NYCU-ECE DCS-2023

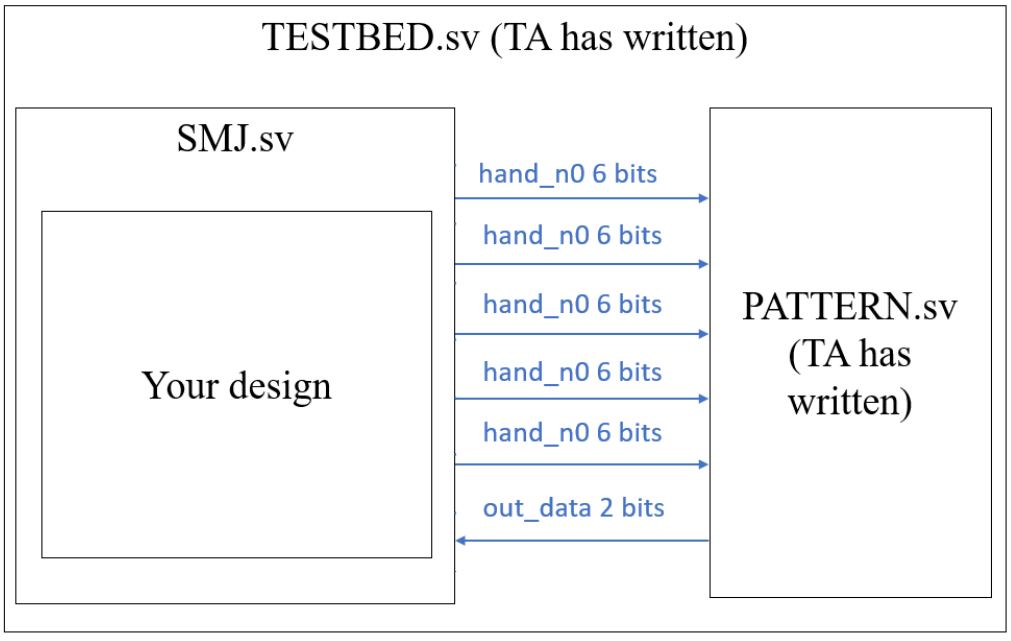
HW01

Design: Simplified Mahjong Judgment

Prepare materials

1. Unzip from the TA directory folder: % tar -xvf ~dcsta01/HW01.tar
2. Unzip the folder HW01 containing the following:
   a. 00_TESTBED/
   b. 01_RTL/
   c. 02_SYN/
   d. 03_GATE/
   e. 09_UPLOAD/

Block Diagram



Design Description:

The goal of this assignment is to complete a simple Mahjong judgment. You will receive five inputs representing five Mahjong tiles respectively, and finally output an out_data to determine whether the winning condition is completed, and if so, what type of win it is (either "sequence plus pair" or "triplets plus pair").

The five Mahjong tiles are composed of six bits, where the first two bits represent the type of tile: 2'b00 represents honor tiles(字牌), 2'b01 represents character tiles(萬子), 2'b10 represents bamboo tiles (索子), and 2'b11 represents dot tiles(筒子).

The remaining four bits represent the number on the tile. For honor tiles, 0~6 represent East, South, West, North, Red, Green, and White respectively. For the other three types, 0~8 represent the numbers 1~9 of that tile type.

When there is no winning hand, output 2'b00.
When the condition is impossible, such as having five same tiles or tiles that are unmentioned (ex: 001000, 111010…), output 2'b01.
When winning with tiles of a sequence and a pair, output 2'b10.
When winning with tiles of a triplet and a pair, output 2'b11.

For example: {hand_n0, hand_n1, hand_n2, hand_n3, hand_n4} =
{000000, 110001, 110011,000000, 110010} :
represent {East, 2 dots, 4 dots, East, 3 dots}, this meets the winning condition of Mahjong (with sequence). Out_data should be 2’b10

{000000, 111000, 111000,000000, 111000} :
represent {East, 9 dots, 9 dots, East, 9 dots}, this meets the winning condition of Mahjong (with triplet). Out_data should be 2’b11

{100010, 010010, 110100,010000, 110110} :
represent { 3 bamboos, 3 characters, 5 dots, 1 character, 5 dots }, this doesn’t meet the winning condition of Mahjong.
Out_data should be 2’b00

{001111, 111111, 011100, 010000, 110110} :
represent { unknown, unknown, unknown, 1 character, 5 dots }, appearing with unmentioned tiles.
Out_data should be 2’b01

Finally, a 2-bit number out_data will be output, and the pattern in testbench will test this 2-bit before the next set of input data comes in.

Inputs

Signal name	Number of bits	Description
hand_n0	6 bits	A random number in the range 0~63 First two bits: 2'b00 : honor tiles 2'b01 : represents character tiles 2'b10 : represents bamboo tiles 2'b11 : represents dot tiles Last four bits: For honor tiles, 0~6 represent East, South, West, North, Red, Green, and White respectively. For the other three types, 0~8 represent the numbers 1~9 of that tile type.
hand_n1	6 bits	
hand_n2	6 bits	
hand_n3	6 bits	
hand_n4	6 bits	

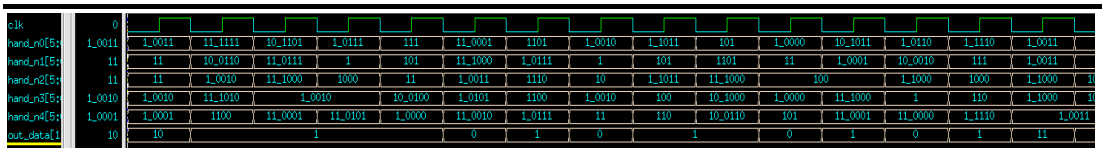
Outputs

Signal name	Number of bits	Description
out_data	2 bits	When there is nothing, output 2'b00. When the condition is impossible, such as having five same tiles or tiles that are unmentioned, output 2'b01. When winning with tiles of a sequence and a pair, output 2'b10. When winning with tiles of a triplet and a pair, output 2'b11.

Specifications

1. Top module name: SMJ (File name : SMJ.sv)
2. Please use Systemverilog to complete your homework.
3. Please use combination circuit to complete your homework.
5. 02_SYN can not have any error and latch.

Example waveform



Upload file

1. Please use 09_upload to upload code.
2. Please upload report_dcsxx.pdf to new E3, xx is your server account.

Grading policy

1. Pass the RTL& Synthesis simulation. 70%
2. Area 15%
3. Report 15%