

选课时间段: 周五 6-8 节

序号 (座位号): _____

杭州电子科技大学

实 验 报 告

课程名称: EDA 技术

实验名称: 8 位频率计和硬件消抖电路设计

指导老师: 岳克强

学生姓名: 黄继升

学生学号: 16041321

学生班级: 16040313

所学专业: 电子信息工程

实验日期: 2017.12.9

一.实验目的

1. 利用 Verilog 设计 8 位频率计
2. 学习去抖动硬件电路的设计与测试方法。

二.实验仪器设备或关键器材

1. Quartus II 软件
2. EDA 实验箱上的 FPGA 开发板

三.实验原理

1. 基于 Verilog 代码的频率计设计
 - (1) 根据频率的定义和频率测量的基本原理；
 - (2) 测定信号的频率必须有一个脉宽为 1s 的输入信号脉冲计数允许的信号；1s 计数结束后，计数值被锁入锁存器，计数器清 0，为下一测频计数周期做好准备；
 - (3) 将实验程序下载到 FPGA 开发板上进行测试。
2. 硬件消抖动电路设计

FPGA 中的硬件去抖动电路十分常用，在实际工程和实验设计中都会经常用到。程序中给出了一种去除双边沿抖动或毛刺的电路设计。它的主要原理是分别用两个计数器对输入信号的高电平和低电平的持续时间（脉宽）进行计数（在时间上是同时但独立计数）。只有当高电平的计数时间大于某值，则判为遇到正常信号，输出 1；若低电平的计数时间大于某值，则输出 0。

四.实验内容以及操作:

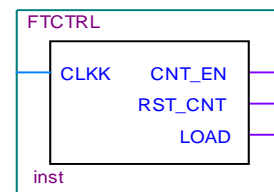
1. 基于 Verilog 代码的频率计设计

实验内容：测频控制信号可以由一个独立的发生器来产生，即如下的 FTCTRL 代码。设计要求 FTCTRL 的计数使能信号 CNT_EN 能产生一个 1s 脉宽的周期信号，并对频率计中的 32 位二进制计数器 COUNTER32B 的 ENABL 使能端进行同步控制。当 CNT_EN 高电平时允许计数，低电平时停止计数，并保持其所计的脉冲数。在停止计数期间，首先需要有一个锁存信号 LOAD 的上跳沿将计数器在前一秒钟的计数值锁存进锁存器 REG32B 中，并由外部的十六进制 7 段译码器译出，显示计数值。锁存信号后，必须有一清 0 信号 RST_CNT 对计数器清 0，为下一秒钟的计数操作做准备。

用 Verilog 设计另两个模块：REG32B 和 COUNTER32B，并对它们单独仿真测试。

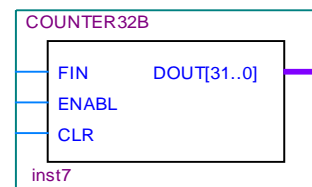
FTCTRL 程序代码：

```
1 module FTCTRL (CLKK, CNT_EN, RST_CNT, LOAD);
2     input CLKK;
3     output CNT_EN, RST_CNT, LOAD;
4     wire CNT_EN, LOAD;
5     reg RST_CNT, Div2CLK;
6     always @(posedge CLKK)
7         Div2CLK <= ~Div2CLK;
8     always @(CLKK or Div2CLK)
9     begin
10         if (CLKK <= 1'b0 & Div2CLK <= 1'b0)
11             RST_CNT <= 1'b1;
12         else
13             RST_CNT <= 1'b0;
14     end
15     assign LOAD = ~Div2CLK;
16     assign CNT_EN = Div2CLK;
17 endmodule
```

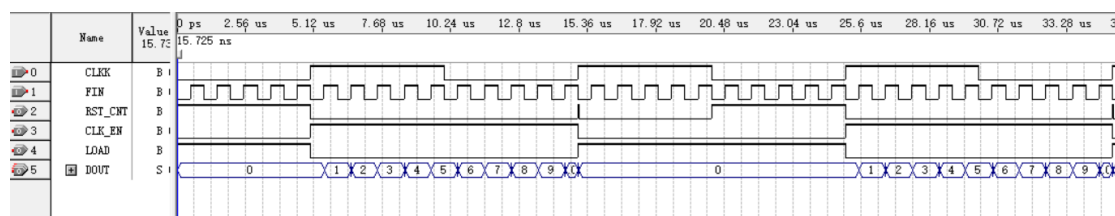
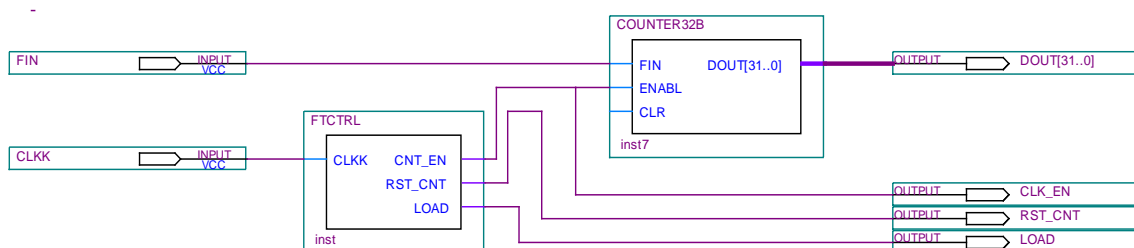


