*Assignment-5*

*16-bit MAC*

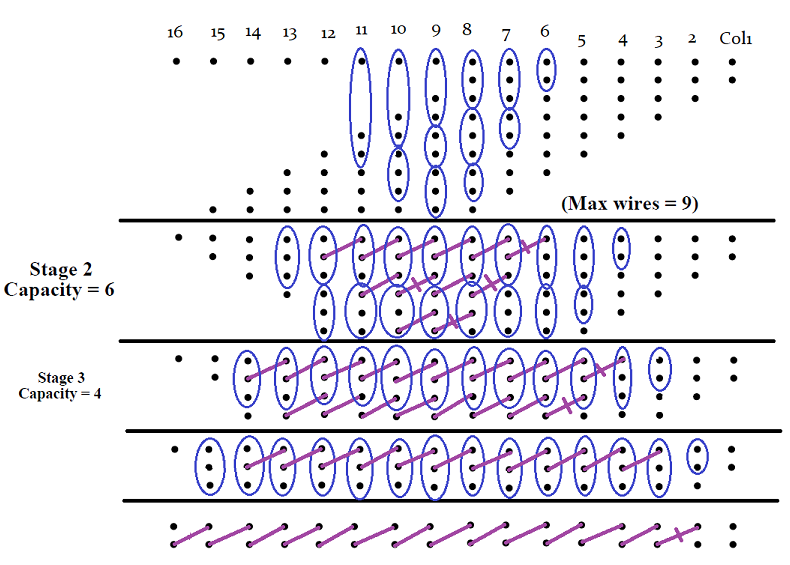
Vagadia Yash Dinesh

Roll Number:20D070086

Course:EE671 VLSI Design

**Q-1**

Designing a 16-bit MAC with two 8-bit numbers as a input and a 16-bit number C as a input, in VHDL and simulating it using a test bench.



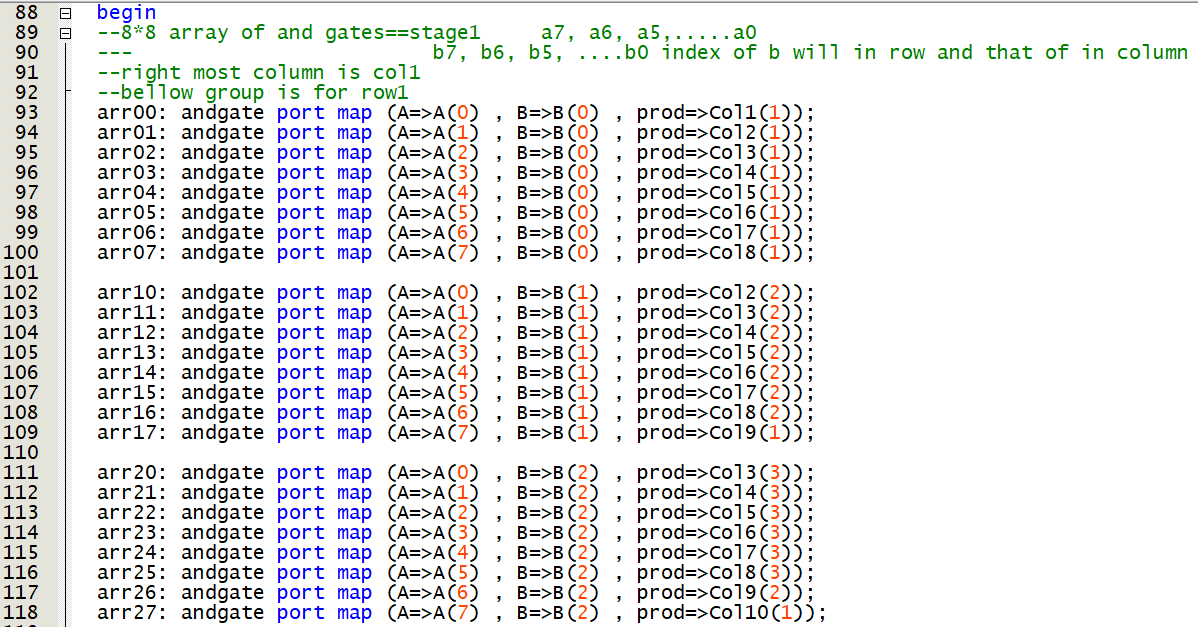
Dot diagram of MAC, wire reduction using DADDA scheme. From the 1st stage wires are named Colj(i), where j column number and i the index for element in that column.

