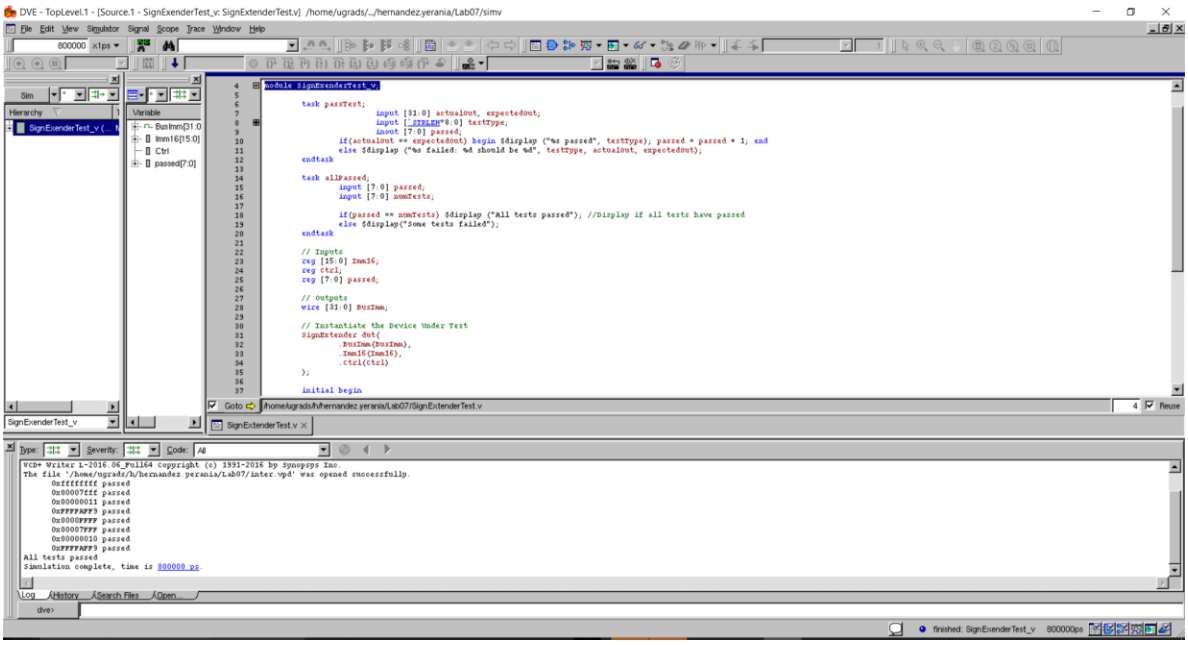
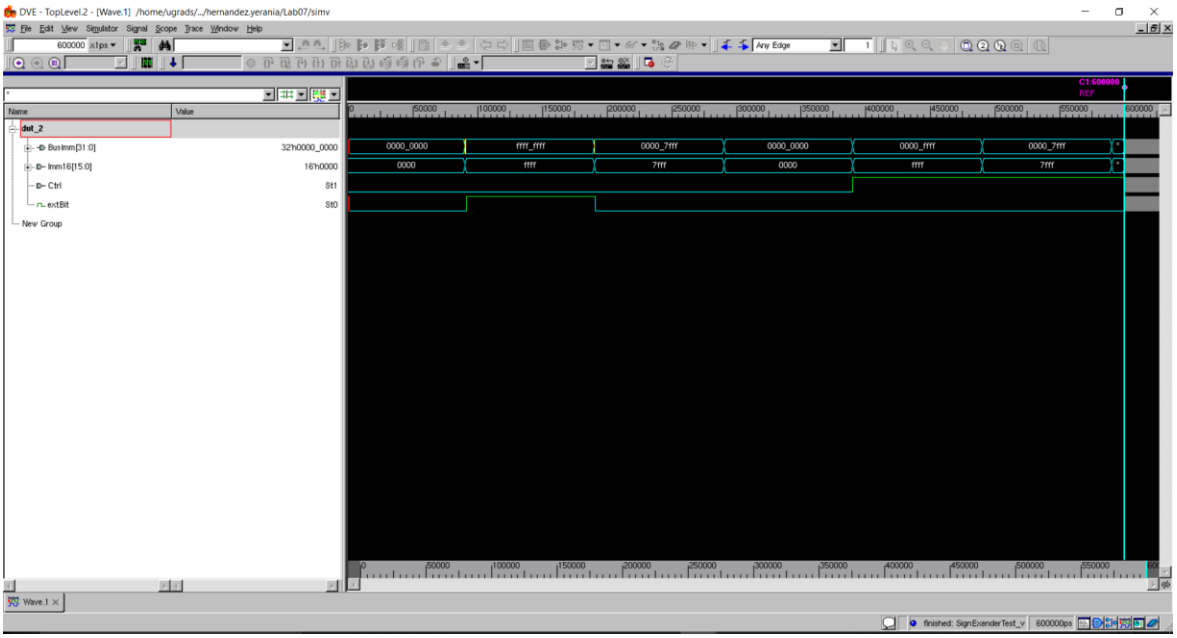
Lab 7

