

## Понятие последовательностного автомата

Последовательностные цифровые устройства часто называют последовательностными схемами, последовательностными автоматами, дискретными автоматами с памятью, многотактными автоматами.

Кроме комбинационных устройств, рассмотренных в предыдущей главе, существует класс цифровых устройств, в которых при одинаковых воздействиях на входе, на выходе автомата могут возникать различные выходные состояния. Состояние выхода такого устройства зависит не только от того, какие сигналы присутствуют на его входах в данный момент времени, но и от того, какие последовательности сигналов поступали на входы устройства в предшествующие моменты времени, т.е. как говорят, автомат помнит свою предысторию и хранит ее в памяти. Поэтому такие устройства называют последовательностными или многотактными автоматами.

Для описания последовательностного автомата с памятью, помимо состояний входов  $X(t)$  и выходов  $Y(t)$ , необходимо также знать состояние памяти автомата, как говорят, его внутреннее состояние  $S(t)$ .

В общем виде, последовательностный автомат рассматривается состоящим из двух частей: комбинационного устройства (КУ) и памяти, состоящей из элементов памяти (ЭП) (рис. 3.1).

В качестве элементов памяти могут быть применены как однобитовые элементы памяти (различные типы триггеров), так и многобитовые (многозарядные) цепочки триггеров.

Функционирование (т.е. изменение состояния устройства) многотактного автомата происходит в дискретные моменты времени, ход которого обозначается натуральными числами  $t = 1, 2, 3$  и т.д. В каждый момент дискретного времени  $t$  автомат находится в определенном состоянии  $S(t)$ , воспринимает через входы соответствующую данному моменту комбинацию входных переменных  $X(t)$ , выдает на выходах некоторую функцию выхода  $Y(t)$ , определяемую как

$$Y(t) = f(S(t), X(t)),$$

и переключается в новое состояние  $S(t+1)$ , которое определяется функцией переходов  $\varphi$  как

$$S(t+1) = \varphi(S(t), X(t)).$$

Закон функционирования последовательностных автоматов может задаваться в виде уравнений, таблиц и графов. Под законом функционирования понимается совокупность правил, описывающих последовательность переключения состояний автомата и последовательность выходных сигналов в зависимости от последовательности поступления входных сигналов.



Рис. 3.1. Структурная схема последовательностного автомата; ЭП - элементы памяти

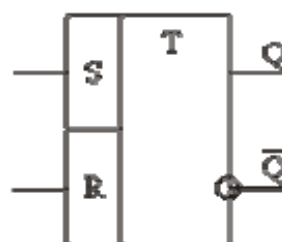
ЭП часто реализуется на основе бистабильных ячеек (БЯ). ЭП бывают: динамические, статические и квазидинамические. Наибольшее распространение получили первых два вида: статические и динамические. Статические ЭП реализуются на БЯ. Динамические ЭП строятся, используя свойство хранения заряда между затвором и истоком полевых транзисторов МДП структуры.

БЯ - является простейшим типом триггера, реализованного с помощью элементов базиса И - НЕ или ИЛИ - НЕ с соответствующими обратными связями и позволяет хранить один бит цифровой информации (бит - единица цифровой информации, соответствующая одной логической "1" или логическому "0").

Триггер - это последовательностная схема с двумя состояниями, каждое из которых при определенных условиях на входах поддерживается постоянным (т.е. стабильным). Каждому из этих состояний ставится в соответствие логическое значение, которое "хранит" триггер (если на выходе триггера высокий уровень напряжения - "1" и "0" - в противном случае). Таким образом, в последовательностной схеме для представления значений каждой переменной, которую нужно хранить для использования в настоящем или в будущем, следует использовать отдельный триггер. Совокупное состояние последовательностной схемы, запоминающее устройство которой реализовано на триггерах, представляет собой просто комбинацию состояний этих триггеров.

Вообще говоря, у триггера должна быть по крайней мере одна выходная линия, представляющая логическое значение, соответствующее состоянию триггера. Когда на выходной линии логическая 1, говорят, что триггер установлен, в противном случае говорят, что триггер сброшен. Триггер имеет несколько входных линий, сигналы на которых (вместе с текущим состоянием триггера) определяют следующее состояние триггера. От функций входных линий зависит тип триггера.

Простейшим триггером является RS - триггер. RS - триггер имеет два входа и два выхода. Входы и выходы триггера имеют свои обозначения. Один из входов триггера называется установочным входом и обозначается буквой **S** (от английского set - установить), а другой - входом сброса и обозначается буквой **R** (от reset - сбросить). Триггер (рис. 3.2) имеет два симметричных выхода. На одном выходе (условно называемом прямым выходом) сигнал представляется без отрицания (выход Q), а на другом - с отрицанием ( $\bar{Q}$  - инверсный выход).



### Рис. .3.2. Графическое обозначение RS - триггера

Триггеры бывают переключающимися уровнем и фронтом тактирующего сигнала ( импульса ). Несмотря на большое разнообразие триггеров, практически все триггеры строятся на базе RS -триггеров.

© Ёжич , ВВ-2-06  
[vv206.selfip.org](http://vv206.selfip.org)

## Триггеры

Триггер – это логическая схема с положительной обратной связью, могущая неограниченно долго находиться в одном из двух устойчивых состояний ( 0 и 1 ) которые обеспечиваются положительной обратной связью а не входным сигналом. Триггер скачком, лавинообразно меняет одно состояние на другое под воздействием входного сигнала.

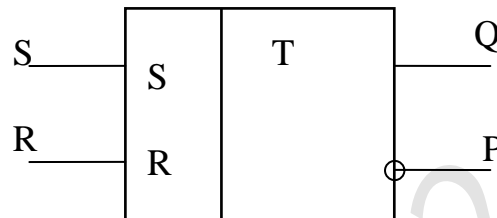
Триггеры делятся на симметричные и несимметричные (триггеры Шмитта). Триггеры Шмитта рассматриваются как импульсные устройства, а симметричные делятся :

1. По способу записи информации на:
  - а) асинхронные,
  - б) синхронные (тактируемые).
2. По способу управления записью информации на:
  - а) статические,
  - б) динамические – по фронту переднему или заднему,
  - в) двухступенчатые – по двум фронтам.
3. По способу организации логической связи :
  - а) элементарные с отдельной установкой 0 и 1 – RS–триггеры
  - б) с приемом информации по одному входу – D–триггеры
  - в) со счетным входом – T–триггеры
  - г) универсальные с отдельной установкой 0 и 1 – JK–триггеры
  - д) комбинированные ( смешанные )

## RS – триггер.

Триггер имеет два выхода : прямой Q и инверсный  $P = \bar{Q}$ . Перевод триггера в состояние  $Q=1$  называют установкой (set) и вход, по которому это осуществляется, обозначается S.

Перевод триггера в нулевое состояние  $Q=0$  называют сбросом (reset) и вход сброса обозначают R. Триггер является асинхронным, если его переключение происходит в момент поступления управляющих сигналов. Триггеры характеризуются таблицей переключений.



Обозначение RS-триггера на схемах.

$S_n$	$R_n$	$Q_n$	$Q_{n+1}$	$P_{n+1}$
0	0	0	0	1
0	0	1	1	0
0	1	0	0	1
0	1	1	0	1
1	0	0	1	0
1	0	1	1	0
1	1	0	X	X
1	1	1	X	X

Таблица переключений асинхронного триггера.

Составленная по таблице переключений триггера карта Карно имеет вид:

$S_n, R_n, Q_n$	0 0	0 1	1 1	1 0
0	0	0	X	1
1	1	0	X	1

Столбец карты Карно, соответствующий ситуации  $S_n, R_n=11$ , называется столбцом неопределенности. В зависимости от возможных значений логических уровней в этом столбце, получаются различные схемы триггеров.

### RS-триггер с инверсными входами.

Клетки столбца неопределенности принимают значения 1. Это означает, что при  $S_n, R_n=1$  выходное состояние триггера становится единичным, независимо от того, в каком состоянии он был ранее. Составленная по карте Карно формула триггера в этом случае имеет вид:

$$Q_{n+1} = S_n + Q_n * \overline{R_n}$$

Для построения RS-триггера на элементах И-НЕ служит формула, полученная в результате преобразования исходной формулы, с применением 1-го постулата Де Моргана.

$$Q_{n+1} = \overline{\overline{S_n} * \overline{Q_n} * \overline{R_n}}$$

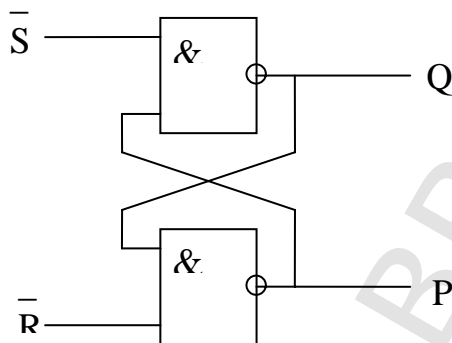


Схема RS-триггера с инверсными входами на элементах И-НЕ.

### RS-триггер с прямыми входами.

Клетки столбца неопределенности принимают значение 0. Это означает, что при  $S_n, R_n=11$  триггер устанавливается в нулевое состояние, независимо от предыдущего состояния.

Составленная по карте Карно формула триггера в этом случае имеет вид

$$Q_{n+1} = S_n * R_n + Q_n * \overline{R_n}$$

Для построения RS-триггера на элементах ИЛИ-НЕ служит формула, полученная в результате преобразования исходной формулы.

$$Q_{n+1} = \overline{\overline{R_n + S_n + Q_n}}$$

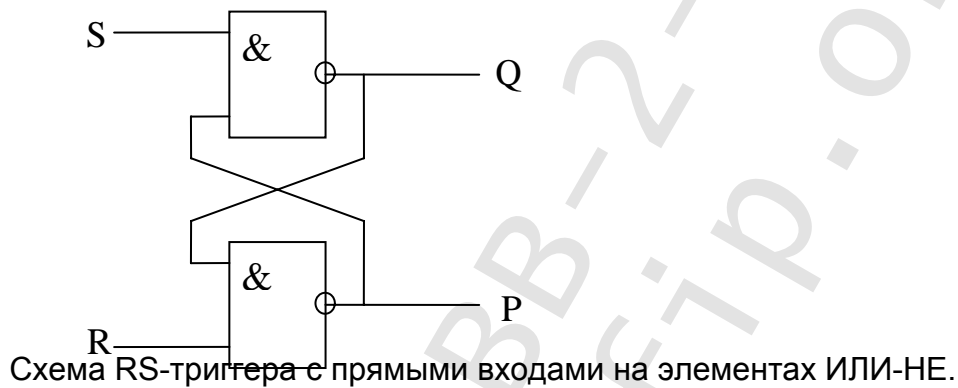


Схема RS-триггера с прямыми входами на элементах ИЛИ-НЕ.

## Е - триггер.

Клетки столбца неопределенности принимают значения, соответствующие состоянию триггера до прихода сигналов  $S_n$   $R_n=1$ . Т.е. триггер сохраняет предыдущее состояние  $Q_n$ . Формула Е-триггера имеет вид:

$$Q_{n+1} = S_n * \overline{R_n} + S_n * Q_n + \overline{R_n} * Q_n$$

Преобразованная формула для реализации на элементах И-НЕ имеет вид:

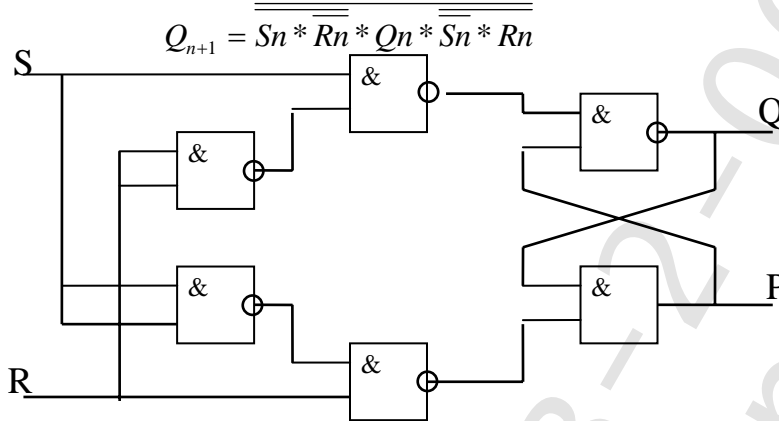


Схема Е-триггера на элементах И-НЕ.



## JK триггеры.

В серии К155 имеется две микросхемы, содержащих JK триггеры. Это микросхемы К155ТВ1 и К155ТВ15.

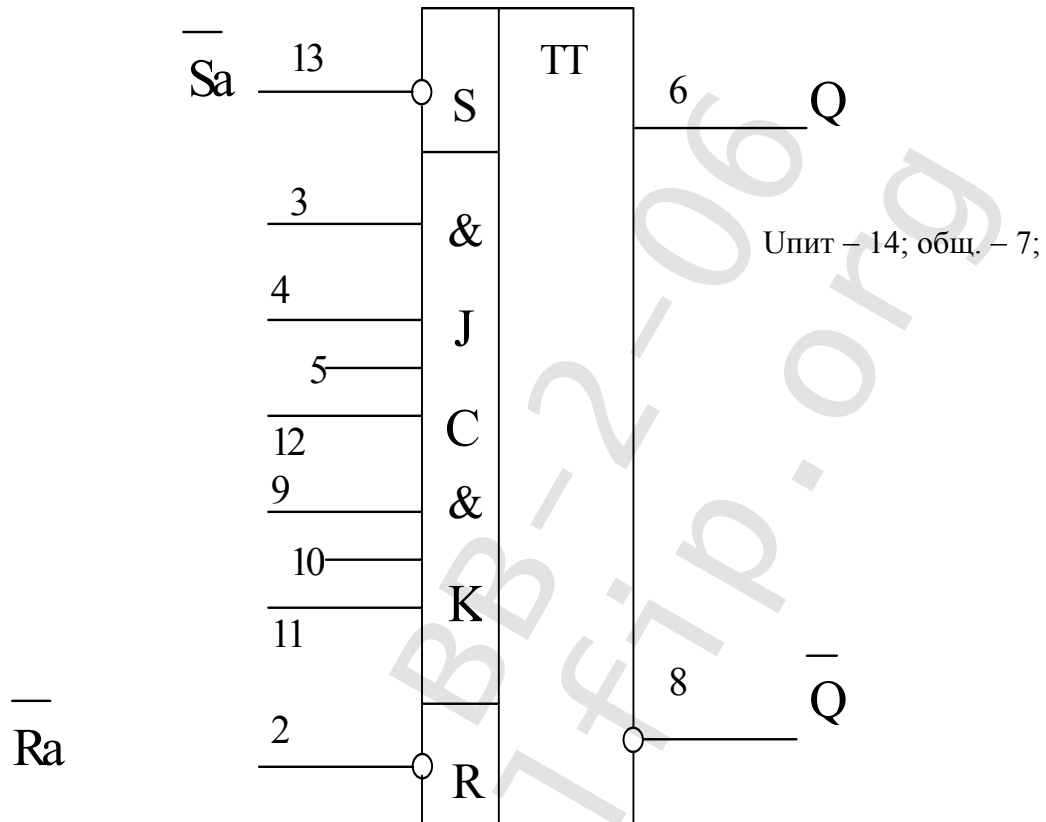


Рис. 1. Условное изображение ИМС К155ТВ1.

JK триггер К155ТВ1 с двух ступенчатым управлением, т.е. имеет MS<sup>~</sup> структуру. Триггер имеет по три входа J и K, связанных операцией И, а также асинхронные входы Ra и Sa для установки триггера в состояние 1 и 0 независимо от сигналов на информационных и синхронизирующем входах.

Асинхронная установка триггера в единичное или нулевое состояние производится импульсами низкого уровня (логический 0 при положительной логике), поступающими на входы Ra или Sa. Состояние входов J, K и C при асинхронном управлении безразлично. В отличие от RS<sup>~</sup> триггера здесь одновременная подача переключающих сигналов на входы Ra и Sa допустима. Когда эти входы не задействованы, на них следует поддерживать напряжение высокого уровня. Таблица состояний триггера К155ТВ1 приведена в табл. 1. Символ  $\uparrow$  означает, что синхроимпульс C изменяется от низкого уровня до высокого уровня (перепад 0,1); символ  $\downarrow$  означает, что синхроимпульс C изменяется от высокого уровня до низкого уровня (перепад 1,0).

Таблица 1.

Таблица переключений триггера К155ТВ1

Sa	Ra	Jn	Kn	C	Q <sub>n+1</sub>	Режим
1	1	x	x	$\uparrow$	Q <sub>n</sub>	хранение
1	1	0	0	$\downarrow$	Q <sub>n</sub>	хранение
1	1	0	1	$\downarrow$	0	синхронный
1	1	1	0	$\downarrow$	1	синхронный
1	1	1	1	$\downarrow$	Q <sub>n</sub>	синхронный

1	0	x	x	x	0	асинхронный
0	1	x	x	x	1	асинхронный
0	0	x	x	x	$Q_n=Q_{n-1}$	асинхронный

JK-триггеры относятся к универсальным устройствам. Их универсальность имеет двойственный характер. Во-первых, эти триггеры с равным успехом могут быть использованы в различных цифровых устройствах: регистрах, цифровых счётчиках, делителях частоты и т.д., во-вторых, путём определённого соединения выводов они легко обращаются в триггеры других типов. Структурная схема JK-триггера с динамическим управлением приведена на рис. 2.

D-триггер может быть получен из JK-триггера с помощью дополнительного инвертора (рис.3).

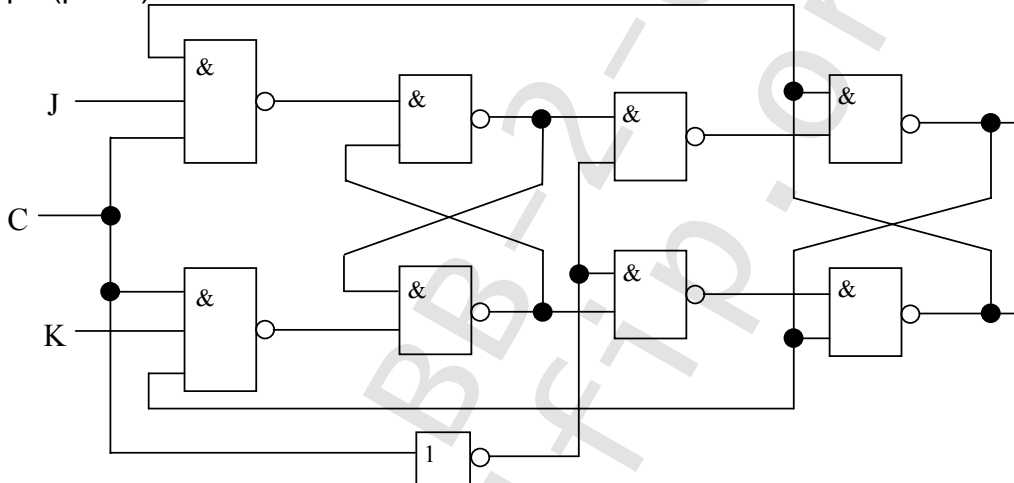


Рис.2. Структурная схема синхронного JK-триггера.

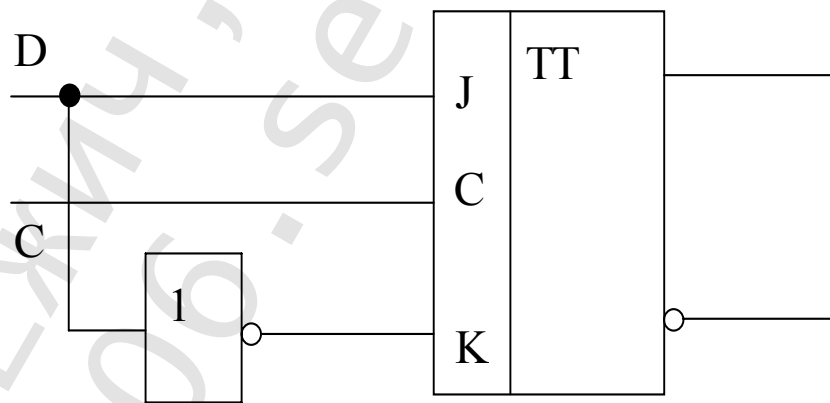
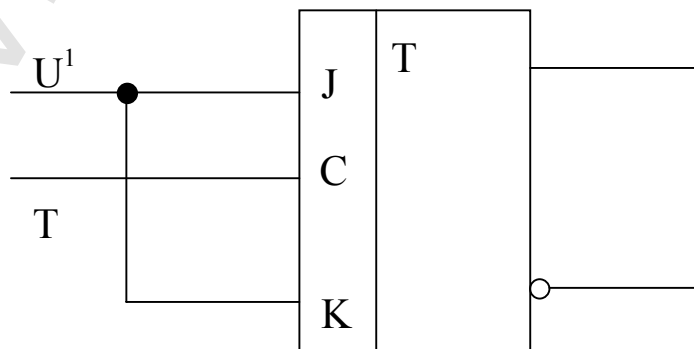


Рис.3. Организация D-триггера на базе JK-триггера.



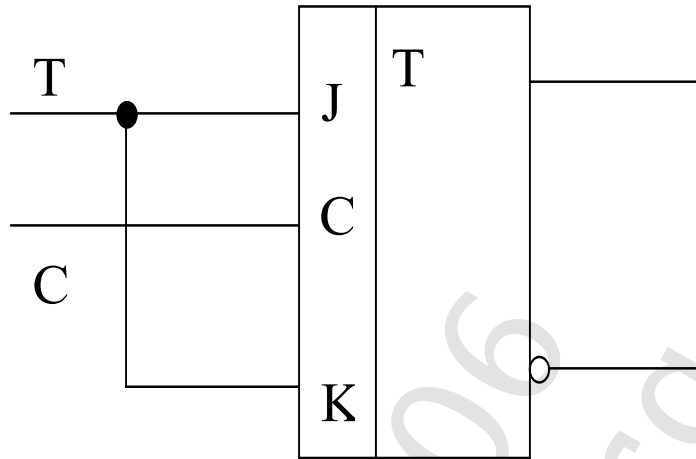


Рис.4. Преобразование JK-триггера в T-триггер:  
а) - асинхронный;  
б) - синхронный.

На рис.4 показаны способы получения T-триггера из универсального JK-триггера. В асинхронном режиме C-вход триггера выполняет роль счетного, в синхронном же режиме C-вход используется по прямому назначению (как синхронизирующий), а счетные импульсы подаются на соединенные входы J и K.

## D-триггер

D-триггер (от английского DELAY) называют информационным триггером, также триггером задержки. D - триггер бывает только синхронным. Он может управляться (переключаться) как уровнем тактирующего импульса, так и его фронтом. Для триггера типа D, состояние в интервале времени между сигналом на входной линии и следующим состоянием триггера формируется проще, чем для любого другого типа.

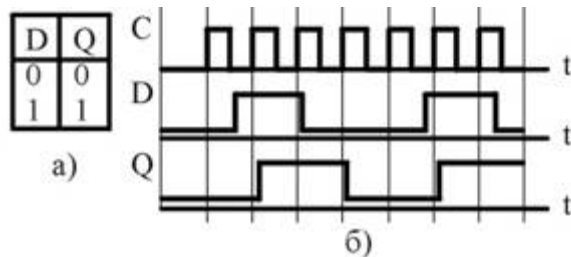


Рис. 3.8. Таблица управления (а) и временные диаграммы (б) D - триггера

По синхроимпульсу D-триггер принимает то состояние, которое имеет входная линия, согласно управляющей таблице состояний, приведенной на рис. 3.8, а. На рис. 3.8, б приведены временные диаграммы, поясняющие его работу.

Как следует из управляющей таблицы, D-триггер имеет как минимум две входные линии: одна - для подачи синхроимпульсов; другая- информационных сигналов. Схемное обозначение D - триггера приведено на рис. 3.9.

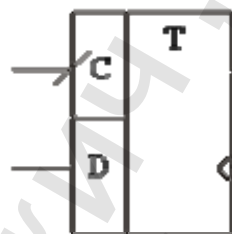


Рис. 3.9. Схемное обозначение D - триггера, синхронизируемого передним фронтом тактирующего импульса

Для получения характеристической формулы воспользуемся полной таблицей состояний (рис. 3.10).

C	D	$\bar{Q}$	Q
0	0	0	0
0	0	1	~
0	1	0	0
0	1	1	~
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Рис. 3.10. Таблица состояний D - триггера; знаком безразличного состояния (~) обозначены значения Q для двух наборов переменных, т.к. изменение состояния триггера при нулевом значении сигнала "С" не происходит

Для минимизации логического выражения (характеристической формулы триггера) можно воспользоваться картой Карно (рис. 3.11, а).

Из рис. 3.11, а следует, что характеристическое уравнение D-триггера содержит всего одну конъюнкцию, т.е.  $Q = CD$ .

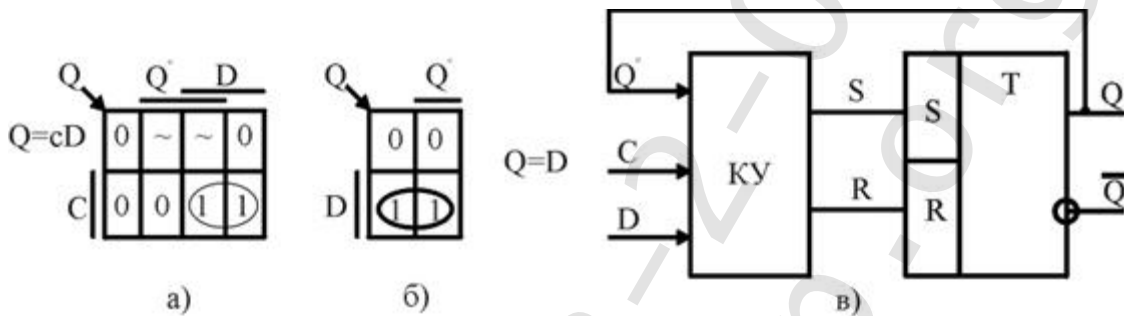


Рис. 3.11 Карты Карно, составленные по таблице состояний (а), управления (б) и функциональная схемы (в) D-триггера

Если учитывать, что при отсутствии синхроимпульсов состояние D-триггера не меняется, то можно упростить таблицу состояния, оставив в ней только переменные  $D$  и  $Q$ . Тогда карта Карно будет выглядеть как на рис. 3.11, б. Сокращенное уравнение D-триггера имеет всего один сигнал – сигнал  $D$ .

