Логические функции

fpmi.ami.nstu.ru

Учебные материалы

Дискретная математика

2. Переключательные функции

Задачи по булевой алгебре

Переключательные (логические) функции

$$f:\{0,1\}^n \to \{0,1\} \qquad \text{(или } f:\{И,\ Л\}^n \to \{И,\ Л\}\text{)}$$

***Набор* (*набор переменных*)** – значения переменных.

Область определения логической функции – множество всевозможных наборов переменных.

Множество наборов переменных можно интерпретировать как вершины n-мерного куба.

Количество различных переключательных функций n аргументов $2^{\left(2^n\right)}$

Способы задания логических функций:

1) таблицей истинности,

2) набором значений
$f(a, b, c)$=(00000111)=(0000 0111),

3) формулой
$f(a, b, c) = a \wedge (b \vee c)$.

a b c	f
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

Переключательные функции 2-х аргументов

№ п/п	x: y:	0011 0101	Обозначения функции	Названия функции
0		0000	const 0	Константа 0
1		0001	$x \wedge y$, xy	Конъюнкция, функция ‘и’
2		0010	$x \leftarrow y$, $x\,\bar{y}$	Запрет ‘y’
3		0011	x	Повтор ‘x’
4		0100	$x \rightarrow y$, $\bar{x}\,y$	Запрет ‘x’
5		0101	y	Повтор ‘y’
6		0110	$x \oplus y$	Сумма по модулю 2, исключающее ‘или’
7		0111	$x \vee y$	Дизъюнкция, соединительное ‘или’
8		1000	$x \downarrow y$	Стрелка Пирса
9		1001	$x \sim y$, $x \Leftrightarrow y$	Эквивалентность
10		1010	$\bar{y}$, $\neg y$	Отрицание y, функция ‘не’
11		1011	$x \subset y$, $x \Leftarrow y$	Обратная импликация
12		1100	$\bar{x}$, $\neg x$	Отрицание x, функция ‘не’
13		1101	$x \supset y$, $x \Rightarrow y$	Прямая импликация
14		1110	x/y	Штрих Шеффера
15		1111	const 1	Константа 1

Формула F называется

– ***тавтологией*** (***тождественно-истинной формулой*** – ТИФ), если при любых значениях переменных $x_1, x_1, ..., x_n$ соответствующая ей функция принимает значение 1;

– ***выполнимой*** (***условно-истинной формулой*** – УИФ), если при некоторых значениях переменных $x_1, x_1, ..., x_n$ соответствующая ей функция принимает значение 1;

– ***тождественно-ложной формулой*** – ТЛФ, если при любых значениях переменных $x_1, x_1, ..., x_n$ соответствующая ей функция принимает значение 0;

– ***опровержимой*** (***условно-ложной формулой***) , если при некоторых значениях переменных $x_1, x_1, ..., x_n$ соответствующая ей функция принимает значение 0.

Эквивалентные преобразования формул:

– ***правило подстановки*** формулы F вместо переменной x: все вхождения переменной x в исходное соотношение должны быть одновременно заменены формулой F.

– ***правило замены*** подформул: если какая-либо формула F, описывающая функцию f, содержит F_1 в качестве подформулы, то замена F_1 на эквивалентную F_2 не изменит функцию f.

Основные теоремы переключательных функций

1. a) $\overline{x / y} = xy$, b) $\overline{xy} = x / y$,
2. a) $\overline{x \sim y} = x \oplus y$, b) $\overline{x \oplus y} = x \sim y$,
3. a) $\overline{x \downarrow y} = x \vee y$, b) $\overline{x \vee y} = x \downarrow y$,
4. a) $y \rightarrow x = x \leftarrow y$, b) $y \subset x = x \supset y$,
5. a) $\overline{x \supset y} = x \leftarrow y$, b) $\overline{x \leftarrow y} = x \supset y$,
6. a) $x \supset y = \bar{x} \vee y$, b) $\overline{x \supset y} = x \bar{y}$,
7. $y \vee x = x \oplus y \oplus xy$,
8. $x(y \oplus z) = xy \oplus xz$,
9. $x \oplus y = \bar{x} y \vee x \bar{y}$,
10. $x \sim y = xy \vee \bar{x}\, \bar{y}$,
11. $x / x = \bar{x}$,
12. a) $x \oplus x = 0$, b) $x \oplus 1 = \bar{x}$, c) $x \oplus 0 = x$,
13. a) $\bar{1} = 0$, b) $\bar{0} = 1$, c) $\bar{\bar{x}} = x$.

