САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Дисциплина: Архитектура ЭВМ

Отчет

по домашней работе №2

**«Построение сложных логических схем»**

Выполнил: Шеметов Алексей Игоревич

Номер ИСУ: 338978

студ. гр. M3134

Санкт-Петербург

2021

**Цель работы:** моделирование сложных логических схем.

**Инструментарий и требования к работе:** работа выполняется в logisim.

**Синхронный вычитающий счетчик с параллельным переносом**

Для начала разберемся, что вообще из себя представляет синхронный вычитающий счетчик с параллельным переносом.

**Счетчик** – это устройство, на выходах которого получается двоичный код, определяемый числом поступивших импульсов.

**Вычитающий счетчик** (счетчик обратного действия) – счетчик, выходной двоичный код которого уменьшается на единицу с приходом каждого входного импульса.

**Синхронный счетчик с параллельным переносом** – счетчик, у которого при подаче импульса значение меняется сразу на всех триггерах.

Соответственно **синхронный вычитающий счетчик с параллельным переносом** – это счетчик, выходной двоичный код которого уменьшается на единицу с приходом каждого входного импульса, причем импульс поступает сразу на все триггеры.

Перейдем к составлению схемы данного счетчика. В качестве триггеров я решил использовать jk-триггеры. Так как в счетчике мы подаем один и тот же импульс на j и k, то мы сможем получать только 0 или 1 либо будем сохранять предыдущее значение и инвертировать его. В моей схеме присутствует копка reset, которая устанавливает на выходе каждого jk-триггера 1, соответственно лучше использовать вариант с сохранением и инверсией, потому что так мы сможем хранить наши значения, когда на триггеры ничего не поступает, в противном случаем будем просто инвертировать предыдущее значение. Соединять триггеры будем последовательно, передавая в качестве импульса 1, если значение предыдущего триггера равно 1 и значение прошлого условия Q & Qпред, в противном случае импульс будет равен 0 (так в счетчике реализовывается параллельный перенос). В счетчике еще должен присутствовать модуль счета 15. То есть у младшего разряда вместо Q будем выводить не Q, тогда следующему триггеру передадим Q. Еще когда счетчик дойдет до 15, то нужно сделать сброс, таким образом мы вместо значения 15 перейдем сразу в значение 14 (из-за того, что у младшего элемента поменяли Q и не Q). Обозначение на схеме можно увидеть на рисунке №1.

