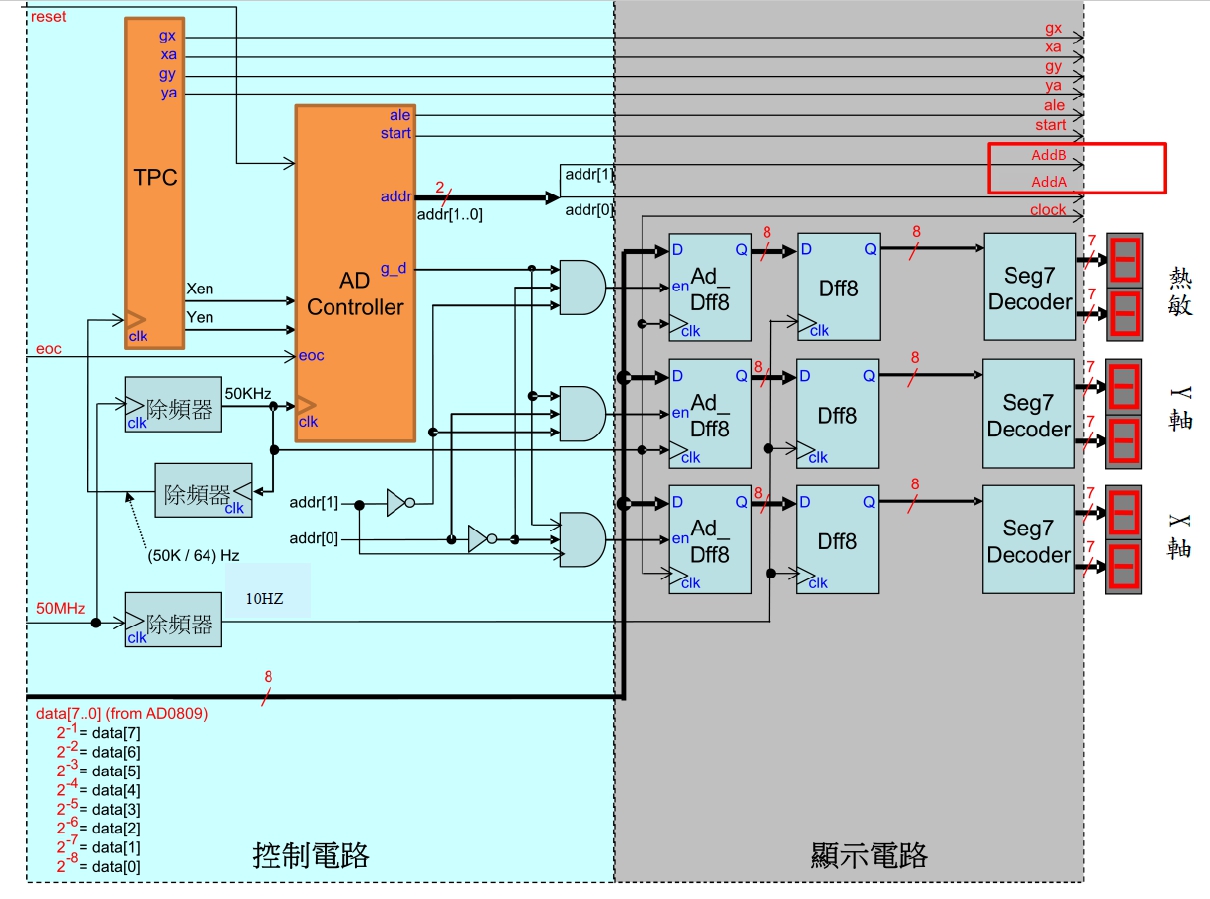
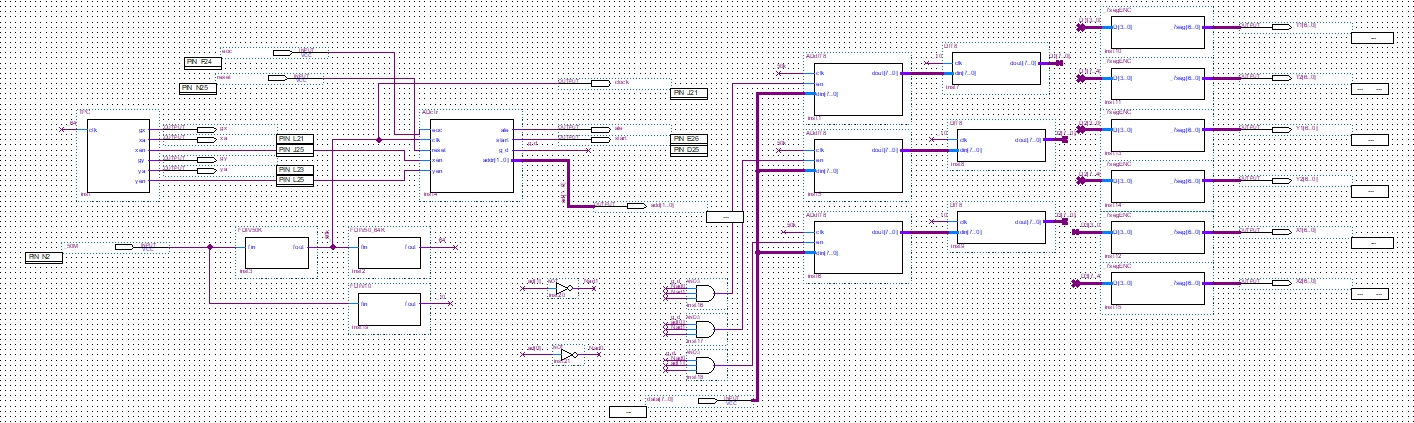
**數位實驗7： 觸控面板原理與控制電路設計**

