EDA技术

第5章 时序电路的Verilog设计



《EDA技术实用教程——Verilog HDL版》
（第七版）
配套课件

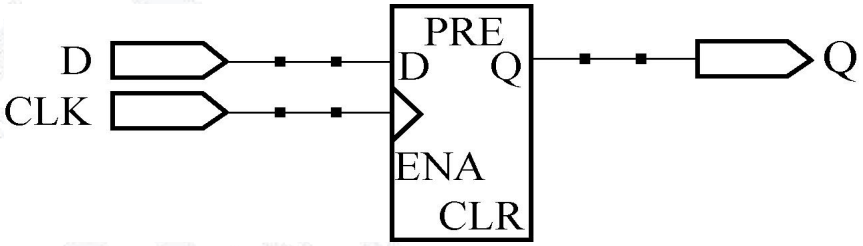
EDA Lab

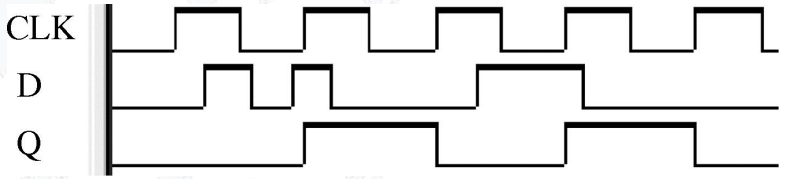
5.1 基本时序元件的Verilog表述

5.1.1 基本D触发器

【例5-1】

```
module DFF1(CLK,D,Q);
      output Q;
      input  CLK, D;
      reg Q;
      always @(posedge CLK)
      Q <= D;
endmodule
```





- posedge CLK是对时钟信号CLK上升沿敏感的敏感信号。每当CLK出现一个上升沿时，敏感信号posedge CLK将启动过程语句。
- 相对应的，还有negedge CLK时钟下降沿敏感。

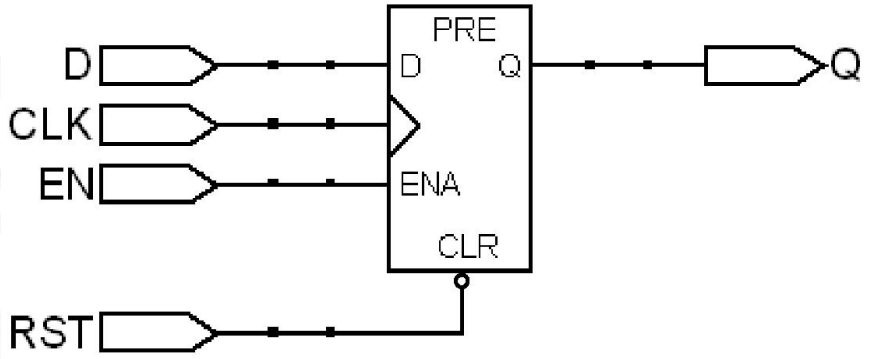
EDA Lab

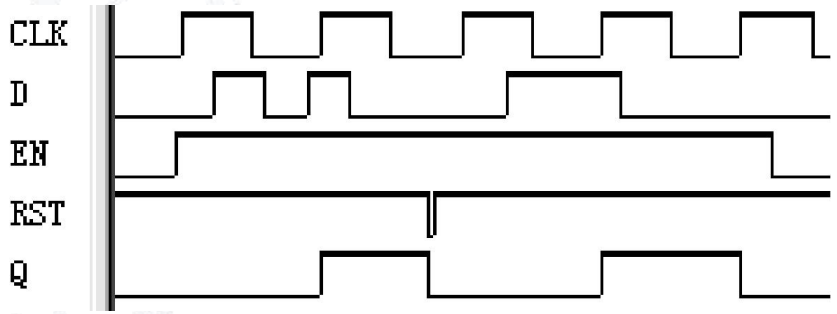
5.1 基本时序元件的Verilog表述

5.1.2 含异步复位和时钟使能的D触发器

【例5-2】

```
module DFF2(CLK,D,Q,RST,EN);
    output Q;          input CLK,D,RST,EN;
    reg Q;
    always @(posedge CLK or negedge RST)
    begin
            if (!RST)   Q <= 0;
       else if (EN)     Q <= D;     end
endmodule
```





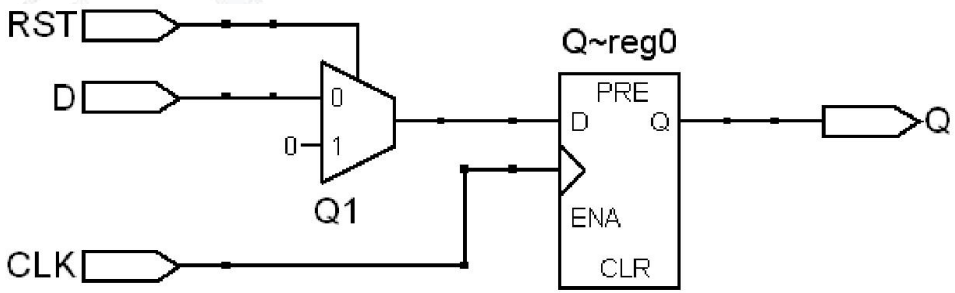
- 只要RST=0，此D触发器的输出端即刻被清0；
- 只有EN=1，时钟上升沿才有效。

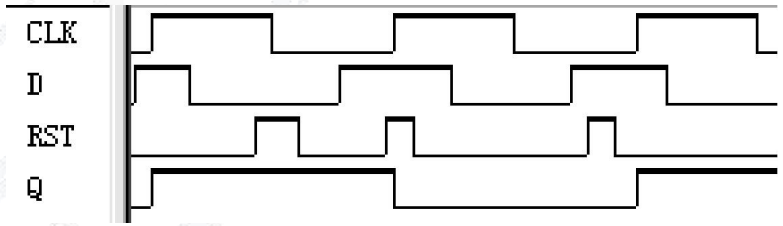
5.1 基本时序元件的Verilog表述

5.1.3 含同步复位控制的D触发器

【例5-3】

```
module DFF3(CLK,D,Q,RST);
    output Q;
    input CLK,D,RST;
    reg Q;
    always @(posedge CLK)
        if (RST==1)   Q = 0;
   else if (RST==0)   Q = D;
endmodule
```





- 同步：控制信号只有在时钟信号有效时才起作用

5.1 基本时序元件的Verilog表述



5.1.3 含同步复位控制的D触发器

【例5-4】

```
module DFF1(CLK,D,Q,RST);
  output Q;  input CLK,D,RST;
  reg Q, Q1;  //注意定义了Q1信号
  always @(RST)  //纯组合过程
   if (RST==1)  Q1=0;  else  Q1=D;
   always @(posedge CLK)
          Q<=Q1;
endmodule
```

【例5-5】

```
module DFF2 (input CLK, input D, input RST, output reg Q);
     always @(posedge CLK)
     Q <= RST ? 1'b0 : D;
endmodule
```

