

ЛАБОРАТОРНАЯ РАБОТА №1	М3138	2023
ПОСТРОЕНИЕ ЛОГИЧЕСКИХ СХЕМ В СРЕДЕ МОДЕЛИРОВАНИЯ	ПОПОВИЧ ВИТАЛИЙ СЕРГЕЕВИЧ	

Цель работы: моделирование логических схем на элементах с памятью.

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

Описание

Составить и описать принцип работы двух схем: счётчика и регистра сдвига с линейной обратной связью.

В работе можно использовать только следующие элементы: NOT, OR, AND, NOR, NAND, XOR, XNOR. Соответственно все триггеры, мультиплексоры и пр. собираются на логических элементах в виде подсхем.

Количество входов всех базовых элементов не ограничено (может быть больше 2х).

В качестве входного элемента используем элемент Контакт, в качестве выходного можно использовать Контакт или Шестнадцатеричный индикатор.

Направление всех логических элементов: Восток. Исключение - элементы выхода, для них направление Запад. В случае использования Шестнадцатеричного индикатора ориентация разветвителя (splitter) может быть любая.

Вариант

1. Асинхронный вычитающий счётчик(Модуль счёта = 22)
2. Тип конфигурации – Галуа(Конфигурация = (7, 6, 5, 4))

Теория

Счётчик – устройство для счёта числа импульсов поданных на вход.
Модуль счёта – число различных состояний, который может принимать счётчик.

Асинхронный вычитающий счётчик – вид вычитающего счётчика, в котором при возникновении переноса разряда(вычитании), происходит последовательный перенос от разряда к разряду.

Практика(Счётчик)

1. RS триггер (“RS” – схема в проекте)(см. Рисунок 1)

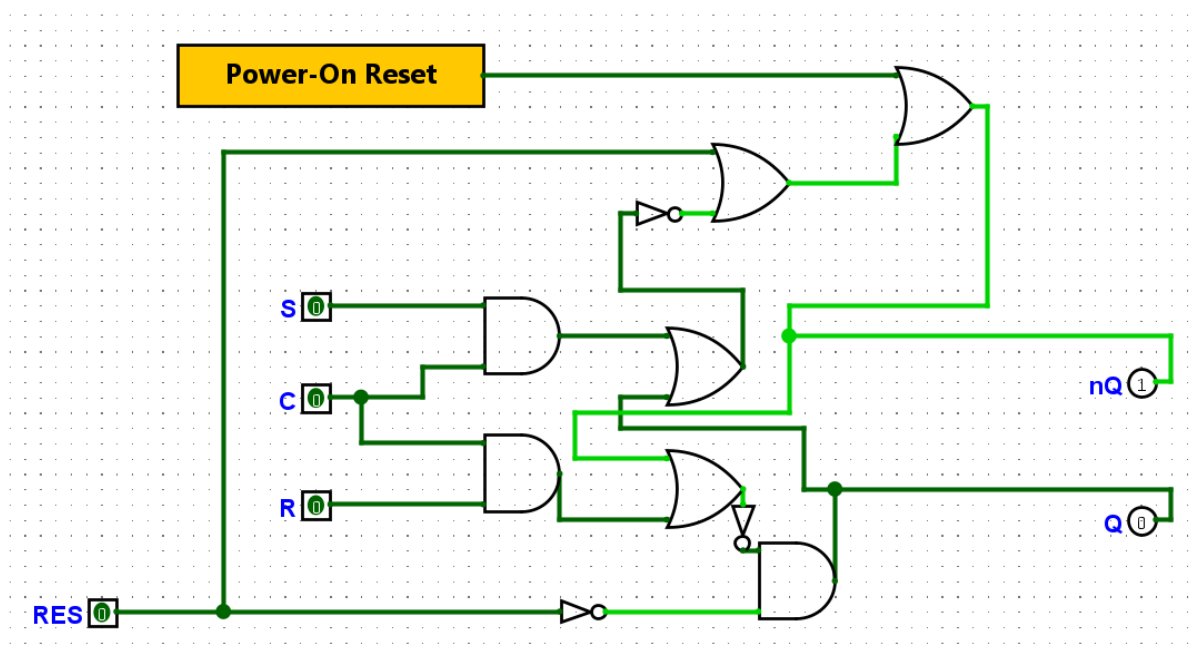


Рисунок 1 – RS триггер

Триггер использует элементы: NOT, AND, OR.

