ЛЕКЦИЯ 5. БУЛЕВЫ ФУНКЦИИ (ПРОДОЛЖЕНИЕ)

МИНИМИЗАЦИЯ НОРМАЛЬНЫХ ФОРМ (ПРОДОЛЖЕНИЕ)

АЛГОРИТМ КВАЙНА-МАККЛАСКИ

Единичный куб и карты Карно трудно превратить в алгоритм для компьютера. Квайн и Маккласки предложили алгоритм, основанный на следующей идее:

- на первом этапе находятся элементарные конъюнкции – кандидаты на включение в минимальную ДНФ;
- на втором определяются те из них, которые действительно будут включены.

Продемонстрируем работу этого алгоритма на примере.

П р и м е р . Используя метод Квайна-Маккласки, найти минимальную ДНФ для функции

$$f = \bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4 + \bar{x}_1\bar{x}_2x_3x_4 + \bar{x}_1x_2\bar{x}_3x_4 + \bar{x}_1x_2x_3x_4 + x_1\bar{x}_2\bar{x}_3\bar{x}_4 + x_1\bar{x}_2x_3x_4 + x_1x_2x_3\bar{x}_4 .$$

Первый этап (склеивание и поглощение). Запишем каждую из элементарных конъюнкций в виде битовой строки и отсортируем битовые строки по числу единиц:

Шаг 0		Шаг 1		Шаг 2	
~~1~~	0001	~~1,2~~	00*1	1,2,3,5	0**1
	------	~~1,3~~	0*01		
~~2~~	0011	~~2,5~~	0*11		
~~3~~	0101	2,6	*011		
~~4~~	1010	~~3,5~~	01*1		
	-----	4,6	101*		
~~5~~	0111	4,7	1*10		
~~6~~	1011				
7	1110				

Элементарные конъюнкции можно объединить, если они различаются ровно по одной переменной. Следовательно, объединение битовых строк возможно, если число символов 1 отличается в них ровно на единицу, то есть объединение возможно только для строк из двух соседних групп. При объединении отсутствующую переменную будем обозначать * (получаем что-то вроде *маски*).

Затем объединяются попарно произведения, состоящие из двух переменных: в объединённых строках появляются уже две *. И т.д., пока это возможно.

Второй этап (построение минимального множества произведений). Мы используем не вычеркнутые строки, которые не использовались при дальнейших объединениях, и начинаем со строк, имеющих наименьшее количество переменных (т.е. наибольшее количество «*») в своей записи. Строим таблицу, в которой по строкам записываются произведения-кандидаты, а по столбцам - исходные элементарные конъюнкции. Нужно найти оптимальное покрытие нашей функции битовыми строками-масками (это означает, что в каждом из столбцов должен иметься хотя бы один X).

	0001	0011	0101	1010	0111	1011	1110
0**1	X	X	X		X		
*011		X				X	
101*				X		X	
1*10				X			X

В качестве окончательного ответа можно выбрать или $\bar{x}_1x_4 + x_1\bar{x}_3\bar{x}_4 + \bar{x}_2x_3x_4$, или $\bar{x}_1x_4 + x_1\bar{x}_3\bar{x}_4 + x_1\bar{x}_2x_3$.

ОБЩИЙ АЛГОРИТМ. Как показано в примере, алгоритм Квайна-Маккласки имеет следующую последовательность шагов для минимизации ДНФ:

1. Выразим каждую из элементарных конъюнкций через битовую строку длины n.
2. Группируем битовые строки по числу единиц.
3. Найдём пары битовых строк, отличающихся в одной позиции; объединим их, заменив эту позицию *.
4. Найдём пары строк, полученных на предыдущем этапе, отличающихся в одной позиции; объединим их, заменив эту позицию *.
5. Продолжаем объединять строки до тех пор, пока это возможно.
6. Отметим все битовые строки, которые не использовались для дальнейшего объединения.
7. Найдём минимальное множество отмеченных булевых строк, покрывающих исходную булеву функцию. **Этот шаг является наиболее трудоёмким – для его реализации возможно использовать поиск с возвращением, т.н. backtracking.**

ФУНКЦИОНАЛЬНАЯ ПОЛНОТА

ДИЗЪЮНКЦИЯ, КОНЪЮНКЦИЯ, ОТРИЦАНИЕ

В любом компьютере используется двоичная система счисления $\Rightarrow$ любая команда процессора – это булева функция. Из теоремы о существовании СДНФ следует, что любая булева функция от любого числа переменных выражается через К, Д и О. Т.е. этих функциональных элементов достаточно для построения любого процессора.

