

# Информатика

## Курс лекций часть 4 - 3

Масловский Владимир Михайлович, к.т.н., доцент кафедры ИУ-10  
РУНЦ «Безопасность» МГТУ им. Р.Э. Баумана, тел. 499 263 6794,  
E-mail: [zi@bmstu.ru](mailto:zi@bmstu.ru), [mvm481@rambler.ru](mailto:mvm481@rambler.ru)

# Основы элементной базы цифровых автоматов

Для согласования параметров элементов ТТЛ и И<sup>2</sup>Л на входах микросхем И<sup>2</sup>Л в качестве трансляторов включаются буферные инверторы (рисунок 2.4, а).

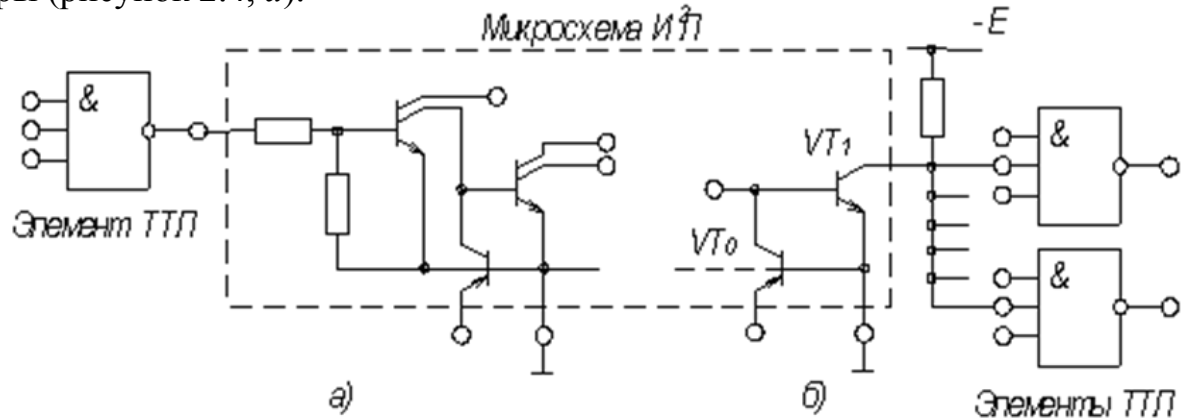


Рисунок 2.4 – Входной (а) и выходной (б) трансляторы И<sup>2</sup>Л

Сопротивление  $R_p$ ,  $R_f$  выбираются из условия отпирания транзистора  $VT1$  при  $U_{ex} = U_{ex}^1$ . Выходным транслятором служит элемент И<sup>2</sup>Л с подключенным внешним резистором  $R_e$  (рисунок 2.4, б), сопротивление которого выбирается из условия:

$$R_e \leq (E - U_{ex}^1) / I_{нпх}^1 \quad (2.3)$$

где  $U_{ex}^1$  – уровень «1», который требуется обеспечить во внешней цепи;  $I_{нпх}^1$  – максимальный ток, отпираемый нагрузкой.

Транзисторы  $VT_0$  и  $VT_1$  выходного транслятора должны иметь достаточно высокие значения коэффициентов  $\alpha$  и  $\beta$ , чтобы обеспечивать выполнение условия насыщения

(2.3) при токе  $I_{нпх}^1 = E / R_e + I_{нпх}^0$ , где  $I_{нпх}^0$  – максимальная величина тока, поступающая от нагрузки при  $U_{вых} = U^0$ . При использовании данных трансляторов микросхемы И<sup>2</sup>Л и ТТЛ совместимы по входным уровням  $U^0$ ,  $U^1$  т.е., выходные сигналы одной из них можно непосредственно использовать для переключения другой и наоборот. Поэтому микросхемы И<sup>2</sup>Л и ТТЛ часто используются в цифровых устройствах совместно.



## Основы элементной базы цифровых автоматов

Большинство схем трансляторов имеет существенно более высокие значения потребляемой мощности  $P_{кр}$  и площади  $A_{кр}$ , чем элементы внутренней структуры БИС. Поэтому при их применении возрастает площадь кристалла и рассеиваемая мощность:

$$P_{\text{кф}} = N_{\text{эл}} P_{\text{эл}} + n_{\text{с}} (a_P - 1) \quad (2.4) \quad A_{\text{кф}} = N_{\text{эл}} A_{\text{эл}} + n_{\text{с}} (a_A - 1) \quad (2.5)$$

где  $n_{\text{с}}$  – число логических входов и выходов микросхемы; значения  $a_P = P_{\text{кф}} / P_{\text{эл}}$ ,  $a_A = A_{\text{кф}} / A_{\text{эл}}$  достигает 3–5. Для уменьшения  $P_{кр}$  и  $A_{кр}$  при проектировании БИС стремятся обеспечить чтобы  $n_{\text{с}} \ll N_{\text{эл}}$ .

В ЦИМ на элементах МДПТЛ и КМДПТЛ возможность уменьшения потребляемой мощности ограничивается значениями порогового напряжения  $E \approx (2-3)U_0$ . В этих микросхемах снижение  $E$  и соответственно перепада  $U_n$  не приводит к повышению быстродействия, так как наряду с уменьшением перепада напряжения  $\Delta U_0 = U \cong E$  на паразитной емкости  $C_n$  возрастают постоянные времени ее перезаряда  $\tau \sim 1/b(E - U_0)$ , где  $b$  – удельная крутизна МДП-транзисторов. Поэтому в БИС и СБИС на МДП-транзисторах все элементы обычно имеют одинаковое напряжение питания  $E$ .

Для внутренних элементов СБИС с целью получения высокой плотности их размещения применяют простые схемы без сложных выходных каскадов, поскольку такие элементы работают при известных заранее и фиксированных нагрузках. Для связи с внешними нагрузками в составе элементов СБИС имеются обычно специальные периферийные буферные элементы, обеспечивающие более высокую выходную мощность.

### 3 ПРЕОБРАЗОВАТЕЛЬНЫЕ УСТРОЙСТВА

Преобразовательные устройства – это функциональные узлы комбинационного типа: преобразователи кодов, мультиплексоры и демультимплексоры, шифраторы и дешифраторы, компараторы и прочие.

#### *3.1 Преобразователи кодов*

Преобразователи кодов изменяют вид кодирования данных. В ЭВМ, цифровой технике используют несколько форм представления информации. Входные и выходные устройства оперируют с привычными человеку десятичными кодами, вычисления ведутся в двоичных кодах, промежуточной формой представления чисел служит двоично-десятичный код. Используют и другие представления данных. В широком смысле слова к преобразователям кодов можно отнести многие цифровые устройства, в частности шифраторы и дешифраторы, однако по традиции эти устройства выделены в отдельные классы.

Двоичные числа могут быть представлены в прямом, обратном или дополнительном кодах



### ***3.1.1 Преобразование прямого кода в обратный***

Преобразование прямого кода в обратный реализуется сложением по модулю 2 значений кода данного разряда и сигнала управления преобразователем (рисунок 3.1, *a*). При этом нулевое значение сигнала управления обеспечивает прохождение на выход схемы прямого кода числа. При единичном значении сигнала управления каждый разряд выходного кода будет инверсией соответствующего разряда входного кода. Если преобразуется число, содержащее знаковый разряд, то его значение может быть непосредственно использовано как сигнал управления.

### ***3.1.2 Преобразование прямого кода в дополнительный***

Преобразование прямого кода в дополнительный (дополнение числа до  $2^n$ ) реализуется сложнее. В этом случае операция преобразования не является поразрядной, и для определения значения кода в каком-либо разряде требуется анализ значений других разрядов. Для получения дополнительного кода нужно проинвертировать все разряды преобразуемого кода и затем прибавить к результату единицу (рисунок 3.1, *б*).

## Основы элементной базы цифровых автоматов

Сопоставление прямого и дополнительного кодов показывает, что последний отличается от первого инвертированием старших разрядов от  $n-1$  до  $i+1$  включительно, где  $i$  – номер первого справа разряда, содержащего единицу. Остальные разряды остаются неизменными. Например, для прямого кода 10100100 дополнительным будет код 01011100. Аналитически это правило записывается в виде:

$$a_{\text{вых}i} = a_{\text{вх}i} \oplus (a_{\text{вх}i-1} + a_{\text{вх}i-2} + \dots + a_{\text{вх}0})$$

согласно чему для получения  $i$ -го разряда дополнительного кода нужно сложить по модулю 2 исходный код этого разряда с дизъюнкцией всех предыдущих (младших) разрядов. Знаковый разряд преобразуемого кода может использоваться как управляющий сигнал (рисунок 3.1, б).