Выходные контакты:

$$1)Q = (0/1)$$
$$2) nQ = (0/1) = \text{NOT}(Q).$$

Входные контакты:

1) RES – сброс сигнала выхода ($Q = 0$, $nQ = 1$), работает независимо от других сигналов.

2) S – подача сигнала на выход ($Q = 1$, $nQ = 0$), работает при $C = 1$

3) R – сброс сигнала на выход ($Q = 0$, $nQ = 1$), работает при $C = 1$

2) C – “обновление” сигнала выходов при использовании S и R.

2. Т триггер (“Т” – схема в проекте) (см. Рисунок 2)

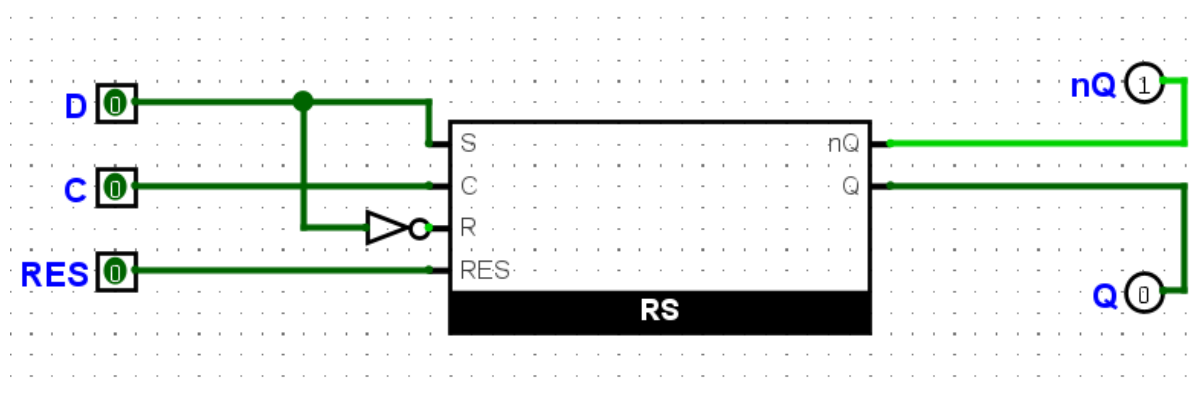


Рисунок 2 – Т триггер

Триггер использует элементы: NOT, RS триггер.

Выходные контакты:

1) Q – выход Q из RS триггера

2) nQ – выход nQ из RS триггера

Входные контакты:

1) RES – сброс сигнала выхода в RS триггере.

2) D – подача ($D=0:Q=0$) ($D=1:Q=1$)/сброс сигнала на выход, работает при $C = 1$.

3) C – “обновление” сигнала выходов при использовании D.

3. D триггер (“D” – схема в проекте) (см. Рисунок 3)

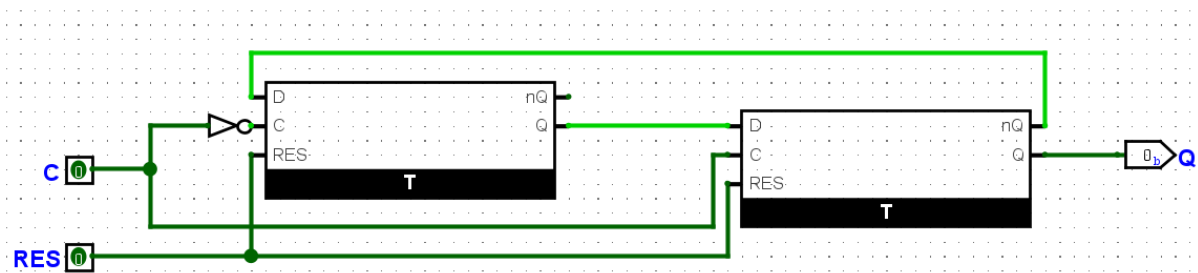


Рисунок 3 – D триггер

Триггер использует элементы: NOT, T триггер.

