

# 第4章

## 时序仿真与硬件实现

# 4.1 VHDL程序输入与仿真测试

## 4.1.1 编辑和输入设计文件

(1) 新建一个文件夹。

(2) 输入源程序。

(3) 文件存盘。

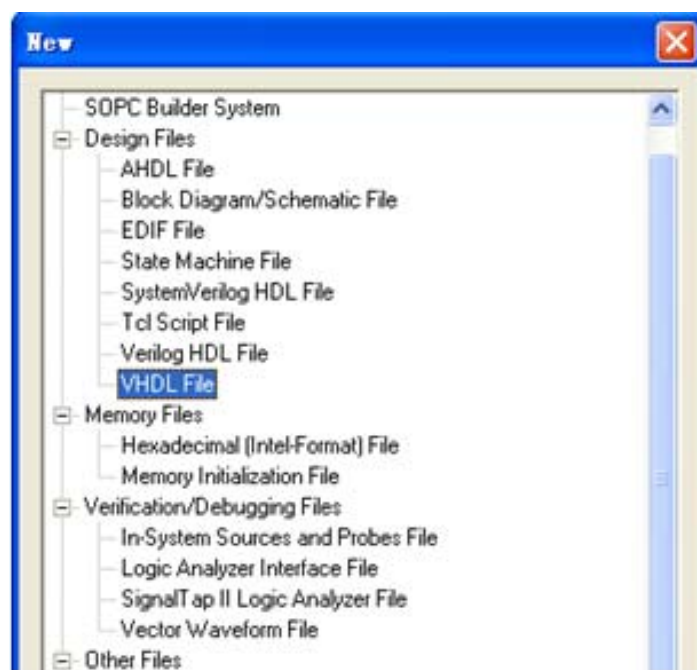


图 4-1 选择编辑文件类型

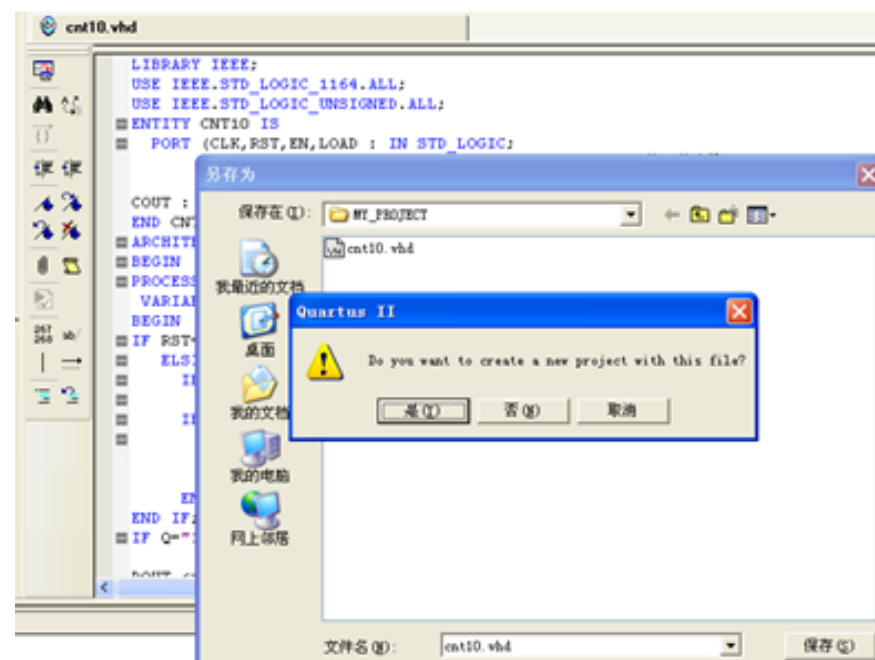


图 4-2 编辑输入源程序并存盘

# 4.1 VHDL程序输入与仿真测试

## 4.1.2 创建工程

- (1) 打开并建立新工程管理窗口。
- (2) 将设计文件加入工程中。
- (3) 选择目标芯片。



图 4-3 利用 New Project Wizard 创建工程 MULT4B

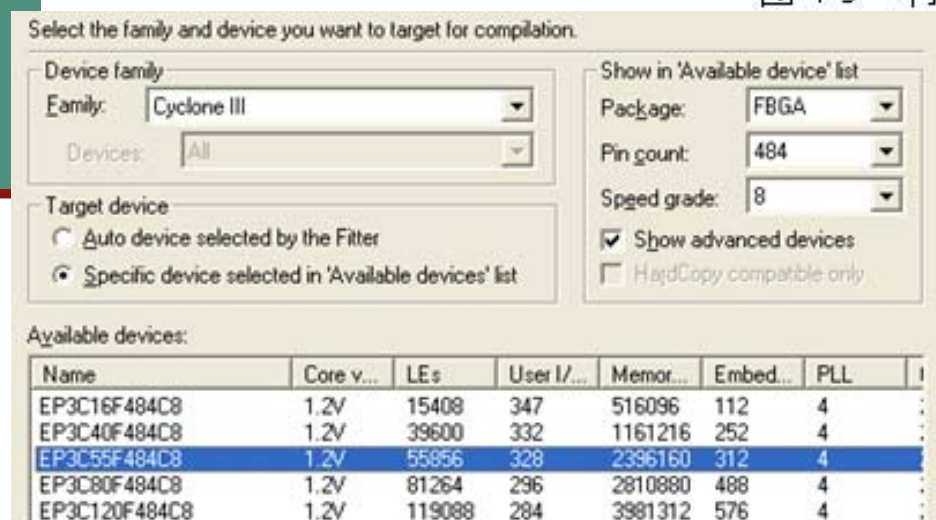


图 4-4 选择目标器件 EP3C55F484C8

(4) 工具设置。

(5) 结束设置。

# 4.1 VHDL程序输入与仿真测试

## 4.1.3 全程编译前约束项目设置

(1) 选择FPGA  
目标芯片。

(2) 选择配置器件  
的工作方式。

(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) 对双功能引脚进行设置。

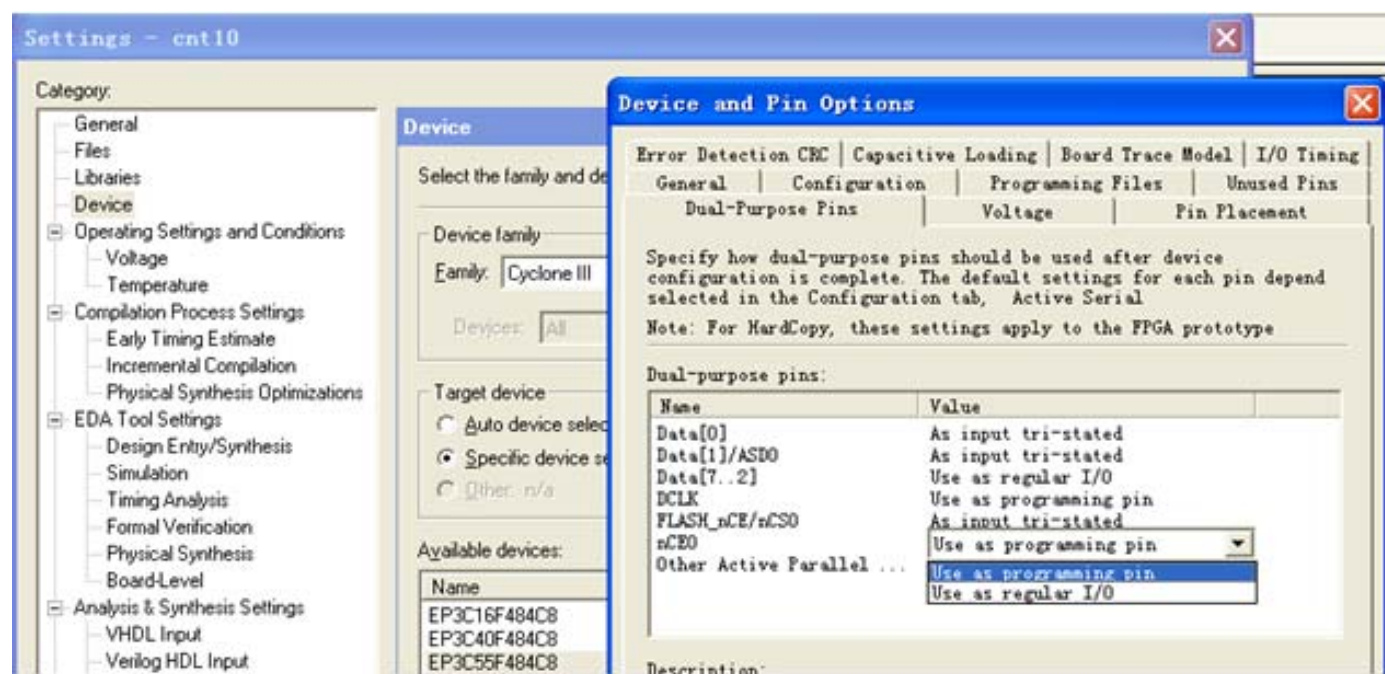


图 4-5 选择配置器件的工作方式

# 4.1 VHDL程序输入与仿真测试

## 4.1.4 全程综合与编译

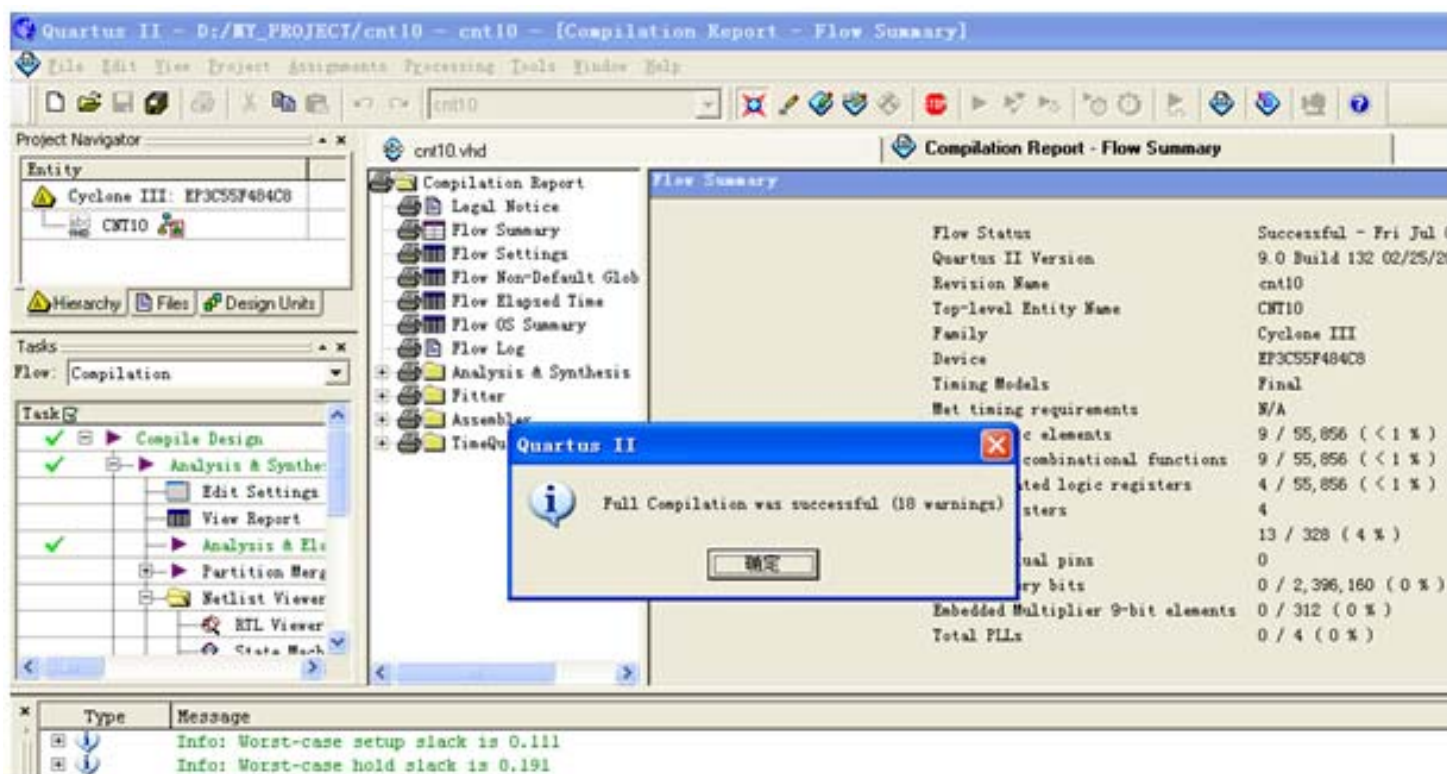


图 4-6 全程编译无错后的报告信息

