Тема 10. Методы минимизации булевых функций. Алгоритм Квайна. Функционально полный базис

Контрольные вопросы

1. С какой целью выполняется минимизация булевой функции?
2. Назовите формы представления булевых функций.
3. Перечислите основные правила и законы булевой алгебры.
4. Что такое простая импликанта, сокращенная и минимальная форма функции?
5. Что такое функционально полный базис? Приведите его примеры.

Минимизация булевых функций методом Квайна

Рассмотрим этапы метода Квайна. Пусть, например, задана функция

$$f_{\text{СДНФ}}(x_1x_2x_3x_4) = \vee(0,\ 2,\ 5,\ 6,\ 7,\ 8,\ 9,\ 10,\ 12,\ 13,\ 14),$$

$$f_{\text{СДНФ}} = \underset{1}{\bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4} \vee \underset{2}{\bar{x}_1\bar{x}_2x_3\bar{x}_4} \vee \underset{3}{\bar{x}_1x_2\bar{x}_3x_4} \vee \underset{4}{\bar{x}_1x_2x_3\bar{x}_4} \vee \underset{5}{\bar{x}_1x_2x_3x_4} \vee$$

$$\vee \underset{6}{x_1\bar{x}_2\bar{x}_3\bar{x}_4} \vee \underset{7}{x_1\bar{x}_2\bar{x}_3x_4} \vee \underset{8}{x_1\bar{x}_2x_3\bar{x}_4} \vee \underset{9}{x_1x_2\bar{x}_3\bar{x}_4} \vee \underset{10}{x_1x_2\bar{x}_3x_4} \vee \underset{11}{x_1x_2x_3\bar{x}_4}.$$

На первом этапе метода Квайна выполняется переход от функции, заданной в форме СДНФ, к сокращенной ДНФ. Суть метода заключается в последовательном выполнении всех возможных склеиваний и затем всех поглощений, что приводит к сокращенной ДНФ.

В результате выполнения первого этапа склеивания, в котором участвуют все конституенты единицы, составляющие СДНФ функции, получаются конъюнкции ранга на единицу меньше. Представим первый этап склеивания в виде табл. 10.1.

Таблица 10.1

Номера конституент, участвующих в склеивании	Переменная, по которой склеиваются конституенты	Полученная в результате склеивания импликанта	Номер полученной импликанты
1	2	3	4
1-2	x_3	$\bar{x}_1\bar{x}_2\bar{x}_4$	1
1-6	x_1	$\bar{x}_2\bar{x}_3\bar{x}_4$	2
2-4	x_2	$\bar{x}_1x_3\bar{x}_4$	3
2-8	x_1	$\bar{x}_2x_3\bar{x}_4$	4
3-5	x_3	$\bar{x}_1x_2x_4$	5
3-10	x_1	$x_2\bar{x}_3x_4$	6
4-11	x_1	$x_2x_3\bar{x}_4$	7

Окончание табл. 10.1

1	2	3	4
4-5	x_4	$\bar{x}_1 x_2 x_3$	8
6-7	x_4	$x_1 \bar{x}_2 \bar{x}_3$	9
6-8	x_3	$x_1 \bar{x}_2 \bar{x}_4$	10
6-9	x_2	$x_1 \bar{x}_3 \bar{x}_4$	11
7-10	x_2	$x_1 \bar{x}_3 x_4$	12
8-11	x_2	$x_1 x_3 \bar{x}_4$	13
9-10	x_4	$x_1 x_2 \bar{x}_3$	14
9-11	x_3	$x_1 x_2 \bar{x}_4$	15

На каждом этапе склеивания необходимо проанализировать, все ли исходные конъюнкции (на первом этапе это конституенты единицы) приняли участие в склеивании, и если какая-то из них ни с чем не склеилась, то она становится простой импликантой. В данном примере все конъюнкции исходной функции приняли участие в склеивании, образовав новые конъюнкции, а значит простых импликант на первом этапе не образовалось.

Полученные конъюнкции снова подвергаются склеиванию. Второй этап склеивания представлен в табл. 10.2.

Таблица 10.2

Номера конъюнкций, участвующих в склеивании	Переменная, по которой склеиваются конъюнкций	Полученная в результате склеивания импликанта	Номер полученной импликанты
1-10	x_1	$\bar{x}_2 \bar{x}_4$	1
2-4	x_3	$\bar{x}_2 \bar{x}_4$	1
3-13	x_1	$x_3 \bar{x}_4$	2
4-7	x_2	$x_3 \bar{x}_4$	2
9-14	x_2	$x_1 \bar{x}_3$	3
10-15	x_2	$x_1 \bar{x}_4$	4
11-12	x_4	$x_1 \bar{x}_3$	3
11-13	x_3	$x_1 \bar{x}_4$	4

На втором этапе склеивания исходные конъюнкции с номерами 5, 6, 8 ($\bar{x}_1 x_2 x_4$, $x_2 \bar{x}_3 x_4$, $\bar{x}_1 x_2 x_3$) не приняли участия в склеивании, а значит они включаются во множество простых импликант и войдут в состав сокращенной ДНФ. Остальные наборы в результате склеивания дали еще четыре конъюнкции низшего ранга: $\bar{x}_2 \bar{x}_4$, $x_3 \bar{x}_4$, $x_1 \bar{x}_3$, $x_1 \bar{x}_4$.

Проанализировав возможность склеивания вновь полученных на третьем этапе импликант, мы видим, что дальнейшее склеивание невозможно, а значит четыре конъюнкции, полученные на втором этапе склеивания, тоже являются простыми импликантами и войдут в состав сокращенной ДНФ:

$$f_{\text{сокр ДНФ}} = \bar{x}_1 x_2 x_4 \vee x_2 \bar{x}_3 x_4 \vee \bar{x}_1 x_2 x_3 \vee \bar{x}_2 \bar{x}_4 \vee x_3 \bar{x}_4 \vee x_1 \bar{x}_3 \vee x_1 \bar{x}_4 .$$

Для формирования тупиковой ДНФ строится импликантная таблица, строки которой отмечаются простыми импликантами сокращенной ДНФ, а столбцы − конституентами единицы исходной СДНФ (табл. 10.3). В строке напротив каждой простой импликанты ставится метка под теми наборами (конституентами единицы), на которых она принимает значение «1». Соответствующие конституенты поглощаются (покрываются) данной простой импликантой.

