

## Лабораторная работа №2 ЛОГИЧЕСКИЕ ЭЛЕМЕНТЫ И СХЕМЫ

**Цель работы:** Ознакомление с основными характеристиками логических элементов и основами синтеза логических схем.

### ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

#### 1 Определения комбинационных и последовательностных устройств

Устройства, реализующие функции алгебры логики, называют *логическими* или *цифровыми* и классифицируют по различным отличительным признакам. Так, по характеру информации на входах и выходах логические устройства подразделяют на устройства последовательного, параллельного и смешанного действия, а по схемному решению и характеру связи между входными и выходными переменными с учётом их изменения по тактам работы – на комбинационные и последовательностные.

В *комбинационных* устройствах значения (0 или 1) сигналов на выходах в каждый конкретный момент времени полностью определяются значениями (комбинацией, набором) действующих в данный момент цифровых входных сигналов. В *последовательностных* же устройствах значения выходных сигналов в  $n$ -такте определяются не только значениями входных сигналов в этом такте, но и зависят от внутренних состояний устройств, которые произошли в результате воздействия входных сигналов в предшествующие такты. Данная работа посвящена изучению простейших комбинационных логических устройств, реализующих логические функции сложения, умножения и отрицания.

#### 2 Основные элементы алгебры логики

Анализ комбинационных устройств удобно проводить с помощью алгебры логики, оперирующей только с двумя понятиями: истинным (логическая 1) и ложным (логический 0). В результате, функции, отображающие информацию, принимают в каждый момент времени только значения 0 или 1. Такие функции называют логическими, а сигналы (входные и выходные переменные) – двоичными (бинарными). Схемные элементы, при помощи которых осуществляется преобразование поступающих на их входы двоичных сигналов и непосредственное выполнение предусмотренных логических операций, называют логическими устройствами.

В общем случае логическое устройство может иметь  $n$  входов и  $m$  выходов. Рассматривая входные сигналы  $x_1, x_2, \dots, x_n$  в качестве аргументов, можно соответствующие выходные сигналы представлять в виде функции  $y_i = f(x_0, x_1, x_2, \dots, x_n)$  с помощью операций алгебры логики.

*Функции алгебры логики* (ФАЛ), иногда называемые *переключательными* функциями, обычно представляют в алгебраической форме (в виде математического выражения), например  $y_i = (x_0 \wedge x_1) \vee (x_1 \wedge x_2)$ , или в виде таблиц истинности (комбинационных таблиц).

*Таблица истинности* содержит всевозможные комбинации (наборы) бинарных значений входных переменных с соответствующими им бинарными значениями выходных переменных; каждому набору входных сигналов соответствует определенное значение выходного сигнала – значение логической функции  $y_i$ . Максимальное число возможных различных наборов (строк) зависит от числа входных переменных  $n$  и равно  $2^n$ .

В булевой алгебре выделяют три основные функции: конъюнкция, дизъюнкция, отрицание. Остальные функции являются производными от приведенных выше.

*Основные логические операции* состоят из следующих элементарных преобразований двоичных сигналов:

- *логическое сложение* или *дизъюнкция*, обозначаемое символом " $\vee$ " (или "+") и называемое также операцией ИЛИ. При этом число аргументов (слагаемых  $x$ ) может быть любым. Эта операция для функции двух переменных  $x_1$  и  $x_2$  описывается в виде логической формулы

$$y = x_1 \vee x_2 = x_1 + x_2.$$

Это значит, что  $y$  истинно (равно 1), если истинно хотя бы одно из слагаемых  $x_1$  или  $x_2$ . И только в случае, когда все слагаемые  $x$  равны 0, результат логического сложения  $y$  также равен 0. Условное обозначение, таблица истинности и другие показатели этой логической функции приведены во втором столбце табл. 1;

- *логическое умножение* или *конъюнкция*, обозначаемое символом " $\wedge$ " (или ".") и называемое также операцией И. При этом число аргументов (сомножителей  $x$ ) может быть любым. Эта операция для функции двух переменных  $x_1$  и  $x_2$  описывается в виде логической формулы

$$y = x_1 \wedge x_2 = x_1 \cdot x_2 = x_1 x_2.$$

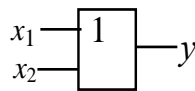
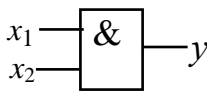
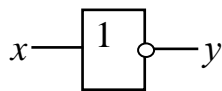
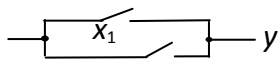
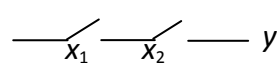
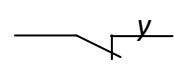
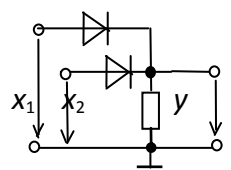
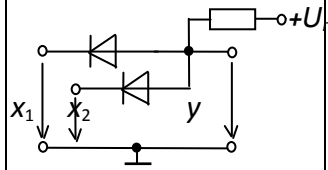
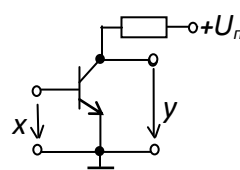

