

# *Lab10: VGA 显示*

*基于 Nexys 4 FPGA 平台*

## Lab 10: VGA 显示

### 实验简介

本实验旨在使读者学会 Xilinx 的 XPS 和 SDK 工具的使用，导入 VGA 的 IP 核，完成三角函数图像、一次函数图像的显示。

### 实验目标

在完成本实验后，您将学会：

- VGA 端口的设计过程。

### 实验过程

本实验旨在指导读者使用 Xilinx 的 XPS 工具，导入 VGA 的 IP 核，编写 C 文件，完成三角函数图像、一次函数图像、抛物线图像在 VGA 显示器上的显示。

实验由以下步骤组成：

1. 在 XPS 中建立工程
2. 导入 VGA IP 核
3. 端口映射
4. 将工程导入 SDK
5. 在 SDK 中添加 C 语言源程序
6. 将 VGA 端口与 Nexys4 链接
7. 在 Nexys 上进行测试验证，查看实验结果

### 实验环境

#### ◆ 硬件环境

1. PC 机
2. Nexys 4 FPGA 平台
3. VGA 显示器

#### ◆ 软件环境

Xilinx ISE Design Suite 14.3（FPGA 开发工具）

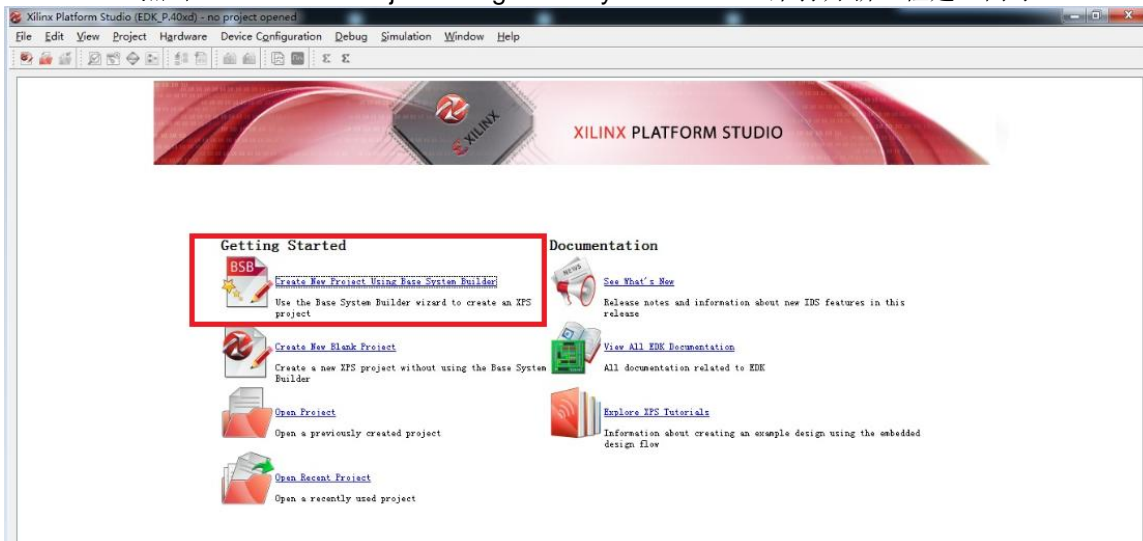
注：所有源代码可在 \*\*/lab10/Sources/ 目下找到

## 第一步 创建工程

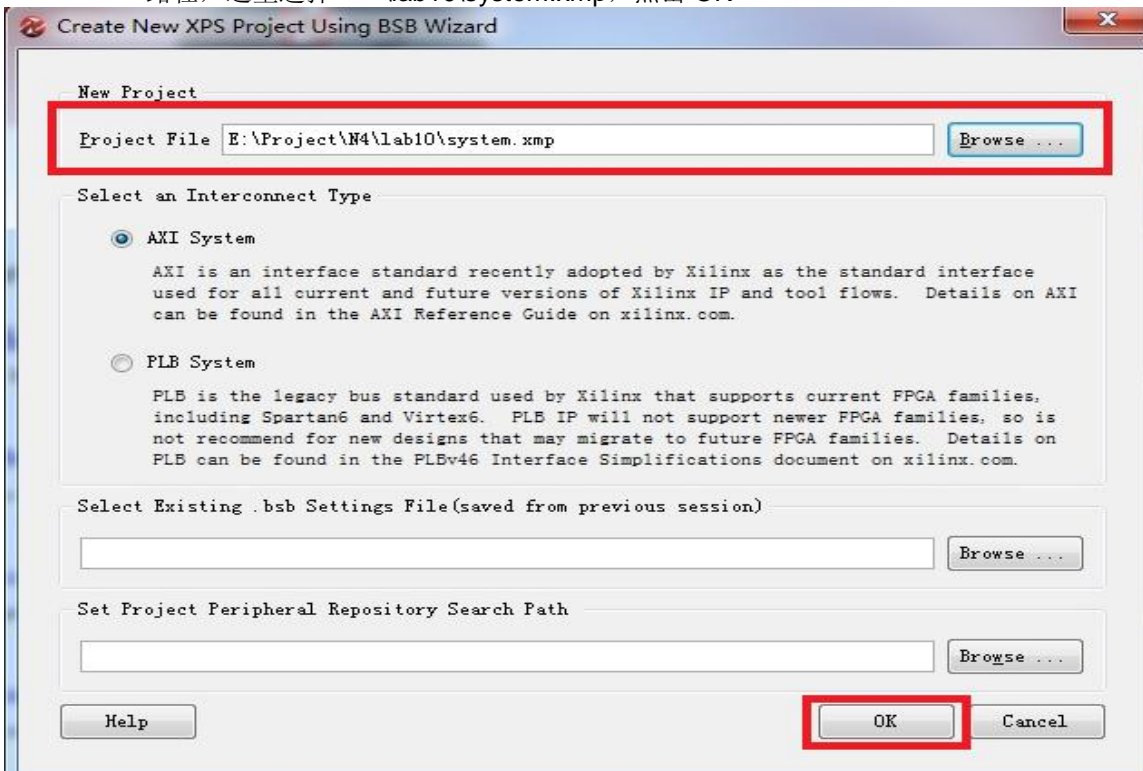
### 1-1. 运行 Xilinx Platform Studio，创建一个新的工程

1-1.1. 选择开始菜单-所有程序--Xilinx Design Tools--ISE Design Suit 14.3--EDK--Xilinx Platform Studio。点击运行 Xilinx Platform Studio。

1-1.2. 点击 Create New Project Using Base Sysetm Builder 来打开新工程建立向导



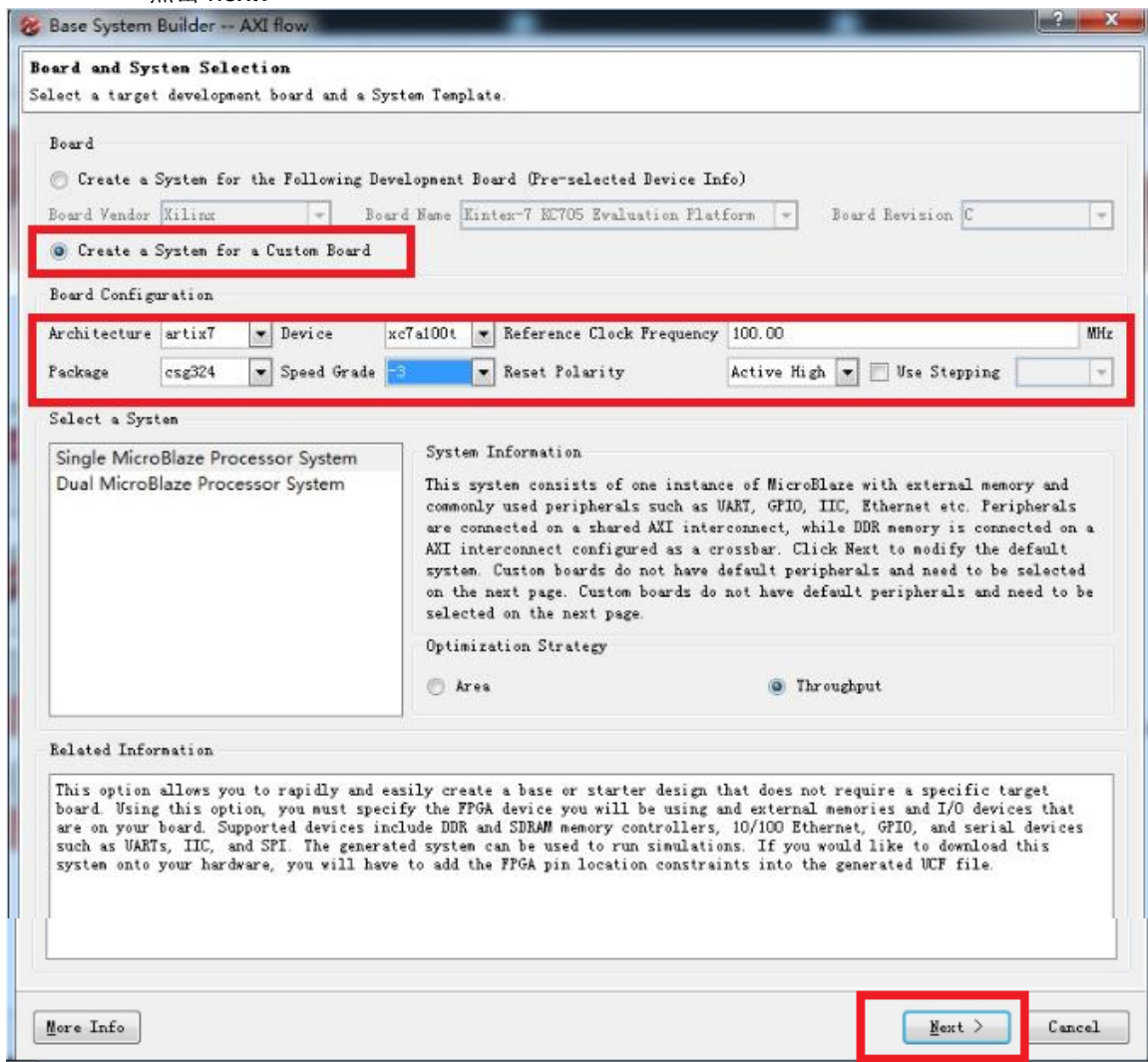
1-1.3. 出现 Create New XPS Project Using BSB Wizard 对话框，在 Project File 处选择自己的工程路径，这里选择\*\*\*\*\lab10\system.xmp，点击 OK



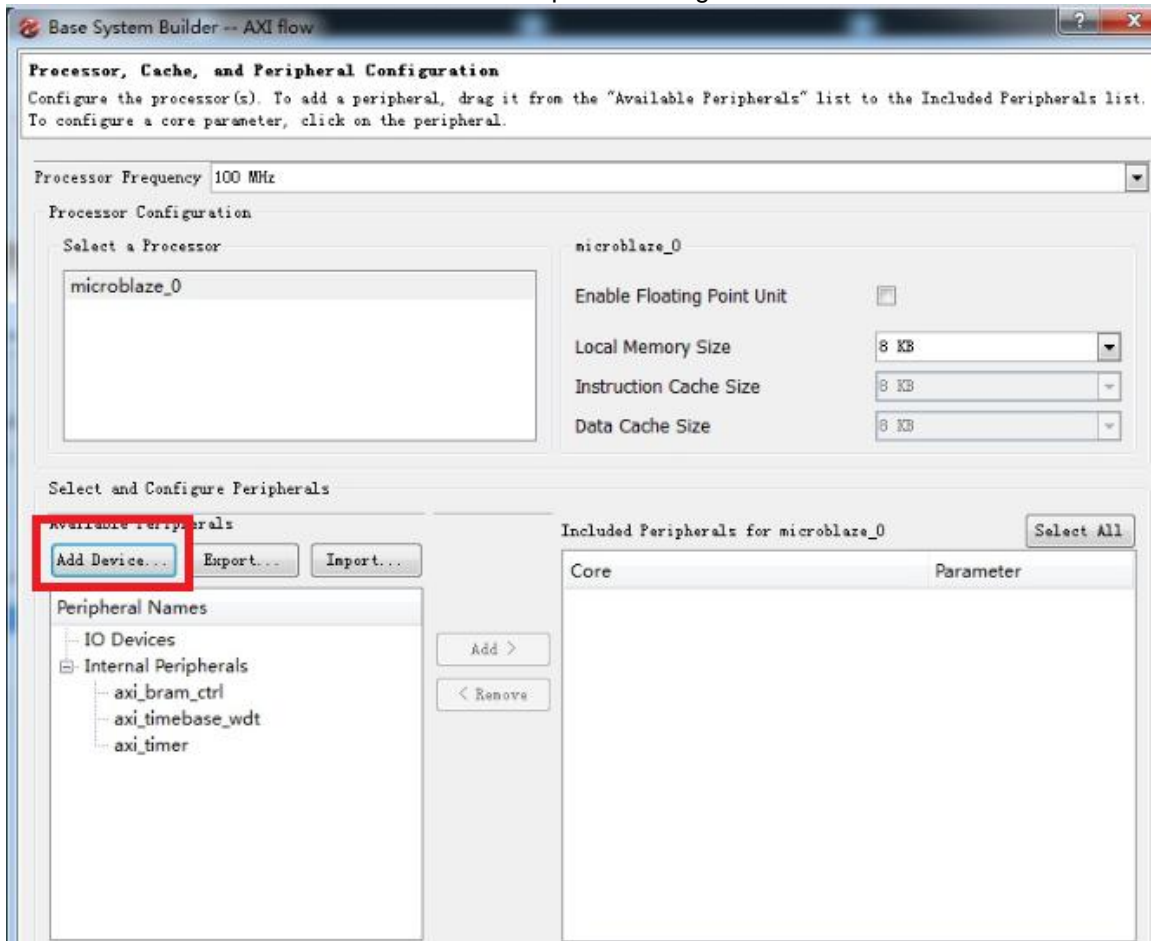
1-1.4. 出现 Board and System Selection 对话框，选择 Create a System for a Custom Board Board Configuration 中的配置信息如下：

