

（深圳）

实验报告

开课学期： 2021秋季

课程名称：数字逻辑设计（实验）

实验名称： 十六进制计算器设计

实验性质： 综合设计型

实验学时： 6 地点： t2506

学生班级： 6班

学生学号： 200110618

学生姓名： 邓皓元

评阅教师：

报告成绩：

实验与创新实践教育中心制

2021年12月

注：本设计报告中各个部分如果页数不够，请大家自行扩页，原则是一定要把报告写详细，能说明设计的成果和特色。报告中应该叙述设计中的每个模块。设计报告将是评定每个人成绩的重要组成部分（**设计内容及报告写作**都作为评分依据）。

|  |
| --- |
| 设计的功能描述 |
| 概述基本功能、详细描述自行扩展的功能  基本功能：一个十六进制的计算器，支持加、减、乘、求商，求余，平方六种运算，且支持连续运算功能；  按键开关S1作为异步复位信号，且当S1为1时，计算器将被复位；  按键开关S2作为每次计算启动信号（类似等于号功能）；  操作数由拨码开关SW15-SW0输入，操作数1为SW15-SW8，操作数2为SW7-SW0；  连续运算时，输入的数据为操作数2；（平方运算时，为上一次计算结果）  当前运算功能由拨码开关SW23-SW21来决定：  s2-1  计算结果实时输出到数码管上； |
| 系统功能详细设计 |
| 用硬件框图描述系统主要功能及各模块之间的相互关系  屏幕截图 2021-12-20 120338  clk\_div(输入clk，输出clk\_g):将输入的clk分频成clk\_g，输出clk\_g；  calculator\_hex(输入clk\_g、rst、button、func、num1、num2，输出cal\_result)：利用输入的clk\_g、rst、button、func、num1、num2，计算出cal\_result；  calculator\_display(输入clk\_g、rst、button、cal\_result，输出led\_en、led\_cx)：利用输入的clk\_g、rst、button、cal\_result，输出led\_en、led\_cx；将cal\_result在数码管上显示，实现计算器结果在数码管上显示功能； |
| 各模块描述 |
| 包括模块功能，输入、输出端口、变量含义及主要设计代码  clk\_div(输入clk，输出clk\_g):将输入的100MHz clk分频成10MHz clk\_g，用于时钟分频（必须），且分频的频率为10MHz；  calculator\_hex(输入clk\_g、rst、button、func、num1、num2，输出cal\_result)：利用输入的clk\_g、rst、button、func、num1、num2，计算出cal\_result；实现计算器计算功能；  变量含义：  reg start；用以表示是否是第一次计算  //用于消抖的变量  reg another\_button;//输出的消抖后的button信号  reg t1;//刚开始变化的信号  reg t2;//15ms之后的信号  reg judge;//判断条件，用以控制计数器的开始  reg [31:0] cnt;//计数器  reg [31:0] cal;  reg flag;  主要设计代码：  always@(posedge clk\_g or negedge rst\_n)  begin  if(~rst\_n)  begin  cal\_result<=32'd0;  start<=1'd1;  end  else if(button==1'd1 && start==1'd1)//若开发板验证则把button==1'd1改成cal==32'd1  begin  case(func)  3'b000:cal\_result<=num1+num2;  3'b001:cal\_result<=num1-num2;  3'b010:cal\_result<=num1\*num2;  3'b011:cal\_result<=num1/num2;  3'b100:cal\_result<=num1%num2;  default:;  endcase  start<=1'd0;  end  else if(button==1'd1 && start==1'd0)//若开发板验证则把button==1'd1改成cal==32'd1  begin  case(func)  3'b000:cal\_result<=cal\_result+num2;  3'b001:cal\_result<=cal\_result-num2;  3'b010:cal\_result<=cal\_result\*num2;  3'b011:cal\_result<=cal\_result/num2;  3'b100:cal\_result<=cal\_result%num2;  3'b101:cal\_result<=cal\_result\*cal\_result;  default:;  endcase  end  end  //消抖模块  always@(posedge clk\_g or negedge rst\_n)  begin  if(~rst\_n)  begin  another\_button<=1'd0;  t1<=1'd0;  t2<=1'd0;  judge<=1'd0;  cnt<=32'd0;  end  else  begin  if(judge==1'd1)  begin  if(cnt==cnt\_max)  begin  cnt<=32'd0;  t2=button;  judge<=1'd0;  if(t2==t1)  another\_button<=button;  else;  end  else cnt<=cnt+32'd1;  end  else if(button!