



EGOPC_Lab07(v0.1)2018/07/30

VGA图像显示实验流程

Author: E-ELEMENTS

E-Mail: support@e-elements.com

概述

VGA(Video Graphics Array)是IBM在1987年随PS/2机一起推出的一种视频传输标准, 具有分辨率高、显示速率快、颜色丰富等优点, 在彩色显示器领域得到了广泛的应用。本实验使用Vivado2017.4开发工具在EGOPC开发板上采用Verilog HDL语言编写VGA的显示程序, 使得E-ELEMENTS的在屏幕上动态显示。

说明

本实验使用Vivado Design Suite(Vivado)2017.4创建和构建硬件设计, 并结合EGOPC硬件平台为载体完成本实验。本实验手册提供一套完整的实验设计流程, 可作为使用者学习和熟悉VGA显示的参考, 并允许使用者以此为基础拓展VGA显示系统。

本实验使用平台为E-ELEMENTS EGOPC平台, 可关注E-ELEMENTS微信公众号获取相关信息以及Xilinx大学计划最新消息。



环境

硬件环境

- PC
- EGOPC平台
- Type-c数据线
- VGA显示器

软件环境

- 64位操作系统(Windows)
- 安装有Vivado Design Suite2017.4工具

使用规范

电路板建议在绝缘平台上使用, 否则可能引起电路板损坏。

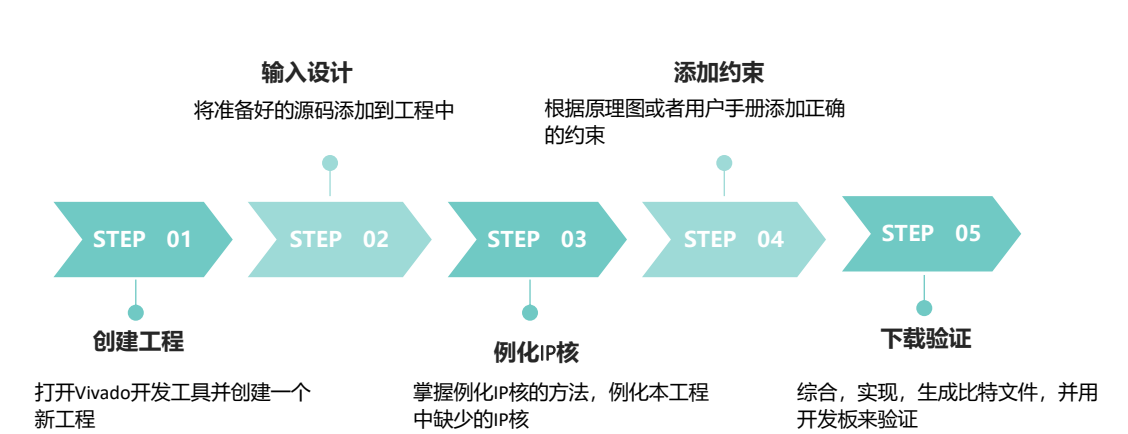
- 电路使用时应防止静电。
- 液晶显示器件或模块结雾时, 不要通电工作, 防止电极化学反应, 产生断线。
- 电源正负极、输入/输出端口定义时需谨慎, 避免应接反引起开发板的损坏。
- 保持电路板的表面清洁。
- 小心轻放, 避免不必要的硬件损伤

实验目标

当完成本实验后，会学习到：

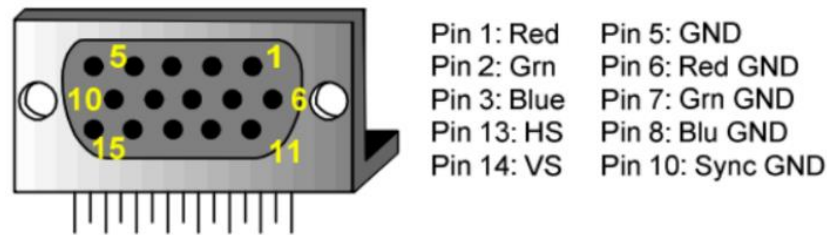
- Vivado开发工具的设计流程
- IP核的添加
- IP核的例化
- VGA图像显示方法