С ростом объемов постоянных запоминающих устройств стало рациональным выполнять преобразователи кодов на основе памяти (табличным методом). Таблица преобразования двоично-десятичных кодов в двоичные в пределах полутора тетрад, не включающая младших разрядов, одинаковых для обоих кодов, имеет следующий вид:

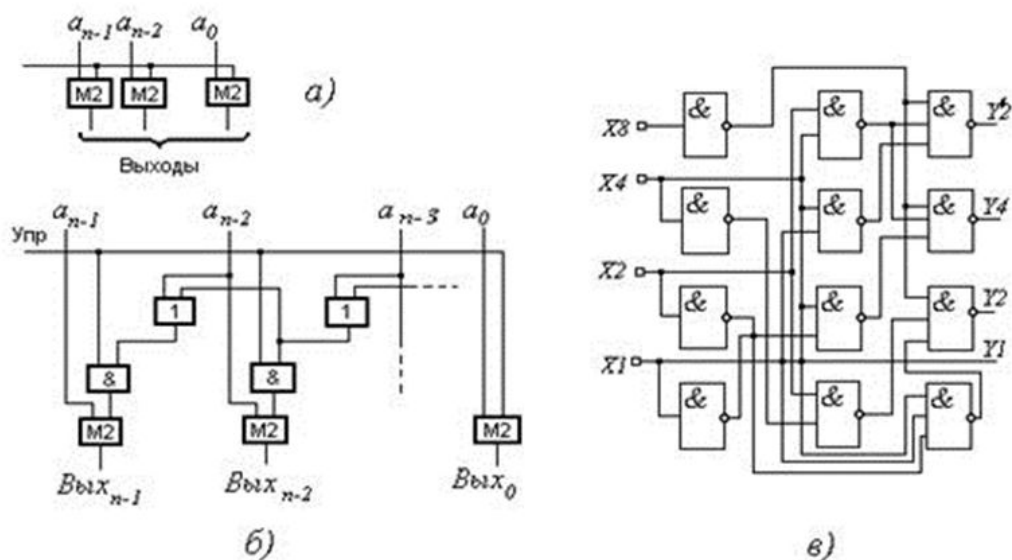


Рисунок 3.1 – Схемы преобразователей прямого кода в обратный (а), дополнительный (б) и кода 8421 > 2421 (в)



## Основы элементной базы цифровых автоматов

Таблица 3.1 – Преобразования двоично-десятичных кодов

Десятичные числа	Входы	Строб	Выходы
	$a_5 a_4 a_3 a_2 a_1$		$z_5 z_4 z_3 z_2 z_1$
0,1	0 0 0 0 0	1	0 0 0 0 0
2,3	0 0 0 0 1	1	0 0 0 0 1
4,5	0 0 0 1 0	1	0 0 0 1 0
6,7	0 0 0 1 1	1	0 0 0 1 1
8,9	0 0 1 0 0	1	0 0 1 0 0
10,11	0 1 0 0 0	1	0 0 1 0 1
12,13	0 1 0 0 1	1	0 0 1 1 0
14,15	0 1 0 1 0	1	0 0 1 1 1
.....	.....	.....	.....
38,39	1 1 1 0 0	1	1 0 0 1 1
Любые	X X X X X	0	1 1 1 1 1

Условное обозначение преобразователя кодов показано на рисунке 3.2, а.

Для преобразования кодов большей разрядности собираются схемы с каскадированием рассмотренных шестиразрядных преобразователей.

Схемы для преобразования двух и трех тетрад двоично-десятичного кода (рисунок 3.2, б, в) содержат соответственно 2 и 6 микросхем ПЗУ и имеют логическую глубину 2 и 5.

# Основы элементной базы цифровых автоматов

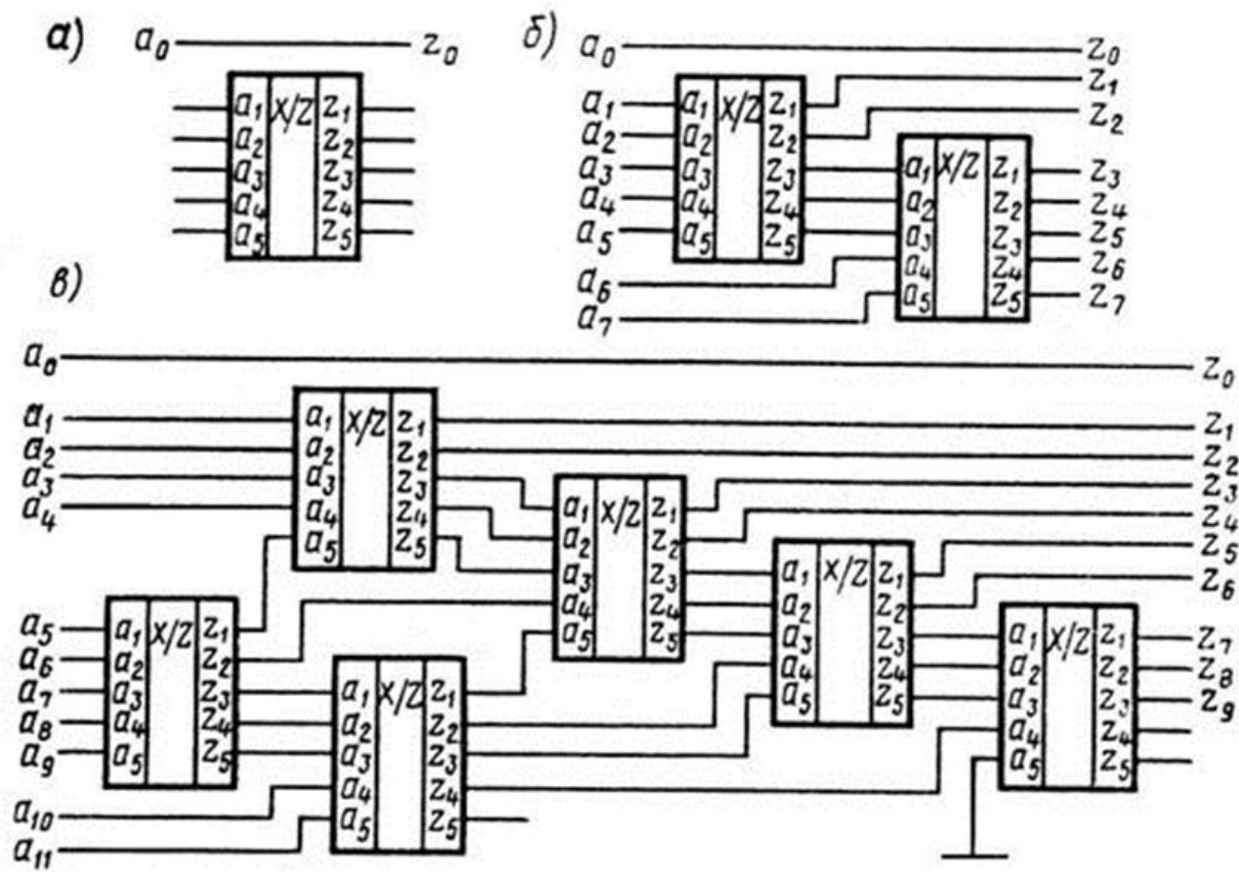


Рисунок 3.2 – Условное обозначение преобразователя кодов (а), и схемы преобразователей для двух (б) и трех (в) тетрад двоично-десятичного кода

Покажем правильность функционирования схемы на рисунке 3.2, б. Анализ работы схемы требует знания таблицы ее функционирования.



## Основы элементной базы цифровых автоматов

Выше была для краткости приведена лишь часть таблицы функционирования шестиразрядного преобразователя, однако при необходимости легко получить любую строку этой таблицы, руководствуясь следующей закономерностью: в пяти первых строках таблицы выходы совпадают со входами; в пяти следующих строках выходное слово есть входное, из которого вычитается 3; в пяти следующих для получения выходного слова нужно вычесть из входного 6; и наконец, в пяти последних строках выход равен входу за вычетом 9.

Возьмем для примера число 55, которое в двоично-десятичном коде выражается словом 01010101 и в двоичном коде словом 00110111. На рисунке 3.3 показаны сигналы, появляющиеся на выходах и входах шестиразрядных преобразователей при подаче на схему двоично-десятичного кода числа 55, свидетельствующие о правильности функционирования схемы.

С помощью каскадирования строятся и схемы большей разрядности. При этом для преобразователя шести тетрад, например, требуется 28 микросхем, а логическая глубина схемы равна 13.

