

Литература

1. Авдеев, В. А. Периферийные устройства: интерфейсы, схемотехника, программирование. - М.: ДМК Пресс, 2009.
2. Лехин С.Н. Схемотехника ЭВМ. – СПб.: БХВ-Петербург, 2010.
3. Чижма С.Н. Основы схемотехники. – Омск: Апельсин, 2008.
4. Титце У., Шенк К. Полупроводниковая схемотехника. – М: ДМК Пресс, 2008
5. Новиков Ю.В. Введение в цифровую схемотехнику – М.: БИНОМ.ЛЗ, ИНТУИТ.РУ, 2007.

6. Бойко В.И., Гуржий А.Н., Жуйков В.Я. и др.
Схемотехника электронных систем. Аналоговые и импульсные устройства. – С-Пб.: БХВ-Петербург, 2004.
7. Бойко В.И., Гуржий А.Н., Жуйков В.Я. и др.
Схемотехника электронных систем. Цифровые устройства. – С-Пб.: БХВ-Петербург, 2004.
8. Хоровиц П., Хилл У. Искусство схемотехники. – М.: Мир, 2003.
9. Фролкин В.Т., Попов Л.Н. Импульсные и цифровые устройства. – М.: Радио и связь, 1992.

10. Баскаков С.И. Радиотехнические цепи и сигналы. М.: Мир, 2000.
11. Гоноровский И.С. Радиотехнические цепи и сигналы. – М.: Дрофа, 2006.
12. Крылов В.В., Корсаков С.Я. Основы теории цепей для системотехников. Уч. пособие. – М.: Высшая школа, 1990.
13. Угрюмов Е.П. Цифровая схемотехника. – СПб: ВНУ - Санкт-Петербург, 2000.
14. Корис Р., Шмидт-Вальтер Х. Справочник инженера-схемотехника. – М.: Техносфера, 2008.

В курсе **электротехники** изучаются свойства элементов, законы и методы расчета электрических линейных цепей, рассматриваются особенности трехфазных и нелинейных цепей, процессы в цепи при воздействии на нее изменяющихся напряжений, а также переходные процессы.

В курсе **электроники** даются сведения по устройству и применению полупроводниковых элементов в различных приборах.

Схемотехника — научно-техническое направление, охватывающее проблемы проектирования и исследования схем электронных устройств радиотехники и связи, вычислительной техники, автоматики и др. областей техники.

ЦИФРОВЫЕ УСТРОЙСТВА

Элементная база вычислительной техники стремительно развивается благодаря успехам полупроводниковой технологии. Есть микросхемы, содержащие более миллиарда компонентов типа транзистора на одном кристалле. Общие принципы построения цифровых схем остаются неизменными и основаны на булевой алгебре и применении стандартных функциональных узлов.

Элементом цифрового устройства называют наименьшую функциональную часть, на которые можно разделить цифровое устройство при его проектировании и анализе. Набор элементов, объединенных общей конструкцией, технологией, способом представления информации, организацией связей образует **систему элементов**. Система должна быть функционально полной, чтобы реализовать любые логические функции путем суперпозиции простейших функций, предусмотренных системой.

ЦИФРОВЫЕ УСТРОЙСТВА

Элементы малой и средней интеграции содержат в едином корпусе наборы однотипных элементов с отдельными входами и выходами. Элементы высокой степени интеграции объединяют в одном корпусе наборы различных элементов, объединенных в законченные функциональные узлы.

По функциональному назначению элементы делят на **логические, запоминающие и специальные.**

Логические элементы выполняют логические функции и относятся к классу комбинационных схем, в которых выходной сигнал в некоторый момент времени зависит только от входных сигналов, действующих в тот же момент времени. **Запоминающие элементы** служат для хранения цифровой информации и относятся обычно к классу последовательностных схем, в которых выходные состояния определяются не только состояниями входов в данный момент времени, но и внутренним состоянием самой схемы. **Специальные элементы** обеспечивают сопряженную работу узлов и подсистем ЭВМ, они могут быть цифровыми, аналоговыми и цифроаналоговыми.

ЦИФРОВЫЕ УСТРОЙСТВА

Цифровые устройства разделяются на два класса: **комбинационные** устройства, которые не обладают памятью, и **последовательностные** устройства, обладающие памятью. Действия комбинационных устройств однозначно определяются только текущими значениями входных сигналов. На результат работы последовательностных устройств влияние оказывают значения входных сигналов как в настоящий, так и в предыдущий моменты времени.

В цифровых устройствах обычно используются микроэлектронные компоненты, на входах и выходах которых сигналы могут принимать одно из двух возможных значений (тока или напряжения). Таким образом, цифровые устройства реализуют логические функции, которые задают связь между входными и выходными логическими сигналами (а в последовательностных устройствах – еще и внутренними состояниями).

Теоретической основой цифровых устройств, описываемых логическими функциями, является алгебра логики (булева алгебра).

ЦИФРОВЫЕ УСТРОЙСТВА

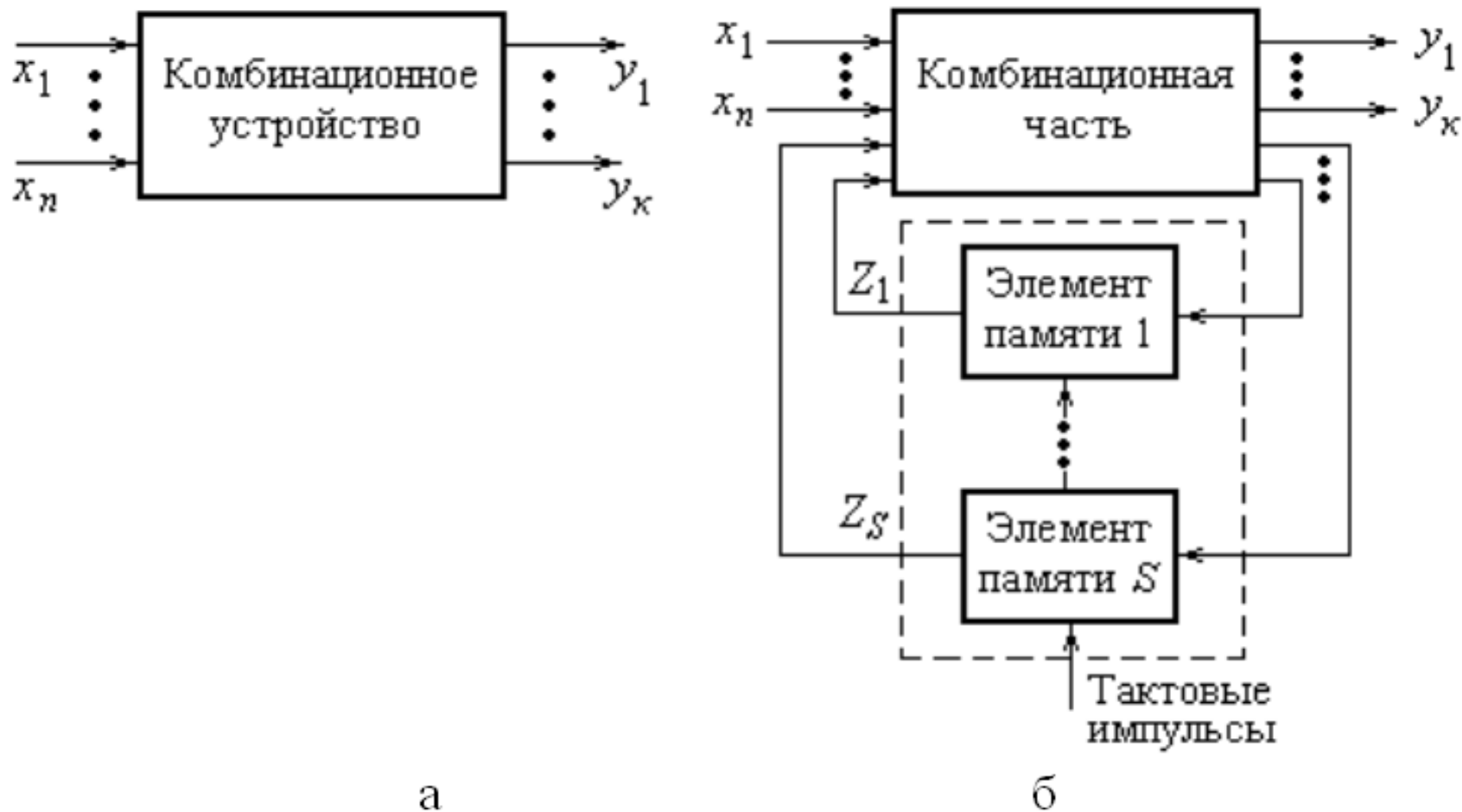


Рис. Структура а - комбинационного и б - последовательного цифровых устройств

Логические вентили

Логическая функция может быть задана с помощью таблицы истинности или структурной формулы.








В таблице истинности для каждого возможного набора значений входных переменных указывается набор значений выходных переменных. В таблице истинности количество столбцов равно сумме числа входных N_{in} и выходных сигналов N_{out} . Количество строк (не считая заголовочной) равно $2^{N_{in}}$. Пример таблицы истинности для логической функции трех переменных $y = f(x_0, x_1, x_2)$ показан в табл. 1.

Таблица 1. Пример таблицы истинности для логической функции трех переменных

x_2	x_1	x_0	y
0	0	0	$f(0,0,0)$
0	0	1	$f(0,0,1)$
0	1	0	$f(0,1,0)$
0	1	1	$f(0,1,1)$
1	0	0	$f(1,0,0)$
1	0	1	$f(1,0,1)$
1	1	0	$f(1,1,0)$
1	1	1	$f(1,1,1)$

При задании логической функции с помощью формулы используются различные логические операции, основными из которых являются операции логического умножения (И), сложения (ИЛИ) и отрицания (НЕ) (см. табл. 2). С помощью этих операций можно описать любую логическую функцию. Для удобства иногда используют дополнительные логические операции, такие как исключающее ИЛИ, исключающее И, а также операции И-НЕ, ИЛИ-НЕ.

Таблица 2. Виды элементарных логических операций

Название	Вид	Функция	Таблица истинности		
AND И		$A \cdot B$	x_1	x_0	y
			0	0	0
			0	1	0
			1	0	0
			1	1	1
OR ИЛИ		$A + B$	x_1	x_0	y
			0	0	0
			0	1	1
			1	0	1
			1	1	1
NOT Инвертор		\overline{A}	x	y	
			0	1	
			1	0	
NAND И-НЕ		$\overline{A \cdot B}$	x_1	x_0	y
			0	0	1
			0	1	1
			1	0	1
			1	1	0
NOR ИЛИ-НЕ		$\overline{A + B}$	x_1	x_0	y
			0	0	1
			0	1	0
			1	0	0
			1	1	0
XOR Исключающее ИЛИ (либо)		$A \oplus B$	x_1	x_0	y
			0	0	0
			0	1	1
			1	0	1
			1	1	0
XNOR Либо-не		$\overline{A \oplus B}$ $A \odot B$	x_1	x_0	y
			0	0	1
			0	1	0
			1	0	0
			1	1	1

Логические вентили

При задании логической функции с помощью формулы используются различные логические операции, основными из которых являются операции логического умножения (И), сложения (ИЛИ) и отрицания (НЕ) (см. табл. 2). С помощью этих операций можно описать любую логическую функцию. Для удобства иногда используют дополнительные логические операции, такие как исключающее ИЛИ, исключающее И, а также операции И-НЕ, ИЛИ-НЕ.








Реализация этих операций выполняется электронной схемой на основе транзисторов, которая называется логическим вентилями. **Логический вентиль** – это базовый элемент цифровой схемы, выполняющий элементарную логическую операцию, преобразуя множество входных логических сигналов в выходной логический сигнал. Логика работы вентиля основана на логических (битовых) операциях над входными двоичными сигналами. Например, при реализации вентилях в ТТЛ логике используются биполярные транзисторы, у которых один, а несколько эмиттеров.

