

# 第6章



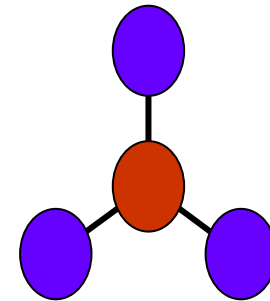
## 组合电路时序分析与自动化设计

# 6.1 传统数字技术存在的问题

---



- (1) 低速。
- (2) 设计规模小。
- (3) 分析技术无法适应需要。
- (4) 效率低成本高。
- (5) 可靠性低。
- (6) 体积大功耗大。
- (7) 功能有限。
- (8) 无法实现升级。
- (9) 知识产权不易保护。



# 6.2 数字系统自动设计流程

## 6.2.1 设计输入

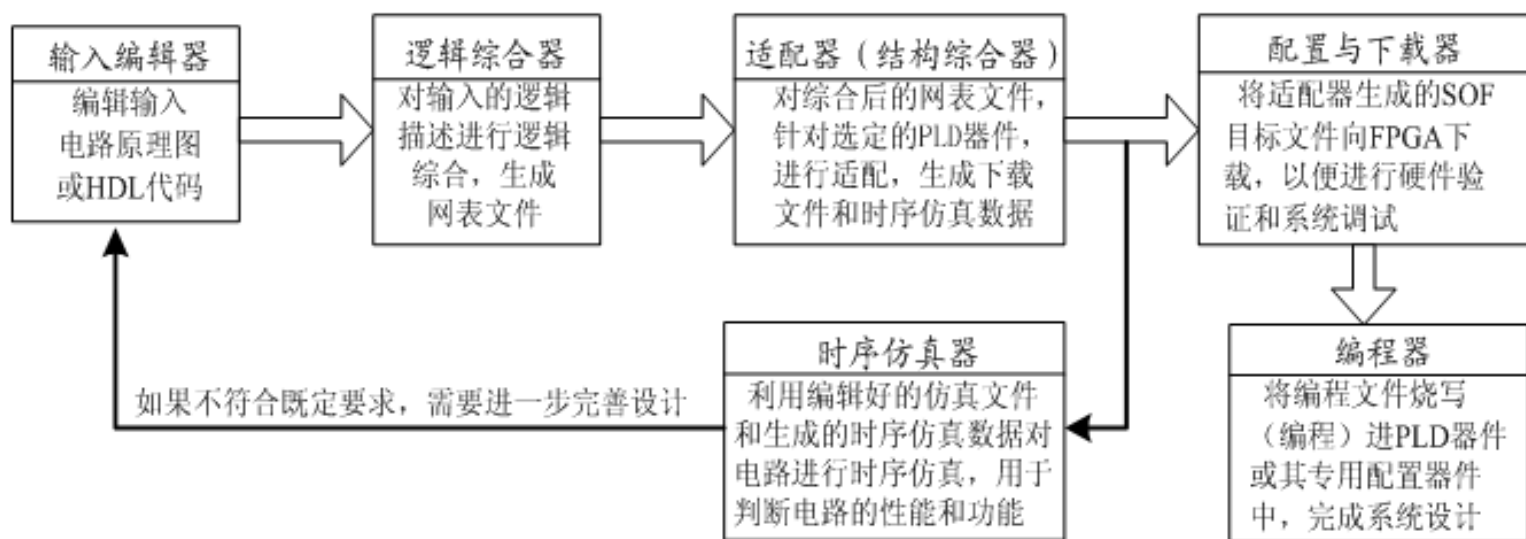


图 6-1 针对 PLD 的数字系统开发流程

1. 图形输入

2. HDL文本输入

# 6.2 数字系统自动设计流程

---

6.2.2 硬件描述语言

6.2.3 综合

6.2.4 适配

6.2.5 仿真

6.2.6 硬件测试

# 6.3 原理图输入法逻辑电路设计

## 6.3.1 原理图编辑输入方法

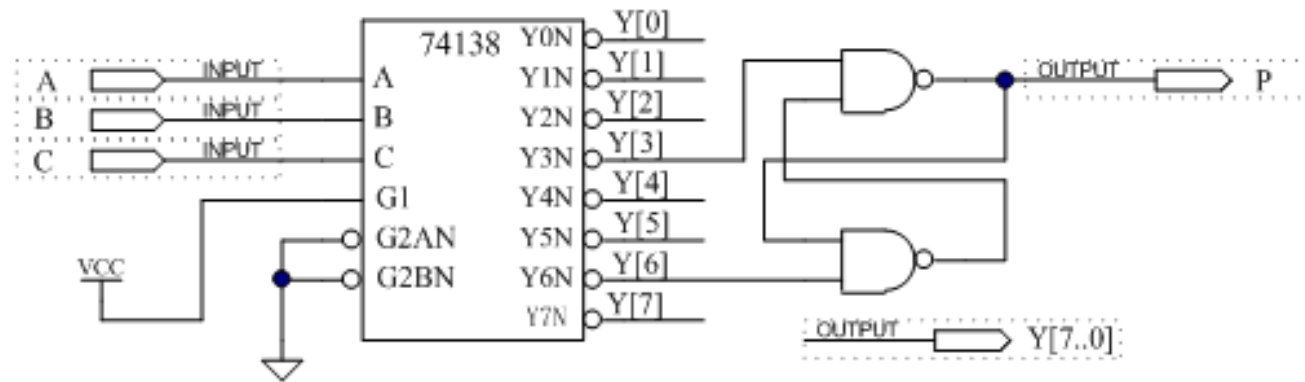


图 6-2 脉冲发生器示例电路图

# 6.3 原理图输入法逻辑电路设计

## 6.3.1 原理图编辑输入方法

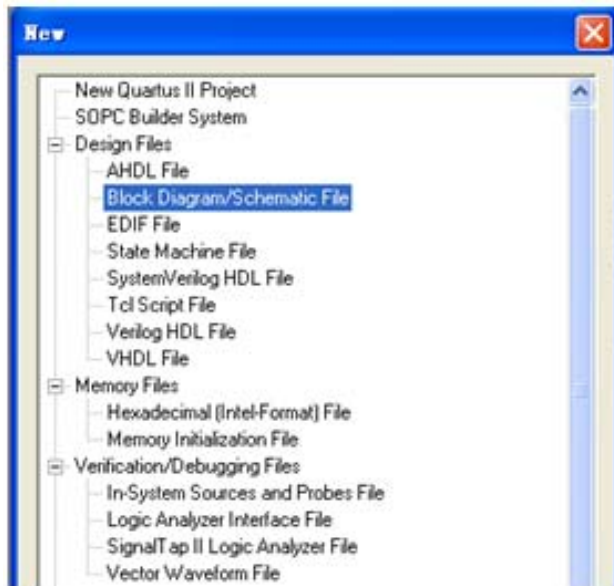


图 6-3 选择原理图编辑文件类型

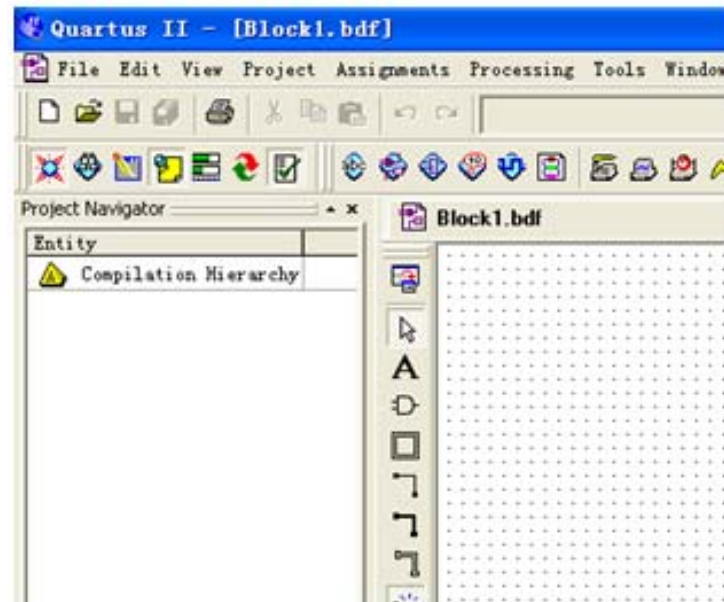


图 6-4 打开原理图编辑窗

# 6.3 原理图输入法逻辑电路设计

## 6.3.2 创建工程



图 6-5 利用“New Project Wizard”创建工程 top

