ICLab Lab4-Leading Zeros Counter Report

104062625 周朋毅

這份報告我將分為以下兩點作說明：設計、測試以及結果討論。我會先談論如何構想整份作業以及這份作業詳細的組織，配合測試其正確性的資料，就實驗結果和個人心得作探討。

設計

就我所知，Leading Zeros Counter的有不只一種實作方法，而一開始我一直深怕著如果使用原本在C語言的寫法來寫的話（也就是使用for迴圈來計算），有可能會讓lab5的要求（也就是合成lab4的電路）無法達成。還好後來問了助教之後才知道，其實使用for迴圈是可行的，於是我就設計了兩個state來實作這次的電路。

這兩個state分別是INPUT和OUTPUT，其中OUTPUT負責觸發OVALID的信號，以及將一些計算用的變數歸零，其餘的工作都是由INPUT來負責，包含不論input是valid與否、是使用turbo mode與否、以及計算零的個數等等。

最一開始的版本幾乎就是照著C語言的寫法來實作的，果然就如老師上課所告誡的一般，這個版本產生了非常大量的錯誤，最主要的錯誤就在於，因為我把for迴圈寫在state transition的always block裡面，所以會有重複計算的情形發生。

後來我就把for迴圈獨立出來到一個always block裡面，配合FSM常用的延遲更新方法，就把上面提到的問題給解決了。至於如何不要多算零，或是在turbo mode中如何提前終止計算，我都是使用一些flag或是counter配合延遲更新來實作的。

測試及結果討論

這次我先使用README檔裡面的小測資作為我debug的基準，小測資裡包含了normal mode以及turbo mode中有提前終止以及沒有提前終止，一共三種情況，當我終於把bug都找出來並修正之後，就拿這次作業有發放的test pattern來測試，一經測試才發現我還有兩個地方沒有修正，其中一個是即使在發現有1的情況下，我還是會多算了後面的0；以及最莫名的，關於我for環圈所使用的變數i，不知原因為何的overflow了。第一個問題在我多設了一個flag後就解決了；第二個問題比較麻煩，我後來不停的嘗試才發現是因為別的變數有錯誤，才導致i出現錯誤。

另外這次作業比較有趣的地方就是初次使用了Github，一開始還因為沒有設定remote path所以我沒辦法把已經commit的版本同步到server上，後來都是不停的在錯誤中學習，雖然還沒有完全掌握Github的使用精隨，但是這個工具對於大型一點的專案開發真的是太有幫助了。

而這次有提到要把test pattern中一些錯誤找出來，經過模擬之後，發現在lzc\_w4c4gold.dat以及lzc\_w16c16gold.dat分別有1個以及21個錯誤，當時我還以為是我的design又有甚麼沒有考慮到的錯誤，但是看完test pattern並人工算了一遍之後，才發現那是偷偷藏起來的小錯誤。