

第2章

系统设计与测试基础

2.1 原理图输入设计方法的特点

◎ 利用**EDA**工具进行原理图输入设计的优点是，设计者不必具备许多诸如编程技术、硬件语言等新知识就能迅速入门，完成基于**EDA**的较大规模的数字系统设计。

2.2 原理图输入方式基本设计流程

2.2.1 建立工作库文件夹和存盘原理图空文件

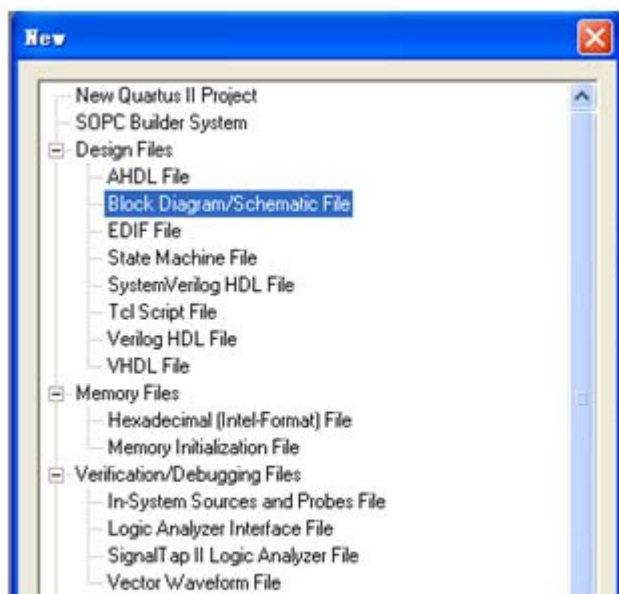


图 2-1 选择文件类型

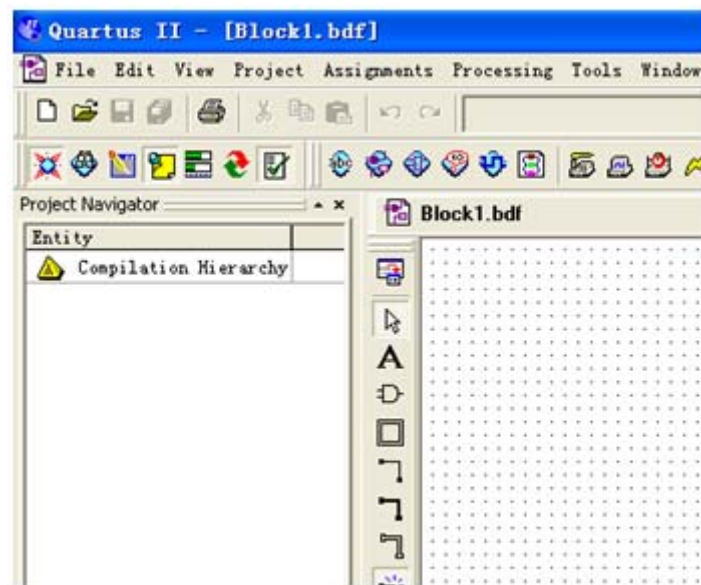


图 2-2 打开原理图编辑窗

2.2 原理图输入方式基本设计流程

2.2.2 创建工程



图 2-3 利用“New Preject Wizard”创建工程

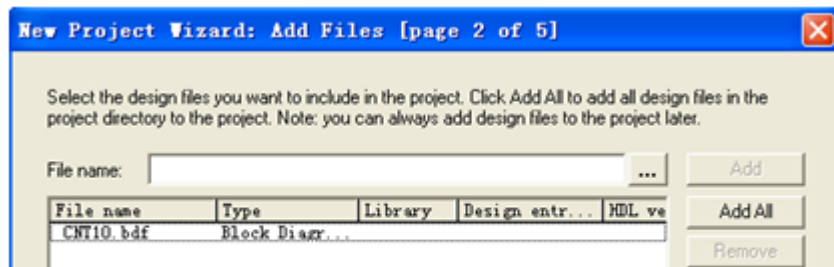


图 2-4 将所有相关的文件都加入进此工程

2.2 原理图输入方式基本设计流程

2.2.2 创建工程

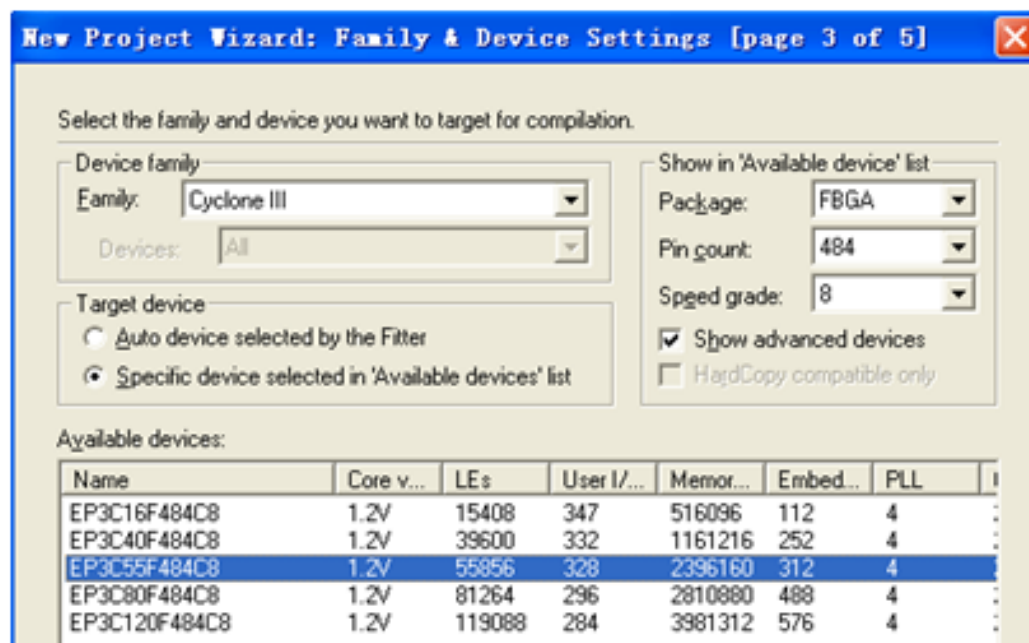


图 2-5 选择目标器件 EP3C55F484C8 型 FPGA

2.2 原理图输入方式基本设计流程

2.2.2 创建工程

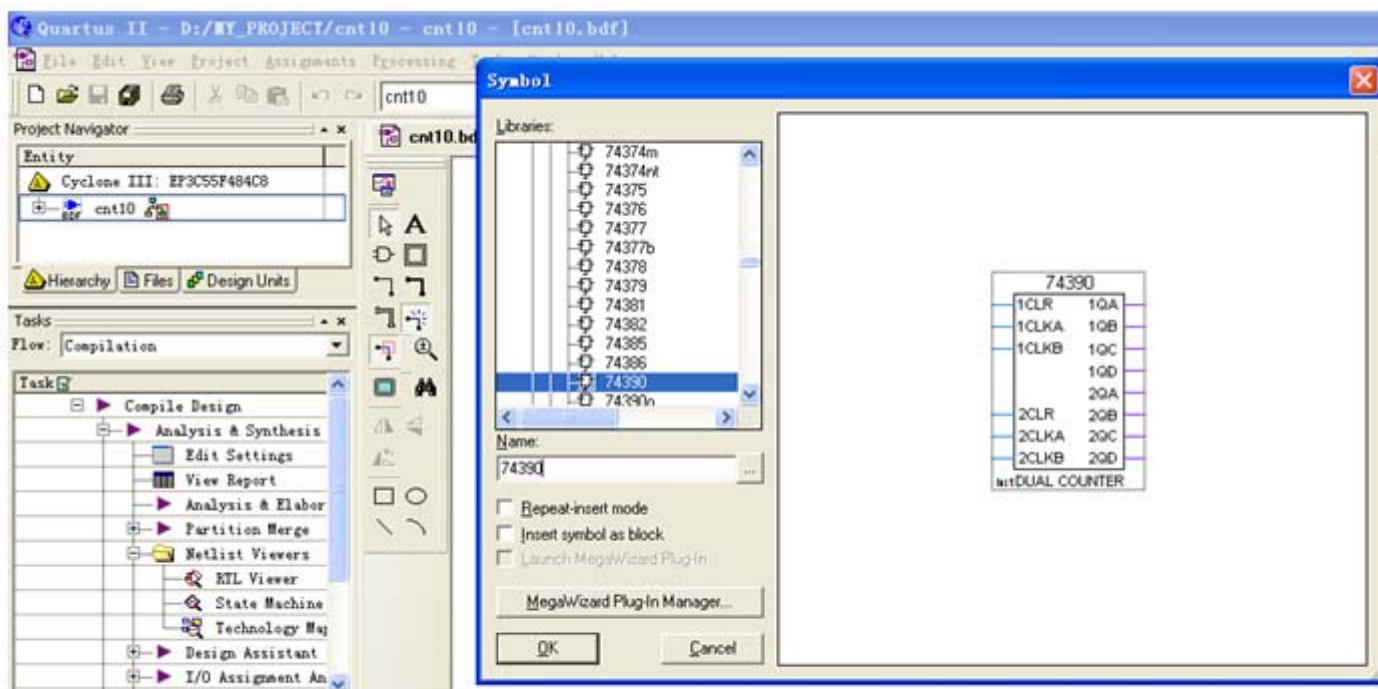


图 2-6 CNT10 工程管理窗（右侧是元件调用对话框调出需要的宏功能元件：74390 的图示）

2.2 原理图输入方式基本设计流程

2.2.3 功能初步分析

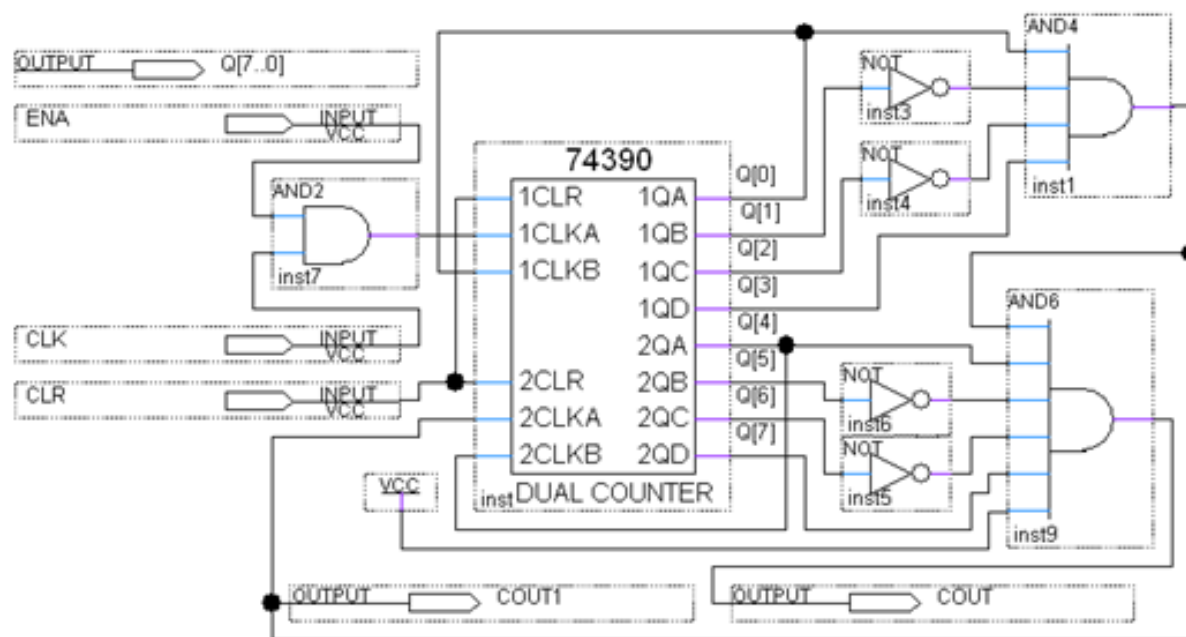
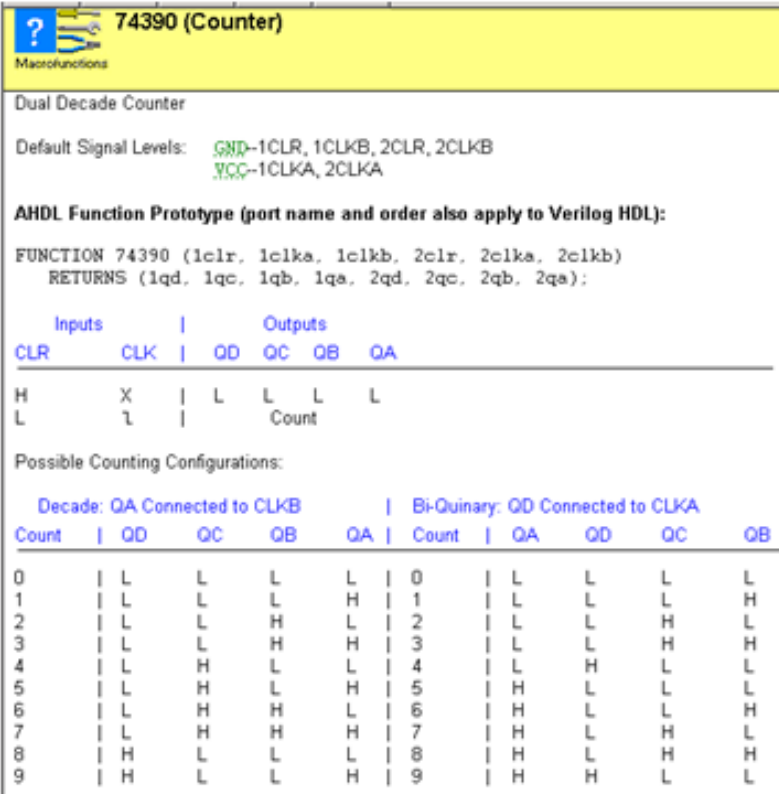


图 2-7 编辑好的两位十进制计数器 CNT10 的电路图

2.2 原理图输入方式基本设计流程

2.2.3 功能初步分析



74390 (Counter)
Macrofunctions

Dual Decade Counter

Default Signal Levels: **GND**-1CLR, 1CLKB, 2CLR, 2CLKB
VCC-1CLKA, 2CLKA

AHDL Function Prototype (port name and order also apply to Verilog HDL):