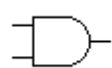

Это значит, что  $y$  истинно (равно 1), если истинны сомножители  $x_1$  и  $x_2$ . В случае, если хотя бы один из сомножителей равен 0, результат логического умножения  $y$  равен 0. Условное обозначение, таблица истинности и другие показатели логической функции И приведены в третьем столбце табл. 1;

- *логическое отрицание* или *инверсия*, обозначаемое чёрточкой над переменной и называемое операцией НЕ. Эта операция записывается в виде

$$y = \bar{x}.$$

Это значит, что  $y$  истинно (равно 1), если  $x$  ложно (равно 0), и наоборот. Очевидно, что операция  $y$  выполняется над одной переменной  $x$  и её значение всегда противоположно этой переменной (см. четвертый столбец табл. 1).

Таблица 1

Формы отображения основных логических функций																																							
Наименование функции	Дизъюнкция	Конъюнкция	Инверсия																																				
Символическая	$\vee$ или $+$	$\wedge$ или $\bullet$	$\bar{x}$																																				
Буквенная	ИЛИ	И	НЕ																																				
Условная графическая																																							
Аналитическая	$y = x_1 \vee x_2 = x_1 + x_2$	$y = x_1 \wedge x_2 = x_1 x_2$	$y = \bar{x}$																																				
Табличная (истинности)	<table><tr><th><math>x_1</math></th><th><math>x_2</math></th><th><math>y</math></th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	$x_1$	$x_2$	$y$	0	0	0	0	1	1	1	0	1	1	1	1	<table><tr><th><math>x_1</math></th><th><math>x_2</math></th><th><math>y</math></th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	$x_1$	$x_2$	$y$	0	0	0	0	1	0	1	0	0	1	1	1	<table><tr><th><math>x</math></th><th><math>y</math></th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table>	$x$	$y$	0	1	1	0
$x_1$	$x_2$	$y$																																					
0	0	0																																					
0	1	1																																					
1	0	1																																					
1	1	1																																					
$x_1$	$x_2$	$y$																																					
0	0	0																																					
0	1	0																																					
1	0	0																																					
1	1	1																																					
$x$	$y$																																						
0	1																																						
1	0																																						
Контактная																																							
Схемотехническая																																							
Международное обозначение (Electronics Workbench)																																							

Основные логические операции ИЛИ, И и НЕ позволяют аналитически описать, а логические элементы ИЛИ (*дизъюнктор*), И (*конъюнктор*) и НЕ (*инвертор*) – реализовать комбинационное устройство любой степени сложности, т. е. операции  $y = x_1 + x_2$ ,  $y = x_1 x_2$  и  $y = \bar{x}$  обладают функциональной полнотой и составляет функционально полный набор.

В качестве примера рассмотрим функцию неравнозначности  $y$  двух переменных  $x_1$  и  $x_2$ , принимающая значение 1 при  $x_1 \neq x_2$  и значение 0 при  $x_1 = x_2 = 0$  или при  $x_1 = x_2 = 1$ , т. е.  $y = \bar{x}_1 x_2 + x_1 \bar{x}_2$ . Операцию неравнозначности чаще называют *суммированием по модулю 2* и обозначают  $y = x_1 \oplus x_2$ .

Примеры контактной и простейшей схемной реализаций дизъюнктора, конъюнктора и инвертора приведены в табл. 1.

### 3 Базовые логические элементы

Особое значение в цифровой электронике имеют универсальные (базовые) логические элементы, способные образовать функционально полный набор, с помощью которых можно реализовать синтез устройств любой сложности. При интегральной технологии удобство изготовления одного базового элемента имеет решающее значение. Поэтому базовые логические устройства составляют основу большинства цифровых ИМС.

