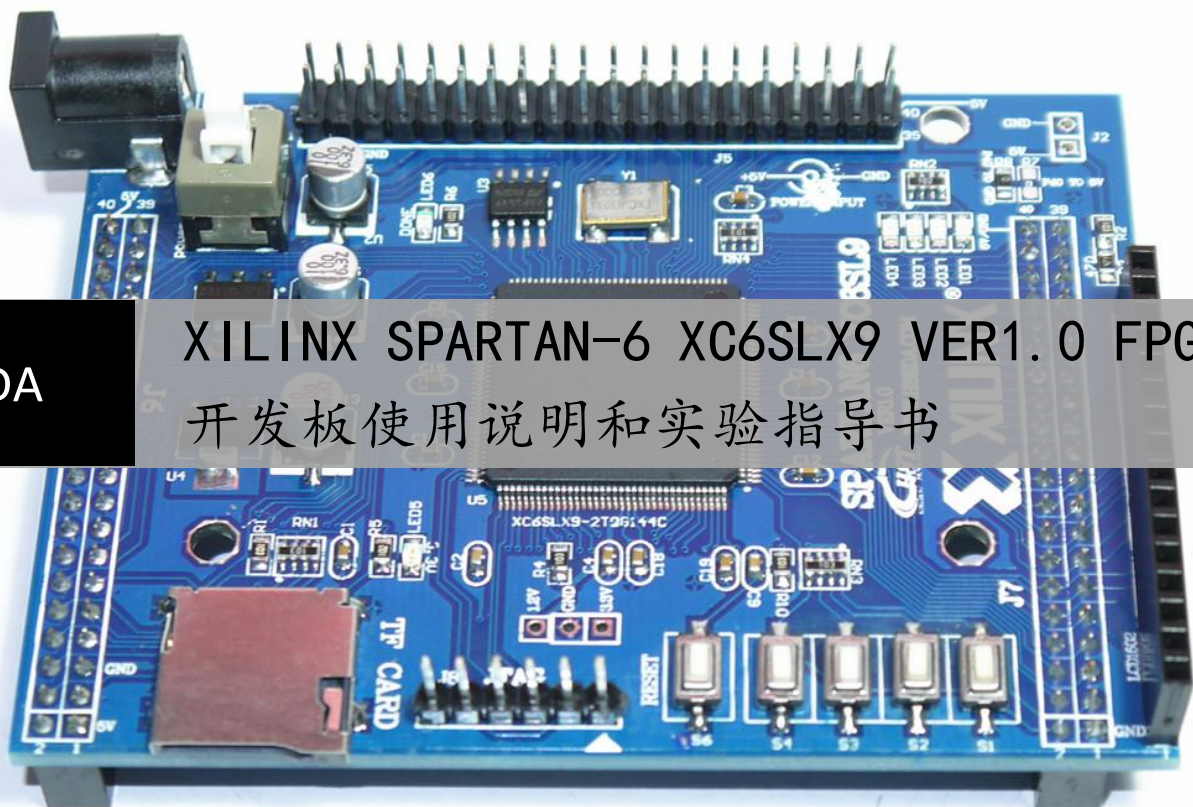


HSEDA

# XILINX SPARTAN-6 XC6SLX9 VER1.0 FPGA 开发板使用说明和实验指导书



## 目录

目录.....	0
一、系统组合结构图.....	5
核心板结构布局图.....	5
PCB 布局展示.....	5
核心板与部分扩展板组合结构展示.....	6
XC6SLX9 与 EDA6.1 主板组合成 EDA/SOPC 开发系统.....	6
二、系统实验效果图.....	7
三、配套光盘内容部分展示.....	7
四、XC6SLX9 VER1.0 FPGA 核心板资源.....	7
五、 配套软件和开发环境.....	7
六、 实验例程清单.....	7
一.单独核心板实验清单.....	7
1. 流水灯.....	7
3. 按键控制.....	7
4. LCD1602 液晶显示驱动.....	7
5. RS232 通信实验（要配合我们的 NET/USB/LCD7INCH 扩展板）.....	7
二、EDA6.1 主板配套实验.....	8
1.VHDL 实验代码清单:.....	8
2.verilog 实验代码清单:.....	8
七、发货清单（标准配置）.....	8
八.可选配件.....	9
九、电路分析以及说明.....	9
1.说明.....	9
1.电源和供电.....	9
2.发光二极管（LED）.....	10
3. 按键.....	10
5.LCD 接口(J4 扩展 I/O).....	11
4. TF 卡插槽.....	11
6. 高速，同步 SDRAM.....	12
8.串行 FLASH（配置芯片）.....	14
9.JTAG 编程调试接口.....	14
10.时钟源.....	15
11.复位电路.....	15
十、原理图:.....	16
十一、实验指导.....	16
1.实验前准备:.....	16
2.EDA6.1 主板使用说明.....	16
3.EDA6.1 主板原理图.....	16
创建一个 EDA 实验的详细过程.....	17

实验目的.....	17
开发工具.....	17
操作系统.....	17
实验步骤.....	17
第一部分 VHDL 实验.....	26
实验一、I/O 口控制.....	26
一、实验目的.....	26
二、实验过程.....	26
三、实验结果.....	26
实验二、分频器（VHDL）.....	27
一、实验目的.....	27
二、实验过程.....	27
三、实验结果.....	27
实验三、流水灯.....	27
一、实验目的.....	27
二、实验过程.....	27
三、实验结果.....	27
实验四、3_8 译码器.....	27
一、实验目的.....	27
二、实验过程.....	27
三、实验结果.....	27
实验五、移位运算器.....	27
一、实验目的.....	27
二、实验过程.....	27
三、实验结果.....	28
实验六、RAM.....	28
一、实验目的.....	28
二、实验过程.....	28
三、实验结果.....	29
实验七、UART（VHDL）.....	29
一、实验目的.....	29
二、实验过程.....	29
三、实验结果.....	29
实验八、PS2 鼠标.....	29
一、实验目的.....	29
二、实验过程.....	29
三、实验结果.....	30
实验九、LCD1602.....	30
一、实验目的.....	30
二、实验过程.....	30
三、实验结果.....	30
实验十、LCD12864 并行输入（VHDL）.....	30
一、实验目的.....	30
二、实验过程.....	30
三、实验结果.....	30
第二部分：VERILOG 实验.....	31
实验一、比较器.....	31
一、实验目的.....	31

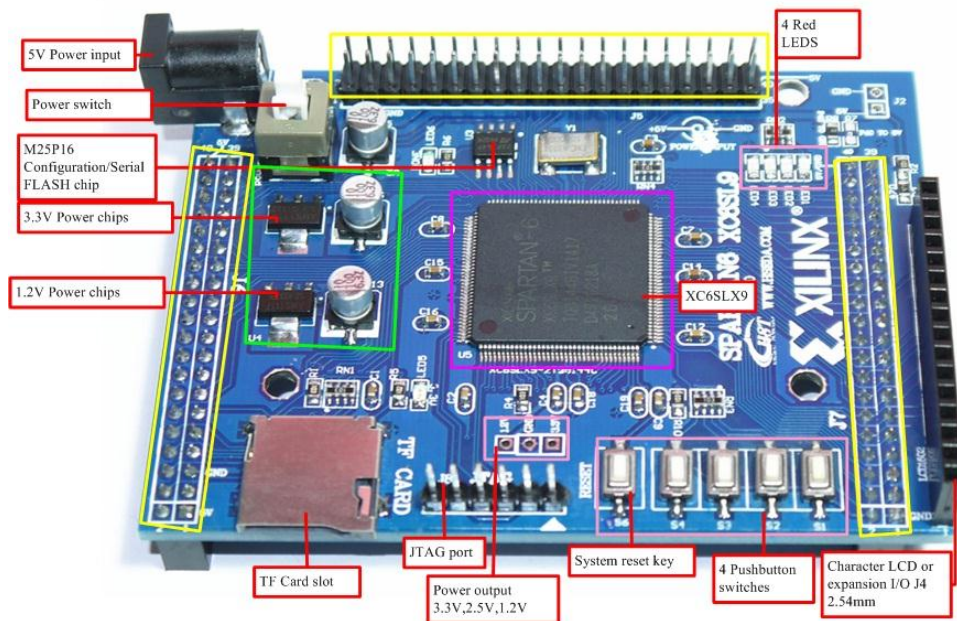
二、实验过程.....	31
三、实验结果.....	31
实验二、4bit 加法器.....	31
一、实验目的.....	31
二、实验过程.....	31
三、实验结果.....	31
实验三、多路选择器.....	31
一、实验目的.....	31
二、实验过程.....	31
三、实验结果.....	31
实验四、2、4、8 分频器 (Verilog) .....	32
一、实验目的.....	32
二、实验过程.....	32
三、实验结果.....	32
实验五、数码管.....	32
一、实验目的.....	32
二、实验过程.....	32
三、实验结果.....	32
实验六、HEX 转BCD 码.....	32
一、实验目的.....	32
二、实验过程.....	32
三、实验结果.....	33
实验七、计数器.....	33
一、实验目的.....	33
二、实验过程.....	33
三、实验结果.....	33
实验八、并行输入串行输出.....	33
一、实验目的.....	33
二、实验过程.....	33
三、实验结果.....	33
实验九、数字时钟.....	34
一、实验目的.....	34
二、实验过程.....	34
三、实验结果.....	34
实验十、IR.....	34
一、实验目的.....	34
二、实验过程.....	34
三、实验结果.....	34
实验十一、ROM.....	34
一、实验目的.....	34
二、实验过程.....	34
三、实验结果.....	35
实验十二、PWM.....	35
一、实验目的.....	35
二、实验过程.....	35
三、实验结果.....	35
实验十三、MUSIC.....	35
一、实验目的.....	35

二、实验过程.....	35
三、实验结果.....	35
实验十四、UART.....	35
一、实验目的.....	35
二、实验过程.....	35
三、实验结果.....	36
实验十五、PS2_UART.....	36
一、实验目的.....	36
二、实验过程.....	36
三、实验结果.....	36
实验十六、PS2_LCD.....	37
一、实验目的.....	37
二、实验过程.....	37
三、实验结果.....	37
实验十七、LCD12864 并行输入 (Verilog).....	37
一、实验目的.....	37
二、实验过程.....	37
三、实验结果.....	37
实验十八、串行 LCD12864.....	37
一、实验目的.....	37
二、实验过程.....	37
三、实验结果.....	37



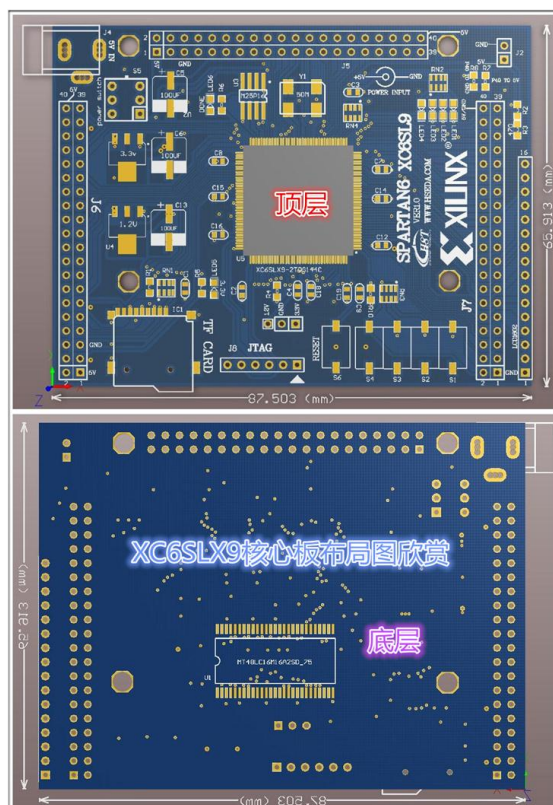
## 一、系统组合结构图

核心板结构布局图

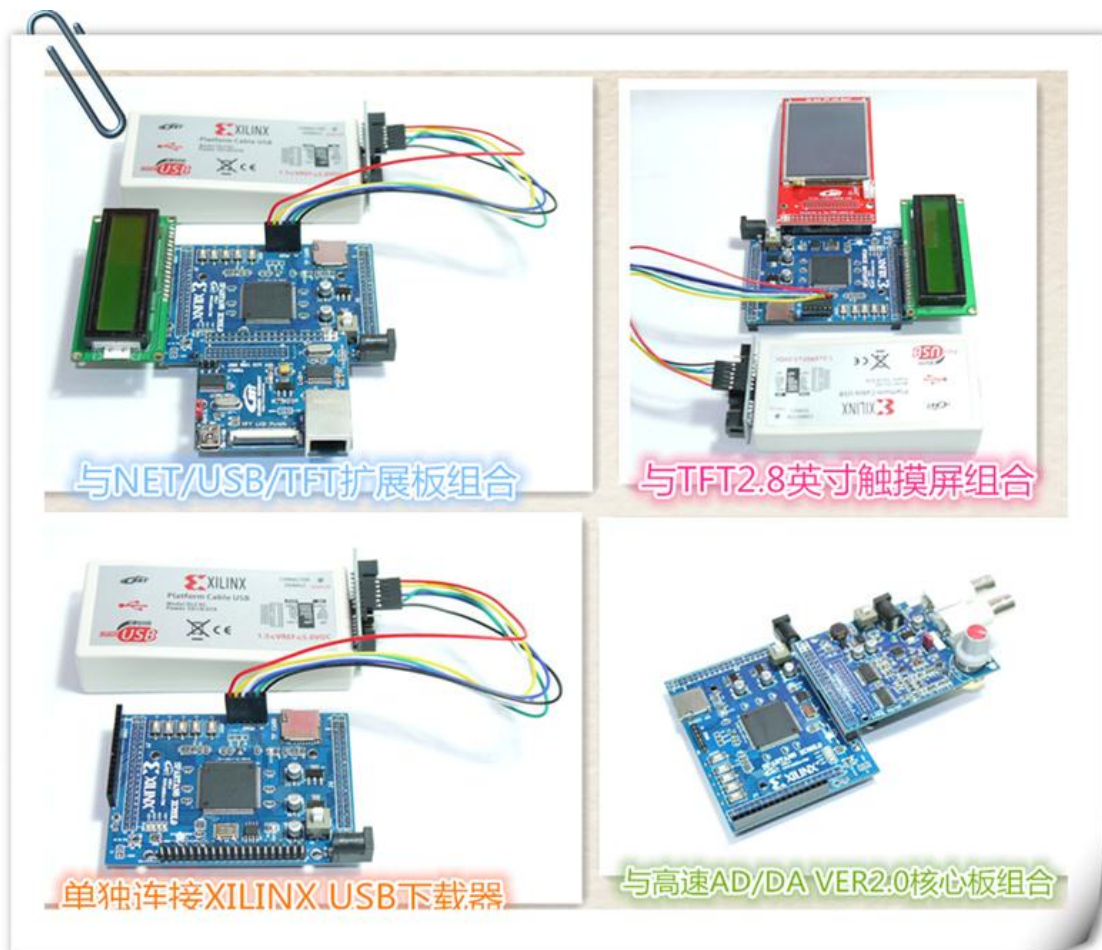


PCB 布局展示

(资深工程师手工布线，2层 PCB 设计)