```
FUNCTION 74390 (1clr, 1clka, 1clkb, 2clr, 2clka, 2clkb)
  RETURNS (1qd, 1qc, 1qb, 1qa, 2qd, 2qc, 2qb, 2qa):
```

| Inputs | | Outputs | | | | |
|--------|-----|---------|----|----|----|--|
| CLR | CLK | QD | QC | QB | QA | |
| H | X | L | L | L | L | |
| L | 1 | Count | | | | |

Possible Counting Configurations:

| Decade: QA Connected to CLKB | | | | | Bi-Quinary: QD Connected to CLKA | | | | |
|------------------------------|----|----|----|----|----------------------------------|----|----|----|----|
| Count | QD | QC | QB | QA | Count | QA | QD | QC | QB |
| 0 | L | L | L | L | 0 | L | L | L | L |
| 1 | L | L | L | H | 1 | L | L | L | H |
| 2 | L | L | H | L | 2 | L | L | H | L |
| 3 | L | L | H | H | 3 | L | L | H | H |
| 4 | L | H | L | L | 4 | L | H | L | L |
| 5 | L | H | L | H | 5 | H | L | L | L |
| 6 | L | H | H | L | 6 | H | L | L | H |
| 7 | L | H | H | H | 7 | H | L | H | L |
| 8 | H | L | L | L | 8 | H | L | H | H |
| 9 | H | L | L | H | 9 | H | H | L | L |