Это выражение подразумевает наличие сигнала  $C$ , т.к. в его отсутствии переключение состояния D-триггера не происходит.

Отметим, что D-триггеры могут переключаться как уровнем синхроимпульса, так и его фронтом. В технической литературе D-триггер, управляемый уровнем синхроимпульса, известен также как триггер-защелка.

**Пример синтеза D-триггера, управляемого уровнем синхроимпульса из асинхронного одноступенчатого RS-триггера.** Для этого D-триггер представляют как совокупность RS-триггера и комбинационного входного устройства, т.е. представляется так, что входными линиями RS-триггера управляет комбинационное устройство (КУ), согласно характеристической формуле D-триггера (см. рис. 3.11, в). Входными переменными КУ являются сигналы  $Q^0$ ,  $C$ ,  $D$ , а выходными (функциями) -  $S$  и  $R$ . Если учесть, что сигналы  $R$  и  $S$  являются для RS-триггера управляющими сигналами, то таблица состояний синтезируемого триггера будет содержать пять столбцов: два столбца - для переменных D-триггера -  $D$  и  $Q^0$ , один - для функции  $Q$  (выходной сигнал синтезируемого триггера, он же является выходным сигналом базового RS-триггера) и два столбца - для переменных  $R$  и  $S$  RS-триггера

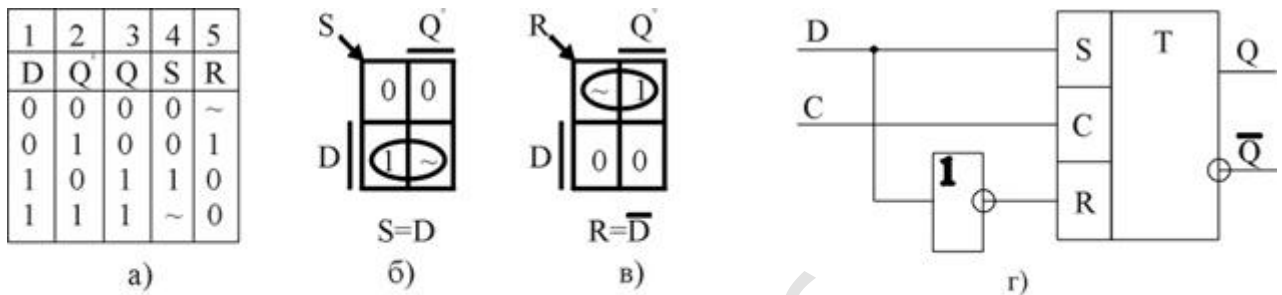


Рис. 3.12. Комбинированная таблица состояний а), карты Карно б) и в), и функциональная схема, синтезируемого D-триггера

Столбцы 1, 2, 3 соответствуют таблице состояний D-триггера, а в столбцы 4, 5 записываются значения сигналов R и S, при подаче которых на входы RS-триггера, последний должен принимать такие же состояния, что и D-триггер. Это обусловлено тем, что RS-триггер является выходным узлом D-триггера.

Из таблицы (рис.3.12, а, первая строка) следует: если триггер находился в состоянии "0" ( $Q^0 = 0$ ), чтобы он сохранил это состояние и после поступления очередного синхроимпульса (отметим, что переключение состояния триггера происходит только при наличии синхроимпульса, а каждая строка таблицы состояний соответствует новому синхроимпульсу) на входе S (RS-триггера) необходимо поддерживать уровень "0", а на входе R - любой уровень, т.к. когда триггер находится в состоянии "0", он сохраняет это состояние независимо от состояния сигнала R.

Для второй строки таблицы состояний  $Q^0 = 1$ , а новое состояние триггера "0" ( $Q = 0$ ), следовательно, необходимо подать на вход R - уровень логической 1 и т.д. Для каждой строки, где  $Q^0 = 1$ , это состояние триггера сохранится независимо от значения сигнала S (т.к. при  $S = 0$  - режим хранения, а при  $S = 1$  - запись единицы).

После заполнения таблицы состояний, используя карты Карно (рис. 3.12, б и в), записывают логические выражения для функций комбинационного устройства S и R (следует помнить, что эти сигналы являются функциями аргументов  $Q^0$ , D и входными переменными для RS-триггера).

По полученным логическим выражениям (см. рис. 3.12, б и в) можно построить схему D-триггера (рис. 3.12, г).

Рассмотренный выше D-триггер синтезирован на базе синхронного RS-триггера. Его можно синтезировать и на базе двухступенчатого, а также - простого, асинхронного RS-триггера. Как уже было отмечено выше, переключение D-триггера происходит только при наличии (поступлении) синхроимпульса. С учетом этого, логические функции S и R можно записать в виде

$$S = C D;$$

$$R = C \cdot \bar{D}$$

Схема, реализующая эти функции, содержит два элемента конъюнкции и один инвертор. На рис. 3.13 приведена схема D-триггера, построенного на базе асинхронного RS-триггера.

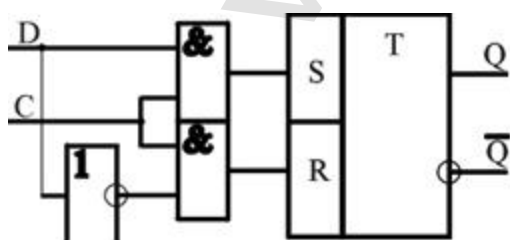
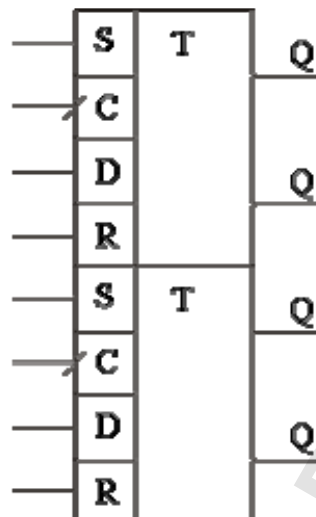


Рис. 3.13. D-триггер, управляемый уровнем синхроимпульса, построенный на базе асинхронного RS-триггера

На рис. 3.14 приведено обозначение D-триггера K1533TM2, выпускаемого промышленностью в виде интегральной микросхемы (ИМС).



**Рис. 3.14. Схемное обозначение ИМС K1533TM2; входы S и R-асинхронные (входы предустановки)**

Обычно, в одном корпусе ИМС содержится два D-триггера, управляемых фронтом. D-триггеры в интегральном исполнении имеют также дополнительные асинхронные входы управления S и R. Функции асинхронных входов не зависят от сигналов синхронизации. Отметим, что асинхронные входы имеют и другие типы триггеров. Поскольку дополнительные входы “предустановка” и “очистка”, с помощью которых триггер может быть установлен в нужное состояние независимо от сигналов на других входах, включая синхронизирующий, работают независимо от синхронизации, их называют **асинхронными**.

Входы “предустановки” и “очистки” напоминают соответствующие входы S и R обычного несинхронизируемого RS-триггера. При подаче “1” на вход R и “0” на вход S (рис. 3.14) триггер устанавливается в состояние “0”. При подаче “1” на оба эти входа поведение триггера не определено, т.е. комбинация S=1, R=1 является запрещенной. При подаче на эти входы “1” поведение триггера полностью определяется другими входными сигналами и синхросигналом.

## Т-триггер

Т-триггер - это счетный триггер. Т-триггер имеет один вход (вспомогательные входы принудительной установки "0" и "1" не рассматриваются), куда подают тактирующие (счетные) импульсы. После подачи каждого тактирующего импульса состояние Т-триггера меняется в обратное (инверсное) предыдущему состоянию (аналогично состоянию JK-триггера при комбинации входных переменных  $J=1$  и  $K=1$ ). Т-триггеры строятся только на базе двухступенчатых (RS, D, JK) триггеров.

Т-триггер можно синтезировать из любого типа двухступенчатого триггера. Рассмотрим пример синтеза Т-триггера из JK-триггера. Для этого Т-триггер представим как совокупность комбинационного устройства КУ и JK-триггера (рис. 3.19).

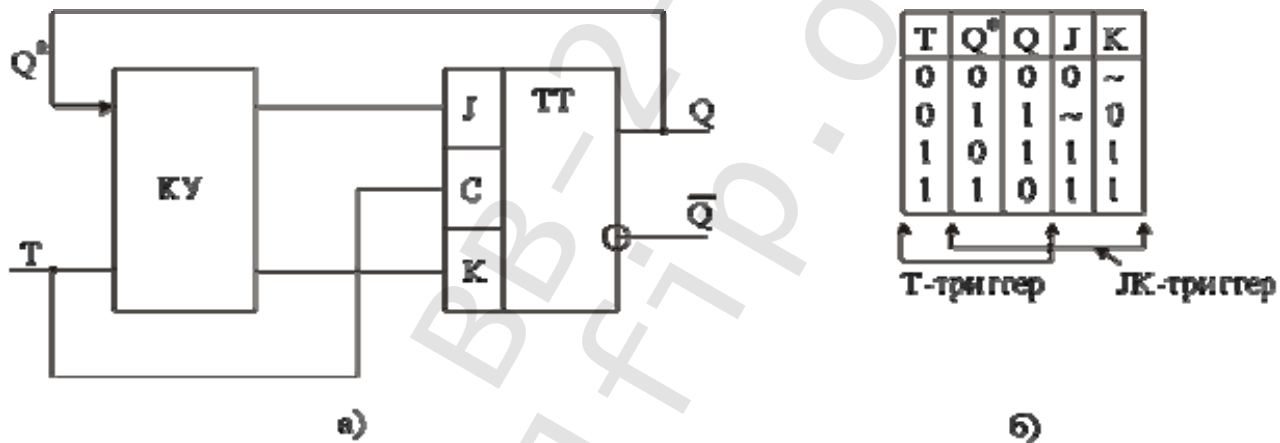


Рис 3.19. Функциональная схема Т-триггера, синтезируемого на базе JK-триггера а) и его состояний б)

Приведенное на рис.3.19, (а) комбинационное устройство должно обеспечить на выходах J и K соответствующие сигналы управления RS-триггером (при подаче на его входы сигналов Q и T), в соответствии с таблицей состояний (рис. 3.19, б). Карты Карно, с помощью которых получены минимальные формы логических выражений для функций J и K, приведены на рис 3.20.

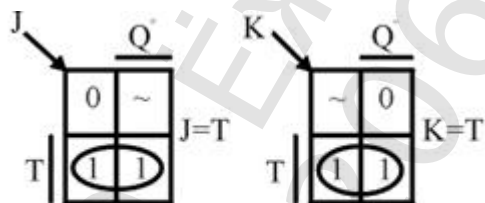


Рис. 3.20. Карты Карно функций J и K



Из полученных логических выражений следует, что для построения Т-триггера, достаточно объединить входы С, J, К JK-триггера, как показано на рис. 3.21.

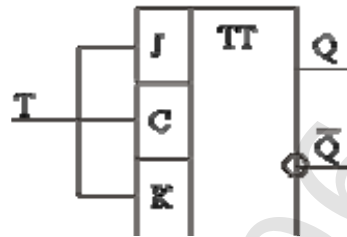


Рис. 3.21. Схема Т – триггера, выполненного на базе JK - триггера

Счетный триггер можно синтезировать и на базе D-триггера. Рассмотрим пример реализации Т-триггера на базе D-триггера, управляемого фронтом синхроимпульса. Совмещенная таблица состояний синтезируемого триггера и D-триггера приведена на рис. 3.22, (а). Карта Карно, используемая для минимизации логического выражения функции, КУ, содержит всего один контур с двумя клетками и, следовательно, функция будет содержать тоже всего одну переменную (рис. 3.22, б).

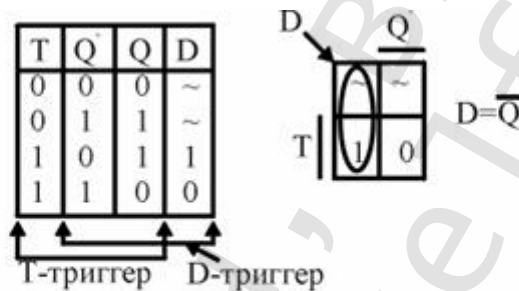


Рис. 3.22. Таблица состояний а) и карта карно б) Т-триггера синтезируемого из D-триггера

Из логического выражения функции **D** следует, что для получения счетного триггера из D-триггера, достаточно соединить инверсный выход D-триггера с его входом "D" (рис. 3.23).

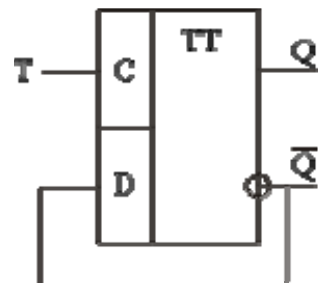


Рис. 3.23. Схема счетного триггера, выполненного на базе D - триггера

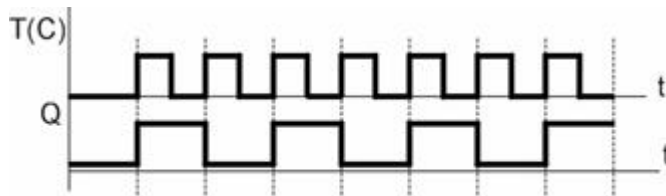


Рис. 3.24. Временные диаграммы счетного D-триггера

Временные диаграммы, поясняющие принцип работы счетного триггера приведены на рис. 3.24. Из временных диаграмм следует, что переключение состояния счетного триггера, выполненного на базе D-триггера, происходит в моменты поступления передних фронтов синхроимпульсов. Частота следования импульсов на выходе счетного триггера в два раза меньше частоты входных синхроимпульсов, что позволяет их использовать в качестве делителей частоты. Если один счетный триггер позволяет делить частоту на два, то для реализации делителя частоты на четыре потребуется два триггера, соединенных последовательно и т.д.

## Регистры

Назначение регистров - хранение и преобразование многоразрядных чисел . Регистры обладают большими функциональными возможностями. Они используются в качестве управляющих и запоминающих устройств генераторов и преобразователей кодов , счетчиков, делителей частоты, узлов временной задержки .

Элементами структуры регистров являются синхронные триггеры D, RS или JK - типа с динамическим или статическим управлением. Так как одиночный триггер может запомнить один разряд (бит) двоичной информации, то его можно считать одноразрядным регистром. Для запоминания многоразрядных чисел используют "линейки " из нескольких триггеров - по количеству разрядов чисел. В схемы регистров входят также комбинационные элементы , роль которых в данном случае вспомогательная .

Занесение информации в регистр называют операцией ввода или считывания. Запись информации в регистр не требует его предварительного обнуления (сброса).

Все регистры в зависимости от функциональных свойств подразделяются на две категории : накопительные регистры (памяти, хранения) и сдвиговые .

В свою очередь сдвиговые регистры делятся: по способу ввода и вывода информации - на параллельные ,последовательные и комбинированные (параллельно-последовательные и последовательно-параллельные); по направлению передачи (сдвига) информации - на однонаправленные и реверсивные.

## Регистры памяти .

Регистры памяти - простейший вид регистров . Их назначение ~ хранить двоичную информацию небольшого объема в течение короткого промежутка времени. Эти регистры представляют собой набор синхронных триггеров , каждый из которых хранит один разряд двоичного числа. Ввод (запись, загрузка) и вывод (считывание) информации производится одновременно во всех разрядах параллельным кодом. Запись обеспечивается тактовым импульсом. С приходом очередного тактового импульса происходит обновление записанной информации.

Сигналы на выходах триггеров характеризуют выходную информацию. Считывание может производиться в прямом или в обратном коде (в последнем случае - с инверсных выходов). Регистры памяти (хранения) представляют собой , по существу, наборы триггеров с независимыми информационными входами и (обычно) общим тактовым входом. В качестве регистров подобного рода могут быть использованы без дополнительных элементов многие типы синхронных триггеров. Особенно пригодны микросхемы, содержащие в одном корпусе несколько самостоятельных триггеров, например: K155TM8, K155TM5, K155TM7, K155TM8 и др., которые можно рассматривать как четырехразрядные регистры памяти.

На рис.1 приведена функциональная схема ИМС K155TM8, которая содержит четыре D-триггера с динамическим управлением. Микросхема имеет 4 отдельных информационных входа, прямые и инверсные выходы Q и  $\bar{Q}$  от каждого триггера и общие для всех триггеров выходы синхронизации и установки нулей (сброса). Запись данных, поступивших на информационные входы, происходит одновременно во всех триггерах по фронту 0,1 сигнала на входе C , при этом на входе R должен быть высокий уровень (логическая 1). Установка триггеров в нулевое состояние выполняется сигналом НИЗКОГО уровня (лог. 0) на входе R. При использовании микросхемы K155TM8 в качестве четырехразрядного регистра памяти она изображается как на рис.2.

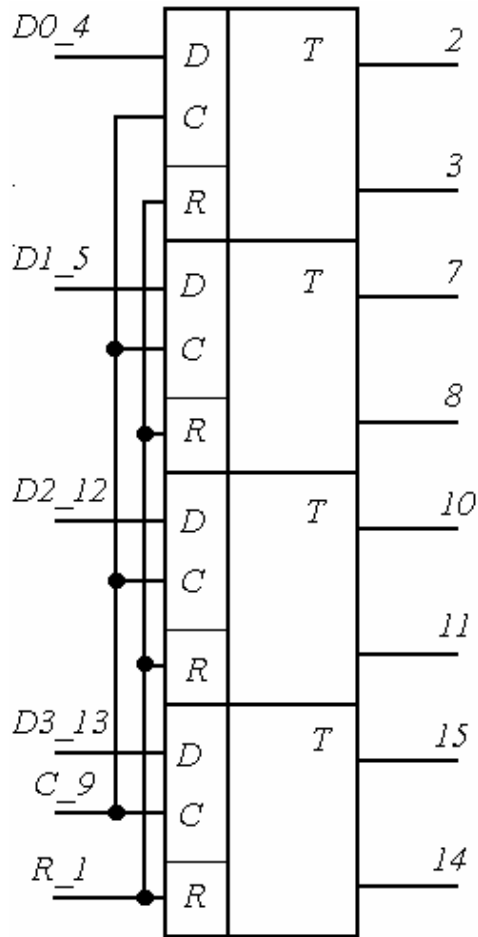


Рис. 1. Функциональная схема ИМС К155ТМ8

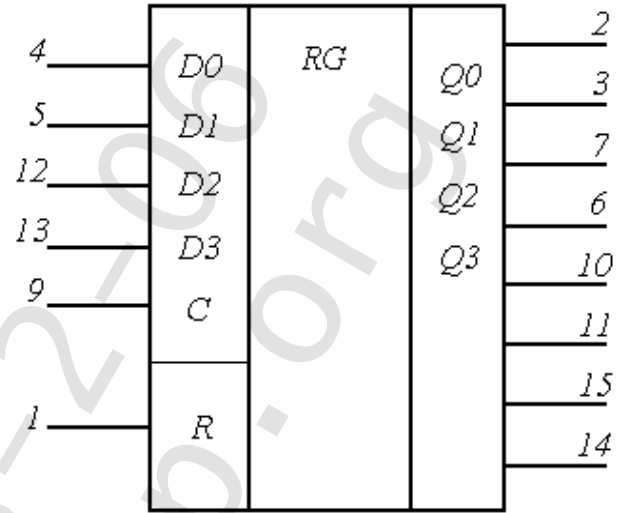


Рис. 2. Условное обозначение регистра памяти на ИМС К155ТМ8.

## Регистровая память .

Микросхемы регистровой памяти содержат несколько регистров. На рис.3 показана структурная схема регистровой памяти 4 x 4 (четыре регистра 4-х - разрядных слов).

Вход загрузки требуемого регистра выбирается дешифратором записи DC на основании адреса записи WA , т. е. кода номера загружаемого регистра. Запись данных, присутствующих на шине DIN, происходит в момент поступления сигнала разрешения записи WE. Выходы регистров мультиплексором MS подключаются к выходной шине DOUT. Номер регистра, с которого происходит чтение, определяет код адреса чтения RA . Выдачу данных в шину DOUT разрешает сигнал RE. Поскольку дешифрация адреса записи и адреса чтения производится двумя независимыми узлами, имеющими автономные адресные входы WA и RA , регистровая память может одновременно записывать число в один из регистров и читать число из другого.

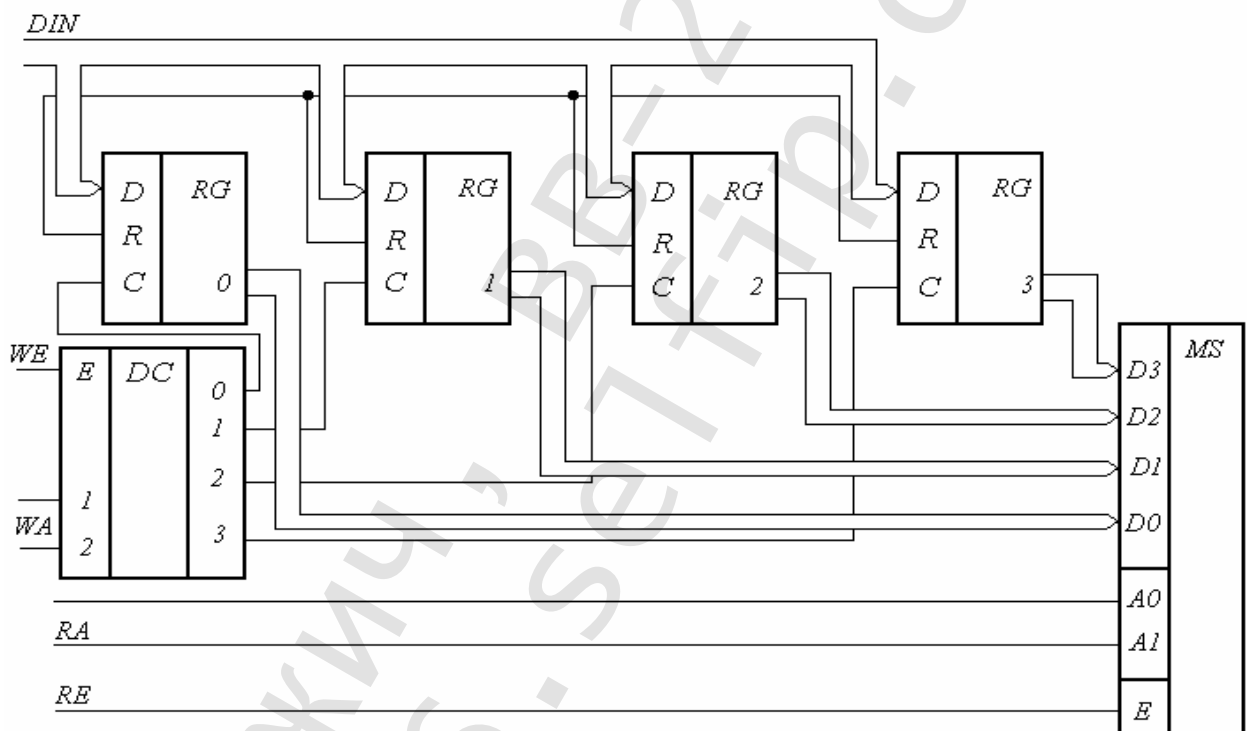


Рис.3. Структурная схема регистровой памяти 4 x 4 .

## Регистры сдвига

Кроме операции хранения данных регистры могут использоваться и для операции сдвига данных с целью преобразования двоичного последовательного кода в параллельный и наоборот. Это так называемые сдвиговые регистры или регистры сдвига, которые получаются путем цепочного соединения триггеров.

Суть сдвига состоит в том, что по сигналу синхроимпульса происходит одновременная перезапись содержимого каждого триггера в соседний триггер. При этом не меняется само двоичное слово (число), записанное в регистре, оно лишь сдвигается на один разряд и только содержимое последнего триггера ТТ3 пропадает из регистра, а на вход первого ТТ0 поступает новый бит(см. рис.1)

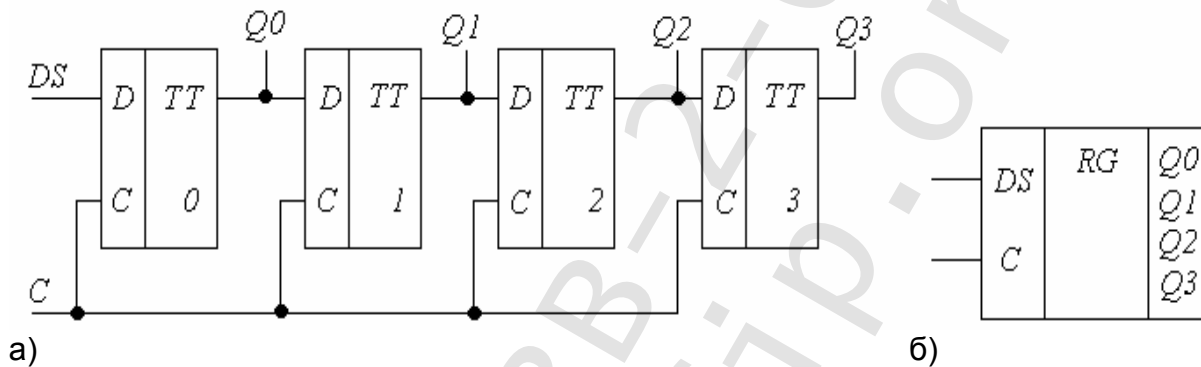


Рис. 1. Сдвигающий регистр

- а) схема на непрозрачных триггерах
- б) условное обозначение

Сдвигающий регистр в отличие от регистра памяти должен обязательно состоять из непрозрачных триггеров, иначе при первом же сигнале сдвига бит, поступивший на вход первого триггера регистра сдвига, сразу же пройдет на выход этого триггера и соответственно, на вход второго триггера, а значит и на выход второго триггера и т. д. до последнего триггера регистра сдвига.