实验流程

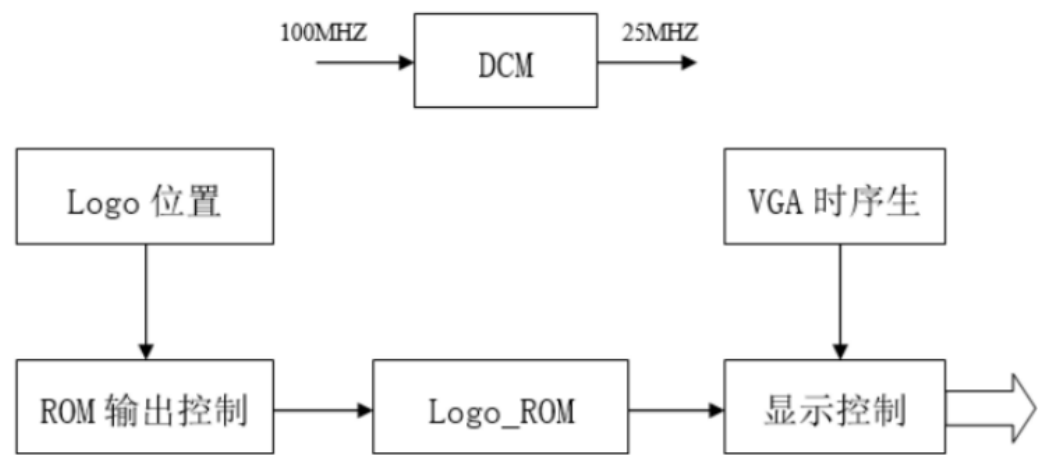


实验原理

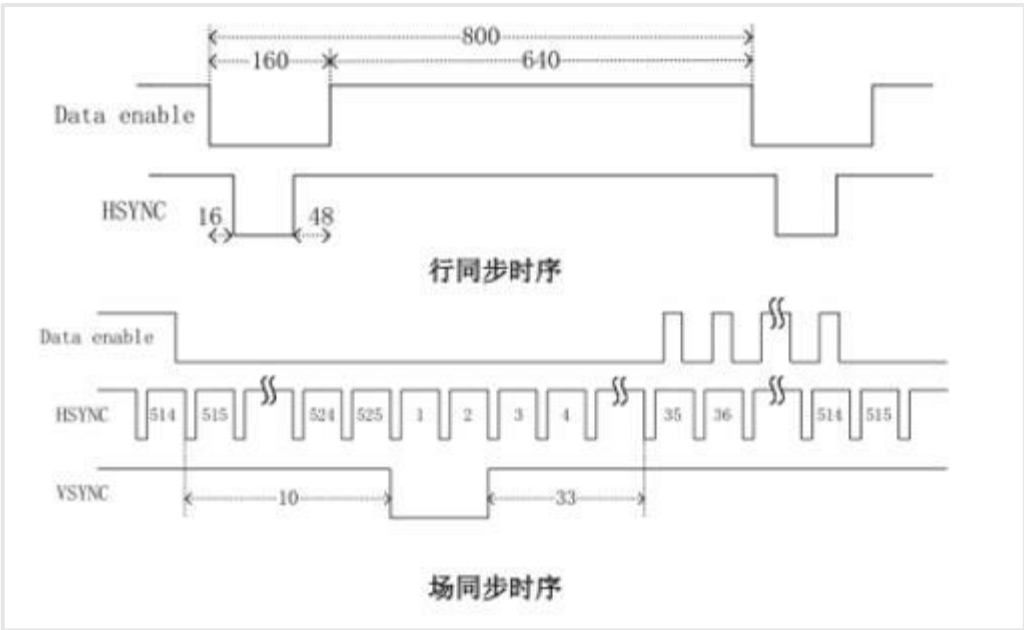
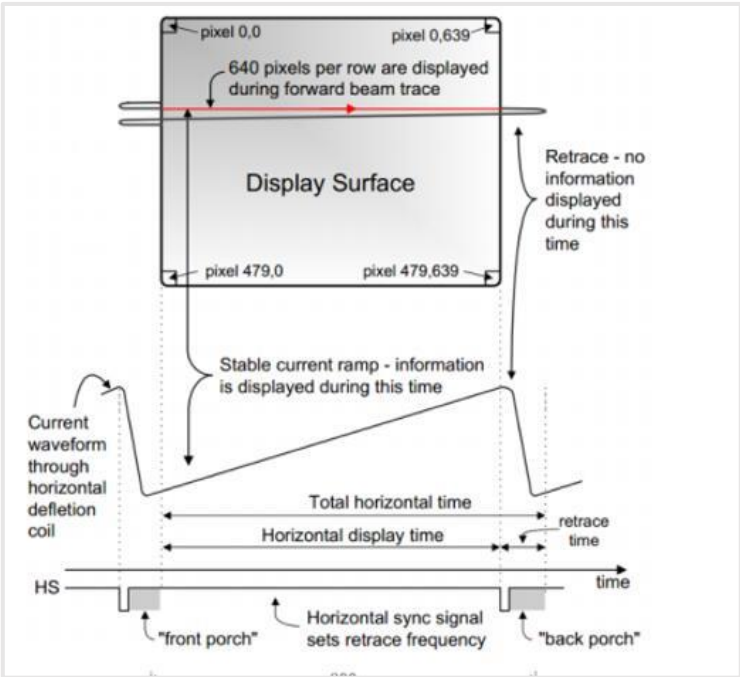
VGA接口是显示器上应用最为广泛的接口类型，它是一种D型接口，上面共有15个针孔，信号连接如下图所示：