图 2-8 来自 Macrofunctions 的 74390 的真值表

2.2 原理图输入方式基本设计流程

2.2.4 编译前设置

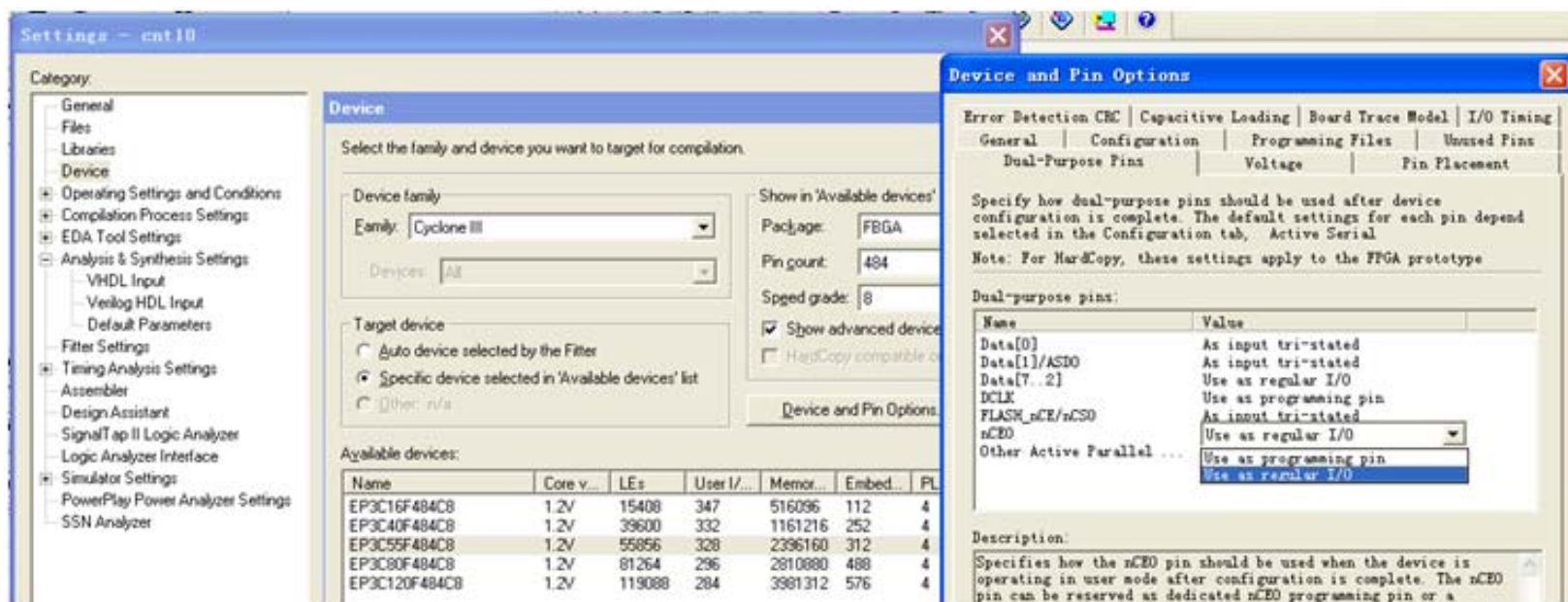


图 2-9 由 Settings 对话框选择目标器件 EP3C55F484C8 以及设定双功能引脚

2.2 原理图输入方式基本设计流程

2.2.4 编译前设置

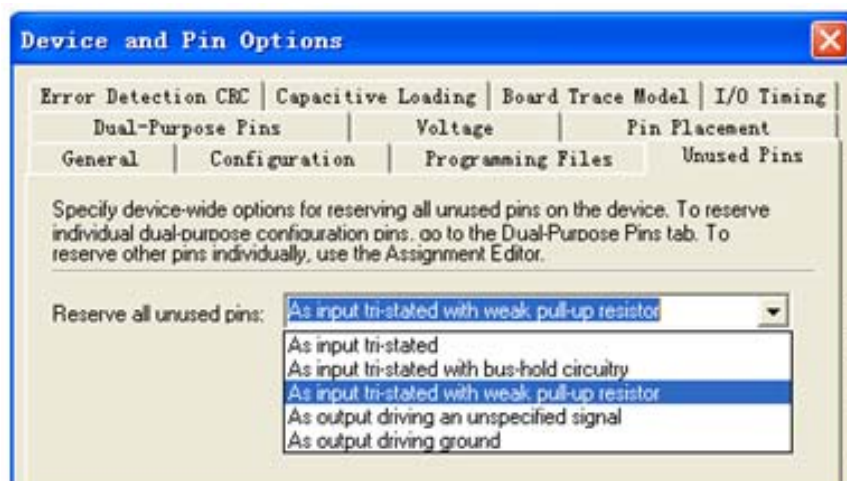


图 2-10 选择闲置引脚的状态

2.2 原理图输入方式基本设计流程

2.2.5 全程编译

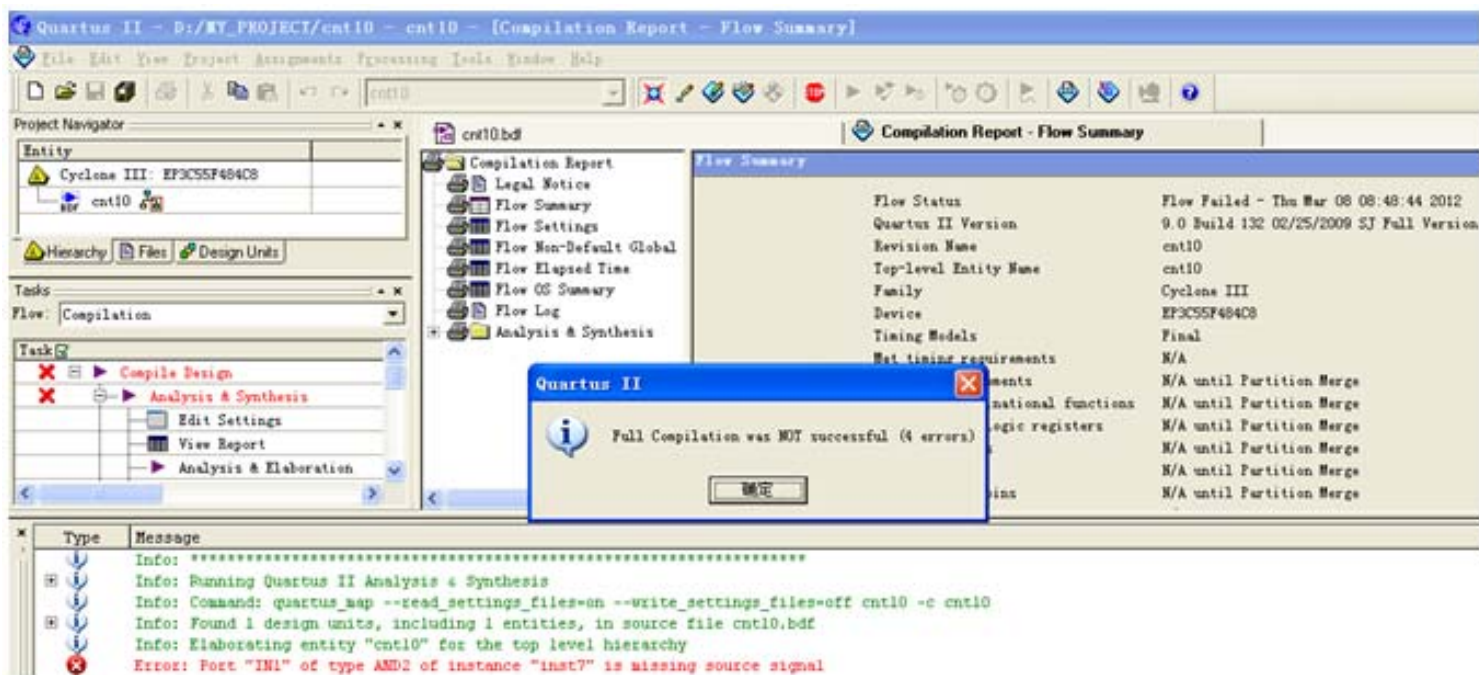


图 2-11 全程编译后出现报错信息

2.2 原理图输入方式基本设计流程

2.2.6 时序仿真测试电路功能

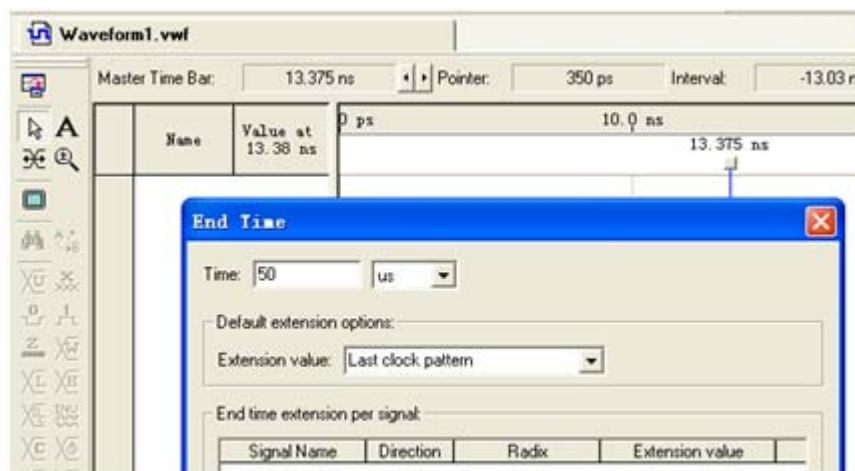


图 2-12 设置仿真时间长为 50 微秒

