

Лабораторная работа №1	М3136	2022
Построение логических схем в среде моделирования	Артемьев Иван Вадимович	

**Цель работы:** моделирование логических схем на элементах с памятью.

**Инструментарий и требования к работе:** работа выполняется в среде моделирования Logisim evolution.

**Описание:** Необходимо сделать две схемы: счетчик и регистр сдвига с линейной обратной связью.

**Вариант:** Счетчик: Синхронный суммирующий счетчик с модулем счета – 13. Регистр с линейной обратной связью: тип конфигурации: Галуа, конфигурация: (8, 6, 5, 4).

### Счетчик

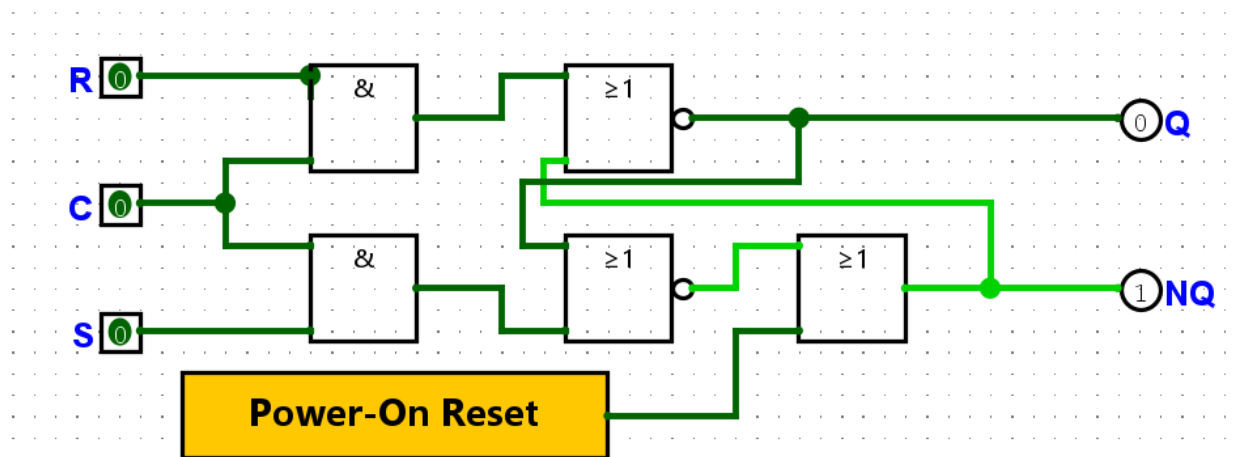


Рисунок №1 – схема синхронного RS-триггера

Синхронный RS-триггер(выше приведена его схема – подсхема в проекте называется RS): это триггер со следующей таблицей истинности (см. таблица №1). (если  $s = 0$ , то сохранение)  $s = 1$ :

S	R	Q
0	0	сохр.
0	1	0
1	0	1
1	1	-

Таблица №1 – таблица истинности RS-триггера

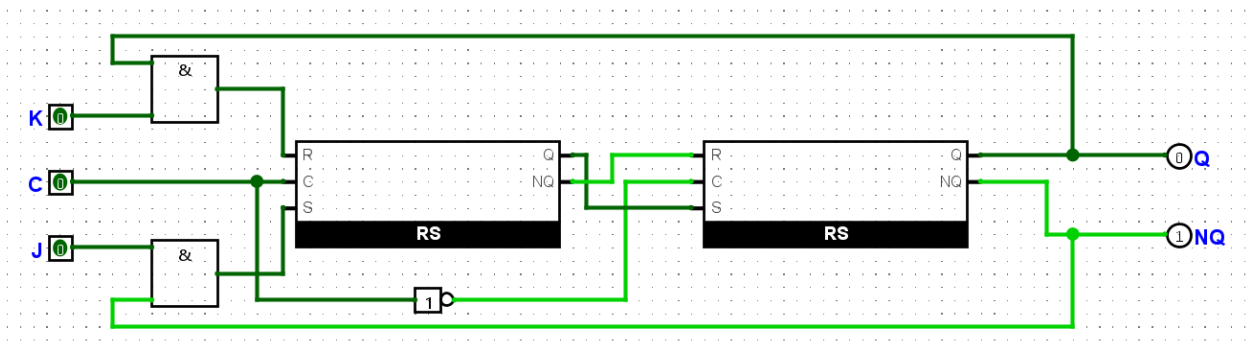


Рисунок №2 – схема синхронного JK-триггера

JK-триггер(выше приведена его схема – подсхема в проекте называется JK): это триггер со следующей таблицей истинности.

