

同济大学计算机系

数字逻辑课程综合实验报告



学 号 2352595

姓 名 张嘉麟

专 业 计算机科学与技术

授课老师 郭玉臣

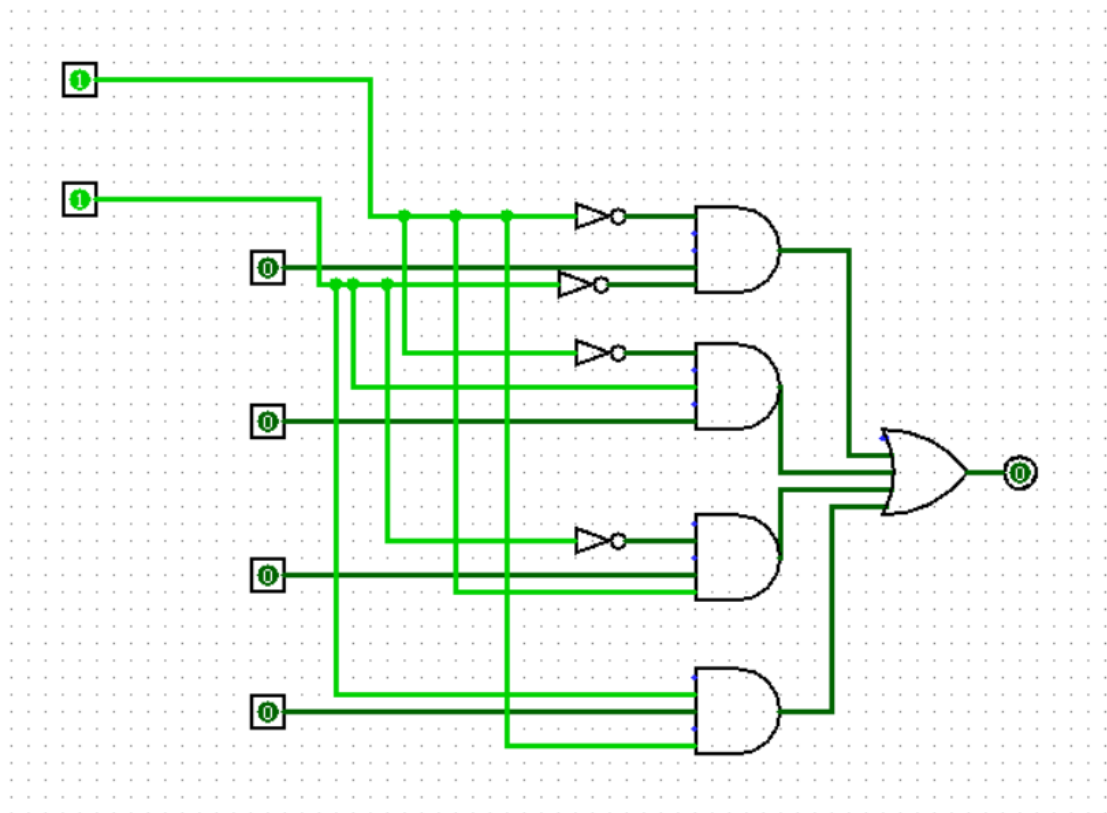
一、实验内容

二、硬件逻辑图

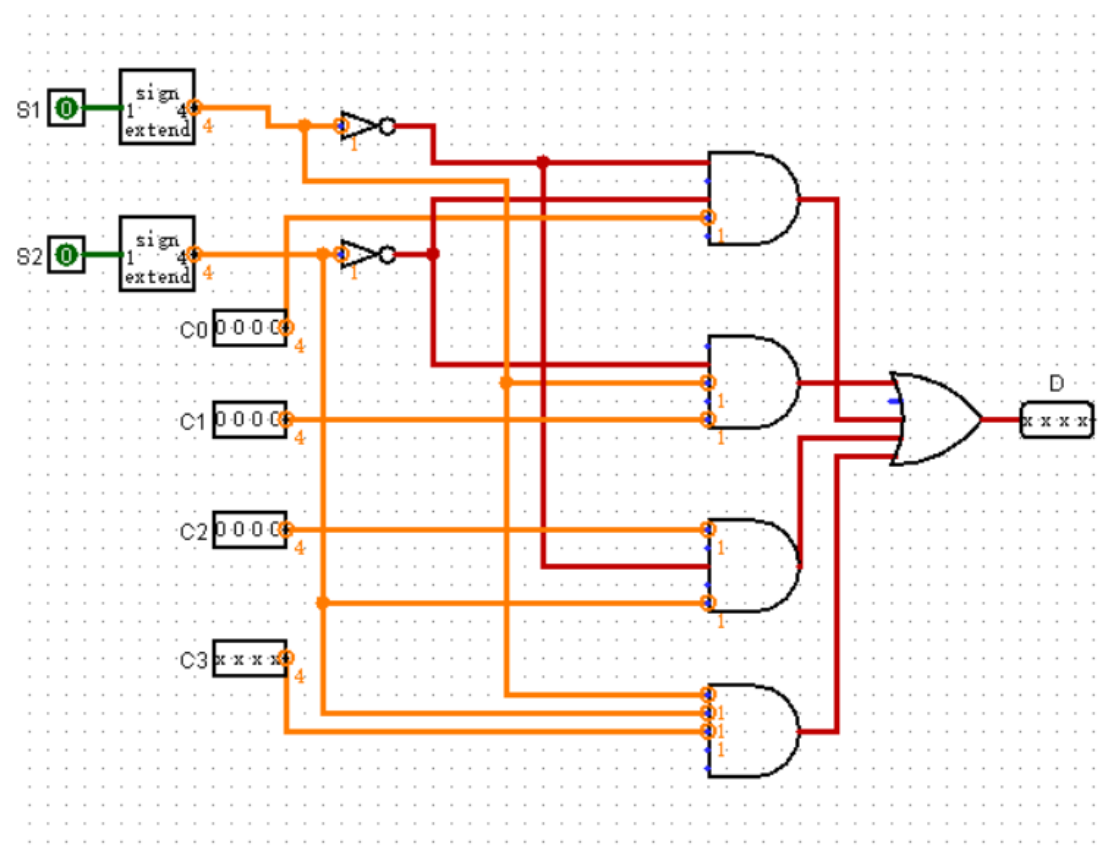