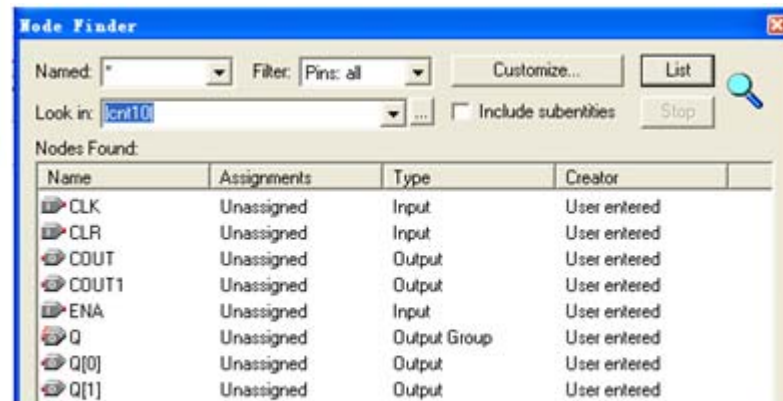


图 2-13 Node Finder 窗中待拖入的信号节点

2.2 原理图输入方式基本设计流程

2.2.6 时序仿真测试电路功能

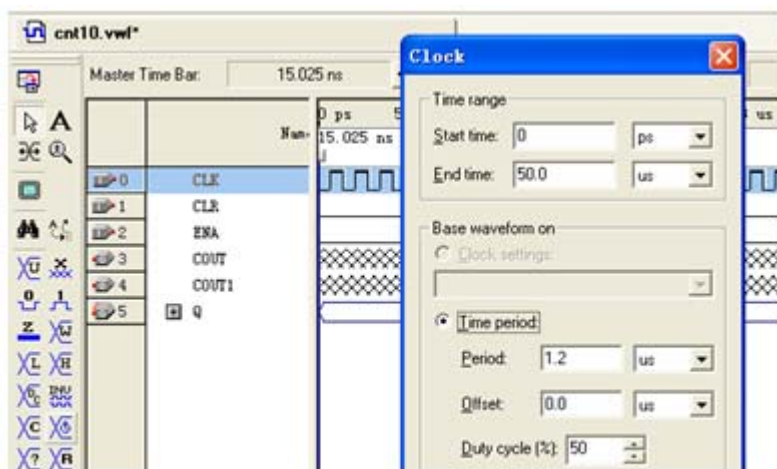


图 2-14 准备给 CLK 设置时钟（右窗为 CLK 设置周期）

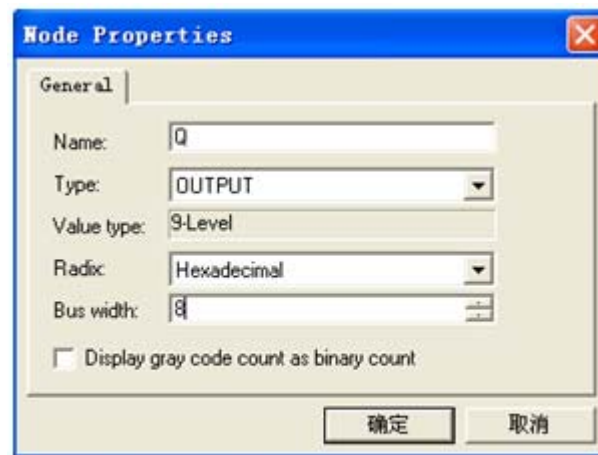


图 2-15 为总线 Q 设置 16 进制数制

2.2 原理图输入方式基本设计流程

2.2.6 时序仿真测试电路功能

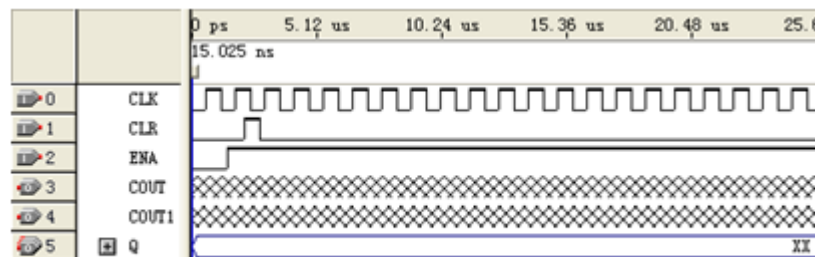


图 2-16 设置好的激励信号波形图

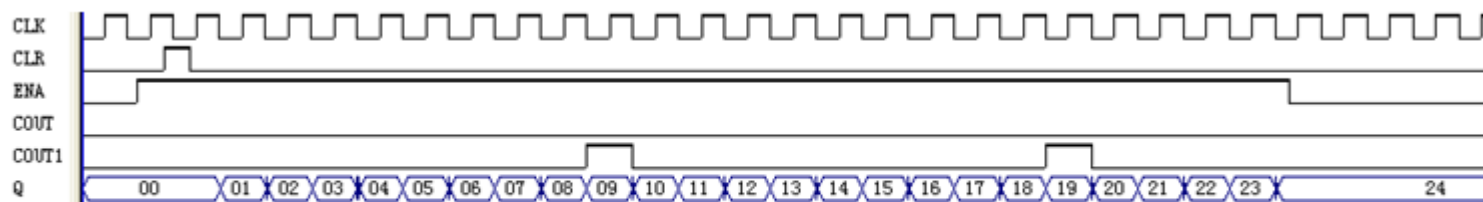


图 2-17 图 2-7 电路的仿真波形

2.3 引脚设置和编程下载

