

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Локтионова Оксана Геннадьевна  
Должность: проректор по учебной работе  
Дата подписания: 24.04.2024 12:05:32  
Уникальный программный ключ:  
0b817ca911e6668abb13a5d426d39e5f1c11eabbf73e943df4a4851fda56d089

## МИНОБРАЗОВАНИЯ РОССИИ

Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра вычислительной техники

УТВЕРЖДАЮ  
Проректор по учебной работе  
О.Г. Локтионова  
«20» 4 2023г.



### ПРОЕКТИРОВАНИЕ ФУНКЦИОНАЛЬНЫХ УЗЛОВ КОМБИНАЦИОННОГО ТИПА

Методические указания к практической работе  
для студентов направления 09.03.01

Курск 2023

УДК 519.713.1

Составитель: И.Е. Чернецкая

Рецензент

Кандидат технических наук, доцент *Т.Н. Конаныхина*

**Проектирование функциональных узлов комбинационного типа:** методические указания к практической работе для студентов направления 09.03.01/ Юго-Зап. гос. ун-т; сост.; И.Е. Чернецкая. – Курск, 2023. - 27 с.: - ил. 21 , табл. 4.– Библиогр.: с. 27.

Содержат сведения по вопросам проектирования функциональных узлов комбинационного типа: сумматоров, шифраторов и дешифраторов, мультиплексоров и демультимплексоров, компараторов. Теоретический материал проиллюстрирован на примерах.

Методические указания соответствуют рабочей программе дисциплины «Теория автоматов».

Предназначены для студентов направления 09.03.01 Информатика и вычислительная техника.

Текст печатается в авторской редакции

Подписано в печать . Формат 60\*84 1/16.  
Усл. печ. л.1,60. Уч.-изд. л. 1,42. Тираж 50 экз. Заказ 1282. Бесплатно.  
Юго-Западный государственный университет.  
305040 Курск, ул. 50 лет Октября, 94.

## Оглавление

Цель работы	4
1 Теоретические основы	4
2 Проектирование сумматоров	4
3 Проектирование шифраторов и дешифраторов	12
4 Проектирование мультиплексоров и демультимплексоров	19
5 Проектирование компараторов	23
6 Порядок выполнения работ	26
7 Содержание отчета	26
8 Примерные задания для реализации	26
9 Контрольные вопросы	27
Список литературы	27

## **Цель работы**

Цель выполнения лабораторной работы заключается в ознакомлении с проектированием функциональных узлов комбинационного типа.

## **1 Теоретические основы**

Интегральные логические элементы являются основой для построения цифровых устройств, выполняющих более сложные операции и относящихся к классу комбинационных устройств.

Задача синтеза функционального узла комбинационного типа (КФУ) заключается в построении оптимальной схемы проектируемого узла, моделирующей закон функционирования цифрового автомата без памяти, представленного одной булевой функцией или системой булевых функций. К требованиям оптимальности могут быть отнесены стоимость и сложность оборудования, быстродействие и надежность, однородность структуры и др.

Основные из КФУ: дешифраторы и шифраторы; мультиплексоры и демультиплексоры; двоичные сумматоры; цифровые компараторы и мажоритарные элементы; преобразователи кодов и др.

## **2 Проектирование сумматоров**

**Сумматоры** – это цифровые функциональные устройства, предназначенные для выполнения операции сложения чисел, представленных в различных кодах.

По характеру действия сумматоры подразделяются на комбинационные, не имеющие элементов памяти, и накапливающие – запоминающие результаты вычислений при снятии входных сигналов.

В дальнейшем будут рассматриваться только комбинационные сумматоры, на основе которых выполняется большинство суммирующих ИС.

С помощью сумматора можно выполнять также операции вычитания, умножения, деления, преобразования чисел в дополнительный код и другие дополнительные операции. Классифицируются сумматоры по трем основным признакам:

- 1) по числу входов (полусумматоры, одноразрядные и многоразрядные сумматоры);
- 2) по способу тактирования;
- 3) по системе счисления.

По способу тактирования различают синхронные и асинхронные сумматоры. В синхронных сумматорах на суммирование любой пары чисел

отводится максимальное время. В этом случае время, отводимое на суммирование используется в большинстве случаев неэффективно. Так, в ряде работ доказано, что средняя длина распространения переноса -  $\Pi_{cp}^{p.n.} < n$ , что  $\Pi_{cp}^{p.n.} \approx \log_2 n$ , где  $n$  - количество разрядов в числе.

В асинхронных сумматорах на суммирование любой пары чисел отводится необходимое время. В таких сумматорах имеются цепи, которые определяют момент окончания распространения переноса.

По системе счисления различают сумматоры двоичные, двоично-десятичные и др. Двоично-десятичные сумматоры выполняют действия над десятичными числами, разряды которых закодированы двоичными тетрадами. Обычный способ построения двоично-десятичного сумматора предусматривает первичное суммирование тетрад обычным двоичным сумматором и последующую коррекцию результата.

### Проектирование полусумматоров

Полусумматором называют КФУ с двумя входами ( $a, b$ ) и двумя выходами ( $S, P$ ), на которых вырабатываются сигналы суммы и переноса. Условное графическое изображение и таблица истинности полусумматора приведены на рис. 3.1, а, б.

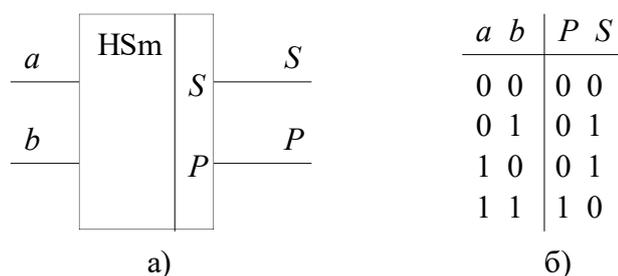


Рисунок 1 – Полусумматор: а) – условное графическое обозначение; б) – таблица истинности

Из таблицы истинности (см. рисунок 1, б) следует:

$$\begin{cases} S = \bar{a}b \vee a\bar{b} = (a \vee \bar{b})(\bar{a} \vee b), \\ P = ab. \end{cases} \quad (1)$$

Преобразуем выражение (3.1) к виду, удобному для реализации на элементах И-НЕ:

$$\begin{cases} S = \overline{\overline{ab} \vee \overline{a\bar{b}}} = \overline{\overline{ab} \cdot \overline{a\bar{b}}}, \\ P = \overline{\overline{ab}}. \end{cases} \quad (2)$$

Функциональная схема полусумматора, построенная по выражениям (2) показана на рисунке 2, а. Как показывает опыт, простая интерпретация минимальных форм булевых функций функциональными компонентами, обычно не приводит к минимальным результатам. Чтобы минимизировать

затраты оборудования в схеме необходимо минимизировать число инверсий над входными переменными. Для этого используем следующее правило:

$$\overline{XY} = (\overline{X \vee Y}) \cdot Y = \overline{X \cdot Y} \cdot Y. \quad (3)$$

Используя правило (3) преобразуем выражение (2) к виду:

$$\begin{cases} S = \overline{ab} \cdot b \vee \overline{ab} \cdot a = \overline{\overline{\overline{ab}} \cdot b} \cdot \overline{\overline{\overline{ab}} \cdot a}, \\ P = \overline{\overline{ab}}. \end{cases} \quad (4)$$

По выражению (4) построена схема полусумматора на элементах типа И-НЕ, приведенная на рисунке 2, б.

