ЛОГИЧЕСКИЕ ЭЛЕМЕНТЫ И СХЕМЫ

1. Цель работы

Ознакомление с основными характеристиками логических элементов и основами синтеза логических схем.

2. 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.

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 $x_1, x_2, \ldots, x_n$ в качестве аргументов, можно соответствующие выходные сигналы представлять в виде функции $y_i = f(x_0, x_1, x_2, \ldots, x_n)$ 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:

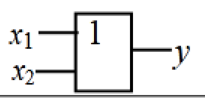
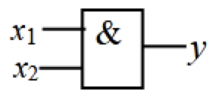
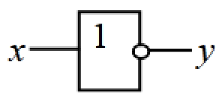
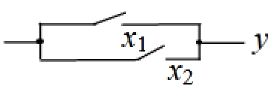
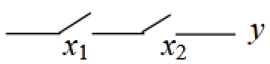
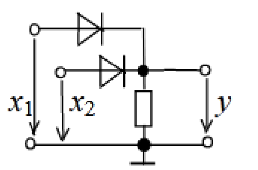
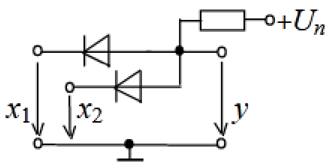
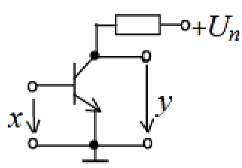
• *логическое сложение* или *дизъюнкция*, обозначаемое символом "∨" (или "+") и называемое также операцией ИЛИ. При этом число аргументов (слагаемых x) может быть любым. Эта операция для функции двух переменных x_1 и x_2 описывается в виде логической формулы $y = x_1 \vee x_2 = x_1 + x_2$. Это значит, 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 этой логической функции приведены во втором столбце табл. 1;

• *логическое умножение* или *конъюнкция*, обозначаемое символом "∧" (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	Дизъюнкция	Конъюнкция	Инверсия
Символическая	∨ или +	∧ или ·	$\bar{x}$
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)	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			
Схемо-техническая			

Основные логические операции ИЛИ, И и НЕ позволяют аналитически описать, а логические элементы ИЛИ (*дизъюнктор*), И (*конъюнктор*) и НЕ (*инвертор*) – реализовать 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 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 $y = x_1 \oplus x_2$.

3. БАЗОВЫЕ ЛОГИЧЕСКИЕ ЭЛЕМЕНТЫ.

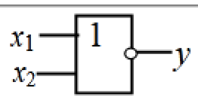
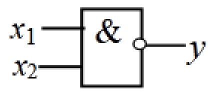
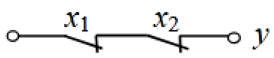
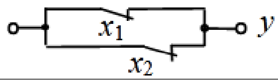
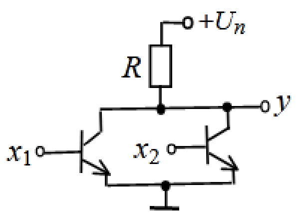
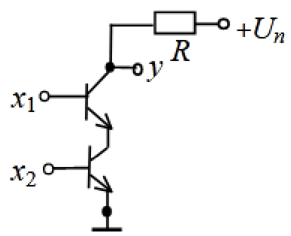
Особое значение в цифровой электронике имеют универсальные (базовые) логические элементы, способные образовать функционально полный набор, с помощью которых можно реализовать синтез устройств любой сложности. К универсальным логическим операциям (устройствам) относят две разновидности базовых элементов:

• *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 $x_1 = x_2 = 1$: $y = x_1 | x_2 = \overline{x_1 x_2}$.

Т а б л и ц а 2

Формы отображения базовых логических функций

<table>
<tr><th>Наименование</th><th>Функция Пирса</th><th>Функция Шеффера</th></tr>
<tr><td>Символическая</td><td>$\downarrow$</td><td>$|$</td></tr>
<tr><td>Буквенная</td><td>ИЛИ-НЕ</td><td>И-НЕ</td></tr>
<tr><td>Условная графическая</td><td>
</td><td>
</td></tr>
<tr><td>Аналитическая</td><td>$y = x_1 \downarrow x_2$</td><td>$y = x_1 | x_2$</td></tr>
<tr><td>Табличная (истинности)</td><td><table>
<tr><th>x_1</th><th>x_2</th><th>y</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></td><td><table>
<tr><th>x_1</th><th>x_2</th><th>y</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></td></tr>
<tr><td>Контактная</td><td>
</td><td>
</td></tr>
<tr><td>Схемо-техническая</td><td>
</td><td>
</td></tr>
</table>

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 *p-n-p*-типа, эмиттеры которых заземлены, и схемы И-НЕ, в которой последовательно включены два биполярных транзистора *p-n-p*-типа (эмиттер нижнего транзистора подключен к земле) и нагрузочный резистор R.