（实验步骤中要求用 logisim 画图的实验，在该部分给出 logisim 原理图，否则该部分在实验报告中不用写）

2.1 数据选择器

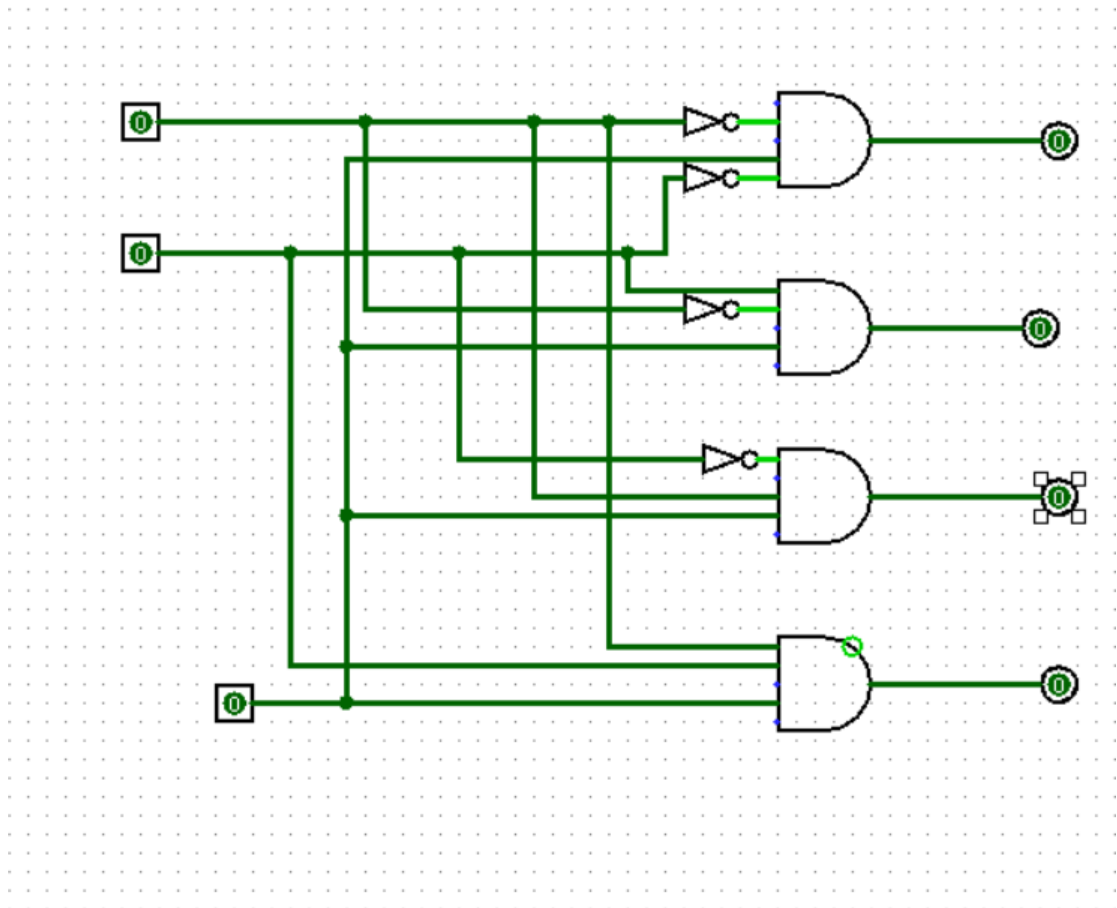
2.1.1 selector41 (1bit)



