`timescale 1ns / 1ps

//////////////////////////////////////////////////////////////////////////////////

// Company:

// Engineer:

//

// Create Date: 14:16:13 11/13/2017

// Design Name:

// Module Name: vga

// Project Name:

// Target Devices:

// Tool versions:

// Description:

//

// Dependencies:

//

// Revision:

// Revision 0.01 - File Created

// Additional Comments:

//

//////////////////////////////////////////////////////////////////////////////////

module vga\_syncIndex(

clk,

hsync,vsync,

red,green,blue,

hc,vc,

blank,

choice,

b);

input clk;

output hsync;

output vsync;

output reg[2:0] red,green;

output reg [1:0] blue;

output [9:0] hc, vc;

output blank;

input [2:0]choice;

input [1:0]b;

reg read = 0;

reg wea = 1;

reg [11:0] addra = 0;

reg [8:0] addra1 = 0;

reg [23:0] in1 = 0;

reg [23:0] in2 = 0;

reg [7:0]A;

reg [7:0] tred,tgreen,tblue;

wire [23:0] out1,out2,out3;

reg [23:0] out;

integer i;

reg [7:0]temp;

reg [7:0] r2;

reg [7:0] g2;

reg [7:0] b2;

bram inpic (

.clka(clk), // input clka

.wea(read), // input [0 : 0] wea

.addra(addra), // input [11 : 0] addra

.dina(in1), // input [23 : 0] dina

.douta(out1) // output [23 : 0] douta

);

bram1 outpic (

.clka(clk), // input clka

.wea(wea), // input [0 : 0] wea

.addra(addra), // input [11 : 0] addra

.dina(in2), // input [23 : 0] dina

.douta(out2) // output [23 : 0] douta

);

wm water (

.clka(clk), // input clka

.wea(read), // input [0 : 0] wea

.addra(addra1), // input [8 : 0] addra

.dina(in1), // input [23 : 0] dina

.douta(out3) // output [23 : 0] douta

);

wire pixel\_clk;

reg pcount = 0;

wire ec = (pcount == 0);

always @ (posedge clk) pcount <= ~pcount;

assign pixel\_clk = ec;

reg hsync =0,vsync=0,hblank=0,vblank=0;

reg [9:0] hc=0;

reg [9:0] vc=0;

wire hsyncon,hsyncoff,hreset,hblankon;

assign hblankon = ec & (hc == 639);

assign hsyncon = ec & (hc == 648);

assign hsyncoff = ec & (hc == 742);

assign hreset = ec & (hc == 789);

wire blank = (vblank | (hblank & ~hreset));

wire vsyncon,vsyncoff,vreset,vblankon;

assign vblankon = hreset & (vc == 479);

assign vsyncon = hreset & (vc == 488);

assign vsyncoff = hreset & (vc == 490);

assign vreset = hreset & (vc == 523);

always @(posedge clk) begin

hc <= ec ? (hreset ? 0 : hc + 1) : hc;

hblank <= hreset ? 0 : hblankon ? 1 : hblank;

hsync <= hsyncon ? 0 : hsyncoff ? 1 : hsync;

vc <= hreset ? (vreset ? 0 : vc + 1) : vc;

vblank <= vreset ? 0 : vblankon ? 1 : vblank;

vsync <= vsyncon ? 0 : vsyncoff ? 1 : vsync;

end

always @(posedge pixel\_clk)

begin

if (hc >= 240 && hc <= 259 && vc >= 240 && vc <= 259) //Watermark

begin

r2 = {out1[23], out1[22], out1[21], out1[20], out1[19], out1[18], out1[17], out1[16]} / 2 +

{out3[23], out3[22], out3[21], out3[20], out3[19], out3[18], out3[17], out3[16]} / 2;

g2 = {out1[15], out1[14], out1[13], out1[12], out1[11], out1[10], out1[9], out1[8]} / 2 +

{out3[15], out3[14], out3[13], out3[12], out3[11], out3[10], out3[9], out3[8]}/2;

b2 = {out1[7], out1[6], out1[5], out1[4], out1[3], out1[2], out1[1], out1[0]} / 2 +

{out3[7], out3[6], out3[5], out3[4], out3[3], out3[2], out3[1], out3[0]}/2;

out = { r2,g2,b2};

if(addra1<399)

addra1 = addra1 + 1;

else

addra1 = 0;

end

else

out = out1;

if(choice == 0) //Pixel Effect

begin

if (addra%4 < 2)

begin

for(i=0; i<24; i=i+1)

begin

in2[i] = out[i];

end

end

else

begin

for(i=0; i<24; i=i+1)

begin

in2[i] = 0;

end

end

end

if(choice == 1) //Grayscale

begin

A=({out[23],out[22],out[21],out[20],out[19],out[18],out[17],out[16]}/4)

