

浙江大学

本科实验报告

课程名称： 数字逻辑设计

姓 名： 蒋奕

学 院： 计算机学院

系： 计算机系

专 业： 计算机科学与技术

学 号： 3210103803

指导教师： 马德

2022 年 12 月 5 日

浙江大学实验报告

课程名称: 数字逻辑设计 实验类型: 绘图实验+模块调用

实验项目名称: 寄存器和寄存器传输设计

学生姓名: 蒋奕 专业: 计算机科学与技术 学号: 3210103803

同组学生姓名: 任庭旭 指导老师: 马德

实验地点: 东 4-509 实验日期: 2022 年 12 月 4 日

一、实验目的:

- ①掌握寄存器传输电路的工作原理。
- ②掌握寄存器传输电路的设计方法。
- ③掌握 ALU 和寄存器传输电路的综合应用。

二、实验内容

实验一: 采用寄存器传输原理设计计数器

验证寄存器的设置初值功能, 验证寄存器自增、自减功能, 合理设计 4 位数码管上的显示内容。

Top Module 主要代码:

```
wire [3:0] Load_A, Co;
wire [3:0] A, A_IN, A1;
wire [31:0] clk_div;
MyRegister4b RegA(.clk(clk), .IN(A_IN), .Load(Load_A), .OUT(A));
Load_Gen m0(.clk(clk), .clk_1ms(clk_div[17]), .btn_in(SW[2]),
    .Load_out(Load_A)); //寄存器 A 的 Load 信号
clkdiv m3(clk, 1'b0, clk_div);
AddSub4b m4(.A(A), .B(4'b0001), .Ctrl(SW[0]), .S(A1)); //自增/自减逻辑
assign A_IN = (SW[15] == 1'b0)? A1: 4'b0000; //2 选 1 多路复用器,
复位寄存器初值
DispNum m8(.clk(clk), .HEXS({A, A1, A_IN, 4'b0000}), .LES(4'b0),
    .points(4'b0), .RST(1'b0), .AN(AN), .Segment(SEGMENT));
.....
```