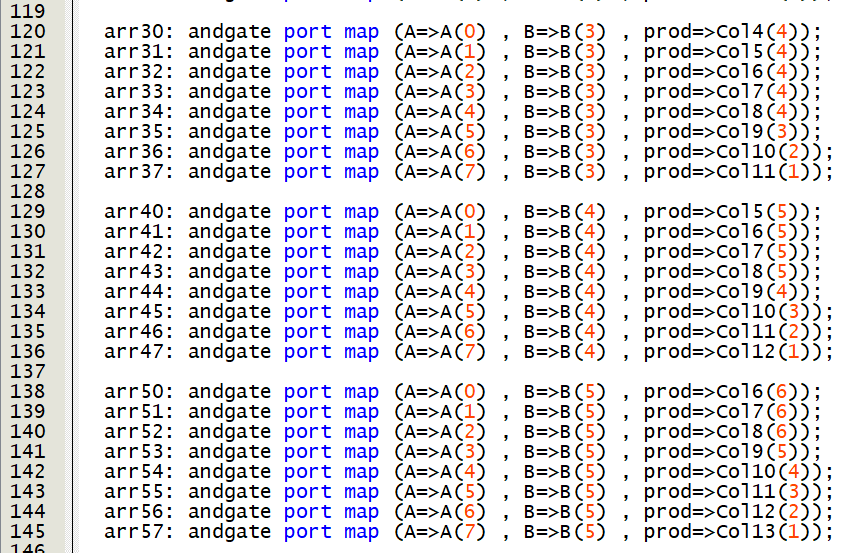
From stage-2 output are named Wj(i) and same way from stage-3,4,5 as X,Z,Y. These are wires are defined as signals at the start of architecture.

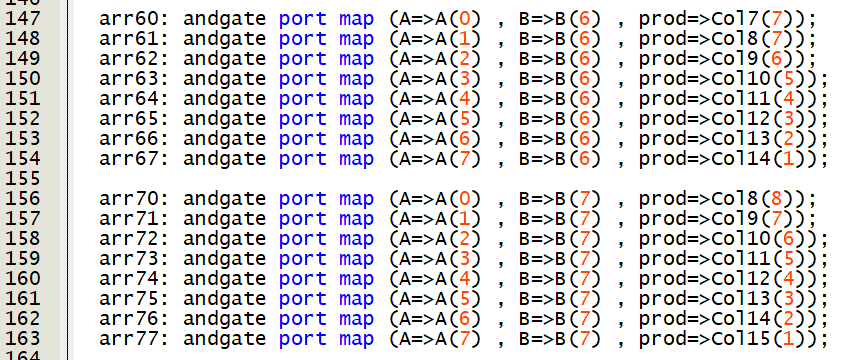
Full adders, Half adder and logarithmic adder entities are defined in *Gates.vhdl* file using the given basic gates.

**a.**

An 8 cross 8 array of and gates is defined to generate partial product bits.