# 4.1 VHDL程序输入与仿真测试

## 4.1.5 仿真测试

(1) 打开波形编辑器。

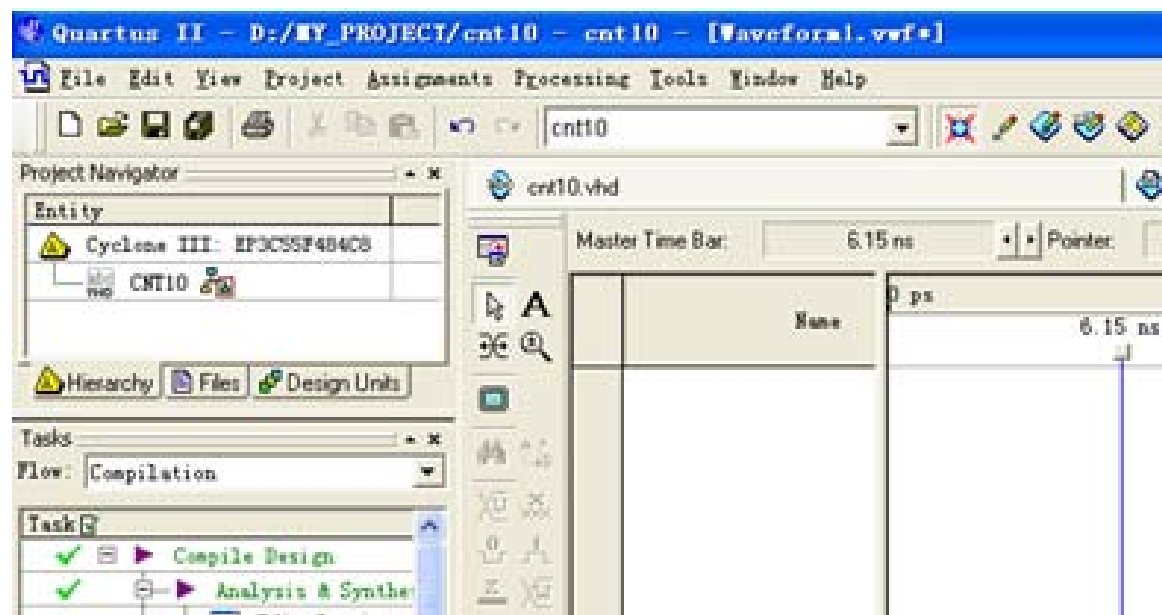


图 4-7 激励信号波形编辑器窗口

# 4.1 VHDL程序输入与仿真测试

## 4.1.5 仿真测试

(2) 设置仿真时间区域。

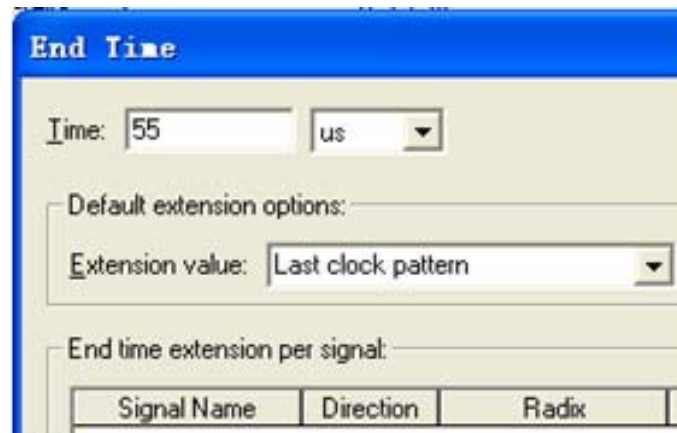


图 4-8 设置仿真时间长度

(3) 波形文件存盘。

# 4.1 VHDL程序输入与仿真测试

## 4.1.5 仿真测试

(4) 将工程**CNT10**的端口信号节点选入波形编辑器中。

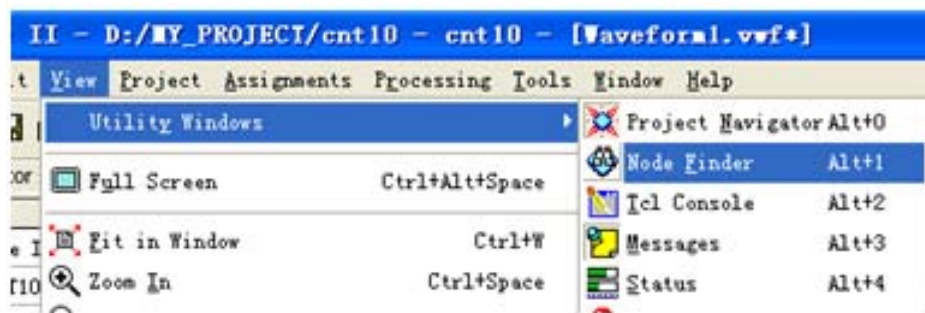


图 4-9 打开信号节点查询窗口

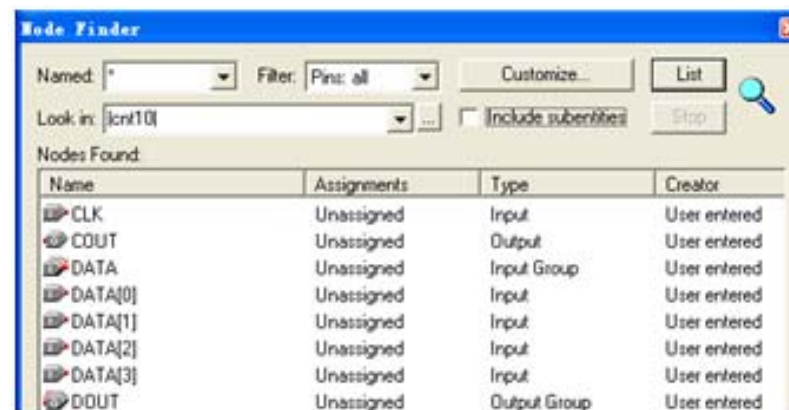


图 4-10 向波形编辑器拖入信号节点



# 4.1 VHDL程序输入与仿真测试

## 4.1.5 仿真测试

(5) 设置激励信号波形。

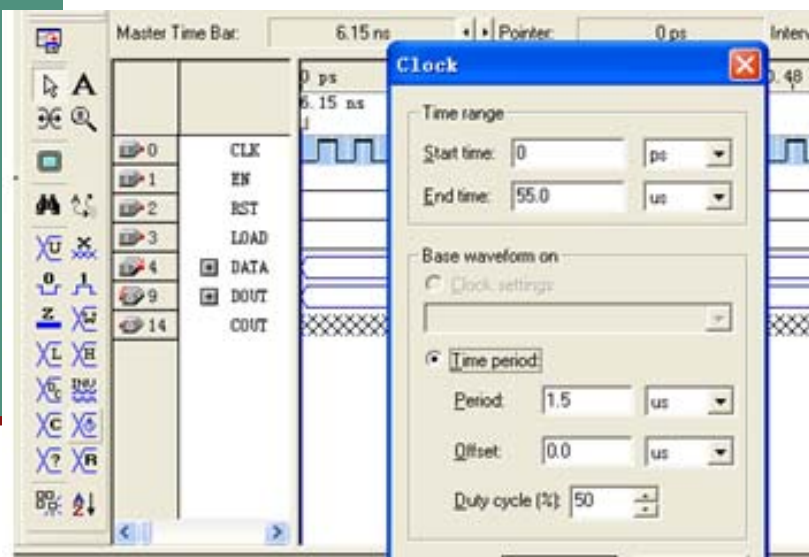


图 4-11 设置好的激励波形图，及选择总线数据格式



图 4-12 设置信号显示数据格式

# 4.1 VHDL程序输入与仿真测试

## 4.1.5 仿真测试

(6) 图4-13是最后设置好的vwf仿真激励波形文件图。

(7) 仿真器参数设置。

(8) 启动仿真器。

(9) 观察仿真结果。

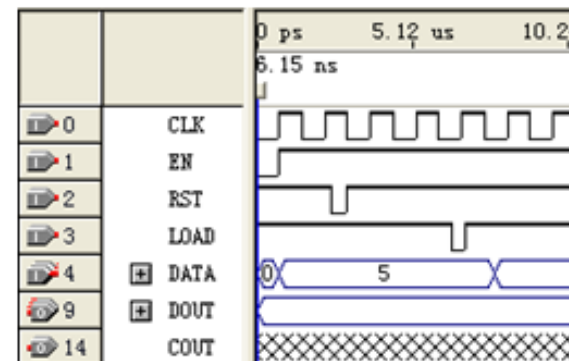
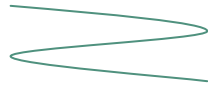