2.1.2 selector41 (4bit)



2.2 数据分配器



三、模块建模

（该部分要求对实验中建模的所有模块进行功能描述，并列出各模块建模的verilog 代码）

5.1 数据选择器

```
module selector41(
    input [3:0] iC0, // 四位输入信号 c0
    input [3:0] iC1, // 四位输入信号 c1
    input [3:0] iC2, // 四位输入信号 c2
    input [3:0] iC3, // 四位输入信号 c3
    input iS1,      // 选择信号 s1
    input iS0,      // 选择信号 s0
    output [3:0] oZ // 四位输出信号 z
);
assign oZ = (iS1 == 0 && iS0 == 0) ? iC0 :
            (iS1 == 0 && iS0 == 1) ? iC1 :
            (iS1 == 1 && iS0 == 0) ? iC2 :
            iC3;
endmodule
```

5.2 数据分配器

```
module de_selector14(  
    input iC, //输入信号 c  
    input iS1, //选择信号 s1  
    input iS0, //选择信号 s0  
    output oZ0, //输出信号 z0  
    output oZ1, //输出信号 z1  
    output oZ2, //输出信号 z2  
    output oZ3 //输出信号 z3  
);  
  
assign oZ0 = (~iS1 & ~iS0) ? iC : 1'b1; // 当 iS1 iS0 为 00 时, oZ0 输出 iC  
assign oZ1 = (~iS1 & iS0) ? iC : 1'b1; // 当 iS1 iS0 为 01 时, oZ1 输出 iC  
assign oZ2 = (iS1 & ~iS0) ? iC : 1'b1; // 当 iS1 iS0 为 10 时, oZ2 输出 iC  
assign oZ3 = (iS1 & iS0) ? iC : 1'b1; // 当 iS1 iS0 为 11 时, oZ3 输出 iC  
  
endmodule
```

5.3 八路数据传输实验

```
module transmission8(  
    input [7:0] iData, // 输入信号 D7~D0  
    input A, B, C, // 选择信号 S2~S0  
    output reg [7:0] oData // 输出信号 f0~f7  
);  
  
always @(iData, A, B, C, oData)  
begin  
    case({A, B, C})  
        3'b000:  
            begin  
                oData[0]=iData[0];  
                oData[1]=1;  
                oData[2]=1;  
                oData[3]=1;  
                oData[4]=1;  
                oData[5]=1;  
                oData[6]=1;  
                oData[7]=1;  
            end  
        3'b001:  
            begin  
                oData[0]=1;  

```

```

        oData[1]=iData[1];
        oData[2]=1;
        oData[3]=1;
        oData[4]=1;
        oData[5]=1;
        oData[6]=1;
        oData[7]=1;
        end

3'b010:

        begin
        oData[0]=1;
        oData[1]=1;
        oData[2]=iData[2];
        oData[3]=1;
        oData[4]=1;
        oData[5]=1;
        oData[6]=1;
        oData[7]=1;
        end

3'b011:

        begin
        oData[0]=1;
        oData[1]=1;
        oData[2]=1;
        oData[3]=iData[3];
        oData[4]=1;
        oData[5]=1;
        oData[6]=1;
        oData[7]=1;
        end

3'b100:

        begin
        oData[0]=1;
        oData[1]=1;
        oData[2]=1;
        oData[3]=1;
        oData[4]=iData[4];
        oData[5]=1;
        oData[6]=1;
        oData[7]=1;
        end

3'b101:
begin