К универсальным логическим операциям (устройствам) относят две разновидности базовых элементов:

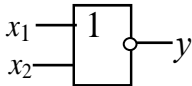
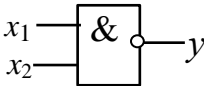
- функцию *Пирса*, обозначаемую символически вертикальной стрелкой  $\downarrow$  (стрелка Пирса) и отображающую операцию ИЛИ-НЕ. Для простейшей функции двух переменных  $x_1$  и  $x_2$  функция  $y = 1$  тогда и только тогда, когда  $x_1 = x_2 = 0$ :

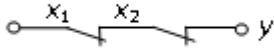
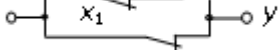
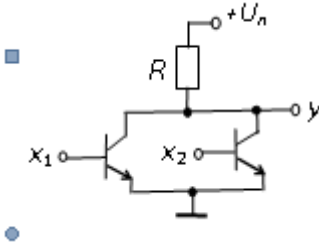
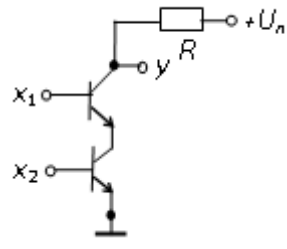


$$y = x_1 \downarrow x_2 = \overline{x_1 + x_2};$$

- функцию *Шеффера*, обозначаемую символически вертикальной черточкой  $|$  (штрих Шеффера) и отображающую операцию И-НЕ. Для простейшей функции двух переменных  $x_1$  и  $x_2$  функция  $y = 0$  тогда и только тогда, когда  $x_1 = x_2 = 1$ :

$$y = x_1 | x_2 = \overline{x_1 x_2}.$$

Таблица 2

Формы отображения базовых логических функций																																
Наименование функции	Функция Пирса	Функция Шеффера																														
Символическая	$\downarrow$	$ $																														
Буквенная	ИЛИ-НЕ	И-НЕ																														
Условная графическая																																
Аналитическая	$y = x_1 \downarrow x_2$	$y = x_1   x_2$																														
Табличная (истинности)	<table border="1" data-bbox="724 1554 983 1758"> <tr> <th><math>x_1</math></th><th><math>x_2</math></th><th><math>y</math></th></tr> <tr> <td>0</td><td>0</td><td>1</td></tr> <tr> <td>0</td><td>1</td><td>0</td></tr> <tr> <td>1</td><td>0</td><td>0</td></tr> <tr> <td>1</td><td>1</td><td>0</td></tr> </table>	$x_1$	$x_2$	$y$	0	0	1	0	1	0	1	0	0	1	1	0	<table border="1" data-bbox="1147 1559 1415 1760"> <tr> <th><math>x_1</math></th><th><math>x_2</math></th><th><math>y</math></th></tr> <tr> <td>0</td><td>0</td><td>1</td></tr> <tr> <td>0</td><td>1</td><td>1</td></tr> <tr> <td>1</td><td>0</td><td>1</td></tr> <tr> <td>1</td><td>1</td><td>0</td></tr> </table>	$x_1$	$x_2$	$y$	0	0	1	0	1	1	1	0	1	1	1	0
$x_1$	$x_2$	$y$																														
0	0	1																														
0	1	0																														
1	0	0																														
1	1	0																														
$x_1$	$x_2$	$y$																														
0	0	1																														
0	1	1																														
1	0	1																														
1	1	0																														

Контактная		
Схемотехническая		
Международное обозначение (Electronics Workbench)		

При одних и тех же значениях аргументов обе функции отображают операцию инверсии. Важнейшие показатели функций Шеффера и Пирса представлены в табл. 2.

В последней строке табл. 2 приведены примеры построения двухвходовой схемы ИЛИ-НЕ, в которой к нагрузочному резистору R подключены коллекторы двух параллельно включенных биполярных транзисторов р-п-р-типа, эмиттеры которых заземлены, и схемы И-НЕ, в которой последовательно включены два биполярных транзистора р-п-р-типа (эмиттер нижнего транзистора подключен к земле) и нагрузочный резистор R.

#### 4 Законы булевой алгебры

Эти законы используются при преобразованиях логических выражений.

Переместительный закон:  $xy = yx$ ;  $x + y = y + x$ ;

сочетательный закон:  $x(yz) = (xy)z = xyz$ ;  $x + (y + z) = (x + y) + z = x + y + z$ ;

распределительный закон:  $x(y + z) = xy + xz$ ;  $x + yz = (x + y)(x + z)$ ;

закон повторения:  $x + x = x$ ;  $x * x = x$ ;

закон обращения: если  $x = y$ , то  $\bar{x} = \bar{y}$ ;

закон двойной инверсии:  $\bar{\bar{x}} = x$ ;

закон универсального множества:  $x * 1 = x$ ;  $x + 1 = 1$ ;

закон дополненности:  $x\bar{x} = 0$ ;  $x + \bar{x} = 1$ ;

закон нулевого множества:  $x * 0 = 0$ ;  $x + 0 = x$ ;

закон поглощения:  $x + x * y = x$ ;  $xy + x\bar{y} = x$ ;

закон склеивания:  $(x + y)(x + \bar{y}) = x$ ;  $xy + x\bar{y} = x$ ;

закон инверсии (закон Де Моргана):  $x\bar{y} = \bar{x} + \bar{y}$ ;  $\bar{x} + \bar{y} = x\bar{y}$ .

