

Последовательностные устройства

Последовательностные устройства – цифровые устройства с памятью, сигнал на выходе которых зависит от комбинации входных сигналов и от сигналов, извлеченных из памяти в данный момент.

Базовым элементом всех последовательностных устройств является **триггер**.

Триггер – ячейка памяти, в которую можно записать 1 бит информации (0 или 1).

Триггеры

Триггер – последовательностное устройство, которое может находиться в одном из двух устойчивых состояний. Переход из одного устойчивого состояния в другое происходит под действием входных сигналов. Такой переход также называют переключением или опрокидыванием.

Классификация триггеров

1. По функциональному признаку, определяющему поведение триггера при воздействии сигналов управления, различают:

- а) *RS*-триггеры;
- б) *D* -триггеры;
- в) *T*-триггеры;
- г) *JK*-триггеры.

2. По способу управления различают:

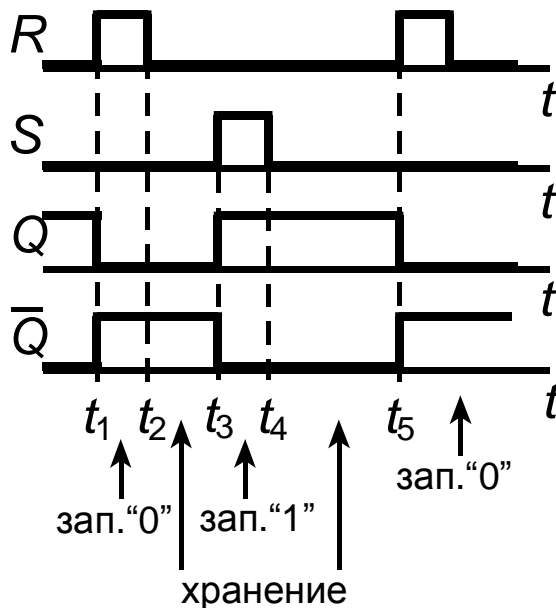
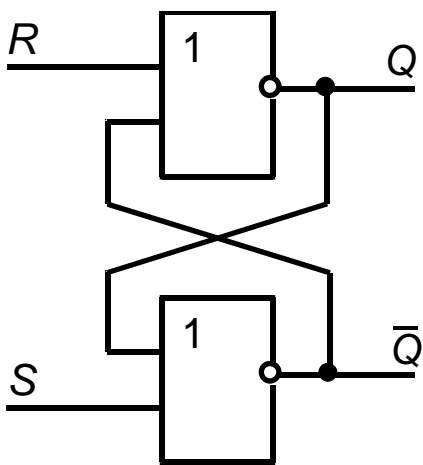
- а) **асинхронные** - переключение из одного состояния в другое происходит с поступлением входного сигнала на информационный вход;
- б) **синхронные (тактируемые)** – *дополнительно к информационному триггер имеет тактовый вход (для подачи тактовых (синхронизирующих) импульсов); переключение триггера осуществляется только при подаче тактирующего (синхронимпульса) импульса.*

Асинхронный RS-триггер (схема на элементах ИЛИ-НЕ)

RS – триггер имеет: 2 информационных входа (R, S); 2 выхода (Q, \bar{Q}).

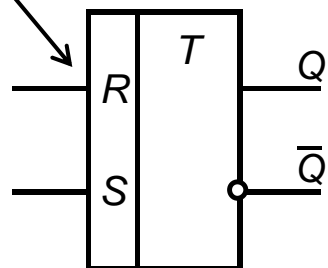
S от англ. *Set* – установка в «1» ($Q = 1$).

R от англ. *Reset* – сброс – установка в «0» ($Q = 0$).



R	S	Q^n	Q^{n+1}	
0	0	0	0	режим хранения (сост. не меняется)
0	0	1	1	
0	1	0	1	запись «1»
0	1	1	1	
1	0	1	0	запись «0»
1	0	0	0	
1	1	X		запрещенное состояние!

Активный
уровень «1»



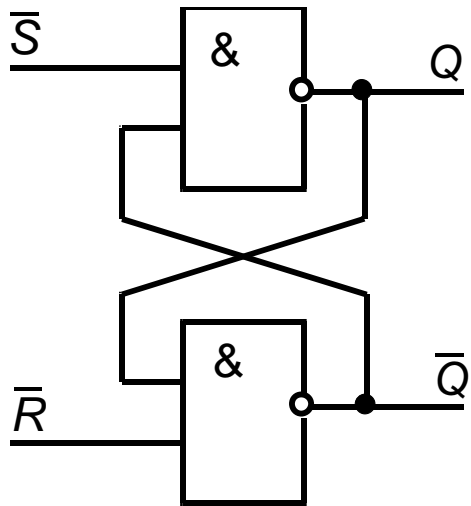
УГО триггера

Q^n - предыдущее состояние (состояние триггера до подачи сигналов R, S).

Q^{n+1} - последующее состояние (состояние триггера после подачи сигналов R, S).

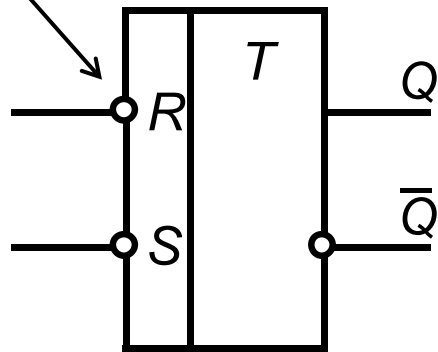
X – безразличное состояние.

Асинхронный RS-триггер (схема на элементах И-НЕ)

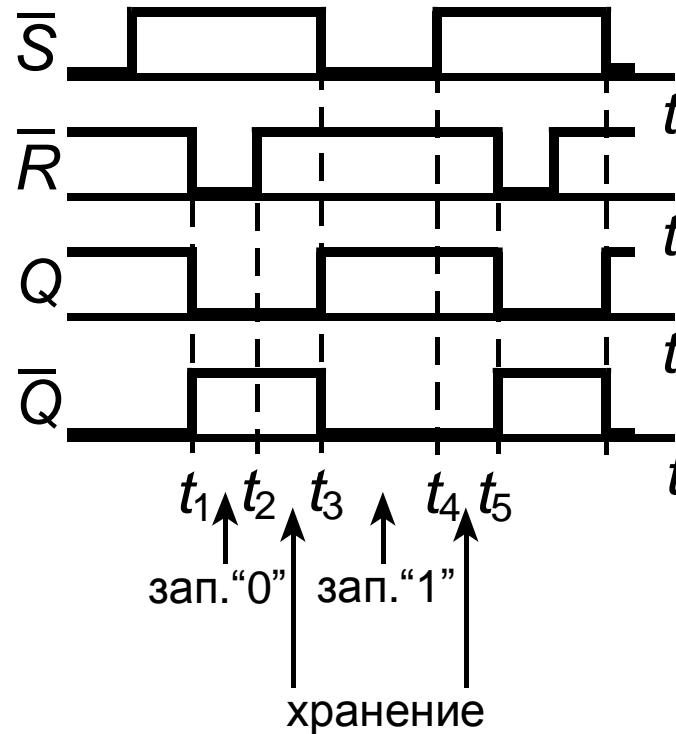


\bar{R}	\bar{S}	Q^n	Q^{n+1}	
0	0	X		запрещенное состояние!
0	1	X	0	запись «0»
1	0	X	1	запись «1»
1	1	X	Q^n	хранение

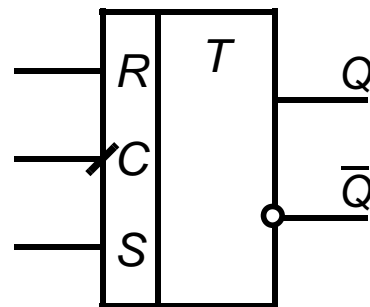
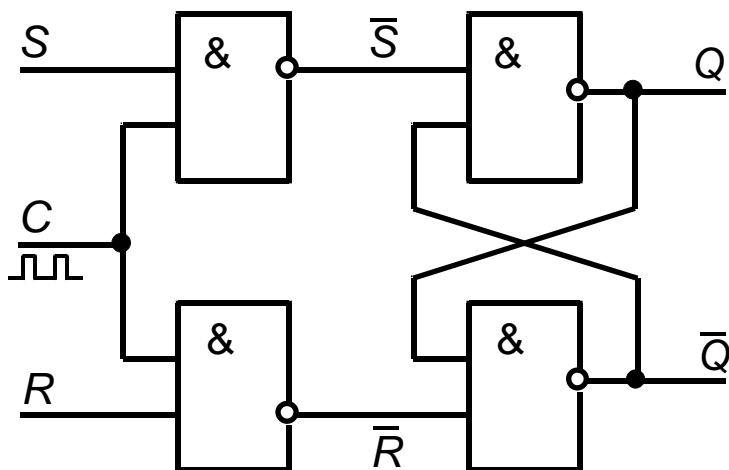
Активный уровень «0»



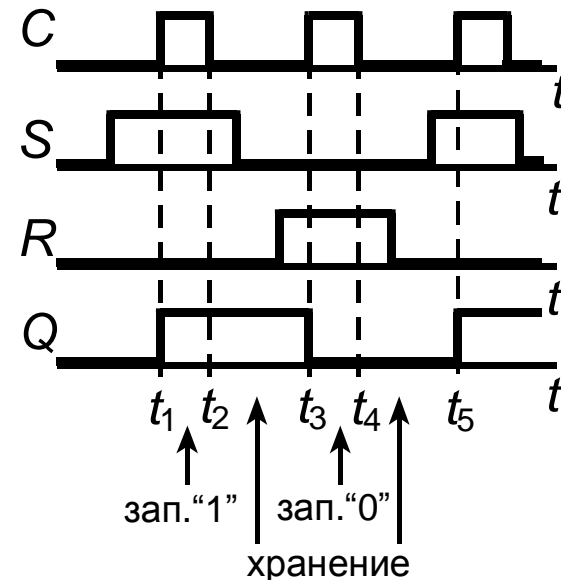
УГО триггера



Синхронный RS-триггер (RST-триггер)



УГО триггера



S	R	C	Q^n	Q^{n+1}	
X	X	0	X	Q^n	хранение
1	0	1	X	1	запись «1»
0	1	1	X	0	запись «0»
1	1	1	X		неопределенность (запрещенный режим)

Переключение триггера производится только при наличии тактового сигнала.

Вход C или CLK - тактовый вход, сокр. от англ. «**CL**ock» - тактировать.

