實驗主題(Lab15)

B083022053

黃啟桓

Exercise 1:

Exercise 1: Four-bit universal shift register

- Design and verify the four-bit universal shift register using Verilog HDL
  - structural (gate-level) modeling

Exercise 2:

Exercise 2: BCD ripple counter (1/3)

- Design and verify the two-decade BCD ripple counter using Verilog HDL
  - structural (gate-level) modeling

實驗內容

一、 Exercise 1:

Exercise 1: Four-bit universal shift register

- Design and verify the four-bit universal shift register using Verilog HDL
  - structural (gate-level) modeling

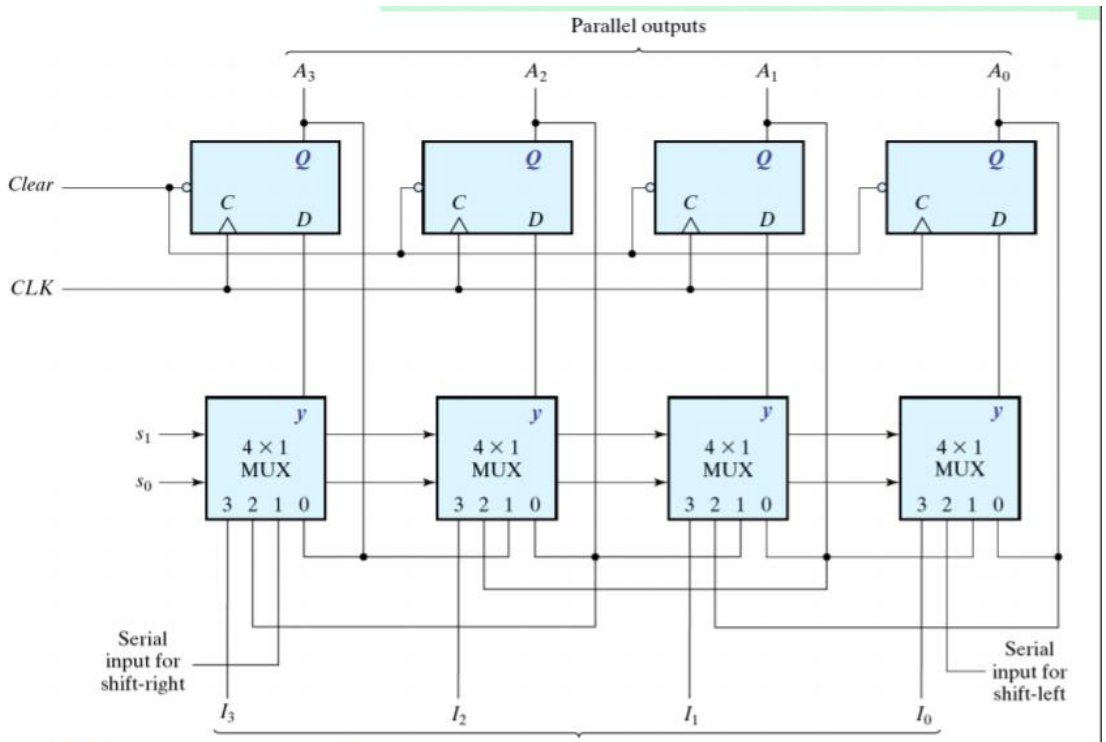


Fig. 6.7 Four-bit universal shift register (continued)

Verilog 程式碼見按照上圖連線即可，而上圖中有 4x1MUX 和 D flip flop 所以我們需要這兩個 submodule

```
`timescale 1ns / 1ps

module D_flip_flop (Q, D, Clock, Reset);
    output Q;
    input Clock, D, Reset;

    wire n1,n2,n3,n4,n5,n6;
    nand(n1, n3, Clock, Reset);
    nand(n2, n1, Clock, n4);
    nand(n3, n1, n4);
    nand(n4, n2, D, Reset);
    nand(n5, n1, n6);
    nand(n6, n2, n5, Reset);
    buf(Q, n5);
endmodule
module mux_4x1_beh(Y, I0, I1, I2, I3, Select);
    output  Y;
    input I0, I1, I2, I3;
    input [1: 0] Select;
    wire s0_b,s1_b,n0,n1,n2,n3;
    not (s0_b, Select[0]);
    not (s1_b, Select[1]);
    and(n0, I0, s1_b, s0_b);
    and(n1, I1, Select[1], s0_b);
    and(n2, I2, s1_b, Select[0]);
    and(n3, I3, Select[1], Select[0]);
    or(Y,n0,n1,n2,n3);
endmodule

module Shift_Register_4_beh (
output [3: 0] A_par, // Register output
input [3: 0] I_par, // Parallel input
input s1,s0, // Select inputs
MSB_in, LSB_in, // Serial inputs
CLK, Clear_b // Clock and Clear_b
);
    wire y0,y1,y2,y3;
    mux_4x1_beh MUX0(y0, A_par[0], A_par[1], LSB_in, I_par[0],{s1,s0});
    mux_4x1_beh MUX1(y1, A_par[1], A_par[2], A_par[0], I_par[1],{s1,s0});
    mux_4x1_beh MUX2(y2, A_par[2], A_par[3], A_par[1], I_par[2],{s1,s0});
    mux_4x1_beh MUX3(y3, A_par[3], MSB_in, A_par[2], I_par[3],{s1,s0});

    D_flip_flop DFF0(A_par[0], y0, CLK, Clear_b);
    D_flip_flop DFF1(A_par[1], y1, CLK, Clear_b);
    D_flip_flop DFF2(A_par[2], y2, CLK, Clear_b);
    D_flip_flop DFF3(A_par[3], y3, CLK, Clear_b);
endmodule
```


