

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное
учреждение высшего образования
«СЕВАСТОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Кафедра информационных технологий и компьютерных систем

ДИСКРЕТНАЯ МАТЕМАТИКА И КОМПЬЮТЕРНАЯ ЛОГИКА. БУЛЕВА АЛГЕБРА

Методические указания

к выполнению лабораторных работ по дисциплине
для студентов дневной формы обучения
по направлениям подготовки

09.03.01 «Информатика и вычислительная техника»,
09.03.04 «Программная инженерия»

Севастополь
СевГУ
2023

УДК 519.6(076.5)
ББК 22.176я73
Д482

Р е ц е н з е н т :

Д. В. Заморёнова - канд. техн. наук, доцент кафедры ПСАТП

Ответственный за выпуск:

А. А. Брюховецкий - канд. техн. наук, доцент, заведующий кафедрой
информационных технологий и компьютерных систем

Составители: О. В. Ченгарь, А. А. Малицкая

Д482 Дискретная математика и компьютерная логика. Булева алгебра : методические указания к выполнению лабораторных работ по дисциплине для студентов дневной формы обучения по направлениям подготовки 09.03.01 «Информатика и вычислительная техника», 09.03.04 «Программная инженерия» / Севастопольский государственный университет, Институт информационных технологий, кафедра информационных технологий и компьютерных систем ; сост.: О. В. Ченгарь, А. А. Малицкая. – Севастополь : СевГУ, 2023. – 46 с. – Текст : электронный.

В методических указаниях приводятся теоретические сведения и задания для выполнения лабораторных работ в процессе изучения учебной дисциплины «Дискретная математика и компьютерная логика».

УДК 519.6(076.5)
ББК 22.176я73

Методические указания рассмотрены и утверждены на заседании кафедры «Информационные технологии и компьютерные системы», протокол № 5 от 04 февраля 2022 г.

© Ченгарь О. В., Малицкая А. А., сост., 2023
© ФГАОУ ВО «Севастопольский
государственный университет», 2023

СОДЕРЖАНИЕ

1. ЦЕЛЬ И ЗАДАЧИ ВЫПОЛНЕНИЯ ЛАБОРАТОРНЫХ РАБОТ	4
2. ТЕМАТИКА ЛАБОРАТОРНЫХ РАБОТ.....	5
3. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ К ВЫПОЛНЕНИЮ ЛАБОРАТОРНЫХ РАБОТ	6
3.1. Системы счисления. Основные понятия и определения. Классификация, перевод из одной системы счисления в другую	6
3.2. Представление чисел в ЦВМ. Прямой, обратный и дополнительный коды. Выполнение операций сложения, вычитания	12
3.3. Логические функции двух переменных	14
3.4. Построение СДНФ и СКНФ функции.....	15
3.5. Минимизация булевой функции на картах Карно	17
3.6. Минимизация булевой функции методом Квайна-Мак-Класки	19
3.7. Приведение функции к базису «И-НЕ», «ИЛИ-НЕ»	23
3.8. Построение логической схемы функции	23
3.9. Каскадный метод минимизации функционального представления....	24
4. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНЫХ РАБОТ	38
5. ВАРИАНТЫ ЗАДАНИЙ	42
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	46

1. ЦЕЛЬ И ЗАДАЧИ ВЫПОЛНЕНИЯ ЛАБОРАТОРНЫХ РАБОТ

Целью выполнения лабораторных работ является систематизация теоретических знаний, полученных за время обучения, с использованием современных программных и технических средств, а также приобретение и закрепление навыков самостоятельной работы. Лабораторные работы включают в себя ряд заданий, направленных на самостоятельное продолжение изучения основных разделов дискретной математики, которые были рассмотрены в предыдущих разделах курса.

В результате выполнения лабораторных работ студенты должны приобрести следующие практические умения и навыки:

- выбора и использования предложенных методов решения задачи минимизации булевых функций;
- формализации и алгоритмизации задачи;
- планирования, проведения и анализа результатов вычислительных экспериментов;
- оформления технической документации в соответствии с ГОСТ.

2. ТЕМАТИКА ЛАБОРАТОРНЫХ РАБОТ

Темы лабораторных заданий отвечают содержанию и задачам дисциплины «Дискретная математика и компьютерная логика» и непосредственно связаны с тематикой лабораторных занятий, проводимых во втором семестре в соответствии с рабочей программой дисциплины, а также научно-исследовательских и научно-методических работ, которые выполняются на кафедре.

Темы лабораторных работ включают следующие основные типы заданий.

1. Системы счисления. Перевод из одной системы счисления в другую.
2. Представление чисел в ЦВМ. Прямой, обратный и дополнительный коды. Выполнение операций сложения, вычитания.
3. Логические функции двух переменных. Реализация функций формулами.
4. Построение совершенной дизъюнктивной нормальной формы (СДНФ) и совершенной конъюнктивной нормальной формы (СКНФ).
5. Минимизация булевой функции на картах Карно посредством упрощения ДНФ и КНФ функции методом Закревского.
6. Минимизация булевой функции методом Квайна-Мак-Класки в классе ДНФ и КНФ.
7. Каскадный метод минимизации функционального представления схем.
8. Преобразование функции к заданному логическому базису.
9. Логическое проектирование функциональной схемы в заданном базисе.

Задания на лабораторные работы включают перечень задач для реализации, методы решения, варианты для определения входных данных, список рекомендуемой литературы.

3. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ К ВЫПОЛНЕНИЮ ЛАБОРАТОРНЫХ РАБОТ

3.1 Системы счисления. Основные понятия и определения. Классификация, перевод из одной системы счисления в другую.

3.1.1 Системы счисления

Система счисления - совокупность приемов и правил наименования и записи чисел.

Любая, предназначенная для практического применения, система счисления должна обеспечивать:

- возможность представления любого числа;
- единственность представления;
- простоту оперирования числами.

Все системы счисления делят на позиционные и непозиционные.

Непозиционная система счисления – система, для которой значение символа не зависит от его положения в числе.

В *позиционных системах счисления* (ПСС) значение символа зависит от его положения в числе. Любая ПСС характеризуется основанием q .

Основание q позиционной системы счисления – количество знаков или символов, используемых для изображения числа в данной системе. Если q – целое положительное число, позиционная система счисления называется *естественной*.

Любое число в любой позиционной системе счисления можно записать следующим образом:

$$A_{(q)} = a_n \cdot q^n + a_{n-1} \cdot q^{n-1} + \dots + a_1 \cdot q^1 + a_0 \cdot q^0 + a_{-1} \cdot q^{-1} + \dots + a_{-m} \cdot q^{-m} \quad (3.1)$$

или

$$A_{(q)} = \sum_{i=-m}^{i=n} a_i \cdot q^i,$$

где $A_{(q)}$ – произвольное число, записанное в ПСС с основанием q ; $(n+1)$ – количество разрядов целой части числа; m – количество разрядов дробной части числа; a_i – символы ПСС.

Позиции символов (цифр), отсчитываемые от запятой, называются разрядами. В ПСС значение каждого разряда больше значения соседнего справа разряда в число раз, равное основанию q системы.

Длиной числа называют количество позиций (разрядов) в записи числа. Чем выше значение q , тем меньше разрядов требуется для записи числа.

Если заданы основание ПСС q и количество разрядов для записи чисел n , то максимальное и минимальное числа определяют по формулам:

$$A_{q \max} = q^n - 1, \quad A_{q \min} = -(q^n - 1).$$

Интервал между максимальным и минимальным числами называется *диапазоном* представления чисел в заданной ПСС.

Наибольшее распространение в ЭВМ имеет двоичная система счисления. В этой системе используются только две цифры: 0 и 1.

Поскольку человеку удобнее пользоваться десятичными числами, а ЭВМ работает в двоичной системе счисления, то необходим перевод из одной ПСС в другую. Одно и то же число в системах счисления с основаниями q_1 и q_2 можно представить следующим образом:

$$A_{q_1} = \sum_{i=-m}^{i=n} a_i \cdot q_1^i = \sum_{i=-s}^{i=k} b_i \cdot q_2^i = A_{q_2}.$$

Значит, задача перевода чисел из системы счисления с основанием q_1 в систему счисления с основанием q_2 заключается в определении коэффициентов b_i нового ряда, изображающего число в системе с основанием q_2 .

Для любой ПСС основание изображается числом **10** в своей.

3.1.2 Перевод из одной системы счисления в другую

В общем виде целое число в системе с основанием q_2 записывается в виде

$$A_{q_2} = b_k \cdot q_2^k + b_{k-1} \cdot q_2^{k-1} + \dots + b_1 \cdot q_2^1 + b_0 \cdot q_2^0. \quad (3.2)$$

Переписав это выражение по схеме Горнера, получим

$$A_{q_2} = (\dots(b_k \cdot q_2 + b_{k-1}) \cdot q_2 + \dots + b_1) \cdot q_2 + b_0. \quad (3.3)$$

Разделив правую часть выражения (3.3) на величину основания q_2 , получим целую часть $(\dots(b_k \cdot q_2 + b_{k-1}) \cdot q_2 + \dots + b_1)$ и остаток, представляющий собой коэффициент b_0 . Разделив полученное частное на q_2 , также получим целую часть от этого деления и остаток – коэффициент b_1 .

Таким образом, перевод целых чисел из ПСС с основанием q_1 в ПСС с основанием q_2 заключается в следующем:

исходное число, записанное в системе с основанием q_1 , разделить на q_2 , выраженное в системе счисления с основанием q_1 , и определить остаток;

частное снова разделить на q_2 и т.д., пока полученное частное не станет меньше, чем q_2 .

Искомое число записывается в таком порядке: последнее частное, последний остаток, предпоследний остаток и т.д., первый остаток.

Пример: Перевести десятичное число $A_{(10)} = 46$ в двоичную и шестнадцатеричную системы счисления.

Решение.

$$\begin{array}{l}
 1) \quad \begin{array}{r|l} 46 & 2 \\ \hline 46 & 23 \\ \hline 22 & 11 \\ \hline 11 & 5 \\ \hline 5 & 2 \\ \hline 4 & 2 \\ \hline 2 & 1 \\ \hline 2 & 1 \\ \hline 0 & \end{array} \\
 \begin{array}{l} b_0 = 0 \\ b_1 = 1 \\ b_2 = 1 \\ b_3 = 1 \\ b_4 = 0 \end{array}
 \end{array}
 \quad
 \begin{array}{l}
 2) \quad \begin{array}{r|l} 46 & 16 \\ \hline 32 & 2 = b_1 \\ \hline \end{array} \\
 b_0 = 14
 \end{array}$$

Ответ: $A_{(2)} = 101110$; $A_{(16)} = 2E$, т.к. на основании таблицы 3.1 можно записать $b_1 = 2$; $b_0 = E$.

Обратный перевод в десятичную систему счисления можно осуществить, используя соотношение (3.2).

Таблица 3.1 – Наиболее часто используемые системы счисления

Таблица 3.1

$q=10$	$q=2$	$q=8$	$q=16$
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10

Пример: Перевести двоичное число $A_{(2)} = 101110$ и шестнадцатеричное число $A_{(16)} = 2E$ в десятичную систему счисления.

Решение.

$$1. A_{(10)} = 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 46.$$

2. На основании таблицы 1.1 заменим шестнадцатеричную цифру **E** ее десятичным эквивалентом 14, тогда $A_{(10)} = 2 \cdot 16^1 + 14 \cdot 16^0 = 46$.

Ответ: $A_{(10)} = 46$.

Остается заметить, что для перевода чисел из ПСС с основанием q_1 в ПСС с основанием q_2 , ни одна из которых не является десятичной, все арифметические операции (деление, умножение, сложение и вычитание) нужно производить в ПСС с основанием q_1 . Например, при переводе числа из пятеричной системы в троичную все вычисления должны производиться в пятеричной системе. Так как для человека привыкшего к десятичной системе, это крайне неудобно, то в этом случае целесообразно воспользоваться десятичной системой в качестве промежуточной, т.е. сделать перевод из пятеричной системы в десятичную, а затем из десятичной в троичную.