В некоторых сдвиговых регистрах сдвиг происходит не только в одну сторону, а и влево и вправо - это так называемые реверсивные регистры.

Чтобы обеспечить реверс сдвига и параллельную запись сразу во все триггеры к D-входу каждого  $i$ -того триггера подключён мультиплексор, который при подаче " 1 " на один из управляющих входов - SL(Shift Left - сдвиг влево), SR(Shift Right - сдвиг вправо) или PL(Parallel load - параллельная загрузка) подключает вход  $i$ -того триггера соответственно к выходу младшего соседа (направление А), старшего соседа (направление В) или к выходу параллельной загрузки  $D_i$  (см. рис.2). Точка А самого младшего триггера (разряда) является входом DR, точка В самого старшего разряда - входом DL (входы DL и DR используются для наращивания разрядности реверсивных регистров). Выход  $i$ -го триггера подключён к соответствующим входам мультиплексоров соседних разрядов. По С-сигналу триггеры регистра принимают информацию с направлений, диктуемых мультиплексорами.

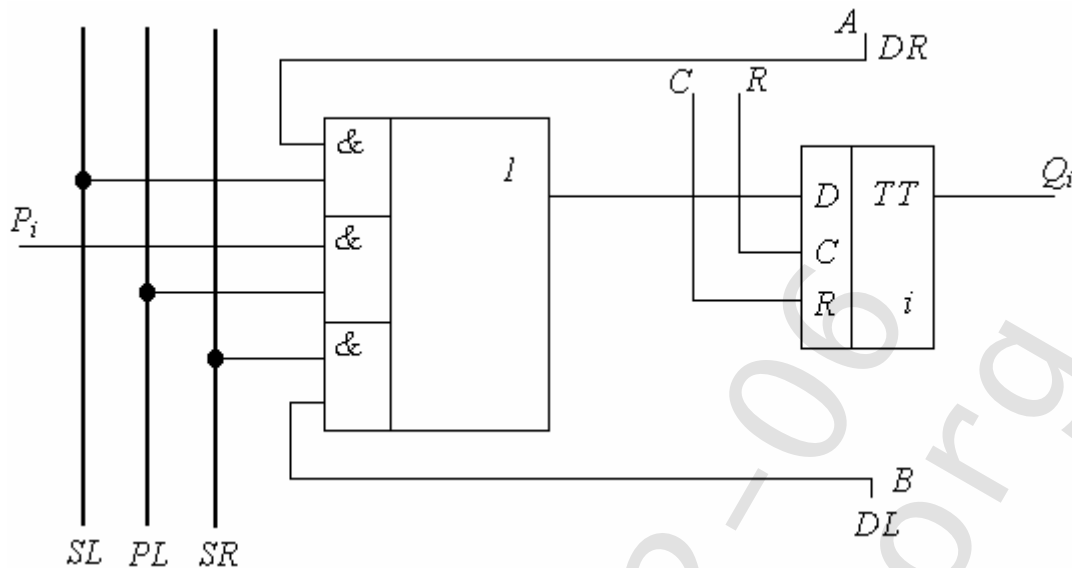


Рис.2 Схема одного разряда реверсивного регистра.

На рис.3 представлена принципиальная схема сдвигового регистра с параллельным и последовательным вводом данных.

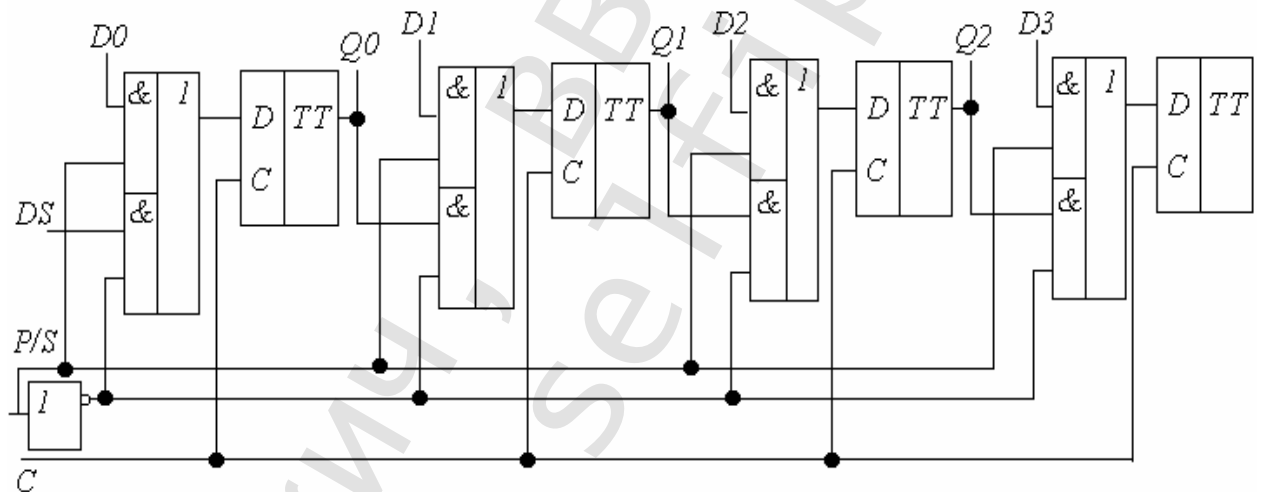


Рис.3 Сдвиговый регистр с возможностью параллельного и последовательного ввода данных.

DS - Data Serial - вход для последовательного ввода данных

P/S - Parallel/Serial - выбор режима работы

Чтобы 4-х разрядное двоичное число, представленное в параллельном коде, преобразовать в последовательный код, надо сначала разряды этого числа подать на входы D0,D1,D2,D3 регистра, а одновременно с этим подать разрешение на параллельную запись, т.е. P/S = 1 и подать один или несколько импульсов на C-вход регистра.

После этого нужно запретить параллельную запись в регистр и тем самым разрешить последовательный режим регистра (P/S = 0). Теперь на каждый синхромпульс выход D3 будет выдавать поочерёдно разряды записанного двоичного числа.

Обычно последовательный код начинается младшим разрядом, а в данном регистре первым на выход поступит разряд числа, записанный ранее на триггер ТТ3, значит при параллельной записи следует проследить, чтобы младший разряд был записан в триггер ТТ3, а старший - в ТТ0.



Рассмотрим схему преобразования параллельного кода в последовательный, приведённую на рис.4. Данные параллельным кодом загружаются в регистр RG1, пока P/S=1. Затем при P/S=0 RG1 переходит в режим сдвига (последовательный режим) и в течение 4-х импульсов на C- входе RG1 содержимое этого регистра с его выхода Q3 разряд за разрядом передаётся на RG2 на вход DS, где записывается последовательным кодом благодаря подаче 4-х синхроимпульсов на вход C RD2. После этого данные могут быть считаны параллельным кодом с выходов Q0-Q3 регистра RG2.

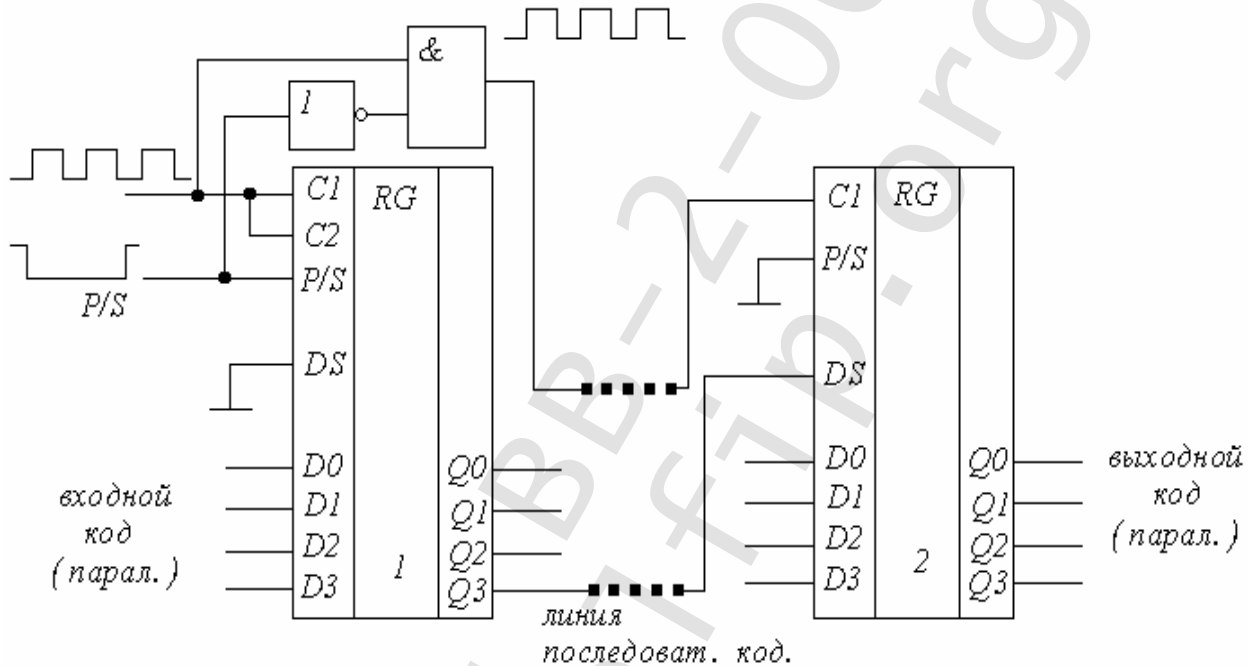


Рис.4 Преобразование параллельного кода в последовательный

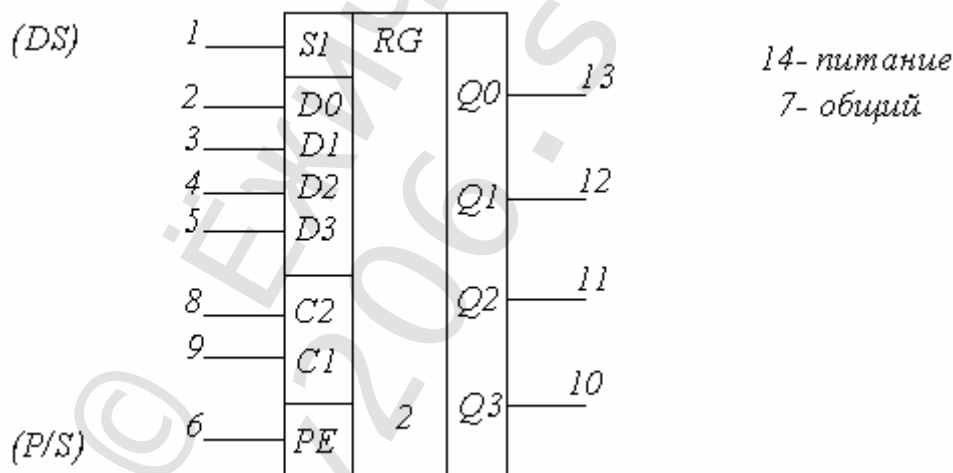


Рис.5 Условное обозначение и цоколевка микросхемы K155IP1

### Кольцевые счетчики.

На базе регистров сдвига можно построить кольцевые счетчики - счетчики Джонсона. Счетчик Джонсона имеет коэффициент пересчета, вдвое больший числа составляющих его триггеров. В частности, если счетчик состоит из трех триггеров ( $m=3$ ), то он будет иметь шесть устойчивых состояний. Счетчик Джонсона используется в системах автоматики в качестве распределителей импульсов и т.д.

Таблица состояний счетчика Джонсона (рис. 3.29) содержит  $2m$  ( $m$  - количество триггеров в составе регистра) строк и  $m$ -столбцов. Количество разрядов счетчика определяется количеством триггеров (рис. 3.29). Рассмотрим схему трехразрядного счетчика Джонсона, выполненного на базе D-триггеров (регистр сдвига реализован на D-триггерах). Для построения кольцевого счетчика достаточно соединить инверсный выход последнего триггера регистра (последнего разряда) с входом "D" (с входом, предназначенным для ввода последовательной информации) первого триггера.

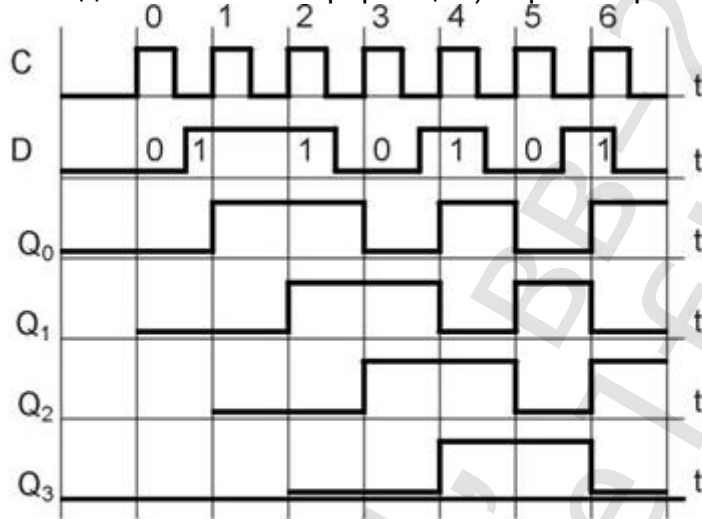


Рис. 3.28. Временные диаграммы, поясняющие работу регистра сдвига

а)

N	$Q_2$	$Q_1$	$Q_0$
1	0	0	1
2	0	1	1
3	1	1	1
4	1	1	0
5	1	0	0
6	0	0	0

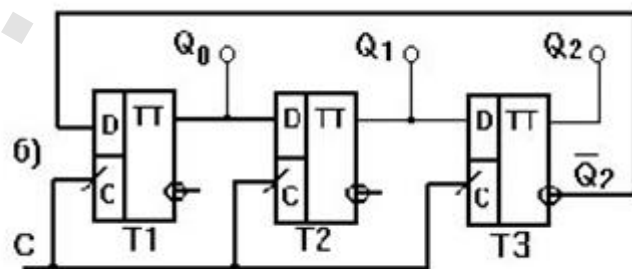


Рис. 3.29. Таблица состояний а) и схема б) счетчика Джонсона на трехразрядном регистре сдвига

Предположим, что вначале все триггеры находятся в состоянии "0", т.е.  $Q_0=Q_1=Q_2=0$ . При этом на входе "D" первого триггера присутствует уровень "1", т.к.  $\bar{Q}_2=1$ . Первым синхроимпульсом в триггер T1 запишется "1", вторым - единица запишется в первый триггер, из первого - во второй и т.д. до тех пор, пока на всех выходах регистра не будет "1". После заполнения регистра единицами, на

инверсном выходе триггера Т3 появится  $\bar{Q}_3=0$  и четвертым синхроимпульсом в Т1 запишется логический "0" (рис. 3.29, б).

После поступления последующих трех синхроимпульсов регистр обнуляется и на его вход "D" снова подается уровень "1". Таким образом, цикл повторения состояния кольцевого счетчика состоит из шести тактов синхросигнала. Как видим, при работе в начале от первого триггера до последнего триггера распространяется "волна единиц", а затем "волна нулей". Код, в котором работает счетчик **Джонсона**, называют кодом **Либау-Крейга**.

### Генераторы чисел.

На базе кольцевых счетчиков можно реализовать генераторы различных двоичных чисел. Вывод генерируемых чисел можно осуществлять как в параллельном, так и в последовательном коде.

В качестве примера рассмотрим работу трехразрядного (очевидно, что количество разрядов может быть и более трех) генератора чисел, реализованного на базе D-триггеров. Генератор чисел представим как совокупность трехразрядного регистра сдвига и комбинационного устройства КУ, выходной сигнал которого служит "источником" информации для регистра сдвига (рис. 3.30). Входными переменными КУ являются выходные сигналы разрядных триггеров регистра сдвига  $Q_0, Q_1, Q_2$ . Цикл повторения чисел определяется  $2^m$  - тактами синхросигнала (максимальный цикл). В таблице состояний генератора чисел использованы (рис. 3.31, а) следующие обозначения:  $N_n$  - десятичный эквивалент двоичного числа, реализуемого в параллельном коде; D - функция аргументов  $Q_0, Q_1, Q_2$ . Эту функцию можно рассматривать как последовательный код. Для получения минимальной дизъюнктивной формы записи функции D построим карту Карно (рис. 3.31, б).

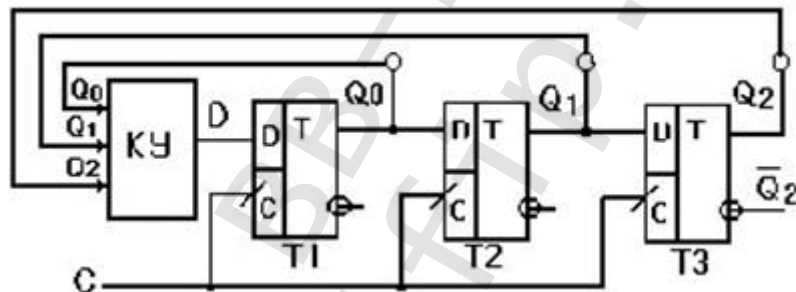
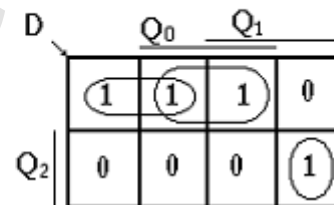


Рис. 3.30. Структурная схема трехразрядного генератора чисел на D - триггерах

$N_n$	$Q_2$	$Q_1$	$Q_0$	D
0	0	0	0	1
1	0	0	1	1
3	0	1	1	1
7	1	1	1	0
6	1	1	0	1
5	1	0	1	0
2	0	1	0	0
4	1	0	0	0



$$D = \bar{Q}_2 \cdot \bar{Q}_1 + \bar{Q}_2 \cdot Q_0 + Q_2 \cdot Q_1 \cdot \bar{Q}_0$$

Рис. 3.31. Таблица состояний а) и карта Карно б) трехразрядного генератора чисел

Схема комбинационного устройства содержит три элемента конъюнкции и один элемент дизъюнкции (рис. 3.32). При построении схемы КУ дополнительные инверторы не потребуются, т.к. триггеры регистра имеют как прямые, так и инверсные выходы.

Использование регистров сдвига в кольцевых схемах представляет практический интерес, так как его структурные схемы состоят из повторяющихся триггерных цепей, обладают наращиваемостью и пригодны для различных применений.

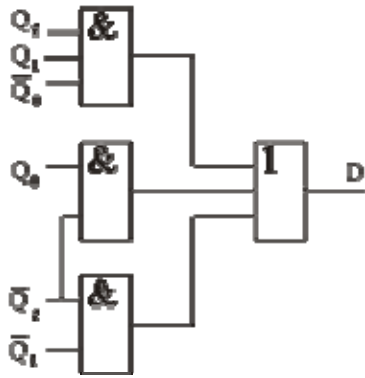


Рис. 3.32. Схема комбинационного устройства - формирователя управляющих сигналов генератора чисел

© Ёжич, ВВ-2-06  
[vv206.selfip.org](http://vv206.selfip.org/)

## Счетчики

На базе счетных триггеров можно построить цифровое устройство, получившее название **электронного счетчика**. Электронные счетчики (далее, просто счетчики) позволяют вести подсчет электрических импульсов, количество которых (поступивших на вход счетчика) представляется, обычно, в параллельном коде. Счетчики могут отличаться модулем счета и типом счетной последовательности, которая, в частности, может быть двоичной, двоично-десятичной, в коде Грея и т.п. Цифровые последовательностные устройства, выполненные по схеме счетчика, но имеющие один счетный вход и один выход называются **делителями** частоты. Таким образом, любой счетчик может служить в качестве делителя частоты, если используется информация только одного из его выходов. Так как счетчики и делители имеют единую структуру, основное внимание будет уделено синтезу счетчиков.

Счетчики и делители подразделяются на **асинхронные** и **синхронные**. У синхронных счетчиков все разрядные триггеры синхронизируются параллельно одними и теми же синхроимпульсами, поступающими из источника этих импульсов. Асинхронные счетчики имеют последовательную синхронизацию, т.е. каждый последующий разрядный триггер синхронизируется выходными импульсами триггера предыдущего разряда. Асинхронные счетчики иногда называют **последовательными**, а синхронные счетчики - **параллельными**.

**Синхронные счетчики**, в свою очередь, подразделяются на параллельно-синхронные и последовательно-синхронные. Параллельные счетчики имеют более высокую скорость счета, чем асинхронные.

**Счетчики**, независимо от способа синхронизации, подразделяются на счетчики прямого счета (суммирующие) и на счетчики обратного счета (вычитающие). В интегральном исполнении выпускаются также реверсивные счетчики, в которых имеется специальный вход для переключения режима работы, т.е. направления счета. Многие типы счетчиков, выпускаемые промышленностью в интегральном исполнении, имеют дополнительные входы предустановки, позволяющие использовать эти счетчики в режиме регистра памяти.

В качестве разрядных триггеров счетчиков и делителей могут быть использованы двухступенчатые D-триггеры, T- и JK-триггеры.

Счетчики относятся к последовательностным устройствам с циклически повторяющейся последовательностью состояний. Число, соответствующее количеству импульсов (поступивших на вход счетчика), при котором счетчик "возвращается" в исходное состояние, называется **модулем** или **коэффициентом** счета. Модуль счета, обычно, обозначают буквой **M** (или  $K_{сч}$ ). Например, максимальный модуль счета счетчика из двух триггеров равен  $M = 2^2 = 4$ , трех триггеров -  $M = 2^3 = 8$  и т.д. В общем случае для  $n$  - разрядного счетчика -  $M = 2^n$ . Модуль счета счетчика численно совпадает с модулем деления делителя частоты. Счетчик по модулю 8 позволяет реализовать (без дополнительных схемных затрат) делитель частоты на 8. Это значит, что данный делитель делит частоту входной импульсной последовательности на 8.

### Асинхронный счетчик.

Асинхронный двоичный счетчик представляет собой совокупность последовательно соединенных триггеров (D - или JK ), каждый из которых ассоциируется с битом в двоичном представлении числа. Если в счетчике  $m$  триггеров, то число возможных состояний счетчика равно  $2^m$ , и, следовательно, модуль счета  $M$  также равен  $2^m$ . Счетная последовательность в двоичном суммирующем счетчике начинается с нуля и доходит до максимального числа  $2^m - 1$ , после чего снова проходит через нуль и повторяется. В вычитающем двоичном счетчике последовательные двоичные числа перебираются в обратном порядке, и при повторении последовательности максимальное число следует за нулем.

Рассмотрим устройство двоичного суммирующего счетчика по модулю  $M=16$ , выполненного на базе JK-триггеров (рис. 3.33, а).

Как видно из рис. 3.33, (а), синхронизирующие входы всех триггеров, кроме крайнего левого (Т1), соединены с выходами предыдущих триггеров. Поэтому состояние триггера меняется в ответ на изменение состояния предыдущего триггера.

Из таблицы состояния счетчика (рис. 3.33, б) легко заметить, что значение разряда в выбранной позиции меняется тогда, когда в соседней справа позиции состояние переходит из "1" в "0", управление триггерами осуществляется задним фронтом синхроимпульсов (отрицательным перепадом напряжения импульса синхронизации).

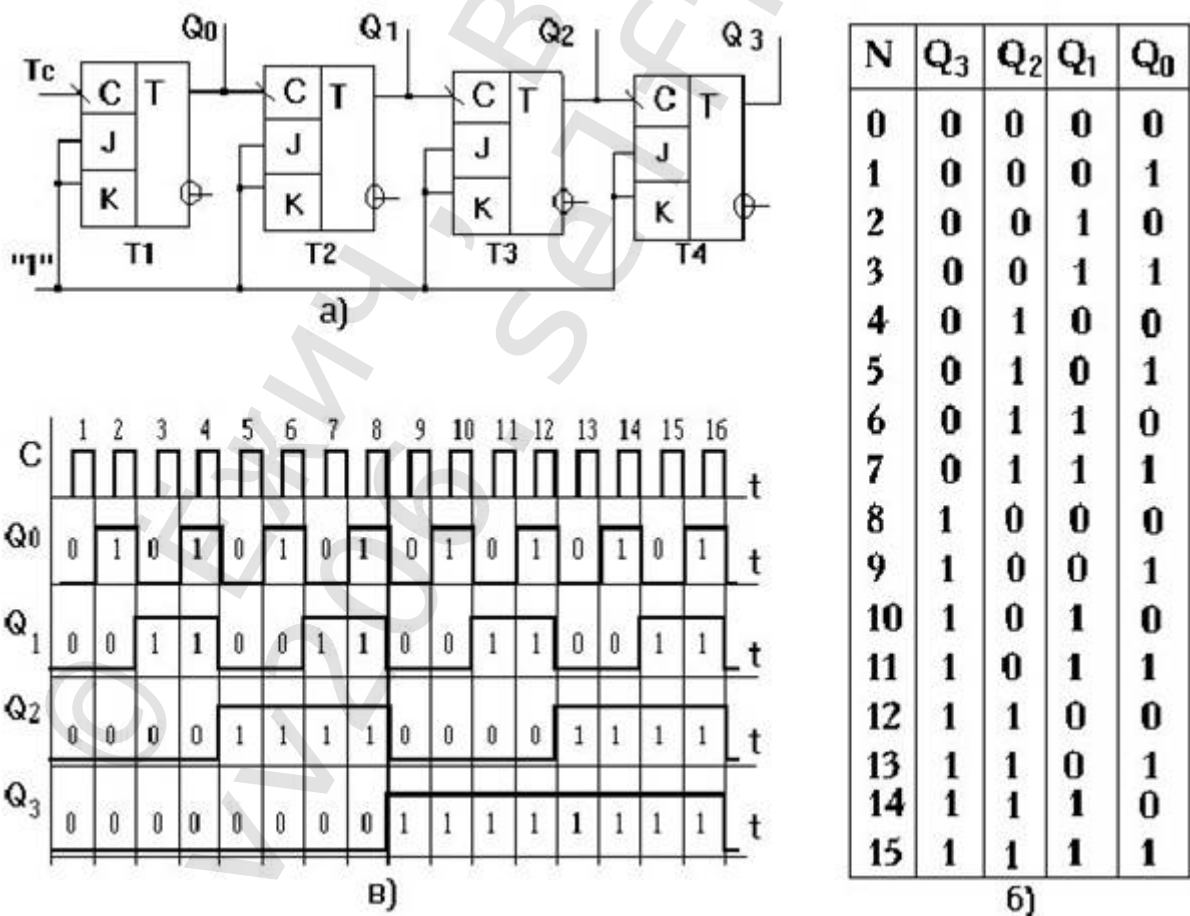


Рис. 3.33. Схема а), таблица состояний триггеров б) и временные диаграммы, поясняющие работу в) последовательного четырехразрядного счетчика на JK - триггерах

Временные диаграммы, поясняющие работу асинхронного суммирующего счетчика приведены на рис. 3.33, в.

