



## 1 Цель работы

Изучение особенностей функций алгебры логики (ФАЛ), построение логических схем по заданной формуле с использованием различных элементарных базисов, изучение методов преобразования ФАЛ, освоение методов минимизации ФАЛ с помощью карт Карно.

## 2 Основные понятия

### 2.1 Понятие комбинационной схемы и функции алгебры логики

Комбинационной логической схемой (КС) называется схема, значение сигнала на выходе которой в любой момент времени однозначно определяется значением сигналов на её входах.

Комбинационная логическая схема имеет  $n$  входов и один выход  $y$  (рис. 1). Входы и выходы являются двоичными, т. е. могут принимать только два значения:  $0$  или  $1$ . Внутренняя структура КС построена также на двоичных элементах.

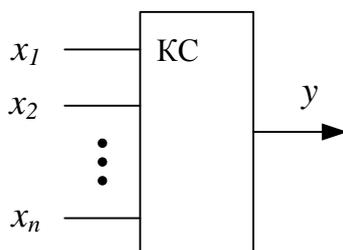


Рис. 1. Комбинационная схема с  $n$  входами

Примером двоичного входа является кнопка  $x$ , которая может быть нажата ( $x = 1$ ) или не нажата ( $x = 0$ ), а примером двоичного выхода – лампочка  $y$ , которая может гореть ( $y = 1$ ) или не гореть ( $y = 0$ ).

Математическим аппаратом для описания КС являются ФАЛ (булевы функции). ФАЛ от  $n$  переменных  $f(x_1, x_2, \dots, x_n)$  можно задать с помощью таблицы истинности (табл. 1).

Таблица 1

Таблица истинности

№	$x_1$	$x_2$	$x_3$	$y$
0	0	0	0	1
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	1
5	1	0	1	0
6	1	1	0	1
7	1	1	1	0

Таблица содержит восемь строк, каждая из которых соответствует одному из возможных состояний всех трех кнопок  $x_1$ ,  $x_2$ ,  $x_3$ . Значение  $y$  в строке определяет состояние лампочки при данном состоянии входов (рис. 2). Например, запись в строке № 0 означает, что если все три кнопки не нажаты ( $x_1 = x_2 = x_3 = 0$ ), то лампочка горит ( $y = 1$ ).

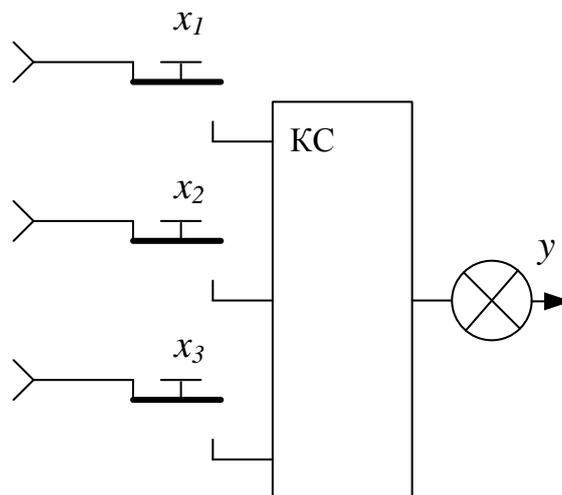


Рис. 2. Комбинационная схема

Особенностью ФАЛ является то, что все переменные и сами функции принимают только два значения. Областью определения ФАЛ от  $n$  переменных является множество двоичных наборов, число которых равно  $2^n$ . Каждому двоичному набору сопоставляется нулевое или единичное значение функции. Соответственно областью значений ФАЛ является множество  $\{0, 1\}$ .

Задать ФАЛ можно перечислением тех двоичных наборов, на которых функция равна 1. Эти наборы называются разрешенными. Например, функция  $y$  из таблицы 1 задается множеством  $(000, 011, 100, 110)$ . Каждый двоичный набор  $N$  есть некоторое двоичное число, которое может быть переведено в десятичное по формуле:

$$N = \sum_{i=0}^n C_i \cdot 2^i, \quad (1)$$

где  $i$  – номер разряда,  $i = 0, 1, 2, \dots, n$ ;

$C_i$  – коэффициент при  $i$ -м разряде,  $C_i = 0$  или 1;

$2^i$  – вес  $i$ -го разряда.

Например,  $110 = 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 6$ .

В таблице 1 в левом крайнем столбце приведены десятичные эквиваленты двоичных чисел. Таким образом, ФАЛ можно задать также с помощью множества десятичных чисел, соответствующих разрешенным двоичным наборам. Например, функция  $y$  из таблицы 1 задается следующим множеством:  $(0, 3, 4, 6)$ .

Можно показать, что число ФАЛ от  $n$  переменных равно  $2^{2^n}$ . Это число быстро растет с ростом  $n$ . Поэтому уже при  $n = 4$  практически нет возможности изучить каждую функцию. Однако оказывается, что в этом нет необходимости. Существуют три функции, называемые *инверсия* (*НЕ* – логическое отрицание), *дизъюнкция* (*ИЛИ* – логическое сложение) и *конъюнкция* (*И* – логическое умножение), с помощью которых можно выразить все другие ФАЛ от любого числа переменных. Данные три функции образуют функционально полную систему функций, или базис.

Функции *И*, *ИЛИ*, *НЕ* задаются соответственно таблицами 2, 3, 4, из которых следует их содержательный смысл. Функция *И* равна 1 только тогда, когда обе переменные равны 1. Обозначается следующим образом:

$$y = x_1 \cdot x_2 = x_1 \& x_2. \quad (2)$$

Функция *ИЛИ* равна 1, если хотя бы одна из переменных равна 1. Обозначается следующим образом:

$$y = x_1 \vee x_2. \quad (3)$$

Функция *НЕ* принимает значение, противоположное значению входной переменной. Обозначается следующим образом:

$$y = \bar{x}. \quad (4)$$

Таблица 2

Таблица истинности функции И

№	$x_1$	$x_2$	$y$
0	0	0	0
1	0	1	0
2	1	0	0
3	1	1	1

Таблица 3

Таблица истинности функции ИЛИ

№	$x_1$	$x_2$	$y$
0	0	0	0
1	0	1	1
2	1	0	1
3	1	1	1

Таблица 4

Таблица истинности функции НЕ

№	$x$	$y$
0	0	1
1	1	0

Введение специальных знаков логических операций *И*, *ИЛИ*, *НЕ* позволяет задавать ФАЛ в алгебраической форме.