图 4-13 编辑好的 vwf 文件

# 4.1 VHDL程序输入与仿真测试

## 4.1.6 RTL图观察器应用

**RTL Viewer**



HDL的RTL级图形观察器

**Technology Map Viewer**



HDL对应的FPGA底层门级布局观察器

**State Machine Viewer**



HDL对应状态机的状态图观察器

## 4.2 引脚锁定与硬件测试

### 4.2.1 引脚锁定

表 4-1 基于 EP3C55F484 的 55F+开发板的引脚锁定表

CLK	LOAD	COUT	DATA(3)	DATA(2)	DATA(1)	DATA(0)
键 1,K1	键 4,K4	LED1	拨码 4	拨码 4	拨码 4	拨码 4
Pin: AA3	Pin: Y7	Pin: B22	Pin: N22	Pin: M22	Pin: L21	Pin: L22
DOUT[3..0]	DOUT(3)	DOUT(2)	DOUT(1)	DOUT(0)	EN	RST
LED 显示	A3	A2	A1	A0	键 3,K3	键 2,K2
引脚	Pin: A3	Pin: A4	Pin: C6	Pin: H21	Pin: V3	Pin: AB3

## 4.2 引脚锁定与硬件测试

### 4.2.1 引脚锁定

- (1) 假设现在已打开了**CNT10**工程。
- (2) 选择**Assignments** → **Assignment Editor**命令

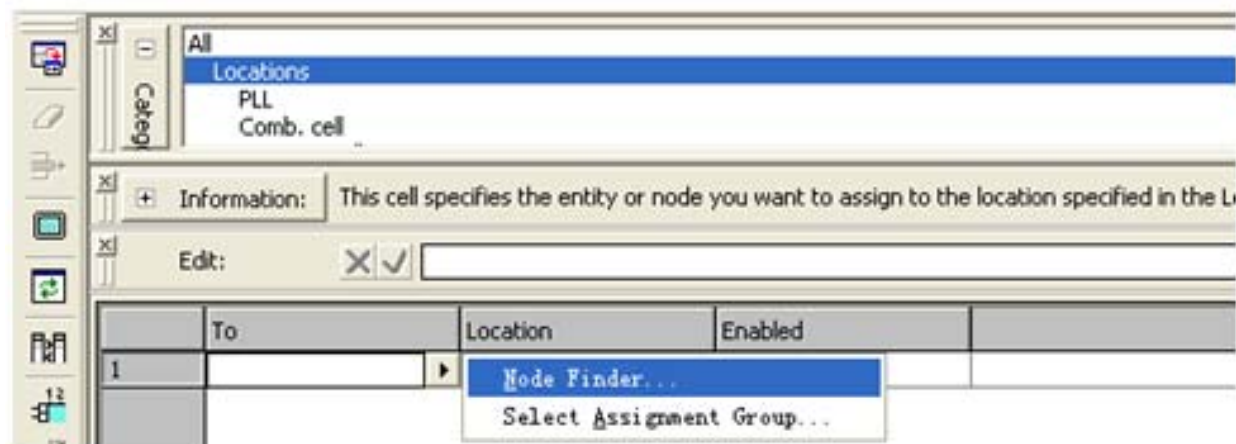


图 4-14 利用 Assignment Editor 编辑器锁定 FPGA 引脚

## 4.2 引脚锁定与硬件测试

### 4.2.1 引脚锁定

(3) 双击TO栏的《new》

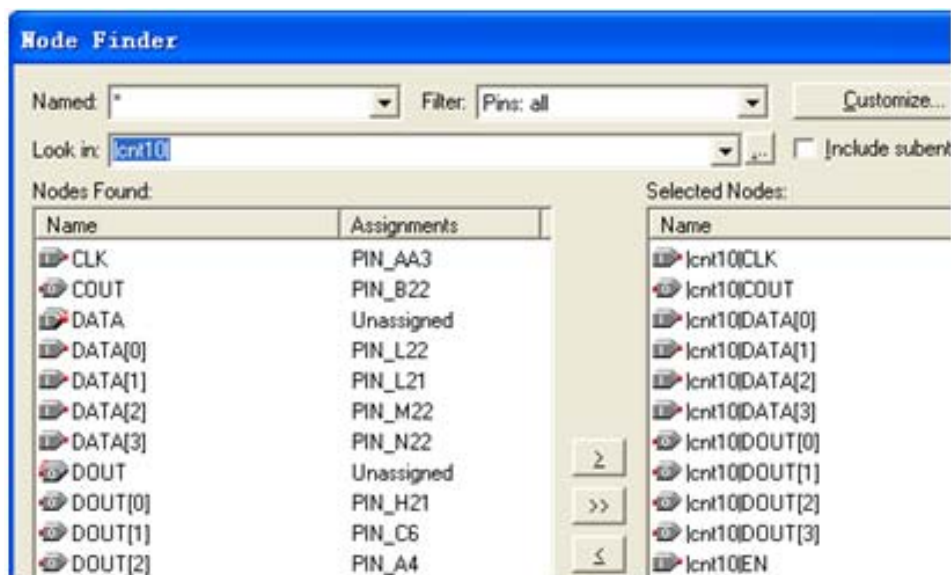


图 4-15 利用 Node Finder 工具选择需要锁定引脚的信号

	To	Location
1	CLK	PIN_AA3
2	COUT	PIN_B22
3	DATA[0]	PIN_L22
4	DATA[1]	PIN_L21
5	DATA[2]	PIN_M22
6	DATA[3]	PIN_N22
7	DOUT[0]	PIN_H21
8	DOUT[1]	PIN_C6
9	DOUT[2]	PIN_A4
10	DOUT[3]	PIN_A3
11	EN	PIN_V3
12	LOAD	PIN_Y7
13	RST	PIN_AB3
14	<<new>>	

