ТСИС

(Технические средства информационных систем)

Программное обеспечение информационных систем (1-40 01 73)

Гр. 60325, 60326

Логические основы ЭВМ. Минимизация.

Лекция 3

(По материалам Мухаметова В.Н.)

Ковалевский Вячеслав Викторович

Ковалевский Вячеслав Викторович

4096tb@gmail.com

Тема письма:

БГУИР. … .

Лекция 1. Представление информации. Системы счисления.

Формат с фиксированной запятой

План лекции:

- История развития вычислительной техники.
- Понятие информации.
- Принцип программного управления.
- Двоичная и шестнадцатеричная системы счисления.
- Прямой и дополнительный код.
- Арифметические действия в Формате ФЗ.
- Переполнение.

Экзаменационные вопросы:

- Информационная система. Информация. История развития компьютера.
- Позиционные системы счисления. Перевод чисел из одной системы счисления в другую.
- Арифметика ЭВМ. Представление чисел в форме с фиксированной точкой.
- Сложение в формате с фиксированной точкой. Переполнение.
- Операция вычитания с фиксированной точкой. Дополнительный код числа.

Лекция 2. Формат с плавающей запятой. Стандарт IEEE 754. Погрешности. Обратная польская запись

План лекции:

- Формат чисел с плавающей запятой.
- Стандарт IEEE 754.
- Особенности операций в формате с плавающей запятой.
- Переполнение порядков.
- Точность вычислений.
- Обратная польская запись.

Экзаменационные вопросы:

- Представление чисел в форме с плавающей точкой. Мантисса и характеристика числа.
- Нормализованные и денормализованные числа. Погрешность представления числа.
- Арифметические операции в формате с плавающей точкой.
- Стандарт IEEE 754.
- Формат BCD. Представление текстовой информации. ASCII.

Лекция 3. Логические основы ЭВМ. Минимизация.

План лекции:

- Понятия алгебры логики.
- Аксиомы и законы алгебры логики.
- Логические функции: конъюнкция, дизъюнкция, инверсия и другие функции.
- Преобразование логических выражений.
- Логические элементы.
- Логические (комбинационные) схемы.
- Понятие о минимизации логических выражений.

Экзаменационные вопросы:

- Алгебра логики. Переменные и константы алгебры логики.
- Законы и аксиомы алгебры логики. Логические функции.
- Конъюнкция. Дизъюнкция. Инверсия. Функционально полная система ЛФ. Функции И-НЕ, ИЛИ-НЕ, Исключающее ИЛИ.
- Формы представления ЛФ. Таблица истинности. СДНФ и СКНФ. Переход от одной формы к другой.
- Преобразование логических выражений. Склеивание. Минимизация логических выражений.

Булева алгебра

Джордж Буль (George Boole)
02.11.1815 — 08.12.1864
Известный английский математик и логик. Автор «логических операторов» и «двоичной системы», оперирующие двумя видами сигналов - наличие сигнала (1) или его отсутствие (0).
Сама идея об использования 1 и 0 в качестве основных операторов математической логики была высказана ещё в работах Лейбница, однако, именно Буль сумел довести его идеи до совершенства.

Алгебра логики (Булева алгебра)

Алгебра логики рассматривает высказывания и их взаимосвязь только с точки зрения их истинности либо ложности.

Если x – это высказывание, то в алгебре логики

x = True (x = Истина)

либо

x = False (x = Ложь)

Константы АЛ

Логические функции

Независимые высказывания называют аргументами. Высказывания, истинность либо ложность которых зависит от истинности либо ложности других высказываний, называют логическими функциями, зависящими от своих аргументов:

$$y = f(x),\ y = f(x1,\ x2,\ x3)$$

и т.п.

Для упрощения записей значения «Ложь» и «Истина» обозначают нулем и единицей (***0*** и ***1***).

Логические переменные могут принимать только эти два значения.

Примеры:

x = 0

x1 = 0

x2 = 1

y = 0

Alpha = 1

Примеры:

X = 0

x1 = Ложь

X2 = 1

y = False

Alpha = Истинна

Omega = True

Аксиомы Булевой алгебры

Аксиомы конъюнкции

0*0=0 0*1=0 1*1=1

Аксиомы дизъюнкции

0+0=0 0+1=1 1+1=1

Аксиомы отрицания

If x=0, then $\overline{x}$=1 If x=1, then $\overline{x}$=0

Теоремы Булевой алгебры

Теоремы исключения констант

$x*0=0$ $x*1=x$ $x+1=1$ $x+0=x$

Теоремы идемпотентности (тавтологии, повторения)

$x*x=x$ $x+x=x$

Теорема противоречия

$x*\overline{x}=0$

Теорема "исключённого третьего"

$x+\overline{x}=1$

Законы Булевой алгебры

Ассоциативный (сочетательный) закон

$x1*(x2*x3) = (x1*x2)*x3$ $x1+(x2+x3) = (x1+x2)+x3$

Коммутативный (переместительный) закон

$x1*x2 = x2*x1$ $x1+x2 = x2+x1$

Дистрибутивный (распределительный) закон

$x1*(x2+x3) = x1*x2+x1*x3$ $\underline{x1+(x2*x3) = (x1+x2)*(x1+x3)}$

Закон поглощения (элиминации)

$x1+(x1*x2) = x1$ $x1*(x1+x2) = x1$

Закон склеивания (исключения)

$(x1*x2)+(x1*\overline{x2}) = x1$ $(x1+x2)*(x1+\overline{x2}) = x1$

Правило де Мо́ргана

$$\overline{x1 * x2} = \overline{x1} + \overline{x2}$$

$$\overline{x1 \wedge x2} = \overline{x1} \vee \overline{x2}$$

или

$$\overline{x1 + x2} = \overline{x1} * \overline{x2}$$

$$\overline{x1 \vee x2} = \overline{x1} \wedge \overline{x2}$$

Правило де Мóргана

Отрицание **конъюнкции** есть **дизъюнкция** отрицаний:

$$\overline{x1*x2}=\overline{x1}+\overline{x2} \qquad \overline{x1\wedge x2}=\overline{x1}\vee\overline{x2}$$

или

Отрицание **дизъюнкции** есть **конъюнкция** отрицаний:

$$\overline{x1+x2}=\overline{x1}*\overline{x2} \qquad \overline{x1\vee x2}=\overline{x1}\wedge\overline{x2}$$

Формы представления логических функций

- Таблица истинности
- Аналитическое выражение
- Логическая схема

Таблица истинности

Таблица истинности описывает значения логической функции на всех наборах ее аргументов.

