

# 同济大学计算机系

## 数字逻辑课程实验报告



学 号 2350752

姓 名 田思宇

专 业 计算机科学与技术

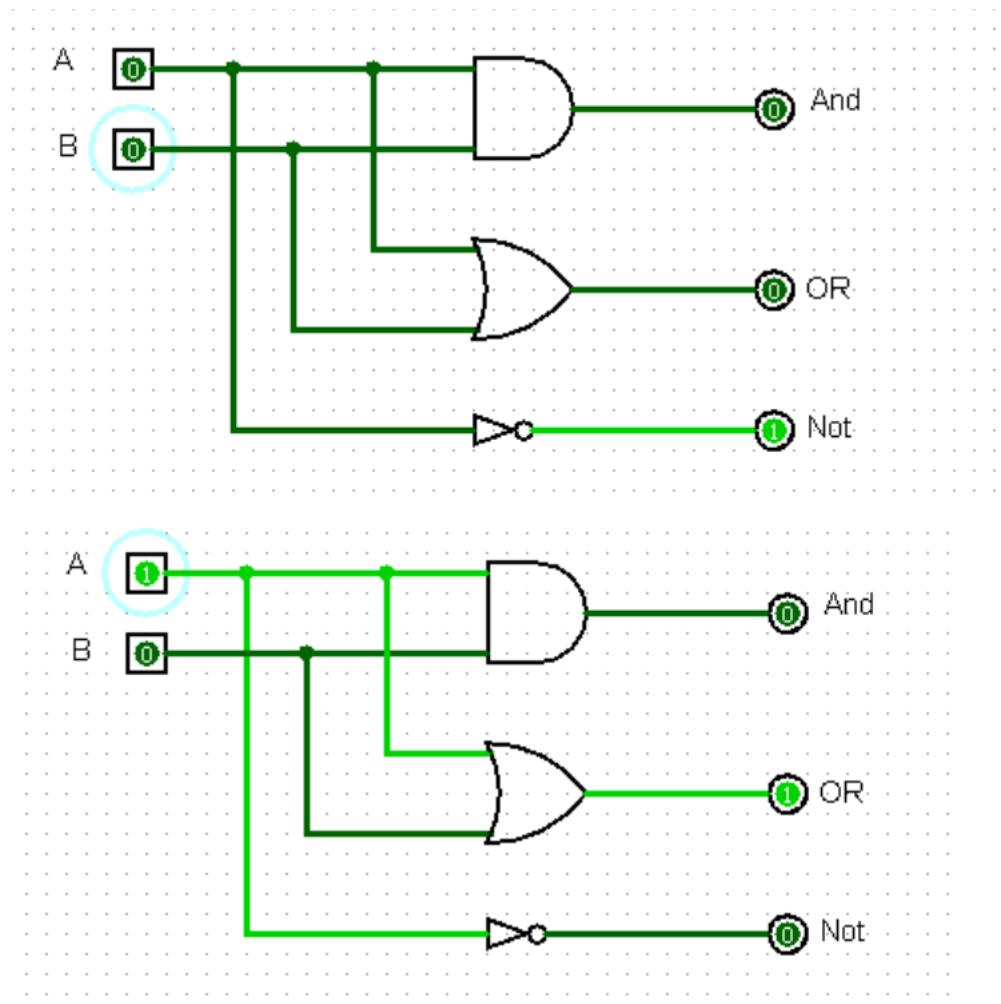
授课老师 张冬冬

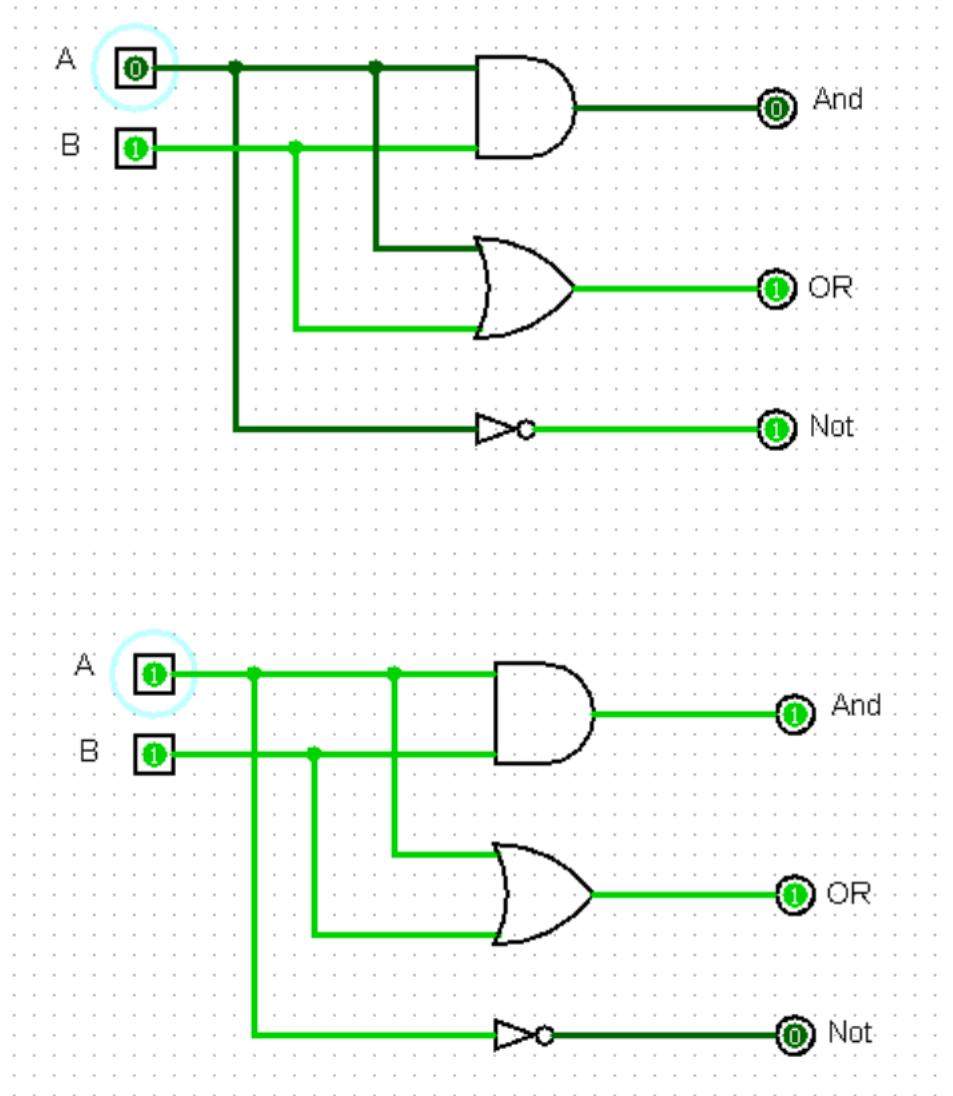
## 一、实验内容

基本门电路与数据扩展描述

1. 基本与或非门实验
2. 三态门实验
3. 数据扩展实验

## 二、硬件逻辑图





### 三、模块建模

#### 实验一：基本与或非门实验

1、模块功能描述：该模块设计了与门、或门、非门，共有 a、b 两个输入，三个输出，其中与门和或门的输入是 a、b，非门的输入是 a。

2、代码：

(结构型描述)

```
module logic_gates_1(iA,iB,oAnd,oOr,oNot);
    input iA, iB;
    output oAnd,oOr,oNot;
    and and_inst(oAnd, iA,iB);
    or  or_inst(oOr, iA,iB);
    not not_inst(oNot, iA);
endmodule
```

(数据流型描述)

```
module logic_gates_2(iA,iB,oAnd,oOr,oNot);
  input iA, iB;
  output oAnd,oOr,oNot;
  assign oAnd = iA & iB;
  assign oOr = iA | iB;
  assign oNot = ~iA;
endmodule
```