图 4-16 引脚锁定窗口

(4) 注意在键入所希望的引脚编号

## 4.2 引脚锁定与硬件测试

### 4.2.2 编译文件下载

(1) 打开编程窗和配置文件。

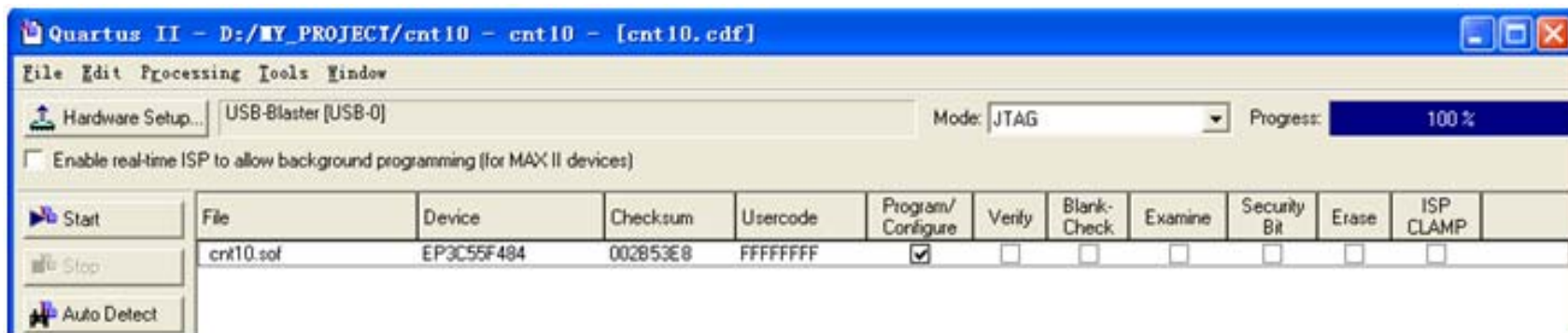


图 4-17 选择 JTAG 编程模式

(2) 设置编程器。

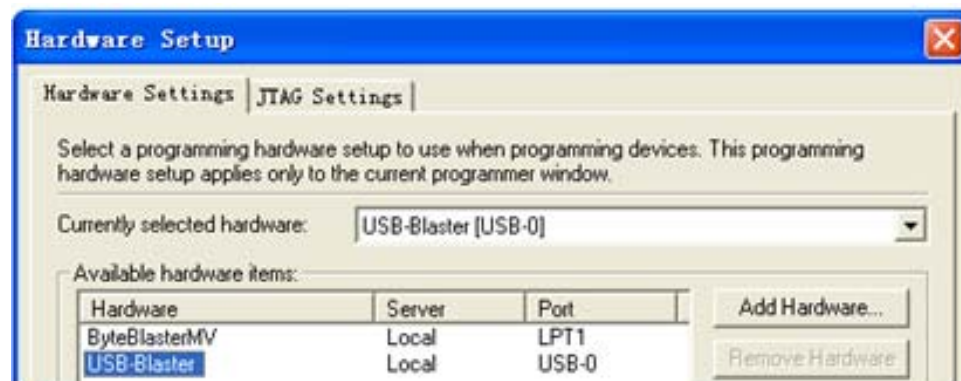


图 4-18 加入编程下载方式

(3) 硬件测试。

## 4.2 引脚锁定与硬件测试

### 4.2.3 JTAG间接编程模式

1. 将SOF文件转化为JTAG间接配置文件。

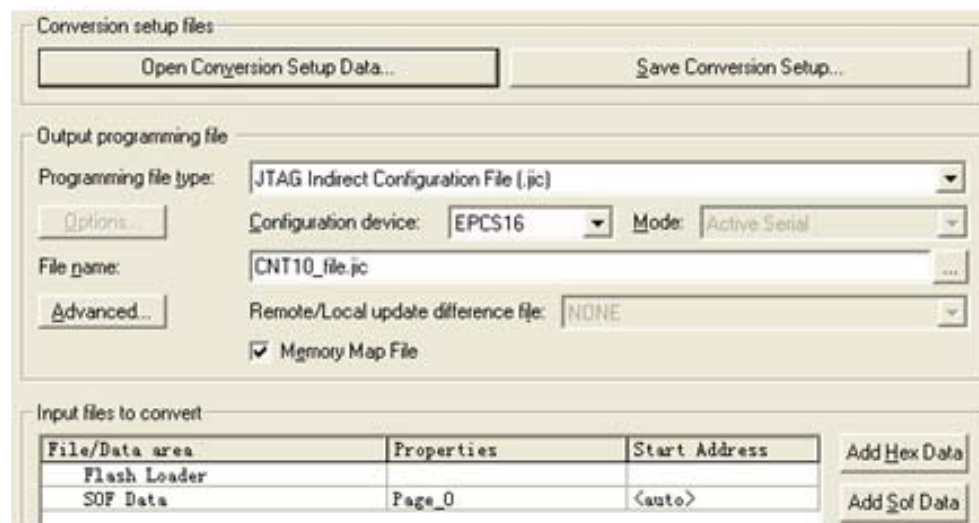


图 4-19 设定 JTAG 间接编程文件



## 4.2 引脚锁定与硬件测试

### 4.2.3 JTAG间接编程模式

1. 将SOF文件转化为JTAG间接配置文件。



图 4-20 选择目标器件 EP3C55



图 4-21 加入 SOF 文件，并选择压缩模式

## 4.2 引脚锁定与硬件测试

### 4.2.3 JTAG间接编程模式

2. 下载JTAG间接配置文件。

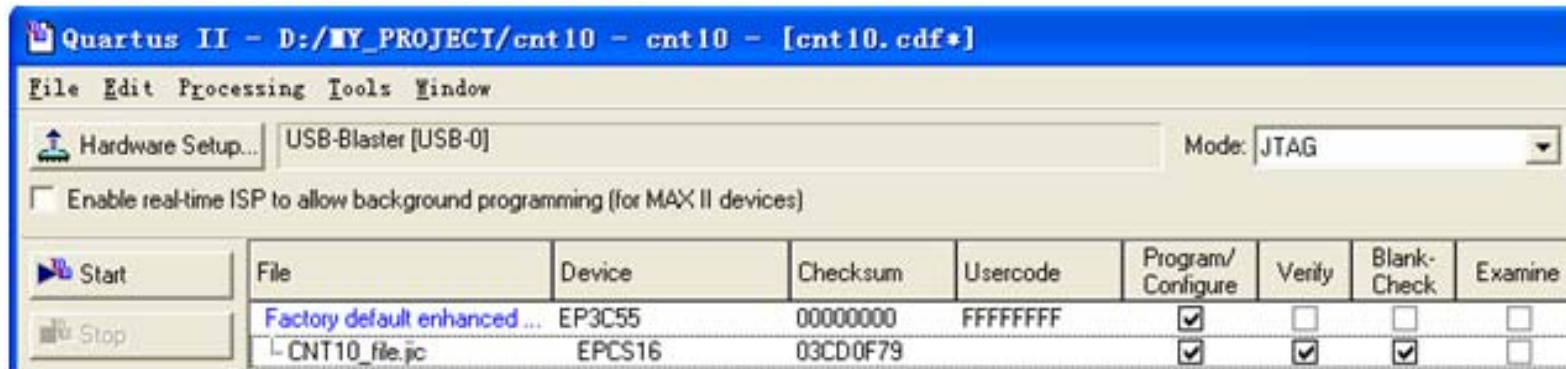


图 4-22 用 JTAG 模式对配置器件 EPCS16 进行编程

## 4.2 引脚锁定与硬件测试

### 4.2.4 USB-Blaster驱动程序安装方法



图 4-17 选择 JTAG 编程模式

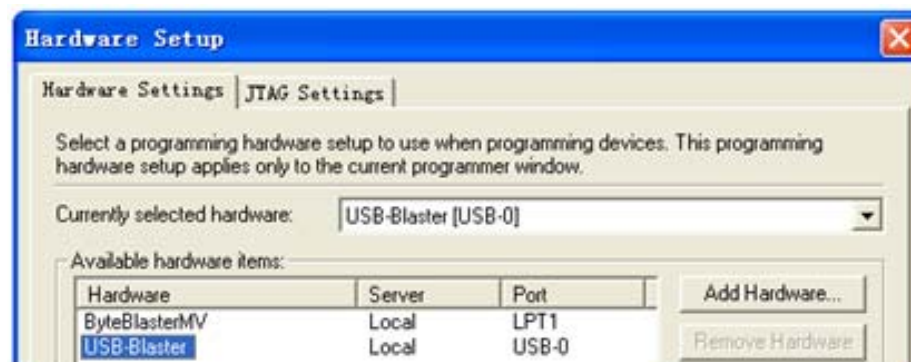


图 4-18 加入编程下载方式

## 4.3 电路原理图设计流程

### 4.3.1 用原理图输入方式设计半加器

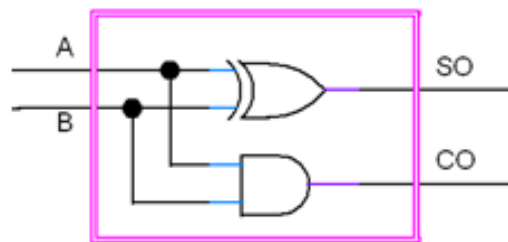


图 4-23 半加器的电路结构

A	B	SO	CO
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

图 4-24 半加器的真值表

$$SO = A \oplus B ; CO = A \cdot B$$

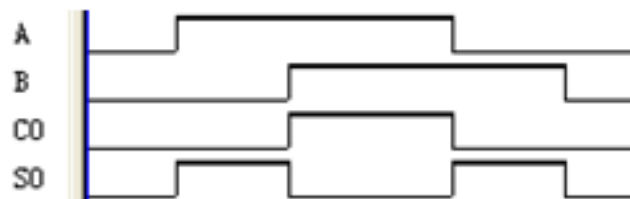


图 4-25 半加器的仿真功能波形图

## 4.3 电路原理图设计流程

### 4.3.1 用原理图输入方式设计半加器

- (1) 打开原理图编辑窗。
