**Lab 10: An Introduction to High-Speed Addition**

Faizan Bangash

TA : Saurav Kumar Sahu

ECEN 248 – 302

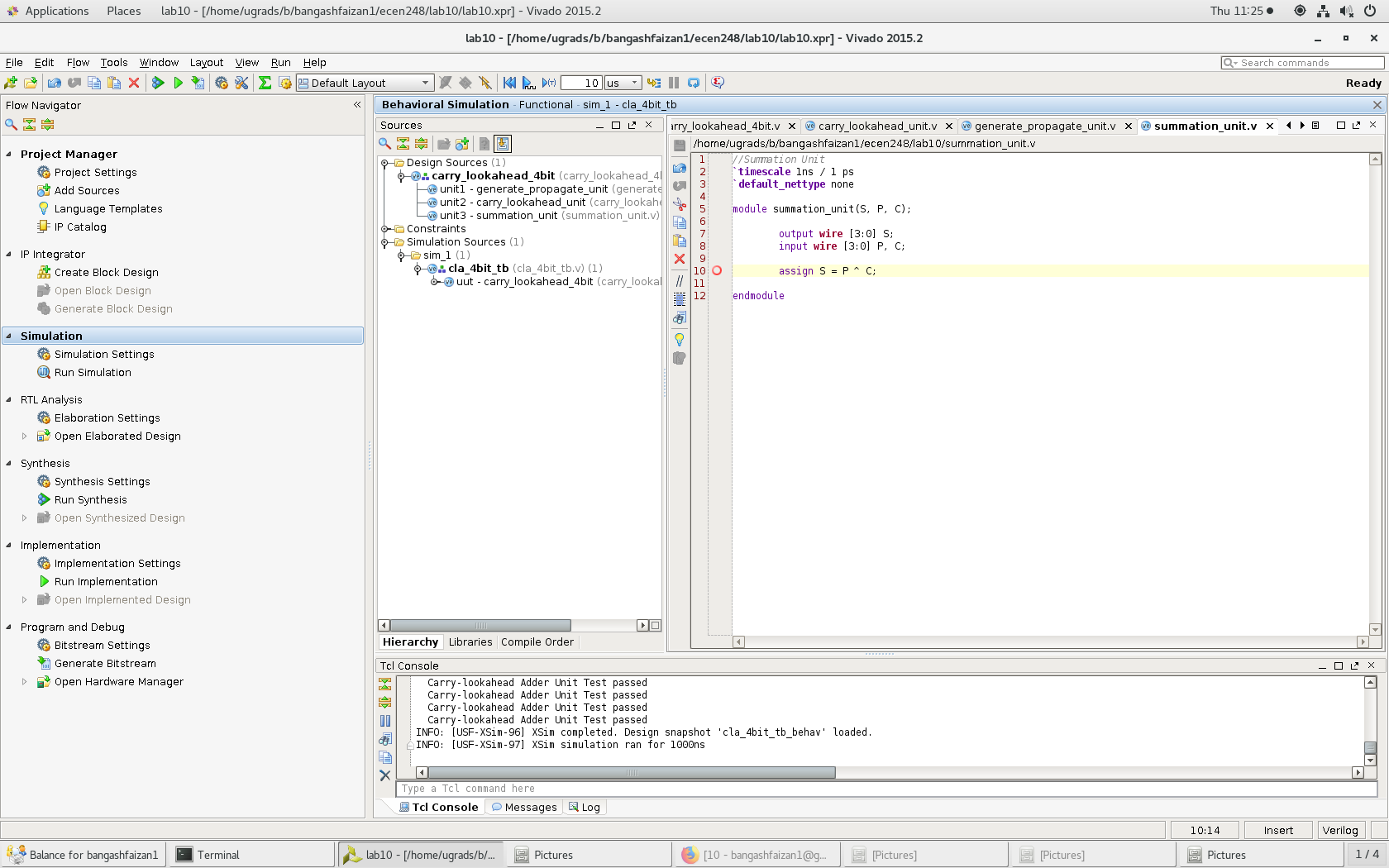
July 31st, 2018.

**Objectives** –

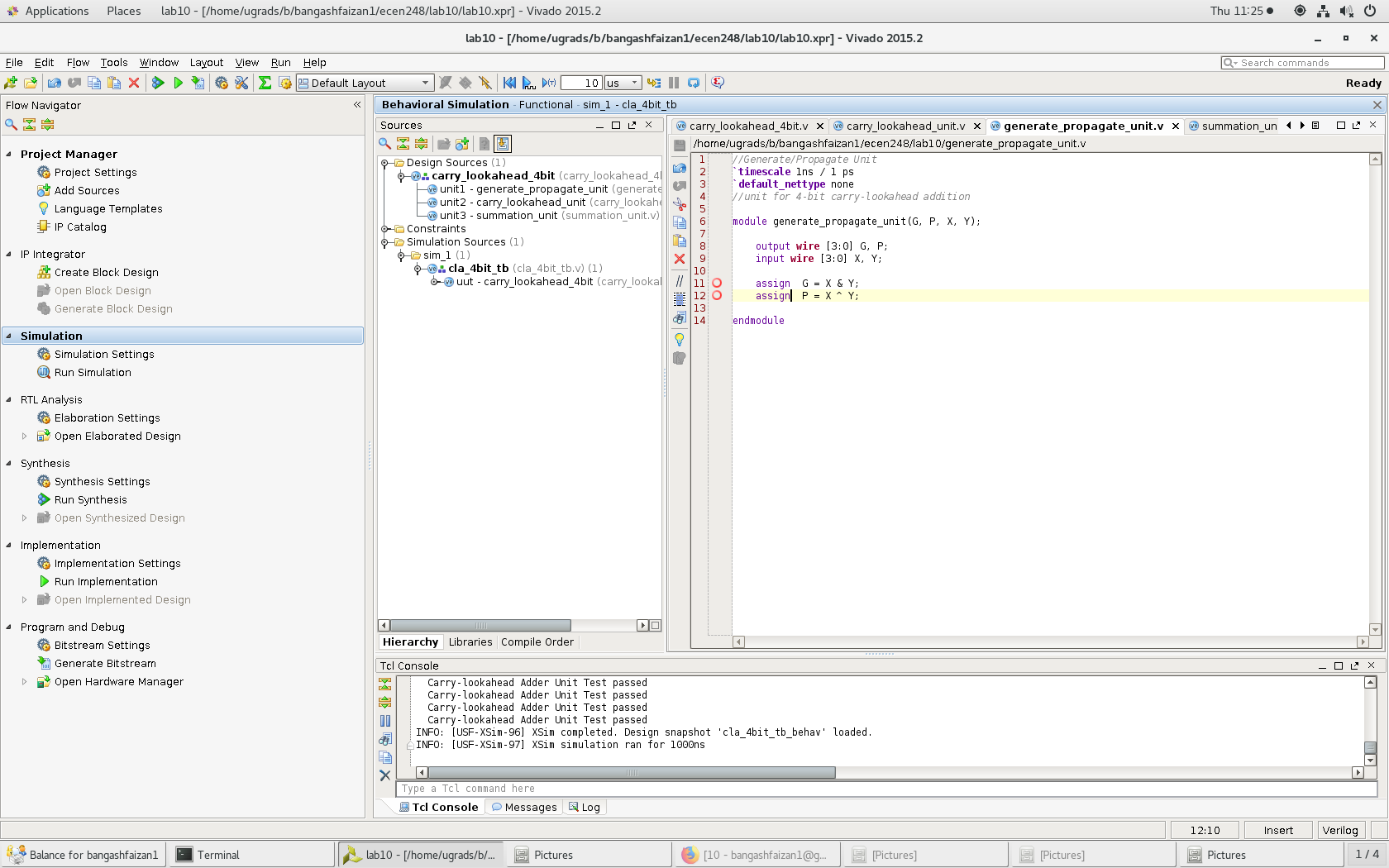
This lab emphasizes that previously made circuits such as the ripple carry adder are not sufficient for high speed operations. The delays in the old circuits start to add up as we add more numbers so it is not practical. Because of this, we make the carry-lookahead adder that mixes dataflow and structural coding in verilog.