+({out[15],out[14],out[13],out[12],out[11],out[10],out[9],out[8]}/4)

+({out[7],out[6],out[5],out[4],out[3],out[2],out[1],out[0]}/4);

in2 = {A,A,A};

end

if(choice == 2) //Brightness

begin

for(i=0;i<8;i=i+1)

temp[i] = out[i];

if(b==0)

temp = temp/2+255/2;

else if(b == 1)

temp = temp/4\*3+255/4;

else if(b == 2)

temp = temp/8\*7+255/8;

else

temp = temp/16\*15+255/16;

for(i=0;i<8;i=i+1)

in2[i]=temp[i];

for(i=0;i<8;i=i+1)

temp[i] = out[i+8];

if(b==0)

temp = temp/2+255/2;

else if(b == 1)

temp = temp/4\*3+255/4;

else if(b == 2)

temp = temp/8\*7+255/8;

else

temp = temp/16\*15+255/16;

for(i=0;i<8;i=i+1)

in2[i+8]=temp[i];

for(i=0;i<8;i=i+1)

temp[i] = out[i+16];

if(b==0)

temp = temp/2+255/2;

else if(b == 1)

temp = temp/4\*3+255/4;

else if(b == 2)

temp = temp/8\*7+255/8;

else

temp = temp/16\*15+255/16;

for(i=0;i<8;i=i+1)

in2[i+16]=temp[i];

end

if(choice == 3) //sepia

begin

A=({out[23],out[22],out[21],out[20],out[19],out[18],out[17],out[16]}/4)

+({out[15],out[14],out[13],out[12],out[11],out[10],out[9],out[8]}/4)

+({out[7],out[6],out[5],out[4],out[3],out[2],out[1],out[0]}/4);

in2 = {8'b00000000,A,A};

end

if(choice == 4) //blue filter

begin

A=({out[23],out[22],out[21],out[20],out[19],out[18],out[17],out[16]}/4)

+({out[15],out[14],out[13],out[12],out[11],out[10],out[9],out[8]}/4)

+({out[7],out[6],out[5],out[4],out[3],out[2],out[1],out[0]}/4);

in2 = {A, 8'b00000000,8'b00000000};

end

if(choice == 5) //magenta filter

begin

A=({out[23],out[22],out[21],out[20],out[19],out[18],out[17],out[16]}/4)

+({out[15],out[14],out[13],out[12],out[11],out[10],out[9],out[8]}/4)

+({out[7],out[6],out[5],out[4],out[3],out[2],out[1],out[0]}/4);

in2 = {A, 8'b00000000,A};

end

if(choice == 6) //green filter

begin

A=({out[23],out[22],out[21],out[20],out[19],out[18],out[17],out[16]}/4)

+({out[15],out[14],out[13],out[12],out[11],out[10],out[9],out[8]}/4)

+({out[7],out[6],out[5],out[4],out[3],out[2],out[1],out[0]}/4);

in2 = {8'b00000000, A, 8'b00000000};

end

if(choice == 7) //cyan filter

begin

A=({out[23],out[22],out[21],out[20],out[19],out[18],out[17],out[16]}/4)

+({out[15],out[14],out[13],out[12],out[11],out[10],out[9],out[8]}/4)

+({out[7],out[6],out[5],out[4],out[3],out[2],out[1],out[0]}/4);

in2 = {A, A, 8'b00000000};

end

if(blank == 0 && hc >= 240 && hc < 290 && vc >= 240 && vc < 290)

begin

tblue = {out2[23], out2[22], out2[21], out2[20], out2[19], out2[18], out2[17], out2[16]}/64;

tgreen = {out2[15], out2[14], out2[13], out2[12], out2[11], out2[10], out2[9], out2[8]}/32;

tred = {out2[7], out2[6], out2[5], out2[4], out2[3], out2[2], out2[1], out2[0]}/32;

red = {tred[2], tred[1], tred[0]};

green = {tgreen[2], tgreen[1], tgreen[0]};

blue = {tblue[1], tblue[0]};

if(addra <2499)

addra = addra + 1;

else

addra = 0;

end

else

begin

red = 0;

green = 0;

blue = 0;

end

end

endmodule