**第十四組**

**許銘森B063011058**

**整體架構圖**

****

****

**各模組的Verilog code和波形模擬**

**除頻器 50K：**

module FDIV50K(fin,fout);

input fin;

output fout;

wire [31:0] \_DIVN,DIVN;

reg [31:0] count;

reg fout;

assign DIVN=32'd1000; // DIVN為被除數，50M/1000=50k

assign \_DIVN = DIVN >>1;

always @(posedge fin)

if(count>=DIVN-32'd1)

count=0;

else

count = count +32'd1;

always @ (negedge fin)

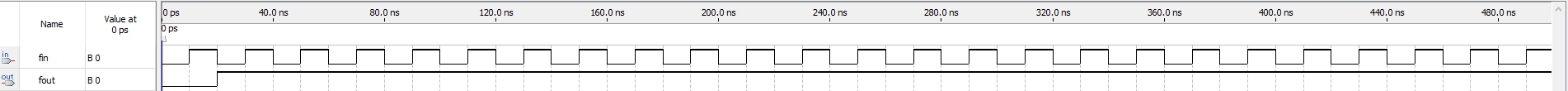
if(count>=\_DIVN) //count超過\_DIVN時輸出為0，小於則輸出為1

fout=1'd0;

else

fout=1'd1;

endmodule

****

**除頻器 10：**

module FDIV10(fin,fout);

input fin;

output fout;

wire [31:0] \_DIVN,DIVN;

reg [31:0] count;

reg fout;

assign DIVN=32'd5000000; // DIVN為被除數，50M/5000000=10

assign \_DIVN = DIVN >>1;

always @(posedge fin)

if(count>=DIVN-32'd1)

count=0;

else

count = count +32'd1;

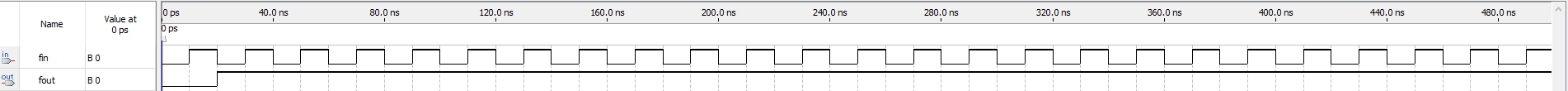
always @ (negedge fin)

if(count>=\_DIVN) //count超過\_DIVN時輸出為0，小於則輸出為1

fout=1'd0;

else

fout=1'd1;

