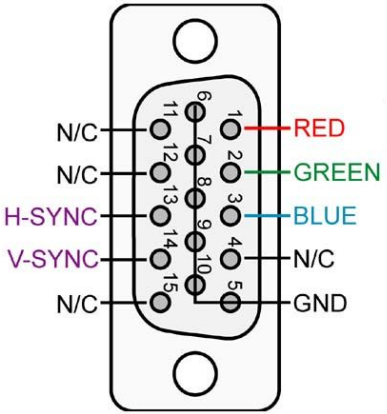
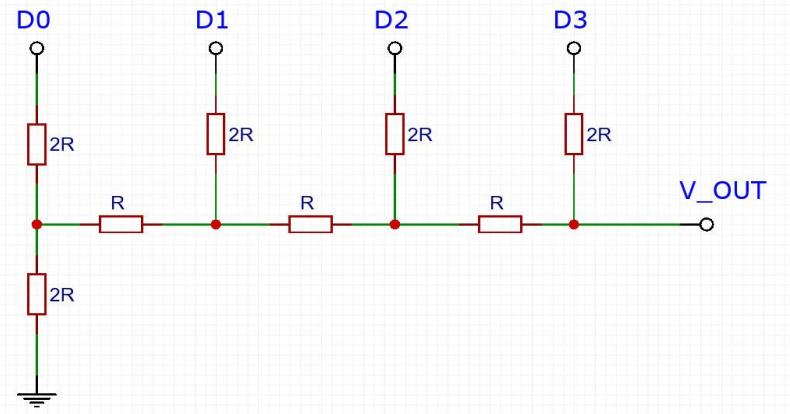
Интерфейс VGA

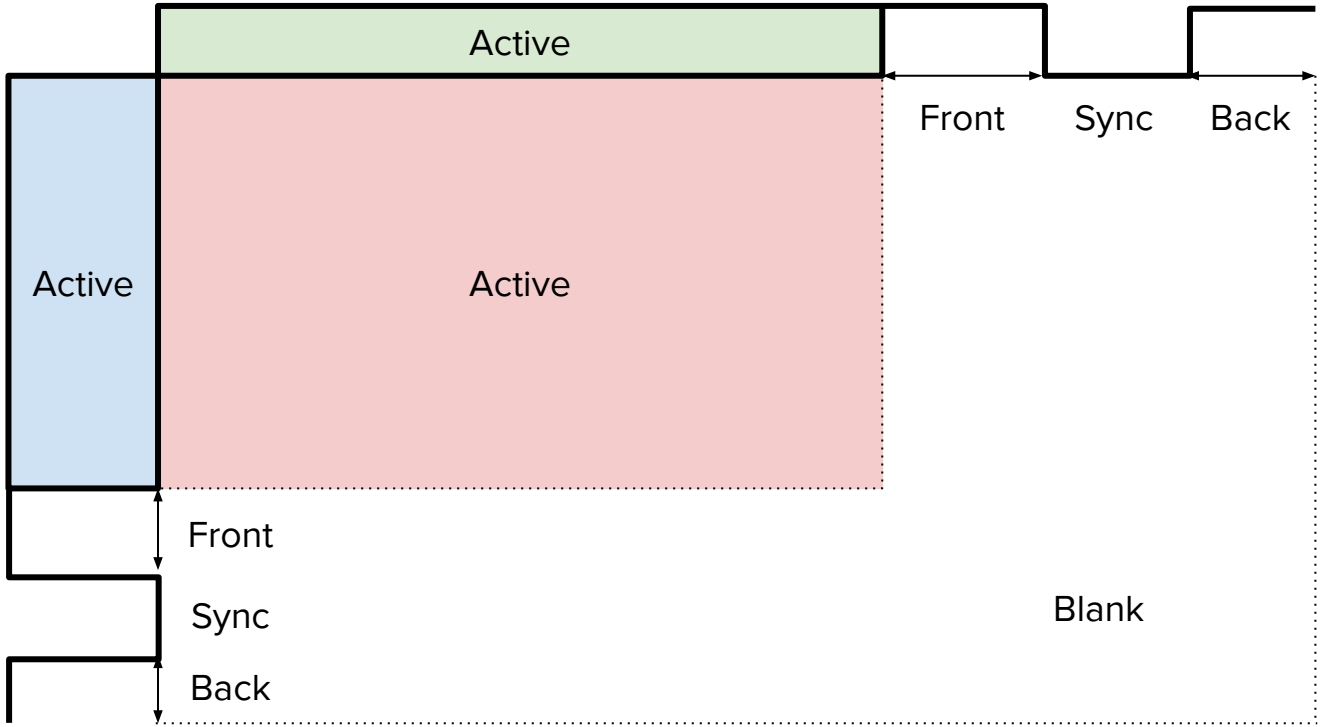
VGA

VGA (Video Graphics Array) — аналоговый формат видеосигналов для вывода изображения на мониторы