Таблица 10.3

Исходные наборы простых импликант	$\bar{x}_1 \bar{x}_2 \bar{x}_3 \bar{x}_4$ 1	$\bar{x}_1 \bar{x}_2 x_3 \bar{x}_4$ 2	$\bar{x}_1 x_2 \bar{x}_3 x_4$ 3	$\bar{x}_1 x_2 x_3 \bar{x}_4$ 4	$\bar{x}_1 x_2 x_3 x_4$ 5	$x_1 \bar{x}_2 \bar{x}_3 \bar{x}_4$ 6
$\bar{x}_1 x_2 x_4$			*		*	
$x_2 \bar{x}_3 x_4$			*			
$\bar{x}_1 x_2 x_3$				*	*	
$\bar{x}_2 \bar{x}_4$	*	*				*
$x_3 \bar{x}_4$		*		*		
$x_1 \bar{x}_3$						*
$x_1 \bar{x}_4$						*

Продолжение табл. 10.3

Исходные наборы простых импликант	$x_1 \bar{x}_2 \bar{x}_3 x_4$ 7	$x_1 \bar{x}_2 x_3 \bar{x}_4$ 8	$x_1 x_2 \bar{x}_3 \bar{x}_4$ 9	$x_1 x_2 \bar{x}_3 x_4$ 10	$x_1 x_2 x_3 \bar{x}_4$ 11
$\bar{x}_1 x_2 x_4$					
$x_2 \bar{x}_3 x_4$				*	
$\bar{x}_1 x_2 x_3$					
$\bar{x}_2 \bar{x}_4$		*			
$x_3 \bar{x}_4$		*			*
$x_1 \bar{x}_3$	*		*	*	
$x_1 \bar{x}_4$		*	*		*

Из общего числа простых импликант необходимо отобрать их минимальное количество, исключив лишние. Формирование тупиковых форм и выбор минимального покрытия начинается с выявления обязательных простых импликант, то есть таких, которые (и только они) покрывают некоторый исходный набор. В данном случае обязательными простыми импликантами будут $\bar{x}_2\bar{x}_4$ и $x_1\bar{x}_3$, т. к. только $\bar{x}_2\bar{x}_4$ покрывает конституенту с номером 1 и только $x_1\bar{x}_3$ покрывает конституенту с номером 7. Эти наборы обязательно войдут в минимальное покрытие. Эти обязательные простые импликанты покрывают также конституенты с номерами 2, 6, 8, 9, 10. Остались непокрытыми конституенты с номерами 3, 4, 5, 11. Для того, чтобы их покрыть по таблице выбираем простые импликанты $\bar{x}_1x_2x_4$, которая покроет 3 и 5 конституенты, и $x_3\bar{x}_4$, которая покроет 4 и 11 конституенты.

Таким образом, минимальная ДНФ функции

$$f_{\text{МДНФ}} = \bar{x}_2\bar{x}_4 \vee x_1\bar{x}_3 \vee \bar{x}_1x_2x_4 \vee x_3\bar{x}_4.$$

Практическое задание

Выполнить минимизацию булевой функции методом Квайна:

а) $f_{\text{СДНФ}} = x_1x_2x_3\bar{x}_4 + \bar{x}_1x_2\bar{x}_3\bar{x}_4 + \bar{x}_1x_2x_3\bar{x}_4 + x_1x_2\bar{x}_3x_4 + \bar{x}_1\bar{x}_2x_3x_4 + x_1\bar{x}_2\bar{x}_3x_4 + \bar{x}_1\bar{x}_2\bar{x}_3x_4 + x_1\bar{x}_2x_3\bar{x}_4 + \bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4$;

б) $f_{\text{СДНФ}} = x_1x_2x_3\bar{x}_4 + \bar{x}_1x_2\bar{x}_3\bar{x}_4 + \bar{x}_1x_2x_3\bar{x}_4 + \bar{x}_1x_2x_3x_4 + \bar{x}_1\bar{x}_2x_3x_4 + x_1\bar{x}_2\bar{x}_3x_4 + x_1x_2x_3x_4 + x_1\bar{x}_2x_3\bar{x}_4 + \bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4$;

в) $f_{\text{СДНФ}} = x_1\bar{x}_2x_3x_4 + \bar{x}_1x_2x_3\bar{x}_4 + \bar{x}_1x_2x_3x_4 + x_1x_2\bar{x}_3x_4 + \bar{x}_1\bar{x}_2x_3x_4 + x_1\bar{x}_2\bar{x}_3x_4 + \bar{x}_1x_2\bar{x}_3x_4 + x_1\bar{x}_2x_3\bar{x}_4 + \bar{x}_1x_2\bar{x}_3\bar{x}_4$;

г) $f_{\text{СДНФ}} = x_1\bar{x}_2x_3x_4 + \bar{x}_1x_2\bar{x}_3\bar{x}_4 + \bar{x}_1x_2x_3\bar{x}_4 + x_1x_2\bar{x}_3x_4 + \bar{x}_1\bar{x}_2x_3x_4 + x_1\bar{x}_2\bar{x}_3x_4 + \bar{x}_1\bar{x}_2x_3\bar{x}_4 + x_1\bar{x}_2x_3\bar{x}_4$;

д) $f_{\text{СДНФ}} = x_1x_2x_3\bar{x}_4 + x_1x_2\bar{x}_3\bar{x}_4 + \bar{x}_1\bar{x}_2\bar{x}_3x_4 + x_1\bar{x}_2x_3x_4 + \bar{x}_1x_2\bar{x}_3x_4 + x_1\bar{x}_2\bar{x}_3x_4 + x_1\bar{x}_2\bar{x}_3\bar{x}_4 + x_1\bar{x}_2x_3\bar{x}_4 + \bar{x}_1x_2\bar{x}_3\bar{x}_4$;

е) $f_{\text{СДНФ}} = x_1x_2\bar{x}_3x_4 + \bar{x}_1x_2\bar{x}_3\bar{x}_4 + \bar{x}_1x_2x_3\bar{x}_4 + x_1\bar{x}_2x_3x_4 + \bar{x}_1\bar{x}_2x_3x_4 + x_1\bar{x}_2\bar{x}_3x_4 + \bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4 + x_1\bar{x}_2\bar{x}_3\bar{x}_4 + \bar{x}_1\bar{x}_2\bar{x}_3x_4$;

ж) $f_{\text{СДНФ}} = x_1x_2x_3x_4 + \bar{x}_1x_2\bar{x}_3\bar{x}_4 + \bar{x}_1x_2\bar{x}_3x_4 + \bar{x}_1x_2x_3x_4 + \bar{x}_1\bar{x}_2x_3x_4 + x_1\bar{x}_2\bar{x}_3x_4 + \bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4 + x_1\bar{x}_2\bar{x}_3\bar{x}_4$;

