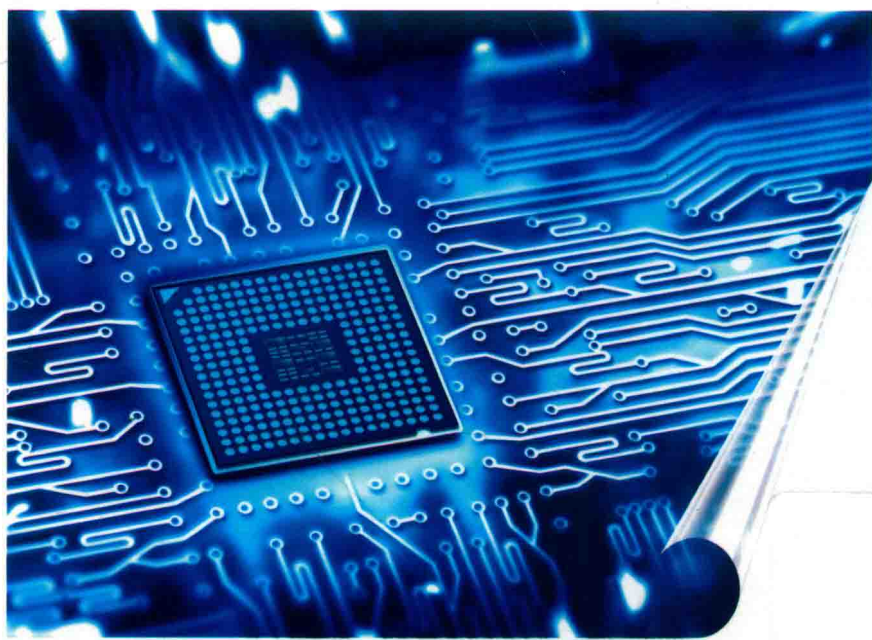


理工科电子信息类 **DIY** 系列丛书

# Verilog HDL

## 实验教程



● 钱敏 曲波 黄旭 编 著  
胡丹峰 黄秋萍

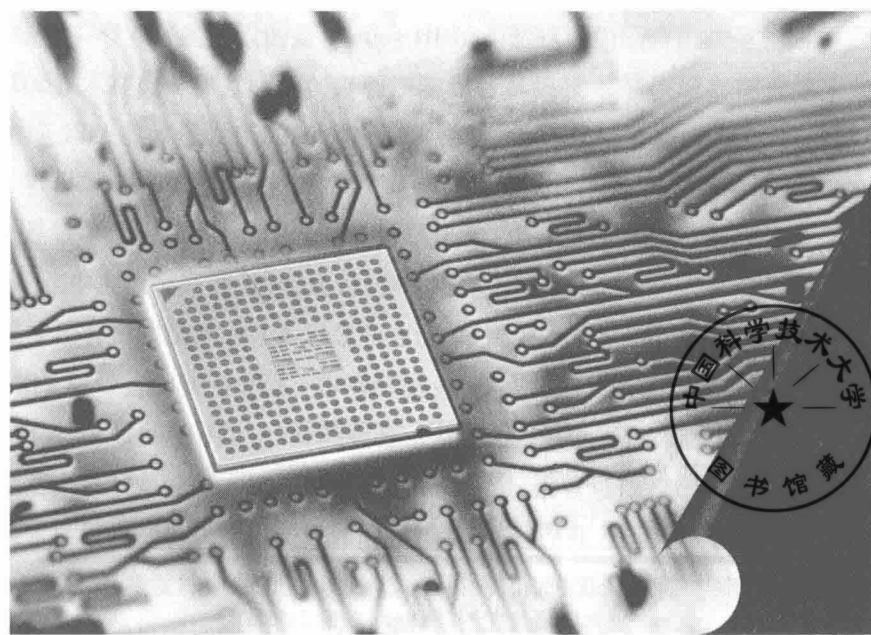


苏州大学出版社

理工科电子信息类DIY系列丛书

# Verilog HDL

## 实验教程



● 钱敏 曲波 黄旭 编 著  
胡丹峰 黄秋萍

## 图书在版编目(CIP)数据

Verilog HDL 实验教程 / 钱敏等编著. —苏州: 苏州大学出版社, 2018. 12  
(理工科电子信息类 DIY 系列丛书)  
ISBN 978-7-5672-2699-9

I. ①V… II. ①钱… III. ①VHDL 语言 - 程序设计 - 实验 - 高等学校 - 教材 IV. ①TP312 - 33

中国版本图书馆 CIP 数据核字(2018)第 282886 号

## Verilog HDL 实验教程

钱敏 曲波 黄旭 胡丹峰 黄秋萍 编著  
责任编辑 征慧

---

苏州大学出版社出版发行

(地址: 苏州市十梓街 1 号 邮编: 215006)

宜兴市盛世文化印刷有限公司印装

(地址: 宜兴市万石镇南漕河滨路 58 号 邮编: 214217)

---

开本 787 mm × 1 092 mm 1/16 印张 12.5 字数 274 千

2018 年 12 月第 1 版 2018 年 12 月第 1 次印刷

ISBN 978-7-5672-2699-9 定价: 38.00 元

---

苏州大学版图书若有印装错误, 本社负责调换

苏州大学出版社营销部 电话: 0512-67481020

苏州大学出版社网址 <http://www.sudapress.com>

苏州大学出版社邮箱 [sdcbs@suda.edu.cn](mailto:sdcbs@suda.edu.cn)