Для функции, зависящей от **n** аргументов, рассматривается $\mathbf{N=2^n}$ значений.

Пример таблицы истинности

x1	x2	x3	y
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

Эта же таблица в более компактном виде с нумерацией наборов:

№	**x1x2x3**	**y**
0	**0 0 0**	**0**
1	**0 0 1**	**0**
2	**0 1 0**	**1**
3	**0 1 1**	**1**
4	**1 0 0**	**0**
5	**1 0 1**	**1**
6	**1 1 0**	**1**
7	**1 1 1**	**0**

Аналитическое выражение

При аналитической записи функция представляется либо в виде логической суммы элементарных логических произведений (дизъюнкции элементарных конъюнкций), либо в виде логического произведения элементарных логических сумм (конъюнкции элементарных дизъюнкций).

Первая форма записи имеет название дизъюнктивной нормальной формы (ДНФ),

Вторая - конъюнктивной нормальной формы (КНФ).

Аналитическое представление логических функций

ДНФ:

x1	x2	y
0	0	**1**
0	1	**1**
1	0	**1**
1	1	0

$$y=\overline{x1x2}+\overline{x1}x2+x1\overline{x2}$$

КНФ:

x1	x2	y
0	0	**0**
0	1	**0**
1	0	1
1	1	**0**

$$y=(\overline{x1}+\overline{x2})(x1+\overline{x2})(x1+x2)$$

СДНФ и СКНФ

СДНФ – совершенная дизъюнктивная нормальная форма представления логической функции. СДНФ – это дизъюнкция конъюнкций.

СКНФ – совершенная конъюнктивная нормальная форма представления логической функции. СКНФ – это конъюнкция дизъюнкций.

СДНФ (совершенная дизъюнктивная нормальная форма)

СДНФ логической функции – это дизъюнкция конституент единицы (минтермов), соответствующих наборам входных переменных, для которых функция равна 1.

В общем случае СДНФ можно представить в форме:

$$\left(x_1, x_2, ..., x_n\right) = \bigcup_{1} x_1^{a_1} \cap x_2^{a_2} \cap ... \cap x_n^{a_n}$$

где a_1, a_2, … , a_n – двоичный набор,

$$x_i^{a_i} = \begin{cases} x_i, & если \quad a = 1 \\ \overline{x_i}, & если \quad a = 0 \end{cases}$$

СКНФ (совершенная конъюнктивная нормальная форма)

СКНФ логической функции – это конъюнкция конституент нуля (макстермов), соответствующих входным наборам, для которых функция равна 0.

В общем случае СКНФ можно представить в форме:

$$(x_1, x_2, \ldots, x_n) = \bigcap_0 x_1^{a_1} \cup x_2^{a_2} \cup \ldots \cup x_n^{a_n}$$

где a_1, a_2, … , a_n – двоичный набор,

$$x_i^{a_i} = \begin{cases} x_i, & если \quad a = 1 \\ \overline{x_i}, & если \quad a = 0 \end{cases}$$

СДНФ и СКНФ

Совершенная – во всех членах присутствуют все аргументы.

Нормальная – «без скобок».

Дизъюнктивная – это дизъюнкция конъюнкций.

или

Конъюнктивная – это конъюнкция дизъюнкций.

СДНФ и СКНФ

Совершенная дизъюнктивная нормальная форма (СДНФ)

Функция представляется суммой групп. Каждая группа состоит из произведения, в которую входят все переменные.

Например:

$$f(x1,x2,x3) = \overline{x}1 \cdot x2 \cdot x3 + x1 \cdot \overline{x}2 \cdot x3 + x1 \cdot x2 \cdot \overline{x}3$$

Совершенная конъюнктивная нормальная форма (СКНФ)

Функция представляется произведением групп. Каждая группа состоит из суммы, в которую входят все переменные.

Например:

$$f(x1,x2,x3) = (\overline{x}1+x2+x3) \cdot (x1+\overline{x}2+x3) \cdot (x1+x2+\overline{x}3)$$

Примеры СДНФ и СКНФ

СДНФ – это дизъюнкция конъюнкций.

$$y = (\overline{x}_1 \wedge x_2 \wedge \overline{x}_3) \vee (\overline{x}_1 \wedge x_2 \wedge x_3) \vee (x_1 \wedge \overline{x}_2 \wedge x_3) \vee (x_1 \wedge x_2 \wedge \overline{x}_3)$$

$$y = \overline{x}_1 x_2 \overline{x}_3 + \overline{x}_1 x_2 x_3 + x_1 \overline{x}_2 x_3 + x_1 x_2 \overline{x}_3$$

СКНФ – это конъюнкция дизъюнкций.

$$y = (x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee x_2 \vee \overline{x}_3) \wedge (\overline{x}_1 \vee x_2 \vee x_3) \wedge (\overline{x}_1 \vee \overline{x}_2 \vee \overline{x}_3)$$

$$y = (x_1 + x_2 + x_3)(x_1 + x_2 + \overline{x}_3)(\overline{x}_1 + x_2 + x_3)(\overline{x}_1 + \overline{x}_2 + \overline{x}_3)$$

СДНФ из таблицы истинности

Чтобы записать СДНФ функции, нужно записать все конституенты единицы (т.е. конъюнкции), причем переменная, принимающая на соответствующем наборе значение «0», записывается с инверсией.