J	K	Q
0	0	сохр.
0	1	0
1	0	1
1	1	инв.

Таблица №2 – таблица истинности JK-триггера

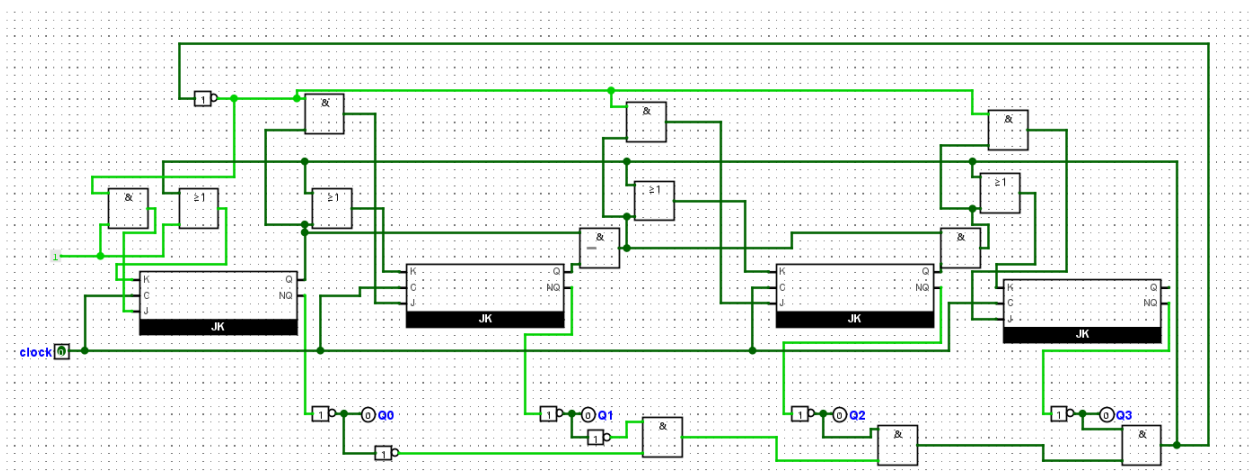


Рисунок №3 – схема синх. сумм. счетчика

Мне достался синхронный суммирующий счетчик (моя схема приведена сверху – подсхема в проекте называется SSC). По сути, суммирующий счетчик – это схема, которая считает количество импульсов, переданное ей. Синхронность в этой схеме означает, что входные импульсы поступают к разрядам одновременно, при этом все разряды переходят в новые состояния одновременно. Модуль счета счетчика – это значение, после которого все регистры обнуляются и счет начинается заново. В моём случае это число 13, а это значит, что после 12(1100) идёт 0 (0000). Его таблица истинности выглядит следующим образом (см. Таблицу №3):

№ импульса	Q3	Q2	Q1	Q0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	0	0	0	0
14	0	0	0	1
15	0	0	1	0
16	0	0	1	1
...	...	...	...	...

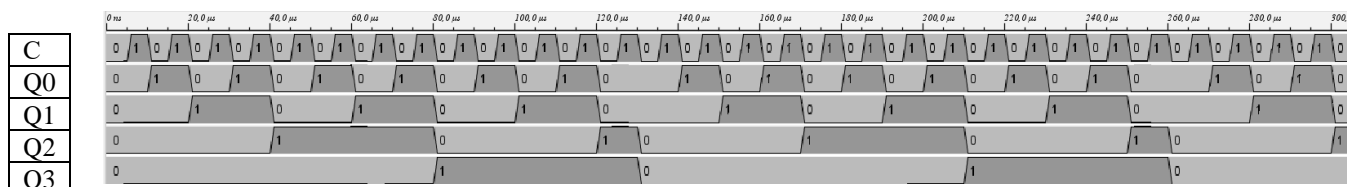
Таблица №3 – Таблица истинности счётчика (mod 13)

Сперва реализуем счетчик без модуля 13, в котором будут 4 регистра, то есть модулем будет число 16 (если мы сделаем схему с тремя регистрами, то максимальное число в ней, которое можно получить это будет 7, что меньше 13, поэтому необходимо минимум 4 триггера).