Старшинство операций (по убыванию)

$\neg$	$/$	$\downarrow$	$\rightarrow$ $\leftarrow$	$\wedge$	$\vee$	$\oplus$	$\supset$ $\subset$	$\sim$

Булева алгебра [логических функций]

Основное множество – все множество логических функций;

Сигнатура булевой алгебры – $\{\wedge, \vee, \neg\}$.

Элементы основного множества – классы эквивалентности формул (классы формул, представляющих одну и ту же функцию).

Теорема. Всякая представленная формулой логическая функция может быть представлена булевой формулой (т.е. как суперпозиция функций $\wedge$, $\vee$, $\neg$).

Булевых формул для каждой логической функции бесконечно много.

Эквивалентные соотношения в булевой алгебре

Аксиомы конъюнкции	Аксиомы дизъюнкции	Название аксиомы
$x \wedge y = y \wedge x$	$x \vee y = y \vee x$	коммутативность
$x \wedge (y \wedge z) = (x \wedge y) \wedge z$	$x \vee (y \vee z) = (x \vee y) \vee z$	ассоциативность
$x \wedge x = x$	$x \vee x = x$	идемпотентность
$x \wedge (y \vee z) = x \wedge y \vee x \wedge z$	$x \vee y \wedge z = (x \vee y) \wedge (x \vee z)$	дистрибутивность
$x \wedge (x \vee y) = x$	$x \vee x \wedge y = x$	поглощение
$x \wedge 1 = x$ $x \wedge 0 = 0$	$x \vee 1 = 1$ $x \vee 0 = x$	аксиомы 1 и 0
$\overline{x \wedge y} = \overline{x} \vee \overline{y}$	$\overline{x \vee y} = \overline{x} \wedge \overline{y}$	законы де Моргана
аксиома «противоречия»: $x \wedge \overline{x} = 0$	аксиома «исключенного третьего»: $x \vee \overline{x} = 1$	

Аксиомы отрицания

$$\overline{1} = 0 \qquad \overline{0} = 1 \qquad \overline{\overline{x}} = x$$

Построение булевой формулы для таблично заданной функции.

Обозначим $x^{\alpha} = \begin{cases} x, & \alpha = 1 \\ \bar{x}, & \alpha = 0 \end{cases}$ Тогда $x^{\alpha} = \begin{cases} 1, & x = \alpha \\ 0, & x \neq \alpha \end{cases}$

Элементарная коньюнкция (произведение) – конъюнкция любого числа переменных, взятых по одному разу, с отрицанием или без отрицания.

Элементарнаяе дизъюнкция – дизъюнкция любого числа переменных, взятых по одному разу, с отрицанием или без отрицания.

Конституентой единицы (K^1) данного набора ($\alpha_1, \alpha_2, \ldots, \alpha_n$) называется элементарная конъюнкция *всех* переменных, образующих этот набор: $K^1_{(\alpha_1, \alpha_2, \ldots, \alpha_n)}(x_1, x_2, \ldots, x_n) = x_1^{\alpha_1} x_2^{\alpha_2} \ldots x_n^{\alpha_n}$.

Конституентой нуля (K^0) данного набора ($\alpha_1, \alpha_2, \ldots, \alpha_n$) называется элементарная дизъюнкция всех переменных, образующих этот набор: $K^0_{(\alpha_1, \alpha_2, \ldots, \alpha_n)}(x_1, x_2, \ldots, x_n) = \overline{x_1^{\alpha_1}} \vee \overline{x_2^{\alpha_2}} \vee \ldots \vee \overline{x_n^{\alpha_n}}$.

Лемма 1. Конституента единицы равна 1 только на «своем» наборе.

▯ На произвольном фиксированном наборе $x_1 = \beta_1$, $x_2 = \beta_2$, … , $x_n = \beta_n$

$$K^1_{(\alpha_1, \alpha_2, \ldots, \alpha_n)}(\beta_1, \beta_2, \ldots, \beta_n) = \beta_1^{\alpha_1} \beta_2^{\alpha_2} \ldots \beta_n^{\alpha_n} = \begin{cases} 1, \text{ если } & \beta_1 = \alpha_1, \ldots \; \beta_n = \alpha_n \\ 0, \text{ если } & \beta_1 \neq \alpha_1, \ldots \text{ или } \; \beta_n \neq \alpha_n \end{cases} ▯$$

Лемма 2. Конституента нуля равна 0 только на «своем» наборе.