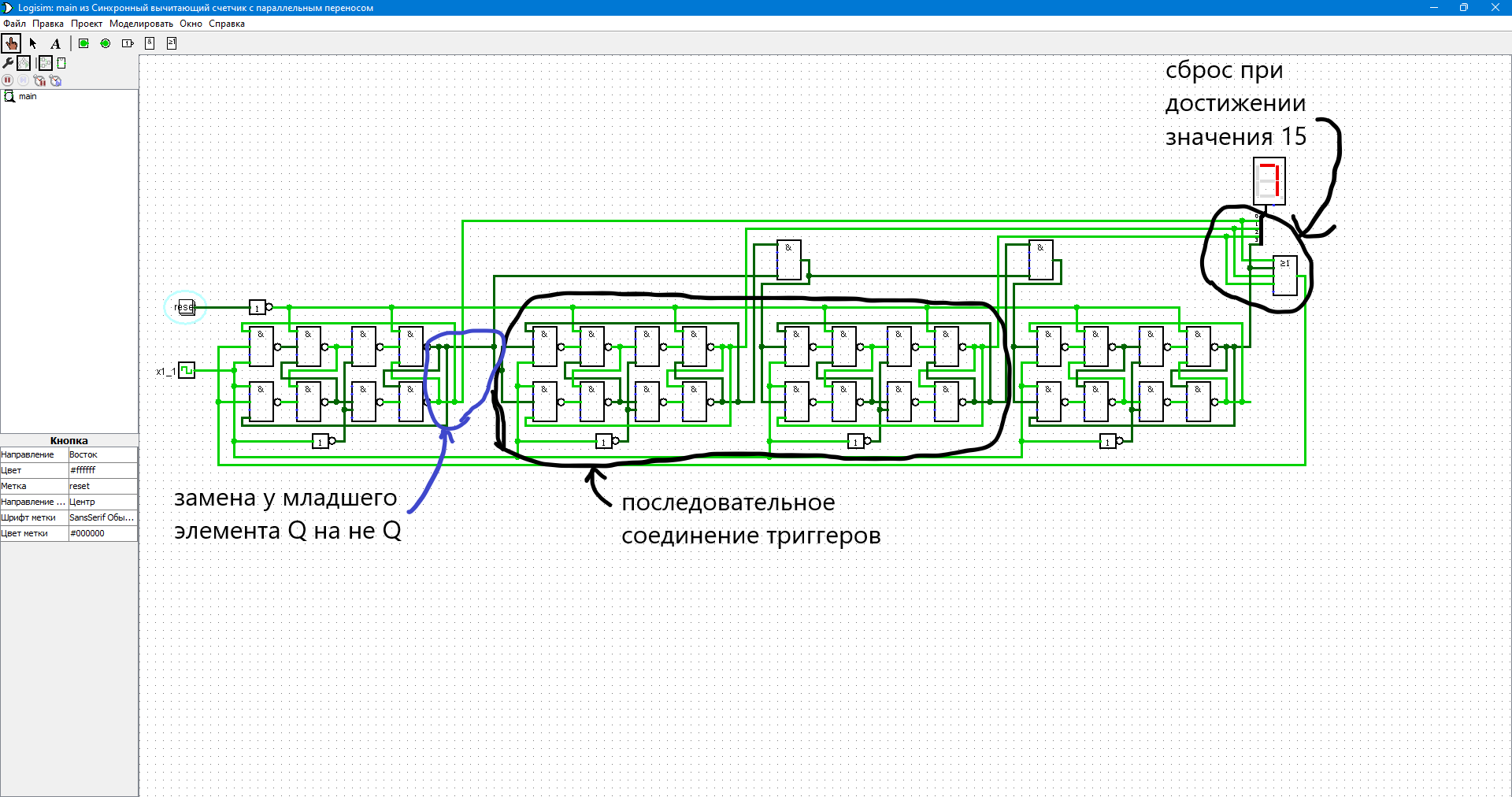


Рис. №1 - Схема синхронного вычитающего счетчика с параллельным переносом + подписи к пояснению реализации

Теперь давайте разберемся с тем, как работает счетчик. Изначально нужно установить значения на jk-триггерах, то есть нажать кнопку reset иначе значение на триггерах будет неопределенно. После с поступлением каждого такта меняется значение каждого из триггеров, а именно, если у прошлый триггер меняет значение с 0 на 1, то данный триггер меняет свое значение с 1 до 0. У младшего триггера значение меняется с приходом каждого импульса равного 1. Ниже приведена временная диаграмма для данного счетчика (рис 2).

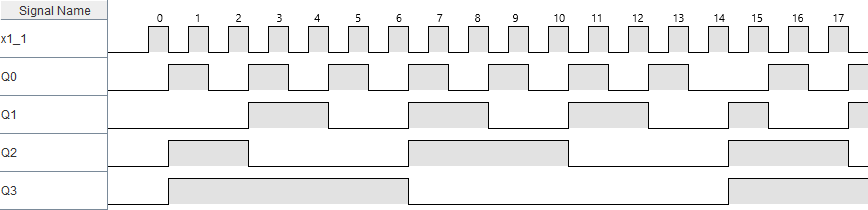


Рис. №2 - Временная диаграмма вычитающего счетчика с параллельным переносом

**Построение схемы взятия квадратного корня из 8-ми битного двоичного числа**

Для построения данной схемы составим таблицу истинности (таблица №1) и для каждого значения Q3…0 будем строить МДНФ (рис. №3-6). Q3 \* 8 + Q2 \* 4 + Q1 \* 2 + Q0 – результат квадратного корня. В таблице №1 показаны только строки, в которых прошлое значение (Q3…0) отличается от нынешнего (значение предварительно было округленно к 0 и переведено к 2-ичной системе, где старший разряд соответствует Q3 и т.д.).

