**实验二 组合电路设计实验报告**

**姓名**：  **学号**：1120190699

**班级**： **手机**：

1. **实验题目**

设计一个组合电路，输入一个4位的数字，输出一个3位的二进制数字，且输出数字的值近似等于输入数字的平方根。例如，如果平方根的值等于3.5或者更大的值，则四舍五入记为4。如果平方根的值小于3.5大于等于2.5，则记为3。

1. **实验约束**

* 电路设计时只能使用与非门和非门进行实现。
* 采用Verilog实现时使用结构化描述方式。

1. **电路设计**
   1. **规范化**

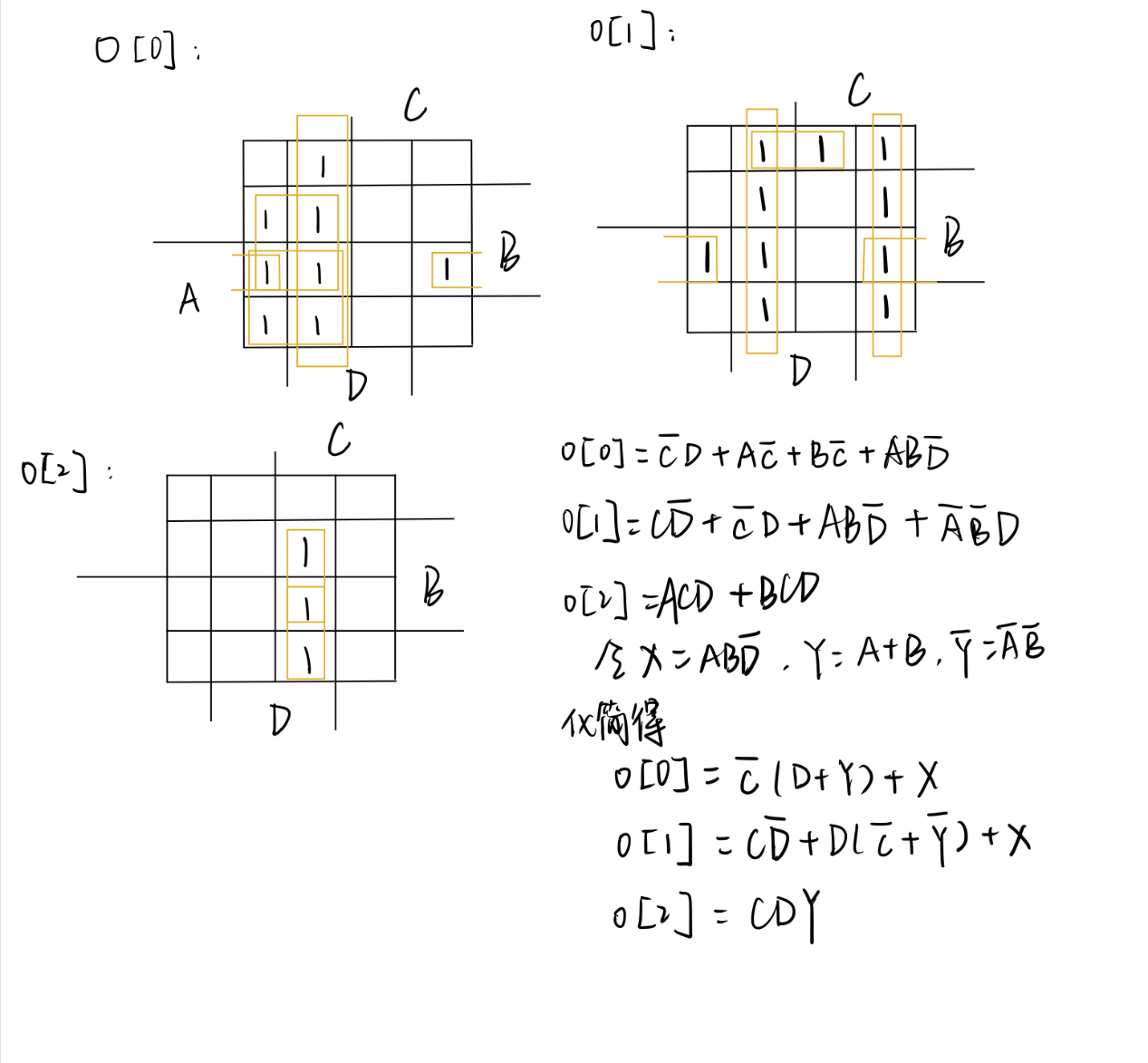
输入：一个4bit位宽的向量wire i

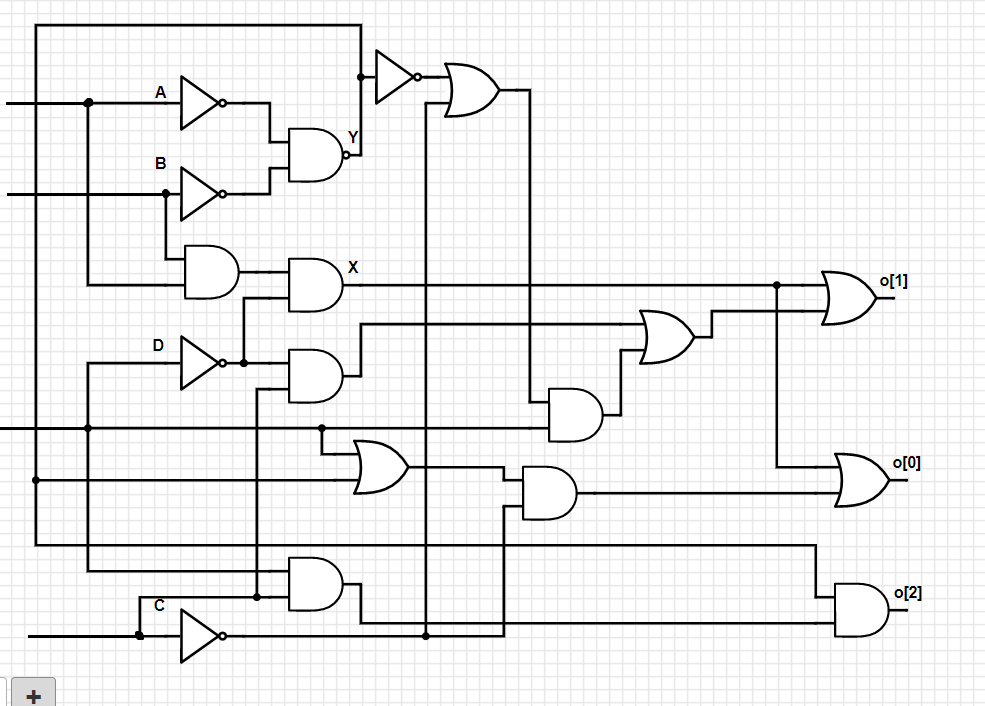
电路行为：根据数组i的情况为输出数组o赋值相应的平方根结果

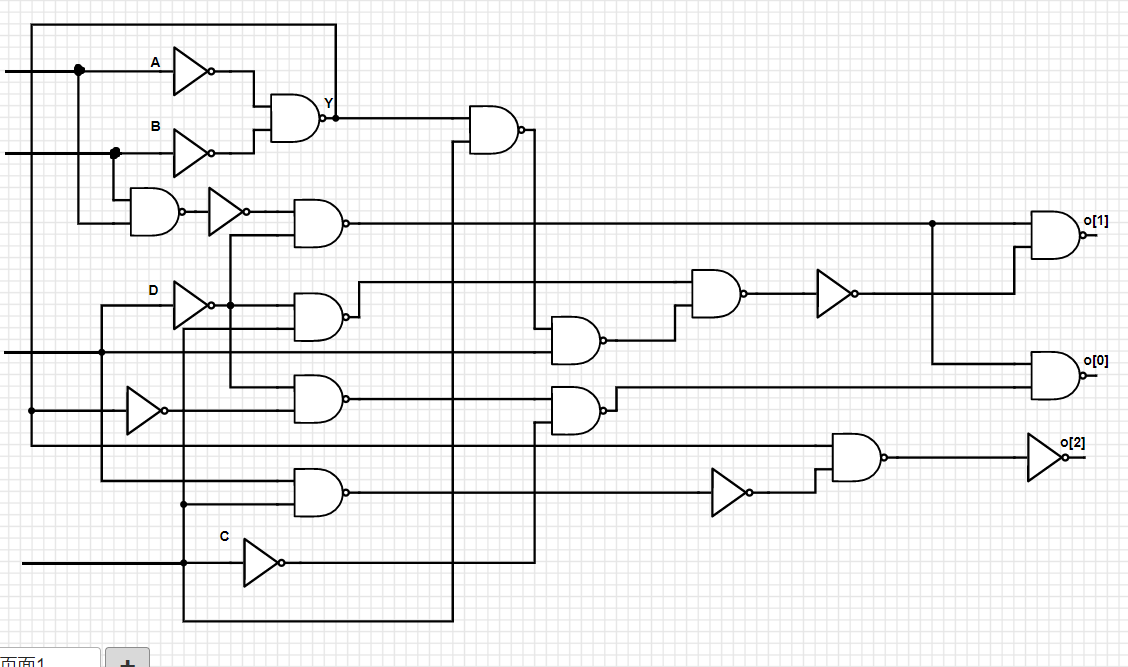