Перевод правильных дробей из одной ПСС в другую

В общем виде число, представляющее собой правильную дробь, записанное в ПСС с основанием q_1 , имеет вид

$$A_{q_1} = a_{-1} \cdot q_1^{-1} + \dots + a_{-m} \cdot q_1^{-m}.$$

Тогда в новой системе счисления с основанием q_2 это число будет изображено как

$$A_{q_2} = b_{-1} \cdot q_2^{-1} + \dots + b_{-s} \cdot q_2^{-s} \quad (3.4)$$

Если переписать выражение (1.4) по схеме Горнера, то получим:

$$A_{q_2} = q_2^{-1} \cdot (b_{-1} + q_2^{-1} \cdot (b_{-2} + \dots + q_2^{-1} \cdot b_{-s}) \dots). \quad (3.5)$$

При умножении правой части выражения на q_2 получим новую неправильную дробь, в целой части которой будет цифра b_{-1} (умножение на величину q_2 используется вместо деления на величину $q_2^{-1} = 1/q_2$). Затем умножив оставшуюся дробную часть на величину основания q_2 , получим дробь, в целой части которой будет b_{-2} , и т.д. Повторяя процесс умножения k раз, найдем k цифр числа в новой системе счисления. При этом все действия должны выполняться по правилам q_1 -арифметики, и следовательно, в целой части получающихся дробей будут появляться эквиваленты цифр новой системы счисления, записанные в исходной системе счисления.

При переводе правильных дробей из одной ПСС в другую можно получить дробь в виде бесконечного ряда (например, при переводе числа $A_{(10)} = 0,8$ в двоичную систему получается бесконечная дробь).

Пример: Перевести десятичную дробь $A_{(10)} = 0,68$ в двоичную систему счисления с точностью до 6 двоичных разрядов и в шестнадцатеричную – с точностью до 4 шестнадцатеричных разрядов.

Решение.

$$\begin{array}{l}
 1) \quad \begin{array}{r|l} 0, & 68 \\ \hline & 2 \\ \hline b_{-1} = 1, & 36 \\ & 2 \\ \hline b_{-2} = 0, & 72 \\ & 2 \\ \hline b_{-3} = 1, & 44 \\ & 2 \\ \hline b_{-4} = 0, & 88 \\ & 2 \\ \hline b_{-5} = 1, & 76 \\ & 2 \\ \hline b_{-6} = 1, & 52 \end{array} \\
 2) \quad \begin{array}{r|l} 0, & 68 \\ \hline & 16 \\ \hline b_{-1} = 10, & 88 \\ & 16 \\ \hline b_{-2} = 14, & 08 \\ & 16 \\ \hline b_{-3} = 1, & 28 \\ & 16 \\ \hline b_{-4} = 4, & 48 \end{array}
 \end{array}$$

Ответ: $A_{(2)} = 0,101011$.

Заменив десятичные цифры, получающиеся в целой части, на шестнадцатеричные, можно записать:

$$b_{-1} = A; b_{-2} = E; b_{-3} = 1; b_{-4} = 4. \quad A_{(16)} = 0,AE14.$$

Перевод в десятичную систему можно осуществить, пользуясь соотношением (1.4). В случае, если перевод из десятичной системы в ПСС с основанием q_i осуществлялся с точностью до k разрядов, то при обратном переводе возникает погрешность.

Пример: Перевести двоичное число $A_{(2)} = 0,101011$ и шестнадцатеричное число $A_{(16)} = 0,AE14$ в десятичную систему счисления.

Решение.

$$1. \quad A_{(10)} = 1 \cdot 2^{-1} + 0 \cdot 2^{-2} + 1 \cdot 2^{-3} + 0 \cdot 2^{-4} + 1 \cdot 2^{-5} + 1 \cdot 2^{-6} = 0,671875.$$

Ответ: $A_{(10)} = 0,671875$.

2. На основании таблицы 1.1 заменяем шестнадцатеричные цифры их десятичными эквивалентами:

$$A_{(10)} = 10 \cdot 16^{-1} + 14 \cdot 16^{-2} + 1 \cdot 16^{-3} + 4 \cdot 16^{-4} = 0,679988$$

Ответ: $A_{(10)} = 0,679988$.

Для перевода неправильных дробей из одной системы счисления в другую необходим отдельный перевод целой и дробной частей по правилам, описанным выше. Полученные результаты записывают в виде новой неправильной дроби в ПСС с основанием q_2 .

3.1.4 Системы счисления с кратными основаниями

ПСС с основаниями q_1 и q_2 называются системами с кратными основаниями, если выполняется равенство $q_1 = q_2^k$, где $k = 2, 3, 4, \dots$

Таким образом, для двоичной системы счисления кратными будут системы с основаниями 4, 8, 16, 32 и т.д. Для троичной ПСС кратными будут системы с основаниями 9, 27, 81 и т.д.

Правила перевода чисел из системы в систему, основания которых кратные, основываются на табличном методе перевода, при котором исходное число с основанием q_1 (причем $q_1 < q_2$) разбивается на группы по k разрядов, двигаясь от запятой вправо и влево. Затем каждую получившуюся диаду, триаду или тетраду цифр заменяют эквивалентной цифрой системы с основанием q_2 . Крайние группы при необходимости дополняют нулями. Например, для перевода из двоичной системы счисления в четверичную, каждые две двоичные цифры (диады) заменяются их четверичным эквивалентом.

Пример: Перевести двоичное число $A_{(2)} = 10011101,1001011$ в четверичную и шестнадцатеричную системы.

Решение.

$$1. A_{(2)} = \underbrace{10}_{2} \underbrace{01}_{1} \underbrace{11}_{3} \underbrace{01}_{1}, \underbrace{10}_{2} \underbrace{01}_{1} \underbrace{01}_{1} \underbrace{10}_{2}$$

Ответ: $A_{(4)} = 2131,2112$.

$$2. A_{(2)} = \underbrace{1001}_{9} \underbrace{1101}_D, \underbrace{1001}_{9} \underbrace{0110}_6$$

Ответ: $A_{(16)} = 9D,96$.

В случае обратного перевода из q_1 в q_2 (если $q_1 > q_2$) каждая цифра числа в системе с основанием q_1 заменяется соответствующей диадой, триадой, тетрадой и т.д. цифр системы с основанием q_2 .

Пример: Перевести восьмеричное число $A_{(8)} = 305,62$ в двоичную систему счисления.

Решение.

$$A_{(8)} = \underbrace{3}_{011} \underbrace{0}_{000} \underbrace{5}_{101}, \underbrace{6}_{110} \underbrace{2}_{010}$$

Ответ: $A_{(2)} = 11000101,11001$.

Шестнадцатеричная (и реже восьмеричная) система счисления используется при составлении программ для более короткой и удобной записи двоичных кодов команд, т.к. эти системы не требуют специальных операций для перевода в двоичную систему.

При переводе десятичных кодов в двоичные, для ускорения процесса перевода целесообразно воспользоваться промежуточной системой счисления, имеющей основание, кратное 2.

3.2. Представление чисел в ЦВМ. Прямой, обратный и дополнительный коды. Выполнение операций сложения, вычитания.

3.2.1. Представление чисел в ЦВМ

Все числа в ЦВМ представляются в двух формах: с фиксированной точкой и с плавающей точкой.

Целые числа (представляется в форме с фиксированной точкой) имеют вид:

$$\langle \text{знак} \rangle \langle \text{абсолютное значение целого числа} \rangle.$$

Знак занимает один разряд: 0 означает “+”, 1 означает “-”. Значение целого числа занимает все остальные (n-1) разряды слова ($n \in \{16, 32, 64\}$), начиная с правого разряда.

Вещественные числа (представляется в форме с плавающей точкой) имеет вид:

$$\langle \text{порядок} \rangle \langle \text{мантисса} \rangle.$$

Мантисса представляет знак числа и все его старшие значащие цифры, которые удастся разместить в разрядах слова, отведенных для представления мантиссы. Старшие цифры числа занимают левые разряды слова. Мантисса отвечает следующим ограничениям (в этом случае она называется *нормализованной*):

$$1/q \leq | \langle \text{мантисса} \rangle | < 1. \quad (3.6)$$

Чтобы сохранить значение числа, используется *порядок*, который указывает положение точки в числе. Порядок представляется как целое число со знаком, и равен числу разрядов, на которое приходится сдвинуть точку в числе, чтобы превратить его в мантиссу.

В ЦВМ порядки и мантиссы вычисляются для чисел, *переведенных в двоичную систему счисления*, и представляются в двоичной системе. В мантиссе нуль целых и точка не ставятся, а только подразумеваются, мантисса содержит знак и значащие цифры числа. Порядок представляется целым числом со знаком. Общая структура представления чисел с плавающей точкой имеет следующий вид:

$$\langle \text{знак порядка} \rangle \langle \text{порядок} \rangle \quad \langle \text{знак числа} \rangle \langle \text{мантисса} \rangle$$

Пример:

Двоичное число	Форма плавающей точки	
+ 1011.100101	+ 4	+ 0 0100 0 1011100101
- 011.100101	+ 2	- 0 0010 1 1110010100
- 0.0000010111	- 5	- 1 0101 1 1011100000

3.2.2. Прямой, обратный и дополнительный коды

Основная операция в ЦВМ – сложение. Операция вычитания чисел сводится к выполнению операции сложения с учетом знаков – алгебраическому сложению. Операции умножения и деления двоичной арифметики сводятся к выполнению сложения, вычитания и сдвига. Таким образом, все операции арифметики сводятся к алгебраическому сложению и сдвигу.

В большинстве ЭВМ основное устройство - многоразрядный двоичный сумматор, построенный на основе полных одноразрядных двоичных сумматоров. Арифметика двоичной системы определяется таблицами сложения и умножения:

+	0	1
0	0	1
1	1	¹ 0

x	0	1
0	0	0
1	0	1

¹0 означает, что разряд суммы равен 0, однако возникает единица переноса в следующий разряд.

Для представления чисел в ЦВМ используются прямой, обратный и дополнительный коды.

Прямым кодом числа называется представление в виде знака:

<знак> < абсолютное значение числа > ,

Обратным кодом числа называется такое преобразование прямого кода, в котором положительное число не изменяется, а для отрицательного числа все разряды модуля числа инвертируются, т.е. 0 заменяется на 1, а 1 - на 0, знак остается 1.

Пример:

$$A_{\text{пр}} = 1 \ 11101010, \quad A_{\text{обр}} = 1 \ 00010101,$$

$$A_{\text{пр}} = 0 \ 11101010 = A_{\text{обр}} = 0 \ 11101010$$

В обратном коде число 0 имеет два изображения: 0 000...0 и 1 111...1. Для обеспечения однозначности представления при получении отрицательного нуля, т.е. комбинации 1 111...1, необходимо эту комбинацию заменить на 0 000...0.

Дополнительный код отрицательного числа получается прибавлением единицы к младшему разряду обратного кода этого числа. Дополнительный код положительного числа совпадает с прямым.

$$A_{\text{пр}} = 1 \ 101011000$$
$$A_{\text{доп}} = 1 \ 010101000$$

Для алгоритмического сложения чисел с учетом знаков используется обратный или дополнительный коды. Знаковые разряды участвуют в сложении наряду с числовыми.

При сложении в обратных кодах, если возникает единица переноса из знакового разряда, то она суммируется к младшему разряду суммы, результат получается в обратном коде.

Признаком переполнения разрядной сетки сумматора при сложении положительных чисел является отрицательный результат, а при сложении отрицательных чисел - положительный результат.

При сложении в дополнительных кодах возникающая единица переноса из знакового разряда игнорируется, результат получается в дополнительном коде.

Признак переполнения разрядной сетки тот же, что и для обратного кода, т.е. знак результата противоположен знакам операндов. Кроме этого, переполнение при сложении в дополнительном коде может возникнуть и при совпадении знаков исходных операндов и знака результата. Это происходит при сложении отрицательных чисел, сумма которых по модулю равна $2n$. В этом случае возникает запрещенная комбинация дополнительного кода 1.00...00.

Переводы из обратного и дополнительного кодов в прямой осуществляются по правилам прямого перевода.

$$\begin{array}{ll}
 A = -234 & A_{\text{пр}} = 1 \ 11101010, & A_{\text{обр}} = 1 \ 00010101 \\
 B = -10 & B_{\text{пр}} = 1 \ 00001010 & B_{\text{обр}} = 1 \ 11110101 \\
 \hline
 C_{\text{обр}} = A_{\text{обр}} + B_{\text{обр}} = & 1 \ 0000 \ 1010 \\
 & + \qquad \qquad \qquad 1 \\
 C_{\text{пр}} = & 1 \ 1111 \ 0100 \\
 C = & -244
 \end{array}$$

3.3. Логические функции двух переменных

В таблице 3.2 представлены все булевы функции двух переменных.