#### 5 Логические функции

Любое логическое выражение, составленное из  $n$  переменных  $x_n, x_{n-1}, \dots, x_1$  с помощью конечного числа операций алгебры логики, можно рассматривать как некоторую функцию  $n$  переменных, называемую логической. В соответствии с аксиомами алгебры логики функция может

принимать в зависимости от значения переменных значение 0 или 1. Функция  $n$  логических переменных может быть определена для  $2^n$  значений переменных, соответствующих всем возможным значениям  $n$ -разрядных двоичных чисел.

Основной интерес представляют следующие функции двух переменных  $x$  и  $y$ :

$f_1(x,y) = x * y$  – логическое умножение,

$f_2(x,y) = x + y$  – логическое сложение,

$f_3(x,y) = \overline{x * y}$  – логическое умножение с инверсией,

$f_4(x,y) = \overline{x + y}$  – логическое сложение с инверсией,

$f_5(x,y) = x \oplus y = \overline{x y} + \overline{\overline{x} \overline{y}} = \overline{x y} + \overline{\overline{x y}} = \overline{x y} + x y$  – суммирование по модулю два или «Исключающее ИЛИ»,

$f_6(x,y) = \overline{x \oplus y} = x y + \overline{x y} = x y + \overline{x y}$  – равнозначность.

## 6 Представление логических функций математическими выражениями

Наиболее распространенным способом задания логических функций является табличная форма. Таблицы истинности позволяют полно и однозначно установить все существующие логические связи.

При табличном представлении логических функций их записывают в одной из канонических форм: совершенной дизъюнктивной нормальной форме (СДНФ) или совершенной конъюнктивной нормальной форме (СКНФ).

Математическое выражение логической функции в СДНФ получают из таблицы истинности следующим образом: для каждого набора аргументов, на котором функция равна 1, записывают элементарные произведения переменных, причем переменные, значения которых равны нулю, записывают с инверсией. Полученные произведения, называемые конституентами единицы или минтермами, суммируют.

Запишем логическую функцию  $y$  трех переменных  $a$ ,  $b$  и  $c$ , представленной в виде табл. 3, в СДНФ:

Таблица 3

$$y(a, b, c) = \overline{a}bc + a\overline{b}c + ab\overline{c} + abc.$$

№	$a$	$b$	$c$	$y$
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	1
6	1	1	0	1
7	1	1	1	1

совершенной конъюнктивной нормальной формой называют логическое произведение элементарных сумм, в каждую из которых аргумент или его отрицание входят один раз.

При этом для каждого набора аргументов таблицы истинности, на котором функция  $y$  равна 0, составляют элементарную сумму, причем переменные, значения которых равны 1, записывают с отрицанием. Полученные суммы, называемые конституентами нуля или макстермами, объединяют операцией логического умножения.

Для функции (табл. 3) СКНФ

$$y(a, b, c) = (a + b + c)(a + b + \overline{c})(a + \overline{b} + c)(\overline{a} + b + c).$$

## 7 Переход от логической функции к логической схеме

Для построения логической схемы необходимо логические элементы, предназначенные для выполнения логических операций, располагать, начиная от входа, в порядке, указанном в булевом выражении.

Построим структуру логического устройства, реализующего логическую функцию трех переменных

$$y = (a + b + c)(a + b + \bar{c})(\bar{a} + b + c)(\bar{a} + \bar{b} + c).$$

Слева располагаем входы  $a$ ,  $b$  и  $c$  с ответвлениями на три инвертора, затем четыре элемента ИЛИ и, наконец, элемент И на выходе (рисунок 1).

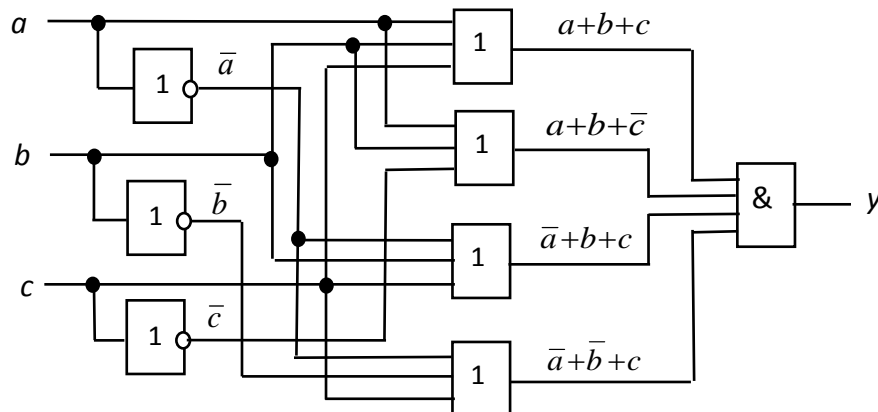


Рисунок 1

Итак, любую логическую функцию можно реализовать непосредственно по выражениям, представленным в виде СДНФ или СКНФ. Однако, полученная таким образом схема, как правило, не оптимальна с точки зрения её практической реализации: она громоздка, содержит много логических элементов и возникают трудности в обеспечении её высокой надёжности.

