

Министерство образования и науки Республики Беларусь  
Учреждение образования  
«Белорусский государственный университет информатики и  
радиоэлектроники»

Будько А.А.

ЦИФРОВЫЕ УСТРОЙСТВА  
Технология IDL

*Учебно-методическое пособие для выполнения  
лабораторных работ*

*Рекомендовано УМО вузов Республики Беларусь по образованию  
в области информатики и радиоэлектроники в качестве  
учебно-методического пособия для студентов по специальностям  
«Радиотехника», «Радиоэлектронные системы», «Радиоэлектронная защита  
информации», «Радиоинформатика»*

Минск БГУИР 2010

УДК 004.3\*144(076)  
ББК 32.973.26-04я7  
Б90

А в т о р  
Будько А.А.

Р е ц е н з е н т ы :

Кафедра связи учреждения образования «Военная академия Республики Беларусь» (протокол № 3 от 27.09.2010г.)

Заведующий кафедры радиосвязи и радиовещания УО «Высший государственный колледж связи», кандидат технических наук, доцент А.И.Корзун.

**Цифровые устройства. Технология IDL:** учебно-методическое пособие для студентов радиотехнических специальностей / Будько А.А.. – Минск: БГУИР, 2010. – 134 с.

Учебно-методическое пособие является руководством к выполнению лабораторных работ с физическим моделированием основных цифровых устройств на лабораторной установке IDL 800. Пособие содержит теоретические сведения, рекомендации и задания по синтезу, физическому моделированию и исследованию типовых и оригинальных цифровых устройств и также может быть использовано при проведении практических занятий.

УДК 004.3\*144(076)  
ББК 32.973.26-04я7

© Будько А.А.

© УО Белорусский государственный университет информатики и радиоэлектроники, 2010

## СОДЕРЖАНИЕ

1. Лабораторная работа № 1. Исследование логических элементов.....	4
2. Лабораторная работа № 2. Синтез комбинационных устройств в заданном базисе логических элементов.....	24
3. Лабораторная работа № 3. Исследование сумматоров, вычитателей и суммирования/вычитания в дополнительном коде.....	34
4. Лабораторная работа № 4. Исследование преобразователей кодов.....	43
5. Лабораторная работа № 5. Синтез комбинационных схем с использованием мультиплексоров и демультимплексоров.....	53
6. Лабораторная работа № 6. Триггеры.....	66
7. Лабораторная работа № 7. Регистры и их применение.....	89
8. Лабораторная работа № 8. Генераторы последовательностей.....	98
9. Лабораторная работа № 9. Исследование асинхронных счетчиков.....	104
10. Лабораторная работа № 10. Исследование синхронных счетчиков.....	117
11. Литература.....	130

## Лабораторная работа №1

### ИССЛЕДОВАНИЕ ЛОГИЧЕСКИХ ЭЛЕМЕНТОВ

#### 1. ЦЕЛЬ ЛАБОРАТОРНОЙ РАБОТЫ

1. Изучение базовых логических функций;
2. Исследование функционирования основных логических элементов.

#### 2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Все цифровые устройства построены на элементах, которые выполняют те или иные логические операции. Для анализа и синтеза цифровых устройств используется аппарат алгебры логики. В общем случае цифровые устройства разделяются на два типа:

- комбинационные, и
- последовательностные.

В комбинационных устройствах выходной сигнал в любой момент времени зависит только от значений входных сигналов в тот же момент времени.

В последовательностных устройствах выходной сигнал в любой момент времени зависит как от значений входных сигналов в данный момент времени, так и от значений входных сигналов в предыдущие моменты времени. Для этого эти устройства имеют память. В последовательностных устройствах комбинационные устройства являются составной частью, поэтому комбинационные устройства изучаются первыми.

Рассмотрим комбинационное устройство, блок-схема которого показана на рис. 1.1.



Рис. 1.1. Блок-схема комбинационного устройства

Входные и выходные сигналы комбинационного устройства могут принимать только два логических значения 0 или 1, т.е.  $x_i \in 0,1$ ;  $y_i \in 0,1$ . Любая совокупность входных сигналов может быть представлена вектором

$$X = \{x_1, x_2, \dots, x_n\}$$

и называется входным набором.

Очевидно, что существует  $2^n$  различных входных наборов. Сопоставим каждому входному набору определенное значение выходного сигнала

$y = f(x_1, x_2, \dots, x_n)$ . Тогда работа комбинационной схемы (устройства) может быть описана с помощью функции, отображающей множество входных наборов в значение выходной переменной  $Y$ .

*Определение.* Функцией алгебры логики (ФАЛ)  $f(x_1, x_2, \dots, x_n)$  называется функция, дающая однозначное отображение множества векторов  $X$  в переменную  $Y$ .

Так как число различных входных наборов равно  $2^n$ , то любая ФАЛ может быть задана в виде таблицы со  $2^n$  строками, табл. 1.1.

Таблица 1.1

$x_1$	$x_2$	...	$x_{n-1}$	$x_n$	$f(x_1, x_2, \dots, x_n)$
0	0	...	0	0	$\alpha_1$
0	0	...	0	1	$\alpha_2$
0	0	...	1	0	$\alpha_3$
...	...	...	...	...	...
1	1	...	0	0	$\alpha_{2^n-3}$
1	1	...	0	1	$\alpha_{2^n-2}$
1	1	...	1	0	$\alpha_{2^n-1}$
1	1	...	1	1	$\alpha_{2^n}$

В левой части таблицы перечислены все возможные входные наборы, а в правой записаны значения функции на этих наборах.

Каждая ФАЛ представляет собой двоичный набор длины  $2^n$ , а число возможных таких наборов равно  $2^{2^n}$ , поэтому справедливо следующее утверждение, что число различных функций алгебры логики, зависящих от  $n$  аргументов, конечно и равно  $2^{2^n}$ .

Рассмотрим несколько примеров. Пусть  $n = 1$ , тогда число функций алгебры логики  $2^{2^1} = 4$ . Эти функции приведены в табл. 1.2.

Таблица 1.2

$x$	$f_0$	$f_1$	$f_2$	$f_3$
0	0	0	1	1
1	0	1	0	1

Функции  $f_0$  и  $f_3$  — логические константы (константа нуля и константа единицы). Функция  $f_1$  называется функцией тождества или просто тождеством, т.к.  $f_1 = x$ , а функция  $f_2$  называется функцией отрицания или просто отрицанием,  $f_2 = \bar{x}$  (читается «не X»), функция НЕ.

Для  $n = 2$  существует  $2^{2^n} = 2^{2^2} = 16$  функций алгебры логики; они приведены в табл. 1.3.

Таблица 1.3

$x_1$	$x_2$	$f_0$	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	$f_6$	$f_7$	$f_8$	$f_9$	$f_{10}$	$f_{11}$	$f_{12}$	$f_{13}$	$f_{14}$	$f_{15}$
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
Операц. символ		•	←		←		⊕	+	↓	⊙		→		→	↑		

Рассмотрим эти функции.

Функции  $f_0(x_1, x_2)$  и  $f_{15}(x_1, x_2)$  - логические константы. Функция  $f_0(x_1, x_2) = 0 = \overline{f_{15}(x_1, x_2)}$  - константа нуль (функция нуль). Технически  $f_0$  реализуется генератором нуля, рис. 1.1а.

Функция  $f_{15}(x_1, x_2) = 1 = \overline{f_0(x_1, x_2)}$  - константа единица (функция единица). Технически реализуется генератором единицы, рис. 1.1 б.

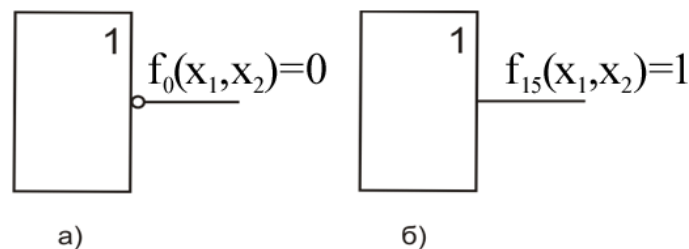


Рис. 1.1. Условное обозначение генераторов логических констант

Будем продолжать рассматривать функции в табл. 1.3 по парам, так как по отношению к любой функции вторая функция в паре является инверсной.

Функция  $f_1(x_1, x_2)$  - конъюнкция, логическое умножение, функция И,

$$f_1(x_1, x_2) = x_1 \& x_2 = x_1 \cdot x_2 = x_1 x_2 = \overline{f_{14}(x_1, x_2)}.$$

Технически реализуется логическим элементом И, как показано на рис.1.2а.

Функция  $f_{14}(x_1, x_2)$  - функция Шеффера, функция И-НЕ,

$$f_{14}(x_1, x_2) = \overline{x_1 x_2} = \overline{f_1(x_1, x_2)}.$$

Технически реализуется логическим элементом Шеффера, элементом И-НЕ, см. рис. 1.2б.

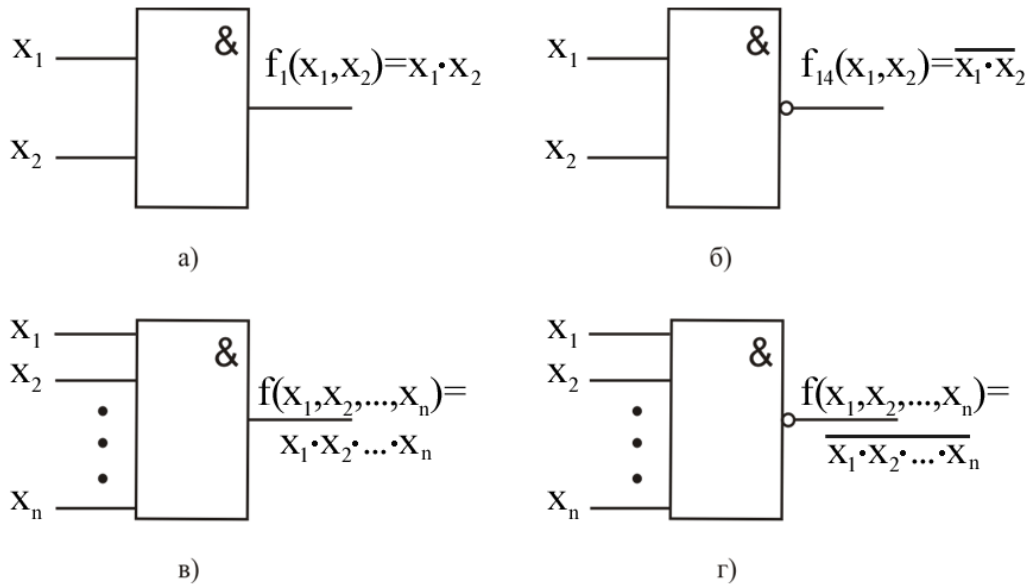


Рис.1.2. Условное обозначение логических элементов И и И-НЕ

Функции И и И-НЕ могут быть, как и соответствующие им логические элементы, с произвольным числом переменных (входов), рис. 1.2в, г.

Функция  $f_2$  - запрет 1-го аргумента,

$$f_2(x_1, x_2) = x_1 \leftarrow x_2 = x_1 \cdot \bar{x}_2 = f_{13}(x_1, x_2).$$

Технически реализуется элементом запрета, рис. 1.3а.

Функция  $f_{13}$  - импликация от 1-го аргумента ко второму,

$$f_{13}(x_1, x_2) = x_1 \rightarrow x_2 = \bar{x}_1 + x_2 = \bar{f}_2(x_1, x_2).$$

Технически реализуется импликатором, рис. 1.3б.

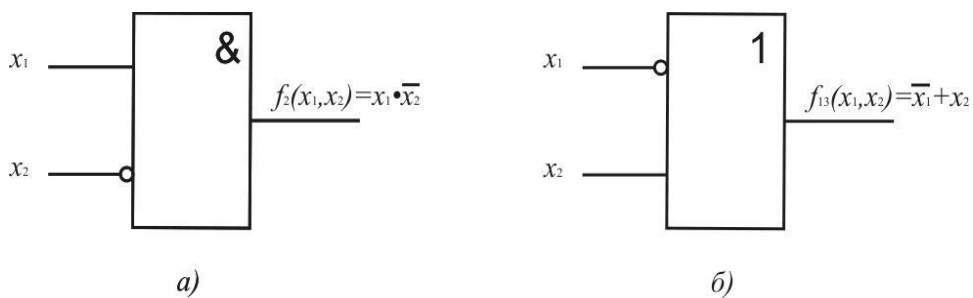


Рис. 1.3. Условное обозначение элемента запрета а) импликатора б)

Функция  $f_3$  - повторение первого аргумента (функция ДА),

$$f_3(x_1, x_2) = x_1 = f_{12}(x_1, x_2).$$

Технически реализуется повторителем, рис 1.4а.

Функция  $f_{12}$  - отрицание первого аргумента (функция НЕ),

$$f_{12}(x_1, x_2) = \bar{x}_1 = \bar{f}_3(x_1, x_2).$$

Технически реализуется инвертором, рис.1.4б.

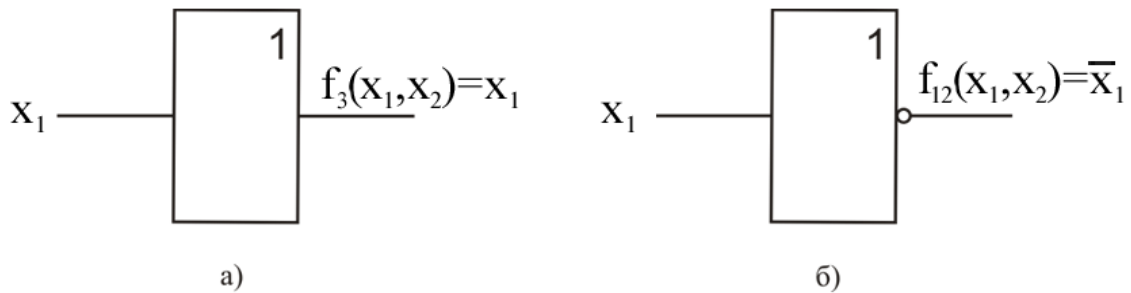


Рис. 1.4. Условные обозначения повторителя и инвертора

Функция  $f_4$  - запрет 2-го аргумента,

$$f_4(x_1, x_2) = x_2 \leftarrow x_1 = \bar{x}_1 \cdot x_2 = \bar{f}_{11}(x_1, x_2).$$

Функция  $f_{11}$  - импликация от 2-го аргумента к 1-му,

$$f_{11}(x_1, x_2) x_2 \rightarrow x_1 = x_1 + \bar{x}_2 = f_4(x_1, x_2).$$

Функция  $f_5$  - повторение 2-го аргумента,

$$f_5(x_1, x_2) = x_2 = \bar{f}_{10}(x_1, x_2).$$

Функция  $f_{10}$  - отрицание второго аргумента,

$$f_{10}(x_1, x_2) = \bar{x}_2 = \bar{f}_5(x_1, x_2).$$

Функция  $f_6$  - неравнозначность, ИСКЛЮЧАЮЩЕЕ ИЛИ,

$$f_6(x_1, x_2) = x_1 \oplus x_2 = \bar{x}_1 x_2 + x_1 \bar{x}_2 = \bar{f}_9(x_1, x_2).$$

Технически реализуется логическим элементом «ИСКЛЮЧАЮЩЕЕ ИЛИ», рис.1.5а.

Функция  $f_9$  - равнозначность, эквивалентность, «ИСКЛЮЧАЮЩЕЕ ИЛИ-НЕ»,

$$f_9(x_1, x_2) = x_1 \odot x_2 = x_1 x_2 + \bar{x}_1 \cdot \bar{x}_2 = \overline{x_1 \oplus x_2} = \bar{f}_6(x_1, x_2).$$

Технически реализуется элементом равнозначность, «ИСКЛЮЧАЮЩЕЕ ИЛИ-НЕ», рис.1.5б.

Функции неравнозначность и равнозначность могут быть, как и соответствующие им логические элементы, с произвольным числом переменных (входов), рис. 1.5в,г. Функция неравнозначность равна 1, если число аргументов равных 1 нечетно.

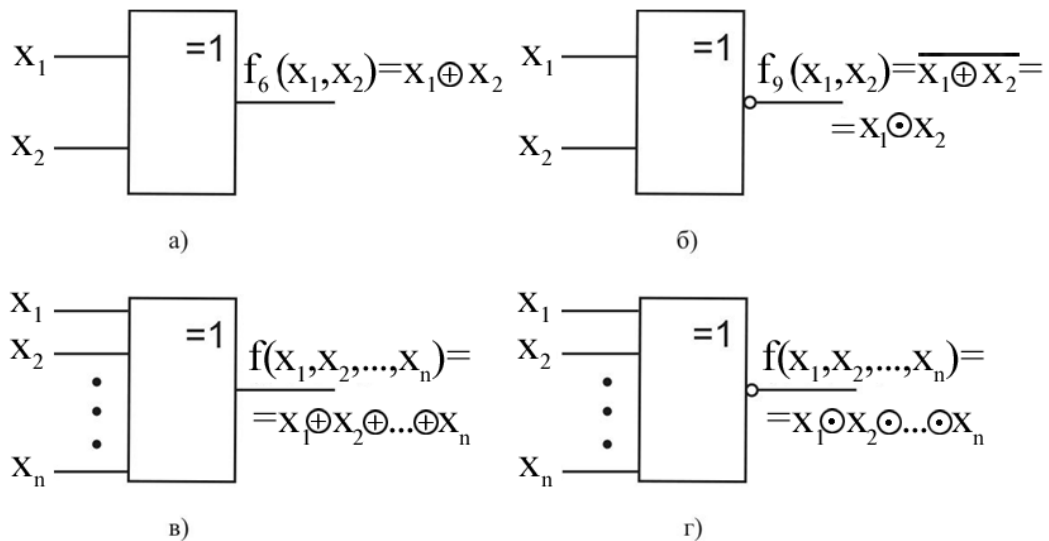


Рис. 1.5. Логические элементы, реализующие функции неравнозначность и равнозначность

Функция  $f_7$  - дизъюнкция, функция ИЛИ,

$$f_7(x_1, x_2) = x_1 + x_2 = \overline{f_8(x_1, x_2)}.$$

Технически реализуется элементом ИЛИ, рис. 1.6а.

Функция  $f_8$  - функция Пирса или функция Вебба (функция ИЛИ-НЕ),

$$f_8(x_1, x_2) = \overline{x_1 + x_2} = \overline{f_7(x_1, x_2)}.$$

Технически реализуется элементом Пирса или Вебба (элемент ИЛИ-НЕ), рис.1.6б.

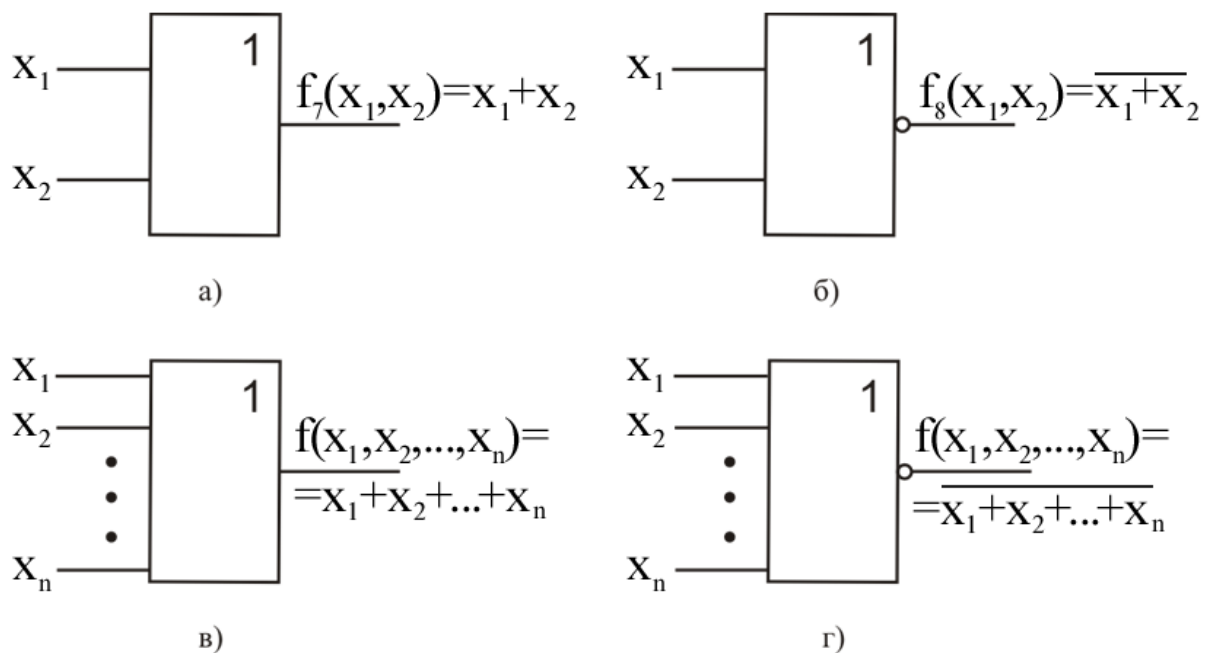


Рис.1.6. Логические элементы ИЛИ и ИЛИ-НЕ

Функции ИЛИ и ИЛИ-НЕ могут быть, как и соответствующее им логические элементы, с произвольным числом переменных (входов), рис.1.6в,г.

Значение функций двух переменных в общей теории логических функций состоит в том, что с их помощью может быть представлена любая сколько угодно сложная ФАЛ. Средством для такого представления является суперпозиция булевых функций или подстановка одних логических функций вместо аргументов в другие функции. Возможность такой подстановки обуславливается тем, что в силу определения, области значений функций и их аргументов совпадают.

Для выражения сложных логических функций достаточно использовать не все элементарные функции, а только их некоторую часть, называемую базисом или системой.

Система элементарных функций  $F_1, F_2, \dots, F_k$  называется функционально полной, если любую функцию алгебры логики можно записать в виде формулы через функции  $F_1, F_2, \dots, F_k$ .

Минимальным базисом называется такая функционально полная система,  $F_1, F_2, \dots, F_m$  для которой удаление любой одной из входящих в нее функций превращает эту систему в функционально не полную.

Примерами полных систем являются:

1.  $F_1(x_1, x_2) = x_1 \cdot x_2$ ,  $F_2(x_1, x_2) = x_1 + x_2$ ,  $F_3(x) = \bar{x}$ ;
2.  $F_1(x_1, x_2) = x_1 + x_2$ ,  $F_2(x) = \bar{x}$ ;
3.  $F_1(x_1, x_2) = x_1 \cdot x_2$ ,  $F_2(x) = \bar{x}$ ;
4.  $F(x_1, x_2) = \overline{x_1 + x_2}$ ;
5.  $F(x_1, x_2) = \overline{x_1 \cdot x_2}$ ;
6.  $F_1(x_1, x_2) = x_1 \cdot x_2$ ;  $F_2(x_1, x_2) = x_1 \oplus x_2$ ;  $F_3(x) = 1$ ;

ФАЛ можно изменять, упрощать. Для проведения таких манипуляций используются основные законы алгебры логики, правила, теоремы упрощения, которые представлены в виде табл. 1.4.

Таблица 1.4

Основные законы, правила и теоремы алгебры логики	
Для оператора (+)	Для оператора (•)
Переместительный закон (коммутативный)	
$x_1 + x_2 = x_2 + x_1$	$x_1 \cdot x_2 = x_2 \cdot x_1$
Сочетательный закон (ассоциативный)	
$(x_1 + x_2) + x_3 = x_1 + (x_2 + x_3)$	$(x_1 \cdot x_2) \cdot x_3 = x_1 (x_2 \cdot x_3)$
Распределительный закон (дистрибутивный)	
$x_1 \cdot (x_2 + x_3) = x_1 x_2 + x_1 x_3$	$x_1 + x_2 \cdot x_3 = (x_1 + x_2) \cdot (x_1 + x_3)$
Закон отрицания (закон де Моргана)	
$\overline{x_1 + x_2} = \bar{x}_1 \cdot \bar{x}_2$	$\overline{x_1 \cdot x_2} = \bar{x}_1 + \bar{x}_2$
Операции с логической 1 и 0	
$x + 0 = x$	$x \cdot 0 = 0$
$x + 1 = 1$	$x \cdot 1 = x$
Правило повторения (идемпотентности)	
$x + x + \dots + x = x$	$x \cdot x \cdot \dots \cdot x = x$
Правило дополнительности	
$x + \bar{x} = 1$	$x \cdot \bar{x} = 0$
Правило двойного отрицания	
$\overline{\bar{x}} = x$	
Теоремы упрощения	
$x_1 x_2 + x_1 \bar{x}_2 = x_1$	$(x_1 + x_2)(x_1 + \bar{x}_2) = x_1$
$x_1(x_1 + x_2) = x_1$	$x_1 + x_1 x_2 = x_1$
$x_1(\bar{x}_1 + x_2) = x_1 x_2$	$x_1 + x_1 x_2 = x_1 + x_2$

### 3. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

Оборудование и компоненты: универсальная лабораторная установка IDL-800, ИС 1533ЛА3(74ALS00) – четыре логических элемента 2И-НЕ, ИС 1533ЛЕ1(74ALS02) – четыре логических элемента 2ИЛИ-НЕ, ИС 1533ЛН1(74ALS04) – шесть логических элементов НЕ, ИС 1533ЛИ1(74ALS08) – четыре логических элемента 2И, ИС 1533ЛЛ1(74ALS32) – четыре логических элемента 2ИЛИ, ИС 1533ЛП5(74ALS86) – четыре двухвходовых логических элемента ИСКЛЮЧАЮЩЕЕ ИЛИ.

### 3.1. Исследование логических элементов И

#### Двухвходовый элемент И

1. Разместить ИС 1533ЛИ1 на наборной панели IDL-800.
2. Вывод 14 ИС соединить с источником питания +5V, а вывод 7 с общей шиной установки.
3. Собрать схему, как показано на рис. 1.7а.
4. Изменяя состояния входов  $x_1$  и  $x_2$  с помощью переключателей SW заполнить таблицу истинности логического элемента 2И, таблица 1.5а.

#### Многовходовый элемент И

1. Используя двухвходовые элементы И, собрать трехвходовый элемент И, реализующий функцию

$$f(x_1, x_2, x_3) = (x_1 \cdot x_2) \cdot x_3 = x_1 \cdot x_2 \cdot x_3,$$

как показано на рис.1.8 а.

2. Изменяя состояния входов  $x_1, x_2, x_3$ , исследовать работу схемы, заполнить таблицу истинности логического элемента 3И, табл. 1.5б.

Таблица 1.5а

Входы		Выход
$x_1$	$x_2$	$f(x_1, x_2)$
0	0	
0	1	
1	0	
1	1	

Таблица 1.5б

Входы			Выходы	
$x_1$	$x_2$	$x_3$	$x_1 \cdot x_2$	$x_1 \cdot x_2 \cdot x_3$
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

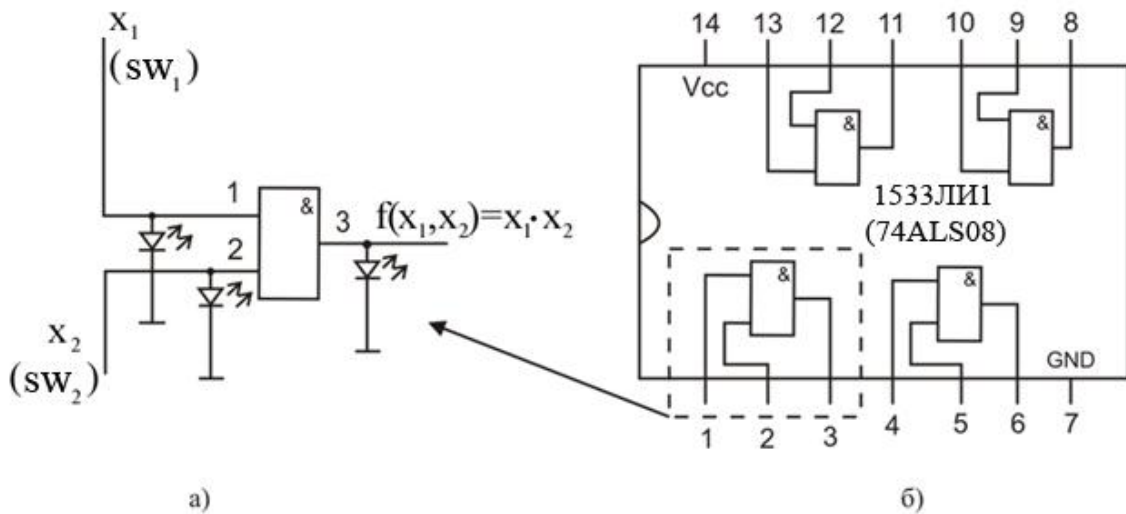


Рис 1.7

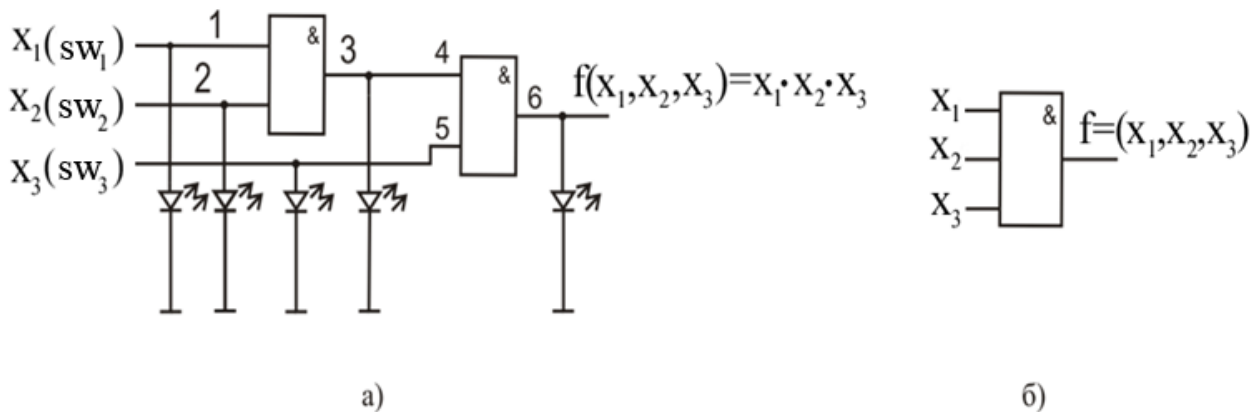


Рис 1.8

### 3.2. Исследование логических элементов ИЛИ

#### Двухвходовый элемент ИЛИ

1. Разместить ИС 1533ЛЛ1 (74ALS32) на наборной панели IDL-800.
2. Вывод 14 ИС соединить с источником питания +5V, а вывод 7 с общей шиной установки.
3. Собрать схему, как показано на рис. 1.9а.
4. Изменяя состояние входов  $x_1, x_2$ , исследовать работу двухвходового элемента ИЛИ, заполнить таблицу истинности логического элемента, табл. 1.6.

## Многовходовой элемент ИЛИ

1. Используя двухвходовые элементы ИЛИ собрать трехвходовый элемент ИЛИ, реализующий функцию

$$f(x_1, x_2, x_3) = (x_1 + x_2) + x_3 = x_1 + x_2 + x_3,$$

как показано на рис 1.10а.

2. Изменяя состояния входов  $x_1, x_2, x_3$ , исследовать работу схемы, заполнить таблицу истинности логического элемента ЗИЛИ, таблица 1.7.

Таблица 1.6

Входы		Выход
$x_1$	$x_2$	$f(x_1, x_2)$
0	0	
0	1	
1	0	
1	1	

Таблица 1.7

Входы			Выходы	
$x_1$	$x_2$	$x_3$	$x_1 + x_2$	$x_1 + x_2 + x_3$
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

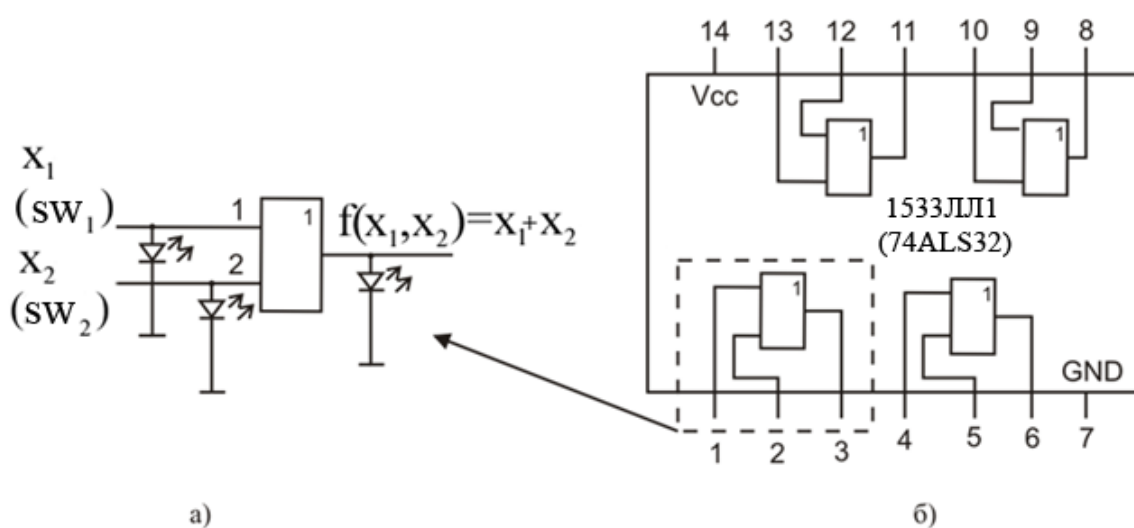


Рис 1.9

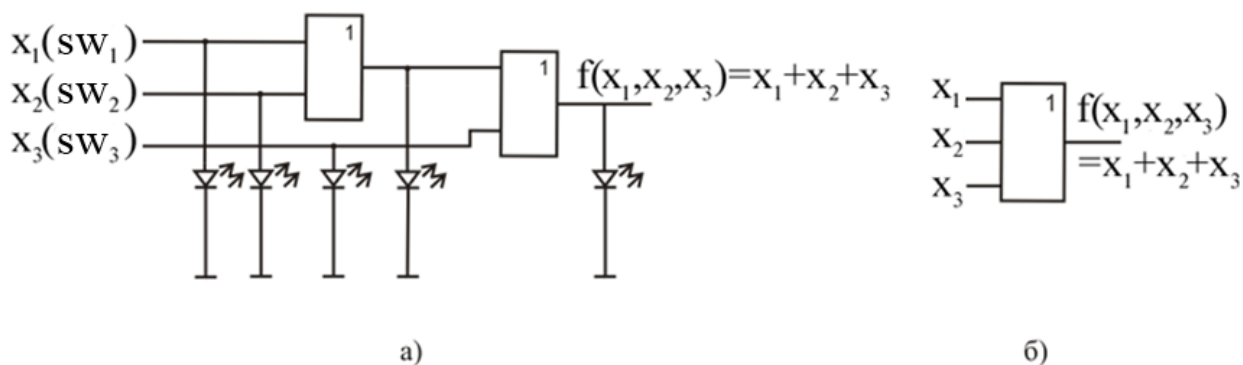


Рис 1.10

### 3.3. Исследование инвертора

1. Разместить ИС 1533ЛН1 (74ALS04) на наборной панели IDL-800.
2. Вывод 14 ИС соединить с источником питания +5V, а вывод 7 с общей шиной установки.
3. Собрать схему, как показано на рис. 1.11а.
4. Изменяя состояние входа  $X$ , исследовать работу инвертора, заполнить таблицу истинности инвертора, табл. 1.8.

Таблица 1.8

Вход	Выход
$x$	$f(x) = \bar{x}$
0	
1	

### 3.4. Исследование логического элемента И-НЕ

#### Двухвходовый элемент И-НЕ

1. Разместить ИС 1533ЛА3(74ALS00) на наборной панели IDL-800.
2. Вывод 14 ИС соединить с источником питания +5V, а вывод 7 с общей шиной установки.
3. Собрать схему, как показано на рис.12а.
4. Изменяя состояния входов  $x_1$  и  $x_2$ , исследовать работу двухвходового элемента И-НЕ, заполнить таблицу истинности, табл. 1.9.

Таблица 1.9

ВХОДЫ		ВЫХОД
$x_1$	$x_2$	$f(x_1, x_2)$
0	0	
0	1	
1	0	
1	1	

Таблица 1.10

ВХОД	ВЫХОД
$x$	$f(x) = \bar{x}$
0	
1	

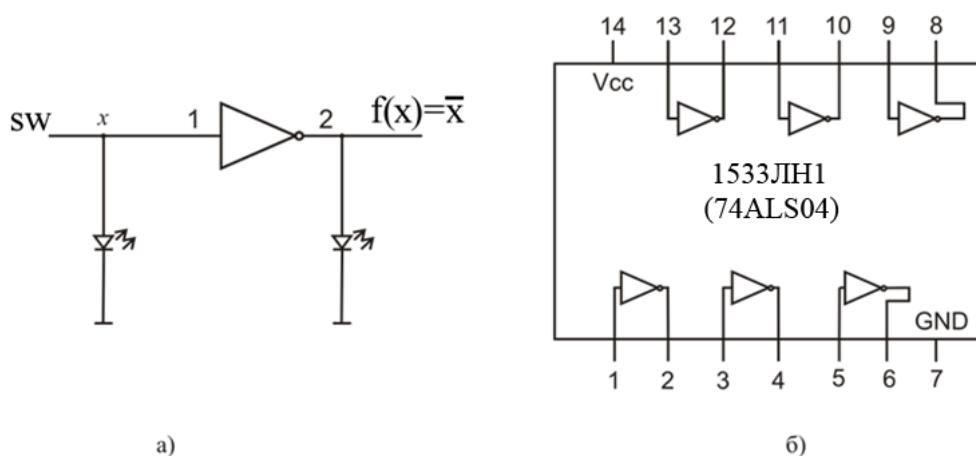


Рис 1.11

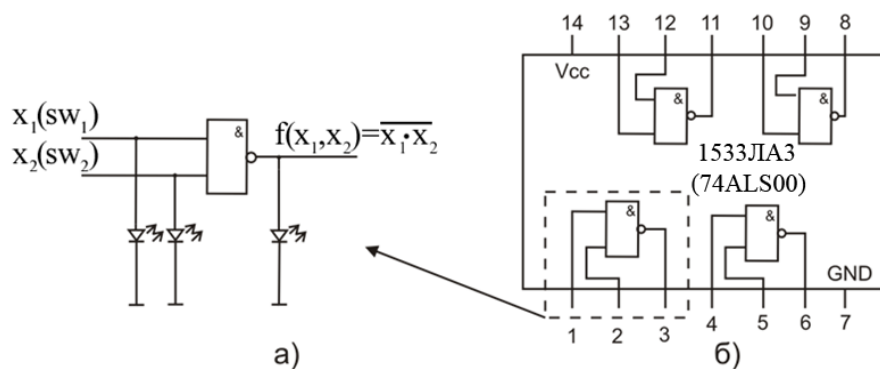


Рис 1.12

### Двухвходовый элемент И-НЕ, используемый как инвертор

Если соединить входы двухвходового элемента И-НЕ, то элемент будет работать как инвертор.

1. Собрать схему, как показано на рис. 1.13 а.

2. Изменяя состояние входа X, исследовать работу схемы, заполнить таблицу истинности, табл. 1.10.

### Многовходовый элемент И-НЕ

1. Используя двухвходовые элементы И-НЕ, собрать трехвходовый элемент И-НЕ, реализующий функцию  $f(x_1, x_2, x_3) = x_1 \cdot x_2 \cdot x_3$ , рис. 1.13 б.

2. Изменяя состояния входов  $x_1, x_2, x_3$ , исследовать работу схемы, заполнить таблицу истинности, табл. 1.11.

### Реализация функции ИЛИ, с помощью логических элементов И-НЕ

1. Используя двухвходовые элементы И-НЕ, собрать схему, реализующую операцию ИЛИ, как показано на рис. 1.13в.

2. Изменяя состояния входов  $x_1$  и  $x_2$ , исследовать работу схемы, заполнить таблицу истинности, табл. 1.12.

Таблица 1.11

Входы			Выходы		
$x_1$	$x_2$	$x_3$	$\overline{x_1 \cdot x_2}$	$x_1 \cdot x_2$	$\overline{x_1 \cdot x_2 \cdot x_3}$
0	0	0			
0	0	1			
0	1	0			
0	1	1			
1	0	0			
1	0	1			
1	1	0			
1	1	1			

Таблица 1.12

Входы		Выходы		
$x_1$	$x_2$	$\bar{x}_1$	$\bar{x}_2$	$x_1 + x_2$
0	0			
0	1			
1	0			
1	1			

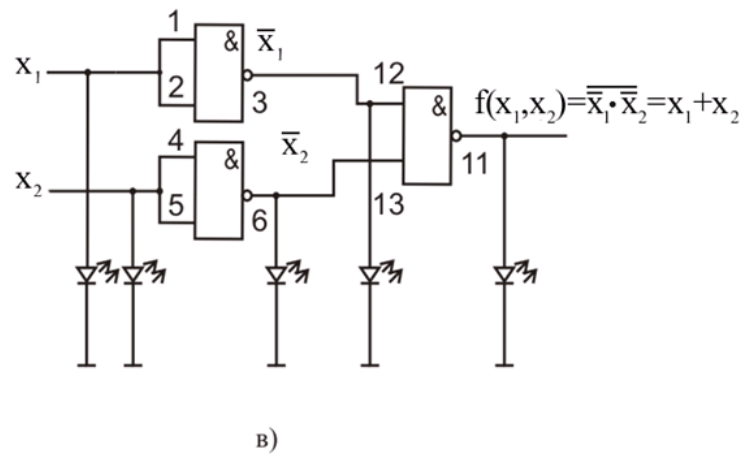
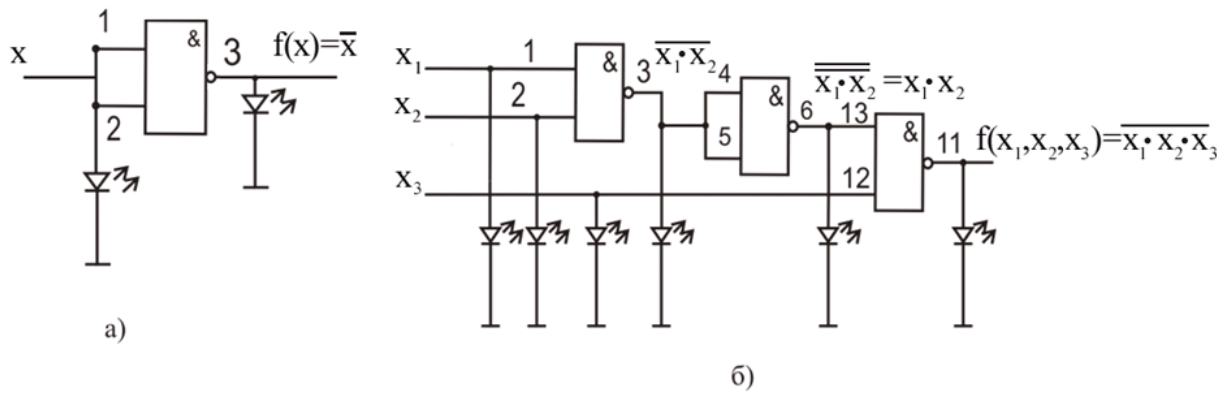


Рис 1.13

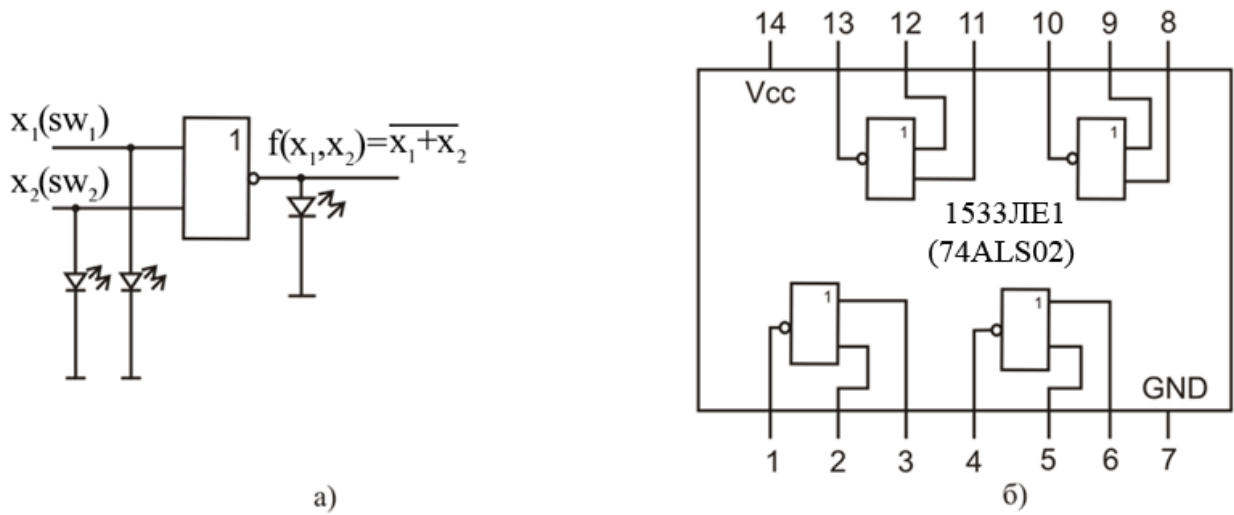


Рис 1.14

### 3.5. Исследование логического элемента ИЛИ-НЕ

#### Двухвходовый элемент ИЛИ-НЕ

1. Разместить ИС 1533ЛЕ1 (74ALS02) на наборной панели IDL-800.
2. Вывод 14 ИС соединить с источником питания +5V, а вывод 7 с общей шиной установки.
3. Собрать схему, как показано на рис. 1.14 а.
4. Изменяя состояние входов  $x_1$  и  $x_2$ , исследовать работу элемента 2ИЛИ-НЕ, заполнить таблицу истинности, табл. 1.13.