=t1)  begin  judge<=1'd1;  t1=button;  end  else another\_button=button;  end  end  //cal计数器利用消抖后的another\_button  always@(posedge clk\_g or negedge rst\_n)  begin  if(~rst\_n)  begin  cal<=32'd0;  flag<=1'd0;  end  else if(another\_button==1'd1)  begin  cal<=32'd0;  flag<=1'd1;  end  else if(cal==cal\_max)  begin  cal<=32'd0;  flag<=1'd0;  end  else if(flag==1'd1) cal<=cal+32'd1;  end  endmodule |
| 调试报告 |
| 仿真波形截图及仿真分析  calculator\_hex的波形如下：  屏幕截图 2021-12-20 171959  波形分析：  sequence\_detection 模块完成的是设计一个十六进制的计算器，支持加、减、乘、求商，求余，平方六种运算，且支持连续运算的功能  故需要关注输入的clk\_g、rst、button、func、num1、num2和输出的cal\_result、led\_en、led\_cx。  从上述波形可以看出，在复位信号 rst\_n 撤销后：  在7195ns时，func为0(000)、num1为06、num2为04、在button按下的一个周期之后之后，cal\_result输出正确的结果6+4=10(a)，符合预期；  在8795ns时，func为2(010)、num1为00、num2为0c、在button按下的一个周期之后之后，cal\_result输出正确的结果10(a)\*12(c)=120(78)，符合预期；  在10395ns时，func为5(101)、num1为00、num2为05、在button按下的一个周期之后之后，cal\_result输出正确的结果120(78)\*120(78)=14400(3840)，符合预期；  在11995ns时，func为1(001)、num1为00、num2为c8、在button按下的一个周期之后之后，cal\_result输出正确的结果14400(3840)-200(c8)=14200(3778)，符合预期；  在13595ns时，func为3(011)、num1为00、num2为08、在button按下的一个周期之后之后，cal\_result输出正确的结果14200(3778)/8=1775(6ef)，符合预期；  在15195ns时，func为5(101)、num1为00、num2为00、在button按下的一个周期之后之后，cal\_result输出正确的结果1775(6ef)\*1775(6ef)=3150625(301321)，符合预期；  led\_en和led\_cx的输出则直接通过观察开发板上的显示结果进行检验  故根据上述分析，sequence\_detection 模块完成了设计一个十六进制的计算器，支持加、减、乘、求商，求余，平方六种运算，且支持连续运算的功能 |
| 设计过程中遇到的问题及解决方法 |
| 在支持加、减、乘、求商，求余，平方六种运算，且支持连续运算的功能的十六进制计算器的设计中，我遇到了关于button开关抖动的问题button按钮在按下和松开的一瞬间button信号将会进行难以预测的抖动问题，这将会导致开发板上的计算出错误的结果，因此需要进行消抖。处理思路如下图所示（询问了助教）：  20201218185522944  为了达到消抖后的效果，我设计了一个基于延时进行消抖的模块（延时15ms），这个模块检测了在button信号发生改变之后，经过15ms间隔，button信号是否仍为改变的状态，若是则another\_button赋值为此时的button信号，若否则another\_button保持不变。  有了这个模块之后，把another\_button信号替代button信号作为开发板使用的计算启动信号来进行上板验证，得到了正确的二极管输出结果，开发板验证成功！（在这里谢谢四班刘洋冰助教小哥） |
| 课程设计总结 |
| 包括设计的总结和还需改进的内容以及收获  在这次支持加、减、乘、求商，求余，平方六种运算，且支持连续运算的功能的十六进制计算器的设计实验中，我巩固了时钟IP以及数码管控制器的使用。  calculator\_top囊括了div\_clk、calculator\_hex、calculator\_display三个模块，其中calculator\_display沿用了实验四中数码管控制器的内容进行了计算结果的数码管显示，calculator\_hex则是在每次button（开发板为another\_button）启动信号按下之后对是否是第一次计算进行判断（start是否为1，若为1则是第一次、若为0则不是第一次，且通过rst复位按钮重置为1），之后根据func、num1、num2计算出cal\_result。  在这次实验中，我巩固了verilog语言的运用知识、提高了语言使用规范性，对vivado软件的操作更加得心应手。除此之外，我对于verilog语言的调试能力和排错能力也有很大的提升，对于回避一些容易出现的问题也有了一定的心得和体会。最后，我认为数字逻辑设计实验课的重要性是无可比拟的，verilog这门逻辑语言在课上学到的知识完全不足以让我写出真正逻辑正确且正常运行的代码，实验不仅提高了我的动手能力，更提高了我对于verilog这门语言的理解，让我对数字逻辑的知识有了更加深刻的认识。  verilog真是一门有意思又让人捉摸不透的语言，其中的奥妙选哟我花更多的时间去探求。数字世界，精彩无限！ |