Sign Extender (VCS DVE Window "Heir.1")



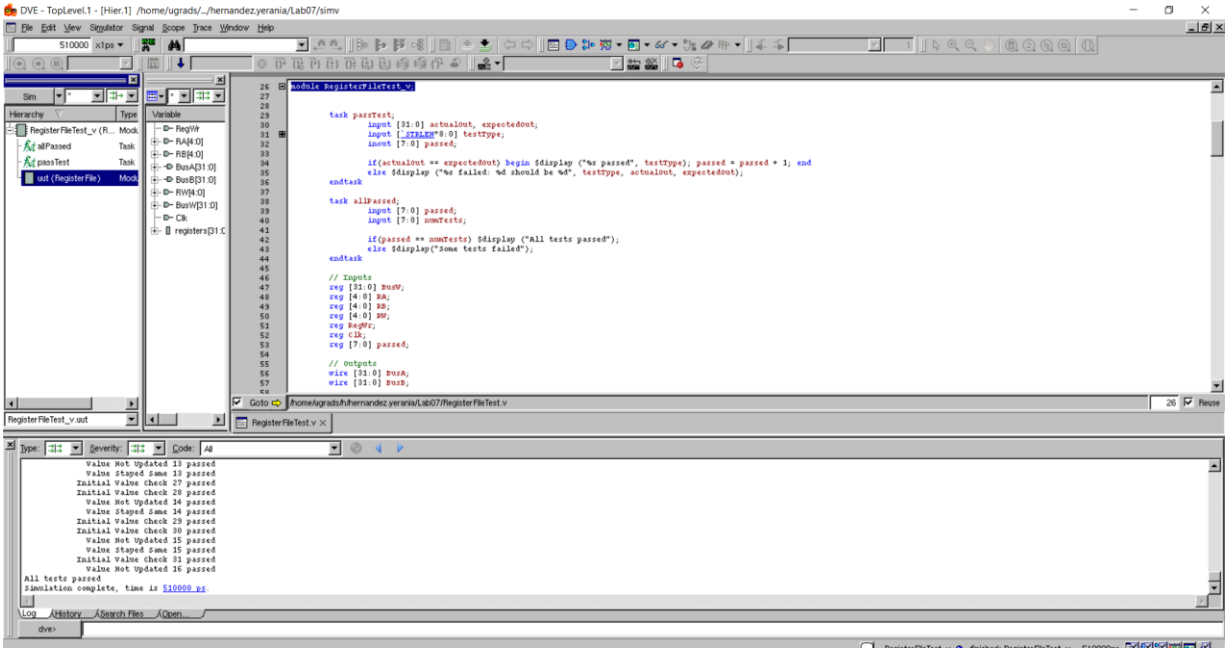
Sign Extender (Simulation Waveform "Wave.1")