# 前 言

随着现代电子技术的迅速发展,数字系统的硬件设计正朝着速度快、体积小、容量大、重量轻的方向发展。推动该潮流迅猛发展的就是日趋进步和完善的 ASIC 技术。目前,数字系统的设计可以直接面向用户需求,根据系统的行为和功能要求,自上而下地逐层完成相应的描述、综合、优化、仿真与验证,直至生成整个电子系统。其中绝大部分设计过程可以通过计算机自动完成,即电子设计自动化(Electronic Design Automation,EDA)。

实现 EDA 的主要载体就是硬件描述语言,目前主要流行的语言有 VHDL 和 Verilog HDL。尽管国内较早流行的是 VHDL,但在 IC 设计界更为流行的是 Verilog HDL。现在国内高校在微电子相关专业普遍开设的是 Verilog HDL,当然这两门语言内在是相通的。

目前 EDA 技术在电子信息、通信、自动控制和计算机技术等领域发挥着越来越重要的作用,为了适应 EDA 技术的发展和高校的教学要求,我们编写了 Verilog HDL 的实验教程,教程突出了 Verilog HDL 的实用性,以及面向工程实际的特点和学生自主创新能力的培养。Verilog HDL 是数字电路的后续课程,为了更好地和数字电路衔接,我们分两章介绍了组合电路和时序电路中典型电路的设计,通过这些实验,读者能够掌握 Verilog 语言的一般编程方法、硬件描述语言程序设计的基本思想和方法,尽快进入 EDA 的设计实践阶段,熟悉 EDA 开发工具和相关软硬件的使用方法。本书的第 4 章给出了 15 个综合设计型实验,这些实验涉及的技术领域宽,而且具有很好的自主创新的启示性,每个实验都给出了一个设计提示和参考方案,这些方案只是许多方案中的一种,仅供参考,读者可以自己设计其他方案。通过这些实验,读者能够掌握模块化程序设计的思想和方法,提高分析问题和解决问题的能力。

利用硬件描述语言设计电路完成后,必须借助 EDA 的工具软件才能使此设计在 FPGA 上完成硬件实现,并进行硬件验证。为了让读者快速掌握 EDA 工具软件的使用,本书的第 1 章介绍了 Quartus 的使用方法,使用的版本是 Quartus II 9.0。读者只要根据书中的步骤,就能掌握包括设计输入、综合、适配、仿真和编程下载的方法。考虑到有的学校和专业的硬件实现平台还未来得及更新换代,我们在附录 3 中介绍了 MAX + plus II EDA 工具软件的使用。目前在 IC 设计界,编译型仿真软件 ModelSim 使用相当广泛,本书在附录 4 中也进行了简要介绍。Xilinx 器件和设计软件在国内 FPGA 市场也占相当份额,附录 5 中我们对 ISE 集成开发环境及简要使用方法作了一些必要的介绍。

书中的所有实验都通过了 EDA 工具的仿真测试并通过了 FPGA 平台的硬件验证,每个

实验都给出了详细的实验目的、实验原理或设计说明与提示以及实验报告的要求,教师可以根据学时数、教学实验的要求以及不同的学生对象,布置不同任务的实验项目。

本书第1章—第3章、附录3—附录6由钱敏老师编写;第4章中的4.1—4.5由曲波老师编写,4.6—4.10由黄旭老师编写,4.11—4.15由胡丹峰老师编写;附录1—附录2由黄秋萍老师编写。本书在编写过程中引用了诸多学者和专家的著作和研究成果,在这里向他们表示衷心的感谢。

由于作者水平有限且时间仓促,错误和不当之处在所难免,敬请读者不吝赐教。

编 者  
2018年12月

# Contents

## 目录

### 第1章 Quartus II 入门向导

- 1.1 建立工作库文件夹和编辑设计文件 ..... (001)
- 1.2 编译 Verilog HDL 文件 ..... (003)
- 1.3 时序仿真 ..... (006)
- 1.4 创建顶层文件 ..... (010)
- 1.5 引脚设置与硬件验证 ..... (013)

### 第2章 组合电路设计

- 2.1 编码器设计 ..... (016)
- 2.2 译码器设计 ..... (019)
- 2.3 数据选择器设计 ..... (024)
- 2.4 加法器设计 ..... (028)
- 2.5 乘法器设计 ..... (031)
- 2.6 七人表决器设计 ..... (036)

### 第3章 时序电路设计

- 3.1 触发器设计 ..... (039)
- 3.2 寄存器设计 ..... (043)
- 3.3 计数器设计 ..... (047)

- 3.4 模可变 16 位计数器设计 ..... (051)
- 3.5 序列检测器设计 ..... (053)

### 第4章 综合设计型实验

- 4.1 数字秒表设计 ..... (055)
- 4.2 频率计设计 ..... (056)
- 4.3 多功能数字钟设计 ..... (058)
- 4.4 彩灯控制器设计 ..... (060)
- 4.5 交通灯控制器设计 ..... (061)
- 4.6 密码锁设计 ..... (062)
- 4.7 数控脉宽可调信号发生器设计 ..... (064)
- 4.8 出租车计费器设计 ..... (065)
- 4.9 万年历设计 ..... (066)
- 4.10 数字电压表设计 ..... (066)
- 4.11 波形发生器设计 ..... (068)
- 4.12 自动售货机控制电路设计 ..... (069)
- 4.13 电梯控制器电路设计 ..... (071)
- 4.14 自动打铃系统设计 ..... (072)
- 4.15 步进电机细分驱动控制电路设计 ..... (073)

|      |                    |       |
|------|--------------------|-------|
| 附录 1 | GW48 SOC/EDA 系统    |       |
|      | 使用说明 .....         | (075) |
| 附录 2 | NH-TIV 型 EDA 实验开发  |       |
|      | 系统使用说明 .....       | (093) |
| 附录 3 | MAX + plus II 使用指导 |       |
|      | .....              | (106) |

|      |                      |       |
|------|----------------------|-------|
| 附录 4 | ispEXPERT System 3.0 |       |
|      | 使用指导 .....           | (138) |
| 附录 5 | Xilinx ISE9.1i 入门指导  |       |
|      | .....                | (167) |
| 附录 6 | ModelSim 使用入门        |       |
|      | 指导 .....             | (183) |