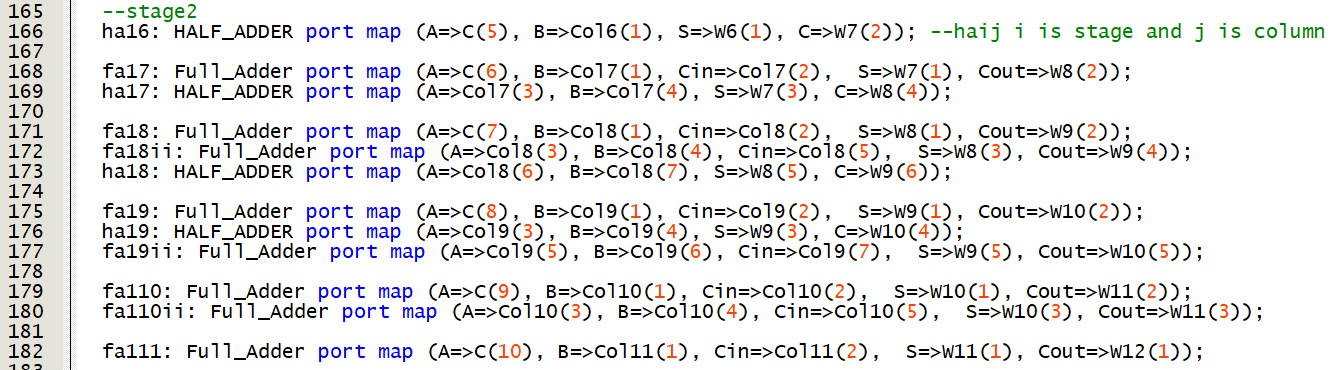


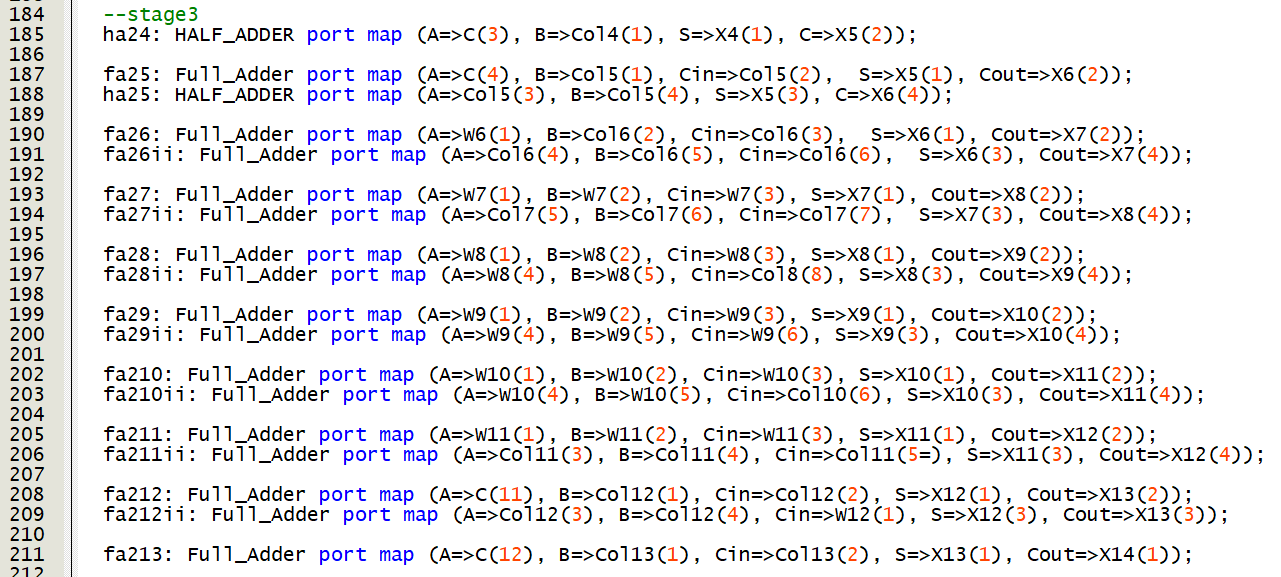




**b.**

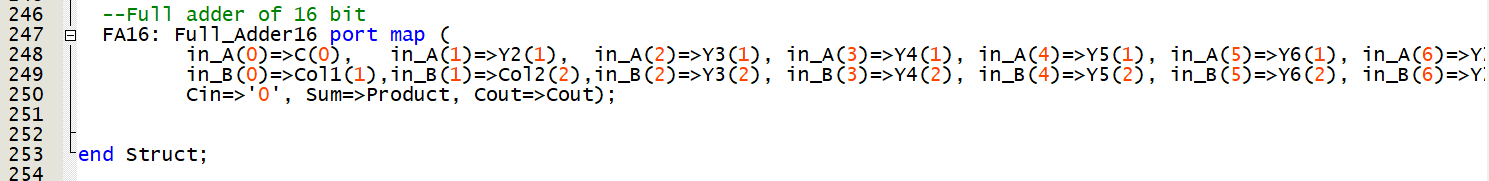
Then Half adders and full adders are used for wire reduction as shown in dot diagram.