- (2) 建立一个初始原理图。



图 4-26 选择打开元件输入窗

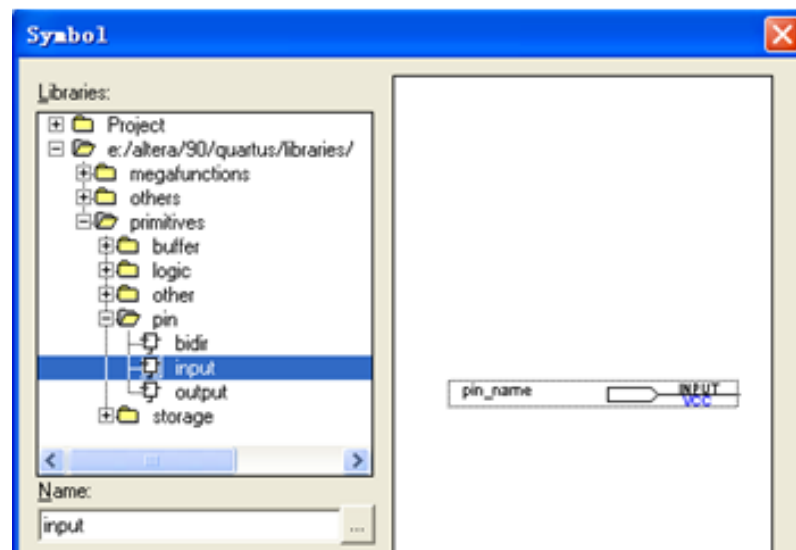


图 4-27 在元件输入对话框输入引脚

## 4.3 电路原理图设计流程

### 4.3.1 用原理图输入方式设计半加器

#### (3) 原理图文件存盘。

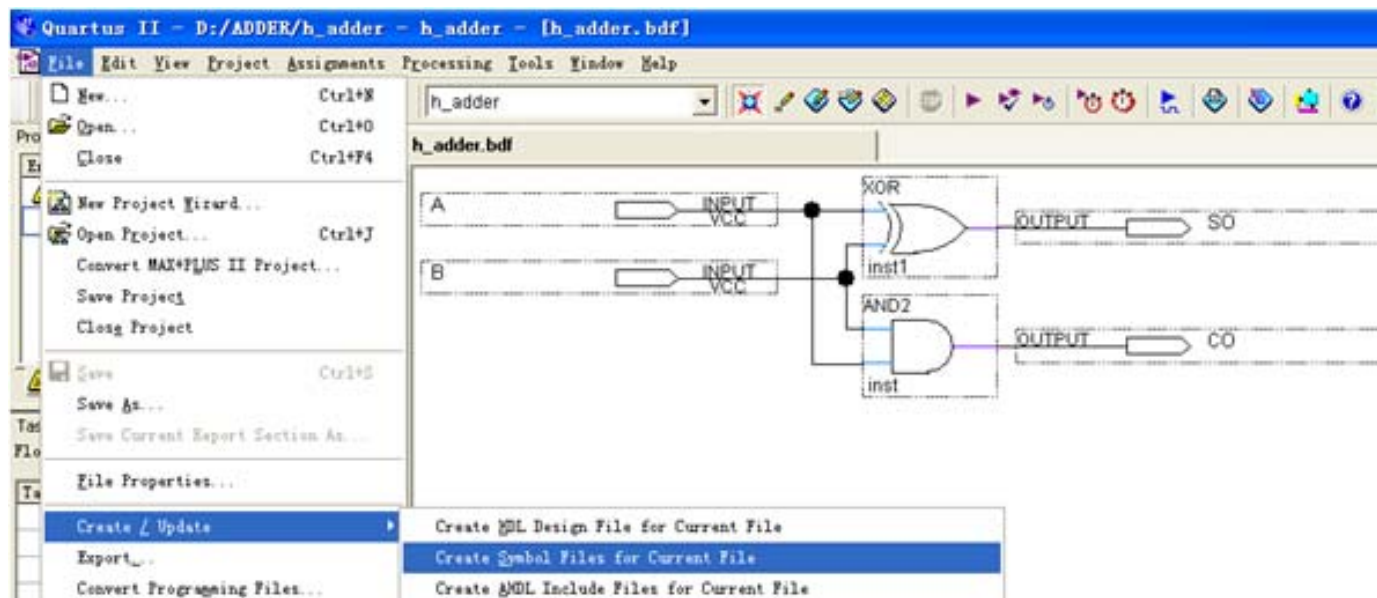


图 4-28 完成设计，并将半加器封装成一个元件，以便在更高层设计中调用

## 4.3 电路原理图设计流程

### 4.3.1 用原理图输入方式设计半加器

(4) 创建原理图文件为顶层设计的工程。

(5) 绘制半加器原理图。

(6) 测试半加器。

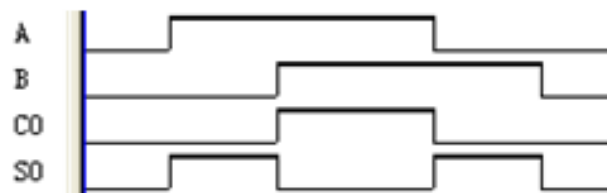


图 4-25 半加器的仿真功能波形图

## 4.3 电路原理图设计流程

### 4.3.2 完成全加器顶层设计

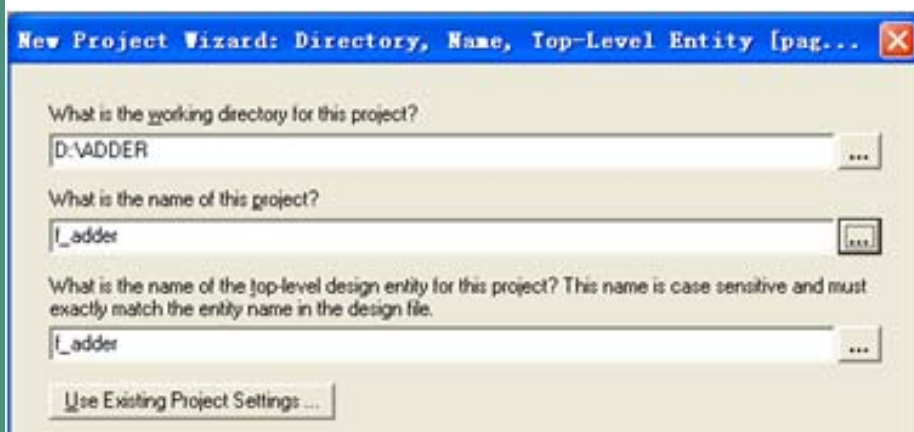


图 4-29 全加器 f\_adder.bdf 工程设置

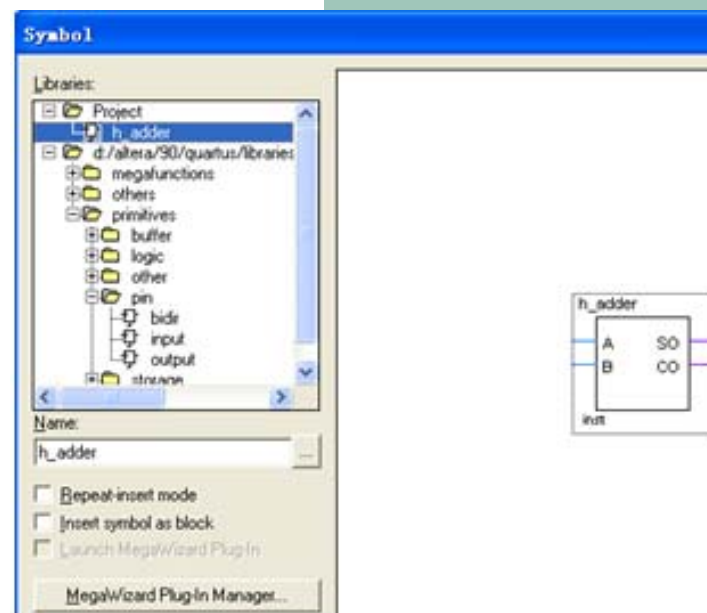


图 4-30 在 f\_adder 工程下加入半加器

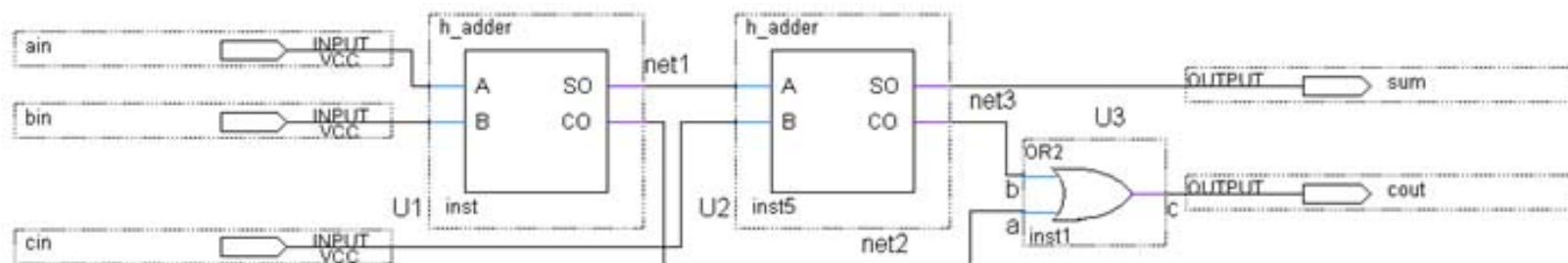


图 4-31 全加器 f\_adder 电路图



## 4.3 电路原理图设计流程

### 4.3.3 对全加器进行时序仿真和硬件测试

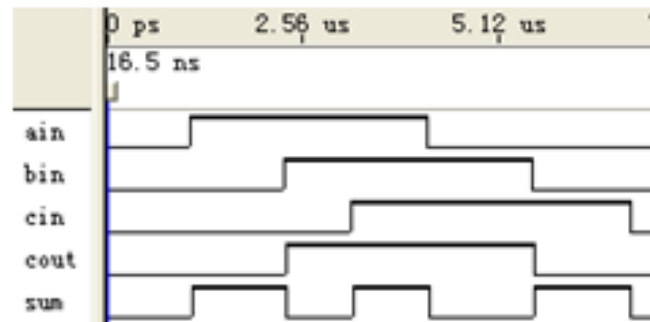


图 4-32 全加器的仿真波形

## 4.4 利用属性表述实现引脚锁定

### 【例 4-1】

```
ARCHITECTURE ONE 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 "AA3" ;
    attribute chip_pin of EN  : signal is "V3" ;
    attribute chip_pin of DATA : signal is "N22,M22,L21,L22" ;
    ...
BEGIN
```

Edit -> Insert Template -> VHDL -> Synthesis Attributes;