Рассмотрим, например, функцию

$$y = \bar{x}_1 \cdot x_2 \cdot x_3 \vee \bar{x}_3 \cdot (\bar{x}_1 \vee x_2). \quad (5)$$

Для того чтобы перейти от алгебраической записи ФАЛ к таблице истинности, используют *аксиомы алгебры логики*, вытекающие непосредственно из таблиц 2, 3, 4:

$$0 \cdot 0 = 0 \cdot 1 = 1 \cdot 0 = 0, \quad 1 \cdot 1 = 1; \quad (6)$$

$$0 \vee 0 = 0, \quad 0 \vee 1 = 1 \vee 0 = 1 \vee 1 = 1; \quad (7)$$

$$\bar{0} = 1, \quad \bar{1} = 0. \quad (8)$$

Рассчитаем, например, значение функции (5) при  $x_1=1$ ,  $x_2=0$ ,  $x_3=1$ :

$$\begin{aligned}
 y &= \bar{x}_1 \cdot x_2 \cdot x_3 \vee \bar{x}_3 \cdot (\bar{x}_1 \vee x_2) = \bar{1} \cdot 0 \cdot 1 \vee \bar{1} \cdot (\bar{1} \vee 0) = \\
 &= 0 \cdot 0 \cdot 1 \vee 0 \cdot (0 \vee 0) = 0 \vee 0 = 0.
 \end{aligned}
 \tag{9}$$

Если рассчитать подобным образом значение функции на всех наборах, то получим таблицу истинности (таблица 1).

Доказательство того факта, что система функций *И*, *ИЛИ*, *НЕ* образует базис, состоит в указании алгоритма обратного перехода от таблицы истинности к алгебраической формуле, содержащей знаки только трех операций  $\{И, ИЛИ, НЕ\}$ . Этот алгоритм заключается в следующем:

1) в таблице истинности выбираются все наборы, на которых функция равна 1;

2) выписываются конъюнкции, соответствующие этим наборам. При этом, если переменная  $x_i$  входит в данный набор как 0, то она записывается с отрицанием, в противном случае она записывается без отрицания;

3) все полученные конъюнкции соединяются между собой знаками дизъюнкции.

Применение данного алгоритма к таблице 1 дает следующий результат:

$$y = \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3 \vee \bar{x}_1 \cdot x_2 \cdot x_3 \vee x_1 \cdot \bar{x}_2 \cdot \bar{x}_3 \vee x_1 \cdot x_2 \cdot \bar{x}_3. \tag{10}$$

Полученная форма записи ФАЛ называется дизъюнктивной совершенной нормальной формой (ДСНФ).

Переход от таблицы истинности к алгебраической записи может быть осуществлен с использованием другого алгоритма:

1) в таблице истинности выбираются все наборы, на которых функция равна 0 (такие наборы называются запрещенными);

2) выписываются дизъюнкции, соответствующие этим наборам. При этом, если переменная  $x_i$  входит в данный набор как 0, то она записывается без отрицания, в противном случае она записывается с отрицанием;

3) все полученные дизъюнкции соединяются между собой знаками конъюнкции.

Применение данного алгоритма к таблице 1 дает следующий результат:

$$y = (x_1 \vee x_2 \vee \bar{x}_3) \cdot (x_1 \vee \bar{x}_2 \vee x_3) \cdot (\bar{x}_1 \vee x_2 \vee \bar{x}_3) \cdot (\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3). \tag{11}$$

Полученная форма записи ФАЛ называется конъюнктивной совершенной нормальной формой (КСНФ).

## 2.2 Законы алгебры логики

Свойства функций *И*, *ИЛИ*, *НЕ* называются законами алгебры логики. Приведем некоторые из них, которые потребуются в дальнейшем:

$$\overline{\overline{x}} = x; \quad (12)$$

$$x \vee x = x; \quad x \cdot x = x; \quad (13)$$

$$x_1 \vee x_2 = x_2 \vee x_1; \quad x_1 \cdot x_2 = x_2 \cdot x_1; \quad (14)$$

$$\overline{x_1 \vee x_2} = \overline{x_1} \cdot \overline{x_2}; \quad (15)$$

$$\overline{x_1 \cdot x_2} = \overline{x_1} \vee \overline{x_2}; \quad (16)$$

$$x_1 \cdot (x_2 \vee x_3) = x_1 \cdot x_2 \vee x_1 \cdot x_3. \quad (17)$$

Функционально полная система функций называется минимальной, если удаление из нее хотя бы одной функции делает систему неполной. Базис  $\{И, ИЛИ, НЕ\}$  не является минимальным, так как из него можно исключить либо функцию *И*, либо функцию *ИЛИ* с сохранением полноты. Это доказывается тем, что функция *И*(*ИЛИ*) может быть выражена через функции *ИЛИ*(*И*) и *НЕ* с помощью формул де Моргана (15) и (16). Взяв отрицание над левой и правой частями формул (15) и (16) и учитывая формулу (12) – закон двойного отрицания, получим:

$$\overline{\overline{x_1 \vee x_2}} = \overline{\overline{x_1 \cdot x_2}}; \quad (18)$$

$$\overline{\overline{x_1 \cdot x_2}} = \overline{\overline{x_1 \vee x_2}}. \quad (19)$$

Таким образом, системы  $\{И, НЕ\}$  и  $\{ИЛИ, НЕ\}$  образуют минимальные базисы. Формулы (18) и (19) позволяют переходить от одной формы записи к другой.

К законам алгебры логики относятся также формулы с константами:

$$x_1 \vee \overline{x_1} = 1; \quad x_1 \vee 0 = x_1; \quad x_1 \vee 1 = 1; \quad (20)$$

$$x_1 \cdot \overline{x_1} = 0; \quad x_1 \cdot 0 = 0; \quad x_1 \cdot 1 = x_1. \quad (21)$$

## 2.3 Реализация функций алгебры логики

Понятие базиса играет существенную роль при реализации ФАЛ на логических элементах. Логическим элементом называется элемент, реализующий некоторую логическую функцию. На рис. 3, а, б, в

приведены соответственно обозначения логических элементов, реализующих функции *И*, *ИЛИ*, *НЕ*.