# 第 1 章 Quartus II 入门向导

Quartus II 软件的操作顺序：

- 编辑 Verilog HDL 程序(使用 Text Editor)；
- 编译 Verilog HDL 程序(使用 Compiler)；
- 仿真验证 Verilog HDL 程序(使用 Waveform Editor、Simulator)；
- 进行芯片的时序分析(使用 Timing Analyzer)；
- 安排芯片脚位(使用 Floorplan Editor)；
- 下载程序至芯片(使用 Programmer)。

下面以 4 位二进制计数器和七段译码为例介绍 Quartus II Verilog HDL 文件的使用方法，使用的版本是 Quartus II 9.0。

## 1.1 建立工作库文件夹和编辑设计文件

### 1. 新建文件夹

可以利用 Windows 资源管理器新建一个文件夹，如“E: \edaexp”，文件夹名不能用中文，不能建在桌面，也不要建在 C 盘。

### 2. 创建工程

执行“File”→“New Project Wizard”命令，如图 1.1 所示，创建工程，工程名可直接用文件的实体名，如图 1.2 中的“top”，然后单击“Finish”按钮。

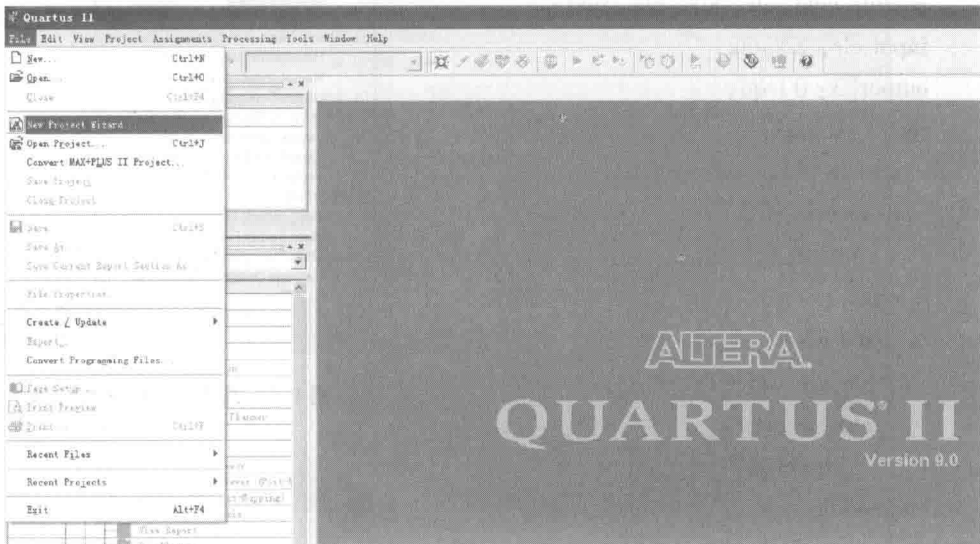


图 1.1 创建工程



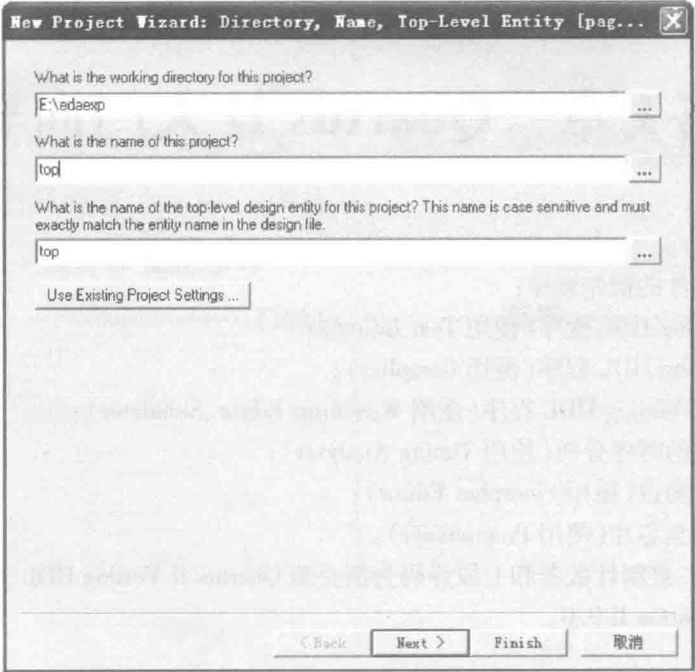


图 1.2 创建工程 top

3. 新建 Verilog HDL 文件

执行“File”→“New”命令,弹出如图 1.3 所示的对话框,选择“Verilog HDL File”。

4. 编辑 Verilog HDL 文件

输入 4 位二进制加法计算器 Verilog HDL 源程序,并另存为实体名 cnt4,如图 1.4 所示。

```
module cnt4( clr, EN, clk, qd );
input clr, EN, clk;
output[3: 0] qd;
reg[3: 0] qd;
always @ (posedge clk)
if( clr)
    qd = 0;
else
    if ( EN)
        qd = qd + 1;
    else
        qd = qd;
endmodule
```

