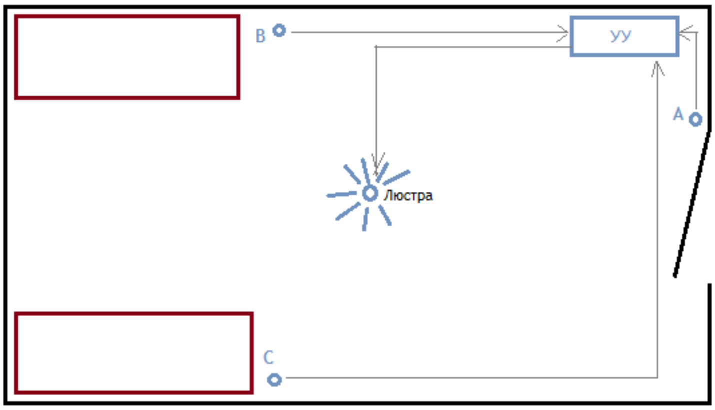
Введение в логику. Свойства логических операций
Элементы схемотехники. Логические элементы. Построение логических схем

Задача про братьев-школьников

Рассмотрим принцип действия устройства, изготовленного двумя братьями-школьниками для включения и выключения электрической люстры в своей комнате.

Люстра подключается к данному устройству, а оно подсоединяется к трем обычным выключателям.

Один из них (А) – установлен у двери, два других (В и С) в изголовье кроватей каждого из братьев. Изобразим комнату и расположение выключателей на рисунке



Любой из братьев, войдя в комнату, может зажечь свет выключателем А, а приготовившись ко сну, может потушить свет в комнате используя выключатели В или С.

Если теперь кто-то войдет в комнату, он сможет вновь включить свет выключателем А. А потушить – любым из трех выключателей.

Чтобы разобраться, по какому принципу построена данная конструкция, построим математическую модель работы «устройства управления люстрой» в целом.

Начнем с самых общих описаний.

Наша система содержит выключатели А, В и С, сигналы от которых обрабатываются устройством управления (УУ), которое анализирует входные сигналы и «принимает решение» - зажигать или гасить люстру.

С точки зрения математики в УУ должна быть реализована функция от трех переменных, в которой каждая упорядоченная тройка значений А, В и С соответствует единственному значению F(A, B, C).

Условимся считать А = 1, если выключатель А находится в положении **ВКЛ** и А = 0, если в положении **ВЫКЛ.** Аналогично для выключателей В и С.

Сама функция F(A, B, C) также принимает только два значения – 1 или 0 (свет в комнате горит или не горит).

Создадим математическую модель нашей системы управления люстрой – построим таблицу состояний, в которой для каждой тройки значений (А, В, С) укажем значение функции F.

На основе наблюдений за системой получим:

A	B	C	F(A, B, C)	**Пояснение**
0	0	0	**0**	*Ни один выключатель не включен, свет в комнате не горит*
1	0	0	**1**	*Школьник входит в комнату и выключателем A (у входа) включает свет*
1	1	0	**0**	*Ложась спать, выключает свет выключателем у своей кровати B*
0	1	0	**1**	*В комнату входит второй школьник и щелкает выключателем A – свет вкл.*
0	1	1	**0**	*Ложась спать этой школьник гасит свет своим выключателем C*
0	0	1	**1**	*Первый школьник проснулся и вкл.свет свои выключателем B*
1	0	1	**0**	*Выходя из комнаты щелкает выключателем A и гасит свет*
1	1	1	**1**	*Если в предыдущей ситуации нажать B, то свет погаснет*

Никаких других комбинаций из трех значений независимых переменных (A, B, C) нет. Общее количество можно подсчитать по формуле $N = 2^n$ (где n = 3)

Итак, функция F полностью определена.

Определение. Функции, которые, как и их аргументы, принимают значения только в области из двух элементов, называют булевыми (или логическими) функциями.

Булевы функции находят широкое применение при математическом описании многих процессов, с их помощью исчерпывающим образом описывается действие многих автоматических устройств.

Найдем аналитическое выражение для функции F, воспользовавшись следующим алгоритмом:

1. В таблице истинности отмечаем наборы переменных (строки), на которых значение функции F равно единице.
2. выписываем искомую формулу в виде **суммы** стольких слагаемых, сколько строк отмечено.
3. Записываем каждое слагаемое как **произведение** всех аргументов.
4. в каждом произведении ставим «черту» над переменной, если её значение в соответствующей строке = 0.
5. Формула готова.

В нашем случае: F (*A, B, C*) = $A \cdot \overline{B} \cdot \overline{C} + \overline{A} \cdot B \cdot \overline{C} + \overline{A} \cdot \overline{B} \cdot C + A \cdot B \cdot C$

Действия, представленные в данной формуле выполняются по правилам математической логики (логики высказываний или булевой алгебре).