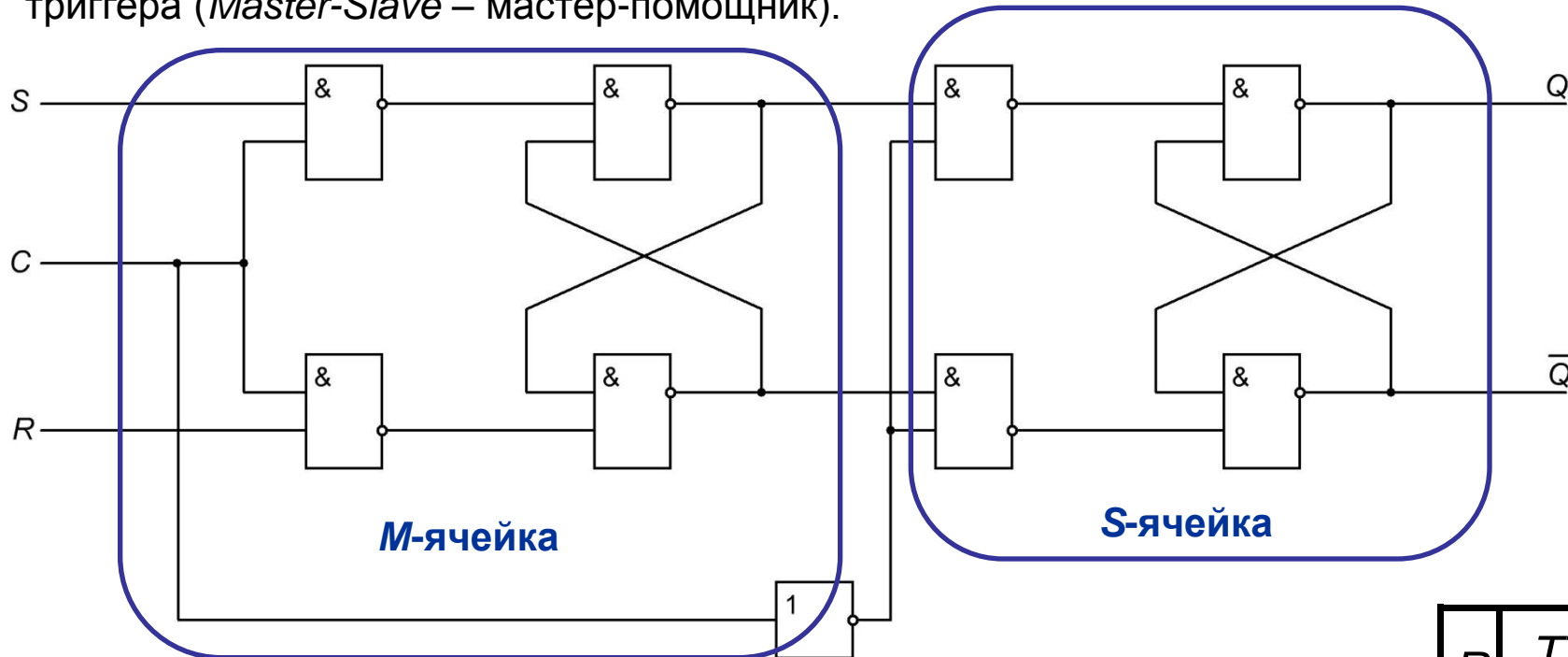
При $C = 1$ триггер становится прозрачным по входам S и R, т.е. любые изменения сигналов на входах S и R отражаются на входах \bar{S} и \bar{R} , \Rightarrow триггер переключается в соответствии с таблицей истинности. Триггер становится **прозрачным**.

Для нормальной работы триггера во время действия синхроимпульса (сигнала по входу C) сигналы R и S **не должны изменяться!**

При $C = 0$ первые два элемента работают как инверторы, триггер находится в режиме хранения информации $Q^{n+1} = Q^n$.

Двухтактный RS-триггер (MS-схема)

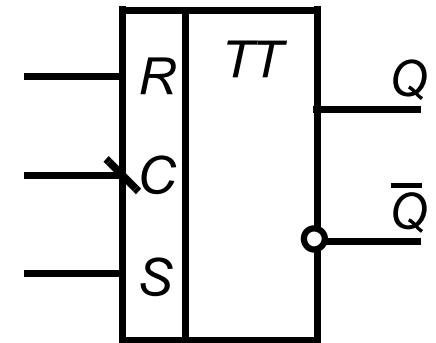
Для устранения прозрачности схемы триггера при $C = 1$ используется MS-схема триггера (Master-Slave – мастер-помощник).



при $C = 1$ ячейка M прозрачна по вх. S, R ;

ячейка S заблокирована, в ней сохраняется предыдущая ин-фа (предыдущее состояние).

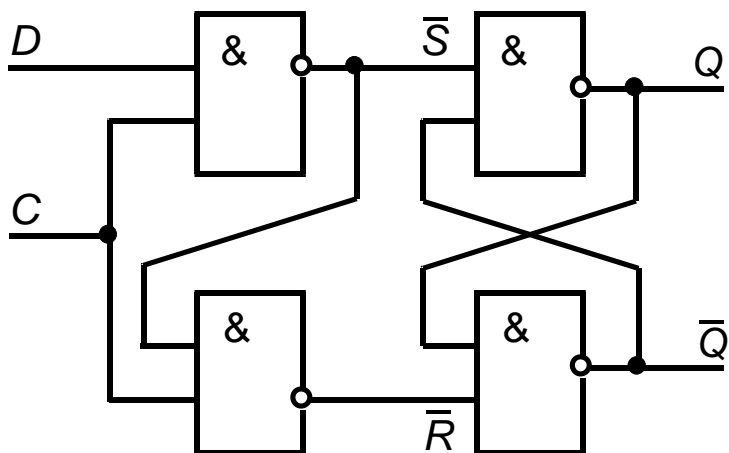
при $C = 0$ информация из M -ячейки переписывается в S -ячейку.



Т.о. работа триггера осуществляется в 2 такта: в первом такте – установка 1 триггера (M -ячейки); во втором такте информация из M -ячейки переписывается в S -ячейку.

D-триггер

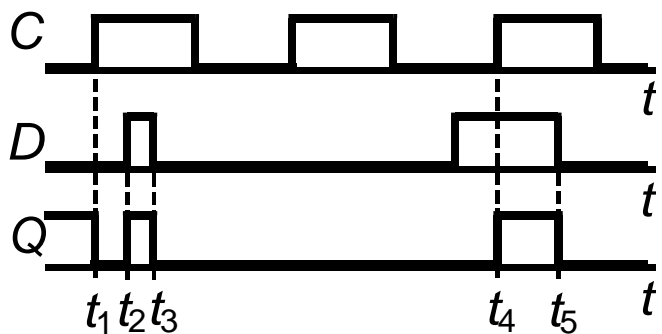
D-триггер имеет специальный информационный вход. Другие названия: триггер задержки, фиксатор или прозрачная защелка.



$$Q^{n+1} = D$$

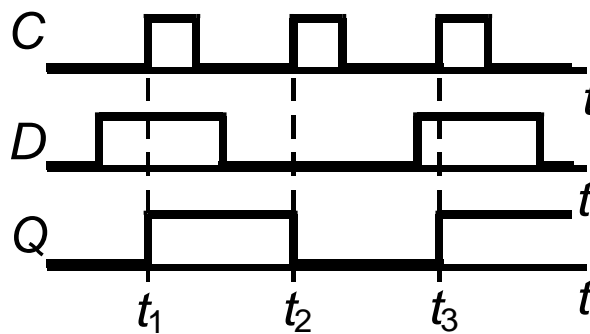
C	D	Q^n	Q^{n+1}	
0	X	Q^n	Q^n	хранение
1	1	X	1	запись «1»
1	0	X	0	запись «0»

Прозрачная защелка

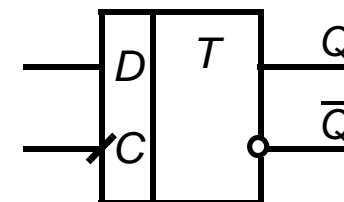


при $C = 1$ изменение сигнала на входе D вызывает изменение сигнала на выходе Q .

Триггер задержки



Триггер задерживает (хранит) информацию до следующего синхроимпульса.

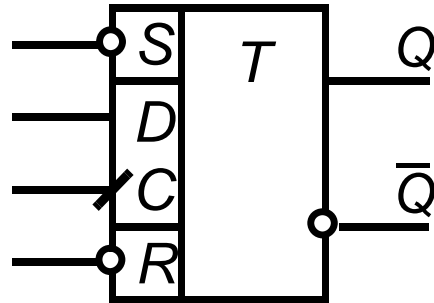


УГО D-триггера

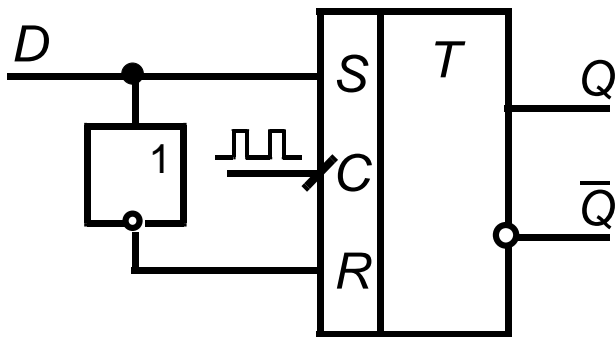
D-триггер

асинхронные входы

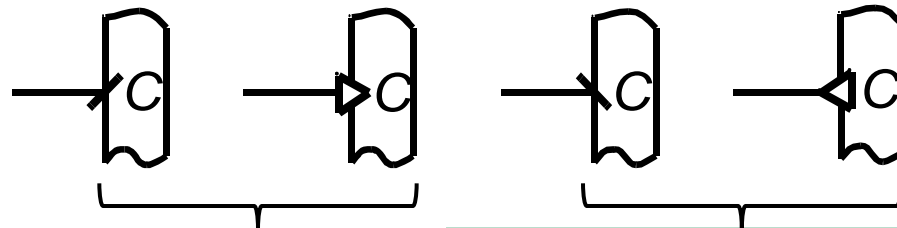
D-триггер с входами
предустановки R и S.



D-триггер можно построить на
основе синхронного RS-триггера.



\bar{S}	\bar{R}	C	D	Q		
0	1	X	X	1	асинхр. уст-ка «1»	обычный асинхр. RS-триггер
1	0	X	X	0	асинхр. уст-ка «0»	
0	0	X	X		запрещенное состояние	
1	1		1	1	синхр. уст-ка «1»	обычный D-триггер
1	1		0	0	синхр. уст-ка «0»	

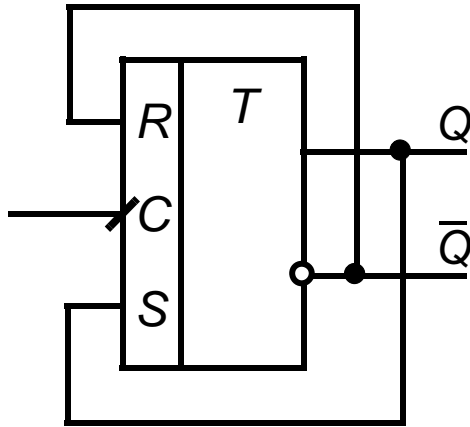


Переключение происходит при переходе
сигнала на тактовом входе из 0 в 1 () , т.е.
по переднему фронту синхроимпульса.