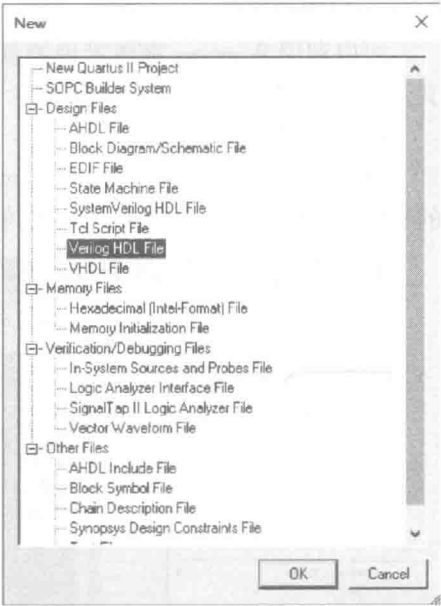


图 1.3 选择 Verilog HDL 文件



图 1.4 编辑 Verilog HDL 源程序

## 1.2 编译 Verilog HDL 文件

在对工程进行编译处理前,要进行一些相应的设置。

### 1. 选择 FPGA 目标芯片

选择“Assignments”→“Settings”命令,在打开的“Settings-cnt 10”对话框的“Category”栏中选择“Device”,选择 ACEX1K 系列中的 EP1K30TC144-3 为目标芯片,如图 1.5 所示。

目标芯片也可在创建工程的时候选择确定。

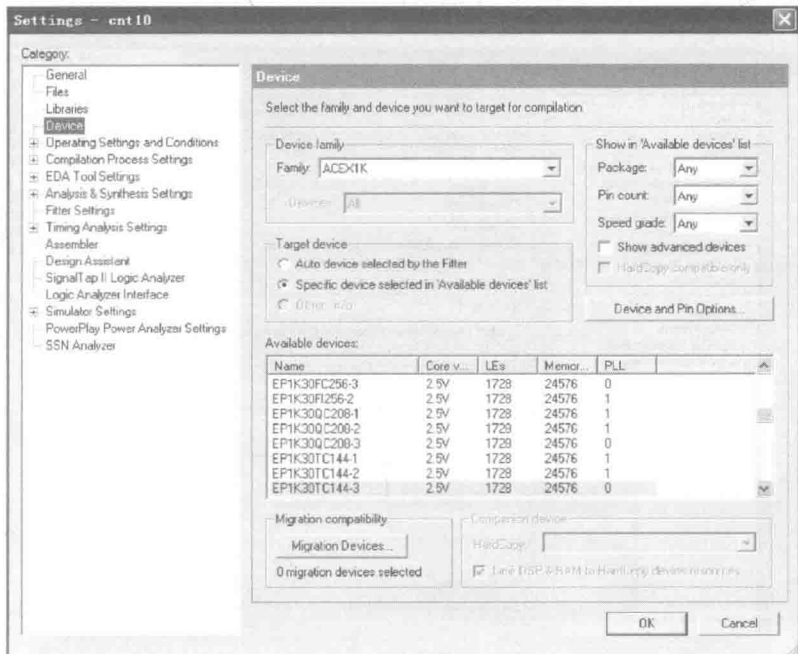


图 1.5 芯片选择

## 2. 器件的其他设置

在图 1.5 中,单击“Device and Pin Options”按钮,打开如图 1.6 所示的对话框。

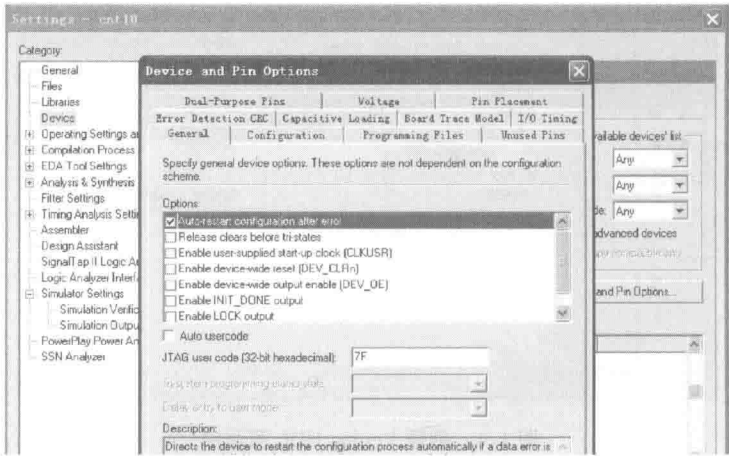


图 1.6 器件的设置

- (1) 在“Options”项中选择“Auto-restart configuration after error”复选框。
  - (2) 在“Configuration”项中选择“Passive Parallel synchronous”复选框。
  - (3) 在“Unused Pins”项中选择“As Output Driving Ground”复选框。
- 其他可不选。

## 3. 选择确认 VerilogHDL 语言版本

在“Category”项中选择“Analysis & Synthesis Settings”→“Verilog HDL Input”命令,在“Verilog version”下选中“Verilog-2001”单选按钮,如图 1.7 所示。

