**Computer Organization 2022**

**HOMEWORK 1**

系級: 資訊113 學號: F94081076 姓名: 郭立晨

**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, 螢幕擷取畫面 的圖片

自動產生的描述**

**一張含有 文字 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)

當正緣發生時，執行第一個always區塊，並檢查reset狀態，若reset為1則不執行動作，僅將所有變數設為預設值；反之，若為0則使用non-blocking寫法同時將陣列中的每個元素往下一個位置放置，也同時將最新的input X放入陣列第一個元素，且計算出陣列中所有元素的和。反覆執行以上步驟直到input X達到九個，則下一步可以開始求近似平均的步驟。隨著input X持續更新陣列的元素，sum跟著改變而使第二個always區塊被觸發，並使用blocking寫法求出平均數再找出陣列中最接近平均數又不大於平均數的元素，最後並得到近似平均數Y。之後每更新一次輸入就會馬上輸出當前的近似平均數結果。

**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。)

本次作業作為新手練習我覺得很適合，題目不過於簡單，也不會太困難，可以很好的從實作中理解nonblocking與blocking要如何適當的使用。硬體描述語言與軟體的思維很不一樣，在某些狀況下nonblocking的寫法可以使效率大大增加，因為是同時並行的執行所有動作，就像電路的每個區塊都是同時在執行輸入輸出一樣。舉例來說，若是要將陣列中的每個元素向右移一位，因為是同時執行則不需要考慮暫存元素的值，可以直接寫出直接簡潔明瞭的語法。

雖然成功地寫出第一次的作業，但其實不知道自己寫的是否能有最佳的效率，希望能透過一次次的練習改進自己的寫法，減少語法上的累贅，以期望在實現實體電路時能有較低的成本與效率。

我認為這次作業遇到最大的困難是需要決定always的觸發條件，應該要使用正緣還是負緣才會比較容易，或是使用其他變數去觸發會更方便後面執行。有時候改了想解決的問題卻會造成其他地方出了問題，或是不小心忽略了時間上的執行先後順序而導致不理想的結果產生。感覺都是初學者很容易遇到且跟軟體混淆的問題，所以解決的方法還是累積經驗多練習，以增加解題的技巧與速度。