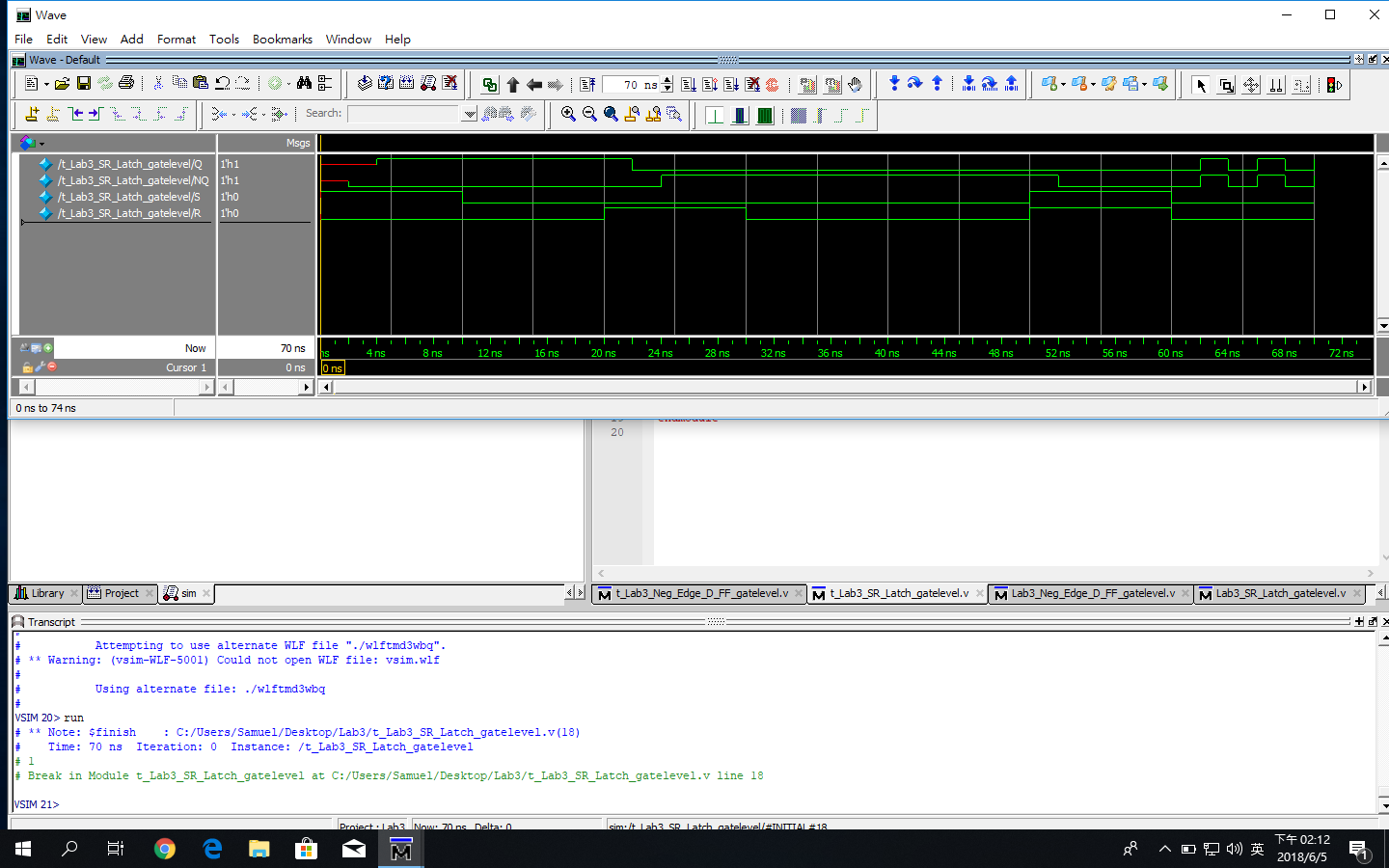
(1)