通用VGA显示卡系统主要由控制电路、显示缓存区和程序BIOS程序三个部分组成。控制电路主要完成时序发生、显示缓冲区数据操作、主时钟选择和D/A转换等功能；显示缓冲区提供显示数据缓存空间；视频BIOS作为控制程序固化在显示卡ROM中。本实验的系统架构如下图所示：

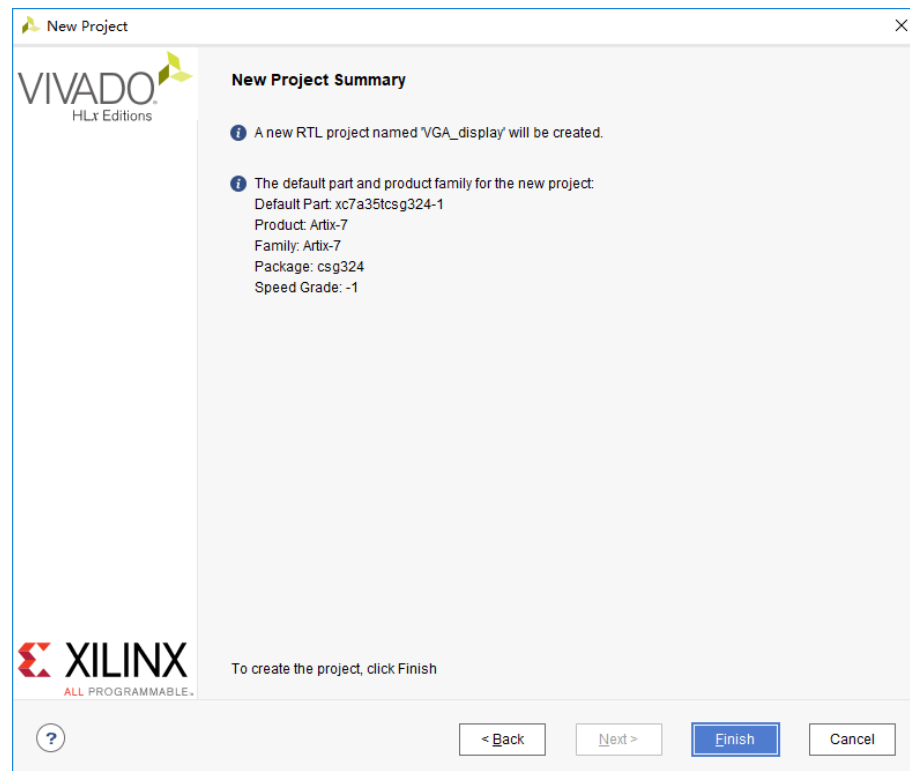


要实现VGA显示，主要还是实现VGA时序的控制，时序标准参见下图。行时序和帧时序都需要产生同步脉冲sync、显示后沿back porch、显示时序段active和显示前沿front porch四个部分。