Architecture : **artix7**  
Device : **xc7a100t**  
Clock Frequency : **100.00Mhz**  
Package : **csg324**  
Speed Grade : **-3**  
Reset Polarity : **Active High**

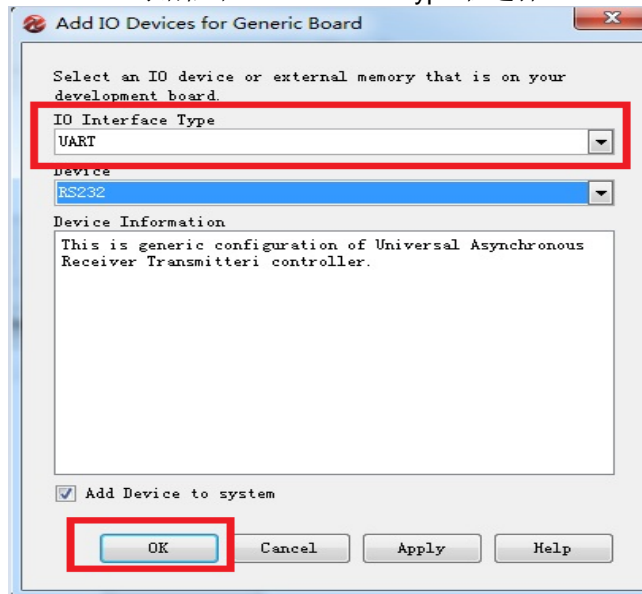
点击 next。



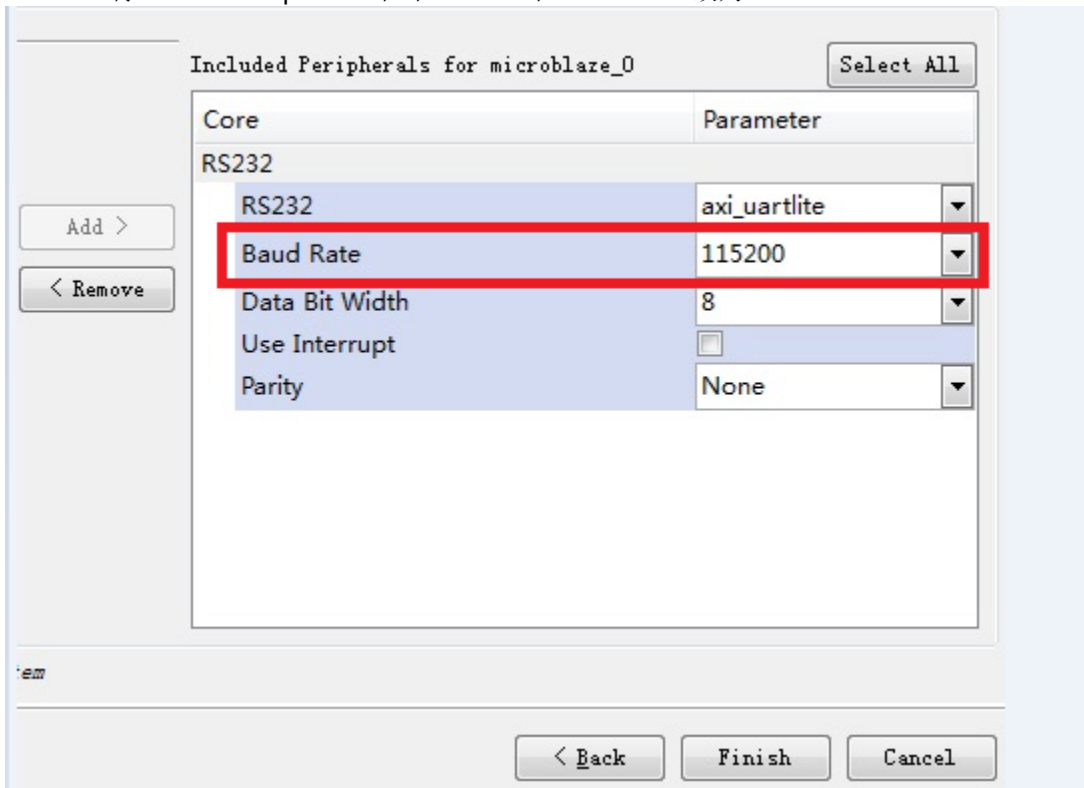
1-1.5. 出现 Processor, Cache, and Peripheral Configuration 对话框, 点击 Add Device



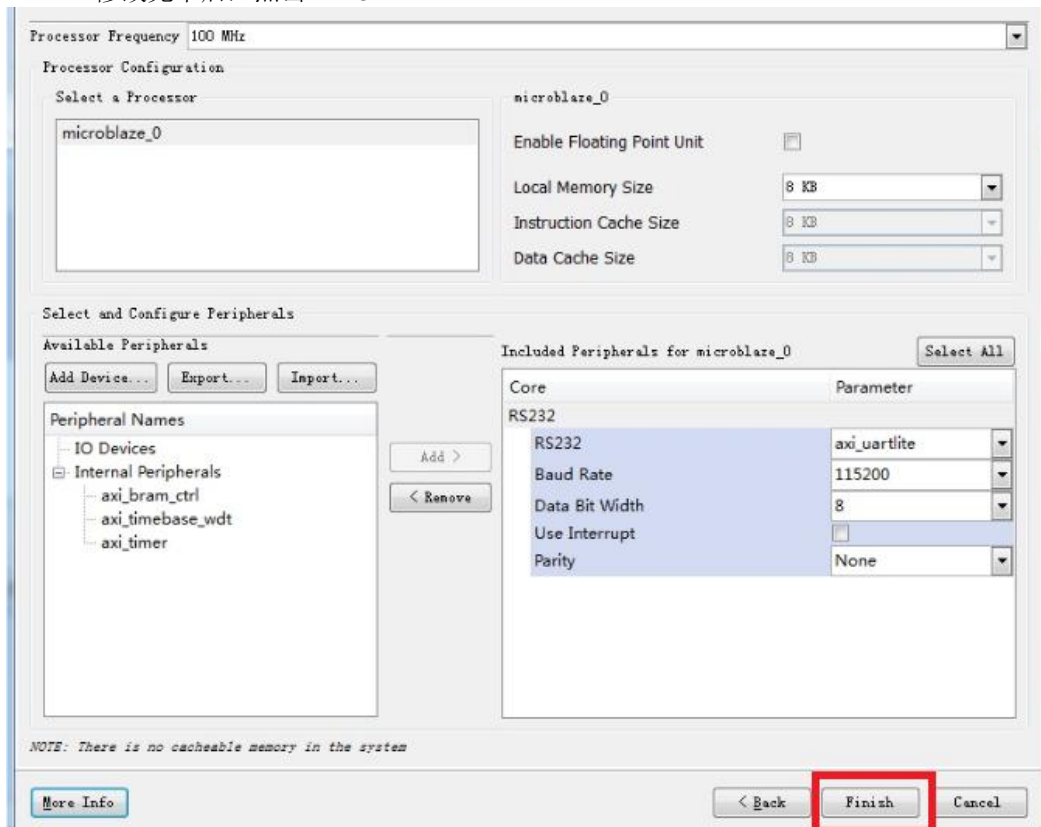
1-1.6. 出现 Add IO Device 对话框 在 IO Interface Type 栏选择 UART, 点击 OK



1-1.7. 将 Include Peripherals 栏中 RS232 中 Baud Rate 改为 115200。



修改完毕后，点击 Finish



1-1.8.至此，一个只带有串口 IP 核的 EDK 工程创建完毕

The screenshot displays the Xilinx EDK (Vivado) environment. On the left is the IP Catalog with a tree view showing categories like Analog, Arithmetic, Bus and Bridge, etc. In the center is a block diagram showing a system with a processor and various peripheral blocks connected via buses. On the right is the 'Bus Interfaces' table, which lists the components and their versions.

Name	Bus Name	IP Type	IP Version
axi4lite_0		axi_interco...	1.06.a
microblaz...		lmb_v10	2.00.b
microblaz...		lmb_v10	2.00.b
microblaz...		microblaze	8.40.b
microblaz...		bram_block	1.00.a
microblaz...		lmb_bram...	3.10.b
microblaz...		lmb_bram...	3.10.b
debug_mo...		mdm	2.10.a
RS232		axi_uartlite	1.02.a
clock_gen...		clock_gen...	4.03.a
proc_sys_r...		proc_sys_r...	3.00.a

Legend:

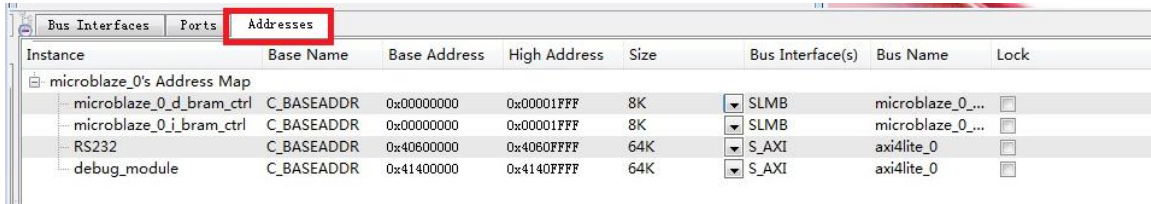
- Master (blue square), Slave (green circle), Master/Slave (purple diamond), Target (red triangle), Initiator (yellow circle)
- Connected (blue line), Unconnected (grey line), Monitor (green line)
- Production (green star), License (paid) (yellow star), License (eval) (orange star), Local (blue star), Pre Production (red star), Beta (purple star), Development (pink star)
- Superseded (yellow triangle), Discontinued (orange triangle)



## 第二步 配置工程参数，添加 VGA IP 核并配置其参数

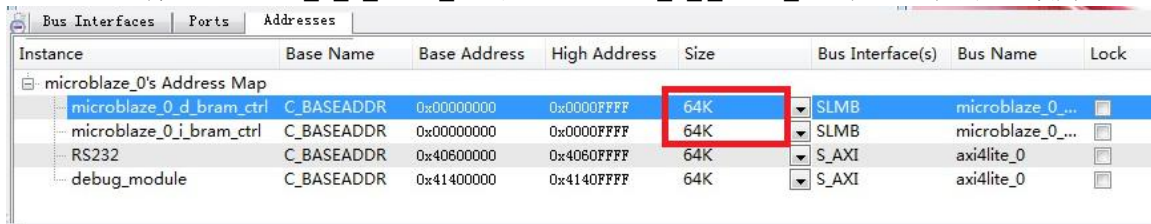
### 2-1.修改 Microblaze 的 bram 大小

#### 2-1.1. 选择 Address 栏



Instance	Base Name	Base Address	High Address	Size	Bus Interface(s)	Bus Name	Lock
microblaze_0's Address Map							
microblaze_0_d_bram_ctrl	C_BASEADDR	0x00000000	0x00001FFF	8K	SLMB	microblaze_0_...	<input type="checkbox"/>
microblaze_0_i_bram_ctrl	C_BASEADDR	0x00000000	0x00001FFF	8K	SLMB	microblaze_0_...	<input type="checkbox"/>
RS232	C_BASEADDR	0x40600000	0x4060FFFF	64K	S_AXI	axi4lite_0	<input type="checkbox"/>
debug_module	C_BASEADDR	0x41400000	0x4140FFFF	64K	S_AXI	axi4lite_0	<input type="checkbox"/>

#### 2-1.2. 将 microblaze\_0\_d\_bram\_ctrl 和 microblaze\_0\_i\_bram\_ctrl 中 Size 栏的 8K 改为 64K



Instance	Base Name	Base Address	High Address	Size	Bus Interface(s)	Bus Name	Lock
microblaze_0's Address Map							
microblaze_0_d_bram_ctrl	C_BASEADDR	0x00000000	0x0000FFFF	64K	SLMB	microblaze_0_...	<input type="checkbox"/>
microblaze_0_i_bram_ctrl	C_BASEADDR	0x00000000	0x0000FFFF	64K	SLMB	microblaze_0_...	<input type="checkbox"/>
RS232	C_BASEADDR	0x40600000	0x4060FFFF	64K	S_AXI	axi4lite_0	<input type="checkbox"/>
debug_module	C_BASEADDR	0x41400000	0x4140FFFF	64K	S_AXI	axi4lite_0	<input type="checkbox"/>

