

Министерство образования Республики Беларусь

Учреждение образования  
«Гомельский государственный технический  
университет имени П. О. Сухого»

Кафедра «Промышленная электроника»

**Ю. Е. Котова, Л. А. Захаренко**

**СХЕМОТЕХНИКА  
В СИСТЕМАХ УПРАВЛЕНИЯ.  
ЦИФРОВАЯ СХЕМОТЕХНИКА**

**ПРАКТИКУМ**

**по выполнению лабораторных работ  
для студентов специальности 1-53 01 07  
«Информационные технологии  
и управление в технических системах»  
дневной формы обучения**

**Электронный аналог печатного издания**

**Гомель 2019**

УДК 681.58/621.3(075.8)  
ББК 32я73  
К73

*Рекомендовано к изданию научно-методическим советом  
факультета автоматизированных и информационных систем  
ГГТУ им. П. О. Сухого  
(протокол № 8 от 05.03.2018 г.)*

Рецензент: доцент кафедры «Автоматизированный электропривод» ГГТУ  
им. П. О. Сухого канд. техн. наук, доц. *М. Н. Погуляев*

**Котова, Ю. Е.**

К73

Схемотехника в системах управления. Цифровая схемотехника : практикум по выполнению лаборатор. работ для студентов специальности 1-53 01 07 «Информационные технологии и управление в технических системах» днев. формы обучения / Ю. Е. Котова, Л. А. Захаренко. – Гомель : ГГТУ им. П. О. Сухого, 2019. – 69 с. – Систем. требования: PC не ниже Intel Celeron 300 МГц ; 32 Mb RAM ; свободное место на HDD 16 Mb ; Windows 98 и выше ; Adobe Acrobat Reader. – Режим доступа: <https://elib.gstu.by>. – Загл. с титул. экрана.

ISBN 978-985-535-405-6.

Предназначен для получения и закрепления знаний, требуемых в рамках учебной программы по дисциплине «Схемотехника в системах управления» на лабораторных занятиях и при самостоятельной работе.

Для студентов специальности 1-53 01 07 «Информационные технологии и управление в технических системах» дневной формы обучения.

УДК 681.58/621.3(075.8)  
ББК 32я73

ISBN 978-985-535-405-6

© Котова Ю. Е., Захаренко Л. А., 2019  
© Учреждение образования «Гомельский  
государственный технический университет  
имени П. О. Сухого», 2019

## **Лабораторная работа № 1**

### **Исследование логических элементов**

**Цель работы:** изучение основных логических функций алгебры логики и способы их получения с помощью базовых логических элементов.

#### **Используемое оборудование:**

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

#### **Основные теоретические сведения**

Называемая в честь английского математика Дж. Буля *булевой алгеброй*, алгебра логики составляет теоретическую основу логики, теории алгоритмов и логического проектирования цифровых схем. Отличие *булевой алгебры* от традиционной главным образом состоит в том, что первая оперирует константами и переменными, принимающими только два возможных значения – 0 или 1.

*Булева переменная* – это параметр, который в различных случаях принимает значение 0 или 1. Эти переменные часто используют для представления уровня напряжения в проводнике или на контакте ввода–вывода схемы. Например, булево значение 0 может быть назначено для представления любого напряжения в диапазоне от 0 до 0,8 В, в то время как значение 1 характерно для представления любого напряжения в диапазоне от 2 до 5 В.

Таким образом, булевы 0 и 1 не являются реальными величинами, а представляют состояния переменной напряжения, или так называемые *логические уровни*. Напряжение в цифровой схеме представляет логический уровень 0 или логический уровень 1 – в зависимости от присущих ему действительных численных значений.

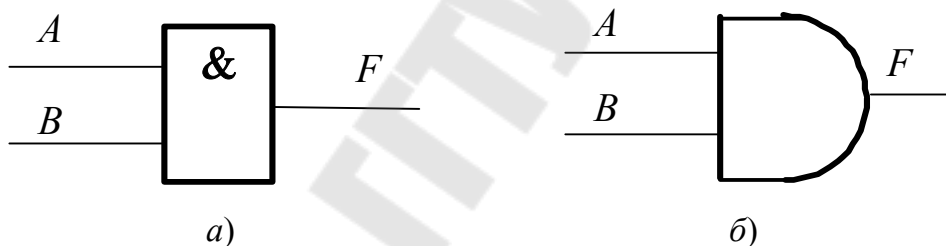
Логическая функция может быть задана словесно, алгебраическим выражением и таблицей, которая называется таблицей истинности. Действия над двоичными переменными производятся по правилам логических операций.

Простейших логических операций три: *отрицание* (инверсия, операция НЕ), *логическое умножение* (конъюнкция, операция И) и *логическое сложение* (дизъюнкция, операция ИЛИ). Более сложные логические преобразования можно свести к указанным операциям.

### ***Логические элементы, реализующие элементарные функции алгебры логики***

Наиболее часто встречаются следующие названия и буквенные обозначения функции И: логическое умножение, конъюнкция, совпадение, AND, И. Возможные виды алгебраической записи функции И:  $F = A \wedge B$ ;  $F = A \cdot B$ ;  $F = AB$ .

Условное графическое обозначение (УГО) логического элемента, реализующего функцию И, изображено на рис. 1.1.



*Рис. 1.1.* УГО логического элемента, реализующего функцию И:  
*a* – в отечественных схемах; *б* – обозначение в соответствии с американской системой MILSPEC 806B

Наиболее часто встречаются следующие названия и буквенные обозначения функции ИЛИ: логическое сложение, дизъюнкция, OR, ИЛИ. Алгебраическая запись функции ИЛИ:  $F = A \vee B$ ;  $F = A + B$ .

Условное графическое обозначение логического элемента, реализующего функцию И, изображено на рис. 1.2.

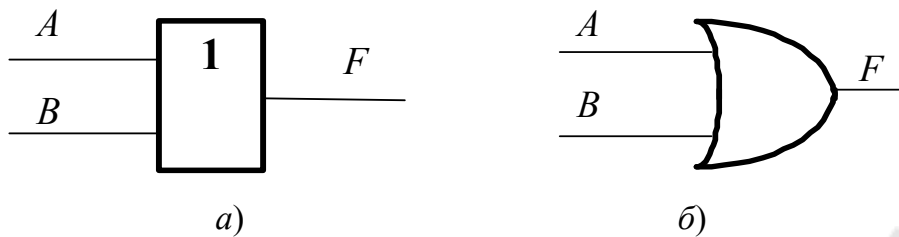


Рис. 1.2. УГО логического элемента, реализующего функцию ИЛИ:  
*a* – в отечественных схемах; *б* – обозначение в соответствии  
 с американской системой MILSPEC 806B

Наиболее часто встречаются следующие названия и буквенные обозначения функции НЕ: *логическое отрицание, инверсия, дополнение, NOT, НЕ*. Возможные виды алгебраической записи функции НЕ:  $F = A'$ ,  $F = \bar{A}$ .

Условное графическое обозначение логического элемента, реализующего функцию НЕ, изображено на рис. 1.3.

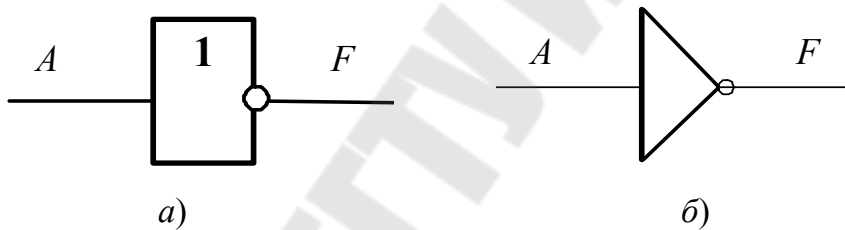


Рис. 1.3. УГО логического элемента, реализующего функцию НЕ:  
*a* – в отечественных схемах; *б* – обозначение в соответствии  
 с американской системой MILSPEC 806B

Хочется отметить еще один логический элемент для реализации двухвходовой функции ИСКЛЮЧАЮЩЕЕ ИЛИ, булево выражение которого в двухвходовом случае совпадает с булевым выражением для сумматора по модулю два, и имеет следующий вид:  $F = \bar{A}B + A\bar{B}$ .

Условное графическое обозначение логического элемента, реализующего функцию ИСКЛЮЧАЮЩЕЕ ИЛИ, изображено на рис. 1.4.

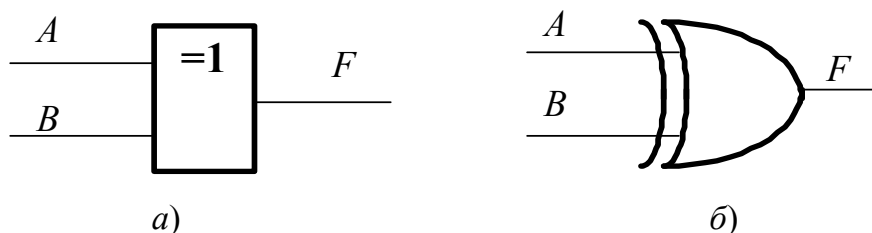


Рис. 1.4. УГО логического элемента, реализующего функцию ИСКЛЮЧАЮЩЕЕ ИЛИ: *a* – в отечественных схемах; *б* – обозначение в соответствии с американской системой MILSPEC 806B

Следует заметить, что данные элементы очень широко применяются в цифровой схемотехнике.

Три вышеописанные логические функции И, ИЛИ, НЕ, с помощью которых можно получить все остальные логические функции, называют булевым базисом.

### ***Инвертирующие базисы, отрицательная логика***

Логика называется *положительной*, если высокий потенциал отображает единицу, а низкий – ноль. Если наоборот, высокий потенциал отображает ноль, а низкий – единицу, то логика называется *отрицательной*. Данное правило называют логическим соглашением.

Самым важным следствием применения отрицательной логики является то, что при переходе от положительной логики к отрицательной функция И превращается в ИЛИ и наоборот. Благодаря этому переходу от И к ИЛИ и удастся с помощью одностипных элементов инвертирующего базиса получать все остальные логические функции. Об этом говорят два постулата де Моргана:  $\overline{AB} = \overline{A} + \overline{B}$ ;  $\overline{A + B} = \overline{A} \overline{B}$ .

Если логический элемент в положительной логике реализует функцию И, то в отрицательной логике этот же элемент реализует функцию ИЛИ, и наоборот, логический элемент ИЛИ положительной логики реализует функцию И в отрицательной логике. Иногда объединяют две булевы функции (при этом одной из них является НЕ) и получившийся логический элемент считают базовым для получения всех остальных логических функций.

Элемент И–НЕ называют также: штрих Шеффера (*Sheffer stroke*), *NAND* (сокращение от *NOT AND*). Алгебраическая запись функции И–НЕ:  $F = \overline{AB}$ .

Условное графическое обозначение логического элемента, реализующего функцию И–НЕ, изображено на рис. 1.5.

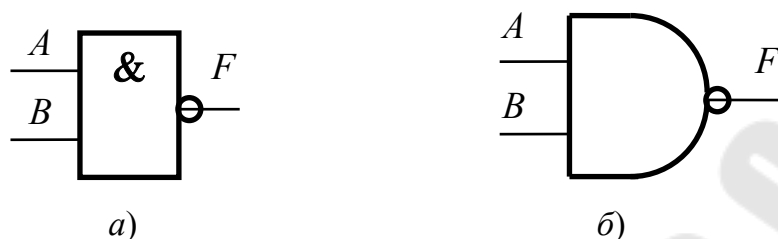


Рис. 1.5. УГО логического элемента, реализующего функцию И–НЕ:  
*a* – в отечественных схемах; *б* – обозначение в соответствии с американской системой MILSPEC 806B

Элемент ИЛИ–НЕ называют также: стрелка Пирса (Pierce arrow), NOR (сокращение от NOT OR). Алгебраическая запись функции ИЛИ–НЕ:  $F = A \downarrow B$ ;  $F = \overline{A + B}$ .

Условное графическое обозначение логического элемента, реализующего функцию И–НЕ, изображено на рис. 1.6.

Применение наряду с положительной логикой и отрицательной логики позволяет любое сложное логическое преобразование выполнить с применением только логических элементов И–НЕ или только ИЛИ–НЕ.

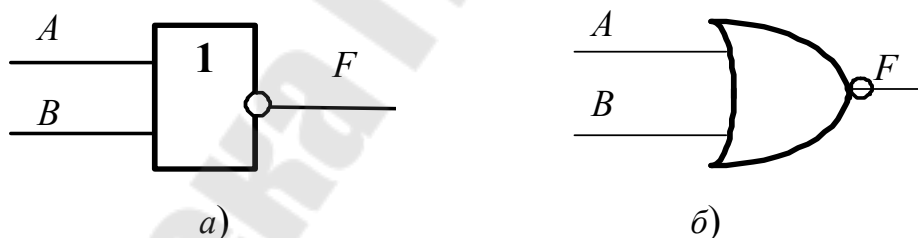
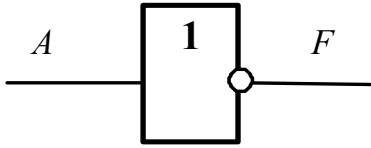
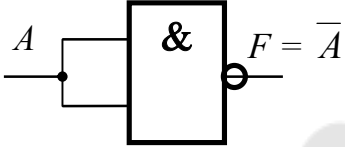
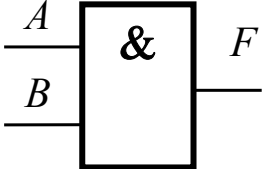
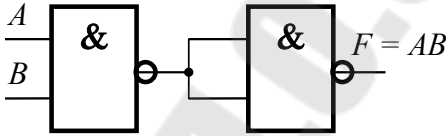
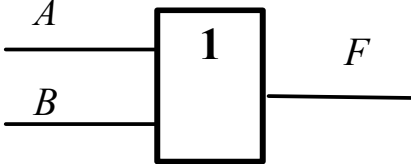
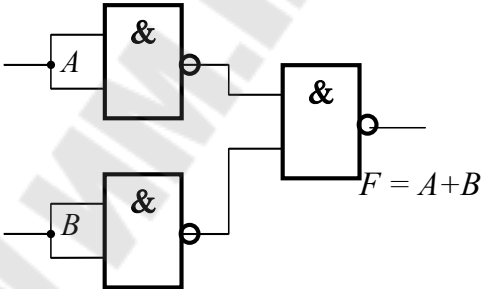


Рис. 1.6. УГО логического элемента, реализующего функцию И–НЕ:  
*a* – в отечественных схемах; *б* – обозначение в соответствии с американской системой MILSPEC 806B

В табл. 1.1 представлены схемы, реализующие функции с помощью только логических элементов 2И–НЕ.

Схемы, реализующие функции с помощью логических элементов 2И–НЕ

НЕ		
И		
ИЛИ		

### Порядок выполнения работы

**Меры безопасности:** включать установку IDL-800 разрешается после того, как преподаватель проверит собранную для исследования схему. Запрещается оставлять включенную установку без присмотра. По окончании работы необходимо выключить установку, сложить соединительные провода и обесточить рабочее место.

#### 1. Исследование логических элементов И

Чтобы определить нахождение первой ножки, нужно найти на корпусе «ключик». Порядок нумерации выводов микросхем приведен на рис. 1.7 (вид сверху). Нумерация начинается слева от ключа и продолжается далее против хода часовой стрелки.



Ключ бывает такой  
или такой

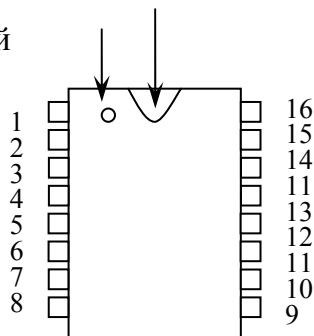


Рис. 1.7. Нумерация выводов микросхем (вид сверху)

### 1.1. Двухвходовый элемент И

Разместить ИС 1533ЛИ1 на наборной панели IDL-800.

Вывод 14 ИС соединить с источником питания  $+5V$ , а его вывод 7 – с общей шиной установки (рис. 1.8, а). Собрать схему, как показано на рис. 1.8, б.

Изменяя состояния входов  $A$  и  $B$  с помощью переключателей  $SW$ , заполнить таблицу истинности логического элемента 2И (табл. 1.2).

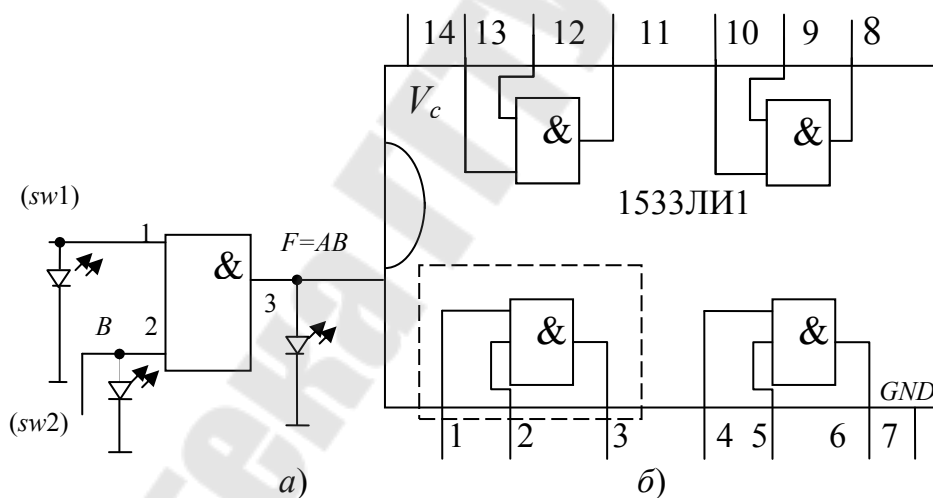


Рис. 1.8. Двухвходовый логический элемент И:  
а – схема исследования; б – внутреннее строение ИС 1533ЛИ1

Таблица 1.2

Таблица истинности логического элемента 2И

$A$	$B$	$F = AB$
0	0	
0	1	

$A$	$B$	$F = AB$
1	0	
1	1	

### 1.2. Трехвходовый элемент И

Используя двухвходовые элементы И, собрать трехвходовый элемент И, реализующий функцию  $F = ABC$ , как показано на рис. 1.9.