▯ На произвольном фиксированном наборе $x_1 = \beta_1$, $x_2 = \beta_2$, … , $x_n = \beta_n$

$$K^0_{(\alpha_1, \alpha_2, \ldots, \alpha_n)}(\beta_1, \beta_2, \ldots, \beta_n) = \overline{\beta_1^{\alpha_1}} \vee \overline{\beta_2^{\alpha_2}} \vee \ldots \vee \overline{\beta_n^{\alpha_n}} == \begin{cases} 0, \text{ если } & \beta_1 = \alpha_1, \ldots \; \beta_n = \alpha_n \\ 1, \text{ если } & \beta_1 \neq \alpha_1, \ldots \text{ или } \; \beta_n \neq \alpha_n \end{cases} ▯$$

***Дизъюнктивной нормальной формой* (*ДНФ*)** функции называется такая элементарная конъюнкция или дизъюнкция нескольких различных элементарных коньюнкций, что таблица истинности ДНФ совпадает с таблицей истинности самой функции.

***Конъюнктивной нормальной формой* (*КНФ*)** функции называется такая элементарная дизъюнкция или конъюнкция нескольких различных элементарных дизъюнкций, что таблица истинности ДНФ совпадает с таблицей истинности самой функции.

***Совершенная ДНФ* (*СДНФ*):** $f_{\text{СДНФ}}\left(x_1, x_2, \ldots, x_n\right) = \bigvee\limits_{f(\alpha_1,\alpha_2,\ldots,\alpha_n)=1} K^1_{(\alpha_1,\alpha_2,\ldots,\alpha_n)}$.

***Совершенная КНФ* (*СКНФ*):** $f_{\text{СКНФ}}\left(x_1, x_2, \ldots, x_n\right) = \bigwedge\limits_{f(\alpha_1,\alpha_2,\ldots,\alpha_n)=0} K^0_{(\alpha_1,\alpha_2,\ldots,\alpha_n)}$.

Формулы, получаемые перестановкой коньюнкт и дизъюнкт, считаются одной и той же СДНФ (СКНФ).

Лемма 1. Всякая переключательная функция, не равная тождественно 0, представима и притом однозначно в СДНФ.

Лемма 2. Всякая переключательная функция, не равная тождественно 1, представима и притом однозначно СКНФ.

Минимизация булевых функций

Число вхождений переменной – количество раз, которое она встречается (с отрицанием или без) в алгебраическом выражении этой функции. Каждая булева функция имеет конечное число вхождений переменных.

Задача минимизации булевых функций: получить алгебраическое выражение (формулу) булевой функции с наименьшим числом вхождений переменных.

Импликантой $g(x_1, x_2, ..., x_n)$ данной функции $f(x_1, x_2, ..., x_n)$ называется функция:

$$g(x_1, x_2, ..., x_n) = \begin{cases} 0, \text{ на наборах, где } f(x_1, x_2, ..., x_n) = 0 \\ 0 \text{ или } 1, \text{ на наборах, где } f(x_1, x_2, ..., x_n) = 1 \end{cases}$$

Лемма 1. Функция g является импликантой функции f тогда и только тогда, когда переключательная функция $g \supset f = \mathrm{const}\,1$ равна const 1.

Лемма 2. Если $g = a \vee b$ и g является импликантой функции f, то a и b также являются импликантами функции f.

Лемма 3. Конституенты единицы тех наборов, на которых данная функция равна единице, являются импликантами данной функции.

Лемма 4. Любая собственная часть конституенты единицы фиксированного набора сохраняет единицу на этом фиксированном наборе.

Простая импликанта булевой функции – элементарная конъюнкция, которая является импликантой, и никакая собственная ее часть импликантой не является.

Сокращенной ДНФ (***Сокр. ДНФ***) – дизъюнкция всех простых импликант функции. Формулы, получаемые перестановкой конъюнкт и дизъюнкт, считаются одной и той же Сокр. ДНФ.

Теорема. Всякая логическая функция представима, причём однозначно, в виде некоторой Сокр. ДНФ.

Лишними называются те ***импликанты***, удаление которых из Сокр. ДНФ функции не изменяет ее таблицы истинности.

Тупиковая ДНФ (***ТДНФ***) – дизъюнкцией простых импликант, среди которых нет лишних. Различные ТДНФ одной и той же функции $f(x_1, x_2,..., x_n)$ могут содержать разное число вхождений переменных.

Минимальная ДНФ (***МДНФ***) – ТДНФ с наименьшим числом вхождений переменных. У одной функции $f(x_1, x_2,..., x_n)$ может быть несколько различных МДНФ, но все они имеют одинаковую длину.