Выходные контакты:

1)Q – значение выходного сигнала(0/1)

Входные контакты:

1)RES – сброс сигнала выхода($Q = 0$)

2)C – подача “смены” сигнала на выход, при следующем переключении.

Таблица истинности для C и Q

C	0	1	0	1	0	1	0	1	0
Q	0	0	0	1	1	0	0	1	1

“Смена” сигнала происходит не на следующем такте, а через один.

3.Асинхронный вычитающий счётчик по модулю 22(“Main” – схема в проекте)(см. Рисунок 4)

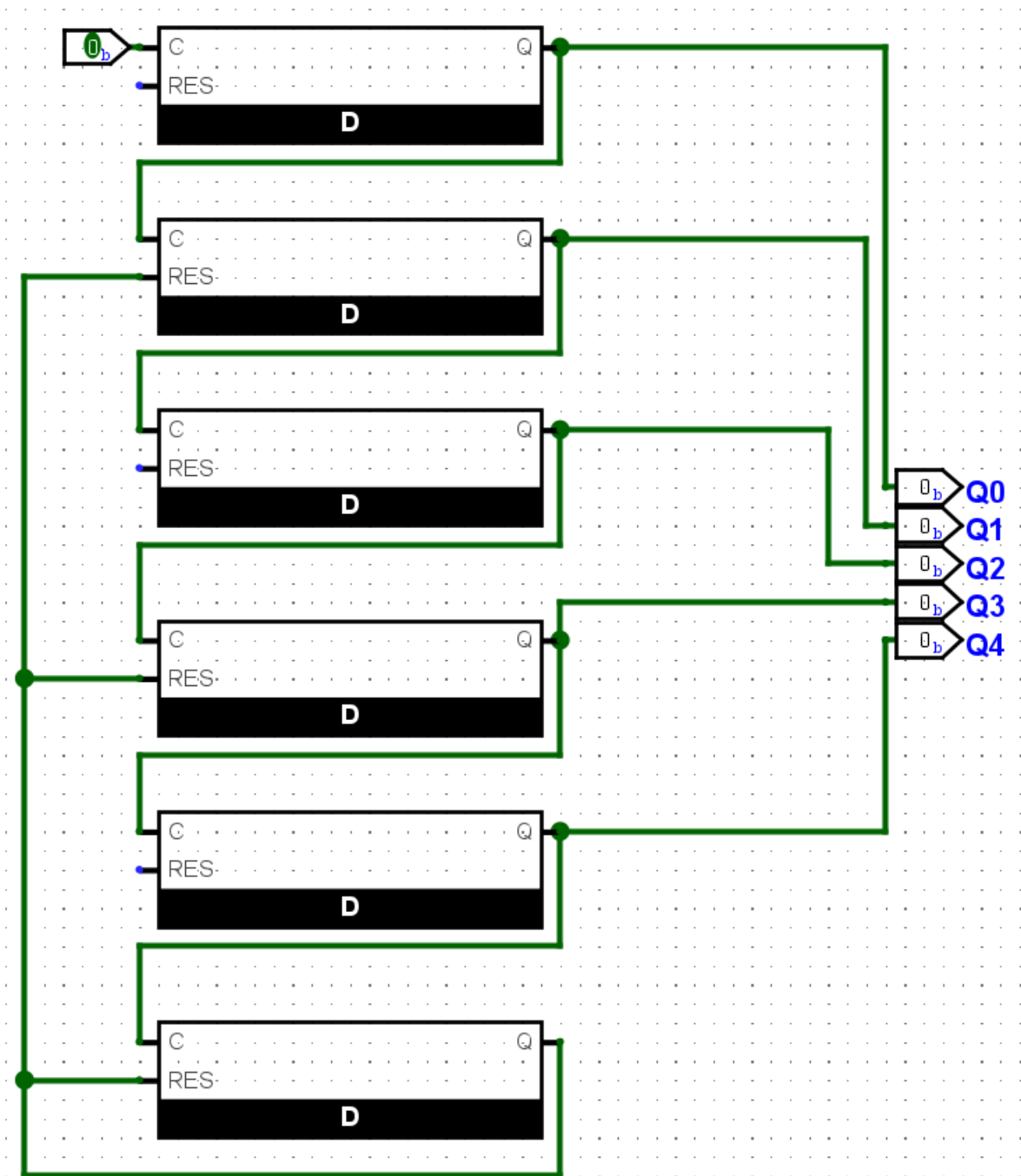


Рисунок 4 - Асинхронный вычитающий счётчик по модулю

22

Счётчик использует: D триггер.