#### Двухвходовый элемент ИЛИ-НЕ, используемый как инвертор

Если соединить входы элемента ИЛИ-НЕ, то элемент будет работать как инвертор.

1. Собрать схему, как показано на рис.1.15 а.
2. Изменяя состояние входа  $x$ , исследовать работу схемы, заполнить таблицу истинности, табл. 1.14.

Таблица 1.13

Входы		Выход
$x_1$	$x_2$	$f(x_1, x_2) = \overline{x_1 + x_2}$
0	0	
0	1	
1	0	
1	1	

Таблица 1.14

Вход	Выход
$x$	$f(x) = \bar{x}$
0	
1	

#### Многовходовый элемент ИЛИ-НЕ

1. Используя двухвходовые элементы ИЛИ-НЕ, собрать трехвходовый элемент ИЛИ-НЕ, реализующий функцию  $f(x_1, x_2, x_3) = \overline{x_1 + x_2 + x_3}$ , рис. 1.15б.
2. Изменяя состояние входов  $x_1, x_2, x_3$ , исследовать работу схемы, заполнить таблицу истинности, табл. 1.15.

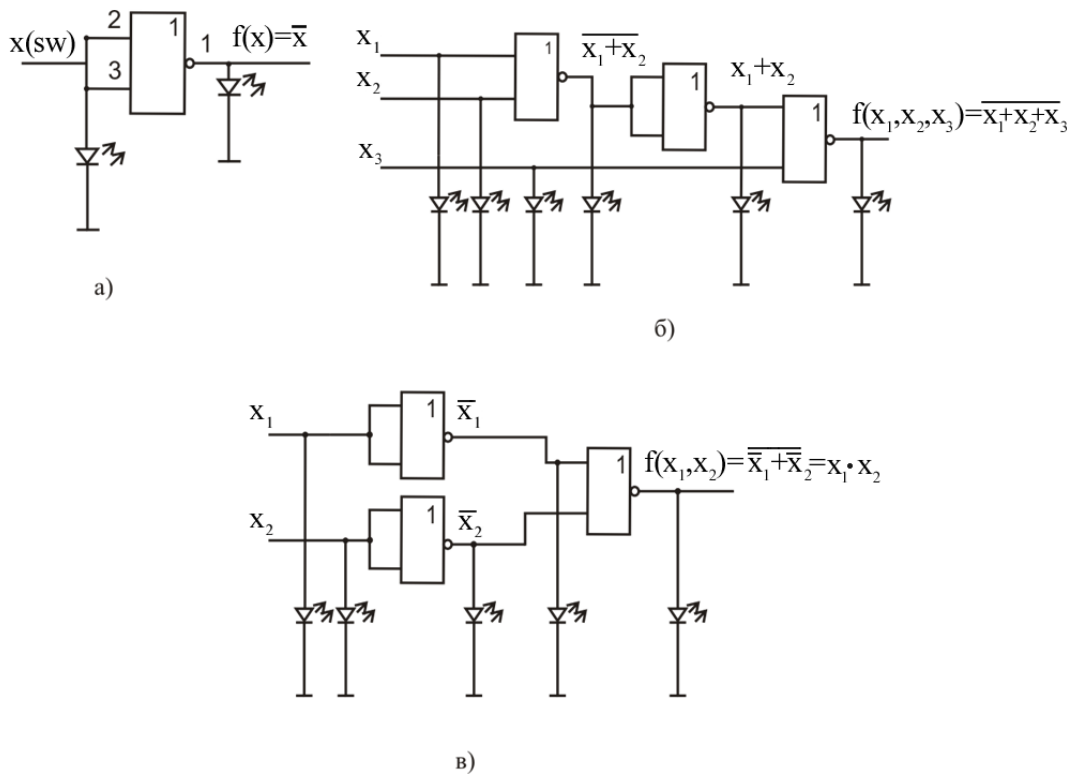


Рис 1.15

### Реализация функции И с помощью логических элементов ИЛИ-НЕ

- Используя двухвходовые элементы ИЛИ-НЕ, собрать схему, реализующую операцию И, как показано на рис 1.15в.
- Изменяя состояния входов  $x_1, x_2$ , исследовать работу схемы, заполнить таблицу истинности, табл. 1.16.

Таблица 1.15

Входы			Выходы		
$x_1$	$x_2$	$x_3$	$\overline{x_1 + x_2}$	$x_1 + x_2$	$\overline{x_1 + x_2 + x_3}$
0	0	0			
0	0	1			
0	1	0			
0	1	1			
1	0	0			
1	0	1			
1	1	0			
1	1	1			

Таблица 1.16

Входы		Выходы		
$x_1$	$x_2$	$\bar{x}_1$	$\bar{x}_2$	$x_1 \cdot x_2$
0	0			
0	1			
1	0			
1	1			

### 3.6 Исследование логического элемента ИСКЛЮЧАЮЩЕЕ ИЛИ

#### Двухвходовый элемент ИСКЛЮЧАЮЩЕЕ ИЛИ

1. Разместить ИС 1533ЛП5 (74ALS86) на наборной панели IDL-800.
2. Выход 14 ИС соединить с источником питания +5V, а вывод 7 с общей шиной установки.
3. Собрать схему, как показано на рис. 1.16а.
4. Изменяя состояние входов  $x_1$  и  $x_2$ , исследовать работу двухвходового элемента ИСКЛЮЧАЮЩЕЕ ИЛИ, заполнить таблицу истинности, табл. 1.17.

#### Многовходовый элемент ИСКЛЮЧАЮЩЕЕ ИЛИ

1. Используя двухвходовые элементы ИСКЛЮЧАЮЩЕЕ ИЛИ, собрать схему трехвходового элемента ИСКЛЮЧАЮЩЕЕ ИЛИ, рис.1.17а.
2. Изменяя состояния входов  $x_1, x_2, x_3$ , исследовать работу схемы, заполнить таблицу истинности, табл. 1.18.

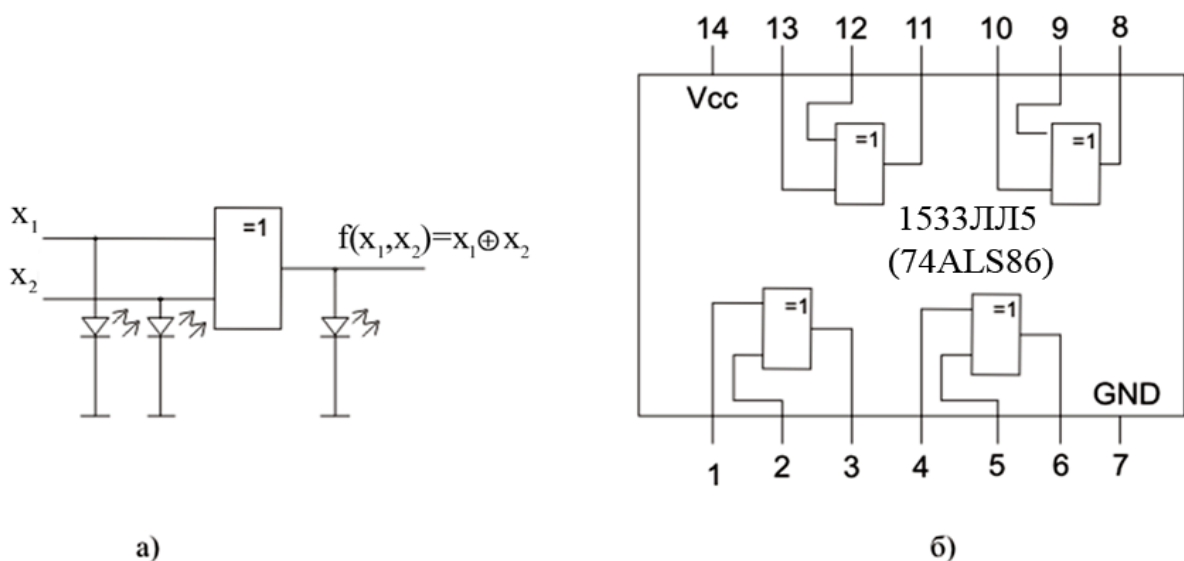
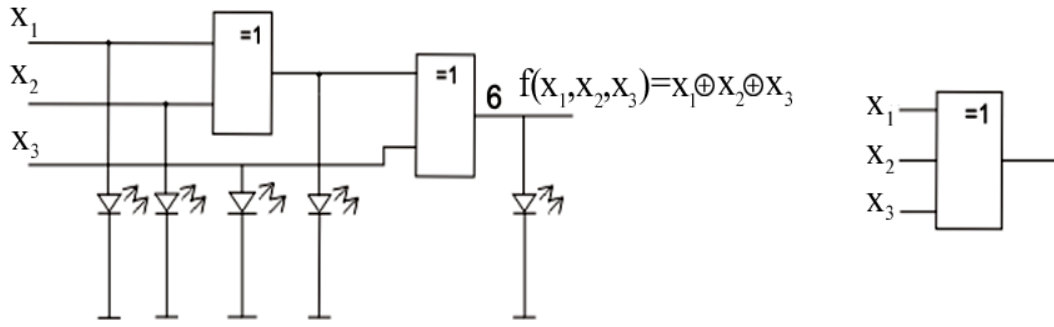
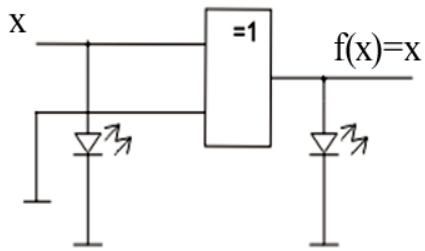


Рис 1.16

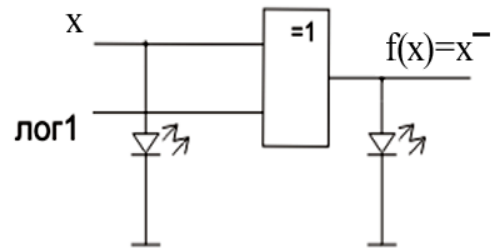


а)

б)



в)



г)

Рис 1.17

Таблица 1.17

Входы		Выход
$x_1$	$x_2$	$f(x_1, x_2) = x_1 \oplus x_2$
0	0	
0	1	
1	0	
1	1	

Таблица 1.18

Входы			Выходы	
$x_1$	$x_2$	$x_3$	$x_1 \oplus x_2$	$x_1 \oplus x_2 \oplus x_3$
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

Логический элемент ИСКЛЮЧАЮЩЕЕ ИЛИ, используемый как повторитель

1. Собрать схему, как показано на рис. 1.17в.
2. Изменяя состояние входа  $x$ , исследовать работу схемы, заполнить таблицу истинности, табл. 1.19.

Логический элемент ИСКЛЮЧАЮЩЕЕ ИЛИ, используемый как инвертор

1. Собрать схему, как показано на рис. 1.17 г.
2. Изменяя состояния входа  $x$ , исследовать работу схемы, заполнить таблицу истинности, табл. 1.20.

Таблица 1.19

Вход	Выход
$x$	$f(x) = x$
0	
1	

Таблица 1.20

Вход	Выход
$x$	$f(x) = \bar{x}$
0	
1	

#### 4. СОДЕРЖАНИЕ ОТЧЕТА

1. Цель работы.
2. Схемы, исследуемые в работе.
3. Таблицы, отражающие результаты исследований.
4. Выводы по результатам исследований.

#### 5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что называется функцией алгебры логики?
2. Основные ФАЛ для двух переменных.
3. Что такое полная система ФАЛ?
4. Основные логические элементы, их обозначения.
5. Многовходовые логические элементы.

## Лабораторная работа №2

# СИНТЕЗ КОМБИНАЦИОННЫХ УСТРОЙСТВ В ЗАДАННОМ БАЗИСЕ ЛОГИЧЕСКИХ ЭЛЕМЕНТОВ

### 1. ЦЕЛЬ РАБОТЫ

Изучение представления функций алгебры логики (ФАЛ) в дизъюнктивной нормальной форме и конъюнктивной нормальной форме. Реализация этих функций в базисах И-ИЛИ, И-НЕ; ИЛИ-И, ИЛИ-НЕ.

### 2. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Любая ФАЛ может быть представлена логическим выражением в одной из следующих форм:

- дизъюнктивной нормальной форме,
- конъюнктивной нормальной форме.

Это, конечно, не означает, что ФАЛ не может быть представлена в другой форме. ФАЛ может быть представлена в различных формах, но эти две формы удобны при упрощении ФАЛ стандартными методами.

#### Рассмотрим пример:

Пусть дана следующая логическая функция

$$f(x_1, x_2, x_3) = (x_1 + x_2 \bar{x}_3)(x_2 + x_1 x_3) \quad (2.1)$$

#### Необходимо:

1. Реализовать эту функцию, используя логические элементы.
2. Определить – возможно ли реализовать эту функцию с использованием только логических элементов И-НЕ и ИЛИ-НЕ и если возможно, построить такие схемы.
3. Определить – возможно, ли упростить эту функцию, если да – то упростить.
4. Построить схемы по упрощенным выражениям.
5. Сравнить полученные схемы.

#### Решение:

Из выражения для  $f(x_1, x_2, x_3)$  видно, что имеются три переменные  $x_1, x_2, x_3$ . При этом переменная  $x_3$  дана в нормальном (неинверсном) виде и в инверсном виде  $\bar{x}_3$ .

Схема может быть построена путем рассмотрения выражения (2.1) и выбора логических элементов для реализации соответствующих термов выражения. Примем, что переменные возможны в неинвертированном и инвертированном виде.

Первый терм  $x_1$  представляет собой только одну переменную, а второй терм имеет переменные  $x_2$  и  $\bar{x}_3$ . Очевидно, что второй терм не что иное, как элементарная конъюнкция и этот терм может быть реализован, используя двухвходовый логический элемент И. Комбинация первых двух термов реализуется с помощью элемента ИЛИ, как показано на рис. 2.1а.

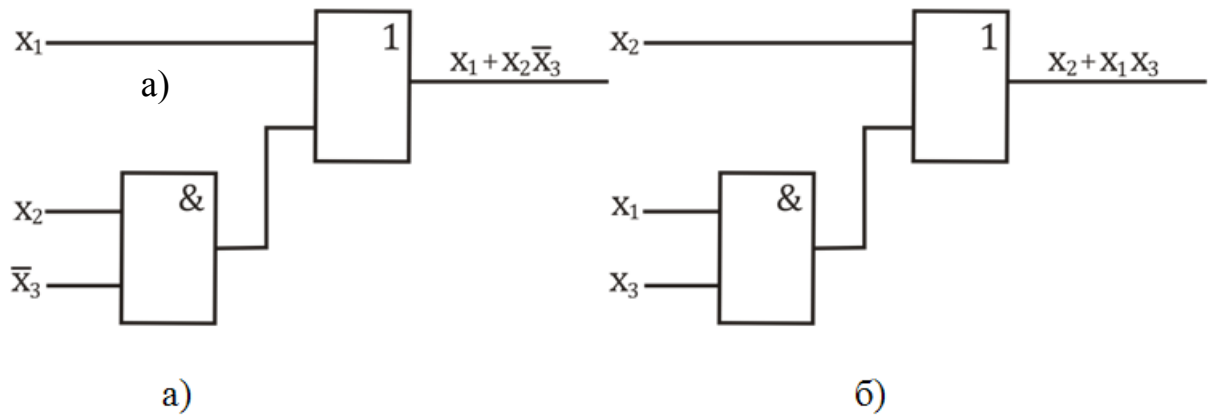


Рис. 2.1

Третий терм  $x_2$  снова просто одна переменная и четвертый терм представляет собой конъюнкцию двух переменных  $x_1x_3$ . Схема для реализации этих двух термов строится также как и для реализации двух предыдущих, рис. 2.1б.

Теперь, с использованием дополнительного логического элемента И, построим полную схему, которая реализует данную функцию (рис.2.2).

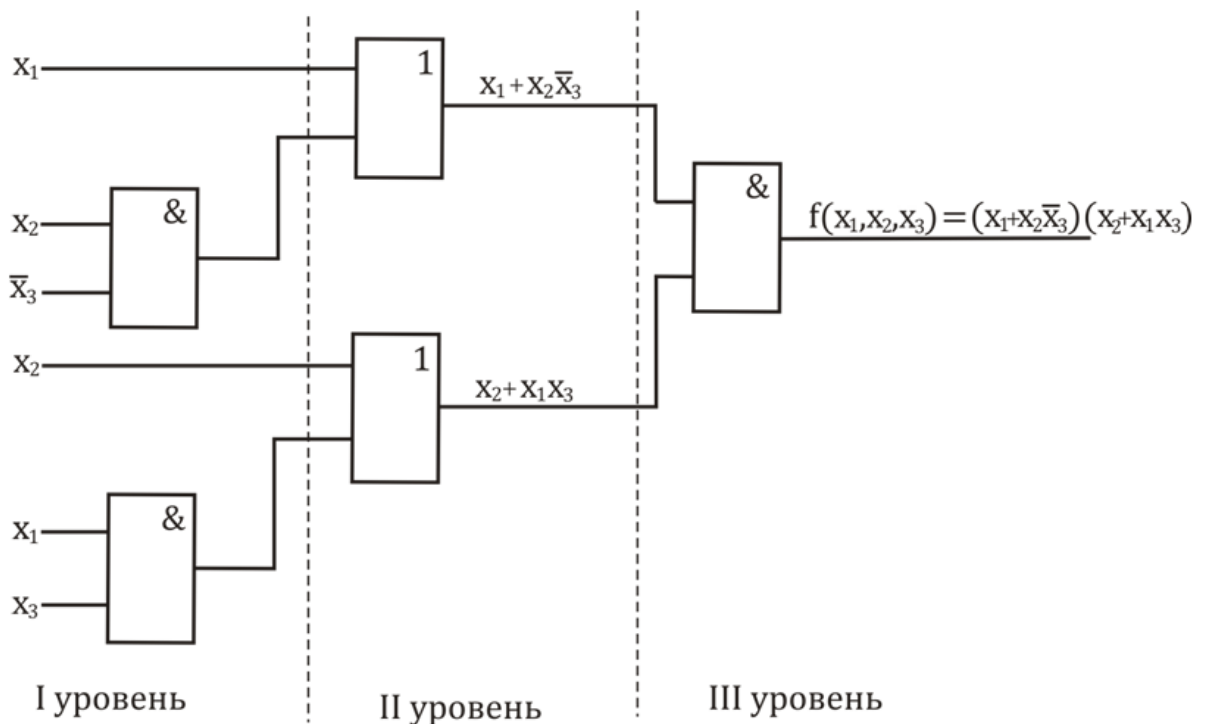


Рис. 2.2

Для реализации схемы на рис.2.2 требуется три двухвходовых элемента И и два двухвходовых элемента ИЛИ. Такая реализация известна как трехуровневая реализация.

### Дизъюнктивная нормальная форма

Преобразуем выражение (2.1) следующим образом:

$$f(x_1, x_2, x_3) = (x_1 + x_2 \bar{x}_3)(x_2 + x_1 x_3) = x_1 x_2 + x_1 x_1 x_3 + x_2 \bar{x}_3 x_2 + x_2 \bar{x}_3 x_1 x_3 = x_1 x_2 + x_1 x_3 + x_2 \bar{x}_3 \quad (2.2)$$

Представление ФАЛ в таком виде (2.2), известно как представление ФАЛ в *дизъюнктивной нормальной форме*. Итак, формы ФАЛ, представляющие дизъюнкцию элементарных конъюнкций, называются дизъюнктивными нормальными формами (ДНФ). Под элементарной конъюнкцией понимается логическое произведение отдельных переменных в нормальном или инвертированном виде. Функция, представленная в ДНФ может быть реализована используя И-ИЛИ конфигурацию, рис 2.3.

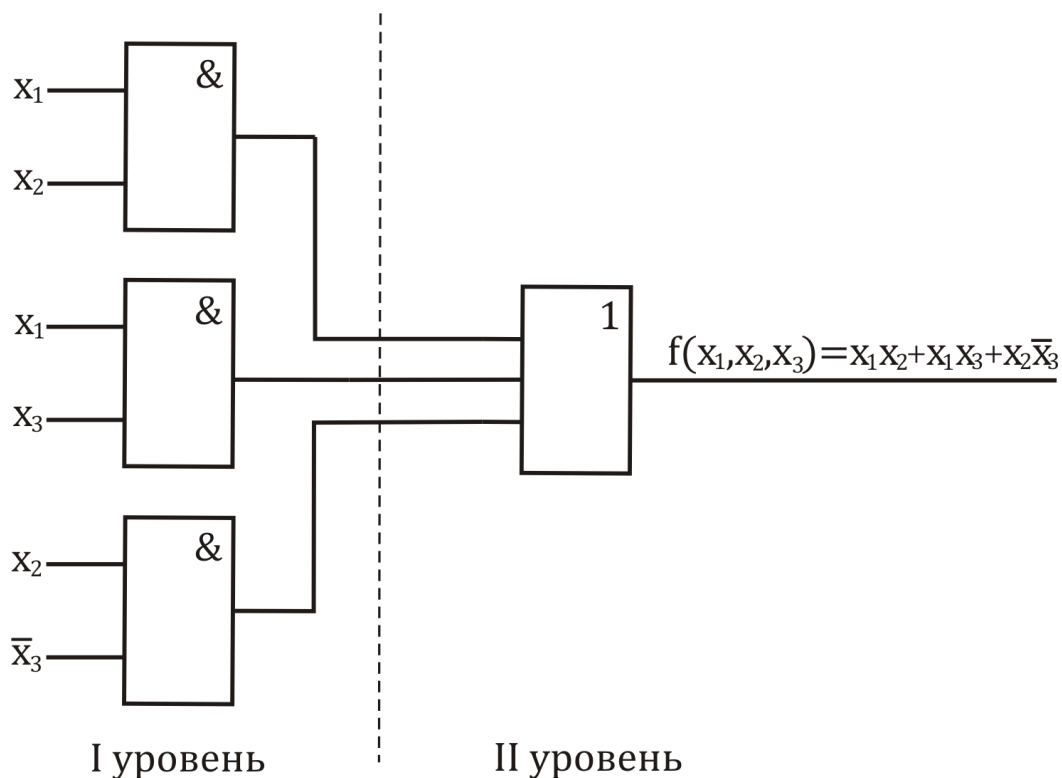


Рис.2.3. Реализация ФАЛ представленной в ДНФ (И-ИЛИ конфигурация)

Реализация ФАЛ рис.2.3 известна как двухуровневая реализация. Первый уровень состоит из И элементов, и второй уровень состоит из элемента ИЛИ.

Используя закон де Моргана, выражение (2.2) можно переписать следующим образом:

$$f(x_1, x_2, x_3) = \overline{\overline{x_1 x_2} + \overline{x_1 x_3} + \overline{x_2 \bar{x}_3}} = \overline{\overline{x_1 x_2} \cdot \overline{x_1 x_3} \cdot \overline{x_2 \bar{x}_3}} \quad (2.3)$$

Теперь функция (2.3) может быть реализована с использованием только И-НЕ элементов, рис. 2.4.

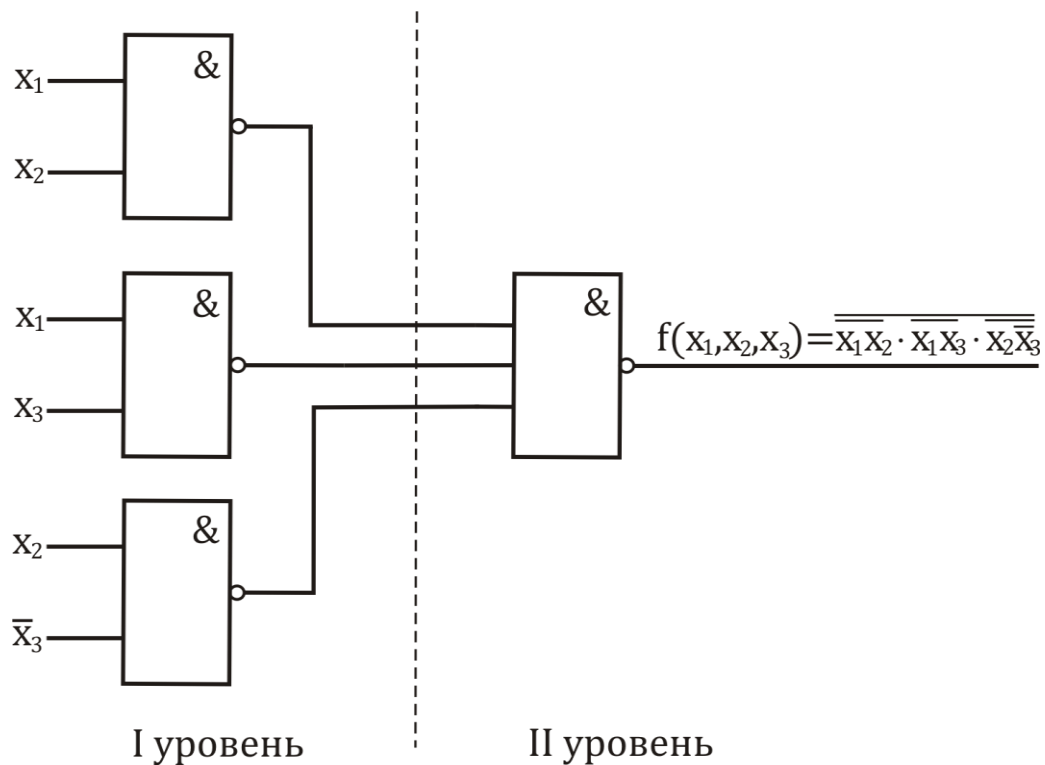


Рис.2.4. Реализация ФАЛ, используя только И-НЕ элементы

Вывод: для того, чтобы реализовать ФАЛ с помощью только элементов И-НЕ, необходимо представить ФАЛ в ДНФ, а затем использовать двойное инвертирование и закон де Моргана.

Опять преобразуем исходную ФАЛ, используя распределительный закон для оператора И.

$$\begin{aligned} f(x_1, x_2, x_3) &= (x_1 + x_2 \bar{x}_3)(x_2 + x_1 x_3) = (x_1 + x_2)(x_1 + \bar{x}_3)(x_2 + x_1)(x_2 + x_3) = \\ &= (x_1 + x_2)(x_1 + \bar{x}_3)(x_2 + x_3) \end{aligned} \quad (2.4)$$

Представление ФАЛ в виде (2.4) известно, как представление ФАЛ в конъюнктивной нормальной форме. Итак, формы ФАЛ, представляющие конъюнкцию элементарных дизъюнкций, называются конъюнктивными нормальными формами (КНФ). Под элементарной дизъюнкцией понимается логическая сумма отдельных переменных в нормальном или инвертированном виде. Функция, представленная в КНФ, может быть реализована, используя ИЛИ-И конфигурацию, рис. 2.5.

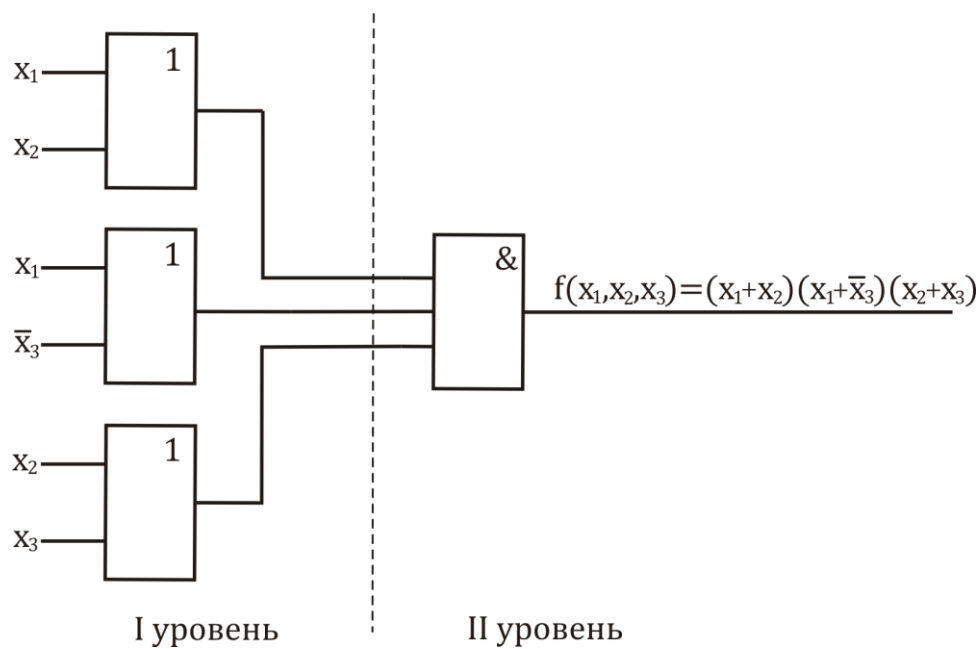


Рис. 2.5. Реализация ФАЛ представленной в КНФ (ИЛИ-И конфигурация)

Используя двойную инверсию и закон де Моргана, выражение (2.4) можно переписать следующим образом:

$$f(x_1, x_2, x_3) = \overline{\overline{(x_1 + x_2)(x_1 + \bar{x}_3)(x_2 + x_3)}} = \overline{\overline{x_1 + x_2 + \bar{x}_1 + \bar{x}_3 + x_2 + x_3}} \quad (2.5)$$

Теперь функция может быть реализована с использованием только ИЛИ-НЕ элементов, рис. 2.6.

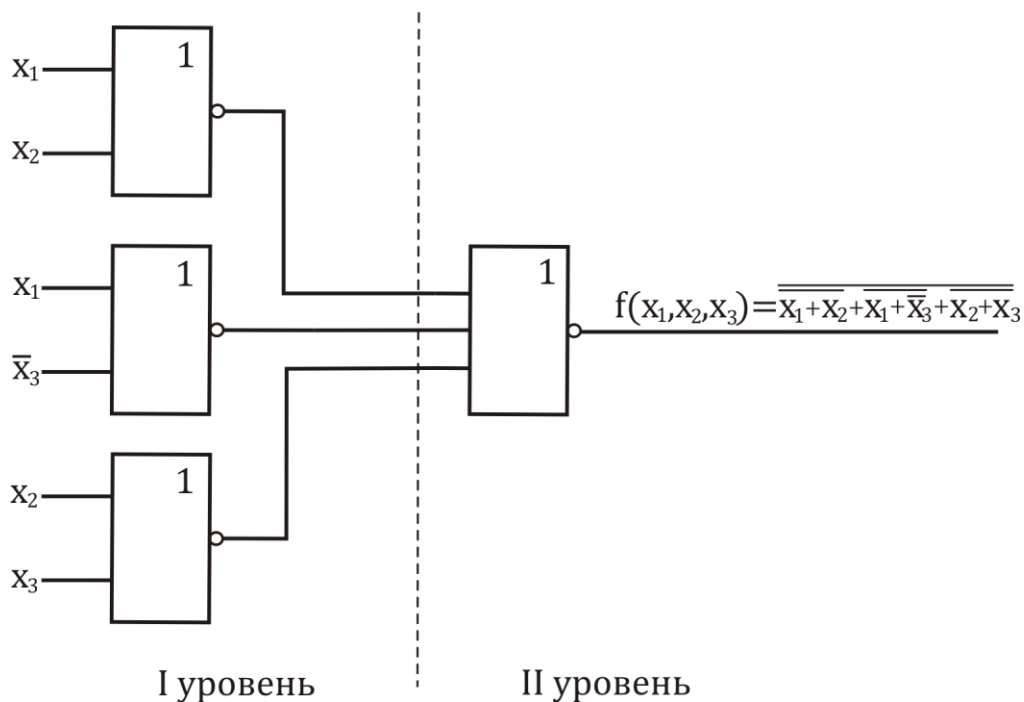


Рис. 2.6. Реализация ФАЛ используя только ИЛИ-НЕ элементы

Вывод: Для того чтобы реализовать ФАЛ с помощью только ИЛИ-НЕ элементов, необходимо представить ФАЛ в КНФ, а затем использовать двойное инвертирование и закон де Моргана.

Теперь рассмотрим, как выражения (2.2) и (2.4) могут быть упрощены.

$$\text{а) } f(x_1, x_2, x_3) = x_1 x_2 + x_1 x_3 + x_2 \bar{x}_3 = x_1 x_2 (x_3 + \bar{x}_3) + x_1 x_3 + x_2 \bar{x}_3 = x_1 x_2 x_3 + x_1 x_2 \bar{x}_3 + x_1 x_3 + x_2 \bar{x}_3 = x_1 x_3 (x_2 + 1) + x_2 \bar{x}_3 (x_1 + 1) = x_1 x_3 + x_2 \bar{x}_3 \quad (2.6)$$

$$\text{б) } f(x_1, x_2, x_3) = (x_1 + x_2)(x_1 + \bar{x}_3)(x_2 + x_3) = (x_1 + x_2 + x_3 \bar{x}_3)(x_1 + \bar{x}_3)(x_2 + x_3) = (x_1 + x_2 + x_3)(x_1 + x_2 + \bar{x}_3)(x_1 + \bar{x}_3)(x_2 + x_3) = (x_1 + \bar{x}_3)(x_2 + x_3) \quad (2.7)$$

Реализация выражений (2.6) и (2.7), используя только элементы И-НЕ и ИЛИ-НЕ, показана на рис. 2.7.

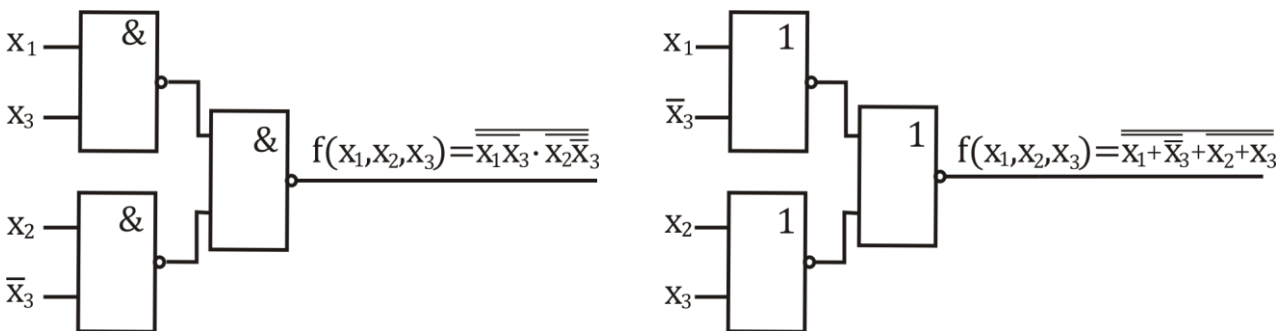


Рис.2.7. Реализация функции после упрощения

### Сравнение полученных схем

а) Реализация функции в соответствии с выражением (2.1) потребовала максимальное количество элементов и полученная схема оказалась трехуровневой, что снижает её быстродействие.

б) Реализация функции в соответствии с выражениями (2.3) и (2.5) является очень полезным, поскольку используется только один тип логических элементов (И-НЕ/ИЛИ-НЕ), которые находятся в одном корпусе ИС. Эти схемы двухуровневые.

в) Для реализации функции в соответствии с выражениями (2.6) и (2.7) требуется минимальное количество элементов, поэтому упрощение логических выражений является очень полезными.

Рассмотрим снова выражение (2.2), в котором функция представлена в ДНФ и выражение (2.4), в котором функция представлена в КНФ. В этих выражениях индивидуальные термы не содержат все переменные. Если все термы в ДНФ и КНФ содержат все переменные, то такие ДНФ и КНФ называют совершенными. Каждый индивидуальный терм в совершенной дизъюнктивной нормальной форме (СДНФ) называется минтермом, а каждый индивидуальный

терм в совершенной конъюнктивной нормальной форме (СКНФ) называется макстермом.

Функция в ДНФ может быть преобразована в СДНФ путем логического умножения термов (конъюнкций) ДНФ с термами, образованными путем логического сложения переменной и ее отрицания, которая отсутствует в исходных термах.

**Пример 1.** Преобразовать функцию  $f(x_1, x_2, x_3) = x_1 x_2 + x_1 x_3 + x_2 \bar{x}_3$  в СДНФ

### *Решение*

$$\begin{aligned} f(x_1, x_2, x_3) &= x_1 x_2 + x_1 x_3 + x_2 \bar{x}_3 = x_1 x_2 (x_3 + \bar{x}_3) + x_1 x_3 (x_2 + \bar{x}_2) + x_2 \bar{x}_3 (x_1 + \bar{x}_1) = \\ &= x_1 x_2 x_3 + x_1 x_2 \bar{x}_3 + x_1 x_2 x_3 + x_1 \bar{x}_2 x_3 + x_1 x_2 \bar{x}_3 + \bar{x}_1 x_2 \bar{x}_3 = \\ &= x_1 x_2 x_3 + x_1 x_2 \bar{x}_3 + x_1 \bar{x}_2 x_3 + \bar{x}_1 x_2 \bar{x}_3 \end{aligned}$$

Функция в КНФ может быть преобразована в СКНФ путем логического сложения термов (дизъюнкций) КНФ с термами, образованными путем логического умножения переменной и ее отрицания, которая отсутствует в исходном терме.

**Пример 2.** Преобразовать функцию  $f(x_1, x_2, x_3) = (x_1 + \bar{x}_3)(x_2 + x_3)$  в СКНФ.

### *Решение*

$$\begin{aligned} f(x_1, x_2, x_3) &= (x_1 + \bar{x}_3)(x_2 + x_3) = (x_1 + \bar{x}_3 + x_2 \bar{x}_2)(x_2 + x_3 + x_1 \bar{x}_1) = \\ &= (x_1 + x_2 + \bar{x}_3)(x_1 + \bar{x}_2 + \bar{x}_3)(x_1 + x_2 + x_3)(\bar{x}_1 + x_2 + x_3) \end{aligned}$$

Если ФАЛ имеет  $n$ - переменных, то число возможных минтермов (конституент единицы или составляющих единицы) равно  $2^n$ . Также число возможных макстермов (конституент нуля или составляющих нуля) равно  $2^n$ .

В таблице 2.1 представлены все возможные минтермы и макстермы для четырех переменных. В левой части таблицы записаны все наборы переменных. Минтерм для любого набора переменных записывается в виде конъюнкции. Причем, если на данном наборе переменная равна единице, то переменная записывается в нормальном (неинвертированном) виде и если переменная равна нулю, то переменная записывается в инвертированном виде. Макстерм для любого набора переменных записывается в виде дизъюнкции. Причем, если на данном наборе переменная равна единице, то она записывается в инвертированном виде и если переменная равно нулю, то она записывается в нормальном виде.

В правой части таблицы представлена ФАЛ, описывающая работу устройства сравнения, которое сравнивает два двоичных двухразрядных числа  $A(x_1, x_2)$  и  $B(x_3, x_4)$ . Если эти два числа равны, на выходе устройства сравнения должна быть единица, а если не равны – ноль.

Таблица 2.1

Переменные	Минтермы	Макстермы	Функция У
$X_1 X_2 X_3 X_4$	$m_i$	$M_i$	$A(X_1, X_2) = B(X_3, X_4)$
0 0 0 0	$\bar{X}_1 \cdot \bar{X}_2 \cdot \bar{X}_3 \cdot \bar{X}_4 = m_0$	$x_1 + X_2 + X_3 + X_4 = M_0$	1
0 0 0 1	$\bar{X}_1 \cdot \bar{X}_2 \cdot \bar{X}_3 \cdot X_4 = m_1$	$X_1 + X_2 + X_3 + \bar{X}_4 = M_1$	0
0 0 1 0	$\bar{X}_1 \cdot \bar{X}_2 \cdot X_3 \cdot \bar{X}_4 = m_2$	$X_1 + X_2 + \bar{X}_3 + X_4 = M_2$	0
0 0 1 1	$\bar{X}_1 \cdot \bar{X}_2 \cdot X_3 \cdot X_4 = m_3$	$X_1 + X_2 + \bar{X}_3 + \bar{X}_4 = M_3$	0
0 1 0 0	$\bar{X}_1 \cdot X_2 \cdot \bar{X}_3 \cdot \bar{X}_4 = m_4$	$X_1 + \bar{X}_2 + X_3 + X_4 = M_4$	0
0 1 0 1	$\bar{X}_1 \cdot X_2 \cdot \bar{X}_3 \cdot X_4 = m_5$	$X_1 + \bar{X}_2 + X_3 + \bar{X}_4 = M_5$	1
0 1 1 0	$\bar{X}_1 \cdot X_2 \cdot X_3 \cdot \bar{X}_4 = m_6$	$X_1 + \bar{X}_2 + \bar{X}_3 + X_4 = M_6$	0
0 1 1 1	$\bar{X}_1 \cdot X_2 \cdot X_3 \cdot X_4 = m_7$	$X_1 + \bar{X}_2 + \bar{X}_3 + \bar{X}_4 = M_7$	0
1 0 0 0	$X_1 \cdot \bar{X}_2 \cdot \bar{X}_3 \cdot \bar{X}_4 = m_8$	$\bar{X}_1 + X_2 + X_3 + X_4 = M_8$	0
1 0 0 1	$X_1 \cdot \bar{X}_2 \cdot \bar{X}_3 \cdot X_4 = m_9$	$\bar{X}_1 + X_2 + X_3 + \bar{X}_4 = M_9$	0
1 0 1 0	$X_1 \cdot \bar{X}_2 \cdot X_3 \cdot \bar{X}_4 = m_{10}$	$\bar{X}_1 + X_2 + \bar{X}_3 + X_4 = M_{10}$	1
1 0 1 1	$X_1 \cdot \bar{X}_2 \cdot X_3 \cdot X_4 = m_{11}$	$\bar{X}_1 + X_2 + \bar{X}_3 + \bar{X}_4 = M_{11}$	0
1 1 0 0	$X_1 \cdot X_2 \cdot \bar{X}_3 \cdot \bar{X}_4 = m_{12}$	$\bar{X}_1 + \bar{X}_2 + X_3 + X_4 = M_{12}$	0
1 1 0 1	$X_1 \cdot X_2 \cdot \bar{X}_3 \cdot X_4 = m_{13}$	$\bar{X}_1 + \bar{X}_2 + X_3 + \bar{X}_4 = M_{13}$	0
1 1 1 0	$X_1 \cdot X_2 \cdot X_3 \cdot \bar{X}_4 = m_{14}$	$\bar{X}_1 + \bar{X}_2 + \bar{X}_3 + X_4 = M_{14}$	0
1 1 1 1	$X_1 \cdot X_2 \cdot X_3 \cdot X_4 = m_{15}$	$\bar{X}_1 + \bar{X}_2 + \bar{X}_3 + \bar{X}_4 = M_{15}$	1

Наборы переменных в левой части таблицы и значения функции в правой части таблицы представляют собой таблицу истинности ФАЛ, описывающую работу устройства сравнения.

Из таблицы истинности ФАЛ может быть записана в СДНФ и в СКНФ.

В СДНФ ФАЛ записывается из таблицы истинности как дизъюнкция минтермов, на которых функция принимает значение 1.

$$\begin{aligned}
 Y &= m_0 + m_5 + m_{10} + m_{15} = \sum m(0, 5, 10, 15) = \\
 &= \bar{x}_1 \bar{x}_2 \bar{x}_3 \bar{x}_4 + \bar{x}_1 x_2 \bar{x}_3 x_4 + x_1 \bar{x}_2 x_3 \bar{x}_4 + x_1 x_2 x_3 x_4
 \end{aligned} \tag{2.8}$$

В СКНФ ФАЛ из таблицы истинности записывается как конъюнкция макстермов, на которых функция принимает значение 0.

$$\begin{aligned}
 Y &= M_1 M_2 M_3 M_4 M_6 M_7 M_8 M_9 M_{11} M_{12} M_{13} M_{14} = \\
 &= \Pi M(1, 2, 3, 4, 6, 7, 8, 9, 11, 12, 13, 14) = \\
 &= (x_1 + x_2 + x_3 + \bar{x}_4)(x_1 + x_2 + \bar{x}_3 + x_4) \cdot \dots \cdot (\bar{x}_1 + \bar{x}_2 + \bar{x}_3 + x_4)
 \end{aligned} \tag{2.9}$$

Уравнения (2.8) и (2.9) представляют одну и ту же функцию, выраженную в минтермах и макстермах.

Эти два представления обладают свойством дополнительности. И если имеется выражение для функции в минтермах (в СДНФ), то выражение в макстермах (в СКНФ) может быть, получено используя это свойство и наоборот.