Алгебра логики позволяет преобразовать формулы, описывающие сложные высказывания с целью их упрощения. Это помогает в конечном итоге определить оптимальную структуру того или иного логического устройства, реализующего любую сложную функцию. Под оптимальной структурой принято понимать такое построение логического устройства, при котором число входящих в его состав элементов минимально.

## 8 Переход от десятичной формы к двоичной

При заполнении таблиц истинности и в других случаях можно использовать, в частности, код 1-2-4-8..., позволяющий оперативно преобразовать двоичное число в десятичное и наоборот, как это показано ниже.

Пример 1. Преобразуем двоичное число 101100 в десятичное. Для

$$\begin{array}{r} 32168421 = 32+8+4 = 44(10) \\ 1\ 0\ 1100(2) \end{array}$$

этого записываем двоичное число, над его разрядами размещаем разряды кода  $1 - 2 - 4 - 8 - 16 - 32$ , как в данном примере, и суммируем те десятичные числа этого кода, которые размещены над единицами двоичного числа.

Пример 2. Преобразуем десятичное число 40 в двоичный код. Для этого, записываем код  $1 - 2 - 4 - 8 - 16 - 32$  так, чтобы старший разряд этого кода

$$\begin{array}{r} 32 \ 16 \ 8 \ 4 \ 2 \ 1 = 32+8=40 \\ \rightarrow 40(10)=101000(2) \\ 1 \ 0 \ 1 \ 0 \ 0 \ 0 \end{array}$$

не превышал преобразуемого числа. Затем суммируем только те десятичные числа кода, которые образуют преобразуемое десятичное число. Под этими числами кода ставим единицы, а под остальными нули, как показано в данном примере.

## ПРАКТИЧЕСКАЯ ЧАСТЬ

**Задание 1. Преобразовать** в соответствующую форму следующие числа:

Вариант	Из десятичной в двоичную	Из двоичной в десятичную
1,6,11,16	7, 45,114	101, 11100, 1000010
2,7,12,17	5, 36,103	100, 10110, 1101000
3,8,13,18	3, 39,96	011, 01111, 1001001
4,9,14,19	4, 47, 99	010, 10001, 1010100
5,10,15,20	6, 42, 109	111, 11011, 1110011

Так же запишите в двоичной форме дату **своего рождения**.

Результат **записать** в отчет.

**Задание 2. Запустить** среду Electronics Workbench. **Собрать** на рабочем поле среды схему для испытания *основных и базовых логических элементов* (рисунок 2) и **установить** в диалоговых окнах компонентов их параметры или режимы работы. **Скопировать** схему на страницу отчёта.

Схема (рисунок 2) собрана на двоичных основных [OR (ИЛИ), AND (И) и NOT (НЕ)] и универсальных (базовых) [NAND (И-НЕ) и XOR (ИЛИ-НЕ)] логических элементах. В схему включены ключи 1, 2, ..., 9, пробники, источник постоянного напряжения 12 В и логический анализатор XLA1.

Оперируя ключами **1, 2, ..., 9, сформировать** все возможные комбинации аргументов  $x_1$  и  $x_2$  (00, 10, 01 и 11) на входе дизъюнктора (**OR**), конъюнктора (**AND**), штриха Шеффера (**NAND**) и стрелки Пирса (**NOR**) и **записать** значения выходных логических функций  $y_k$  (0 или 1) в таблицу 4.

Заметим, что если ключ замкнут, то на этот вход элемента будет подана логическая единица (положительный потенциал 12 В), а при разомкнутом ключе – логический ноль. Поскольку инвертор (**NOT**) имеет один вход, то для формирования двух значений входного сигнала (логической единицы или логического нуля) достаточно одного ключа **5**.

Значения функций исследуемых элементов можно контролировать с помощью пробников: если выходной сигнал элемента равен логической единице, то включенный на выходе этого элемента пробник светится.