Выходные контакты:

1) Q_x – значение выходного сигнала для x -го бита

Входные контакты:

1) C – при переключении из $C = 0$ в $C = 1$, уменьшает значение счётчика на 1.

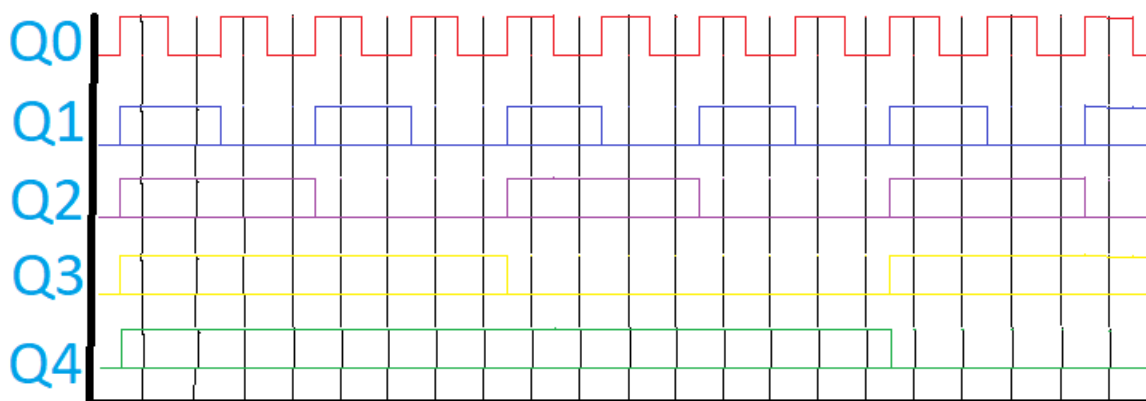
Для каждого бита используется свой собственный D – триггер.

Нумерация триггеров – Сверху вниз начиная с 0. При каждом переключении состояния С из 0 в 1, подается сигнал на смену значения в нулевом D триггере, если до “смены” нулевой D триггер имеет сигнал выхода $Q0 = 0$, то он переходит в состояние $Q0 = 1$ и подаёт сигнал на смену значения следующему D триггеру, уже для первого бита($Q1$) и так далее для всех последующих D триггеров.

Самый последний D триггер служит для того, чтобы обновить счётчик по модулю 22, когда уменьшается счётчик при нулевом состоянии всех выходов. Выход последнего D триггера подключен к входам “RES”, для триггеров битов: 1 и 3 и к самому себе.

При попытке уменьшить счётчик в состоянии 0 для всех выходов, каждый подается сигнал на последний триггер для переключения состояния(всегда из 0 в 1).Который, в свою очередь, подает сигнал на сброс значения битов 1 и 3.

Таким образом, при уменьшении счётчика на значениях: 00000(число – 0), мы переходим на значениях: 10101(число – 21).



Временная диаграмма асинхронного вычитающего счётчика по модулю 22

Практика(Регистр)

1.Setter(“Setter” – схема в проекте)(см. Рисунок 5)