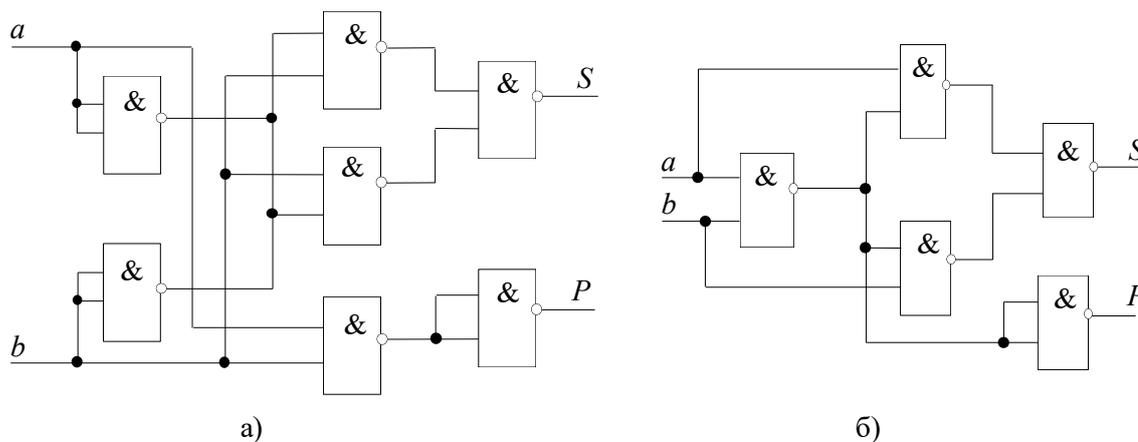


Рисунок 2 – Полусумматор, функциональные схемы на элементах И-НЕ

Она содержит на два элемента меньше, чем эквивалентная ей схема на рисунке 2, а.

### Проектирование одноразрядных сумматоров

Одноразрядным сумматором называют КФУ с тремя входами и двумя выходами. Это основной элемент многоразрядных сумматоров. Он обеспечивает арифметическое сложение одноразрядных двоичных чисел  $a_i$ ,  $b_i$  и переноса из предыдущего разряда  $P_{i-1}$  с образованием на выходах суммы  $S_i$  и переноса в старший разряд  $P_i$ . Условное графическое обозначение и таблица истинности одноразрядного сумматора приведены на рисунке 3, а и б.

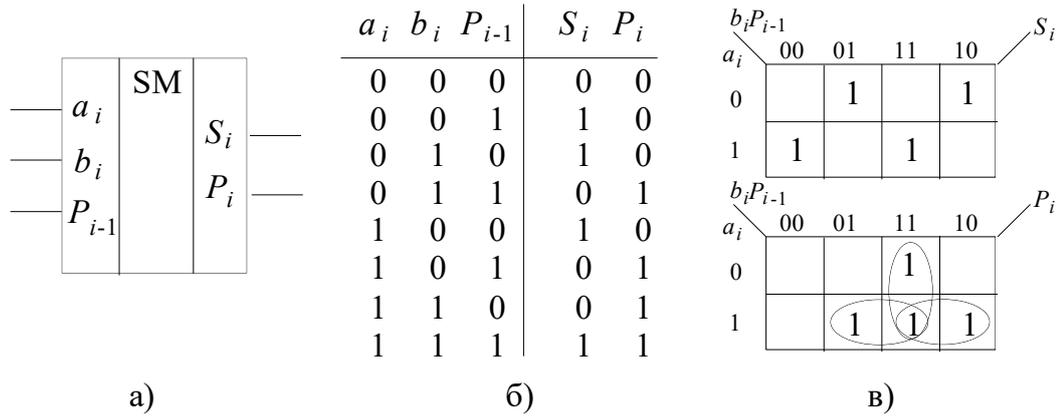


Рисунок 3 – Одноразрядный сумматор: а) – условное графическое обозначение; б) – таблица истинности; в) – карта Карно

Для минимизации функций  $S_i$ ,  $P_i$  отобразим их на карте Карно (см. рисунок 3, в). Функция  $S_i$  минимизации не подлежит, поэтому запишем ее в СДНФ.

$$S_i = \bar{a}_i \cdot \bar{b}_i \cdot P_{i-1} \vee \bar{a}_i \cdot b_i \cdot \bar{P}_{i-1} \vee a_i \cdot \bar{b}_i \cdot \bar{P}_{i-1} \vee a_i \cdot b_i \cdot P_{i-1}. \quad (5)$$

Функцию  $P_i$  после минимизации можно записать в виде

$$P_i = a_i \cdot b_i \vee a_i \cdot P_{i-1} \vee b_i \cdot P_{i-1} = a_i \cdot b_i \vee P_{i-1} \cdot (a_i \vee b_i). \quad (6)$$

При проектировании одноразрядных сумматоров учитываются особенности выбранной системы логических элементов. Рассмотрим пример синтеза одноразрядного сумматора на логических элементах типа И-НЕ. Для реализации функции  $S_i$  выполним следующие преобразования выражения (5):

$$\begin{aligned} S_i &= \bar{a}_i \cdot \bar{b}_i \cdot P_{i-1} \vee \bar{a}_i \cdot b_i \cdot \bar{P}_{i-1} \vee a_i \cdot \bar{b}_i \cdot \bar{P}_{i-1} \vee a_i \cdot b_i \cdot P_{i-1} = \\ &= P_{i-1} \cdot (\bar{a}_i \cdot \bar{b}_i \vee a_i \cdot b_i) \vee \bar{P}_{i-1} \cdot (\bar{a}_i \cdot b_i \vee a_i \cdot \bar{b}_i) = P_{i-1} \cdot C_i \vee \bar{P}_{i-1} \cdot \bar{C}_i, \end{aligned}$$

где  $C_i = \bar{a}_i \cdot \bar{b}_i \vee a_i \cdot b_i$ .

Полученные выражения для  $S_i$ ,  $C_i$ ,  $P_i$  преобразуем к виду, удобному для реализации на элементах типа И-НЕ:

$$\begin{aligned} C_i &= \overline{\overline{\bar{a}_i \cdot \bar{b}_i \vee a_i \cdot b_i}} = \overline{\overline{\bar{a}_i} \cdot \overline{\bar{b}_i} \cdot \overline{a_i} \cdot \overline{b_i}}, \\ S_i &= \overline{\overline{P_{i-1} \cdot C_i \vee \bar{P}_{i-1} \cdot \bar{C}_i}} = \overline{\overline{P_{i-1} \cdot C_i} \cdot \overline{\bar{P}_{i-1} \cdot \bar{C}_i}}, \\ P_i &= \overline{\overline{a_i \cdot b_i \vee P_{i-1} \cdot (a_i \vee b_i)}} = \overline{\overline{a_i \cdot b_i} \cdot \overline{P_{i-1} \cdot (a_i \vee b_i)}}. \end{aligned} \quad (7)$$

Схема одноразрядного сумматора, реализующая уравнения (7), представлена на рисунке 4.

Для упрощения схемной реализации функции  $S_i$  выполним следующие преобразования. Представим  $S_i$  как функцию четырех

переменных:  $a_i$ ,  $b_i$ ,  $P_{i-1}$  и  $P_i$ . Для этого составим таблицу истинности (см. рисунок 5, а), где на нереальных входных наборах функция  $S_i$  принимает неопределенные значения (\*). После минимизации с помощью карты Карно (см. рисунок 5, б) запишем

$$S_i = a_i \bar{P}_i \vee P_{i-1} \bar{P}_i \vee b_i \bar{P}_i \vee a_i b_i P_{i-1}. \quad (8)$$

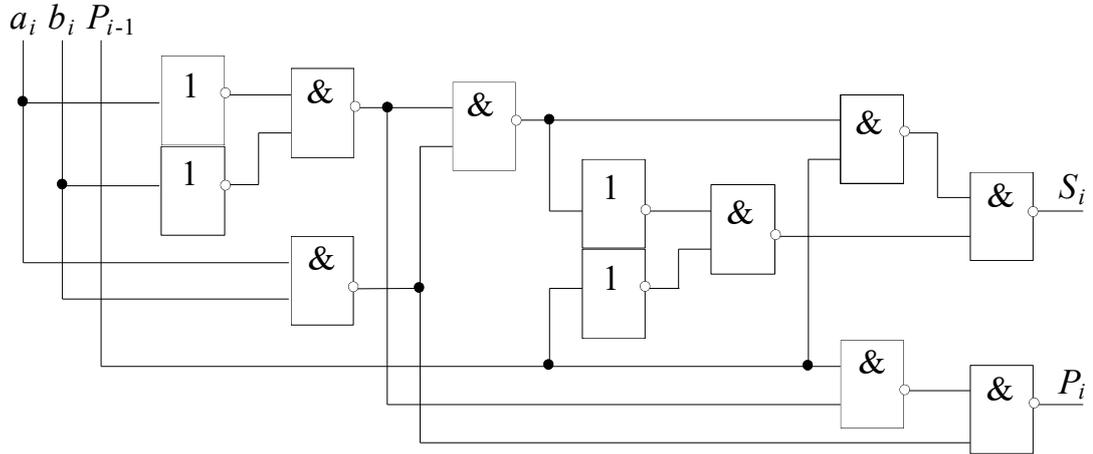
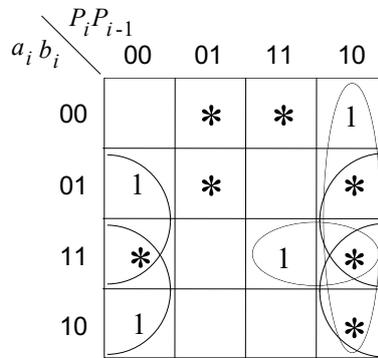