Итак, если бы мы каждому новому JK-триггеру подавали на входы J и K значение Q предыдущего триггера, то когда на предыдущем была единица, он бы тоже менялся, но нам необходимо, чтобы когда на всех предыдущих единица он менялся (инвертировался), поэтому мы делаем «И» от всех предыдущих значений триггеров.

Теперь у нас получился счетчик с модулем 16, сделаем из него счётчик с модулем 13, следующим образом: когда мы доходим до 12, то есть регистры равны 1, 1, 0, 0, мы обнуляем все JK-триггеры. Для этого мы, зная что если на вход J подать 0, а на вход K подать 1, результат Q будет 0.

Ниже приведена временная диаграмма:



## Регистр сдвига с линейной обратной связью

Мне достался регистр с типом конфигурации Галуа и самой конфигурацией (8, 6, 5, 4). Как работает РСЛОС? Сперва, регистр сдвига – это просто схема, которая сдвигает биты к следующим разрядам, а первый в который ничего не сдвигается мы сами подаём какое-то значение, последний мы выводим. А если мы добавим регистру линейную обратную связь, то нам больше не придётся ему что-либо передавать при каждом сдвиге, потому что теперь в самый первый бит мы будем подавать значение, полученное из последующих битов. Конкретно, в типе конфигурации Галуа, мы берём последний бит и делаем XOR вместе с битами, которые соответствуют заданной конфигурации. Для реализации хранения бита мы используем два D-триггера. (я его назвал MyTrigger)

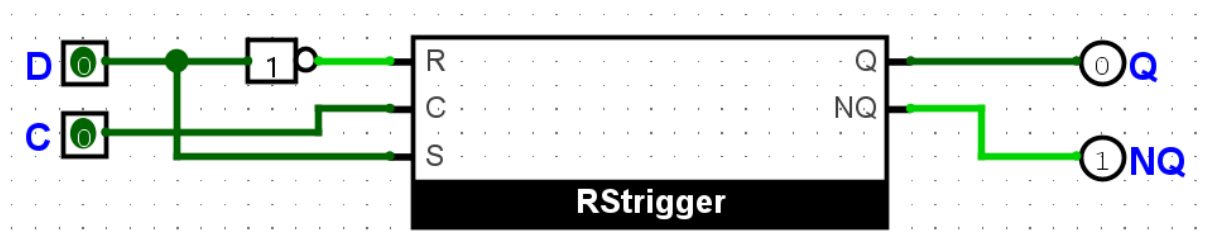


Рисунок №6 – D-триггер

D-триггер(его схема приведена сверху – подсхема называется Dtrigger):  
описание RS-триггера есть выше. Ниже приведена таблица истинности D-триггера:

C	D	Q
1	0	0
1	1	1
0	x	сохр.

