[1 (47). Понятие функционально-полного набора логических элементов. 3](#_Toc137924676)

[2 (41). Условное графическое обозначение элементов "И", "ИЛИ", "НЕ", и реализация данных функций на элементах «И-НЕ». 3](#_Toc137924677)

[3 (27). Принцип построения электронной схемы логического элемента «ИЛИ». 5](#_Toc137924678)

[4 (26). Принцип построения электронной схемы логического элемента «И». 5](#_Toc137924679)

[5 (33). В чем состоит различие понятий "комбинационная схема" и " цифровой автомат". Примеры устройств, относящихся к двум этим классам устройств 6](#_Toc137924681)

[6 (35,56). Пример применения закона Де Моргана при преобразованиях комбинационных схем. 7](#_Toc137924682)

[7 (17). Принципы преобразований комбинационных схем из одного базиса в другой. Пояснить примером. 8](#_Toc137924683)

[8 (2). Комбинационные схемы, воспроизводящие функции И и ИЛИ на элементах Шеффера. 9](#_Toc137924684)

[9 (31). Определение ДНФ двоичной функции. 9](#_Toc137924685)

[10 (30). Определение КНФ двоичной функции. 9](#_Toc137924686)

[11 (4). Технология минимизации булевых функций с помощью карт Карно. 9](#_Toc137924687)

[12 (12). Определение полного и неполного дешифратора. Примеры. 11](#_Toc137924688)

[13 (37). Шифратор на 10 входов. Принцип синтеза схемы шифратора. 12](#_Toc137924689)

[14 (6). Пример дешифратора с разрешающим входом. 13](#_Toc137924690)

[15 (55). Каскадный дешифратор. Преимущества по сравнению с линейным дешифратором. 14](#_Toc137924691)

[16 (46). Матричный дешифратор, схема построения и преимущество перед линейным дешифратором. 14](#_Toc137924692)

[17 (18). Определение и пример реализации мультиплексора. 15](#_Toc137924693)

[18 (42). Схема демультиплексора. Пояснить принцип работы. 16](#_Toc137924694)

[19 (10). Назначение и классификация триггеров. 17](#_Toc137924695)

[20 (16). Реализация RS-триггера на элементах Шеффера и Пирса. 18](#_Toc137924696)

[21 (54). Принцип работы асинхронного RS-триггера, на примере схемы на четырёх биполярных транзисторах. 19](#_Toc137924697)

[22 (1). Различия в схемах построения и работе асинхронных и синхронных RS-триггеров. 20](#_Toc137924698)

[23 (11). Принцип построения MS-триггеров (с задержкой) и особенности их работы. 21](#_Toc137924699)

[24 (21). Схема построения и принцип записи информации в триггерах типа MS (c задержкой). 22](#_Toc137924700)

[25 (25). Различие в работе триггеров потенциального и динамического типа. Изобразить на временных диаграммах. 23](#_Toc137924701)

[26 (49). Временная диаграмма, таблица переходов и принцип построения универсального JK - триггера. 24](#_Toc137924702)

[27 (8). Схема построения универсального J-K- триггера и его временная диаграмма работы. 25](#_Toc137924703)

[28 (3). Построение D-триггера на основе универсального JK-триггера. 25](#_Toc137924704)

[29 (45). Таблица переходов, временная диаграмма и схема построения синхронного D-триггера на элементах И-НЕ. 26](#_Toc137924705)

[30 (9). Назначение установочных RS-входов в счётном Т-триггере. 27](#_Toc137924706)

[31 (15,58,60). Варианты реализации счётного T-триггера. Назначение асинхронных S и R входов. 27](#_Toc137924707)

[32 (0). Организация регистра сдвига и схема сдвигающего регистра на D-триггерах. 30](#_Toc137924708)

[33 (20). Варианты обнуления содержимого регистра сдвига. 30](#_Toc137924709)

[34 (39). Пример построения регистра с параллельным занесением кода. 31](#_Toc137924710)

[35 (36). Схема универсального регистра и принцип управления переключения режимов работы. 32](#_Toc137924711)

[36 (7). Какой недостаток характерен для асинхронных счетчиков. Показать на примере временной диаграммы. 33](#_Toc137924712)

[37 (13). Пример организации счетчика по модулю десять. Перечислите варианты методов ограничения модуля счётчика. 34](#_Toc137924713)

[38 (44). Построение вычитающего асинхронного счетчика на J-K триггерах. Временная диаграмма работы счетчика. 36](#_Toc137924714)

[39 (62). Синхронный суммирующий счётчик на JK-триггерах. Принцип управления переключения разрядов. 37](#_Toc137924715)

[4- (34). Принцип работы и схема построения синхронного счётчика на J-K триггерах. 37](#_Toc137924716)

[41 (14). Схема реверсивного счетчика на JK-триггерах, пояснить принцип управления направлением счёта. 38](#_Toc137924717)

[42 (22). Счетчик Джонсона: схема построения и диаграмма работы. 39](#_Toc137924718)

[43 (51). Организация двоично-десятичного сумматора. Назначение сумматора коррекции. 42](#_Toc137924719)

[44 (52). Варианты реализации сумматоров с различными схемами организации переносов. 43](#_Toc137924720)

[45 (40). Реализация многоразрядного сумматора последовательного действия. 46](#_Toc137924721)

[46 (63). Особенность структуры сумматора модифицированного обратного кода. 47](#_Toc137924722)

[47 (38). Особенность структуры сумматора обратного модифицированного кода. Обнаружение переполнения. Пояснить примером. 48](#_Toc137924723)

[48 (32). Выполнение операции вычитания на сумматорах. Сумматор дополнительного модифицированного кода. Принцип построения. 49](#_Toc137924724)

[49 (43). Схема и принцип построения полного вычитателя. 49](#_Toc137924725)

[50 (50). Схема построения универсального сумматора-вычитателя. Пояснить принцип управления режимами. 50](#_Toc137924726)

[51 (48). Особенности схем построения инкременторов и декременторов. 52](#_Toc137924727)

[52 (59). Применение компараторов. Пример синтеза схемы одноразрядного компаратора. 53](#_Toc137924728)

[53 (57). Многоразрядный компаратор на равенство. Пример схемы реализации. 54](#_Toc137924729)

[54 (53). Множительный блок. Методика построения множительного блока. 55](#_Toc137924730)

[55 (61). Структура простого АЛУ, предназначенного для выполнения арифметических операций. Интерфейс микросхемы АЛУ. 56](#_Toc137924731)

[56 (19). Поясните принцип хранения бита информации в динамическом ОЗУ. 57](#_Toc137924732)

[57 (24). Принцип построения однократно-программируемого постоянного запоминающего устройства. 58](#_Toc137924733)

[58 (28). Принцип построения перепрограммируемого постоянного запоминающего устройства. 60](#_Toc137924734)

[59 (23). Принцип построения схемы масочного постоянного запоминающего устройства. 62](#_Toc137924735)

[60 (29). Принцип построения статического оперативного запоминающего устройства. 64](#_Toc137924736)

ОСНОВЫ СХЕМОТЕХНИКИ

# **1 (47). Понятие функционально-полного набора логических элементов.**

МБ НЕ НАДО (Понятие логического элемента. Логические операции над двоичными переменными реализуются схемами, которые называются логическими элементами. Число входов логического элемента соответствует числу аргументов воспроизводимой им булевой функции)

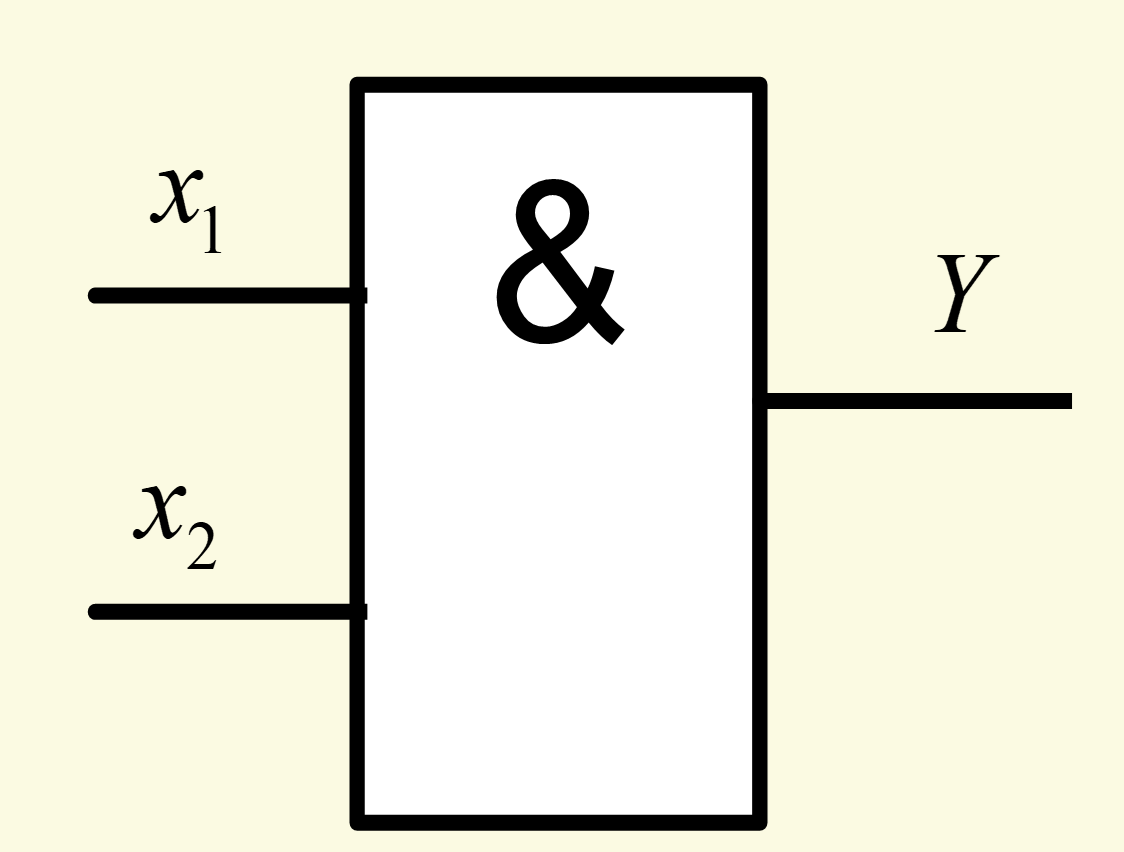
**Функционально-полный набор –** это набор функций, с помощью суперпозиции которых можно реализовать любую другую булеву функцию.

**Набор функций дизъюнкции, конъюнкции и инверсии,** который соответствует трём операциям булевой алгебры-логики,получил название *Основного функционально-полного набора*.

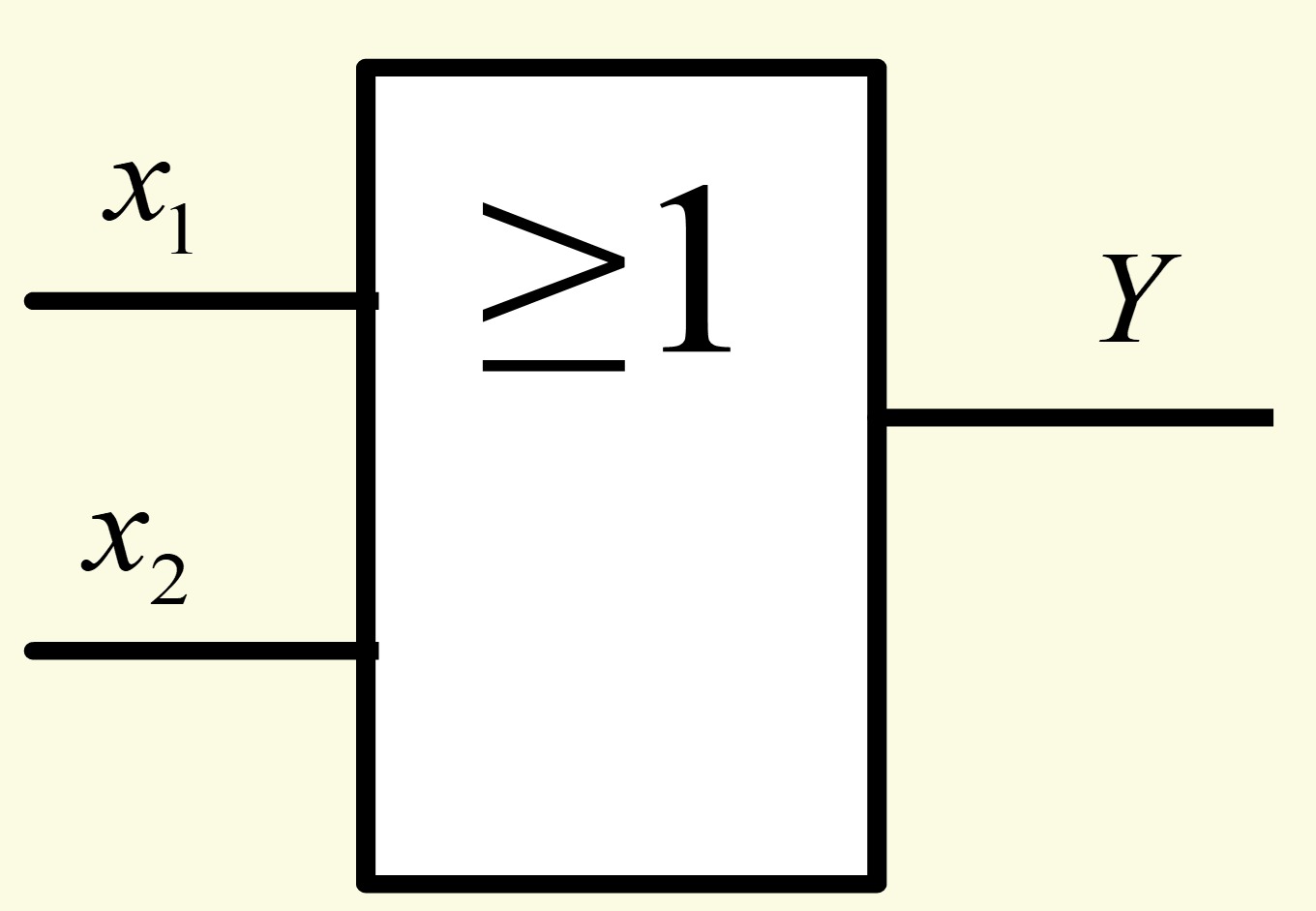
Двумя другими функционально полными наборами являются **функции** **Пирса и Шеффера**.

# **2 (41). Условное графическое обозначение элементов "И", "ИЛИ", "НЕ", и реализация данных функций на элементах «И-НЕ».**

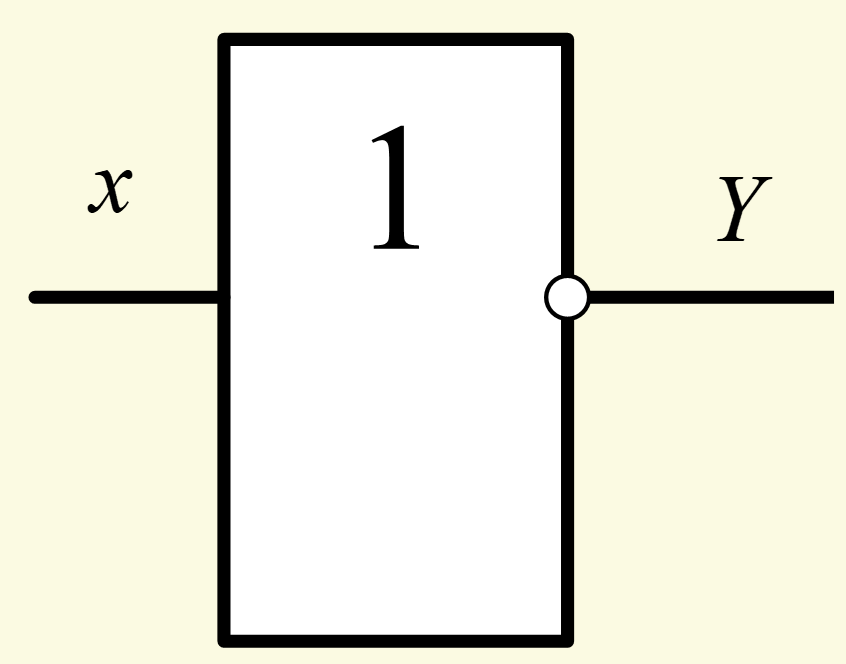
Логический элемент «И»



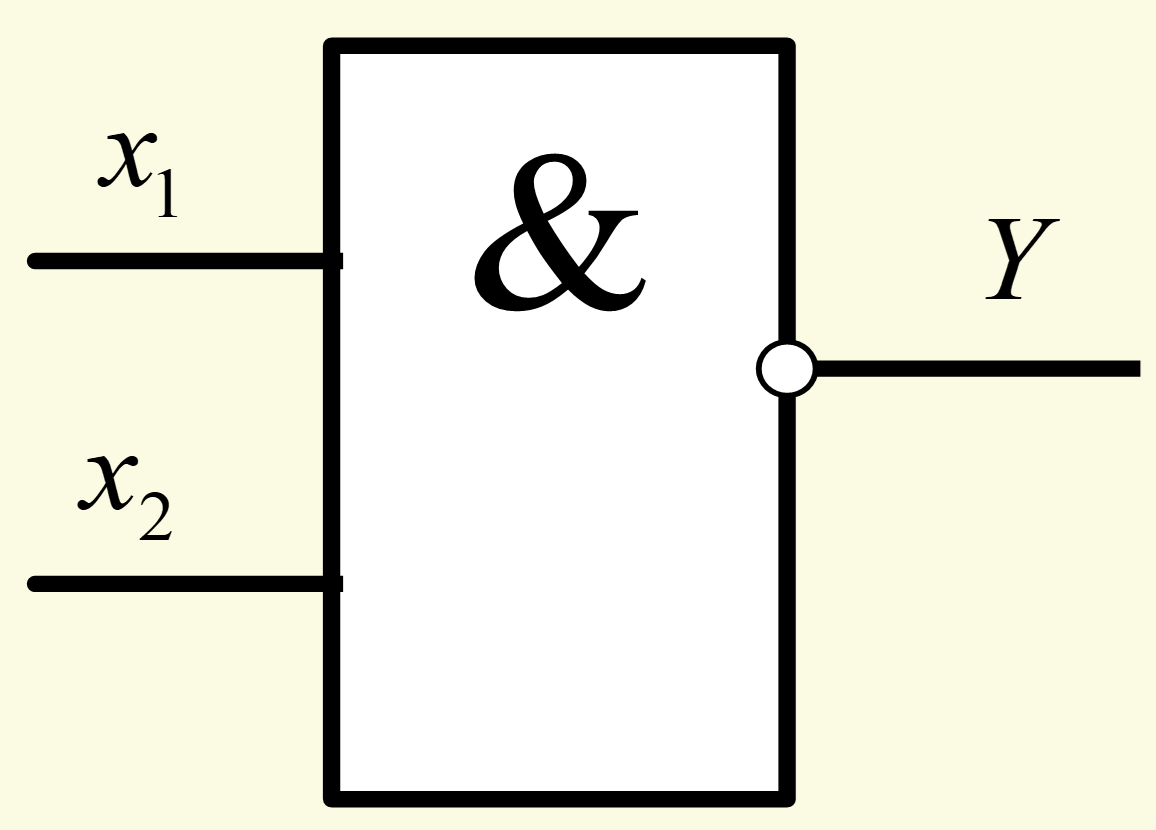
Логический элемент «ИЛИ»



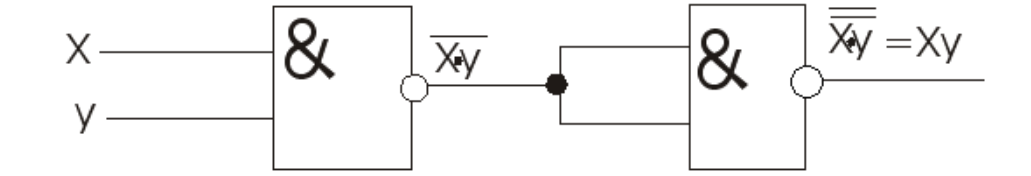
Логический элемент «НЕ»



Логический элемент «И-НЕ»



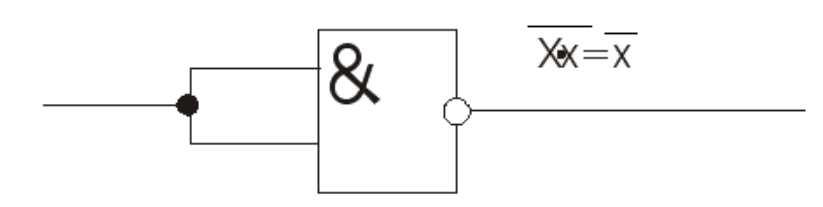
Реализация «И»:



Реализация «ИЛИ»:



Реализация «НЕ»:

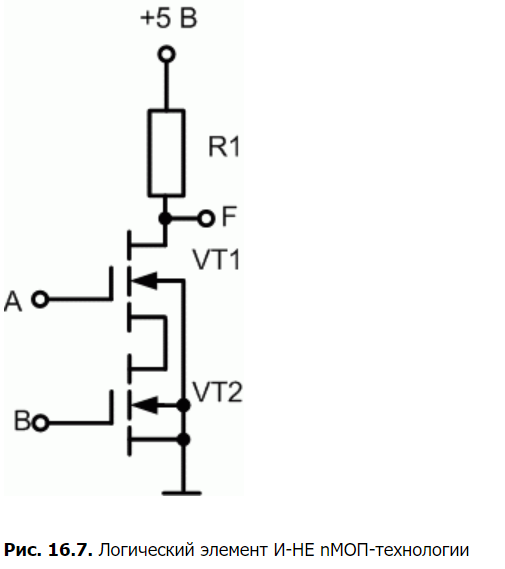
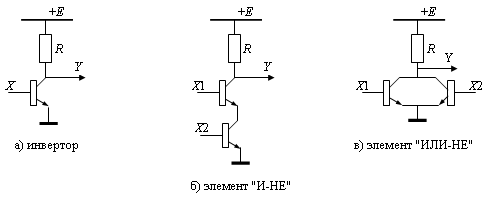


# **3 (27). Принцип построения электронной схемы логического элемента «ИЛИ».**

На каждый из входов может подаваться сигнал в виде какого-то напряжения (единица) или его отсутствия (ноль). На резисторе R появиться напряжение даже при его появлении на каком – либо из диодов.