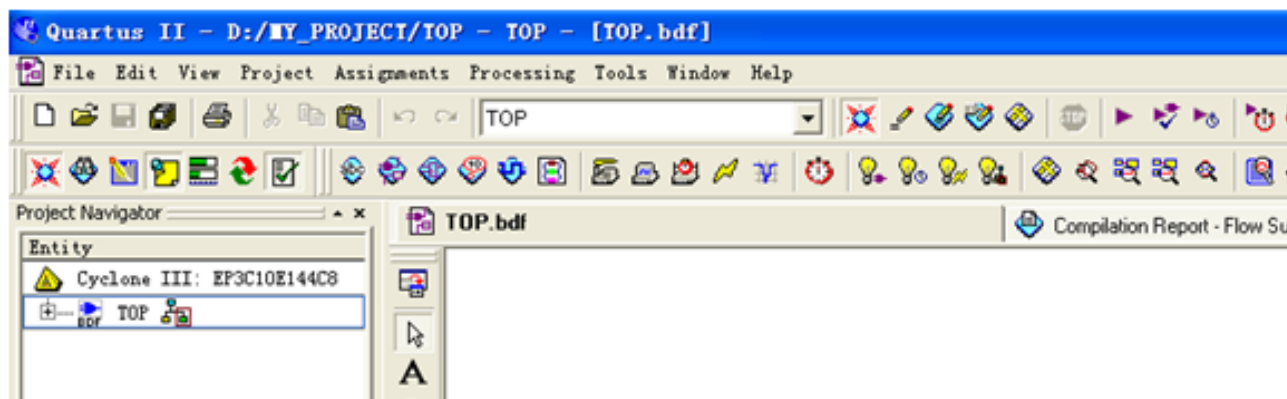


图 6-6 top 工程管理窗

# 6.3 原理图输入法逻辑电路设计

## 6.3.2 创建工程

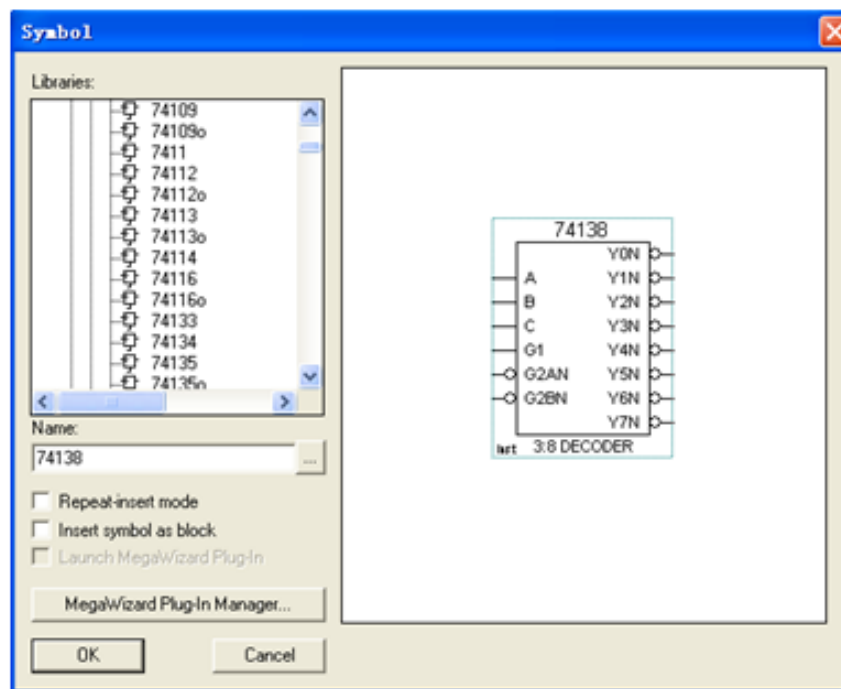


图 6-7 在元件调用对话框调出宏功能元件 74138

## 6.3.3 功能分析



# 6.3 原理图输入法逻辑电路设计

## 6.3.4 编译前设置

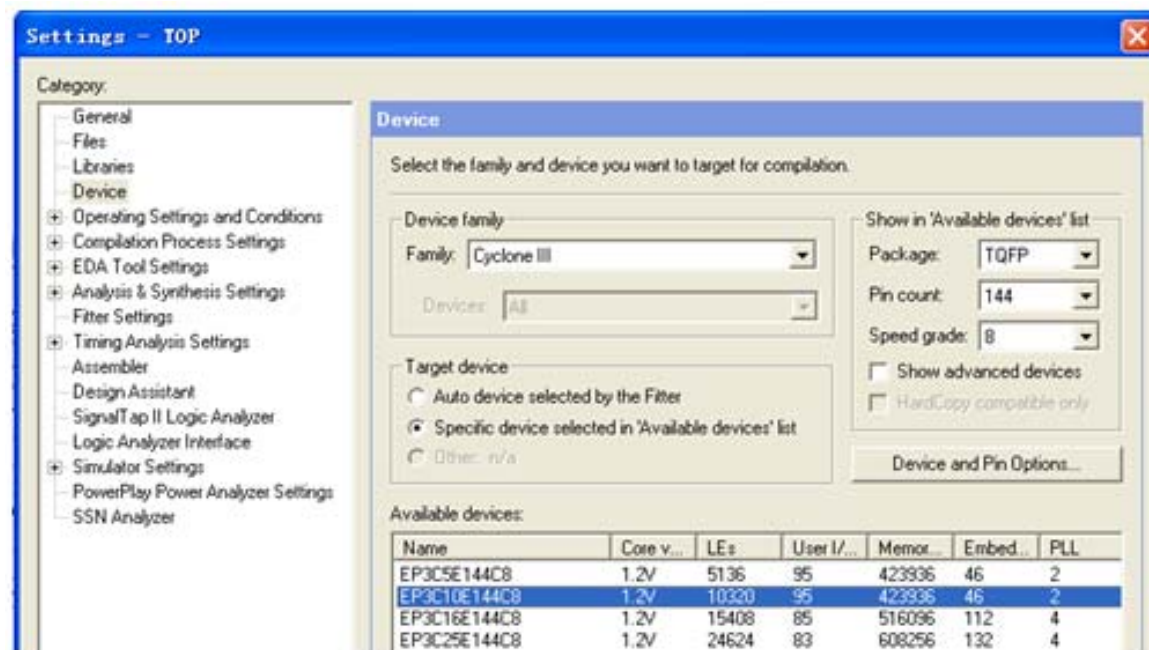


图 6-8 由 Settings 对话框选择目标器件 EP3C10E144C8

# 6.3 原理图输入法逻辑电路设计

## 6.3.4 编译前设置

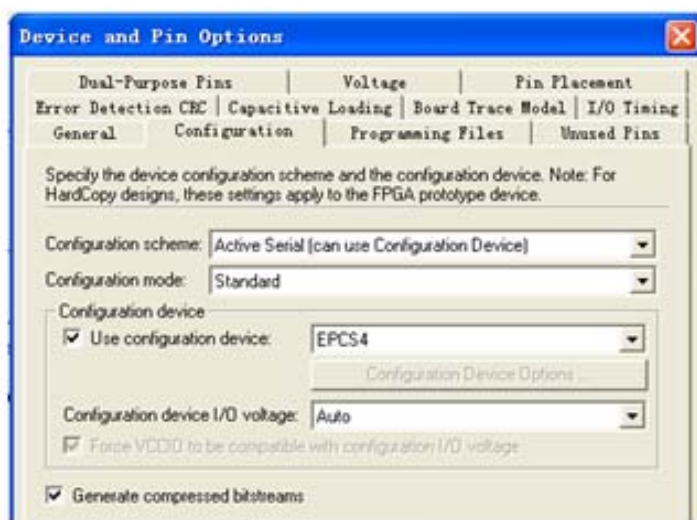


图 6-9 选择配置器件型号和压缩方式

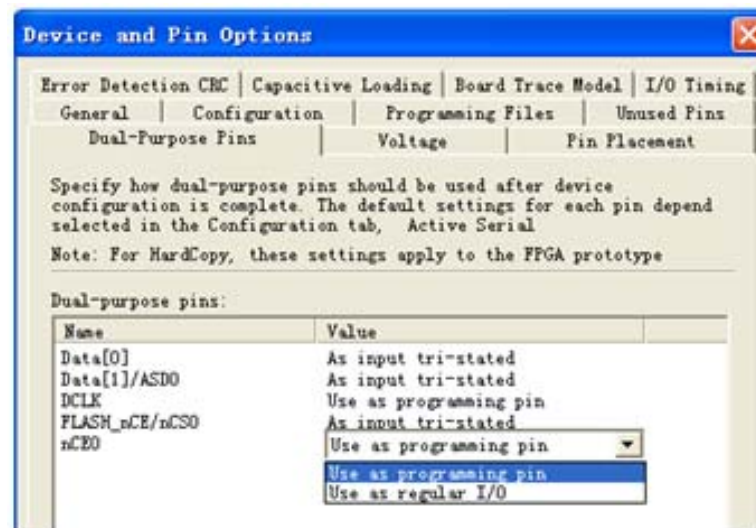


图 6-10 指定 nCEO 脚为普通 I/O 端口

# 6.3 原理图输入法逻辑电路设计

## 6.3.5 全程编译

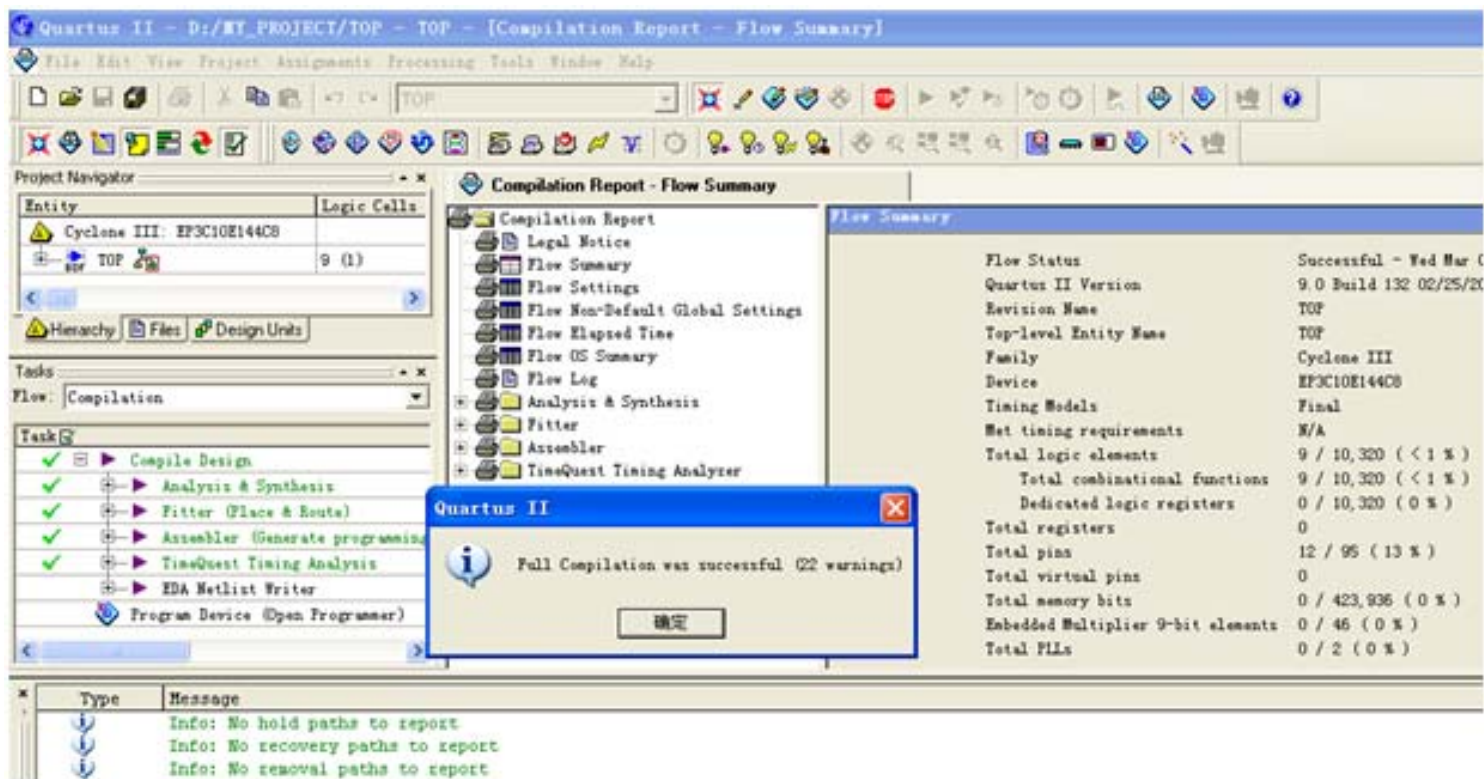


图 6-11 全程编译后工程管理窗的情况

# 6.3 原理图输入法逻辑电路设计

## 6.3.6 逻辑功能测试

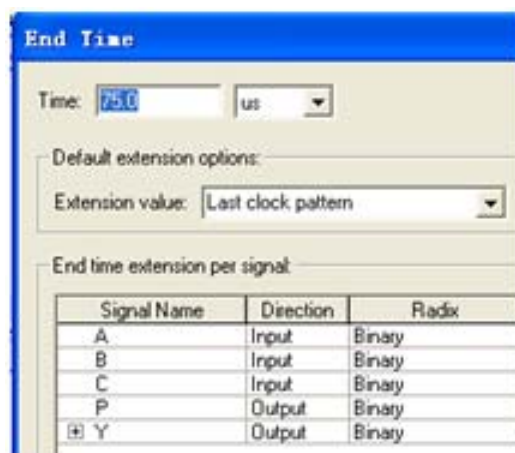


图 6-12 设置仿真时间长度

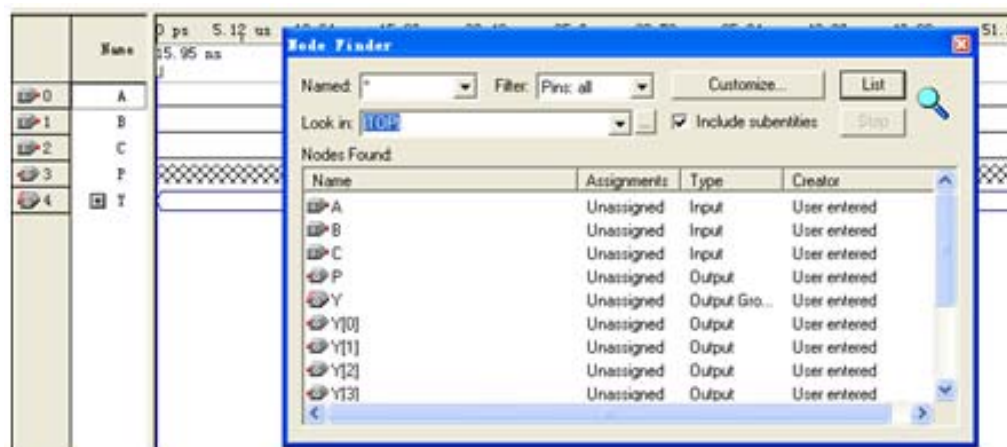