В настоящее время в современных микросхемах и цифровых устройствах доминируют логические вентили на основе полевых транзисторов, однако в прошлом для создания вентилях использовались и другие устройства, например, электромагнитные реле, гидравлические, а также механические устройства. В качестве альтернативных логических вентилях исследуются вентили на основе квантовых устройств, нейронов и молекул.

При разработке цифрового устройства логические вентили соединяют между собой, при этом выход используемого вентиля должен быть подключён к одному или к нескольким входам других вентилях.

Виды элементарных логических операций, их схематические обозначения, аналитическая запись и таблицы истинности представлены в табл. 2.

Таблица 2. Виды элементарных логических операций

Название	Вид	Функция	Таблица истинности		
AND И		$A \cdot B$	x_1	x_0	y
			0	0	0
			0	1	0
			1	0	0
			1	1	1
OR ИЛИ		$A + B$	x_1	x_0	y
			0	0	0
			0	1	1
			1	0	1
			1	1	1
NOT Инвертор		\overline{A}	x	y	
			0	1	
			1	0	
NAND И-НЕ		$\overline{A \cdot B}$	x_1	x_0	y
			0	0	1
			0	1	1
			1	0	1
			1	1	0
NOR ИЛИ-НЕ		$\overline{A + B}$	x_1	x_0	y
			0	0	1
			0	1	0
			1	0	0
			1	1	0
XOR Исключающее ИЛИ (либо)		$A \oplus B$	x_1	x_0	y
			0	0	0
			0	1	1
			1	0	1
			1	1	0
XNOR Либо-не		$\overline{A \oplus B}$ $A \odot B$	x_1	x_0	y
			0	0	1
			0	1	0
			1	0	0
			1	1	1

ЦИФРОВЫЕ УСТРОЙСТВА

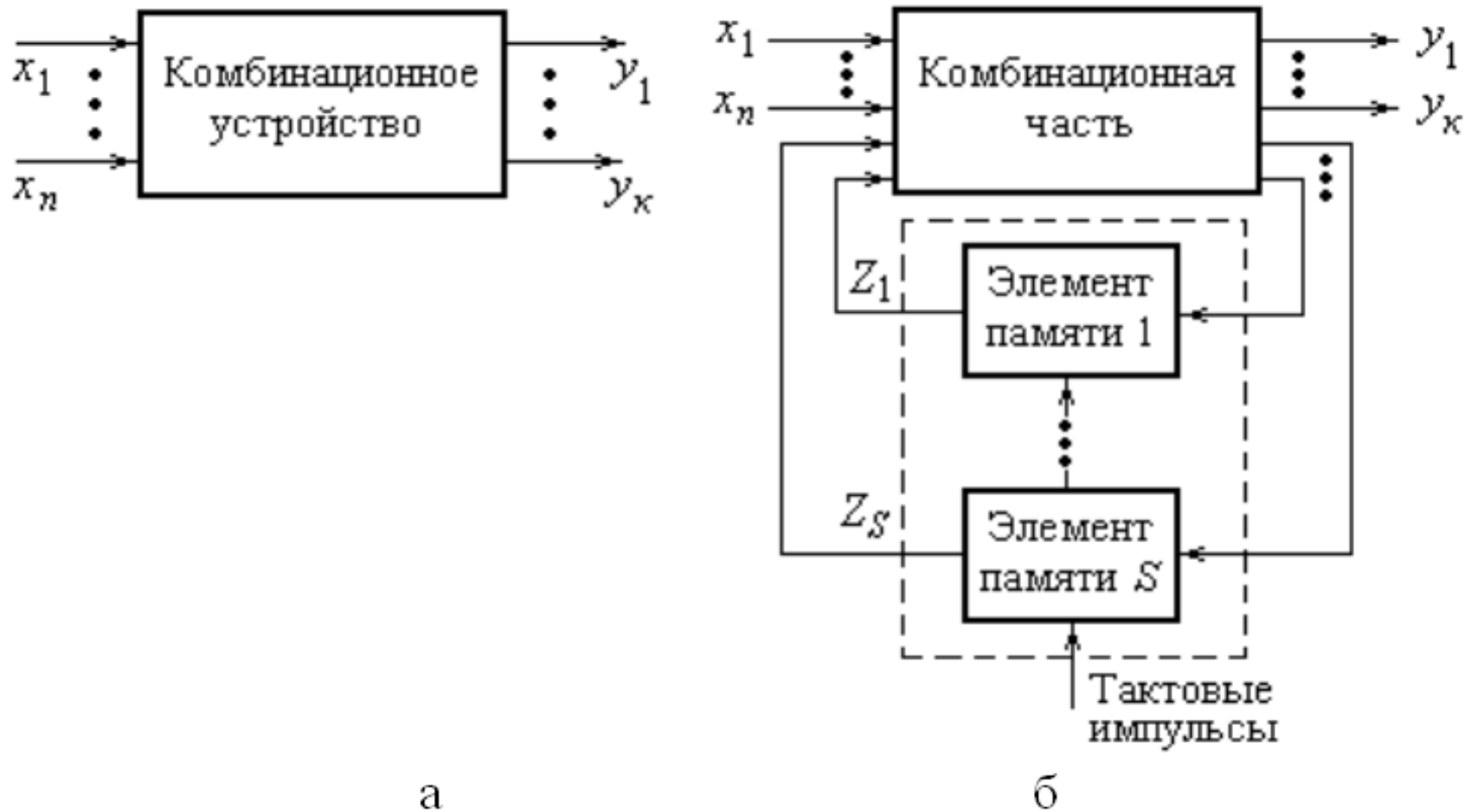


Рис. Структура а - комбинационного и б - последовательного цифровых устройств

КОМБИНАЦИОННЫЕ ЦИФРОВЫЕ УСТРОЙСТВА

Комбинационные цифровые устройства предназначены для преобразования цифровых сигналов на основе комбинационных логических схем, и, следовательно, выходные состояния таких устройств не зависят от предыстории, а однозначно определяются входными сигналами в рассматриваемые моменты времени. Другими словами комбинационные устройства не содержат элементы памяти. К основным типам комбинационных устройств относятся сумматоры, дешифраторы и шифраторы, мультиплексоры и демультиплексоры, преобразователи кодов, схемы сравнения, пороговые и мажоритарные элементы и др.

Сумматоры

Различают полные сумматоры и полусумматоры. *Полный сумматор* предназначен для сложения трех одnorазрядных чисел A , B и C , где в качестве числа C выступает перенос из предыдущего разряда P . В результате выполнения операции сложения в каждом разряде, кроме суммы, может возникнуть перенос в старший разряд P .

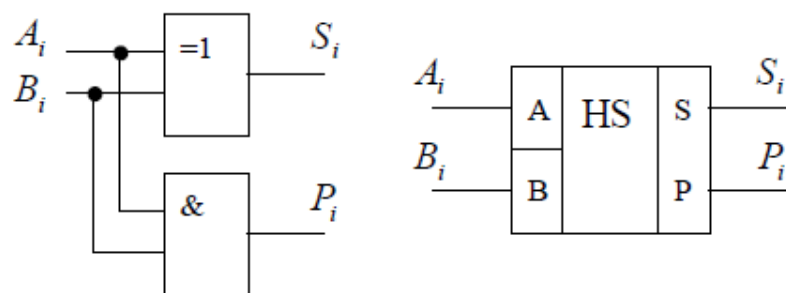
Полусумматор при выполнении операции сложения не учитывает переноса из предыдущего разряда.

Таблица истинности полусумматора

A_i	B_i	S_i	P_i
0	0	0	0
1	0	1	0
0	1	1	0
1	1	0	1

$$S_i = A_i \bar{B}_i + \bar{A}_i B_i = A_i \oplus B_i;$$

$$P_i = A_i B_i.$$



Полусумматор

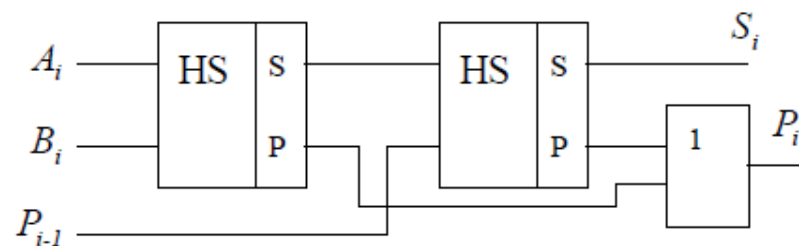


Схема полного сумматора

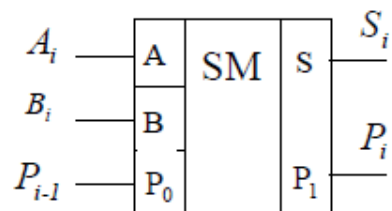


Таблица истинности полного сумматора

№	A_i	B_i	P_{i-1}	S_i	P_i
0	0	0	0	0	0
1	0	0	1	1	0
2	1	0	0	1	0
3	1	0	1	0	1
4	0	1	0	1	0
5	0	1	1	0	1
6	1	1	0	0	1
7	1	1	1	1	1

$$S_i = \bar{A}_i \bar{B}_i P_{i-1} + A_i \bar{B}_i \bar{P}_{i-1} + \bar{A}_i B_i \bar{P}_{i-1} + A_i B_i P_{i-1} = A_i \oplus B_i \oplus P_{i-1};$$

$$P_i = A_i \bar{B}_i P_{i-1} + \bar{A}_i B_i P_{i-1} + A_i B_i \bar{P}_{i-1} + A_i B_i P_{i-1} = (A_i \oplus B_i) P_{i-1} + A_i B_i.$$

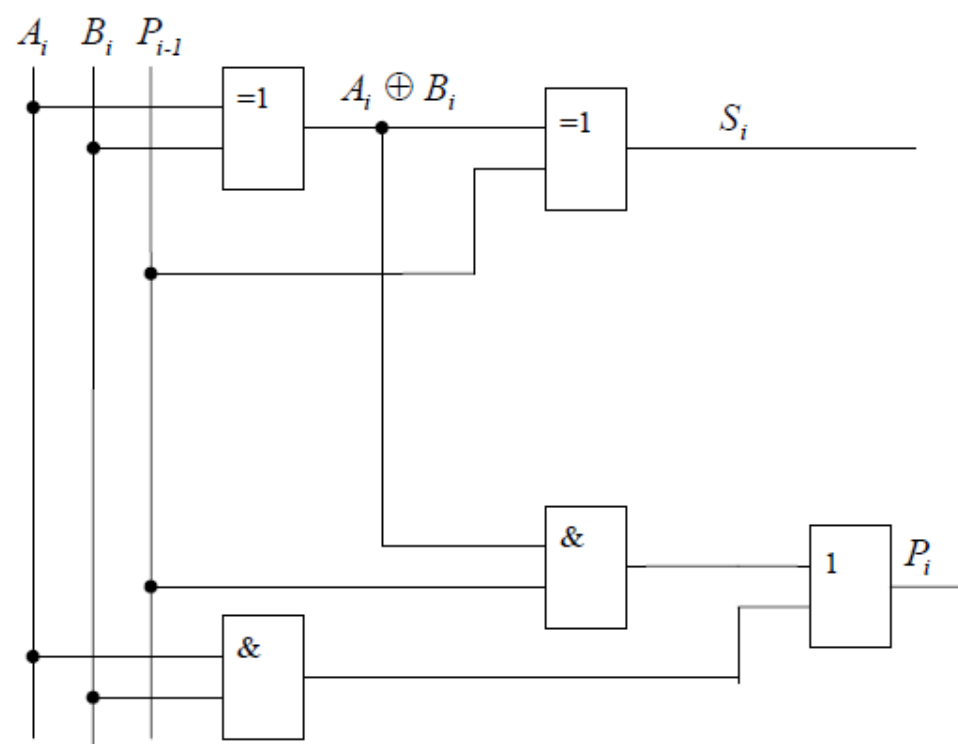
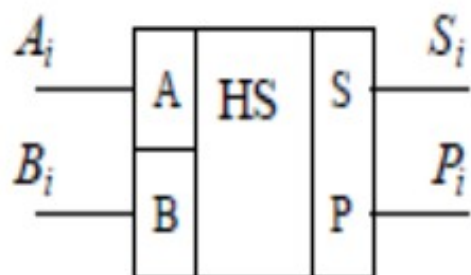
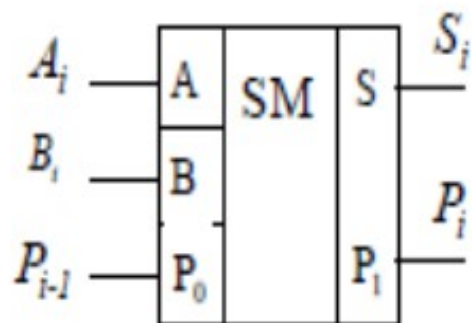


