

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение высшего образования
«**Национальный исследовательский университет ИТМО**»

Факультет Программной инженерии и компьютерной техники

Лабораторная работа №1
по дисциплине «Функциональная схемотехника»

Вариант: 7

Преподаватель:
Табунщик Сергей Михайлович

Выполнил:
Захарченко Р. В.
Перминов Ю. К.

Группа: Р3331

Санкт-Петербург, 2025

Оглавление

Цели работы.....	3
Вариант.....	3
Часть 1.....	3
Часть 2.....	6
Выводы:.....	8

Цели 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.

Вариант

7

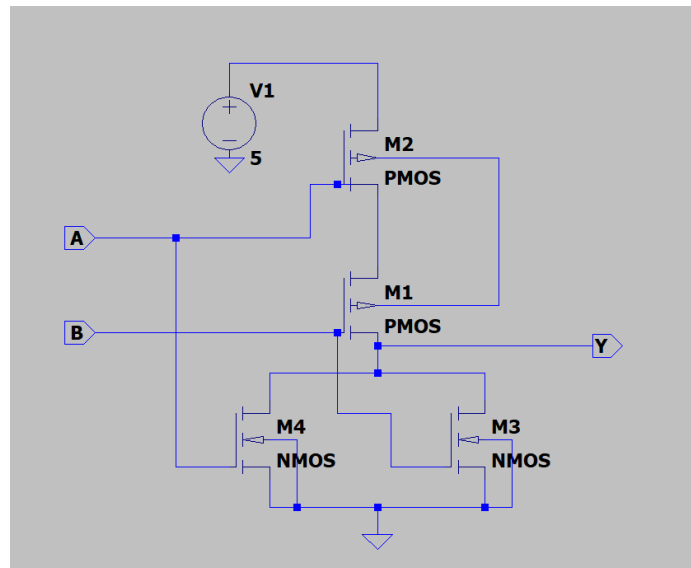
NOR

Четырехразрядный двоичный сумматор с переносом

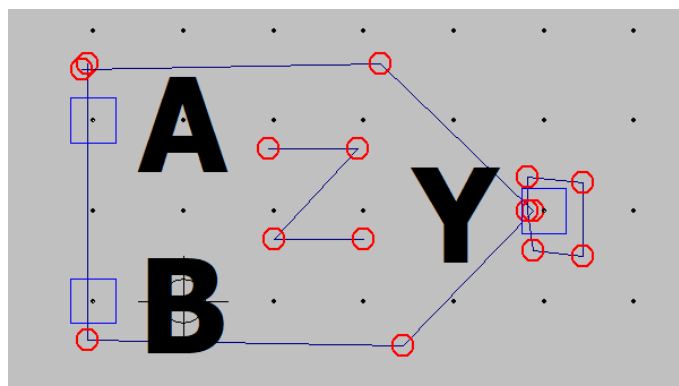
Часть 1

1. Постройте в LTspice на транзисторах схему вентиль, составляющего основу логического базиса согласно варианту задания