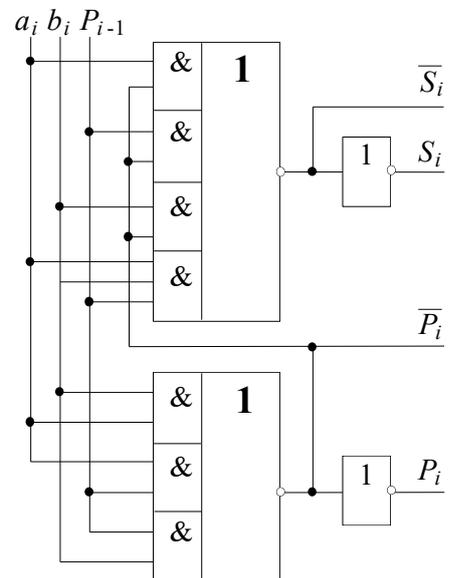
Рисунок 4 – Одноразрядный сумматор

$a_i$	$b_i$	$P_{i-1}$	$P_i$	$S_i$
0	0	0	0	0
0	0	0	1	*
0	0	1	0	1
0	0	1	1	*
0	1	0	0	1
0	1	0	1	*
0	1	1	0	*
0	1	1	1	0
1	0	0	0	1
1	0	0	1	*
1	0	1	0	*
1	0	1	1	0
1	1	0	0	*
1	1	0	1	0
1	1	1	0	*
1	1	1	1	1

а)



б)



в)

Рисунок 5 – Одноразрядный сумматор: а) – табличное представление функции  $S_i$ ; б) – минимизация функции  $S_i$ ; в) – функциональная схема

Используя полученные выражения для суммы (8) и переноса (6) легко построить схему одноразрядного сумматора, которая положена в основу микросхемы сумматора (см. рисунок 5, в).

Быстродействие одноразрядных сумматоров оценивают задержками распространения сигналов по четырем трактам: «слагаемое - сумма» ( $c, s$ ), «слагаемое - перенос» ( $c, p$ ), «перенос - сумма» ( $p, s$ ) и «перенос - перенос» ( $p, p$ ).

## Проектирование многоразрядных сумматоров

Многоразрядные сумматоры делятся на последовательные и параллельные. В последовательном многоразрядном сумматоре (на рисунке 6) подача слагаемых начинается с младшего разряда. Образуется поразрядная сумма и перенос, который запоминается на один такт, а затем поступает на вход сумматора, вместе со слагаемыми следующего разряда. Такой процесс продолжается до окончательного формирования результата, выдаваемого последовательным кодом. Время выработки суммы  $n$ -разрядных чисел:  $t = n \cdot t_c$ , где  $t_c$  – длительность суммирования в одноразрядном сумматоре.

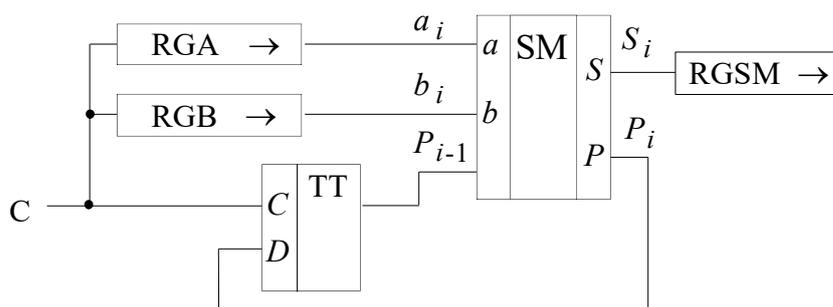


Рисунок 6 – Структура последовательного многоразрядного сумматора

Наибольшее распространение получили параллельные многоразрядные сумматоры, которые строятся с использованием необходимого числа одноразрядных сумматоров, на которые входные переменные подаются параллельным кодом. По способу организации межразрядных переносов параллельные сумматоры делятся на схемы с последовательным и параллельным переносом и с групповой структурой. Условное графическое обозначение комбинационного параллельного сумматора приведено на рисунке 7.

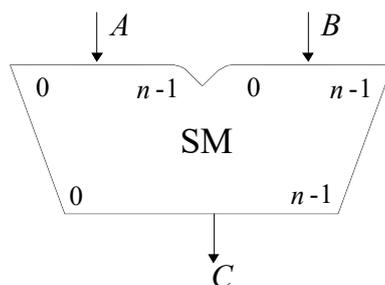


Рисунок 7 – Параллельный комбинационный сумматор. Условное графическое обозначение

Параллельный сумматор с последовательным переносом (рисунок 8) принимает слагаемые параллельно. После этого образуются предварительные поразрядные суммы, а после появления и распространения переносов суммы принимают окончательное значение. Время сложения определяется по формуле:

$$t_c = t_{c,p} + (n-2) \cdot t_{p,p} + t_{p,s} \cong n \cdot t_{p,p}, \quad (9)$$

где  $t_{c,p}$  – время выработки переноса в младшем разряде;  $t_{p,p}$  – время распространения переноса через разряд;  $t_{p,s}$  – время выработки суммы в старшем разряде после поступления на его вход переноса от предпоследнего разряда.

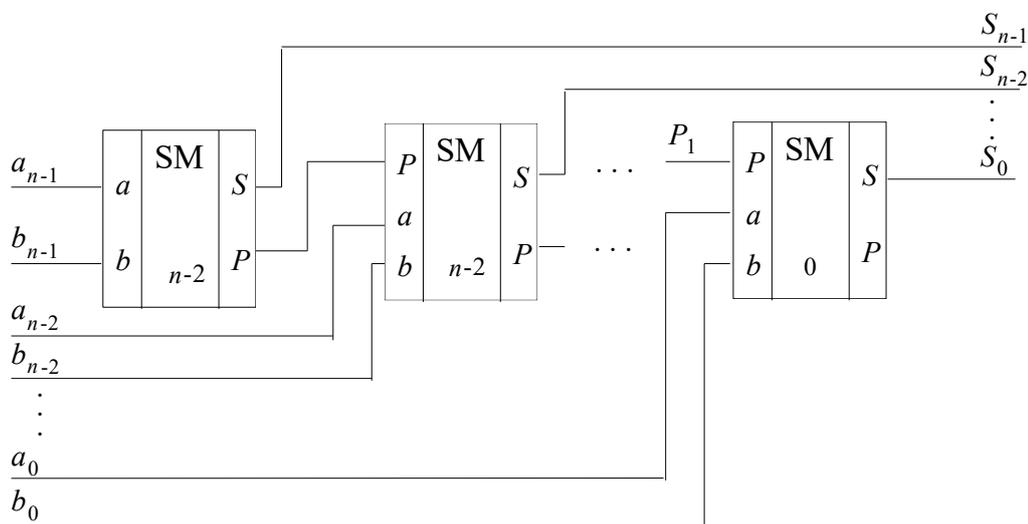


Рисунок 8 – Схема параллельного сумматора с последовательным переносом

Основную часть времени суммирования занимает распространение переноса через разряды сумматора, поэтому минимальность  $t_{p,p}$  имеет большое значение. Рассмотренные выше схемы одноразрядных сумматоров

(рисунки 4 и 5) имеют время  $t_{p,p} = 2\tau$ , где  $\tau$  – время задержки логического элемента.

### Проектирование сумматора «по модулю два»

Это устройство с двумя входами ( $a$  и  $b$ ), на выходе у которого сигнал "1" появляется только в том случае, когда на входах действуют противоположные сигналы, т. е. "0" и "1".