## 核心板与部分扩展板组合结构展示



## XC6SLX9 与 EDA6.1 主板组合成 EDA/SOPC 开发系统



## 二、系统实验效果图

## 三、配套光盘内容部分展示

## 四、XC6SLX9 VER1.0 FPGA 核心板资源

XILINX SPARTAN-6 XC6SLX9 VER1.0 核心板系统资源

外设	数量	说明
FPGA	1	FPGA 芯片型号: xc6slx9tqg144, 逻辑单元 : 9152, 乘法器: 16, RAM: 666 k
配置芯片	1	采用常用的 M25P16 16Mb 串行 FLASH, 保存配置数据和相关程序
SDRAM	1	一片 8M×16Bit SDRAM , 最高 143MHz 读写速度 128Mb,超大容量,我们已经预留了扩展 I/O, 可以直接扩展到 16M X 16bit SDRAM (32M byte). 独立 I/O 控制
时钟	1	采用一个高精度 50M 有源晶振
LCD	1	一个标准 LCD1602 液晶接口, 方便用户使用。
按键	4	带上拉电位, 用户按键输入。
电源	2	3.3V/800MA, 1.2V/800MA, 并单独引出
JTAG 端口	1	调试 FPGA 或给板载串行 FLASH 编程
LED	4	4 个红色 LED
复位按键	1	系统复位按键, 带上拉。也可以做为用户按键输入。
TF 卡插槽	1	提供一个标准的 TF 卡插槽, 支持 FAT16 和 FAT32 位模式。可以保存数据到 TF 卡
扩展 I/O 口	3	3 个扩展 I/O 插座 (J5/37) (J6/19+J7/36) = 55 (55 个独立引出 I/O),J5 是复用端口

## 五、配套软件 and 开发环境

开发环境: ISE 13.4 全功能破解版 (仅供学习, 请不要用于商业目的)

## 六、实验例程清单

### 1.单独核心板实验清单

1. 流水灯
2. 分频器
3. 按键控制
4. LCD1602 液晶显示驱动
5. RS232 通信实验 (要配合我们的 NET/USB/LCD7INCH 扩展板)



## 2、EDA6.1 主板配套实验

(这部分代码要配合 EDA6.1 主板进行实验)

(分 VHDL 和 VERILOG 2 部分实验代码，更多实验会陆续增加)

### 1. VHDL 实验代码清单：

2. 3\_8 译码器
3. 分频器
4. 移位运算器
5. 创建一个内部 RAM
6. 流水灯
7. LCD1602 显示驱动
8. LCD12864 显示驱动带中文字库液晶显示中文 (ST7920)
9. 按键控制 LED 实验
10. RS232 串口通讯实验
11. PS2 鼠标控制实验

### 2. verilog 实验代码清单：

1. 分频器
2. BCD 转 7 段码
3. 4 位加法器
4. 多路选择器
5. 16 进制 HEX 转 BCD 码
6. 计数器
7. 比较器
8. 并行转串行
9. 创建一个内部 ROM
10. 数字钟
11. RS232 串口通讯
12. PS2 键盘到 LCD 显示控制
13. PS2 键盘到 RS232 通讯
14. 音乐演奏
15. 红外遥控
16. PWM 控制电机转速
17. LCD12864 显示驱动带中文字库液晶显示中文 (ST7920)
18. 板载图像液晶显示图片

## 七、发货清单 (标准配置)

1. XILINX XC6SLX9 FPGA 核心板 一块
2. LCD1602 液晶屏 一个

3. USB 转 DC 供电线 1 条
4. 配套资料和实验例程开发环境 （通过网盘给用户）

## 八.可选配件

1. 2.8 英寸 TFT 触摸屏 VER1.1
2. 30 万像素摄像数字头
3. USB+网络+TFT7 英寸屏扩展板
4. Xilinx Platform USB Cable 下载线
5. 音频视频采集开发板 VIDEO3.1
6. 音频采集 SD/CF 卡扩展板
7. LCD12864 液晶（带中文字库）
8. EDA4.1 主板
9. EDA5.1 主板
10. EDA6.1 主板
11. EDA7.0 底板
12. USB2.0 VGA/SD 卡 扩展板
13. 高速 AD/DA 模块 VER2.0
14. AD/DA 模块 VER1.0

标准配置发货清单				
				
XC6SLX9 FPGA核 心板一块	LCD1602液晶一个	USB转DC电源线一条	百度网盘 随时随地·轻松分享· 配套资料网盘给用户	

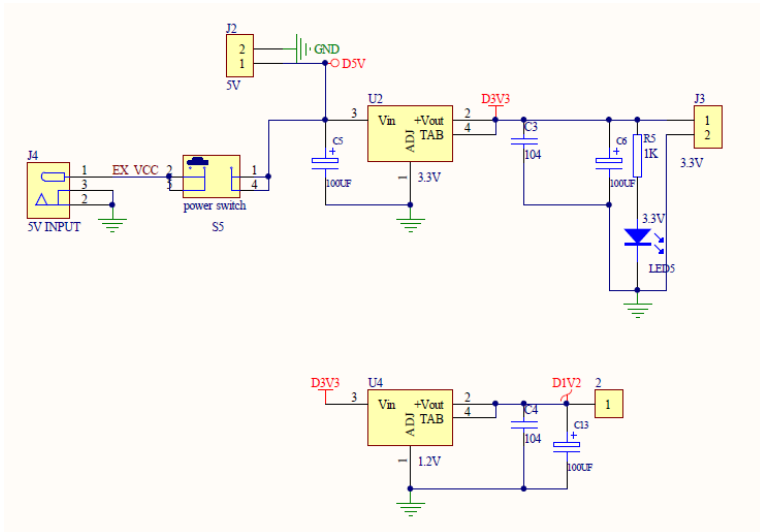
## 九. 电路分析以及说明

### 1.说明

由于电路可能会后续升级，这部分只作为参考，具体版本请看配套光盘的原理图。

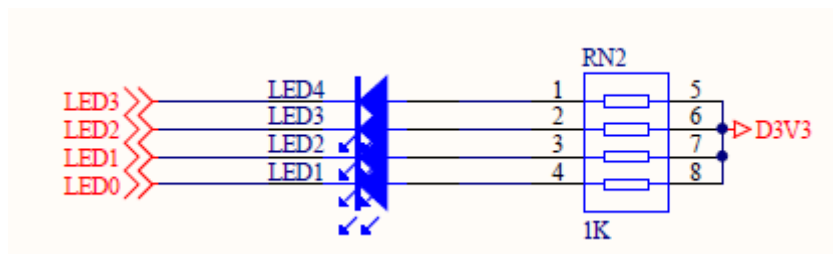
### 1.电源和供电

核心板提供3.3V/800MA，1.2V/800MA,所有电源都进行了引出，方便用户使用。



## 2. 发光二极管 (LED)

电路连接

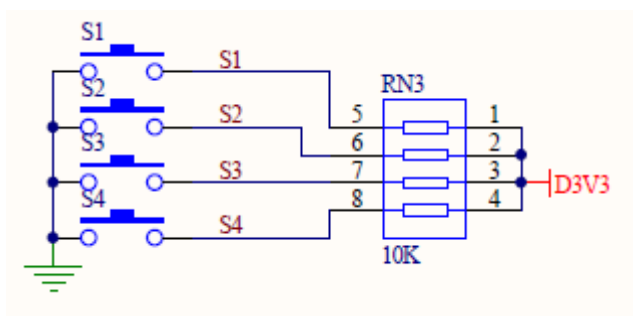


LED 对应管脚

器件名	网络名	FPGA 映射管脚
LED1	LED0	47
LED2	LED1	48
LED3	LED2	50
LED4	LED3	51

## 3. 按键

电路连接



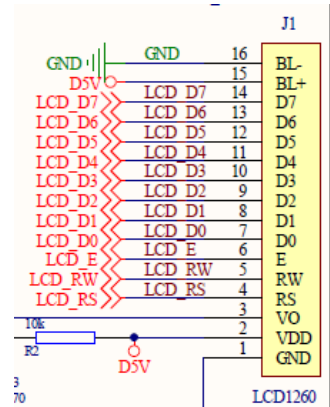
板上按键对应管脚

器件名	网络名	FPGA 映射管脚
S1	S1	6
S2	S2	5

S3	S3	2
S4	S4	1
S6	SYS_RESET	56

## 5. LCD 接口(J4 扩展 I/O)

可以直接插上LCD1602液晶使用，电路连接如下：

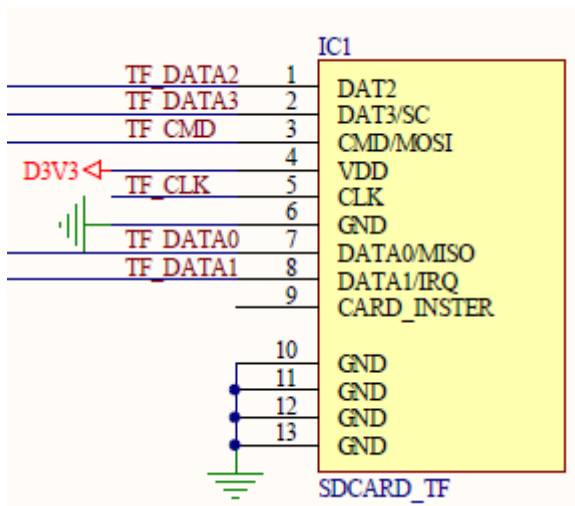


LCD1602 对应管脚(J4)

器件名	网络名	FPGA 映射管脚
D0	LCD_D0	14
D1	LCD_D1	15
D2	LCD_D2	16
D3	LCD_D3	17
D4	LCD_D4	21
D5	LCD_D5	22
D6	LCD_D6	23
D7	LCD_D7	27
E	LCD_E	12
RW	LCD_RW	11
RS	LCD_RS	10

## 4. TF 卡插槽

可以兼容FAT16和FAT32格式的TF卡，电路连接如下：



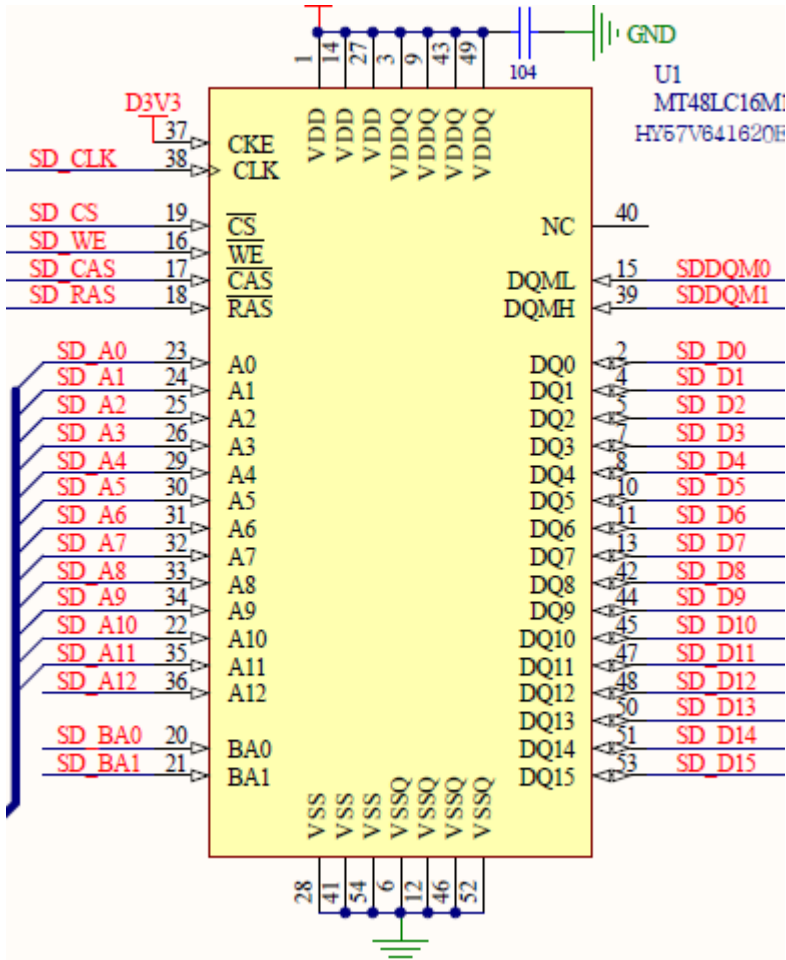
TF 卡座对应管脚

器件名	网络名	FPGA 映射管脚
1	TF_DATA0	82
2	TF_DATA1	81
3	TF_DATA2	87
4	TF_DATA3	85
5	TF_CMD	84
6	TF_CLK	83