## 4.5 宏模块逻辑功能查询

```
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 4-33 74138 真值表

## 4.6 SignalTap II的使用方法

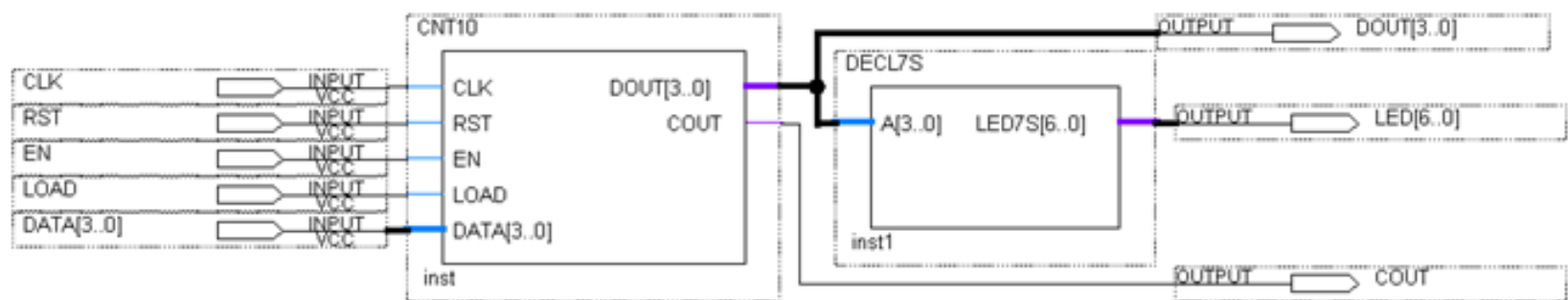


图 4-34 十进制计数器设计示例电路

## 4.6 SignalTap II的使用方法

1. 打开SignalTap II编辑窗口
2. 调入待测信号

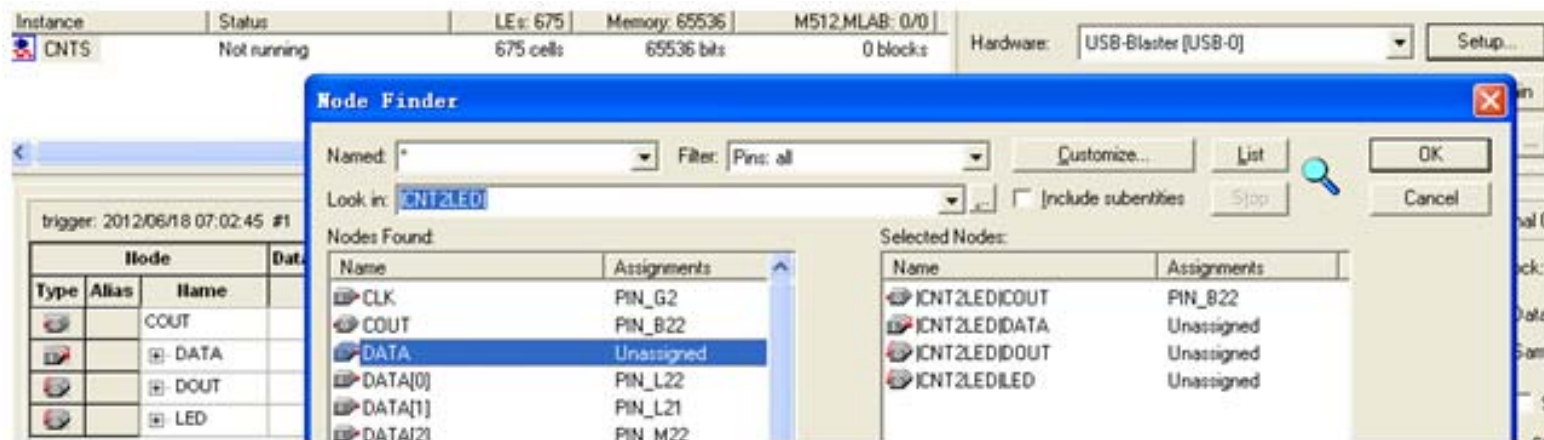


图 4-35 输入逻辑分析仪测试信号

# 4.6 SignalTap II的使用方法

## 3. SignalTap II参数设置



图 4-36 SignalTap II 编辑窗口



图 4-37 设置 EN 为触发信号

## 4.6 SignalTap II的使用方法

### 4. 文件存盘



图 4-38 选择或删除 SignalTap II 文件加入综合编译

### 5. 编译下载

# 4.6 SignalTap II的使用方法

## 6. 启动SignalTap II进行采样与分析

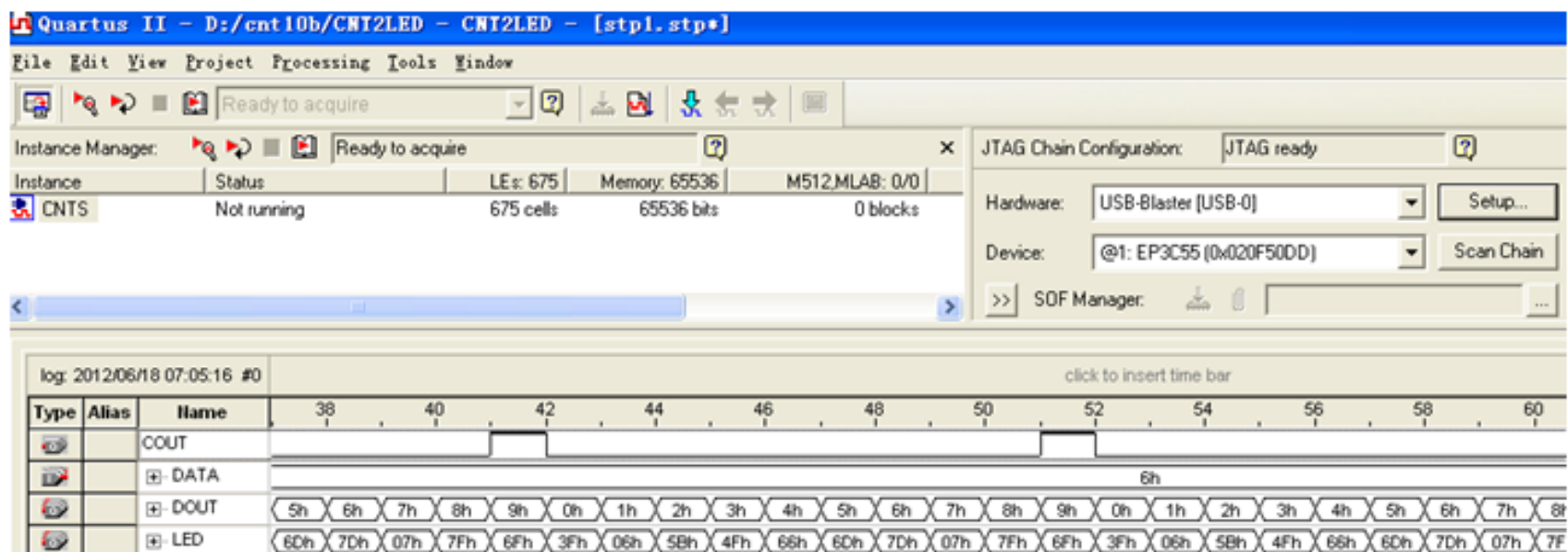


图 4-39 SignalTap II 实时数据采集显示界面



## 4.6 SignalTap II的使用方法

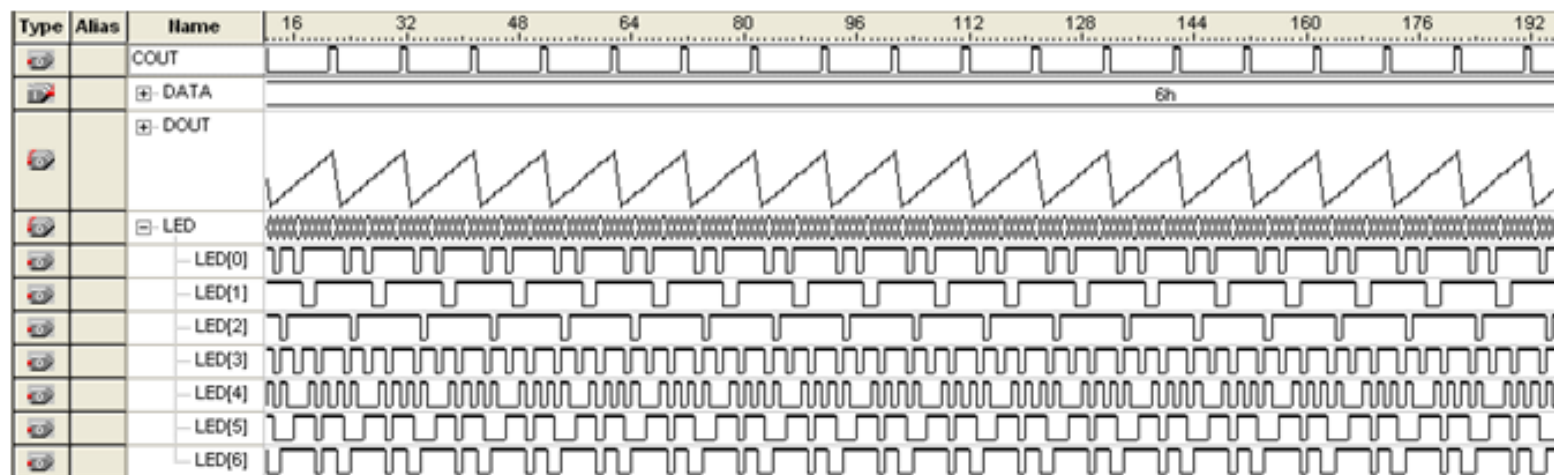


图 4-40 SignalTap II 数据窗口显示对硬件系统实时测试采样后的信号波形

```
ARCHITECTURE ONE OF xxx IS  
attribute chip_pin of CLK0 : signal is "G21"; -- 逻辑分析仪采样时钟
```

### 7. SignalTap II的其他设置和控制方法

## 4.7 编辑SignalTap II的触发信号



图 4-36 SignalTap II 编辑窗口

# 实验

---

4-1. 多路选择器设计实验

4-2. 十六进制7段数码显示译码器设计

【例 4-2】

```

LIBRARY IEEE ;
USE IEEE.STD_LOGIC_1164.ALL ;
ENTITY DECL7S IS
    PORT ( A : IN STD_LOGIC_VECTOR(3 DOWNTO 0);
          LED7S : OUT STD_LOGIC_VECTOR(6 DOWNTO 0));
END ;
ARCHITECTURE one OF DECL7S IS
BEGIN
    PROCESS( A ) BEGIN
        CASE A IS
            WHEN "0000" => LED7S <= "0111111" ;
            WHEN "0001" => LED7S <= "0000110" ;
            WHEN "0010" => LED7S <= "1011011" ;
            WHEN "0011" => LED7S <= "1001111" ;
            WHEN "0100" => LED7S <= "1100110" ;
            WHEN "0101" => LED7S <= "1101101" ;
            WHEN "0110" => LED7S <= "1111101" ;
            WHEN "0111" => LED7S <= "0000111" ;
            WHEN "1000" => LED7S <= "1111111" ;
            WHEN "1001" => LED7S <= "1101111" ;
            WHEN "1010" => LED7S <= "1110111" ;
            WHEN "1011" => LED7S <= "1111100" ;
            WHEN "1100" => LED7S <= "0111001" ;
            WHEN "1101" => LED7S <= "1011110" ;
            WHEN "1110" => LED7S <= "1111001" ;
            WHEN "1111" => LED7S <= "1110001" ;
            WHEN OTHERS => NULL ;
        END CASE ;
    END PROCESS ;
END ;

```

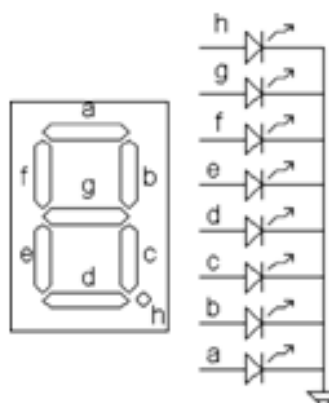


图 4-41 共阴数码管

表 4-1 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

---

**4-3. 计数器设计实验**

**4-4. 硬件消抖动电路设计**

### 【例 4-3】