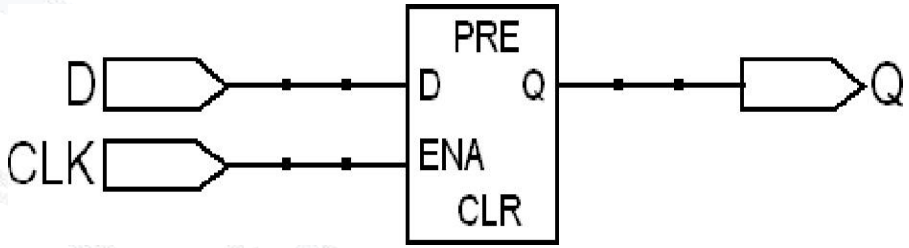
EDA Lab

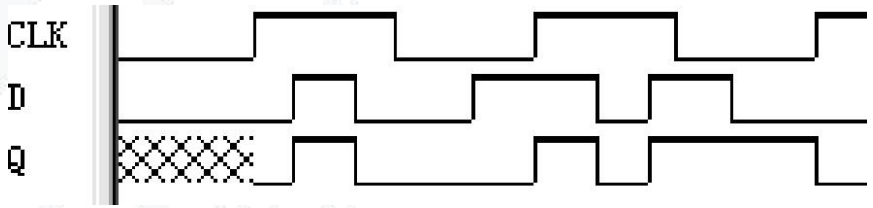
5.1 基本时序元件的Verilog表述

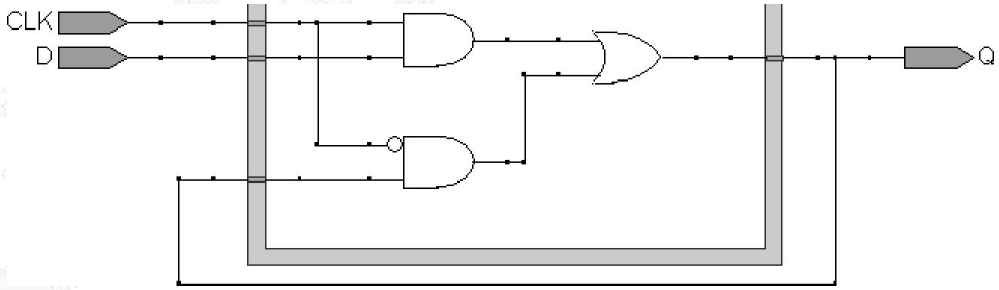
5.1.4 基本锁存器

【例5-6】

```
module LATCH1(CLK,D,Q);
    output Q;
    input CLK,D;
    reg Q;
    always @(D or CLK)
     if(CLK)  Q <= D;
endmodule
```







5.1 基本时序元件的Verilog表述

5.1.5 含清0控制的锁存器

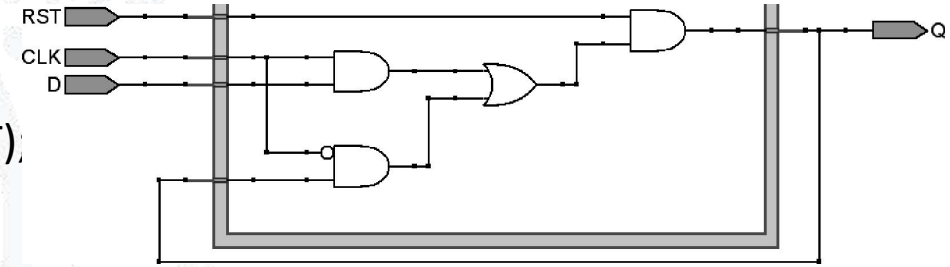
含异步清0锁存器的Verilog设计表述可以有多种，例5-7和例5-8给出了两种完全不同风格的设计示例。

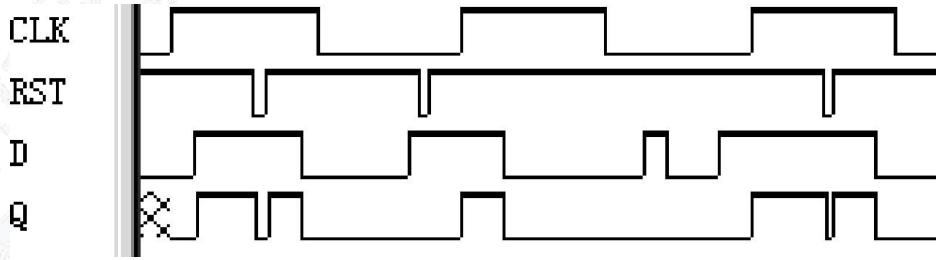
【例5-7】

```
module LATCH2 (CLK,D,Q,RST);
  output Q;
 input CLK,D,RST;
  assign Q = (!RST)? 0:(CLK ? D:Q);
endmodule
```

【例5-8】

```
module LATCH3 (CLK,D,Q,RST);
  output Q;
 input CLK,D,RST;
  reg Q;
  always @(D or CLK or RST)
 if(!RST)  Q<=0;
    else   if(CLK)   Q<=D;
endmodule
```





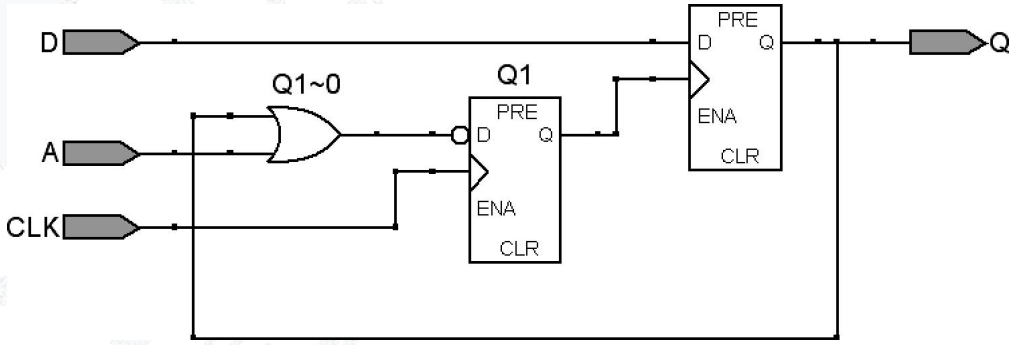
5.1 基本时序元件的Verilog表述

5.1.6 异步时序电路的Verilog表述特点

含异步清0锁存器的Verilog设计表述可以有多种，例5-7和例5-8给出了两种完全不同风格的设计示例。

【例5-9】

```
module AMOD(D,A,CLK,Q);
output Q;  input A,D,CLK;
reg Q,Q1;
always @(posedge CLK)  Q1 = ~(A|Q);
always @(posedge Q1)  Q =  D;
endmodule
```



5.1 基本时序元件的Verilog表述



5.1.7 时钟过程表述的特点和规律

边沿触发型时序模块的Verilog设计，可遵循以下几个规律：

（1）若将某信号A定义为边沿敏感时钟信号，则必须在敏感信号表中给出对应的表述，如posedge A或negedge A；但在always过程结构中不能再出现信号A了。