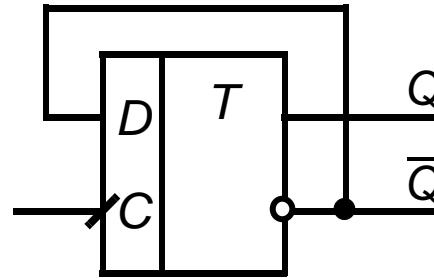
Срабатывание происходит при
переходе тактового сигнала из 1 в
0 () , т.е. по заднему фронту
(срезу) синхроимпульса.

T-триггер (счетный триггер)

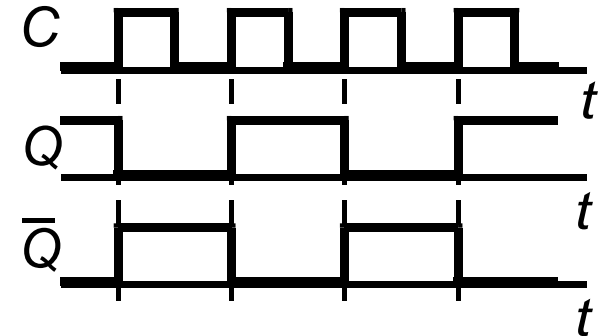
Счетный T-триггер специально не выпускается, но может быть построен на базе других триггеров. Триггер имеет тактовый вход C, => триггер - синхронный.



T-триггер на базе RST-триггера



T-триггер на базе D-триггера



Диаграммы работы T-триггера

Данный триггер осуществляет деление частоты на 2.

$$f_{\text{ВЫХ}} = \frac{f_{\text{ВХ}}}{2}$$

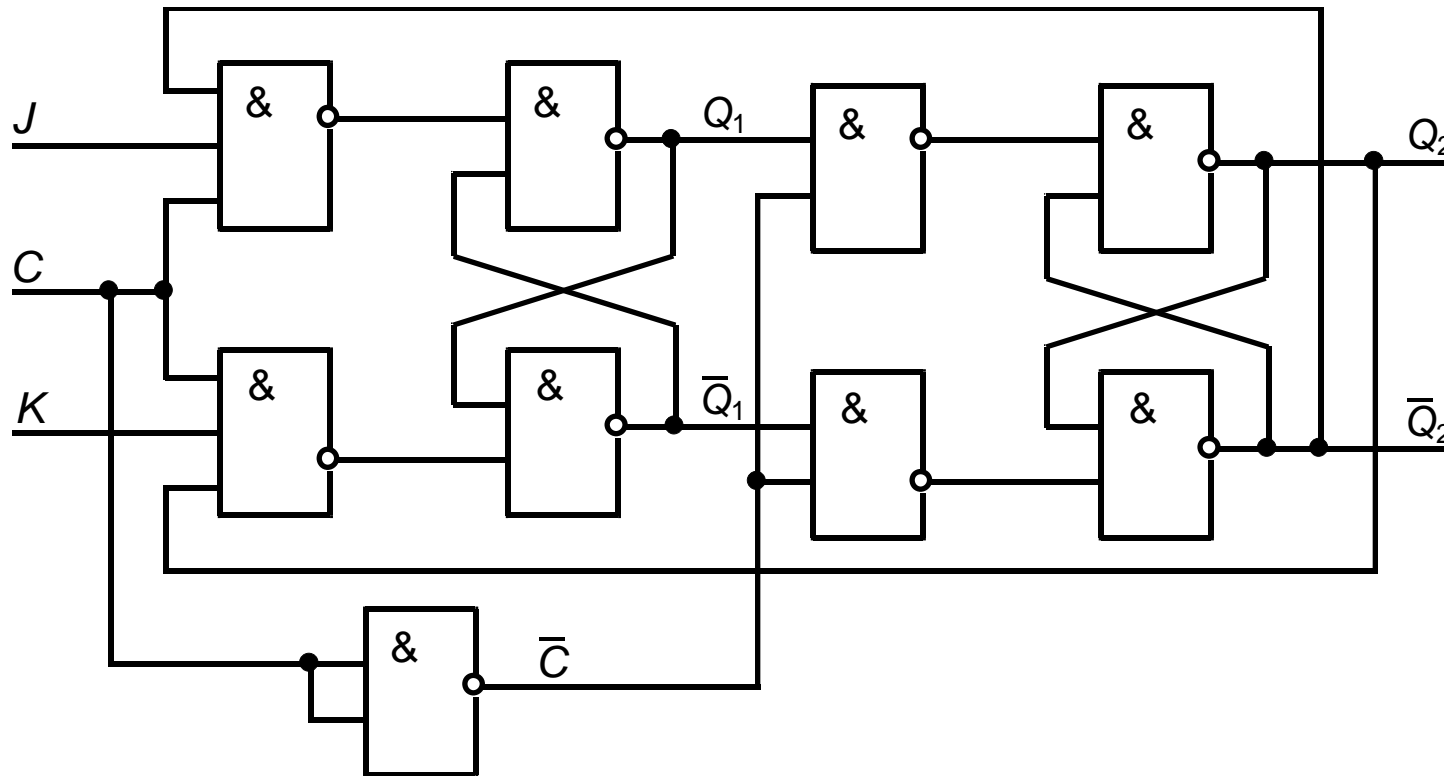
Т.е. на каждые два тактовых импульса на выходе выдает один импульс.

$f_{\text{ВХ}}$ – частота входного сигнала; $f_{\text{ВЫХ}}$ – частота выходного сигнала.

JK-триггер

JK-триггер – наиболее широко используемый универсальный триггер. На его основе могут быть построены другие типы триггеров. JK-триггер получается из T-триггера введением дополнительных входов *J* и *K*.

Триггер имеет: информационные входы *JK* (от англ. *jump* и *keep*), тактовый вход *C*.



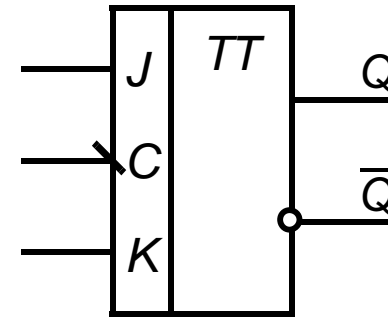
В данном случае JK-триггер выполнен по MS-схеме. Дополнительно могут присутствовать входы асинхронной установки *R* и *S*.

JK-триггер

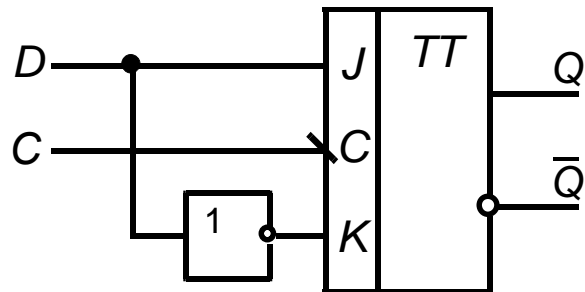
Работу JK-триггера можно описать функцией:

$$Q^{n+1} = \overline{Q}^n \cdot J + \overline{K} \cdot Q^n$$

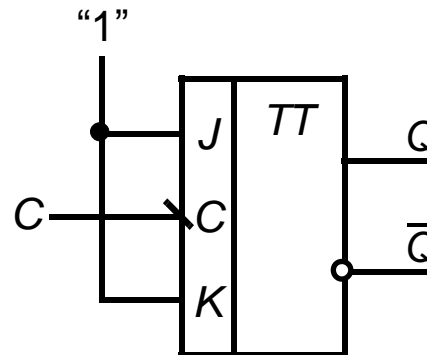
C	J	K	Q^n	Q^{n+1}	
0	0	0	X	Q^n	хранение
0	0	1	X	Q^n	
0	1	0	X	Q^n	
0	1	1	X	Q^n	
	0	0	X	Q^n	хранение
	0	1	1	0	запись «0»
	1	0	0	1	запись «1»
	1	1	Q^n	\overline{Q}^n	переключение



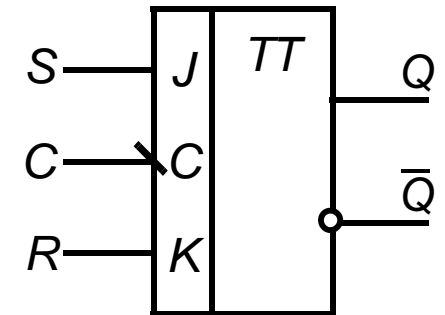
УГО JK-триггера



D-триггер на базе JK-триггера



T-триггер на базе JK-триггера



RS-триггер на базе JK-триггера

Ссылки

1. <http://www.khaer.com.ua/downloads/ke/rus/tema2/2.1.htm> (анимация работы триггера)

Счетчики импульсов

Счетчик импульсов – функциональный узел для подсчёта импульсов, поданных на его вход. счётчики строятся на триггерах.

Информационная емкость - коэффициент счёта ($K_{сч}$) – это число в двоичном коде, до которого можно производить счёт (максимальное значение.)

$$K_{сч} = 2^n, \text{ где } n \text{ – число триггеров.}$$

Если $K_{сч} \neq 2^n$, то счётчик имеет произвольный коэффициент счёта.

Классификация

По принципу действия:

- 1. асинхронные** (последовательные) – каждый последующий триггер срабатывает от предыдущего, а сигналы счёта подаются на первый триггер.
- 2. синхронные** – тактовый сигнал подается сразу на все разряды, но срабатывание последующего триггера происходит после того, как единицами заполнены все младшие разряды.

По способу обработки информации:

- 1. прямого счёта** – каждый последующий импульс прибавляет «1» к имеющейся информации в счётчике (счёт на увеличение, счёт на сложение).
- 2. обратного счёта** – с каждым тактовым импульсом от информации в счётчике вычитается «1» (счёт на вычитание).
- 3. реверсивные** – считают как в прямом (на сложение), так и в обратном (на вычитание) направлении.

Асинхронный счётчик

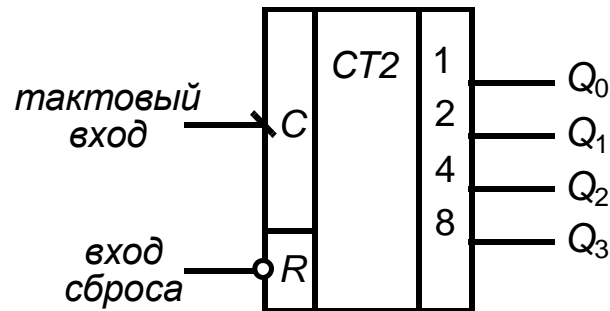
(или счётчик с последовательным или сквозным переносом)

Дес. код	Q_3	Q_2	Q_1	Q_0
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	1	1	0	1
14	1	1	1	0
15	1	1	1	1

Из таблицы:

1. значение Q_i изменяется только тогда, когда в младшем разряде происходит переход сигнала с «1» на «0» (счётчики последовательного типа);
2. значение Q_i изменяется, когда предыдущие младшие разряды заполнены единицами (счётчики параллельного типа).

Для организации счёта в обратном направлении необходимо все выходы Q соединить с тактовыми входами C . В этом случае с приходом тактового импульса информация будет убывать. Диаграммы в счётчике обратного счёта получаются инверсией диаграмм выходов Q (кроме C).