## 6. 高速，同步 SDRAM

核心板标配默认采用的采用的是美光的MT48LC8M16是目前应用比较多的8M×16Bit SDRAM。我们预留了大容量（256Mb）SDRAM的IO，所以PCB板上地址线是（A0-A12一共13位地址线），也就是可以直接选配MT48LC16M16芯片。这样方便用户自己将来扩充。





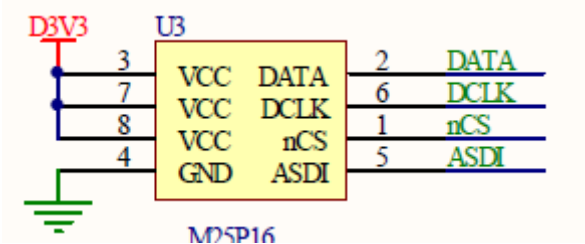
SDRAM 对应管脚 U1

器件名	网络名	FPGA 映射管脚
A0	SD_A0	112
A1	SD_A1	111
A2	SD_A2	105
A3	SD_A3	104
A4	SD_A4	92
A5	SD_A5	93
A6	SD_A6	94
A7	SD_A7	95
A8	SD_A8	97
A9	SD_A9	98
A10	SD_A10	114
A11	SD_A11	99
A12	SD_A12	100
BA0	SD_BA0	116
BA1	SD_BA1	115
DQM0	SDDQM0	121
DQM1	SDDQM1	102
nWE	SD_WE	120
nCAS	SD_CAS	119
nRAS	SD_RAS	118

nCS	SD_CS	117
CLK	SD_CLK	101
DQ0	SD_D0	144
DQ1	SD_D1	143
DQ2	SD_D2	142
DQ3	SD_D3	141
DQ4	SD_D4	140
DQ5	SD_D5	139
DQ6	SD_D6	138
DQ7	SD_D7	137
DQ8	SD_D8	123
DQ9	SD_D9	124
DQ10	SD_D10	126
DQ11	SD_D11	127
DQ12	SD_D12	131
DQ13	SD_D13	132
DQ14	SD_D14	133
DQ15	SD_D15	134

## 8. 串行 Flash（配置芯片）

采用一片常用的 16MB 串行 FLASH 做配置芯片，保存配置程序和相关数据

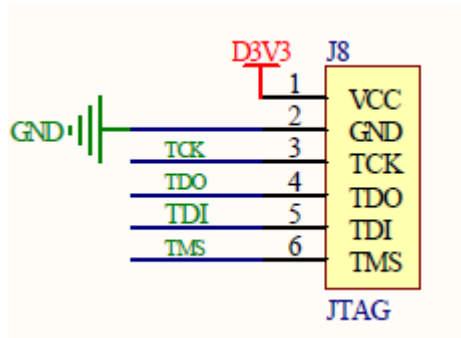


M25P16 (U3) 对应管脚

器件名	网络名	FPGA 映射管脚
DATA0	DATA	65
DCLK	DCLK	70
SCE	NCS	38
SDO	ASDI	64

## 9. JTAG 编程调试接口

电路连接

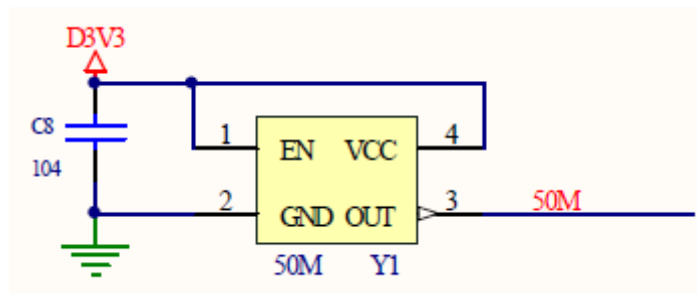


采用XILINX的标准6PIN单排接口,这种接口方便,占用PCB面积比较少,可以用来调试FPGA和对串行FLASH进行编程操作。

## 10.时钟源

采用一个标准的50M有源晶振,方便用户使用。

电路连接

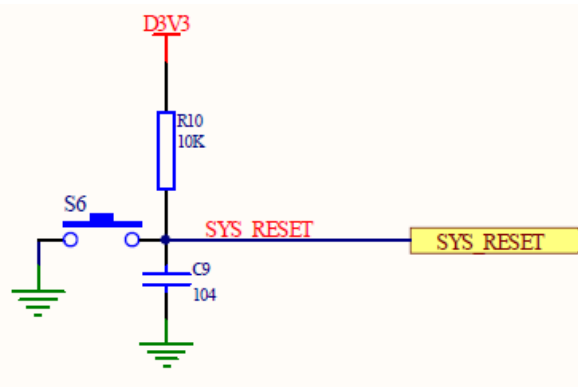


系统时钟对应管脚

器件名	网络名	FPGA 映射管脚
Y1(50M)	50M	55

## 11.复位电路

电路连接



S6	SYS_RESET	56
----	-----------	----

## 十. 原理图:

(见附页)

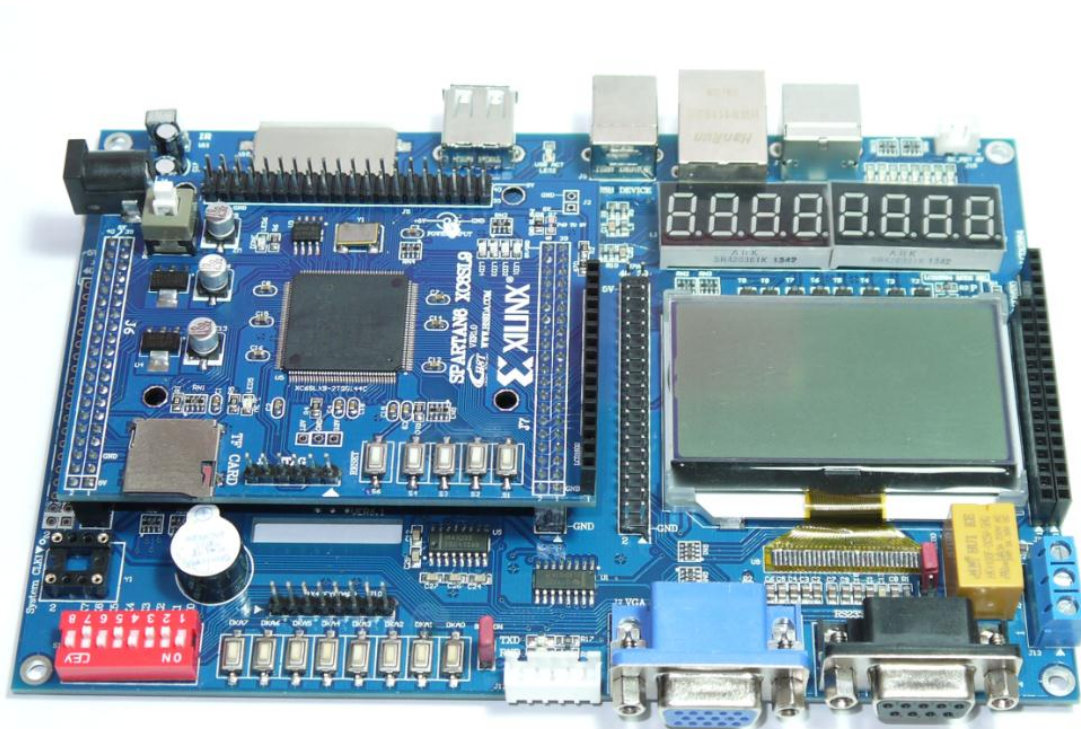
## 十一. 实验指导

### 1.实验前准备:

1.这部分实验要配合我们的 EDA6.1 主板才能进行实验,因为核心板的资源有限,所以要和我们的 EDA6.1 主板进行结合使用。

将核心板插到 EDA6.1 主板上面的 J6, J7 端口。注意核心板的 J6 对应 EDA 主板的 J6 位置, J7 对应主板的 J7 位置,不要插反或插错位置。否则将会烧毁核心板或主板。

这是与 EDA6.1 主板组合结构图



### 2.EDA6.1 主板使用说明

这部分将 EDA6.1 主板使用说明和相关电路分析。

### 3.EDA6.1 主板原理图

见附页 (EDA6.1 主板原理图)