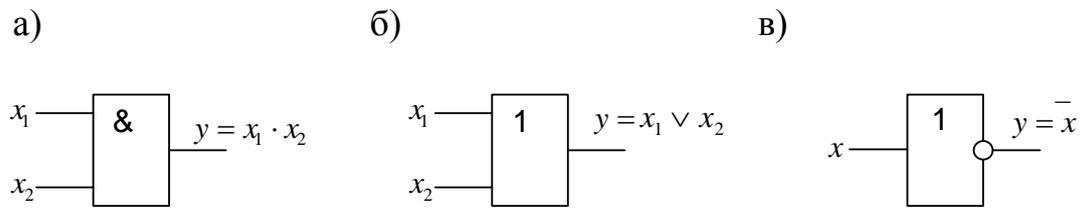


Рис. 3. Логические элементы *И*, *ИЛИ*, *НЕ*

Они могут быть построены на транзисторах, интегральных микросхемах, магнитных кольцах и других элементах. На рис. 4, а, б, в показан пример реализации функций *И*, *ИЛИ*, *НЕ* на релейно-контактных элементах.

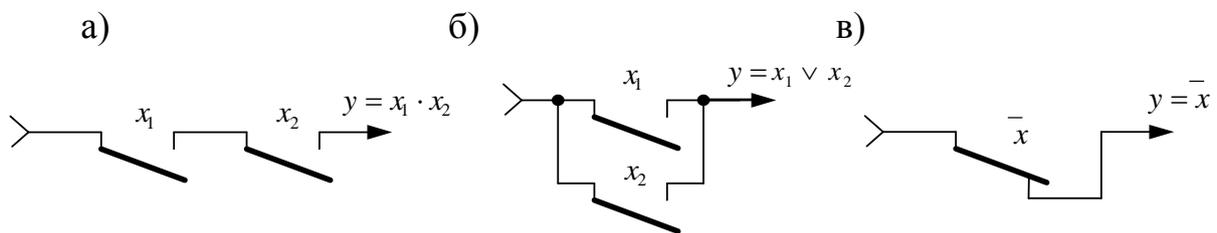


Рис. 4. Реализация функций *И*, *ИЛИ*, *НЕ* на релейно-контактных элементах

*Построение схем в базисе {И, ИЛИ, НЕ}* производится по следующим правилам.

1. Для реализации каждой логической операции используется соответствующий логический элемент.

2. Порядок выполнения операций: *НЕ*, *И*, *ИЛИ*.

3. Выражения в скобках и под знаком отрицания реализуются в первую очередь.

На рис. 5 приведен пример построения схемы для функции  $f = x_3 \cdot (x_2 \vee \bar{x}_1)$  в базисе *{И, ИЛИ, НЕ}*.

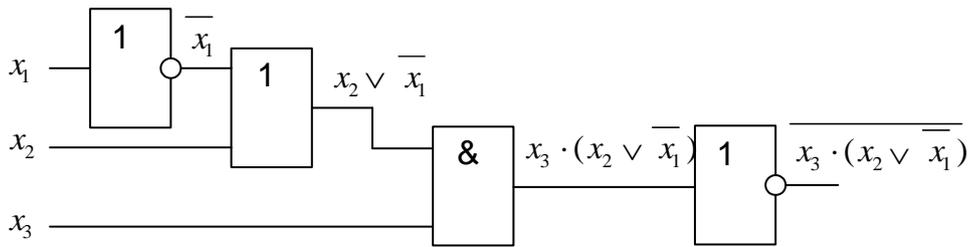


Рис. 5. Реализация ФАЛ в базисе {И, ИЛИ, НЕ}

Построение релейно-контактных схем производится по следующим правилам.

1. Порядок выполнения операций: НЕ, И, ИЛИ.
2. Конъюнкция реализуется за счет последовательного соединения контактов (рис. 4, а), дизъюнкция – за счет параллельного (рис. 4, б).
3. Переменной без отрицания соответствует фронтовой контакт (рис. 4, а, б), переменной с отрицанием – тыловой (рис. 4, в).
4. Выражения в скобках и под знаком отрицания реализуются в первую очередь.
5. Если в формуле есть знак отрицания над выражением, то она преобразуется с помощью формул де Моргана (15), (16) до тех пор, пока знак отрицания не останется только над переменными.

Альтернативой является применение дополнительного реле.

На рис. 6 приведен пример построения схемы по функции  $f = \overline{x_3 \cdot (x_2 \vee \bar{x}_1)}$  с использованием дополнительного реле.

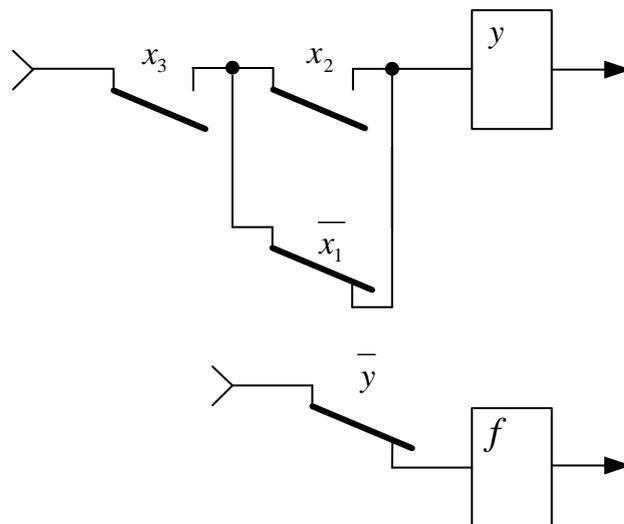


Рис. 6. Реализация схемы с использованием дополнительного реле

Для реализации ФАЛ, заданной логической формулой, необходимо иметь столько типов логических элементов, сколько функций содержится в базе, в котором записана ФАЛ.

Реализация ФАЛ возможна также на базе одного логического элемента. Такими элементами являются:

- 1) элемент *И-НЕ* (рис. 7, а), реализующий функцию Шеффера;
- 2) элемент *ИЛИ-НЕ* (рис. 7, б), реализующий функцию Вебба.