**c.** At finally 16-bit logarithmic adder is used from assignment-4 to final answer. From DUT final product and Carry is output.



There are 4 files in the project made in Quartus-MAC, DUT, Testbench and Gates

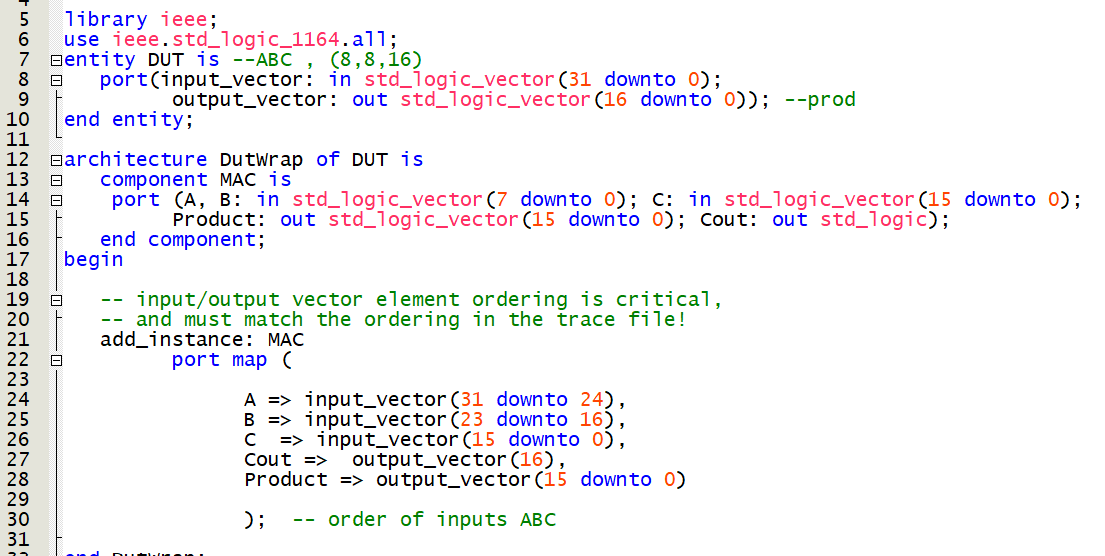
In MAC, structural code is written, DUT is for test the device.

Tracefile.txt (in which input combinations are written) is generated using python *tracefile.py*