## 创建一个 EDA 实验的详细过程

### 流水灯实验

**实验目的：**使用华升泰克公司 XILINX 开发板 SPARTAN6 XC6SL9 VER1.0 进行流水灯实验

**开发工具：**Xilinx ISE Design Suite 13.4

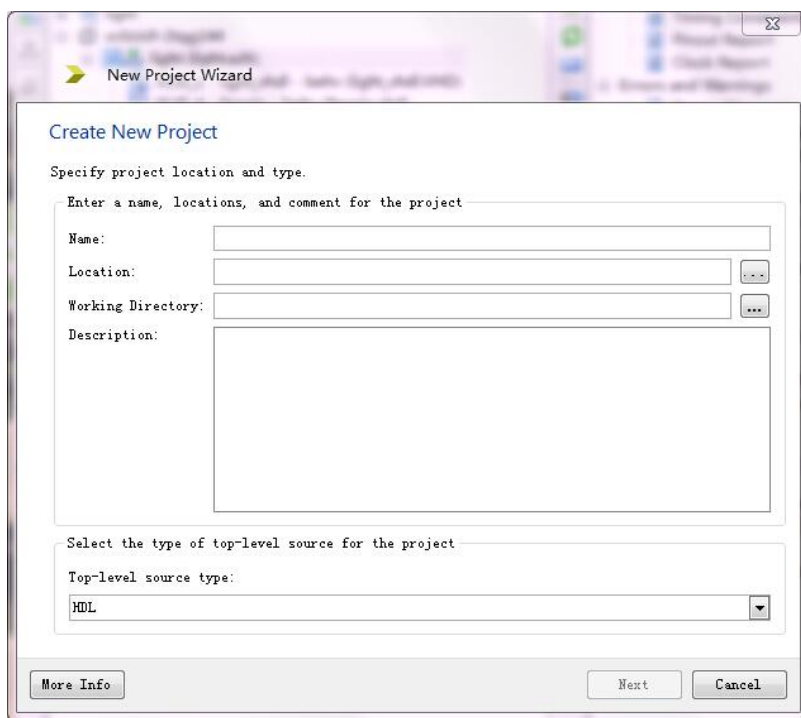
**操作系统：**Windows 7 64Bit

**实验步骤：**



打开软件，点击图标

新建一个工程，点击 File->New Project，进入新建工程向导



我们要做一个流水灯实验，工程名 Name 中输入工程名 Water\_Lights

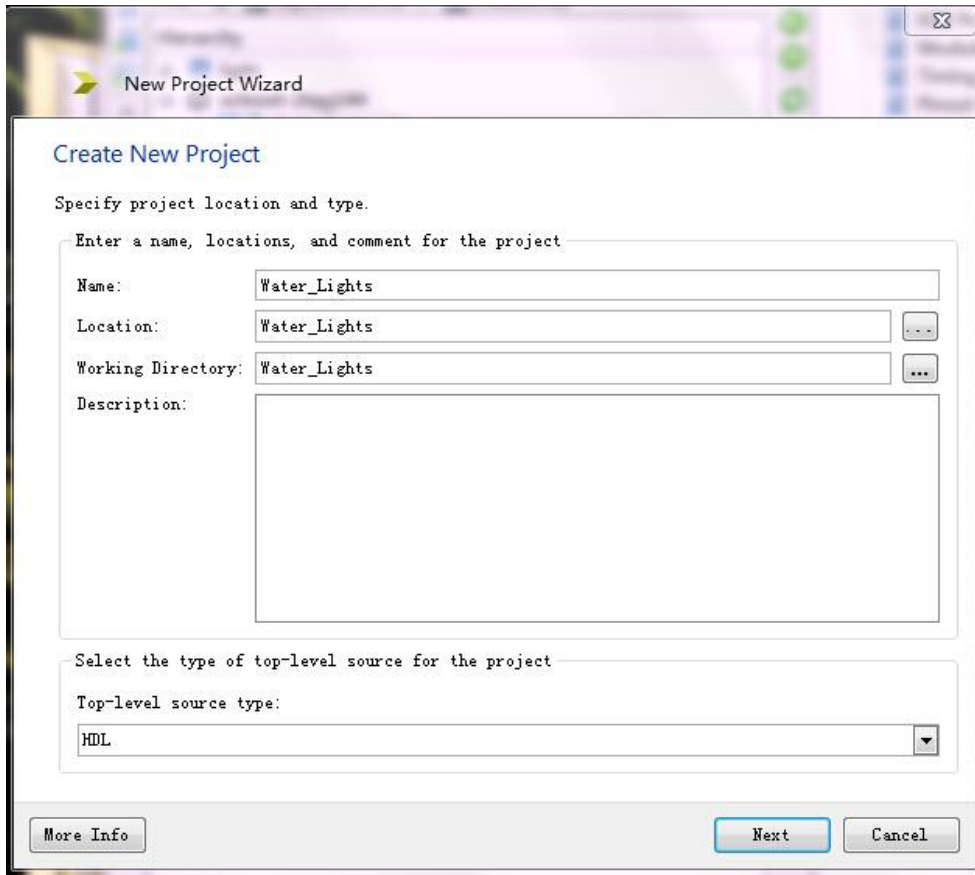
Location 是工程地址，这可以自行输入，但路径名中不能有汉字或空格。

Working Directory 是工作目录，和 Location 一样，由系统自动保持一致，不用管

软件很人性化，当我们输入工程名 Name 时，Location 和 Working Directory 会自己创建相同名字文件夹，我们将文件夹放在 E 盘

输入工程名：

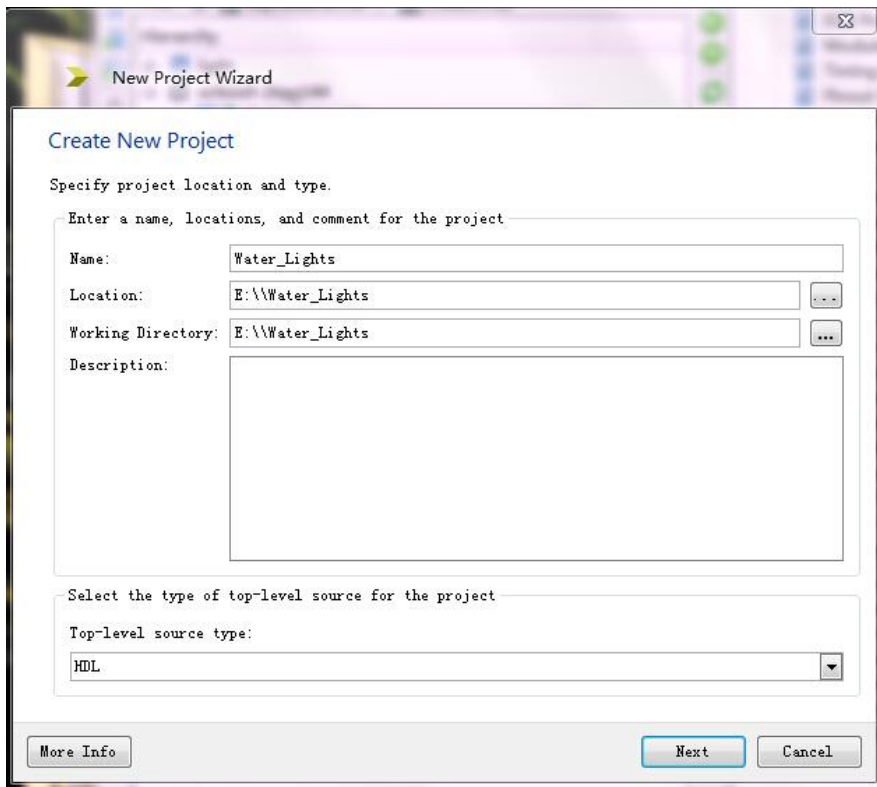




点击本地磁盘(E):



点击确定后:

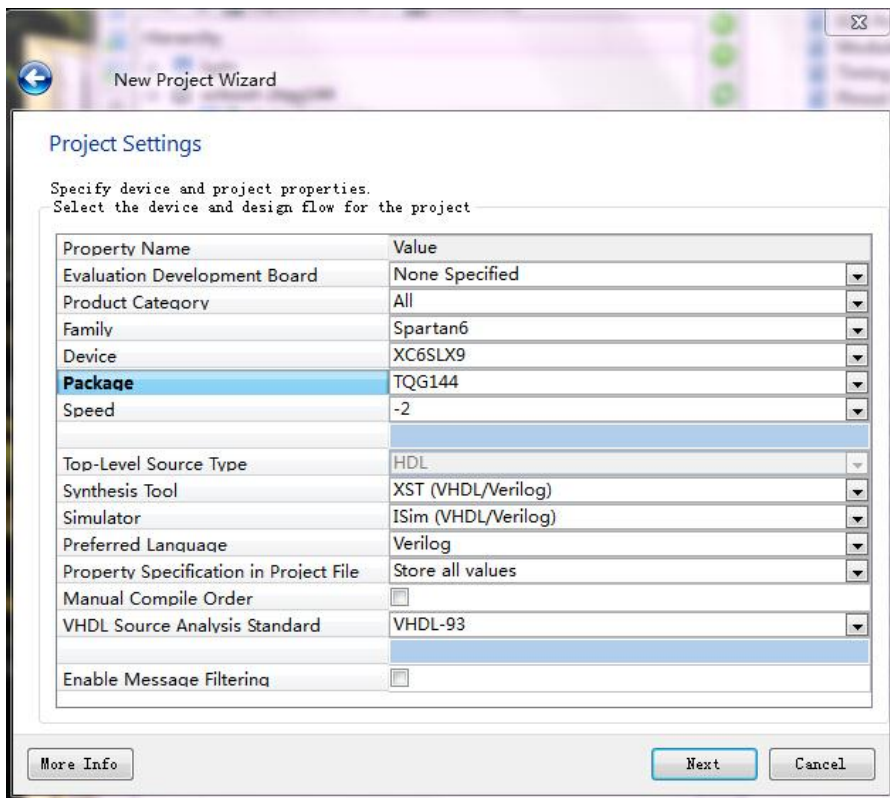


Description 是文件描述

Top-level source type 是顶层文件类型，我们用 Verilog 语言编写，选择缺省值 HDL

以上两项可以不用管

点击 **Next**，继续下一步

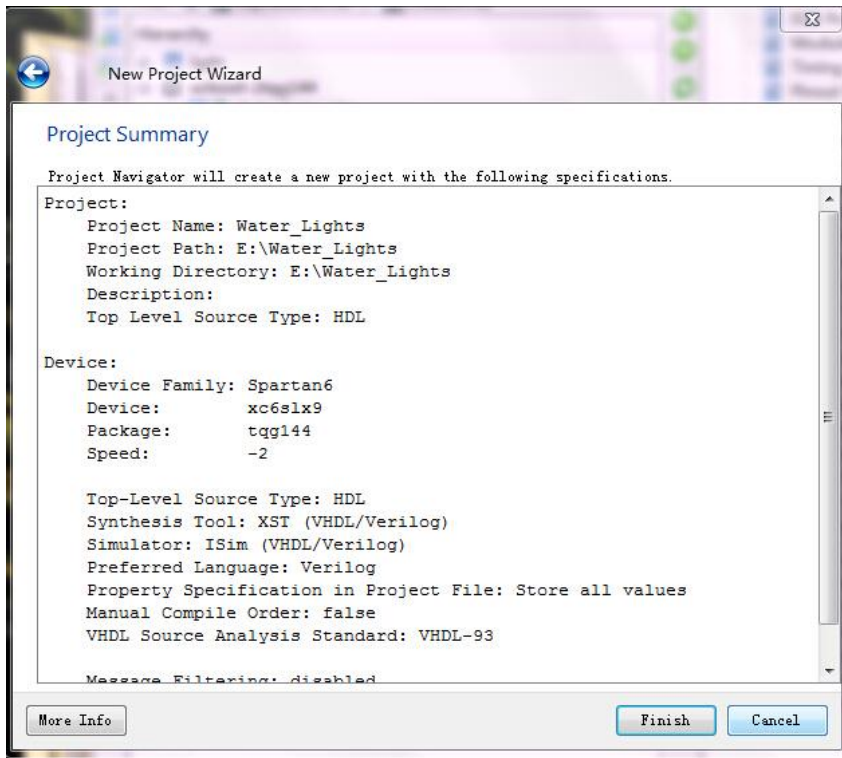


其中有点可以选择注意：

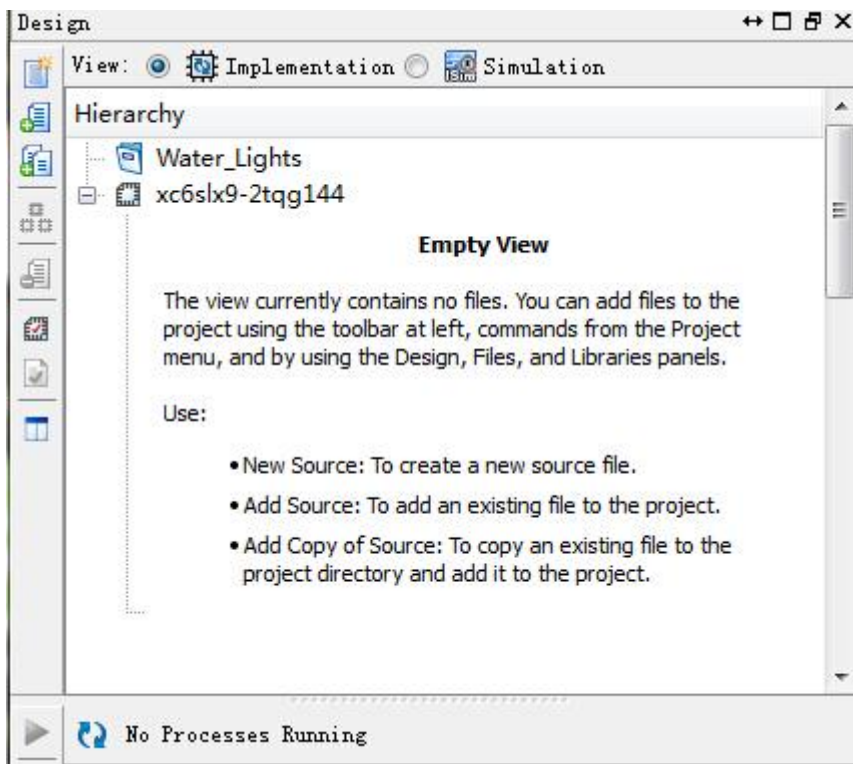
Evaluation Development Board 是指 ISE 推出的特定开发板，我们使用我们公司自己推出的开发板，选择 None Specified


Family 是选择芯片系列，根据开发板上芯片，我们选择 XC6SLX9  
Package 是指芯片引脚个数，根据开发板上芯片，我们选择 TQG144  
Preferred Language 是首选语言，我们选择 Verilog

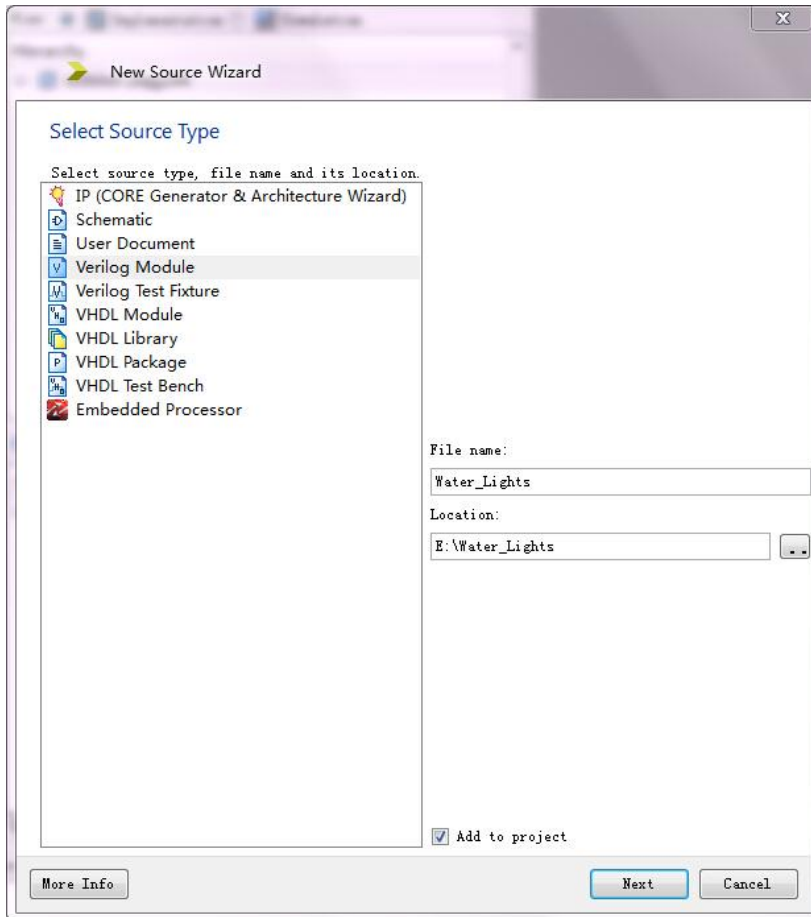
点击 ，继续下一步




这是对工程的一些描述，可以不用管，点击 Finish，完成工程建立。



此时工程是个空的，不包含任何文件，我们要添加一个流水灯的程序，点击 Project->New Source，或右击  选择 New Source，进入新建文件向导。

