中国科学技术大学计算机学院

《数字电路实验》报告

实验题目：存储器

学生姓名：郭耸霄

学生学号：PB20111712

完成日期：2021 年 11 月 29 日

计算机实验教学中心制

2020 年 09 月

实 验 报 告

11 系 20 级 3 班　　郭耸霄 PB20111712　　2021 年 11 月 29 日

1 实验题目

存储器。

2 实验目的

控制画笔在画布 (显示屏) 上绘画:

1. 画布分辨率：200x150，画笔位置 (x, y)，x = 0 199, y = 0 149, 复位时 (100, 75)。

2. 画笔移动方向 (dir)：上/下/左/右，按钮控制。

画笔颜色 (rgb)：12 位 (红 r 绿 g 蓝 b 各 4 位)，开关设置。

绘画开关 (draw)：处于绘画状态时，移动画笔同时绘制颜色，否则仅移动画笔。

prgb, hs, vs: 显示器接口信号。

3 实验环境

1.Surface Pro 7 Model 1866 i7 实验设备。

2.Nexys 4 DDR 实验平台。

3.Vivado 2019.1 EDA 工具。

4 实现 VDT，显示白色背景

4.1 逻辑设计

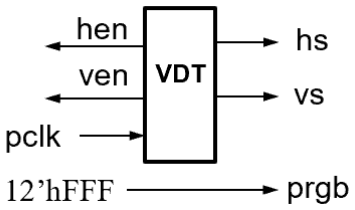

图 1: VDT 数据通路

4.2 设计文件

4.2.1 分频器

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 28.11.2021 08:27:01
// Design Name:
// Module Name: divider
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////
```

```
module DIR(input clk,
           output reg pclk);
    always @(posedge clk) begin
        pclk = ~pclk;
    end

endmodule
```

4.2.2 视频显示定时

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 28.11.2021 10:30:59
// Design Name:
// Module Name: VDT
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////


module VDT(input pclk,
           rstn,
           output hen,
           ven,
           reg hs,
           reg vs);
    parameter HSW = 120;
    parameter HBP = 64;
    parameter HEN = 800;
    parameter HFP = 56;
    parameter VSW = 6;
    parameter VBP = 23;
    parameter VEN = 600 ;
    parameter VFP = 37;
    reg[10:0] hcnt;
    reg[9:0]vcnt;
    //change hcnt
    always @(posedge pclk) begin
        if (!rstn)
            hcnt <= HEN+HFP;
        else if (hcnt == HSW+HBP+HEN+HFP-1)
            hcnt <= 0;
        else
            hcnt <= hcnt+1;
    end
    //change vcnt
    always @(posedge pclk) begin
        if (!rstn)
            vcnt <= VEN+VFP;
        else if (vcnt == VSW+VBP+VEN+VFP-1)
            vcnt <= 0;
        else if (hcnt == HEN+HFP-1)
            vcnt <= vcnt+1;
            end
            //change hs
            always @(posedge pclk) begin
                if (!rstn||hcnt == HEN+HFP+HSW-1)
                    hs <= 0;
                else if (hcnt == HEN+HFP-1)
                    hs <= 1;
                    end
                    //change vs
                    always @(posedge pclk) begin
                        if (!rstn||vcnt == VEN+VFP+VSW-1)
                            vs <= 0;
                        else if (vcnt == VEN+VFP-1)
                            vs <= 1;
                            end
```

```
assign hen = hcnt<HEN;
assign ven = vcnt<VEN;
endmodule
```

4.2.3 顶层模块

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 28.11.2021 16:45:16
// Design Name:
// Module Name: test
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////


module test(input clk,
            rstn,
            output reg[11:0] prgb,
            output hs,
            output vs);
    wire hen,ven;
    DIR dir(.clk(clk),.pclk(pclk));
    VDT vdt(.pclk(pclk),.rstn(rstn),.hen(hen),.ven(ven),.hs(hs),.vs(vs));
    always @(*) begin
        if (hen&&ven)
            prgb <= 12'hfff;
        else
            prgb <= 0;
    end
endmodule
```

4.3 RTL 分析

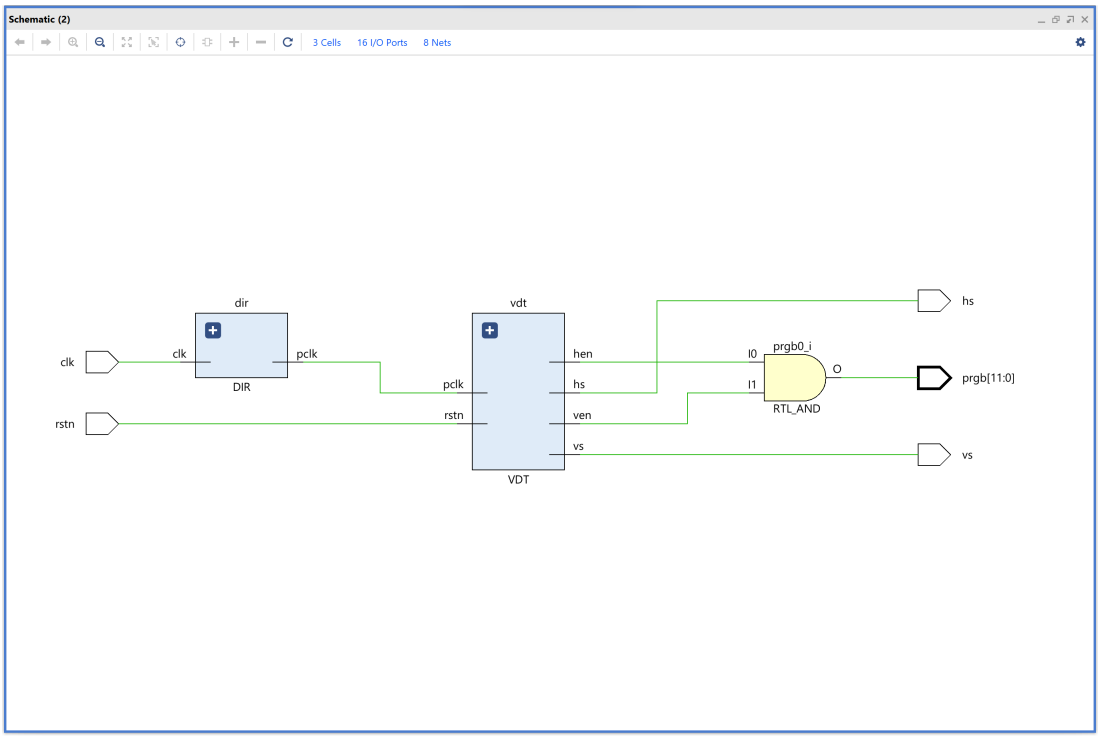

图 2: RTL 分析电路图

4.4 约束文件

```
## This file is a general .xdc for the Nexys4 DDR Rev. C
## To use it in a project:
## - uncomment the lines corresponding to used pins
## - rename the used ports (in each line, after get_ports) according to the top level signal names in the project

## Clock signal
set_property -dict { PACKAGE_PIN E3    IOSTANDARD LVCMOS33 } [get_ports { clk }]; #IO_L12P_T1_MRCC_35 Sch=clk100mhz
create_clock -add -name sys_clk_pin -period 10.00 -waveform {0 5} [get_ports {clk}];


##Switches

#set_property -dict { PACKAGE_PIN J15   IOSTANDARD LVCMOS33 } [get_ports { SW[0] }]; #IO_L24N_T3_RS0_15 Sch=sw[0]
#set_property -dict { PACKAGE_PIN L16   IOSTANDARD LVCMOS33 } [get_ports { SW[1] }]; #IO_L3N_T0_DQS_EMCCLK_14 Sch=sw[1]
#set_property -dict { PACKAGE_PIN M13   IOSTANDARD LVCMOS33 } [get_ports { SW[2] }]; #IO_L6N_T0_D08_VREF_14 Sch=sw[2]
#set_property -dict { PACKAGE_PIN R15   IOSTANDARD LVCMOS33 } [get_ports { SW[3] }]; #IO_L13N_T2_MRCC_14 Sch=sw[3]
#set_property -dict { PACKAGE_PIN R17   IOSTANDARD LVCMOS33 } [get_ports { SW[4] }]; #IO_L12N_T1_MRCC_14 Sch=sw[4]
#set_property -dict { PACKAGE_PIN T18   IOSTANDARD LVCMOS33 } [get_ports { SW[5] }]; #IO_L7N_T1_D10_14 Sch=sw[5]
#set_property -dict { PACKAGE_PIN U18   IOSTANDARD LVCMOS33 } [get_ports { SW[6] }]; #IO_L17N_T2_A13_D29_14 Sch=sw[6]
#set_property -dict { PACKAGE_PIN R13   IOSTANDARD LVCMOS33 } [get_ports { SW[7] }]; #IO_L5N_T0_D07_14 Sch=sw[7]
#set_property -dict { PACKAGE_PIN T8    IOSTANDARD LVCMOS18 } [get_ports { SW[8] }]; #IO_L24N_T3_34 Sch=sw[8]
#set_property -dict { PACKAGE_PIN U8    IOSTANDARD LVCMOS18 } [get_ports { SW[9] }]; #IO_25_34 Sch=sw[9]
#set_property -dict { PACKAGE_PIN R16   IOSTANDARD LVCMOS33 } [get_ports { SW[10] }]; #IO_L15P_T2_DQS_RDWR_B_14 Sch=sw[10]
#set_property -dict { PACKAGE_PIN T13   IOSTANDARD LVCMOS33 } [get_ports { SW[11] }]; #IO_L23P_T3_A03_D19_14 Sch=sw[11]
#set_property -dict { PACKAGE_PIN H6    IOSTANDARD LVCMOS33 } [get_ports { SW[12] }]; #IO_L24P_T3_35 Sch=sw[12]
#set_property -dict { PACKAGE_PIN U12   IOSTANDARD LVCMOS33 } [get_ports { SW[13] }]; #IO_L20P_T3_A08_D24_14 Sch=sw[13]
#set_property -dict { PACKAGE_PIN U11   IOSTANDARD LVCMOS33 } [get_ports { SW[14] }]; #IO_L19N_T3_A09_D25_VREF_14 Sch=sw[14]
#set_property -dict { PACKAGE_PIN V10   IOSTANDARD LVCMOS33 } [get_ports { SW[15] }]; #IO_L21P_T3_DQS_14 Sch=sw[15]


## LEDs

#set_property -dict { PACKAGE_PIN H17   IOSTANDARD LVCMOS33 } [get_ports { LED[0] }]; #IO_L18P_T2_A24_15 Sch=led[0]
#set_property -dict { PACKAGE_PIN K15   IOSTANDARD LVCMOS33 } [get_ports { LED[1] }]; #IO_L24P_T3_RS1_15 Sch=led[1]
#set_property -dict { PACKAGE_PIN J13   IOSTANDARD LVCMOS33 } [get_ports { LED[2] }]; #IO_L17N_T2_A25_15 Sch=led[2]
#set_property -dict { PACKAGE_PIN N14   IOSTANDARD LVCMOS33 } [get_ports { LED[3] }]; #IO_L8P_T1_D11_14 Sch=led[3]
#set_property -dict { PACKAGE_PIN R18   IOSTANDARD LVCMOS33 } [get_ports { LED[4] }]; #IO_L7P_T1_D09_14 Sch=led[4]
#set_property -dict { PACKAGE_PIN V17   IOSTANDARD LVCMOS33 } [get_ports { LED[5] }]; #IO_L18N_T2_A11_D27_14 Sch=led[5]
#set_property -dict { PACKAGE_PIN U17   IOSTANDARD LVCMOS33 } [get_ports { LED[6] }]; #IO_L17P_T2_A14_D30_14 Sch=led[6]
```

```
#set_property -dict { PACKAGE_PIN U16   IOSTANDARD LVCMOS33 } [get_ports { LED[7] }]; #IO_L18P_T2_A12_D28_14 Sch=led[7]
#set_property -dict { PACKAGE_PIN V16   IOSTANDARD LVCMOS33 } [get_ports { LED[8] }]; #IO_L16N_T2_A15_D31_14 Sch=led[8]
#set_property -dict { PACKAGE_PIN T15   IOSTANDARD LVCMOS33 } [get_ports { LED[9] }]; #IO_L14N_T2_SRCC_14 Sch=led[9]
#set_property -dict { PACKAGE_PIN U14   IOSTANDARD LVCMOS33 } [get_ports { LED[10] }]; #IO_L22P_T3_A05_D21_14 Sch=led[10]
#set_property -dict { PACKAGE_PIN T16   IOSTANDARD LVCMOS33 } [get_ports { LED[11] }]; #IO_L15N_T2_DQS_DOUT_CSO_B_14 Sch=led[11]
#set_property -dict { PACKAGE_PIN V15   IOSTANDARD LVCMOS33 } [get_ports { LED[12] }]; #IO_L16P_T2_CSI_B_14 Sch=led[12]
#set_property -dict { PACKAGE_PIN V14   IOSTANDARD LVCMOS33 } [get_ports { LED[13] }]; #IO_L22N_T3_A04_D20_14 Sch=led[13]
#set_property -dict { PACKAGE_PIN V12   IOSTANDARD LVCMOS33 } [get_ports { LED[14] }]; #IO_L20N_T3_A07_D23_14 Sch=led[14]
#set_property -dict { PACKAGE_PIN V11   IOSTANDARD LVCMOS33 } [get_ports { LED[15] }]; #IO_L21N_T3_DQS_A06_D22_14 Sch=led[15]

#set_property -dict { PACKAGE_PIN R12   IOSTANDARD LVCMOS33 } [get_ports { LED16_B }]; #IO_L5P_T0_D06_14 Sch=led16_b
#set_property -dict { PACKAGE_PIN M16   IOSTANDARD LVCMOS33 } [get_ports { LED16_G }]; #IO_L10P_T1_D14_14 Sch=led16_g
#set_property -dict { PACKAGE_PIN N15   IOSTANDARD LVCMOS33 } [get_ports { LED16_R }]; #IO_L11P_T1_SRCC_14 Sch=led16_r
#set_property -dict { PACKAGE_PIN G14   IOSTANDARD LVCMOS33 } [get_ports { LED17_B }]; #IO_L15N_T2_DQS_ADV_B_15 Sch=led17_b
#set_property -dict { PACKAGE_PIN R11   IOSTANDARD LVCMOS33 } [get_ports { LED17_G }]; #IO_0_14 Sch=led17_g
#set_property -dict { PACKAGE_PIN N16   IOSTANDARD LVCMOS33 } [get_ports { LED17_R }]; #IO_L11N_T1_SRCC_14 Sch=led17_r


##7 segment display

#set_property -dict { PACKAGE_PIN T10   IOSTANDARD LVCMOS33 } [get_ports { CA }]; #IO_L24N_T3_A00_D16_14 Sch=ca
#set_property -dict { PACKAGE_PIN R10   IOSTANDARD LVCMOS33 } [get_ports { CB }]; #IO_25_14 Sch=cb
#set_property -dict { PACKAGE_PIN K16   IOSTANDARD LVCMOS33 } [get_ports { CC }]; #IO_25_15 Sch=cc
#set_property -dict { PACKAGE_PIN K13   IOSTANDARD LVCMOS33 } [get_ports { CD }]; #IO_L17P_T2_A26_15 Sch=cd
#set_property -dict { PACKAGE_PIN P15   IOSTANDARD LVCMOS33 } [get_ports { CE }]; #IO_L13P_T2_MRCC_14 Sch=ce
#set_property -dict { PACKAGE_PIN T11   IOSTANDARD LVCMOS33 } [get_ports { CF }]; #IO_L19P_T3_A10_D26_14 Sch=cf
#set_property -dict { PACKAGE_PIN L18   IOSTANDARD LVCMOS33 } [get_ports { CG }]; #IO_L4P_T0_D04_14 Sch=cg

#set_property -dict { PACKAGE_PIN H15   IOSTANDARD LVCMOS33 } [get_ports { DP }]; #IO_L19N_T3_A21_VREF_15 Sch=dp

#set_property -dict { PACKAGE_PIN J17   IOSTANDARD LVCMOS33 } [get_ports { AN[0] }]; #IO_L23P_T3_FOE_B_15 Sch=an[0]
#set_property -dict { PACKAGE_PIN J18   IOSTANDARD LVCMOS33 } [get_ports { AN[1] }]; #IO_L23N_T3_FWE_B_15 Sch=an[1]
#set_property -dict { PACKAGE_PIN T9    IOSTANDARD LVCMOS33 } [get_ports { AN[2] }]; #IO_L24P_T3_A01_D17_14 Sch=an[2]
#set_property -dict { PACKAGE_PIN J14   IOSTANDARD LVCMOS33 } [get_ports { AN[3] }]; #IO_L19P_T3_A22_15 Sch=an[3]
#set_property -dict { PACKAGE_PIN P14   IOSTANDARD LVCMOS33 } [get_ports { AN[4] }]; #IO_L8N_T1_D12_14 Sch=an[4]
#set_property -dict { PACKAGE_PIN T14   IOSTANDARD LVCMOS33 } [get_ports { AN[5] }]; #IO_L14P_T2_SRCC_14 Sch=an[5]
#set_property -dict { PACKAGE_PIN K2    IOSTANDARD LVCMOS33 } [get_ports { AN[6] }]; #IO_L23P_T3_35 Sch=an[6]
#set_property -dict { PACKAGE_PIN U13   IOSTANDARD LVCMOS33 } [get_ports { AN[7] }]; #IO_L23N_T3_A02_D18_14 Sch=an[7]


##Buttons

set_property -dict { PACKAGE_PIN C12   IOSTANDARD LVCMOS33 } [get_ports { rstn }]; #IO_L3P_T0_DQS_AD1P_15 Sch=cpu_resetn

#set_property -dict { PACKAGE_PIN N17   IOSTANDARD LVCMOS33 } [get_ports { BTNC }]; #IO_L9P_T1_DQS_14 Sch=btnc
#set_property -dict { PACKAGE_PIN M18   IOSTANDARD LVCMOS33 } [get_ports { BTNU }]; #IO_L4N_T0_D05_14 Sch=btnu
#set_property -dict { PACKAGE_PIN P17   IOSTANDARD LVCMOS33 } [get_ports { BTNL }]; #IO_L12P_T1_MRCC_14 Sch=btnl
#set_property -dict { PACKAGE_PIN M17   IOSTANDARD LVCMOS33 } [get_ports { BTNR }]; #IO_L10N_T1_D15_14 Sch=btnr
#set_property -dict { PACKAGE_PIN P18   IOSTANDARD LVCMOS33 } [get_ports { BTND }]; #IO_L9N_T1_DQS_D13_14 Sch=btnd