Таблица №1 – таблица истинности для корня

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| X7 | X6 | X5 | X4 | X3 | X2 | X1 | X0 | Q3 | Q2 | Q1 | Q0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 |
| 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |

На вход схеме подается импульс, задаваемый контактами X7…0 (ниже присутствует 16-тиричный индикатор для лучшего представления числа). Выход показан на 16-тиричном индикаторе, расположенном справа от схемы (Q3 подключен к 3-ему разряду, Q2 ко 2 и т.д.).

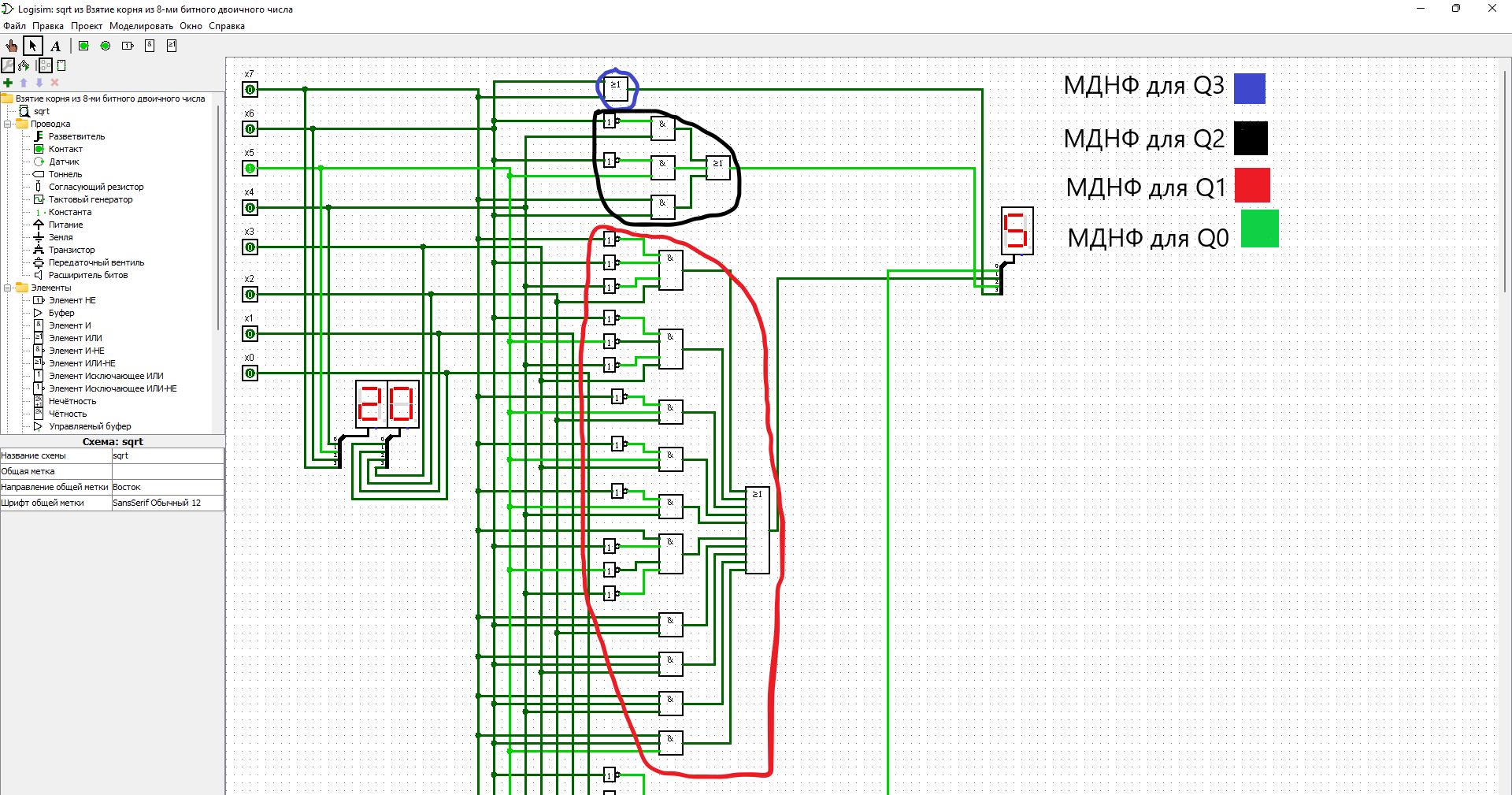


Рис. №3 – схема извлечения квадратного корня 1ч.

