



# EDA技术实用教程

## 第4章

# Quartus II应用向导

# 4.1 基本设计流程

## 4.1.1 建立工作库文件夹和编辑设计文件

(1) 新建一个文件夹

(2) 输入源程序

(3) 文件存盘

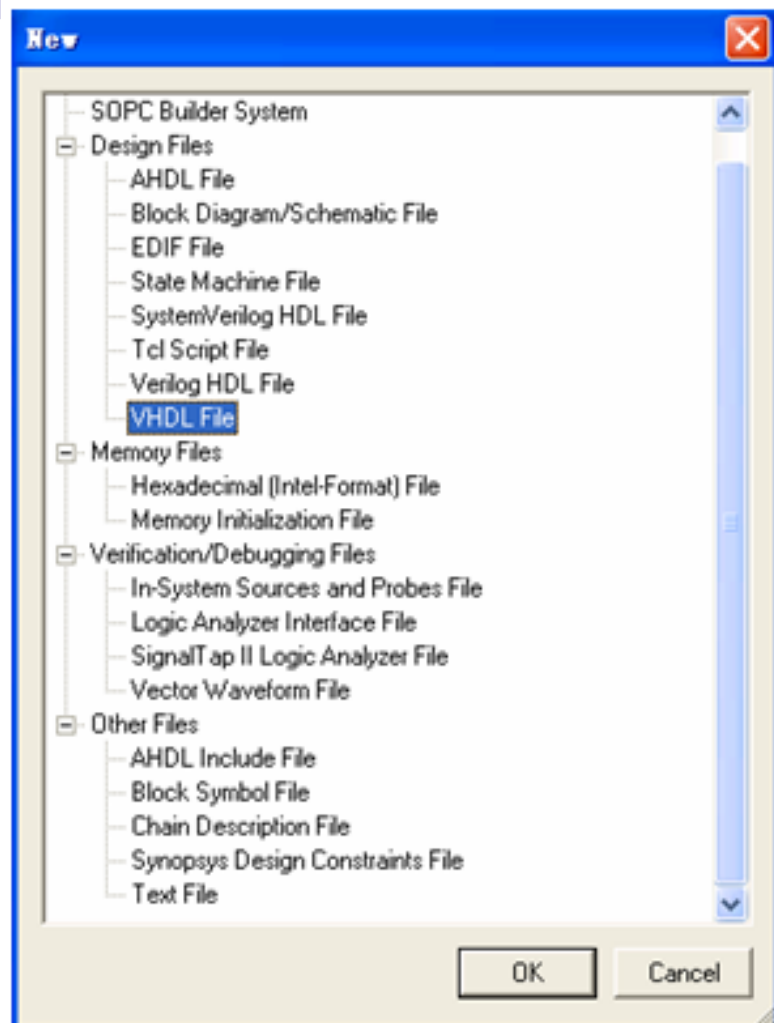


图 4-1 选择编辑文件类型

# 4.1 基本设计流程

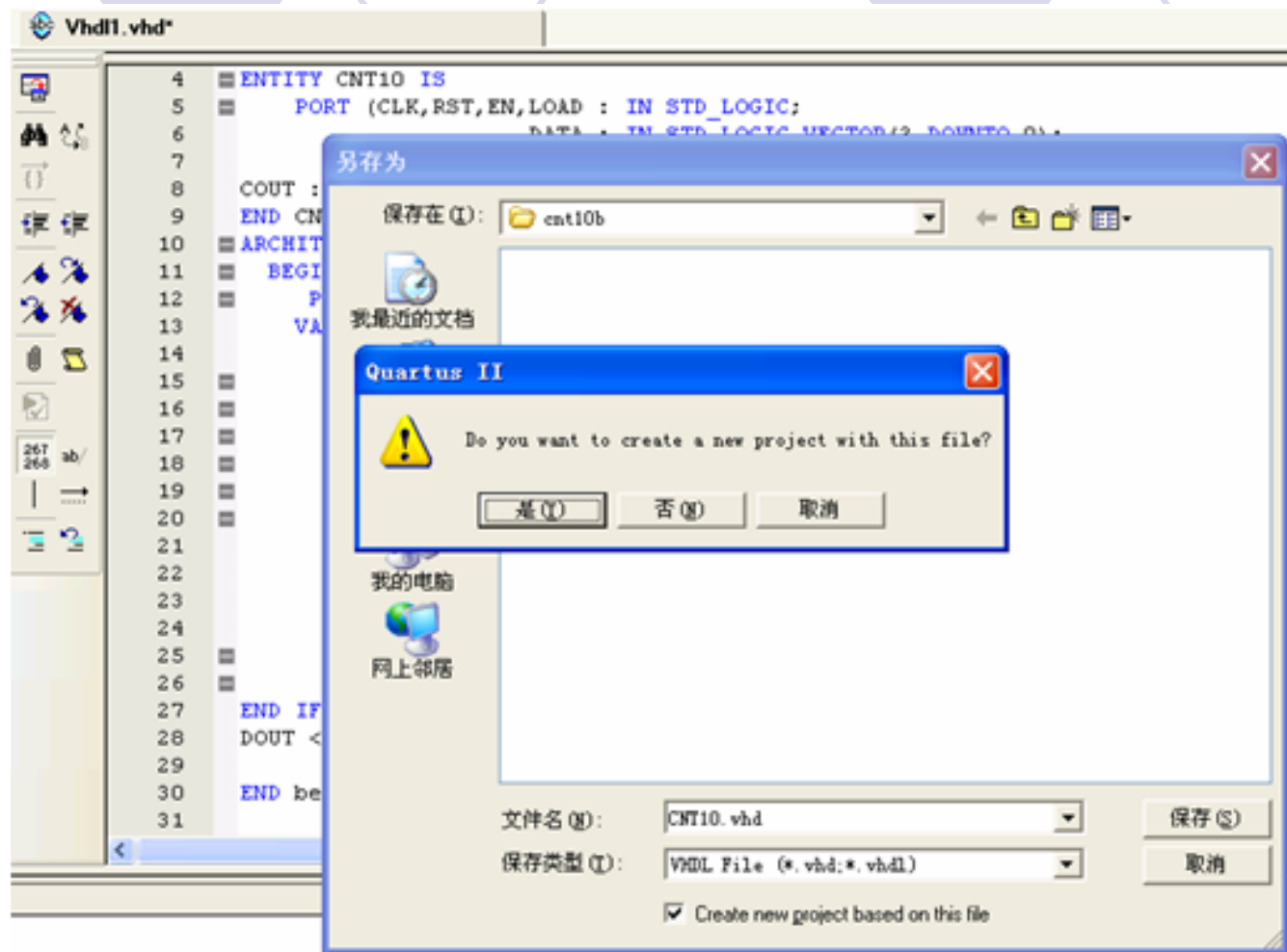


图 4-2 编辑输入源程序并存盘

# 4.1 基本设计流程

## 4.1.2 创建工程

(1) 打开并建立新工程管理窗口



图 4-3 利用 New Project Wizard 创建工程 cnt10

# 4.1 基本设计流程

## 4.1.2 创建工程

(2) 将设计文件加入工程中

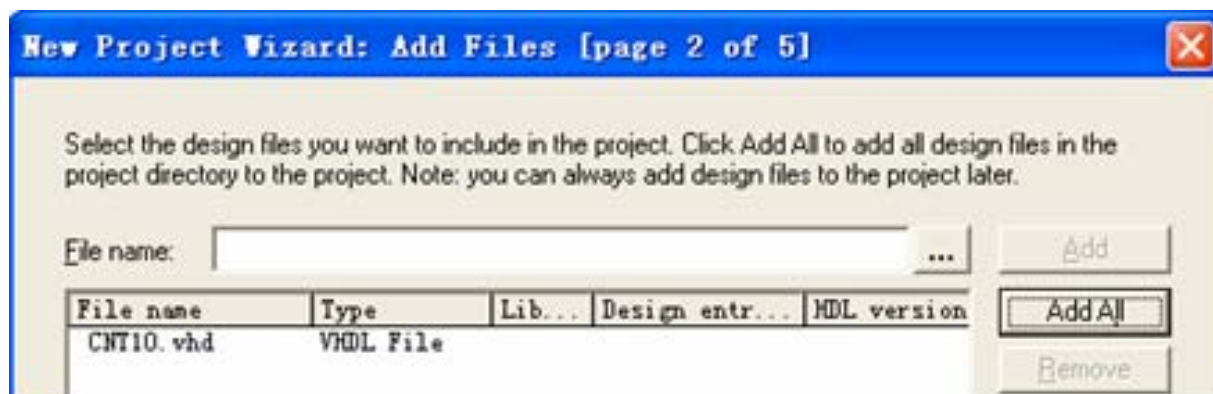


图 4-4 将所有相关的文件都加入进此工程

# 4.1 基本设计流程

## 4.1.2 创建工程

### (3) 选择目标芯片

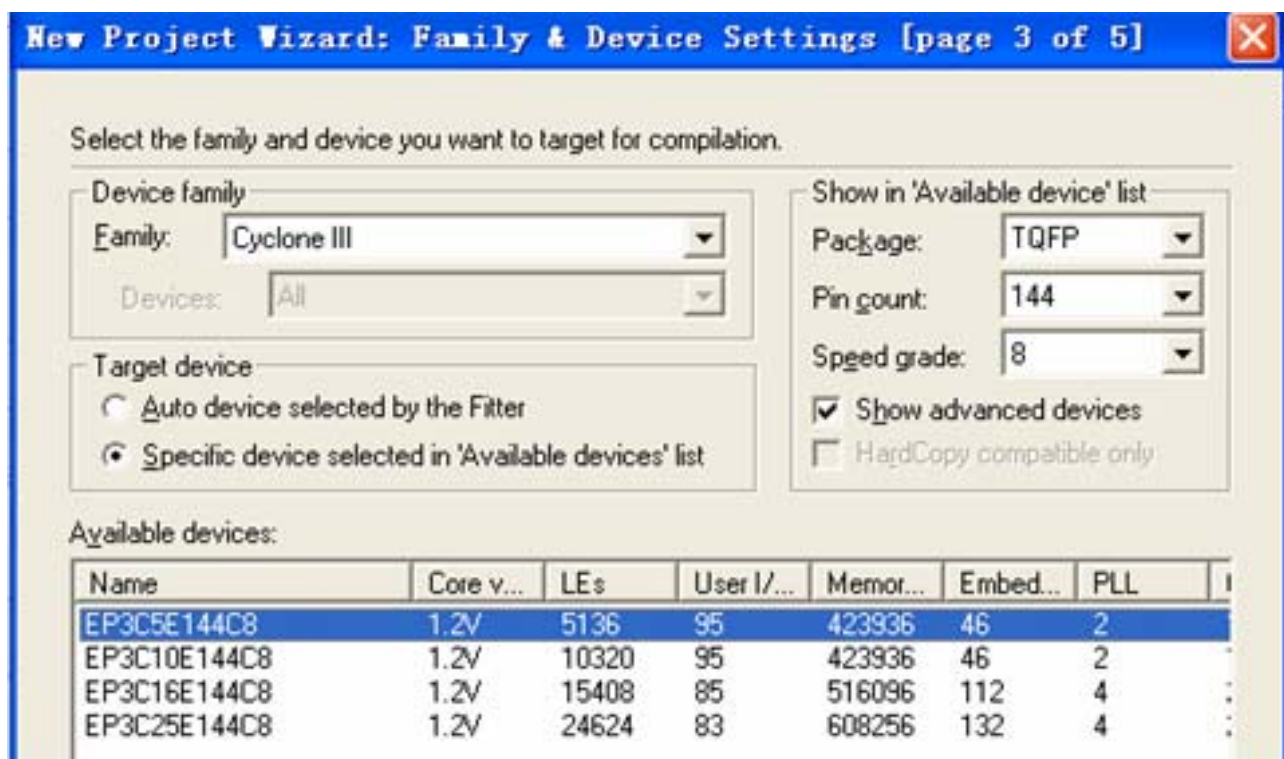
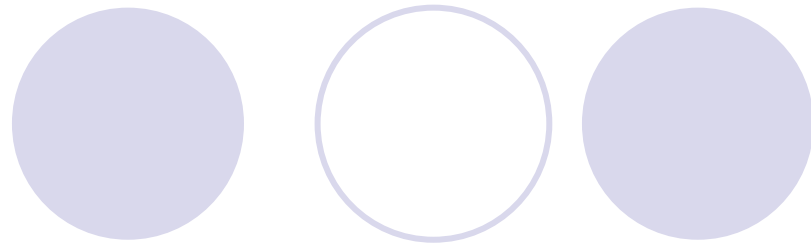


图 4-5 选择目标器件 EP3C5E144C8

# 4.1 基本设计流程



## 4.1.2 创建工程

(4) 工具设置

(5) 结束设置

# 4.1 基本设计流程

## 4.1.3 编译前设置

- (1) 选择**FPGA**目标芯片
- (2) 选择配置器件的工作方式

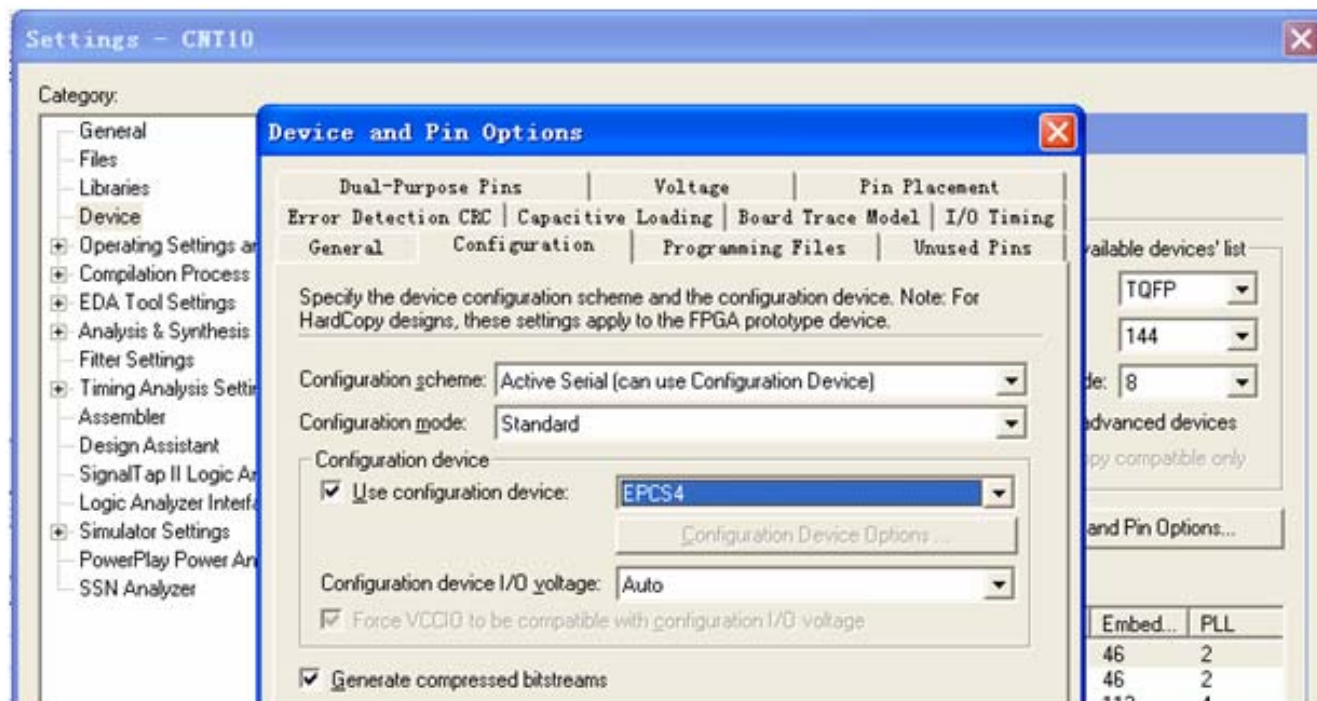


图 4-6 选择配置器件的工作方式



# 4.1 基本设计流程

## 4.1.3 编译前设置

- (3) 选择配置器件和编程方式
- (4) 选择目标器件引脚端口状态

- (5) 选择确认VHDL语言版本

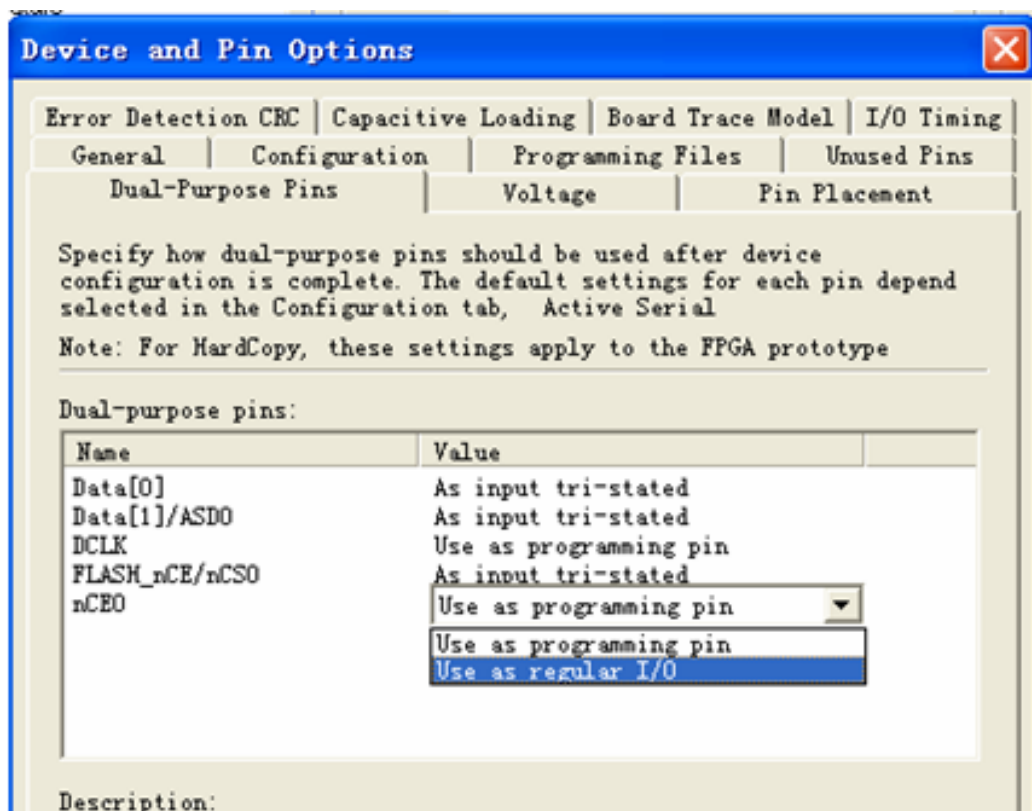


图 4-7 指定 nCEO 脚为普通 I/O pin

# 4.1 基本设计流程

## 4.1.4 全程编译



图 4-8 全程编译无错后的报告信息

# 4.1 基本设计流程

## 4.1.5 时序仿真

(1) 打开波形编辑器

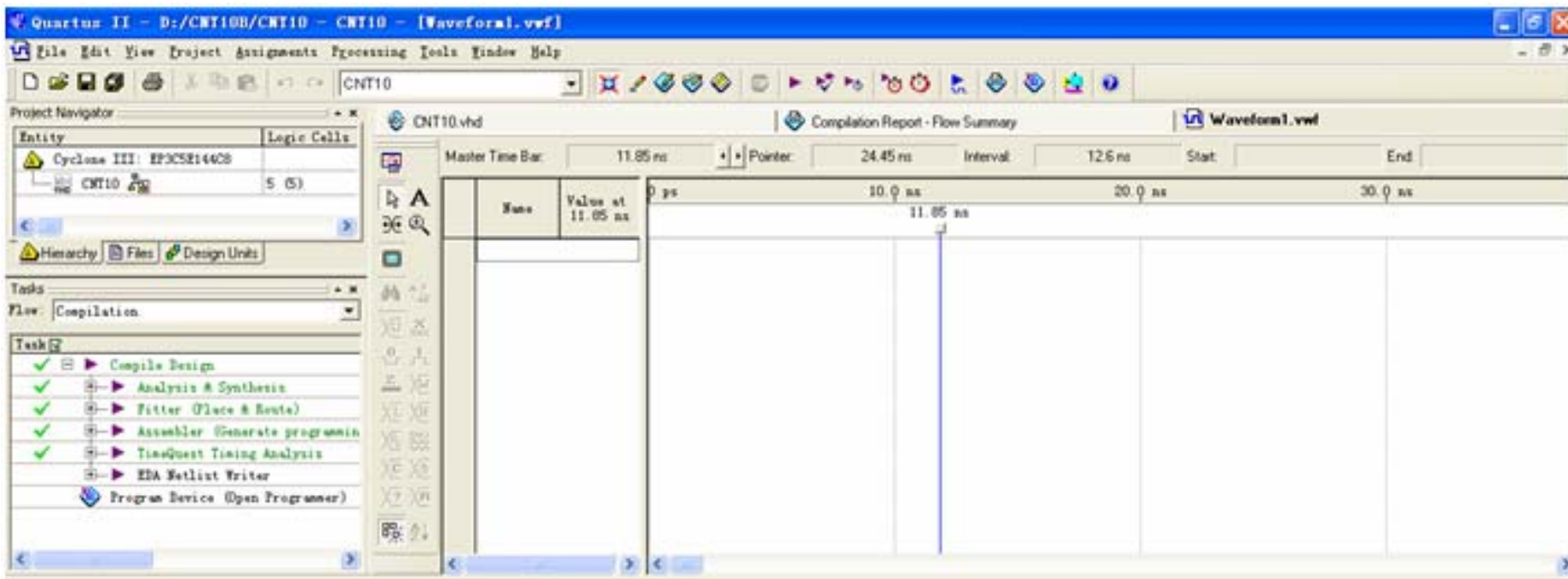


图 4-9 激励信号波形编辑器窗口

# 4.1 基本设计流程

## 4.1.5 时序仿真

(2) 设置仿真时间区域



图 4-10 设置仿真时间长度

(3) 波形文件存盘

# 4.1 基本设计流程

## 4.1.5 时序仿真

(4) 将工程**CNT10**的端口信号节点选入波形编辑器中

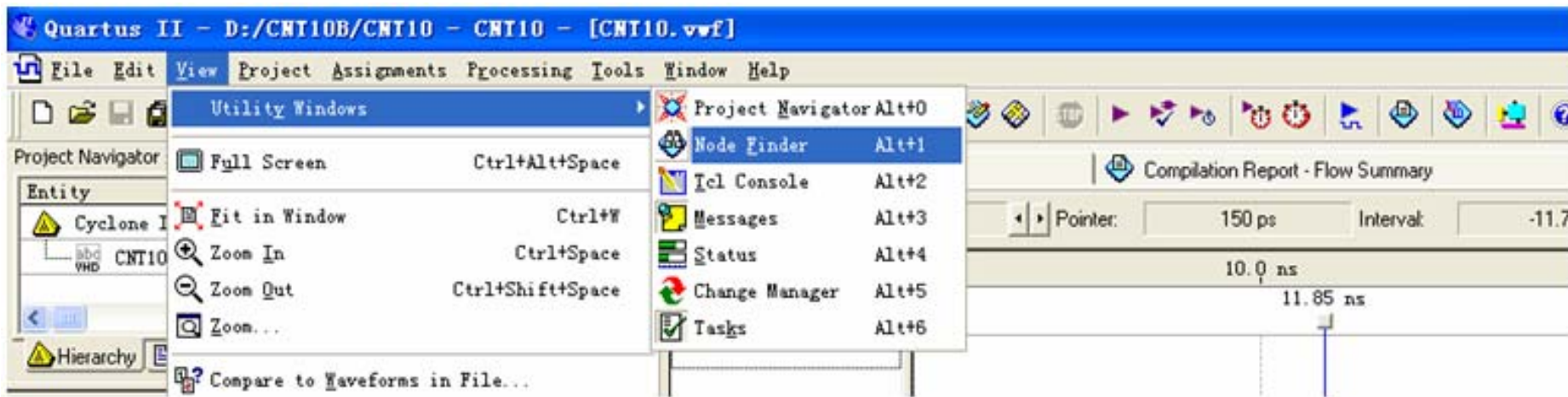


图 4-11 打开信号节点查询窗口

# 4.1 基本设计流程

## 4.1.5 时序仿真

(4) 将工程**CNT10**的端口信号节点选入波形编辑器中

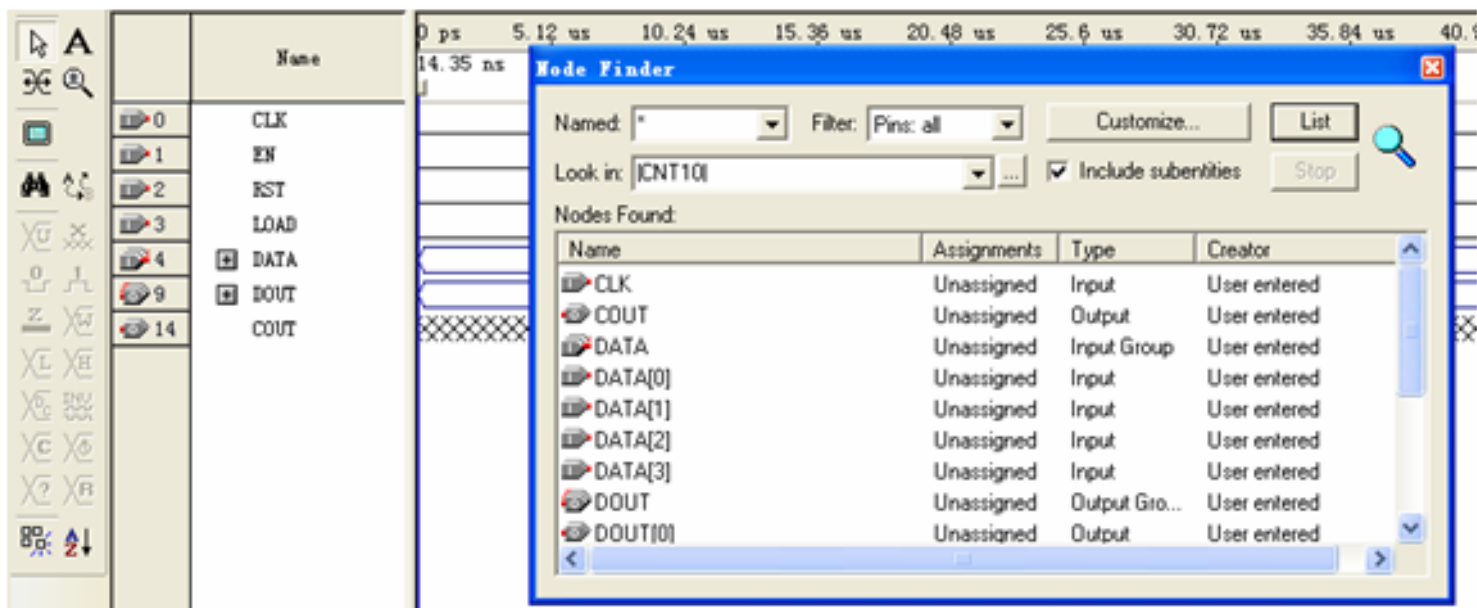


图 4-12 向波形编辑器拖入信号节点



# 4.1 基本设计流程

## 4.1.5 时序仿真

(5) 编辑输入波形(输入激励信号)