##Pmod Headers


##Pmod Header JA

#set_property -dict { PACKAGE_PIN C17   IOSTANDARD LVCMOS33 } [get_ports { JA[1] }]; #IO_L20N_T3_A19_15 Sch=ja[1]
#set_property -dict { PACKAGE_PIN D18   IOSTANDARD LVCMOS33 } [get_ports { JA[2] }]; #IO_L21N_T3_DQS_A18_15 Sch=ja[2]
#set_property -dict { PACKAGE_PIN E18   IOSTANDARD LVCMOS33 } [get_ports { JA[3] }]; #IO_L21P_T3_DQS_15 Sch=ja[3]
#set_property -dict { PACKAGE_PIN G17   IOSTANDARD LVCMOS33 } [get_ports { JA[4] }]; #IO_L18N_T2_A23_15 Sch=ja[4]
#set_property -dict { PACKAGE_PIN D17   IOSTANDARD LVCMOS33 } [get_ports { JA[7] }]; #IO_L16N_T2_A27_15 Sch=ja[7]
#set_property -dict { PACKAGE_PIN E17   IOSTANDARD LVCMOS33 } [get_ports { JA[8] }]; #IO_L16P_T2_A28_15 Sch=ja[8]
#set_property -dict { PACKAGE_PIN F18   IOSTANDARD LVCMOS33 } [get_ports { JA[9] }]; #IO_L22N_T3_A16_15 Sch=ja[9]
#set_property -dict { PACKAGE_PIN G18   IOSTANDARD LVCMOS33 } [get_ports { JA[10] }]; #IO_L22P_T3_A17_15 Sch=ja[10]


##Pmod Header JB

#set_property -dict { PACKAGE_PIN D14   IOSTANDARD LVCMOS33 } [get_ports { JB[1] }]; #IO_L1P_T0_AD0P_15 Sch=jb[1]
#set_property -dict { PACKAGE_PIN F16   IOSTANDARD LVCMOS33 } [get_ports { JB[2] }]; #IO_L14N_T2_SRCC_15 Sch=jb[2]
#set_property -dict { PACKAGE_PIN G16   IOSTANDARD LVCMOS33 } [get_ports { JB[3] }]; #IO_L13N_T2_MRCC_15 Sch=jb[3]
#set_property -dict { PACKAGE_PIN H14   IOSTANDARD LVCMOS33 } [get_ports { JB[4] }]; #IO_L15P_T2_DQS_15 Sch=jb[4]
#set_property -dict { PACKAGE_PIN E16   IOSTANDARD LVCMOS33 } [get_ports { JB[7] }]; #IO_L11N_T1_SRCC_15 Sch=jb[7]
#set_property -dict { PACKAGE_PIN F13   IOSTANDARD LVCMOS33 } [get_ports { JB[8] }]; #IO_L5P_T0_AD9P_15 Sch=jb[8]
#set_property -dict { PACKAGE_PIN G13   IOSTANDARD LVCMOS33 } [get_ports { JB[9] }]; #IO_0_15 Sch=jb[9]
#set_property -dict { PACKAGE_PIN H16   IOSTANDARD LVCMOS33 } [get_ports { JB[10] }]; #IO_L13P_T2_MRCC_15 Sch=jb[10]


##Pmod Header JC

#set_property -dict { PACKAGE_PIN K1    IOSTANDARD LVCMOS33 } [get_ports { JC[1] }]; #IO_L23N_T3_35 Sch=jc[1]
#set_property -dict { PACKAGE_PIN F6    IOSTANDARD LVCMOS33 } [get_ports { JC[2] }]; #IO_L19N_T3_VREF_35 Sch=jc[2]
```

```
#set_property -dict { PACKAGE_PIN J2    IOSTANDARD LVCMOS33 } [get_ports { JC[3] }]; #IO_L22N_T3_35 Sch=jc[3]
#set_property -dict { PACKAGE_PIN G6    IOSTANDARD LVCMOS33 } [get_ports { JC[4] }]; #IO_L19P_T3_35 Sch=jc[4]
#set_property -dict { PACKAGE_PIN E7    IOSTANDARD LVCMOS33 } [get_ports { JC[7] }]; #IO_L6P_T0_35 Sch=jc[7]
#set_property -dict { PACKAGE_PIN J3    IOSTANDARD LVCMOS33 } [get_ports { JC[8] }]; #IO_L22P_T3_35 Sch=jc[8]
#set_property -dict { PACKAGE_PIN J4    IOSTANDARD LVCMOS33 } [get_ports { JC[9] }]; #IO_L21P_T3_DQS_35 Sch=jc[9]
#set_property -dict { PACKAGE_PIN E6    IOSTANDARD LVCMOS33 } [get_ports { JC[10] }]; #IO_L5P_T0_AD13P_35 Sch=jc[10]


##Pmod Header JD

#set_property -dict { PACKAGE_PIN H4    IOSTANDARD LVCMOS33 } [get_ports { JD[1] }]; #IO_L21N_T3_DQS_35 Sch=jd[1]
#set_property -dict { PACKAGE_PIN H1    IOSTANDARD LVCMOS33 } [get_ports { JD[2] }]; #IO_L17P_T2_35 Sch=jd[2]
#set_property -dict { PACKAGE_PIN G1    IOSTANDARD LVCMOS33 } [get_ports { JD[3] }]; #IO_L17N_T2_35 Sch=jd[3]
#set_property -dict { PACKAGE_PIN G3    IOSTANDARD LVCMOS33 } [get_ports { JD[4] }]; #IO_L20N_T3_35 Sch=jd[4]
#set_property -dict { PACKAGE_PIN H2    IOSTANDARD LVCMOS33 } [get_ports { JD[7] }]; #IO_L15P_T2_DQS_35 Sch=jd[7]
#set_property -dict { PACKAGE_PIN G4    IOSTANDARD LVCMOS33 } [get_ports { JD[8] }]; #IO_L20P_T3_35 Sch=jd[8]
#set_property -dict { PACKAGE_PIN G2    IOSTANDARD LVCMOS33 } [get_ports { JD[9] }]; #IO_L15N_T2_DQS_35 Sch=jd[9]
#set_property -dict { PACKAGE_PIN F3    IOSTANDARD LVCMOS33 } [get_ports { JD[10] }]; #IO_L13N_T2_MRCC_35 Sch=jd[10]


##Pmod Header JXADC

#set_property -dict { PACKAGE_PIN A14   IOSTANDARD LVDS     } [get_ports { XA_N[1] }]; #IO_L9N_T1_DQS_AD3N_15 Sch=xa_n[1]
#set_property -dict { PACKAGE_PIN A13   IOSTANDARD LVDS     } [get_ports { XA_P[1] }]; #IO_L9P_T1_DQS_AD3P_15 Sch=xa_p[1]
#set_property -dict { PACKAGE_PIN A16   IOSTANDARD LVDS     } [get_ports { XA_N[2] }]; #IO_L8N_T1_AD10N_15 Sch=xa_n[2]
#set_property -dict { PACKAGE_PIN A15   IOSTANDARD LVDS     } [get_ports { XA_P[2] }]; #IO_L8P_T1_AD10P_15 Sch=xa_p[2]
#set_property -dict { PACKAGE_PIN B17   IOSTANDARD LVDS     } [get_ports { XA_N[3] }]; #IO_L7N_T1_AD2N_15 Sch=xa_n[3]
#set_property -dict { PACKAGE_PIN B16   IOSTANDARD LVDS     } [get_ports { XA_P[3] }]; #IO_L7P_T1_AD2P_15 Sch=xa_p[3]
#set_property -dict { PACKAGE_PIN A18   IOSTANDARD LVDS     } [get_ports { XA_N[4] }]; #IO_L10N_T1_AD11N_15 Sch=xa_n[4]
#set_property -dict { PACKAGE_PIN B18   IOSTANDARD LVDS     } [get_ports { XA_P[4] }]; #IO_L10P_T1_AD11P_15 Sch=xa_p[4]


##VGA Connector

set_property -dict { PACKAGE_PIN A3    IOSTANDARD LVCMOS33 } [get_ports { prgb[0] }]; #IO_L8N_T1_AD14N_35 Sch=vga_r[0]
set_property -dict { PACKAGE_PIN B4    IOSTANDARD LVCMOS33 } [get_ports { prgb[1] }]; #IO_L7N_T1_AD6N_35 Sch=vga_r[1]
set_property -dict { PACKAGE_PIN C5    IOSTANDARD LVCMOS33 } [get_ports { prgb[2] }]; #IO_L1N_T0_AD4N_35 Sch=vga_r[2]
set_property -dict { PACKAGE_PIN A4    IOSTANDARD LVCMOS33 } [get_ports { prgb[3] }]; #IO_L8P_T1_AD14P_35 Sch=vga_r[3]
set_property -dict { PACKAGE_PIN C6    IOSTANDARD LVCMOS33 } [get_ports { prgb[4] }]; #IO_L1P_T0_AD4P_35 Sch=vga_g[0]
set_property -dict { PACKAGE_PIN A5    IOSTANDARD LVCMOS33 } [get_ports { prgb[5] }]; #IO_L3N_T0_DQS_AD5N_35 Sch=vga_g[1]
set_property -dict { PACKAGE_PIN B6    IOSTANDARD LVCMOS33 } [get_ports { prgb[6] }]; #IO_L2N_T0_AD12N_35 Sch=vga_g[2]
set_property -dict { PACKAGE_PIN A6    IOSTANDARD LVCMOS33 } [get_ports { prgb[7] }]; #IO_L3P_T0_DQS_AD5P_35 Sch=vga_g[3]
set_property -dict { PACKAGE_PIN B7    IOSTANDARD LVCMOS33 } [get_ports { prgb[8] }]; #IO_L2P_T0_AD12P_35 Sch=vga_b[0]
set_property -dict { PACKAGE_PIN C7    IOSTANDARD LVCMOS33 } [get_ports { prgb[9] }]; #IO_L4N_T0_35 Sch=vga_b[1]
set_property -dict { PACKAGE_PIN D7    IOSTANDARD LVCMOS33 } [get_ports { prgb[10] }]; #IO_L6N_T0_VREF_35 Sch=vga_b[2]
set_property -dict { PACKAGE_PIN D8    IOSTANDARD LVCMOS33 } [get_ports { prgb[11] }]; #IO_L4P_T0_35 Sch=vga_b[3]
set_property -dict { PACKAGE_PIN B11   IOSTANDARD LVCMOS33 } [get_ports { hs}]; #IO_L4P_T0_15 Sch=vga_hs
set_property -dict { PACKAGE_PIN B12   IOSTANDARD LVCMOS33 } [get_ports { vs}]; #IO_L3N_T0_DQS_AD1N_15 Sch=vga_vs


##Micro SD Connector

#set_property -dict { PACKAGE_PIN E2    IOSTANDARD LVCMOS33 } [get_ports { SD_RESET }]; #IO_L14P_T2_SRCC_35 Sch=sd_reset
#set_property -dict { PACKAGE_PIN A1    IOSTANDARD LVCMOS33 } [get_ports { SD_CD }]; #IO_L9N_T1_DQS_AD7N_35 Sch=sd_cd
#set_property -dict { PACKAGE_PIN B1    IOSTANDARD LVCMOS33 } [get_ports { SD_SCK }]; #IO_L9P_T1_DQS_AD7P_35 Sch=sd_sck
#set_property -dict { PACKAGE_PIN C1    IOSTANDARD LVCMOS33 } [get_ports { SD_CMD }]; #IO_L16N_T2_35 Sch=sd_cmd
#set_property -dict { PACKAGE_PIN C2    IOSTANDARD LVCMOS33 } [get_ports { SD_DAT[0] }]; #IO_L16P_T2_35 Sch=sd_dat[0]
#set_property -dict { PACKAGE_PIN E1    IOSTANDARD LVCMOS33 } [get_ports { SD_DAT[1] }]; #IO_L18N_T2_35 Sch=sd_dat[1]
#set_property -dict { PACKAGE_PIN F1    IOSTANDARD LVCMOS33 } [get_ports { SD_DAT[2] }]; #IO_L18P_T2_35 Sch=sd_dat[2]
#set_property -dict { PACKAGE_PIN D2    IOSTANDARD LVCMOS33 } [get_ports { SD_DAT[3] }]; #IO_L14N_T2_SRCC_35 Sch=sd_dat[3]


##Accelerometer

#set_property -dict { PACKAGE_PIN E15   IOSTANDARD LVCMOS33 } [get_ports { ACL_MISO }]; #IO_L11P_T1_SRCC_15 Sch=acl_miso
#set_property -dict { PACKAGE_PIN F14   IOSTANDARD LVCMOS33 } [get_ports { ACL_MOSI }]; #IO_L5N_T0_AD9N_15 Sch=acl_mosi
#set_property -dict { PACKAGE_PIN F15   IOSTANDARD LVCMOS33 } [get_ports { ACL_SCLK }]; #IO_L14P_T2_SRCC_15 Sch=acl_sclk
#set_property -dict { PACKAGE_PIN D15   IOSTANDARD LVCMOS33 } [get_ports { ACL_CSN }]; #IO_L12P_T1_MRCC_15 Sch=acl_csn
#set_property -dict { PACKAGE_PIN B13   IOSTANDARD LVCMOS33 } [get_ports { ACL_INT[1] }]; #IO_L2P_T0_AD8P_15 Sch=acl_int[1]
#set_property -dict { PACKAGE_PIN C16   IOSTANDARD LVCMOS33 } [get_ports { ACL_INT[2] }]; #IO_L20P_T3_A20_15 Sch=acl_int[2]


##Temperature Sensor

#set_property -dict { PACKAGE_PIN C14   IOSTANDARD LVCMOS33 } [get_ports { TMP_SCL }]; #IO_L1N_T0_AD0N_15 Sch=tmp_scl
#set_property -dict { PACKAGE_PIN C15   IOSTANDARD LVCMOS33 } [get_ports { TMP_SDA }]; #IO_L12N_T1_MRCC_15 Sch=tmp_sda
#set_property -dict { PACKAGE_PIN D13   IOSTANDARD LVCMOS33 } [get_ports { TMP_INT }]; #IO_L6N_T0_VREF_15 Sch=tmp_int
#set_property -dict { PACKAGE_PIN B14   IOSTANDARD LVCMOS33 } [get_ports { TMP_CT }]; #IO_L2N_T0_AD8N_15 Sch=tmp_ct

##Omnidirectional Microphone

#set_property -dict { PACKAGE_PIN J5    IOSTANDARD LVCMOS33 } [get_ports { M_CLK }]; #IO_25_35 Sch=m_clk
```

```
#set_property -dict { PACKAGE_PIN H5    IOSTANDARD LVCMOS33 } [get_ports { M_DATA }]; #IO_L24N_T3_35 Sch=m_data
#set_property -dict { PACKAGE_PIN F5    IOSTANDARD LVCMOS33 } [get_ports { M_LRSEL }]; #IO_0_35 Sch=m_lrsel


##PWM Audio Amplifier

#set_property -dict { PACKAGE_PIN A11   IOSTANDARD LVCMOS33 } [get_ports { AUD_PWM }]; #IO_L4N_T0_15 Sch=aud_pwm
#set_property -dict { PACKAGE_PIN D12   IOSTANDARD LVCMOS33 } [get_ports { AUD_SD }]; #IO_L6P_T0_15 Sch=aud_sd


##USB-RS232 Interface

#set_property -dict { PACKAGE_PIN C4    IOSTANDARD LVCMOS33 } [get_ports { UART_TXD_IN }]; #IO_L7P_T1_AD6P_35 Sch=uart_txd_in
#set_property -dict { PACKAGE_PIN D4    IOSTANDARD LVCMOS33 } [get_ports { UART_RXD_OUT }]; #IO_L11N_T1_SRCC_35 Sch=uart_rxd_out
#set_property -dict { PACKAGE_PIN D3    IOSTANDARD LVCMOS33 } [get_ports { UART_CTS }]; #IO_L12N_T1_MRCC_35 Sch=uart_cts
#set_property -dict { PACKAGE_PIN E5    IOSTANDARD LVCMOS33 } [get_ports { UART_RTS }]; #IO_L5N_T0_AD13N_35 Sch=uart_rts

##USB HID (PS/2)

#set_property -dict { PACKAGE_PIN F4    IOSTANDARD LVCMOS33 } [get_ports { PS2_CLK }]; #IO_L13P_T2_MRCC_35 Sch=ps2_clk
#set_property -dict { PACKAGE_PIN B2    IOSTANDARD LVCMOS33 } [get_ports { PS2_DATA }]; #IO_L10N_T1_AD15N_35 Sch=ps2_data


##SMSC Ethernet PHY

#set_property -dict { PACKAGE_PIN C9    IOSTANDARD LVCMOS33 } [get_ports { ETH_MDC }]; #IO_L11P_T1_SRCC_16 Sch=eth_mdc
#set_property -dict { PACKAGE_PIN A9    IOSTANDARD LVCMOS33 } [get_ports { ETH_MDIO }]; #IO_L14N_T2_SRCC_16 Sch=eth_mdio
#set_property -dict { PACKAGE_PIN B3    IOSTANDARD LVCMOS33 } [get_ports { ETH_RSTN }]; #IO_L10P_T1_AD15P_35 Sch=eth_rstn
#set_property -dict { PACKAGE_PIN D9    IOSTANDARD LVCMOS33 } [get_ports { ETH_CRSDV }]; #IO_L6N_T0_VREF_16 Sch=eth_crsdv
#set_property -dict { PACKAGE_PIN C10   IOSTANDARD LVCMOS33 } [get_ports { ETH_RXERR }]; #IO_L13N_T2_MRCC_16 Sch=eth_rxerr
#set_property -dict { PACKAGE_PIN C11   IOSTANDARD LVCMOS33 } [get_ports { ETH_RXD[0] }]; #IO_L13P_T2_MRCC_16 Sch=eth_rxd[0]
#set_property -dict { PACKAGE_PIN D10   IOSTANDARD LVCMOS33 } [get_ports { ETH_RXD[1] }]; #IO_L19N_T3_VREF_16 Sch=eth_rxd[1]
#set_property -dict { PACKAGE_PIN B9    IOSTANDARD LVCMOS33 } [get_ports { ETH_TXEN }]; #IO_L11N_T1_SRCC_16 Sch=eth_txen
#set_property -dict { PACKAGE_PIN A10   IOSTANDARD LVCMOS33 } [get_ports { ETH_TXD[0] }]; #IO_L14P_T2_SRCC_16 Sch=eth_txd[0]
#set_property -dict { PACKAGE_PIN A8    IOSTANDARD LVCMOS33 } [get_ports { ETH_TXD[1] }]; #IO_L12N_T1_MRCC_16 Sch=eth_txd[1]
#set_property -dict { PACKAGE_PIN D5    IOSTANDARD LVCMOS33 } [get_ports { ETH_REFCLK }]; #IO_L11P_T1_SRCC_35 Sch=eth_refclk
#set_property -dict { PACKAGE_PIN B8    IOSTANDARD LVCMOS33 } [get_ports { ETH_INTN }]; #IO_L12P_T1_MRCC_16 Sch=eth_intn