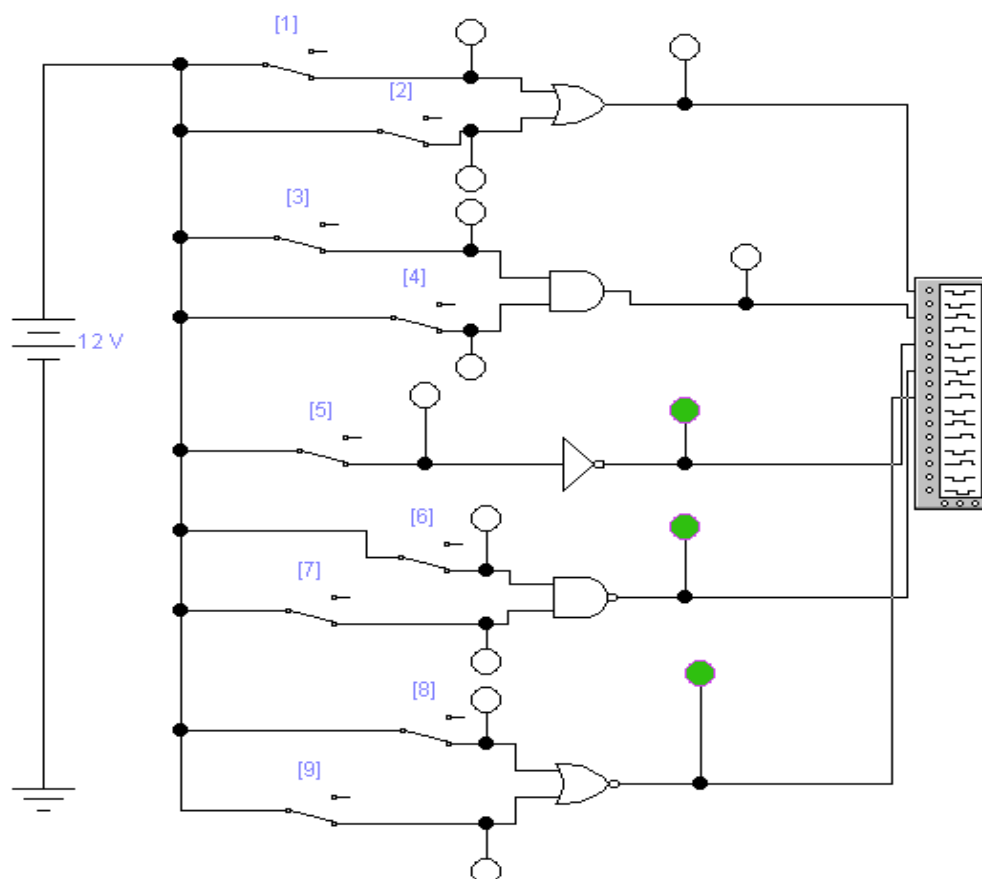


Рисунок 2

Для удобства измерения сигналов выходы логических элементов подключены к входам 1, 3, 5, 7 и 9 анализатора **XLA1**. При моделировании происходит медленная развёртка временных диаграмм в окне анализатора. По достижению интервала времени, равном 70...80% ширины окна, следует посредством кнопки **Run/Stop** выключать процесс моделирования.

Таблица 4

Дизъюнктор [ИЛИ ( <b>OR</b> )]			Конъюнктор [И ( <b>AND</b> )]			Инвертор [НЕ ( <b>NOT</b> )]		Штрих Шеффера [И-НЕ ( <b>NAND</b> )]			Стрелка Пирса [ИЛИ-НЕ ( <b>NOR</b> )]		
$x_1$	$x_2$	$y$	$x_1$	$x_2$	$y$	$x$	$y$	$x_1$	$x_2$	$y$	$x_1$	$x_2$	$y$
0	0		0	0		0		0	0		0	0	
0	1		0	1				0	1		0	1	
1	0		1	0		1		1	0		1	0	
1	1		1	1				1	1		1	1	

**Заменить** источник постоянного напряжения на генератор прямоугольных импульсов с частотой 1 Гц и напряжением 5 В. **Замкнуть** ключи. В отчет **вставить** результат с дисплея логического анализатора.

***Перепроектировать*** схему с рисунка 2 таким образом, чтобы проверка логических элементов ***осуществлялась без использования источников питания.***

**Задание 3. "Перетащить"** из библиотеки на рабочее поле среды Electronics Workbench необходимые логические элементы и **собрать** схему для реализации заданной в таблице 5 логической функции у с тремя аргументами а, в и с. **Скопировать** собранную логическую схему в отчёт.

