

АЛГЕБРА ЛОГИКИ

- Основными понятиями алгебры логики являются понятие логической переменной и логической функции.
- Логической переменной (аргументом) называется величина которая может принимать одно из двух значений («0» или «1»)
- Логической функцией называется функция двоичных переменных которая также может принимать одно из двух возможных состояний («0» или «1»)

Множество состояний (значений), которые могут принимать как аргументы так и функции, равно двум.

Для этих состояний в алгебре логики определяются отношение эквивалентности обозначаемое (=) и три операции:

логического сложения (дизъюнкции)

обозначаемое- + или \vee

логического умножения (конъюнкции)

обозначаемое- \bullet или \wedge или $\&$

логического отрицания (инверсии)

обозначаемое- $\bar{\quad}$

При выполнении выше перечисленных операций отношения эквивалентности имеют вид:

а) дизъюнкция

$$0+0=0$$

$$0+1=1$$

$$1+0=1$$

$$1+1=1$$

б) конъюнкция

$$0 \bullet 0=0$$

$$0 \bullet 1=0$$

$$1 \bullet 0=0$$

$$1 \bullet 1=1$$

в) инверсия

$$\overline{0}=1$$

$$\overline{1}=0$$

Законы алгебры логики:

1. Переместительный закон

для логического сложения: $x_1 + x_2 = x_2 + x_1$,

для логического умножения: $x_1 \cdot x_2 = x_2 \cdot x_1$.

2. Сочетательный закон

для логического сложения: $(x_1 + x_2) + x_3 = x_1 + (x_2 + x_3)$,

для логического умножения: $(x_1 \cdot x_2) \cdot x_3 = x_1(x_2 \cdot x_3)$.

3. Распределительные законы

первый распределительный закон: $x_1(x_2 + x_3) = x_1 \cdot x_2 + x_1 \cdot x_3$, второй

распределительный закон: $x_1 + x_2 \cdot x_3 = (x_1 + x_2)(x_1 + x_3)$.

4. Законы отрицания (правило де Моргана)

для дизъюнкции: $x1 + x2 = x1 \cdot x2$

для конъюнкции: $x1 \cdot x2 = x1 + x2$

двойное отрицание: $x = x$

5. Законы поглощения

дизъюнктивное поглощение: $x1 + x1 \cdot x2 = x1 \cdot (1 + x2) = x1$

конъюнктивное поглощение: $x1 \cdot (x1 + x2) = x1 \cdot x1 + x1 \cdot x2 = x1 + x1 \cdot x2 = x2$

6. Законы склеивания

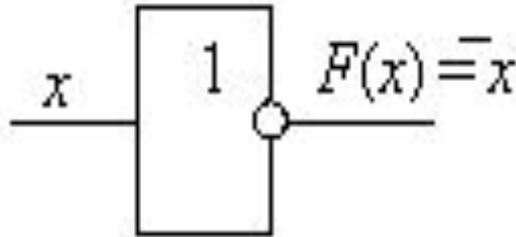
дизъюнктивное склеивание: $x1 \cdot x2 + x1 \cdot x2 = x1 \cdot (x2 + x2) = x1$

конъюнктивное склеивание: $(x1 + x2)(x1 + x2) = x1 + x1 \cdot x2 + x1 \cdot x2 + x2 = x1$

Логическая связь НЕ (логическое отрицание)

Отрицанием высказывания x называют сложное высказывание $F(x)$, которое истинно, когда x ложно и ложно, когда x истинно.

$$F(x) = \bar{x}$$

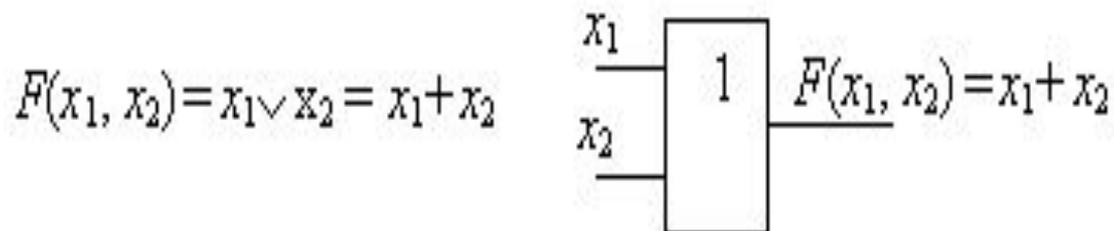


x	$F(x) = \bar{x}$
0	1
1	0

Аналитическая функция, УГО и таблица истинности элемента НЕ

Логическая связь ИЛИ – сложение (дизъюнкция) высказываний

Дизъюнкцией двух высказываний x_1 и x_2 называется сложное высказывание $F(x_1, x_2)$, которое ложно только в одном случае, когда x_1 и x_2 одновременно ложны ($x_1 = 0$ и $x_2 = 0$). Во всех остальных случаях высказывание $F(x_1, x_2)$ истинно.