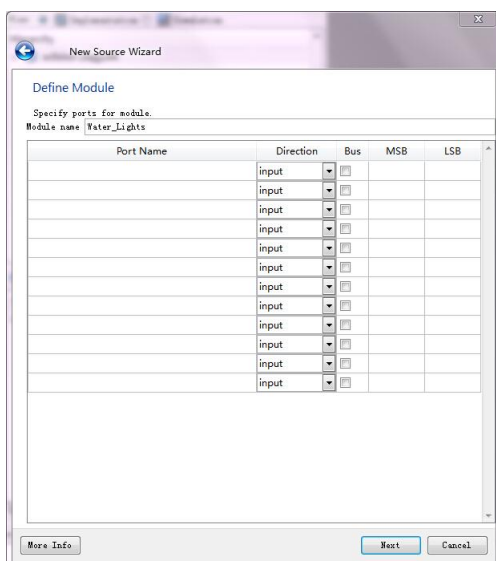


我们程序使用 Verilog 编写，选择  Verilog Module

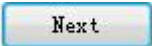
File name 一栏填 Verilog 文件名 Water\_Lights

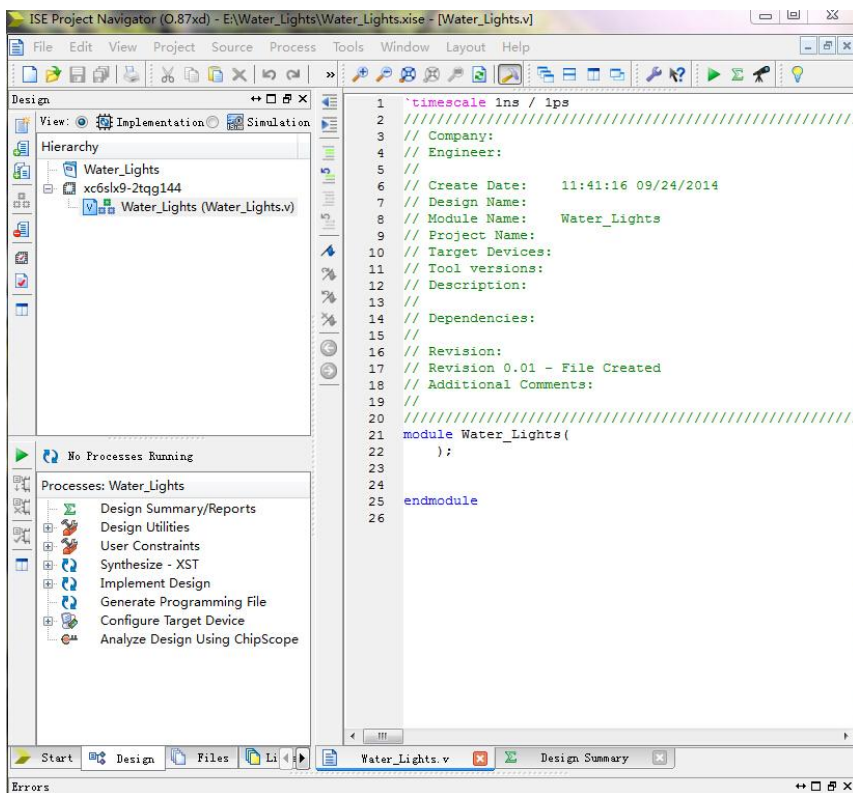
Location 一栏是文件地址，可以不用改

点击 ，继续下一步

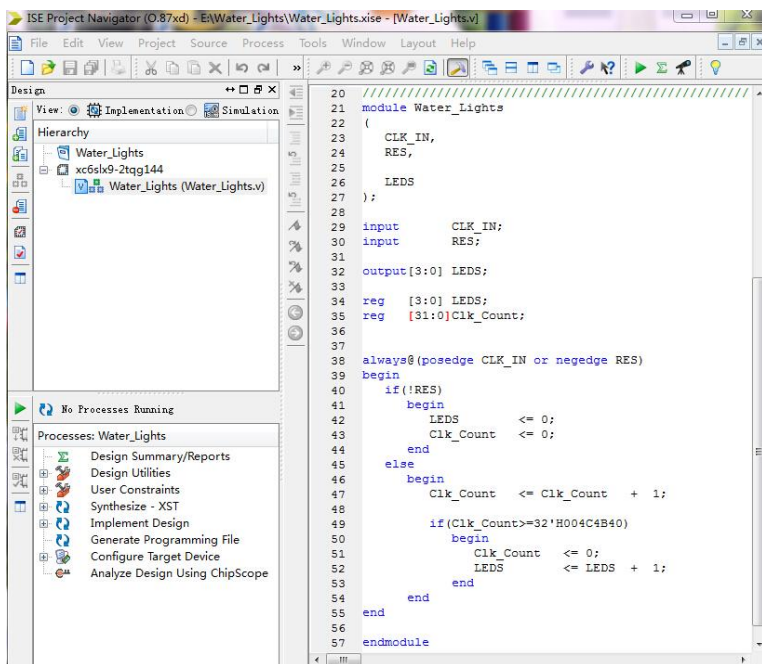


这是对模块名称，输入输入输出的设置，先缺省，在程序中设置

点击 ，继续下一步。出现窗口是对模块的一些描述，点击 Finish，完成模块创立

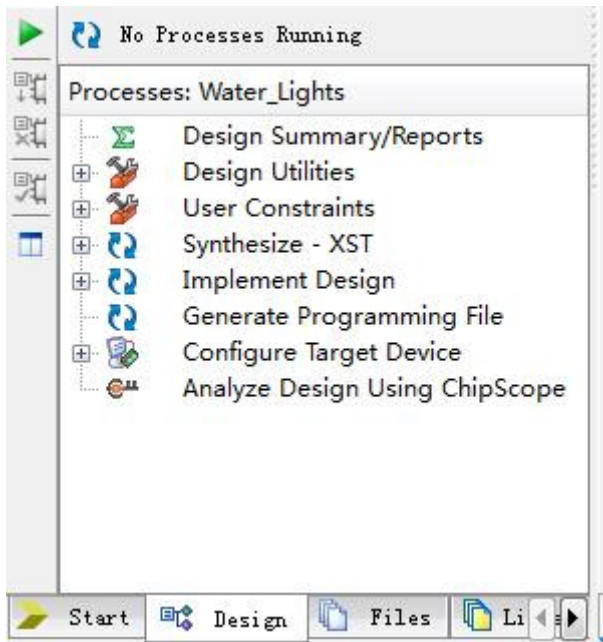




## 编写流水灯程序



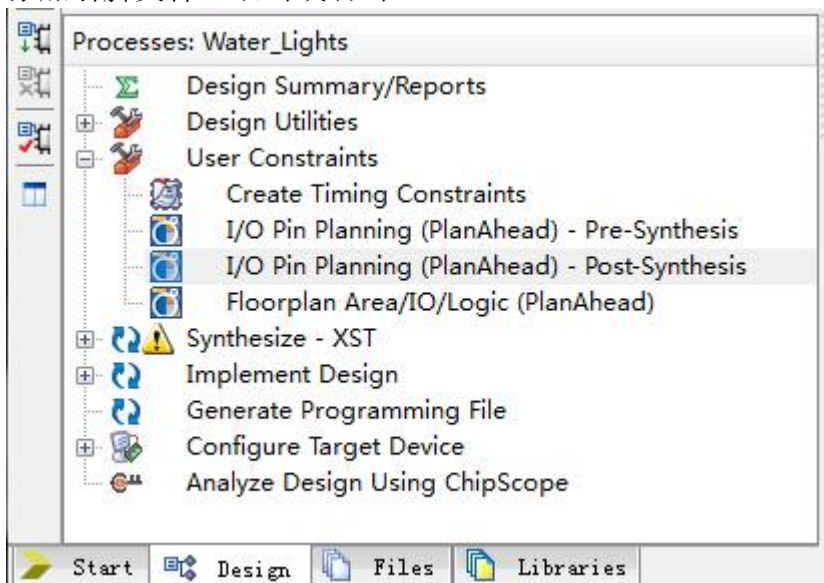
双击或右键 Run




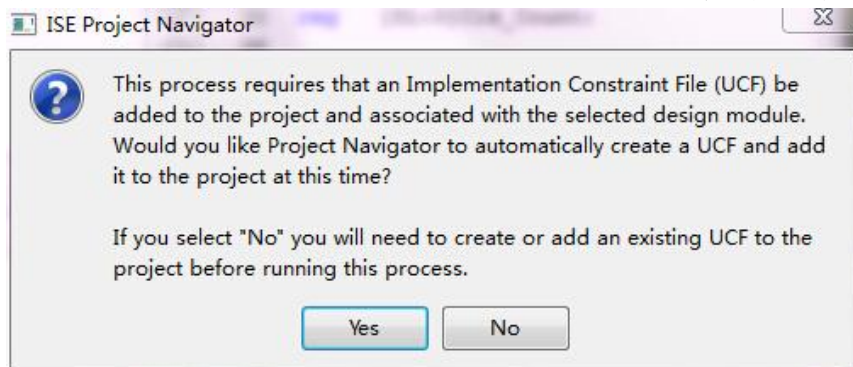


中的  Synthesize - XST 对程序进行综合，综合后选择如果标记  Synthesize - XST，表明程序有问题，需要检查，如果是勾或警告，表明程序没多大问题


添加引脚文件，双击或右击



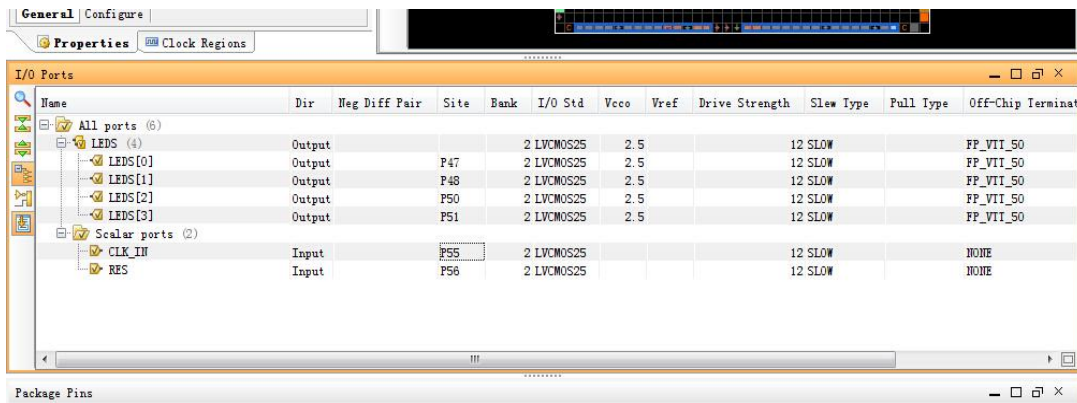
中的  I/O Pin Planning (PlanAhead) - Post-Synthesis 会出现一个界面



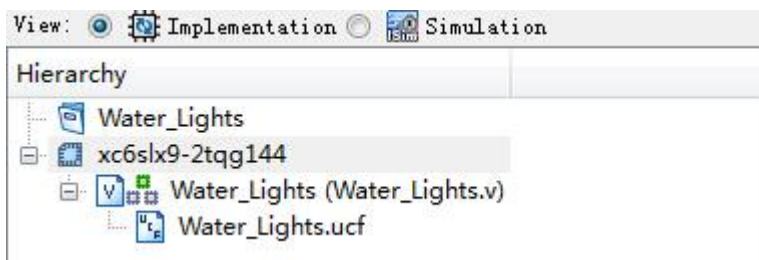
由于引脚文件是. ucf

如果点击 No，需要在 Project->New Source 中创建一个新  Implementation Constraints File，并添加到工程中

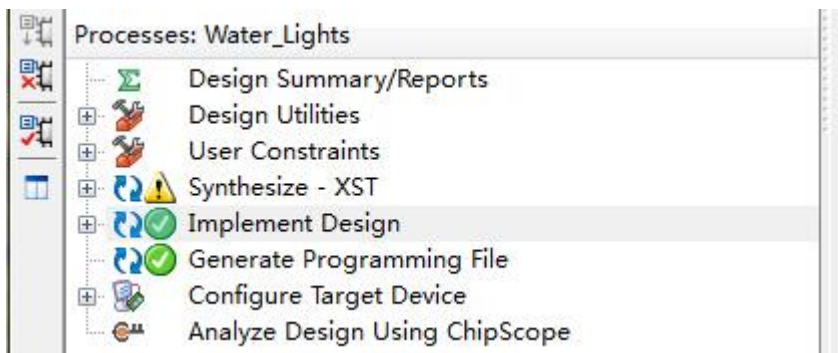
如果点击 Yes，可以由系统在这里生成。点击 YES，等待几秒钟。在出现的窗口中，添加引脚



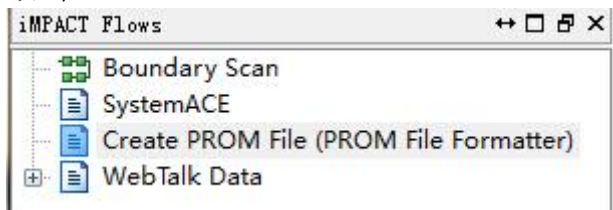
关闭窗口，这是，你会发现，多了一个 Water\_Lights.ucf 文件



点击顶层文件  Water\_Lights (Water\_Lights.v)，在 Processes:Water\_Lights 窗口中，运行 Implement Design、Generate Programming File