# **4 (26). Принцип построения электронной схемы логического элемента «И».**

Если хотя – бы к одному из входов будет сигнал равный нулю, то через диод будет протекать ток. Падение напряжения на диоде стремится к нулю, соответственно на выходе тоже будет ноль. На выходе сможет появится сигнал только при условии, что все диоды будут закрыты, то есть на всех входах будет сигнал. Рассчитаем уровень сигнала на выходе устройства:

****Убираем в б) и в) землю, ставим на ее место Y – получаем И и ИЛИ. Биполярные транзисторы спокойно меняются на МОП (рисунок справа)

# **27.**

# **5 (33). В чем состоит различие понятий "комбинационная схема" и " цифровой автомат". Примеры устройств, относящихся к двум этим классам устройств**

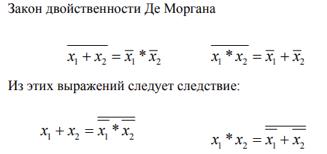
Комбинационная схема – устройство без памяти. Выходное значение зависит только от входного в текущий момент

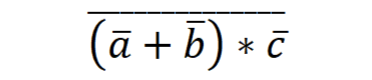
Цифровой автомат – устройство с памятью. Выходное значение зависит от входного в текущий момент и от значения в памяти.

КС – мультиплексоры, шифраторы, т.е. элементы реализующие булевы функции.

ЦА – триггеры, регистры, счетчики

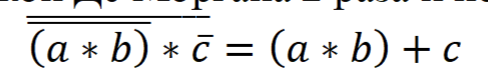
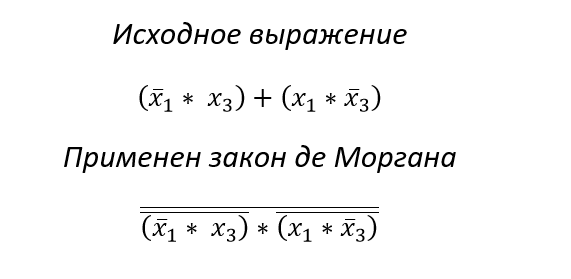
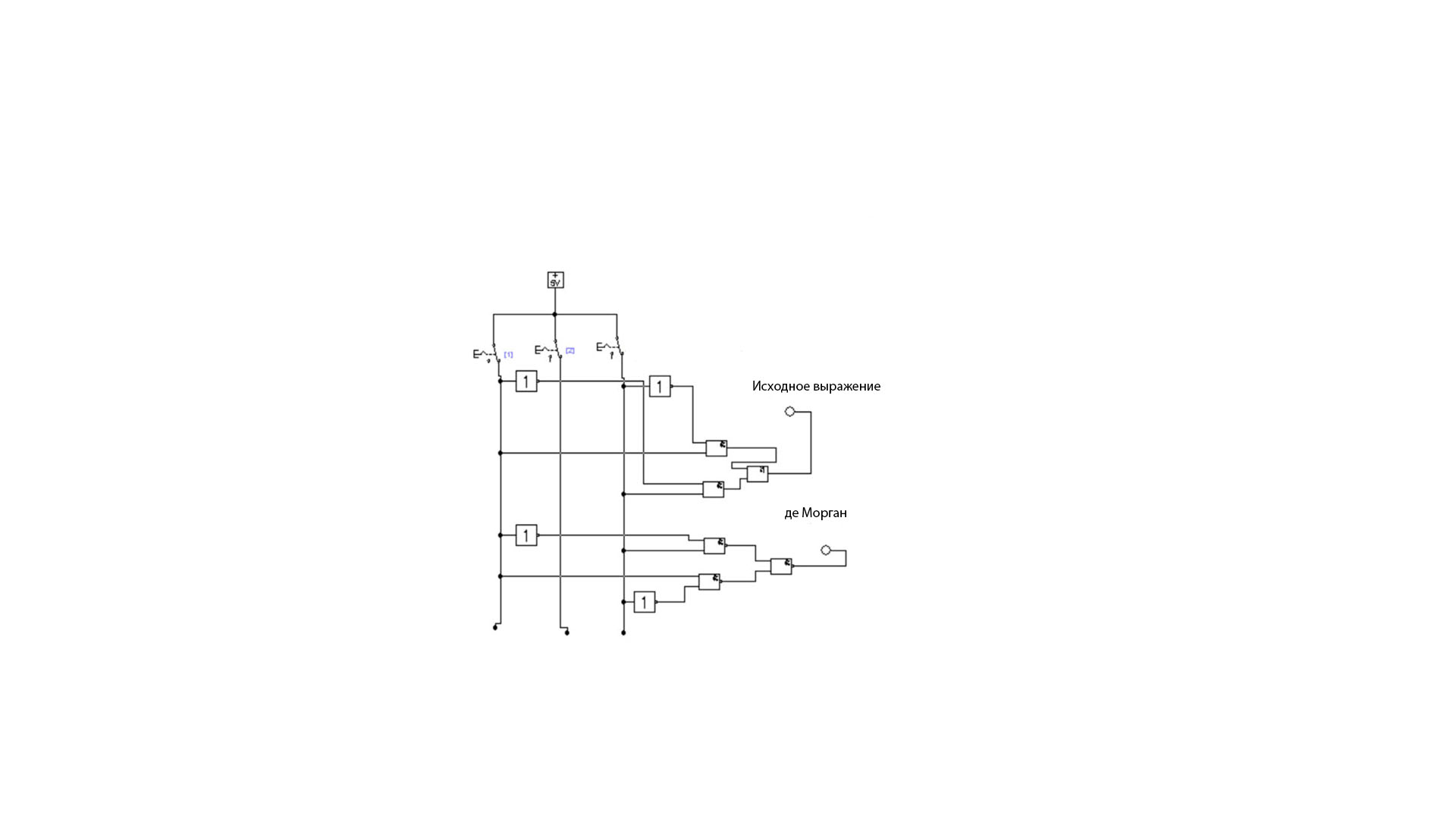
# **6 (35,56). Пример применения закона Де Моргана при преобразованиях комбинационных схем.**



С помощью этих законов можно упростить функцию, которой соответствует Комб. Схема, и перестроить заново по новой функции. Пример:

Например, для функции , требующей в таком виде 4 элемента НЕ и по элементу И и ИЛИ можно применить закон Де Моргана 2 раза и получить:

Что сокращает количество элементов до 2.



# **7 (17). Принципы преобразований комбинационных схем из одного базиса в другой. Пояснить примером.**

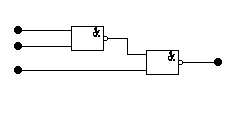
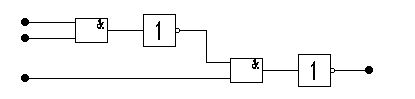
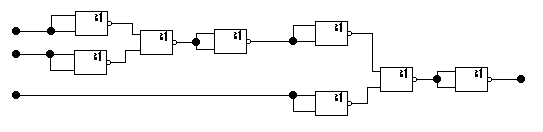
Базисов много. Наиболее интересные из них – полные. Полные – значит, можно реализовать любую логическую функцию.

Полные базисы: И-НЕ-ИЛИ (стандартный), Шеффера, Пирса.

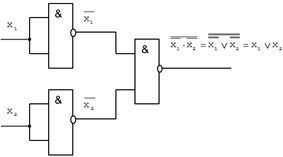
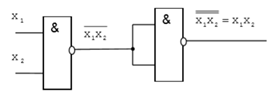
2 закона Де Моргана позволяют переводить КНФ и ДНФ в базисы НЕ-ИНЕ, НЕ-ИЛИНЕ, но базисы ИНЕ и ИЛИНЕ полные, значит, КНФ и ДНФ можно представить в базисах ИНЕ и ИЛИНЕ.

Из любого базиса комбинационную схему можно преобразовать в базис И, ИЛИ, НЕ. Так как любой другой базис реализуется на этих элементах, то не составит труда получить схему на нужном нам базисе.

Пример:

Схема на базисе Шеффера:  
  
Схема на базисе И, ИЛИ, НЕ:   
  
Схема на базисе Пирса:  


# **8 (2). Комбинационные схемы, воспроизводящие функции И и ИЛИ на элементах Шеффера.**



# **9 (31). Определение ДНФ двоичной функции.**

ДНФ функции – такая форма, в которой функция приведена к дизъюнкции конъюнкций переменных/их инверсий. При этом в 1 конъюнкции переменная может содержаться только 1 раз



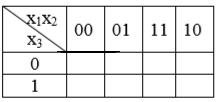
# **10 (30). Определение КНФ двоичной функции.**

КНФ функции – такая форма, в которой функция приведена к конъюнкции элементарных дизъюнкций переменных/их инверсий. При этом в 1 дизъюнкции переменная может содержаться только 1 раз



# **11 (4). Технология минимизации булевых функций с помощью карт Карно.**

Карту Карно можно рассматривать как графическое представление совокупности всех наборов переменных для данного числа переменных. Каждый набор переменных изображается на карте в виде клетки. Таким образом, при n=3 карта имеет 8 клеток, а при n=6 – 64 клетки.



Карта Карно образуется путем такого расположения клеток, при котором наборы переменных, находящиеся в соседних клетках, отличаются значением одной переменной. В картах Карно соседними считаются также крайние клетки каждого столбца или строки. Расположенные в них наборы переменных отличаются значением одной переменной.

Минтермы логической функции, т.е. наборы двоичных переменных, при которых эта функция равна 1, отмечаются единицами в соответствующих клетках. Для наборов переменных не входящих в логическую функцию соответствующие им клетки остаются пустыми.

Логическая функция, записанная в СДНФ или заданная в виде таблицы истинности, переносится на карту Карно. Затем карта покрывается контурами. В контур может входить 2n рядом расположенных клеток, содержащих единичное значение логической функции, т.е. 2,4,8 и т.д. точек. Допускается пересечение контуров. Два минтерма, находящиеся в соседних клетках, т.е. в одном контуре, могут быть заменены одним логическим произведением, содержащим на одну переменную меньше. Исключается та переменная, которая меняет своё значение при переходе из одной клетки в другую. В общем случае, наличие единиц в 2n соседних клетках позволяет исключить n переменных.

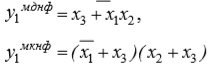
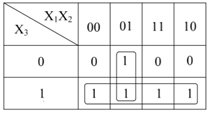
При минимизации с помощью карт Карно рекомендуется следовать следующему правилу:

Необходимо образовывать контура в которые входило бы максимально возможное количество клеток с минтермами - произведение будет наиболее простым. Контуров должно быть как можно меньше, чтобы было меньше слагаемых.

После покрытия карты контурами производится их анализ с точки зрения уменьшения числа переменных. На основе анализа контуров записывается минимизированная ДНФ (МДНФ) логической функции в виде логической суммы логических произведений двоичных переменных. При этом двоичные переменные, имеющие единичное значение, записываются без инверсии, а имеющие нулевое значение с инверсией.

у1 = (0, 1, 0, 1, 0, 1, 0, 1)

Карта Карно для этой функции:



ДЕШИФ, ШИФ, МУЛЬТ, ДЕМУЛЬТ

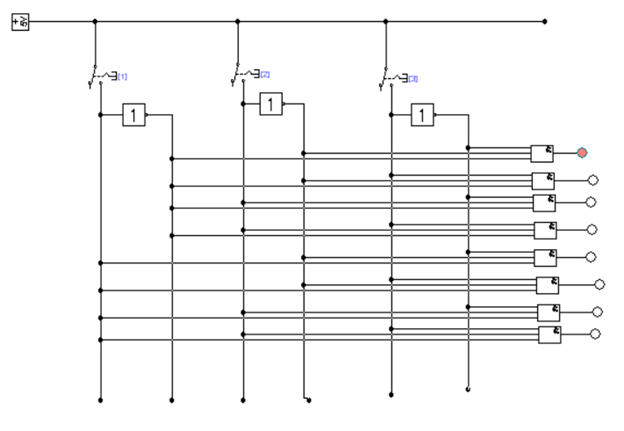
# **12 (12). Определение полного и неполного дешифратора. Примеры.**

Дешифратором называется комбинационная схема с несколькими входами и выходами, преобразующая код, подаваемый на входы, в сигнал на одном из выходов (унарный код). То есть если на входы дешифратора подаются двоичные переменные, то на одном из выходов дешифратора вырабатывается сигнал 1, а на остальных выходах сохраняются нули.

Число входов и выходов в полном дешифраторе связано соотношением m = 2n, где n – число входов, а m – число выходов.

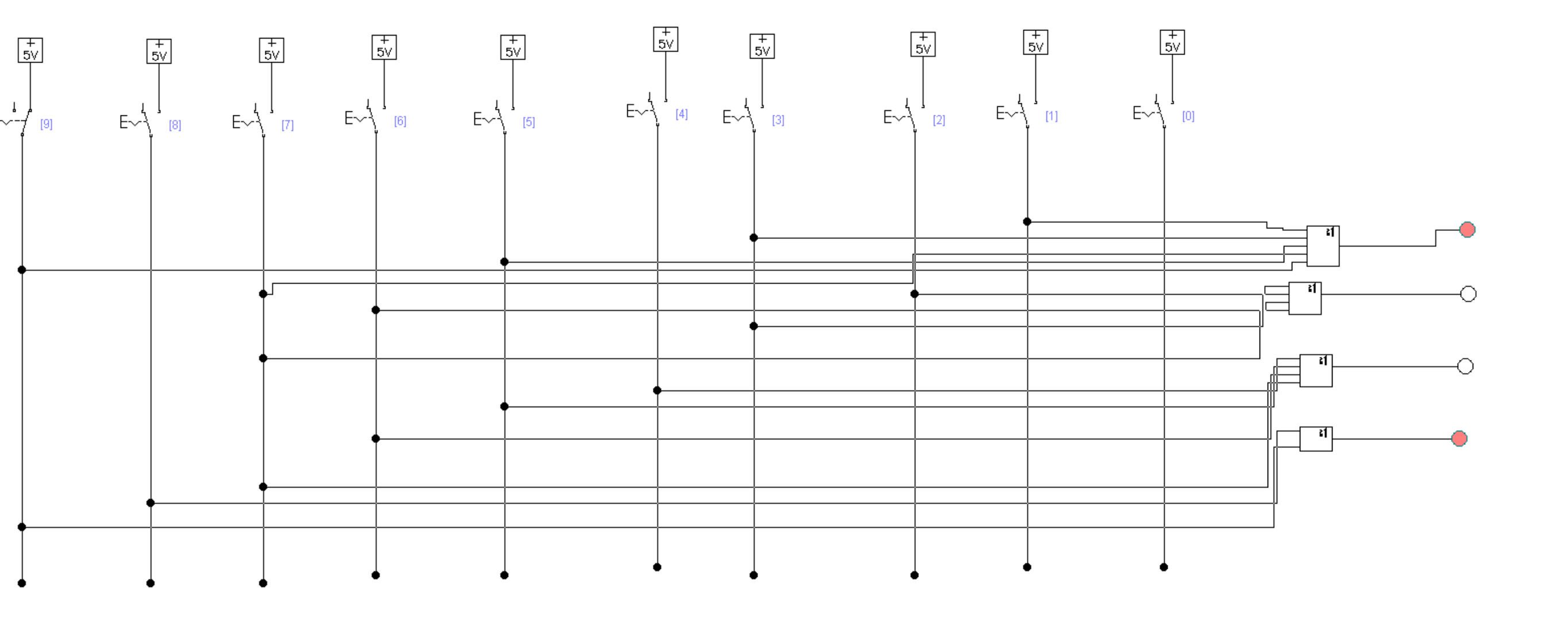
Если в работе дешифратора используется неполное число выходов, то такое дешифратор называется неполным. Например, дешифратор, имеющий 4 входа и 16 выходов будет полным, а если бы выходов было только 10, то он являлся бы неполным.

Пример полного дешифратора



Дешифратор имеет 3 входа и 8 выходов. Если убрать из данного дешифратора хотя бы 1 выход, то он станет неполным.

# **13 (37). Шифратор на 10 входов. Принцип синтеза схемы шифратора.**

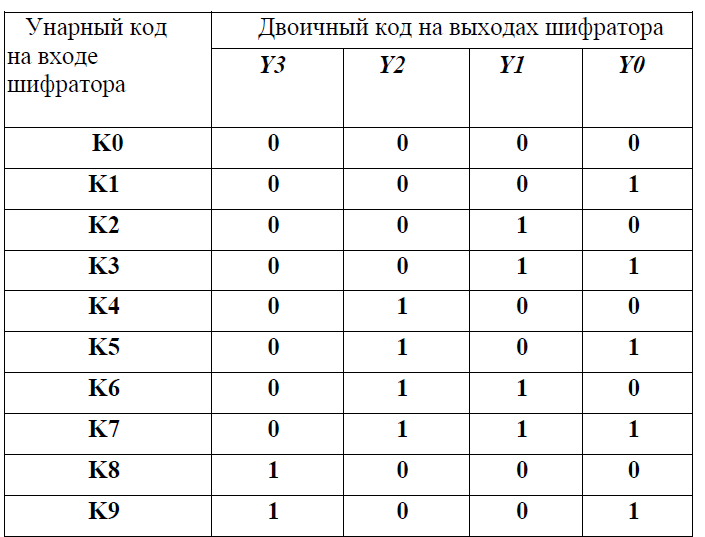
****

Принцип синтеза схемы шифратора:

Построим таблицу соответствия унарного кода на входе шифратора (К0…..К9) двоичным значениям его четырех выходов Y0…Y3

(Предполагается, что сигнал, соответствующий логической единице, в каждый момент времени подается только на один вход.)

Число входов и выходов в полном шифраторе связано соотношением n = , где n — число входов, m — число выходов.



Используя данную таблицу соответствия, запишем логические выражения, включая в логическую сумму те входные переменные, которые соответствуют единице некоторой выходной переменной. Так, на выходе Y0, будет логическая «1» тогда, когда логическая «1» будет или на входе K1, или K3, или K5, или K7, или K9, т. е.:

Y0= K1 + K3 + K5 + K7 + K9,

Y1 = K2 + K3 + K6 + K7,

Y2 = K4 + K5 + K6 + K7,

Y3 = K8+ K9.

По данным выражениям строится схема шифратора 10х4.

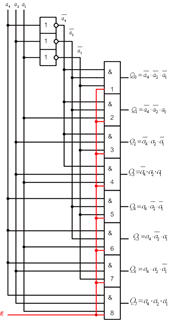
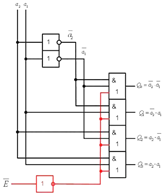
# **14 (6). Пример дешифратора с разрешающим входом.**

*Дешифратором* называется комбинационная схема с несколькими входами и выходами, преобразующая код, подаваемый на входы, в сигнал на одном из выходов (так называемый унарный код).

Если на входы дешифратора подаются двоичные переменные, то на одном из выходов дешифратора вырабатывается сигнал 1, а на остальных выходах сохраняются нули.

