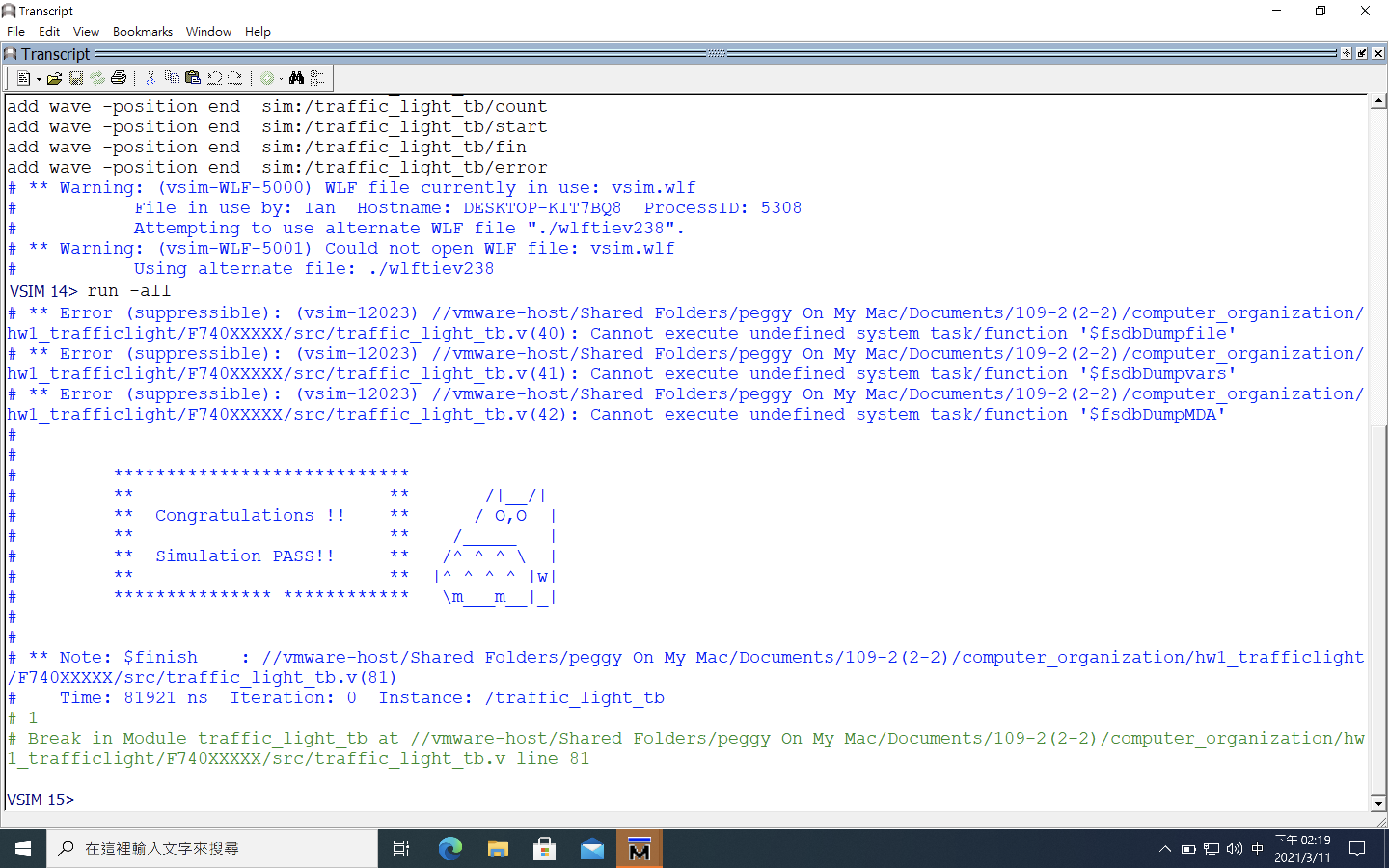
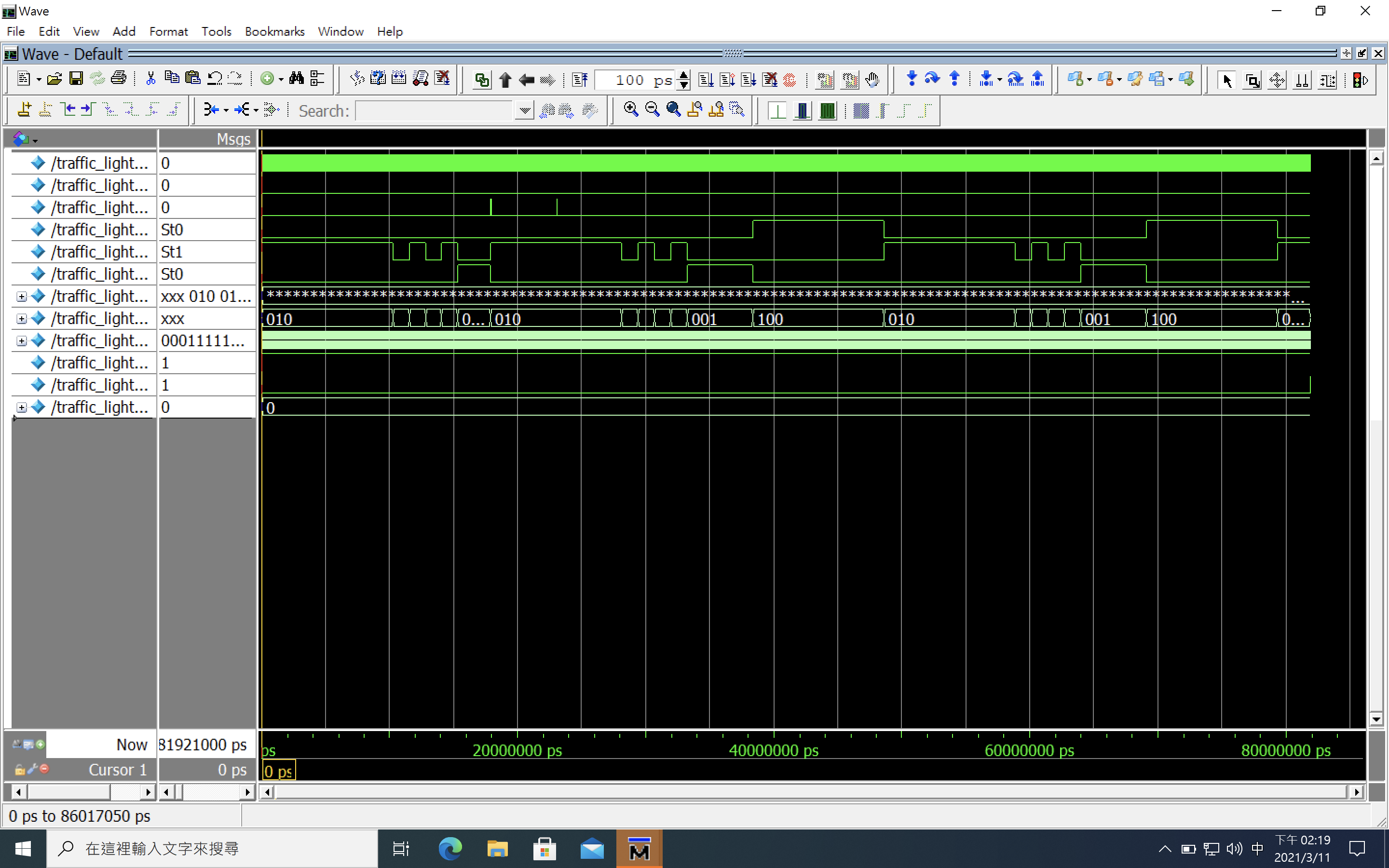
**Computer Organization 2019**

