



Lab 8

pipeline

Simple operation with pipeline

- Mode0: $\text{in_1} * \text{in_2}$
- Mode1: $\text{in_3} + \text{in_4}$
- Mode2: $\text{in_1} * \text{in_2} * \text{in_3} * \text{in_4}$
- Think twice before writing code

pipe.sv

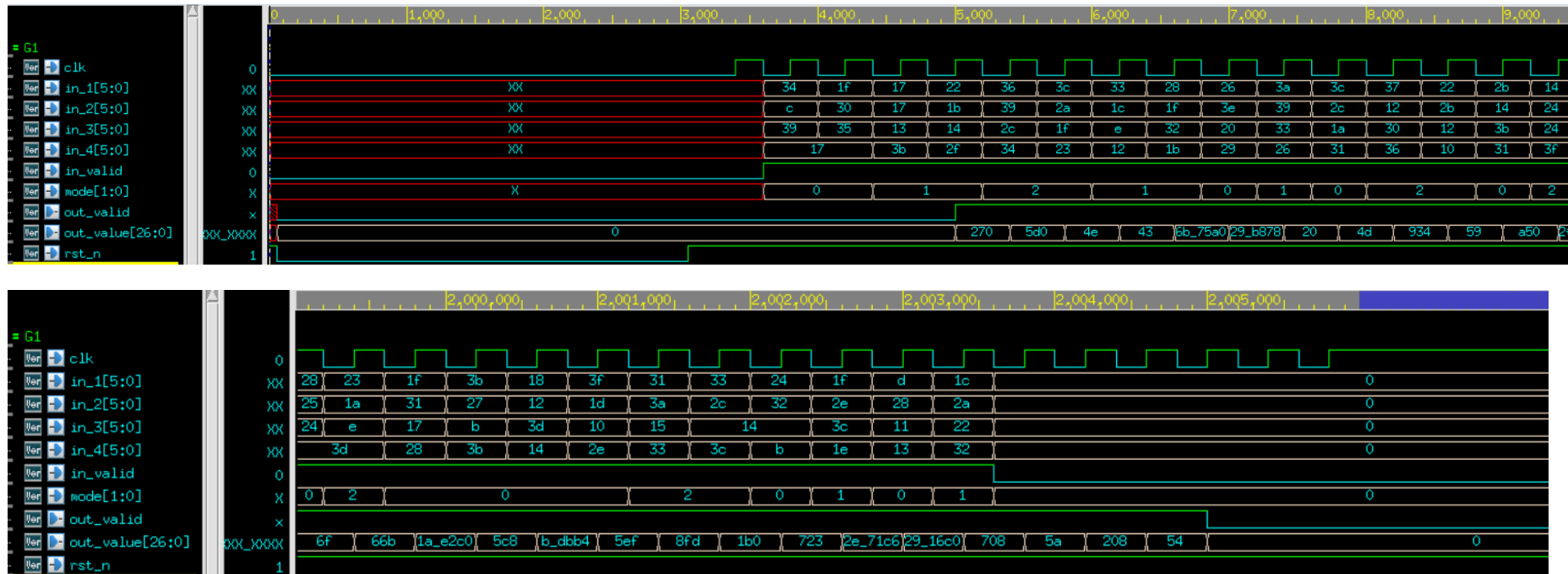
Input Signal	Bit Width	Definition
clk	1	4 ns
rst_n	1	Asynchronous reset when reset negedge, all output should be zero
in_1	6	Value:0~63, won't stop until pattern over.
in_2	6	Value:0~63, won't stop until pattern over.
in_3	6	Value:0~63, won't stop until pattern over.
in_4	6	Value:0~63, won't stop until pattern over.
mode	2	0 or 1 or 2, choose calculate mode
in_valid	1	High until all inputs are given
Output Signal	Bit Width	Definition
out_valid	1	High until all answers are done.
out_value	27	Output answer

Spec

- Top module name : PIPE (File name: PIPE.sv)
- All output signal should be reset (=0) after the reset signal is asserted.
- 01_RTL PASS
- 02_SYN result cannot include any **error** and **latches**.
- After synthesis, your slack in timing report **should be MET**
- 03_GATE PASS
- You must use **pipeline**. Since clock is 4ns.
- After output done , output signal should be zero.

Output & Waveform

- Waveform



In_invalid and out_valid pull high continuously

Command

- `tar -xvf ~dcsTA01/Lab08.tar`
- Upload
 - `cd 09_upload`
 - `./01_upload`
 - `./02_download demoX`

DEMO1 : 5/2 17:30:00

DEMO2 : 5/3 23:59:59