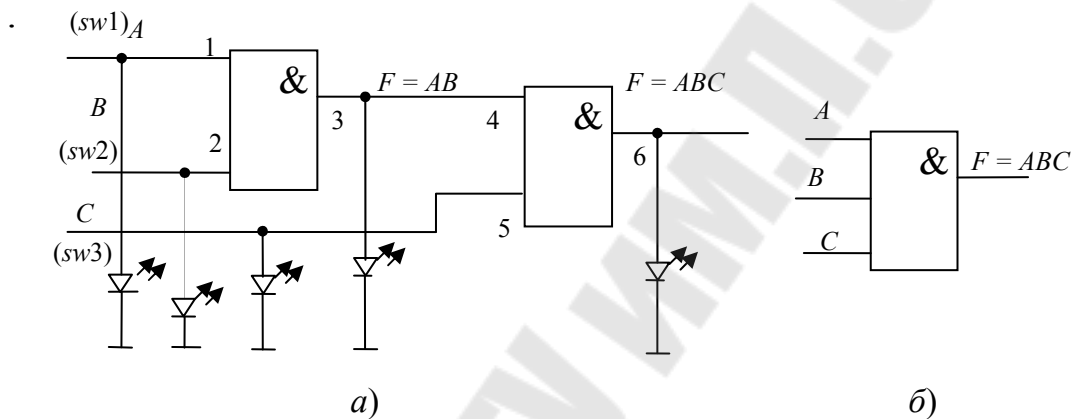


Рис. 1.9. Трехвходовый элемент И:  
а – схема исследования; б – его УГО

Изменяя состояния входов  $A$ ,  $B$  и  $C$  (комбинации от трех нулей до трех единиц), исследовать схему работы и заполнить таблицу истинности логического элемента 3И (табл. 1.3).

Таблица 1.3

Таблица истинности логического элемента 3И

Входы			Выходы	
$A$	$B$	$C$	$F = AB$	$F = ABC$
0	0	0		
0	0	1		
...	...	...	...	...
1	1	0		
1	1	1		

## 2. Исследование логических элементов ИЛИ

### 2.1. Двухвходовый элемент ИЛИ

Разместить ИС 1533ЛЛ1 на наборной панели IDL-800. Вывод 14 ИС соединить с источником питания  $+5V$ , а его вывод 7 – с общей шиной установки (рис. 1.10, а). Собрать схему, как показано на рис. 1.10, б).

Изменяя состояния входов  $A$  и  $B$  с помощью переключателей  $SW$ , заполнить таблицу истинности (табл. 1.4) логического элемента 2ИЛИ.

Таблица 1.4

Таблица истинности логического элемента 2ИЛИ

$A$	$B$	$F = A + B$
0	0	
0	1	
1	0	
1	1	

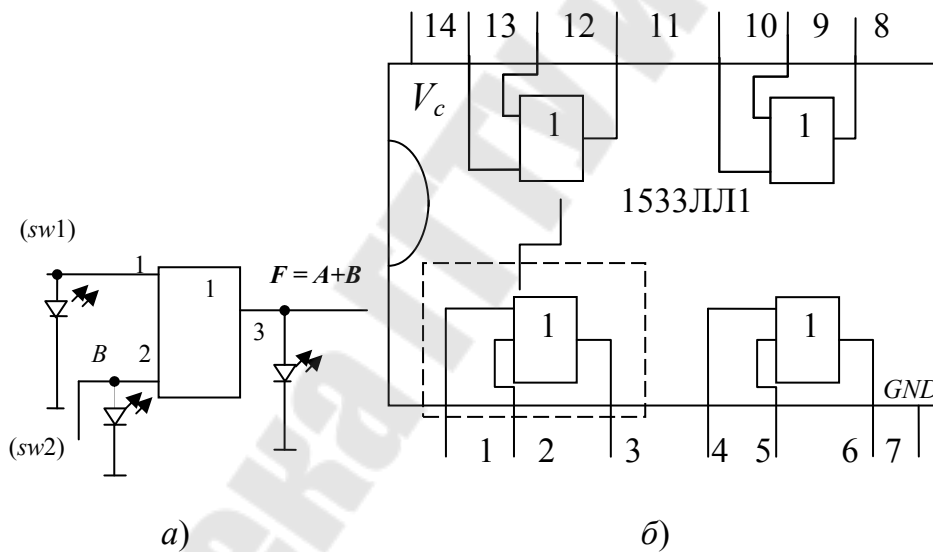


Рис. 1.10. Двухвходовый элемент ИЛИ:  
а – схема исследования; б – внутреннее строение логического элемента 2ИЛИ – ИС 1533ЛЛ1

### 2.2. Трехвходовый элемент ИЛИ

Используя двухвходовые элементы И, собрать трехвходовый элемент И, реализующий функцию  $F = A + B + C$ , как показано на рис. 1.11. Изменяя состояния входов  $A$ ,  $B$  и  $C$ , исследовать схему работы и заполнить таблицу истинности логического элемента 3ИЛИ (табл. 1.5).

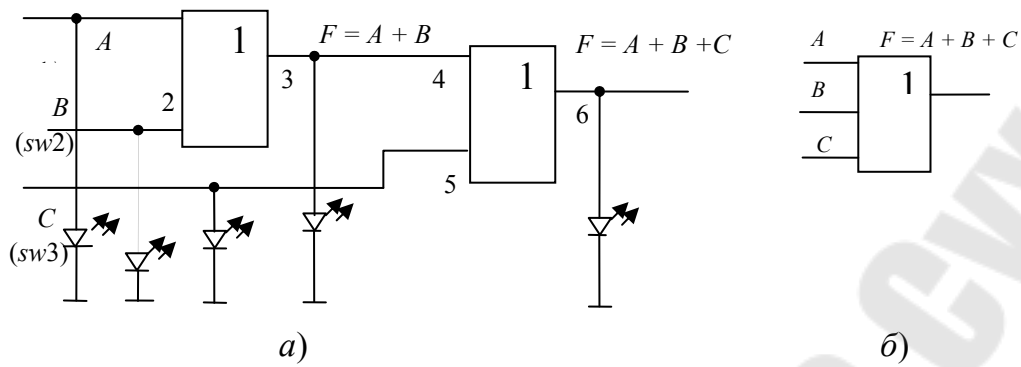


Рис. 1.11. Трехвходовый элемент ИЛИ:  
*a* – схема исследования; *б* – его УГО

### 3. Исследование инвертора

Разместить ИС 1533ЛН1 на наборной панели IDL-800. Вывод 14 ИС соединить с источником питания  $+5V$ , а его вывод 7 – с общей шиной установки (рис. 1.12, *a*). Собрать схему, как показано на рис. 1.12, *б*.

Таблица 1.5

Таблица истинности логического элемента ЗИЛИ

Входы			Выходы	
<i>A</i>	<i>B</i>	<i>C</i>	$F = A + B$	$F = A + B + C$
0	0	0		
0	0	1		
–	–	–	–	–
1	1	0		
1	1	1		

Изменяя состояния входа *A* с помощью переключателей *SW*, заполнить таблицу истинности инвертора.

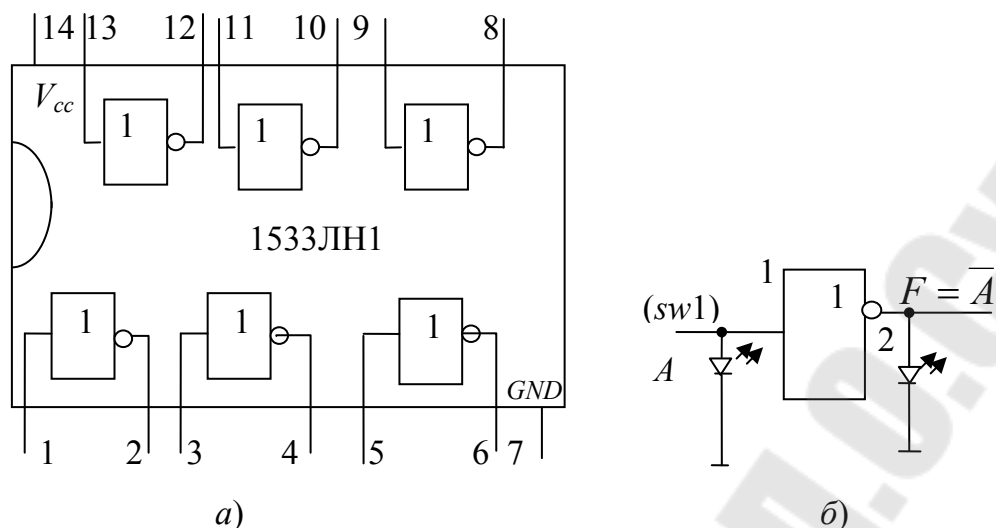


Рис. 1.12. Инвертор 1533ЛН1:  
а – схема исследования; б – внутреннее строение

#### 4. Исследование логических элементов И-НЕ

##### 4.1. Двухвходовый элемент И-НЕ

Разместить ИС 1533ЛА3 на наборной панели IDL-800. Вывод 14 ИС соединить с источником питания  $+5V$ , а его вывод 7 – с общей шиной установки (рис. 1.13, а). Собрать схему, как показано на рис. 1.13, б.

Изменяя состояния входов  $A$  и  $B$  с помощью переключателей  $SW$ , заполнить таблицу истинности логического элемента 2И-НЕ аналогично таблице истинности логического элемента 2И.

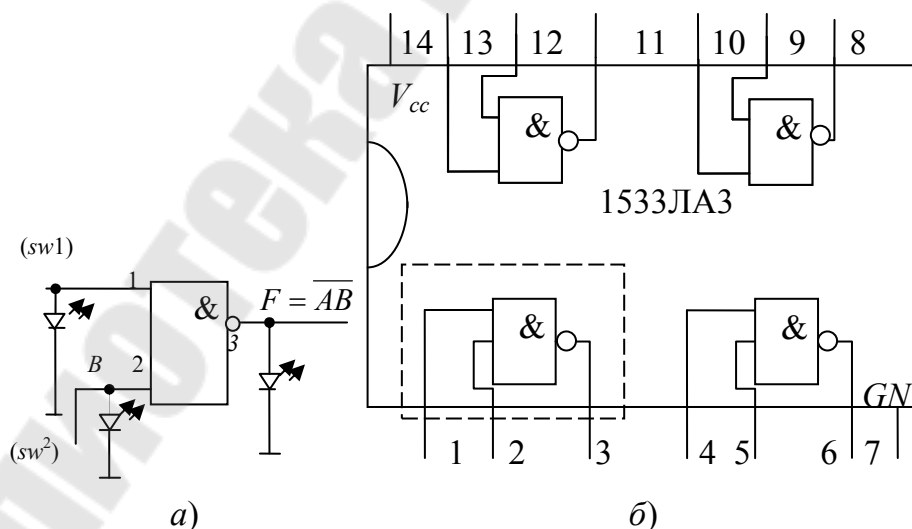


Рис. 1.13. Двухвходовый элемент И-НЕ: а – схема исследования;  
б – внутреннее строение логического элемента 1533ЛА3

#### 4.2. Трехвходовый элемент И-НЕ

Используя двухвходовые элементы И-НЕ, собрать трехвходовый элемент И-НЕ, реализующий функцию  $F = \overline{ABC}$ , как показано на рис. 1.14.

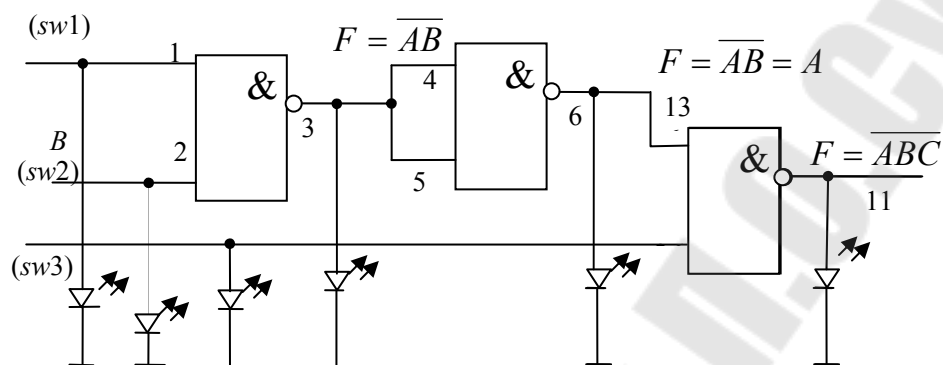


Рис. 1.14. Схема исследования трехвходового элемента И-НЕ

Изменяя состояния входов  $A$ ,  $B$  и  $C$ , исследовать схему работы и заполнить таблицу истинности логического элемента 3И-НЕ (табл. 1.6).

Таблица 1.6

Таблица истинности логического элемента 3И-НЕ

Вход			Выход		
$A$	$B$	$C$	$F = \overline{AB}$	$F = AB$	$F = \overline{ABC}$
0	0	0			
–	–	–	–	–	–
1	1	1			

#### 4.3. Реализация функции ИЛИ при помощи логических элементов И-НЕ

Используя двухвходовые элементы И-НЕ, собрать схему, реализующую операцию ИЛИ, как показано на рис. 1.15.

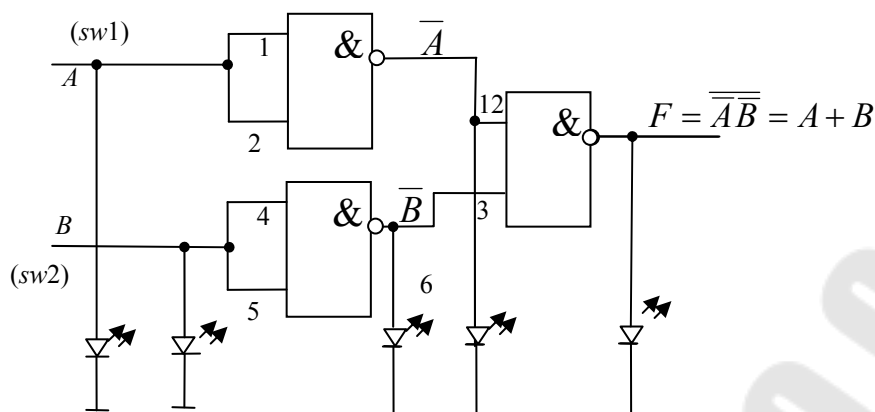


Рис. 1.15. Реализация функции 2ИЛИ при помощи элементов 2И-НЕ

Изменяя состояния входов  $A$  и  $B$ , исследовать схему работы и заполнить таблицу истинности полученного логического элемента 2И-НЕ (табл. 1.7).

Таблица 1.7

Таблица истинности логического элемента 2ИЛИ

Входы		Выходы		
$A$	$B$	$\bar{A}$	$\bar{B}$	$F = A + B$

## 5. Исследование логических элементов ИЛИ-НЕ

### 5.1. Двухвходовый элемент ИЛИ-НЕ

Разместить ИС 1533ЛЕ1 на наборной панели IDL-800. Вывод 14 ИС соединить с источником питания  $+5V$ , а его вывод 7 – с общей шиной установки (рис. 1.16, а). Собрать схему, как показано на рис. 1.16, б.

Изменяя состояния входов  $A$  и  $B$  с помощью переключателей  $SW$ , заполнить таблицу истинности логического элемента 2ИЛИ-НЕ.

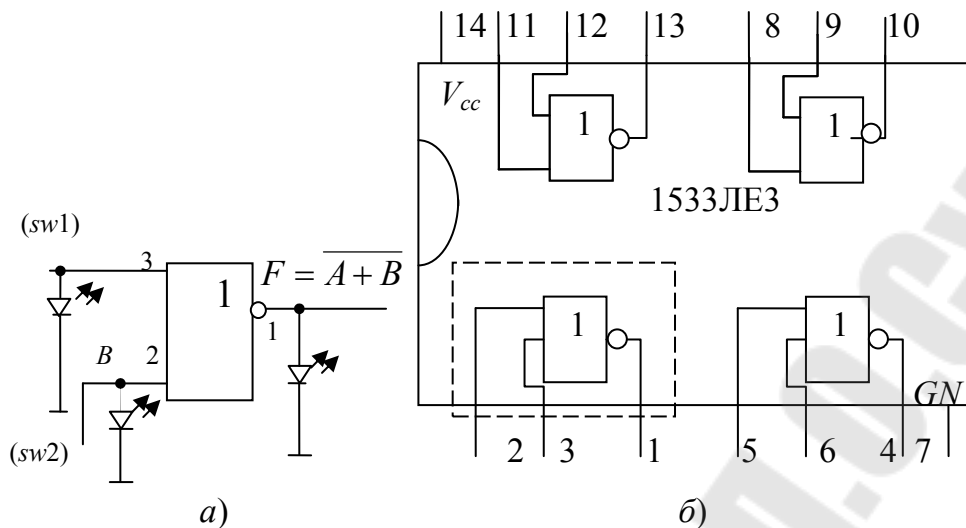


Рис. 1.16. Двухвходовый элемент ИЛИ-НЕ:  
 а – схема исследования двухвходового элемента ИЛИ-НЕ;  
 б – внутреннее строение логического элемента 1533ЛЕ3

### 5.2. Трехвходовый элемент ИЛИ-НЕ

Используя двухвходовые элементы ИЛИ-НЕ, собрать трехвходовый элемент ИЛИ-НЕ, реализующий функцию  $F = \overline{A+B+C}$ , как показано на рис. 1.17.

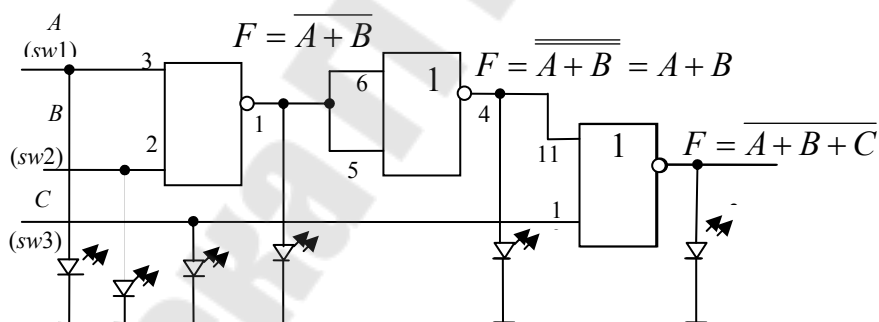


Рис. 1.17. Схема исследования трехвходового элемента И-НЕ

### 5.3. Логический элемент ЗИЛИ-НЕ

Изменяя состояния входов  $A$ ,  $B$  и  $C$ , исследовать схему работы и заполнить таблицу истинности логического элемента ЗИЛИ-НЕ (табл. 1.8).