Таблица истинности и логическое уравнение имеют вид:

$a$	$b$	$y$
0	0	0
0	1	1
1	0	1
1	1	0

$$y = \bar{a}b \vee a\bar{b}$$

Название «по модулю два» этот сумматор получил потому, что у соответствует значению младшего разряда при суммировании одноразрядных двоичных чисел  $A$  и  $B$ . Построим в базисе **И–НЕ** схему сумматора «по модулю два» (см. рисунок 9)

$$y = \bar{a}b \vee a\bar{b} = \overline{\overline{\bar{a}b} \cdot \overline{a\bar{b}}} = \overline{\overline{\bar{a}b} \cdot \overline{a\bar{b}}}$$

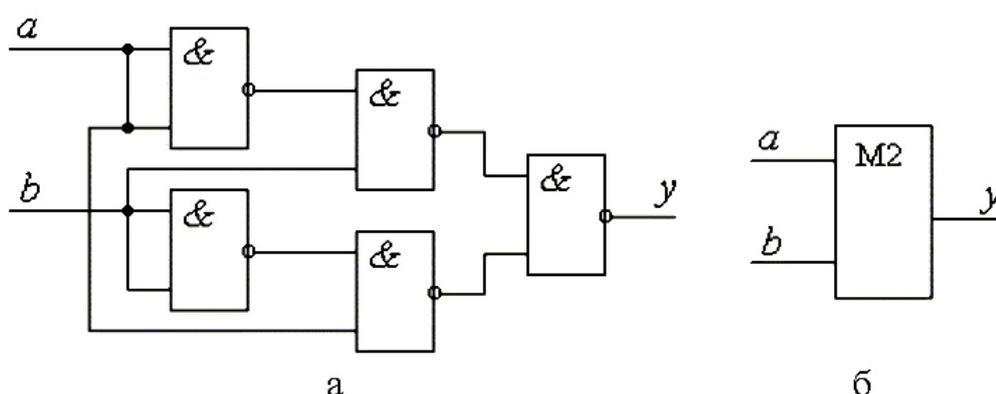


Рисунок 9 – Реализация сумматора по модулю два: а – принципиальная схема; б – функциональная схема

Сумматор не обладает памятью.

### 3 Проектирование шифраторов и дешифраторов

Шифрация и дешифрация (сжатие данных и обратное преобразование) являются основными видами преобразования информации.

#### Дешифраторы

Дешифратор – это КФУ, имеющий  $n$  входов и  $2^n$  выходов, осуществляющий преобразование входного двоичного  $n$ -разрядного кода в сигнал на одном из выходов. По каждому выходу дешифратора реализуется конstituента единицы  $K_i^1$  (или ее отрицание).

Различают полные и неполные дешифраторы. Число выходов полного дешифратора  $N_{вых} = 2^n$ , неполного -  $N_{вых} < 2^n$ . Условное графическое обозначение полного дешифратора на два входа и его таблица истинности приведены на рисунок 10, а, б.

$E$  – вход, на который подается сигнал, разрешающий дешифрирование.

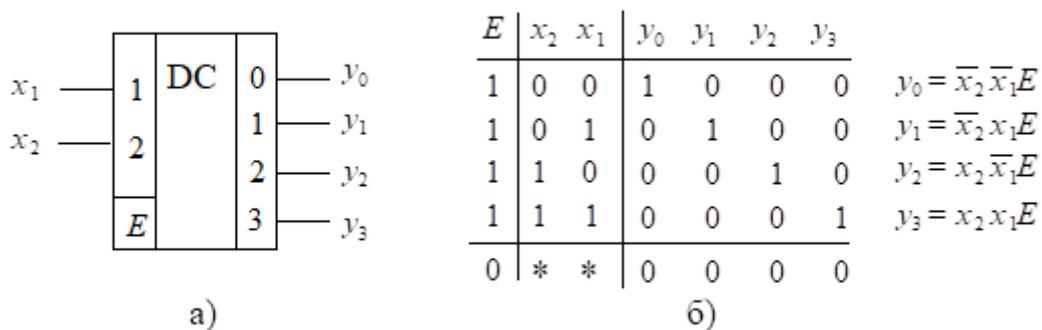


Рисунок 10 – Дешифратор: а) – условное графическое изображение; б) – таблица истинности

Аналитическое описание дешифратора в форме СДНФ:

$$Y_i = \alpha_i \cdot E, \quad i = 0, 1, \dots, n, \quad (10)$$

где  $\alpha_i$  -  $i$ -ый минтерм  $n$  входных переменных,

$E$  - сигнал, разрешающий дешифрирование.

Для неполных дешифраторов имеются безразличные (неопределенные) входные наборы, которые можно использовать при минимизации выходных функций. Например, при проектировании дешифратора «1 из 10» безразличными являются входные наборы отмеченные знаком \* на карте Карно (рисунок 10, а, б).

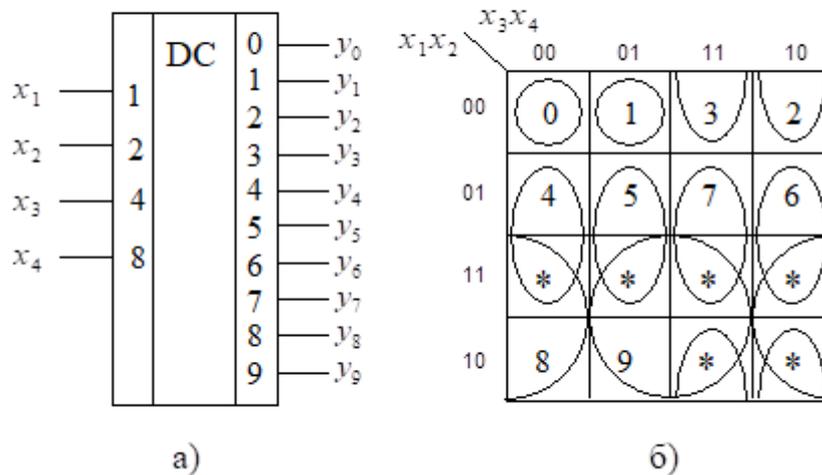


Рисунок 10 – Неполный дешифратор «1 из 10»: а) – условное графическое изображение; б) – минимизация функций

После совместной минимизации на рабочих и соседних к ним неопределенных наборах получим следующие логические уравнения:

$$\begin{aligned}
 y_0 &= \bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4, & y_4 &= x_2\bar{x}_3\bar{x}_4, & y_8 &= x_1\bar{x}_4, \\
 y_1 &= \bar{x}_1\bar{x}_2\bar{x}_3x_4, & y_5 &= x_2\bar{x}_3x_4, & y_9 &= x_1x_4, \\
 y_2 &= \bar{x}_2x_3\bar{x}_4, & y_6 &= x_2x_3\bar{x}_4, \\
 y_3 &= \bar{x}_2x_3x_4, & y_7 &= x_2x_3x_4,
 \end{aligned}
 \tag{11}$$

Если при проектировании дешифратора не проводить совместной минимизации рабочих и неопределенных наборов, то схемная реализация будет иметь большую стоимость и топологические размеры будут значительно большими.

В схемах ЭВМ дешифраторы устанавливаются на выходах регистров или счетчиков и служат для преобразования кода слова, находящегося в регистре (в счетчике) в управляющий сигнал на одном из выходов дешифратора.

### Использование дешифраторов

Рассмотрим некоторые области применения дешифраторов в разрабатываемых устройствах.

1. Для реализации логических функций.

Например, реализовать на основе дешифратора логическую функцию вида:

$$Y = a\bar{b} \vee \bar{a}\bar{b}c \vee b\bar{c}.$$

*Решение.* Исходная функция задана в форме ДНФ. Преобразуем ее в СДНФ:

$$Y = a\bar{b}(c \vee \bar{c}) \vee \bar{a}\bar{b}c \vee b\bar{c}(a \vee \bar{a}) = a\bar{b}c \vee a\bar{b}\bar{c} \vee \bar{a}\bar{b}c \vee a\bar{b}\bar{c} \vee \bar{a}b\bar{c}.$$

Полученное выражение реализуется логической схемой, представленной на рисунке 11.

Для получения схемы достаточно определить выходы дешифратора, соответствующие входящим в функцию конституентам единицы и соединить их с входами дизъюнктора. Если на входы дешифратора будут поданы входные переменные, то на выходе дизъюнктора сформируется значение функции.