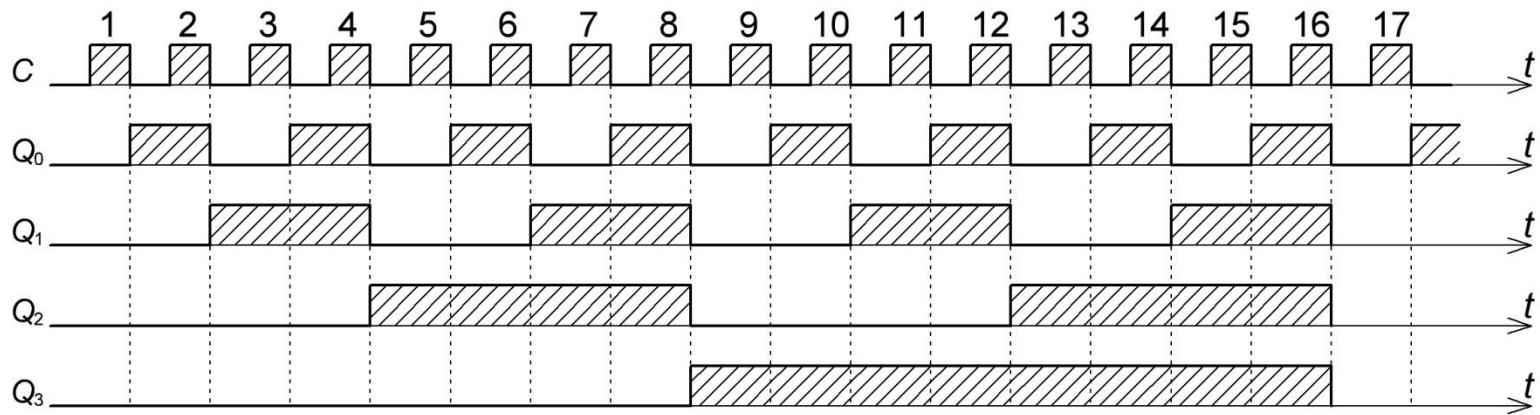
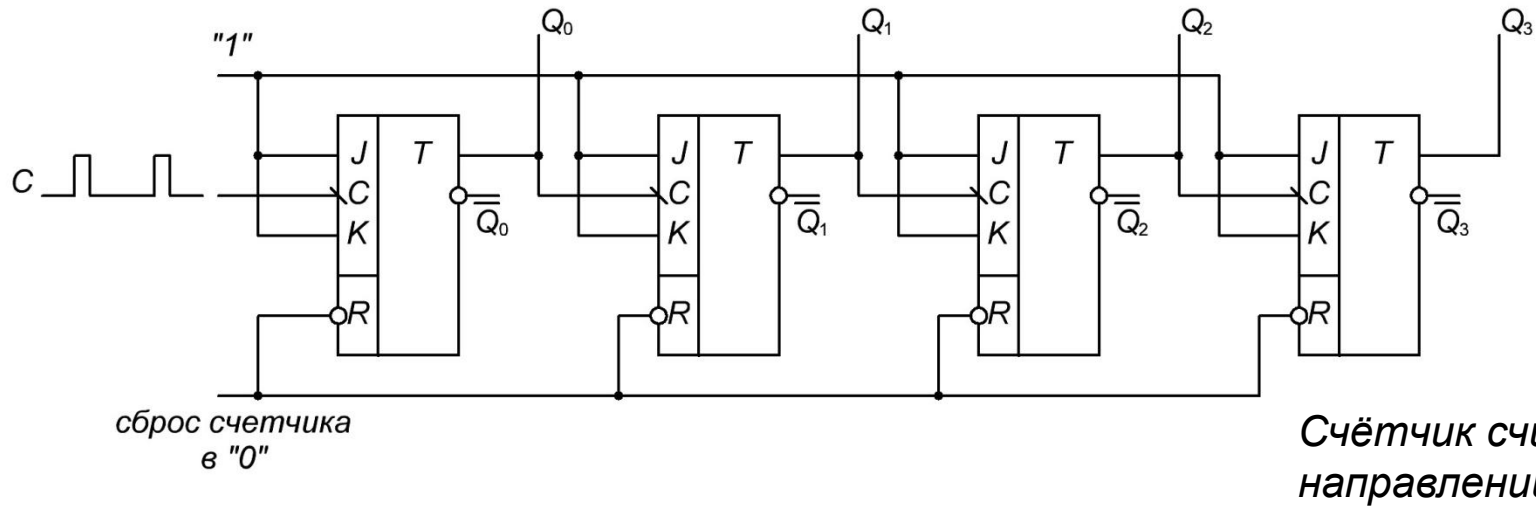


УГО счётчика импульсов

Обозначение:

СТ2 – двоичный счётчик;
СТ2/10, СТ10 – двоично-десятичный счётчик, считает в двоичном коде от 0 до 9, на десятом импульсе сбрасывается и считает заново.

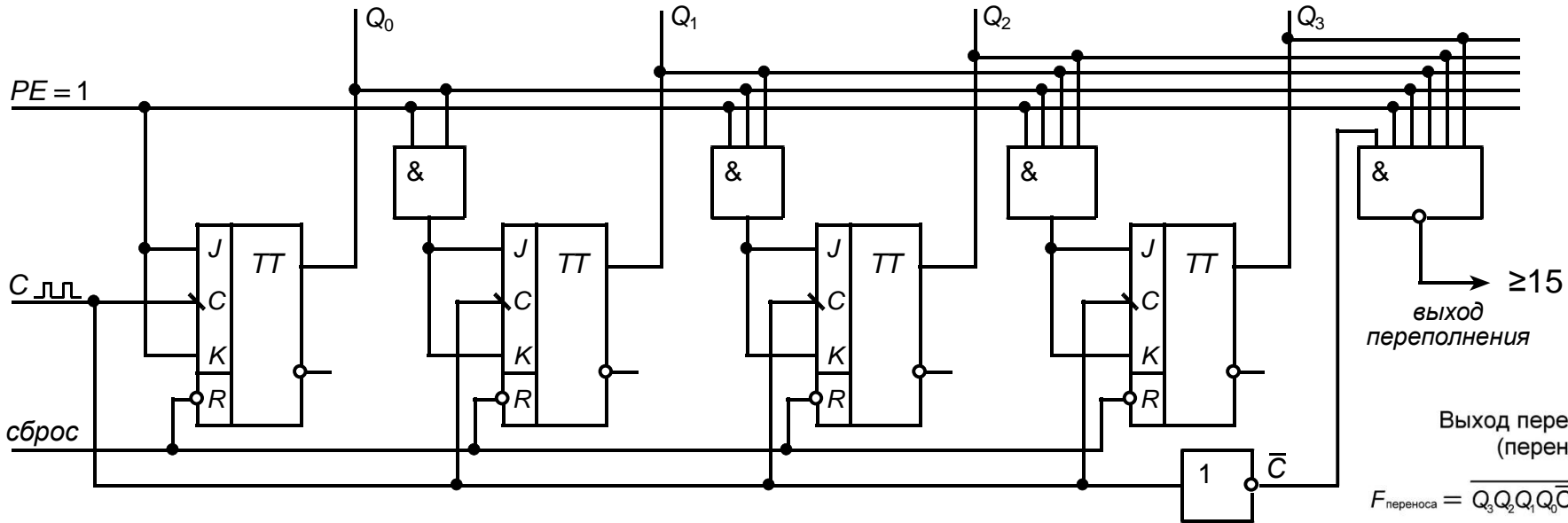
Асинхронный счётчик



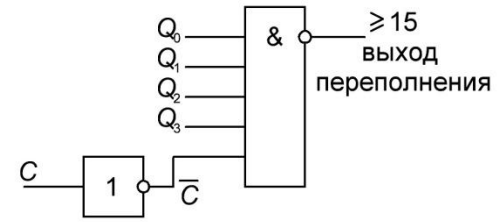
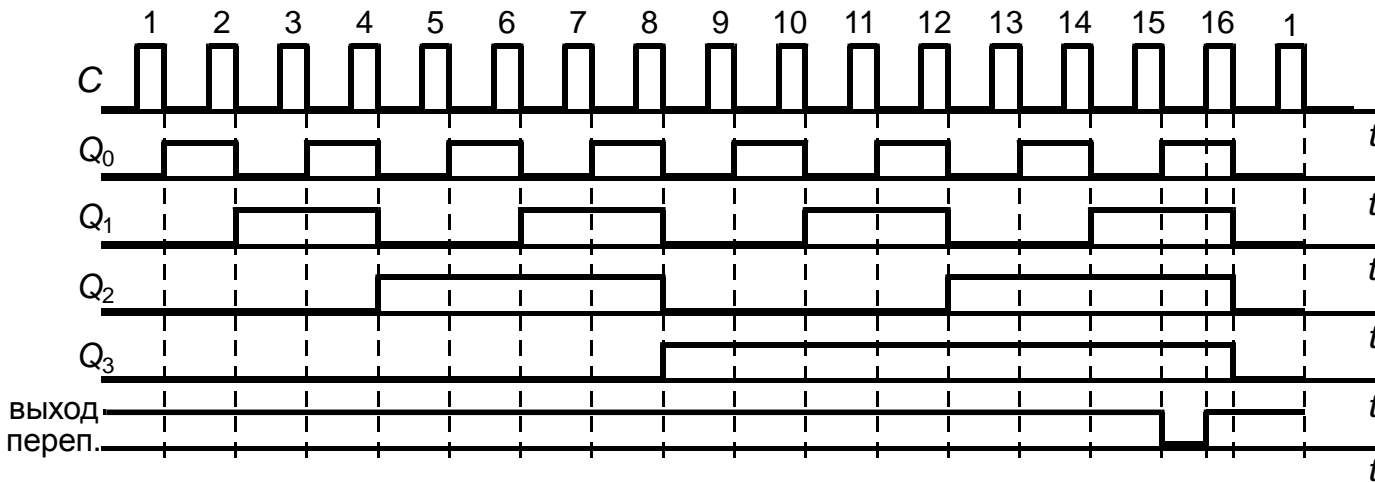
Счётчик считает в прямом направлении (счёт на сложение).

Счётчик является асинхронным, т.к. выходные сигналы появляются последовательно во времени (значение Q_i изменяется, когда в младшем разряде происходит переход сигнала с «1» на «0»).