图 6-13 从 Node Finder 窗向波形编辑器拖入信号节点

# 6.3 原理图输入法逻辑电路设计

## 6.3.6 逻辑功能测试

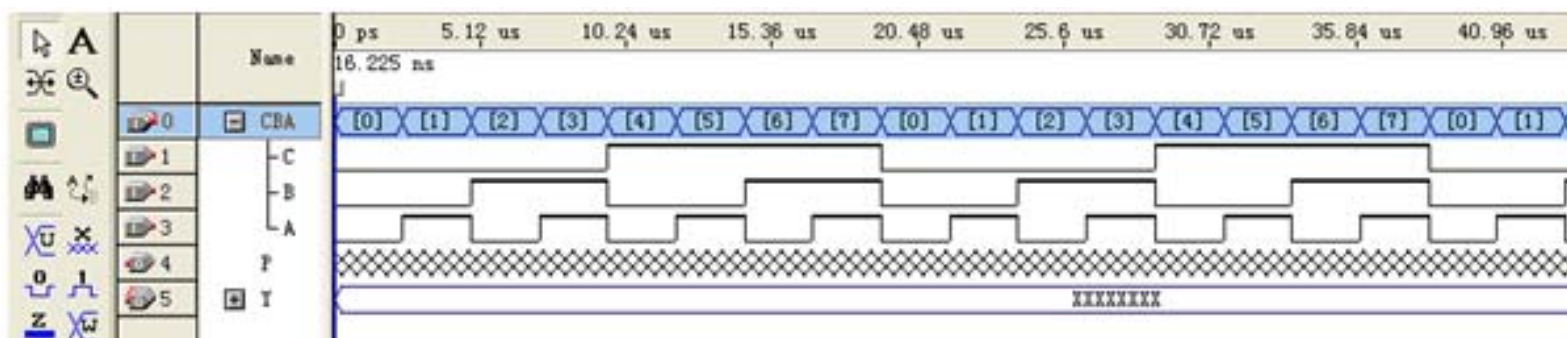


图 6-14 设置好的仿真文件激励波形图

# 6.3 原理图输入法逻辑电路设计

## 6.3.6 逻辑功能测试

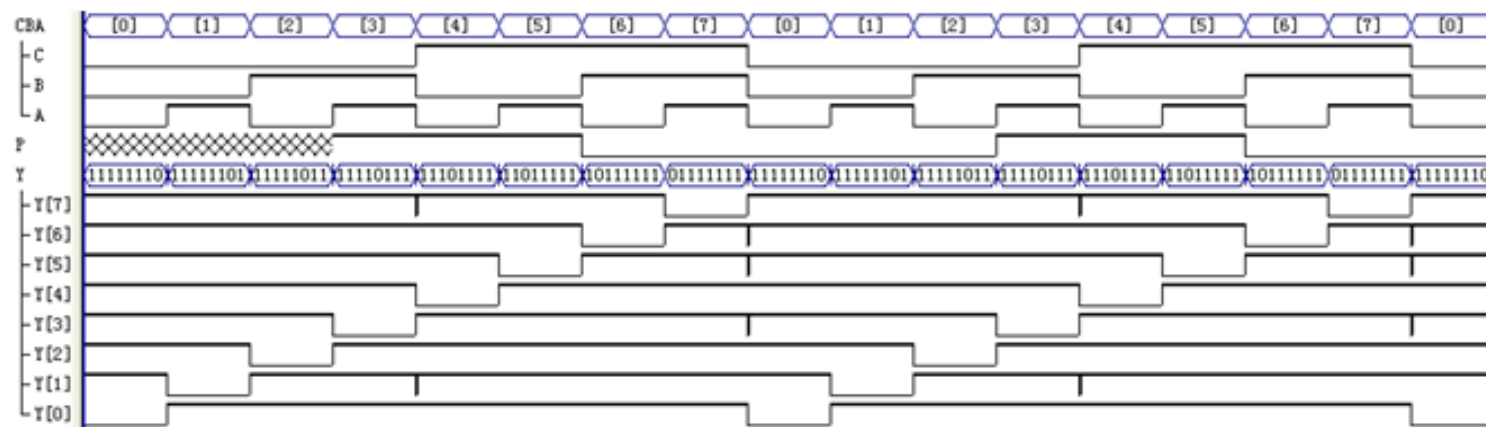


图 6-15 图 6-2 电路的仿真波形输出

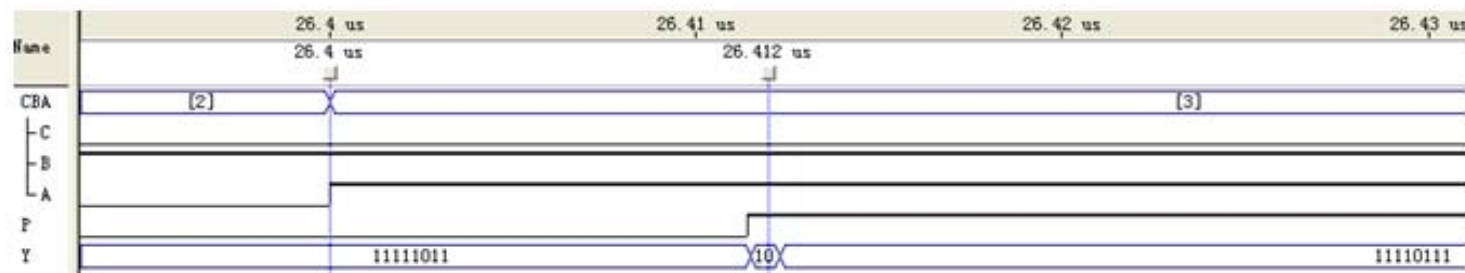


图 6-16 输入 A 与输出 P 的延时波形显示

# 6.4 硬件测试

## 6.4.1 引脚锁定

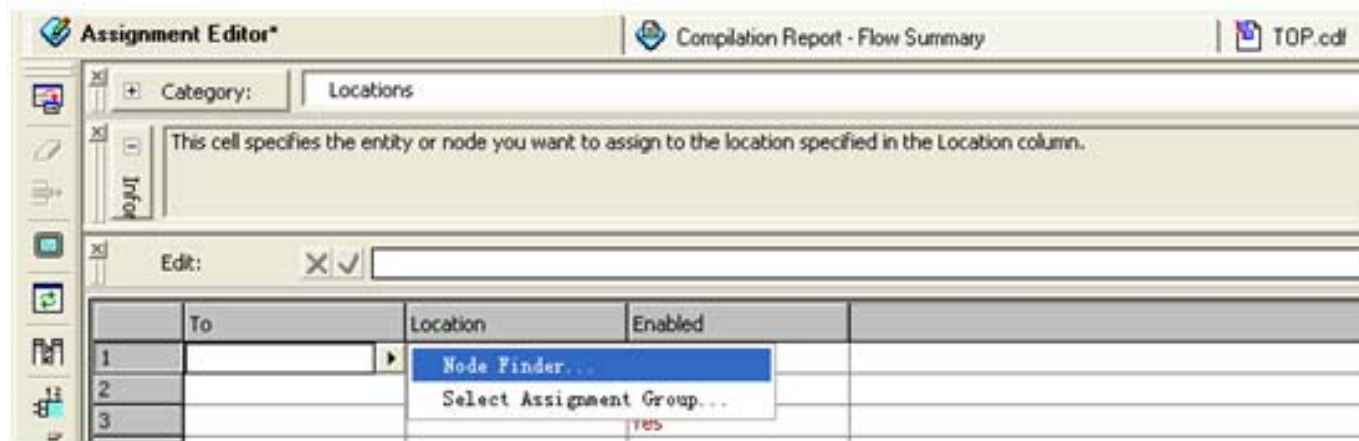


图 6-17 利用 Assignment Editor 编辑器锁定 FPGA 引脚

# 6.4 硬件测试

## 6.4.1 引脚锁定

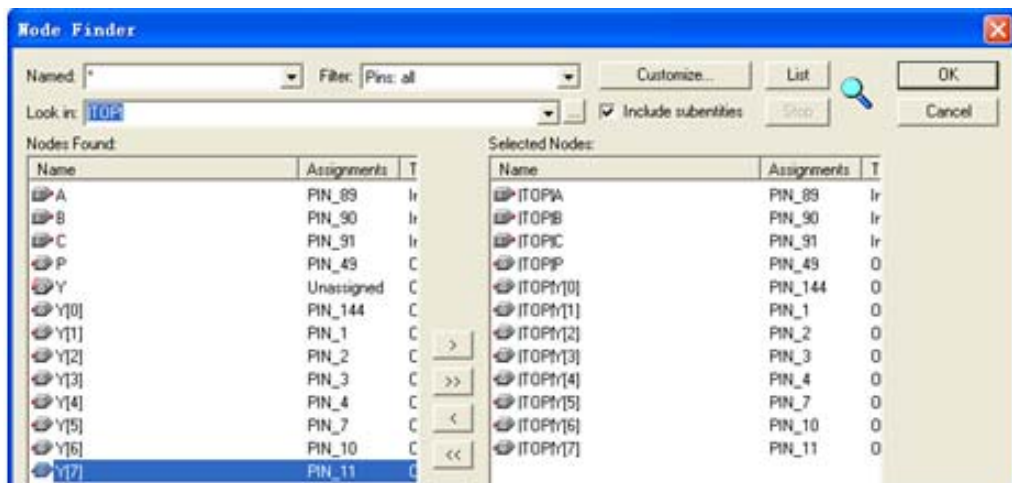


图 6-18 选择需要锁定的引脚信号

	To	Location
1	LED A	PIN_89
2	LED B	PIN_90
3	LED C	PIN_91
4	P	PIN_49
5	Y[0]	PIN_144
6	Y[1]	PIN_1
7	Y[2]	PIN_2
8	Y[3]	PIN_3
9	Y[4]	PIN_4
10	Y[5]	PIN_7
11	Y[6]	PIN_10
12	Y[7]	PIN_11
13	<<new>>	