© Ёжич, ВВ-2-06  
[vv206.selfip.org](http://vv206.selfip.org/)



### Параллельные счетчики (синхронные счетчики).

Как было уже сказано выше, параллельные счетчики бывают двух типов: синхронные параллельные и синхронные последовательные.

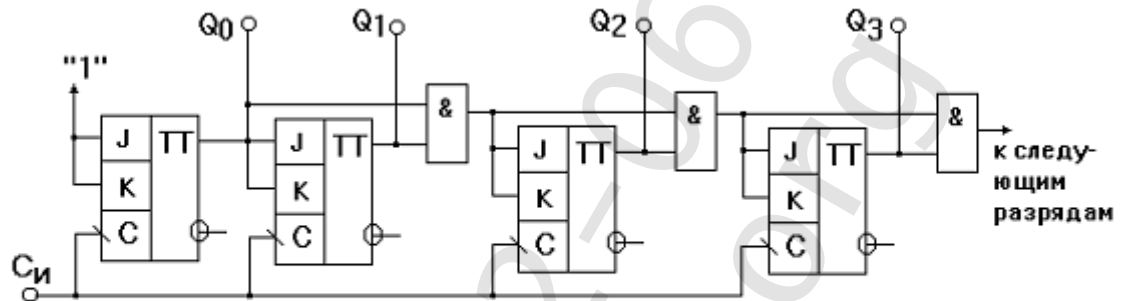


Рис. 3.35. Синхронный последовательный суммирующий счетчик на JK - триггерах

**Синхронный последовательный счетчик.** По способу подачи синхроимпульсов такие счетчики параллельные, т.е. синхроимпульсы поступают на все триггеры счетчика параллельно, а по способу управления (подачи управляющих импульсов) - последовательные. Схема синхронного последовательного счетчика, реализованного на JK-триггерах, приведена на рис. 3.35.

Синхронный последовательный счетчик обладает повышенным быстродействием, однако, за счет последовательного формирования управляющих уровней, на входы "J" и "K" счетных триггеров, быстродействие несколько уменьшается. От этого недостатка лишены параллельные синхронные счетчики, в которых формирование управляющих уровней и их подача на соответствующие входы триггеров счетчика осуществляется одновременно, т.е. параллельно. Пример реализации параллельного синхронного счетчика иллюстрирован на рис. 3.36.

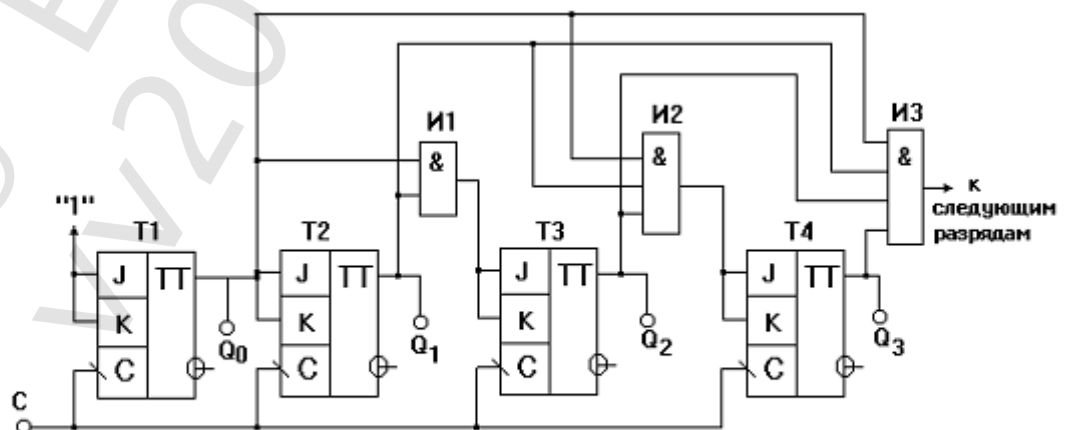


Рис. 3.36. Параллельный синхронный счетчик на JK - триггерах

Поскольку счетчик имеет одну общую линию синхронизации, состояние триггеров меняется синхронно, т.е. те триггеры, которые по синхроимпульсу должны изменить свое состояние, делают это одновременно, что существенно повышает быстродействие синхронных счетчиков.

© Ёжич, ВВ-2-06  
vv206.selfip.org

### Счетчики обратного счета (вычитающие счетчики).

На рис. 3.34 приведена схема асинхронного трехразрядного двоичного вычитающего счетчика, построенного на базе D-триггеров. Отметим, что условия для изменения состояний триггеров вычитающих счетчиков аналогичны условиям для суммирующих счетчиков с той лишь разницей, что они должны “опираться” на значения инверсных, а не прямых выходов триггеров. Следовательно, рассмотренный выше счетчик можно превратить в вычитающий, просто переключив входы “С” триггеров с выходов Q на выходы  $\bar{Q}$ . Когда в качестве разрядных триггеров используются D-триггеры, синхронизируемые передним фронтом синхроимпульсов, для получения вычитающего счетчика (асинхронного) входы “С” последующих триггеров соединяются с прямыми выходами предыдущих, также как в счетчике прямого счета, построенного на JK-триггерах.

Работа вычитающего счетчика на D-триггерах наглядно иллюстрирована на рис. 3.34, (б). Из рис. 3.34 следует, что после нулевого состояния всех триггеров, с приходом первого синхроимпульса они устанавливаются в состояние “1”. Поступление второго синхроимпульса приводит к уменьшению этого числа на одну единицу и т.д. После поступления восьмого импульса, снова, все триггеры обнуляются и цикл счета повторяется, что соответствует модулю M=8.

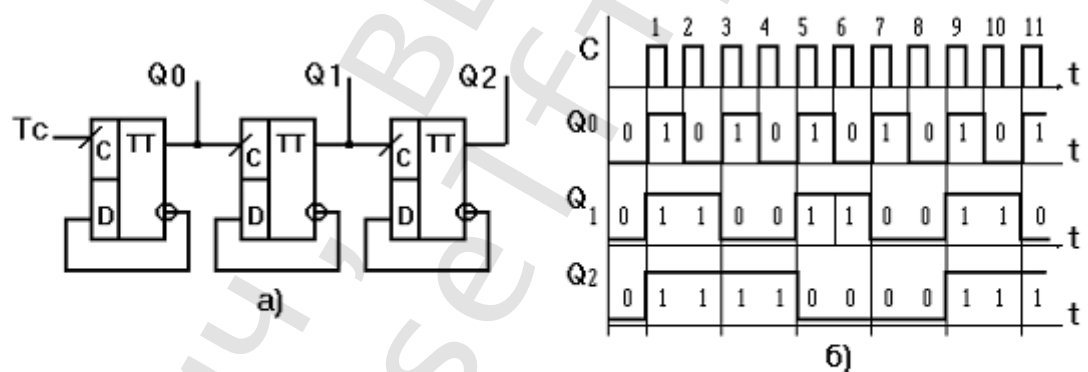


Рис. 3.34. Схема а) и временные диаграммы вычитающего трехразрядного счетчика на D - триггерах

В некоторых случаях необходимо, чтобы счетчик мог работать как в прямом, так и в обратном направлении счета. Такие **счетчики** называются **реверсивными**. Реверсивные счетчики могут быть как асинхронного, так и синхронного типа. Они строятся путем применения логических коммутаторов (мультиплексоров) в цепях связи между триггерами. Так, например, асинхронный реверсивный двоичный счетчик можно построить, если обеспечить подачу сигналов с прямого (при суммировании) или с инверсного (при вычитании) выхода предыдущего JK- или T-триггера на счетный вход последующего. В случае, когда реверсивный счетчик строится на базе D-триггеров, управляемых передним фронтом, для получения режима прямого счета следует соединить инверсный выход предыдущего с счетным входом последующего триггера.

Все рассмотренные типы счетчиков могут быть использованы в цифровых устройствах “умеренного” быстродействия, когда частота следования синхроимпульсов не превышает критического значения, при котором время задержки установки триггеров последних (старших) разрядов счетчика становится соизмеримым с длительностью периода входных тактовых импульсов. В связи с

этим, асинхронные счетчики строятся на относительно небольшое количество разрядов, так как при большем количестве разрядов выходные сигналы триггеров старших разрядов появляются позднее, чем управляющие фронты синхроимпульсов (поступающих на вход первого триггера) .

© Ёжич , ВВ-2-06  
vv206.selfip.org

### Счетчики с произвольным коэффициентом счета.

Принцип построения подобного класса счетных устройств состоит в исключении нескольких состояний обычного двоичного счетчика, являющихся избыточными для счетчиков с коэффициентом пересчета, отличающимися от двоичных. При этом избыточные состояния исключаются с помощью обратных связей внутри счетчика.

Число избыточных состояний для любого счетчика определяется из следующего выражения:

$$M = 2^m - K_{сч},$$

где  $M$  - число запрещенных состояний,  $K_{сч}$  - требуемый коэффициент счета;  $2^m$  - число устойчивых состояний двоичного счетчика.

Задача синтеза счетчика с произвольным коэффициентом счета заключается в определении необходимых обратных связей и минимизации их числа. Требуемое количество триггеров определяется из выражения

$$n = \lceil \log_2 K_{сч} \rceil,$$

где  $\lceil \log_2 K_{сч} \rceil$  - двоичный логарифм заданного коэффициента пересчета  $K_{сч}$ , округленный до ближайшего целого числа.

В каждом отдельном случае приходится применять какие-то конкретные методы получения требуемого коэффициента пересчета. Существует несколько методов получения счетчиков с заданным коэффициентом пересчета  $K_{сч}$ . Один из этих методов заключается в немедленном сбросе в "0" счетчика, установившегося в комбинацию, соответствующему числу  $K_{сч}$ . Его называют также **методом автосброса**. Рассмотрим пример реализации счетчика с  $K_{сч}=10$  методом автосброса. Очевидно, что "сбрасывая" двоичный четырехразрядный счетчик на нуль каждый раз, когда он будет принимать состояние 1010, можно обеспечить "возврат" счетчика в исходное состояние после каждых десяти импульсов. Подобный прием удобно применять при использовании счетчиков в интегральном исполнении, имеющих ячейки конъюнкции (**И**) на входах установки в нуль, как это сделано в микросхеме К1533ИЕ5. В данном примере (рис. 3.37) организованы соединения, обеспечивающие коэффициент пересчета  $K_{сч} = 10$ .

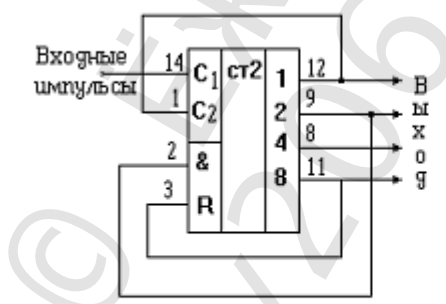


Рис. 3.37. Пример реализации счетчика с  $K_{сч}=10$

Таблица 3.1

К1533ИЕ5	Коэффициенты пересчета					
	3	5	6	9	10	12
Вход	14	14	14	14	14	14
Выход	9,12	8,9,12	8,9,12	все	все	все
Соединения	1-12	1-12	1-12	1-12	1-12	1-12
выводов	2-12	2-12	2-9	2-12	2-9	2-8
	3-9	3-8	3-8	3-11	3-11	3-11

Как следует из рис. 3.37, роль ячейки, выявляющей факт достижения кодовой комбинации 1010 на выходах счетчика, играет ячейка **И**, уже имеющаяся на входе сброса ИМС К1533ИЕ5.

В таблице 3.1 поясняются конфигурации соединений для получения различных коэффициентов пересчета с помощью счетчика К1533ИЕ5. Наиболее

очевидные варианты получения коэффициентов (2, 4, 8, 16 ) в таблице не указаны. В графе “Соединения” таблицы указано, какие выводы микросхемы должны быть соединены между собой: например, указание 1-12 означает, что нужно соединить вывод 1 с выводом 12. В строках “Ввод” и “Выход” таблицы указаны номера выводов микросхемы, на которые следует подавать входные импульсы и с которых надлежит снимать выходные, соответственно. Следует отметить, что ИМС К1533ИЕ5 состоит из четырех счетных триггеров, один из которых имеет отдельные выводы входа и выхода, а остальные три триггера соединены последовательно по схеме асинхронного счетчика.

**Синтез счетчика с произвольным коэффициентом счета.** Один из методов проектирования счетчиков с заданным коэффициентом счета заключается в построении таблицы переходов, в первых столбцах которых будут отражены текущие состояния триггеров счетчика, а в последующих - следующие за ними состояния. Анализ таблицы позволяет установить те переходы, которые должны быть “сделаны” триггерами, входящими в состав счетчика. Затем с помощью управляющей таблицы соответствующего триггера находят значения логических функций на управляющих входах триггеров, позволяющие осуществить эти переходы.

Рассмотрим пример синтеза синхронного двоично-десятичного счетчика на базе JK-триггеров. На рис. 3.38 показан граф, поясняющий последовательность переходов десятичного счетчика, в таблице 3.2 - таблица переходов.

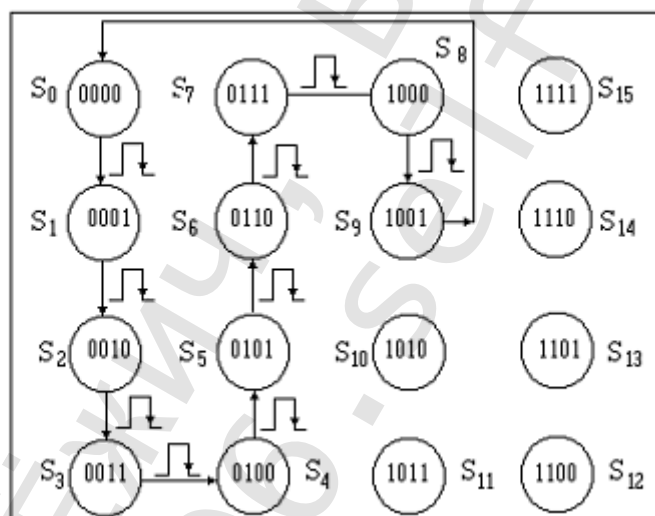


Рис. 3.38. Граф переходов двоично-десятичного счетчика

В правой части таблицы 3.2 приведены значения входных сигналов четырех триггеров. Для поиска этих значений должны быть проанализированы реализованные переходы, а затем с помощью управляющей таблицы определены соответствующие значения “J” и “K” входов триггеров.

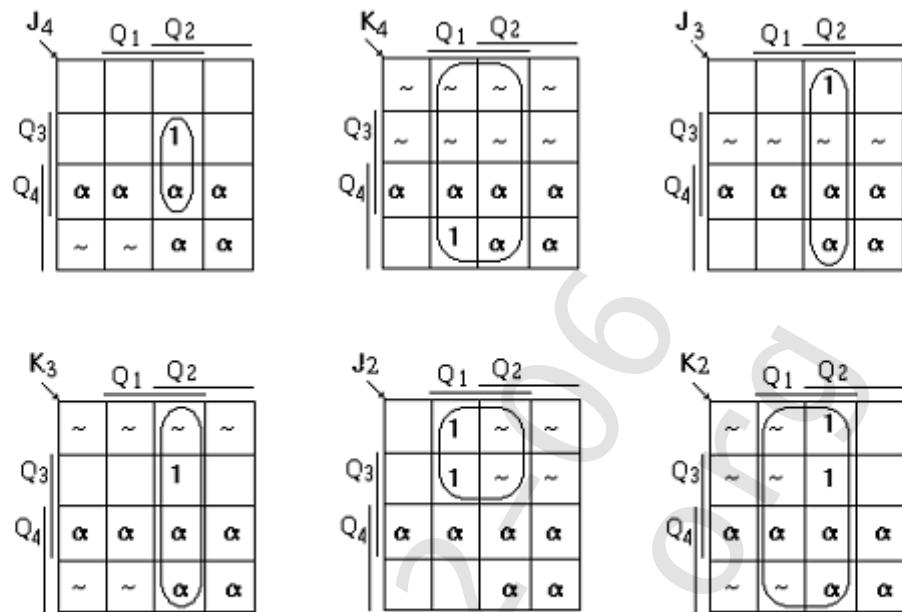


Рис. 3.39. Карты Карно для функций комбинационной логики двоично-десятичного счетчика

На рис.3.39 приведены карты Карно для логических функций, которым должны соответствовать сигналы, присутствующие на управляющих входах триггеров ( нулевые значения функций в клетки карты Карно не записаны).

После упрощения с помощью карт Карно полученные логические выражения, используемые для управления входами “J” и “K”, выглядят

$$\begin{aligned} J_4 &= Q_1 Q_2 Q_3 ; & K_4 &= Q_1 \\ J_3 &= Q_1 Q_2 ; & K_3 &= Q_1 Q_2 ; \\ J_2 &= \overline{Q_4} Q_1 & K_2 &= Q_1 \end{aligned}$$

Просмотр столбцов J<sub>1</sub> и K<sub>1</sub> в табл. 3.2 показывает, что все значения либо “~”, либо “1”. Так как безразличные состояния могут также участвовать в процессе упрощения, то все клетки карты Карно для J<sub>1</sub> и K<sub>1</sub> оказываются заполненными символами “~”, “1” и “α”. Следовательно,

$$J_1 = K_1 = 1$$

На рис. 3.40 показана схема двоично-десятичного синхронного счетчика.

Если счетчик из-за какой-либо неисправности окажется в одном из запрещенных (неиспользуемых) состояний, то его работа может быть прервана специальным сигналом и также может быть подан сигнал тревоги о неисправности в схеме счетчика. Обнаружить это позволяет схема, реализующая выражение, описывающее функцию неиспользуемых состояний

$$f_H = Q_2 Q_4 + Q_3 Q_4 .$$

На рис. 3.41 показано, как эта схема используется для формирования цепи аварийной сигнализации и генерации блокирующего сигнала синхронизации.

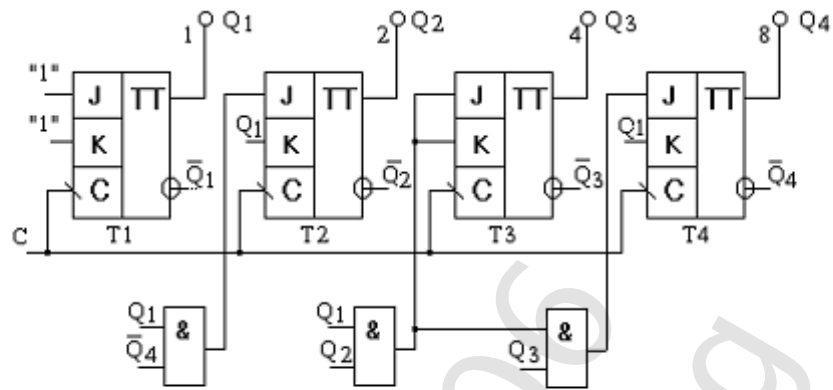


Рис. 3.40. Схема реализации двоично-десятичного синхронного счетчика

Выражение , описывающее блокирующий сигнал синхронизации, имеет вид

$$C' = \bar{f}_n \cdot C$$

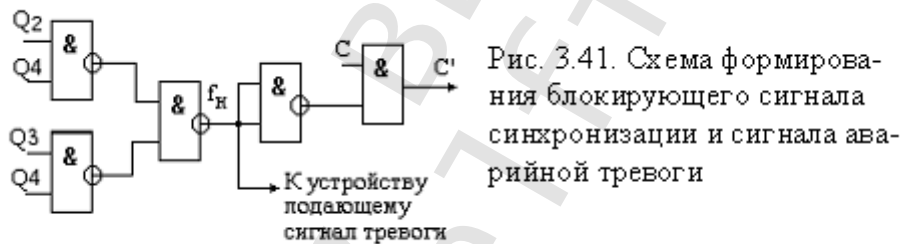


Рис. 3.41. Схема формирования блокирующего сигнала синхронизации и сигнала аварийной тревоги

Следовательно, когда  $f_n = 1$ , то  $C' = 0$ , и синхроимпульсы будут отсутствовать до тех пор, пока счетчик не выйдет из запрещенного состояния.

Из схемы формирования блокирующего сигнала синхронизации следует, что логика её функционирования ориентирована на то, чтобы исключить возможность появления неиспользуемых комбинаций выходных сигналов. Действительно, в коде числа двоично-десятичного счетчика отсутствуют комбинации 0110, 0011, следовательно, их появление свидетельствует о неисправности системы.

Временные диаграммы счетчика (рис. 3.40), заданного графом переходов (рис. 3.38) и таблицей переходов 3.2, приведены на рис. 3.42.



### Формирователь коротких импульсов с применением линий задержки.

Формирователь коротких импульсов формирует импульсы, длительность которых существенно меньше длительности исходных импульсов. Для построения схемы формирователя потребуются один элемент конъюнкции, один инвертор и линия задержки. Длительность выходного импульса формирователя определяется длительностью времени задержки линии задержки  $\Delta t_3$  и средним временем распространения сигнала через инвертор  $t_{3\text{ срЭ1}}$ . На рис. 4.1. приведена схема формирователя, а на рис.4.2 (а) и (б) - временные диаграммы, иллюстрирующие её работу. Из рис 4.2 (а) следует, что для формирования импульса от переднего фронта (исходного импульса) необходимо подавать на линию задержки инвертированный импульс.

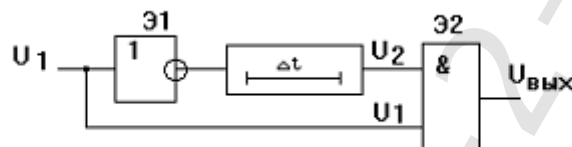


Рис. 4.1. Формирователь коротких импульсов с линией задержки

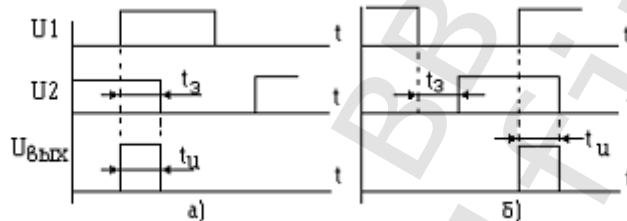


Рис. 4.2. Временные диаграммы, иллюстрирующие работу формирователя коротких импульсов

В случае формирования импульса от заднего фронта нужно инвертировать незадержанный (прямой) сигнал, т.е. сигнал, подаваемый на элемент "И" минуя линию задержки (рис. 4.2, б).



Рис. 4.3. Формирователь коротких импульсов с использованием элементов логики

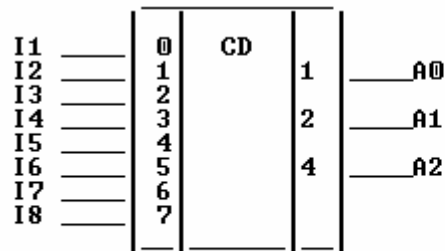
Использование в формирователях линий задержки не всегда оправдано экономически и из конструктивных соображений. Если не требуется формирование строго определенной длительности коротких импульсов, в формирователях в качестве **линии задержки применяются логические элементы** (рис. 4.3). Так как каждый логический элемент обладает свойством задерживать распространение сигнала, поэтому время задержки в такой схеме будет определяться числом используемых элементов логики  $n$

$$\Delta t_3 = t_{3\text{ срЭ1}} + t_{3\text{ срЭ2}} + \dots + t_{3\text{ срЭn}} = n t_{3\text{ срЭ}},$$

где  $t_{3\text{ срЭ}}$  - среднее время задержки одного логического элемента. Считается, что инвертор имеет значительно меньшее время задержки сигнала, и в качестве элементов задержки используются логические элементы с малым быстродействием.

## Шифратор

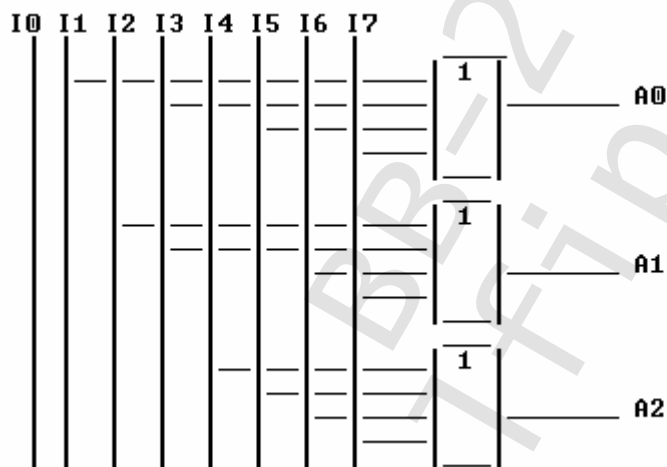
Шифратор (coder) выполняет функцию преобразования унарного кода в двоичный. При подаче сигнала на один из входов (обязательно на один, не более) на выходе появляется двоичный код номера активного входа.