Синхронный счётчик

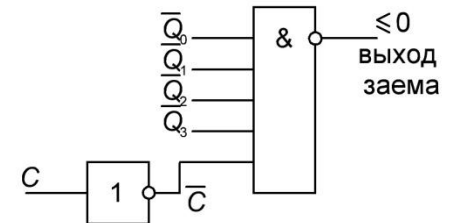


PE = 1 – разрешение счёта



Выход заема

$$F_{\text{заема}} = \overline{\overline{Q_3} \overline{Q_2} \overline{Q_1} \overline{Q_0}}$$

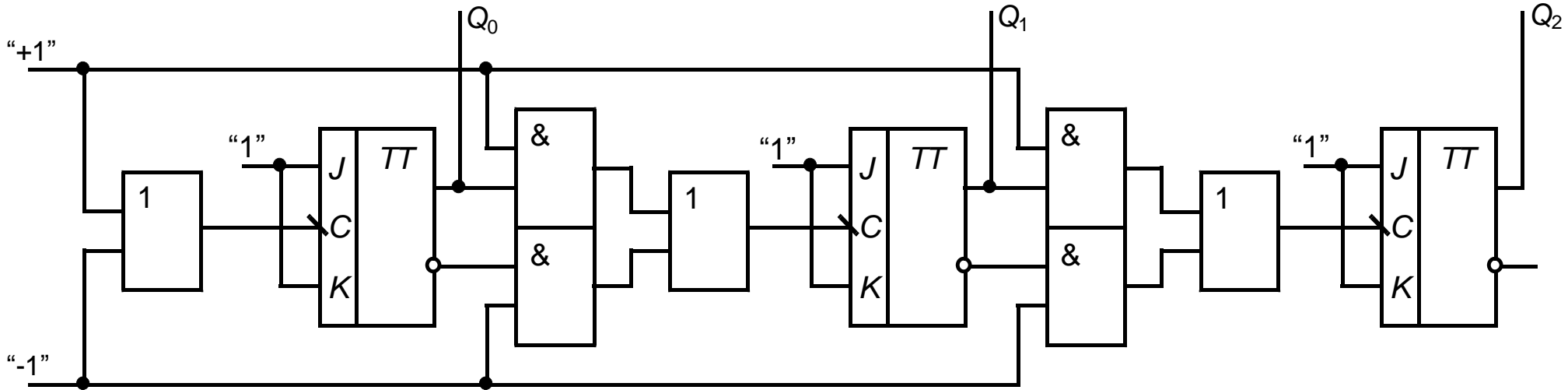


Синхронный счётчик, т.к. выходные сигналы появляются по каждому тактовому импульсу. Счётчик с параллельным переносом (или параллельного типа), т.е. значение Q_i меняется, когда младшие разряды заполнены единицами.

Реверсивный счётчик

Реверсивный счётчик считает как в прямом, так и в обратном направлении.

Как правило, имеет два тактовых входа: на увеличение (+1) и на уменьшение (-1).



В режиме вычитания входные импульсы подаются на вход «-1», при этом на вход «+1» подаётся лог. 0.

В режиме сложения входные импульсы подаются на вход «+1», а на вход «-1» следует подать лог. 0.

В данной схема – счётчик с последовательным переносом. Триггеры работают в как счётные триггеры.

Счётчики с предварительной установкой

Ряд счётчиков имеет информационные входы для предварительной записи (параллельной загрузки) информации. Сигналы, подаваемые на эти входы устанавливают триггеры счётчика в соответствующее состояние. В результате на выходах устанавливается двоичный код числа, с которого счётчик при следующем тактовом импульсе начнёт счёт.

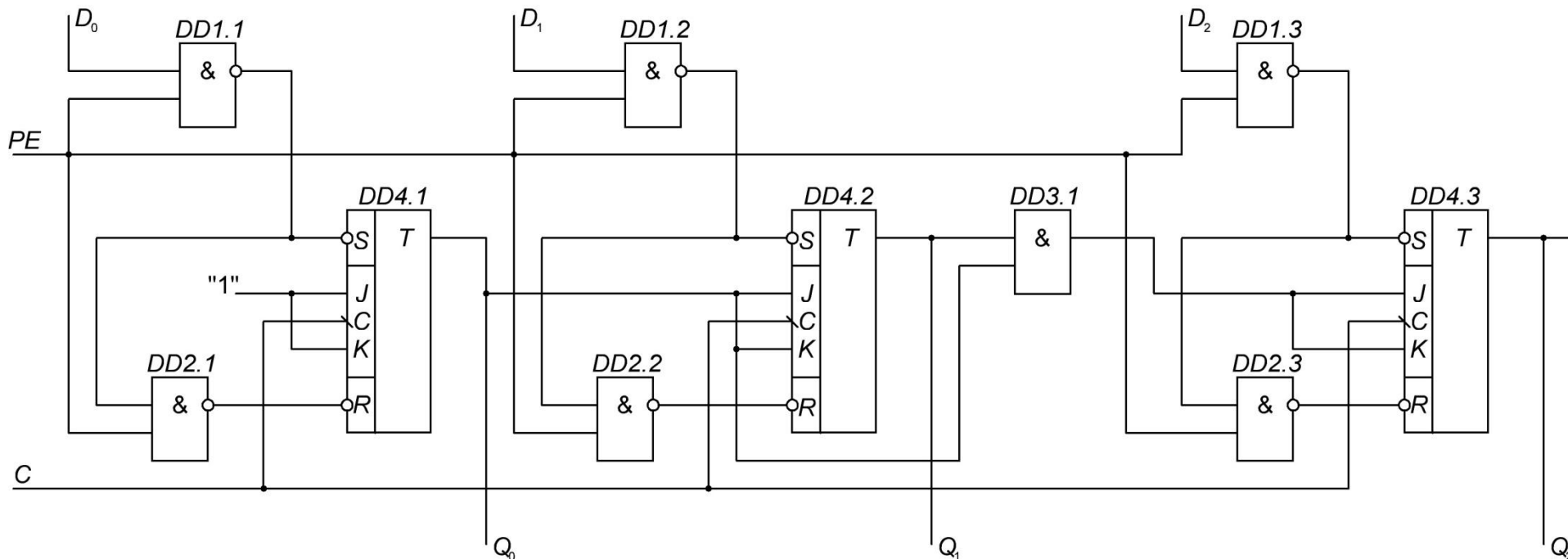
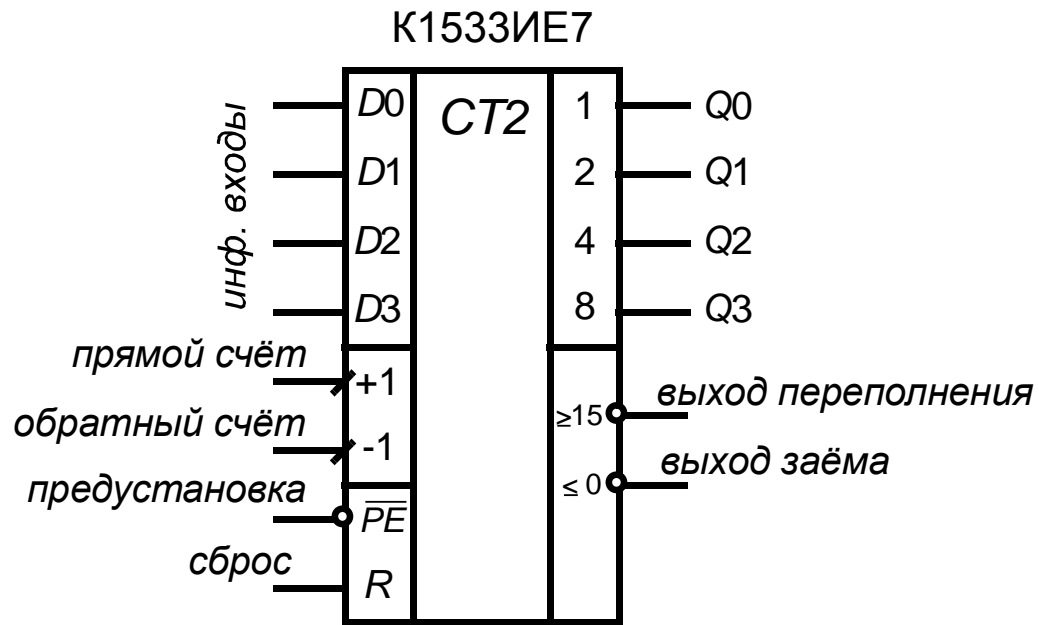


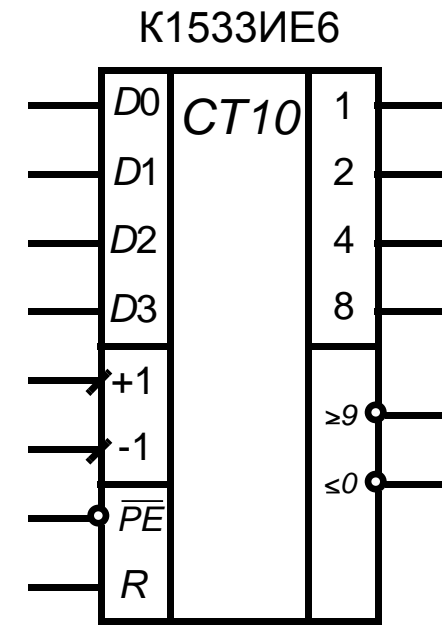
Схема трёхразрядного счётчика с предварительной установкой

На информационных входах D_0-D_2 устанавливаем число в двоичном коде. Запись в счётчик (предустановка) осуществляется при $PE = 1$. В этом случае на выходах Q_0-Q_2 появится двоичный код $Q_i = D_i$. С поступлением следующего тактового импульса счётчик начнёт счёт с установленного на выходах числа. Вход PE (*preset enable*) имеет приоритет над J, K и C .

Счётчики с предварительной установкой



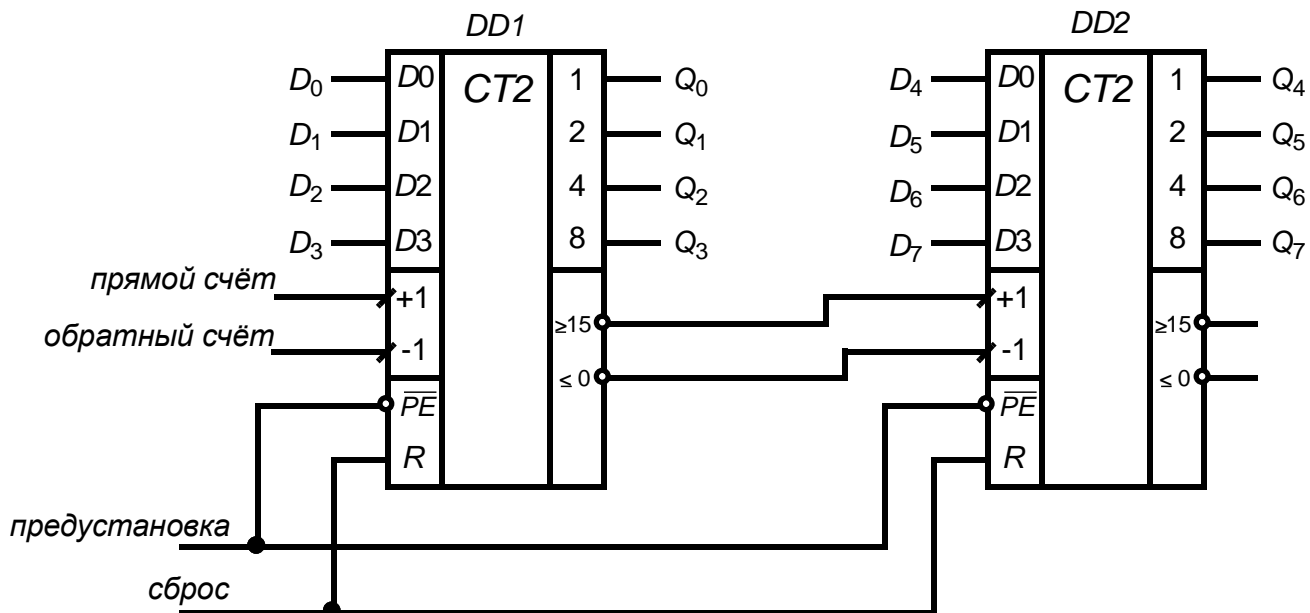
Двоичный счётчик с
предустановкой



Двоично-десятичный
счётчик с предустановкой

Предустановка счётчика осуществляется подачей «0» на вход *PE*.