Все полученные конъюнкции соединить знаком дизъюнкции.

СДНФ из таблицы истинности

Таблица истинности:

x1	x2	x3	y
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

СДНФ:

$$y = \overline{x1}\,x2\,\overline{x3} + \overline{x1}\,x2\,x3 + x1\,\overline{x2}\,x3 + x1\,x2\,\overline{x3}$$

Функционально полная система логических функций (ФПС ЛФ)

(Булев или логический базис) это такой набор логических функций,
который позволяет используя только
эти функции, записать любую, сколь угодно сложную, логическую функцию.

Конъюнкция (И)

Конъюнкция истинна тогда и только тогда, когда истинны все ее аргументы

x1	x2	y
0	0	0
0	1	0
1	0	0
1	1	1

$$y = x1 \wedge x2$$

$$y = x1 \& x2$$

$$y = x1 * x2$$

$$y = x1x2$$

Дизъюнкция (ИЛИ)

Дизъюнкция истинна, если истинен хотя бы один из ее аргументов.

x1	x2	y
0	0	0
0	1	1
1	0	1
1	1	1

$$y = x1 \vee x2$$

$$y = x1 + x2$$

Отрицание (Инверсия)

Инверсия принимает значение, противоположное значению ее аргумента

x	y
0	1
1	0

$$y = \overline{x}$$

И-НЕ (Not AND, NAND)

x1	x2	y
0	0	**1**
0	1	**1**
1	0	**1**
1	1	**0**

$$y = \overline{x1 \wedge x2}$$

$$y = \overline{x1 \& x2}$$

$$y = \overline{x1 * x2}$$

$$y = \overline{x1x2}$$

ИЛИ-НЕ (Not OR, NOR)

x1	x2	y
0	0	**1**
0	1	**0**
1	0	**0**
1	1	**0**

$$y = \overline{x1 \vee x2}$$

$$y = \overline{x1 + x2}$$

Исключающее ИЛИ (XOR)