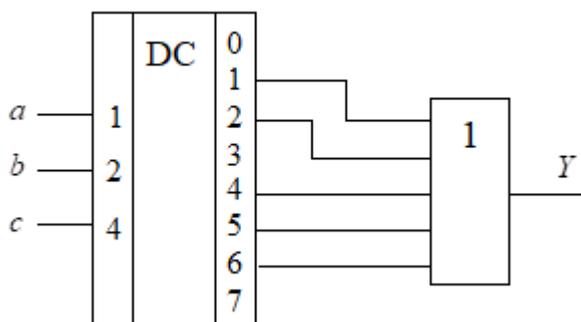


Рисунок 11 – Реализация логической функции  $Y = a\bar{b}c \vee a\bar{b}\bar{c} \vee \bar{a}\bar{b}c \vee a\bar{b}\bar{c} \vee \bar{a}b\bar{c}$  на основе дешифратора

2. В качестве преобразователя двоично-десятичного кода в семисегментный код в устройствах визуальной индикации десятичных цифр на световом табло (см. рисунок 12).

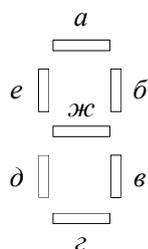


Рисунок 12 – Индикации десятичных цифр на световом табло

Проектирование таких преобразователей осуществляется на основе таблицы истинности (см. таблицу 1) дешифратора «1 из 10» и системы уравнений:

$$a = \overline{y_1 \vee y_4}, \quad z = \overline{y_1 \vee y_4 \vee y_7}, \quad e = \overline{y_1 \vee y_2 \vee y_3 \vee y_7},$$

$$\bar{b} = \overline{y_5 \vee y_6}, \quad \bar{d} = \overline{y_0 \vee y_2 \vee y_6 \vee y_8}, \quad \bar{ж} = \overline{y_0 \vee y_1 \vee y_7},$$

$$в = y_2,$$

где  $y_0 \div y_9$  – выходные шины дешифратора «1 из 10» (см. рисунок 13).

Таблица 1

Дес. число	ВХОДЫ				ДС	ВЫХОДЫ						
	$x_1$	$x_2$	$x_3$	$x_4$		$a$	$\bar{b}$	$в$	$z$	$\bar{d}$	$e$	$\bar{ж}$
0	0	0	0	0	$y_0$	1	1	1	1	1	1	0
1	0	0	0	1	$y_1$	0	1	1	0	0	0	0
2	0	0	1	0	$y_2$	1	1	0	1	1	0	1
3	0	0	1	1	$y_3$	1	1	1	1	0	0	1
4	0	1	0	0	$y_4$	0	1	1	0	0	1	1
5	0	1	0	1	$y_5$	1	0	1	1	0	1	1
6	0	1	1	0	$y_6$	1	0	1	1	1	1	1
7	0	1	1	1	$y_7$	1	1	1	0	0	0	0
8	1	0	0	0	$y_8$	1	1	1	1	1	1	1
9	1	0	0	1	$y_9$	1	1	1	1	0	1	1

Схема дешифратора-преобразователя приведена на рисунке 13.

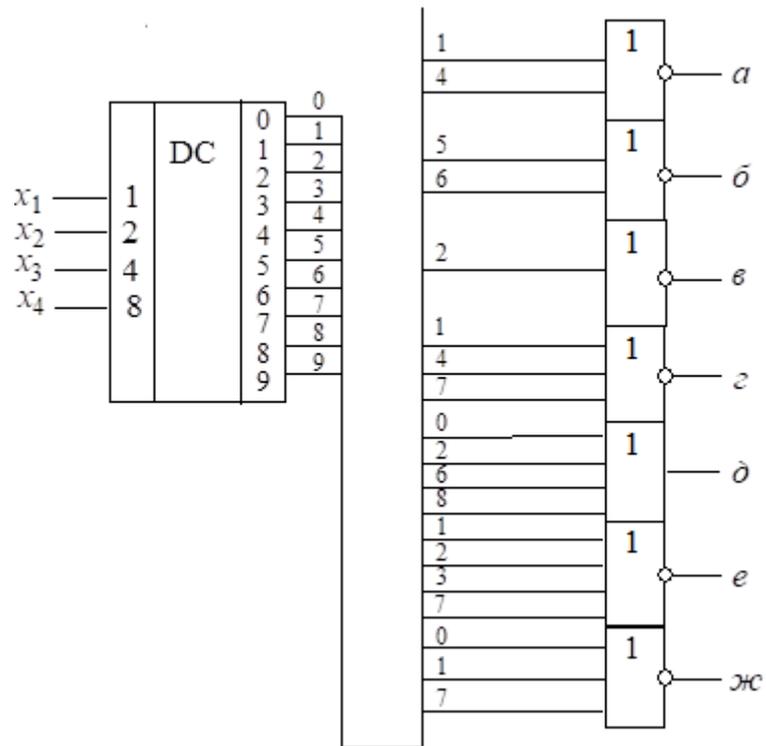


Рисунок 13 – Схема дешифратора-преобразователя двоично-десятичного кода в семисегментный код в устройствах визуальной индикации десятичных цифр на световом табло

3. Для преобразования кодов. В качестве примера приведена таблица истинности преобразователя (см. таблицу 2) двоично-десятичного кода 8-4-2-1 в код 2-4-2-1:

Таблица 2

Десятичное число	Код				DC	Код			
	8	4	2	1		2	4	2	1
	$x_1$	$x_2$	$x_3$	$x_4$		$A$	$B$	$C$	$D$
0	0	0	0	0	$y_0$	0	0	0	0
1	0	0	0	1	$y_1$	0	0	0	1
2	0	0	1	0	$y_2$	0	0	1	0
3	0	0	1	1	$y_3$	0	0	1	1
4	0	1	0	0	$y_4$	0	1	0	0
5	0	1	0	1	$y_5$	1	0	1	1
6	0	1	1	0	$y_6$	1	1	0	0
7	0	1	1	1	$y_7$	1	1	0	1
8	1	0	0	0	$y_8$	1	1	1	0
9	1	0	0	1	$y_9$	1	1	1	1

Из таблицы истинности записываются уравнения выходных функций для реализации преобразователя (12)

$$\begin{aligned}
 A &= y_5 \vee y_6 \vee y_7 \vee y_8 \vee y_9, \\
 B &= y_4 \vee y_6 \vee y_7 \vee y_8 \vee y_9, \\
 C &= y_2 \vee y_3 \vee y_5 \vee y_8 \vee y_9, \\
 D &= y_1 \vee y_3 \vee y_5 \vee y_7 \vee y_9.
 \end{aligned}
 \tag{12}$$

## Шифраторы

Шифратор – это КФУ, преобразующий сигнал на одном из  $2^n$  входов в  $n$ -разрядный двоичный код на выходах. Если число входов меньше, чем  $2^n$ , то шифратор неполный. Одно из основных применений шифратора - ввод данных с клавиатуры, при котором нажатие на клавишу с десятичной цифрой должно приводить к передаче в устройство этой цифры в двоичном коде. Данная функция реализуется неполным шифратором «10-4» (см. рисунок 14).

$$\begin{cases}
 y_4 = x_1 \vee x_3 \vee x_5 \vee x_7 \vee x_9, \\
 y_3 = x_2 \vee x_3 \vee x_6 \vee x_7, \\
 y_2 = x_4 \vee x_5 \vee x_6 \vee x_7, \\
 y_1 = x_8 \vee x_9.
 \end{cases}
 \tag{13}$$

Таблица 3

Дес. число	Входы										Выходы			
	$x_0$	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	$x_9$	$y_1$	$y_2$	$y_3$	$y_4$
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0	0	0	0	1
2	0	0	1	0	0	0	0	0	0	0	0	0	1	0
3	0	0	0	1	0	0	0	0	0	0	0	0	1	1
4	0	0	0	0	1	0	0	0	0	0	0	1	0	0
5	0	0	0	0	0	1	0	0	0	0	0	1	0	1
6	0	0	0	0	0	0	1	0	0	0	0	1	1	0
7	0	0	0	0	0	0	0	1	0	0	0	1	1	1
8	0	0	0	0	0	0	0	0	1	0	1	0	0	0
9	0	0	0	0	0	0	0	0	0	1	1	0	0	1

Из таблицы истинности (см. таблицу 3) получаем логические уравнения для шифратора «10-4»:

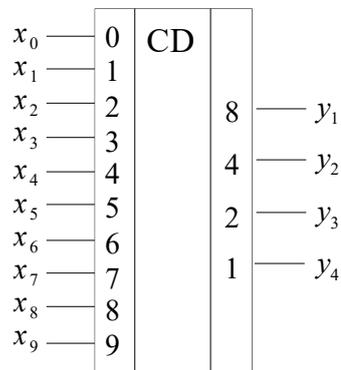


Рисунок 14 – Шифратор «10-4». Условное графическое обозначение

## 4 Проектирование мультиплексоров и демультиплексоров

### Мультиплексоры

Мультиплексор – это КФУ, служащий для последовательного опроса состояний большого числа переменных и передачи их на единственный выход.