下载程序到芯片，将开发板通电，并与下载器连接。右击点 Run 或双击 Configure Target Device，如出现警告，点击 OK。下载文件有两种：一种是烧录到串行 FLASH，重新上电可运行，即.mcs 文件；一种是下载到芯片中，重新上电会擦除，即.bit 文件。Bit 文件由运行 Generate Programming File 时生成。Mcs 文件要手动生成，我们现在先根据.bit 文件，生成 mcs 文件。双击

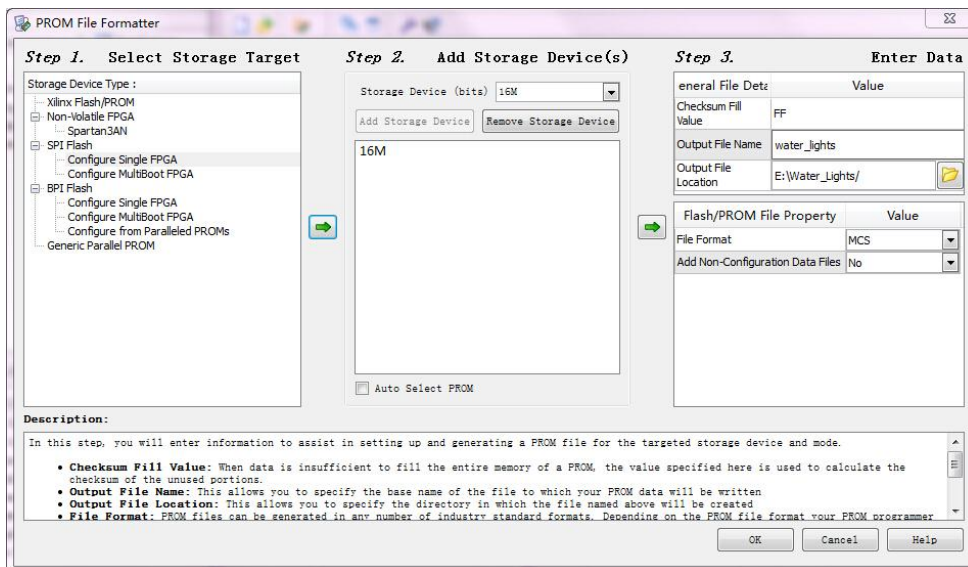


中的 Create PROM File(PROM File Formatter)，出现界面。我们公司的 SPARTAN8 XC6SL9 VER1.0 开发板的配置芯片是 25P16VP，16M SPI FLASH

在 Step 1.Select Storage Target 中双击 SPI FLASH 下的 Configure Single FPGA

在 Step 2.Add Storage Device(s)中 Storage Device (bits)选择 32M，点击下方的 Add Storage Device，配置芯片大小选择好后，点击后面的绿色箭头，进入第三步

在 Step 3. Enter Data 中 Output File Name 栏是生成.mcs 文件名, 为了统一, 我们填写 water\_lights. Output File Location 是输出.mcs 文件地址, 不用改, 其它都为缺省值  
点击 OK

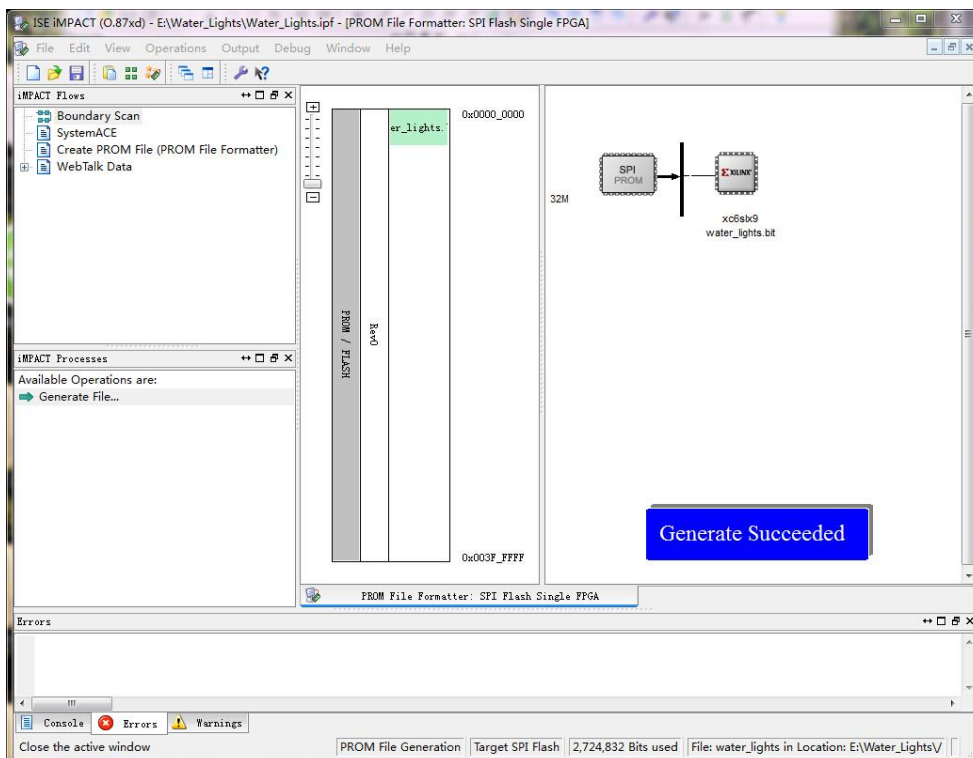



第一个警告 Add Device 是添加.bit 文件, 点击 YES 后, 选择 E:\Water\_Lights 文件下的 water\_lights.bit 文件

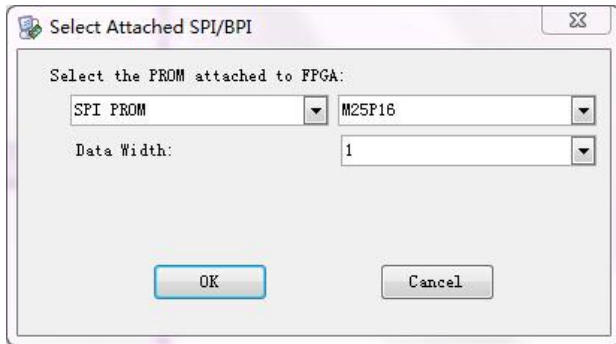
第二个警告不用管, 选择 No

第三个警告选择 OK

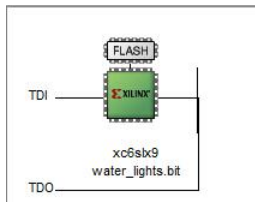
双击 Generate File, 会显示 Generate Succeeded



单击 iMPACT Flows 界面下的 Boundary Scan, 在 Boundary Scan 界面中, 右击选择 initialize Chain 或点击工具栏中 , 扫描通道, 如果扫描失败, 说明开发板或下载器连接有问题。成功后会弹出窗口, 首先选择 water\_lights.bit 文件, 在弹出的 Attach SPI or BPI PROM 窗口是确定添加.mcs 文件的, 我们前面已经生成, 可以添加, 点击 Yes, 否则不添加, 点击 No。在弹出的窗口选择 water\_lights.mcs 后, 弹出选择配置芯片界面, 如下:

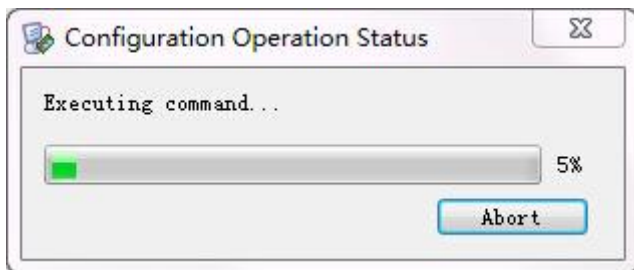


点击 OK，出现的界面也点 OK 即可



当要下载 FLASH 时，右击 FLASH，在出现的下拉界面中选择 Program，等待出现 Program Succeeded，此时 LED 会显示流水灯

当要下载到芯片上时，右击 xc6slx9，在出现的下拉界面中选择 Program，点击 OK，等待出现 Program Succeeded，此时 LED 会显示流水灯



## 第一部分 VHDL 实验

### 实验一、I/O 口控制

程序位置：(XC6SL9\_V6.1\_experiment\VHDL\s1\_KEY2LED)

#### 一、实验目的

熟悉简单的 I/O 控制

#### 二、实验过程

1. 打开 FPGA 电源，打开 s30\_KEY2LED 文件夹下的 KEY2LED.xise，双击 Processes 下 Manage Configuration Project (iMPACT)，进入下载界面；右击 xc6slx9，选择 Program 下载 key2led.bit
2. 具体设计参考代码

#### 三、实验结果

S4-S1 控制核心板上 LED4-LED1

如：按下 S1，LED1 亮




## 实验二、分频器 (VHDL)

程序位置: (XC6SL9\_V6.1\_experiment\VHDL\s2\_fenpin\_VHDL\Proj)

### 一、实验目的

熟悉用 VHDL 语言编写分频器模块

### 二、实验过程

1. 打开 FPGA 电源, 打开 Proj 文件夹下的 fenpin\_VHDL.xise, 双击 Processes 下  Manage Configuration Project (iMPACT), 进入下载界面; 右击 xc6slx9, 选择 Program 下载 fenpin\_vhdl.bit
2. 通过主板 SD2-SD0 观察分频效果
3. 具体设计参考代码

### 三、实验结果

主板 SD2、SD0 同频率闪烁, SD1 闪烁频率低


## 实验三、流水灯

程序位置: (XC6SL9\_V6.1\_experiment\VHDL\s3\_light\Proj)

### 一、实验目的

编写流水灯, 可根据自己喜好做出多种模式

### 二、实验过程

1. 打开 FPGA 电源, 打开 Proj 文件夹下的 light.xise, 双击 Processes 下  Manage Configuration Project (iMPACT), 进入下载界面; 右击 xc6slx9, 选择 Program 下载 light.bit
2. 观赏流水灯
3. 具体设计参考代码

### 三、实验结果

看 SD7-SD0 美妙的变化

注: 主板上 SD7-SD0 高电平点亮


## 实验四、3\_8 译码器

程序位置: (XC6SL9\_V6.1\_experiment\VHDL\s4\_yima\_38\Proj)

### 一、实验目的

用 ISE 软件编写 3\_8 译码器

### 二、实验过程

1. 打开 FPGA 电源, 打开 Proj 文件夹下的 yima\_38.xise, 双击 Processes 下  Manage Configuration Project (iMPACT), 进入下载界面; 右击 xc6slx9, 选择 Program 下载 yima\_38.bit
2. 拨码开关 DK2-DK0 为译码器输入, SD7-SD0 为译码器输出, 改变 DK2-DK0 的值, 观察 LED 变化规律
3. 具体设计参考代码

### 三、实验结果

当 DK2-DK0 的值为“101”时, SD7-SD0 输出“0010-0000”

注: 拨码开关向上为高电平, 向下为低电平

主板 SD7-SD0 为高电平点亮

## 实验五、移位运算器

程序位置: (XC6SL9\_V6.1\_experiment\VHDL\s5\_shifter\Proj)

### 一、实验目的

了解移位运算器功能

### 二、实验过程

1. 打开 FPGA 电源, 打开 Proj 文件夹下的 shifter.xise, 双击 Processes 界面下的



Analyze Design Using Chipscope 后，点击 ，出现界面，右击 DEV:0 MyDevice0(XC6SLX9)，选择 Configure，导入 shifter.bit 到 XC6SLX9 中，双击 Trigger Setup 和 Waveform 出现触发和波形界面，运行 

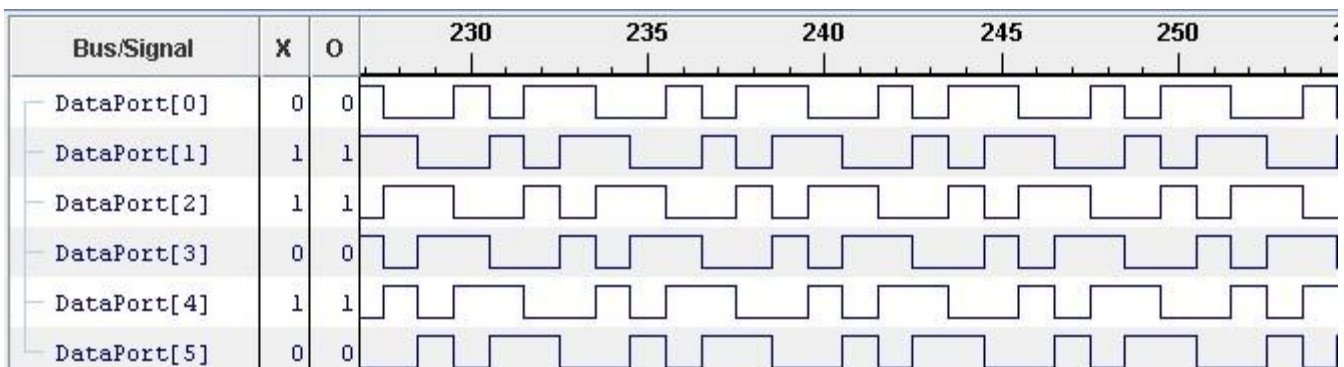
2. 可通过调节按键改变值，DataPort[5:0]是移位输出，可通过输出波形进行分析，DK5-DK0 设置好后，通过 DK7-DK6 启动