А можно ли обойтись меньшим количеством элементов?

ПОЛНЫЕ СИСТЕМЫ ФУНКЦИЙ

Определение 1. Пусть $F = \{f_1, ..., f_n\}$ - система булевых функций. Замыканием $[F]$ называется множество всех функций, которые можно получить суперпозицией функций из F.

Определение 2. Система функций называется замкнутой, если $[F] = F$.

Определение 3. Система функций называется полной, если $[F] = P_n$, где P_n - множество всех булевых функций от n переменных.

Пример 1. Система $\{\wedge \vee \neg\}$ - полная. Это следует из теоремы о существовании СДНФ.

Пример 2. Система $\{\wedge \neg\}$ - тоже полная (закон де Моргана).

Пример 3. Система $\{\vee \neg\}$ - тоже полная (закон де Моргана).

Пример 4. Система $\{\downarrow\}$ - полная.

$x \downarrow y = \overline{x \vee y}$

$x \downarrow x = \overline{x \vee x} = \bar{x}$

$x \wedge y = \overline{\bar{x} \vee \bar{y}} = (x \downarrow x) \downarrow (y \downarrow y)$

Пример 5. Система $\{|\}$ - полная.

$x | y = \overline{x \wedge y}$

$x \downarrow x = \overline{x \wedge x} = \bar{x}$

$x \vee y = \overline{\bar{x} \wedge \bar{y}} = (x | x) | (y | y)$

КРИТЕРИЙ ПОСТА

О п р е д е л е н и е **4.** Система функций F называется *предполной*, если она не полная, но после добавления к ней любой функции, которая не входит в $[F]$, она становится полной.

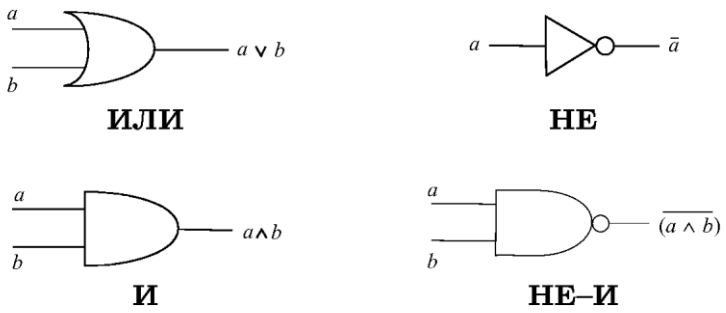
Оказывается (можно это доказать), что множество всех предполных замкнутых классов исчерпывается следующим списком:

1) T_0 - функции, сохраняющие 0;
2) T_1 - функции, сохраняющие 1;
3) S - самодвойственные функции (т.е. при замене всех аргументов на противоположные, значение функции тоже меняется на противоположное);
4) M - монотонные функции (если все аргументы меньше или равны, то и значение функции меньше или равно);
5) L - линейные функции (это *полиномы Жегалкина* первой степени: $a_0 \oplus a_1 x_1 \oplus ... \oplus a_n x_n$).

Т е о р е м а (*критерий Поста*). Система функций F является полной тогда и только тогда, когда она не принадлежит целиком ни одному из классов T_0, T_1, S, M, L.

Д о к а з а т е л ь с т в о : без доказательства.

СХЕМЫ ИЗ ФУНКЦИОНАЛЬНЫХ ЭЛЕМЕНТОВ*



П р и м е р .

Пример 9.8. Что получится на выходе функциональной схемы, представленной на рис. 9.9?

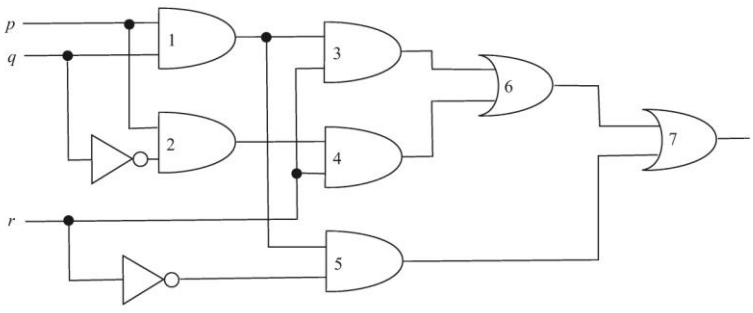


Рисунок 9.9. Функциональная схема

Таблица 9.10

Вентиль	Вход	Выход
1	$p,\ q$	pq
2	$p,\ \bar{q}$	$p\bar{q}$
3	$pq,\ r$	pqr
4	$p\bar{q}, r$	$p\bar{q}r$
5	$pq,\ \bar{r}$	$pq\bar{r}$
6	$pqr,\ p\bar{q}r$	$pqr \vee p\bar{q}r$
7	$pqr \vee p\bar{q}r,\ pq\bar{r}$	$pqr \vee p\bar{q}r \vee pq\bar{r}$

Таким образом, на выходе схемы получится функция $pqr \vee p\bar{q}r \vee pq\bar{r}$.