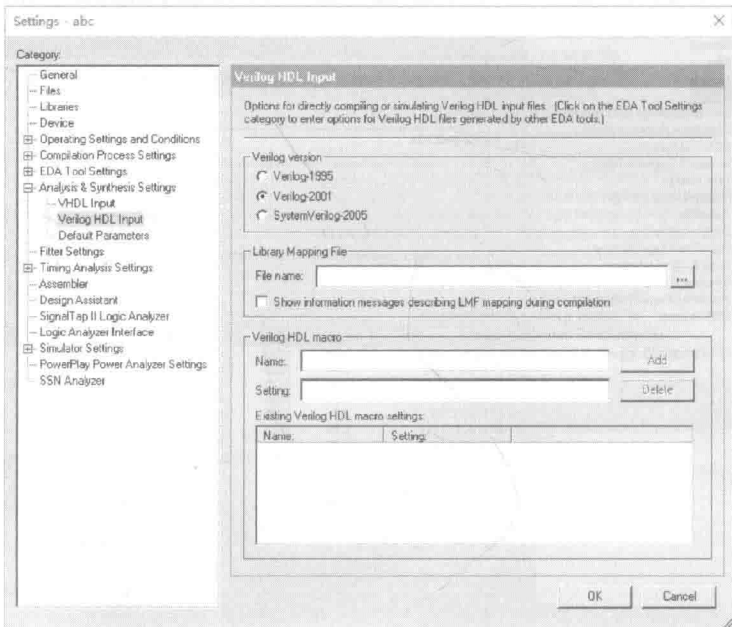


图 1.7 选择 Verilog HDL 版本

#### 4. 全程编译

(1) 在全程编译前,选择“Project”→“Set as Top-Level Entity”命令,使当前的 cnt4 成为顶层文件,如图 1.8 所示。

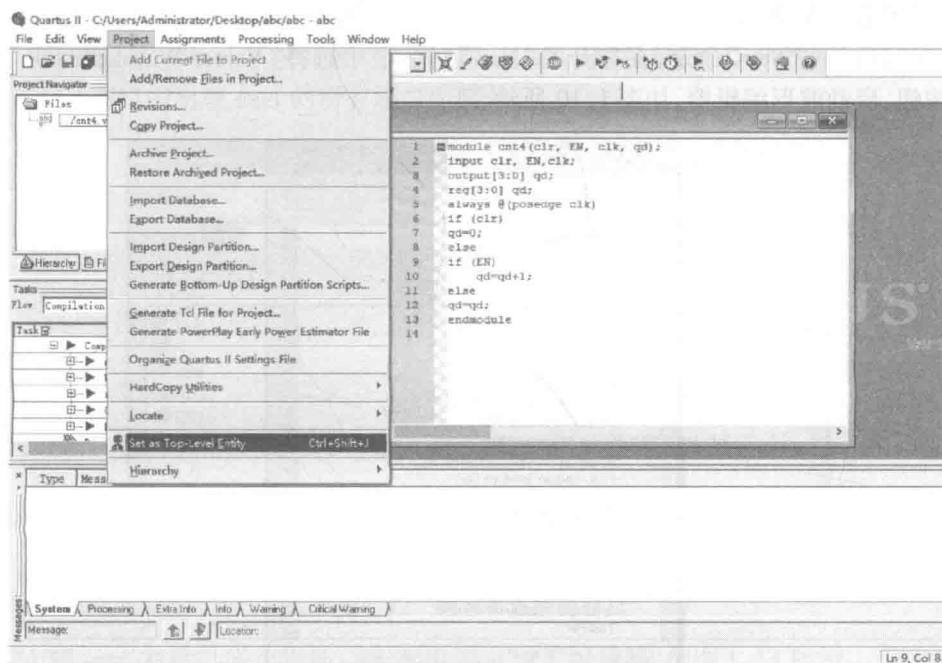


图 1.8 设置为顶层文件

(2) 选择“Processing”→“Start Compilation”命令,进行全程编译,完成后弹出如图 1.9 所示的“Quartus II”对话框。

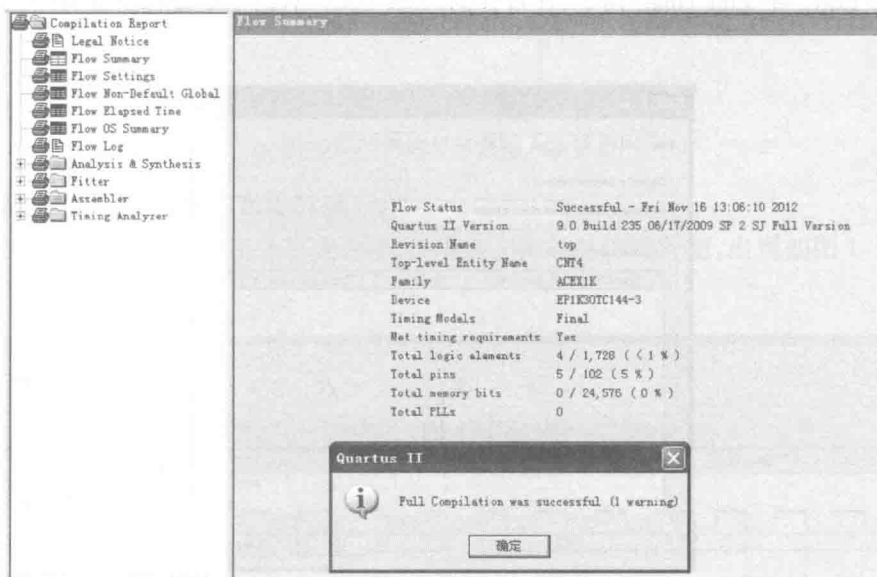


图 1.9 编译界面

## 1.3 时序仿真

### 1. 打开波形编辑器

选择“File”→“New”命令,在打开的“New”对话框中选择“Vector Waveform File”,单击“OK”按钮,启动波形编辑器,如图 1.10 所示。