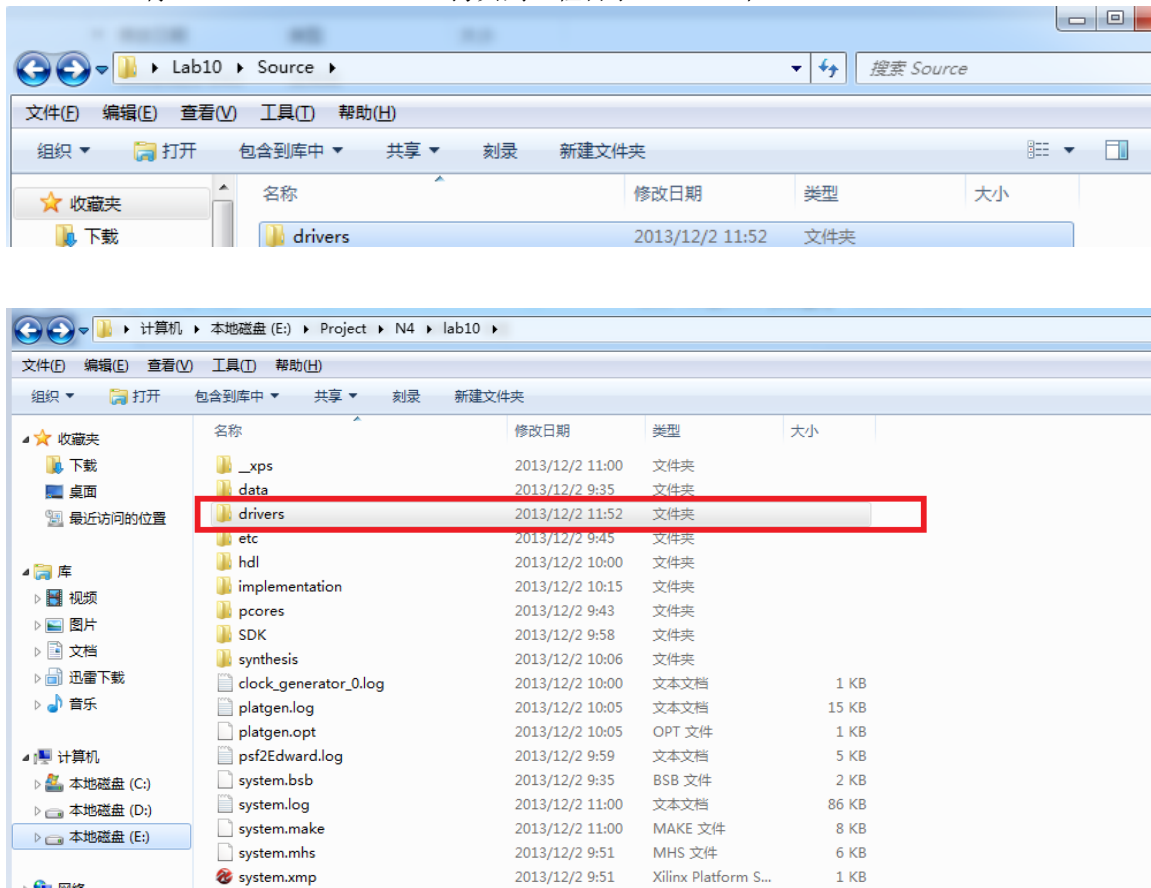
### 2-2.导入 VGA 的 IP 核，该 IP 核源代码与配置信息可在\*\*\lab10\Source 目录下找到

#### 2-2.1.将 \*\*\lab10\Source\ip\_vga\_v1\_00\_a 拷贝到工程目录 \*\*\lab10\pcores\下

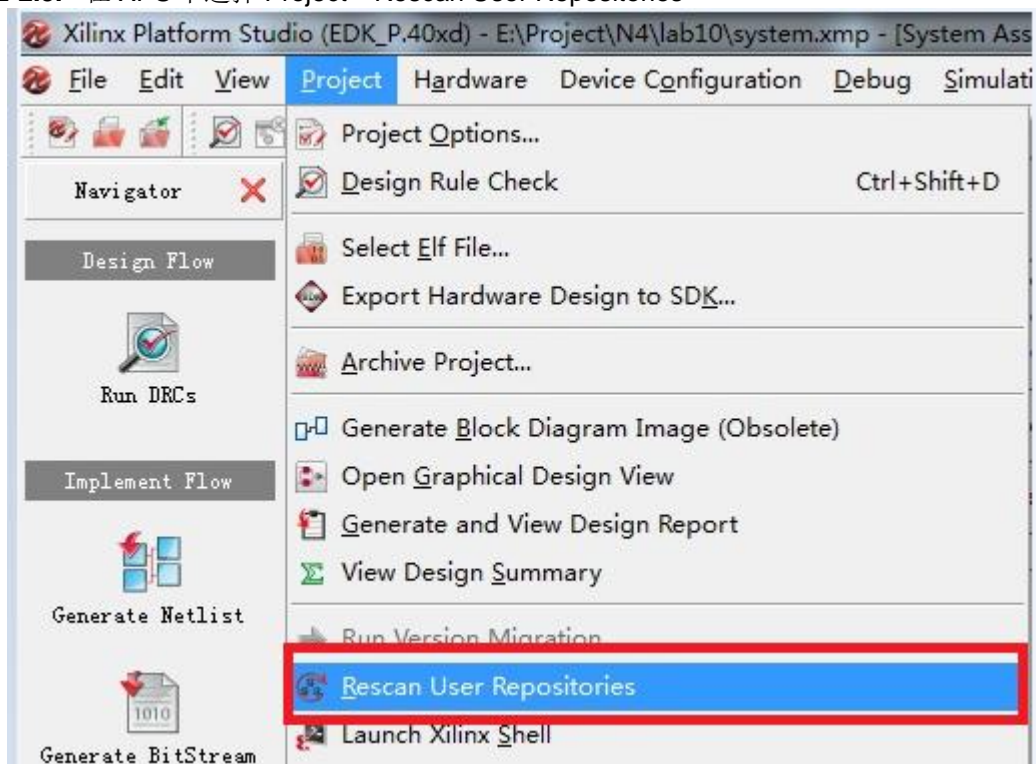




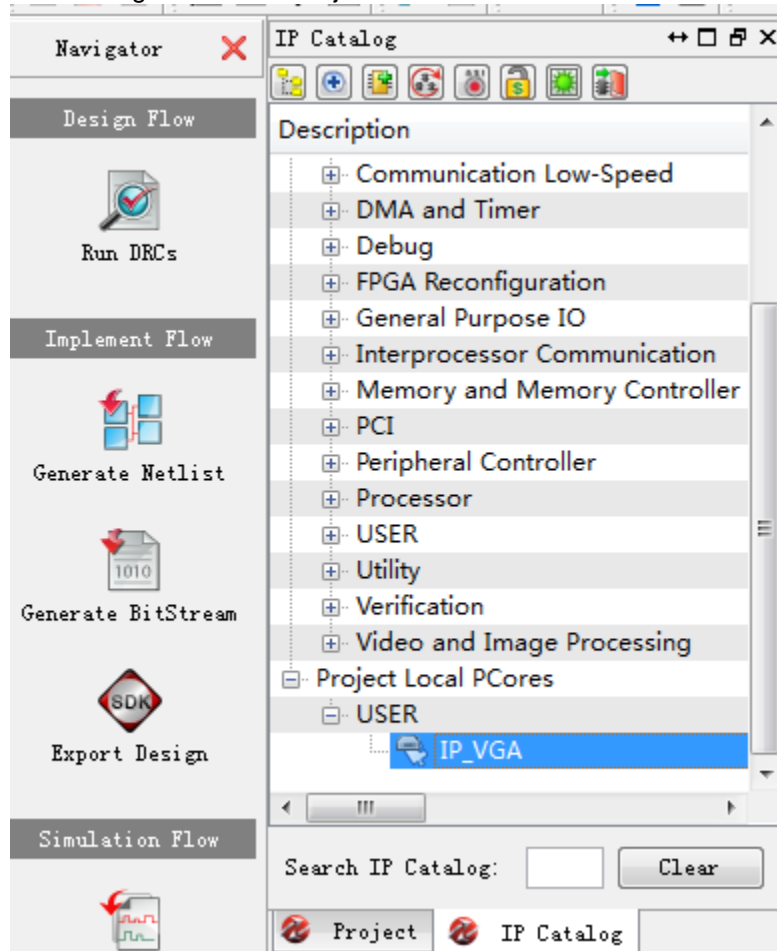
## 2-2.2. 将\*\*\lab10\Source\drivers 拷贝到工程目录 \*\*lab10\下