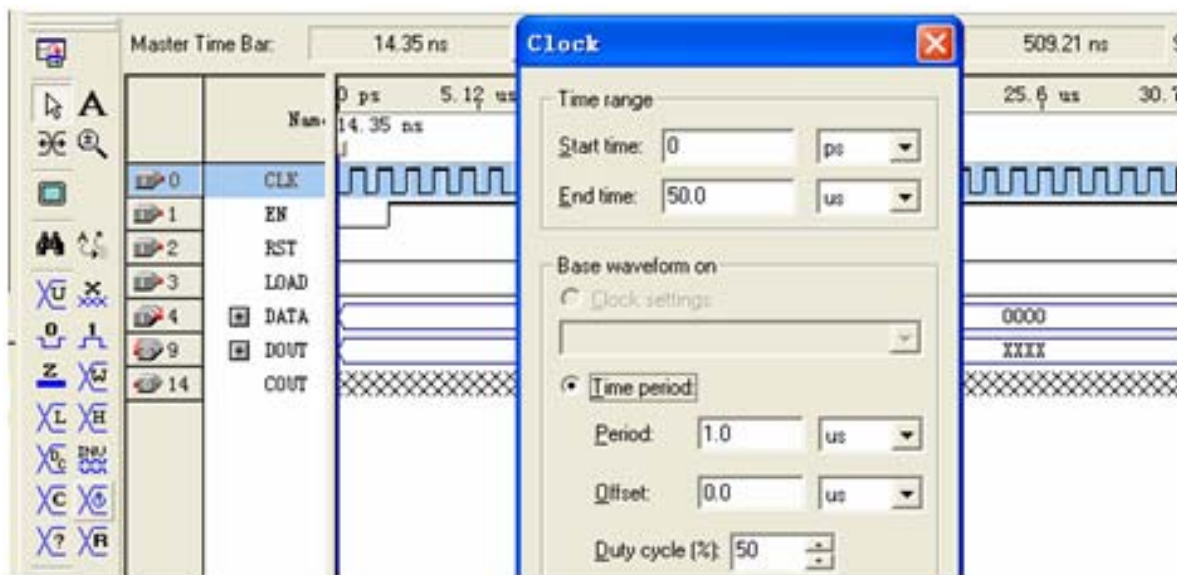


图 4-13 设置时钟 CLK 的周期

# 4.1 基本设计流程

## 4.1.5 时序仿真

### (6) 总线数据格式设置和参数设置

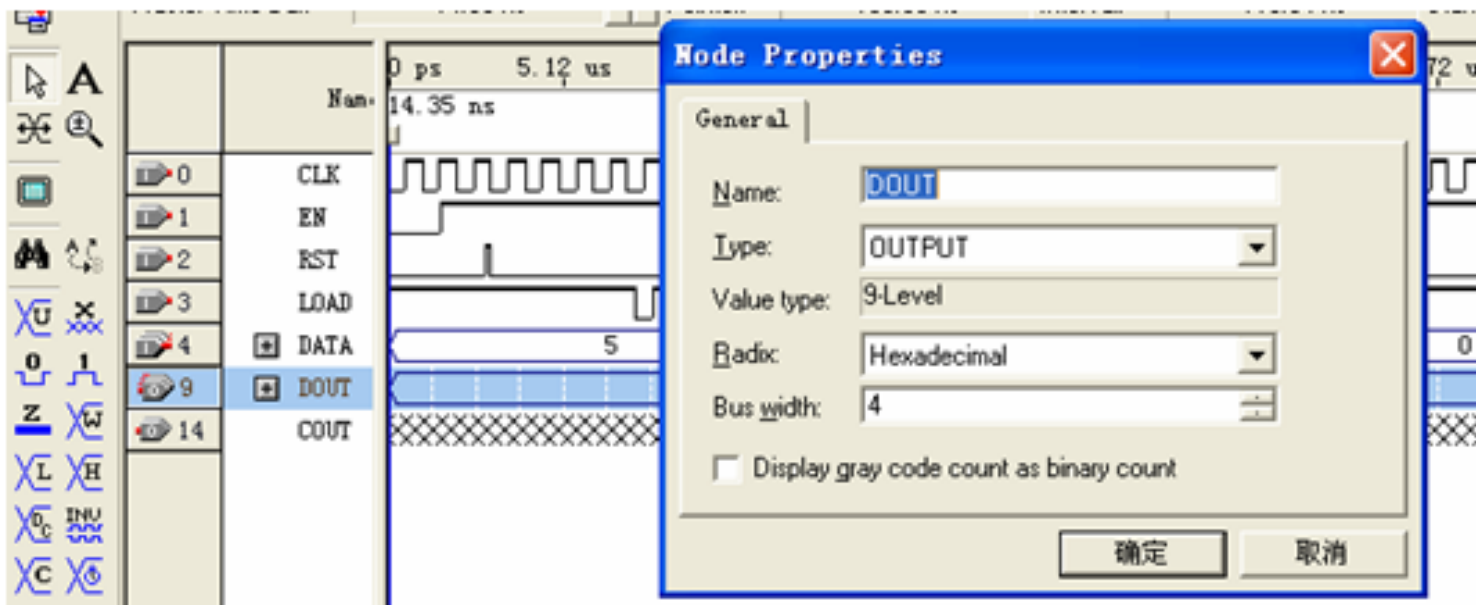


图 4-14 设置好的激励波形图，选择总线数据格式



# 4.1 基本设计流程

## 4.1.5 时序仿真

(6) 总线数据格式设置和参数设置

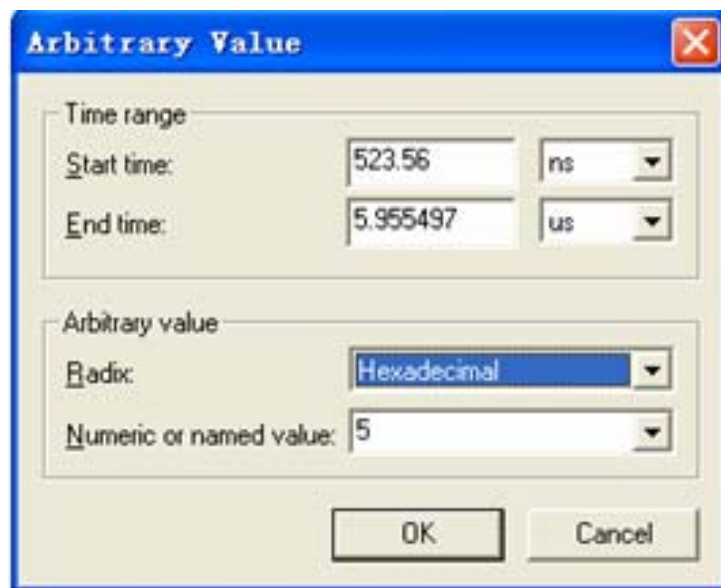
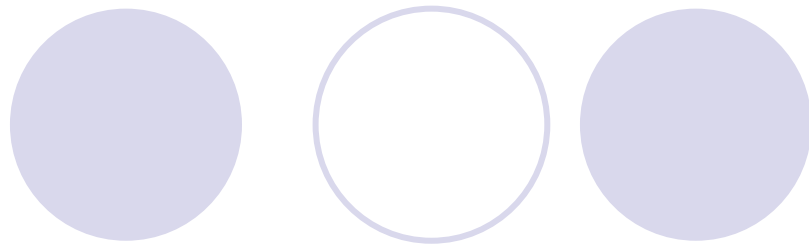


图 4-15 设置输入数据

# 4.1 基本设计流程



## 4.1.5 时序仿真

(6) 总线数据格式设置和参数设置

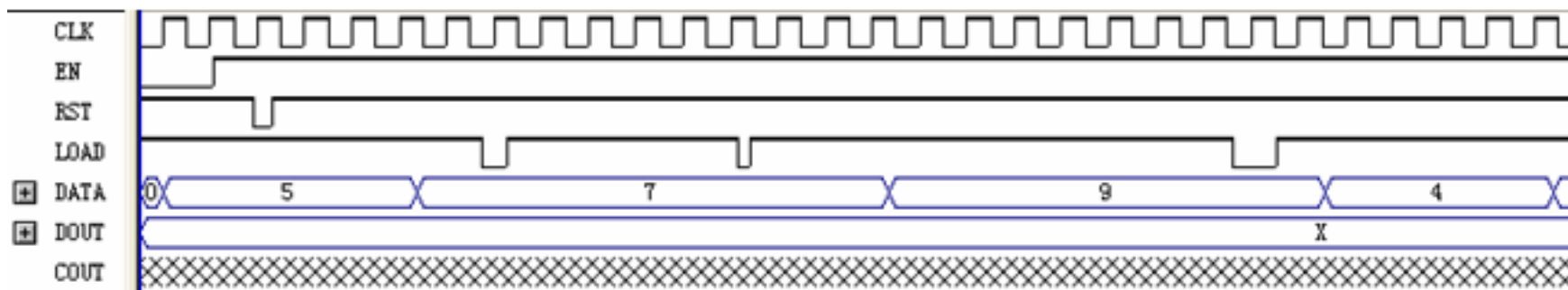


图 4-16 最后设置好的 vwf 仿真激励波形文件图

# 4.1 基本设计流程

## 4.1.5 时序仿真

### (7) 仿真器参数设置

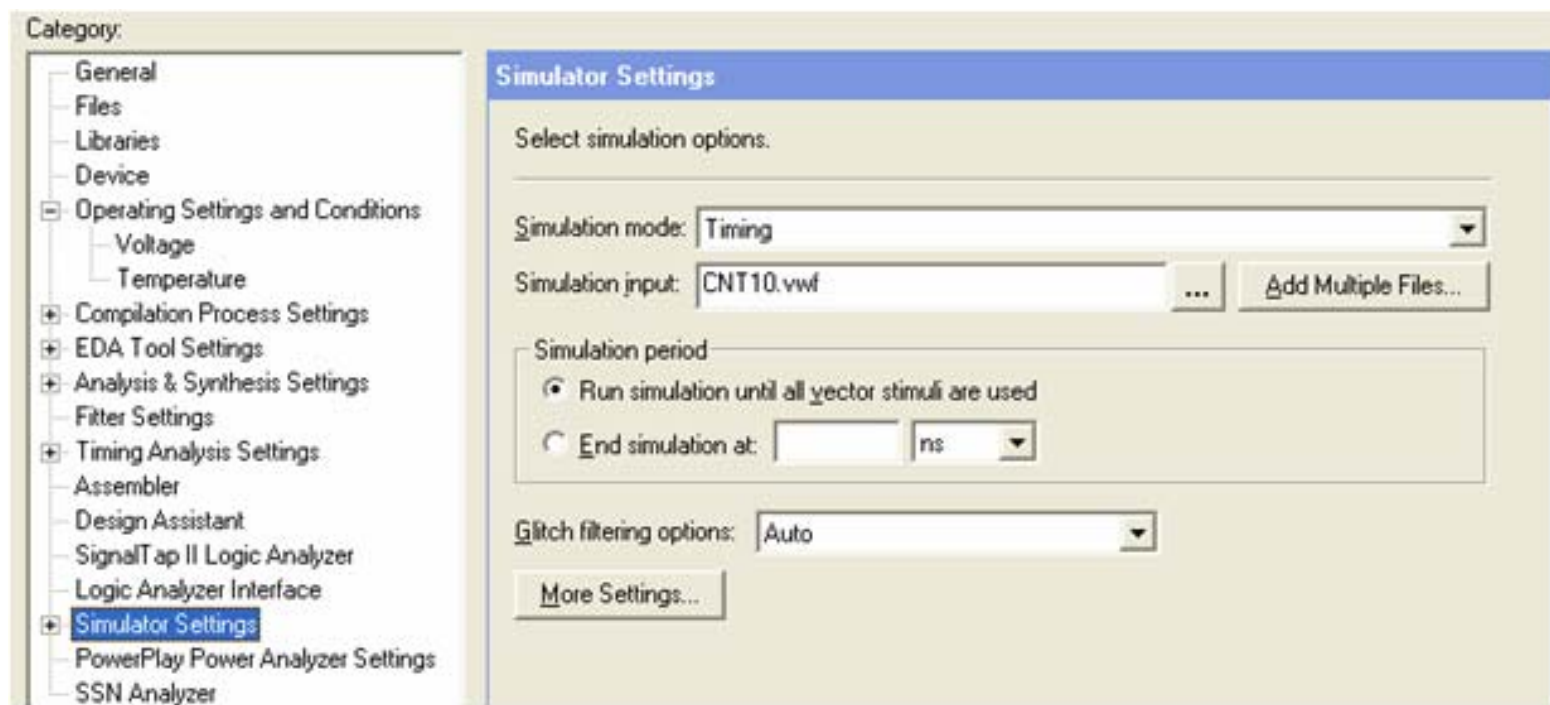
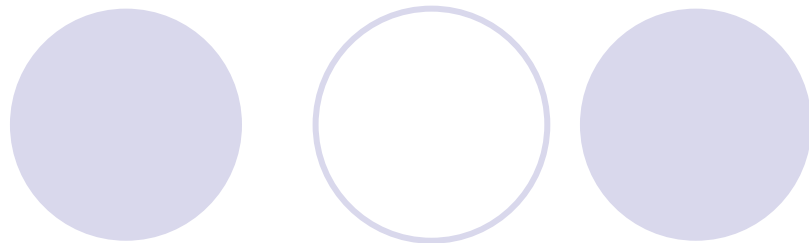


图 4-17 选择仿真模式

# 4.1 基本设计流程



## 4.1.5 时序仿真

(8) 启动仿真器

(9) 观察仿真结果

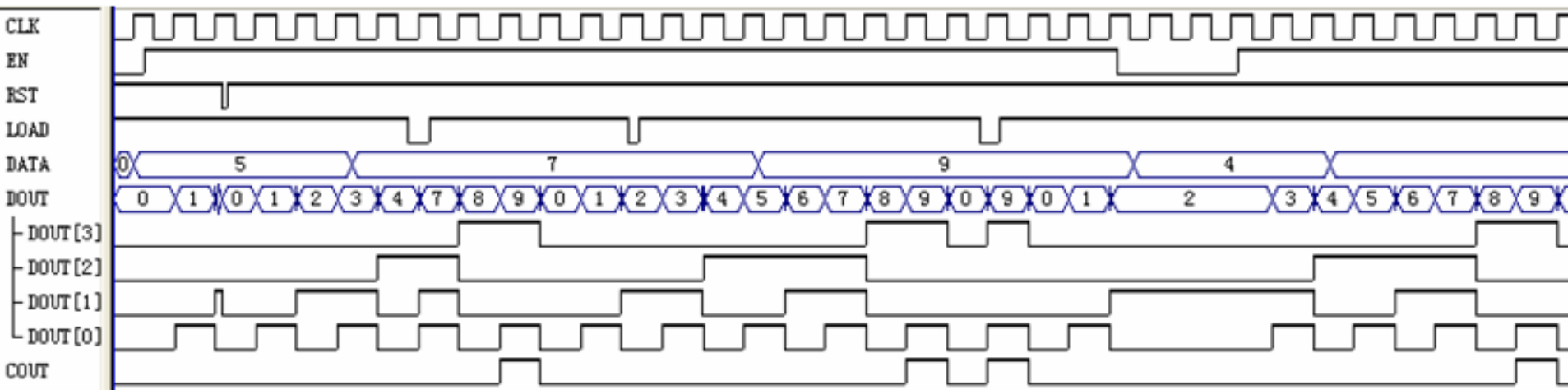


图 4-18 仿真波形输出报告 Simulation Report

# 4.1 基本设计流程

## 4.1.6 应用RTL电路图观察器

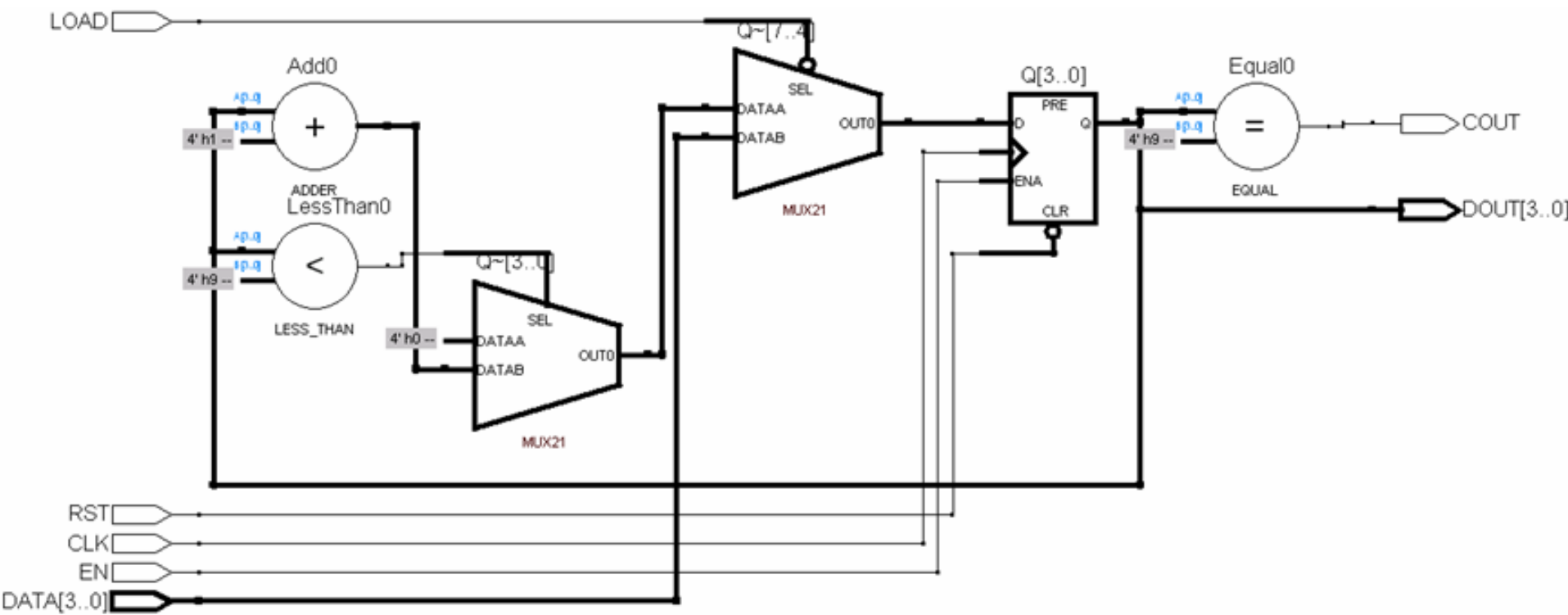


图 3-14 例 3-22 的 RTL 电路图

# 4.2 引脚设置与硬件验证

## 4.2.1 引脚锁定

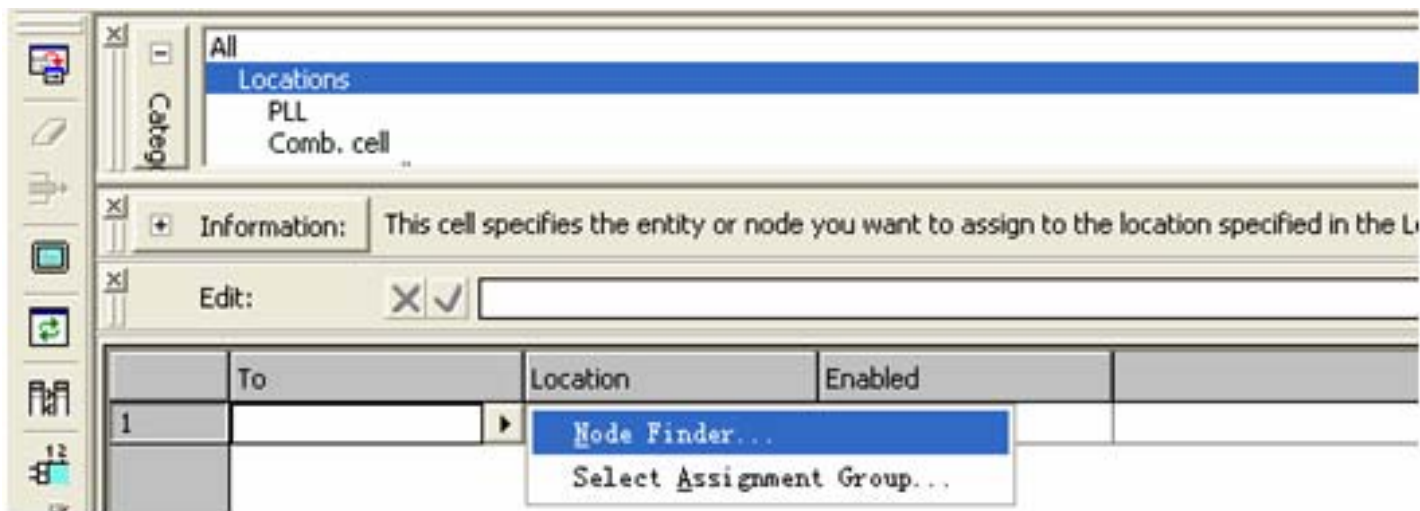


图 4-19 利用 Assignment Editor 编辑器锁定 FPGA 引脚

# 4.2 引脚设置与硬件验证

## 4.2.1 引脚锁定

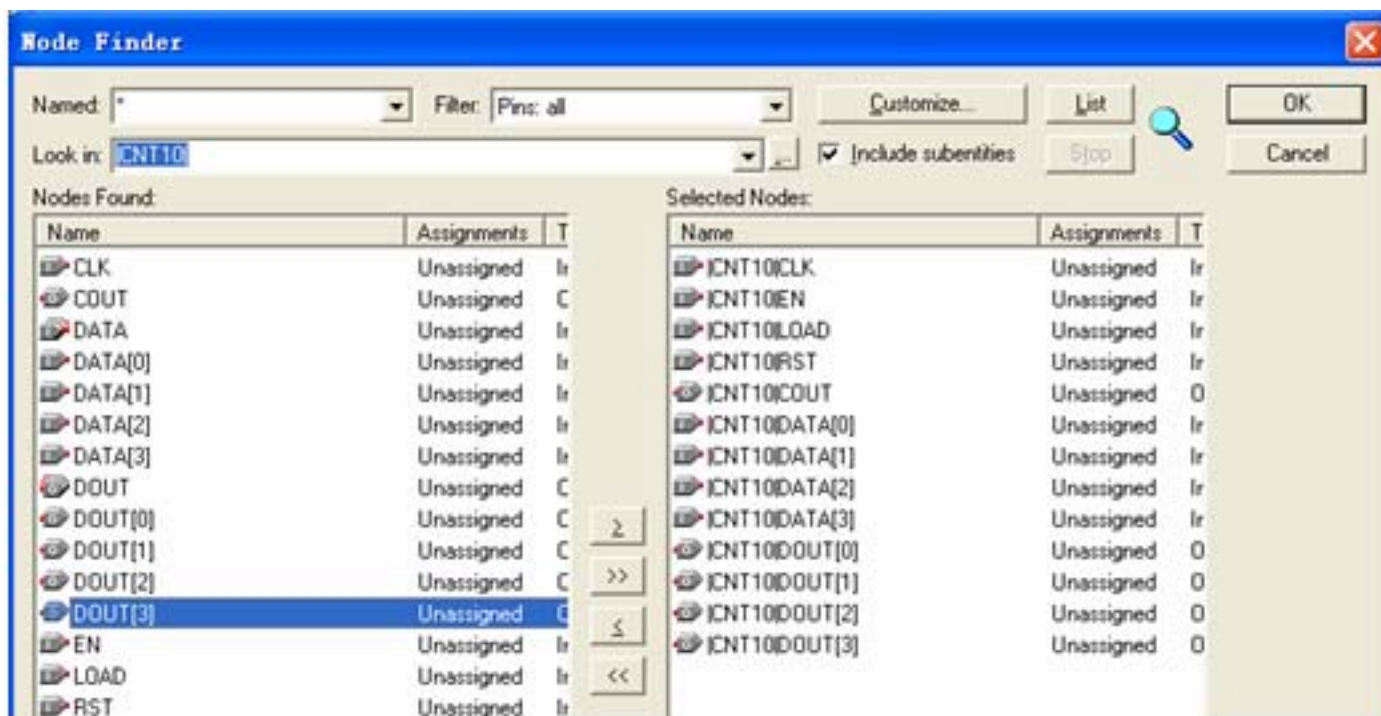


图 4-20 利用 Node Finder 工具选择需要锁定引脚的信号

## 4.2 引脚设置与硬件验证

### 4.2.1 引脚锁定













|    | To  | Location | Enabled |
|----|---|----------|---------|
| 1  |  CLK       | PIN_69   | Yes     |
| 2  |  EN        | PIN_68   | Yes     |
| 3  |  RST       | PIN_66   | Yes     |
| 4  |  LOAD      | PIN_67   | Yes     |
| 5  |  DATA[0]   | PIN_88   | Yes     |
| 6  |  DATA[1]   | PIN_89   | Yes     |
| 7  |  DATA[2]   | PIN_90   | Yes     |
| 8  |  DATA[3]   | PIN_91   | Yes     |
| 9  |  DOUT[0]   | PIN_32   | Yes     |
| 10 |  DOUT[1]  | PIN_46   | Yes     |
| 11 |  DOUT[2] | PIN_44   | Yes     |
| 12 |  DOUT[3] | PIN_43   | Yes     |
| 13 | <<new>>   |          |         |

