2023 FPGA Midterm Exercise

Topic：Ten Point Half

I. Introduction

Ten-point-half is a kind of poker card game. There are two characters in this game. One is the dealer and the other is the player. The players aim to get larger values of cards in total than the dealer to win the game. In contrast, the dealer gets higher values than the players to win. The only restriction is that both players and dealer can't have their cards greater than 10.5 in total.

II. Lab Introduction

In this Lab, there are only 1 dealer and 1 player participating in the game. In the Beginning state, both of the two get 1 card which ranges from 1 to 13. The numbers larger than 10 represent a half-point in this game, and two cards of half-point should be 1 point. Both of player and dealer can decide to Hit Cards or not, but the limit number of cards in hand is five. (Including the first card in the Beginning state.)

Examples: In the Beginning state, the player gets 3 and the dealer gets 9.

<E.g.1>The player hit a card and the number is 7, which means he has 10 in total. At the same time, if the dealer chooses not to Hit Cards, the player wins.

<E.g.2>The player hit a card and the number is 5, which means he has 8 in total. At the same time, if the dealer chooses not to Hit Cards, the dealer wins.

<E.g.3>The player hit a card and the number is 6, which means he has 9 in total. At the same time, if the dealer chooses not to Hit Cards, the dealer wins.

<E.g.4>The player hit a card and the number is 6, which means he has 9 in total. At the same time, if the dealer chooses to Hit Cards and the number is 11, which means the dealer has 9.5 in total. The dealer wins.

<E.g.5>The player hit a card and the number is 6, which means he has 9 in total. At the same time, if the dealer chooses to hit a card and the number is 3, which means the dealer has 12 in total. The player wins.

<E.g.6>The player Hit Cards and the number is 3, then he hit a card again and the number is 6, which means he has 12 in total. At the same time, if the dealer chooses to hit a card and the number is 6, which means the dealer has 12 in total. The dealer wins. (In this case, since the player is already busted, no matter whether the dealer is busted or not, the dealer wins.)

Now we are going to talk about how to design it and some constraints you should know. First, you have to press the "**btn_m**" to enter the **Beginning state** in each round. The game will last for four rounds, then enter the **Done State**.

Beginning state

The player will get 1 card first and then the dealer will get another card from **LUT.v**. First, you need to instantiate the **LUT.v** into **tenthirty.v**. When you need to get a card, the "**pip**" signal rises, at the same time, the "**number**" signal will delay one clock cycle and then send a number to **tenthirty.v.** (This number represents the cards you get.) After the **Beginning state**, the FSM should enter the **Hit Cards state.**

NOTE：No seven-segment display is needed in this state.

Hit Cards state

You are using **LUT.v** for the Hit Cards state, the usage of **LUT.v** is the same as before. In this state, pressing the "**btn_m**" signal means to Hit Cards and the "**btn_r"** signal represents entering the next state. Start from the player, if pressing the "**btn_r"** signal, the turn goes to the dealer to Hit Cards. If the player hits a card, only one card will be sent at a time and the limit times of hitting cards are 4. (cause the total number of cards in hand is 5) If the player gets 5 cards in hand or he is busted, the turn immediately goes to the dealer. Also, the dealer can only hit a card at a time and if he gets 5 cards in hand or he is busted, the turn immediately goes to **Compare state.**

Compare state

- ✧ If none of the both is busted, the winner is the one who has bigger total values of cards in hand.
- ✧ If the total values of the two are the same, the dealer wins.
- ✧ If the player is busted, the dealer wins. (No matter what the value of the dealer is.)

After the **Compare state** is finished, 1 round is done. Press the "**btn_r**" signal to enter the next round. Also, press the "**btn_m**" signal to start the game.

Done state

When the 4th round is done the game ends.

Seven-Segment Display

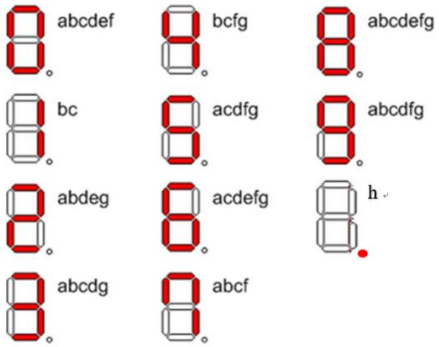


Fig 1. seven-segment display table

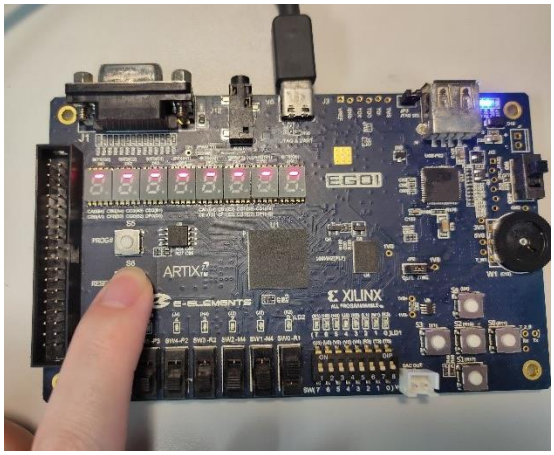


Fig 2. The Reset-display of the seven-segment display:

(1). Beginning state

In the Beginning state, you don't have to show the seven-segment values

(2). Hit Cards state

From the rightmost to the left are the values of cards that the dealer/player gets in this state. (The rightmost seven-segment display is the first card he gets in the **Beginning state** and the second seven-segment display is the first card he gets in the **Hit Cards state**, and so on…

Note: He can only have 5 cards in hand, so only 5 seven-segment display (from right to left) are used for showing each point of a card he got.

