February 2, 2021
Alejandro Herrera Gonzalez

Minilab 2 Report

In the testbench for minilab two, the method utilized is somewhat trivial.

At the start of the test bench, we will generate 8 simple values using for loops for for the logic signals that will drive the values of Ain, Bin, and Cin, placing all the values in an array.

At the start, the bench tests that the active low reset functions properly.

Since there is no accounting for overflow in the operation of Ain and Bin, the test bench first tests that the tpumac module properly sums up the value Ain*Bin + Cout when the WrEn is not asserted and when en is also asserted with the 8 values generated. This bench also tests that when WrEn is asserted the value is Cin is passed onto Cout and that the module holds the same value when the en is low. This is done by randomly raising the value of WrEn and cross checking that in the next clock cycle the correct value is at Cout, Aout, and Bout.

The last thing that the bench tests is that the module can perform passing the value of Cin to Cout and the operation Ain*Bin + Cout when all the bits being used. To do this we will have large negative and positive values for the Ain, Bin, and Cin. First, we will load a large value to Cout. Then we will perform the Ain*Bin + Cout operation. In both instances we will cross check Aout, Bout, and Cout with the expected values.

At every check we will increase a counter and print expected and obtained output values if a discrepancy. To pass the bench we must have zero discrepancies between expected values and actual outcome values.