**Design**:

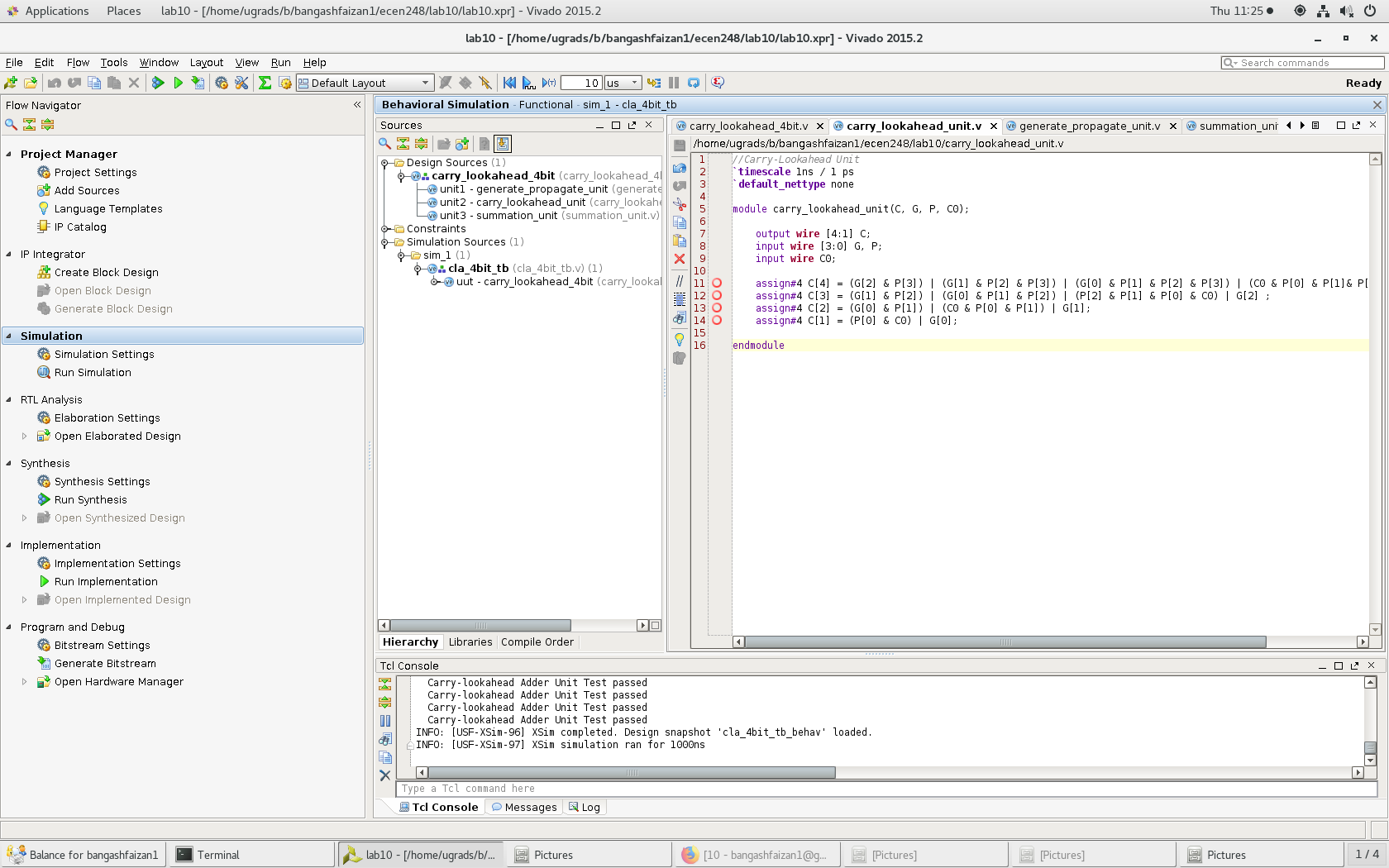
1. In Experiment 1, we tested the programs written in the pre-lab seen below. However, in part 2 of this experiment, we added delays to the sub-modules. These are included below instead of the original pre-lab code.