Входы мультиплексора разделяются на информационные (входы данных), и адресные (управляющие).

Код (адрес), поступающий на управляющие входы, определяет информационный вход, сигнал с которого передается на выход. Чаще всего используют мультиплексоры на 4, 8, 16 входов. Условное графическое изображение мультиплексора на четыре входа («4-1») и его таблица истинности приведены на рисунке 15, а и б, где  $E$  – сигнал, разрешающий мультиплексирование;  $x_0, x_1, x_2, x_3$  – информационные входы;  $A_2, A_1$  – управляющие входы;  $Y$  – выход.

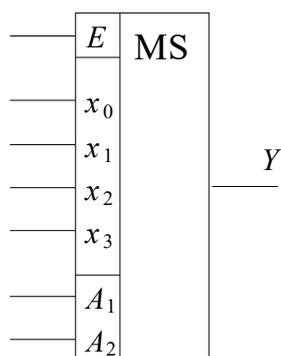
Из таблицы истинности (см. рисунок 15, б) получим СДНФ функции  $Y$ :

$$Y = x_0 \bar{A}_2 \bar{A}_1 E \vee x_1 \bar{A}_2 A_1 E \vee x_2 A_2 \bar{A}_1 E \vee x_3 A_2 A_1 E. \quad (14)$$

СДНФ выходной функции мультиплексора на  $2^n$  входов имеет вид:

$$Y = \bigvee_i x_i \alpha_i E, \quad i=0, 1, \dots, 2^n - 1, \quad (15)$$

где  $\alpha_i$  – минтерм (конституента 1), соответствующий  $i$ -му адресному набору. Данная функция далее может быть реализована в заданном базисе элементов.



а)

$E$	$A_1$	$A_2$	$Y$
1	0	0	$x_0$
1	0	1	$x_1$
1	1	0	$x_2$
1	1	1	$x_3$
0	*	*	0

б)

Рисунок 15 – Мультиплексор «4-1»

Мультиплексор можно рассматривать как преобразователь параллельной информации в последовательную. Мультиплексор на большое число входов, как правило, приходится строить из мультиплексоров меньшей размерности.

## Применение мультиплексоров

**Пример 1:** реализовать на мультиплексоре типа «8-1» логическую функцию трех переменных:

$$Y = \vee(1, 2, 6, 7) = \bar{a}\bar{b}c \vee \bar{a}b\bar{c} \vee ab\bar{c} \vee abc.$$

*Решение.* Функция задана в СДНФ. Используется мультиплексор типа «8-1». На адресные входы подаются входные переменные, а информационные входы, соответствующие входящим в функцию конституентам единицы соединяются с шинами питания (1), остальные информационные входы соединяются с шинами земли (0). На выходе мультиплексора формируется значение функции (см. рисунок 16).

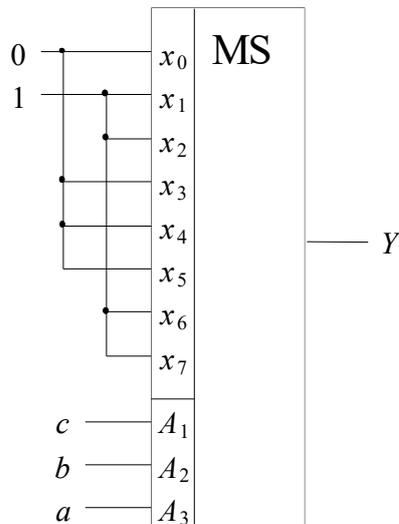


Рисунок 16 – Реализация функции  $Y = \vee(1, 2, 6, 7)$  на мультиплексоре типа «8-1»

**Пример 2:** реализовать на мультиплексоре типа «8-1» логическую функцию четырех переменных:

$$Y(a, b, c, d) = \bar{a}\bar{b}c \vee \bar{a}bcd \vee ab\bar{c} \vee abc\bar{d}.$$

*Решение.* В качестве управляющих сигналов используются переменные  $a$ ,  $b$ , и  $c$ , которые подаются на адресные входы мультиплексора. На информационные входы поступают переменные  $d$ ,  $\bar{d}$ , 0, 1:

$$\begin{aligned} x_0 &= 0, & x_3 &= d, & x_6 &= 0, \\ x_1 &= 1, & x_4 &= 0, & x_7 &= \bar{d}, \\ x_2 &= 0, & x_5 &= 1. \end{aligned}$$

**Пример 3:** реализовать с помощью мультиплексора типа «4-1» функцию трех переменных  $Y = f(a, b, c)$ :

$$Y = \vee(0, 2, 3, 8).$$

Сначала выбирается любое сочетание двух переменных  $ab$ ,  $ac$ ,  $bc$ , которые являются управляющими и подаются на адресные входы мультиплексора. На информационные входы в этом случае могут быть поданы четыре функции одной (третьей) переменной. На рисунке 17 показано соответствие информационных входов мультиплексора  $x_0, x_1, x_2, x_3$  определенным адресным (управляющим) входам мультиплексора «4-1». Например, если в качестве управляющих применить сигналы  $a$  и  $b$ , то входу  $x_0$  будут соответствовать две клетки карты Карно, для которых  $a = b = 0$ ; входу  $x_1 - a = 0, b = 1$ ;  $x_2 - a=1, b=0$ ;  $x_3 - a = b = 1$  (см. рисунок 17, а). Таким образом, карта Карно на три переменные разбивается как бы на четыре двухклеточные карты на одну переменную. Затем минимизируется набор из четырех функций одной переменной и получаются необходимые значения сигналов на информационных входах мультиплексора для реализации заданной логической функции.

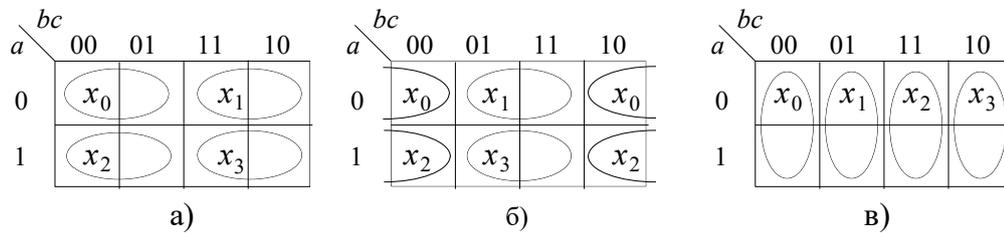


Рисунок 17 – Соответствие информационных входов мультиплексора «4-1» управляющим сигналам  $a$  и  $b$  (а),  $a$  и  $c$  (б),  $b$  и  $c$  (в)

*Решение.* Выберем в качестве управляющих сигналов переменные  $a$  и  $b$ . Функция  $Y = \vee (0, 2, 3, 6)$  представлена на карте Карно (см. рисунок 18, а). Минимизируем набор из четырех функций от одной переменной  $c$ , соответствующих каждому информационному входу ( $x_0, x_1, x_2, x_3$ ). Обе клетки, соответствующие  $x_1$ , помечены единицей и, следовательно на вход  $x_1$  подается сигнал  $c \vee \bar{c} = 1$ . Оставшиеся входные функции получаем с помощью карты Карно (см. рисунок 18, а) :  $x_0 = \bar{c}$  ;  $x_2 = 0$ ;  $x_3 = \bar{c}$  .