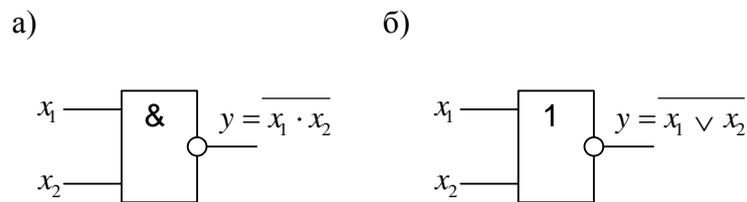


Рис. 7. Реализация функций Шеффера и Вебба на логических элементах

Каждый из этих элементов составляет функционально полную систему алгебры логики. На рис. 8 показана реализация основных ФАЛ на элементе *ИЛИ-НЕ*. На рис. 9 показана реализация основных ФАЛ на элементе *И-НЕ*.

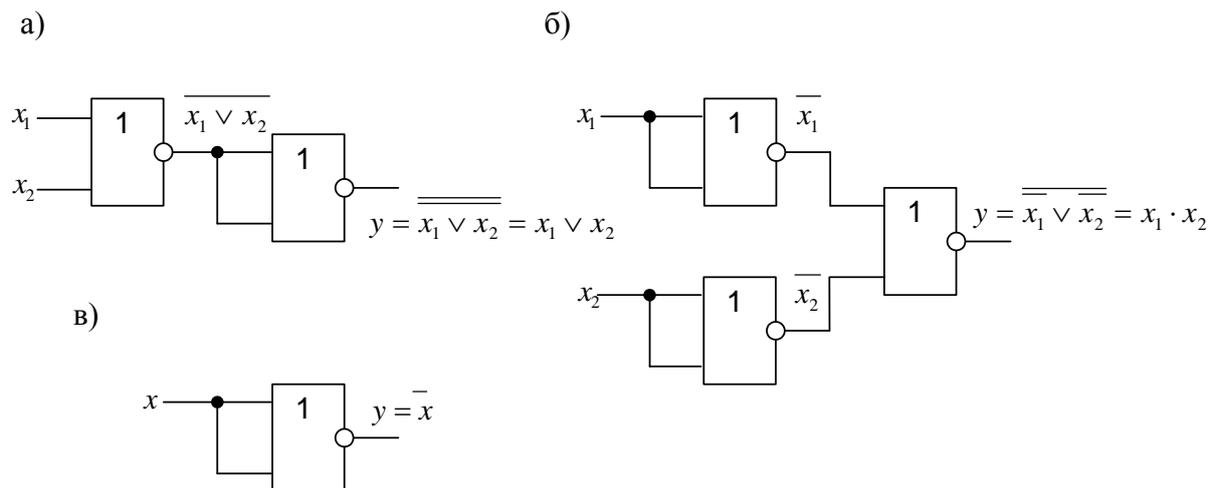


Рис. 8. Реализация функций *И*, *ИЛИ*, *НЕ* с использованием элемента Вебба

а) б)

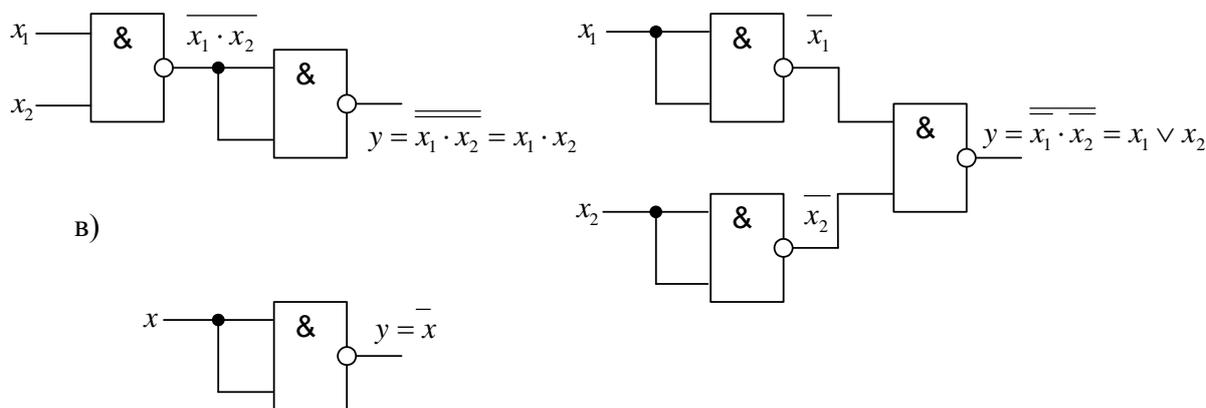


Рис. 9. Реализация функций И, ИЛИ, НЕ с использованием элемента Шеффера

Построение схем в базисах {И-НЕ} и {ИЛИ-НЕ} производится путем замены каждого элемента схемы, реализованной в базисе {И, ИЛИ, НЕ}, на эквивалентный элемент, реализованный в соответствующем базисе (рис. 8, 9).

#### 2.4 Минимизация ФАЛ с помощью карт Карно

Так как одна и та же ФАЛ может быть записана с помощью различных формул, то возникает задача получения минимальной формы записи (с минимальным числом букв), которая потребует минимальных затрат аппаратуры при реализации.

Рассмотрим процесс минимизации ФАЛ с помощью карт Карно. Карта Карно может быть составлена для любого количества переменных, однако удобно работать при количестве переменных не более шести. Карта Карно (рис. 10) является другой формой представления таблицы истинности (табл. 5). Каждая клетка карты соответствует строке таблицы истинности (двоичному набору). Часть клеток (половина), которым соответствует значение  $x_i = 1$ , отмечается чертой. Соответственно в областях, не обозначенных чертой, переменная  $x_i = 0$ .

Таблица 5

Таблица истинности

№	$x_1$	$x_2$	$x_3$	$x_4$	$y$
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	1
5	0	1	0	1	1
6	0	1	1	0	1

7	0	1	1	1	1
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	1
11	1	0	1	1	0
12	1	1	0	0	1
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	1

Для задания ФАЛ в карте Карно надо проставить 1 в тех клетках, которые соответствуют разрешенным наборам. В карте на рис. 10 задана ФАЛ из табл. 5.

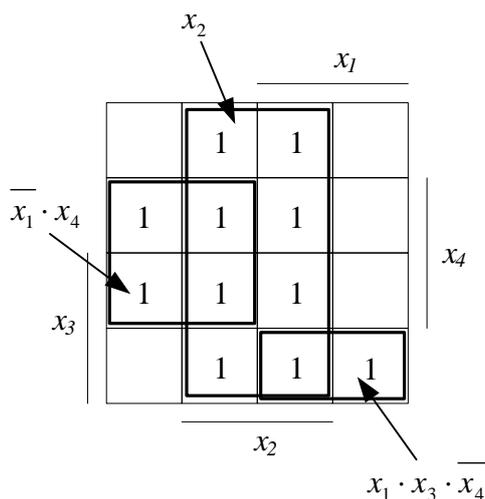


Рис. 10. Карта Карно

Минимизация ФАЛ по карте Карно заключается в объединении соседних клеток в прямоугольные контуры, причем соседними считаются и клетки, разделенные внешней границей карты.