x1	x2	y
0	0	**0**
0	1	**1**
1	0	**1**
1	1	**0**

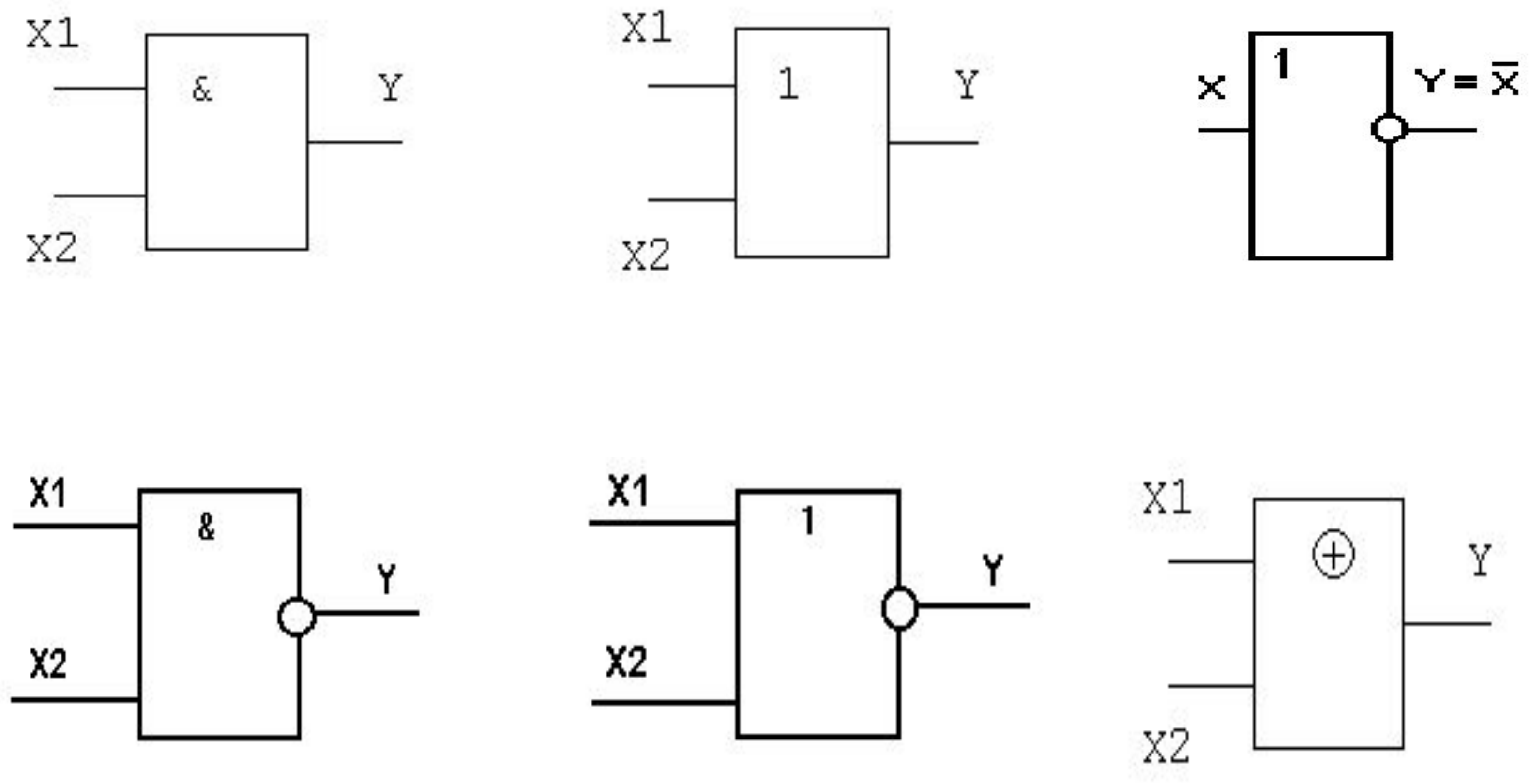
$$y = x1 \oplus x2$$

Логические элементы

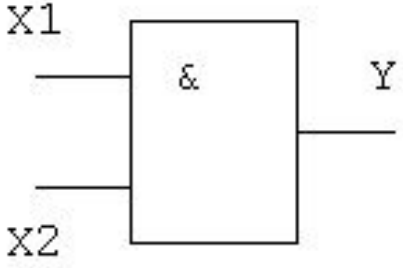
Это устройства, предназначенные для обработки информации в цифровой форме (последовательности сигналов как правило в двоичной логике («1» и «0»)

Физически логические элементы могут быть механическими, электромеханическими (на электромагнитных реле), электронными (на диодах и транзисторах), пневматическими, гидравлическими, оптическими и др.

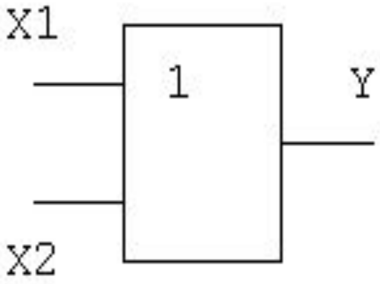
Логические элементы



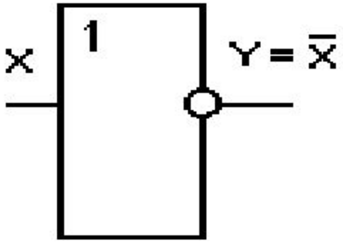
Логические элементы



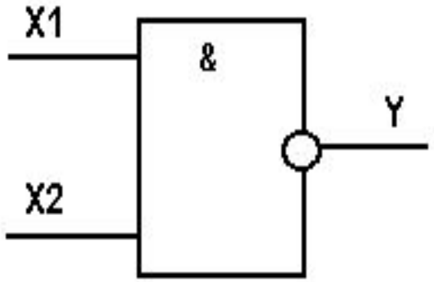
И



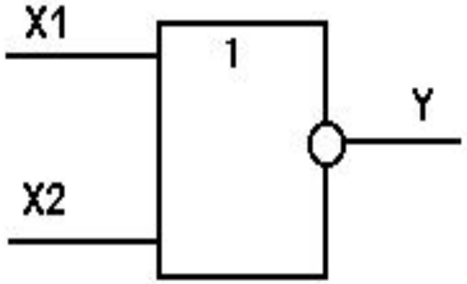
ИЛИ



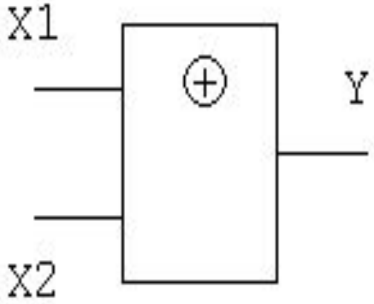
НЕ



И-НЕ

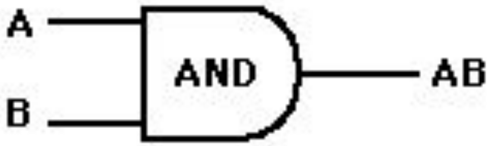


ИЛИ-НЕ

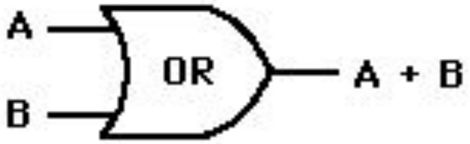


Исключающее ИЛИ

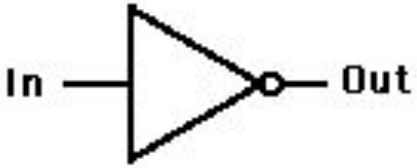
Элементы И, ИЛИ, НЕ в альтернативном обозначении