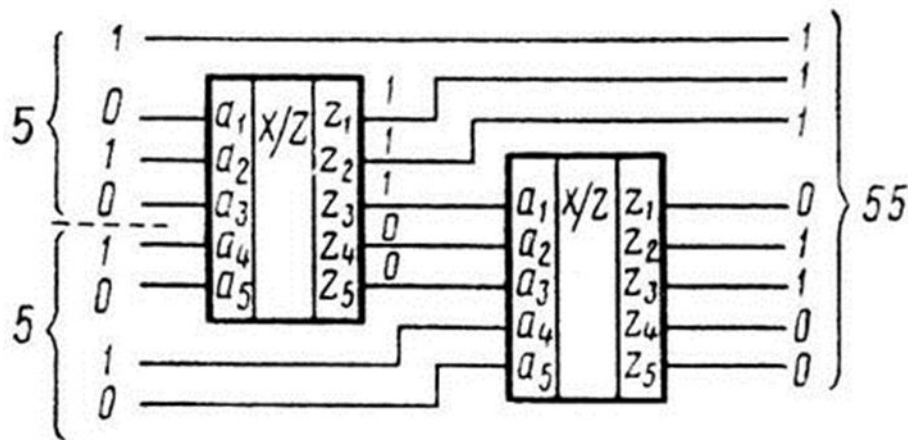


Рисунок 3.3 – Пример преобразования конкретного кода

Выше отмечено, что для преобразователя двоично-десятичного кода в двоичный в пределах полутора тетрад остаются избыточными три выхода ПЗУ. Это позволяет дополнительно реализовать на той же микросхеме преобразование двоично-десятичных кодов в обратные или дополнительные в пределах одной тетрады. Нехватка одного выхода не препятствует решению указанных задач, так как в обоих случаях один из разрядов оказывается одинаковым для входных и выходных кодов

# Основы элементной базы цифровых автоматов

(см. таблицу функционирования для обоих преобразований, имеющую следующий вид):

Таблица 3.2 – Преобразование двоично-десятичного кода в дополнительные

Двоично-десятичное слово	Входной код				Дополнение до 9			Дополнение до 10		
	$a_4$	$a_3$	$a_2$	$a_1$	$z_8$	$z_7$	$z_6$	$z_8$	$z_7$	$z_6$
0	0	0	0	0	1	0	0	0	0	0
1	0	0	0	1	1	0	0	1	0	0
2	0	0	1	0	0	1	1	1	0	0
3	0	0	1	1	0	1	1	0	1	1
4	0	1	0	0	0	1	0	1	1	0
5	0	1	0	1	0	1	0	0	1	0
6	0	1	1	0	0	0	1	1	0	0
7	0	1	1	1	0	0	1	0	1	1
8	1	0	0	0	0	0	0	1	0	0
9	1	0	0	1	0	0	0	0	0	1

При преобразовании двоично-десятичного кода в дополнение до девяти общими для входного и выходного кодов оказываются вторые разряды справа, а при преобразовании в дополнение до десяти — первые. Поэтому можно применить схемы на рисунке 3.4, где программированию подлежат всего три выхода, которые не были заняты в преобразователе шестизрядного двоично-десятичного кода в двоичный. Таблица программирования ПЗУ определяется столбцами, отмеченными в таблице символами выходов  $z_6, z_7, z_8$ .

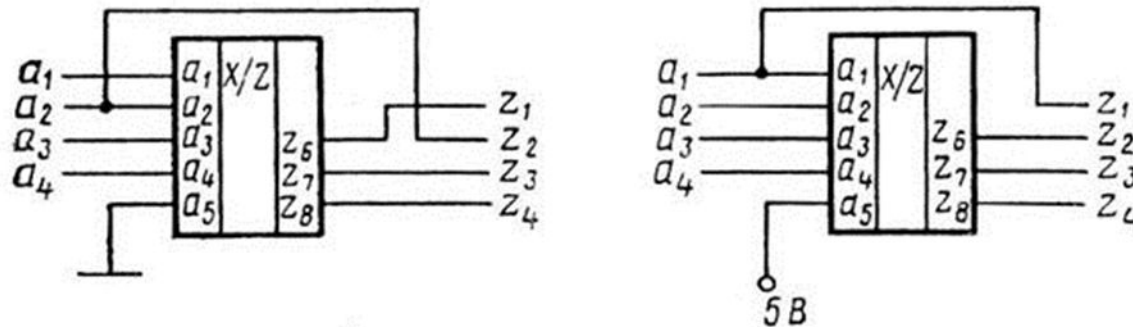


Рисунок 3.4 – Схемы для построения преобразователей кодов с дополнительными возможностями

## Основы элементной базы цифровых автоматов

При работе в режимах преобразования двоично-десятичного кода в дополнение до 9 или 10 вход  $a_5$  остается не занятым информационными сигналами и используется как управляющий. Если на него подать нуль, то получится набор входных кодов, по которым запрограммирован один из видов преобразования (например, в дополнение до 9), а если подать единицу, то получится набор входных кодов, по которым запрограммирован другой вид преобразования (рисунок 3.4).

На ПЗУ реализуются и преобразователи двоичных кодов в двоично-десятичные. При этом на входе можно задать шестиразрядные коды (пять входов имеет ПЗУ, а младший разряд — общий для обоих кодов и не нуждается в преобразовании) и получить на выходе двоично-десятичный код в пределах полутора тетрад. Для преобразования слов большей разрядности составляются схемы из нескольких шестиразрядных преобразователей

## *3.1.3 Преобразователи одного кода в другой*

Широкое применение в вычислительной технике находят преобразователи кодов, преобразующие числовую информацию из одной двоичной формы в другую (например, код 8421 в коды: 2421, Грея, дополнение до 9 (10), 5421, 7421, Джонсона (таблица 3.3) и другие ).

Особенность кода Грея состоит в том, что при переходе к каждому последующему в коде изменяется значение только одного двоичного разряда. В коде Джонсона переход к последующему числу осуществляется путём последовательной замены 0 на 1, начиная справа до заполнения всех разрядов 1, а затем заменой 1 на 0, до заполнения нулями. Коды с дополнением до 9, 10 с «избытком 9» используются для выполнения сложения и вычитания двоично-десятичных чисел. Для преобразования кодов можно пользоваться двумя методами: методом, основанным на преобразовании исходного двоичного кода в десятичный и последующего преобразования десятичного представления в требуемый двоичный код; метод синтеза логического устройства комбинационного типа непосредственно реализующего данное преобразование.

# Основы элементной базы цифровых автоматов

Таблица 3.3 – Преобразование одного кода в другой

номер	8421	7421	5421	Код Айкена 2421	Код Грея	Код с избытком 3 $N+3$	Дополнение до 9 $9-N$	Код Джонсона	Дополнение до 10 $10-N$
0	<b>0000</b>	0000	0000	<b>0000</b>	0000	0011	1001	00000	1010
1	<b>0001</b>	0001	0001	<b>0001</b>	0001	0100	1000	10000	1001
2	<b>0010</b>	0010	0010	<b>0010</b>	0011	0101	0111	11000	1000
3	<b>0011</b>	0011	0011	<b>0011</b>	0010	0110	0110	11100	0111
4	<b>0100</b>	0100	0100	<b>0100</b>	0110	0111	0101	11110	0110
5	<b>0101</b>	0101	1000	<b>1011</b>	0111	1000	0100	11111	0101
6	<b>0110</b>	0110	1001	<b>1100</b>	0101	1001	0011	01111	0100
7	<b>0111</b>	1000	1010	<b>1101</b>	0100	1010	0010	00111	0011
8	<b>1000</b>	1001	1011	<b>1110</b>	1100	1011	0001	00011	0010
9	<b>1001</b>	1010	1100	<b>1111</b>	1101	1100	0000	00001	0001

Рассмотрим построение преобразователя кода 8421 в код 2421 на основе второго метода. В таблице 3.3 приведено соответствие комбинаций обоих кодов. Каждую из переменных  $y'_8, y_4, y_2, y_1$  можно рассматривать функцией  $x_8, x_4, x_2, x_1$  и, следовательно, можно представить через аргументы соответствующей логической функции в СДНФ, которую легко минимизировать с помощью таблиц Вейча. На рисунке 3.5 приведена логическая схема преобразователя кодов 8421 в 2421 на элементах И-НЕ.

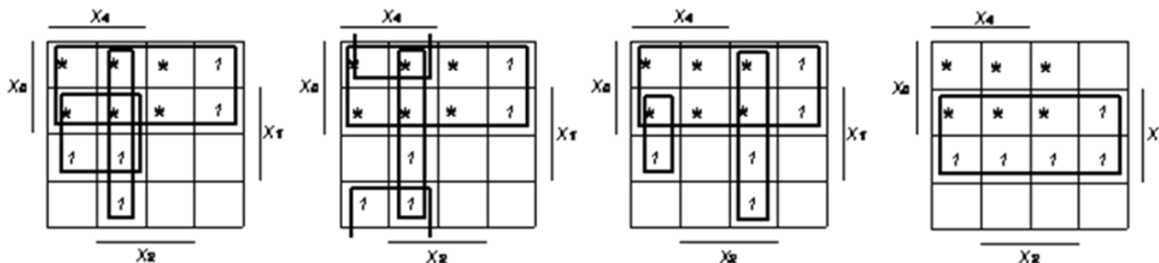


Рисунок 3.5 – Логическая схема преобразователя кодов 8421 в 2421 на элементах И-НЕ



## 3.1.4 Программируемая логическая матрица (ПЛМ)

ПЛМ – комбинационная ЦИМ, предназначенная для преобразования кодов, выполнения произвольных логических операций, генерирования булевых функций или микропрограммного управления. ПЛМ состоит из двух ступеней: программируемого дешифратора и программируемого шифратора. Элементы первой ступени образуют конъюнкции входных переменных, входящих в МДНФ (или СДНФ) выходных функций. Элементы второй ступени выполняют дизъюнкции полученных конъюнктивных членов, формируя заданные выходные функции. Число элементов первой ступени равно общему числу различных конъюнктивных членов в МДНФ всех выходных функций. Число элементов второй ступени равно числу выходных функций. Число переменных, входящих в один конъюнктивный член, ограничивается допустимым числом  $M$  входов элементов первой ступени. Максимальное число конъюнктивных членов в выходной функции ограничивается допустимым числом  $M$  входов элементов второй ступени. На рисунке 3.6 изображена функциональная схема ПЛМ и принципиальная электрическая схема на  $n$ -МОП транзисторах.

