CSE 140L Lab 1

Amy Nguyen A16125627; (Mark Lorenzo, A15955698); (Jared Villanueva, A15821317);

Academic Integrity

Your work will not be graded unless the signatures of all members of the group are present beneath the honor code.

To uphold academic integrity, students shall:

- Complete and submit academic work that is their own and that is an honest and fair representation of their knowledge and abilities at the time of submission.
- Know and follow the standards of CSE 140L and UCSD.

Please sign (type) your name(s) below the following statement:

I pledge to be fair to my classmates and instructors by completing all of my academic work with integrity. This means that I will respect the standards set by the instructor and institution, be responsible for the consequences of my choices, honestly represent my knowledge and abilities, and be a community member that others can trust to do the right thing even when no one is watching. I will always put learning before grades, and integrity before performance. I pledge to excel with integrity.

(Amy Nguyen)
(Mark Lorenzo)
(Jared Villanueva)

Free Response

Please answer the following questions.

1. What keyword do you use in SystemVerilog to indicate combinational logic for a single line of code? (2 pts)

always_comb() or **assign** can be used for a single line of code.

2. What keyword do you use in SystemVerilog to indicate combinational logic for multiple lines of code? (2 pts)

always_comb() is used to indicate combinational logic for multiple lines of code

3. What keyword do you use in SystemVerilog to indicate sequential logic? (2 pts)

always_ff()

4. What symbol indicates blocking assignment in SystemVerilog? (2 pts)

The **=** symbol indicates blocking assignments.

5. What symbol indicates nonblocking assignment in SystemVerilog? (2 pts)

The **<=** symbol indicates non-blocking assignments.

6. Why do we need nonblocking assignments in SystemVerilog? (2.5 pts)

Non-blocking assignments are for assignments without blocking the procedural flow. This is for when you want to make register assignments simultaneously, and when you want them to not affect each other.

Citation: http://www.asic-world.com/tidbits/blocking.html

7. Why do we need blocking assignments in SystemVerilog? (2.5 pts)

Blocking assignments are for assignments when you want to block the procedural flow. This is for when you want to make register assignments sequentially, and when you want them to affect each other.

Citation: http://www.asic-world.com/tidbits/blocking.html

Screenshots

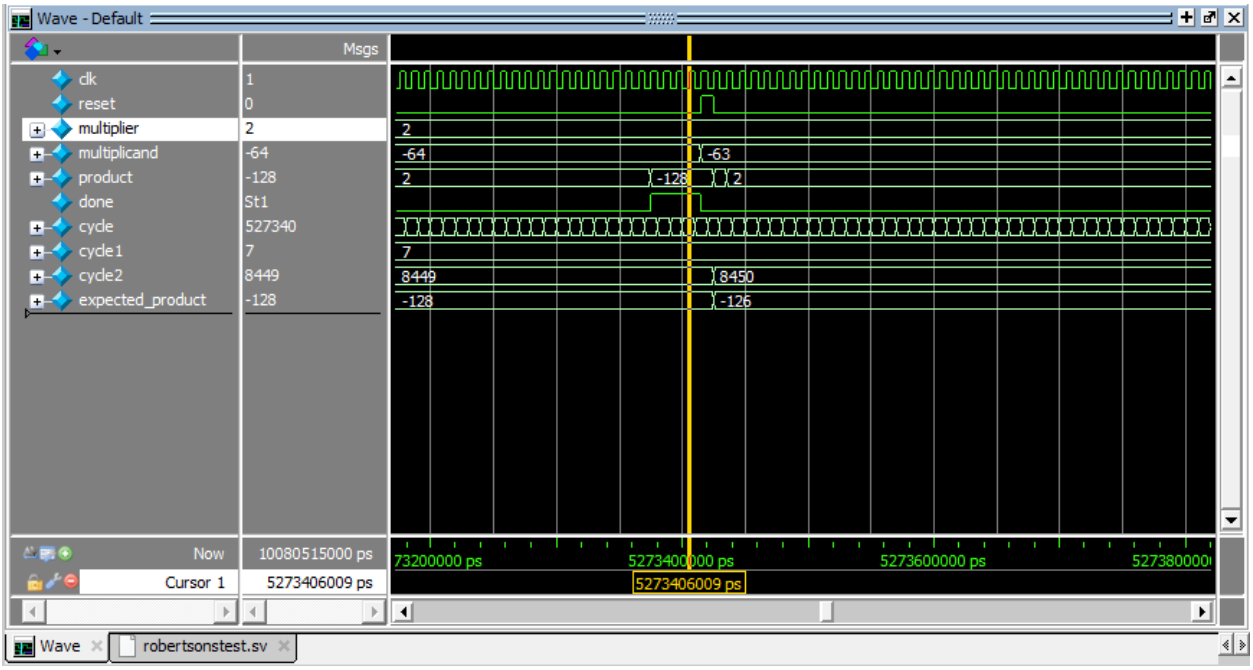
Screenshot of your ModelSim transcript, showing the expected results vs. yours. (3 pts)

Make sure the comprehensive test output is included.