```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
USE IEEE.STD_LOGIC_UNSIGNED.ALL;
ENTITY ERZP IS
    PORT ( CLK,KIN : IN STD_LOGIC; --工作时钟和输入信号
          KOUT : OUT STD_LOGIC ); --消抖动后的输出信号
END;
ARCHITECTURE BHV OF ERZP IS
    SIGNAL KL,KH : STD_LOGIC_VECTOR (3 DOWNTO 0);
BEGIN
PROCESS (CLK,KIN,KL,KH ) BEGIN
    IF CLK'EVENT AND CLK = '1' THEN
        IF (KIN='0') THEN KL<=KL+1; --对键输入的低电平脉宽计数
        ELSE KL<="0000"; END IF; --若出现高电平，则计数器清 0
        IF (KIN='1') THEN KH<=KH+1; --同时对键输入的高电平脉宽计数
        ELSE KH<="0000"; END IF; --若出现高电平，则计数器清 0
        IF (KH>"1100") THEN KOUT<='1';--对高电平脉宽计数一旦大于 12，则输出 1
        ELSIF (KL>"0111") THEN KOUT<='0';--对低电平脉宽计数若大于 7，则输出 0
        END IF; END IF;
    END PROCESS;
END;
```



图 4-42 例 4-3 消抖动电路仿真波形

