**《数字逻辑》实验报告**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **姓名** | | *黄婧婧，蔡欣彤* | | **年级** | | *2020级* |
| **学号** | | *20204237,20204187* | | **专业、班级** | | *计科02班，信安01班* |
| **实验名称** | **8.乘除法器的设计与实现** | | | | | |
| **实验时间** | **2021.11.18** | | **实验地点** | | **D1410** | |
| **实验成绩** |  | | **实验性质** | | **□验证性 □设计性 □综合性** | |
| 教师评价：  □算法/实验过程正确； □源程序/实验内容提交 □程序结构/实验步骤合理；  □实验结果正确； □语法、语义正确； □报告规范；  评语：  评价教师签名（电子签名）： | | | | | | |
| 一、实验目的  1、学习和掌握乘法器的设计与实现。  2、学习和掌握除法器的设计与实现。 | | | | | | |
| 二、实验项目内容  1．8 位乘法器的设计与实现。  2．8 位除法器的设计与实现。  3．显示模块的设计与实现。  4．顶层模块的设计与实现。 | | | | | | |
| 三、实验设计  1.实验原理  （1）乘法器    （2）除法器  类比十进制除法，利用移位运算。将被除数前8位拼接上8位的0，除数后8位拼接上8位的0。共循环8次，每次先将被除数左移一位，类比除法运算中的向后多取一位，将被除数与除数相比，若大于等于除数，则将被除数减去除数，并在被除数最后加上1，表示这一位整除得到1，在后续一位中，得到的1也会向前移动。循环完成后，被除数前8位为余数，后8位为商。  2.输入输出设计  使用三个开关m、a\_or\_b、show分别实现以下功能：  m=0: 乘法器  m=1: 除法器  a\_or\_b=0: 此时拨动开关，输入的为b的值  a\_or\_b=1: 此时拨动开关，输入的为a的值  show=0: 数码管上显示a、b的值  show=1: 数码管上显示结果的值。当为乘法器时，数码管以16进制显示乘法结果。当此时为除法器时，数码管的前两位以16进制显示商，后两位以16进制显示余数。 | | | | | | |
| 四、实验过程或算法(关键步骤、核心代码注解等）  1.乘法器模块  module multiply(  input [7:0] a, // 输入a  input [7:0] b, //输入b  input flag, //开始信号  output [15:0] ans //结果输出  );  reg [15:0] tempa; //储存a，方便进行数据的向左移位  reg [15:0] tempb;  reg [15:0] ans;    integer i;  always@(\*)  if(flag) begin //得到开始信号，进行计算  begin  tempa = a; //赋值  tempb = b;  ans = 16'b0;  for(i=0;i<8;i=i+1) begin //8位乘法器，进行8次循环  if(tempb[0]) //判断tempb的最后一位是否为1，如果为1，则进行乘法运算，即给答案加上此时tempa的值  ans=ans+tempa;  tempa=tempa << 1'b1; //tempa向左移动1位  tempb=tempb >> 1'b1; //tempb向右移动1位  end  end  end  else begin //没有开始信号，输出0  ans=15'h0000;  end    endmodule  2.除法器模块  module division(  input [7:0] a,  input [7:0] b,  input flag,  output [7:0] quotient, //商  output [7:0] remainder //余数  );  reg [15:0] tempa; //储存a，方便进行移位  reg [15:0] tempb;  reg [7:0] quotient;  reg [7:0] remainder;    integer i;  always@(\*)  if(b==0) begin //除数为0，输出ffff  quotient=8'hff;  remainder=8'hff;  end  else if(flag) begin  tempa=a;  tempb={b,8'h00};  for(i=0;i<8;i=i+1) begin //8位除法器，循环8次  tempa=tempa << 1'b1; //tempa向左移动一位，模拟除法计算中向后多取一位  if(tempa>=tempb) //tempa>=tempb则进行除法运算，即tempa-tempb+1  tempa=tempa-tempb+1; //加1是因为tempa最后一位为0，之后tempa向左移位时，加上的1也会向左移位，模拟商的运算  else  tempa=tempa;  end  quotient=tempa[7:0]; //商  remainder=tempa[15:8]; //余数  end  else begin //没有开始信号，输出0  quotient=8'h00;  remainder=8'h00;  end  endmodule  3.