Условное обозначение шифратора.

I	A2	A1	A0
I0	0	0	0
I1	0	0	1
I2	0	1	1
I3	0	1	0
I4	1	0	0
I5	1	0	1
I6	1	1	0
I7	1	1	1

Таблица состояний шифратора.



Структурная схема шифратора.

Приоритетный шифратор отличается от шифратора наличием дополнительной логической схемы выделения активного уровня старшего входа для обеспечения условия работоспособности шифратора (только один уровень на входе активный). Уровни сигналов на остальных входах схемой игнорируются. Схема выделения строится на конъюнкторах и инверторах таким образом, чтобы любое число старших нулей, образуя после инверторов логические единицы, не влияло на работу конъюнкторов в цепи младших входов. Любая самая старшая единица (после инвертора -логический ноль) запирает конъюнкторы младших входов.

## Приоритетный шифратор (PRCD).

Выпускаются микросхемы приоритетных шифраторов, в которых условие влияния только одной входной лог.1 на схему шифратора обеспечивается дополнительной схемой приоритета на его входе. В приоритетном шифраторе входной код может содержать сколько угодно единиц, но выходной код шифратора будет соответствовать номеру того входа, на который полагается старшая единица. Это не обязательно будет старший разряд входного кода. Приоритет – преимущественное право.

На входе приоритетного шифратора стоит схема выделения старшей единицы, в которой все входные единицы, кроме старшей, заменяются нулями.

На схеме выделения старшей единицы из трехразрядного входного кода  $X_2, X_1, X_0$ , где  $X_0$  – младший разряд, вход EI (ENABLE INPUT) предназначен для подачи разрешения или запрета на всю эту схему, а выход EO (ENABLE OUTPUT) – для формирования сигнала разрешения работы такой же схемы на разряды меньше, чем  $X_0$ .

Эти два сигнала EI и EO позволяют собрать цепочку (столбец) из нескольких схем выделения старшей 1, если трех разрядов недостаточно.

В микросхемах, где есть вход EO он обозначает вход разрешения третьего Z состояния выхода этой микросхемы, а где EO является входом, то есть показан на правой выходной стороне условного обозначения микросхемы, там этот выход EO предназначен для соединения цепочкой нескольких типа приоритетного шифратора.

Работа схемы: нули в старших разрядах входного кода поражают на выходах ЛЭ – Е этих разрядов единицы и никак не влияют на работу элементов – Е более младших разрядов. Любая единица во входном коде порождает на выходе ЛЭ – Е этого разряда нулевой уровень, который, поступая на вход ЛЭ – Е всех более младших разрядов, закрывает эти ЛЭ – Е, и установит тем самым на выходе этих разрядов нули независимо от того, какие уровни приходят в эти разряды во входном коде.

Объединение схемы выделения старшей единицы и классического шифратора дает приоритетный шифратор PRIORITY ENCODER сокращенно PRCD. Пример PRCD K155IB1. Здесь имеется вход GS – GROUP SIGNAL – групповой сигнал, на этом выходе активный уровень, в данном случае низкий уровень, когда есть активный уровень хотя бы на одном из разрядов входного кода шифратора.

Выходы EO и GS образуют тракт групповых сигналов и служат для наращивания разрядности.

## Наращивание PRCD.

Если требуется нарастить – разряда выходного кода, это можно сделать с помощью нескольких ЛЭ. Пример приращения – и разрядного приоритетного шифратора K155ИВ1 в 1 – и разрядный с помощью трех ЛЭ.

Здесь появление единицы на входах 8 и 9 запирает по входу Е1 всю микросхему PRCD, но проходит на вход А3'.

Т.к. двоичный код 9=1001 имеет единицу в младшем разряде входного кода, то появилось соединение входа 9 с выходом ЛЭ И в разряде А0 входного кода.

Если требуется нарастить приоритетный шифратор вдвое, то два PRCD соединяют следующим образом.

## Дешифратор

Дешифратор преобразует входной двоичный код в такой выходной код, в котором только на одном из всех выходов дешифратора имеется единица. Такой выходной код называется унарным. Иногда можно встретить другие названия этого кода – унитарный или позиционный. В положительной логике единицей является высокий уровень, но для большинства ТТЛ дешифраторов активным является низкий уровень. Номер активного выхода соответствует двоичному входному коду.

Полным называют дешифратор,  $m$  выходов которого используют все возможные наборы  $n$  входных переменных, т.е.  $m = 2^n$ .

Если число выходов меньше, то такой дешифратор называется неполным ( $m < 2^n$ ).

Дешифраторы используют когда нужно обращаться к различным цифровым устройствам, и при этом номер устройства – его адрес – представлен двоичным кодом, поэтому входы дешифратора иногда называют адресными входами, и обычно их нумеруют не порядковыми номерами 0, 1, 2, 3, 4, 5..., а в соответствии с двоичными весами разрядов 1, 2, 4, 8, 16 ... В соответствии с числом входов и выходов дешифраторы называют “3 – 8” – (три в восемь), “4 – 10” (четыре в десять, неполный).

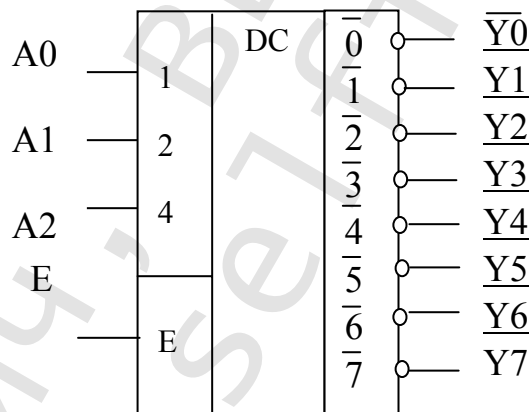


Рис.10.1. Условное графическое обозначение полного дешифратора “3 – 8” с входом E разрешения и с инверсными выходами.

Вход E (ENABLE – разрешение) называют разрешающим, стробирующим, управляющим. Так как через вход E можно передавать информацию (данные) на какой либо из выходов (или на все выходы поочередно), то дешифратор, имеющий E вход, иногда называют демультиплексором и обозначают соответственно DMX (DEMULTIPLEXER) или DX.

Таблица 10.1

Таблица истинности полного дешифратора “3 – 8” с входом E разрешения и с инверсными выходами.

A2	A1	A0	$\bar{Y}7$	$\bar{Y}6$	$\bar{Y}5$	$\bar{Y}4$	$\bar{Y}3$	$\bar{Y}2$	$\bar{Y}1$	$\bar{Y}0$
0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	1	0	0

### Линейный дешифратор.

Наиболее простой по структуре дешифратор, называемый линейным, состоит из  $m$  – входных элементов И. Схема такого дешифратора приведена на рис.10.3. В микросхеме дешифратора имеются входные инверторы – буферные усилители, назначение которых – свести к минимуму нагрузку на выходы предыдущих схем.

Если нет разрешения по входам Е, то все выходы схемы И закрыты, и независимо от состояния адресных входов ни на одном из выходов нет активного уровня. Разрешение Е, подаваемое на все выходные схемы И, заставляет иметь дополнительный вход на каждой схеме И.

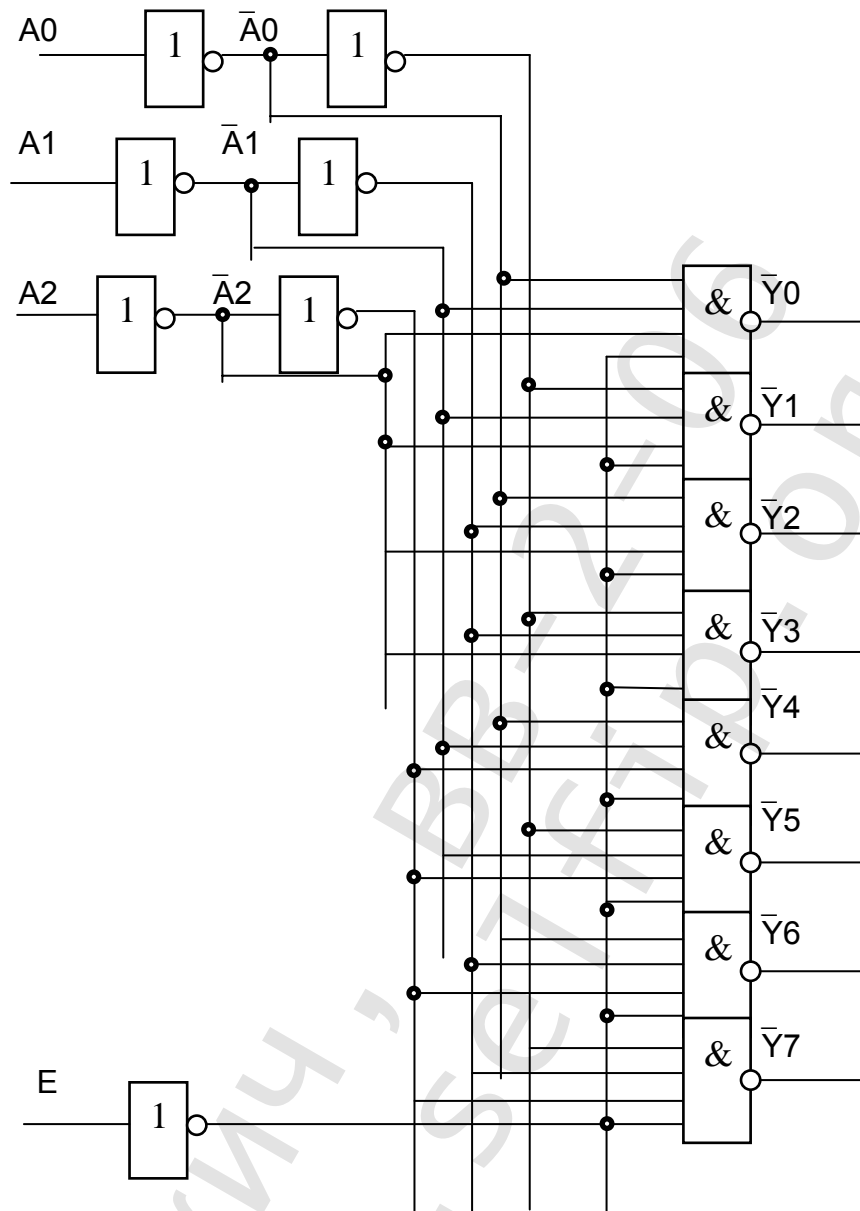


Рис.10.3. Схема линейного дешифратора "3 – 8" с входом E разрешения и с инверсными выходами.

### Матричный (прямоугольный) дешифратор.

Схема матричного (прямоугольного) дешифратора с инверсными выходами приведена на рис 10.5.

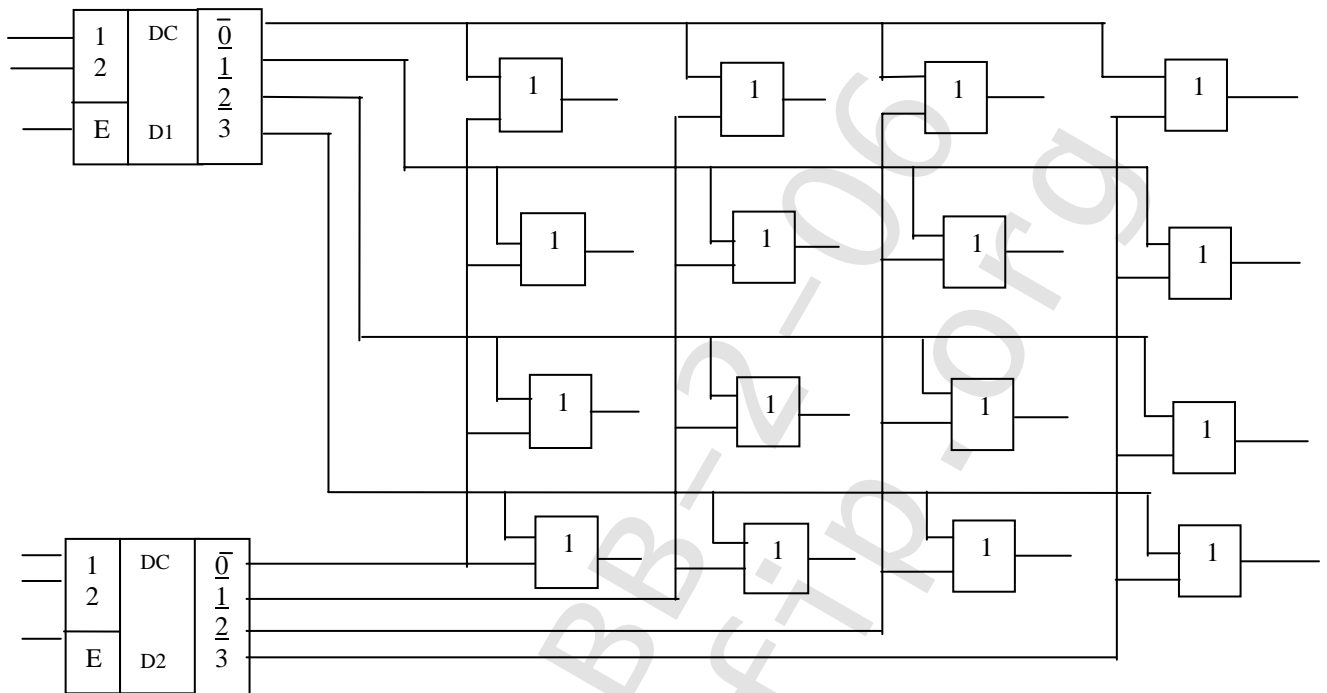


Рис.10.5. Схема матричного (прямоугольного) дешифратора

Если бы выходы первого каскада дешифратора были прямыми, а не инверсными, то для выходного каскада нужны были схемы не 2ИЛИ, а 2И.

Разбивать выходной код для такого дешифратора рекомендуется поровну, или как можно ближе к этому, тогда меньше аппаратные затраты.

Любой из входов E разрешения дешифраторов первого каскада запирает весь матричный дешифратор, т.к. запираются все его строки или столбцы.



### Каскадное соединение дешифраторов.

Приведенное на рис.10.4 соединение работает как дешифратор 5 – 32.

Такое соединение позволяет построить дешифратор с числом выходов большим, чем у имеющихся в наличии микросхем.

Здесь младшие разряды входного кода  $a_1, a_2, a_4$  поступают на входы всех дешифраторов DD2... DD5 выходного каскада, а старшие разряды входного кода – на вход дешифратора первого каскада.

Активным из всех 32 выходов будет лишь один выход у того дешифратора второго каскада, разрешение на входе E которого поступает с дешифратора первого каскада DD1.

Пример: Входной код 01111(15). На первом каскаде дешифратора два старших разряда входного кода сделают активным выход 1 DD1, значит разрешение на втором каскаде будет подано на DD3, а три младших разряда входного кода сделают активным выход 7 этой микросхемы, следовательно выход 15 всего дешифратора будет единственным активным.

В нашем случае входной 5 – разрядный код разбит на две группы по 2 и 3 разряда. Разбивать входной код на группы можно различными способами, и в пределах каждому разряду входного кода может соответствовать свой отдельный каскад дешифратора. Такой дешифратор является пирамидальным. Естественно, сумма задержек в таком пирамидальном дешифраторе максимальна – это недостаток, но такой дешифратор состоит только из двух входных элементов – иногда это удобно.

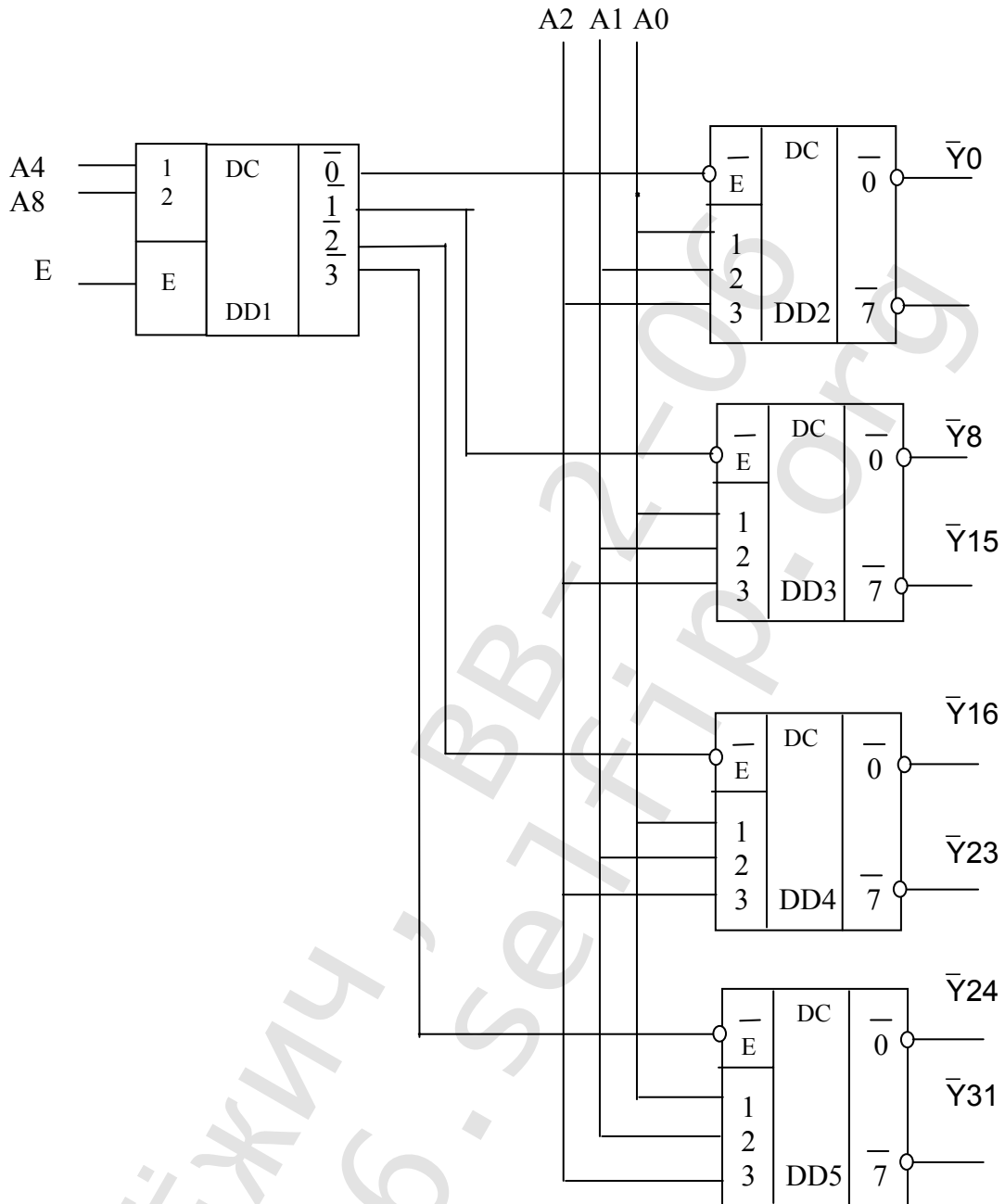


Рис.10.4. Схема каскадного дешифратора “5 – 32” с входом E разрешения и с инверсными выходами.

### **Сравнительная характеристика дешифраторов.**

1. Линейный дешифратор имеет минимальную задержку. Но у него максимальные из всех типов дешифраторов аппаратные затраты.

2. Матричный (прямоугольный) дешифратор самый экономичный по оборудованию, имеет среднюю задержку.

3. Каскадный дешифратор имеет наибольшую задержку при числе каскадов больше двух, но удобен при использовании готовых микросхем – дешифраторов.

Микросхемы дешифраторов часто имеют не один, а два или даже три входа Е разрешения, причем некоторое из них прямые, а другие – инверсные. Такие входы удобно использовать при наращивании дешифратора, собирая как бы каскадный дешифратор, но вместо первого каскада дешифратора можно использовать входы разрешения микросхем – дешифраторов.

## Мультиплексор.

Мультиплексированием (MULTIPLEX) называют передачу данных от нескольких источников по одному каналу поочередно.

В качестве примера может служить переключка студентов на лекции – передача данных от каждого студента по спискам групп через преподавателя в журнал. Обратный процесс – из журнала через декана некоторые студенты получают данные – это демультимплексирование.

В цифровой технике мультиплексор имеет  $m$  информационных входов данных  $D_0, D_1, D_2, D_3, \dots, D_m$ ;  $n$  адресных входов  $A_1, A_2, A_3, \dots, A_n$ ; и один выход данных.  $m=2^n$ .

Двоичный код на входах адреса определяет номер того входа данных, с которого информация проходит на выход MS в этот момент.

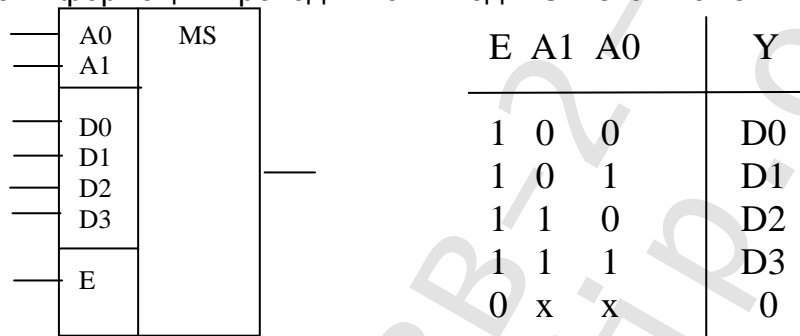


Рис.10.6. Условное графическое обозначение мультиплексора и его таблица истинности.

Обозначения MS отличается от MUX, которого требует ГОСТ, потому что в нем отражена еще одна функция – выборка, селекция данных из определенного, указанного адресом источника, которая обозначается S – ELECT – выбирать. Обозначение MS встречается чаще всего, но иногда бывает MUX, и изредка SL.

В состав мультиплексора обязательно входит дешифратор адреса в том или ином виде. В первом варианте мультиплексора, приведенном ниже на рис.10.7, дешифратор адреса выделен в отдельный узел.

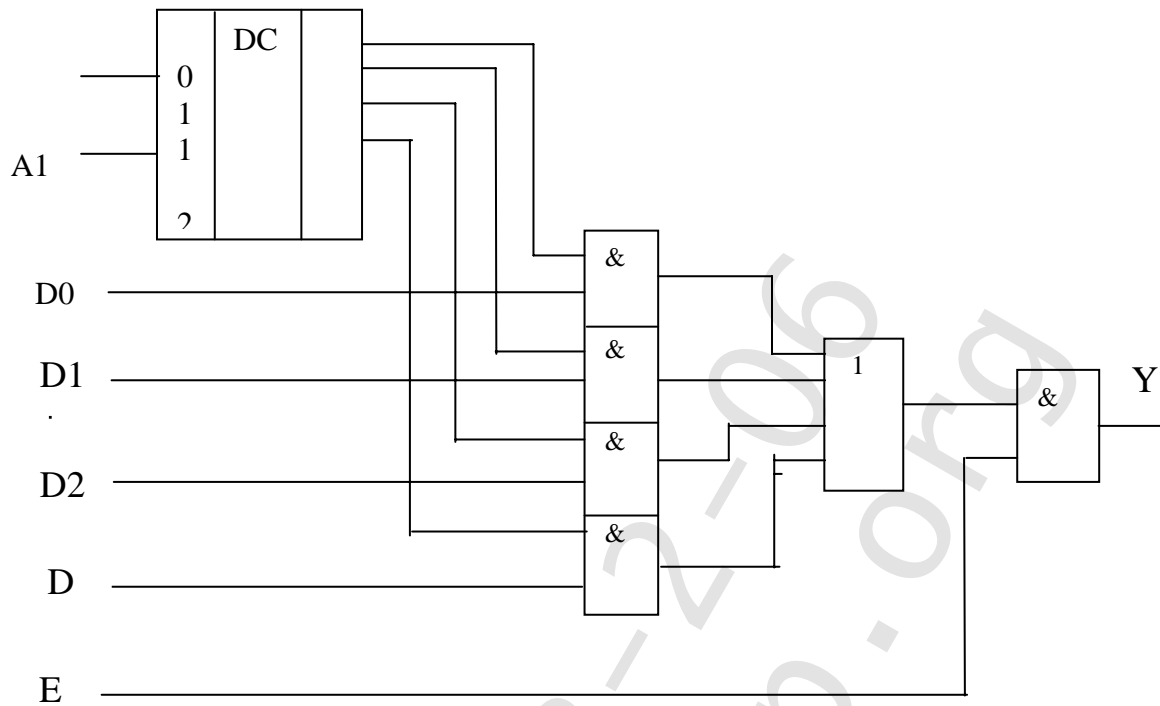


Рис.10.7. Схема мультиплексора с явно выделенным дешифратором.

Как известно линейный дешифратор состоит из группы инверторов по адресным входам и линейки – столбца – элементов И.

Столбец элементов И мультиплексора может выполнять функцию линейного дешифратора, если эти элементы И, но с большим числом входов подключить к выходам инверторов по адресным линиям в соответствии с таблицей истинности дешифратора.

Второй вариант схемы мультиплексора такой же разрядности, полученный с учётом вышеизложенного, считается минимизированным для базиса И – НЕ. Так построен мультиплексор К155КП2, схема которого приведена на рис. 10.8.