2.3.1 引脚锁定

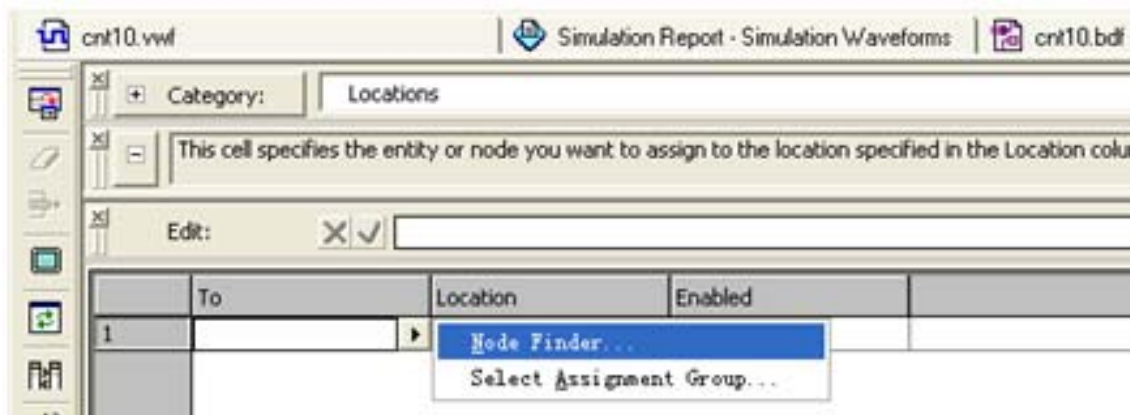


图 2-18 利用 Assignment Editor 编辑器锁定 FPGA 引脚

2.3 引脚设置和编程下载

2.3.1 引脚锁定

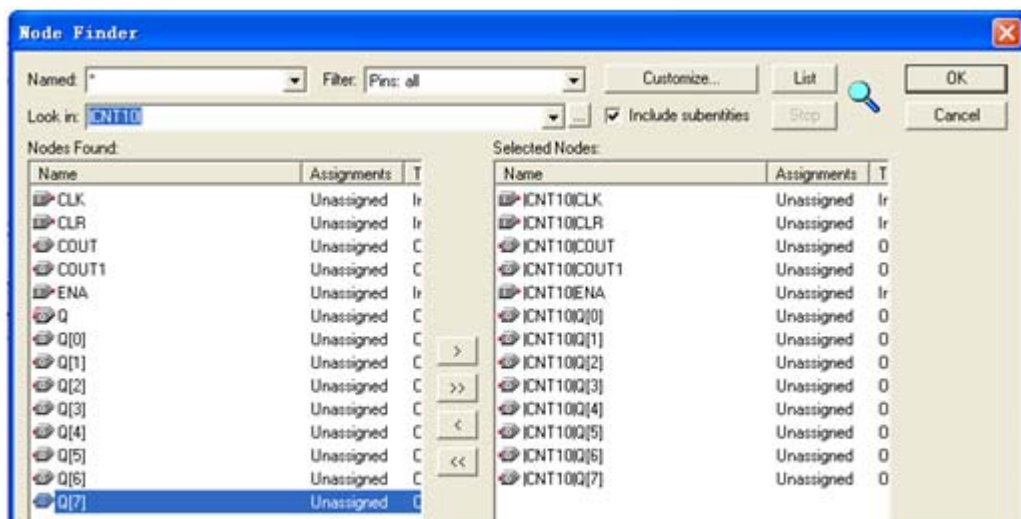


图 2-19 选择需要锁定的引脚信号

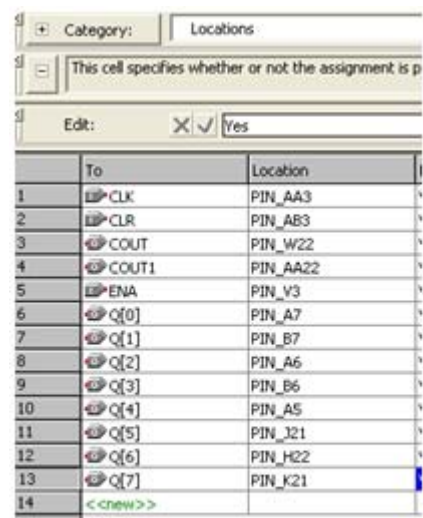


图 2-20 引脚锁定对话框

2.3 引脚设置和编程下载

2.3.2 配置文件下载

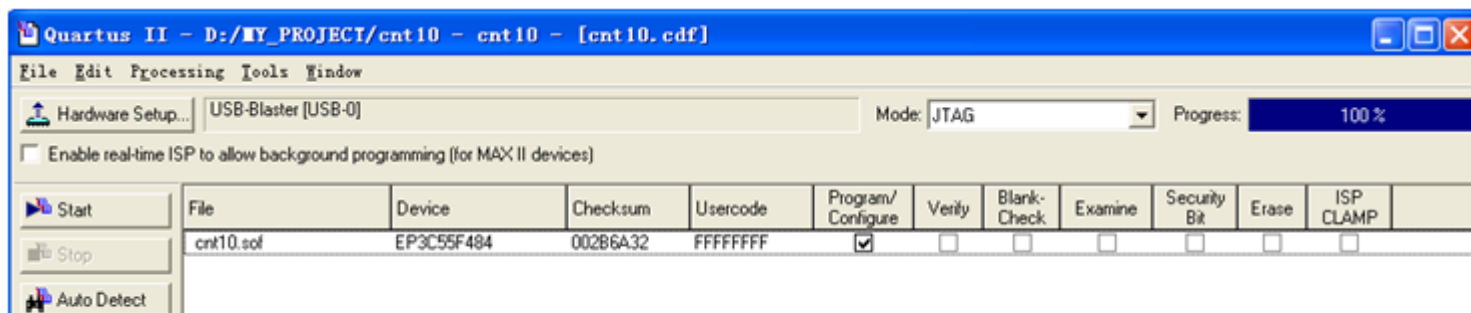


图 2-21 选择编程下载文件和下载模式



图 2-22 加入编程下载方式

2.3 引脚设置和编程下载

2.3.3 JTAG间接编程模式

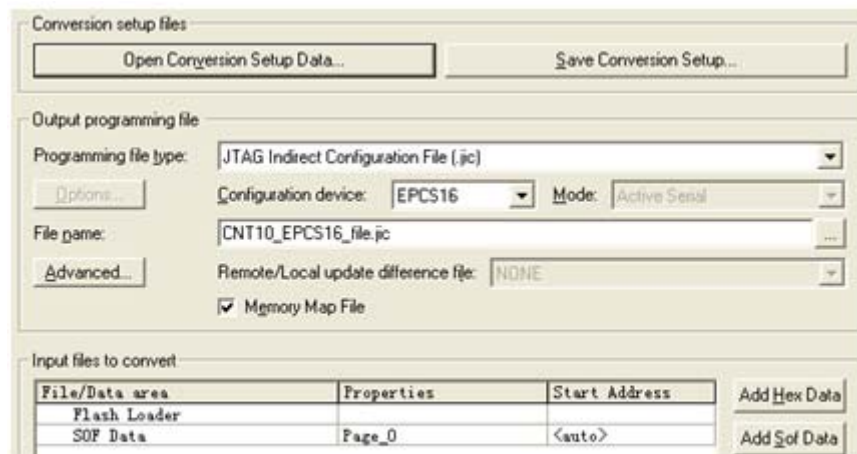


图 2-23 设定 JTAG 间接编程文件