（2）若将某信号B定义为对应于时钟的电平敏感的异步控制信号，则除了在敏感信号表中给出对应的表述外，如posedge B或negedge B，在always过程结构中必须明示信号B的逻辑行为。

EDA Lab

5.1 基本时序元件的Verilog表述



5.1.7 时钟过程表述的特点和规律

（3）若将某信号定义为对应于时钟的同步控制信号（或仅仅是同步输入信号），则绝不可以以任何形式出现在敏感信号表中。

- 敏感信号表中不允许出现混合信号
- 敏感信号表一旦含有posedge或negedge的边沿敏感信号后（即某单边沿），所有其他普通变量都不能放在敏感信号表中了

EDA Lab

5.1 基本时序元件的Verilog表述



5.1.7 时钟过程表述的特点和规律

- 若定义某变量（如RST）为异步低电平敏感信号，则在if条件句中应该对敏感信号表中的信号有匹配的表述，如以下3种表述方式都是正确的：

```
always @ (posedge CLK or negedge RST) begin if (!RST)    ...
always @ (posedge CLK or negedge RST) begin if (RST==0)   ...
always @ (posedge CLK or negedge RST) begin if (!RST==1) ...
```

EDA Lab

5.1 基本时序元件的Verilog表述

5.1.7 时钟过程表述的特点和规律

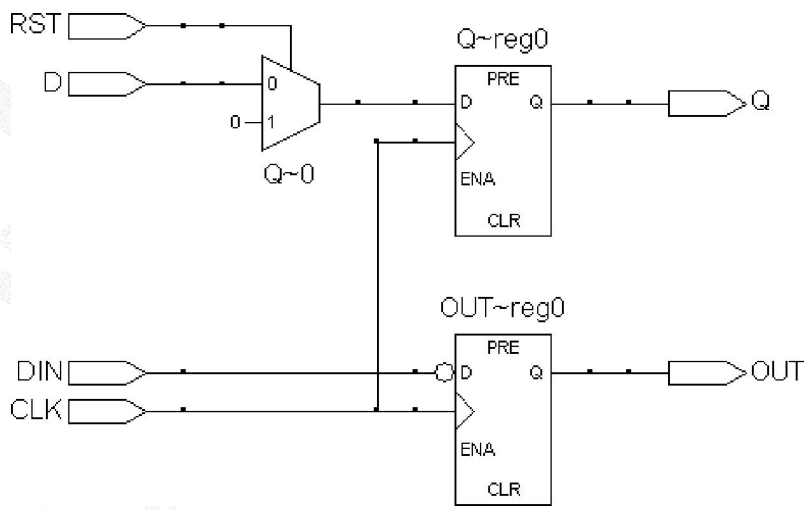


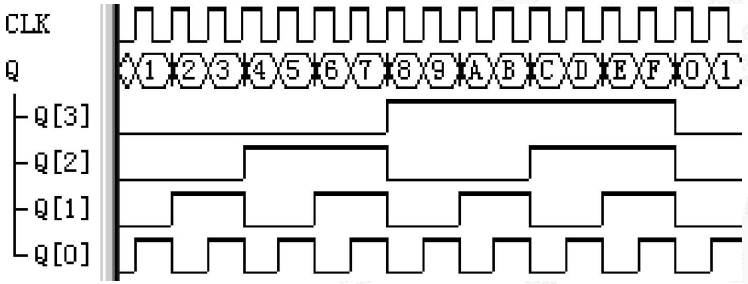
图5-14 例5-10的RTL电路图

【例5-10】

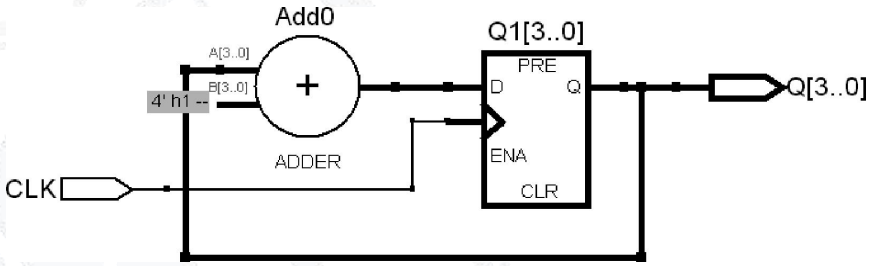
```
module DFF5(CLK,D,Q,RST,DIN,OUT);
output Q,OUT; input CLK,D,RST,DIN;
    reg Q,OUT;
always @(posedge CLK)  begin    OUT = !DIN;
if (RST==1)  Q=0;
else if(RST==0)  Q=D;  end
endmodule
```

5.2 二进制计数器及其Verilog表述

5.2.1 简单加法计数器及其Verilog表述



4位加法计数器工作时序图



4位加法计数器RTL电路图

【例5-11】

```
module CNT4(CLK,Q);
   output[3:0] Q;    input  CLK;
   reg[3:0] Q1;
   always @(posedge CLK)
        Q1 <= Q1+1;
   assign Q=Q1;
endmodule
```

【例5-12】

```
module CNT4 (CLK,Q);
   output[3:0] Q;
   input  CLK;
   reg[3:0] Q;
   always @(posedge CLK)
        Q <= Q+1;
endmodule
```

EDA Lab

5.2 二进制计数器及其Verilog表述

5.2.2 实用加法计数器设计

一个含异步复位、同步计数使能和可预置的十进制计数器

【例5-13】

```
module CNT10 (CLK,RST,EN,LOAD,COUT,DOUT,DATA);
    input CLK,EN,RST,LOAD;          // 时钟，时钟使能，复位，数据加载控制信号
    input[3:0] DATA;                // 4位并行加载数据
    output[3:0] DOUT;               // 4位计数输出
    output COUT;                    // 计数进位输出
    reg[3:0] Q1;      reg COUT;
    assign DOUT = Q1;               // 将内部寄存器的计数结果输出至DOUT
```