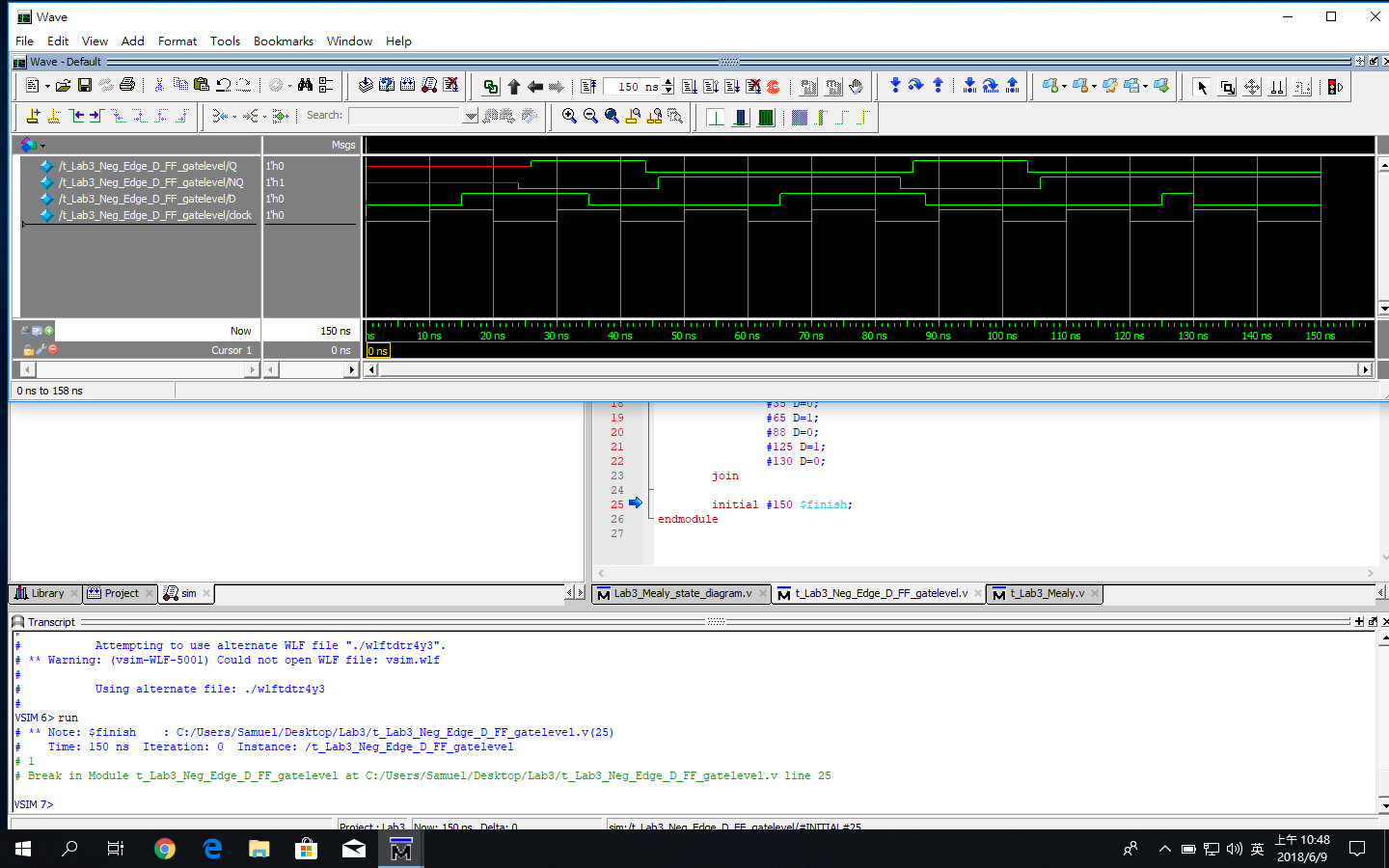
2A 之模擬結果波形圖

S=1, R=0為set狀態，經過4ns的propagation delay後，Q變為1而NQ變為0，S變為0，此時S=0, R=0，輸出不變。