图 4-21 表格方式引脚锁定窗口



# 4.2 引脚设置与硬件验证

## 4.2.1 引脚锁定

表 4-1 基于 KX-7C5E+开发板的 EP3C5 开发板的引脚锁定表

| 信号控制端  | 拨码 4    | 拨码 3    | 拨码 2    | 拨码 1    | 键 K5    | 键 K6    | 键 K7    | 键 K8    |
|--------|---------|---------|---------|---------|---------|---------|---------|---------|
| 功能信号设定 | D[3]    | D[2]    | D[1]    | D[0]    | RST     | LOAD    | EN      | CLK     |
| 引脚编号   | Pin: 91 | Pin: 90 | Pin: 89 | Pin: 88 | Pin: 66 | Pin: 67 | Pin: 68 | Pin: 69 |
| 功能信号设定 | DOUT3   | DOUT2   | DOUT1   | DOUT0   | COUT    |         |         |         |
| 引脚编号   | Pin:43  | Pin:44  | Pin:46  | Pin:32  | Pin:144 |         |         |         |

# 4.2 引脚设置与硬件验证

## 4.2.2 编译文件下载

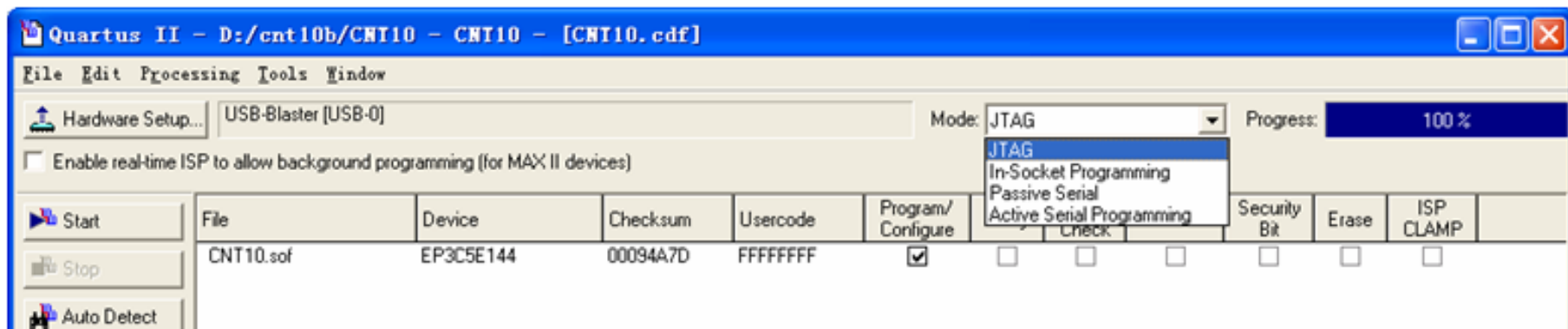


图 4-22 选择编程下载文件

# 4.2 引脚设置与硬件验证

## 4.2.2 编译文件下载



图 4-23 加入编程下载方式

# 4.2 引脚设置与硬件验证

## 4.2.2 编译文件下载

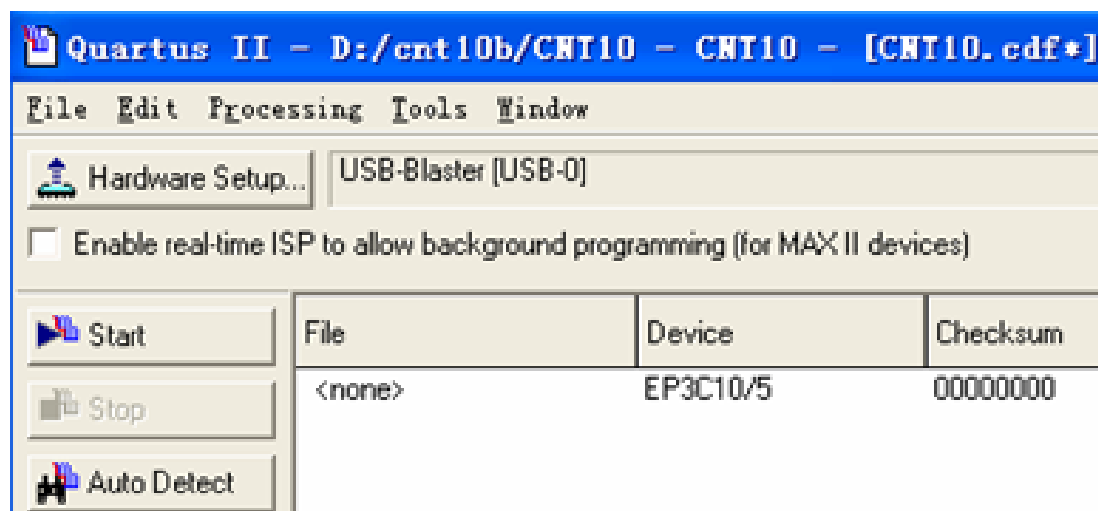


图 4-24 检测实验板上的 FPGA 器件

## 4.2 引脚设置与硬件验证

### 4.2.3 AS模式编程

### 4.2.4 JTAG间接模式编程配置器件

1. 将SOF文件转化为JTAG间接配置文件

## 4.2 引脚设置与硬件验证

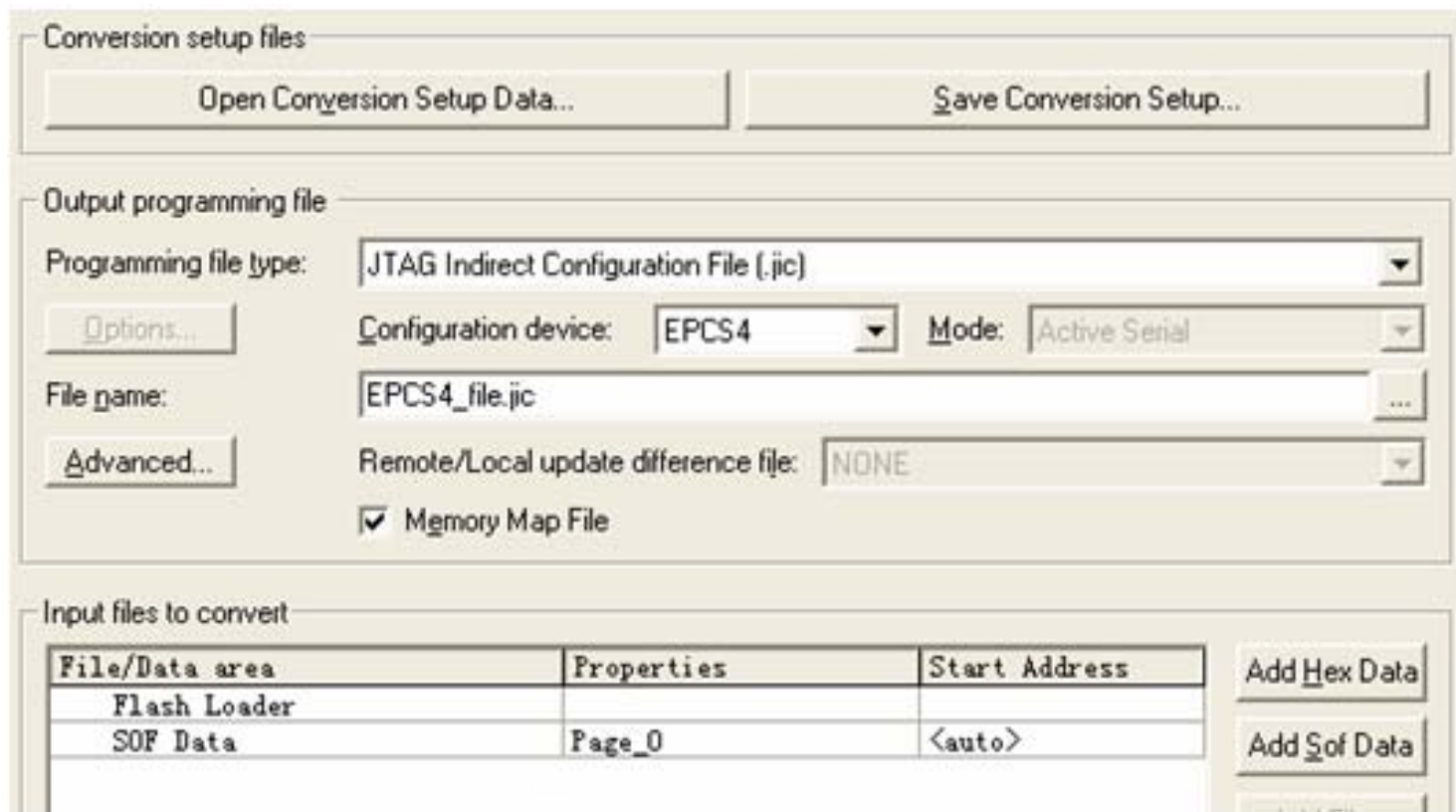


图 4-25 设定 JTAG 间接编程文件

## 4.2 引脚设置与硬件验证

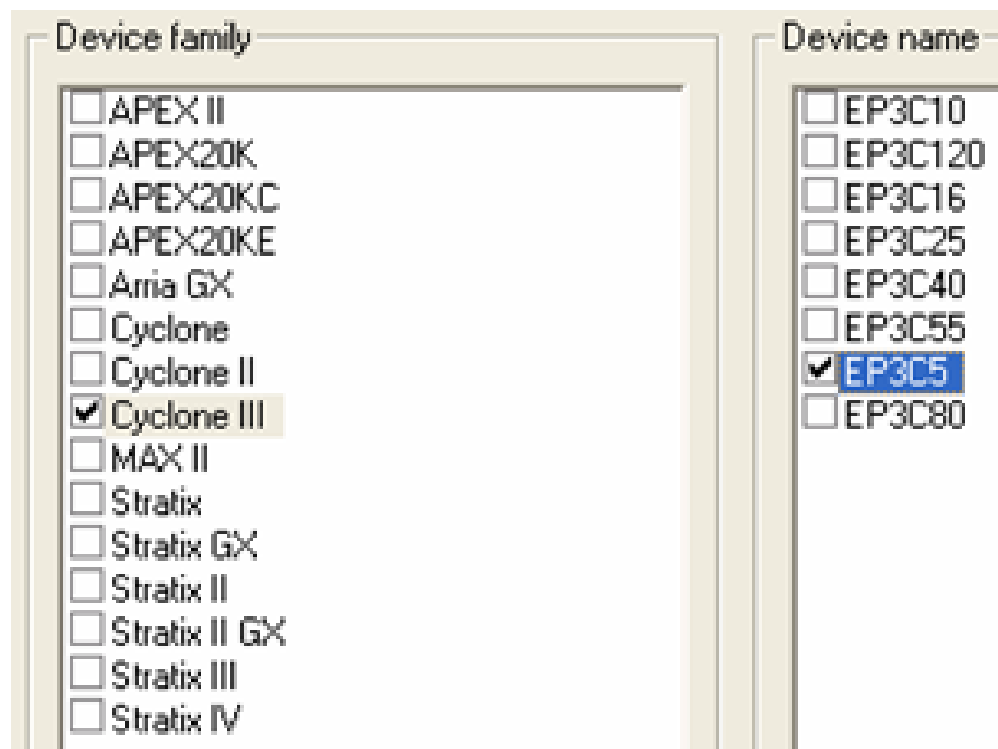


图 4-26 选择目标器件 EP3C40

## 4.2 引脚设置与硬件验证

### 4.2.4 JTAG间接模式编程配置器件

#### 1. 将SOF文件转化为JTAG间接配置文件

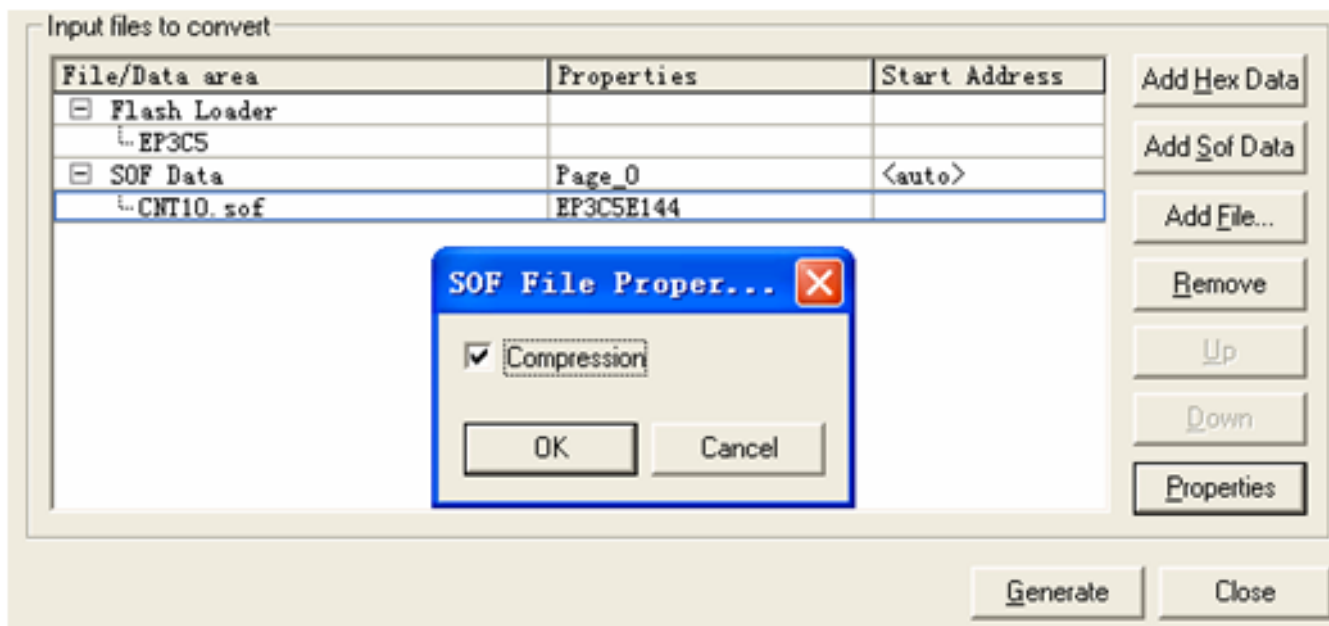


图 4-27 选定 SOF 文件后，选择文件压缩



# 4.2 引脚设置与硬件验证

## 4.2.4 JTAG间接模式编程配置器件

### 2. 下载JTAG间接配置文件

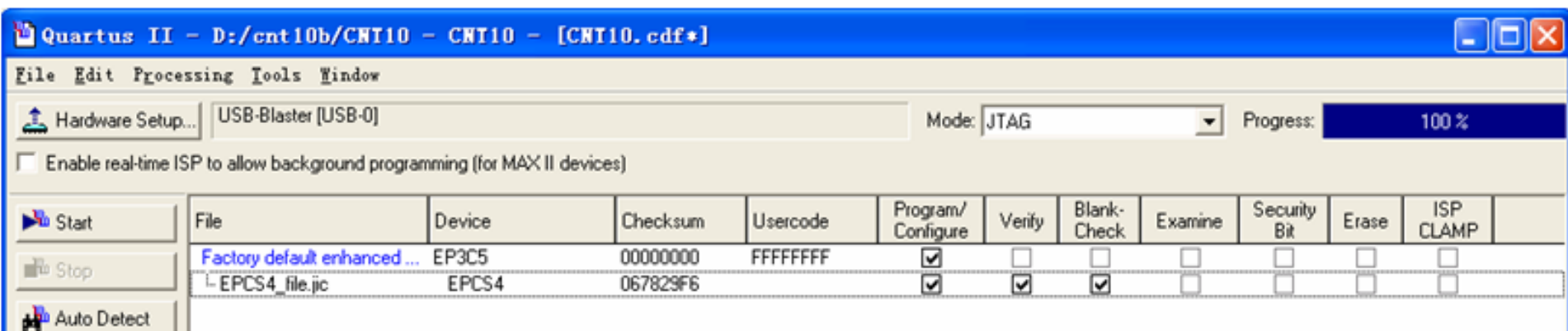


图 4-28 用 JTAG 模式对配置器件 EPCS4 进行间接编程

# 4.2 引脚设置与硬件验证

## 4.2.5 USB-Blaster编程配置器件使用方法

## 4.2.6 其他的锁定引脚方法

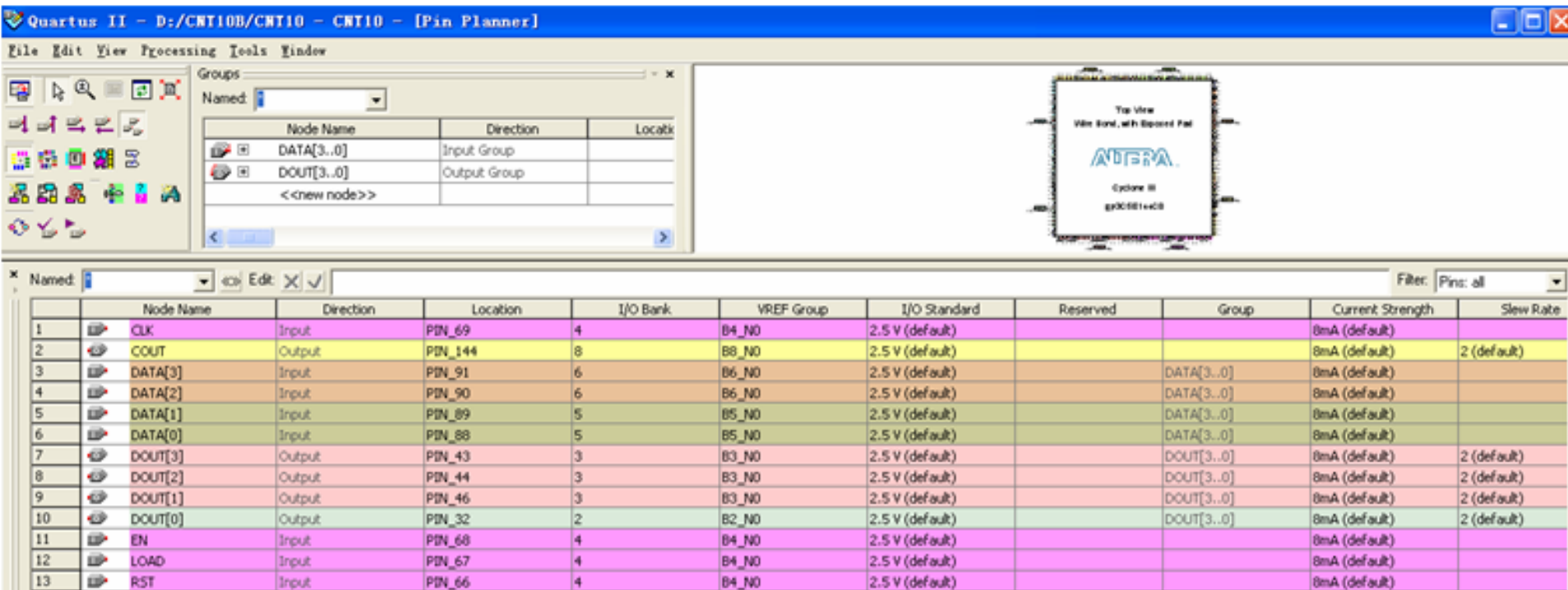


图 4-29 图形方式引脚锁定窗口

## 4.2 引脚设置与硬件验证

### 4.2.6 其他的锁定引脚方法

#### 【例 4-1】对应基于 EP3C5 的 5E+系统

```
ARCHITECTURE behav OF CNT10 IS
```

```
attribute chip_pin : string; -- chip_pin 被定义为字符串数据类型 string
```

```
attribute chip_pin of CLK : signal is "69"; --无抖动信号锁定在 P25
```

```
attribute chip_pin of RST : signal is "66";
```

```
attribute chip_pin of EN : signal is "68";
```

```
attribute chip_pin of LOAD : signal is "67";
```

```
attribute chip_pin of COUT : signal is "144";
```

```
attribute chip_pin of DATA : signal is "91, 90, 89, 88";
```

```
attribute chip_pin of DOUT : signal is "43, 44, 46, 32";
```

```
BEGIN
```