EDA Lab

5.2 二进制计数器及其Verilog表述



5.2.2 实用加法计数器设计

一个含异步复位、同步计数使能和可预置的十进制计数器

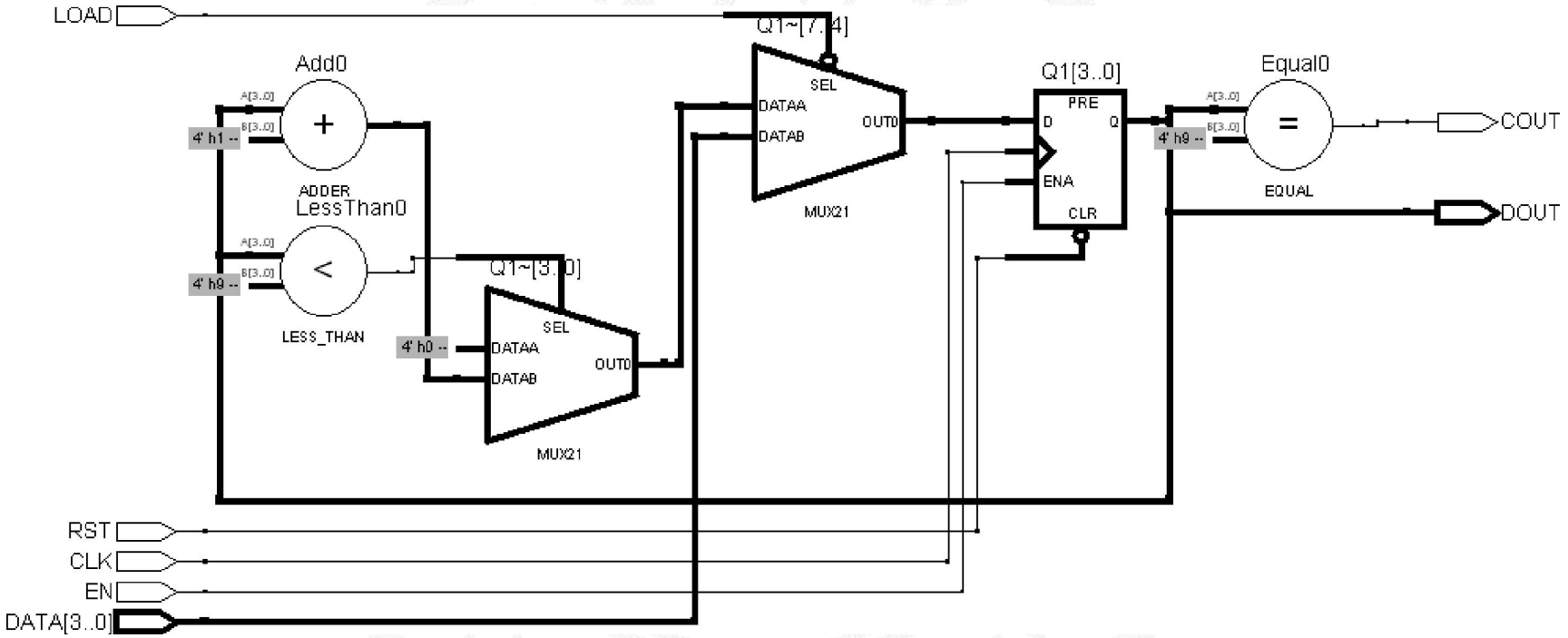
【例5-13】

```
    always @(posedge CLK or negedge RST)    //时序过程
       begin
          if (!RST)    Q1 <= 0;          //RST=0时，对内部寄存器单元异步清0
          else  if (EN)                  //同步使能EN=1，则允许加载或计数
              begin
                  if (!LOAD)      Q1<=DATA;        //当LOAD=0，向内部寄存器加载数据
                  else  if (Q1<9)   Q1 <= Q1+1;    //当Q1小于9时，允许累加
                  else    Q1 <= 4'b0000;           //否则一个时钟后清0返回初值
              end
       end
    always @(Q1)                                  //组合过程
        if (Q1==4'h9)  COUT = 1'b1;  else   COUT = 1'b0;
endmodule
```

EDA Lab

5.2 二进制计数器及其Verilog表述

5.2.2 实用加法计数器设计



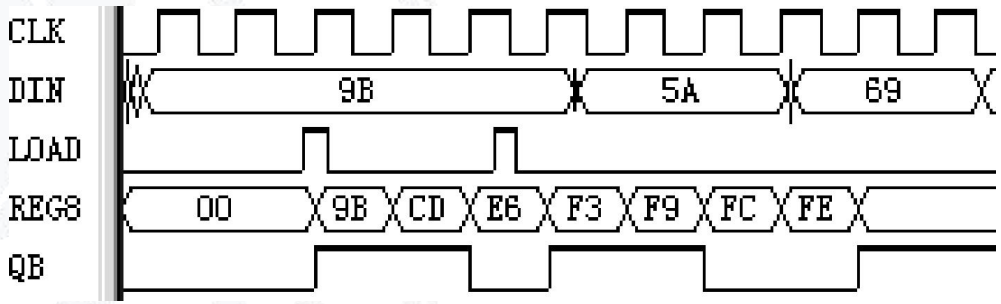
Quartus对例5-13综合和后得到的RTL电路图

5.3 移位寄存器的Verilog表述与设计

5.3.1 含同步预置功能的移位寄存器设计

【例5-14】

```
module SHFT1(CLK,LOAD,DIN,QB);
   output QB;
   input CLK,LOAD;
   input[7:0] DIN;
   reg[7:0] REG8;
   always @(posedge CLK)
     if (LOAD)      REG8<=DIN;
     else REG8[6:0]<=REG8[7:1];
   assign QB = REG8[0];
endmodule
```



5.3 移位寄存器的Verilog表述与设计



5.3.2 使用移位操作符设计移位寄存器

【例5-15】

```
module SHIF4 (DIN,CLK,RST,DOUT);
      input CLK,DIN,RST;    output DOUT;
      reg[3:0] SHFT;
      always@(posedge CLK or posedge RST)
         if(RST)   SHFT<=4'B0;
         else begin
             SHFT <=(SHFT >> 1);  SHFT[3] <= DIN;
         end
       assign   DOUT   = SHFT[0];
endmodule
```

- >>右移或<<左移，移出腾空的位用0填补。
- 有符号数右移>>>/左移<<<，右移一律将符号位即最高位填补移出的位，左移操作同普通左移<<

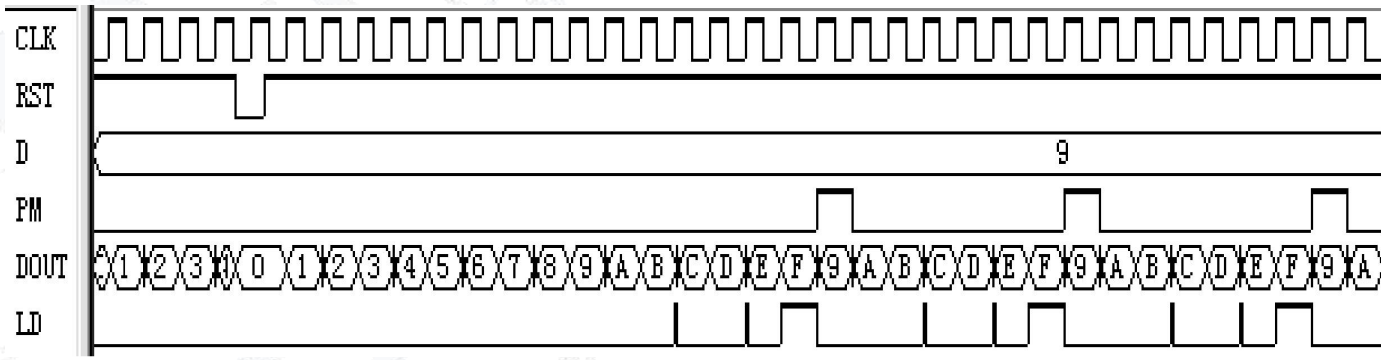
EDA Lab

5.4 可预置型计数器设计

5.4.1 同步加载计数器

【例5-16】

```
module FDIV0 (
input CLK,RST,input[3:0]D,
output PM,output[3:0]DOUT);
   reg[3:0] Q1;      reg FULL;
   wire LD;
   always @(posedge CLK or negedge RST)
     if (!RST)  begin Q1<=0; FULL<=0;
end
     else if (LD) begin Q1<=D; FULL<=1;
end
     else  begin Q1 <= Q1+1; FULL<=0;
end
   assign LD=(Q1==4'B1111);
   assign PM=FULL;
   assign DOUT=Q1;
endmodule
```