Таблица истинности логического элемента ЗИЛИ-НЕ

Входы		Выходы			
$A$	$B$	$A$	$F = \overline{A+B}$	$F = A+B$	$F = \overline{A+B+C}$

#### 5.4. Реализация функции И при помощи логических элементов ИЛИ-НЕ

Используя двухвходовые элементы ИЛИ-НЕ, собрать схему, реализующую операцию И, как показано на рис. 1.18.

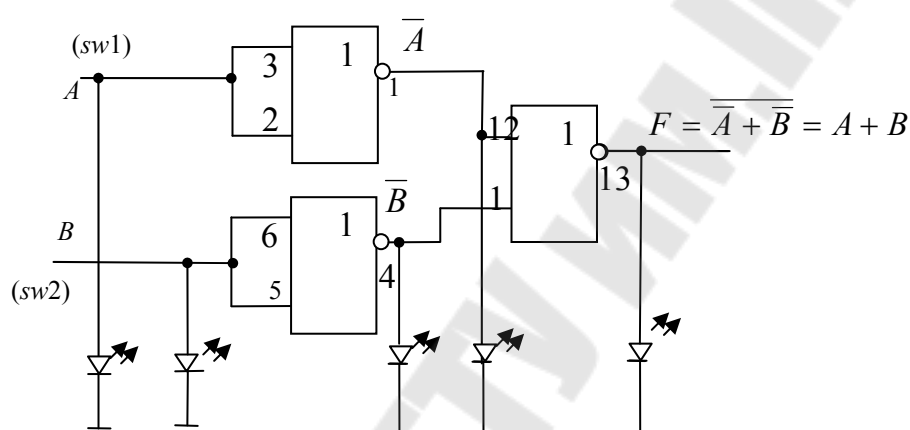


Рис. 1.18. Реализация функции И при помощи элементов ИЛИ-НЕ

Изменяя состояния входов  $A$  и  $B$ , исследовать схему работы и заполнить таблицу истинности полученного логического элемента ЗИЛИ-НЕ.

## 6. Исследование логических элементов ИСКЛЮЧАЮЩЕЕ ИЛИ

### 6.1. Двухвходовый элемент ИСКЛЮЧАЮЩЕЕ ИЛИ

Разместить ИС 1533ЛП5 на наборной панели IDL-800. Вывод 14 ИС соединить с источником питания  $+5V$ , а его вывод 7 – с общей шиной установки (рис. 1.19, а). Собрать схему, как показано на рис. 1.19, б.

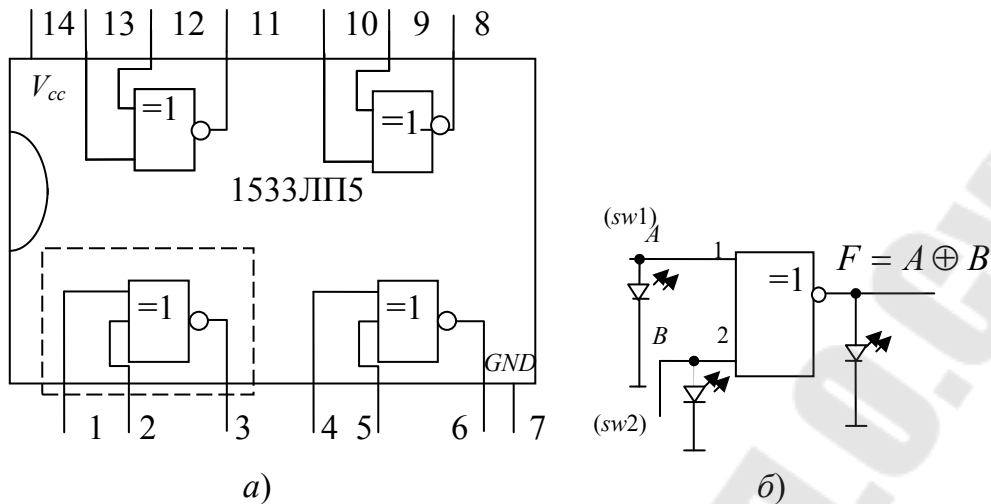


Рис. 1.19. Двухвходовый элемент ИСКЛЮЧАЮЩЕЕ ИЛИ:  
 а – схема исследования; б – внутреннее строение  
 логического элемента 1533ЛПИ5

Изменяя состояния входов  $A$  и  $B$  с помощью переключателей  $SW$ , заполнить таблицу истинности логического элемента ИСКЛЮЧАЮЩЕЕ ИЛИ.

### 6.2. Трехвходовый элемент ИСКЛЮЧАЮЩЕЕ ИЛИ

Используя двухвходовые элементы ИСКЛЮЧАЮЩЕЕ ИЛИ, собрать трехвходовый элемент ИСКЛЮЧАЮЩЕЕ ИЛИ, реализующий функцию  $F = A \oplus B \oplus C$ , как показано на рис. 1.20.

Изменяя состояния входов  $A$ ,  $B$  и  $C$ , исследовать схему работы и заполнить таблицу истинности трехвходового логического элемента ИСКЛЮЧАЮЩЕЕ ИЛИ (табл. 1.9).

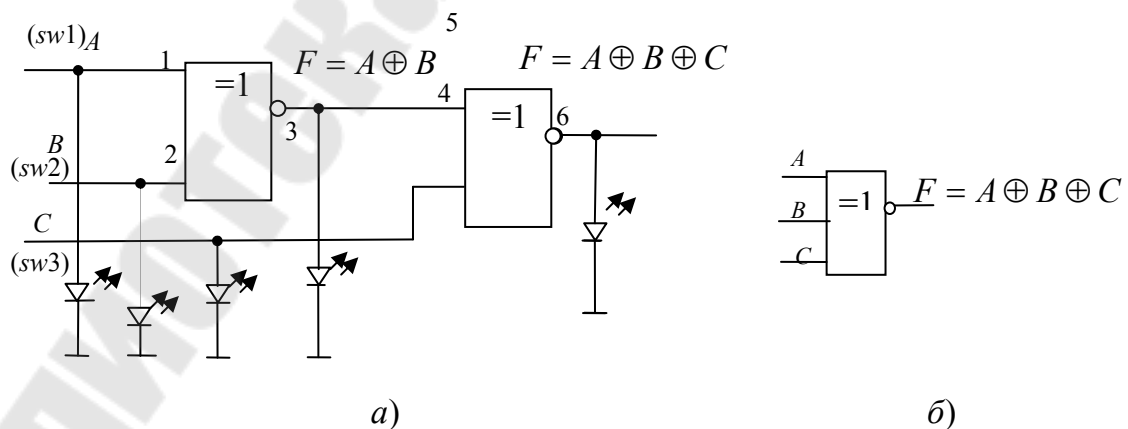


Рис. 1.20. Трехвходовый элемент ИСКЛЮЧАЮЩЕЕ ИЛИ:  
 а – схема исследования трехвходового элемента;  
 б – его условно-графическое обозначение

Таблица истинности логического элемента ИСКЛЮЧАЮЩЕЕ ИЛИ

Входы		Выходы	
$A$	$B$	$A$	$F = A \oplus B$
0	0	0	0

### Содержание отчета

1. Наименование и цель работы.
2. Исследуемые схемы логических функций с результатами исследований в виде таблиц истинности.
3. Выводы по результатам исследований.

### Контрольные вопросы

1. Таблицы истинности основных логических функций.
2. Положительная и отрицательная логика. Дополнения.
3. Нарисовать для логического элемента, заданного преподавателем, его условное графическое обозначение.
4. Обосновать минимальное количество типов логических элементов для реализации основных логических функций.
5. Схема логического элемента ИСКЛЮЧАЮЩЕЕ ИЛИ. Обосновать с помощью схемы таблицу истинности.

## **Лабораторная работа № 2**

### **Синтез комбинационных логических схем по заданной логической функции**

**Цель работы:** изучение принципа построения комбинационных логических схем по заданной логической функции с использованием имеющегося базиса логических элементов.

#### **Используемое оборудование:**

- универсальная лабораторная установка IDL-800;
- четыре логических элемента 2И-НЕ – ИС 1533ЛА3 (74ALS00);
- четыре логических элемента 2ИЛИ-НЕ – ИС 1533ЛЕ1 (74ALS02);
- три логических элемента 3И-НЕ – ИС 1533ЛА3 (74ALS04);
- два логических элемента 4И-НЕ – ИС 1533ЛА6 (74ALS08);
- четыре логических элемента 2ИЛИ – ИС 1533ЛЛ1 (74ALS32).

#### **Основные теоретические сведения**

##### *Дизъюнктивные и конъюнктивные формы записи функций алгебры логики*

*Совершенной дизъюнктивной нормальной формой (СДНФ) называют наиболее полную форму записи логического выражения. Эта форма записи представляет собой сумму, каждое слагаемое которой является произведением всех входных аргументов или их инверсий, например:  $F = \overline{A}\overline{B}\overline{C} + \overline{A}B\overline{C} + A\overline{B}\overline{C} + ABC$ .*

СДНФ является избыточной, но логические функции, записанные в СДНФ, легко сравнивать между собой, их удобно преобразовывать в таблицы истинности и составлять по ним карты Карно. Булево выражение, полученное из таблицы истинности логической функции, имеет совершенную дизъюнктивную нормальную форму.

В некоторых случаях более удобной формой записи логического выражения является *совершенная конъюнктивная нормальная форма (СКНФ)*. Это произведение сомножителей, каждый из которых является суммой всех входных аргументов или их инверсий, например:  $F = (\overline{A} + B + \overline{C})(\overline{A} + B + C)(\overline{A} + \overline{B} + C)(A + B + C)$ . Так же, как и СДНФ, СКНФ является явно избыточной.

## Преобразование таблицы истинности в булево выражение

Допустим, имеется логическая функция  $F$  для трех переменных:  $A$ ,  $B$  и  $C$ , заданная в виде следующей таблицы истинности (табл. 2.1).

Из всех возможных восьми комбинаций входных переменных  $A$ ,  $B$  и  $C$  данная функция  $F$  равна единице только для тех четырех комбинаций, которые записаны в виде логических произведений  $P_0$ ,  $P_2$ ,  $P_3$  и  $P_7$  в правой части таблицы, в разделе «Примечания». При остальных наборах входных переменных функция  $F$  равна нулю.

Смысл каждого булева выражения в том, чтобы показать, при каких сочетаниях входных переменных или их инверсий заданная функция  $F$  равна единице.

Таблица 2.1

**Пример логической функции  $F$  для трех переменных  $A$ ,  $B$  и  $C$ ,  
заданной в виде таблицы истинности**

Номер	$A$	$B$	$C$	$F$	Примечания
0	0	0	0	1	$P_0 = \overline{ABC}$
1	0	0	1	0	–
2	0	1	0	1	$P_2 = \overline{A}BC$
3	0	1	1	1	$P_3 = \overline{A}BC$
4	1	0	0	0	–
5	1	0	1	0	–
6	1	1	0	0	–
7	1	1	1	1	$P_7 = ABC$

Поскольку функция будет иметь такое значение при любом из наборов  $P_0$ ,  $P_2$ ,  $P_3$ ,  $P_7$  независимо друг от друга, то их можно соединить между собой знаком ИЛИ, логическим сложением:  $F = P_0 + P_2 + P_3 + P_7$ .

Каждый из наборов  $P_0$ ,  $P_2$ ,  $P_3$ ,  $P_7$  является таким сочетанием входных переменных или их инверсий, которые только при совместном их воздействии обеспечивают единичное состояние выходной функции.

Следовательно, каждый такой набор состоит из всех входных переменных или их инверсий, связанных между собой функцией И, логическим умножением:  $P_0 = \overline{ABC}$ ;  $P_2 = \overline{A}BC$ ;  $P_3 = \overline{A}BC$ ;  $P_7 = ABC$ .

Исходя из этого, получаем результирующее выражение:

$$F = \overline{A}\overline{B}\overline{C} + \overline{A}B\overline{C} + A\overline{B}\overline{C} + ABC.$$

Как можно заметить, это выражение записано в СДНФ.

### *Минимизация логических функций по картам Карно*

Для минимизации логических функций очень удобно пользоваться картами Карно или очень схожими с ними диаграммами Вейча.

Карта Карно изображает в виде графических квадратов (клеток) все возможные комбинации переменных, причем переменные, определяющие координаты клеток карты, размещают так, чтобы при переходе из одной клетки в соседнюю, как по горизонтали, так и по вертикали, изменялась только одна переменная.

Таблица истинности для четырех переменных включает 16 строк, следовательно, карта Карно должна состоять из 16 клеток (рис. 2.1).

Если требуется получить карту Карно для какой-либо функции, сначала надо записать эту функцию в СДНФ, – в совершенной дизъюнктивно нормальной форме, или в виде таблицы истинности. Каждое слагаемое булева выражения в СДНФ, или каждая единица в столбце функции таблицы истинности, задается на карте Карно единицей в соответствующей клетке. Координаты этой клетки содержат те же входные переменные и их инверсии, что и данное слагаемое СДНФ булева выражения (или данная строка таблицы истинности).

У карты Карно для четырех переменных клетки крайнего левого столбца должны рассматриваться как соседние для клеток крайнего правого столбца, а клетки верхней строки – как соседние для клеток нижней строки. Другими словами можно сказать, что эта карта расположена на поверхности цилиндра (склеили правый край карты с левым), изогнутого и растянутого так, что его верхний срез соединяется с нижним срезом; при этом цилиндр превращается в тор.

Правила упрощения заполненной карты Карно заключаются в следующем:

- соседние две, четыре, восемь или другое число единиц, равное степени двойки, обводят общим контуром;
- контур должен быть прямоугольным без изгибов или наклонов;
- каждый контур превращает все входящие в него единицы в одну, т. е. объединенные таким образом слагаемые СДНФ булева выражения дают одно слагаемое в упрощенном выражении;
- те входные переменные, которые входят в координаты данного контура совместно со своими инверсиями, исключаются из слагаемого, которое дает этот контур в упрощенном выражении.

Рассмотрим пример упрощения булевых выражений с помощью карты Карно для 4-х переменных. Пусть задана функция:

$$F = \overline{A}\overline{B}\overline{C}\overline{D}_1 + \overline{A}B\overline{C}\overline{D}_2 + A\overline{B}\overline{C}\overline{D}_3 + \overline{A}\overline{B}C\overline{D}_4 + \overline{A}B\overline{C}D_5.$$

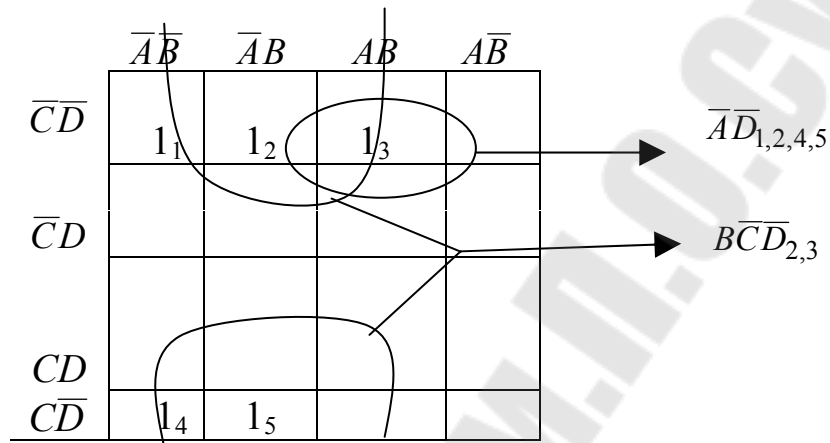


Рис. 2.1. Пример минимизации булевой функции  $F$  с помощью карты Карно для четырех переменных

В этом примере минимизации булевой функции  $F$  (рис. 2.1) контур из двух единиц  $1_2$  и  $1_3$ , соответствующих второму и третьему слагаемым в исходном булевом выражении, дает возможность опустить  $A$  и  $\overline{A}$ . После этого в нем остается произведение  $B\overline{C}\overline{D}$ . В контуре из четырех единиц  $1_1, 1_2, 1_4$  и  $1_5$ , соответствующие другим четырем слагаемым из исходного булева выражения, попарно опускаются  $B$  и  $\overline{B}$ ,  $C$  и  $\overline{C}$ , так что в результате этого верхний контур дает произведение  $\overline{A}\overline{D}$ . Карта Карно представляется в данном случае свернутой в цилиндр, в котором верхний край совмещается с нижним. Полученное упрощенное выражение:  $F = \overline{A}\overline{D} + B\overline{C}\overline{D}$ . Этот пример показывает также, что контуры могут накладываться друг на друга (сколько угодно раз).

### **Синтез комбинационных логических схем по заданным булевым выражениям**

Как правило, построение и расчет любой схемы осуществляется начиная с ее выхода. Допустим, задано булево выражение:  $F = \overline{B}A + B\overline{A} + C\overline{B}$ .

1. Первый этап: выполняется логическое сложение, логическая операция ИЛИ, считая входными переменными функции  $\overline{B}A$ ,  $B\overline{A}$  и  $C\overline{B}$  (рис. 2.2).

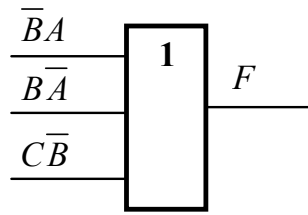


Рис. 2.2. Первый этап построения комбинационных логических схем по заданному булеву выражению

2. Второй этап: к входам элемента ИЛИ подключаются логические элементы И, входными переменными которых являются уже  $A$ ,  $B$ ,  $C$  и их инверсии (рис. 2.3).

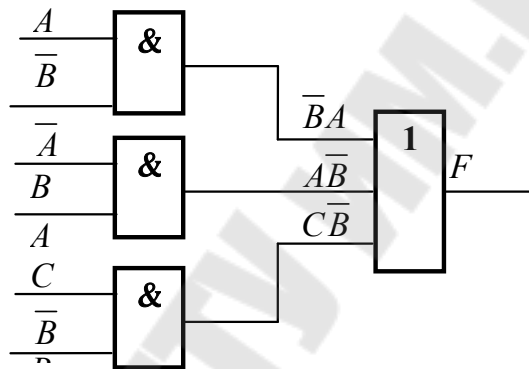


Рис. 2.3. Второй этап построения комбинационных логических схем по заданному булеву выражению

3. Третий этап: для получения инверсий  $\overline{A}$  и  $\overline{B}$  на соответствующих входах ставят инверторы (рис. 2.4).