В общем случае дешифратор с n входами имеет 2n выходов, так как n-разрядный код входного слова может принимать 2n различных значений и каждому из этих значений соответствует сигнал единицы на одном из выходов дешифратора. Число входов и выходов в так называемом полном дешифраторе связано соотношением m=2n, где n — число входов, а m — число выходов.

В схеме дешифратора может быть организован дополнительный вход для сигнала разрешения Е. При Е=0 дешифратор не работает даже при наличии кода на его входах, а при Е=1 работает как обычный линейный дешифратор.



Функциональная схема трех-входного дешифратора с прямыми входами и выходами и разрешающим сигналом

Функциональная схема дешифратора с прямыми входами и выходами и инверсным разрешающим сигналом.

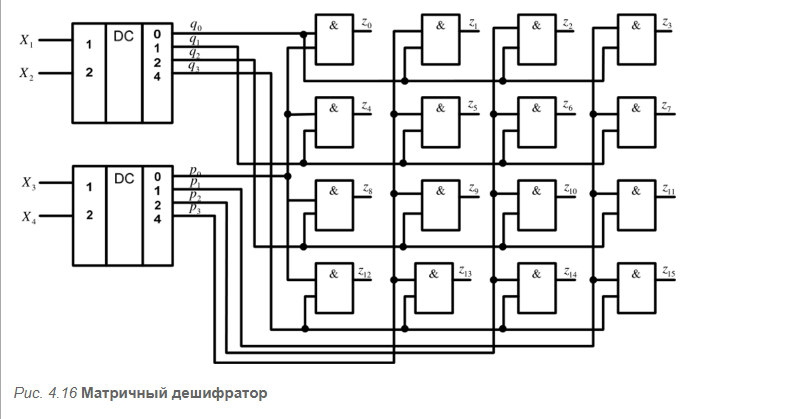
# **15 (55). Каскадный дешифратор. Преимущества по сравнению с линейным дешифратором.**

Из логических элементов, являющихся дешифраторами, можно строить *дешифраторы на большее число входов*, при этом, как правило, используются дешифраторы с дополнительными входами сигнала разрешения. *Каскадное включение* таких схем позволяет легко наращивать число дешифрируемых переменных.

# **16 (46). Матричный дешифратор, схема построения и преимущество перед линейным дешифратором.**

- Число входов равно числу разрядов двоичного числа N.

- Число выходов определяется количеством двоичных чисел этого разряда. M <= 2^N, M – число выходов



*Применение дешифраторов.*

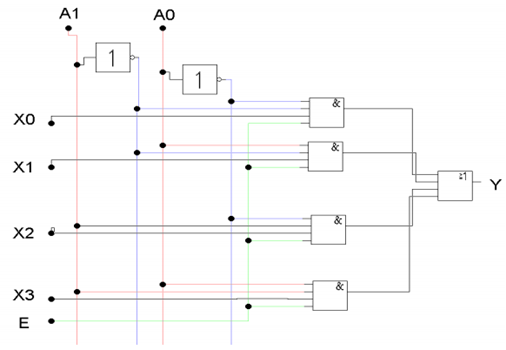
Дешифраторы применяются для преобразования различных кодов для отображения необходимой информации на индикаторах.

*Преимущество матричных дешифраторов.*

У линейных дешифраторов число используемых ЛЭ (логических элементов) равно числу выходов и с увеличением разрядности кода возрастает. В то время достоинством матричных дешифраторов является возможность использования для их построения ЛЭ с малым числом входов. Их выходы переплетаются сеткой. В узел сетки ставится И. Число выходов матричного дешифратора равно произведению числа выходов составляющих его дешифраторов.

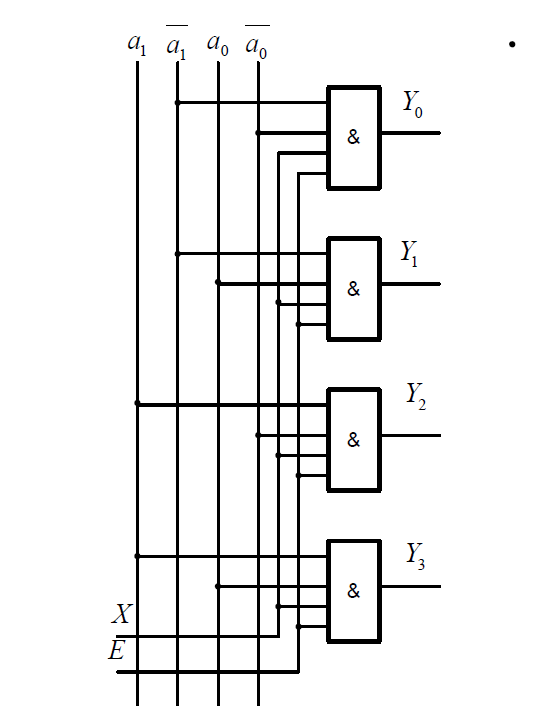
Матричный дешифратор наиболее эффективное решение по числу выходов если число дешифраторов ограничено, но требуется много элементов И, в отличие от каскадной схемы.

# **17 (18). Определение и пример реализации мультиплексора.**

**Мультиплексоры** осуществляют подключение одного из входных каналов к единственному выходному каналу под управлением управляющего (адресующего) слова. Разрядности каналов могут быть различными, мультиплексоры для коммутации многоразрядных слов составляются из одноразрядных. В цифровой технике мультиплексор имеет m информационных входов данных X1, Х2, Х3, ….Хm, n адресных входов и один выход данных. Зависимость количества информационных входов от количества разрядов адреса определяется соотношением m=2n. Двоичный код на входах адреса определяет номер того входа данных, с которого информация проходит на выход мультиплексора в этот момент.  


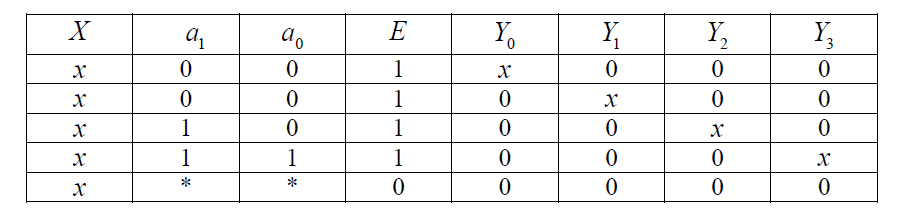
# **18 (42). Схема демультиплексора. Пояснить принцип работы.**

Схема демультиплексора с четырьмя выходами (Y) и двумя разрядами адреса (a) и одним информационным входом (X).



Демультиплексор позволяет подключать один информационный вход к нескольким выходам.

Работа демультиплексора соответствует таблице истинности





Значения сигналов на выходе демультиплексора будет определяться из следующих выражений:

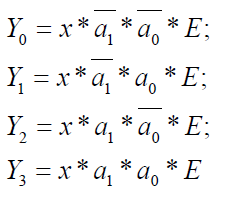
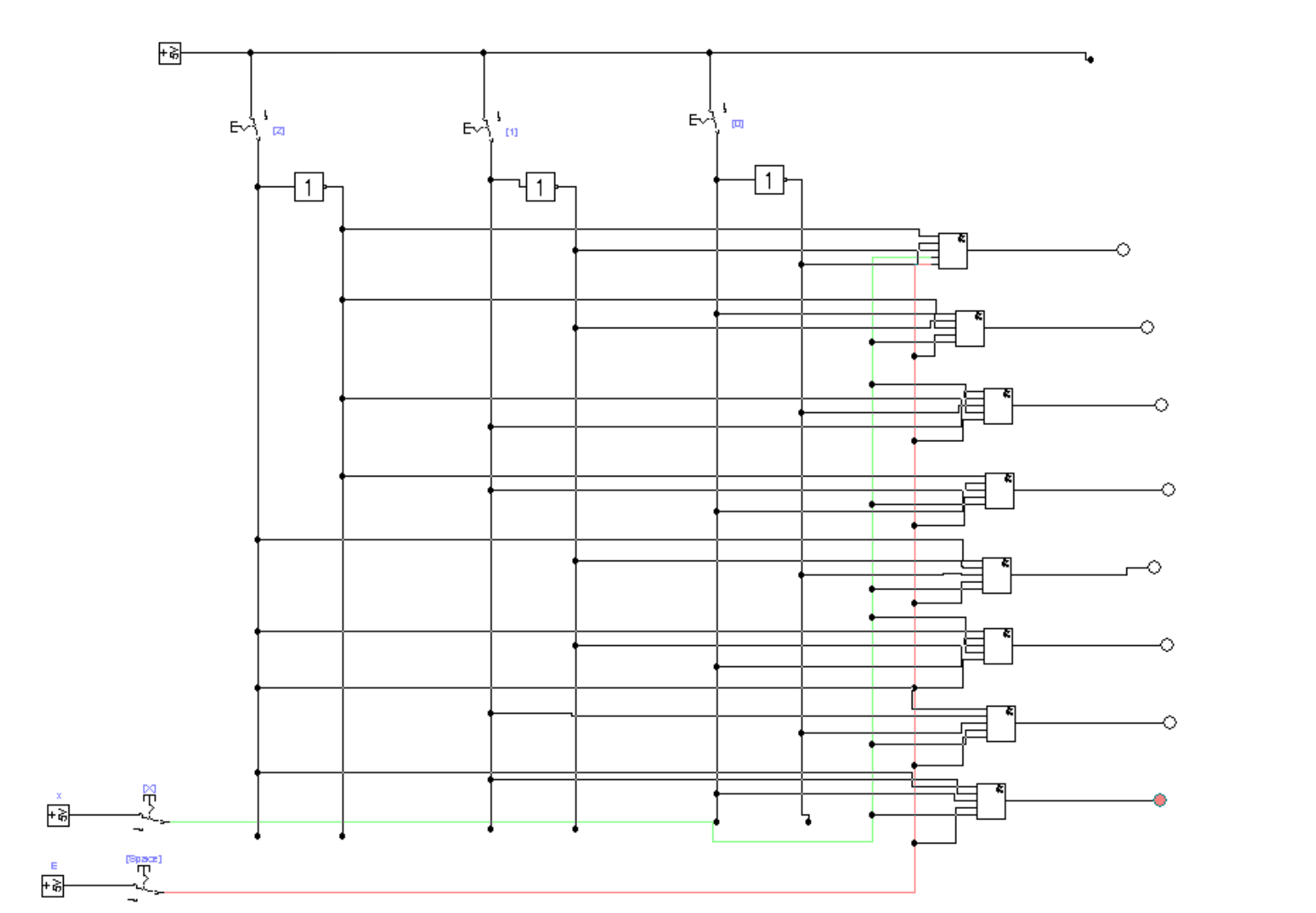


Схема в wb:



ТРИГГЕРЫ

# **19 (10). Назначение и классификация триггеров.**

Триггером называют ЦИФРОВОЙ АВТОМАТ (зап. Ячейка + комбинационная схема управления) с положительной обратной связью, имеющую два устойчивых состояния. Триггеры предназначены для хранения значения одной логической переменной (или значения одноразрядного двоичного числа, при хранении многоразрядных двоичных чисел для запоминания каждого разряда числа используется отдельный триггер).

Классификация триггеров:

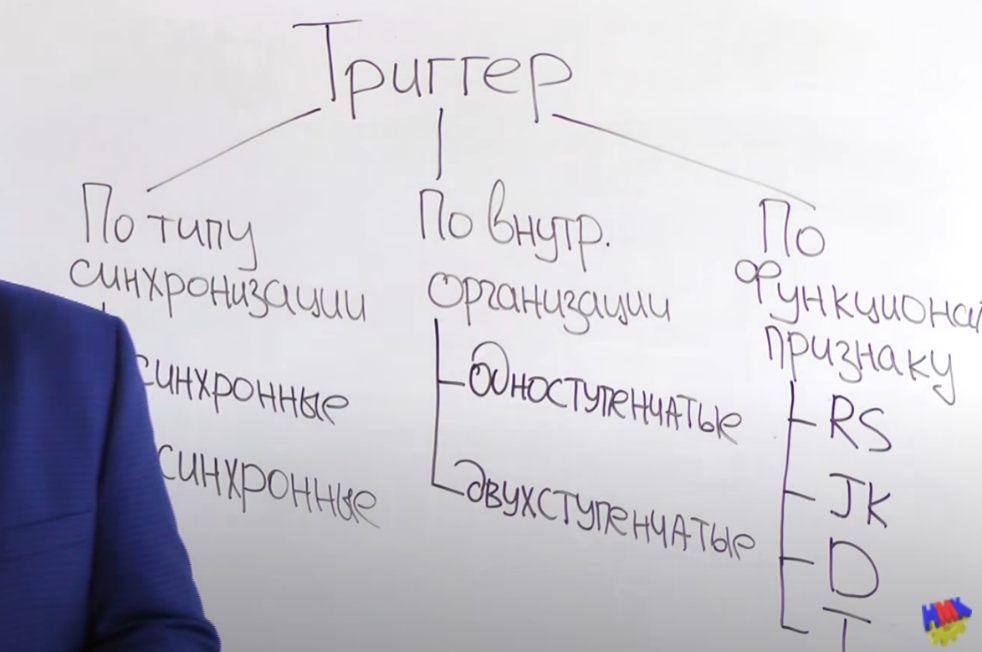
**1.** По функциональному признаку (этот признак определяет назначение триггера и в ряде случаев является решающим при выборе типа триггера для проектируемого вычислительного устройства или узла) различают RS, D, JK, T и другие типы триггеров.

**2.** По способу записи информации в триггер (этот признак характеризует способ записи информации и временную диаграмму работы триггера, т.е. определяет ход процесса записи информации в триггер) различают асинхронные и синхронные триггеры.

Запись информации в асинхронный триггер осуществляется в произвольный момент времени непосредственно с поступлением информационного сигнала на один из установочных входов триггера.

Синхронные триггеры помимо информационных входов содержат один или несколько синхронизирующих входов. Запись информации в такие триггеры осуществляется только при подаче синхронизирующего импульса (С). Синхронные триггеры подразделяются на триггеры, работающие по уровню С – потенциальные триггеры и триггеры динамического типа, срабатывающие по началу или концу синхроимпульса.

**3.** одно-двух-ступенчатые

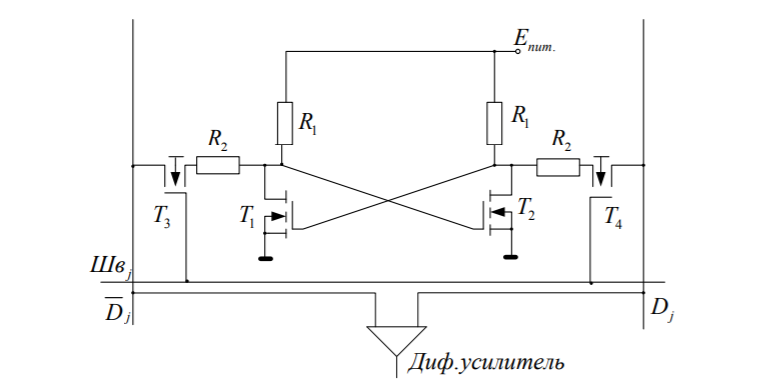


# **20 (16). Реализация RS-триггера на элементах Шеффера и Пирса.**

На элементах Шеффера:  
  
На элементах Пирса:  


# **21 (54). Принцип работы асинхронного RS-триггера, на примере схемы на четырёх биполярных транзисторах.**

Схемотехническую основу ОЗУ статического типа составляют регистры на основе триггеров. Современные схемы ОЗУ строятся на полевых транзисторах. Среди транзисторных технологий наиболее распространённой стала схема триггера, выполненная по К-МОП технологии. На рисунке представлена запоминающая ячейка (RS-триггер) для хранения одного бита информации на основе n-МОП транзисторов.



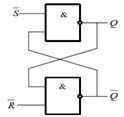
На поступает высокий потенциал от дешифратора адреса для выборки строки памяти, который открывает вспомогательные транзисторы T3 и T4 у всех триггеров, подключённых к данной шине выборки. Через столбцовые шины выборки и и вспомогательный дифференциальный усилитель можно считывать состояние запоминающей ячейки (RS – триггера). Столбцовые шины позволяют через них установить состояние триггера. Установочный сигнал снижает стоковое напряжение запертого основного транзистора хранения, что приводит к переводу противоположного транзистора в запертое состояние. Резисторы R1 служат для уменьшения ёмкостных токов. Вместо сопротивления R1, как правило, используется n-МОП транзистор с нулевым напряжением затвора. Дифференциальный усилитель необходим из-за режима очень малых токов через столбцовые шины.

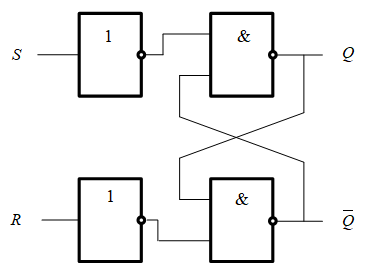
# **22 (1). Различия в схемах построения и работе асинхронных и синхронных RS-триггеров.**

Запись информации в *асинхронный триггер* осуществляется в произвольный момент времени непосредственно с поступлением информационного сигнала на один из установочных входов триггера.

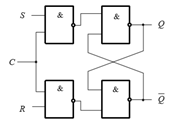
*Синхронные триггеры* помимо информационных входов содержат один или несколько *синхронизирующих входов*. Запись информации в такие триггеры осуществляется только при *подаче синхронизирующего импульса* (*С*).

*Асинхронным триггером RS-типа* (RS -триггером) называется логическое устройство с двумя устойчивыми состояниями, имеющее два информационных входа R и S, такие что, при S = 1 и R = 0 триггер принимает состояние 1 (Q =1), а при R = 1 и S = 0 триггер принимает состояние 0 (Q = 0). При значениях Rt=St=1 состояние триггера является неопределенным.

****Асинхронный R-S триггер с инверсными входами на элементах «И-НЕ».



Асинхронный RS-триггер с прямыми входами на элементах «И-НЕ».

В отличие от асинхронного, ***синхронный*** ***триггер RS-типа*** на каждом информационном входе имеет дополнительные схемы совпадения, первые входы которых объединены и на них подаются синхронизирующие сигналы. Вторые входы схем совпадения являются информационными. Таким образом, наличие схем совпадения определяет то обстоятельство, что триггер будет срабатывать от сигналов R и S только при наличии синхронизирующего импульса.

Синхронный RS-триггер с прямыми

входами на элементах «И-НЕ»

# **23 (11). Принцип построения MS-триггеров (с задержкой) и особенности их работы.**

Схема двухступенчатого синхронного RS-триггера (MS-триггера)

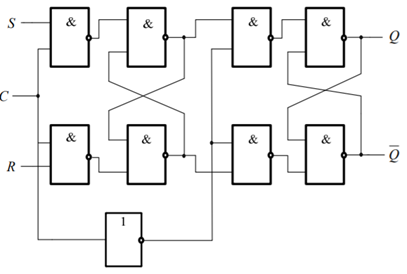
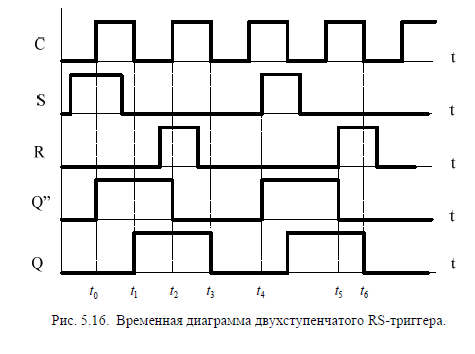
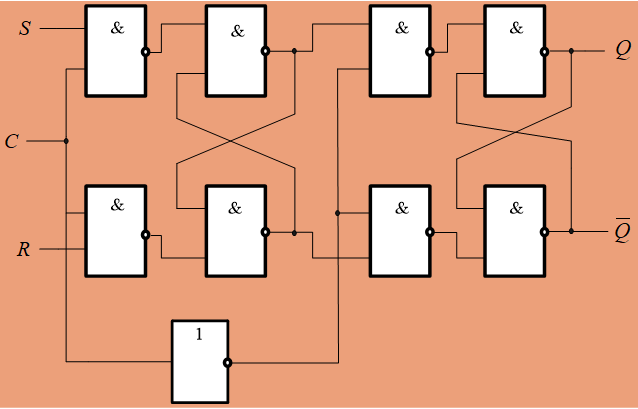


Схема двухступенчатого RS-триггера состоит из двух одноступенчатых RS-триггеров и инвертора в цепи синхронизации. При поступлении единичного синхросигнала входная информация заносится в первый одноступенчатый RS-триггер. Второй RS-триггер при этом будет хранить информацию, относящуюся к предыдущему такту. По окончании действия синхросигнала (С = 0), первый триггер перейдет в режим хранения, а второй перепишет из него новое значение выходного сигнала. Таким образом, в триггерах с внутренней задержкой новая информация на выходе устанавливается только после окончания действия синхронизирующего импульса, что является их характерной особенностью и позволяет строить произвольные схемы из таких триггеров, в том числе подавать сигналы с выхода триггера на его входы.