(行为描述)

```
module logic_gates_3(iA,iB,oAnd,oOr,oNot);
  input iA, iB;
  output oAnd,oOr,oNot;
  reg oAnd, oOr, ONot;
  always @ (*)
    begin
      oAnd = iA & iB;
      oOr = iA | iB;
      oNot = ~ iA;
    end
endmodule
```

## 实验二：三态门实验

1、模块功能描述：本模块共两个输入，一个是 a，另一个是使能端 ena，当使能端输入为 0 时候，输出永远为高阻抗，当使能端输入为 1 时，输出与输入 a 的结果相同。

2、代码：

```
module three_state_gates(iA,iEna,oTri);
  input iA;
  input iEna;
  output oTri;
  assign oTri = (iEna==1)? iA:'bz;
endmodule
```

## 实验三：数据拓展

1、模块功能描述：本模块设定 16 位的输入，将其拓展为 32 位，实际上只要修改 width 的值，就可以实现任意小于 32 位的输入。判断 16 位输入的最高有效位是否为 1，如果为 1 则高位全部拓展为 1，如果是 0 则高位全部拓展为 0。

2、代码：

```
module extend #(parameter WIDTH = 16)
(
  input [WIDTH-1:0] a,
  input sext,
  output [31:0]b
);
  assign b=sext?{{(32-WIDTH){a[WIDTH-1]}},a}:{(32-WIDTH){1'b0}},a};
endmodule
```

## 四、测试模块建模

实验一：

```
`timescale 1ns/1ns
module logic_gates_tb;
reg iA;
reg iB;
wire oAnd;
wire oOr;
wire oNot;
initial
begin
iA = 0;
#40 iA = 1;
#40 iA = 0;
#40 iA = 1;
#40 iA = 0;
end
initial
begin
iB = 0;
#40 iB = 0;
#40 iB = 1;
#40 iB = 1;
#40 iB = 0;
end
logic_gates_1
logic_gates_inst(.iA(iA),.iB(iB),.oAnd(oAnd),.oOr(oOr),.oNot(oNot));
endmodule
```

实验二：

```
`timescale 1ns/1ns
module three_state_gates_tb;
reg iA;
reg iEna;
wire oTriState;
three_state_gates uut (.iA(iA), .iEna(iEna), .oTriState(oTriState));
initial
begin
iA = 0;
```

```
#40 iA = 1;
#40 iA = 0;
#40 iA = 1;
end
initial
begin
iEna = 1;
#20 iEna = 0;
#40 iEna= 1;
#20 iEna = 0;
end
endmodule
```

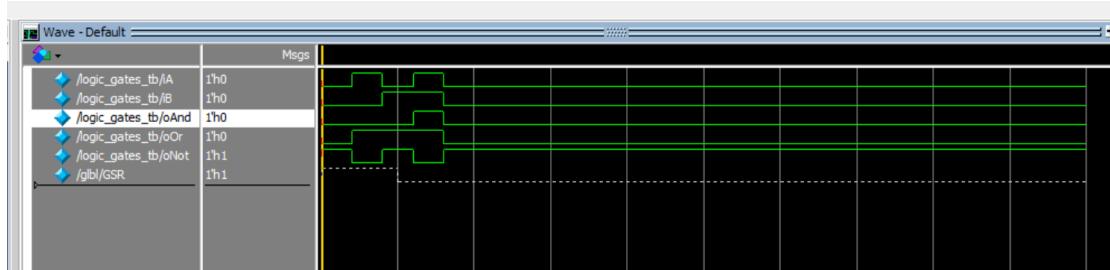
### 实验三：

```
`timescale 1ns/1ns
module extend_tb;
reg [15:0] a;
reg sext;
wire [31:0] b;
extend uut (.a(a),.sext(sext),.b(b));
initial
begin
a = 0;
sext = 0;
#100;
sext = 1;
a = 16'h0000;
#100;
sext = 0;
a = 16'h8000;
#100;
sext = 1;
a = 16'h8000;
#100;
sext = 0;
a = 16'hffff;
#100;
sext = 1;
a = 16'hffff;
#100;
end
endmodule
```