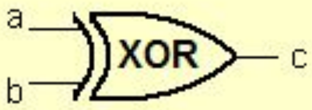
A	B	Out
0	0	0
0	1	0
1	0	0
1	1	1



A	B	Out
0	0	0
0	1	1
1	0	1
1	1	1



In	Out
0	1
1	0



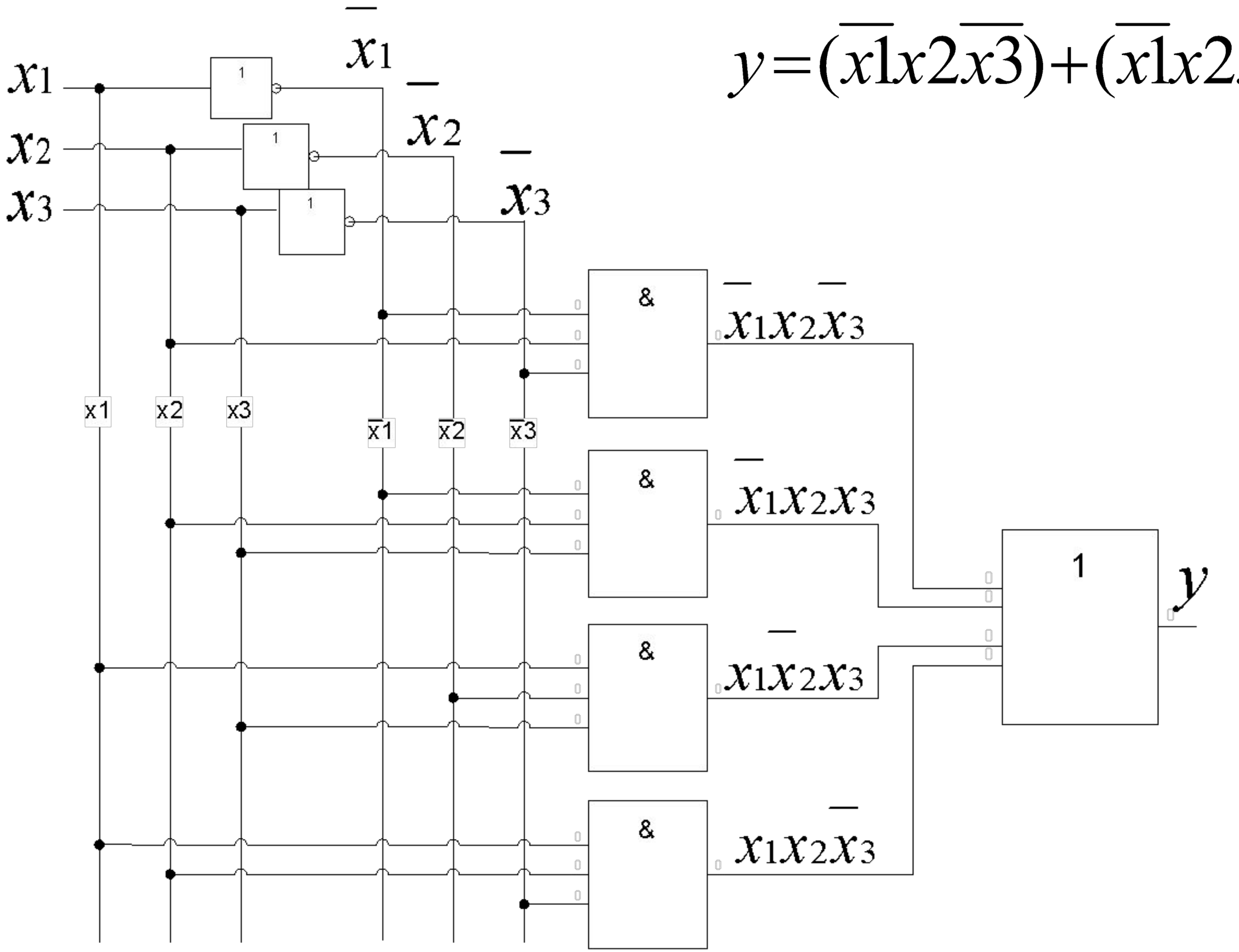
a b	c
0 0	0
0 1	1
1 0	1
1 1	0

Логические (комбинационные) схемы

Логическая схема (ЛС), или схема «без памяти», состоит из логических элементов (ЛЭ), соединенных между собой (выходы одних ЛЭ соединены со входами других ЛЭ), причем обратные связи отсутствуют.

Пример логической схемы

$$y=(\overline{x1}x2\overline{x3})+(\overline{x1}x2x3)+(x1\overline{x2}x3)+(x1x2\overline{x3})$$



Минимизация логических функций

Преобразование СДНФ или СКНФ логической функции к минимальному виду аналитической записи называется процессом минимизации функции.

В процессе минимизации той или иной логической функции, обычно учитывается, в каком базисе эффективнее будет реализовать ее минимальную форму при помощи электронных схем.

Аксиомы алгебры логики

$$x \cdot 0 = 0$$

$$x + 0 = x$$

$$x \cdot 1 = x$$

$$x + 1 = 1$$

$$x \cdot x = x$$

$$x + x = x$$

$$x + \overline{x} = 1$$

$$x \cdot \overline{x} = 0$$

Склеивание

$$a \cdot x + a \cdot \overline{x} = a \cdot (x + \overline{x}) = a \cdot 1 = a$$

Таким образом, $a \cdot x + a \cdot \overline{x} = a$

Такое преобразование называется ***склеиванием***.

Конъюнкции $a \cdot x$ и $a \cdot \overline{x}$ называются

соседними. Они «склеиваются по x»

Примеры склеивания

$$x1 \cdot x2 \cdot \overline{x3} + x1 \cdot x2 \cdot x3 = x1 \cdot x2 \cdot (x3 + \overline{x3}) =$$

$$= x1 \cdot x2 \cdot 1 = x1 \cdot x2$$

$$\overline{x1} \cdot \overline{x2} \cdot \overline{x3} + \overline{x1} \cdot x2 \cdot \overline{x3} = \overline{x1} \cdot \overline{x3}$$

$$\overline{x1} \cdot x2 \cdot \overline{x3} \cdot x4 + x1 \cdot x2 \cdot \overline{x3} \cdot x4 = x2 \cdot \overline{x3} \cdot x4$$

Алгоритмические методы минимизации

Позволяют проводить упрощение функции более просто, быстро и безошибочно. К таким методам относятся:

- метод Квайна
- метод карт Карно
- метод испытания импликант
- метод импликантных матриц
- метод Квайна-Мак-Класки

и др.

Эти методы наиболее пригодны для обычной практики, особенно минимизация логической функции с использованием карт Карно.

Карты Карно

Карты Карно были изобретены в 1952 Эдвардом В. Вейчем и усовершенствованы в 1953 Морисом Карно, физиком из «Bell Labs», и были призваны помочь упростить цифровые электронные схемы.

В карту Карно булевы переменные передаются из таблицы истинности и упорядочиваются с помощью кода Грея, в котором каждое следующее число отличается от предыдущего только одним разрядом.

Метод карт Карно сохраняет наглядность при числе переменных не более шести.