	Метод Квайна	**Метод Блейка**
Область применения	СДНФ	Любая ДНФ
Формула склеивания	$Ax \vee A\overline{x} = Ax \vee A\overline{x} \vee A$	$Ax \vee B\overline{x} = Ax \vee B\overline{x} \vee AB$
Формула поглощения	$AB \vee A = A$	
Количество шагов	Не превышает число переменных функции	Заранее не известно
Результат алгоритма	Сокращенная ДНФ	

Построение МДНФ из Сокр.ДНФ с помощью таблицы Квайна

Таблица Квайна для функции f:

1) заголовки строк – все простые импликанты функции f,
2) заголовки столбцов – все конституенты единиц функции f,
3) в таблице ставится *, если простая импликанта равна 1 на том наборе, на котором конституента равна единице.

Алгоритм минимизации

Шаг 1. Для всех столбцов, содержащих ровно одну *, выписать в $f_{\text{МДНФ}}$ (через дизъюнкцию) соответствующие простые импликанты.

Шаг 2. Удалить из таблицы все столбцы, в которых на пересечении со строками, выбранными на шаге 1, стоит *.

Шаг 3. Удалить из таблицы все строки, соответствующие выбранными на шаге 1 простым импликантам, а также все строки, в которых нет ни одной *.

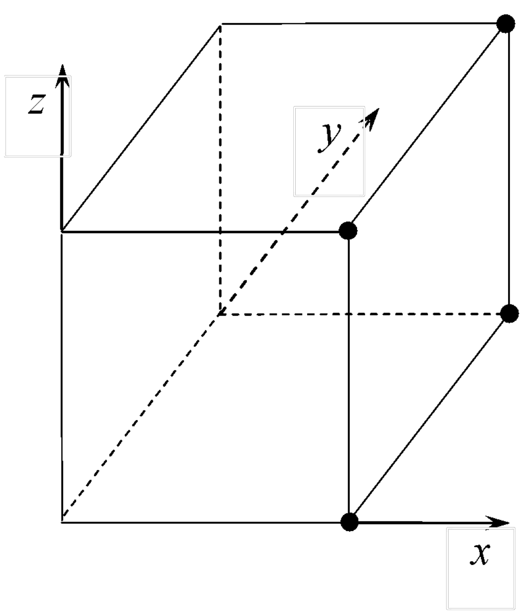
Шаг 4. Повторять шаги 1–3 до тех пор, пока таблица изменяется.

Шаг 5. Из оставшихся строк перебором всех возможных вариантов выбрать минимальное по суммарному числу вхождений переменных множество простых импликант так, чтобы все оставшиеся столбцы имели на пересечении хотя бы с одной из выбранных строк *.

Графическая минимизация логических функций

Пусть функция $f(x_1, x_2, ..., x_n)$ изображена на n-мерном кубе (вершины куба образуют область определения).

Если необходимо задать m-грань (m-куб) n-мерного булевого куба, то необходимо ровно (n–m) равенств вида $x_i = 0$ или $x_i = 1$, где $i = \overline{1, n}$. Если же записывать по отдельности все вершины этой m-грани, то понадобится nm таких равенств. Именно это наблюдение лежит в основе графической минимизации.



Пример. $f(x, y, z) = (0000\,1111)$.

1) $x=1, y=0, z=0$ или $x=1, y=0, z=1$ или
$x=1, y=1, z=0$ или $x=1, y=1, z=1$;

2а) $x=1, y=1$ или $x=1, y=0$;

2б) $x=1, z=1$ или $x=1, z=0$;

3) $x=1$.

Метод карт Карнапа♦

Картой Карнапа для 4-х переменных называется таблица 4×4, в которой

– каждая клетка соответствует вершине четырехмерного куба,

– если склеить карту по вертикали и горизонтали в тор, то соседние клетки карты будут соответствовать соседним вершинам куба.

Соответствие вершинам куба задают явно, указывая координаты строк и столбцов таблицы.

$*$ ставится в клетки, которым соответствует набор с единичным значением функции.

В качестве карты Карнапа для функции трёх переменных берут часть карты для функции четырёх переменных.

Будем называть ***гранью*** множество звездочек карты, соответствующее какой-либо грани куба (отметим, что грань на карте Карнапа – всегда множество соседних звездочек). Мощность грани (количество *) может быть равна 1, 2, 2^2, ..., 2^n, где n – размерность функции.

Максимальной гранью данной $*$ карты будем называть максимальную по количеству $*$ грань, содержащую данную $*$. ***Полезной мощностью*** грани будем называть количество ещё непокрытых звездочек, покрываемых данной гранью.