##Quad SPI Flash

#set_property -dict { PACKAGE_PIN K17   IOSTANDARD LVCMOS33 } [get_ports { QSPI_DQ[0] }]; #IO_L1P_T0_D00_MOSI_14 Sch=qspi_dq[0]
#set_property -dict { PACKAGE_PIN K18   IOSTANDARD LVCMOS33 } [get_ports { QSPI_DQ[1] }]; #IO_L1N_T0_D01_DIN_14 Sch=qspi_dq[1]
#set_property -dict { PACKAGE_PIN L14   IOSTANDARD LVCMOS33 } [get_ports { QSPI_DQ[2] }]; #IO_L2P_T0_D02_14 Sch=qspi_dq[2]
#set_property -dict { PACKAGE_PIN M14   IOSTANDARD LVCMOS33 } [get_ports { QSPI_DQ[3] }]; #IO_L2N_T0_D03_14 Sch=qspi_dq[3]
#set_property -dict { PACKAGE_PIN L13   IOSTANDARD LVCMOS33 } [get_ports { QSPI_CSN }]; #IO_L6P_T0_FCS_B_14 Sch=qspi_csn
```

4.5 综合设计

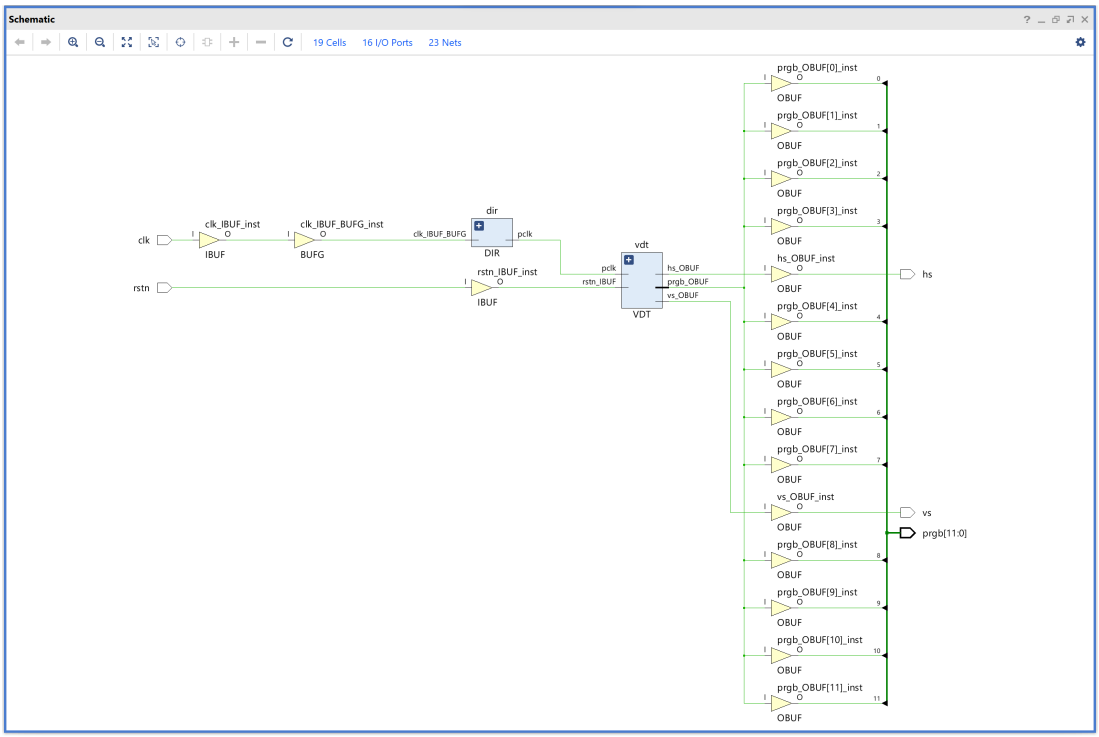

图 3: 综合设计电路图

4.6 电路资源使用情况

```
Copyright 1986–2019 Xilinx, Inc. All Rights Reserved.
------------------------------------------------------------------------------------------------------------------------
| Tool Version : Vivado v.2019.1 (win64) Build 2552052 Fri May 24 14:49:42 MDT 2019
| Date         : Sun Nov 28 17:36:29 2021
| Host         : DESKTOP-DM7M7LO running 64-bit major release  (build 9200)
| Command      : report_utilization -file test_utilization_synth.rpt -pb test_utilization_synth.pb
| Design       : test
| Device       : 7a100tcsg324-1
| Design State : Synthesized
------------------------------------------------------------------------------------------------------------------------

Utilization Design Information

Table of Contents
-----------------
1. Slice Logic
1.1 Summary of Registers by Type
2. Memory
3. DSP
4. IO and GT Specific
5. Clocking
6. Specific Feature
7. Primitives
8. Black Boxes
9. Instantiated Netlists

1. Slice Logic
--------------

+----------------------------+------+-------+-----------+-------+
|          Site Type         | Used | Fixed | Available | Util% |
+----------------------------+------+-------+-----------+-------+
| Slice LUTs*                |   42 |     0 |     63400 |  0.07 |
|   LUT as Logic             |   42 |     0 |     63400 |  0.07 |
|   LUT as Memory            |    0 |     0 |     19000 |  0.00 |
| Slice Registers            |   26 |     0 |    126800 |  0.02 |
|   Register as Flip Flop    |   26 |     0 |    126800 |  0.02 |
|   Register as Latch        |    0 |     0 |    126800 |  0.00 |
| F7 Muxes                   |    0 |     0 |     31700 |  0.00 |
| F8 Muxes                   |    0 |     0 |     15850 |  0.00 |
```

* Warning! The Final LUT count, after physical optimizations and full implementation, is typically lower. Run opt_design after synthesis, if not already completed, for a more realistic count.				

1.1 Summary of Registers by Type

Total	Clock Enable	Synchronous	Asynchronous
0	_	-	-
0	_	-	Set
0	_	-	Reset
0	_	Set	-
0	_	Reset	-
0	Yes	-	-
0	Yes	-	Set
0	Yes	-	Reset
13	Yes	Set	-
13	Yes	Reset	-

2. Memory

Site Type	Used	Fixed	Available	Util%
Block RAM Tile	0	0	135	0.00
RAMB36/FIFO*	0	0	135	0.00
RAMB18	0	0	270	0.00

* Note: Each Block RAM Tile only has one FIFO logic available and therefore can accommodate only one FIFO36E1 or one FIFO18E1. However, if a FIFO18E1 occupies a Block RAM Tile, that tile can still accommodate a RAMB18E1

3. DSP

Site Type	Used	Fixed	Available	Util%
DSPs	0	0	240	0.00

4. IO and GT Specific

Site Type	Used	Fixed	Available	Util%
Bonded IOB	16	0	210	7.62
Bonded IPADs	0	0	2	0.00
PHY_CONTROL	0	0	6	0.00
PHASER_REF	0	0	6	0.00
OUT_FIFO	0	0	24	0.00
IN_FIFO	0	0	24	0.00
IDELAYCTRL	0	0	6	0.00
IBUFDS	0	0	202	0.00
PHASER_OUT/PHASER_OUT_PHY	0	0	24	0.00
PHASER_IN/PHASER_IN_PHY	0	0	24	0.00
IDELAYE2/IDELAYE2_FINEDELAY	0	0	300	0.00
ILOGIC	0	0	210	0.00
OLOGIC	0	0	210	0.00

5. Clocking

Site Type	Used	Fixed	Available	Util%
BUFGCTRL	1	0	32	3.13
BUFIO	0	0	24	0.00
MMCME2_ADV	0	0	6	0.00
PLLE2_ADV	0	0	6	0.00
BUFMRCE	0	0	12	0.00
BUFHCE	0	0	96	0.00

| BUFR | 0 | 0 | 24 | 0.00 |

6. Specific Feature

Site Type	Used	Fixed	Available	Util%
BSCANE2	0	0	4	0.00
CAPTUREE2	0	0	1	0.00
DNA_PORT	0	0	1	0.00
EFUSE_USR	0	0	1	0.00
FRAME_ECCE2	0	0	1	0.00
ICAPE2	0	0	2	0.00
PCIE_2_1	0	0	1	0.00
STARTUPE2	0	0	1	0.00
XADC	0	0	1	0.00

7. Primitives

Ref Name	Used	Functional Category
LUT6	16	LUT
OBUF	14	IO
LUT4	13	LUT
FDSE	13	Flop & Latch
FDRE	13	Flop & Latch
LUT5	12	LUT
LUT3	9	LUT
LUT2	6	LUT
LUT1	2	LUT
IBUF	2	IO
BUFG	1	Clock

8. Black Boxes

Ref Name	Used

9. Instantiated Netlists

Ref Name	Used

4.7 下载结果

图 4: 下载结果

5 实现 DCU，显示彩色方格图案

5.1 逻辑设计

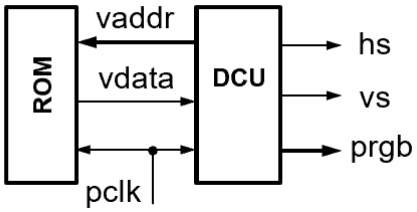

图 5: DCU 数据通路

5.2 设计文件

5.2.1 视频显示合成

```
‘timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 28.11.2021 10:39:04
// Design Name:
// Module Name: VDS
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////


module VDS(input hen,
           ven,
           pclk,
           rstn,
           [11:0]rdata,
           output reg [14:0]raddr,
           reg[11:0]prgb);
    always @(posedge pclk) begin
        if (hen&&ven)
            prgb = rdata;
        else
            prgb = 0;
    end
    reg [19:0]cnt;
    always @(posedge pclk) begin
        if (!rstn||cnt == 480000-1)
            cnt <= 0;
        else if (hen&&ven)
            cnt <= cnt+1;
            end
            always @(posedge pclk) begin
                if (hen&&ven)
                    raddr <= (cnt/3200)*200+(cnt%800)/4;
                else
                    raddr <= 30000;
            end
            endmodule
```

5.2.2 只读存储器

```
-- (c) Copyright 1995-2021 Xilinx, Inc. All rights reserved.
--
```

```
-- This file contains confidential and proprietary information
-- of Xilinx, Inc. and is protected under U.S. and
-- international copyright and other intellectual property
-- laws.
--
-- DISCLAIMER
-- This disclaimer is not a license and does not grant any
-- rights to the materials distributed herewith. Except as
-- otherwise provided in a valid license issued to you by
-- Xilinx, and to the maximum extent permitted by applicable
-- law: (1) THESE MATERIALS ARE MADE AVAILABLE "AS IS" AND
-- WITH ALL FAULTS, AND XILINX HEREBY DISCLAIMS ALL WARRANTIES
-- AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING
-- BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-
-- INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and
-- (2) Xilinx shall not be liable (whether in contract or tort,
-- including negligence, or under any other theory of
-- liability) for any loss or damage of any kind or nature
-- related to, arising under or in connection with these
-- materials, including for any direct, or any indirect,
-- special, incidental, or consequential loss or damage
-- (including loss of data, profits, goodwill, or any type of
-- loss or damage suffered as a result of any action brought
-- by a third party) even if such damage or loss was
-- reasonably foreseeable or Xilinx had been advised of the
-- possibility of the same.
--
-- CRITICAL APPLICATIONS
-- Xilinx products are not designed or intended to be fail-
-- safe, or for use in any application requiring fail-safe
-- performance, such as life-support or safety devices or
-- systems, Class III medical devices, nuclear facilities,
-- applications related to the deployment of airbags, or any
-- other applications that could lead to death, personal
-- injury, or severe property or environmental damage
-- (individually and collectively, "Critical
-- Applications"). Customer assumes the sole risk and
-- liability of any use of Xilinx products in Critical
-- Applications, subject only to applicable laws and
-- regulations governing limitations on product liability.
--
-- THIS COPYRIGHT NOTICE AND DISCLAIMER MUST BE RETAINED AS
-- PART OF THIS FILE AT ALL TIMES.
--
-- DO NOT MODIFY THIS FILE.

-- IP VLNV: xilinx.com:ip:blk_mem_gen:8.4
-- IP Revision: 3

LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
USE ieee.numeric_std.ALL;

LIBRARY blk_mem_gen_v8_4_3;
USE blk_mem_gen_v8_4_3.blk_mem_gen_v8_4_3;

ENTITY blk_mem_gen_1 IS
  PORT (
    clka : IN STD_LOGIC;
    addra : IN STD_LOGIC_VECTOR(5 DOWNTO 0);
    douta : OUT STD_LOGIC_VECTOR(1 DOWNTO 0)
  );
END blk_mem_gen_1;

ARCHITECTURE blk_mem_gen_1_arch OF blk_mem_gen_1 IS
  ATTRIBUTE DowngradeIPIdentifiedWarnings : STRING;
  ATTRIBUTE DowngradeIPIdentifiedWarnings OF blk_mem_gen_1_arch: ARCHITECTURE IS "yes";
  COMPONENT blk_mem_gen_v8_4_3 IS
    GENERIC (
      C_FAMILY : STRING;
      C_XDEVICEFAMILY : STRING;
      C_ELABORATION_DIR : STRING;
      C_INTERFACE_TYPE : INTEGER;
      C_AXI_TYPE : INTEGER;
      C_AXI_SLAVE_TYPE : INTEGER;
      C_USE_BRAM_BLOCK : INTEGER;
      C_ENABLE_32BIT_ADDRESS : INTEGER;
      C_CTRL_ECC_ALGO : STRING;
      C_HAS_AXI_ID : INTEGER;
      C_AXI_ID_WIDTH : INTEGER;
      C_MEM_TYPE : INTEGER;
      C_BYTE_SIZE : INTEGER;
```

```
    C_ALGORITHM : INTEGER;
    C_PRIM_TYPE : INTEGER;
    C_LOAD_INIT_FILE : INTEGER;
    C_INIT_FILE_NAME : STRING;
    C_INIT_FILE : STRING;
    C_USE_DEFAULT_DATA : INTEGER;
    C_DEFAULT_DATA : STRING;
    C_HAS_RSTA : INTEGER;
    C_RST_PRIORITY_A : STRING;
    C_RSTRAM_A : INTEGER;
    C_INITA_VAL : STRING;
    C_HAS_ENA : INTEGER;
    C_HAS_REGCEA : INTEGER;
    C_USE_BYTE_WEA : INTEGER;
    C_WEA_WIDTH : INTEGER;
    C_WRITE_MODE_A : STRING;
    C_WRITE_WIDTH_A : INTEGER;
    C_READ_WIDTH_A : INTEGER;
    C_WRITE_DEPTH_A : INTEGER;
    C_READ_DEPTH_A : INTEGER;
    C_ADDRA_WIDTH : INTEGER;
    C_HAS_RSTB : INTEGER;
    C_RST_PRIORITY_B : STRING;
    C_RSTRAM_B : INTEGER;
    C_INITB_VAL : STRING;
    C_HAS_ENB : INTEGER;
    C_HAS_REGCEB : INTEGER;
    C_USE_BYTE_WEB : INTEGER;
    C_WEB_WIDTH : INTEGER;
    C_WRITE_MODE_B : STRING;
    C_WRITE_WIDTH_B : INTEGER;
    C_READ_WIDTH_B : INTEGER;
    C_WRITE_DEPTH_B : INTEGER;
    C_READ_DEPTH_B : INTEGER;
    C_ADDRB_WIDTH : INTEGER;
    C_HAS_MEM_OUTPUT_REGS_A : INTEGER;
    C_HAS_MEM_OUTPUT_REGS_B : INTEGER;
    C_HAS_MUX_OUTPUT_REGS_A : INTEGER;
    C_HAS_MUX_OUTPUT_REGS_B : INTEGER;
    C_MUX_PIPELINE_STAGES : INTEGER;
    C_HAS_SOFTECC_INPUT_REGS_A : INTEGER;
    C_HAS_SOFTECC_OUTPUT_REGS_B : INTEGER;
    C_USE_SOFTECC : INTEGER;
    C_USE_ECC : INTEGER;
    C_EN_ECC_PIPE : INTEGER;
    C_READ_LATENCY_A : INTEGER;
    C_READ_LATENCY_B : INTEGER;
    C_HAS_INJECTERR : INTEGER;
    C_SIM_COLLISION_CHECK : STRING;
    C_COMMON_CLK : INTEGER;
    C_DISABLE_WARN_BHV_COLL : INTEGER;
    C_EN_SLEEP_PIN : INTEGER;
    C_USE_URAM : INTEGER;
    C_EN_RDADDRA_CHG : INTEGER;
    C_EN_RDADDRB_CHG : INTEGER;
    C_EN_DEEP_SLEEP_PIN : INTEGER;
    C_EN_SHUTDOWN_PIN : INTEGER;
    C_EN_SAFETY_CKT : INTEGER;
    C_DISABLE_WARN_BHV_RANGE : INTEGER;
    C_COUNT_36K_BRAM : STRING;
    C_COUNT_18K_BRAM : STRING;
    C_EST_POWER_SUMMARY : STRING
  );
  PORT (
    clka : IN STD_LOGIC;
    rsta : IN STD_LOGIC;
    ena : IN STD_LOGIC;
    regcea : IN STD_LOGIC;
    wea : IN STD_LOGIC_VECTOR(0 DOWNTO 0);
    addra : IN STD_LOGIC_VECTOR(5 DOWNTO 0);
    dina : IN STD_LOGIC_VECTOR(1 DOWNTO 0);
    douta : OUT STD_LOGIC_VECTOR(1 DOWNTO 0);
    clkb : IN STD_LOGIC;
    rstb : IN STD_LOGIC;
    enb : IN STD_LOGIC;
    regceb : IN STD_LOGIC;
    web : IN STD_LOGIC_VECTOR(0 DOWNTO 0);
    addrb : IN STD_LOGIC_VECTOR(5 DOWNTO 0);
    dinb : IN STD_LOGIC_VECTOR(1 DOWNTO 0);
    doutb : OUT STD_LOGIC_VECTOR(1 DOWNTO 0);
    injectsbiterr : IN STD_LOGIC;
    injectdbiterr : IN STD_LOGIC;
```