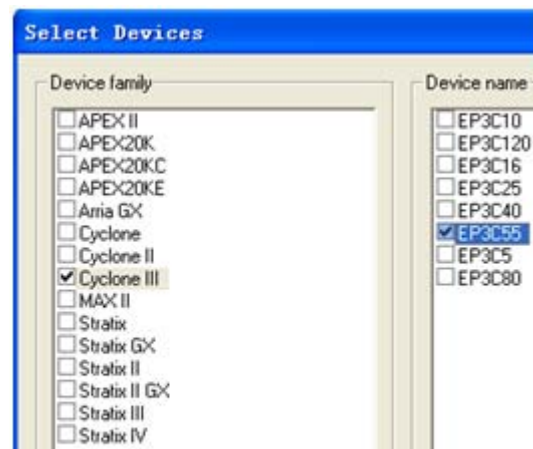


图 2-24 选择目标器件 EP3C55

2.3 引脚设置和编程下载

2.3.3 JTAG间接编程模式

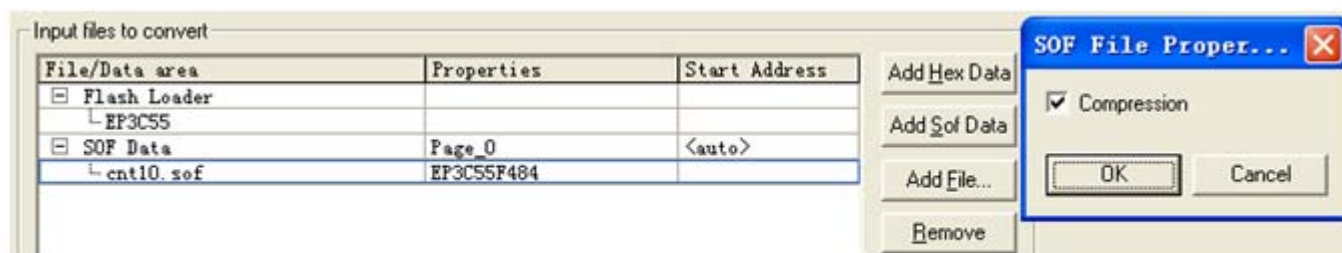


图 2-25 加入 SOF 文件，选择文件压缩（右窗）

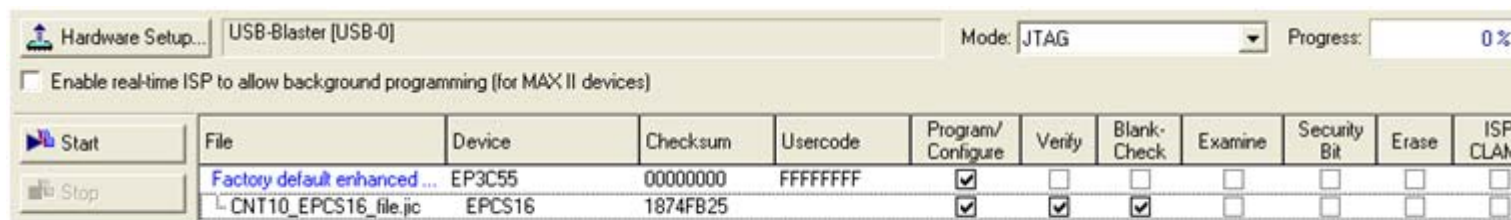


图 2-26 用 JTAG 模式对配置器件 EPCS16 进行编程

2.3.4 USB-Blaster编程配置器安装方法

2.4 层次化设计

1. 构建元件符号

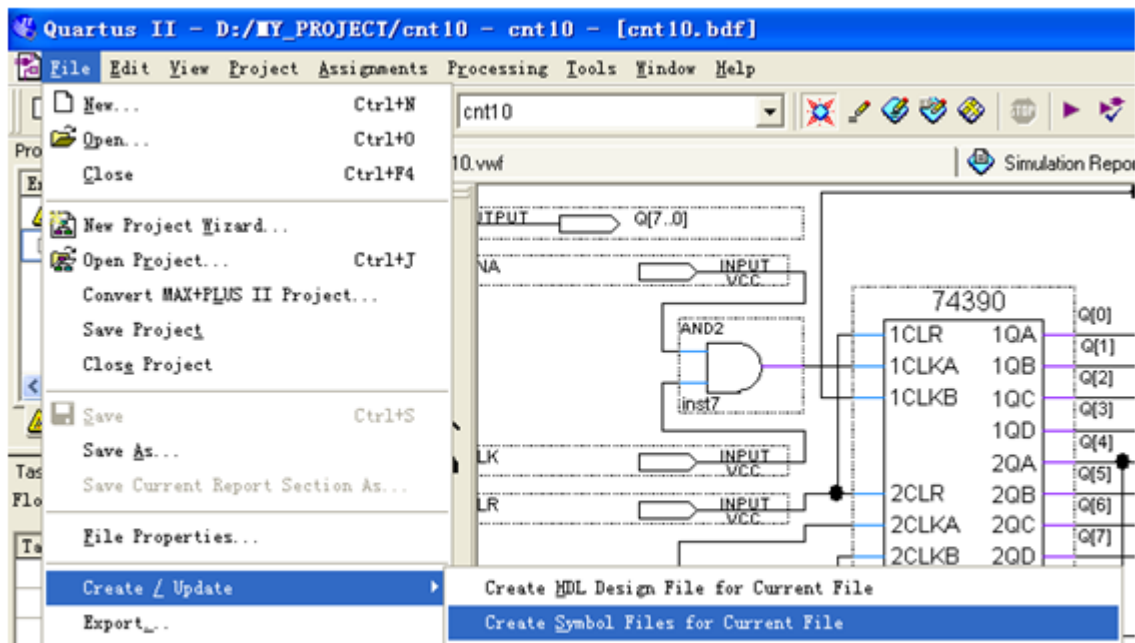


图 2-27 将当前电路原理图设计生成为一个元件 (Symbol) 模块

2.4 层次化设计

2. 构建顶层文件

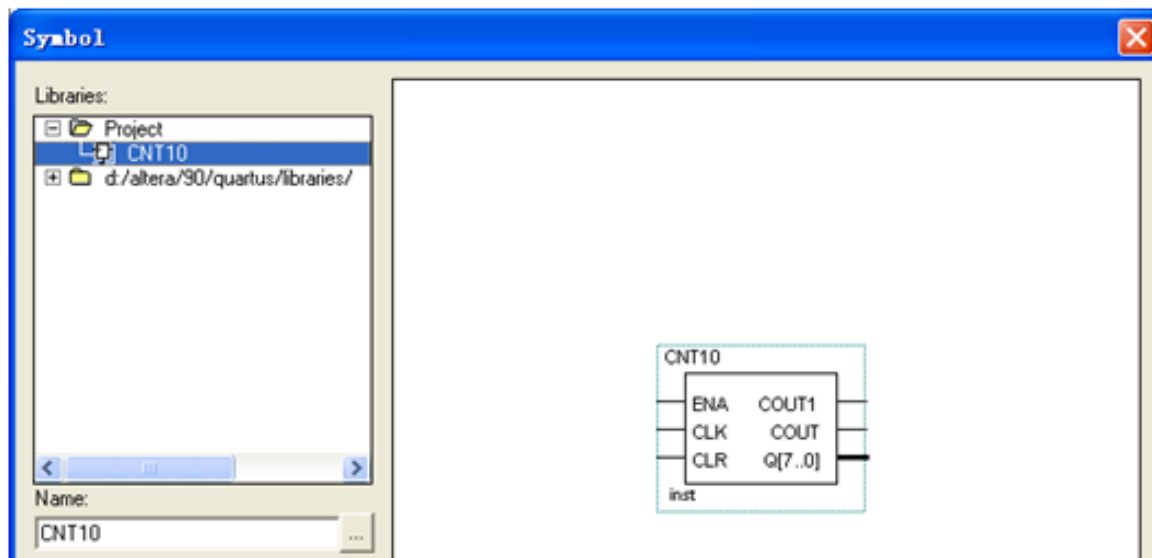