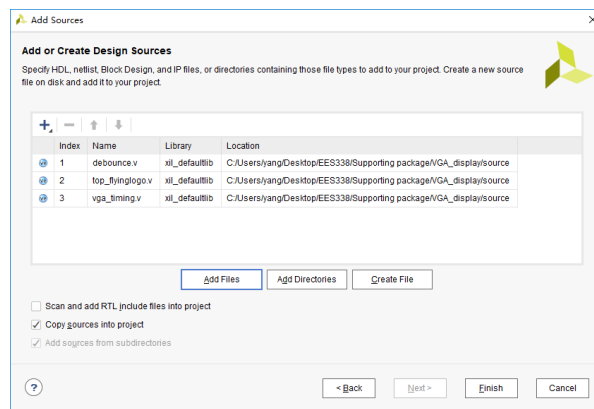
创建工程

1. 双击桌面图标启动Vivado2017.4，并点击Create Project新建一个新的设计工程，进入创建引导后，点击 Next 进入下一步。
2. 输入工程名，并选择工程目录后，点击Next。
3. 选择创建类型为RTL Project设计，点击Next。
4. 在Search界面输入XC7A35TCSG324-1,选择该器件点击Next。
5. 在弹出的New Project Summary中重新检查所选器件型号是否与板卡芯片型号一致，若确认无误，点击 Finish 完成工程创建。创建工程结果如下图所示。

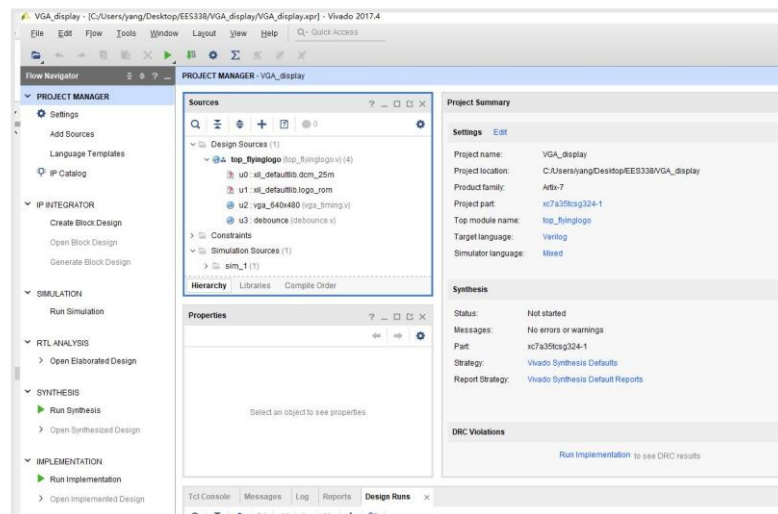


输入设计

1. 在左侧 “Flow Navigator” 栏中的 “Project Manager” 下点击 “Add Source” ， 在弹出的窗口中选择 “Add or create design source” ， 点击 “Next” 。
2. 选择 “Add Files” 。
3. 在配套资料包中对应工程目录下 “source” 的文件夹中，选择添加实验的源文件，点击 “OK” 。



4. 点击 “Finish” 。
5. 在source窗口下可以查看添加的源文件。

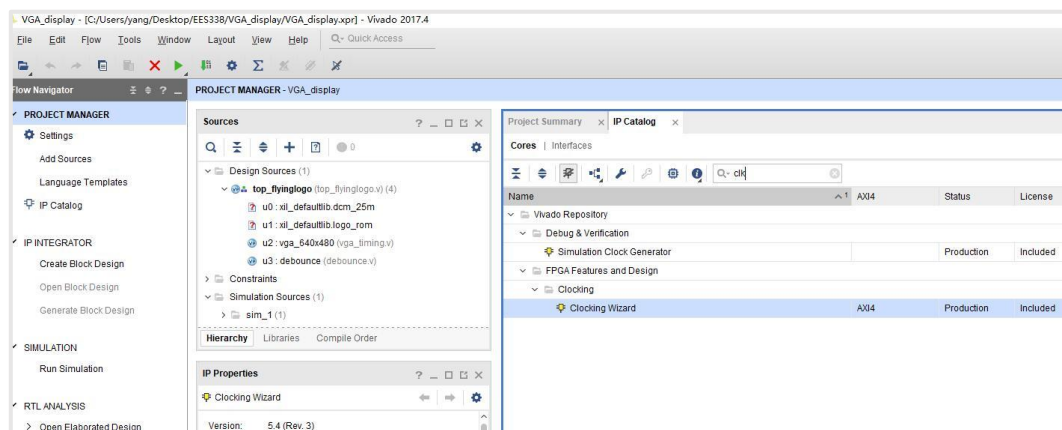


例化IP核

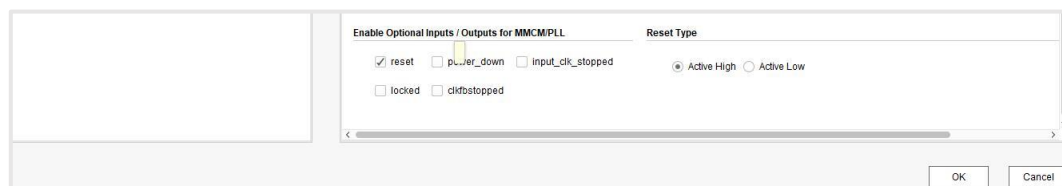
1. 在上图我们可以看到有两个模块缺失，一个是时钟模块，另一个是ROM模块。我们将通过例化IP核来完成这两个模块的设计。