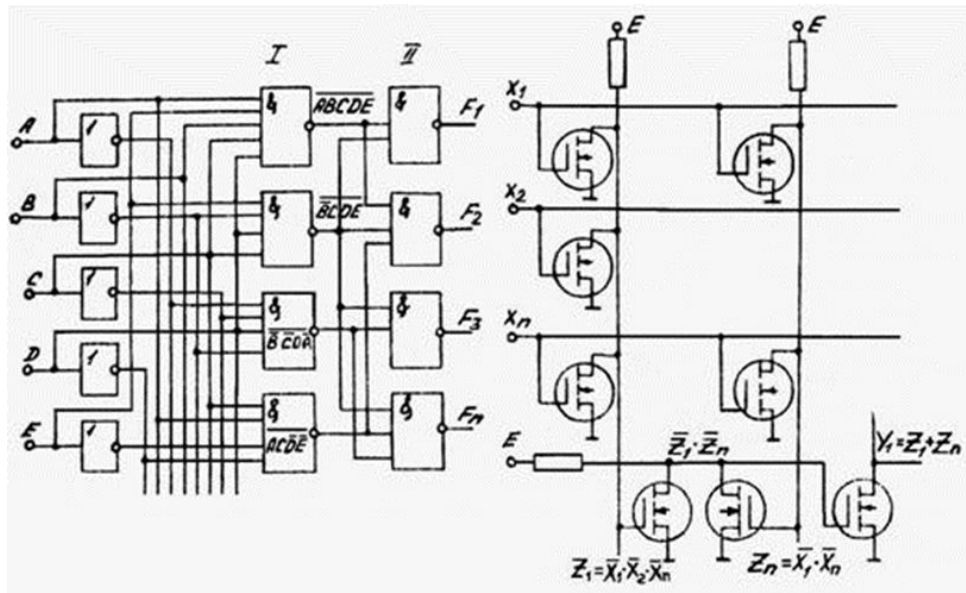


Рисунок 3.6 – Функциональная и принципиальная электрическая схема ПЛМ на  $n$ -МОП транзисторах



# Основы элементной базы цифровых автоматов

## 3.2 Дешифраторы и шифраторы

Дешифраторы и шифраторы по существу принадлежат к числу преобразователей кодов. С понятием шифрации связано представление о сжатии данных, с понятием дешифрации — обратное преобразование.

### 3.2.1 Дешифраторы

Дешифраторы – устройства, предназначенные для преобразования двоичного  $n$ -разрядного кода в комбинацию управляющих сигналов.

В условных обозначениях дешифраторов и шифраторов используются буквы  $DC$  и  $CD$  (от слов *decoder* и *coder* соответственно). Двоичные дешифраторы преобразуют двоичный код в код «1 из  $N$ ». Иными словами, в зависимости от входного кода на выходе возбуждается одна из цепей. Так как с помощью  $n$ -разрядного двоичного кода можно отобразить  $2^n$  кодовых комбинаций, число выходов полного дешифратора равно  $2^n$ . Если часть входных наборов не используется, то дешифратор называют неполным и у него  $N_{вых} < 2^n$ .

Функционирование дешифратора описывается системой логических выражений вида:

$$F_i = m_i; \quad i = \overline{0, 2^n - 1}$$

где  $m_i$  – минтермы  $n$  входных переменных.

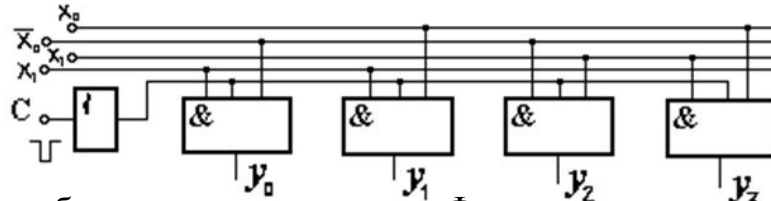


В зависимости от разрядности дешифруемого кода и функциональных возможностей логических интегральных схем дешифратор можно выполнить на основе одноступенчатой (или линейной) и многоступенчатой схем дешифрации. Среди многоступенчатых схем можно выделить прямоугольные (матричные) и пирамидальные схемы построения дешифраторов. Дешифраторы широко используются в устройствах вывода информации и управления

## 3.2.1.1 Линейные дешифраторы

Линейные дешифраторы представляют собой совокупность схем И, формирующих управляющий сигнал только на одном из выходов, в то время как на остальных выходах сигнал отсутствует (рисунок 3.7).

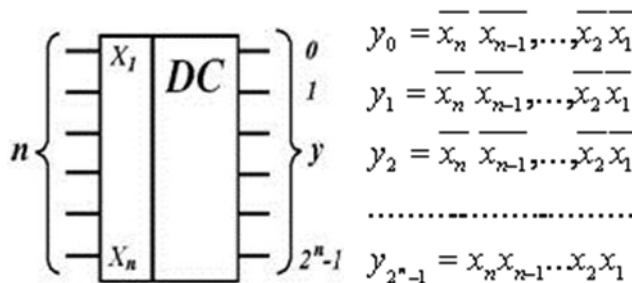
Рисунок 3.7 – Линейный дешифратор на два входа



Поэтому дешифраторы часто называют избирательными схемами. Функционирование дешифратора можно описать с помощью таблицы истинности (таблица 3.4) или логических функций:

Таблица 3.4 – Таблица истинности

$X_n$	$X_{n-1}$	...	$X_2$	$X_1$	$Y_0$	$Y_1$	$Y_2$	...	$Y_{2^n-1}$
0	0	...	0	0	1	0	0	...	0
0	0	...	0	1	0	1	0	...	0
0	0	...	1	0	0	0	1	...	0
...	...	...	...	...	...	...	...	...	...
1	1	...	1	1	0	0	0	...	1



где  $X_1, X_2, \dots, X_n; Y_0, Y_1, \dots, Y_{2^n-1}$  – сигналы на входах и выходах дешифратора. Для линейного дешифратора на  $n$  входов требуется элементов И с  $n$  входами, Линейные дешифраторы наиболее эффективны, когда разрядность входного кода  $n$  не превышает число  $n$  входов схемы И типовом элементе ЦИМ. Быстродействие линейных дешифраторов является наибольшим по сравнению с другими схемами дешифрации, так как время установления  $t_{уст}$  управляющего сигнала на входе дешифратора равно средней задержке  $t_{зд.ср}$  типового элемента ЦИМ ( $t_{уст} = t_{зд.ср}$ ). Основным недостатком – требуется большая нагрузочная способность элементов регистра (триггеров), к которым подключаются входы дешифраторов, равная  $m_{тр} = 2^{n-1}$ .





## Основы элементной базы цифровых автоматов

В дешифраторах часто предусматривается операция стробирования (тактирования), разрешающая выработку выходных сигналов только в определенных интервалах времени. Стробирование может осуществляться введением дополнительного входа параллельно информационным в каждый элемент дешифратора, либо блокированием всех элементов через одну из входных цепей. В последнем случае при нулевом значении сигнала строга искусственно формируются нули в цепях прямого и инверсного значений переменной  $a_i$ , что обеспечивает наличие хотя бы одного нуля в числе входных величин для каждого элемента дешифратора. При этом все выходы приводятся к нулевым значениям, так как любая переменная в прямом или инверсном виде поступает на все элементы дешифратора. При единичном значении строга восстанавливается нормальная цепь передачи переменной  $a_i$  на входы дешифратора

Описанные варианты можно назвать стробированием по выходу и стробированием по входу.

Время установления выходного сигнала  $t_y = \max(t_3^{10}, t_3^{01})$ , т.к. при изменении состояния дешифратора на одном из выходов сигнал возбуждения появляется, а на другом — исчезает. При стробировании по выходу задержка относительно начала разрешающего сигнала

составляет  $t_y = t_3^{01}$  а при стробировании по входу  $t_y = 2t_3 + t_3^{01}$

Нагрузка, создаваемая дешифратором на источник сигналов, вычисляется с учетом того, что каждая переменная присутствует на входе каждого элемента в прямом либо инверсном виде, причем оба вида используются в равной мере. Следовательно, к источникам сигналов  $a_i$

и  $\overline{a_i}$  будет подключено число входов  $K_y = 2^{n-1}$

## Основы элементной базы цифровых автоматов

Одноступенчатый дешифратор – наиболее быстродействующий, но его реализация при значительной разрядности входного слова затруднена, поскольку требует применения логических элементов с большим числом входов (равным  $n+1$  для вариантов со стробированием по выходу) и сопровождается большой нагрузкой на источники входных сигналов. Обычно одноступенчатыми выполняются дешифраторы на небольшое число входов, определяемое возможностями элементов применяемой серии микросхем. Возможность выполнения дешифраторов в одном корпусе ИМС ограничивается большим числом внешних выводов, главную часть которых составляют информационные выходы. Дешифратор на пять входов имеет не менее 40 внешних выводов (5 входов, 32 выхода, вход стробирования, цепи питания и общей точки), если инверсии аргументов вырабатываются внутри схемы дешифратора.

