Санкт-Петербургский Национальный Исследовательский Университет

Информационных Технологий, Механики и Оптики

Факультет ПИиКТ

Лабораторная работа 1

по дисциплине

«Функциональная схемотехника»

Вариант 3

Выполнил: студент группы Р33111

Киселёв Сергей Владимирович

г. Санкт-Петербург

2021 г.

**Задание**

1. Опишите на Verilog HDL на вентильном уровне модуль, реализующий функцию БОЭ в указанном логическом базисе согласно варианту задания.
2. Разработайте тестовое окружение для созданного модуля.
3. Проведите моделирование работы схемы.
4. Составьте отчет по результатам выполнения заданий второй части лабораторной работы.

**Вариант**

Схема мажоритарного контроля с 5-ю входами, состоящая из NOR

**Git**

https://github.com/kamikadze328/Circuit-Design/tree/main/lab1

**Код разработанного модуля БОЭ**

`timescale 1ns / 1ps

module lab1 (

input a1,

input a2,

input a3,

input a4,

input a5,

output out

);

wire not\_a1, not\_a2, not\_a3, not\_a4, not\_a5,

a1a2, a3a4, a1a5, a2a5, not\_a1a2, not\_a3a4, not\_a1a5, not\_a2a5,

b1, b2, b3, b4, b5, b6, b7, b8, b9, b10,

c1, c2, c3, c4, c5, c6, c7, c8, c9, c10,

not\_c1, not\_c2, not\_c3, not\_c4, not\_c5, not\_c6, not\_c7, not\_c8, not\_c9;

nor(not\_a1, a1, a1);

nor(not\_a2, a2, a2);

nor(not\_a3, a3, a3);

nor(not\_a4, a4, a4);

nor(not\_a5, a5, a5);

nor(a1a2, not\_a1, not\_a2);

nor(a3a4, not\_a3, not\_a4);

nor(a1a5, not\_a1, not\_a5);

nor(a2a5, not\_a2, not\_a5);

nor(not\_a1a2, a1a2, a1a2);

nor(not\_a3a4, a3a4, a3a4);

nor(not\_a1a5, a1a5, a1a5);

nor(not\_a2a5, a2a5, a2a5);

nor(b1, not\_a1a2, not\_a3);

nor(b2, not\_a1a2, not\_a4);

nor(b3, not\_a1a2, not\_a5);

nor(b4, not\_a3a4, not\_a1);

nor(b5, not\_a3a4, not\_a2);

nor(b6, not\_a3a4, not\_a5);

nor(b7, not\_a1a5, not\_a3);

nor(b8, not\_a1a5, not\_a4);

nor(b9, not\_a2a5, not\_a3);

nor(b10, not\_a2a5, not\_a4);

nor(c1, b1, b2);

nor(not\_c1, c1, c1);

nor(c2, not\_c1, b2);

nor(not\_c2, c2, c2);

nor(c3, not\_c2, b3);

nor(not\_c3, c3, c3);

nor(c4, not\_c3, b4);

nor(not\_c4, c4, c4);

nor(c5, not\_c4, b5);

nor(not\_c5, c5, c5);

nor(c6, not\_c5, b6);

nor(not\_c6, c6, c6);

nor(c7, not\_c6, b7);

nor(not\_c7, c7, c7);

nor(c8, not\_c7, b8);

nor(not\_c8, c8, c8);

nor(c9, not\_c8, b9);

nor(not\_c9, c9, c9);

nor(c10, not\_c9, b10);

nor(out, c10, c10);

endmodule

**Код разработанного тестового окружения БОЭ**

`timescale 1ns / 1ps

module adder\_tb ;

reg a1\_in, a2\_in, a3\_in, a4\_in, a5\_in;

wire out\_out;

lab1 lab1\_1 (

.a1(a1\_in),

.a2(a2\_in),

.a3(a3\_in),

.a4(a4\_in),

.a5(a5\_in),

.out(out\_out)

);

integer i ;

reg [4:0] test\_val;

reg expected\_val;

initial begin

for(i=0; i<32; i=i+1) begin

test\_val = i;

a1\_in = test\_val[0];

a2\_in = test\_val[1];

a3\_in = test\_val[2];

a4\_in = test\_val[3];

a5\_in = test\_val[4];

expected\_val = (a1\_in + a2\_in + a3\_in + a4\_in + a5\_in) > 2;