Полезная мощность может быть равна 0, 1, 2, 3, ..., m, где m – мощность данной грани.

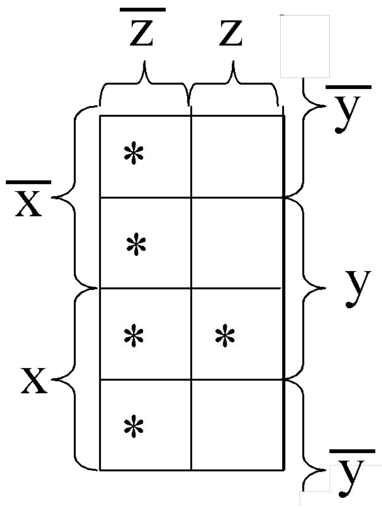
$\boxed{!!!}$ Полезная мощность определена только для максимальной грани.

$\boxed{!!!}$ Независимо от шага алгоритма мощность максимальной грани для каждой звездочки одинакова (она зависит только от самой функции), а полезная мощность максимальной грани изменяется во время работы алгоритма, а по его завершению для всех звездочек равна 0.

♦ Карнап (Carnap) Рудольф (1891-1970 гг.) – нем.-амер. философ и логик

Пример 1

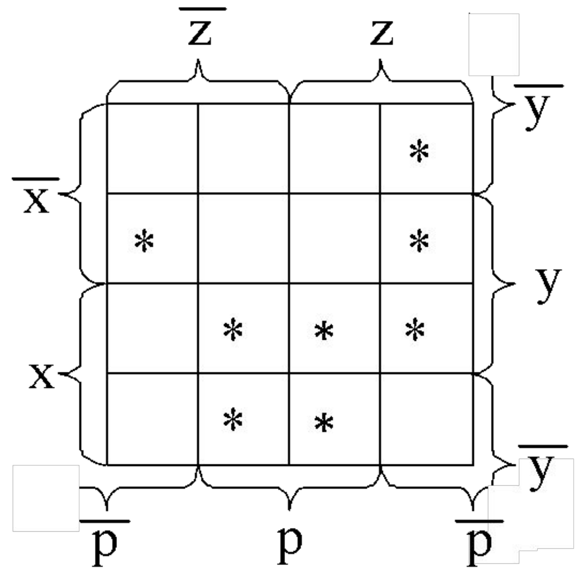
$$f(x,y,z) = (1010\ \ 1011)$$



x,y,z	f
0 0 0	1
0 0 1	1
0 1 0	1
0 1 1	1
1 0 0	1
1 0 1	1
1 1 0	1
1 1 1	1
1 1 0 1	1

Пример 2

$$f(x,y,z,p) = (0010\ \ 1010\ \ 0101\ \ 0111)$$



x,y,z,p	f
0 0 0 0	0
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	0
1 0 0 1	1
1 0 1 0	0
1 0 1 1	1
1 1 0 0	0
1 1 0 1	1
1 1 1 0	1
1 1 1 1	1

Алгоритм минимизации по карте Карнапа

Шаг 1. Найти непокрытую, немаркированную звездочку. Если все звездочки покрыты, то перейти на шаг 7. Если все непокрытые звездочки маркированы, то перейти на шаг 5.

Шаг 2. Среди всех возможных вариантов граней для данной звездочки выбрать грани, максимальные по мощности. Если такая грань единственная, то обвести ее и перейти на шаг 6.

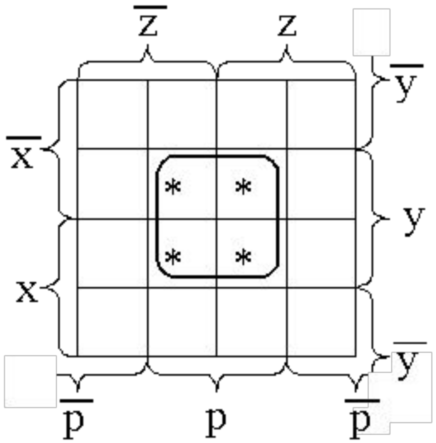
Шаг 3. Среди граней максимальной мощности выбрать грань с максимальной полезной мощностью. Если такая грань единственная, то обвести ее и перейти на шаг 6.

Шаг 4. Маркировать звёздочку и перейти на шаг 1.