4. 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:

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

Таблица 3

№	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, значение которых равно 1, записывают с отрицанием. Полученные суммы, называемые *конституентами нуля* или *макстермами*, объединяют операцией логического умножения.

Для функции (табл. 3) СКНФ

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

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.

5. 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 (комбинационную схему), 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 (рис. 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.

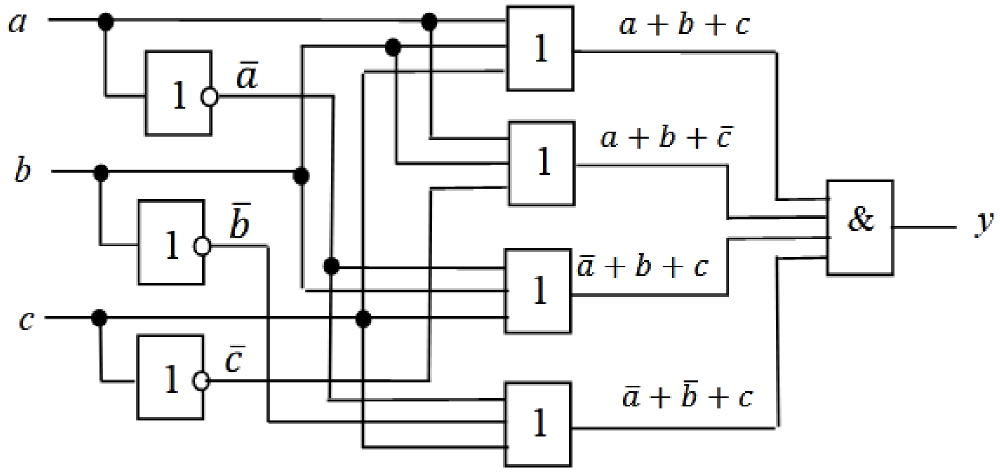


Рис. 1. Комбинационная схема

Примечание. Обозначения элементов в схемах соответствуют ГОСТ 2.743—91 (ознакомиться можно в папке Литература к курсу на платформе СДО СВГУ https://sdo.svgu.ru/

Курс 09.03.03 (Б) Основы элементной базы вычислительной техники (Марсенич И.А.)

https://sdo.svgu.ru/course/view.php?id=678

ЗАДАНИЕ

Составить отчет по самостоятельной работе, в котором указать:

- номер индивидуального варианта;
- исходные данные;
- построить таблицу истинности логической функции y с помощью программы на языке Python *(привести код программы на Python и собственно таблицу истинности функции) – можно сделать скриншот ил вставить в отчет текст кода);*
- аналитическую запись функции y в виде СДНФ;
- аналитическую запись функции y в виде СКНФ;
- построить комбинационную (логическую) схему устройства, реализующего функцию y по исходной формуле;
- построить комбинационную схему логического устройства, реализующего функцию y (представленную в СДНФ);
- построить комбинационную схему логического устройства, реализующего функцию y (представленную в СКНФ).