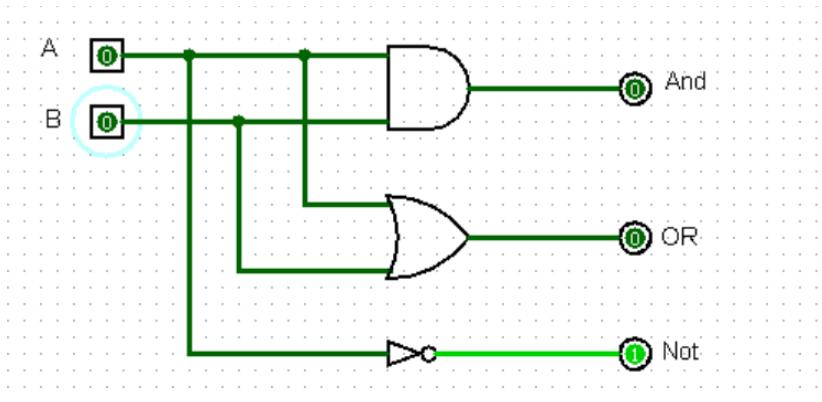
## 五、实验结果

### 实验一：

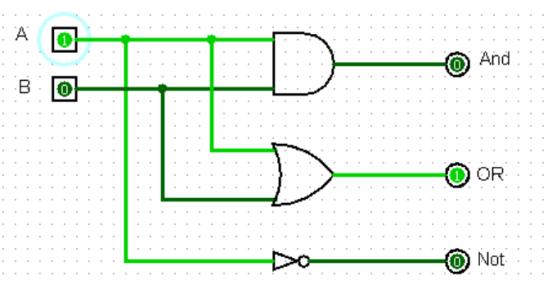
仿真波形图如下图所示



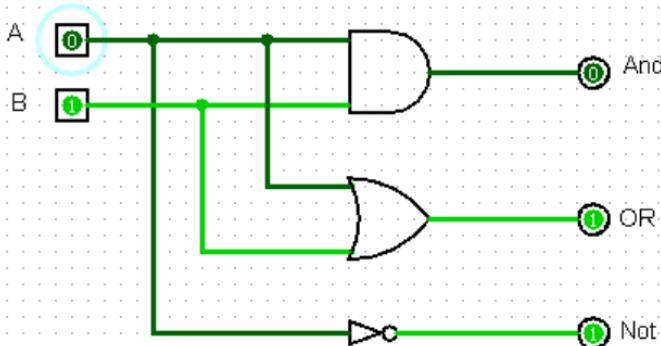
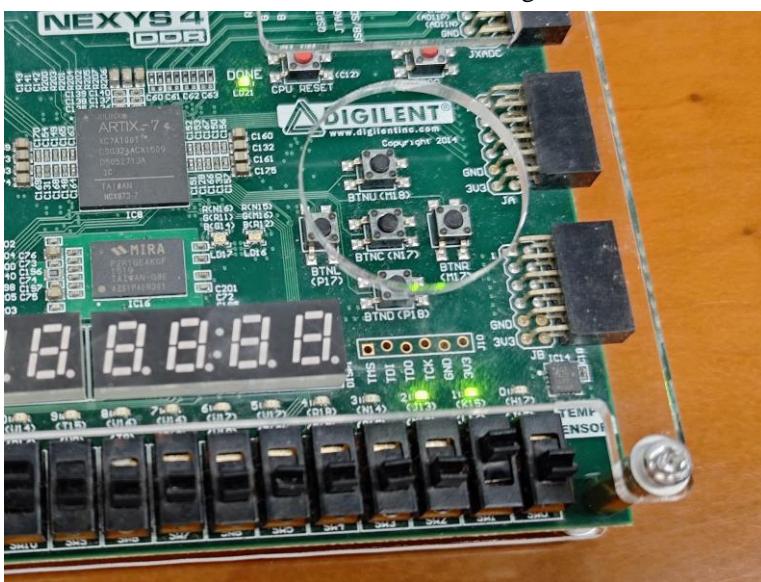
1、a 和 b 输入均为 0，输出结果与 logisim 一致