*Минимизация* производится по следующим правилам.

1. Все единицы должны быть включены в прямоугольные контуры.
2. Во всех клетках контура должны стоять единицы.
3. Число клеток в контуре должно быть кратным степени числа 2: 1, 2, 4, 8, ... .
4. Контуры могут накладываться друг на друга.
5. Контуры, все клетки которых уже вошли в другие контуры, являются лишними.
6. Для получения наиболее простой формулы надо выбирать контуры с максимальным числом клеток.

7. Каждому контуру соответствует конъюнкция, составленная из переменных, значения которых не изменяются во всех клетках контура.

8. Конъюнкции контуров объединяются знаками дизъюнкции.

Ниже представлена ФАЛ, соответствующая карте Карно на рис. 10:

$$f = x_2 \vee \overline{x_1} \cdot x_4 \vee x_1 \cdot x_3 \cdot \overline{x_4}.$$

Минимизируем функцию, представленную своими разрешёнными наборами, с использованием карт Карно (рис. 11):

$$f = \{0, 1, 2, 3, 4, 5, 6, 7, 14, 15, 16, 17, 22, 23, 30, 31\}.$$

В карте Карно от пяти переменных два контура, удовлетворяющие указанным правилам, объединяются в один контур, если они расположены симметрично относительно центральной оси. Например, контур на рис. 11, соответствующий конъюнкции  $x_3 \cdot x_4$ .

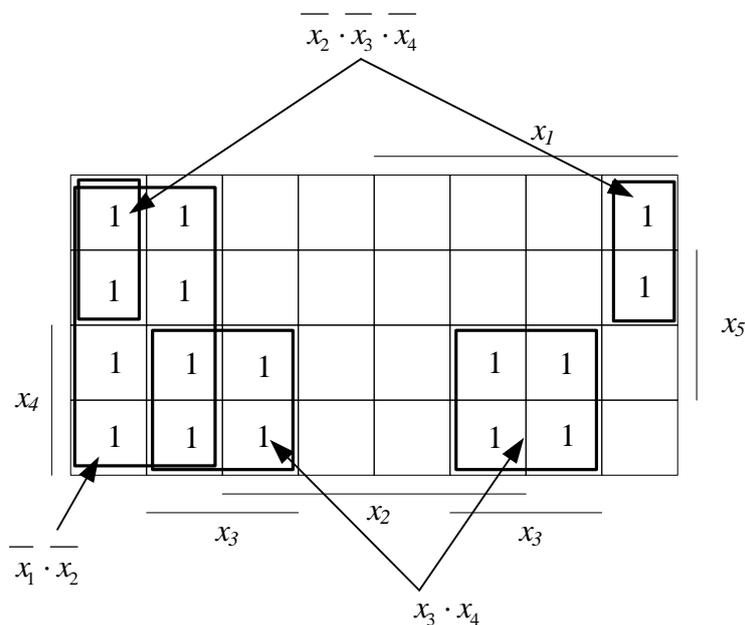


Рис. 11. Минимизация по карте Карно

В итоге мы получаем следующую функцию:

$$f = \overline{x_1} \cdot \overline{x_2} \vee x_3 \cdot x_4 \vee \overline{x_2} \cdot \overline{x_3} \cdot \overline{x_4}.$$

### 3 Методика выполнения работы

1. Ознакомиться с разделом 2 данных методических указаний.

2. Получить вариант у преподавателя.
3. Для функции  $f_1$  выполнить следующее:
  - построить релейно-контактную схему;
  - построить таблицу истинности;
  - записать ДСНФ и КСНФ;
  - построить схему в базисе  $\{И, ИЛИ, НЕ\}$ ;
  - построить схему в базисе Шеффера  $\{И-НЕ\}$ ;
  - построить схему в базисе Вебба  $\{ИЛИ-НЕ\}$ ;
  - записать исходную формулу в базисе  $И, НЕ$ ;
  - записать исходную формулу в базисе  $ИЛИ, НЕ$ .
4. Для функции  $f_2$  выполнить следующее:
  - осуществить минимизацию по карте Карно;
  - записать минимизированную ФАЛ.

## 4 Пример выполнения работы

### 4.1 Исходные данные

$$f_1 = \overline{x_1 \vee x_2} \cdot x_3 \vee x_4; \quad f_2 = \{1, 2, 3, 6, 7, 10, 11, 13, 14\}.$$

### 4.2 Построение релейно-контактной схемы

Исходя из правил построения релейно-контактных схем, функцию  $f_1$  необходимо преобразовать, исключив знаки отрицания над выражением (используем формулы де Моргана (15) и (16)):

$$f_1 = \overline{x_1 \vee x_2} \cdot x_3 \vee x_4 = \overline{x_1} \cdot \overline{x_2} \cdot x_3 \vee x_4.$$

Релейно-контактная схема представлена на рис. 12.

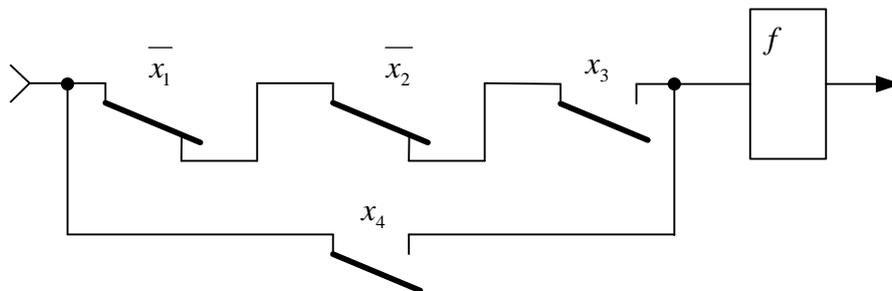


Рис. 12. Релейно-контактная схема

Применим дополнительное реле для построения схемы (рис. 13).