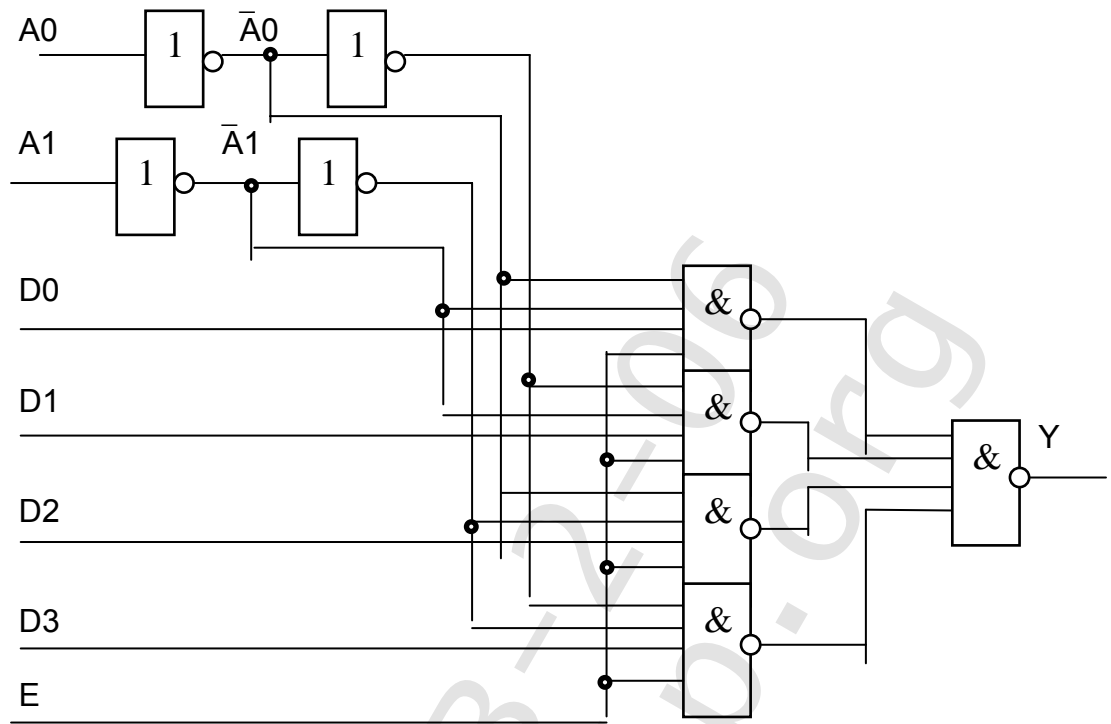


Рис.10.8. Схема мультиплексора К155КП2, минимизированного для базиса И – НЕ.

### Способы наращивания мультиплексов.

Почти у всех микросхем мультиплексов есть входы Е разрешения. Эти входы Е разрешения могут использоваться как дополнительные адресные входы. Можно объединить два мультиплектора, получив тем самым мультиплектор с удвоенным числом входов.

На рис.10.9. приведен пример как из двух мультиплексов 4–1 можно собрать мультиплектор 8–1.

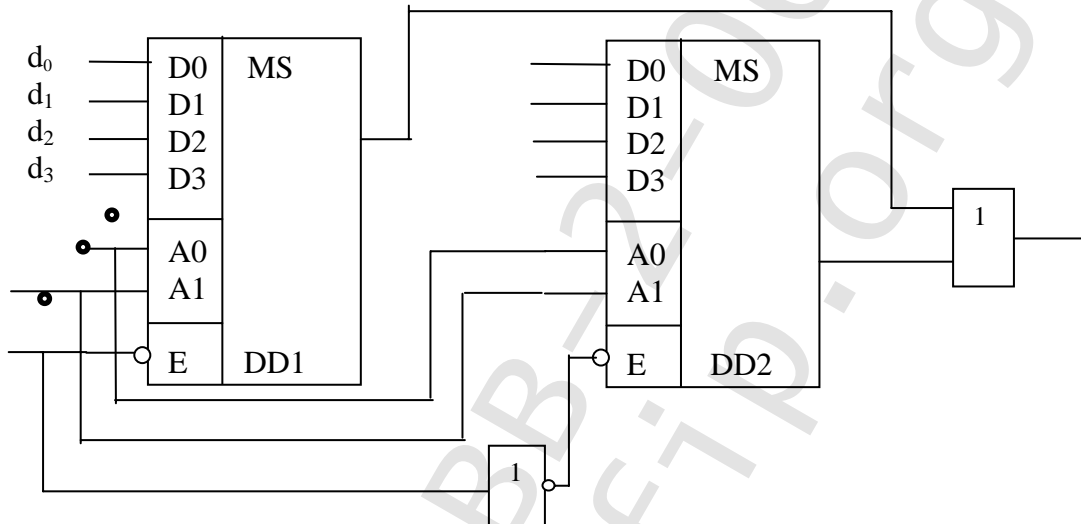


Рис.10.9. Пример объединения двух мультиплексов с целью получения мультиплектора с удвоенным числом входов

Когда удваивание числа входов мультиплектора недостаточно, применяют так называемое пирамидальное или каскадное соединение мультиплексов, показанное на рис 10.10. Обычно применяют два, реже три и более каскадов.

Младшие разряды адресного кода подаются параллельно на все адресные входы первого каскада, а старшие разряды адресного кода подаются на второй каскад или, если они есть, то и на последующие каскады мультиплектора.

На рис 10.10. из четырех мультиплексов с четырьмя входами данных (4 – 1) собрана первая ступень а всего из пяти таких мультиплексов собран мультиплектор с шестнадцатью входами (16 – 1).

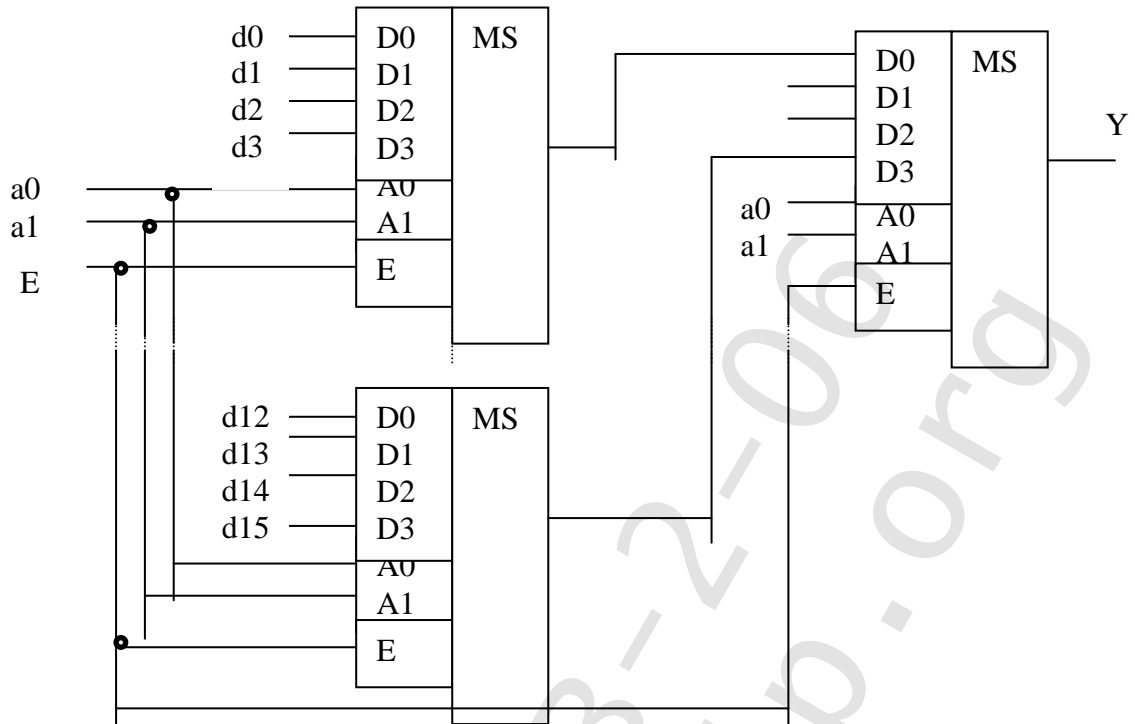


Рис.10.10. Пример объединения пяти мультиплексоров с четырьмя входами данных (4 – 1) для получения мультиплексора с шестнадцатью входами (16 – 1)



### Мультиплексор как универсальный логический элемент.

С помощью мультиплексора можно реализовать любую логическую функцию с числом входных переменных, равных числу адресных входов мультиплексора.

Допустим задана функция от трех переменных, которую уже нельзя упростить:

$$F = \bar{A} B C + A \bar{B} C + A B \bar{C}.$$

Для реализации этой функции в базисе И–НЕ преобразуем ее до:

$$F = \overline{(\bar{A} B C) (A \bar{B} C) (A B \bar{C})}.$$

По заданному выражению построим схему на инвертирующих логических элементах НЕ и И–НЕ:

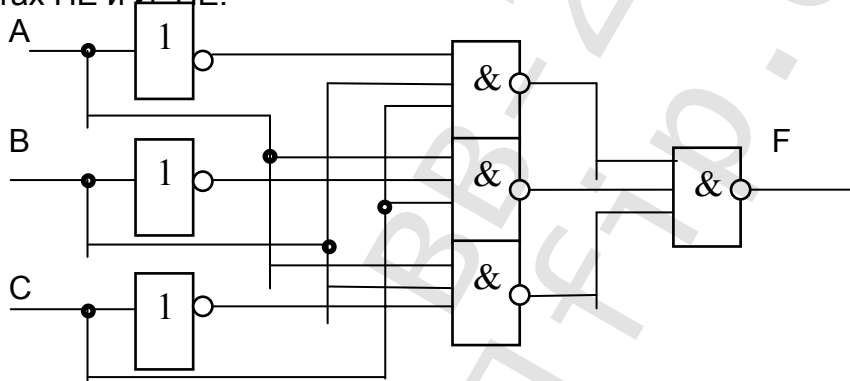


Рис.10.11. Схема, реализующая логическое выражение

$$F = \overline{(\bar{A} B C) (A \bar{B} C) (A B \bar{C})}.$$

Эта функция для своей реализации в базисе И–НЕ требует четыре логических элемента 3И–НЕ и три инвертора, что составляет 1,8 корпуса ИМС. Вся схема имеет задержку, равную  $3t_{зд.ср.}$ .

Запишем таблицу истинности данной исходной функции в обычном виде :

N	A	B	C	F
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	0

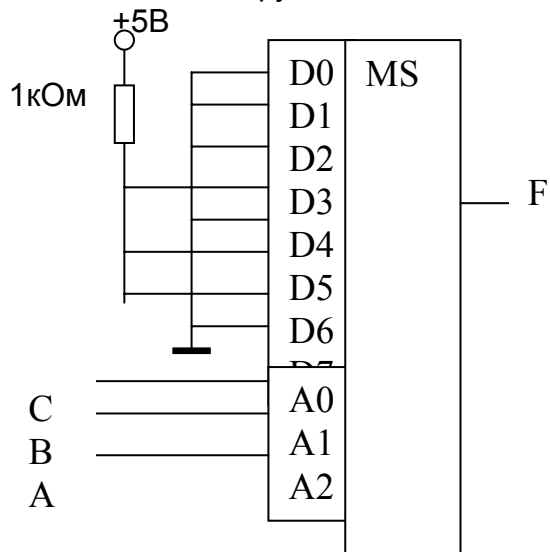


Рис.10.12. Таблица истинности, соответствующая исходной функции, и схема на базе мультиплексора 8–1, реализующая эту функцию

Для реализации таблицы истинности, соответствующей исходной функции, берем мультиплексор с числом входов, равных числу входных переменных данной функции, то есть мультиплексор с тремя адресными входами, и соответственно с  $2^3 = 8$  входами данных. Соединяем входы данных мультиплексора D0...D7 с лог. 0 и лог. 1 в соответствии с таблицей истинности и получаем схему, реализующую заданную функцию. При этом потребовался один корпус ИМС типа К155КП5.

Есть способ сокращения аппаратных затрат еще почти вдвое. Для этого таблица истинности разбивается на группы в две строки.

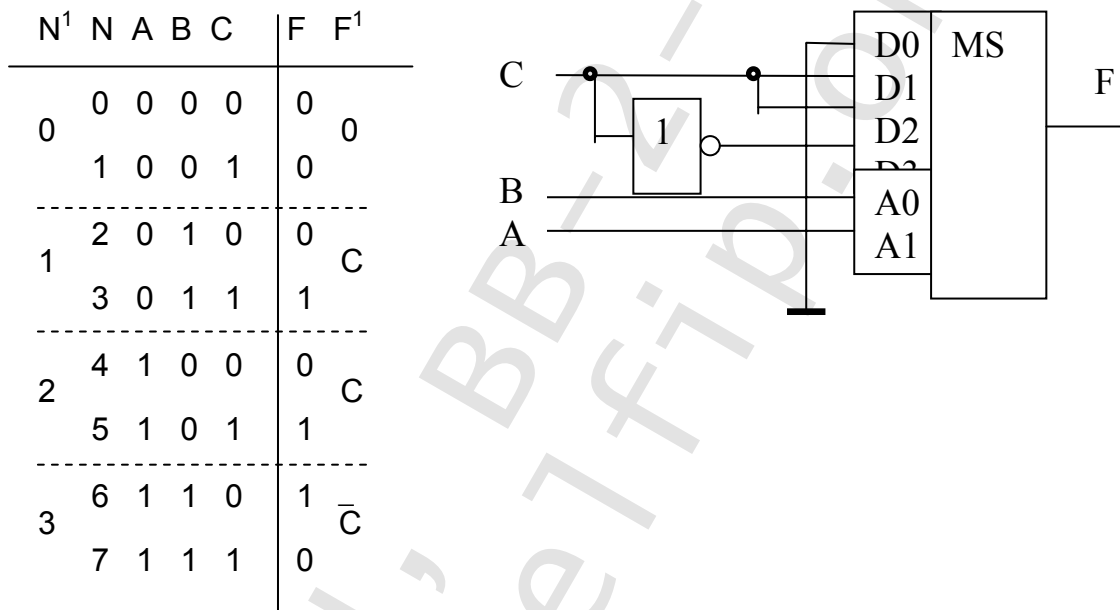


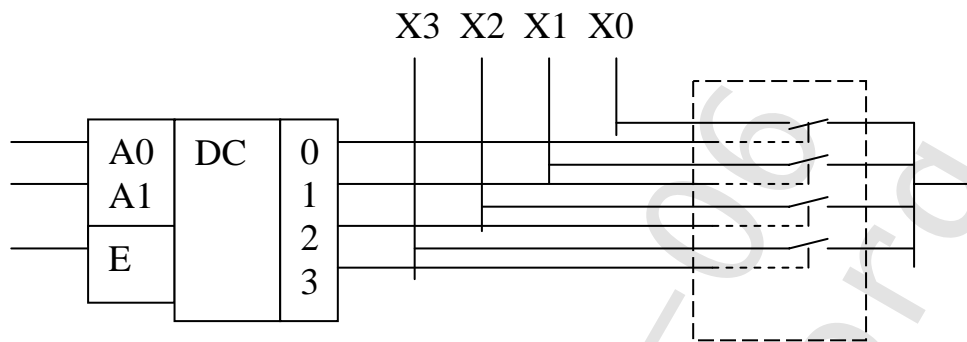
Рис.10.13. Таблица истинности, соответствующая исходной функции, и схема на базе мультиплексора 4–1, реализующая эту функцию

Для такой реализации заданной функции берем мультиплексор вдвое меньше предыдущего и соединяем его входы в соответствии с таблицей истинности с лог. 0, лог.1, а также с прямой или инвертированной старшей входной переменной.

В данном случае аппаратные затраты составили 0,66 корпуса ИМС(0,5 корпуса MS КП2 и 1/6 корпуса инвертора).

### Особенности КМОП мультиплексоров.

КМОП мультиплексоры строятся на основе дешифраторов и двунаправленных ключей коммутации, как показано на рис.10.14.



Такие мультиплексоры имеют три следующих существенных отличия от остальных.

1. Поскольку ключи двунаправленные, то их выходы можно использовать как входы, значит мультиплексор можно использовать как демультимплексор.

2. Наличие входа E разрешения позволяет закрыть сразу все ключи – это равносильно тому, что вход мультиплексора и выходы демультимплексора имеют третье – остояние с высоким выходным сопротивлением.

3. КМОП мультиплексоры могут коммутировать не только цифровые, но и аналоговые сигналы.

## Преобразователи кодов

Под преобразованием кодов понимается преобразование  $n$ -разрядных двоичных чисел, представляющих информацию в одном заданном коде, в  $m$ -разрядные двоичные числа, представляющие эту информацию в другом коде. Наиболее распространены следующие два подхода к построению преобразователей кодов.

Первый подход - синтез  $m$  независимых одновыходных функций по заданной таблице истинности – таблице соответствия кодов.

Допустим, имеется таблица истинности преобразователя 2-х разрядного двоичного кода в 3-х разрядный:

$a_1$	$a_0$		$b_2$	$b_1$	$b_0$
0	0		1	0	0
0	1		0	0	1
1	0		0	1	1
1	1		0	0	0

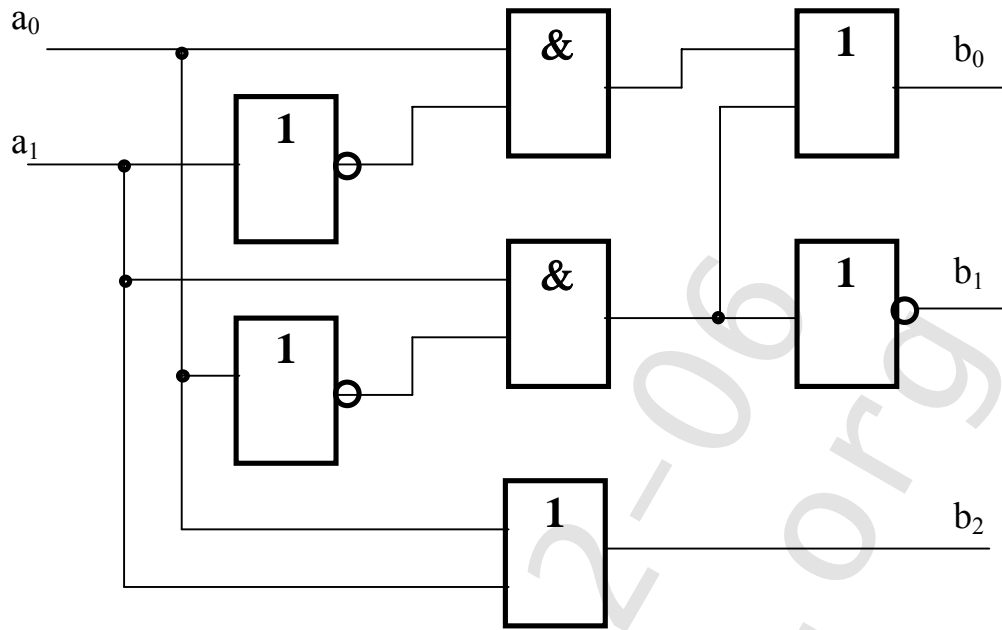
Считая  $b_0$ ,  $b_1$  и  $b_2$  независимыми одновыходными функциями, запишем для каждой из них булевы выражения.

$$b_0 = \bar{a}_1 a_0 + a_1 \bar{a}_0,$$

$$b_1 = a_1 \bar{a}_0,$$

$$b_2 = \bar{a}_1 \bar{a}_0.$$

Используя приведенные булевы выражения с учетом того, что инверсии некоторых переменных и произведение  $a_1 \bar{a}_0$  встречаются не в одной функции, составляем схему преобразователя с применением (где это возможно) одних и тех же логических элементов:



Второй подход - построение преобразователя кодов по методу "дешифратор-шифратор".

В этом случае по заданной таблице истинности составляется таблица соответствия десятичных цифр. В рассмотренном примере эта таблица запишется следующим образом

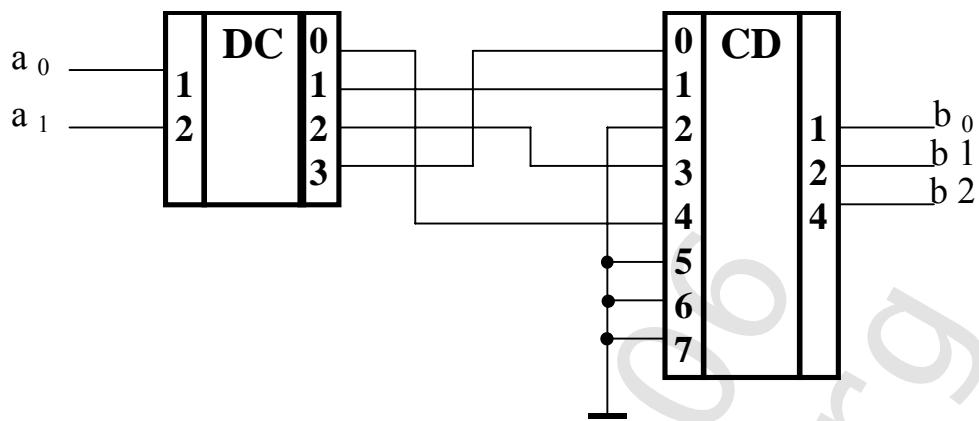
A	B
0	4
1	1
2	3
3	0

Метод построения преобразователя заключается в следующем.

Сначала строится схема преобразователя двоичного кода в унарный, т.е. такой код, когда "1" может быть только на одном из  $N$  выходов преобразователя, номер которого совпадает с числом, представленным входным двоичным кодом. (такой преобразователь называется дешифратором). Число выходов дешифратора равно  $N = 2^n$ , где  $n$  – число разрядов входного кода дешифратора.

Затем строится схема преобразователя, осуществляющего обратную операцию, т.е. преобразование унарного кода в двоичный. Число входов такого преобразователя, который называется шифратором, равно  $M = 2^m$ , где  $m$  – число разрядов выходного кода шифратора.

Схема преобразователя кода образуется соединением выходов дешифратора и входов шифратора в соответствии с таблицей соответствия десятичных цифр. Здесь DC - обозначение дешифратора (DECODER), CD - обозначение шифратора (ENCODER).



### Контроль четности.

В системах передачи двоичной информации с целью повышения надежности передачи широко используется специфическая арифметическая операция - проверка паритета двоичных чисел.

Суть ее заключается в суммировании по модулю 2 всех разрядов с целью выяснения четности числа. При передаче информации по линии связи вследствие влияния помех могут возникать искажения, приводящие к искажениям информации. Если, например, передается код  $1001 = 9$  и вследствие помех произойдет сбой во втором разряде слева, на приемный конец поступит  $1101 = 13$ . В общем случае без специальной проверки факта ошибки не установить.

Простой и эффективный способ обнаружения ошибок основан на допущении, что в каждый момент времени ошибка может возникнуть только в одном разряде и проявляется она в лишней единице или в потере единицы. В обоих случаях число единиц изменяется на одну. Таким образом, если передаваемое слово содержит четное число единиц по всем разрядам, а на конце линии передачи это число окажется нечетным, значит, появилась ошибка.

Реализация этого метода осуществляется с помощью специальных устройств сравнения (схем контроля четности), которые выпускаются в микросхемном исполнении как самостоятельные устройства.

На передающем конце схема сравнения формирует дополнительный бит (1 или 0), так называемый паритетный или контрольный бит, который добавляется к выходной информации. Назначение паритетного бита - доводить число единиц в каждом передаваемом слове до четного или нечетного в зависимости от принятой системы кодирования. При всех передачах информации, включая запись в память и считывание, контрольный бит передается вместе со словом. На приемном конце происходит проверка паритета (от англ. - соответствие, аналогия) поступивших сигналов. Если он правилен, разрешается прием. Если на линии связи имеет место искажение передаваемой информации, происходит включение сигнализатора ошибок. Паритет может быть четным и нечетным. В случае нечетного паритета дополнительный бит формируется таким образом, чтобы сумма всех единиц в передаваемом слове, включая проверочный (контрольный) бит, была нечетной. Для четного наоборот. К примеру, в числе 0111 число единиц нечетно. Поэтому для нечетного паритета дополнительный бит должен быть нулем, а для четного - единицей. Передаваемое слово будет:

00111 - нечетный паритет;

10111 - четный паритет.

Здесь контрольный бит расположен слева, т. е. Занимает старший разряд передаваемого слова. На практике нечетный паритет используется чаще. Контроль нечетности позволяет фиксировать полное пропадание информации, поскольку слово из одних нулей (включая контрольный бит) противоречит нечетному паритету.

Принято считать при проверке как четности так и нечетности, что при правильном коде на входах на выходе формируется логическая "1", при ошибочном - логический "0". Для организации схем сравнения применяют логические элементы "исключающее ИЛИ", исполняющие роль сумматоров по модулю 2 (т. е. сумматоров, сигналом переноса которых пренебрегают). На рис.3 показано такое устройство на 4 разряда. Структура схемы проверки четности (нечетности) - многоступенчатая. В первой ступени (ярусе) попарно суммируются все биты слова. Выходные сигналы первого яруса служат входными для второго - и так последовательно до окончательного определения четности (нечетности) суммы единиц всего слова. Полученный результат на последнем этапе сравнивается с контрольным сигналом, задающим вид используемого паритета.

Если принят четный паритет, т.е. число единиц в слове, включая паритетный бит, должно быть четным, то контрольный сигнал должен быть равен сумме по модулю 2 всех информационных разрядов слова. Для нечетного паритета контрольный сигнал является инверсией указанной суммы (табл. 1).