Таблица 3.2

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

В таблице 3.3 представлены названия и даны пояснения по каждой из 16 функций.

Таблица 3.3

Ф-ция	Название	Описание через $\wedge, \vee, \neg$	Читается, как...
f_0	константа $\emptyset$	$\emptyset$	$\emptyset$
f_1	конъюнкция x_1 и x_2	$x_1 \wedge x_2$	x_1 и x_2
f_2	отрицание по x_2	$x_1 \wedge \bar{x}_2$	<i>неверно, что если x_1, то x_2</i>
f_3	переменная x_1	x_1	x_1
f_4	отрицание по x_1	$\bar{x}_1 \wedge x_2$	<i>неверно, что если x_2, то x_1</i>
f_5	переменная x_2	x_2	x_2
f_6	сумма по mod 2 ($x_1 \oplus x_2$)	$x_1 \wedge \bar{x}_2 \vee \bar{x}_1 \wedge x_2$	x_1 <i>неравнозначно</i> x_2
f_7	дизъюнкция x_1 и x_2	$x_1 \vee x_2$	x_1 <i>или</i> x_2
f_8	стрелка Пирса $x_1 \downarrow x_2$	$\bar{x}_1 \wedge \bar{x}_2$	<i>ни x_1, ни x_2</i>
f_9	эквиваленция $x_1 \sim x_2$	$\bar{x}_1 \wedge \bar{x}_2 \vee x_1 \wedge x_2$	x_1 <i>равнозначно</i> x_2
f_{10}	отрицание x_2	$\bar{x}_2$	<i>не x_2</i>
f_{11}	импликация по x_2 ($x_2 \rightarrow x_1$)	$\bar{x}_2 \vee x_1$	<i>если x_2, то x_1</i>
f_{12}	отрицание x_1	$\bar{x}_1$	<i>не x_1</i>
f_{13}	импликация по x_1 ($x_1 \rightarrow x_2$)	$\bar{x}_1 \vee x_2$	<i>если x_1, то x_2</i>
f_{14}	штрих Шеффера ($x_1 x_2$)	$\bar{x}_1 \vee \bar{x}_2$	<i>не верно, что x_1 и x_2</i>
f_{15}	константа 1	1	1

3.4 Построение СДНФ и СКНФ функции

Кроме табличного задания функции алгебры логики применяются различные аналитические методы. К ним относятся – дизъюнктивная и конъюнктивная формы.

Набор значений переменных можно обобщенно представить, как последовательность констант $\sigma_i, 1 \leq i \leq n: \sigma_1 \sigma_2 \dots \sigma_n$.

Введем условное обозначение:

$$x_i^{\sigma_i} = \begin{cases} x_i, & \text{при } \sigma_i = 1 \\ \bar{x}_i, & \text{при } \sigma_i = 0 \end{cases} \quad (3.7)$$

где совокупность x_i – порождающие множества;

σ_i – первичный терм;

n – число порождающих множеств.

Произведение вида $x_1^{\sigma_1} x_2^{\sigma_2} \dots x_n^{\sigma_n}$ называют **элементарной конъюнкцией** и записывают в векторной форме следующим образом:

$$K_{\tilde{\sigma}}(\tilde{x}) = x_1^{\sigma_1} x_2^{\sigma_2} \dots x_n^{\sigma_n} \quad (3.8)$$

В такой записи вектор переменных $\tilde{x}$ определяет, какие переменные образуют конъюнкцию, а набор $\tilde{\sigma}$ задает порядок расстановки знаков инверсии (отрицания) над ними.

Формула вида $D = \bigvee_{i=1}^k K_i$, где K_i - различные элементарные конъюнкции, называется **дизъюнктивной нормальной формой (ДНФ)**.

При $f(x) \neq 0$ тождественно, ее можно представить в виде:

$$f(x_1, \dots, x_n) = \bigvee_{\substack{\sigma_1 \dots \sigma_n \\ f(\sigma_1 \dots \sigma_n)=1}} x_1^{\sigma_1} \dots x_n^{\sigma_n} \quad (3.9)$$

Такое разложение функции носит название **совершенной дизъюнктивной нормальной формы (СДНФ)**. Т.е. СДНФ называется ДНФ, в которой нет равных элементарных конъюнкций, и все они содержат одни и те же переменные, причём каждую переменную только один раз (возможно с отрицанием).

Длина СДНФ равна числу наборов, на которых функция $f(x) = 1$.

Алгоритм построения СДНФ:

1) Выбрать в таблице функции все наборы аргументов, на которых функция обращается в единицу

2) Вычислить конъюнкции, соответствующие этим наборам аргументов. При этом, если аргумент x_i входит в данный набор как 1, он вписывается без изменения в конъюнкцию, соответствующую данному набору. Если x_i входит как 0, то в конъюнкцию вписывается его отрицание.

3) Все полученные конъюнкции соединены между собой знаками дизъюнкции.

Соответственно выражение

$$D_{\tilde{\sigma}}(\tilde{x}) = x_1^{\sigma_1} \vee x_2^{\sigma_2} \vee \dots \vee x_n^{\sigma_n} \quad (3.10)$$

где $\tilde{\sigma} = (\sigma_1, \sigma_2, \dots, \sigma_n)$ – какой-либо двоичный набор, причем среди переменных x_i могут быть совпадающие, называется **элементарной дизъюнкцией**.

Формула вида $K = \bigwedge_{i=1}^k D_i$, где D_i - различные элементарные дизъюнкции, называется **конъюнктивной нормальной формой (КНФ)**. Т.е. всякую конъюнкцию элементарных дизъюнкций называют конъюнктивной нормальной формой, то есть КНФ.

При $f(x) \neq 1$ тождественно, ее можно представить в виде:

$$f(x_1, \dots, x_n) = \bigwedge_{\substack{\sigma_1 \dots \sigma_n \\ f(\sigma_1 \dots \sigma_n)=0}} x_1^{\bar{\sigma}_1} \dots x_n^{\bar{\sigma}_n} \quad (3.11)$$

Такое разложение функции носит название **совершенной конъюнктивной нормальной формы (СКНФ)**. Т.е. совершенной КНФ (СКНФ) называется КНФ, в которой нет равных элементарных дизъюнкций, и все они содержат одни и те же переменные, причём каждую переменную только один раз (возможно с отрицанием).

Так как существует взаимно однозначное соответствие между таблицей истинности и СДНФ (СКНФ) функции $f(x_1, \dots, x_n)$, то **СДНФ (СКНФ) – единственна.**

Пример: Построить СДНФ и СКНФ $f(x) = (10001100)$

x_1	x_2	x_3	$f(\tilde{x})$
0	0	0	1
0	0	1	0
	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

СДНФ:

$$f(x) = (10001100) = \\ = \bar{x}_1 \bar{x}_2 \bar{x}_3 \vee x_1 \bar{x}_2 \bar{x}_3 \vee x_1 \bar{x}_2 x_3,$$

СКНФ:

$$f(x) = (x_1 \vee x_2 \vee \bar{x}_3)(x_1 \vee \bar{x}_2 \vee x_3)(x_1 \vee \bar{x}_2 \vee \bar{x}_3)(\bar{x}_1 \vee \bar{x}_2 \vee x_3)(\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3).$$

3.5 Минимизация булевой функции на картах Карно

Матричная (двумерная) форма наиболее удобна при работе на листе бумаги. Для сопоставления 2^n наборов элементам матрицы множество x переменных разбивается на два подмножества младших и старших переменных:

$$x = \{x_1, \dots, x_n\} = \left\{x_1, \dots, x_{\left\lfloor \frac{n}{2} \right\rfloor}\right\} \cup \left\{x_{\left\lfloor \frac{n}{2} \right\rfloor + 1}, \dots, x_n\right\},$$

где $\left\lfloor \frac{n}{2} \right\rfloor$ – наименьшее целое, большее или равное $\frac{n}{2}$.

Наборы значений множества $\{x_1, \dots, x_{\left\lfloor \frac{n}{2} \right\rfloor}\}$ младших переменных сопоставляются столбцам матрицы, наборы старших – строкам. Данное сопоставление можно провести упорядочиванием множества наборов.

Широко распространены два варианта упорядочивания: использования **последовательного двоичного кода** и использование **кода Грея**.

Принцип построения кода Грея:

- 1) Последовательность начинается с нулевого набора (0...0)
- 2) При переходе от очередного набора к следующему изменяется значение только одной переменной – той младшей переменной, которая приводит к получению нового кода по отношению ко всем ранее построенным. Построенная в результате таблица – карта Карно.

Этапы заполнения карты Карно в классе ДНФ.

- 1) Все наборы, присутствующие в функции заносятся в карту с помощью единиц в соответствующие клетки.
- 2) Выделяют интервалы на карте по следующим правилам:
 - а) в один интервал объединяют только соседние единицы по вертикали или горизонтали;
 - б) в один интервал можно объединить 2^k единиц, где $k=0,1,2,3,4,\dots$;
 - с) карта циклически замкнута по вертикали и горизонтали;

d) в выделенный интервал должно быть объединено максимально возможное количество единиц.

Таким образом, процесс минимизации сводится к нахождению наиболее крупных групп из 2^k ($k=0,1,2,3,4,\dots$) соседних единичных клеток. Причем каждая из единичных клеток должна входить в некоторую группу (блок, контур), а общее количество таких максимальных групп должно быть минимально. Простой импликант соответствующий максимальному блоку будет содержать в себе символы тех переменных, значения истинности которых совпадают у всех объединенных клеток.

Пример: Минимизировать функцию трех переменных методом карты Карно в классе ДНФ.

x_3	x_2	x_1	$f(\tilde{x})$
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

Строим карту Карно для трех переменных и отмечаем единицей наборы, заданные функцией.

Затем выделяем максимальные интервалы

		x_1			
		$x_2 x_1$			
		00	01	11	10
x_3	0	1	1	0	1
x_3	1	1	0	1	1

x_2

$$f(x_1 x_2 x_3) = \bar{x}_3 \bar{x}_2 \vee x_3 x_2 \vee \bar{x}_1.$$

При правильном объединении функцию больше минимизировать невозможно.

Карта Карно для $n=5$:

$x_3x_2x_1$		x_3							
x_5x_4		000	001	011	010	110	111	101	100
x_5	00								
	01								
	11								
	10								
		x_1			x_1				
		x_2							

Особенности заполнения карты Карно в классе КНФ.

1. Все наборы, отсутствующие в функции ($=0$) заносятся в карту с помощью нулей в соответствующие клетки.
2. Выделяют интервалы на карте по таким же правилам, как и в классе ДНФ (только 0).
3. Результат записывается в следующем виде:
 - выписываются дизъюнкции, соответствующие выделенным интервалам: если x_i входит как 0, он вписывается без изменений в дизъюнцию, если x_i входит как 1, то в дизъюнцию вписывается его отрицание;
 - все полученные дизъюнкции соединяются между собой знаками конъюнкции

3.6 Минимизация булевой функции методом Квайна-Мак-Класки

Теорема Квайна. Если, исходя из СДНФ булевой функции, провести все возможные операции склеивания конъюнкций этой формы, затем всех возможных результатов склеивания, наконец выполнять все поглощения, получится сокращенная ДНФ.

Основа метода состоит из пяти положений:

- склеиваются могут только наборы соседних классов M_1^i и M_1^{i+1} ($0 \leq i \leq n - 1$);
- если организовывать процесс склеивания так, что результаты склеиваний наборов помещаются в новое множество, то наборы исходного множества и нового множества не склеиваются; при склеивании наборов M_1^i и M_1^{i+1} получается новое множество $M_1^{i \text{ н}}$ наборов, содержащих i единиц;
- если наборы склеиваются, то они могут участвовать в других склеиваниях, но должны считаться поглощенными;

- если наборы исходного множества не склеивались, то они уже не будут поглощаться, т.е. соответствующие им конъюнкции являются простыми импликантами и должны быть включены в сокращенную ДНФ;
- новое множество результатов склеиваний может в следующей итерации считаться исходным и с наборами этого множества снова нужно производить все возможные склеивания.

