

NCTU-EE DCS-2019

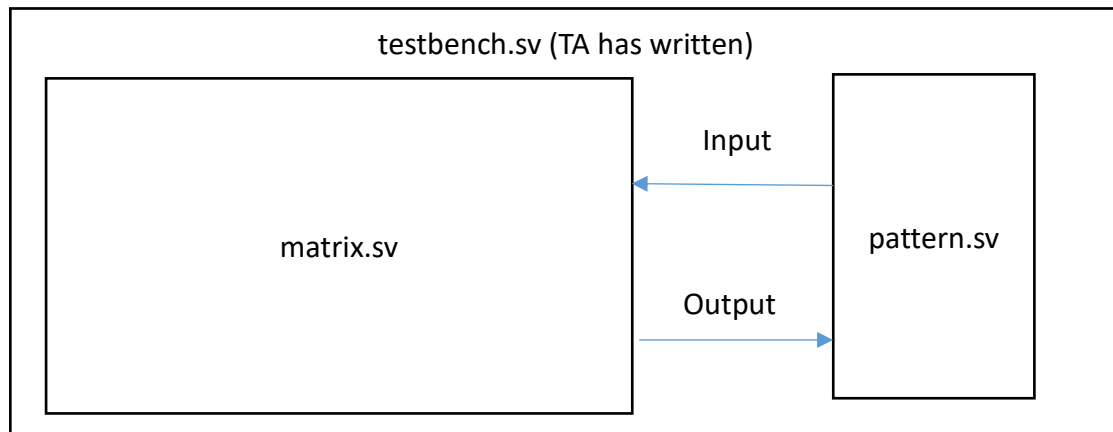
HW05

Design: Signed Complex Matrix Operation

資料準備

1. 從 TA 目錄資料夾解壓縮
% tar -xvf ~dcsta01/hw05.tar
2. 解壓縮資料夾 hw05 包含以下：
 - A. 00_TESTBED/
 - B. 01_RTL/
 - C. 02_SYN/

Block Diagram



設計描述

本次作業目標設計 2x2 複數(Complex)矩陣乘法運算硬體。

本次作業將 2 個 2x2 矩陣(A, B)相乘，兩個矩陣中的數值皆是複數，複數分為兩個部分，一是實部(real)、另一則是虛部(image)。

Input 矩陣中，實部和虛部的數字都是 7 bits 其內容由整數和小數組成，並用 2's complement 表示的有號數內容。(bit 6 is signed bit, bit 5 – bit3 is integer part, bit 2 – bit0 is decimal part)。

Output 矩陣中，一樣有實部和虛部，但數字只有將 decimal part round down 的 integer part。

矩陣乘法舉例如下：

$$A = \begin{bmatrix} -2 + 5.875i & -4 - 1.5i \\ -0.875 + 5.875i & 2.625 - 4.75i \end{bmatrix}$$

$$B = \begin{bmatrix} 1.125 + 3.875i & -5.25 + 7.125i \\ -6.5 + 7.875i & -5.125 - 5i \end{bmatrix}$$

$$A \times B = C = \begin{bmatrix} 12 + 23i & -19 - 18i \\ -4 + 54i & -75 + 26i \end{bmatrix}$$

注意：負數(negative number) round down 是取較小的數字，例如：-22.89i round down 為-23i

$$A = \begin{bmatrix} x11 + y11i & x12 + y12i \\ x21 + y21i & x22 + y22i \end{bmatrix}, B = \begin{bmatrix} z11 + w11i & z12 + w12i \\ z21 + w21i & z22 + w22i \end{bmatrix} \text{ Input 矩陣 A, B}$$

輸入會依照以下順序：

實部 in_real : x11 → x12 → x21 → x22 → z11 → z12 → z21 → z22

虛部 in_image : y11 → y12 → y21 → y22 → w11 → w12 → w21 → w22

本次作業有一特殊訊號：**busy**

這個訊號是由同學 design 給予 pattern，當 busy 訊號為 0 時，pattern 會按照上述順序在 nededge clock 持續給予 input 資料，且 in_valid 拉起。

反之，busy 訊號為 1 時，pattern 則為暫停給予 input 資料，in_valid 下降，代表硬體在運作。

Out_valid 訊號，同學們可以設計在任意時間拉起、也可以等 in_valid 落下後，等多個 cycle 拉起也可以。

當 out_valid 拉起，pattern 會檢查資料，依照以下順序

$$C = \begin{bmatrix} O11 + K11i & O12 + K12i \\ O21 + K21i & O22 + K22i \end{bmatrix}$$