### 3. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

Оборудование и компоненты: универсальная лабораторная установка IDL-800, ИС 1533ЛА3(74ALS00)-четыре логических элемента 2И-НЕ, ИС 1533ЛА4(74ALS10) -три логических элемента 3И-НЕ, ИС 1533ЛЕ1(74ALS02) - четыре логических элемента 2ИЛИ-НЕ, ИС 1533ЛЕ4(74ALS27) -три логических элемента 3ИЛИ-НЕ. ФАЛ задана следующей таблицей истинности

Табл. 3.1

Входы			Выход
$x_1$	$x_2$	$x_3$	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

#### 3.1. Исследование представления ФАЛ в ДНФ

*Для заданной таблицы истинности:*

- Записать логическое выражение для Y в СДНФ.
- Нарисовать схему и реализовать ее только с помощью элементов И-НЕ. Проверить таблицу истинности.
- Упростить логическое выражение для Y, нарисовать схему и реализовать ее только с помощью элементов И-НЕ. Проверить таблицу истинности.

#### 3.2. Исследование представление ФАЛ в КНФ

*Для заданной таблицы истинности:*

- Записать логическое выражение для Y в СКНФ.

б) Нарисовать схему и реализовать ее только с помощью элементов ИЛИ-НЕ. Проверить таблицу истинности.

в) Упростить логическое выражение для  $Y$ , нарисовать схему и реализовать ее только с помощью элементов ИЛИ-НЕ. Проверить таблицу истинности.

#### **4. СОДЕРЖАНИЕ ОТЧЕТА**

1. Цель работы.
2. Схемы, исследуемые в работе.
3. Таблицы, отражающие результаты исследований.
4. Выводы по результатам исследований.

#### **5. КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. Как представляется ФАЛ в ДНФ?
2. Что такое СДНФ?
3. Что необходимо, чтобы реализовать ФАЛ, используя только элементы И-НЕ?
4. Как представляется ФАЛ в КНФ?
5. Что такое СКНФ?
6. Что необходимо, чтобы реализовать ФАЛ, используя только ИЛИ-НЕ элементы?

# Лабораторная работа № 3

## ИССЛЕДОВАНИЕ СУММАТОРОВ, ВЫЧИТАТЕЛЕЙ И СУММИРОВАНИЯ/ВЫЧИТАНИЯ В ДОПОЛНИТЕЛЬНОМ КОДЕ

### 1. ЦЕЛЬ ЛАБОРАТОРНОЙ РАБОТЫ

- исследование функционирования сумматоров и вычитателей;
- исследование суммирования/вычитания в дополнительных кодах.

### 2. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Несомненно, что наиболее базовой арифметической операцией является суммирование двух двоичных цифр. Это простое сложение состоит из четырех возможных элементарных, а именно  $0+0=0$ ,  $0+1=1$ ,  $1+0=1$  и  $1+1=10$ . Первые три операции дают сумму, чья длина есть одна цифра. Однако, когда обе слагаемых равны единице, двоичная сумма состоит из двух цифр. Старшая цифра называется переносом. Когда слагаемые числа состоят из нескольких цифр, то полученный перенос прибавляется к следующей более значащей паре бит. Комбинационная схема, осуществляющая сложение двух бит называется полусумматором. Если схема осуществляет сложение трех бит (два значащих и перенос из предыдущего разряда), то такая схема называется полным сумматором.

#### Полусумматор

Полусумматор может быть использован для сложения двух наименее значащих бит  $A_0$  и  $B_0$  двух чисел, где отсутствует входной перенос. Условное обозначение полусумматора показано на рис.3.1.

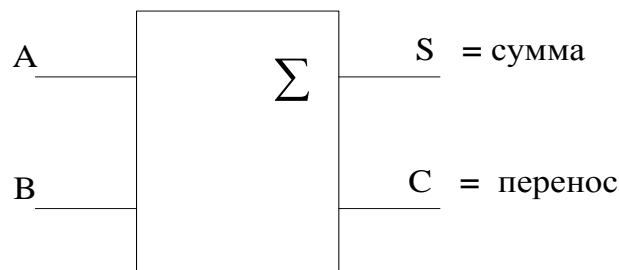


Рис. 3.1. Условное обозначение полусумматора

Таблица истинности полусумматора

Входы		Выходы	
A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Из таблицы истинности мы можем получить логические выражения для S и C.

$$S = \bar{A}B + A\bar{B} = A \oplus B; \quad (3.1)$$

$$C = AB \quad (3.2)$$

В соответствии с выражениями (3.1) и (3.2) полусумматор может быть легко реализован с помощью логических элементов ИСКЛЮЧАЮЩЕ ИЛИ и И, рис.3.2.

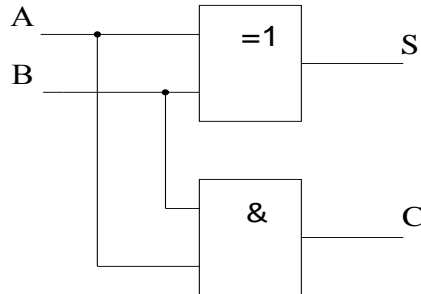


Рис. 3.2. Схема полусумматора

### Полный сумматор

Полусумматор имеет только два входа и не имеет возможности суммировать перенос из младшего разряда при многобитовом сложении. Для этих целей используется третий вход, и схема осуществляет сложение  $A_n, B_n$  и  $C_{n-1}$ , где  $A_n, B_n$  это биты чисел A и B в разряде  $n$ , а  $C_{n-1}$  это перенос при сложении из  $n-1$  разряда.

Таблица истинности полного сумматора

Входы			Выходы	
$A_n$	$B_n$	$C_{n-1}$	$S_n$	$C_n$
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Логические выражения для  $S_n$  и  $C_n$  могут быть упрощены с помощью карт Карно, рис.3.3.

	$B_n$	$C_{n-1}$		
	$C_{n-1}$			
$A_n$	00	01	11	10
0		①	0 <sub>1</sub>	①
1	①	0 <sub>2</sub>	①	

	$B_n$	$C_{n-1}$		
	$C_{n-1}$			
$A_n$	00	01	11	10
0		0	①	
1		①	①	①

Рис. 3.3. Упрощение выражений для  $S_n$  и  $C_n$  полного сумматора

Логическое выражение для  $S_n$  может быть упрощено в базисе функций исключающее ИЛИ.

$$\begin{aligned}
 S_n &= (\bar{A}_n \bar{B}_n C_{n-1} + \bar{A}_n B_n \bar{C}_{n-1}) + (A_n \bar{B}_n \bar{C}_{n-1} + A_n B_n C_{n-1}) = \\
 &= \bar{A}_n (B_n \oplus C_{n-1}) + A_n (\overline{B_n \oplus C_{n-1}}) = A_n \oplus B_n \oplus C_{n-1}
 \end{aligned} \tag{3.3}$$

ФАЛ для  $C_n$  может быть покрыта с помощью обычного логического соседства, а также диагонального соседства, рис.3.3.

$$\begin{aligned}
 C_n &= (A_n \bar{B}_n C_{n-1} + \bar{A}_n B_n C_{n-1}) + (A_n B_n C_{n-1} + A_n B_n \bar{C}_{n-1}) = \\
 &= C_{n-1} (A_n \oplus B_n) + A_n B_n
 \end{aligned} \tag{3.4}$$

В соответствии с выражениями (3.3) и (3.4) на рис.3.4 показана схема полного сумматора.

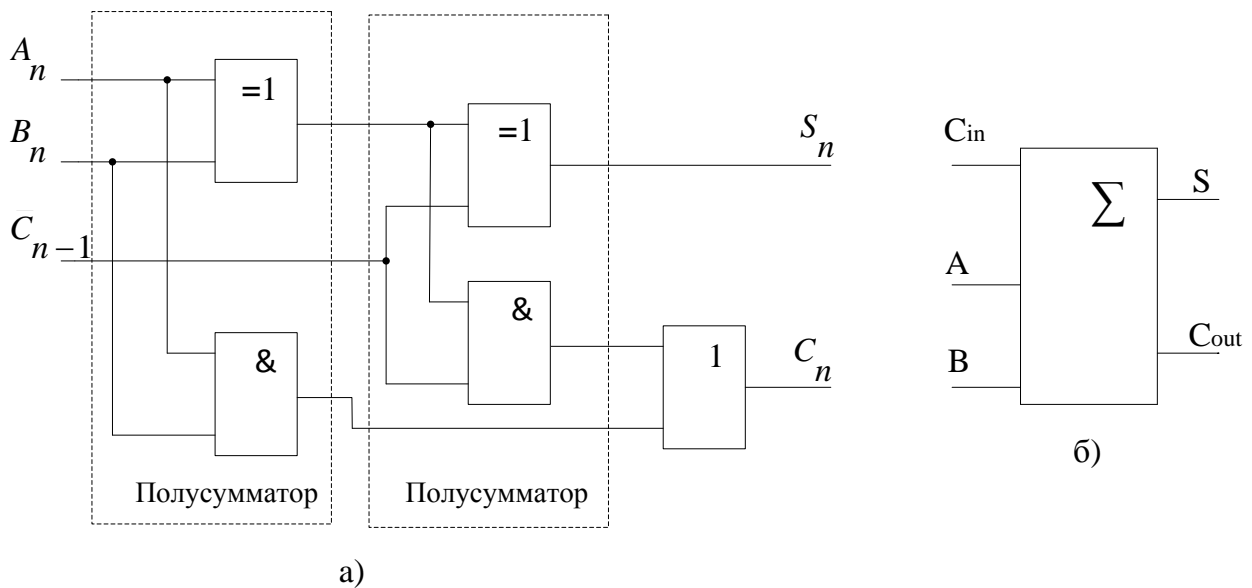


Рис. 3.4. Полный сумматор: а) логическая схема, б) условное обозначение

## Параллельный сумматор

Для сложения четырехразрядных двоичных чисел четыре полных сумматора могут быть соединены, как показано на рис.3.5, для получения четырехразрядного параллельного сумматора. Два четырехразрядных двоичных числа подаются на входы А и В четырех полных сумматоров.  $\hat{A}_0, \hat{A}_0$  - это, конечно, наименее значимые биты. Выход переноса  $\tilde{N}_{out}$  каждого сумматора соединен непосредственно с входом переноса  $\tilde{N}_{in}$  следующего более значимого полного сумматора. Выход  $\tilde{N}_{out}$  наиболее значимого разряда является переносом всей схемы. Биты переноса передаются через схему от разряда к разряду и индивидуальная сумма будет правильной, когда перенос из предыдущего разряда появится в данном разряде. Это означает, что выходная или полная сумма появится, когда все биты переноса пройдут через схему. Поэтому такую схему, рис.3.5а, более точно называют псевдопараллельным сумматором.

В интегральном исполнении выпускается четырехразрядный параллельный сумматор 1533ИМЗ(74ALS83), рис.3.5б.

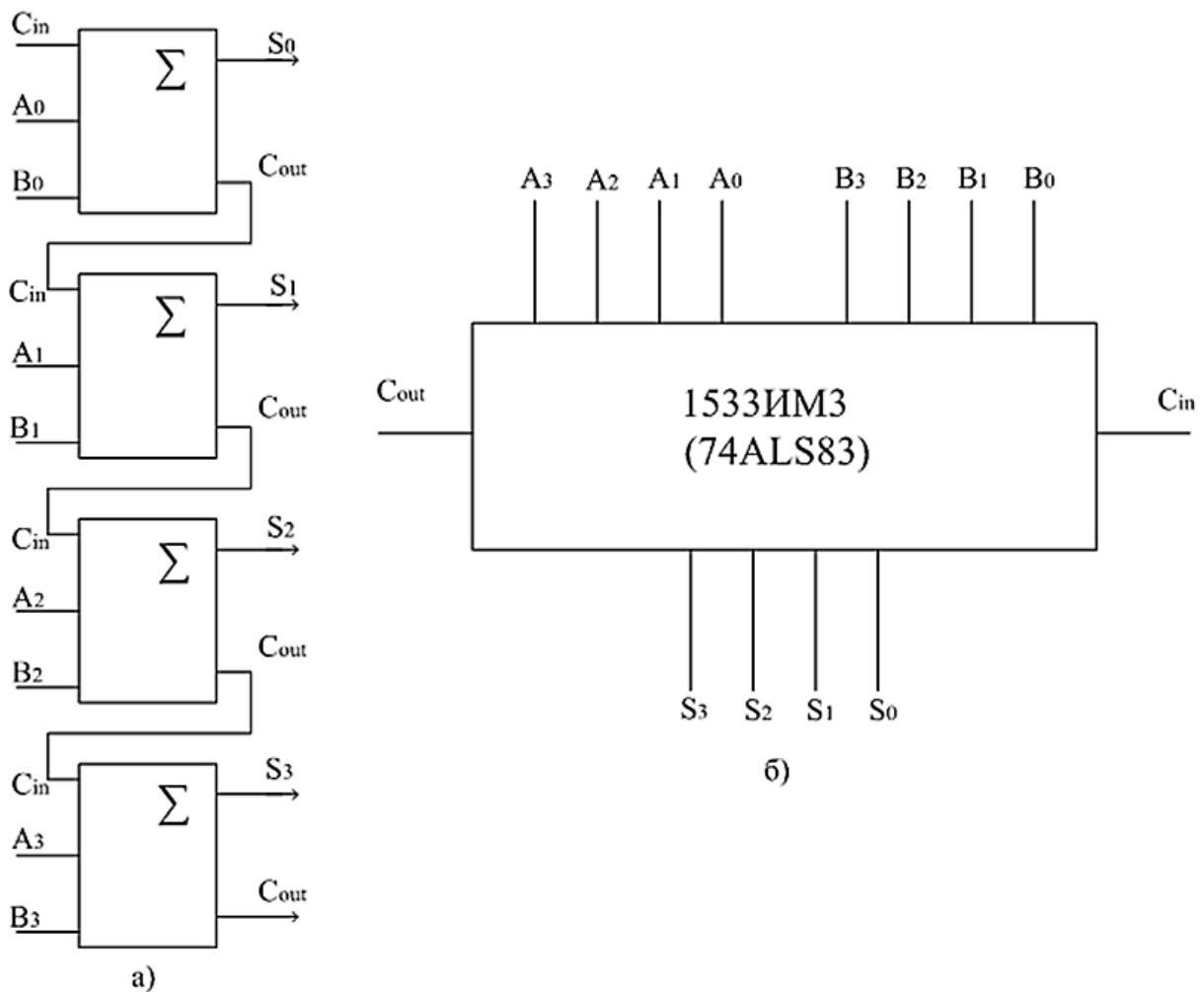


Рис. 3.5. Параллельный сумматор

## Вычитатели

**Полувывчитатель.** Логическая схема, которая осуществляет вычитание В(вычитаемое) из А(уменьшаемое), где А и В это однобитовые числа, называется полувывчитателем. Процесс вычитания может быть представлен с помощью таблицы истинности.

Таблица истинности полувывчитателя

Входы		Выходы	
A	B	D	C
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Здесь А и В это входы, а выходы D–разность и С - заем.  
Из таблицы истинности

$$D = \bar{A}B + A\bar{B} = A \oplus B; \quad (3.5)$$

$$C = \bar{A}B \quad (3.6)$$

В соответствии с выражениями (3.5) и (3.6) логическая схема полувывчитателя показана на рис.3.6.

Рис.3.6. Логическая схема полувывчитателя

**Полный вычитатель.** Подобно полному сумматору, мы нуждаемся в полном вычитателе, для выполнения многобитового вычитания, где заем из предыдущего разряда присутствует. Таким образом полный вычитатель имеет три входа:  $A_n$  (уменьшаемое),  $B_n$ (вычитаемое) и  $C_{n-1}$ (заем от предыдущего разряда) и два выхода  $D_n$  (разность) и  $C_n$  (заем).

Таблица истинности полного вычитателя

Входы			Выходы	
$A_n$	$B_n$	$C_{n-1}$	$D_n$	$C_n$
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Таблица истинности и карта Карно для выхода  $D_n$  точно такие же, как и для  $S_n$  в полном сумматоре, поэтому

$$D_n = A_n \oplus B_n \oplus C_{n-1} \quad (3.7)$$

Карта Карно для  $C_n$  показана на рис.3.7.

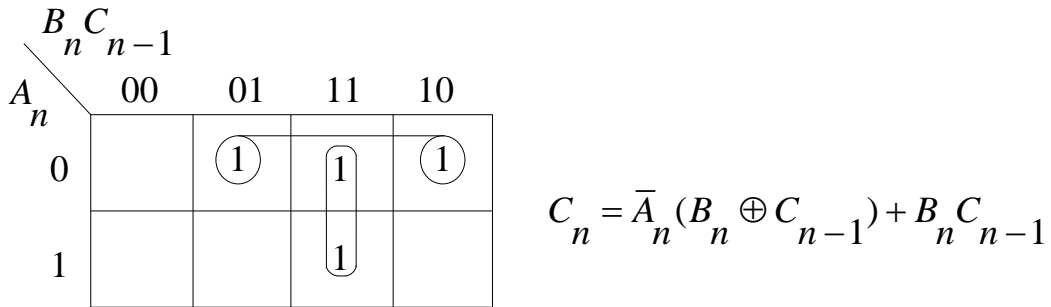


Рис.3.7. Упрощение для  $C_n$  с помощью карты Карно

Существует три варианта упрощения выражения для заем  $C_n$ . Два из них дают схему полного сумматора состоящего из двух полувычитателей и схемы ИЛИ. Однако наиболее интересным представляется третий вариант, рис. 3.7, выражение в котором

$$C_n = (\bar{A}_n \bar{B}_n C_{n-1} + \bar{A}_n B_n C_{n-1}) + (\bar{A}_n B_n C_{n-1} + A_n B_n C_{n-1}) = \bar{A}_n (B_n \oplus C_{n-1}) + B_n C_{n-1} \quad (3.8)$$

Логическая схема полного вычитателя для этого варианта показана на рис.3.8.

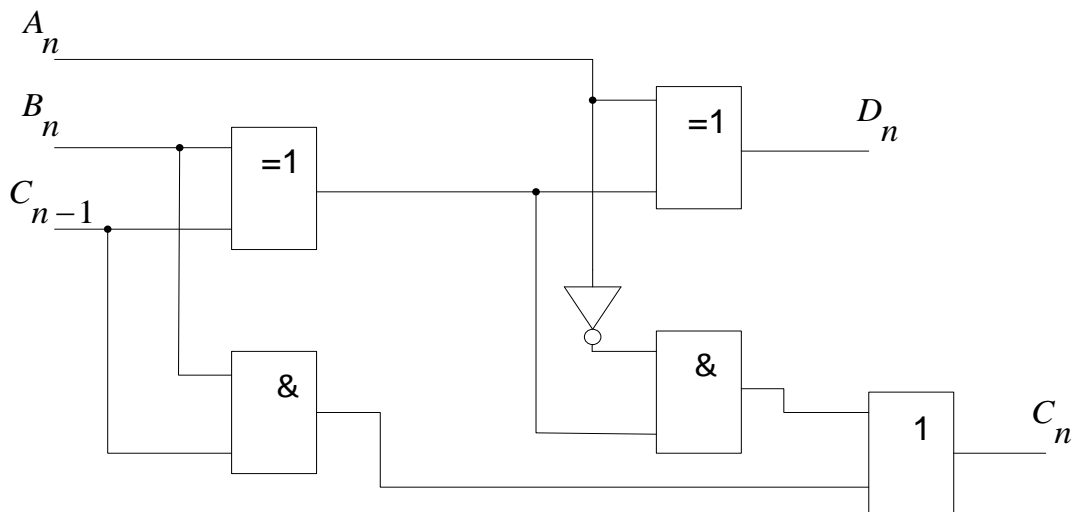


Рис.3.8. Полный вычитатель

Полный вычитатель рис.3.8, по сравнению с обычной схемой, не состоит из двух полувычитателей, а содержит на один логический элемент (инвертор) меньше, и время распространения сигнала через схемы на один элемент меньше, т.е. схема полного вычитателя рис.3.8 более быстродействующая по сравнению с обычной схемой полного вычитателя.

### Суммирование/Вычитание в системе счисления с дополнительным кодом.

Вычитание двоичных чисел может осуществляться с использованием полных двоичных вычитателей, однако этот метод используется редко. Взамен этого вычитания обычно используется арифметика в системе счисления с дополнительным кодом и тогда только двоичные сумматоры используются для выполнения операций суммирования и вычитания. Даже с учетом того, что потребуется еще и схема для получения дополнения, этот метод является предпочтительным.

Обычно используется комбинированная суммирующая/вычитающая схема, в которой вид выполняемой операции (суммирование или вычитание зависит от управляющего сигнала).

Рассмотрим схему рис.3.9, в которой параллельный сумматор соединен таким образом, чтобы выполнять или суммирование, или вычитание.

Когда управляющий сигнал имеет низкий уровень (лог.0) на входе  $C_{in}$ , то и на одном из входов схем ИСКЛЮЧАЮЩЕЕ ИЛИ также лог.0, поэтому биты числа В передаются на входы сумматора без изменения. Схема работает как сумматор.

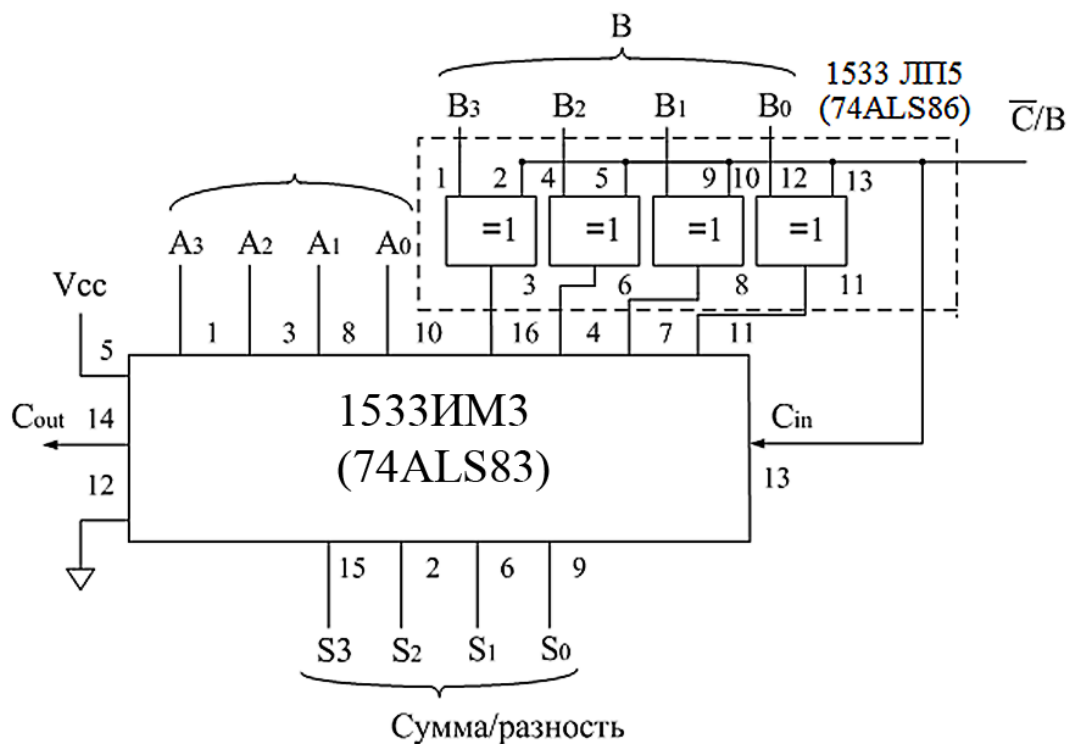


Рис.3.9. Сумматор / Вычитатель

Когда управляющий сигнал имеет высокий уровень, на вход  $C_{in}$  подается лог.1 и каждая схема ИСКЛЮЧАЮЩЕЕ ИЛИ инвертирует биты числа В. Таким образом осуществляется инвертирование каждого бита числа В и прибавление 1 (по входу  $C_{in}$ ) и в результате осуществляется операция дополнения с числом В. При этом суммирование числа А с дополнением числа В равносильно вычитанию числа В из числа А.

### 3. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

**Оборудование и компоненты:** универсальная лабораторная установка IDL-800, ИС 1533ЛП5(74ALS86)-четыре логических элемента ИСКЛЮЧАЮЩЕЕ ИЛИ, ИС 1533ЛИ1(74ALS08)-четыре логических элемента 2И, ИС 1533ЛЛ1(74ALS32)-четыре логических элемента 2ИЛИ, ИС 1533ИМ3(74ALS83)-четырёхразрядный сумматор, ИС 1533ЛН1 (74ALS04)-шесть инверторов.

#### 3.1. Исследование сумматоров.

##### *Полусумматор*

1. Используя логические элементы, собрать схему полусумматора, рис.3.2.
2. Изменяя состояния входов А и В, исследовать работу полусумматора, получить таблицу истинности полусумматора.

##### *Полный сумматор*

1. Используя логические элементы, собрать схему полного сумматора, рис.3.4.
2. Изменяя состояния входов  $A_n, B_n, C_{n-1}$ , исследовать работу полного сумматора, получить таблицу истинности полного сумматора.

#### 3.2. Исследование вычитателей

##### *Полувывчитатель*

1. Используя логические элементы, собрать схему полувывчитателя, рис.3.6.
2. Изменяя состояния входов А и В, исследовать работу полувывчитателя, получить таблицу истинности полувывчитателя.

##### *Полный вычитатель*

- 1.Используя логические элементы, собрать схему полного вычитателя, рис. 3.8.

2. Изменяя состояния входов  $A_n, B_n, C_{n-1}$ , исследовать работу полного вычитателя, получить таблицу истинности полного вычитателя.

### **3.3. Исследование суммирования/вычитания в дополнительном коде**

1. Собрать схему сумматора/вычитателя, рис.3.9.
2. Исследовать работу сумматора/вычитателя, используя несколько примеров. Числа А и В представляются в дополнительном коде.
3. Объяснить результаты экспериментов.

## **4. СОДЕРЖАНИЕ ОТЧЕТА**

1. Цель работы.
2. Схемы, исследуемые в работе.
3. Таблицы, отражающие результаты исследований.
4. Выводы по результатам исследований.

## **5. КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. Какие комбинационные схемы называются полу/полным сумматором (вычитателем)?
2. Нарисуйте схему полу/полного сумматора (вычитателя).
3. Напишите таблицу истинности полу/полного сумматора (вычитателя).
4. Примеры суммирования/вычитания в дополнительном коде.

## Лабораторная работа №4

### ИССЛЕДОВАНИЕ ПРЕОБРАЗОВАТЕЛЕЙ КОДОВ.

#### 1. ЦЕЛЬ ЛАБОРАТОРНОЙ РАБОТЫ

Синтез преобразователей двоичных кодов и исследование их функционирования.

#### 2. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Для представления информации, в цифровых устройствах используются двоичные коды. И хотя любой код, используемый для представления информации, имеет двоичную форму, однако правильная интерпретация этих кодов возможна только, если этот код известен. Наиболее часто используемые двоичные коды представлены в табл.4.1.

Таблица 4.1

Десятичные числа	Прямой двоичный код	BCD код	Код с избытком 3 (Excess3)	Код Грея
	$B_3 B_2 B_1 B_0$	$D C B A$	$E_3 E_2 E_1 E_0$	$G_3 G_2 G_1 G_0$
0	0 0 0 0	0 0 0 0	0 0 1 1	0 0 0 0
1	0 0 0 1	0 0 0 1	0 1 0 0	0 0 0 1
2	0 0 1 0	0 0 1 0	0 1 0 1	0 0 1 1
3	0 0 1 1	0 0 1 1	0 1 1 0	0 0 1 0
4	0 1 0 0	0 1 0 0	0 1 1 1	0 1 1 0
5	0 1 0 1	0 1 0 1	1 0 0 0	0 1 1 1
6	0 1 1 0	0 1 1 0	1 0 0 1	0 1 0 1
7	0 1 1 1	0 1 1 1	1 0 1 0	0 1 0 0
8	1 0 0 0	1 0 0 0	1 0 1 1	1 1 0 0
9	1 0 0 1	1 0 0 1	1 1 0 0	1 1 0 1
10	1 0 1 0			1 1 1 1
11	1 0 1 1			1 1 1 0
12	1 1 0 0			1 0 1 0
13	1 1 0 1			1 0 1 1
14	1 1 1 0			1 0 0 1
15	1 1 1 1			1 0 0 0

**Прямой двоичный код** использует для представления чисел в двоичной системе счисления. В таблице 2.1 прямой двоичный код представляет числа от 0 до 15.

**Двоично-десятичный 8-4-2-1 код (BCD код)** используется для представления десятичных цифр. Числа 8,4,2 и 1 являются весами разрядов. Запись десятичной цифры в коде 8-4-2-1 совпадает с записью двоичных чисел от 0 до 9, а  $n$ -разрядное десятичное число представляется с помощью тетрад, каждая из которых состоит из четырех двоичных разрядов (например,  $395_{10} = 0011\ 1001\ 0101$ ).

**Двоично-десятичный код с избытком 3 (Excess-3 код).**

Excess-3 код (также используемый для представления десятичных цифр) образуется от соответствующих представлений цифр в BCD коде путем прибавления двоичного числа 0011. Код с избытком 3 является самодополняющим кодом. Правила преобразования прямого кода с избытком 3 в дополнительный с избытком 3 и правила обратного преобразования такие же, как и для двоичного дополнительного кода. Поэтому код с избытком 3 часто удобнее использовать для выполнения арифметических операций. При этом для сложения четырехразрядных кодов можно использовать четырехразрядные двоичные сумматоры.

**Код Грея.**

В коде Грея десятичные числа представлены в двоичном виде таким образом, что представление каждого числа отличается от предыдущего как и от последующего только в одном бите (разряде).

Код Грея является зеркальным и может быть построен следующим образом:

1. Одноразрядный код Грея состоит из кодовых слов, 0 и 1, которые представляют десятичные числа 0 и 1.
2. В  $n$ -разрядном ( $n \geq 2$ ) коде Грея, первые  $2^{n-1}$  кодовых слов повторяют  $(n-1)$  разрядный код Грея с приписанным 0 в крайнем левом разряде.
3. Последние  $2^{n-1}$  кодовые слова представляют кодовые слова  $(n-1)$  разрядного кода Грея, записанными в обратном порядке (как будто между первыми  $2^{n-1}$  и последними  $2^{n-1}$  кодовыми словами размещено зеркало) и дописанной 1 в крайнем левом разряде.

Например, определим 1-разрядный, двухразрядный и трехразрядный код Грея.

Для одноразрядного кода Грея

Десятичные числа	Код Грея
0	0
1	1

Для двухразрядного кода Грея

Десятичные числа	Код Грея
0	00
1	01 ← <i>зеркало</i>
2	11
3	10

### Для трехразрядного кода Грея

Десятичные числа	Код Грея
0	000
1	00 <u>1</u>
2	011
3	<u>010</u> ← — <i>зеркало</i>
4	110
5	111
6	101
7	100

Очень часто в цифровых системах возникает задача преобразования двоичной информации, представленной в одном двоичном коде в другой код.

### Преобразователь двоичного кода в код Грея

Блок-диаграмма такого преобразователя на четыре разряда показана на рис.4.1

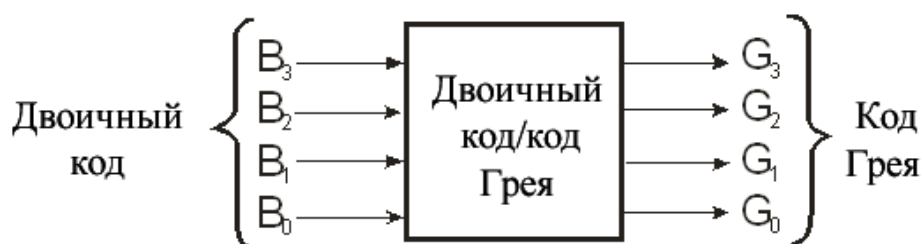


Рис. 4.1 Блок-диаграмма преобразователя двоичного кода в код Грея

Таблица истинности такого преобразователя представлена табл.4.2.

Таблица 4.2

Двоичный код				Код Грея			
$B_3$	$B_2$	$B_1$	$B_0$	$G_3$	$G_2$	$G_1$	$G_0$
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1

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

Упростим выражения каждого выхода преобразователя с помощью карт Карно, как показано на рис.4.2

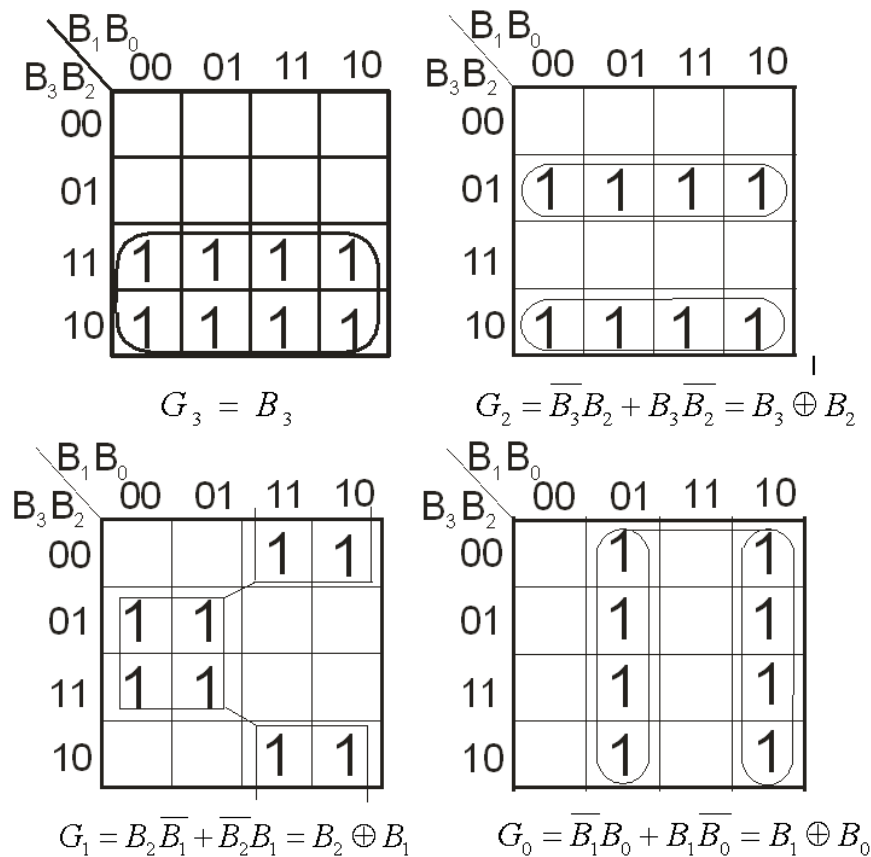


Рис. 4.2. Карты Карно преобразователя двоичного кода в код Грея.

На основании выражений для  $G_3 = B_3$ ,  $G_2 = B_3 \oplus B_2$ ,  $G_1 = B_2 \oplus B_1$  и  $G_0 = B_1 \oplus B_0$  может быть построена логическая схема четырехразрядного преобразователя двоичного кода в код Грея, рис.4.3. Эта схема может быть обобщена на любое число разрядов.

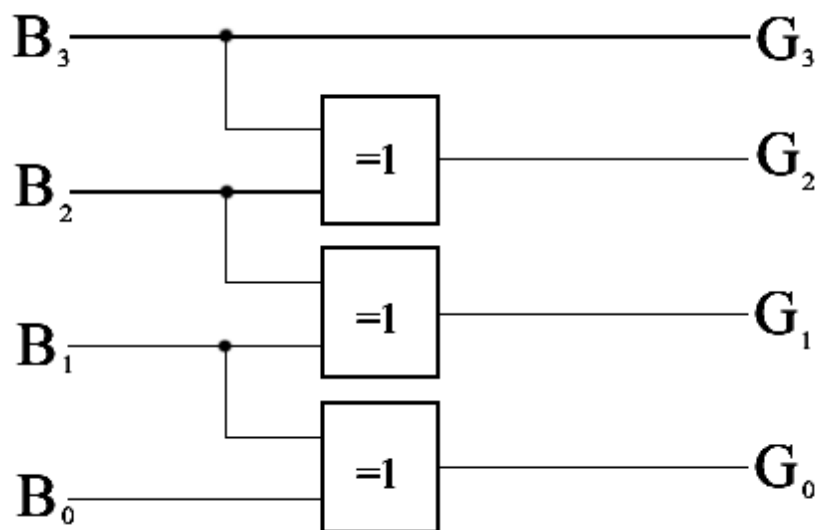


Рис. 4.3 Преобразователь двоичного кода в код Грея.

Из логической схемы преобразователя двоичного кода в код Грея вытекает и общее правило для преобразования любого двоичного кода в соответствующий код Грея, показанное на рис.4.4

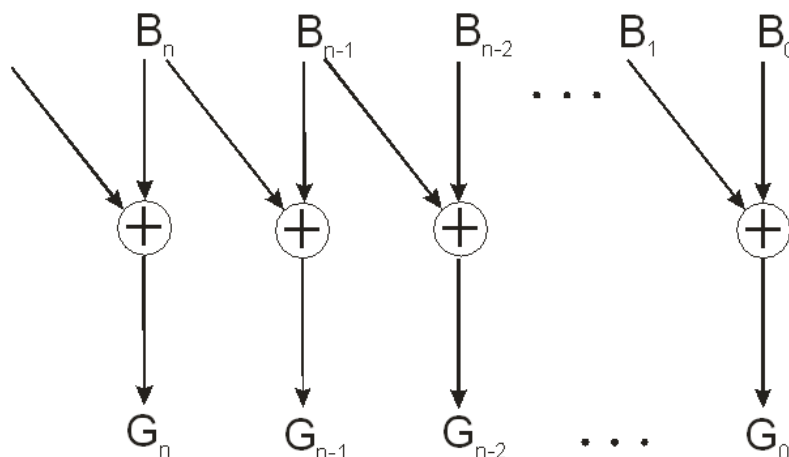


Рис. 4.4. Общая схема преобразования двоичного числа в код Грея.

### Преобразователь кода Грея в двоичный код

Используя таблицу 4.2, можно с помощью карт Карно упростить выражения для этого преобразователя. Только в данном случае входными переменными будут  $G_3, G_2, G_1, G_0$ , а выходными  $B_3, B_2, B_1, B_0$ , рис.4.5.

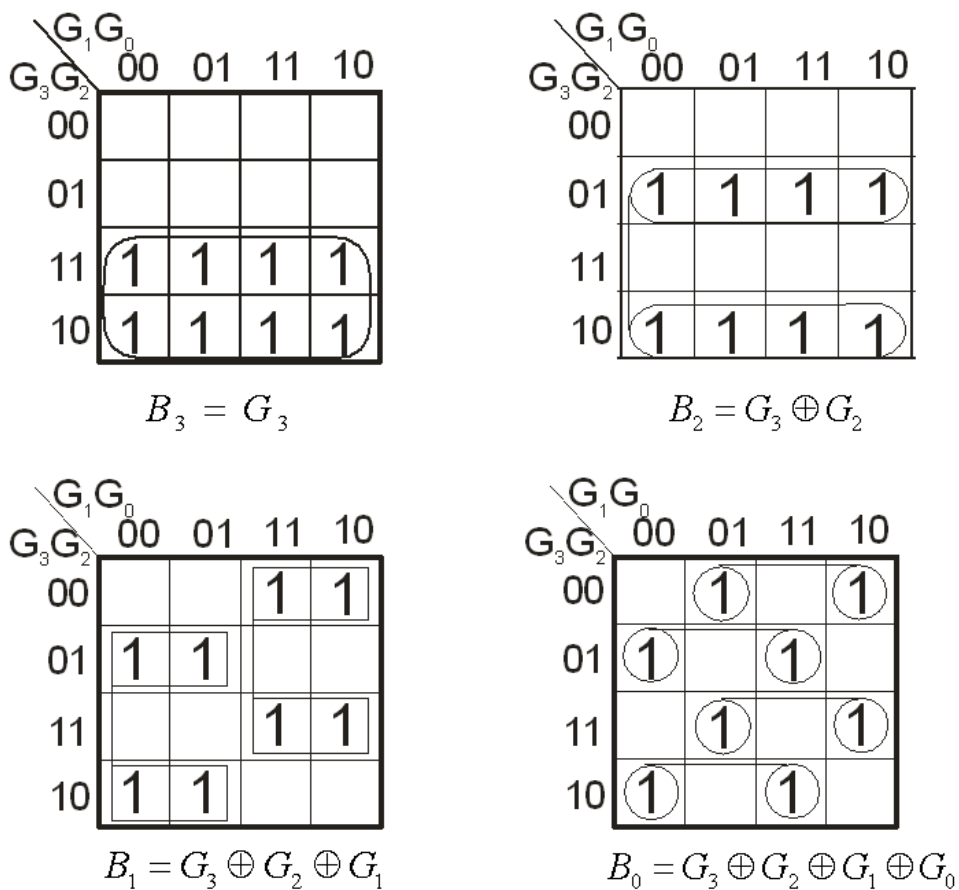


Рис. 4.5. Карты Карно для преобразователя кода Грея в двоичный код.

В соответствии с выражениями, полученными из карт Карно, логическая схема преобразователя построена на рис.4.6а.

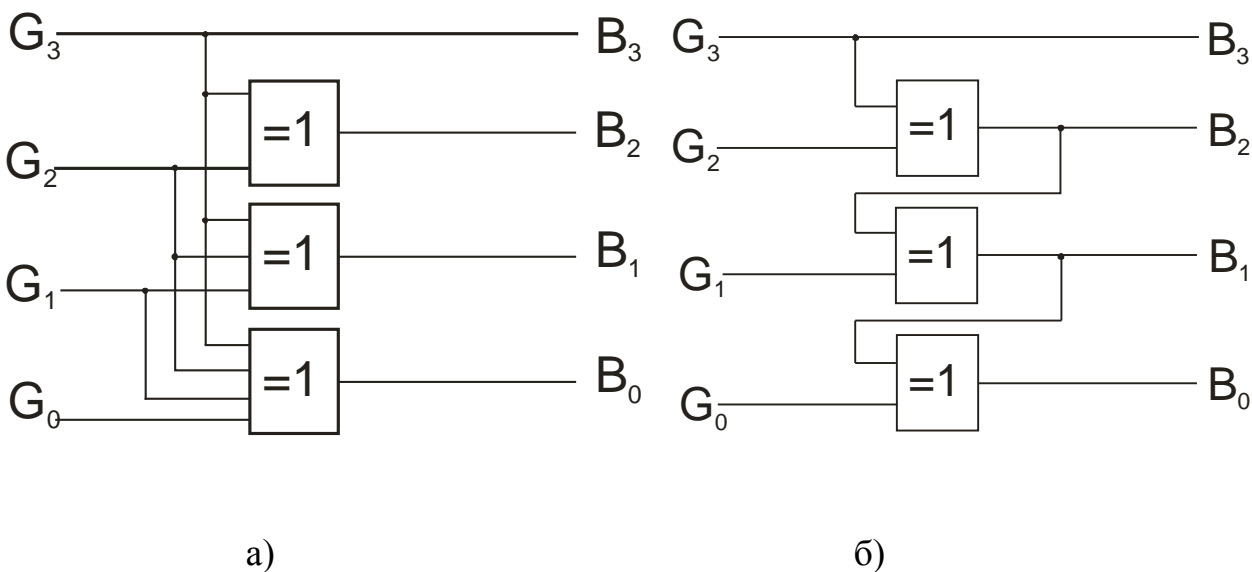


Рис. 4.6. Преобразователь кода Грея в двоичный код.

Логическая схема рис.4.6а сложнее логической схемы рис.4.5. Однако она может быть упрощена, если переписать выражения для  $B_3, B_2, B_1, B_0$  следующим образом:

$$\begin{aligned}
 B_3 &= G_3 \\
 B_2 &= G_3 \oplus G_2 \\
 B_1 &= G_3 \oplus G_2 \oplus G_1 = B_2 \oplus G_1 \\
 B_0 &= G_3 \oplus G_2 \oplus G_1 \oplus G_0 = B_1 \oplus G_0
 \end{aligned}
 \tag{4.1}$$

Логическая схема, построенная по этим выражениям, рис.4.6б не сложнее, чем логическая схема рис.4.5, но необходимо иметь в виду, что быстродействие схемы, рис. 4.6б более низкое по сравнению с другими схемами.

### Преобразователь VCD кода в прямой двоичный код

Рассмотрим схему преобразования VCD кода в двоичный код показанную на рис.4.7. Входами являются две тетрады:  $D_0 C_0 B_0 A_0$  представляющая единицы и  $D_1 C_1 B_1 A_1$  представляющая десятки. Выходом является семиразрядный двоичный код  $b_6 b_5 b_4 b_3 b_2 b_1 b_0$ . На схеме показаны также вес каждого VCD входа и каждого двоичного выхода. Разряды в двоично-десятичном представлении имеют десятичный вес 8,4,2,1 в каждой тетраде, однако каждая тетрада отличается от предыдущей весовым коэффициентом-10(одна десятичная цифра от предыдущей).



Рис. 4.7. Схема преобразования двухразрядного VCD кода в прямой двоичный код

Десятичные веса каждого бита в двоично-десятичном представлении, рис.4.7, могут быть представлены двоичными эквивалентами, как приведено в табл.4.3. Используя эти веса, можно осуществить преобразование двоично-

десятичного кода в двоичный путем двоичного суммирования двоичных эквивалентов тех бит, которые в BCD представлении равны единицам.

