash shown in the schematic.

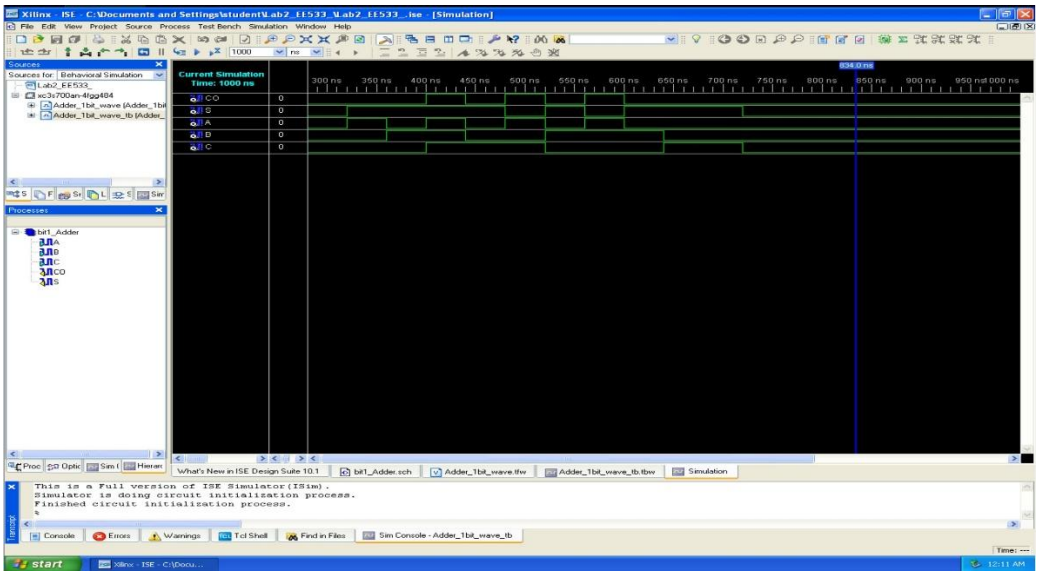
i. Schematic



ii. Testbench



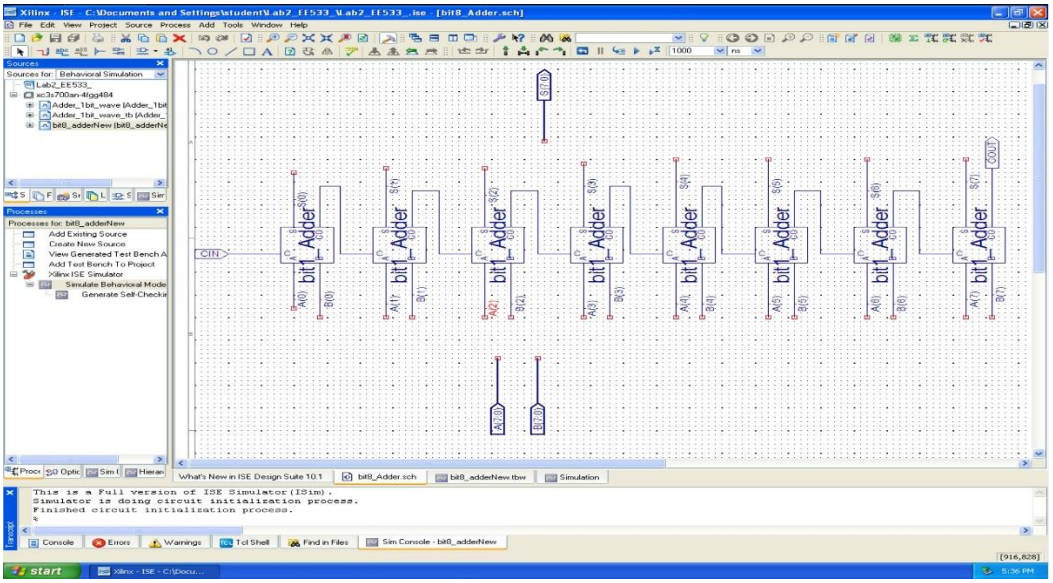
iii. Waveform/Simulation



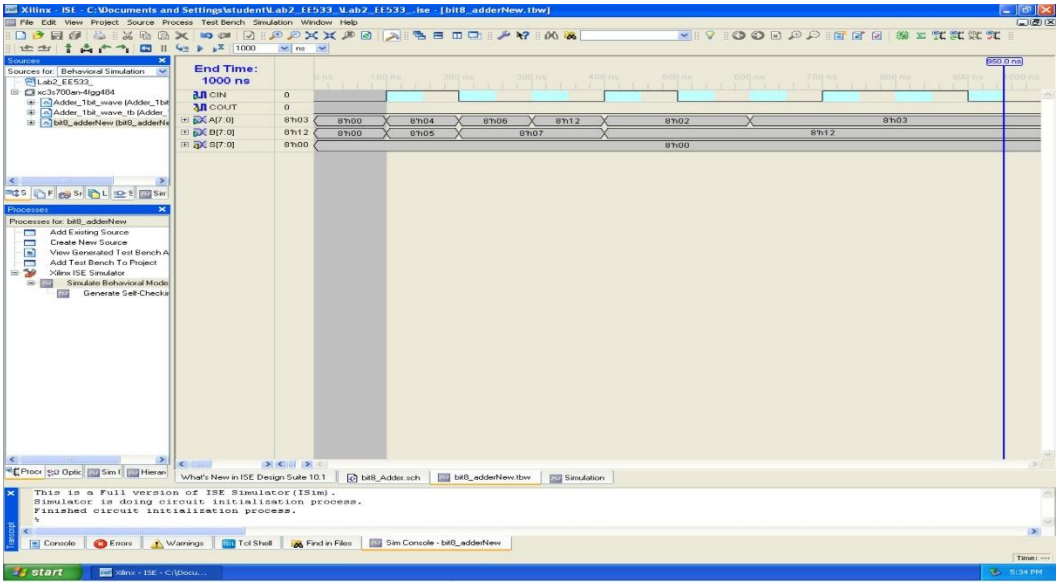
b. 8-bit Adder

This adder was designed by instantiating 8 x 1-bit Adders in cascaded fashion and made connections in a 8-bit bus.