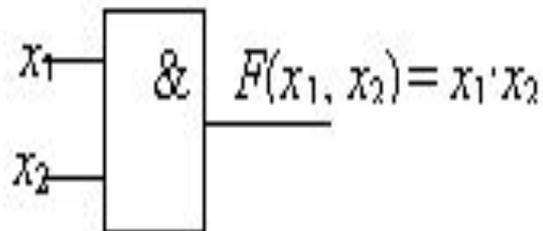
x_1	x_2	$F(x_1, x_2)$
0	0	0
0	1	1
1	0	1
1	1	1

Аналитическая функция, УГО и таблица истинности элемента ИЛИ

Логическая связь И (конъюнкция высказываний)

Конъюнкцией высказываний x_1 и x_2 , называется сложное высказывание $F(x_1, x_2)$, которое истинно только в одном случае, когда x_1 и x_2 одновременно истинны ($x_1 = 1$ и $x_2 = 1$). Во всех остальных случаях высказывание $F(x_1, x_2)$ ложно.

$$F(x_1, x_2) = x_1 \wedge x_2 = x_1' x_2$$



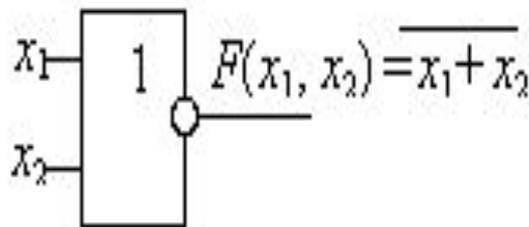
x_1	x_2	$F(x_1, x_2)$
0	0	0
0	1	0
1	0	0
1	1	1

Аналитическая функция, УГО и таблица истинности элемента И

Логическая связь отрицание дизъюнкции (операция Пирса)

Логической связью *отрицание дизъюнкции* высказываний x_1 и x_2 , называется сложное высказывание $F(x_1, x_2)$, которое истинно только в том случае, когда x_1 и x_2 одновременно ложны ($x_1 = 0$ и $x_2 = 0$). Во всех остальных случаях высказывание $F(x_1, x_2)$ ложно.

$$F(x_1, x_2) = x_1 \downarrow x_2 = \overline{x_1 + x_2}$$



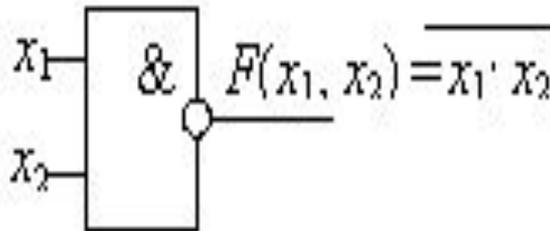
x_1	x_2	$F(x_1, x_2)$
0	0	1
0	1	0
1	0	0
1	1	0

Аналитическая функция, УГО и таблица истинности элемента ИЛИ-НЕ

Логическая связь отрицание конъюнкции (операция Шеффера)

Логической связью *отрицание конъюнкции* высказываний x_1 и x_2 , называется сложное высказывание $F(x_1, x_2)$, которое ложно только в том случае, когда x_1 и x_2 одновременно истинны ($x_1 = 1$ и $x_2 = 1$). Во всех остальных случаях высказывание $F(x_1, x_2)$ истинно.

