Санкт-Петербургский национальный исследовательский университет

информационных технологий, механики и оптики

**Лабораторная работа №1**

**Дисциплина «Функциональная схемотехника»**

**Выполнил:**

Провоторов Александр Владимирович

Группа P33112

**Преподаватель:**

Кустарев Павел Валерьевич

***Цель работы.***

1. Получить базовые знания о принципах построения цифровых интегральных схем с использованием технологии КМОП.
2. Познакомиться с технологией SPICE-моделирования схем на транзисторах.
3. Получить навыки описания схем базовых операционных элементов (БОЭ) комбинационного типа на вентильном уровне с использованием языка описания аппаратуры Verilog HDL.

# *Задание в соответствии с вариантом.*

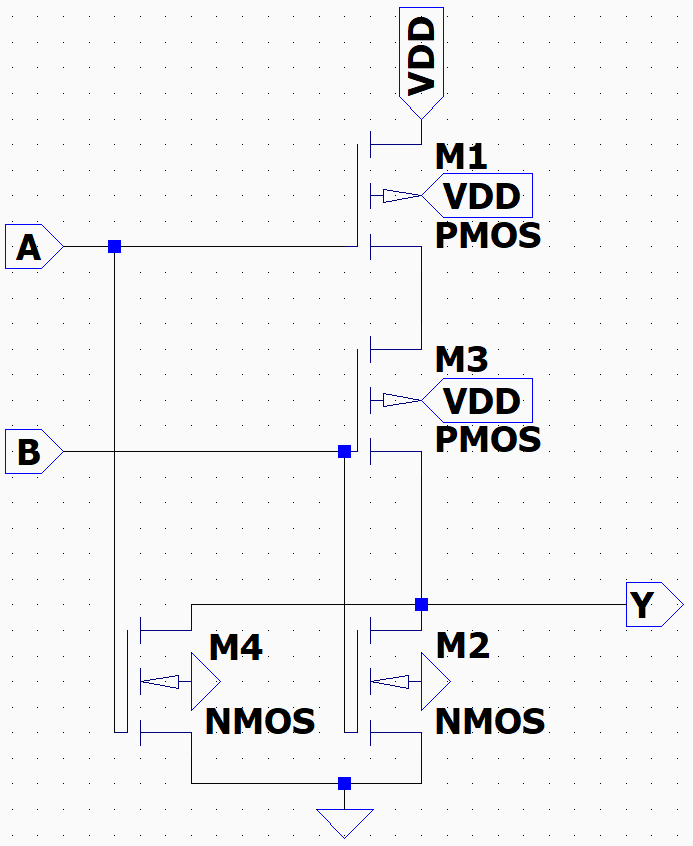
|  |  |  |
| --- | --- | --- |
| **№ варианта** | **Логический базис** | **БОЭ** |
| 9 | NOR | Мультиплексор «4 в 1» |

## Отчет о выполнении задания части 1.

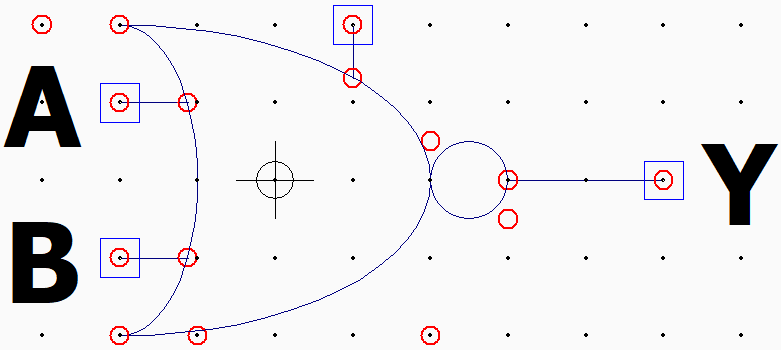
### Таблица истинности разработанного вентиля.

|  |  |  |
| --- | --- | --- |
| A | B | Y |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