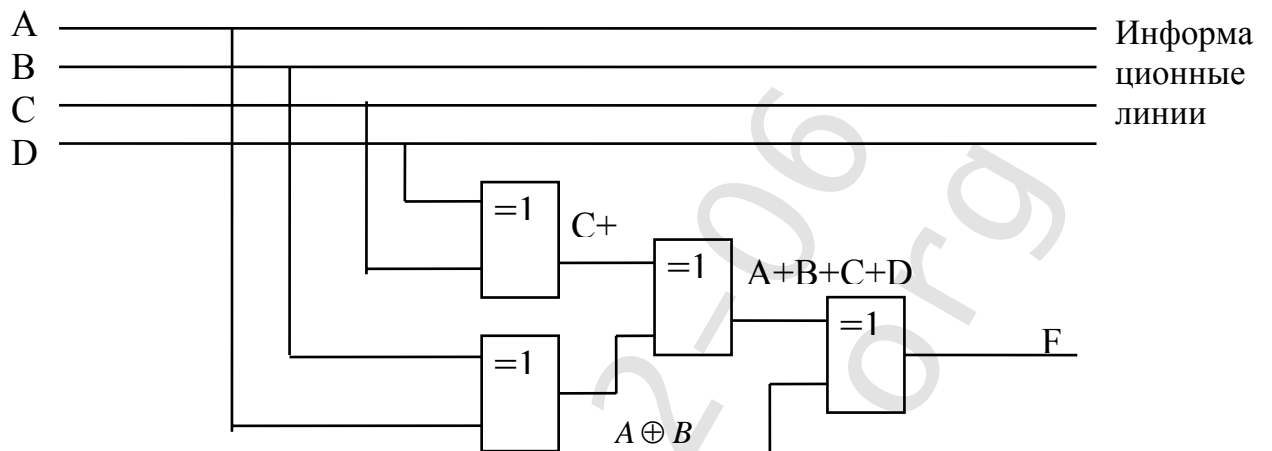


Рис. 3. Схема формирования паритетного бита.

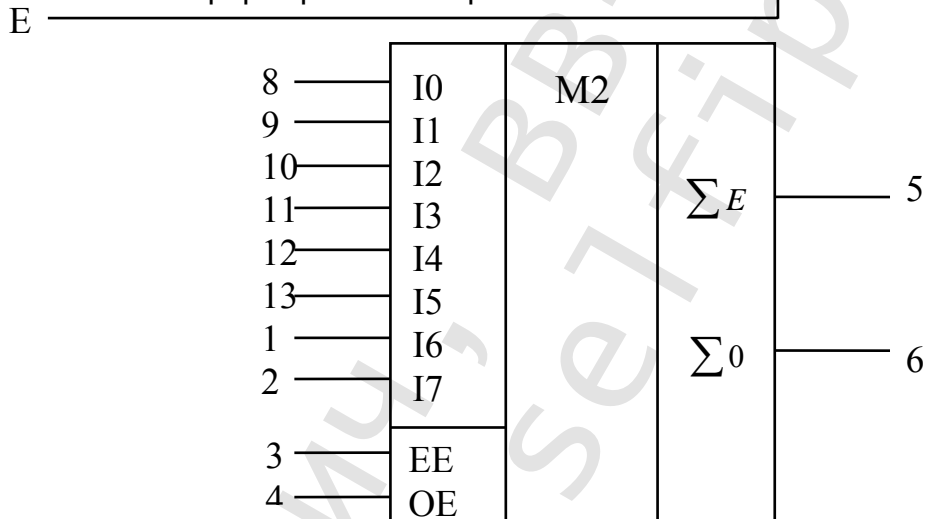


Рис. 4. Условное изображение микросхемы К155ИП2.

Таблица 1

Таблица истинности для схемы рис. 3

Входы				Выход	
A	B	C	D	при E = 0	при E = 1
0	0	0	0	0	1
0	0	0	1	1	0
0	0	1	0	1	0
0	0	1	1	0	1
0	1	0	0	1	0
0	1	0	1	0	1
0	1	1	0	0	1
0	1	1	1	1	0
1	0	0	0	1	0
1	0	0	1	0	1
1	0	1	0	0	1



1	0	1	1	1	0
1	1	0	0	0	1
1	1	0	1	1	0
1	1	1	0	1	0
1	1	1	1	0	1

Таким образом, независимо от паритета четырехразрядного слова на информационных линиях А, В, С и D паритет пятиразрядного кода А,В,С,Д всегда будет одинаков. Это следует из того факта, что если сумма А,В,С,Д нечетна (четна), то при F=0 (F=1) и их общая сумма также нечетна. Потенциал на входе E (E=0 или E=1) определяет таким образом вид используемого паритета.

Устройства для проверки четности двоичных слов выпускаются в виде самостоятельных изделий в нескольких сериях микросхем. Они находят применение также в качестве сумматоров по модулю 2 и поэтому иногда называются схемами свертки по модулю 2.

В устройствах ТТЛ 155-й серии контроль четности осуществляется микросхемой К155ИП2. Она имеет (рис.4.) восемь информационных входов I0 - I7, два разрешающих входа для задания вида паритета: четный EE (even enable) и нечетный OE (odd enable), два выхода  $\sum E$  и  $\sum 0$ . Оба выхода взаимодополняют.

Микросхема может работать в режиме четного и нечетного паритета как в положительной, так и в отрицательной логике.

Функциональные свойства микросхемы характеризует табл.2

Сигналами на входах EE и OE можно обеспечить разные режимы работы микросхемы, в частности управление полярностью выходных сигналов, создание девятого информационного входа, каскадирование микросхем К155ИП2 с целью повышения разрядности контролируемых слов.

Таблица 2

Таблица истинности микросхемы К155ИП2

Число единиц на входах	Входы		Выходы	
	EE	EO	$\sum E$ (четное)	$\sum 0$ (нечет)
четное	1	0	1	0
нечетное	1	0	0	1
четное	0	1	0	1
нечетное	0	1	1	0
X	1	1	0	0
X	0	0	1	1

Примечание. Символ x означает любое (четное или нечетное) число единиц на входах I0 - I7.

При поступлении на вход EE уровня U (H-уровня), а на вход OE уровня U (L-уровня) схема реализует функцию "контроль четности" при четном числе единиц на информационных входах I0-I7 на четном выходе E будет высокий уровень U. При нечетном числе единиц на этих входах состояния выходов E и O меняются на противоположные и обеспечивают контроль четности. Когда на обоих разрешающих входах действуют сигналы одного уровня, на обоих выходах образуются инверсные им сигналы.

## Сумматоры

Основной элементарной операцией, выполняемой над кодами чисел в цифровых устройствах, является арифметическое сложение.

**Сумматор — логический** операционный узел, выполняющий **арифметическое** сложение кодов двух чисел. При арифметическом сложении выполняются и другие дополнительные операции: учёт знаков чисел, выравнивание порядков слагаемых и тому подобное. Указанные операции выполняются в арифметическо-логических устройствах (АЛУ) или процессорных элементах, ядром которых являются сумматоры.

Сумматоры классифицируют по различным признакам.

**В зависимости от системы счисления** различают:

- двоичные;
- двоично-десятичные (в общем случае двоично-кодированные);
- десятичные;
- прочие (например, амплитудные).

**По количеству одновременно обрабатываемых разрядов складываемых чисел:**

- одноразрядные,
- многоразрядные.

**По числу входов и выходов одноразрядных двоичных сумматоров:**

- четвертьсумматоры (элементы “сумма по модулю 2”; элементы “исключающее ИЛИ”), характеризующиеся наличием двух входов, на которые подаются два одноразрядных числа, и одним выходом, на котором реализуется их арифметическая сумма;
- полусумматоры, характеризующиеся наличием двух входов, на которые подаются одноимённые разряды двух чисел, и двух выходов: на одном реализуется арифметическая сумма в данном разряде, а на другом — перенос в следующий (более старший разряд);
- полные одноразрядные двоичные сумматоры, характеризующиеся наличием трёх входов, на которые подаются одноимённые разряды двух складываемых чисел и перенос из предыдущего (более младшего) разряда, и двумя выходами: на одном реализуется арифметическая сумма в данном разряде, а на другом — перенос в следующий (более старший разряд).

**По способу представления и обработки складываемых чисел** многоразрядные сумматоры подразделяются на:

- последовательные, в которых обработка чисел ведётся поочерёдно, разряд за разрядом на одном и том же оборудовании;
- параллельные, в которых слагаемые складываются одновременно по всем разрядам, и для каждого разряда имеется своё оборудование.

Параллельный сумматор в простейшем случае представляет собой  $n$  одноразрядных сумматоров, последовательно (от младших разрядов к старшим) соединённых цепями переноса. Однако такая схема сумматора характеризуется сравнительно невысоким быстродействием, так как формирование сигналов суммы и переноса в каждом  $i$ -ом разряде производится лишь после того, как поступит сигнал переноса с  $(i-1)$ -го разряда. **Таким образом, быстродействие сумматора определяется временем распространения сигнала по цепи переноса. Уменьшение этого времени — основная задача при построении параллельных сумматоров.**

Для уменьшения времени распространения сигнала переноса применяют: **конструктивные решения**, когда используют в цепи переноса наиболее быстродействующие элементы; тщательно выполняют монтаж без длинных проводников и паразитных ёмкостных составляющих нагрузки и (наиболее часто) структурные методы ускорения прохождения сигнала переноса.

**По способу организации межразрядных переносов параллельные сумматоры, реализующие структурные методы**, делят на сумматоры:

- с последовательным переносом;
- с параллельным переносом;
- с групповой структурой;
- со специальной организацией цепей переноса.

Три первых структуры будут подробно рассмотрены в последующих статьях. **Среди сумматоров со специальной организацией цепей переноса можно указать:**

- сумматоры со сквозным переносом, в которых между входом и выходом переноса одноразрядного сумматора оказывается наименьшее число логических уровней [1];
- сумматоры с двухпроводной передачей сигналов переноса [1, 2];
- сумматоры с условным переносом (вариант сумматора с групповой структурой, позволяющий уменьшить время суммирования в 2 раза при увеличении оборудования в 1,5 раза) [3];
- асинхронные сумматоры, вырабатывающие признак завершения операции суммирования, при этом среднее время суммирования уменьшается, поскольку оно существенно меньше максимального.

Сумматоры, которые имеют постоянное время, отводимое для суммирования, независимое от значений слагаемых, называют **синхронными**.

**По способу выполнения операции сложения и возможности сохранения результата сложения** можно выделить три основных вида сумматоров:

- комбинационный, выполняющий микрооперацию “ $S = A$  плюс  $B$ ”, в котором результат выдаётся по мере его образования (это комбинационная схема в общепринятом смысле слова);
- сумматор с сохранением результата “ $S = A$  плюс  $B$ ”;
- накапливающий, выполняющий микрооперацию “ $S = S$  плюс  $B$ ”.

Последние две структуры строятся либо на счётных триггерах (сейчас практически не используются), либо по структуре “комбинационный сумматор – регистр хранения” (сейчас наиболее употребляемая схема).

**Важнейшими параметрами сумматоров** являются:

- разрядность;
- статические параметры:  $U_{вх}$ ,  $U_{вх}$ ,  $I_{вх}$  и так далее, то есть обычные параметры интегральных схем;
- динамические параметры. Сумматоры характеризуются четырьмя задержками распространения:
- от подачи входного переноса до установления всех выходов суммы при постоянном уровне на всех входах слагаемых;
- от одновременной подачи всех слагаемых до установления всех выходов суммы при постоянном уровне на входе переноса;
- от подачи входного переноса до установления выходного переноса при постоянном уровне на входах слагаемых;

- от подачи всех слагаемых до установления выходного переноса при постоянном уровне на входах слагаемых.

© Ёжич, ВВ-2-06  
vv206.selfip.org

## Четверть сумматор

Простейшим двоичным суммирующим элементом является четверть сумматор. Происхождение названия этого элемента следует из того, что он имеет в два раза меньше выходов и в два раза меньше строк в таблице истинности по сравнению с полным двоичным одноразрядным сумматором. Наиболее известны для данной схемы названия: элемент “сумма по модулю 2” и элемент “исключающее ИЛИ”. Схема (рис. 1) имеет два входа  $a$  и  $b$  для двух слагаемых и один выход  $S$  для суммы. Работу её отражает таблица истинности 1 (табл. 1), а соответствующее уравнение имеет вид

$$S = \bar{a}b + a\bar{b} = a \oplus b \quad (1)$$

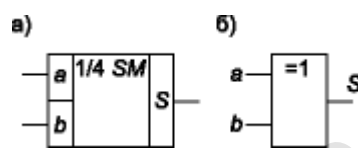


Рис. 1

Таблица 1

a	b	S
0	0	0
0	1	1
1	0	1
1	1	0

Данный элемент выпускается в виде интегральных схем (ИС) типа ЛП5 (серии 133, 155, 530, 531, 533, 555, 1531, 1533); ЛП12 (555); ЛП107 (100, 500, 1500); ЛП2 (561, 564); ЛП14 (1561) и т. п.

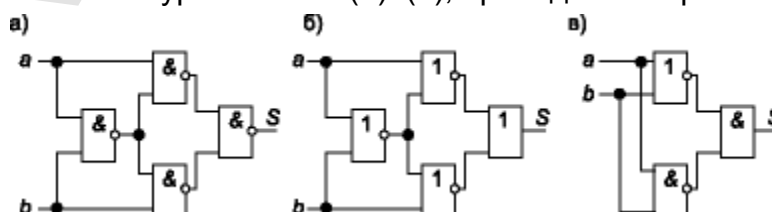
Реализуем четвертьсумматор в базисах И-НЕ, ИЛИ-НЕ и с использованием только одного инвертора, для чего преобразуем уравнение (1):

$$\begin{aligned} S &= \bar{a}b + a\bar{b} = \bar{a}a + \bar{a}b + \bar{b}b + a\bar{b} = \\ &= a(\bar{a} + b) + b(\bar{a} + \bar{b}) = a\bar{a}b + b\bar{a}\bar{b} = \overline{\overline{a\bar{a}b} \cdot \overline{b\bar{a}\bar{b}}} \end{aligned} \quad (2)$$

$$\begin{aligned} S &= \bar{a}b + a\bar{b} = \bar{a}a + \bar{a}b + \bar{b}b + a\bar{b} = \\ &= \overline{\overline{a(a+b)}} + \overline{\overline{b(a+b)}} = \overline{a + a + b} + \overline{b + a + b} \end{aligned} \quad (3)$$

$$\begin{aligned} S &= \bar{a}b + a\bar{b} = \bar{a}a + \bar{a}b + \bar{b}b + a\bar{b} = \\ &= \bar{a}(a+b) + \bar{b}(a+b) = (a+b)(\bar{a} + \bar{b}) = (a+b)\bar{a}\bar{b} \end{aligned} \quad (4)$$

Схемы, полученные по уравнениям (2)–(4), приведены на рис. 2.



**Рис. 2**

© Ёжич, ВВ-2-06  
[vv206.selfip.org](http://vv206.selfip.org/)

## Полусумматор

**Полусумматор** (рис. 3) имеет два входа  $a$  и  $b$  для двух слагаемых и два выхода:  $S$  — сумма,  $P$  — перенос. Обозначением полусумматора служат буквы HS (half sum — полусумма). Работу его отражает таблица истинности 2 (табл. 2), а соответствующие уравнения имеют вид:

$$\left. \begin{aligned} S &= \bar{a}b + a\bar{b} = a \oplus b \\ P &= ab \end{aligned} \right\} \quad (5)$$

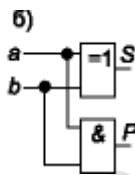
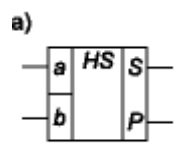


Рис. 3

Таблица 2

a	b	P	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Из уравнений (5) следует, что для реализации полусумматора требуется один элемент “исключающее ИЛИ” и один двухвходовый вентиль И (рис. 3б).

## Полный одноразрядный двоичный сумматор

Он (рис. 4) имеет три входа:  $a$ ,  $b$  — для двух слагаемых и  $p$  — для переноса из предыдущего (более младшего) разряда и два выхода:  $S$  — сумма,  $P$  — перенос в следующий (более старший) разряд. Обозначением полного двоичного сумматора служат буквы  $SM$ . Работу его отражает таблица истинности 3 (табл. 3).

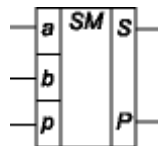


Рис. 4

Таблица 3

№ наб.	a	b	p	P	S
0	0	0	0	0	0
1	0	0	1	0	1
2	0	1	0	0	1
3	0	1	1	1	0
4	1	0	0	0	1
5	1	0	1	1	0
6	1	1	0	1	0
7	1	1	1	1	1

Отметим два момента. **Первый:** в табл. 2 и 3 выходные сигналы  $P$  и  $S$  не случайно расположены именно в такой последовательности. Это подчеркивает, что  $PS$  рассматривается как двухразрядное двоичное число, например,  $1 + 1 = 2_{10} = 10_2$ , то есть  $P = 1$ , а  $S = 0$  или  $1 + 1 + 1 = 3_{10} = 11_2$ , то есть  $P = 1$ , а  $S = 1$ .

**Второй:** выходные сигналы  $P$  и  $S$  полного двоичного сумматора относятся к классу самодвойственных функций алгебры логики. **Самодвойственными** называют функции, инвертирующие своё значение при инвертировании всех переменных, от которых они зависят. Обратите внимание, что  $P$  и  $S$  для четвертьсумматора и полусумматора не являются самодвойственными функциями! Преимущества, вытекающие из этого свойства полного двоичного сумматора, будут рассмотрены при анализе возможностей ИС типа 155ИМ1.

Уравнения, описывающие работу полного двоичного сумматора, представленные в совершенной дизъюнктивной нормальной форме (СДНФ), имеют вид:

$$\left. \begin{aligned} S &= \bar{a}\bar{b}p + \bar{a}b\bar{p} + a\bar{b}\bar{p} + abp \\ P &= \bar{a}bp + a\bar{b}p + ab\bar{p} + abp \end{aligned} \right\} \quad (6)$$

Уравнение для переноса может быть минимизировано:

$$P = ab + ap + bp. \quad (7)$$

При практическом проектировании сумматора уравнения (6) и (7) могут быть преобразованы к виду, удобному для реализации на заданных логических элементах с некоторыми ограничениями (по числу логических входов и др.) и удовлетворяющему предъявляемым к сумматору требованиям по быстродействию.

Например, преобразуем уравнения (6) следующим образом:



$$\left. \begin{aligned} S &= (\bar{a}b + a\bar{b})p + (\bar{a}\bar{b} + ab)p = S'p + S''p \\ P &= ab + \bar{a}bp + a\bar{b}p = ab + S'p = P' + P'' \end{aligned} \right\} \quad (8)$$

Из выражений (8) следует, что полный двоичный сумматор может быть реализован на двух полусумматорах и одном двухвходовом элементе ИЛИ. Соответствующая схема приведена на рис. 5.

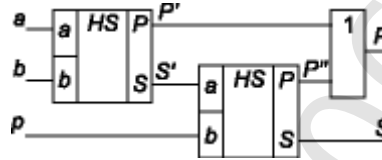


Рис. 5

Из выражения (8) для S также следует:

$$S = a E b E p. \quad (9)$$

*Примечание.* Так как операция E в выражении (9) коммутативна (переменные можно менять местами), то следует, что три входа полного двоичного сумматора абсолютно равноправны и на любой из них можно подавать любую входную переменную. Это полезно помнить, разводя печатные платы, на которых установлены ИС сумматоров.

К настоящему времени разработано большое число схем сумматоров. Доказано (нашим отечественным ученым Вайнштейном), что при использовании только одного инвертора нельзя реализовать полный двоичный сумматор со сложностью  $P_{\text{кв}} < 16$ , а при двух инверторах —  $P_{\text{кв}} < 14$ , где  $P_{\text{кв}}$  — вес по Квайну, используемый как оценка сложности любых комбинационных схем.  $P_{\text{кв}}$  — это общее число входов всех логических элементов схемы **без учёта инверторов**.

	S		P	
a	x	0	x	1
	x	1	0	x
	x	0	x	1
	1	x	x	0
		P		

Рис. 6

Покажем, используя два метода, как была получена рациональная (с использованием только одного инвертора) схема полного двоичного сумматора, являющаяся основой схем ИС сумматоров типа 7480, 155ИМ1 и др.

**Первый метод** основан на использовании значения выходного переноса P как вспомогательной переменной при определении выходной суммы S (табл. 4). В табл. 4 при наборах переменных, являющихся нереальными (например, единичное значение переноса при нулевых значениях всех входных переменных), поставлены безразличные значения (крестик) для функции S, которые можно доопределять произвольным образом.

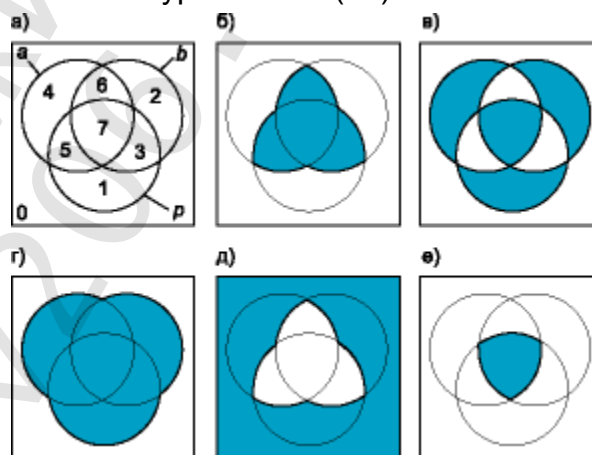
Таблица 4

№ наб.	a	b	p	P	S
--------	---	---	---	---	---

0	0	0	0	0	0
1	0	0	0	1	x
2	0	0	1	0	1
3	0	0	1	1	x
4	0	1	0	0	1
5	0	1	0	1	x
6	0	1	1	0	x
7	0	1	1	1	0
8	1	0	0	0	1
9	1	0	0	1	x
10	1	0	1	0	x
11	1	0	1	1	0
12	1	1	0	0	x
13	1	1	0	1	0
14	1	1	1	0	x
15	1	1	1	1	1

Из карты Карно для функции  $S$  (рис. 6) следует:  
 $S = abr + Pa + Pb + Pr = abr + P(a + b + p)$ . (10)

**Второй метод** основан на применении диаграмм Венна. На рис. 7а показана диаграмма Венна для трех переменных  $a, b, p$ ; области, ограниченные окружностями, соответствуют переменным  $a, b, p$ , а области, обозначенные цифрами от 0 до 7 — соответствующим конъюнкциям (например,  $5 = abr$ ). Область, заштрихованная на рис. 7б, очевидно, соответствует функции  $P = ab + ap + bp$ . Функция  $S$  представлена заштрихованной областью на рис. 7в. Её можно представить суммой произведения функции  $a + b + p$  (рис. 7г) на функцию  $ab + ap + bp$  (рис. 7д) и функции  $abr$  (рис. 7е). Очевидно, что в этом случае получается выражение для  $S$ , аналогичное уравнению (10).



**Рис. 7**

Схема сумматора, реализованного по уравнениям (7) и (10), приведена на рис. 8а. В данной схеме используются многоходовые логические элементы И и ИЛИ. Если использовать только двухходовые элементы, то получаются схемы, приведённые на рис. 8б,в.

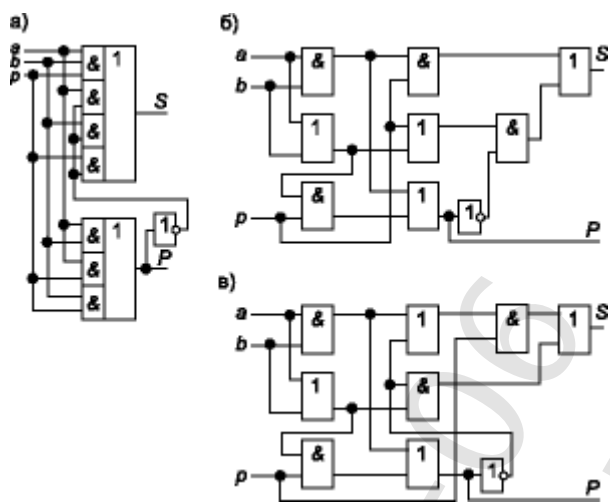


Рис. 8

## Компараторы кодов

Сравнение многоразрядных чисел основывается на проверке равенства цифр чисел. Пусть даны два числа  $a_3 a_2 a_1 a_0$  и  $b_3 b_2 b_1 b_0$ . Сравняются  $b_3$  и  $a_3$ ,  $b_2$  и  $a_2$ ,  $b_1$  и  $a_1$ ,  $b_0$  и  $a_0$ , по результатам сравнения делается вывод: если совпали и третьи цифры, и вторые, и первые, и нулевые, то числа одинаковы. Видно, что результаты сравнения должны подаваться на схему И.