Таблица 4.3.

BCD биты	Десятичные веса	Двоичные эквиваленты						
		$B_6$	$B_5$	$B_4$	$B_3$	$B_2$	$B_1$	$B_0$
$A_0$	1	0	0	0	0	0	0	1
$B_0$	2	0	0	0	0	0	1	0
$C_0$	4	0	0	0	0	1	0	0
$D_0$	8	0	0	0	1	0	0	0
$A_1$	10	0	0	0	1	0	1	0
$B_1$	20	0	0	1	0	1	0	0
$C_1$	40	0	1	0	1	0	0	0
$D_1$	80	1	0	1	0	0	0	0

Например, преобразуем 01010101 (двоично-десятичное представление десятичного 55) в двоичный эквивалент. Для этого запишем двоичные эквиваленты для всех «1» в двоично-десятичном представлении, а затем просуммируем их.

$$\begin{array}{r}
 01010101 \\
 \begin{array}{l}
 \rightarrow \\
 \rightarrow \\
 \rightarrow \\
 \rightarrow \\
 \rightarrow
 \end{array}
 \end{array}
 \begin{array}{r}
 \text{(BCD)} \\
 0000001 \text{ (двоичный код 1)} \\
 0000100 \text{ (двоичный код 4)} \\
 0001010 \text{ (двоичный код 10)} \\
 + 0101000 \text{ (двоичный код 40)} \\
 \hline
 110111 \text{ (двоичный код 55)}
 \end{array}$$

Отсюда следует один из способов реализации преобразователя двоично-десятичного BCD кода в прямой двоичный с помощью двоичных сумматоров. На рис.4.8 показана схемная реализация такого преобразователя с помощью двух четырехразрядных двоичных сумматоров 1533ИМ3(74ASL83).

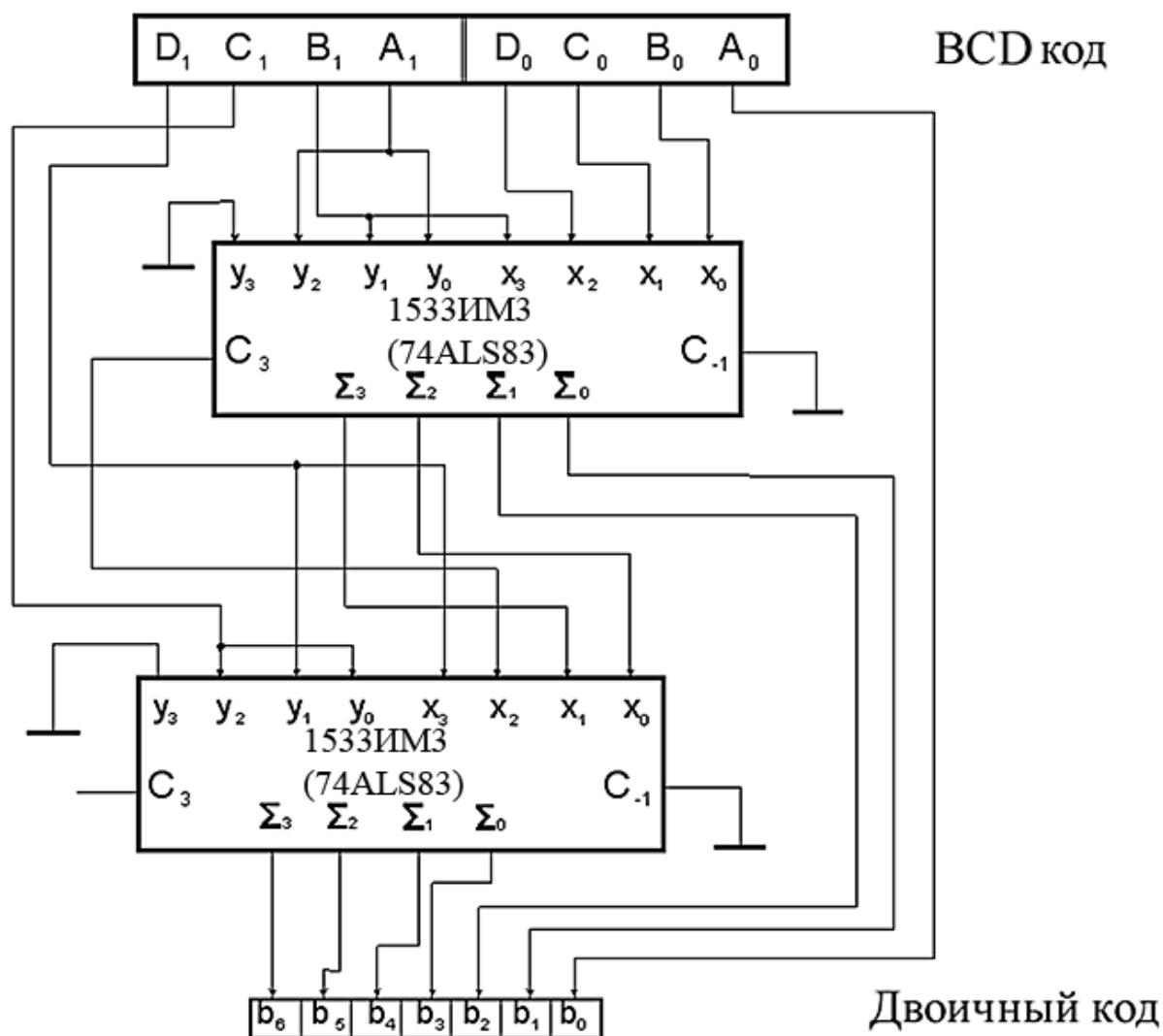


Рис. 4.8. Преобразователь двоично-десятичного в двоичный код.

### 3. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

**Оборудование и компоненты:** универсальная лабораторная установка IDL-800, ИС 1533ЛП5(74ALS86)–четыре двухвходовых логических элемента ИСКЛЮЧАЮЩЕЕ ИЛИ, ИС 1533ИМЗ(74ALS83)–четырёхразрядный двоичный сумматор.

#### 3.1 Исследование преобразователя BCD кода в двоично-десятичный код с избытком 3(Excess-3 код).

1. Разработать и реализовать схему преобразователя, используя четырехразрядный двоичный сумматор.
2. Проверить работу преобразователя.

### **3.2. Исследование преобразователя двоично-десятичного кода с избытком 3 в BCD код.**

1. Разработать и реализовать схему преобразователя используя четырех-разрядный двоичный сумматор.
2. Проверить работу преобразователя.

### **3.3. Исследование преобразователя двоичного кода в код Грея.**

1. Собрать, используя логические элементы ИСКЛЮЧАЮЩЕЕ ИЛИ, схему преобразователя.
2. Исследовать работу преобразователя.

### **3.4. Исследование преобразователя кода Грея в двоичный код.**

1. Собрать схему преобразователя используя логические элементы ИСКЛЮЧАЮЩЕЕ ИЛИ.
2. Исследовать работу преобразователя.

### **3.5. Исследование преобразователя двоично-десятичного BCD кода в двоичный код.**

1. Собрать схему преобразователя, рис.4.8.
2. Исследовать работу преобразователя.

## **4. СОДЕРЖАНИЕ ОТЧЕТА**

1. Цель работы.
2. Схемы, исследуемые в работе.
3. Таблицы, отражающие результаты исследований.
4. Выводы по результатам исследований.

## **5. КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. Объяснить работу преобразователей кодов, исследуемых в работе.

## Лабораторная работа №5

# СИНТЕЗ КОМБИНАЦИОННЫХ СХЕМ С ИСПОЛЬЗОВАНИЕМ МУЛЬТИПЛЕКСОРОВ И ДЕМУЛЬТИПЛЕКСОРОВ

## 1. ЦЕЛЬ ЛАБОРАТОРНОЙ РАБОТЫ

Целью работы является изучение мультиплексоров и демультимплексоров, а также синтеза комбинационных схем на основе мультиплексоров и на основе демультимплексоров.

## 2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Традиционный синтез комбинационных схем (КС) включает в себя минимизацию ФАЛ и реализацию минимальной ФАЛ с помощью логических элементов. Используя этот метод, некоторые КС схемы были синтезированы и реализованы как отдельные ИС. Среди них мультиплексоры и демультимплексоры, которые широко представлены как ИС средней степени интеграции. Мультиплексоры и демультимплексоры могут успешно использоваться для реализации различных комбинационных устройств. При этом уменьшается количество требуемых ИС, повышается надежность и снижается стоимость реализации КС.

### Мультиплексоры

Мультиплексор (или селектор данных) это комбинационная схема, которая коммутирует один из  $2^m$  входных сигналов на один выход. Выбор информационного входа, который коммутируется на выход, осуществляется с помощью  $m$  адресных входов. Условные обозначения мультиплексора показаны на рис. 5.1.

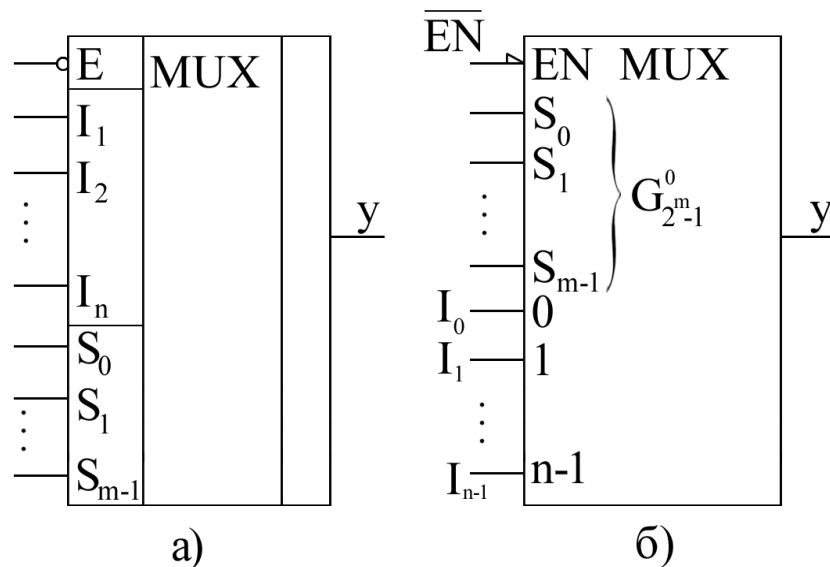


Рис. 5.1. Условное обозначение мультиплексора:

а) используемое в отечественной справочной литературе;

б) рекомендуемое к использованию международным стандартом IEEE

Для выбора одного из  $n$  входов, коммутируемого на выход, требуется группа из  $m$  адресных входов, где  $2^m = n$ . В зависимости от цифрового кода на адресных входах, один из входов выбирается и соединяется с выходом. Обычно стробируемый (G) или разрешающий (EN) вход используется для каскадного соединения мультиплексоров и, который обычно активный при низком уровне, т.е. разрешает работу мультиплексора, когда сигнал на этом входе низкий, лог. 0.

Число информационных входов, коммутируемых на выход  $Y$ , составляет  $n = 2, 4, 8, 16$ . При  $n = 4$  мультиплексор имеет размерность 4:1. Это 4-канальный одноразрядный мультиплексор, на выход которого передается один из четырех входных сигналов. Рассмотрим, как построить мультиплексор 4:1, таблица истинности которого приведена в таблице 5.1.

Таблица 5.1

Адресные входы		Выход
$S_1$	$S_0$	$Y$
0	0	$I_0$
0	1	$I_1$
1	0	$I_2$
1	1	$I_3$

Выход  $Y$  может быть записан как

$$Y = (\overline{S_1} \cdot \overline{S_0} \cdot I_0 + \overline{S_1} \cdot S_0 \cdot I_1 + S_1 \cdot \overline{S_0} \cdot I_2 + S_1 \cdot S_0 \cdot I_3) \quad (5.1)$$

Выражение (5.1) преобразуем, используя двойную инверсию и закон Де Моргана

$$Y = \overline{\overline{S_1 S_0 I_0} \cdot \overline{S_1 S_0 I_1} \cdot \overline{S_1 S_0 I_2} \cdot \overline{S_1 S_0 I_3}} \quad (5.2)$$

Выражение (5.2) реализуется, как показано на рис. 5.2.

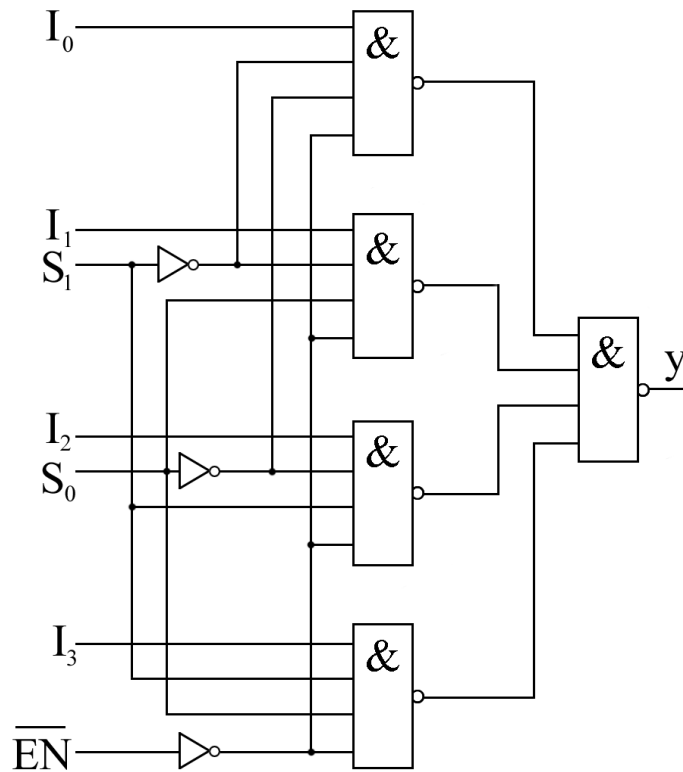


Рис. 5.2. Логическая схема мультиплексора 4:1 с разрешающим входом  $\overline{EN}$

Промышленностью выпускаются мультиплексоры размерностью 8:1 и 16:1 со стробирующим входом и без него. Также выпускаются двухразрядные мультиплексоры 4:1 и четырехразрядные мультиплексоры 2:1.

Мультиплексоры могут быть использованы как логические элементы для синтеза комбинационных схем. Использование мультиплексоров дает следующие преимущества:

- 1) Не требуется упрощение ФАЛ;
- 2) Минимизируется число требуемых интегральных схем;
- 3) Синтез КС упрощается.

Для реализации КС с использованием мультиплексора требуется представление ФАЛ таблицей истинности или в СДНФ, или в СКНФ. Синтез КС сводится к следующему:

- 1) Определяются десятичные номера каждого минтерма ФАЛ и входы мультиплексора, соответствующие этим номерам соединяются с лог. 1.
- 2) Все остальные входы соединяются с лог. 0.
- 3) Входные переменные ФАЛ подаются на адресные входы.

**Пример 5.1.** Реализовать следующую ФАЛ, используя мультиплексор

$$f(x_1, x_2, x_3, x_4) = \sum m(0,1,2,5,7,8,11,14)$$

**Решение.** ФАЛ четырех переменных, поэтому мультиплексор должен быть с четырьмя адресными входами, рис. 5.3.

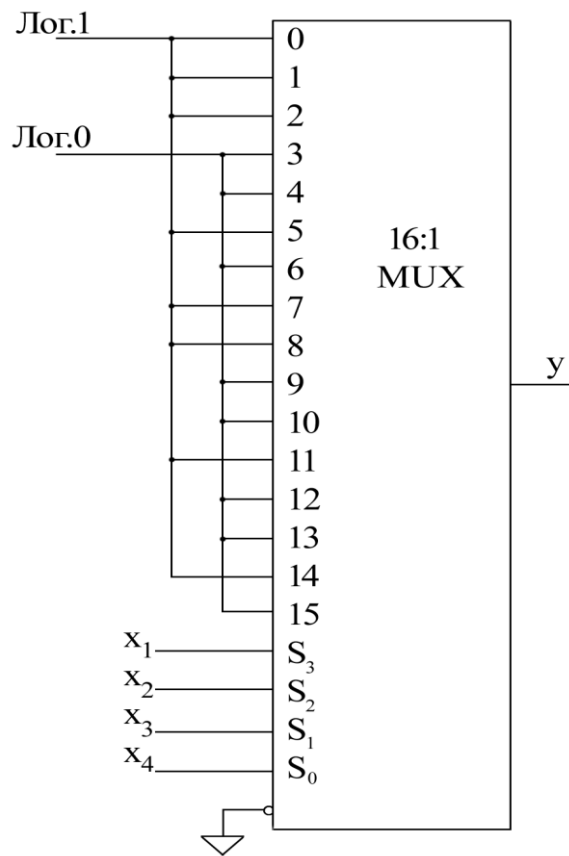


Рис. 5.3 Реализация ФАЛ  $f(x_1, x_2, x_3, x_4)$  из примера 5.1

### Реализация функции $m+1$ переменных на мультиплексорах с $m$ адресными входами.

Рассмотрим это на примере.

**Пример 5.2.** Реализовать ФАЛ, заданную таблицей истинности, таблица 5.2.

ФАЛ четырех переменных может быть реализована, используя мультиплексор размерностью 8:1, рис. 5.4. Переменные ФАЛ  $x_1, x_2, x_3$  подсоединим к адресным входам  $S_2, S_1, S_0$  соответственно. Переменную, не подсоединенную к адресным входам, называют «выделенной». Без выделенной переменной наборы переменных  $x_1, x_2, x_3$  образуют пары. В таблице истинности эти пары выделены пунктирными линиями.

Теперь рассмотрим соотношения между выделенной переменной  $x_4$  и выходом для каждой пары. При этом возможны четыре варианта, когда выход  $Y$  не зависит от переменной  $x_4$  и равен 0 или 1, и когда выход  $Y$  зависит от

Таблица 5.2

Входы				Выход	
$x_1$	$x_2$	$x_3$	$x_4$	$Y$	
0	0	0	0	1	1
0	0	0	1	1	1
0	0	1	0	0	$x_4$
0	0	1	1	1	$x_4$
0	1	0	0	0	1
0	1	0	1	1	1
0	1	1	0	1	1
0	1	1	1	1	1
1	0	0	0	0	0
1	0	0	1	0	0
1	0	1	0	1	$\bar{x}_4$
1	0	1	1	0	$\bar{x}_4$
1	1	0	0	0	$x_4$
1	1	0	1	1	$x_4$
1	1	1	0	1	$\bar{x}_4$
1	1	1	1	0	$\bar{x}_4$

переменной  $x_4$  и равен  $x_4$  или  $\bar{x}_4$ , как отмечено справа от таблицы истинности. Исходя из этого, на информационные входы мультиплексора и подается лог. 0, лог. 1,  $x_4$  или  $\bar{x}_4$ , как показано на рис.5.4.

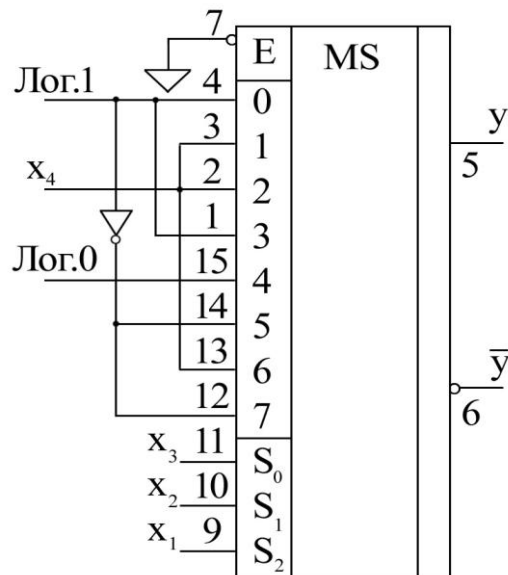


Рис. 5.4. Реализация ФАЛ, пример 5.2, с использованием мультиплексора КР1533КП7(74ALS151)



**Пример 5.3.** Построить мультиплексор размерностью 32:1. Такой мультиплексор может быть построен, как показано на рис. 5.6, используя два мультиплексора 16:1.

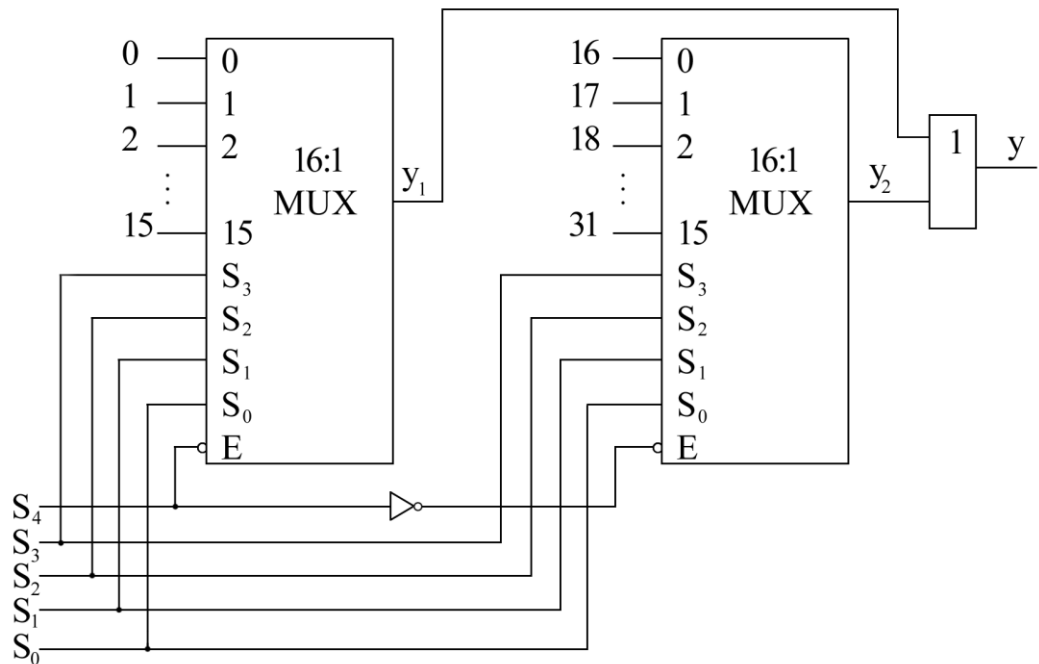


Рис. 5.6. Мультиплексор 32:1, используя два мультиплексора 16:1

**Пример 5.4.** Построить мультиплексор размерностью 256:1. Такой мультиплексор может быть построен по древовидной схеме, используя 17 мультиплексоров 16:1, рис. 5.7.

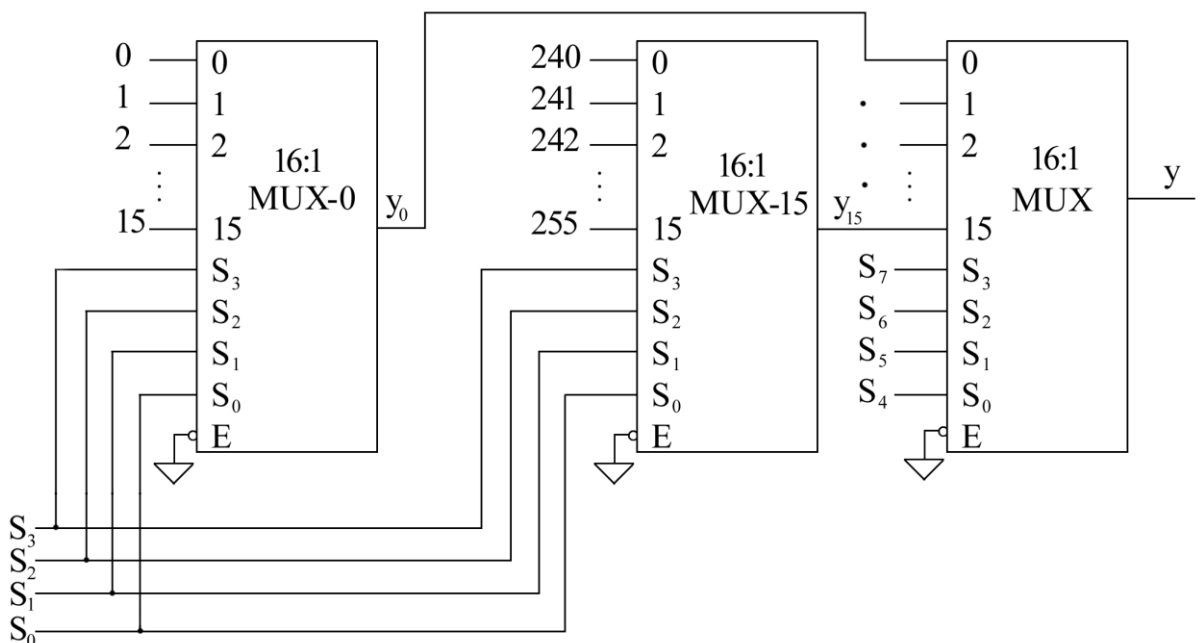


Рис. 5.7. Мультиплексор 256:1, используя 17 мультиплексоров 16:1

## Декодеры/Демультимплексоры и их использование в синтезе комбинационных схем

В интегральном исполнении декодеры (дешифраторы) реализуются с использованием элементов И-НЕ, и поэтому на выходах таких декодеров генерируются минтермы в инверсном виде. Большинство интегральных декодеров имеют один или несколько стробирующих или разрешающих входов. Схема декодера 2:4 с разрешающим входом, построенная на элементах И-НЕ, показана на рис. 5.8.

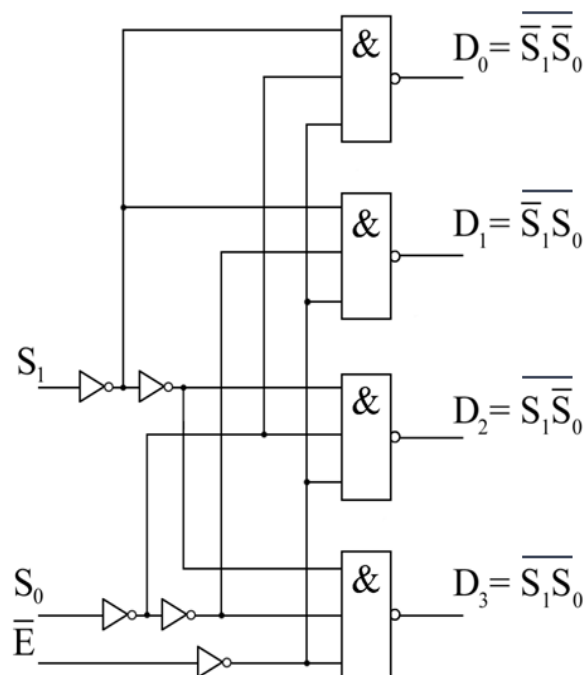


Рис. 5.8. Логическая схема декодера 2:4 с разрешающим входом  $\bar{E}$

Таблица истинности декодера 2:4 дана в таблице 5.3.

Если разрешающий вход  $\bar{E} = 1$ , то все выходы декодера равны 1 независимо от значения входов  $S_1$  и  $S_0$ . Когда разрешающий вход  $\bar{E} = 0$ , схема работает, как декодер с инверсными выходами. Условное обозначение схемы показано на рис. 5.9.

Таблица 5.3

$\bar{E}$	$S_1$	$S_0$	$D_0$	$D_1$	$D_2$	$D_3$
1	x	x	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0

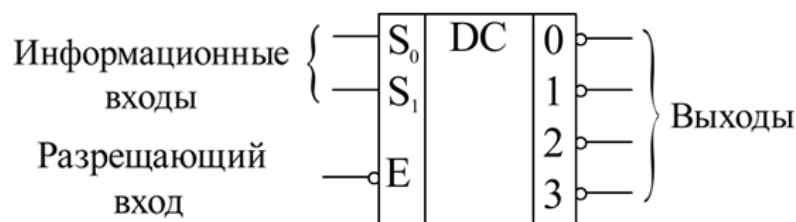


Рис. 5.9. Условное обозначение декодера 2:4

Декодер с разрешающим входом может работать как демультиплексор. Демультиплексор это комбинационная схема, которая принимает информацию на единственный вход и передает эту информацию на один из  $n = 2^m$  возможных выходов. Выбор определенного выхода осуществляется с помощью адресных входов. Декодер, который мы рассмотрели, может работать как демультиплексор, если вход  $\bar{E}$  использовать как информационный вход, а информационные входы декодера  $S_1$  и  $S_0$  использовать как адресные входы. Условное обозначение демультиплексора показано на рис. 5.10.

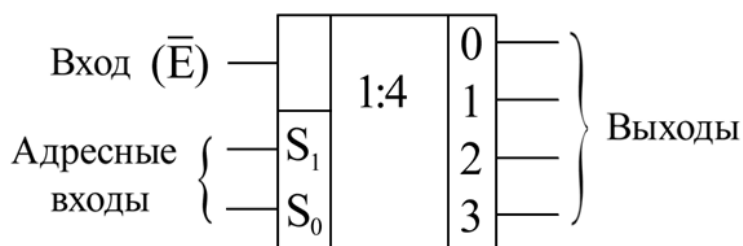


Рис. 5.10. Условное обозначение схемы рис. 5.9, используемой как демультиплексор

Информация с единственного входа демультиплексора поступает на один из выходов, в зависимости от двоичных значений на адресных входах. Это можно проверить, используя таблицу истинности, таблица 5.3.

Поскольку для декодера и демультиплексора используется одна и та же логическая схема, декодер с разрешающим входом называется **декодером/демультиплексором**. Демультиплексор осуществляет операцию обратную мультиплексору.

**Декодеры/демультиплексоры** используются в синтезе комбинационных схем. Особенно эти устройства полезны при синтезе КС с несколькими выходами. Декодеры/демультиплексоры в интегральном исполнении возможны как 2:4, 3:8 и 4:16 линий. Выходы таких устройств, как правило, имеют низкий активный уровень.

Рассмотрим использование декодеров/демультиплексоров в синтезе комбинационных схем.

**Пример. 5.5.** Реализовать схему полного сумматора, используя декодер/демультиплексор.

**Решение.** Из таблицы истинности для полного сумматора

$$S_n = \overline{A_n} \cdot \overline{B_n} \cdot C_{n-1} + \overline{A_n} \cdot B_n \cdot \overline{C_{n-1}} + A_n \cdot \overline{B_n} \cdot \overline{C_{n-1}} + A_n \cdot B_n \cdot C_{n-1}$$

$$C_n = \overline{A_n} \cdot B_n \cdot C_{n-1} + A_n \cdot \overline{B_n} \cdot C_{n-1} + A_n \cdot B_n \cdot \overline{C_{n-1}} + A_n \cdot B_n \cdot C_{n-1}$$

Используя двойную инверсию и закон Де Моргана, преобразуем эти выражения

$$S_n = \overline{\overline{\overline{A_n} \cdot \overline{B_n} \cdot C_{n-1}} \cdot \overline{\overline{\overline{A_n} \cdot B_n \cdot \overline{C_{n-1}}}} \cdot \overline{\overline{\overline{A_n} \cdot \overline{B_n} \cdot \overline{C_{n-1}}}} \cdot \overline{\overline{\overline{A_n} \cdot B_n \cdot C_{n-1}}}} \quad (5.3)$$

$$C_n = \overline{\overline{\overline{A_n} \cdot B_n \cdot C_{n-1}} \cdot \overline{\overline{\overline{A_n} \cdot \overline{B_n} \cdot C_{n-1}}}} \cdot \overline{\overline{\overline{A_n} \cdot B_n \cdot \overline{C_{n-1}}}} \cdot \overline{\overline{\overline{A_n} \cdot B_n \cdot C_{n-1}}}} \quad (5.4)$$

Или функция  $S_n$  может быть записана как  $S_n = \overline{D_1 \cdot D_2 \cdot D_4 \cdot D_7}$ , а функция  $C_n$  может быть записана как  $C_n = \overline{D_3 \cdot D_5 \cdot D_6 \cdot D_7}$ .

Для реализации полного сумматора необходимо использовать декодер/демультиплексор 3:8 линий, например КР1533ИД7(SN74ALS138). Схема полного сумматора показана на рис. 5.12.

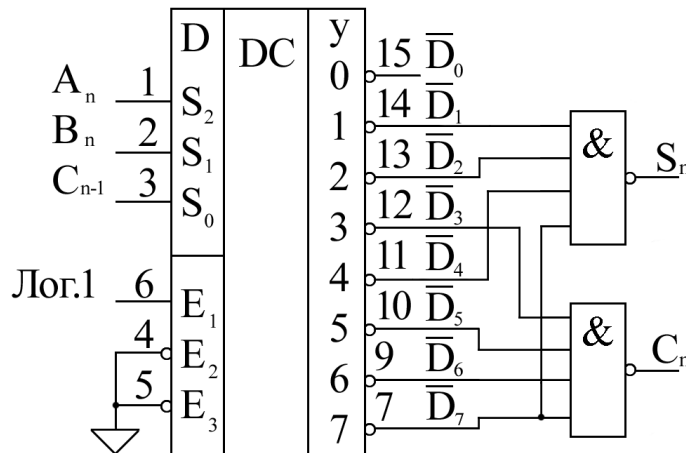


Рис. 5.12. Реализация полного сумматора с использованием декодера/демультиплексора

При синтезе КС с использованием мультиплексоров дополнительные элементы не используются, в то время как их необходимо использовать при синтезе КС на основе декодеров/демультиплексоров. Тем не менее, несмотря на это, декодеры/демультиплексоры являются более экономичными в случаях синтеза многовыходовых КС. В таких случаях один мультиплексор требуется для реализации каждого выхода, а при использовании декодера/демультиплексора только дополнительные схемы И-НЕ.

### Увеличение размерности декодеров/демультиплексоров

Поскольку ИС КР1533ИД3 является самым большим (4:16 линий) декодером/демультиплексором, то для увеличения размерности декодеров/демультиплексоров используются методы, показанные на рис. 5.13 и рис. 5.14.

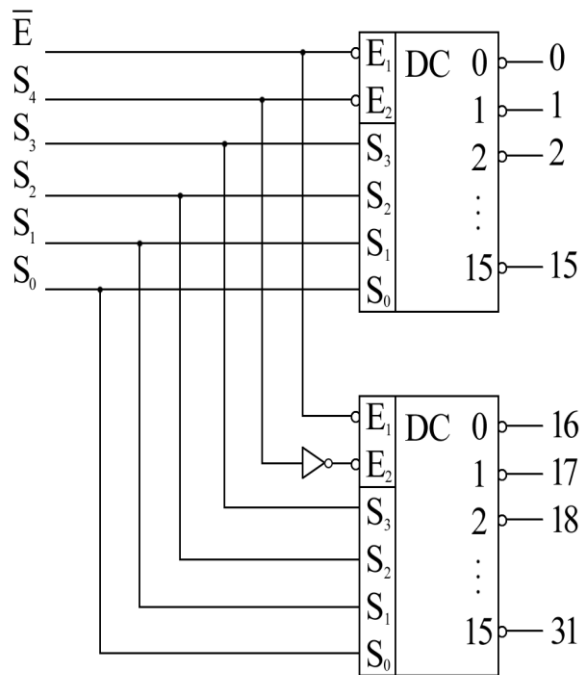


Рис. 5.13. Построение 5:32 линии декодера/демультиплексора, используя два 4:16 линии декодера/демультиплексора

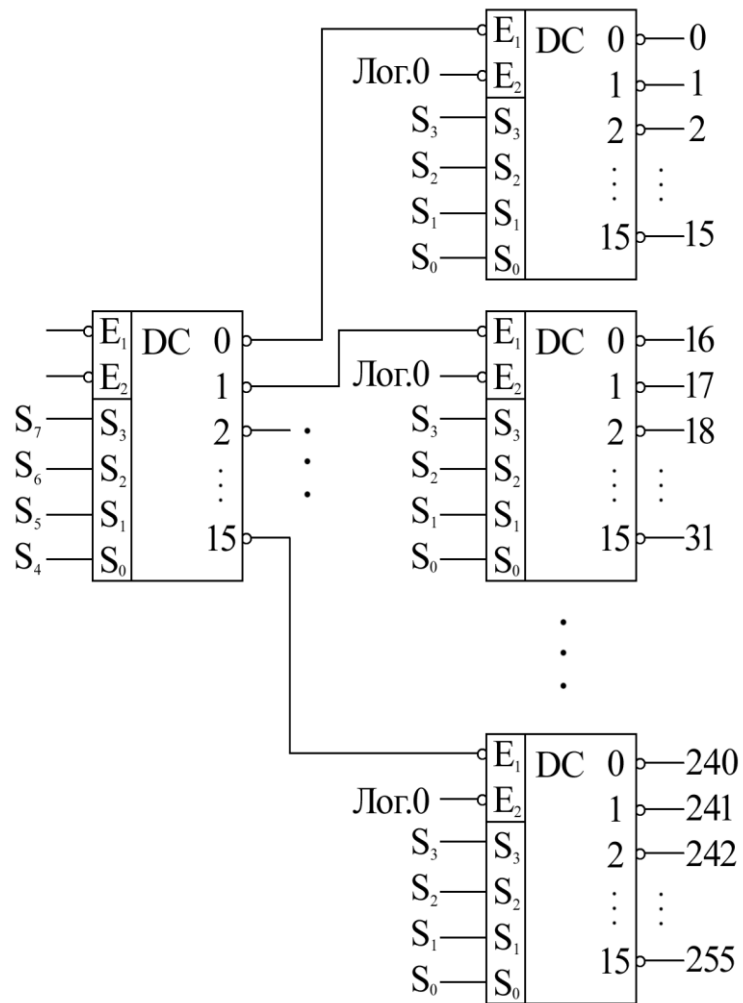


Рис. 5.14. Построение 8:256 линий декодера/демультиплексора, используя 17 декодеров/демультиплексоров 4:16 линий

### 3. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

**Оборудование и компоненты:** универсальная лабораторная установка IDL-800, ИС 1533КП7(74ALS151), ИС 1533ИД7(74ALS138), ИС 1533ЛА1(74ALS20), ИС 1533ЛН1(74ALS04).

#### 3.1. Исследование работы мультиплексора.

1. Установить ИС 1533КП7 на наборной панели IDL-800.
2. Вывод 16 ИС соединить с источником питания +5V, а вывод 8 с общей шиной установки.
3. Изменяя значения адресных входов от 000 до 111 и входные данные (0,1), проследить за состоянием выходов мультиплексора и проверить работу мультиплексора. Результаты отразить в таблице истинности мультиплексора.

#### 3.2. Синтезировать и реализовать, используя мультиплексор ИС 1533КП7, следующие ФАЛ:

а)  $f(x_1, x_2, x_3) = \sum m(\quad)$ ,  
б)  $f(x_1, x_2, x_3, x_4) = \sum m(\quad)$ .

#### 3.3. Исследование работы декодера/демультиплексора.

1. Установить ИС 1533ИД7 на наборной панели IDL-800.
2. Вывод 16 ИС соединить с источником питания +5V, а вывод 8 с общей шиной установки.
3. Изменяя значения адресных входов от 000 до 111 проследить за состояниями выходов. Результаты наблюдений отразить в таблице истинности и получить выражения для выходов  $D_0 \div D_7$ .

#### 3.4. Исследование работы полного сумматора на основе декодера/демультиплексора.

1. Собрать схему полного сумматора, рис. 5.12.
2. Исследовать работу полного сумматора, результаты отразить в таблице истинности.

#### 3.5. Синтезировать и исследовать работу полного вычитателя на основе декодера/демультиплексора.

- 3.6. Синтезировать и исследовать работу компаратора для сравнения двухразрядных двоичных чисел.

#### **4. СОДЕРЖАНИЕ ОТЧЕТА**

1. Цель работы.
2. Схемы, исследуемые в работе.
3. Таблицы, отражающие результаты исследований.
4. Выводы по результатам исследований.

#### **5. КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. Дать определение мультиплексора и объяснить его работу.
2. Объяснить, каким образом мультиплексоры используются для синтеза КС.
3. Дать основные преимущества использования мультиплексоров при реализации ФАЛ.
4. Дать определение декодера/демультиплексора и объяснить его работу.
5. Объяснить, каким образом декодеры/демультиплексоры используются для реализации ФАЛ.
6. Почему декодеры/демультиплексоры наиболее удобны для реализации многовыходовых ФАЛ.

## Лабораторная работа №6 ТРИГГЕРЫ

### 1. ЦЕЛЬ ЛАБОРАТОРНОЙ РАБОТЫ

- изучение основ теории, методов логического синтеза и функционирования основных типов триггеров.

### 2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Триггером называется устройство, имеющее два устойчивых состояния и способное под действием управляющих сигналов скачкообразно переходить из одного состояния в другое. Одно состояние называется единичным, а второе - нулевым. В общем случае, триггер имеет два выхода- прямой  $Q$  и инверсный  $\bar{Q}$ , поскольку логическое состояние одного выхода всегда инверсно логическому состоянию другого. Состояние триггера определяется логическим уровнем на прямом выходе. Если на прямом выходе имеется потенциал соответствующий лог.1, то триггер находится в единичном состоянии или говорят, что триггер установлен (при этом потенциал на инверсном выходе соответствует лог.0). И если на прямом выходе имеется потенциал, соответствующий лог.0, то триггер находится в нулевом состоянии или говорят, что триггер сброшен (и при этом потенциал на инверсном выходе соответствует лог.1).

В качестве основных классификационных признаков используются функциональный признак и способ записи информации.

По функциональному признаку, т.е. по виду характеристического уравнения, связывающего логические переменные на входах и выходах триггера в момент срабатывания  $t_n$  и после срабатывания  $t_{n+1}$ , различают триггеры RS-, D-, JK-, T- и других типов.

По способу записи информации триггеры делятся на асинхронные (не тактируемые) и синхронные (тактируемые). В асинхронных триггерах запись информации происходит под действием изменений входных сигналов с момента подачи их на информационные входы. В синхронных триггерах запись информации происходит только при подаче сигнала синхронизации.

По способу синхронизации триггеры подразделяются на три категории в зависимости от того, какие параметры синхросигнала используются для записи информации:

- а) со статическим управлением;
- б) двухступенчатые, управляемые синхроимпульсом;
- в) с динамическим управлением.

Синхронный триггер со статическим управлением воспринимает информационные сигналы, когда синхросигнал достигает своего активного уровня. Характерной особенностью этого типа является то, что смена управляющего сигнала в течение времени действия импульса синхронизации вызывает новые

срабатывания триггеров, т.е. синхронные триггеры со статическим управлением при активном уровне синхросигнала ведут себя подобно асинхронным.

Асинхронные триггеры и синхронные триггеры со статическим управлением имеют ограниченное применение. Например, эти триггеры не могут использоваться в счётчиках или регистрах сдвига. В зарубежной литературе триггерные устройства подразделяются на два типа –latch(s) и flip-flop(s) (FF). Асинхронные триггеры и синхронные со статическим управлением относятся к latch типу, а двухступенчатые триггеры управляемые синхроимпульсом и триггеры с динамическим управлением относятся к FF типу. Двухступенчатые триггеры, управляемые импульсом и триггеры с динамическим управлением являются более универсальными.

Двухступенчатые триггеры, управляемые импульсом, воспринимают информационные сигналы, когда синхросигнал изменяет своё состояние с низкого на высокое, а затем снова на низкое, т.е. управляются импульсом.

Синхронный триггер с динамическим управлением воспринимает информационные сигналы только в момент действия положительного перехода (переход  $0 \rightarrow 1 = \uparrow$ ) или в момент действия отрицательного перехода (переход  $1 \rightarrow 0 = \downarrow$ ) синхроимпульса. Вход триггера С (или СК) называется прямым динамическим, если переключение триггера осуществляется положительным перепадом импульса синхронизации. Вход называется инверсным динамическим, если переключение триггера осуществляется отрицательным перепадом синхросигнала. Характерной особенностью триггеров с динамическим управлением является то, что в остальное время импульса синхронизации триггер не реагирует на информационные сигналы и остаётся в прежнем состоянии независимо от уровня синхросигнала.

В двухступенчатых триггерах переход в новое состояние происходит после окончания действия синхроимпульса.

**Асинхронный RS- триггер (SR-latch)** – это устройство с двумя устойчивыми состояниями, имеющее два входа S(Set- установка) и R(Reset- сброс) и два выхода Q и  $\bar{Q}$ . Асинхронный триггер функционирует в соответствии с таблицей истинности, табл. 6.1.

Таблица 6.1

$t_n$			$t_{n+1}$	Режим работы
$Q_n$	$S_n$	$R_n$	$Q_{n+1}$	
0	0	0	0	Хранение информации
1	0	0	1	
0	1	0	1	Установка лог.1
1	1	0	1	
0	0	1	0	Установка лог.0
1	0	1	0	
0	1	1	x	Неопределённость
1	1	1	x	Неопределённость

Если входы  $S_n$  и  $R_n$  оба в состоянии лог.0, то триггер не изменяет свое состояние, т.е. триггер хранит один бит информации.

Если  $S_n = 1$  и  $R_n = 0$ , то независимо от того в каком состоянии триггер находился, следующее состояние триггера  $Q_{n+1} = 1$ .

Если  $S_n = 0$  и  $R_n = 1$ , то независимо от того в каком состоянии находился триггер, следующее его состояние  $Q_{n+1} = 0$ .

Комбинация входных сигналов  $S_n = R_n = 1$  является неопределенной, поскольку триггер после воздействия на входах активных уровней может равновероятно перейти как в нулевое, так и в единичное состояние. Поэтому одновременная подача активных уровней на входы S и R не допускается.

Работа RS-триггера также может быть представлена функцией на карте Карно и логическим уравнением. Входные и выходные переменные триггера в момент срабатывания  $t_n$  и после срабатывания  $t_{n+1}$  можно представить функциональной зависимостью

$$Q_{n+1} = f(S_n, R_n, Q_n), \quad (6.1)$$

где  $R_n, S_n$  - состояние информационных входов,

$Q_n$  - значение выходного сигнала триггера в момент времени  $t_n$ ,

$Q_{n+1}$  - значение выходного сигнала триггера в момент времени  $t_{n+1}$ .

Функция алгебры логики, табл.6.1 может быть представлена картой Карно, рис. 6.1. Значения ФАЛ, показанные значком x, указывают на то, что данная ФАЛ является недоопределённой. При минимизации ФАЛ мы можем её доопределить, так как нам это удобно, чтобы получить минимальную форму, рис.6.1.

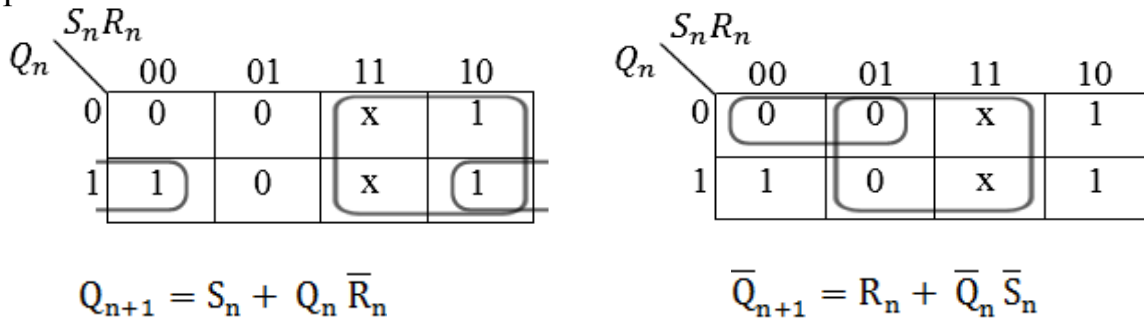


Рис.6.1. Карты Карно для минимизации выходных функций RS-триггера

Выражения

$$Q_{n+1} = S_n + Q_n \bar{R}_n, \quad (6.2)$$

$$\bar{Q}_{n+1} = R_n + \bar{Q}_n \bar{S}_n, \quad (6.3)$$

используются для построения RS-триггера.

Применяя инверсию и закон де Моргана, выражения (6.2) и (6.3) могут быть преобразованы следующим образом:

$$\overline{Q_{n+1}} = S_n + \overline{Q_n} + R_n, \quad (6.4)$$

$$Q_{n+1} = R_n + \overline{Q_n} + S_n. \quad (6.5)$$

Выражения (6.4) и (6.5) используются для реализации RS-триггера на элементах ИЛИ-НЕ.

Применяя двойную инверсию и закон де Моргана, выражения (6.2) и (6.3) могут быть преобразованы следующим образом:

$$Q_{n+1} = \overline{\overline{S_n} \overline{Q_n} \overline{R_n}}, \quad (6.6)$$

$$\overline{Q_{n+1}} = \overline{\overline{R_n} \overline{Q_n} \overline{S_n}}. \quad (6.7)$$

Выражения (6.6) и (6.7) используются для реализации RS-триггера на элементах И-НЕ.

Реализация RS-триггера на элементах ИЛИ-НЕ показана на рис.6.2а,б.

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

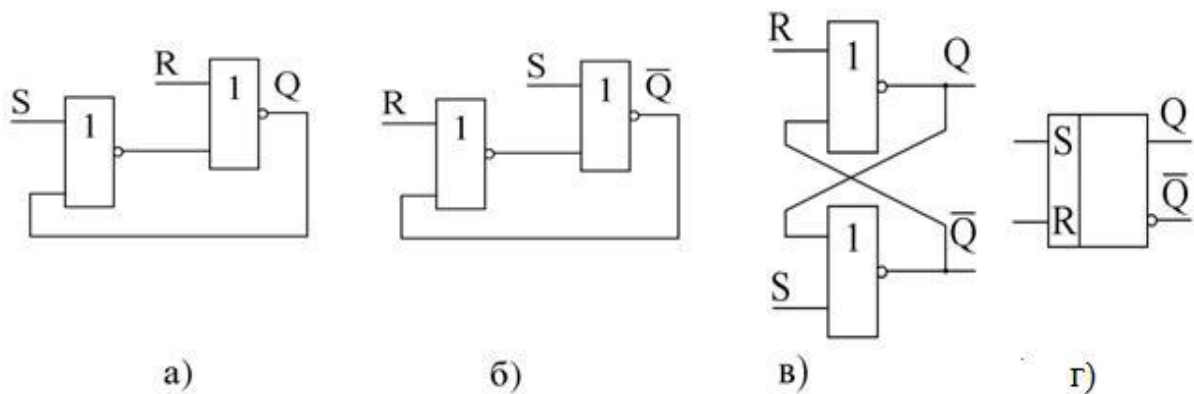


Рис. 6.2. RS-триггеры на элементах ИЛИ-НЕ

Обычно RS-триггер изображается с расположением элементов ИЛИ-НЕ как показано на рис. 6.2в. На рис. 6.2г показано условное изображение RS-триггера.

При анализе работы RS-триггера и при синтезе других триггеров используется таблица переходов RS-триггера (таблица 6.2), которая определяет комбинации входных сигналов необходимых для того или иного перехода триггера, т.е. триггер в момент времени  $t_n$  находится в каком-то состоянии ( $Q_n = 0$  или  $Q_n = 1$ ) и определяются состояния входов, чтобы триггер перешёл в определённое следующее состояние  $Q_{n+1}$ .

Таблица 6.2

Настоящее состояние	Следующее состояние	Требуемые входы	
		$S_n$	$R_n$
$Q_n$	$Q_{n+1}$		
0	0	0	x
0	1	1	0
1	0	0	1
1	1	x	0

На рис. 6.3 приведены временные диаграммы поясняющие принцип работы асинхронного RS-триггера, рис.6.2.

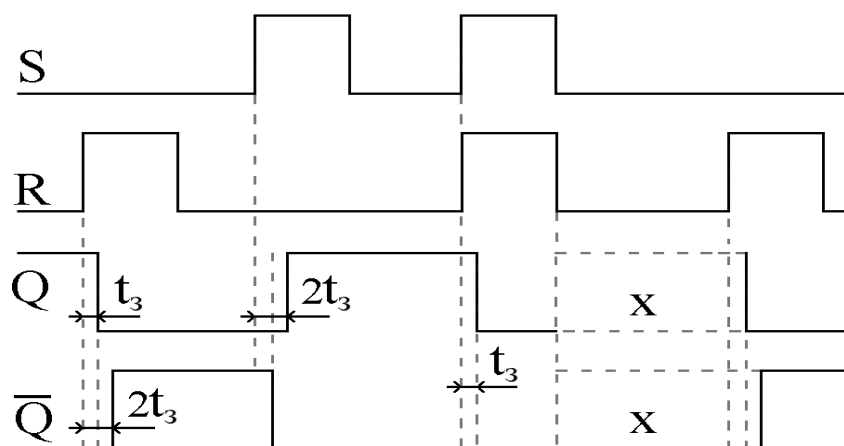


Рис. 6.3. Временные диаграммы работы асинхронного RS-триггера

Для устойчивого функционирования триггера длительность сигнала на входах R и S должна быть не меньше времени переключения триггера,  $t_n \geq t_3$ . Информационные сигналы должны поступать на входы поочерёдно и только после окончания переходных процессов в триггере, тогда максимальная частота переключения триггера будет равна:

$$f_{\max} = \frac{1}{2 t_3} \quad (6.8)$$

Однако при  $f_{\max}$ , длительность выходных сигналов не будет превышать  $t_{3, \text{ср}}$ . Такие сигналы не являются достаточными для надежной передачи информации в логическую цепь, поэтому максимальная рабочая частота триггера,  $f_p$  определяется как:

$$f_p = \frac{1}{3 t_{3, \text{ср}}} \quad (6.9)$$

## Асинхронный RS-триггер на элементах И-НЕ, $\overline{R}\overline{S}$ - триггер ( $\overline{S}\overline{R}$ -latch)

Воспользуемся выражениями (6.6) и (6.7) и реализуем RS-триггер на элементах И-НЕ. Эта реализация показана на рис.6.4 .

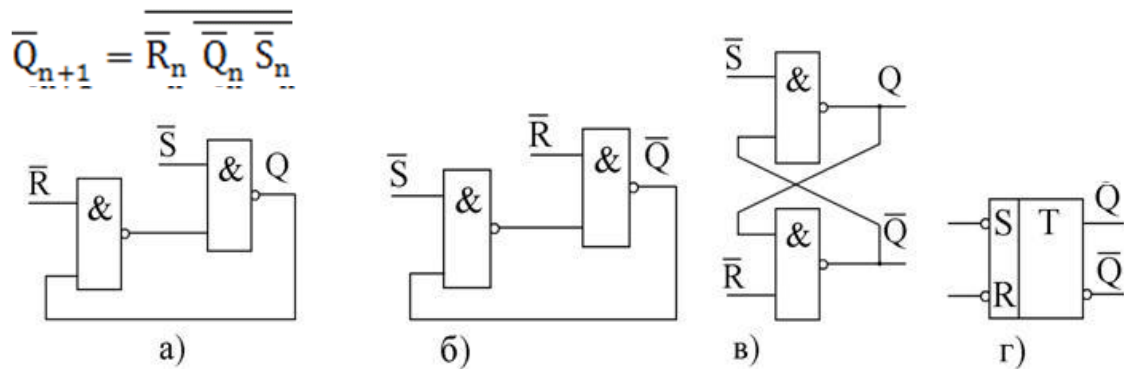


Рис. 6.4. Асинхронный RS - триггер:  
 а, б - построение RS - триггера,  
 в - изображение триггера на элементах И-НЕ  
 г- условное изображение  $\overline{R}\overline{S}$ -триггера.

RS- триггер на элементах И-НЕ, рис. 6.4, называется асинхронным триггером с инверсными входами, . Характеристическая таблица или таблица истинности такого триггера представлена табл.6.3, а таблица переходов, табл. 6.4.

Таблица 6.3

$t_n$			$t_{n+1}$
$Q_n$	$\overline{S}_n$	$\overline{R}_n$	$Q_{n+1}$
0	1	1	0
1	1	1	1
0	0	1	1
1	0	1	1
0	1	0	0
1	1	0	0
0	0	0	x
1	0	0	x

Таблица 6.4

Настоящее состояние	Следующее состояние	Требуемые выходы	
		$\overline{S}_n$	$\overline{R}_n$
$Q_n$	$Q_{n+1}$		
0	0	1	x
0	1	0	1
1	0	1	0
1	1	x	1

На рис. 6.5. показаны временные диаграммы поясняющие принцип работы асинхронного RS-триггера на элементах И-НЕ.

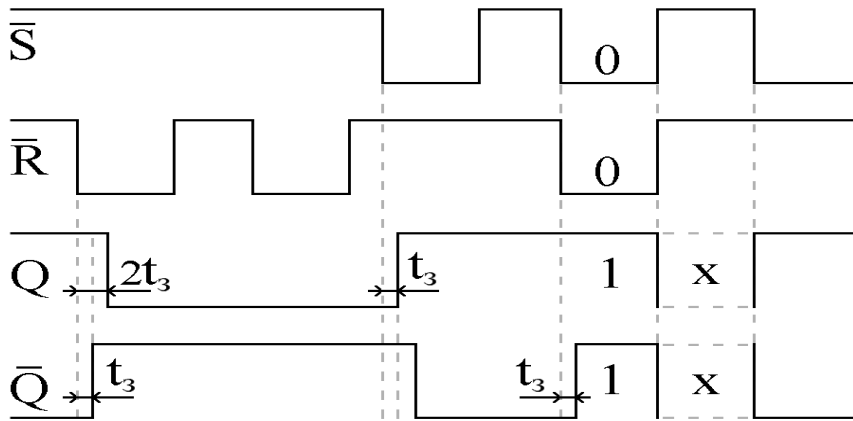


Рис. 6.5. Временные диаграммы работы  $\bar{R}\bar{S}$ -триггера

### Синхронный RS-триггер (SR-latch) со статическим управлением

Асинхронный RS-триггер является элементарной запоминающей ячейкой и используется при синтезе других более сложных триггеров. В том числе, синхронный RS-триггер со статическим управлением может быть синтезирован на основе базового асинхронного RS – триггера с инверсными входами. В этом случае, блок-схема синхронного RS-триггера со статическим управлением состоит из асинхронного RS - триггера и комбинационной схемы (КС), как показано на рис. 6.6а.

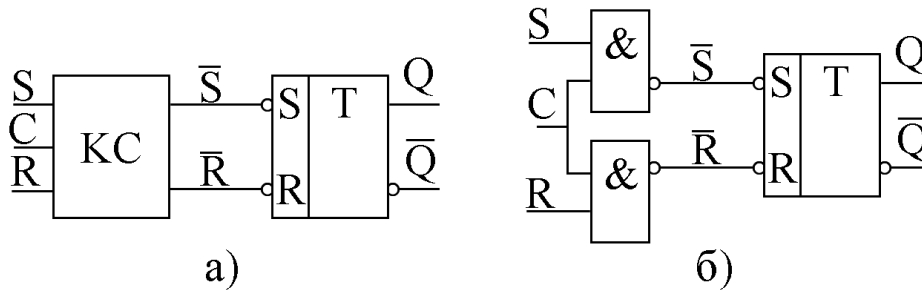


Рис. 6.6. Синхронный RS-триггер со статическим управлением

Задачей синтеза является определение функции  $\bar{S}_n$  и  $\bar{R}_n$  и построение КС, которая соответствующим образом управляет работой асинхронного  $\bar{R}\bar{S}$ -триггера. Для этого построим таблицу истинности функций  $\bar{S}_n$  и  $\bar{R}_n$  (табл. 6.5) и минимизируем эти функции с помощью карт Карно. Таблицу истинности для функций  $\bar{S}_n$  и  $\bar{R}_n$  строим на основании таблицы истинности синхронного RS-триггера и таблицы переходов асинхронного RS - триггера.

Таблица 6.5

$C_n$	$Q_n$	$S_n$	$R_n$	$Q_{n+1}$	$\bar{S}_n$	$\bar{R}_n$
0	0	0	0	0	1	x
0	1	0	0	1	x	1
0	0	1	0	0	1	x
0	1	1	0	1	x	1
0	0	0	1	0	1	x
0	1	0	1	1	x	1
0	0	1	1	0	1	x
0	1	1	1	1	x	1
<hr/>						
1	0	0	0	0	1	x
1	1	0	0	1	x	1
1	0	1	0	1	0	1
1	1	1	0	1	x	1
1	0	0	1	0	1	x
1	1	0	1	0	1	0
1	0	1	1	x	x	x
1	1	1	1	x	x	x

$C_n Q_n \backslash S_n R_n$	00	01	11	10
00	1	1	1	1
01	x	x	x	x
11	x	1	x	x
10	1	1	x	0

$$\bar{S}_n = \bar{C}_n + \bar{S}_n = \overline{C_n S_n}$$

$C_n Q_n \backslash S_n R_n$	00	01	11	10
00	x	x	x	x
01	1	1	1	1
11	1	0	x	1
10	x	x	x	1

$$\bar{R}_n = \bar{C}_n + \bar{R}_n = \overline{C_n R_n}$$

На основании выражений для  $\bar{S}$  и  $\bar{R}$  блок-схема, рис. 6.6а, преобразуется в логическую схему синхронного триггера со статическим управлением, рис 6.6б. Полная логическая схема синхронного триггера на элементах И-НЕ и условное обозначение этого триггера показаны на рис. 6.7.

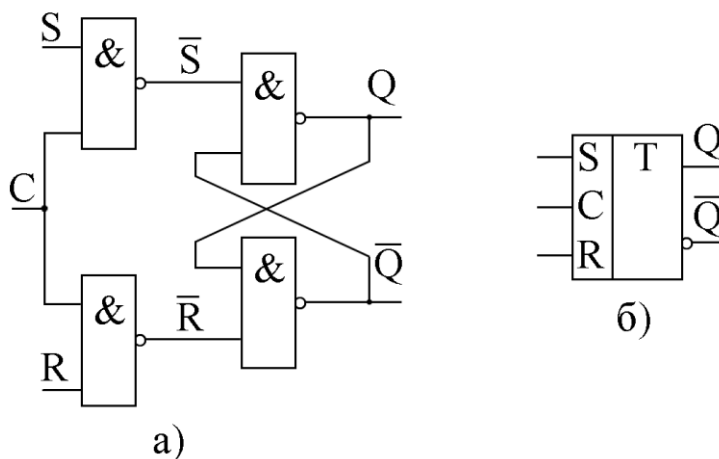


Рис. 6.7. Логическая схема и условное обозначение синхронного RS-триггера со статическим управлением

## D-триггер

D-триггер имеет информационный вход D (data, delay) и вход синхронизации C. Триггер принимает информационные сигналы по разрешению синхросигнала и повторяет их на выходе с некоторой задержкой. Синтез синхронного D-триггера (D-latch) осуществим на основе базового асинхронного D-триггера. Блок-схема такого триггера показана на рис.6.8 а.

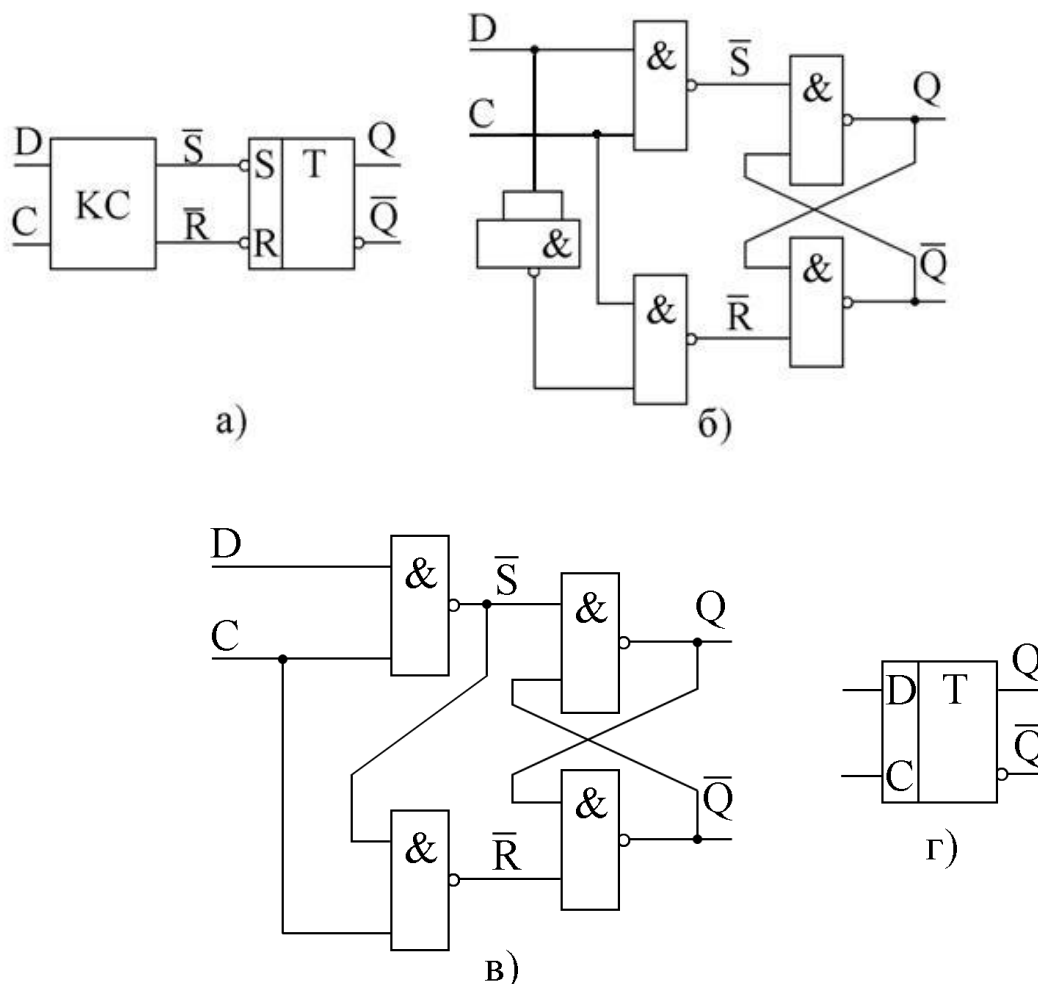


Рис. 6.8 Синхронный D-триггер со статическим управлением

Из блок-схемы (рис 6.8а) очевидно, что для получения логической схемы синхронного D-триггера со статическим управлением, необходимо определить комбинационную схему (КС) генерирующую функции  $\bar{S}$  и  $\bar{R}$  такими, чтобы базовый RS - триггер функционировал как D-триггер. Для получения КС составим таблицу истинности для функций  $\bar{S}_n$  и  $\bar{R}_n$ , на основе таблицы истинности синхронного D-триггера со статическим управлением и таблицы переходов асинхронного RS - триггера, табл. 6.6, и с помощью карт Карно минимизируем выражения для  $\bar{S}_n$  и  $\bar{R}_n$ .

Таблица 6.6.

$C_n$	$D_n$	$Q_n$	$Q_{n+1}$	$\bar{S}_n$	$\bar{R}_n$
1	0	0	0	1	x
1	0	1	0	1	0
1	1	0	1	0	1
1	1	1	1	x	1
0	0	0	0	1	x
0	0	1	1	x	1
0	1	0	0	1	x
0	1	1	1	x	1

$C_n$	$D_n Q_n$	00	01	11	10
0	0	1	x	x	1
1	1	1	1	x	0

$$\bar{S}_n = \bar{C}_n + \bar{D}_n = \bar{C}_n \bar{D}_n$$

$C_n$	$D_n Q_n$	00	01	11	10
0	0	x	1	1	x
1	1	x	0	1	1

$$\bar{R}_n = \bar{C}_n + D_n = \bar{C}_n \bar{D}$$

Используя выражения для  $\bar{S}$  и  $\bar{R}$ , построим логическую схему синхронного D-триггера со статическим управлением, рис. 6.8б.

Логическая схема синхронного D-триггера со статическим управлением может быть упрощена, если при построении КС использовать не минимальное выражение для  $\bar{R}_n$ , а как показано на рис. 6.9.

$$\bar{R}_n = \bar{C}_n + C_n D_n = \overline{C_n \cdot \bar{C}_n \bar{D}_n}$$

$C_n$	$D_n Q_n$	00	01	11	10
0	0	x	1	1	1
1	1	x	0	1	1

Рис. 6.9 Карта Карно для  $\bar{R}_n$

При использовании, для построения триггера выражений  $\bar{S}_n = \bar{C}_n \cdot \bar{D}_n$  и  $\bar{R}_n = \overline{C_n \cdot \bar{C}_n \bar{D}_n}$  получается логическая схема синхронного D-триггера со статическим управлением, показанная на рис. 6.8в.

### JK-триггеры

JK-триггер это модернизированный RS-триггер в котором неопределённое состояние RS-триггера доопределено таким образом, что при  $J=K=1$  триггер переключается в противоположное состояние. При этом вход J соответствует входу S, а вход K- входу R.

Рассмотрим синтез JK-триггера со статическим управлением, используя базовый асинхронный RS - триггер. Блок-схема триггера показана на рис. 6.10а.

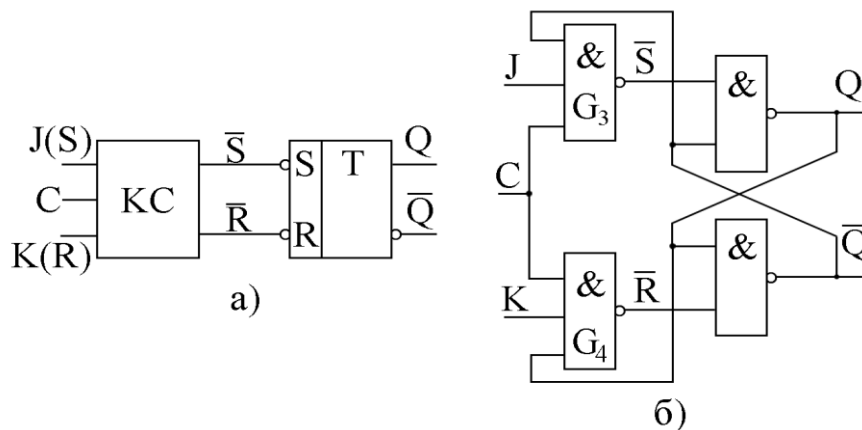


Рис. 6.10. Синхронный JK-триггер со статическим управлением

Для синтеза JK-триггера со статическим управлением составим таблицу истинности для функций  $\bar{S}$  и  $\bar{R}$ , табл. 6.7.

$C_n Q_n \backslash J_n K_n$	00	01	11	10
00	1	1	1	1
01	x	x	x	x
11	x	1	1	x
10	1	1	0	0

$$\bar{S}_n = \bar{C}_n + Q_n + \bar{J}_n = \overline{C_n \bar{Q}_n J_n}$$

$C_n Q_n \backslash J_n K_n$	00	01	11	10
00	x	x	x	x
01	1	1	1	1
11	1	0	0	1
10	x	x	1	1

$$\bar{R}_n = \bar{C}_n + \bar{Q}_n + \bar{K}_n = \overline{C_n Q_n K_n}$$

Таблица 6.7

$C_n$	$Q_n$	$J_n$	$K_n$	$Q_{n+1}$	$\bar{S}_n$	$\bar{R}_n$
1	0	0	0	0	1	x
1	1	0	0	1	x	1
1	0	1	0	1	0	1
1	1	1	0	1	x	1
1	0	0	1	0	1	x
1	1	0	1	0	1	0
1	0	1	1	1	0	1
1	1	1	1	0	1	0
0	0	0	0	0	1	x
0	1	0	0	1	x	1
0	0	1	0	0	1	x
0	1	1	0	1	x	1
0	0	0	1	0	1	x
0	1	0	1	1	x	1
0	0	1	1	0	1	x
0	1	1	1	1	x	1

Минимизируем функции  $\bar{S}_n$  и  $\bar{R}_n$  с помощью карт Карно и преобразуем их к форме удобной для реализации с помощью элементов И-НЕ. Схема триггера показана на рис. 6.10б.

Рассмотрим работу синхронного JK-триггера со статическим управлением, когда  $J=K=1$  и на вход синхронизации поступают синхроимпульсы. Когда

на входе  $C$  уровень лог.0 на выходах логических элементов  $G_3$  и  $G_4$  уровни лог.1 и асинхронный триггер на элементах  $G_1$  и  $G_2$  сохраняет своё состояние. Когда на вход  $C$  поступает импульс синхронизации, т.е.  $C=1$ , то импульс будет передаваться через один из логических элементов  $G_3$  или  $G_4$ , вход которого, соединённый с выходом триггера, будет в данный момент равен лог.1. Если  $Q=1$ , выход логического элемента  $G_4$  становится равным нулю, когда поступает импульс синхронизации и триггер обнуляется. Если  $\bar{Q}=1$ , выход  $G_3$  становится равным нулю, при подаче импульса синхронизации и триггер устанавливается. В любом случае состояние триггера изменяется.

Однако, когда  $J=K=1$ ,  $Q=0$  и импульс синхронизации действует на входе, то после временного интервала  $\Delta t$ , равного времени задержки распространения сигнала через два элемента И-НЕ ( $G_3$  и  $G_4$ ), выход триггера изменится на  $Q=1$ . Теперь мы имеем  $J=K=1$  и  $Q=1$ , и после другого временного интервала  $\Delta t$  выход будет изменяться на  $Q=0$ . Следовательно, мы можем сделать вывод, что в течение длительности  $t_u$ , импульса синхронизации, схема рис. 6.10 б имеет неустойчивое состояние, т.е. находится в автоколебательном режиме, а после окончания импульса синхронизации состояние триггера будет неизвестно. Это говорит о том, что синхронный JK-триггер со статическим управлением не может быть реализован на одной элементарной запоминающей ячейке.

Казалось, этой ситуации можно было бы избежать, если бы  $t_u < \Delta t < T$ . Однако выполнить это невозможно в виду очень малого времени задержки распространения сигнала в ИС.

Поэтому, практически используемыми JK-триггерами являются: двухступенчатые, управляемые синхроимпульсом триггеры, и триггеры с динамическим управлением.

Управляемые синхроимпульсом триггеры строятся по 2-ступенчатой M-S (Master-Slave) конфигурации. JK-триггер M-S типа состоит из каскада двух RS-триггеров с обратной связью с выхода второго на вход первого, рис. 6.11.

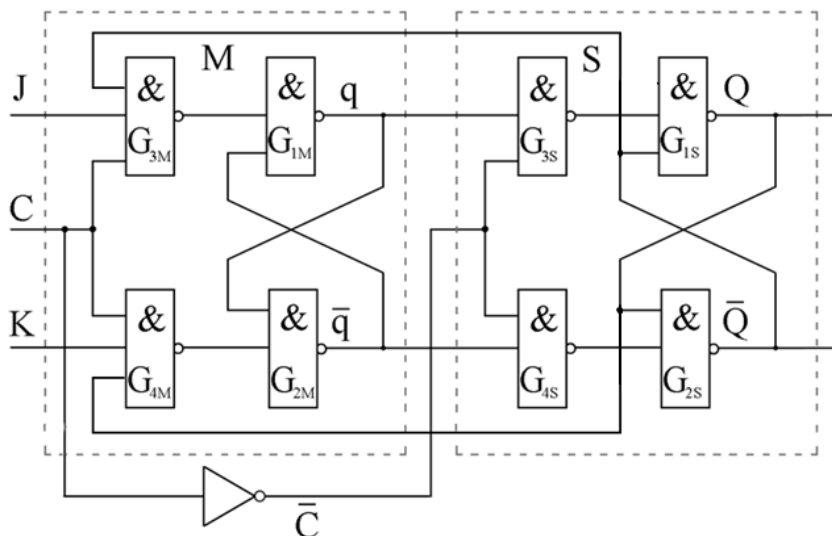


Рис. 6.11. Двухступенчатый JK-триггер с инвертором.

Когда  $C=1$ , первый триггер имеет разрешение и выходы  $q$  и  $\bar{q}$  зависят от состояния входов  $J$  и  $K$ . В то же время, на второй триггер подается запрет, поскольку  $\bar{C}$ . Когда  $C$  изменяется на низкий уровень,  $C=0$  ( $\bar{C}=1$ ), на первый триггер подается запрет, а на второй триггер разрешение, поскольку теперь его синхровход  $\bar{C}=1$ . Поэтому выходы  $Q$  и  $\bar{Q}$  повторяют состояния на выходах  $q$  и  $\bar{q}$ , соответственно. Поскольку второй триггер всегда повторяет состояние первого, его назвали  $S$  (slave), а первый  $M$  (master). В этой схеме входы  $G_{3M} = \bar{Q}$  и  $G_{4M} = Q$  не меняются в течение импульса синхронизации, поэтому  $JK$ -триггер не может находиться в автоколебательном режиме. Состояние  $M$ - $S$  триггера изменяется при отрицательном перепаде импульса синхронизации.

Имеется другая версия  $JK$ -триггера в конфигурации  $M$ - $S$ , рис. 6.12.

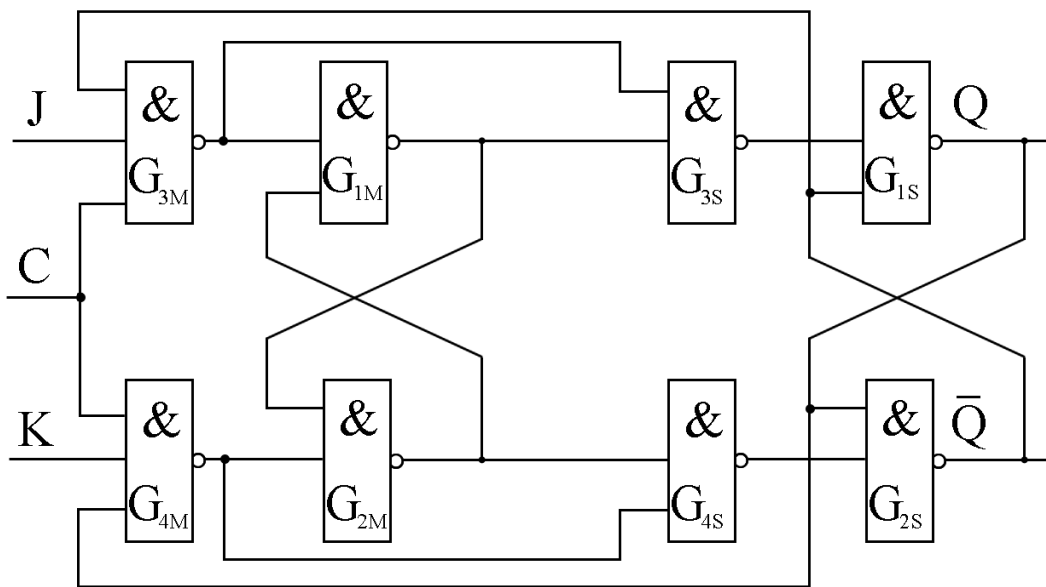


Рис 6.12 Двухступенчатый  $JK$ -триггер с запрещающими связями

Условное обозначение  $JK$ -триггера показано на рис. 6.13.

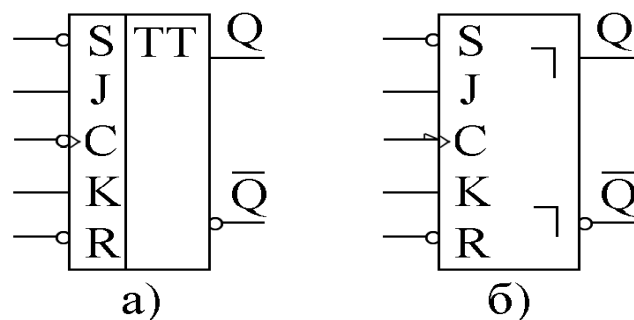


Рис. 6.13. Условное обозначение  $JK$ -триггера

## Т-триггеры

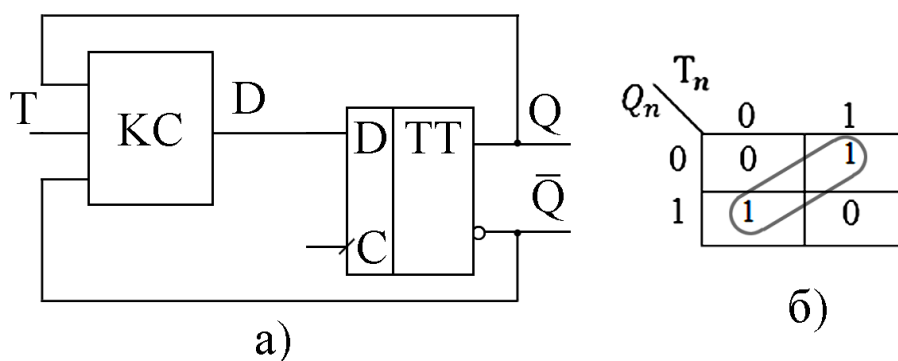
Т-триггер имеет один вход Т (trigger or toggle). Т-триггер изменяет свое состояние на противоположное каждым импульсом синхронизации, если  $T=1$  и сохраняет свое состояние, если  $T=0$ . Т-триггеры не производятся в интегральном исполнении, но могут быть легко построены из JK- и D- триггеров. Исходя из определения Т- и JK- триггеров очевидно, что Т-триггер получается из JK- триггера , если входы J и K объединить, рис. 6.15а, поэтому Т-триггер иногда рассматривают как одноходовый вариант JK-триггера .

Для того, чтобы преобразовать D-триггер в Т-триггер, воспользуемся таблицей переходов для обоих триггеров, табл. 6.8.

Таблица 6.8

Настоящее состояние	Следующее состояние	D-триггер	Т-триггер
$Q_n$	$Q_{n+1}$	$D_n$	$T_n$
0	0	0	0
0	1	1	1
1	0	0	1
1	1	1	0

Блок-схема Т-триггера на основе D-триггера показана на рис. 6.14а.



$$D_n = Q_n \oplus T_n$$

Рис. 6.14 Блок-схема преобразования D-триггера в Т-триггер

Из блок-схемы, рис. 6.14, очевидно, что необходимо синтезировать такую КС, чтобы D-триггер функционировал как Т-триггер. ФАЛ, описывающую эту КС, имеет входные переменные  $T_n$  и  $Q_n$  , а выходная переменная  $D_n$  (см. табл. 6.8). Функция  $D_n(Q_n, T_n)$  может быть представлена и упрощена с помощью карты Карно, рис. 14 б.

В соответствии с полученным выражением

$$D_n = Q_n \oplus T_n, \quad (6.10)$$

схема Т-триггера будет выглядеть, как показано на рис. 6.15б.

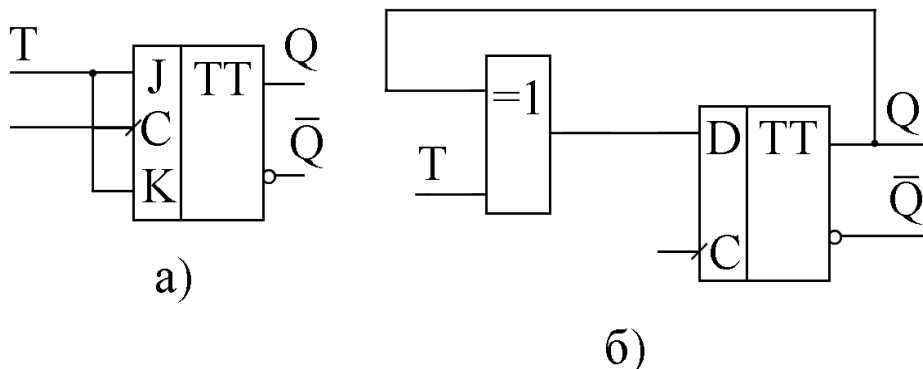


Рис. 6.15 Т-триггер на основе J-K-триггера и на основе D-триггера

Если вход  $T=1$ , то Т-триггер меняет свое состояние всякий раз, когда поступает импульс синхронизации.

Если переменная  $T_n$  в уравнении (6.10) равна единице, то уравнение может быть переписано следующим образом

$$D_n = Q_n \oplus 1 = \overline{Q_n} \quad (6.11)$$

Уравнение (6.11) показывает, что схему делителя на 2 можно построить простым соединением  $\overline{Q_n}$  с D входом, рис 6.16.

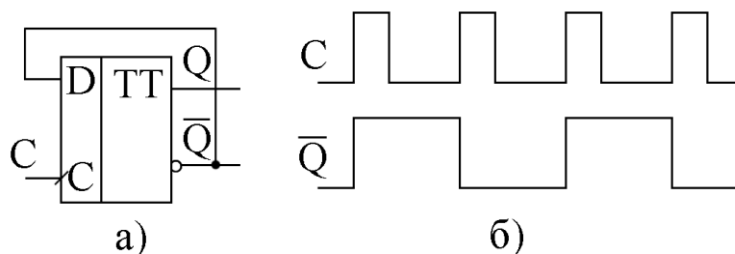


Рис. 6.16 Схема делителя на 2 и его временные диаграммы

Необходимо отметить, что Т-триггер не может быть построен на основе триггера со статическим управлением (latch).

### Синхронный D-триггер с динамическим управлением.

В триггерах с динамическим управлением, изменения выходного состояния происходят в момент перехода синхросигнала с нулевого уровня на единичный ( $0 \rightarrow 1$ ), либо наоборот ( $1 \rightarrow 0$ ), при достижении порогового уровня. При этом информационные входы триггера в этот момент запираются, и триг-

гер становится нечувствительным к изменениям входных сигналов до тех пор, пока синхросигнал не вернется в исходное нулевое состояние, и другой синхроимпульс не поступит на синхровход. Если триггер переключается положительным перепадом синхросигнала, то вход С называется прямым динамическим, если отрицательным- то инверсным динамическим.

Схема синхронного D-триггера с динамическим управлением приведена на рис. 6.17.

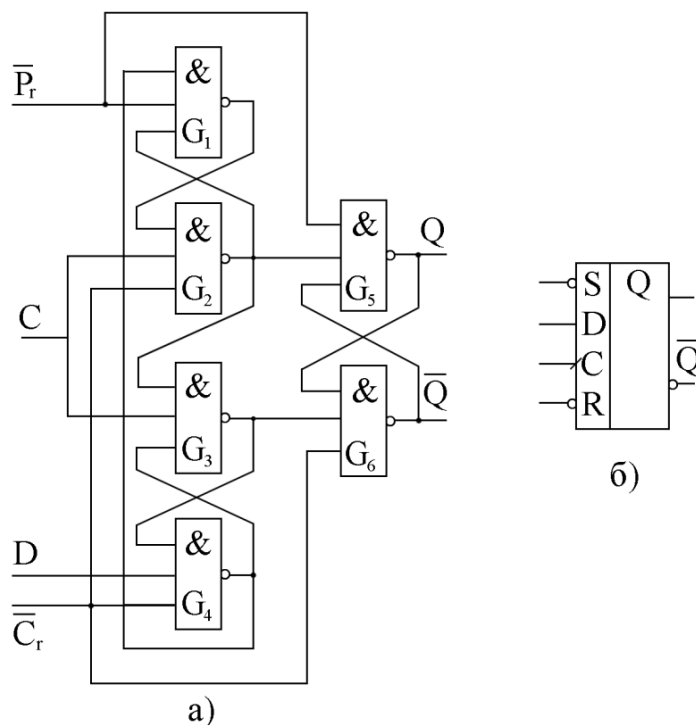
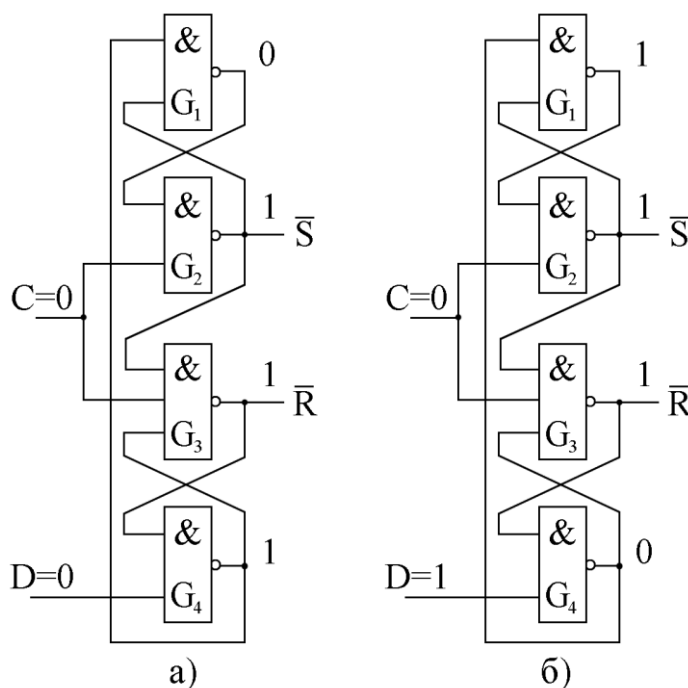


Рис. 6.17 Логическая схема и условное обозначение синхронного D-триггера с динамическим управлением (КР1533ТМ2)



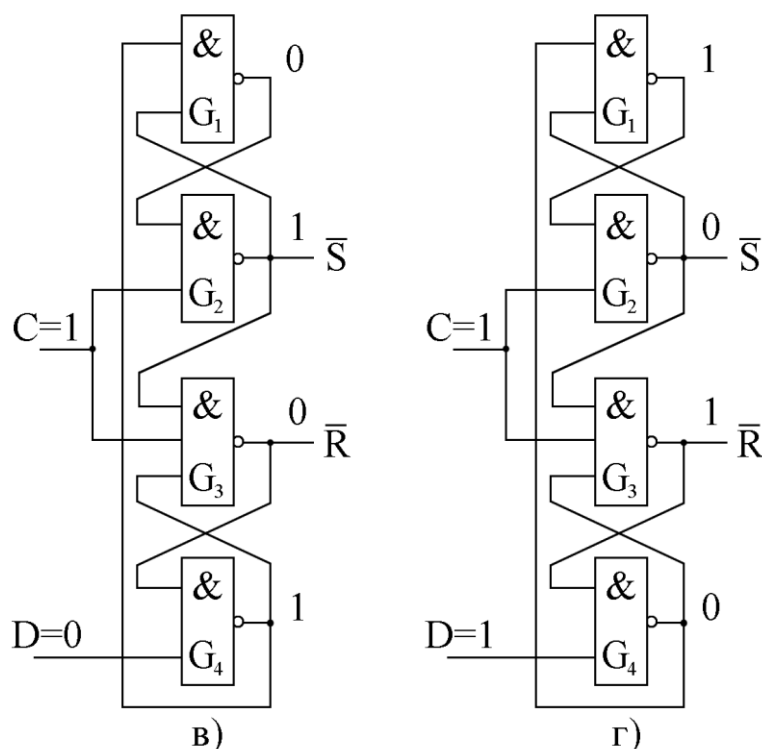


Рис. 6.18 Работа синхронного D-триггера с динамическим управлением

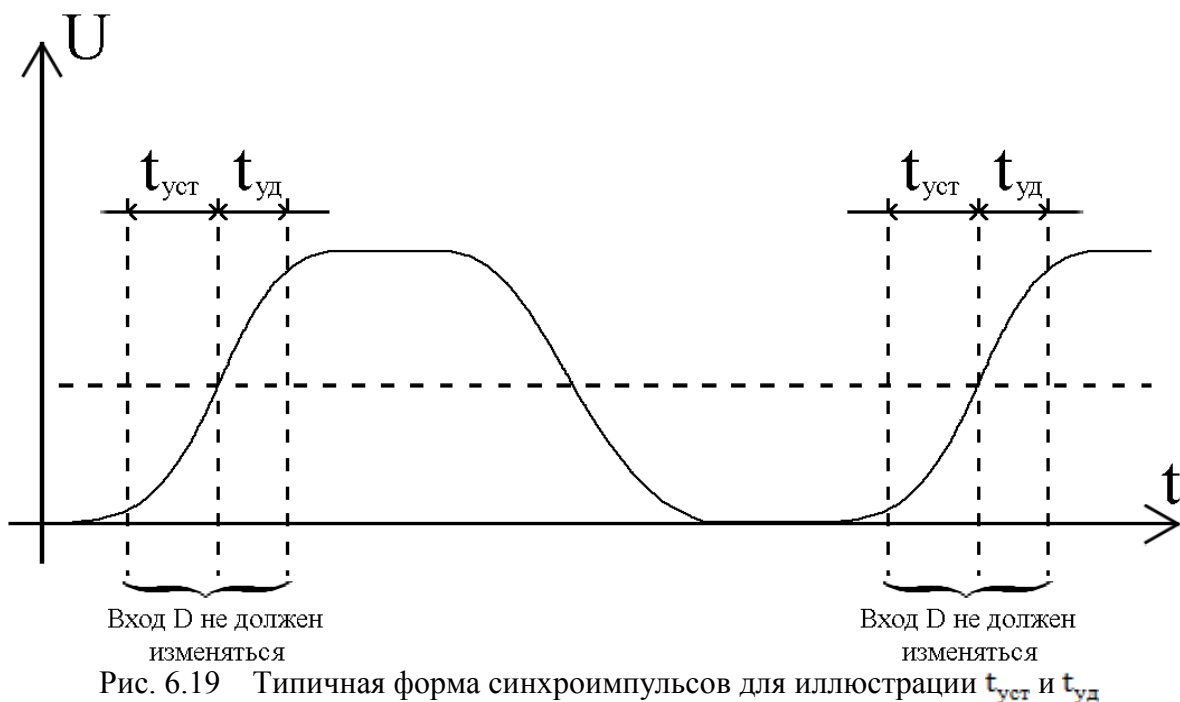
Схема состоит из трех базовых  $\overline{RS}$  – триггеров. И-НЕ элементы  $G_1$  и  $G_2$  образуют один базовый  $\overline{RS}$  – триггер, элементы  $G_3$  и  $G_4$  образуют другой  $\overline{RS}$  – триггер, Третий  $\overline{RS}$  – триггер, состоящий из  $G_5$  и  $G_6$ , является выходом всей схемы.

Работу синхронного D-триггера с динамическим управлением рассмотрим с помощью рис. 6.18, где логические элементы  $G_1$  и  $G_4$  используются, чтобы показать все возможные переходы. На установочные входы  $\overline{R}$  и  $\overline{S}$  поданы логические единицы, поэтому для простоты на рис. 6.18 они опущены. Выходы логических элементов  $G_2$  и  $G_3$  являются управляющими  $\overline{S}$  и  $\overline{R}$  входами асинхронного  $\overline{RS}$  – триггера. На рис. 6.18а, б показаны значения на выходах  $G_1 - G_4$ , когда СК=0. На вход D может подаваться логический 0 или логическая 1. В любом случае СК=0 и на выходах  $G_2$  и  $G_3$  лог. 1, т.е.  $\overline{S} = \overline{R} = 1$  и выходной  $\overline{RS}$  – триггер находится в режиме хранения информации. Если D=0, на выходе  $G_4$  лог. 1 и на выходе  $G_1$  лог.0. Если D=1, то на выходе  $G_4$  лог.0 и на выходе  $G_1$  лог.1. При этих двух состояниях, когда на входе СК=0, триггер не меняет свое состояние независимо от того, меняется ли состояние информационного входа D.

Теперь рассмотрим поведение синхронного D-триггера с динамическим управлением, когда на синхровход подается лог. 1. Если D=0, когда на вход C поступает лог.1, то  $\overline{S}$  остается в состоянии лог.1, а вход  $\overline{R}$  переходит в состояние лог.0. Этот лог.0 устанавливает триггер в состояние  $Q=0$  ( $\overline{Q} = 1$ ), а также

поступает на один из входов  $G_4$  и запирает вход D, блокируя любые изменения на входе D. Выход  $G_4$  может изменяться лишь, когда синхровход возвращается в состояние лог.0, однако теперь оба входа  $\bar{S}$  и  $\bar{R}$  устанавливаются в состояние  $\bar{S}=\bar{R} = 1$ , запрещая тем самым любые изменения выхода D-триггера (и исключая неустойчивое состояние).

При анализе работы триггера с динамическим управлением необходимо принимать во внимание, что существует определенный промежуток времени, состоящий из времени установления и времени удержания, в течение которого состояние входа D не должно изменяться. Время установления,  $t_{уст}$  равно времени задержки распространения через элементы  $G_4$  и  $G_1$ , поскольку изменения на входе D приводят к изменению выходов этих элементов. Время удержания,  $t_{уд}$  равно времени задержки распространения через элемент  $G_3$ , чтобы гарантировать, что  $\bar{R} = 0$  и удерживает выход элемента  $G_4$  в состоянии лог. 1, независимо от состояния входа D. Эти временные интервалы можно также пояснить с помощью рис. 6.19.



Если  $D=1$ , когда  $СК=1$ , то  $\bar{S}$  становится равным лог. 0, а  $\bar{R}$  остается равным лог. 1, что устанавливает на выходе  $Q=1$ . Изменения на входе D пока  $СК=1$  не изменяет логические уровни  $\bar{S}$  и  $\bar{R}$  потому, что на выходе  $G_1$  сохраняется лог. 1. Когда синхроимпульс заканчивается и на входе СК устанавливается лог. 0,  $\bar{S} = \bar{R}=1$ , что сохраняет состояние D-триггера.

Таким образом, когда на входе С происходит положительный перепад, значения входа D передается на выход Q. Изменения на входе D, когда  $СК=1$

или когда происходит отрицательный перепад синхросигнала, или когда  $СК=0$ , не приводят к изменению состояния D-триггера.

### JK-триггер с динамическим управлением

JK-триггер с динамическим управлением может быть построен с использованием логической схемы рис. 6.17, а также и другой логической схемы, которая является основой для построения триггеров ТВ6, ТВ9, ТВ10, ТВ11. На рис. 6.20 представлена логическая схема и условное обозначение JK-триггера, подобного ТВ9.

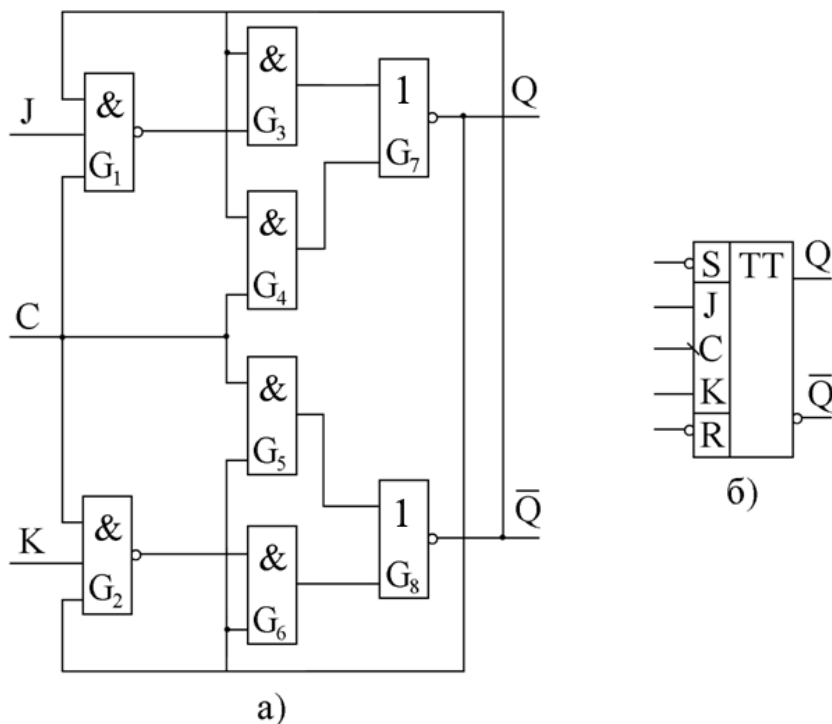


Рис. 6.20 Логическая схема и условное обозначение JK-триггера с инверсным динамическим управлением

Основным достоинством схемы рис.6.20а является то, что время удержания  $t_{уд}$  для нее равняется 0, что повышает ее быстродействие.

Рассмотрим особенности схемной реализации и работу этой схемы. Для нормального функционирования, задержка распространения сигнала через элементы  $G_1$  и  $G_2$  превышает суммарную задержку остальных элементов И, ИЛИ-НЕ. Изменения состояния триггера происходят по отрицательному перепаду синхросигнала, т.е. тактовый вход С триггера является инверсным динамическим. Проанализируем работу схемы для  $J=K=1$ . Пусть триггер находится в состоянии  $Q_n = 0$ , а на синхровходе действует лог.1 ( $C=1$ ), тогда на выходе элемента  $G_1$  действует лог. 0, на выходе  $G_3$  – лог.0, на выходе  $G_4$  – лог.1, которая гарантирует на выходе триггера  $Q=0$ . На инверсном выходе триггера установиться  $\bar{Q} = 1$ , поскольку на выходах элементов  $G_2, G_5$  и  $G_6$  уровень лог.0.

Теперь при переходе  $C$  от лог.1 к лог. 0 (переход  $1 \rightarrow 0 = \downarrow$ ) на выходах элементов  $G_3$  и  $G_4$  устанавливается лог.0, а на выходе  $G_7$ -лог.1, т.е. выход триггера  $G_{n+1} = 1$ . Лог. 1 с выхода  $G_7$  поступает на вход  $G_6$ , на второй вход которого поступает лог. 1 с выхода  $G_2$ , поэтому на выходе  $G_6$  появляется лог.1, а на выходе  $G_8$  – лог.0, т.е.  $\bar{Q}=0$ . Лог. 0 с выхода элемента  $G_8$  поступает на вход элемента  $G_3$  прежде, чем на втором входе появиться лог.1 с выхода элемента  $G_1$ , т.к. время распространения сигнала через элемент  $G_1$  больше, чем суммарное время распространения сигнала через элементы  $G_3 - G_8$ . При возвращении синхросигнала в состояние лог.1 ( $C=1$ ), на выходах элементов  $G_3$  и  $G_4$  сохраняется лог.0, т.к. на один из входов  $G_3$  и  $G_4$  поступает лог.0 с инверсного выхода триггера, что сохраняет на выходе  $Q=1$ . На выходе элемента  $G_5$  появляется лог.1 обеспечивая сохранение на инверсном выходе  $\bar{Q} = 0$ . При поступлении следующего отрицательного перепада синхросигнала, плечи триггера работают аналогично. Таким образом, при  $J=K=1$  триггер изменяет свое состояние на противоположное каждым отрицательным перепадом синхроимпульса.

### 3. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

**Оборудование и компоненты:** универсальная лабораторная установка IDL-800, интегральные схемы: 1533ЛА3 (74ALS00)-четыре логических элемента 2И-НЕ, 1533ЛЕ1 (74ALS02)-четыре логических элемента 2ИЛИ-НЕ, 1533ЛА3 (74ALS10)-три логических элемента 3И-НЕ, 1533 ТМ2 (74ALS74)-два синхронных динамических D-триггера, 1533 ТМ7(74ALS75)-четыре D-триггера со статическим управлением, 1533 ТВ6 (74ALS107)- два синхронных динамических JK-триггера, 1533ТР2 (74ALS279)-четыре синхронных  $\bar{R}\bar{S}$  – триггера.

#### Часть 1. Построение и исследование триггеров на элементах И-НЕ, ИЛИ-НЕ.

##### 3.1. Асинхронный $\bar{R}\bar{S}$ – триггер на элементах ИЛИ-НЕ.

1. Используя логические элементы ИЛИ-НЕ, собрать асинхронный  $\bar{R}\bar{S}$  – триггер (рис. 6.2).
2. Исследовать работу триггера, подавая на входы триггера различные комбинации входных сигналов. Получить таблицу истинности асинхронного  $\bar{R}\bar{S}$  – триггера.

##### 3.2 Асинхронный $\bar{R}\bar{S}$ -триггер на элементах И-НЕ, $\bar{R}\bar{S}$ – триггер.

1. Используя логические элементы И-НЕ, собрать асинхронный  $\bar{R}\bar{S}$ -триггер (рис. 6.4).

2. Исследовать работу триггера, подавая на его входы различные комбинации входных сигналов. Получить таблицу истинности асинхронного  $\overline{RS}$  – триггера.

### 3.3 Синхронный $\overline{RS}$ – триггер со статическим управлением

1. Используя логические элементы И-НЕ, собрать синхронный  $\overline{RS}$ -триггер со статическим управлением (рис. 6.7).

2. Исследовать работу триггера, подавая на его входы различные комбинации входных сигналов R, S, C. Получить таблицу истинности синхронного  $\overline{RS}$  – триггера со статическим управлением.

### 3.4 Синхронный D-триггер со статическим управлением

1. Используя логические элементы И-НЕ, собрать синхронный D-триггер со статическим управлением (рис. 6.8).

2. Исследовать работу триггера, подавая на входы различные комбинации входных сигналов. Получить таблицу истинности синхронного D – триггера со статическим управлением.

### 3.5 JK-триггер M-S конфигурации

1. Используя логические элементы И-НЕ, собрать двухступенчатый JK-триггер со статическим управлением (рис. 6.11 или рис 6.12).

2. Исследовать работу двухступенчатого JK-триггера, подавая на входы различные комбинации входных сигналов.

**Внимание!** Для подачи синхроимпульса использовать антидребезговую кнопку.

4. Построить таблицы истинности JK-триггера и состояния на выходах всех элементов триггера.

5. Построить временные диаграммы работы JK-триггера для случая J=K=1.

## Часть 2. Исследование модульных триггеров

### 3.6. Асинхронные $\overline{RS}$ – триггеры, 1533 TP2.

1. Вставить ИС 1533 TP2 в наборную панель IDL-800 и подсоединить питание. Условное обозначение TP2 показано на рис. 6.21а.

2. Исследовать работу триггера с одним  $\overline{S}$  входом, получить таблицу истинности.

3. Исследовать работу триггера с двумя  $\bar{S}$  входами, получить таблицу истинности. Сделать выводы.

### 3.7 Синхронный D-триггер с прямым динамическим входом, 1533 TM2.

1. Вставить ИС 1533 TM2 в наборную панель IDL-800 и подсоединить питание. Условное обозначение TM2 показано на рис. 6.21б.

2. Исследовать асинхронные операции установки,  $\bar{S}$  и сброса  $\bar{R}$ . Результаты представить в табличной форме.

3. Исследовать синхронную работу D-триггера. Результаты представить в табличной форме.

4. Преобразовать D-триггер в схему делителя на 2 (рис.6.16). Исследовать работу делителя, построить временные диаграммы работы делителя на 2.

(Для исследования работы делителя на 2 можно воспользоваться также функциональным генератором IDL-800 и осциллографом.)

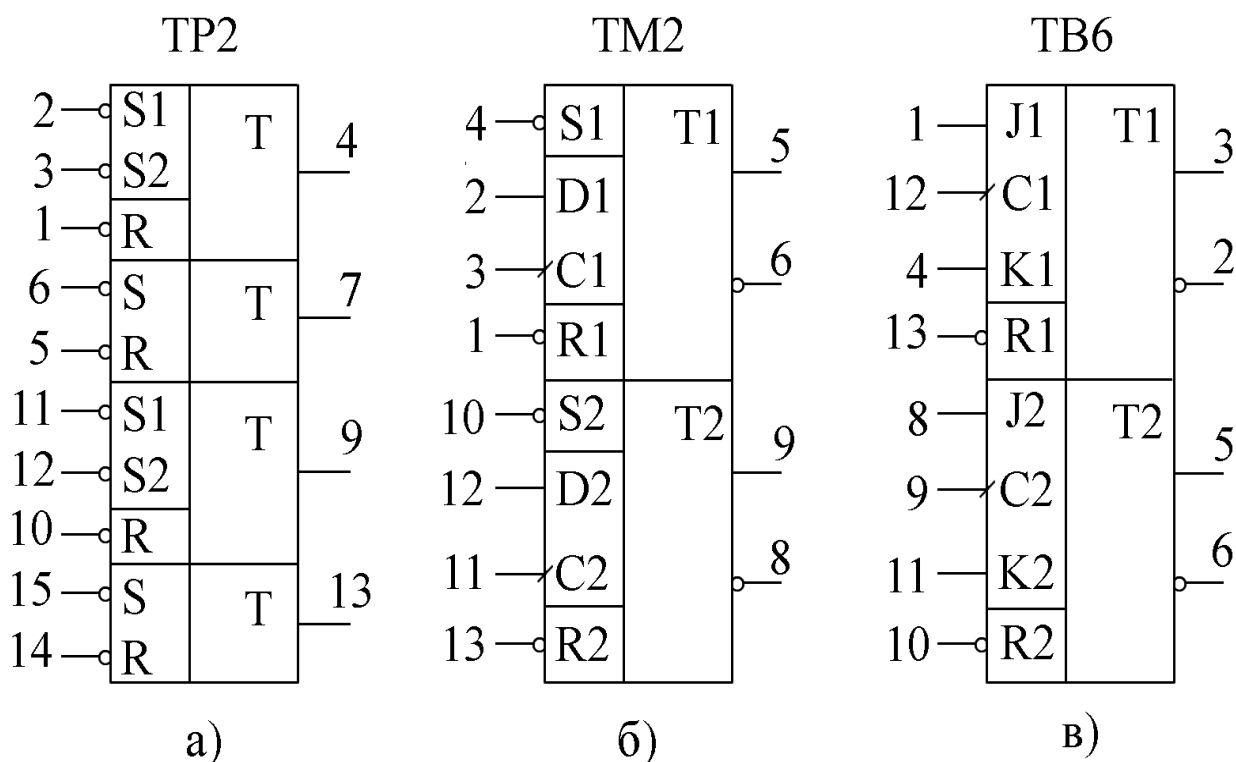


Рис. 6.21

### 3.8 Синхронный JK-триггер с инверсным динамическим входом, 1533 ТВ6.

1. Вставить ИС ТВ6 в наборную панель IDL-800 и подсоединить питание. Условное обозначение ТВ6 показано на рис. 6.21в.
2. Исследовать работу JK-триггера. Результаты представить в табличной форме.
3. Преобразовать JK-триггер в D-триггер. Для этого вход J использовать как D вход и соединить его через инвертор с входом K. Исследовать работу, полученного D-триггера. Результаты представить в табличной форме.

## **4. СОДЕРЖАНИЕ ОТЧЕТА**

- 4.1 Цель работы.
- 4.2 Схемы, исследуемые в работе.
- 4.3 Таблицы и временные диаграммы работы исследуемых триггеров.
- 4.4 Выводы по результатам исследования.

## **5. КОНТРОЛЬНЫЕ ВОПРОСЫ**

- 5.1 Дайте определение триггера.
- 5.2 Какие признаки используются при классификации триггеров?
- 5.3 Что такое таблица истинности или характеристическая таблица триггера?
- 5.4 Что такое таблица переходов триггера?
- 5.5 В чем состоит отличие синхронных триггеров и асинхронных?
- 5.6 Чем различаются синхронные триггеры со статическим управлением от синхронных триггеров с динамическим управлением?
- 5.7 Почему не могут быть построены T- и JK-триггеры со статическим управлением?
- 5.8 Объяснить принцип действия двухступенчатого триггера (M-S триггера).
- 5.9 Преобразуйте RS-, D-, JK-триггер в T-триггер.
- 5.10. Пояснить работу D-триггера с динамическим управлением.
- 5.11. Пояснить работу JK-триггера с динамическим управлением.
- 5.12. В чем заключается различие и сходство RS-триггера и JK-триггера?

## Лабораторная работа №7

### РЕГИСТРЫ И ИХ ПРИМЕНЕНИЕ

#### 1. ЦЕЛЬ ЛАБОРАТОРНОЙ РАБОТЫ

- 1.1. Изучение принципов построения регистров;
- 1.2. Исследование режимов работы и применения регистров.

#### 2. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Триггер может хранить (запоминать) один бит цифровой информации (1 или 0). Его также можно назвать одноразрядным регистром. Группа триггеров, предназначенная для хранения двоичной информации (один триггер на каждый бит информации), называется регистром. Регистры находят применение в различных цифровых устройствах, включая микропроцессоры.

Данные могут вводиться в регистр (записываться) в последовательной форме (бит за битом) или в параллельной форме (все биты одновременно) и могут выводиться из регистра в последовательной или параллельной форме.

Регистры классифицируются в зависимости от того, в какой форме информация вводится в регистр и в какой форме выводится.

Существует четыре возможности:

1. Последовательно – последовательный регистр;
2. Последовательно – параллельный регистр;
3. Параллельно – последовательный регистр;
4. Параллельно – параллельный регистр.

Регистры строятся, используя триггеры (RS, JK, D), и широко представлены как ИС средней степени интеграции.

Регистры, в которые данные вводятся или выводятся в последовательной форме, называются **сдвигающими**. Биты информации, находящиеся в триггерах регистра, сдвигаются то ли вправо, то ли влево при подаче синхроимпульсов. В некоторых регистрах информация может сдвигаться или вправо, или влево, в зависимости от специального управляющего сигнала. Такие регистры называются **реверсивными**. Если регистр может работать во всех четырех режимах и также как реверсивный, то такой регистр называется **универсальным**.

Параллельные регистры предназначены для запоминания и хранения двоичной информации, поэтому параллельные регистры называют **регистрами хранения** или **регистрами памяти**. Такие регистры осуществляют операции записи и считывания информации параллельным кодом. Параллельные регистры могут использоваться в качестве буферных регистров, а также для преобразования прямого двоичного кода в обратный код и наоборот.

При построении параллельных регистров могут использоваться синхронные триггеры со статическим управлением (latch), а также синхронные триггеры с динамическим управлением и двухступенчатые (RS, JK, D) триггеры.

На рисунке 7.1 приведены структуры регистров ИР22 и ИР23. Микросхемы ИР22 и ИР23 – это восьмиразрядные параллельные регистры на D- триггерах. Причем регистр ИР22 построен на D- триггерах со статическим управлением, а ИР23 – с динамическим управлением.

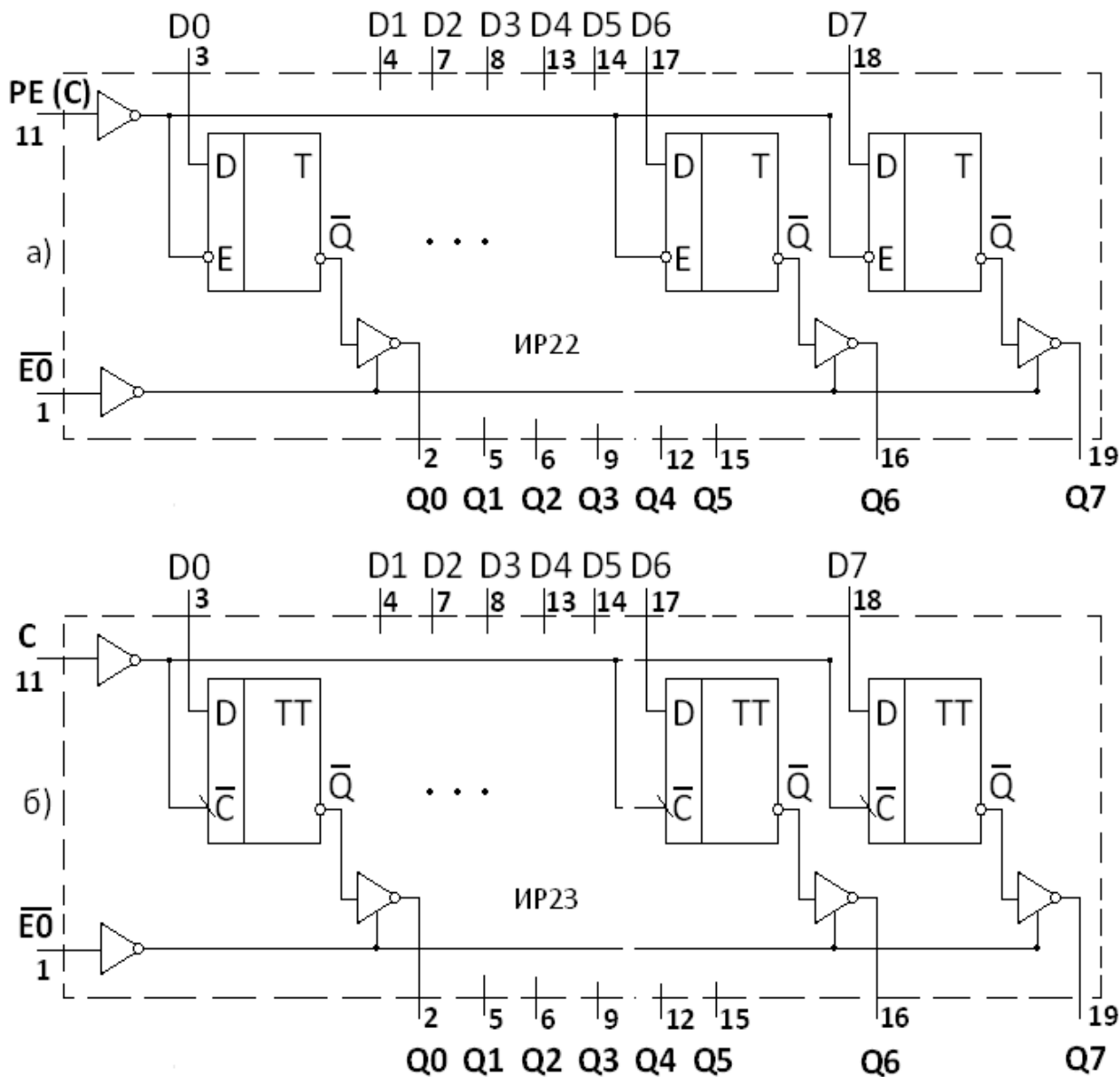


Рис 7.1 Логическая структура регистров ИР22,ИР23

Регистры ИР22 и ИР23 имеют выходные буферные усилители с тремя состояниями. Третье высокоимпедансное состояние  $Z$  можно установить с помощью вывода разрешения  $\overline{EO}$  (Enable Output), если подать на него напряжение высокого уровня. Выходные буферные усилители обладают высокой нагрузочной способностью.

Регистры состоят из восьми D- триггеров с входами разрешения параллельной записи PE (Preset Enable) (для ИР23 – вход С). Если на входе PE действует высокий уровень напряжения, то данные от входов D0 – D7 записываются в триггеры регистра.

Если на вход  $\overline{EO}$  подано напряжение низкого уровня, то данные из D- триггеров регистра пройдут на выходы Q0 – Q7.

Регистр ИР23 принимает информацию синхронно с положительным перепадом тактового импульса, подаваемого на вход С.

В настоящее время выпускается большое количество регистров различного назначения. Однако наиболее универсальными являются регистры, которые могут работать во всех четырех режимах. Примером такого регистра может быть микросхема ИР16.

Микросхема ИР16 – это четырехразрядный регистр сдвига с третьим состоянием выхода. Логическая структура регистра показана на рис. 7.2 . Он построен на четырех синхронных RS- триггерах с инверсным динамическим управлением. RS- триггеры включены по схеме D- триггеров (вход S соединен через инвертор со входом R). На выходах регистра Q0 – Q3 поставлены буферные усилители с повышенной нагрузочной способностью и тремя состояниями. Регистр имеет входы данных D0 – D3 ,вход разрешения параллельной загрузки и сдвига  $\overline{RE}$  , тактовый вход С, вход последовательной загрузки данных S1, вход разрешения выходам EO и выходы Q0 – Q3 .

Если на вход  $\overline{RE}$  ,подать напряжения высокого уровня, то данные от входов D0 – D3 параллельно загружаются в регистр синхронно с отрицательным перепадом импульса синхронизации на входе  $\overline{C}$  . Когда на входе  $\overline{RE}$  действует напряжение низкого уровня, то загрузка данных в регистр происходит последовательно от входа S1, а сдвиг данных вправо от Q3 к Q0 синхронно с каждым отрицательным перепадом тактового импульса на входе  $\overline{C}$  .

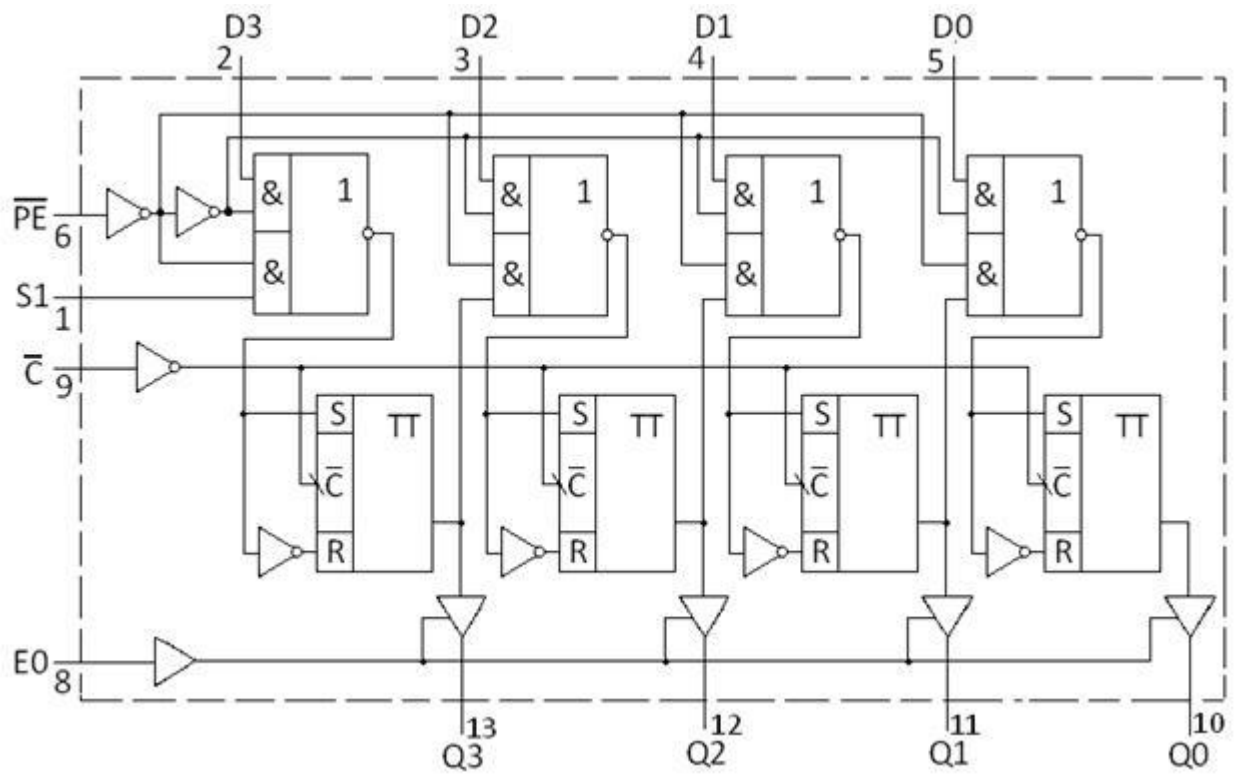


Рис. 7.2 Логическая структура регистра ИР16

Если на вывод разрешения выходам E0 подать напряжение низкого уровня, то выходы Q0 – Q3 перейдут в Z-состояние. На рис. 7.3 показаны условные обозначения и цоколевка ИР22, ИР23 и ИР16.

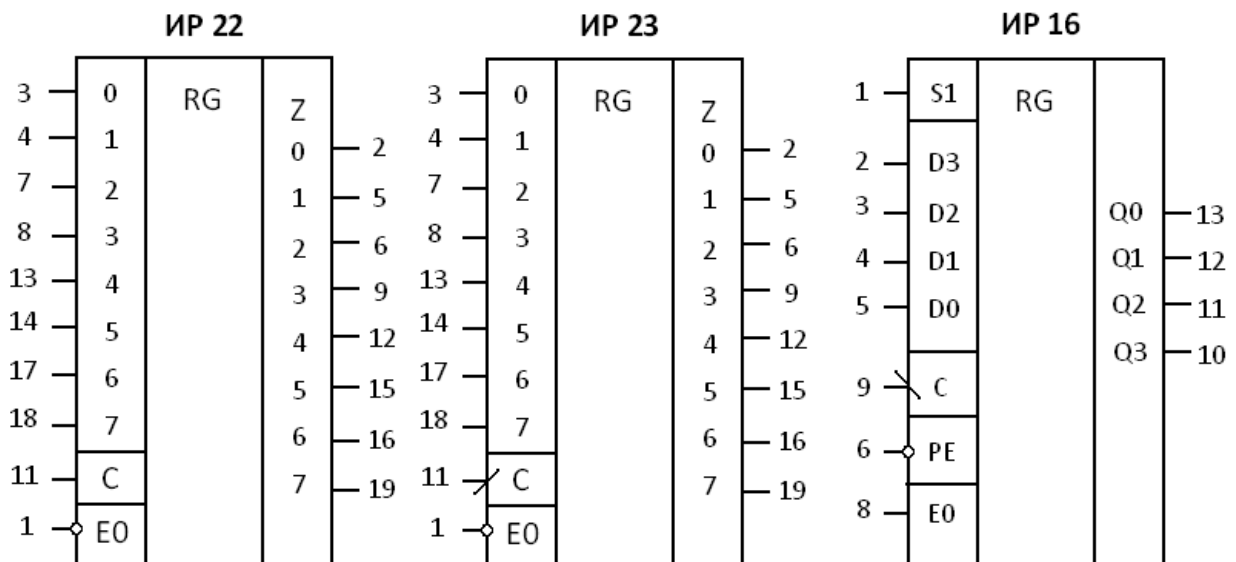


Рис. 7.3 Условные обозначения и цоколевка ИР22, ИР23 и ИР16

Реверсивные регистры сдвига могут осуществлять сдвиг информации как вправо (SR-Shift Right), так и влево (SL-Shift Left), в зависимости от сигнала на входе управления M. Пример построения реверсивного счетчика показан на рис 7.4.



из триггеров, будет циркулировать по регистру при подаче синхроимпульсов. Такая схема называется **кольцевым счетчиком**. На выходах триггеров генерируется непрерывающиеся последовательности импульсов, которые могут быть полезны для различных приложений.

Схема может быть использована для подсчета импульсов. Число сосчитанных импульсов определяется единичным уровнем на выходе соответствующего триггера. Модуль счета такого счетчика равен числу разрядов регистра,  $\text{mod} = N$ . Эта схема может рассматриваться и как делитель на  $N$  ( $N:1$ ).

**Счетчик Джонсона.** Если выход  $\overline{Q0}$  соединить с последовательным входом, то такая схема называется **счетчиком Джонсона**. Если в такой схеме, после обнуления регистра, подать импульсы синхронизации, то на выходах триггеров будут генерироваться сигналы формы меандра.

Счетчик Джонсона – это делитель на  $2N$  или модуль счета такого счетчика  $\text{mod} = 2N$ .

### 3. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

Оборудование и компоненты: универсальная лабораторная установка IDL-800, ИС КР1533ТМ2 (74ALS74) – два синхронных D- триггера с динамическим управлением, ИС КР1533ЛА3 (74ALS00) – четыре логических элемента 2И-НЕ, ИС КР1533ЛА4 (74ALS10) – три логических элемента 3И-НЕ, ИС КР555ИР16 (74LS295) – четырехразрядный регистр сдвига.

**Собрать схему четырехразрядного регистра сдвига, показанную на рис. 7.5, используя две ИС КР1533ТМ2 и ИС КР1533ЛА3**

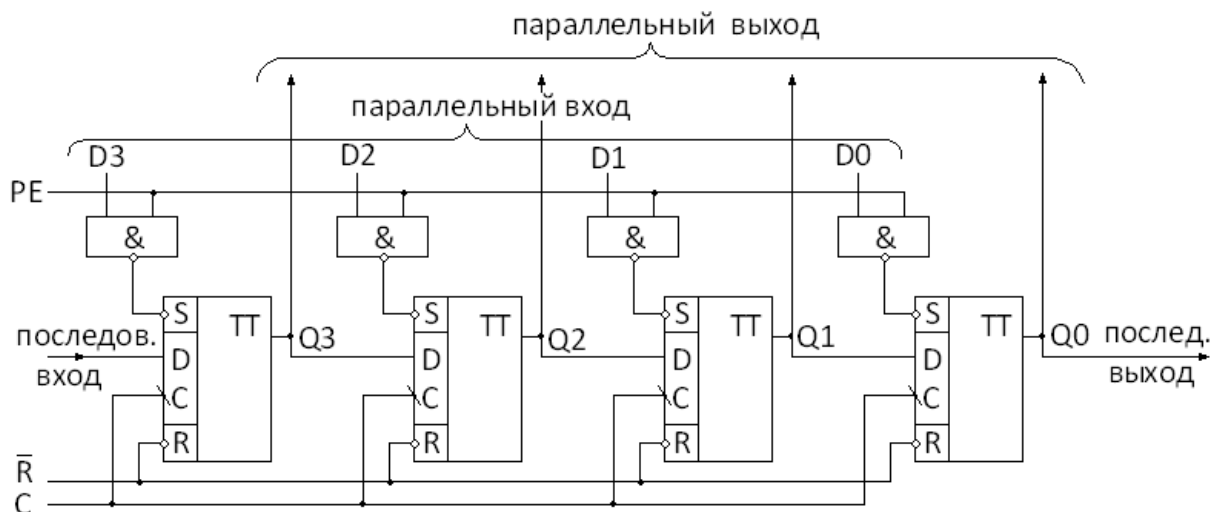


Рис. 7.5 Четырехразрядный регистр сдвига

### 3.2 Исследовать функционирование регистра сдвига, рис. 7.5 . Эта схема может работать во всех четырех режимах.

3.21 Последовательный вход, последовательный / параллельный выход. Исследовать работу регистра, подавая на его вход следующее четырехразрядное слово: 1011. Для любого другого слова работа регистра будет аналогична. На вход PE подать низкий уровень напряжения, затем обнулить регистр. Для этого на вход  $\bar{R}$  подать низкий уровень напряжения, а затем установить  $\bar{R} = 1$ . Теперь, подавая импульсы синхронизации (использовать антидребезговую кнопку), проследить, как данные с последовательного входа будут сдвигаться в регистр. Наблюдая за состоянием выходов регистра, убедиться, что данные соответствуют временным диаграммам рис. 7.6 .

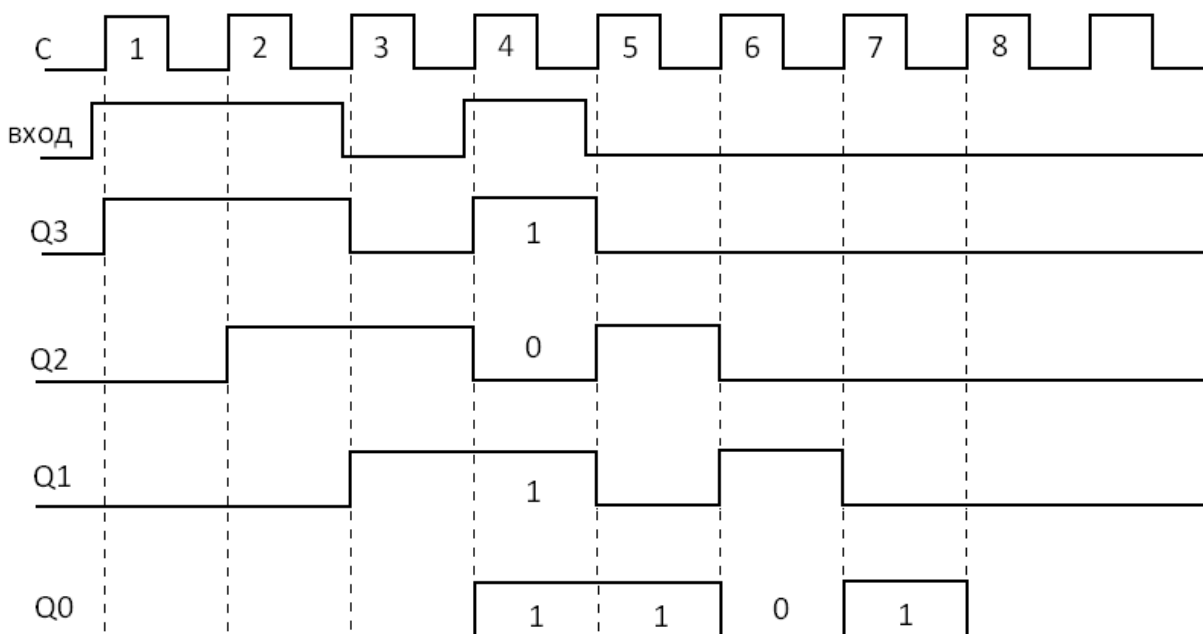


Рис. 7.6 Временные диаграммы работы регистра: последовательный вход, последовательный / параллельный выход

3.22 Параллельный вход, последовательный / параллельный выход. Вначале обнулим регистр и установим  $R = 1$ . Установим входы  $D3 = 1, D2 = 0, D1 = 1$  и  $D0 = 1$ , а затем на вход PE подадим высокий уровень напряжения и данные с входов  $D3, D2, D1, D0$  запишутся в регистр. Теперь данные доступны в параллельном виде на выходах триггеров регистра. Подадим на синхровход регистра импульсы синхронизации и на выходе  $Q0$  получим выходные данные в последовательном виде, рис. 7.7 .

3.23 Повторить п.2.1 и п.2.2 для входных данных 1101. Построить временные диаграммы.

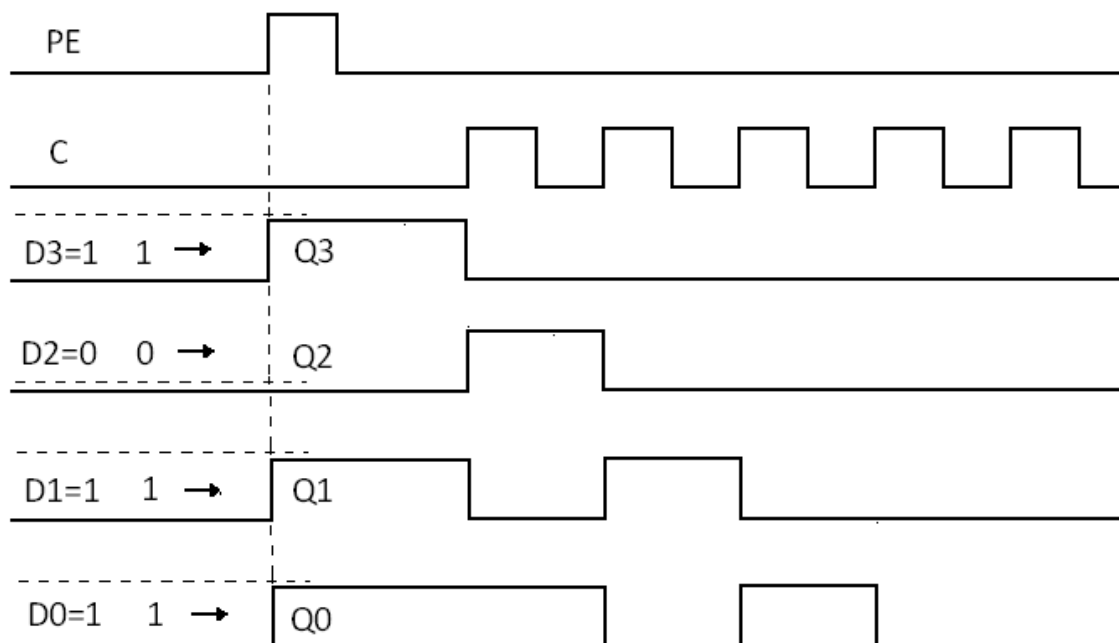


Рис. 7.7 Временные диаграммы, поясняющие работу регистра в режиме: последовательный вход, последовательный / параллельный выход.

### 3.3 Исследование кольцевого счетчика.

3.31 На основе регистра сдвига рис 7.5 построить кольцевой счетчик. Для этого соединить последовательный выход регистра (выход Q0) с последовательным входом.

3.32. Для исследования кольцевого счетчика вначале обнулить регистр, затем записать в него 0001 в параллельном виде и подать синхросимпульсы. Результаты наблюдения табулировать, табл. 7.1.

Таблица 7.1

C	Q3	Q2	Q1	Q0
0				
1				
2				
3				
4				
5				
6				
7				

3.33 Построить временные диаграммы работы кольцевого счетчика, сделать выводы.

### **3.4 Исследование счетчика Джонсона.**

3.41 На основе регистра сдвига рис. 7.5 построить счетчик Джонсона. Для построения счетчика Джонсона необходимо соединить инверсный выход  $\overline{Q_0}$  с последовательным входом регистра.

3.42 Для исследования работы счетчика Джонсона вначале обнулить регистр, а затем подать синхроимпульсы. Результаты наблюдений табулировать (табл. 7.1).

3.43 Построить временные диаграммы работы счетчика Джонсона, сделать выводы.

## **4. СОДЕРЖАНИЕ ОТЧЕТА**

- 4.1 Цель работы.
- 4.2 Схемы исследуемых в работе устройств.
- 4.3 Таблицы и временные диаграммы, отражающие результаты исследований.
- 4.4 Выводы по результатам исследований.

## **5. КОНТРОЛЬНЫЕ ВОПРОСЫ**

- 5.1. Объяснить работу регистров во всех четырех режимах работы.
- 5.2 . Как строить реверсивный регистр сдвига?
- 5.3 . Как строится кольцевой счетчик?
- 5.4 . Чему равен модуль счета кольцевого счетчика?
- 5.5 . Как строится счетчик Джонсона?
- 5.6 . Чему равен модуль счета счетчика Джонсона?

## Лабораторная работа №8

### ГЕНЕРАТОРЫ ПОСЛЕДОВАТЕЛЬНОСТЕЙ.

#### 1. ЦЕЛЬ ЛАБОРАТОРНОЙ РАБОТЫ

- 1.1 Изучение синтеза генераторов последовательностей;
- 1.2 Исследование работы генераторов последовательностей.

#### 2. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Схема, которая генерирует заданную последовательность бит синхронно с импульсами синхронизации, называется генератором последовательностей. Такие генераторы используются как:

1. Счетчики;
2. Генераторы псевдослучайных последовательностей;
3. Генераторы заданной последовательности и заданного периода;
4. Генераторы кодов.

Генераторы последовательностей являются одним из наиболее интересных применений регистров сдвига. Блок-схема генератора последовательностей показана на рисунке 8.1

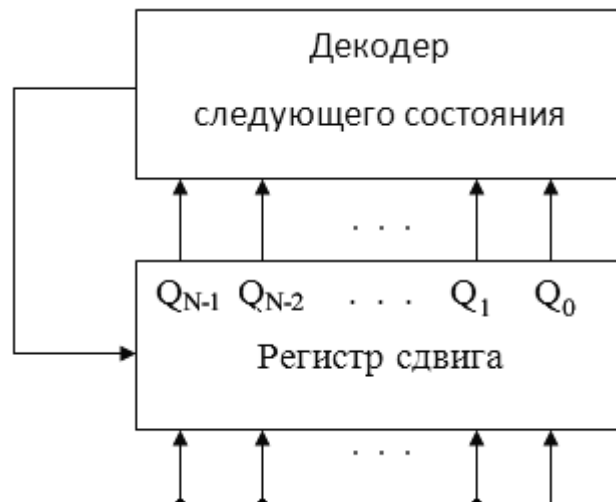


Рис. 8.1 Блок-схема генератора последовательностей

Выход декодера следующего состояния  $Y$  - это функция от  $Q_{N-1}$ ,  $Q_{N-2}$ , ...,  $Q_0$ , т.е.  $Y = f(Q_{N-1}, Q_{N-2}, \dots, Q_0)$ . Это схема подобна схеме кольцевого счетчика ( $Y=Q_0$ ) или счетчика Джонсона ( $Y = \overline{Q_0}$ ). Кольцевой счётчик и счётчик Джонсона являются частными случаями генераторов последовательностей.

Синтез генераторов последовательностей рассмотрим на примерах.

Пример 1. Синтезировать генератор следующей последовательности  
... 1101011 ...

Решение. При синтезе генератора заданной последовательности необходимо определить число разрядов регистра сдвига и комбинационную схему декодера следующего состояния.

Минимально возможное число триггеров  $N$  в регистре сдвига для генерирования последовательности длиной  $S$  бит определяется из того, что

$$N \geq \log_2(S + 1) \quad (8.1)$$

В данном примере  $S=7$ , поэтому минимально возможное значение  $N=3$ . Однако это не значит, что это число триггеров является достаточным. Если данная последовательность ведет к семи различным состояниям регистра, то тогда три триггера будет достаточно, в противном случае число триггеров придется увеличить. Запишем состояние регистра в виде таблицы 8.1.

Таблица 8.1

Число синхроимпульсов	Выходы триггеров		
	$Q_2$	$Q_1$	$Q_0$
1	1	1	1
2	1	1	1
3	0	1	1
4	1	0	1
5	0	1	0
6	1	0	1
7	1	1	0

Допускаем, что данная последовательность генерируется на выходе  $Q_2$ . В таком случае на выходах  $Q_1$  и  $Q_0$  будет та же последовательность, только задержанная на один и два такта, соответственно. Из таблицы 8.1 видно, что не все состояния регистра отличаются от других (первая и вторая строки одинаковы, а также четвертая и шестая), что означает - число триггеров регистра  $N=3$  не является достаточным. Поэтому примем число  $N=4$  и построим таблицу аналогичную табл. 8.2.

Поскольку в данном случае все состояния регистра являются отличными (разными), добавим в эту таблицу колонку, в которой запишем требуемую последовательность на входе регистра,  $Y$ .

Таблица 8.2

Число синхроимпульсов	Выходы триггеров				Y
	Q <sub>3</sub>	Q <sub>2</sub>	Q <sub>1</sub>	Q <sub>0</sub>	
1	1	1	1	0	1
2	1	1	1	1	0
3	0	1	1	1	1
4	1	0	1	1	0
5	0	1	0	1	1
6	1	0	1	0	1
7	1	1	0	1	1
8 (1)	1	1	1	0	1

Схему декодера получим, упрощая функцию  $Y = f(Q_3, Q_2, Q_1, Q_0)$  с помощью карты Карно, рис 8.1

		Q <sub>1</sub> Q <sub>0</sub>			
	Q <sub>3</sub> Q <sub>2</sub>	00	01	11	10
00		X	X	X	X
01		X	1	1	X
11		X	1	0	1
10		X	X	0	1

$$Y = \overline{Q_3} + \overline{Q_1} + \overline{Q_0} = \overline{Q_3 \cdot Q_1 \cdot Q_0}$$

Рис. 8.1. Упрощение ФАЛ декодера следующего состояния

Упрощенная схема генератора последовательности ... 1101011 ... показана на рис. 8.2

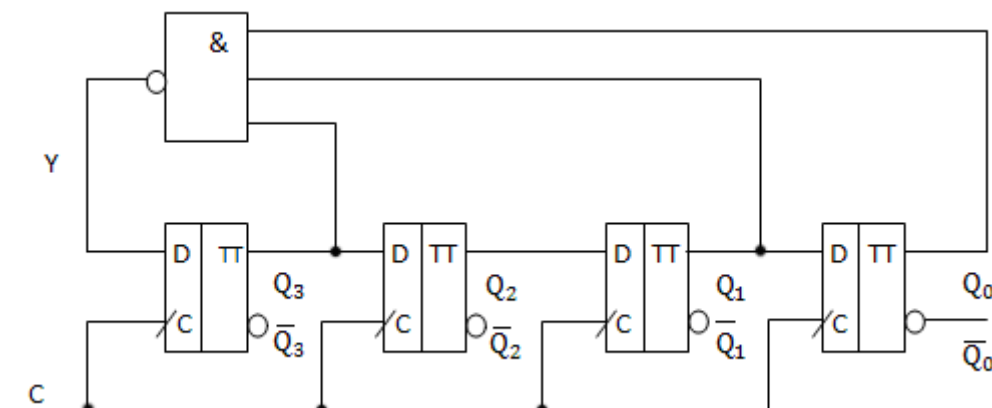
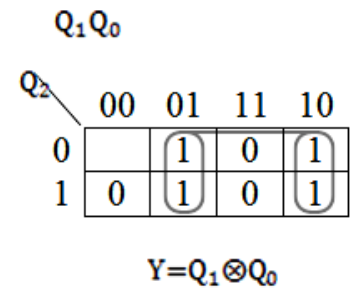


Рис. 8.2 Логическая структура генератора последовательности ... 1101011 ...

Пример 2. Синтезировать генератор последовательности ... 1101001... .  
Решение. Минимально возможное число триггеров регистра  $N=3$ . Проверим является ли это число достаточным. Для этого построим таблицу 8.3.

Таблица 8.3

Импульсы синхронизации	Выходы триггеров $Y$			
	$Q_2$	$Q_1$	$Q_0$	
1	1	1	1	0
2	0	1	1	0
3	0	0	1	1
4	1	0	0	0
5	0	1	0	1
6	1	0	1	1
7	1	1	0	1



Из таблицы 8.3 видно, что все состояния регистра являются неодинаковыми, т.е. число триггеров  $N=3$  является достаточным для реализации генератора, и поэтому добавим в эту таблицу колонку с требуемой на входе регистра последовательностью. Схему декодера получим, упрощая функцию  $Y = f(Q_2, Q_1, Q_0)$  с помощью карты Карно.

На рисунке 8.3 показана упрощенная схема генератора последовательности ... 1101001... .

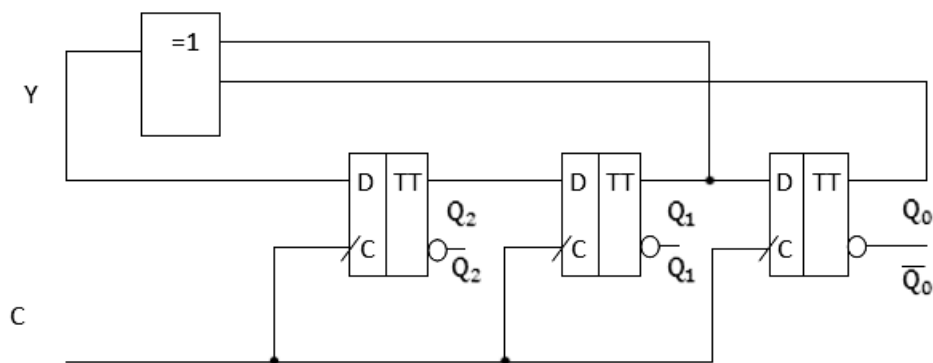


Рис. 8.3 Логическая структура генератора последовательности ... 1101001... .

Длина последовательности, генерируемой генератором, рис. 8.3, равна  $S = 2^N - 1 = 2^3 - 1 = 7$ .

Генераторы, которые генерируют последовательности длины

$$S = 2^N - 1 \quad (8.2)$$

называются генераторами последовательностей максимальной длины. Такие генераторы широко используются для генерирования помехоустойчивых кодов.

### **3. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ**

Оборудование и компоненты: универсальная лабораторная установка IDL-800, ИС 1533ИР16 (74ALS295)- четырехразрядный регистр сдвига, ИС 1533ЛА4 (74ALS10)- три логических элемента 3И-НЕ, ИС 1533ЛП5 (74ALS86)- четыре двухвходовых логических элемента ИСКЛЮЧАЮЩЕЕ ИЛИ.

#### **3.1 Исследование работы генератора последовательности, рис 8.2.**

1. Используя ИС 1533ИР16 и 1533ЛА4, собрать схему генератора последовательности.
2. Подавая одиночные синхроимпульсы ( использовать антидребезговую кнопку), исследовать работу генератора последовательности.
3. Результаты представить в табличной форме.

#### **3.2 Исследование работы генератора последовательности, рис. 8.3.**

1. Используя ИС К555ИР16 и 1533ЛП5, собрать схему генератора последовательности.
2. Подавая одиночные синхроимпульсы, исследовать работу генератора.
3. Результаты представить в табличной форме.

#### **3.3 Синтез и исследование генератора последовательности ...1001011....**

1. Синтезировать и построить генератор последовательности ...1001011....
2. Исследовать работу генератора, результаты предоставить в табличной форме.

#### **3.4 Синтезировать и исследовать генератор последовательности, заданной преподавателем.**

### **4. СОДЕРЖАНИЕ ОТЧЕТА**

- 4.1. Цель работы.
- 4.2. Схемы генераторов исследуемых в работе.
- 4.3. Таблицы результатов наблюдений.
- 4.4. Выводы.

## 5. КОНТРОЛЬНЫЕ ВОПРОСЫ

5.1. Чему равняется минимально возможное число триггеров для генератора последовательности длиной  $S=27$ ?

5.2. Какие генераторы называются генераторами последовательности максимальной длины?

5.3. Привести примеры генераторов последовательности максимальной длины.

5.4. Объяснить работу генераторов, исследованных в работе.

## Лабораторная работа №9

### ИССЛЕДОВАНИЕ АСИНХРОННЫХ СЧЕТЧИКОВ

#### 1. ЦЕЛЬ ЛАБОРАТОРНОЙ РАБОТЫ

- изучение методов построения и исследование функционирования основных типов асинхронных счётчиков.

#### 2. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Цифровой счётчик - это группа триггеров соединённых так, чтобы считать число импульсов поданных на вход и фиксировать число подсчитанных импульсов в том или ином коде.

Основными характеристиками счётчика являются коэффициент (модуль) счёта и быстродействие.

Коэффициент или модуль счёта характеризует число устойчивых состояний счётчика.

Быстродействие счётчика зависит от используемой элементной базы и схемы построения.

Счётчики классифицируются по ряду признаков:

- по быстродействию и способу организации внутренних связей: асинхронные и синхронные;
- по направлению счёта: суммирующие, вычитающие, реверсивные;
- по модулю счёта: двоичные, двоично-десятичные или другим модулем счёта.

Классификационные признаки независимы и могут встречаться в разных сочетаниях.

В наиболее общем случае рассматриваются асинхронные и синхронные счётчики. В случае асинхронных счётчиков триггеры перебрасываются не одновременно, а последовательно и в случае синхронных счётчиков триггеры перебрасываются одновременно. Основным достоинством асинхронных счётчиков являются их схемная простота, а недостатком – низкое быстродействие. Основным достоинством синхронных счётчиков является их более высокое быстродействие, а недостатком – более сложная схемная реализация.

Рассмотрим построение асинхронных счётчиков. Для этого рассмотрим счётную, последовательность, табл. 9.1. Число состояний в такой последовательности 8, что требует использования трех триггеров ( $2^3=8$ ).

Таблица 9.1

Счёт	Состояние счётчика		
	Q <sub>2</sub>	Q <sub>1</sub>	Q <sub>0</sub>
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
8(0)	0	0	0

Выход Q<sub>0</sub> наименее значащего триггера (разряда) изменяться каждым счётным импульсом. Это может быть реализовано путём использования триггера Т-типа, при T=1. Состояние на выходе Q<sub>1</sub> меняется всякий раз, когда выход Q<sub>0</sub> меняется из 1 в 0. Поэтому, если выход Q<sub>0</sub> соединить с входом синхронизации следующего Т-триггера с инверсным динамическим входом, то выход Q<sub>1</sub> будет менять состояние всякий раз, когда выход Q<sub>0</sub> осуществляет переход из 1 в 0 (отрицательный перепад синхроимпульса). Таким же образом переход из 1 в 0 с выхода Q<sub>1</sub> изменяет состояние Q<sub>2</sub>, что достигается путём соединения Q<sub>1</sub> с синхровходом следующего триггера. Аналогичным образом может быть построен асинхронный счётчик с большим числом разрядов или модулем счёта.

На рис.9.1а показана схема асинхронного счётчика с использованием триггеров с инверсным динамическим входом (ТВ6). Входы J и K соединены вместе, образуя вход Т-триггера, и на них подана лог.1.

На рис.9.1б показаны временные диаграммы, поясняющие работу счётчика. На прямых входах триггеров (Q<sub>2</sub>,Q<sub>1</sub>,Q<sub>0</sub>) отражается состояние суммирующего счётчика, когда с каждым счётным импульсом состояние счётчика увеличивается. В то же самое время, этот счётчик можно рассматривать как вычитающий, если информацию о состоянии счётчика снимать с инверсных выходов триггеров (Q<sub>2</sub>,Q<sub>1</sub>,Q<sub>0</sub>).

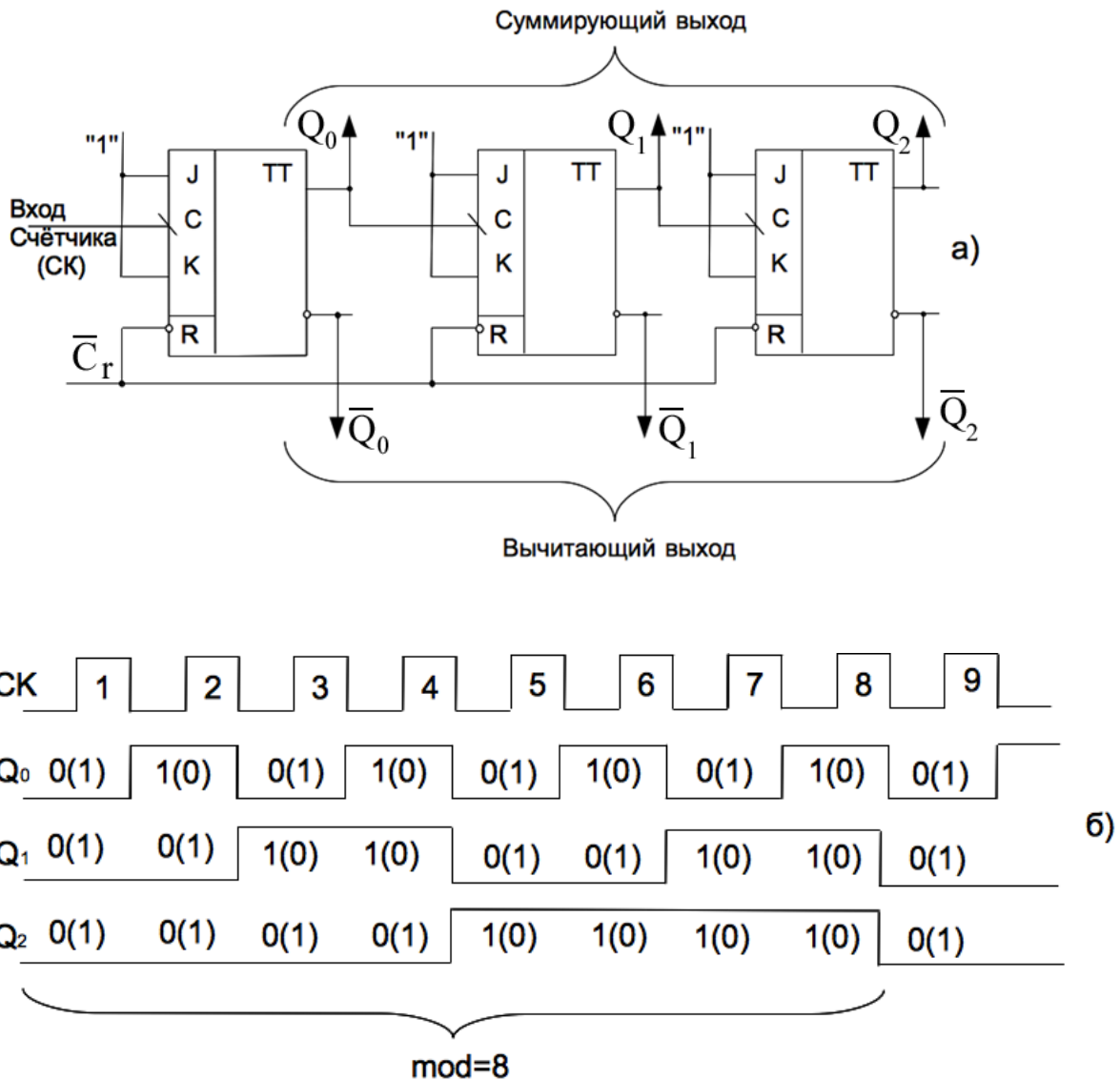


Рис. 9.1 Асинхронный счётчик на триггерах с инверсным динамическим синхровходом

На рис.9.2а показан вариант схемы асинхронного счётчика на тех же триггерах с инверсным динамическим входом. На этот раз инверсные входы триггеров соединены с синхровходом последующих триггеров. В этой схеме суммирующий счётчик получается, если снимать информацию о состоянии счётчика с инверсных входов триггеров ( $Q_2, Q_1, Q_0$ ) и вычитающий счётчик, если снимать информацию с прямых выходов триггеров ( $Q_2, Q_1, Q_0$ ).

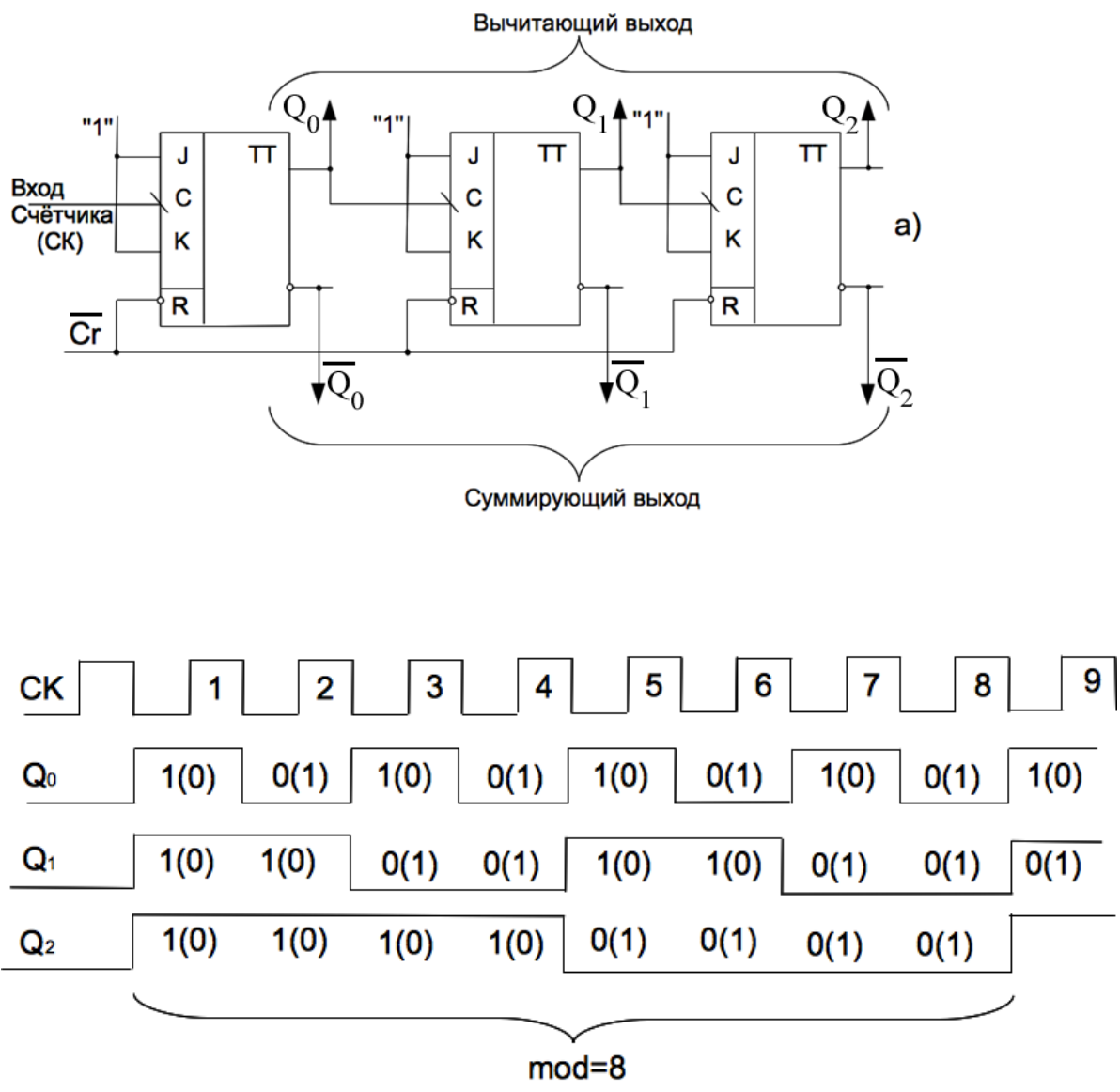


Рис.9.2 Асинхронный счётчик на триггерах с инверсным динамическим синхровходом

Рассмотрим теперь, как строятся асинхронные счётчики на основе триггеров с прямым динамическим входом (например ТМ2). На рис.9.3 показаны схемы и временные диаграммы, поясняющие работу счётчика. Если в счётчике на триггерах с прямым динамическим синхровходом прямые выходы триггеров соединены с синхровходами последующих, то суммирующий и вычитающий выходы счётчика поменяются местами.

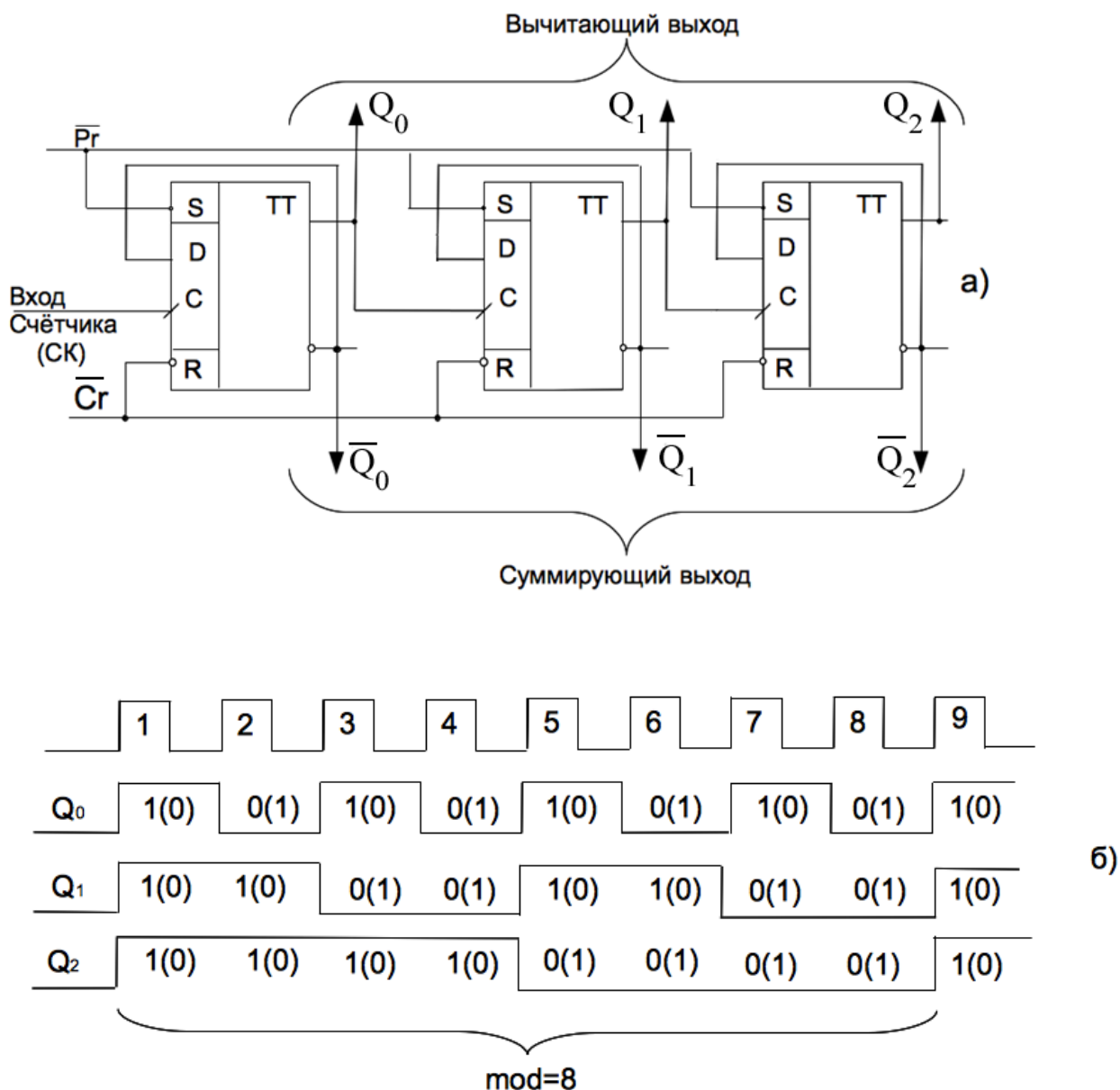


Рис 9.3 Асинхронный счётчик на триггерах с прямым динамическим синхровходом.

Асинхронные счётчики построенные на двухступенчатых триггерах М-S типа, которые перебрасываются отрицательным перепадам синхроимпульса, работают аналогично счётчикам на триггерах с инверсным динамическим синхровходом .

Реверсивные счётчики. Счётчики способные работать как в прямом, так и обратном направлении называются реверсивными. При построении реверсивного счётчика изменение направления счёта достигается переключением межразрядных связей. На рис.9.4 показан пример построения асинхронного реверсивного счётчика.

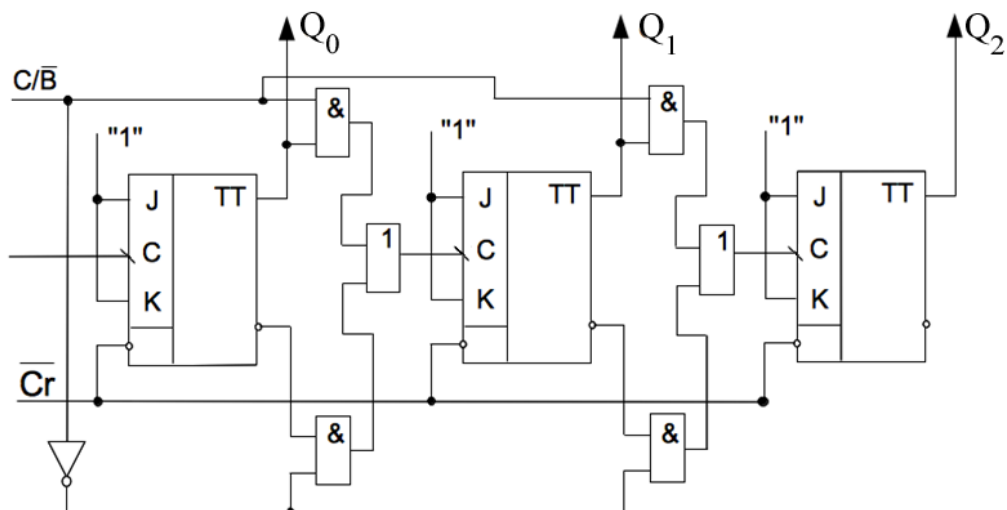


Рис. 9.4 Реверсивный асинхронный счётчик

В зависимости от сигнала на управляющем входе  $C/\bar{B}$  счётчик (рис 9.4) может осуществлять счёт в прямом или обратном направлении.

### Счётчики с произвольным модулем счёта.

Счётчики, которые мы рассмотрели, имеют модуль счёта,  $\text{mod}=2^N$  (2, 4, 6, 8, 16...). Однако при проектировании цифровых устройств часто возникают необходимости построить счётчик с модулем счёта,  $\text{mod}\neq 2^N$ . Принцип построения таких счётчиков состоит в исключении избыточных состояний с помощью включения обратных связей внутри счётчика либо методом управляемого сброса в нуль, когда в нём устанавливается определённое состояние.

Итак, если необходимо получить счётчик с каким-то модулем счёта,  $\text{mod}$ , то число триггеров необходимых при этом, определяется исходя из  $\text{mod}\leq 2^N$ .

Например, при  $N=4$  любой  $\text{mod}$  в пределах от 9 до 16 может быть получен. Если необходимо получить счётчик с  $\text{mod}=10$ , то шесть состояний не используется. При этом, если счёт осуществляется в натуральном ВСД коде, то счётчик будет проходить последовательно состояния как показано на рис.9.5. И это, подобно двоичному счётчику

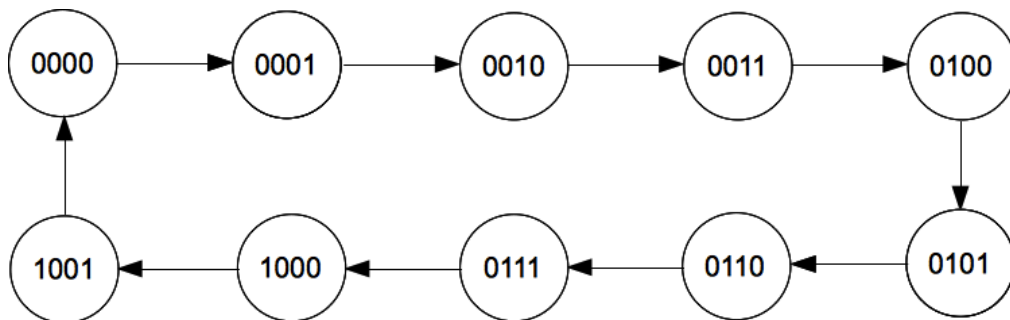


Рис. 9.5

за исключением того, что после состояния 1001 (код десятичного числа 9) следующим состоянием будет 0000 (код десятичного числа 0). Синтез десятичного асинхронного счётчика, или любого другого асинхронного счётчика с  $\text{mod} \neq 2^N$ , или счётчика работающего не в прямом двоичном коде является сложной проблемой, поскольку не существует прямой процедуры синтеза.

Рассмотрим в качестве примера построение и работу двоично-десятичного асинхронного счётчика, рис.9.6.

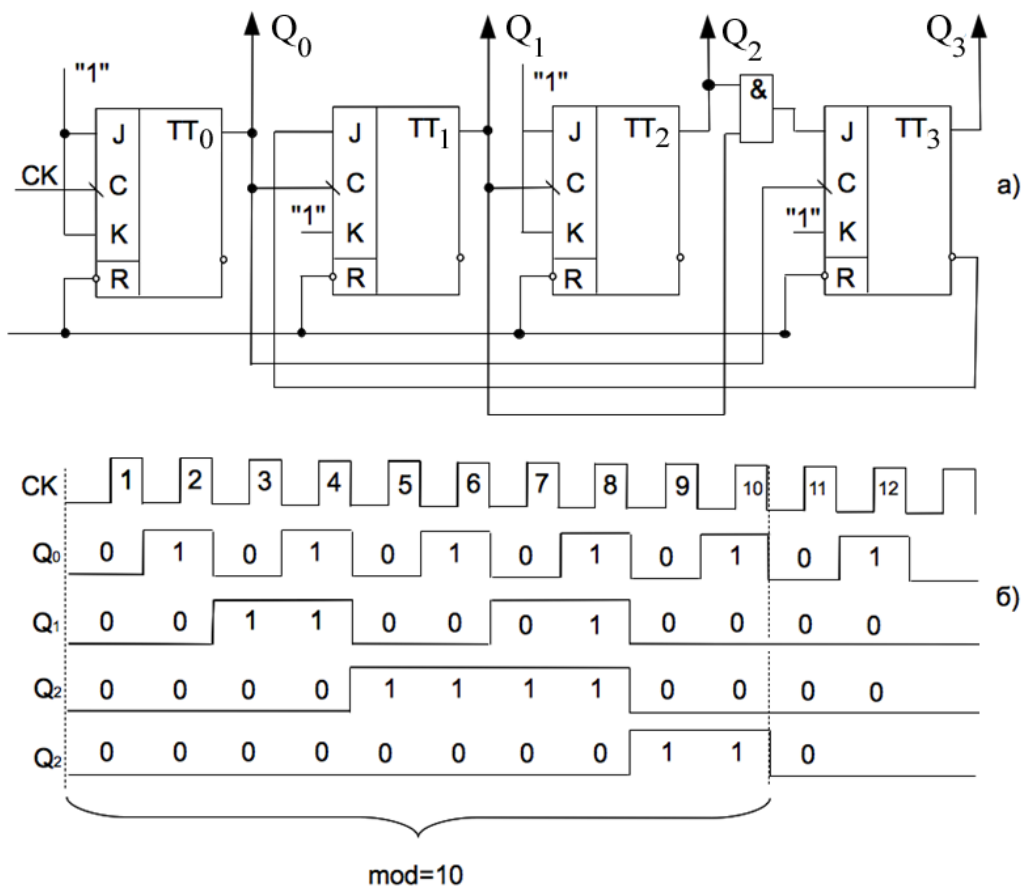


Рис. 9.6 Асинхронный двоично-десятичный счётчик

Триггеры в счётчике, рис.9.6 перебрасываются с отрицательным перепадом синхросигнала. В схеме счётчика выход  $Q_0$  соединён с синхровходами двух триггеров ТТ<sub>1</sub> и ТТ<sub>3</sub>, а вход  $Q_1$  соединён с синхровходом ТТ<sub>2</sub>. Входы J и K соединены либо постоянно с лог.1, либо с выходами триггеров, как показано на рис.9.6а. Работа счётчика может быть объяснена исходя из условий переброса триггеров. Вспомним, что, если синхроимпульс имеет переход из 1 в 0, триггер устанавливается в единичное состояние при J=1

и  $K=0$ , обнуляется при  $J=0$  и  $K=1$ , меняет своё состояние на противоположное при  $J=K=1$  и не изменяет своё состояние при  $J=K=0$ .

Из схемы рис.9.6а очевидно, что:

1. Выход  $Q_0$  меняет своё состояние каждым отрицательным перепадом импульсов СК.
2. Выход  $Q_1$  меняет своё состояние, если  $Q_3=0$  и  $Q_0$  имеет переход из 1 в 0.
3. Выход  $Q_2$  меняет своё состояние всякий раз когда  $Q_1$  имеет переход из 1 в 0.
4. Выход  $Q_3$  меняет своё состояние, когда  $Q_1Q_2=1$  и  $Q_0$  имеет переход из 1 в 0. Выход  $Q_3$  обнуляется, если  $Q_1$  или  $Q_2$  равны 0 и  $Q_0$  имеет переход из 1 в 0.

Временные диаграммы, поясняющие работу двоично-десятичного асинхронного счётчика, приведены на рис.9.6б.

### Интегральные схемы асинхронных счётчиков.

Мы рассмотрели построение асинхронных счётчиков, используя отдельные триггеры. Ряд асинхронных счётчиков выпускаются промышленностью как интегральные схемы средней степени интеграции. Все ИС состоят из четырёх М-*S* триггеров. Загрузка, установка и сброс (обнуление) осуществляются асинхронно, т.е. независимо от импульсов синхронизации.

В зависимости от особенностей счётчиков, связанных с загрузкой, установкой и сбросом, эти счётчики условно разделяются на три группы.

К первой группе относится асинхронный счётчик 1533 ИЕ2 (74ALS90), который имеет входы установки и сброса. Блок-схема счётчика показана на рис.9.7. Счётчик состоит из четырёх триггеров объединённых внутри, как счётчик по  $mod=2$  и счётчик по  $mod=5$ . Эти счётчики могут использоваться

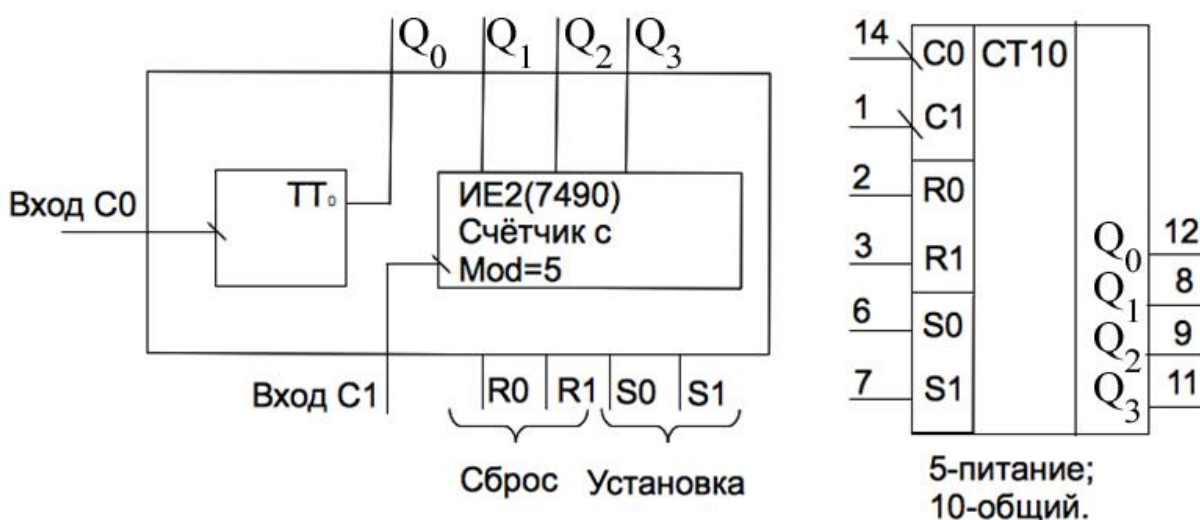


Рис.9.7 Асинхронный счётчик ИЕ2

независимо или в комбинации. Если выход счётчика с  $\text{mod}=2$  ( $Q_0$ ) соединить со входом счётчика по  $\text{mod}=5$ , то образуется двоично-десятичный счётчик. Если соединить выход счётчика с  $\text{mod}=5$  ( $Q_3$ ) со входом счётчика  $\text{mod}=2$ , то образуется счётчик-делитель на 10. В счётчике имеются два входа «сброс»  $R_0$  и  $R_1$ , на которые необходимо подать лог.1 для обнуления счётчика. Когда на оба входа «Установка»  $S_0$  и  $S_1$  подаётся лог.1 двоично-десятичный счётчик устанавливается в состояние 1001.

Внутренняя схема и временные диаграммы работы счётчика ИЕ2 соответствуют двоично-десятичному счётчику, рассмотренному ранее (рис.9.6).

Ко второй группе асинхронных счётчиков можно условно отнести счётчики которые имеют только входы сброса. Блок схема счетчиков 155ИЕ4, 1533ИЕ5 показана на рис.9.8.

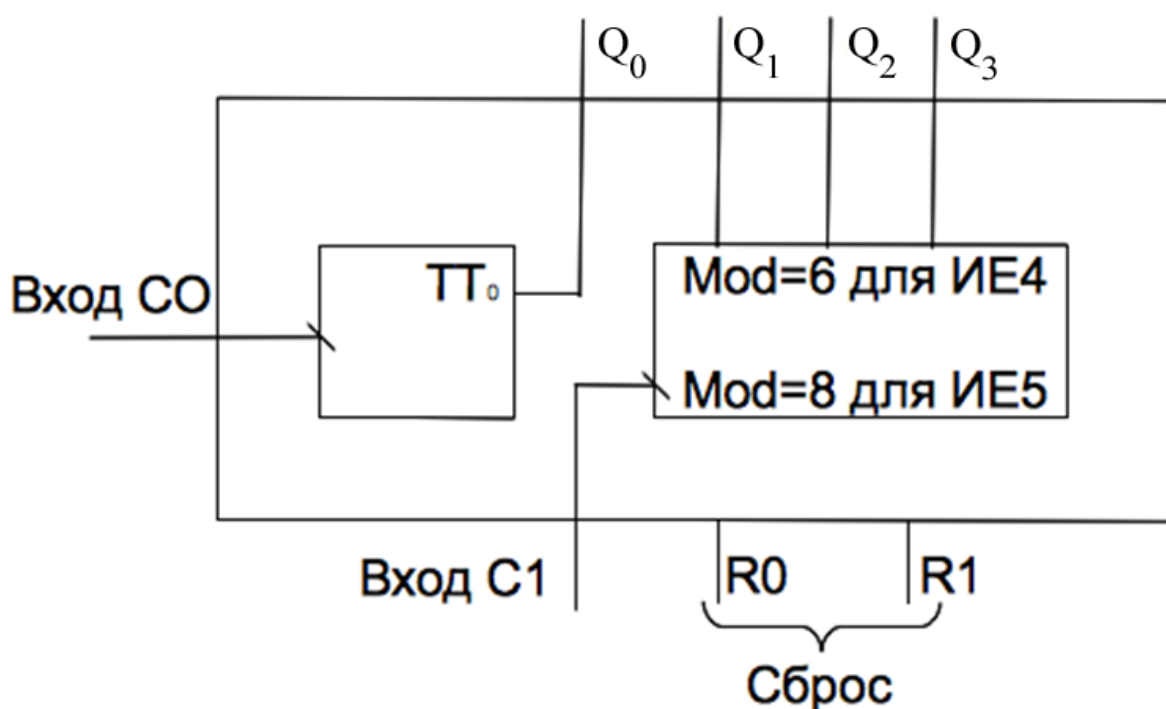


Рис.9.8 Блок-схема асинхронных счетчиков ИЕ4, ИЕ5.

Работа этого счетчика аналогична ИЕ2, за исключение того, что входы установки отсутствуют и счетчик по  $\text{mod}=6$  осуществляет счёт не в натуральном двоичном коде. Эта последовательность дана в табл. 9.2

Таблица 9.2

$Q_3$	$Q_2$	$Q_1$
0	0	0
0	0	1
0	1	0
1	0	0
1	0	1
1	1	0

ИС555ИЕ20 (74LS390) содержит два двоично-десятичных счётчика, подобных ИЕ2. Для каждого счётчика имеется один вход «Сброс» ( $R$ ). ИС1533ИЕ19 (74ALS393) содержит два 4 – разрядных двоичных счётчика с входом «Сброс» ( $R$ ).

К третьей группе асинхронных счётчиков относится счётчики 555ИЕ14 (74LS176, 74LS196 ) и 555ИЕ15 (74LS177, 74LS197). Эти счётчики являются версиями счётчиков ИЕ2 и ИЕ5 с предварительной установкой состояния. Блок схема счётчиков показана на рис. 9.9.

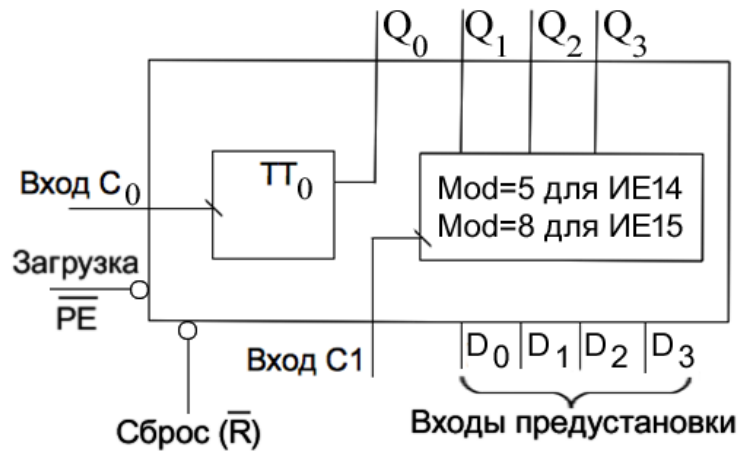


Рис. 9.9 Блок-схема асинхронных счётчиков ИЕ14, ИЕ15

Счётчики обнуляются при подаче лог.0 на вход  $\bar{R}$ . Подача лог.0 на вход  $\bar{PE}$  (в это время на входе «Сброс» должно быть  $\bar{R} = 1$ ) останавливает счёт и в счётчик загружается число, установленное на входах предустановки. В режиме счёта на оба входа «Сброс» и «Загрузка» должны быть поданы лог.1.

Условное обозначение и цоколёвка ИС ИЕ4, ИЕ5, ИЕ14, ИЕ15 показаны на рис.9.10.

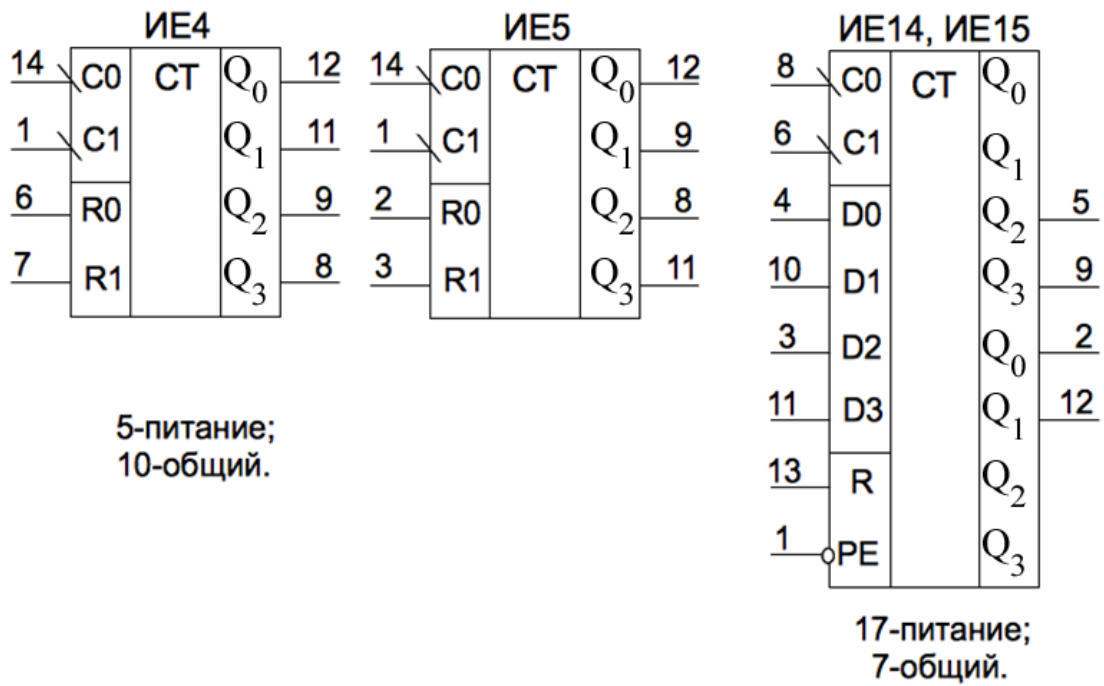


Рис. 9.10 Условные обозначения ИС ИЕ4, ИЕ5, ИЕ14, ИЕ15

Входы асинхронных счётчиков «Загрузка», «Установка», «Сброс», могут использоваться для изменения модуля и последовательности счёта. Например, на рис. 9.11а, показана схема счётчика с  $\text{mod} = 12$ , построенная на основе ИС ИЕ5. На рис.9.11б показана схема десятичного счетчика, последовательность состояний которого соответствуют коду с избытком 3.

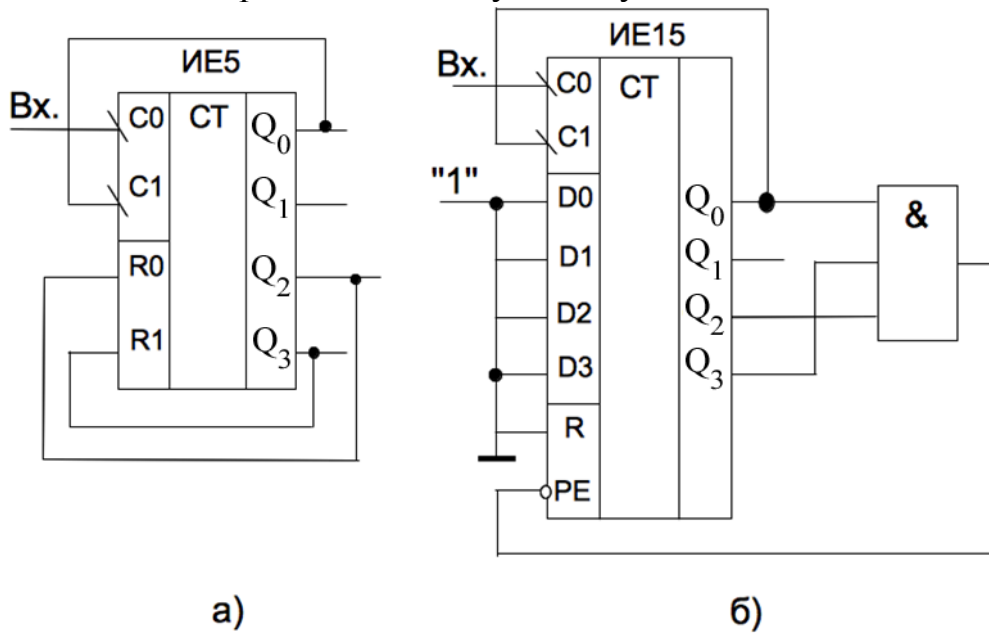


Рис. 9.11

#### **4. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ**

Оборудование и компоненты: универсальная лабораторная установка IDL – 800, интегральные схемы 1533ТВ6 (74ALS107) – два JK триггера с инверсным динамическим синхровходом, 1533ТМ2 (74ALS107) – два D триггера с прямым динамическим синхровходом, 1533ЛИ1 (74ALS08) – четыре логических элемента 2И, 1533ИЕ2 (74ALS 90) – двоично-десятичный асинхронный счётчик, 1533ИЕ5 (74ALS93) – асинхронный двоичный счётчик, 155ИЕ4 (7492) – счётчик – делитель на 12, 555ИЕ15 (74LS197) – асинхронный двоичный счётчик, 1533ИЕ19 (74ALS393) – два двоичных асинхронных счётчика.

##### **3.1. Исследование асинхронного двоичного счётчика на триггерах с инверсным динамическим синхровходом .**

1. Построить асинхронный четырёхразрядный счётчик на триггерах ТВ6.
2. Исследовать работу счётчика. Результаты наблюдений представить в табличном виде и в виде временных диаграмм.

##### **3.2. Исследование асинхронного двоичного счётчика на триггерах с прямым динамическим синхровходом.**

1. Построить асинхронный четырёхразрядный счётчик на триггерах ТМ2.
2. Исследовать работу счётчика. Результаты наблюдений представить в табличном виде и в виде временных диаграмм.

##### **3.3. Исследование двоично-десятичного счётчика .**

1. Построить асинхронный двоично-десятичный счётчик, рис. 9 .б.
2. Исследовать работу счётчика. Результаты представить в табличном виде и в виде временных диаграмм.

##### **3.4. Исследование асинхронного счётчика ИЕ2 .**

1. Построить двоично – десятичный счётчик на основе ИЕ2.
2. Исследовать работу счётчика. Результаты представить в табличном виде и в виде временных диаграмм
3. Построить счётчик – делитель на 10 на основе ИЕ 2.
4. Исследовать работу счётчика-делителя. Результаты представить в табличном виде и в виде временных диаграмм.

##### **3.5. Исследование асинхронного счётчика ИЕ 4.**

1. Построить счётчик с  $\text{mod} = 12$  на основе ИЕ 4.

2. Исследовать работу счётчика. Результаты представить в табличном виде и в виде временных диаграмм.

### **3.6. Исследование асинхронного счётчика ИЕ 5.**

1. Построить счётчик с  $\text{mod} = 12$  на основе ИЕ 5, рис. 9.11а.
2. Исследовать работу счетчика. Результаты представить в табличном виде и в виде временных диаграмм.

### **3.7. Исследование асинхронного счётчика ИЕ 15.**

1. Построить десятичный счётчик на основе ИЕ 15, рис. 9.11б.
2. Исследовать работу счётчика. Результаты представить в виде таблицы и в виде временных диаграмм.

**3.8. Исследовать работу асинхронного счётчика ИЕ 14. Задание получить у преподавателя.**

**3.9. Исследовать работу асинхронного счётчика ИЕ 19. Задание получить у преподавателя.**

## **4. СОДЕРЖАНИЕ ОТЧЕТА**

- 4.1. Цель работы.
- 4.2. Схемы, исследуемые в работе.
- 4.3. Таблицы наблюдений и временные диаграммы работы исследуемых счётчиков.
- 4.4. Выводы по результатам наблюдений.

## **5. КОНТРОЛЬНЫЕ ВОПРОСЫ**

- 5.1. Дать определение цифрового счётчика.
- 5.2. По каким признакам классифицируются счётчики?
- 5.3. Нарисовать схемы суммирующего и вычитающего счётчиков на триггерах с прямым и с инверсным динамическим входом.
- 5.4. Чем различаются между собой асинхронные и синхронные счётчики?
- 5.5. Каковы основные достоинства и недостатки асинхронных счётчиков?
- 5.6. Охарактеризовать основные ИС асинхронных счётчиков.
- 5.7. Объяснить использование входов «Загрузка», «Сброс» и «Установка».

## Лабораторная работа №10

### ИССЛЕДОВАНИЕ СИНХРОННЫХ СЧЕТЧИКОВ

#### 2. ЦЕЛЬ ЛАБОРАТОРНОЙ РАБОТЫ

– изучение основ теории методов синтеза и исследование работы синхронных счётчиков.

#### 3. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Максимальное время переключения асинхронного счётчика, когда выход счётчика последовательно изменяется 11...1 на 00...0, и это ограничивает его быстродействие. Быстродействие можно повысить, если все триггеры счётчика тактировать одновременно. В этом случае счётчик называется синхронным. Основу синхронного счётчика составляют, как правило, синхронные триггеры с динамическим управлением. Все синхровходы триггеров объединены образуя счётный вход счётчика. Количество триггеров зависит от модуля счёта ( $\text{mod} < 2^N$ ). В задачу синтеза синхронного счётчика входит определение связей управляющих входов триггеров и их выходов, чтобы триггеры переключались в соответствии с заданной последовательностью состояний счётчика. Синхронные счётчики могут быть синтезированы с использованием системных методов. Однако прежде, чем рассматривать такой метод, рассмотрим интуитивный метод. Для этого воспользуемся табл. 10.1 в которой дана счётная последовательность и соответствующие состояния счётчика для  $\text{mod} = 8$ .

Таблица 10.1

Счёт	Состояния счётчика		
	Q <sub>2</sub>	Q <sub>1</sub>	Q <sub>0</sub>
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Из табл. 10.1 видно, что выход Q<sub>0</sub> младшего триггера счётчика изменяется каждым счётным импульсом. Это может быть достигнуто, если использовать T-триггер с T=1. Выход Q<sub>1</sub> изменяет своё состояние всякий раз, когда Q<sub>0</sub> изменяется с 1 на 0. Поэтому, если соединить выход Q<sub>0</sub> с входом T<sub>1</sub>, то Q<sub>1</sub> будет изменять свое состояние, когда Q<sub>0</sub>=1 Q<sub>0</sub> =1 (T<sub>1</sub>=Q<sub>0</sub>=1) и будет оставаться без изменения, когда Q<sub>0</sub>=T<sub>1</sub>=0. Из табл.10.1 также видно, что Q<sub>2</sub> изменяет своё со-

стояние всякий раз, когда  $Q_1$  и  $Q_0$  оба равны 1. Это может быть реализовано, если вход наиболее значащего триггера  $ТТ_2 = Q_1 \cdot Q_0$ . В результате получим схему синхронного счётчика ( $\text{mod} = 8$ ), показанную на рис.10.1. Аналогичным образом могут быть построены синхронные счётчики с  $\text{mod} = 2N$ .

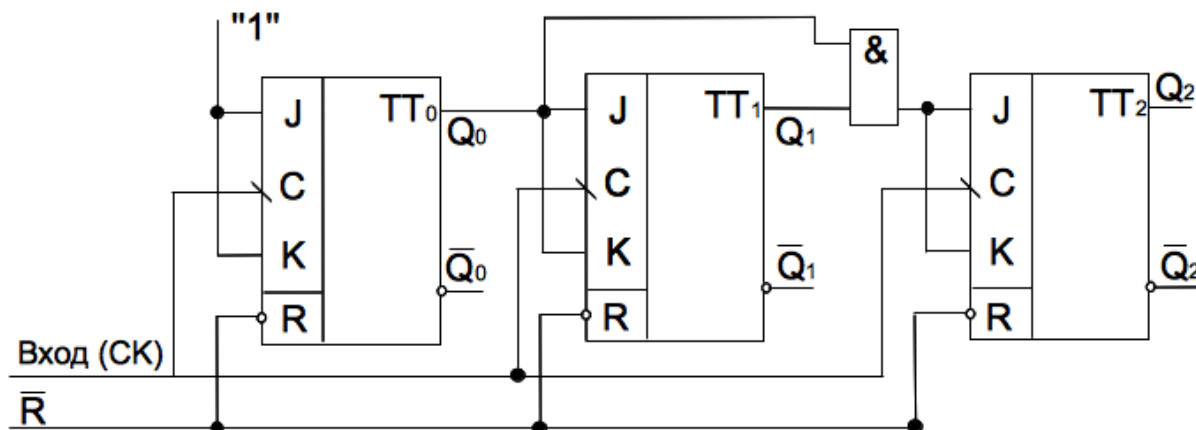


Рис.10.1 Синхронный счётчик с  $\text{mod}=8$

## СИНТЕЗ СИНХРОННЫХ СЧЁТЧИКОВ

Синхронный счётчик для любой заданной последовательности счёта и модуля счёта может быть синтезирован следующим образом:

1. Необходимо определить требуемое число триггеров, исходя из:  
 $N \geq \log_2 m$ ,  
 где  $m$  – модуль счёта.
2. Записать счётную последовательность в табличном виде.
3. Определить состояния входов триггеров, которые должны быть для перехода в требуемые следующие состояния, исходя из настоящего состояния и таблицы переходов триггеров.
4. Приготовить карты Карно для каждого входа триггеров в терминах выходов триггеров, как входных переменных. Используя Карно метод получить минимизированные выражения для каждого входа триггеров.
5. Построить схему счётчика, используя триггеры и логические элементы, в соответствии с минимизированными выражениями.

Пример 1. Построить двоично-десятичный счётчик, который имеет десять состояний, поэтому для его построения необходимо ( $N \geq \log_2 10$ )  $N=4$  триггера. Построим таблицу состояний и таблицу истинности для входов триггеров, табл.10.2.

Таблица 10.2.

Состояния счётчика	Входы триггеров							
	Q <sub>3</sub> Q <sub>2</sub> Q <sub>1</sub> Q <sub>0</sub>	J <sub>3</sub> K <sub>3</sub>	J <sub>2</sub> K <sub>2</sub>	J <sub>1</sub> K <sub>1</sub>	J <sub>0</sub> K <sub>0</sub>			
0 0 0 0	0 x	0 x	0 x	1 x				
0 0 0 1	0 x	0 x	1 x	x 1				
0 0 1 0	0 x	0 x	x 0	1 x				
0 0 1 1	0 x	1 x	x 1	x 1				
0 1 0 0	0 x	x 0	0 x	1 x				
0 1 0 1	0 x	x 0	1 x	x 1				
0 1 1 0	0 x	x 0	x 0	1 x				
0 1 1 1	1 x	x 1	x 1	x 1				
1 0 0 0	x 0	0 x	0 x	1 x				
1 0 0 1	x 1	0 x	0 x	x 1				
0 0 0 0								

Используя Карно метод, получим минимизированные выражения для всех входов J и K, как показано на рис.10.2. По выражениям, полученным на рис.10.2. строим синхронный двоично-десятичный счётчик, рис.10.3.

	$Q_1Q_0$		00	01	11	10
$Q_3Q_2$	00					
	01			1		
	11	x	x	x	x	
	10	x	x	x	x	

$$J_2 = Q_2Q_1Q_0$$

	$Q_1Q_0$		00	01	11	10
$Q_3Q_2$	00	x	x	x	x	
	01	x	x	x	x	
	11	x	x	x	x	
	10		1	x	x	

$$K_3 = Q_0$$

	$Q_1Q_0$		00	01	11	10
$Q_3Q_2$	00			1		
	01	x	x	x	x	
	11	x	x	x	x	
	10			x	x	

$$J_2 = Q_1Q_0$$

	$Q_1Q_0$		00	01	11	10
$Q_3Q_2$	00	x	x	x	x	
	01			1		
	11	x	x	x	x	
	10	x	x	x	x	

$$K_2 = Q_1Q_0$$

	$Q_1Q_0$		00	01	11	10
$Q_3Q_2$	00		1	x	x	
	01		1	x	x	
	11	x	x	x	x	
	10			x	x	

$$J_1 = \bar{Q}_3Q_0$$

	$Q_1Q_0$		00	01	11	10
$Q_3Q_2$	00	x	x	1		
	01	x	x	1		
	11	x	x	x	x	
	10	x	x	x	x	

$$K_1 = Q_0$$

	$Q_1Q_0$		00	01	11	10
$Q_3Q_2$	00	1	x	x	1	
	01	1	x	x	1	
	11	x	x	x	x	
	10	1	x	x	x	

$$J_0 = 1$$

	$Q_1Q_0$		00	01	11	10
$Q_3Q_2$	00	x	1	1	x	
	01	x	1	1	x	
	11	x	x	x	x	
	10	x	1	x	x	

$$K_0 = 1$$

Рис.10.2 Минимизация функций J и K с помощью карт Карно



Эти счётчики имеют два отдельных разрешающих входа ENT и ENP. Подавая на любой из этих входов лог.0 счёт останавливается асинхронно. Выход последовательного переноса RC находится в состоянии лог.0 и становится лог.1 всякий раз, когда счётчик достигает своего максимального значения (двоичное число 9 для двоично-десятичного счётчика и двоичное 15 для двоичного счётчика). Таблица функционирования счётчиков группы 1 дана в табл. 10.3.

Таблица 10.3.

L	ENP	ENT	Cr	СК	Операция
0	x	x	1	↑	Установка (загрузка)
1	0	1	1	x	Остановка счёта
1	x	0	1	x	Остановка счёта, невозможность RC
x	x	x	0	*	Сброс в нуль
1	1	1	1	↑	Прямой счёт

\* - x для ИЕ9, ИЕ10

↑ для ИЕ11, ИЕ18

**Пример2.** Построить делитель на 11, используя ИЕ18. Использовать RC выход и установочные входы.

**Решение.** Для получения делителя на 11, счётчик устанавливается в двоичное состояние 0101 (десятичное 5). Когда счёт достигнет 1111, на выходе RC появляется лог.1. Эта лог.1 используется для загрузки данных с установочных входов в счётчик, рис.10.4.

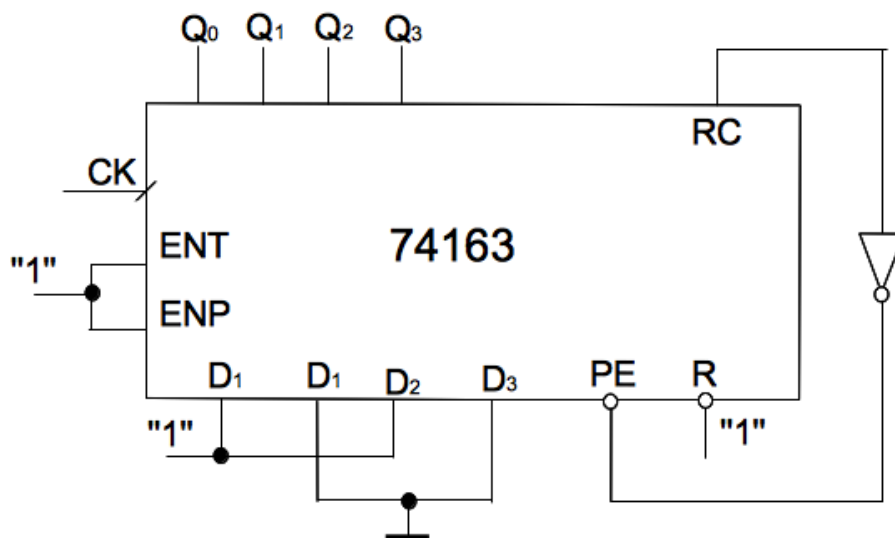


Рис.10.4. Схема делителя на 11

Итак, для получения делителя на  $m$ , на установочных входах должно быть:  $D=16-m$  для двоичного счётчика и  $D=10-m$  для десятичного счётчика.

Входы ENT, ENP и RC могут использоваться для организации каскадного соединения счётчиков, рис.10.5

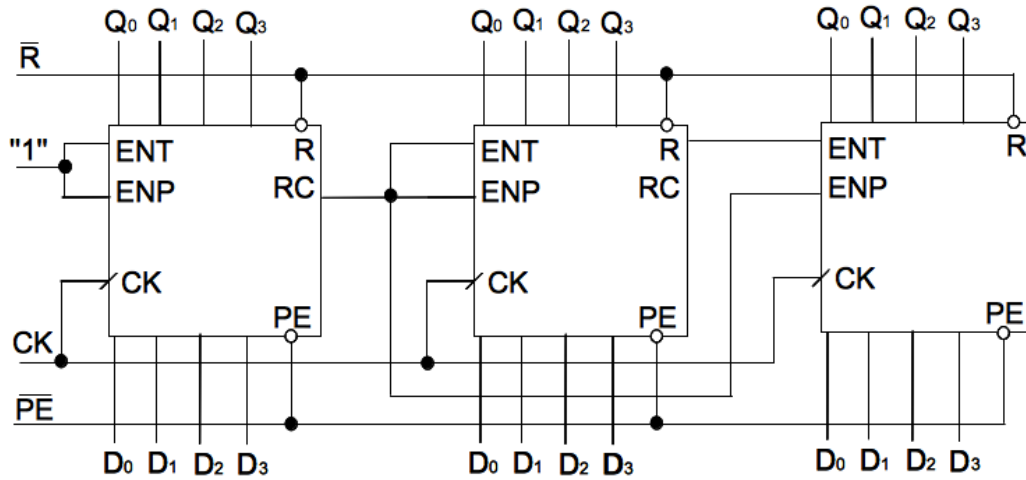


Рис.10.5 Каскадное соединение счётчиков первой группы.

**Синхронные счётчики группы 2.** К этой группе относятся счётчики ИЕ16(74168) и ИЕ17(74169). Это соответственно двоично-десятичный и двоичный реверсивные счётчики с синхронной установкой и без сброса, рис.10.6. Функция ENT и ENP та же, что и у счётчиков первой группы, но в этом случае эти входы активны при низком логическом входе. Выход последовательного переноса в нормальном состоянии равен лог.1 и переходит в лог.0, когда счёт достигает максимального значения при прямом счёте или, когда счёт достигает минимального значения при обратном счёте.

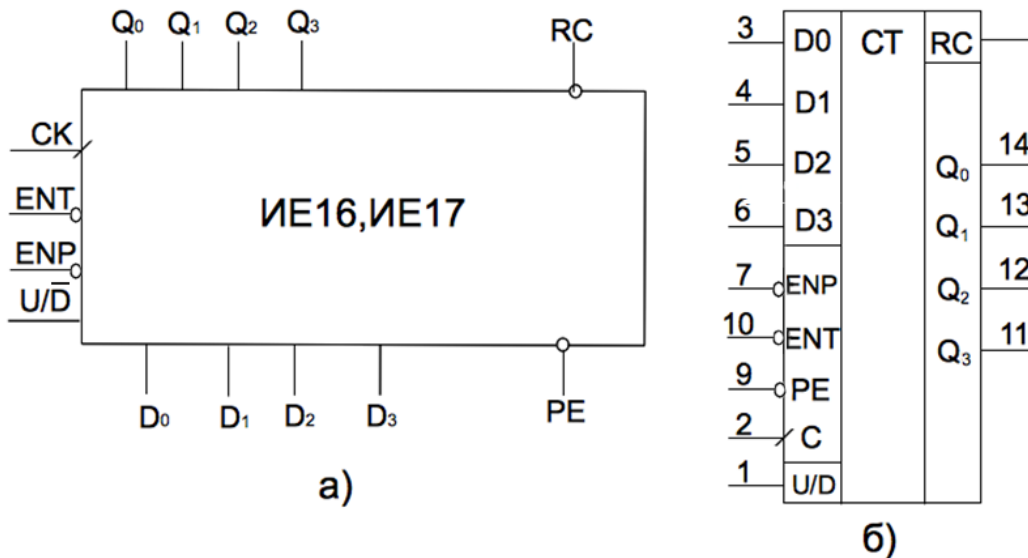


Рис.10.6 Блок-диаграмма, условное обозначение и цоклёвка ИЕ16, ИЕ17

Таблица функционирования счётчиков ИЕ16 и ИЕ17 дана в табл.10.4

Таблица.10.4.

PE	ENP	ENT	U/D	СК	Операция
0	x	x	x	↑	Установка
1	1	0	x	x	Остановка счёта
1	x	1	x	x	Остановка счёта, запрет переноса
1	0	0	1	↑	Прямой счёт
1	0	0	0	↑	Обратный счёт

Сигнал на входе  $U/\overline{D}$  определяет направление счёта:  $U/D=1$  для прямого счёта и  $U/\overline{D}=0$  для обратного. В счётчиках этой группы отсутствует вход сброса, R. Поэтому, если необходимо остановить счёт до достижения максимального значения, схема И-НЕ должна быть использована для детектирования состояния соответствующего требуемому числу и выход схемы И-НЕ соединяется с входом установки,  $\overline{PE}$ . А входы установки дают необходимое начальное состояние счётчика.

**Пример3.** Построить счётчик с начальным состоянием 0011 и конечным 1100, используя ИЕ17.

**Решение**

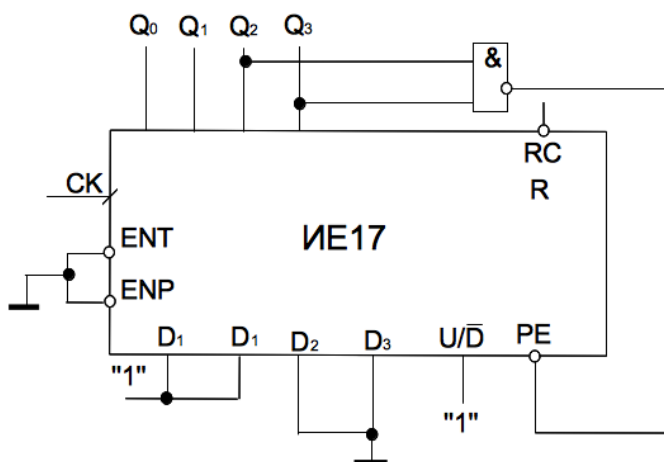


Рис.10.7 Двоично-десятичный счётчик, осуществляющий счёт в Excess-3 коде

Когда счётчик, рис.10.7 достигнет состояния 1100 на выходе элемента И-НЕ и на входе PE появится лог.0 и следующим синхроимпульсом (положительным перепадом) счётчик установится в состояние 0011. Необходимо учитывать, что установка счётчика осуществляется синхронно.

Каскадное соединение счётчиков группы 2 осуществляется аналогично каскадному соединению счётчиков первой группы.

**Синхронные счётчики группы 3.** К этой группе относятся счётчики ИЕ12(74190) и ИЕ13(74191). Это соответственно двоично-десятичный и двоичный реверсивные счётчики с асинхронной установкой и без сброса. Эти счётчики имеют только один низкий активный разрешающий вход, EN. Выход MAX/MIN (рис.10.8) используется для определения максимального или минимального состояния счётчика. На этом выходе нормально лог.0, а лог.1 появляется, когда максимальное состояние счётчика 1001 для ИЕ12 и 1111 для ИЕ13 при прямом счёте или, когда минимальное состояние 0000 при обратном счёте.

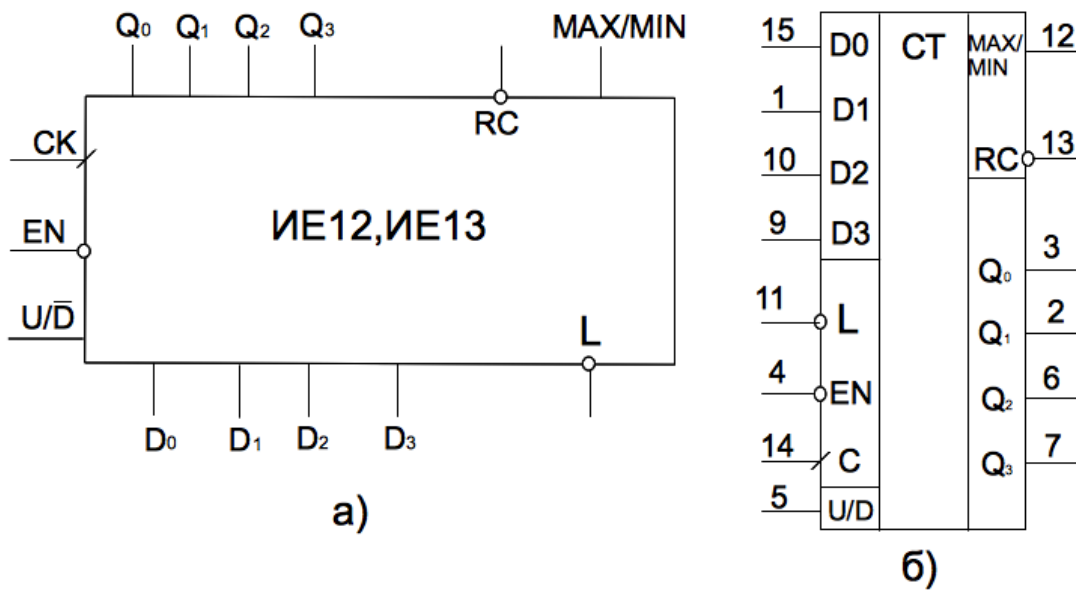


Рис.10.8 Блок-диаграммы, условное обозначение и цоклёвка ИЕ12,ИЕ13

Выход RC нормально в состоянии лог.1 и переходит в лог.0, когда счётчик достигает MAX/MIN точки и импульс синхронизации становится равным 0. В табл.10.5 представлена таблица функционирования счётчиков ИЕ12, ИЕ13.

Таблица 10.5

L	ENAB	U/D	СК	Операция
x	1	x	x	Остановка счёта
0	0	x	x	Установка счётчика
1	0	0	↑	Прямой счёт
1	0	1	↑	Обратный счёт

Синхронные счётчики третьей группы могут каскадироваться различными путями. Максимальное быстродействие будет, если использовать параллельный перенос, рис.10.9.

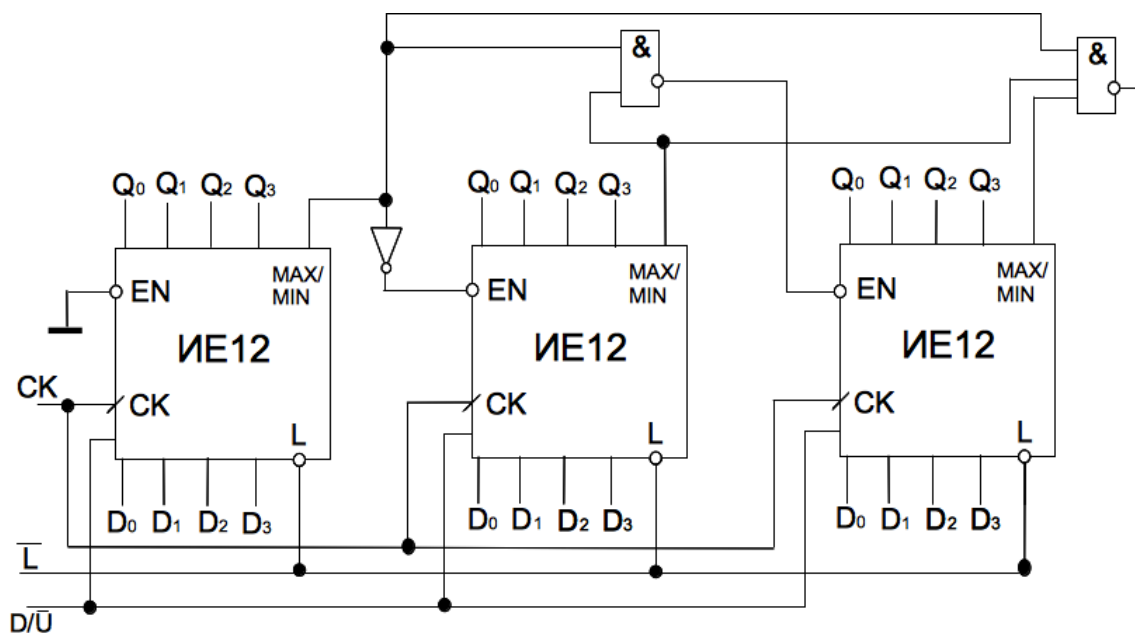


Рис.10.9. Каскадные соединения счётчиков третьей группы

**Синхронные счётчики группы 4.** К этой группе относятся счётчики ИЕ6(74192) и ИЕ7(74193). Это соответственно двоично-десятичный и двоичный реверсивные счётчики с асинхронной загрузкой и сбросом. У этих счётчиков, рис.10.10 импульсы для прямого счёта подаются на вход С-U при этом вход С-D соединяется с лог.1 и импульсы для обратного счёта подаются на вход С-D, при этом вход С-U соединяется с лог.1.

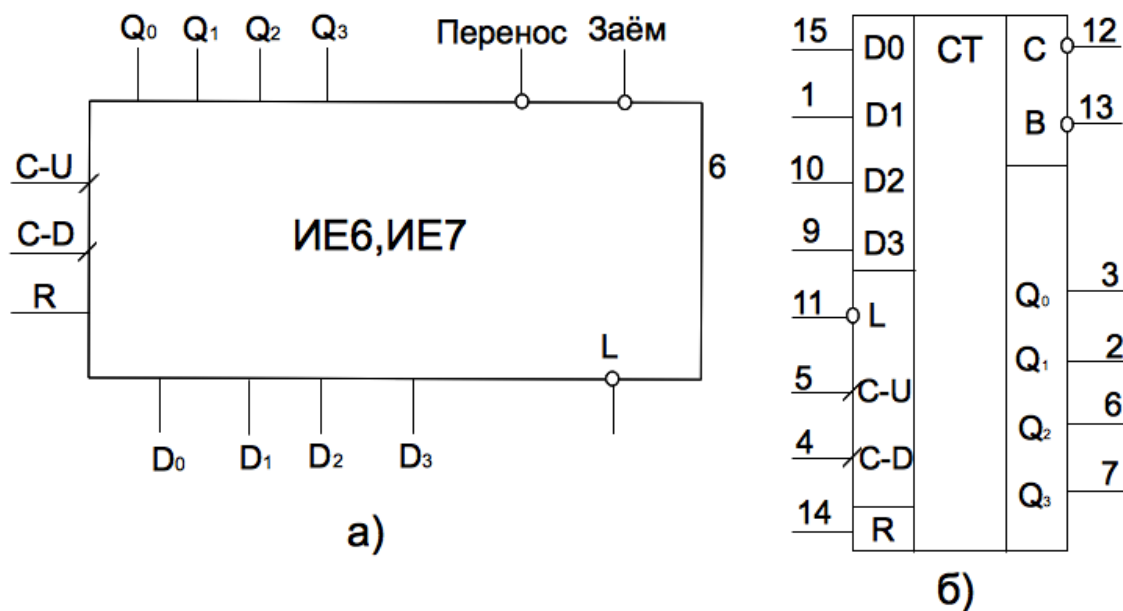


Рис.10.10. Блок-диаграмма, условное обозначение и цоколёвка ИЕ6 и ИЕ7

Выход С (перенос) и В (заём) обычно находятся в состоянии лог.1. Выход переноса переходит в лог.0, когда счётчик достигает максимального значения

при прямом счёте и вход C-U находится в лог.0. Выход заём остаётся в лог .1, когда счётчик работает от прямого C-U входа.

Функция заём при обратном счёте аналогична переносу при прямом счёте.

Таблица функционирования счётчиков ИЕ6 и ИЕ7 дана в табл.10.6.

Таблица 10.6

L	R	C-U	C-D	Операция
x	1	x	x	Сброс
1	0	↑	1	Прямой счёт
1	0	↑	↑	Обратный счёт
0	0	x	x	Установка
1	0	1	1	Остановка счёта

При каскадировании этих счётчиков перенос и заём каждой ступени должны быть соединены с входами C-U и C-D последующей ступени соответственно. Для управления импульсами на прямой и обратный счётный вход может быть использована схема, показанная на рис.10.11.

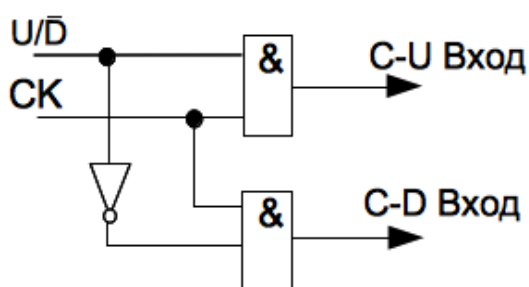


Рис.10.11. Схема для управления счётными импульсами при прямом и обратном счёте.

### 3. Порядок выполнения лабораторной работы

**Оборудование и компоненты:** универсальная лабораторная установка IDL-800, интегральные схемы: -1533ЛИ1(74ALS08), 1533ТВ6(74ALS107), 1533И9(74ALS160), 1533ИЕ10(74ALS161), 1533ИЕ11(74ALS163), 1533ИЕ18(74ALS163), 1533ИЕ16(74ALS168), 1533ИЕ17(74ALS169), 1533ИЕ12(74ALS190), 1533ИЕ13(74ALS191), 1533ИЕ6(74ALS192), 1533ИЕ7(74ALS193).

### **3.1 Исследование двоично-десятичного счётчика.**

1. Используя ИС 1533ТВ6 и 1533ЛИ1 собрать схему двоично-десятичного счётчика, рис.10.3.
2. Подавая на счётный вход одиночные импульсы, исследовать работу счётчика.
3. Результаты исследований предоставить в табличной форме и в виде временных диаграмм.

### **3.2 Синтез и исследование синхронного счётчика, работающего в коде Грея.**

1. Синтезировать счётчик, работающий в коде Грея, для  $\text{mod}=8$ .
2. Используя ИС 1533ТВ6 и 1533ЛИ1 собрать схему синхронного счётчика кода Грея.
3. Подавая на счётный вход одиночные импульсы, исследовать работу счётчика.
4. Результаты исследований представить в табличной форме и в виде временных диаграмм.

### **3.3 Исследование модульных счётчиков первой группы.**

1. Построить делитель на 7 на ИС 1533ИЕ9. Использовать РС выход и установочные входы.
2. Исследовать работу делителя на 7.
3. Результаты представить в табличном виде и в виде временных диаграмм.
4. Построить и исследовать работу двухдекадного счётчика в режиме прямого счёта и в режим обратного счёта.

### **3.4 Исследование модульных счётчиков второй группы.**

1. Построить двоично-десятичный счётчик, осуществляющий счёт в Excess-3 коде, рис.10.7.
2. Исследовать работу счётчика.
3. Результаты представить в виде таблиц и в виде временных диаграмм.

### **3.5 Исследование модульных счётчиков третьей группы.**

1. Исследовать работу ИС 1533ИЕ12(13) в режиме прямого и в режиме обратного счёта.
2. Результаты представить в табличном виде и в виде временных диаграмм. Временные диаграммы должны содержать состояния счётчика на всех выходах, включая РС и MAX/MIN.

### **3.6 Исследование модульных счётчиков четвёртой группы.**

1. Используя ИС 1533ИЕ7 построить двухдекадный счётчик.
2. Исследовать работу счётчика в режиме прямого и обратного счёта.
3. Результаты исследований представить в виде таблиц и виде временных диаграмм.

## **4. СОДЕРЖАНИЕ ОТЧЕТА**

- 4.1 Цель работы.
- 4.2 Схемы, исследуемые в работе
- 4.3 Результаты исследований: - таблицы, временные диаграммы.
- 4.4 Выводы

## **5. КОНТРОЛЬНЫЕ ВОПРОСЫ**

- 5.1 Дать определение электронного счётчика.
- 5.2 Чем отличается синхронный счётчик от асинхронного?
- 5.3 Каковы достоинства и недостатки синхронных счётчиков?
- 5.4 В чём заключается синтез синхронного счётчика?
- 5.5 Дать характеристику и особенности модульных счётчиков.

## ЛИТЕРАТУРА

1. Нефедов А.В. Интегральные схемы и их зарубежные аналоги: Справочник Т.10. – М.: Радио Софт, 2001.-544 с.
2. Богданович М.И. Цифровые интегральные микросхемы: Справочник/М.И. Богданович и др.- Минск: Полымя, 1996.-523 с.
3. Браммер Ю.А. Цифровые устройства. Учебное пособие для вузов/Ю.А. Браммер, И.Н. Пашук. - М.: Высш.шк., 2004. – 229 с.
4. Угрюмов Е.П. Цифровая схемотехника: Учебное пособие для вузов/Е.П. Угрюмов. – СПб: БХВ-Петербург, 2004. – 528 с.
5. Jain R.P. Modern Digital Electronics.-Tata McGran-Hill-New Delhi, 1997-500 p.
6. Ronald J.Tocci. Digital Systems, Principles and Applications.-Prentice-Hall of India. –New Delhi, 1998.-627 p.
7. Victor P. Nelson, H. Troy Naqle, Bill D.Carroll, J.David Irwin. Digital Logic Circuit Analysis and Design.-Prentice Hall, Inc.-New Jersey, 1995.-842 p.
8. M.Morris Mano. Digital Logic and Computer Design.-Prentice-Hall of India. - New Delhi, 1998.-612 p.
9. Charles H.Roth, Jr. Fundamentals of Logic Design.-Jaico Publishing House.- Delhi, 1999.-770 p.

Учебное издание

Будько Анатолий Антонович

## ЦИФРОВЫЕ УСТРОЙСТВА

Учебно-методическое пособие для выполнения  
лабораторных и практических занятий

по курсам  
«Основы цифровой и микропроцессорной техники» и «Цифровые и микропро-  
цессорные устройства»

часть 1  
для студентов радиотехнических специальностей

Автор:

Будько Анатолий Антонович

Редактор  
Корректор  
Компьютерная верстка

---

Подписано в печать	Формат 60x84 1/16.	Бумага офсетная.
Гарнитура «Таймс».	Печать ризографическая.	Усл.печ.л.
Уч.изд л. 8,0.	Тираж 300 экз.	Заказ

---

Издатель и полиграфическое исполнение:  
Учреждение образования  
«Белорусский государственный университет информатики и радиоэлектрони-  
ки»

ЛИ № 02330/0494371 от 16.03.2009. ЛП № 02330/0494175 от 30.04.2009  
220013, Минск, П.Бровки, 6