# 4.3 嵌入式逻辑分析仪使用方法

## 1. 打开SignalTap II编辑窗口

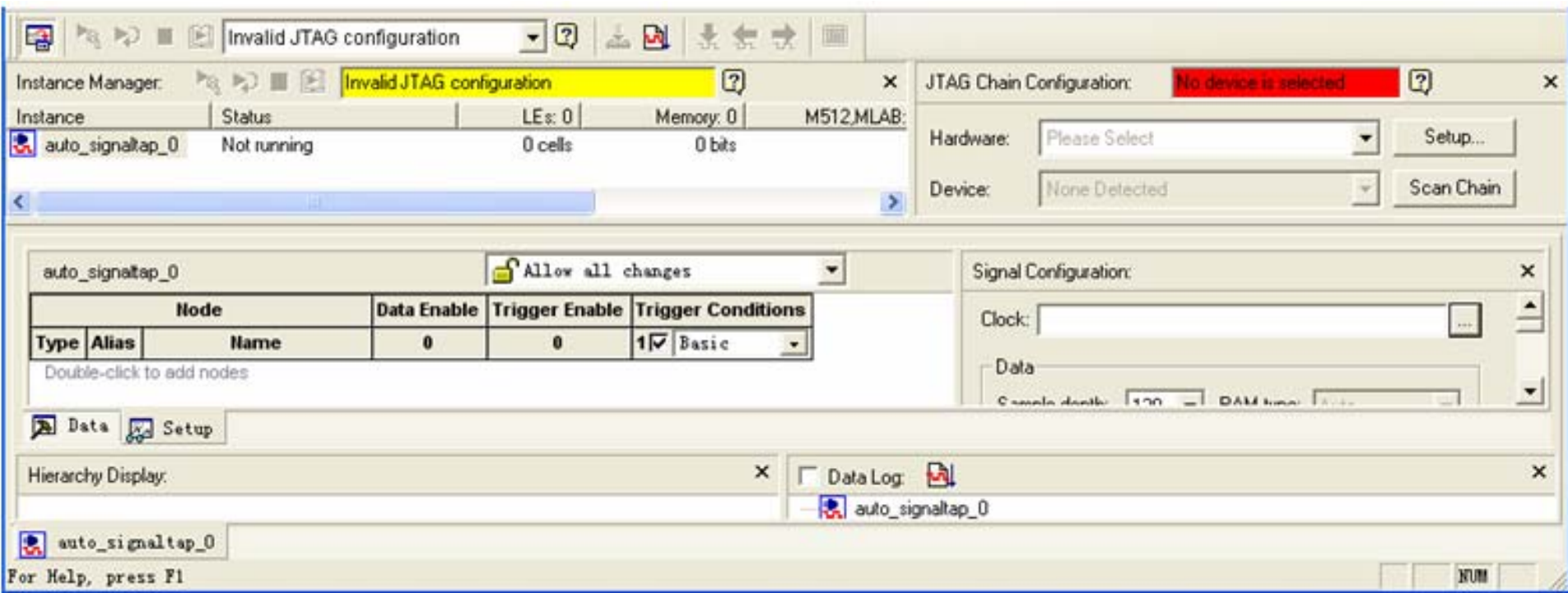


图 4-30 SignalTap II 编辑窗口

# 4.3 嵌入式逻辑分析仪使用方法

## 2. 调入待测信号

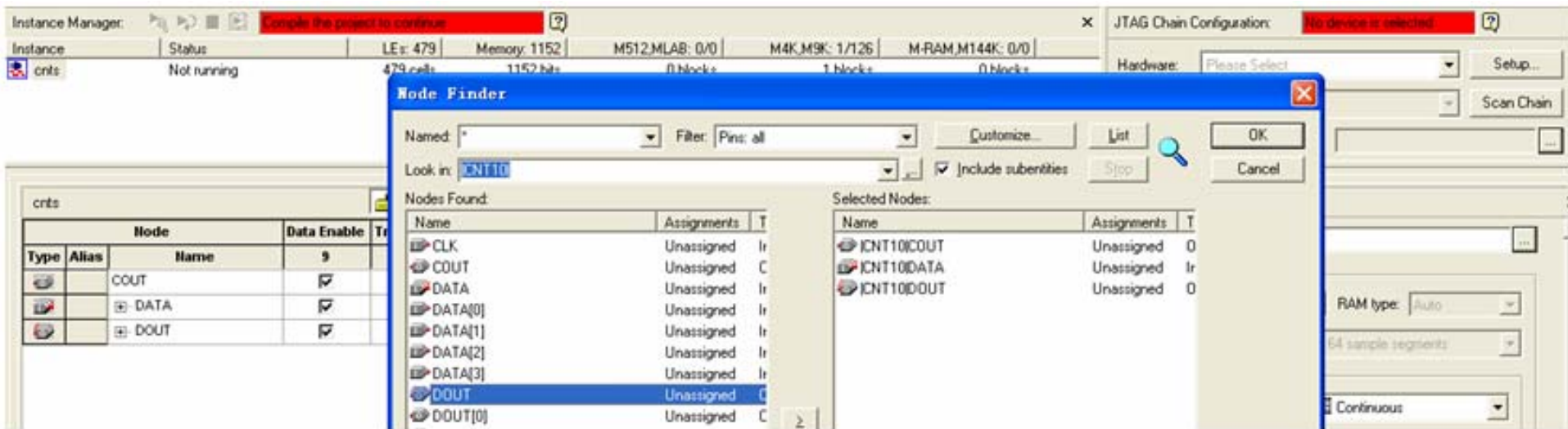


图 4-31 输入逻辑分析仪测试信号

# 4.3 嵌入式逻辑分析仪使用方法

## 2. 调入待测信号

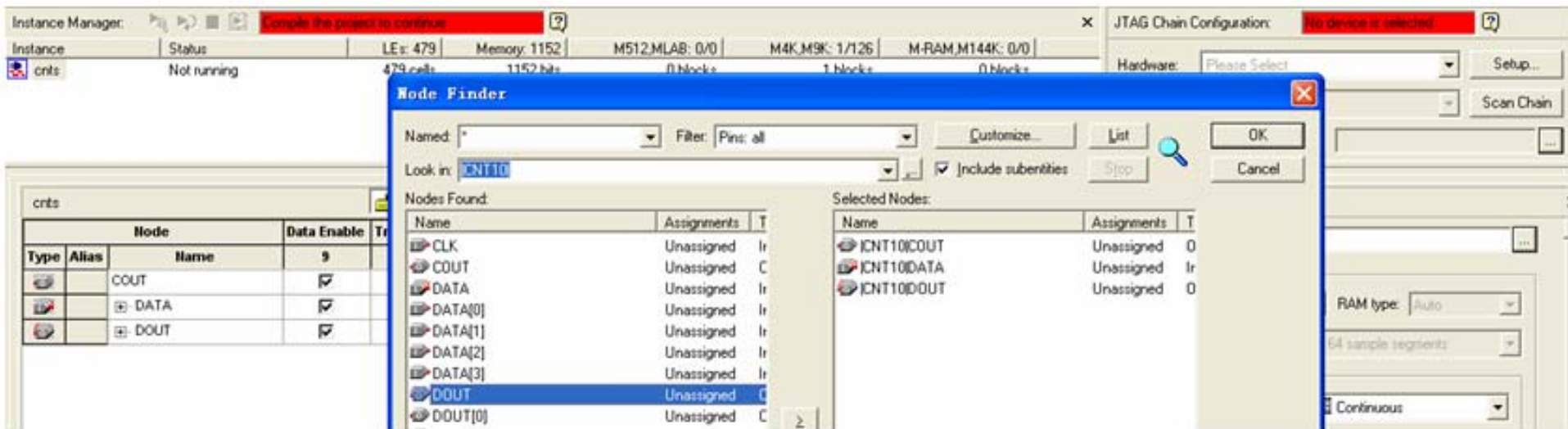


图 4-31 输入逻辑分析仪测试信号



# 4.3 嵌入式逻辑分析仪使用方法

## 3. SignalTap II参数设置

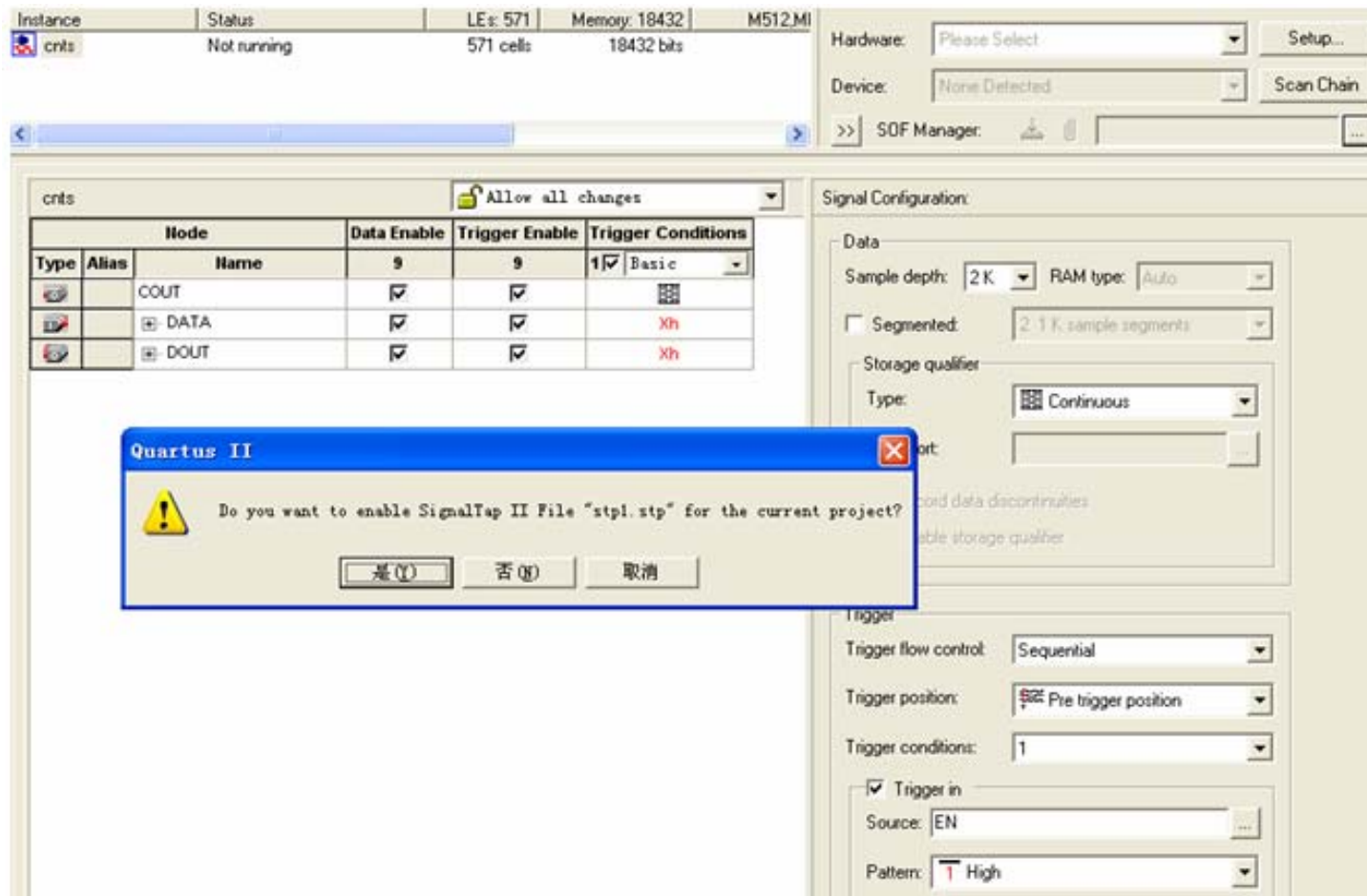


图 4-32 SignalTap II 编辑窗口

# 4.3 嵌入式逻辑分析仪使用方法

## 4. 文件存盘

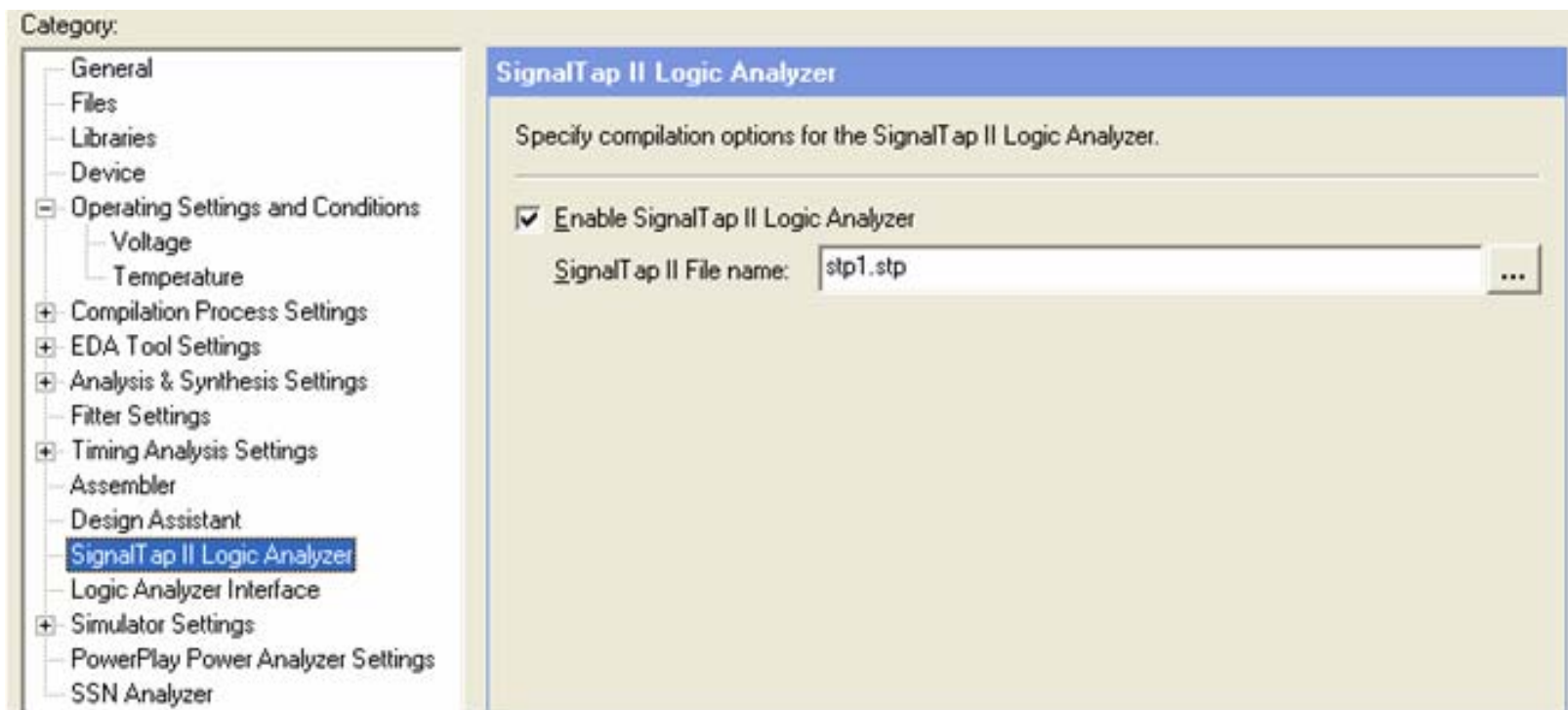


图 4-33 选择或删除 SignalTap II 文件加入综合编译



# 4.3 嵌入式逻辑分析仪使用方法

## 5. 编译下载

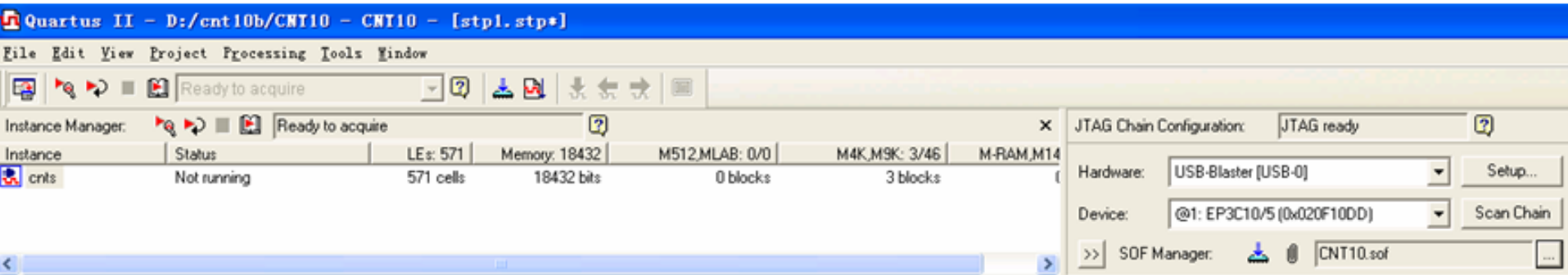


图 4-34 设定 SignalTap II 对 FPGA 的通信接口

# 4.3 嵌入式逻辑分析仪使用方法

## 6. 启动SignalTap II进行采样与分析

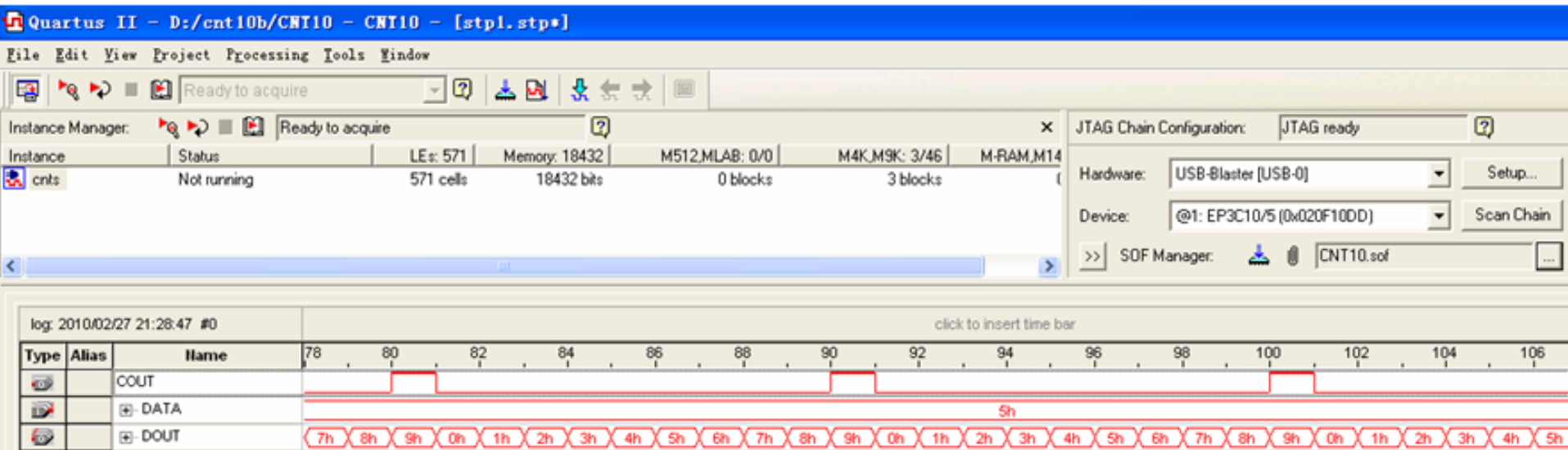


图 4-35 下载 CNT10.sof 并启动 SignalTap II

## 4.3 嵌入式逻辑分析仪使用方法

### 6. 启动SignalTap II进行采样与分析

attribute chip\_pin of CLK0 : signal is "25"; -- 逻辑分析仪采样时钟

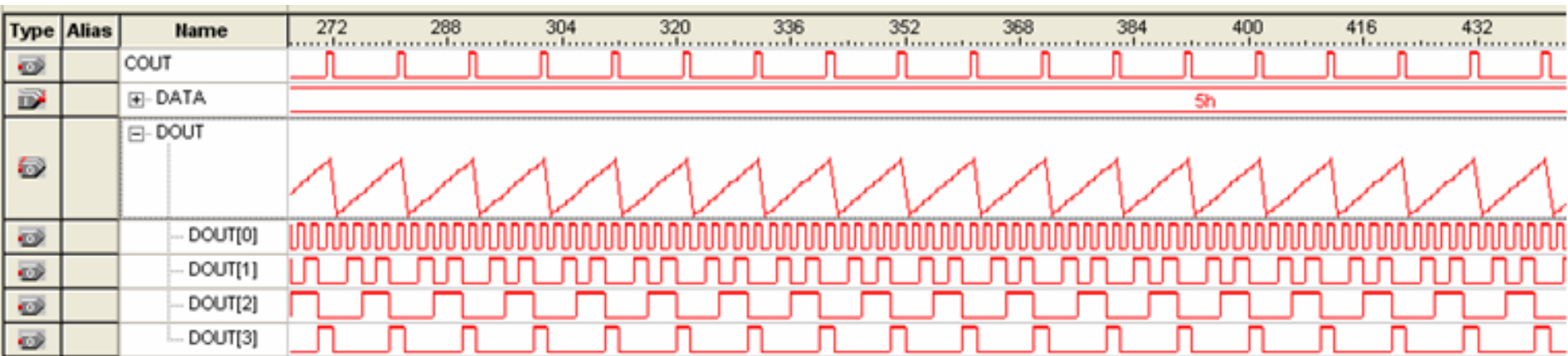


图 4-36 SignalTap II 数据窗口设置后的信号波形

### 7. SignalTap II的其他设置和控制方法

## 4.4 编辑SignalTap II的触发信号

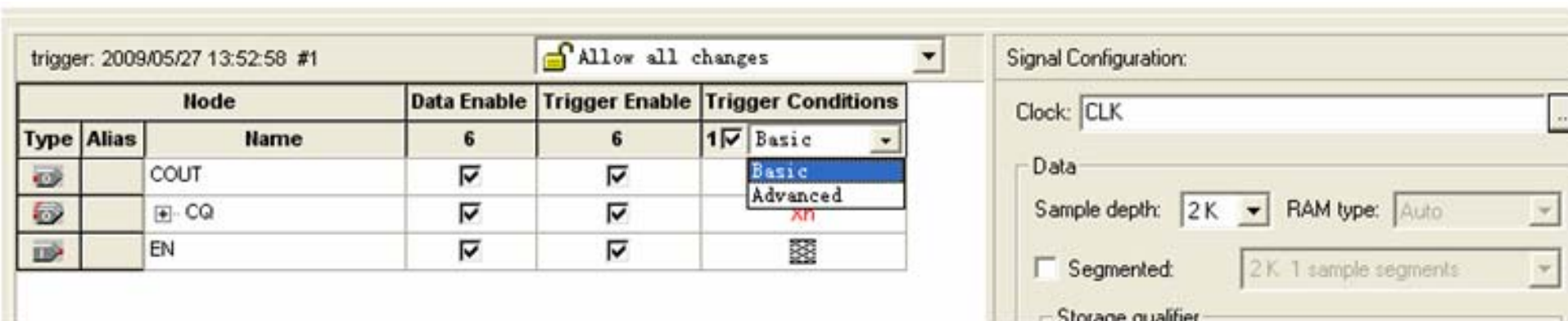


图 4-37 选择高级触发条件

## 4.4 编辑SignalTap II的触发信号

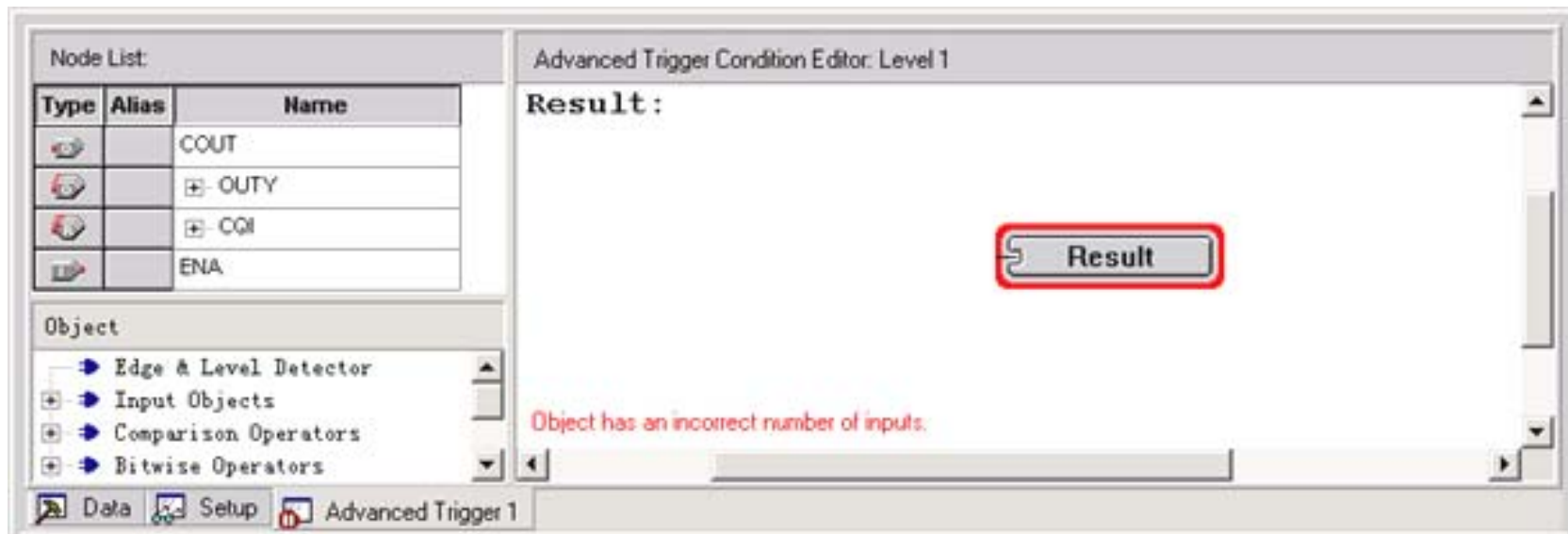


图 4-38 进入触发条件函数编辑窗口

## 4.4 编辑SignalTap II的触发信号

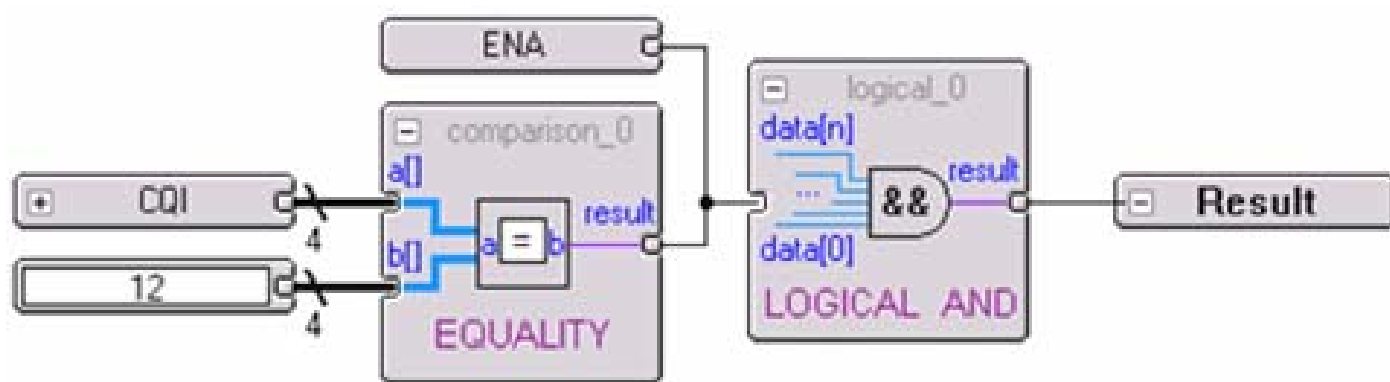


图 4-39 编辑触发函数

# 4.5 原理图输入设计方法

## 4.5.1 层次化设计流程

1. 为本项工程设计建立文件夹
2. 建立原理图文件工程和仿真

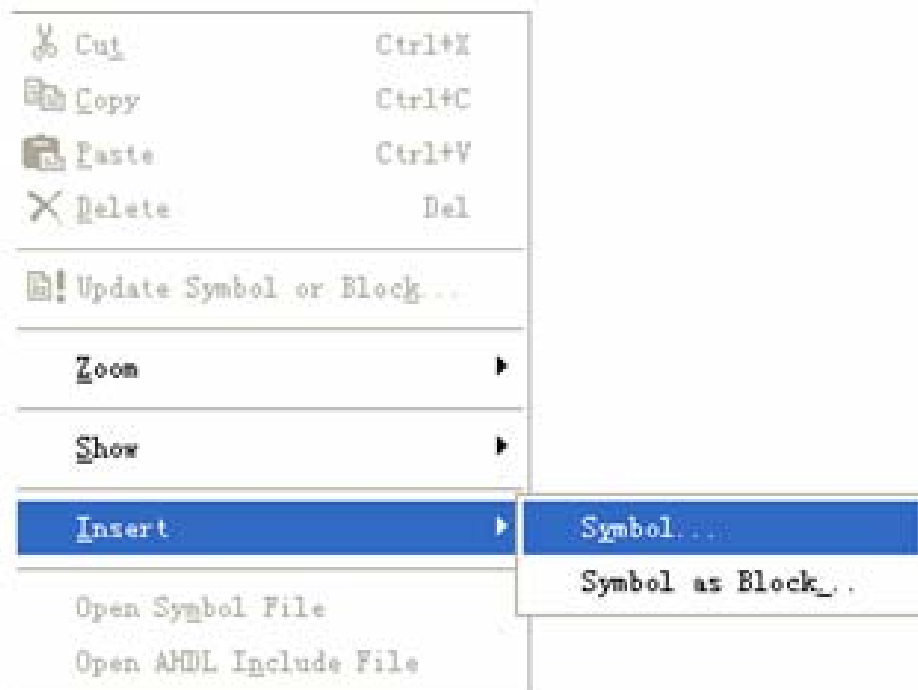


图 4-40 选择打开元件输入窗

# 4.5 原理图输入设计方法

## 4.5.1 层次化设计流程

### 2. 建立原理图文件工程和仿真

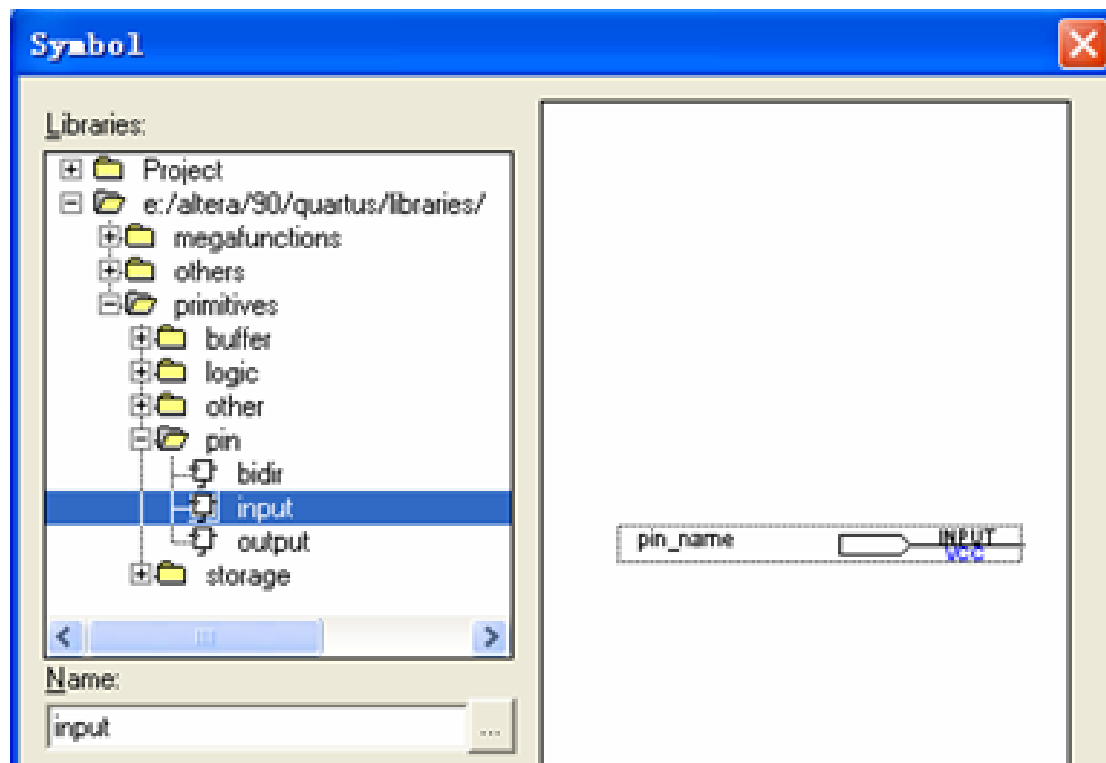


图 4-41 在元件输入对话框输入引脚



# 4.5 原理图输入设计方法

## 4.5.1 层次化设计流程

### 2. 建立原理图文件工程和仿真

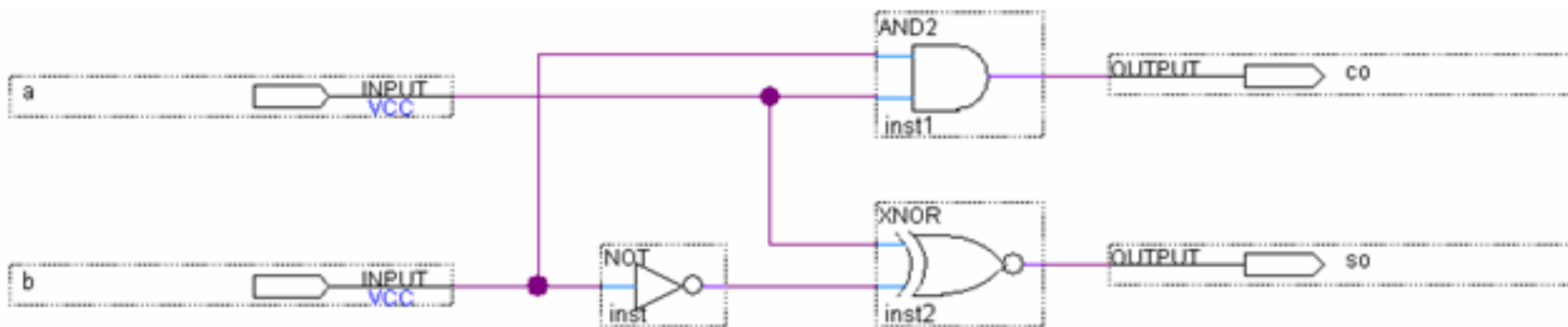


图 4-42 半加器原理图

# 4.5 原理图输入设计方法

## 4.5.1 层次化设计流程

### 2. 建立原理图文件工程和仿真

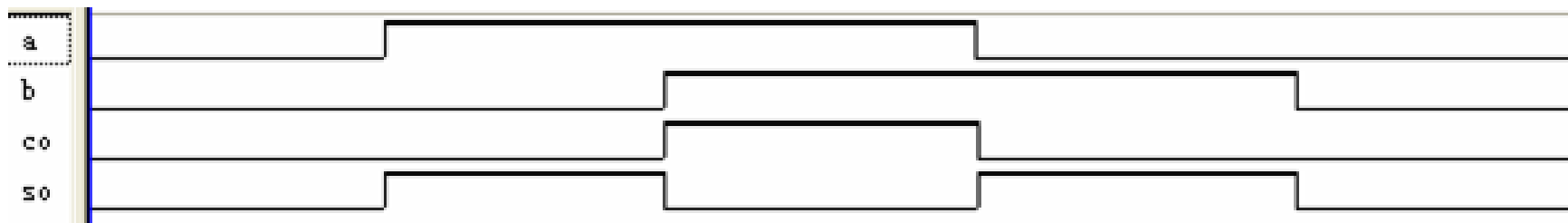


图 4-43 半加器仿真波形

# 4.5 原理图输入设计方法

## 4.5.1 层次化设计流程

### 3. 将设计项目设置成可调用的元件

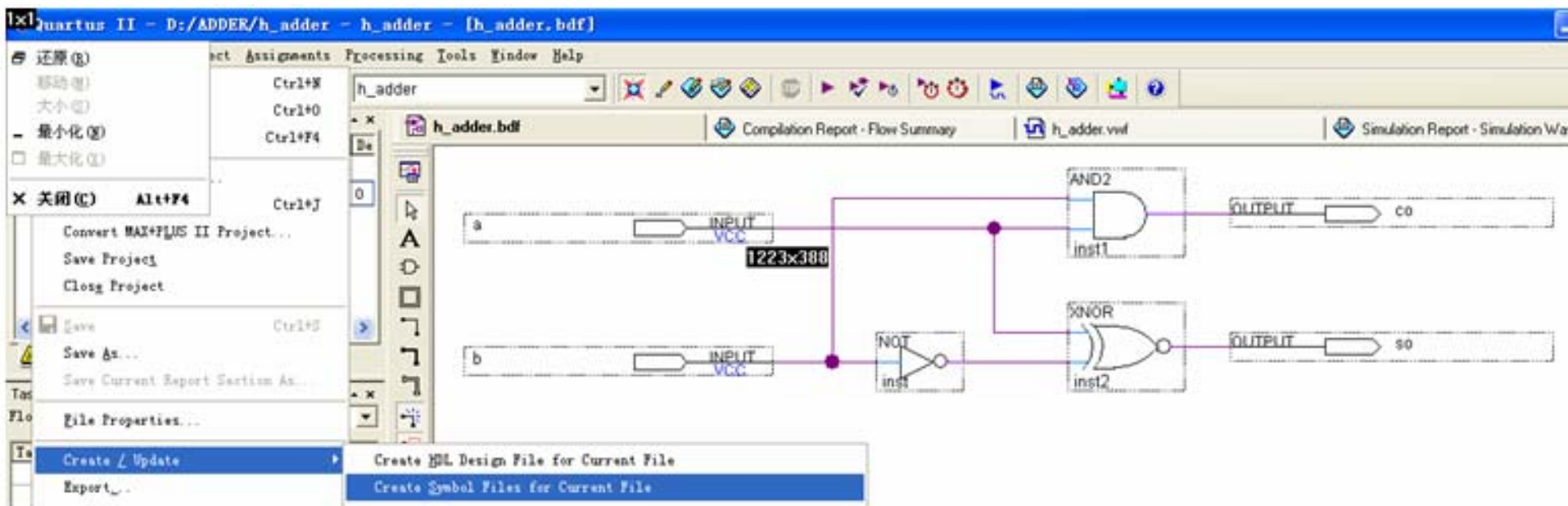


图 4-44 将半加器封装成一个元件，以便在更高层设计中调用