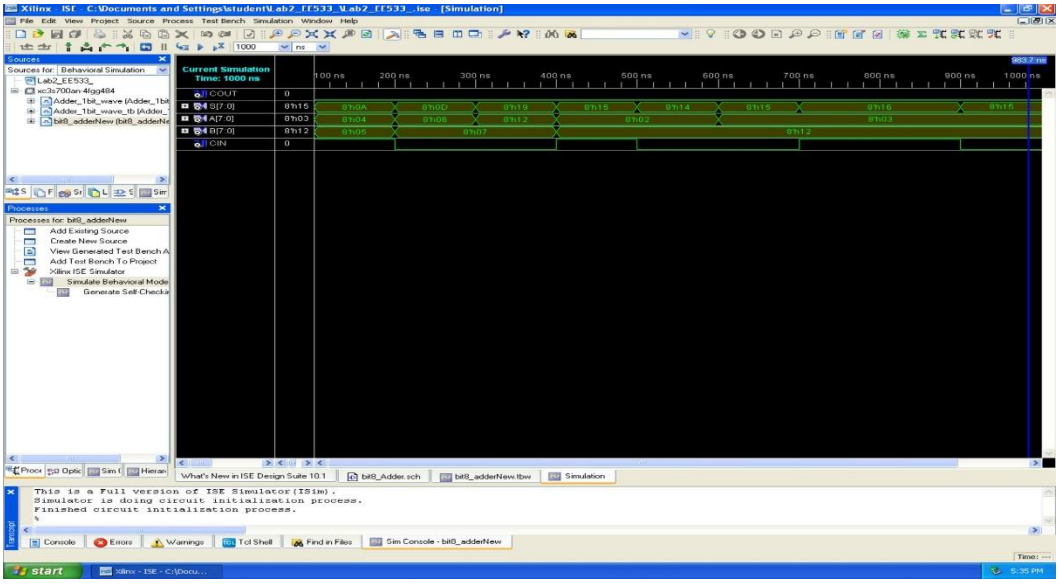
i. Schematic



ii. Testbench



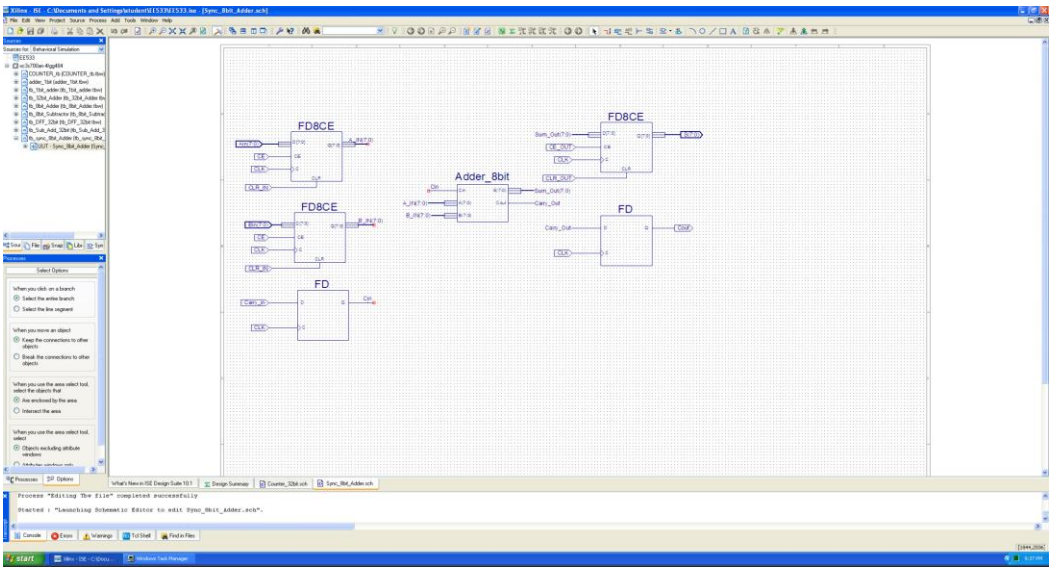
iii. Waveform/Simulation



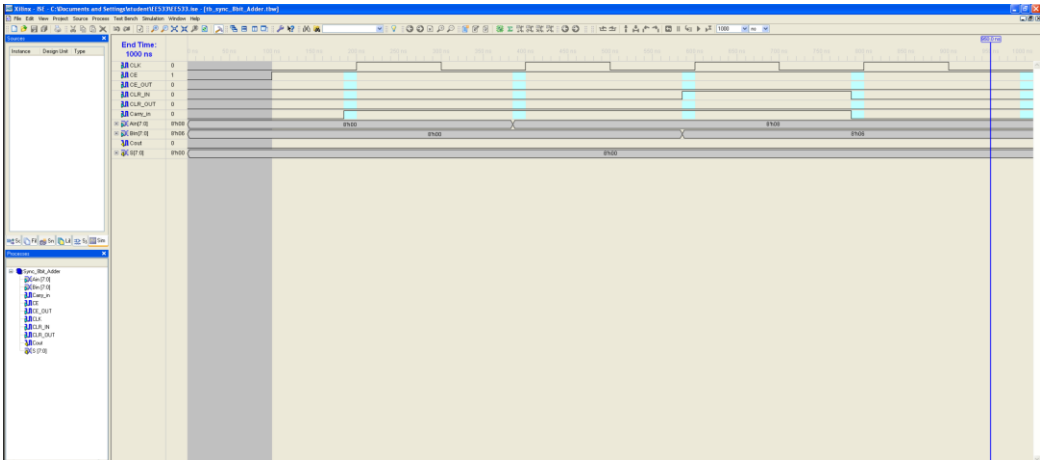
c. 8-bit Synchronous Adder

A CLK was added to the 8-bit adder circuit as shown in the schematic below.