输出：一个3bit位宽的向量wire o

* 1. **形式化**

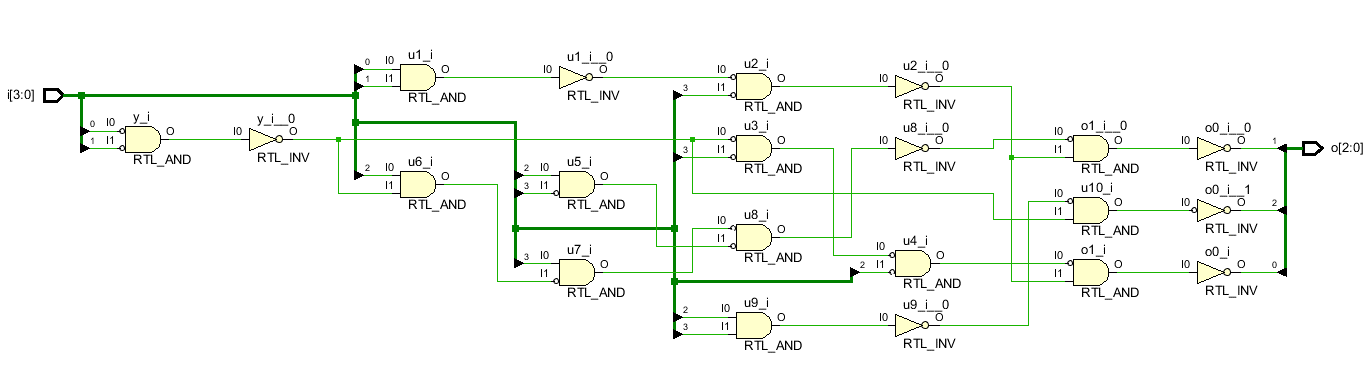
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Input输入 | | | | Output输出 | | |
| I[3] | **I[2]** | **I[1]** | **I[0]** | **O[2]** | **O[1]** | **O[0]** |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 |

* 1. **优化**
  2. **工艺映射**

工艺映射后，自己画的图如下：

****

Vivado中直接截取电路图如下：



1. **电路实现**

//仅使用nand与非门 not反相器对电路图进行结构化描述实现

//nand (output,input,...)

//not (output,...,input)

module sqrt(i,o);

/\*先定义好输入是一个一个4bit位宽的向量i，输出是一个3bit位宽的向量o\*/

input[3:0] i;

output[2:0] o;

//由于wire类型是verilog语言默认的数据类型，因此为方便起见以下的节点将省略wire声明

not (inva,i[0]),(invb,i[1]),(invc,i[2]),(invd,i[3]);

//先描述o[0]的输出

nand

y(y\_out,inva,invb),

u1(u1\_out,i[0],i[1]);

not

invu1(invu1\_out,u1\_out),

invy(invy\_out,y\_out);

nand

u2(u2\_out,invu1\_out,invd),

u3(u3\_out,invy\_out,invd),

u4(u4\_out,u3\_out,invc);

nand (o[0],u4\_out,u2\_out);