з) $f_{\text{СДНФ}} = \bar{x}_1x_2\bar{x}_3x_4 + x_1x_2\bar{x}_3\bar{x}_4 + \bar{x}_1x_2x_3\bar{x}_4 + x_1\bar{x}_2x_3x_4 + \bar{x}_1\bar{x}_2x_3\bar{x}_4 + x_1x_2\bar{x}_3x_4 + \bar{x}_1\bar{x}_2\bar{x}_3x_4 + x_1\bar{x}_2x_3\bar{x}_4$;

и) $f_{\text{СДНФ}} = x_1x_2x_3\bar{x}_4 + \bar{x}_1\bar{x}_2x_3\bar{x}_4 + \bar{x}_1x_2\bar{x}_3x_4 + \bar{x}_1x_2x_3x_4 + \bar{x}_1\bar{x}_2x_3x_4 + x_1\bar{x}_2x_3\bar{x}_4 + \bar{x}_1\bar{x}_2\bar{x}_3x_4 + x_1\bar{x}_2\bar{x}_3\bar{x}_4$;

к) $f_{\text{СДНФ}} = x_1x_2x_3\bar{x}_4 + \bar{x}_1\bar{x}_2x_3\bar{x}_4 + \bar{x}_1x_2\bar{x}_3x_4 + x_1x_2\bar{x}_3\bar{x}_4 + \bar{x}_1\bar{x}_2x_3x_4 + x_1\bar{x}_2x_3\bar{x}_4 + \bar{x}_1\bar{x}_2\bar{x}_3x_4 + x_1\bar{x}_2\bar{x}_3\bar{x}_4$.

Тема 11. Минимизация булевых функций. Карты Вейча (Карно). Минимизация не полностью определенных булевых функций

Контрольные вопросы

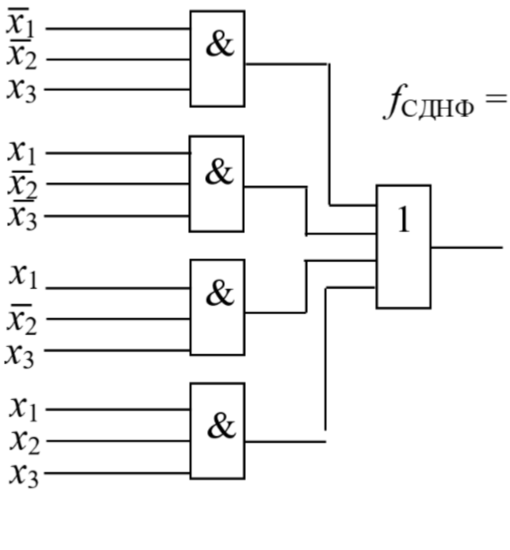
1. Что такое соседние наборы?
2. В чем отличие карт Вейча от карт Карно?
3. Что такое код Грея?
4. Могут ли контуры на минимизирующей карте пересекаться?
5. Сформулируйте основные правила минимизации булевых функций с помощью карт.

Минимизация булевых функций с применением карт Вейча

Таблица11.1

№ п/п	$x_1x_2x_3$	f
0	0 0 0	0
1	0 0 1	0
2	0 1 0	0
3	0 1 1	1
4	1 0 0	0
5	1 0 1	0
6	1 1 0	1
7	1 1 1	1

Более подробно теоретические основы методов минимизации булевых функций (БФ) с использованием карт Вейча (Карно) рассмотрены в [4, с. 95−103]. Остановимся на некоторых примерах минимизации функций, представленных в дизъюнктивной нормальной форме (ДНФ) и конъюнктивной нормальной форме (КНФ), а также более общими случаями их представления − совершенной ДНФ и совершенной КНФ. В качестве исходной формой записи булевой функции для ее минимизации может быть использована таблица истинности (табл. 11.1), схема, описывающая поведение БФ (рис. 11.1), аналитическая форма записи и математическая (более компактная) форма записи БФ. Примеры их приведены ниже.



$$f_{\text{СДНФ}} = \overline{x}_1\overline{x}_2x_3 + x_1\overline{x}_2\overline{x}_3 + x_1\overline{x}_2x_3 + x_1x_2x_3$$

$$f_{\text{СДНФ}} = \text{V } 1, 4, 5, 7$$

Рис. 11.1. Схемное задание БФ

Не останавливаясь на теоретических основах построения карты Вейча и минимизации БФ с ее использованием (подробно рассмотренным в [4]) приведем несколько примеров ее использования.

Пример 1. Выполнить минимизацию БФ (получить $f_{\text{МДНФ}}$ и $f_{\text{МКНФ}}$) заданной аналитически:

$$f_{\text{СДНФ}} = x_1x_2\overline{x}_3\overline{x}_4 + x_1x_2x_3\overline{x}_4 + x_1\overline{x}_2x_3\overline{x}_4 + x_1\overline{x}_2\overline{x}_3\overline{x}_4 + x_1\overline{x}_2x_3x_4 + x_1\overline{x}_2\overline{x}_3x_4 + \overline{x}_1x_2x_3x_4 +$$
$$+ \overline{x}_1\overline{x}_2\overline{x}_3x_4.$$

В результате минимизации получаем на карте четыре контура, объединяющие все восемь единичных наборов рис. 11.2.

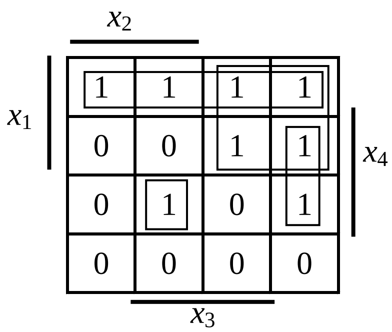


Рис. 11.2. Контуры на единичных наборах для примера 1

Запишем получившуюся (единственную) тупиковую форму:

$$f_{\text{МДНФ}} = x_1\overline{x}_4 + x_1\overline{x}_2 + \overline{x}_2\,\overline{x}_3x_4 + \overline{x}_1x_2x_3x_4.$$

В случае формирования минимальной БФ в форме КНФ контуры образуются на нулевых наборах (рис 11.3).

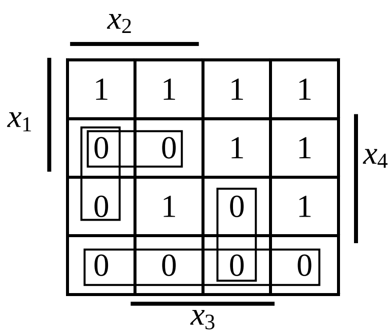


Рис. 11.3. Контуры на нулевых наборах для примера 1

В результате минимизации так же получены четыре контура, образованные на нулевых наборах. Запишем получившуюся тупиковую форму:

$$f_{\text{МКНФ}} = (\overline{x}_2+x_3+\overline{x}_4)\,(\overline{x}_1+\overline{x}_2+\overline{x}_4)\,(x_1+x_2+\overline{x}_3)\,(x_1+x_4).$$

Пример 2. Выполнить минимизацию БФ (получить $f_{\text{МДНФ}}$ и $f_{\text{МКНФ}}$) заданной таблицей истинности (табл. 11.2).