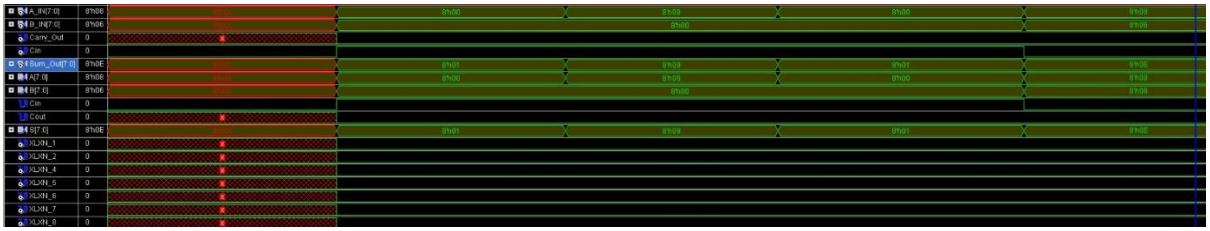
i. Schematic

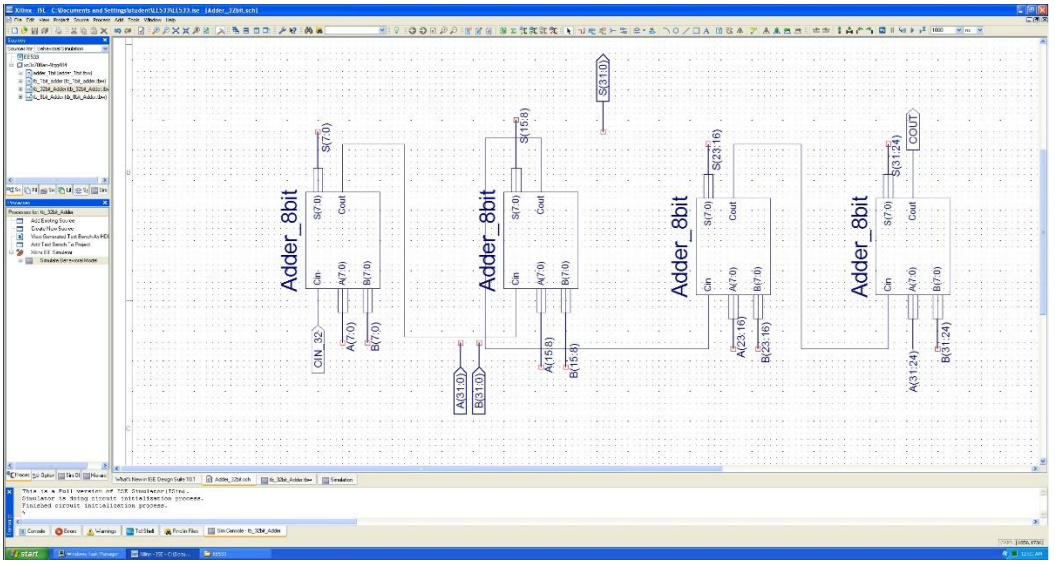


ii. Testbench

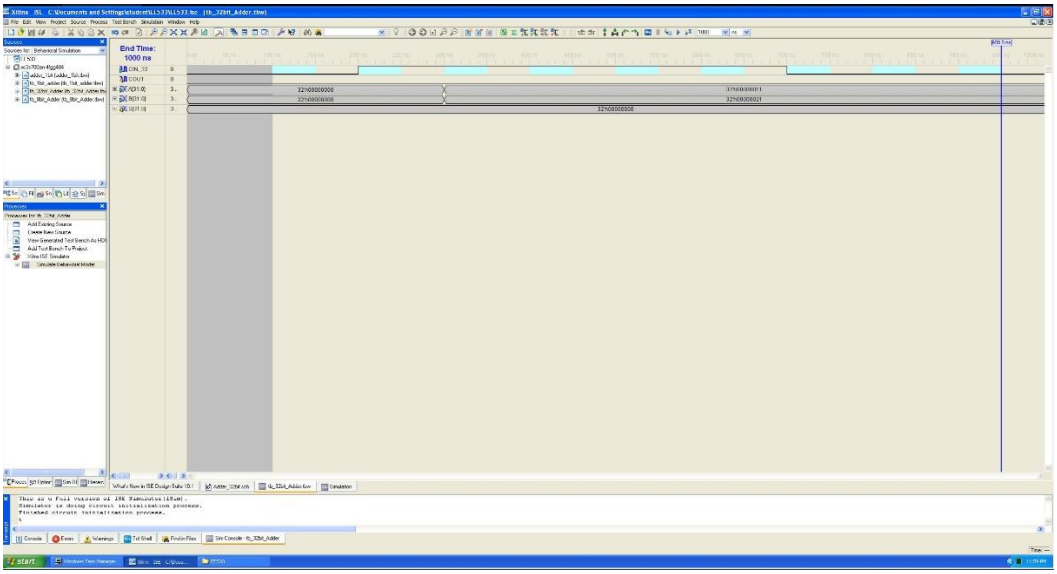


iii. Waveform/Simulation

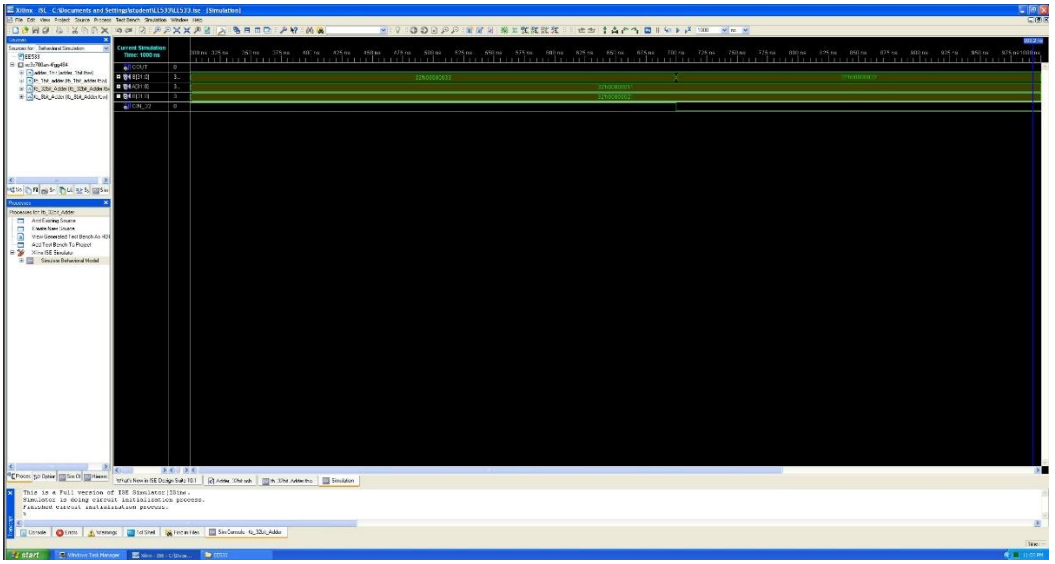




ii. Testbench