```
      eccpipece : IN STD_LOGIC;
      sbiterr : OUT STD_LOGIC;
      dbiterr : OUT STD_LOGIC;
      rdaddrecc : OUT STD_LOGIC_VECTOR(5 DOWNTO 0);
      sleep : IN STD_LOGIC;
      deepsleep : IN STD_LOGIC;
      shutdown : IN STD_LOGIC;
      rsta_busy : OUT STD_LOGIC;
      rstb_busy : OUT STD_LOGIC;
      s_aclk : IN STD_LOGIC;
      s_aresetn : IN STD_LOGIC;
      s_axi_awid : IN STD_LOGIC_VECTOR(3 DOWNTO 0);
      s_axi_awaddr : IN STD_LOGIC_VECTOR(31 DOWNTO 0);
      s_axi_awlen : IN STD_LOGIC_VECTOR(7 DOWNTO 0);
      s_axi_awsize : IN STD_LOGIC_VECTOR(2 DOWNTO 0);
      s_axi_awburst : IN STD_LOGIC_VECTOR(1 DOWNTO 0);
      s_axi_awvalid : IN STD_LOGIC;
      s_axi_awready : OUT STD_LOGIC;
      s_axi_wdata : IN STD_LOGIC_VECTOR(1 DOWNTO 0);
      s_axi_wstrb : IN STD_LOGIC_VECTOR(0 DOWNTO 0);
      s_axi_wlast : IN STD_LOGIC;
      s_axi_wvalid : IN STD_LOGIC;
      s_axi_wready : OUT STD_LOGIC;
      s_axi_bid : OUT STD_LOGIC_VECTOR(3 DOWNTO 0);
      s_axi_bresp : OUT STD_LOGIC_VECTOR(1 DOWNTO 0);
      s_axi_bvalid : OUT STD_LOGIC;
      s_axi_bready : IN STD_LOGIC;
      s_axi_arid : IN STD_LOGIC_VECTOR(3 DOWNTO 0);
      s_axi_araddr : IN STD_LOGIC_VECTOR(31 DOWNTO 0);
      s_axi_arlen : IN STD_LOGIC_VECTOR(7 DOWNTO 0);
      s_axi_arsize : IN STD_LOGIC_VECTOR(2 DOWNTO 0);
      s_axi_arburst : IN STD_LOGIC_VECTOR(1 DOWNTO 0);
      s_axi_arvalid : IN STD_LOGIC;
      s_axi_arready : OUT STD_LOGIC;
      s_axi_rid : OUT STD_LOGIC_VECTOR(3 DOWNTO 0);
      s_axi_rdata : OUT STD_LOGIC_VECTOR(1 DOWNTO 0);
      s_axi_rresp : OUT STD_LOGIC_VECTOR(1 DOWNTO 0);
      s_axi_rlast : OUT STD_LOGIC;
      s_axi_rvalid : OUT STD_LOGIC;
      s_axi_rready : IN STD_LOGIC;
      s_axi_injectsbiterr : IN STD_LOGIC;
      s_axi_injectdbiterr : IN STD_LOGIC;
      s_axi_sbiterr : OUT STD_LOGIC;
      s_axi_dbiterr : OUT STD_LOGIC;
      s_axi_rdaddrecc : OUT STD_LOGIC_VECTOR(5 DOWNTO 0)
    );
  END COMPONENT blk_mem_gen_v8_4_3;
  ATTRIBUTE X_CORE_INFO : STRING;
  ATTRIBUTE X_CORE_INFO OF blk_mem_gen_1_arch: ARCHITECTURE IS "blk_mem_gen_v8_4_3,Vivado 2019.1";
  ATTRIBUTE CHECK_LICENSE_TYPE : STRING;
  ATTRIBUTE CHECK_LICENSE_TYPE OF blk_mem_gen_1_arch : ARCHITECTURE IS "blk_mem_gen_1,blk_mem_gen_v8_4_3,{}";
  ATTRIBUTE CORE_GENERATION_INFO : STRING;
  ATTRIBUTE CORE_GENERATION_INFO OF blk_mem_gen_1_arch: ARCHITECTURE IS "blk_mem_gen_1,blk_mem_gen_v8_4_3,{x_ipProduct=Vivado 2019.1,x_ipVendor=xilinx
"em_gen_1.mif,C_INIT_FILE=blk_mem_gen_1.mem,C_USE_DEFAULT_DATA=1,C_DEFAULT_DATA=0,C_HAS_RSTA=0,C_RST_PRIORITY_A=CE,C_RSTRAM_A=0,C_INITA_VAL=0,C_HAS_ENA
"E_DEPTH_B=64,C_READ_DEPTH_B=64,C_ADDRB_WIDTH=6,C_HAS_MEM_OUTPUT_REGS_A=0,C_HAS_MEM_OUTPUT_REGS_B=0,C_HAS_MUX_OUTPUT_REGS_A=0,C_HAS_MUX_OUTPUT_REGS_B=
"OWN_PIN=0,C_EN_SAFETY_CKT=0,C_DISABLE_WARN_BHV_RANGE=0,C_COUNT_36K_BRAM=0,C_COUNT_18K_BRAM=1,C_EST_POWER_SUMMARY=Estimated Power for IP
_     2.048762 mW}";
  ATTRIBUTE X_INTERFACE_INFO : STRING;
  ATTRIBUTE X_INTERFACE_PARAMETER : STRING;
  ATTRIBUTE X_INTERFACE_INFO OF douta: SIGNAL IS "xilinx.com:interface:bram:1.0 BRAM_PORTA DOUT";
  ATTRIBUTE X_INTERFACE_INFO OF addra: SIGNAL IS "xilinx.com:interface:bram:1.0 BRAM_PORTA ADDR";
  ATTRIBUTE X_INTERFACE_PARAMETER OF clka: SIGNAL IS "XIL_INTERFACENAME BRAM_PORTA, MEM_SIZE 8192, MEM_WIDTH 32, MEM_ECC NONE, MASTER_TYPE OTHER, READ_
  ATTRIBUTE X_INTERFACE_INFO OF clka: SIGNAL IS "xilinx.com:interface:bram:1.0 BRAM_PORTA CLK";
BEGIN
  U0 : blk_mem_gen_v8_4_3
    GENERIC MAP (
      C_FAMILY => "artix7",
      C_XDEVICEFAMILY => "artix7",
      C_ELABORATION_DIR => "./",
      C_INTERFACE_TYPE => 0,
      C_AXI_TYPE => 1,
      C_AXI_SLAVE_TYPE => 0,
      C_USE_BRAM_BLOCK => 0,
      C_ENABLE_32BIT_ADDRESS => 0,
      C_CTRL_ECC_ALGO => "NONE",
      C_HAS_AXI_ID => 0,
      C_AXI_ID_WIDTH => 4,
      C_MEM_TYPE => 3,
      C_BYTE_SIZE => 9,
      C_ALGORITHM => 1,
      C_PRIM_TYPE => 1,
      C_LOAD_INIT_FILE => 1,
```

```
    C_INIT_FILE_NAME => "blk_mem_gen_1.mif",
    C_INIT_FILE => "blk_mem_gen_1.mem",
    C_USE_DEFAULT_DATA => 1,
    C_DEFAULT_DATA => "0",
    C_HAS_RSTA => 0,
    C_RST_PRIORITY_A => "CE",
    C_RSTRAM_A => 0,
    C_INITA_VAL => "0",
    C_HAS_ENA => 0,
    C_HAS_REGCEA => 0,
    C_USE_BYTE_WEA => 0,
    C_WEA_WIDTH => 1,
    C_WRITE_MODE_A => "WRITE_FIRST",
    C_WRITE_WIDTH_A => 2,
    C_READ_WIDTH_A => 2,
    C_WRITE_DEPTH_A => 64,
    C_READ_DEPTH_A => 64,
    C_ADDRA_WIDTH => 6,
    C_HAS_RSTB => 0,
    C_RST_PRIORITY_B => "CE",
    C_RSTRAM_B => 0,
    C_INITB_VAL => "0",
    C_HAS_ENB => 0,
    C_HAS_REGCEB => 0,
    C_USE_BYTE_WEB => 0,
    C_WEB_WIDTH => 1,
    C_WRITE_MODE_B => "WRITE_FIRST",
    C_WRITE_WIDTH_B => 2,
    C_READ_WIDTH_B => 2,
    C_WRITE_DEPTH_B => 64,
    C_READ_DEPTH_B => 64,
    C_ADDRB_WIDTH => 6,
    C_HAS_MEM_OUTPUT_REGS_A => 0,
    C_HAS_MEM_OUTPUT_REGS_B => 0,
    C_HAS_MUX_OUTPUT_REGS_A => 0,
    C_HAS_MUX_OUTPUT_REGS_B => 0,
    C_MUX_PIPELINE_STAGES => 0,
    C_HAS_SOFTECC_INPUT_REGS_A => 0,
    C_HAS_SOFTECC_OUTPUT_REGS_B => 0,
    C_USE_SOFTECC => 0,
    C_USE_ECC => 0,
    C_EN_ECC_PIPE => 0,
    C_READ_LATENCY_A => 1,
    C_READ_LATENCY_B => 1,
    C_HAS_INJECTERR => 0,
    C_SIM_COLLISION_CHECK => "ALL",
    C_COMMON_CLK => 0,
    C_DISABLE_WARN_BHV_COLL => 0,
    C_EN_SLEEP_PIN => 0,
    C_USE_URAM => 0,
    C_EN_RDADDRA_CHG => 0,
    C_EN_RDADDRB_CHG => 0,
    C_EN_DEEPSLEEP_PIN => 0,
    C_EN_SHUTDOWN_PIN => 0,
    C_EN_SAFETY_CKT => 0,
    C_DISABLE_WARN_BHV_RANGE => 0,
    C_COUNT_36K_BRAM => "0",
    C_COUNT_18K_BRAM => "1",
    C_EST_POWER_SUMMARY => "Estimated Power for IP     :     2.048762 mW"
  )
  PORT MAP (
    clka => clka,
    rsta => '0',
    ena => '0',
    regcea => '0',
    wea => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 1)),
    addra => addra,
    dina => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 2)),
    douta => douta,
    clkb => '0',
    rstb => '0',
    enb => '0',
    regceb => '0',
    web => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 1)),
    addrb => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 6)),
    dinb => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 2)),
    injectsbiterr => '0',
    injectdbiterr => '0',
    eccpipece => '0',
    sleep => '0',
    deepsleep => '0',
    shutdown => '0',
```

```
        s_aclk => '0',
        s_aresetn => '0',
        s_axi_awid => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 4)),
        s_axi_awaddr => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 32)),
        s_axi_awlen => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 8)),
        s_axi_awsize => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 3)),
        s_axi_awburst => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 2)),
        s_axi_awvalid => '0',
        s_axi_wdata => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 2)),
        s_axi_wstrb => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 1)),
        s_axi_wlast => '0',
        s_axi_wvalid => '0',
        s_axi_bready => '0',
        s_axi_arid => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 4)),
        s_axi_araddr => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 32)),
        s_axi_arlen => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 8)),
        s_axi_arsize => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 3)),
        s_axi_arburst => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 2)),
        s_axi_arvalid => '0',
        s_axi_rready => '0',
        s_axi_injectsbiterr => '0',
        s_axi_injectdbiterr => '0'
      );
END blk_mem_gen_1_arch;
```

5.2.3 显示控制单元

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 28.11.2021 18:33:42
// Design Name:
// Module Name: DCU
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////


module DCU(input clk,
           rstn,
           output hs,
           vs,
           [11:0]prgb);
    wire hen,ven,pclk;
    wire[1:0]rdata;
    wire[5:0]raddr;
    DIR dir(.clk(clk),.pclk(pclk));
    VDT vdt(.pclk(pclk),.rstn(rstn),.hen(hen),.ven(ven),.hs(hs),.vs(vs));
    VDS vds(.hen(hen),.ven(ven),.pclk(pclk),.rstn(rstn),.rdata(rdata),.raddr(raddr),.prgb(prgb));
    blk_mem_gen_1 rom(.addra(raddr),.clka(pclk),.douta(rdata));
endmodule
```

5.3 RTL 分析

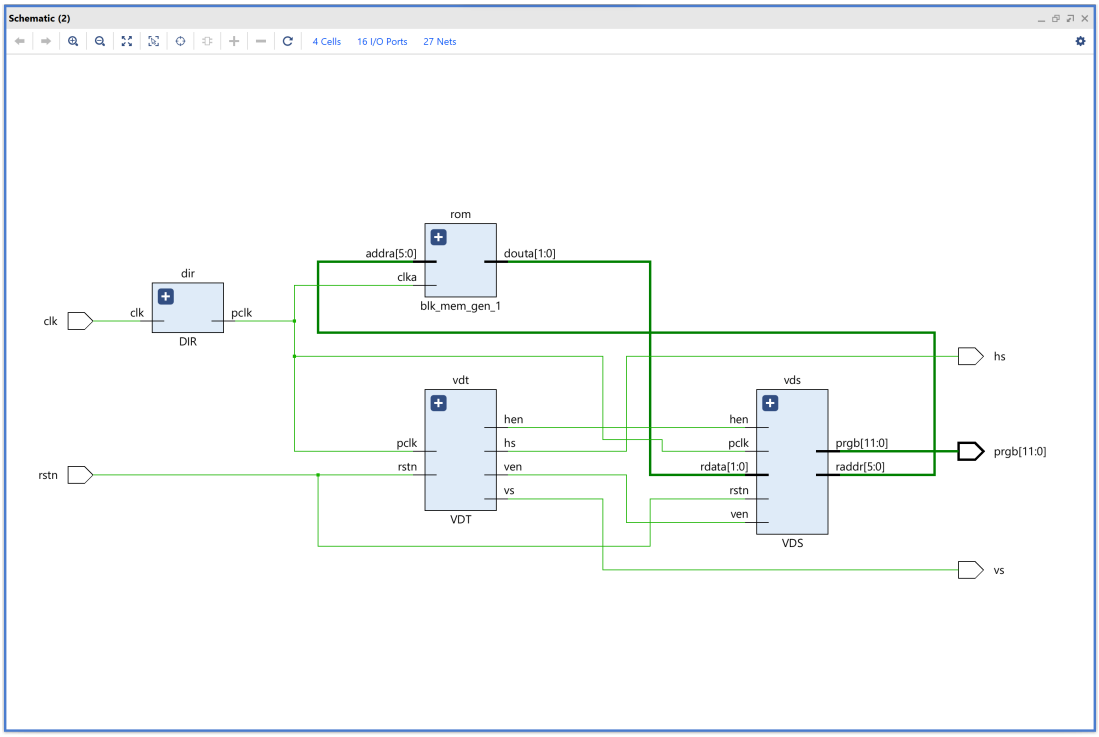

图 6: RTL 分析电路图

5.4 综合设计

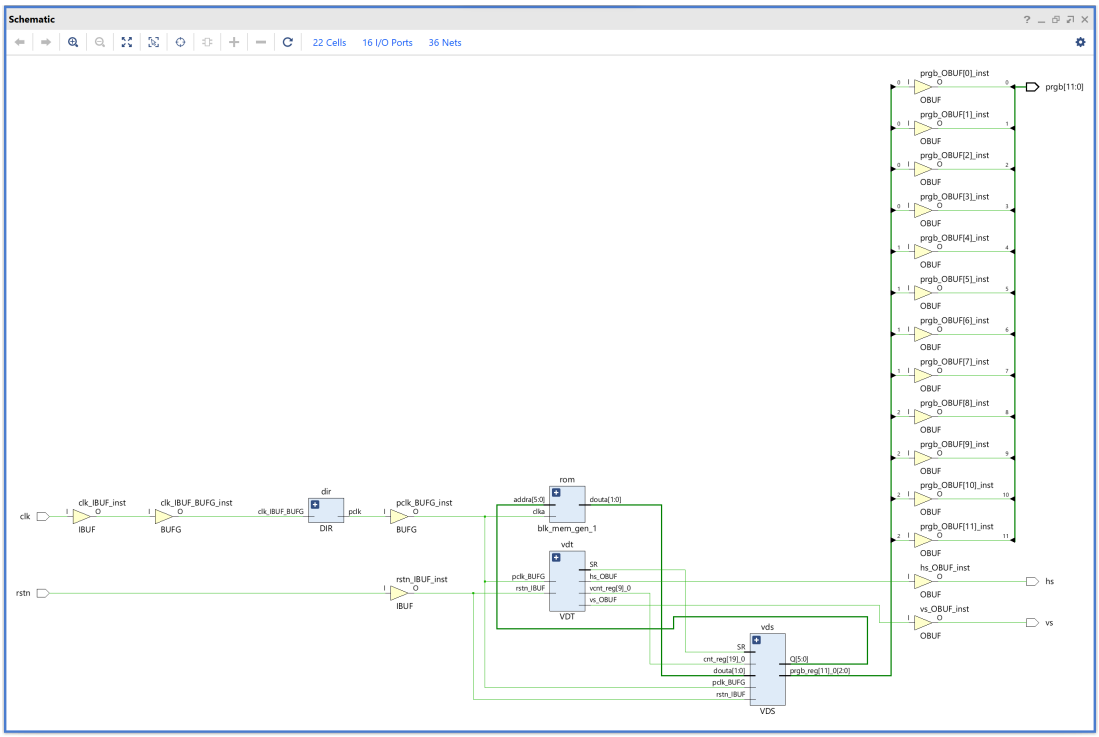

图 7: 综合设计电路图

5.5 电路资源使用情况

```
| Tool Version : Vivado v.2019.1 (win64) Build 2552052 Fri May 24 14:49:42 MDT 2019
| Date         : Sun Nov 28 20:07:02 2021
| Host         : DESKTOP-DM7M7LO running 64-bit major release  (build 9200)
| Command      : report_utilization -file DCU_utilization_synth.rpt -pb DCU_utilization_synth.pb
| Design       : DCU
| Device       : 7a100tcsg324-1
| Design State : Synthesized
```

Utilization Design Information

Table of Contents

1. Slice Logic

Site Type	Used	Fixed	Available	Util%
Slice LUTs*	214	0	63400	0.34
LUT as Logic	214	0	63400	0.34
LUT as Memory	0	0	19000	0.00
Slice Registers	53	0	126800	0.04
Register as Flip Flop	53	0	126800	0.04
Register as Latch	0	0	126800	0.00
F7 Muxes	0	0	31700	0.00
F8 Muxes	0	0	15850	0.00

* Warning! The Final LUT count, after physical optimizations and full implementation, is typically lower. Run opt_design after synthesis, if not already completed, for a more realistic count.