$$F(x_1, x_2) = x_1 \uparrow x_2 = \overline{x_1 \cdot x_2}$$

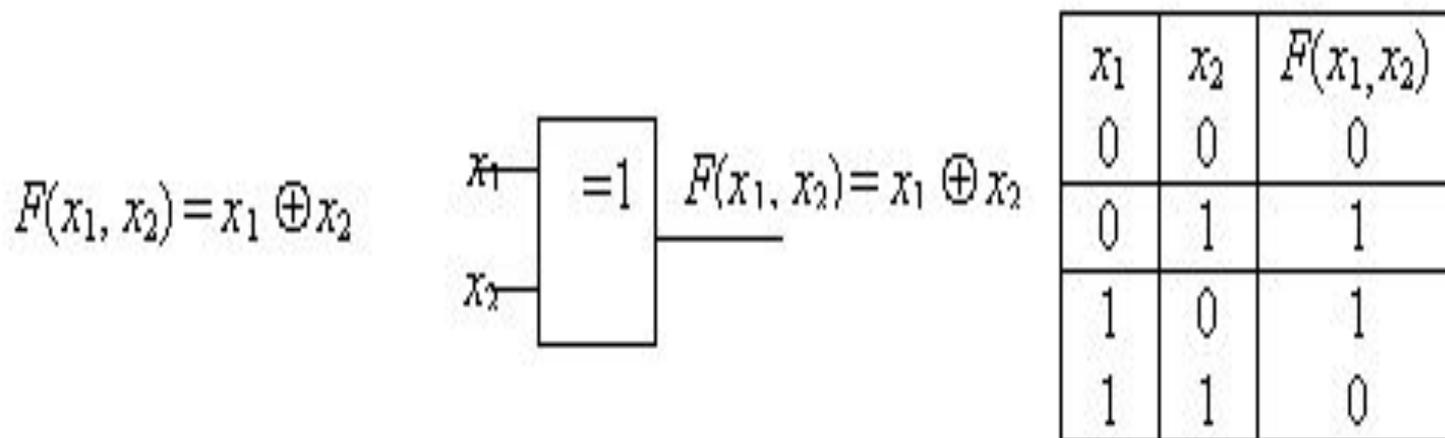


x_1	x_2	$F(x_1, x_2)$
0	0	1
0	1	1
1	0	1
1	1	0

Аналитическая функция, УГО и таблица истинности элемента И-НЕ

Логическая связь отрицание равнозначности (операция ИЛИ-ИЛИ)

Логической связью *отрицание равнозначности* высказываний x_1 и x_2 , называется сложное высказывание $F(x_1, x_2)$, которое истинно тогда, когда значения истинности высказываний x_1 и x_2 не совпадают и ложно, когда значения истинности высказываний x_1 и x_2 совпадают.



Аналитическая функция, УГО и таблица истинности элемента
исключающее ИЛИ

СИНТЕЗ ЛОГИЧЕСКИХ СХЕМ

- **Сущность задач синтеза заключается в том, чтобы спроектировать логическое устройство**

- **Основные этапы синтеза логических схем**

На *первом этапе* на основании заданного алгоритма работы синтезируемого устройства формируют словесное описание логической функции.

Пример: Считать, что цель обнаружена (выходной сигнал равен 1) только в тех случаях, когда не менее чем два входных сигнала равны "1". Во всех остальных случаях цель не обнаружена (выходной сигнал равен 0).

На **втором этапе** на основе словесной формы логической функции составляют табличную форму. Затем по таблице истинности находят аналитическую форму представления логической функции в виде СДНФ или СКНФ

Пример:

x_1	x_2	x_3	$F(x_1, x_2, x_3)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

СДНФ примет вид : $F(x_1, x_2, x_3) = \bar{x}_1 \bar{x}_2 x_3 + x_1 \bar{x}_2 x_3 + x_1 x_2 \bar{x}_3 + x_1 x_2 x_3$.

СКНФ примет вид :

$$F(x_1, x_2, x_3) = (x_1 + x_2 + x_3)(x_1 + x_2 + \bar{x}_3)(x_1 + \bar{x}_2 + x_3)(\bar{x}_1 + x_2 + x_3).$$