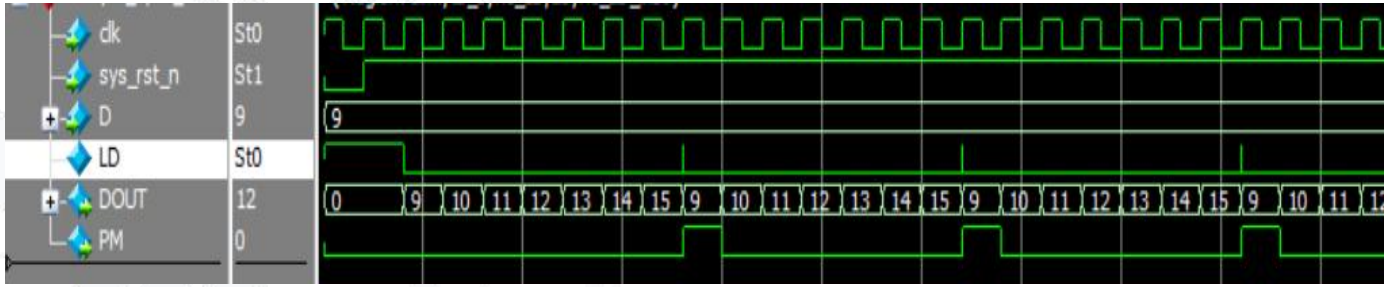
仿真波形

5.4 可预置型计数器设计

5.4.2 异步加载计数器

【例5-17】

```
module fdiv1 (CLK,PM,D,DOUT,RST);
  input CLK;  input RST; input[3:0] D;
  output PM; output[3:0] DOUT;
  reg[3:0] Q1;   reg FULL;
  wire LD;
  always @(posedge CLK or posedge LD or
negedge RST)
    if (!RST)  begin Q1<=0;  FULL<=0; end
    else  if(LD)   begin Q1<=D;   FULL<=1;
end
    else  begin Q1<=Q1+1; FULL<=0; end
  assign LD=(Q1==4'B0000);
  assign PM=FULL;
  assign DOUT=Q1;
endmodule
```



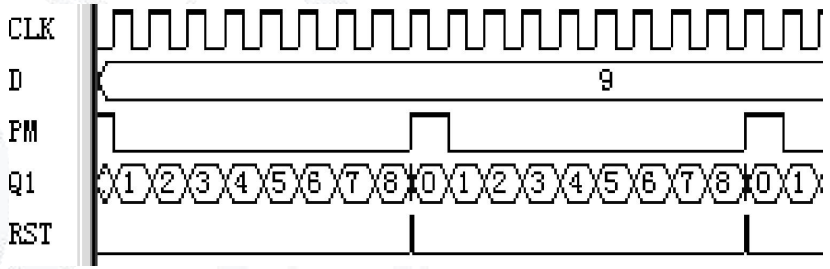
仿真波形

5.4 可预置型计数器设计

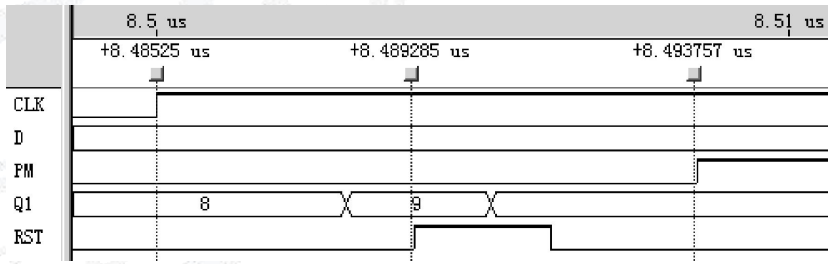
5.4.3 异步清0加载计数器

【例5-18】

```
module fdiv1 (CLK,PM,D);
    input CLK; input[3:0] D;
    output PM;
    reg FULL;
    reg[3:0] Q1;
    wire RST;
    always @(posedge CLK or posedge RST)
        if (RST)  begin Q1<=0;  FULL<=1; end
        else  begin Q1<=Q1+1;  FULL<=0; end
    assign RST = (Q1==D);
    assign  PM = FULL;
endmodule
```



仿真波形



RST信号展开后的时序

由于异步复位，计数进入9的瞬间立即被置0

5.4 可预置型计数器设计

5.4.4 同步清0加载计数器

【例5-18】

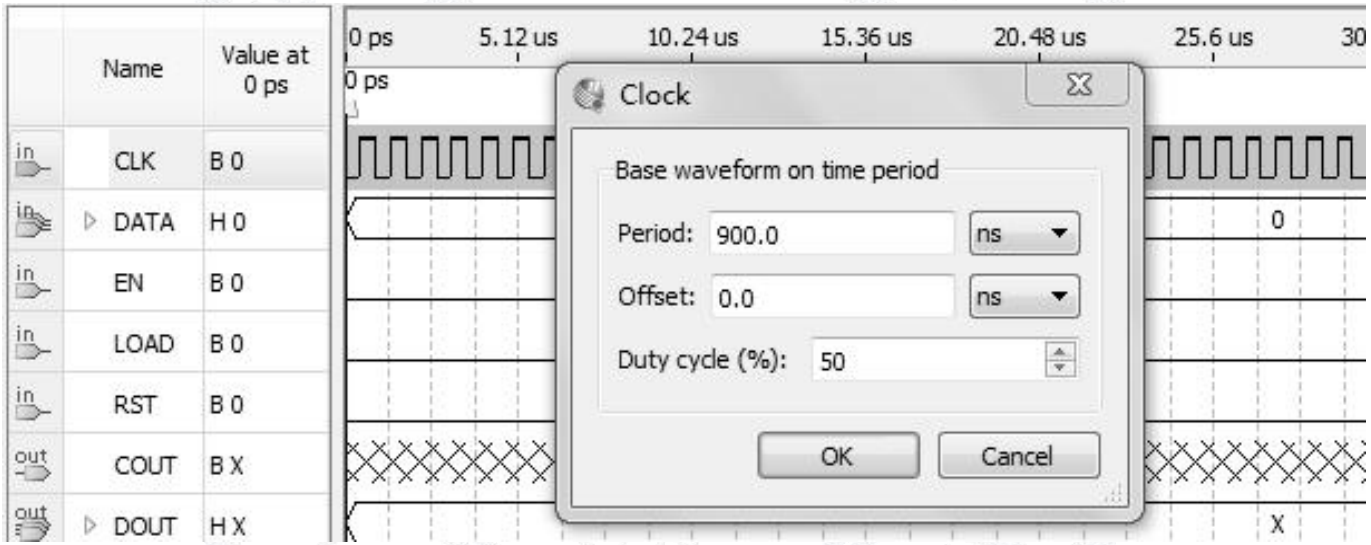
```
module fdiv1 (CLK,PM,D );
  input CLK; input[3:0] D;    output PM ; reg FULL;
  (* synthesis,probe_port,keep *)  reg [3:0] Q1;
  (* synthesis,probe_port,keep *)  wire RST ;
   always @(posedge CLK or posedge RST)
       if (RST) begin Q1<=0;
          FULL<=1;  end
        else  begin Q1<=Q1+1;
          FULL<=0 ;  end
    assign RST = (Q1==D);
    assign  PM = FULL ;
 endmodule
```