3. 具体设计参考代码

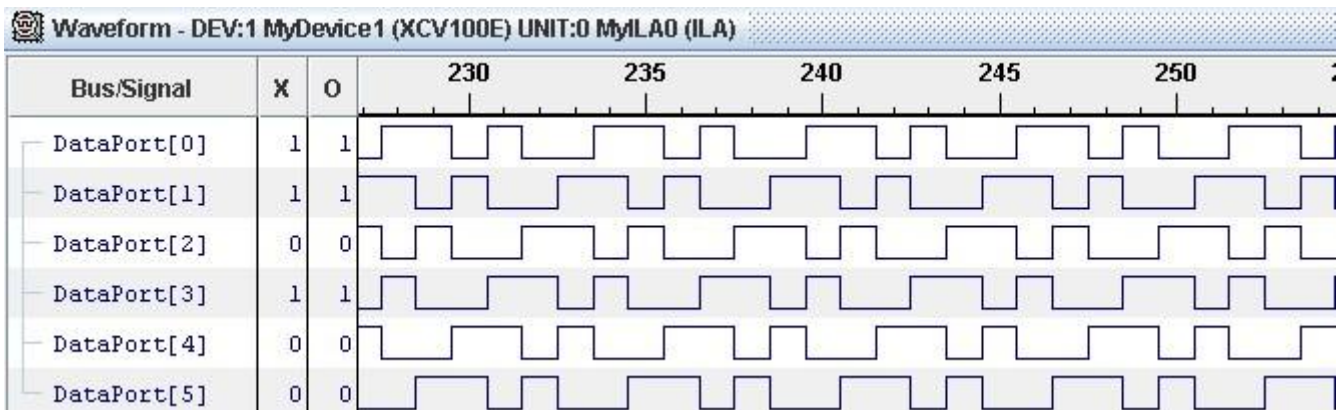
### 三、实验结果

DK5-DK0 数据输入  
DK7-DK6 控制位  
00 左移  
01 右移  
00 =>10 扫描输入数据  
10 =>00 启动  
SD5-0 移位输出

可通过在线逻辑分析看结果，如：DK7-DK6 00；DK5-DK0 010110；结果如图：



如：DK7-DK6 01；DK5-DK0 010110；结果如图：



注：拨码开关向上为高电平，向下为低电平  
主板 SD5-0 高电平点亮


### 实验六、RAM

程序位置：(XC6SL9\_V6.1\_experiment\VHDL\s6\_LPM\_RAM\Proj)

#### 一、实验目的

学会控制 RAM，进行数据存取

#### 二、实验过程

1. 打开 FPGA 电源，打开 Proj 文件夹下的 LPM\_RAM.xise，双击 Processes 下  Manage Configuration Project (iMPACT)，进入下载界面；右击 xc6slx9，选择 Program 下载 lpm\_ram.bit
2. DK3~0 数据，DK7~4 构成 4 位地址。核心板 S1 弹起写，按下读。S1 弹起时高电平。操作比较复杂，要注意

## 3. 具体设计参考代码

## 三、实验结果

写数据时：S1 弹起，DK7~4 输入“0000”，DK0~3 输入“0001”，表明 RAM 地址 0000，存储数据 0001

读数据时：S1 按下，DK7~4 输入“0000”，表明从 RAM 地址 0000，读取数据 0001

存数据时，先按下 S1，保持读的状态。再设置地址，设置数据，弹起 S1，存入数据，最后按下 S1，避免同一地址数据实时更新

## 实验七、UART (VHDL)

程序位置：(XC6SL9\_V6.1\_experiment\VHDL\s7\_uart)

## 一、实验目的

熟悉 VHDL 语言的 UART 驱动

## 二、实验过程

1. 将 RS232 数据线接在主板 RS232 接口上，打开 USRT 数据接收软件

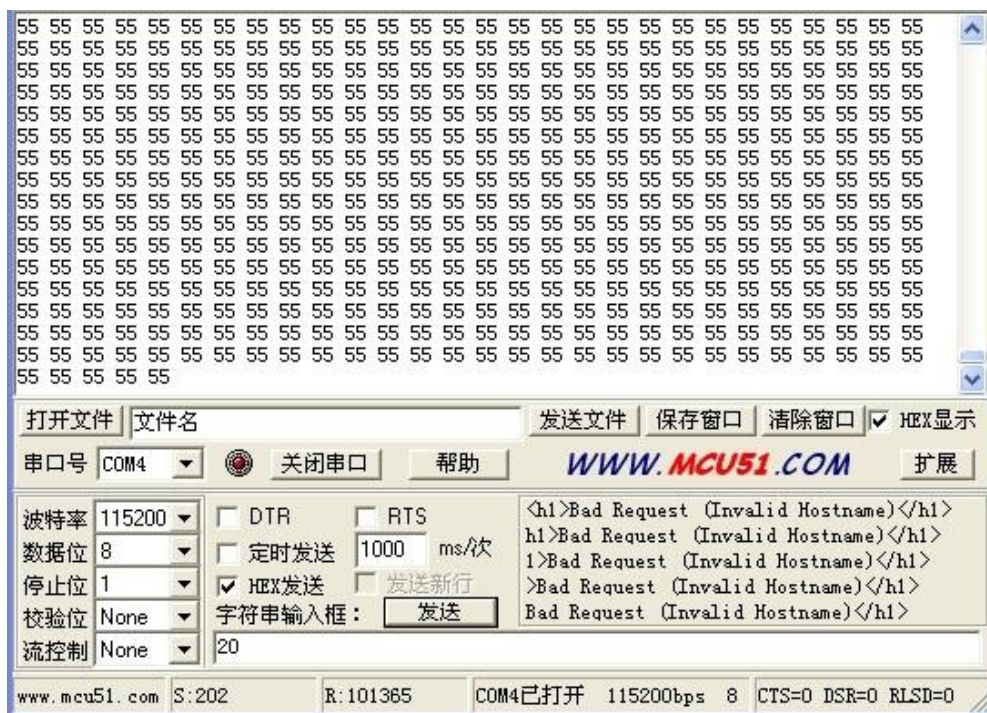
2. 打开 FPGA 电源，打开 s31\_uart 文件夹下的 uart.xise，双击 Processes 下 Manage Configuration Project (iMPACT)，进入下载界面；右击 xc6slx9，Program 下载 uart.bit

## 3. 具体设计参考代码

## 三、实验结果

核心板按键 S1 是使能 xc6slx9 发送，设置为 HEX 接收，电脑接收。按一下 S1，软件显示接收到 55

如图设置，设置为 HEX 发送，电脑发送 20，主板接收，数码管显示 20



## 实验八、PS2 鼠标

程序位置：(XC6SL9\_V6.1\_experiment\VHDL\s8\_ps2mouse)

## 一、实验目的

了解带 PS2 鼠标驱动原理

## 二、实验过程

1. 将带有 PS2 接口的鼠标与开发板主板上的 PS2 接口连接

2. 打开 FPGA 电源，打开 s32\_ps2mouse 文件夹下的 ps2mouse.xise，双击 Processes 下 Manage Configuration Project (iMPACT)，进入下载界面；右击 xc6slx9，选择 Program 下载 ps2mouse.bit

## 3. 具体设计参考代码

### 三、实验结果

通过移动鼠标改变数码管值，左边四个表示 X 方向值，右边四个表示 Y 方向值


### 实验九、LCD1602

程序位置：(XC6SL9\_V6.1\_experiment\VHDL\s9\_LCD1602\Proj)

#### 一、实验目的

了解 LCD1602 驱动原理

#### 二、实验过程

1. 将 LCD1602 插在主板 LCD1602 专用接口上
2. 打开 FPGA 电源，打开 Proj 文件夹下的 LCD1602.xise，双击 Processes 下  Manage Configuration Project (iMPACT)，进入下载界面；右击 xc6slx9，选择 Program 下载 lcd1602.bit
3. 具体设计参考代码

#### 三、实验结果

LCD1602 上显示 Welecom www.HSEDA.com!! 并移屏显示

注：如果 LCD1602 不显示，可以调节 LCD\_CONST，即专用滑动变阻器，调节屏亮度


### 实验十、LCD12864 并行输入 (VHDL)

程序位置：(XC6SL9\_V6.1\_experiment\VHDL\s10\_12864VHDL\Proj)

#### 一、实验目的

熟悉 VHDL 语言的 LCD12864 驱动

#### 二、实验过程

1. 将 LCD12864 插在主板上 LCD12864 专用接口
2. 打开 FPGA 电源，打开 Proj 文件夹下的 LCD12864.xise，双击 Processes 下  Manage Configuration Project (iMPACT)，进入下载界面；右击 xc6slx9，选择 Program 下载 lcd12864vhd1.bit
3. 具体设计参考代码

#### 三、实验结果

LCD12864 显示汉字“班班稗拜败佰摆百”等等

## 第二部分：Verilog 实验


### 实验一、比较器

程序位置：(XC6SL9\_V6.1\_experiment\Verilog\s1\_comparator\Proj)

#### 一、实验目的

了解比较器原理

#### 二、实验过程

1. 打开 FPGA 电源，打开 Proj 文件夹下的 comparator.xise，双击 Processes 下  Manage Configuration Project (iMPACT)，进入下载界面；右击 xc6slx9，选择 Program 下载 comparator.bit
2. DK3-DK0：数据输入 A；DK7-DK4：数据输入 B；主板 SD1-SD0：比较情况
3. 具体设计参考代码

#### 三、实验结果

A = B SD1 亮 SD0 灭

A > B SD1 灭 SD0 亮

A < B SD1 灭 SD0 灭

注：主板 SD1-0 高电平点亮


### 实验二、4bit 加法器

程序位置：(XC6SL9\_V6.1\_experiment\Verilog\s2\_adder\_4bit\Proj)

#### 一、实验目的

编写加法器模块，并用数码管显示结果

#### 二、实验过程

1. 打开 FPGA 电源，打开 Proj 文件夹下的 adder\_4bit.xise，双击 Processes 下  Manage Configuration Project (iMPACT)，进入下载界面；右击 xc6slx9，选择 Program 下载 adder\_4bit.bit
2. DK0-DK3 数据 A 输入，DK4-DK7 数据 B 输入。相加结果十六进制表示。最低位数码管数据输出，小数点显示进位
3. 具体设计参考代码

#### 三、实验结果

DK7-DK4 输入“1011”，DK3-0 输入“0011”，数码管显示“E”

注：拨码开关向上为高电平，向下为低电平

主板数码管共阴


### 实验三、多路选择器

程序位置：(XC6SL9\_V6.1\_experiment\Verilog\s3\_selector\Proj)

#### 一、实验目的

了解多路选择器模块

#### 二、实验过程

1. 打开 FPGA 电源，打开 Proj 文件夹下的 selector.xise，双击 Processes 下  Manage Configuration Project (iMPACT)，进入下载界面；右击 xc6slx9，选择 Program 下载 selector.bit
2. DK0、DK1、DK2、DK3 分别为四个输入信号，DK7-DK6 选择开关，主板 SD0 输出结果
3. 具体设计参考代码

#### 三、实验结果

选择开关“00” SD0 输出对应 DK0 的情况，不管其它输入信号

选择开关“10” SD0 输出对应 DK2 的情况，不管其它输入信号



注：拨码开关向上为高电平，向下为低电平

主板 SD0 高电平点亮




## 实验四、2、4、8 分频器（Verilog）

程序位置：(XC6SL9\_V6.1\_experiment\Verilog\s4\_fenpin\Proj)

### 一、实验目的

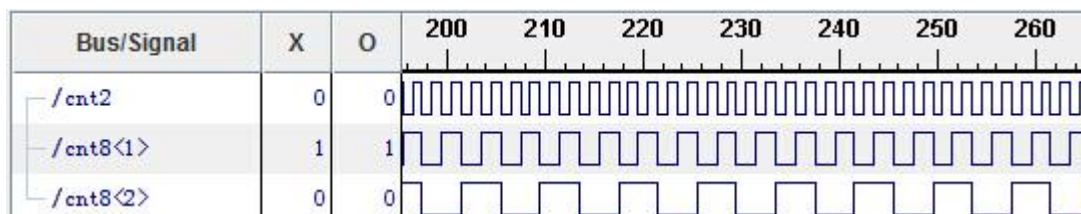
编写 2、4、8 译码器，并进行 ChipScope 在线逻辑分析

### 二、实验过程

1. 打开 FPGA 电源，打开 Proj 文件夹下的 fenpin.xise，双击 Processes 界面下的  Analyze Design Using ChipScope 后，点击 ，出现界面，右击 DEV:0 MyDevice0(XC6SLX9)，选择 Configure，导入 fenpin.bit 到 XC6SLX9 中，双击 Trigger Setup 和 Waveform 出现触发和波形界面，运行 
2. 主板 LED1~LED3 显示三种分频效果，频率高，用肉眼看不出区别，可通过逻辑分析仪或波形仿真查看结果
3. 具体设计参考代码

### 三、实验结果

逻辑分析仪查看结果如图：




## 实验五、数码管

程序位置：(XC6SL9\_V6.1\_experiment\Verilog\s5\_BCD7deg\Proj)