# 4.5 原理图输入设计方法

## 4.5.1 层次化设计流程

### 4. 设计全加器顶层文件



图 4-45 全加器 f\_adder.bdf 工程设置窗口

# 4.5 原理图输入设计方法

## 4.5.1 层次化设计流程

### 4. 设计全加器顶层文件

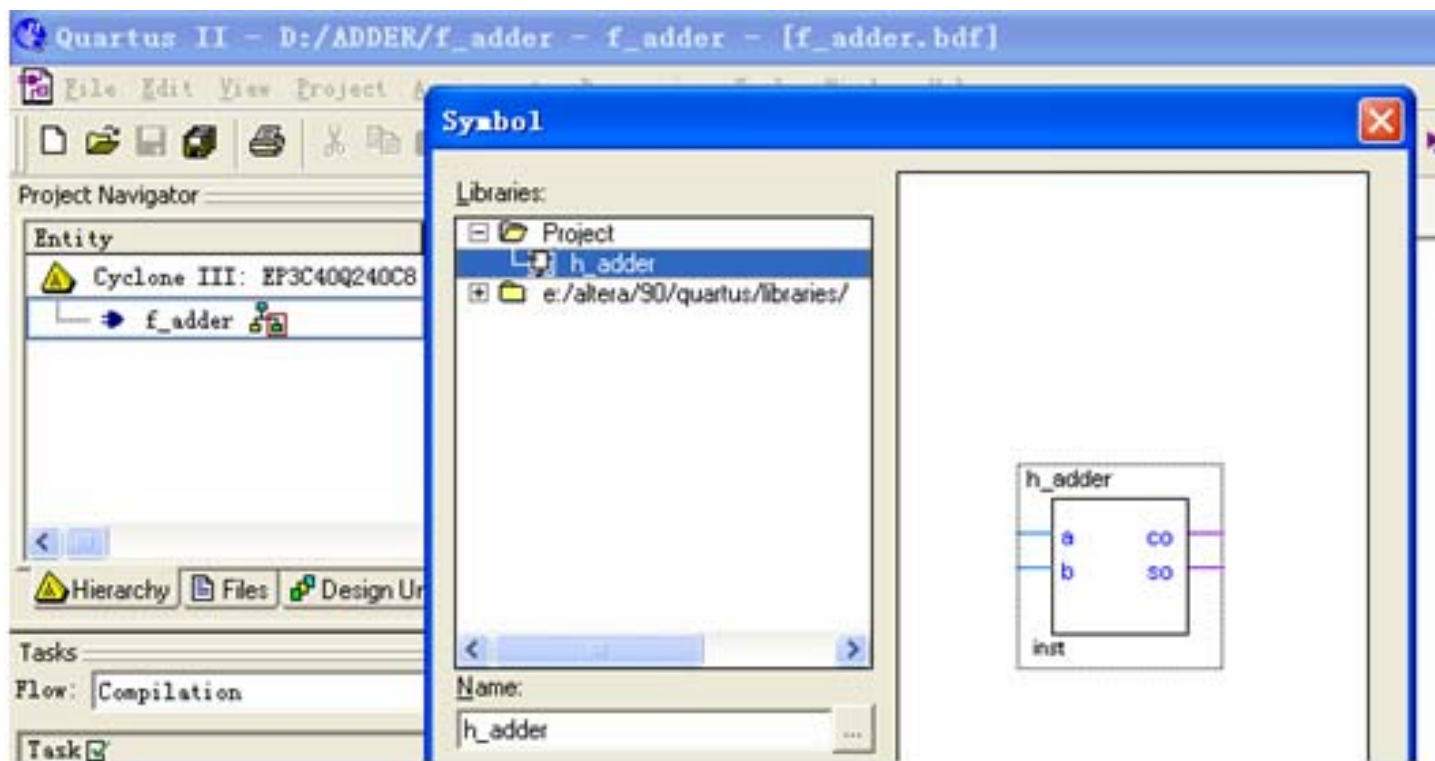


图 4-46 在 f\_adder 工程下的原理图编辑窗中加入半加器元件

# 4.5 原理图输入设计方法

## 4.5.1 层次化设计流程

### 4. 设计全加器顶层文件

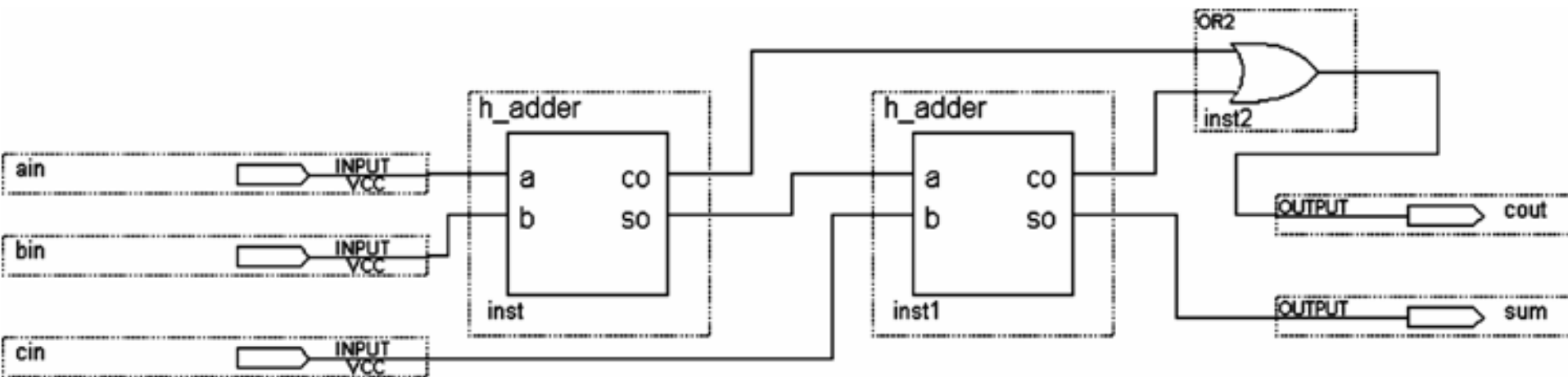


图 4-47 连接好的全加器原理图 f\_adder.bdf

# 4.5 原理图输入设计方法

## 4.5.1 层次化设计流程

### 5. 将设计项目进行时序仿真

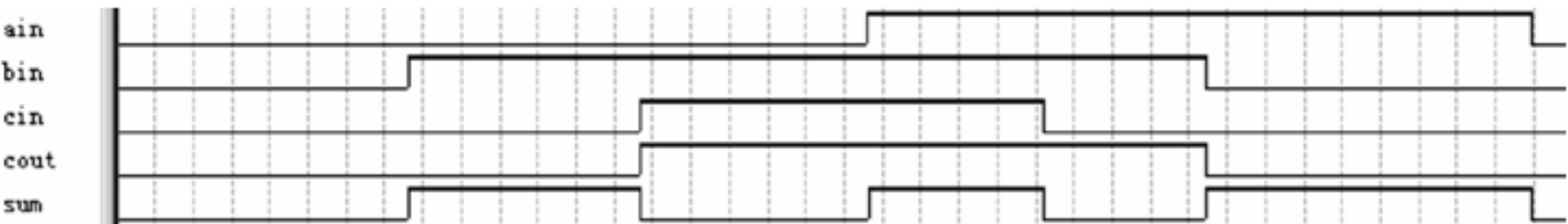


图 4-48 全加器工程 f\_adder 的仿真波形

# 4.5 原理图输入设计方法

## 4.5.2 应用宏模块的多层次原理图设计

### 1. 计数器设计

(1) 设计电路原理图

(2) 建立工程



# 4.5 原理图输入设计方法

## (2) 建立工程

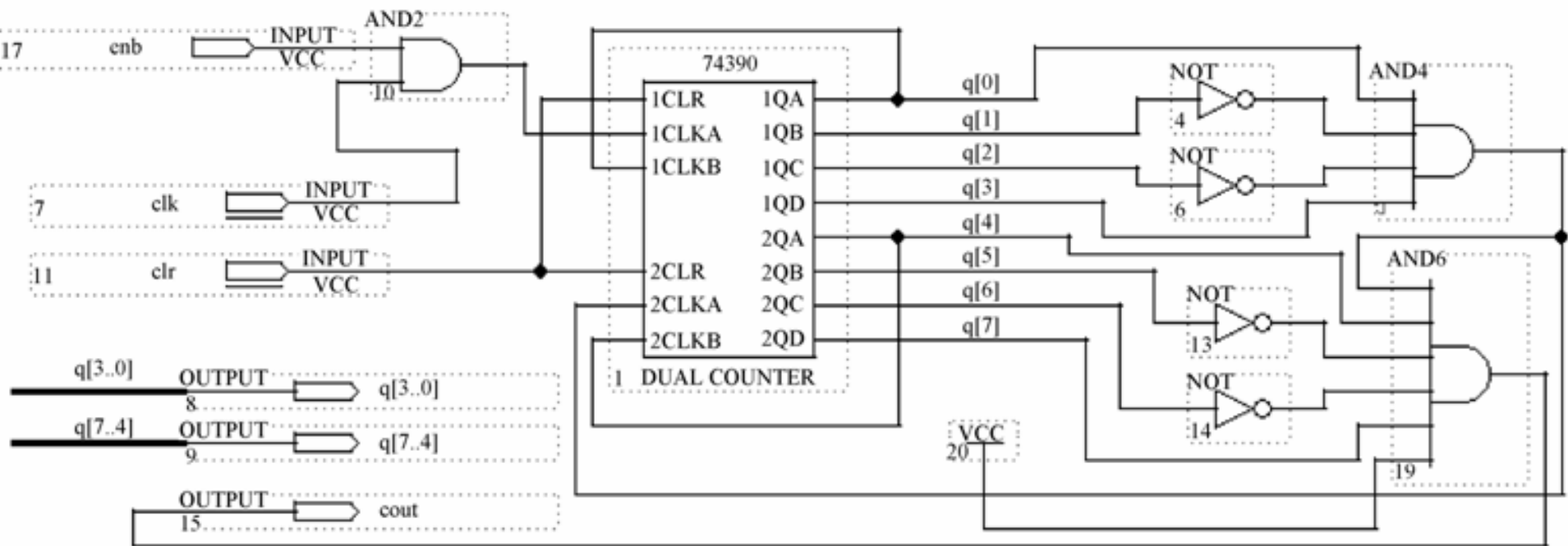


图 4-49 含有时钟使能的 2 位十进制计数器

# 4.5 原理图输入设计方法

## (3) 系统仿真

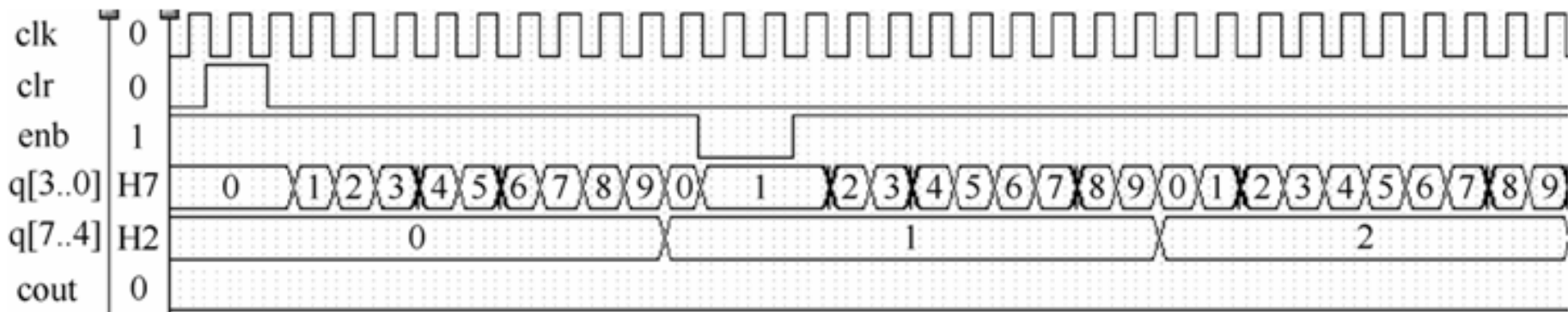


图 4-50 两位十进制计数器工作波形

## (4) 生成元件符号

# 4.5 原理图输入设计方法

## 2. 频率计主结构电路设计

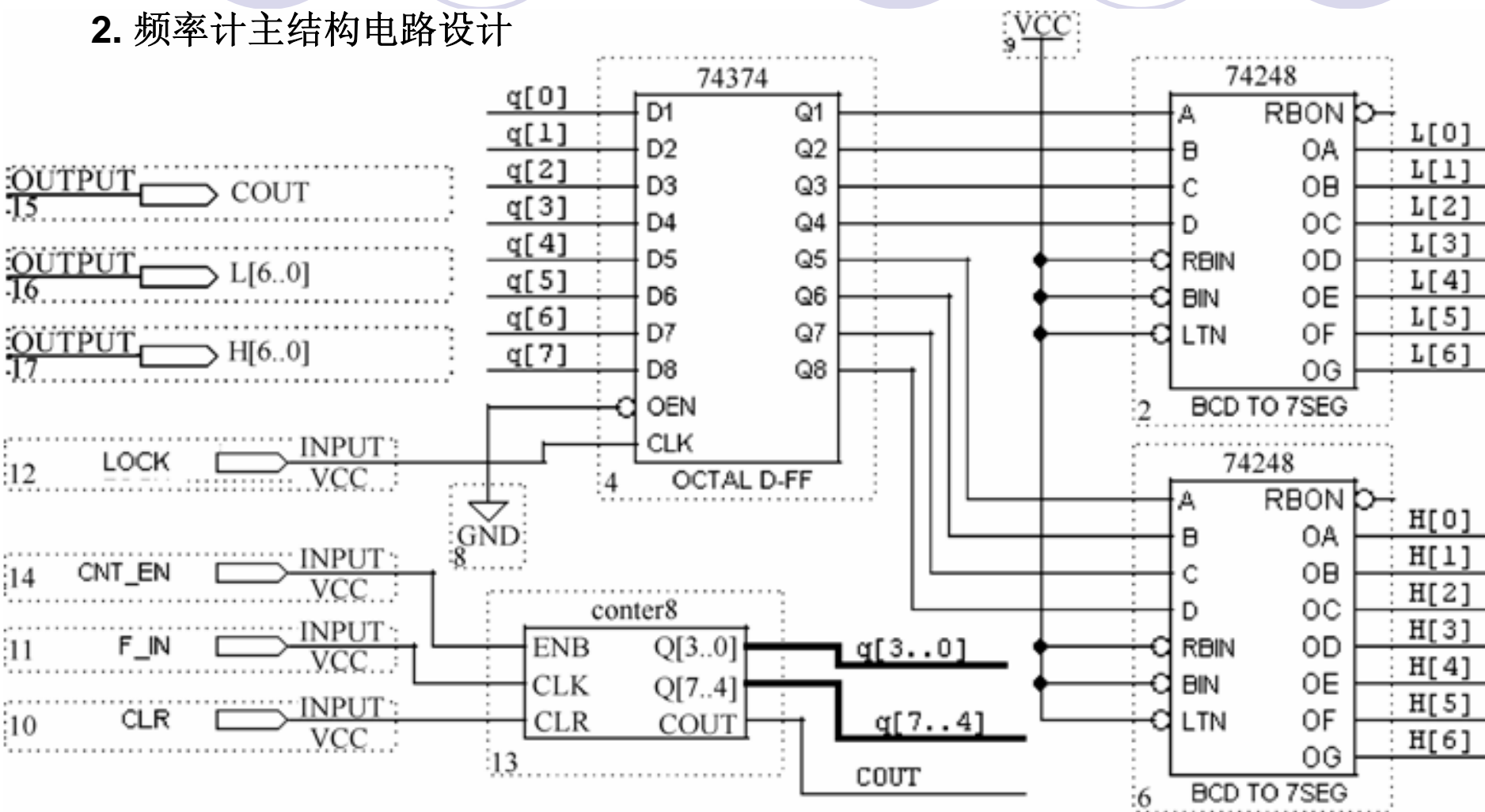


图 4-51 2 位十进制频率计顶层设计原理图文件

# 4.5 原理图输入设计方法

## 4.5.2 应用宏模块的多层次原理图设计

### 2. 频率计主结构电路设计

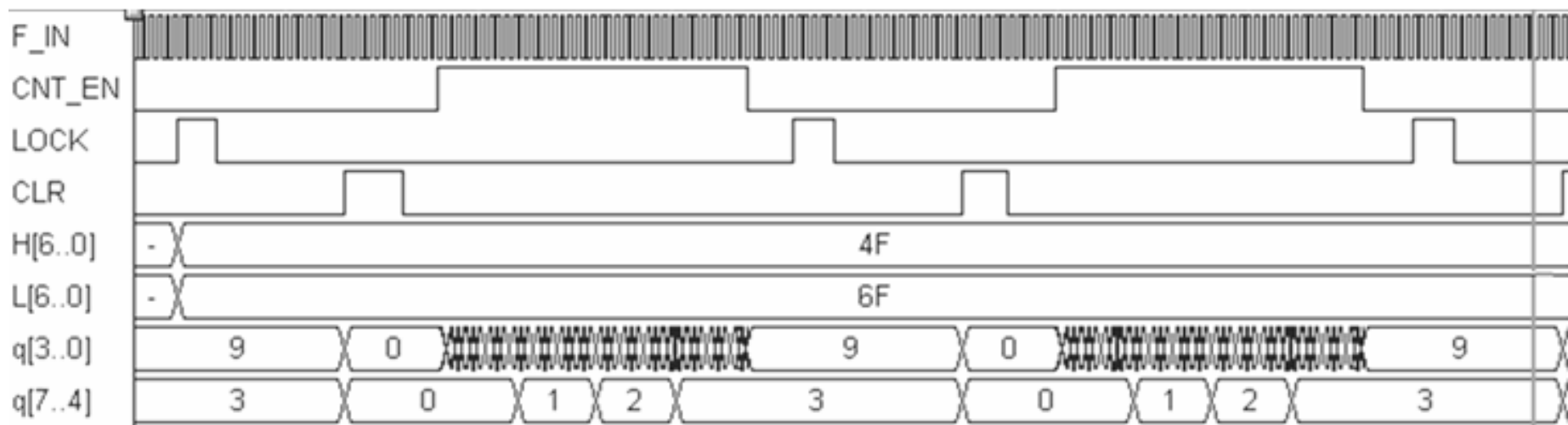


图 4-52 两位十进制频率计测频仿真波形

# 4.5 原理图输入设计方法

## 4.5.2 应用宏模块的多层次原理图设计

### 3. 时序控制电路设计

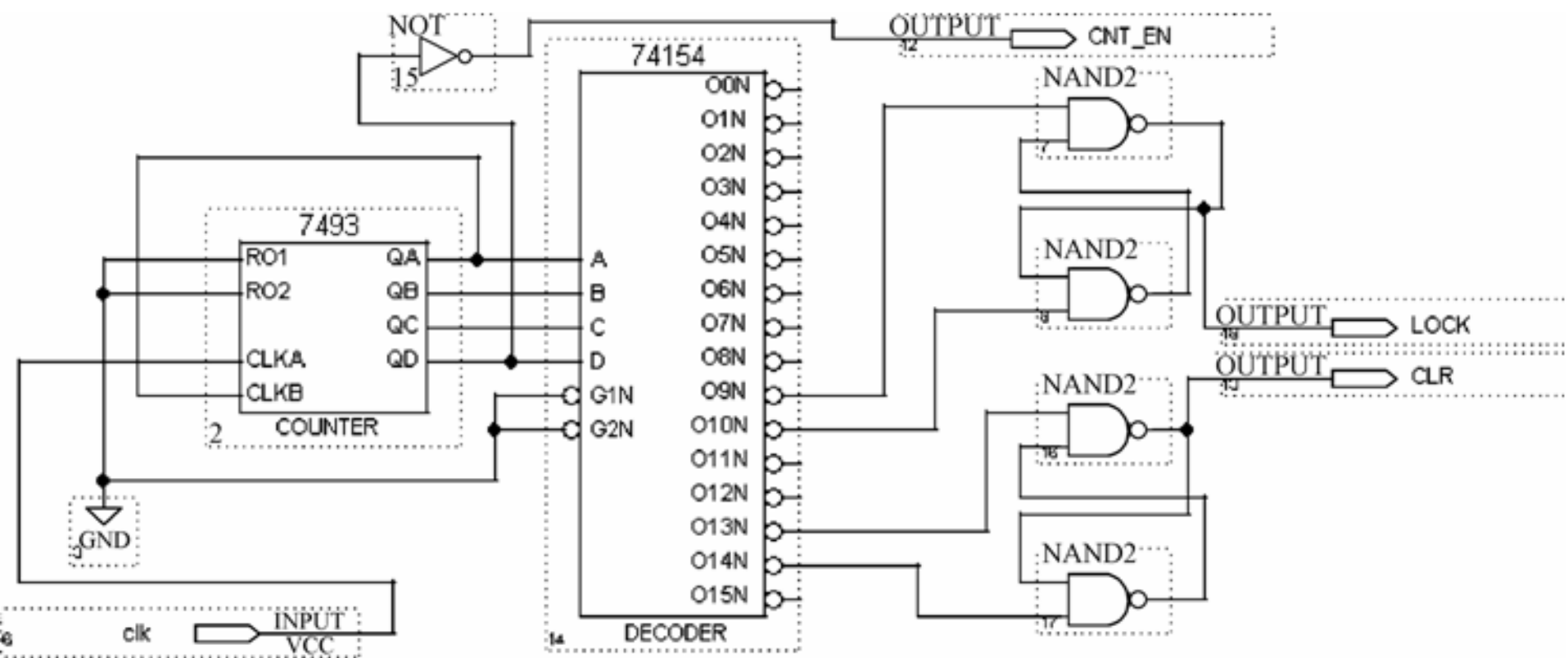


图 4-53 测频时序控制电路

# 4.5 原理图输入设计方法

## 4.5.2 应用宏模块的多层次原理图设计

### 3. 时序控制电路设计

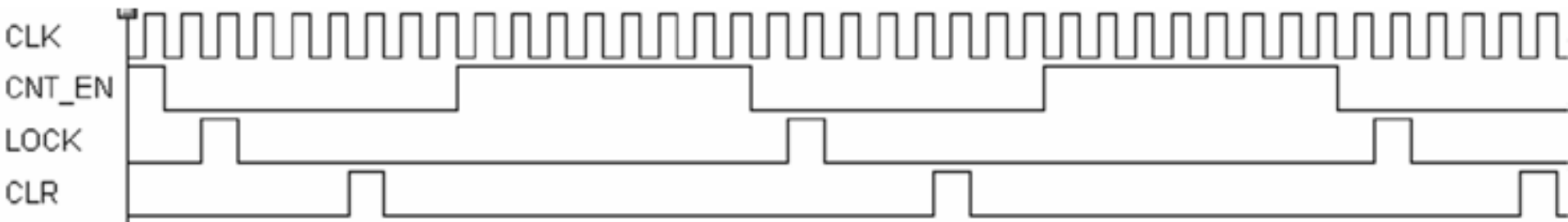


图 4-54 测频时序控制电路工作波形

# 4.5 原理图输入设计方法

## 4. 顶层电路设计

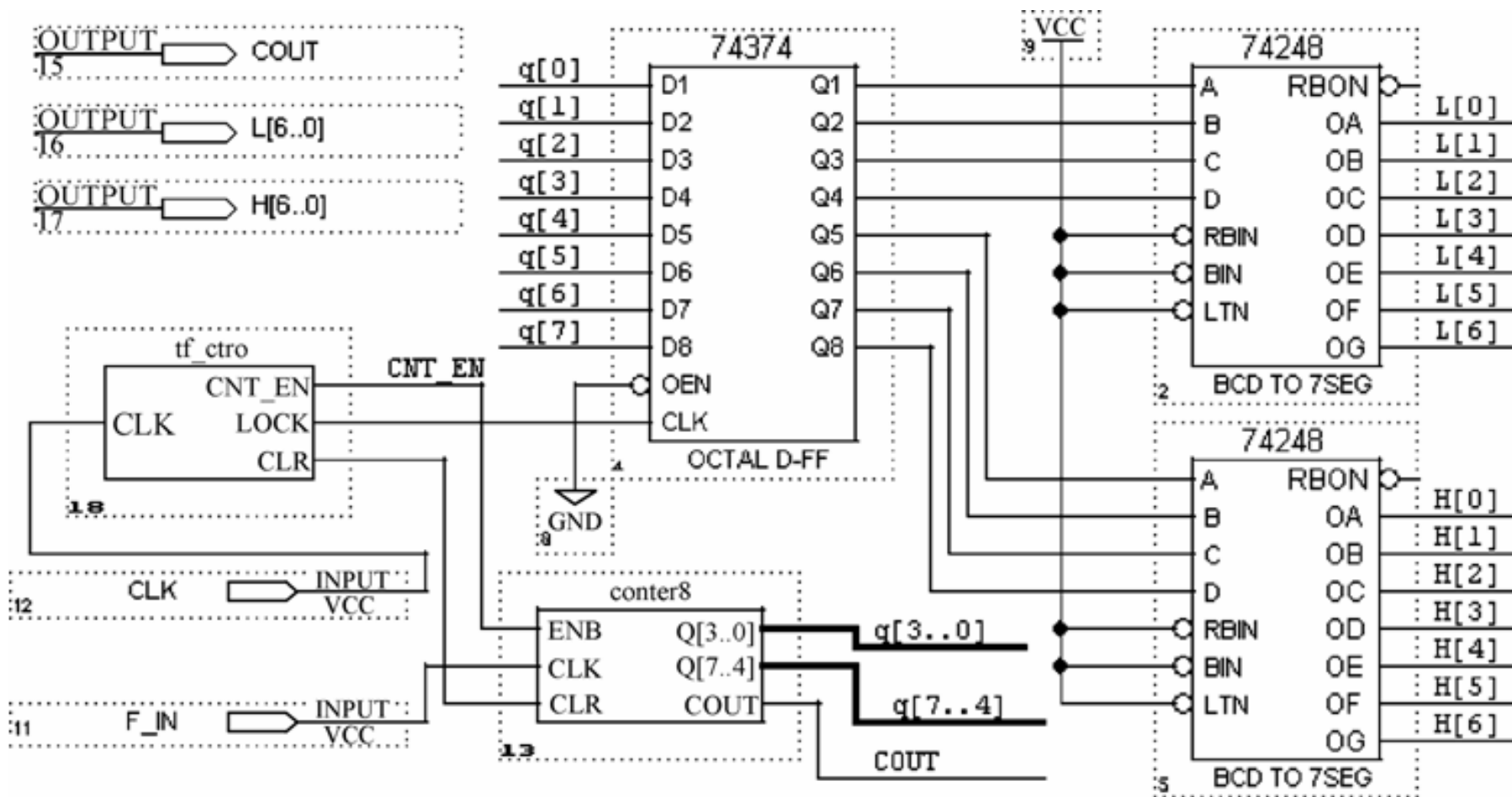


图 4-55 频率计顶层电路原理图

# 4.5 原理图输入设计方法

## 4.5.2 应用宏模块的多层次原理图设计

### 4. 顶层电路设计

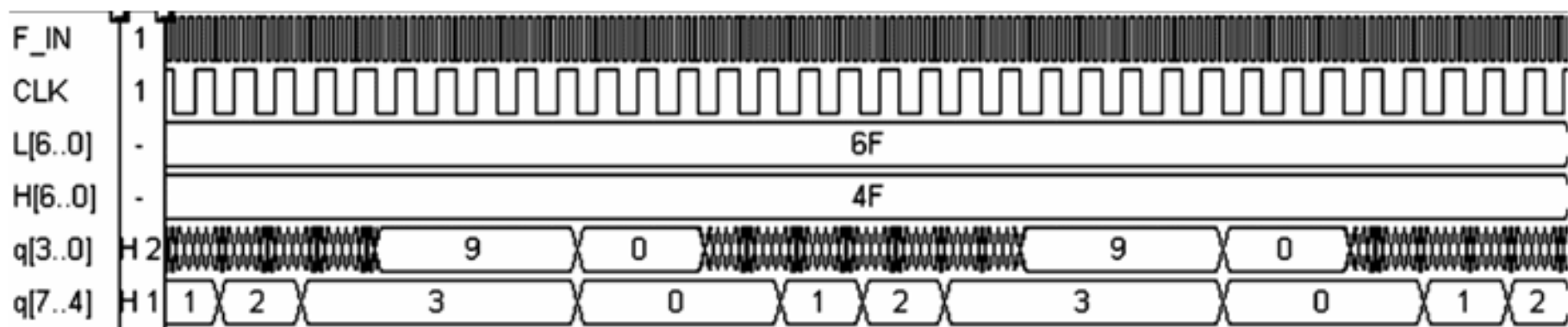


图 4-56 频率计工作时序波形



# 4.5 原理图输入设计方法

## 4.5.3 74系列宏模块逻辑功能真值表查询

```
FUNCTION 74138 (g1, g2an, g2bn, c, b, a)  
  RETURNS (y0n, y1n, y2n, y3n, y4n, y5n, y6n, y7n);
```

| Inputs |     |        |   |   |  | Outputs |     |     |     |     |     |     |     |
|--------|-----|--------|---|---|--|---------|-----|-----|-----|-----|-----|-----|-----|
| Enable |     | Select |   |   |  | Y0N     | Y1N | Y2N | Y3N | Y4N | Y5N | Y6N | Y7N |
| G1     | G2* | C      | B | A |  |         |     |     |     |     |     |     |     |