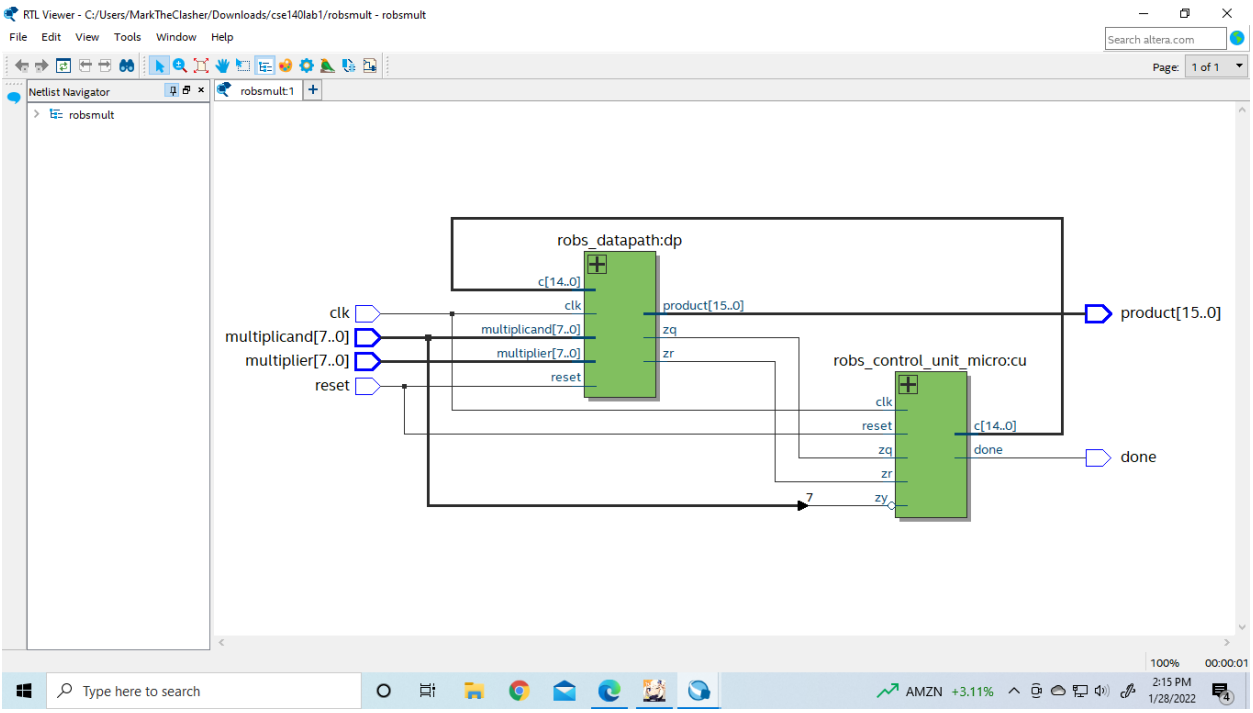
```
VSIM 3> run -all
# Simulation succeeded 0000 = 0000 = 00 * 00
# Simulation succeeded      0 =      0 =    0 *    0
# Simulation succeeded 001e = 001e = 05 * 06
# Simulation succeeded     30 =     30 =    5 *    6
# Simulation succeeded ffdd = ffdd = 07 * fb
# Simulation succeeded    -35 =    -35 =    7 *   -5
# Simulation succeeded ffe2 = ffe2 = fb * 06
# Simulation succeeded    -30 =    -30 =   -5 *    6
# Simulation succeeded ffc8 = ffc8 = f9 * 08
# Simulation succeeded    -56 =    -56 =   -7 *    8
# Simulation succeeded 001e = 001e = fb * fa
# Simulation succeeded     30 =     30 =   -5 *   -6
# Simulation succeeded 0024 = 0024 = f7 * fc
# Simulation succeeded     36 =     36 =   -9 *   -4
# clock cycles =     1008051, test cycles =           7, sequence_cycles =       16384
# ** Note: $stop    : C:/Users/MarkTheClasher/Downloads/csel40labl/robertsonstest.sv(92)
#    Time: 10080515 ns  Iteration: 0  Instance: /robertsonstest
# Break in Module robertsonstest at C:/Users/MarkTheClasher/Downloads/csel40labl/robertsonstest.sv line 92
```

Screenshot a portion of your ModelSim waveform viewer, showing a done flag / reset cycle, the incoming operand values, and the product value. (3 pts)

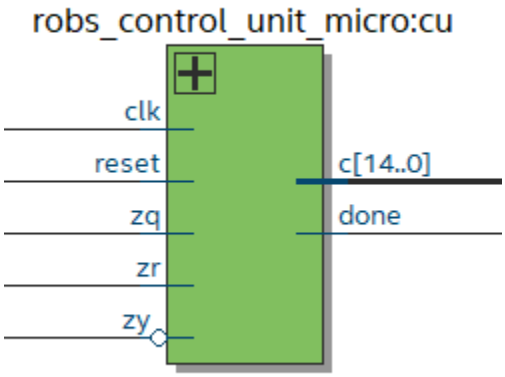
We want to see at least all the variables from `robertsontest`, this can be done by right clicking on `robertsontest` and then “Add wave.” Please zoom in to the level so that we can see when done is 1, you have the correct product for at least one pair of multiplier and multiplicand. It is fine if you show multiple pairs.



Screenshot of your Quartus RTL Viewer top-level (control block and data path block) (3 pts)



Screenshot of your Quartus RTL Viewer top level of the control block (3 pts)



Screenshot of your Quartus RTL Viewer top level of the data path block (3 pts)