****endmodule

module FDIV50\_64K(fin,fout);

input fin;

output fout;

wire [31:0] \_DIVN,DIVN;

reg [31:0] count;

reg fout;

assign DIVN=32'd64; // DIVN為被除數

assign \_DIVN = DIVN >>1;

always @(posedge fin)

if(count>=DIVN-32'd1)

count=0;

else

count = count +32'd1;

always @ (negedge fin)

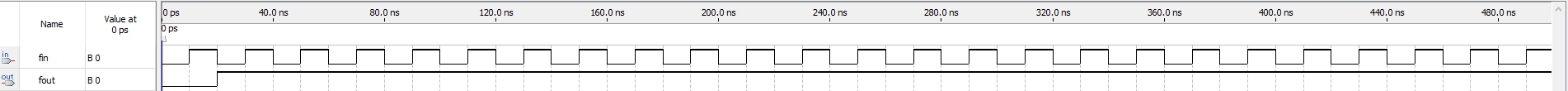
if(count>=\_DIVN) //count超過\_DIVN時輸出為0，小於則輸出為1

fout=1'd0;

else

fout=1'd1;

endmodule

****

**Ad Dff8：**

module ADdff8(clk,en,din,dout);

input clk,en;

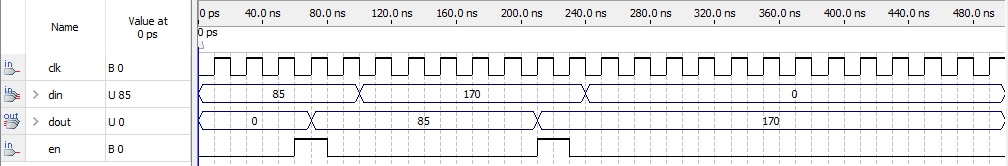
input [7:0]din;

output reg[7:0]dout;

always @(posedge clk)

dout=(en==1'd1)?din:dout; //當enable為1時dout=din，為0時dout則不變

endmodule

****

**Dff8：**

module Dff8(clk,din,dout);

input clk;

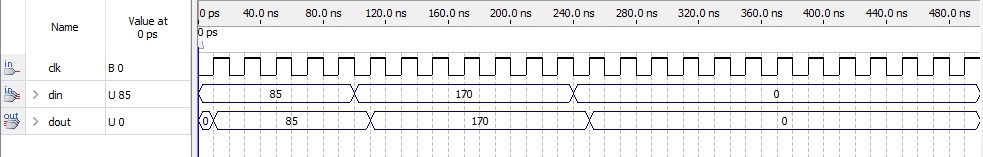
input [7:0]din;

output reg[7:0]dout;

always@(posedge clk) //正緣觸發時dout才輸出

dout=din;

endmodule

****

**Seg7\_Decoder：**

module \_7segENC(D,\_7seg);

input [3:0]D;

output [6:0]\_7seg;

reg [6:0]\_7seg;

always @(D)

begin

case(D)

4'd0:\_7seg=7'b1000000;

4'd1:\_7seg=7'b1111001;

4'd2:\_7seg=7'b0100100;

4'd3:\_7seg=7'b0110000;

4'd4:\_7seg=7'b0011001;

4'd5:\_7seg=7'b0010010;

4'd6:\_7seg=7'b0000010;

4'd7:\_7seg=7'b1111000;

4'd8:\_7seg=7'b0000000;

4'd9:\_7seg=7'b0011000;

4'd10:\_7seg=7'b0001000;

4'd11:\_7seg=7'b0000011;

4'd12:\_7seg=7'b1000110;

4'd13:\_7seg=7'b0100001;