always @ (posedge CLK)

只需要改异步清0加载计数器的敏感信号表，其他部分不动

5.5 时序电路硬件设计与仿真示例

5.5.1 编辑电路、创建工程和仿真测试



设置时钟CLK的周期

5.5 时序电路硬件设计与仿真示例

5.5.2 FPGA硬件测试

表 5-1　基于 HX1006A 的 10CL006YU356 引脚锁定情况

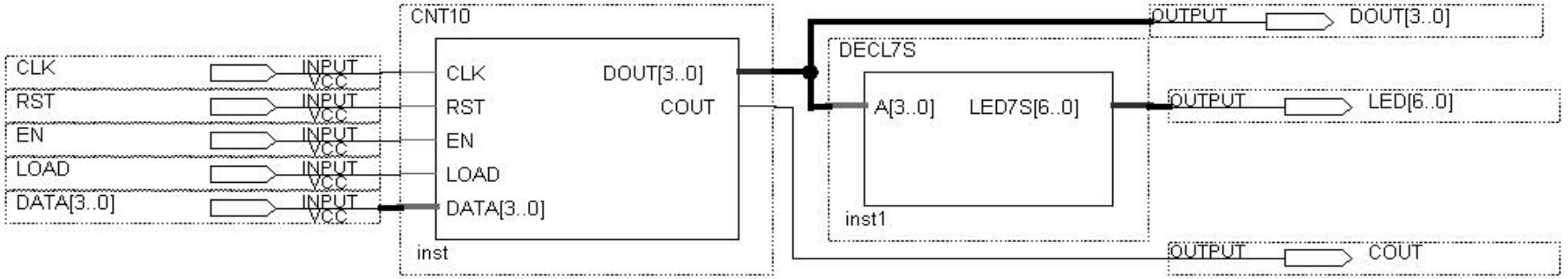
计数器信号名	CLK	EN	LOAD	RST	DATA（3）	DATA（2）	DATA（1）
开发板功能名	LED7 红	KEY3	KEY2	KEY1	SW4	SW3	SW2
对应 FPGA 引脚	C15	E1	F3	M1	M2	M15	M16
计数器信号名	DATA（0）	COUT	DOUT（3）	DOUT（2）	DOUT（1）	DOUT（0）	CLK50M
开发板功能名	SW1	LED6 红	LED4	LED3	LED2	LED1	50M 时钟
对应 FPGA 引脚	E16	D15	K8	T9	R9	T10	E15