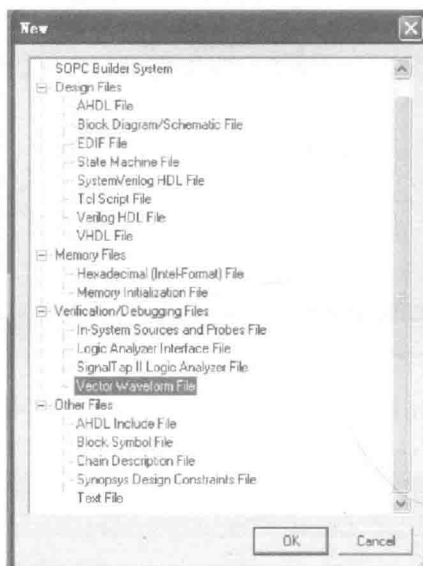


图 1.10 打开波形编辑器

### 2. 设置仿真时间区域

选择“Edit”→“End Time”命令,在打开的“End Time”对话框中设置仿真时间,如图 1.11 所示。

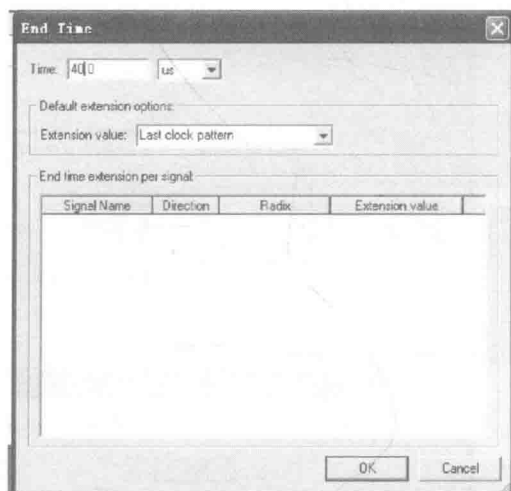


图 1.11 设置仿真时间

### 3. 波形文件存盘

选择“File”→“Save As”命令,以默认名 cnt4. vwf 存盘。

### 4. 将工程 cnt4 的端口信号节点选入波形编辑器

(1) 选择“View”→“Utility Windows”→“Node Finder”命令,弹出如图 1.12 所示的“Node Finder”对话框,在“Filter”下拉列表中选择“Pins: all”,然后单击“List”按钮,于是在下方的 Nodes Found 窗口中出现 cnt4 的所有端口引脚名。

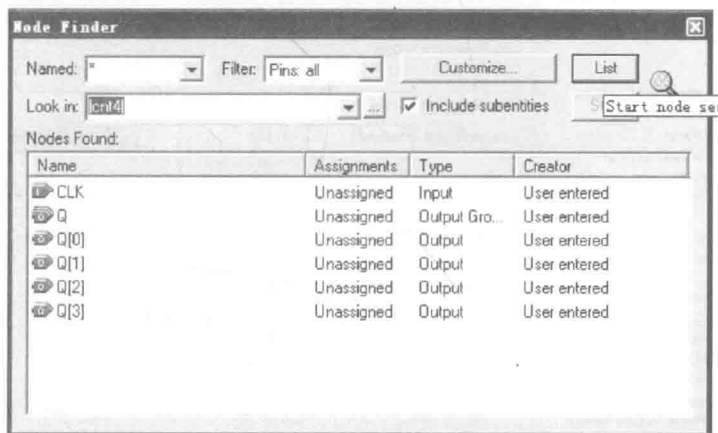


图 1.12 信号节点查询窗口

(2) 将 cnt4 的端口信号节点 CLK 和 Q 拖入波形编辑器,如图 1.13 所示。

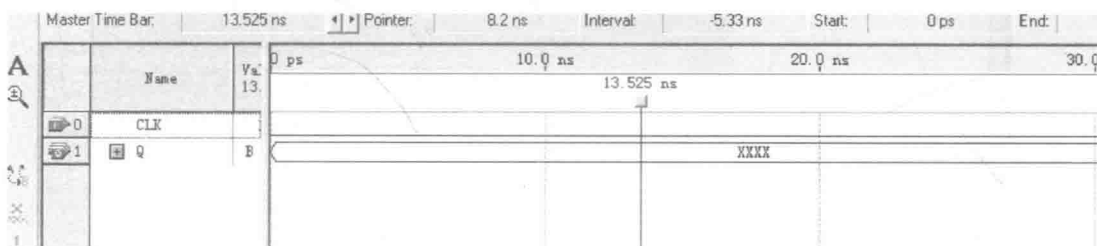


图 1.13 将信号节点拖入编辑器

### 5. 编辑输入波形(输入激励信号)

单击时钟信号 CLK,使之变成蓝色条,再单击左列的时钟设置键,出现如图 1.14 所示的窗口,然后存盘。

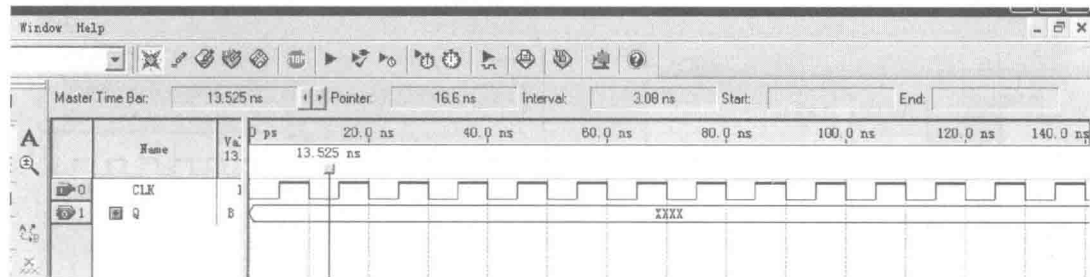


图 1.14 编辑输入信号

## 6. 仿真器参数设置

选择“Assignments”→“Settings”命令,在打开的“Settings-top”对话框的“Category”栏中选择“Simulator Settings”,在“Select simulation options”下的“Simulation mode”下拉菜单中选择“Timing”,如图 1.15 所示。