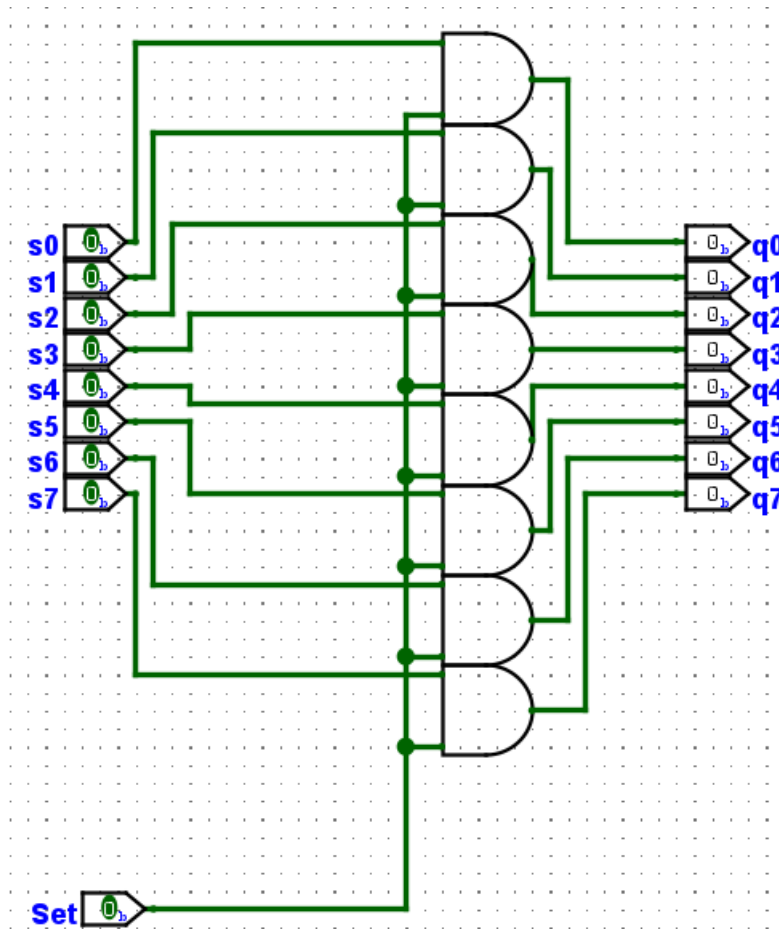


Рисунок 5 – Setter

Использует: AND.

Выходные контакты:

1) Q_x – значение выходного сигнала для x -го бита

Входные контакты:

1) s_x – подать сигнал на x -ый бит(Работает при $Set = 1$)

2) Set – подает сигнал на “обновление” сигнала для каждого бита

Используется для установки начального состояния битов для регистра.

2.D триггер(“Dtrig” – схема в проекте)(см. Рисунок 6)

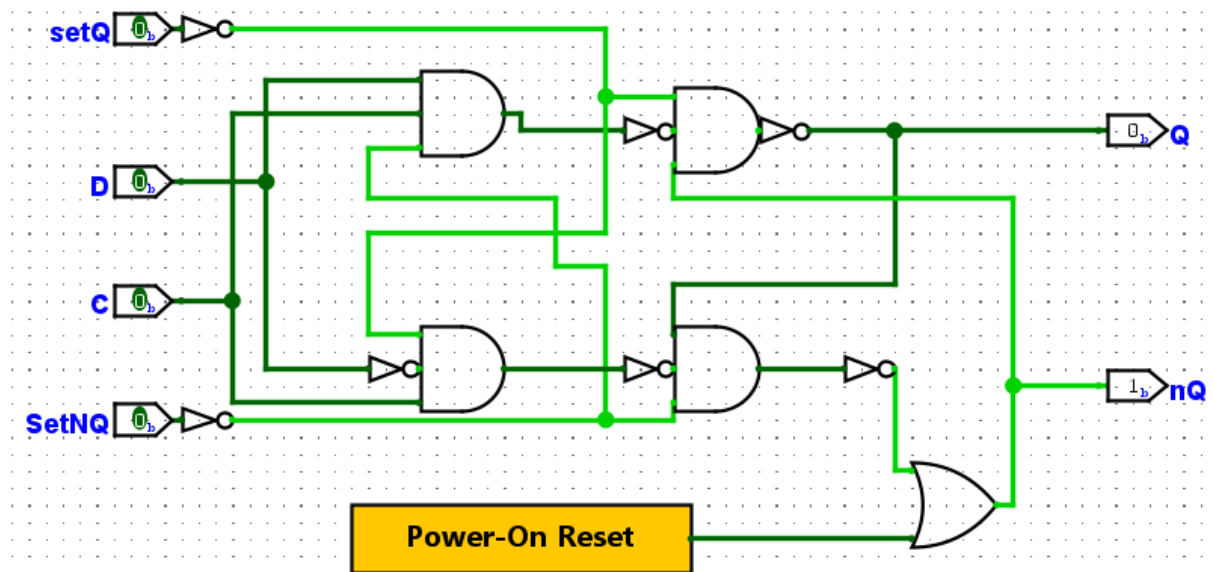


Рисунок 6 – D триггер

Использует: AND, OR.