建立仿真波形文件, 进行仿真。

实验二: 基于多路选择器总线的寄存器传输

验证 A、B、C 寄存器的设置初值功能, 验证 A、B 寄存器的自增、自减功能, 验证 A、B、C 寄存器之间的传输功能, 合理设计 4 位数码管上的显示内容。

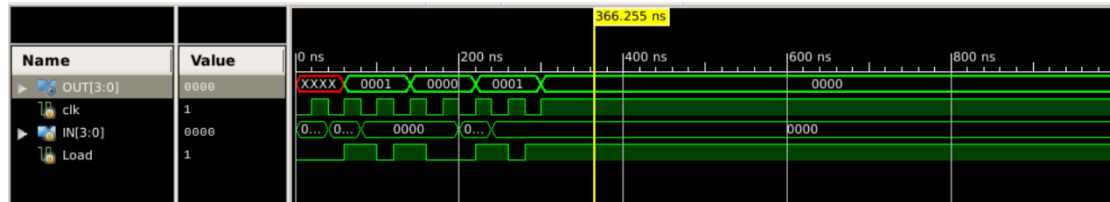
建立仿真波形文件 (包含正向计数和反向计数), 进行仿真。

实验三、基于 ALU 的数据传输应用设计

验证 A、B、C 寄存器的设置初值功能，验证 A、B 寄存器的自增、自减功能，验证 ALU 运算功能，验证寄存器传输功能，合理设计 4 位数码管上的显示内容。

二、实验数据记录和处理

实验一：采用寄存器传输原理设计计数器

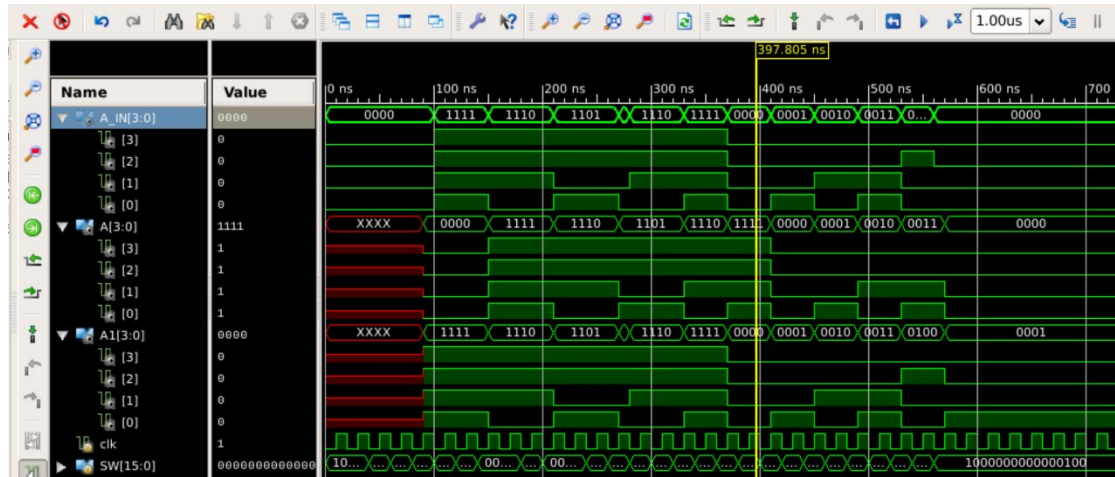


4 位 register 实验仿真波形如上
对应的 Verilog 代码如下：

```
module MyRegister4b(  
    input wire clk,  
    input [3:0] IN,  
    input wire Load,  
    output reg [3:0] OUT  
);  
always @ (posedge clk) begin  
    if (Load) OUT <= IN;  
end  
endmodule
```

Top 模块：

```
module top(  
    input wire clk,  
    input wire [15:0] SW,  
    output wire [3:0] A_IN,  
    output wire [3:0] A,  
    output wire [3:0] A1  
);  
wire [3:0] Load_A;  
wire [31:0] clk_div;  
MyRegister4b RegA(.clk(clk), .IN(A_IN), .Load(Load_A), .OUT(A));  
Load_Gen  
m0(.clk(clk), .clk_1ms(clk_div[17]), .btn_in(SW[2]), .Load_out(Load_A))  
;  
clkdiv m3(clk, 1'b0, clk_div);  
AddSub4b m4(.A(A), .B(4'b0001), .Ctrl(SW[0]), .S(A1));  
assign A_IN = (SW[15] == 1'b0)? A1: 4'b0000;  
endmodule
```