20ns時S=0, R=1為reset狀態，經過4ns的propagation delay之後，Q變為0而NQ變為1，R變為0，此時S=0, R=0，輸出不變。

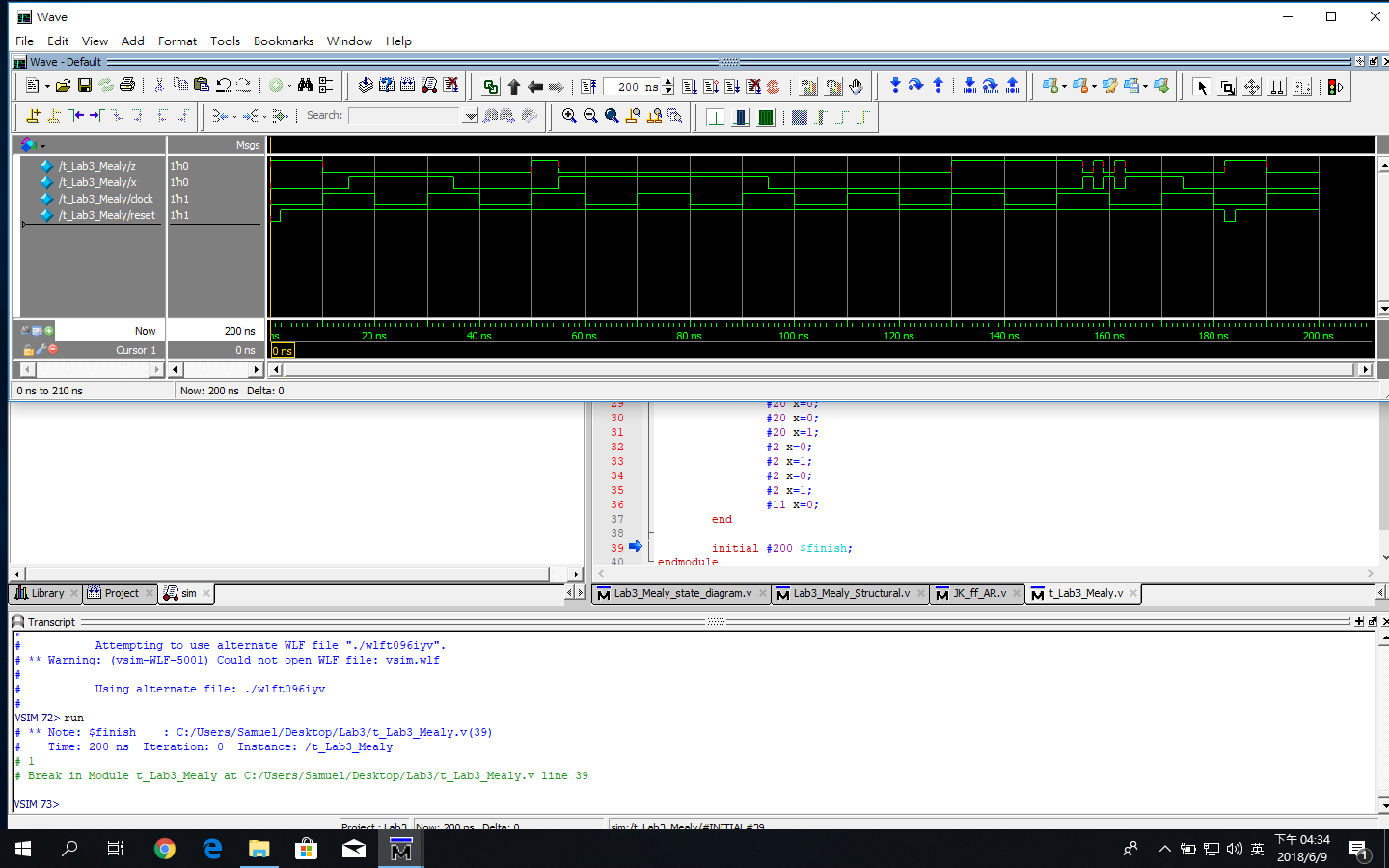
50ns時S=1, R=1為indeterminate狀態，因此產生了不斷亂跳的波形。

(2)