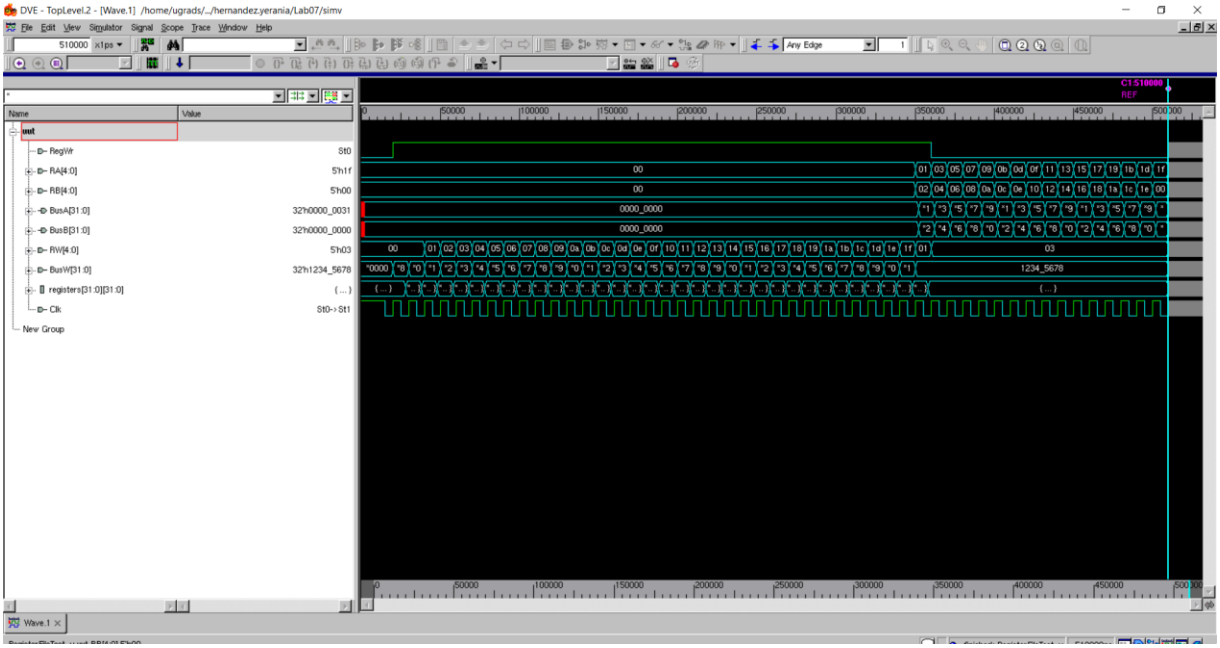
Sign Extender Explanation

According to the test bench, all the cases have passed. Analyzing the simulation, it becomes obvious that when the Ctrl is 0, then the 16-bit values are sign extended, that either having a 0 or 1 bit. Extreme cases were tested as well as two other numbers that were randomly picked to test sign extension on them, such as negative values. When Ctrl is 1, some of the same cases were tested but to see if the zero is extended, which is the case based on the simulation.

Register File (VCS DVE Window "Heir.1")



Register File (Simulation Waveform "Wave.1")

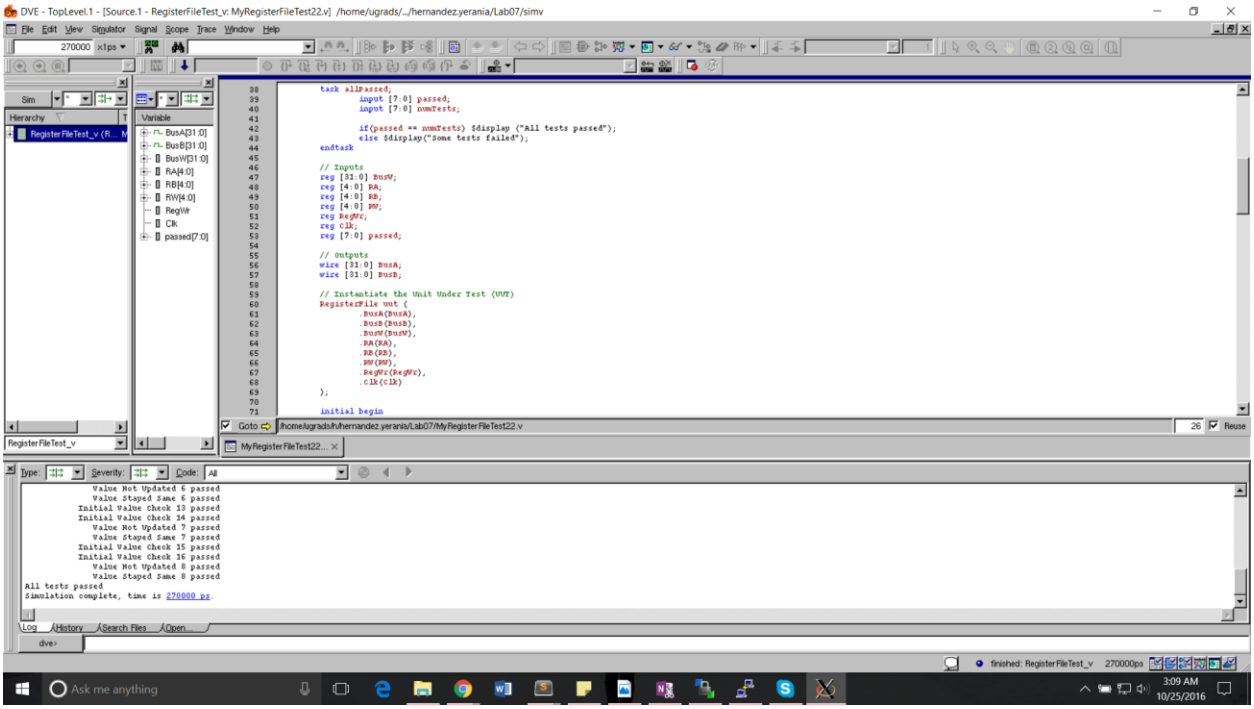


Register File Explanation:

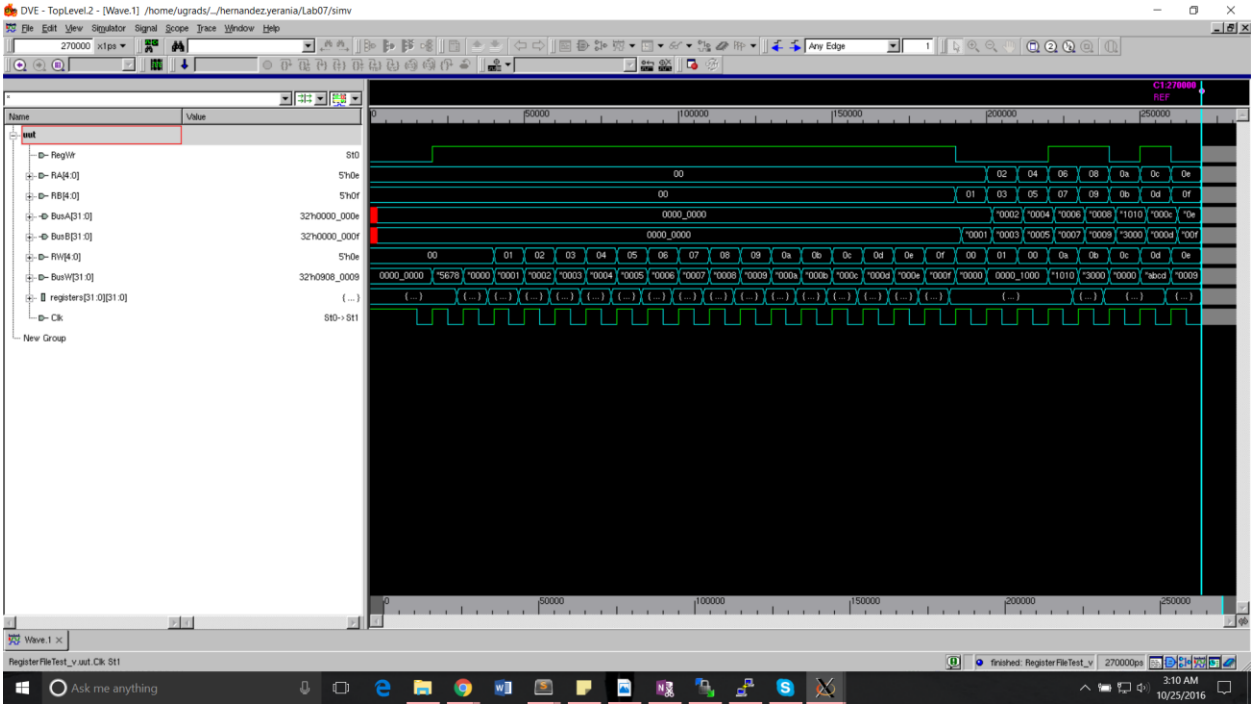
According to the test bench, all test cases have passed. Based on the simulation, we can tell that the register is actually functioning properly as it reads in the values from RA, RB, and RW and outputs there value in their proper destination (BusA and BusB). It also does not write unless there is a negative clock edge, which is obvious based on the conditions necessary for the code.

As soon as the value of RegWr is 0, however, the last value contained at the negative edge of RW's clock is contained and is what is written in Bus W.

My Register File (VCS DVE Window "Heir.1")