Выходные контакты:

- 1) Q – значение выходного сигнала(0/1)
- 2) nQ – обратное значение выходного сигнала(0/1)($nQ = \text{not}(Q)$)

Входные контакты:

- 1) setQ – установить значение 1 выходного сигнала($Q = 1$)
- 2) setNQ – установить значение 0 выходного сигнала($Q = 0$)
- 3) D - установить значение 1(при $D = 1$) выходного сигнала(работает при $C = 1$)
- 4) C - подать сигнал на “обновление” значения выходного сигнала.

С помощью setQ и setNQ независимо от других входных данных можем установить соответствующее значение на выход.

Иначе используем D для выбора устанавливаемого сигнала на выход($D = 0: Q = 0$, $D = 1: Q = 1$) И обновляем значение выхода при $C = 1$. При $C = 0$ выходное значение остаётся прежним.

3.Динамический D триггер (“DDtrig” – схема в проекте)(см. Рисунок 7)

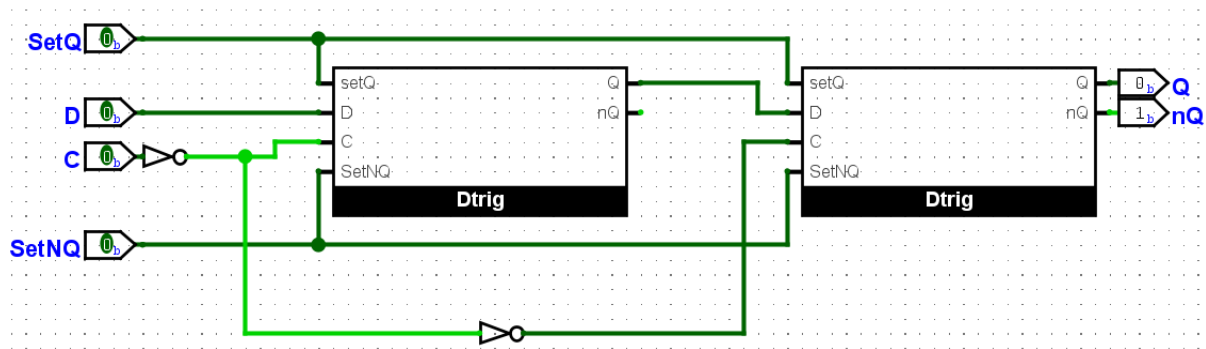


Рисунок 7 – Динамический D триггер

Аналогичная работа как и у обычного D триггера.

Однако теперь, сигнал D обновляет значение Q не “в реальном времени”, даже если $C = 1$. Значение Q устанавливается в зависимости от значения D на прошлой “Смене”(переключении значения C)

Такой функционал работает за счёт того, что при любом значении C(0/1) один из триггеров всегда будет не в активном состоянии.

Таблица истинности для C,D,Q

D	0	1	1	1	1	0	0	0	0
C	0	0	1	0	1	1	0	1	0
Q	0	0	0	0	1	1	1	0	0
T(time)	0	1	2	3	4	5	6	7	8

4. Регистр сдвига с линейной обратной связью (“Main” – схема в проекте)(см. Рисунок 8)