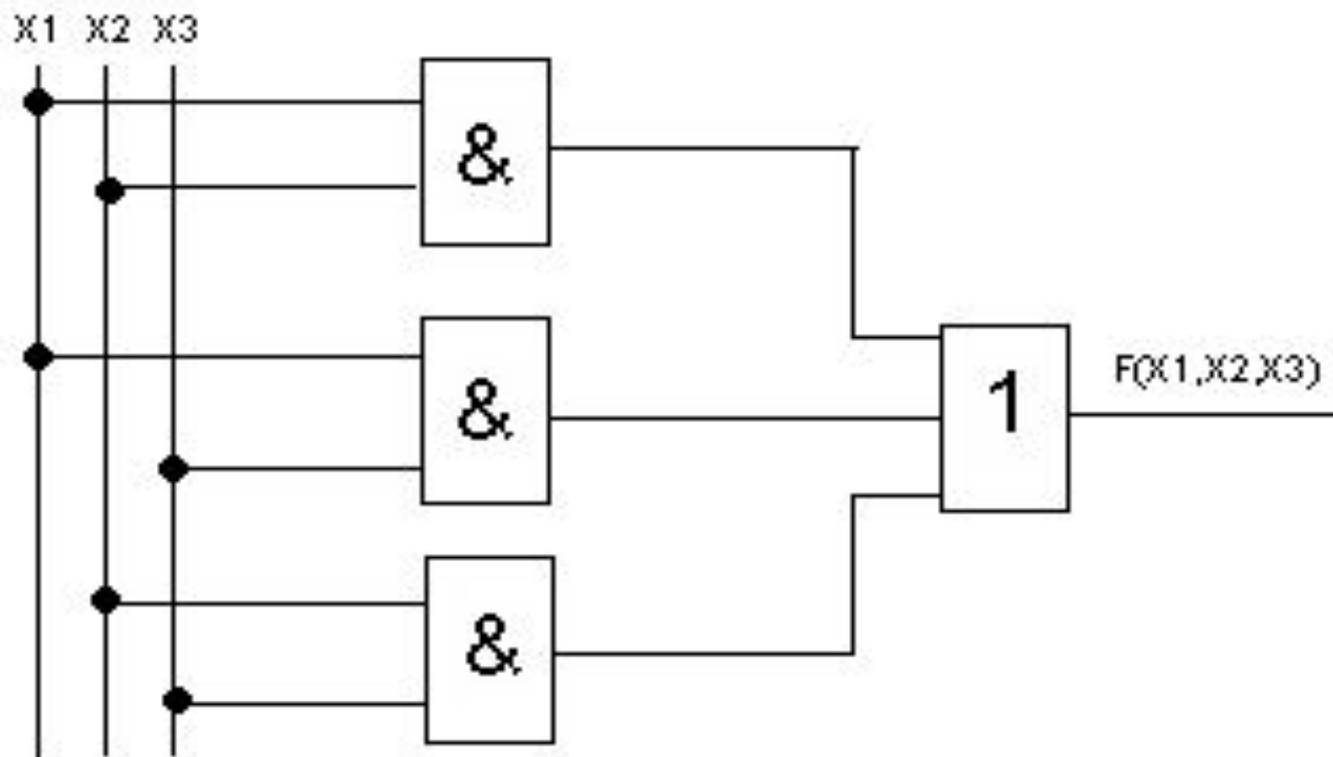
Третий этап. На этом этапе осуществляется минимизация СДНФ или СКНФ с целью уменьшения числа логических элементов, необходимых для разработки функциональной схемы синтезируемого устройства. Для минимизации логических функций используются как аналитические, так и графические методы.

Сущность **аналитических методов** сводится к последовательному применению законов алгебры логики и их следствий

$$F(X1, X2, X3) = X2X3(X1 + X1) + X1X3(X2 + X2) + X1X2(X3 + X3) \Rightarrow$$

$$F(X1, X2, X3) = X2X3 + X1X3 + X1X2$$

На **четвертом этапе** на основе минимизированной логической функции разрабатывается функциональная схема синтезируемого устройства и проверяется правильность ее работы.



Куб Карно

Куб Карно́ — графический способ минимизации переключательных (булевых) функций, обеспечивающий относительную простоту работы с большими выражениями и устранение потенциальных гонок. Представляет собой операции попарного неполного склеивания и элементарного поглощения. Карты Карно рассматриваются как перестроенная соответствующим образом таблица истинности функции.

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

$X_3 X_4$	00	01	11	10	
$X_1 X_2$	00	1	0	0	1
	01	1	0	0	1
	10	0	1	1	0
	11	1	0	0	1

Принципы минимизации

Основным методом минимизации логических функций, представленных в виде СДНФ или СКНФ, является операция попарного неполного склеивания и элементарного поглощения. Операция попарного склеивания осуществляется между двумя термами (членами), содержащими одинаковые переменные, вхождения которых (прямые и инверсные) совпадают для всех переменных, кроме одной. В этом случае все переменные, кроме одной, можно вынести за скобки, а оставшиеся в скобках прямое и инверсное вхождение одной переменной подвергнуть склейке. Например:

$$\bar{X}_1 X_2 X_3 X_4 \vee \bar{X}_1 X_2 \bar{X}_3 X_4 = \bar{X}_1 X_2 X_4 (X_3 \vee \bar{X}_3) = \bar{X}_1 X_2 X_4 \cdot 1 = \bar{X}_1 X_2 X_4.$$