//描述o[1]的输出

nand

u5(u5\_out,i[2],invd),

u6(u6\_out,i[2],y\_out),

u7(u7\_out,i[3],u6\_out),

u8(u8\_out,u7\_out,u5\_out);

not (invu8,u8\_out);

nand (o[1],invu8,u2\_out);

//描述o[2]的输出

nand

u9(u9\_out,i[2],i[3]);

not (invu9,u9\_out);

nand

u10(u10\_out,invu9,y\_out);

not (o[2],u10\_out);

endmodule

1. **电路验证**
   1. **TestBench**

module testbench();

reg[3:0] i;

wire[2:0] o;

//仿真开始时，先定义输入为0010，输出应为001

initial begin

i=4'b0010;

//10纳秒以后，定义输入为0101，输出应为010

#10;

i=4'b0101;

//又过了10纳秒，定义输入为1000，输出应为011

#10;

i=4'b1000;

//又过了10纳秒，定义输入为1101，输出应为100

#10;

i=4'b1101;

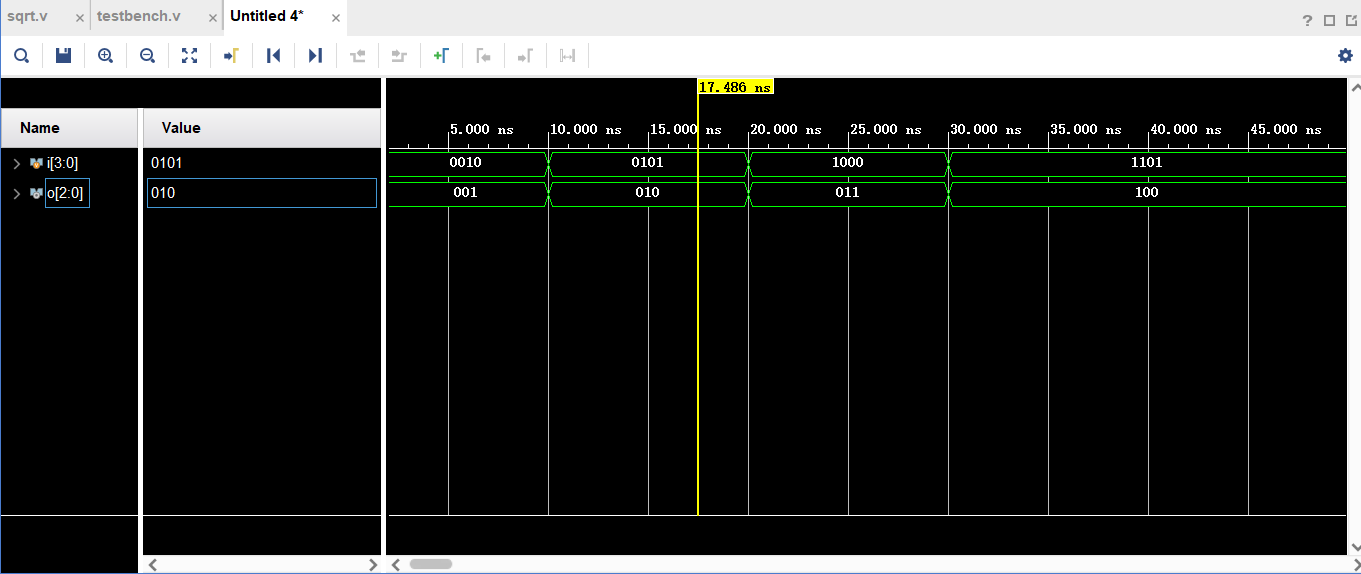
end

//寄存器i为输入，wire向量记录输出

sqrt test(i,o);

endmodule

* 1. **仿真结果**



数组i记录了求根电路模块的输入，数组o记录了求根电路模块的输出。在测试文件testbench中，首先定义输入i为0010，此时输出o为001；10纳秒以后，定义输入i为0101，此时输出o为010；10纳秒以后，改变输入i为1000，输出o相应变为011；10纳秒以后，改变输入i为1101，输出o变为100，均符合模块设计预期，可见该组合电路设计成功。

1. **实验心得**