Таблица №4 – таблица истинности D-триггера

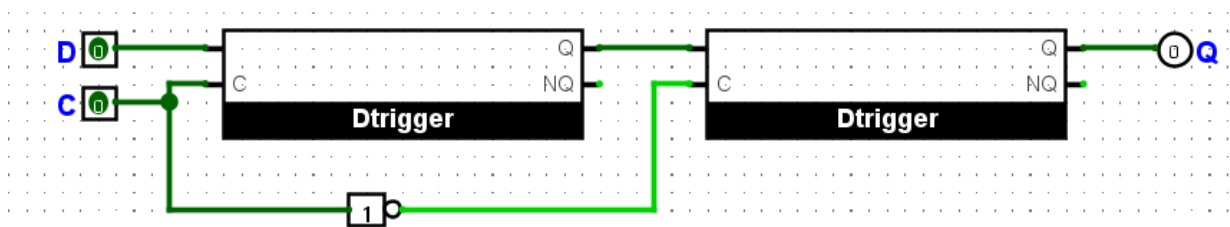
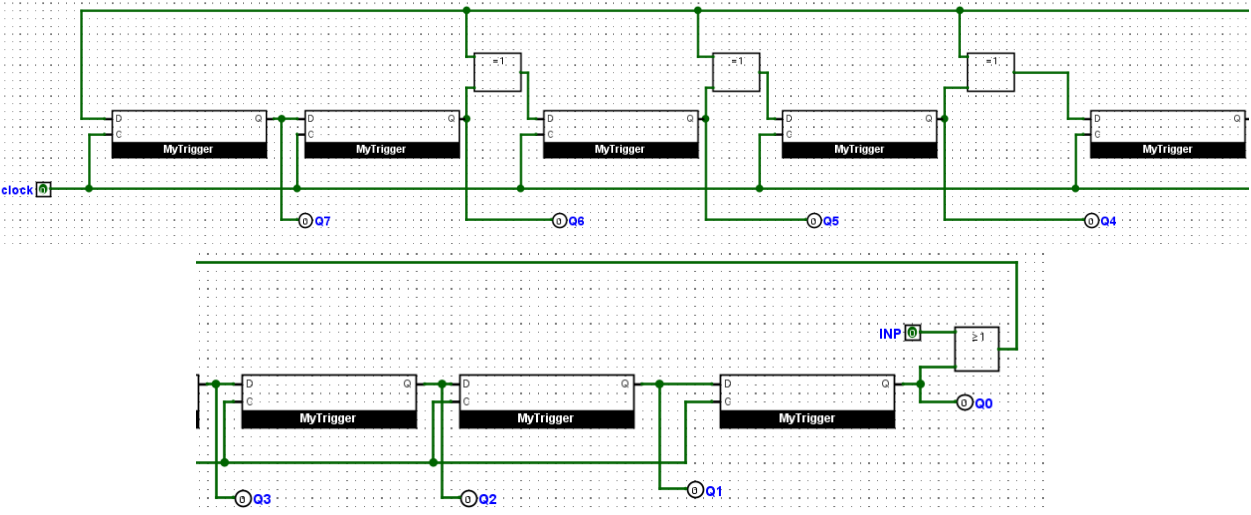


Рисунок №7 – схема MyTrigger

MyTrigger(его схема приведена сверху – подсхема называется MyTrigger):

По сути он является «буфером», когда на C подаётся импульс, этот триггер сохраняет новое переданное ему значение из D, и выдаёт предыдущее переданное ему значение.

Ниже я привожу свою схему РСЛОС(в проекте подсхема называется – registr)



Рисунки №8,9 – схема РСЛОС (конф. Галуа)

Для того чтобы передать какие-то начальные значения, я создал дополнительный вход INP, который если выставить на 1, то будет передаваться значение на провод, который идёт к предыдущим триггерам.

Ниже я привожу таблицу истинности для РСЛОС, если начальным значением передать одну единичку.

№ имп.	Q0	Q1	Q2	Q3	Q4	Q5	Q6	Q7
0	0	0	0	1	1	1	0	1
1	0	0	1	1	1	0	1	0
2	0	1	1	1	0	1	0	0
3	1	1	1	0	1	0	0	0
4	1	1	0	0	1	1	0	1
5	1	0	0	0	0	1	1	1
6	0	0	0	1	0	0	1	1
7	0	0	1	0	0	1	1	0
8	0	1	0	0	1	1	0	0
9	1	0	0	1	1	0	0	0
10	0	0	1	0	1	1	0	1
11	0	1	0	1	1	0	1	0
12	1	0	1	1	0	1	0	0
13	0	1	1	1	0	1	0	1
14	1	1	1	0	1	0	1	0
15	1	1	0	0	1	0	0	1
16	1	0	0	0	1	1	1	1
...	...	...	...	...	...	...	...	...

Таблица № 5 – начало таблицы истинности РСЛОС