| X      | H   | X      | X | X |  | H       | H   | H   | H   | H   | H   | H   | H   |
| L      | X   | X      | X | X |  | H       | H   | H   | H   | H   | H   | H   | H   |
| H      | L   | L      | L | L |  | L       | H   | H   | H   | H   | H   | H   | H   |
| H      | L   | L      | L | H |  | H       | L   | H   | H   | H   | H   | H   | H   |
| H      | L   | L      | H | L |  | H       | H   | L   | H   | H   | H   | H   | H   |
| H      | L   | L      | H | H |  | H       | H   | H   | L   | H   | H   | H   | H   |
| H      | L   | H      | L | L |  | H       | H   | H   | H   | L   | H   | H   | H   |
| H      | L   | H      | L | H |  | H       | H   | H   | H   | H   | L   | H   | H   |
| H      | L   | H      | H | L |  | H       | H   | H   | H   | H   | H   | L   | H   |
| H      | L   | H      | H | H |  | H       | H   | H   | H   | H   | H   | H   | L   |

\* G2 = G2AN + G2BN

图 4-57 74138 真值表

## 4.6 keep属性应用

```
SIGNAL keep_wire : STD_LOGIC ;--对标准位矢量也以此相同定义方式
attribute keep : boolean;
attribute keep of keep_wire : signal is true;
```

### 【例 4-2】

```
ARCHITECTURE fd1 OF f_adder IS
. . .
SIGNAL d, f : STD_LOGIC;
SIGNAL e : STD_LOGIC;
    attribute keep : boolean;
    attribute keep of e : signal is true;
BEGIN
```

## 4.6 keep属性应用

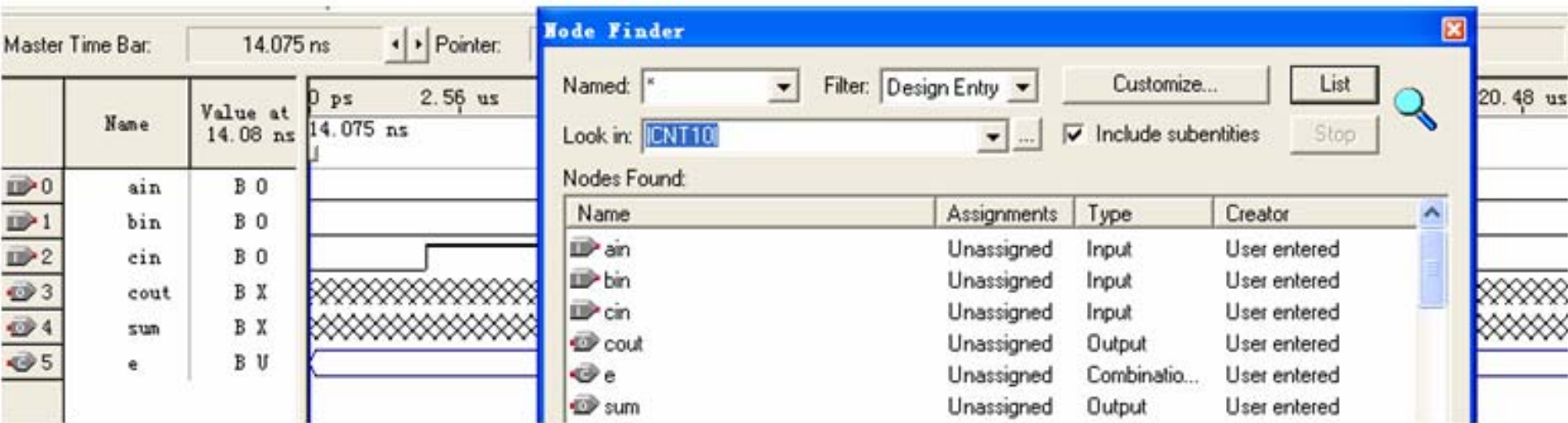


图 4-58 向仿真激励信号波形编辑窗调入信号 e

## 4.7 SignalProbe使用方法

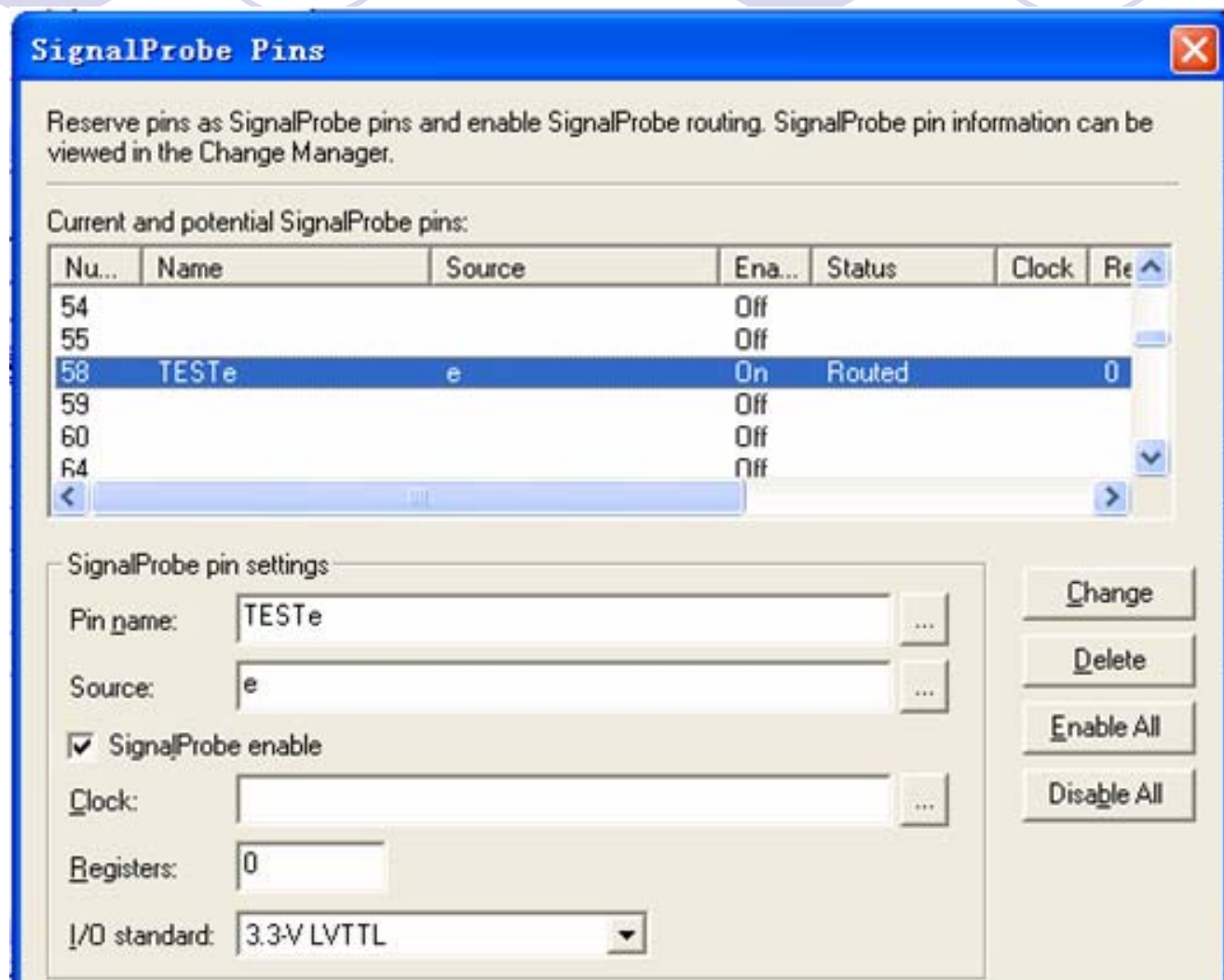
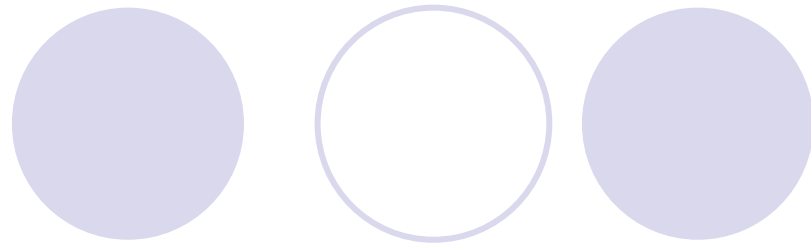


图 4-59 在 SignalProbe 对话框设置探测信号 e

## 4.8 Settings设置



- (1) 修改工程设置。
- (2) 指定HDL设置。
- (3) 指定时序设置。
- (4) 指定编译器设置。
- (5) 指定仿真器设置。
- (6) 指定软件构建设置。
- (7) 指定HardCopy时序设置。

## 4.9 适配器Fitter设置

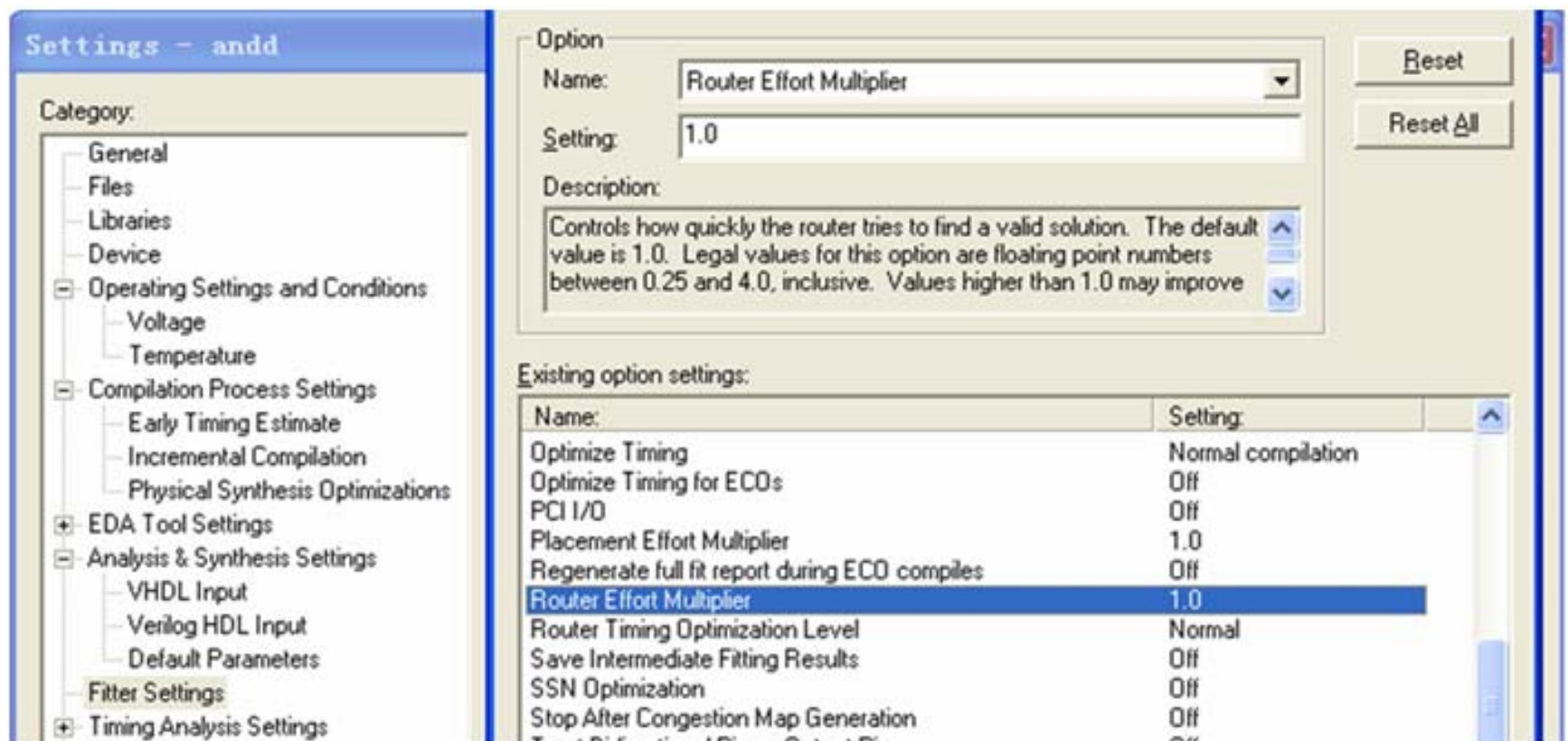


图 4-60 布线倍增器优化程度指数选择

## 4.10 HDL版本设置及**Analysis & Synthesis**功能

**Analysis & Synthesis**构建单个工程数据库，将所有设计文件集成在设计实体或工程层次结构中。

编译报告窗口和**Report**窗口的信息区域显示出**Analysis & Synthesis**生成的任何信息。**Status**窗口记录工程编译期间在**Analysis & Synthesis**中处理所花的时间。