Таблица 11.2

№ п/п	$x_1x_2x_3x_4$	f
1	2	3
0	0 0 0 0	0
1	0 0 0 1	1
2	0 0 1 0	1
3	0 0 1 1	0
4	0 1 0 0	0
5	0 1 0 1	0
6	0 1 1 0	0
7	0 1 1 1	0
8	1 0 0 0	0
9	1 0 0 1	1
10	1 0 1 0	0
11	1 0 1 1	1
12	1 1 0 0	1
13	1 1 0 1	0
14	1 1 1 0	1
15	1 1 1 1	1

По приведенной в таблице истинности функции заполним карту Вейча и выделим на ней контуры (рис. 11.4). Пунктиром обозначены те из них, которые покрывают общие наборы.

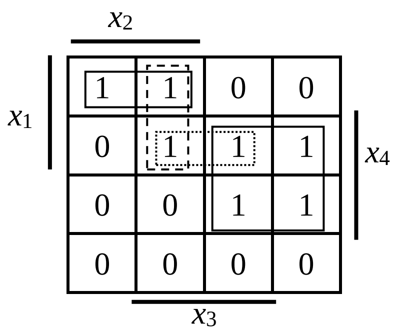


Рис. 11.4. Контуры на единичных наборах для примера 2

В результате получим две тупиковые формы БФ, одновременно и являющимися минимальными формами.

$f^{1}_{\text{МДНФ}} = x_1x_2\overline{x_4} + x_1x_3x_4 + \overline{x_2}x_4$ – 1-я тупиковая форма,
$f^{2}_{\text{МДНФ}} = x_1x_2\overline{x_4} + x_1x_2x_3 + \overline{x_2}x_4$ – 2-я тупиковая форма.

Пример 3. Выполнить минимизацию БФ (получить $f_{\text{МДНФ}}$ и $f_{\text{МКНФ}}$) заданной выражением

$f_{\text{СДНФ}} = \text{V}\ 1, 2, 6, 9, 10, 11, 12, 14, 15.$

В приведенном выражении показаны номера строк таблицы истинности, в которых значение БФ равно единице. Номера строк таблицы истинности начинаются с нуля. Построим карту Вейча для заданной БФ (рис. 11.5).

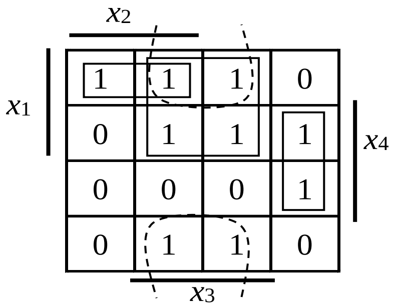


Рис. 11.5. Карта Вейча для ДНФ

Для выделенных на карте контуров, образованных на единичных наборах (см. рис. 11.5), запишем минимальную ДНФ:

$$f_{МДНФ} = x_1x_2\overline{x}_4 + x_3\overline{x}_4 + x_1x_3 + \overline{x}_2\overline{x}_3x_4.$$

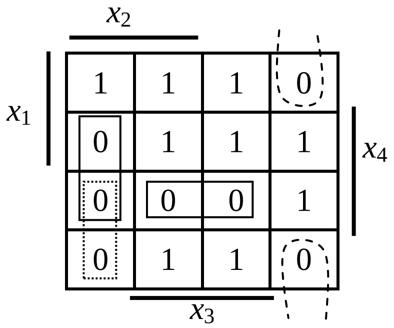


Рис. 11.6. Карта Вейча для КНФ

При формировании контуров на нулевых наборах (рис. 11.6) получим минимальную КНФ:

$$f_{МКНФ} = x_2x_3x_4 + \overline{x}_2x_3\overline{x}_4 + x_1\overline{x}_2x_3 + x_1\overline{x}_3\overline{x}_4.$$

Пример 4. Записать минимальную БФ, заданную картой Вейча (рис. 11.7).

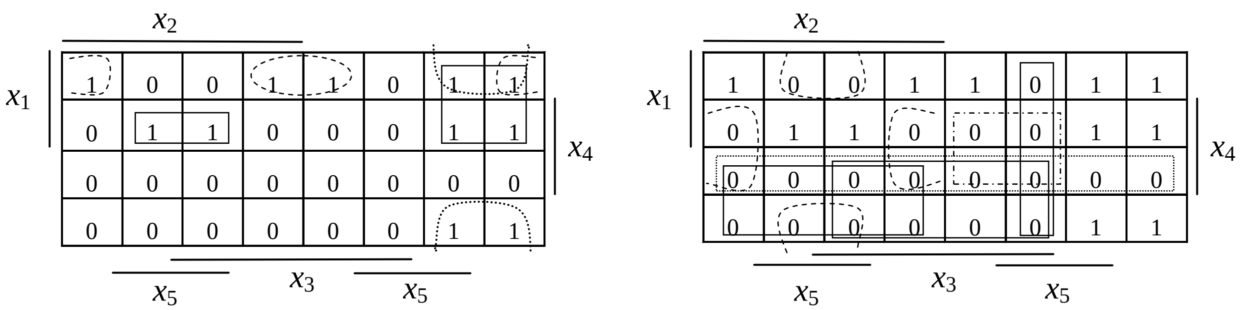


Рис. 11.7. Карта Вейча для примера 4

$f_{МДНФ} = x_1\overline{x}_4\overline{x}_5 + \overline{x}_2\overline{x}_3\overline{x}_4 + x_1\overline{x}_2\overline{x}_3 + x_1x_2x_4x_5,$

$f_{МКНФ} = (\overline{x}_2 + x_4 + \overline{x}_5)(\overline{x}_2 + \overline{x}_4 + x_5)(x_2 + \overline{x}_3 + \overline{x}_4)(x_2 + \overline{x}_3 + \overline{x}_5)(x_1 + \overline{x}_4)(x_1 + \overline{x}_2)(x_1 + \overline{x}_3).$

Р
А

Рассмотрим пример на минимизацию не полностью определенных БФ. В отличие от приведенных выше примеров такие БФ содержат наборы, на которых они не определены (не существуют). Таким наборам можно (для улучшения результата минимизации) приписывать либо единичные, либо нулевые значения.

Пример 5. Для неполностью определенной БФ (рис.11.8) получить минимальную ДНФ (КНФ).