Элементы математической логики

История логики насчитывает около двух с половиной тысячелетии. На первом этапе до начала XX века она развивалась очень медленно. Перемены, происшедшие в логике приблизили ее к реальному мышлению и тем самым к человеческой деятельности. Первоначально математическая логика рассматривалась как базис для логического обоснования различных областей математики. В последние три десятилетия математическая (символическая) логика находит применение во многих областях, в частности, в кибернетике, теории электронных вычислительных машин, теории алгоритмов, электротехнике.

Математическая логика рассматривает схематизированные и в известной степени идеализированные понятия. В логических задачах исходными данными являются не только числа, но и сложные и весьма запутанные высказывания. Во многих случаях необходима ЭВМ для решения логических задач. Умение использовать логические операции повышает эффективность программирования. В самой математической логике немало задач, решение которых можно найти с помощью ЭВМ. Одним из примеров таких задач является поиск тождественно истинных высказываний (тавтологий) в алгебре высказываний. Основным разделом математической логики, лежащим в основе других ее разделов, является логика высказываний.

Элементы логики высказываний

Рассмотрим некоторый класс объектов и установим свойства этих объектов, а также отношения и операции над ними. *Простые высказывания* являются простейшим объектом логики высказываний. Рассмотрим несколько примеров простых высказываний: "Некоторые свойства мышления не моделируются средствами современной кибернетики", "Верста больше километра", " Число 100 больше числа 10". В двузначной логике единственный способ оценки сводится к утверждению или отрицанию истинности высказывания. В такой логике есть смысл абстрагироваться от различий между суждением и высказыванием и рассматривать их как синонимы.

Пользуясь простыми высказываниями можно образовывать *сложные или составные высказывания*, в которых простые входят в качестве элементарных составляющих. Таким образом, в отличие от сложного, простое высказывание не поддается расчленению на высказывания. В образовании сложных высказываний используются слова: *и, или, тогда и только тогда, когда (в том и только в том случае), если ... , то ... , нет.*

Рассмотрим несколько примеров сложных высказываний. 1) Если идет дождь, то солнце не светит. 2) Если ветер дует, то нет дождя.

При первоначальном изучении логики высказываний обращают внимание не на содержание, а на истинность или ложность высказываний. Основная задача логики высказываний заключается в том, чтобы на основании истинности или ложности простых высказываний определить истинность или ложность сложных высказываний. Среди сложных высказываний можно выделить соединительные, разделительные, условные, эквивалентные и высказывание с внешним отрицанием.

Логические операции часто называют логическими связками и для них используется система символических обозначений. Простые высказывания будем обозначать большими буквами A, B, C и т.д., а значения истинности буквами I (либо 1) для "истинно" и L (либо 0) для "ложно". Для булевых переменных, то есть переменных, которые принимают только два значения, называемые "истиной" (TRUE) и "ложью" (FALSE) и обозначаемые соответственно 1 и 0, определены следующие логические операции (здесь приведены различные обозначения, встречающиеся в литературе):

1	$\neg$, $\overline{\ }$ NOT	логическое НЕ (***отрицание***)
2	$\times$, $\wedge$, & , и , AND	логическое И (***произведение***)
3	+ , $\vee$, или , OR	логическое ИЛИ (***сумма***)
4	$\oplus$, $\forall$	логическое исключающее ИЛИ
5	$\rightarrow$	импликация
6	$\leftrightarrow$, $\equiv$	эквиваленция (двойная импликация), тождество

В алгебре логики изучаются логические операции, производимые над высказываниями. Высказывания могут быть истинными или ложными. Применяя к простым высказываниям логические операции, можно строить составные высказывания.

Основными логическими операциями являются:

1. ***Отрицание*** (инверсия, логическое НЕ)

Смысл операции: результат меняется на противоположный (вместо истины — ложь, вместо лжи — истина).

Обозначение[1]: ¬

Таблица истинности:

A	¬A
0	1
1	0

2. ***Логическое сложение*** (дизъюнкция, логическое ИЛИ)

Смысл операции: результат — истина, если хотя бы один операнд — истина (операндом называется то значение или та переменная, над которым (которой) осуществляется операция).

Обозначение: ∨

Другое обозначение: +

Таблица истинности:

A	B	A ∨ B
0	0	0
0	1	1
1	0	1
1	1	1

3. ***Логическое умножение*** (конъюнкция, логическое И)

Смысл операции: результат — истина, если оба операнда — истина.

Обозначение: ∧

Другие обозначения: &, ·

Таблица истинности:

A	B	A ∧ B
0	0	0
0	1	0
1	0	0
1	1	1

Эти три операции считаются базовыми. Остальные рассмотренные нами операции являются дополнительными. Их всегда можно выразить через базовые операции.

4. ***Следование*** (импликация)

Смысл операции: из лжи может следовать что угодно, а из истины — только истина.