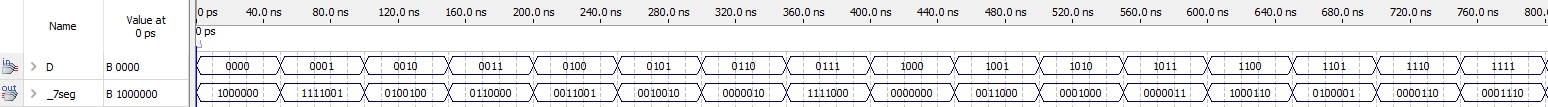
4'd14:\_7seg=7'b0000110;

4'd15:\_7seg=7'b0001110;

endcase

end

endmodule



**AD Controller：**

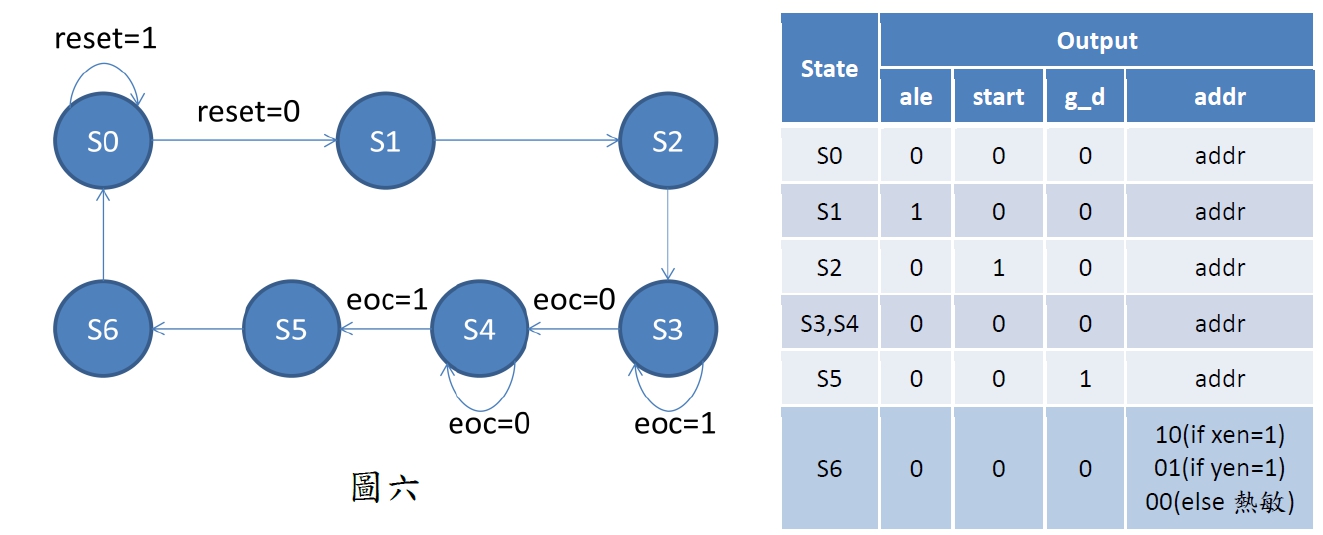
module ADctr(eoc,clk,reset,xen,yen,ale,start,g\_d,addr);

input eoc,clk,reset,xen,yen;

output ale,start,g\_d;

output reg[1:0]addr;

reg[2:0]ADctr;

reg[2:0]cs,ns;

always @(posedge clk)

cs<=ns;

assign ale=ADctr[2];

assign start=ADctr[1];

assign g\_d=ADctr[0];

always@(cs)

case(cs) //每個狀態的輸出

3'd0:ADctr<=3'b000; //ADctr的3個位元分別代表ale,start,g\_d

3'd1:ADctr<=3'b100;

3'd2:ADctr<=3'b010;

3'd3:ADctr<=3'b000;

3'd4:ADctr<=3'b000;

3'd5:ADctr<=3'b001;

3'd6:

begin

ADctr<=3'b000;//

if(xen==1'd1)

addr=2'b10;

else if(yen==1'd1)

addr=2'b01;

else

addr=2'b00;

end

endcase

always@(\*) //轉換狀態的條件

case(cs)

3'd0:ns=(reset)?3'd0:3'd1;

3'd1:ns=3'd2;//change state

3'd2:ns=3'd3;