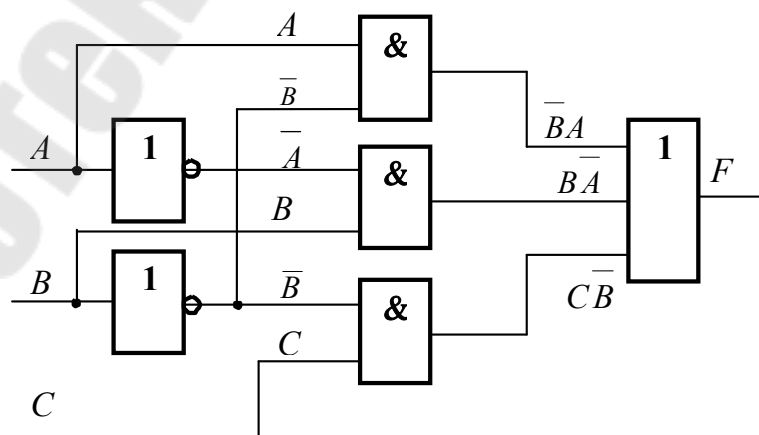


Рис. 2.4. Второй этап построения комбинационных логических схем по заданному булеву выражению



Данное построение основано на следующей особенности: поскольку значениями логических функций могут быть только нули и единицы, то любые логические функции могут быть представлены как аргументы других более сложных функций.

Таким образом, построение комбинационной логической схемы осуществляется с выхода к входу.

### ***Особенности построения логических схем в инвертирующих базисах***

Первой особенностью построения логических схем в инвертирующих базисах считается непрямая зависимость между простотой булева выражения и минимальностью соответствующей ему логической схемы. Другими словами, самое минимизированное булево выражение не всегда дает схему, минимальную по количеству инвертирующих логических элементов.

Вторая особенность построения логических схем в инвертирующих базисах приводится без доказательства.

Если в произвольной цифровой схеме (комбинационной) произвести следующую последовательность действий:

- проинвертировать все входные и выходные сигналы;
- все элементы И заменить на ИЛИ, а ИЛИ – на И, то реализуемая схемой функция не изменится.

### **Порядок выполнения работы**

**Меры безопасности:** включать установку IDL-800 разрешается после того, как преподаватель проверит собранную для исследования схему. Запрещается оставлять включенную установку без присмотра. По окончании работы необходимо выключить установку, сложить соединительные провода и обесточить рабочее место.

1. Составить таблицу истинности для четырех переменных ( $ABCD$ ) таким образом, чтобы при совпадении кодовой комбинации с номером каждого члена бригады по журналу группы, в двоичном коде и коде Грея функция принимала значение «1».

**Примечания:** 1. При номере члена бригады больше 15 в таблицу записывается значение функции «1» при номере, соответствующем разности номера члена и числа 15. – 2. Значение функции «1» в составленной таблице должно встречаться не менее четырех раз. Если это условие не удалось выполнить, то следует дописать «1» при  $N = 0$ , а также, при необходимости, рядом с номером одного из членов бригады в сторону увеличения номеров.

Таблица кода Грея

Десятичное число	Число в коде Грея
0	0000
1	0001
2	0011
3	0010
4	0110
5	0111
6	0101
7	0100
8	1100
9	1101
10	1111
11	1110
12	1010
13	1011
14	1001
15	1000

2. По составленной таблице истинности записать булево выражение.
3. Упростить полученное выражение по методу карт Карно.
4. Посчитать, сколько раз встречается в упрощенном булевом выражении каждая из переменных ( $ABCD$ ) и ее дополнение. Составить комбинационную логическую схему.
5. Составить комбинационную логическую схему в базисе И-НЕ (1533ЛИЗ). Собрать полученную схему на наборной панели IDL-800. Вывод 14 ИС соединить с источником питания  $+5V$ , а его вывод 7 – с общей шиной установки.
6. Изменяя состояния входных переменных  $A, B, C, D$  с помощью переключателей  $SW0-SW3$ , а выход  $F$  контролируя светодиодом, убедиться в работоспособности собранной схемы. Проверить полученную таблицу истинности и показать преподавателю.
7. Составить комбинационную логическую схему в базисе ИЛИ-НЕ (ИС 1533ЛЕ1). Собрать полученную схему на наборной панели IDL-800. Вывод 14 ИС соединить с источником питания  $+5V$ , а его вывод 7 – с общей шиной установки.

8. Изменяя состояния входных переменных  $A$ ,  $B$ ,  $C$ ,  $D$  с помощью переключателей  $SW0$ – $SW3$ , а выход  $F$  контролируя светодиодом, убедиться в работоспособности собранной схемы. Проверить полученную таблицу истинности и показать преподавателю.

### Содержание отчета

1. Наименование и цель работы.
2. Перечень членов бригады с указанием их номера по журналу группы.
3. Таблица соответствия номеров по журналу группы и чисел в двоичном коде и коде Грея (с учетом примечания 1).
4. Булево выражение, карта Карно, упрощенное булево выражение.
5. Схема соединения логических элементов, реализующая упрощенное булево выражение.
6. Комбинационная логическая схема в базисе И-НЕ.
7. Комбинационная логическая схема в базисе ИЛИ-НЕ.
8. Выводы по результатам исследований.

### Контрольные вопросы

1. Законы булевой алгебры.
2. Постулаты де Моргана.
3. Минимизация логических функций алгебраическим способом.
4. Минимизация логических функций по карте Карно.
5. Составление комбинационных схем в базисе ИЛИ-НЕ по таблице истинности, заданной преподавателем.
6. Составление комбинационных схем в базисе И-НЕ по таблице истинности, заданной преподавателем.

## **Лабораторная работа № 3**

### **Исследование арифметических устройств**

**Цель работы:** изучение принципа действия и исследование свойств арифметических устройств: сумматора и вычитателя.

#### **Используемое оборудование:**

- универсальная лабораторная установка IDL-800;
- четыре логических элемента ИСКЛЮЧАЮЩИЕ ИЛИ – ИС 1533ЛП5;
- четыре логических элемента 2И – ИС 1533ЛИ1;
- четыре логических элемента 2ИЛИ – ИС 1533ЛЛ1;
- шесть инверторов – ИС 1533ЛН1;
- четырехразрядный сумматор – ИС К155ИМ3 или К555ИМ6.

#### **Основные теоретические сведения**

##### *Арифметические устройства*

*Арифметическими* называют устройства цифровой электроники, выполняющие арифметические действия с двоичными числами: сложения, вычитания, умножения и деления. К арифметическим устройствам относят также устройства, выполняющие специальные арифметические операции, как то: выявление четности заданных чисел (определение паритета) и сравнения двух чисел.

Особенность арифметических устройств состоит в том, что сигналам приписываются не логические, а арифметические значения 1 и 0 и действия над ними подчиняются законам двоичной арифметики. Хотя арифметические устройства оперируют с численными величинами, для описания их работы также удобно пользоваться таблицами истинности.

Важнейшая из арифметических операций – суммирование (сложение). Помимо прямого назначения она используется и при других операциях: вычитание – это сложение, в котором вычитаемое вводится в обратном или дополнительном коде, а умножение и деление – это последовательное сложение и вычитание.

Арифметические устройства выпускаются в виде готовых изделий в составе многих серий цифровых микросхем.

## Сумматор

Сумматором (от *SUMMER*, *SUMMATOR*) называется схема, предназначенная для сложения чисел в двоичном коде.

Сумматор двух одноразрядных слагаемых называется *полусумматором* и обозначается *HS HALF SUM* – половина суммы.

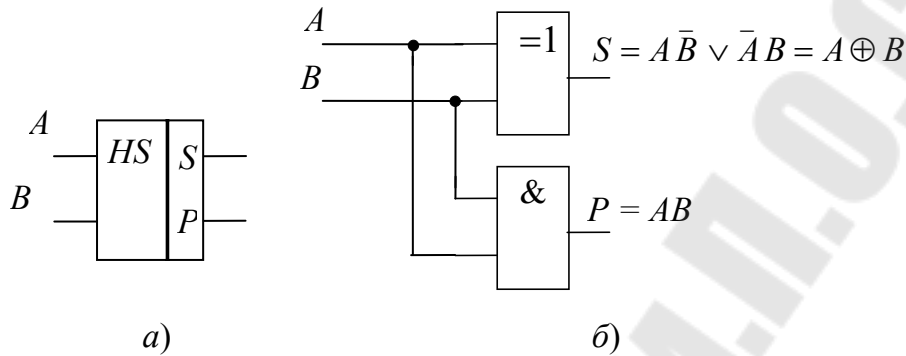


Рис. 3.1. Полусумматор: а – условное обозначение; б – схема

Выход  $P = AB$  называют *переносом* (переполнением) и иногда обозначают  $C$  или  $CR$  (от *CARRY* – перенос).

Чтобы не путать обозначения логических и арифметических действий при описании арифметических устройств, знаком «+» будем обозначать только арифметическую сумму, а знаком « $\vee$ » – логическую функцию ИЛИ, т. е. логическое сложение. Знак « $\oplus$ » будет означать суммирование по модулю 2 для арифметических двоичных переменных и функцию ИСКЛЮЧАЮЩИЕ ИЛИ для логических.

При сложении двух многоразрядных двоичных чисел, кроме двух входов слагаемых, в сумматоре каждого разряда должен быть предусмотрен еще вход для переноса из младшего разряда. Полусумматор имеет только два входа и пригоден для суммирования только одноразрядных слагаемых и не пригоден для суммирования всех других разрядов слагаемых.

Для сложения двух слагаемых любой разрядности с учетом переноса из младшего разряда предназначен одноразрядный полный сумматор или просто сумматор.

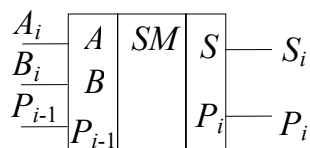


Рис. 3.2. Условное обозначение сумматора

Сумматор можно получить из двух полусумматоров (рис. 3.3). Первый полусумматор *HS1* складывает два слагаемых и выдает промежуточные сумму  $S'_i$  и бит переноса  $P'_i$ . Второй полусумматор *HS2* суммирует бит переноса предыдущего разряда  $P_{i-1}$  с промежуточной суммой  $S_{i-1}$ , в результате получается полная сумма  $S_i$ . Бит переноса получаем при участии двух полусумматоров и дополнительных логических элементов ИЛИ.

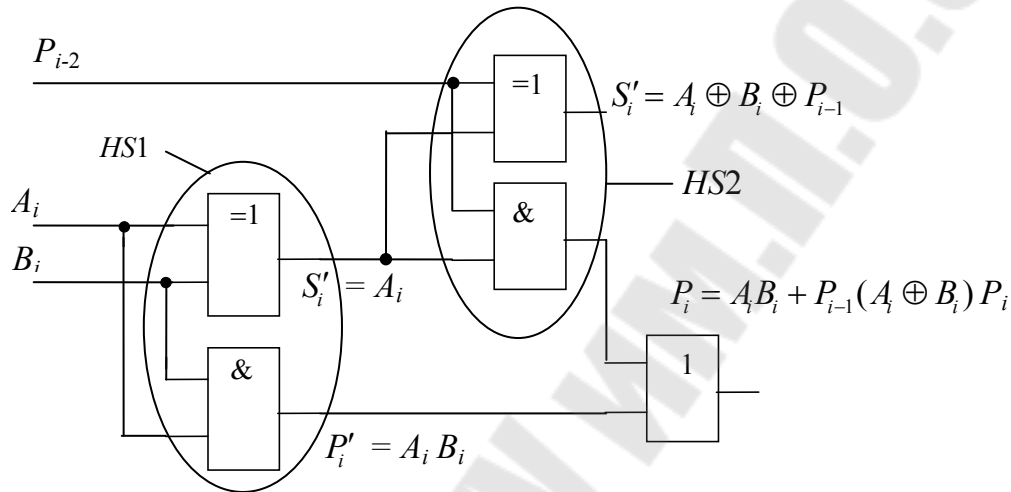


Рис. 3.3. Сумматор, полученный из двух полусумматоров

По схеме запишем булевы выражения для выходов  $S_i$  и  $P_i$ :

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

$$P_i = \bar{P}_{i-1} A_i B_i \vee P_{i-1} \bar{A}_i B_i \vee P_{i-1} A_i \bar{B}_i \vee P_{i-1} A_i B_i.$$

Булевы выражения для выходов  $S_i$  и  $P_i$  после минимизации и с учетом друг друга выглядят следующим образом:

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

$$P_i = P_{i-1}(A_i + B_i) + A_i B_i.$$

### **Последовательное и параллельное суммирование**

Одноразрядный сумматор может быть использован для суммирования многоразрядных двоичных чисел, если они представлены последовательным кодом (рис. 3.4), в котором младшие разряды следуют раньше старших. В этом случае сигнал с выхода переноса подается на вход переноса этого же сумматора через цепь задержки,

обеспечивающую хранение бита переноса на время одного такта следования импульсов входных цифровых слагаемых. Это самый медленный способ суммирования многоразрядных чисел, но он самый экономичный по аппаратурным затратам.

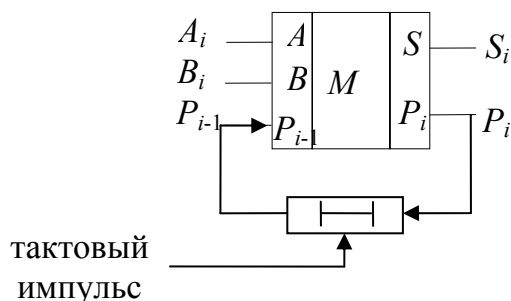


Рис. 3.4. Последовательное суммирование

В случае *параллельного суммирования* (рис. 3.5) число сумматоров должно быть равно числу разрядов суммируемых чисел. Быстродействие ограничено задержкой переноса ( $T_{зд.общ} = nt_{зд}P_i$ ), так как формирование сигналов суммы  $S_n$  и переноса  $P_n$  старшего разряда не может произойти до тех пор, пока сигнал переноса младшего разряда не распространится последовательно по всей цепочке.

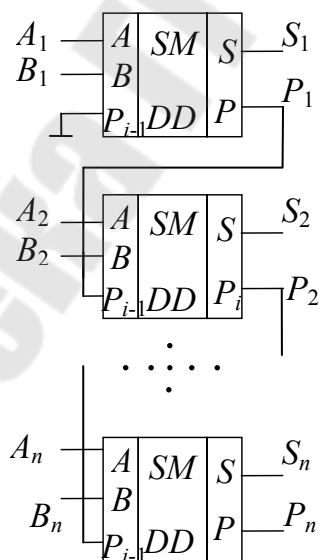


Рис. 3.5. Параллельное суммирование с последовательным переносом

### Вычитатель (субтрактор)

В устройствах дискретной техники операция вычитания обычно заменяется сложением уменьшаемого с вычитаемым, когда последнее представлено в *дополнительном* коде. Устройство, выполняющее данную функцию, называют вычитатель.

Это значит, что для вычитания числа  $B$  из числа  $A$  достаточно произвести их сложение, однако при этом вычитаемое  $B$  должно быть представлено в дополнительном коде.

*Отрицанием* называется логическая операция преобразования положительного числа в отрицательное или отрицательного числа в положительное. Если двоичные числа со знаком представляются в виде дополнительного кода, отрицание эквивалентно простому преобразованию в дополнительный код. Чтобы продемонстрировать это, преобразуем число  $+9$ . В двоичной форме вместе с битом знака это число будет выглядеть как  $01001$ .

Если преобразовать его в дополнительный код, получим  $10111$ . Ясно, что теперь это уже отрицательное число, так как знаковый бит, стоящий в первом разряде, равен  $1$ .

Действительно, число  $10111$  представляет собой  $-9$ , которое является отрицательным. Точно так же можно начинать и с числа  $-9$ , которое в двоичной форме будет выглядеть как  $10111$ . Если преобразовать это число в дополнительный код, получим  $01001$ , т. е. число  $+9$ . Порядок преобразований показан на рис. 3.6.

Двоичный код числа	→	0	1	0	0	1	= +9
Операция отрицания (обратный)	+	1	0	1	1	0	
Преобразование в дополнительный код						1	
Результат	→	1	0	1	1	1	= -9 <sub>доп</sub>
Снова осуществляем операцию отрицания	+	0	1	0	0	0	
Преобразование в дополнительный код						1	
Результат	→	0	1	0	0	1	= +9

Рис. 3.6. Пример перевода положительного числа в отрицательное и обратно



Если разность, полученная при таком способе вычитания, положительна, т. е.  $A > B$ , то результат будет представлен в прямом коде, а в разряде старше старшего, т. е. в  $(n+1)$ -м разряде, образуется единица.

Логическая схема, осуществляющая операцию вычитания однобитового числа  $B$  из однобитового числа  $A$ , называется *полувычитателем*.

На его выходах реализуются следующие булевы выражения:

$D = A\bar{B} \vee \bar{A}B = A \oplus B$  на выходе  $D$  – результат вычитания,  $C_i = \bar{A}_i B_i$  на выходе  $C$  – результат заема.

Схема полувычитателя на логических элементах приведена на рис. 3.7.

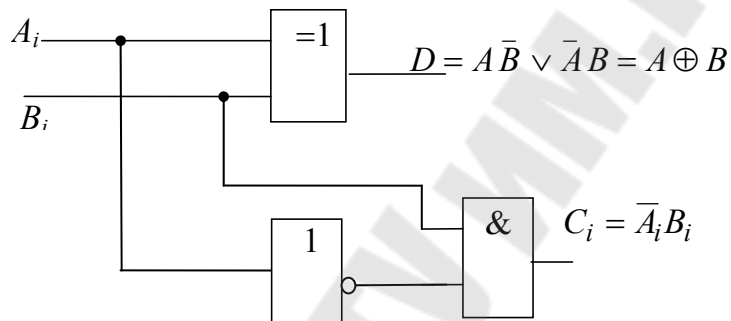


Рис. 3.7. Схема полувычитателя на логических элементах

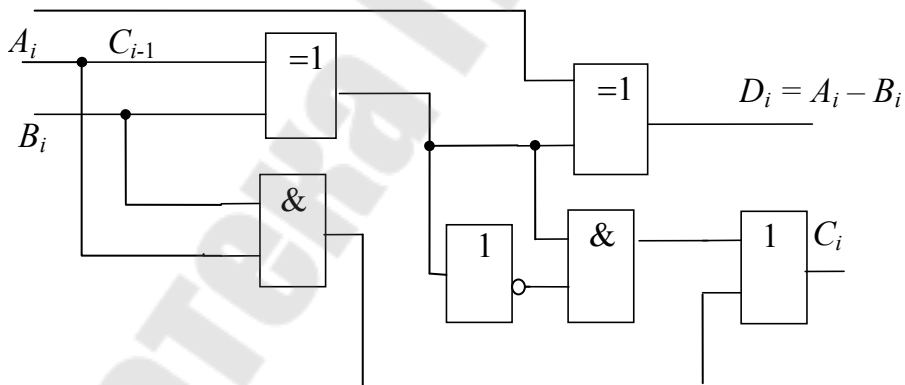


Рис. 3.8. Схема вычитателя на логических элементах

Для выполнения побитового вычитания многоразрядных чисел необходим полный вычитатель (рис. 3.8), где присутствует заем из старшего разряда. Такое устройство имеет три входа и два выхода, как и полный сумматор. Булевы выражения, реализуемые на его выходах, выглядят следующим образом:

$$C_i = \bar{A}_i(B_i \oplus C_{i-1}) + B_i C_{i-1}; \quad D_i = C_{i-1} \oplus A_i \oplus B_i.$$

В работе используется четырехразрядный сумматор К555ИМ6, условное изображение которого показано на рис. 3.9.