图 6-19 引脚锁定对话框



# 6.4 硬件测试

## 6.4.2 对FPGA编程配置

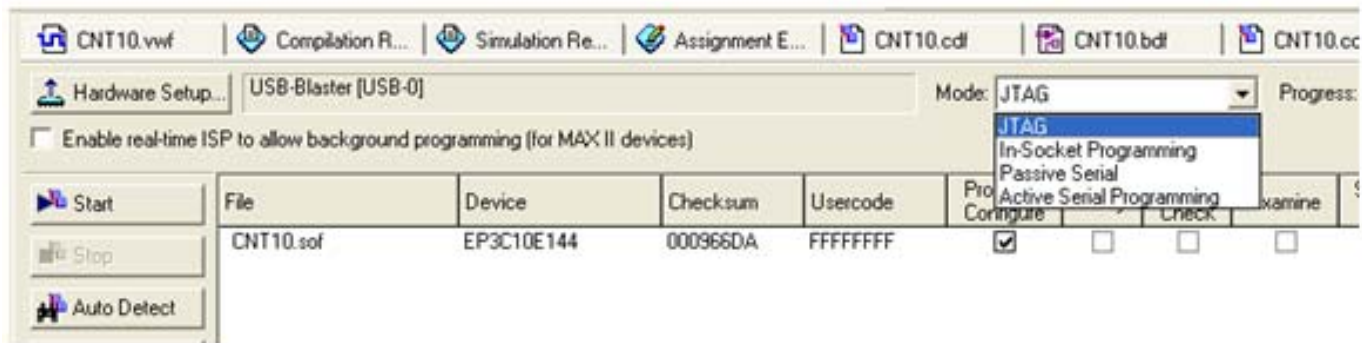


图 6-20 选择编程下载文件和下载模式



图 6-21 加入编程下载方式

# 6.4 硬件测试

## 6.4.3 对FPGA配置器件编程

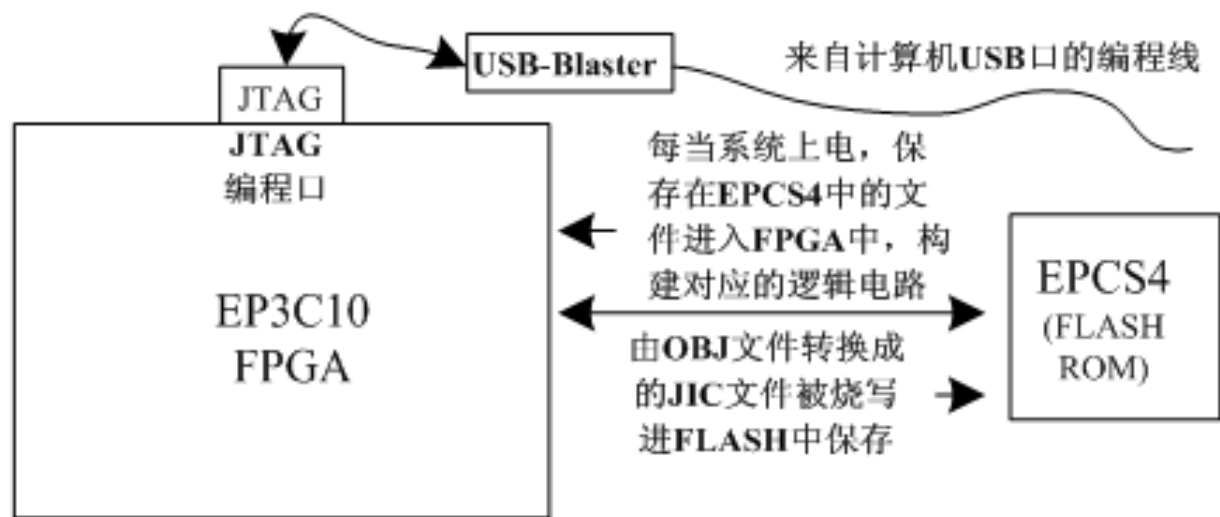


图 6-22 利用 USB-Blaster 经由 FPGA 向 FPGA 的专用配置器件 EPCS4 编程下载

# 6.4 硬件测试

## 6.4.3 对FPGA配置器件编程

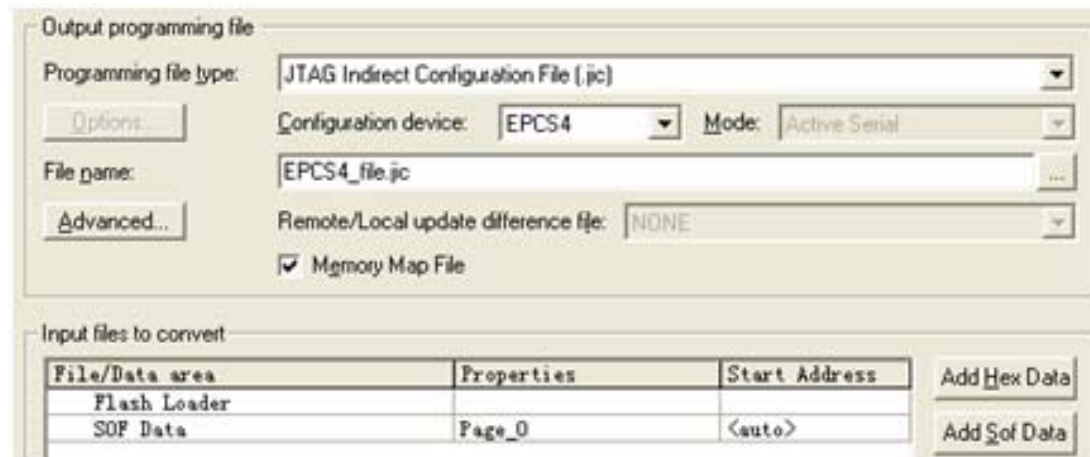


图 6-23 设定 JTAG 间接编程文件

# 6.4 硬件测试

## 6.4.3 对FPGA配置器件编程



图 6-24 选择目标器件 EP3C10

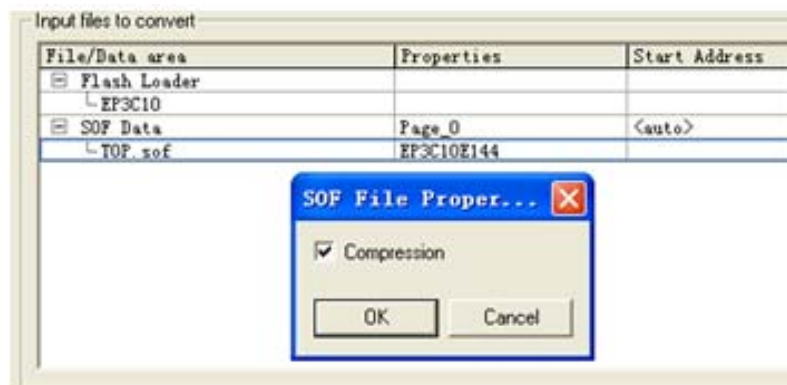


图 6-25 选定 SOF 文件后，选择文件压缩

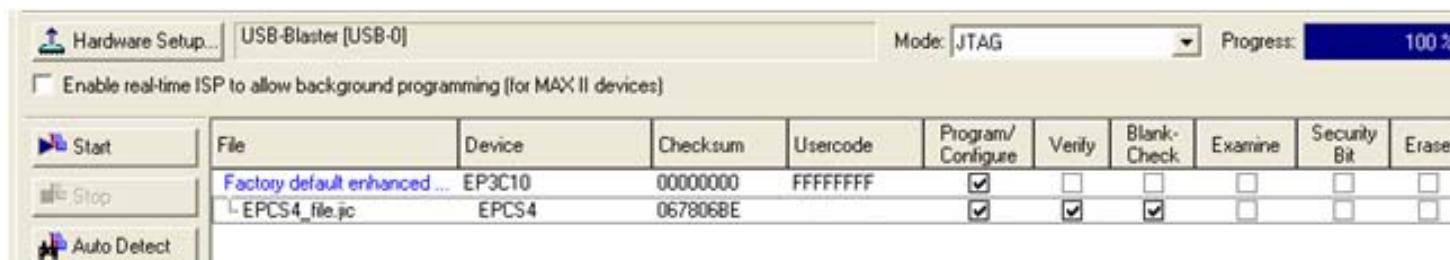


图 6-26 用 JTAG 模式经由 FPGA 对配置器件 EPCS4 进行间接编程

# 6.5 用HDL来表述广义译码器

## 6.5.1 用HDL表述真值表与电路设计

### 1. HDL表述

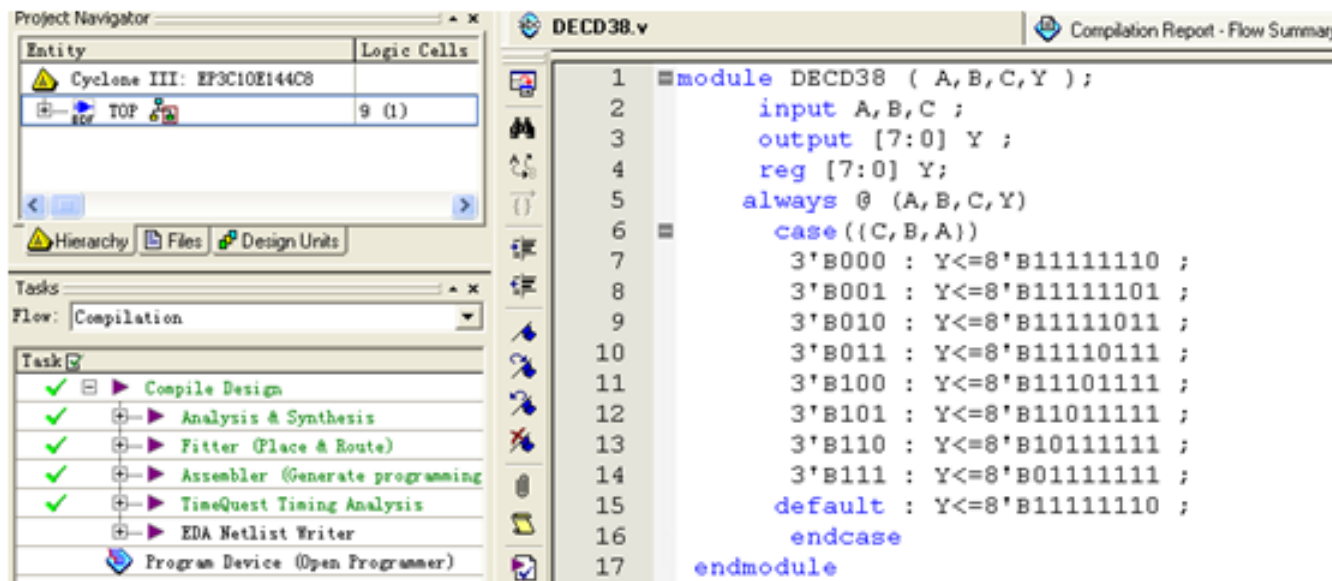


图 6-27 3-8 译码器真值表对应的 HDL 的 case 语句表述

# 6.5 用HDL来表述广义译码器

## 6.5.1 用HDL表述真值表与电路设计

### 2. 将Verilog文本表述转化为电路元件

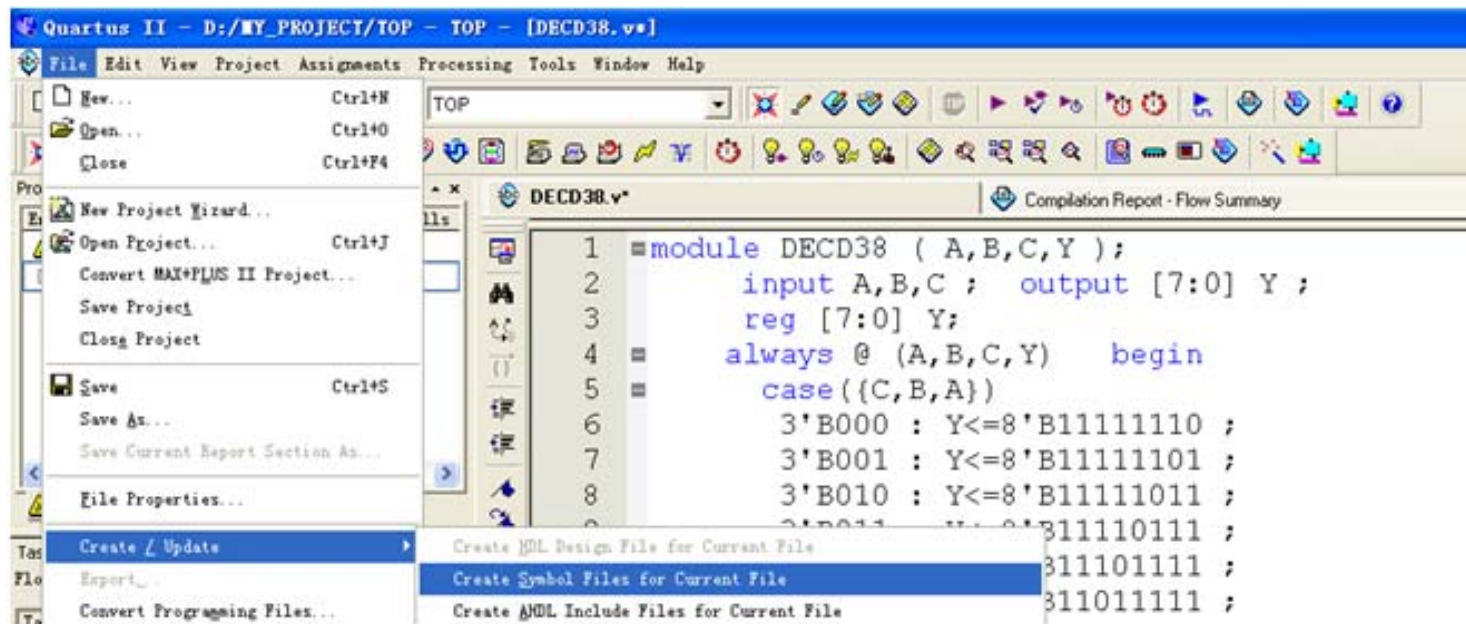


图 6-28 将 HDL 程序变成一个可以调用的原件模块

# 6.5 用HDL来表述广义译码器

## 6.5.1 用HDL表述真值表与电路设计

- 3. 完成电路设计
- 4. 逻辑功能测试

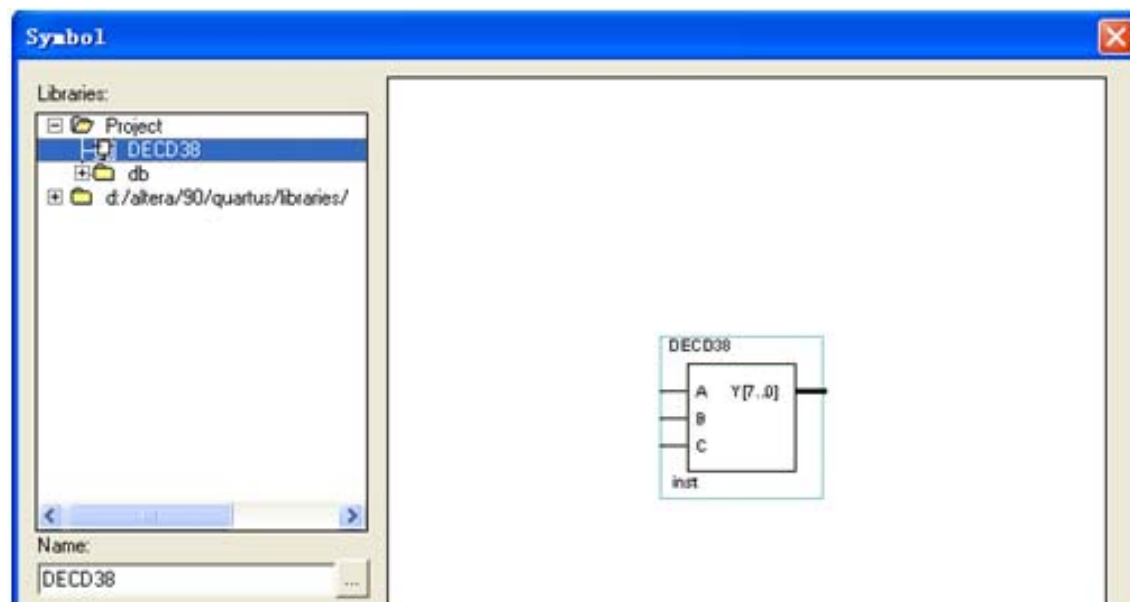


图 6-29 选择已生成好的元件 DECD38

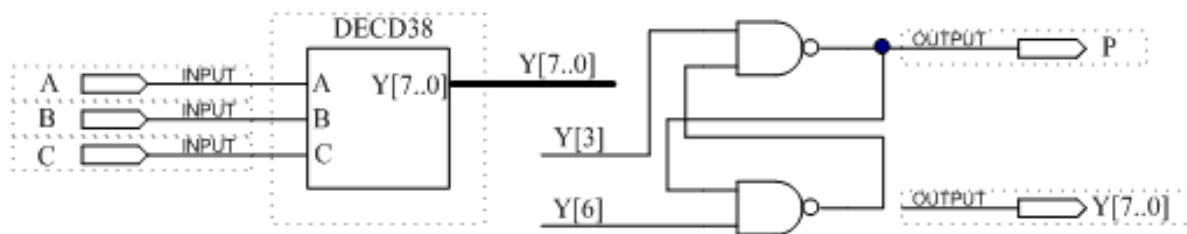


图 6-30 用HDL文本表述的3-8译码器DECD38连接好电路

# 6.5 用HDL来表述广义译码器

## 6.5.2 三人表决电路的HDL表述方式

```
1 module JG3 (A,B,C,X,Y);
2   input A,B,C; output X,Y ;
3   reg X,Y ;