При построении прямоугольного двухступенчатого дешифратора для его реализации требуется

общее количество схем И, определяемая из выражений  $\sum M = 2^n + 2 \cdot 2^{n/2}$  при  $n$  четном и  $\sum M = 2^{(n-1)/2} + 2^{(n+1)/2} + 2^n$

при  $n$  нечетном. При этом требуется  $2^n$  двухвходовых схем И для матричного дешифратора,  $2^x$   $x$ -входовых схем для первого линейного дешифратора и  $2^{n-x}$   $(n-x)$ -входовых схем И для второго линейного дешифратора. При  $n > 4$  прямоугольные дешифраторы примерно в 2 раза экономичнее пирамидальных. Следует отметить, что к нагрузочной способности ЦИМ предконечной ступени прямоугольных дешифраторов предъявляются достаточно высокие требования, т.е.

$$m_{\text{гр}} = 2^{n/2}$$

## 3.2.1.2 Пирамидальные дешифраторы

Пирамидальные дешифраторы на  $n$  входов имеют  $x=n-1$  ступеней, причем на каждой ступени используются только двухвходовые схемы И (рисунок 3.8). Количество элементов И в  $i$ -й ступени составляет  $2^{i+1}$ , где  $i$  – номер ступени. Общее количество схем И для пирамидального дешифратора определяется по формуле

$$\Sigma И = \sum_{i=1}^{n-1} 2^{i+1}$$

Недостаток пирамидальных дешифраторов: различные входы дешифраторов оказывают неравномерную нагрузку на элементы регистров, а большое число ступеней снижает его быстродействие  $t_{уст} = xt_{зд.ср}$ .

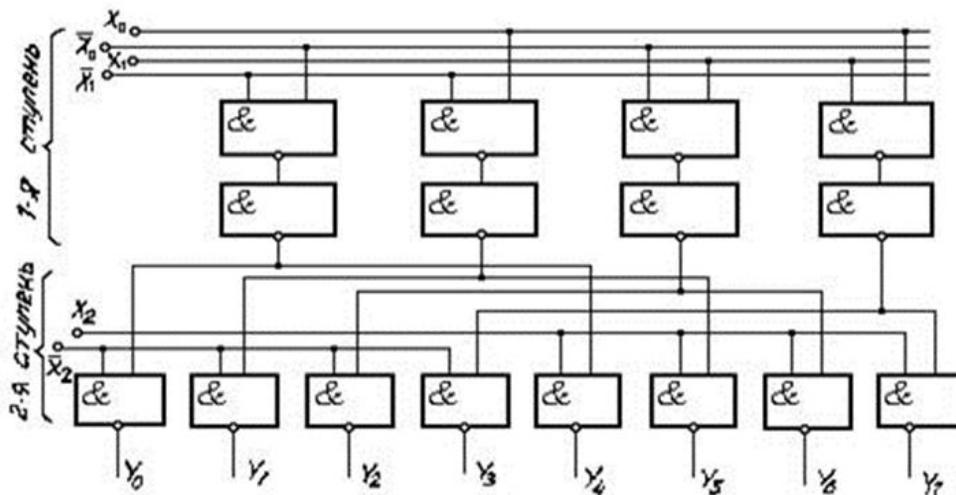


Рисунок 3.8 – Пирамидальный дешифратор



## 3.2.1.3 Прямоугольные дешифраторы

В прямоугольных дешифраторах  $n$  его входов разбиваются на две группы по  $n/2$  переменных в каждой группе при четном  $n$ , при нечетном  $n$  группы должны содержать по  $(n+1)/2$  и  $(n-1)/2$  переменных (рисунок 3.9). Для каждой из двух групп строится линейный дешифратор. Эти дешифраторы составляют первую ступень дешифрации. Затем по матричной схеме с помощью элементов И на два входа каждая выходная шина одного дешифратора объединяется с каждой выходной шиной другого и таким образом получается вторая ступень дешифрации.

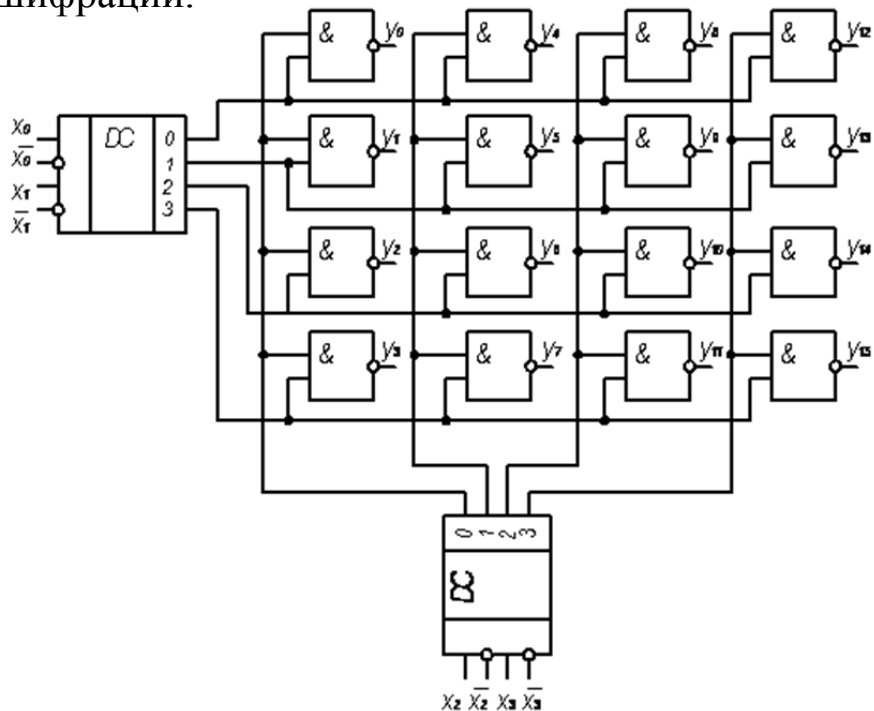


Рисунок 3.9 –  
Прямоугольный дешифратор



### **3.2.1.4 Схема наращивания размерности дешифраторов**

ЦИМ дешифраторов в настоящее время имеют три входа (серии 100, К500, К555) и удвоенные два входа, что позволит построение дешифраторов большой размерности. Рассмотрим на примере дешифрации шестизначного слова на трехходовых стробируемых дешифраторах (рисунок 3.10, *a*) состоит из девяти одинаковых ИМС. Общее стробирование осуществляется по входу  $C$  первого дешифратора (при  $C=0$  на всех выходах первого дешифратора вырабатываются нулевые сигналы, поэтому нули будут вырабатываться и на всех выходах дешифраторов второго яруса). На входы первого дешифратора поданы три старших разряда входного слова, значение которых при  $C=1$  определяет номер возбужденной шины на выходе, отпирающей один из дешифраторов второго яруса по его стробирующему входу. Выбранный таким образом дешифратор расшифровывает три младших разряда слова. Например, для дешифрации слова  $111011 = 59$  имеем на выходе дешифратора первого яруса код  $111$ , что возбуждает его седьмой выход и отпирает  $DC9$ . На входе дешифратора  $DC9$  действует код  $011$ , поэтому единица появится на его третьем выходе, т. е. на  $59$  выходе, что и требуется.

В неполных дешифраторах часть входных наборов не является рабочей (не используется). Это обстоятельство можно рассматривать как возможность упрощения дешифратора — нерабочим наборам можно поставить в соответствие любое значение выходов.