Таблица 5

1	$y = (\bar{a}b + \bar{c})(\bar{a} + \bar{b} + c)(a + b + c).$
2	$y = (a + b + \bar{c})(\bar{a} + \bar{b}c)(a + \bar{b} + \bar{c}).$
3	$y = (b + a\bar{c})(\bar{a} + bc)(a + \bar{b} + c).$
4	$y = (\bar{a}\bar{b} + \bar{c})(a + \bar{b} + c)(ab + \bar{c}).$
5	$y = (a + \bar{b}c)(\bar{a} + b + \bar{c})(ab + c).$
6	$y = (ab + c)(a + \bar{b} + \bar{c})(\bar{a} + b + c).$
7	$y = (a + \bar{b} + \bar{c})(\bar{a} + \bar{b}c)(a + \bar{b} + \bar{c}).$
8	$y = (\bar{b} + \bar{a}c)(a + b\bar{c})(a + \bar{b} + c).$
9	$y = (\bar{a} + b\bar{c})(\bar{a} + b + c)(a\bar{b} + c).$
10	$y = (a + b\bar{c})(a + \bar{b} + \bar{c})(a\bar{b} + c).$
11	$y = (a\bar{b} + c)(a + b + \bar{c})(\bar{a} + b + c).$
12	$y = (\bar{a} + b + \bar{c})(\bar{a} + b\bar{c})(a + \bar{b} + \bar{c}).$
13	$y = (b + \bar{a}c)(a + \bar{b}c)(a + \bar{b} + c).$
14	$y = (\bar{a}b + c)(a + b + \bar{c})(a\bar{b} + c).$
15	$y = (\bar{a} + bc)(\bar{a} + b + \bar{c})(\bar{a}\bar{b} + c).$
16	$y = (a\bar{b} + c)(a + b + \bar{c})(\bar{a} + \bar{b} + \bar{c}).$
17	$y = (\bar{a} + \bar{b} + c)(a + b\bar{c})(\bar{a} + b + c).$
18	$y = (\bar{b} + \bar{a}c)(a + \bar{b}\bar{c})(\bar{a} + b + \bar{c}).$
19	$y = (ab + c)(\bar{a} + b + \bar{c})(\bar{a}\bar{b} + c).$
20	$y = (\bar{a} + b\bar{c})(a + \bar{b} + c)(\bar{a}\bar{b} + \bar{c}).$

**Заполнить** таблицу истинности полученной функции (таблица 6), записать ее в отчет.

a	b	c	y
0	0	0	
1	0	0	
1	1	0	
1	0	1	
0	1	0	
0	1	1	
0	0	1	
1	1	1	

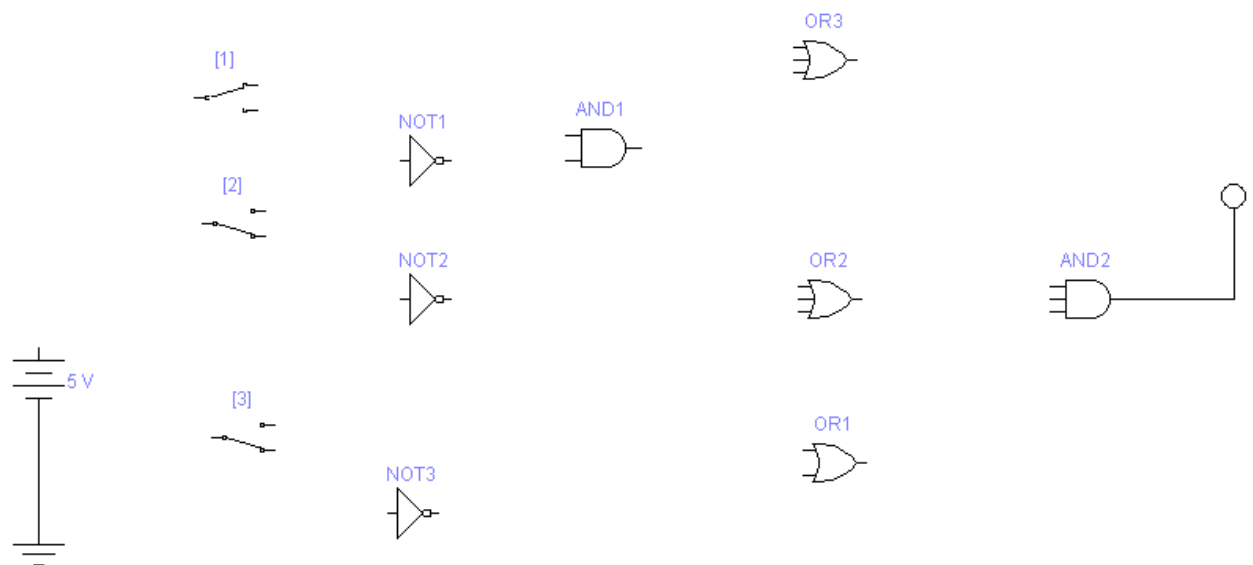
В качестве примера соберём схему для реализации логической функции

$$y = (ab + \bar{c})(\bar{a} + \bar{b} + c)(a + b + c).$$

Анализ функции показывает, что для построения логической схемы нам потребуются три инвертора, три дизъюнктора, причем один дизъюнктор с двумя, а два – с тремя входами, и два конъюнктора, причём один с двумя, а другой с тремя входами.