1.1 Summary of Registers by Type

Total	Clock Enable	Synchronous	Asynchronous
0	_	-	-
0	_	-	Set
0	_	-	Reset
0	_	Set	-
0	_	Reset	-
0	Yes	-	-
0	Yes	-	Set
0	Yes	-	Reset
14	Yes	Set	-
39	Yes	Reset	-

2. Memory

Site Type	Used	Fixed	Available	Util%
Block RAM Tile	0	0	135	0.00
RAMB36/FIFO*	0	0	135	0.00
RAMB18	0	0	270	0.00

* Note: Each Block RAM Tile only has one FIFO logic available and therefore can accommodate only one FIFO36E1 or one FIFO18E1. However, if a FIFO18E1 occupies a Block RAM Tile, that tile can still accommodate a RAMB18E1

3. DSP

Site Type	Used	Fixed	Available	Util%
DSPs	0	0	240	0.00

4. IO and GT Specific

Site Type	Used	Fixed	Available	Util%
Bonded IOB	16	0	210	7.62
Bonded IPADs	0	0	2	0.00
PHY_CONTROL	0	0	6	0.00
PHASER_REF	0	0	6	0.00
OUT_FIFO	0	0	24	0.00
IN_FIFO	0	0	24	0.00
IDELAYCTRL	0	0	6	0.00
IBUFDS	0	0	202	0.00
PHASER_OUT/PHASER_OUT_PHY	0	0	24	0.00
PHASER_IN/PHASER_IN_PHY	0	0	24	0.00
IDELAYE2/IDELAYE2_FINEDELAY	0	0	300	0.00
ILOGIC	0	0	210	0.00
OLOGIC	0	0	210	0.00

5. Clocking

Site Type	Used	Fixed	Available	Util%
BUFGCTRL	2	0	32	6.25
BUFIO	0	0	24	0.00
MMCME2_ADV	0	0	6	0.00
PLLE2_ADV	0	0	6	0.00
BUFMRCE	0	0	12	0.00
BUFHCE	0	0	96	0.00
BUFR	0	0	24	0.00

6. Specific Feature

Site Type	Used	Fixed	Available	Util%
BSCANE2	0	0	4	0.00
CAPTUREE2	0	0	1	0.00
DNA_PORT	0	0	1	0.00
EFUSE_USR	0	0	1	0.00
FRAME_ECCE2	0	0	1	0.00
ICAPE2	0	0	2	0.00
PCIE_2_1	0	0	1	0.00
STARTUPE2	0	0	1	0.00
XADC	0	0	1	0.00

7. Primitives

Ref Name	Used	Functional Category
LUT2	65	LUT
LUT6	64	LUT
LUT4	49	LUT
LUT3	48	LUT
FDRE	39	Flop & Latch
CARRY4	38	CarryLogic
LUT5	25	LUT
LUT1	18	LUT
OBUF	14	IO
FDSE	14	Flop & Latch
IBUF	2	IO
BUFG	2	Clock

8. Black Boxes

Ref Name	Used
blk_mem_gen_1	1

9. Instantiated Netlists

Ref Name	Used

v

5.6 下载结果

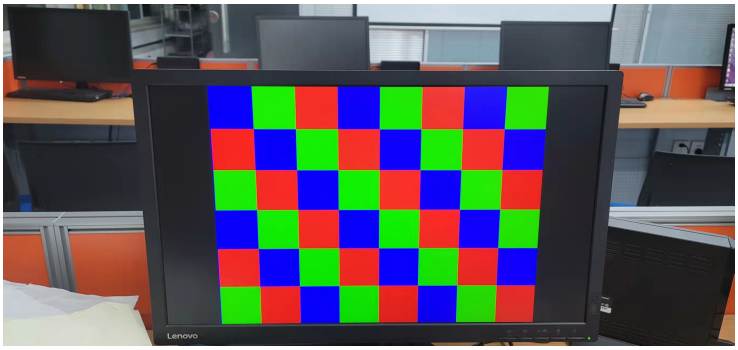

图 8: 下载结果

6 实现 PCU 绘画

6.1 逻辑设计

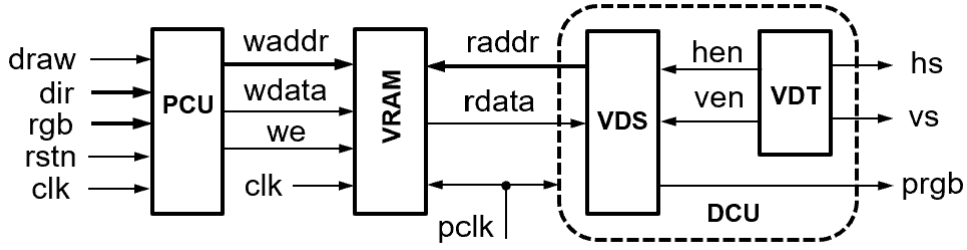

图 9: PNT 数据通路

6.2 设计文件

6.2.1 取边沿

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 13.11.2021 14:55:17
```

```
// Design Name:
// Module Name: PS
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////


module PS(input a,
          clk,
          output reg p);
    reg r,s,in_delay;
    always @(posedge clk) begin
        if (a)r <= 1;
        else r  <= 0;
    end
    always @(posedge clk) begin
        if (r)s <= 1;
        else s  <= 0;
    end
    always @(posedge clk) begin
        in_delay <= s;
    end
    always @(*) begin
        if (s&&!in_delay)
            p <= 1;
        else
            p <= 0;
    end
endmodule
```

6.2.2 去抖动

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 13.11.2021 15:41:51
// Design Name:
// Module Name: DB
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////


module DB(input x,
          clk,
          output reg y);
    reg [15:0]cnt;
    initial begin
        cnt = 0;
    end
    always @(*) begin
        if (cnt == 50000)
            y = x;
            end
            always @(posedge clk) begin
                if (cnt == 50000)
                    cnt        <= 0;
                    else  cnt <= cnt + 1;
                    end
                    endmodule
```

6.2.3 绘画控制单元

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 26.11.2021 17:29:42
// Design Name:
// Module Name: PCU
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////


module PCU(input clk,
           rstn,
           draw,
           [3:0]dir,
           [11:0]rgb,
           output reg [14:0]waddr,
           output [11:0]wdata,
           output we);
    assign we    = draw;
    assign wdata = rgb;
    always @(posedge clk) begin
        if (!rstn)
            waddr = 15100;
        else if (dir[0]&&waddr>199)
            waddr = waddr-200;
        else if (dir[1]&&waddr<29800)
            waddr = waddr+200;
        else if (dir[2]&&waddr%200 != 0)
            waddr = waddr-1;
        else if (dir[3]&&waddr%200 != 199)
            waddr = waddr+1;
            end
            endmodule
```

6.2.4 视频存储器

```
-- (c) Copyright 1995-2021 Xilinx, Inc. All rights reserved.
--
-- This file contains confidential and proprietary information
-- of Xilinx, Inc. and is protected under U.S. and
-- international copyright and other intellectual property
-- laws.
--
-- DISCLAIMER
-- This disclaimer is not a license and does not grant any
-- rights to the materials distributed herewith. Except as
-- otherwise provided in a valid license issued to you by
-- Xilinx, and to the maximum extent permitted by applicable
-- law: (1) THESE MATERIALS ARE MADE AVAILABLE "AS IS" AND
-- WITH ALL FAULTS, AND XILINX HEREBY DISCLAIMS ALL WARRANTIES
-- AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING
-- BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-
-- INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and
-- (2) Xilinx shall not be liable (whether in contract or tort,
-- including negligence, or under any other theory of
-- liability) for any loss or damage of any kind or nature
-- related to, arising under or in connection with these
-- materials, including for any direct, or any indirect,
-- special, incidental, or consequential loss or damage
-- (including loss of data, profits, goodwill, or any type of
-- loss or damage suffered as a result of any action brought
-- by a third party) even if such damage or loss was
-- reasonably foreseeable or Xilinx had been advised of the
-- possibility of the same.
--
-- CRITICAL APPLICATIONS
-- Xilinx products are not designed or intended to be fail-
```

```
-- safe, or for use in any application requiring fail-safe
-- performance, such as life-support or safety devices or
-- systems, Class III medical devices, nuclear facilities,
-- applications related to the deployment of airbags, or any
-- other applications that could lead to death, personal
-- injury, or severe property or environmental damage
-- (individually and collectively, "Critical
-- Applications"). Customer assumes the sole risk and
-- liability of any use of Xilinx products in Critical
-- Applications, subject only to applicable laws and
-- regulations governing limitations on product liability.
--
-- THIS COPYRIGHT NOTICE AND DISCLAIMER MUST BE RETAINED AS
-- PART OF THIS FILE AT ALL TIMES.
--
-- DO NOT MODIFY THIS FILE.

-- IP VLNV: xilinx.com:ip:blk_mem_gen:8.4
-- IP Revision: 3

LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
USE ieee.numeric_std.ALL;

LIBRARY blk_mem_gen_v8_4_3;
USE blk_mem_gen_v8_4_3.blk_mem_gen_v8_4_3;

ENTITY blk_mem_gen_0 IS
  PORT (
    clka : IN STD_LOGIC;
    ena : IN STD_LOGIC;
    wea : IN STD_LOGIC_VECTOR(0 DOWNTO 0);
    addra : IN STD_LOGIC_VECTOR(14 DOWNTO 0);
    dina : IN STD_LOGIC_VECTOR(11 DOWNTO 0);
    clkb : IN STD_LOGIC;
    addrb : IN STD_LOGIC_VECTOR(14 DOWNTO 0);
    doutb : OUT STD_LOGIC_VECTOR(11 DOWNTO 0)
  );
END blk_mem_gen_0;

ARCHITECTURE blk_mem_gen_0_arch OF blk_mem_gen_0 IS
  ATTRIBUTE DowngradeIPIdentifiedWarnings : STRING;
  ATTRIBUTE DowngradeIPIdentifiedWarnings OF blk_mem_gen_0_arch: ARCHITECTURE IS "yes";
  COMPONENT blk_mem_gen_v8_4_3 IS
    GENERIC (
      C_FAMILY : STRING;
      C_XDEVICEFAMILY : STRING;
      C_ELABORATION_DIR : STRING;
      C_INTERFACE_TYPE : INTEGER;
      C_AXI_TYPE : INTEGER;
      C_AXI_SLAVE_TYPE : INTEGER;
      C_USE_BRAM_BLOCK : INTEGER;
      C_ENABLE_32BIT_ADDRESS : INTEGER;
      C_CTRL_ECC_ALGO : STRING;
      C_HAS_AXI_ID : INTEGER;
      C_AXI_ID_WIDTH : INTEGER;
      C_MEM_TYPE : INTEGER;
      C_BYTE_SIZE : INTEGER;
      C_ALGORITHM : INTEGER;
      C_PRIM_TYPE : INTEGER;
      C_LOAD_INIT_FILE : INTEGER;
      C_INIT_FILE_NAME : STRING;
      C_INIT_FILE : STRING;
      C_USE_DEFAULT_DATA : INTEGER;
      C_DEFAULT_DATA : STRING;
      C_HAS_RSTA : INTEGER;
      C_RST_PRIORITY_A : STRING;
      C_RSTRAM_A : INTEGER;
      C_INITA_VAL : STRING;
      C_HAS_ENA : INTEGER;
      C_HAS_REGCEA : INTEGER;
      C_USE_BYTE_WEA : INTEGER;
      C_WEA_WIDTH : INTEGER;
      C_WRITE_MODE_A : STRING;
      C_WRITE_WIDTH_A : INTEGER;
      C_READ_WIDTH_A : INTEGER;
      C_WRITE_DEPTH_A : INTEGER;
      C_READ_DEPTH_A : INTEGER;
      C_ADDRA_WIDTH : INTEGER;
      C_HAS_RSTB : INTEGER;
      C_RST_PRIORITY_B : STRING;
      C_RSTRAM_B : INTEGER;
```

```
    C_INITB_VAL : STRING;
    C_HAS_ENB : INTEGER;
    C_HAS_REGCEB : INTEGER;
    C_USE_BYTE_WEB : INTEGER;
    C_WEB_WIDTH : INTEGER;
    C_WRITE_MODE_B : STRING;
    C_WRITE_WIDTH_B : INTEGER;
    C_READ_WIDTH_B : INTEGER;
    C_WRITE_DEPTH_B : INTEGER;
    C_READ_DEPTH_B : INTEGER;
    C_ADDRB_WIDTH : INTEGER;
    C_HAS_MEM_OUTPUT_REGS_A : INTEGER;
    C_HAS_MEM_OUTPUT_REGS_B : INTEGER;
    C_HAS_MUX_OUTPUT_REGS_A : INTEGER;
    C_HAS_MUX_OUTPUT_REGS_B : INTEGER;
    C_MUX_PIPELINE_STAGES : INTEGER;
    C_HAS_SOFTECC_INPUT_REGS_A : INTEGER;
    C_HAS_SOFTECC_OUTPUT_REGS_B : INTEGER;
    C_USE_SOFTECC : INTEGER;
    C_USE_ECC : INTEGER;
    C_EN_ECC_PIPE : INTEGER;
    C_READ_LATENCY_A : INTEGER;
    C_READ_LATENCY_B : INTEGER;
    C_HAS_INJECTERR : INTEGER;
    C_SIM_COLLISION_CHECK : STRING;
    C_COMMON_CLK : INTEGER;
    C_DISABLE_WARN_BHV_COLL : INTEGER;
    C_EN_SLEEP_PIN : INTEGER;
    C_USE_URAM : INTEGER;
    C_EN_RDADDRA_CHG : INTEGER;
    C_EN_RDADDRB_CHG : INTEGER;
    C_EN_DEEPSLEEP_PIN : INTEGER;
    C_EN_SHUTDOWN_PIN : INTEGER;
    C_EN_SAFETY_CKT : INTEGER;
    C_DISABLE_WARN_BHV_RANGE : INTEGER;
    C_COUNT_36K_BRAM : STRING;
    C_COUNT_18K_BRAM : STRING;
    C_EST_POWER_SUMMARY : STRING
  );
  PORT (
    clka : IN STD_LOGIC;
    rsta : IN STD_LOGIC;
    ena : IN STD_LOGIC;
    regcea : IN STD_LOGIC;
    wea : IN STD_LOGIC_VECTOR(0 DOWNTO 0);
    addra : IN STD_LOGIC_VECTOR(14 DOWNTO 0);
    dina : IN STD_LOGIC_VECTOR(11 DOWNTO 0);
    douta : OUT STD_LOGIC_VECTOR(11 DOWNTO 0);
    clkb : IN STD_LOGIC;
    rstb : IN STD_LOGIC;
    enb : IN STD_LOGIC;
    regceb : IN STD_LOGIC;
    web : IN STD_LOGIC_VECTOR(0 DOWNTO 0);
    addrb : IN STD_LOGIC_VECTOR(14 DOWNTO 0);
    dinb : IN STD_LOGIC_VECTOR(11 DOWNTO 0);
    doutb : OUT STD_LOGIC_VECTOR(11 DOWNTO 0);
    injectsbiterr : IN STD_LOGIC;
    injectdbiterr : IN STD_LOGIC;
    eccpipece : IN STD_LOGIC;
    sbiterr : OUT STD_LOGIC;
    dbiterr : OUT STD_LOGIC;
    rdaddrecc : OUT STD_LOGIC_VECTOR(14 DOWNTO 0);
    sleep : IN STD_LOGIC;
    deepsleep : IN STD_LOGIC;
    shutdown : IN STD_LOGIC;
    rsta_busy : OUT STD_LOGIC;
    rstb_busy : OUT STD_LOGIC;
    s_aclk : IN STD_LOGIC;
    s_aresetn : IN STD_LOGIC;
    s_axi_awid : IN STD_LOGIC_VECTOR(3 DOWNTO 0);
    s_axi_awaddr : IN STD_LOGIC_VECTOR(31 DOWNTO 0);
    s_axi_awlen : IN STD_LOGIC_VECTOR(7 DOWNTO 0);
    s_axi_awsize : IN STD_LOGIC_VECTOR(2 DOWNTO 0);
    s_axi_awburst : IN STD_LOGIC_VECTOR(1 DOWNTO 0);
    s_axi_awvalid : IN STD_LOGIC;
    s_axi_awready : OUT STD_LOGIC;
    s_axi_wdata : IN STD_LOGIC_VECTOR(11 DOWNTO 0);
    s_axi_wstrb : IN STD_LOGIC_VECTOR(0 DOWNTO 0);
    s_axi_wlast : IN STD_LOGIC;
    s_axi_wvalid : IN STD_LOGIC;
    s_axi_wready : OUT STD_LOGIC;
    s_axi_bid : OUT STD_LOGIC_VECTOR(3 DOWNTO 0);
```

