# 实验六源代码

**实验代码**

timescale 1ns / 1ps

//////////////////////////////////////////////////////////////////////////////////

// Company:

// Engineer:

//

// Create Date: 14:27:09 04/23/2021

// Design Name:

// Module Name: ALU

// Project Name:

// Target Devices:

// Tool versions:

// Description:

//

// Dependencies:

//

// Revision:

// Revision 0.01 - File Created

// Additional Comments:

//

//////////////////////////////////////////////////////////////////////////////////

module ALU(ALU\_OP,A,B,F,ZF,OF);

input [3:0]ALU\_OP;

input [32:1]A;

input [32:1]B;

output [32:1]F;

output ZF;

output OF;

//define input & output

reg [32:1]F;

reg C=0,OF,ZF;

always@(\*)

begin

C = 0;

OF = 0;

case(ALU\_OP)

4'b0000:begin F = A&B;end

4'b0001:begin F = A|B;end

4'b0010:begin F = A^B;end

4'b0011:begin F = ~(A|B);end

4'b0100:begin {C,F}=A+B; OF= A[32]^B[32]^F[32]^C; end

4'b0101:begin {C,F}=A-B; OF= A[32]^B[32]^F[32]^C; end

4'b0110:begin F = A<B;end

4'b0111:begin F = B<<A;end

4'b1000:begin {C,F} = A + 1; OF = C!=0; end

endcase

ZF = F==0;

end

endmodule

测试代码

initial begin

A=32'h55555555; B=32'h99999999;

ALU\_OP = 4'b0000; #100;//按位与

ALU\_OP = 4'b0001; #100;//按位或

ALU\_OP = 4'b0010; #100;//按位异或

ALU\_OP = 4'b0011; #100;//按位或非

ALU\_OP = 4'b0100; #100;//相加

ALU\_OP = 4'b0101; #100;//相减

ALU\_OP = 4'b0110; #100;//判断A<B

ALU\_OP = 4'b0111; #100;//B逻辑左移A

ALU\_OP = 4'b1000; #100;//自加1

A=32'h00FF00FF; B=32'h5F5F5F5F;

ALU\_OP = 4'b0000; #100;//按位与

ALU\_OP = 4'b0001; #100;//按位或

ALU\_OP = 4'b0010; #100;//按位异或

ALU\_OP = 4'b0011; #100;//按位或非

ALU\_OP = 4'b0100; #100;//相加

ALU\_OP = 4'b0101; #100;//相减

ALU\_OP = 4'b0110; #100;//判断A<B

ALU\_OP = 4'b0111; #100;//B逻辑左移A

ALU\_OP = 4'b1000; #100;//自加1

A=32'hF0F0F0F0; B=32'h00000004;

ALU\_OP = 4'b0000; #100;//按位与

ALU\_OP = 4'b0001; #100;//按位或

ALU\_OP = 4'b0010; #100;//按位异或

ALU\_OP = 4'b0011; #100;//按位或非

ALU\_OP = 4'b0100; #100;//相加

ALU\_OP = 4'b0101; #100;//相减

ALU\_OP = 4'b0110; #100;//判断A<B

ALU\_OP = 4'b0111; #100;//B逻辑左移A

ALU\_OP = 4'b1000; #100;//自加1

A=32'h00000004; B=32'hEFEFEFEF;

ALU\_OP = 4'b0000; #100;//按位与

ALU\_OP = 4'b0001; #100;//按位或

ALU\_OP = 4'b0010; #100;//按位异或

ALU\_OP = 4'b0011; #100;//按位或非

ALU\_OP = 4'b0100; #100;//相加

ALU\_OP = 4'b0101; #100;//相减

ALU\_OP = 4'b0110; #100;//判断A<B

ALU\_OP = 4'b0111; #100;//B逻辑左移A

ALU\_OP = 4'b1000; #100;//自加1

end

endmodule