Алгоритм построения множества наборов (максимальных), соответствующим конъюнкциям сокращенной ДНФ:

1. Выписывается множество M_1 наборов и разбивается на классы M_1^i ($0 \leq i \leq n$). Исходное множество M_p максимальных наборов пусто.
2. Производятся все возможные склеивания наборов множеств M_1^i и M_1^{i+1} . Склеивающиеся наборы отмечаются, результат склеивания записывается с приведением подобных во множество M_1^{i+1} . Этот пункт выполняется для всех $i = 1, 2, \dots, n - 1$.
3. Если во множествах M_1^i ($0 \leq i \leq n$) есть неотмеченные наборы, то они вносятся во множество M_p максимальных наборов.
4. Если множества M_1^{i+1} не пусты, то индекс i уничтожается и для них выполняются пп.2.-4, иначе множество максимальных наборов построено.

Таким образом реализуется первый этап минимизации булевых функций. На втором этапе строится таблица покрытий и решается задача о наилучшем решении в зависимости от критерия (всех минимальных или всех кратчайших).

Столбцы таблицы покрытий ставятся в соответствие наборам множества M_1 , строки – наборам множества M_p .

Элементы T_{ij} таблицы покрытий = 1, если набор $m_j \in M_1$ входит в разложение $m_i \in M_p$. Для удобства наборы множества M_p можно обозначить буквами латинского алфавита.

Для поиска минимального покрытия вес строки определяется числом внешних переменных наборов множества M_p .

Пример Минимизировать функцию 4-х переменных методом Квайна-Мак-Класки

	x_1	x_2	x_3	x_4	$f(\tilde{x})$
	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	1
3	0	0	1	1	1
4	0	1	0	0	1
	0	1	0	1	0
5	0	1	1	0	1
6	0	1	1	1	1
7	1	0	0	0	1
8	1	0	0	1	1
9	1	0	1	0	1
	1	0	1	1	0
10	1	1	0	0	1
11	1	1	0	1	1
	1	1	1	0	0
12	1	1	1	1	1

	M_1	
1.	0 0 0 1	✓
2.	0 0 1 0	✓
3.	0 1 0 0	✓
4.	1 0 0 0	✓
5.	0 0 1 1	✓
6.	0 1 1 0	✓
7.	1 0 0 1	✓
8.	1 0 1 0	✓
9.	1 1 0 0	✓
10.	0 1 1 1	✓
11.	1 1 0 1	✓
12.	1 1 1 1	✓

	M_1^H	
	1 - 0 0	✓
	1 0 - 0	✓
	1 0 0 -	✓
	- 1 0 0	
	0 1 - 0	
	- 0 1 0	
	0 - 1 0	✓
	0 0 1 -	✓
	0 0 - 1	
	- 0 0 1	
	1 1 0 -	✓
	0 1 1 -	✓
	0 - 1 1	✓
	1 - 0 1	✓
	- 1 1 1	
	1 1 - 1	

	M_1^{HH}	
	1 - 0 -	
	0 - 1 -	

$$M_p = \{1-0-, 0-1-, 10-0, -100, 01-0, -010, 00-1, -001, -111, 11-1\}$$

Составляем таблицу Квайна (в строках - M_p , в столбцах - M_1):

		1	2	3	4	5	6	7	8	9	10	11	12	
A	1-0-				1			1		1		1		2
B	0-1-		1			1	1				1			2
C	10-0				1				1					3
D	-100			1						1				3
E	01-0			1			1							3
F	-010		1						1					3
G	00-1	1				1								3
H	-001	1						1						3
I	-111										1		1	3
J	11-1											1	1	3

После решения задачи покрытия таблицы Квайна получаем два минимальных покрытия: AEFGI и BCDHJ ($S=14$).

$$f_1(x_1x_2x_3x_4) = x_1\bar{x}_3 \vee \bar{x}_1x_2\bar{x}_4 \vee \bar{x}_2x_3\bar{x}_4 \vee \bar{x}_1\bar{x}_2x_4 \vee x_2x_3x_4$$

$$f_2(x_1x_2x_3x_4) = \bar{x}_1x_3 \vee x_1\bar{x}_2\bar{x}_4 \vee x_2\bar{x}_3\bar{x}_4 \vee \bar{x}_2\bar{x}_3x_4 \vee x_1x_2x_4$$

Минимизация в классе КНФ заключается в том, что надо найти такую КНФ для заданной булевой функции, которая содержала бы минимальное число букв (минимальная КНФ). Очевидно, что такая КНФ будет содержать и минимальное число дизъюнкций (является кратчайшей КНФ).

Аналогично ДНФ можно определить и безыбыточную КНФ.

Алгоритм построения множества наборов, соответствующих дизъюнкциям сокращенной КНФ, аналогичен алгоритму для ДНФ. Только на первом этапе выписывается множество M_0 наборов и разбивается на классы M_0^i ($0 \leq i \leq n$).

На втором этапе строится аналогичная таблица покрытий и решается задача о наилучшем решении в зависимости от критерия (всех минимальных или всех кратчайших), только столбцы таблицы покрытий ставятся в соответствие наборам множества M_0 .

Полученный результат записывается следующим образом:

– выписываются дизъюнкции, соответствующие полученному решению (интервалам): если x_i входит как 0, он вписывается без изменений в дизъюнкцию, если x_i входит как 1, то в дизъюнкцию вписывается его отрицание;

– все полученные дизъюнкции соединяются между собой знаками конъюнкции

3.7 Приведение функции к базису «И-НЕ», «ИЛИ-НЕ»

Функционально-полная система алгебры Буля – набор функций P_k , с помощью которых может быть выражена любая функция из P_k .

Базисом алгебры Буля называется функционально-полная система, которая перестает быть таковой при исключении из неё любой функции.

Примером алгебры с функционально-полной системой, но не являющейся базисом является функция вида :

$$A = \langle M, \vee, \&, - \rangle$$

На основании законов де Моргана из неё можно получить алгебры с базисами «ИЛИ-НЕ» и «И-НЕ»: $A_1 = \langle M, \vee, - \rangle$, $A_2 = \langle M, \&, - \rangle$.

Пример преобразования функции к базису «ИЛИ-НЕ»:

$$f(x_1, x_2, x_3) = x_1 \bar{x}_2 x_3 \vee \bar{x}_1 x_3 = \bar{\bar{x}_1} \vee \bar{x}_2 \vee \bar{x}_3 \vee \bar{x}_1 \vee \bar{x}_3$$

3.8 Построение логической схемы функции

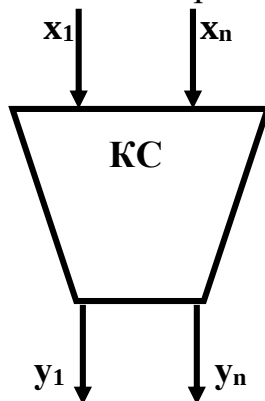
Комбинаторной системой называется автомат выполняющий преобразование x в y , где:

$x = \{ x_1, x_2, \dots, x_n \}$ – множество входных булевых переменных

$y = \{ y_1, y_2, \dots, y_n \}$ – множество выходных булевых функций

Комбинаторная система представляет собой схему, состоящую из логических элементов. Каждый логический элемент реализует элементарную булеву функцию. Минимальная совокупность логических элементов, достаточная для реализации произвольной системы, называется элементарным базисом.

Отличительной особенностью комбинаторных систем является то, что выходной сигнал в момент времени T является функцией только входного сигнала в момент времени T . В общем виде на структурных и функциональных схемах комбинаторные системы изображаются в виде перевернутой трапеции.



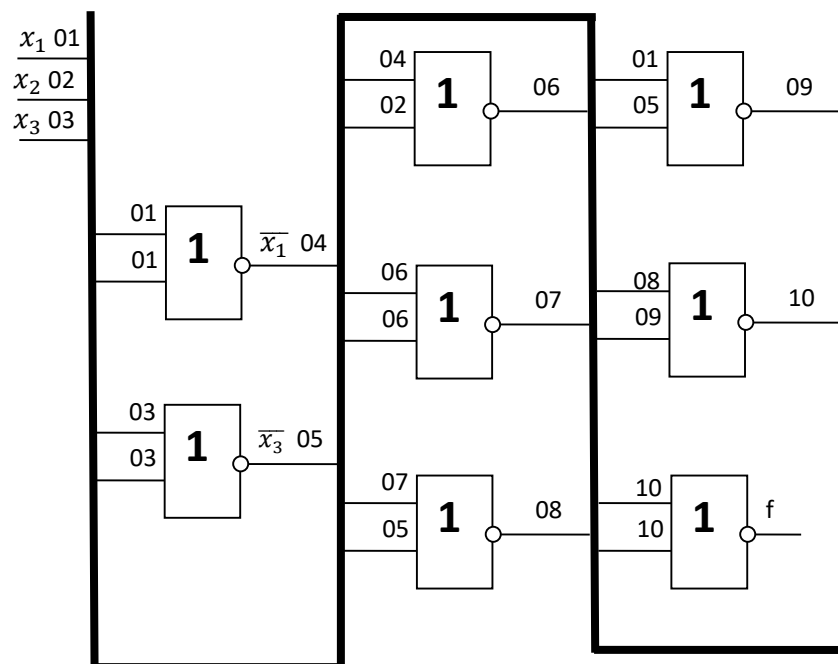
Алгоритм синтеза логических функций с малым числом переменных в заданном базисе:

- 1) функция записывается в СДНФ по заданной таблице истинности;

- 2) находится минимальная форма (одним из рассмотренных выше методом);
- 3) операции дизъюнкции, конъюнкции и отрицания выражаются через функции заданного базиса;
- 4) записывается выражение функции в элементах базиса и строится логическая схема.

Пример логического проектирования функциональной схемы функции в базисе «ИЛИ-НЕ» (используются только двухвходовые логические элементы «ИЛИ-НЕ»):

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



Если число аргументов функций велико, то описанный способ синтеза комбинаторных схем становится громоздким и трудоемким.

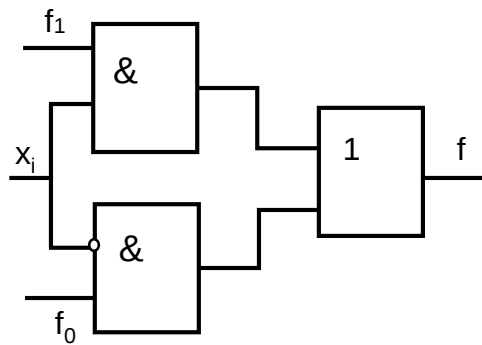
3.9 Каскадный метод минимизации функционального представления

Основанием для метода является **теорема Шеннона**, исходя из которой производная функции может быть представлена в следующем виде:

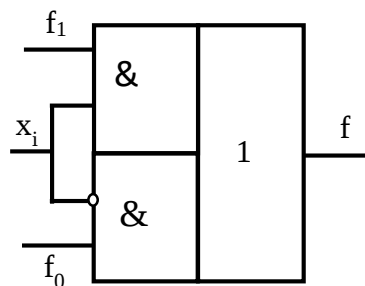
$$\begin{aligned}
 f(x_1, \dots, x_n) &= \bar{x}_i f(x_1, \dots, x_{i-1}, \emptyset, x_{i+1}, \dots, x_n) + x_i f(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n) \\
 &= \bar{x}_i f(\emptyset) + x_i f(1)
 \end{aligned}$$

$$f(x_1, \dots, x_n) = \bar{x}_i f(\emptyset) + x_i f(1)$$

Эту функцию можно реализовать следующей схемой:



Это устройство – каскадный элемент, который на схемах обозначают в следующем виде:



С помощью данного каскада получено представление от n аргументов, через остаточные функции f_0 и f_1 для функций от $n-1$ аргумент, с помощью исключения x_i переменной.

На втором шаге каскада реализуется исключение следующей переменной. И далее поступаая также получаютс остаточные функции.

Это – каскадный метод, при котором каждый каскад соответствует исключению некоторой переменной.

Для одной и той же функции можно получить несколько реализаций. Число реализаций равно:

$$N (N - 1)(N - 2) - \dots = n!$$

Отсюда следует, что найти самую эффективную реализацию простым перебором весьма затруднительно.

