

# 实验报告

2021 年 4 月 30 日

成绩: \_\_\_\_\_

姓名	*****	学号	*****	班级	*****
专业	计算机科学与技术		课程名称	计算机组成原理课程设计	
任课老师	曾虹	指导老师	曾虹	机位号	31
实验序号	4	实验名称	寄存器堆设计实验		
实验时间	2021.4.30	实验地点	一教 225	实验设备号	31

## 一、实验程序源代码

```
//ALU.v
`timescale 1ns / 1ps

module ALU(
    input [2:0] OP,
    input [31:0] A,
    input [31:0] B,

    output reg ZF, //零值

    output reg OF, //溢出

    output reg [31:0] F
);
    reg CF;
    always@(*)
    begin
        CF = 0;
        case(OP)
            3'b000: begin F = A & B; end
            3'b001: begin F = A | B; end
            3'b010: begin F = A ^ B; end
            3'b011: begin F = ~(A | B); end
            3'b100: begin {CF, F} = A + B; end
            3'b101: begin {CF, F} = A - B; end
            3'b110: begin F = A < B; end
            3'b111: begin F = B << A; end
        endcase
        ZF = F == 0;
        OF = A[31] ^ B[31] ^ F[31] ^ CF;
    end
endmodule
```

```
    end  
endmodule
```

```
//REG.v
```

```
`timescale 1ns / 1ps
```

```
module REG(  
    input [4:0] R_Addr_A,  
    input [4:0] R_Addr_B,  
    input [4:0] W_Addr,  
    input [31:0] W_Data,  
    input Write_Reg,  
    input Reset,  
    input CLK,  
    output [31:0] R_Data_A,  
    output [31:0] R_Data_B  
);  
    integer i;  
    reg [31:0] REG_Files[0:31];  
    initial REG_Files[0]=32'h 0011_4514;  
    initial REG_Files[1]=32'h 0191_9810;  
    initial REG_Files[2]=32'h 2333_3333;  
  
    assign R_Data_A = REG_Files[R_Addr_A];  
    assign R_Data_B = REG_Files[R_Addr_B];  
  
    always@(posedge CLK or posedge Reset)  
    begin  
        if(Reset)  
            for(i = 0; i < 32; i = i + 1)  
                REG_Files[i] = 0;  
        else  
            if (Write_Reg)  
                REG_Files[W_Addr] = W_Data;  
        end  
    end  
endmodule
```

```
//REG_ALU.v
```

```
`timescale 1ns / 1ps
```

```
module REG_ALU(  
    input CLK,  
    input Reset,  
    input [4:0]R_Addr_A,
```

```

input [4:0]R_Addr_B,
input [4:0]W_Addr,
    input Write_Reg,
input [2:0] OP,
output ZF,
output OF,
output [31:0]F,
    output [31:0]R_Data_A,
output [31:0]R_Data_B
);

REG r(R_Addr_A,R_Addr_B,W_Addr,F,Write_Reg,
      Reset,CLK,R_Data_A,R_Data_B);
ALU a(OP,R_Data_A,R_Data_B,ZF,OF,F);
endmodule

```

## 二、仿真测试代码

```

`timescale 1ns / 1ps
module Test;
    reg CLK;
    reg Reset;
    reg [4:0] R_Addr_A;
    reg [4:0] R_Addr_B;
    reg [4:0] W_Addr;
    reg Write_Reg;
    reg [2:0] OP;

    wire ZF;
    wire OF;
    wire [31:0] F;
    wire [31:0] R_Data_A;
    wire [31:0] R_Data_B;

    REG_ALU uut (
        .CLK(CLK),
        .Reset(Reset),
        .R_Addr_A(R_Addr_A),
        .R_Addr_B(R_Addr_B),
        .W_Addr(W_Addr),
        .Write_Reg(Write_Reg),
        .OP(OP),
        .ZF(ZF),
        .OF(OF),
        .F(F),

```

```

        .R_Data_A(R_Data_A),
        .R_Data_B(R_Data_B)
    );
    integer i;

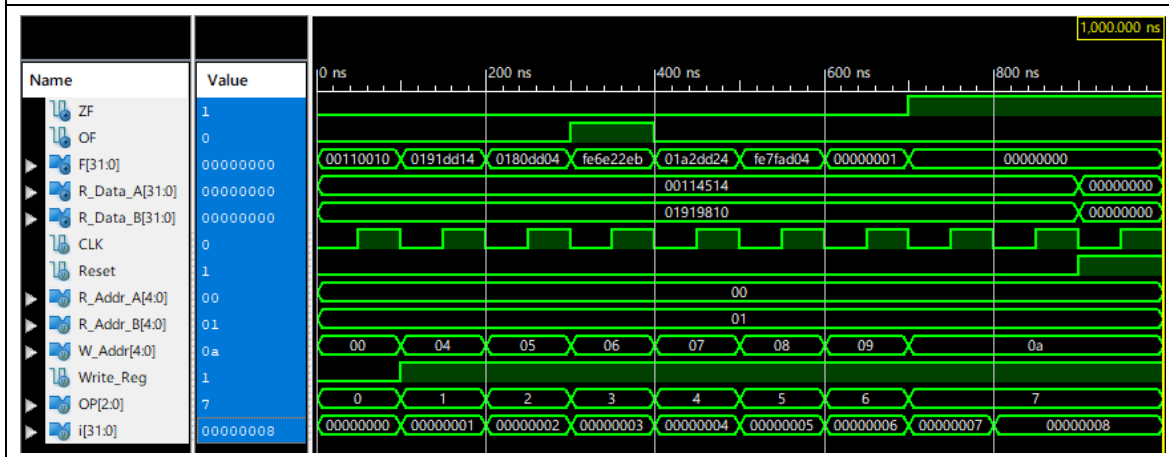
    initial begin
        // Initialize Inputs
        CLK = 0;
        Reset = 0;
        R_Addr_A = 0;
        R_Addr_B = 5'b00001;
        W_Addr = 0;
        Write_Reg = 0;
        OP = 0;
        i = 0;

        #100;
        Write_Reg = 1;
        for(i = 1; i < 8; i = i + 1)
        begin
            OP = i;
            W_Addr = i + 3;
            #100;
        end
        #100;
        Reset=1;

    end
    always #50 CLK=~CLK;
endmodule

```

### 三、仿真波形



#### 四、思考与探索

测试文件中进行了对 ALU 与 REG 的调试，REG 中预置了几个数据，ALU 将直接调用他们进行运算，并且把结果直接写回寄存器中，经确认，寄存器中的结果无误。

本次实验主要卡住的地方是，在寄存器与 ALU 级联的时候，在 ALU 的输出端 ZF 与 OF 上把它们定为 reg 类型了，导致仿真时输出一直为高阻态，其他没有什么大问题。