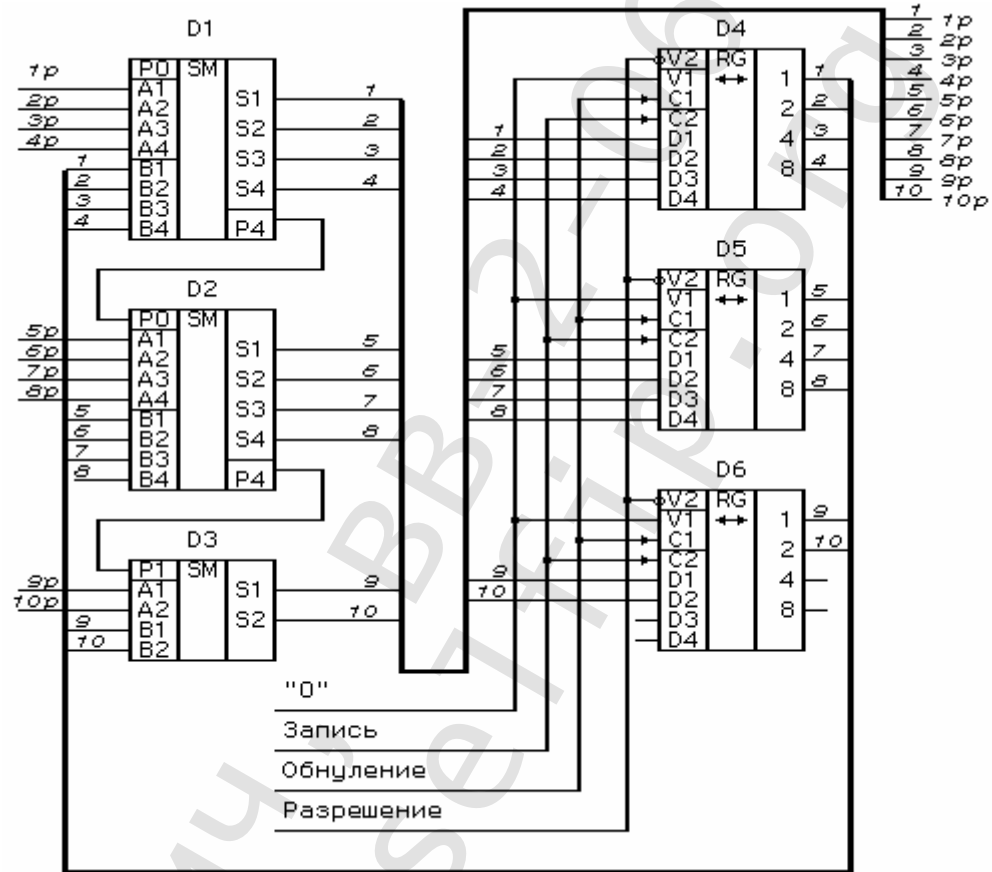


Таблица истинности поразрядного сравнения имеет вид

$a_i$	$b_i$	$F_i$
0	0	1
0	1	0
1	0	0
1	1	1

Переключательная функция в СДНФ:

$$F_i^- = \bar{a}_i \cdot b_i + a_i \cdot \bar{b}_i$$

Результирующая переключательная функция

$$z = F_0 \cdot F_1 \cdot F_2 \cdot F_3$$

С помощью законов логики алгебры возможно изменить выражение на z:

$$z = \overline{F_0 \cdot F_1 \cdot F_2 \cdot F_3} = \overline{F_0} + \overline{F_1} + \overline{F_2} + \overline{F_3} ,$$

где  $F_i = \bar{\alpha}_i \cdot b_i + \alpha_i \cdot \bar{b}_i .$

Переключательную функцию F позволяют реализовать логические двухвходовые элементы "Исключающее ИЛИ" и "Сложение по модулю 2".



На рис.267 показан вариант схемы сравнения

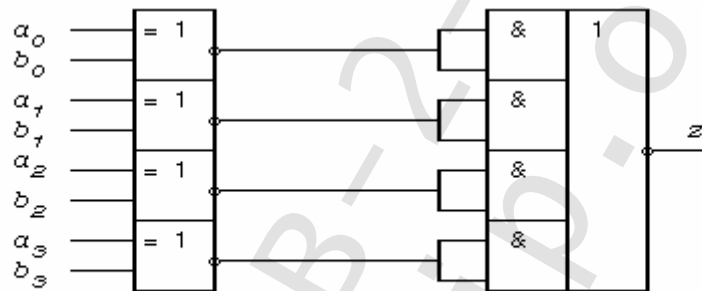
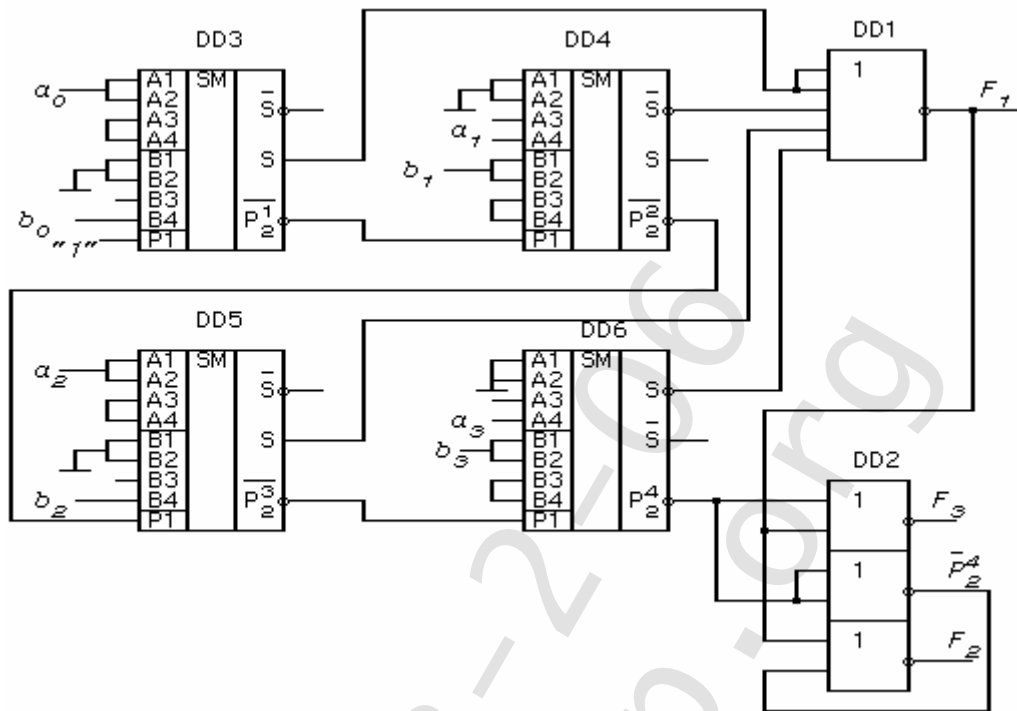


Рис. 267. Реализация схемы сравнения на ИС 155ЛП5 и 155ЛР3

Возможно построение более сложной схемы сравнения (рис.269), которая определяет равенство чисел, а также, какое из чисел больше. Числа A и B подаются на вычитающее устройство на основе сумматоров типа ИМ1. Если числа одинаковы, то результат будет равен нулю. Выходы сумматоров подключены к схеме ИЛИ - НЕ, на выходе которой сигнал логической единицы получится, если на нее подаются все нули. Если  $A > B$ , то на выходе старшего сумматора обязательно возникает сигнал переноса. Работу устройства можно описать следующей таблицей истинности:

$A, B$	$F_2^A$	$F_1$	$F_2$	$F_3$
$A = B$	Н	1	0	0
$A > B$	1	0	1	0
$A < B$	0	0	0	1

Рис. 268



DD1 - 133 ЛЕ3, DD2 - 133ЛЕ1, DD3 - 133 ИМ1

Выход  $F_1$  уже получен, сформируем выход  $F_2$ , зная  $P$  и  $F_1$ .

$$F_2 = P_2^A \cdot \bar{F}_1 = \overline{\bar{P}_2^A + F_1},$$

т.е. выход  $F_2$  получается при подаче сигналов с выходов  $F_2$  и  $P_2^A$  на ЛЭ ИЛИ-НЕ. В этом случае информацию о числах  $A$  и  $B$  несет двоичный код  $F_1F_2$ : при  $F_1F_2 = 10$   $A = B$ , при  $F_1F_2 = 01$   $A > B$ , при  $F_1F_2 = 00$   $A < B$ .

Если построить третий выход  $F_3$ , то информацию о числах  $A$  и  $B$  можно получить по наличию логической 1 на одном из выходов:

$$F_3 = \bar{P}_2^A \cdot \bar{F}_1 = \overline{P_2^A + F_1}.$$

Примечание. Особенностью построения вычитающего устройства (см.7.1) является изменение значения полученного на выходах переноса и суммы. Для нечетных сумматоров все соответствует обозначению, для четных - результат суммирования получается при инверсном выходе, а выход переноса становится прямым.

Поэтому при построении пятиразрядной схемы сравнения выходы  $F_1$  и  $F_2$  поменяются местами.

В настоящее время промышленностью выпускаются готовые четырехразрядные схемы сравнения чисел (рис.270,271, таблица 45).

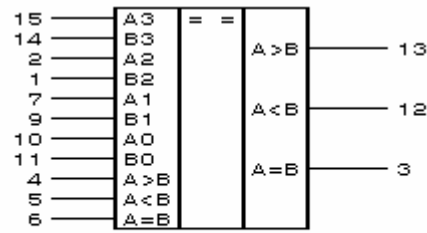


Рис. 270.

... (К134, КР134) СП1

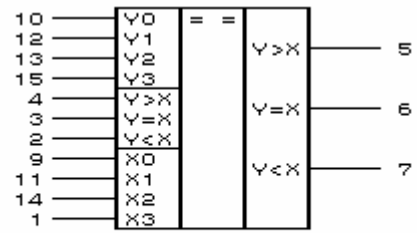


Рис. 271.

(530, 531, 533, К555, 1533) СП1

Таблица 45

Условное обозначение ИС	P <sub>пот</sub> , мВт	t <sub>с</sub> <sup>1/0</sup> , нс	t <sub>с</sub> <sup>0/1</sup> , нс	Число Вывод. ИС	Выводы питания	
					+ 5В	Общ.
К134СП1	45	200	200	16	16	8
КР134СП1	45	200	200	16	16	8
530СП1	550	17	18	16	16	8
К555СП1	100	30	36	16	16	8
1533СП1	55	40	40	16	16	8

## Арифметико-логическое устройство

### Общие сведения, функции и классификация

Арифметико-логическое устройство функционально можно разделить на две части :

- а) микропрограммное устройство (устройство управления), задающее последовательность микрокоманд (команд);
- б) операционное устройство (АЛУ), в котором реализуется заданная последовательность микрокоманд (команд).

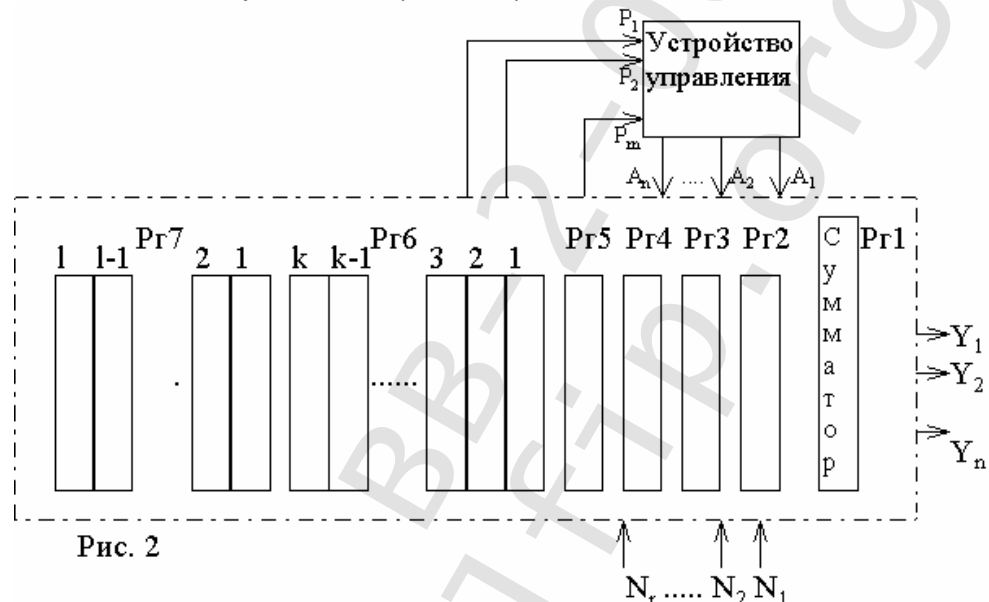


Рис. 2

Структурная схема АЛУ и его связь с другими блоками машины показаны на рисунке 2. В состав АЛУ входят регистры  $Rg1 - Rg7$ , в которых обрабатывается информация, поступающая из оперативной или пассивной памяти  $N_1, N_2, \dots, N_s$ ; логические схемы, реализующие обработку слов по микрокомандам, поступающим из устройства управления.

Закон переработки информации задает микропрограмма  $M$ , которая записывается в виде последовательности микрокоманд  $A_1, A_2, \dots, A_{n-1}, A_n$ . При этом различают два вида микрокоманд: внешние, то есть такие микрокоманды, которые поступают в АЛУ от внешних источников и вызывают в нем те или иные преобразования информации (на рис. 2 микрокоманды  $A_1, A_2, \dots, A_n$ ), и внутренние, которые генерируются в АЛУ и воздействуют на микропрограммное устройство, изменяя естественный порядок следования микрокоманд. Например, АЛУ может генерировать признаки в зависимости от результата вычислений  $\varphi, \omega, \Theta$  и др. ( $\varphi$  – признак переполнения,  $\omega$  – признак отрицательного числа,  $\Theta$  – признак равенства 0 всех разрядов числа). На рис. 2 эти микрокоманды обозначены  $p_1, p_2, \dots, p_m$ .

Результаты вычислений из АЛУ передаются по кодовым шинам записи  $y_1, y_2, \dots, y_s$ , в ОЗУ.

Функции регистров, входящих в АЛУ:

$Rg1$  – сумматор (или сумматоры) – основной регистр АЛУ, в котором образуется результат вычислений;

$Rg2, Rg3$  – регистры слагаемых, сомножителей, делимого или делителя (в зависимости от выполняемой операции);

$Rg4$  – адресный регистр (или адресные регистры), предназначен для запоминания (иногда и формирования) адреса операндов и результата;

$Rg6$  –  $k$  индексных регистров, содержимое которых используется для формирования адресов;



Rг7 – I вспомогательных регистров, которые по желанию программиста могут быть аккумуляторами, индексными регистрами или использоваться для запоминания промежуточных результатов.

Часть операционных регистров является программно-доступной, то есть они могут быть адресованы в команде для выполнения операций с их содержимым. К ним относятся :

- сумматор,
- индексные регистры,
- некоторые вспомогательные регистры.

остальные регистры программно-недоступные, так как они не могут быть адресованы в программе . Операционные устройства можно классифицировать по виду обрабатываемой информации, по способу обработки информации и логической структуре. Подробная классификация АЛУ показана на рис. 3.

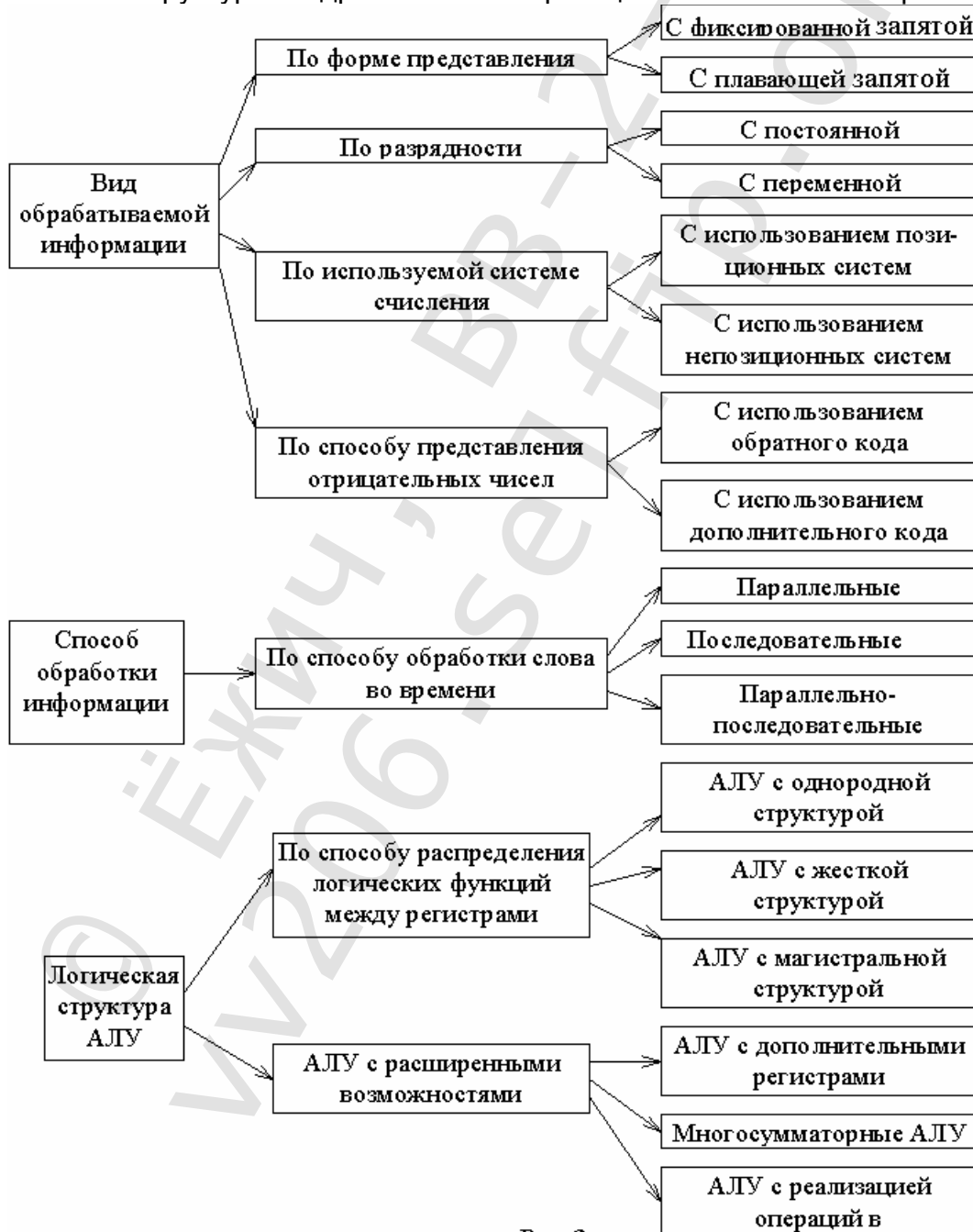


Рис. 3

Сложность логической структуры АЛУ в определенной степени можно охарактеризовать количеством отличающихся друг от друга микроопераций,

необходимых для выполнения всего комплекса задач, поставленных перед АЛУ. На входе каждого регистра собраны соответствующие логические схемы, обеспечивающие такие связи между регистрами, что позволяют реализовать заданный набор микроопераций.

Выполнение операций над словами сводится к выполнению последовательности микрокоманд, которые управляют передачей слов в АЛУ и действиями по преобразованию слов. Порядок выполнения микрокоманд определяется алгоритмом выполнения операций. Следовательно, связи между регистрами АЛУ и функции, которые должны выполнять регистры, зависят в основном от принятой методики выполнения операций : арифметических, логических и специальной арифметики.

Перечень операций, выполняемых в АЛУ, зависит от назначения цифровой вычислительной машины и от функций, выполняемых АЛУ при обеспечении работы остальных устройств машины. При представлении операций в виде последовательностей микроопераций АЛУ должно состоять из элементов; реализующих эти микрооперации.

Таким образом, структура АЛУ определяется набором микроопераций, необходимых для выполнения заданных арифметических, логических и специальных операций , а задачу построения АЛУ можно свести к задаче определения набора микроопераций, который позволяет составить микропрограмму любой из заданных операций. Такой набор легко получить, если записать микропрограммы всех операций, выполняемых в АЛУ, и выбрать из них все микрооперации, входящие в микропрограммы хотя бы один раз. Однако, если при этом алгоритм операций выбирать произвольно, то количество микроопераций, входящих в полный набор, может оказаться слишком большим и, следовательно, АЛУ будет сложным.

Для получения более простой схемы АЛУ алгоритмы арифметических и логических операций следует выбирать из условия получения минимального набора микроопераций. При этом необходимо учитывать требование обеспечения заданного быстродействия АЛУ: слишком ограниченный набор микроопераций может привести к “длинным микропрограммам некоторых операций”, что увеличивает время выполнения данных операций.

### **Алгоритмы сложения (вычитания) и умножения в АЛУ**

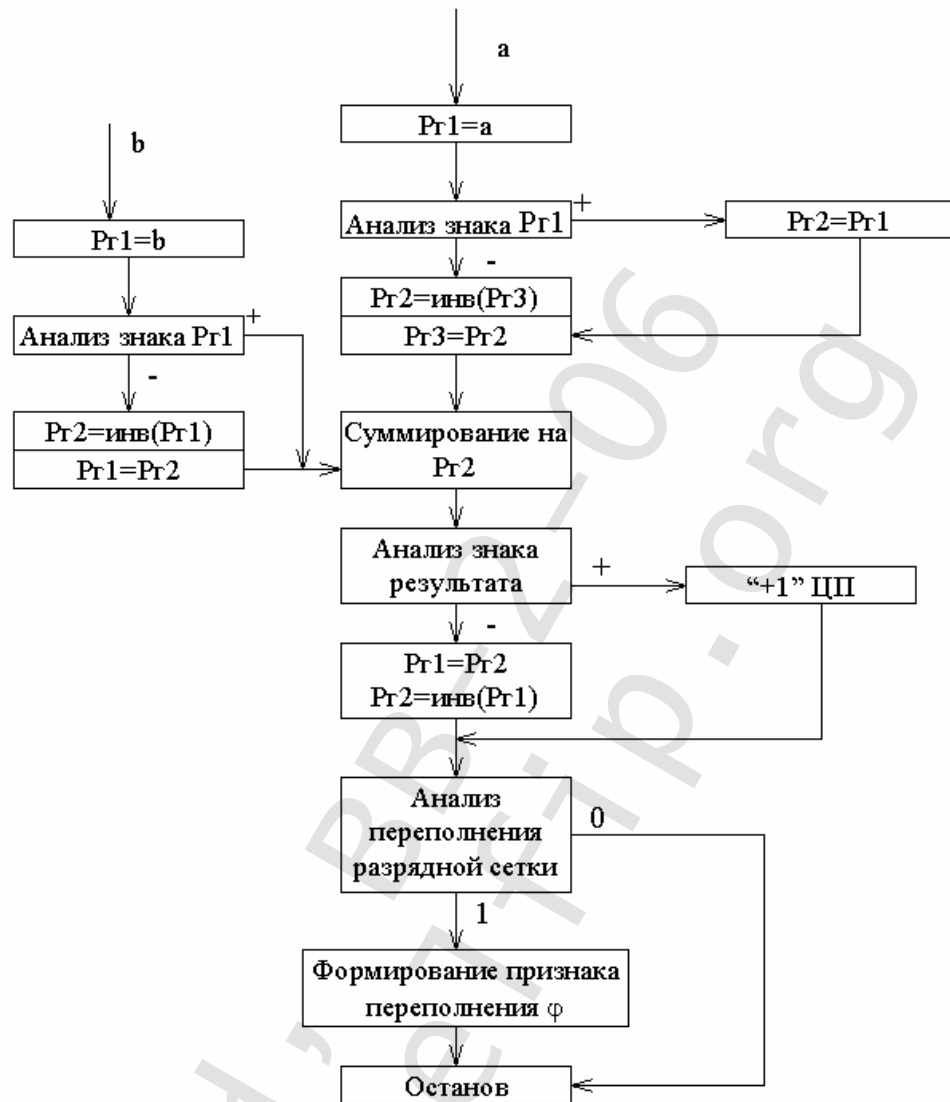


Рис.4

Структурная схема микропрограммы сложения показана на рис. 4. Выполнение этого алгоритма состоит в следующем:

Первое слагаемое  $a$  устанавливается на  $Pr1$ , анализируется его знак: если знак отрицательный, то операнд инвертируется и передается на  $Pr3$ , если положительный - передается без инверсии через  $Pr2$  на  $Pr3$

Второе слагаемое также устанавливается на  $Pr1$  и анализируется его знак: если знак отрицательный, то операнд инвертируется, если положительный - сразу начинается суммирование операндов на  $Pr2$  ( сумматоре )

После суммирования анализируется знак результата: если результат отрицательный, то он инвертируется, если положительный - добавляется "+1" ЦП к младшему разряду результата и выполняется анализ признаков переполнения

В случае переполнения разрядной сетки машины формируется признак переполнения  $\phi$ , если переполнение отсутствует, то выполняется переход на конец микропрограммы сложения.

Для того, чтобы структурная схема, показанная на рис. 4 могла выполнять операцию вычитания, достаточно перед выполнением операции проинвертировать знак второго слагаемого.

Теперь рассмотрим алгоритм умножения. Умножение двоичных чисел с фиксированной запятой можно свести к последовательности сдвигов и сложений.. Наиболее удобен следующий алгоритм: умножение начинается с младших

разрядов множителя, который сдвигается вправо, сумма частичных произведений также сдвигается вправо, множимое - неподвижно. На рис. 5 показана графическая интерпретация этого алгоритма.



Рис. 5

В начале операции все регистры устанавливаются в нулевое состояние.

Множимое и множитель располагаются в определенных регистрах, предусматриваются также регистры, в которых образуется сумма частичных произведений.

Анализируется младший разряд множителя: если он имеет значение "1", то к сумме частичных произведений прибавляется множимое.

Производится сдвиг суммы частичных произведений и множителя на один разряд вправо.

Действия 3 и 4 повторяются  $n$  раз ( $n$  - разрядность сомножителей)

Структурная схема микропрограммы умножения показана на рис. 6. Ввиду громоздкости деление здесь не рассматривается.

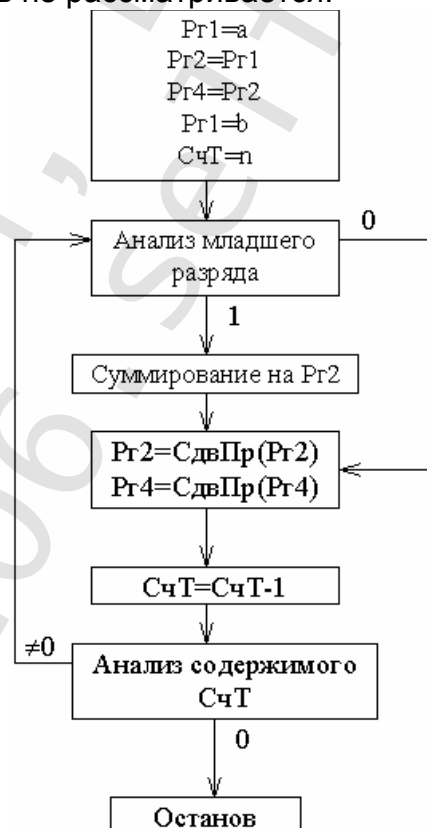


Рис.6