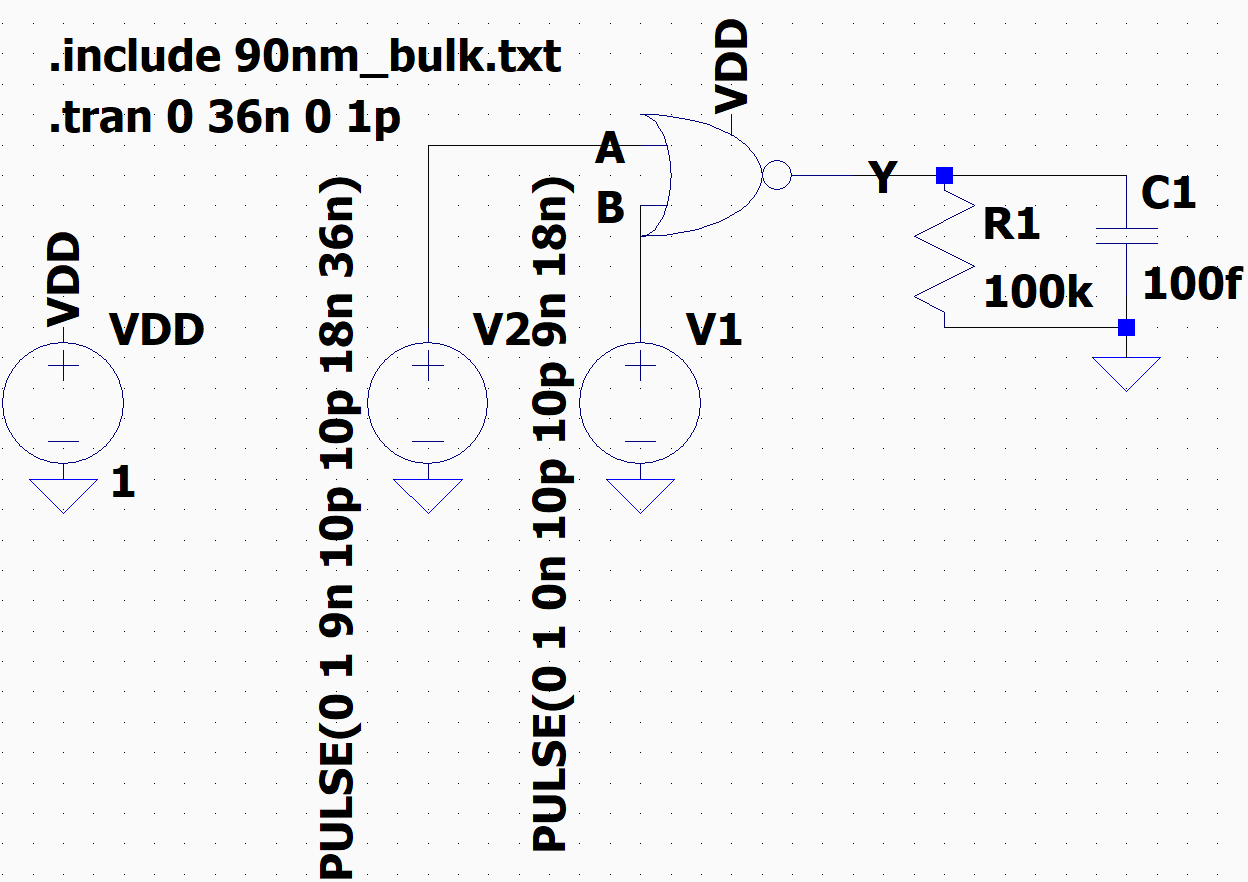
### Схема элемента ИЛИ-НЕ в LTSpice



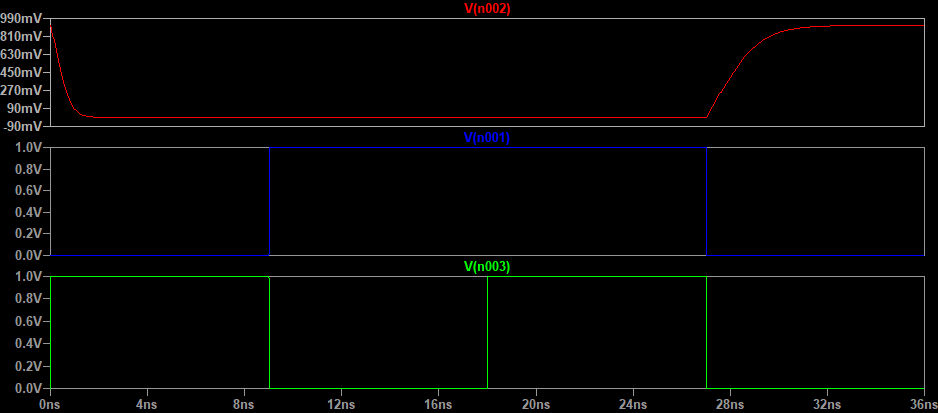
### Символ разработанного символа в соответствии со стандартом ANSI



### Схема тестирования, составленная из двух батареек с задержкой в пульсирующем сигнале, для