### 一、实验目的

学会控制数码管，显示 0-F

### 二、实验过程

1. 打开 FPGA 电源，打开 Proj 文件夹下的 BCD7deg.xise，双击 Processes 下  Manage Configuration Project (iMPACT)，进入下载界面；右击 xc6slx9，选择 Program 下载 bcd7deg.bit
2. 拨码开关 DK3-DK0 为数据输入，观察数码管随拨码开关的变化规律
3. 具体设计参考代码

### 三、实验结果

拨码开关 DK3-DK0 输入“0000”，数码管显示 0；输入“1111”，显示 F。小数点不显示

注：拨码开关向上为高电平，向下为低电平

主板数码管共阴


## 实验六、HEX 转 BCD 码

程序位置：(XC6SL9\_V6.1\_experiment\Verilog\s6\_HEX\_BCD\Proj)

### 一、实验目的

了解 HEX 转 BCD 码规则，及如何实现

### 二、实验过程

1. 打开 FPGA 电源，打开 Proj 文件夹下的 HEX\_BCD.xise，双击 Processes 下  Manage Configuration Project (iMPACT)，进入下载界面；右击 xc6slx9，选择 Program 下载 hex\_bcd.bit

2. 拨码开关 DK7-DK0 HEX 码输入。核心板 LED4-LED1: BCD 输出高 4 位; 主板 SD7-SD0: BCD 输出低 8 位

3. 具体设计参考代码

### 三、实验结果

DK7-DK0 输入 “0000-1101”

十六进制表示 0x0d

十进制表示 13

BCD 码表示 “0000-0001-0011”

即 0 1 3

注: 拨码开关向上为高电平, 向下为低电平

主板 SD7-0 高电平点亮

核心板 LED4-LED1 低电平点亮


## 实验七、计数器

程序位置: (XC6SL9\_V6.1\_experiment\Verilog\s7\_counter\Proj)

### 一、实验目的

学会调用 IP 核, 如计数器模块, 并对其进行控制, 计数结果用数码管显示

### 二、实验过程

1. 打开 FPGA 电源, 打开 Proj 文件夹下的 counter.xise, 双击 Processes 下  Manage Configuration Project (iMPACT), 进入下载界面; 右击 xc6slx9, 选择 Program 下载 counter.bit

2. 观察数码管

3. 具体设计参考代码

### 三、实验结果

数码管显示计数值 0-16

数据从 0-16 连续计数后会停止计时是由于 TESTCTL 的 TSTEN 的控制, 清零是 TESTCTL 中 CLR\_CNT 端口的控制

可以改变 fp 模块中的 QN 初始值来控制送给计数器的时钟个数进而改变计数范围

注: 主板数码管共阴




## 实验八、并行输入串行输出

程序位置: (XC6SL9\_V6.1\_experiment\Verilog\s8\_parallel\_serial\Proj)

### 一、实验目的

了解如何由并行数据转为串行数据, 并学会操作在线逻辑分析 ChipScope

### 二、实验过程

1. 打开 FPGA 电源, 打开 Proj 文件夹下的 parallel\_serial.xise, 双击 Processes 界面下的  Analyze Design Using Chipscope 后, 出现界面点击 , 右击 DEV:0 MyDevice0(XC6SLX9), 选择 Configure, 导入 parallel\_serial.bit 到 xc6slx9 中, 进入 ChipScope 界面, 双击 Trigger Setup 和 Waveform 出现触发和波形界面, 运行 

2. 改变拨码开关电平, 运行查看波形

3. 具体设计参考代码

### 三、实验结果

拨码开关 DK7-DK0: 输入 8 位信号; 主板 SD0: clk\_out 输出; 主板 SD1: data1 输出

拨码开关输入 0x50, 输出结果如下图:



Bus/Signal	X	O	405	425	445	465	485	505	525	545
/dataS_0_IBUF	0	0								
/dataS_1_IBUF	0	0								
/dataS_2_IBUF	0	0								
/dataS_3_IBUF	0	0								
/dataS_4_IBUF	1	1								
/dataS_5_IBUF	0	0								
/dataS_6_IBUF	1	1								
/dataS_7_IBUF	0	0								
/dataI_OBUF	1	1								
/clk_out_OBUF	0	0								

注：拨码开关向上为高电平，向下为低电平  
主板 SD1-0 高电平点亮

## 实验九、数字时钟

程序位置：(XC6SL9\_V6.1\_experiment\Verilog\s9\_clock\Proj)

### 一、实验目的

制作简易时钟，用数码管显示

### 二、实验过程

1. 打开 FPGA 电源，打开 Proj 文件夹下的 clock.xise，双击 Processes 下 Manage Configuration Project (iMPACT)，进入下载界面；右击 xc6slx9，选择 Program 下载 clock.bit
2. 数码管分别显示时、分、秒。通过对系统时钟分频得出时间，会有误差
3. 具体设计参考代码

### 三、实验结果

低两位数码管跑秒，中间两位数码管跑分，高两位数码管跑时，按照始终规则 60 进 1  
注：主板数码管共阴

## 实验十、IR

程序位置：(XC6SL9\_V6.1\_experiment\Verilog\s10\_IR\Proj)

### 一、实验目的

了解红外传输

### 二、实验过程

1. 打开 FPGA 电源，打开 Proj 文件夹下的 IR.xise，双击 Processes 下 Manage Configuration Project (iMPACT)，进入下载界面；右击 xc6slx9，选择 Program 下载 ir\_demo.bit
2. 对着红外接收器，随便按红外发射器。数码管显示接收到的红外编码，可根据红外发射器按键对应接收到的编码进行反译。
3. 具体设计参考代码

### 三、实验结果

按键 5 数码管显示 0F37 即该红外编码 5 对应 0f37  
注：主板数码管共阴

## 实验十一、ROM

程序位置：(XC6SL9\_V6.1\_experiment\Verilog\s11\_LPM\_ROM\Proj)

### 一、实验目的

学会控制 ROM，进行数据读取

### 二、实验过程

1. 打开 FPGA 电源，打开 Proj 文件夹下的 LPM\_ROM.xise，双击 Processes 下 Manage Configuration Project (iMPACT)，进入下载界面；右击 xc6slx9，选择 Program 下载 lpm\_rom.bit

2. 拨码开关 DK5-DK0 输入 ROM 地址, ROM 输出数据后, 转 BCD, 核心板 LED4~1 显示高 4 位, 主板 SD7~0 显示低八位。ROM 的数据可通过 ROM\_data.coe 查看

3. 具体设计参考代码

### 三、实验结果

拨码开关 DK5-DK0 设置 ROM 地址为“000000”。ROM 地址“0000”存数 0x3f, 转为 BCD 码为 63。查看 LED 情况: 核心板 LED4~1 “0000”, 主板 SD7~0 “0110\_0011”。也可根据 ROM\_data.coe 查看其他数

注: 拨码开关向上为高电平, 向下为低电平

主板 SD7-SD0 高电平点亮

核心板 LED4-LED1 低电平点亮

## 实验十二、PWM


程序位置: (XC6SL9\_V6.1\_experiment\Verilog\s12\_PWM\Proj)

### 一、实验目的

通过驱动直流电机, 熟悉 PWM 控制

### 二、实验过程

1. 将直流电机插在 DC\_MOT 即 J15 端口上

2. 打开 FPGA 电源, 打开 Proj 文件夹下的 PWM.xise, 双击 Processes 下  Manage Configuration Project (iMPACT), 进入下载界面; 右击 xc6slx9, 选择 Program 下载 pwm.bit

3. 具体设计参考代码

### 三、实验结果

DK3-DK0 可控制输出频率, 改变直流电机转速。DK3-DK0 为“1111”时, 转速最快

注: 拨码开关向上为高电平, 向下为低电平

## 实验十三、MUSIC


程序位置: (XC6SL9\_V6.1\_experiment\Verilog\s13\_MUSIC\Proj)

### 一、实验目的

驱动 BUZZER, 播放 MUSIC

### 二、实验过程

1. BUZ\_ON 即 J9 插上插针帽

2. 打开 FPGA 电源, 打开 Proj 文件夹下的 MUSIC.xise, 双击 Processes 下  Manage Configuration Project (iMPACT), 进入下载界面; 右击 xc6slx9, 选择 Program 下载 pwm\_music.bit

3. 具体设计参考代码

### 三、实验结果

BUZZER 播放音乐《两只老虎》

## 实验十四、UART


程序位置: (XC6SL9\_V6.1\_experiment\Verilog\s14\_UART\_in\Proj)

### 一、实验目的

了解 UART 传输原理, 及实际操作

### 二、实验过程

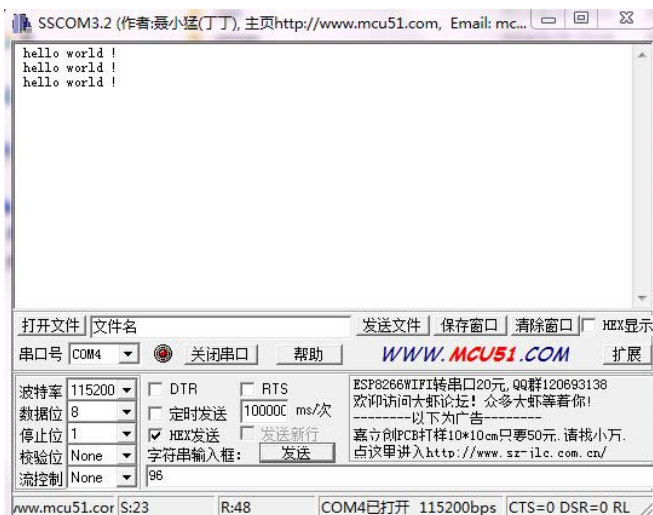
1. 同 RS232 线将电脑 USB 接口与主板上 RS232 接口连接。打开 UART 接收软件, 根据 USB 接口, 选择串口号, 波特率 115200

2. 打开 FPGA 电源, 打开 Proj 文件夹下的 UART\_in.xise, 双击 Processes 下  Manage Configuration Project (iMPACT), 进入下载界面; 右击 xc6slx9, 选择 Program 下载 uart\_ctl.bit

3. 具体设计参考代码

### 三、实验结果

复位时接受数据如下图：



发送数据时设置为 HEX 发送

### 实验十五、PS2\_UART

程序位置：(XC6SL9\_V6.1\_experiment\Verilog\s15\_PS2\_RS232\Proj)

#### 一、实验目的

将 PS2 传输与 UART 传输结合起来，电脑键盘与电脑通过 FPGA 通信

#### 二、实验过程

1. 带 PS2 接口键盘连接主板 PS2 接口，电脑通过 RS232 线与主板 RS232 接口连接，打开 UART 接收软件，根据 USB 接口，选择串口号，波特率 115200
2. 打开 FPGA 电源，打开 Proj 文件夹下的 PS2\_RS232.xise，双击 Processes 下 Manage Configuration Project (iMPACT)，进入下载界面；右击 xc6slx9，选择 Program 下载 ps2\_rs232.bit
3. 具体设计参考代码

#### 三、实验结果

当按下按键后，UART 接收软件上会显示按键值。如下图：




## 实验十六、PS2\_LCD

程序位置: (XC6SL9\_V6.1\_experiment\Verilog\s16\_PS2\_LCD\Proj)

### 一、实验目的

了解 PS2 传输原理

### 二、实验过程

1. 带有 PS2 接口的电脑键盘通过 PS2 线与主板 PS2 接口连接, LCD1602 插在主板 LCD1602 接口上
2. 打开 FPGA 电源, 打开 Proj 文件夹下的 PS2\_LCD.xise, 双击 Processes 下  Manage Configuration Project (iMPACT), 进入下载界面; 右击 xc6slx9, 选择 Program 下载 ps2\_lcd.bit
3. 具体设计参考代码

### 三、实验结果

按 a, LCD1602 显示 a。速度不要太快, 稍等 1s 可以在 lcd 上显示输入的字符。其中键盘上的 backspace 键是用来清屏的。当 lcd 上显示满字符时, 在按下按键自动清屏, 从第一行显示


## 实验十七、LCD12864 并行输入 (Verilog)

程序位置: (XC6SL9\_V6.1\_experiment\Verilog\s17\_12864lcd\Proj)

### 一、实验目的

熟悉 Verilog 语言的 LCD12864 驱动

### 二、实验过程

1. 将 LCD12864 插在主板上 LCD12864 专用接口
2. 打开 FPGA 电源, 打开 Proj 文件夹下的 12864lcd.xise, 双击 Processes 下  Manage Configuration Project (iMPACT), 进入下载界面; 右击 xc6slx9, Program 下载 lcd12864.bit
3. 具体设计参考代码

### 三、实验结果

LCD12864 显示汉字“祆翱熬敖凹盎昂肮”等等


## 实验十八、串行 LCD12864

程序位置: (XC6SL9\_V6.1\_experiment\Verilog\s18\_LCD12864\_PiKaQiu)

### 一、实验目的

了解串行 LCD 驱动原理, 显示皮卡丘图片

### 二、实验过程

1. 打开 FPGA 电源, 打开 s33\_LCD12864\_PiKaQiu 文件夹下的 LCD12864.xise, 双击 Processes 下  Manage Configuration Project (iMPACT), 进入下载界面; 右击 xc6slx9, 选择 Program 下载 LCD12864.bit
2. 具体设计参考代码

### 三、实验结果

串行 LCD 上显示皮卡丘动画