**NCTU-CS Computer Organization**

**Lab0**

**Design Description**

This lab is for practice only. It is designed to help you be familiar with the environment. You don’t need to hand in your code.

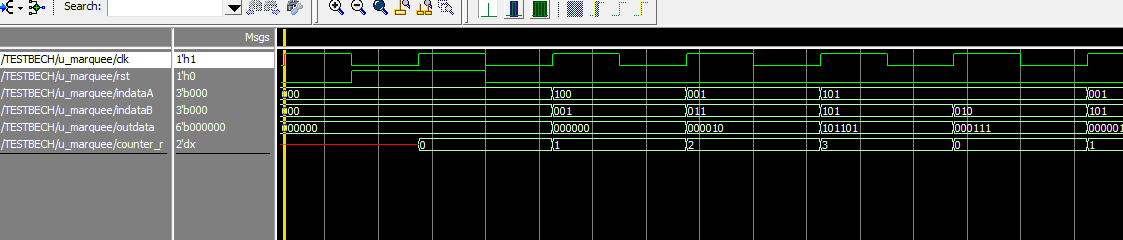
In this lab, you are going to design a simple marquee(跑馬燈). Your design has to process 2 input data and output 4 different patterns.

**Design Specification**

1. Top module: marquee
2. Input port: clk, rst, indataA[2:0], indataB[2:0]
3. Output port: outdata[5:0]
4. Clock cycle is 5ns.
5. There is an active high reset signal.
6. Define a counter in the design. Use it to change your output pattern.
7. Your design should be synthesizable.
8. The corresponding pattern is as followed:

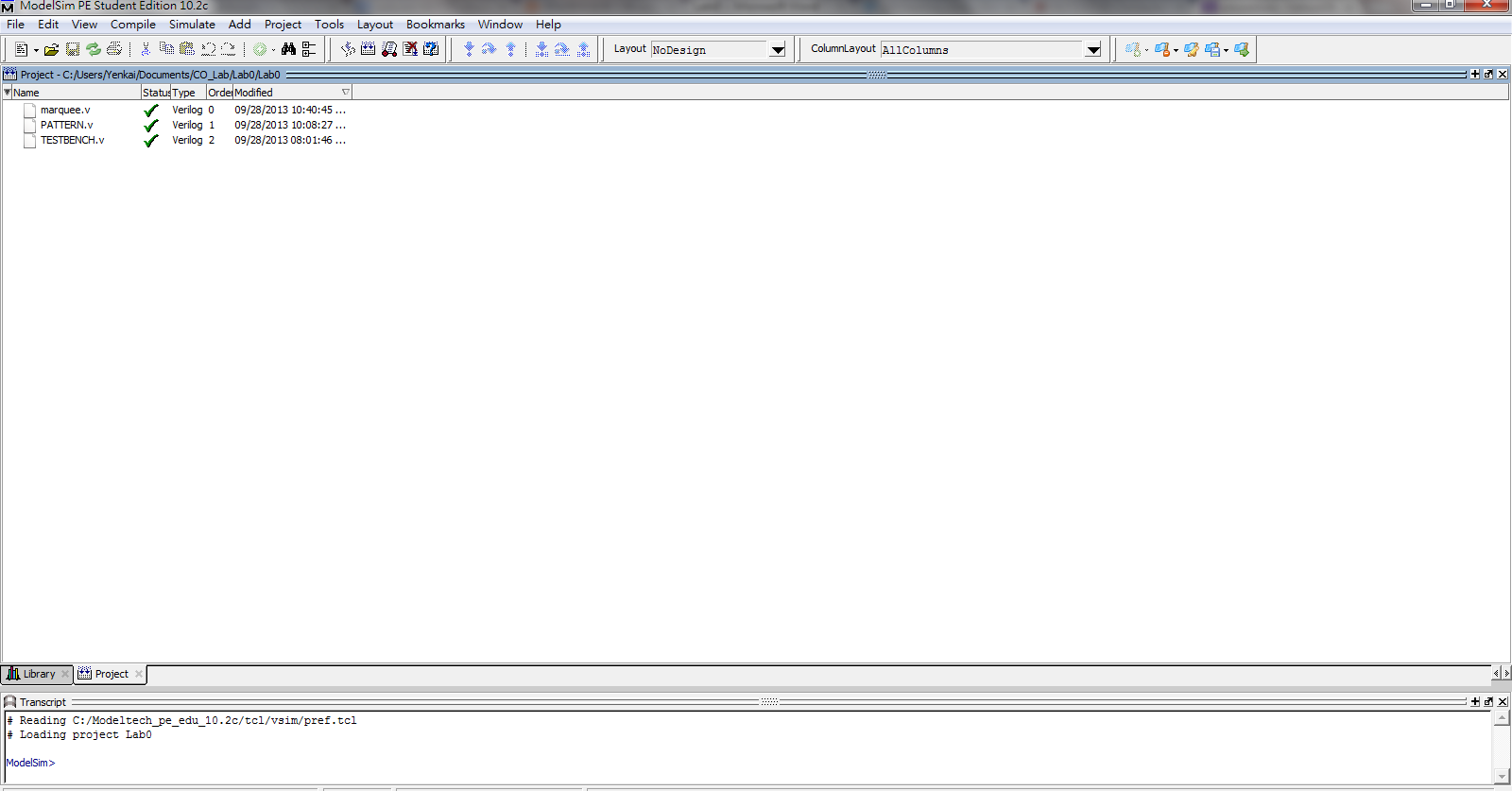
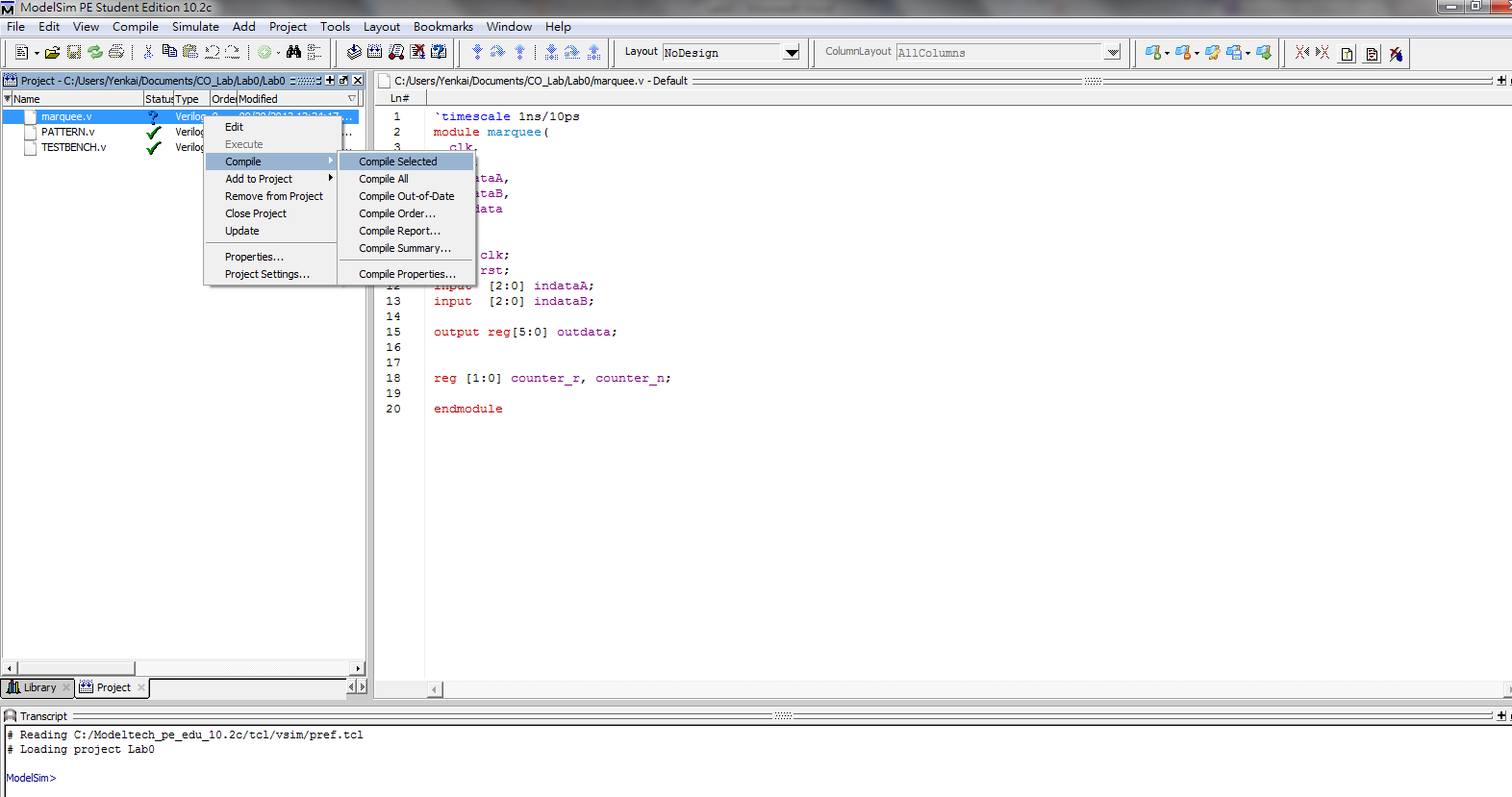
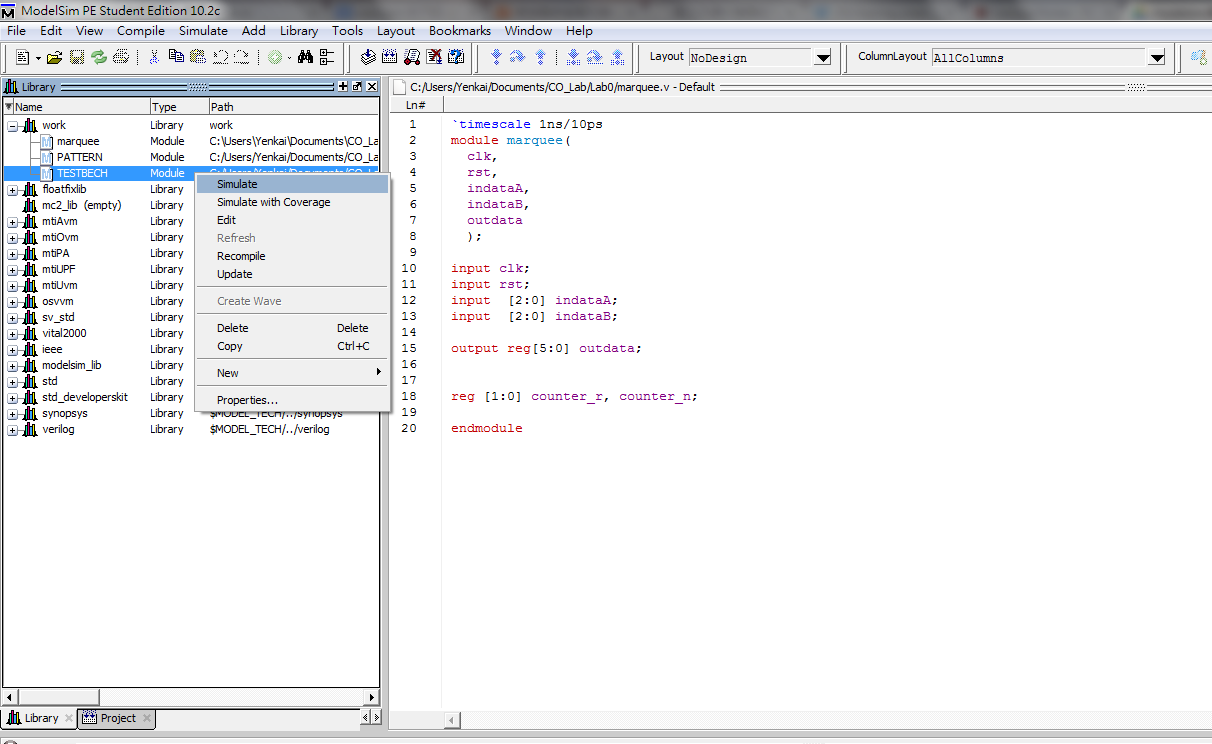
|  |  |  |
| --- | --- | --- |
| Counter | operation |  |
| 0 | OR | outdata = indataA | indataB; |
| 1 | AND | outdata = indataA & indataB; |
| 2 | XOR | outdata = indataA ^ indataB; |
| 3 | MERGE | outdata = {indataA , indataB}; |