достижения всевозможных комбинаций входных сигналов



### Временная диаграмма тестирования вентиля



### Результат измерения задержки распространения сигнала через вентиль

### 

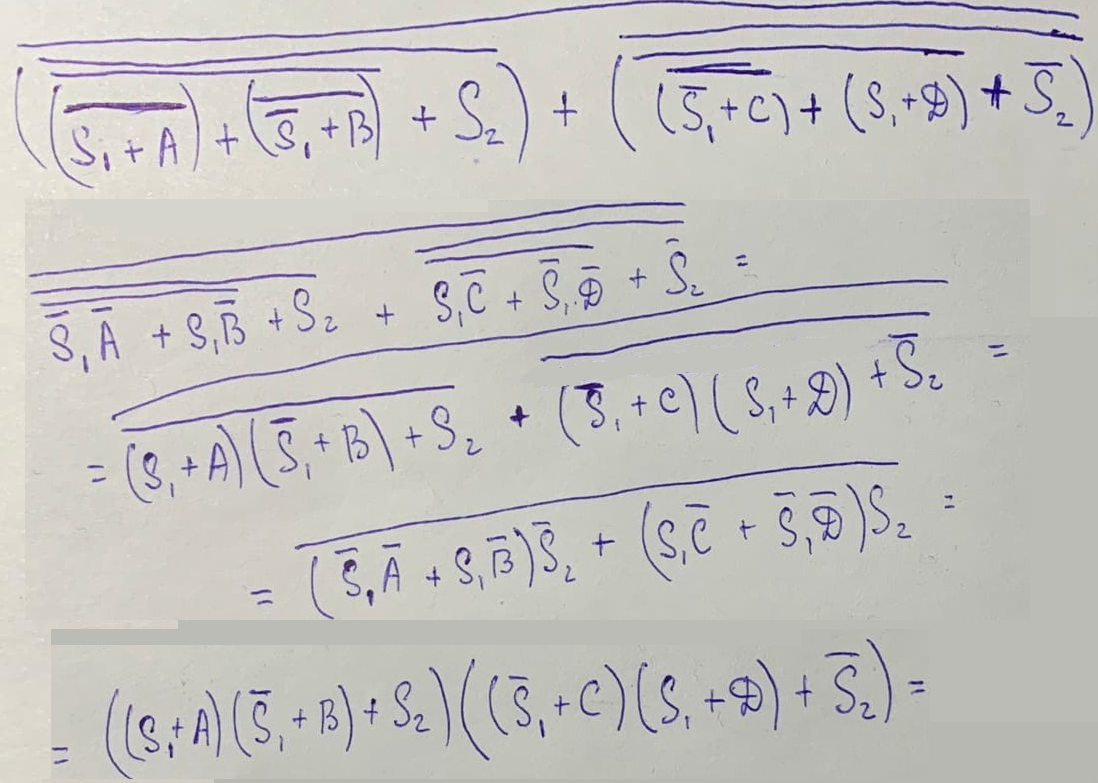
Задержка распространения сигнала: tcd=30.661433-27.22628 = 3.4532 нс