Поэтому для метода должна быть найдена эффективная стратегия. Это позволяет сделать аппарат производных от булевых функций.

Производная от функции n -аргументов:

$$\frac{\partial f(x_1, \dots, x_n)}{\partial x_i} = f(x_1, \dots, x_{i-1}, \emptyset, x_{i+1}, \dots, x_n) \oplus f(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n)$$

Пример:

Найти производную от функции двух переменных

$$f(x_1, x_2) = x_1 + x_2$$
$$\frac{\partial f(x_1, x_2)}{\partial x_1} = x_2 \oplus (1 + x_2) = \bar{x}_2$$

Повторные производные определяются следующим образом:

$$\frac{\partial^2 f}{\partial x_i \partial x_j} = \frac{\partial}{\partial x_i} \left(\frac{\partial f}{\partial x_j} \right)$$

и т. д.

Производная от булевой функции определяет условия, при которых функция зависит от x_i

$$f(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) \neq f(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n)$$

т.е. при изменении x_i меняется значение функции. (функция зависит от производной)

Если производная равна 0, то функция от x_i не зависит, т.е. при изменении x_i значение функции не меняется.

Рассмотрим **пример:**

$$f(x_1 x_2 x_3 x_4) = x_1 x_2 x_3 + \bar{x}_1 x_4 + x_2 x_3 + x_1 \bar{x}_3 x_4$$

Определить при каких условиях функция зависит от переменной x_3 . Найдем соответствующую производную:

$$\frac{\partial f}{\partial x_3} = (\bar{x}_1 x_4 + x_1 x_4) \oplus (x_1 x_2 + \bar{x}_1 x_4 + x_2) = x_4 \oplus (x_2 + \bar{x}_1 x_4)$$

Ответом на поставленную задачу будут значения x_1, x_2, x_4 при которых

$$\frac{\partial f}{\partial x_3} = 1$$

Составим соответствующую таблицу:

x_1	x_2	x_4	$\frac{\partial f}{\partial x_3}$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

x_1	x_2	x_4	$\frac{\partial f}{\partial x_3}$
0	1	0	1
1	0	1	1
1	1	0	1

На этих наборах функция зависит от переменной x_3 .

Введем определение. **Весом производной булевой функции** называется количество возможных наборов, при которых производная равна 1. Исходя из предыдущего примера сложность производной по x_3 ($\frac{\partial f}{\partial x_3}$) равна 3.

Чем больше состояний имеет функция, при которых она зависит от x_i , тем сильнее она зависит от этой переменной.

Пример: Определить, от какой переменной функция зависит сильнее.

$$f(x_1, x_2, x_3, x_4) = x_1 x_2 x_3 + \bar{x}_1 x_4 + x_2 x_3 + x_1 \bar{x}_3 x_4$$

$$W\left(\frac{\partial f}{\partial x_3}\right) = 3$$

$$\frac{\partial f}{\partial x_1} = (x_4 + x_2 x_3) \oplus (x_2 x_3 + \bar{x}_3 x_4)$$

x_1	x_2	x_4	$\frac{\partial f}{\partial x_1}$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

$$W\left(\frac{\partial f}{\partial x_1}\right) = 1$$

$$\frac{\partial f}{\partial x_2} = (\bar{x}_1 x_4 + x_1 \bar{x}_3 x_4) \oplus (x_1 x_3 + \bar{x}_1 x_4 + x_3 + x_1 \bar{x}_3 x_4) = (x_4 \bar{x}_1 + x_4 \bar{x}_3) \oplus (x_3 + x_4)$$

x_1	x_2	x_4	$\frac{\partial f}{\partial x_2}$
0	0	0	0
0	0	1	0

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

$$W\left(\frac{\partial f}{\partial x_2}\right) = 3$$

$$\frac{\partial f}{\partial x_4} = (x_1 x_2 x_3 + x_2 x_3) \oplus (x_1 x_2 x_3 + \bar{x}_1 + x_2 x_3 + x_1 \bar{x}_3) = x_2 x_3 \oplus (x_2 + \bar{x}_1 + \bar{x}_3)$$

x_1	x_2	x_4	$\frac{\partial f}{\partial x_4}$
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

$$W\left(\frac{\partial f}{\partial x_4}\right) = 5$$

Таким образом, сильнее всего функция зависит от переменной x_4 . При этом видно, что, чем сильнее зависимость функции, тем меньше сложность остаточной функции при ее исключении.

Для функции заданной таблично удобно использовать **табличный метод определения веса производной**.

Рассмотрим на примере.

Функция от четырех переменных задана таблицей истинности:

x_1	x_2	x_3	x_4	f
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1

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

Для заданной функции определить веса производных по x_1, x_2, x_3, x_4 .

Столбцы 1 и 0 заполняются следующим образом.

Вместо исключаемой переменной в каждый набор подставляются значения 1 или 0 и выписывают соответствующие значения функции из исходной таблицы.

Столбец d получается сложением по mod 2 соответствующих значений столбцов 1 и 0.

x_1 :

x_2	x_3	x_4	1	0	d
0	0	0	1	0	1
0	0	1	1	1	1
0	1	0	1	1	0
0	1	1	0	1	1
1	0	0	1	0	1
1	0	1	1	1	0
1	1	0	0	0	0
1	1	1	0	1	1

$$W_1 = 5$$

x_2 :

x_1	x_3	x_4	1	0	d
0	0	0	0	0	0
0	0	1	1	1	0
0	1	0	0	1	1
0	1	1	1	1	0
1	0	0	1	1	0
1	0	1	1	0	1
1	1	0	0	1	1
1	1	1	0	0	0

$$W_2 = 3$$

x_3 :

x_1	x_2	x_4	1	0	d
0	0	0	1	0	1
0	0	1	1	1	0
0	1	0	0	0	0
0	1	1	1	1	0
1	0	0	1	1	0
1	0	1	0	0	0
1	1	0	0	1	1
1	1	1	0	1	1

$$W_3 = 3$$

x_4 :

x_1	x_2	x_3	1	0	d
0	0	0	1	0	1
0	0	1	1	1	0
0	1	0	1	0	1
0	1	1	1	0	1
1	0	0	0	1	1
1	0	1	0	1	1
1	1	0	1	1	0
1	1	1	0	0	0

$$W_4 = 5$$

Анализ производных показал, что для данного примера первыми исключаемыми переменными будут x_1 и x_4 .

Рассмотрим еще один пример.

Функция 4-х переменных задана таблицей истинности. Для минимизации функционального представления каскадным методом найти первую исключаемую переменную одним из следующих способов:

- 1) с использованием веса производной;
- 2) табличным;
- 3) на картах Карно.

x_1	x_2	x_3	x_4	$f(\tilde{x})$
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1

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

1) метод исключения переменной с использованием веса производной

Для применения данного метода построим СДНФ заданной функции:

$$f(x) = \bar{x}_1\bar{x}_2\bar{x}_3x_4 \vee \bar{x}_1\bar{x}_2x_3x_4 \vee \bar{x}_1x_2\bar{x}_3\bar{x}_4 \vee \bar{x}_1x_2x_3\bar{x}_4 \vee \bar{x}_1x_2x_3x_4 \\ \vee x_1\bar{x}_2\bar{x}_3x_4 \vee x_1x_2\bar{x}_3x_4 \vee x_1x_2x_3x_4$$

Весом производной функции называется количество возможных наборов, при которых производная равна 1.

Чем больше состояний имеет функция, при которых она зависит от x_i , тем сильнее она зависит от этой переменной.

При этом, чем сильнее зависимость функции, тем меньше сложность остаточной функции при ее исключении.

Поэтому, выясним от какой переменной функция зависит сильнее.

Для чего возьмем производные по каждой из четырех переменных.

$$\frac{\partial f}{\partial x_1} = (\bar{x}_2\bar{x}_3x_4 \vee \bar{x}_2x_3x_4 \vee x_2\bar{x}_3\bar{x}_4 \vee x_2x_3\bar{x}_4 \vee x_2x_3x_4) \\ \oplus (\bar{x}_2\bar{x}_3x_4 \vee x_2\bar{x}_3x_4 \vee x_2x_3x_4) \\ = (\bar{x}_2x_4 \vee x_3x_4 \vee x_2\bar{x}_4 \vee x_2x_3) \oplus (\bar{x}_3x_4 \vee x_2x_4)$$

Составим соответствующую таблицу:

x_2	x_3	x_4	$\frac{\partial f}{\partial x_1}$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1

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

$$W_{\frac{\partial f}{\partial x_1}} = 4.$$

$$\begin{aligned} \frac{\partial f}{\partial x_2} &= (\bar{x}_1 \bar{x}_3 x_4 \vee \bar{x}_1 x_3 x_4 \vee x_1 \bar{x}_3 x_4) \\ &\oplus (\bar{x}_1 \bar{x}_3 \bar{x}_4 \vee \bar{x}_1 x_3 \bar{x}_4 \vee \bar{x}_1 x_3 x_4 \vee x_1 \bar{x}_3 x_4 \vee x_1 x_3 x_4) \\ &= (\bar{x}_1 x_4 \vee \bar{x}_3 x_4) \oplus (\bar{x}_1 \bar{x}_4 \vee \bar{x}_1 x_3 \vee x_3 x_4 \vee x_1 x_4) \end{aligned}$$

Составим соответствующую таблицу:

x_1	x_3	x_4	$\frac{\partial f}{\partial x_2}$
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

$$W_{\frac{\partial f}{\partial x_2}} = 4.$$

$$\begin{aligned} \frac{\partial f}{\partial x_3} &= (\bar{x}_1 \bar{x}_2 x_4 \vee \bar{x}_1 x_2 \bar{x}_4 \vee x_1 \bar{x}_2 x_4 \vee x_1 x_2 x_4) \\ &\oplus (\bar{x}_1 \bar{x}_2 x_4 \vee \bar{x}_1 x_2 \bar{x}_4 \vee \bar{x}_1 x_2 x_4 \vee x_1 x_2 x_4) \\ &= (\bar{x}_1 \bar{x}_2 x_4 \vee \bar{x}_1 x_2 \bar{x}_4 \vee x_1 x_4) \oplus (\bar{x}_1 x_4 \vee \bar{x}_1 x_2 \vee x_2 x_4) \end{aligned}$$

Составим соответствующую таблицу:

x_1	x_2	x_4	$\frac{\partial f}{\partial x_3}$
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	0
1	1	1	0

$$W_{\frac{\partial f}{\partial x_3}} = 2.$$

$$\begin{aligned} \frac{\partial f}{\partial x_4} &= (\bar{x}_1 x_2 \bar{x}_3 \vee \bar{x}_1 x_2 x_3) \\ &\oplus (\bar{x}_1 \bar{x}_2 \bar{x}_3 \vee \bar{x}_1 \bar{x}_2 x_3 \vee \bar{x}_1 x_2 \bar{x}_3 \vee x_1 \bar{x}_2 \bar{x}_3 \vee x_1 x_2 \bar{x}_3 \vee x_1 x_2 x_3) \\ &= (\bar{x}_1 x_2) \oplus (\bar{x}_1 \bar{x}_2 \vee \bar{x}_2 \bar{x}_3 \vee \bar{x}_1 x_3 \vee x_2 x_3 \vee x_1 \bar{x}_3 \vee x_1 x_2) \end{aligned}$$

Составим соответствующую таблицу:

x_1	x_2	x_3	$\frac{\partial f}{\partial x_4}$
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

$$W_{\frac{\partial f}{\partial x_4}} = 6.$$

Оказалось, что сильнее всего функция зависит от переменной x_4 , поэтому ее следует учитывать в качестве первой исключаемой переменной.

2) *табличный метод исключения переменной*

Для функции заданной таблично удобно использовать табличный метод определения веса производной.