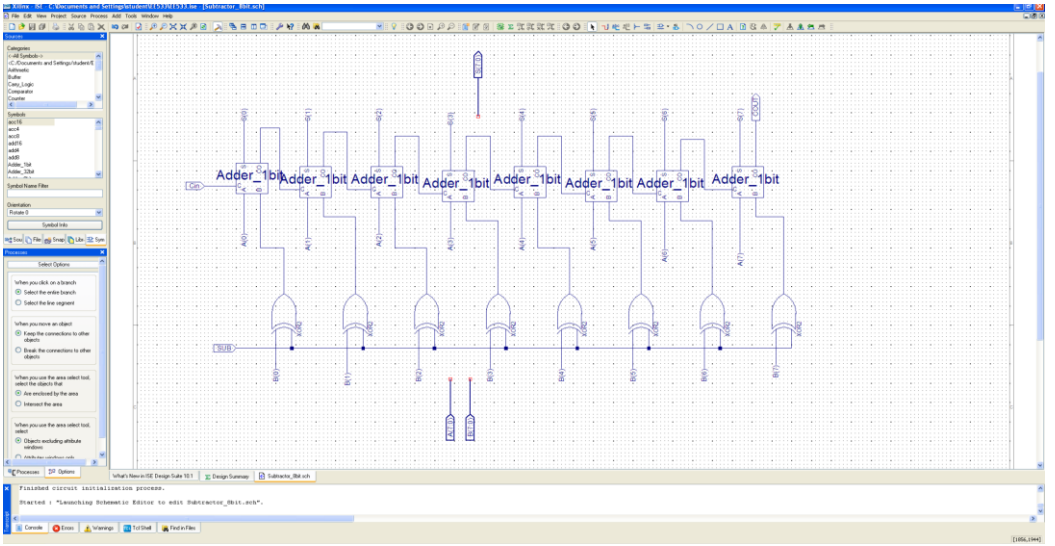


iii. Waveform/Simulation

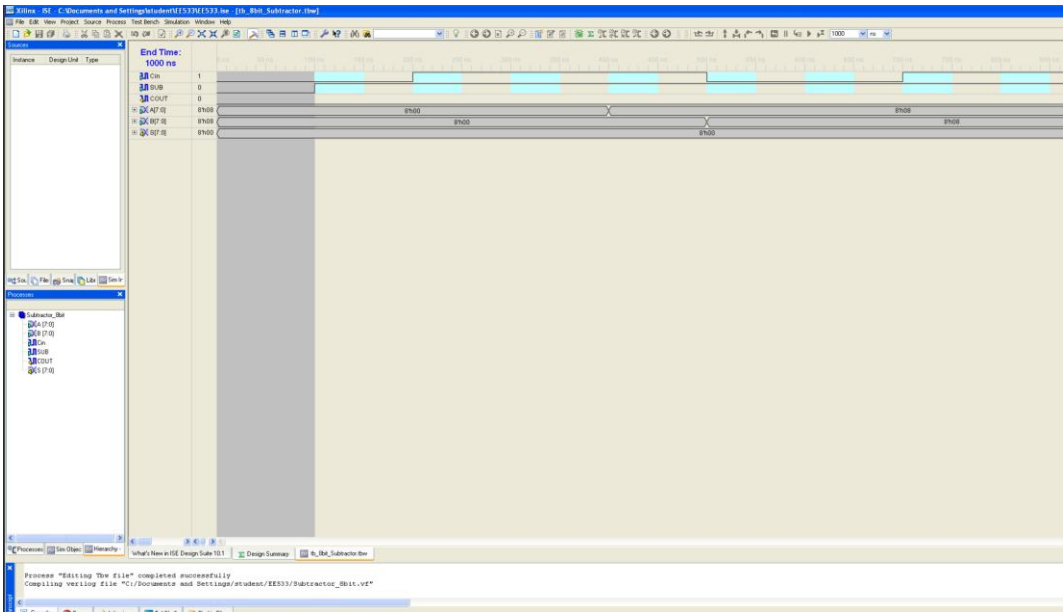


e. 32-bit Subtractor

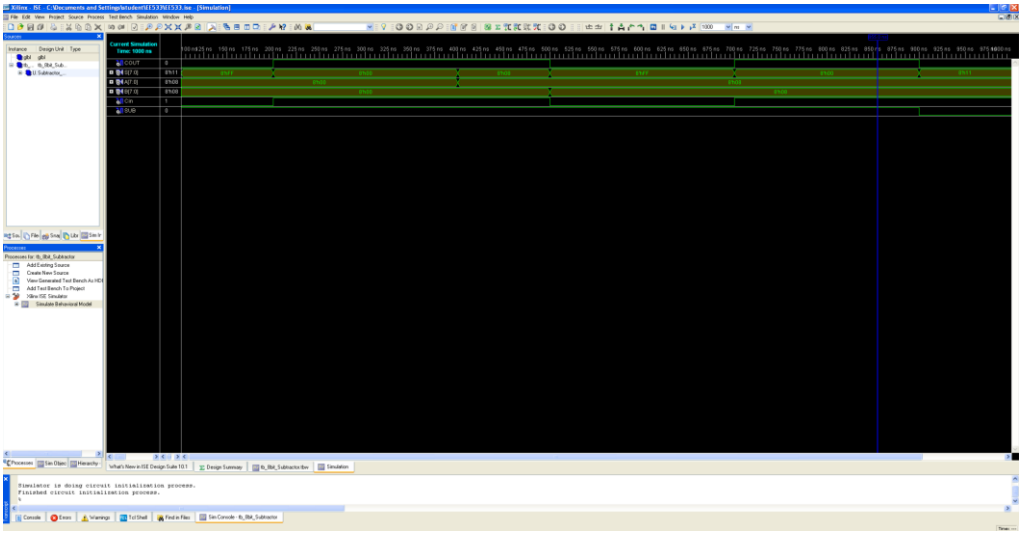
i. Schematic



ii. Testbench

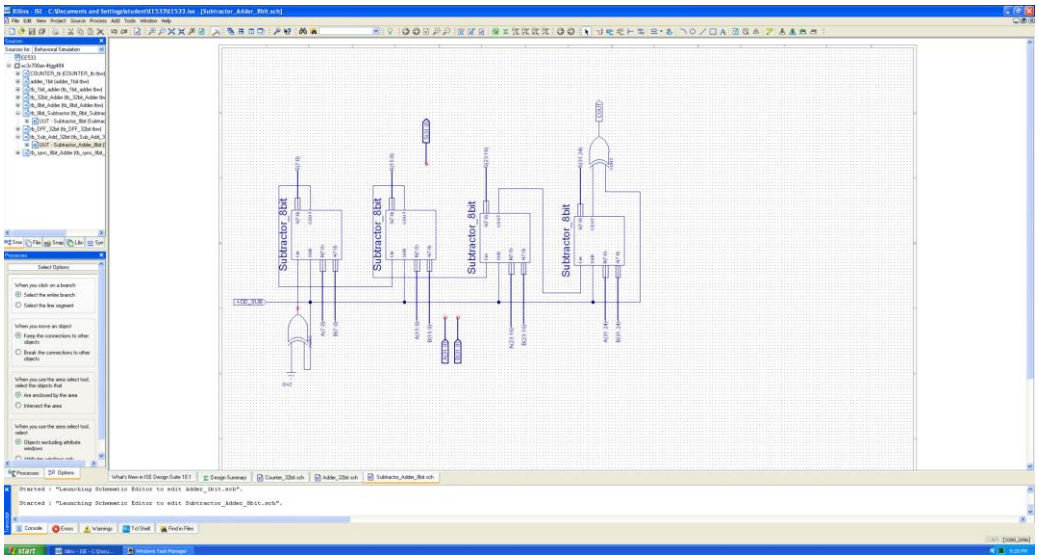