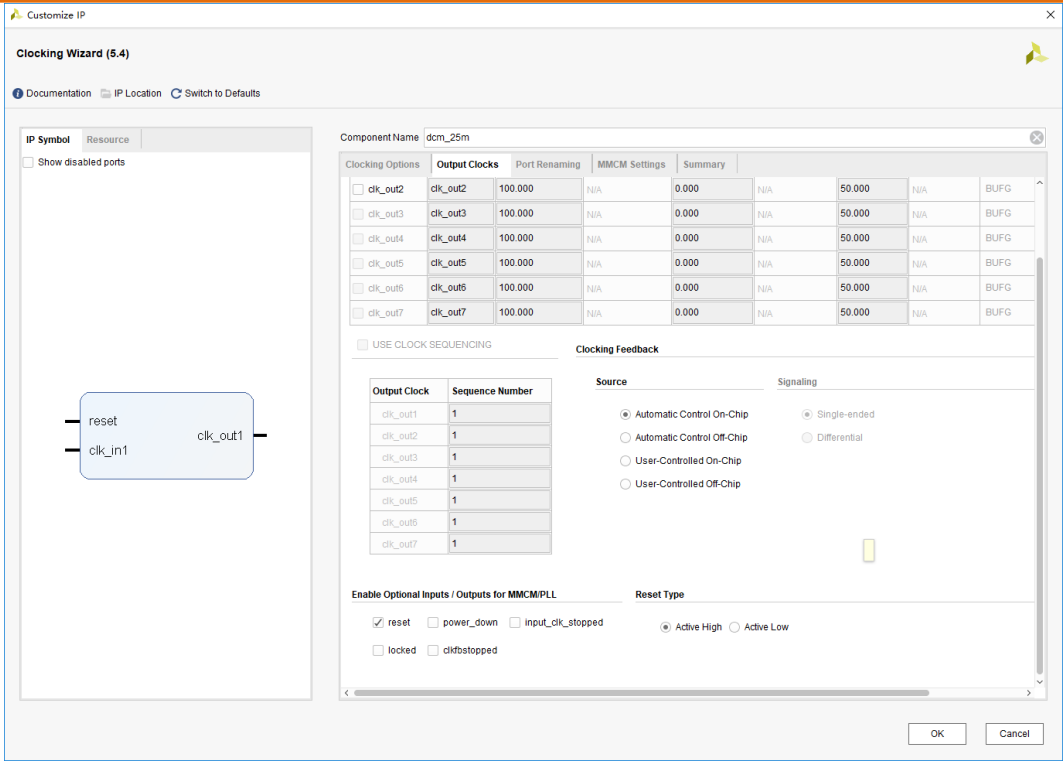
本实验需要用25MHz的时钟驱动VGA控制器，接下来的步骤通过例化时钟IP，将板卡上的100M板载时钟分频得到25MHz时钟。

在左侧“Flow Navigator”栏中的“Project Manager”下点击“IP Catalog”。在界面右侧弹出的“IP Catalog”窗口中选择“FPGA Features and Design->Clocking->Clocking Wizard”。