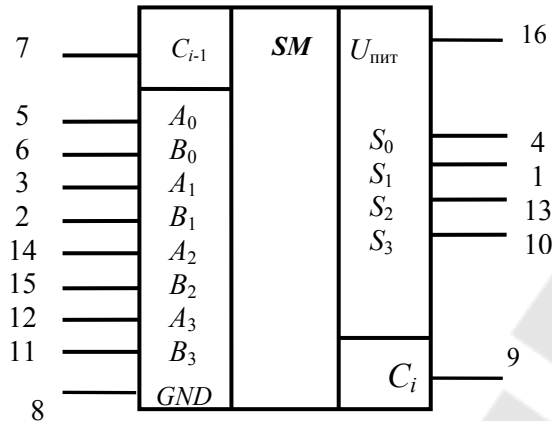


Рис. 3.9. Условное изображение и цоколевка микросхемы К555ИМ6

Принцип действия ее основан на параллельном суммировании чисел разных разрядов с параллельным переносом, который вырабатывается как функция только слагаемых и входного переноса всего сумматора. Вход переноса  $C_{i-1}$  имеется только у младшего разряда, а выход – только у старшего  $C_i$ .

Результат на выходах суммы и переноса описывается следующим выражением:

$$\begin{aligned} \sum_{A,B} &= C_{i-1} + 2^0(A_0 + B_0) + 2^1(A_1 + B_1) + 2^2(A_2 + B_2) + 2^3(A_3 + B_3) = \\ &= 2^0 \sum 0 + 2^1 \sum 1 + 2^2 \sum 2 + 2^3 \sum 3 + 2^4 \sum C_i. \end{aligned}$$

Микросхема может быть использована для операций с числами, представленными не только в положительной, но и в отрицательной логике. В режиме положительной логики вход  $C_{i-1}$  нельзя оставлять открытым, на него надо подать потенциал логического нуля.

Четырехразрядный сумматор 155ИМ5 можно применять и в качестве вычитателя. Операция вычитания выполняется путем сложения уменьшаемого с вычитаемым в дополнительном коде.

Операции сложения и вычитания можно совместить в одном узле, если инверторы заменить элементами ИСКЛЮЧАЮЩЕЕ ИЛИ, как это представлено на рис. 3.10. Эти элементы в зависимости от уровня напряжения на управляющем входе работают как повторители или инверторы.

При операции вычитания (лог. 1) вычитаемое  $B = B_3B_2B_1B_0$  преобразуется в обратный код (элементы ИСКЛЮЧАЮЩЕЕ ИЛИ работают как инверторы), и к нему по входу  $C_{i-1}$  прибавляется единица. Результат сложения числа  $A = A_3A_2A_1A_0$  с числом  $B$  в дополнительном коде формируется на выходах 3, 2, 1, 0. Здесь  $S_0 = A_0 + B_0 + 1$  (за счет  $C_{i-1} = 1$ );  $S_1 = A_1 + B_1 + 1$  и т. д. Бит высшего разряда отбрасывается, поэтому выход переноса  $C_i$  остается свободным.

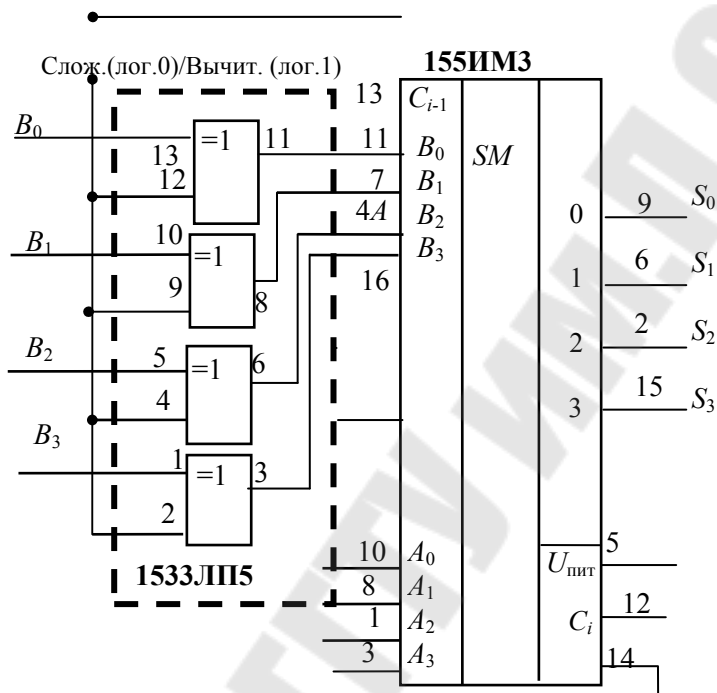


Рис. 3.10. Использование ИС 155ИМЗ в качестве сумматора и вычитателя

## Порядок выполнения работы

**Меры безопасности:** включать установку IDL-800 разрешается после того, как преподаватель проверит собранную для исследования схему. Запрещается оставлять включенную установку без присмотра. По окончании работы необходимо выключить установку, сложить соединительные провода и обесточить рабочее место.

### 1. Исследования сумматора

#### 1.1. Исследование полусумматора

Собрать на наборной панели IDL-800 схему полусумматора на логических элементах ИС 1533ЛП5 и ИС 1533ЛИ1 (рис. 3.1). Вывод

14 ИС соединить с источником питания  $+5V$ , а его вывод 7 – с общей шиной установки.

Таблица 3.1

**Таблица истинности полусумматора**

Входы		Выходы	
$A$	$B$	$\Sigma_i$	$P_i$

Переменные  $A_i, B_i$  задавать переключателями  $SW0-SW1$ , выходные состояния  $\Sigma_i$  и  $P_i$  контролировать светодиодными индикаторами. Убедиться в правильном функционировании сумматора, снять таблицу истинности, свести ее в табл. 3.1.

### 1.2. Исследование полного сумматора

Собрать схему одноразрядного сумматора на логических элементах (рис. 3.3). Переменные  $A_i, B_i$  задавать переключателями, выходные состояния  $\Sigma_i$  и  $P_i$  контролировать светодиодными индикаторами. Убедиться в правильном функционировании сумматора, снять таблицу истинности, свести ее в табл. 3.2.

Таблица 3.2

**Таблица истинности сумматора**

Входы			Выходы	
$A$	$B$	$P_{i-1}$	$\Sigma_i$	$P_i$

## 2. Исследование вычитателей

### 2.1. Исследование полувывчитателя

Собрать на наборной панели IDL-800 схему полувывчитателя на логических элементах (рис. 3.10). Вывод 14 ИС соединить с источником питания  $+5V$ , а его вывод 7 – с общей шиной установки.

Переменные  $A_i, B_i$  задавать переключателями, выходные состояния  $D_i$  и  $C_i$  контролировать светодиодными индикаторами. Убедиться в правильном функционировании сумматора, снять таблицу истинности, свести ее в табл. 3.3.

Таблица 3.3

Таблица истинности полувычитателя

Входы		Выходы	
$A$	$B$	$D_i$	$C_i$

### 2.2. Исследование полного вычитателя

Собрать схему одноразрядного сумматора на логических элементах (рис. 3.8). Переменные  $A_i$ ,  $B_i$  задавать переключателями, выходные состояния  $D_i$  и  $C_i$  контролировать светодиодными индикаторами. Убедиться в правильном функционировании сумматора, снять таблицу истинности, свести ее в табл. 3.4.

Таблица 3.4

Таблица истинности вычитателя

Входы			Выходы	
$A$	$B$	$C_{i-1}$	$\Sigma_i$	$C_i$

### 3. Исследование суммирования/вычитания в дополнительном коде на микросхеме ИС 155ИМ3(ИМ6)

Собрать схему исследования сумматора/вычитателя, изображенную на рис. 3.10. Задавая переключателями 4-разрядные числа  $A$  и  $B$ , осуществить сложение и вычитание двоичных чисел, контролируя выходы сумматора светодиодными индикаторами. Результат свести в табл. 3.5.

Таблица 3.5

Результат исследования микросхемы ИС 155ИМ3

Входы			Выходы	
$C_{i-1}$	$A_3A_2A_1A_0$	$B_3B_2B_1B_0$	$S_3S_2S_1S_0$	$C_i$
0	0011	1110		
0	1110	0101		
0	1010	1000		
0	0011	1101		
1	0011	1110		
1	1110	0101		
1	1010	1000		
1	0011	1101		

Записать полученные таблицы для отчета и убедиться в правильности суммирования с помощью перевода суммируемых чисел  $A$  и  $B$  в десятичные числа.

### **Содержание отчета**

1. Наименование и цель работы.
2. Исследуемые схемы сумматоров (вычитателей) с результатами исследований в виде таблиц истинности.
3. Выводы по результатам исследований.

### **Контрольные вопросы**

1. Полусумматор и сумматор. Уравнения, схемы.
2. Полувычитатель и вычитатель. Уравнения, схемы.
3. Использование сумматора в режиме вычитателя.
4. Последовательный и параллельный перенос в многоразрядных сумматорах.
5. Суммирование многоразрядных чисел, представленных в последовательном коде.

## **Лабораторная работа № 4**

### **Исследование преобразователей кодов**

**Цель работы:** изучить принцип действия и исследовать свойства преобразователей двоичных кодов.

#### **Используемое оборудование:**

- универсальная лабораторная установка IDL-800;
- четыре логических элемента ИСКЛЮЧАЮЩИЕ ИЛИ (ИС 1533ЛП5);
- два четырехразрядных сумматора (ИС 155ИМ3 или 555ИМ6);
- четыре логических элемента 2И – ИС 1533ЛИ1;
- четыре логических элемента 2ИЛИ – ИС 1533ЛЛ1;
- шесть инверторов – ИС 1533ЛН1.

#### **Основные теоретические сведения**

При передаче информации по линиям связи удобно использовать коды, позволяющие, например, уменьшить вероятность появления ошибки или даже исправлять ее в дальнейшем. Примерами таких кодов являются коды, построенные по принципу 2 из 5 (в которых из пяти символов два всегда имеют единичные значения), коды с проверкой четности или нечетности, коды Хемминга и др.

В связи с этим всегда стоит задача преобразования информации из одного кода в другой. Эти задачи решают комбинационные устройства – преобразователи кодов.

Преобразователем кода называется комбинационное устройство, предназначенное для изменения вида кодирования информации. Как и всякое комбинационное устройство преобразователь кодов характеризуется таблицей истинности, ставящей в соответствие кодам, подаваемым на вход, коды, снимаемые с выхода устройства.

Эта таблица должна давать однозначное соответствие входным и выходным кодам. Эта таблица является основанием для синтеза логической структуры конкретного преобразователя кода.

Преобразование из одного кода в другой занимает важное место в работе с цифровыми устройствами.

Частным случаем преобразователей кодов являются шифраторы и дешифраторы.

Под преобразованием двоичных кодов понимается преобразование  $n$ -разрядных двоичных чисел, представляющих информацию в одном заданном коде, в  $m$ -разрядные двоичные числа, представляющие эту информацию в другом коде.

Условное графическое изображение преобразователя кодов приведено на рис. 4.1.

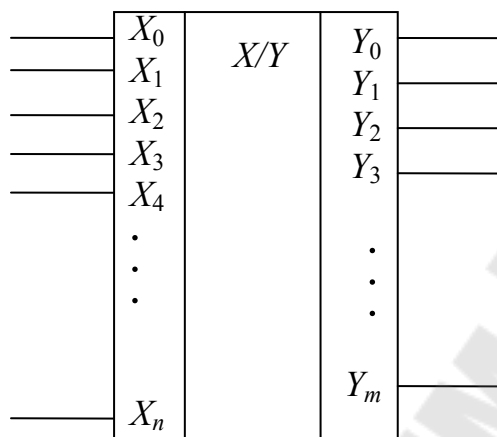


Рис. 4.1. Условное графическое изображение преобразователя кодов

Рассмотрим один из наиболее распространенных подходов к синтезу преобразователей кодов: синтез  $n$  независимых одновыходных функций по заданной таблице истинности – таблице соответствия кодов.

Допустим, имеется таблица истинности преобразователя 2-разрядного двоичного  $a_1a_0$  кода в 3-разрядный  $b_2b_1b_0$  (табл. 4.1). Считая  $b_0$ ,  $b_1$  и  $b_2$  независимыми одновыходными функциями, запишем для каждой из них булевы выражения:

$$b_0 = a_1\bar{a}_0 + \bar{a}_1a_0, \quad b_1 = a_1\bar{a}, \quad b_2 = \bar{a}_1\bar{a}_0.$$

Таблица 4.1

**Таблица истинности преобразования 2-разрядного двоичного кода в 3-разрядный**

$a_1$	$a_0$	$b_2$	$b_1$	$b_0$
0	0	1	0	0
0	1	0	0	1
1	0	0	1	1
1	1	0	0	0



Используя приведенные булевы выражения с учетом того, что инверсии некоторых переменных и произведение  $a_0$  встречаются не в одной функции, составляем схему преобразователя с применением (где это возможно) одних и тех же логических элементов (рис. 4.2):

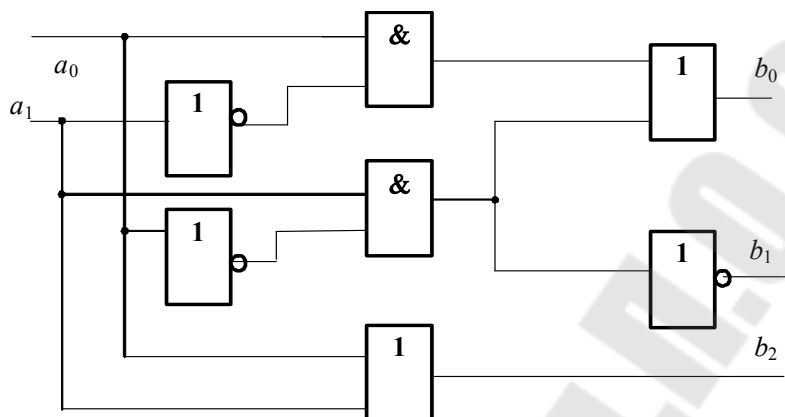


Рис. 4.2. Пример построения преобразователя кодов по заданному булеву выражению

На основе двоичной системы счисления в цифровой технике разработаны различные формы представления чисел, которые иногда называют кодами. Естественную запись двоичного числа называют двоичным кодом.

Так же как большие десятичные числа для удобства чтения разбивают при записи на тройки, так и большие двоичные числа обычно разбивают на четверки, – тетрады. Две тетрады или восемь двоичных цифр называют байт – *BYTE*. Число  $2^{10} = 1024$  называют в цифровой технике словом «килобит» в нарушение точного значения данного слова – 1000.

В цифровой аппаратуре, в основном при индикации показаний десятичными цифрами или при задании параметров десятичными датчиками, широко применяются различные двоично-десятичные коды. Самый распространенный из них *BCD*-код (сокращенное *BINARY CODED DECIMAL* – двоично-кодированная десятичная цифра), который порой называют позиционным 8421-кодом, или натуральным двоично-десятичным кодом. В этом коде каждая десятичная цифра представляется своей отдельной тетрадой – четверкой двоичных цифр, например:

$$526_{10} = \begin{array}{ccc} 5 & 2 & 6 \\ 0101 & 0010 & 0110 \end{array}_{BCD}$$

С помощью 4-х битов можно составить 16 различных сочетаний единиц и нулей, а для десятичных цифр достаточно 10 сочетаний. шесть сочетаний избыточны, поэтому возможны различные позиционные двоично-десятичные коды. Отличаются эти коды порядком следования степеней основания в представлении десятичных цифр. Так *BCD*-код имеет следующие веса разрядов:  $2^3, 2^2, 2^1, 2^0 = 8421$ .

Другой позиционный двоично-десятичный код – код Эйкена – имеет следующие веса разрядов:  $2^1, 2^2, 2^1, 2^0 = 2421$ .

Десятичное число  $9_{10}$ , записанное в *BCD*-коде, равно  $1001_{BCD}$ , а это же число, записанное в коде Эйкена, имеет следующий вид:  $1111_{\text{Эйкен}}$ .

Еще один распространенный двоично-десятичный код, – код с избытком 3 (*EXCESS-3 CODE*). В нем каждая десятичная цифра кодируется двоичной тетрадой, в которой взвешенная сумма разрядов больше этой десятичной цифры на три. Так, десятичная цифра 9 записывается тетрадой 1100, для которой взвешенная сумма разрядов  $8 \cdot 1 + 4 \cdot 1 + 2 \cdot 0 + 1 \cdot 0 = 12$ , что на 3 больше 9.

Этот код называют самодополняющимся до 10, т. е. его младшие 5 цифр (от 0 до 4) являются зеркальным отражением, инверсией старших 5 цифр (от 5 до 9); например, число  $0 = 0011_{EX3}$  является инверсным числу  $9 = 1100_{EX3}$ , число  $1 = 0100_{EX3}$  является инверсным числу  $8 = 1011_{EX3}$  и т. д. Достоинством кода с избытком 3 является его повышенная надежность при передаче информации в канале связи, поскольку количество единиц в его числах в среднем равно количеству нулей.

### ***Преобразователь кода Грея в двоичный позиционный***

*Код Грея* – отраженный двоичный код, рефлексный (от *reflect* – отражать) код для первых восьми чисел, представлен в табл. 4.1.