```

```

oData[0]=1;
oData[1]=1;
oData[2]=1;
oData[3]=1;
oData[4]=1;
oData[5]=iData[5];
oData[6]=1;
oData[7]=1;
end

3'b110:
begin
oData[0]=1;
oData[1]=1;
oData[2]=1;
oData[3]=1;
oData[4]=1;
oData[5]=1;
oData[6]=iData[6];
oData[7]=1;
end

3'b111:
begin
oData[0]=1;
oData[1]=1;
oData[2]=1;
oData[3]=1;
oData[4]=1;
oData[5]=1;
oData[6]=1;
oData[7]=iData[7];
end

endcase
end

endmodule

```

四、测试模块建模

（要求列写各建模模块的 test bench 模块代码）

5.1 数据选择器

```
`timescale 1ns / 1ns
module selector41_tb;
reg [3:0] iC0;
reg [3:0] iC1;
reg [3:0] iC2;
reg [3:0] iC3;
reg iS0;
reg iS1;
wire [3:0] oZ;
selector41 uut(
.iC0(iC0),
.iC1(iC1),
.iC2(iC2),
.iC3(iC3),
.iS0(iS0),
.iS1(iS1),
.oZ(oZ)
);
initial
begin
iC0 = 2'b0001;
iC1 = 2'b0010;
iC2 = 2'b0100;
iC3 = 2'b1000;
iS0 = 0;
#40 iS0 = 1;
#40 iS0 = 0;
#40 iS0 = 1;
end
initial
begin
iS1 = 0;
#40 iS1 = 0;
#40 iS1 = 1;
#40 iS1 = 1;
end
endmodule
```

5.2 数据分配器

```
`timescale 1ns / 1ns
module de_selector14_tb;
// 输入信号定义
```



```

reg iC;
reg iS1;
reg iS0;
// 输入信号 c
// 选择信号 s1
// 选择信号 s0
// 输出信号定义
wire oZ0;
// 输出信号 z0
    wire oZ1;    // 输出信号 z1
    wire oZ2;    // 输出信号 z2
    wire oZ3;    // 输出信号 z3
// 实例化被测试的模块
de_selector14 uut (
    .iC(iC),
    .iS1(iS1),
    .iS0(iS0),
    .oZ0(oZ0),
    .oZ1(oZ1),
    .oZ2(oZ2),
    .oZ3(oZ3)
);
// 仿真初始块
initial
begin
// 设置初始输入信号
iC = 0; iS1 = 0; iS0 = 0;
#10;
iC = 1; iS1 = 0; iS0 = 0;
#10;
iC = 1; iS1 = 0; iS0 = 1;
#10;
end
endmodule

```

5.3 八路数据传输实验

```

module tb_transmission8;

    // 输入信号定义
    reg [7:0] iData; // 输入信号 D7~D0
    reg A, B, C;     // 选择信号 S2~S0

    // 输出信号定义
    wire [7:0] oData; // 输出信号 f0~f7

```

```

// 实例化被测模块
transmission8 uut (
    .iData(iData),
    .A(A),
    .B(B),
    .C(C),
    .oData(oData)
);

// 仿真初始块
initial
begin
    A=0;
    B=0;
    C=0;
    #100 A=0;B=0;C=1;
    #100 A=0;B=1;C=0;
    #100 A=0;B=1;C=1;
    #100 A=1;B=0;C=0;
    #100 A=1;B=0;C=1;
    #100 A=1;B=1;C=0;
    #100 A=1;B=1;C=1;
    #100;
end

initial
begin
    iData = 8'b00000000;
    #800;
end

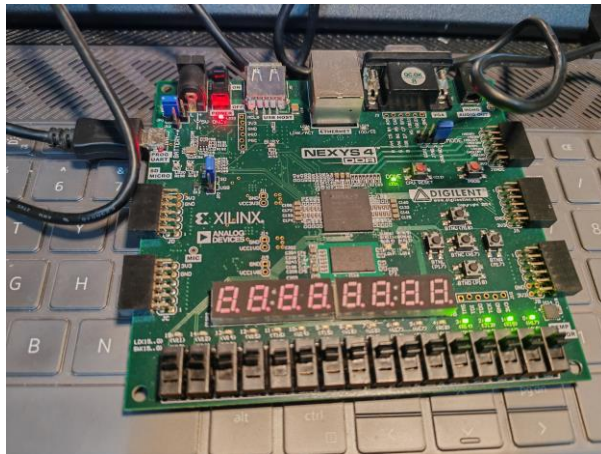
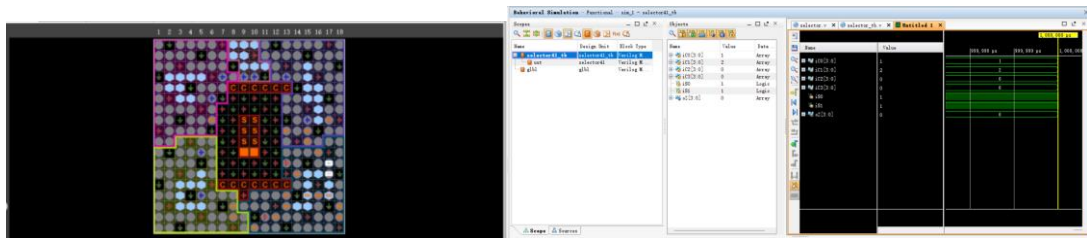
endmodule