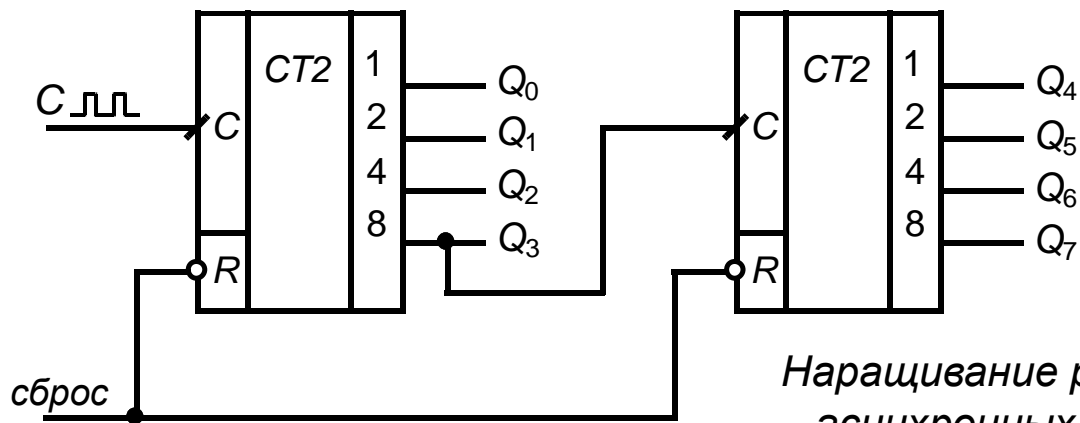
Способы увеличения разрядности счётчиков



Наращивание разрядности синхронных счётчиков с предустановкой

$D_0 - D_7$ – информационные входы
 $Q_0 - Q_7$ – выходы

Для наращивания разрядности счётчиков используются выходы переполнения (переноса) и заёма.



Наращивание разрядности асинхронных счётчиков

$2^8 = 256$
 считает от 0 до 255

Счётчики с произвольным коэффициентом счёта

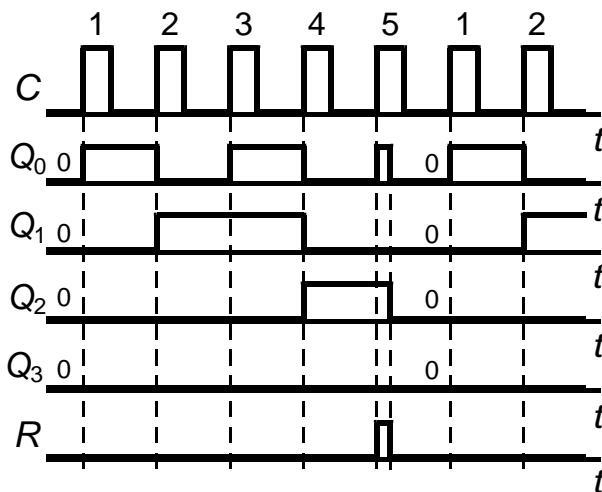
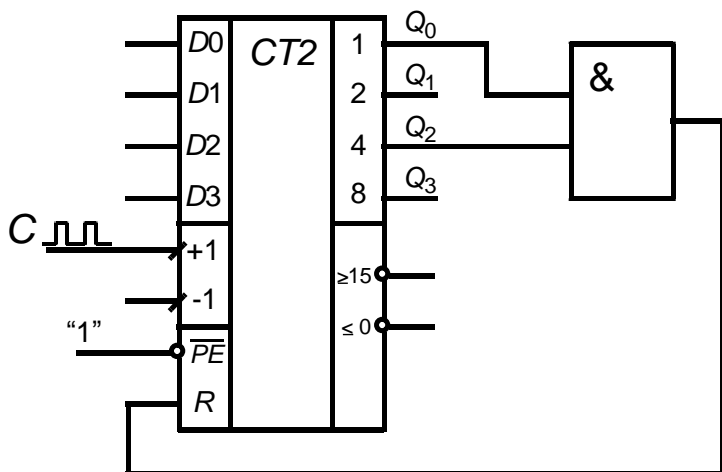
Два способа организации счётчиков с произвольным коэффициентом счёта:

1. метод управляемого сброса;
2. метод предварительной установки.

1. Метод управляемого сброса

Реализовать схему совпадения и выдать результат на вход сброса. Схема совпадения выполняется на логике.

Пример. Реализовать счётчик с коэффициентом счёта 5 на суммирующем счётчике.



$$K_{сч} = 5$$

$$5_{10} = 0101_2$$

Счётчик считает начиная с 0000 на увеличение. Как только на выходах Q_0 и Q_2 появятся единицы (что будет соответствовать числу 0101) на выходе элемента И появится лог. 1, которая поступая на вход R , сбрасывает счётчик. Далее счёт начинается заново.

Можно построить аналогичный счётчик на вычитающем счётчике. Т.к. счёт начинается с 15, схему совпадения необходимо делать на число

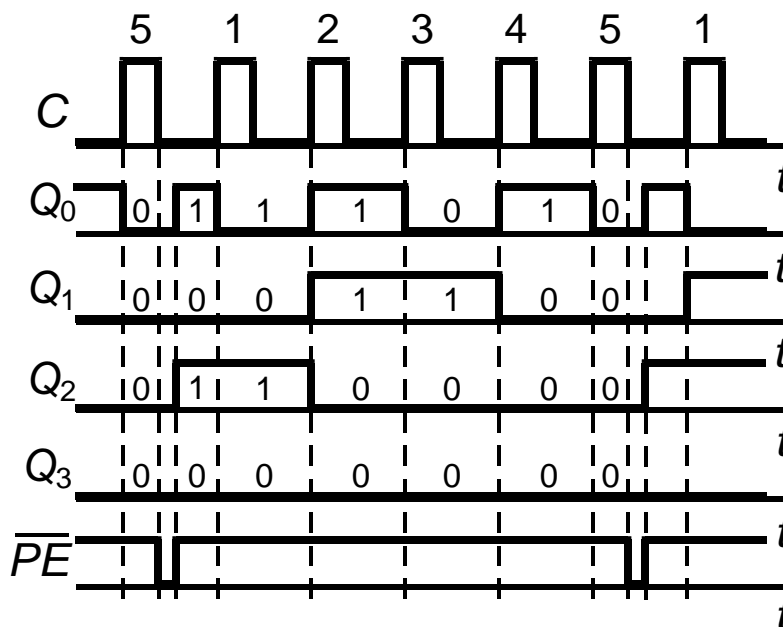
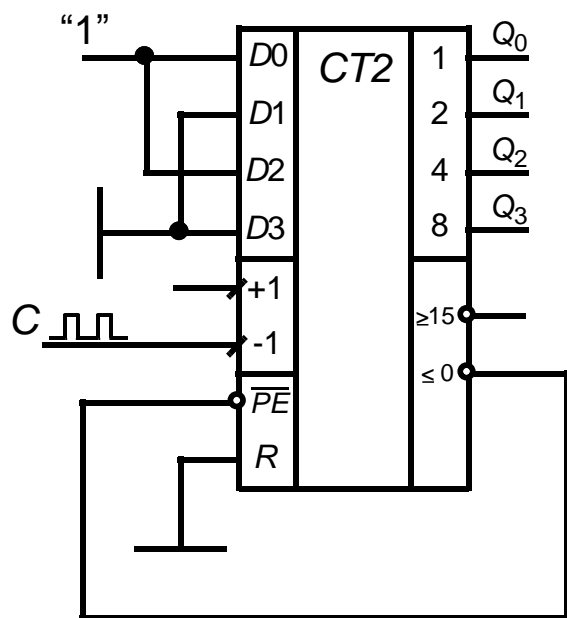
$$10_{10} = 15_{10} - 5_{10} = 1010_2.$$

Счётчики с произвольным коэффициентом счёта

2. Метод предварительной установки

Пример 1. Реализовать счётчик с коэффициентом счёта 5. Счётчик работает на уменьшение.

Т.к. счётчик работает на уменьшение (вычитание), то необходимо использовать выход заёма (≤ 0) для загрузки кода в счётчик. Коэффициент счёта задаётся числом в двоичном коде, загружаемое через информационные входы.



$$K_{сч} = 5$$

$$5_{10} = 0101_2$$

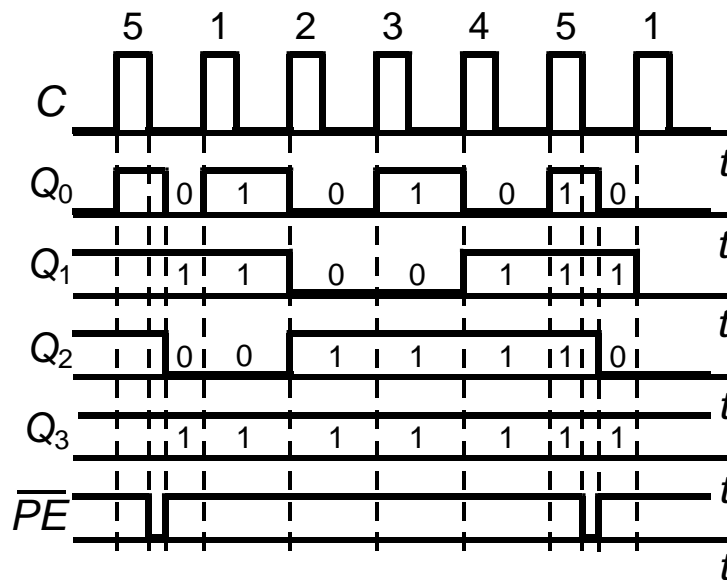
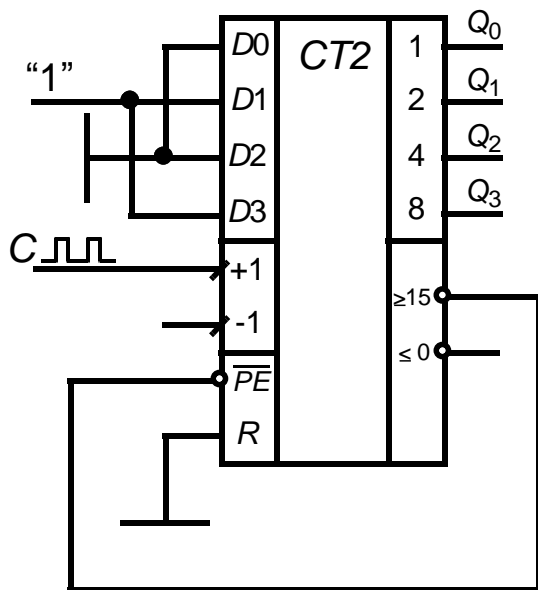
Для заданного коэффициента счёта на информационные входы необходимо подать число $5_{10} = 0101_2$. С каждым поступающим на вход импульсом содержимое счётчика уменьшается на 1. Когда счётчик обнулится (на выходах 0000) на выходе заёма появится лог. 0, который снова загрузит число $5_{10} = 0101_2$, и счёт начнётся заново.