# **24 (21). Схема построения и принцип записи информации в триггерах типа MS (c задержкой).**

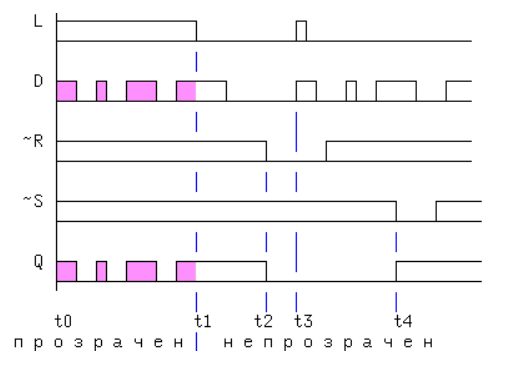
В триггерах с внутренней задержкой новая информация на выходе устанавливается только после окончания действия синхронизирующего импульса.

Подобный принцип обмена информацией реализован в ***двухступенчатых RS-триггерах с задержкой.*** Простейшая схема двухступенчатого RS-триггера с управляющим инвертором:

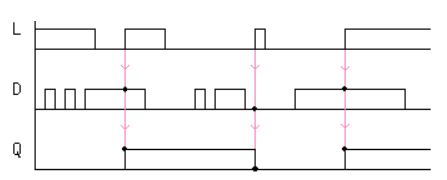
# **25 (25). Различие в работе триггеров потенциального и динамического типа. Изобразить на временных диаграммах.**

Особенность триггеров потенциального типа состоит в возможности многоразового изменения состояния триггеров при установленном в единицу сигнале синхроимпульса. В отличие от них синхронные триггеры динамического типа изменяют своё состояние только в момент изменения синхроимпульса из нуля в единицу, либо из единицы в ноль. При установившемся значении единичного синхроимпульса динамический триггер не меняет своё состояние при подаче новой комбинации установочных сигналов.

D-триггер потенциального типа:



D-триггер динамического типа:



Первый копируюет D если есть синхроимпульс (тут L). Второй – копирует только если L переключился из 0 в 1.

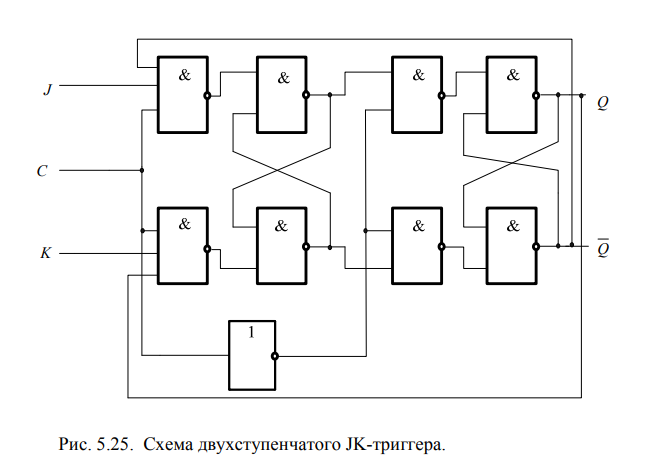
Временные диаграммы триггеров с динамическим входом существенно изменяются. Действие асинхронных входов такое же, как в D-триггере со статическим управлением, поэтому на временных диаграммах они не указаны

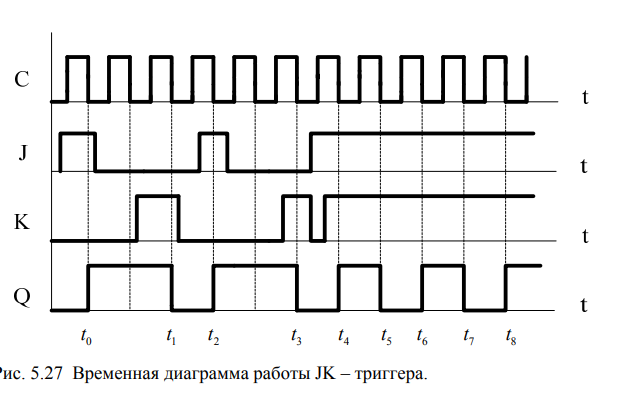
# **26 (49). Временная диаграмма, таблица переходов и принцип построения универсального JK - триггера.**

Универсальный JK-триггер. Триггером JK-типа называется логическое устройство с двумя устойчивыми состояниями и двумя информационными входами. Триггер функционирует в соответствии со следующей таблицей переходов

***ТАБЛИЦА ПЕРЕХОДОВ***

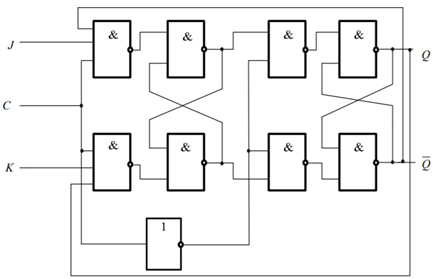
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ct | Jt | Kt | Qt | Qt+1 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 0 |

***СХЕМА ВРЕМЕННАЯ ДИАГРАММА***

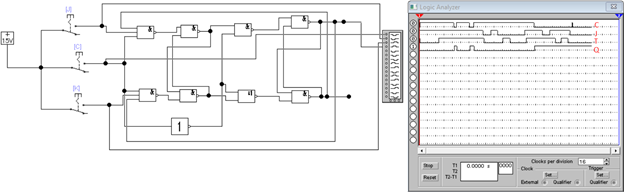


# **27 (8). Схема построения универсального J-K- триггера и его временная диаграмма работы.**

Схема двухступенчатого JK-триггера

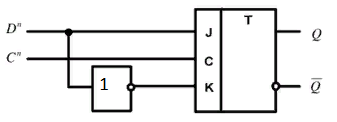


Соберем в Electronic Workbench и снимем временную диаграмму работы.



# **28 (3). Построение D-триггера на основе универсального JK-триггера.**

D-триггер имеет один информационный вход D. Для преобразования схемы JK- триггера в схему D- триггера информационный сигнал D подаётся непосредственно на вход J и одновременно через инвертор на вход K.

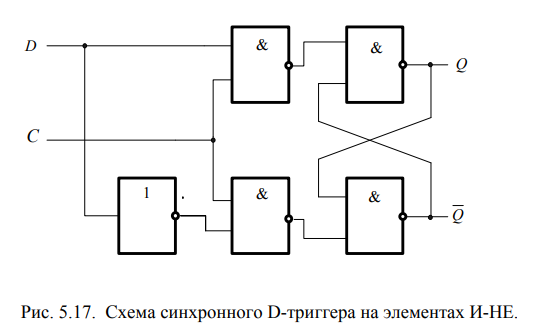


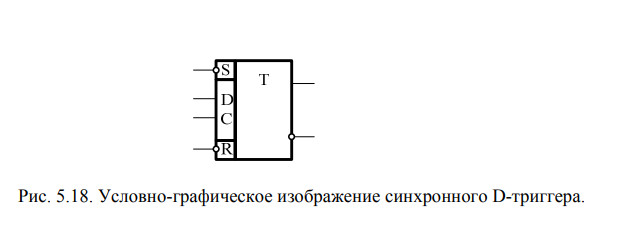
# **29 (45). Таблица переходов, временная диаграмма и схема построения синхронного D-триггера на элементах И-НЕ.**

Триггером D-типа называется логическое устройство с двумя устойчивыми состояниями и одним информационным входом D и входом синхроимпульса С.

***ТАБЛИЦА ПЕРЕХОДОВ***

|  |  |  |  |
| --- | --- | --- | --- |
| Сt | Dt | Qt | Qt+1 |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |



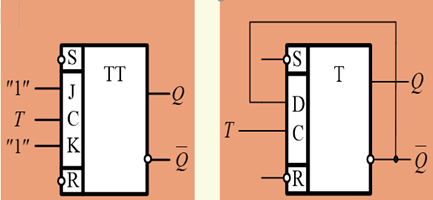


# **30 (9). Назначение установочных RS-входов в счётном Т-триггере.**

Установочные RS-входы приоритетны. Они устанавливают T-триггер (обнуляют или устанавливают единицу) независимо от входного сигнала T. Данные входы могут использоваться, например, для ограничения счетчика по модулю счета.

# **31 (15,58,60). Варианты реализации счётного T-триггера. Назначение асинхронных S и R входов.**

**Ответ 1)** Построить T-триггер можно на основе JK-триггеров и D-триггеров. Схема Т-триггера на основе D-триггера работает по начальному (переднему) фронту синхросигнала, а схема Т-триггера на основе JK-триггера по отрицательному (заднему) фронту синхросигнала.



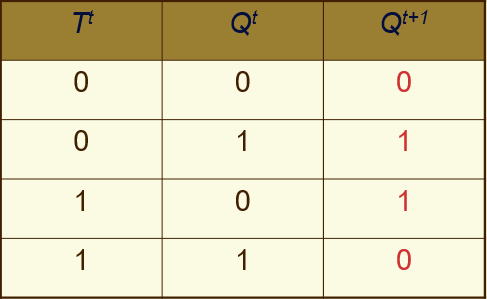
Вход R и вход S служат для установки или сброса значения триггера независимо от входа T, что позволяет управлять счетом триггера.

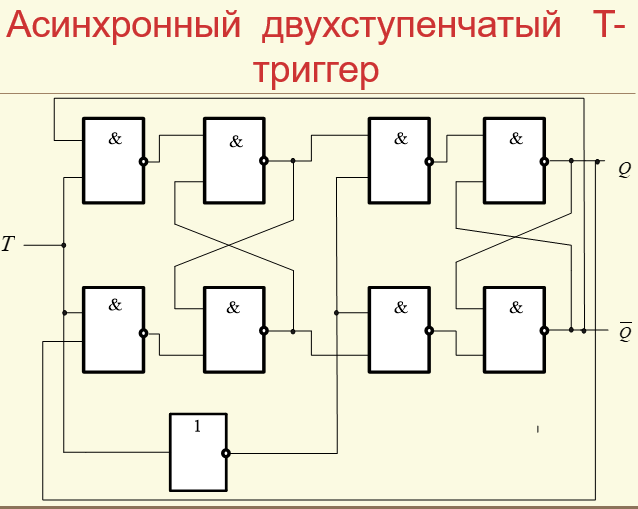
**Ответ 2)** **T-триггер** (Триггер со счетным входом)

Это логическое устройство с двумя устойчивыми состояниями и с одним информационным T-входом.

T-триггер меняет свое состояние на противоположное всякий раз, когда на T-вход поступает управляющий (счетный) сигнал.

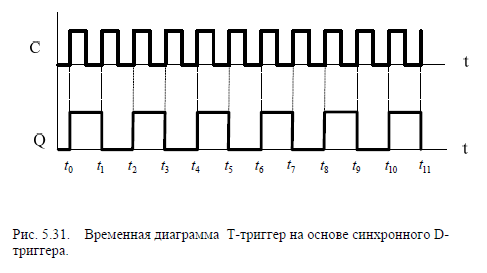
Асинхронный T-триггер функционирует в соответствии со следующей таблицей переходов:

****

****

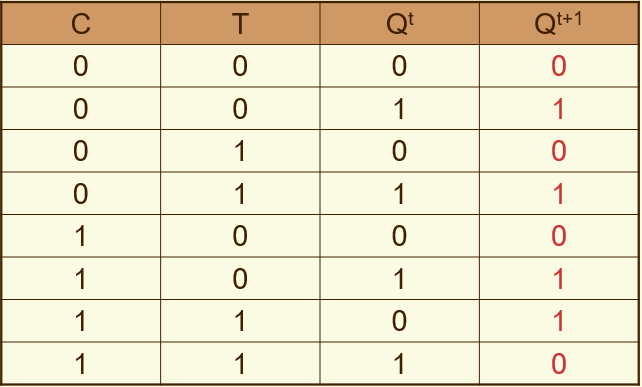
**Синхронных Т-триггеров не бывает.** Сигнал синхроимпульса заменен на Т. Эта схема то же что и JK, где J = K = 1 убрали из схемы за ненадобностью.

**На D-триггере одноступенчатый, на JK – двухступенчатый.** Первый срабатывает на фронт волны, второй на спад.

****

****

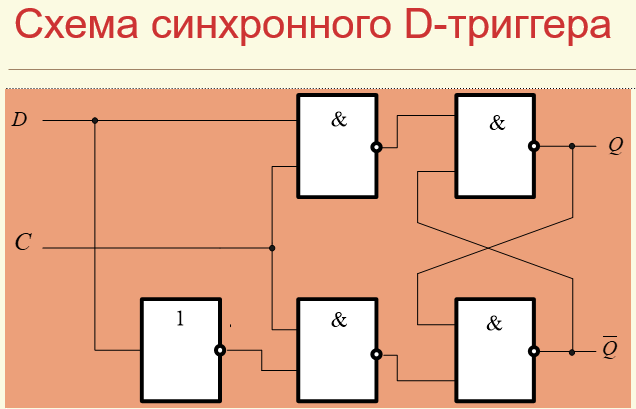
Синхронный T-триггер



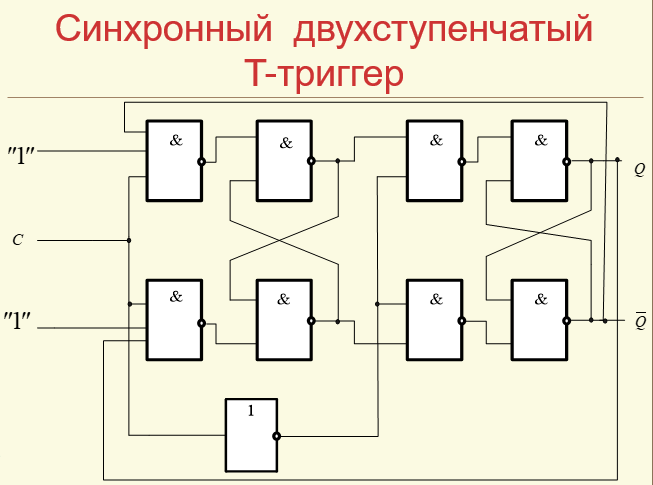
**Построить T-триггер можно на основе JK-триггеров и D-триггеров.**

Схема на основе D-триггера работает по начальному (переднему) фронту синхросигнала, а на основе JK-триггера по отрицательному (заднему) фронту синхросигнала.

Для получения счетного триггера из D-триггера необходимо соединить инверсный выход триггера с его D-входом.



Для получения счетного триггера из JK-триггера следует на входы J и K подать уровень логической единицы.

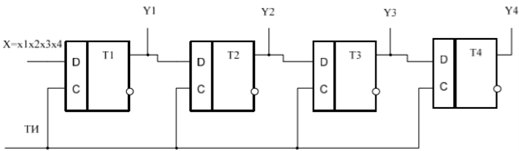


Асинхронные входы S и R необходимы для предварительной установки триггера в определенное состояние (логической 1 или 0).

РЕГИСТРЫ

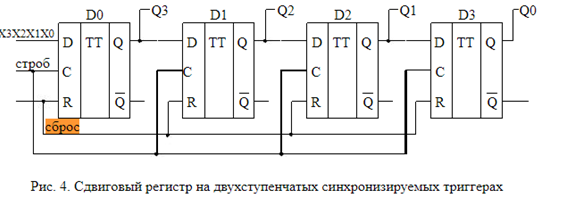
# **32 (0). Организация регистра сдвига и схема сдвигающего регистра на D-триггерах.**

В регистрах этого типа осуществляется сдвиг слова влево или вправо на заданное число разрядов. 3а один такт происходит сдвиг на один разряд. Применяются эти регистры в основном для преобразования параллельного кода в последовательный и наоборот, а также в арифметических устройствах при выполнении операций над специальными кодами. Сдвигающие регистры выполняются на D-триггерах. Количество триггеров в регистре зависит от разрядности преобразуемого машинного слова. На рисунке ниже изображен регистр сдвига вправо.

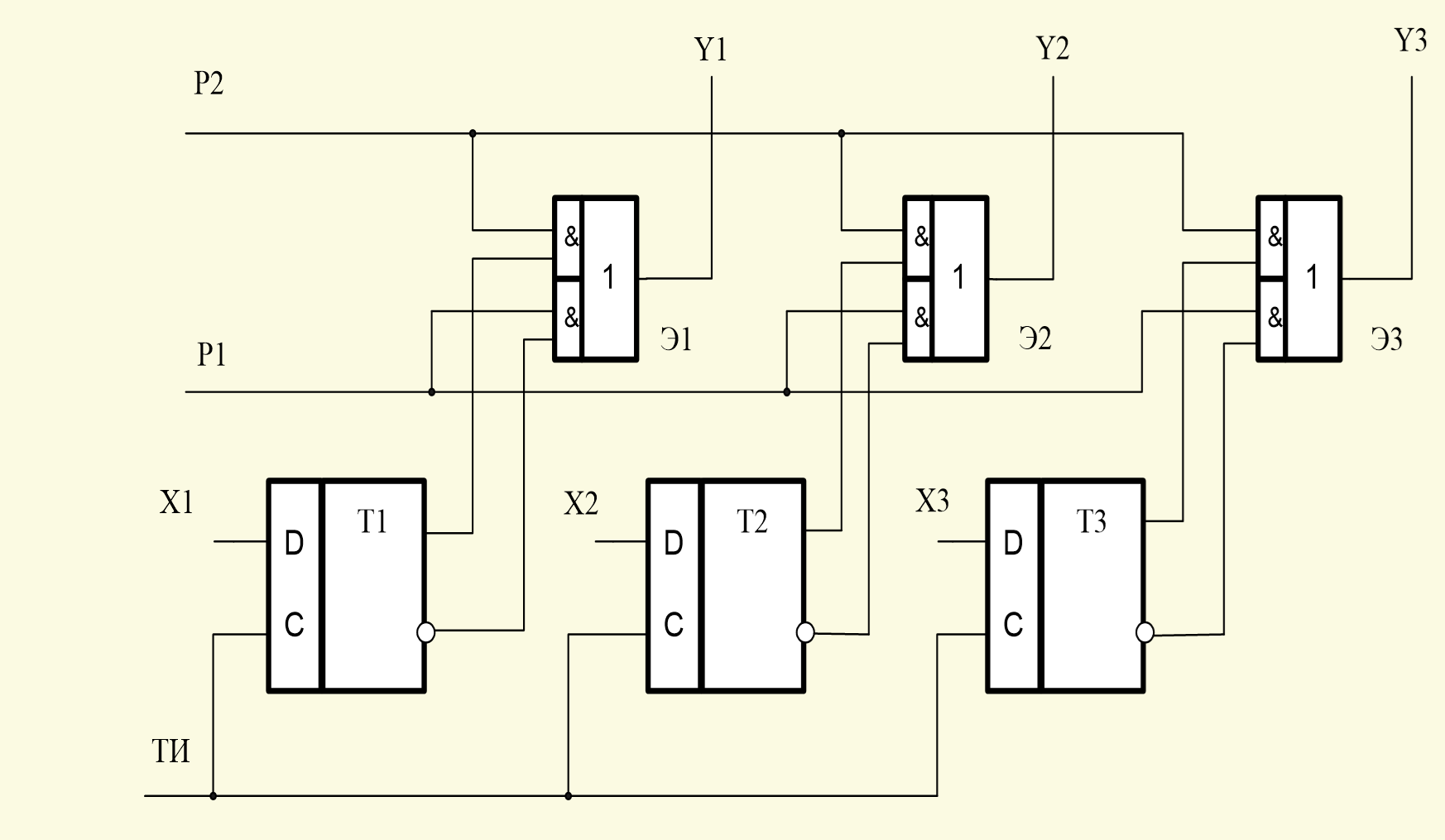
  
Число *X=x1x2x3x4* в последовательном коде поступает на вход D-триггера Т1 младшим разрядом вперёд. При поступлении 1-го ТИ x4 записывается в T1. По второму ТИ x4 переписывается в T2, а в T1 записывается x3 и т.д. Таким образом, после трёх тактирующих импульсов всё число окажется записанным в регистр и может быть считано в параллельном коде с выходов *Y1, Y2, Y3, Y4.*

# **33 (20). Варианты обнуления содержимого регистра сдвига.**

1. Подать на вход регистра некоторое количество нулей, не меньшее числа разрядов регистра.

