NCTU-EE DCS – 2017

Lab Test

Design: Matrix Multiplier

Data Preparation

1. Extract test data from TA's directory:

   % tar xvf ~dcsta02/test.tar

2. The extracted LAB directory contains:
   a. Exercise/ :your design

Design Description and Example Wave

In this test, you need to implement a 2 by 2 matrix multiplier. There will be 2 input matrix and a number to rotate matrixes (at the first cycle). According to the input signal to rotate these two matrix and do the matrix multiplication.

Following is the example:

0111

The matrix A is $\begin{bmatrix} in1 & in2 \\ in3 & in4 \end{bmatrix}$, and the matrix B is $\begin{bmatrix} in5 & in6 \\ in7 & in8 \end{bmatrix}$

If in0 = 8'd0, out matrix = $\begin{bmatrix} in1 & in2 \\ in3 & in4 \end{bmatrix} \times \begin{bmatrix} in5 & in6 \\ in7 & in8 \end{bmatrix} = \begin{bmatrix} out0 & out1 \\ out2 & out3 \end{bmatrix}$

If in0 = 8'b0000_1001, out matrix = $\begin{bmatrix} in4 & in3 \\ in2 & in1 \end{bmatrix} \times \begin{bmatrix} in7 & in5 \\ in8 & in6 \end{bmatrix}$

$c0 = a1 \times b1 + a2 \times b3$
$c1 = a1 \times b2 + a2 \times b4$
$c2 = a3 \times b1 + a4 \times b3$
$c3 = a3 \times b2 + a4 \times b4$

(inN: Nth input of in[7:0]) 順時轉

(outN: Nth output of out[16:0])

in0[1:0] is the rotating number for matrix B, and in0[3:2] is for matrix A.

According to the value of in0[1:0] and in0[3:2] to turn the matrixes clockwise.

in 0

① a3 a1
a4 a2

輸出 4 cycle

② a4 a3
a2 a1

③ a2 a4
a1 a3

Inputs

1. All input signals will be **synchronized** at **negative edge** of the clock.

I/O	Signal name	Description
Input	**clk**	Clock
Input	**rst_n**	Asynchronous active-low reset. You should set all your outputs to 0 when rst_n is low
Input	**in_valid**	**in[7:0]** is valid when in_valid is high.
Input	**in[7:0]**	The unsigned **in[7:0]** is valid only when **in_valid** is high, and is delivered for 9 cycles. The first cycle is the rotating number, its range is from 0 to 8‘d15.