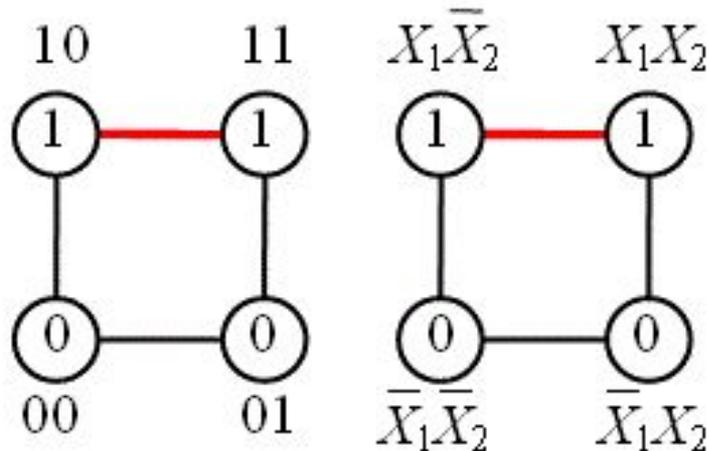
Аналогично для КНФ:

$$(\bar{X}_1 \vee X_2 \vee X_3 \vee X_4)(\bar{X}_1 \vee X_2 \vee \bar{X}_3 \vee X_4) = \bar{X}_1 \vee X_2 \vee X_4 \vee X_3 \bar{X}_3 = \bar{X}_1 \vee X_2 \vee X_4 \vee 0 = \bar{X}_1 \vee X_2 \vee X_4.$$