Счётчики с произвольным коэффициентом счёта

2. Метод предварительной установки

Пример 2. Реализовать счётчик с коэффициентом счёта 5. Счётчик работает на увеличение.

Поскольку счётчик работает на увеличение, то необходимо использовать выход переполнения (переноса), а на информационные входы подать число равное $(15_{10} - K_{сч})$.



$$K_{сч} = 5$$

$$5_{10} = 0101_2$$

На информационные входы необходимо подать число:

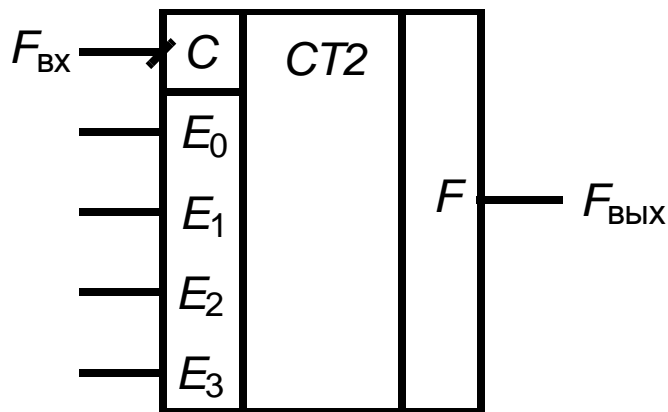
$$15_{10} - 5_{10} = 10_{10} = 1010_2.$$

Счётчик отсчитывает 5 импульсов. На выходе переполнения формируется сигнал (лог. 0), который вновь загружает число $10_{10} = 1010_2$ в счётчик, и счёт начинается заново. Перезагрузка счётчика происходит при его переполнении, когда $Q_0 = Q_1 = Q_2 = Q_3 = 1$.

Счётчики с произвольным коэффициентом счёта

Счётчики с установкой коэффициента счёта

Счётчики с установкой коэффициента счёта – специальные счётчики-делители частоты. Коэффициент деления (счёта) задаётся двоичным числом на информационных входах E_5-E_0 .



Частота выходных импульсов определяется по формуле:

$$F_{\text{ВЫХ}} = \frac{F_{\text{ВХ}}}{K_{\text{дел}}} = \frac{F_{\text{ВХ}}}{E_0 \cdot 2^0 + E_1 \cdot 2^1 + E_2 \cdot 2^2 + E_3 \cdot 2^3}$$

Регистры

Регистр – последовательностное устройство, предназначенное для приема, хранения, передачи и преобразования информации.

Регистр может выполнять следующие операции:

- установка регистра в исходное состояние (сброс – запись нулевого кода);
- запись двоичного слова в регистр (в последовательной и/или параллельной форме);
- хранение информации;
- сдвиг хранимой информации влево или вправо;
- преобразование хранимой информации из последовательной формы в параллельную и наоборот;
- поразрядные логические операции.

Регистры строятся на базе *D*-триггеров.

Различают **регистры памяти** и **регистры сдвига**.

Регистры памяти предназначены для временного хранения информации (данных).

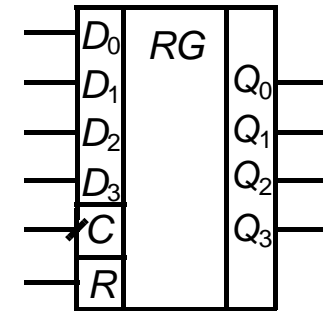
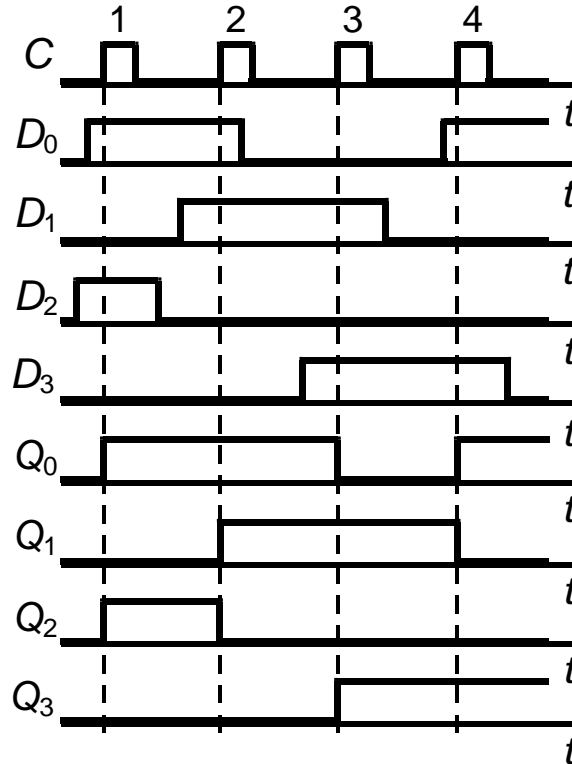
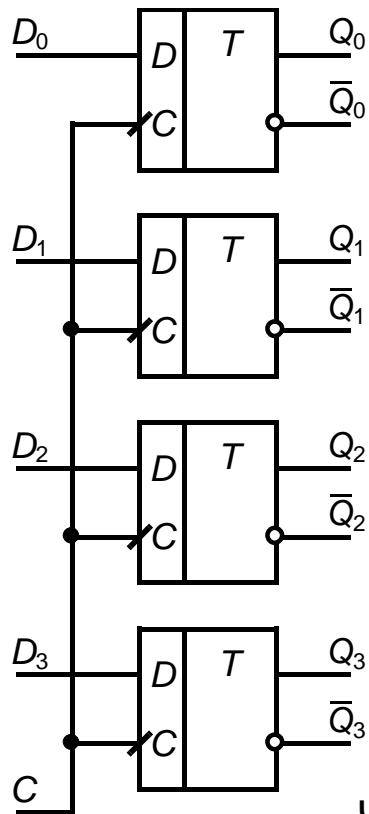
Регистры сдвига (сдвиговые регистры) предназначены как для временного хранения информации, так и для ее сдвига в сторону младших разрядов (вправо), так и в сторону старших разрядов (влево) без изменения информации. Кроме того, регистры сдвига могут преобразовывать информацию из параллельной формы в последовательную и наоборот.

Параллельный код (форма) – информация (двоичное число), появляющаяся на всех разрядах регистра одновременно.

Последовательный код (форма) – информация (двоичное число), появляющаяся на одном разряде регистра последовательно с каждым тактовым импульсом.

Регистры памяти (параллельные регистры)

Регистры памяти осуществляют: прием (запись), хранение и передачу информации. Параллельными регистры называют из-за того, что запись двоичного числа осуществляется во все разряды регистра одновременно, т.е. в параллельном коде. Все операции выполняются также в параллельном коде, поэтому разряды регистра между собой не связаны.



УГО регистра памяти

Запись информации в триггеры регистра может происходить по переднему и заднему фронту, а также по уровню.

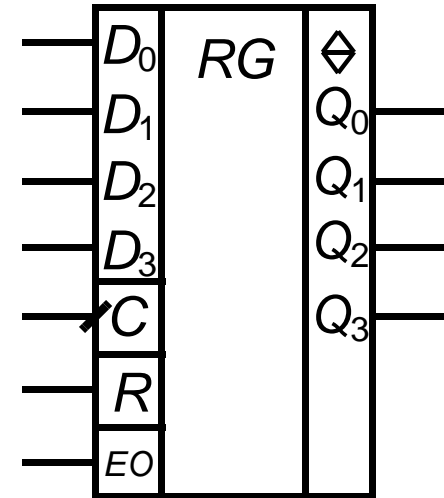
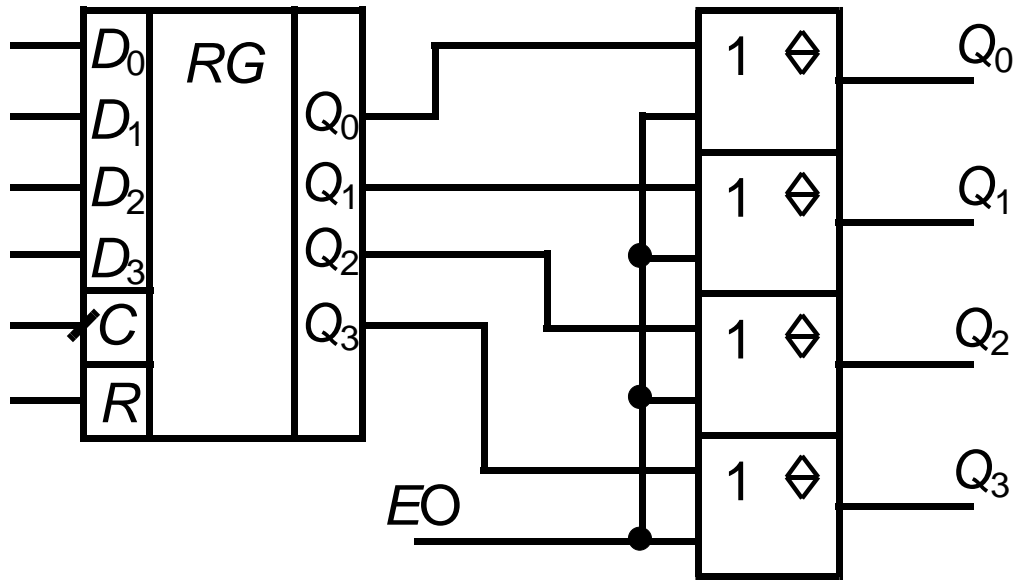
Часто регистры снабжаются входами асинхронного сброса R .

Регистры, работающие по фронту называются **регистрами с динамическим тактовым входом**.

Регистры работающие по уровню – **со статическим тактовым входом**.

Регистры памяти с тремя состояниями на выходе

Такие регистры образуются добавлением к регистру элементов с тремя состояниями.