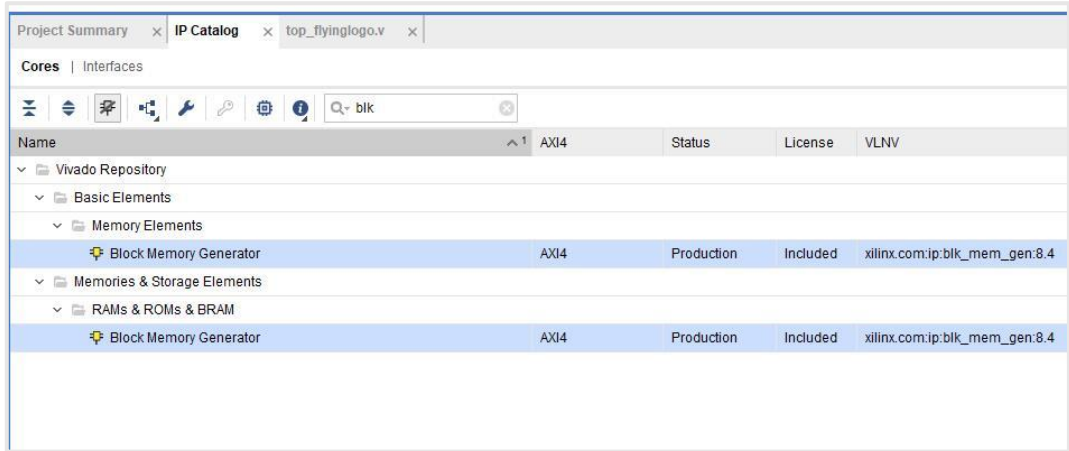
2. 双击打开“Clocking Wizard”，在“Customize IP”窗口中将 Clocking Wizard 的 Component Name 改为“dcm_25m”。同时在“Output Clocks”标签页下将“clk_out1”改为25MHz，然后下拉到界面下方，取消locked选项。



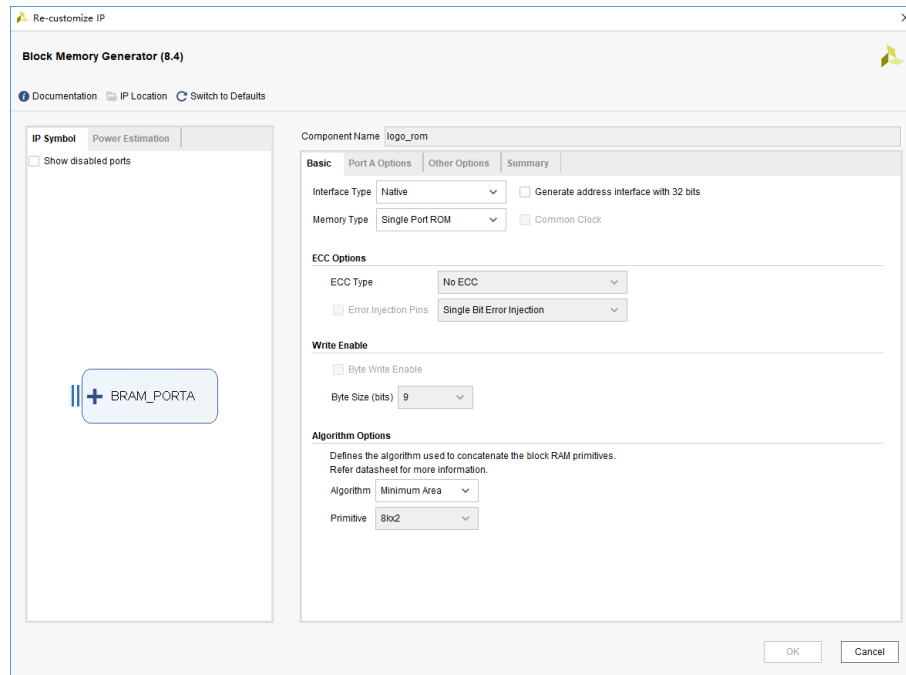


3. 点击OK，在弹出的窗口中选择 Out of context per IP，然后点击 “Generate” 。该过程结束后在弹出的窗口中点击 “OK” 确认。

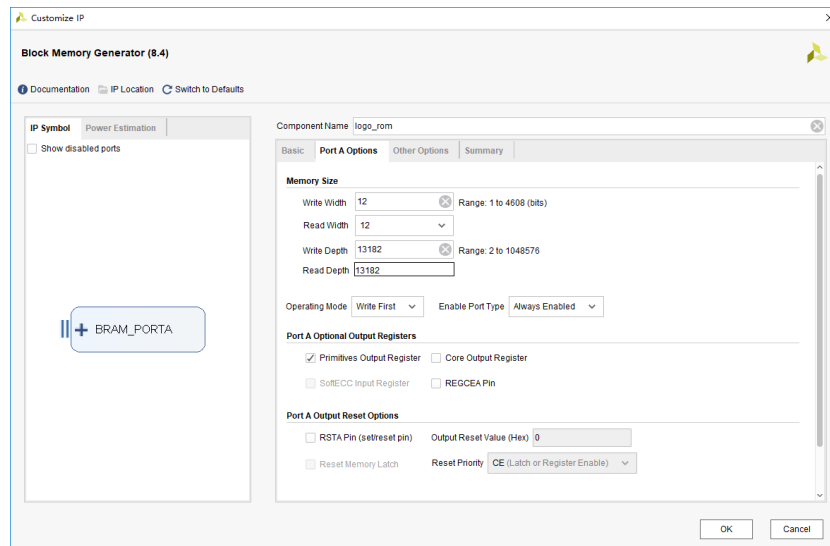
4. 至此，我们已经完成时钟IP的例化，接下来我们通过例化ROM的IP核，并对其进行初始化，将我们要显示的图片数据存放在ROM中。同样在 “IP Catalog” 窗口中选择 “Memories & Storage Elements→RAMs & ROMs &BRAM→Block Memory Generator”。