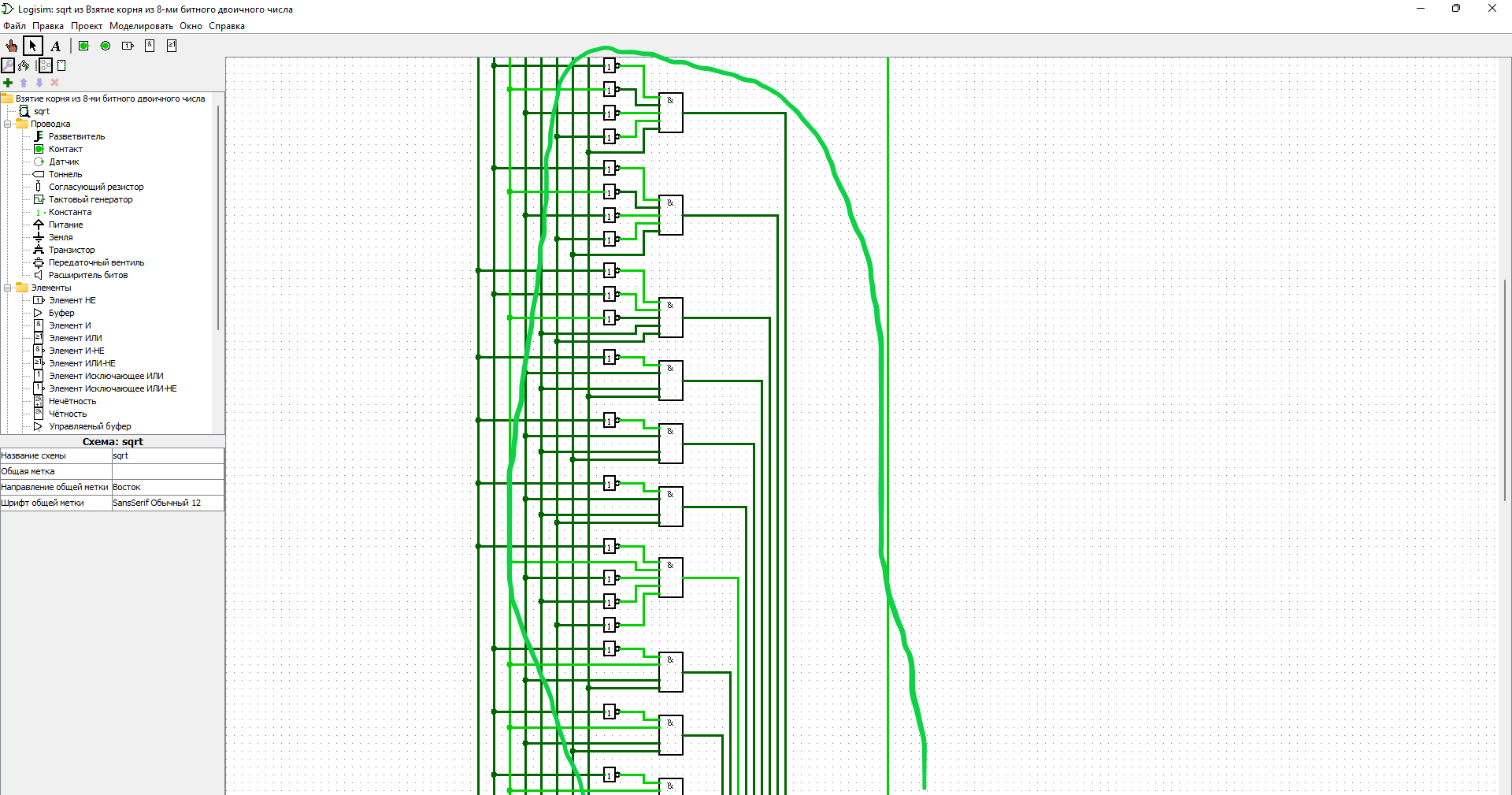


Рис. №4 – схема извлечения квадратного корня 2ч.