iii. Waveform

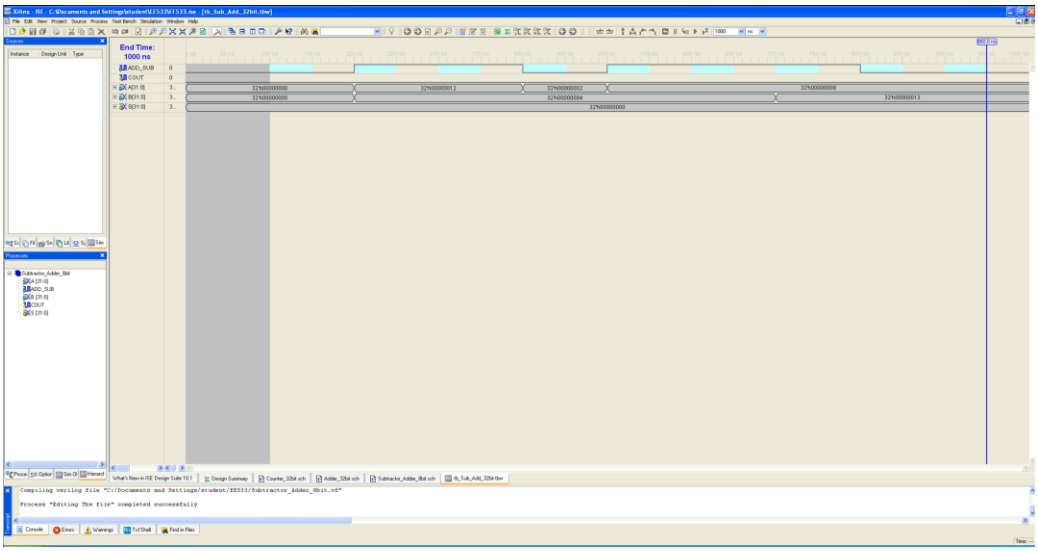


f. 32-bit Subtractor/Adder: This is the schematic design for 32-bit adder/subtractor circuit which implements both addition and subtraction using the 1-bit bus line labelled as ‘SUB_ADD’