## 2-2.3. 在 XPS 下选择 Project—Rescan User Repositories

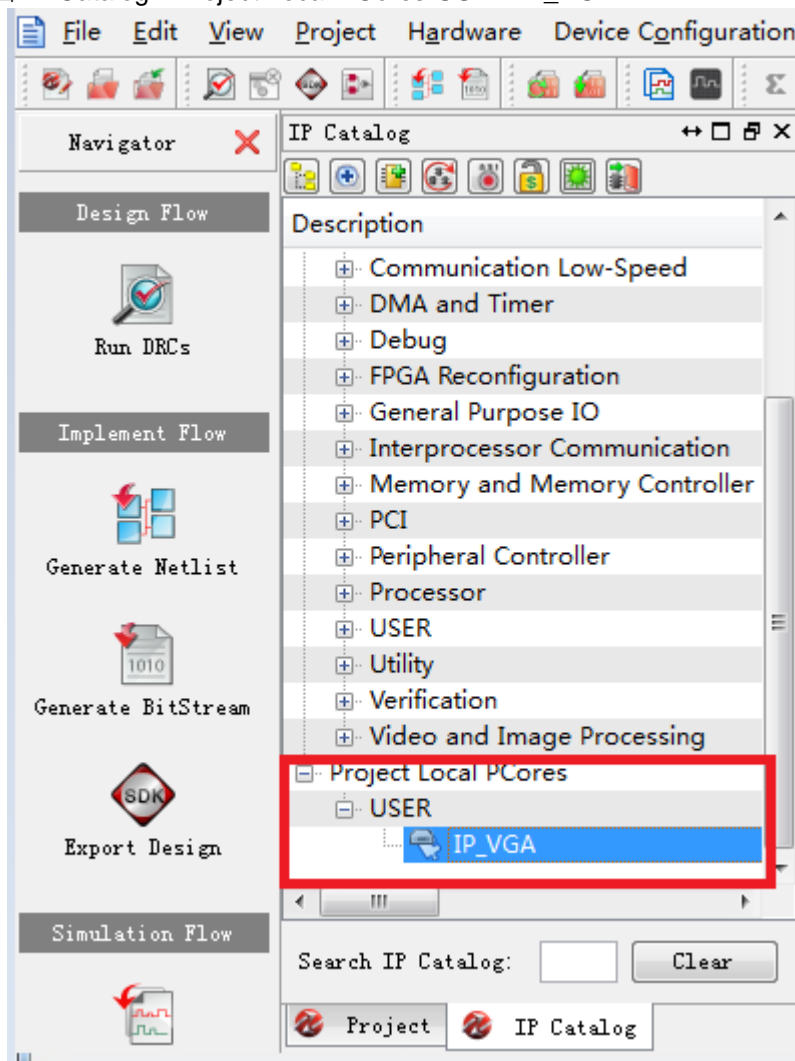


**2-2.4.** 选择 IP Catalog 可以看到在 Project Local PCores 中添加了刚刚导入的 IP 核 IP\_VGA

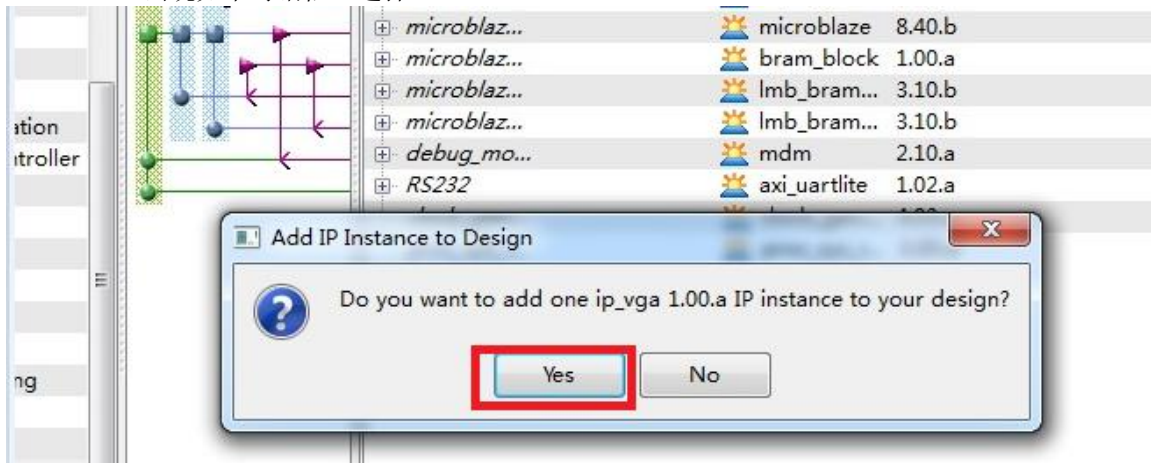


## 2-3. 添加 VGA IP 核到工程中

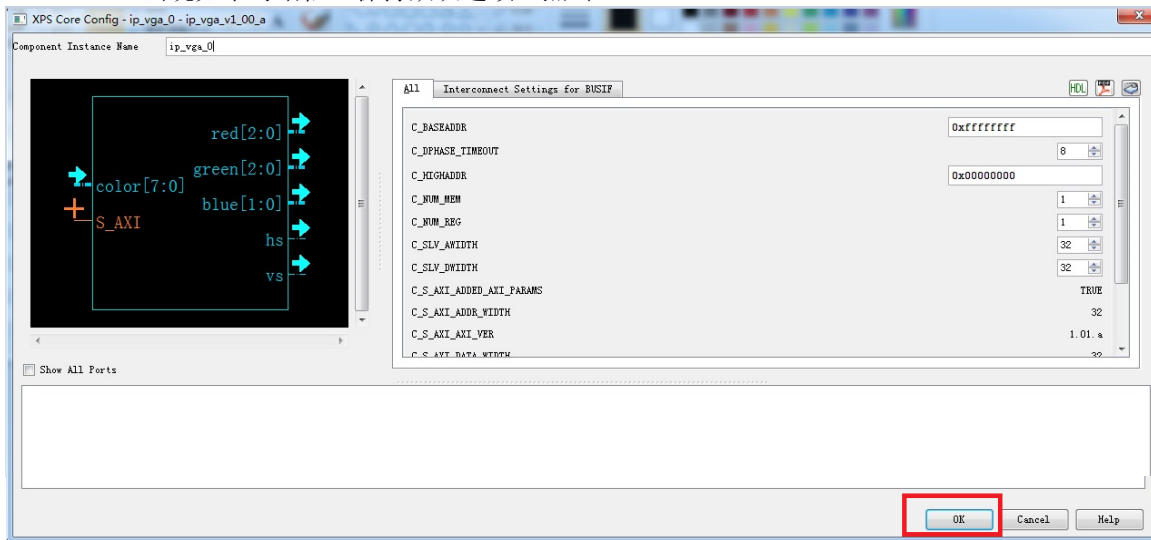
### 2-3.1. 双击 IP Catalog—Project Local PCores—USER—IP\_VGA