# Основы элементной базы цифровых автоматов

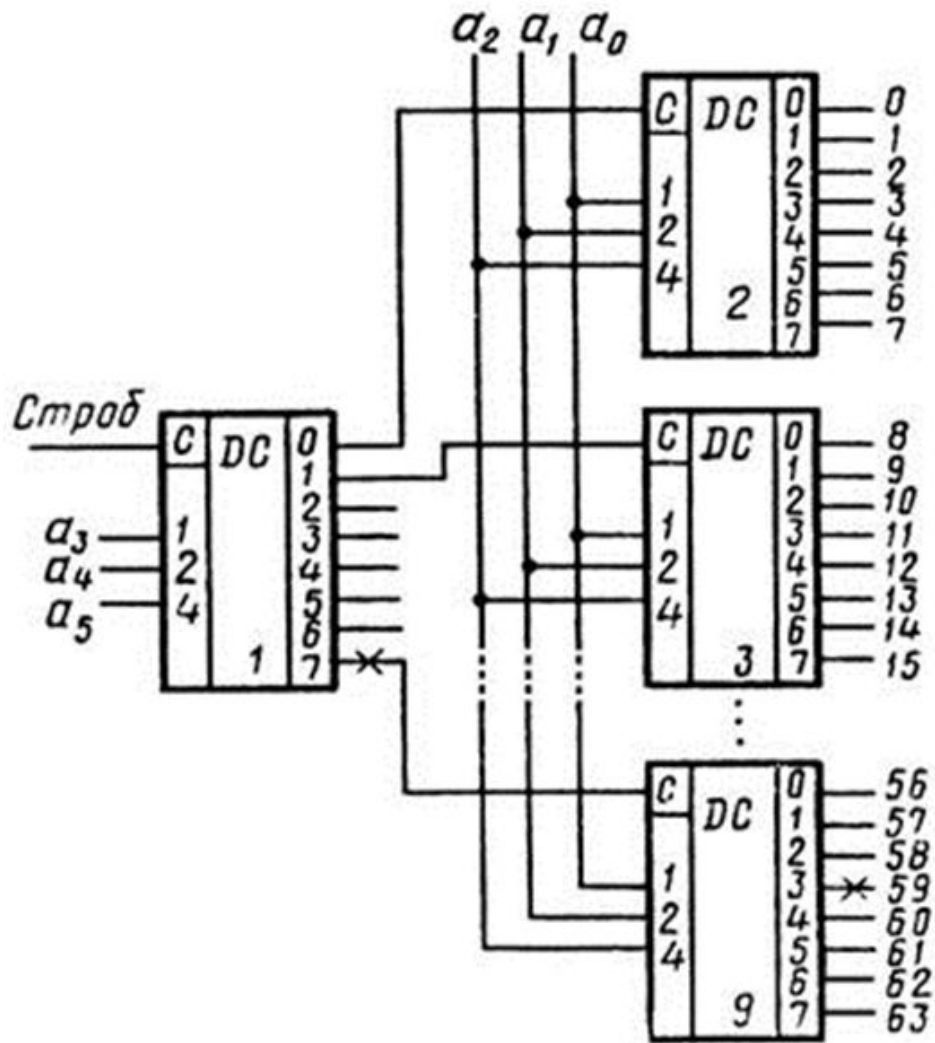


Рисунок 3.10 – Схема наращивания размерности дешифраторов

# Основы элементной базы цифровых автоматов

## 3.2.2 Шифраторы

Таблица 3.5 – Таблица функционирования шифратора

Возбужденный вход	Выход $a_3 a_2 a_1 a_0$
$F_0$	0000
$F_1$	0001
$F_2$	0010
$F_3$	0011
$F_4$	0100
$F_5$	0101
$F_6$	0110
$F_7$	0111
$F_8$	1000
$F_9$	1001

Шифраторы – устройства, осуществляющие преобразование десятичных чисел в двоичный код. Шифратор содержит  $m$  входов, последовательно пронумерованных десятичными числами  $(0, 1, 2, \dots, m-1)$  и  $n$  выходов. Подача сигнала на один из входов приводит к появлению на выходах  $n$ -разрядного двоичного кода, соответствующего номеру возведенного входа (таблица 3.5). Шифраторы широко используются в разнообразных устройствах ввода информации в цифровые системы.

Двоичные шифраторы преобразуют код «1 из  $N$ » в двоичный код, т. е. выполняют микрооперацию, обратную микрооперации дешифраторов. При возбуждении одной из входных цепей шифратора на его выходах формируется слово, отображающее номер возбужденной цепи. Полный двоичный шифратор имеет  $2^n$  входов и  $n$  выходов. Одно из основных применений шифратора – ввод данных с клавиатуры, при котором нажатие клавиши с десятичной цифрой должно приводить к передаче в устройство двоичного кода данной цифры (тетрады двоично-десятичного кода). В этом случае нужен неполный шифратор «10x4» (рисунок 3.11, а), на примере которого рассмотрим принципы построения шифраторов.

Из таблицы 3.5 следует, что

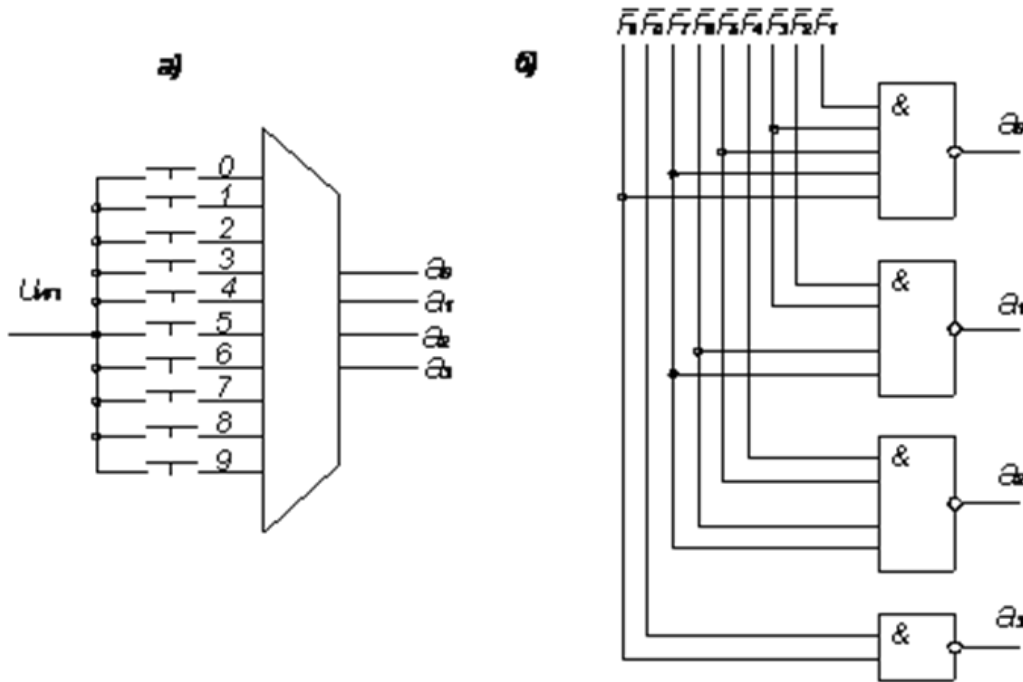
$$a_0 = F_1 + F_3 + F_5 + F_7 + F_9;$$

$$a_1 = F_2 + F_3 + F_6 + F_7;$$

$$a_2 = F_4 + F_5 + F_6 + F_7;$$

$$a_3 = F_8 + F_9.$$

## Основы элементной базы цифровых автоматов



На основании этих выражений строится шифратор, показанный на рисунке 3.11, б.

Рисунок 3.11 – Схема ввода данных с клавиатуры (а) и дешифратор «10x4» (б)

Для реализации шифратора на элементах ТТЛ, часто применяемых в периферийных устройствах, следует выразить полученные соотношения через операцию И–НЕ. Тогда

$$a_0 = \overline{F_1 * F_3 * F_5 * F_7 * F_9};$$

$$a_1 = \overline{F_2 * F_3 * F_6 * F_7};$$

$$a_2 = \overline{F_4 * F_5 * F_6 * F_7};$$

$$a_3 = \overline{F_8 * F_9}.$$



## Основы элементной базы цифровых автоматов

Для указания старшей единицы (поиск разряда слова) используют шифраторы приоритета (рисунок 3.12). На вход этого шифратора поступает восьмиразрядное слово (инвертированное), на его выходе в виде трёхразрядного кода вырабатывается номер разряда содержащего старшую единицу.

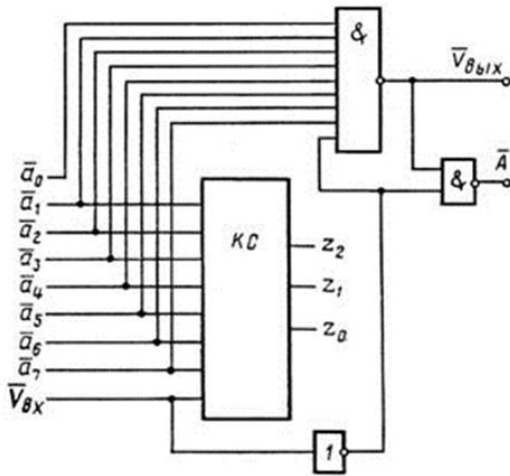


Рисунок 3.12 – Упрощенная схема шифратора приоритета

При нулевом значении  $\bar{V}_{вх}$  разрешается работа шифратора и снимается блокировка с элементов И-НЕ. Сигнал  $\bar{V}_{вых}$  играет роль стробующего для следующей за данной группой разрядов группы младших разрядов данного слова. Только при условии нулевых значений всех разрядов данного слова и  $\bar{V}_{вх}$  на выходе  $\bar{V}_{вых}$  появится нуль, разрешающий работу младшей группы, что и требуется по логике работы многоразрядного шифратора. Сигнал  $A=1$  оповещает о наличии хотя бы одной единицы в анализируемом слове.

Функции  $Z_2, Z_1, Z_0$  реализующая с помощью комбинационной схемы КС, структура которой определяется полученными выше зависимостями. Кроме того, схема имеет дополнительный вход  $\bar{V}_{вх}$  и два дополнительных выхода  $\bar{V}_{вых}$  и  $\bar{A}$ , обеспечивающие возможность каскадирования шифраторов приоритета для анализа положения старшей единицы в слове, разрядность которого превышает 8.

Вход  $\bar{V}_{вх}$  осуществляет стробирование шифратора. При единичном значении  $\bar{V}_{вх}$  шифратор блокирован и можно производить смену входного слова без появления на выходе ложных сигналов. В это же время сигналы  $\bar{V}_{вх}$  и  $\bar{A}$  также становятся единичными (в данном случае это пассивные сигналы).

## Основы элементной базы цифровых автоматов

Указатели старшей единицы с выходом в коде «один из  $N$ » можно получить путем дешифрации выхода шифратора приоритетов. Если позволяют требования к быстродействию, то эту же задачу можно решить проще — путем последовательного опроса разрядов в схеме на рисунке 3.13, где единичный сигнал опроса, подаваемый на схему со стороны старшего разряда, может распространяться от разряда к разряду (вправо) только до первого разряда, содержащего единицу.