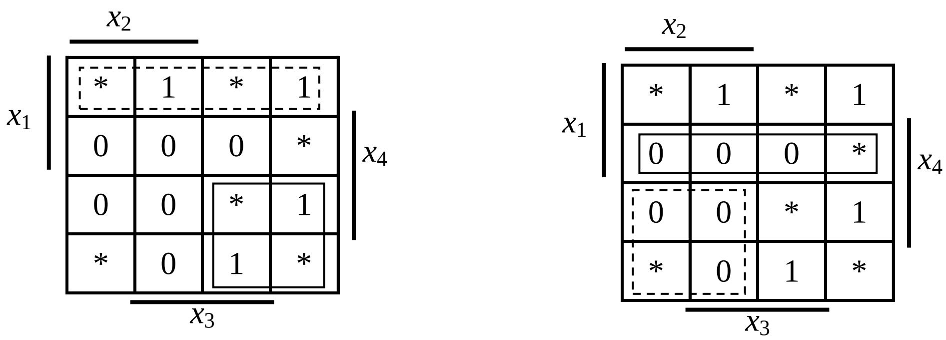


Рис. 11.8. Карта Вейча для примера 5

$f_{\text{МДНФ}} = x_1\overline{x}_4 + \overline{x}_1\overline{x}_2,$

$f_{\text{МКНФ}} = (\overline{x}_1 + \overline{x}_4)(x_1 + \overline{x}_2).$

Минимизация булевых функций с применением карт Карно

Отличие минимизирующих карт Карно от карт Вейча состоит в том, что в картах Карно строкам и столбцам ставятся в соответствие не буквы, составляющие БФ, а двоичные наборы – коды Грея [4, с. 98]. Таким образом, карта Карно наиболее эффективна по отношению к таблицам истинности. Рассматриваемые ниже примеры ориентированы на минимизацию не полностью определенных БФ, как наиболее общий случай БФ. Это же справедливо и для обычных БФ.

Пример 6. Выполнить минимизацию не полностью определенной БФ (получить $f_{\text{МДНФ}}$ и $f_{\text{МКНФ}}$), заданной таблицей истинности (табл. 11.3).

Таблица 11.3

№п.п	$x_1x_2x_3x_4$	f
0	0 0 0 0	0
1	0 0 0 1	1
2	0 0 1 0	*
3	0 0 1 1	1
4	0 1 0 0	*
5	0 1 0 1	0
6	0 1 1 0	0
7	0 1 1 1	0
8	1 0 0 0	0
9	1 0 0 1	1
10	1 0 1 0	0
11	1 0 1 1	1
12	1 1 0 0	1
13	1 1 0 1	*
14	1 1 1 0	1
15	1 1 1 1	1

По приведенной в таблице истинности функции заполним карту Карно и выделим на ней контуры. На рис. 11.9 представлены две карты соответственно для получения минимальной БФ в форме ДНФ и КНФ.

x_1x_2 \ x_3x_4	00	01	11	10
00	0	1	1	*
01	*	0	0	0
11	1	*	1	1
10	1	1	1	0

x_1x_2 \ x_3x_4	00	01	11	10
00	0	1	1	*
01	*	0	0	0
11	1	*	1	1
10	0	1	1	0

Рис. 11.9. Карты Карно для примера 6

$f_{\text{МДНФ}} = x_1x_2 + x_3x_4$,
$f_{\text{МКНФ}} = (x_1 + x_4)(x_1 + \overline{x}_2)(x_2 + \overline{x}_3 + x_4)$.

Пример 7. Выполнить минимизацию, используя карты Карно (рис. 11.10), и получить $f_{\text{МДНФ}}$ и $f_{\text{МКНФ}}$ для БФ, заданной выражением

$f_{\text{СДНФ}}$ = V 0, 3, 8, 12, 14, 24, 17, 19, 26, 28 – единичные наборы.

f^* = V 1, 5, 10, 13, 16, 18, 21, 29, 30 – наборы, на которых БФ не определена.

x_1x_2 \ $x_3x_4x_5$	000	001	011	010	110	111	101	100
00	1	*	1	0	0	0	*	0
01	1	0	0	*	1	0	*	1
11	1	0	0	1	*	0	*	1
10	*	1	1	*	0	0	*	0

x_1x_2 \ $x_3x_4x_5$	000	001	011	010	110	111	101	100
00	1	*	1	0	0	0	*	0
01	1	0	0	*	1	0	*	1
11	1	0	0	1	*	0	*	1
10	*	1	1	*	0	0	*	0

Рис. 11.10. Карты Карно для примера 7

$f_{\text{МДНФ}} = \overline{x}_2\overline{x}_3\overline{x}_4 + \overline{x}_2\overline{x}_3x_5 + x_2\overline{x}_5$,
$f_{\text{МКНФ}} = (x_2 + \overline{x}_4 + x_5)(x_2 + \overline{x}_3)(\overline{x}_2 + \overline{x}_5)$.

Практические задания

1. Выполнить минимизацию БФ по ее числовому заданию. В условии $f_{\text{СДНФ}}$ – единичные наборы, f^* – наборы, на которых БФ не определена.

а) $f_{\text{СДНФ}}$ = V 1, 3, 6, 8, 10, 12, 18, f^* = V 2, 11, 14, 15;
б) $f_{\text{СДНФ}}$ = V 0, 2, 5, 7, 15, 18, 23, 24, 30, f^* = V 1, 3, 12, 14, 19;
в) $f_{\text{СДНФ}}$ = V 1, 2, 4, 16, 17, 19, 22, 29, f^* = V 0, 3, 6, 14, 18;
г) $f_{\text{СДНФ}}$ = V 0, 3, 6, 9, 13, 14, 22, f^* = V 1, 2, 5, 10, 23, 30;
д) $f_{\text{СДНФ}}$ = V 1, 3, 6, 10, 11, 24, 26, 27, 31, f^* = V 5, 7, 12, 25, 30;
е) $f_{\text{СДНФ}}$ = V 1, 4, 5, 7, 19, 24, 28, f^* = V 3, 6, 8, 12, 17;

ж) $f_{СДНФ}$ = V 0, 2, 6, 7, 16, 20, 22, 30, f^* = V 1, 4, 12, 23, 25;
з) $f_{СДНФ}$ = V 0, 6, 9, 13, 16, 18, 24, f^* = V 1, 2, 8, 12, 21, 22;
и) $f_{СДНФ}$ = V 1, 7, 9, 12, 17, 20, 22, 28, f^* = V 2, 3, 5, 14, 30;
к) $f_{СДНФ}$ = V 0, 2, 3, 4, 5, 12, 16, 18, 24, f^* = V 1, 7, 8, 9, 21.