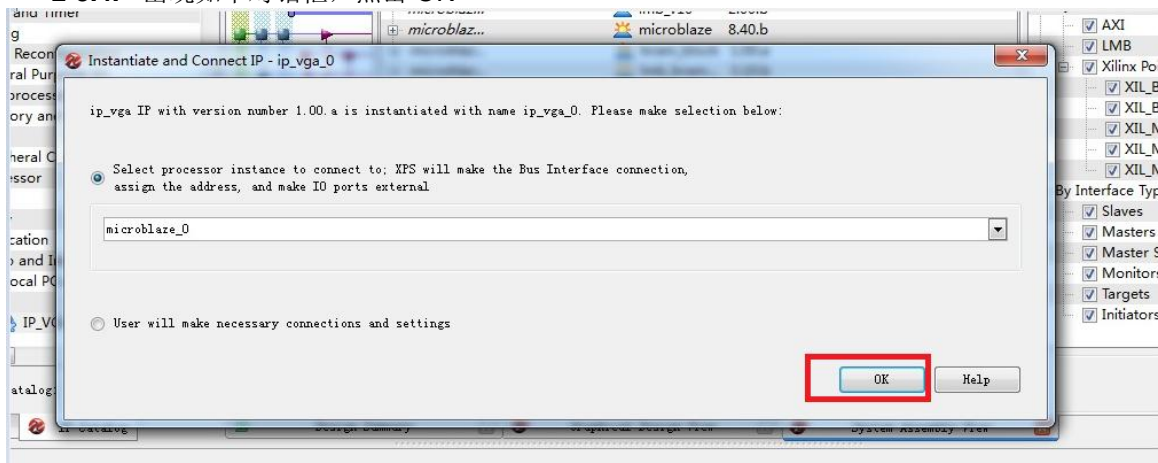
### 2-3.2. 出现如下对话框，选择 Yes



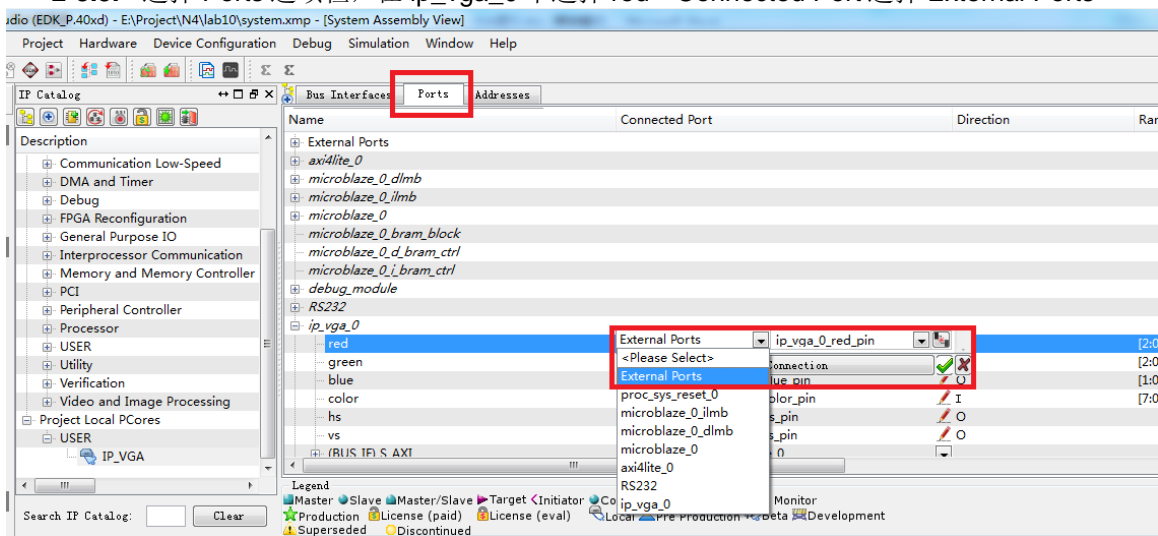
### 2-3.3. 出现如下对话框，保持默认选项，点击 OK



### 2-3.4. 出现如下对话框，点击 OK



### 2-3.5. 选择 Ports 选项栏，在 ip\_vga\_0 中选择 red—Connected Port 选择 External Ports

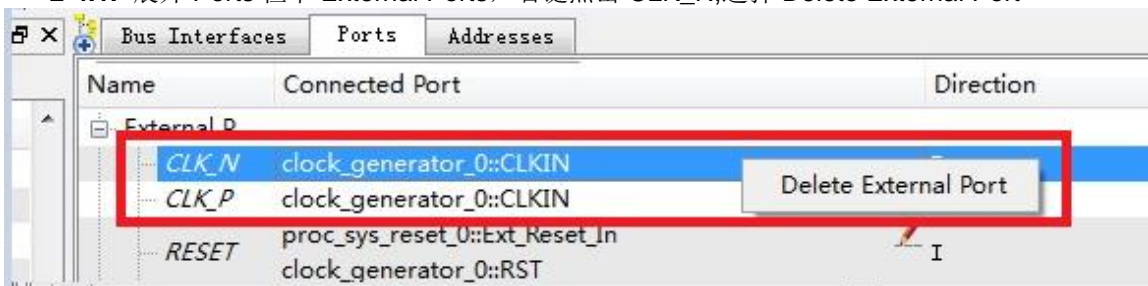


### 2-3.6. 重复 2-1.5 将 green,blue,color,hs,vs make External



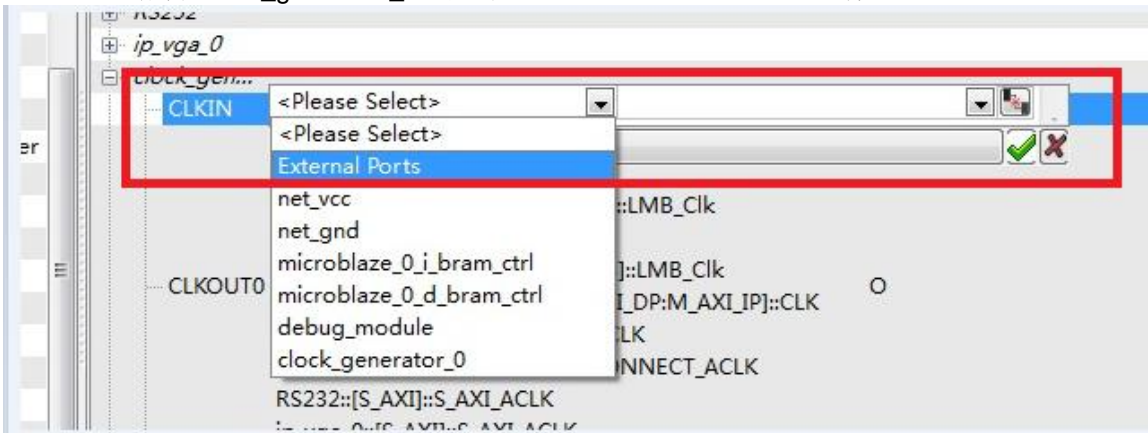
## 2-4. 修改系统时钟

### 2-4.1. 展开 Ports 栏中 External Ports, 右键点击 CLK\_N, 选择 Delete External Port

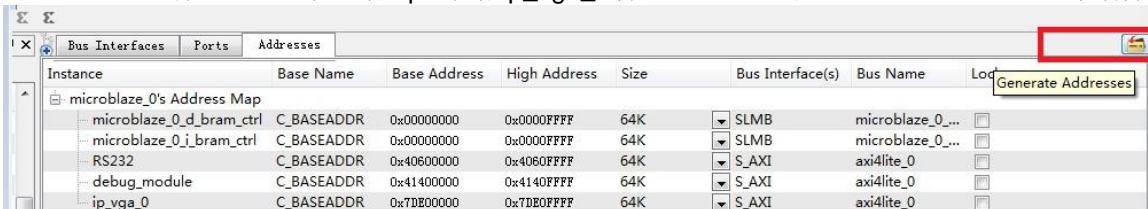


### 2-4.2. 重复 2-2.1 将 CLK\_P delete external port

### 2-4.3. 展开 clock\_generator\_0, 点击 CLKIN--Connected Port, 选择 External Ports



### 2-4.4. 选择 Address 栏, 若 xps 未给 ip\_vga\_0 分配地址, 点击 Generator Addresses 手动分配地址

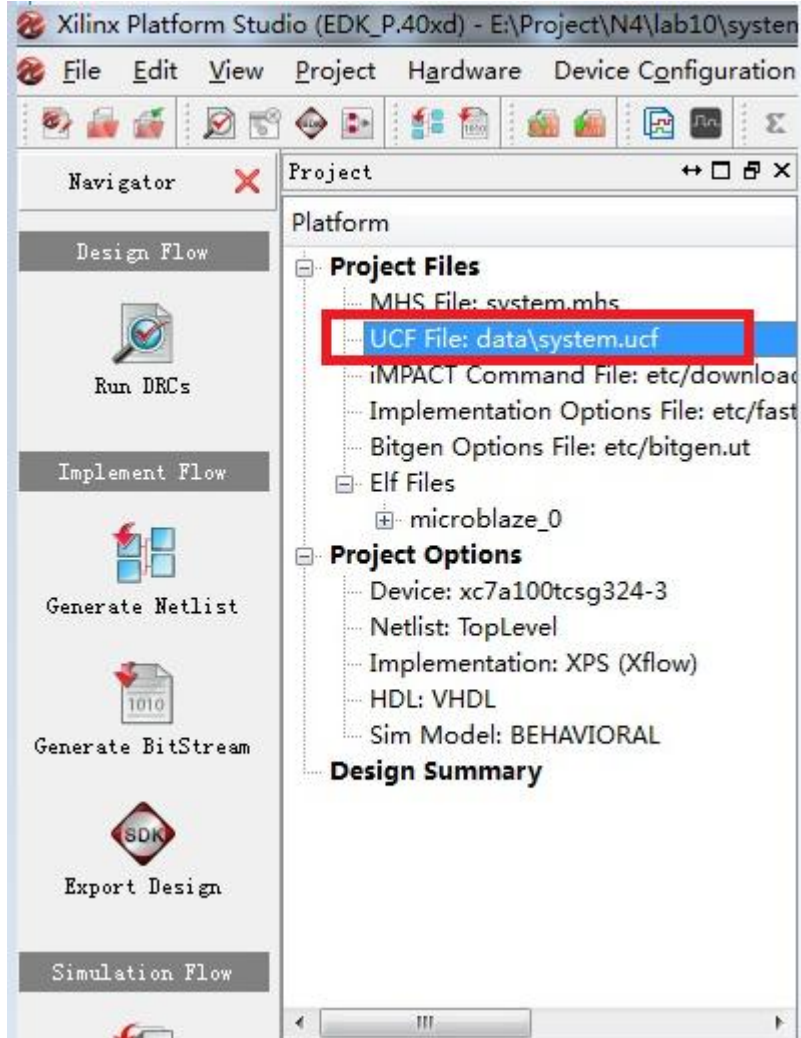




## 第三步 添加用户约束文件并将工程导入 SDK

### 3-1. 打开 UCF 文件，根据需求进行修改

#### 3-1.1. 选择 Project 栏，双击 UCF File: data\system.ucf



### 3-1.2. 这里手动输入 LOC（引脚位置）约束代码，保存

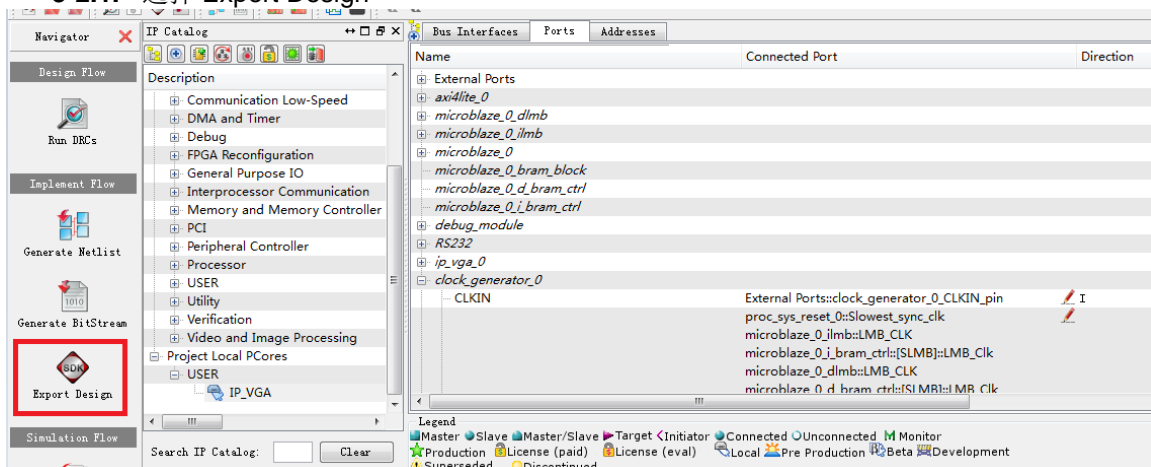
```

1  # Clock signal
2  NET "clock_generator_0_CLKIN_pin" LOC = "E3" | IOSTANDARD = "LVCMOS33";
3  NET "clock_generator_0_CLKIN_pin" TNM_NET = sys_clk_pin;
4  TIMESPEC TS_sys_clk_pin = PERIOD sys_clk_pin 100 MHz HIGH 50%;
5
6
7  NET "RESET" LOC = "R10" | IOSTANDARD = "LVCMOS33"; #Bank = 14, P
8
9
10 # USB-RS232 Interface
11 NET "RS232_Uart_1_sin" LOC = "C4" | IOSTANDARD = "LVCMOS33"; #
12 NET "RS232_Uart_1_sout" LOC = "D4" | IOSTANDARD = "LVCMOS33"; #
13
14
15 # Switches
16 NET "ip_vga_0_color_pin<0>" LOC = "U9" | IOSTANDARD = "LVCMOS33";
17 NET "ip_vga_0_color_pin<1>" LOC = "U8" | IOSTANDARD = "LVCMOS33";
18 NET "ip_vga_0_color_pin<2>" LOC = "R7" | IOSTANDARD = "LVCMOS33";
19 NET "ip_vga_0_color_pin<3>" LOC = "R6" | IOSTANDARD = "LVCMOS33";
20 NET "ip_vga_0_color_pin<4>" LOC = "R5" | IOSTANDARD = "LVCMOS33";
21 NET "ip_vga_0_color_pin<5>" LOC = "V7" | IOSTANDARD = "LVCMOS33";
22 NET "ip_vga_0_color_pin<6>" LOC = "V6" | IOSTANDARD = "LVCMOS33";
23 NET "ip_vga_0_color_pin<7>" LOC = "V5" | IOSTANDARD = "LVCMOS33";
24
25 # VGA Connector
26 NET "ip_vga_0_red_pin<0>" LOC = "B4" | IOSTANDARD = "LVCMOS33";
27 NET "ip_vga_0_red_pin<1>" LOC = "C5" | IOSTANDARD = "LVCMOS33";
28 NET "ip_vga_0_red_pin<2>" LOC = "A4" | IOSTANDARD = "LVCMOS33";
29 NET "ip_vga_0_blue_pin<1>" LOC = "D7" | IOSTANDARD = "LVCMOS33";
30 NET "ip_vga_0_blue_pin<2>" LOC = "D8" | IOSTANDARD = "LVCMOS33";
31 NET "ip_vga_0_green_pin<0>" LOC = "A5" | IOSTANDARD = "LVCMOS33";
32 NET "ip_vga_0_green_pin<1>" LOC = "B6" | IOSTANDARD = "LVCMOS33";
33 NET "ip_vga_0_green_pin<2>" LOC = "A6" | IOSTANDARD = "LVCMOS33";
34 NET "ip_vga_0_hs_pin" LOC = "B11" | IOSTANDARD = "LVCMOS33";
35 NET "ip_vga_0_vs_pin" LOC = "B12" | IOSTANDARD = "LVCMOS33";
36

```

## 3-2. 生成比特流，将工程导入 SDK

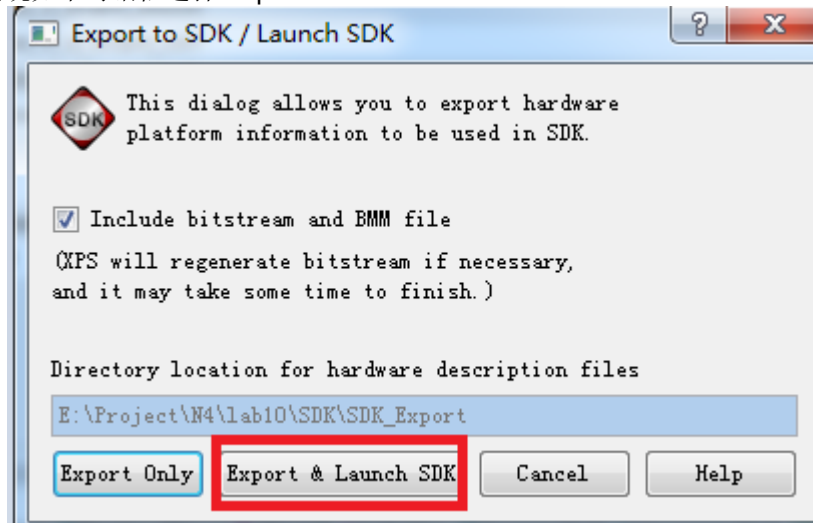
### 3-2.1. 选择 Export Design



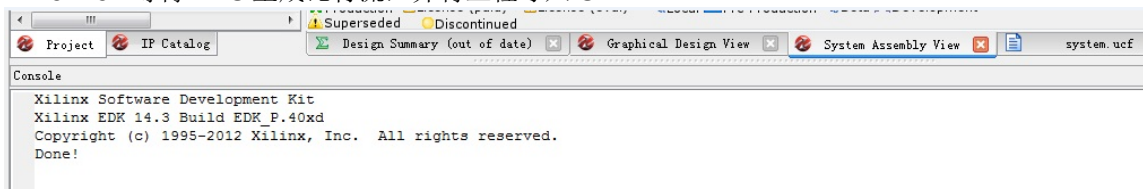
The screenshot shows the Vivado IDE interface. On the left, the 'Design Flow' pane is visible, with the 'Generate BitStream' section expanded. The 'Export Design' option is highlighted with a red box. The 'IP Catalog' pane shows a list of IP blocks, including 'Communication Low-Speed', 'DMA and Timer', 'Debug', 'FPGA Reconfiguration', 'General Purpose IO', 'Interprocessor Communication', 'Memory and Memory Controller', 'PCI', 'Peripheral Controller', 'Processor', 'USER', 'Utility', 'Verification', 'Video and Image Processing', 'Project Local PCores', and 'USER'. The 'Bus Interfaces' pane shows a list of connected ports and their directions, including 'External Ports', 'axi4lite\_0', 'microblaze\_0\_dlm', 'microblaze\_0\_ilmb', 'microblaze\_0', 'microblaze\_0\_bram\_block', 'microblaze\_0\_d\_bram\_ctrl', 'microblaze\_0\_i\_bram\_ctrl', 'debug\_module', 'RS232', 'ip\_vga\_0', 'clock\_generator\_0', and 'CLKIN'. The 'Ports' pane shows the connected ports and their directions, including 'External Ports::clock\_generator\_0\_CLKIN\_pin', 'proc\_sys\_reset\_0::Slowest\_sync\_clk', 'microblaze\_0\_ilmb::LMB\_CLK', 'microblaze\_0\_i\_bram\_ctrl::[SLMB]::LMB\_Clk', 'microblaze\_0\_dlm::LMB\_CLK', and 'microblaze\_0\_d\_bram\_ctrl::[SLMB]::LMB\_Clk'.



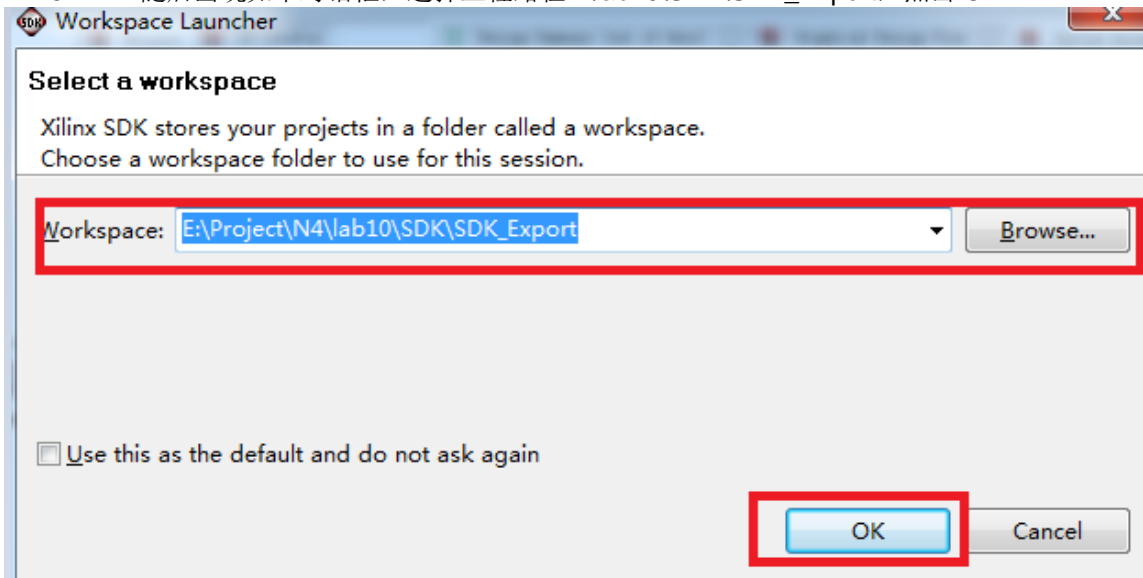
### 3-2.2. 出现如下对话框选择 Export & Launch SDK



### 3-2.3. 等待 XPS 生成比特流，并将工程导入 SDK



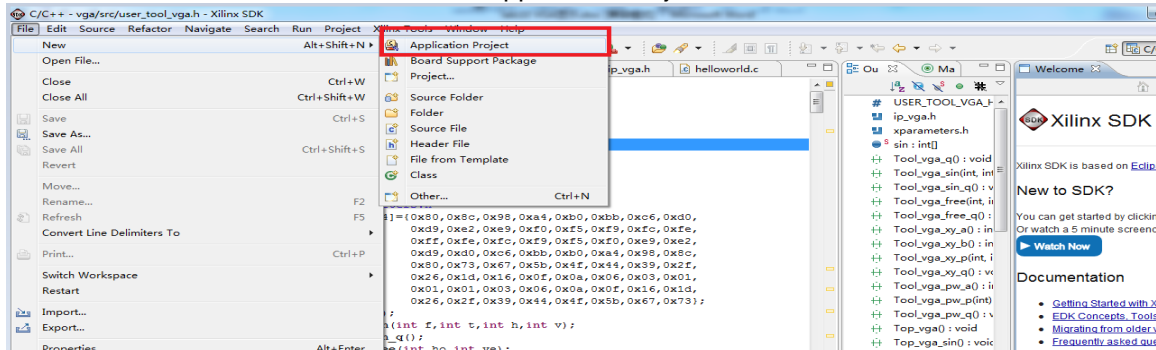
### 3-2.4. 随后出现如下对话框，选择工程路径\*\*\lab10\SDK\SDK\_Export，点击 OK