数码管显示模块  module hexsegs(  input clk,  input [15:0] s,  output reg [3:0] an,  output reg [6:0] segs  );  reg [20:0] count=0; //计数，实现降频功能  always@(posedge clk)  begin  count=count+1;  end    reg [3:0] data=4'b0000; //显示的数值  initial begin  an=4'b1110;  end    always@(posedge clk) //段控制  case(count[20:19]) //通过count来控制现在输出的数据和位置  2'b00:  begin an=4'b1101; data=s[7:4];end  2'b01:  begin an=4'b1011; data=s[11:8];end  2'b10:  begin an=4'b0111; data=s[15:12];end  2'b11:  begin an=4'b1110; data=s[3:0];end  endcase    always@(posedge clk) //位控制  case(data) //将data的值显示在数码管上  4'h0: segs=7'b000\_0001; //显示0  4'h1: segs=7'b100\_1111; //显示1，下同  4'h2: segs=7'b001\_0010;  4'h3: segs=7'b000\_0110;  4'h4: segs=7'b100\_1100;  4'h5: segs=7'b010\_0100;  4'h6: segs=7'b010\_0000;  4'h7: segs=7'b000\_1111;  4'h8: segs=7'b000\_0000;  4'h9: segs=7'b000\_0100;  4'ha: segs=7'b000\_1000;  4'hb: segs=7'b110\_0000;  4'hc: segs=7'b111\_0010;  4'hd: segs=7'b100\_0010;  4'he: segs=7'b011\_0000;  4'hf: segs=7'b011\_1000;  default:  segs=7'b111\_1111;  endcase  endmodule  4.顶层模块  module top(  input clk,  input m,  input a\_or\_b, //选择输入a或b  input show, //选择显示a，b的值还是结果的值  input [7:0] num, //输入数据  output [6:0] segs,  output [3:0] an  );  reg [7:0] a=8'h00;  reg [7:0] b=8'h00;  reg [15:0] s;  wire [15:0] mul\_ans; //乘法运算结果  wire [7:0] q; //除法的商  wire [7:0] r; //除法的余数    always@(\*)  if(a\_or\_b) //选择输入a还是b的值  a=num;  else  b=num;    always@(\*)  case({m,show}) //m=0为除法器，m=1为除法器，show=0显示ab的值，show=1显示结果的值  2'b00:s={a,b};  2'b01:s=mul\_ans;  2'b10:s={a,b};  2'b11:s={q,r};  endcase    multiply u0(a,b,~m,mul\_ans);  division u1(a,b,m,q,r);  hexsegs u2(clk,s,an,segs);  endmodule | | | | | | |
| 五、实验过程中遇到的问题及解决情况    按照提示在约束文件中加入  set\_property CLOCK\_DEDICATED\_ROUTE FALSE [get\_nets a\_or\_b\_IBUF]  生成比特流成功 | | | | | | |
| 六、实验结果及分析和（或）源程序调试过程  1.乘法器仿真  module multiply\_sim(  );  reg [7:0] a=8'h2a;  reg [7:0] b=8'h02;  reg flag=0;  wire [15:0] ans;    multiply u(a,b,flag,ans);    initial begin  #100 flag=1;  #100 a=8'h01;  #100 a=8'hff;  #100 b=8'hff;  end    endmodule    2.除法器仿真  module divisionsim(  );  reg [7:0] a=8'h2a;  reg [7:0] b=8'h02;  reg flag=0;  wire [7:0] q;  wire [7:0] r;    division u(a,b,flag,q,r);    initial begin  #100 flag=1;  #100 a=8'h01;  #100 a=8'hff;  #100 b=8'hff;  #100 b=8'h00;  end  endmodule    3.顶层模块仿真  module top\_sim(  );  reg clk=0;  reg m=0;  reg a\_or\_b=0;  reg show=0;  reg [7:0] num=8'h0a;  wire [6:0] segs;  wire [3:0] an;    always #20 clk=~clk;  top u(clk,m,a\_or\_b,show,num,segs,an);    initial begin  #100 a\_or\_b=1;  #20 num=8'h03;  #100 show=1;  #100 begin m=1;show=0;end  #100 a\_or\_b=1;  #20 num=8'hff;  #100 a\_or\_b=0;  #20 num=8'hfc;  end  endmodule | | | | | | |
| 七、小组分工情况说明  组内成员独立完成实验，遇到问题讨论交流。  本次实验报告由黄婧婧完成。 | | | | | | |