Максимальная частота работы вентиля: ν=1tcd=289.586 МГц

## Таблица истинности разработанного БОЭ.

Разработанное БОЭ – мультиплексор 4 в 1, который передает один из входных сигналов (A, B, C, D) на выход, согласно управляющему сигналу (S1, S2). Данный мультиплексор был реализован в базисе NOR.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| S1 | S2 | A | B | C | D | f |
| 0 | 0 | 0 | X | X | X | 0 |
| 0 | 0 | 1 | X | X | X | 1 |
| 0 | 1 | X | 0 | X | X | 0 |
| 0 | 1 | X | 1 | X | X | 1 |
| 1 | 0 | X | X | 0 | X | 0 |
| 1 | 0 | X | X | 1 | X | 1 |
| 1 | 1 | X | X | X | 0 | 0 |
| 1 | 1 | X | X | X | 1 | 1 |





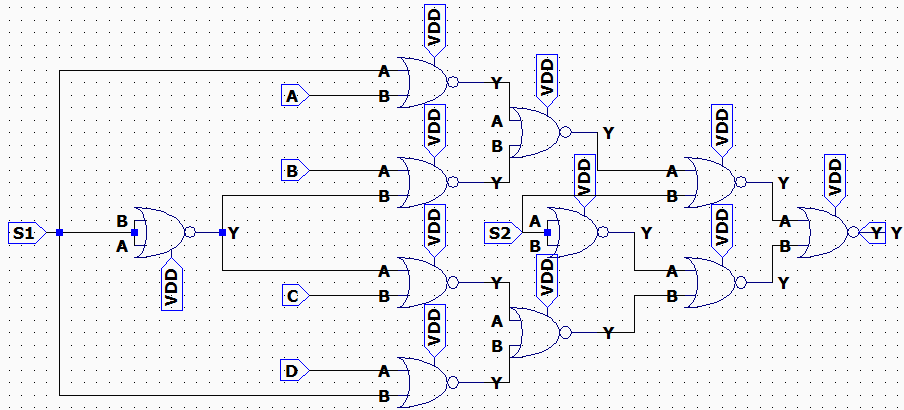
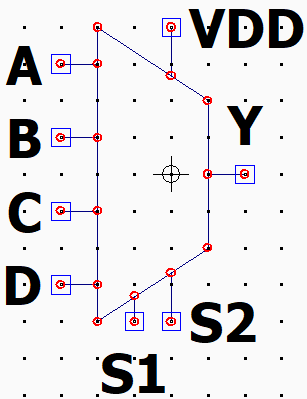
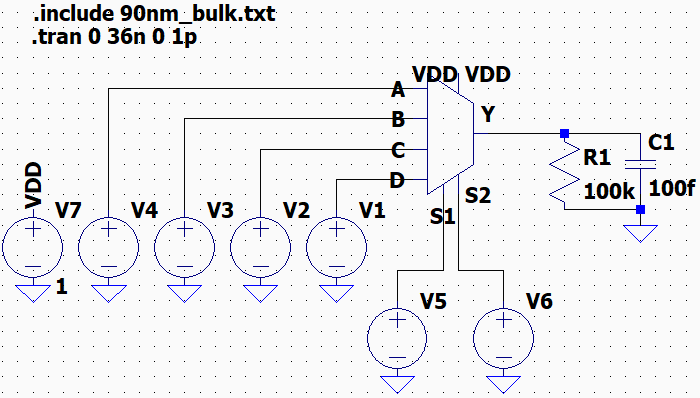


Схема мультиплексора 4 в 1, составленная мною в LTSpice в базисе NOR.