2. Вход R обеспечивает сброс всех разрядов регистра. (В зависимости от конструкции триггера он может быть инвертированным)  


# **34 (39). Пример построения регистра с параллельным занесением кода.**

****

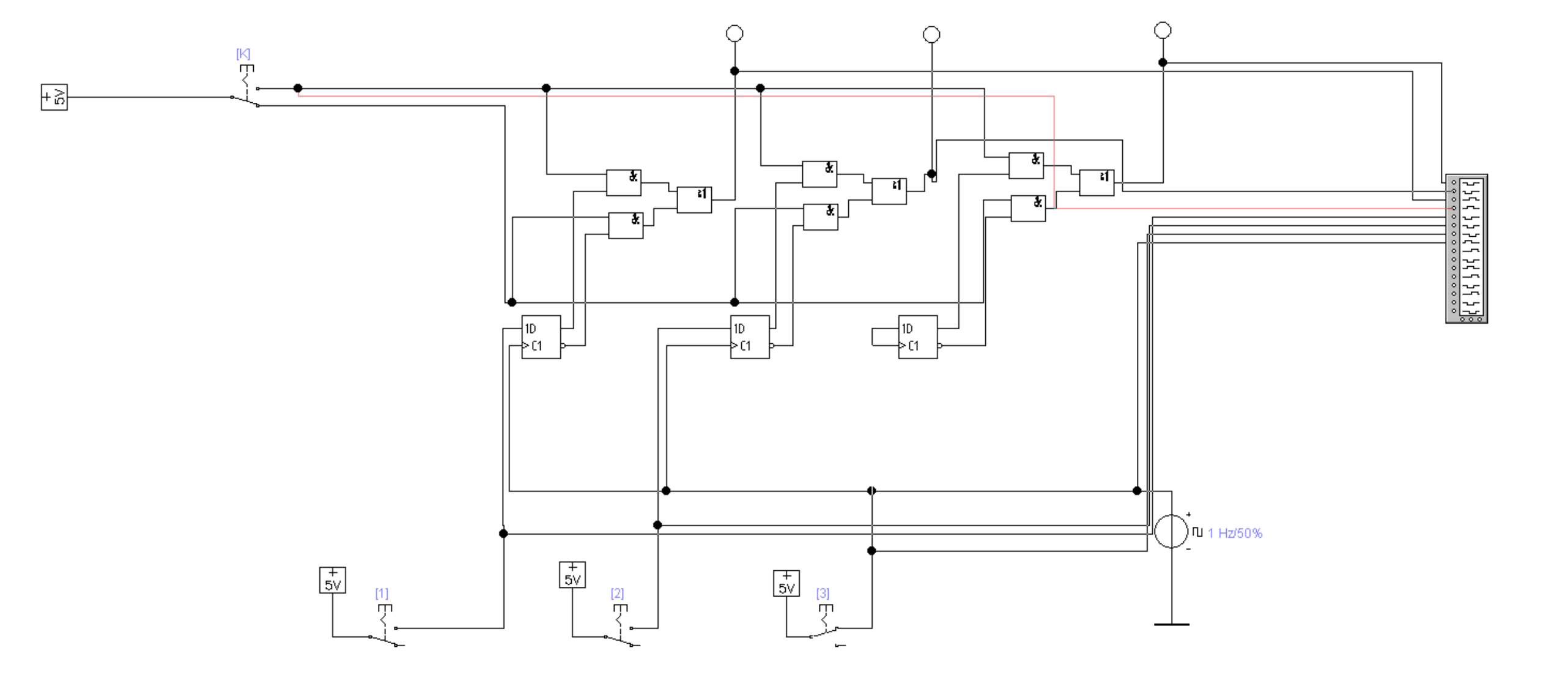
Эта схема еще и определяет формат выдачи кода. P1 = 1 – выдача в обратном коде. P2 = 1 – в прямом

Регистр выполнен на D-триггерах (T1-T3). Запись информации осуществляется по входам D в соответствии с уравнением D-триггера:



Входное слово X=x3x2x1 поступает на входы D- триггеров и при ТИ=1 записывается в регистр.

Схема в wb:



# **35 (36). Схема универсального регистра и принцип управления переключения режимов работы.**

1 переключатель – режим регистра параллельного занесения

(переключатели 0, 9, 8 подают сигналы на разряды)

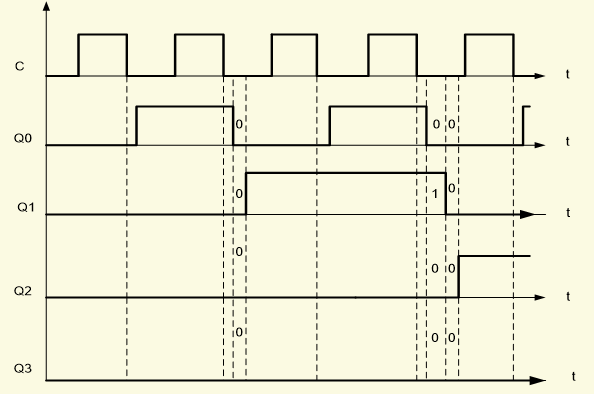
2 переключатель – режим сдвигающего регистра влево (переключатель 4 подает сигнал на регистр)

3 переключатель – режим сдвигающего регистра вправо (переключатель 5 подает сигнал на регистр)

СЧЕТЧИКИ

# **36 (7). Какой недостаток характерен для асинхронных счетчиков. Показать на примере временной диаграммы.**

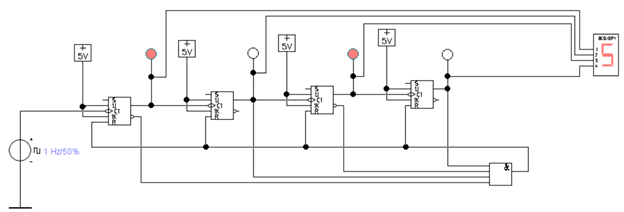
Основным недостатком асинхронных счетчиков является задержка, возникающая из-за конечного времени переключения счетного триггера. В представленной ниже временной диаграмме показано, что из-за накладки задержек от разряда к разряду образуются временные интервалы (короткие), в которых счетчик принимает неверные, не последовательные значения.



На данной временной диаграмме видно, что после кода 0001 до установления кода 0010 существует интервал времени с неверным кодом 0000. После перехода с кода 0011 присутствует перед правильным кодом 1000 еще два неверных кода: 0010 и 0000.

Указанный недостаток требует введения дополнительного стробирующего сигнала на разрешение считывания содержимого счетчика, который учитывает все указанные задержки переключения для всех разрядов.

# **37 (13). Пример организации счетчика по модулю десять. Перечислите варианты методов ограничения модуля счётчика.**



Ксч = 10, т.е. счетчик должен иметь 10 состояний: от 0 до 9 в десятичной системе и от 0000 до 1001 в двоичной системе. Определили какие разряды счетчика будут находится в единичном состоянии при записи в счетчик Ксч. Прямые выходы этих разрядов завели на входы логической схемы «И» и далее в цепь установки «0». Таким образом, при достижении счетчиком значения Ксч он автоматически возвращается в состояние 0000 и счет начинается сначала.

Варианты методов ограничения модуля счета: (Это из презентации)

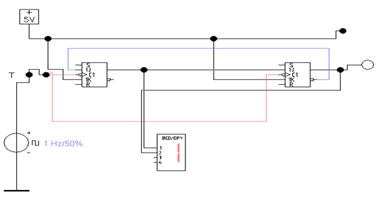
1. Работа счетчика ведется до достижения кода M-1. M – обозначение модуля счетчика.

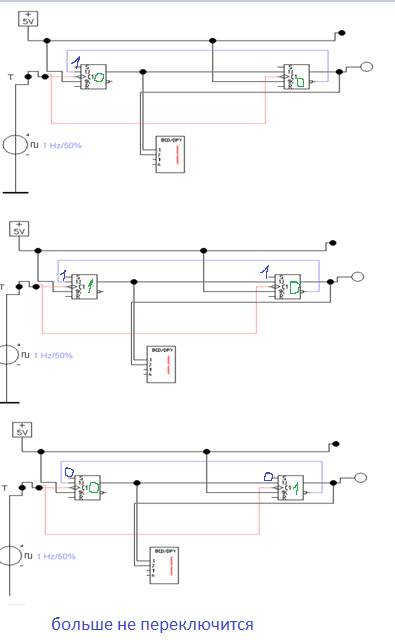
Это Счетчик с естественным порядком счета, у которого уменьшение числа устойчивых состояний достигается засчет сбрасывания счетчика в нулевое состояние при записи в него заданногочисла сигналов.

При обнаружении следующего по порядку значения логический элемент И или И-НЕ вырабатывает сигнал обнуления счетчика в исходное состояние счета. Схема этого способа показана выше. Минус данного способа состоит в том, что нет возможности управлять изменением модуля счетчика.

1. Для исключения излишних состояний счетчика в его схему вносятся дополнительные нестандартные межразрядные связи. Минус данного способа так же в том, что нет возможности изменения модуля счета.

JK – триггеры.



  
  
  
Это счетчик по модулю 3 из презентации. 3 = 2 + 1. Последний правый счетный триггер называется единичным триггером в данной схеме. Он нужен только для того, чтобы модуль счета был ограничен нечетным числом. Счетный сигнал подается на оба триггера (как в синхронном счетчике). Связь по передаче счетного сигнала из младшего разряда к следующему заведена только на вход J единичного триггера, а вход K единичного триггера подсоединен к 1. Организована дополнительная (синяя) обратная связь с инверсного выхода единичного триггера на вход J первого разряда для блокировки переключения при коде 10, т.е. состояния 1 в старшем (правом) разряде. (на циферблат не смотрим)

1. Двоичный счетчик перед началом счета по входам асинхронной установки (входам параллельной загрузки) загружается числа модуль до 2 в степени n (доп. код от модуля). В таком счетчике формируется либо схема обнаружения переполнения (всех 1), либо схема обнаружения всех 0 во всех разрядах. В этих двух случаях вырабатывается блокировка счетного сигнала и переход к повторной параллельной загрузке исходного кода.

Речь идет об ограничении счетчика снизу, т.е. счет ведется не от 0. При достижении 11…11 происходит загрузка по входам S нужного кода.

Достоинства данного способа:

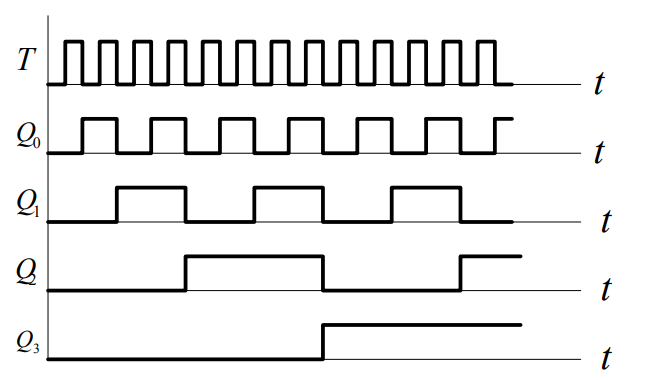
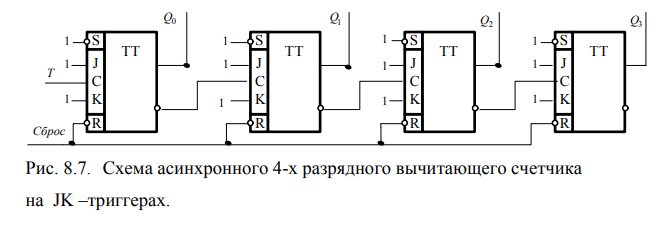
- типовая схема связей счетчика без дополнительных связей;

- использование имеющихся во всех счетчиках тракта параллельной загрузки;

- легкая смена значения нужного модуля счета

Недостаток состоит в том, что в таком счетчике неестественная последовательность получаемых кодов. Поэтому данный способ применяется, когда показания счетчика не важны, а используется сигнал переполнения, либо нулевого значения.

# **38 (44). Построение вычитающего асинхронного счетчика на J-K триггерах. Временная диаграмма работы счетчика.**

Вычитающий асинхронный счетчик на JK-триггерах можно получить, если инверсный выход предыдущего триггера соединить со входом синхроимпульса С последующего триггера. Схема такого счётчика представлена ниже

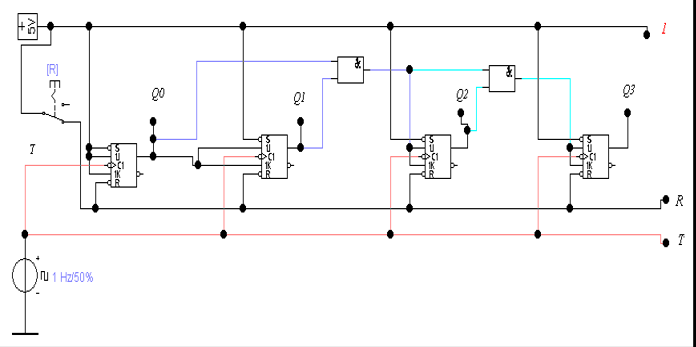
# **39 (62). Синхронный суммирующий счётчик на JK-триггерах. Принцип управления переключения разрядов.**

Особенностью счётчиков синхронного типа является подача счётного импульса одновременно на все разряды счётчика.

Управление правильностью переключения разрядов счётчика возлагается на логические элементы «И».

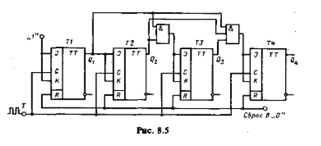
Логический элемент «И» включает счётный режим на данном разряде счётчика подачей **J=K=1**, если младшие разряды по отношению к данному приняли состояние единиц.

Это условие касается **суммирующего синхронного счётчика**.



# **4- (34). Принцип работы и схема построения синхронного счётчика на J-K триггерах.**

Синхронный счетчик на J-K триггерах основан на способности J-K триггера менять свое значение при поступлении синхроимпульса и 1 на J и K входах



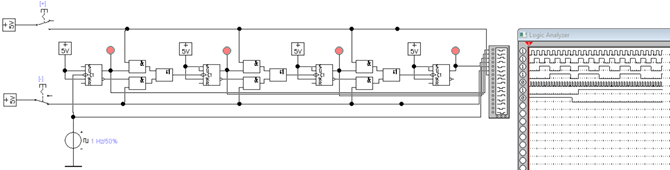
Поэтому

1. синхроимпульс подходит ко всем счетчикам одновременно

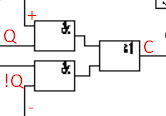
2. счетчик 1 разряда изменяет значение с частотой синхроимпульса

3. Счетчики более высоких разрядов изменяют значение только при наличии 1 на всех предыдущих , т.е. на его J-K входы подключено логическое И всех предыдущих выходов. Вход R служит для обнуления счетчика

# **41 (14). Схема реверсивного счетчика на JK-триггерах, пояснить принцип управления направлением счёта.**



В зависимости от выбранного режима (сложение или вычитание) на вход последующего триггера подается сигнал либо с инверсного (при суммировании), либо с прямого (при вычитании) выходов предыдущего триггера. Это реализуется с помощью такой схемы



это мультиплексор на 2 адреса:

# **42 (22). Счетчик Джонсона: схема построения и диаграмма работы.**

Это кольцевой счетчик, у которого одна из связей между триггерами сделана перекрестной, т.е. вход одного из триггеров соединен не с прямым, а с инверсным выходом предыдущего триггера (как в ленте Мёбиуса).

Непозиционная двоичная система.

000

001

011

111

110

100

000

Типа как бегущая строка

Для получения схемы счетчика Джонсона необходимо соединить

инверсный выход старшего разряда регистра с входом младшего.

Изменяя частоту синхроимпульсов, можно добиться разной скорости

изменения состояний.

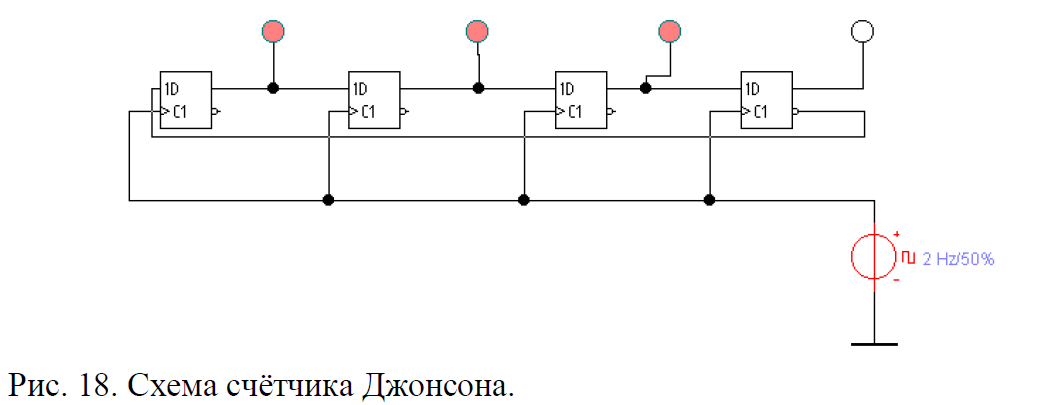
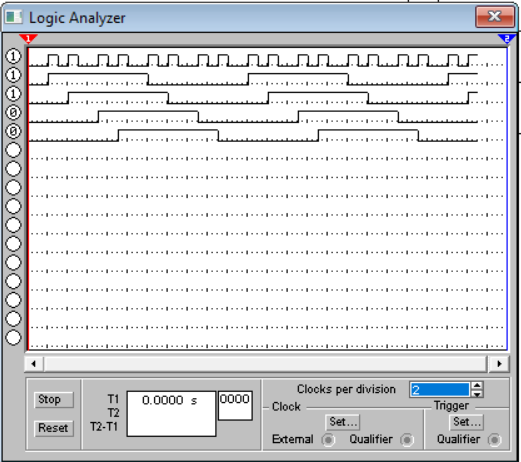
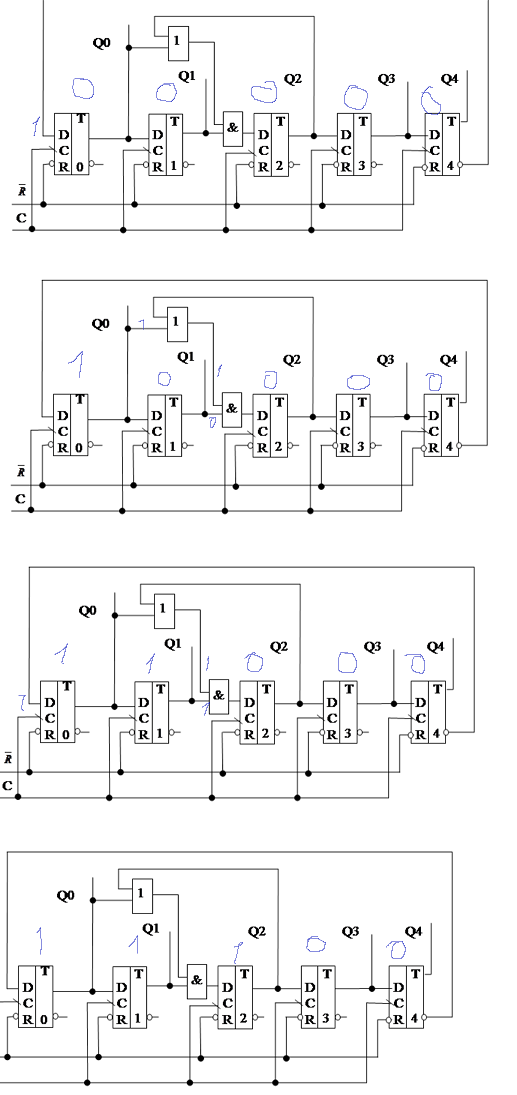


Диаграмма работы счётчика Джонсона:



в счетчике Джонсона по кольцу бежит волна единиц и нулей. Счетчик Джонсона тоже боится зацикливания ошибочных состояний, хотя и в меньшей степени, чем простой кольцевой счетчик. Для исправления ошибок, чтобы ошибка циркулировала в счетчике меньше половины кольца, в счетчик введены ИЛИ и И (см ниже), которые обеспечивают возвращение счетчика к правильной работе.

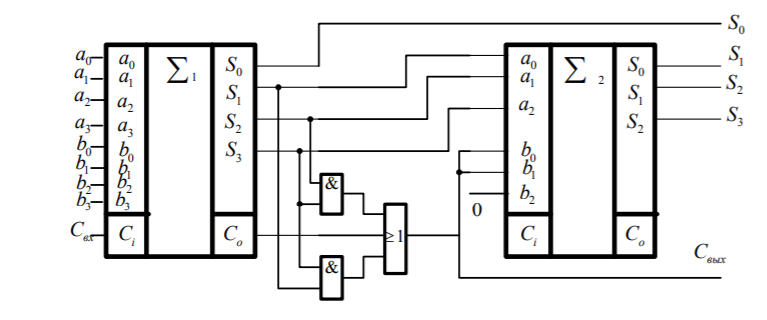
Число единичек, идущих последовательно, увеличивается от 0 до числа триггеров. Между ними не появляется нулей. Следующие поправки не допускают нахождение нулей между единицами центральных разрядов.