Если в данном разряде нуль, то сигнал опроса пропускается дальше, а на выходе данного разряда остается нулевой сигнал. Если же в данном разряде единица, то на его выходе конъюнктор оказывается блокированным нулевым значением инвертированной переменной и дальнейшее распространение сигнала опроса прекращается. Одновременно на выходе данного разряда возникает единичный сигнал.

Максимальное время, необходимое для опроса всех разрядов слова, пропорционально числу разрядов слова и равно  $nt_k$ , где  $t_k$  — задержка конъюнктора.

Эта же схема может вместе с двоичным шифратором, подключенным к ее выходу, давать номер разряда, содержащего старшую единицу.

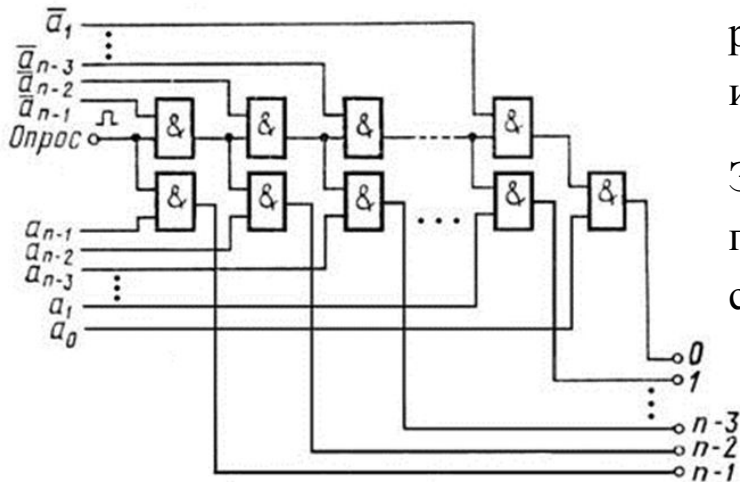


Рисунок 3.13 – Схема указателя старшей единицы с последовательным опросом разрядов

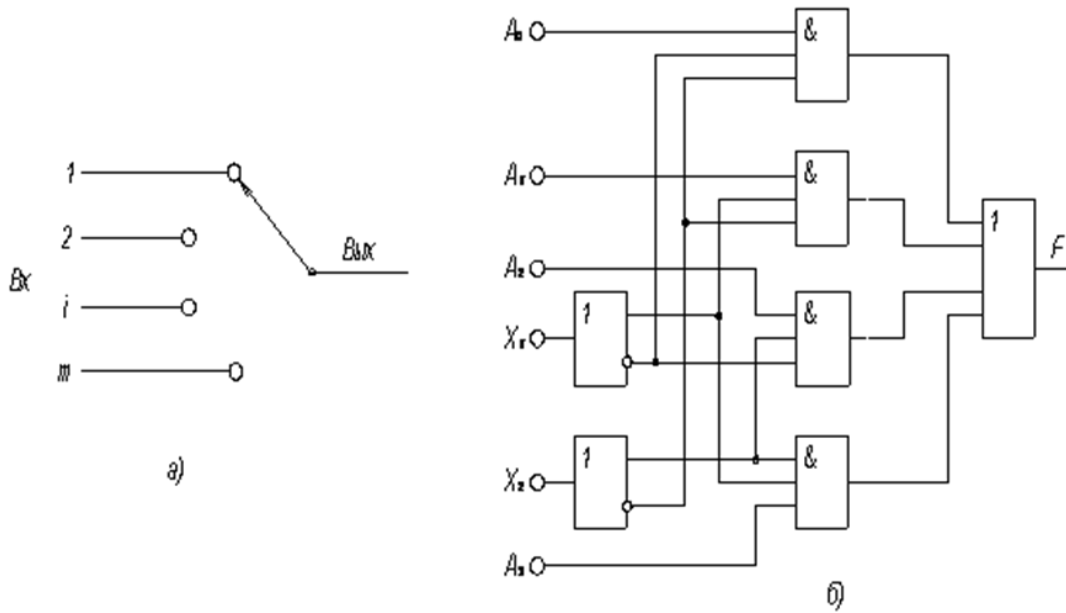
### 3.3 Мультиплексоры – демультиплексоры

#### 3.3.1 Мультиплексоры

Мультиплексор – коммутатор логических сигналов, обеспечивающий передачу информации, поступающей по нескольким входным линиям связи, на одну выходную линию (рисунок 3.14, *а*). Выбор входной линии  $A_i$  осуществляется в соответствии с поступающим адресным кодом. При наличии  $m$  адресных входов можно реализовать  $M=2^m$  комбинаций адресных сигналов, каждая из которых обеспечивает выбор одной из  $M$  входных линий. Мультиплексор состоит из дешифратора адреса входной линии, схем И и схемы объединения ИЛИ. Функциональная схема мультиплексора приведена на рисунке 3.14, *б*. Двоичный код, воздействующий на адресные входы, откроет одну из схем И, которая соединит с выходом соответствующую входную линию. При этом информация на выходе определяется состоянием выбранного входного канала и не зависит от состояния других каналов.

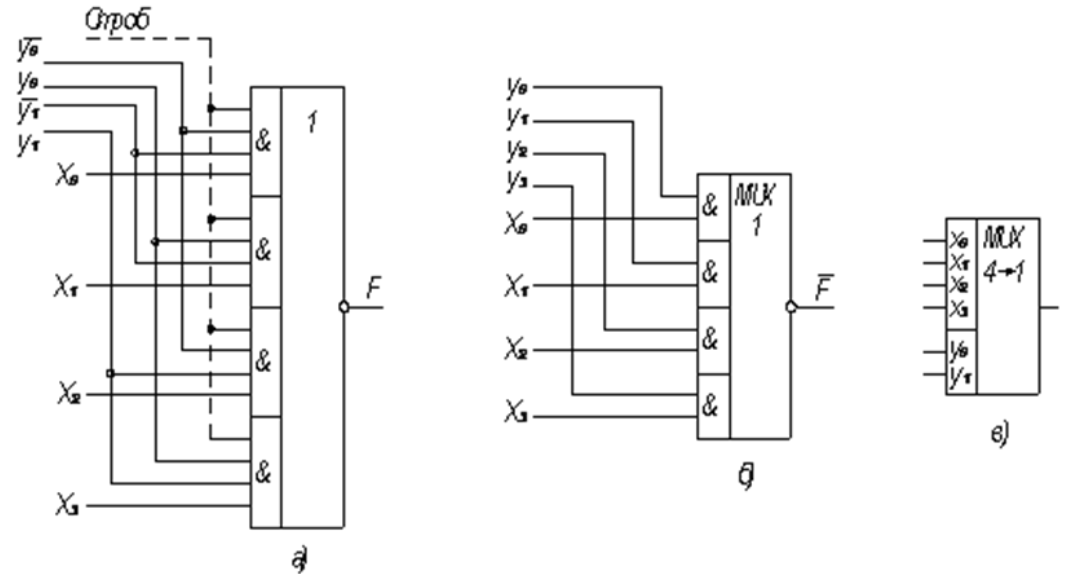
Мультиплексор «4>1», выполненный на элементах И-ИЛИ-НЕ, показан на рисунке 3.15, *а*. В условных графических обозначениях функция мультиплексирования именуется MUX (от слова *multiplexor*). Пример обозначения для мультиплексора «4>1» показан на рисунке 3.15, *в*. Управление мультиплексором может производиться не только с помощью двоичного кода, но и кодом «1 из N». В этом случае число управляющих входов увеличивается становится равным числу информационных входов (рисунок 3.15, *б*). Такой режим мультиплексора используется, в частности, в межразрядных цепях реверсивных счетчиков и регистров.

# Основы элементной базы цифровых автоматов



3.14 – Принцип работы мультиплексора (а), принцип реализации (б)

Рисунок 3.15 – Схема мультиплексора «4→1» на элементах И-ИЛИ-НЕ (а), схема управления кодом «1 из N» (б), пример обозначения (в)



## Основы элементной базы цифровых автоматов

В сериях микросхем встречаются мультиплексоры «4>1», «8>1», «16>1» Мультиплексоры на большее число входов, как правило, приходится строить из мультиплексоров меньшей размерности. Если необходим мультиплексор «N>1», а имеются ИМС с числом входов  $N_1$ , то потребуются  $L$  ИМС, где  $L = \lceil N/N_1 \rceil$ , которые совместно обеспечат нужное число входов. Для каждой ИМС разрядность управляющего кода составит  $n_1 = \log_2 N_1$ , тогда как разрядность управляющего кода всей схемы в целом равна  $n = \lceil \log_2 N \rceil$ . Число разрядов, равное разности  $n - n_1$ , используется для организации поочередной передачи выходов отдельных ИМС и общий выходной канал. При этом имеет значение тип выходного каскада ИМС. Если это каскады обычного типа, то потребуется дополнительно объединяющий мультиплексор на выходе схемы (рисунок 3.16).