В последовательности чисел кода Грея все разряды ( $G_2G_1G_0$ ), кроме самого левого, подчиняются следующему правилу: любая сплошная группа разрядов, считая справа, по некоторому закону перебирает все свои возможные комбинации, а затем начинает перебирать их в обратном порядке. Так, разряды кода Грея  $G_1$  и  $G_0$  сначала перебирают комбинации 00, 01, 11, 10, затем – те же комбинации в обратном порядке: 10, 11, 01, 00.

Закон справедлив вплоть до группы в один разряд: разряд  $s$  изменяется сначала в одну сторону – 0,1, затем обратно – 1,0 и снова в том же цикле. Этот закон объясняет одно из названий кода «отра-

женный» и позволяет строить последовательность чисел кода любой разрядности. Код Грея – **непозиционный**, т. е. веса его разрядов не определяются занимаемыми ими местами, как в обычном двоичном коде, который относится к классу позиционных.

В коде Грея при переходе между любыми соседними числами изменяется значение всегда только одного разряда.

Благодаря этому свойству код широко применяется в преобразователях углового положения вала в цифровой код, построенных на базе кодового диска или кодового барабана. В оптическом кодовом диске единицы и нули кодируются прозрачными и непрозрачными областями.

Таблица 4.2

Таблица кода Грея для 8 чисел

Десятичное число	Двоичное число $B_2B_1B_0$	Число в коде Грея $G_2G_1G_0$
0	000	000
1	001	001
2	010	011
3	011	010
4	100	110
5	101	111
6	110	101
7	111	100

Число  $G$ , записанное в коде Грея, можно преобразовать в число  $B$  в двоичном коде с помощью следующего выражения:  $B_i = G_i \oplus B_{i+1}$ . Преобразование кода Грея в двоичный позиционный код производят по схеме, приведенной на рис. 4.3.

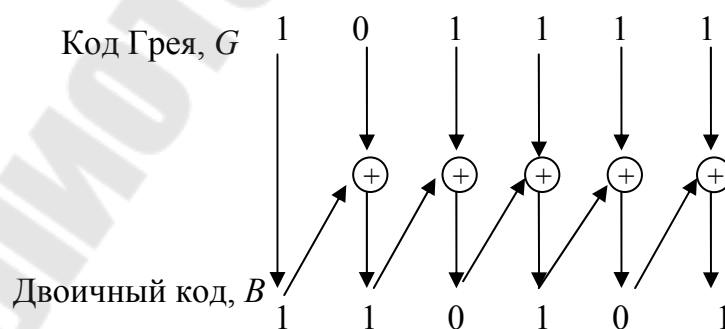


Рис. 4.3. Пример преобразования числа в коде Грея в позиционный код

Два варианта построения схем преобразования кода Грея в позиционный показаны на рис. 4.4. Первый (параллельная схема на рис. 4.4, а) имеет меньшую задержку, но дороже по затратам оборудования, второй (последовательная схема на рис. 4.4, б) – схема с большей задержкой, но экономичнее по аппаратным затратам.

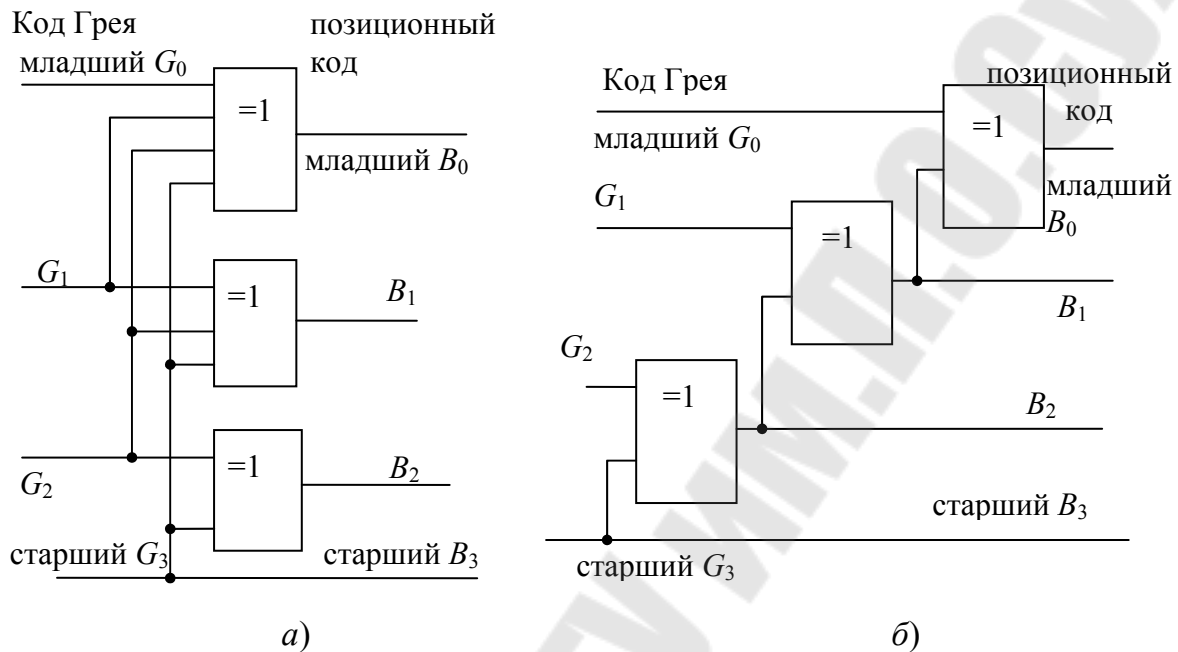


Рис. 4.4. Преобразователи кода Грея в двоичный позиционный

Переводить числа из позиционного кода в код Грея приходится значительно реже. Алгоритм перевода прост: каждый  $i$ -й, считая слева, разряд числа в коде Грея равен сумме по модулю 2  $i$ -го и  $(i-1)$ -го слева разрядов того же числа, представленного позиционным кодом.

Число  $B$ , записанное в двоичном коде, можно преобразовать в число  $G$  в коде Грея (рис. 4.5) с помощью следующего выражения:

$$G_i = B_i \oplus B_{i+1}.$$

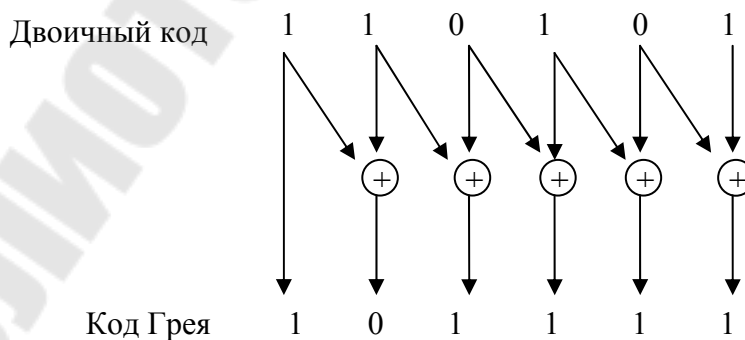


Рис. 4.5. Пример преобразования двоичного числа в код Грея

Схема преобразователя кода Грея в позиционный двоичный представлена на рис. 4.6.

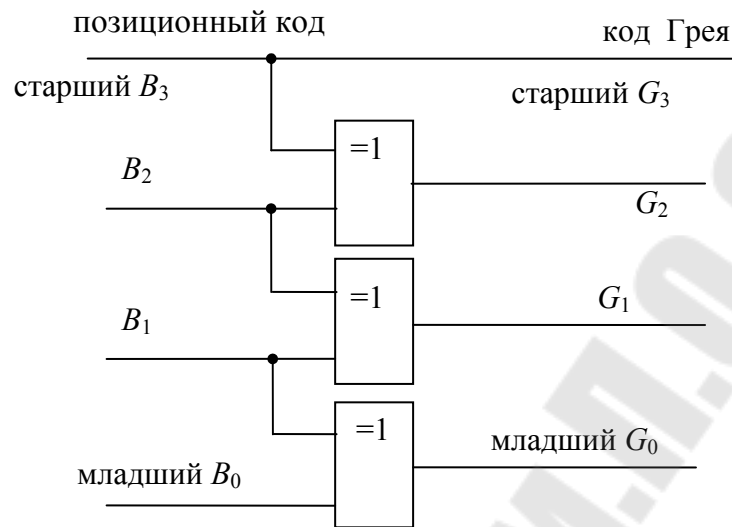


Рис. 4.6. Схема преобразователя позиционного двоичного кода в код Грея

### Преобразователь BCD-кода в прямой двоичный код

Рассмотрим схему преобразования BCD-кода в двоичный код, изображенную на рис. 4.7. На вход будут подаваться две тетрады двоичного кода:  $A_1B_1C_1D_1$ , представляющая десятки, и  $A_0B_0C_0D_0$ , представляющая единицы.

На выходе будет семиразрядный двоичный код  $b_6 b_5 b_4 b_3 b_2 b_1 b_0$ .



Рис. 4.7. Структура преобразователя BCD-кода в прямой двоичный код

Десятичные веса каждого бита в двоично-десятичном виде могут быть представлены двоичными эквивалентами (табл. 4.3). Используя это, можно осуществить преобразование *BCD*-кода в двоичный код путем суммирования двоичных эквивалентов тех бит, которые в двоично-десятичном представлении равны единице.

Таблица 4.3

Двоичные эквиваленты десятичных весов *BCD*-битов

<i>BCD</i> -биты	Десятичные веса	Двоичные эквиваленты						
		$b_6$	$b_5$	$b_4$	$b_3$	$b_2$	$b_1$	$b_0$
$A_1$	80	1	0	1	0	0	0	0
$B_1$	40	0	1	0	1	0	0	0
$C_1$	20	0	0	1	0	1	0	0
$D_1$	10	0	0	0	1	0	1	0
$A_0$	8	0	0	0	1	0	0	0
$B_0$	4	0	0	0	0	1	0	0
$C_0$	2	0	0	0	0	0	1	0
$D_0$	1	0	0	0	0	0	0	1

Например, преобразуем число  $63_{10}$ . Запишем это число в *BCD*-коде:  $63_{10} = 01100011_{BCD}$ . Теперь просуммируем двоичные эквиваленты всех единиц и десятков весов разрядов этого числа (рис. 4.8).

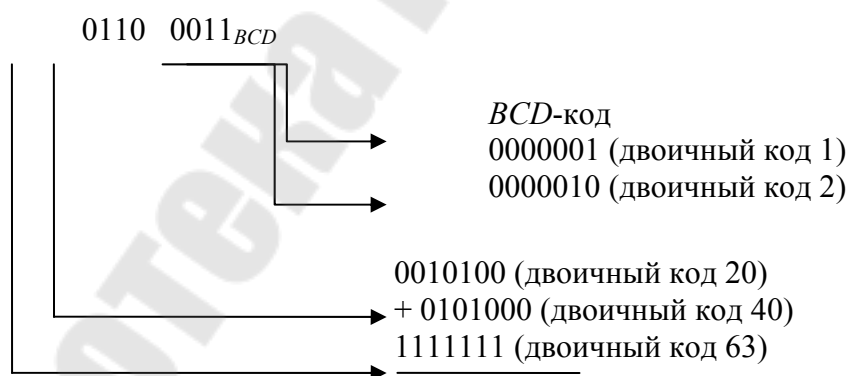


Рис. 4.8. Преобразование числа  $63_{10}$  в прямой двоичный код

На основании рассмотренного примера можно разработать одну из возможных схем преобразования двоично-десятичного *BCD*-кода в прямой двоичный на основе двоичных сумматоров. На рис. 4.9 пред-

ставлена схема такого преобразователя на основе двух четырехразрядных двоичных сумматоров (К155ИМ3 или К555ИМ6, цоколевка которых приведена в лабораторной работе № 3).

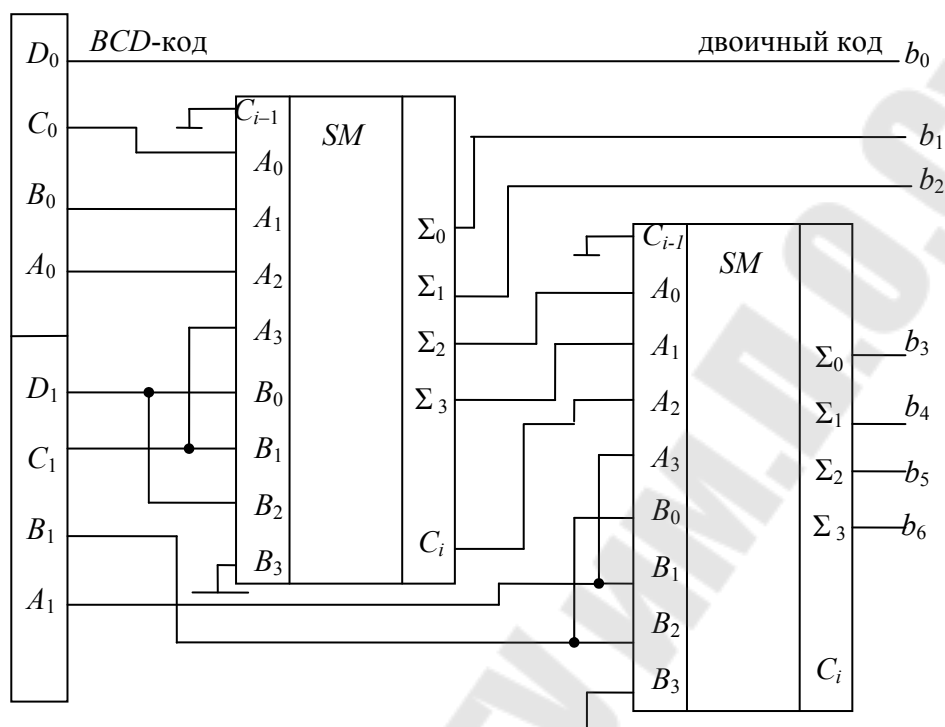


Рис. 4.9. Преобразователь BCD-кода в прямой двоичный код

## Порядок выполнения работы

**Меры безопасности:** включать установку IDL-800 разрешается после того, как преподаватель проверит собранную для исследования схему. Запрещается оставлять включенную установку без присмотра. По окончании работы необходимо выключить установку, сложить соединительные провода и обесточить рабочее место.

### 1. Исследование преобразователя кодов, построенного по заданной таблице истинности

По заданной преподавателем таблице истинности записать булевы выражения для выходных переменных, считая их независимыми одновыходными функциями.

Синтезировать схему преобразователя кодов на логических элементах по полученным булевым выражениям. Входные переменные  $a_i$  и их инверсии задавать переключателями, выходные состояния  $b_i$  контролировать светодиодными индикаторами.

Собрать схему преобразователя двоичных кодов на наборной панели IDL-800 в базе логических элементов И (ИС 1533ЛИ1), ИЛИ (ИС 1533ЛЛ1), НЕ (ИС 1533ЛН1). Убедиться в правильном функционировании преобразователя.

Схему преобразователя и таблицу истинности зарисовать для отчета.

## **2. Исследование преобразователя позиционного двоичного кода в код Грея**

Собрать на наборной панели IDL-800 схему преобразователя позиционного двоичного кода в код Грея на логических элементах ИСКЛЮЧАЮЩЕЕ ИЛИ (ИС 1533ЛП5) (см. рис. 4.6). Входные переменные  $b_i$  и их инверсии задавать переключателями, выходные состояния переменных  $g_i$  контролировать светодиодными индикаторами.

Убедиться в правильном функционировании преобразователя. Результаты свести в табл. 4.4.

## **3. Исследование преобразователя кода Грея в позиционный двоичный код**

Собрать на наборной панели IDL-800 схему преобразователя кода Грея в позиционный двоичный на логических элементах ИСКЛЮЧАЮЩЕЕ ИЛИ (см. рис. 4.4). Входные переменные  $g_i$  и их инверсии задавать переключателями, выходные состояния переменных  $b_i$  контролировать светодиодными индикаторами.

Убедиться в правильном функционировании преобразователя.

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

Таблица 4.4

Результат преобразования BCD-кода в двоично-десятичный код с избытком три

Десятичные числа	Двоичные числа	Двоично-десятичные числа		Код Грея
		8421	Excess-3	
0	0000	0000		
1	0001	0001		
2	0010	0010		
3	0011	0011		
4	0100	0100		
5	0101	0101		
6	0110	0110		



Десятичные числа	Двоичные числа	Двоично-десятичные числа		Код Грея
		8421	Excess-3	
7	0111	0111		
8	1000	1000		
9	1001	1001		
10	1010	0001	0000	
11	1011	0001	0001	
12	1100	0001	0010	
13	1101	0001	0011	
14	1110	0001	0100	
15	1111	0001	0101	
16	10000	0001	0110	
17	10001	0001	0111	
18	10010	0001	1000	
19	10011	0001	1001	
20	10100	0010	0000	

На основе четырехразрядного сумматора разработать схему преобразователя *BCD*-кода в двоично-десятичный код с избытком три (*Excess-3*). Собрать разработанную схему на панели установки IDL-800. Входные переменные задавать переключателями, выходные состояния переменных контролировать светодиодными индикаторами. Схему зарисовать для отчета, результаты исследования свести в табл. 4.3.

### **5. Исследование преобразователя *BCD*-кода в прямой двоичный код**

Собрать схему преобразователя *BCD*-кода в прямой двоичный код (см. рис. 4.9) на основе двух четырехразрядных двоичных сумматоров (К155ИМ3). Входные переменные  $A_1B_1C_1D_1$  и  $A_0B_0C_0D_0$  задавать переключателями, выходные состояния переменных  $b_0$ – $b_6$  контролировать светодиодными индикаторами.

Убедиться в правильном функционировании преобразователя.

## Содержание отчета

1. Наименование и цель работы.
2. Исследуемые в работе схемы преобразователей с результатами исследований в виде таблиц истинности.
3. Выводы по результатам исследований.

## Контрольные вопросы

1. Преобразование двоичного кода в код Грея.
2. Преобразование кода Грея в двоичный код.
3. Преобразование двоичных чисел в *BСD*-код.
4. Преобразование двоичных чисел в код с избытком три.
5. Объяснить работу преобразователя в двоично-десятичный код с избытком три.
6. Объяснить работу преобразователей кодов, исследуемых в работе.

## **Лабораторная работа № 5**

### **Синтез комбинационных схем на дешифраторах и мультиплексорах**

**Цель работы:** изучение принципов построения дешифраторов и мультиплексоров, а также исследование особенностей синтеза комбинационных схем на основе этих цифровых устройств.