The leftmost three seven-segment display show total values, as the following figure shows:

Fig 3. seven-segment display handcard e.g 1

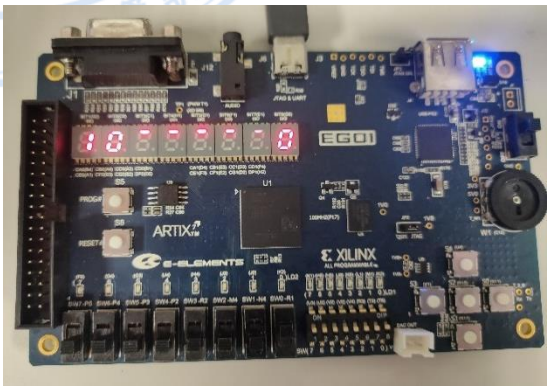


Fig 4. seven-segment display handcard e.g 2

(3). Compare state

In the **Compare state**, 3 of the seven-segment display on the leftmost are used for the display of the total values of the dealer's cards in hand, and 3 of the seven-segment display on the rightmost are used for displaying the total value of the player's cards in hand. (Even if the dealer/player is busted, the seven-segment display should show the value)

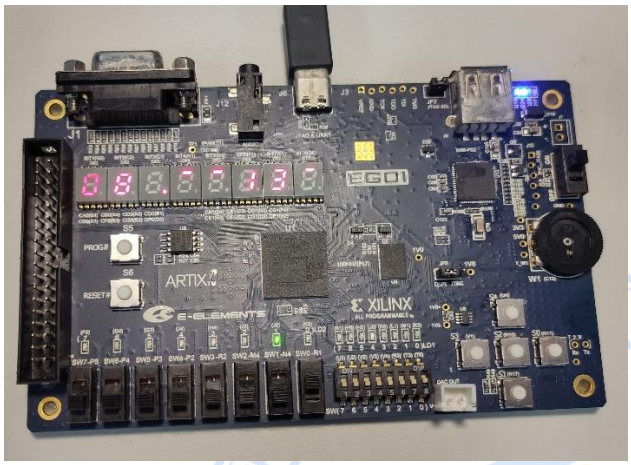


Fig 5. seven-segment display Compare state

LED

In the **Compare state**, LED are used for displaying the winner. The "**led"** signal from LSB to MSB are represented as the winner is the player, the winner is the dealer, and the **Done state** respectively. For other states (excludes **Compare state and Done state**) the "**led"** signal should be turned off.

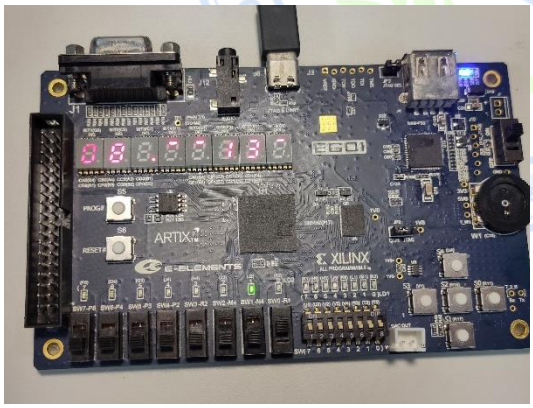


Fig 6. seven-segment display led e.g 1

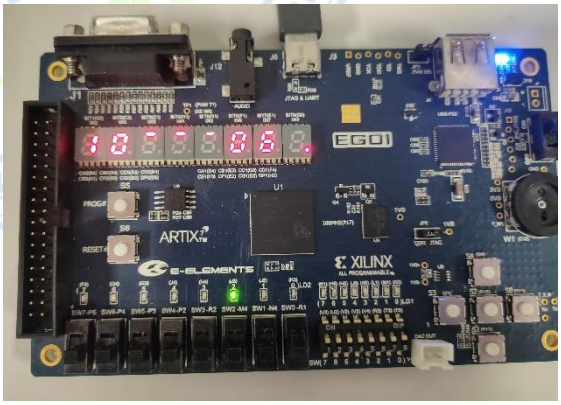


Fig 7. seven-segment display led e.g 2

The URL of the Demo videos is below:

https://www.youtube.com/watch?v=MHQ68WXCOEY&list=PLn0Y9lYJqqvGrmoE9heed0lfZpIog0h0&index=1

III. I/O Description

1. LUT.v

Signal Name	I/O	Width	Simple Description
clk	I	1	Posedge triggered Clock
rst_n	I	1	Asynchronous negedge Reset
pip	I	1	Pip 訊號拉起時，會將 number 訊號延遲一個 cycle 後送出
number	O	4	表示撲克牌數值，數值從 1~13