# 4.11 Chip Planner应用

## 4.11.1 Chip Planner应用实例

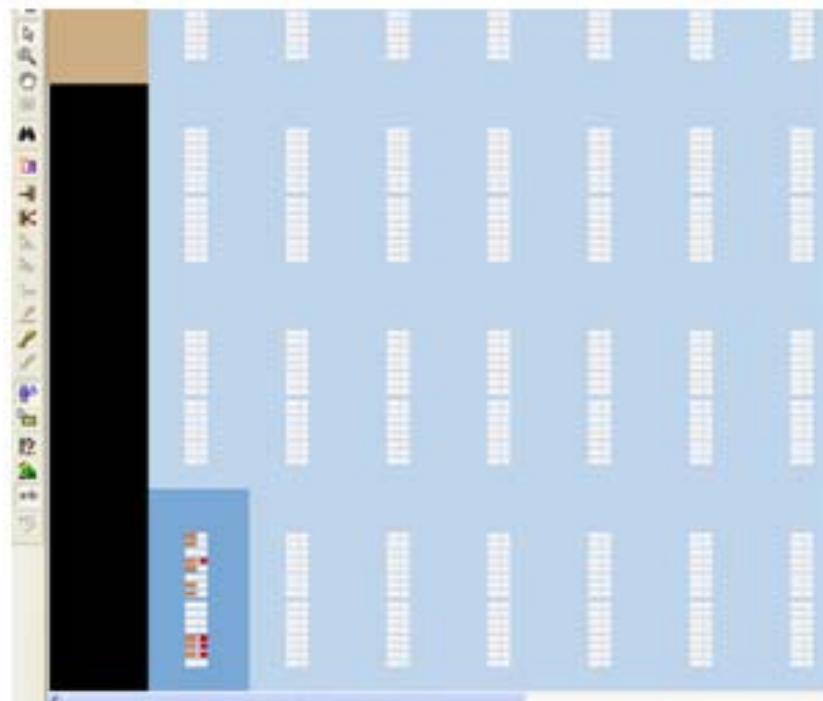


图 4-61 左下侧是已占用的 LAB



# 4.11 Chip Planner应用

## 4.11.1 Chip Planner应用实例

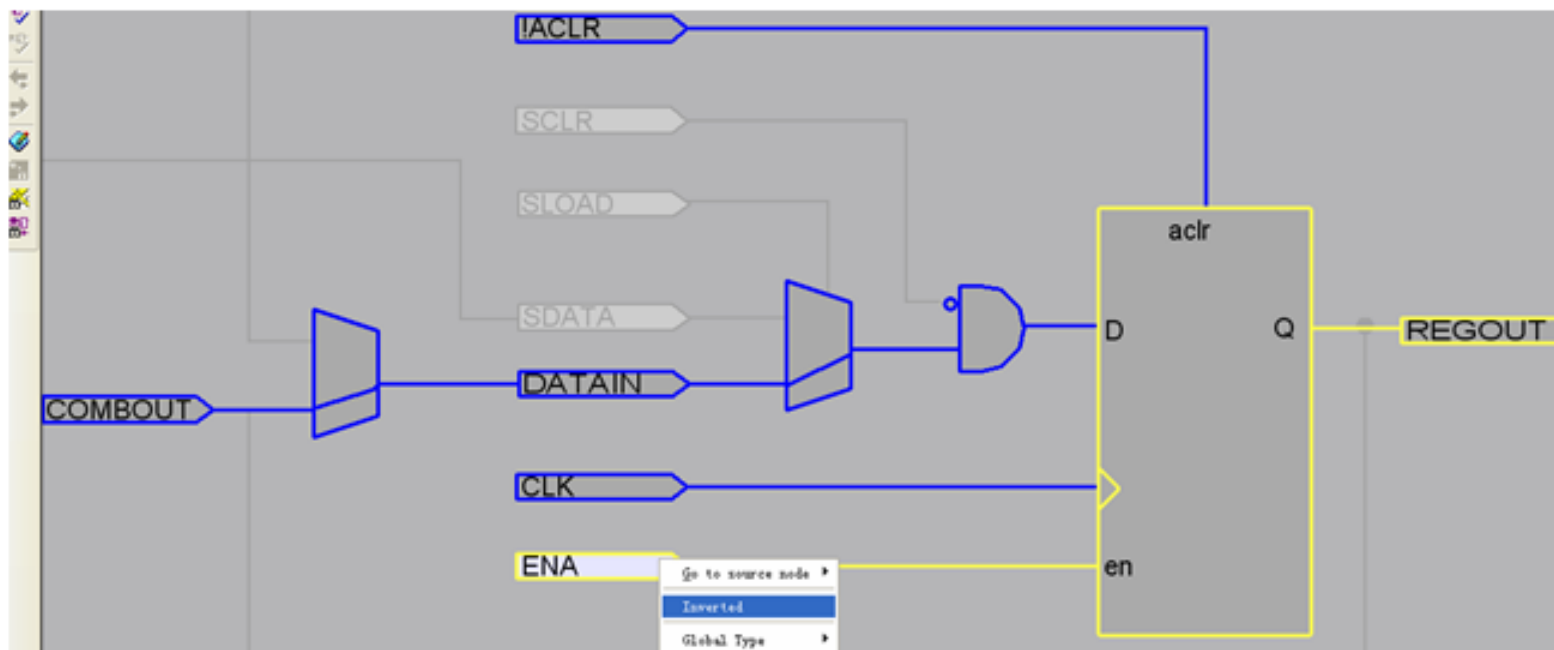


图 4-62 Resource Property Editor 的门级原理图编辑窗口

# 4.11 Chip Planner应用

## 4.11.2 Chip Planner功能说明

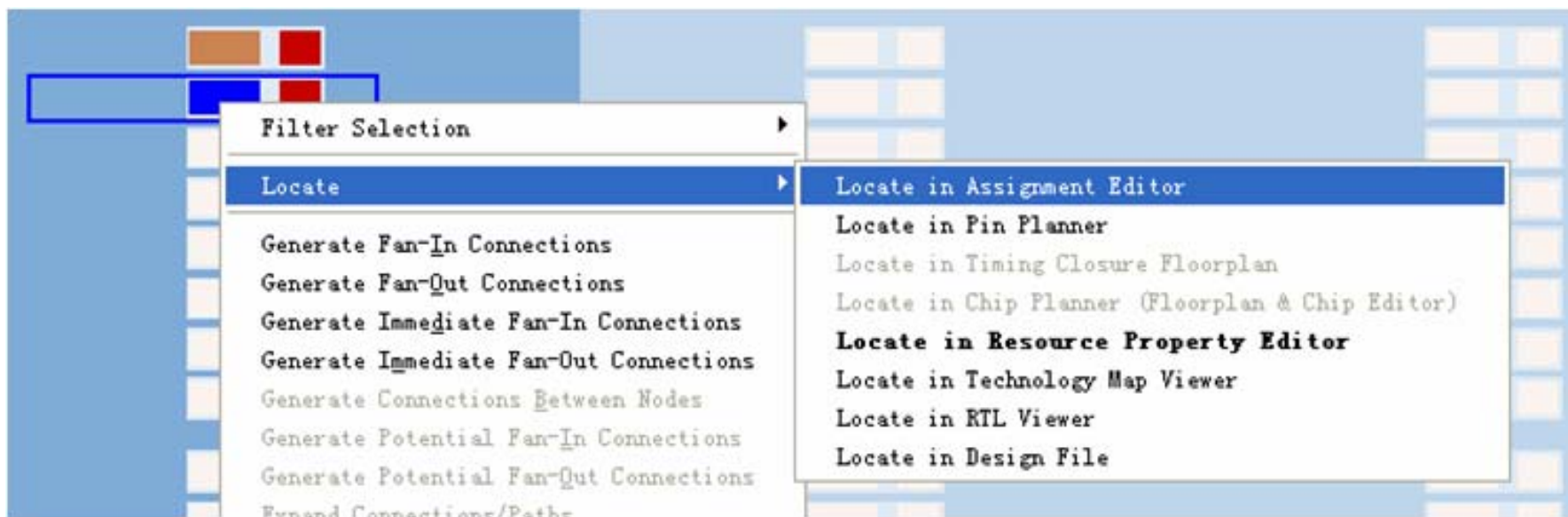


图 4-63 Locate 多项选择

# 4.11 Chip Planner应用

## 4.11.2 Chip Planner功能说明

The screenshot displays the Chip Planner interface with a logic circuit diagram. The circuit includes a block 'A' with inputs B and C, and output COMBOUT. This output is connected to a multiplexer that selects between DATAIN and another input. The selected signal is connected to the D input of a register 'acr'. The register also has an enable input 'en' and an asynchronous clear input 'ACL'. The output of the register is REGOUT. The circuit is connected to various input ports: DATAIN, DATA, SCL, SDA, SW, RST, and RSTO.

| Input Port name | Signal name         | Latch info | Inverted |
|-----------------|---------------------|------------|----------|
| Register Node   |                     |            |          |
| -SLOAD          | <Disconnected>      | N/A        | False    |
| -DATAIN         | CNT10 Q1[0]~3       | N/A        | False    |
| -SDATA          | <Disconnected>      | N/A        | False    |
| -ACL            | CNT10 RST~inputctrl | N/A        | True     |
| -SCLR           | <Disconnected>      | N/A        | False    |

| Output Port name   | Signal name    | Latch info |
|--------------------|----------------|------------|
| Register Node      |                |            |
| REGOUT             | CNT10 Q1[0]    | N/A        |
| Combinational Node |                |            |
| COMBOUT            | CNT10 Q1[0]~3  | N/A        |
| COU                | <Disconnected> | N/A        |

| Properties/Modes | Values |
|------------------|--------|
| Sum LUT Mask     | OPOF   |
| Carry LUT Mask   | N/A    |
| Operation Mode   | normal |
| Latch Type       | none   |

| Properties     | Values |
|----------------|--------|
| Sum Equation   | IC     |
| Carry Equation | N/A    |

| Node:  | CNT10 Q1[0] | Go To |
|--------|-------------|-------|
| Q      |             |       |
| ACL    | 706/678 ps  |       |
| CLK    | 667/600 ps  |       |
| DATAIN | 99/115 ps   |       |

图 4-64 打开属性和端口连接窗口

# 4.11 Chip Planner应用

## 4.11.3 利用Change Manager检测底层逻辑

- (1) 更改编号。
- (2) 节点名称 (**Node Name**) 。
- (3) 更改类型 (**Change Type**) 。
- (4) 旧值 (**Old Value**) 。
- (5) 目标值 (**Target Value**) 。
- (6) 当前值 (**Current Value**) 。
- (7) 用户添加的有关**ECO**更改的备注。
- (8) 状态 (**Status**) \_\_\_\_\_

