

浙江大学

本科实验报告

课程名称:	计算机组成
姓 名:	姜雨童
学 院:	计算机科学与技术学院
专 业:	计算机科学与技术
邮 箱:	3220103450@zju.edu.cn
QQ 号:	1369218489
电 话:	19550103468
指导教师:	马德
报告日期:	2024 年 4 月 7 日

浙江大学实验报告

课程名称： 计算机组成 实验类型： 综合

实验项目名称： Lab 3: 复杂操作实现

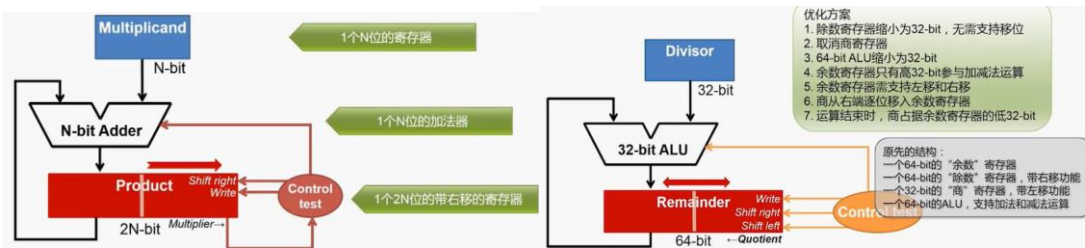
学生姓名： 姜雨童 学号： 33220103450 同组学生姓名： /

实验地点： 紫金港东四 509 室 实验日期： 2024 年 4 月 3 日

一、操作方法与实验步骤

实验目的：

设计实现 32 位乘法器和除法器，实现结构如下图所示



实验步骤：

Testbench 文件已经给出，因此只需要完成.v 文件的 Verilog 代码并进行仿真测试即可。

我的代码如下所示（debug 部分在模块三——讨论与心得给出）：

（备注：32 位乘法器模块并没有使用优化后的算法。）

```
module mul32(  
    input clk,  
    input rst,  
    input [31:0] multiplicand,  
    input [31:0] multiplier,  
    input start,  
    output reg [63:0] product,  
    output reg finish  
);  
  
reg [5:0] counter;  
reg [63:0] op1;  
reg [63:0] op2;  
  
always @(posedge clk or posedge rst) begin
```

```

    if (start) begin
        finish <= 0;
        product <= 0;
        counter <= 0;
        op1 <= {32'b0, multiplicand};
        op2 <= {32'b0, multiplier};
    end
    else if (counter < 32) begin
        if(op2[0] == 1)
            product <= product + op1;
        op1 <= {op1[62:0],1'b0};
        op2 <= {1'b0, op2[63:1]};
        counter <= counter + 1;
    end
    else begin
        finish <= 1;
    end
end

endmodule

```

```

module div32(
    input clk,
    input rst,
    input start,
    input [31:0] dividend,
    input [31:0] divisor,
    output finish,
    output [31:0] quotient,
    output [31:0] remainder
);
    reg [63:0] op;
    reg [5:0] counter;
    reg done;

    always @(posedge clk or posedge rst) begin
        if (rst) begin
            counter <= 0;
            done <= 0;
            op <= 0;
        end
        if(start) begin
            done <= 0;
            counter <= 0;

```

```

        op <= {32'h0,dividend};
    end
    else if(counter < 6'd32) begin
        counter <= counter + 1'b1;
        op = op << 1;
        if(op[63:32] >= divisor) begin
            op[63:32] = op[63:32] - divisor[31:0];
            op[0] <= 1;
        end else
            op[0] <= 0;
        end
    end
    else done <= 1'b1;
end

assign finish = done;
assign quotient = op[31:0];
assign remainder = op[63:32];

endmodule

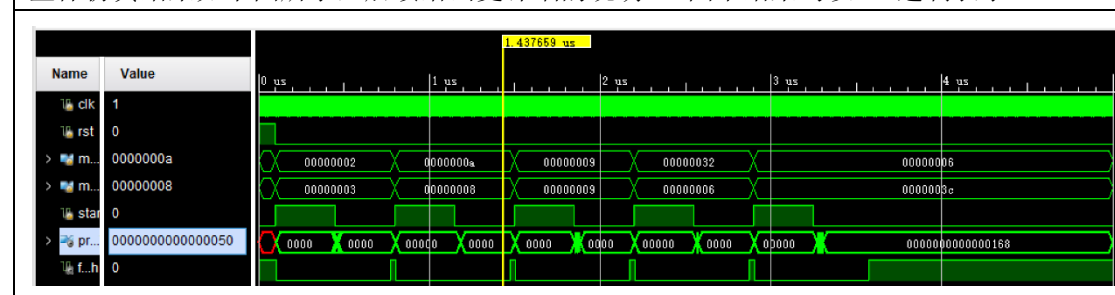
```

