

2023 Digital IC Design Homework 2

NAME	王聖凱
Student ID	N96114297
Functional Simulation Result	
Score	100
<pre> # ----- # ---- Simulation finish, ALL PASS, Score = 100 ---- # ----- # ** Note: \$finish : C:/Users/amdnclab/111-2IC/NewHW2/file/tb.v(139) # Time: 102 us Iteration: 1 Instance: /testfixture # ----- </pre>	
Description of your design	
<p>總共使用到兩個 stack，一個為儲存 data(order stack)，一個為儲存進站火車(stack stack)；變數部分有使用到 answer(此次測資共有幾台火車)、count(計算從 stack 成功 pop 出去的數量，用來判斷是否跟 answer 一樣，一樣即代表此順序可以)、digit(push 進 stack 的火車編號)、ptr1(用來當作 order stack 的 index)、top1(用來維持 data 一定在 order stack 最上方的 index)、ptr2(用來當作 stack stack 的 index)；狀態分為 4 種，表示 stack stack 為空的 IDLE、表示要做 push 的 PUSH、表示要做 pop 的 POP、表示判斷完要輸出 result 的 OUTPUT。</p> <p>reset 信號進來後做所有變數的初始化與清空兩個 stack。</p> <p>當有 data 進來時先判斷 answer 是否為 0，為 0 則代表這筆 data 為表示待會總共有幾台火車，並且把它存為 answer，反之，這筆 data 為判斷火車是否可以以這個順序出去的資料，將它依序存到 order stack。沒有 data 進來時則一直讓 answer 等於自己。</p> <p>狀態部分為：</p> <ol style="list-style-type: none"> 1. IDLE： <p>表示 stack stack 為空，為空時把 valid 拉低並把下個狀態轉為 PUSH，等待有測資進來。</p> 2. PUSH： <p>在 push 之前要先判斷目前進站火車的編號有沒有大於 data 所提供的火車數量，如果大於代表此順序不成功，將狀態轉為 OUTPUT，反之，將要進站的火車一一 push 進 stack stack，成功 push 完後要將 digit+1，讓接下來的火車編號正確，並把狀態轉為 POP。</p> 3. POP： <p>先判斷 count 是否等於 answer，等於即轉為 OUTPUT 狀態準備將 valid 拉高輸出 result=1，不等於則判斷目前 ptr2 是否小於 0，</p> 	

小於 0 代表 stack 已經清空，將狀態轉為 IDLE，ptr2 不小於則比較 stack[ptr2]是否等於 order[ptr1]，如果等於將 count + 1、ptr2 - 1(把 stack 頂端的數字 pop 掉)，ptr1 + 1(判斷下個順序可不可以)，並將狀態維持在 POP(有時候需要連續 pop)，如果 stack[ptr2]的值不等於 order[ptr1]，要 ptr2 + 1，讓進站火車繼續 push 到 stack 的頂端(將狀態轉為 PUSH)。

4. OUTPUT：

先判斷 count 是否等於 answer，等於則 result = 1，不等於則 result = 0。且到了 OUTPUT 狀態即表示完成一串 data 的判斷，所以要先將 valid 拉高輸出 result，輸出完後要將所有變數初始化與兩個 stack 要清空，以便繼續判斷。