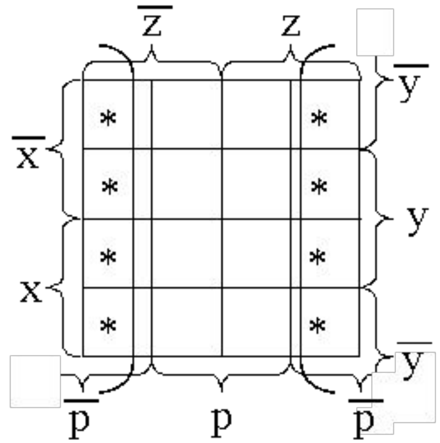
Шаг 5. Для произвольной непокрытой маркированной звездочки обвести любую из граней с максимальной полезной мощностью (отличной от нуля, так как сама звездочка еще не покрыта).

Шаг 6. Снять все маркировки. Перейти на шаг 1.

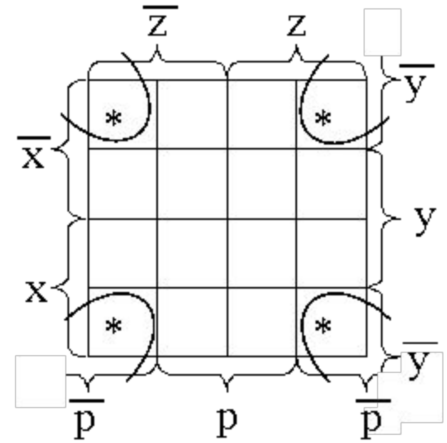
Шаг 7. Проверить, нет ли граней, полностью покрытых другими гранями, т.е. лишних импликант.



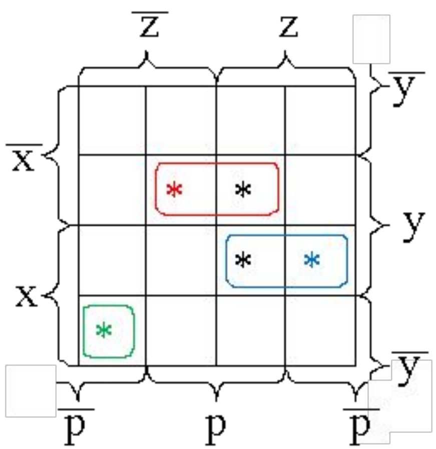
$y\ p$



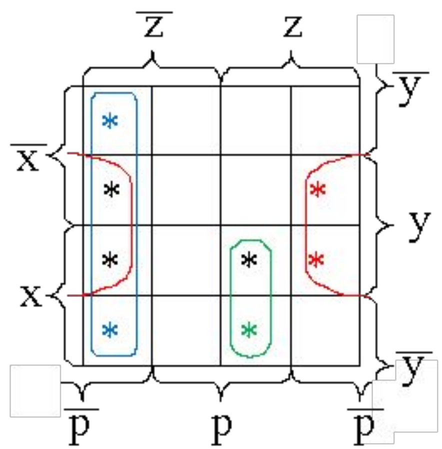
$\bar p$



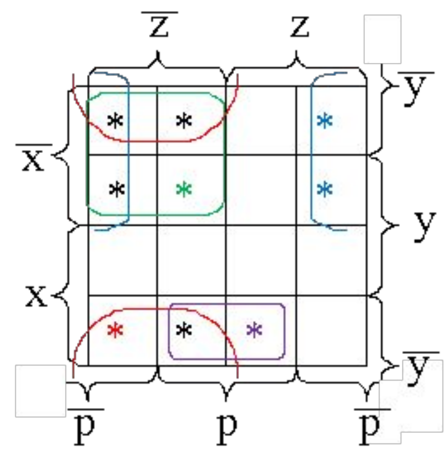
$\bar y\ \bar p$



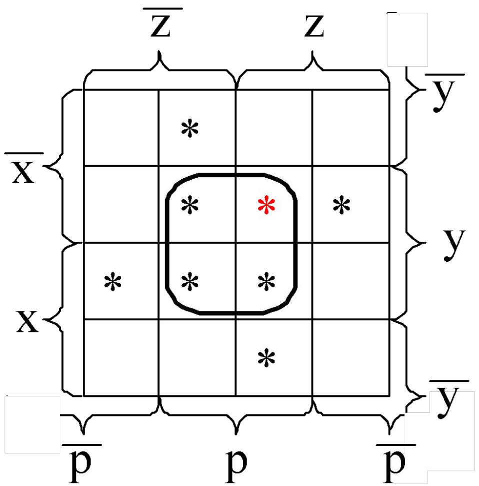
$x\ \bar y\ \bar z\ \bar p \vee \bar x\ y\ p \vee x\ y\ z$