Обозначение: →

Таблица истинности:

A	B	A → B
0	0	1
0	1	1
1	0	0
1	1	1

5. ***Равносильность*** (эквиваленция)

Смысл операции: результат — истина, если операнды одинаковые.
Обозначение: ≡
Другое обозначение: ↔
Таблица истинности:

A	B	A ↔ B
0	0	1
0	1	0
1	0	0
1	1	1

6. ***Сложение по модулю 2*** (исключающее ИЛИ, в просторечье XOR)

Смысл операции: результат — истина, если операнды разные.
Обозначение: ⊕
Таблица истинности:

A	B	A ⊕ B
0	0	0
0	1	1
1	0	1
1	1	0

Если в логическом выражении используется несколько логических операций, их порядок определяется ***приоритетами логических операций***:

0) выражение в скобках,

1) логическое НЕ (инверсия),

2) логическое И (конъюнкция),

3) логическое ИЛИ (дизъюнкция), исключающее ИЛИ (эти две операции имеют одинаковый приоритет и выполняются слева направо),

4) следование (импликация),

5) равносильность (эквиваленция).

Как вы, вероятно, поняли, для изменения порядка выполнения логических операций используются скобки.

Для преобразования (упрощения) булевых выражений полезно знать следующие свойства.

1. Законы коммутативности $A + B = B + A$ $A \cdot B = B \cdot A$	**5. Законы идемпотентности** $A + A = A$ $A \cdot A = A$
2. Законы ассоциативности $A + (B + C) = (A + B) + C$ $A \cdot (B \cdot C) = (A \cdot B) \cdot C$	**6. Закон двойного отрицания** $\overline{\overline{A}} = A$
3. Законы дистрибутивности $A \cdot (B + C) = (A \cdot B) + (A \cdot C)$ $A + (B \cdot C) = (A + B) \cdot (A + C)$	**7. Инфолюция** $A \to B = \overline{A} + B$ $A \leftrightarrow B = (A \cdot B) + (\overline{A} \cdot \overline{B})$ $A \leftrightarrow B = (\overline{A} + B) \cdot (A + \overline{B})$
4. Законы де Моргана $\overline{A + B} = \overline{A} \cdot \overline{B}$ $\overline{A \cdot B} = \overline{A} + \overline{B}$	**8. Законы, включающие 0 и 1** $A + \overline{A} = 1$, $A \cdot \overline{A} = 0$ $A + 1 = 1$, $A \cdot 1 = A$ $A + 0 = A$, $A \cdot 0 = 0$ $\overline{1} = 0$

Закон двойственности. Если в каком-либо тождестве все операции сложения (+) заменить на умножение (·) и наоборот, а все встречающиеся в тождестве символы 0 заменит на 1 и наоборот, то полученное выражение будет тождеством, двойственным данному.

ЗАДАНИЕ 1. Упростите формулы:

1) $A + A \cdot B = \ldots$
2) $A \cdot (A + B) = \ldots$
3) $\overline{A} \cdot (A + B) = \ldots$
4) $A + \overline{A} \cdot B = \ldots$

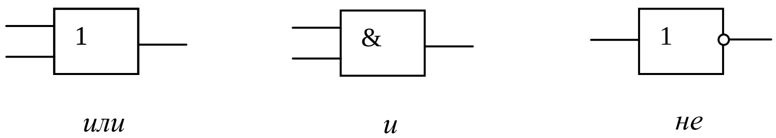
Элементы схемотехники. Логические элементы

Довольно часто при проектировании отдельных узлов компьютера необходимо решить проблему построения логических и электрических схем по заданным функциям, имея лишь описание алгоритма его работы. В этом случае, воспользовавшись указанными данными, можно найти математическую формулу данного узла и на ее основании построить логическую схему.

Конкретный вид электрической схемы, использованной для реализации заданной логической функции, как правило, не имеет существенного значения.

Техническое устройство, реализующее логическую операцию, может рассматриваться просто как **логический элемент,** внутренняя структура которого не конкретизируется.

На схемах логические элементы изображаются следующим образом:



Сложные логические (принципиальные, функциональные) схемы можно конструировать из логических элементов, используя основные понятия и формулы булевой алгебры. Обозначим входные сигналы буквами, сформируем логическую формулу и максимально упростим ее. Затем конструируем соответствующую схему, заменяя каждую логическую операцию соответствующим логическим элементом.

Алгоритм построения логической формулы в виде СДНФ по таблице истинности

1. В таблице истинности отмечаем наборы переменных, на которых значение функции f равно единице.
2. Записываем для каждого отмеченного набора конъюнкцию всех переменных следующим образом: если значение некоторой переменной в этом наборе равно 1, то в конъюнкцию включаем саму переменную, в противном случае — ее отрицание.
3. Все полученные конъюнкции связываем операциями дизъюнкции.

Алгоритм построения логической формулы в виде СКНФ по таблице истинности

1. В таблице истинности отмечаем наборы переменных, на которых значение функции / равно нулю.
2. Записываем для каждого отмеченного набора дизъюнкцию всех переменных следующим образом: если значение некоторой переменной в этом наборе равно 0, то в конъюнкцию включаем саму переменную, в противном случае — ее отрицание.
3. Все полученные дизъюнкции связываем операциями конъюнкции.