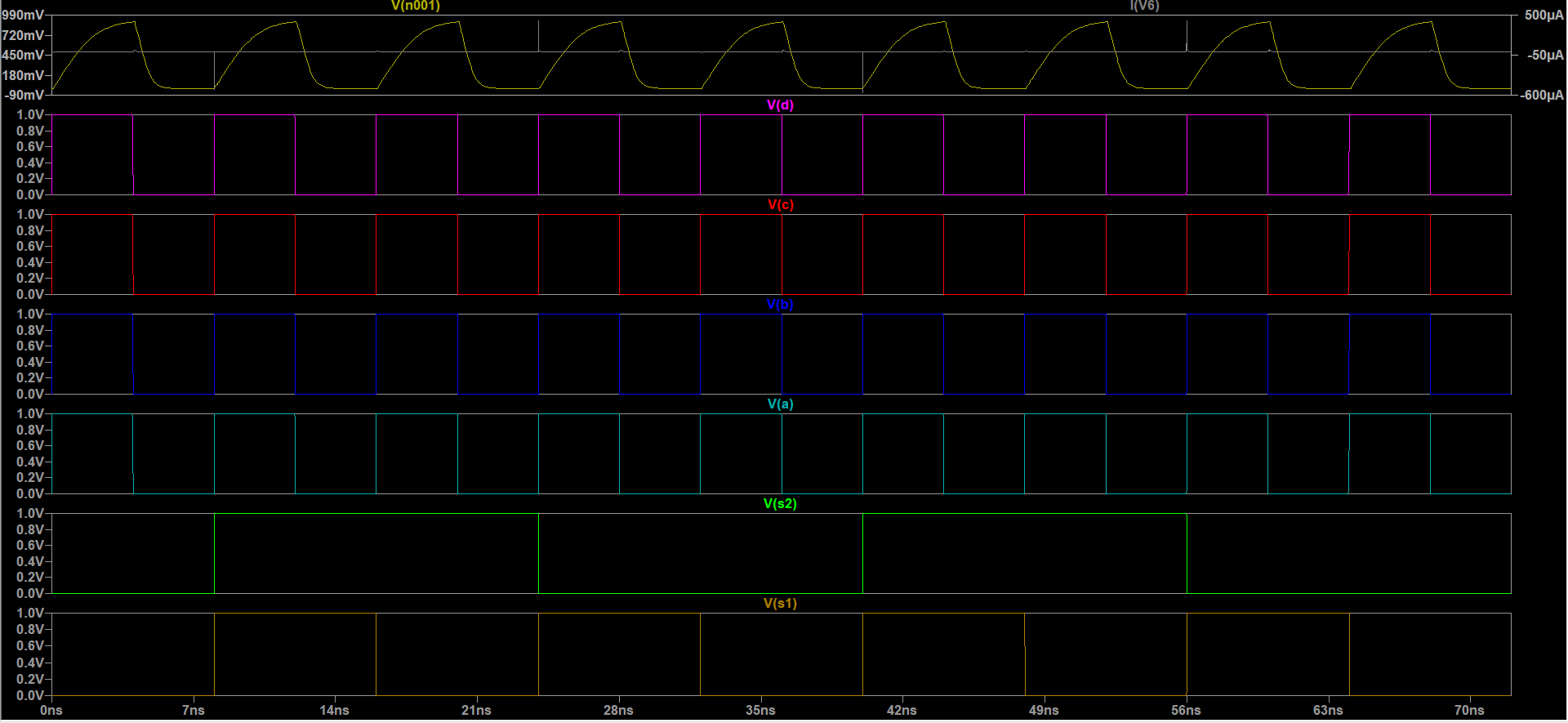
### Символ разработанного БОЭ



### Схема тестирования разработанного БОЭ



### Временная диаграмма процесса тестирования БОЭ.



### Результат измерения задержки распространения сигнала через БОЭ.

Задержка распространения сигнала: tcd=36-32.292906=3.7070938 нс

Максимальная частота работы вентиля: ν=1tcd=269.753 МГц

## Выводы по заданию части 1.

В ходе выполнения задания части 1, я познакомился с основными принципами работы в LTSpice, а также провел тестирование построенной модели. Как оказалось, многие физические законы хорошо моделируются данной программой и на основе этих данных можно точно моделировать реальные ситуации, что может быть полезно в условиях проектирования микросхем (очевидно при достаточно точном описании модели).