随后进行仿真测试，这部分结果在模块二——实验结果与分析中给出。

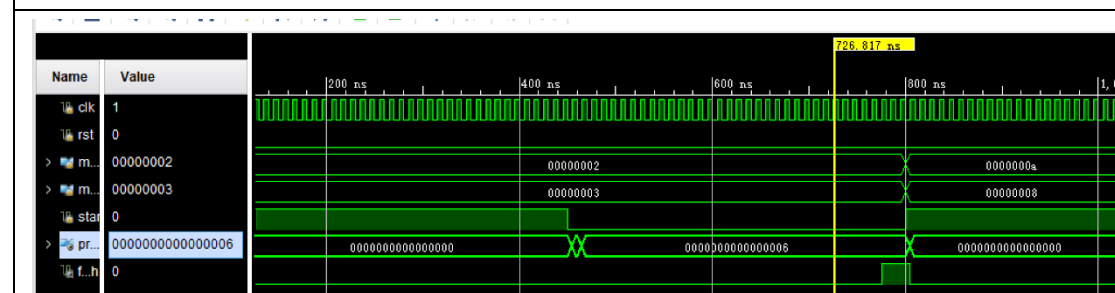
二、实验结果与分析

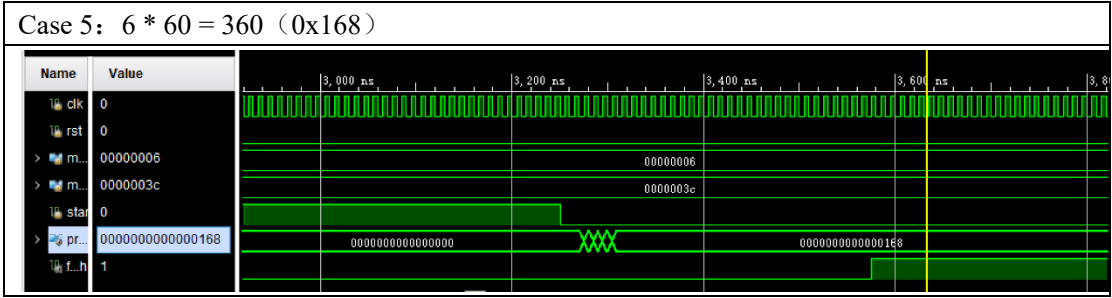
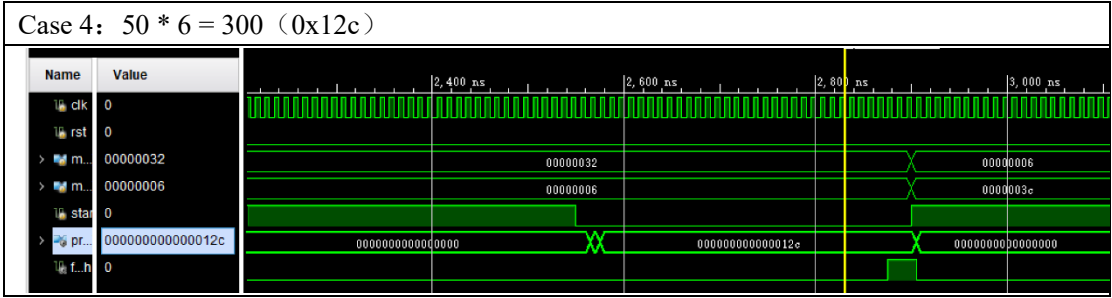
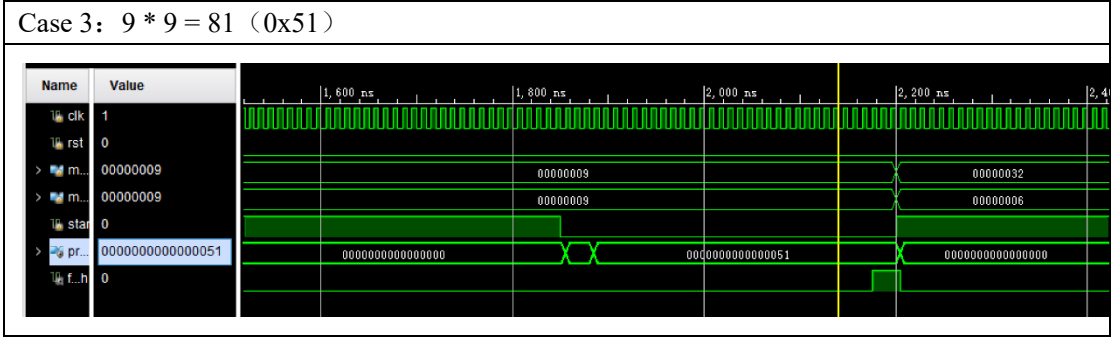
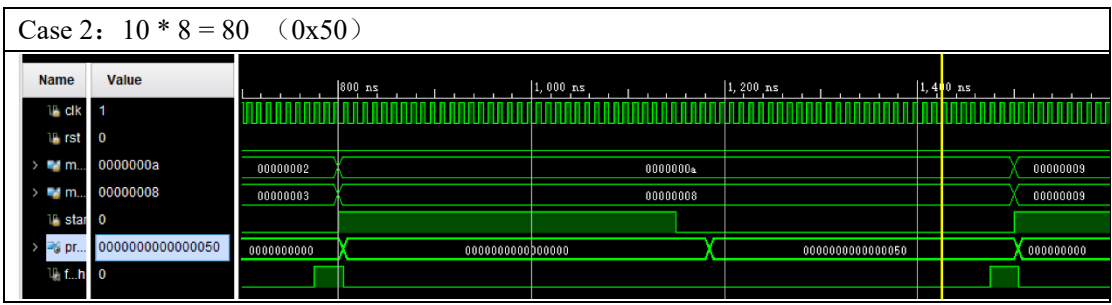
Mul32 仿真结果:

整体仿真结果如下图所示，后续给出更详细的说明。（图中结果均以 16 进制表示）

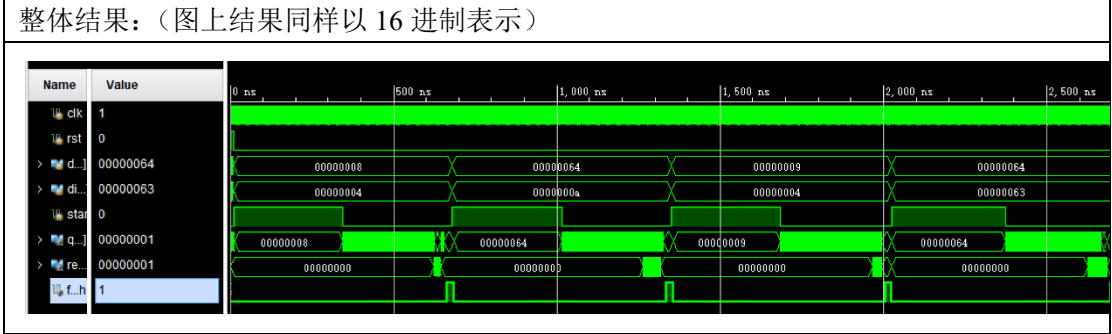


Case 1: $2 * 3 = 6$ (0x6)