Пример таблицы истинности, СДНФ, СКНФ

Таблица истинности:

№	x1x2x3	y
0	**0 0 0**	**0**
1	**0 0 1**	**0**
2	**0 1 0**	**1**
3	**0 1 1**	**1**
4	**1 0 0**	**0**
5	**1 0 1**	**1**
6	**1 1 0**	**1**
7	**1 1 1**	**0**

СДНФ: $y = \overline{x1}x2\overline{x3} + \overline{x1}x2x3 + x1\overline{x2}x3 + x1x2\overline{x3}$

СКНФ: $y = (\overline{x1} + \overline{x2} + \overline{x3})(\overline{x1} + \overline{x2} + x3)(x1 + \overline{x2} + \overline{x3})(x1 + x2 + x3)$

Карты Карно (диаграммы Вейча)

№	x1x2x3	y
0	**0 0 0**	**0**
1	**0 0 1**	**0**
2	**0 1 0**	**1**
3	**0 1 1**	**1**
4	**1 0 0**	**0**
5	**1 0 1**	**1**
6	**1 1 0**	**1**
7	**1 1 1**	**0**

Перепишем таблицу истинности функции следующим образом:

x_1 \ x_2x_3	00	01	11	10
0	0	0	1	1
1	0	1	0	1

Код Грея

0	0	0
1	0	1
2	1	1
3	1	0

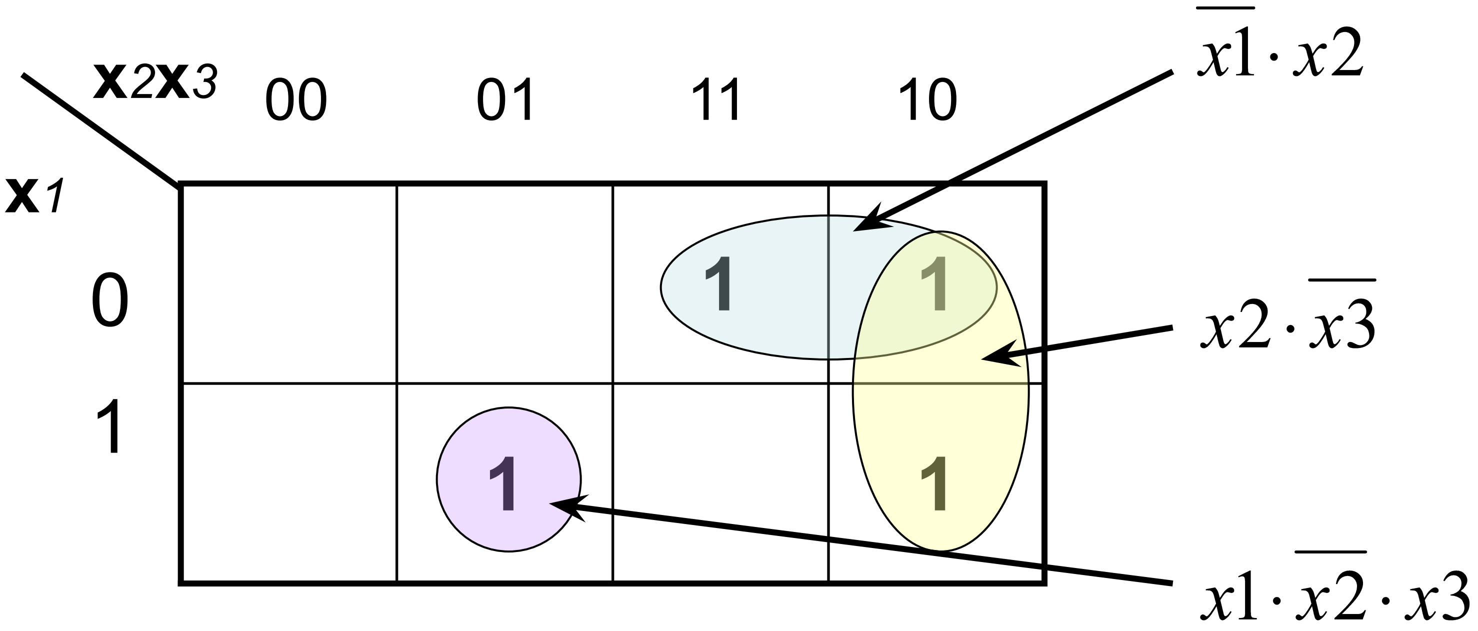
Карты Карно (диаграммы Вейча)

Если убрать нули, то получим:

x_1 \ x_2x_3	00	01	11	10
0			1	1
1		1		1

Карты Карно (диаграммы Вейча)

Единицы, расположенные в соседних клетках, соответствуют соседним конъюнкциям.



$$y = \overline{x1} \cdot x2 + x2 \cdot \overline{x3} + x1 \cdot \overline{x2} \cdot x3$$

Карты Карно (диаграммы Вейча)

X_1	X_2	X_3	X_4	F
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
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

X_3X_4 / X_1X_2	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	0	1	1	0
10	1	0	0	1

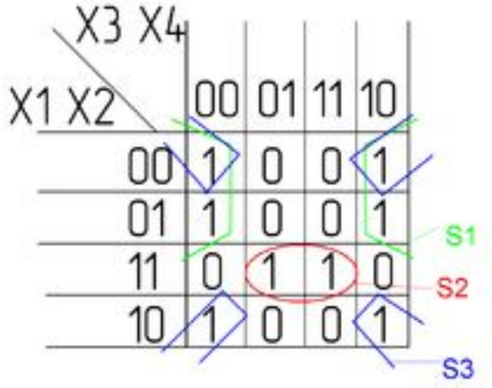
X_3X_4 / X_1X_2	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	0	1	1	0
10	1	0	0	1

X_3X_4 / X_1X_2	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	0	1	1	0
10	1	0	0	1

Карты Карно (диаграммы Вейча)

X_1X_2 \ X_3X_4	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	0	1	1	0
10	1	0	0	1