Summation unit

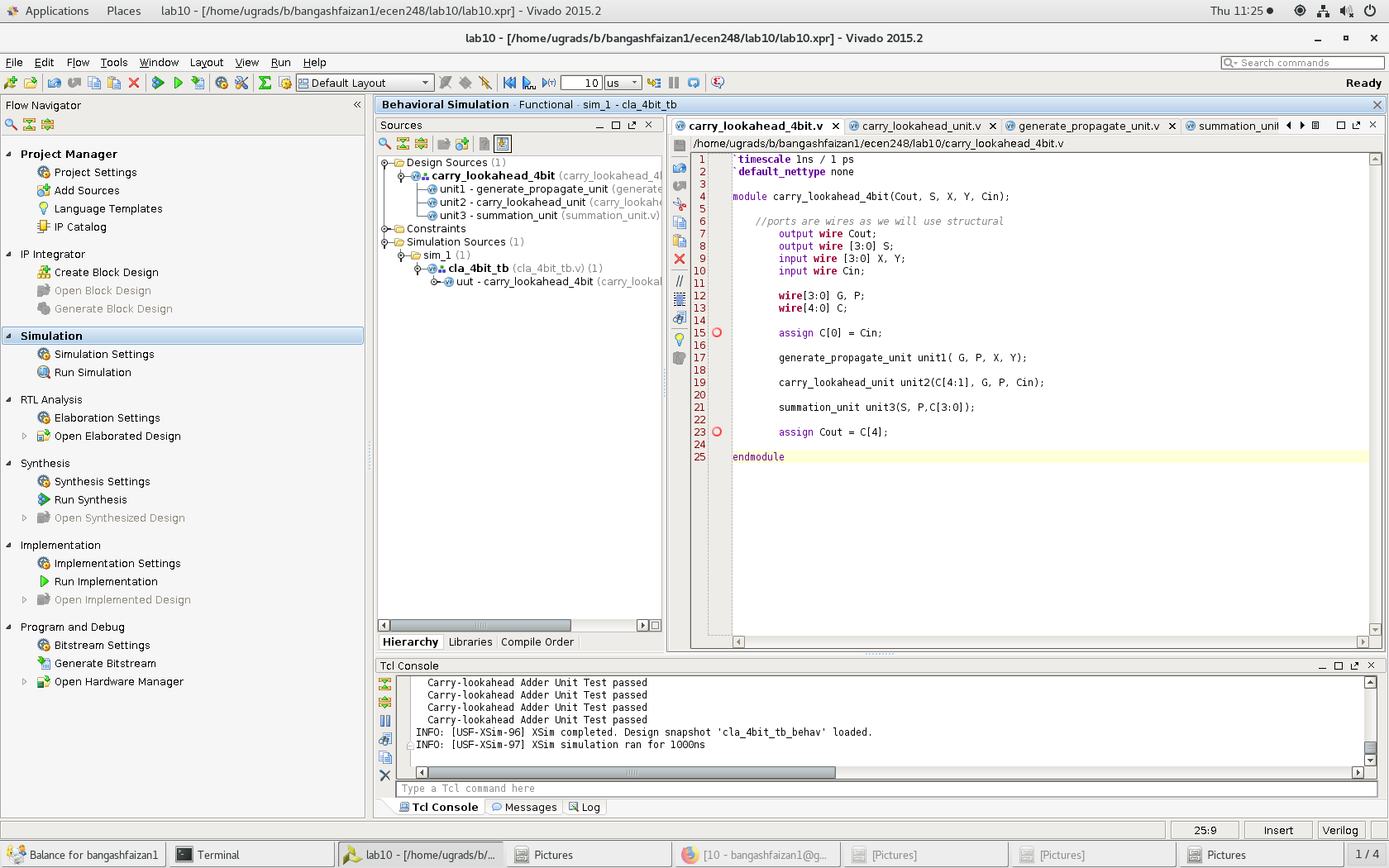


Propagate unit



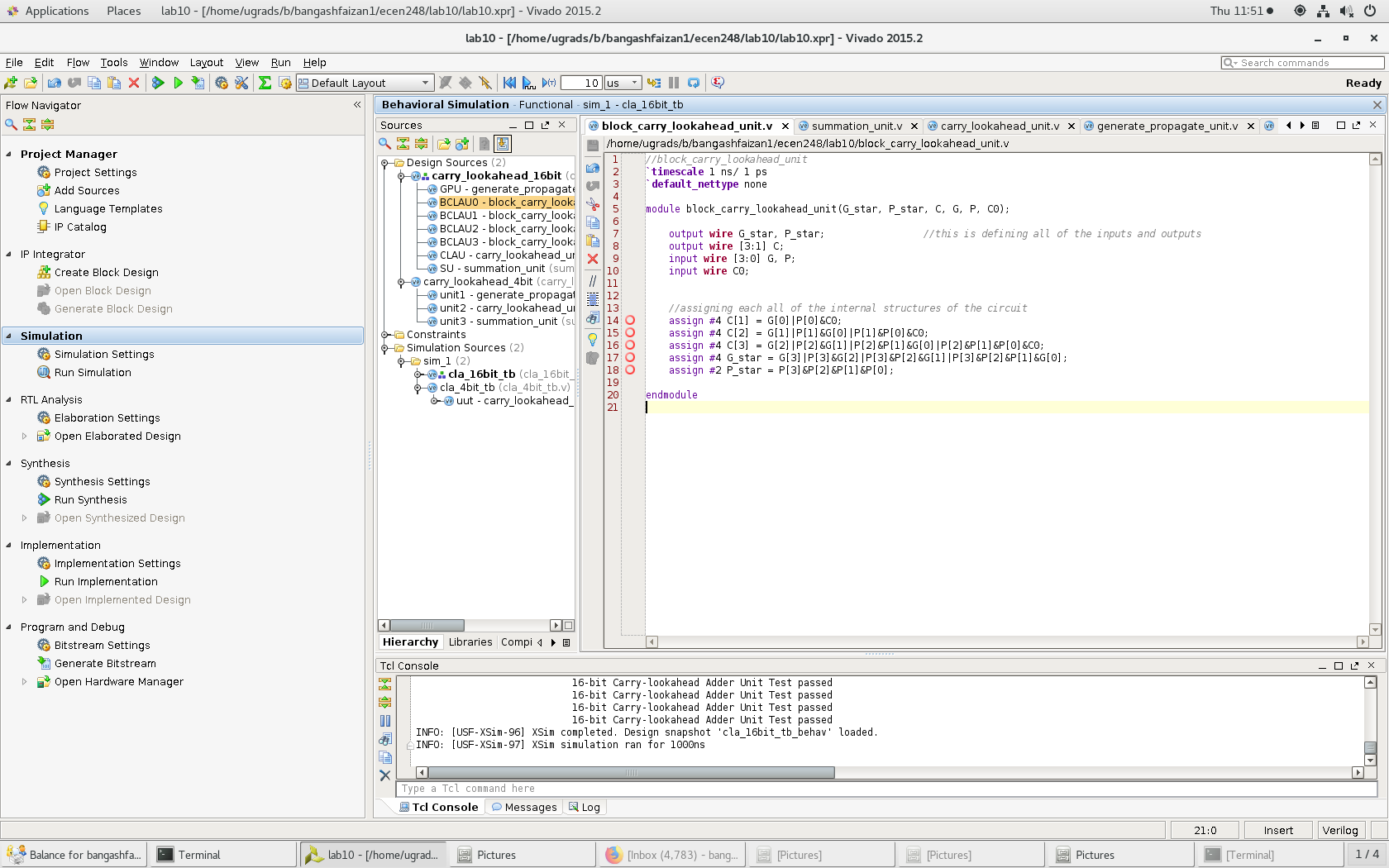
Carrylookahead unit

1. Here is the carry lookahead 4 bit adder that implements the sub-modules seen in the prelab as well..



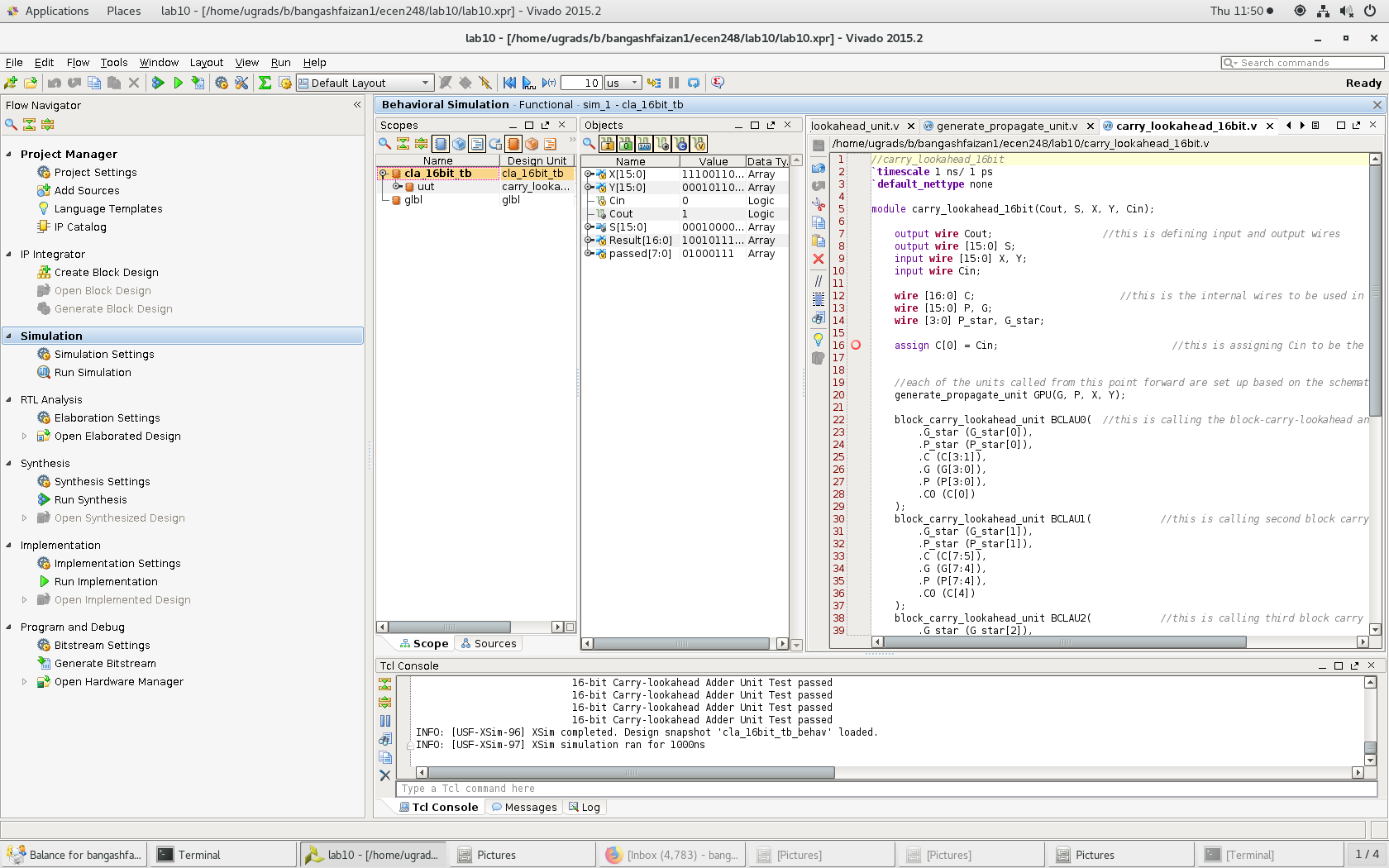