2. Выполнить минимизацию БФ по ее заданию:

1) картой Вейча (записать минимальную БФ в ДНФ);
2) картой Вейча (записать минимальную БФ в КНФ);
3) картой Карно (записать минимальную БФ в ДНФ);
4) картой Карно (записать минимальную БФ в КНФ).

а)

0	*	1	0	0	1	*	0
1	1	0	*	1	1	*	1
1	0	1	*	*	1	*	1
*	1	1	*	0	*	*	0

б)

1	*	1	0	0	0	*	1
*	1	*	*	1	0	*	1
1	0	0	1	*	0	*	1
*	1	1	*	0	0	*	0

в)

0	*	1	0	1	1	*	0
0	0	0	*	1	1	*	1
1	1	1	*	*	1	*	1
*	0	*	*	0	*	*	0

г)

1	*	1	0	0	0	*	0
*	1	*	*	1	0	*	1
1	0	0	1	*	0	1	1
*	0	0	*	0	0	*	0

д)

1	*	1	*	1	1	*	0
0	0	0	*	1	1	*	1
0	*	1	*	*	1	*	0
*	0	*	*	0	*	0	0

е)

0	*	1	0	0	1	*	0
0	0	0	*	1	1	*	1
1	*	1	*	*	1	*	1
*	1	0	*	0	0	*	0

ж)

0	*	1	0	0	0	*	1
*	1	*	*	1	0	*	1
0	*	1	1	*	0	*	0
*	1	1	*	0	0	*	0

з)

0	*	1	0	0	0	*	1
*	1	*	*	1	0	*	1
0	*	1	1	*	0	*	0
*	1	1	*	0	0	*	0

и)

0	*	1	0	1	1	*	0
0	0	0	*	1	1	*	1
1	1	1	*	*	1	*	1
*	0	*	*	0	*	*	0

к)

0	*	1	0	0	0	*	1
*	1	*	*	1	0	*	1
0	*	1	1	*	0	*	0
*	1	1	*	0	0	*	0

Тема 12. Кубическое представление булевых функций. Алгоритм Квайна − Мак-Класки. Алгоритм извлечения (Рота)

Контрольные вопросы

1. Что такое кубическое покрытие булевой функции?
2. Что такое кодовое расстояние?
3. На использовании каких операций основан алгоритм Рота?
4. Что такое L-экстремаль?
5. В чем состоит алгоритм ветвления?

Метод Квайна − Мак-Класки

Рассмотрим пример минимизации логической функции методом Квайна − Мак-Класки. Пусть задана функция

$$f_{\text{СДНФ}}(x_1x_2x_3x_4) = \vee\ (0, 3, 5, 7, 8, 10, 12, 13, 14, 15).$$

Сформируем кубический комплекс K, состоящий из кубов нулевой размерности:

$$K = \{0000, 0011, 0101, 0111, 1000, 1010, 1100, 1101, 1110, 1111\}.$$

На первом этапе рассматриваемого метода все исходные n-кубы разбиваются на непересекающиеся подгруппы по количеству единиц в кубе. Так как склеиваться могут только те кубы, у которых кодовое расстояние равно единице, то имеет смысл попарное сравнение производить только с кубами из соседних групп.

Выполним разбиение комплекса K на группы, получим

$$K_0^0 = \{0000\},\ K_1^0 = \{1000\},\ K_2^0 = \begin{Bmatrix} 0011 \\ 0101 \\ 1010 \\ 1100 \end{Bmatrix},\ K_3^0 = \begin{Bmatrix} 0111 \\ 1101 \\ 1110 \end{Bmatrix},\ K_4^0 = \{1111\}.$$

Попарное сравнение проводим только между соседними по номеру группами кубов. В результате сравнения групп кубов K_0^0 и K_1^0 получим новую группу K_1^1:
K_0^0 и $K_1^0 \Rightarrow K_1^1 = \{x000\}$.

Аналогично анализируем кубы из других соседних групп:

$$K_1^0 \text{ и } K_2^0 \;\Rightarrow K_2^1 = \begin{Bmatrix} 10x0 \\ 1x00 \end{Bmatrix},\; K_2^0 \text{ и } K_3^0 \;\Rightarrow K_3^1 = \begin{Bmatrix} 0x11 \\ 01x1 \\ x101 \\ 1x10 \\ 11x0 \\ 110x \end{Bmatrix},\; K_3^0 \text{ и } K_4^0 \;\Rightarrow K_4^1 = \begin{Bmatrix} x111 \\ 11x1 \\ 111x \end{Bmatrix}.$$

После выполнения первого этапа метода простых импликант не выявлено, т. к. все кубы участвовали в образовании новых кубов. Кубы второй размерности могут образоваться только при склеивании кубов, у которых свободная координата совпадает, следовательно полученные кубы первой размерности разобьем на группы в зависимости от местоположения свободной координаты в кубе:

$$K_1^1 = \begin{Bmatrix} x000 \\ x101 \\ x111 \end{Bmatrix},\; K_2^1 = \begin{Bmatrix} 1x00 \\ 0x11 \\ 1x10 \end{Bmatrix},\; K_3^1 = \begin{Bmatrix} 10x0 \\ 01x1 \\ 11x0 \\ 11x1 \end{Bmatrix},\; K_4^1 = \begin{Bmatrix} 110x \\ 111x \end{Bmatrix}.$$

Выполним сравнение и склеивание кубов внутри каждой из групп. В результате получим кубы второй размерности.

Из первой группы K_1^1 в результате склеивания получим куб второй размерности $x1x1$, а куб $x000$ является простой импликантой, т. к. не принял участия в склеивании кубов.

Из второй группы K_2^1 в результате склеивания получим куб второй размерности $1xx0$, а куб $0x11$ является простой импликантой.

Из третьей группы K_3^1 в результате склеивания получим кубы второй размерности $1xx0$, $x1x1$, $11xx$, простых импликант в этой группе нет, т. к. все кубы приняли участие в образовании кубов новой размерности.

Из четвертой группы K_4^1 в результате склеивания получим куб второй размерности $11xx$.

Среди вновь образованных кубов нет кубов, совпадающих по местоположению свободных координат, таким образом, формирование новых кубов больше не произойдет.

В результате получено множество простых импликант:

$$f_{\text{сокр. ДНФ}} = \{x000,\ 0x11, x1x1,\ 1xx0,\ 11xx,\ \}.$$

На втором этапе метода строится импликантная таблица (табл. 12.1). Формирование минимального покрытия сводится к выявлению обязательных простых импликант и построению на их основе тупиковых форм.