4   always @(A,B,C,X,Y)
5   = case ( {A,B,C} )
6     3'B000 : begin X<=0; Y<=1; end
7     3'B001 : begin X<=0; Y<=0; end
8     3'B010 : begin X<=0; Y<=0; end
9     3'B011 : begin X<=0; Y<=0; end
10    3'B100 : begin X<=0; Y<=0; end
11    3'B101 : begin X<=1; Y<=0; end
12    3'B110 : begin X<=1; Y<=0; end
13    3'B111 : begin X<=1; Y<=0; end
14    default : begin X<=1; Y<=0; end
15  endcase
16 endmodule
```

(a)

```
1 module JG3 (A,B,C,X,Y);
2   input A,B,C; output X,Y ;
3   reg X,Y ;
4   always @(A,B,C,X,Y)
5   = case ( {A,B,C} )
6     3'B000 : {X,Y} <=2'B01;
7     3'B001 : {X,Y} <=2'B00;
8     3'B010 : {X,Y} <=2'B00;
9     3'B011 : {X,Y} <=2'B00;
10    3'B100 : {X,Y} <=2'B00;
11    3'B101 : {X,Y} <=2'B10;
12    3'B110 : {X,Y} <=2'B10;
13    3'B111 : {X,Y} <=2'B10;
14    default : {X,Y} <=2'B10;
15  endcase
16 endmodule
```

(b)

图 6-31 表 4-3 对应的两种 case 语句的表述方式



# 6.5 用HDL来表述广义译码器

## 6.5.2 三人表决电路的HDL表述方式

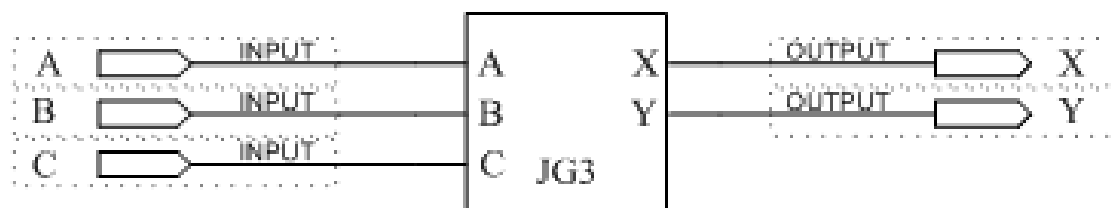


图 6-32 表决电路测试电路原理图

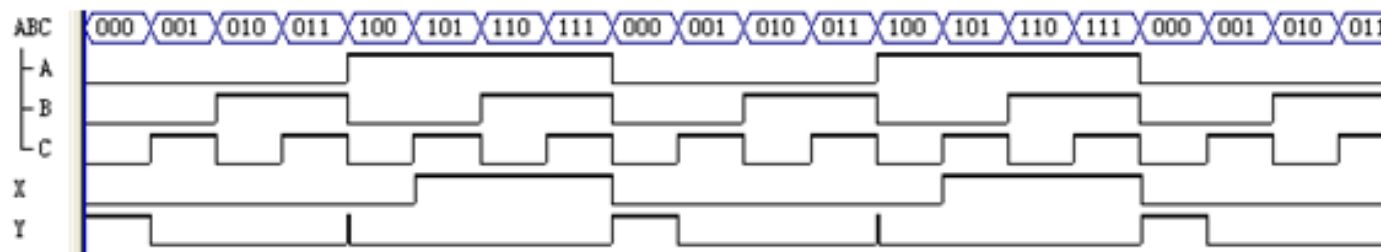


图 6-33 三人表决电路的仿真波形

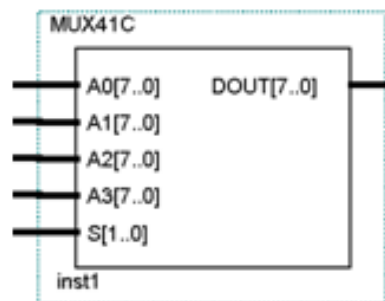
# 6.5 用HDL来表述广义译码器

## 6.5.3 用HDL对真值表的其他表述方式

### 1. 文字表达方式的多路选择器设计

```
Simulation Report - Flow Summary | MUX41C.v*