3'd3:ns=(eoc)?3'd3:3'd4;//當eoc為1時維持原狀態，為0時進入下一個狀態

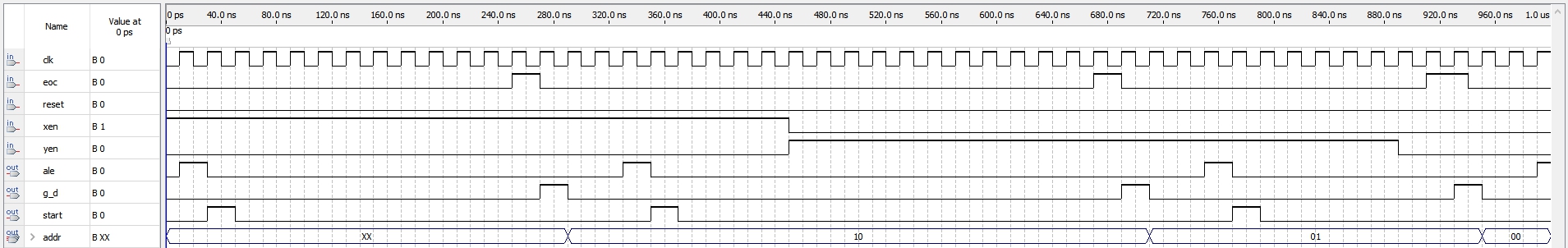
3'd4:ns=(eoc)?3'd5:3'd4;//當eoc為0時維持原狀態，為1時進入下一個狀態

3'd5:ns=3'd6;

3'd6:ns=3'd0;

endcase

endmodule

****

從上圖可見狀態輸出波形的確與上面的狀態圖一致

**TPC:**

module TPC(clk,gx,xa,xen,gy,ya,yen);

input clk;

output gx,xa,xen,gy,ya,yen;

reg[5:0]TPctr;

reg[2:0]cs,ns;

always @(posedge clk)

cs<=ns;

assign gx=TPctr[5];

assign xa=TPctr[4];

assign xen=TPctr[3];

assign gy=TPctr[2];

assign ya=TPctr[1];

assign yen=TPctr[0];

always@(cs)

case(cs)

3'd0:TPctr<=6'b000100;//TPctr的6個位元分別代表gx,xa,xen,gy,ya,yen

3'd1:TPctr<=6'b011100;

3'd2:TPctr<=6'b010100;

3'd3:TPctr<=6'b000100;

3'd4:TPctr<=6'b100000;

3'd5:TPctr<=6'b100011;

3'd6:TPctr<=6'b100010;

3'd7:TPctr<=6'b100000;

endcase

always@(\*) //轉換狀態的條件

case(cs)

3'd0:ns=3'd1;

3'd1:ns=3'd2;

3'd2:ns=3'd3;

3'd3:ns=3'd4;

3'd4:ns=3'd5;

3'd5:ns=3'd6;

3'd6:ns=3'd7;

3'd7:ns=3'd0;

endcase

endmodule

****第1個cycle將gx降為0讓面板放電

第2個cycle將xa與xen拉起

第3個cycle讓xen降為0

第4個cycle再讓xa降為0

第5個cycle則是讓gx拉為1，並且把gy降為0

第6個cycle將ya與yen拉起

第7個cycle讓yen降為0

第8個cycle讓ya降為0

**實驗心得**

這次實驗用到了ADC0809，只是這次用了3個通道，這個IC主要功能是把類比轉數位訊號。這次的實驗都也用了四線式觸控面板，其中有x軸和y軸，再加上熱敏電阻，所以會有3個類比訊號。這次實驗用到的第二顆IC就是74HC244，因爲x軸和y軸不能同時讀取，所以需要這顆IC充當開關控制ADC0809的通道。這次實驗大致順利，比較有趣的是用了一顆IC控制另一顆IC，其中模組TPC沒有提供狀態圖，我們需要整合以往控制IC的狀態機觀念，寫出這次需要達到的效果。這次實驗是最後一次數位實驗，感覺到自己的確有把前6次的知識運用在其中。