Carry\_lookahead 4 bit

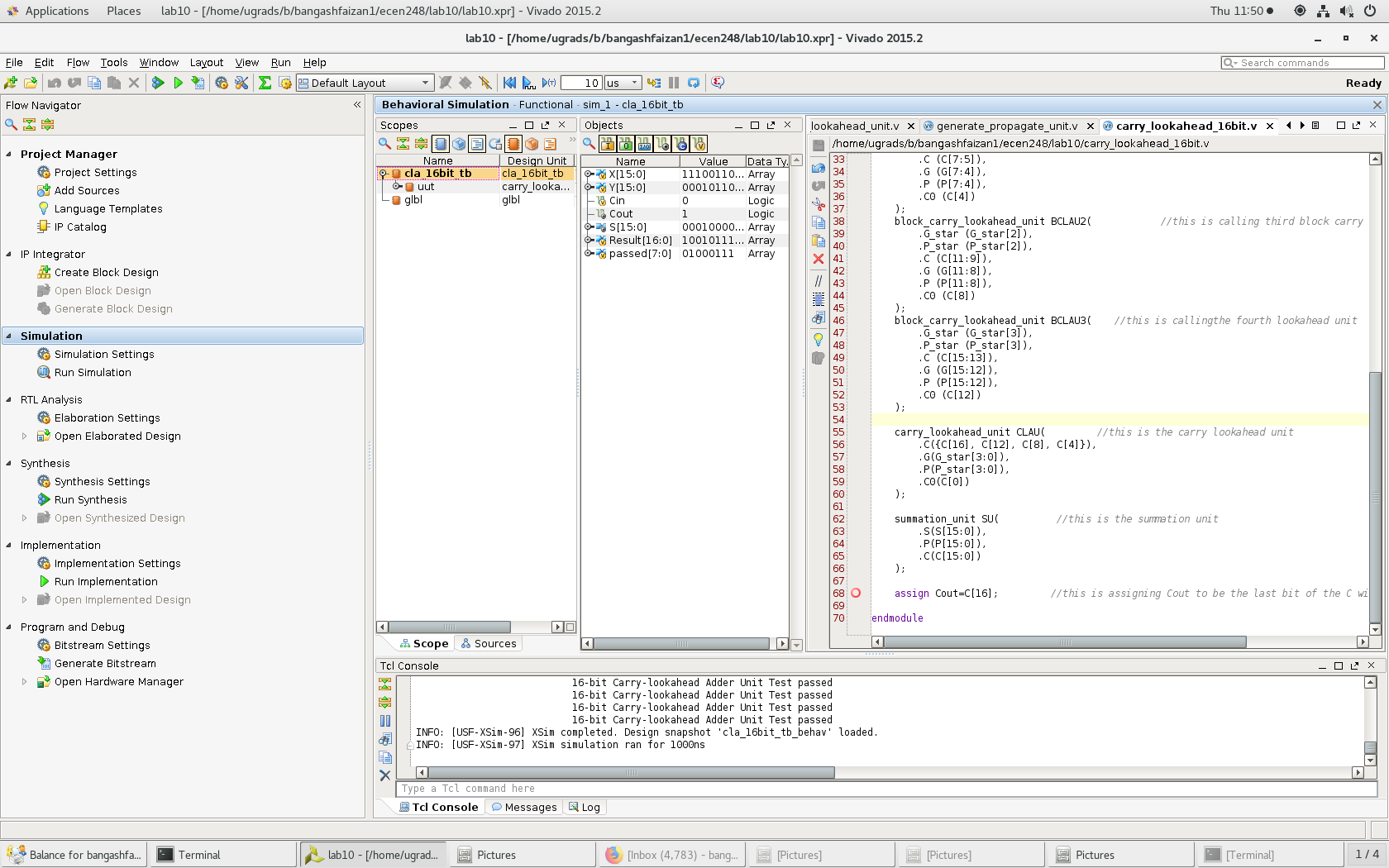
1. For Experiment 2 in this lab, we design a two-level carry lookahead unit to add 16-bit numbers together. Firstly, we made a block carry-lookahead adder which can be seen below