1  module MUX41C ( A0,A1,A2,A3,S,DOUT );
2      input [7:0] A0,A1,A2,A3 ; input [1:0]S;
3      output [7:0] DOUT ;
4      reg [7:0] DOUT ;
5      always @ (A0,A1,A2,A3,S,DOUT)
6      case(S)
7          2'B00 : DOUT <= A0 ;
8          2'B01 : DOUT <= A1 ;
9          2'B10 : DOUT <= A2 ;
10         2'B11 : DOUT <= A3 ;
11         default : DOUT <= 8'B00000000;
12     endcase
13 endmodule
```

(a) 8位4选1型多路选择器的 case 语句描述



(b) 由(a)代码生成的原理图原件符号

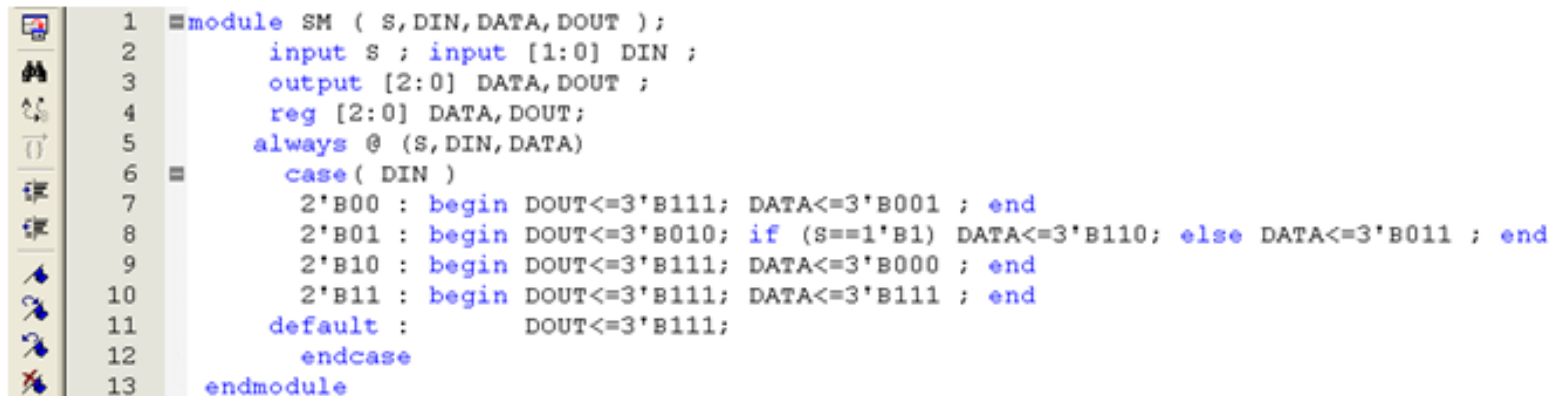
图 6-34 8位四通道选一型多路选择器的“真值表”描述及转换后的原理图元件模块

# 6.5 用HDL来表述广义译码器

## 6.5.3 用HDL对真值表的其他表述方式

### 2. 含有条件判定情况的真值表的case语句表述

```
begin D<=2'b01; if (S==1) begin A<=2'b10; B<=2'b00; end
      else begin A<=2'b01; B<=2'b11; end ; end
```



```
1 module SM ( S, DIN, DATA, DOUT );
2     input S ; input [1:0] DIN ;
3     output [2:0] DATA, DOUT ;
4     reg [2:0] DATA, DOUT;
5     always @ ( S, DIN, DATA)
6     case( DIN )
7         2'B00 : begin DOUT<=3'B111; DATA<=3'B001 ; end
8         2'B01 : begin DOUT<=3'B010; if (S==1'B1) DATA<=3'B110; else DATA<=3'B011 ; end
9         2'B10 : begin DOUT<=3'B111; DATA<=3'B000 ; end
10        2'B11 : begin DOUT<=3'B111; DATA<=3'B111 ; end
11        default : DOUT<=3'B111;
12    endcase
13 endmodule
```