COUNTER32B 程序代码：

```
1 module COUNTER32B (FIN, ENABL, CLR, DOUT);
2     input FIN;
3     input CLR;
4     input ENABL;
5     output [31:0] DOUT;
6     reg [31:0] CQI;
7     always @(posedge FIN or posedge CLR )
8         if (CLR == 1'b1)
9             CQI <= {32{1'b0}};
10        else
11        begin
12            if (ENABL == 1'b1)
13                CQI <= CQI + 1;
14        end
15        assign DOUT = CQI;
16 endmodule
```



实验电路图:






实验内容: 根据原理说明, 具体设计, 仿真此消抖电路, 最后硬件实现此项设计。验证消抖的方法之一就是设计一个计数器来测试信号的抖动情况。此计数器的时钟端口由一有抖动的键输入, 此键中间加上一个去抖动电路, 按键后观察其计数情况。再比较当去除去抖动电路后的计数情况。

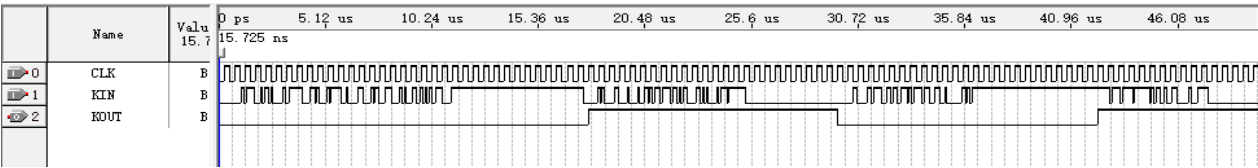
```

1  module ERZP (CLK, KIN,KOUT);
2      input CLK, KIN;
3      output KOUT; reg KOUT;
4      reg[3:0] KH,KL;
5      always @(posedge CLK) if(!KIN) KL<=KL+1;
6          else KL<=4'b0000;
7      always @(posedge CLK) if(KIN) KH<=KH+1;
8          else KH<=4'b0000;
9      always @(posedge CLK) begin
10         if(KH>4'b1100) KOUT<=1'B1;
11         else if(KL>4'b0111) KOUT<=1'B0;
12     end
13 endmodule

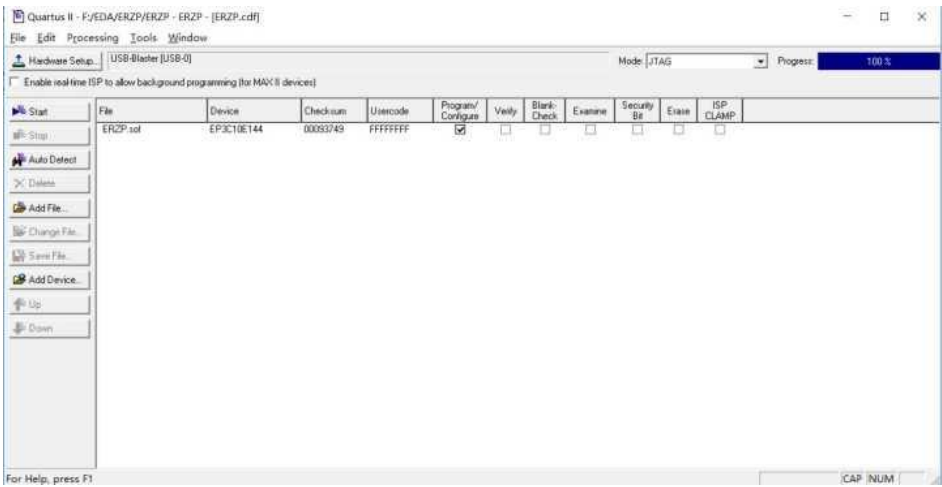
```

	Node Name	Direction	Location	I/O Bank	VREF Group	I/O Standard	Reserved
1	 CLK	Input	PIN_22	1	B1_N0	2.5 V (default)	
2	 KIN	Input	PIN_91	6	B6_N0	2.5 V (default)	
3	 KOUT	Output	PIN_52	3	B3_N0	2.5 V (default)	
4	<<new node>>						

新建波形图，设置 endtime 为 100μ s，CLK 设置为 1μ s，KIN 按如下设置，保存并输出波形图：



将 FPGA 和计算机相连，将程序下载到开发板上，如下所示：



最后在实验箱上实现功能，通过一抖动的按键来控制 LED 灯的来回闪烁，检测消抖电路的实现效果。

五.实验感想:

这次实验总的来说还是很容易的，因为老师简化了对频率计的设计，只是要求做出 FTCTRL 模块和 32 位计数器模块，并且通过连接实现电路的十分频。其实我事前已经做好了充分预习，并设计了 REG32B 32 位锁存器，顶层电路的代码设计，十六进制 7 段译码器的代码的设计，但是能不能在 FPGA 上实现出来就是另一回事了。关于消抖电路的设计也是仿照课本上的代码打的，锁定了引脚后在实验箱上的 FPGA 的开发板上也实现了功能，还是很容易的。但是事后一定要清楚理解消抖电路的代码设计原理，这样才能真正掌握对 Verilog 代码的使用和消抖电路的理解。