Name
Value
0 ns
20 ns
40 ns
60 ns
80 ns
100 ns
120 ns
140 ns
160 ns
180 ns
200.000 ns
s1
s0
clk
1
0
a
X
5
2
3
7
f

結果如上圖

二、Exercise　2:

Exercise 2: BCD ripple counter (1/3)

- Design and verify the two-decade BCD ripple counter using Verilog HDL
  - structural (gate-level) modeling

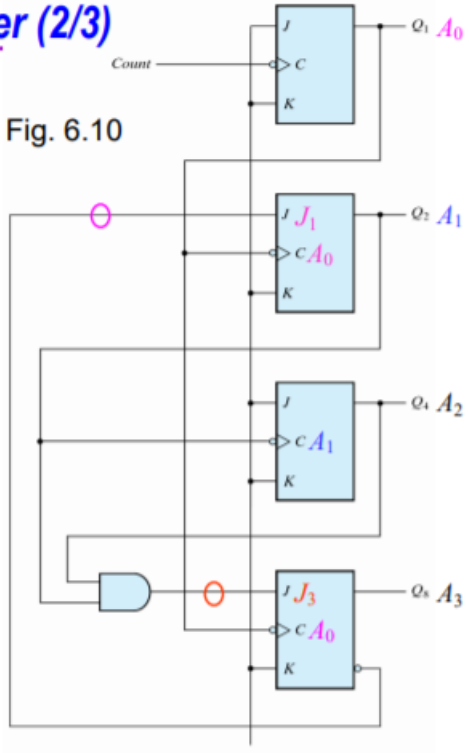
Exercise 2: BCD ripple counter (2/3)

- BCD ripple counter

A_3	J_3	A_2	A_1	J_1	A_0
0	0	0	0	1	0
0	0	0	0	1	1
0	0	0	1	1	0
0	0	0	1	1	1
0	0	1	0	1	0
0	0	1	0	1	1
0	1	1	1	1	0
0	1	1	1	1	1
1	0	0	0	0	0
1	0	0	0	0	1

J	K	Q(t + 1)
0	0	$Q(t)$
0	1	0
1	0	1
1	1	$Q'(t)$

Fig. 6.10



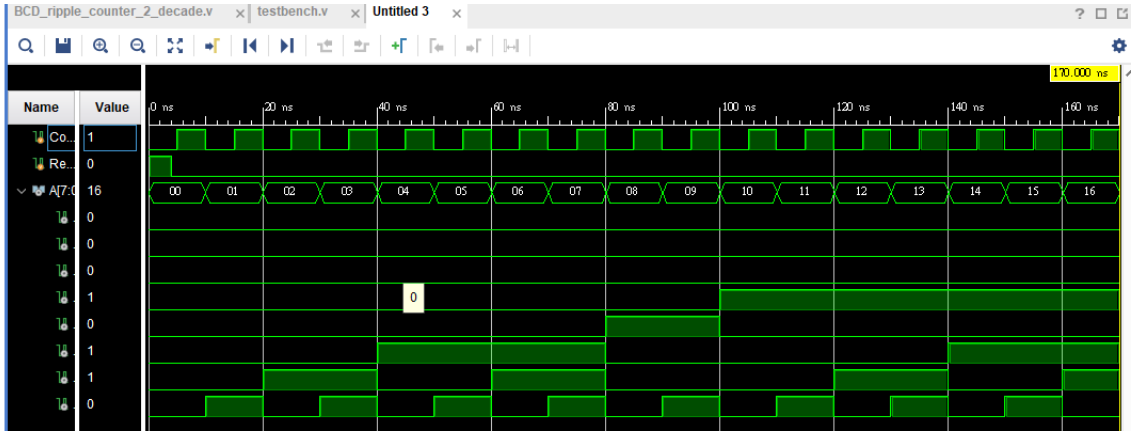
十進位計數器最重要的就是 truth table，

知道 truth table 就能推得 Fig. 6. 10

按照 Fig. 6. 10 連線就是 1 個 10 進位位元的輸出

把個位數的 A3 當成十位數 A0 的 Clock 就能產生 2 個 10 進位位元的輸出

```
3   module JK_ff (Q, clock, J, K, Reset);
4       output Q;
5       input J, K, clock, Reset;
6   //  reg Q;
7       wire Q_, K_, D_in, n1, n2,n3,n4;
8
9       reg Q;
10      not (Q_ , ~Q);
11      always @ (posedge clock, negedge Reset)
12      if (Reset == 0) Q <= 1'b0;
13      else Q <=  J&~Q|~K&Q;
14  endmodule
15
16  module BCD_ripple_counter (output [3:0]A,input clock, logic);
17      JK_ff JK0(A[0], ~clock, logic, logic, logic);
18      JK_ff JK1(A[1], ~A[0], ~A[3], logic, logic);
19      JK_ff JK2(A[2], ~A[1], logic, logic, logic);
20      JK_ff JK3(A[3], ~A[0], A[1]&A[2], logic, logic);
21  endmodule
22
23  module BCD_ripple_counter_2_decade (output [7:0]A,input clock, logic);
24      BCD_ripple_counter D0(A[3:0], clock, ~logic);
25      BCD_ripple_counter D1(A[7:4], A[3], ~logic);
26  endmodule
```



實驗心得

1. Structural level modeling 中邏輯閘使用函數表示
2. Structural level modeling 可以很直觀的轉換成 Schematic
3. Structural level modeling 需要宣告邏輯閘之間的線路
4. 所有的 modeling 都需要注意該邏輯閘是否有交換律或結合律