```

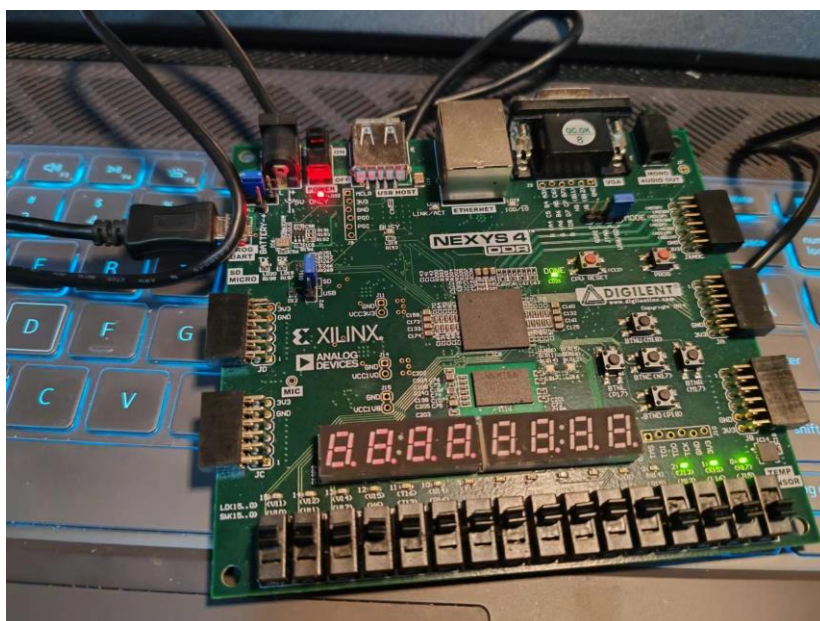
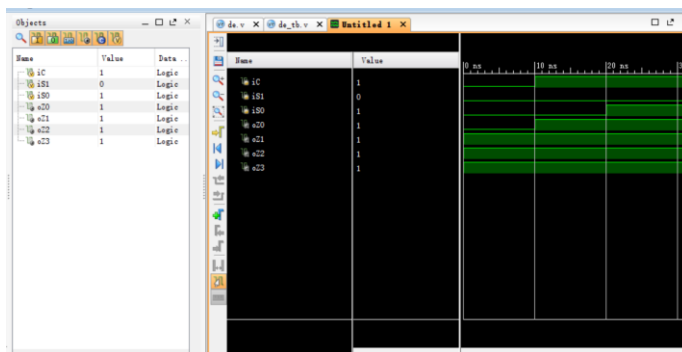
五、实验结果

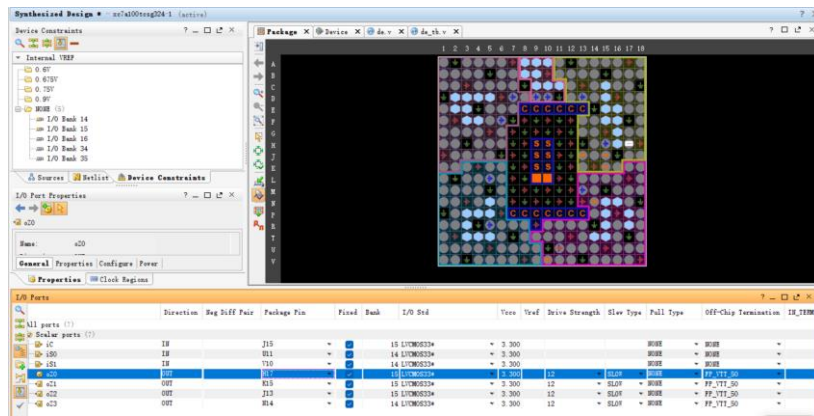
（该部分可截图说明，要求 logisim 逻辑验证图、modelsim 仿真波形图、以及下板后的实验结果贴图（实验步骤中没有下板要求的实验，不需要下板贴图））

5.1 数据选择器



5.2 数据分配器





5.3 八路数据传输实验