$$f_1 = \overline{x_1 \vee x_2} \cdot x_3 \vee x_4 = \overline{y} \cdot x_3 \vee x_4;$$

$$y = x_1 \vee x_2.$$

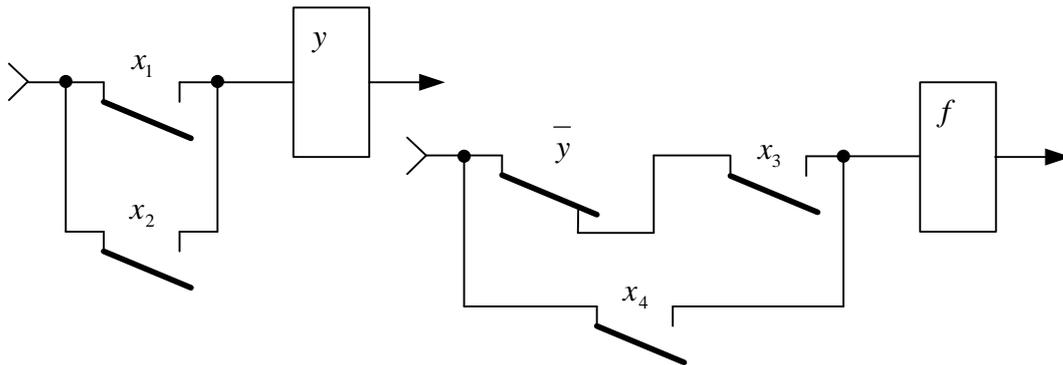


Рис. 13. Релейно-контактная схема с использованием дополнительного реле

### 4.3 Построение таблицы истинности

Построение таблицы удобно проводить поэтапно, выделяя столбец для отдельных выражений исходной формулы (табл. 6).

Таблица 6

Таблица истинности

№	$x_1$	$x_2$	$x_3$	$x_4$	$x_1 \vee x_2$	$\overline{x_1 \vee x_2}$	$\overline{x_1 \vee x_2} \cdot x_3$	$f_1 = \overline{x_1 \vee x_2} \cdot x_3 \vee x_4$
0	0	0	0	0	0	1	0	0
1	0	0	0	1	0	1	0	1
2	0	0	1	0	0	1	1	1
3	0	0	1	1	0	1	1	1
4	0	1	0	0	1	0	0	0
5	0	1	0	1	1	0	0	1
6	0	1	1	0	1	0	0	0
7	0	1	1	1	1	0	0	1
8	1	0	0	0	1	0	0	0
9	1	0	0	1	1	0	0	1
10	1	0	1	0	1	0	0	0
11	1	0	1	1	1	0	0	1
12	1	1	0	0	1	0	0	0
13	1	1	0	1	1	0	0	1
14	1	1	1	0	1	0	0	0
15	1	1	1	1	1	0	0	1

#### 4.4 ДСНФ и КСНФ

ДСНФ исходной функции:

$$f_1 = \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} \cdot \overline{x_4} \vee \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} \cdot x_4 \vee \overline{x_1} \cdot \overline{x_2} \cdot x_3 \cdot \overline{x_4} \vee \overline{x_1} \cdot \overline{x_2} \cdot x_3 \cdot x_4 \vee \overline{x_1} \cdot x_2 \cdot \overline{x_3} \cdot \overline{x_4} \vee \overline{x_1} \cdot x_2 \cdot \overline{x_3} \cdot x_4 \vee \overline{x_1} \cdot x_2 \cdot x_3 \cdot \overline{x_4} \vee \overline{x_1} \cdot x_2 \cdot x_3 \cdot x_4.$$

КСНФ исходной функции:

$$f_1 = (x_1 \vee x_2 \vee x_3 \vee x_4) \cdot (x_1 \vee \overline{x_2} \vee x_3 \vee x_4) \cdot (x_1 \vee \overline{x_2} \vee \overline{x_3} \vee x_4) \& \& (\overline{x_1} \vee x_2 \vee x_3 \vee x_4) \cdot (\overline{x_1} \vee x_2 \vee \overline{x_3} \vee x_4) \cdot (\overline{x_1} \vee \overline{x_2} \vee x_3 \vee x_4) \& \& (\overline{x_1} \vee \overline{x_2} \vee \overline{x_3} \vee x_4).$$

#### 4.5 Построение схемы в базисе {И, ИЛИ, НЕ}

Схема, реализующая функцию  $f_1 = \overline{x_1 \vee x_2 \cdot x_3 \vee x_4}$  в базисе {И, ИЛИ, НЕ}, представлена на рис. 14.

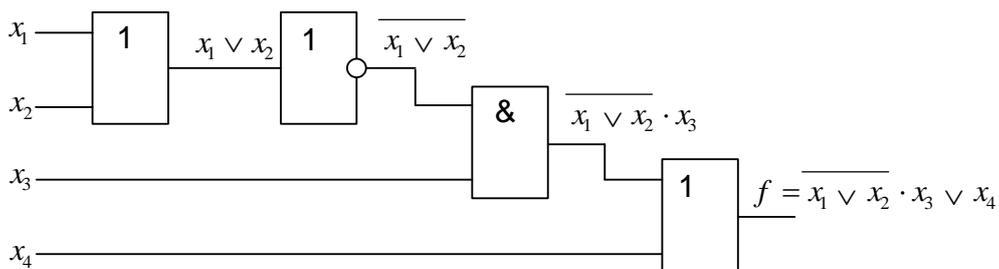


Рис. 14. Реализация функции  $f_1 = \overline{x_1 \vee x_2 \cdot x_3 \vee x_4}$  в базисе {И, ИЛИ, НЕ}

#### 4.6 Построение схемы в базисе Шеффера {И-НЕ}

Схема, реализующая функцию  $f = \overline{x_1 \vee x_2 \cdot x_3 \vee x_4}$  в базисе Шеффера, представлена на рис. 15. Пунктиром показана возможность исключения двух элементов по закону двойного отрицания (12).