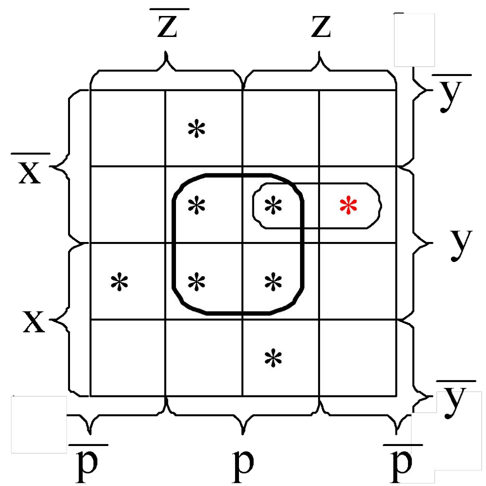
$\bar z\ \bar p \vee y\ \bar p \vee x\ z\ p$



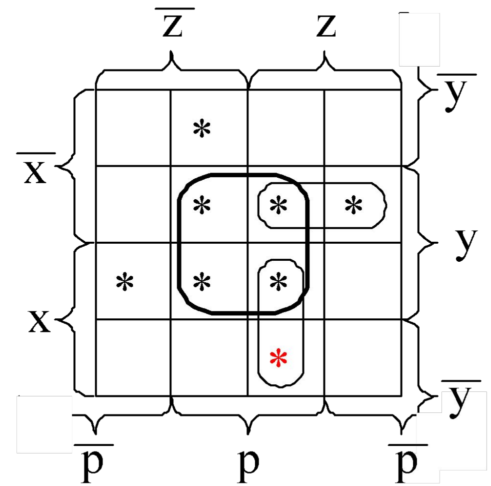
$\bar x\ \bar z \vee \bar x\ \bar p \vee \bar y\ \bar z \vee x\ \bar y\ p$



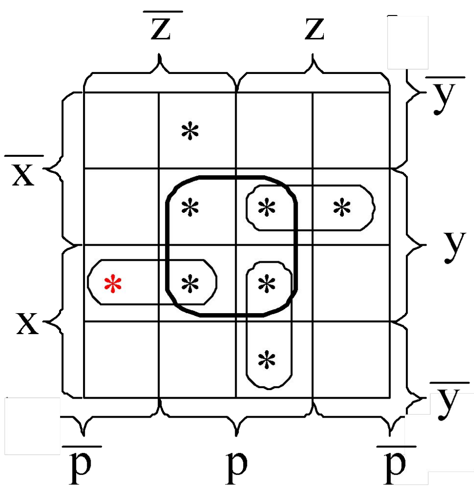
$\bar{z}$ z
$\bar{y}$ y $\bar{y}$
$\bar{x}$ x
$\bar{p}$ p $\bar{p}$



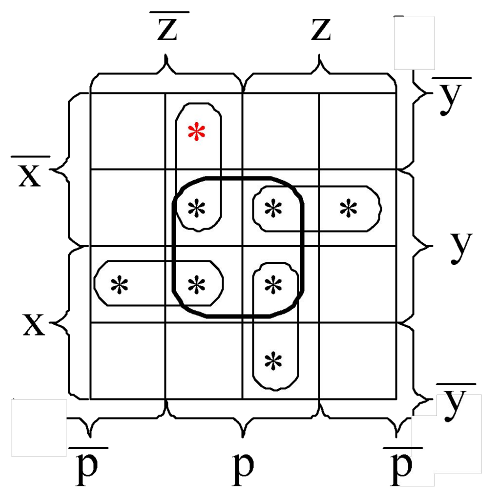
$\bar{z}$ z
$\bar{y}$ y $\bar{y}$
$\bar{x}$ x
$\bar{p}$ p $\bar{p}$



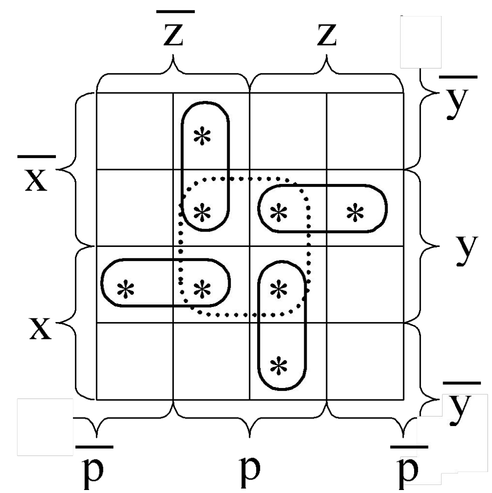
$\bar{z}$ z
$\bar{y}$ y $\bar{y}$
$\bar{x}$ x
$\bar{p}$ p $\bar{p}$