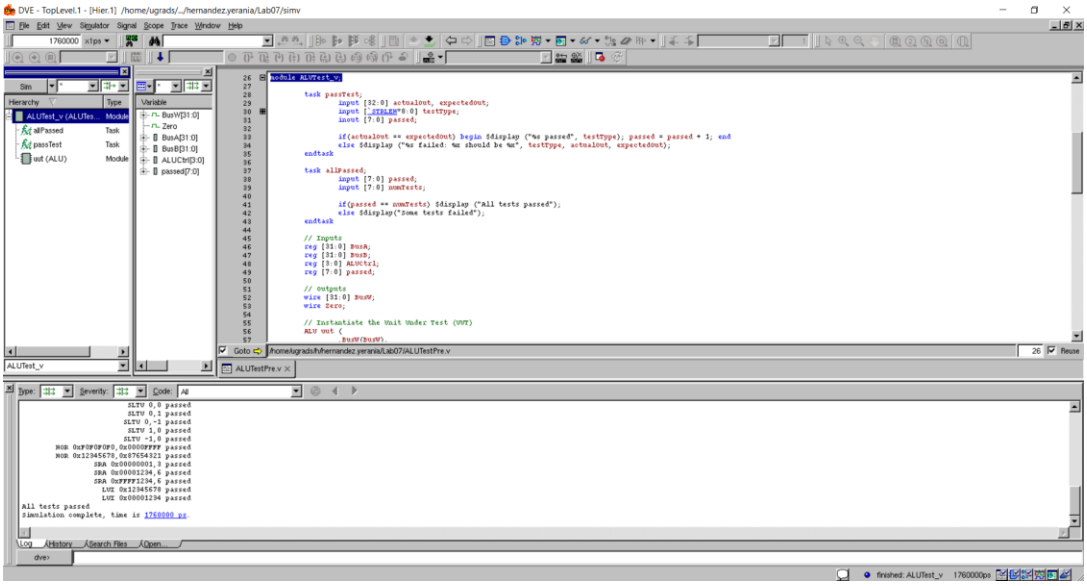
My Register File (Simulation Waveform "Wave.1")



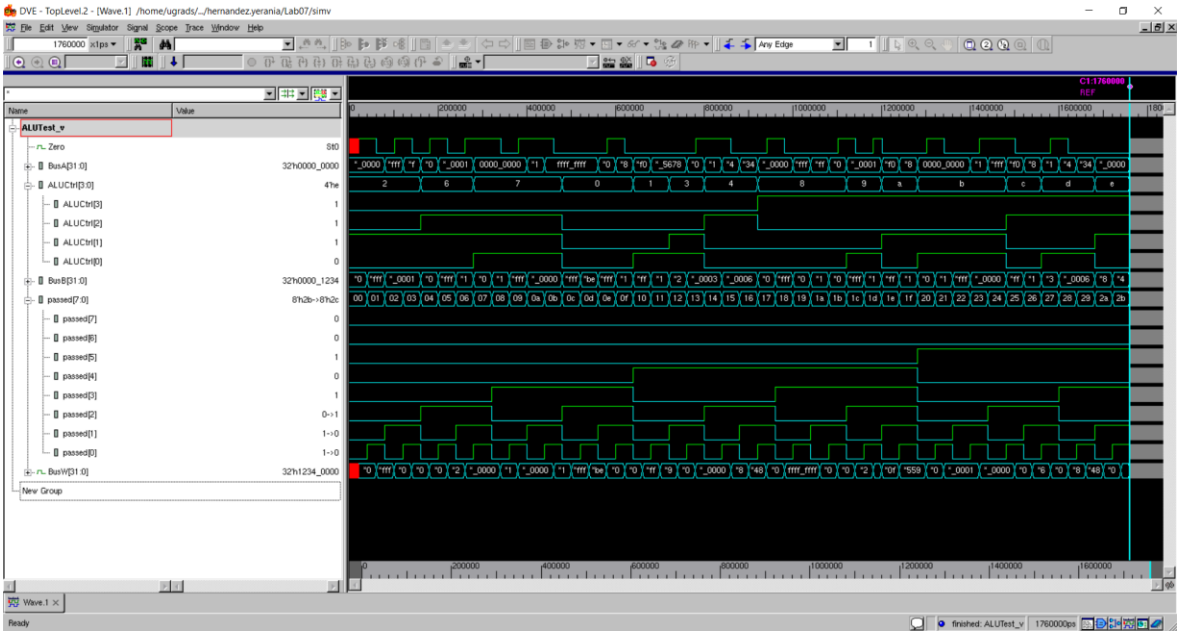
My Register Explanation:

My Register is the same code as the Register File with different test case scenarios simulated based on the lab's specifications. These test cases are testing a specific scenario when register values can be replaced with addresses and vice versa depending on the way they are written within the code. This proves the table that was provided in the lab in that future instructions can be affected by previous instructions if not careful, as well as it can be easy to replace the content within the current instruction. Based on the simulation, the test cases are similar to Register File and they function properly, such as when A and B both have addresses already written by Bus W and therefore, the new instruction uses these addresses instead of the previous addresses.

ALU (VCS DVE Window "Heir.1")



ALU (Simulation Waveform "Wave.1")



ALU Explanation:

According to the test bench, all tests have passed. Based on the simulation, you can tell there are various cases that were tested for each of the operations and a variety of the same operations as well. This test file contains test cases from the prelab and cases that were previously given. It gives a good idea on how each of the operations function when simulated. For example, it uses the ALUCtrl to know what operation it will operate, then it performs the operation, and returns the result in BusW, such as addition which is executed based on the default of it being signed.