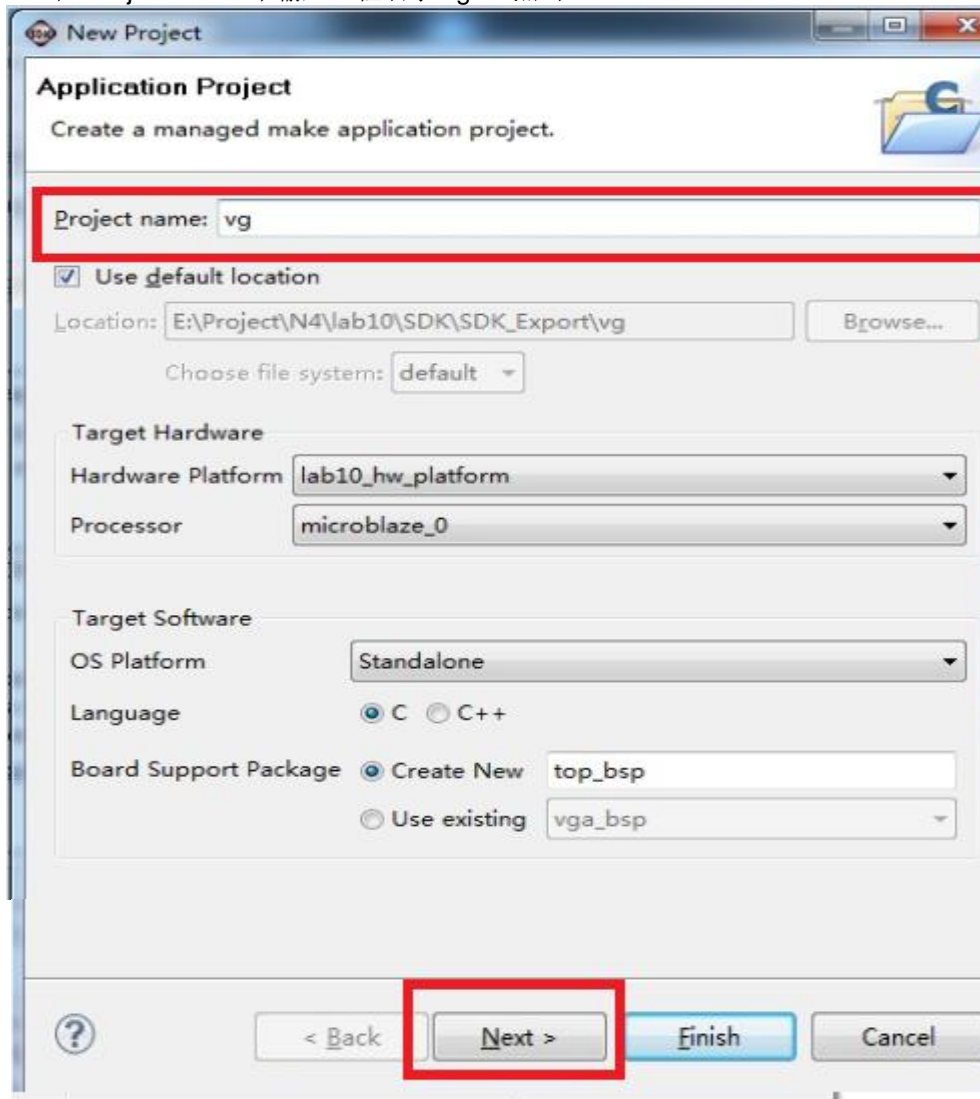
## 第四步 添加 APP

### 4-1. 创建新的工程

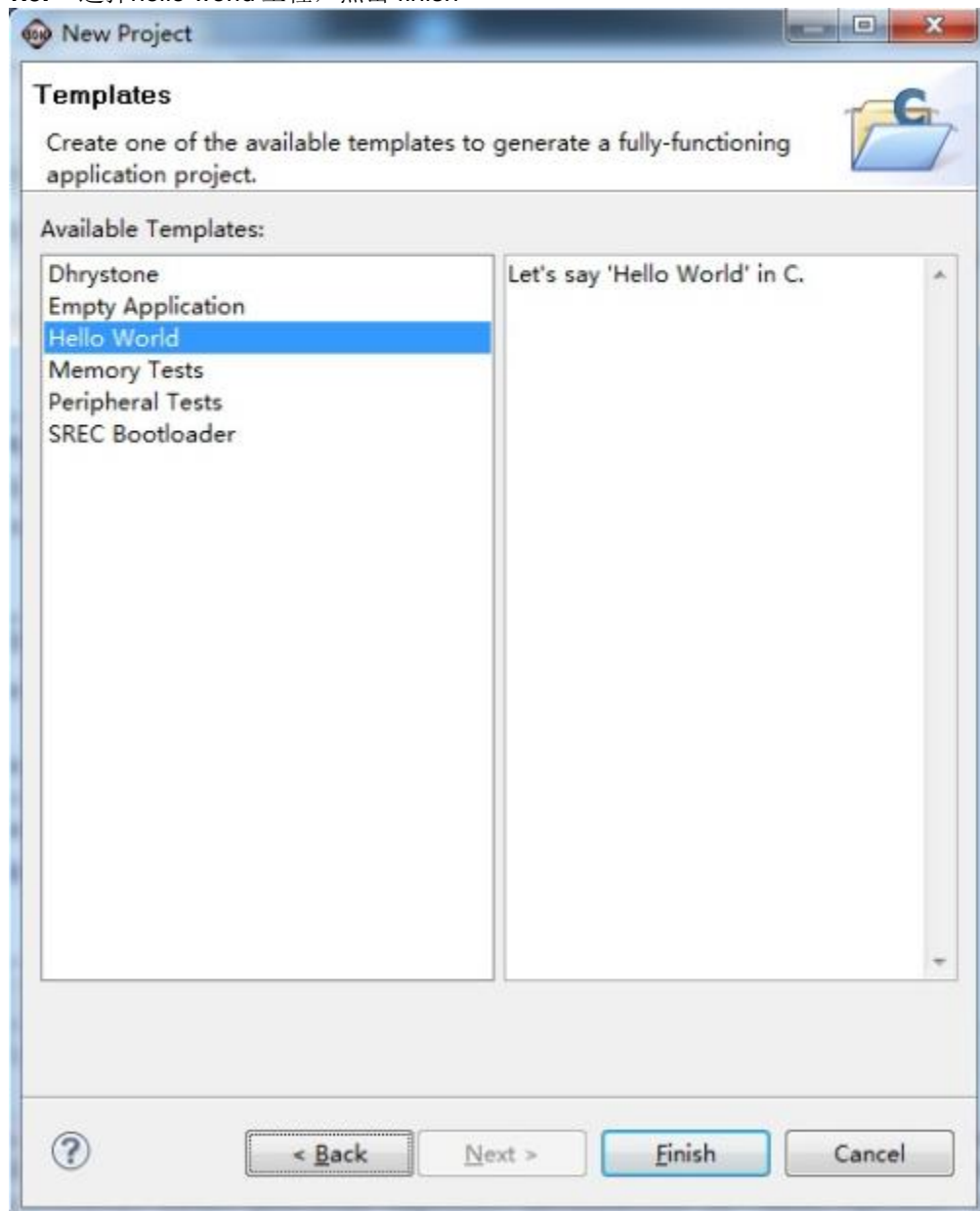
#### 4-1.1. 在 SDK 中选择 File—New—Application Project



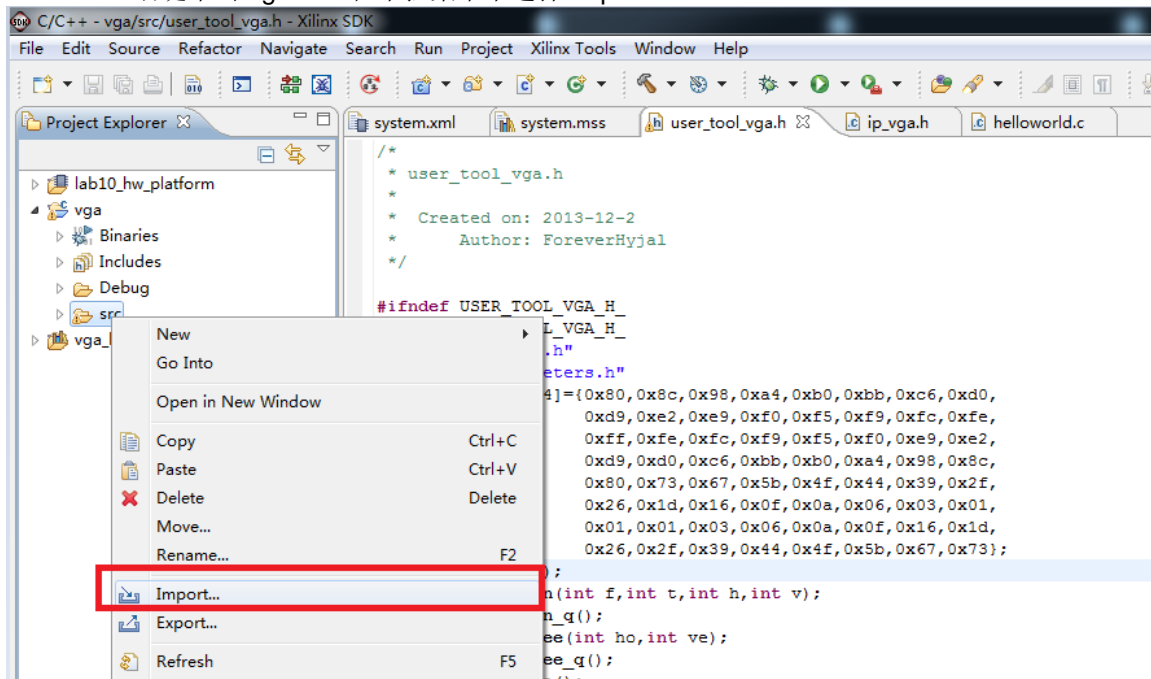
#### 4-1.2. 在 Project name 中输入工程名字 vga，点击 next



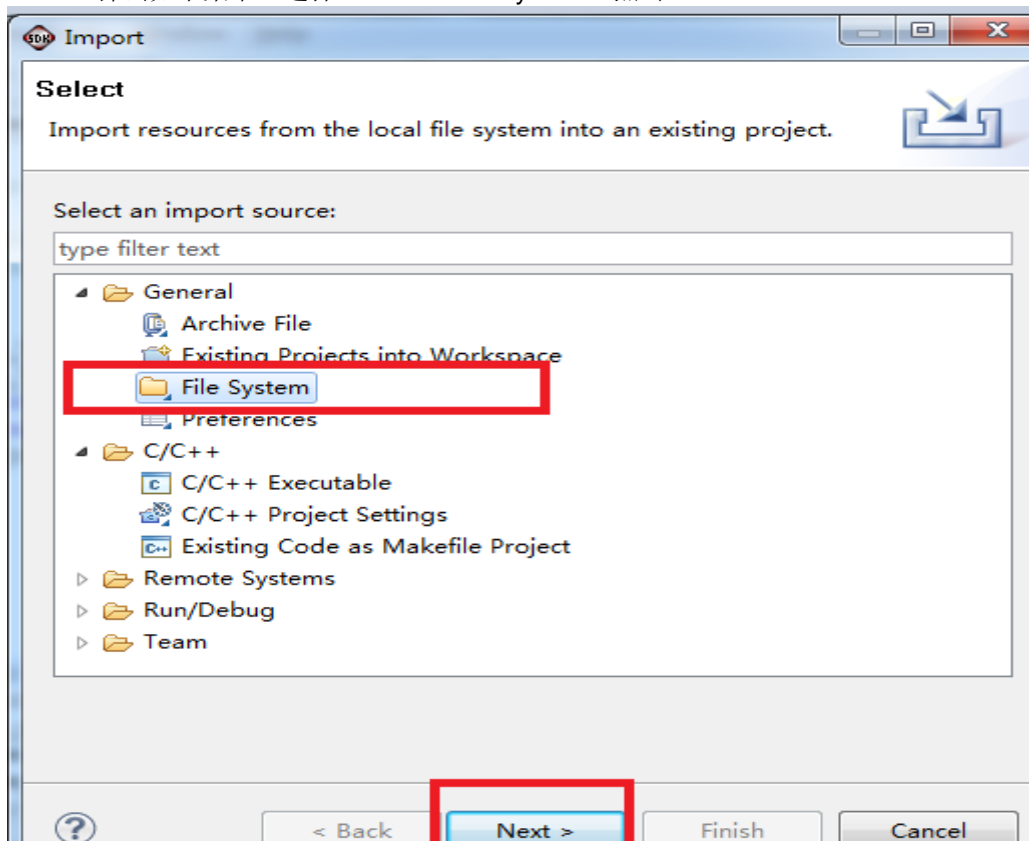
4-1.3. 选择hello world 工程， 点击 finish



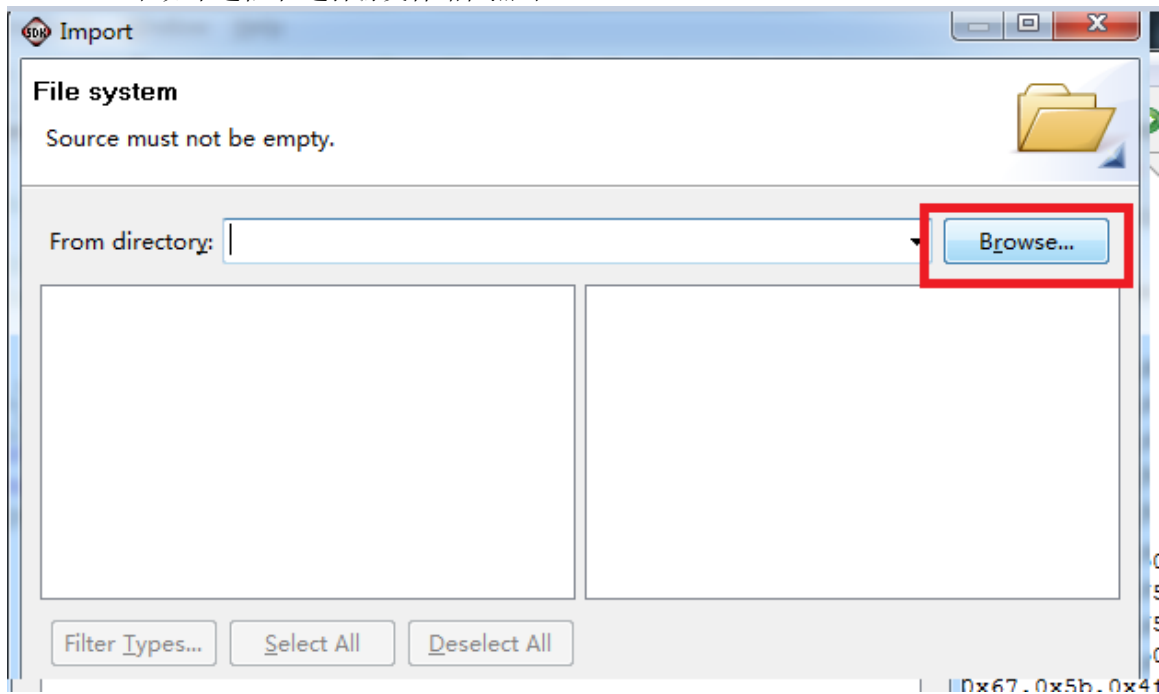
#### 4-1.4. 右键单击 vga-src 在下拉菜单中选择 Import