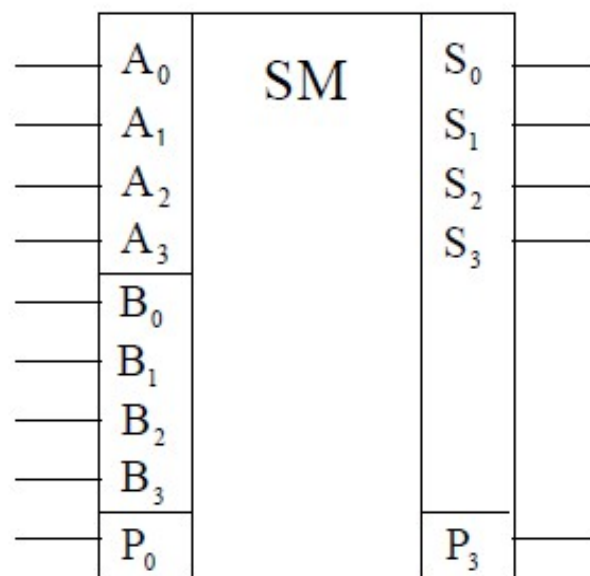
Схема полного сумматора



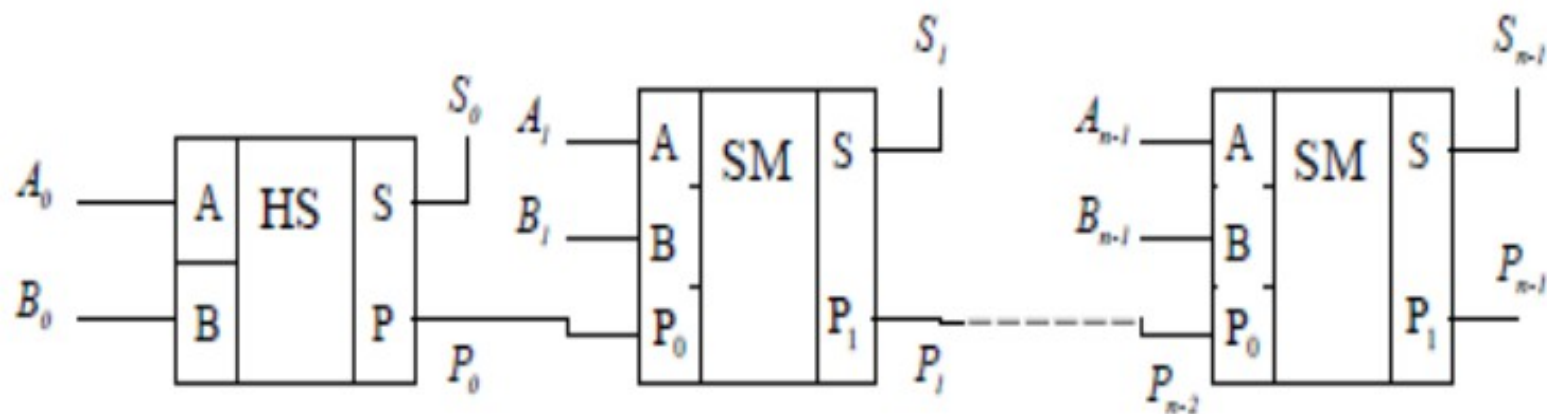
Полусумматор



полный сумматор



четырёхразрядный сумматор



Многоразрядный полный сумматор

Дешифраторы

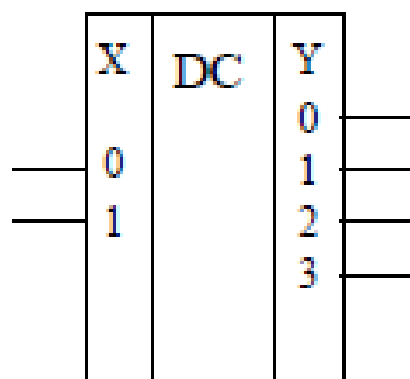
Дешифратором называется комбинационная схема, преобразующая двоичный n -разрядный код в унитарный 2^n - разрядный код, все разряды которого, за исключением одного, равны нулю, то есть дешифратор реализует на каждом своем выходе функцию конstituенты единицы n переменных. Дешифраторы бывают полные и неполные. В полном дешифраторе выполняется условие

$$N = 2^n,$$

где n - число входов;

N - число выходов.

В *неполном дешифраторе* имеется также n входов, а количество выходов $N < 2^n$.



условное обозначение дешифратора 2x4:

Таблица истинности дешифратора 2×4

X_1	X_0	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

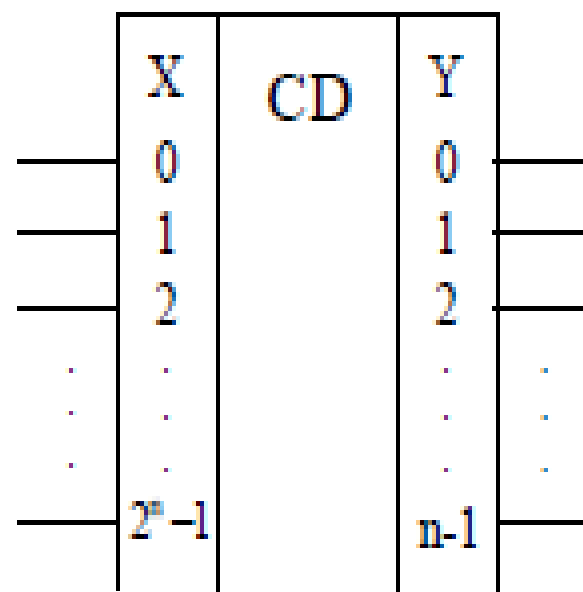
Уравнения для выходов дешифратора будут иметь вид:

$$Y_0 = \bar{X}_0 \bar{X}_1; \quad Y_1 = X_0 \bar{X}_1; \quad Y_2 = \bar{X}_0 X_1; \quad Y_3 = X_0 X_1.$$

Для расширения числа входов и выходов
можно использовать каскадное соединение дешифраторов.

Шифраторы

Шифраторы выполняют функцию обратную дешифраторам - при числе входов $N=2^n$, на которые подается унитарный код, устройство имеет n выходов, на которых формируется двоичный код. При этом только на одном из входов должна быть 1.



Условное обозначение
шифратора

Таблица истинности шифратора 4×2

Унитарный код				ДК	
X_3	X_2	X_1	X_0	Y_1	Y_0
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1

Уравнения для выходов шифратора будут иметь вид:

$$Y_0 = \bar{X}_3 \bar{X}_2 X_1 \bar{X}_0 + X_3 \bar{X}_2 \bar{X}_1 \bar{X}_0 = \bar{X}_2 \bar{X}_0 (X_1 \oplus X_3);$$

$$Y_1 = \bar{X}_3 X_2 \bar{X}_1 \bar{X}_0 + X_3 \bar{X}_2 \bar{X}_1 X_0 = \bar{X}_1 \bar{X}_0 (X_2 \oplus X_3).$$

X_1	X_0	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

$$Y_0 = \bar{X}_0 \bar{X}_1; \quad Y_1 = X_0 \bar{X}_1;$$

$$Y_2 = \bar{X}_0 X_1; \quad Y_3 = X_0 X_1.$$

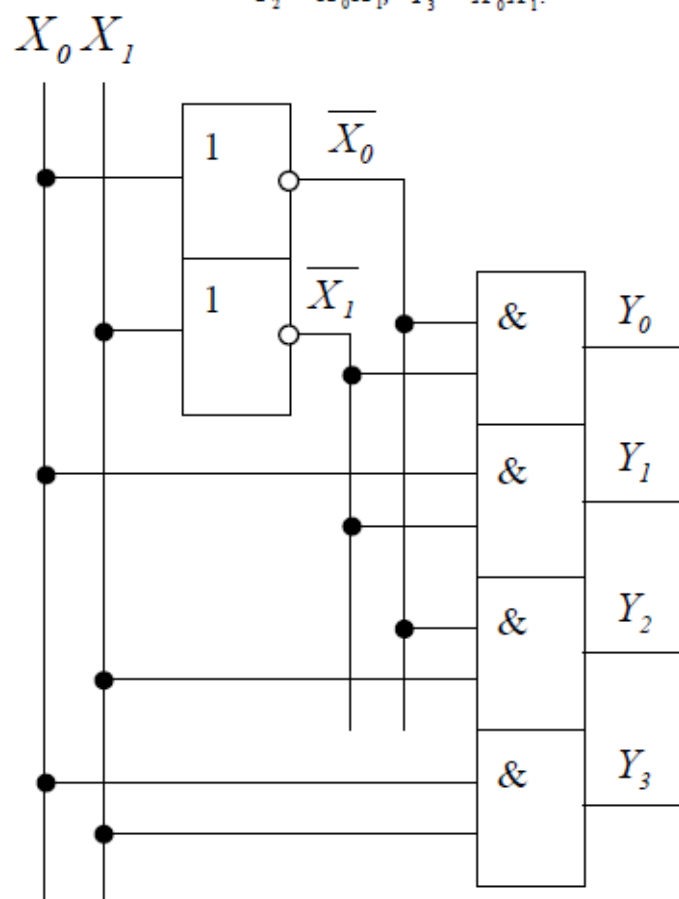


Схема дешифратора 2×4

Унитарный код				ДК	
X_3	X_2	X_1	X_0	Y_1	Y_0
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1