Столбцы 1 и 0 заполняются следующим образом. Вместо исключаемой переменной в каждый набор подставляются значения 1 или 0 и выписывают соответствующие столбцы значений функции из исходной таблицы. Столбец d получается сложением по mod 2 соответствующих значений столбцов 1 и 0.

x_1 :

x_2	x_3	x_4	1	0	d
0	0	0	0	0	0
0	0	1	1	1	0
0	1	0	0	0	0
0	1	1	0	1	1
1	0	0	0	1	1
1	0	1	1	0	1
1	1	0	0	1	1
1	1	1	1	1	0

$$W_{x_1} = 4.$$

x_2 :

x_1	x_3	x_4	1	0	d
0	0	0	1	0	1
0	0	1	0	1	1
0	1	0	1	0	1
0	1	1	1	1	0
1	0	0	0	0	0
1	0	1	1	1	0
1	1	0	0	0	0
1	1	1	1	0	1

$$W_{x_2} = 4.$$

x_3 :

x_1	x_2	x_4	1	0	d
0	0	0	0	0	0
0	0	1	1	1	0
0	1	0	1	1	0
0	1	1	1	0	1
1	0	0	0	0	0
1	0	1	0	1	1
1	1	0	0	0	0
1	1	1	1	1	0

$$W_{x_3} = 2.$$

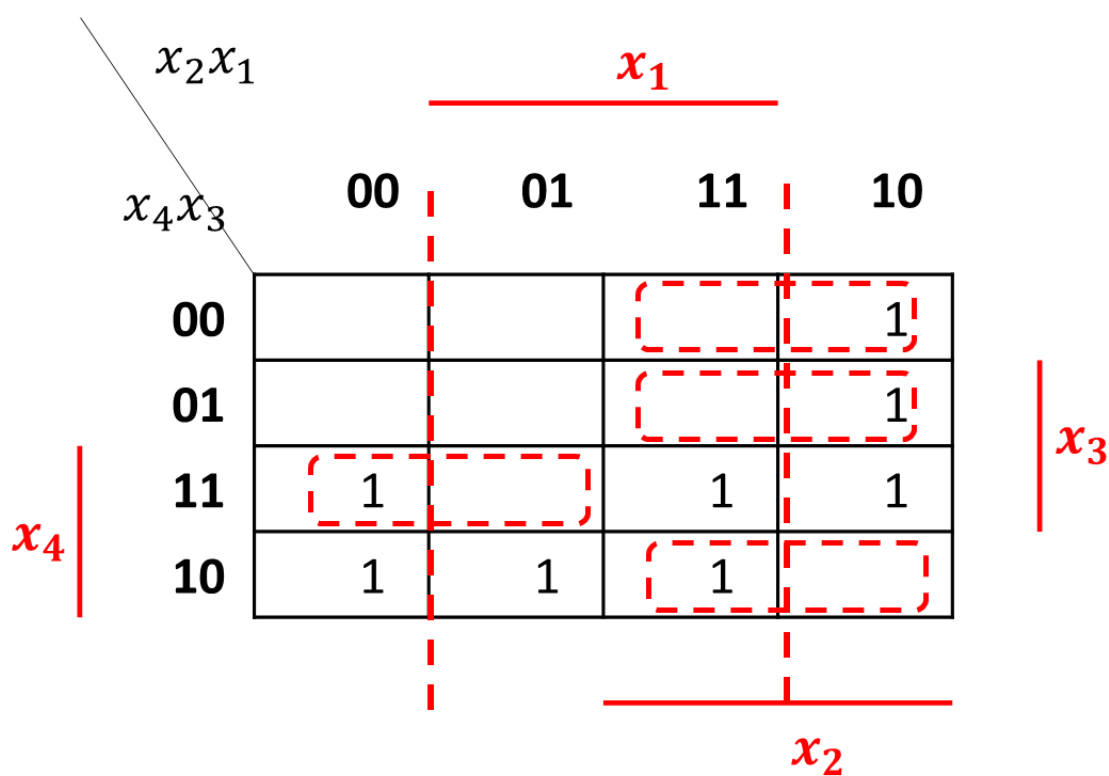
x_4 :

x_1	x_2	x_3	1	0	d
0	0	0	1	0	1
0	0	1	1	0	1
0	1	0	0	1	1
0	1	1	1	1	0
1	0	0	1	0	1
1	0	1	0	0	0
1	1	0	1	0	1
1	1	1	1	0	1

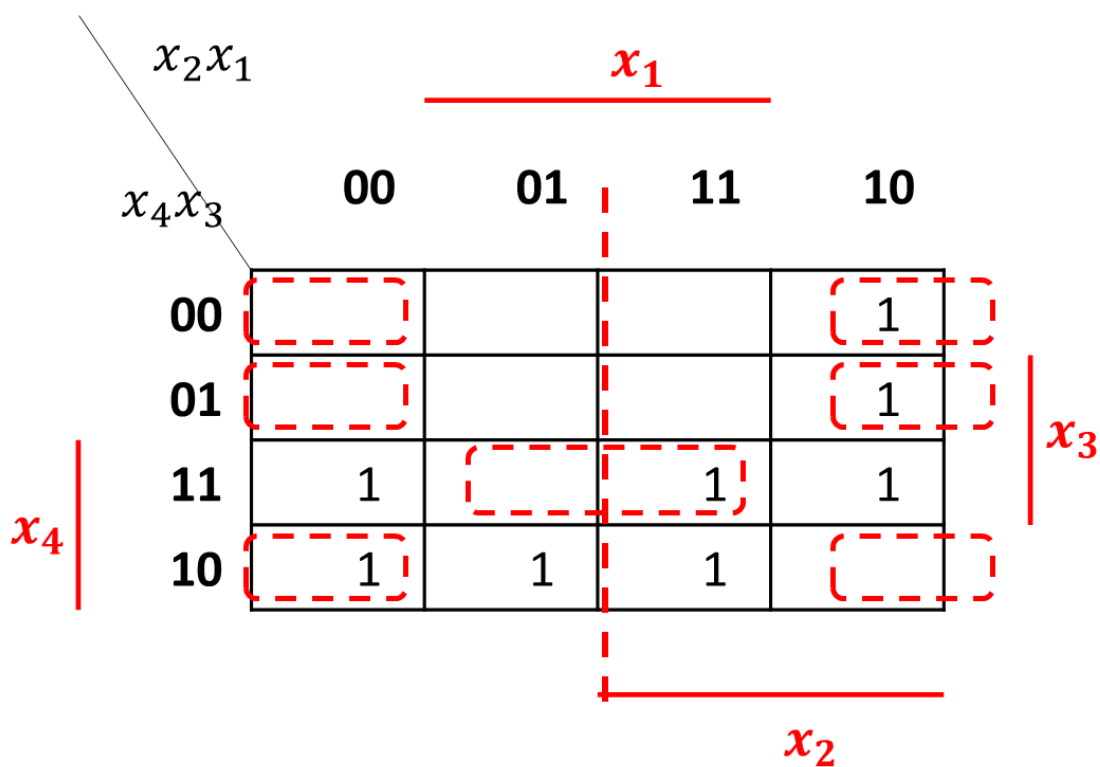
$$W_{x_4} = 6.$$

Аналогично оказалось, что сильнее всего функция зависит от переменной x_4 , поэтому ее следует учитывать в качестве первой исключаемой переменной.

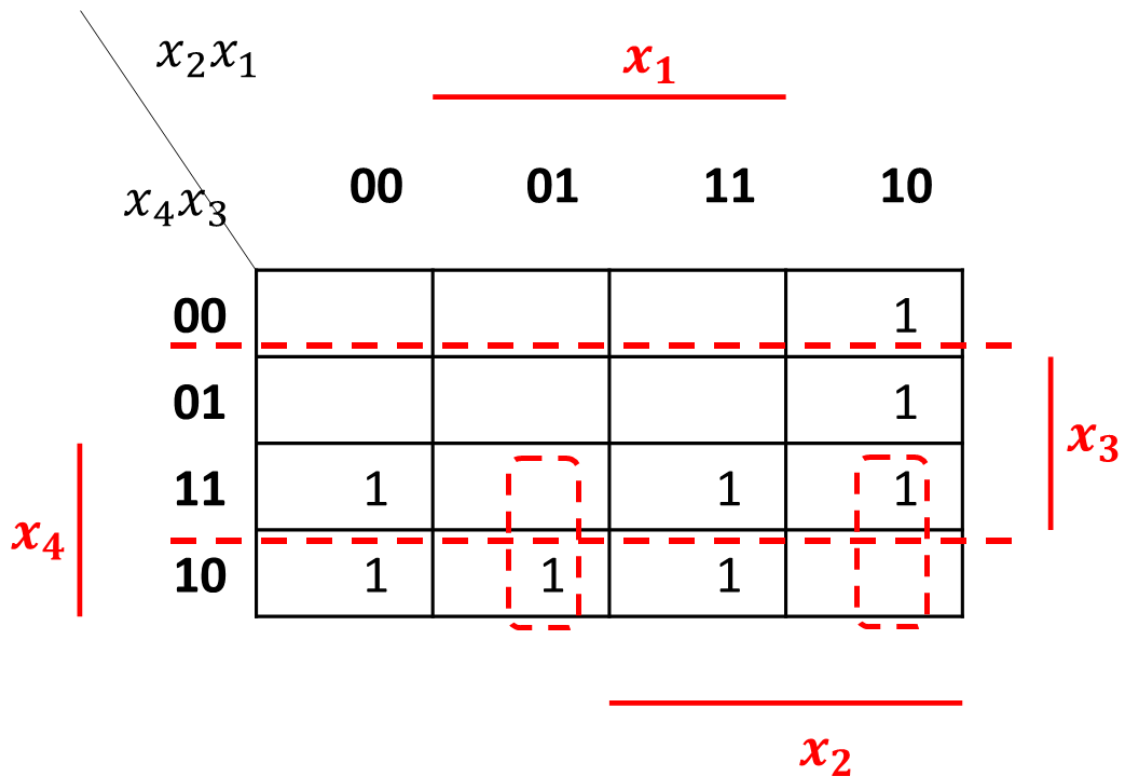
3) *метод исключения переменной на картах Карно*