# 实验

## 4-5. 应用宏模块设计数字频率计

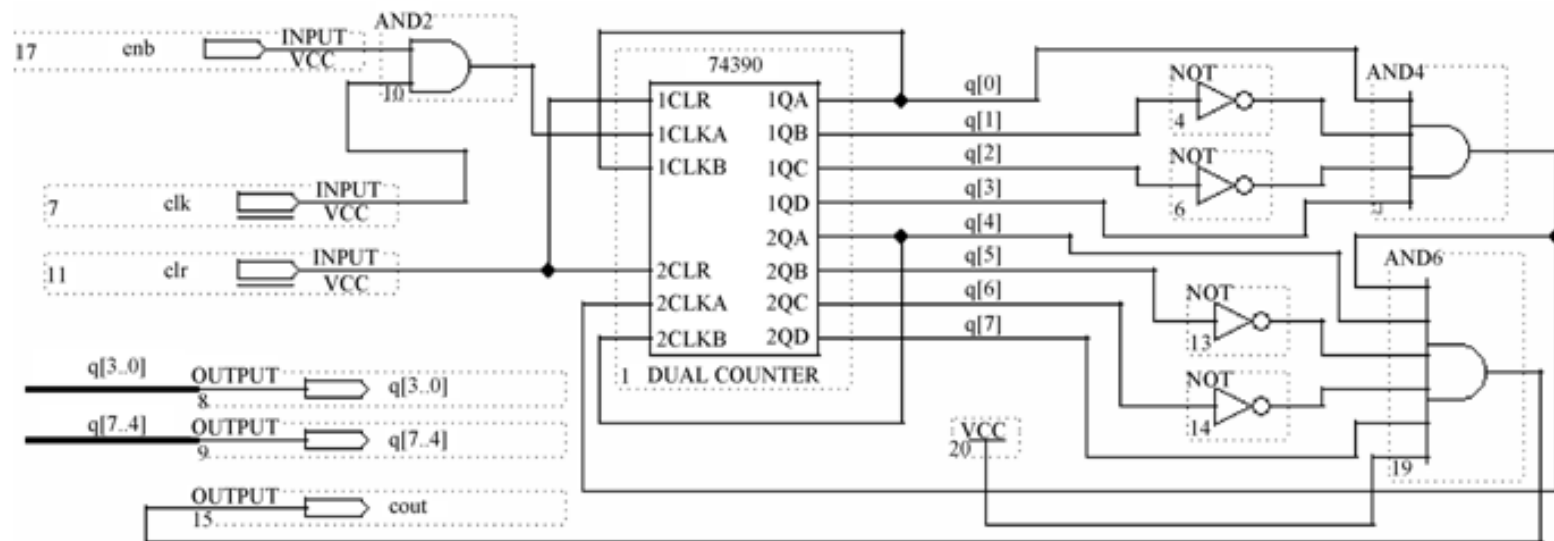


图 4.43 含有时钟使能的 2 位十进制计数器

# 实验

## 4-5. 应用宏模块设计数字频率计

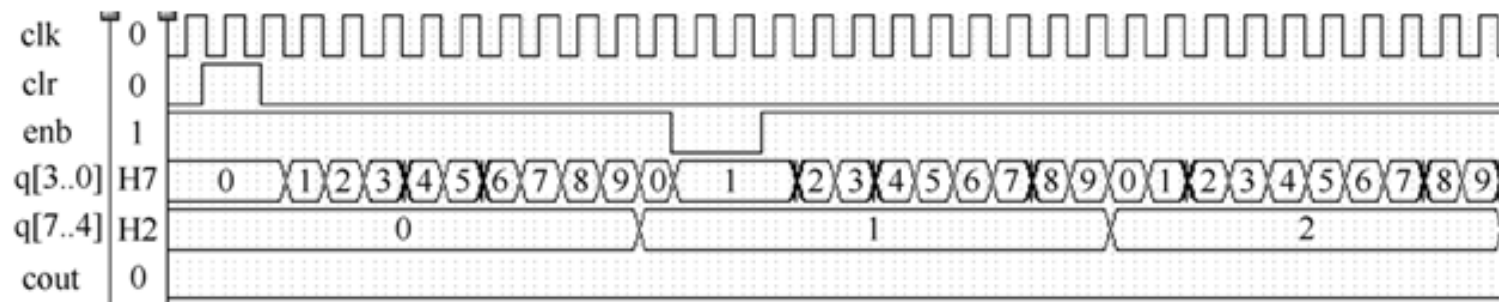


图 4-44 两位十进制计数器工作波形



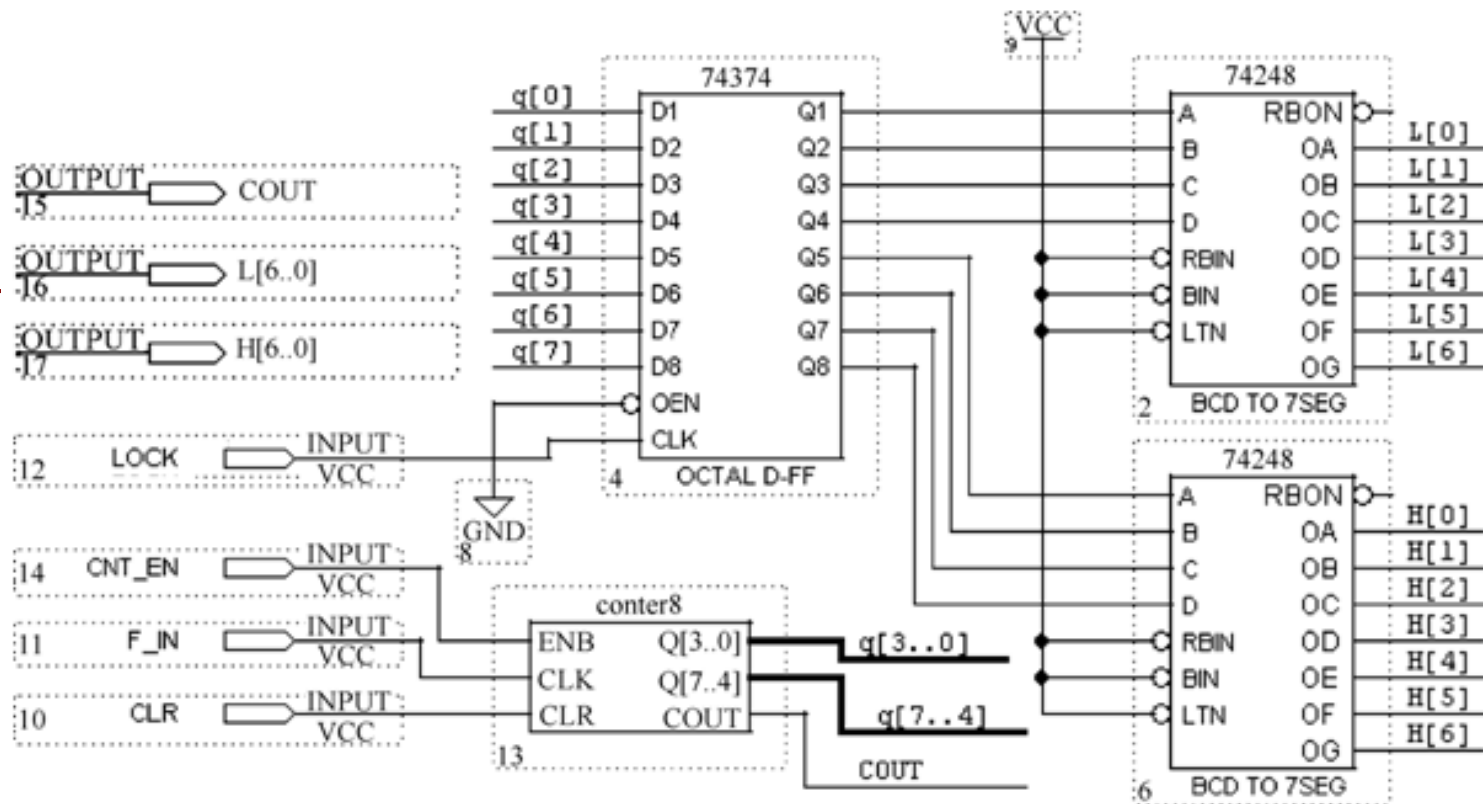


图 4-45 2 位十进制频率计顶层设计原理图文件

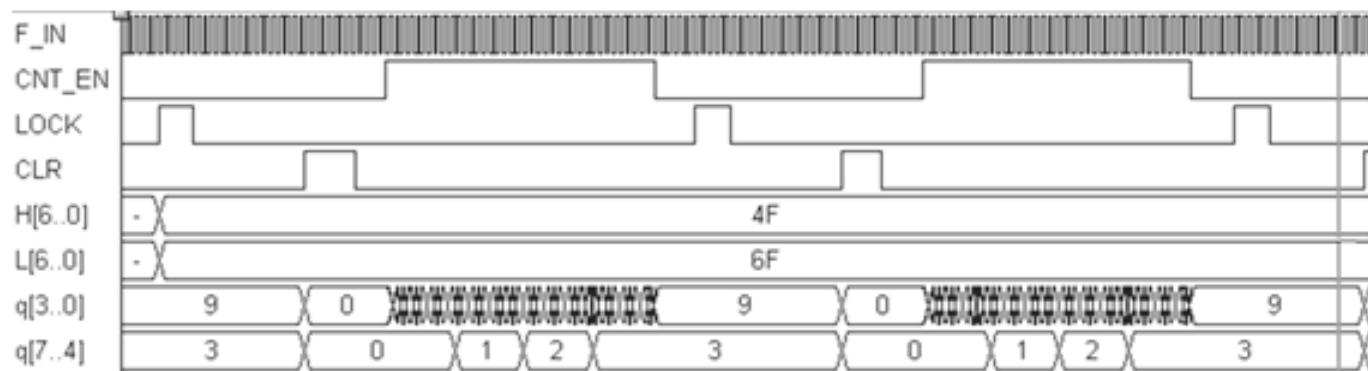


图 4-46 两位十进制频率计测频仿真波形

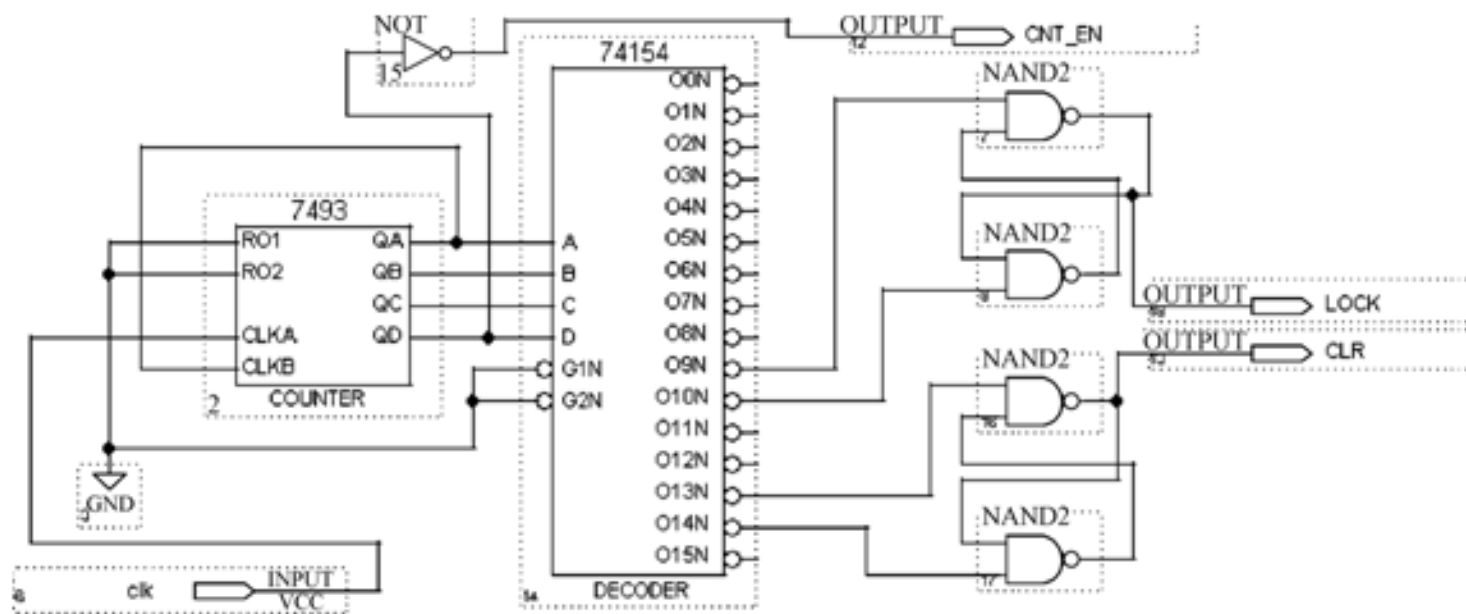


图 4-47 测频时序控制电路

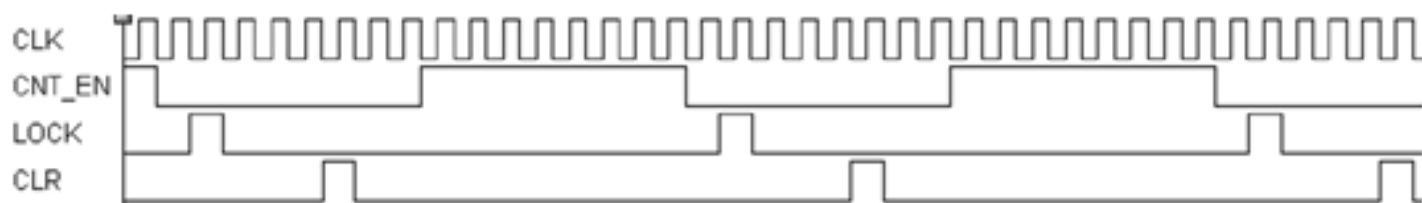


图 4-48 测频时序控制电路工作波形

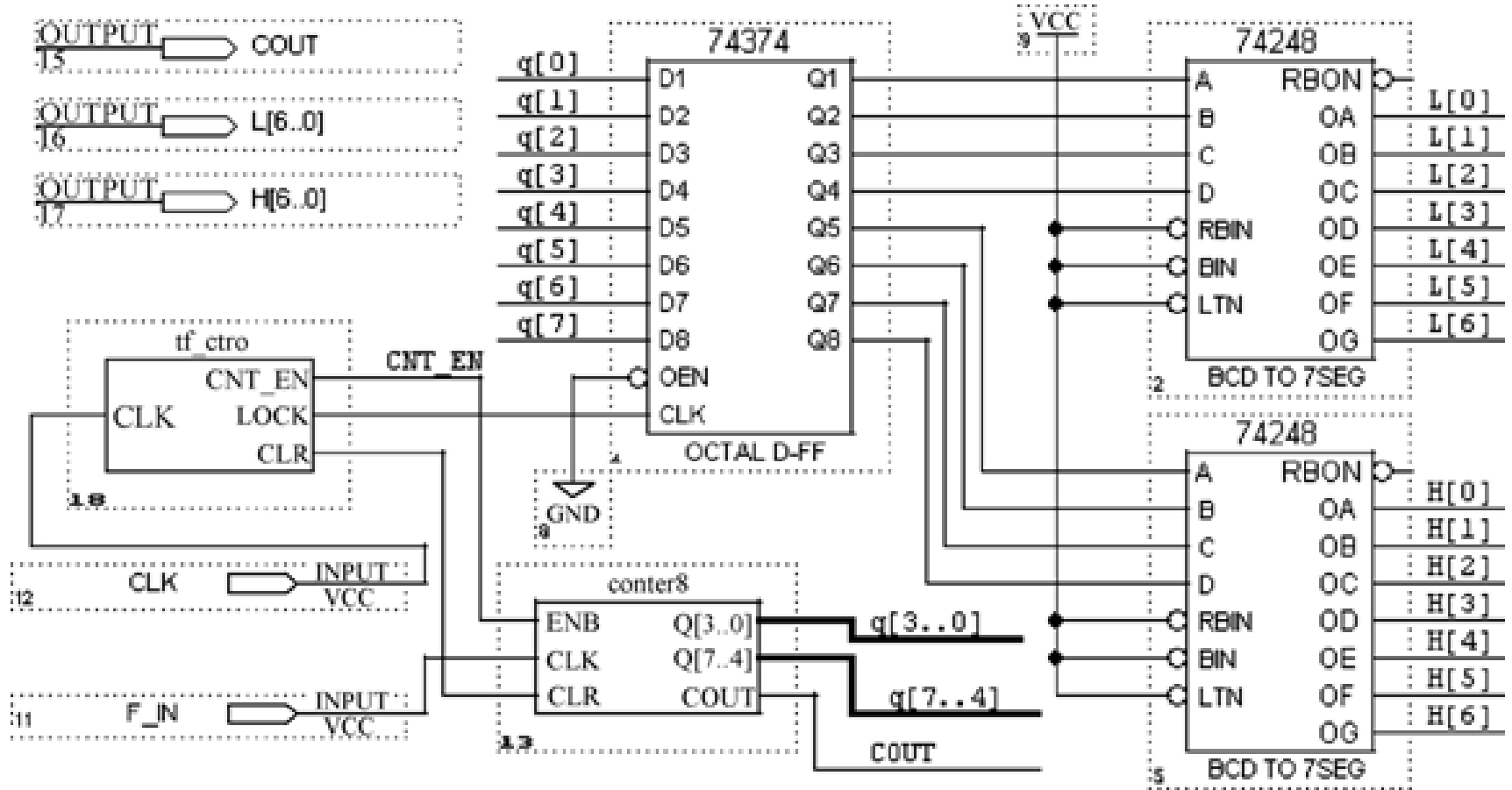


图 4-49 频率计顶层电路原理图

# 实验

## 4-5. 应用宏模块设计数字频率计

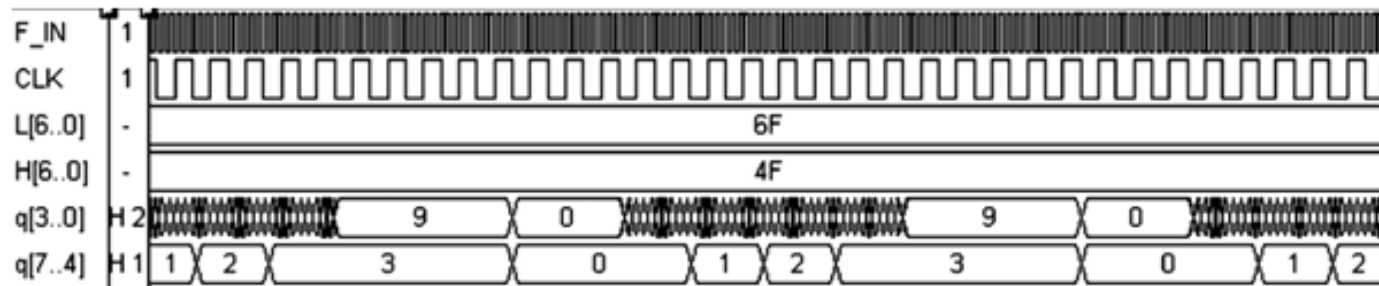


图 4-50 频率计工作时序波形

# 实验

## 4-6 数码扫描显示电路设计

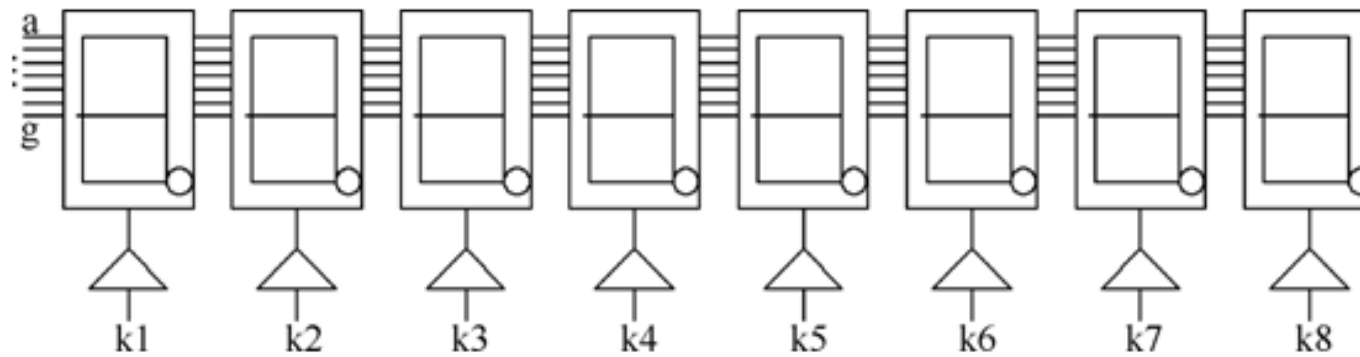


图 4-51 8 位数码扫描显示电路

# 实验

## 4-7 串行静态显示控制电路设计

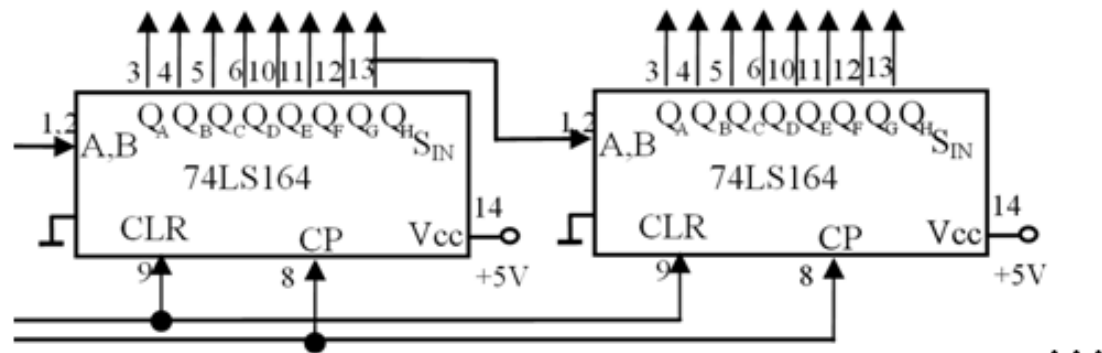


图 4-52 串/并转换数码管静态显示电路

## 4-8 不同类型的移位寄存器设计实验