#### **Используемое оборудование:**

- универсальная лабораторная установка IDL-800;
- дешифратор-демультиплексор 3 : 8 с тремя разрешающими входами (ИС 155ИД7);
- мультиплексор 8 : 1 с разрешающим входом (ИС 1533КП15);
- четыре логических элемента 2И – ИС 1533ЛИ1;
- два логических элемента 4И-НЕ – ИС 1533ЛА1;
- инвертор ИС 1533ЛН1.

#### **Основные теоретические сведения**

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

*Дешифраторы* осуществляют преобразование входного двоичного кода в *унитарный* (иногда называют «*унарный*») код, т. е. код, включающий в себя *только одну логическую единицу*, а остальные – логические нули. Дешифраторы, например, могут использоваться в многоканальных цифровых системах для выбора одного из каналов в соответствии с входным кодом. Количество выходов *полного* дешифратора  $N$  определяется по формуле  $N = 2^n$ , где  $n$  – количество разрядов входного кода. При обозначении иногда указывается число входов и выходов дешифратора, например, «дешифратор 2÷4» имеет

$n = 2$  входа и  $N = 4$  выхода. Если же  $N < 2^n$ , то дешифратор называют *неполным*.

На рис. 5.1 приведен пример условного обозначения дешифратора 2 : 4, имеющего разрешающий вход  $E$ .



Рис. 5.1. Условное графическое обозначение дешифратора 2 : 4, имеющего разрешающий вход  $E$

В табл. 5.1 представлены булевы выражения выходных функций дешифратора 2 : 4:

$$D_0 = \overline{A_1} \overline{A_0}, \quad D_1 = \overline{A_1} A_0, \quad D_2 = A_1 \overline{A_0}, \quad D_3 = A_1 A_0.$$

Таблица 5.1

Таблица состояний дешифратора 2 : 4

$E$	$A_1$	$A_0$	$D_0$	$D_1$	$D_2$	$D_3$
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1
0	1	1	0	0	0	0
1	$x$	$x$	0	0	0	0

Существует два способа образования разрешающего входа  $E$ :

*Первый способ* – увеличение числа входов ЛЭ на единицу и объединение этих дополнительных входов. При этом для дешифратора на конъюнкторах образуется прямой разрешающий вход, а для дешифратора на дизъюнкторах – инверсный. На рис. 5.2 приведена структурная схема дешифратора, в которой образование разрешающего входа осуществляется за счет увеличения числа входов ЛЭ.

*Второй способ* – управление по одной из переменных. Используется то обстоятельство, что на всех ЛЭ, образующих дешифратор, обязательно имеется переменная или ее дополнение любого разряда кода адреса. Если сделать поступление переменной и ее дополнения одного из разрядов зависимым от разрешения  $E$ , то тем самым создается условие образования разрешающего входа.

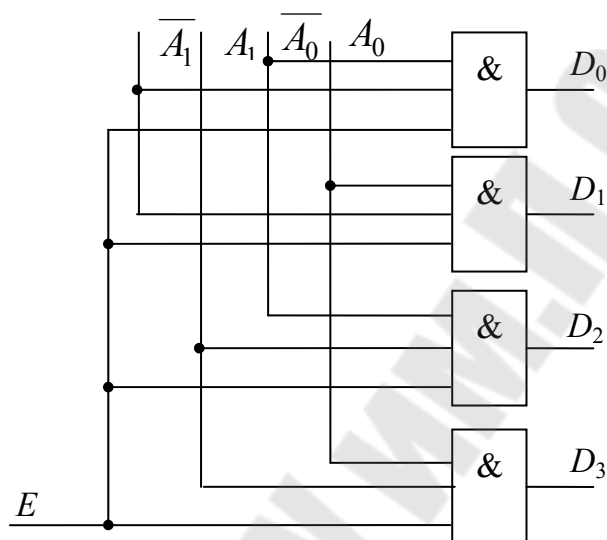
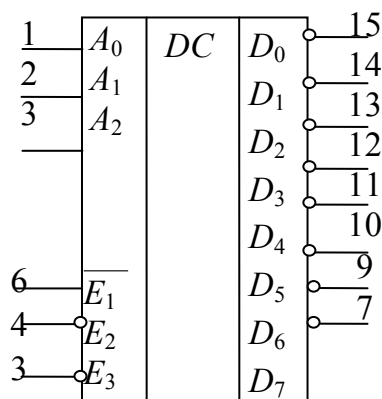


Рис. 5.2. Структурная схема дешифратора, в которой образование разрешающего входа осуществляется за счет увеличения числа входов логических элементов

Микросхемы дешифраторов часто имеют не один, а два или даже три входа  $E$  разрешения, причем некоторые из них прямые, а другие – инверсные. Такие входы удобно использовать при наращивании разрядности дешифратора, собирая каскадный дешифратор.



вывод 16 – питание; вывод 8 – общий

Рис. 5.3. Цоколевка дешифратора К155ИД7

Микросхема К155ИД7 представляет собой дешифратор-демультиплексор 3 : 8 с тремя разрешающими входами, связанными конъюнкцией. Один из разрешающих входов прямой, а два инверсные. Наличие нескольких разрешающих входов позволяет наращивать разрядность дешифратора объединением нескольких микросхем К155ИД7.

Пример наращивания разрядности за счет объединения разрешающих входов приведен на рис. 5.4.

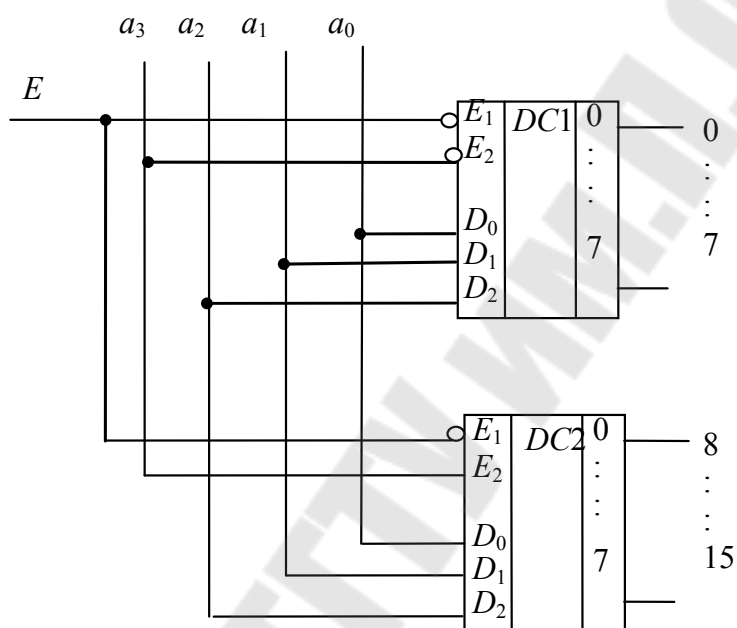


Рис. 5.4. Наращивание разрядности дешифраторов за счет объединения разрешающих входов

На рис. 5.5 показана группа из пяти дешифраторов, соединенных в два каскада. Вся группа работает как дешифратор 5 : 32. Два старших разряда адреса  $a_{16}$  и  $a_8$  расшифровываются дешифратором 2 : 4 DC4, который по входам  $E$  управляет четырьмя дешифраторами 3 : 8 второго каскада.

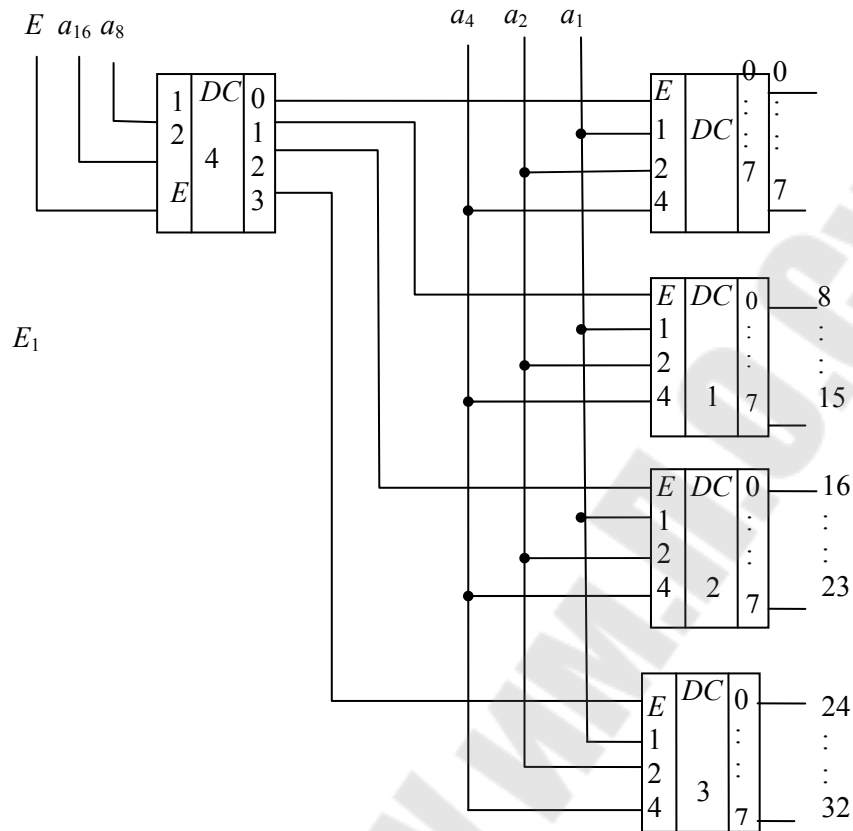


Рис. 5.5. Каскадное соединение дешифраторов 5 : 32

Младшие разряды адреса  $a_4, a_2, a_1$  поступают на все дешифраторы второго каскада, но открытым по входу  $E$  оказывается лишь один из них. Ему и будет принадлежать единственный активный из всех 32-х выходов. Так, при поступлении кода  $a_{16}a_8a_4a_2a_1 = 01111$  у  $DC4$  сигнал появится на выходе 1, а по входу  $E$  будет открыт  $DC1$ . Остальные дешифраторы второго каскада будут заперты. Разряды адреса  $a_4a_2a_1=111$  вызовут появление единицы на выходе 7 дешифратора  $DC1$ , т. е. на выходе 15 всего составного дешифратора, что соответствует заданному адресу. Принцип используется при построении дешифраторов на много выходов из микросхем дешифраторов с меньшим числом выходов.

В рассмотренном примере 5-разрядный адрес был разбит на две группы в 2 и 3 разряда, и это определило структуру всей схемы. В общем случае многоразрядный адрес можно разбить на группы различными способами, и каждому способу будет соответствовать свой вариант схемы многокаскадного (не обязательно двухкаскадного) дешифратора. Варианты будут отличаться задержкой и аппаратурными

затратами, и можно ставить задачу выбора оптимальной в заданной серии элементов структуры.

Информация с единственного информационного входа демультиплектора поступает на один из выходов в зависимости от комбинации двоичных чисел на адресных входах.

Дешифратор также можно использовать в качестве демультиплектора, используя разрешающий вход  $E$  в качестве информационного (рис. 5.6).

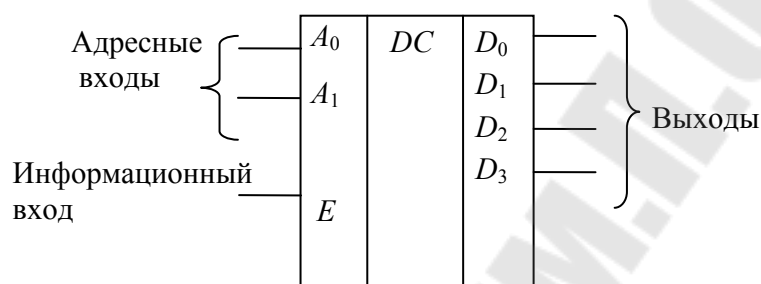


Рис. 5.6. Использование дешифратора 2 : 4 в качестве демультиплектора

Это утверждение можно проверить, используя таблицу истинности дешифратора 2 : 4 (см. табл. 5.1).

Дешифраторы-демультиплексоры используются в синтезе комбинационных схем. Особенно эти устройства полезны при синтезе комбинационных схем 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 в качестве примера реализацию схемы полного сумматора на основе дешифратора К155ИД7.

По таблице истинности полного сумматора, исследованного в лабораторной работе № 3, составим булевы выражения для выходов  $\Sigma_i$  и  $P_i$  (табл. 5.2):

$$\Sigma_i = \overline{A_i} \overline{B_i} P_{i-1} + \overline{A_i} B_i \overline{P_{i-1}} + A_i \overline{B_i} \overline{P_{i-1}} + A_i B_i P_{i-1};$$

$$P_i = \overline{A_i} B_i P_{i-1} + A_i \overline{B_i} P_{i-1} + A_i B_i \overline{P_{i-1}} + A_i B_i P_{i-1}.$$



Таблица истинности сумматора

Входы			Выходы	
$A$	$B$	$P_{i-1}$	$\Sigma_i$	$P_i$
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

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

$$\Sigma_i = \overline{\overline{A_i B_i P_{i-1}} \cdot \overline{A_i B_i P_{i-1}} \cdot \overline{A_i B_i P_{i-1}} \cdot \overline{A_i B_i P_{i-1}}};$$

$$P_i = \overline{\overline{A_i B_i P_{i-1}} \cdot \overline{A_i B_i P_{i-1}} \cdot \overline{A_i B_i P_{i-1}} \cdot \overline{A_i B_i P_{i-1}}}.$$

После чего функция  $\Sigma_i$  может быть записана как  $\Sigma_i = \overline{D_1 D_2 D_4 D_7}$ , а функция  $P_i$  может быть записана в виде  $P_i = \overline{D_3 D_5 D_6 D_7}$ .

Для реализации полного сумматора будем использовать дешифратор-демультиплексор К155ИД7 (SN74ALS138). Схема полного сумматора изображена на рис. 5.7.

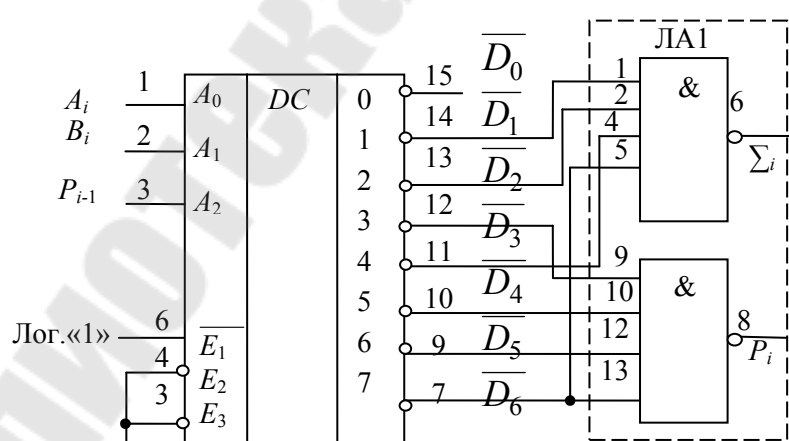


Рис. 5.7. Реализации полного сумматора на базе КР155ИД7

Как видно из рассмотренного примера, при составлении комбинационных схем на основе дешифраторов-демультиплексоров требуется использование дополнительных логических элементов. Этого можно избежать, реализуя функции на базе мультиплексоров.

Мультиплексированием (*MULTIPLEX*) называют передачу данных от нескольких источников по одному каналу поочередно. *Мультиплексор* (или селектор данных) – это функциональное устройство, осуществляющее подключение (коммутацию) одного из нескольких информационных входов данных к выходу. Номер выбранного входа соответствует двоичному коду, подаваемому на адресные входы мультиплексора. В цифровой схемотехнике мультиплексор имеет один выход;  $m$  информационных входов данных  $D_0, D_1, D_2, D_3, \dots, D_m$  и  $n$  адресных входов  $A_1, A_2, A_3, \dots, A_n$ , причем  $m = 2^n$ .

На рис. 5.8 приведен пример условного обозначения мультиплексора на схемах. Число информационных входов  $D_0-D_m$ , коммутируемых на выход  $Y$ , составляет  $m = 4$ . Такой мультиплексор имеет размерность 4 : 1.

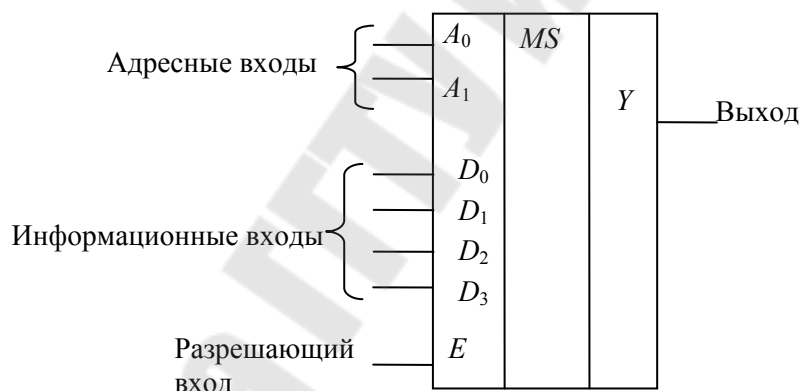


Рис. 5.8. Условное графическое обозначение мультиплексора 4 : 1 с разрешающим входом

Таблица истинности мультиплексора с двумя адресными входами приведена в табл. 5.3.

Таблица 5.3

Таблица состояний мультиплексора

$E$	$A_1$	$A_0$	$Y$
1	0	0	$D_0$
1	0	1	$D_1$
1	1	0	$D_2$
1	1	1	$D_3$

Наличие разрешающего входа  $E$  позволяет увеличивать число информационных входов вдвое путем последовательного соединения разрешающих входов двух мультиплексоров (наращивание разрядности). На рис. 5.9 представлена схема получения мультиплексора 8 : 1 на базе двух мультиплексоров 4 : 1. Последовательное соединение разрешающих входов позволяет получить третий адресный вход мультиплексора.

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 в качестве универсального логического элемента для синтеза комбинационных схем. Возможность реализации заданной логической функции основана на принципе работы мультиплексора.