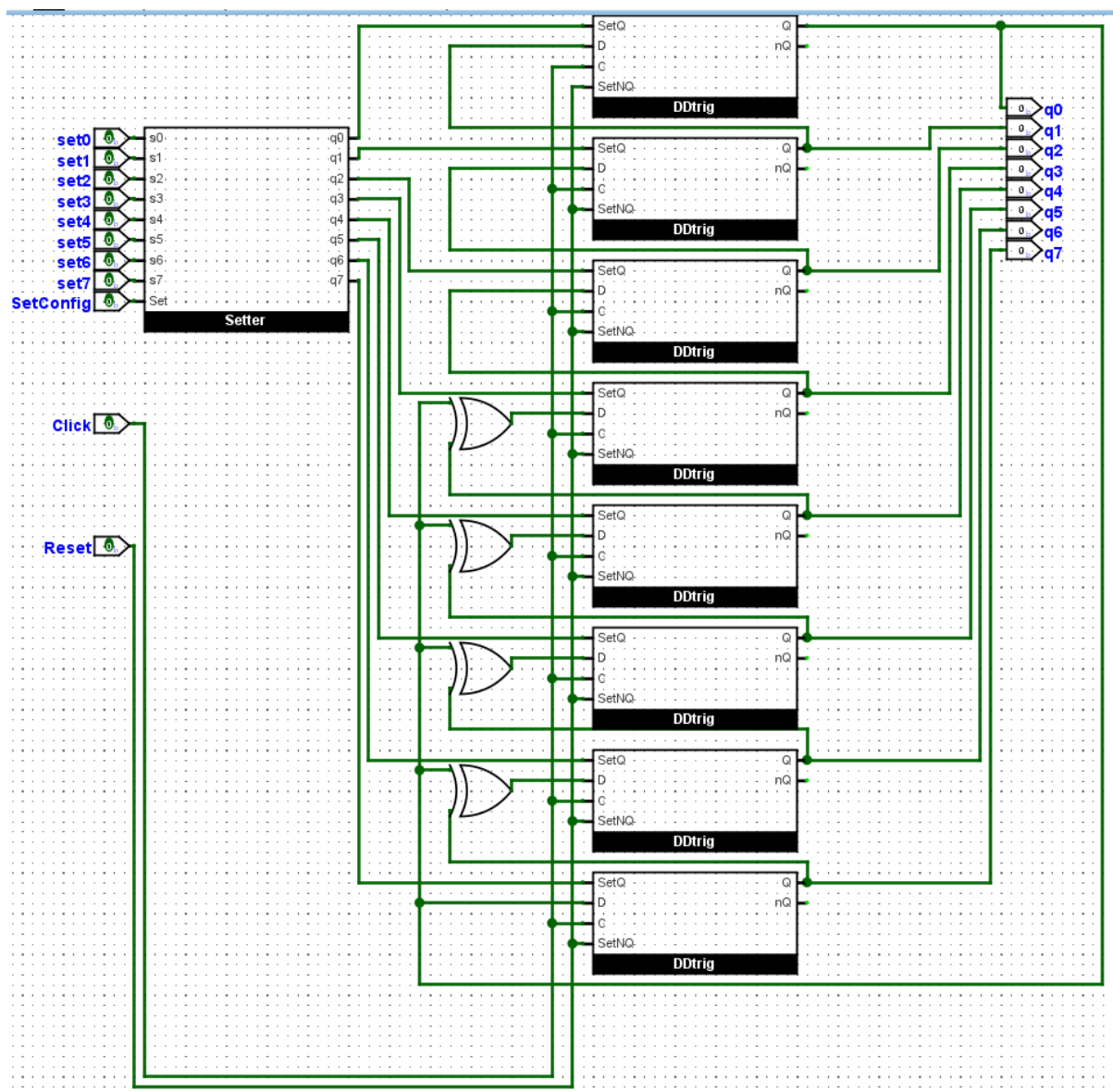


Рисунок 8 – Регистр сдвига с линейной обратной связью

Тип конфигурации – Галуа

Конфигурация:(7, 6, 5, 4)

Использует: XOR, динамический D триггер, setter.

Выходные контакты:

1)Q_x – значение выходного сигнал(0/1) для x-го бита

Входные контакты:

1)Click – при переключении из 0 в 1, происходит один такт для регистра.

2)Reset – сброс всех регистра.

3)А также входные значения для setter. С помощью которого задаётся стартовые значения регистра(Qx).

Для каждого такта(переключения Click из 0 в 1) регистр происходит сдвиг битов от старших к младшим.

Это достигается тем, что выход каждого триггера подключен к входу D более младшего триггера. Таким образом, если на текущем триггере стоит выходное значение 1, то это значит, что на предыдущем такте, более старший триггер(который подключен к нашему D) имел значение 1.(Аналогично для 0)

Стоит отметить, что нулевой(самый младший бит) циклически сдвигает значение своего бита на самый старший бит.

При этом при сдвиге нулевого бита, хог-ится(со значением нулевого бита) значения битов: 6,5,4,3.

Это достигается тем, что к соответствующим триггерам подключено(к входу D) значение самого младшего бита и значение более старшего триггера, пропущенные через элемент XOR.