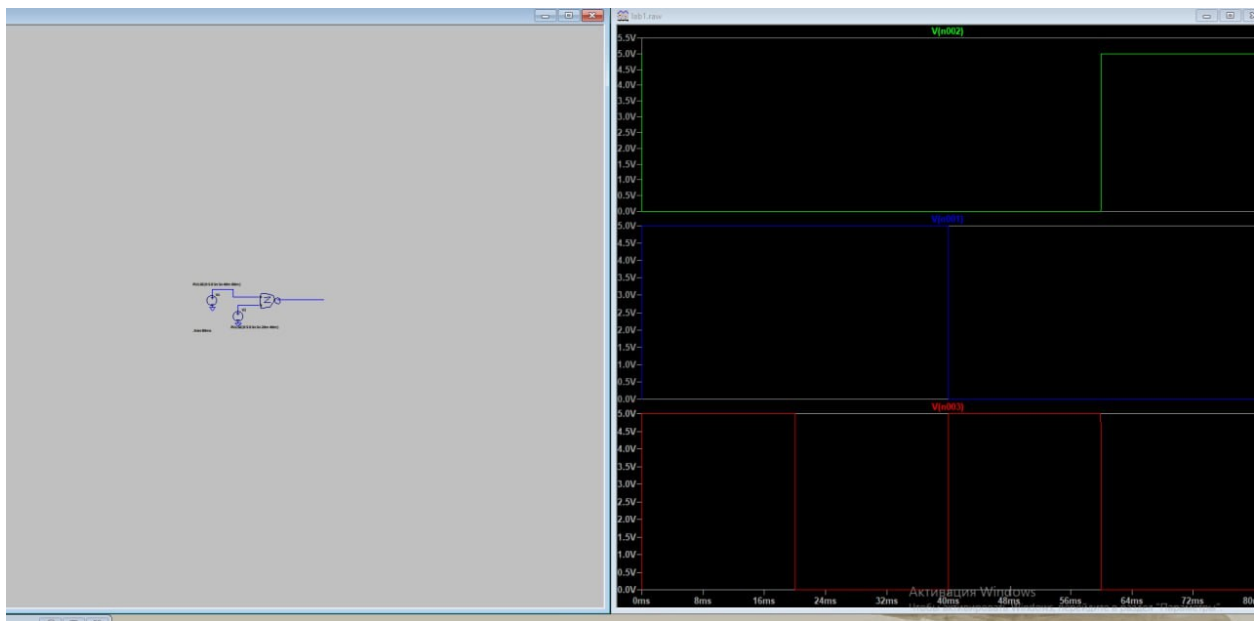
Согла



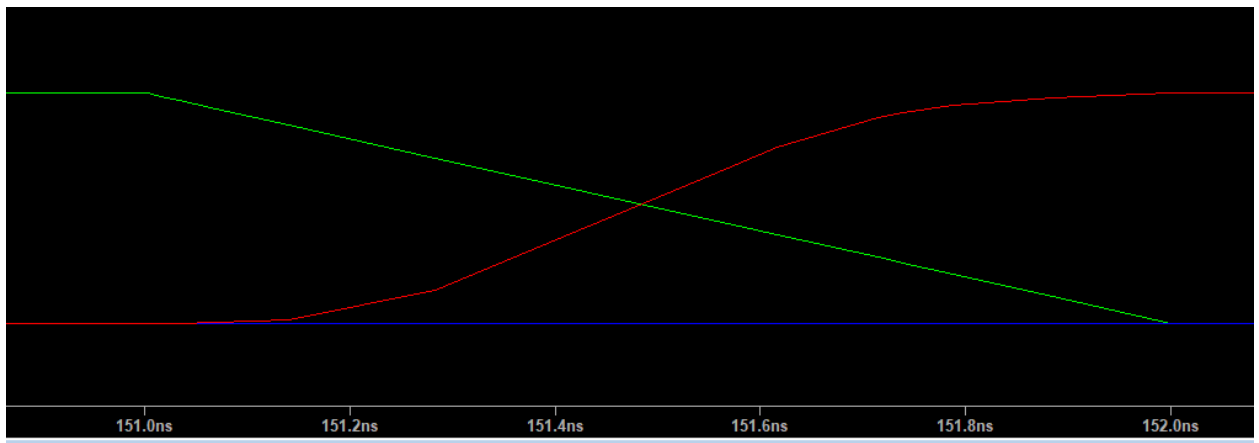
2. Создайте символ для разработанного вентиль как иерархического элемента.



3. С использованием созданного иерархического элемента постройте схему тестирования вентиля.



4. Проведите моделирование работы схемы и определите задержку распространения сигнала через тестируемый вентиль.



~1нс

5. Определите максимальную частоту изменения входных сигналов, при которой построенная схема сохраняет работоспособность.

$$f_{max} \approx \frac{1}{2 \cdot t_{pd}}$$

$$f_{max} = 1/(2 \cdot 1\text{нс}) = 500 \text{ MHz}$$

2. Ключевой параметр – задержка распространения tpd

- tpd измеряется как время от переключения входа (обычно по уровню 50% Vdd) до переключения выхода (тоже 50% Vdd).
- Для симметрии обычно берут среднее из tpd_rise и tpd_fall.

