**電路實驗報告**

**資工二甲**

**11127137 黃乙家**

**11127141 梁凱哲**

1. Verilog Code

Counter.v

`timescale 1ns/1ns

module Counter\_and\_sel( dout, clk, rst, sel, a, b );

output reg[3:0] dout;

input wire clk, rst, sel;

input wire[3:0] a, b;

always @(posedge clk)

begin

if(rst)

dout <= 4'b0000;

else

if(sel)

dout <= dout + 1;

else

dout <= a & b;

end

endmodule

TM\_Counter.v

`timescale 1ns/1ns

module TM;

reg sel, clk, rst;

reg [3:0] a, b;

wire [3:0] dout;

parameter t = 200;

parameter th = 100;

Counter\_and\_sel U\_counter\_and\_sel( dout, clk, rst, sel, a, b);

always #th clk = ~clk;

initial begin

clk = 0;

sel = 1;

rst = 1;

a = 4'b0000;

b = 4'b0000;

#t rst = 0;

#(13\*t) sel = 0;

a = 4'b0101;

b = 4'b1001;

#t

a = 4'b1011;

b = 4'b0010;

#t $stop;

end

endmodule

1. 結果一張含有 文字, 螢幕擷取畫面, 軟體, 多媒體軟體 的圖片

   自動產生的描述

從圖中可發現 dout 每當 Clock 訊號往高位時且sel為1時會+1，到了第15個週期sel變為0，dout設為a&b，結果正確。

1. 心得

黃乙家：做完這次實驗後我對 Verilog 這個硬體描述語言有了初步的了解，也學會如何使用 ModelSim 以不同設計架構模擬多工計數器。希望之後如果有需要自己開發電路時能夠順利解決問題與需求。

梁凱哲：相比上週的計數器電路要注意的細節更多，整體也更加複雜，做完此次實驗後對於Verilog語言了解更多，語法及使用也更加流暢，做之後功課也會較為順利。