Таблица 12.1

Простые импликанты	Минтермы									
	0000	0011	0101	0111	1000	1010	1100	1101	1110	1111
*x*000	*				*					
0*x*11		*		*						
*x*1*x*1			*	*				*		*
1*xx*0					*	*	*		*	
11*xx*							*	*	*	*

Из табл. 12.1 следует, что простые импликанты *x*000, 0*x*11, *x*1*x*1 являются обязательными и образуют ядро функции, т. е. будут обязательно входить во все тупиковые ДНФ. Эти обязательные импликанты покроют кубы 0000, 1000, 0011, 0111, 0101, 1101, 1111. Оставшиеся две простые импликанты (1*xx*0 и 11*xx*) не являются обязательными. Из исходного множества единичных кубов остались непокрытыми три набора (1010, 1100, 1110), и все они могут быть покрыты одной простой импликантой 1*xx*0.

Следовательно, получена минимальная ДНФ функции:

$$f_{\text{МДНФ}} = \{x000,\ 0x11,\ x1x1,\ 1xx0\},$$

$$f_{\text{МДНФ}} = \bar{x}_2\bar{x}_3\bar{x}_4 \vee \bar{x}_1 x_3 x_4 \vee x_2 x_4 \vee x_1 \bar{x}_4.$$

Алгоритм Рота

Рассмотрим пример минимизации логической функции методом Рота. Исходное покрытие функции задано множествами кубов L (единичных) и N (безразличных):

$$L = \begin{pmatrix} 01100 \\ 01110 \\ 10001 \\ 11000 \\ 11011 \end{pmatrix} \qquad N = \begin{pmatrix} 01000 \\ 10011 \\ 11001 \\ 11010 \\ \end{pmatrix}$$

Алгоритм Рота реализуется в несколько этапов:

− нахождение множества Z простых импликант комплекса K;

− выделение L-экстремалей на множестве Z;

− применение алгоритма ветвления при отсутствии L-экстремалей;

− нахождение абсолютно минимального покрытия из некоторого множества избыточных покрытий.

Первый этап алгоритма Рота – нахождение множества простых импликант. Поиск простых импликант производится пошагово с использованием операции умножения кубов. Операция выполняется путем заполнения таблиц до тех пор, пока происходит образование новых кубов.

Сформируем исходное покрытие C_0, заданное объединением множеств кубов L и N. Выполняется операция (C_0*C_0) (табл. 12.2).

Таблица 12.2

C_0*C_0	01100	01110	10001	11000	11011	01000	10011	11001	11010
01100	–								
01110	011x0	–							
10001			–						
11000				–					
11011					–				
01000	01x00			x1000		–			
10011			100x1		1x011		–		
11001			1x001	1100x	110x1			–	
11010				110x0	1101x				–
A_1	011x0 01x00		100x1 1x001	x1000 1100x 110x0	1x011 110x1 1101x				

В результате операции формируется множество новых кубов A_1, образовавшихся в результате склеивания кубов из исходного множества C_0, а также кубов, которые не образуют новых кубов (включаются в множество Z_0):

$$A_1 = \begin{pmatrix} 011x0 \\ 01x00 \\ 100x1 \\ 1x001 \\ x1000 \\ 1100x \\ 110x0 \\ 1x011 \\ 110x1 \\ 1101x \end{pmatrix}, \quad Z_0 = \varnothing$$

На первом этапе в образовании новых кубов множества A_1 приняли участие все кубы исходного множества C_0, поэтому нет кубов, которые следовало бы включить во множество Z_0. Необходимо также сформировать множество $B_1 = C_0 - Z_0 = C_0$. Далее формируется множество $C_1 = A_1 \cup B_1$, но поскольку все кубы множества B_1 (C_0) уже покрыты кубами множества A_1, то множество $C_1 = A_1$.

Следующий этап – выполнение операции C_1*C_1 представлен в табл. 12.3.

Таблица 12.3

C_1*C_1	011x0	01x00	100x1	1x001	x1000	1100x	110x0	1x011	110x1	1101x
011x0	–									
01x00		–								
100x1			–							
1x001				–						
x1000					–					
1100x						–				
110x0							–			
1x011				1x0x1				–		
110x1			1x0x1				110xx		–	
1101x						110xx				–
A_2			1x0x1	1x0x1		110xx	110xx			

Из полученных новых кубов образуется множество A_2, а из кубов, которые не участвовали в образовании новых, – множество Z_1:

$$A_2 = \begin{Bmatrix} 1x0x1 \\ 110xx \end{Bmatrix}, \quad Z_1 = \begin{Bmatrix} 011x0 \\ 01x00 \\ x1000 \end{Bmatrix}.$$

Множество C_2, как и на предыдущем этапе, формируется из кубов множества A_2. Следующий этап – выполнение операции C_2*C_2 представлен в табл. 12.4. Из табл. 12.4 следует, что A_3 = Ø. Таким образом, новых кубов при выполнении операции C_2*C_2 не было получено:

$$Z_2 = \begin{Bmatrix} 1x0x1 \\ 110xx \end{Bmatrix}, \; C_3 = A_3 = \varnothing.$$

Таблица 12.4

C_2*C_2	1x0x1	110xx
110xx	–	
110xx		–
A_3		

На этом процесс выявления простых импликант окончен. Таким образом, сформировано множество простых импликант Z:

$$Z = \bigcup_{i=0}^{2} Z_i = Z_0 \cup Z_1 \cup Z_2 = \begin{Bmatrix} 011x0 \\ 01x00 \\ x1000 \\ 1x0x1 \\ 110xx \end{Bmatrix}.$$

Далее необходимо выяснить, не содержатся ли в этом множестве «лишние» простые импликанты. Для этого переходим к следующему этапу.

Второй этап алгоритма Рота – определение L-экстремалей. Для определения L-экстремалей выполняется операция вычитания (#) кубов, результат представлен в таблице (табл.12.5).

Таблица 12.5

$z\#(Z-z)$	$011x0$	$01x00$	$x1000$	$1x0x1$	$110xx$
$011x0$	–	$zz0zz$ 01000	$1zyzz$ $x1000$	$y0yzy$ $1x0x1$	$yzyz1$ $110xx$
$01x00$	$zzz1z$ 01110	–	$1zzzz$ 11000	$y0z1y$ $1x0x1$	$yzz11$ $110xx$
$x1000$	$zzyyz$ 01110	$zzzzz$ ø	–	$z0z1y$ $1x0x1$	$zzz11$ $1101x$ $110x1$
$1x0x1$	$yzyzy$ 01110		$zzzzy$ 11000	–	$zzzz0$ 11010 $zzzzz$
$110xx$	$yzyzz$ 01110		$zzzzz$ ø	$z0zzz$ $100x1$	–
Остаток	01110	Ø	Ø	$100x1$	11010

Если после последовательного вычитания из некоторой простой импликанты всех остальных получаем в качестве остатка куб, содержащий единичный набор из множества L, то данная простая импликанта будет обязательной, или L-экстремалью. Проверим, принадлежат ли остатки множеству L с помощью операции пересечения кубов (табл. 12.6). Из табл. 12.6 следует, что в остатках 01110 и $100x1$ содержатся наборы из множества L единичных наборов функции. Это значит, что простые импликанты $011x0$ и $1x0x1$ являются L-экстремалями, а куб 11010 не пересекается с кубами комплекса L, и значит соответствующая ему простая импликанта не является L-экстремалью.