- |       |
|-------|
| 1、待定  |
| 2、已应用 |
| 3、无效  |
| 4、未应用 |

# 4.12 Synplify Pro的应用及其与Quartus II接口

## 4.12.1 Synplify Pro设计指南

### 1. 启动Synplify

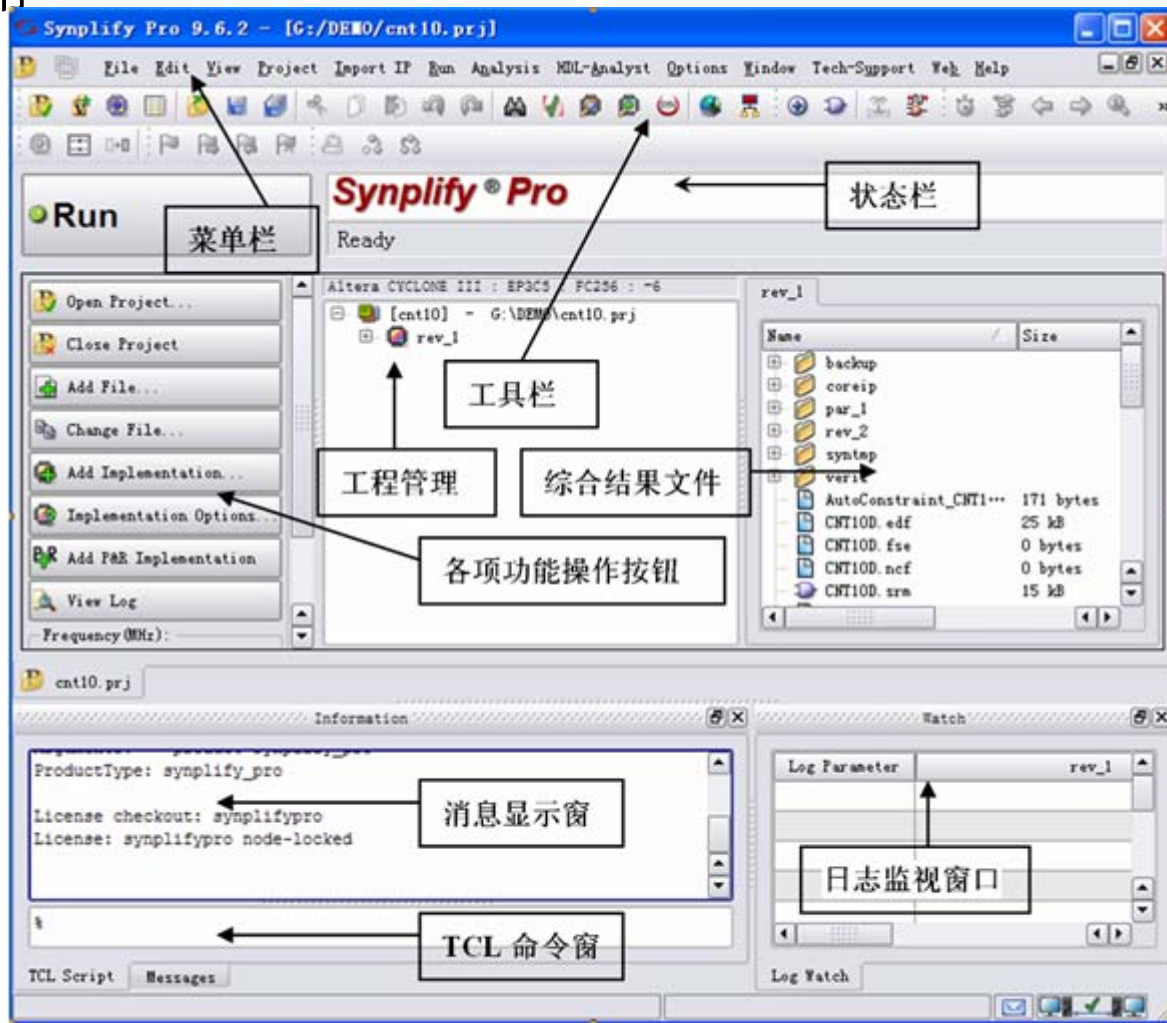


图4-65 Synplify Pro启动后界面

# 4.12 Synplify Pro的应用及其与Quartus II接口

## 4.12.1 Synplify Pro设计指南

### 2. 创建工程

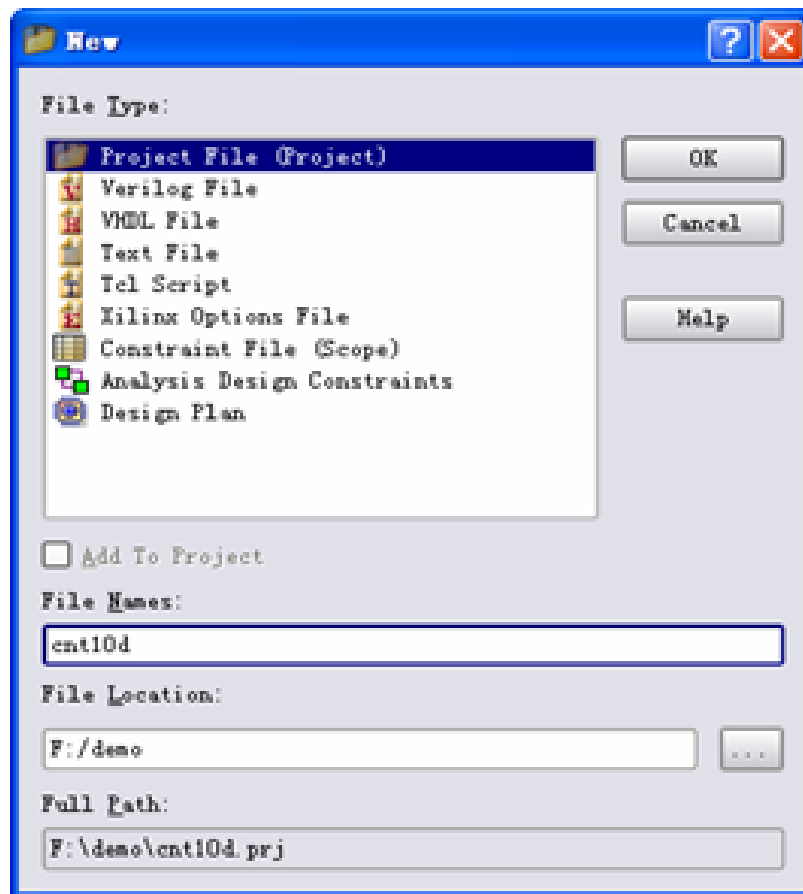


图4-66 创建工程

# 4.12 Synplify Pro的应用及其与Quartus II接口

## 4.12.1 Synplify Pro设计指南

3. 加入源文件
4. 选择顶层文件
5. 设置工程属性

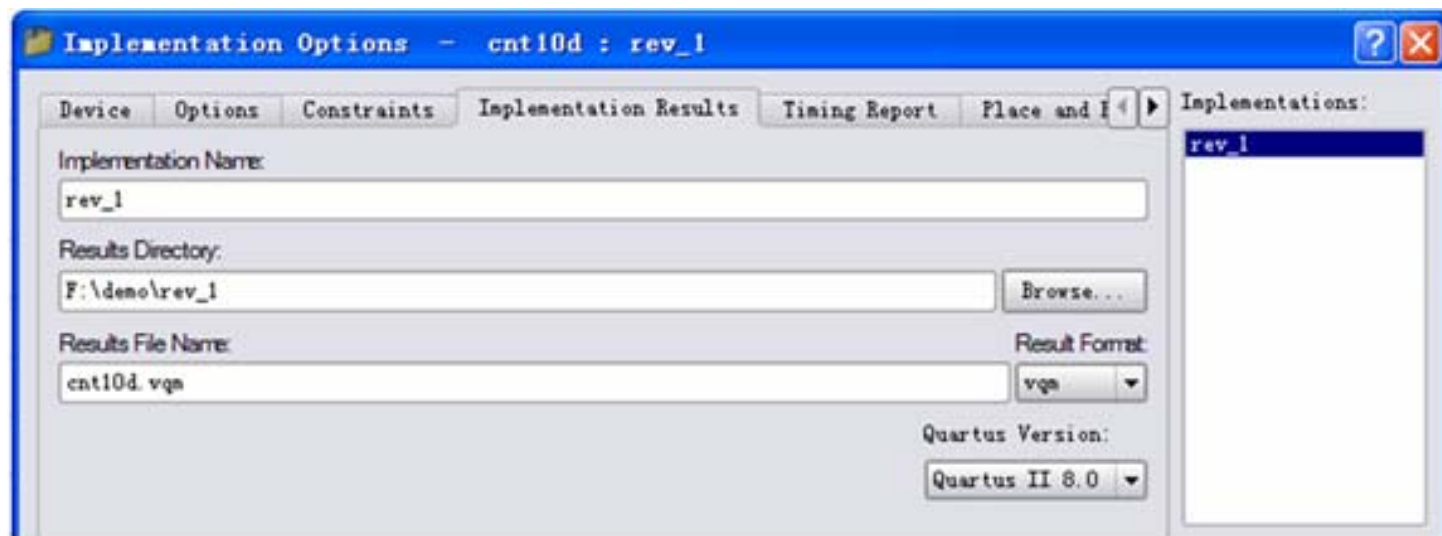


图4-67 综合目标详细信息

# 4.12 Synplify Pro的应用及其与Quartus II接口

## 4.12.1 Synplify Pro设计指南

6. 综合前设置约束
7. 综合
8. 检测结果

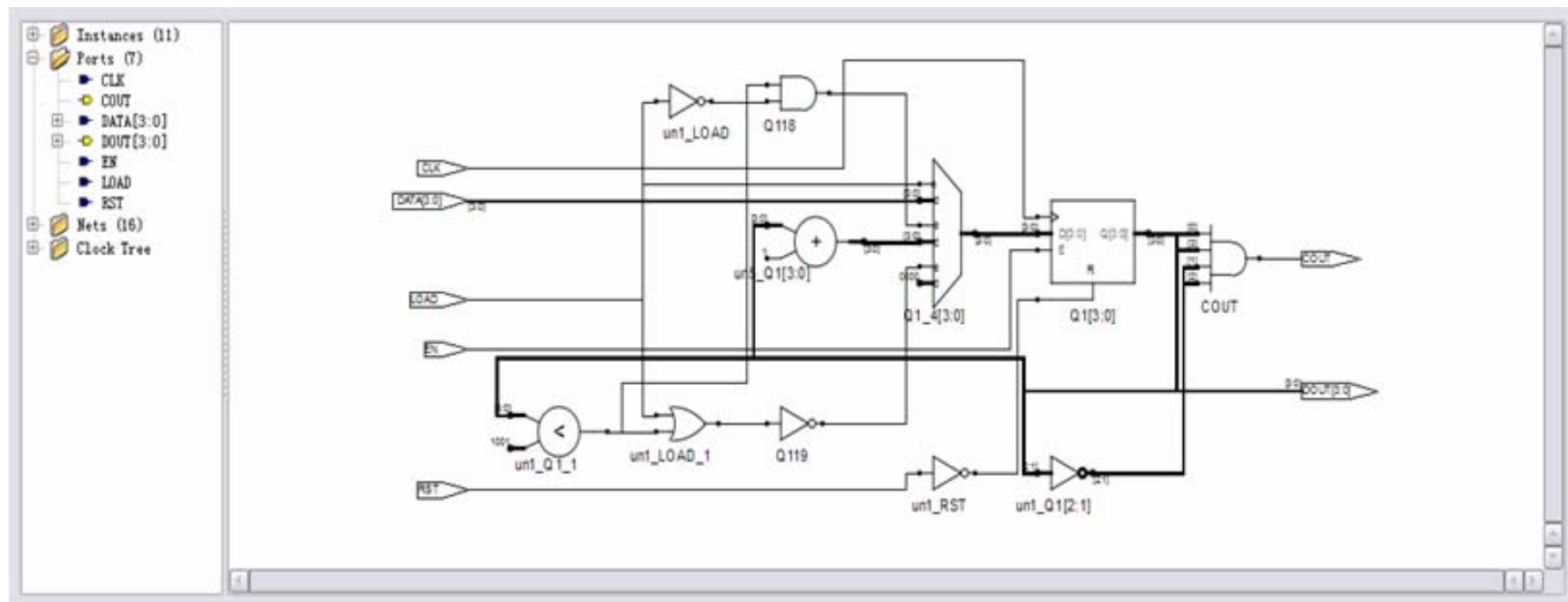


图4-68 Synplify的RTL级电路图（注，此图的细节须通过Synplify了解）



# 4.12 Synplify Pro的应用及其与Quartus II接口

## 4.12.2 Synplify Pro与Quartus II的接口方法

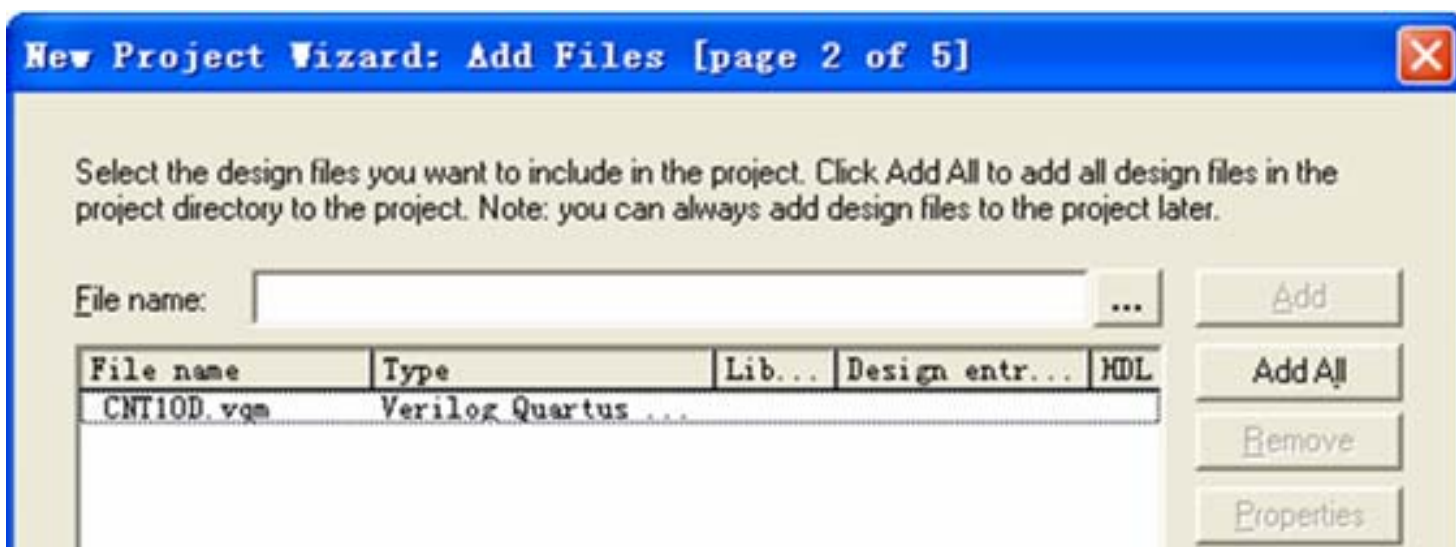


图 4-69 加入 Cnt10d.vqm 文件

# 4.12 Synplify Pro的应用及其与Quartus II接口

## 4.12.2 Synplify Pro与Quartus II的接口方法

### 1. Synplify软件路径设置

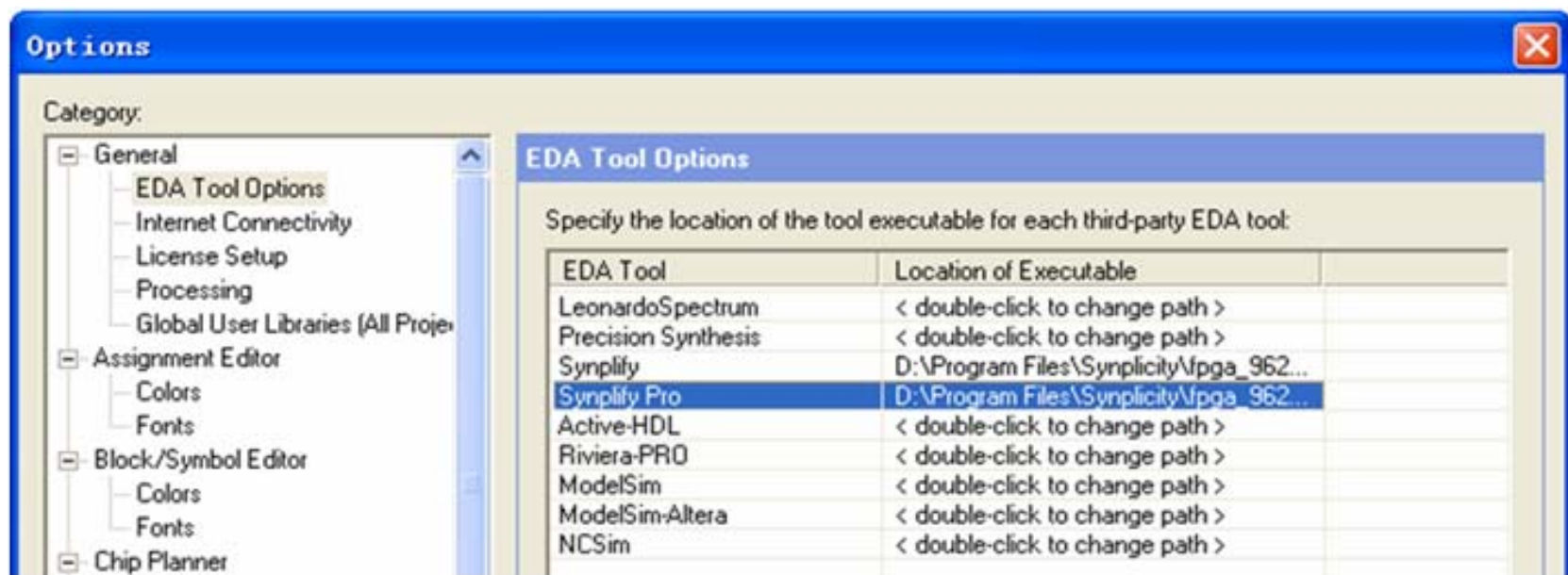


图4-70 Synplify软件路径设置

# 4.12 Synplify Pro的应用及其与Quartus II接口

## 4.12.2 Synplify Pro与Quartus II的接口方法

### 2. 设置Synplify Pro综合器

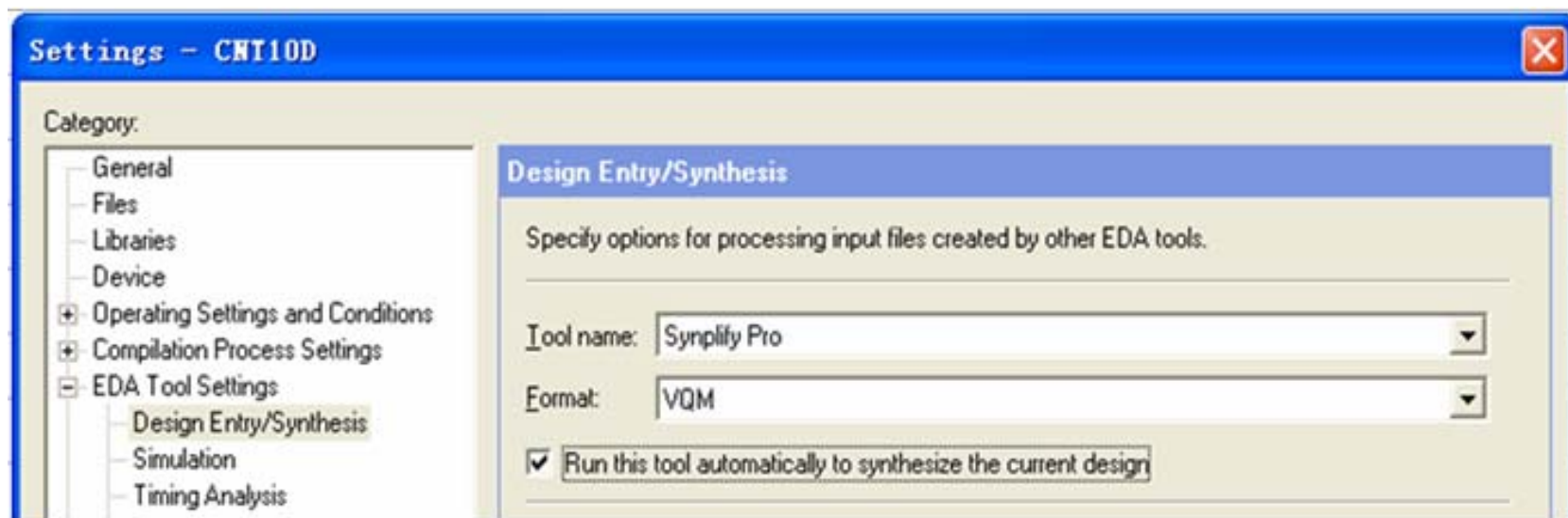


图4-71 设置综合器



# 习 题

**4-1** 归纳利用**Quartus II**进行**VHDL**文本输入设计的流程：从文件输入一直到**SignalTap II**测试。

**4-2** 由图**4-35**和图**4-36**，详细说明工程**CNT10**的硬件工作情况。

**4-3** 如何为设计中的**SignalTap II**加入独立采样时钟？试给出完整的程序和对它的实测结果。

**4-4** 参考**Quartus II**的**Help**，详细说明**Assignments**菜单中**Settings**对话框的功能。

(1) 说明其中的**Timing Requirements & Options**的功能、使用方法和检测途径。

(2) 说明其中的**Compilation Process**的功能和使用方法。

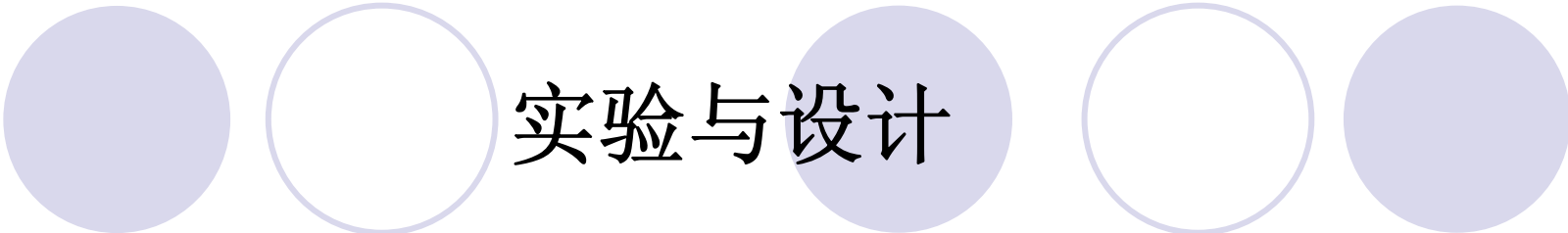
(3) 说明**Analysis & Synthesis Setting**的功能和使用方法，以及其中的**Synthesis Netlist Optimization**的功能和使用方法。

(4) 说明**Fitter Settings**中的**Design Assistant**和**Simulator**功能，举例说明它们的使用方法。

**4-5** 概述**Assignments**菜单中**Assignment Editor**的功能，举例说明。

# 习 题

- 4-6** 用**74148**和与非门实现**8421BCD**优先编码器，用三片**74139**组成一个**5-24**译码器。
- 4-7** 用**74283**加法器和逻辑门设计实现一位**8421BCD**码加法器电路，输入输出均是**BCD**码，**CI**为低位的进位信号，**CO**为高位的进位信号，输入为两个**1**位十进制数**A**，输出用**S**表示。
- 4-8** 用原理图输入方式设计一个**7**人表决电路，参加表决者**7**人，同意为**1**，不同意为**0**，同意者过半则表决通过，绿指示灯亮；表决不通过则红指示灯亮。
- 4-9** 基于原理图输入方式，用**D**触发器构成按循环码(**000**->**001**->**011**->**111**->**101**->**100**->**000**)规律工作的六进制同步计数器。
- 4-10** 基于原理图输入方式，应用**4**位全加器和**74374**构成**4**位二进制加法计数器。如果使用**74299**、**74373**、**D**触发器和非门来完成上述功能，应该有怎样的电路？
- 4-11** 用一片**74163**和两片**74138**构成一个具有**12**路脉冲输出的数据分配器。要求在原理图上标明第**1**路到第**12**路输出的位置。若改用一片**74195**代替以上的**74163**，试完成同样的设计。
- 4-12** 用同步时序电路对串行二进制输入进行奇偶校验，每检测**5**位输入，输出一个结果。当**5**位输入中**1**的数目为奇数时，在最后一位的时刻输出**1**。
- 4-13** 用**7490**设计模为**872**的计数器，且输出的个位、十位、百位都应符合**8421**码权重。



# 实验与设计

## 4-1 设计含异步清零和同步加载与时钟使能的计数器

(1) 实验目的:

(2) 实验原理:

(3) 实验内容1:

(4) 实验内容2:

(5) 实验内容3:

(6) 实验内容4:

(7) 实验内容5:

# 实验与设计

## (8) 实验内容6:

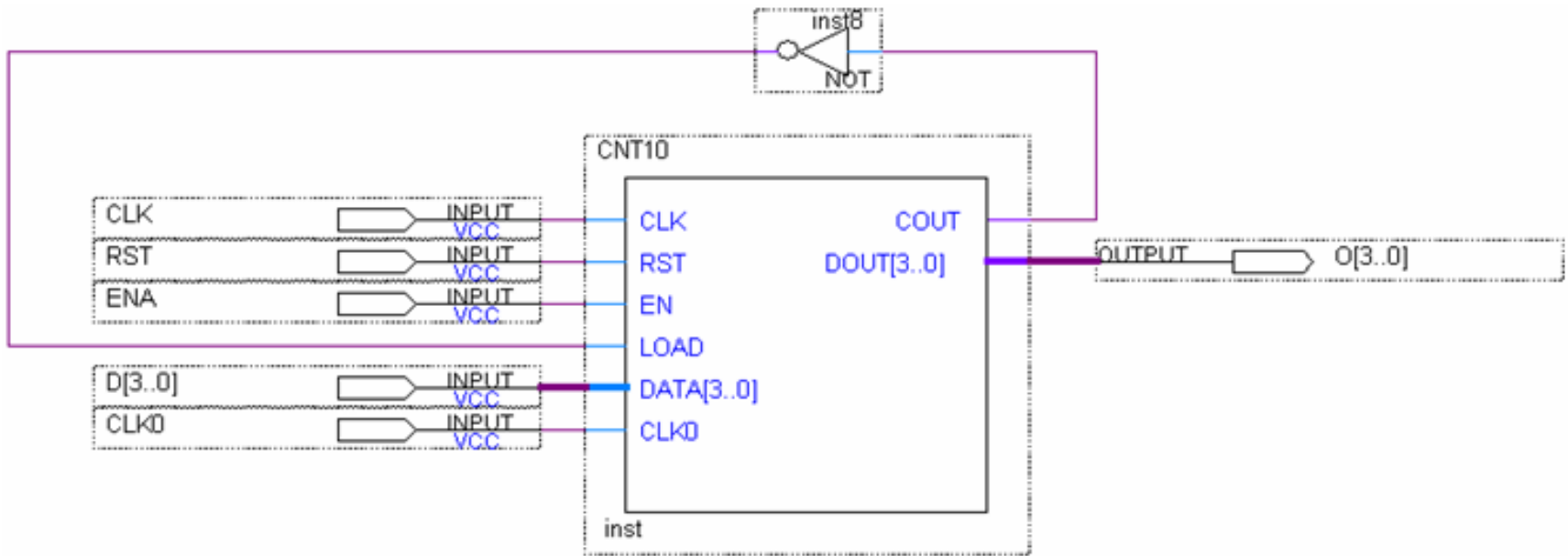
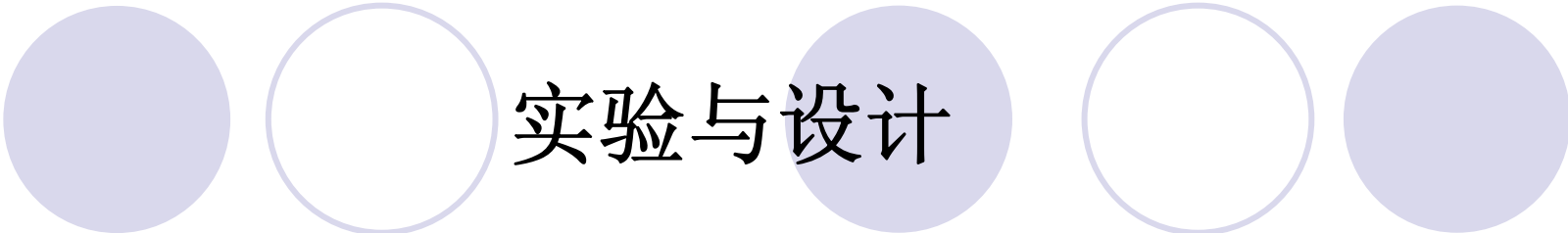


图 4-72 原理图示意图

## (9) 实验内容7:

## (10) 实验报告:



# 实验与设计

## 4-2 4选1多路选择器设计实验

(1) 实验目的:

(2) 实验内容1:

(3) 实验内容2:

(4) 实验内容3:

(5) 实验报告:

对于5E+系统的演示, 下载

**/KX\_7C5EE+/EXPERIMENTs/EXP0\_MUX41/MUX41B**。分别按下或放开键**K1**, **K2**, 蜂鸣器将发出4种不同声音。



# 实验与设计

## 4-4 十六进制7段数码显示译码器设计

- (1) 实验目的:
- (2) 实验原理:

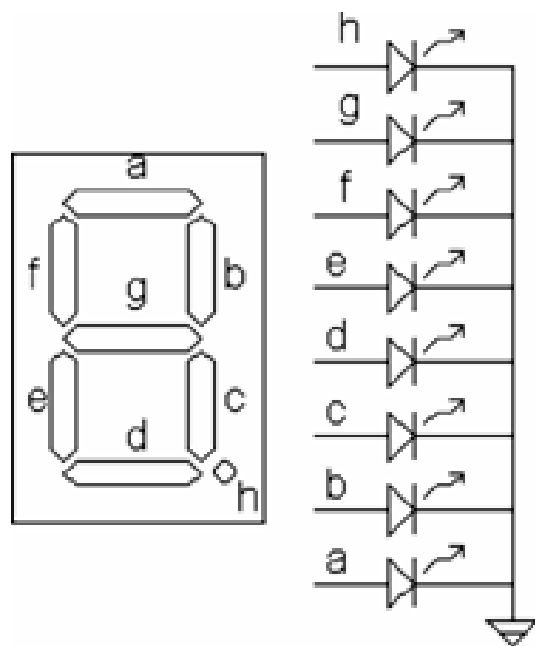


图 4-74 共阴数码管

表 4-2 7 段译码器真值表

| 输入码  | 输出码     | 代表数据 |
|------|---------|------|
| 0000 | 0111111 | 0    |
| 0001 | 0000110 | 1    |
| 0010 | 1011011 | 2    |
| 0011 | 1001111 | 3    |
| 0100 | 1100110 | 4    |
| 0101 | 1101101 | 5    |
| 0110 | 1111101 | 6    |
| 0111 | 0000111 | 7    |
| 1000 | 1111111 | 8    |
| 1001 | 1101111 | 9    |
| 1010 | 1110111 | A    |
| 1011 | 1111100 | B    |
| 1100 | 0111001 | C    |
| 1101 | 1011110 | D    |
| 1110 | 1111001 | E    |
| 1111 | 1110001 | F    |

# 实验与设计

## (3) 实验内容1:

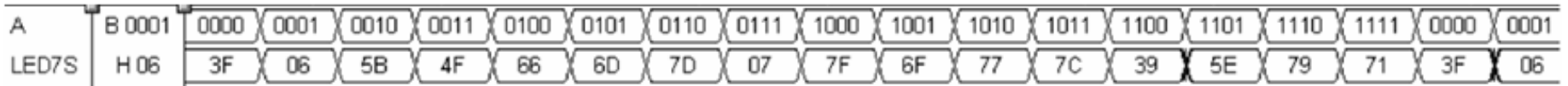


图 4-73 7 段译码器仿真波形

## (4) 实验内容2:

## (5) 实验内容3:

## (6) 实验内容4:

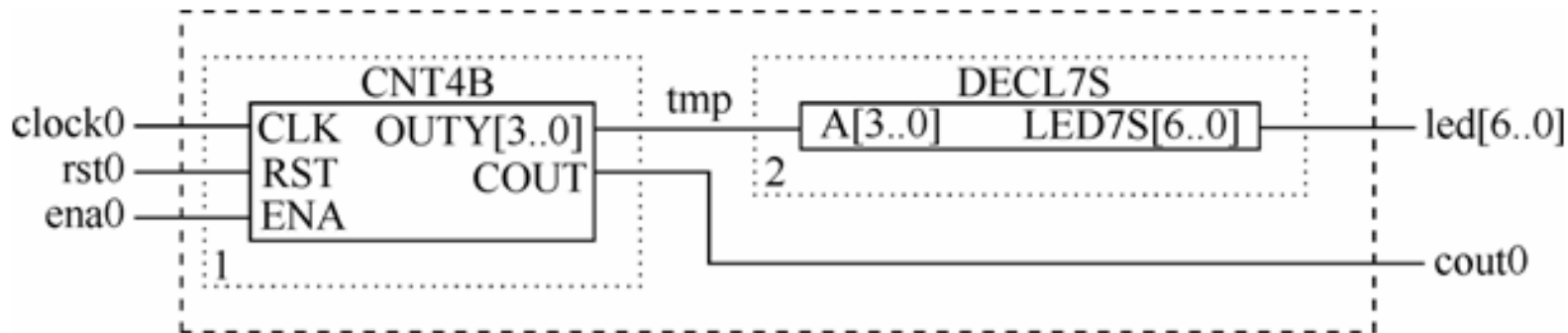
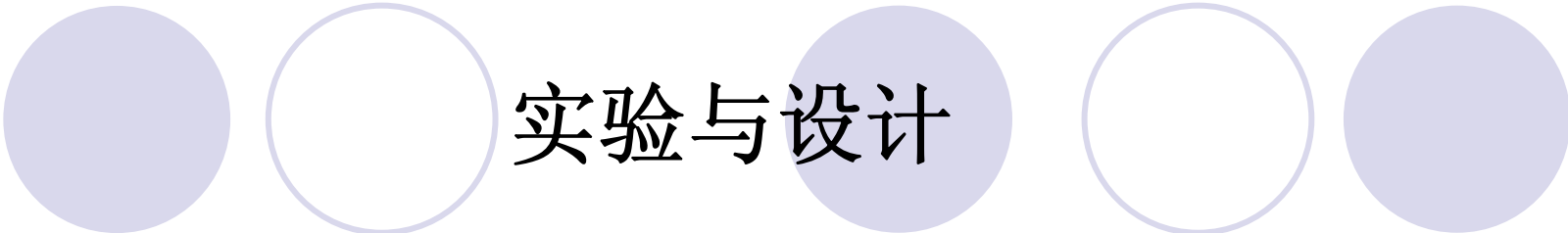


图 4-75 计数器和译码器连接电路的顶层文件原理图



# 实验与设计

## 4-5 原理图输入法设计8位十进制显示的频率计

(1) 实验目的:

(2) 原理说明:

(3) 实验内容1:

(4) 实验内容2:

(5) 实验内容3:

5E+系统的演示示例:

`/KX_7C5EE+/EXPERIMENTs/EXP10_FTEST_6LED/CNT6B。`

# 实验与设计

## 4-6 数码扫描显示电路设计

- (1) 实验目的:
- (2) 实验原理:

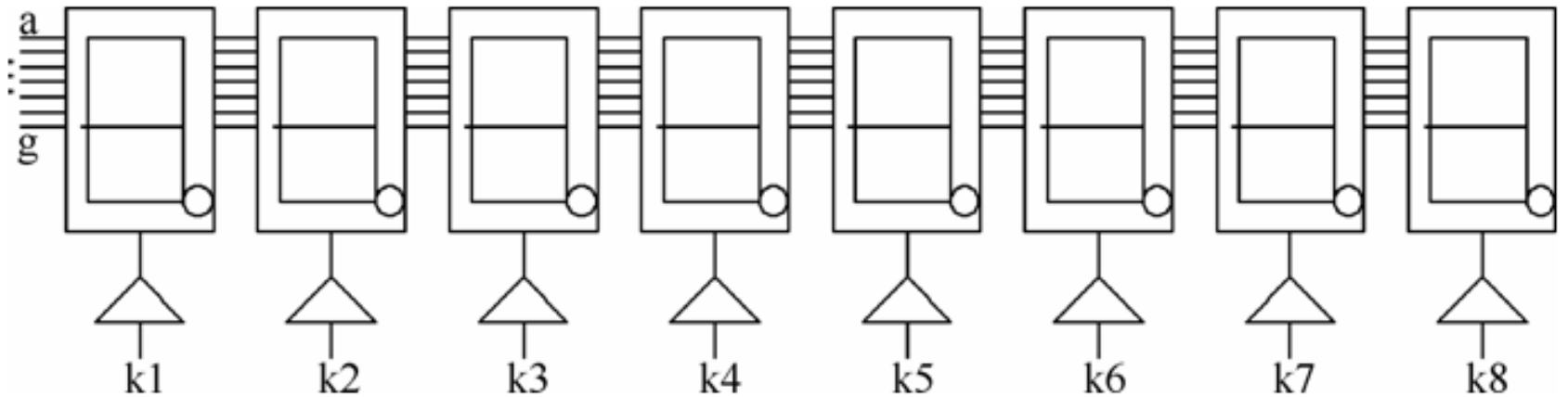


图 4-76 8 位数码扫描显示电路

- (3) 实验内容:

演示示例: /KX\_7C5EE+/EXPERIMENTs/EXP29\_SCAN\_LED/SCAN\_LED。