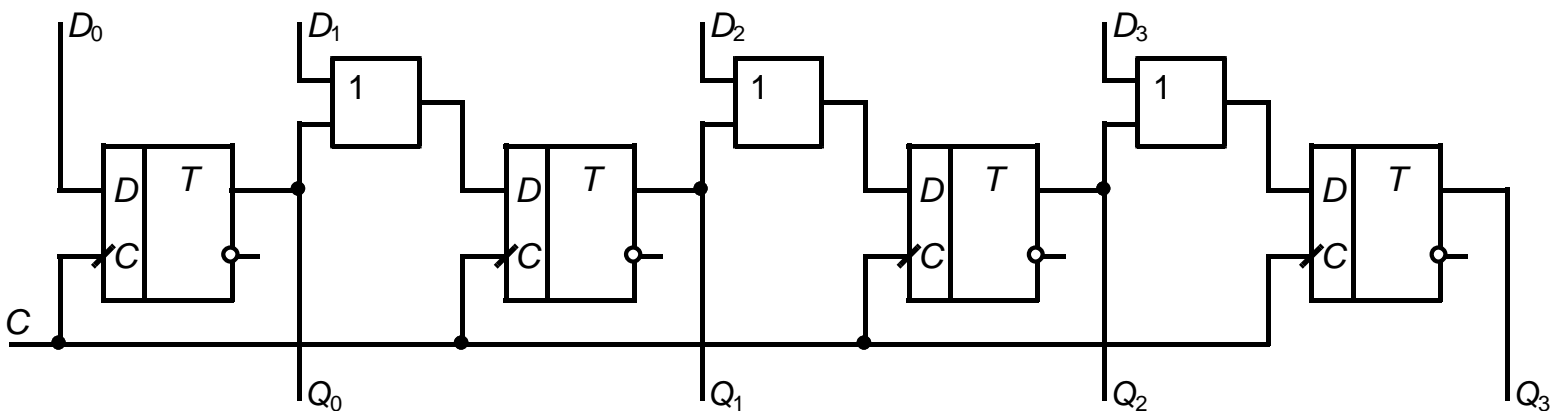
УГО регистра памяти с тремя состояниями

Регистры памяти выпускаются 4-х, 8-х и 16-ти разрядными.

Назначение: используются в микропроцессорных системах при работе на общую нагрузку (шина адреса, шина данных).

Регистры сдвига (последовательные регистры)

Для создания регистров триггеры соединяются последовательно.



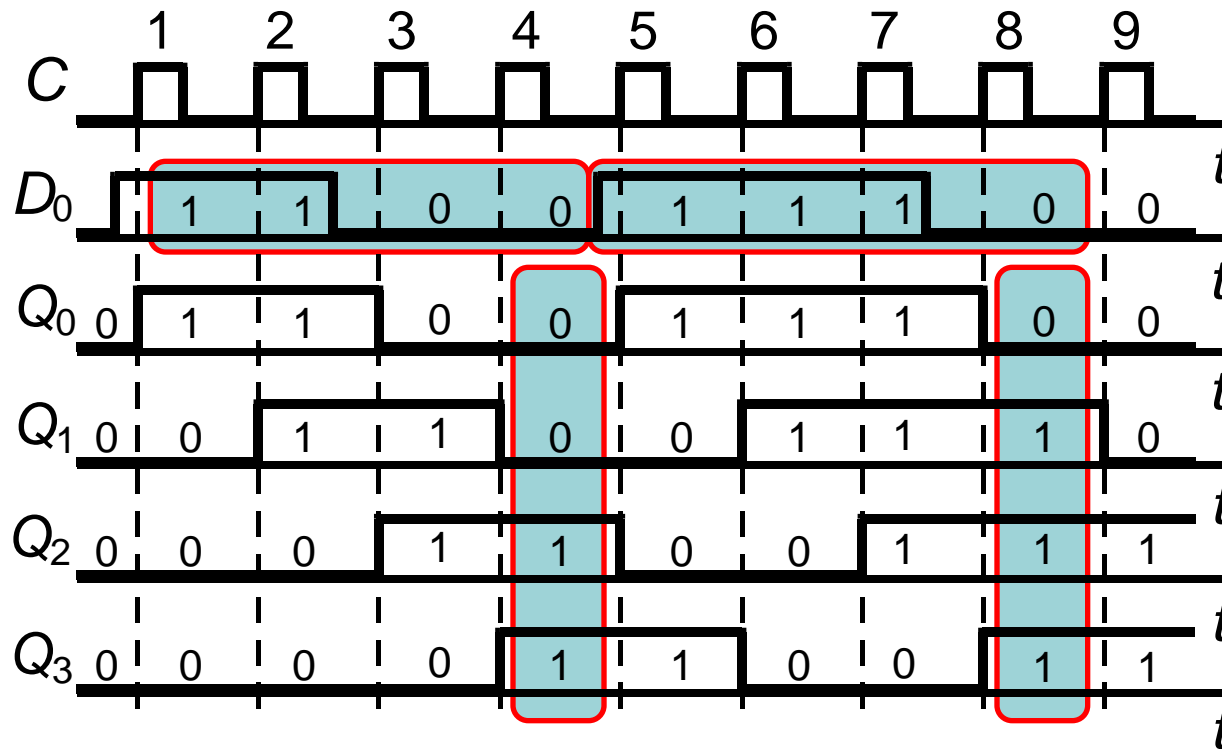
Для установки в исходное состояние (запись нулевого кода) триггеры снабжаются входом сброса R (на схеме не показан).

Данный регистр представляет собой последовательно соединенные двоичные ячейки памяти (триггеры). Под действием тактовых импульсов состояния ячеек памяти сдвигаются (передаются) на последующие ячейки.

Схема регистра позволяет преобразовать информацию из последовательного в параллельный код и наоборот.

Регистры сдвига

1. Преобразование информации из последовательного кода в параллельный

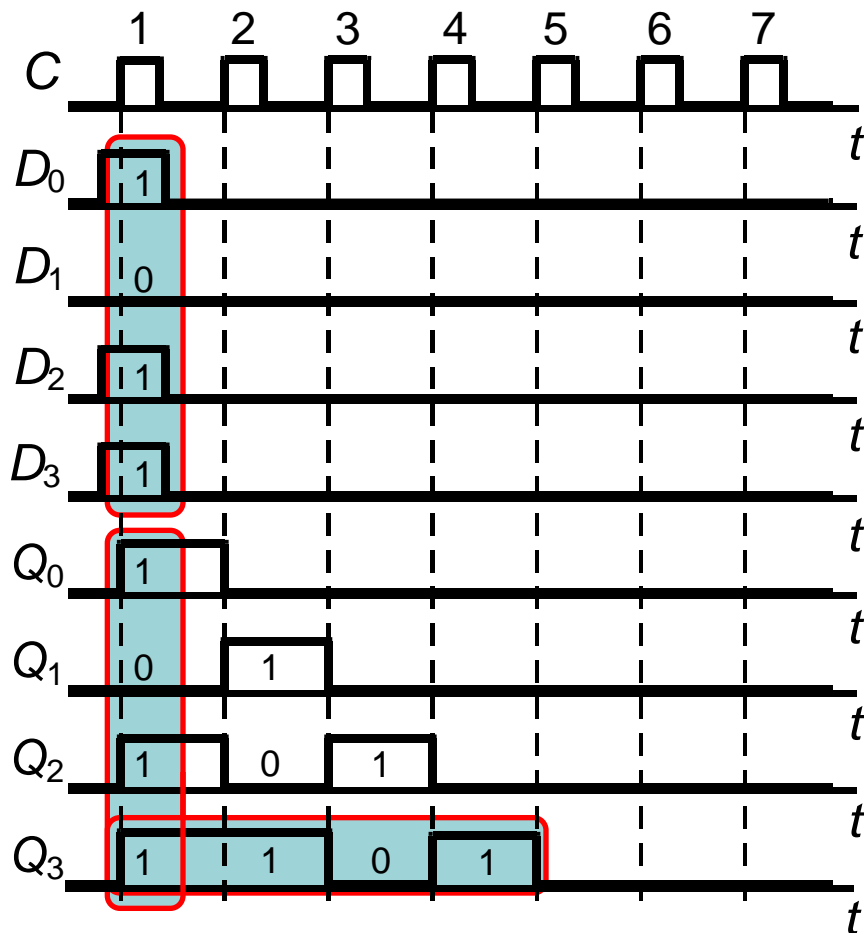


Вход D_0 используется в качестве информационного. На вход D_0 поступает информация в последовательном коде, т.е. данные вводятся в регистр последовательно (поразрядно). D -триггер – триггер с задержкой (передает информационный сигнал с D -входа на выход Q с задержкой в один такт).

Из диаграмм видно, что данные со входа D_0 последовательно передаются (сдвигаются) с каждым тактовым импульсом. За 4 такта двоичное число появляется на выходах Q_0-Q_3 в параллельном коде.

Регистры сдвига

2. Преобразование информации из параллельного кода в последовательный



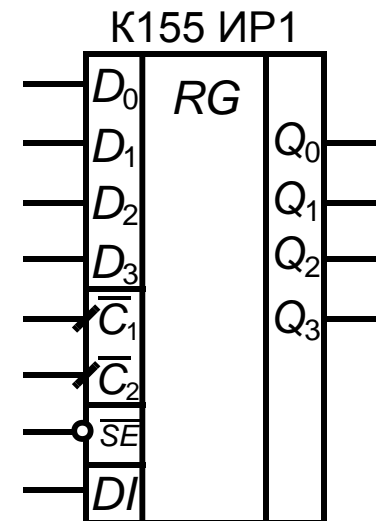
В первом такте информация на D -входах записывается в триггеры и с каждым последующим тактовым импульсом сдвигается влево (в сторону старших разрядов). На выходе Q_3 через 4 такта получаем информацию в последовательном коде.

Существуют универсальные регистры, выполняющие все описанные ранее операции.

SE – выбор режима (память/сдвиг): параллельная или последовательная загрузка;

C_1 – тактовый вход регистра сдвига;

C_2 – тактовый вход регистра памяти.



Регистры сдвига

Математические операции умножения и деления

С помощью регистров можно выполнять операции умножения и деления на 2.

Умножение хранящегося в регистре числа на 2 выполняется путем его сдвига влево и записью «0» в младший разряд.

Пример.

$00011_2 = 3_{10}$ - пусть задано данное число

$00110_2 = 6_{10}$ - сдвигаем влево (умножаем на 2)

$01100_2 = 12_{10}$ - сдвигаем влево (умножаем на 2)

Таким образом умножили число 3 на 4 и получили 12.

Деление на 2 осуществляется сдвигом хранящегося в регистре числа вправо и записью «0» в старший разряд, причем деление на 2 **целочисленное**.

Пример.

$01110_2 = 14_{10}$ - пусть задано число 14

$00111_2 = 7_{10}$ - сдвигаем вправо (делим на 2)

$00011_2 = 3_{10}$ - сдвигаем вправо (делим на 2)

В итоге разделили число 14 на 4 и получили целое число 3.

Арифметико-логическое устройство (АЛУ)

АЛУ - устройство, предназначенное для выполнения логических и математических операций над двоичными числами. АЛУ входит в состав любого микропроцессора.

ALU – Arithmetic Logic Unit

Разобрать самостоятельно!

1. Лачин В.И., Савёлов Н.С. Электроника: Учебное пособие. – Ростов н/Д: Феникс, 2007. – 703, [1] с.