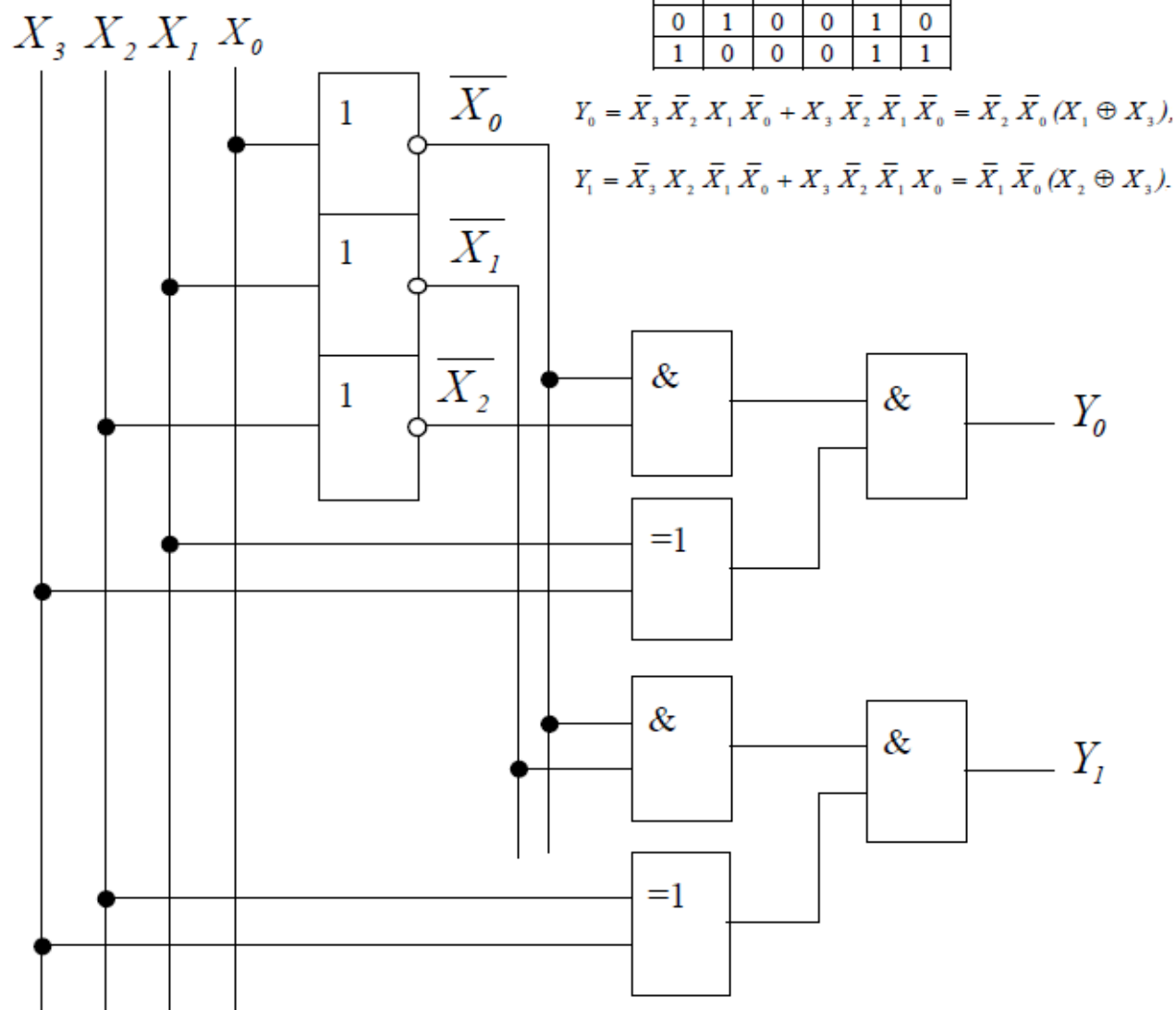


Схема шифратора 4×2

Шифраторы используются для сжатия информации при ее передаче, т.к. количество выходных линий у него меньше количества входных линий $n < N$. Для обратного преобразования на другом конце передачи используются дешифраторы. В цифровых системах это позволяет использовать для связи между различными устройствами ограниченное число линий связи. Другой широкой областью применения шифраторов является кодирование алфавитно-цифровой информации при ее вводе с кнопочных пультов или клавиатуры. Так, например, при нажатии одной из клавиш на одном из входов шифратора появляется «1», а на остальных входах должны быть нули. Этот входной унитарный код преобразуется в двоичный код, который обрабатывается цифровыми устройствами. Однако при этом возможны случаи одновременного нажатия нескольких клавиш, что может вызвать некорректную работу шифратора. Для исключения таких ситуаций каждому из входов назначают свой приоритет. Обычно, чем выше номер входа, тем выше его приоритет. Например, при ошибочной входной комбинации 0011 шифратор будет работать с комбинацией 0010. Такие шифраторы называются приоритетными шифраторами.

ШИФРАТОРЫ И ДЕШИФРАТОРЫ

Итак, основным назначением шифраторов и дешифраторов является изменение кода, в котором представлено входное число. Дешифраторы позволяют обнаружить заданные комбинации сигналов на своем входе и запустить на выполнение другие компоненты системы, а шифраторы, в свою очередь, на основании номера активного входа генерируют заданные комбинации на выходе.

Некоторые дешифраторы преобразуют N -разрядный не двоичный, а троичный или m -ичный код в m^N -ичный одноединичный (унитарный) код, где m – основание системы счисления. Логический сигнал активен на том выходе, порядковый номер которого соответствует не двоичному, а троичному или m -ичному коду. Есть также шифраторы, которые выполняют преобразование позиционного n -разрядного кода в N -ичный не двоичный, а троичный или m -ичный код.

Для корректной работы шифратора только на один его вход можно подавать сигнал (единицу), при этом на всех остальных входах сигнала быть не должно (нули). **Приоритетный шифратор** применяется в случае, когда более, чем на одном входе может присутствовать активный сигнал. В таких случаях приоритетный шифратор обрабатывает (шифрует) первый вход, на котором будет присутствовать единица.

ПРИОРИТЕТНЫЙ ШИФРАТОР

Интересно, что с помощью приоритетного шифратора можно очень просто приблизительно вычислять логарифм по основанию 2 целого числа. Логарифм по основанию 2 числа X можно представить в следующем виде (используется экспоненциальная запись с основанием 2):

$$X = 2^p M,$$

$$\log_2 X = \log_2(2^p M) = p + \log_2 M \approx p,$$

где p – порядок числа X , а M – мантисса числа x , $1 \leq M < 2$. Формулу (1) можно реализовать с помощью приоритетного шифратора. В таблице ниже для примера представлена реализация логарифма по основанию 2 чисел от 0 до 15. Красным цветом выделены единицы, по которым срабатывает приоритетный шифратор. На выходе такого шифратора генерируется результат: необходимое значение порядка числа.

ПРИОРИТЕТНЫЙ ШИФРАТОР

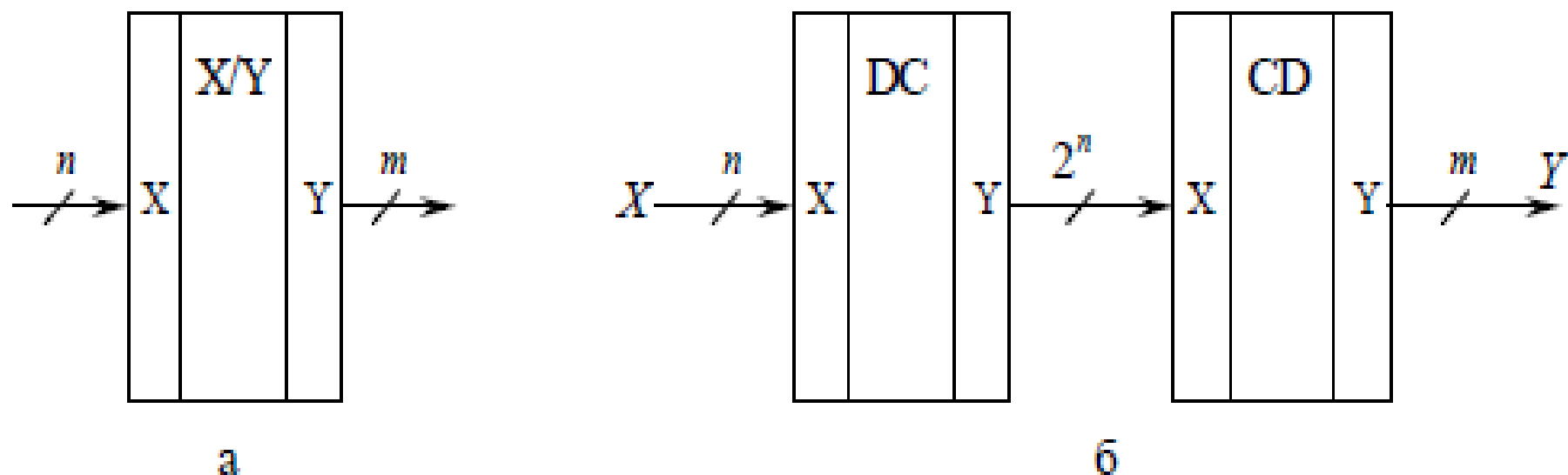
Реализация логарифма по основанию 2

на основе приоритетного шифратора для чисел от 0 до 15

Десятичный код	Прямой двоичный код				P
	x_3	x_2	x_1	x_0	
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	1
3	0	0	1	1	1
4	0	1	0	0	2
5	0	1	0	1	2
6	0	1	1	0	2
7	0	1	1	1	2
8	1	0	0	0	3
9	1	0	0	1	3
10	1	0	1	0	3
11	1	0	1	1	3
12	1	1	0	0	3
13	1	1	0	1	3
14	1	1	1	0	3
15	1	1	1	1	3

Преобразователи кодов

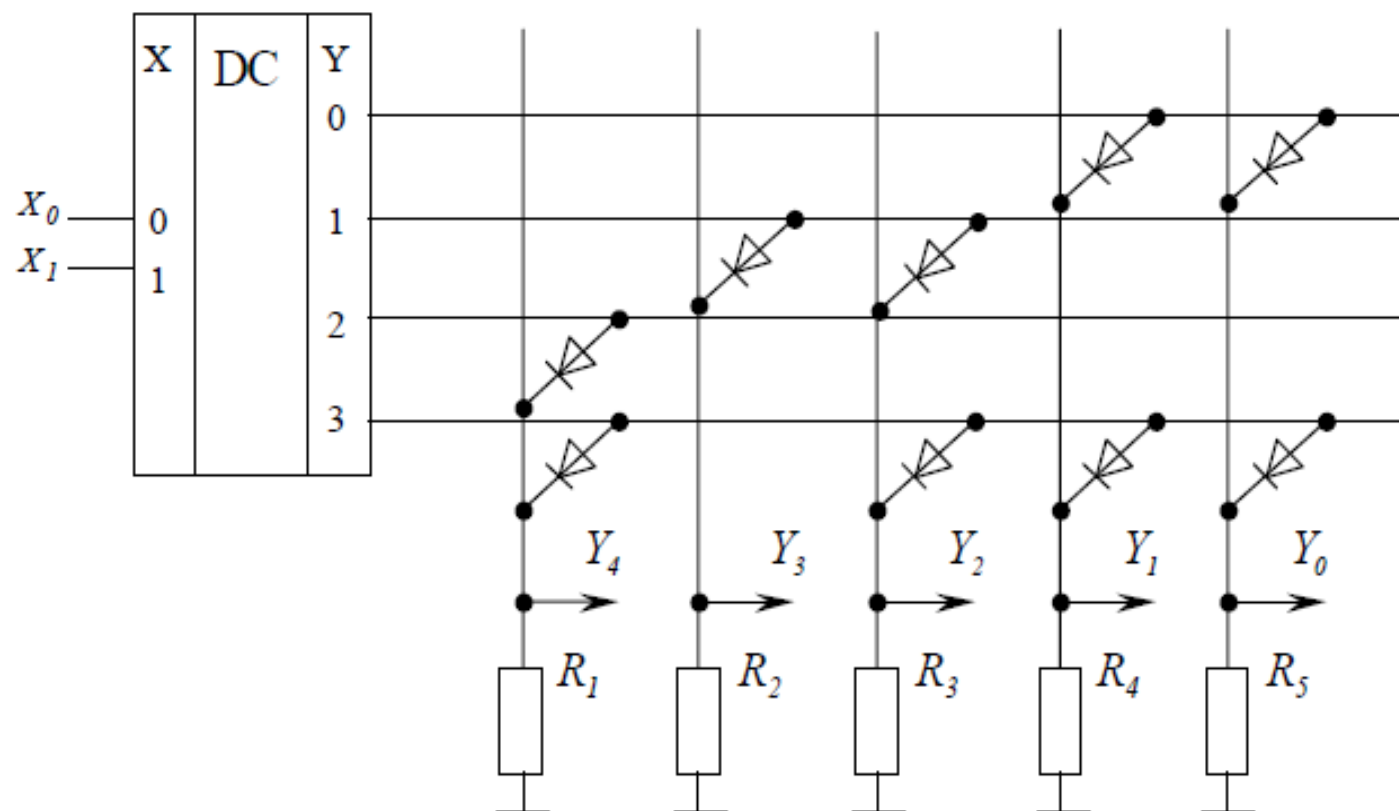
К преобразователям кодов относят широкий класс комбинационных устройств для преобразования n -разрядного кода в m -разрядный. При этом соотношения между n и m могут быть различными. С этой точки зрения дешифраторы ($n < m$) и шифраторы ($n > m$) тоже являются преобразователями кодов.