3. Максимальная частота

Максимальная частота изменения входных сигналов оценивается как:

$$f_{max} \approx \frac{1}{2 \cdot t_{pd}}$$

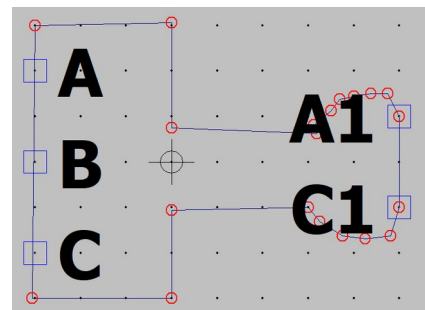
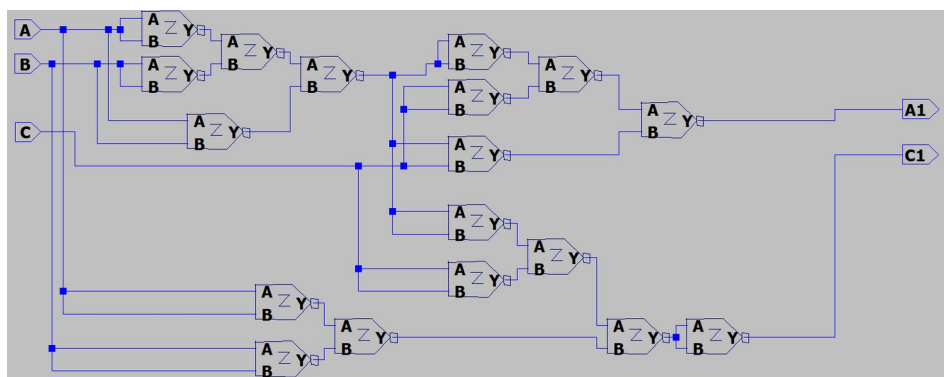
Почему делим на 2: вход должен держать состояние хотя бы один полный период «0→1→0» (т.е. high + low), иначе выход не успеет перейти туда-обратно.

Иногда пишут приближённо:

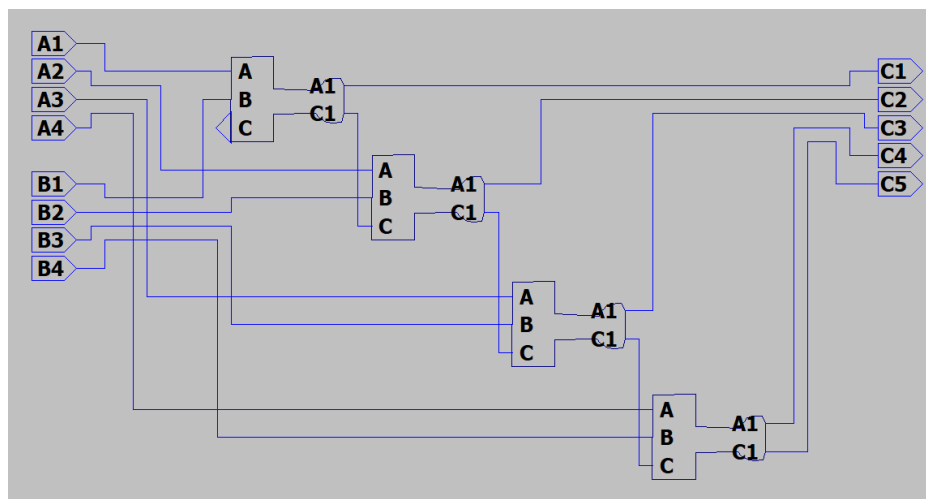
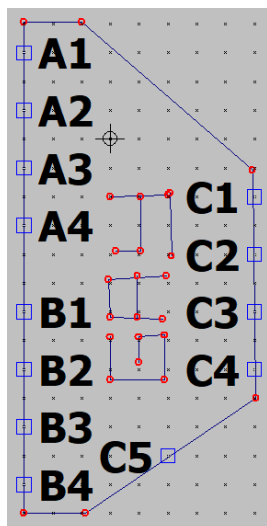
6. Постройте БОЭ на базе созданного вентиля согласно варианту задания.

7. Создайте символ для построенного БОЭ.