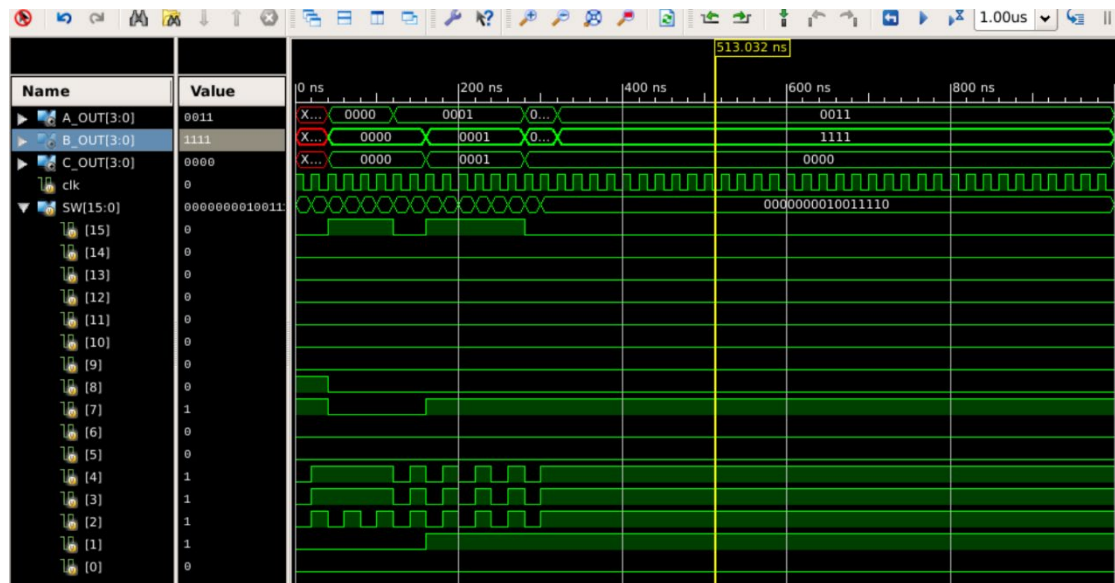
实验仿真波形如上
对应的 Verilog 仿真代码如下：

```
`timescale 1ns / 1ps
module top_test_2;
    reg clk;
    reg [15:0] SW;
    wire [3:0] A_IN,A,A1;
    top uut (
        .clk(clk),
        .SW(SW),
        .A_IN(A_IN),
        .A(A),
        .A1(A1)
    );
    initial begin
        clk = 0;SW = 16'b1000_0000_0000_0100;#40;
        clk = 0; SW[0] = 1;SW[2] = 0;SW[15] = 1;SW[2] = 0;#20;
        SW[2] = 1;#20;SW[2] = 0;#20;
        SW[15] = 0;SW[2] = 0;#20;SW[2] = 1;#20;SW[2] = 0;#20;
        SW[2] = 0;#20;SW[2] = 1;#20;SW[2] = 0;#20;SW[2] = 0;#20;
        SW[2] = 1;#20;SW[2] = 0;#20;SW[0] = 0;SW[2] = 0;#20;
        SW[2] = 1;#20;SW[2] = 0;#20;SW[2] = 1;#20;
        SW[2] = 0;#20;SW[2] = 1;#20;SW[2] = 0;#20;SW[2] = 1;#20;
        SW[2] = 0;#20;SW[2] = 1;#20;SW[2] = 0;#20;SW[2] = 1;#20;
        SW[2] = 0;#20;SW[2] = 1;#20;
        SW[15] = 1;#20;
    end
    always begin
        clk = 0;#10;
        clk = 1;#10;
    end
endmodule
```

实验二：基于多路选择器总线的寄存器传输

Top 模块：

```
module top(
    input clk,
    input [15:0] SW,
    output wire [3:0] A_OUT,//
    output wire [3:0] B_OUT,//
    output wire [3:0] C_OUT//
);
    wire [31:0] clk_div;
    wire [3:0] A_IN, B_IN, C_IN;
    wire Load_A, Load_B, Load_C;
    wire [3:0] I0_A, I0_B, I0_C, I1;
    Load_Gen
m0(.clk(clk), .clk_1ms(clk_div[17]), .btn_in(SW[2]),.Load_out(Load_A))
;
    Load_Gen
m1(.clk(clk), .clk_1ms(clk_div[17]), .btn_in(SW[3]),.Load_out(Load_B))
;
    Load_Gen
m2(.clk(clk), .clk_1ms(clk_div[17]), .btn_in(SW[4]),.Load_out(Load_C))
;
    clkdiv m3(clk, 1'b0, clk_div);
    AddSub4b m4(.A(A_OUT), .B(4'b0001), .Ctrl(SW[0]), .S(I0_A));
    AddSub4b m5(.A(B_OUT), .B(4'b0001), .Ctrl(SW[1]), .S(I0_B));
    assign A_IN = (SW[15] == 1'b0) ? I0_A : I1;
    assign B_IN = (SW[15] == 1'b0) ? I0_B : I1;
    assign C_IN = (SW[15] == 1'b0) ? 4'b0000 : I1;
    Mux4to14b m9(A_OUT, B_OUT, C_OUT, 4'b0000, SW[8:7], I1);
    MyRegister4b RegA( clk, A_IN, Load_A, A_OUT );
    MyRegister4b RegB( clk, B_IN, Load_B, B_OUT );
    MyRegister4b RegC( clk, C_IN, Load_C, C_OUT );
endmodule
```