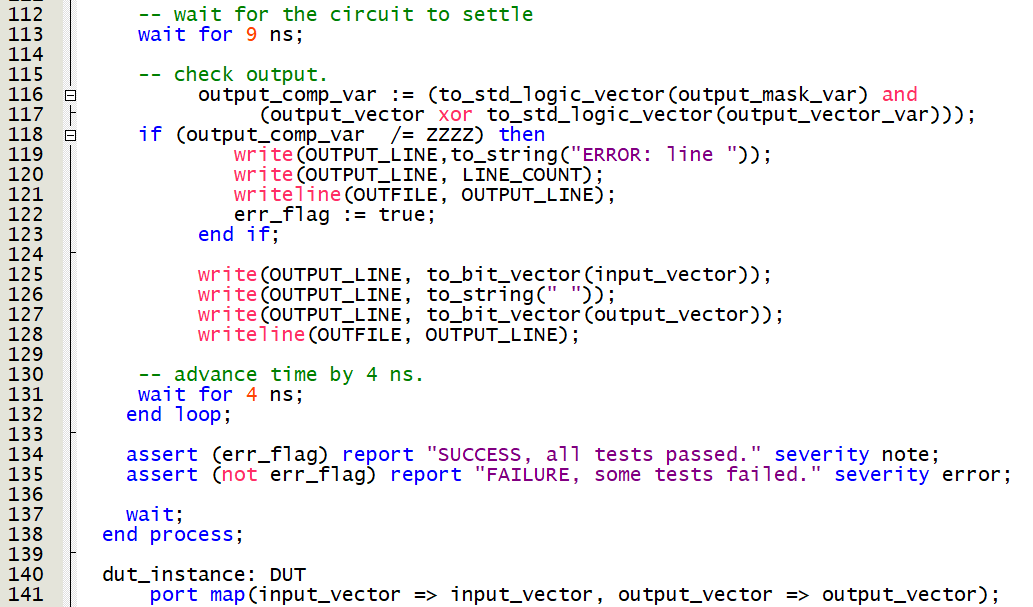
I am using 20 random combinations of inputs for verification and a max value.

For an input—10101100000001111010001110011110 and output-01010100001010010

In simulation it starts at **247 ns** and settles at **249.8 ns**, so **2.8 ns** delay to generate final output.