Условное обозначение и схема построения преобразователей кода

Таблица преобразования преобразователя кода

X_1	X_0	Y_4	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	1	1
0	1	0	1	1	0	0
1	0	1	0	0	1	0
1	1	1	0	1	1	1



Пример реализации преобразователя кода

Преобразователь кодов (кодопреобразователь) – это комбинационное цифровое устройство, которое изменяет код представления числа. Наиболее часто используемые кодопреобразователи: двоично-десятичный код в семисегментный, двоично-десятичный код в двоичный, двоичный код в двоично-десятичный, двоичный код в код Грея, двоичный код в код Хэмминга, двоичный код в код m из n .

При использовании двоично-десятичного кода каждый разряд десятичного числа записывается в виде его четырехбитного прямого двоичного кода. Пример таблицы истинности для кодопреобразователя из прямого двоичного кода в двоично-десятичный для чисел от 0 до 15 представлен в таблице.

Десятичный код	Прямой двоичный код				Двоично-десятичный код							
0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	0	0	0	0	1
2	0	0	1	0	0	0	0	0	0	0	1	0
3	0	0	1	1	0	0	0	0	0	0	1	1
4	0	1	0	0	0	0	0	0	0	0	1	0
5	0	1	0	1	0	0	0	0	0	0	1	1
6	0	1	1	0	0	0	0	0	0	0	1	0
7	0	1	1	1	0	0	0	0	0	0	1	1
8	1	0	0	0	0	0	0	0	0	1	0	0
9	1	0	0	1	0	0	0	0	0	1	0	1
10	1	0	1	0	0	0	0	0	1	0	0	0
11	1	0	1	1	0	0	0	0	1	0	0	1
12	1	1	0	0	0	0	0	0	1	0	0	1
13	1	1	0	1	0	0	0	0	1	0	1	0
14	1	1	1	0	0	0	0	0	1	0	1	1
15	1	1	1	1	0	0	0	0	1	0	1	1

*Таблица истинности
для кодопреобразователя
из прямого двоичного кода
в двоично-десятичный
для чисел от 0 до 15*

Двоично-десятичный код

При помощи четырех **бит** можно закодировать шестнадцать цифр. Из них используются 10. Остальные 6 комбинаций в двоично-десятичном коде являются запрещенными. Таблица соответствия двоично-десятичного кода и десятичных цифр:

Двоично-десятичный код				Десятичный код
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9

Двоично-десятичный код				Дополнительный символ
1	0	1	0	* (звёздочка)
1	0	1	1	# (решётка)
1	1	0	0	+ (плюс)
1	1	0	1	- (минус)
1	1	1	0	, (десятичная запятая)
1	1	1	1	Символ гашения

Двоично-десятичный код также применяется в телефонной связи. В этом случае кроме десятичных цифр кодируются символы '*' или '#' или любые другие. Для записи этих символов в двоично-десятичном коде используются запрещенные комбинации.

Преобразователи кодов

В качестве примера преобразователя кодов, выпускаемых в виде ИС, можно привести схемы, обеспечивающие преобразование информации из двоичного в двоично-десятичный код. Частным случаем преобразователей кода являются шифраторы и дешифраторы.

В качестве примера рассмотрим преобразователь двоично-десятичного кода в код для семисегментных светодиодных индикаторов (рис.). На рисунке также показан фрагмент подключения одного сегмента к выходу схемы с общим эмиттером и приведены начертания первых пяти цифр.

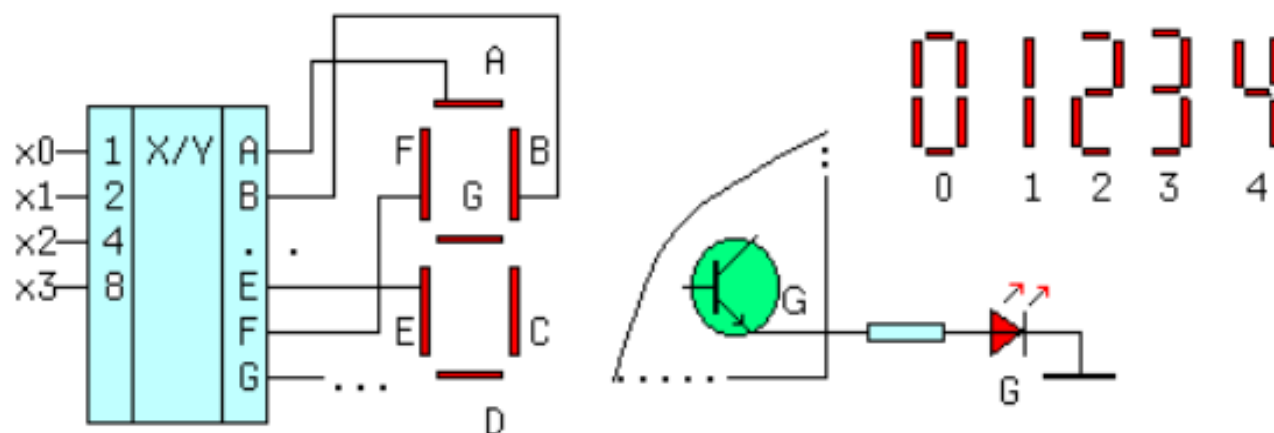


Рис.

Такой преобразователь должен иметь четыре входа, т.к. для кодирования десятичных цифр от 0 до 9 достаточно четырех двоичных, и семь выходов, по одному на каждый сегмент.

Семисегментный код

Таблица истинности для кодопреобразователя из прямого двоичного кода в семисегментный код

Двоичный код				Семисегментный код							Отображаемые цифры
x_3	x_2	x_1	x_0	g	f	e	d	c	b	a	
0	0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	0	0	0	1	1	0	1
0	0	1	0	1	0	1	1	0	1	1	2
0	0	1	1	1	0	0	1	1	1	1	3
0	1	0	0	1	1	0	0	1	1	0	4
0	1	0	1	1	1	0	1	1	0	1	5
0	1	1	0	1	1	1	1	1	0	1	6
0	1	1	1	0	0	0	0	1	1	1	7
1	0	0	0	1	1	1	1	1	1	1	8
1	0	0	1	1	1	0	1	1	1	1	9
1	0	1	0	1	1	1	0	1	1	1	A
1	0	1	1	1	1	1	1	1	0	0	B
1	1	0	0	0	1	1	1	0	0	1	C
1	1	0	1	1	0	1	1	1	1	0	D
1	1	1	0	1	1	1	1	0	0	1	E
1	1	1	1	1	1	1	0	0	0	1	F

Одним из простейших устройств индикации (для отображения цифр и чисел) является семисегментный индикатор (обычно светодиодный) (см. рис.). Такие индикаторы, как следует из названия, состоят из семи светодиодов (иногда добавляют восьмой светодиод – для отображения точки). Зажигая различные комбинации светодиодов (см. рис.) можно отображать цифры от 0 до 9. Сегменты обозначаются буквами от A до G; восьмой сегмент (десятичная точка, decimal point, DP), предназначен для отображения десятичных дробных чисел.

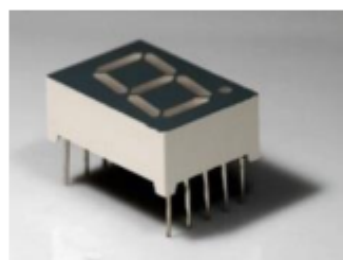


Рис. Внешний вид семисегментного индикатора с точкой

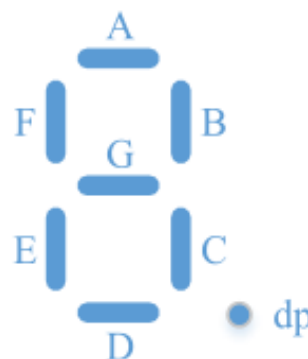


Рис. Схематичное обозначение сегментов



Рис. Пример изображения на семисегментном индикаторе цифр 2 и 9

Код Грея (одношаговый) (часто слово “одношаговый” пропускают) – это такой двоичный код, в котором любые два смежных слова различаются только в одной позиции. Другими словами, расстояние Хэмминга между любыми двумя смежными словами равно единице, включая первое и последнее слова (циклически замкнутый одношаговый код Грея).

Десятичный код	Прямой двоичный код			Код Грея		
	x_2	x_1	x_0	y_2	y_1	y_0
0	0	0	0	0	0	0
1	0	0	1	0	0	1
2	0	1	0	0	1	1
3	0	1	1	0	1	0
4	1	0	0	1	1	0
5	1	0	1	1	1	1
6	1	1	0	1	0	1
7	1	1	1	1	0	0

*Таблица истинности для
кодпреобразователя из прямого
двоичного кода
в одношаговый код Грея*

Перевод N-разрядного числа из прямого двоичного кода в двоичный код Грея выполняется по следующей формуле: $y_i = x_i \text{ XOR } x_{i+1}$, $i = 0, 1, \dots, N-1$. Т. е. для вычисления i-го разряда в коде Грея необходимо взять исключающее ИЛИ от i-го и (i + 1)-го разрядов в прямом двоичном коде. При $i = N-1$ считать $x_{i+1} = 0$. Здесь x_i – разряды исходного числа в прямом двоичном коде, а y_i – разряды числа в коде Грея.

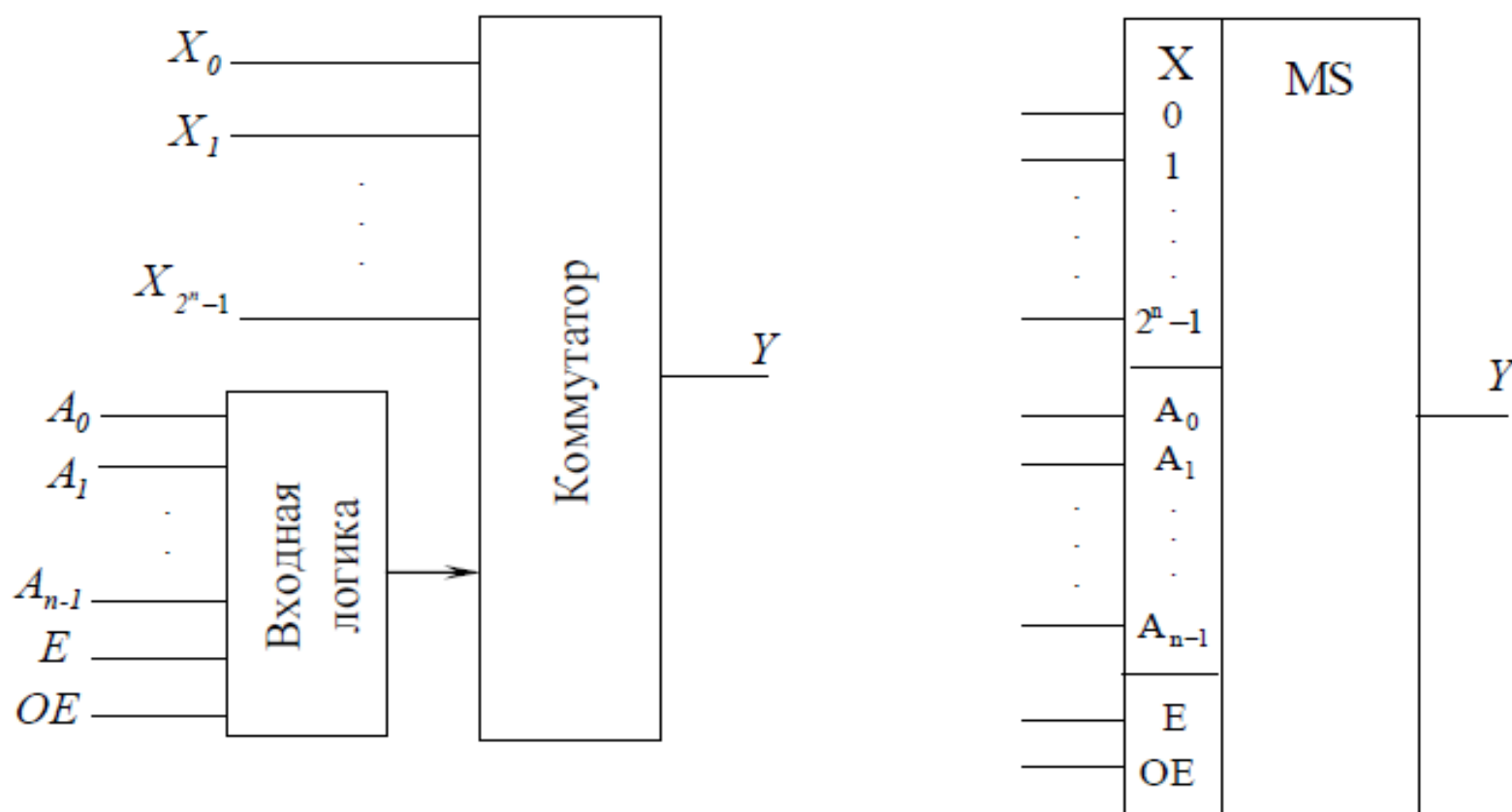
Перевод N-разрядного числа из прямого двоичного кода в двоичный код Грея выполняется по следующей формуле: $y_i = x_i \text{ XOR } x_{i+1}$, $i = 0, 1, \dots, N-1$. Т.е. для вычисления i-го разряда в коде Грея необходимо взять исключающее ИЛИ от i-го и (i + 1)-го разрядов в прямом двоичном коде. При $i = N-1$ считать $x_{i+1} = 0$. Здесь x_i – разряды исходного числа в прямом двоичном коде, а y_i – разряды числа в коде Грея.