```
      s_axi_bresp : OUT STD_LOGIC_VECTOR(1 DOWNTO 0);
      s_axi_bvalid : OUT STD_LOGIC;
      s_axi_bready : IN STD_LOGIC;
      s_axi_arid : IN STD_LOGIC_VECTOR(3 DOWNTO 0);
      s_axi_araddr : IN STD_LOGIC_VECTOR(31 DOWNTO 0);
      s_axi_arlen : IN STD_LOGIC_VECTOR(7 DOWNTO 0);
      s_axi_arsize : IN STD_LOGIC_VECTOR(2 DOWNTO 0);
      s_axi_arburst : IN STD_LOGIC_VECTOR(1 DOWNTO 0);
      s_axi_arvalid : IN STD_LOGIC;
      s_axi_arready : OUT STD_LOGIC;
      s_axi_rid : OUT STD_LOGIC_VECTOR(3 DOWNTO 0);
      s_axi_rdata : OUT STD_LOGIC_VECTOR(11 DOWNTO 0);
      s_axi_rresp : OUT STD_LOGIC_VECTOR(1 DOWNTO 0);
      s_axi_rlast : OUT STD_LOGIC;
      s_axi_rvalid : OUT STD_LOGIC;
      s_axi_rready : IN STD_LOGIC;
      s_axi_injectsbiterr : IN STD_LOGIC;
      s_axi_injectdbiterr : IN STD_LOGIC;
      s_axi_sbiterr : OUT STD_LOGIC;
      s_axi_dbiterr : OUT STD_LOGIC;
      s_axi_rdaddrecc : OUT STD_LOGIC_VECTOR(14 DOWNTO 0)
    );
  END COMPONENT blk_mem_gen_v8_4_3;
  ATTRIBUTE X_CORE_INFO : STRING;
  ATTRIBUTE X_CORE_INFO OF blk_mem_gen_0_arch: ARCHITECTURE IS "blk_mem_gen_v8_4_3,Vivado 2019.1";
  ATTRIBUTE CHECK_LICENSE_TYPE : STRING;
  ATTRIBUTE CHECK_LICENSE_TYPE OF blk_mem_gen_0_arch : ARCHITECTURE IS "blk_mem_gen_0,blk_mem_gen_v8_4_3,{}";
  ATTRIBUTE CORE_GENERATION_INFO : STRING;
  ATTRIBUTE CORE_GENERATION_INFO OF blk_mem_gen_0_arch: ARCHITECTURE IS "blk_mem_gen_0,blk_mem_gen_v8_4_3,{x_ipProduct=Vivado 2019.1,x_ipVendor=xilinx
"e_file_loaded,C_INIT_FILE=blk_mem_gen_0.mem,C_USE_DEFAULT_DATA=1,C_DEFAULT_DATA=fff,C_HAS_RSTA=0,C_RST_PRIORITY_A=CE,C_RSTRAM_A=0,C_INITA_VAL=0,C_HAS_
"_B=12,C_WRITE_DEPTH_B=32000,C_READ_DEPTH_B=32000,C_ADDRB_WIDTH=15,C_HAS_MEM_OUTPUT_REGS_A=0,C_HAS_MEM_OUTPUT_REGS_B=0,C_HAS_MUX_OUTPUT_REGS_A=0,C_HAS_
"EP_PIN=0,C_EN_SHUTDOWN_PIN=0,C_EN_SAFETY_CKT=0,C_DISABLE_WARN_BHV_RANGE=0,C_COUNT_36K_BRAM=11,C_COUNT_18K_BRAM=0,C_EST_POWER_SUMMARY=Estimated Power f
_     17.470026 mW}";
  ATTRIBUTE X_INTERFACE_INFO : STRING;
  ATTRIBUTE X_INTERFACE_PARAMETER : STRING;
  ATTRIBUTE X_INTERFACE_INFO OF doutb: SIGNAL IS "xilinx.com:interface:bram:1.0 BRAM_PORTB DOUT";
  ATTRIBUTE X_INTERFACE_INFO OF addrb: SIGNAL IS "xilinx.com:interface:bram:1.0 BRAM_PORTB ADDR";
  ATTRIBUTE X_INTERFACE_PARAMETER OF clkb: SIGNAL IS "XIL_INTERFACENAME BRAM_PORTB, MEM_SIZE 8192, MEM_WIDTH 32, MEM_ECC NONE, MASTER_TYPE OTHER, READ_
  ATTRIBUTE X_INTERFACE_INFO OF clkb: SIGNAL IS "xilinx.com:interface:bram:1.0 BRAM_PORTB CLK";
  ATTRIBUTE X_INTERFACE_INFO OF dina: SIGNAL IS "xilinx.com:interface:bram:1.0 BRAM_PORTA DIN";
  ATTRIBUTE X_INTERFACE_INFO OF addra: SIGNAL IS "xilinx.com:interface:bram:1.0 BRAM_PORTA ADDR";
  ATTRIBUTE X_INTERFACE_INFO OF wea: SIGNAL IS "xilinx.com:interface:bram:1.0 BRAM_PORTA WE";
  ATTRIBUTE X_INTERFACE_INFO OF ena: SIGNAL IS "xilinx.com:interface:bram:1.0 BRAM_PORTA EN";
  ATTRIBUTE X_INTERFACE_PARAMETER OF clka: SIGNAL IS "XIL_INTERFACENAME BRAM_PORTA, MEM_SIZE 8192, MEM_WIDTH 32, MEM_ECC NONE, MASTER_TYPE OTHER, READ_
  ATTRIBUTE X_INTERFACE_INFO OF clka: SIGNAL IS "xilinx.com:interface:bram:1.0 BRAM_PORTA CLK";
BEGIN
  U0 : blk_mem_gen_v8_4_3
    GENERIC MAP (
      C_FAMILY => "artix7",
      C_XDEVICEFAMILY => "artix7",
      C_ELABORATION_DIR => "./",
      C_INTERFACE_TYPE => 0,
      C_AXI_TYPE => 1,
      C_AXI_SLAVE_TYPE => 0,
      C_USE_BRAM_BLOCK => 0,
      C_ENABLE_32BIT_ADDRESS => 0,
      C_CTRL_ECC_ALGO => "NONE",
      C_HAS_AXI_ID => 0,
      C_AXI_ID_WIDTH => 4,
      C_MEM_TYPE => 1,
      C_BYTE_SIZE => 9,
      C_ALGORITHM => 1,
      C_PRIM_TYPE => 1,
      C_LOAD_INIT_FILE => 0,
      C_INIT_FILE_NAME => "no_coe_file_loaded",
      C_INIT_FILE => "blk_mem_gen_0.mem",
      C_USE_DEFAULT_DATA => 1,
      C_DEFAULT_DATA => "fff",
      C_HAS_RSTA => 0,
      C_RST_PRIORITY_A => "CE",
      C_RSTRAM_A => 0,
      C_INITA_VAL => "0",
      C_HAS_ENA => 1,
      C_HAS_REGCEA => 0,
      C_USE_BYTE_WEA => 0,
      C_WEA_WIDTH => 1,
      C_WRITE_MODE_A => "NO_CHANGE",
      C_WRITE_WIDTH_A => 12,
      C_READ_WIDTH_A => 12,
      C_WRITE_DEPTH_A => 32000,
      C_READ_DEPTH_A => 32000,
      C_ADDRA_WIDTH => 15,
```

```
    C_HAS_RSTB => 0,
    C_RST_PRIORITY_B => "CE",
    C_RSTRAM_B => 0,
    C_INITB_VAL => "0",
    C_HAS_ENB => 0,
    C_HAS_REGCEB => 0,
    C_USE_BYTE_WEB => 0,
    C_WEB_WIDTH => 1,
    C_WRITE_MODE_B => "WRITE_FIRST",
    C_WRITE_WIDTH_B => 12,
    C_READ_WIDTH_B => 12,
    C_WRITE_DEPTH_B => 32000,
    C_READ_DEPTH_B => 32000,
    C_ADDRB_WIDTH => 15,
    C_HAS_MEM_OUTPUT_REGS_A => 0,
    C_HAS_MEM_OUTPUT_REGS_B => 0,
    C_HAS_MUX_OUTPUT_REGS_A => 0,
    C_HAS_MUX_OUTPUT_REGS_B => 0,
    C_MUX_PIPELINE_STAGES => 0,
    C_HAS_SOFTECC_INPUT_REGS_A => 0,
    C_HAS_SOFTECC_OUTPUT_REGS_B => 0,
    C_USE_SOFTECC => 0,
    C_USE_ECC => 0,
    C_EN_ECC_PIPE => 0,
    C_READ_LATENCY_A => 1,
    C_READ_LATENCY_B => 1,
    C_HAS_INJECTERR => 0,
    C_SIM_COLLISION_CHECK => "ALL",
    C_COMMON_CLK => 0,
    C_DISABLE_WARN_BHV_COLL => 0,
    C_EN_SLEEP_PIN => 0,
    C_USE_URAM => 0,
    C_EN_RDADDRA_CHG => 0,
    C_EN_RDADDRB_CHG => 0,
    C_EN_DEEPSLEEP_PIN => 0,
    C_EN_SHUTDOWN_PIN => 0,
    C_EN_SAFETY_CKT => 0,
    C_DISABLE_WARN_BHV_RANGE => 0,
    C_COUNT_36K_BRAM => "11",
    C_COUNT_18K_BRAM => "0",
    C_EST_POWER_SUMMARY => "Estimated Power for IP     :     17.470026 mW"
  )
  PORT MAP (
    clka => clka,
    rsta => '0',
    ena => ena,
    regcea => '0',
    wea => wea,
    addra => addra,
    dina => dina,
    clkb => clkb,
    rstb => '0',
    enb => '0',
    regceb => '0',
    web => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 1)),
    addrb => addrb,
    dinb => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 12)),
    doutb => doutb,
    injectsbiterr => '0',
    injectdbiterr => '0',
    eccpipece => '0',
    sleep => '0',
    deepsleep => '0',
    shutdown => '0',
    s_aclk => '0',
    s_aresetn => '0',
    s_axi_awid => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 4)),
    s_axi_awaddr => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 32)),
    s_axi_awlen => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 8)),
    s_axi_awsize => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 3)),
    s_axi_awburst => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 2)),
    s_axi_awvalid => '0',
    s_axi_wdata => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 12)),
    s_axi_wstrb => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 1)),
    s_axi_wlast => '0',
    s_axi_wvalid => '0',
    s_axi_bready => '0',
    s_axi_arid => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 4)),
    s_axi_araddr => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 32)),
    s_axi_arlen => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 8)),
    s_axi_arsize => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 3)),
    s_axi_arburst => STD_LOGIC_VECTOR(TO_UNSIGNED(0, 2)),
```

```
        s_axi_arvalid => '0',
        s_axi_rready => '0',
        s_axi_injectsbiterr => '0',
        s_axi_injectdbiterr => '0'
      );
END blk_mem_gen_0_arch;
```

6.2.5 绘画模块

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 28.11.2021 10:05:44
// Design Name:
// Module Name: PNT
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////


module PNT(input clk,
           rstn,
           input up,
           input down,
           input left,
           input right,
           input [3:0] red,
           input [3:0] green,
           input [3:0] blue,
           input draw,
           output [3:0] pred,
           output [3:0] pgreen,
           output [3:0] pblue,
           output hs,
           output vs);
    wire[14:0]waddr,raddr;
    wire[11:0]wdata,rdata;
    wire we,pclk,hen,ven,up_tb,down_tb,left_tb,right_tb,up_ps,down_ps,left_ps,right_ps;
    DB db0(up,clk,up_tb);
    PS ps0(up_tb,clk,up_ps);
    DB db1(down,clk,down_tb);
    PS ps1(down_tb,clk,down_ps);
    DB db2(left,clk,left_tb);
    PS ps2(left_tb,clk,left_ps);
    DB db3(right,clk,right_tb);
    PS ps3(right_tb,clk,right_ps);
    DIR dir(clk,pclk);
    PCU pcu(clk,rstn,draw,{right_ps,left_ps,down_ps,up_ps},{red,green,blue},waddr,wdata,we);
    blk_mem_gen_0 vram(.addra(waddr),.clka(clk),.dina(wdata),.addrb(raddr),.clkb(pclk),.ena(we),.wea(we),.doutb(rdata));
    VDS vds(hen,ven,pclk,rstn,rdata,raddr,{pred,pgreen,pblue});
    VDT vdt(pclk,rstn,hen,ven,hs,vs);
endmodule
```