EDA Lab

5.6 SignalTap II的使用方法

SignalTap II嵌入式逻辑分析仪：用以捕捉目标芯片内部系统信号节点处的信息或总线上的数据流，且不影响原硬件系统正常工作。

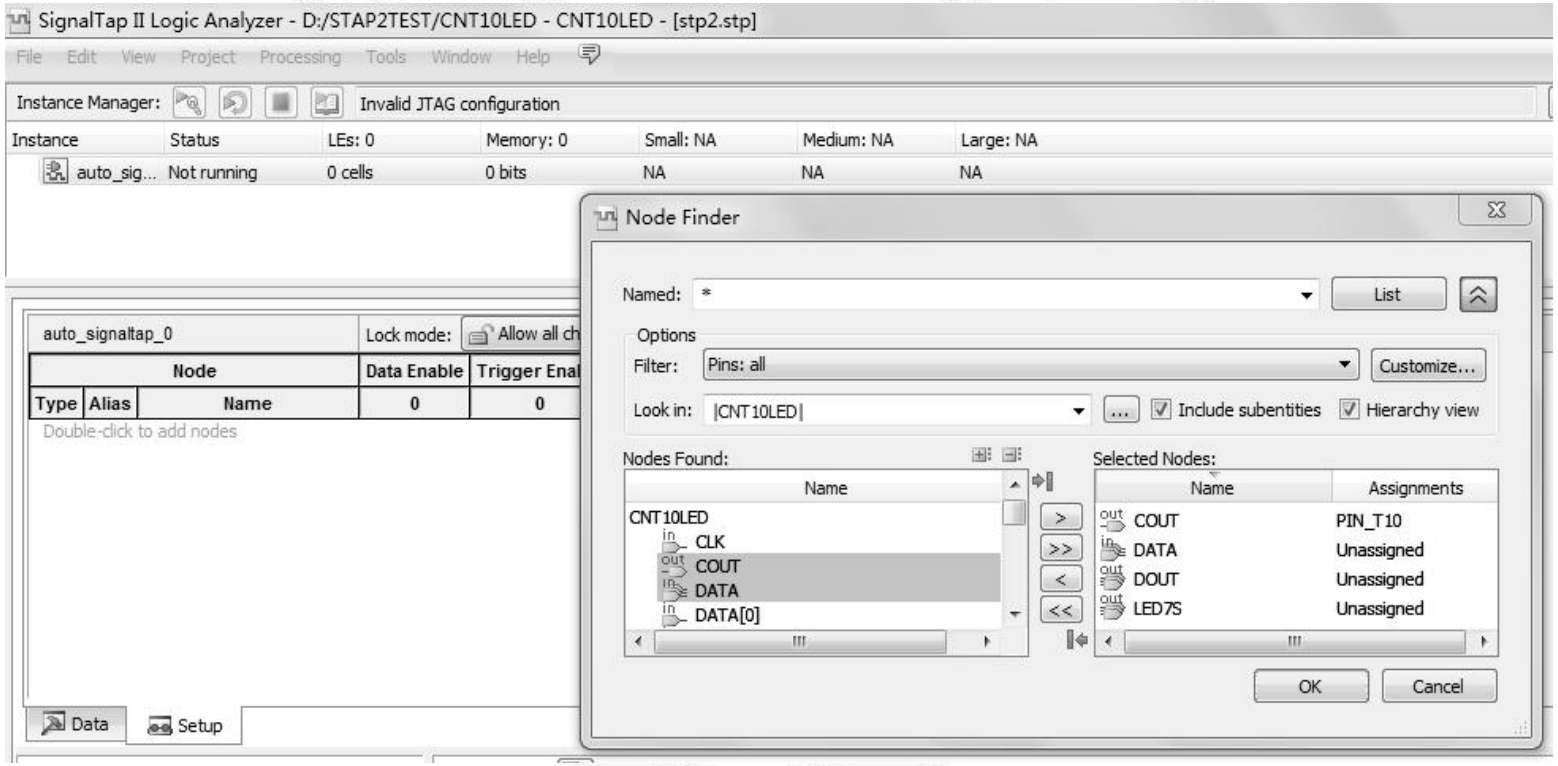
1. **打开SignalTap II编辑窗口**
2. **调入待测信号**



十进制计数器设计示例电路

5.6 SignalTap II的使用方法

3. SignalTap II参数设置

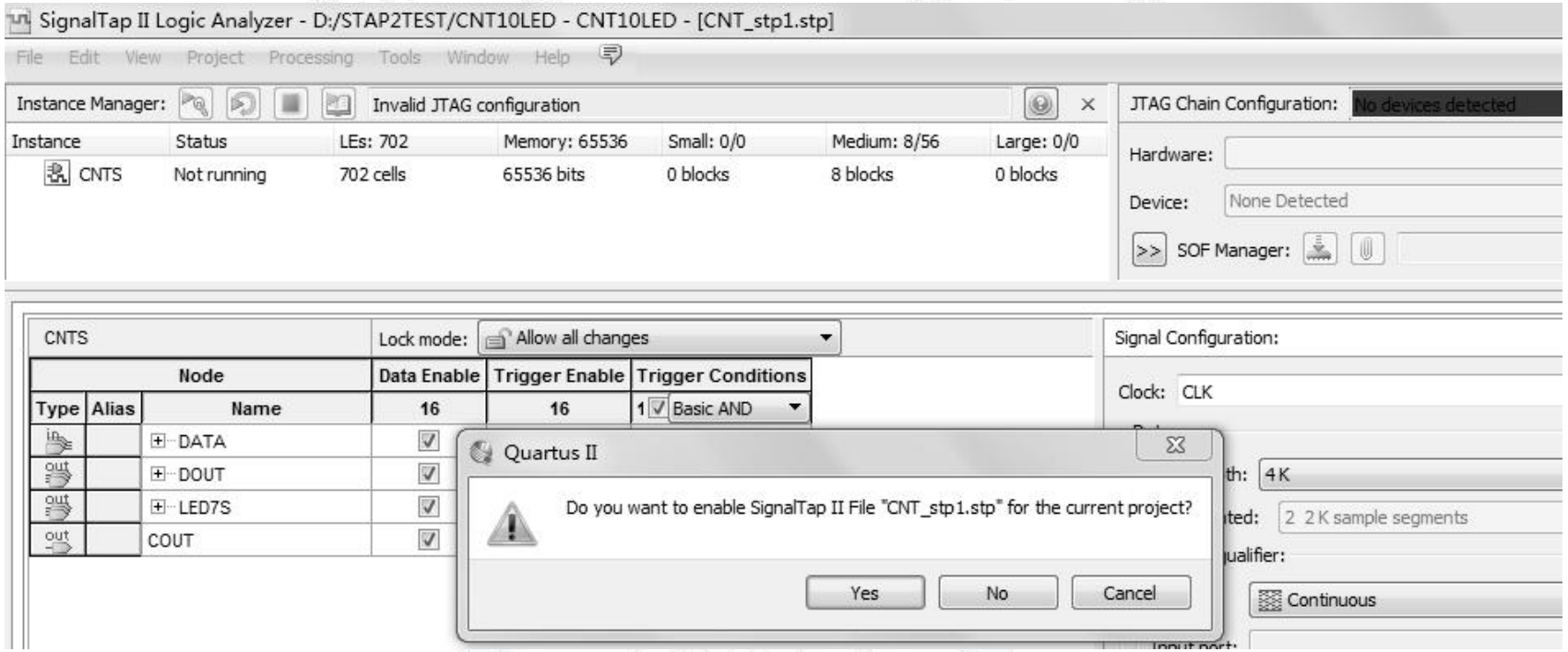


输入逻辑分析仪测试信号

EDA Lab

5.6 SignalTap II的使用方法

3. SignalTap II参数设置



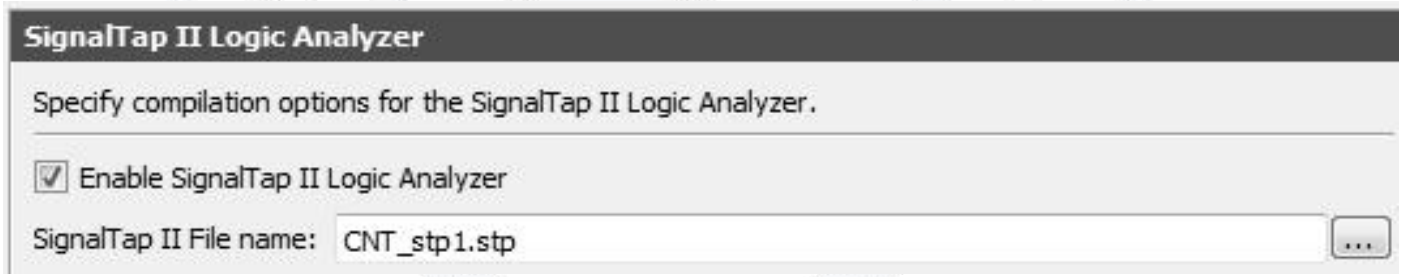
SignalTap II编辑窗口

5.6 SignalTap II的使用方法



4. 文件存盘

- File->Save As->Do you want to enable SignalTap II…->选择是，表示同意再次编译时将此SignalTap II文件与工程绑在一起综合/适配->如果选择否，则必须自己去设置，方法是选择Assignment->Settings->Category->SignalTap II Logic Analyzer
- 当利用SignalTap II将芯片中的信号全部测试结束后，不要忘了将SignalTap II的部件从芯片中除去。