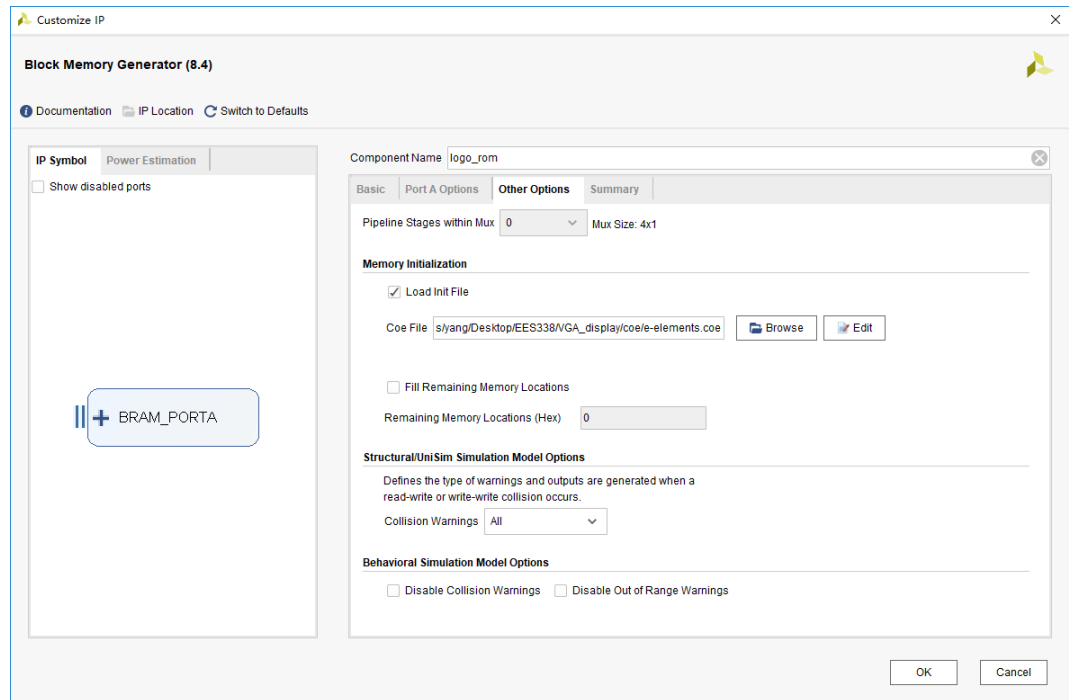
5. 双击打开Block Memory Generator, 在 Customize IP 窗口中, 将 Component Name 改为 logo_rom, 在 Basic 标签页下, Memory Type 选择 Single Port ROM。



6. 在 Port A Options 标签页下, 将 Port A Width 设置为 12, Port A Depth 设置为 13182 (图片为 169×78), Enable Port Type 设置为 Always Enable。



7. 将Supporting package文件夹下对应工程目录的 coe 文件夹拷贝到当前工程目录下。在 Other Options 标签页中勾选 Load Init File，然后通过点击右侧的文件夹图标，将目录指向刚刚拷贝的 coe 文件夹，双击其中的初始化文件 e-elements.coe，将其加载到ROM中。



8. 点击 OK，在弹出的窗口中选择 Out of context per IP，然后点击“Generate”。

添加约束

1. 在界面左侧 Flow Navigator 一栏中点击 RTL Analysis 下的 Open Elaborated Design，然后在菜单栏中选择 Layout→I/O Planning。在 “I/O Ports” 窗口中对输入输出信号添加约束。首先在 “I/O Std” 一栏通过下拉按钮选择 “LVCMOS33”，将所有信号的电平标准设置 3.3V。在 “Site” 一栏分配各个信号在 FPGA 芯片上引脚的位置，各信号的具体位置可查看板卡的原理图。