Возможность поглощения следует из очевидных равенств

$$A \vee \bar{A} = 1; A \bar{A} = 0.$$

X_1	X_2	$f(X_1, X_2)$
0	0	0
0	1	0
1	0	1
1	1	1

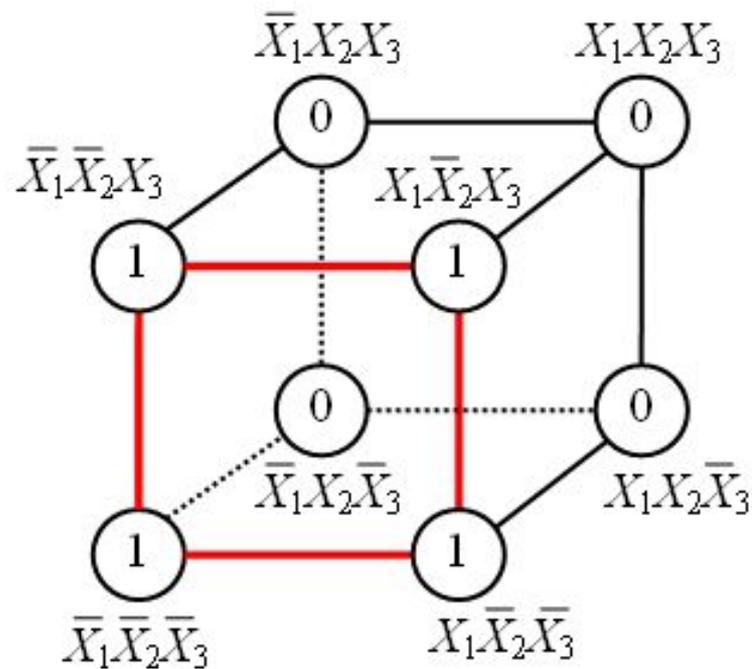


		X_2	
		0	1
X_1	1	1	1
	0	0	0

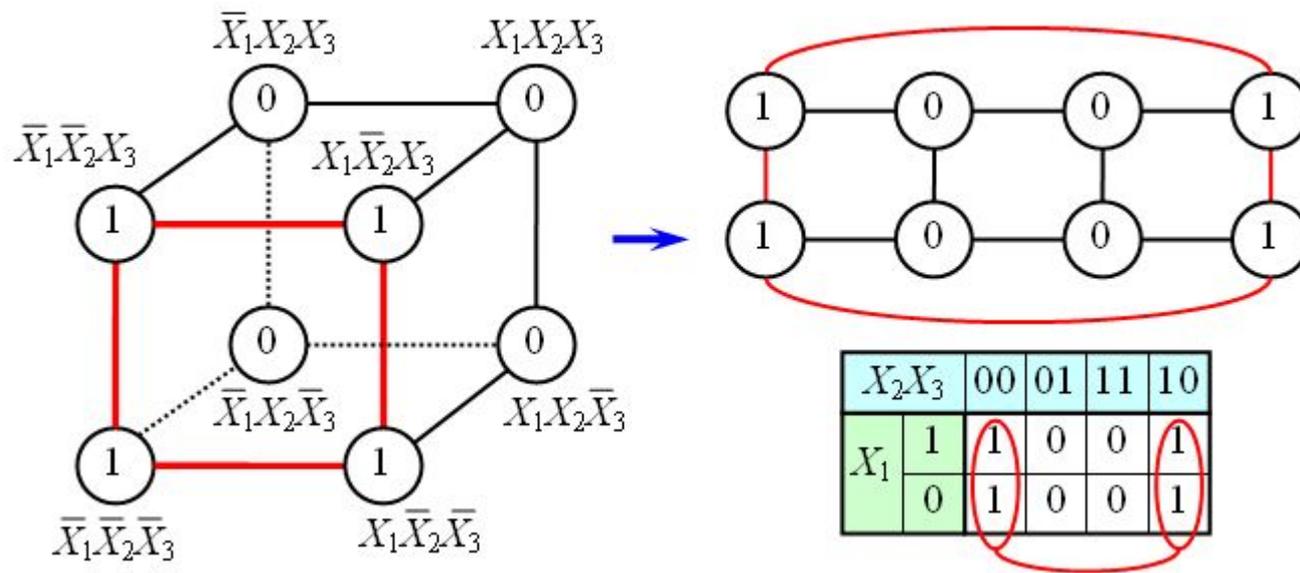
На рисунке изображена простая таблица истинности для функции из двух переменных, соответствующий этой таблице 2-мерный куб (квадрат), а также 2-мерный куб с обозначением членов СДНФ и эквивалентная таблица для группировки термов

В случае функции трёх переменных приходится иметь дело с трёхмерным кубом. Это сложнее и менее наглядно, но технически возможно. На рисунке в качестве примера показана таблица истинности для булевой функции трёх переменных и соответствующий ей куб.

X_1	X_2	X_3	$f(X_1, X_2, X_3)$
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0



Для упрощения работы с булевыми функциями большого числа переменных был предложен следующий удобный приём. Куб, представляющий собой структуру термов, разворачивается на плоскость как показано на рисунке. Таким образом появляется возможность представлять булевы функции с числом переменных больше двух в виде плоской таблицы. При этом следует помнить, что порядок кодов термов в таблице (00 01 11 10) не соответствует порядку следования двоичных чисел, а клетки, находящиеся в крайних столбцах таблицы, соседствуют между собой.



Синтез логической схемы

Синтезом логической схемы называется процедура получения логической схемы реализующую заданную логическую функцию. При синтезе логических схем используются логические элементы.

Набор логических функций, с помощью которого можно представить любую переключательную функцию называется функционально полной системой или базисом. Обычно функционально полными системами являются базисы 1) И, ИЛИ, НЕ, 2) И, НЕ, 3) ИЛИ, НЕ, 4) И-ИЛИ-НЕ, 5) И-НЕ, 6) ИЛИ-НЕ.

Основным является базис И, ИЛИ, НЕ так как с помощью него можно записать любую сложную функцию в виде СДНФ и СКНФ.