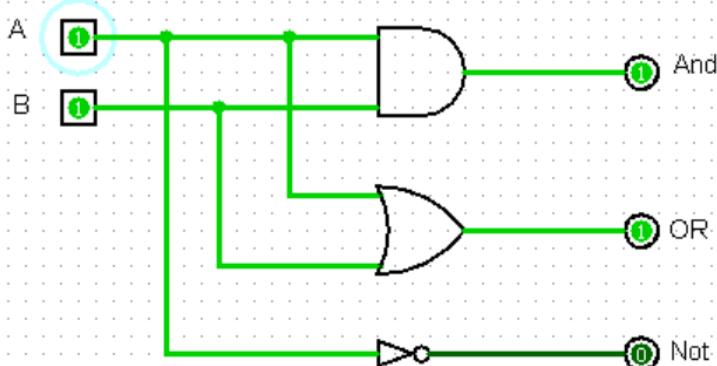
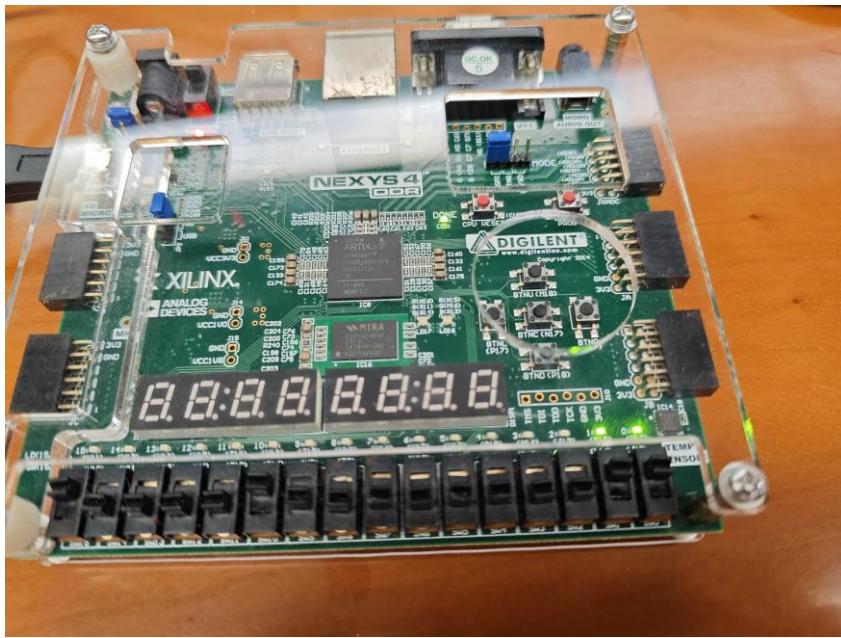
2、a 输入为 1， b 输入为 0，输出结果与 logisim 一致