## Отчет о выполнении задания части 2.

### Код разработанного модуля БОЭ (реализует вышеописанную логику).

**`timescale 1ns / 1ps**

**module mux1to4(**

**input A,**

**input B,**

**input C,**

**input D,**

**output Y,**

**input S1,**

**input S2**

**);**

**wire nS1, nS2, nY;**

**wire [ 3 : 0 ] nvar\_tmp;**

**wire [ 1 : 0 ] nres1;**

**wire [ 1 : 0 ] nres2;**

**nor( nS1, S1, S1 );**

**nor( nS2, S2, S2 );**

**nor( nvar\_tmp[ 0 ], A, S1 );**

**nor( nvar\_tmp[ 1 ], B, nS1 );**

**nor( nvar\_tmp[ 2 ], C, nS1 );**

**nor( nvar\_tmp[ 3 ], D, S1 );**

**nor ( nres1 [ 0 ], nvar\_tmp [ 0 ], nvar\_tmp [ 1 ] );**

**nor ( nres1 [ 1 ], nvar\_tmp [ 2 ], nvar\_tmp [ 3 ] );**

**nor ( nres2 [ 0 ], nres1 [ 0 ], S2 );**

**nor ( nres2 [ 1 ], nres1 [ 1 ], nS2 );**

**nor ( Y, nres2 [ 0 ], nres2 [ 1 ] );**

**endmodule**

### Код разработанного тестового окружения БОЭ.

В данном тесте мы просто перебираем всевозможные сигналы и сравниваем полученное от модуля разработанного мультиплексора значение со значением по логике работы мультиплексора.

**`timescale 1ns / 1ps**

**module mux\_test;**

**reg S1i, S2i;**

**reg Ai, Bi, Ci, Di;**

**wire Y\_out;**

**mux1to4 mux\_test1 (**

**.Ai(Ai),**

**.Bi(Bi),**

**.Ci(Ci),**

**.Di(Di),**

**.Y(Y\_out),**

**.S1(S1i),**

**.S2(S2i)**

**);**

**integer i;**

**reg [ 5 : 0 ] test\_val;**

**reg [ 3 : 0 ] expected\_val;**

**initial begin**

**for ( i = 0; i < 64; i = i + 1 ) begin**

**test\_val = i;**

**S1i = test\_val[ 4 ];**

**S2i = test\_val[ 5 ];**

**Ai = test\_val[ 0 ];**

**Bi = test\_val[ 1 ];**

**Ci = test\_val[ 2 ];**

**Di = test\_val[ 3 ];**

**#10**

**if (**

**Y\_out == test\_val[ {test\_val[4] , test\_val[5]} ]**

**) begin**

**$display( "The demux output is correct!!! S1i=%b, S2i = %b, Ai = %b, Bi = %b, Ci = %b, Di = %b, Y\_out = %b", S1i, S2i, Ai, Bi, Ci, Di, Y\_out );**

**end else begin**

**$display( "The demux output is incorrect!!! S1i=%b, S2i = %b, Ai = %b, Bi = %b, Ci = %b, Di = %b, Y\_out = %b, expected\_val = %b", S1i, S2i, Ai, Bi, Ci, Di, Y\_out, test\_val[ {test\_val[4] , test\_val[5]} ] );**

**end**

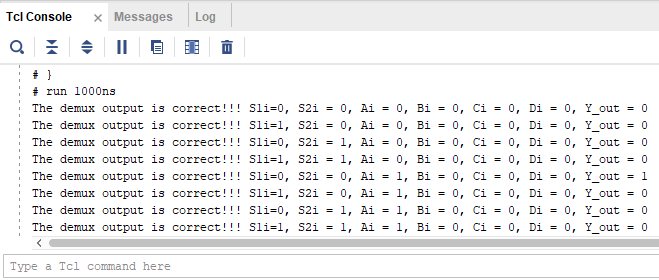
**end**

**#10 $stop;**

**end**

**endmodule**

### Пример вывода тестового покрытия



### Временная диаграмма процесса тестирования БОЭ.

## Вывод по заданию части 2.

В процессе проделанной работы я познакомился с основами реализации моделей схем в Vivado, а также спроектировал свою простейшую схему. Данная среда разработки имеет и плюс и минус – отсутствие надобности в графической реализации интерфейса, за счет чего удобнее видеть и прописывать, какая логика и для чего реализуются, но также и труднее понимать, как схема будет работать в живую и как ток по ней будет протекать.