VGA

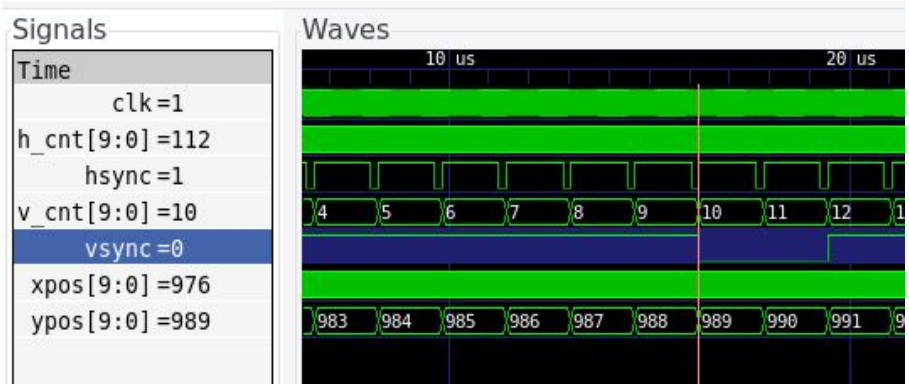


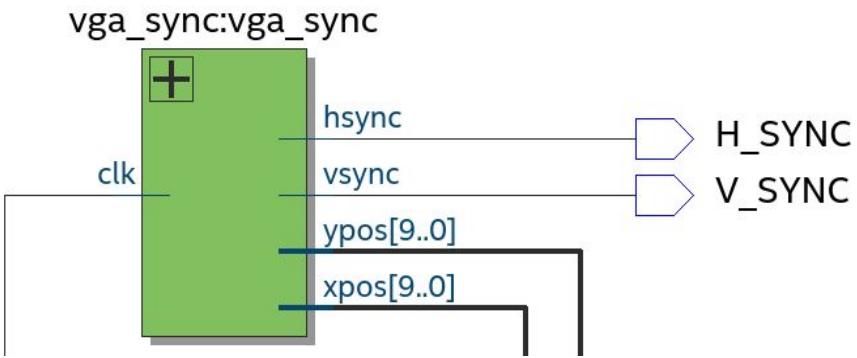
640x480@60Hz — 25.175 MHz		
Horizontal	Active	640
	Front	16
	Sync	96
	Back	48
Vertical	Active	480
	Front	10
	Sync	2
	Back	33

vga_sync.v

```
module vga_sync(
    input clk,

    output reg hsync = 1,
    output reg vsync = 1,
    output [9:0]xpos,
    output [9:0]ypos
);

parameter H_front_t = 16;
parameter H_sync_t = 96;
parameter H_back_t = 48;
parameter H_active_t = 640;
parameter H_blank_t = H_front_t + H_sync_t + H_back_t;
parameter H_total_t = H_blank_t + H_active_t;

parameter V_front_t = 10;
parameter V_sync_t = 2;
parameter V_back_t = 33;
parameter V_active_t = 480;
parameter V_blank_t = V_front_t + V_sync_t + V_back_t;
parameter V_total_t = V_blank_t + V_active_t;

reg [9:0]h_cnt = 10'h3FF;
reg [9:0]v_cnt = 10'h3FF;

assign xpos = h_cnt - H_blank_t;
assign ypos = v_cnt - V_blank_t;

always @(posedge clk) begin
    if (h_cnt == H_total_t - 1)
        h_cnt <= 0;
    else
        h_cnt <= h_cnt + 1;

    if (h_cnt == H_front_t - 1)
        hsync <= 0;
    else if (h_cnt == H_front_t + H_sync_t - 1) begin
        hsync <= 1;

        if (v_cnt == V_total_t - 1)
            v_cnt <= 0;
        else
            v_cnt <= v_cnt + 1;

        if (v_cnt == V_front_t - 1)
            vsync <= 0;
        else if (v_cnt == V_front_t + V_sync_t - 1)
            vsync <= 1;
    end
end
```