СУМ-РЫ, ВЫЧИТ-ЛИ, ИНКРЕМЕН-Ы, ДЕКРЕМЕН-РЫ, МНОЖ-Е БЛОКИ, КОМПАРАТОРЫ

# **43 (51). Организация двоично-десятичного сумматора. Назначение сумматора коррекции.**

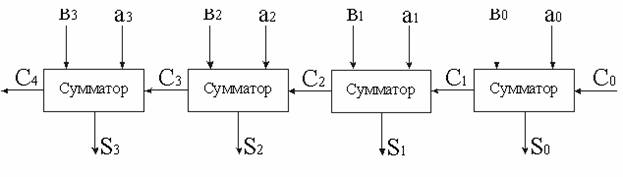
Двоично-десятичные сумматоры выполняют действия над десятичными числами, разряды которых заданы двоичными тетрадами. Способ построения двоично-десятичного сумматора основывается на суммировании двоичных тетрад аргументов и коррекцию результата на дополнительном сумматоре, если сумма двух двоичных тетрад дала результат, превышающий число 9. Если при сложении тетрад и входного переноса результат лежит в диапазоне от 10 до 15, необходимо организовать перенос в соседнюю тетраду с одновременным уменьшением полученного числа на 10. Вычитание числа 10 можно заменить сложением с дополнительным кодом числа 10, имеющим двоичное представление 0110. *Второй корректирующий сумматор в этом случае должен прибавить корректирующую поправку 0110 и обеспечить порождение переноса в старшую тетраду.* Схема двоично-десятичного сумматора представлена на рисунке:



Сумматор с индексом 1 формирует значения аргументов, которые могут лежать в диапазоне от 0 до 31. Для обнаружения сумм, превышающих число 9 введены два логических элемента «И», которые в этих комбинациях выдают логическую единицу, поступающую на логический элемент «ИЛИ». Выход элемента «ИЛИ» формирует выходной перенос в следующую тетраду. Этот же сигнал логической единицы порождает добавление кода 0110 (по входам b0 и b1). При результате сложения на первом сумматоре больше числа 15 выдаётся выходной перенос, который поступает на следующую тетраду.

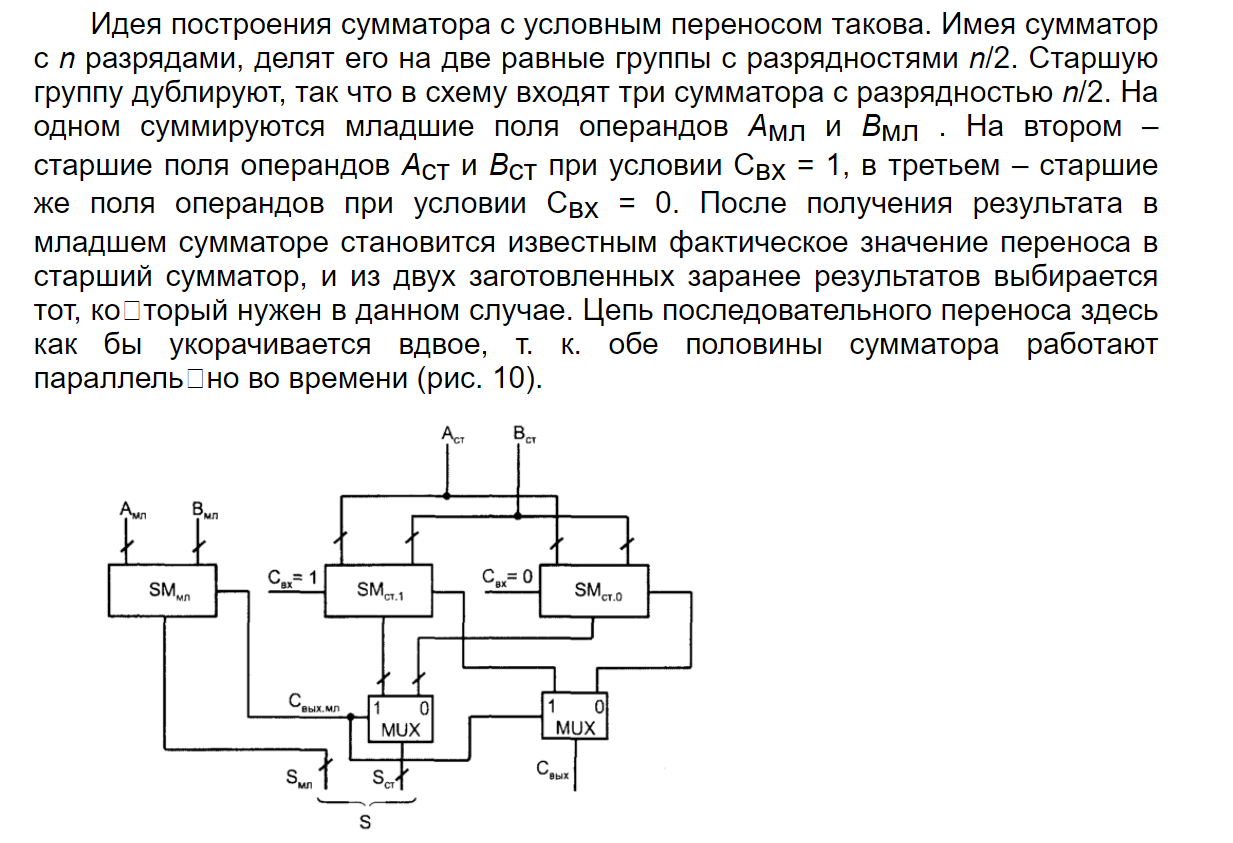
# **44 (52). Варианты реализации сумматоров с различными схемами организации переносов.**

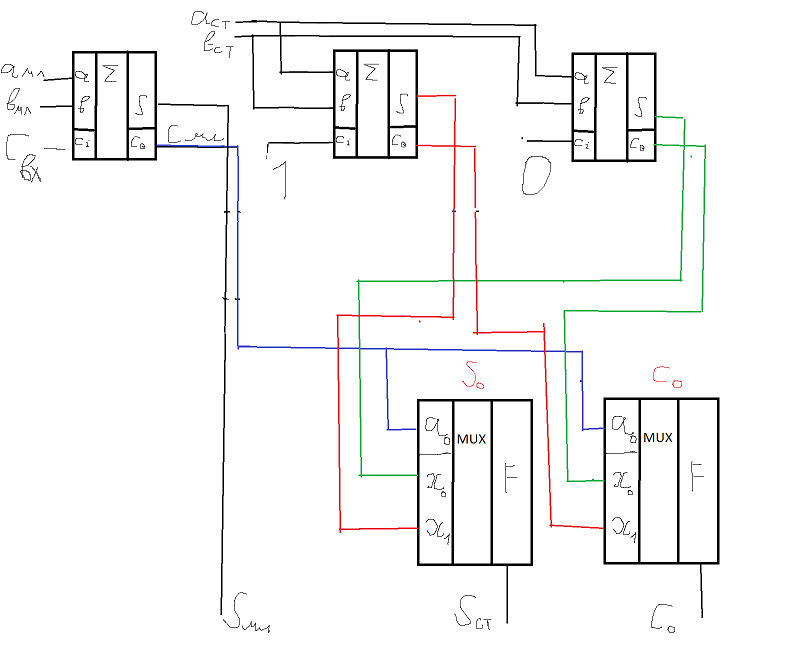
По способу организации межразрядных переносов делят на:

Сумматоры с ***последовательным переносом,***При подаче слагаемых цифры их разрядов поступают на соответствующие одноразрядные сумматоры. Каждый из одноразрядных сумматоров формирует на своих выходах цифру соответствующего разряда суммы и перенос, передаваемый на вход одноразрядного сумматора следующего (более старшего) разряда. Такая организация процесса организации переноса, называемая последовательным переносом, снижает быстродействие многоразрядных сумматоров, т.к. получение результата в старшем разряде сумматора обеспечивается только после завершения распространения переноса по всем разрядам  


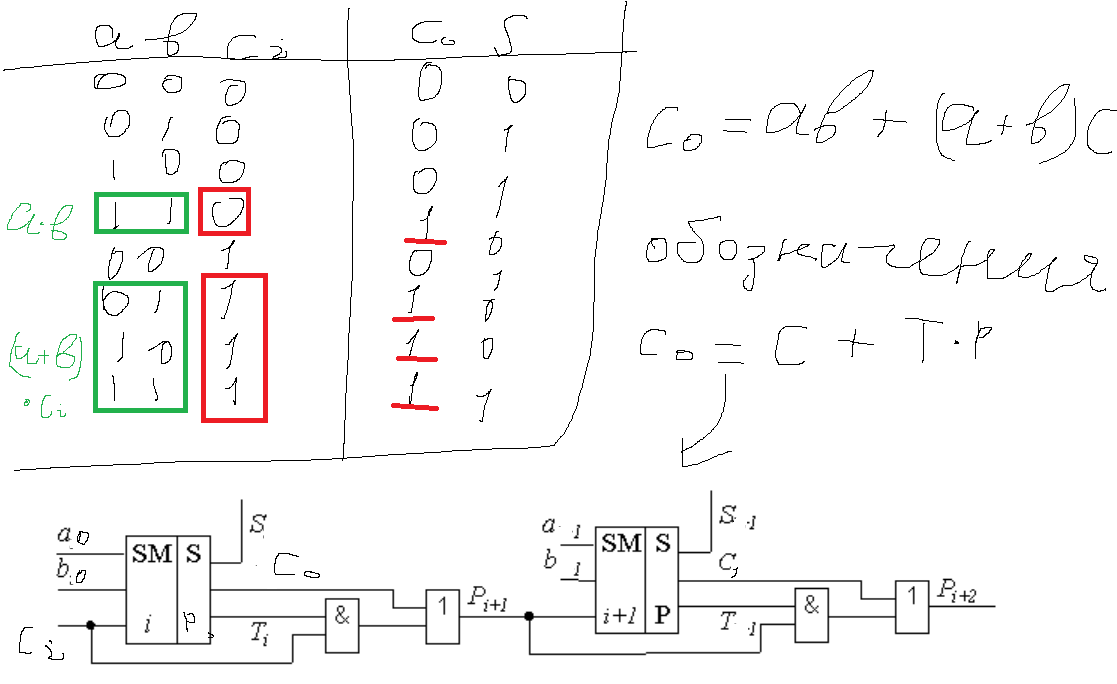
Сумматоры с ***параллельным переносом,***При этом все сигналы переноса вычисляются непосредственно по значениям входных переменных  


Сумматоры с ***условным переносом***,   
Сумматор разрядности n делят пополам на младшую и старшую группу разрядов. Старшую группу дублируют. В итоге сумматор состоит из трёх групп разрядов размерностью n/2. На одной старшей группе суммируются разряды при условии, что перенос из младшей группы отсутствовал, а на другой старшей, что перенос был. При завершении операции на младших разрядах становится ясным какой результат использовать на старших группах. Выбором старшей группы управляют мультиплексоры, управляемые переносом из младшей группы.





Сумматоры со ***сквозным переносом***



# **45 (40). Реализация многоразрядного сумматора последовательного действия.**

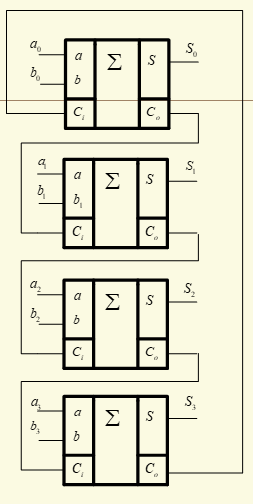
В последовательном сумматоре обработка чисел ведётся поочерёдно, разряд за разрядом на одном и том же оборудовании.

Состоит из одноразрядного сумматора, выход pi+1 которого соединен с входом pi через элемент задержки, параметры которого согласованы со скоростью поступления разрядов слагаемых на входы сумматора. Операция суммирования во всех разрядах слагаемых осуществляется с помощью одного и того же одноразрядного сумматора, но последовательно во времени, начиная с младших разрядов. Такое построение сумматора возможно за счет того, что слагаемые поступают в последовательной форме.



Очевидное достоинство сумматора последовательного действия заключается в малом объеме оборудования, требуемого для его построения. Однако связанная с этим необходимость в последовательной обработке разрядов приводит к крайне низкому быстродействию.

# **46 (63). Особенность структуры сумматора модифицированного обратного кода.**

Особенностью сумматора обратного кода является наличие связи по переносу между сумматором знаков и сумматором младших разрядов. Схема **сумматора** **обратного кода**:

При выполнении операций в дополнительном и обратном коде включая знаки чисел происходит нарушение правильного значения знака результата из-за переполнения разрядной сетки числовой части и поступления вызванного переполнением переноса в сумматор знаков.

Аппаратно определить данную ситуацию не представляется возможным. Для решения данной проблемы было предложено использовать модифицированный дополнительный и **модифицированный обратный код**.

Отличие состоит в дублировании битов, обозначающих код знаков слагаемых. Благодаря такому дублированию, возникающий перенос искажает только один из знаков результата в модифицированном коде.

Признаком переполнения является несовпадение знаков результата выполнения операции сложения. Это несовпадение легко обнаруживается при помощи логического элемента «Исключающее ИЛИ».

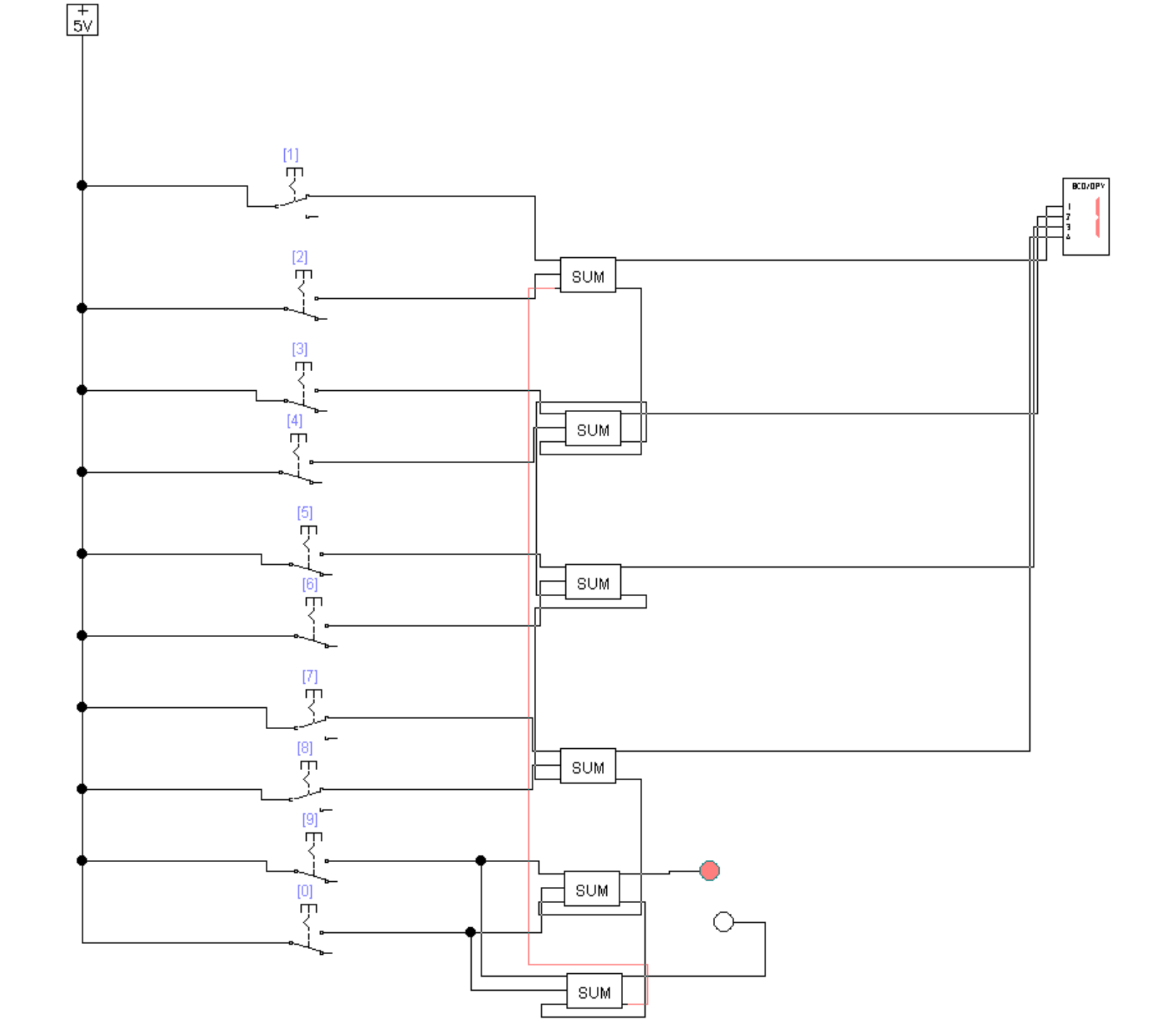
# **47 (38). Особенность структуры сумматора обратного модифицированного кода. Обнаружение переполнения. Пояснить примером.**

Особенностью структуры сумматора модифицированного обратного кода является наличие связи по переносу между сумматором знаков и сумматором младших разрядов, а также дублирование битов, обозначающих код знаков слагаемых. Благодаря такому дублированию, возникающий перенос искажает только один из знаков результата в модифицированном коде. Признаком переполнения является несовпадение знаков результата выполнения операции сложения. Это несовпадение легко обнаруживается при помощи логического элемента «Исключающее ИЛИ».

Пример:

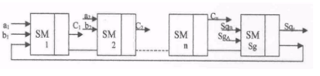
9 + 8 = 17 (00 1001 + 00 1000 = 01 0001). При сложении двух прямых кодов получились разные биты знаков, что свидетельствует о переполнении.

Схема:

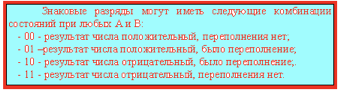


# **48 (32). Выполнение операции вычитания на сумматорах. Сумматор дополнительного модифицированного кода. Принцип построения.**

Чтобы выполнить операцию вычитания на сумматоре, необходимо перевести число или в дополнительный, или в обратный код. В дополнительном коде сумматор строится так же, как и в прямом, только старший выход переноса подключается к младшему входу переноса.



При суммировании модифицированного доп.кода под разряд отводится 2 знака, следовательно 2х разрядный сумматор знаков составляется так же, однако его результат будет свидетельствовать о верности проведенной операции:

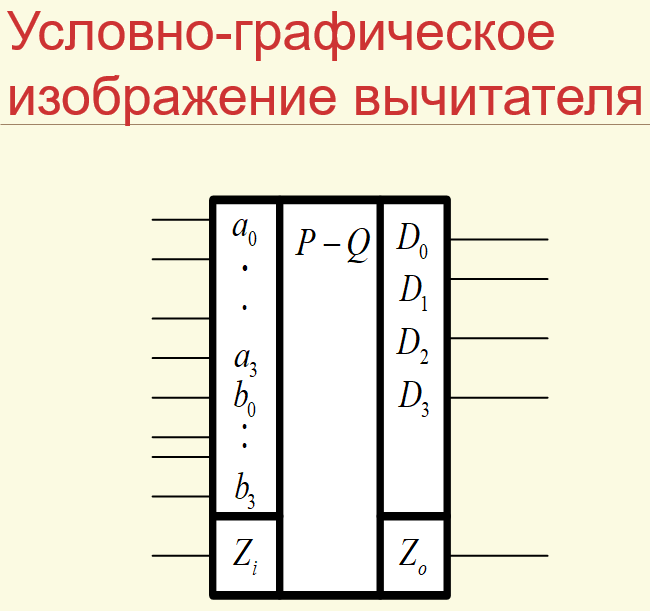


# **49 (43). Схема и принцип построения полного вычитателя.**

Вычитатель – устройство комбинационного типа, предназначенное для выполнения операции вычитания двоичных чисел.