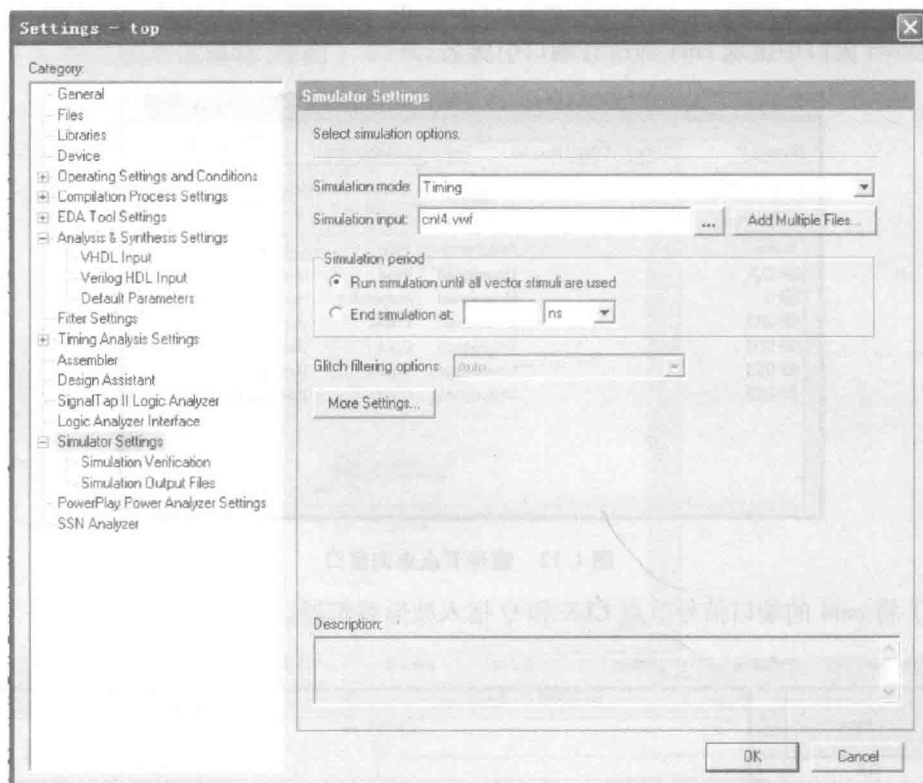


图 1.15 仿真器参数设置

## 7. 启动仿真器

选择“Processing”→“Start Simulation”命令,直到出现“Simulation was successful”,仿真结束。

## 8. 观察仿真结果

仿真结果如图 1.16 所示。

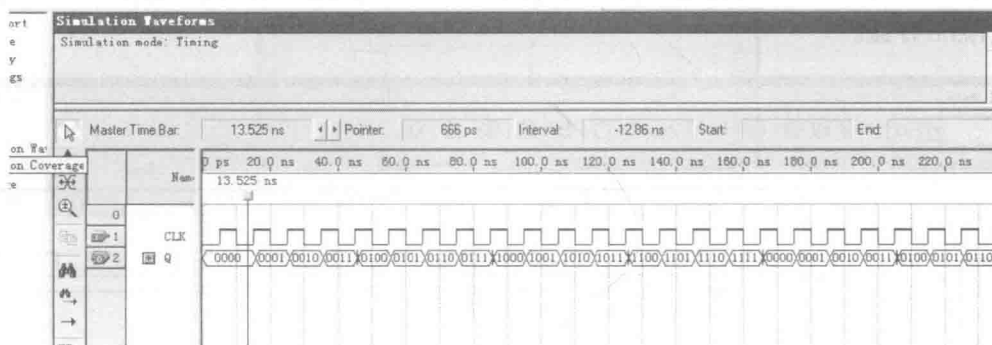


图 1.16 仿真结果

## 9. 应用 RTL 电路图观察器

(1) 选择“Tool”→“Netlist Viewers”命令,选择“RTL Viewer”,可看到生成的 RTL 级电路图形,如图 1.17 所示。

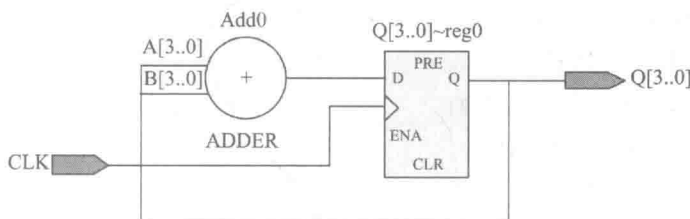


图 1.17 RTL 级电路

(2) 选择“Tool”→“Netlist Viewers”命令,选择“Technology Map Viewer”,可看到 FPGA 底层的门级电路,如图 1.18 所示。