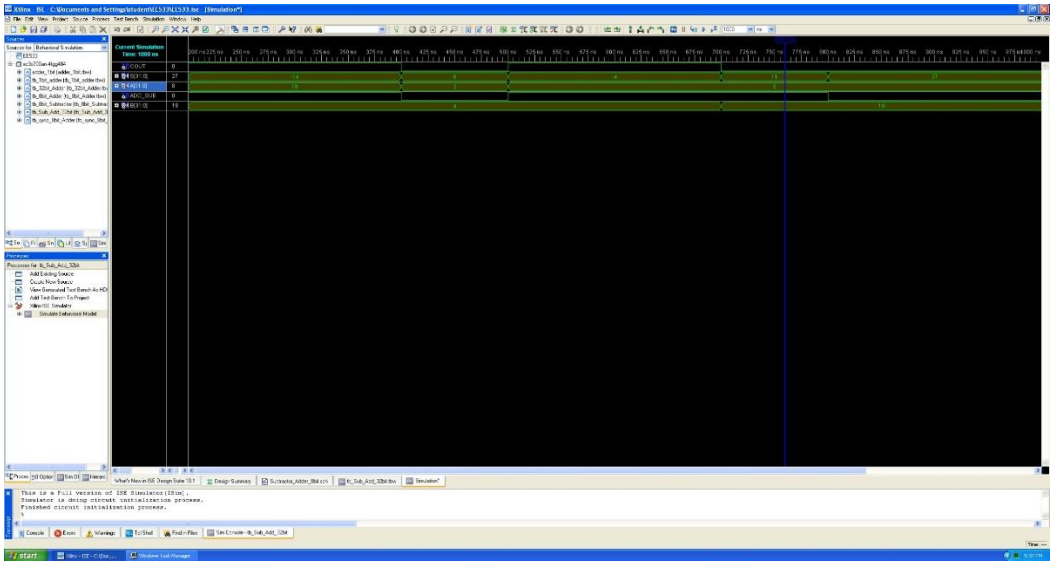
i. Schematic



ii. Testbench

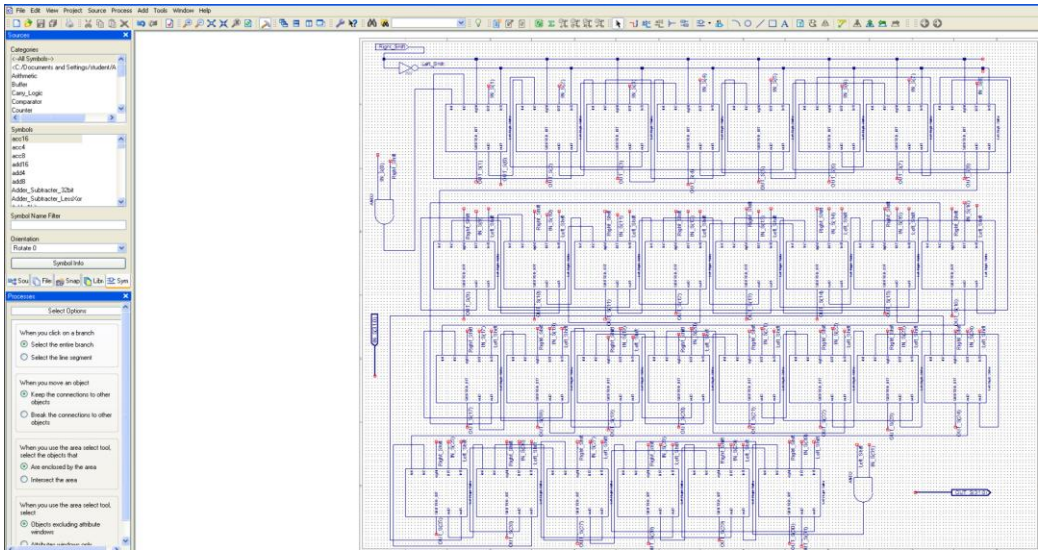


iii. Waveform/Simulation

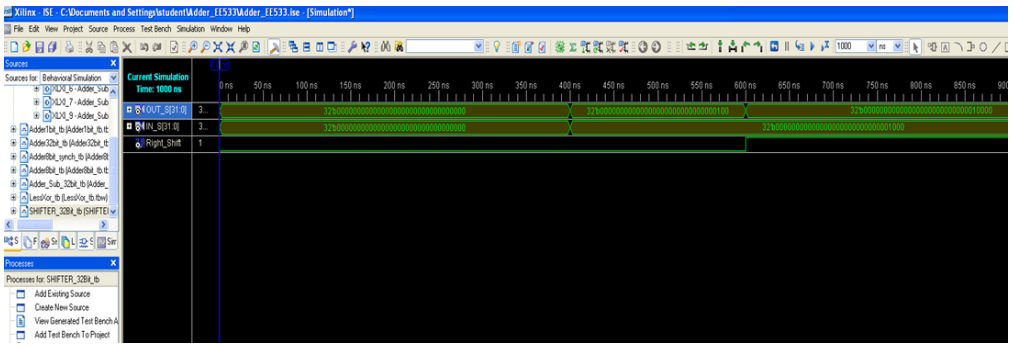


g. Left-Right 32-bit Shifter: This SHIFTER circuit performs both Left and Right shifting as shown in the schematic module below.

i. Schematic



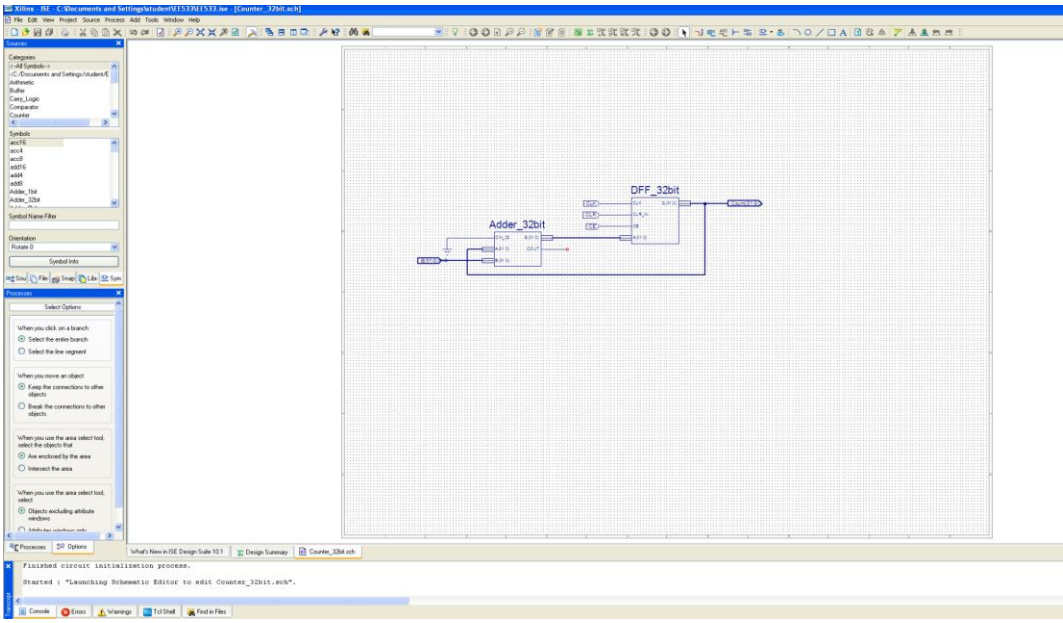