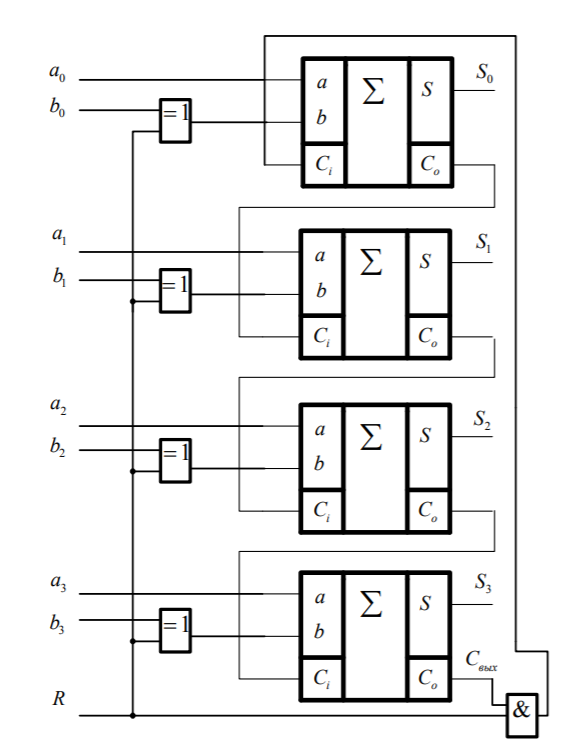
Полный вычитатель имеет помимо входов аргументов и третий вход входного заёма из младшего разряда





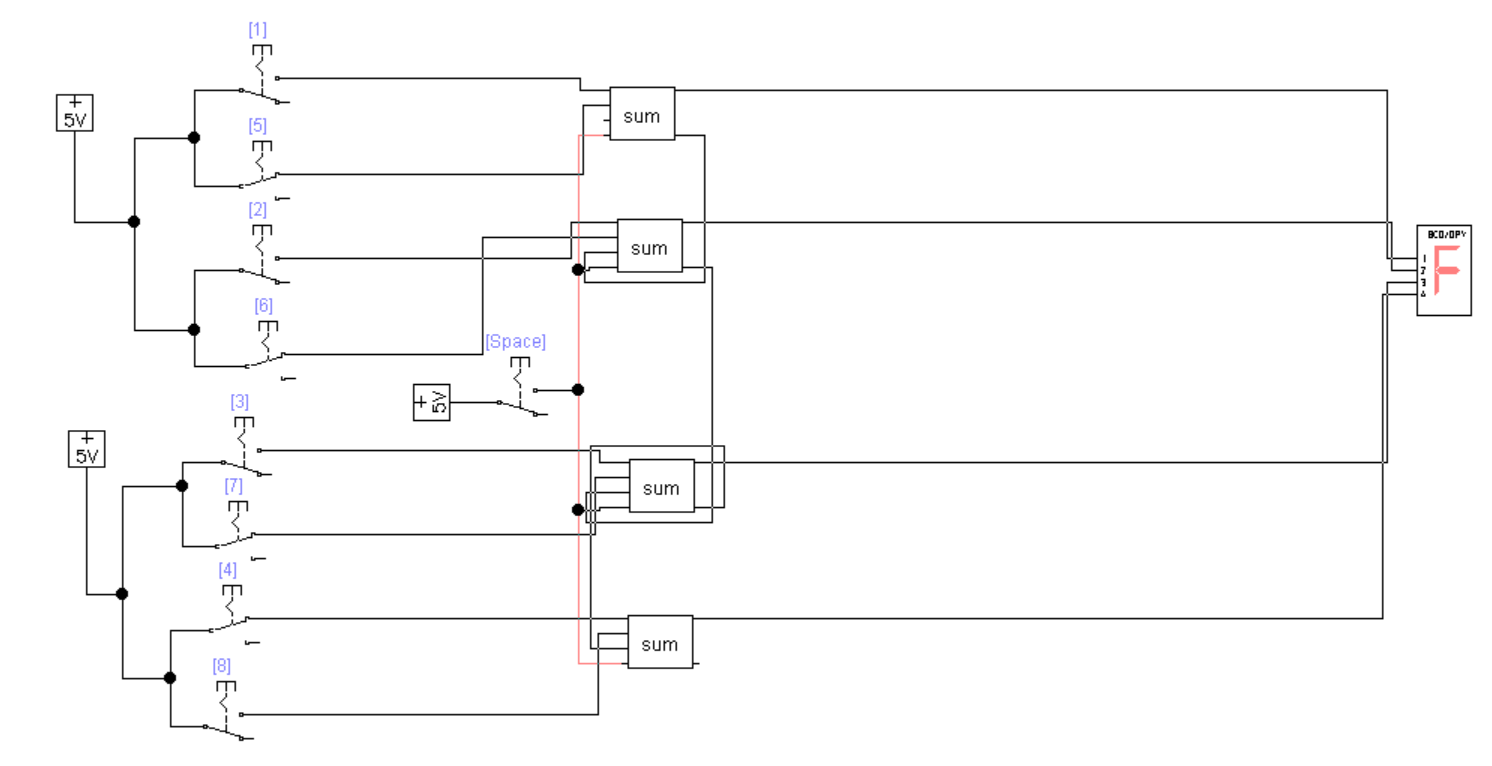
# **50 (50). Схема построения универсального сумматора-вычитателя. Пояснить принцип управления режимами.**

Особенность данного сумматора состоит в том, что в нём предусмотрено переключение между операцией суммирование и вычитание. Ограничение в его применения заключается в получении не прямого, а дополнительного кода при выполнении операции c = a−b, если a ≤ b. Схема построения универсального многоразрядного сумматора-вычитателя приведена на рисунке



(схема из учебника)



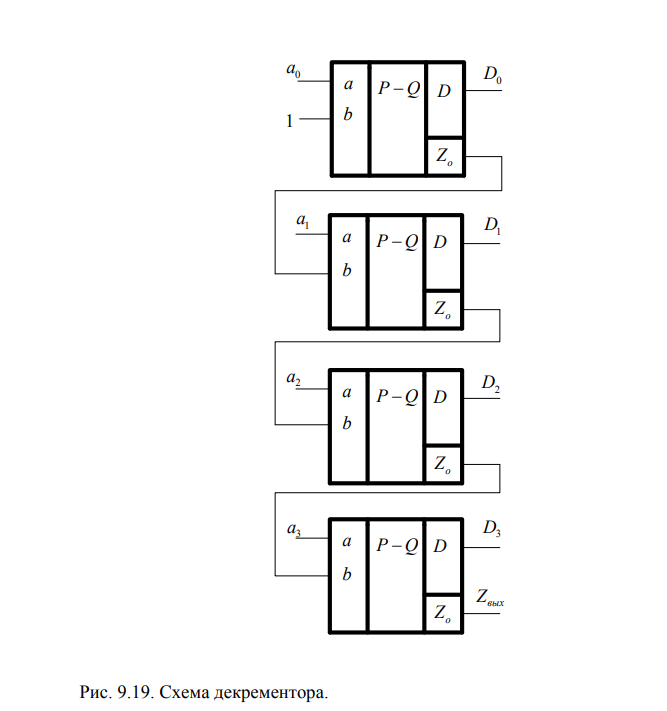


(схема из лаб)

Управления переключением режимов реализовано двоичным сигналом R. При R = 0 выполняется суммирование. Аргумент b проходит логические элементы «Исключающее ИЛИ» без изменений. При R =1, аргумент b попадает на входы сумматоров в инверсном коде. При этом возникающий перенос выхода старшего разряда подаётся через открытый элемент «И» на вход переноса младшего разряда.

# **51 (48). Особенности схем построения инкременторов и декременторов.**

**Инкрементор** — это электронное устройство, выполняющее арифметическое сложение многоразрядного кода с единицей младшего разряда.

**Декрементор** — это электронное устройство, выполняющее арифметическое вычитание из многоразрядного кода единицы младшего разряда.

Эти устройства строятся на основе полусумматоров и полувычитателей в силу того, что входы второго аргумента, начиная со второго разряда 190 задействованы под приём переноса (или заёма) из предыдущего младшего разряда. На вход второго аргумента младшего разряда подаётся единица.

# **52 (59). Применение компараторов. Пример синтеза схемы одноразрядного компаратора.**

Компараторы относятся к арифметическим устройствам. Они выполняют сравнение двух чисел*,* заданных в двоичных (либо других) кодах. Компаратор выдаёт сигналы на своих выходах о равенстве равноразрядных чисел А и В, либо их неравенство (А<В, А>В).

***Компараторы применяются*:**

- для выявления нужного числа в потоке чисел

- для отметки времени в часовых приборах

- для выполнения условного перехода в вычислительных устройствах

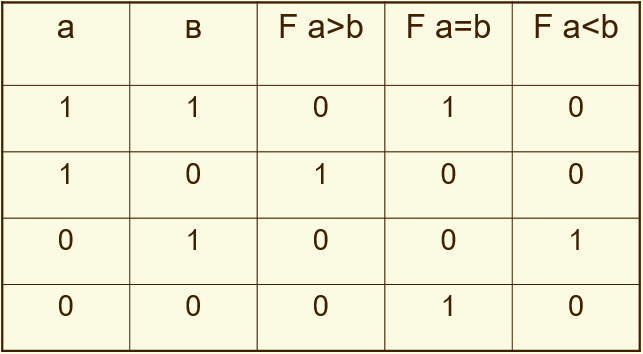
В *устройствах автоматики* **компараторы применяются**:

- для сигнализации о выходе величины за пределы допуска

- в приводах следящих систем для определения направления воздействия ликвидирующего рассогласование

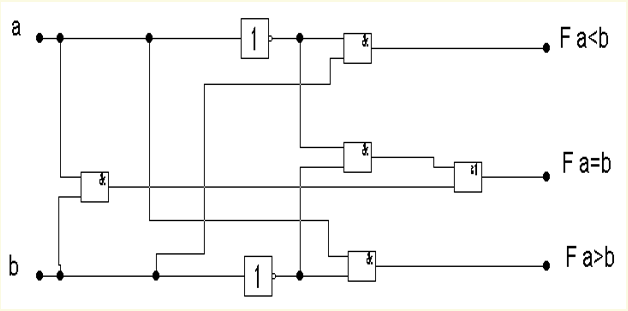
- при построении счётчиков по произвольному основанию

**Синтез одноразрядного компаратора:**

****

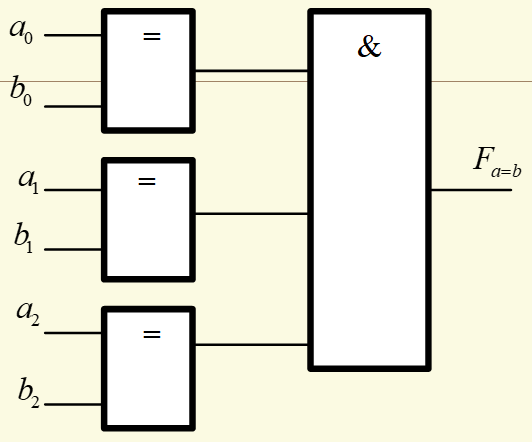
На соответствующем выходе компаратора 1, когда сравниваемые входные двухразрядные числа находятся в нужном соотношении.

F a>b =a!b, F a=b =ab+!a!b, F a<b =!ab



# **53 (57). Многоразрядный компаратор на равенство. Пример схемы реализации.**

Трёхразрядный компаратор, выполняющий сравнение двух трёхразрядных кодов на равенство:

****

При сравнении разряда на выходе логического элемента «равнозначность» выдаётся логическая единица. Выходы всех элементов «равнозначность» сведены на входы логического элемента «И». На выходе логического элемента «И» логическая единица выдаётся при равенстве всех разрядов.

# **54 (53). Множительный блок. Методика построения множительного блока.**

Выполнение операции умножения производится с помощью аппаратных средств процессоров, а также быстродействующих множительных блоков.

Структура множительного блока синтезируется на основе логики выполнения умножения n-разрядных целых операндов:

A2A1A0 \* B2B1B0 рассмотрим пример умножения двух трёхразрядных целых чисел

Операция умножения в столбик:

А2А1А0

\* В2В1В0

А2В0 А1В0 А0В0

+ А2В1 А1В1 А0В1

А2В2 А1В2 А0В2

Р5 Р4 Р3 Р2 Р1 Р0

Блок построен на логических элементах «И» и сумматорах.

Сумматоры изображены упрощенно, некоторые из них являются полусумматорами – те что не имеют входного переноса – крайние правые на каждой строке

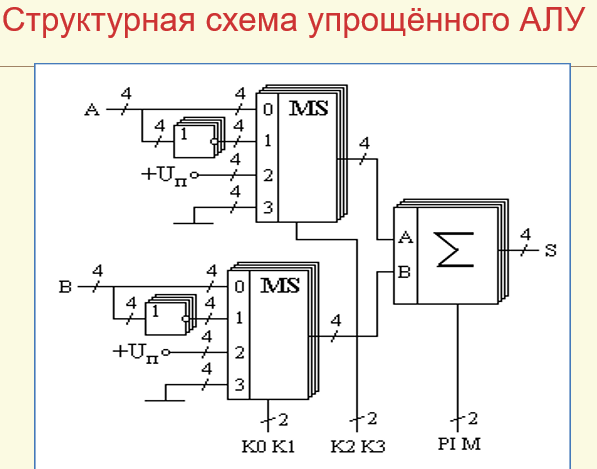


АЛУ, ЗУ

# **55 (61). Структура простого АЛУ, предназначенного для выполнения арифметических операций. Интерфейс микросхемы АЛУ.**

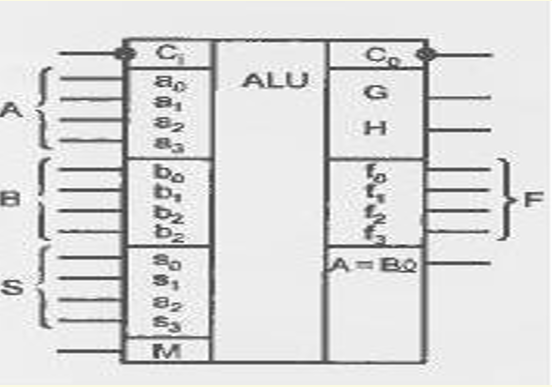
АЛУ (арифметико-логическое устройство) – узел электронно-вычислительных устройств, предназначенный для выполнения логических, арифметических операций и операций сравнения.

Основой АЛУ служит сумматор, схема которого дополнена логикой, расширяющей функциональные возможности АЛУ и обеспечивающей его перестройку с одной операции на другую.

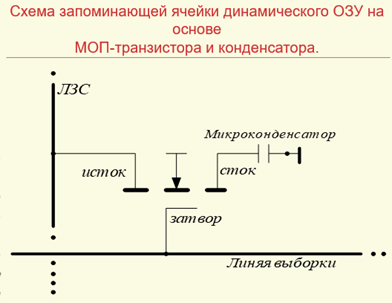


В данной схеме используются четырех-входовые коммутаторы (мультиплексоры: К0-3 адресные) , для управления каждым из которых достаточно двух бит. То есть для управления всей схемой в целом достаточно четырёх проводов управления. В зависимости от значений управляющих сигналов K0, K1, K2, K3 устройство будет совершать различные математические операции над входными значениями операндов A и В.

**АЛУ как отдельная микросхема**

АЛУ имеет входы операндов А и В, входы выбора операций S, вход переноса **C**i и вход М (Моdе), сигнал которого задает тип выполняемых операций: логические (М = 1) или арифметико-логические (М = 0). Результат операции вырабатывается на выходах F, выходы G и Н дают функции генерации и прозрачности, используемые для организаций параллельных переносов при наращивании размерности АЛУ. Сигнал **C**0 — выходной перенос, а выход А = В есть выход сравнения на равенство с открытым коллектором.

# **56 (19). Поясните принцип хранения бита информации в динамическом ОЗУ.**

Принцип хранения данных в динамическом ЗУ основан на создании заряда на микроконденсаторе в составе полевого транзистора МОП-типа, являющегося вместе с конденсатором запоминающей ячейкой динамического ОЗУ. Одно из преимуществ динамического ЗУ – в 4-5 раз меньшая площадь запоминающей ячейки по сравнению со статическим ЗУ. Однако из-за малой ёмкости конденсатора в 0.1….0.2 пФ. Его необходимо перезаряжать через несколько миллисекунд. Эта технология получила термин: регенерация памяти. Однако этот недостаток не повлиял на бурное развитие и совершенствование схем подобной динамической памяти.

Ключевой транзистор запоминающей ячейки с встроенным в структуру стока микроконденсатором подключается к линии записи считывания посредством подключения к ней истока. Сток транзистора не имеет внешнего вывода и образует одну из обкладок конденсатора. Другая обкладка конденсатора соединяется с заземлённой подложкой. Между обкладками располагается тонкий слой поликремния. В режиме хранения транзистор заперт. При выборке запоминающего элемента на затвор подаётся напряжение, отпирающее транзистор. Запоминающая ёмкость (конденсатор) через проводящий канал транзистора подключается к линии записи-считывания. При различном состоянии конденсатора состояние потенциала линии записи-считывания изменяется по-разному.

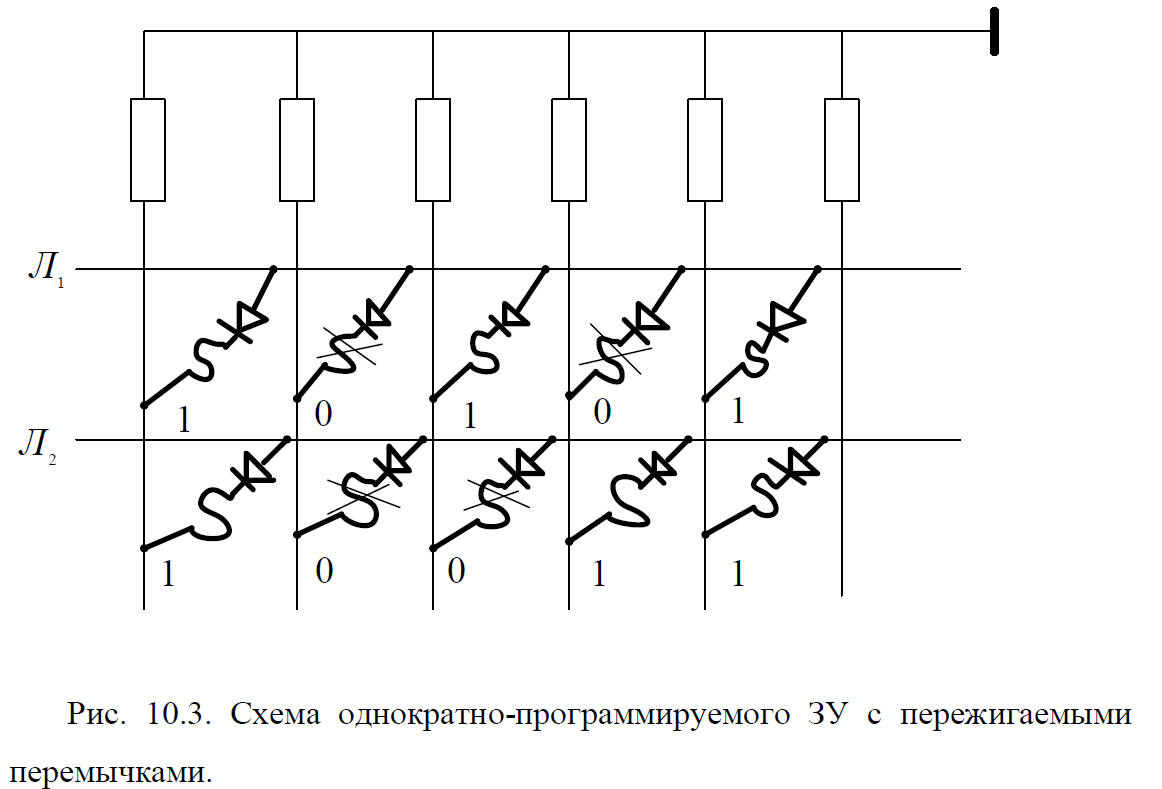
Если конденсатор хранил заряд, то этот заряд, стекая на ЛЗС, увеличивает её номинальное значение потенциала.

Если конденсатор не был заряжен, то потенциал ЛЗС понизится, т.к. произойдёт зарядка конденсатора. Данный вид считывания является «разрушающим», т.е. требует восстановления данных после их считывания (регенерации).

# **57 (24). Принцип построения однократно-программируемого постоянного запоминающего устройства.**

ЗУ типа PROM программируются пользователем устранением или созданием

перемычек рис 10.3.



В исходном состоянии ЗУ имеет все перемычки, а при

программировании часть их ликвидируется путем расплавления импульсами

тока (большой амплитуды и длительности). Эти перемычки включаются в

электроды диодов или транзисторов. Перемычки изготавливаются

металлическими (нихром) и поликристаллическими (кремниевыми).