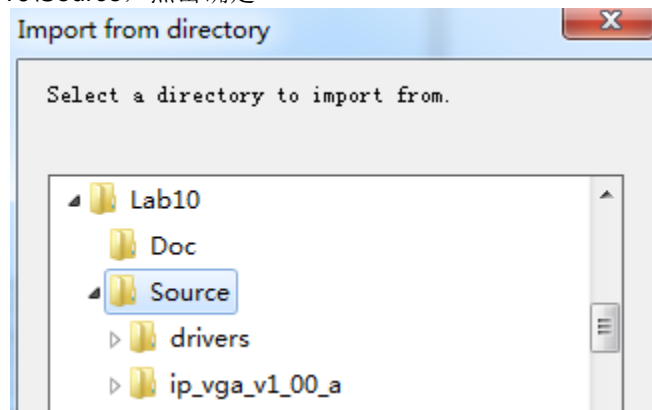
#### 4-1.5. 弹出如下菜单，选择 General-File System，点击 Next



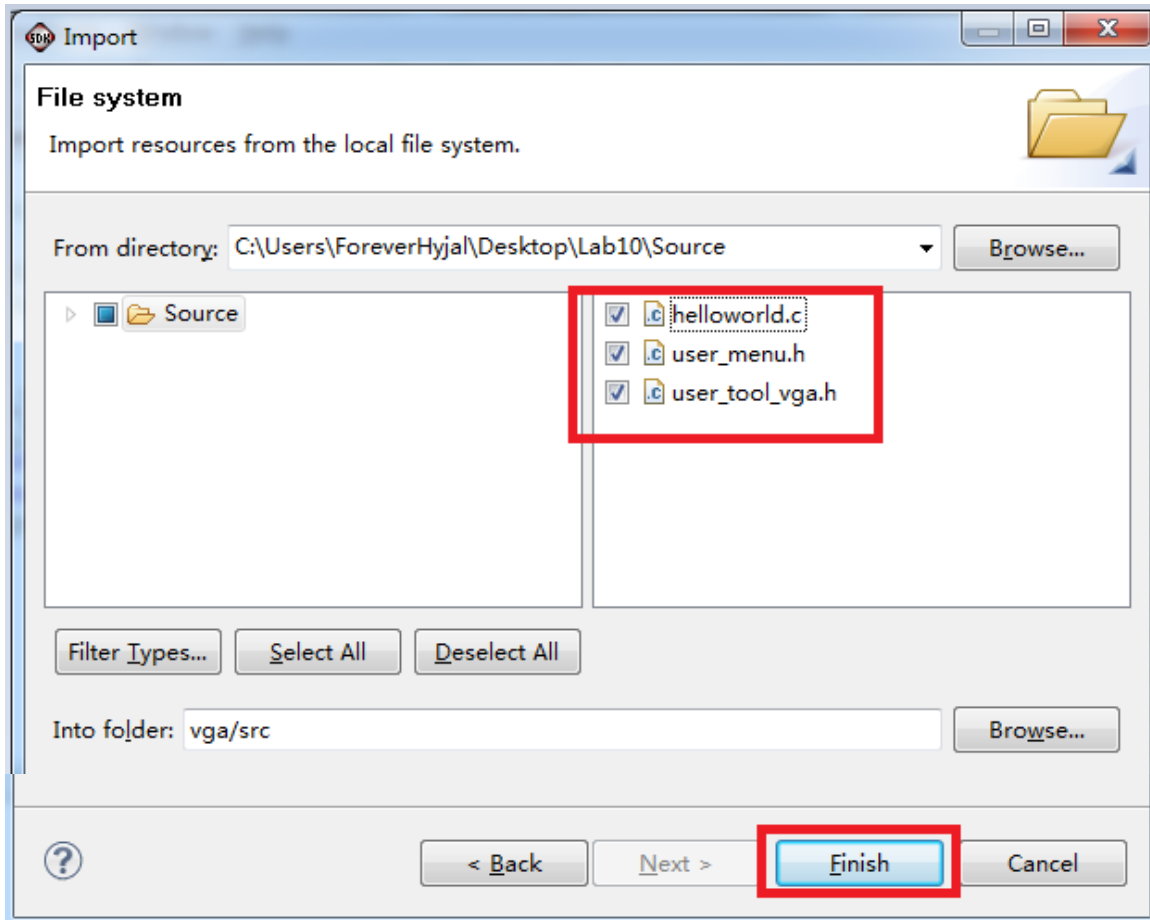
4-1.6. 在如下选框中 选择源文件路径点击 Browse



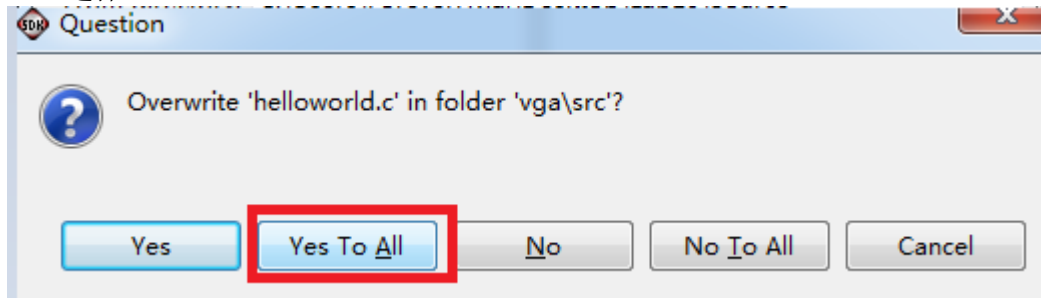
4-1.7. 选择\*\*\lab10\Source, 点击确定



#### 4-1.8. 勾选 3 个 C 文件，点击 Finish

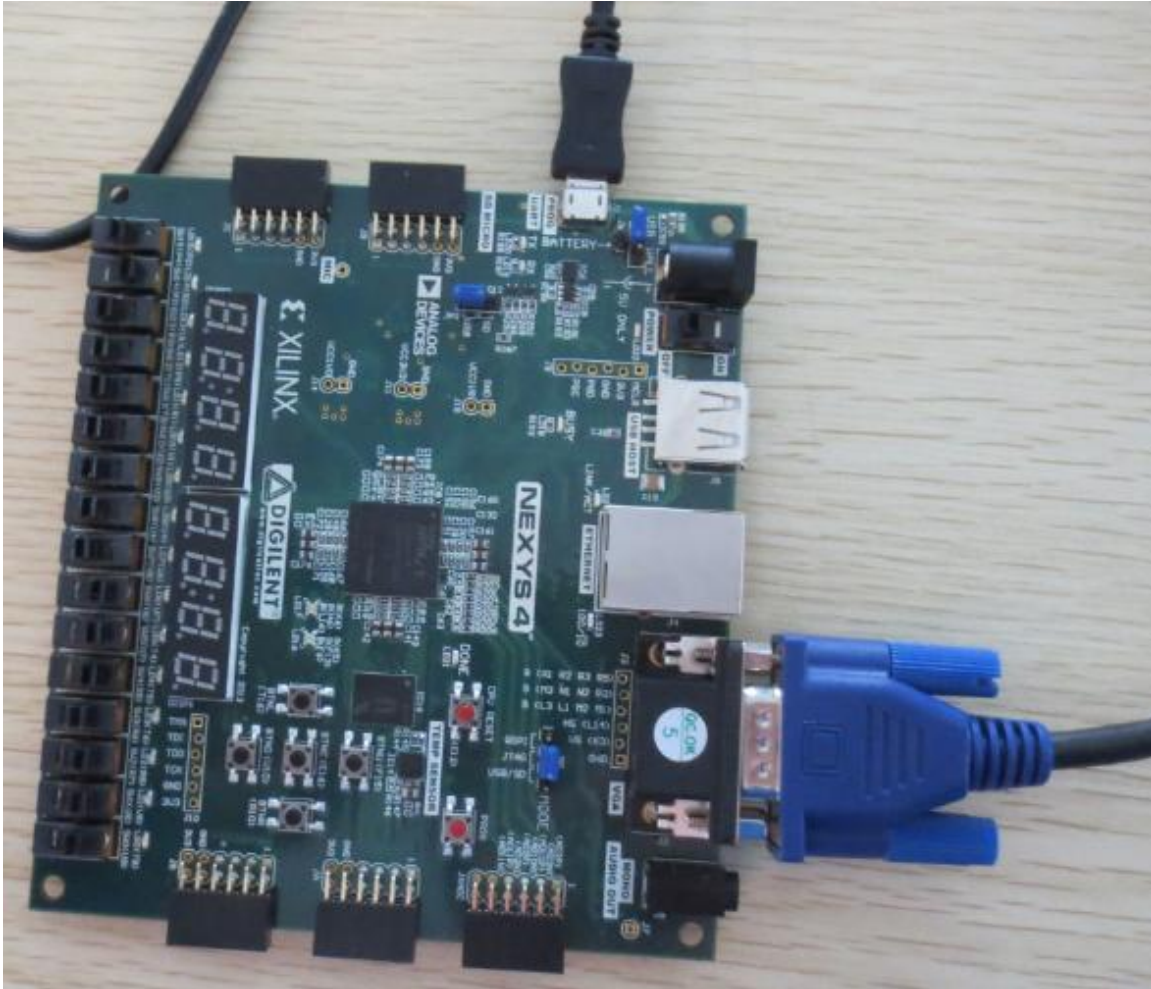


#### 4-1.9. 选择 Yes to All



## 第五步 上板验证

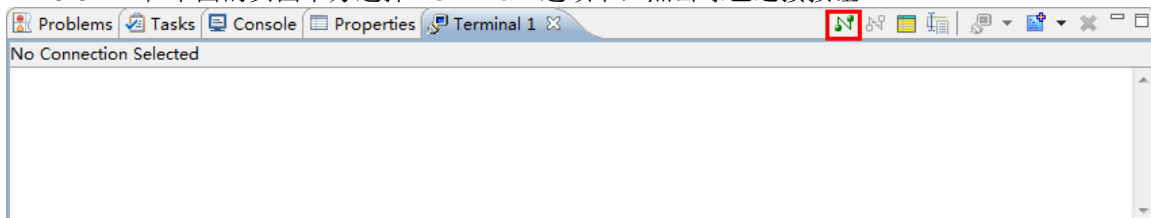
### 5-1. 将 VGA 显示器的 VGA 接口与 Nexys4 上的 VGA 端口链接