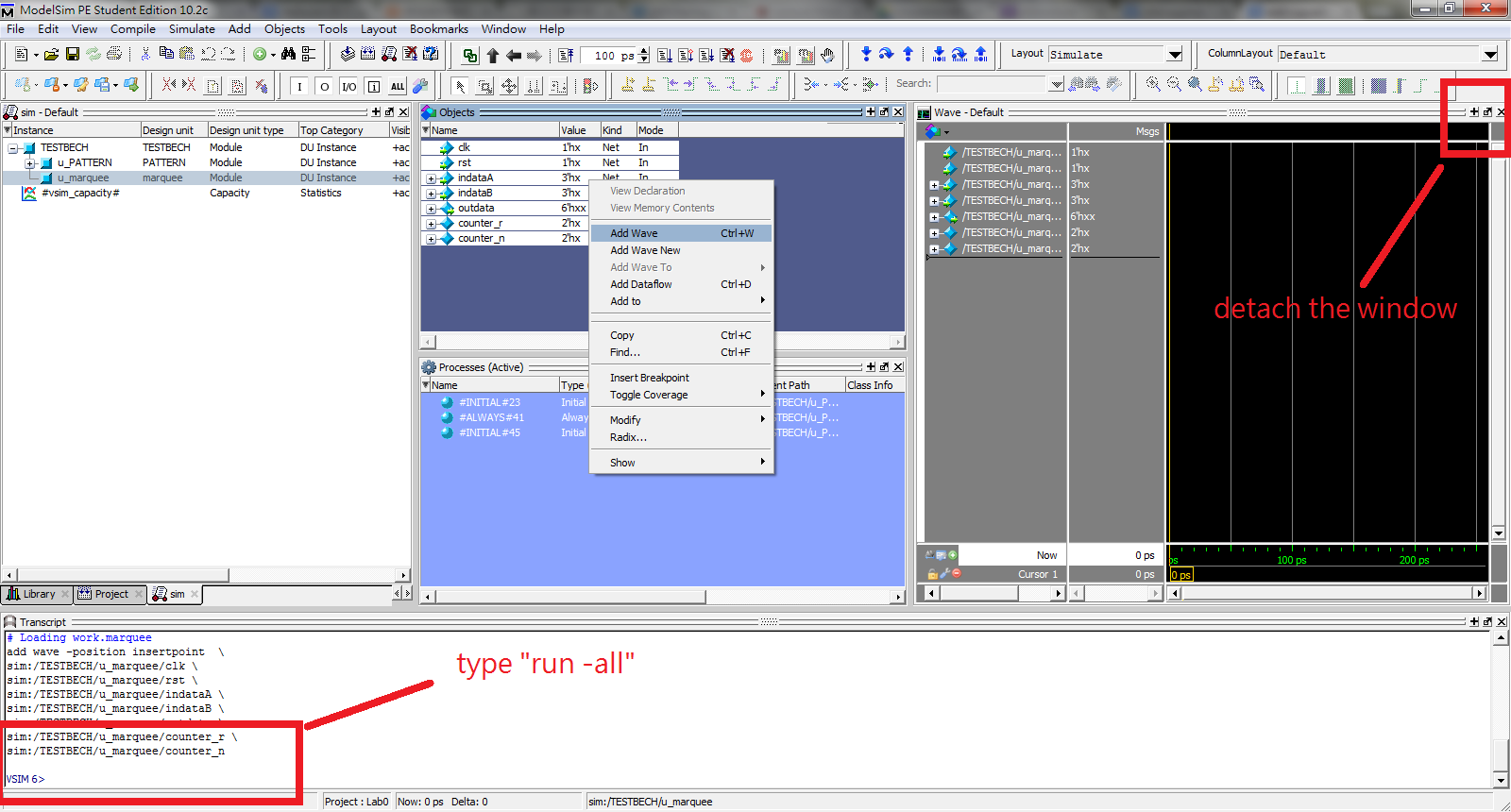
The waveform is as followed:

  
 The reset signal is sent at the falling edge of the clock. When the reset signal comes back to zero, your internal counter has to accumulate 1 at the rising edge of the clock.

The operation depends on the value of the counter. For example, the output is the bitwise OR of indataA and indataB when counter is 0.

**Design procedure**

1. Create new project and import three files:   
   marquee.v , TESTBENCH.v , PATTERN.v
2. marquee.v is the design template.   
   Modify this file to match the specification.  
   Compile your code before the simulation.
3. Select TESTBENCH.v to start the simulation.
4. Select signals to trace the waveform for debuging



1. Type “run -all” in the command line below. The message will show on the transcript. It tells you whether you pass or fail the test.