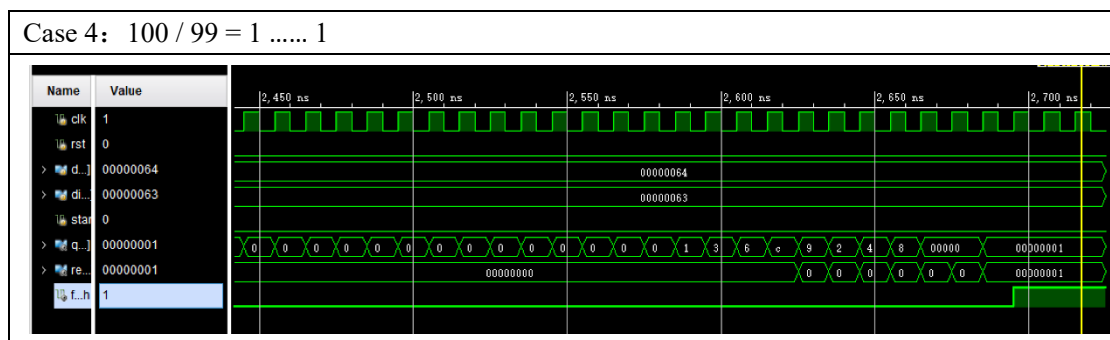
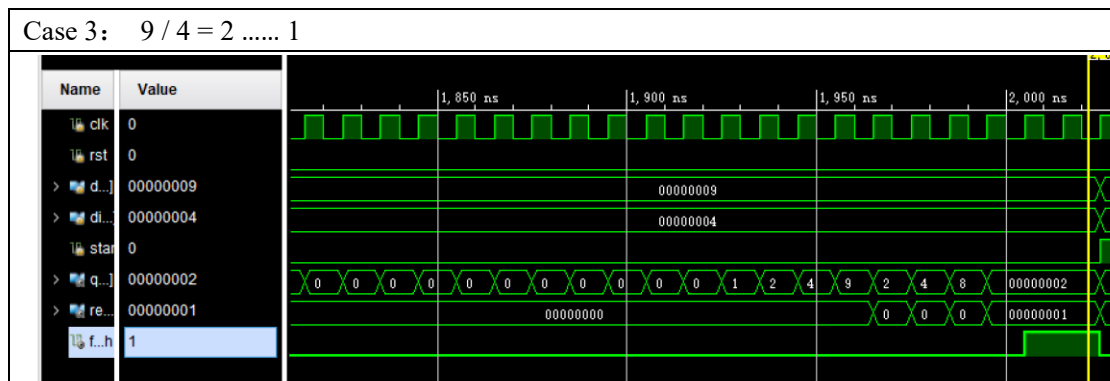
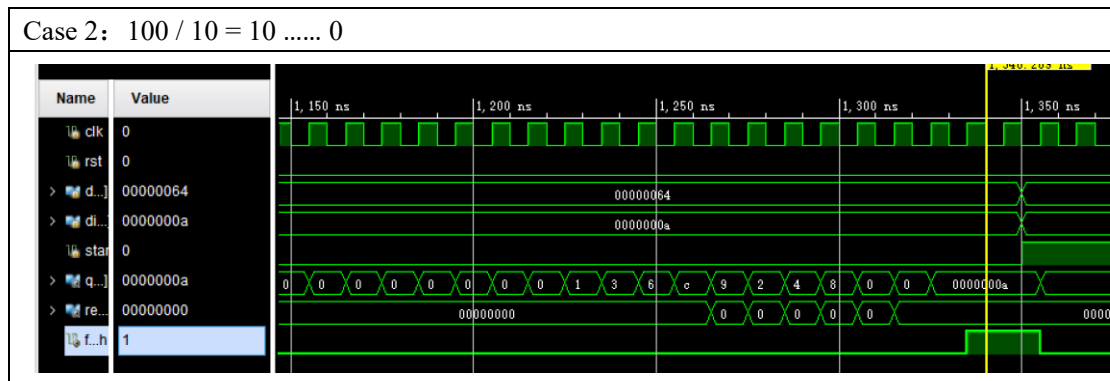
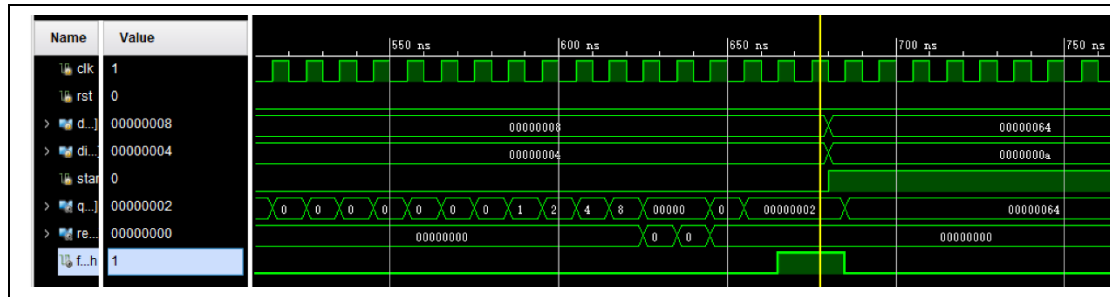




Div32 仿真结果:



Case 1: $8 / 4 = 2 \dots 0$



三、讨论、心得

心得:

因为只涉及到了.v文件的编写, 因此感觉 lab3 更考察对乘法器、除法器的理解, 以及 verilog