# 6.5 用HDL来表述广义译码器

## 6.5.3 用HDL对真值表的其他表述方式

### 2. 含有条件判定情况的真值表的case语句表述

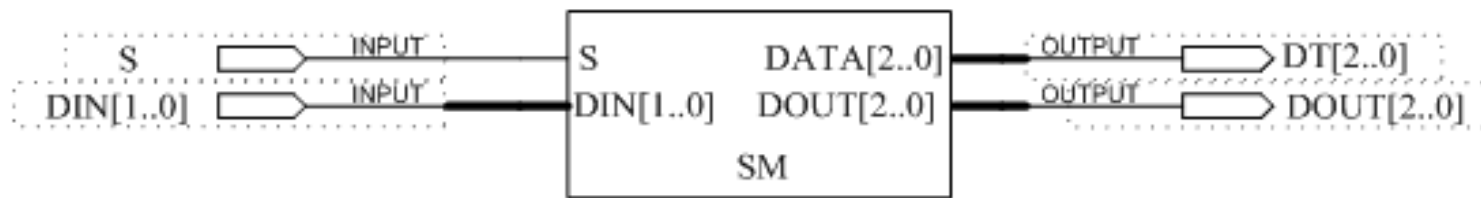


图 6-36 对应图 6-35 程序的测试电路原理图

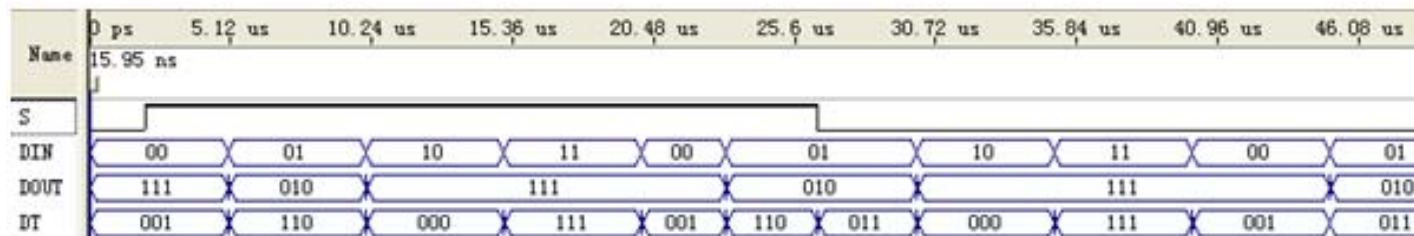


图 6-37 图 6-36 电路的仿真波形

# 6.6 数字方法去抖动和延时电路设计

## 6.6.1 数字去抖动电路设计

### 1. 脉冲参数和信号频率概念

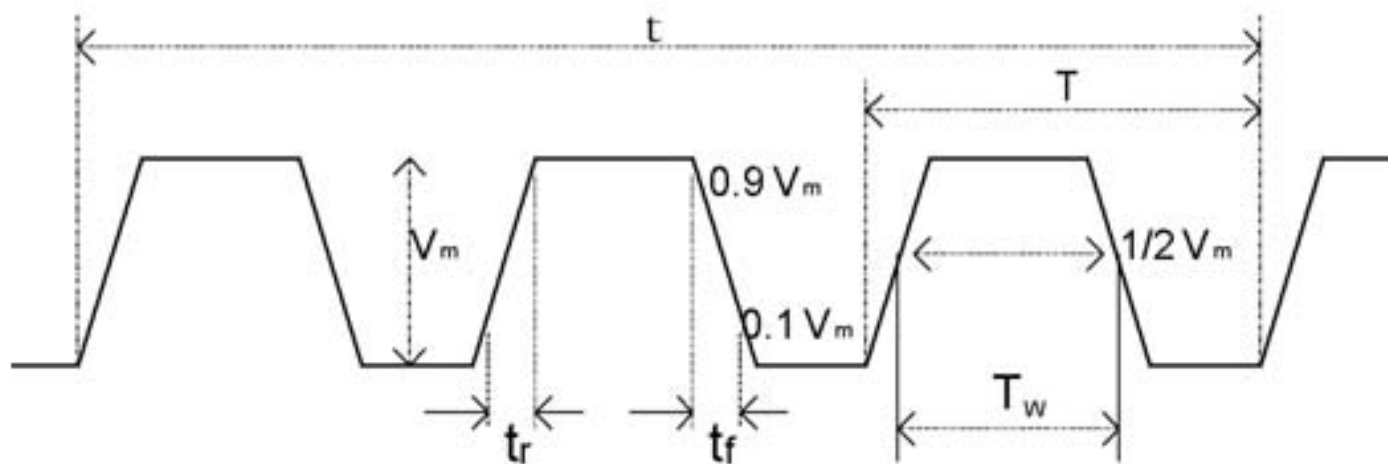


图 6-38 脉冲波形概念和参数说明图

# 6.6 数字方法去抖动和延时电路设计

## 6.6.1 数字去抖动电路设计

### 2. 数字去抖动电路设计



图 6-39 在信号上升与下降沿含随机干扰抖动信号的信号

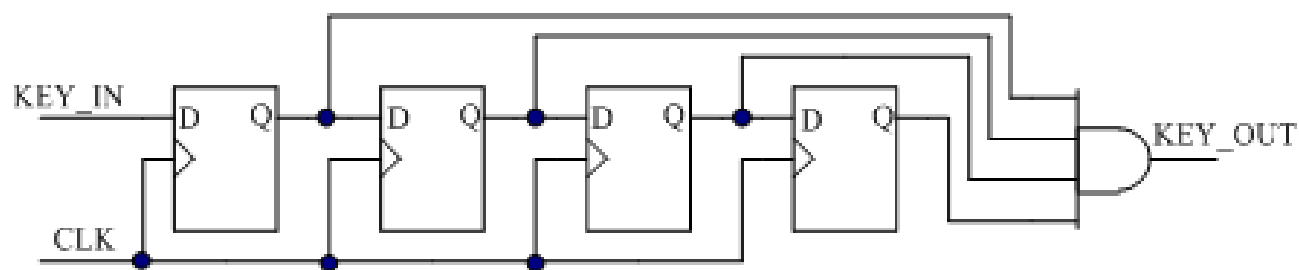


图 6-40 数字消抖动电路

# 6.6 数字方法去抖动和延时电路设计

## 6.6.1 数字去抖动电路设计

### 3. 时序仿真

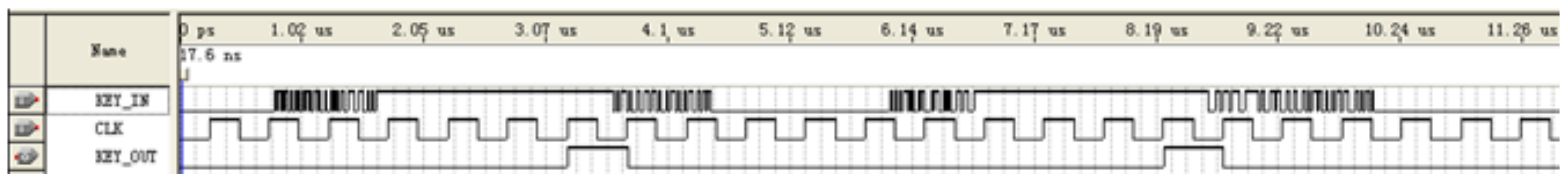


图 6-41 消抖动电路仿真波形

# 6.6 数字方法去抖动和延时电路设计

## 6.6.1 数字去抖动电路设计

### 3. 时序仿真

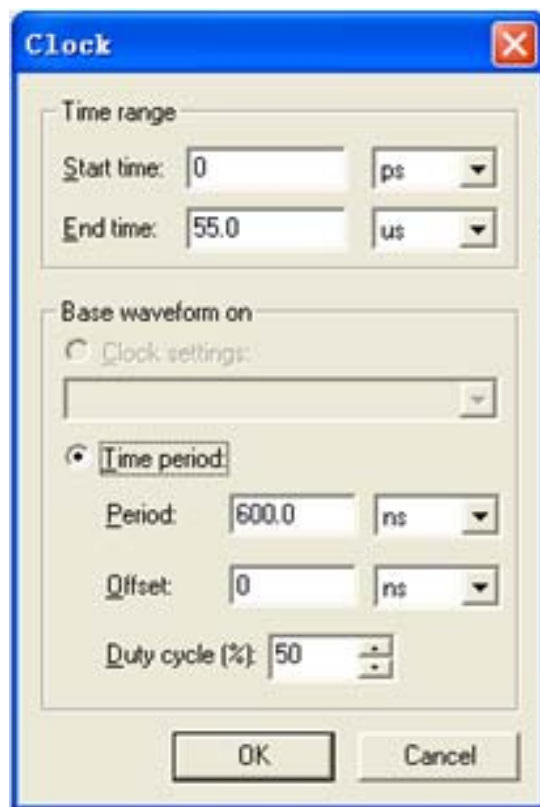


图 6-42 设置时钟周期

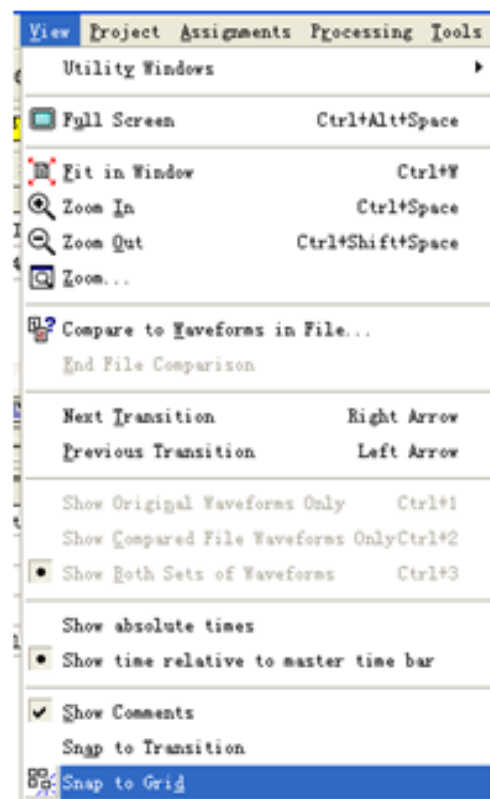


图 6-43 关闭分格限制