实验仿真波形如上
对应的 Verilog 仿真代码如下：

```
`timescale 1ns / 1ps
module top_test;
    reg clk;
    reg [15:0] SW;
    wire [3:0] A_OUT,B_OUT,C_OUT;
    top uut (
        .clk(clk),
        .SW(SW),
        .A_OUT(A_OUT),
        .B_OUT(B_OUT),
        .C_OUT(C_OUT)
    );
    initial begin
        SW[0] = 0;SW[1] = 0;SW[5] = 0;SW[6] = 0;SW[9] = 0;SW[10] = 0;
        SW[11] = 0;SW[12] = 0;SW[13] = 0;SW[14] = 0;SW[15] = 0;
        SW[7] = 1; SW[8] = 1;SW[15] = 0;SW[2] = 0;SW[3] = 0;SW[4] = 0;#20;
        SW[2] = 1;SW[3] = 1;SW[4] = 1;#20;
//A SUB
        SW[15] = 1;SW[7] = 0;SW[8] = 0;SW[2] = 0;#20;
        SW[2] = 1;#20;SW[2] = 0;#20;SW[2] = 1;#20;
//I1 to all register
        SW[15] = 0;SW[2] = 0;SW[3] = 0;SW[4] = 0;#20;
        SW[2] = 1;SW[3] = 1;SW[4] = 1;#20;
//B add
        SW[15] = 1;SW[7] = 1;SW[8] = 0;SW[1] = 1;SW[2] = 0;SW[3] = 0;SW[4]
= 0;#20;
        SW[2] = 1;SW[3] = 1;SW[4] = 1;#20;SW[2] = 0;SW[3] = 0;SW[4] = 0;#20;
        SW[2] = 1;SW[3] = 1;SW[4] = 1;#20;SW[2] = 0;SW[3] = 0;SW[4] = 0;#20;
```

```

    SW[2] = 1;SW[3] = 1;SW[4] = 1;#20;SW[15] = 0;SW[2] = 0;SW[3] = 0;
    SW[4] = 0;#20;
    SW[2] = 1;SW[3] = 1;SW[4] = 1;#20;
end
always begin
    clk=1;#10;clk=0;#10;
end
endmodule