#10

if(out\_out == expected\_val) begin

$display("The lab1 output is correct!!!");

end else begin

$display("The lab1 output is wrong!!!");

end

$display("[a1, a2, a3, a4, a5] = [%b, %b, %b, %b, %b]; out=%b;\n", a1\_in, a2\_in, a3\_in, a4\_in, a5\_in, out\_out);

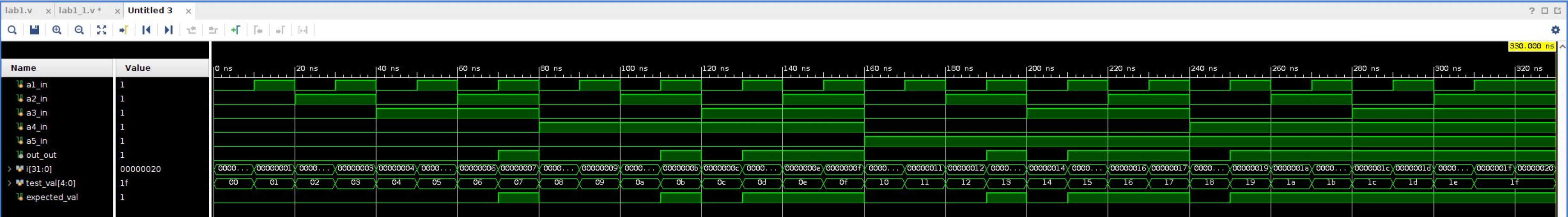
end

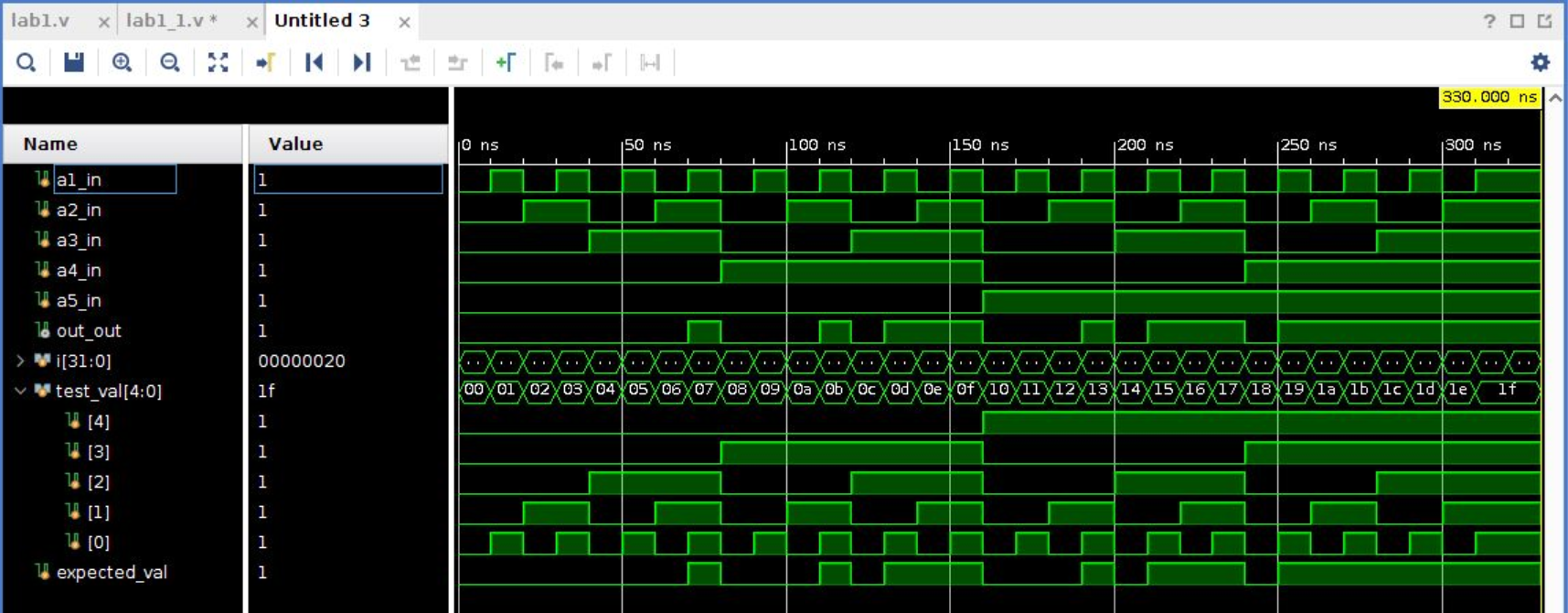
#10 $stop;

end

endmodule

**Временная диаграмма процесса тестирования БОЭ**

****

****