2. Tenthirty.v

Signal Name	I/O	Width	Simple Description
clk	I	1	Posedge triggered Clock
rst_n	I	1	Asynchronous negedge Reset
btn_m	I	1	表示 S2 按鍵
btn_r	I	1	表示 S0 按鍵
seg7_sel	O	8	七段顯示器選擇器，控制八顆七段顯示器何時亮
Seg7	O	8	右邊四顆七段顯示器(DK5~DK8)
Seg7_l	O	8	左邊四顆七段顯示器(DK1~DK4)
led	O	3	LD2(0、1、2)

IV. Specifications

---------- **If you violate this rule, your score will be "0"** ----------

1. Plagiarizing(抄襲) is NOT allowed, violators get "0" points in this lab.
2. In this lab, you CAN'T revise the codes of **LUT.v** and I/O in **tenthirty.v**.

----- **If you violate this rule, your score will be reduced by 20%** -------

3. You MUST use FSM in your design.
4. You CAN'T revise the part of "DON'T TOUCH" in **tenthirty.v**.
5. Please upload the file to iLearning 3.0 with the following format：" **tenthirty_studentID.v**" before 5/17.
6. You can only pull the "**pip**" signal when you need to draw a card.

If only a few students make this midterm project, TA will give the scores depending on the degree of completion of your design, please DON'T give up!

V. Grading Policy(All the grading policy below is 10% for each)

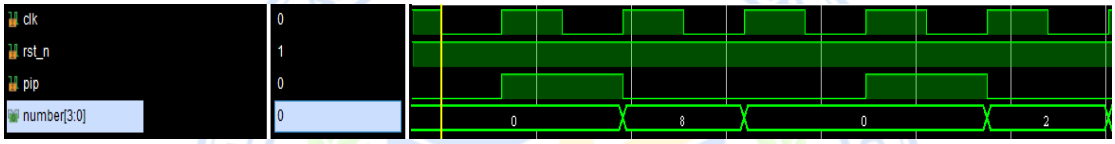
1. Press the “Reset” signal, the seven-segment display works functional correct.
2. When the “**btn_m**” signal is pressed to start the game, your design successfully complete the **Beginning state** and transition to the **Hit Cards state**. Additionally, the design can display the first card obtained by the player or dealer from **LUT.v.**
3. During the first round, the player's card points (and total points in hand) will be correctly displayed on the seven-segment display. Additionally, pressing the "**btn_r**" signal permits the dealer to draw additional cards.
4. During the first round, the dealer's card points (and total points in hand) will be correctly displayed on the seven-segment display. Moreover, pressing the '**btn_r**' button will transition the game to the 'Compare' state.
5. During the **Compare state** of the first round, the total points of both the dealer and the player should be correctly displayed on the seven-segment display, and the LEDs should be turned on accurately to indicate the winner. To end the **Compare state**, you must press the “**btn_r**” signal.
6. The player/dealer could successfully Hit Cards at least one time, and the seven-segment display shows the correct points of cards in their hands.
7. The player/dealer could successfully Hit Cards 4 times (total 5 cards in hand).
8. When the player/dealer is busted, your design could turn to the next state successfully. (The player turns to the dealer for Hit Cards, while the dealer turns to **Compare state**.)
9. If there is anyone is busted, the **Compare state** works properly in this case.
10. Your design could go through the entire 4 rounds and turn into the **Done state**.

<Bonus>

If using "one_shot_pulse" design in your work, you will get another bonus point.(2%)

VI. TA Remind ♥♥♥

1. The clock rate of "dis_clk" needs to work faster than "d_clk".
2. The seg7_temp register is only used for the display format for 8 seven-segment display. (i.e. seg7_temp0 controls the display format of the rightmost seven-segment display.)
3. You may need other registers for storing the number of cards in hand (for dealer/player), DON'T use seg7_temp for this purpose.
4. You may need 1 more bit for registers to store the half-point information, and remember, if there are two half-point cards in hand, the total points should be 1.
5. When using TB, remember to modify the "**d_clk**" signal into the format as **d_clk** = counter[5].
6. You need to self-define your own TB.
7. TB can only check your FSM design flow.
8. LUT Demonstration



9. TB example

```
initial begin
    rst_n = 1;
    set_initaial;
  gap = $urandom_range(1,5);
  repeat(gap)@(negedge clk);
  rst_n = 0;
  repeat(gap)@(negedge clk);
  rst_n = 1;
  repeat(gap)@(negedge d_clk);
  btn_m = 1;
  repeat(2)@(negedge d_clk);
  btn_m = 0;
  repeat(2)@(negedge d_clk);
  btn_r = 1;
  repeat(2)@(negedge d_clk);
  btn_r = 0;
  btn_m = 1;
  repeat(1)@(negedge d_clk);
  btn_m = 0;
  repeat(2)@(negedge d_clk);
  btn_m = 1;
  repeat(6)@(negedge d_clk);
  $finish;
end
```

10. FSM Demonstration