

# VGA 显示矩形框

任务：

- 1、弄清楚 VGA 显示的原理，理解并写在实训报告中
- 2、在 VGA 显示器上显示一个矩形框。

建议步骤：

- 1、建立 VGA 显示所需其他辅助模块，包括分频等；
- 2、建立 VGA 控制模块（用纯硬件实现，也可以先实现 NIOS II，再用 C 实现）；

```
module vga_controller(
    iclk,irst_n,
    ovga_r,ovga_g,
    ovga_b,
    ovga_sync,
    ovga_blank,
    ovga_clock,
    ovga_h_sync,
    ovga_v_sync
);
input  iclk,irst_n; //iclk 应为 25.175MHZ
output[9:0] ovga_r;
output[9:0] ovga_g;
output[9:0] ovga_b;
output ovga_sync; //同步信号输出
output ovga_blank; //消隐信号输出
output ovga_clock; //时钟输出
output reg ovga_h_sync; //水平同步信号输出
output reg ovga_v_sync; //垂直同步信号输出
reg [9:0]h_cnt; //行扫描计数器
reg [9:0]v_cnt; //列扫描计数器

//水平参数（单位：像素）
parameter h_sync_cyc=96,
    h_sync_back=45+3,
    h_sync_act=640,
    h_sync_front=13+3,
    h_sync_total=800;
//垂直参数（单位：线）
parameter v_sync_cyc=2,
    v_sync_back=30+2,
    v_sync_act=480,
```

```

    v_sync_front=9+2,
    v_sync_total=525;
//起始偏移
parameter  x_start=h_sync_cyc+h_sync_back,
    y_start=v_sync_cyc+v_sync_back;
assign ovga_blank=ovga_h_sync & ovga_v_sync;
assign ovga_sync=1'b0;
assign ovga_clock=iclk;

assign valid = (h_cnt > x_start) && (h_cnt < x_start+h_sync_act)
               && (v_cnt > y_start) && (v_cnt < y_start+v_sync_act);

wire[9:0] xpos,ypos;    //有效显示区坐标

assign xpos = h_cnt-10'd144;
assign ypos = v_cnt-10'd34;
//-----
    //显示一个矩形框
wire a_dis,b_dis,c_dis,d_dis;    //矩形框显示区域定位
reg[9:0] aa_dis;
reg[9:0] bb_dis;
reg[9:0] cc_dis;
reg[9:0] dd_dis;
reg[9:0] ee_rdy;
always @(posedge iclk or negedge first_n)
begin
    if(!first_n)
        begin
            aa_dis<=10'd0;
            bb_dis<=10'd0;
            cc_dis<=10'd0;
            dd_dis<=10'd0;
        end
    else
        begin
            if(a_dis)
                aa_dis<=10'h3ff;
            else
                aa_dis<=0;
            if(b_dis)
                bb_dis<=10'h3ff;
            else
                bb_dis<=0;
        end
    end
end

```

```

    if(c_dis)
        cc_dis<=10'h3ff;
    else
        cc_dis<=0;
    if(d_dis)
        dd_dis<=10'h3ff;
    else
        dd_dis<=0;
    if(e_rdy)
        ee_rdy<=10'h3ff;
    else
        ee_rdy<=0;
    end
end

assign a_dis = ( (xpos>=200) && (xpos<=220) )
               && ( (ypos>=100) && (ypos<=380) );

assign b_dis = ( (xpos>=440) && (xpos<=460) )
               && ( (ypos>=100) && (ypos<=380) );

assign c_dis = ( (xpos>=220) && (xpos<=440) )
               && ( (ypos>100) && (ypos<=120) );

assign d_dis = ( (xpos>=220) && (xpos<=440) )
               && ( (ypos>=360) && (ypos<=380) );

    //显示一个小矩形
wire e_rdy;    //矩形的显示有效矩形区域

assign e_rdy = ( (xpos>=310) && (xpos<=330) )
               && ( (ypos>=230) && (ypos<=250) );

//-----
    //r,g,b 控制液晶屏颜色显示，背景显示蓝色，矩形框显示红蓝色
assign ovga_r = valid && e_rdy ? ee_rdy : 1'b0;
assign ovga_g = valid && (a_dis | b_dis | c_dis | d_dis) ? 10'h3ff: 1'b0;
assign ovga_b = valid && (a_dis | b_dis | c_dis | d_dis) ? 10'h3ff: 1'b0;

//生成水平同步信号 h_sync
always @(posedge iclk or negedge first_n)
begin

```

```

if(!first_n)
begin
h_cnt<=0;
ovga_h_sync<=0;
end
else
begin
if(h_cnt<h_sync_total)//计数器计数
h_cnt<=h_cnt+1;
else
h_cnt<=0;
if(h_cnt<h_sync_cyc) //生成 h_sync
ovga_h_sync<=0;
else
ovga_h_sync<=1;
end
end
//参照水平同步生成垂直同步信号 v_sync
always @(posedge iclk or negedge rst_n)
begin
if(!first_n)
begin
v_cnt<=0;
ovga_v_sync<=0;
end
else
begin
if(h_cnt==0)
begin
if(v_cnt<v_sync_total)
v_cnt<=v_cnt+1;
else
v_cnt<=0;
if(v_cnt<v_sync_cyc)
ovga_v_sync<=0;
else
ovga_v_sync<=1;
end
end
end
endmodule

```

顶层电路原理图