Обратный перевод числа из кода Грея в прямой двоичный код выполняется так: $x_i = y_i \text{ XOR } y_{i+1} \dots \text{ XOR } y_{N-1}$. Т.е. i-й разряд в прямом двоичном коде равен исключающему ИЛИ от всех разрядов в коде Грея с номерами не менее i.

Двоичный код Грея широко применяется в качестве манипуляционного кода в различных способах цифровой модуляции. В общем случае удобство применения кода Грея в качестве манипуляционного кода, заключается в том, что при случайном (обычно, ошибочном) переходе между соседними состояниями в системе будет возникать ошибка только в одном разряде.

Мультиплексоры

Мультиплексором называют комбинационное устройство, предназначенное для коммутации цифровых сигналов, поступающих по нескольким информационным входам, на один выход. При этом каждый информационный вход имеет свой адрес, который задается n -разрядным цифровым кодом. Тогда количество информационных входов будет равно $N = 2^n$. Структура мультиплексора условно обозначается записью $(N \rightarrow 1)$.



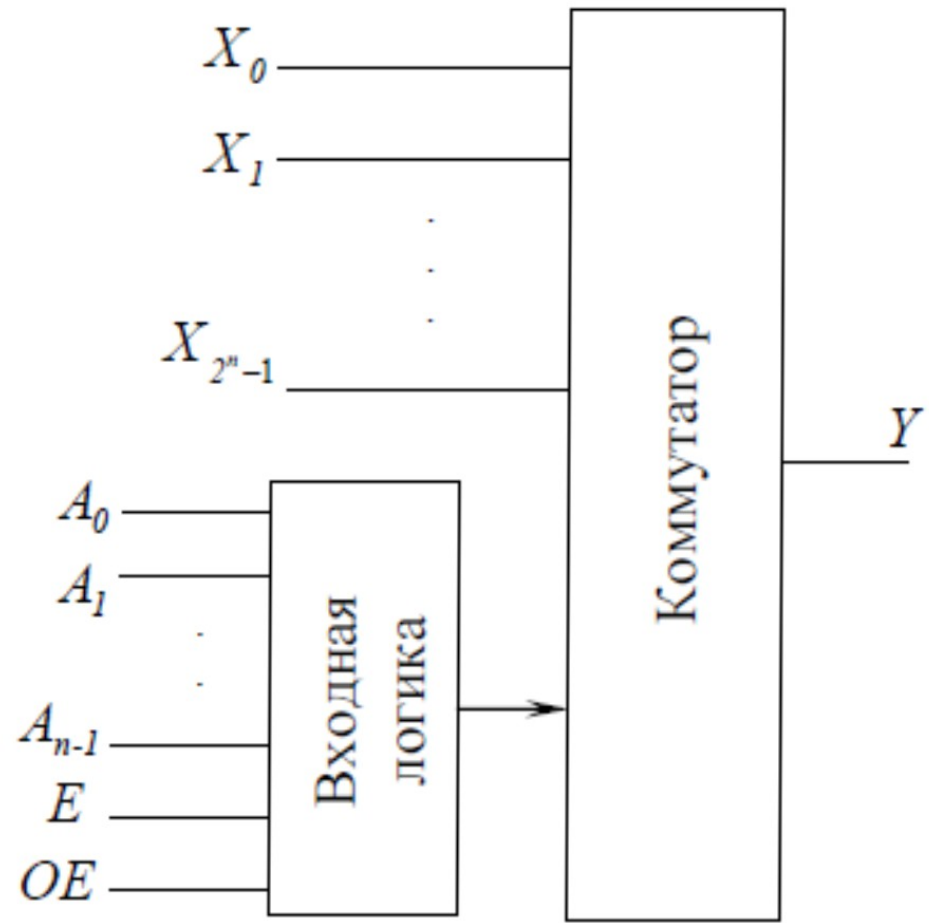
Обобщенная схема и условное обозначение мультиплексора

Мультиплексоры

Выбор входной линии X , информация с которой поступает на выход Y , осуществляется при помощи сигналов, поступающих на адресные входы A .

Входные логические сигналы X поступают на входы коммутатора и через коммутатор передаются на выход Y . Управление коммутатором осуществляется входной логической схемой. На вход логической схемы подаются адресные сигналы A .

Мультиплексоры могут иметь дополнительный управляющий вход E , который может выполнять стробирование выхода Y . Кроме этого, некоторые мультиплексоры могут иметь выход с тремя состояниями: два состояния 0 и 1 и третье состояние — отключенный выход (выходное сопротивление равно бесконечности). Перевод мультиплексора в третье состояние производится сигналом OE .



Обобщенная схема мультиплексора

Таблица истинности
мультиплексора

A_1	A_0	Y	СКНФ
0	0	X_0	$\bar{A}_0 \bar{A}_1 X_0 E$
0	1	X_1	$A_0 \bar{A}_1 X_1 E$
1	0	X_2	$\bar{A}_0 A_1 X_2 E$
1	1	X_3	$A_0 A_1 X_3 E$

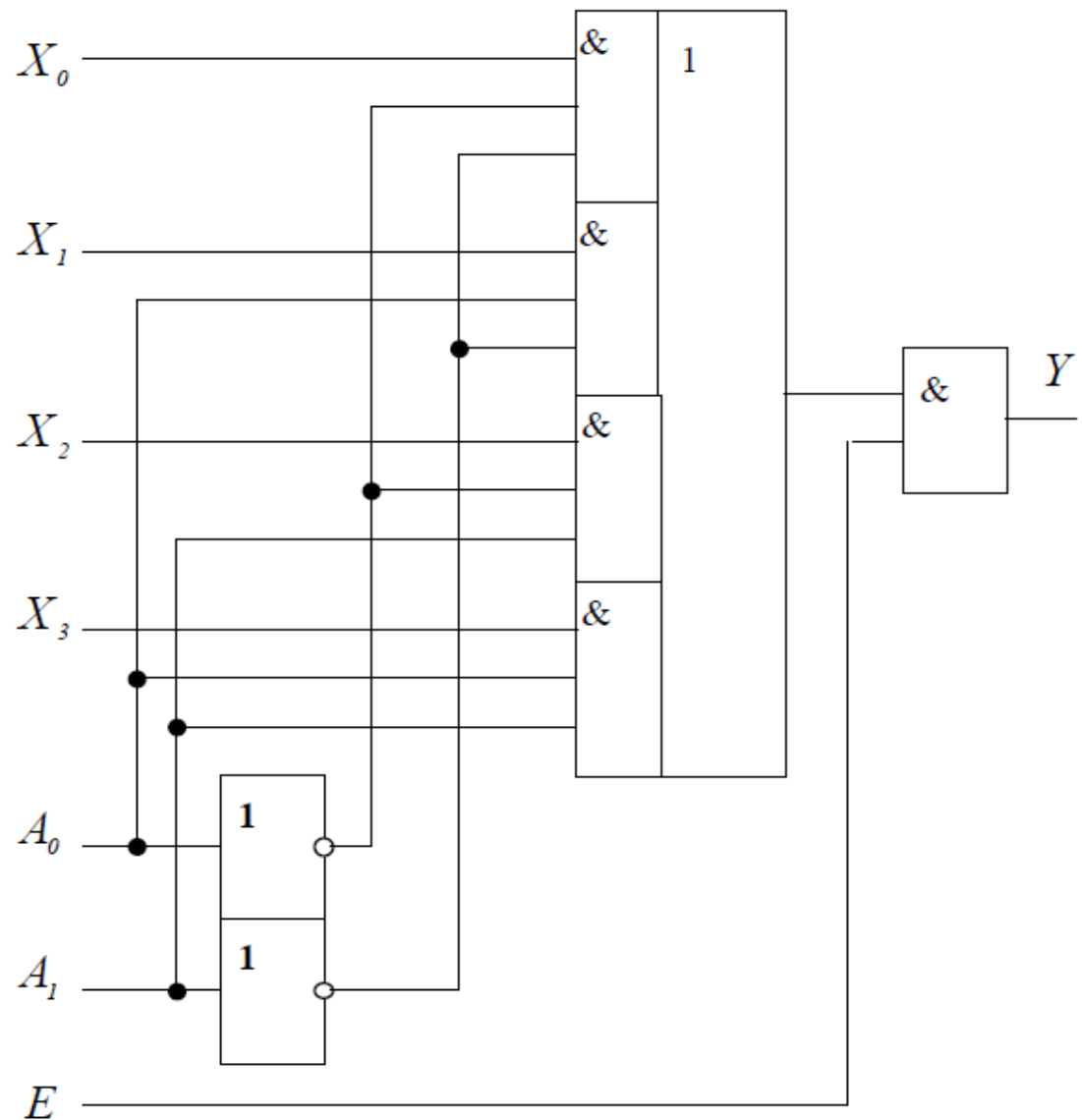


Схема мультиплексора $4 \rightarrow 1$

Функционирование синтезируемого мультиплексора описывается логическим выражением

$$Y = (\bar{A}_0 \bar{A}_1 X_0 + A_0 \bar{A}_1 X_1 + \bar{A}_0 A_1 X_2 + A_0 A_1 X_3) E.$$

Мультиплексоры

Большинство мультиплексоров способно передавать сигналы информации только в одном направлении — от входа на выход. Однако имеются и двунаправленные мультиплексоры (селекторы-мультиплексоры), которые могут передавать информационные сигналы (не только цифровые, но и аналоговые) в обоих направлениях.

Для расширения числа входных линий можно использовать каскадирование мультиплексоров.