vga_sync.v

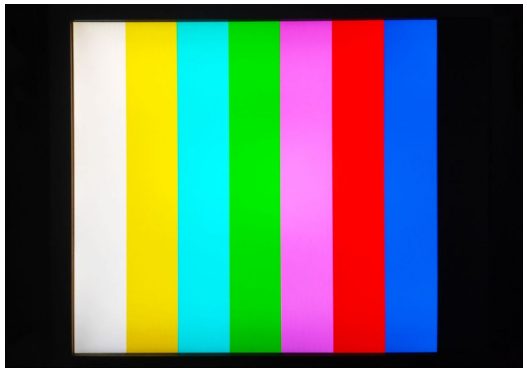

Signals
Waves
Time
clk =1
h_cnt[9:0] =112
hsync =1
v_cnt[9:0] =10
vsync =0
xpos[9:0] =976
ypos[9:0] =989
10 us
20 us
4
5
6
7
8
9
10
11
12
983
984
985
986
987
988
989
990
991

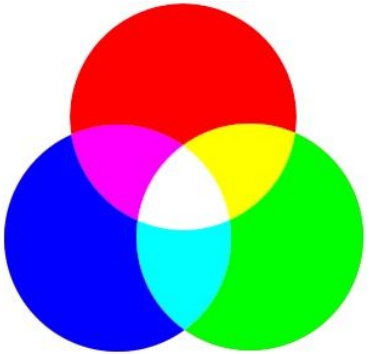

vga_sync:vga_sync
clk
hsync
vsync
ypos[9..0]
xpos[9..0]
H_SYNC
V_SYNC

top.v — генерация цветных полос

```
module top (
    input CLK,
    output H_SYNC, V_SYNC,
    output [4:0]V_R,
    output [5:0]V_G,
    output [4:0]V_B
);

reg [2:0]pix = 3'b111;
assign V_R = {5{pix[1]}};
assign V_G = {6{pix[2]}};
assign V_B = {5{pix[0]}};

wire vga_clk;
pll pll_inst(.inclk0(CLK), .c0(vga_clk));

wire [9:0]x;
wire [9:0]y;

vga_sync vga_sync(vga_clk, H_SYNC, V_SYNC, x, y);
```

```
always @(posedge vga_clk) begin
    if ((x < 640) && (y < 480)) begin
        if (x % 80 == 0)
            pix <= pix - 3'b1;
    end
    else
        pix <= 0;
end
endmodule
```

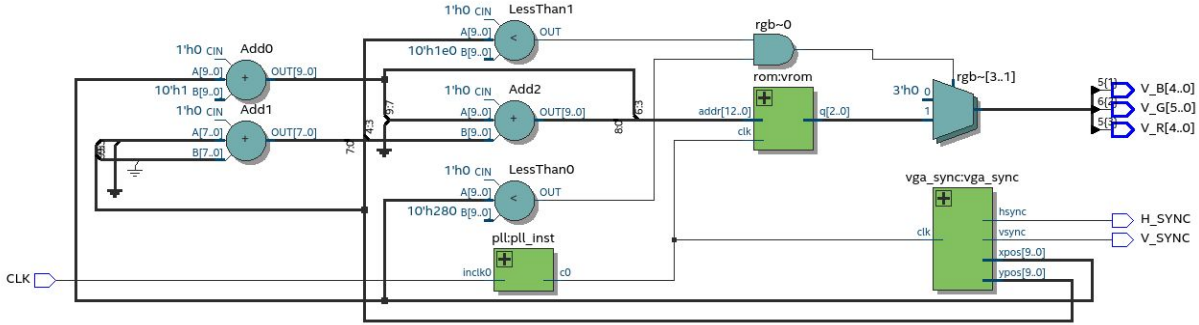




top.v — отображение текста

```
module top (
    input CLK,
    output H_SYNC, V_SYNC,
    output [4:0]V_R,
    output [5:0]V_G,
    output [4:0]V_B
);

wire [9:0]x;
wire [9:0]y;
reg [2:0]rgb;
assign V_R = {5{rgb[0]}};
assign V_G = {6{rgb[1]}};
assign V_B = {5{rgb[2]}};

pll pll_inst(.inclk0(CLK), .c0(vga_clk
```


```
reg [12:0]vrom_addr;
wire [2:0]vrom_q;
wire [9:0]x_fwd = x + 1;

rom #(13, 3, "rom.txt") vrom(vrom_addr, vga_clk, vrom_q);
vga_sync vga_sync(vga_clk, H_SYNC, V_SYNC, x, y);

always @(*) begin
    vrom_addr = x_fwd[9:3] + y[9:3] * 80;
    rgb = ((x < 640) && (y < 480)) ? vrom_q : 3'b0;
end
endmodule
```



GitHub

github.com/viktor-prutyanov/drec-fpga-intro