$\bar{z}$ z
$\bar{y}$ y $\bar{y}$
$\bar{x}$ x
$\bar{p}$ p $\bar{p}$



$\bar{z}$ z
$\bar{y}$ y $\bar{y}$
$\bar{x}$ x
$\bar{p}$ p $\bar{p}$



$\bar{z}$ z
$\bar{y}$ y $\bar{y}$
$\bar{x}$ x
$\bar{p}$ p $\bar{p}$

Логическая функция $f^*(x_1,\ldots,x_n)$ называется ***двойственной*** к $f(x_1,\ldots,x_n)$, если для всех наборов значений переменных $f^*(x_1,\ldots,x_n)=\overline{f}(\overline{x}_1,\ldots,\overline{x}_n)$.

Функция $f(x_1,\ldots,x_n)$ называется ***самодвойственной***, если для всех наборов значений переменных $f^*(x_1,\ldots,x_n)=f(x_1,\ldots,x_n)$, или $f(x_1,\ldots,x_n)=\overline{f}(\overline{x}_1,\ldots,\overline{x}_n)$.

Логическая функция называется ***монотонной***, если для всех сравнимых наборов

$$(\alpha_1,\ldots,\alpha_n)\le(\beta_1,\ldots,\beta_n) \;\Rightarrow\; f(\alpha_1,\ldots,\alpha_n)\le f(\beta_1,\ldots,\beta_n).$$

$(\alpha_1,\ldots,\alpha_n)\le(\beta_1,\ldots,\beta_n) \Leftrightarrow \forall i\ \alpha_i\le\beta_i$.

Алгебра Жегалкина $\langle \{0,1\};\{\wedge,\oplus\}\rangle$.

Теорема. Всякая булева функция представима ***полиномом Жегалкина***, т.е. в виде

$$f(x_1,\ldots,x_n) = a_0 \oplus a_1x_1 \oplus a_2x_2 \oplus \ldots \oplus a_nx_n \oplus$$

$$\oplus a_{n+1}x_1x_2 \oplus \ldots \oplus a_{2n-1}x_1x_n \oplus a_{2n}x_2x_3 \oplus \ldots \oplus a_kx_1x_2\ldots x_n \text{, где } a_i \in \{0,1\}$$

$x\vee y=x\oplus y\oplus xy$, $x(y\oplus z)=xy\oplus xz$, $x\oplus 0=x$,

$\bar{x}=x\oplus 1$, $x\oplus \bar{x}=1$, $x\oplus x=0.$

Лемма. В СДНФ при построении полинома Жегалкина можно все '$\vee$' заменить на '$\oplus$'

Логическая функция называется ***линейной***, если она представима *линейным* полиномом Жегалкина, т.е. в виде

$$f(x_1,\ldots,x_n) = a_0 \oplus a_1x_1 \oplus a_2x_2 \oplus \ldots \oplus a_nx_n \text{, где } a_i \in \{0,1\}.$$

Теорема. Если функция принимает значение 1 на нечетном количестве наборов, то она нелинейна.

Классы Поста:

$\mathbf{P_0}$ – функции, сохраняющие 0: $f(0,0,...,0)=0$.
$\mathbf{P_1}$ – функции, сохраняющие 1: $f(1,1,...,1)=1$.
S – самодвойственные функций.
M – монотонные функций.
L – линейные функций.

Система логических функций $F=\left(f_1,f_2,...,f_n\right)$ называется ***функционально полной***, если всякая логическая функция является некоторой суперпозицией этих функций.

Лемма. Каждый класс Поста замкнут относительно операций подстановки и суперпозиции, т. е. с помощью этих операций можно получить только функции того же класса.

Теорема Поста (сильная). Система логических функций тогда и только тогда является функционально полной, когда для каждого класса $\mathbf{P_0}$, $\mathbf{P_1}$, **S**, **M**, **L** в ней найдется функция, не принадлежащая этому классу.

Теорема Поста (слабая). Система логических функций, содержащая **const 0** и **const 1**, является функционально полной тогда и только тогда, когда она содержит хотя бы одну нелинейную и хотя бы одну немонотонную функцию.

Система логических функций называется ***базисом***, если она функционально полная, а удаление любой функции из этой системы делает ее неполной.

Теорема. Каждый базис содержит не более 4 логических функций.