ii. Waveform/Simulation



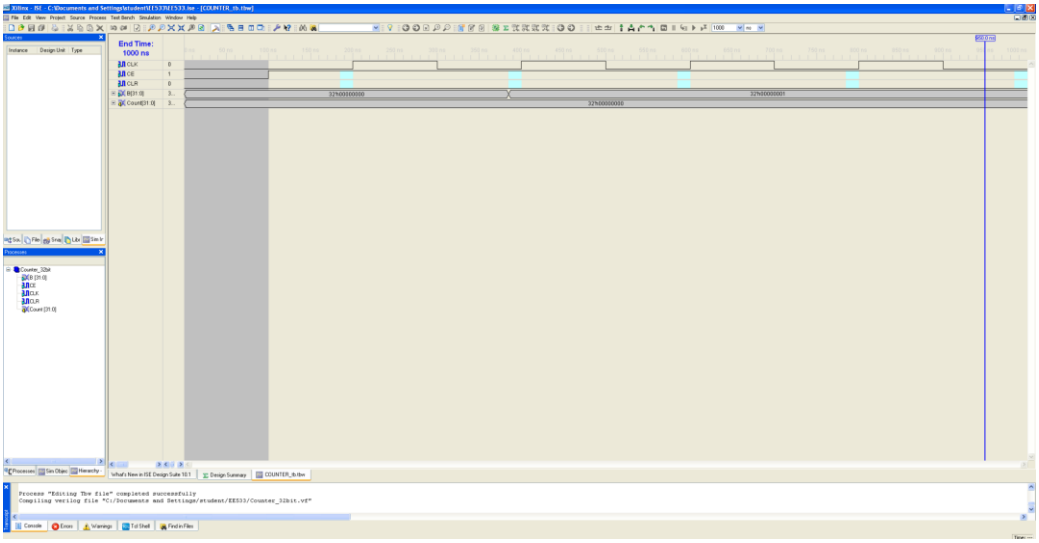
h. Multiplexer 32-bit symbol for 32-bit ALU: This multiplexer (MUX 32-bit (2-bit wide)) is used in the ALU circuit. The following symbol was used in the design.

i. 32-bit Counter: This is the functionality of the 32-bit ALU, where I've incorporated a 32-bit COUNTER circuit.