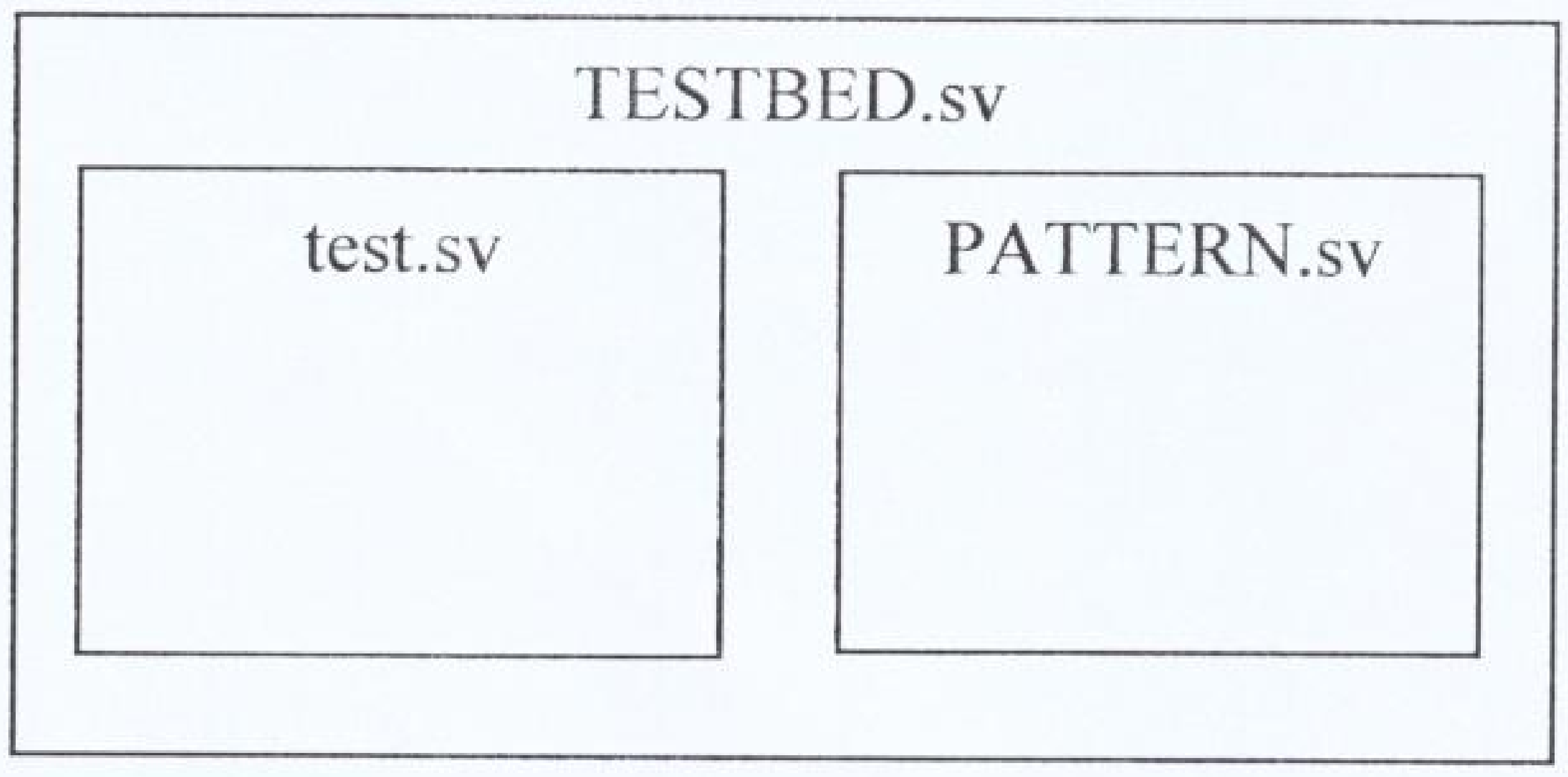
3ns latency < 100 76543210

Outputs

1. Output signals are synchronized at clock positive edge.
2. The TA’s pattern will capture your output for checking at clock negative edge.

I/O	Signal name	Description
Output	**out[16:0]**	**out** should output the answer sequence for continuously 4 cycles.
Output	**out_valid**	**out_valid** should be low after initial reset and not be raised when **in_valid is high**. **out_valid** should be set to high when your output value is ready, and will be high for 4 cycles.

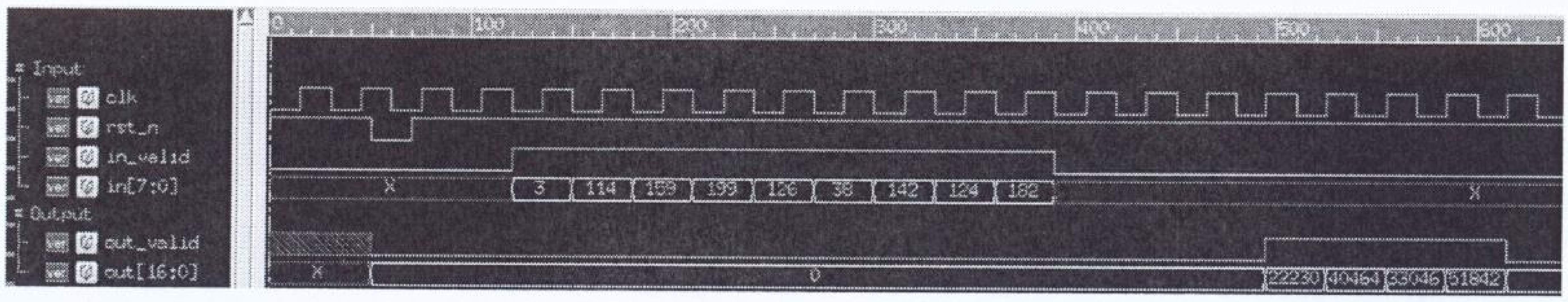
Block Diagram

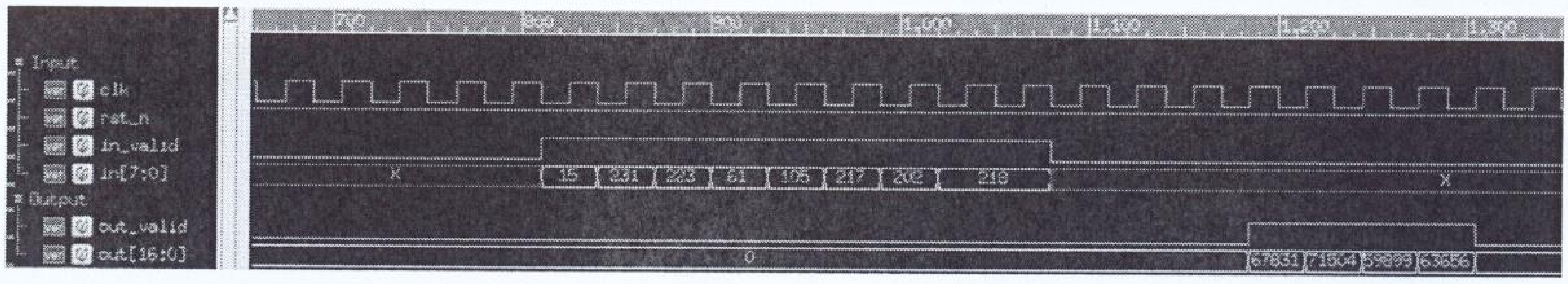


Note

1. Grading policy:
   a. Pass the RTL simulation
   b. Synthesis successful
      (**no Latch and Error**)
      (**slack in the timing report is non-negative**)
   c. Using For loop is not allowed.
   d. All registers should have reset function.
   e. Plagiarism is not allowed.
2. Template folders and reference commands:
   a. 01_RTL/ (RTL simulation) **./01_run**
   b. 02_SYN/ (Synthesis) **./01_run_dc**
      (Check the design which contain **latch** or not in **syn.log**
      (Check the design's timing in /Report/**test.timing**)

Sample Waveform





Specifications

1. Top module name： **test** (Filename: **test.sv**)
2. Input ports: **clk, rst_n, in_valid, in [7:0]**
3. Output ports: **out[16:0], out_valid**
4. It is an **asynchronous active-low reset** and **positive edge clk** architecture
5. The clock period of the design is **3ns**.
6. The input delay is set to **1.5ns**.
7. The output delay is set to **1.5ns**, and the output loading is set to **0.05**.
8. The next group of inputs will come in **1~9** cycles after your **out_valid** is over.
9. After synthesis, please check **syn.log** file that can not include any **Latch & Error**.
10. After synthesis, you can check test.area and test.timing. The area report is valid when the slack in the end of **timing report** is **non-negative**.
11. The latency for one case should be smaller than **100 cycles**.