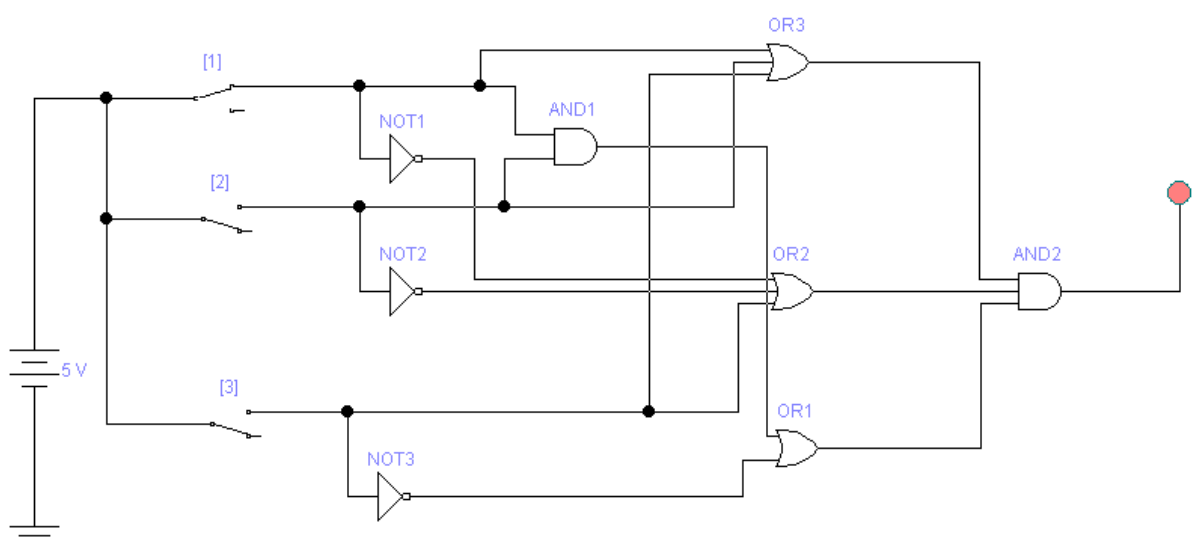
"Перетащим" на рабочее поле среды необходимые модели логических элементов из, располагая их, начиная с входа, а именно:

- три инвертора **NOT** (**NOT1**, **NOT2** и **NOT3**) для получения инверсий  $\bar{a}$ ,  $\bar{b}$  и  $\bar{c}$  аргументов  $a$ ,  $b$  и  $c$ ;
- конъюнктор **AND1** с двумя входами для реализации функции  $ab$ ;
- три дизъюнктора: **OR2** для реализации функции  $y_1 = a + b + c$ , **OR3** для реализации функции  $y_2 = \bar{a} + \bar{b} + c$  и **OR1**, реализующий функцию  $y_3 = ab + \bar{c}$ , разместив их друг под другом (рисунок 3).



Для выполнения функции логического умножения  $y = y_1 y_2 y_3$  добавим в схему конъюнктор **AND2** с тремя входами, к выходу которого подключим логический пробник) для сигнализации появления логической единицы на выходе схемы. "Перетащим" из соответствующих библиотек на рабочее поле источник постоянного напряжения и ключи **1,2,3** расположив их на входе схемы.

Соединив "проводниками" входы и выходы элементов в соответствии с логическими выражениями составляющих заданной функции (рисунок 4), приступим к моделированию. С этой целью вначале щелкнем мышью на кнопке **Run/Stop**, затем нажмём управляющую ключом клавишу с цифрой **1**



клавиатуры. Если соединения элементов выполнены правильно, то пробник

засветится. При выключении ключа **1** пробник гаснет и т. д. Используя клавиши **1,2,3**, заполним таблицу истинности (таблица 6).

### **СОДЕРЖАНИЕ ОТЧЕТА**

1. Титульный лист
2. Цель и задачи работы
3. Перечень приборов, использованных в экспериментах, с их краткими характеристиками.
4. Краткие теоретические сведения (ответы на контрольный вопросы)
5. Изображения электрической схемы для испытания логических элементов и собранной схемы для реализации заданной логической функции.
6. Таблицы истинности, отображающие работу исследуемых логических элементов.
7. Выводы по работе.

### **КОНТРОЛЬНЫЕ ВОПРОСЫ.**

1. Дать определение дизъюнкции, конъюнкции, инверсии ? Привести аналитическое и схематическое обозначение.
2. Какие логические операции осуществляют функции Пирса и функции Шефера ?
3. Законы булевой алгебры ?