Block Carry Lookahead

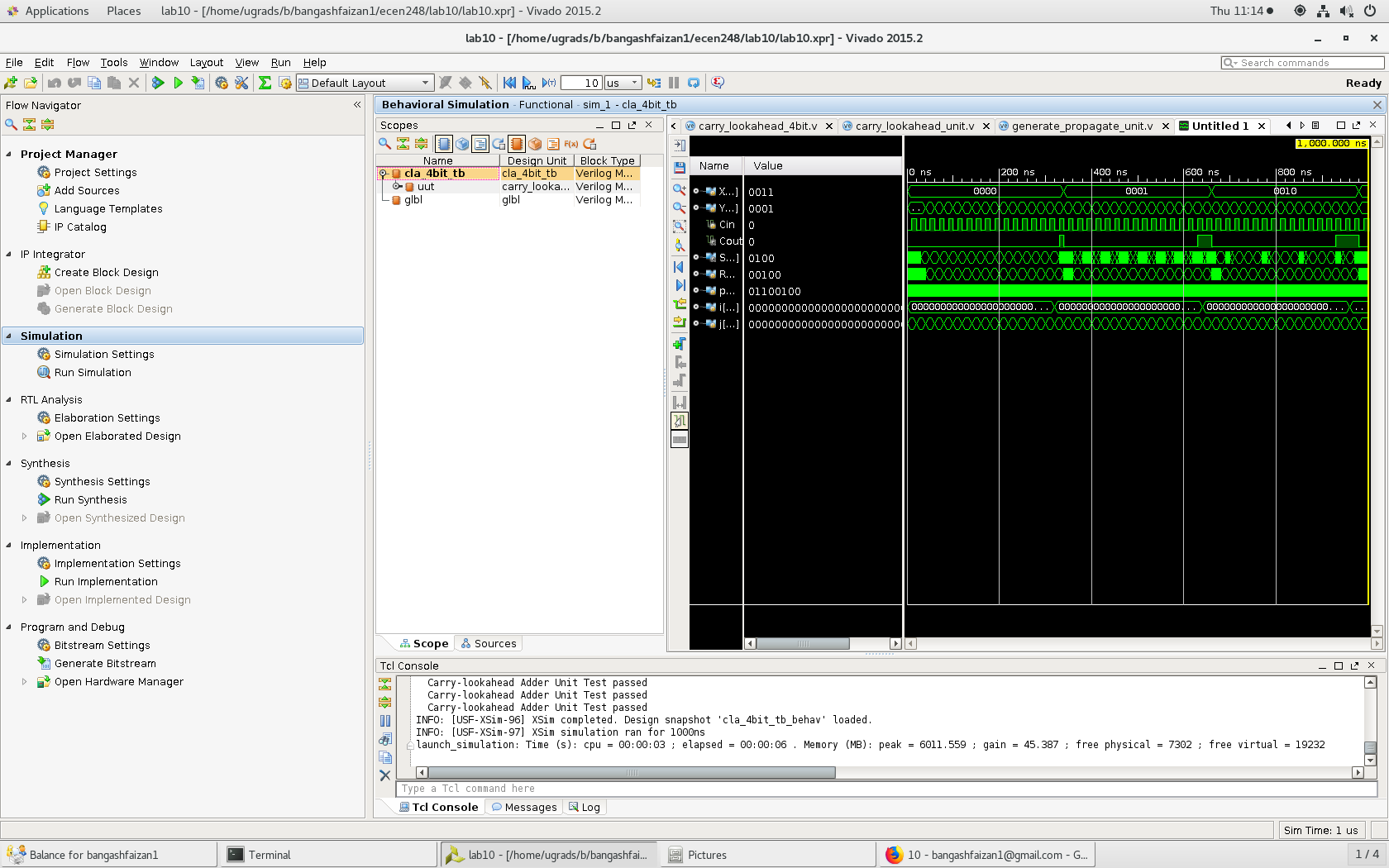
1. Following the Block Carry-Lookahead, we then created the Carry Lookahead Adder 16 bit and removed the delays and expanded the sub-modules from experiment one. The code for the Carry Lookahead Adder 16 bit can be seen below.