Функционирование такой схемы покажем на конкретных примерах. Пусть, например, управляющий код равен 10101. Значит, на выходах мультиплексоров первого яруса будут сигналы с их пятых информационных входов ( $y_2 y_1 y_0 = 101$ ). На выходной мультиплексор подается управляющий код 10, и на выход схемы попадает сигнал  $x_2$  выходного мультиплексора, т. е. пятый выход третьего мультиплексора, номер которого равен 21, что и соответствует двоичному числу 10101. Если ИМС имеют выходы с тремя состояниями, то можно непосредственно объединять эти выходы, а поочередное подключение ИМС к выходной цепи осуществить с помощью дешифратора, управляющего стробирующими входами ИМС (рисунок 3.17). Недостаток такого способа наращивания схем – суммирование емкостей в выходном узле, что в ряде случаев (например, для схем на МОП-транзисторах) может существенно снизить быстродействие мультиплексора.

Мультиплексоры можно использовать для синтеза логических функций от нескольких переменных ( $x_1, x_2, \dots, x_n$ ). Если число адресных входов мультиплексора  $m_{\text{адр}}$ , то из общего числа  $n$  переменных функции  $m_{\text{адр}}$  можно подать на адресные входы. Тогда на информационные входы мультиплексора через дополнительную логическую схему подаются  $n - m_{\text{адр}}$  переменных. Структуру такой логической схемы можно определить табличным методом или с помощью диаграмм Вейча

# Основы элементной базы цифровых автоматов

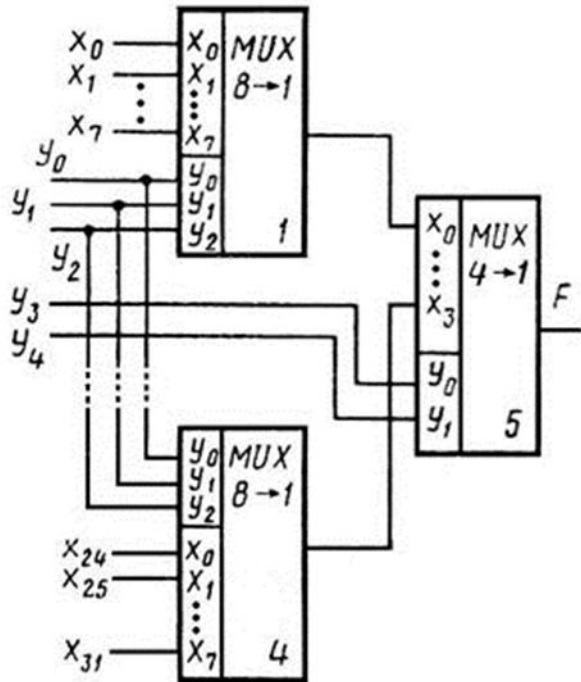


Рисунок 3.16 – Нарращивание размерности мультиплексора

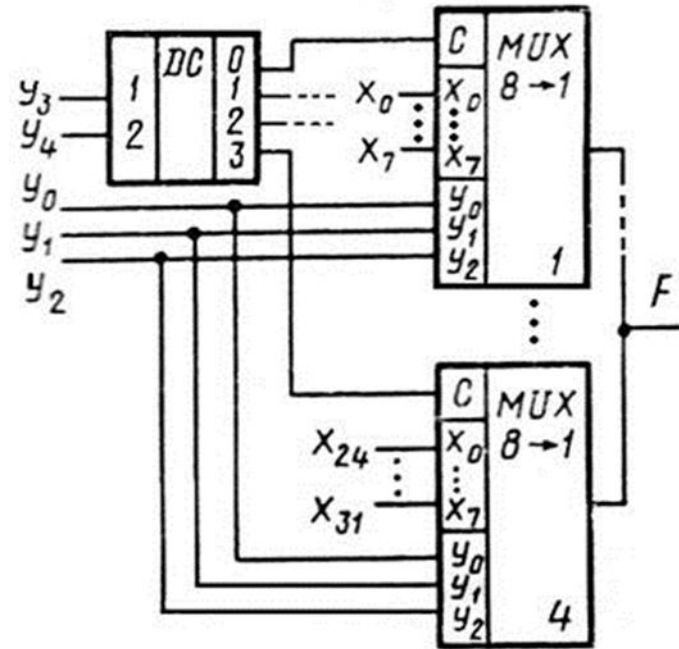


Рисунок 3.17 – Нарращивание размерности мультиплексора, имеющего выходные каскады с тремя состояниями

## Основы элементной базы цифровых автоматов

### 3.3.2 Демультимплексоры

Демультимплексор – коммутатор логических сигналов, обеспечивающий передачу информации, поступающей по одному входу, и в соответствии с адресом направляющий в одну из выходных линий. При передачах данных по общему каналу с разделением во времени нужны не только мультиплексоры, но и устройства обратного назначения, распределяющие данные из одного канала между несколькими приемниками информации. Эта задача реализуется демультимплексорами, функционирование которых можно пояснить с помощью рисунок 3.18, а. Демультимплексор имеет один информационный вход  $n$  адресующих (управляющих) входов и  $2^n$  выходов. Построение демультимплексора «1>4» на элементах И показано на рисунке 3.18, б. Работа демультимплексора описывается логическими выражениями:

$$F_i = x m_i; \quad i = \overline{0, 2^n - 1} \quad \text{где } m_i \text{ – минтермы } n \text{ адресующих переменных.}$$

В условных изображениях демультимплексор обозначается буквами ДМХ (рисунок 3.18, в). Применительно к мультиплексорам и демультимплексорам пользуются также термином «селекторы данных». Функционирование демультимплексора можно заменить таблицей. Например работу демультимплексора с четырьмя выходами (таблица 3.6). При большом числе входов, при необходимости, может быть построено демультимплексорное дерево (рисунок 3.19). Дешифраторы со стробированием (см. рисунок 3.5, а) можно использовать в качестве демультимплексоров. При этом стробирующий вход выполняет функции информационного входа.

# Основы элементной базы цифровых автоматов

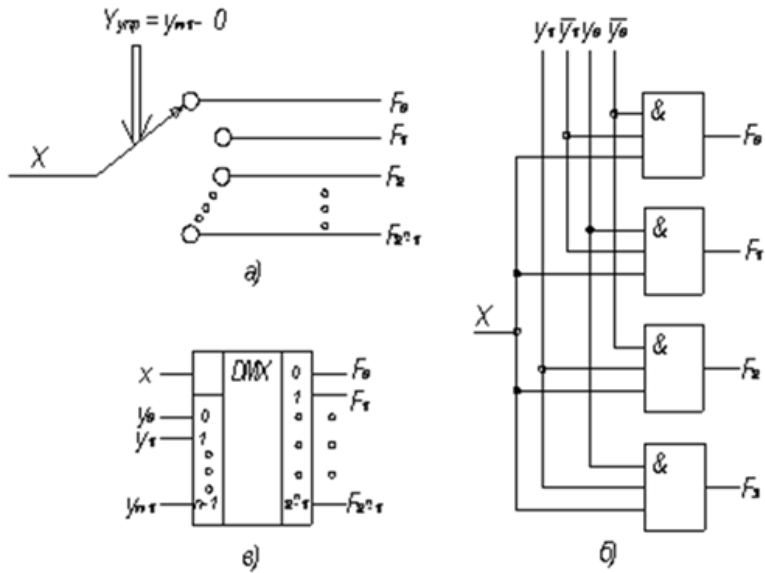


Рисунок 3.18 – Принцип работы (а), схемная реализация (б) и условное обозначение для мультиплектора (в)

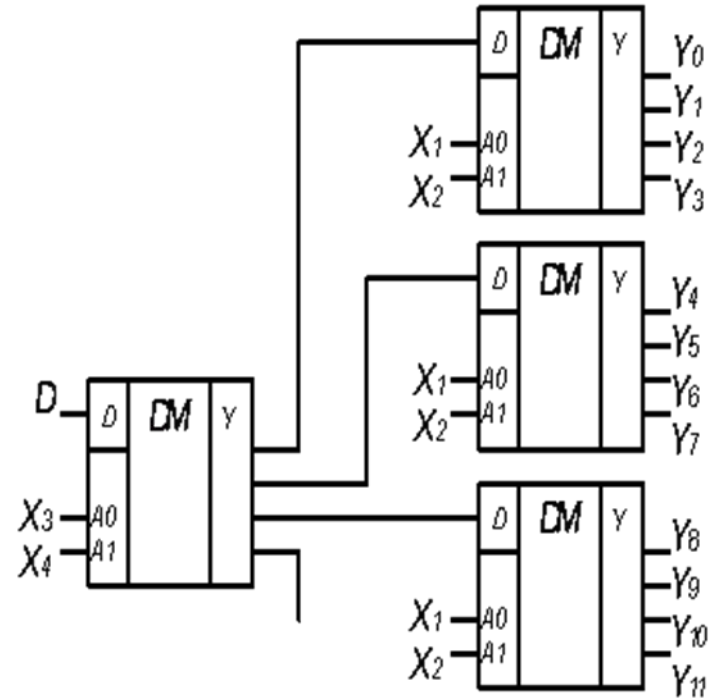


Рисунок 3.19 – Мультиплексорное дерево

Таблица 3.6 –  
Функционирование  
демультиплектора

Адресные входы		Выходы			
$A_1$	$A_0$	0	1	2	3
0	0	Д	0	0	0
0	1	0	Д	0	0
1	0	0	0	Д	0
1	1	0	0	0	Д