**Computer Organization 2022**

**HOMEWORK 3**

系級: 資訊113 學號: F74096247 姓名: 陳映伃

**實驗結果圖:**

(波形圖及模擬完成截圖)

**一張含有 文字, 電子用品 的圖片

自動產生的描述**

**一張含有 文字, 螢幕擷取畫面, 室內 的圖片

自動產生的描述**

**程式運作流程:**

(簡單說明波形變化的意義)

1. reset時，所有值皆為0，除了valid，valid在isstring與ispattern皆為low時為零。
2. 當isstring為high時，我設了一個count\_s以及len\_s來數string的長度，同時將chardata存入我的string陣列中，當當isstring為low時，count\_s會歸零，len\_s會紀錄string的長度。
3. ispattern為high時，也是一樣的概念，只是‘^’和‘$’不會存入pattern中，會設立一個flag\_begin和flag\_end另外處理（遇到這兩個特殊符號，flag會為1）。
4. 當isstring與ispattern皆為low時，valid為1，開始比對string和pattern，若match，則會看flag\_begin和flag\_end是否為high，如果是，則判斷首尾，最後若match仍為1，則輸出match以及match\_index。
5. 最後，比對完一個pattern，match, match\_index, flag\_begin, flag\_end皆會歸為0，等待下一個pattern輸入（或是string）。

**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。)

我這次的作業是用for迴圈去比對string與pattern的，我明白使用verilog不太會用for loop來寫，但我覺得這個方式比較直觀。

我覺得這次作業最大的困難點是處理特殊符號的部分，因為我是match = 1時去判斷特殊符號，如果不符合match改為0繼續比對，這樣造成match不太好追蹤，修改了幾次。

這次，我最大的收穫大概就是透過波形來修改我的code吧。因為一開始波形很亂，我先是整理寫法（該歸零的歸零），讓波形變得易懂之後，錯誤也更好找出來了。

當然這次也有犯一些簡單的錯誤，像是count值設太小，導致overflow等等，表示細節還是要注意一下。

以上是我本次作業的新得。