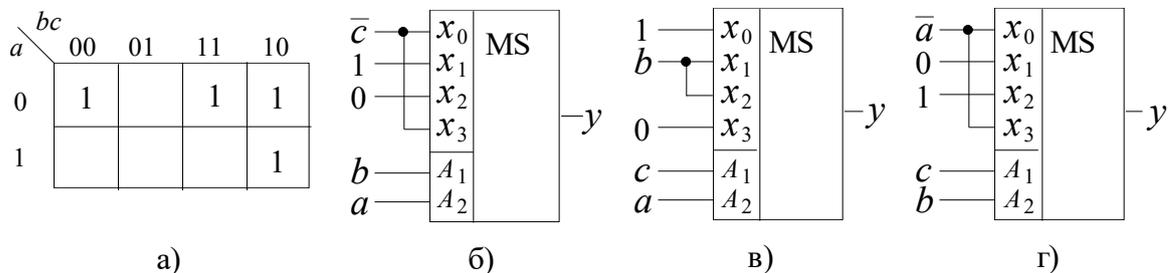


Рисунок 18 – Применение мультиплексора для реализации функции  $Y = \vee (0, 2, 3, 6)$

Подавая на входы мультиплексора найденные значения  $x_i$ , реализуем заданную функцию (см. рисунок 18, б). Если в качестве управляющих выбрать сигналы  $a$  и  $c$  или  $b$  и  $c$ , то получаются новые реализации (см. рисунок 18, в, г).

**Пример 4:** реализовать на мультиплексоре типа «4-1» логическую функцию четырех переменных:

$$Y(a, b, c, d) = \vee (0, 1, 6, 7, 9, 10, 11, 12, 13).$$

В качестве управляющих сигналов принять переменные  $a$  и  $b$ .

*Решение.* Покажем на карте Карно для функции четырех переменных ( $a, b, c, d$ ) расположение информационных сигналов ( $x_0, x_1, x_2, x_3$ ) мультиплексора «4-1» (см. рисунок 19, а). Для каждого из четырех  $x_i$  получим свою четырехклеточную карту Карно для двух переменных  $c$  и  $d$ . Например,  $x_0$  соответствуют четыре клетки с координатами  $a = b = 0$ ;  $x_1 - a = 0, b = 1$ ;  $x_2 - a = 1, b = 0$ ;  $x_3 - a = b = 1$  (см. рисунок 19, а).

Отобразим на карте Карно заданную функцию (см. рисунок 19, б) и проведем для каждого  $x_i$  минимизацию. Получим  $x_0 = \bar{c}$ ,  $x_1 = c$ ,  $x_2 = c \vee d$ ,  $x_3 = \bar{c}$ . Это значит, что для реализации заданной функции на информационные входы мультиплексора «4-1» необходимо подать соответствующие значения переменных  $c$  и  $d$  (см. рисунок 19, в).

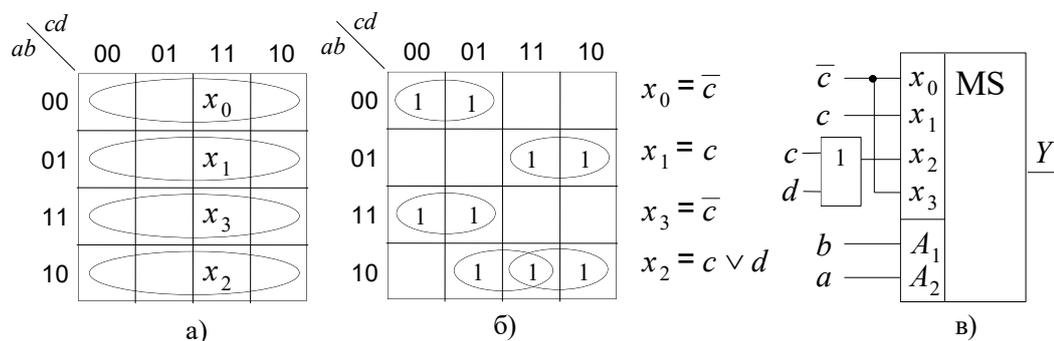


Рисунок 19 – Реализация функции  $Y(a, b, c, d) = \vee (0, 1, 6, 7, 9, 10, 11, 12, 13)$  на мультиплексоре «4-1»

## Демультимплексоры

Демультимплексор – это КФУ, выполняющий распределение входного сигнала  $X$  в соответствии с адресом на одну из  $N$  выходных шин. Условное графическое обозначение демультимплексора с четырьмя выходными шинами («1-4») приведено на рисунке 20.

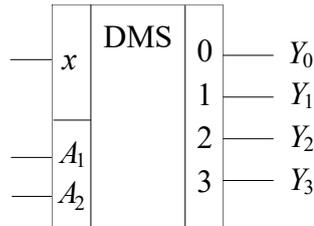


Рисунок 20 – Условное графическое обозначение демультиплексора «1-4»

Функционирование демультиплексора «1-4» описывается следующими логическими уравнениями:

$$\begin{aligned} Y_0 &= x\bar{A}_2\bar{A}_1, & Y_2 &= xA_2\bar{A}_1, \\ Y_1 &= x\bar{A}_2A_1, & Y_3 &= xA_2A_1. \end{aligned} \quad (16)$$

Для демультиплексора с числом выходных шин  $N=2^n$ , где  $n$  – число адресных входов, можно записать:

$$Y_i = x\alpha_i, \quad i=0, 1, \dots, 2^n - 1, \quad (17)$$

где  $\alpha_i$  – минтерм (конституента 1), соответствующий  $i$ -му адресному набору.

Демультиплексоры широко используются для преобразования последовательного кода в параллельный.

## 5 Проектирование компараторов

Компаратор – это КФУ, предназначенный для сравнения кодов двух чисел ( $A, B$ ) по различным признакам: равно ( $=$ ), не равно ( $\neq$ ), больше ( $>$ ), меньше ( $<$ ), больше или равно ( $\geq$ ), меньше или равно ( $\leq$ ). Основные отношения равно и больше. Остальные можно получить через основные.

Выходные функции компаратора определяются следующими выражениями:

$$\begin{aligned} Y_{A \neq B} &= \bar{Y}_{A=B}, & Y_{A \geq B} &= Y_{A=B} \vee Y_{A > B} = \bar{Y}_{A < B}, \\ Y_{A < B} &= \bar{Y}_{A > B}, & Y_{A \leq B} &= Y_{A=B} \vee Y_{A < B} = \bar{Y}_{A > B}. \end{aligned} \quad (18)$$

Схемы сравнения на равенство  $A = B$ , где  $A = a_1 a_2 \dots a_n$  и  $B = b_1 b_2 \dots b_n$  строятся на основе поразрядных операций над одноименными разрядами слов. Значения  $A$  и  $B$  равны, если одновременно равны их одноименные разряды  $a_i, b_i, i=1, 2, \dots, n$ . Из таблицы истинности (см. таблицу 4) следует признак равенства –  $r_i$  и признак неравенства –  $\bar{r}_i$  одноименных разрядов:

$$\begin{aligned} r_i &= \bar{a}_i\bar{b}_i \vee a_i b_i, \\ \bar{r}_i &= \bar{a}_i b_i \vee a_i \bar{b}_i, \end{aligned} \quad i=1, 2, \dots, n. \quad (19)$$

Таблица 4

$a_i$	$b_i$	$\bar{r}_i$	$r_{i=}$	$r_{i>}$	$r_{i<}$
0	0	0	1	0	0
0	1	1	0	0	1
1	0	1	0	1	0
1	1	0	1	0	0

Признак равенства двух  $n$ -разрядных чисел вычисляется как конъюнкция

$$Y_{A=B} = r_1 \cdot r_2 \cdot \dots \cdot r_n = \big\&_{i=1}^n (\bar{a}_i \bar{b}_i \vee a_i b_i) . \quad (20)$$

Аналогично признак неравенства ( $Y_{A \neq B}$ ) вычисляется как конъюнкция

$$Y_{A \neq B} = \big\&_{i=1}^n \bar{r}_i = \big\&_{i=1}^n (\bar{a}_i b_i \vee a_i \bar{b}_i) . \quad (21)$$

В ряде случаев требуется построить компаратор не используя инверсии аргументов. С этой целью выражения (3.20) преобразуются к виду:

$$r_i = \overline{\bar{a}_i b_i \vee a_i \bar{b}_i} = \overline{b_i (\bar{a}_i \vee \bar{b}_i) \vee a_i (\bar{a}_i \vee \bar{b}_i)} = \overline{b_i \bar{a}_i \bar{b}_i \cdot a_i \bar{a}_i \bar{b}_i} , \quad (22)$$

$$\bar{r}_i = b_i \bar{a}_i \bar{b}_i \vee a_i \bar{a}_i \bar{b}_i . \quad (23)$$

Схемы сравнения на больше-меньше также строятся на основе поразрядных операций над одноименными разрядами чисел. Из таблицы истинности можно записать:

$$\begin{aligned} r_{i>} &= a_i \bar{b}_i, \\ r_{i<} &= \bar{a}_i b_i, \end{aligned} \quad \text{где, } i=1, 2, \dots, n.$$

Сравнение выполняется, начиная со старших разрядов. Если  $a_1 > b_1$ , то независимо от младших разрядов  $A > B$  и  $Y_{A>B} = 1$ . Если старшие разряды равны ( $r_1=1$ ), следует сравнивать младшие, применив то же условие и т.д. Таким образом,

$$\begin{aligned} Y_{A>B} &= a_1 \bar{b}_1 \vee r_1 \cdot a_2 \bar{b}_2 \vee r_1 \cdot r_2 \cdot a_3 \bar{b}_3 \vee \dots \vee r_1 \cdot r_2 \cdot \dots \cdot r_{n-1} \cdot a_n \bar{b}_n = \\ &= \bigvee_{i=1}^n a_i \bar{b}_i \cdot R_{i-1} , \quad i=1, 2, \dots, n , \end{aligned} \quad (24)$$

где  $R_{i-1} = \big\&_{i=1}^{i-1} r_i$ ,  $R_0 = 1$ .

Аналогично можно получить выражение для  $Y_{A<B}$ :

$$Y_{A<B} = \bigvee_{i=1}^n \bar{a}_i b_i \cdot R_{i-1}, \quad i=1, 2, \dots, n, \quad R_{i-1} = \bigwedge_{i=1}^{i-1} r_i, \quad R_0 = 1. \quad (25)$$

Заметим, что сравнение можно выполнить с помощью сумматора, вычислив разность  $A-B=A+\lceil B+1$ , где  $(\lceil B+1)$  - дополнительный код значения  $B$ , используемый для замены операции вычитания на сложение. Анализ полученной разности позволяет сделать вывод об отношении чисел  $A$  и  $B$ .

Например, в общем случае компаратор параллельных кодов двух  $m$ -разрядных двоичных чисел представляет собой комбинационную схему с  $2m$  входами и тремя выходами (“равно”, “больше”, “меньше”). При поступлении на входы кодов двух сравниваемых чисел сигнал логической единицы появляется только на одном из выходов. В некоторых случаях компаратор может иметь менее трех выходов.

Одноразрядный компаратор имеет два входа на которые одновременно поступают одноразрядные двоичные числа  $x_1$  и  $x_2$ , и три выхода ( $=, >, <$ ).

Из таблицы истинности логические уравнения компаратора при сравнении  $x_1$  с  $x_2$  получаются в виде

$x_1$	$x_2$	$y^=$	$y^>$	$y^<$
0	0	1	0	0
0	1	0	0	1
1	0	0	1	0
1	1	1	0	0

$$y^= = x_1 x_2 \vee \bar{x}_1 \bar{x}_2,$$

$$y^> = x_1 \bar{x}_2,$$

$$y^< = \bar{x}_1 x_2.$$

Реализация такого компаратора в базисе **И–НЕ** приводит к следующей схеме (см. рисунок 21).

Многоразрядные компараторы обычно выполняют на базе одноразрядных. При этом используется принцип последовательного сравнения разрядов многоразрядных чисел, начиная с их старших разрядов, так как уже на этом этапе, если  $x_{1m} > x_{2m}$ , задача может быть решена

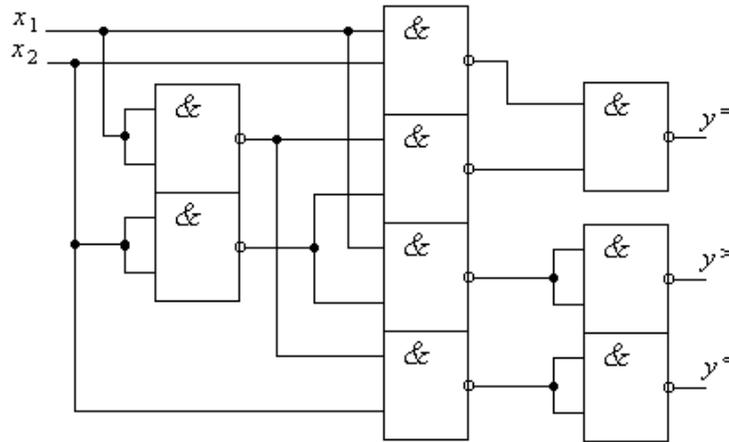


Рисунок 21 – Одноразрядный компаратор двоичных чисел

однозначно, и сравнение следующих за старшими разрядами не потребуется.

## 6 Порядок выполнения лабораторной работы

Получить вариант задания у преподавателя. Разработать КФУ, удовлетворяющее требованиям индивидуального задания.

## 7 Содержание отчета

Отчет должен оформляться на листах А4 рукописным способом и содержать следующее:

1. Титульный лист.
2. Задание на проектирование КФУ.
3. Этапы проектирования.
4. Реализованная схема КФУ.
5. Выводы по работе.

## 8 Примерные задания для реализации

Реализовать функции

$$f_1(x_1, x_2, x_3) = \bar{x}_1 \bar{x}_3 \vee x_1 x_2 \vee x_2 \bar{x}_3 \quad \text{И} \quad f_2(x_1, x_2, x_3) = (\bar{x}_1 \vee x_2)(x_1 \vee \bar{x}_3)(x_2 \vee \bar{x}_3)$$

на основе а) дешифратора ;

б) мультиплексора типа «8-1» ;

в) мультиплексора типа «4-1» .

Реализовать функцию

$$Y = f(x_1, x_2, x_3, x_4, x_5) = \bar{x}_1 x_2 \bar{x}_5 \vee x_3 \bar{x}_4 x_5 \vee \bar{x}_1 \bar{x}_3 \bar{x}_4 \vee x_2 x_4 x_5$$

на основе мультиплексора а) типа «16-1» ;

б) типа «8-1» ;

в) типа «4-1».

## 9 Контрольные вопросы

1. Задача синтеза функционального узла комбинационного типа.
2. Требования оптимальности.
3. Сформулируйте определение сумматора.
4. Классификация сумматоров.
5. Сформулируйте определение полусумматора, приведите условное графическое изображение и таблицу истинности.
6. Сформулируйте определение одноразрядного сумматора, приведите условное графическое изображение и таблицу истинности.
7. Сформулируйте определение многоразрядного сумматора, приведите условное графическое изображение.
8. Сформулируйте определение дешифратора, приведите условное графическое изображение и таблицу истинности.
9. Приведите варианты применения дешифраторов.
10. Сформулируйте определение шифратора, приведите условное графическое изображение и таблицу истинности.
11. Сформулируйте определение мультиплексора, приведите условное графическое изображение и таблицу истинности.
12. Приведите варианты применения мультиплексоров.
13. Сформулируйте определение демультиплексора, приведите условное графическое изображение и таблицу истинности.
14. Сформулируйте определение компаратора, приведите таблицу истинности.

## Список литературы

1. Моисеев, Н.Г. Теория автоматов [Электронный ресурс]: учебное пособие по курсовому проектированию / Н.Г. Моисеев; Поволжский государственный технологический университет. – Йошкар-Ола: ПГТУ, 2015. – 127 с. // Режим доступа - <http://biblioclub.ru/index.php?page=book&id=439263>
2. Акинина, Ю. С. Теория автоматов : учебное пособие / Ю. С. Акинина, С. В. Тюрин. — 2-е изд. — Москва : Ай Пи Ар Медиа, 2023. — 156 с. — ISBN 978-5-4497-1877-8. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/127573.html>