6.3 RTL 分析

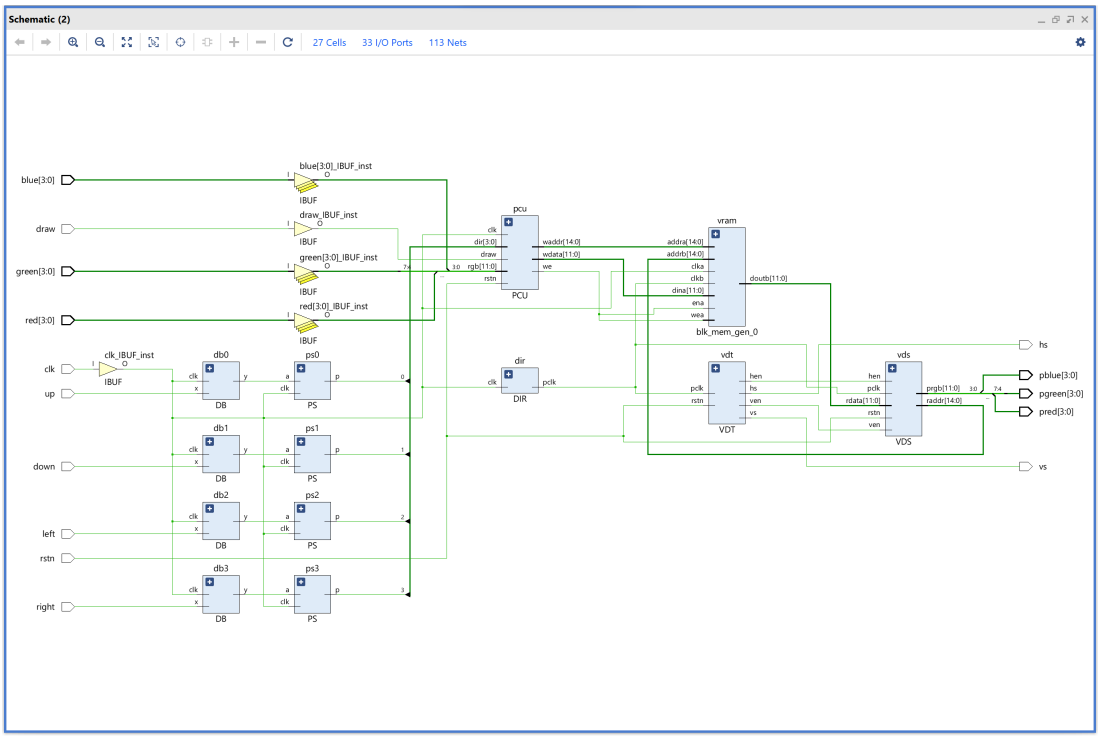

图 10: RTL 分析电路图

6.4 约束文件

```
## This file is a general .xdc for the Nexys4 DDR Rev. C
## To use it in a project:
## - uncomment the lines corresponding to used pins
## - rename the used ports (in each line, after get_ports) according to the top level signal names in the project

## Clock signal
set_property -dict { PACKAGE_PIN E3    IOSTANDARD LVCMOS33 } [get_ports { clk }]; #IO_L12P_T1_MRCC_35 Sch=clk100mhz
create_clock -add -name sys_clk_pin -period 10.00 -waveform {0 5} [get_ports {clk}];


##Switches

set_property -dict { PACKAGE_PIN J15   IOSTANDARD LVCMOS33 } [get_ports { red[0] }]; #IO_L24N_T3_RS0_15 Sch=sw[0]
set_property -dict { PACKAGE_PIN L16   IOSTANDARD LVCMOS33 } [get_ports { red[1] }]; #IO_L3N_T0_DQS_EMCCLK_14 Sch=sw[1]
set_property -dict { PACKAGE_PIN M13   IOSTANDARD LVCMOS33 } [get_ports { red[2] }]; #IO_L6N_T0_D08_VREF_14 Sch=sw[2]
set_property -dict { PACKAGE_PIN R15   IOSTANDARD LVCMOS33 } [get_ports { red[3] }]; #IO_L13N_T2_MRCC_14 Sch=sw[3]
set_property -dict { PACKAGE_PIN R17   IOSTANDARD LVCMOS33 } [get_ports { green[0] }]; #IO_L12N_T1_MRCC_14 Sch=sw[4]
set_property -dict { PACKAGE_PIN T18   IOSTANDARD LVCMOS33 } [get_ports { green[1] }]; #IO_L7N_T1_D10_14 Sch=sw[5]
set_property -dict { PACKAGE_PIN U18   IOSTANDARD LVCMOS33 } [get_ports { green[2] }]; #IO_L17N_T2_A13_D29_14 Sch=sw[6]
set_property -dict { PACKAGE_PIN R13   IOSTANDARD LVCMOS33 } [get_ports { green[3] }]; #IO_L5N_T0_D07_14 Sch=sw[7]
set_property -dict { PACKAGE_PIN T8    IOSTANDARD LVCMOS18 } [get_ports { blue[0] }]; #IO_L24N_T3_34 Sch=sw[8]
set_property -dict { PACKAGE_PIN U8    IOSTANDARD LVCMOS18 } [get_ports { blue[1] }]; #IO_25_34 Sch=sw[9]
set_property -dict { PACKAGE_PIN R16   IOSTANDARD LVCMOS33 } [get_ports { blue[2] }]; #IO_L15P_T2_DQS_RDWR_B_14 Sch=sw[10]
set_property -dict { PACKAGE_PIN T13   IOSTANDARD LVCMOS33 } [get_ports { blue[3] }]; #IO_L23P_T3_A03_D19_14 Sch=sw[11]
#set_property -dict { PACKAGE_PIN H6    IOSTANDARD LVCMOS33 } [get_ports { SW[12] }]; #IO_L24P_T3_35 Sch=sw[12]
#set_property -dict { PACKAGE_PIN U12   IOSTANDARD LVCMOS33 } [get_ports { SW[13] }]; #IO_L20P_T3_A08_D24_14 Sch=sw[13]
#set_property -dict { PACKAGE_PIN U11   IOSTANDARD LVCMOS33 } [get_ports { SW[14] }]; #IO_L19N_T3_A09_D25_VREF_14 Sch=sw[14]
set_property -dict { PACKAGE_PIN V10   IOSTANDARD LVCMOS33 } [get_ports { draw }]; #IO_L21P_T3_DQS_14 Sch=sw[15]


## LEDs

#set_property -dict { PACKAGE_PIN H17   IOSTANDARD LVCMOS33 } [get_ports { LED[0] }]; #IO_L18P_T2_A24_15 Sch=led[0]
#set_property -dict { PACKAGE_PIN K15   IOSTANDARD LVCMOS33 } [get_ports { LED[1] }]; #IO_L24P_T3_RS1_15 Sch=led[1]
#set_property -dict { PACKAGE_PIN J13   IOSTANDARD LVCMOS33 } [get_ports { LED[2] }]; #IO_L17N_T2_A25_15 Sch=led[2]
#set_property -dict { PACKAGE_PIN N14   IOSTANDARD LVCMOS33 } [get_ports { LED[3] }]; #IO_L8P_T1_D11_14 Sch=led[3]
#set_property -dict { PACKAGE_PIN R18   IOSTANDARD LVCMOS33 } [get_ports { LED[4] }]; #IO_L7P_T1_D09_14 Sch=led[4]
#set_property -dict { PACKAGE_PIN V17   IOSTANDARD LVCMOS33 } [get_ports { LED[5] }]; #IO_L18N_T2_A11_D27_14 Sch=led[5]
#set_property -dict { PACKAGE_PIN U17   IOSTANDARD LVCMOS33 } [get_ports { LED[6] }]; #IO_L17P_T2_A14_D30_14 Sch=led[6]
```

```
#set_property -dict { PACKAGE_PIN U16   IOSTANDARD LVCMOS33 } [get_ports { LED[7] }]; #IO_L18P_T2_A12_D28_14 Sch=led[7]
#set_property -dict { PACKAGE_PIN V16   IOSTANDARD LVCMOS33 } [get_ports { LED[8] }]; #IO_L16N_T2_A15_D31_14 Sch=led[8]
#set_property -dict { PACKAGE_PIN T15   IOSTANDARD LVCMOS33 } [get_ports { LED[9] }]; #IO_L14N_T2_SRCC_14 Sch=led[9]
#set_property -dict { PACKAGE_PIN U14   IOSTANDARD LVCMOS33 } [get_ports { LED[10] }]; #IO_L22P_T3_A05_D21_14 Sch=led[10]
#set_property -dict { PACKAGE_PIN T16   IOSTANDARD LVCMOS33 } [get_ports { LED[11] }]; #IO_L15N_T2_DQS_DOUT_CSO_B_14 Sch=led[11]
#set_property -dict { PACKAGE_PIN V15   IOSTANDARD LVCMOS33 } [get_ports { LED[12] }]; #IO_L16P_T2_CSI_B_14 Sch=led[12]
#set_property -dict { PACKAGE_PIN V14   IOSTANDARD LVCMOS33 } [get_ports { LED[13] }]; #IO_L22N_T3_A04_D20_14 Sch=led[13]
#set_property -dict { PACKAGE_PIN V12   IOSTANDARD LVCMOS33 } [get_ports { LED[14] }]; #IO_L20N_T3_A07_D23_14 Sch=led[14]
#set_property -dict { PACKAGE_PIN V11   IOSTANDARD LVCMOS33 } [get_ports { LED[15] }]; #IO_L21N_T3_DQS_A06_D22_14 Sch=led[15]

#set_property -dict { PACKAGE_PIN R12   IOSTANDARD LVCMOS33 } [get_ports { LED16_B }]; #IO_L5P_T0_D06_14 Sch=led16_b
#set_property -dict { PACKAGE_PIN M16   IOSTANDARD LVCMOS33 } [get_ports { LED16_G }]; #IO_L10P_T1_D14_14 Sch=led16_g
#set_property -dict { PACKAGE_PIN N15   IOSTANDARD LVCMOS33 } [get_ports { LED16_R }]; #IO_L11P_T1_SRCC_14 Sch=led16_r
#set_property -dict { PACKAGE_PIN G14   IOSTANDARD LVCMOS33 } [get_ports { LED17_B }]; #IO_L15N_T2_DQS_ADV_B_15 Sch=led17_b
#set_property -dict { PACKAGE_PIN R11   IOSTANDARD LVCMOS33 } [get_ports { LED17_G }]; #IO_0_14 Sch=led17_g
#set_property -dict { PACKAGE_PIN N16   IOSTANDARD LVCMOS33 } [get_ports { LED17_R }]; #IO_L11N_T1_SRCC_14 Sch=led17_r


##7 segment display

#set_property -dict { PACKAGE_PIN T10   IOSTANDARD LVCMOS33 } [get_ports { CA }]; #IO_L24N_T3_A00_D16_14 Sch=ca
#set_property -dict { PACKAGE_PIN R10   IOSTANDARD LVCMOS33 } [get_ports { CB }]; #IO_25_14 Sch=cb
#set_property -dict { PACKAGE_PIN K16   IOSTANDARD LVCMOS33 } [get_ports { CC }]; #IO_25_15 Sch=cc
#set_property -dict { PACKAGE_PIN K13   IOSTANDARD LVCMOS33 } [get_ports { CD }]; #IO_L17P_T2_A26_15 Sch=cd
#set_property -dict { PACKAGE_PIN P15   IOSTANDARD LVCMOS33 } [get_ports { CE }]; #IO_L13P_T2_MRCC_14 Sch=ce
#set_property -dict { PACKAGE_PIN T11   IOSTANDARD LVCMOS33 } [get_ports { CF }]; #IO_L19P_T3_A10_D26_14 Sch=cf
#set_property -dict { PACKAGE_PIN L18   IOSTANDARD LVCMOS33 } [get_ports { CG }]; #IO_L4P_T0_D04_14 Sch=cg

#set_property -dict { PACKAGE_PIN H15   IOSTANDARD LVCMOS33 } [get_ports { DP }]; #IO_L19N_T3_A21_VREF_15 Sch=dp

#set_property -dict { PACKAGE_PIN J17   IOSTANDARD LVCMOS33 } [get_ports { AN[0] }]; #IO_L23P_T3_FOE_B_15 Sch=an[0]
#set_property -dict { PACKAGE_PIN J18   IOSTANDARD LVCMOS33 } [get_ports { AN[1] }]; #IO_L23N_T3_FWE_B_15 Sch=an[1]
#set_property -dict { PACKAGE_PIN T9    IOSTANDARD LVCMOS33 } [get_ports { AN[2] }]; #IO_L24P_T3_A01_D17_14 Sch=an[2]
#set_property -dict { PACKAGE_PIN J14   IOSTANDARD LVCMOS33 } [get_ports { AN[3] }]; #IO_L19P_T3_A22_15 Sch=an[3]
#set_property -dict { PACKAGE_PIN P14   IOSTANDARD LVCMOS33 } [get_ports { AN[4] }]; #IO_L8N_T1_D12_14 Sch=an[4]
#set_property -dict { PACKAGE_PIN T14   IOSTANDARD LVCMOS33 } [get_ports { AN[5] }]; #IO_L14P_T2_SRCC_14 Sch=an[5]
#set_property -dict { PACKAGE_PIN K2    IOSTANDARD LVCMOS33 } [get_ports { AN[6] }]; #IO_L23P_T3_35 Sch=an[6]
#set_property -dict { PACKAGE_PIN U13   IOSTANDARD LVCMOS33 } [get_ports { AN[7] }]; #IO_L23N_T3_A02_D18_14 Sch=an[7]


##Buttons

set_property -dict { PACKAGE_PIN C12   IOSTANDARD LVCMOS33 } [get_ports { rstn }]; #IO_L3P_T0_DQS_AD1P_15 Sch=cpu_resetn

#set_property -dict { PACKAGE_PIN N17   IOSTANDARD LVCMOS33 } [get_ports { BTNC }]; #IO_L9P_T1_DQS_14 Sch=btnc
set_property -dict { PACKAGE_PIN M18   IOSTANDARD LVCMOS33 } [get_ports { up }]; #IO_L4N_T0_D05_14 Sch=btnu
set_property -dict { PACKAGE_PIN P17   IOSTANDARD LVCMOS33 } [get_ports { left }]; #IO_L12P_T1_MRCC_14 Sch=btnl
set_property -dict { PACKAGE_PIN M17   IOSTANDARD LVCMOS33 } [get_ports { right }]; #IO_L10N_T1_D15_14 Sch=btnr
set_property -dict { PACKAGE_PIN P18   IOSTANDARD LVCMOS33 } [get_ports { down }]; #IO_L9N_T1_DQS_D13_14 Sch=btnd


##Pmod Headers


##Pmod Header JA

#set_property -dict { PACKAGE_PIN C17   IOSTANDARD LVCMOS33 } [get_ports { JA[1] }]; #IO_L20N_T3_A19_15 Sch=ja[1]
#set_property -dict { PACKAGE_PIN D18   IOSTANDARD LVCMOS33 } [get_ports { JA[2] }]; #IO_L21N_T3_DQS_A18_15 Sch=ja[2]
#set_property -dict { PACKAGE_PIN E18   IOSTANDARD LVCMOS33 } [get_ports { JA[3] }]; #IO_L21P_T3_DQS_15 Sch=ja[3]
#set_property -dict { PACKAGE_PIN G17   IOSTANDARD LVCMOS33 } [get_ports { JA[4] }]; #IO_L18N_T2_A23_15 Sch=ja[4]
#set_property -dict { PACKAGE_PIN D17   IOSTANDARD LVCMOS33 } [get_ports { JA[7] }]; #IO_L16N_T2_A27_15 Sch=ja[7]
#set_property -dict { PACKAGE_PIN E17   IOSTANDARD LVCMOS33 } [get_ports { JA[8] }]; #IO_L16P_T2_A28_15 Sch=ja[8]
#set_property -dict { PACKAGE_PIN F18   IOSTANDARD LVCMOS33 } [get_ports { JA[9] }]; #IO_L22N_T3_A16_15 Sch=ja[9]
#set_property -dict { PACKAGE_PIN G18   IOSTANDARD LVCMOS33 } [get_ports { JA[10] }]; #IO_L22P_T3_A17_15 Sch=ja[10]


##Pmod Header JB

#set_property -dict { PACKAGE_PIN D14   IOSTANDARD LVCMOS33 } [get_ports { JB[1] }]; #IO_L1P_T0_AD0P_15 Sch=jb[1]
#set_property -dict { PACKAGE_PIN F16   IOSTANDARD LVCMOS33 } [get_ports { JB[2] }]; #IO_L14N_T2_SRCC_15 Sch=jb[2]
#set_property -dict { PACKAGE_PIN G16   IOSTANDARD LVCMOS33 } [get_ports { JB[3] }]; #IO_L13N_T2_MRCC_15 Sch=jb[3]
#set_property -dict { PACKAGE_PIN H14   IOSTANDARD LVCMOS33 } [get_ports { JB[4] }]; #IO_L15P_T2_DQS_15 Sch=jb[4]
#set_property -dict { PACKAGE_PIN E16   IOSTANDARD LVCMOS33 } [get_ports { JB[7] }]; #IO_L11N_T1_SRCC_15 Sch=jb[7]
#set_property -dict { PACKAGE_PIN F13   IOSTANDARD LVCMOS33 } [get_ports { JB[8] }]; #IO_L5P_T0_AD9P_15 Sch=jb[8]
#set_property -dict { PACKAGE_PIN G13   IOSTANDARD LVCMOS33 } [get_ports { JB[9] }]; #IO_0_15 Sch=jb[9]
#set_property -dict { PACKAGE_PIN H16   IOSTANDARD LVCMOS33 } [get_ports { JB[10] }]; #IO_L13P_T2_MRCC_15 Sch=jb[10]


##Pmod Header JC

#set_property -dict { PACKAGE_PIN K1    IOSTANDARD LVCMOS33 } [get_ports { JC[1] }]; #IO_L23N_T3_35 Sch=jc[1]
#set_property -dict { PACKAGE_PIN F6    IOSTANDARD LVCMOS33 } [get_ports { JC[2] }]; #IO_L19N_T3_VREF_35 Sch=jc[2]
```

```
#set_property -dict { PACKAGE_PIN J2    IOSTANDARD LVCMOS33 } [get_ports { JC[3] }]; #IO_L22N_T3_35 Sch=jc[3]
#set_property -dict { PACKAGE_PIN G6    IOSTANDARD LVCMOS33 } [get_ports { JC[4] }]; #IO_L19P_T3_35 Sch=jc[4]
#set_property -dict { PACKAGE_PIN E7    IOSTANDARD LVCMOS33 } [get_ports { JC[7] }]; #IO_L6P_T0_35 Sch=jc[7]
#set_property -dict { PACKAGE_PIN J3    IOSTANDARD LVCMOS33 } [get_ports { JC[8] }]; #IO_L22P_T3_35 Sch=jc[8]
#set_property -dict { PACKAGE_PIN J4    IOSTANDARD LVCMOS33 } [get_ports { JC[9] }]; #IO_L21P_T3_DQS_35 Sch=jc[9]
#set_property -dict { PACKAGE_PIN E6    IOSTANDARD LVCMOS33 } [get_ports { JC[10] }]; #IO_L5P_T0_AD13P_35 Sch=jc[10]


##Pmod Header JD

#set_property -dict { PACKAGE_PIN H4    IOSTANDARD LVCMOS33 } [get_ports { JD[1] }]; #IO_L21N_T3_DQS_35 Sch=jd[1]
#set_property -dict { PACKAGE_PIN H1    IOSTANDARD LVCMOS33 } [get_ports { JD[2] }]; #IO_L17P_T2_35 Sch=jd[2]
#set_property -dict { PACKAGE_PIN G1    IOSTANDARD LVCMOS33 } [get_ports { JD[3] }]; #IO_L17N_T2_35 Sch=jd[3]
#set_property -dict { PACKAGE_PIN G3    IOSTANDARD LVCMOS33 } [get_ports { JD[4] }]; #IO_L20N_T3_35 Sch=jd[4]
#set_property -dict { PACKAGE_PIN H2    IOSTANDARD LVCMOS33 } [get_ports { JD[7] }]; #IO_L15P_T2_DQS_35 Sch=jd[7]
#set_property -dict { PACKAGE_PIN G4    IOSTANDARD LVCMOS33 } [get_ports { JD[8] }]; #IO_L20P_T3_35 Sch=jd[8]
#set_property -dict { PACKAGE_PIN G2    IOSTANDARD LVCMOS33 } [get_ports { JD[9] }]; #IO_L15N_T2_DQS_35 Sch=jd[9]
#set_property -dict { PACKAGE_PIN F3    IOSTANDARD LVCMOS33 } [get_ports { JD[10] }]; #IO_L13N_T2_MRCC_35 Sch=jd[10]


##Pmod Header JXADC

#set_property -dict { PACKAGE_PIN A14   IOSTANDARD LVDS     } [get_ports { XA_N[1] }]; #IO_L9N_T1_DQS_AD3N_15 Sch=xa_n[1]
#set_property -dict { PACKAGE_PIN A13   IOSTANDARD LVDS     } [get_ports { XA_P[1] }]; #IO_L9P_T1_DQS_AD3P_15 Sch=xa_p[1]
#set_property -dict { PACKAGE_PIN A16   IOSTANDARD LVDS     } [get_ports { XA_N[2] }]; #IO_L8N_T1_AD10N_15 Sch=xa_n[2]
#set_property -dict { PACKAGE_PIN A15   IOSTANDARD LVDS     } [get_ports { XA_P[2] }]; #IO_L8P_T1_AD10P_15 Sch=xa_p[2]
#set_property -dict { PACKAGE_PIN B17   IOSTANDARD LVDS     } [get_ports { XA_N[3] }]; #IO_L7N_T1_AD2N_15 Sch=xa_n[3]
#set_property -dict { PACKAGE_PIN B16   IOSTANDARD LVDS     } [get_ports { XA_P[3] }]; #IO_L7P_T1_AD2P_15 Sch=xa_p[3]
#set_property -dict { PACKAGE_PIN A18   IOSTANDARD LVDS     } [get_ports { XA_N[4] }]; #IO_L10N_T1_AD11N_15 Sch=xa_n[4]
#set_property -dict { PACKAGE_PIN B18   IOSTANDARD LVDS     } [get_ports { XA_P[4] }]; #IO_L10P_T1_AD11P_15 Sch=xa_p[4]


##VGA Connector

set_property -dict { PACKAGE_PIN A3    IOSTANDARD LVCMOS33 } [get_ports { pred[0] }]; #IO_L8N_T1_AD14N_35 Sch=vga_r[0]
set_property -dict { PACKAGE_PIN B4    IOSTANDARD LVCMOS33 } [get_ports { pred[1] }]; #IO_L7N_T1_AD6N_35 Sch=vga_r[1]
set_property -dict { PACKAGE_PIN C5    IOSTANDARD LVCMOS33 } [get_ports { pred[2] }]; #IO_L1N_T0_AD4N_35 Sch=vga_r[2]
set_property -dict { PACKAGE_PIN A4    IOSTANDARD LVCMOS33 } [get_ports { pred[3] }]; #IO_L8P_T1_AD14P_35 Sch=vga_r[3]
set_property -dict { PACKAGE_PIN C6    IOSTANDARD LVCMOS33 } [get_ports { pgreen[0] }]; #IO_L1P_T0_AD4P_35 Sch=vga_g[0]
set_property -dict { PACKAGE_PIN A5    IOSTANDARD LVCMOS33 } [get_ports { pgreen[1] }]; #IO_L3N_T0_DQS_AD5N_35 Sch=vga_g[1]
set_property -dict { PACKAGE_PIN B6    IOSTANDARD LVCMOS33 } [get_ports { pgreen[2] }]; #IO_L2N_T0_AD12N_35 Sch=vga_g[2]
set_property -dict { PACKAGE_PIN A6    IOSTANDARD LVCMOS33 } [get_ports { pgreen[3] }]; #IO_L3P_T0_DQS_AD5P_35 Sch=vga_g[3]
set_property -dict { PACKAGE_PIN B7    IOSTANDARD LVCMOS33 } [get_ports { pblue[0] }]; #IO_L2P_T0_AD12P_35 Sch=vga_b[0]
set_property -dict { PACKAGE_PIN C7    IOSTANDARD LVCMOS33 } [get_ports { pblue[1] }]; #IO_L4N_T0_35 Sch=vga_b[1]
set_property -dict { PACKAGE_PIN D7    IOSTANDARD LVCMOS33 } [get_ports { pblue[2] }]; #IO_L6N_T0_VREF_35 Sch=vga_b[2]
set_property -dict { PACKAGE_PIN D8    IOSTANDARD LVCMOS33 } [get_ports { pblue[3] }]; #IO_L4P_T0_35 Sch=vga_b[3]
set_property -dict { PACKAGE_PIN B11   IOSTANDARD LVCMOS33 } [get_ports { hs}]; #IO_L4P_T0_15 Sch=vga_hs
set_property -dict { PACKAGE_PIN B12   IOSTANDARD LVCMOS33 } [get_ports { vs}]; #IO_L3N_T0_DQS_AD1N_15 Sch=vga_vs


##Micro SD Connector

#set_property -dict { PACKAGE_PIN E2    IOSTANDARD LVCMOS33 } [get_ports { SD_RESET }]; #IO_L14P_T2_SRCC_35 Sch=sd_reset
#set_property -dict { PACKAGE_PIN A1    IOSTANDARD LVCMOS33 } [get_ports { SD_CD }]; #IO_L9N_T1_DQS_AD7N_35 Sch=sd_cd
#set_property -dict { PACKAGE_PIN B1    IOSTANDARD LVCMOS33 } [get_ports { SD_SCK }]; #IO_L9P_T1_DQS_AD7P_35 Sch=sd_sck
#set_property -dict { PACKAGE_PIN C1    IOSTANDARD LVCMOS33 } [get_ports { SD_CMD }]; #IO_L16N_T2_35 Sch=sd_cmd
#set_property -dict { PACKAGE_PIN C2    IOSTANDARD LVCMOS33 } [get_ports { SD_DAT[0] }]; #IO_L16P_T2_35 Sch=sd_dat[0]
#set_property -dict { PACKAGE_PIN E1    IOSTANDARD LVCMOS33 } [get_ports { SD_DAT[1] }]; #IO_L18N_T2_35 Sch=sd_dat[1]
#set_property -dict { PACKAGE_PIN F1    IOSTANDARD LVCMOS33 } [get_ports { SD_DAT[2] }]; #IO_L18P_T2_35 Sch=sd_dat[2]
#set_property -dict { PACKAGE_PIN D2    IOSTANDARD LVCMOS33 } [get_ports { SD_DAT[3] }]; #IO_L14N_T2_SRCC_35 Sch=sd_dat[3]


##Accelerometer

#set_property -dict { PACKAGE_PIN E15   IOSTANDARD LVCMOS33 } [get_ports { ACL_MISO }]; #IO_L11P_T1_SRCC_15 Sch=acl_miso
#set_property -dict { PACKAGE_PIN F14   IOSTANDARD LVCMOS33 } [get_ports { ACL_MOSI }]; #IO_L5N_T0_AD9N_15 Sch=acl_mosi
#set_property -dict { PACKAGE_PIN F15   IOSTANDARD LVCMOS33 } [get_ports { ACL_SCLK }]; #IO_L14P_T2_SRCC_15 Sch=acl_sclk
#set_property -dict { PACKAGE_PIN D15   IOSTANDARD LVCMOS33 } [get_ports { ACL_CSN }]; #IO_L12P_T1_MRCC_15 Sch=acl_csn
#set_property -dict { PACKAGE_PIN B13   IOSTANDARD LVCMOS33 } [get_ports { ACL_INT[1] }]; #IO_L2P_T0_AD8P_15 Sch=acl_int[1]
#set_property -dict { PACKAGE_PIN C16   IOSTANDARD LVCMOS33 } [get_ports { ACL_INT[2] }]; #IO_L20P_T3_A20_15 Sch=acl_int[2]


##Temperature Sensor

#set_property -dict { PACKAGE_PIN C14   IOSTANDARD LVCMOS33 } [get_ports { TMP_SCL }]; #IO_L1N_T0_AD0N_15 Sch=tmp_scl
#set_property -dict { PACKAGE_PIN C15   IOSTANDARD LVCMOS33 } [get_ports { TMP_SDA }]; #IO_L12N_T1_MRCC_15 Sch=tmp_sda
#set_property -dict { PACKAGE_PIN D13   IOSTANDARD LVCMOS33 } [get_ports { TMP_INT }]; #IO_L6N_T0_VREF_15 Sch=tmp_int
#set_property -dict { PACKAGE_PIN B14   IOSTANDARD LVCMOS33 } [get_ports { TMP_CT }]; #IO_L2N_T0_AD8N_15 Sch=tmp_ct

##Omnidirectional Microphone

#set_property -dict { PACKAGE_PIN J5    IOSTANDARD LVCMOS33 } [get_ports { M_CLK }]; #IO_25_35 Sch=m_clk
```

```
#set_property -dict { PACKAGE_PIN H5    IOSTANDARD LVCMOS33 } [get_ports { M_DATA }]; #IO_L24N_T3_35 Sch=m_data
#set_property -dict { PACKAGE_PIN F5    IOSTANDARD LVCMOS33 } [get_ports { M_LRSEL }]; #IO_0_35 Sch=m_lrsel


##PWM Audio Amplifier

#set_property -dict { PACKAGE_PIN A11   IOSTANDARD LVCMOS33 } [get_ports { AUD_PWM }]; #IO_L4N_T0_15 Sch=aud_pwm
#set_property -dict { PACKAGE_PIN D12   IOSTANDARD LVCMOS33 } [get_ports { AUD_SD }]; #IO_L6P_T0_15 Sch=aud_sd


##USB-RS232 Interface

#set_property -dict { PACKAGE_PIN C4    IOSTANDARD LVCMOS33 } [get_ports { UART_TXD_IN }]; #IO_L7P_T1_AD6P_35 Sch=uart_txd_in
#set_property -dict { PACKAGE_PIN D4    IOSTANDARD LVCMOS33 } [get_ports { UART_RXD_OUT }]; #IO_L11N_T1_SRCC_35 Sch=uart_rxd_out
#set_property -dict { PACKAGE_PIN D3    IOSTANDARD LVCMOS33 } [get_ports { UART_CTS }]; #IO_L12N_T1_MRCC_35 Sch=uart_cts
#set_property -dict { PACKAGE_PIN E5    IOSTANDARD LVCMOS33 } [get_ports { UART_RTS }]; #IO_L5N_T0_AD13N_35 Sch=uart_rts

##USB HID (PS/2)

#set_property -dict { PACKAGE_PIN F4    IOSTANDARD LVCMOS33 } [get_ports { PS2_CLK }]; #IO_L13P_T2_MRCC_35 Sch=ps2_clk
#set_property -dict { PACKAGE_PIN B2    IOSTANDARD LVCMOS33 } [get_ports { PS2_DATA }]; #IO_L10N_T1_AD15N_35 Sch=ps2_data


##SMSC Ethernet PHY

#set_property -dict { PACKAGE_PIN C9    IOSTANDARD LVCMOS33 } [get_ports { ETH_MDC }]; #IO_L11P_T1_SRCC_16 Sch=eth_mdc
#set_property -dict { PACKAGE_PIN A9    IOSTANDARD LVCMOS33 } [get_ports { ETH_MDIO }]; #IO_L14N_T2_SRCC_16 Sch=eth_mdio
#set_property -dict { PACKAGE_PIN B3    IOSTANDARD LVCMOS33 } [get_ports { ETH_RSTN }]; #IO_L10P_T1_AD15P_35 Sch=eth_rstn
#set_property -dict { PACKAGE_PIN D9    IOSTANDARD LVCMOS33 } [get_ports { ETH_CRSDV }]; #IO_L6N_T0_VREF_16 Sch=eth_crsdv
#set_property -dict { PACKAGE_PIN C10   IOSTANDARD LVCMOS33 } [get_ports { ETH_RXERR }]; #IO_L13N_T2_MRCC_16 Sch=eth_rxerr
#set_property -dict { PACKAGE_PIN C11   IOSTANDARD LVCMOS33 } [get_ports { ETH_RXD[0] }]; #IO_L13P_T2_MRCC_16 Sch=eth_rxd[0]
#set_property -dict { PACKAGE_PIN D10   IOSTANDARD LVCMOS33 } [get_ports { ETH_RXD[1] }]; #IO_L19N_T3_VREF_16 Sch=eth_rxd[1]
#set_property -dict { PACKAGE_PIN B9    IOSTANDARD LVCMOS33 } [get_ports { ETH_TXEN }]; #IO_L11N_T1_SRCC_16 Sch=eth_txen
#set_property -dict { PACKAGE_PIN A10   IOSTANDARD LVCMOS33 } [get_ports { ETH_TXD[0] }]; #IO_L14P_T2_SRCC_16 Sch=eth_txd[0]
#set_property -dict { PACKAGE_PIN A8    IOSTANDARD LVCMOS33 } [get_ports { ETH_TXD[1] }]; #IO_L12N_T1_MRCC_16 Sch=eth_txd[1]
#set_property -dict { PACKAGE_PIN D5    IOSTANDARD LVCMOS33 } [get_ports { ETH_REFCLK }]; #IO_L11P_T1_SRCC_35 Sch=eth_refclk
#set_property -dict { PACKAGE_PIN B8    IOSTANDARD LVCMOS33 } [get_ports { ETH_INTN }]; #IO_L12P_T1_MRCC_16 Sch=eth_intn


##Quad SPI Flash

#set_property -dict { PACKAGE_PIN K17   IOSTANDARD LVCMOS33 } [get_ports { QSPI_DQ[0] }]; #IO_L1P_T0_D00_MOSI_14 Sch=qspi_dq[0]
#set_property -dict { PACKAGE_PIN K18   IOSTANDARD LVCMOS33 } [get_ports { QSPI_DQ[1] }]; #IO_L1N_T0_D01_DIN_14 Sch=qspi_dq[1]
#set_property -dict { PACKAGE_PIN L14   IOSTANDARD LVCMOS33 } [get_ports { QSPI_DQ[2] }]; #IO_L2P_T0_D02_14 Sch=qspi_dq[2]
#set_property -dict { PACKAGE_PIN M14   IOSTANDARD LVCMOS33 } [get_ports { QSPI_DQ[3] }]; #IO_L2N_T0_D03_14 Sch=qspi_dq[3]
#set_property -dict { PACKAGE_PIN L13   IOSTANDARD LVCMOS33 } [get_ports { QSPI_CSN }]; #IO_L6P_T0_FCS_B_14 Sch=qspi_csn
```