### 5-2. 将 Nexys4 与 PC 的 USB 接口连接

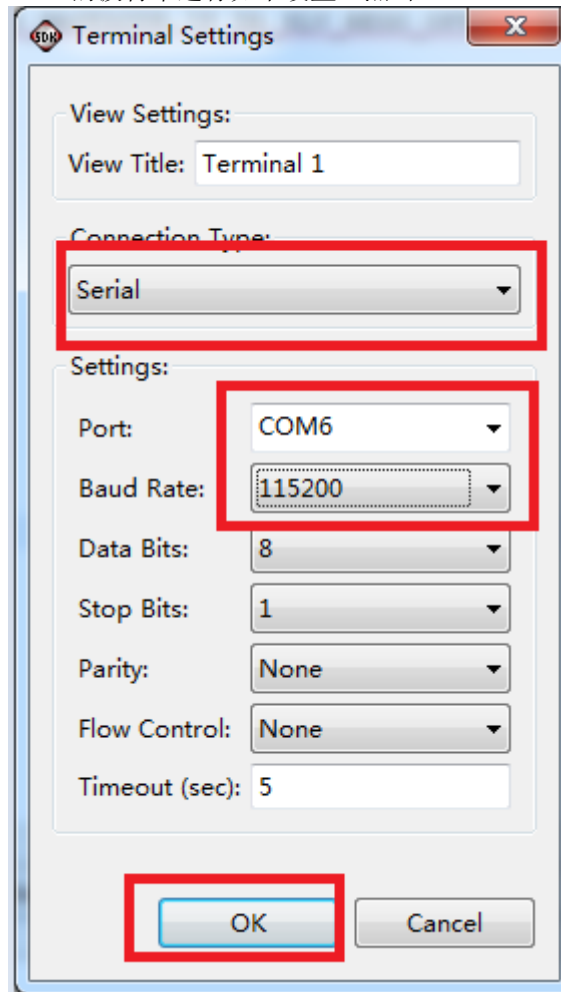
### 5-3. 在 SDK 中打开串口

#### 5-3.1. 在下面的页面下方选择 Terminal1 选项卡，点击绿色连接按钮



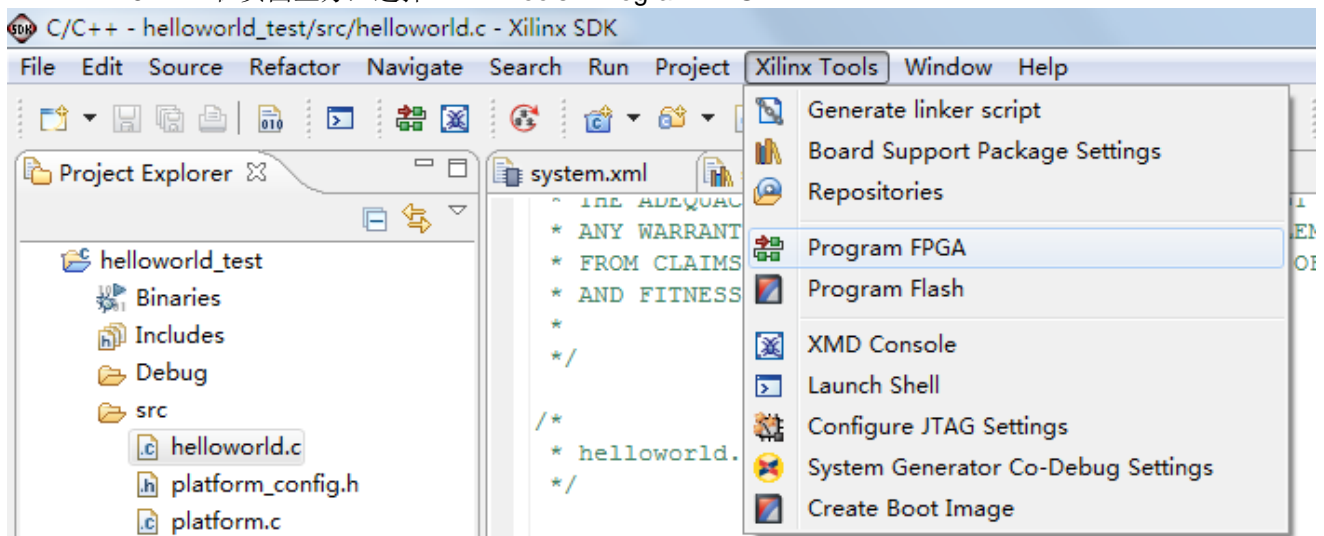


5-3.2. 按照端口号和 XPS 的波特率进行如下设置，点击 OK

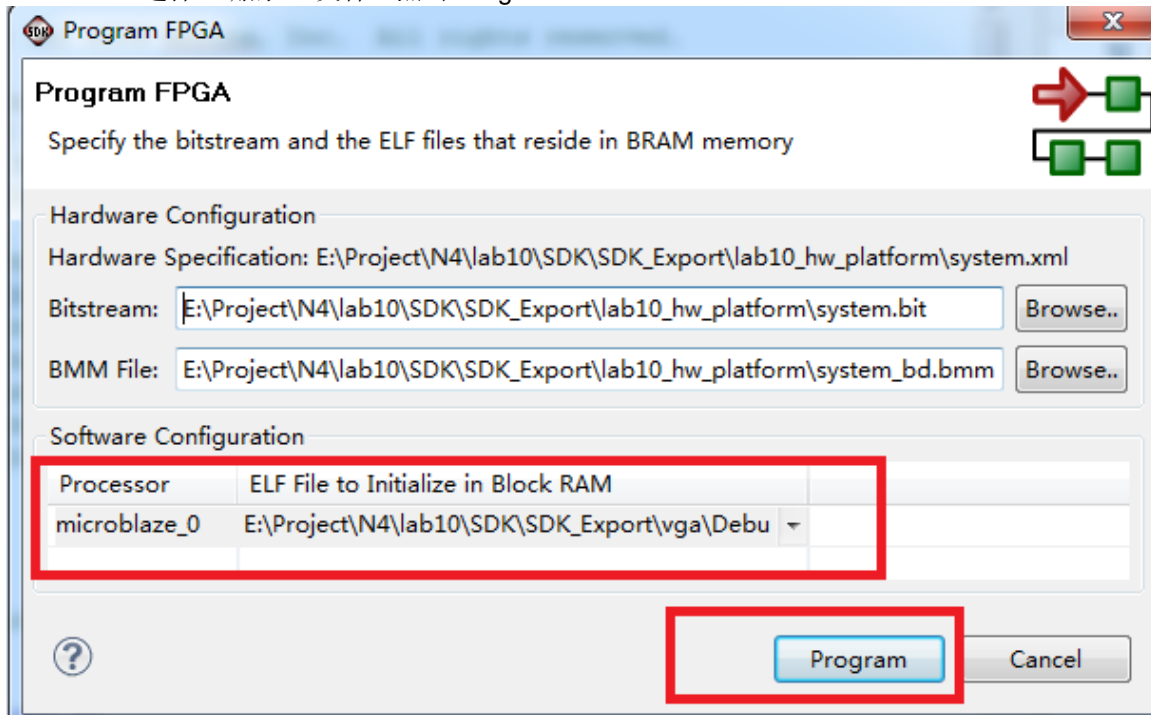


5-4. 将程序下载到板上并运行

5-4.1. 在页面上方，选择 Xilinx Tools –Program FPGA

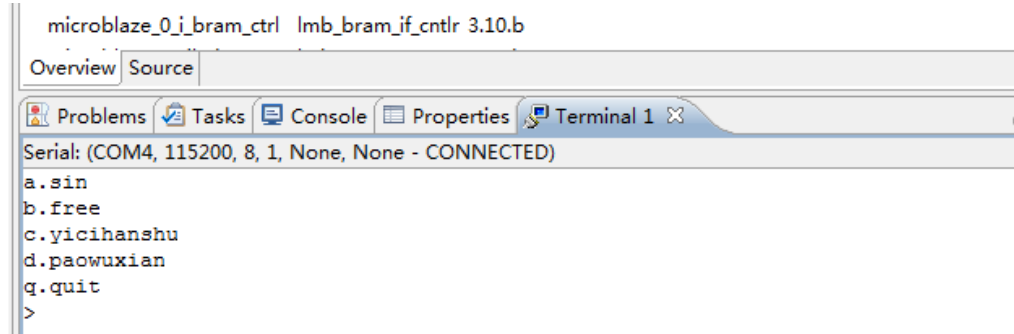


#### 5-4.2. 选择正确的 elf 文件，点击 Program



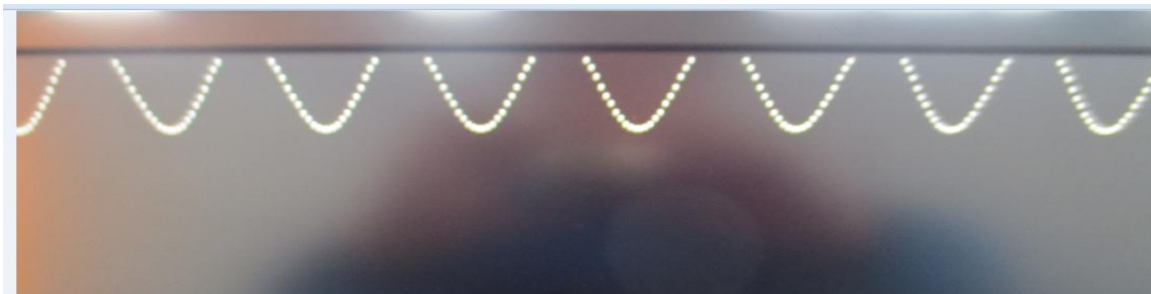
### 5-5. 查看实验结果

#### 5-5.1. 选择串口栏，查看串口信息

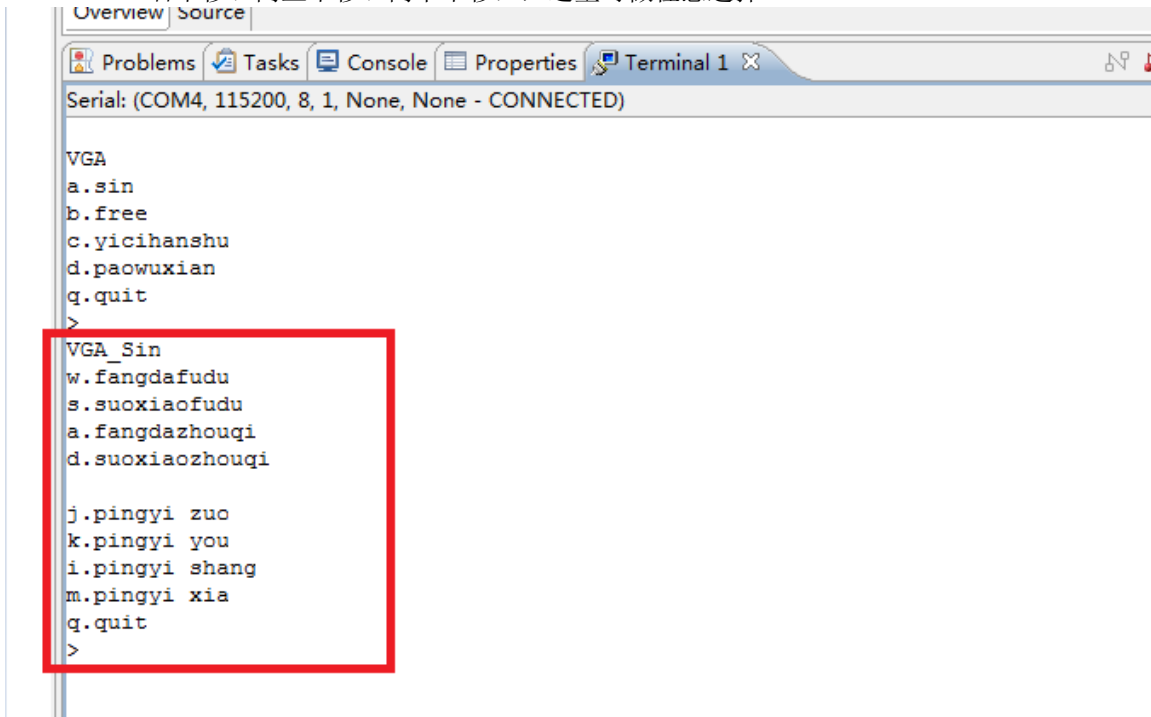


#### 5-5.2. 根据串口信息选择各功能（三角函数、自由画点、一次函数、抛物线、退出）。这里选择 a

#### 5-5.3. 查看 VGA 显示屏结果。若显示屏上无图像显示，拨动 Nexys4 上开关 SW0-SW7，选择图像颜色，这里选择 8 位全部置 1，图像显示白色



**5-5.4.** 查看串口输出，查看功能（依次为放大幅度、缩小幅度、放大周期、缩小周期、向左平移、向右平移、向上平移、向下平移），这里可做任意选择



```
Serial: (COM4, 115200, 8, 1, None, None - CONNECTED)

VGA
a.sin
b.free
c.yicihanshu
d.paowuxian
q.quit
>
VGA_Sin
w.fangdafudu
s.suoxiaofudu
a.fangdazhouqi
d.suoxiaozhouqi

j.pingyi zuo
k.pingyi you
i.pingyi shang
m.pingyi xia
q.quit
>
```

**5-5.5.** 查看结果，拨动开关，选择颜色



**5-5.6.** 按下 Nexys4 上按钮 BTNR 进行复位，查看串口，选择 d

```
j.pingyi zuo
k.pingyi you
i.pingyi shang
m.pingyi xia
q.quit
>
VGA
a.sin
b.free
c.yicihanshu
d.paowuxian
q.quit
>
```

**5-5.7.** 查看串口输出，查看功能（依次为）

```
>
VGA_paowuxian
a. V #jieshu
b.xianshi
q.quit
>
```

**5-5.8.** 选择 a 设置 初速度

```
VGA_paowuxian
a. V #jieshu
b.xianshi
q.quit
>
input V>
```

**5-5.9.** 输入 10 #，选择 b

**5-5.10.** 查看 VGA 显示器结果