**HOMEWORK 1**

系級: 不分系112 學號: F64081020 姓名: 劉翊安

實驗結果圖:(波形圖及模擬完成截圖)

模擬完成截圖：

波形圖：

程式運作流程:(簡單說明波形變化的意義)

燈號變化的規則：

if(輸入端輸入pass訊號，且當前狀態非前1024個狀態的綠燈) 下一個cycle跳回第一個cycle。

else if (輸入端輸入rst號)「立刻」跳回第一個cycle //因rst是非同步正緣觸發，不用等clk訊號

else clk來往下走一個cycle

燈號依據下列順序變化：

綠燈1024 cycles -> 無燈128 cycles -> 綠燈128 cycles -> 無燈128 cycles -> 綠燈128 cycles -> 黃燈 512 cycles -> 紅燈 1024 cycles -> 回到第一個綠燈

程式碼說明：

電路拆成traffic\_control, datapath 兩個部分

1.traffic\_control：

traffic\_control會傳進儲存現在算到第幾個cycle的變數 clock\_counter。透過現在的狀態以及吃到的訊號判斷在這個cycle要輸出什麼燈號。利用next\_state紀錄現在這個cycle應該亮什麼燈。利用current\_state記錄要輸出什麼燈，最後輸出current\_state。

判斷輸出什麼：

if(rst訊號來）current\_state = 綠燈

else if(pass訊號來且不是第一個綠燈狀態）current\_state = 綠燈

else if(clock\_counter == 0）current\_state = 綠燈

else current\_state = next\_state //輸出現在這個cycle對應的燈號

判斷next\_state：

透過判斷現在是第幾個cycle來決定next\_state。

2.datapath：

用clock\_counter計算現在跑到第幾個cycle。cycle以一整個循環為基礎計數，從綠燈第一個cycle開始數到紅燈最後一個cycle。在每一次clk訊號來時將clock\_counter的值傳給traffic\_control。

在數的過程中：

if (rst訊號來) 立刻重數

else if(數到最後一個cycle）clock\_counter歸零

else if(pass訊號來且不是第一個綠燈狀態）clock\_counter歸零

else clock\_counter加一

心得(請寫下完成本次作業的心得、學到哪些東西、困難點的部分。)

在開始進行這次作業以前，第一件事是我無法成功的安裝軟體。在多次請教助教以後發現原來mac系統無法使用作業要求的軟體modulsim，因此在助教的協助下安裝的虛擬機軟體VMware，才終於把軟體下載下來。整個過程大概就花了三四天，這是我第一次接觸到虛擬機。

接著因為沒有修過數位系統實驗，是第一次接出電腦硬體語言verilog，熟悉這個語言又花了幾天。

雖然因為沒有修過數位系統實驗讓這一次作業變得比較困難，但我認為這是一個很好的學習機會，在短時間內學到很多。所有遇到的困難，都是我這次學到的東西。