Варианты заданий

№ варианта	Вид формулы
1.	$y = (a + b)(\bar{c} + d) + (c + \bar{a})d$
2.	$y = (a + d)(\bar{c} + d) + (b + \bar{a})(a + c)$
3.	$y = (a + b)(c + \bar{d}) + (c + \bar{a})b$
4.	$y = (a + b)(\bar{c} + d)(c + \bar{a}) + (d + a)$
5.	$y = (a + b)(\bar{c} + d) + (c + \bar{a})(d + b)$
6.	$y = (a + b + c)(\bar{c} + d) + (c + \bar{a})d$
7.	$y = (a + d + c)(\bar{c} + d) + (b + \bar{a})(a + c)$
8.	$y = (a + b + c)(c + \bar{d}) + (c + \bar{a})b$
9.	$y = (a + b + c)(\bar{c} + d)(c + \bar{a}) + (d + a)$
10.	$y = (a + b + c)(\bar{c} + d) + (c + \bar{a})(d + b)$
11.	$y = (a + b + d)(\bar{c} + d) + (c + \bar{a})d$
12.	$y = (a + d + d)(\bar{c} + d) + (b + \bar{a})(a + c)$
13.	$y = (a + b + d)(c + \bar{d}) + (c + \bar{a})b$
14.	$y = (a + b + d)(\bar{c} + d)(c + \bar{a}) + (d + a)$
15.	$y = (b + d)(\bar{c} + d) + (c + \bar{a})(d + b)$
16.	$y = (b + c)(\bar{c} + d) + (c + \bar{a})d$
17.	$y = (d + c)(\bar{c} + d) + (b + \bar{a})(a + c)$
18.	$y = (b + c)(c + \bar{d}) + (c + \bar{a})b$
19.	$y = (b + c)(\bar{c} + d)(c + \bar{a}) + (d + a)$
20.	$y = (b + c)(\bar{c} + d) + (c + \bar{a})(d + b)$
21.	$y = (d + b + c)(\bar{c} + d) + (c + \bar{a})d$
22.	$y = (d + d + c)(\bar{c} + d) + (b + \bar{a})(a + c)$
23.	$y = (d + b + c)(c + \bar{d}) + (c + \bar{a})b$
24.	$y = (d + b + c)(\bar{c} + d)(c + \bar{a}) + (d + a)$
25.	$y = (d + b + c)(\bar{c} + d) + (c + \bar{a})(d + b)$

Контрольные вопросы

1. Укажите **признаки,** характеризующие основные логические элементы.

На входах логических элементов аналоговые сигналы, а на выходах – цифровые

Операции логического сложения, логического умножения и инверсия не составляют функционально полный набор

Используя основные логические операции И, ИЛИ и НЕ, можно аналитически выразить любую сложную логическую функцию

Минимальный логический базис составляют операции ИЛИ и НЕ или И и НЕ

Входные и выходные сигналы логических элементов могут принимать только два значения: логическую 1 и логический 0

Операция логического сложения совпадает с операцией обычного сложения

2. Укажите **выражение** логической функции двух переменных x_1 и x_2, реализуемой элементом "Стрелка Пирса".

○ $y = \bar{x}_1 x_2 + x_1 \bar{x}_2$ ○ $y = \overline{x_1 x_2}$ ○ $y = \overline{x_1 + x_2}$

○ $y = x_1 \oplus x_2$ ○ $y = x_1 + x_2$ ○ $y = x_1 x_2$

3. Укажите **выражение** логической функции двух переменных x_1 и x_2, реализуемой элементом "Штрих Шеффера".

○ $y = \bar{x}_1 x_2 + x_1 \bar{x}_2$ ○ $y = \overline{x_1 x_2}$ ○ $y = x_1 \oplus x_2$

○ $y = \overline{x_1 + x_2}$ ○ $y = x_1 + x_2$ ○ $y = x_1 x_2$

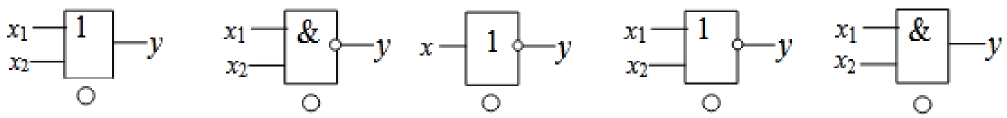
4. Укажите **выражение** логической функции трех переменных *a*, *б* и *c*, записанной в совершенной дизъюнктивной нормальной форме (СДНФ).

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

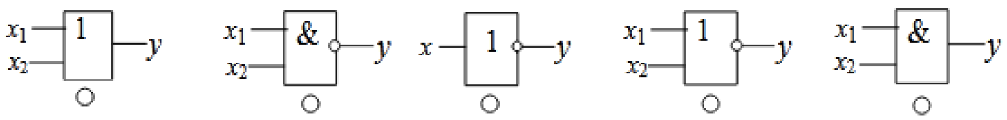
○ $y(a, b, c) = (a + b + c)(a + b + \bar{c})(a + \bar{b} + c)(\bar{a} + b + c)$

○ $y(a, b, c) = (\bar{a}b + c + a\bar{b}c)(ab\bar{c} + \overline{ab} + \bar{c}a)$

5. Укажите **элемент** ИЛИ-НЕ.



6. Укажите **элемент** И.



7. Укажите значение **функции** $y = (ab + \bar{c})(\bar{a} + \bar{b})$, если $a = b = c = 1$.

○ 1 ○ 0