图 2-28 在高一层原理图的当前工程路径中调入元件 CNT10

2.4 层次化设计

2. 构建顶层文件

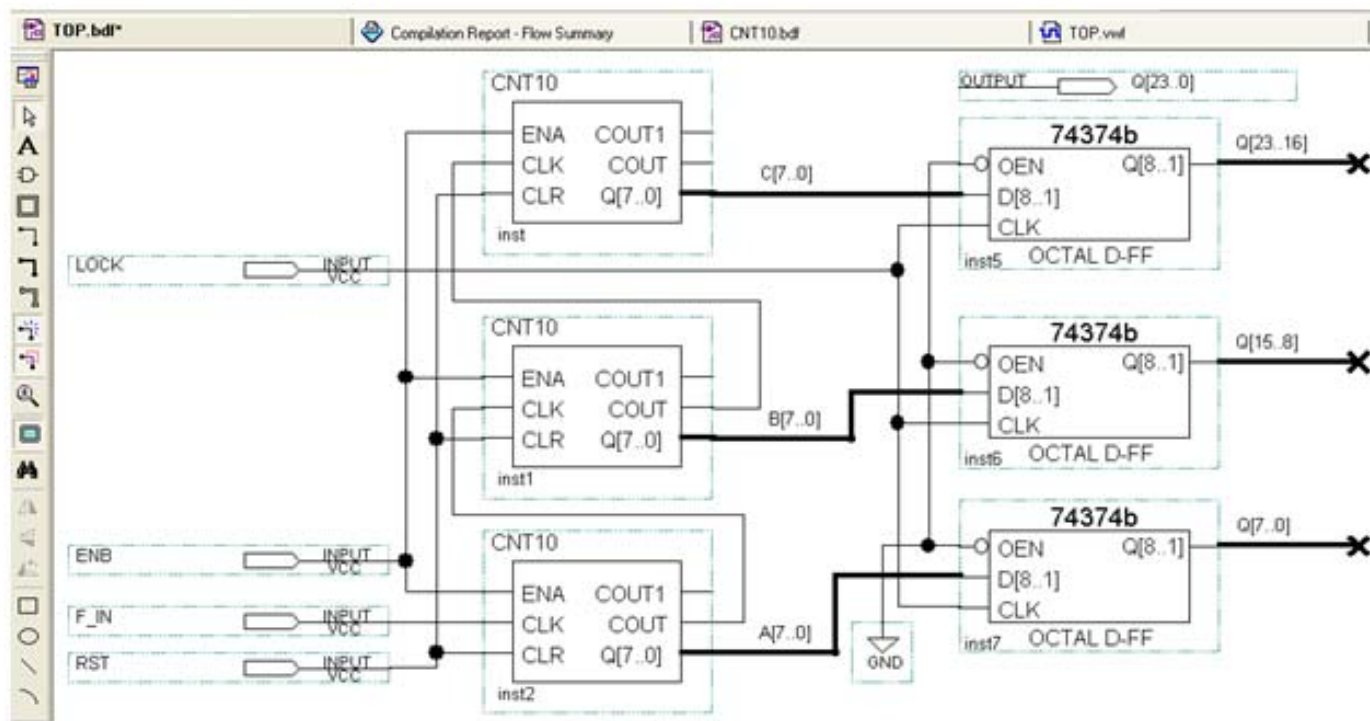


图 2-29 基于元件 CNT10 扩展的 6 位十进制计数器

2.4 层次化设计

3. 功能分析和全程编译

4. 时序仿真



图 2-30 图 2-29 电路的仿真波形图

2.5 SignalTap II的使用方法

1. 打开SignalTap II编辑窗口
2. 调入待测信号

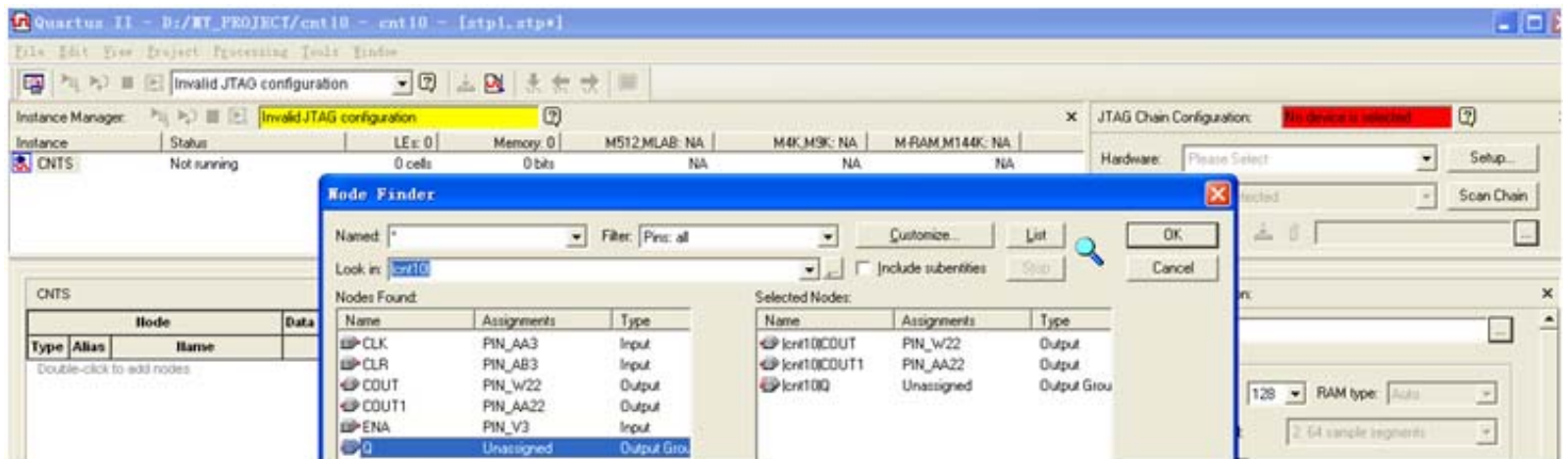


图 2-31 输入逻辑分析仪测试信号

2.5 SignalTap II的使用方法

3. SignalTap II参数设置

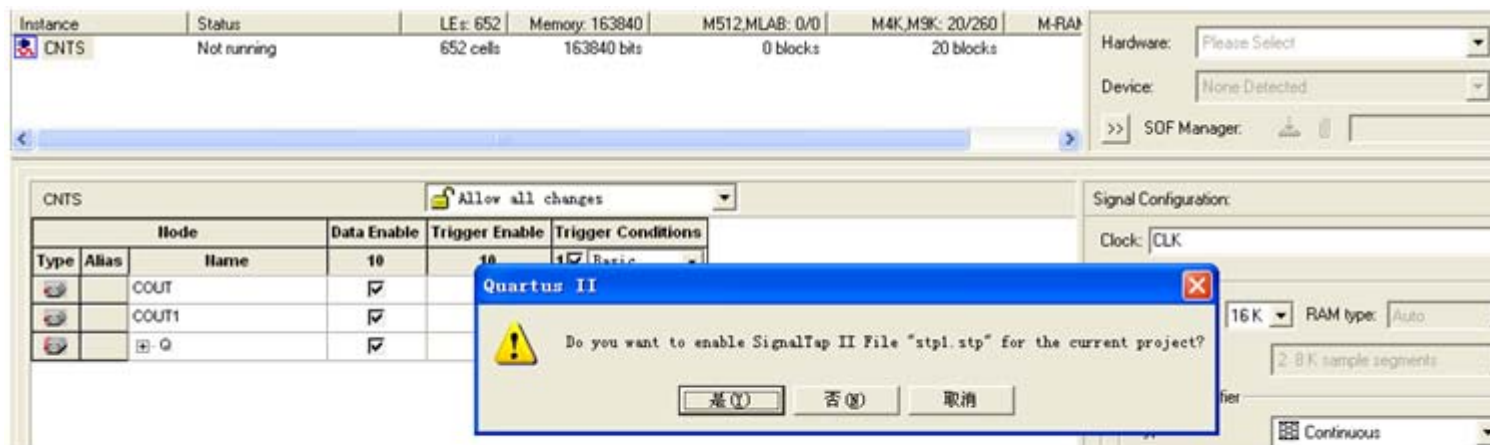


图 2-32 SignalTap II 编辑窗口

4. 文件存盘

2.5 SignalTap II的使用方法