3、a 输入为 0, b 输入为 1, 输出结果与 logisim 一致

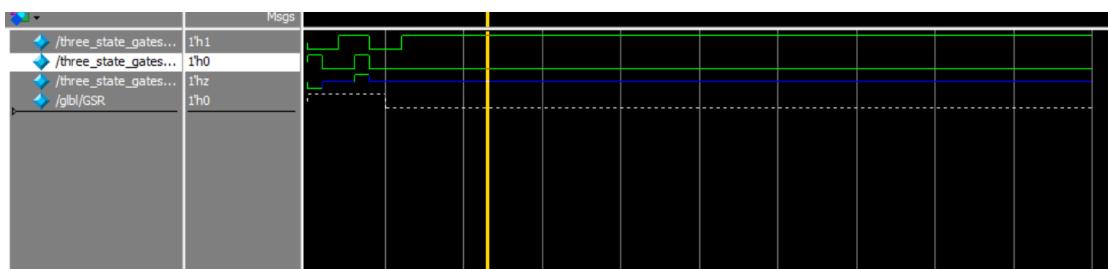


4、a 输入为 1, b 输入为 1, 输出结果与 logisim 一致



## 实验二：

1、仿真波形图如下图所示

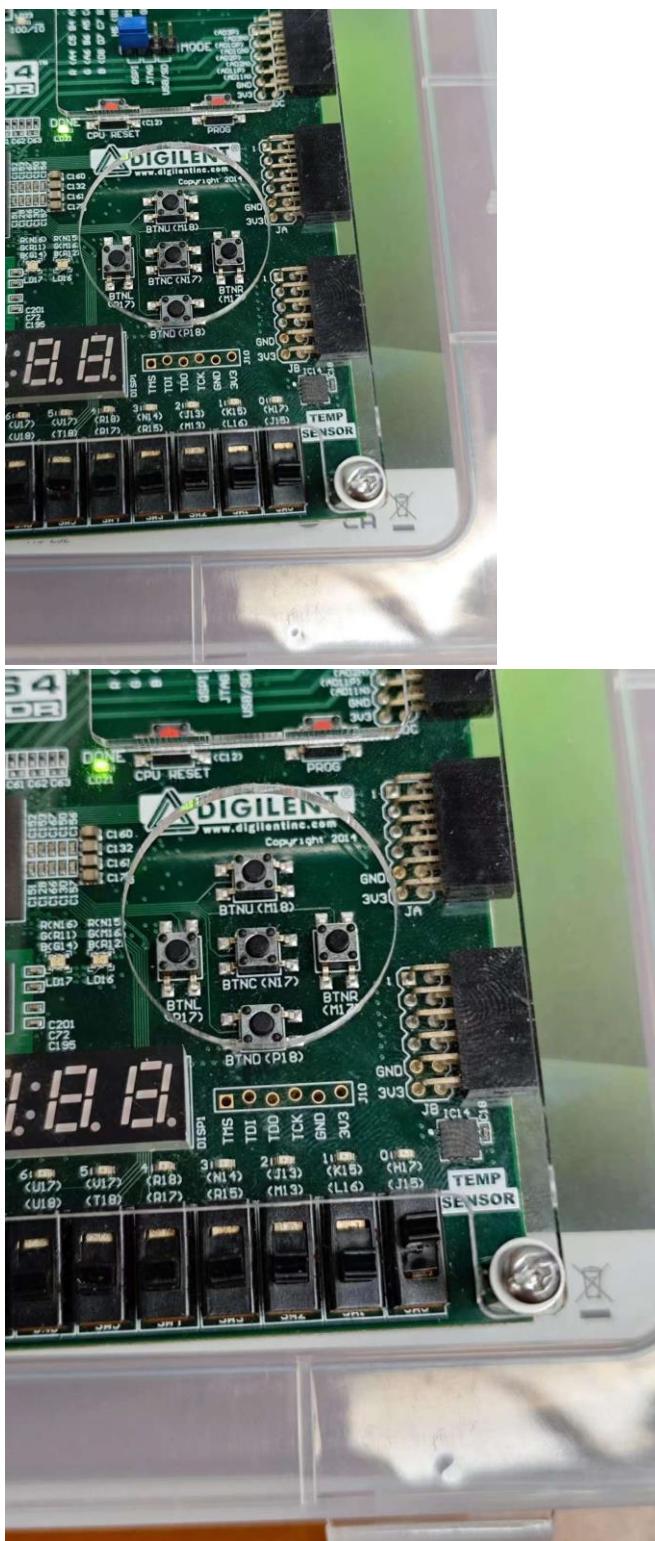


2、下板：

实验分析：

当使能端为 0 时候，无论 a 为 0 还是 1，输出都是高阻抗（灯不亮）。

只有在使能端为 1（按下按钮时），且 a 输入为 1 时候，输出才是 1(指示灯亮)。





### 实验三：

仿真波形图如下图所示