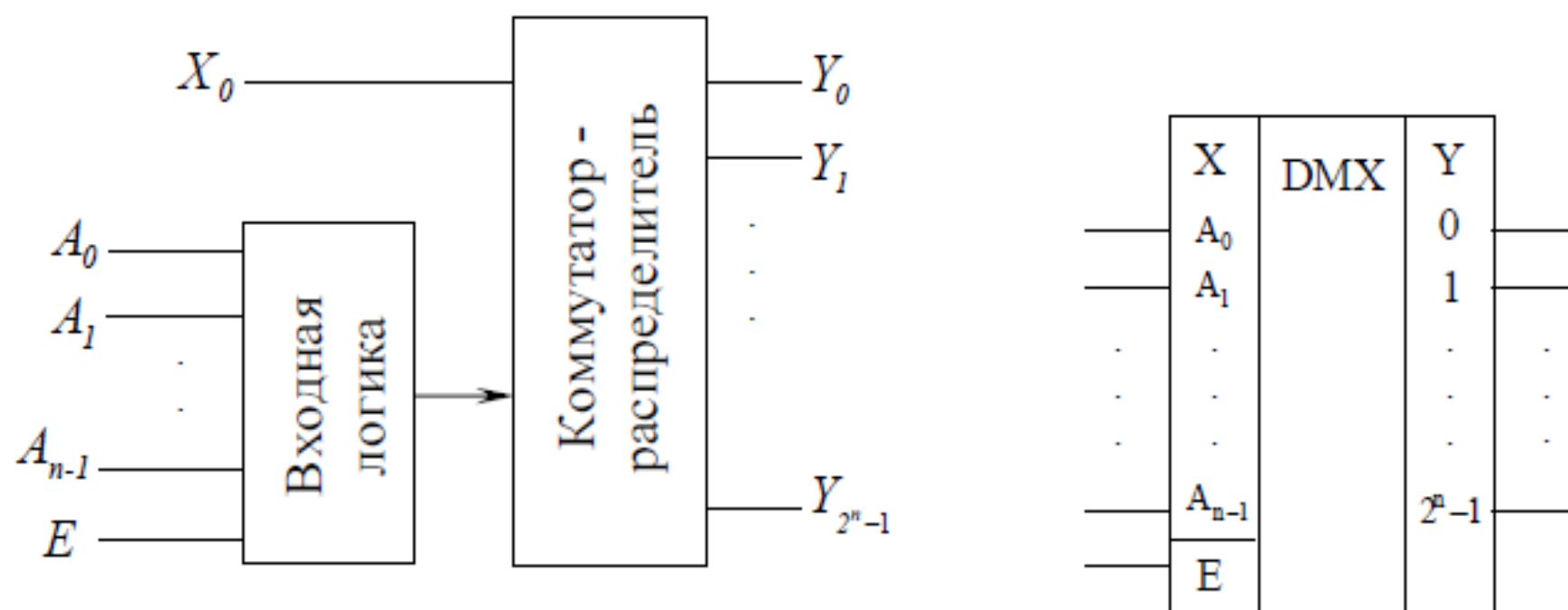
Интегральные микросхемы мультиплексоров можно разделить на группы по следующим признакам:

- по числу входов: 2-, 4-, 8- и 16-входовые;
- по числу мультиплексоров в одном корпусе (числу разрядов);
- по наличию стробирующего входа Е;
- по наличию выхода с тремя состояниями (наличию входа ОЕ);
- по способности передавать сигналы в двух направлениях.

Помимо основного назначения коммутации входных сигналов мультиплексоры находят применение в сдвигающих устройствах, делителях частоты, триггерных устройствах и др.

Демультимплексоры

Демультимплексоры выполняют функцию, обратную мультиплексорам, т.е. один входной сигнал распределяют по нескольким выходам. При этом каждый выход имеет свой адрес, который задается n -разрядным цифровым кодом. Тогда количество выходов будет равно $N = 2^n$. Структура демультимплексора условно обозначается записью $(1 \rightarrow N)$.



Обобщенная схема и условное обозначение демультимплексора

Таблица истинности демультиплексора

A_1	A_0	Y_3	Y_2	Y_1	Y_0	СКНФ
0	0	0	0	0	X	$\overline{A_0} \overline{A_1} X E$
0	1	0	0	X	0	$A_0 \overline{A_1} X E$
1	0	0	X	0	0	$\overline{A_0} A_1 X E$
1	1	X	0	0	0	$A_0 A_1 X E$

Входной сигнал X поступает на вход коммутатора и через него передается на выходы Y . Адресные сигналы A имеют то же значение, что и у мультиплексора. Сигнал стробирования E разрешает передачу входного сигнала через коммутатор.

логические выражения, описывающие работу синтезируемого демультиплексора:

$$Y_0 = \overline{A_0} \overline{A_1} X E; \quad Y_1 = A_0 \overline{A_1} X E;$$

$$Y_2 = \overline{A_0} A_1 X E; \quad Y_3 = A_0 A_1 X E.$$

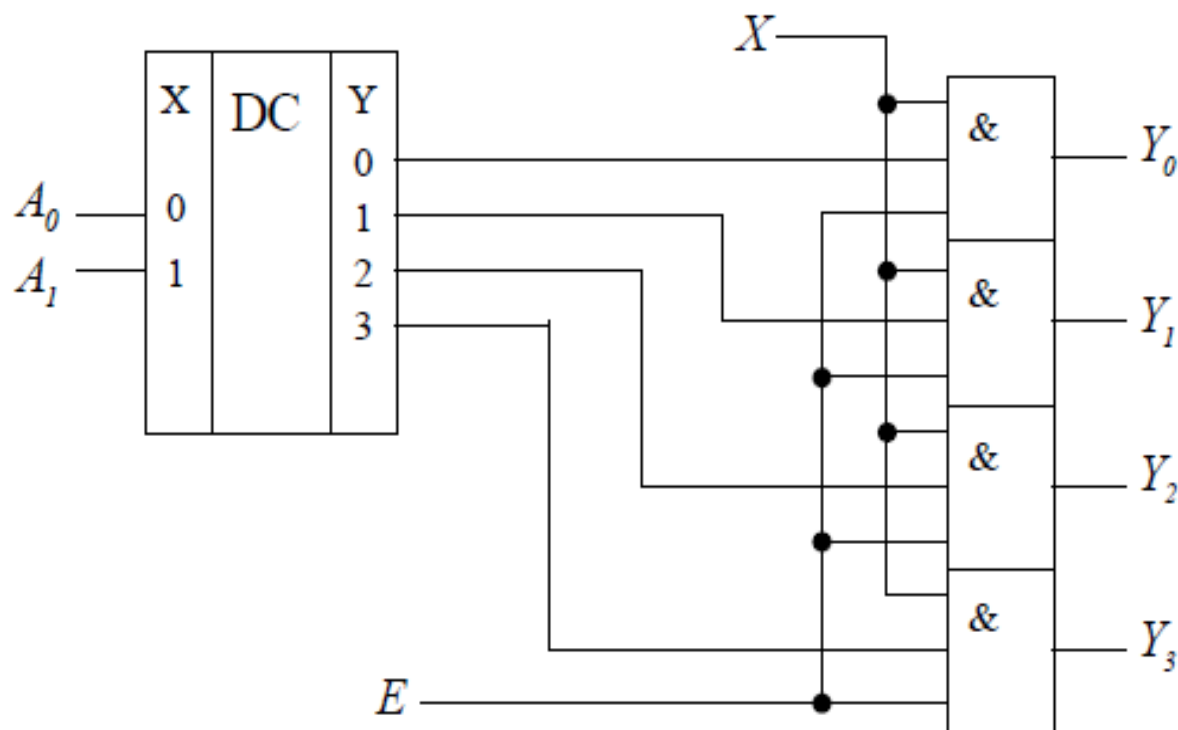


Схема демультиплексора $1 \rightarrow 4$

Демультимплексоры

Демультимплексором называют функциональный узел, который обеспечивает передачу цифровой информации, поступающей по одной линии, на несколько выходных линий. Выбор выходной линии осуществляется при помощи сигналов, поступающих на адресные входы. Таким образом, демультимплексор выполняет преобразование, обратное действию мультиплексора.

Интегральные микросхемы демультимплексоров можно разделить на группы по следующим признакам:

- по числу выходов;
- по числу демультимплексоров в одном корпусе;
- по наличию стробирующего импульса Е;
- по способности передавать сигналы в двух направлениях.

Среди схем коммутации можно особо выделить схемы, которые способны пропускать сигналы в обоих направлениях. К таким элементам относятся коммутационные микросхемы, выполненные по технологии КМОП. Коммутаторы КМОП способны пропускать как аналоговые, так и цифровые сигналы, в них можно менять местами ВХОД и ВЫХОД.

МУЛЬТИПЛЕКСОРЫ И ДЕМУЛЬТИПЛЕКСОРЫ

Мультиплексоры и демultipлексоры решают задачи направления потоков данных в программируемой логике. В зависимости от состояния управляющих сигналов мультиплексоры и демultipлексоры выбирают один из нескольких возможных путей прохождения данных.

Мультиплексоры могут использоваться для реализации логических функций многих переменных, в делителях частоты (в современных ПЛИС так делать категорически не рекомендуется), триггерных устройствах, сдвигающих устройствах и др. Кроме того, они применяются для стробирования сигналов тактирования в целях снижения энергопотребления устройства.

При проектировании цифровых устройств часто возникает задача направления потока данных от одного из множества источников к одному из множества получателей. Т.е. необходимо устройство, которое подключает (коммутирует) определенные входные порты к определенным выходным портам. Эту задачу решает **коммутатор**.

КОММУТАТОР

При проектировании цифровых устройств часто возникает задача направления потока данных от одного из множества источников к одному из множества получателей. Т.е. необходимо устройство, которое подключает (коммутирует) определенные входные порты к определенным выходным портам. Эту задачу решает **коммутатор**.

Рассмотрим коммутатор, который в каждый момент времени работает только с одной парой входного и выходного портов. Такой коммутатор имеет N_{in} информационных входов, N_{out} информационных выходов, M_{in} и M_{out} адресных входов для выбора соединяемых входного и выходного информационных портов соответственно.

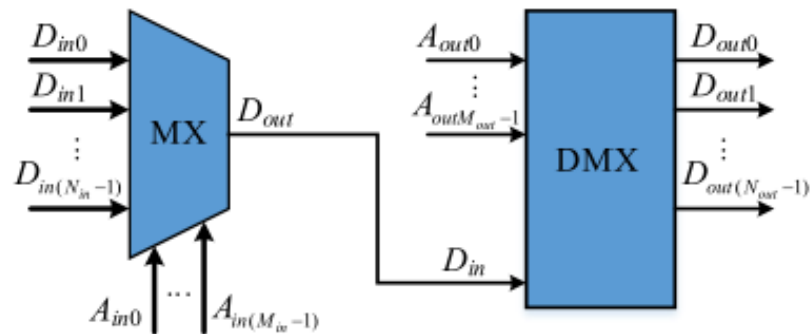


Рис. Последовательное соединение мультиплексора и демультиплексора для построения коммутатора

Таблица истинности для коммутатора 4х4.

A_{in1}	A_{in0}	A_{out1}	A_{out0}	D_{out0}	D_{out1}	D_{out2}	D_{out3}
0	0	0	0	D_{in0}	0	0	0
0	0	0	1	0	D_{in0}	0	0
0	0	1	0	0	0	D_{in0}	0
0	0	1	1	0	0	0	D_{in0}
0	1	0	0	D_{in1}	0	0	0
0	1	0	1	0	D_{in1}	0	0
0	1	1	0	0	0	D_{in1}	0
0	1	1	1	0	0	0	D_{in1}
1	0	0	0	D_{in2}	0	0	0
1	0	0	1	0	D_{in2}	0	0
1	0	1	0	0	0	D_{in2}	0
1	0	1	1	0	0	0	D_{in2}
1	1	0	0	D_{in3}	0	0	0
1	1	0	1	0	D_{in3}	0	0
1	1	1	0	0	0	D_{in3}	0
1	1	1	1	0	0	0	D_{in3}

КОММУТАТОР

Если соединить выходы демультиплексора со входами мультиплексора (см. рис.) при $N_{in} = N_{out}$, то получится устройство с одним информационным входом и одним информационным выходом. Такое устройство позволяет разделять (разуплотнять) один высокоскоростной канал данных на множество низкоскоростных, что упрощает обработку данных в каждом низкоскоростном канале. После того, как данные обработаны, их можно снова объединить в один высокоскоростной канал. По такому принципу работают современные высокоскоростные последовательные трансиверы в FPGA.