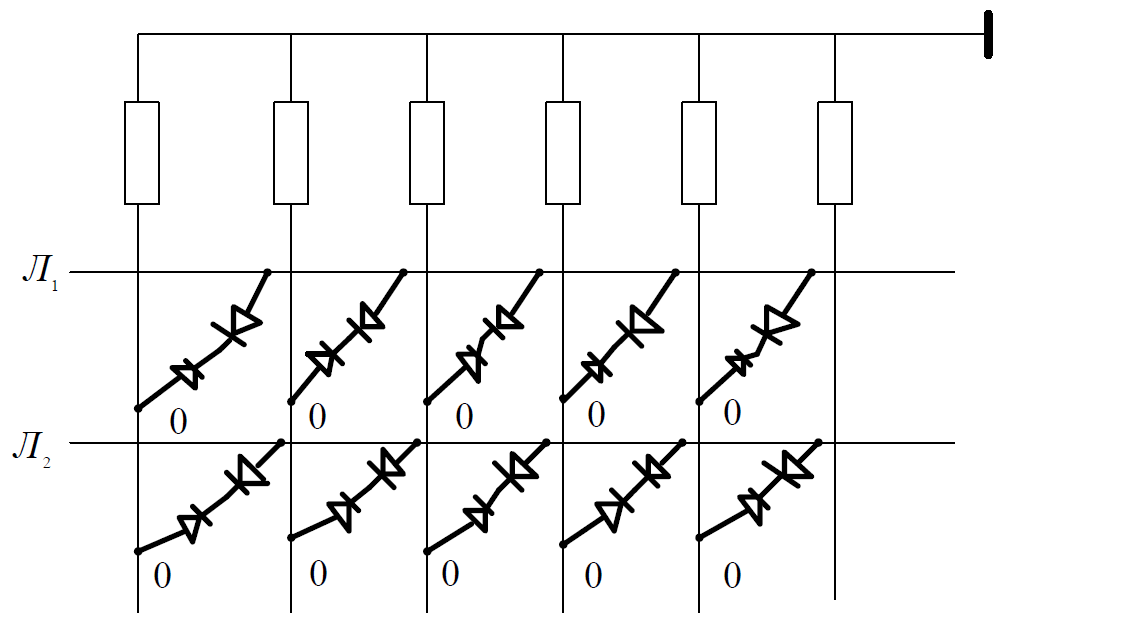
Другой тип перемычки - два встречно включенных диода рис.10.4. В

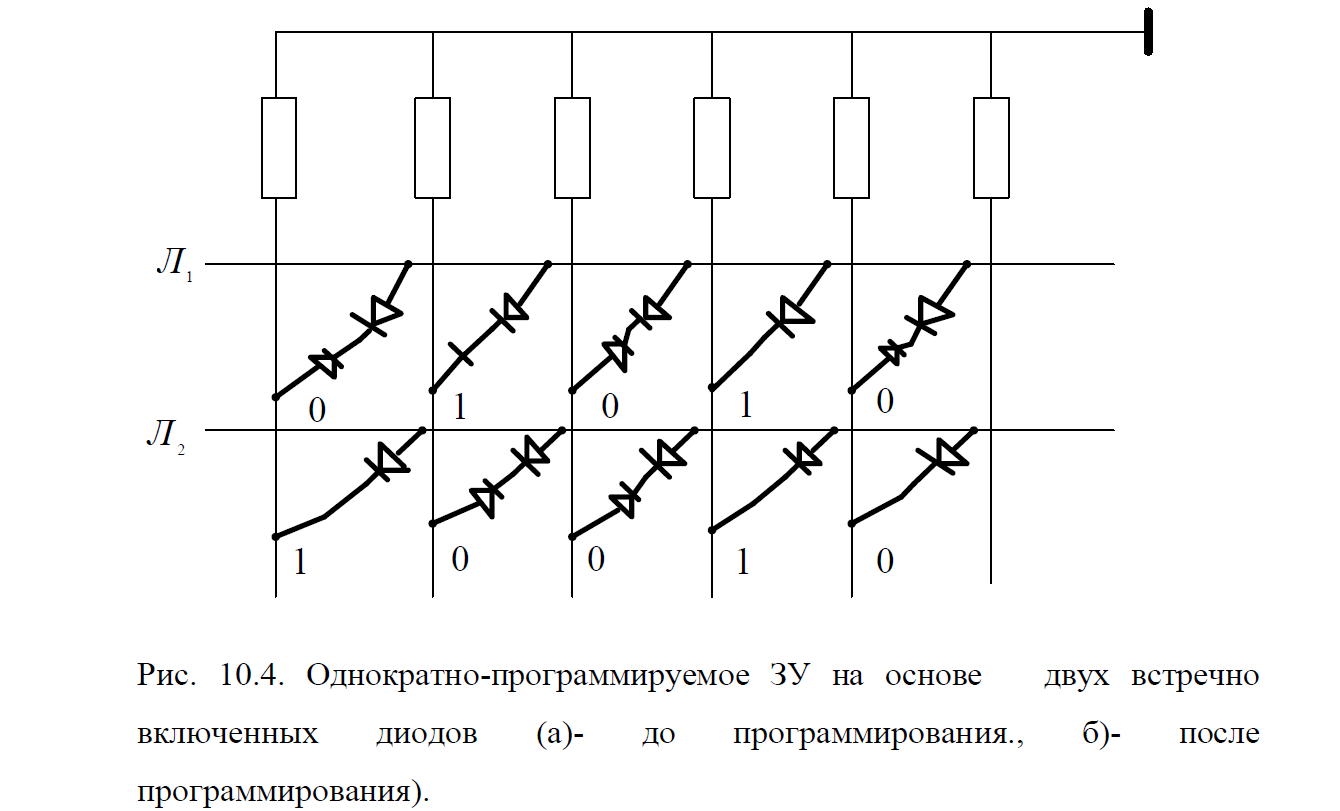
исходном состоянии цепь можно считать разомкнутой. Для записи единицы к

диодам прикладывается высокое напряжение, пробивающее диод,

смещенный в обратном направлении. Диод пробивается с образованием в

нем короткого замыкания.





# **58 (28). Принцип построения перепрограммируемого постоянного запоминающего устройства.**

Перепрограммируемые ПЗУ позволяют многократно осуществлять

прошивку памяти. В настоящее время наибольшее распространение

получили ПЗУ с электрическим стиранием предыдущей записанной в ПЗУ

информации. Ячейка хранения такого ПЗУ строится на основе полевого

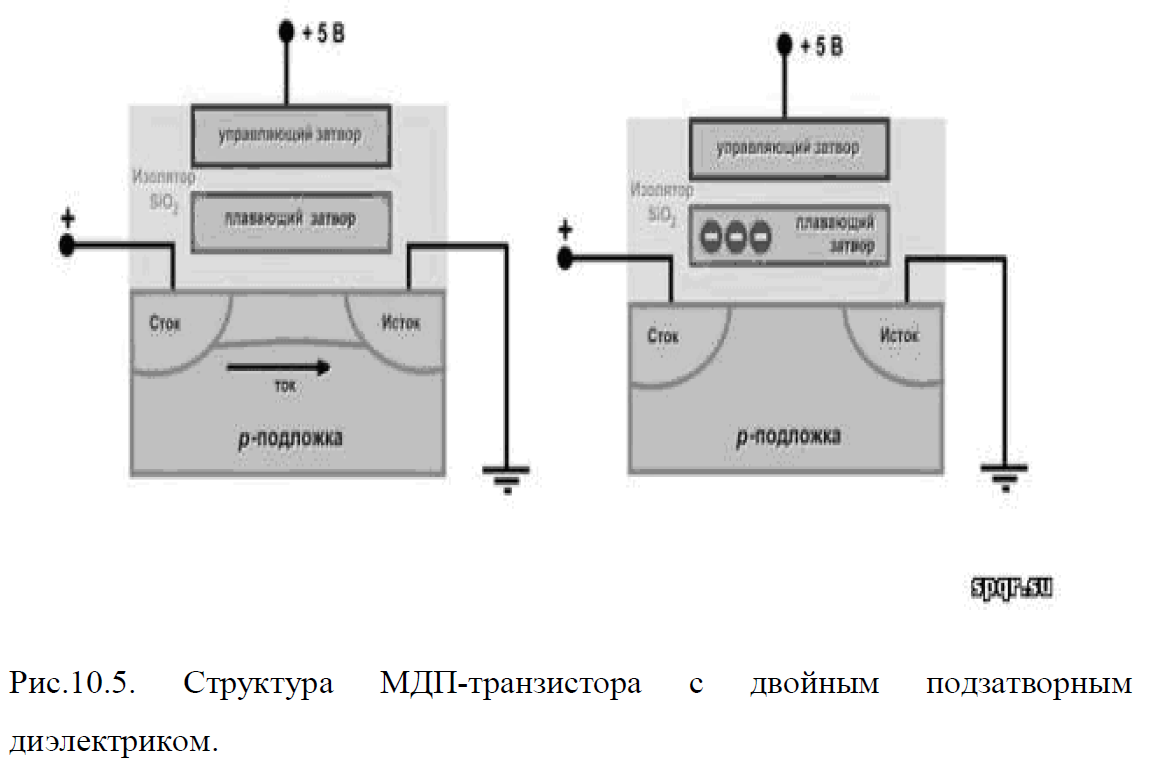
транзистора МДП-типа с двойным подзатворным диэлектриком. Эффект

хранения бита информации основывается на двух состояниях полевых

транзисторов. Транзистор может иметь ≪плавающий≫ заряд в подзатворном пространстве или не иметь его. Если в транзисторе создан заряд, то для перевода транзистора в открытое состояния поданного на затвор напряжения будет недостаточно. Транзистор останется в высокоимпедансном состоянии.

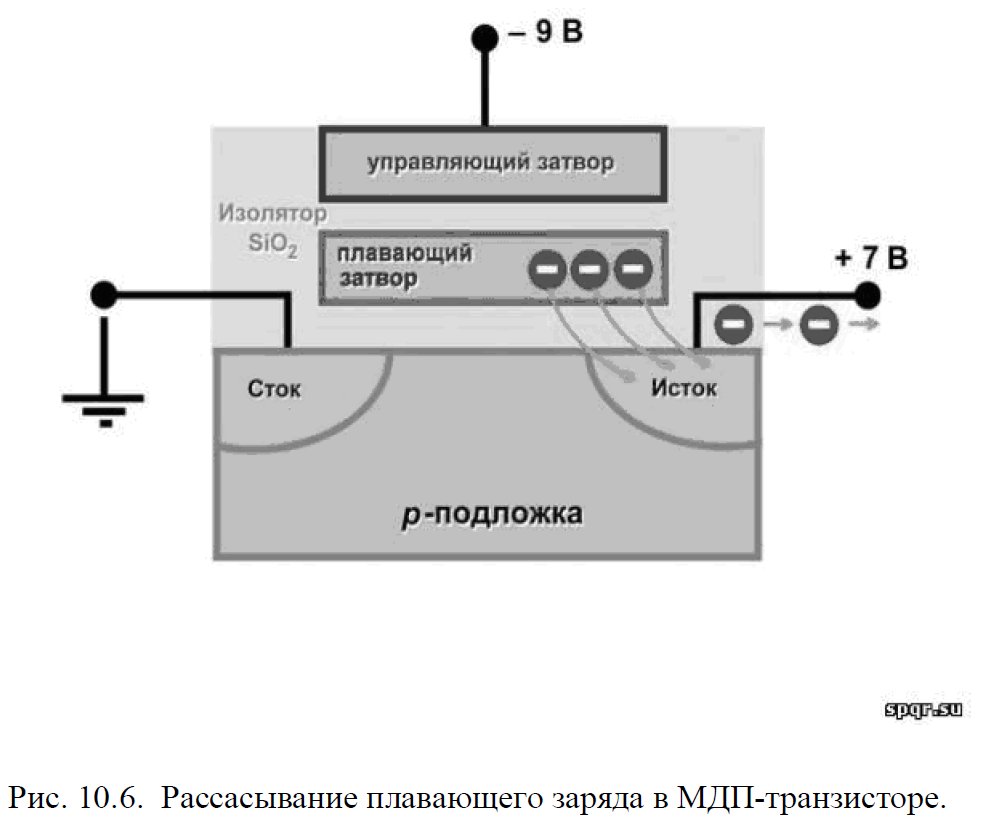
Если заряд не создан, то полевой транзистор переведётся в насыщенное

(открытое) состояние и тем самым обеспечит соответствующее напряжение на линии считывания. На рис. 10.5. представлена структура МДП-транзистора с двойным подзатворным диэлектриком.

 Чтобы записать в плавающий затвор информацию создаём большую разность потенциалов между стоком и истоком и естественно положительный на затвор. Возникает канал. В транзисторе в этом случае протекает большой ток, причём такой силы, что слегка пробивает изолятор и заносит электроны в плавающий затвор. При резком выключении тока электроны остаются в плавающем затворе, так как покинуть его уже не могут ведь энергию для преодоления изолятора даёт большой ток. В общем плавающий затвор остаётся отрицательно заряжен при полном отключении напряжения. Ток через такой транзистор не пойдёт в дальнейшем так как отрицательный заряд эффектом поля закрывает канал. Чтобы стереть заряд на плавающем затворе подаём на затвор большое напряжение "сгоняющее" с плавающего затвора электроны и, следовательно, заряд. При отсутствии заряда на плавающем затворе и при привычной подаче положительного потенциала на затвор канал открывается и ток протекает, создавая падение напряжения, соответствующее логической единице. Как мы видим, есть два устойчивых состояния:

1)ток не идёт через транзистор, есть заряд на плавающем затворе

2)ток идёт, нет заряда на плавающем затворе, а два устойчивых состояния дают право такому транзистору использоваться в качестве запоминающего устройства, которое мы называем обычным языком "флешка"



# **59 (23). Принцип построения схемы масочного постоянного запоминающего устройства.**

Масочные ПЗУ. Масочные ПЗУ могут быть реализованы на основе

диодов, биполярных транзисторов, МОП-транзисторов. Масочные ПЗУ

используются при изготовлении большой серии, и их прошивка

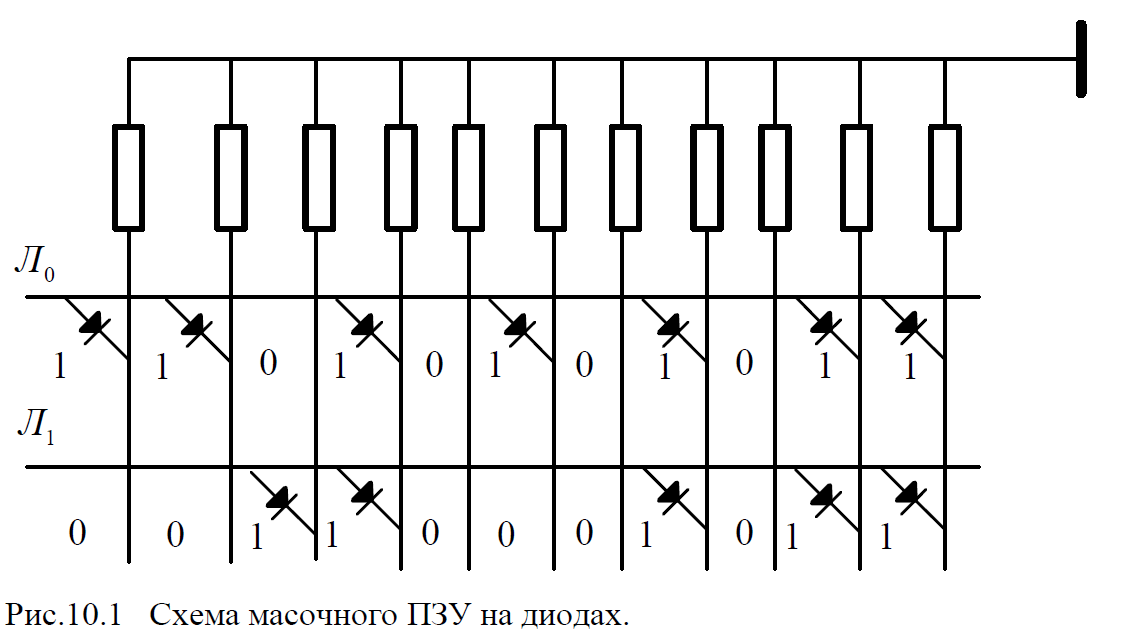
осуществляется на этапе изготовления.

На рис. 10.1 представлена матрица диодных элементов масочного ПЗУ. На

линиях считывания в пересечениях с адресными линиями, там, где «зашита»

логическая единица, диод соединяет адресную линию с разрядной линией

считывания. На пересечениях где, зашит ноль, диод не подключён.



Информация в подобное масочное ЗУ заносится в процессе изготовления

путём металлизации промежутков, позволяющих соединить через диоды

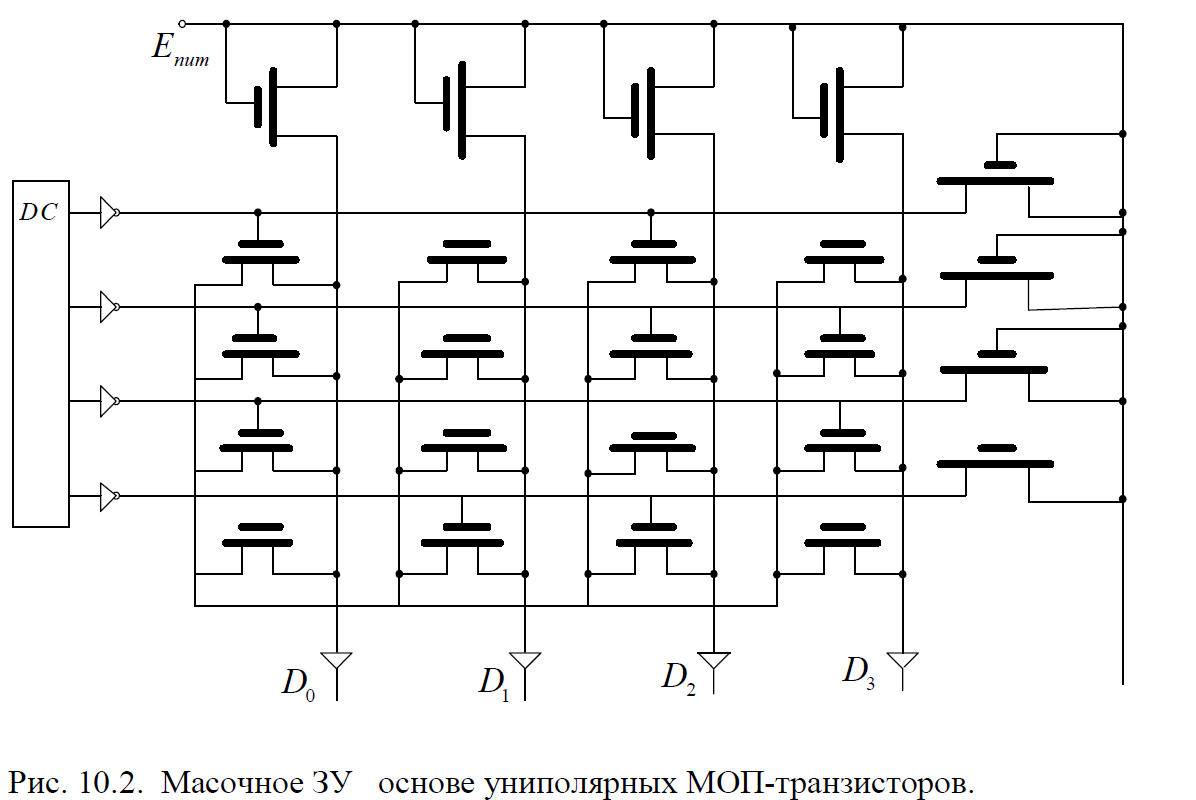
соответствующие линии строк и столбцов. Каждая прошивка требует своего

фотошаблона. Масочные ЗУ наиболее надёжны и допускают высокую

плотность записи информации.

На рис. 10.2 представлено масочное ЗУ на основе униполярных МОП-

транзисторов.



Программирование заключается в подключении затвора к адресной линии в

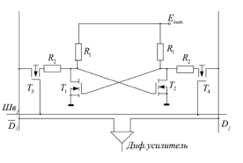
тех местах матрицы памяти, где должна быть записана единица. Там, где

должен быть записан 0, коммутация затвора на адресную линию отсутствует.

# **60 (29). Принцип построения статического оперативного запоминающего устройства.**

ОЗУ делятся на статические и динамические.

Статические ОЗУ – ОЗУ, ячейка которого хранит значение неограниченно долго при подаче питающего напряжения. Ячейка памяти - RS-триггер



Шв – выбор строки (активация ячейки)

С Диф. Усилителя считываем значения. Di,j устанавливают значения. Подобные ячейки объединяются и образуют СОЗУ. Примерная схема работы:

