

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

**А. А. Будько**

**ЦИФРОВЫЕ УСТРОЙСТВА.  
ТЕХНОЛОГИЯ IDL**

*Рекомендовано УМО вузов Республики Беларусь по образованию  
в области информатики и радиоэлектроники в качестве  
учебно-методического пособия для студентов учреждений, обеспечивающих  
получение высшего образования по специальностям  
«Радиотехника» (по направлениям), «Радиоэлектронные системы»,  
«Радиоэлектронная защита информации»*

Минск БГУИР 2011

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

Р е ц е н з е н т ы:

кафедра связи учреждения образования  
«Военная академия Республики Беларусь»  
(начальник кафедры, полковник С. А. Манько);

заведующий кафедрой радиосвязи и радиовещания  
учреждения образования «Высший государственный колледж связи»,  
кандидат технических наук, доцент А. И. Корзун

**Будько, А. А.**

Б90 Цифровые устройства. Технология IDL : учеб.-метод. пособие /  
А. А. Будько. – Минск : БГУИР, 2011. – 136 с. : ил.  
ISBN 978-985-488-685-5.

Приводится описание 10 лабораторных работ по физическому макетированию основных цифровых устройств на лабораторной установке IDL 800 DIGITAL LAB. Даны теоретические сведения, рекомендации и задания по синтезу, физическому макетированию и исследованию типовых и оригинальных цифровых устройств.

Может быть использовано при проведении практических занятий.

УДК 004.3'144(076)  
ББК 32.973.26-04я7

ISBN 978-985-488-685-5

© Будько А. А., 2011  
© УО «Белорусский государственный  
университет информатики  
и радиоэлектроники», 2011

## СОДЕРЖАНИЕ

Лабораторная работа №1. Исследование логических элементов.....	4
Лабораторная работа №2. Синтез комбинационных устройств в заданном базисе логических элементов.....	24
Лабораторная работа №3. Исследование сумматоров, вычитателей и суммирования/вычитания в дополнительном коде.....	34
Лабораторная работа №4. Исследование преобразователей кодов.....	44
Лабораторная работа №5. Синтез комбинационных схем с использованием мультиплексоров и демультимплексоров.....	54
Лабораторная работа №6. Триггеры.....	68
Лабораторная работа №7. Регистры и их применение.....	92
Лабораторная работа №8. Генераторы последовательностей.....	101
Лабораторная работа №9. Исследование асинхронных счетчиков.....	107
Лабораторная работа №10. Исследование синхронных счетчиков.....	120
Литература.....	133
Приложение.....	134

## Лабораторная работа №1

# ИССЛЕДОВАНИЕ ЛОГИЧЕСКИХ ЭЛЕМЕНТОВ

*Цель работы:*

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

### 1. 1. 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;
- 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 длиной  $2^n$ , а число возможных таких наборов равно  $2^{2^n}$ , поэтому справедливо следующее утверждение: 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,  $f_3 = \bar{x}$  (читается «не X»), функция НЕ.

Для  $n = 2$  существует  $2^{2^n} = 2^{2^2} = 16$  функций алгебры логики, которые 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
Опе- рац. 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 ноль (функция ноль). Технически  $f_0$  реализуется генератором нуля (рис. 1.1, а).

Функция  $f_{15}(x_1, x_2) = 1 = \overline{f_0(x_1, x_2)}$  – константа единица (функция единица). Технически реализуется генератором единицы (рис. 1.2, б).

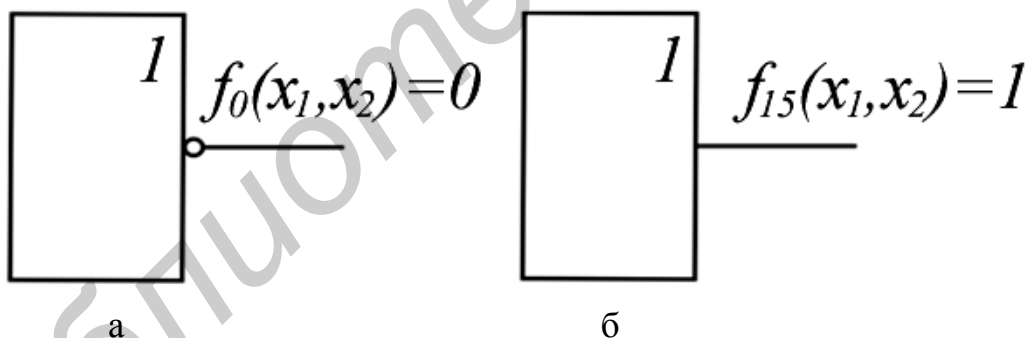


Рис. 1.2. Условное обозначение генераторов логических констант

Будем продолжать рассматривать функции в табл. 1.3 по парам, т. к. 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. 1.3, а.

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, (рис. 1.3, б).

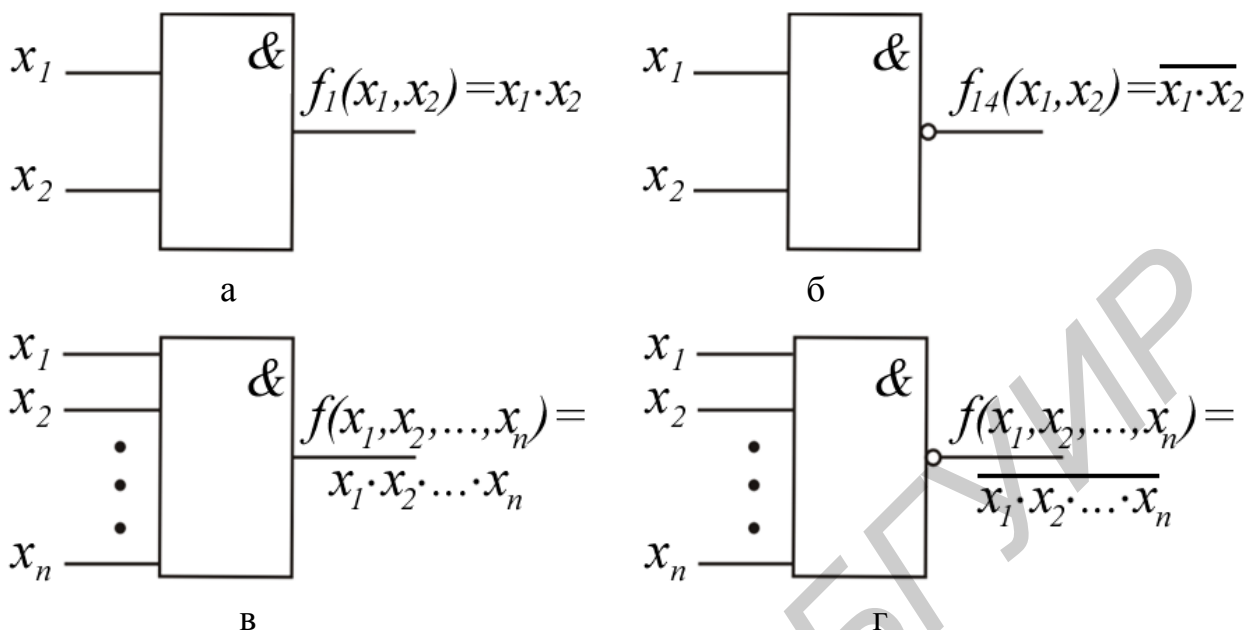


Рис. 1.3. Условное обозначение логических элементов И и И-НЕ

Функции И и И-НЕ могут быть, как и соответствующие им логические элементы, с произвольным числом переменных (входов) (рис. 1.3, в, г).

Функция  $f_2$  – запрет 1-го аргумента :

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

Технически реализуется элементом запрета (рис. 1. 4, а).

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

$$f_{13}(x_1, x_2) = x_1 \rightarrow x_2 = \overline{x_1} + x_2 = \overline{f_2}(x_1, x_2).$$

Технически реализуется импликатором (рис. 1. 4, б).

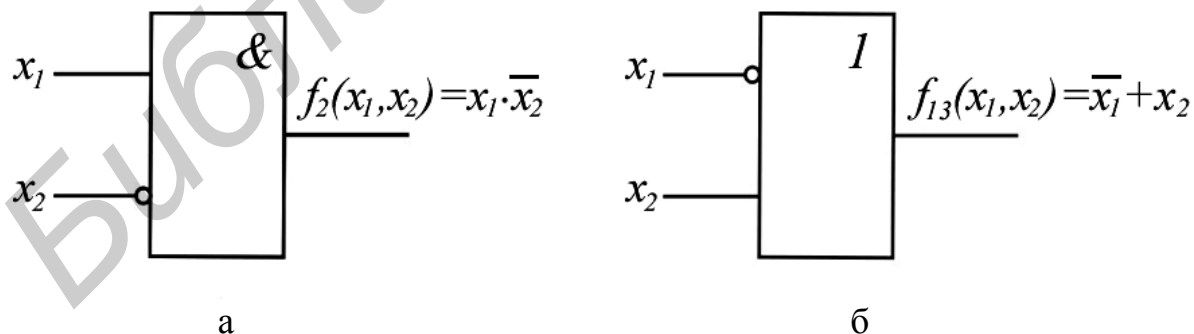


Рис. 1.4. Условное обозначение:  
а – элемента запрета; б – импликатора

Функция  $f_3$  – повторение первого аргумента (функция ДА):

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

Технически реализуется повторителем (рис. 1. 5, а).

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. 5, б).

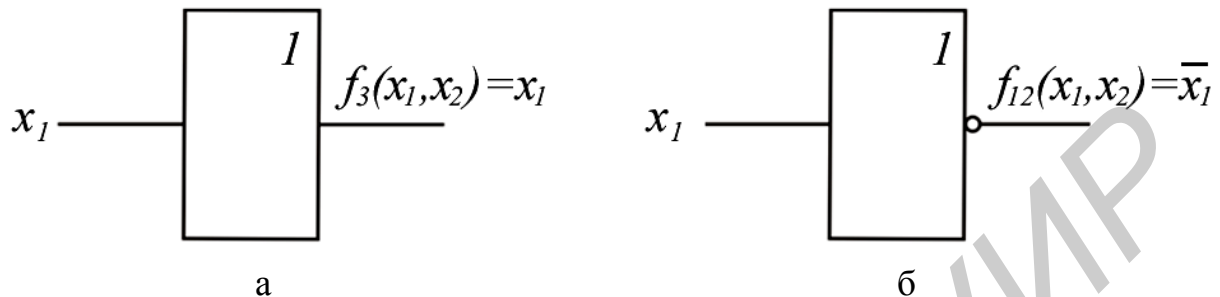


Рис. 1.5. Условные обозначения повторителя и инвертора

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. 6, а).

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

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

Технически реализуется элементом равнозначность, «ИСКЛЮЧАЮЩЕЕ ИЛИ-НЕ» (рис.1.6, б).

Функции неравнозначность и равнозначность могут быть, как и соответствующие им логические элементы, с произвольным числом переменных (входов) (рис. 1.6, в, г). Функция неравнозначность равна 1, если число аргументов, равных 1, нечетно.



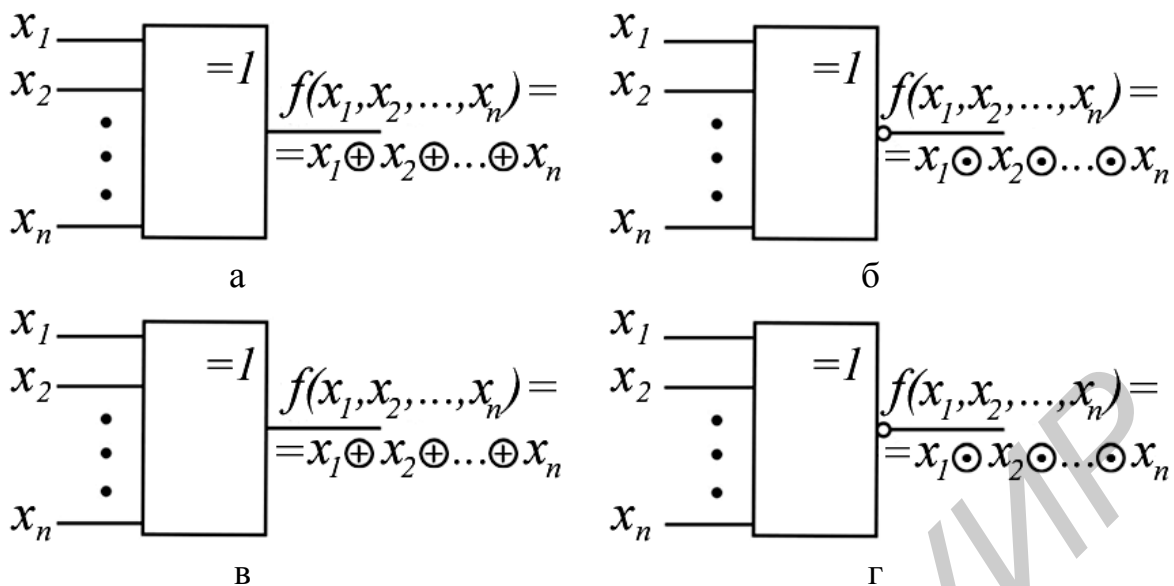


Рис. 1.6. Логические элементы, реализующие функции неравнозначность и равнозначность

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. 1.7, а).

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. 7, б).

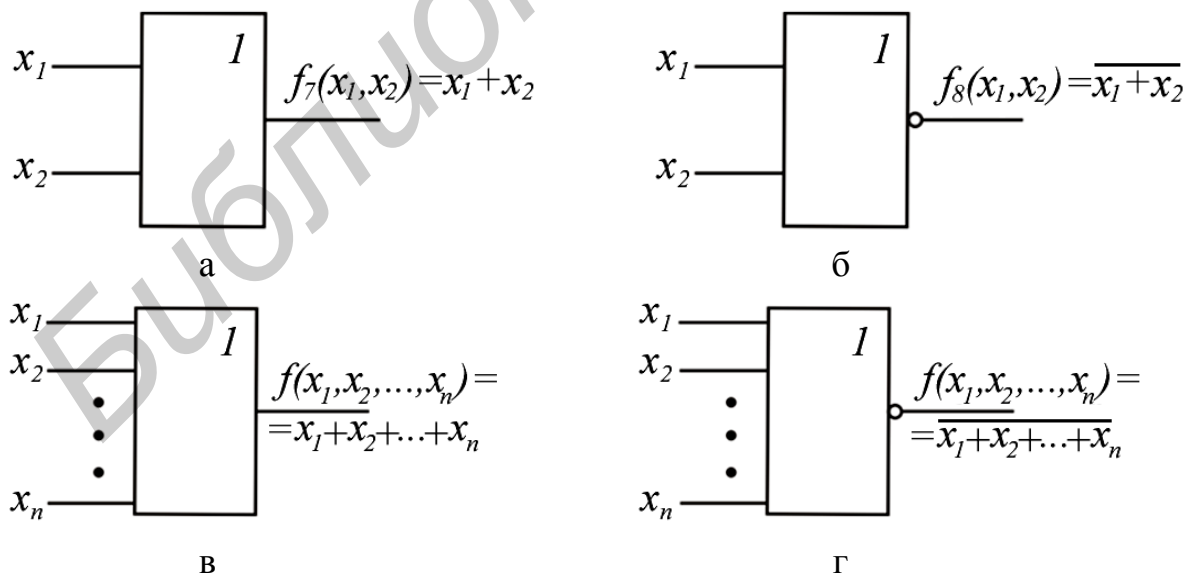


Рис. 1.7. Логические элементы ИЛИ и ИЛИ-НЕ

Функции ИЛИ и ИЛИ-НЕ могут быть, как и соответствующее им логические элементы, с произвольным числом переменных (входов) (рис. 1.7, в, 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 обусловливается 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 неполную.

Примерами полных систем являются:

$$1. F_1(x_1, x_2) = x_1 \cdot x_2, \quad F_2(x_1, x_2) = x_1 + x_2, \quad F_3(x) = \bar{x}.$$

$$2. F_1(x_1, x_2) = x_1 + x_2, \quad F_2(x) = \bar{x}.$$

$$3. F_1(x_1, x_2) = x_1 \cdot x_2, \quad F_2(x) = \bar{x}.$$

$$4. F(x_1, x_2) = \overline{x_1 + x_2}.$$

$$5. F(x_1, x_2) = \overline{x_1 \cdot x_2}.$$

$$6. F_1(x_1, x_2) = x_1 \cdot x_2, \quad F_2(x_1, x_2) = x_1 \oplus x_2, \quad F_3(x) = 1.$$

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	
$x_1 x_2 + x_1 \bar{x}_2 = x_1$	$(x_1 + x_2)(x_1 + \bar{x}_2) = x_1$
$x_1(x_1 + x_2) = x_1$	$x_1 + x_1 x_2 = x_1$
$x_1(\bar{x}_1 + x_2) = x_1 x_2$	$x_1 + \bar{x}_1 x_2 = x_1 + x_2$

## 1.2. 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.

### 1.2.1. 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. 1.8, а.

4. Изменяя состояния входов  $x_1$  и  $x_2$  с помощью переключателей SW, заполнить таблицу истинности логического элемента 2И (табл. 1.5).

### 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.1.9, а.

2. Изменяя состояния входов  $x_1, x_2, x_3$ , исследовать работу схемы, заполнить таблицу истинности логического элемента 3И (табл. 1.6).

Таблица 1.5

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

Таблица 1.6

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		

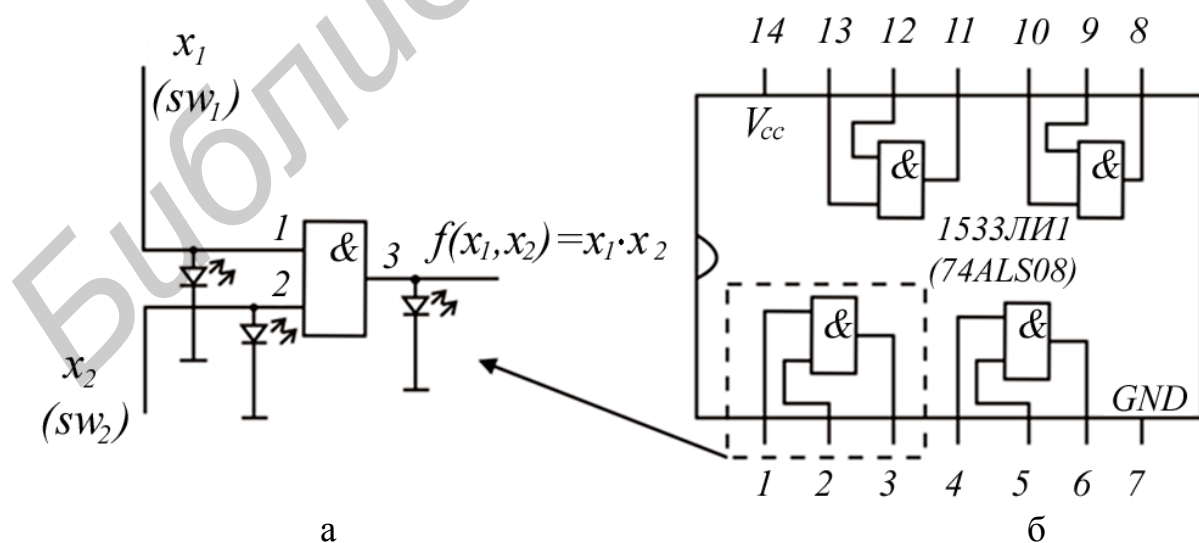


Рис. 1.8. Логический элемент 2И

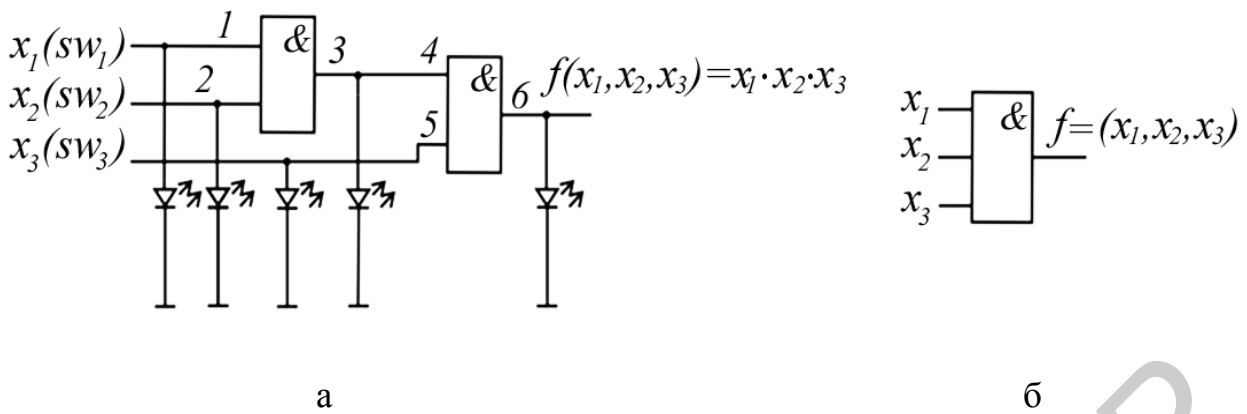


Рис. 1.9. Логический элемент ЗИ

### 1.2.2. Исследование логических элементов ИЛИ

#### *Двухходовый 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. 1.10, а.
4. Изменяя состояние входов  $x_1, x_2$ , исследовать работу двухходового элемента ИЛИ, заполнить таблицу истинности логического элемента (табл. 1.7).

#### *Многоходовой элемент ИЛИ*

1. Используя двухходовые элементы ИЛИ, собрать трехходовый 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 1.11, а.

2. Изменяя состояния входов  $x_1, x_2, x_3$ , исследовать работу схемы, заполнить таблицу истинности логического элемента ЗИЛИ (табл. 1.8).

Таблица 1.7

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

Таблица 1.8

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		

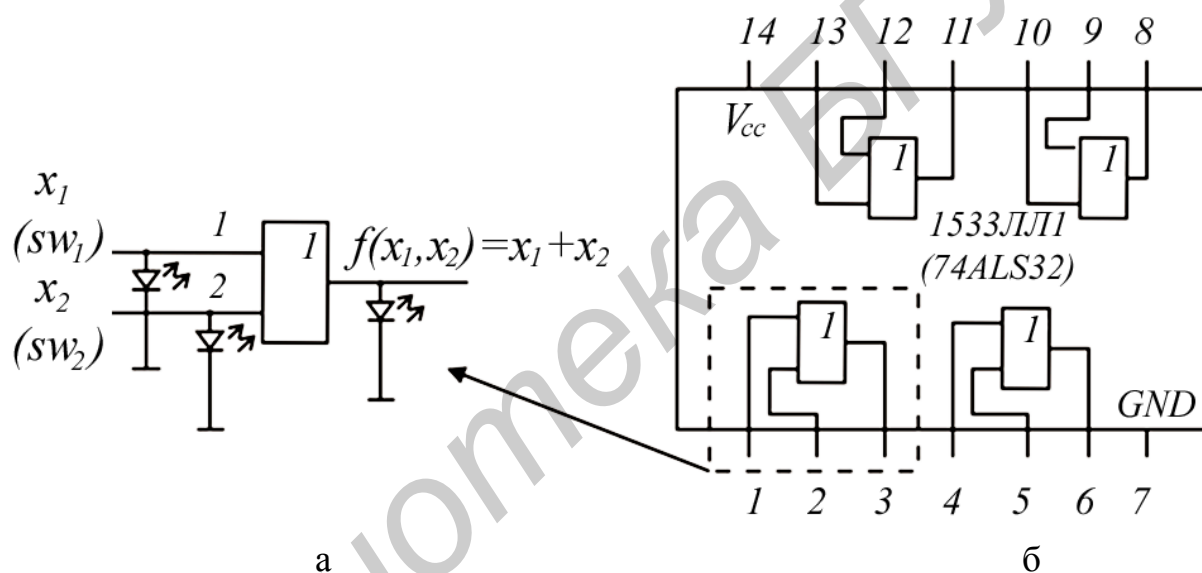


Рис. 1.10. Логический элемент 2И

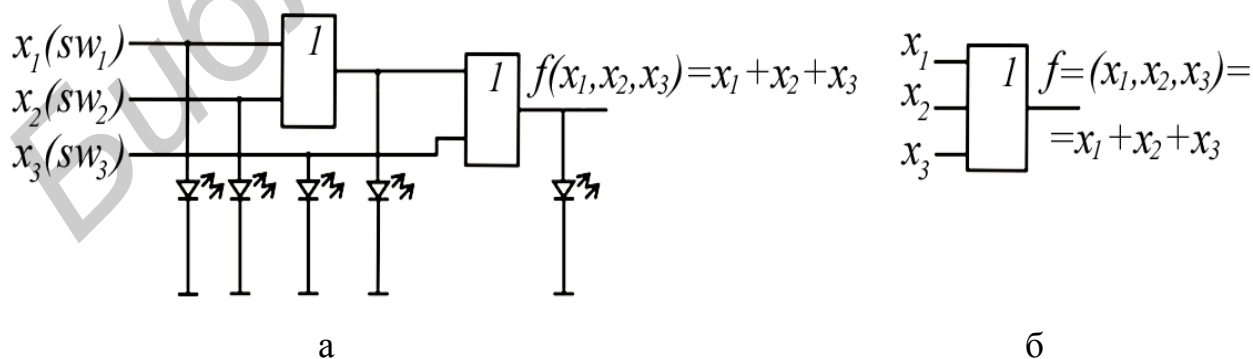


Рис. 1.11. Логический элемент 3ИЛИ

### 1.2.3. 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. 1.12, а.
4. Изменяя состояние входа  $x$ , исследовать работу инвертора, заполнить таблицу истинности инвертора (табл. 1. 9).

Таблица 1.9

Вход	Выход
$x$	$f(x) = \bar{x}$
0	
1	

### 1.2.4. Исследование логического элемента И-НЕ

*Двухходовый 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. 1.13, а.
4. Изменяя состояния входов  $x_1$  и  $x_2$ , исследовать работу двухходового элемента И-НЕ, заполнить таблицу истинности (табл. 1.10).

Таблица 1.10

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

Таблица 1.11

Вход	Выход
$x$	$f(x) = \bar{x}$
0	
1	

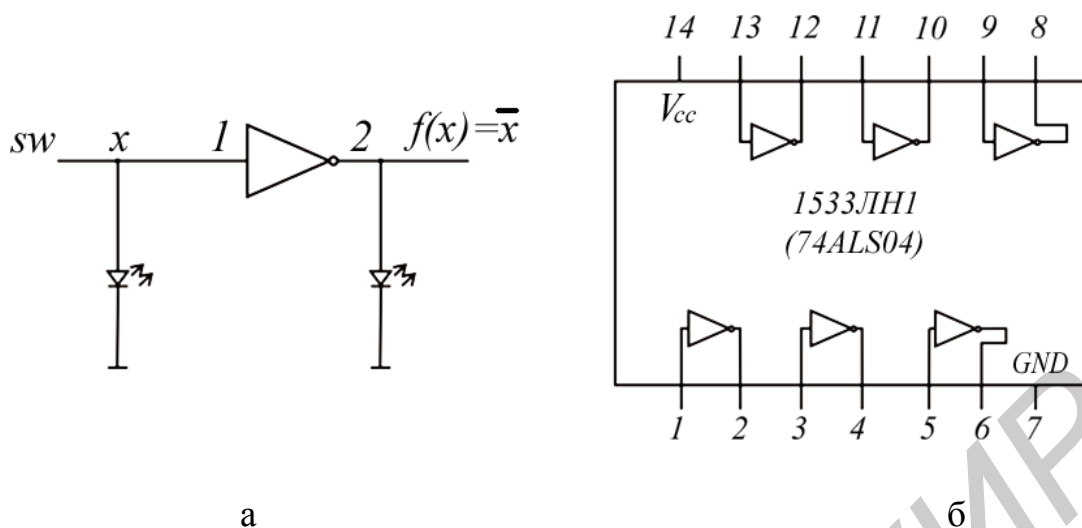


Рис. 1.12. Инвертор

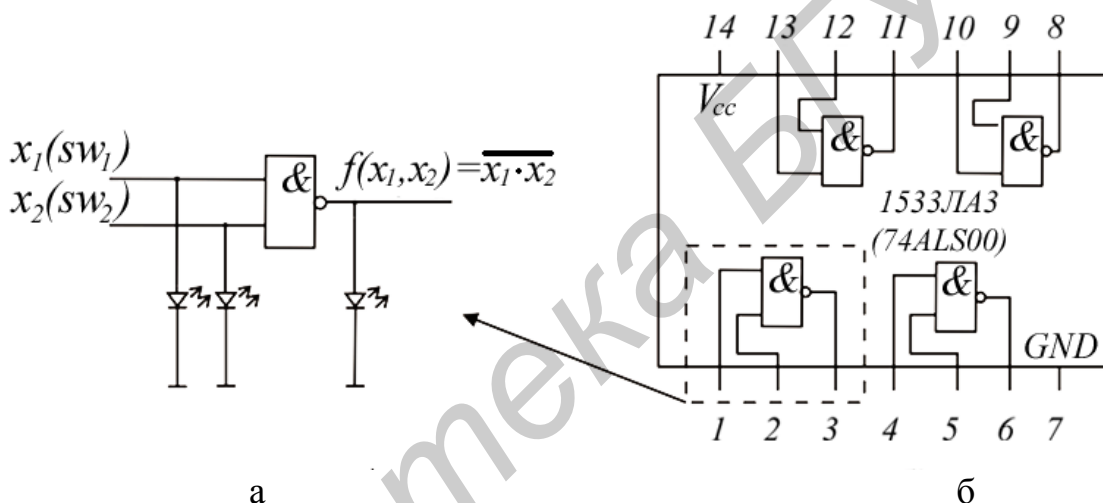


Рис. 1.13. Логический элемент 2И-НЕ

*Двухходовый элемент И-НЕ, используемый как инвертор*

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

1. Собрать схему, как показано на рис. 1.14, а.
2. Изменяя состояние входа X, исследовать работу схемы, заполнить таблицу истинности (табл. 1.11).

*Многоходовый элемент И-НЕ*

1. Используя двухходовые элементы И-НЕ, собрать трехходовый элемент И-НЕ, реализующий функцию  $f(x_1, x_2, x_3) = x_1 \cdot x_2 \cdot x_3$  (рис. 1.14, б).
2. Изменяя состояния входов  $x_1, x_2, x_3$ , исследовать работу схемы, заполнить таблицу истинности (табл. 1.12).



Реализация функции ИЛИ, с помощью логических элементов И-НЕ

1. Используя двухвходовые элементы И-НЕ, собрать схему, реализующую операцию ИЛИ, как показано на рис. 1.14, в.

2. Изменяя состояния входов  $x_1$  и  $x_2$ , исследовать работу схемы, заполнить таблицу истинности (табл. 1.13).

Таблица 1.12

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 1.13

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

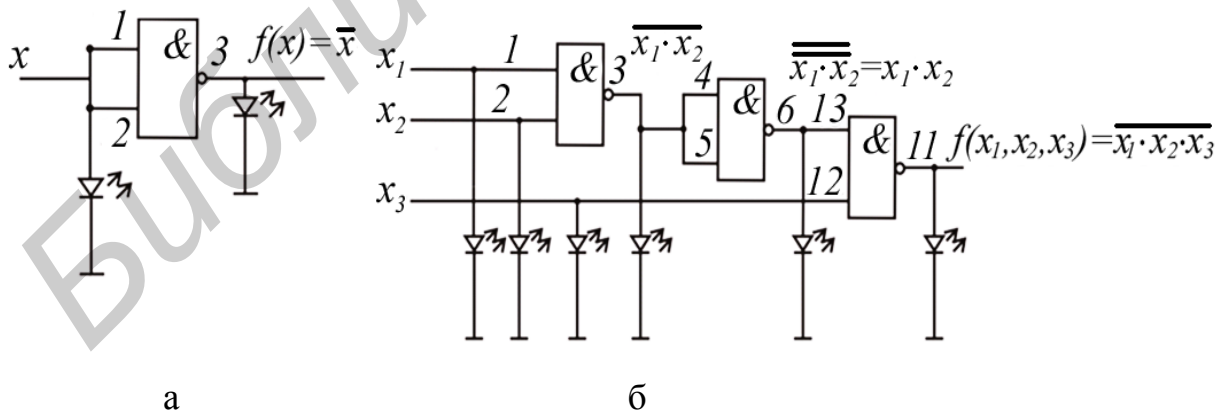


Рис. 1.14. Применение логических элементов И-НЕ (окончание на с. 18)

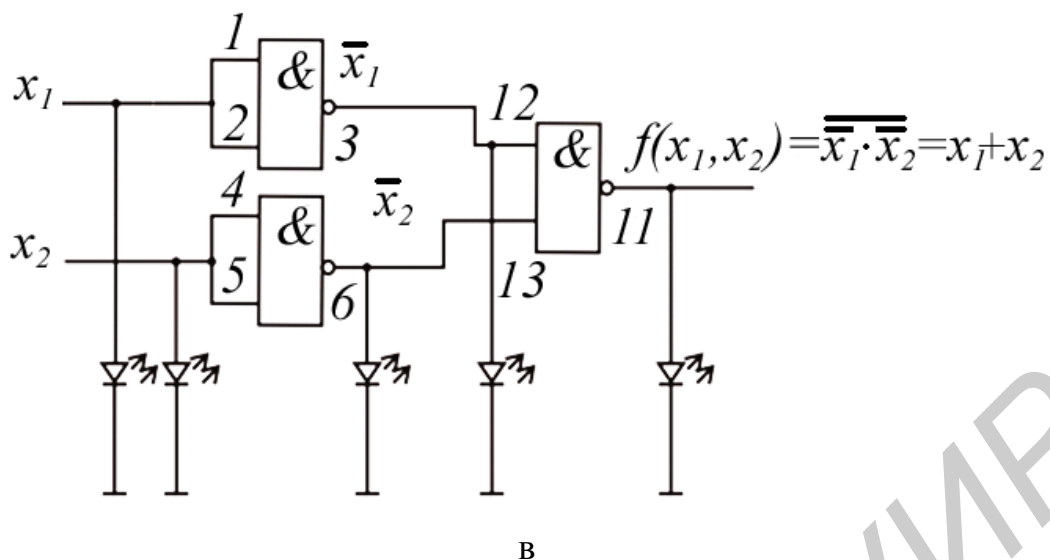


Рис. 1.14. Окончание (начало см. на с. 17)

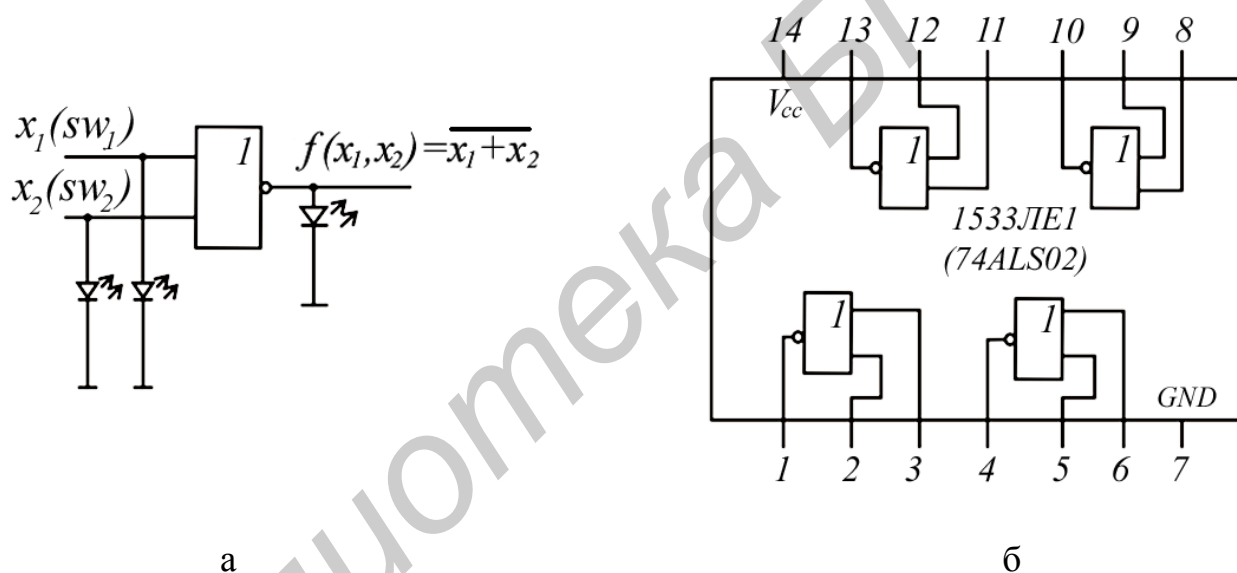


Рис. 1.15. Логический элемент 2ИЛИ-НЕ

### 1.2.5. Исследование логического элемента ИЛИ-НЕ

#### Двухвходовый элемент ИЛИ-НЕ

1. Разместить ИС 1533JE1 (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. 1.15, а.
4. Изменяя состояние входов  $x_1$  и  $x_2$ , исследовать работу элемента 2ИЛИ-НЕ, заполнить таблицу истинности (табл. 1.14).

*Двухходовый 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.1.16, а.
2. Изменяя состояние входа  $x$ , исследовать работу схемы, заполнить таблицу истинности (табл. 1.15).

Таблица 1.14

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 1.15

Вход	Выход
$x$	$f(x) = \bar{x}$
0	
1	

*Многоходовый элемент ИЛИ-НЕ*

1. Используя двухходовые элементы ИЛИ-НЕ, собрать трехходовый элемент ИЛИ-НЕ, реализующий функцию  $f(x_1, x_2, x_3) = \overline{x_1 + x_2 + x_3}$  (рис. 1.16, б).
2. Изменяя состояние входов  $x_1, x_2, x_3$ , исследовать работу схемы, заполнить таблицу истинности (табл. 1.16).

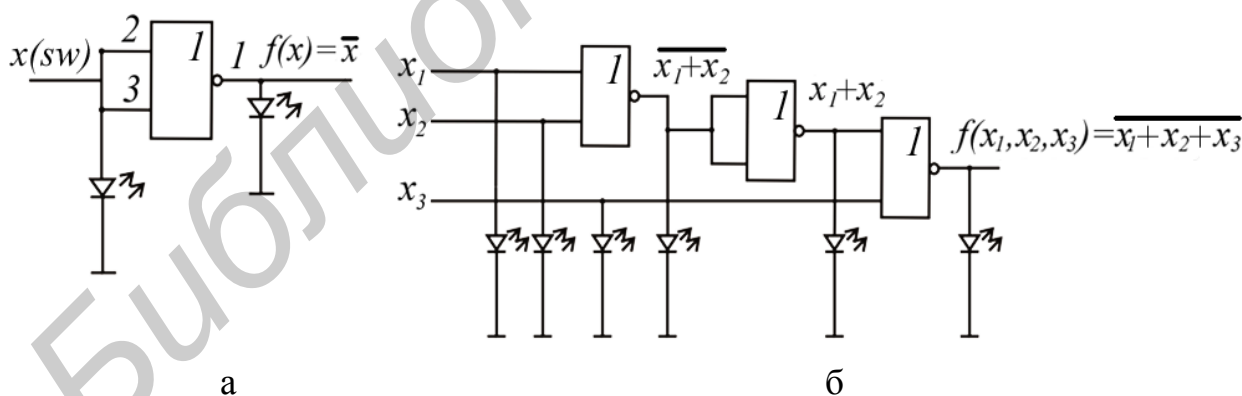
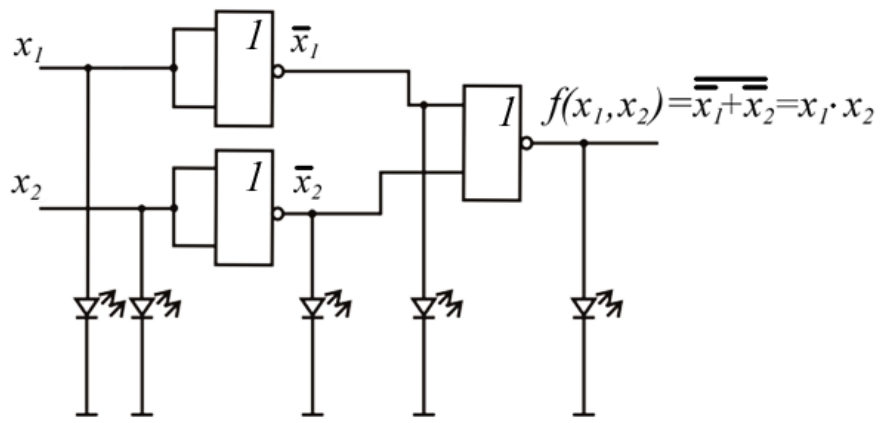


Рис. 1.16. Применение логических элементов ИЛИ-НЕ  
(окончание на с. 20)



в

Рис. 1.16. Окончание (начало см. на с. 19)

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

1. Используя двухвходовые элементы ИЛИ-НЕ, собрать схему, реализующую операцию И, как показано на рис. 1.16, в.
2. Изменяя состояния входов  $x_1, x_2$ , исследовать работу схемы, заполнить таблицу истинности (табл. 1.17).

Таблица 1.16

Входы			Выходы		
$x_1$	$x_2$	$x_3$	$\overline{x_1 + x_2}$	$x_1 + x_2$	$\overline{\overline{x_1 + x_2 + x_3}}$
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 1.17

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

## 1.2.6. 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. 1.17, а.
4. Изменяя состояние входов  $x_1$  и  $x_2$ , исследовать работу двухвходового элемента ИСКЛЮЧАЮЩЕЕ ИЛИ, заполнить таблицу истинности (табл. 1.18).

### Многовходовый элемент ИСКЛЮЧАЮЩЕЕ ИЛИ

1. Используя двухвходовые элементы ИСКЛЮЧАЮЩЕЕ ИЛИ, собрать схему трехвходового элемента ИСКЛЮЧАЮЩЕЕ ИЛИ (рис. 1.18, а).
2. Изменяя состояния входов  $x_1, x_2, x_3$ , исследовать работу схемы, заполнить таблицу истинности (табл. 1.19).

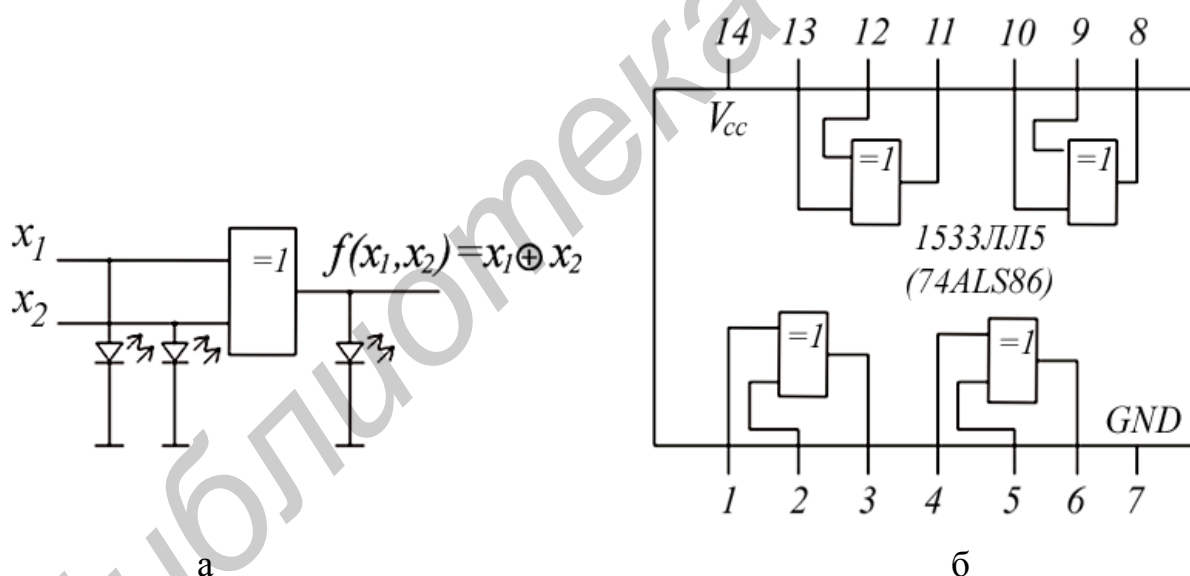
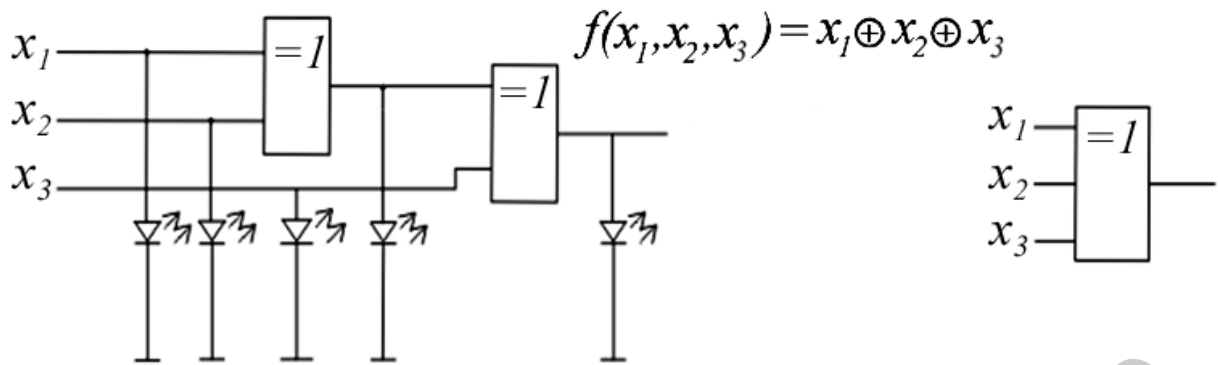
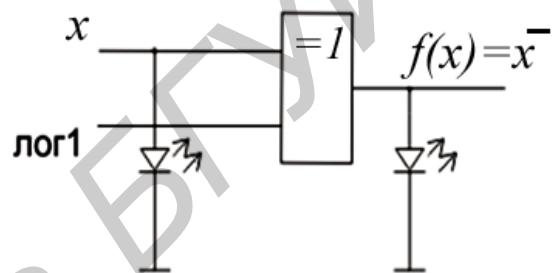
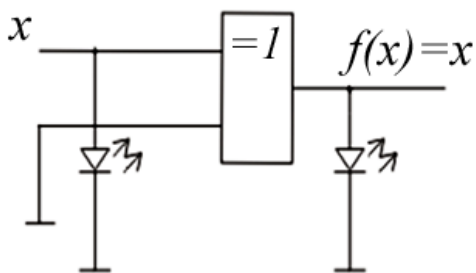


Рис. 1.17. Двухвходовый элемент ИСКЛЮЧАЮЩЕЕ ИЛИ



а

б



в

г

Рис. 1.18. Применение логических элементов ИСКЛЮЧАЮЩЕЕ ИЛИ

Таблица 1.18

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 1.19

Входы			Выходы	
$x_1$	$x_2$	$x_3$	$x_1 \oplus x_2$	$x_1 \oplus x_2 \oplus x_3$
	1		2	3
0	0	0		
0	0	1		
0	1	0		

Окончание табл. 1.19

1			2	3
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. 1.18, в.
2. Изменяя состояние входа  $x$ , исследовать работу схемы, заполнить таблицу истинности (табл. 1.20).

*Логический элемент ИСКЛЮЧАЮЩЕЕ ИЛИ, используемый как инвертор*

1. Собрать схему, как показано на рис. 1.18, г.
2. Изменяя состояния входа  $x$ , исследовать работу схемы, заполнить таблицу истинности (табл. 1.21).

Таблица 1.20

Вход	Выход
$x$	$f(x) = x$
0	
1	

Таблица 1.21

Вход	Выход
$x$	$f(x) = \bar{x}$
0	
1	

### 1.3. СОДЕРЖАНИЕ ОТЧЕТА

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

### 1.4. 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

*Цель работы:*

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

### 2.1. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

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 в другой форме. ФАЛ может быть в различных формах, но эти две формы удобны при упрощении ФАЛ стандартными методами.

Пример 2.1. 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 (2.1) видно, что имеются три переменные  $x_1, x_2, x_3$ . При этом переменная  $x_3$  дана в нормальном (неинверсном) виде и 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. 2.1, а.

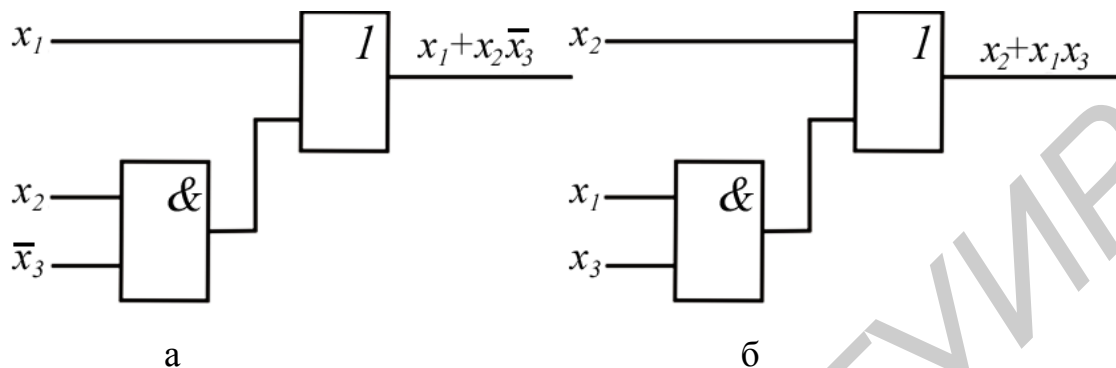


Рис. 2.1. Реализация отдельных термов

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

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

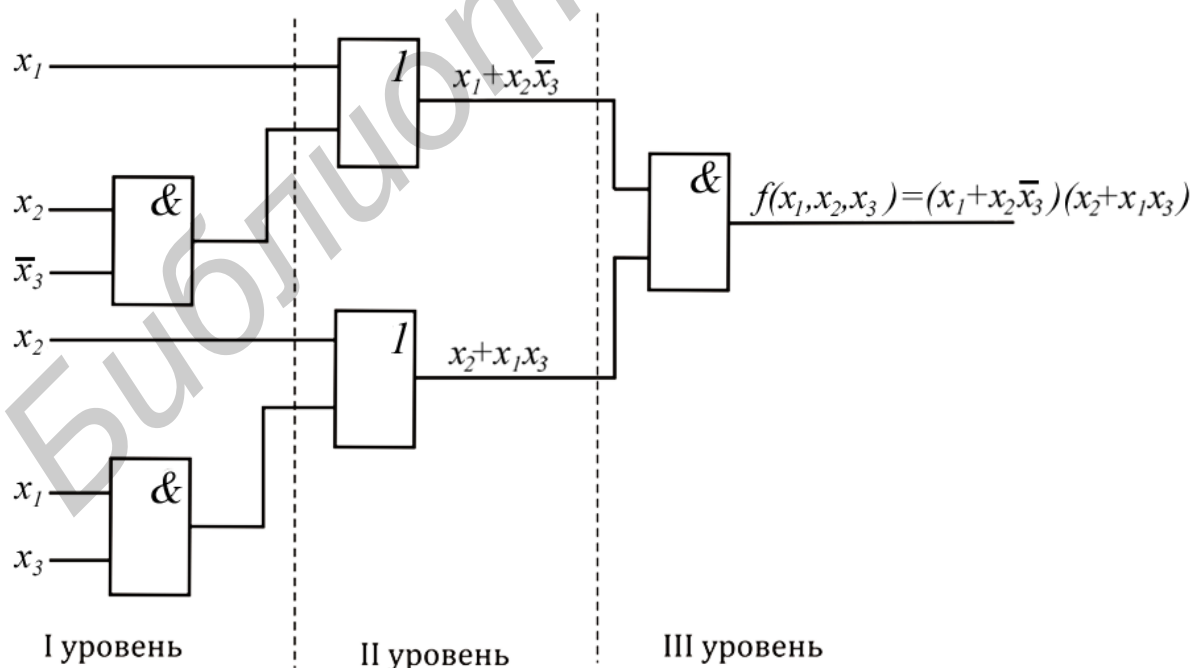


Рис. 2.2. Реализация логической функции в целом

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

### 2.1.1. 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 с помощью И-ИЛИ конфигурации (рис. 2.3).

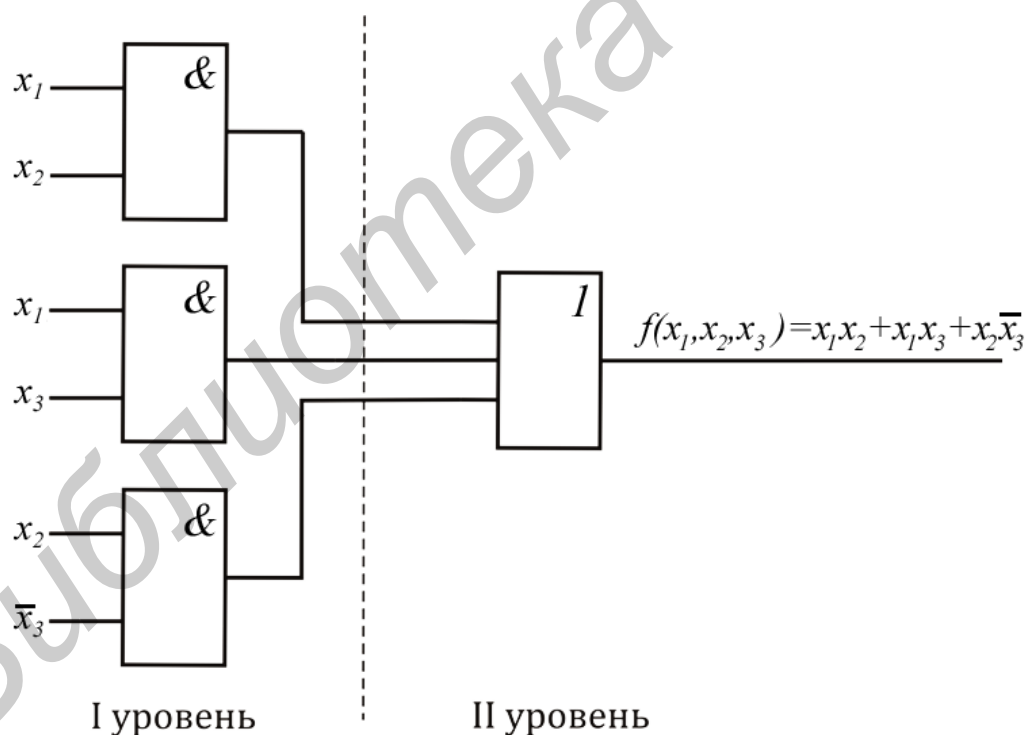


Рис. 2.3. Реализация ФАЛ, представленной в ДНФ (И-ИЛИ конфигурация)

Реализация ФАЛ (см. рис. 2.3) известна как двухуровневая реализация. Первый уровень состоит из И элементов, а второй уровень – из элемента ИЛИ.

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

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

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

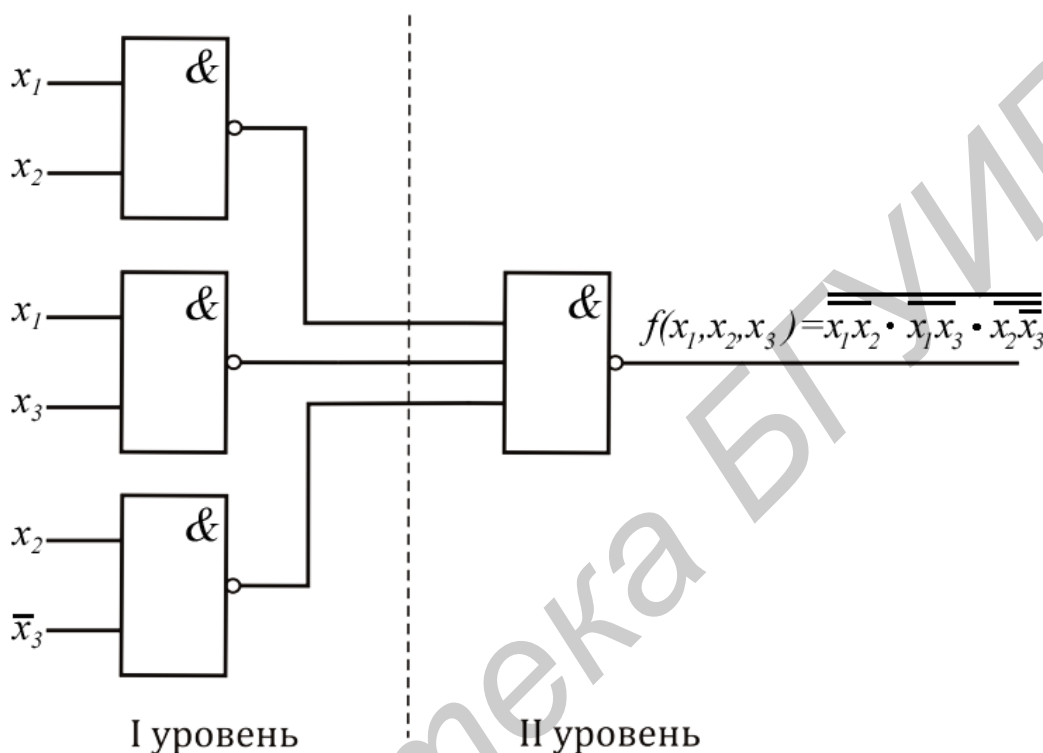


Рис. 2.4. Реализация ФАЛ, используя только И-НЕ элементы

**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 :

$$\begin{aligned} f(x_1, x_2, x_3) &= (x_1 + x_2 \overline{x_3})(x_2 + x_1 x_3) = (x_1 + x_2)(x_1 + \overline{x_3})(x_2 + x_1)(x_2 + x_3) = \\ &= (x_1 + x_2)(x_1 + \overline{x_3})(x_2 + x_3). \end{aligned} \quad (2.4)$$

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 путем использования ИЛИ-И конфигурации (рис. 2.5).

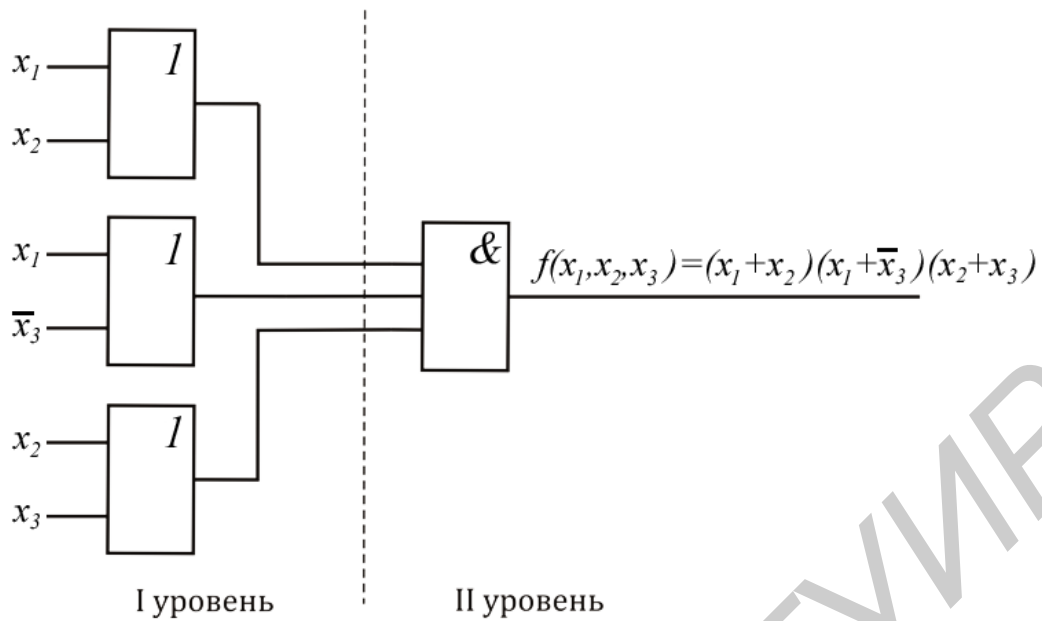


Рис. 2.5. Реализация ФАЛ, представленной в КНФ (ИЛИ-И конфигурация)

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

$$f(x_1, x_2, x_3) = \overline{\overline{(x_1 + x_2)(x_1 + \overline{x_3})(x_2 + x_3)}} = \overline{\overline{x_1 + x_2 + x_1 + \overline{x_3} + x_2 + x_3}}. \quad (2.5)$$

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

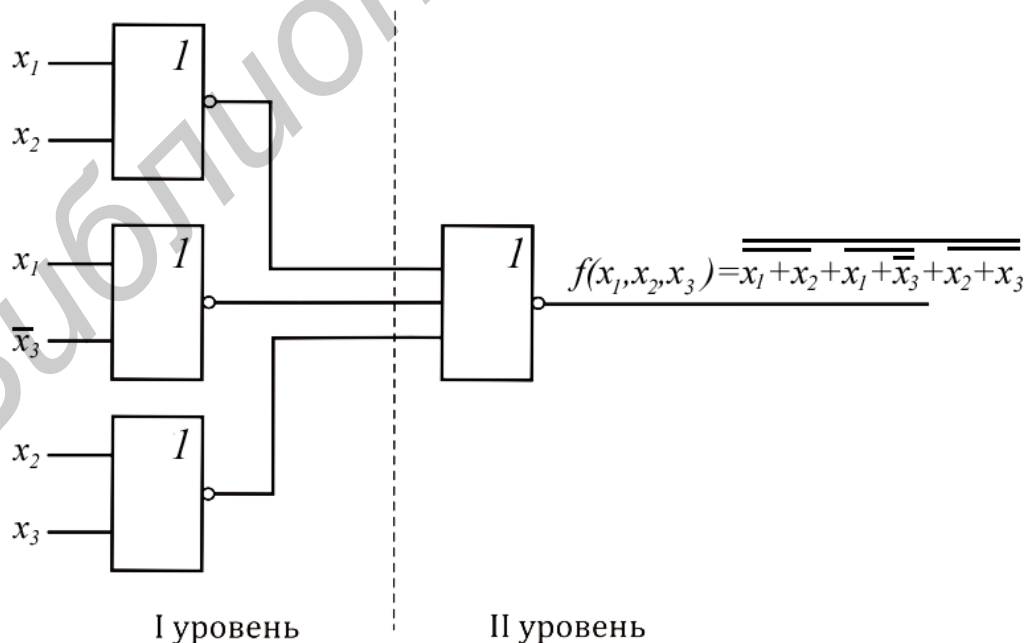


Рис. 2.6. Реализация ФАЛ, используя только ИЛИ-НЕ элементы

**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 :

$$\begin{aligned} \text{a)} f(x_1, x_2, x_3) &= x_1 x_2 + x_1 x_3 + x_2 \bar{x}_3 = x_1 x_2 (x_3 + \bar{x}_3) + x_1 x_3 + x_2 \bar{x}_3 = \\ &= x_1 x_2 x_3 + x_1 x_2 \bar{x}_3 + x_1 x_3 + x_2 \bar{x}_3 = x_1 x_3 (x_2 + 1) + x_2 \bar{x}_3 (x_1 + 1) = x_1 x_3 + x_2 \bar{x}_3; \end{aligned} \quad (2.6)$$

$$\begin{aligned} \text{б)} f(x_1, x_2, x_3) &= (x_1 + x_2)(x_1 + \bar{x}_3)(x_2 + x_3) = (x_1 + x_2 + x_3 \bar{x}_3)(x_1 + \bar{x}_3) \times \\ &\times (x_2 + x_3) = (x_1 + x_2 + x_3)(x_1 + x_2 + \bar{x}_3)(x_1 + \bar{x}_3)(x_2 + x_3) = \\ &= (x_1 + \bar{x}_3)(x_2 + x_3). \end{aligned} \quad (2.7)$$

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

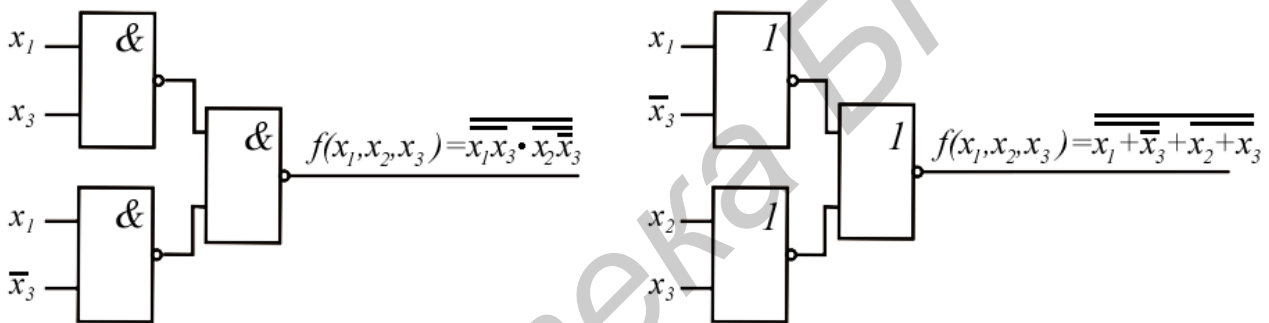


Рис. 2.7. Реализация функции после упрощения

### 2.1.2. Сравнение полученных схем

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

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

3. 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 назы-

вают 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 2.2.** Преобразовать функцию  $f(x_1, x_2, x_3) = x_1x_2 + x_1x_3 + x_2\bar{x}_3$  в СДНФ.

*Решение*

$$\begin{aligned} f(x_1, x_2, x_3) &= x_1x_2 + x_1x_3 + x_2\bar{x}_3 = x_1x_2(x_3 + \bar{x}_3) + x_1x_3(x_2 + \bar{x}_2) + x_2\bar{x}_3(x_1 + \bar{x}_1) = \\ &= x_1x_2x_3 + x_1x_2\bar{x}_3 + x_1x_2x_3 + x_1\bar{x}_2x_3 + \bar{x}_1x_2\bar{x}_3 = x_1x_2x_3 + x_1x_2\bar{x}_3 + x_1\bar{x}_2x_3 + \bar{x}_1x_2\bar{x}_3. \end{aligned}$$

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 2.3.** 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  $2^n$ . Число возможных макстермов (конституент нуля или составляющих нуля) также равно  $2^n$ .

В табл. 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 равна 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 $Y$
$x_1x_2x_3x_4$	$m_i$	$M_i$	$A(x_1, x_2) = B(x_3, x_4)$
0 0 0 0	$\bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4 = m_0$	$x_1 + x_2 + x_3 + x_4 = M_0$	1
0 0 0 1	$\bar{x}_1\bar{x}_2\bar{x}_3x_4 = m_1$	$x_1 + x_2 + x_3 + \bar{x}_4 = M_1$	0
0 0 1 0	$\bar{x}_1\bar{x}_2x_3\bar{x}_4 = m_2$	$x_1 + x_2 + \bar{x}_3 + x_4 = M_2$	0
0 0 1 1	$\bar{x}_1\bar{x}_2x_3x_4 = m_3$	$x_1 + x_2 + \bar{x}_3 + \bar{x}_4 = M_3$	0
0 1 0 0	$\bar{x}_1x_2\bar{x}_3\bar{x}_4 = m_4$	$x_1 + \bar{x}_2 + x_3 + x_4 = M_4$	0
0 1 0 1	$\bar{x}_1x_2\bar{x}_3x_4 = m_5$	$x_1 + \bar{x}_2 + x_3 + \bar{x}_4 = M_5$	1
0 1 1 0	$\bar{x}_1x_2x_3\bar{x}_4 = m_6$	$x_1 + \bar{x}_2 + \bar{x}_3 + x_4 = M_6$	0
0 1 1 1	$\bar{x}_1x_2x_3x_4 = m_7$	$x_1 + \bar{x}_2 + \bar{x}_3 + \bar{x}_4 = M_7$	0
1 0 0 0	$x_1\bar{x}_2\bar{x}_3\bar{x}_4 = m_8$	$\bar{x}_1 + x_2 + x_3 + x_4 = M_8$	0
1 0 0 1	$x_1\bar{x}_2\bar{x}_3x_4 = m_9$	$\bar{x}_1 + x_2 + x_3 + \bar{x}_4 = M_9$	0
1 0 1 0	$x_1\bar{x}_2x_3\bar{x}_4 = m_{10}$	$\bar{x}_1 + x_2 + \bar{x}_3 + x_4 = M_{10}$	1
1 0 1 1	$x_1\bar{x}_2x_3x_4 = m_{11}$	$\bar{x}_1 + x_2 + \bar{x}_3 + \bar{x}_4 = M_{11}$	0
1 1 0 0	$x_1x_2\bar{x}_3\bar{x}_4 = m_{12}$	$\bar{x}_1 + \bar{x}_2 + x_3 + x_4 = M_{12}$	0
1 1 0 1	$x_1x_2\bar{x}_3x_4 = m_{13}$	$\bar{x}_1 + \bar{x}_2 + x_3 + \bar{x}_4 = M_{13}$	0
1 1 1 0	$x_1x_2x_3\bar{x}_4 = m_{14}$	$\bar{x}_1 + \bar{x}_2 + \bar{x}_3 + x_4 = M_{14}$	0
1 1 1 1	$x_1x_2x_3x_4 = m_{15}$	$\bar{x}_1 + \bar{x}_2 + \bar{x}_3 + \bar{x}_4 = M_{15}$	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 СКНФ.

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

$$\begin{aligned}
 Y &= m_0 + m_5 + m_{10} + m_{15} = \sum m(0,5,10,15) = \\
 &= \bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4 + \bar{x}_1\bar{x}_2\bar{x}_3x_4 + x_1\bar{x}_2x_3\bar{x}_4 + x_1\bar{x}_2x_3x_4.
 \end{aligned}
 \tag{2.8}$$

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

$$\begin{aligned}
 Y &= M_1M_2M_3M_4M_6M_7M_8M_9M_{11}M_{12}M_{13}M_{14} = \\
 &= \Pi M(1,2,3,4,6,7,8,9,11,12,13,14) = \\
 &= (x_1 + x_2 + x_3 + \bar{x}_4)(x_1 + x_2 + \bar{x}_3 + x_4) \cdot \dots \cdot (x_1 + \bar{x}_2 + \bar{x}_3 + x_4).
 \end{aligned}
 \tag{2.9}$$

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 путем использования этого свойства и наоборот.

## 2.2. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

**Оборудование и компоненты:** универсальная лабораторная установка IDL-800, ИС 1533ЛА4(74ALS00) – четыре логических элемента 2И-НЕ; ИС 1533ЛА3(74ALS10) – три логических элемента 3И-НЕ; ИС 1533ЛЕ1(74ALS02) – четыре логических элемента 2ИЛИ-НЕ; ИС 1533ЛЕ4(74ALS27) – три логических элемента 3ИЛИ-НЕ. ФАЛ задана таблицей истинности (табл. 2.2).

Таблица 2.2

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

### 2.2.1. Исследование представления ФАЛ в ДНФ

*Для заданной таблицы истинности*

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

### 2.2.2. Исследование представление ФАЛ в КНФ

*Для заданной таблицы истинности*

1. Записать логическое выражение для  $Y$  в СКНФ.



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

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

### 2.3. СОДЕРЖАНИЕ ОТЧЕТА

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

### 2.4. 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

*Цель работы:*

1. Исследование функционирования сумматоров и вычитателей.
2. Исследование суммирования/вычитания в дополнительных кодах.

### 3.1. 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 оба 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*.

#### 3.1.1 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.

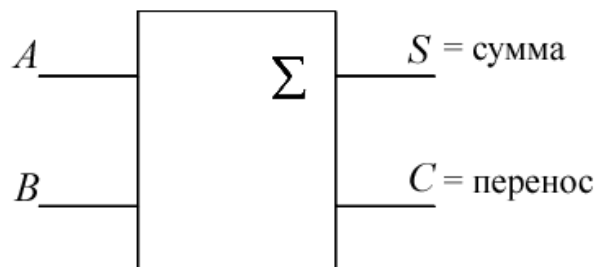


Рис. 3.1. Условное обозначение полусумматора

Таблица 3.1

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 полусумматора (табл. 3.1) 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).

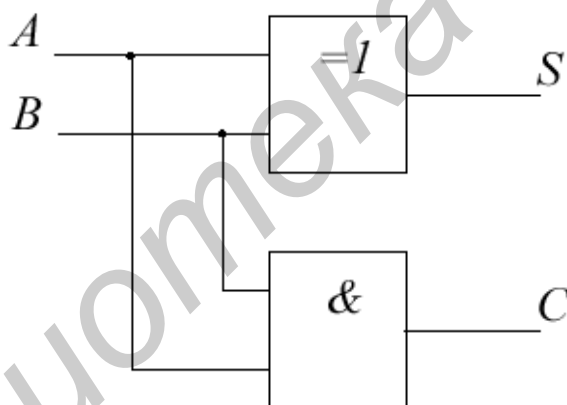


Рис. 3.2. Схема полусумматора

### 3.1.2. 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 3.2

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).



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.

$$\begin{aligned}
 S_n &= (\overline{A_n} \overline{B_n} C_{n-1} + \overline{A_n} B_n \overline{C_{n-1}}) + (A_n \overline{B_n} \overline{C_{n-1}} + A_n B_n C_{n-1}) = \\
 &= \overline{A_n} (B_n \oplus C_{n-1}) + A_n (\overline{B_n} \oplus \overline{C_{n-1}}) = A_n \oplus B_n \oplus C_{n-1}.
 \end{aligned}
 \tag{3.3}$$

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

$$\begin{aligned}
 C_n &= (A_n \overline{B_n} C_{n-1} + \overline{A_n} B_n C_{n-1}) + (A_n B_n C_{n-1} + A_n B_n \overline{C_{n-1}}) = \\
 &= C_{n-1} (A_n \oplus B_n) + A_n B_n.
 \end{aligned}
 \tag{3.4}$$

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

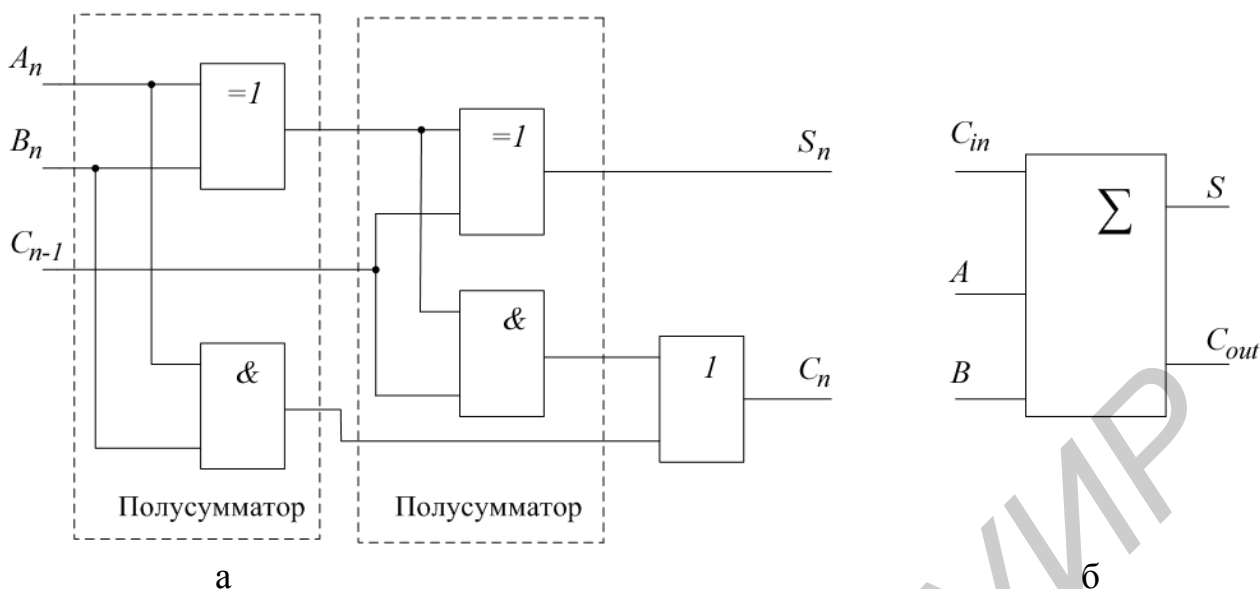


Рис. 3.4. Полный сумматор:  
 а – логическая схема;  
 б – условное обозначение

### 3.1.3. 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  $A$  и  $B$  четырех полных сумматоров.  $A_0, B_0$  – это, конечно, наименее значимые биты. Выход переноса  $C_{out}$  каждого сумматора соединен непосредственно с входом переноса  $C_{in}$  следующего более значимого полного сумматора. Выход  $C_{out}$  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. 3.5, а) более точно называют псевдопараллельным сумматором.

В интегральном исполнении выпускается четырехразрядный параллельный сумматор 1533ИМ3(74ALS83) (рис. 3.5, б).

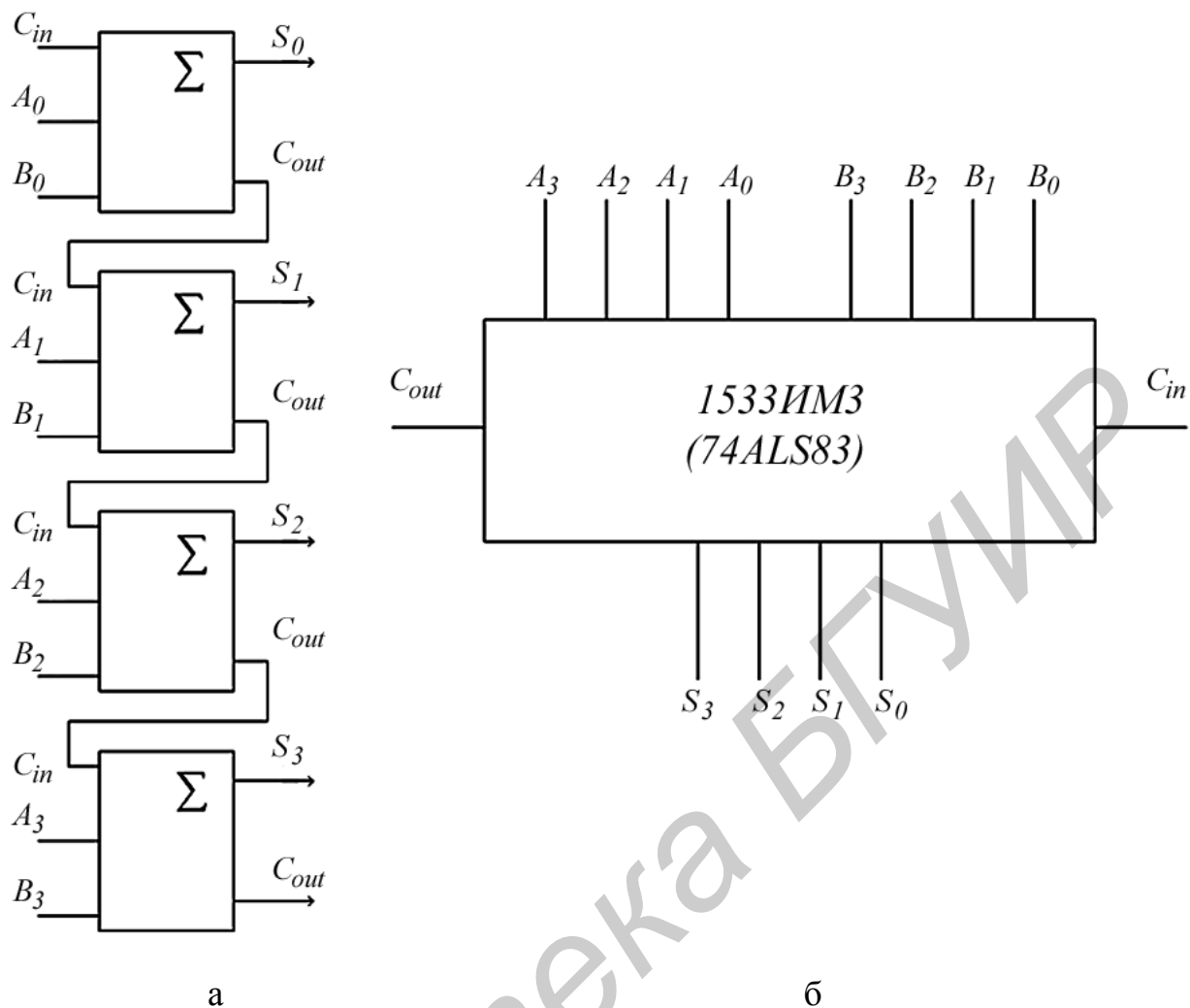


Рис. 3.5. Параллельный сумматор

### 3.1.4. Вычитатели

#### Полувывчитатель

Логическая схема, которая осуществляет вычитание  $B$  (вычитаемое) из  $A$  (уменьшаемое), где  $A$  и  $B$  – это однобитовые числа, называется полувывчитателем. Процесс вычитания может быть представлен с помощью таблицы истинности (табл. 3.3).

Таблица 3.3

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

Здесь  $A$  и  $B$  – это входы, а выходы  $D$  – разность и  $C$  – заем.

Из таблицы истинности (см. табл. 3.3)

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

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

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

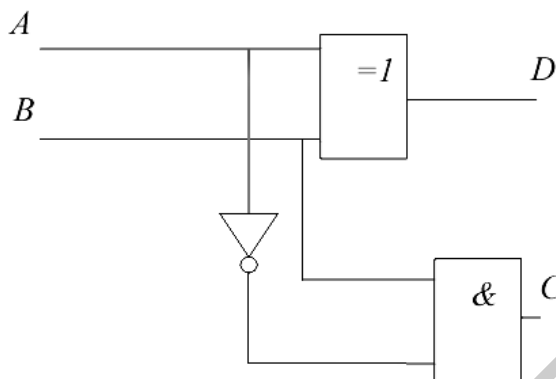


Рис. 3.6. Логическая схема полувывчитателя

### Полный вычитатель

Для выполнения многобитового вычитания необходим полный вычитатель, 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 3.4

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 (табл. 3.4) 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.

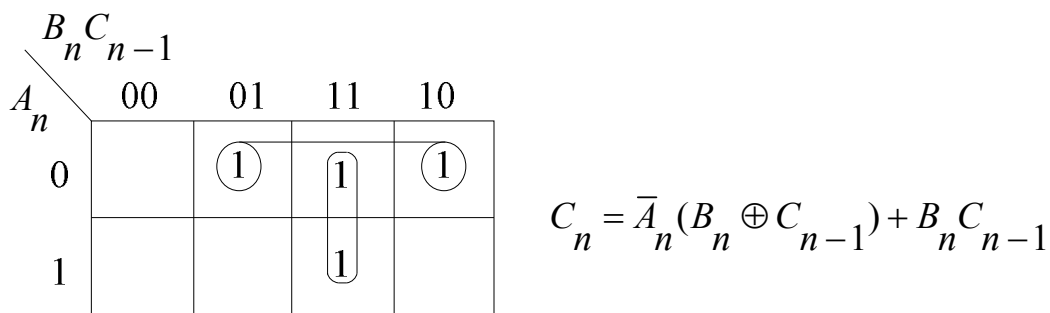


Рис. 3.7. Упрощение для  $C_n$  с помощью карты Карно

Существует три варианта упрощения выражения для заема  $C_n$ . Два из них дают схему полного сумматора, состоящего из двух полувычитателей и схемы ИЛИ. Однако наиболее интересным представляется третий вариант (см. рис. 3.7):

$$\begin{aligned}
 C_n &= (\bar{A}_n \bar{B}_n C_{n-1} + \bar{A}_n B_n \bar{C}_{n-1}) + (\bar{A}_n B_n C_{n-1} + A_n B_n C_{n-1}) = \\
 &= \bar{A}_n (B_n \oplus C_{n-1}) + B_n C_{n-1}.
 \end{aligned}
 \tag{3.8}$$

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

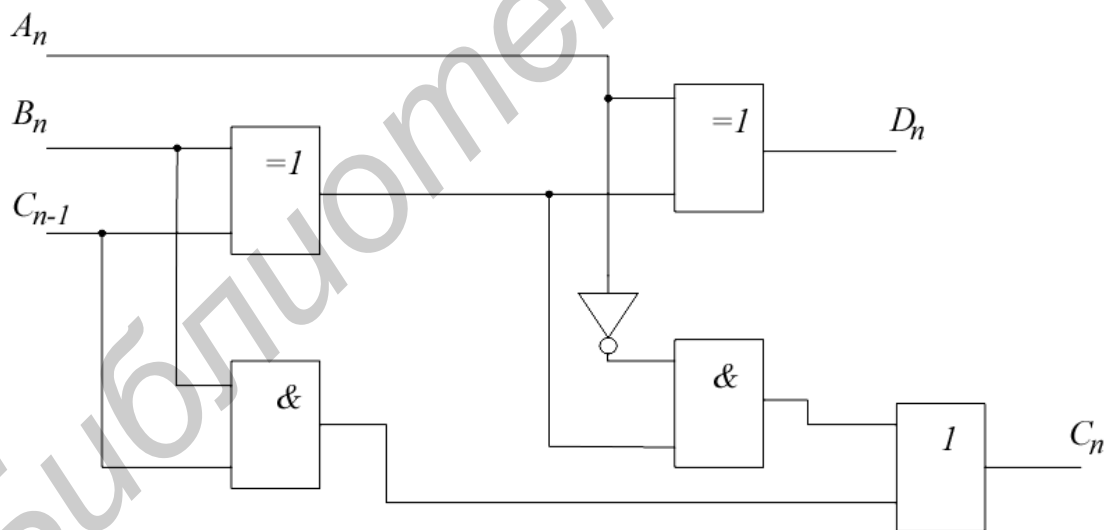


Рис. 3.8. Полный вычитатель

Полный вычитатель (см. 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 (см. рис. 3.8) более быстродействующая по сравнению с обычной схемой полного вычитателя.



### 3.1.5. 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  $B$  передаются на входы сумматора без изменения. Схема работает как сумматор.

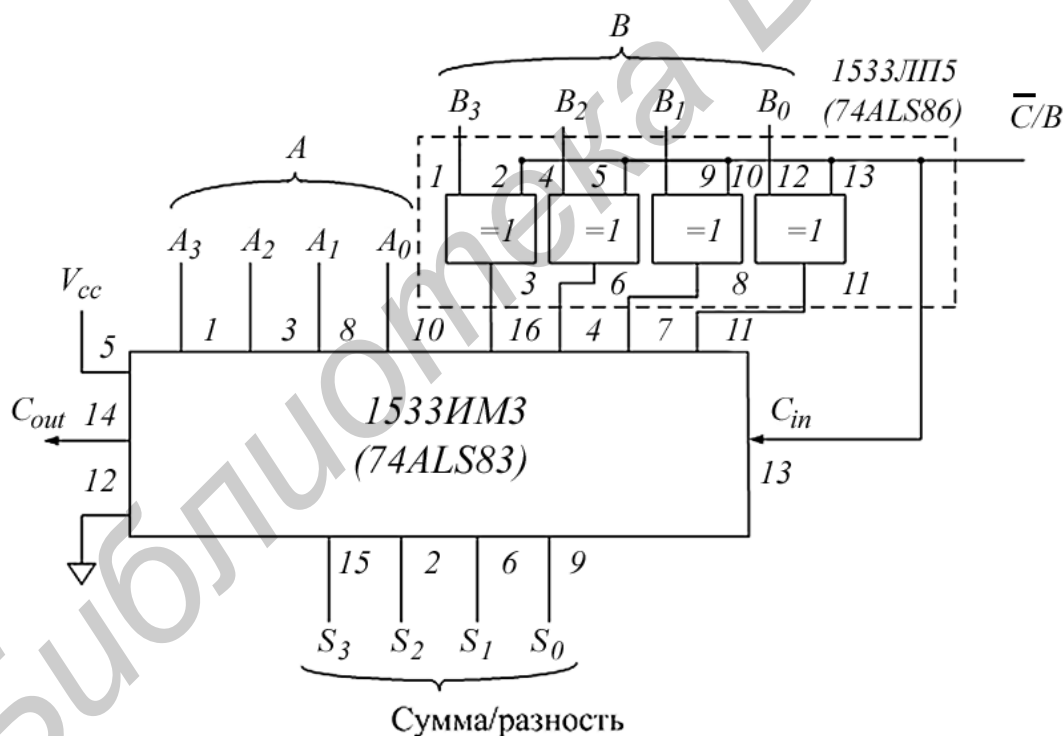


Рис. 3.9. Сумматор/вычитатель

Когда управляющий сигнал имеет высокий уровень, на вход  $C_{in}$  подается лог.1 и каждая схема ИСКЛЮЧАЮЩЕЕ ИЛИ инвертирует биты числа  $B$ . Таким образом осуществляются инвертирование каждого бита числа  $B$  и прибавление 1 (по входу  $C_{in}$ ). В результате выполняется операция дополне-

ния с числом  $B$ . При этом суммирование числа  $A$  с дополнением числа  $B$  равносильно вычитанию числа  $B$  из числа  $A$ .

### 3.2. 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) – четырехразрядный сумматор; ИС 1533ЛН1(74ALS04) – шесть инверторов.

#### 3.2.1. Исследование сумматоров

##### *Полусумматор*

1. Используя логические элементы, собрать схему полусумматора (см. рис. 3.2).
2. Изменяя состояния входов  $A$  и  $B$ , исследовать работу полусумматора, получить таблицу истинности полусумматора (см. табл. 3.1).

##### *Полный сумматор*

1. Используя логические элементы, собрать схему полного сумматора (см. рис. 3.4).
2. Изменяя состояния входов  $A_n, B_n, C_{n-1}$ , исследовать работу полного сумматора, получить таблицу истинности полного сумматора (см. табл. 3.2).

#### 3.2.2. Исследование вычитателей

##### *Полувывчитатель*

1. Используя логические элементы, собрать схему полувывчитателя (см. рис. 3.6).
2. Изменяя состояния входов  $A$  и  $B$ , исследовать работу полувывчитателя, получить таблицу истинности полувывчитателя (см. табл. 3.3).

##### *Полный вычитатель*

1. Используя логические элементы, собрать схему полного вычитателя (см. рис. 3.8).
2. Изменяя состояния входов  $A_n, B_n, C_{n-1}$ , исследовать работу полного вычитателя, получить таблицу истинности полного вычитателя (см. табл. 3.4).

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

1. Собрать схему сумматора/вычитателя (см. рис. 3.9).
2. Исследовать работу сумматора/вычитателя, используя несколько примеров. Числа  $A$  и  $B$  представляются в дополнительном коде.
3. Объяснить результаты экспериментов.

### **3.3. СОДЕРЖАНИЕ ОТЧЕТА**

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

### **3.4. 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

*Цель работы:*

1. Синтез преобразователей двоичных кодов.
2. Исследование функционирования преобразователей двоичных кодов.

#### 4.1. 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 дво- ичный 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*

Используется для представления чисел в двоичной системе счисления. В табл. 4.1 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 слова 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 одноразрядный, 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	0 <u>1</u> ← <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	0 <u>1</u> 0 ← <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.

**4.1.1. 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
$B_3 \ B_2 \ B_1 \ B_0$	$G_3 \ G_2 \ G_1 \ G_0$
1	2
0 0 0 0	0 0 0 0
0 0 0 1	0 0 0 1
0 0 1 0	0 0 1 1

Окончание табл. 4.2

1				2			
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.

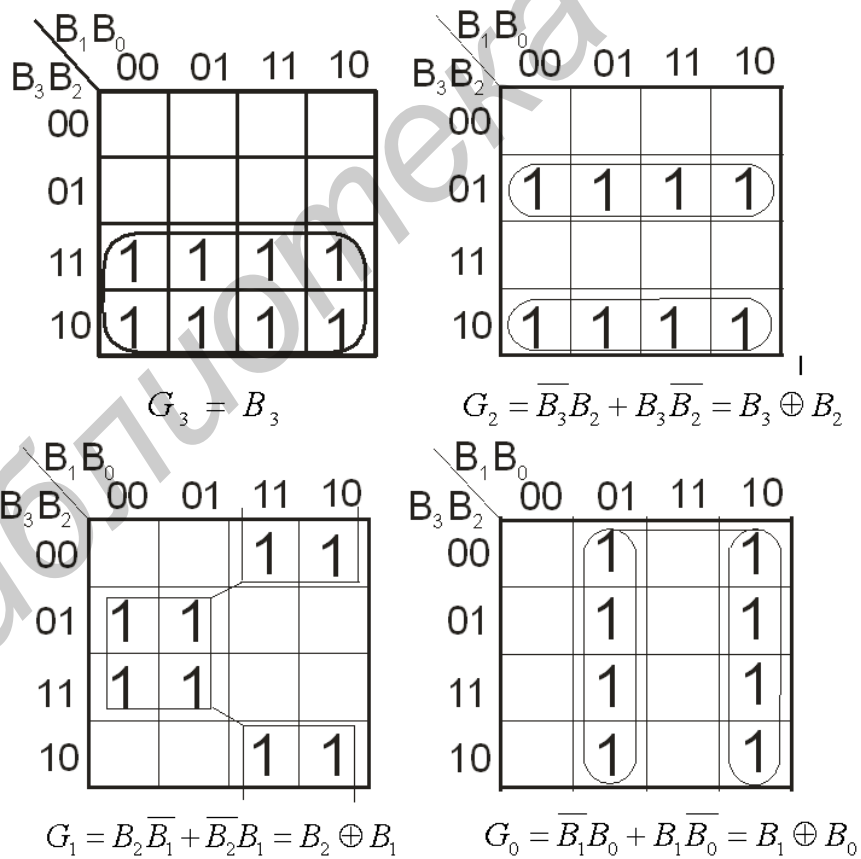


Рис. 4.2. Карты Карно преобразователя двоичного кода в код Грея

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.

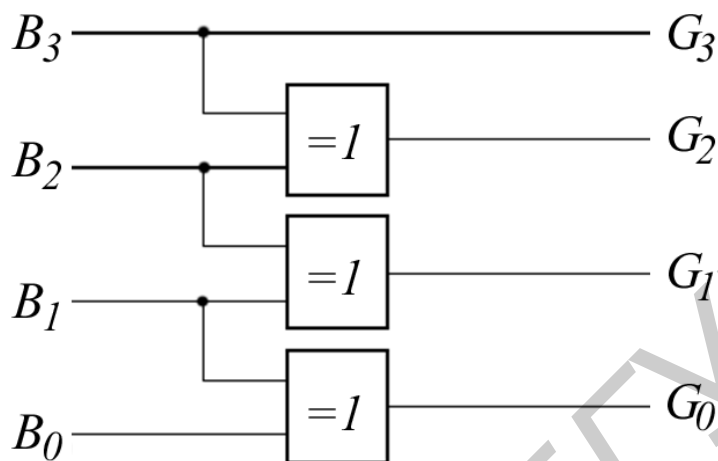


Рис. 4.3. Преобразователь двоичного кода в код Грея

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.

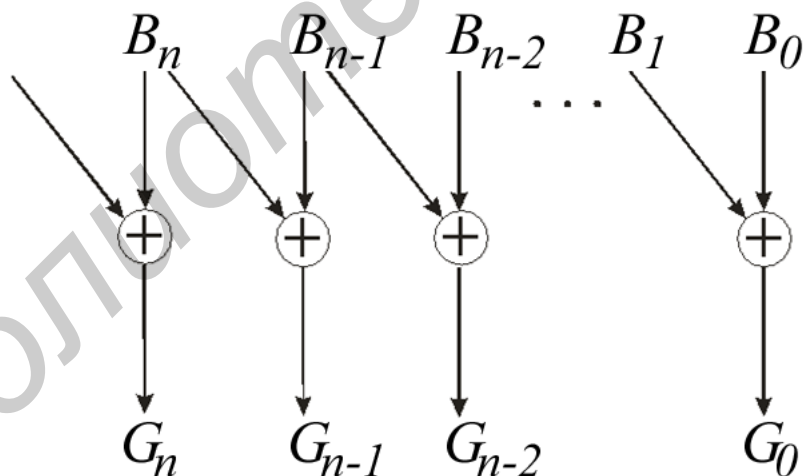


Рис. 4.4. Общая схема преобразования двоичного числа в код Грея

#### 4.1.2. Преобразователь кода Грея в двоичный код

Используя табл. 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).



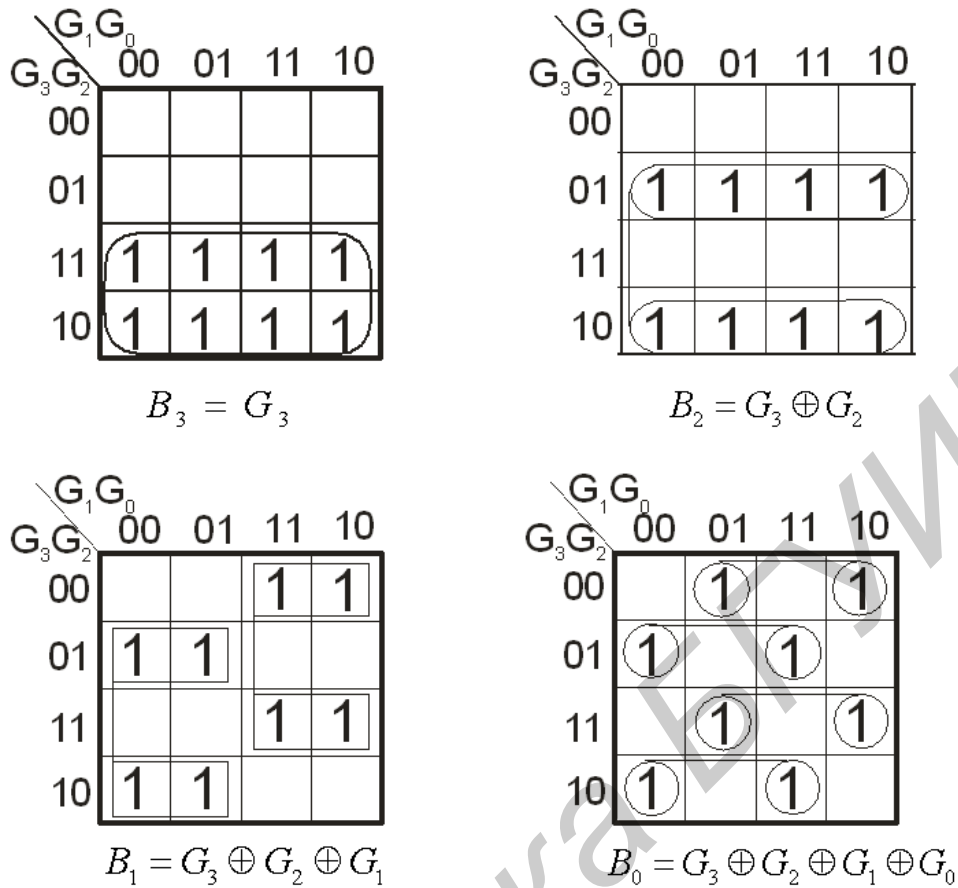


Рис. 4.5. Карты Карно для преобразователя кода Грея в двоичный код

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

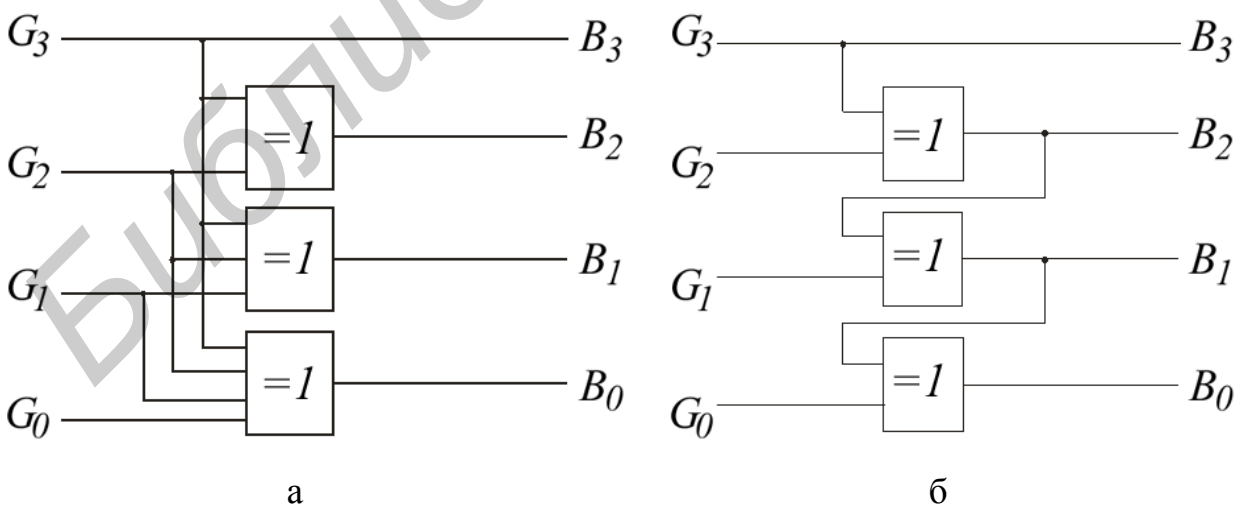


Рис. 4.6. Преобразователь кода Грея в двоичный код

Логическая схема рис. 4.6, а 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:

$$\begin{aligned} B_3 &= G_3; \\ B_2 &= G_3 \oplus G_2; \\ B_1 &= G_3 \oplus G_2 \oplus G_1 = B_2 \oplus G_1; \\ B_0 &= G_3 \oplus G_2 \oplus G_1 \oplus G_0 = B_1 \oplus G_0. \end{aligned} \quad (4.1)$$

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

#### 4.1.3. Преобразователь *BCD*-кода в прямой двоичный код

Рассмотрим схему преобразования *BCD*-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 семизрядный двоичный код  $b_6 b_5 b_4 b_3 b_2 b_1 b_0$ . На схеме показаны также вес каждого *BCD*-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).

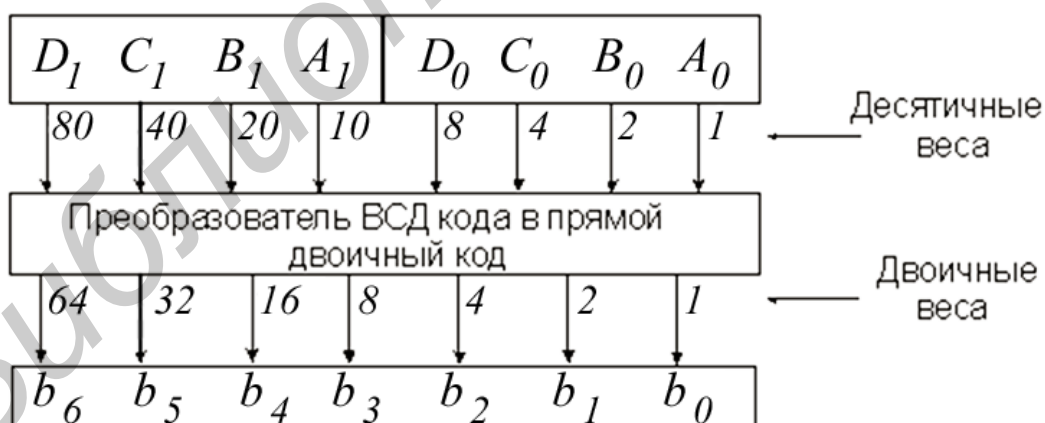


Рис. 4.7. Схема преобразования двухразрядного *BCD*-кода в прямой двоичный код

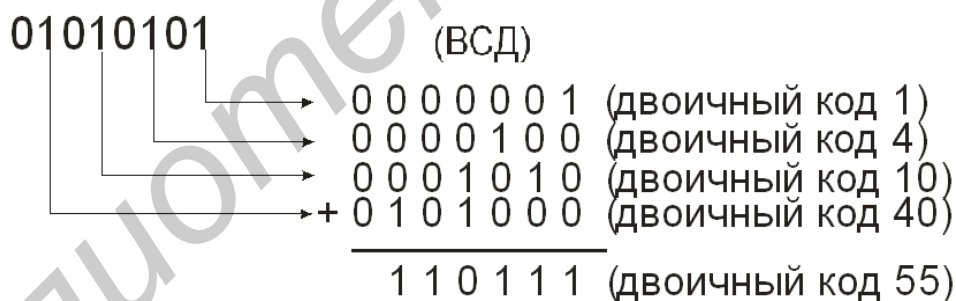
Десятичные веса каждого бита в двоично-десятичном представлении (рис. 4.7) могут быть представлены двоичными эквивалентами, как приведено в табл. 4.3. Используя эти веса, можно осуществить преобразование дво-

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

Таблица 4.3

<i>BCD</i> - биты	Десятичные веса	Двоичные эквиваленты						
		<i>B</i> <sub>6</sub>	<i>B</i> <sub>5</sub>	<i>B</i> <sub>4</sub>	<i>B</i> <sub>3</sub>	<i>B</i> <sub>2</sub>	<i>B</i> <sub>1</sub>	<i>B</i> <sub>0</sub>
<i>A</i> <sub>0</sub>	1	0	0	0	0	0	0	1
<i>B</i> <sub>0</sub>	2	0	0	0	0	0	1	0
<i>C</i> <sub>0</sub>	4	0	0	0	0	1	0	0
<i>D</i> <sub>0</sub>	8	0	0	0	1	0	0	0
<i>A</i> <sub>1</sub>	10	0	0	0	1	0	1	0
<i>B</i> <sub>1</sub>	20	0	0	1	0	1	0	0
<i>C</i> <sub>1</sub>	40	0	1	0	1	0	0	0
<i>D</i> <sub>1</sub>	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.



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).

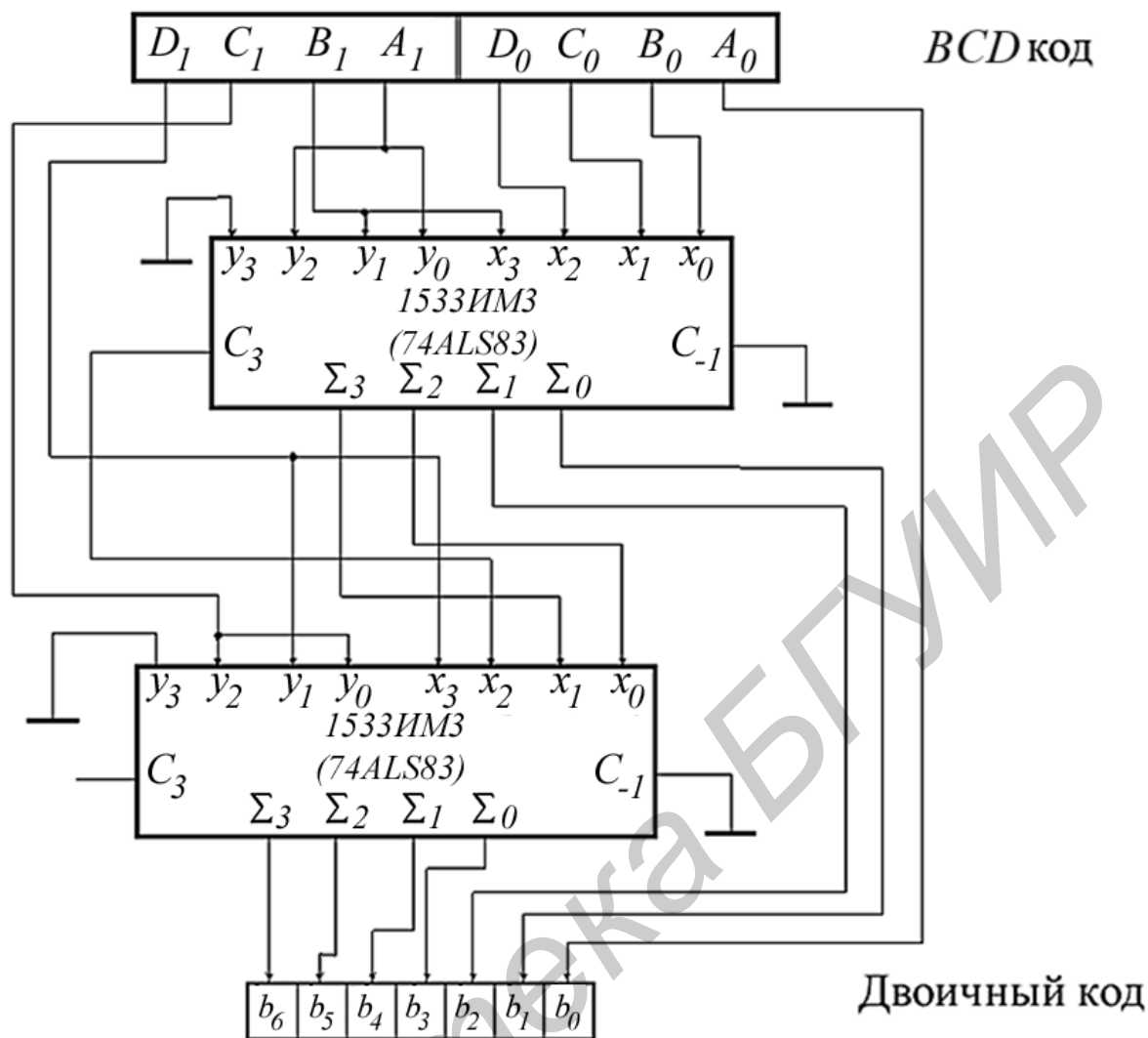


Рис. 4.8. Преобразователь двоично-десятичного в двоичный код

#### 4.2. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

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

##### 4.2.1. 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.

#### **4.2.2. Исследование преобразователя двоично-десятичного кода с избытком 3 в *B**C**D*-код**

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

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

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

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

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

#### **4.2.5. Исследование преобразователя двоично-десятичного *B**C**D* кода в двоичный код**

1. Собрать схему преобразователя (см. рис. 4.8).
2. Исследовать работу преобразователя.

### **4.3. СОДЕРЖАНИЕ ОТЧЕТА**

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

### **4.4. КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. Объяснить работу преобразователей кодов, исследуемых в работе.
2. Объяснить преобразование  $n$ -разрядного двоичного кода в код Грея.

## СИНТЕЗ КОМБИНАЦИОННЫХ СХЕМ С ИСПОЛЬЗОВАНИЕМ МУЛЬТИПЛЕКСОРОВ И ДЕМУЛЬТИПЛЕКСОРОВ

*Цель работы :*

1. Изучение мультиплексоров и демultipлексоров.
2. Изучение синтеза комбинационных схем на основе мультиплексоров и на основе демultipлексоров.

### 5.1. 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 демultipлексоры, которые широко представлены как ИС средней степени интеграции. Мультиплексоры и демultipлексоры могут успешно использоваться для реализации различных комбинационных устройств. При этом уменьшается количество требуемых ИС, повышается надежность и снижается стоимость реализации КС.

#### 5.1.1. 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.

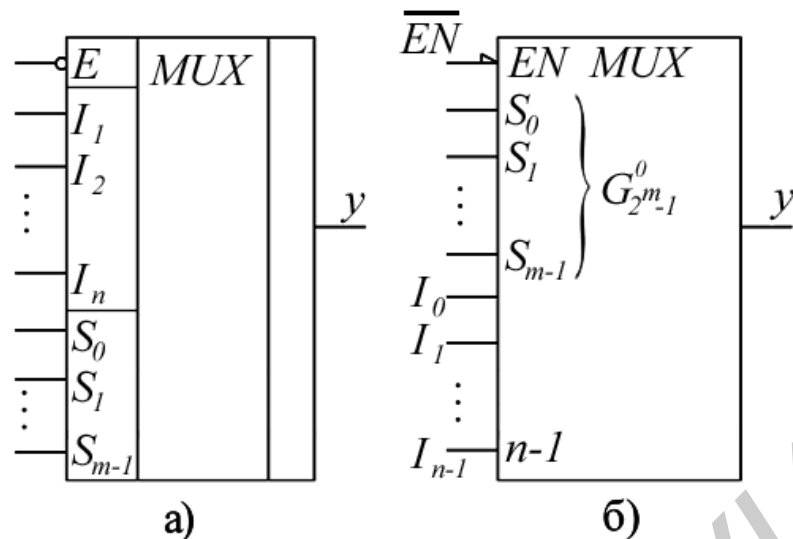


Рис. 5.1. Условное обозначение мультиплексора:  
 а – используемое в отечественной справочной литературе;  
 б – рекомендуемое к использованию международным стандартом IEEE

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. Этот вход 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 табл. 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

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

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.

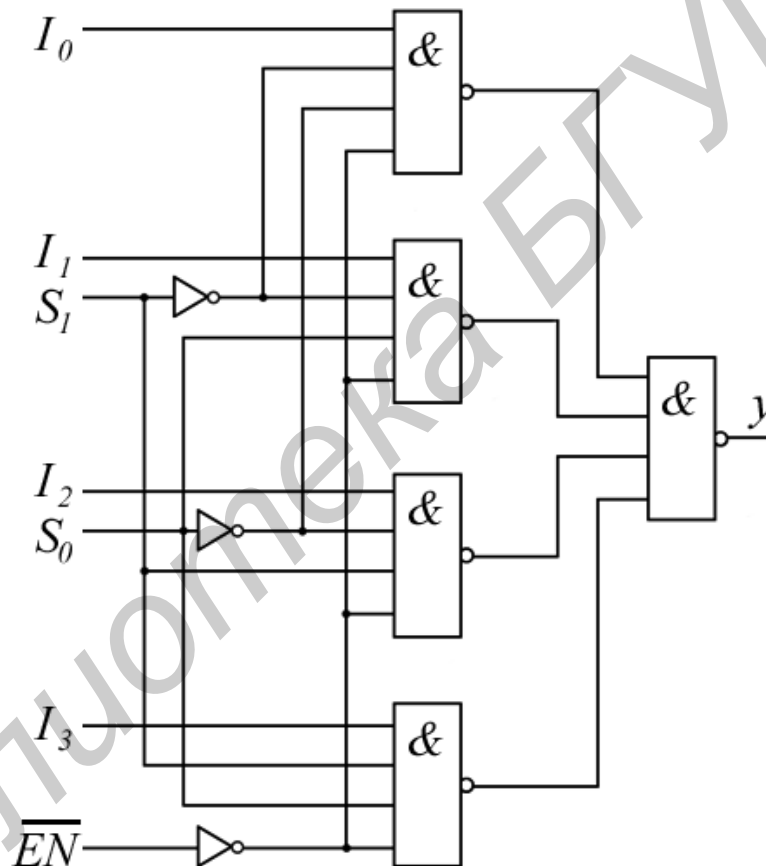


Рис. 5.2. Логическая схема мультиплексора 4:1 с разрешающим входом  $\overline{EN}$

Промышленностью выпускаются мультиплексоры размерностью 8:1 и 16:1 со стробирующим входом и без него, а также 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).

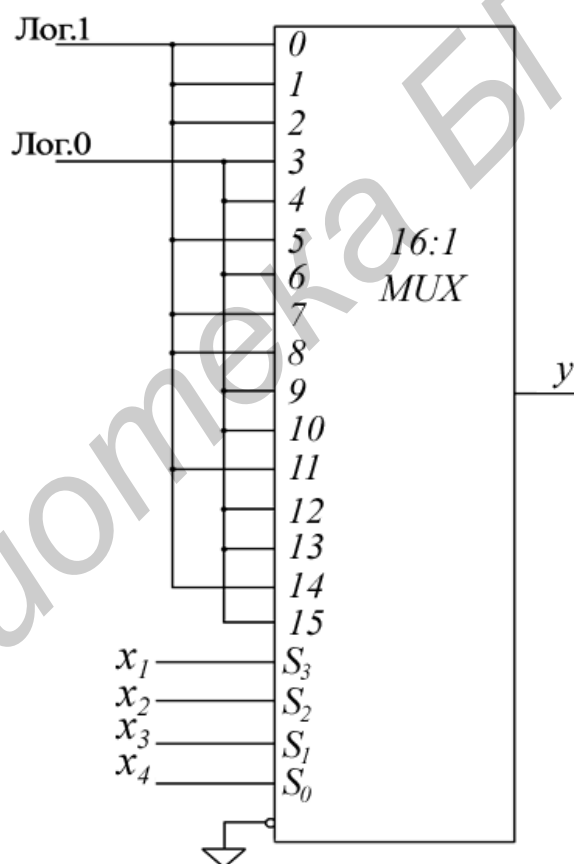


Рис. 5.3. Реализация ФАЛ  $f(x_1, x_2, x_3, x_4)$  из примера 5.1

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

Рассмотрим это на примере.

**Пример 5.2.** Реализовать ФАЛ, заданную таблицей истинности (табл. 5.2).

*Решение.* 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 (см. табл. 5.2) 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	
0	0	1	0	0	$x_4$
0	0	1	1	1	
0	1	0	0	0	$x_4$
0	1	0	1	1	
0	1	1	0	1	1
0	1	1	1	1	
1	0	0	0	0	0
1	0	0	1	0	
1	0	1	0	1	$\bar{x}_4$
1	0	1	1	0	
1	1	0	0	0	$x_4$
1	1	0	1	1	
1	1	1	0	1	$\bar{x}_4$
1	1	1	1	0	

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.

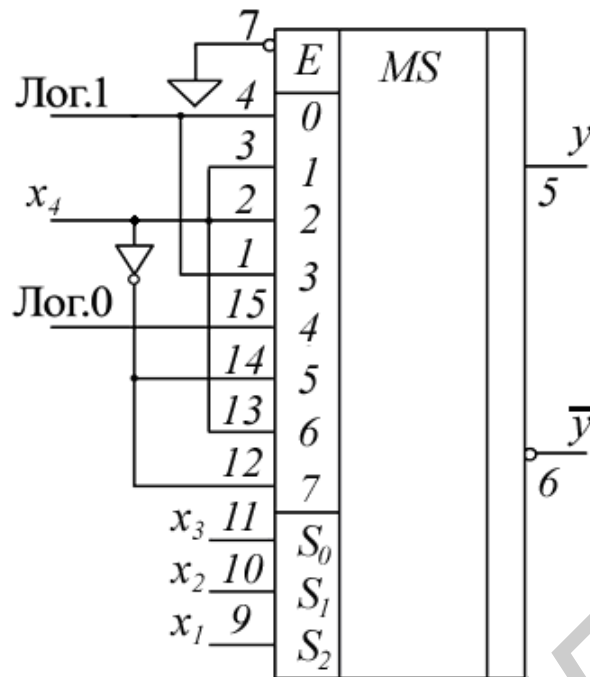


Рис. 5.4. Реализация ФАЛ, пример 5.2, с использованием мультиплексора КР1533КП7(74ALS151)

Для синтеза ФАЛ можно использовать карты Карно (рис. 5.5). Парные наборы на ней расположены рядом, т. е. являются соседними. Объединим эти наборы и пометим числом, соответствующим наборам переменных  $x_1, x_2, x_3$ . Можно заметить, что парные наборы легко выделяются проведением разделительных линий по переменным  $x_1, x_2, x_3$  (рис. 5.5, б). После объединения парных наборов получается карта Карно для трех переменных.

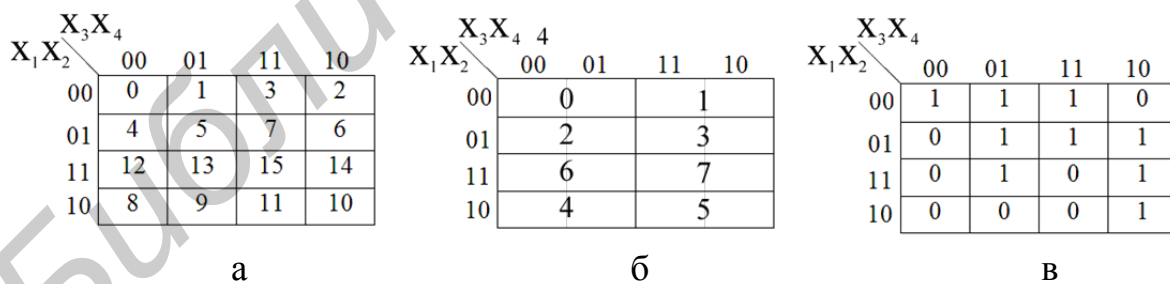


Рис. 5.5. Синтез ФАЛ с помощью карт Карно

Зададим теперь ФАЛ с помощью карты Карно (рис. 5.5, в). Сопоставляя рис. 5.5, б и рис. 5.5, в, можно заметить, что в клетке с номером 0  $Y = 1$  и, следовательно,  $I_0 = 1$ . В клетке с номером 1  $Y = x_4$ , что дает  $I_1 = x_4$ . Про-

должая рассмотрение дальше, получим итоговый результат:  $I_0 = I_3 = 1$ ,  $I_1 = I_2 = I_6 = x_4$ ,  $I_4 = 0$ ,  $I_5 = I_7 = \bar{x}_4$ .

До сих пор мы различали парные наборы по переменной  $x_4$ , которая затем подавалась на информационные входы. Однако в качестве «выделенной» может быть взята любая переменная. Более того, для технической реализации не безразлично, какую переменную следует выделять. Это связано с тем, что выбор выделенной переменной определяет количество информационных входов, на которые подаются константы 0, 1. Такие информационные входы не нагружают предыдущие цепи, поэтому их желательно иметь как можно больше. Очевидно, что для достижения этого в качестве выделенной следует использовать переменную, от которой ФАЛ зависит меньше всего. Последнее можно установить по минимальной дизъюнктивной форме, подсчитав количество вхождений переменной в эту форму как с инверсией, так и без нее.

### 5.1.3. Увеличение размерности мультиплексора

Максимальный размер мультиплексора, выпускаемого промышленностью, 16:1. Мультиплексоры с большим числом входов можно построить из мультиплексоров с меньшим числом входов. Используются два метода, которые мы проиллюстрируем следующими примерами.

**Пример 5.3.** Построить мультиплексор размерностью 32:1. Такой мультиплексор можно построить (рис. 5.6), используя два мультиплексора 16:1.

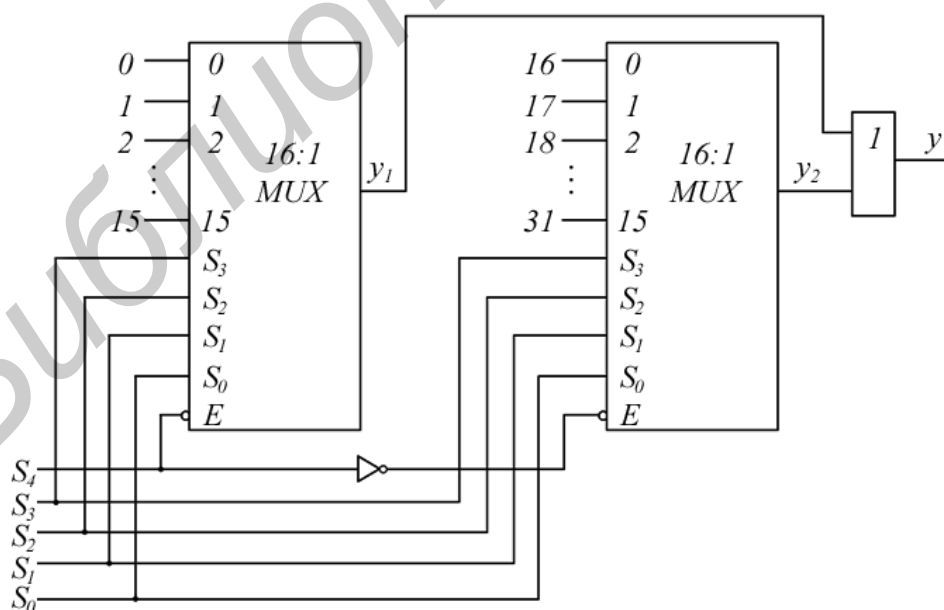


Рис. 5.6. Мультиплексор 32:1 с использованием двух мультиплексоров 16:1

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

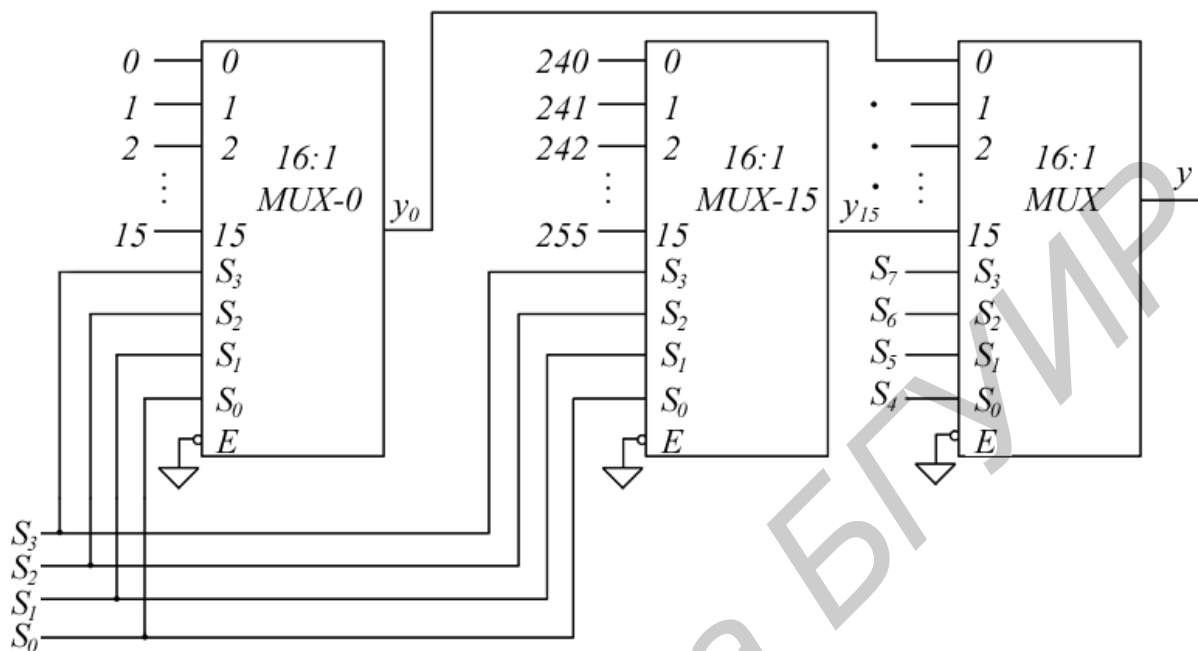


Рис. 5.7. Мультиплексор 256:1с использованием 17 мультиплексоров 16:1

#### 5.1.4. Декодеры/демультиплексоры 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.

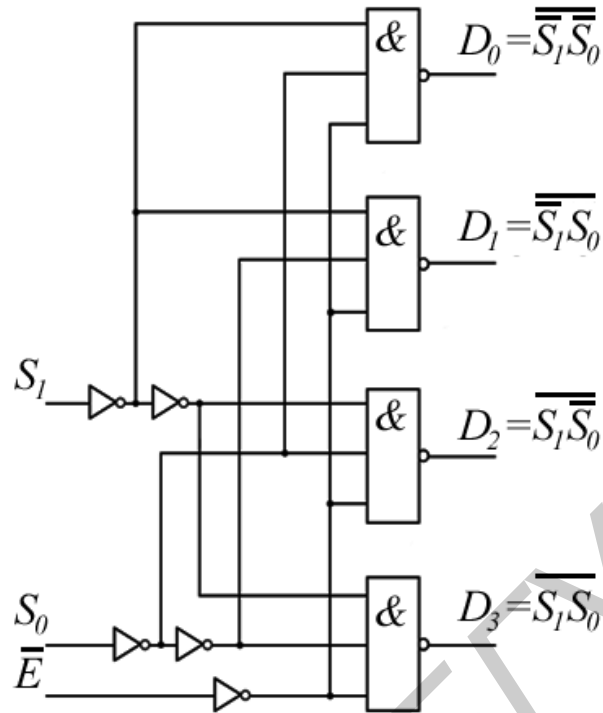


Рис. 5.8. Логическая схема декодера 2:4 с разрешающим входом  $\bar{E}$

Таблица истинности декодера 2:4 дана в табл. 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

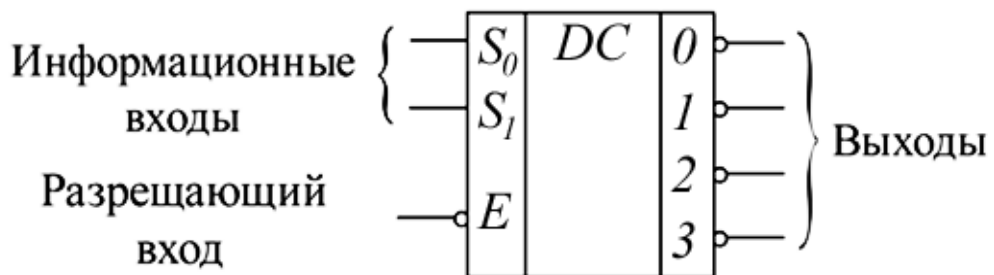


Рис. 5.9. Условное обозначение декодера 2:4

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 может работать как демультиплексор, если вход  $\bar{E}$  использовать как информационный вход, а информационные входы декодера  $S_1$  и  $S_0$  – как адресные входы. Условное обозначение демультиплексора показано на рис. 5.10.

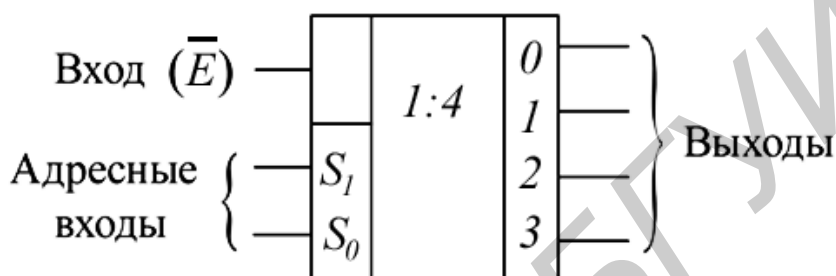


Рис. 5.10. Условное обозначение схемы рис. 5.9, используемой как демультиплексор

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

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

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

Используя двойную инверсию и закон де Моргана, преобразуем эти выражения:

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

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

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  $C_n = D_3 \cdot D_5 \cdot D_6 \cdot D_7$ .

Для реализации полного сумматора необходимо использовать декодер/демультиплексор 3:8 линий, например КР1533ИД7(SN74ALS138). Схема полного сумматора показана на рис. 5.11.

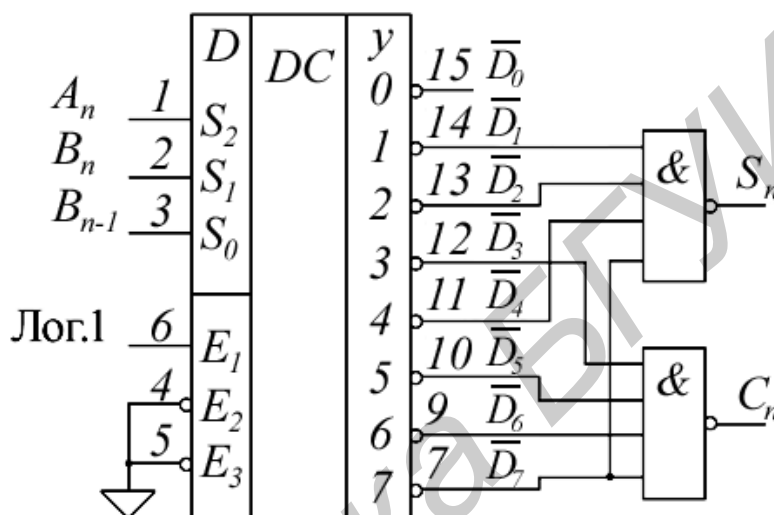


Рис. 5.11. Реализация полного сумматора с использованием декодера/демультиплексора

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 декодеры/демультиплексоры являются более экономичными при синтезе многовыходовых КС. В таких случаях один мультиплексор требуется для реализации каждого выхода, а при использовании декодера/демультиплексора – только дополнительные схемы И-НЕ.

### 5.15. 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. 5.12 и рис. 5.13.



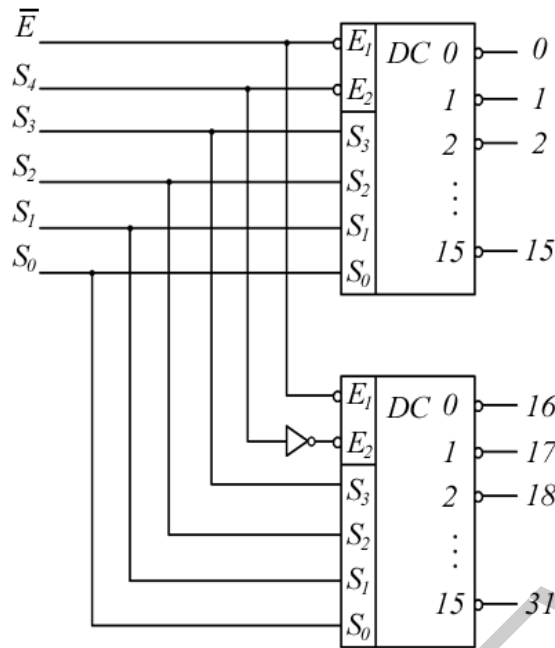


Рис. 5.12. Построение 5:32 линии декодера/демультиплексора, используя два 4:16 линии декодера/демультиплексора

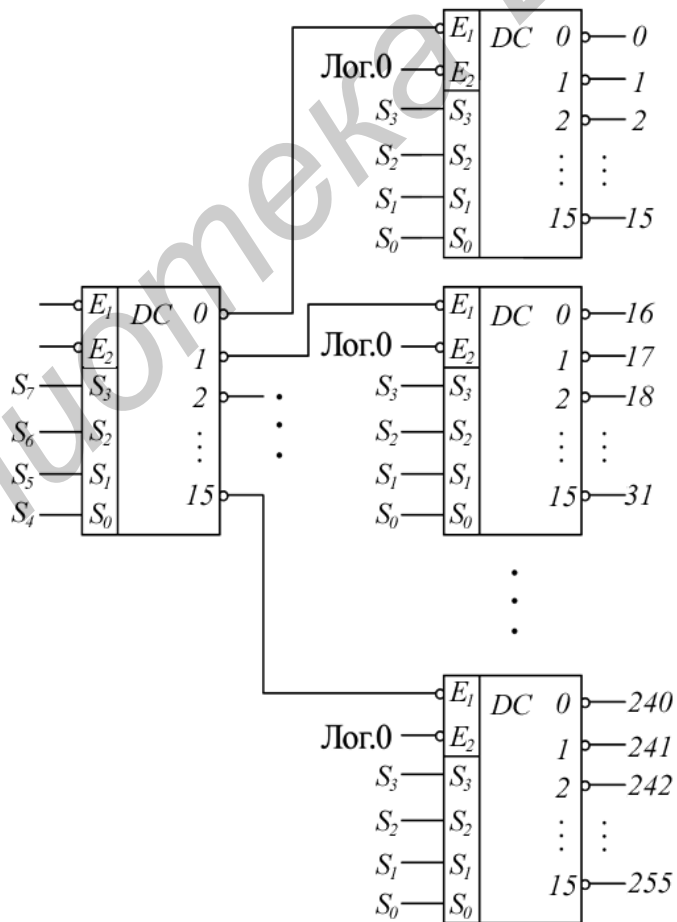


Рис. 5.13. Построение 8:256 линий декодера/демультиплексора, используя 17 декодеров/демультиплексоров 4:16 линий

## 5.2. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

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

### 5.2.1. 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.

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

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

### 5.2.3. 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  $D_0 - D_7$ .

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

1. Собрать схему полного сумматора (см. рис. 5.11).
2. Исследовать работу полного сумматора, результаты отразить в таблице истинности.

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

### 5.2.6. Синтезировать и исследовать работу компаратора для сравнения двухразрядных двоичных чисел.

### 5.3. СОДЕРЖАНИЕ ОТЧЕТА

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

### 5.4. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Дать определение мультиплексора и объяснить его работу.
2. Объяснить, каким образом мультиплексоры используются для синтеза КС.
3. Перечислить 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

#### Цель работы:

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

#### 6.1. 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 счетчиках 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 свое 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 *S* (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 остается в прежнем состоянии независимо от уровня синхросигнала.

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

### 6.1.1. 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

$t_{n+1}$			$t_{n+1}$	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	Неопределенность
1	1	1	x	Неопределенность

Если входы  $S_n$  и  $R_n$  оба в состоянии лог. 0, то триггер не изменяет свое состояние, т. е. триггер хранит 1 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  $R_n=0$ .

Комбинация входных сигналов  $R_n=S_n=1$  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 (см. табл. 6.1) может быть представлена картой Карно (рис. 6.1). Значения ФАЛ, показанные значком  $x$ , указывают на то, что данная ФАЛ является недоопределенной. При минимизации ФАЛ можно ее доопределить так, как нам это удобно, чтобы получить минимальную форму (см. рис. 6.1).

		$S_n R_n$			
		00	01	11	10
$Q_n$	0	0	0	x	1
	1	1	0	x	1

		$S_n R_n$			
		00	01	11	10
$Q_n$	0	0	0	x	1
	1	1	0	x	1

$$Q_{n+1} = S_n + Q_n \bar{R}_n$$

$$\bar{Q}_{n+1} = R_n + \bar{Q}_n \bar{S}_n$$

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:

$$\bar{Q}_{n+1} = \overline{S_n + \bar{Q}_n + R_n}; \quad (6.4)$$

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

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:

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

$$\bar{Q}_{n+1} = \overline{\overline{R_n \bar{Q}_n \bar{S}_n}}. \quad (6.7)$$

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. 6.2, а,б.

$$Q_{n+1} = \overline{\overline{R_n + Q_n + R_n}} \quad \bar{Q}_{n+1} = \overline{\overline{S_n + \bar{Q}_n + R_n}}$$

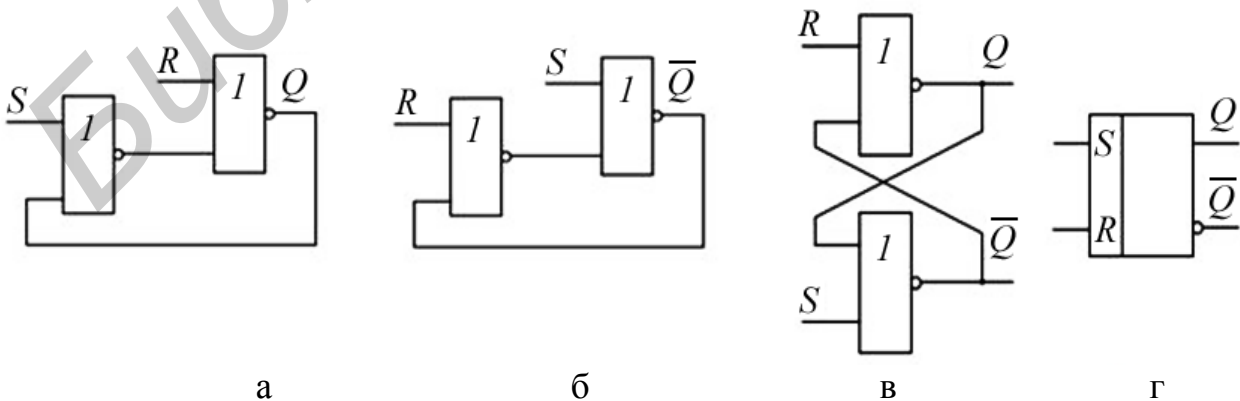


Рис. 6.2. RS-триггеры на элементах ИЛИ-НЕ

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

При анализе работы  $RS$ -триггера и при синтезе других триггеров используется таблица переходов  $RS$ -триггера (табл. 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 перешел в определенное 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 (см. рис. 6.2).

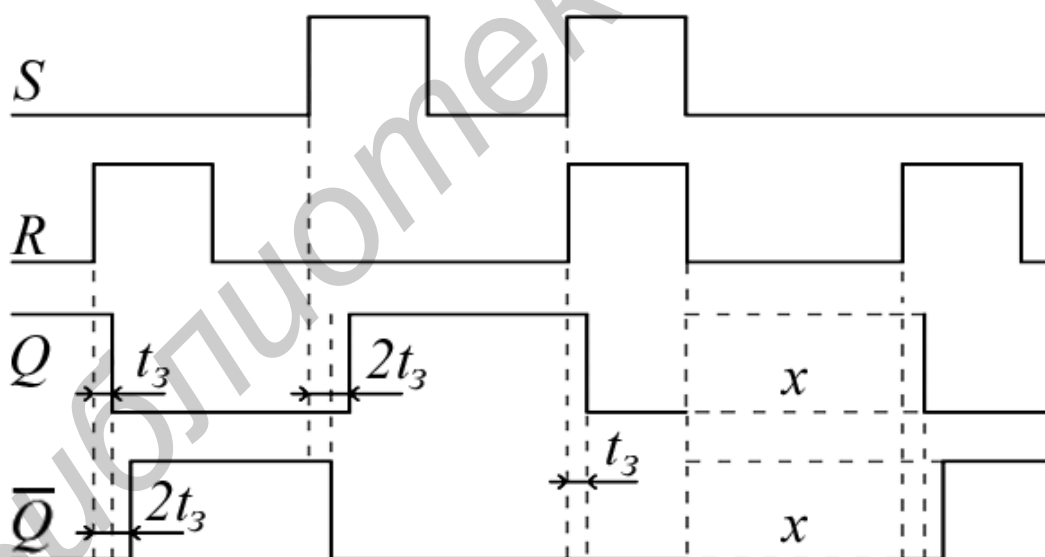


Рис. 6.3. Временные диаграммы работы асинхронного  $RS$ -триггера

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 поочередно и только после окончания переходных процессов в триггере, тогда максимальная частота переключения триггера будет равна



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

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

$$f_p = \frac{1}{2t_{3,сп}}. \quad (6.9)$$

### 6.1.2 Асинхронный $RS$ -триггер на элементах И-НЕ, $\overline{RS}$ -триггер

( $\overline{SR}$ -latch)

Воспользуемся выражениями (6.6) и (6.7) и реализуем  $RS$ -триггер на элементах И-НЕ (рис. 6.4):

$$Q_{n+1} = \overline{\overline{S_n} Q_n \overline{R_n}}; \quad \overline{Q}_{n+1} = \overline{\overline{R_n} Q_n \overline{S_n}}.$$

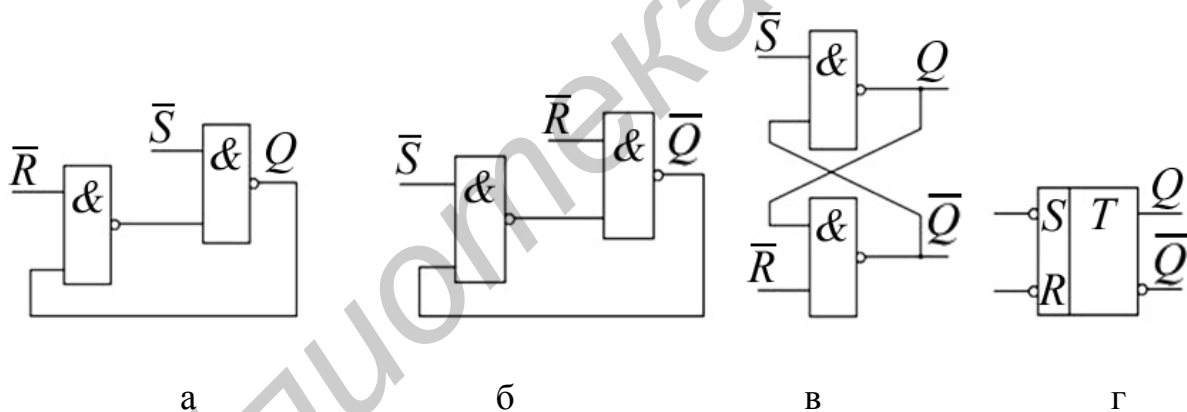


Рис. 6.4. Асинхронный  $RS$ -триггер:  
а, б – построение  $RS$ -триггера;  
в – изображение триггера на элементах И-НЕ;  
г – условное изображение  $\overline{RS}$ -триггера

$RS$ -триггер на элементах И-НЕ (см. 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.

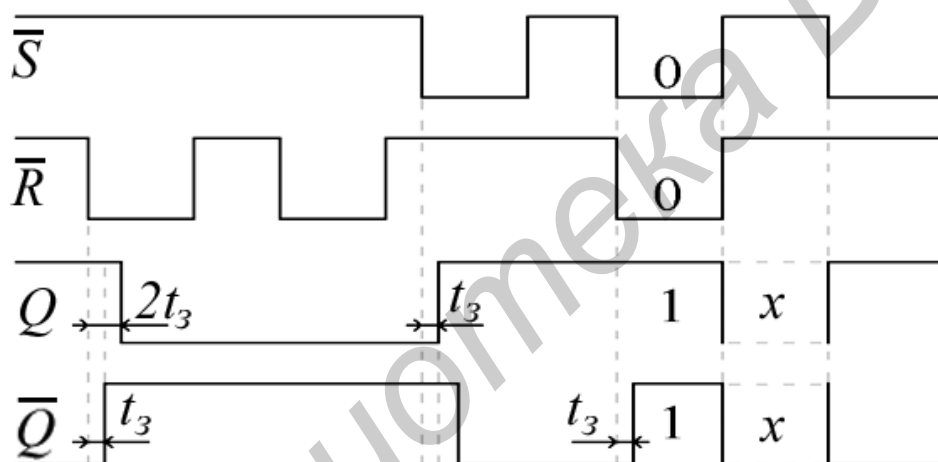
Таблица 6.3

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

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

Рис. 6.5. Временные диаграммы работы  $\overline{RS}$ -триггера

### 6.1.3. 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. 6.6, а.

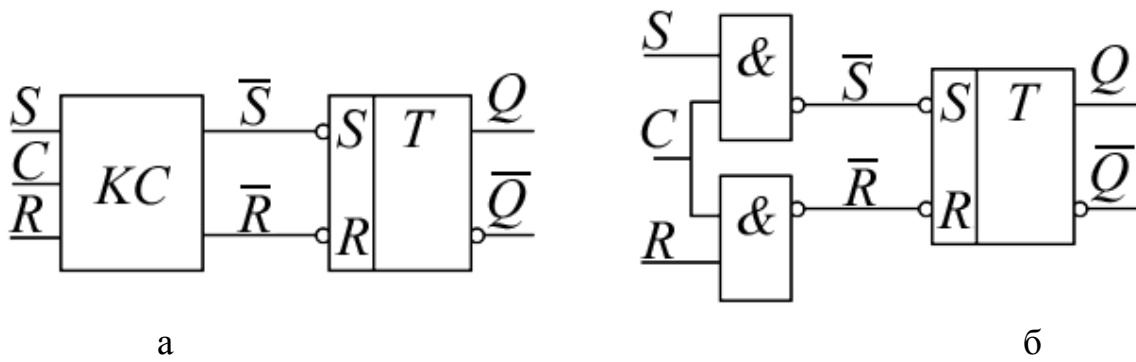


Рис. 6.6. Синхронный RS-триггер со статическим управлением

Задачей синтеза является определение функции  $\bar{S}_n$  и  $\bar{R}_n$  и построение  $KC$ , которая соответствующим образом управляет работой асинхронного  $\overline{RS}$ -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

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

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

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

$$\bar{R}_n = \bar{C}_n + \bar{R}_n = \bar{R}_n \bar{C}_n.$$

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

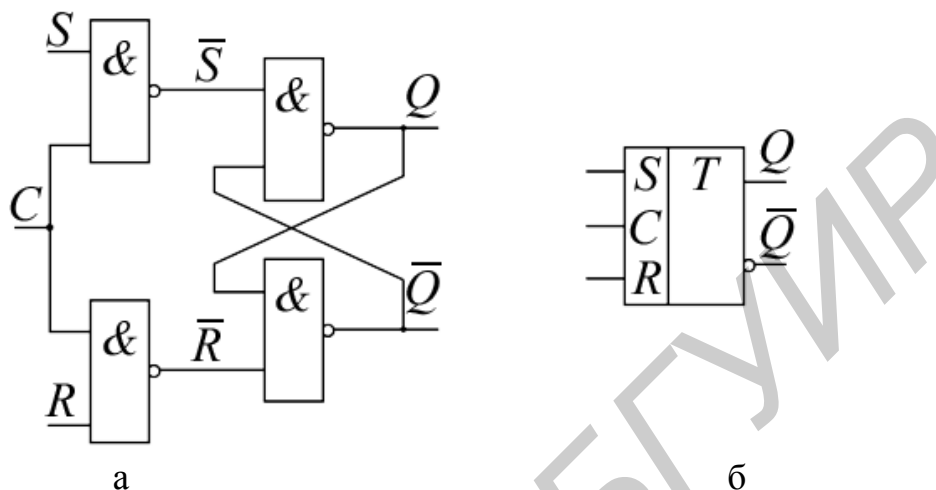


Рис. 6.7. Логическая схема и условное обозначение синхронного  $RS$ -триггера со статическим управлением

#### 6.1.4. $D$ -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  $\bar{R}\bar{S}$ -триггера. Блок-схема такого триггера показана на рис. 6.8, а.

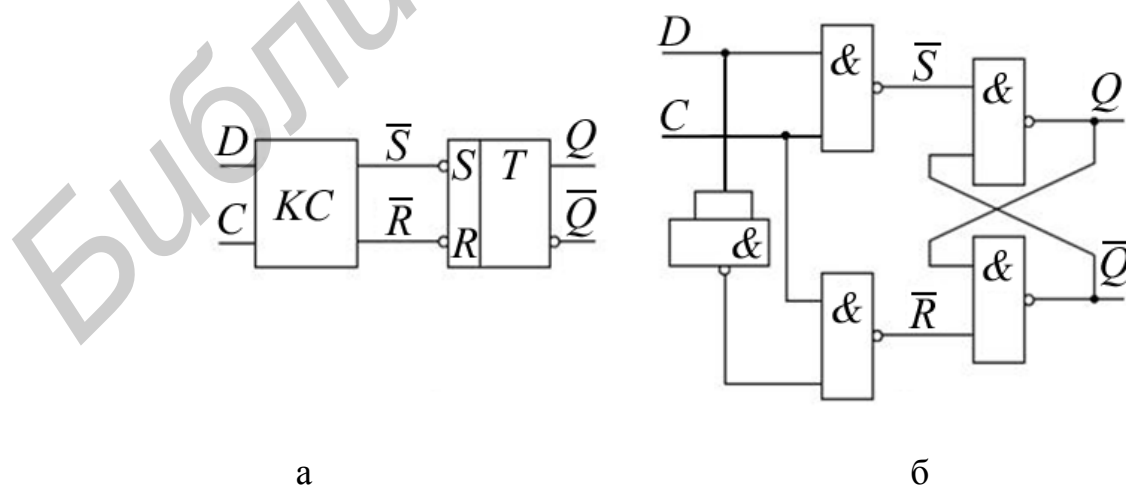


Рис. 6.8. Синхронный  $D$ -триггер со статическим управлением (окончание на с. 77)

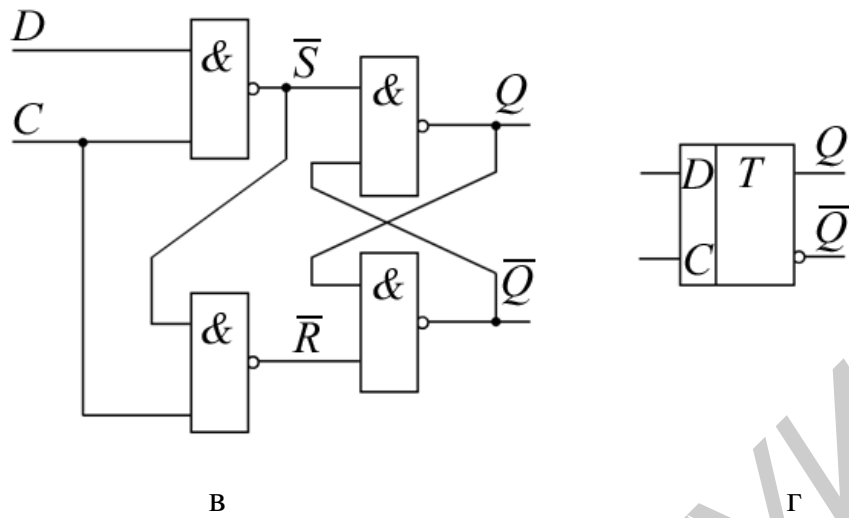
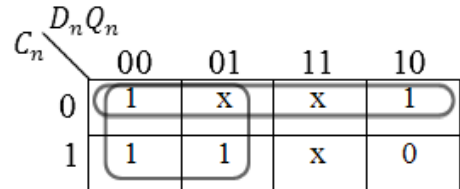


Рис. 6.8. Окончание (начало см. на с. 76)

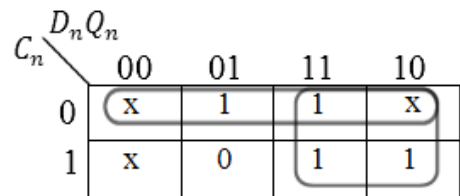
Из блок-схемы (см. рис. 6.8, а) очевидно, что для получения логической схемы синхронного  $D$ -триггера со статическим управлением необходимо определить комбинационную схему (КС), генерирующую функции  $\bar{S}$  и  $\bar{R}$  такими, чтобы базовый  $RS$ -триггер функционировал как  $D$ -триггер. Для 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



$$\bar{S}_n = \bar{C}_n + \bar{D}_n = \overline{D_n C_n};$$



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

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

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.

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

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

Рис. 6.9. Карта Карно для  $\bar{R}_n$

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

### 6.1.5. $JK$ -триггеры

$JK$ -триггер – это модернизированный  $RS$ -триггер, в котором неопределенное 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 приведена на рис. 6.10, а.

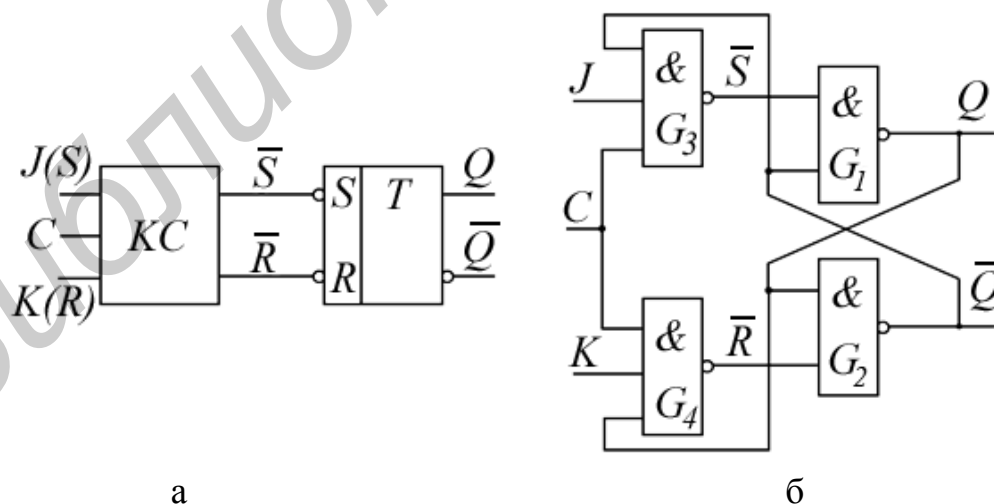


Рис. 6.10. Синхронный  $JK$ -триггер со статическим управлением

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

Таблица 6.7

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

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

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

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

$C_n$	$Q_n$	$J_n$	$K_n$	$Q_{n+1}$	$\bar{S}_n$	$\bar{R}_n$
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 приведена на рис. 6.10, б.

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, тогда на выходах логических элементов  $G_3$  и  $G_4$  уровни лог. 1 и асинхронный триггер на элементах  $G_1$  и  $G_2$  сохраняет свое 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, соединенный 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 изме-

дится на  $Q=1$ . Теперь  $J=K=1$  и  $Q=1$ , и после другого временного интервала  $\Delta t$  выход будет изменяться на  $Q=0$ . Следовательно, можно 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 ввиду 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).

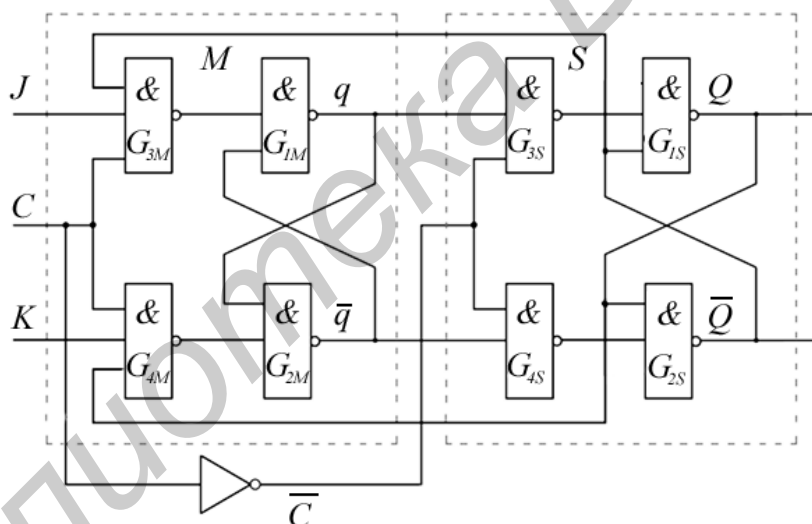


Рис. 6.11. Двухступенчатый  $JK$ -триггер с инвертором

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)

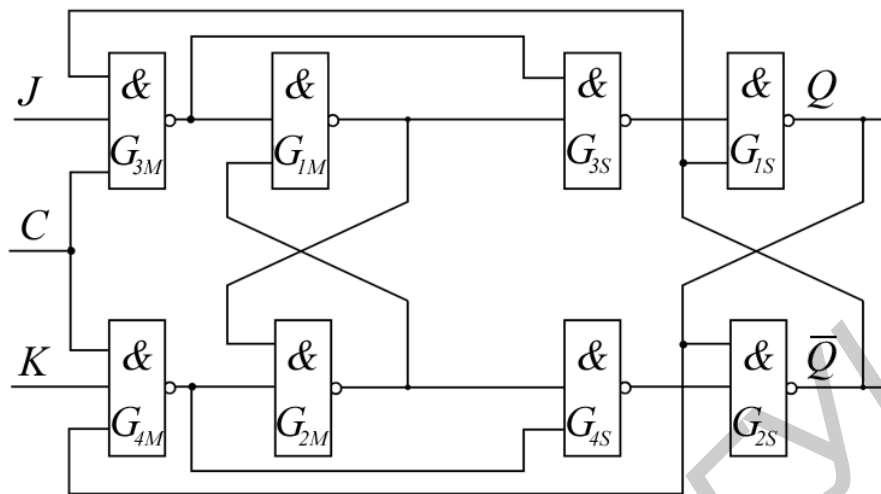


Рис 6.12. Двухступенчатый  $JK$ -триггер с запрещающими связями

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

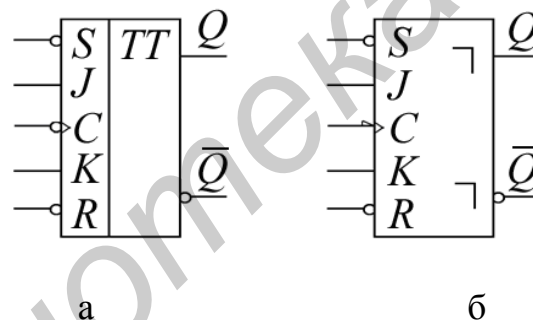


Рис. 6.13. Условное обозначение  $JK$ -триггера

### 6.1.6. $T$ -триггеры

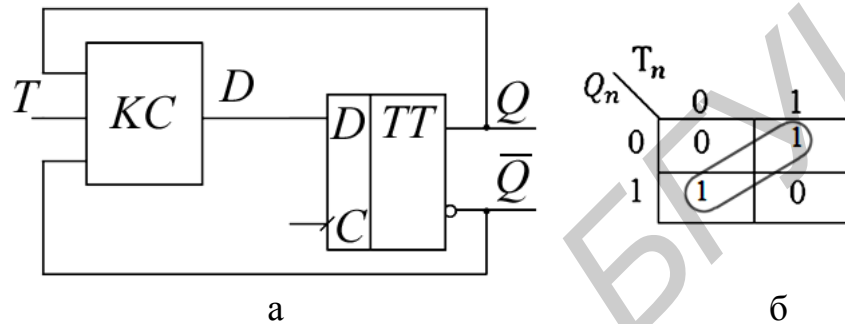
$T$ -триггер имеет один вход  $T$  (*trigger or toggle*).  $T$ -триггер изменяет свое состояние на противоположное с каждым импульсом синхронизации, если  $T=1$  и сохраняет свое состояние, если  $T=0$ .  $T$ -триггеры не производятся в интегральном исполнении, но могут быть легко построены из  $JK$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

Для того чтобы преобразовать  $D$ -триггер в  $T$ -триггер, воспользуемся таблицей переходов для обоих триггеров (табл. 6.8).

Таблица 6.8

Настоящее состояние	Следующее состояние	$D$ -триггер	$T$ -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  $T$ -триггера на основе  $D$ -триггера показана на рис. 6.14, а.



$$D_n = Q_n \oplus T_n$$

Рис. 6.14. Блок-схема преобразования  $D$ -триггера в  $T$ -триггер

Из блок-схемы (см. рис. 6.14) очевидно, что необходимо синтезировать такую КС, чтобы  $D$ -триггер функционировал как  $T$ -триггер. ФАЛ, описывающая 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. 6.14, б).

В соответствии с полученным выражением

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

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

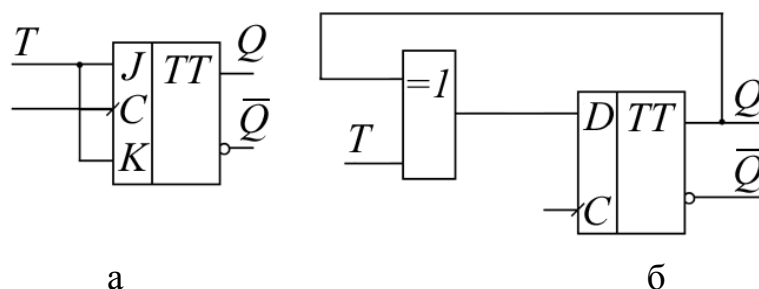


Рис. 6.15.  $T$ -триггер на основе  $J$ - $K$ -триггера и на основе  $D$ -триггера

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

Если переменная  $T_n$  в уравнении (6.10) равна 1, то уравнение может быть переписано следующим образом:

$$D_n = Q_n \oplus 1 = \bar{Q}_n. \quad (6.11)$$

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

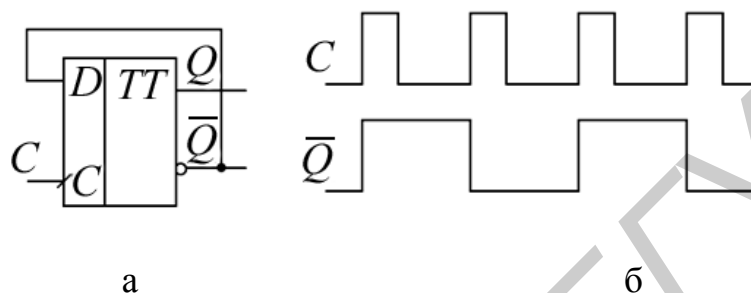


Рис. 6.16. Схема делителя на 2 и его временные диаграммы

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

### 6.1.7. 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 триггер 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  $C$  называется прямым динамическим, если отрицательным – то инверсным динамическим.

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

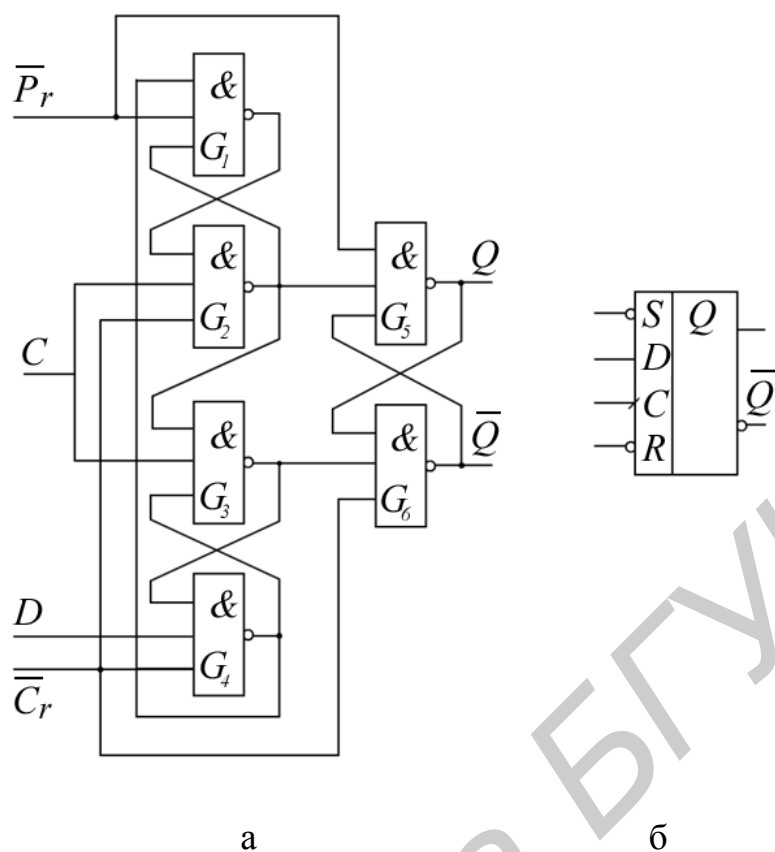


Рис. 6.17. Логическая схема и условное обозначение синхронного  $D$ -триггера с динамическим управлением (КР1533ТМ2)

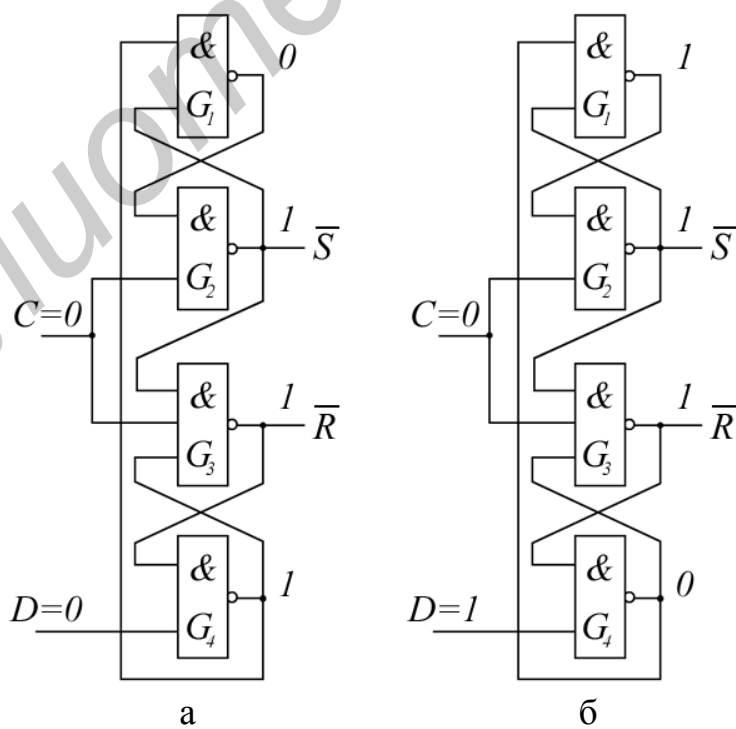


Рис. 6.18. Работа синхронного  $D$ -триггера с динамическим управлением (окончание на с. 85)

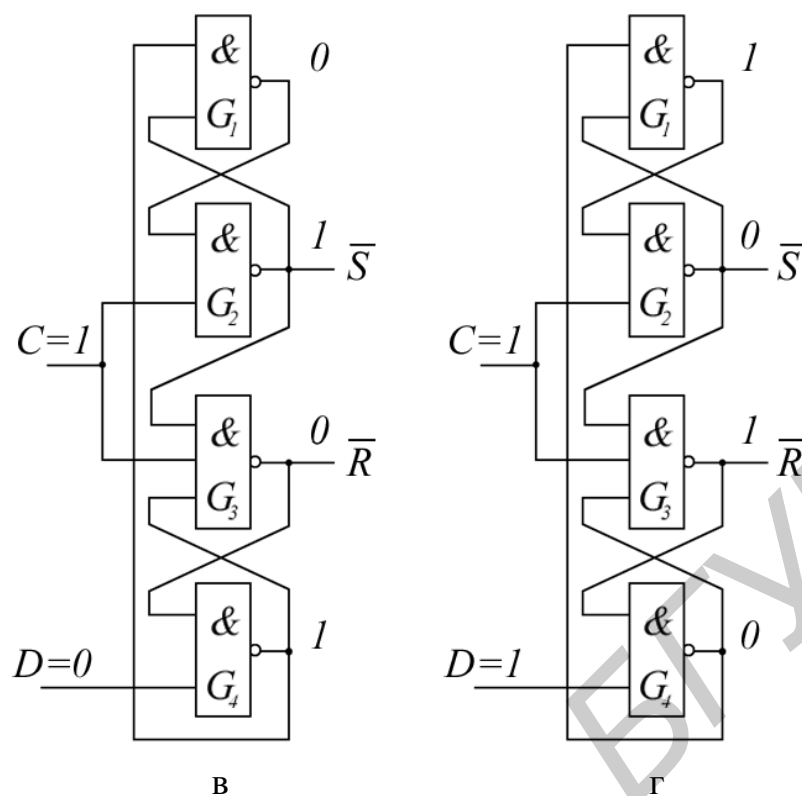


Рис. 6.18. Окончание (начало см. на с. 84)

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  $\overline{P_r}$  и  $\overline{C_r}$  поданы логические единицы, поэтому для простоты на рис. 6.18 они опущены. Выходы логических элементов  $G_2$  и  $G_3$  являются управляющими  $\overline{R}$ - и  $\overline{S}$ - входами асинхронного  $\overline{RS}$ -триггера. На рис. 6.18, а, б показаны значения на выходах  $G_1$ – $G_4$ , когда  $CK=0$ . На вход  $D$  может подаваться лог. 0 или лог. 1. В любом случае  $CK=0$  и на выходах  $G_2$  и  $G_3$  лог. 1, т. е.  $\overline{R} = \overline{S} = 1$  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  $CK=0$ , 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  $\bar{S}$  остается в состоянии лог. 1, а вход  $\bar{R}$  переходит в состояние лог. 0. Этот лог. 0 устанавливает триггер в состояние  $Q=0$  ( $\bar{Q}=1$ ), 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 тогда, когда синхровход возвращается в состояние лог. 0, однако теперь оба входа  $\bar{S}$  и  $\bar{R}$  устанавливаются в состояние  $\bar{R}=\bar{S}=1$ , 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 которых 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 что 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.

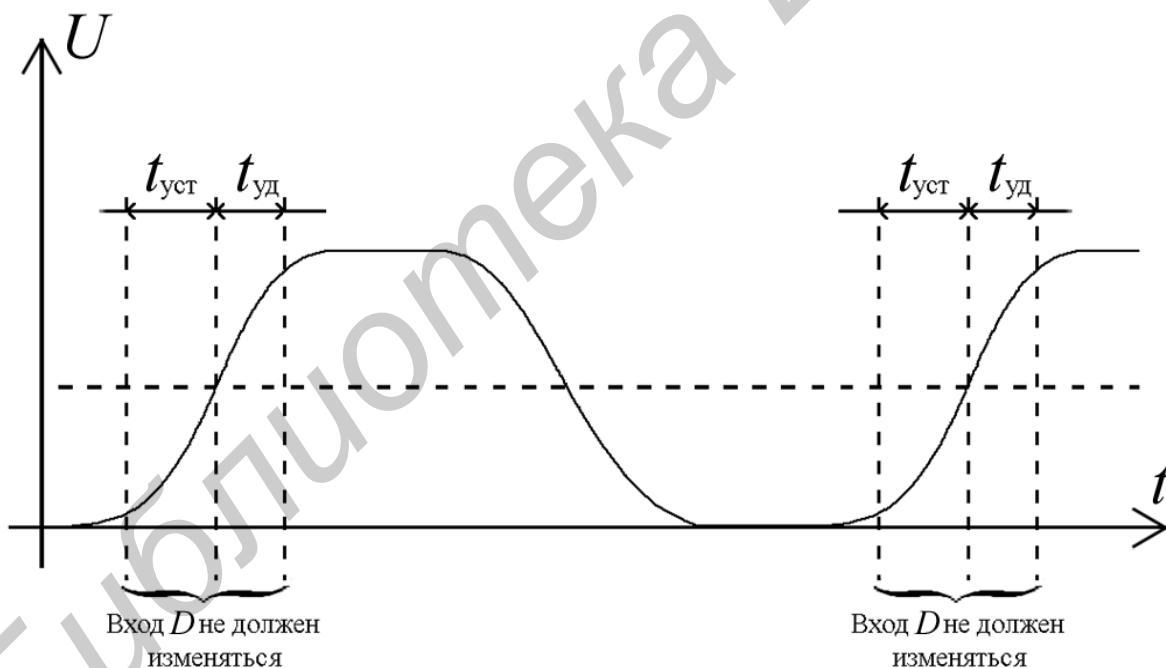


Рис. 6.19. Типичная форма синхроимпульсов для иллюстрации  $t_{уст}$  и  $t_{уд}$

Если  $D=1$ , когда  $CK=1$ , 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  $CK=1$ , не изменяет логические уровни  $\bar{S}$  и  $\bar{R}$  потому, что на выходе  $G_1$  сохраняется лог. 1. Когда синхроимпульс заканчивается и на входе  $CK$  устанавливается лог. 0,  $\bar{R}=\bar{S}=1$ , что сохраняет состояние  $D$ -триггера.

Таким образом, когда на входе  $C$  происходит положительный перепад, значения входа  $D$  передается на выход  $Q$ . Изменения на входе  $D$ , когда  $CK=1$  или когда происходит отрицательный перепад синхросигнала, или когда  $CK=0$ , не приводят к изменению состояния  $D$ -триггера.

### 6.1.8. 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.

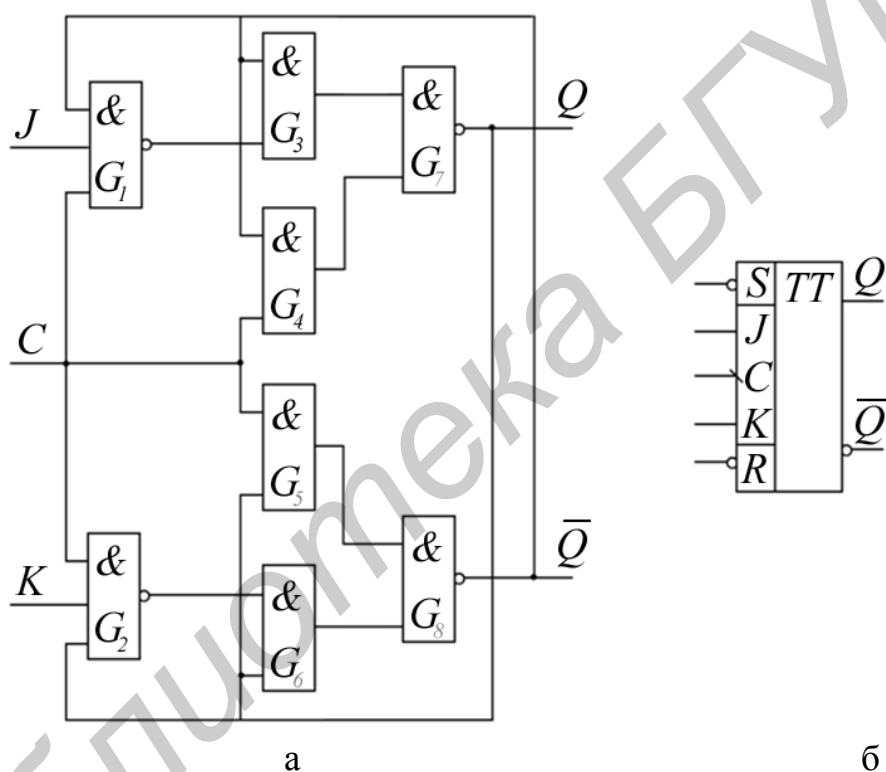


Рис. 6.20. Логическая схема и условное обозначение JK-триггера с инверсным динамическим управлением

Основным достоинством схемы (рис. 6.20, а) 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  $C$  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 установится  $\overline{Q} = 1$ , поскольку на выходах элементов  $G_2, G_5, G_6$  – 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.  $\overline{Q} = 0$ . Лог. 0 с выхода элемента  $G_8$  поступает на вход элемента  $G_3$  прежде, чем на втором входе появится 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  $\overline{Q} = 0$ . 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.

## 6.2. 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  $\overline{RS}$ -триггера.

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

#### 6.2.1. Асинхронный $\overline{RS}$ -триггер на элементах ИЛИ-НЕ

1. Используя логические элементы ИЛИ-НЕ, собрать асинхронный  $\overline{RS}$ -триггер (см. рис. 6.2).



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

### 6.2.2 Асинхронный $\overline{RS}$ -триггер на элементах И-НЕ, $\overline{RS}$ -триггер

1. Используя логические элементы И-НЕ, собрать асинхронный  $\overline{RS}$ -триггер (см. рис. 6.4).

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

### 6.2.3. Синхронный $\overline{RS}$ -триггер со статическим управлением

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

### 6.2.4. Синхронный D-триггер со статическим управлением

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

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

### 6.2.5. JK-триггер M-S-конфигурации

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

### 6.2.6. 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. 6.21,а.

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

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

### 6.2.7. 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,  $\overline{S}$  и сброса  $\overline{R}$ . 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 (см. 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.)

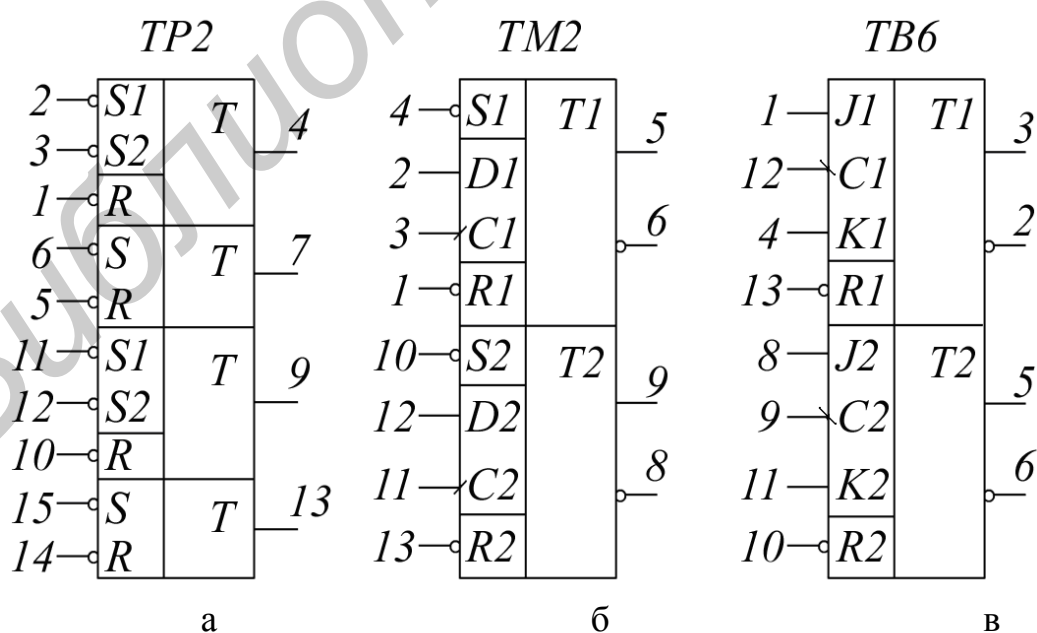


Рис. 6.21. Условные обозначения модульных триггеров

### 6.2.8. 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. 6.21, в.

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.

### 6.3. СОДЕРЖАНИЕ ОТЧЕТА

1. Цель работы.
2. Схемы, исследуемые в работе.
3. Таблицы и временные диаграммы работы исследуемых триггеров.
4. Выводы по результатам исследования.

### 6.4. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Дайте определение триггера.
2. Какие признаки используются при классификации триггеров?
3. Что такое таблица истинности или характеристическая таблица триггера?
4. Что такое таблица переходов триггера?
5. В чем состоит отличие синхронных триггеров и асинхронных?
6. Чем отличаются синхронные триггеры со статическим управлением от синхронных триггеров с динамическим управлением?
7. Почему не могут быть построены T- и JK-триггеры со статическим управлением?
8. Объяснить принцип действия двухступенчатого триггера (M-S триггера).
9. Преобразуйте RS-, D-, JK-триггер в T-триггер.
10. Пояснить работу D-триггера с динамическим управлением.
11. Пояснить работу JK-триггера с динамическим управлением.
12. В чем заключается различие и сходство RS-триггера и JK-триггера?

## Лабораторная работа №7

# РЕГИСТРЫ И ИХ ПРИМЕНЕНИЕ

*Цель работы:*

1. Изучение принципов построения регистров.
2. Исследование режимов работы и применения регистров.

### 7.1. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Триггер может хранить (запоминать) 1 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 с помощью триггеров ( $RS$ ,  $JK$ ,  $D$ ) 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 синхронимпульсов. 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**.

#### 7.1.1. 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 ис-

пользоваться 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 рис. 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.

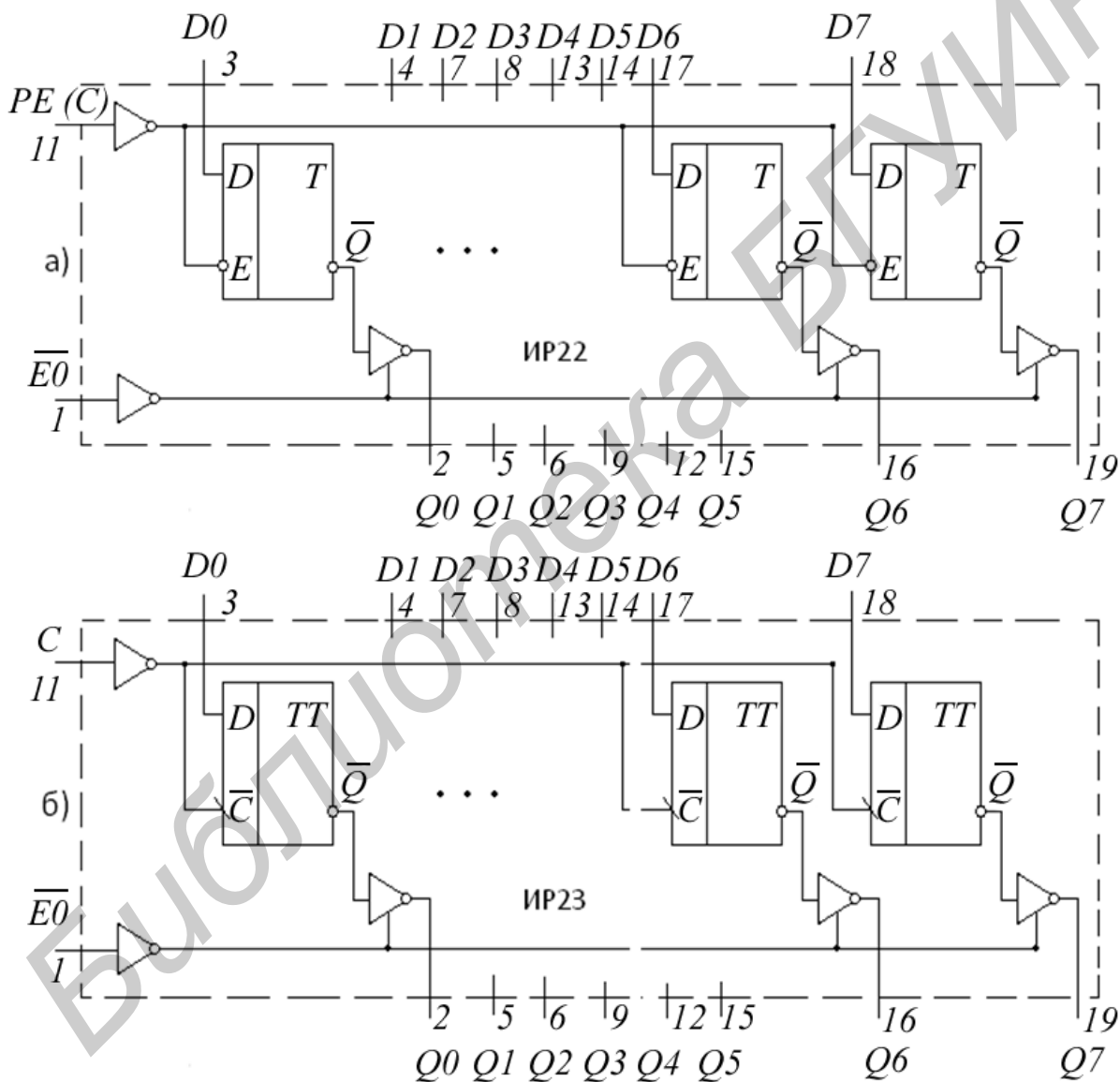


Рис. 7.1. Логическая структура регистров ИР22, ИР23

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  $C$ ). 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  $C$ .

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.

### 7.1.2. Микросхема ИР16

Это четырехразрядный регистр сдвига с третьим состоянием выхода. Логическая структура регистра приведена на рис. 7.2. Регистр 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  $C$ , 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  $\overline{C}$ .

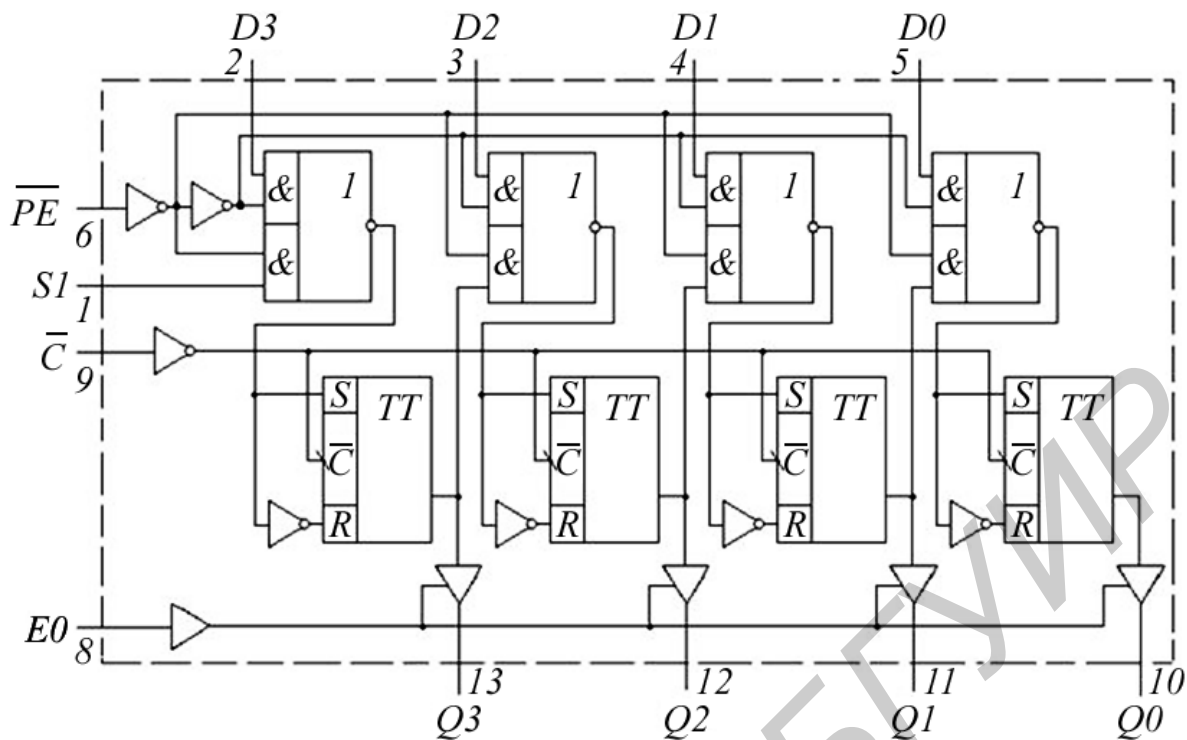


Рис. 7.2. Логическая структура регистра ИР16

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

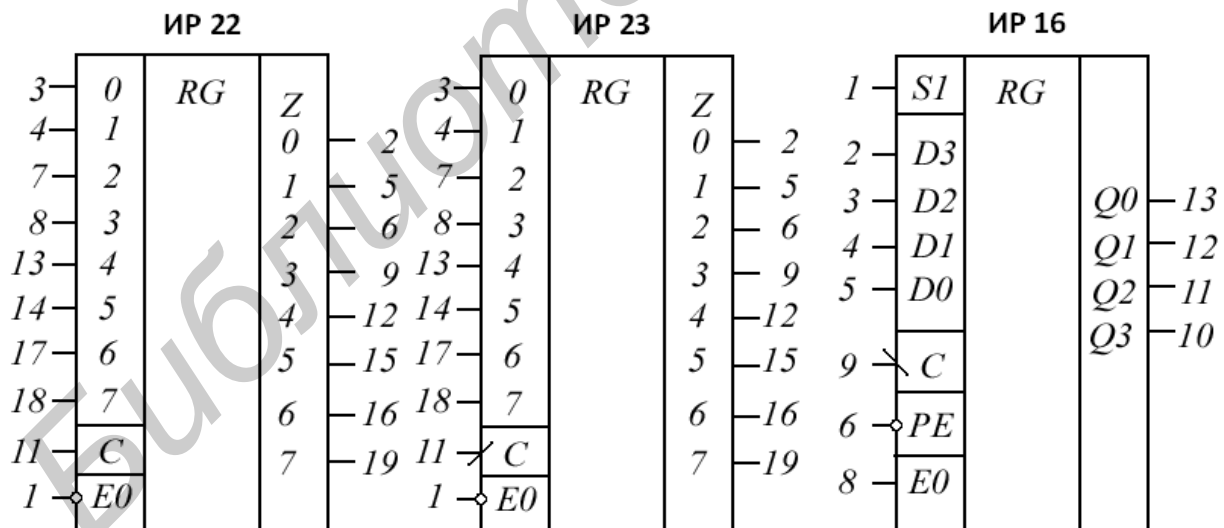


Рис. 7.3. Условные обозначения и цоколевка ИР22, ИР23 и ИР16

### 7.1.3. 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.

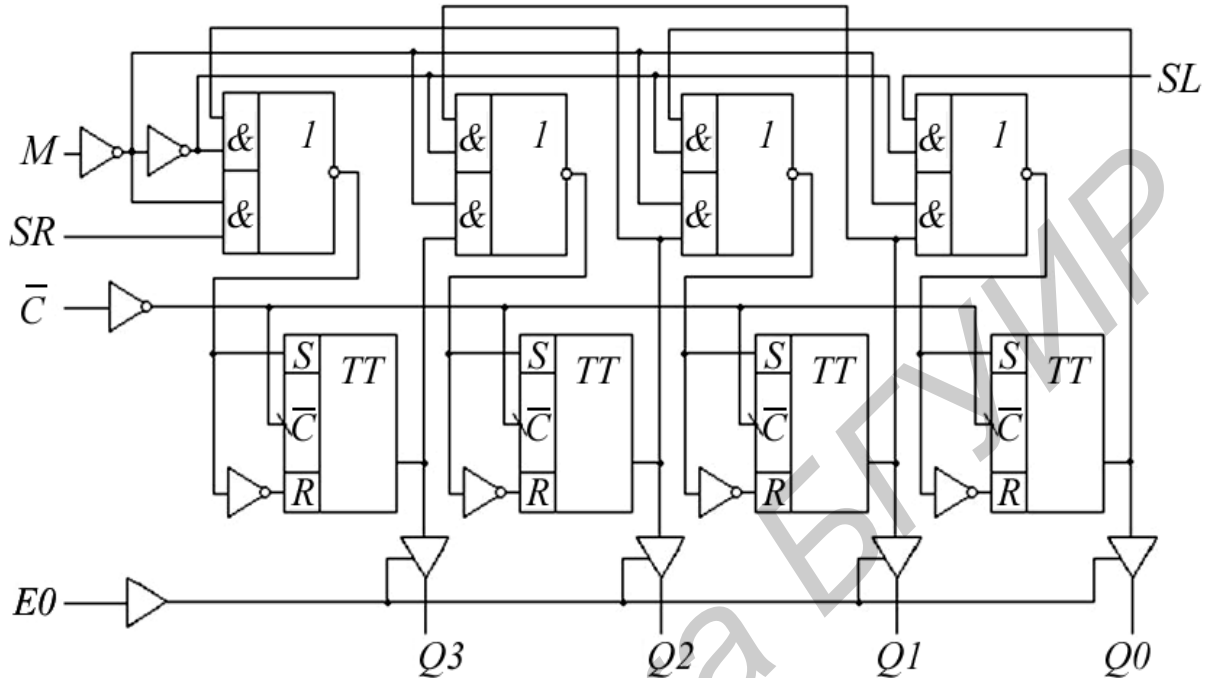


Рис. 7.4. Логическая структура реверсивного регистра

Если на вход  $M$  поступает низкий уровень напряжения, то при подаче тактовых импульсов, информация с входа  $SR$  сдвигается вправо. А если на вход  $M$  поступает высокий уровень напряжения, то при подаче тактовых импульсов информация с входа  $SL$  сдвигается влево.

#### *Применение регистров сдвига*

Изначально регистры сдвига предназначались для временного хранения данных и некоторых манипуляций с этими данными. Рассмотрим некоторые наиболее общие применения регистров сдвига.

#### *Линия задержки*

Регистр сдвига с последовательным вводом и последовательным выводом данных можно использовать для задержки цифровых сигналов на время

$$\Delta t = \frac{N \times 1}{f_c},$$

где  $N$  – число разрядов регистра сдвига;

$f_c$  – частота следования импульсов синхронизации.

Таким образом, цифровой сигнал появляется на выходе регистра сдвига с задержкой на время  $\Delta t$ . Время задержки можно варьировать с помощью



изменения частоты следования импульсов синхронизации и числа триггеров регистра сдвига.

### *Преобразователь информации из последовательного вида в параллельный вид*

Данные в последовательном виде легко преобразовать в параллельную форму с помощью последовательно-параллельного регистра.

### *Преобразователь информации из параллельного вида в последовательный вид*

Данные в параллельном виде легко преобразовать в последовательную форму с помощью параллельно-последовательного регистра.

### *Кольцевой счетчик*

Если последовательный выход регистра сдвига  $Q_0$  соединить с последовательным входом, то единичный бит, записанный в один из триггеров, будет циркулировать по регистру при подаче синхроимпульсов. Такая схема называется **кольцевым счетчиком**. На выходах триггеров генерируется неперекрывающиеся 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  $\overline{Q_0}$  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  $\text{mod} = 2N$ .

## **7.2. 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.

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

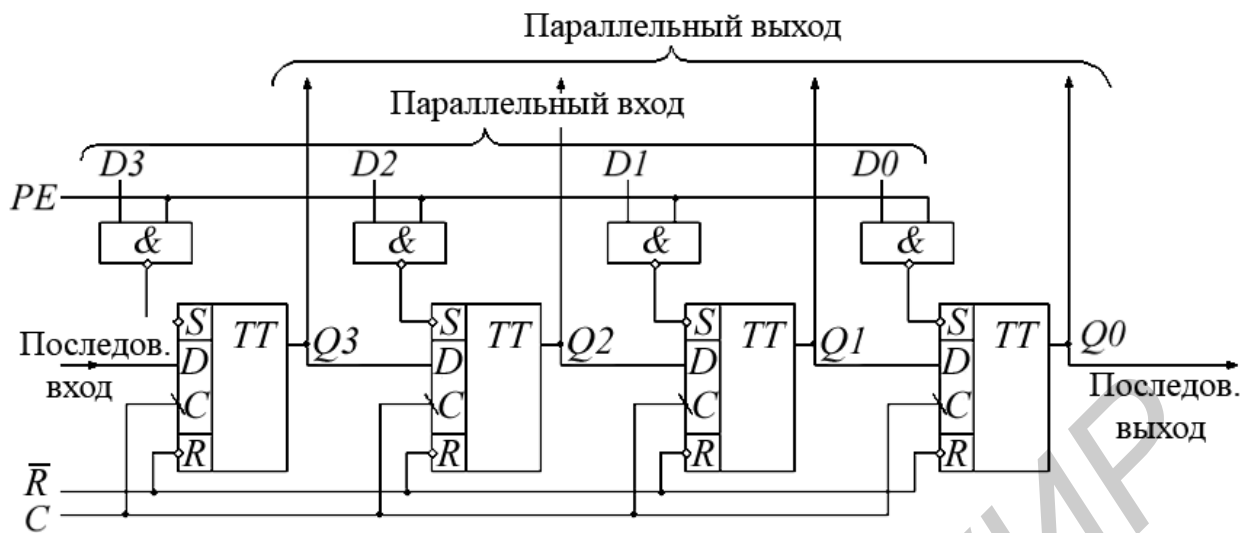


Рис. 7.5. Четырехразрядный регистр сдвига

**7.2.2. Исследовать функционирование регистра сдвига (рис. 7.5). Эта схема может работать во всех четырех режимах**

7.2.2.1. Последовательный вход, последовательный/параллельный выход. Исследовать работу регистра, подавая на его 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 .

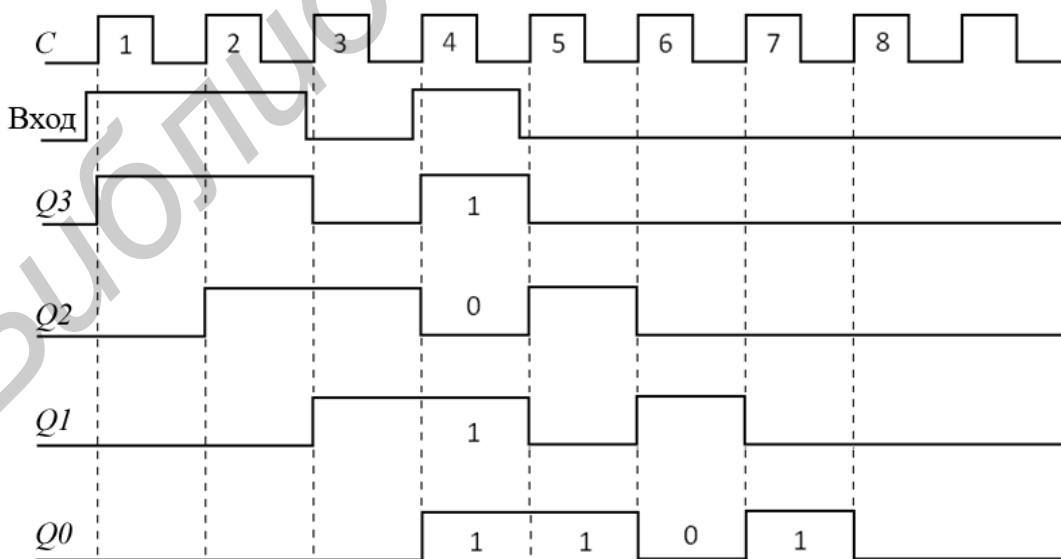


Рис. 7.6. Временные диаграммы работы регистра: последовательный вход, последовательный/параллельный выход

7.2.2.2. 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).

7.2.2.3. Повторить пп. 7.2.2.1 и 7.2.2.2 для входных данных 1101. Построить временные диаграммы.

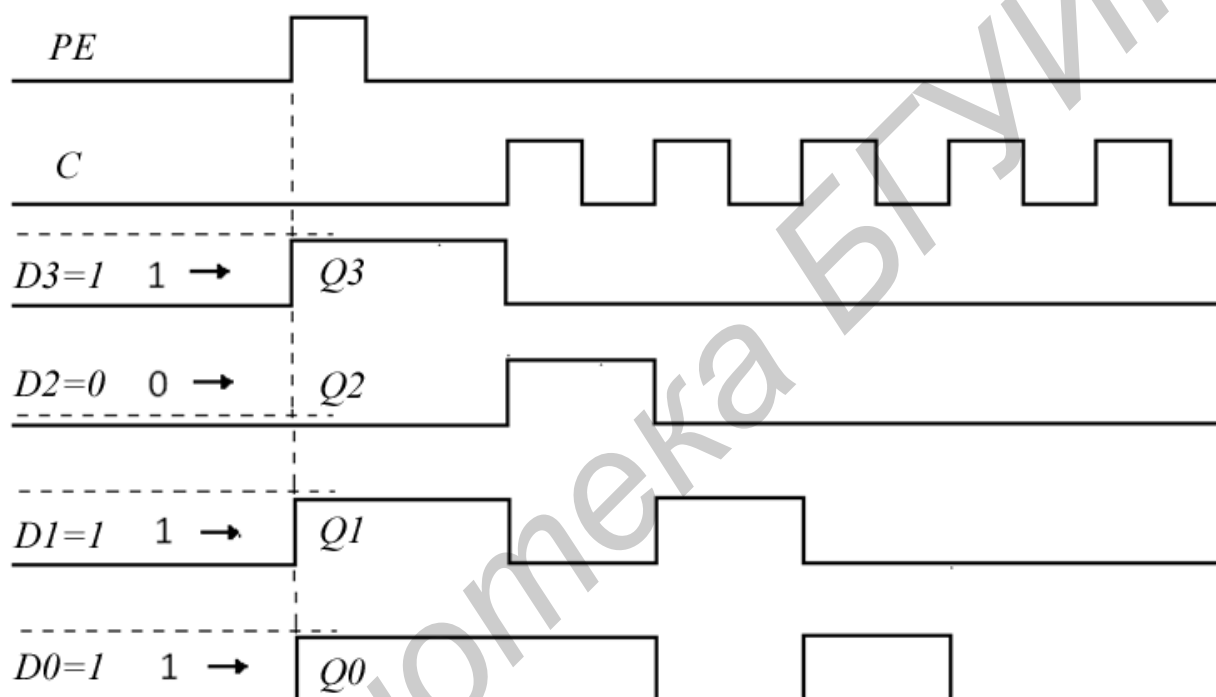


Рис. 7.7. Временные диаграммы, поясняющие работу регистра в режиме: последовательный вход, последовательный/параллельный выход

### 7.2.3. Исследование кольцевого счетчика

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

7.2.3.2. Для исследования кольцевого счетчика вначале обнулить регистр, затем записать в него 0001 в параллельном виде и подать синхроимпульсы. Результаты наблюдения табулировать (табл. 7.1).

Таблица 7.1

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

7.2.3.3. Построить временные диаграммы работы кольцевого счетчика, сделать выводы.

#### 7.2.4. Исследование счетчика Джонсона

7.2.4.1. На основе регистра сдвига (см.рис. 7.5) построить счетчик Джонсона. Для построения счетчика Джонсона необходимо соединить инверсный выход  $\bar{Q}_0$  с последовательным входом регистра.

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

7.2.4.3. Построить временные диаграммы работы счетчика Джонсона, сделать выводы.

### 7.3. СОДЕРЖАНИЕ ОТЧЕТА

1. Цель работы.
2. Схемы исследуемых в работе устройств.
3. Таблицы и временные диаграммы, отражающие результаты исследований.
4. Выводы по результатам исследований.

### 7.4. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Объяснить работу регистров во всех четырех режимах работы.
2. Как строится реверсивный регистр сдвига?
3. Как строится кольцевой счетчик?
4. Чему равен модуль счета кольцевого счетчика?
5. Как строится счетчик Джонсона?
6. Чему равен модуль счета счетчика Джонсона?

## Лабораторная работа №8

# ГЕНЕРАТОРЫ ПОСЛЕДОВАТЕЛЬНОСТЕЙ

Цель работы:

1. Изучение синтеза генераторов последовательностей.
2. Исследование работы генераторов последовательностей.

### 8.1. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

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

- счетчики;
- генераторы псевдослучайных последовательностей;
- генераторы заданной последовательности и заданного периода;
- генераторы кодов.

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

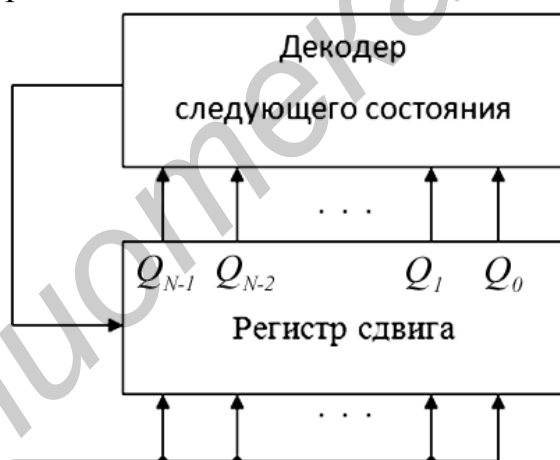


Рис. 8.1. Блок-схема генератора последовательностей

Выход декодера следующего состояния  $Y$  – это функция от  $Q_{N-1}, Q_{N-2}, \dots, Q_0$ , 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 счетчик и счетчик Джонсона являются частными случаями генераторов последовательностей.

Синтез генераторов последовательностей рассмотрим на примерах.

**Пример 8.1.** 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 табл. 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 табл. 8.1 видно, что не все состояния регистра отличаются от других (первая и вторая строки одинаковы, а также четвертая и шестая). Это означает, что 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 (табл. 8.3).

Таблица 8.3

		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

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

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

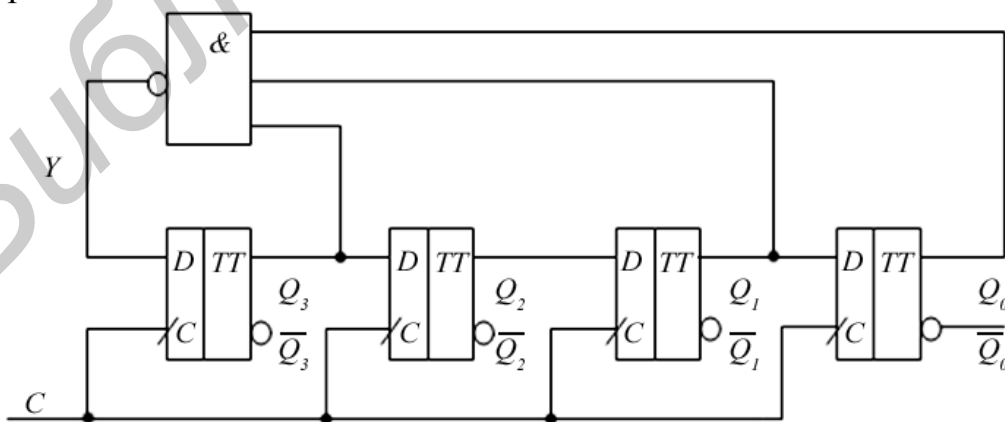


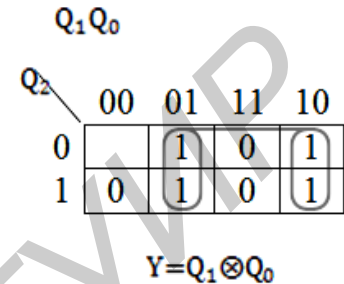
Рис. 8.2. Логическая структура генератора последовательности ... 1101011 ...

**Пример 8.2.** Синтезировать генератор последовательности ... 1101001...

*Решение.* Минимально возможное число триггеров регистра  $N=3$ . Проверим, является ли это число достаточным. Для этого построим табл. 8.4.

Таблица 8.4

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 табл. 8.4 видно, что все состояния регистра являются неодинаковыми, т. е. число триггеров  $N=3$  является достаточным для реализации генератора. 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 рис. 8.3 показана упрощенная схема генератора последовательности ... 1101001...

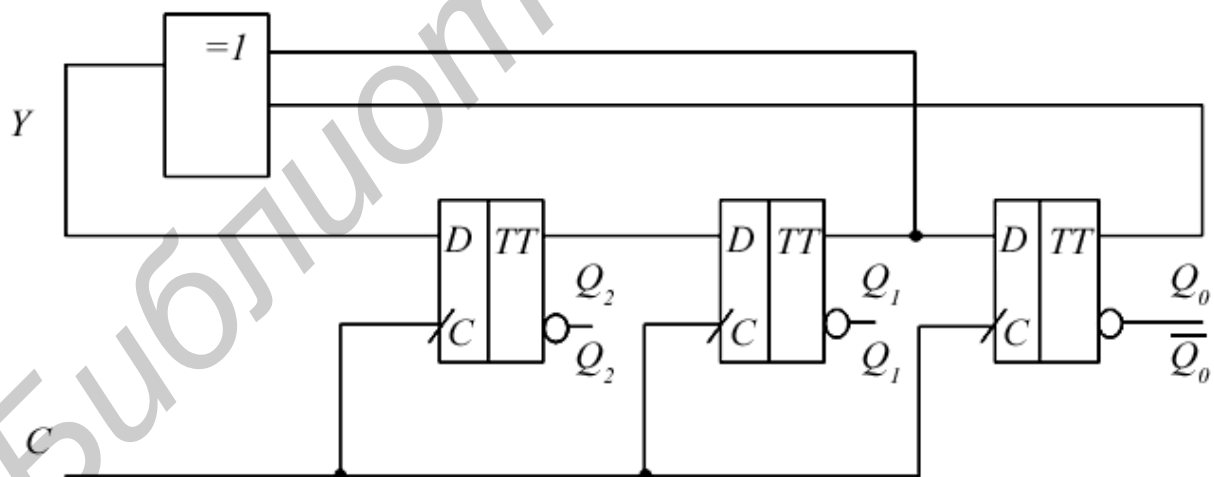


Рис. 8.3. Логическая структура генератора последовательности ... 1101001

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



Генераторы, которые генерируют последовательности длиной

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

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

## 8.2. 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.

### 8.2.1. Исследование работы генератора последовательности (см. 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 виде таблицы.

### 8.2.2. Исследование работы генератора последовательности (см. рис. 8.3)

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

### 8.2.3. Синтез и исследование генератора последовательности ...1001011...

1. Синтезировать и построить генератор последовательности ...1001011....
2. Исследовать работу генератора, результаты представить в табличной форме.
3. Результаты предоставить в виде таблицы.

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

### **8.3. СОДЕРЖАНИЕ ОТЧЕТА**

1. Цель работы.
2. Схемы генераторов исследуемых в работе.
3. Таблицы результатов наблюдений.
4. Выводы.

### **8.4. КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. Чему равняется минимально возможное число триггеров для генератора последовательности длиной  $S=27$ ?
2. Какие генераторы называются генераторами последовательности максимальной длины?
3. Привести примеры генераторов последовательности максимальной длины.
4. Объяснить работу генераторов, исследованных в работе.

## ИССЛЕДОВАНИЕ АСИНХРОННЫХ СЧЕТЧИКОВ

*Цель работы:*

1. Изучение методов построения основных типов асинхронных счетчиков.
2. Исследование функционирования основных типов асинхронных счетчиков.

### 9.1. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Цифровой счетчик – это группа триггеров, соединенных так, чтобы считать число импульсов, поданных на вход, и фиксировать число подсчитанных импульсов в том или ином коде.

Основными характеристиками счетчика являются коэффициент (модуль) счета и быстродействие.

Коэффициент или модуль счета характеризует число устойчивых состояний счетчика.

Быстродействие счетчика зависит от используемой элементной базы и схемы построения.

Счетчики классифицируются по ряду признаков:

- быстродействие и способ организации внутренних связей: асинхронные и синхронные;
- направление счета: суммирующие, вычитающие, реверсивные;
- модуль счета: двоичные, двоично-десятичные или с другим модулем счета.

Классификационные признаки независимы и могут встречаться в разных сочетаниях.

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

#### 9.1.1. Асинхронные счетчики

Чтобы изучить асинхронные счетчики, рассмотрим счетную последовательность (табл. 9.1). Число состояний в такой последовательности 8, что требует использования трех триггеров ( $2^3=8$ ).

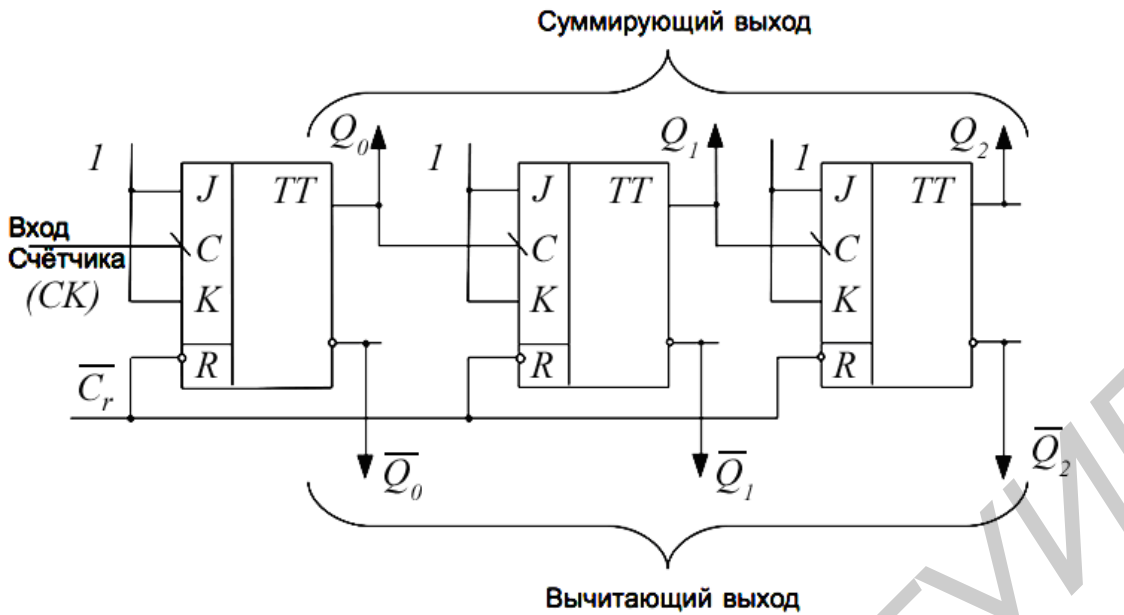
Таблица 9.1

Счет	Состояние счетчика		
	$Q_2$	$Q_1$	$Q_0$
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

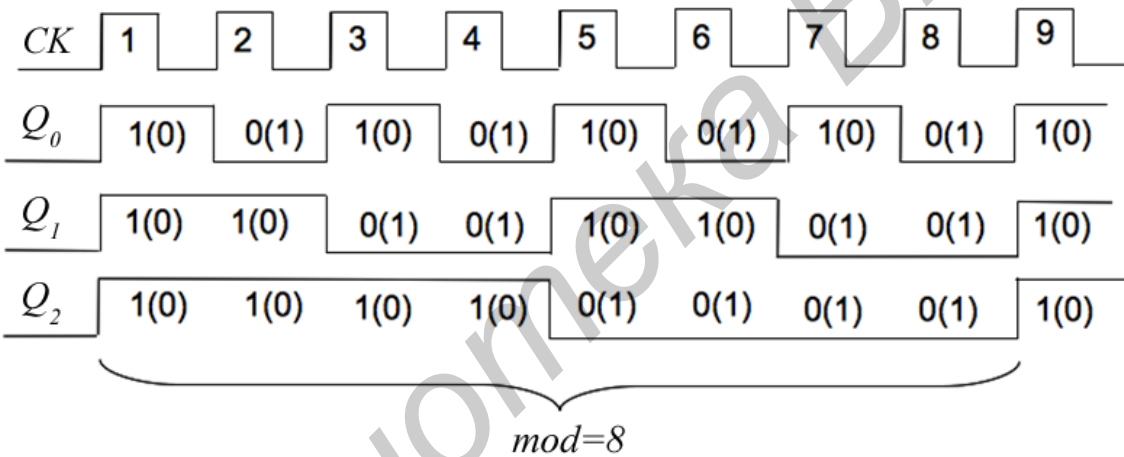
Выход  $Q_0$  наименее значащего триггера (разряда) изменяется каждым счетным импульсом. Это может быть реализовано путем использования триггера  $T$ -типа при  $T=1$ . Состояние на выходе  $Q_1$  меняется всякий раз, когда выход  $Q_0$  меняется из 1 в 0. Поэтому если выход  $Q_0$  соединить с входом синхронизации следующего  $T$ -триггера с инверсным динамическим входом, то выход  $Q_1$  будет менять состояние всякий раз, когда выход  $Q_0$  осуществляет переход из 1 в 0 (отрицательный перепад синхроимпульса). Таким же образом переход из 1 в 0 с выхода  $Q_1$  изменяет состояние  $Q_2$ , что достигается путем соединения  $Q_1$  с синхровходом следующего триггера. Аналогичным образом может быть построен асинхронный счетчик с большим числом разрядов или модулем счета.

На рис. 9.1, а приведена схема асинхронного счетчика с использованием триггеров с инверсным динамическим входом ( $TB6$ ). Входы  $J$  и  $K$  соединены вместе, образуя вход  $T$ -триггера, и на них подана лог. 1.

На рис. 9.1, б приведены временные диаграммы, поясняющие работу счетчика. На прямых входах триггеров ( $Q_2, Q_1, Q_0$ ) отражается состояние суммирующего счетчика, когда с каждым счетным импульсом состояние счетчика увеличивается. В то же самое время этот счетчик можно рассматривать как вычитающий, если информацию о состоянии счетчика снимать с инверсных выходов триггеров ( $Q_2, Q_1, Q_0$ ).



а



б

Рис. 9.1. Асинхронный счетчик на триггерах с инверсным динамическим синхровходом

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

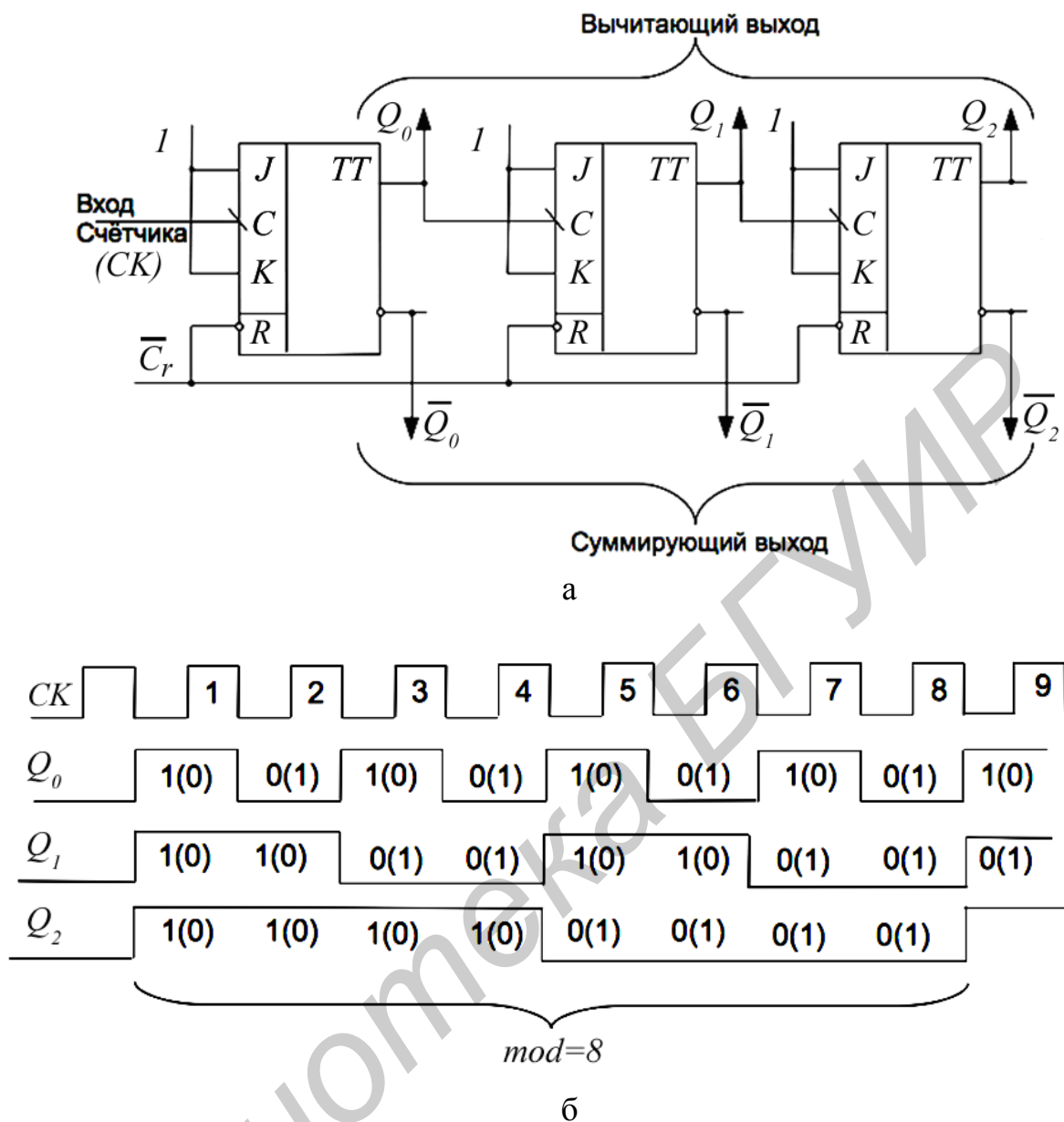


Рис. 9.2. Асинхронный счетчик на триггерах с инверсным динамическим синхровходом

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

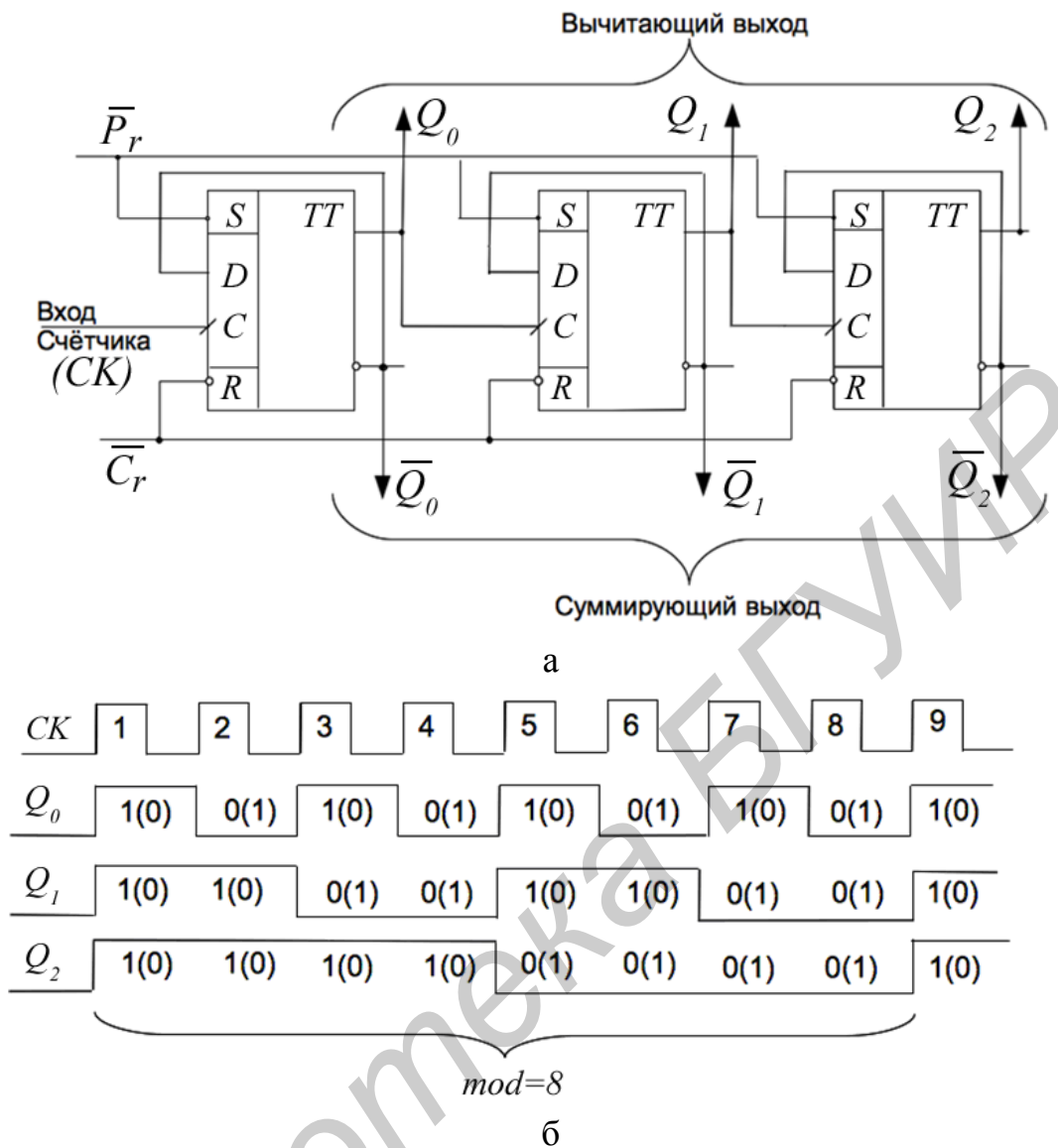


Рис. 9.3. Асинхронный счетчик на триггерах с прямым динамическим синхровходом

Асинхронные счетчики, построенные на двухступенчатых триггерах *M-S*-типа, которые перебрасываются отрицательным перепадом синхроимпульса, работают аналогично счетчикам на триггерах с инверсным динамическим синхровходом.

### 9.1.2. Реверсивные счетчики

Счетчики способные работать как в прямом, так и обратном направлении, называются реверсивными. При построении реверсивного счетчика изменение направления счета достигается переключением межразрядных связей. На рис. 9.4 приведен пример построения асинхронного реверсивного счетчика.

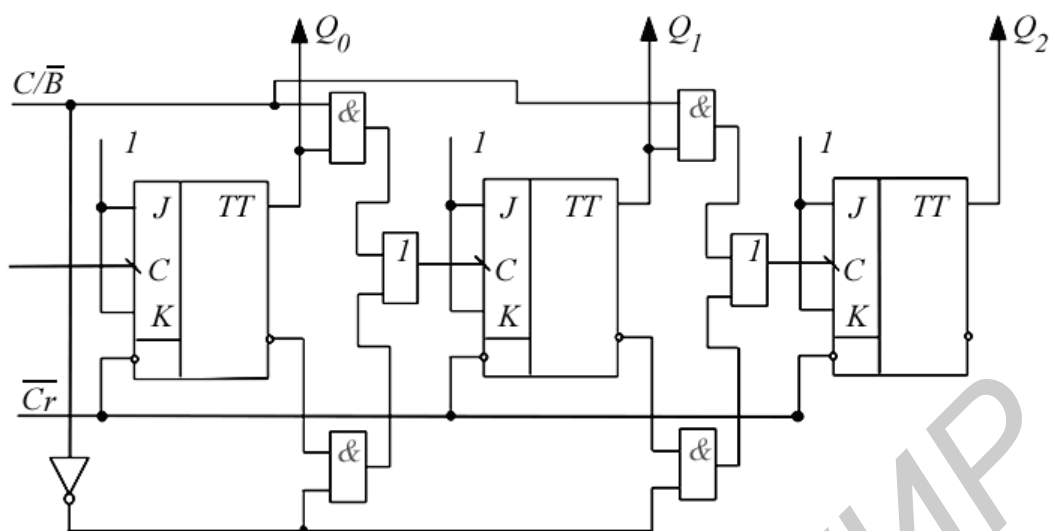


Рис. 9.4. Реверсивный асинхронный счетчик

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

### 9.1.3. Счетчики с произвольным модулем счета

Рассмотренные нами счетчики имеют модуль счета  $\text{mod}=2^N$  (2, 4, 6, 8, 16...). Однако при проектировании цифровых устройств часто возникает необходимость построить счетчик с модулем счета  $\text{mod}\neq 2^N$ . Принцип построения таких счетчиков состоит в исключении избыточных состояний с помощью включения обратных связей внутри счетчика либо методом управляемого сброса в нуль, когда в нем устанавливается определенное состояние.

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

Например, при  $N=4$  может быть получен любой  $\text{mod}$  в пределах от 9 до 16. Если необходимо получить счетчик с  $\text{mod}=10$ , то шесть состояний не используется. При этом если счет осуществляется в натуральном BCD коде, то счетчик последовательно проходит состояния (рис. 9.5), подобно двоичному счетчику, за исключением того, что после состояния 1001 (код десятичного числа 9) следующим состоянием будет 0000 (код десятичного числа 0).

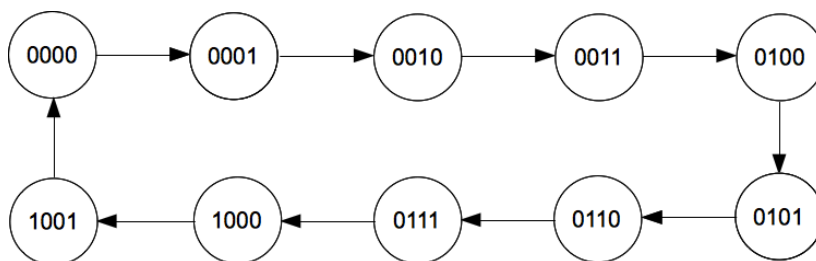


Рис. 9.5. Граф переходов двоично-десятичного счетчика



Синтез десятичного асинхронного счетчика, или любого другого асинхронного счетчика с  $\text{mod} \neq 2^N$ , или счетчика, 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 счетчика (рис. 9.6).

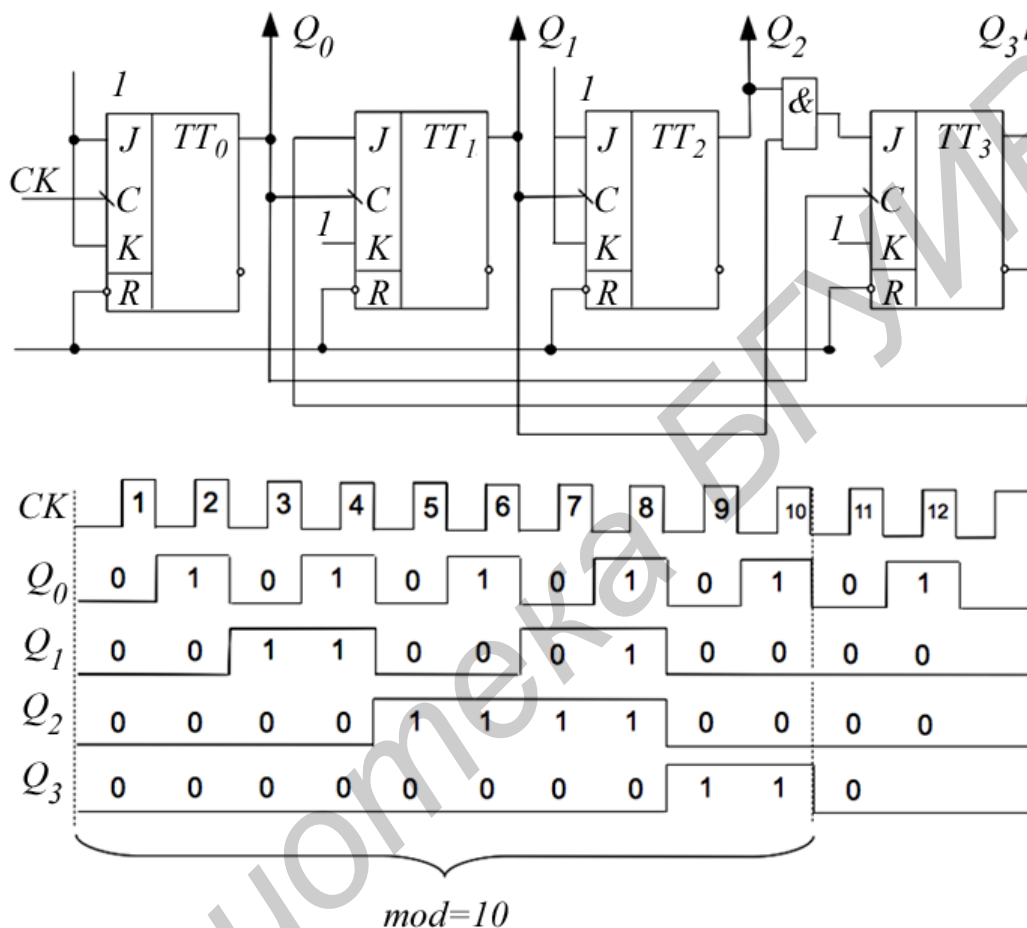


Рис. 9.6. Асинхронный двоично-десятичный счетчик

Триггеры в счетчике (рис. 9.6) перебрасываются с отрицательным перепадом синхросигнала. В схеме счетчика выход  $Q_0$  соединен с синхровходами двух триггеров  $TT_1$  и  $TT_3$ , а вход  $Q_1$  соединен с синхровходом  $TT_2$ . Входы  $J$  и  $K$  соединены либо постоянно с лог. 1, либо с выходами триггеров, как показано на рис. 9.6, а. Работа счетчика 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 свое состояние на противоположное при  $J=K=1$  и не изменяет свое состояние при  $J=K=0$ .

Из схемы рис. 9.6, а очевидно, что:

1. Выход  $Q_0$  меняет свое состояние каждым отрицательным перепадом импульсов  $CK$ .

2. Выход  $Q_1$  меняет свое состояние, если  $Q_3=0$  и  $Q_0$  имеет переход из 1 в 0.

3. Выход  $Q_2$  меняет свое состояние всякий раз, когда  $Q_1$  имеет переход из 1 в 0.

4. Выход  $Q_3$  меняет свое 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 счетчика, приведены на рис. 9.6, б.

#### 9.1.4. Интегральные схемы асинхронных счетчиков

Мы рассмотрели построение асинхронных счетчиков, с использованием отдельных триггеров. Ряд асинхронных счетчиков выпускается промышленностью как интегральные схемы средней степени интеграции. Все ИС состоят из четырех M-S-триггеров. Загрузка, установка и сброс (обнуление) осуществляются асинхронно, т. е. независимо от импульсов синхронизации.

В зависимости от особенностей счетчиков, связанных с загрузкой, установкой и сбросом, эти счетчики условно делятся на три группы.

К первой группе относится асинхронный счетчик 1533 ИЕ2 (74ALS90), который имеет входы установки и сброса. Блок-схема счетчика приведена на рис. 9.7.

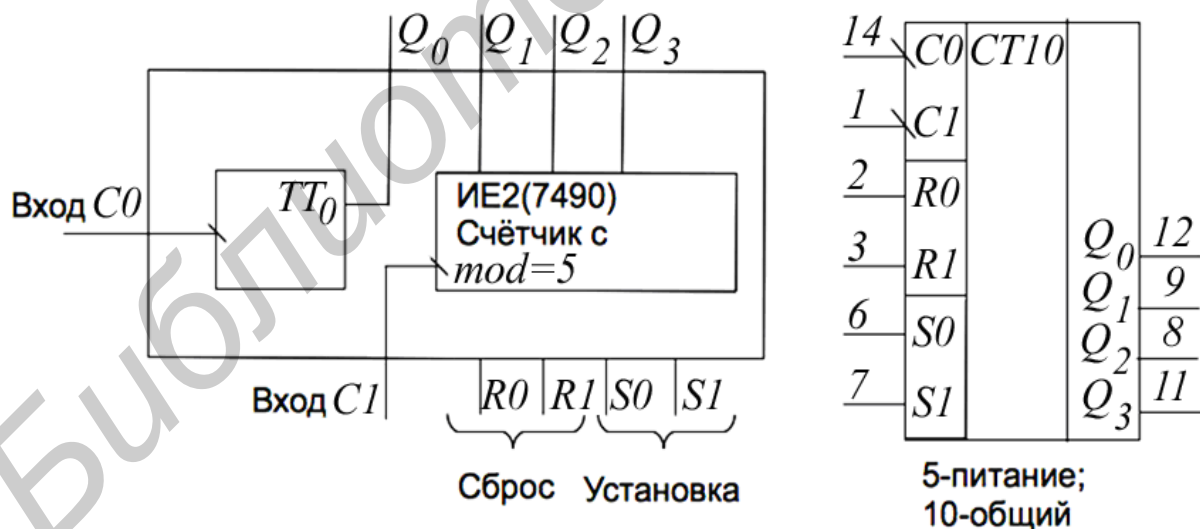


Рис. 9.7. Асинхронный счетчик ИЕ2

Счетчик состоит из четырех триггеров, объединенных внутри, как счетчик по  $mod=2$  и счетчик по  $mod=5$ . Эти счетчики могут использоваться независимо или в комбинации. Если выход счетчика с  $mod=2$  ( $Q_0$ ) соединить со входом счетчика по  $mod=5$ , то образуется двоично-десятичный счетчик.

Если соединить выход счетчика с  $\text{mod}=5$  ( $Q_3$ ) со входом счетчика  $\text{mod}=2$ , то образуется счетчик-делитель на 10. В счетчике имеются два входа: «сброс»  $R0$  и  $R1$ , на которые необходимо подать лог. 1 для обнуления счетчика. Когда на оба входа «Установка»  $S0$  и  $S1$  подается лог.1, двоично-десятичный счетчик устанавливается в состояние 1001.

Внутренняя схема и временные диаграммы работы счетчика ИЕ2 соответствуют двоично-десятичному счетчику, рассмотренному ранее (см. рис. 9.6).

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

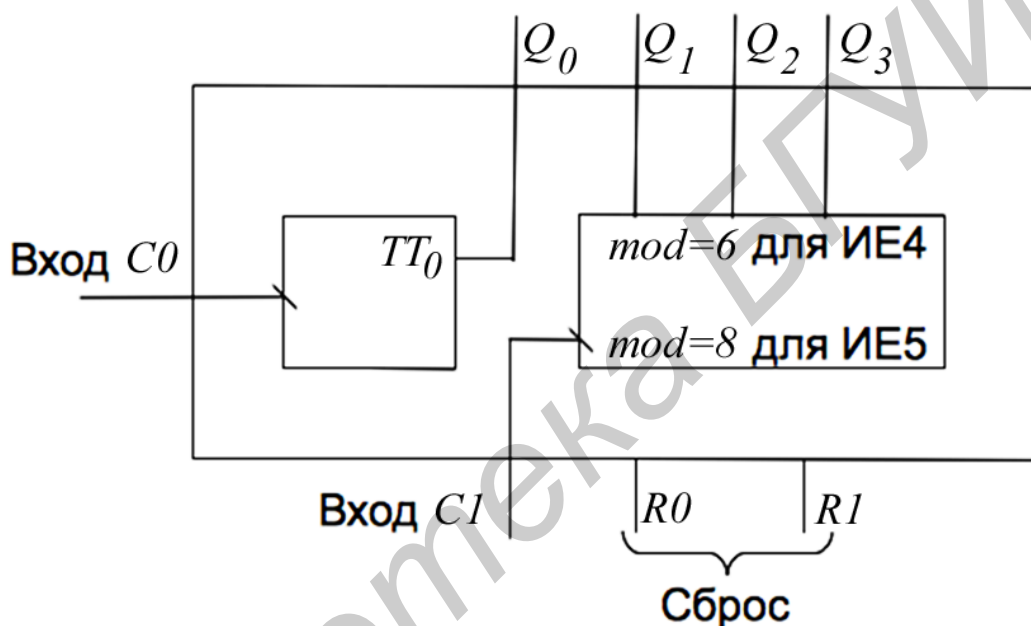


Рис. 9.8. Блок-схема асинхронных счетчиков ИЕ4, ИЕ5

Работа этого счетчика аналогична ИЕ2, за исключением того, что входы установки отсутствуют и счетчик по  $\text{mod}=6$  осуществляет счет 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 счетчика, подобных ИЕ2. Для каждого счетчика имеется один вход «Сброс» ( $R$ ). ИС1533ИЕ19 (74ALS393) содержит два 4-разрядных двоичных счетчика с входом «Сброс» ( $R$ ).

К третьей группе асинхронных счетчиков относятся счетчики 555ИЕ14 (74LS176, 74LS196) и 555ИЕ15 (74LS177, 74LS197), являющиеся версиями счетчиков ИЕ2 и ИЕ5 с предварительной установкой состояния. Блок-схема счетчиков показана на рис. 9.9.

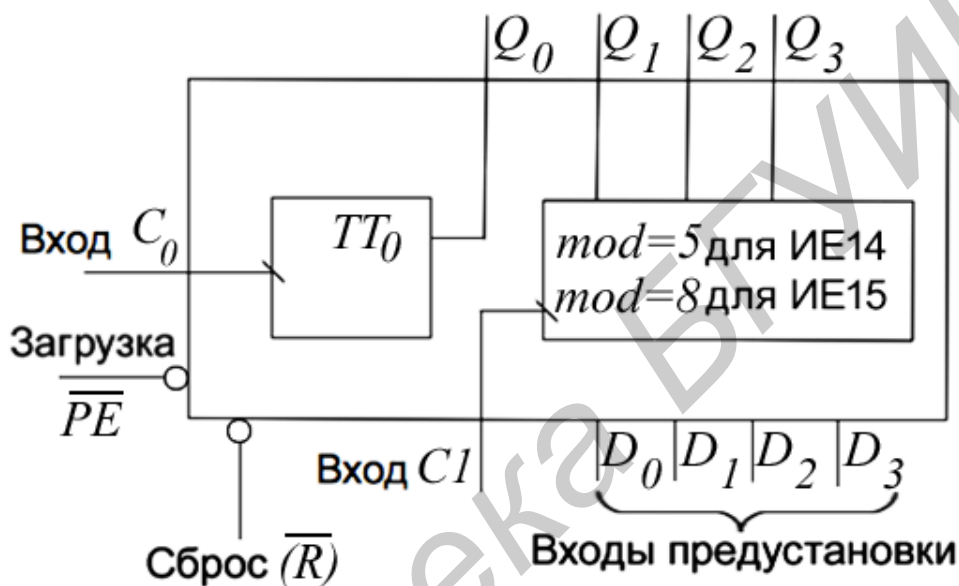


Рис. 9.9. Блок-схема асинхронных счетчиков ИЕ14, ИЕ15

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

Условное обозначение и цоколевка ИС ИЕ4, ИЕ5, ИЕ14, ИЕ15 приведены на рис. 9.10.



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

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

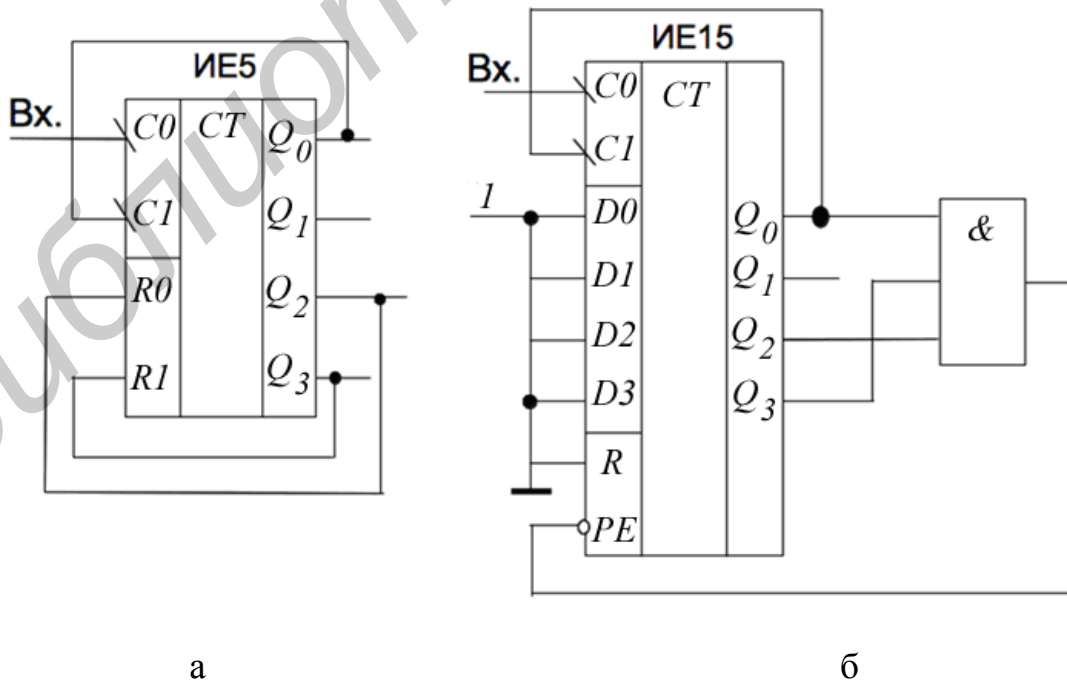


Рис. 9.11. Применение ИС ИЕ5 и ИЕ15

## 9.2. 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 счетчик; 1533ИЕ5 (74ALS93) – асинхронный двоичный счетчик; 155ИЕ4 (7492) – счетчик-делитель на 12; 555ИЕ15 (74LS197) – асинхронный двоичный счетчик; 1533ИЕ19 (74ALS393) – два двоичных асинхронных счетчика.

### 9.2.1. Исследование асинхронного двоичного счетчика на триггерах с инверсным динамическим синхровходом

1. Построить асинхронный четырехразрядный счетчик на триггерах ТВ6.
2. Исследовать работу счетчика. Результаты наблюдений представить в виде таблиц и временных диаграмм.

### 9.2.2. Исследование асинхронного двоичного счетчика на триггерах с прямым динамическим синхровходом

1. Построить асинхронный четырехразрядный счетчик на триггерах ТМ2.
2. Исследовать работу счетчика. Результаты наблюдений представить в виде таблиц и временных диаграмм.

### 9.2.3. Исследование двоично-десятичного счетчика

1. Построить асинхронный двоично-десятичный счетчик (см. рис. 9 .6).
2. Исследовать работу счетчика. Результаты представить в виде таблицы и временных диаграмм.

### 9.2.4. Исследование асинхронного счетчика ИЕ2

1. Построить двоично-десятичный счетчик на основе ИЕ2.
2. Исследовать работу счетчика. Результаты представить в виде таблиц временных диаграмм.
3. Построить счетчик-делитель на 10 на основе ИЕ 2.
4. Исследовать работу счетчика-делителя. Результаты представить в таблицы и временных диаграмм.

### **9.2.5. Исследование асинхронного счетчика ИЕ 4**

1. Построить счетчик с  $\text{mod} = 12$  на основе ИЕ 4.
2. Исследовать работу счетчика. Результаты представить в виде таблицы и временных диаграмм.

### **9.2.6. Исследование асинхронного счетчика ИЕ 5**

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

### **9.2.7. Исследование асинхронного счетчика ИЕ 15**

1. Построить десятичный счетчик на основе ИЕ 15 (см. рис. 9.11, б).
2. Исследовать работу счетчика. Результаты представить в виде таблиц и временных диаграмм.

### **9.2.8. Исследование работы асинхронного счетчика ИЕ 14**

Задание получить у преподавателя.

### **9.2.9. Исследование работы асинхронного счетчика ИЕ 19**

Задание получить у преподавателя.

## **9.3. СОДЕРЖАНИЕ ОТЧЕТА**

1. Цель работы.
2. Схемы, исследуемые в работе.
3. Таблицы наблюдений и временные диаграммы работы исследуемых счетчиков.
4. Выводы по результатам наблюдений.

## **9.4. КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. Дать определение цифрового счетчика.
2. По каким признакам классифицируются счетчики?
3. Нарисовать схемы суммирующего и вычитающего счетчиков на триггерах с прямым и с инверсным динамическим входом.
4. Чем различаются между собой асинхронные и синхронные счетчики?
5. Каковы основные достоинства и недостатки асинхронных счетчиков?
6. Охарактеризовать основные ИС асинхронных счетчиков.
7. Объяснить использование входов «Загрузка», «Сброс» и «Установка».

## Лабораторная работа №10

### ИССЛЕДОВАНИЕ СИНХРОННЫХ СЧЕТЧИКОВ

*Цель работы:*

1. Изучение основ теории и методов синтеза синхронных счетчиков.
2. Исследование работы синхронных счетчиков.

#### 10.1. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

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

Таблица 10.1

Счет	Состояния счетчика		
	$Q_2$	$Q_1$	$Q_0$
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  $Q_0$  младшего триггера счетчика изменяется с каждым счетным импульсом. Это может быть достигнуто, если использовать  $T$ -триггер с  $T=1$ . Выход  $Q_1$  изменяет свое состояние всякий раз, когда  $Q_0$  изменяется с 1 на 0. Поэтому если соединить выход  $Q_0$  с входом  $T_1$ , то



$Q_1$  будет изменять свое состояние, когда  $Q_0=1$ , ( $T_1=Q_0=1$ ), и будет оставаться без изменения, когда  $T_1=Q_0=0$ . Из табл. 10.1 также видно, что  $Q_2$  изменяет свое состояние всякий раз, когда  $Q_1$  и  $Q_0$  оба равны 1. Это может быть реализовано, если вход наиболее значащего триггера  $TT_2=Q_1 \cdot Q_0$ . В результате получим схему синхронного счетчика (mod = 8) (рис. 10.1). Аналогичным образом могут быть построены синхронные счетчики с mod = 2N.

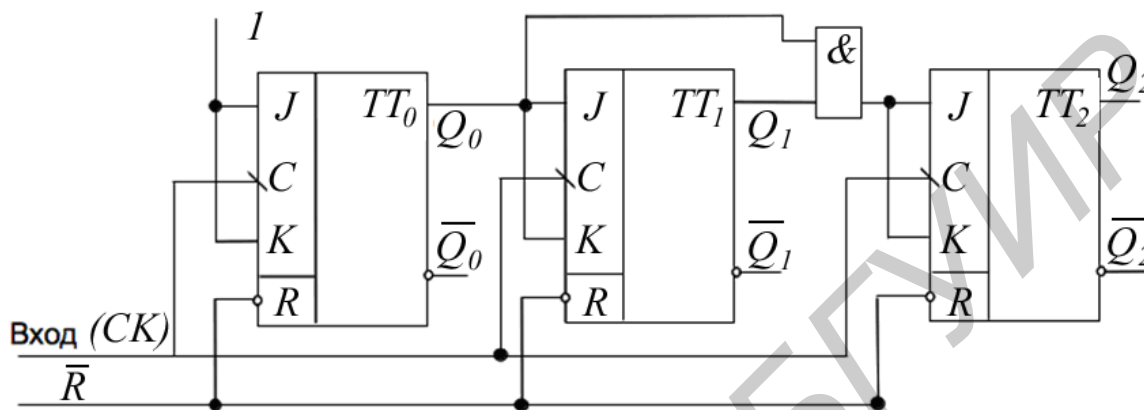


Рис. 10.1. Синхронный счетчик с mod=8

## 10.2. СИНТЕЗ СИНХРОННЫХ СЧЕТЧИКОВ

Синхронный счетчик для любой заданной последовательности счета и модуля счета может быть синтезирован следующим образом.

1. Необходимо определить требуемое число триггеров исходя из 
$$N \geq \log_2 m,$$

где  $m$  – модуль счета.

2. Записать счетную последовательность в табличном виде.
3. Определить состояния входов триггеров, необходимых 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 метод Карно, получить минимизированные выражения для каждого входа триггеров.
5. Построить схему счетчика, используя триггеры и логические элементы, в соответствии с минимизированными выражениями.

**Пример 10.1.** Построить двоично-десятичный счетчик, который имеет десять состояний.

*Решение.* Для этого 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 счетчика $Q_3 Q_2 Q_1 Q_0$	Входы триггеров							
	$J_3$	$K_3$	$J_2$	$K_2$	$J_1$	$K_1$	$J_0$	$K_0$
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 метод Карно, получим минимизированные выражения для всех входов  $J$  и  $K$  (рис. 10.2). Используя эти выражения, строим синхронный двоично-десятичный счетчик (рис. 10.3).

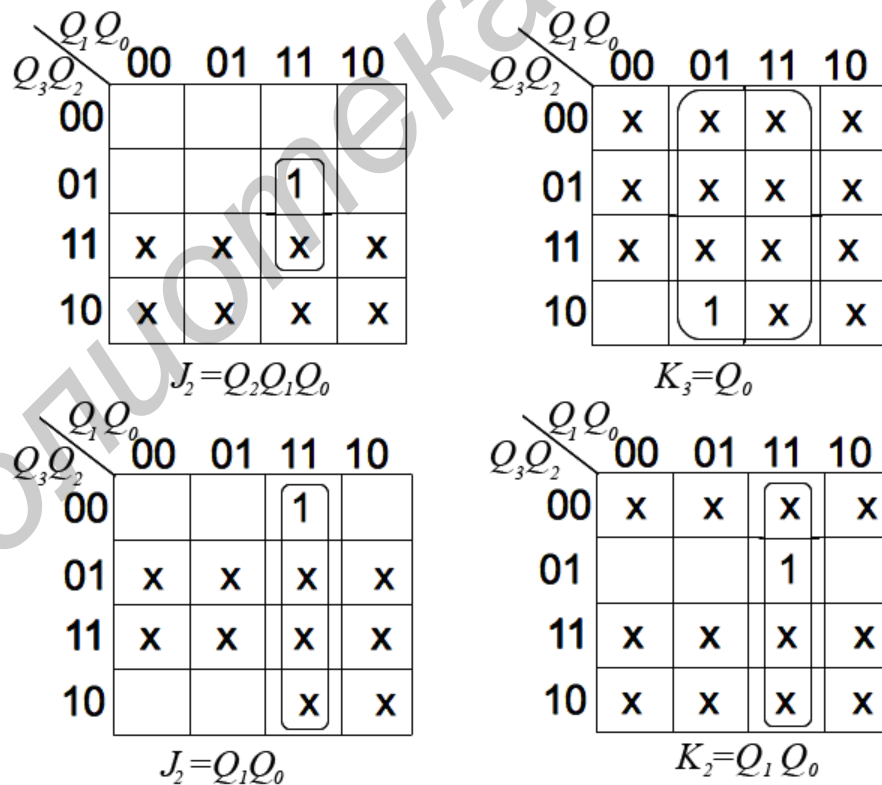


Рис. 10.2. Минимизация функций  $J$  и  $K$  с помощью карт Карно (окончание на с. 123)

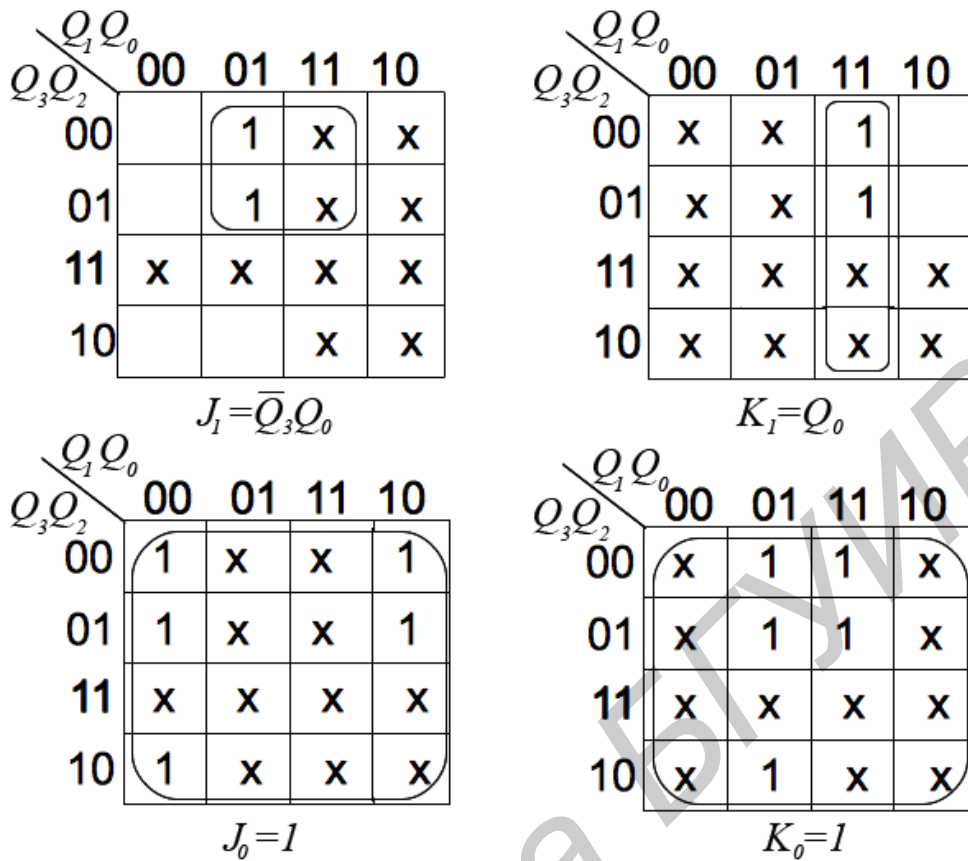


Рис. 10.2. Окончание (начало см. на с. 122)

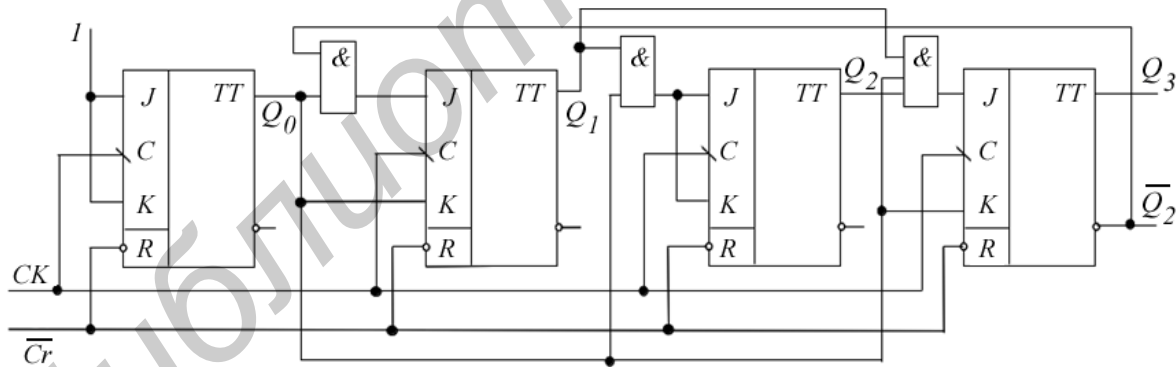


Рис. 10.3 Синхронный двоично-десятичный счетчик

### 10.2.1. Интегральные схемы синхронных счетчиков

Мы рассмотрели синтез синхронных счетчиков. Счетчики с любой счетной последовательностью и любым модулем счета могут быть синтезированы с помощью метода Карно. Промышленностью выпускаются синхронные счетчики как интегральные схемы средней степени интеграции. Все счетчики содержат по четыре триггера с динамическим синхровходом. Изменения со-

стояний триггеров, синхронная загрузка и синхронное обнуление происходят путем положительного перепада синхроимпульсов. В силу некоторых характерных особенностей синхронные счетчики могут быть подразделены на четыре группы.

*Синхронные счетчики группы 1.* К этой группе относятся счетчики ИЕ9(74160), ИЕ10(74161), ИЕ11(74162), ИЕ18(74163). Это двоично-десятичные и двоичные суммирующие счетчики с синхронной загрузкой и асинхронным сбросом. Блок-диаграмма, условное обозначение и цоколевка этих счетчиков приведены на рис. 10.4.

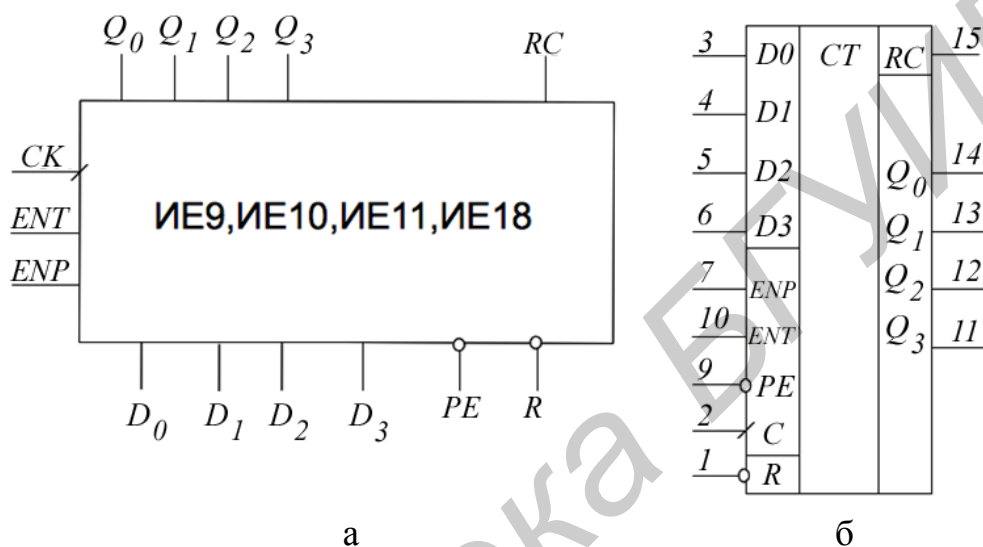


Рис. 10.4. Синхронные счетчики группы 1

Эти счетчики имеют два отдельных разрешающих входа *ENT* и *ENP*. При подаче на любой из этих входов лог. 0 счет останавливается асинхронно. Выход последовательного переноса *RC* находится в состоянии лог. 0 и становится лог. 1 всякий раз, когда счетчик достигает своего максимального значения (двоичное число 9 для двоично-десятичного счетчика и двоичное 15 для двоичного счетчика). Функционирование счетчиков группы 1 отражено в табл. 10.3.

Таблица 10.3

<i>L</i>	<i>ENP</i>	<i>ENT</i>	<i>Cr</i>	<i>CK</i>	Операция
0	x	x	1	↑	Установка (загрузка)
1	0	1	1	x	Остановка счета
1	x	0	1	x	Остановка счета, невозможность <i>RC</i>
x	x	x	0	*	Сброс в нуль
1	1	1	1	↑	Прямой счет

**Примечания:**

x для ИЕ9, ИЕ10.

↑ для ИЕ11, ИЕ18.

**Пример 10.2.** Построить делитель на 11, используя ИЕ18. Использовать  $RC$  выход и установочные входы.

*Решение.* Для получения делителя на 11 счетчик устанавливается в двоичное состояние 0101 (десятичное 5). Когда счет достигнет 1111, на выходе  $RC$  появляется лог. 1, которая используется для загрузки данных с установочных входов в счетчик (рис. 10.5).

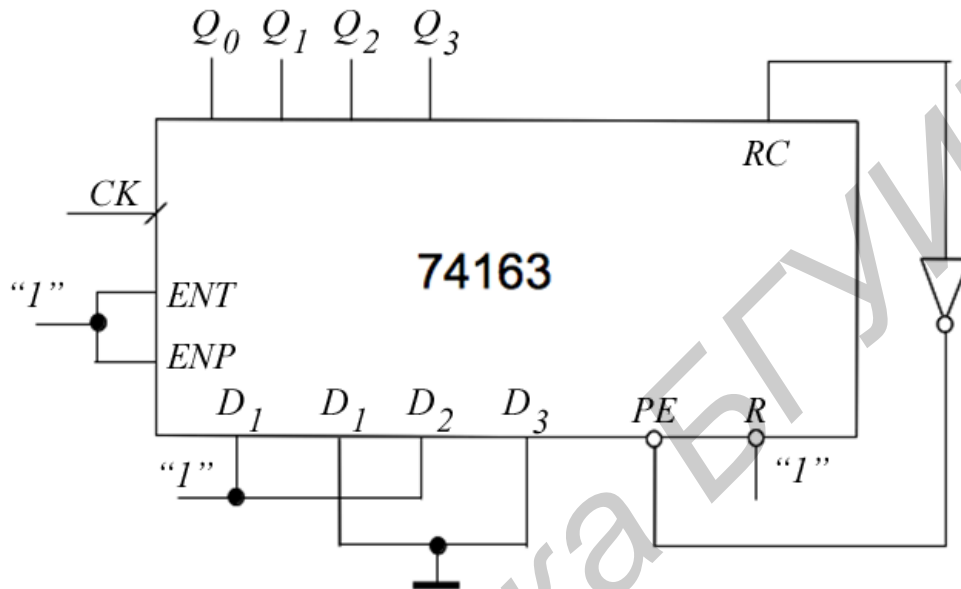


Рис. 10.5. Схема делителя на 11

Итак, для получения делителя на  $m$  на установочных входах должно быть:  $D=16-m$  для двоичного счетчика и  $D=10-m$  для десятичного счетчика. Входы  $ENT$ ,  $ENP$  и  $RC$  могут использоваться для организации каскадного соединения счетчиков (рис. 10.6).

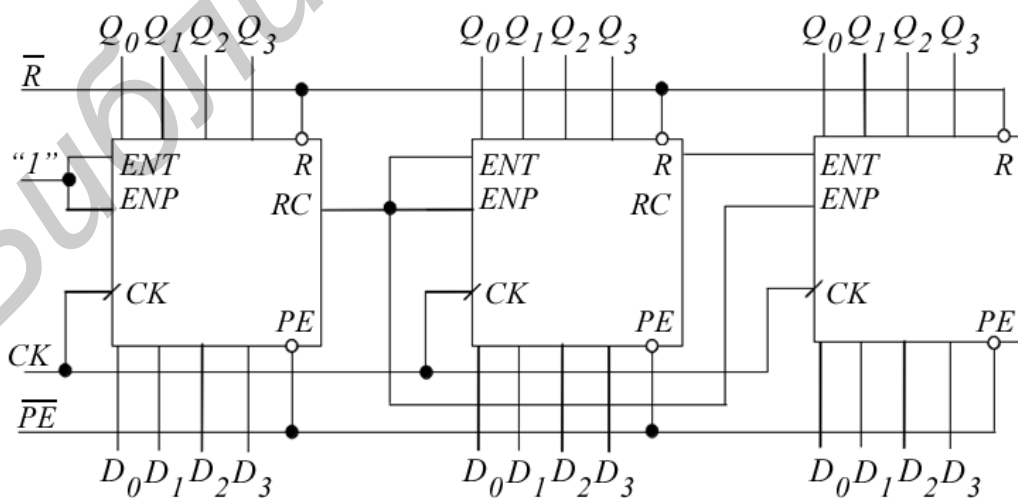


Рис. 10.6. Каскадное соединение счетчиков первой группы

Синхронные счетчики группы 2. К этой группе относятся счетчики ИЕ16(74168) и ИЕ17(74169). Это соответственно двоично-десятичный и двоичный реверсивные счетчики с синхронной установкой и без сброса (рис. 10.7). Функция  $ENT$  и  $ENP$  та же, что и у счетчиков 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 счет достигает максимального значения при прямом счете или когда счет достигает минимального значения при обратном счете.

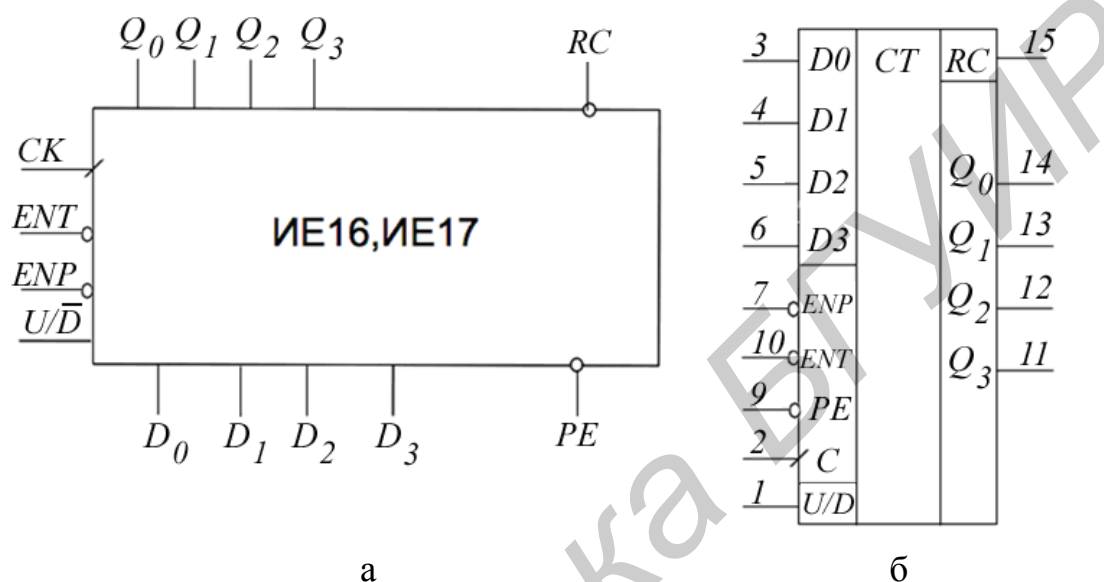


Рис. 10.7. Блок-диаграмма, условное обозначение и цоколевка ИЕ16, ИЕ17

Функционирование счетчиков ИЕ16 и ИЕ17 отражено в табл. 10.4.

Таблица.10.4

$PE$	$ENP$	$ENT$	$U/D$	$CK$	Операция
0	x	x	x	↑	Установка
1	1	0	x	x	Остановка счета
1	x	1	x	x	Остановка счета, запрет переноса
1	0	0	1	↑	Прямой счет
1	0	0	0	↑	Обратный счет

Сигнал на входе  $U/\bar{D}$  определяет направление счета:  $U/D=1$  для прямого счета и  $U/\bar{D}=0$  для обратного. В счетчиках этой группы отсутствует вход сброса  $R$ . Поэтому если необходимо остановить счет 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 счетчика.

**Пример 10.3.** Построить счетчик с начальным состоянием 0011 и конечным 1100, используя ИЕ17.

*Решение*

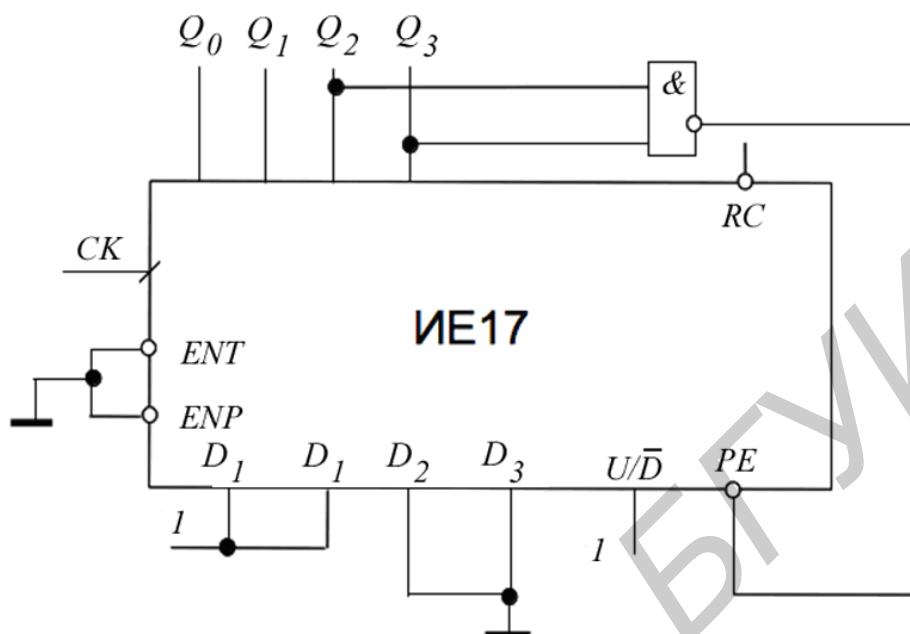


Рис.10.8. Двоично-десятичный счетчик, осуществляющий счет в Excess-3 коде

Когда счетчик (рис. 10.8) достигнет состояния 1100, на выходе элемента И-НЕ и на входе  $PE$  появится лог. 0. Следующим синхроимпульсом (положительным перепадом) счетчик установится в состояние 0011. Необходимо учитывать, что установка счетчика осуществляется синхронно.

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

*Синхронные счетчики группы 3.* К этой группе относятся счетчики ИЕ12(74190) и ИЕ13(74191). Это соответственно двоично-десятичный и двоичный реверсивные счетчики с асинхронной установкой и без сброса. Эти счетчики имеют только один низкий активный разрешающий вход  $EN$ . Выход  $MAX/MIN$  (рис. 10.9) используется для определения максимального или минимального состояния счетчика. На этом выходе нормально лог. 0, а лог. 1 появляется, когда максимальное состояние счетчика 1001 для ИЕ12 и 1111 для ИЕ13 при прямом счете или когда минимальное состояние 0000 при обратном счете.

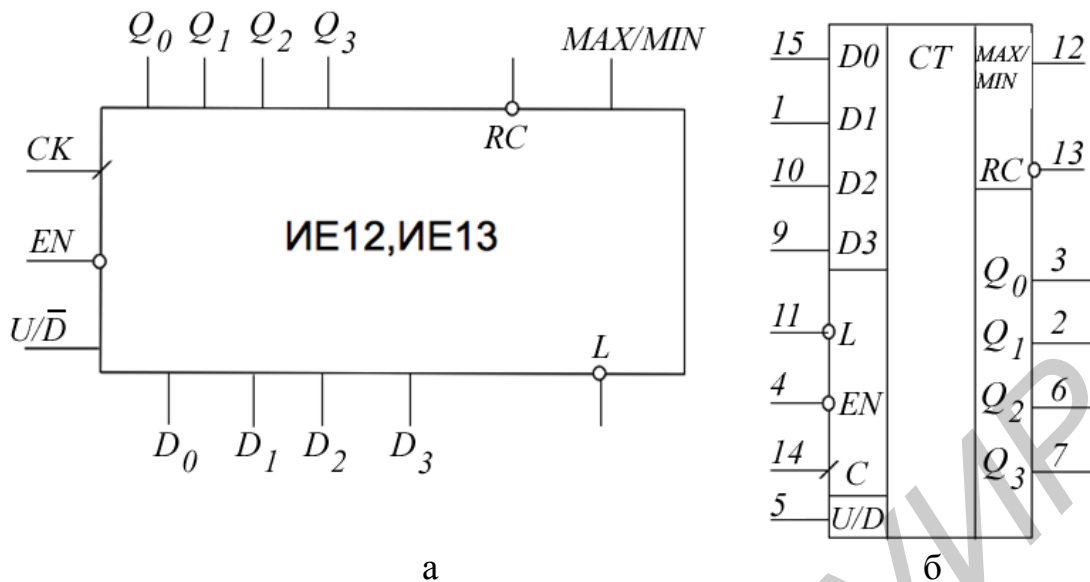


Рис. 10.9. Блок-диаграммы, условное обозначение и цоколевка ИЕ12,ИЕ13

Выход  $RC$  нормально в состоянии лог.1 и переходит в лог. 0, когда счетчик достигает  $MAX/MIN$  точки и импульс синхронизации становится равным 0. Функционирование счетчиков ИЕ12, ИЕ13 отражено в табл. 10.5.

Таблица 10.5

$L$	$ENAB$	$U/D$	$CK$	Операция
x	1	x	x	Остановка счета
0	0	x	x	Установка счетчика
1	0	0	↑	Прямой счет
1	0	1	↑	Обратный счет

Синхронные счетчики третьей группы могут каскадироваться различными путями. Максимальное быстродействие возникает, если использовать параллельный перенос (рис. 10.10).



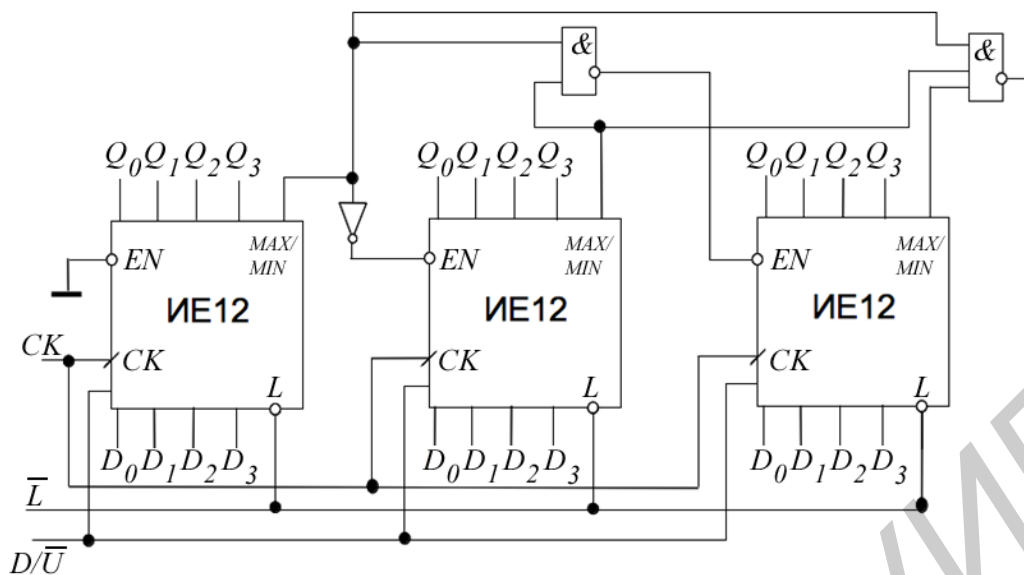


Рис. 10.10. Каскадные соединения счетчиков третьей группы

Синхронные счетчики группы 4. К этой группе относятся счетчики ИЕ6(74192) и ИЕ7(74193). Это соответственно двоично-десятичный и двоичный реверсивные счетчики с асинхронной загрузкой и сбросом. У этих счетчиков (рис.10.11) импульсы для прямого счета подаются на вход  $C-U$ , при этом вход  $C-D$  соединяется с лог.1 и импульсы для обратного счета подаются на вход  $C-D$ , соединяющийся с лог. 1.

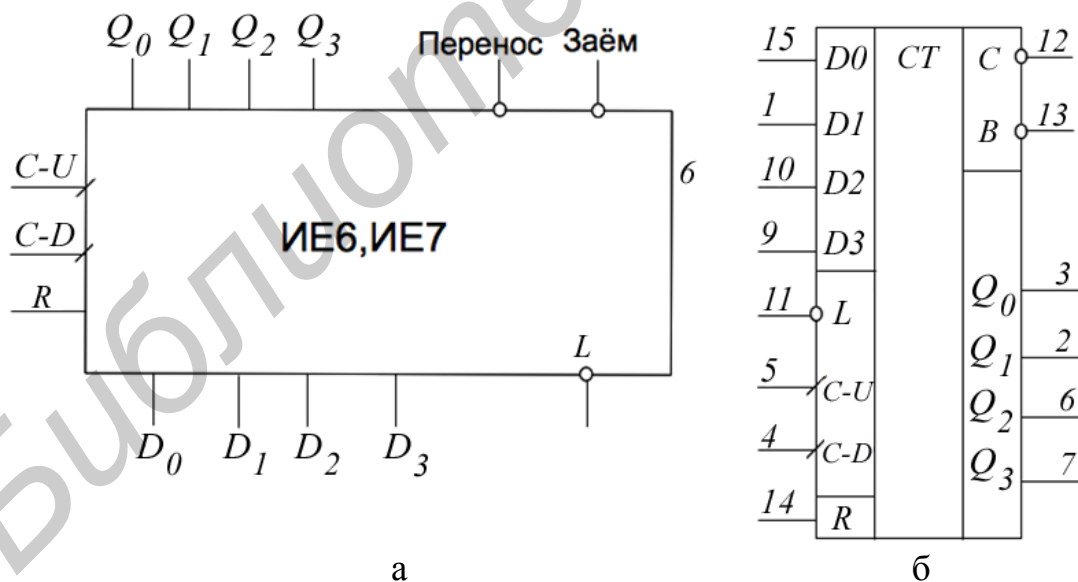


Рис. 10.11. Блок-диаграмма, условное обозначение и цоколевка ИЕ6 и ИЕ7

Выход  $C$  (перенос) и  $B$  (заем) обычно находятся в состоянии лог.1. Выход переноса переходит в лог. 0, когда счетчик достигает максимального

значения при прямом счете и вход  $C-U$  находится в лог. 0. Выход заем остается в лог. 1, когда счетчик работает от прямого  $C-U$  входа.

Функция заем при обратном счете аналогична переносу при прямом счете.

Функционирование счетчиков ИЕ6 и ИЕ7 отражено в табл. 10.6.

Таблица 10.6

$L$	$R$	$C-U$	$C-D$	Операция
x	1	x	x	Сброс
1	0	↑	1	Прямой счет
1	0	↑	↑	Обратный счет
0	0	x	x	Установка
1	0	1	1	Остановка счета

При каскадировании этих счетчиков перенос и заем каждой ступени должны быть соединены с входами  $C-U$  и  $C-D$  последующей ступени соответственно. Для управления импульсами на прямой и обратный счетный вход может быть использована схема, приведенная на рис. 10.12.

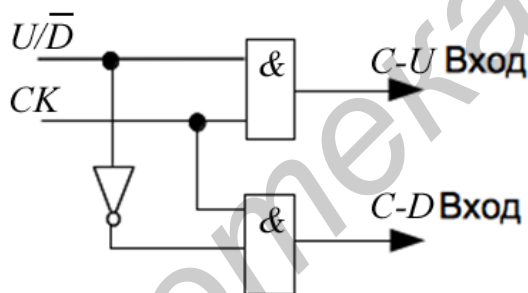


Рис. 10.12. Схема для управления счетными импульсами при прямом и обратном счете

### 10.3. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

*Оборудование и компоненты:* универсальная лабораторная установка IDL-800; интегральные схемы: 1533ЛИ1(74ALS08), 1533ТВ6(74ALS107), 1533И9(74ALS160), 1533ИЕ10(74ALS161), 1533ИЕ11(74ALS162), 1533ИЕ18 (74ALS163), 1533ИЕ16(74ALS168), 1533ИЕ17(74ALS169), 1533ИЕ12(74ALS190), 1533ИЕ13(74ALS191), 1533ИЕ6(74ALS192), 1533ИЕ7(74ALS193).

#### 10.3.1. Исследование двоично-десятичного счетчика

1. Используя ИС 1533ТВ6 и 1533ЛИ1, собрать схему двоично-десятичного счетчика (см. рис. 10.3).

2. Подавая на счетный вход одиночные импульсы, исследовать работу счетчика.

3. Результаты исследований предоставить в табличной форме и в виде временных диаграмм.

### **10.3.2. Синтез и исследование синхронного счетчика, работающего в коде Грея**

1. Синтезировать счетчик, работающий в коде Грея, для  $\text{mod}=8$ .

2. Используя ИС 1533ТВ6 и 1533ЛИ1, собрать схему синхронного счетчика кода Грея.

3. Подавая на счетный вход одиночные импульсы, исследовать работу счетчика.

4. Результаты исследований представить в виде таблиц и временных диаграмм.

### **10.3.3. Исследование модульных счетчиков первой группы**

1. Построить делитель на 7 на ИС 1533ИЕ9. Использовать *RC*-выход и установочные входы.

2. Исследовать работу делителя на 7.

3. Результаты представить в виде таблиц и временных диаграмм.

4. Построить и исследовать работу двухдекадного счетчика в режиме прямого счета и в режим обратного счета.

### **10.3.4. Исследование модульных счетчиков второй группы**

1. Построить двоично-десятичный счетчик, осуществляющий счет в Excess-3 коде (см. рис. 10.8).

2. Исследовать работу счетчика.

3. Результаты представить в виде таблиц и временных диаграмм.

### **10.3.5. Исследование модульных счетчиков третьей группы**

1. Исследовать работу ИС 1533ИЕ12(13) в режиме прямого и в режиме обратного счета.

2. Результаты представить в виде таблиц и временных диаграмм. Временные диаграммы должны содержать состояния счетчика на всех выходах, включая *RC* и *MAX/MIN*.

### **10.3.6. Исследование модульных счетчиков четвертой группы**

1. Используя ИС 1533ИЕ7, построить двухдекадный счетчик.

2. Исследовать работу счетчика в режиме прямого и обратного счета.

3. Результаты исследований представить в виде таблиц и временных диаграмм.

#### **10.4. СОДЕРЖАНИЕ ОТЧЕТА**

1. Цель работы.
2. Схемы, исследуемые в работе.
3. Результаты исследований: таблицы, временные диаграммы.
4. Выводы.

#### **10.5. КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. Дать определение электронного счетчика.
2. Чем отличается синхронный счетчик от асинхронного?
3. Каковы достоинства и недостатки синхронных счетчиков?
4. В чем заключается синтез синхронного счетчика?
5. Дать характеристику и особенности модульных счетчиков.

Библиотека БГУИР

## ЛИТЕРАТУРА

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

## УНИВЕРСАЛЬНАЯ ЛАБОРАТОРНАЯ УСТАНОВКА IDL-800 Digital Lab

Лабораторная установка IDL-800 (рис. П.1) предназначена для физического макетирования и исследования функционирования цифровых логических устройств, а также может быть использована для исследования аналоговых устройств.

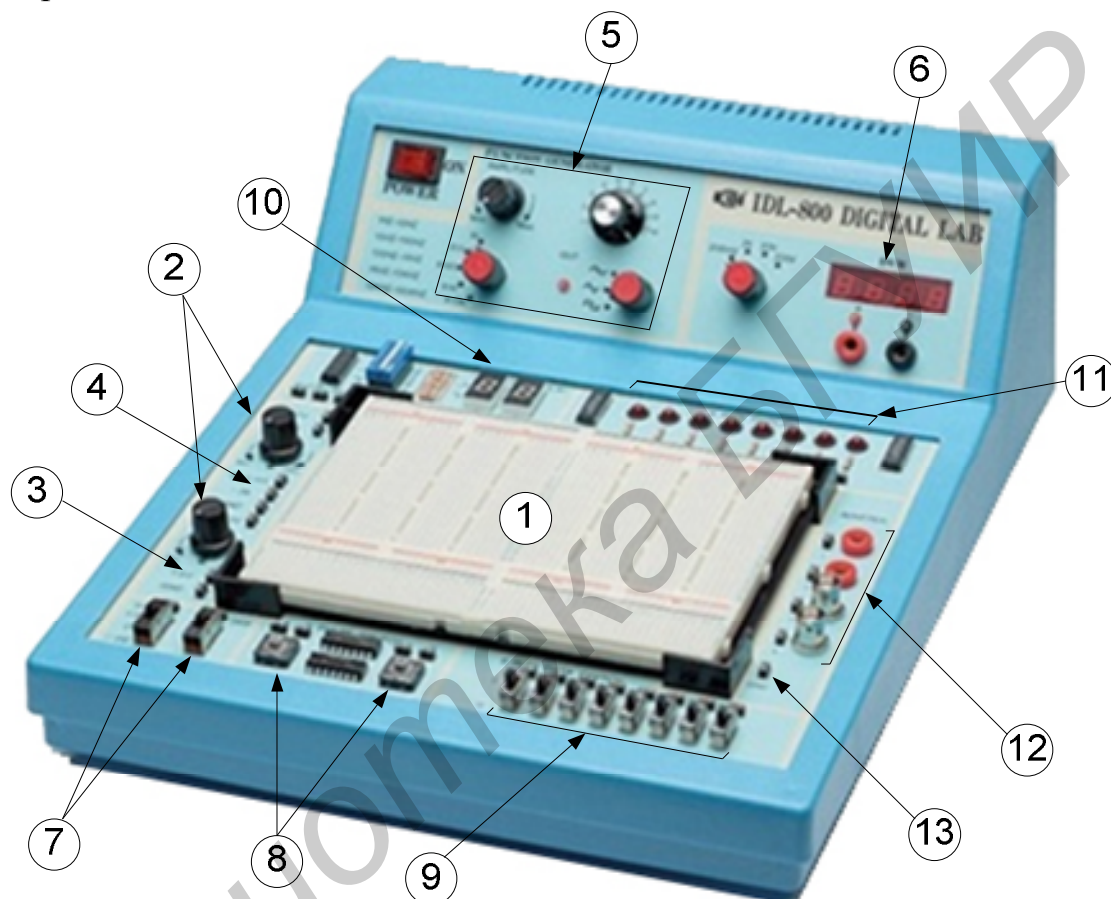


Рис. П.1. Лабораторная установка IDL-800 Digital Lab

Лабораторная установка содержит источники постоянного напряжения, функциональный генератор, цифровой вольтметр, восьмиразрядный светодиодный дисплей, два семисегментных индикатора и логические переключатели. В лабораторную установку входит наборная плата AD-200, на которой собираются исследуемые устройства. Детализация лабораторной установки показана на рис. П.1:

1. Наборная плата AD-200 для моделирования исследуемых устройств имеет 1896 контактов. Расстояния между контактами и размер контактов позволяют вставлять в плату *DIP*-компоненты, а также транзисторы, резисторы, конденсаторы и т. п. Соединения между компонентами осуществляются с помощью проводов диаметром 0,3...0,8 мм.

2. Регулируемые источники электропитания  $0 \dots +15\text{В}$  и  $0 \dots -15\text{В}$  с максимальным выходным током  $300 \text{ мА}$ .

3. Нерегулируемый источник электропитания  $+5\text{В}$  с максимальным выходным током  $1 \text{ А}$ .

4. Нерегулируемый источник электропитания  $-5\text{В}$  с максимальным выходным током  $100 \text{ мА}$ . Все источники электропитания имеют защиту от короткого замыкания.

5. Функциональный генератор, формирующий синусоидальный или треугольный, или меандровый сигнал в диапазонах частот:  $1 \dots 10 \text{ Гц}$ ,  $10 \dots 100 \text{ Hz}$ ,  $100\text{Hz} \dots 1\text{kHz}$ ,  $1 \dots 10 \text{ kHz}$ ,  $10 \text{ kHz} \dots 100 \text{ kHz}$  с плавной регулировкой частоты внутри диапазонов. Амплитуда синусоидального сигнала регулируется в пределах от  $0$  до  $4\text{В}$ . Амплитуда двуполярного треугольного сигнала регулируется в пределах от  $0$  до  $3 \text{ В}$ . Амплитуда двуполярного меандрового сигнала регулируется в пределах от  $0$  до  $4 \text{ В}$ .

6. Цифровой вольтметр имеет 4 диапазона измерений:

$0 - 199,9 \text{ В}$ ;

$0 - 19,99 \text{ В}$ ;

$0 - 1,999 \text{ В}$ ;

$0 - 199,9 \text{ мВ}$ .

Входное сопротивление:  $10 \text{ МОм}$  в любом диапазоне.

*Внимание.* Цифровой вольтметр не имеет защиты от перегрузки.

7. Два функциональных переключателя  $-5\text{В}/\text{ОВ}/+5\text{В}$ .

8. Две антидребезговые кнопки для генерирования одиночных импульсов с прямым и инверсным выходом.

9. 8 переключателей логических уровней «0»/ «1».

10. Два семисегментных индикатора, включенных по схеме с общим катодом.

11. Дисплей на восьми буферезированных светодиодах.

12. Четырехточечный адаптер для подключения измерительных приборов.

13. Контакты общей шины лабораторной установки.

Лабораторная установка IDL-800 Digital Lab вместе с контрольно-измерительными приборами образует рабочее место в исследовательской лаборатории и позволяет проводить физические эксперименты с цифровыми и аналоговыми устройствами.

*Учебное издание*

**Будько Анатолий Антонович**

**ЦИФРОВЫЕ УСТРОЙСТВА.  
ТЕХНОЛОГИЯ IDL**

**УЧЕБНО-МЕТОДИЧЕСКОЕ ПОСОБИЕ**

Редактор *Н.В. Гриневич*  
Корректор *Е. Н. Батурчик*

Подписано в печать 18.08.2011. Формат 60x84 1/16. Бумага офсетная. Гарнитура «Таймс».  
Отпечатано на ризографе. Усл. печ. л. 8,02. Уч.-изд. л. 7,6. Тираж 120 экз. Заказ 896.

Издатель и полиграфическое исполнение: учреждение образования  
«Белорусский государственный университет информатики и радиоэлектроники»  
ЛИ №02330/0494371 от 16.03.2009. ЛП №02330/0494175 от 30.04.2009  
220013, Минск, П. Бровки, 6