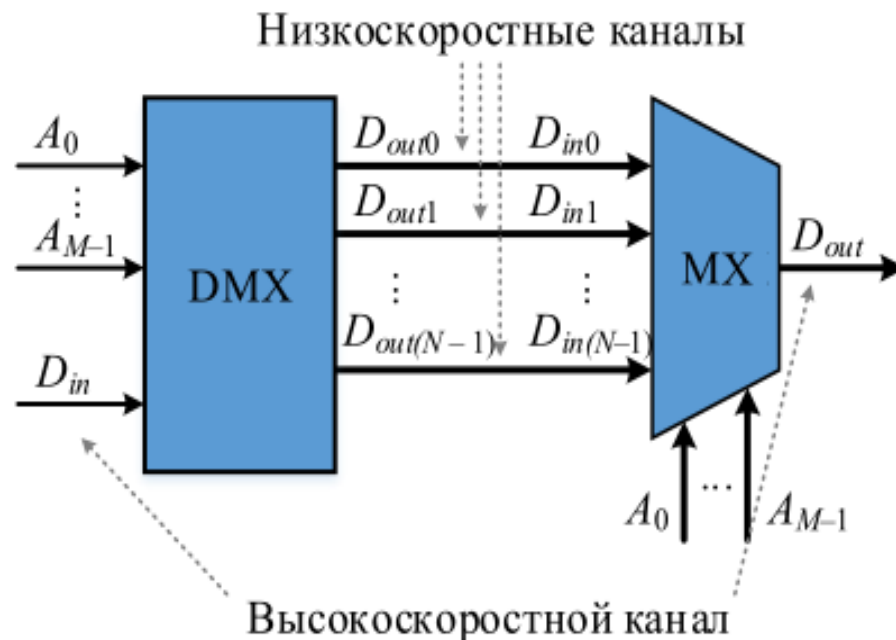


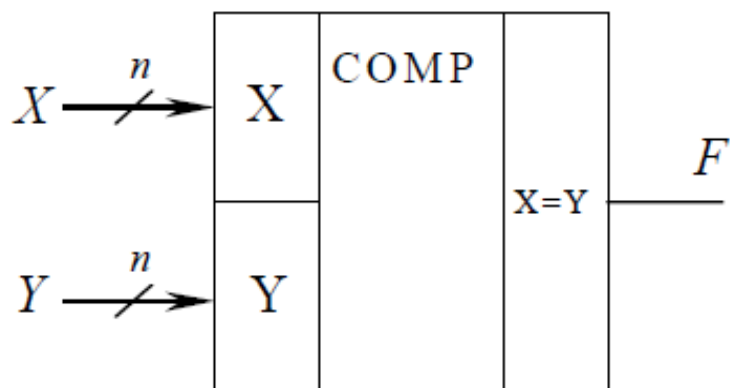
Рис. Последовательное соединение демультиплексора и мультиплексора для разуплотнения высокоскоростного канала

СХЕМЫ СРАВНЕНИЯ

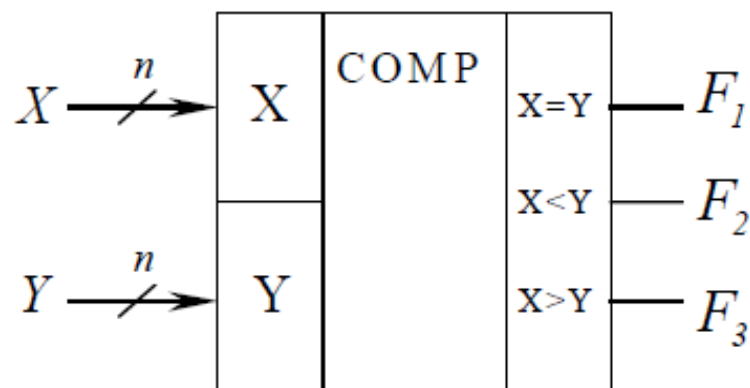
Схемы сравнения или цифровые компараторы предназначены для реализации операции сравнения двух кодов, например X и Y , и бывают двух типов:

- 1) простейшие или схемы равнозначности, выдающие сигнал $F=1$ при $X=Y$;
- 2) универсальные, выдающие три сигнала: $F_1=1$, если $X=Y$;
 $F_2=1$, если $X<Y$; $F_3=1$, если $X>Y$.

Условные обозначения схем сравнения приведены на рис.



а



б

Рис. Условные обозначения схем сравнения

СХЕМЫ СРАВНЕНИЯ

Простейшие схемы сравнения строятся на базе элементов ИСКЛЮЧАЮЩЕЕ ИЛИ по схеме, приведенной на рис. (пример сравнения для 4-разрядных кодов).

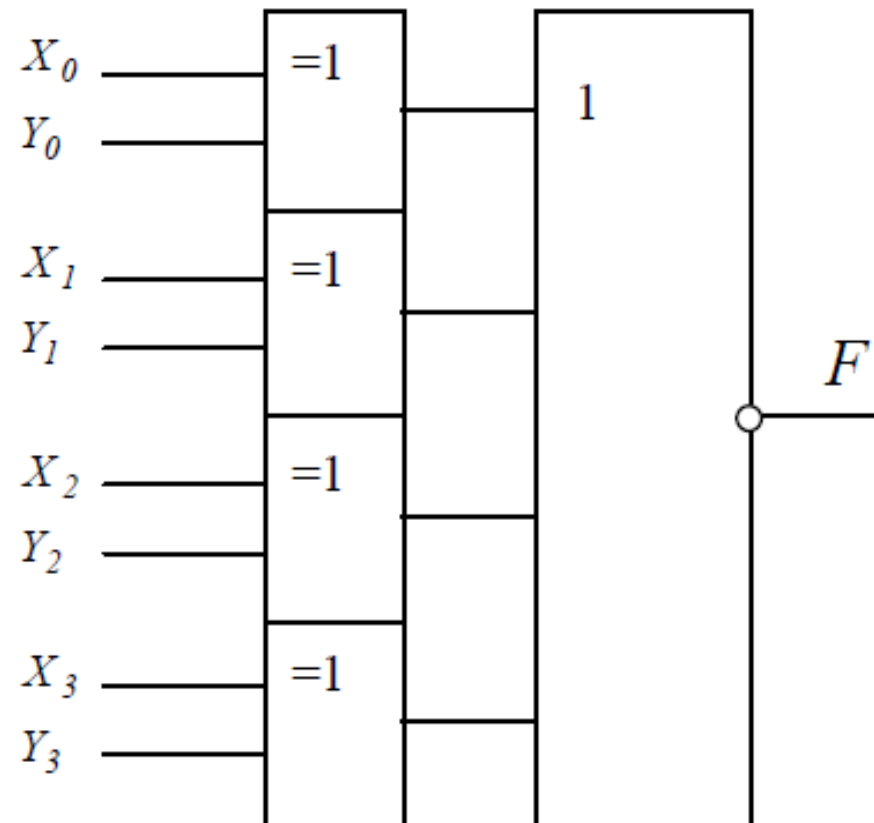
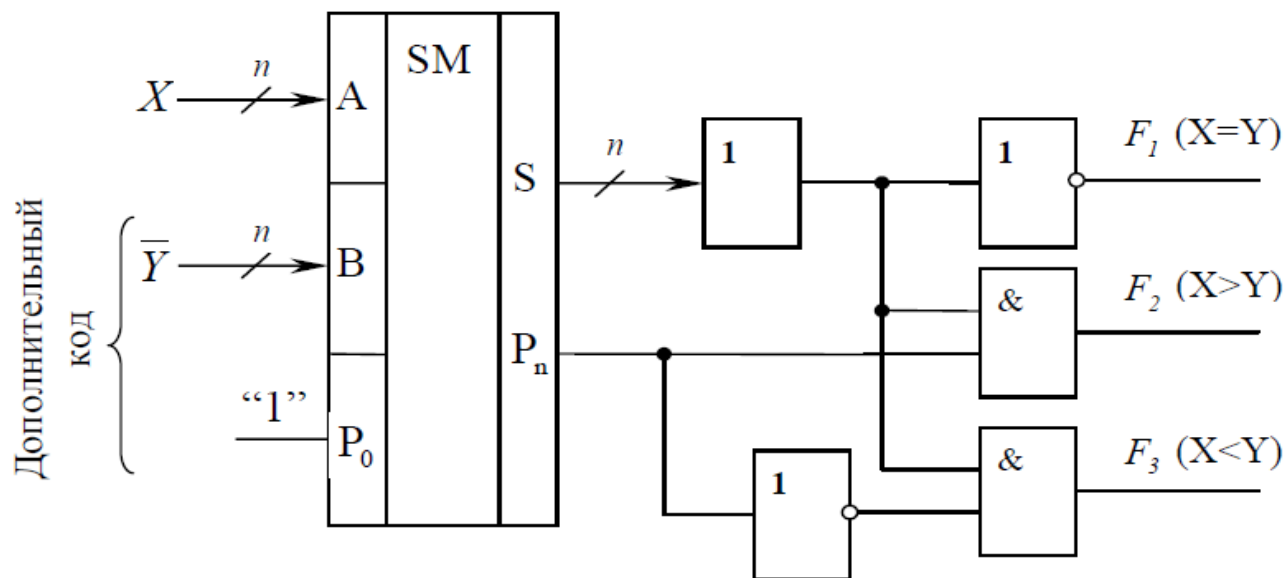


Схема равнозначности 4-разрядных кодов

СХЕМЫ СРАВНЕНИЯ

Для построения универсальных схем сравнения используют сумматоры, на которых реализуют операцию вычитания $X-Y$. Для этого переменная Y подается в обратном коде и на вход переноса сумматора подается «1», что эквивалентно представлению Y в дополнительном коде. На рис. показан пример реализации схемы сравнения n -разрядных кодов.



Универсальная схема сравнения на базе сумматора

Выходная логика построена с учетом реализации следующих соотношений:

- 1) если $X=Y$, то $S=0$ и $F_1=1$;
- 2) если $X<Y$, то $S \neq 0$, $P_n=1$ и $F_2=1$;
- 3) если $X>Y$, то $S \neq 0$, $P_n=0$ и $F_3=1$.

МАЖОРИТАРНЫЕ ЭЛЕМЕНТЫ

Мажоритарный элемент — логический элемент, с чётным или нечётным числом входов и одним выходным сигналом, значение которого совпадает со значением, на большинстве входов. При чётном числе входов большинством считается $n/2+1$, соответственно $n/2$ к большинству не относится. Таким образом, элемент работает по «принципу большинства»: если на большинстве входов будет сигнал «1», то и на выходе схемы установится сигнал «1»; и наоборот, если на большинстве входов будет сигнал «0», то и на выходе установится «0».

Мажоритарные элементы используются в цифровых высоконадёжных системах и устройствах, например, в системах резервирования, в помехоустойчивых телекоммуникационных системах.

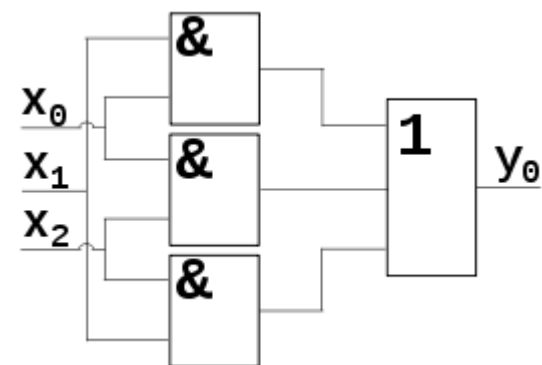


Схема мажоритарного элемента с тремя входами