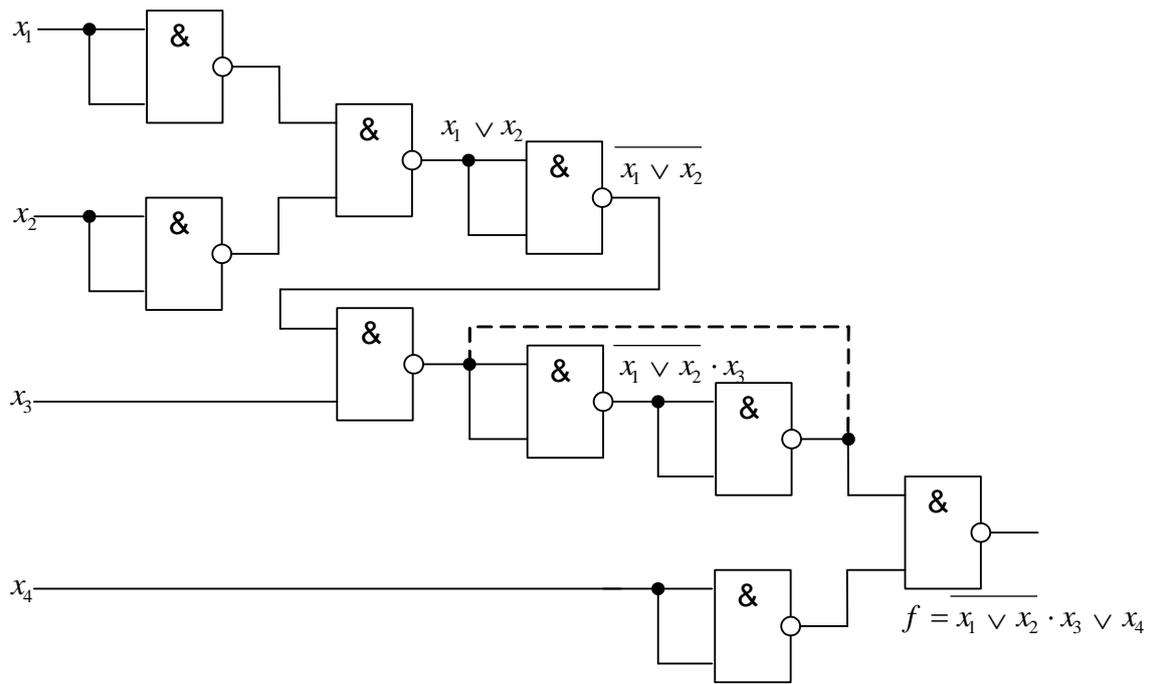


Рис. 15. Реализация функции  $f_1 = x_1 \vee x_2 \cdot x_3 \vee x_4$  в базисе Шеффера

#### 4.7 Построение схемы в базисе Вебба {ИЛИ-НЕ}

Схема, реализующая функцию  $f_1 = x_1 \vee x_2 \cdot x_3 \vee x_4$  в базисе Вебба, представлена на рис. 16. Пунктиром показана возможность исключения двух элементов по закону двойного отрицания (12).

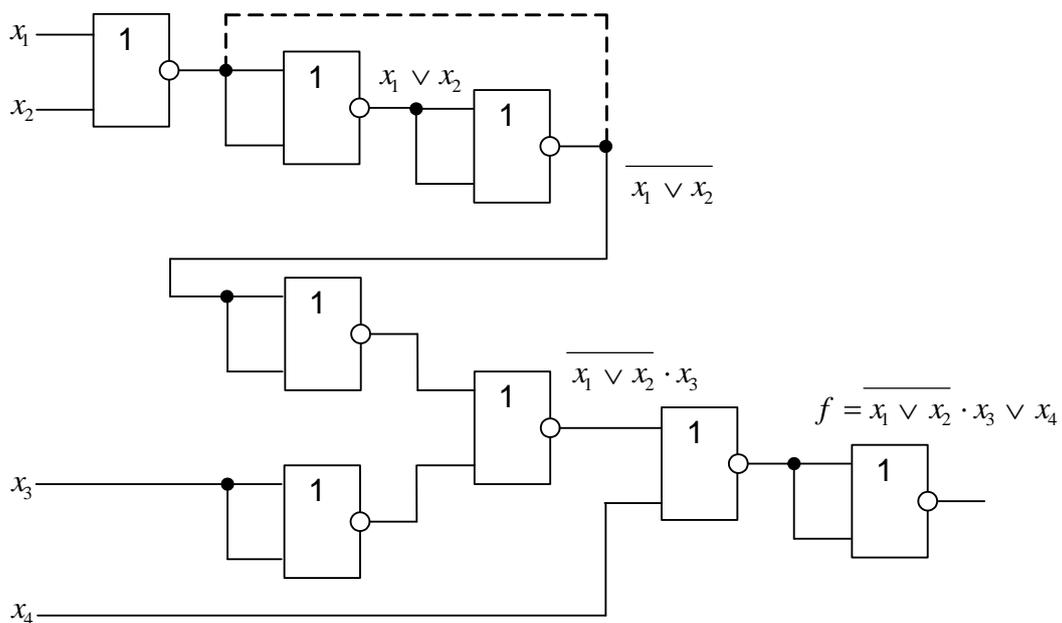


Рис. 16. Реализация функции  $f_1 = x_1 \vee x_2 \cdot x_3 \vee x_4$  в базисе Вебба



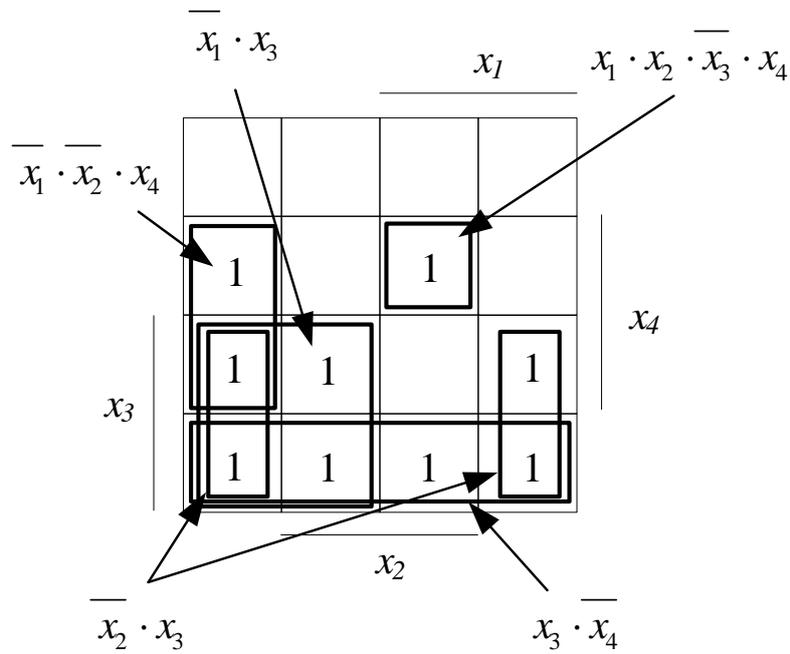


Рис. 17. Минимизация  $f_2$  по карте Карно

Запишем ФАЛ, соответствующую произведенной минимизации:

$$f_2 = x_1 \cdot x_2 \cdot \overline{x_3} \cdot \overline{x_4} \vee \overline{x_2} \cdot \overline{x_3} \vee x_3 \cdot \overline{x_4} \vee \overline{x_1} \cdot x_3 \vee \overline{x_1} \cdot \overline{x_2} \cdot x_4.$$

## 5 Варианты заданий

Варианты заданий представлены в таблице 7.