```

实验三、基于 ALU 的数据传输应用设计

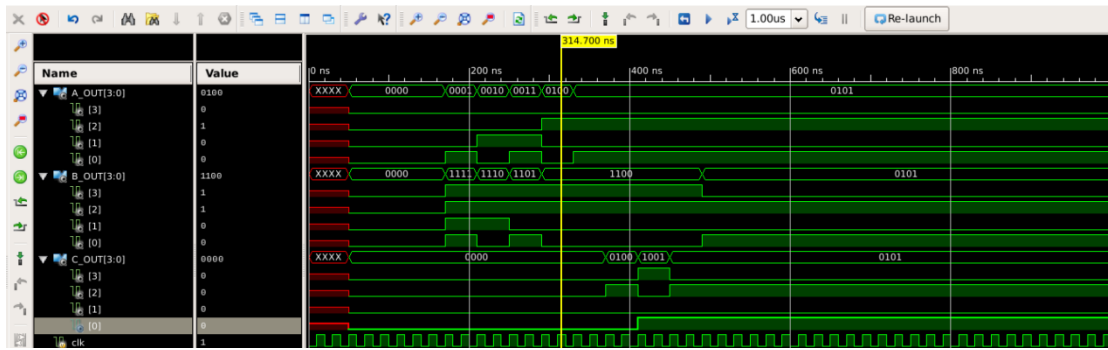
Top 模块

对应的 Verilog 代码如下：

```

module top(
    input clk,
    input [15:0] SW,
    output [3:0] A_OUT,
    output [3:0] B_OUT,
    output [3:0] C_OUT
);
    wire [3:0] A_IN, B_IN, C_IN, I1, I0_A, I0_B, I0_C;
    wire Load_A, Load_B, Load_C, Co;
    Load_Gen m0(.clk(clk), .btn_in(SW[2]), .Load_out(Load_A));
    Load_Gen m1(.clk(clk), .btn_in(SW[3]), .Load_out(Load_B));
    Load_Gen m2(.clk(clk), .btn_in(SW[4]), .Load_out(Load_C));
    AddSub4b m4(.A(A_OUT), .B(4'b0001), .Ctrl(SW[0]), .S(I0_A));
    AddSub4b m5(.A(B_OUT), .B(4'b0001), .Ctrl(SW[1]), .S(I0_B));
    assign A_IN = (SW[15] == 1'b0) ? I0_A : I1;
    assign B_IN = (SW[15] == 1'b0) ? I0_B : I1;
    assign C_IN = (SW[15] == 1'b0) ? I0_C : I1;
    Mux4to14b m9(A_OUT, B_OUT, C_OUT, 4'b0000, SW[8:7], I1);
    Register4b RegA(clk, A_IN, Load_A, A_OUT);
    Register4b RegB(clk, B_IN, Load_B, B_OUT);
    Register4b RegC(clk, C_IN, Load_C, C_OUT);
    MyALU m9(A_OUT, B_OUT, SW[6:5], I0_C, Co);
endmodule

```



实验仿真波形如上
对应的 Verilog 仿真代码如下：

```
`timescale 1ns / 1ps
module top_test;
    reg clk;
    reg [15:0] SW;
    wire [3:0] A_OUT, B_OUT, C_OUT;
    top uut (
        .clk(clk),
        .SW(SW),
        .A_OUT(A_OUT),
        .B_OUT(B_OUT),
        .C_OUT(C_OUT)
    );
    initial begin
        SW[7] = 1;SW[8] = 1;SW[15] = 0;SW[2] = 0;SW[3] = 0;SW[4] = 0;#20;
        SW[2] = 1;SW[3] = 1;SW[4] = 1;#100;
        // Auto increment   Auto minus
        SW[15] = 1;SW[0] = 1;SW[1] = 0;SW[2] = 0;SW[3] = 0;#20;
        SW[2] = 1;SW[3] = 1;#20;SW[2] = 0;SW[3] = 0;#20;
        SW[2] = 1;SW[3] = 1;#20;SW[2] = 0;SW[3] = 0;#20;
        SW[2] = 1;SW[3] = 1;#20;SW[2] = 0;SW[3] = 0;#20;
        SW[2] = 1;SW[3] = 1;#20;SW[2] = 0;#20;SW[2] = 1;#20;
        //ALU test
        SW[6] = 1;SW[5] = 1;SW[4] = 0;#20;SW[4] = 1;#20;
        SW[6] = 1;SW[5] = 0;SW[4] = 0;#20;SW[4] = 1;#20;
        SW[6] = 0;SW[5] = 1;SW[4] = 0;#20;SW[4] = 1;#20;
        // transfer
        SW[7] = 0;SW[8] = 0;SW[15] = 0;SW[2] = 0;SW[3] = 0;SW[4] = 0;#20;
        SW[2] = 1;SW[3] = 1;SW[4] = 1;#100;
    end
    always begin
        clk = 0;#10;clk = 1;#10;
    end
endmodule
```


三、实验结果与分析

实验结果和预测结果一样寄存器传输功能、计数功能、ALU 传输功能均实现，实验操作正确无误。

四、讨论、心得（选填）

实验过程中我体会到要认真对待每一次操作，正如我多次因为不小心而操作不熟练使得实验进程受阻。