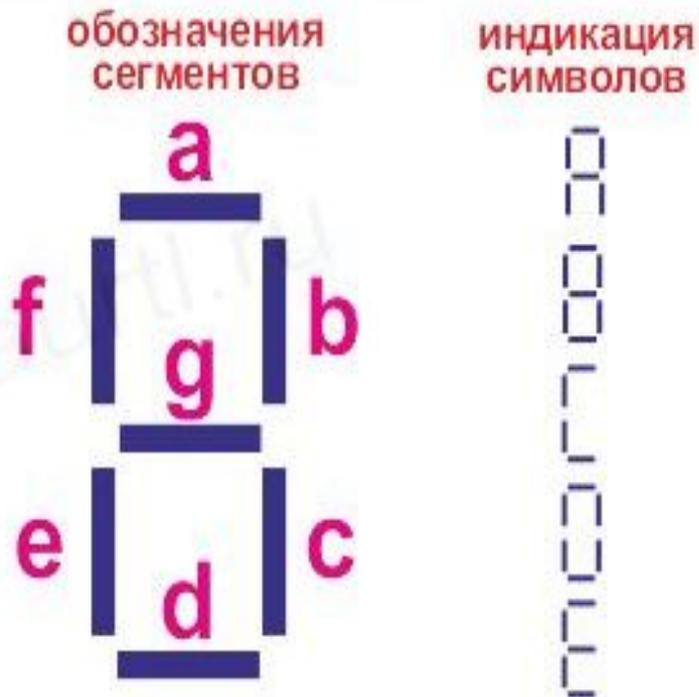
При синтезе комбинационной схемы с несколькими выходами, каждый выход рассматривается как функция входных значений. Полученные функции минимизируются, и затем строится логическая схема в одном из заданных базисов.

При синтезе цифровых автоматов, в таблицу функционирования кроме выходных значений необходимо указывать и внутреннее состояние устройства.

Пример синтеза комбинационной схемы.

Синтезировать комбинационную схему устройства управления семисегментным индикатором в базисе И, ИЛИ, НЕ. Устройство работает согласно таблицы

A	B	C	СИМВОЛ
0	0	0	A
0	0	1	8
0	1	0	C
0	1	1	0
1	0	0	E
1	0	1	любой
1	1	0	любой
1	1	1	любой





Составляем таблицу функционирования устройства, используя символ индикации и обозначение сегментов. При этом получаем таблицу зависимости восьми функций a, b, c, d, e, f, g от переменных (значений входных сигналов). Затем используя **карту Карно**, получаем минимальные дизъюнктивные форму функции. Неопределенные значения при этом заменяем на «0» или «1» в зависимости от эффективности образования групп и, следовательно, минимизации.

переменные

функции

A	B	C	символ	a	b	c	d	e	f	g
0	0	0	A	1	1	1	0	1	1	1
0	0	1	8	1	1	1	1	1	1	1
0	1	0	C	1	0	0	1	1	1	0
0	1	1	0	1	1	1	1	1	1	0
1	0	0	E	1	0	0	1	1	1	1
1	0	1	любой	x	x	x	x	x	x	x
1	1	0	любой	x	x	x	x	x	x	x
1	1	1	любой	x	x	x	x	x	x	x

1- включенные,
0 - выключенные сегменты
для индикации соответствующих
символов

неопределенные значения
заполняем X-ами

a=e=f

AB	00	01	11	10
0	1	1	x1	1
1	1	1	x1	x1

$$a=e=f=1$$

b=c

AB	00	01	11	10
0	1	0	x0	0
1	1	1	x1	x1

$$1) \begin{array}{r} ABC \\ 001 \\ 011 \\ 111 \\ 101 \\ \hline C \end{array} \quad 2) \begin{array}{r} ABC \\ 000 \\ 001 \\ \hline \overline{AB} \end{array}$$

$$b=c=C+\overline{AB}$$

d

AB	00	01	11	10
0	0	1	x1	1
1	1	1	x1	x1

$$1) \begin{array}{r} ABC \\ 001 \\ 011 \\ 111 \\ 101 \\ \hline C \end{array} \quad 2) \begin{array}{r} ABC \\ 010 \\ 011 \\ 110 \\ 111 \\ \hline B \end{array}$$

$$3) \begin{array}{r} ABC \\ 110 \\ 111 \\ 100 \\ 101 \\ \hline A \end{array}$$

$$d=A+B+C$$

g

AB	00	01	11	10
0	1	0	x0	1
1	1	0	x0	x1

$$1) \begin{array}{r} ABC \\ 000 \\ 001 \\ 100 \\ 101 \\ \hline \overline{B} \end{array}$$

$$g=\overline{B}$$



Используя полученные функции синтезируем логическую схему устройства в базисе И, ИЛИ, НЕ. Синтез выполняется в направлении от значений входных сигналов до получения выходных значений. Первоначально из входных переменных получаем необходимые для функций инверсные значения. Затем, поэтапно применяя необходимые элементы для реализации операций, получаем конечные значения функций. Источник логической единиц условно обозначаем точкой (1). При положительной логике ей соответствует напряжение питания всей схемы.