Упростим эту схему: $pqr \vee p\bar{q}r \vee pq\bar{r} = pq \vee pr = p(q \vee r)$, то есть упрощённая схема будет содержать всего два элемента:

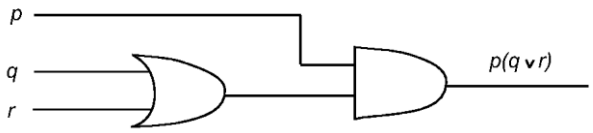


Рисунок 9.11.

КОНТАКТНЫЕ СХЕМЫ*

КОНТАКТ

- устройство, которое в процессе работы может быть в двух состояниях: замкнутом или разомкнутом.

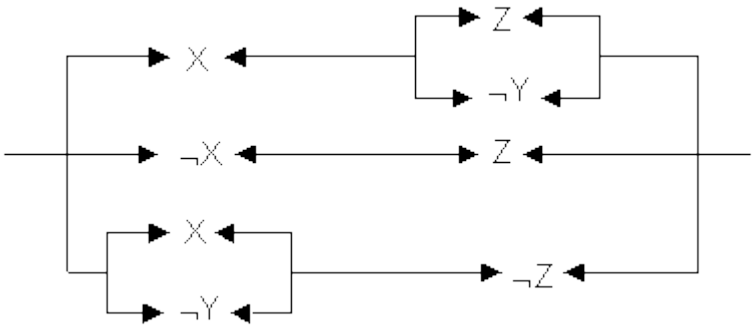
Все контакты делятся на два класса: нормально разомкнутые и нормально замкнутые. Нормально разомкнутые – разомкнуты при X=0, нормально замкнутые – замкнуты при X=0.

Контакты X и Y можно соединять между собой параллельно или последовательно:

С помощью таких соединений из контактов можно составлять **контактные схемы.**

Любую контактную схему можно описать логическим выражением, содержащим три операции: отрицание, дизъюнкцию, конъюнкцию.

Пример 1. Упрощение схемы.



$$f(X,Y,Z) = X(Z \vee \bar{Y}) \vee \bar{X}Z \vee (X \vee \bar{Y})\bar{Z}$$

Упростим эту схему:

$$f(X,Y,Z) = XZ \vee X\bar{Y} \vee \bar{X}Z \vee \overline{YZ}$$

X	Y	Z	f
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

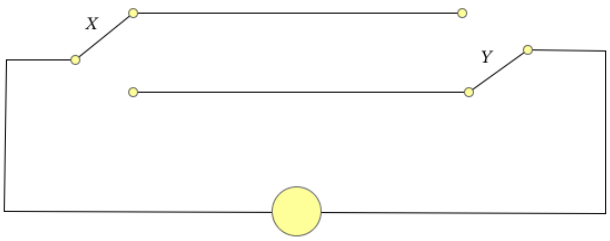
$$f = \bar{Y} \vee Z$$

П р и м е р 2. Требуется, чтобы включение света в комнате осуществлялось с помощью **двух** различных выключателей таким образом, чтобы нажатие на любой из них приводило к включению света, если он был выключен, и выключению, если он был включен. Построить по возможности более простую цепь, удовлетворяющую этому требованию.

$$f(x,y) = x\bar{y} \vee \bar{x}y$$

X	Y	f	f
0	0	a	0
0	1	-a	1
1	0	-a	1
1	1	a	0

Физическая реализация:



П р и м е р 3. Требуется, чтобы включение света в комнате осуществлялось с помощью **трех** различных выключателей таким образом, чтобы нажатие на любой из них приводило к включению света, если он был выключен, и выключению, если он был включен. Построить по возможности более простую цепь, удовлетворяющую этому требованию.

X	Y	Z	f	f
0	0	0	a	0
0	0	1	-a	1
0	1	0	-a	1
0	1	1	a	0
1	0	0	-a	1
1	0	1	a	0
1	1	0	a	0
1	1	1	-a	1

$$f(x,y,z) = x\overline{yz} \vee \bar{x}y\bar{z} \vee \overline{xy}z \vee xyz$$