## Варианты заданий

Номер варианта	Функция $f_1$	Функция $f_2$
1	2	3
1	$f_1 = \overline{(x_1 \cdot x_2 \vee x_3)} \cdot x_4$	$f_2 = \{2,3,6,7,10,11,13,14,15\}$
2	$f_1 = \overline{x_1 \cdot x_2} \cdot (x_3 \vee \overline{x_1} \cdot x_4)$	$f_2 = \{1,5,8,10,12,14\}$
3	$f_1 = (x_1 \vee \overline{x_2} \cdot x_3) \cdot (x_2 \vee \overline{x_4})$	$f_2 = \{1,2,3,5,6,7,14,15\}$
4	$f_1 = \overline{(x_2 \vee x_3 \cdot x_4)} \cdot (\overline{x_1} \vee x_2)$	$f_2 = \{0,1,3,4,5,7,12,13\}$
5	$f_1 = \overline{x_3 \cdot x_4} \vee \overline{x_1} \cdot (x_2 \vee \overline{x_4})$	$f_2 = \{1,3,4,5,7,11,15\}$
6	$f_1 = x_1 \vee x_2 \vee x_3 \vee \overline{x_1} \cdot x_4$	$f_2 = \{1,5,8,9,12,13,14,15\}$
7	$f_1 = (x_3 \vee \overline{x_1}) \cdot (\overline{x_2} \cdot \overline{x_4} \vee \overline{x_3})$	$f_2 = \{0,1,4,5,7,10,11,13,14,15\}$
8	$f_1 = x_1 \cdot (\overline{x_2} \vee x_3) \cdot (\overline{x_4} \vee x_1)$	$f_2 = \{0,1,2,4,5,8,9,12,13,14,15\}$
9	$f_1 = \overline{x_1 \cdot x_2} \cup \overline{x_3 \cdot x_4} \cup x_4$	$f_2 = \{2,3,4,5,10,11\}$
10	$f_1 = \overline{x_1 \vee x_2} \cdot x_4 \vee \overline{x_2} \vee x_3$	$f_2 = \{4,8,9,10,11,12,13,14,15\}$
11	$f_1 = \overline{x_3 \vee x_4} \cdot x_2 \cdot (x_1 \vee \overline{x_4})$	$f_2 = \{0,1,6,7,8,9\}$
12	$f_1 = \overline{(x_1 \vee x_2)} \cdot x_3 \vee \overline{x_3} \cdot x_4$	$f_2 = \{1,5,7,9,13,14,15\}$
13	$f_1 = \overline{(x_1 \cdot x_3 \vee x_4)} \cdot (x_2 \vee x_3)$	$f_2 = \{2,3,5,6,7,8,9,12,13,15\}$
14	$f_1 = \overline{x_2 \vee x_3} \cdot x_4 \vee \overline{x_1} \vee \overline{x_4}$	$f_2 = \{1,3,5,6,7,9,10,11,13,15\}$
15	$f_1 = \overline{(x_1 \vee x_2)} \cdot (x_3 \vee \overline{x_4}) \vee \overline{x_1}$	$f_2 = \{0,1,2,3,4,5,6,7,8,9,11,12,13\}$
16	$f_1 = \overline{(x_1 \vee x_3 \vee x_2)} \cdot \overline{x_4}$	$f_2 = \{3,7,8,10,12,14\}$
17	$f_1 = \overline{(x_1 \cdot x_2 \vee x_3)} \cdot x_4$	$f_2 = \{2,3,4,5,6,7,11,15\}$
18	$f_1 = \overline{(x_1 \cdot x_4 \vee x_2)} \cdot \overline{x_1} \cdot \overline{x_3}$	$f_2 = \{0,1,4,5,7,8,9,11,12,13\}$
19	$f_1 = \overline{x_1 \cdot x_4} \cdot x_3 \vee \overline{x_2} \cdot x_4$	$f_2 = \{0,1,2,3,4,5,6,7,12,15\}$
20	$f_1 = \overline{x_1 \vee x_3} \vee \overline{x_2} \vee \overline{x_1} \cdot \overline{x_4}$	$f_2 = \{1,5,6,8,9,10,11,12,13,14,15\}$
21	$f_1 = \overline{(x_2 \vee x_3)} \cdot (\overline{x_1} \cdot x_4 \vee \overline{x_2})$	$f_2 = \{0,1,2,3,4,5,8,9\}$
22	$f_1 = \overline{(x_1 \cdot x_4 \vee x_3)} \cdot (\overline{x_2} \vee x_1)$	$f_2 = \{0,1,10,11,12,13,14,15\}$
23	$f_1 = \overline{x_4 \cdot x_2} \vee \overline{x_3} \vee \overline{x_2} \cdot x_3$	$f_2 = \{0,2,4,6,11,15\}$
24	$f_1 = \overline{(x_1 \cdot x_4 \vee x_3)} \cdot x_2$	$f_2 = \{3,6,7,11,12,13,14,15\}$
25	$f_1 = \overline{(x_2 \vee x_4)} \cdot x_1 \vee x_3$	$f_2 = \{2,6,9,10,11,13,14,15\}$
26	$f_1 = \overline{x_1 \cdot x_2} \vee \overline{x_3} \vee \overline{x_4}$	$f_2 = \{0,1,3,5,6,7,14,15\}$
27	$f_1 = \overline{(x_1 \vee x_4)} \cdot \overline{x_2} \vee \overline{x_1} \cdot x_3$	$f_2 = \{0,2,3,6,7,9,11,13,15\}$