Таблица 12.6

$z\#(Z-z)\cap L$	01100	01110	10001	11000	11011
01110	Ø	01110	Ø	Ø	Ø
100x1	Ø	Ø	10001	Ø	Ø
11010	Ø	Ø	Ø	Ø	Ø

Таким образом, из табл. 12.6 получено множество L-экстремалей:

$$E = \begin{Bmatrix} 011x0 \\ 1x0x1 \end{Bmatrix}.$$

Далее, необходимо выяснить, какие из вершин комплекса L не покрываются L-экстремалями. Для этого из каждого куба комплекса L вычтем (#) элементы множества E (табл. 12.7). В результате вычитания получим $L_1 = L\#E$.

Таблица 12.7

$L\#E$	01100	01110	10001	11000	11011
$011x0$	*zzzzz* Ø	*zzzzz* Ø	*yyyzy* 10001	*yzyzz* 11000	*yzyzy* 11011
$1x0x1$	*yzyzy* 01100	*yzyzy* 01110	*zzzzz* Ø	*zzzzy* 11000	*zzzzz* Ø

Из табл. 12.7 видно, что не покрывается L-экстремалями куб $L_1 = \{11000\}$. Этот куб необходимо покрыть какими-либо простыми импликантами, которые не стали L-экстремалями:

$$\hat{Z} = Z \backslash E = \begin{Bmatrix} 011x0 \\ 01x00 \\ x1000 \\ 1x0x1 \\ 110xx \end{Bmatrix} \backslash \begin{Bmatrix} 011x0 \\ 1x0x1 \end{Bmatrix} = \begin{Bmatrix} 01x00 \\ x1000 \\ 110xx \end{Bmatrix}.$$

Теперь из полученного множества $\hat{Z}$ надо выбрать куб с минимальной ценой (максимальной размерностью), чтобы покрыть набор $L_1 = \{11000\}$. Очевидно, что этот набор можно покрыть двумя простыми импликантами $x1000$ и $110xx$, причем второй куб ($110xx$) имеет большую размерность, а значит именно его нужно включить в минимальное покрытие.

$$f_{\text{МДНФ}} = \{011x0,\ 1x0x1,\ 110xx\},$$

$$f_{\text{МДНФ}} = \overline{x}_1 x_2 x_3 \overline{x}_5 \vee x_1 \overline{x}_3 x_5 \vee x_1 x_2 \overline{x}_3 .$$

Практические задания

1. Выполнить минимизацию булевой функции заданной таблицей истинности методом Квайна – Мак Класки. Таблица истинности задана аналитически: цифры являются номерами строк таблицы истинности, в которых размещены наборы, на которых функция принимает истинное значение.

а) $f_{\text{СДНФ}}$=V 0, 1, 3, 4, 6, 8, 9, 12, 14, 15, 25, 31;

б) $f_{\text{СДНФ}}$=V 2, 3, 5, 7, 8, 10, 12, 13, 15, 18, 19, 25, 27;

в) $f_{\text{СДНФ}}$=V 0, 2, 4, 7, 8, 10, 12, 15, 18, 27, 31;

г) $f_{\text{СДНФ}}$=V 0, 2, 4, 6, 7, 8, 10, 12, 14, 16, 17, 18, 23;

д) $f_{\text{СДНФ}}$=V7, 8, 9, 12, 13, 14, 15, 20, 21, 25, 28, 30, 31;

е) $f_{СДНФ}$=V 0, 1, 3, 5, 6, 7, 8, 9, 14, 15, 16, 17, 21, 29, 31;

ж) $f_{СДНФ}$=V 0, 3, 4, 7, 8, 9, 11, 12, 13, 15, 24, 31;

з) $f_{СДНФ}$=V 0, 1, 2, 5, 7, 8, 9, 10, 13, 15, 17, 21, 29;

и) $f_{СДНФ}$=V 0, 2, 4, 6, 7, 10, 12, 14, 15, 17, 20, 21, 25;

к) $f_{СДНФ}$=V 2, 3, 4, 5, 7, 8, 12, 18, 21, 22, 23, 24, 28, 30, 31.

2. Используя алгоритм Рота получить минимальную форму булевой функции, исходное покрытие которой задано множествами кубов L (единичных) и N (безразличных):

а) $L=\begin{pmatrix}01100\\01110\\10001\\11000\\11011\end{pmatrix}$, $N=\begin{pmatrix}01000\\10011\\11001\\11010\end{pmatrix}$; б) $L=\begin{pmatrix}0000\\0010\\0110\\1101\\1011\\1001\\1000\end{pmatrix}$, $N=\varnothing$; в) $L=\begin{pmatrix}01101\\01000\\10011\\11001\\11011\end{pmatrix}$, $N=\begin{pmatrix}01001\\10010\\10001\\11010\end{pmatrix}$;

г) $L=\begin{pmatrix}0101\\1101\\0x10\\1000\\1010\end{pmatrix}$, $N=\begin{pmatrix}0x00\\0111\\1011\end{pmatrix}$; д) $L=\begin{pmatrix}0001\\0100\\1100\\1x01\\x110\\0010\end{pmatrix}$, $N=\varnothing$; е) $L=\begin{pmatrix}1001\\010x\\0111\\1110\end{pmatrix}$, $N=\begin{pmatrix}00x0\\1101\\1011\end{pmatrix}$;

ж) $L=\begin{pmatrix}00100\\01000\\01001\\01100\\01111\\10000\\10001\end{pmatrix}$, $N=\begin{pmatrix}00000\\01101\\01110\\10100\end{pmatrix}$; з) $L=\begin{pmatrix}1000\\010x\\0011\\1110\end{pmatrix}$, $N=\begin{pmatrix}0000\\1101\\10x1\end{pmatrix}$; и) $L=\begin{pmatrix}0001\\0010\\0101\\1000\\1100\\1101\\1110\end{pmatrix}$, $N=\begin{pmatrix}0000\\1010\\0111\end{pmatrix}$;

к) $L=\begin{pmatrix}1000\\0101\\0011\\11x0\end{pmatrix}$, $N=\begin{pmatrix}0000\\11x1\\1001\end{pmatrix}$.