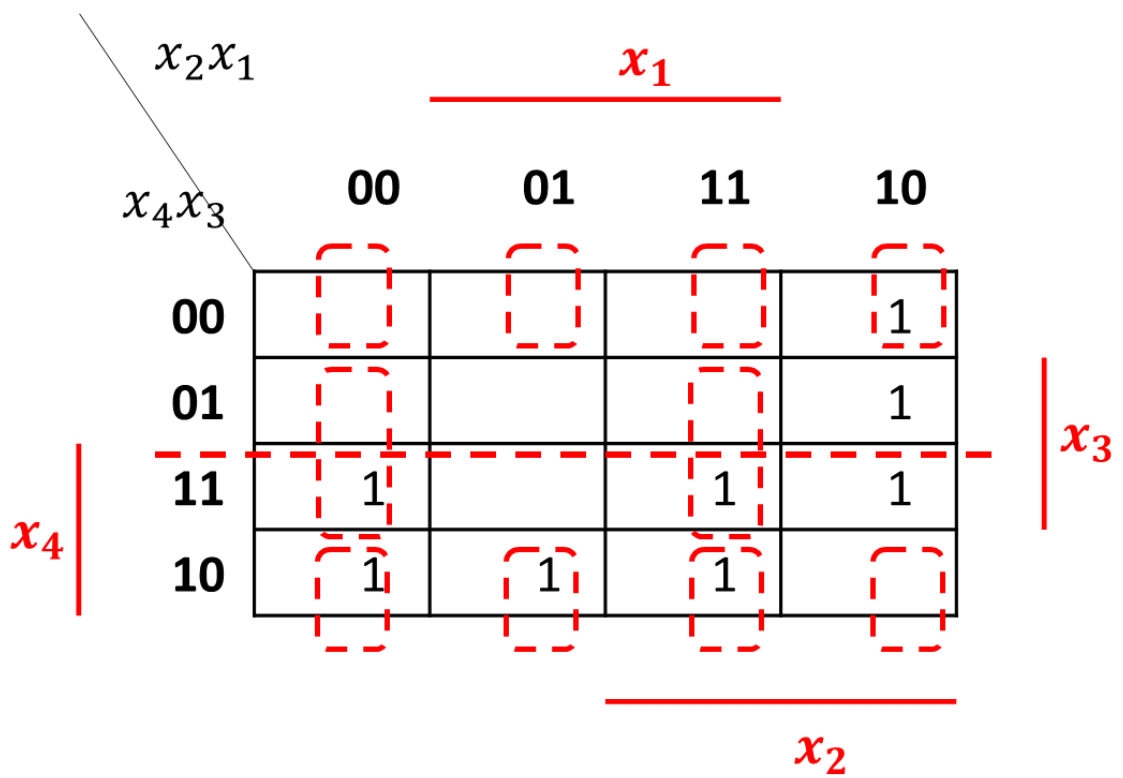
$$W_{x_1} = 4.$$



$$W_{x_2} = 4.$$



$$W_{x_3} = 2.$$



$$W_{x_4} = 6.$$

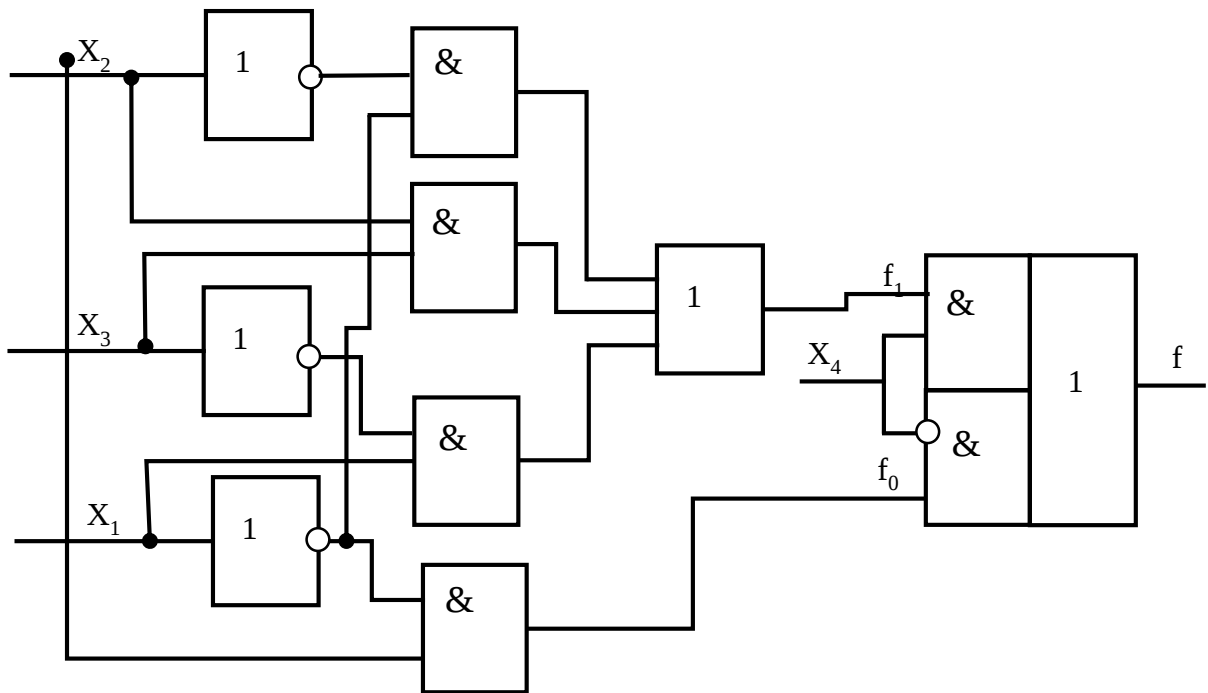
Аналогично оказалось, что сильнее всего функция зависит от переменной x_4 , поэтому ее следует учитывать в качестве первой исключаемой переменной.

Далее строится логическая схема с использованием каскадного элемента, где x_i – исключаемая переменная:

$$\frac{\partial f}{\partial x_4} = (\bar{x}_1 x_2 \bar{x}_3 \vee \bar{x}_1 x_2 x_3) \oplus (\bar{x}_1 \bar{x}_2 \bar{x}_3 \vee \bar{x}_1 \bar{x}_2 x_3 \vee \bar{x}_1 x_2 x_3 \vee x_1 \bar{x}_2 \bar{x}_3 \vee x_1 x_2 \bar{x}_3 \vee x_1 x_2 x_3)$$

$$f_0 = (\bar{x}_1 x_2 \bar{x}_3 \vee \bar{x}_1 x_2 x_3) = \bar{x}_1 x_2$$

$$\begin{aligned} f_1 &= (\bar{x}_1 \bar{x}_2 \bar{x}_3 \vee \bar{x}_1 \bar{x}_2 x_3 \vee \bar{x}_1 x_2 x_3 \vee x_1 \bar{x}_2 \bar{x}_3 \vee x_1 x_2 \bar{x}_3 \vee x_1 x_2 x_3) \\ &= (\bar{x}_1 \bar{x}_2 \vee \bar{x}_2 \bar{x}_3 \vee \bar{x}_1 x_3 \vee x_2 x_3 \vee x_1 \bar{x}_3 \vee x_1 x_2) \\ &= (\bar{x}_1 \bar{x}_2 \vee x_2 x_3 \vee x_1 \bar{x}_3) \end{aligned}$$



4 ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНЫХ РАБОТ

Лабораторная работа № 1 АРИФМЕТИЧЕСКИЕ ОСНОВЫ ЦВМ

1. Задано смешанное число в десятичной системе счисления (см. таблицу 5.1). Выполнить его перевод из одной системы счисления в другую в соответствии с указанной схемой.

2. Заданы целые числа по абсолютной величине $|A|$ и $|B|$. (Числа выбрать из таблицы 5.1, рассматривая целую часть как $|A|$, а дробную- как $|B|$. Например, вариант 20 задает числа $|A|=181$ и $|B|=61$

Представить числа в двоичной системе счисления.

Выполнить сложение знаковых чисел, используя указанный код:

- а) $+A+B$, обратный код,
- б) $-A+B$, обратный код,
- в) $A+(-B)$, дополнительный код,
- г) $-A+(-B)$ дополнительный код.

Результаты представить в прямом коде.

Лабораторная работа № 2 ЛОГИЧЕСКИЕ ФУНКЦИИ ДВУХ ПЕРЕМЕННЫХ. РЕАЛИЗАЦИЯ ФУНКЦИЙ ФОРМУЛАМИ

1. Для заданной по варианту логической функции $f(x_1, x_2, x_3, x_4)$ (табл.5.2) определить приоритет операций (указать над заданной формулой числами)

2. Построить таблицу истинности, для каждой используемой в формуле операции отдельно. Таблицу формировать пошагово, соблюдая при этом приоритет операций.

№	x_1	x_2	x_3	x_4	1	2	3	...	$f(x_1, x_2, x_3, x_4)$
0									
1									
2^{n-1}									

3. Для упрощения вычислений необходимо использовать инструментальные возможности табличного редактора Microsoft Excel. Результат вычисления каждой функции должен быть представлен в отдельном столбце таблицы, посредством ввода в ячейки соответствующих формул. Используемые формулы описать в отчете.

Лабораторная работа № 3
СДНФ и СКНФ БУЛЕВОЙ ФУНКЦИИ.
АНАЛИТИЧЕСКАЯ МИНИМИЗАЦИЯ
ФУНКЦИОНАЛЬНОГО ПРЕДСТАВЛЕНИЯ

1. Для заданной в лабораторной работе №2 логической функции $f(x_1, x_2, x_3, x_4)$ построить СДНФ и СКНФ функции.
2. Посчитать сложность S полученной СДНФ функции.
3. С помощью операций склеивания, поглощения, неполного склеивания и обобщенного склеивания упростить булеву функцию в классе ДНФ.
4. Посчитать сложность S полученной ДНФ функции.
5. Построить таблицу истинности, для полученной ДНФ функции. Сравнить полученную таблицу с исходной.

Лабораторная работа № 4
МЕТОД ЗАКРЕВСКОГО МИНИМИЗАЦИИ
ФУНКЦИОНАЛЬНОГО ПРЕДСТАВЛЕНИЯ

1. Сформировать таблицу истинности функции в соответствии с вариантом задания. Для построения таблицы истинности номер варианта (в десятичной системе счисления) преобразуется в двоичный пятизначный код $N=a_1a_2a_3a_4a_5$.
Например, вариант номер 7 преобразуется в $N=00111$ ($a_1=0, a_2=0, a_3=1, a_4=1, a_5=1$).
2. Булева функция пяти переменных задается таблицей истинности, путем подстановки в таблицу 5.3 значений a_1, a_2, a_3, a_4, a_5 , соответствующих двоичному представлению номера варианта.
3. На основе таблицы истинности построить: нечетные варианты – СДНФ, четные варианты – СКНФ функции.
4. Минимизировать булеву функцию на картах Карно посредством упрощения ДНФ (нечетные варианты) или КНФ (четные варианты) функции методом Закревского.
5. Посчитать сложность S полученного представления функции.

Лабораторная работа № 5

ПРИВЕДЕНИЕ БУЛЕВОЙ ФУНКЦИЙ К ЗАДАННОМУ БАЗИСУ. СИНТЕЗ КОМБИНАЦИОННЫХ СХЕМ

1. Полученную в лабораторной работе № 3 сокращенную ДНФ функции преобразовать к заданному базису:
 - И-НЕ (нечетные варианты);
 - ИЛИ-НЕ (четные варианты).
2. Построить схемную реализацию преобразованной функции в заданном базисе, используя при этом только двухвходовые элементы с инверсным выходом. Логическую схему построить одним из перечисленных способов:
 - с помощью онлайн-конструктора по построению логических схем: <https://www.semestr.online/graph/logic-gate.php>
 - с применением специального программного приложения LogModel.exe.
3. Решить задачу интерпретации логической схемы на ЭВМ (построить таблицу истинности) с помощью выбранного в п.2 инструмента.

Лабораторная работа № 6

МЕТОД КВАЙНА-МАК-КЛАСКИ МИНИМИЗАЦИИ ФУНКЦИОНАЛЬНОГО ПРЕДСТАВЛЕНИЯ

1. Для сформированной в лабораторной работе №4 СДНФ (нечетные варианты) или СКНФ (четные варианты) функции найти тупиковые и минимальные формы методом Квайна-Мак-Класки в классе ДНФ (нечетные варианты) или КНФ (четные варианты).
2. Посчитать сложность S полученных представлений функции.
3. Проанализировать полученные минимальные формы с результатами из л/р №4.

Лабораторная работа № 7
КАСКАДНЫЙ МЕТОД МИНИМИЗАЦИИ
ФУНКЦИОНАЛЬНОГО ПРЕДСТАВЛЕНИЯ

1. Для реализации каскадного метода по таблице истинности, полученной в лабораторной работе №2 найти первую исключаемую переменную заданным по варианту способом (табл. 5.4).
2. Построить логическую схему заданной функции (с каскадным элементом) одним из перечисленных способов:
 - с помощью онлайн-конструктора по построению логических схем:
<https://www.semestr.online/graph/logic-gate.php>
 - с применением специального программного приложения LogModel.exe.
3. Решить задачу интерпретации логической схемы на ЭВМ (построить таблицу истинности) с помощью выбранного в п.2 инструмента.

5 ВАРИАНТЫ ЗАДАНИЙ

Таблица 5.1

Варианты заданий для л/р №1.