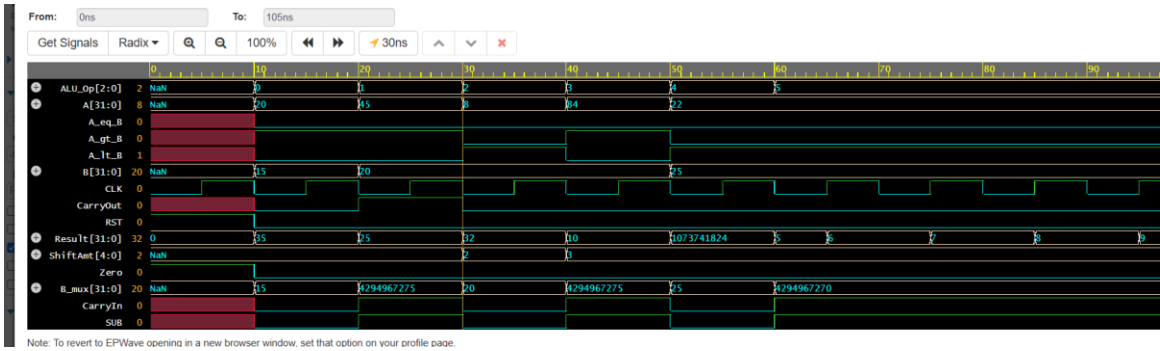
i. Schematic



ii. Testbench



2. Verilog code equivalent for 32-bit ALU
    a. Screen capture of the waveform generated by behavioural simulation tools



    b. Log file of the mapper

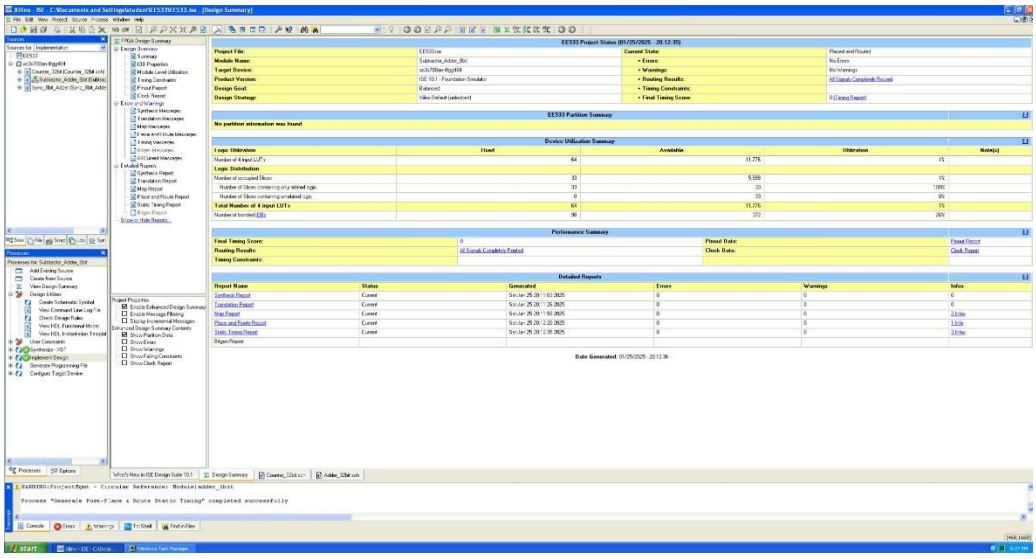
```
Design Information
------------------
Command Line   : map -ise "C:/Documents and
Settings/VerilogLab2/VerilogLab2.ise" -intstyle ise -p xc3s700a-fg484-4 -cm area
-pr off -k 4 -c 100 -o ALU_32bit_top_map.ncd ALU_32bit_top.ngd ALU_32bit_top.pcf

Target Device  : xc3s700a
Target Package : fg484
Target Speed   : -4
Mapper Version : spartan3a -- $Revision: 1.46 $
Mapped Date    : Sat Jan 25 21:43:21 2025

Design Summary
--------------
Number of errors:      0
Number of warnings:    0
Logic Utilization:
  Number of 4 input LUTs:                98 out of  11,776    1%
Logic Distribution:
  Number of occupied Slices:             49 out of   5,888    1%
    Number of Slices containing only related logic:     49 out of      49 100%
    Number of Slices containing unrelated logic:         0 out of      49   0%
      *See NOTES below for an explanation of the effects of unrelated logic.
  Total Number of 4 input LUTs:          98 out of  11,776    1%
  Number of bonded IOBs:                101 out of     372   27%

Peak Memory Usage:  158 MB
Total REAL time to MAP completion:  6 secs
Total CPU time to MAP completion:   5 secs
```

Design Implementation:



    c. Brief comment on the number of gates as compared to the schematic version

The number of gates compared to the actual design version is less than the schematic design implementation. This is because the schematic uses more GATES (due to DFFs & LUTs explicitly). Whereas the Verilog based design

is synthesized into an optimized circuit, which includes less number of gates due to the synthesizer optimizer logic.