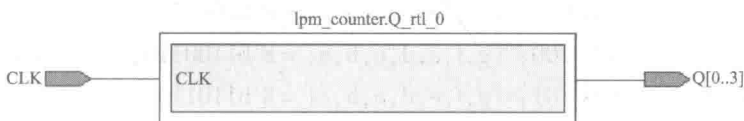


图 1.18 FPGA 门级电路

## 10. 创建元件

选择“File”→“Create/Update”→“Create Symbol Files for Current File”命令,把当前的 cnt4 创建为一个符号元件,如图 1.19 所示。

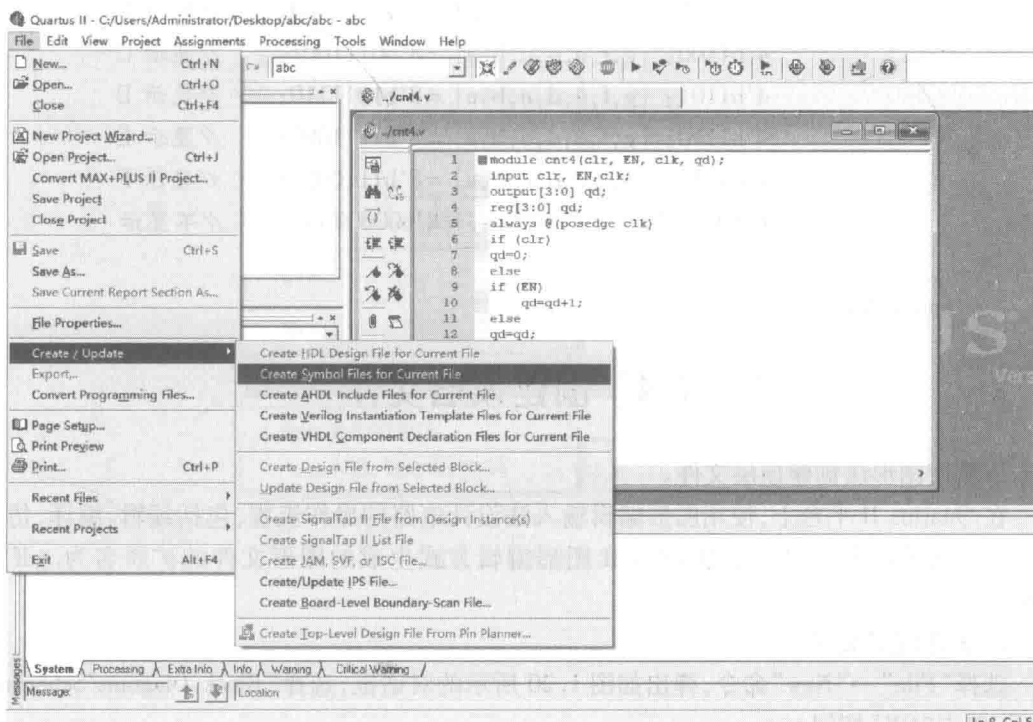


图 1.19 创建元件





## 11. 编辑 DECL7S 的源程序并编译和仿真

重复 1.1 的 3 到 1.3 的 11 的过程,编辑七段译码的源程序并编译和仿真。注意把源程序 DECL7S 放入 cnt4 同一个目录中。程序如下:

```
module DECL7S(AIN,a,b,c,d,e,f,g);
    input[4:1]AIN;
    output a,b,c,d,e,f,g;
    reg a,b,c,d,e,f,g;
    always @(AIN)
        case(AIN)
            4'b0000: {g,f,e,d,c,b,a} = 8'b0111111;    //显示 0
            4'b0001: {g,f,e,d,c,b,a} = 8'b0000110;    //显示 1
            4'b0010: {g,f,e,d,c,b,a} = 8'b1011011;    //显示 2
            4'b0011: {g,f,e,d,c,b,a} = 8'b1001111;    //显示 3
            4'b0100: {g,f,e,d,c,b,a} = 8'b1100110;    //显示 4
            4'b0101: {g,f,e,d,c,b,a} = 8'b1101101;    //显示 5
            4'b0110: {g,f,e,d,c,b,a} = 8'b1111101;    //显示 6
            4'b0111: {g,f,e,d,c,b,a} = 8'b0000111;    //显示 7
            4'b1000: {g,f,e,d,c,b,a} = 8'b1111111;    //显示 8
            4'b1001: {g,f,e,d,c,b,a} = 8'b1101111;    //显示 9
            4'b1010: {g,f,e,d,c,b,a} = 8'b1110111;    //显示 A
            4'b1011: {g,f,e,d,c,b,a} = 8'b1111100;    //显示 B
            4'b1100: {g,f,e,d,c,b,a} = 8'b0111001;    //显示 C
            4'b1101: {g,f,e,d,c,b,a} = 8'b1011110;    //显示 D
            4'b1110: {g,f,e,d,c,b,a} = 8'b1111001;    //显示 E
            4'b1111: {g,f,e,d,c,b,a} = 8'b1110001;    //显示 F
            default: {g,f,e,d,c,b,a} = 8'b0000000;    //不显示
        endcase
    endmodule
```

## 1.4 创建顶层文件

下面用图形法创建顶层文件。

在 Quartus II 平台上,使用图形编辑输入法设计电路的操作流程,包括编辑、编译、仿真和编程下载等基本过程。用 Quartus II 图形编辑方式生成的图形文件的扩展名为 .gdf 或 .bdf。

### 1. 创建图形文件

选择“File”→“New”命令,弹出如图 1.20 所示的对话框,选择“Block Diagram/Schematic File”,单击“OK”按钮。