Номер варианта	Число	Схема перевода
1	129,46	10 — 8 — 2 — 16 — 10
2	143,62	10 — 16 — 2 — 8 — 10
3	174,57	10 — 8 — 2 — 16 — 10
4	157,83	10 — 16 — 2 — 8 — 10
5	182,39	10 — 8 — 2 — 16 — 10
6	138,59	10 — 16 — 2 — 8 — 10
7	141,54	10 — 8 — 2 — 16 — 10
8	132,76	10 — 16 — 2 — 8 — 10
9	177,87	10 — 8 — 2 — 16 — 10
10	167,59	10 — 16 — 2 — 8 — 10
11	193,71	10 — 8 — 2 — 16 — 10
12	184,63	10 — 16 — 2 — 8 — 10
13	193,74	10 — 8 — 2 — 16 — 10
14	186,41	10 — 16 — 2 — 8 — 10
15	136,51	10 — 8 — 2 — 16 — 10
16	148,78	10 — 16 — 2 — 8 — 10
17	196,39	10 — 8 — 2 — 16 — 10
18	155,67	10 — 16 — 2 — 8 — 10
19	163,66	10 — 8 — 2 — 16 — 10
20	181,61	10 — 16 — 2 — 8 — 10
21	129,46	10 — 8 — 2 — 16 — 10
22	143,62	10 — 16 — 2 — 8 — 10
23	174,57	10 — 8 — 2 — 16 — 10
24	157,83	10 — 16 — 2 — 8 — 10
25	182,39	10 — 8 — 2 — 16 — 10
26	138,59	10 — 16 — 2 — 8 — 10
27	141,54	10 — 8 — 2 — 16 — 10
28	132,76	10 — 16 — 2 — 8 — 10
29	177,87	10 — 8 — 2 — 16 — 10
30	167,59	10 — 16 — 2 — 8 — 10
31	193,71	10 — 8 — 2 — 16 — 10

Таблица 5.2

Варианты определения формулы логической функции для л/р №2, 3, 7

№ в-та	Функция $f(x_1, x_2, x_3, x_4)$	№ в-та	Функция $f(x_1, x_2, x_3, x_4)$
1.	$(x_2 \bar{x}_3 \bar{x}_1 x_4) \rightarrow \bar{x}_1 x_4 \oplus (x_1 x_2 \sim x_2 x_4)$	16.	$(\bar{x}_3 x_4 x_1 \bar{x}_2) \rightarrow (\bar{x}_2 x_3) \oplus (\bar{x}_1 x_2 \sim x_1 x_4)$
2.	$((\bar{x}_1 x_2 \downarrow \bar{x}_3 x_4) \oplus (x_2 \vee x_2 \bar{x}_3)) \rightarrow (x_3 \bar{x}_2)$	17.	$((x_3 \oplus x_4) \rightarrow x_1 \bar{x}_3) \downarrow ((x_1 \bar{x}_2) (\bar{x}_3 x_4)) \downarrow x_3 x_4$
3.	$((x_1 \bar{x}_2 \downarrow \bar{x}_3 x_4) \rightarrow (x_2 x_3 \bar{x}_2 x_4)) \oplus (\bar{x}_1 x_4)$	18.	$(x_1 \bar{x}_4 \bar{x}_1 x_2) \rightarrow \bar{x}_1 x_2 \oplus (\bar{x}_1 x_2 \sim x_2 x_4)$
4.	$((x_1 \bar{x}_2 \downarrow \bar{x}_3 x_4) \oplus (\bar{x}_1 x_3 \vee \bar{x}_2 x_4)) \rightarrow (x_1 \bar{x}_4)$	19.	$((\bar{x}_1 x_2 \downarrow \bar{x}_2 x_3) \oplus (x_3 \vee x_3 \bar{x}_4)) \rightarrow (x_1 \bar{x}_2)$
5.	$((\bar{x}_1 x_2 \downarrow \bar{x}_3 x_4) \oplus (x_1 \bar{x}_3 \vee x_2 x_4)) \rightarrow (\bar{x}_3 x_2)$	20.	$((x_1 \bar{x}_2 \downarrow \bar{x}_1 x_4) \rightarrow (x_1 x_3 \bar{x}_1 x_4)) \oplus (\bar{x}_2 x_3)$
6.	$(\bar{x}_3 \bar{x}_4 \oplus x_1 \bar{x}_2) \rightarrow ((\bar{x}_1 x_2 \downarrow x_3 \bar{x}_4) \oplus x_1 \bar{x}_3)$	21.	$(\bar{x}_2 x_3 x_1 \bar{x}_4) \rightarrow (\bar{x}_2 x_3) \oplus (\bar{x}_3 x_4 \sim x_1 x_4)$
7.	$((\bar{x}_1 x_2) \downarrow (\bar{x}_3 x_2)) \oplus (\bar{x}_2 x_4 \bar{x}_1 x_3) \rightarrow (\bar{x}_1 x_2)$	22.	$((x_1 \bar{x}_2 \downarrow \bar{x}_1 x_4) \oplus (\bar{x}_2 x_3 \vee \bar{x}_1 x_4)) \rightarrow (x_2 \bar{x}_3)$
8.	$((x_1 \oplus x_2) \rightarrow x_1 \bar{x}_3) \downarrow ((x_3 \bar{x}_4) (\bar{x}_1 x_2)) \downarrow x_1 x_4$	23.	$((\bar{x}_1 x_2 \downarrow \bar{x}_3 x_4) \oplus (x_1 \bar{x}_3 \vee x_2 x_4)) \rightarrow (\bar{x}_3 x_2)$
9.	$((x_3 \bar{x}_4 \downarrow \bar{x}_1 x_2) \rightarrow (x_2 x_3 \bar{x}_1 x_3)) \oplus (\bar{x}_2 x_4)$	24.	$(\bar{x}_3 \bar{x}_4 \oplus x_1 \bar{x}_2) \rightarrow ((\bar{x}_1 x_2 \downarrow x_3 \bar{x}_4) \oplus x_1 \bar{x}_3)$
10.	$(\bar{x}_3 x_4 x_1 \bar{x}_2) \rightarrow (\bar{x}_2 x_3) \oplus (\bar{x}_1 x_2 \sim x_1 x_4)$	25.	$((\bar{x}_1 x_2) \downarrow (\bar{x}_3 x_2)) \oplus (\bar{x}_2 x_4 \bar{x}_1 x_3) \rightarrow (\bar{x}_1 x_2)$
11.	$((x_3 \oplus x_4) \rightarrow x_1 \bar{x}_3) \downarrow ((x_1 \bar{x}_2) (\bar{x}_3 x_4)) \downarrow x_3 x_4$	26.	$(\bar{x}_3 x_4 x_1 \bar{x}_2) \rightarrow (\bar{x}_2 x_3) \oplus (\bar{x}_1 x_2 \sim x_1 x_4)$
12.	$(x_1 \bar{x}_4 \bar{x}_1 x_2) \rightarrow \bar{x}_1 x_2 \oplus (\bar{x}_1 x_2 \sim x_2 x_4)$	27.	$(\bar{x}_2 x_3 x_1 \bar{x}_4) \rightarrow (\bar{x}_2 x_3) \oplus (\bar{x}_3 x_4 \sim x_1 x_4)$
13.	$((\bar{x}_1 x_2 \downarrow \bar{x}_2 x_3) \oplus (x_3 \vee x_3 \bar{x}_4)) \rightarrow (x_1 \bar{x}_2)$	28.	$((x_1 \bar{x}_2 \downarrow \bar{x}_1 x_4) \oplus (\bar{x}_2 x_3 \vee \bar{x}_1 x_4)) \rightarrow (x_2 \bar{x}_3)$
14.	$((x_1 \bar{x}_2 \downarrow \bar{x}_1 x_4) \rightarrow (x_1 x_3 \bar{x}_1 x_4)) \oplus (\bar{x}_2 x_3)$	29.	$((\bar{x}_1 x_2 \downarrow \bar{x}_3 x_4) \oplus (x_1 \bar{x}_3 \vee x_2 x_4)) \rightarrow (\bar{x}_3 x_2)$
15.	$((x_1 \bar{x}_2 \downarrow \bar{x}_1 x_3) \oplus (\bar{x}_1 x_3 \vee \bar{x}_1 x_4)) \rightarrow (x_2 \bar{x}_4)$	30.	$(\bar{x}_3 \bar{x}_4 \oplus x_1 \bar{x}_2) \rightarrow ((\bar{x}_1 x_2 \downarrow x_3 \bar{x}_4) \oplus x_1 \bar{x}_3)$

Таблица истинности исходной функции для л/р №4, 5, 6

x_1	x_2	x_3	x_4	x_5	$f(\tilde{x})$
0	0	0	0	0	a1
0	0	0	0	1	1
0	0	0	1	0	a1
0	0	0	1	1	1
0	0	1	0	0	a2
0	0	1	0	1	0
0	0	1	1	0	a2
0	0	1	1	1	0
0	1	0	0	0	1
0	1	0	0	1	1
0	1	0	1	0	a3
0	1	0	1	1	a4
0	1	1	0	0	1
0	1	1	0	1	a5
0	1	1	1	0	0
0	1	1	1	1	1
1	0	0	0	0	a3
1	0	0	0	1	0
1	0	0	1	0	a4
1	0	0	1	1	0
1	0	1	0	0	a5
1	0	1	0	1	a3
1	0	1	1	0	1
1	0	1	1	1	1
1	1	0	0	0	0
1	1	0	0	1	a2
1	1	0	1	0	a1
1	1	0	1	1	1
1	1	1	0	0	a2
1	1	1	0	1	1
1	1	1	1	0	a4
1	1	1	1	1	1

Таблица 5.4

Варианты задания для определения исключаемой переменной в каскадном методе минимизации функционального представления схемы для л/р № 7

№ в-та	Метод исключения переменных	№ в-та	Метод исключения переменных
1.	визуальный метод определения веса производной	16.	визуальный метод определения веса производной
2.	табличный метод определения веса производной	17.	табличный метод определения веса производной
3.	метод определения веса производной на картах Карно	18.	метод определения веса производной на картах Карно
4.	визуальный метод определения веса производной	19.	визуальный метод определения веса производной
5.	табличный метод определения веса производной	20.	табличный метод определения веса производной
6.	метод определения веса производной на картах Карно	21.	метод определения веса производной на картах Карно
7.	визуальный метод определения веса производной	22.	визуальный метод определения веса производной
8.	табличный метод определения веса производной	23.	табличный метод определения веса производной
9.	метод определения веса производной на картах Карно	24.	метод определения веса производной на картах Карно
10.	визуальный метод определения веса производной	25.	визуальный метод определения веса производной
11.	табличный метод определения веса производной	26.	табличный метод определения веса производной
12.	метод определения веса производной на картах Карно	27.	метод определения веса производной на картах Карно
13.	визуальный метод определения веса производной	28.	визуальный метод определения веса производной
14.	табличный метод определения веса производной	29.	табличный метод определения веса производной
15.	метод определения веса производной на картах Карно	30.	метод определения веса производной на картах Карно

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Новоселов В.Г. Прикладная математика для системотехников. Дискретная математика в задачах и примерах. / Учебное пособие. / В.Г. Новоселов, А.В. Скатков – Киев: УМК ВО, 1992. – 200 с.
2. Новиков Ф.А. Дискретная математика для программистов/ Ф.А. Новиков. – СПб.: Питер, 2009. – 384 с.
3. Галушкина Ю.И. Конспект лекций по дискретной математике/ Ю.И. Галушкина, А.Н. Марьямов – М.: Айрис–пресс, 2007. – 176 с.
4. Белоусов А.Н. Дискретная математика/А.Н. Белоусов, Д.М. Грачев -М: изд. МВТУ им. Баумана, 2001. - 744с.
5. Иванов Б.Н. Дискретная математика. Алгоритмы и программы/ Б.Н. Иванов -М: Лаборатория базовых знаний, 2001. - 288с.
6. Игошин В.И. Математическая логика и теория алгоритмов. — 2-е изд., стереотип. — М.: Издательский центр «Академия», 2008. — 448 с.
7. Мангушева И.П., Хрусталева П.М. Лекции по дискретной математике. Часть I. Элементы теории множеств и отношений. Функции алгебры логики. Функции k-значной логики. Учебное пособие. -Саратов: Изд-во «Научная книга», 2010. - 68с.
8. Макоха А.Н., Сахнюк П.А., Червяков Н.И. Дискретная математика. Учебное пособие. - М.: ФИЗМАТЛИТ, 2005 г. - 368 с.
9. Москинова Г.И. Дискретная математика. Математика для менеджера в примерах и упражнениях. Учебное пособие, М.: Логос, 2000. - 240 с.
10. Набебин А.А., Тарасиков А.С. Алгебраическая спецификация программных систем М.: ИНЭК, 2012 г. - 84 с.
11. Перязев Н.А. Основы теории булевых функций / Перязев Н.А. Основы теории булевых функций. - М.: Физматлит, 2000. - 109 с.

**ДИСКРЕТНАЯ МАТЕМАТИКА
И КОМПЬЮТЕРНАЯ ЛОГИКА.
БУЛЕВА АЛГЕБРА**

Методические указания

*Составители : **Ченгарь** Ольга Васильевна,
Малицкая Александра Александровна*

В авторской редакции

Изд. № 82/2023. Объем 3 п.л. Усл. печ. л. 2,79. Уч.-изд. л. 2,94.
РИИЦМ ФГАОУ ВО «Севастопольский государственный университет»