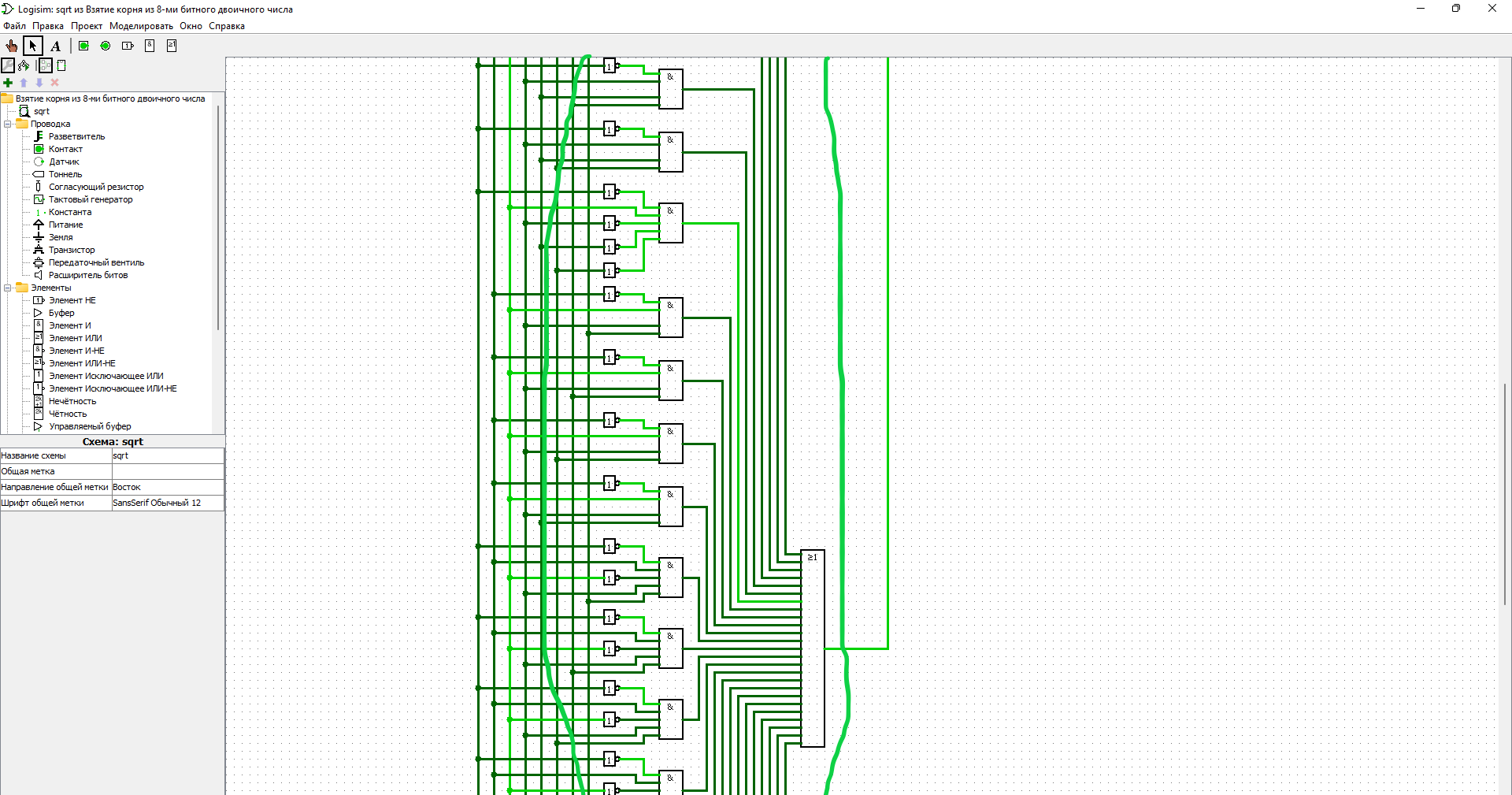


Рис. №5 – схема извлечения квадратного корня 3ч.