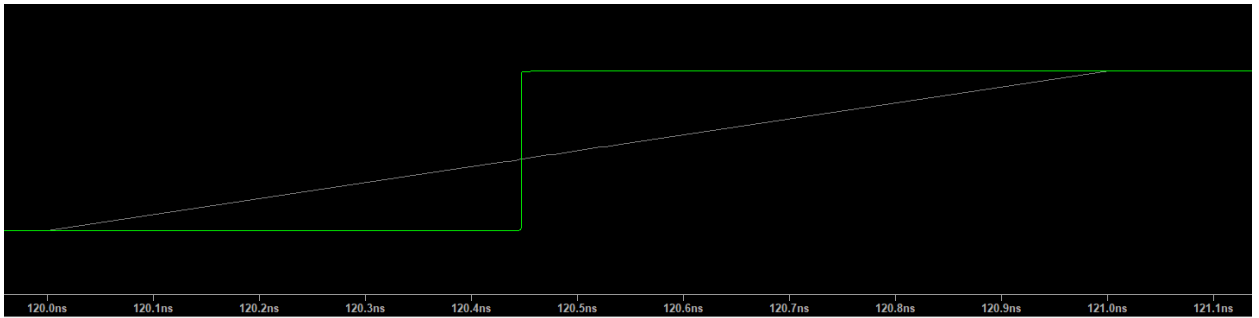
Сумматор для одного бита



Сумматор для двух 4х битных чисел



8. Проведите моделирование работы схемы и определите задержку распространения сигнала через БОЭ.



Видимо 0 ~ ~

тк ltspice не сработал, посчитаем гипотетически, dt для Nor'a – 1нс, а максимальное количество вентилей для прохождения сигнала через всю схему – 28, так что dt для всей схемы – 28 нс

9. Определите максимальную частоту изменения входных сигналов, при которой построенная схема сохраняет работоспособность

$$1/(2 * 28\text{нс}) = 17.8 \text{ MHz}$$

Часть 2

1. Опишите на Verilog HDL на вентильном уровне модуль, реализующий функцию БОЭ в указанном логическом базисе согласно варианту задания.

Код находится в гитхабе по данной ссылке:

https://github.com/tinunadno/functional_circuit_design_lab_1/2task/four_bit_adder_nor.v

2. Разработайте тестовое окружение для созданного модуля.

Код находится в гитхабе по данной ссылке:

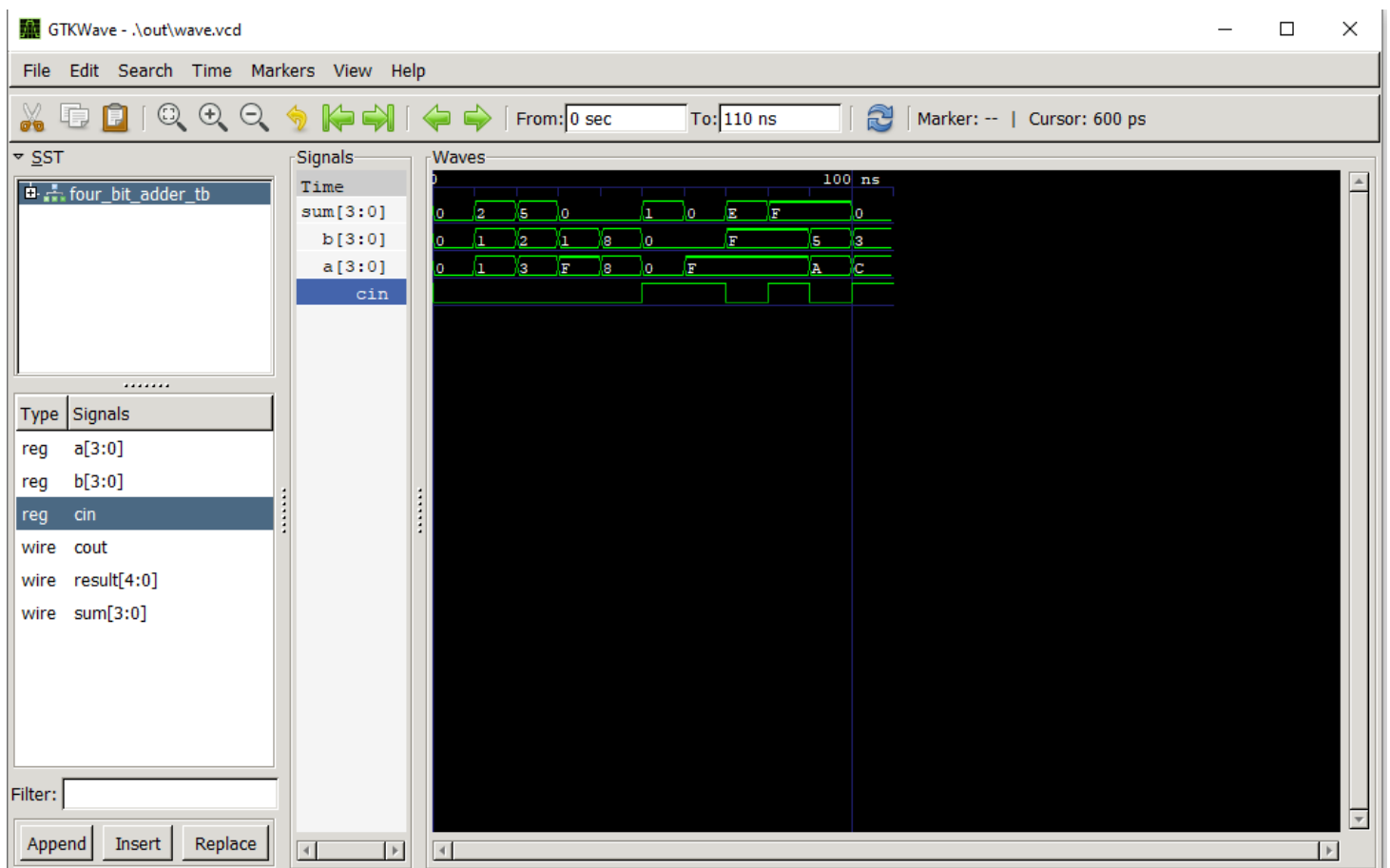
https://github.com/tinunadno/functional_circuit_design_lab_1/2task/four_bit_adder_nor_tb.v.v

3. Проведите моделирование работы схемы.

Результаты в консоли:

```
Time: 0 A=0000 B=0000 Cin=0 → Sum=0000 Cout=0 Result=00000
Time: 10000 A=0001 B=0001 Cin=0 → Sum=0010 Cout=0 Result=00010
Time: 20000 A=0011 B=0010 Cin=0 → Sum=0101 Cout=0 Result=00101
Time: 30000 A=1111 B=0001 Cin=0 → Sum=0000 Cout=1 Result=10000
Time: 40000 A=1000 B=1000 Cin=0 → Sum=0000 Cout=1 Result=10000
Time: 50000 A=0000 B=0000 Cin=1 → Sum=0001 Cout=0 Result=00001
Time: 60000 A=1111 B=0000 Cin=1 → Sum=0000 Cout=1 Result=10000
Time: 70000 A=1111 B=1111 Cin=0 → Sum=1110 Cout=1 Result=11110
Time: 80000 A=1111 B=1111 Cin=1 → Sum=1111 Cout=1 Result=11111
Time: 90000 A=1010 B=0101 Cin=0 → Sum=1111 Cout=0 Result=01111
Time: 100000 A=1100 B=0011 Cin=1 → Sum=0000 Cout=1 Result=10000
```

Результаты в GTKWave:



Выводы:

В ходе лабораторной работы мы сконструировали логический элемент nor и на его основе построили сумматор 4х битных чисел, затем мы построили модель сумматора и элемента в ltspice, замерыли минимальное время переключения и максимальную частоту переключения, так же мы протестировали схему на правильность. Далее мы описали на языке verilog нашу БОЭ в оказанном логическом базисе и разработали для нашей БОЭ тестовое окружение, провели моделирование и убедились в нашей крутости и в том что всё работает.

В ходе лабораторной очень близко (нехотелось конечно на столько близко) познакомились с ltspice и языком verilog