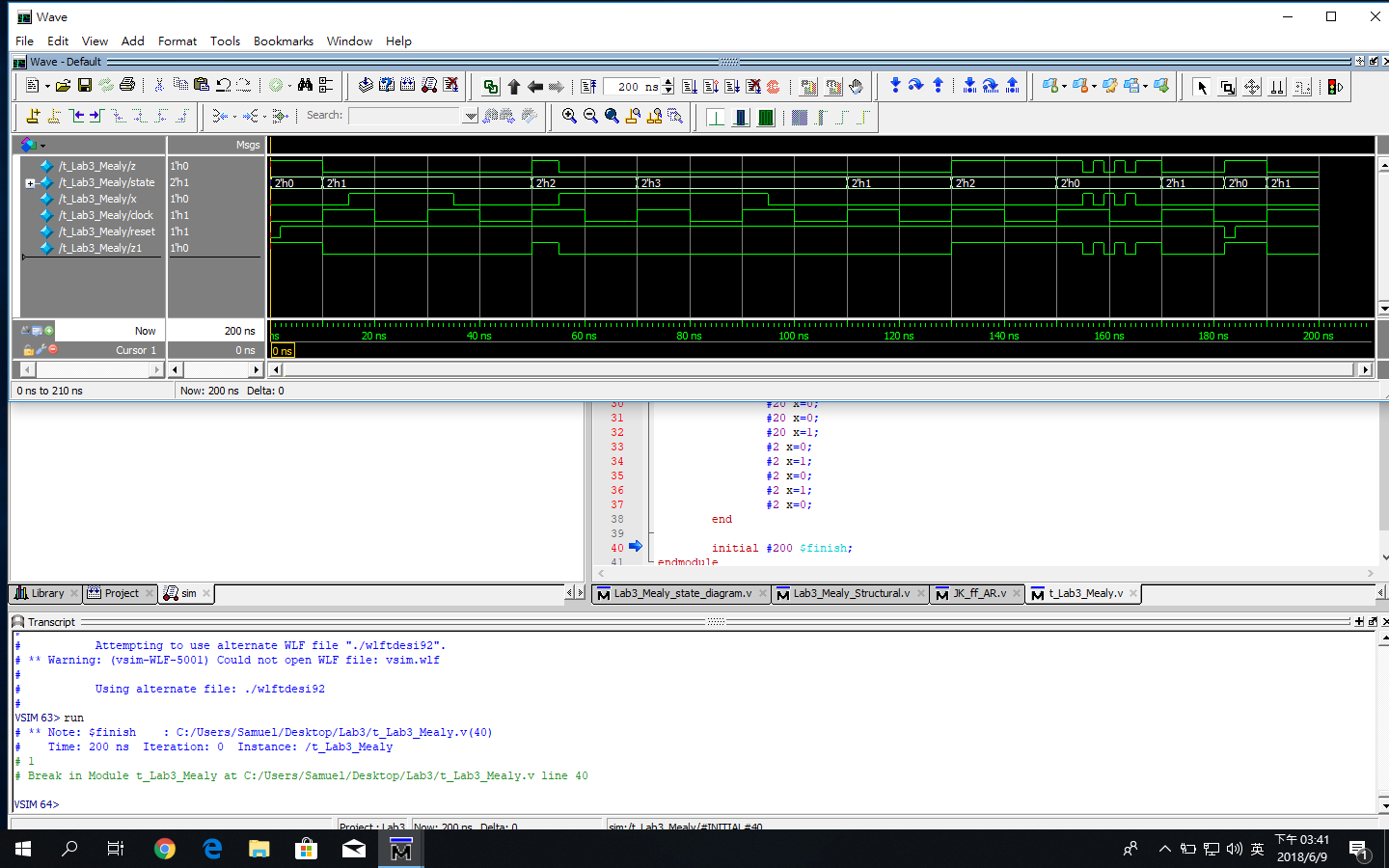
2B 之模擬結果波形圖

20ns時的第一個negdge clock來之前，輸出尚未產生，因此出現紅色的線段。20ns之後，negedge clock來了，此時D為1，經過propagation delay之後，Q變為1而NQ變為0。以此類推之後的波形，與模擬的結果相同，因此應該是正確的。

(3)

2C 之模擬結果波形圖

now的2C模擬結果波形圖



Testbench設計方法：

測試在每個狀態下，input為0和1時，產生的output與next都符合定義。首先reset到S0，接著輸入0到S1，接著輸入1到S1，然後輸入0到S2，接著輸入1到S3，然後輸入1至S3接著輸入0到S1，然後輸入0到S2，接著輸入0到S0，然後輸入1到S3，接著reset到S0。

說明：output與state狀態波形結果皆與定義相同，因此可以認為使用State diagram based model與Structural model 方式寫的程式是對的。

(4) 心得與感想、及遭遇到的問題或困難

這次作業比前兩次難更多，因為會有同步的情形發生，而非都是由上往下有時間次序，而這次還有Edge trigger要討論，不過作出這些Flip-Flop其實多少有點成就感，但有些語法實在是摸很久，透過程式終於能更加瞭解SR Latch在遇到indeterminate情形時電路波形結果是怎麼呈現。