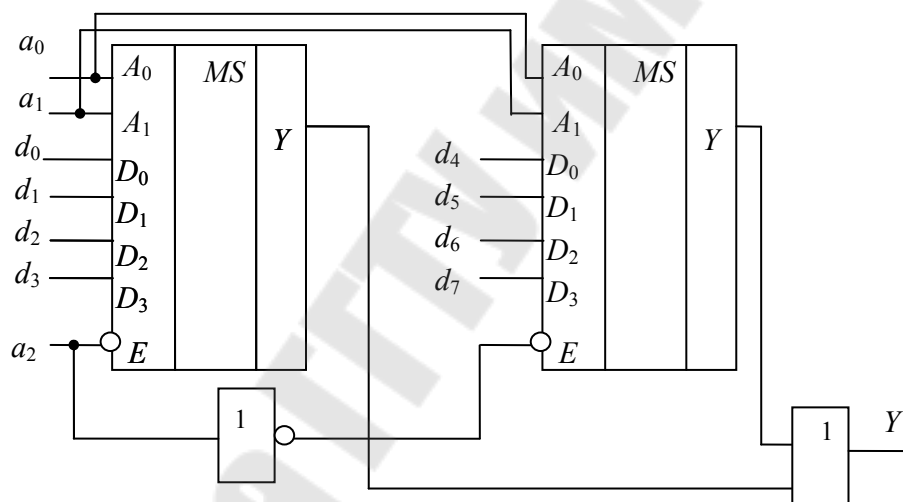


Рис. 5.9. Схема наращивания разрядности мультиплексора за счет разрешающего входа

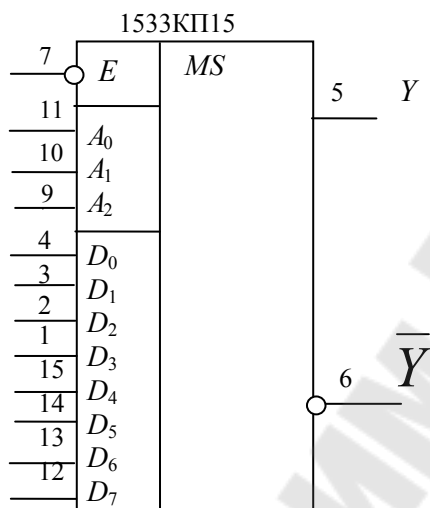
В работе исследуется мультиплексор К1533КП15, цоколевка которого представлена на рис. 5.10.

Использование мультиплексоров дает следующие преимущества при синтезе комбинационных схем:

- не требуется упрощение логического выражения, составленного по таблице истинности;
- минимизируется число требуемых интегральных схем;
- соответственно синтез комбинационных схем упрощается.

Для реализации заданной таблицей истинности функции с использованием мультиплексора необходимо определить десятичные

номера каждого из логических выражений таблицы истинности, для которых выходной сигнал принимает значение  $Y = 1$ . Входы мультиплексора, соответствующие этим номерам, соединить с лог. 1; 2. Все остальные входы соединить с лог. 0. Входные переменные заданной функции подать на адресные входы.



вывод 16 – питание; вывод 8 – общий

Рис. 5.10. УГО и цоколевка мультиплексора К1533КП15

Рассмотрим пример: пусть требуется реализовать функцию, заданную таблицей истинности (табл. 5.4), используя мультиплексор.

Таблица 5.4

#### Пример задания логической функции

$A$	$B$	$C$	$Y$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Определим десятичные номера каждого из логических выражений таблицы истинности, для которых выходной сигнал принимает значение:  $Y = 1$ . Это цифры 1, 3, 6, 7. На входы данных мультиплексо-

ра, соответствующие этим номерам, подадим лог. 1, остальные соединим с общим.

Определим десятичные номера каждого из логических выражений таблицы истинности, для которых выходной сигнал принимает значение:  $Y = 1$ . Это цифры 1, 3, 6, 7. На входы данных мультиплексора, соответствующие этим номерам, подадим лог. 1, остальные соединим с общим.

Входные переменные заданной функции ( $ABC$ ) подадим на адресные входы. Реализация заданной функции на мультиплексоре К1533КП15 представлена на рис. 5.11.

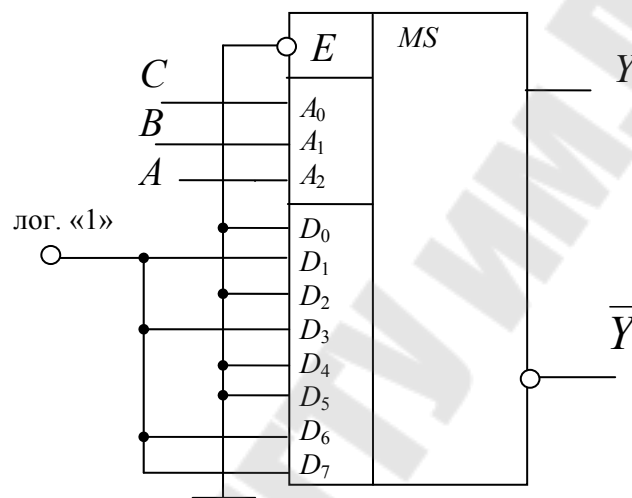


Рис. 5.11. Реализация заданной табл. 5.4 функции

Рассмотрим пример реализации функции  $m+1$  переменных на мультиплексорах с  $m$  адресными входами. Реализуем функцию четырех переменных ( $ABCD$ ), заданную таблицей истинности (табл. 5.5) при помощи мультиплексора 8 : 1.

Таблица 5.5

**Логическая функция четырех переменных**

$A$	$B$	$C$	$D$
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1

Окончание табл. 5.5

<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1

Функция четырех переменных может быть реализована на мультиплексоре размерностью 8 : 1 следующим образом: переменные заданной функции (*ABC*) подключаем к адресным входам  $A_2, A_1, A_0$  соответственно. Переменную *D*, не подключенную к адресным входам, называют «выделенной». Без выделенной переменной наборы переменных *ABC* образуют пары. В табл. 5.6 эти пары выделены пунктирными линиями.

Таблица 5.6

**Функция четырех переменных, реализованных на мультиплексоре 8 : 1**

Переменные				Значение функции	
<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>Y</i>	
0	0	0	0	0	Совпадает с <i>D</i>
0	0	0	1	1	
0	0	1	0	0	0
0	0	1	1	0	
0	1	0	0	0	0
0	1	0	1	0	
0	1	1	0	0	0
0	1	1	1	0	
1	0	0	0	1	Совпадает с $\bar{D}$
1	0	0	1	0	

Переменные				Значение функции	
<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>Y</i>	
1	0	1	0	1	Совпадает с $\bar{D}$
1	0	1	1	0	
1	1	0	0	1	1
1	1	0	1	1	
1	1	1	0	1	1
1	1	1	1	1	

Теперь рассмотрим соотношения между выделенной переменной  $D$  и выходом  $Y$  для каждой пары.

При этом возможны четыре варианта:

- выход не зависит от переменной и равен нулю;
- выход не зависит от переменной  $D$  и равен единице;
- выход  $Y$  зависит от переменной  $D$  и равен ей;
- выход  $Y$  зависит от переменной  $D$  и совпадает с ее инверсией  $\bar{D}$ .

Исходя из вышеизложенного, на информационные входы мультиплексора и подается лог. 0, лог. 1,  $D$  или  $\bar{D}$ . Пример реализации функции четырех переменных, заданной табл. 5.6 на базе мультиплексора с тремя адресными входами, изображен на рис. 5.12.

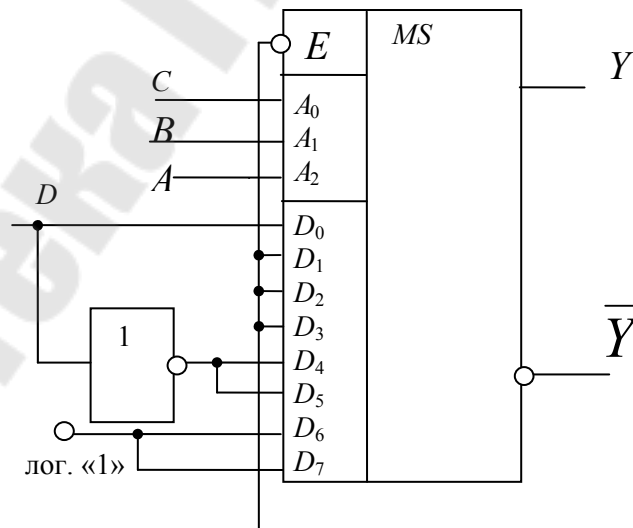


Рис. 5.12. Реализация заданной табл. 5.6 функции на мультиплексоре К1533КП15

## Порядок выполнения работы

**Меры безопасности:** включать установку IDL-800 разрешается после того, как преподаватель проверит собранную для исследования схему. Запрещается оставлять включенную установку без присмотра. По окончании работы необходимо выключить установку, сложить соединительные провода и обесточить рабочее место.

### 1. Исследование дешифраторов

#### 1.1. Исследование декодера-демультиплектора ИС 1533ИД7

Установить ИС 1533 ИД7 на наборную панель установки IDL-800. Вывод 16 соединить с источником питания  $+5V$ , вывод 8 – с общей шиной. Подавая на адресные входы значения от 000 до 111 с помощью переключателей  $SW$ , зафиксировать выходные сигналы. Результаты исследования свести в табл. 5.7.

Таблица 5.7

Таблица состояний дешифратора 1533ИД7

$E_1$	$E_2$	$E_3$	$A_2$	$A_1$	$A_0$	$D_0$	$D_1$	...	$D_6$	$D_7$

#### 1.2. Исследование полного сумматора на основе дешифратора

Собрать схему сумматора, изображенную на рис. 5.7. Для ИС 1533ИД7 вывод 16 соединить с источником питания  $+5V$ , вывод 8 – с общей шиной. Для ИС 1533ЛА1 вывод 14 соединить с источником питания  $+5V$ , вывод 7 – с общей шиной. Подавая на адресные входы дешифратора значения сигналы от 000 до 111 с помощью переключателей  $SW$ , зафиксировать выходные сигналы. Результаты исследования свести в таблицу истинности.

#### 1.3. Исследование полного вычитателя на основе дешифратора

Синтезировать схему полного вычитателя на базе ИС 1533ИД7. Для ИС 1533ИД7 вывод 16 соединить с источником питания  $+5V$ , вывод 8 – с общей шиной. Для ИС 1533ЛА1, 1533ЛН1 вывод 14 соединить с источником питания  $+5V$ , вывод 7 – с общей шиной. Подавая на адресные входы дешифратора значения сигналы от 000 до 111 с помощью переключателей  $SW$ , выходные сигналы свести в таблицу истинности. Полученную схему зарисовать для отчета.



## 2. Исследование мультиплексора

### 2.1. Исследование работы мультиплексора

Установить ИС 1533КП15 на наборную панель установки IDL-800. Вывод 16 соединить с источником питания  $+5V$ , вывод 8 – с общей шиной. Подавая на адресные входы значения от 000 до 111 с помощью переключателей  $SW0-SW2$  и входные данные (1 или 0), убедиться в правильной работе мультиплексора. Результаты исследования свести в табл. 5.8.

Таблица 5.8

Таблица состояний мультиплексора 1533КП15

$E$	$A_2$	$A_1$	$A_0$	$Y$

2.2. Исследование мультиплексора 1533КП15 в качестве универсального логического элемента для синтеза комбинационных схем:

а) реализовать функцию трех переменных (таблицу истинности задает преподаватель), используя мультиплексор 1533КП15;

б) реализовать функцию четырех переменных (таблицу истинности задает преподаватель), используя мультиплексор 1533КП15. Синтезированные схемы зарисовать для отчета.

### Содержание отчета

1. Наименование и цель работы.
2. Исследуемые в работе схемы с результатами исследований в виде таблиц истинности.
3. Выводы по результатам исследований.

### Контрольные вопросы

1. Определение дешифратора.
2. Способы организации разрешающего входа дешифраторов.
3. Способы наращивания разрядности дешифратора.
4. Что такое дешифратор-демультиплексор?
5. Определение мультиплексора.
6. Почему мультиплексор называют универсальным логическим элементом?
7. Способы наращивания разрядности мультиплексора.

## Литература

1. Браммер, Ю. А. Импульсные и цифровые устройства : учеб. пособие для вузов / Ю. А. Браммер, И. Н. Пашук. – М. : Высш. шк., 1999. – 351 с.
2. Быстров, Ю. А. Электронные цепи и микросхемотехника : учеб. для вузов / Ю. А. Быстров, И. Г. Мироненко. – М. : Высш. шк., 2002. – 384 с.
3. Пухальский, Г. И. Проектирование дискретных устройств на интегральных микросхемах : справочник / Г. И. Пухальский, Т. Я. Новоселова. – М. : Радио и связь, 1990. – 304 с.
4. Храбров, Е. А. Цифровая электроника : учеб. пособие для вузов / Е. А. Храбров, Ю. Е. Котова. – Гомель : ГГТУ им. П. О. Сухого, 2013. – 271 с.
5. Безуглов, Д. А. Цифровые устройства и микропроцессоры : учеб. пособие для вузов / Д. А. Безуглов, И. В. Калиенко. – Ростов н/Д : Феникс, 2006. – 400 с.
6. Белоус, А. И. Основы схемотехники микроэлектронных устройств / А. И. Белоус, В. А. Емельянов, А. С. Турцевич. – М. : Техносфера, 2012. – 472 с.

## ПРИЛОЖЕНИЕ

### Описание лабораторной установки IDL-800 Digital Lab

Лабораторная установка IDL-800 (рис. П.1.1) предназначена для физического макетирования и исследования функционирования цифровых логических устройств, а также может быть использована для исследования аналоговых устройств.

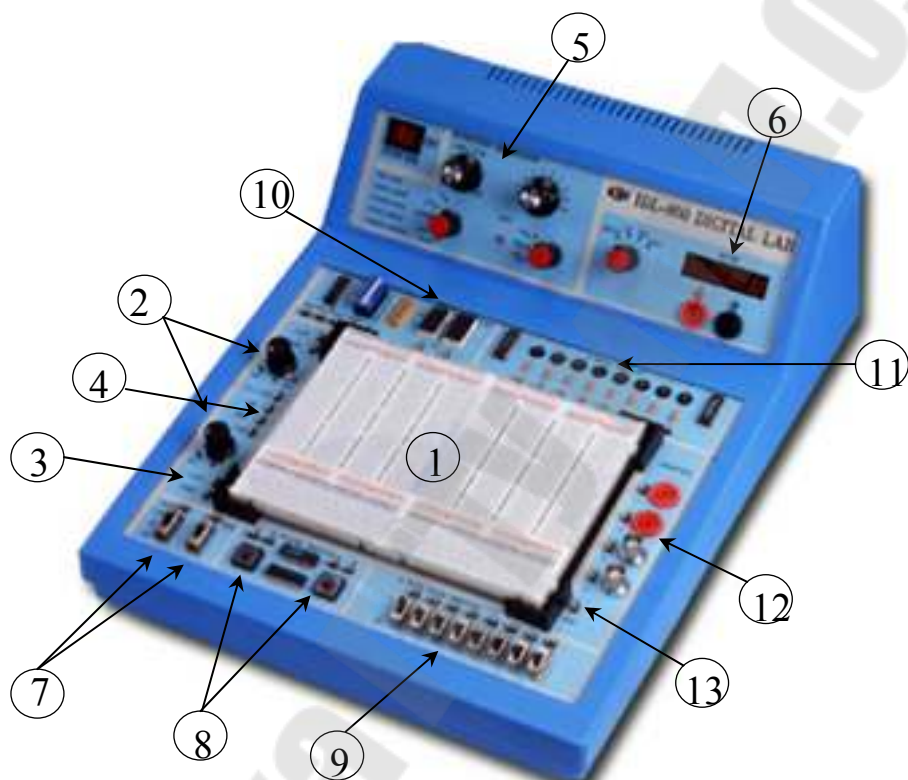


Рис. П.1.1. Лабораторная установка IDL-800 Digital Lab

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:

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  $0 \div +15 \text{ V}$  и  $0 \div -15 \text{ V}$  с максимальным выходным током 300 мА.

3 – нерегулируемый источник электропитания  $+5 \text{ V}$  с максимальным выходным током 1 А.

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

5 – функциональный генератор, формирующий синусоидальный, треугольный, или меандровый сигнал в диапазонах частот: 1–10 Hz, 10–100 Hz, 100 Hz – 1 kHz, 1–10 kHz, 10 Hz – 100 kHz с плавной регулировкой частоты внутри диапазонов. Амплитуда синусоидального сигнала регулируется в пределах от 0 до 4 V. Амплитуда двуполярного треугольного сигнала регулируется в пределах от 0 до 3 V. Амплитуда двуполярного меандрового сигнала регулируется в пределах от 0 до 4 V.

6 – цифровой вольтметр, который имеет 4 диапазона измерений: 1) 0–199,9 V; 2) 0–19,99 V; 3) 0–1,999 V; 4) 0–199,9 mV. Входное сопротивление: 10 МОм в любом диапазоне. ВНИМАНИЕ: цифровой вольтметр не имеет защиты от перегрузки!

7 – два функциональных переключателя  $-5 \text{ V} / 0 \text{ V} / +5 \text{ V}$ .

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

9 – восемь 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.

## СОДЕРЖАНИЕ

Лабораторная работа № 1. Исследование логических элементов.....	3
Лабораторная работа № 2. Синтез комбинационных логических схем по заданной логической функции.....	20
Лабораторная работа № 3. Исследование арифметических устройств.....	28
Лабораторная работа № 4. Исследование преобразователей кодов ...	39
Лабораторная работа № 5. Синтез комбинационных схем на дешифраторах и мультиплексорах.....	51
Литература .....	66
Приложение .....	67

Учебное электронное издание комбинированного распространения

Учебное издание

**Котова Юлия Евгеньевна**  
**Захаренко Леонид Александрович**

**СХЕМОТЕХНИКА  
В СИСТЕМАХ УПРАВЛЕНИЯ.  
ЦИФРОВАЯ СХЕМОТЕХНИКА**

Практикум  
по выполнению лабораторных работ  
для студентов специальности 1-53 01 07  
«Информационные технологии  
и управление в технических системах»  
дневной формы обучения

**Электронный аналог печатного издания**

Редактор *Н. В. Гладкова*  
Компьютерная верстка *И. П. Минина*

Подписано в печать 02.04.19.  
Формат 60x84/16. Бумага офсетная. Гарнитура «Таймс».  
Ризография. Усл. печ. л. 4,18. Уч.-изд. л. 3,53.  
Изд. № 18.  
<http://www.gstu.by>

Издатель и полиграфическое исполнение  
Гомельский государственный  
технический университет имени П. О. Сухого.  
Свидетельство о гос. регистрации в качестве издателя  
печатных изданий за № 1/273 от 04.04.2014 г.  
пр. Октября, 48, 246746, г. Гомель