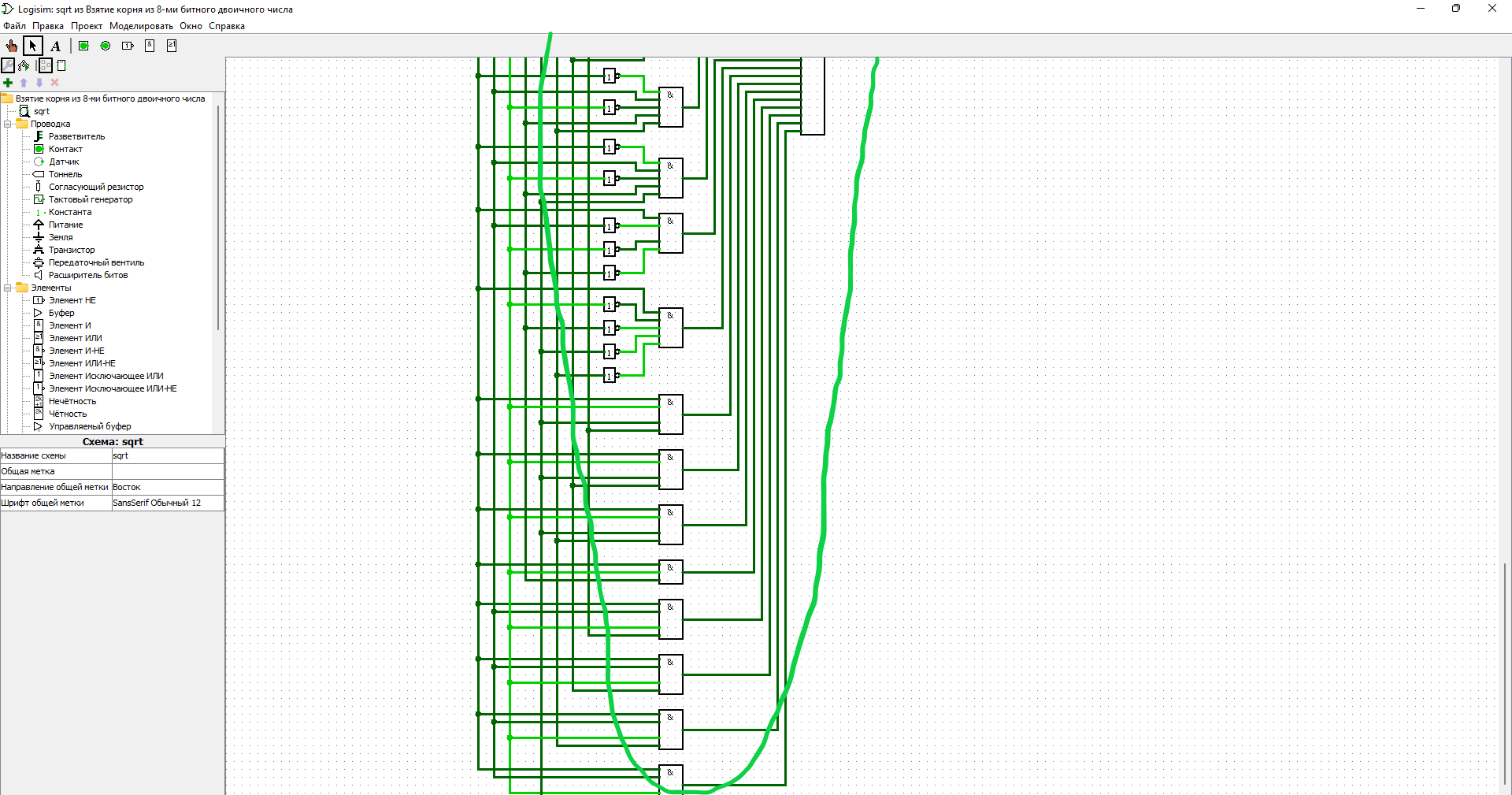


Рис. №6 – схема извлечения квадратного корня 4ч.