Tcl Console Messages Log Reports Design Runs Package Pins I/O Ports x									
Q [Icons]									
Name	Direction	Neg Diff Pair	Package Pin	Fixed	Bank	I/O Std	Vcco	Vref	Drive Strength
vga_b[3]	OUT		E7	✓	35	LVCMOS33*	3.300		12
vga_b[2]	OUT		E5	✓	35	LVCMOS33*	3.300		12
vga_b[1]	OUT		E6	✓	35	LVCMOS33*	3.300		12
vga_b[0]	OUT		C7	✓	35	LVCMOS33*	3.300		12
vga_g (4)	OUT			✓	35	LVCMOS33*	3.300		12
vga_g[3]	OUT		D8	✓	35	LVCMOS33*	3.300		12
vga_g[2]	OUT		A5	✓	35	LVCMOS33*	3.300		12
vga_g[1]	OUT		A6	✓	35	LVCMOS33*	3.300		12
vga_g[0]	OUT		B6	✓	35	LVCMOS33*	3.300		12
vga_r (4)	OUT			✓	35	LVCMOS33*	3.300		12
vga_r[3]	OUT		B7	✓	35	LVCMOS33*	3.300		12
vga_r[2]	OUT		C5	✓	35	LVCMOS33*	3.300		12
vga_r[1]	OUT		C6	✓	35	LVCMOS33*	3.300		12
vga_r[0]	OUT		F5	✓	35	LVCMOS33*	3.300		12
Scalar ports (4)									
clk	IN		T5	✓	34	LVCMOS33*	3.300		
hsync	OUT		D7	✓	35	LVCMOS33*	3.300		12
rst	IN		P15	✓	14	LVCMOS33*	3.300		
vsync	OUT		C4	✓	35	LVCMOS33*	3.300		12

2. 管脚分配完成后点击左上角的保存按钮，在弹出的对话框中 “File Name” 一栏输入约束文件的名称 “display_vga”，点击 “OK”。

3. 在右上角浅蓝色区域点击叉号，点击 “OK” 确认关闭 “Elaborated Design”。在 “Sources” 窗口中的 “Constraints/ constrs_1” 下双击 “display_vga.xdc”，打开刚刚创建的约束文件并查看内容。也可以通过直接添加约束文件的方式来对设计进行管脚分配。

下载验证

1. 在 “Flow Navigator” 一栏中的 “Program and Debug” 下点击 “Generate Bitstream”，此时会提示工程没有实现，点击 “Yes”会自动执行综合及实现过程。
2. 生成比特流文件完成后，选择 “Open Hardware Manager” 并点击 “OK”。
3. 将显示器与板卡上 VGA 接口连接，然后用 Type C 线连接电脑与板卡，并打开电源开关。在 “Hardware Manager” 界面点击 “Open target”，选择 “Auto Connect”。
4. 连接成功后，在目标芯片上右击，选择 “Program Device”。
5. 在弹出的对话框中 “Bitstream File” 一栏中已经自动加载本工程生成的比特流文件，点击 “Program” 对 FPGA 芯片进行编程。
6. 比特流文件下载完成后，在 VGA 显示器上验证实验结果



参考文献

Xilinx User Guide [UG888](#), Vivado Design Suite Tutorial: Design Flows Overview

Xilinx User Guide [UG892](#), Vivado Design Suite User Guide : Design Flows Overview

Xilinx User Guide [UG901](#), Vivado Design Suite User Guide : Synthesis

Xilinx User Guide [UG904](#), Vivado Design Suite User Guide : Implementation

Xilinx User Guide [UG908](#), Vivado Design Suite User Guide : Programming and Debugging

Xilinx User Guide [UG939](#), Vivado Design Suite Designing with IP Tutorial

Xilinx User Guide [UG1199](#), Vivado Design Suite Creating, Packaging Custom IP Tutorial

Xilinx User Guide [UG896](#), Vivado Design Suite User Guide: Designing with IP

Xilinx User Guide [UG1118](#), Vivado Design Suite User Guide Creating and Packaging Custom IP

Xilinx LogiCORE IP Product Guide [PG065](#): Clocking Wizard v5.4

Xilinx LogiCORE IP Product Guide [PG058](#): Block Memory Generator v8.4