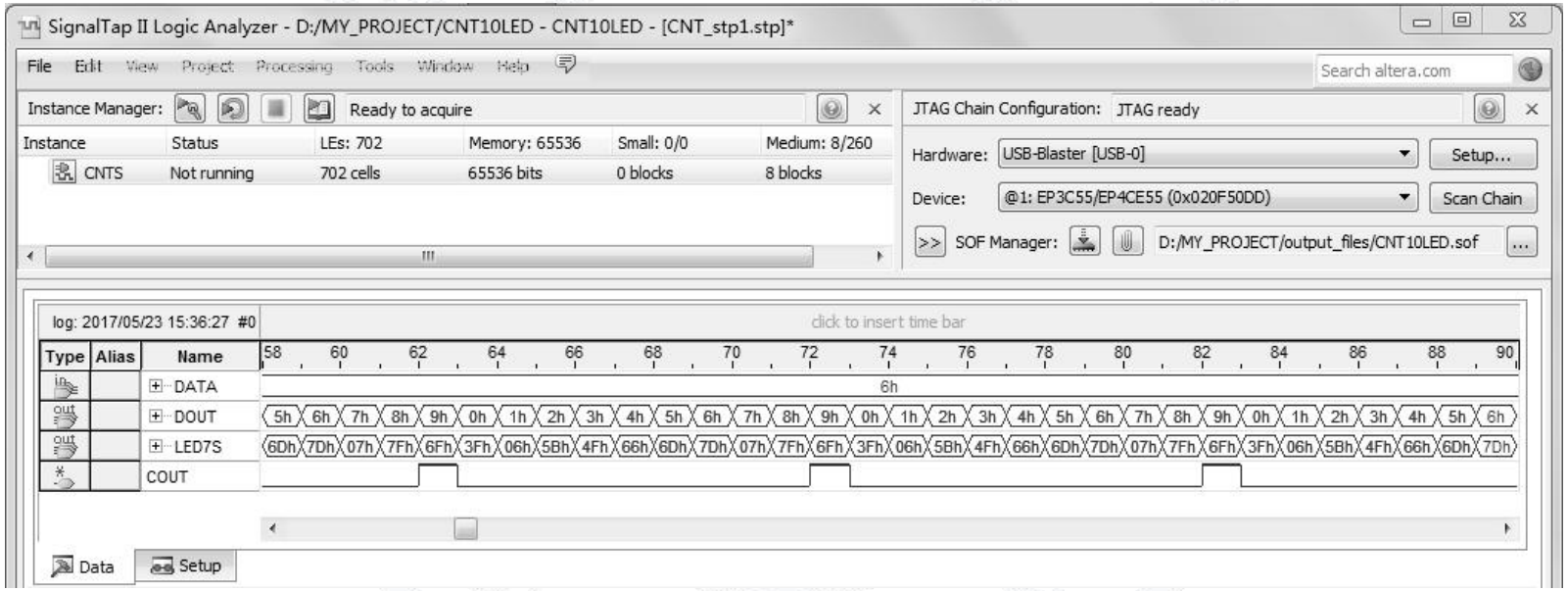


选择或删除SignalTap II文件加入综合编译

EDA Lab

5.6 SignalTap II的使用方法

5. 编译下载

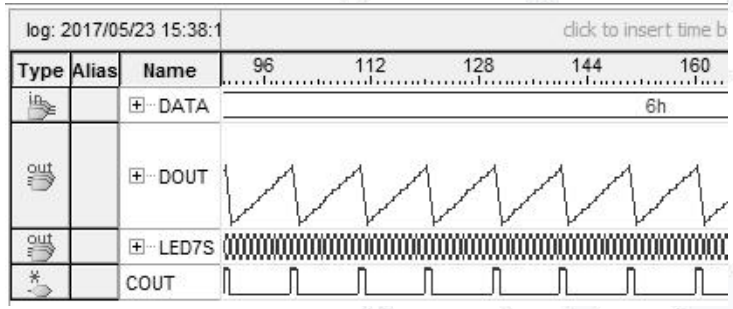


SignalTap II实时数据采样显示界面

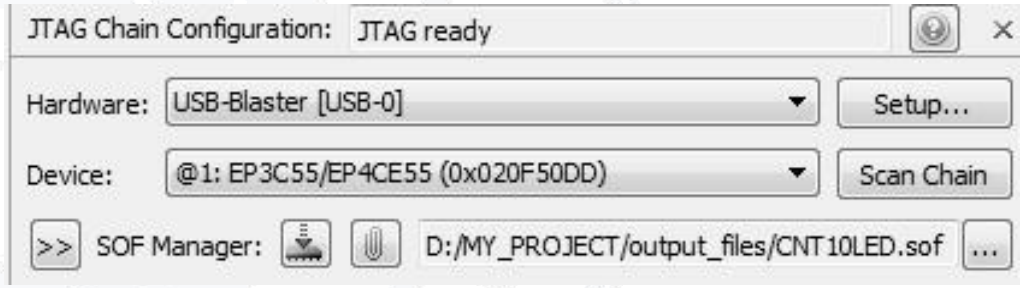
5.6 SignalTap II的使用方法

6. 启动SignalTap II进行采样与分析

单击“+”号可展开总线信号；或选择总线显示模式为Unsigned Line Chart，实现类似模拟波形显示



改变DOUT数据显示的方式

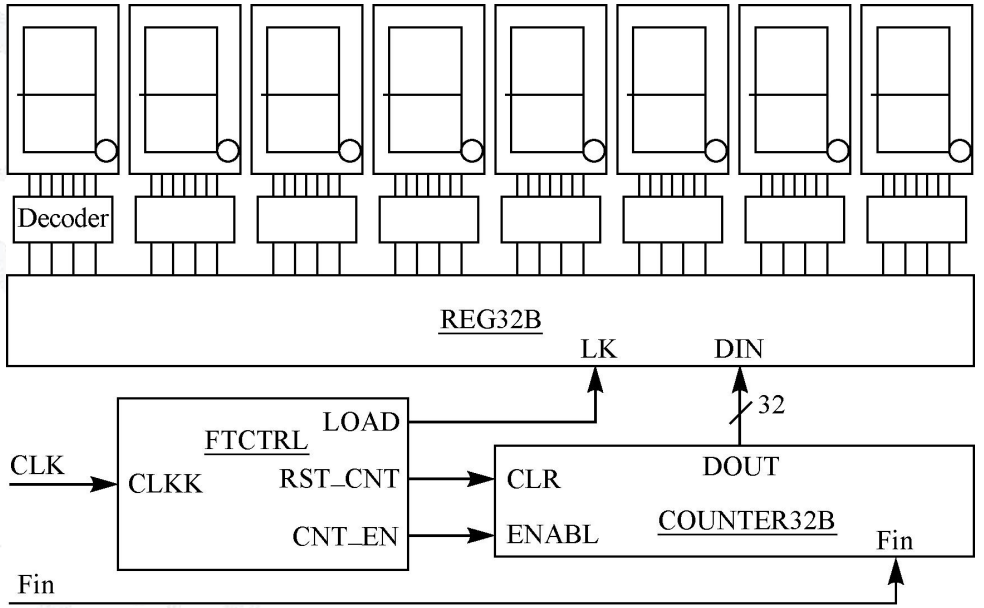


扫描FPGA，并下载SOF文件

```
input CLK    /* synthesis chip_pin = "G22" */ ;        //计数器工作时钟
input CLK0   /* synthesis chip_pin = "B11" */;         //逻辑分析仪采样时钟
```

实验与设计

- 5-1 计数器设计实验
- 5-2 数码扫描显示电路设计
- 5-3 高速硬件除法器设计
- 5-4 不同类型的移位寄存器设计
- 5-5 模可控计数器设计
- 5-6 移位相加型8位硬件乘法器设计
- 5-7 半整数与奇数分频器设计
- 5-8 基于Verilog代码的频率计设计



EDA Lab

5-9 VGA彩条信号显示控制电路设计