# 6.6 数字方法去抖动和延时电路设计

## 6.6.2 数字延时电路的设计与测试

### 1. 设计一个库元件

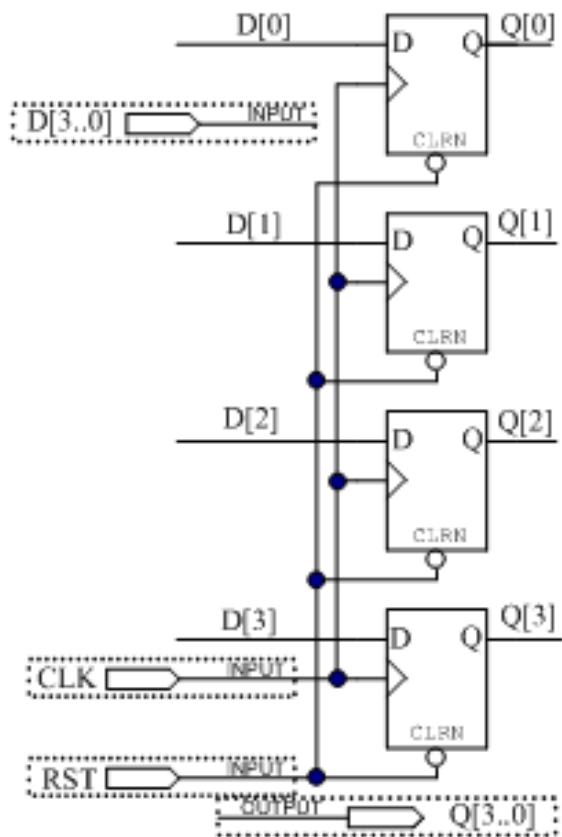


图 6-44 DFF4 四位寄存器电路

# 6.6 数字方法去抖动和延时电路设计

## 6.6.2 数字延时电路的设计与测试

### 2. 设计顶层电路

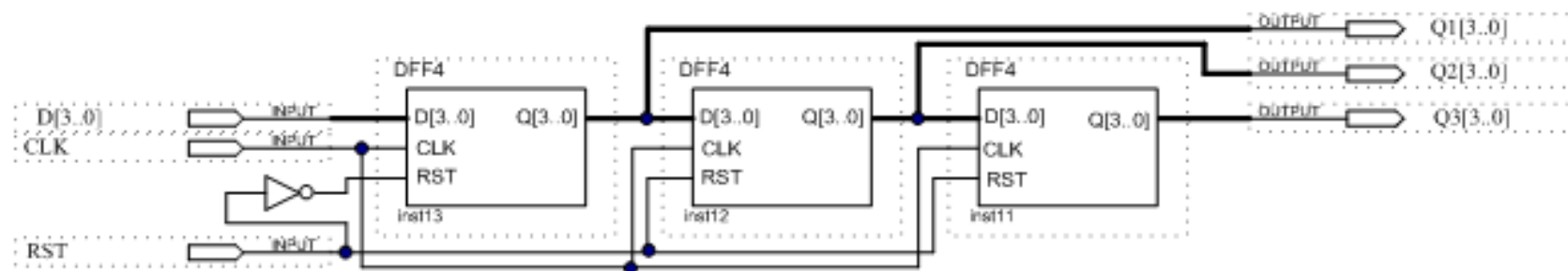


图 6-45 延时测试电路

# 6.6 数字方法去抖动和延时电路设计

## 6.6.2 数字延时电路的设计与测试

### 3. 时序仿真

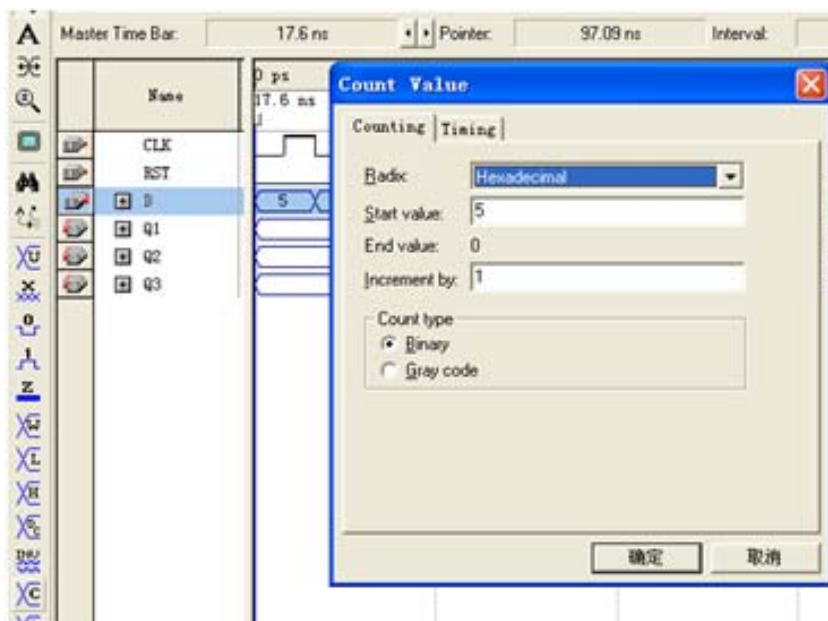


图 6-46 设置仿真用输入数据



图 6-47 设置递增型输入数据时间间隔

# 6.6 数字方法去抖动和延时电路设计

## 6.6.2 数字延时电路的设计与测试

### 3. 时序仿真

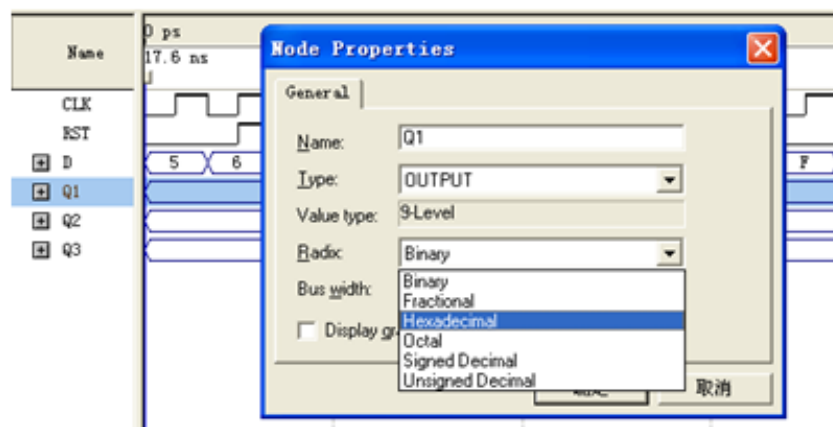


图 6-48 设置仿真信号数据表述格式

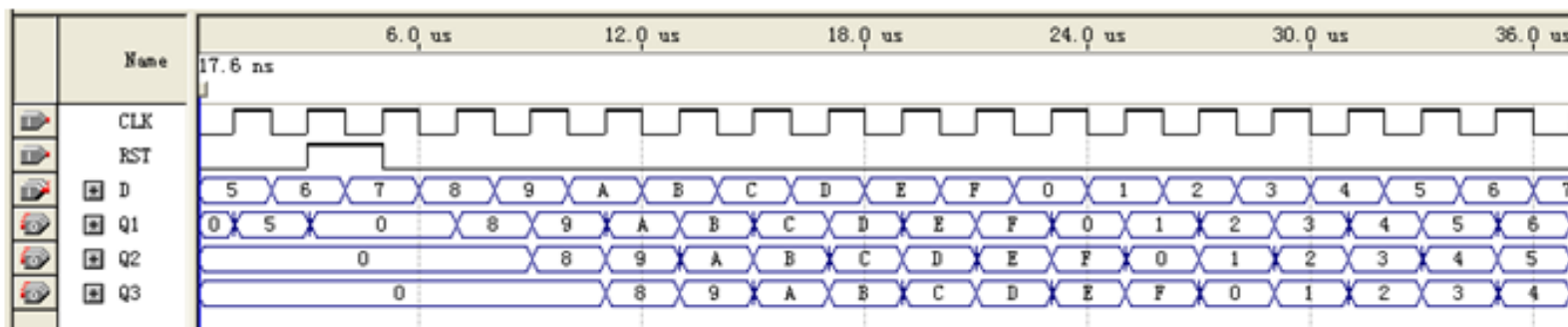
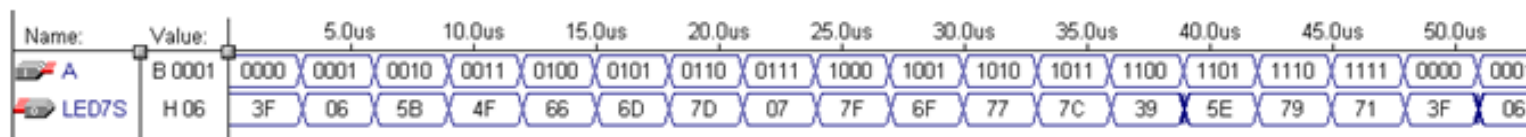


图 6-49 图 6-45 电路仿真波形

# 实验与设计

- 6-1. 用Quartus II库中的宏功能模块74138和与非门实现指定逻辑函数
- 6-2. 用两片7485设计一个8位比较器
- 6-3. 设计8位串行进位加法器
- 6-4. 设计八位十进制数动态扫描显示控制电路
- 6-5. 设计一个16进制7段显示译码器



- 6-6. 设计一个5人表决电路