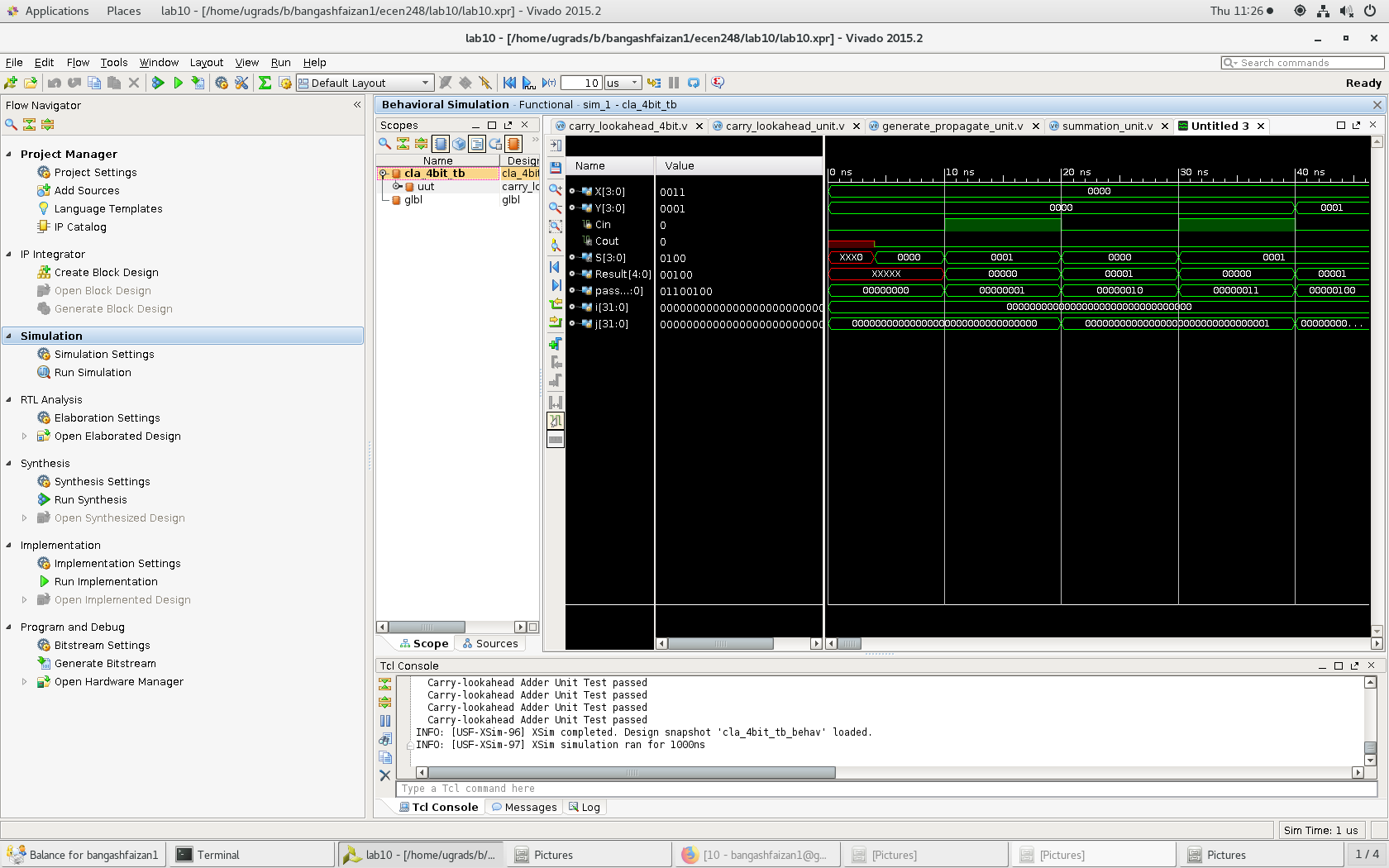


**Results –**

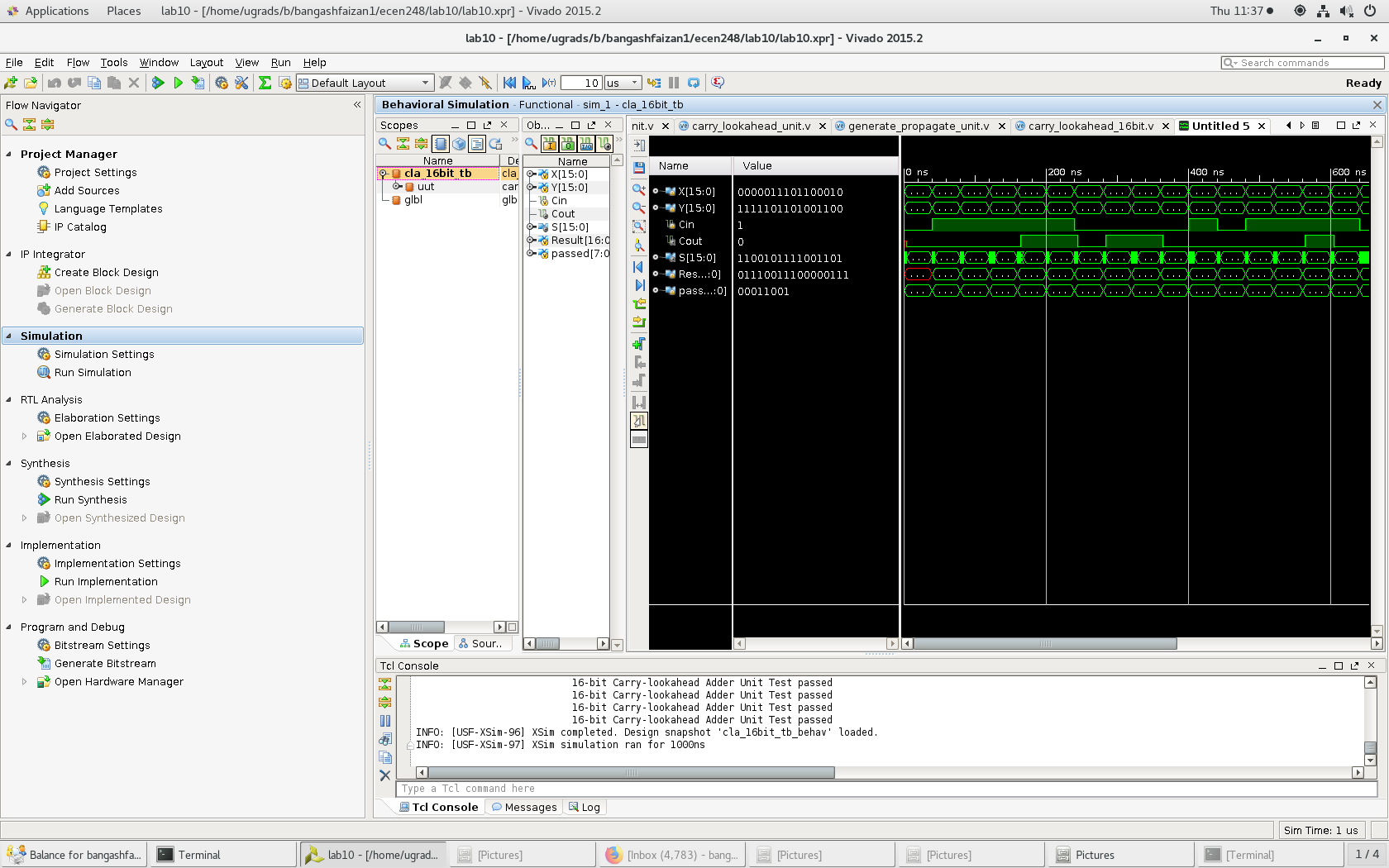
Overall, everything worked as expected. I didn’t run into any issues when working on the lab. All of the tests passed on the first or second attempt depending on if there were any typos or logic error in my code.