實部 out_real : O11 → O12 → O21 → O22

虛部 out_image : K11 → K12 → K21 → K22

當 4 個檢查完，第 5 個 out_valid 則是檢查下一組 input matrix 相乘結果，以此類推。

故 out_valid 可以不連續。

Performance 說明：

這次 performance 採取 A(area) x T (simulation time)。

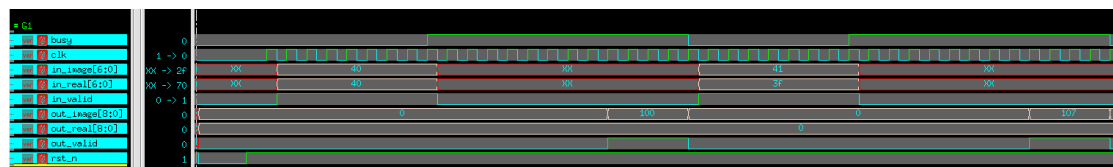
由於 input、output 可以透過 busy 訊號、out_valid 訊號控制，同學們可以自行參考要使用 FSM、pipeline，或是兩者。

同學們，可以修改 02_SYN 中 syn.tcl 的 25 行 **set CYCLE 5.0**，壓縮時間。

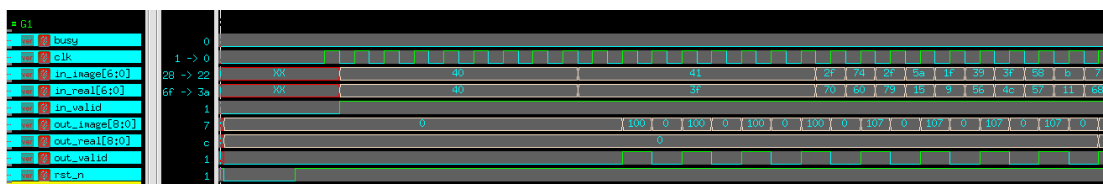
Cycle time default 為 5ns，並且不可高於 5ns。

注意：通常壓縮時間降低、面積會增加。

Waveforms



Out_valid 可以為連續的，並透過 busy 來防止 input 資料繼續輸入。



Out_valid 可以為不連續的，並完全不讓 **busy = 1**，持續讓 **input** 輸入。

Input

Signal name	Number of bit	Description
clk	1 bit	clock
rst_n	1 bit	Asynchronous active-low reset
in_valid	1 bit	當 in_valid 拉起，in_real 及 in_image 開始輸入。
in_real	7 bit	Input 實部。
in_image	7 bits	Input 虛部。

Output

Signal name	Number of	Description
out_valid	1 bit	out_valid 拉起，pattern 開始依照順序檢查，可以不連續。
out_real	9 bits	Output 實部。
out_image	9 bits	Output 虛部。

busy	1 bit	Busy = 0 時，pattern 開始依順序輸入 input，busy = 1 時，pattern 則停止輸入。
------	-------	--

Specification

1. Top module name : matrix (File name: matrix.sv)
2. 所有 output 必須為 0，在非同步負準位 reset。
3. Input 輸入順序及 output 檢查順序不得更改。
4. 02_SYN result 不行有 **error** 且不能有 **latches**。
5. Clock period 最高 5ns，可以自由設定增加 performance。
6. Clock period 以 0.1ns 為單位，例如 4.1ns, 3.2ns..., 不要有 3.17ns, 2.16ns...
7. Input delay = 0.5 * clock period; output delay = 0.5* clock period;

上傳檔案

1. Code 使用 09_upload 上傳。
2. report_dcsxx.pdf, **xx is your server account.** 上傳至 new e3。
3. **重要：synthesis time**，記得在 09_upload 資料夾中輸入 ./01_upload 5.0ns 上傳檔案，5.0ns 可以修改成你合成的 cycle time，如果輸入超出範圍視為 default 5ns。
4. 請 **5/24 9:00 am** 之前上傳

Grading Policy

1. Pass the RTL & Synthesis simulation. 50%
2. Performance(A*T) 40%
3. Report. 10%

Note

Template folders and reference commands:

1. 01_RTL/ (RTL simulation) ./01_run
2. 02_SYN/ (Synthesis) ./01_run_dc

報告請簡單且重點撰寫，不超過兩頁 A4，並包括以下內容

1. 描述你的設計方法，包含但不限於如何加速(減少 **critical path**)或降低面積。
2. 基於以上，畫出你的架構圖(**Block diagram**)或 **FSM diagram**。
3. 心得報告，不侷限於此次作業，對於作業或上課內容都可以寫下。
4. 遇到的困難與如何解決。