Выражение в формате ДНФ:



$$f(X_1, X_2, X_3, X_4) = S_1 \vee S_2 \vee S_3 = \overline{X_1}\,\overline{X_4} \vee X_1 X_2 X_4 \vee X_4 X_2$$

X_1X_2 \ X_3X_4	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	0	1	1	0
10	1	0	0	1

Выражение в формате КНФ:

$$f(X_1, X_2, X_3, X_4) = S_1 S_2 S_3 = (X_1 \vee \overline{X_4})(X_2 \vee \overline{X_4})(\overline{X_1} \vee \overline{X_2} \vee X_4)$$

Пример минимизации логической функции

У мальчика Коли есть мама, папа, дедушка и бабушка. Коля пойдёт гулять на улицу, тогда и только тогда, когда ему разрешат хотя бы двое родственников.
Для краткости обозначим родственников Коли через буквы:

- Мама — x1
- Папа — x2
- Дедушка — x3
- Бабушка — x4

Условимся обозначать согласие родственников единицей, несогласие - нулём. Возможность пойти погулять обозначим буквой f:

- Коля идёт гулять — f = 1,
- Коля гулять не идёт — f = 0.

Пример минимизации логической функции

Составим таблицу истинности:

X1	X2	X3	X4	f
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

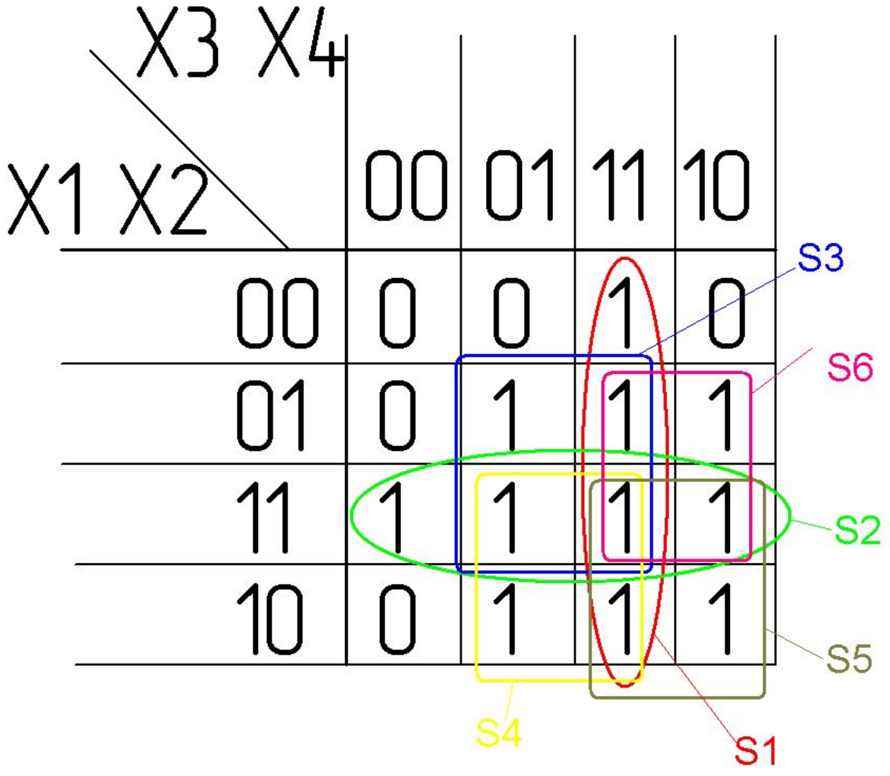
Применяя Код Грея подготовим Карту Карно:

X1 X2 \ X3 X4	00	01	11	10
00				
01				
11				
10				

Заполним Карту Карно согласно таблицы истинности:

X1 X2 \ X3 X4	00	01	11	10
00	0	0	1	0
01	0	1	1	1
11	1	1	1	1
10	0	1	1	1

Пример минимизации логической функции



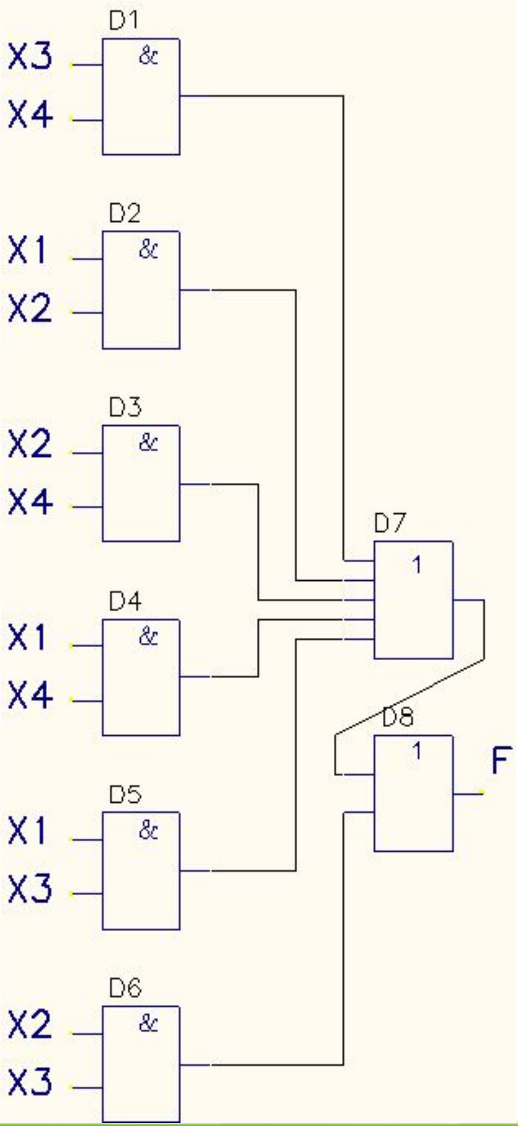
Минимизируем в соответствии с правилами, получим минимальную ДНФ:

$$f(X1, X2, X3, X4) = S1 \vee S2 \vee S3 \vee S4 \vee S5 \vee S6 =$$

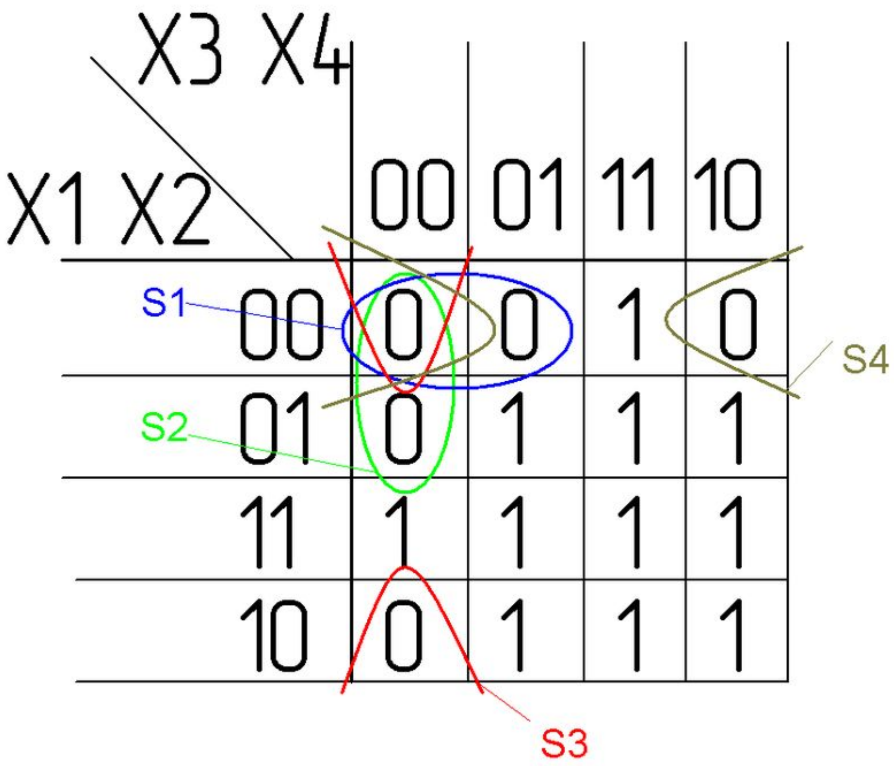
$$= X3X4 \vee X1X2 \vee X2X4 \vee X1X4 \vee X1X3 \vee X2X3$$

Пример минимизации логической функции

По полученной минимальной ДНФ можно построить логическую схему:



Пример минимизации логической функции



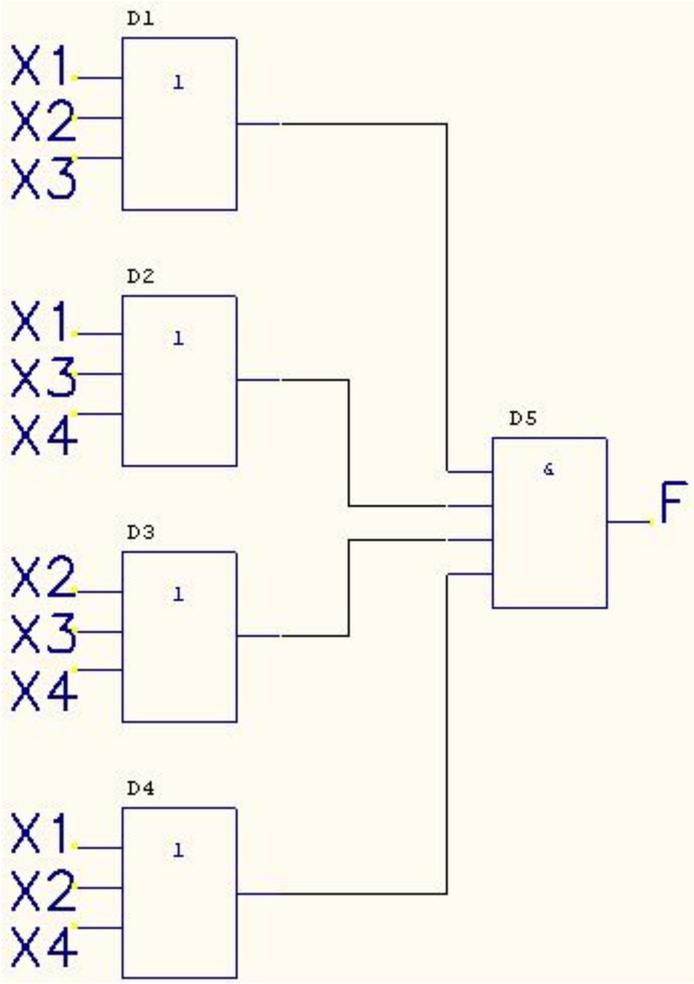
Минимизируем в соответствии с правилами, получим минимальную КНФ:

$$f(X1, X2, X3, X4) = (S1)\ (S2)\ (S3)\ (S4) =$$

$$= (X1 \vee X2 \vee X3)(X1 \vee X3 \vee X4)(X2 \vee X3 \vee X4)(X1 \vee X2 \vee X4)$$

Пример минимизации логической функции

По полученной минимальной КНФ можно построить логическую схему:



ТСИС

(Технические средства информационных систем)
Программное обеспечение информационных систем (1-40 01 73)

- Лекция 3

Логические основы ЭВМ.
Минимизация.

Ковалевский Вячеслав Викторович

4096tb@gmail.com

Тема письма:
БГУИР. … .