5. 编译下载



图 2-33 SignalTap II 数据窗口显示实时在系统采用获得的信号数据

2.5 SignalTap II 的使用方法

6. 启动SignalTap II进行采样与分析

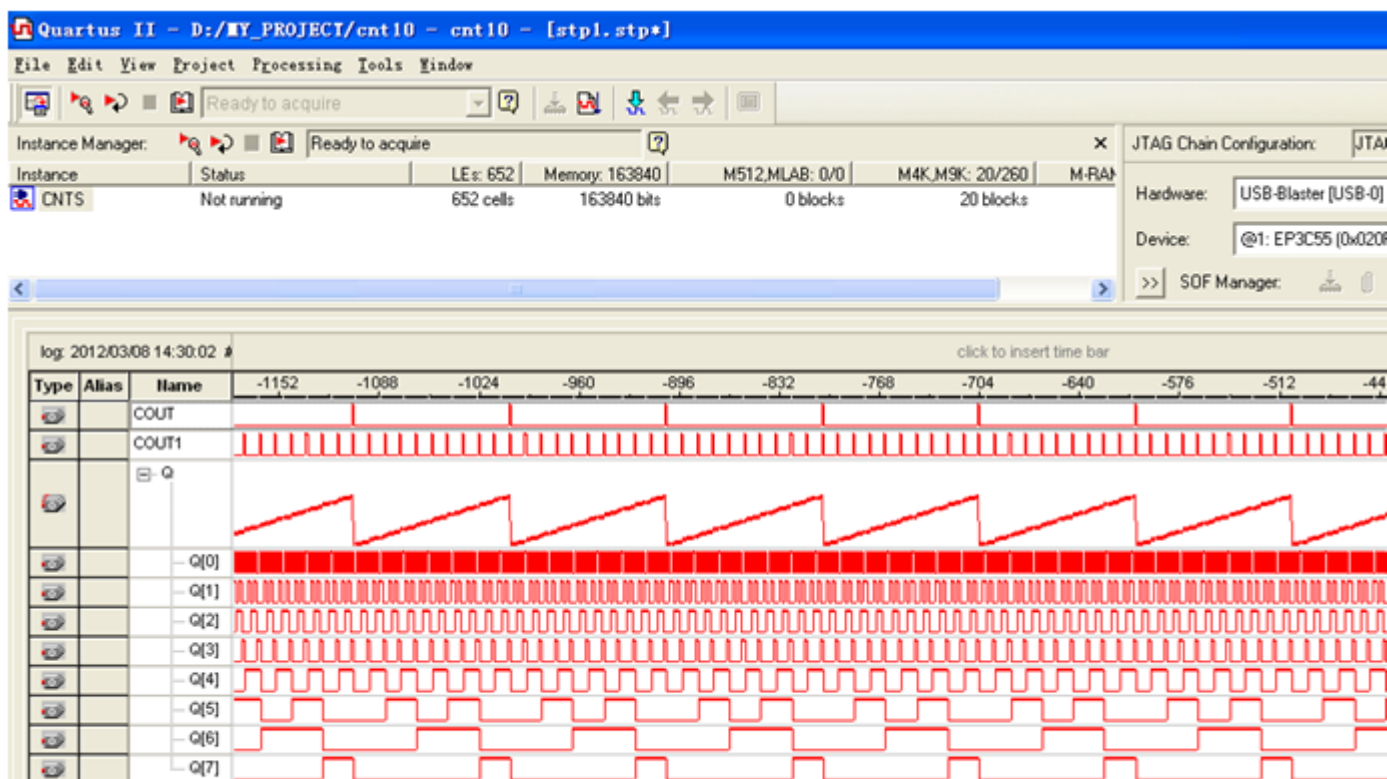


图 2-34 SignalTap II 数据窗口显示对硬件系统实时测试采样后的信号波形

实验

2-1 用原理图输入法设计8位加法器

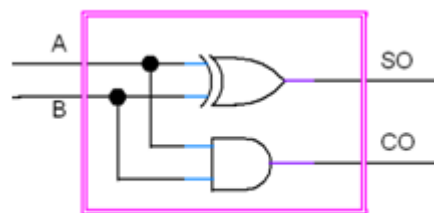


图 2-35 半加器 h_adder 电路图

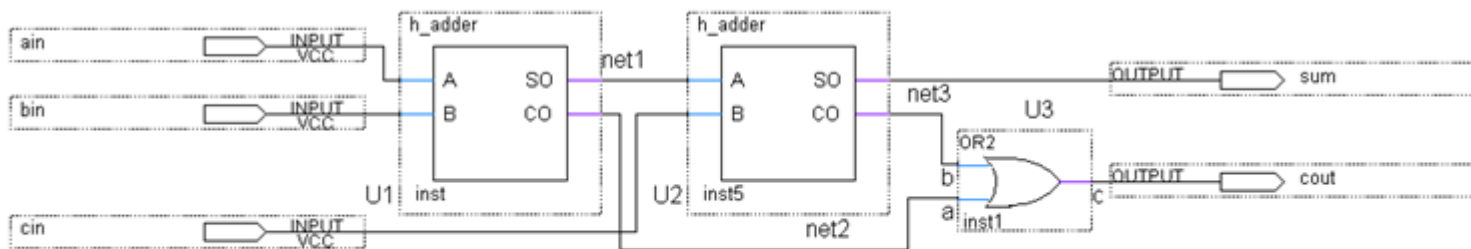


图 2-36 全加器 f_adder 电路图

实验

2-2. 原理图输入法设计频率计

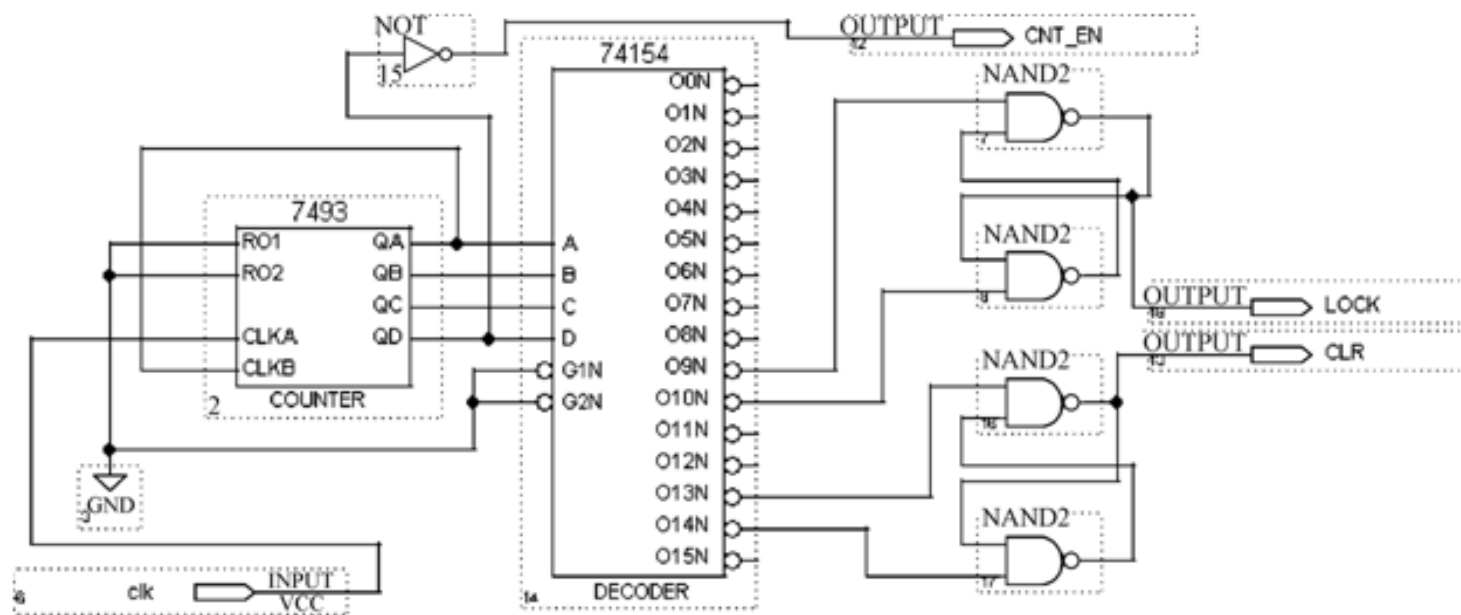


图 2-37 测频控制电路

2-3 计时系统设计