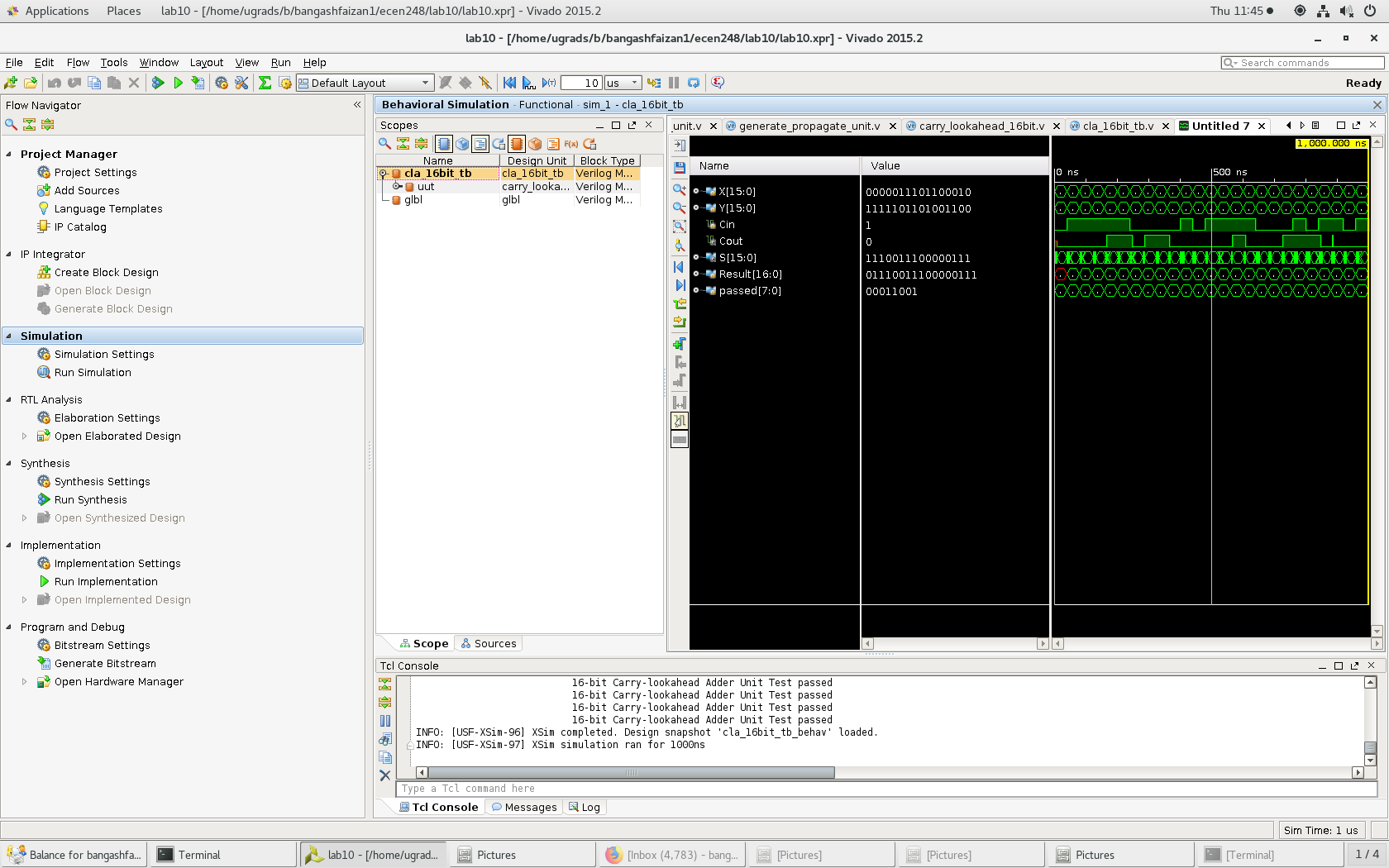
Cla\_4bit no delay



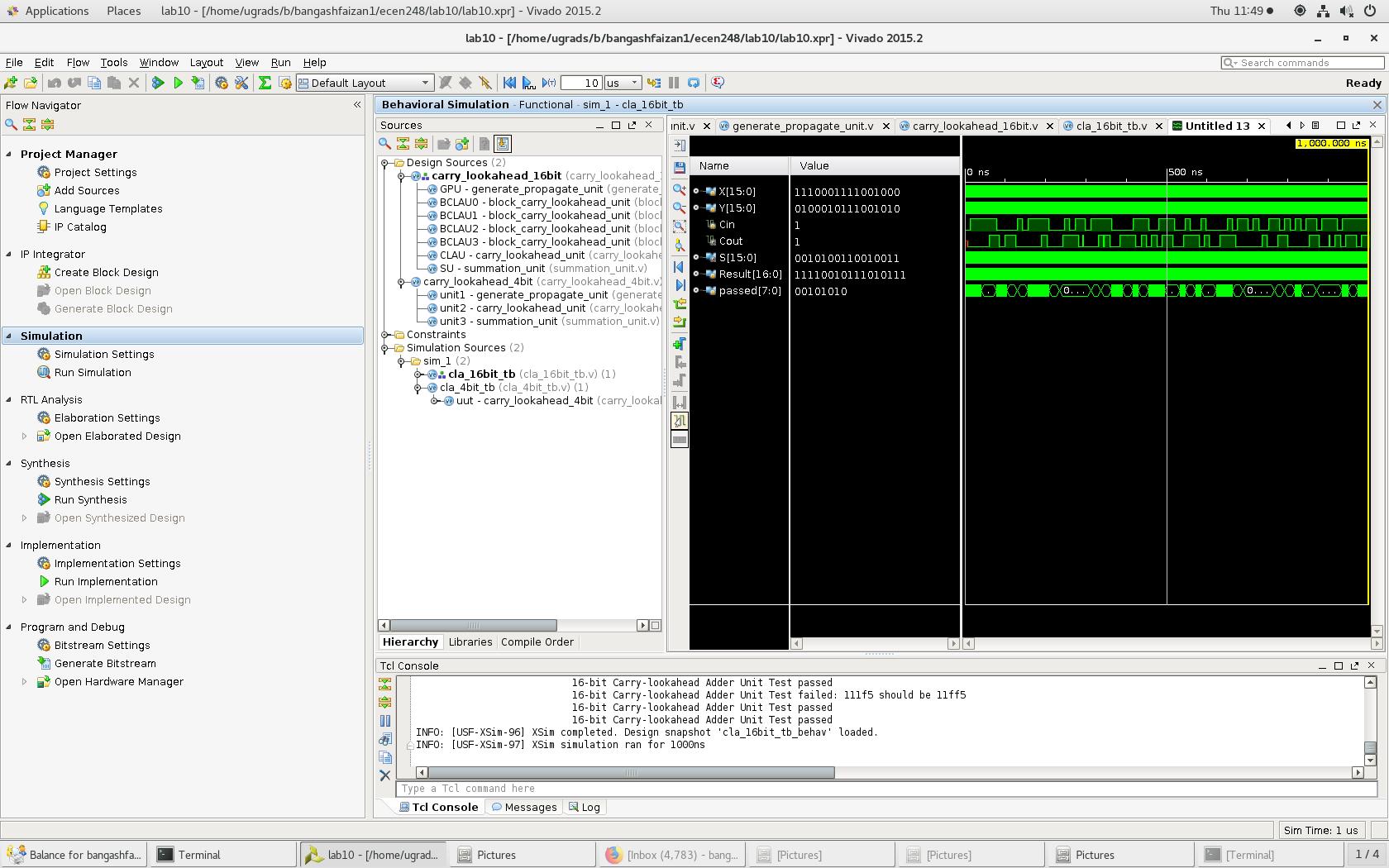
Cla 4bit with propagation delay



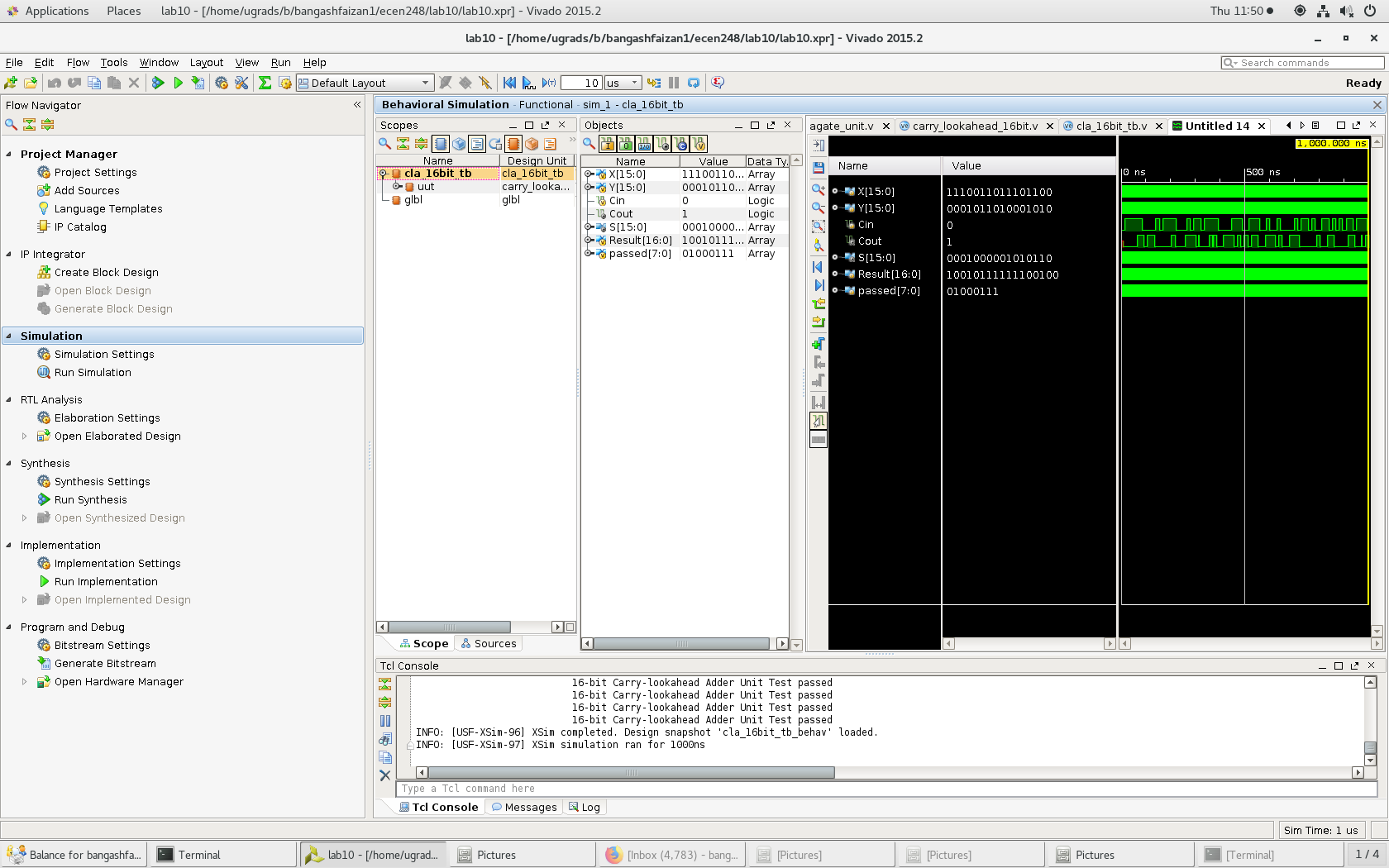
16 bit carry no delays



16 bit carry with propagation delay



16 bit carry with failed tests



16 bit carry with passed test

**Conclusion –**

This lab make it evident that delays are very crucial in circuits. If a delay is too high, a circuit will not function as expected. In this lab, it was clear that a ripple-carry adder was too slow so it had to be replaced with the faster adder. Speed is huge in the real world application of circuits so you will want to have a “perfect” circuit and this lab demonstrated that well.

**Questions:**

1. Include the source code with comments for all modules in lab. You do not have to include test bench code. Code without comments will not be accepted!
   1. Source code is included in the design section of the lab.
2. Include the simulation screenshots requested in the above experiments in addition to the corresponding test bench console output.
   1. No screenshots were required in the lab manual.
3. The gate count is 26 gates.
   1. The propagation delay of the 16-bit is 16 ns, 2-level carry-look ahead adder is 8 gates with agate count of 82.
   2. Yes. I calculated delta-g to be 2 ns and that the coefficient was 8. This time delay of 16 ns was exactly what the time delay ended up being for this circuit.
4. How does the gate-count of the 16-bit *carry-lookahead* adder compare to that of a ripple-carry adder of the same size?
   1. The 16-bit carry look-ahead adder has 82 gates in it and a 16-bit ripple carry adder has about 80 gates. However, for roughly the same amount of gates the 16-bit carry look-ahead adder as the ripple-carry adder.
5. How does the propagation delay of the 4-bit carry look ahead adder compare to that of a ripple-carry adder of the same size. Similarly, how does the 16-bit carry look ahead adder compare to that of a ripple carry adder of the same size.
   1. The 4-bit carry look-ahead adder the propagation delay is 3delta g. The 4-bit ripple carry adder has a delay of 9 delta g. For the 16-bit carry look-ahead adder there is a delay of 8 delta g and for a 16 bit ripple carry adder the delay is upwards of 33 delta g because it has 80 gates lined up in series. The carry look-ahead adder has a smaller delay in both because it works in a more efficient manner than the ripple carry adder. This is because of the usage of different block units woven together to complete a common task using parallel circuitry.