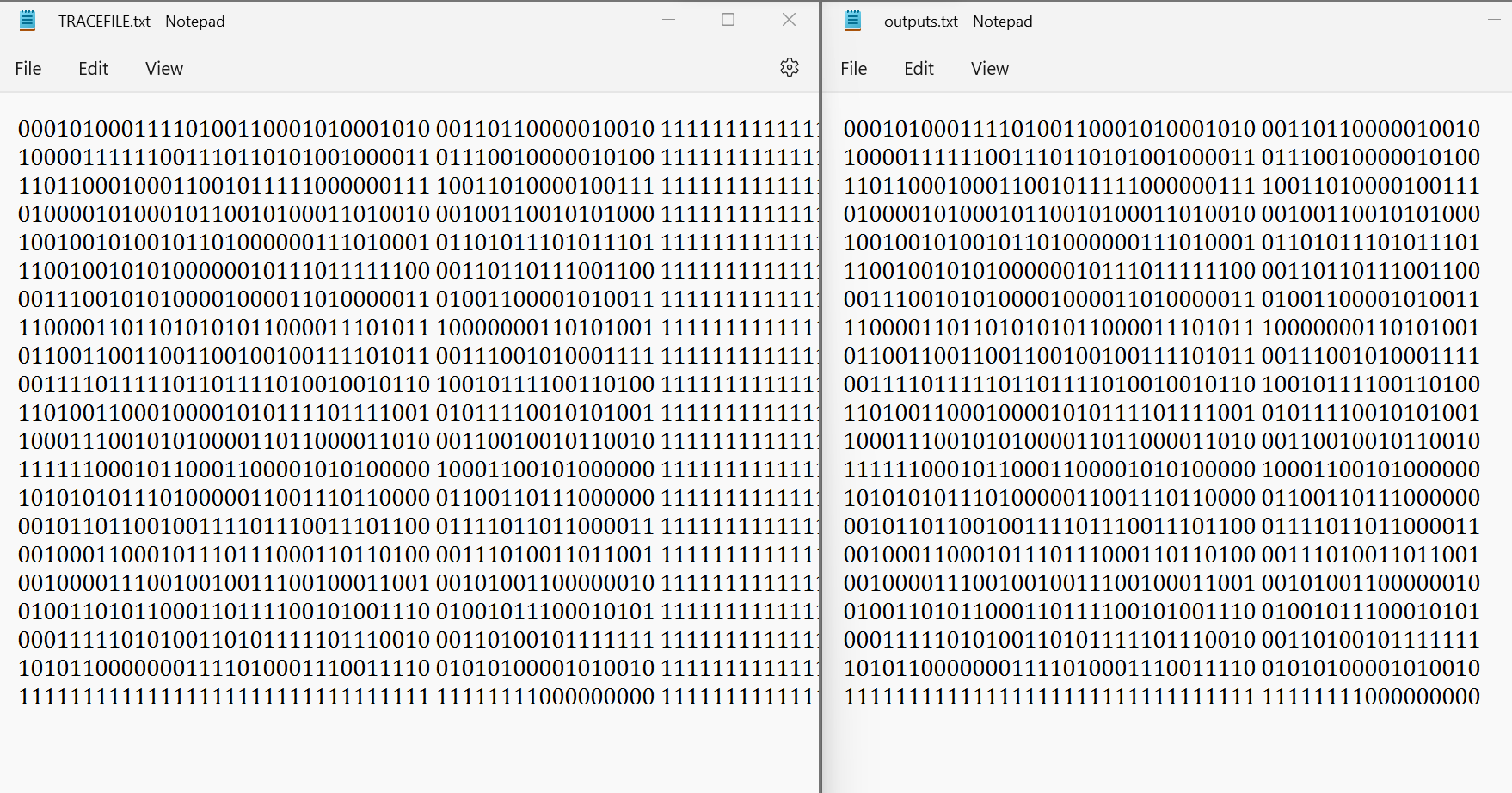
DUT code

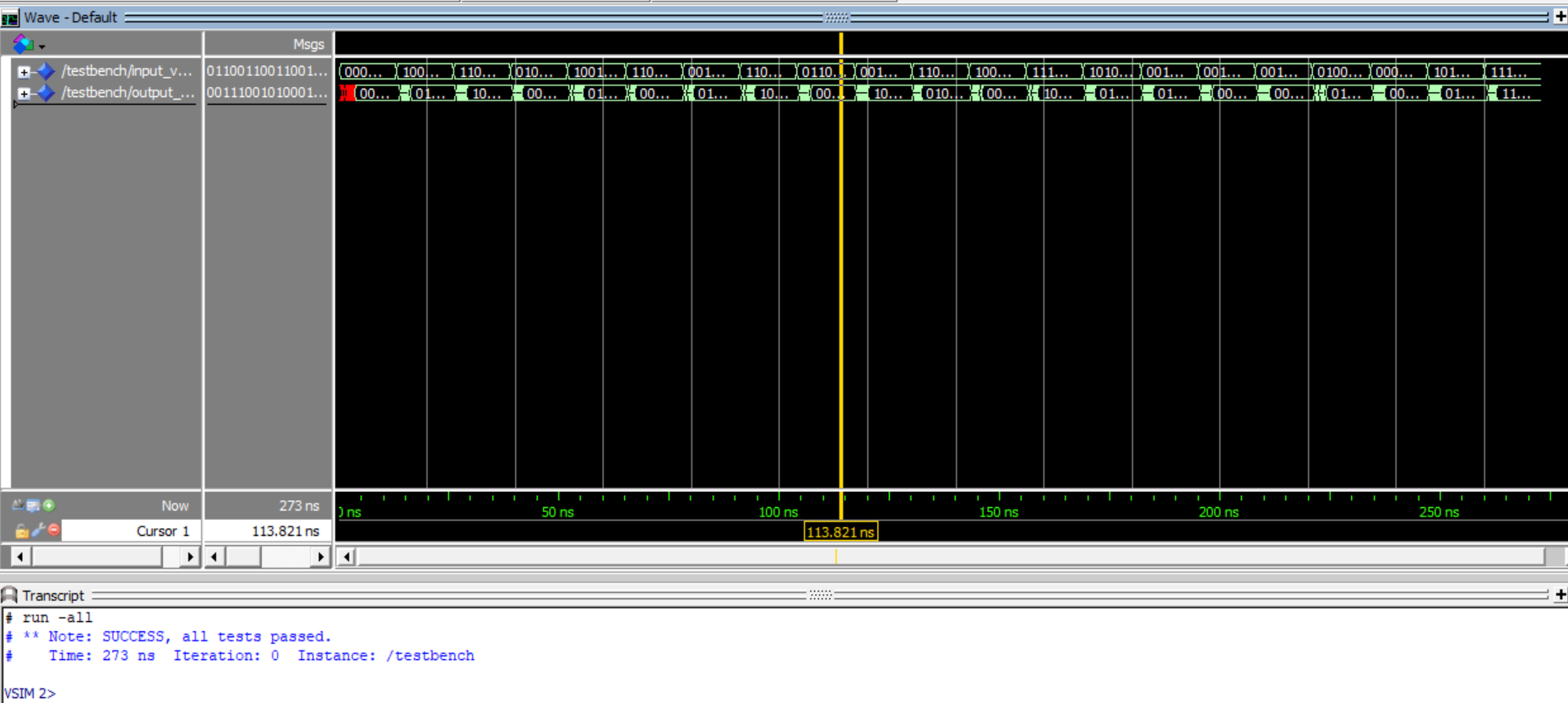


Testbench code

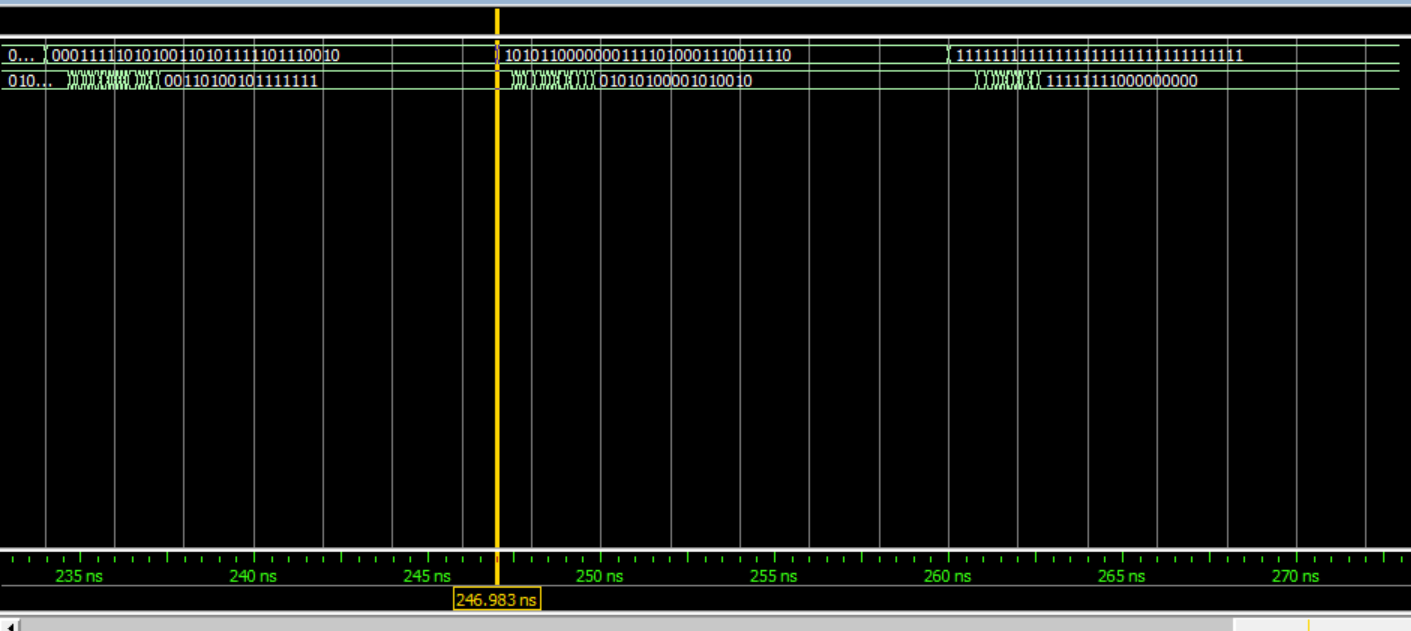
Output file is generated at simulation/modelsim/output.txt

***All the cases passed for the given random combination of input with no error.***



******

**RTL simulation**

****