6.5 综合设计

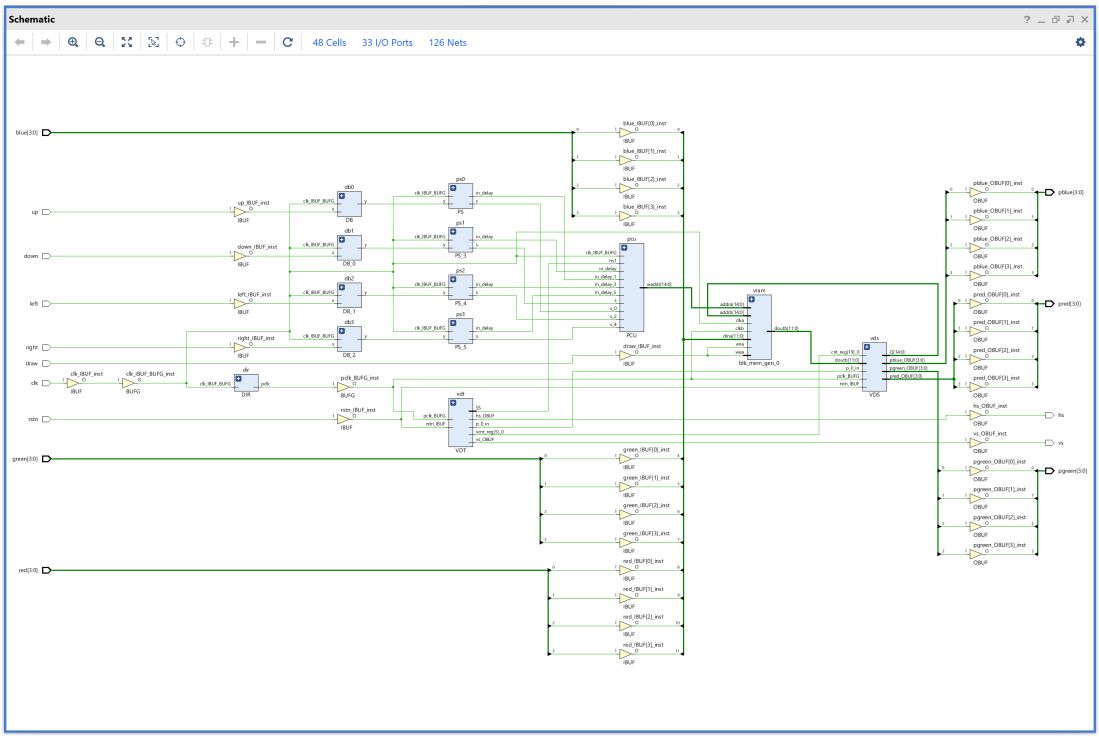

图 11: 综合设计电路图

6.6 电路资源使用情况

Copyright 1986–2019 Xilinx, Inc. All Rights Reserved.

| Tool Version : Vivado v.2019.1 (win64) Build 2552052 Fri May 24 14:49:42 MDT 2019
| Date : Sun Nov 28 21:02:13 2021
| Host : DESKTOP-DM7M7LO running 64-bit major release (build 9200)
| Command : report_utilization -file PNT_utilization_synth.rpt -pb PNT_utilization_synth.pb
| Design : PNT
| Device : 7a100tcsg324-1
| Design State : Synthesized

Utilization Design Information

Table of Contents

1. Slice Logic

Site Type	Used	Fixed	Available	Util%
Slice LUTs*	330	0	63400	0.52
LUT as Logic	330	0	63400	0.52
LUT as Memory	0	0	19000	0.00
Slice Registers	166	0	126800	0.13
Register as Flip Flop	162	0	126800	0.13
Register as Latch	4	0	126800	<0.01
F7 Muxes	0	0	31700	0.00
F8 Muxes	0	0	15850	0.00

* Warning! The Final LUT count, after physical optimizations and full implementation, is typically lower. Run opt_design after synthesis, if not alread				

1.1 Summary of Registers by Type

Total	Clock Enable	Synchronous	Asynchronous
0	_	-	-
0	_	-	Set
0	_	-	Reset
0	_	Set	-
0	_	Reset	-
0	Yes	-	-
0	Yes	-	Set
4	Yes	-	Reset
29	Yes	Set	-
133	Yes	Reset	-

2. Memory

Site Type	Used	Fixed	Available	Util%
Block RAM Tile	0	0	135	0.00
RAMB36/FIFO*	0	0	135	0.00
RAMB18	0	0	270	0.00

* Note: Each Block RAM Tile only has one FIFO logic available and therefore can accommodate only one FIFO36E1 or one FIFO18E1. However, if a FIFO18E1 c				

3. DSP

Site Type	Used	Fixed	Available	Util%
DSPs	0	0	240	0.00

4. IO and GT Specific

Site Type	Used	Fixed	Available	Util%
Bonded IOB	33	0	210	15.71
Bonded IPADs	0	0	2	0.00
PHY_CONTROL	0	0	6	0.00
PHASER_REF	0	0	6	0.00
OUT_FIFO	0	0	24	0.00
IN_FIFO	0	0	24	0.00
IDELAYCTRL	0	0	6	0.00
IBUFDS	0	0	202	0.00
PHASER_OUT/PHASER_OUT_PHY	0	0	24	0.00
PHASER_IN/PHASER_IN_PHY	0	0	24	0.00
IDELAYE2/IDELAYE2_FINEDELAY	0	0	300	0.00
ILOGIC	0	0	210	0.00
OLOGIC	0	0	210	0.00

5. Clocking

Site Type	Used	Fixed	Available	Util%
BUFGCTRL	2	0	32	6.25
BUFIO	0	0	24	0.00
MMCME2_ADV	0	0	6	0.00
PLLE2_ADV	0	0	6	0.00
BUFMRCE	0	0	12	0.00
BUFHCE	0	0	96	0.00
BUFR	0	0	24	0.00

6. Specific Feature

Site Type	Used	Fixed	Available	Util%
BSCANE2	0	0	4	0.00
CAPTUREE2	0	0	1	0.00
DNA_PORT	0	0	1	0.00
EFUSE_USR	0	0	1	0.00
FRAME_ECCE2	0	0	1	0.00
ICAPE2	0	0	2	0.00
PCIE_2_1	0	0	1	0.00
STARTUPE2	0	0	1	0.00
XADC	0	0	1	0.00

7. Primitives

Ref Name	Used	Functional Category
FDRE	133	Flop & Latch
LUT2	99	LUT
LUT4	86	LUT
LUT6	76	LUT
CARRY4	76	CarryLogic
LUT3	71	LUT
LUT5	52	LUT
FDSE	29	Flop & Latch
LUT1	24	LUT
IBUF	19	IO
OBUF	14	IO
LDCE	4	Flop & Latch
BUFG	2	Clock

8. Black Boxes

Ref Name	Used
blk_mem_gen_0	1

9. Instantiated Netlists

Ref Name	Used

6.7 下载结果

图 12: 下载结果

7 实验总结

7.1 Bug 分析

问题 1：显示白色背景时，显示器没有任何显示。

原因 1：为省事，将颜色信号输出与常量相接，导致显示器扫描到屏幕边缘及进行行、场同步时也有颜色信号，以致显示出错。

修改 1：使用时序逻辑判断给出颜色信号的时间，在屏幕边缘给出黑色。

问题 2：显示色块时，显示器颜色混乱且频闪。

原因 2：在同步计数器复位时，判断的当是其值为超尾值时再将其置为 0，导致每次场同步都错开一个像素位。

修改 2：将复位条件设为计数器值与尾值相同。

问题 3：绘图模块下载后，按动上下左右按钮导致的画笔方向更改与设计不一致。

原因 3：在传入方向信号时，使用了位拼接运算符。位拼接运算符中靠右的信号编号为 0，而我起初误认为靠左的信号编号为 0.

修改 3：将位拼接运算符中的元素顺序颠倒。

7.2 收获与思考

本次实验可以说是这一系列实验中最有趣的一次，尤其是学到了显示器的显示原理。虽然在课上讲解时感到这次实验内容很困难，但实际上并没有。经过前几次的实验，我初步地掌握了模块化的数字系统设计方法，这也使我能在这次实验中比较快地完成任务。数据通路和状态图，是数字系统设计中最为关键的两步。仔细、认真地完成它们，可以使后续实验步骤更加容易。本次实验共用时约 9h，比想象中的少一半以上。

8 意见建议

本次实验建议一周完成，剩下一周用在改为两周完成的第 5 次实验。