

Практическая работа №9

«Изучение логических элементов»

Цель работы:

- изучение функциональных характеристик основных логических элементов;
- освоение построения базовых логических схем

Теоретический блок:

Логические операции и их значение в информатике

Логические операции — это операции, которые применяются к значениям, принимающим одно из двух возможных состояний: «истина» (True) или «ложь» (False). В цифровых системах эти состояния часто обозначаются как 1 и 0 соответственно. Логические операции являются основой для построения цифровых схем, которые используются в компьютерах, системах управления, а также во многих других электронных устройствах.

В информатике логические операции используются для принятия решений, обработки данных и управления процессами. Они играют ключевую роль в программировании, проектировании компьютерных систем и разработке алгоритмов. Применяя логические операции, можно описывать сложные условия, проверять выполнение определённых условий и управлять потоками выполнения программ.

Основные логические элементы

В цифровой логике существуют несколько базовых логических элементов, каждый из которых выполняет свою функцию:

1. Логический элемент AND (И):

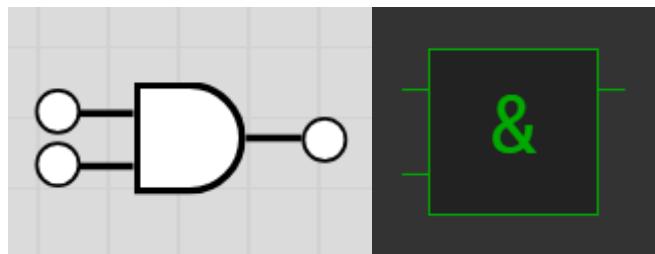


Рисунок 1 – обозначения логического элемента AND (И)

Операция AND возвращает «истину» (1) только тогда, когда все входные значения равны «истине» (1).

Таблица истинности:

Вход А	Вход В	Выход (A AND B)
0	0	0
0	1	0
1	0	0
1	1	1

2. Логический элемент OR (ИЛИ):

Операция OR возвращает «истину» (1), если хотя бы один из входов равен «истине» (1).

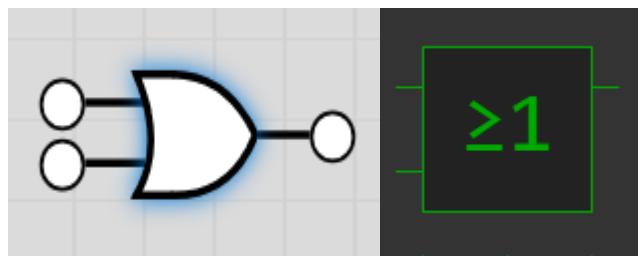


Рисунок 2 – обозначения логического элемента OR (ИЛИ)

Таблица истинности:

Вход А	Вход В	Выход (A OR B)
0	0	0
0	1	1
1	0	1
1	1	1

3. Логический элемент NOT (НЕ):

Операция NOT инвертирует значение входа: если на входе «истина» (1), то на выходе будет «ложь» (0), и наоборот.

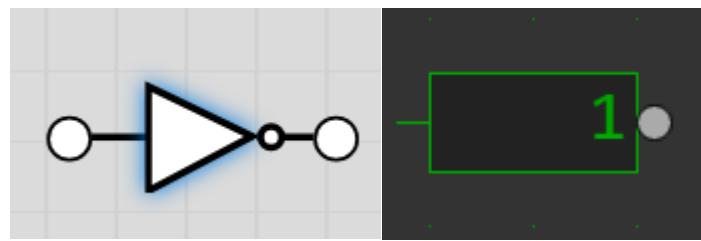


Рисунок 3 – обозначение логического элемента NOT (НЕ)

Таблица истинности:

Вход А	Выход (NOT A)
0	1
1	0

4. Логический элемент NAND (НЕ-И):

Операция NAND является обратной операцией AND. Она возвращает «ложь» (0) только тогда, когда все входные значения равны «истине» (1).

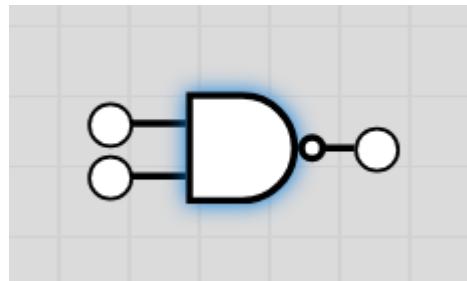


Рисунок 4 – обозначение логического элемента NAND (НЕ-И)

Таблица истинности:

Вход А	Вход В	Выход (A NAND B)
0	0	1
0	1	1
1	0	1
1	1	0

5. Логический элемент NOR (НЕ-ИЛИ):

Операция NOR является обратной операцией OR. Она возвращает «ложь» (0), если хотя бы один из входов равен «истине» (1).

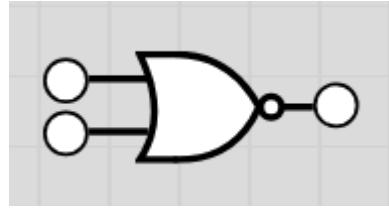


Рисунок 5 – обозначение логического элемента NOR (НЕ-ИЛИ)

Таблица истинности:

Вход А	Вход В	Выход (A NOR B)
0	0	1
0	1	0
1	0	0
1	1	0

6. Логический элемент XOR (Исключающее ИЛИ):

Операция XOR возвращает «истину» (1), если значения на входах различны.

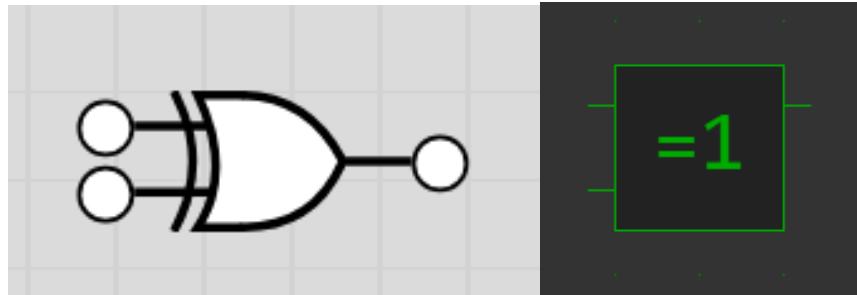


Рисунок 6 – обозначение логического элемента XOR (Исключающее ИЛИ)

Таблица истинности:

Вход А	Вход В	Выход (A XOR B)
0	0	0
0	1	1
1	0	1
1	1	0

7. Логический элемент XNOR (Исключающее НЕ-ИЛИ):

Операция XNOR является обратной операцией XOR. Она возвращает «истину» (1), если значения на входах одинаковы.

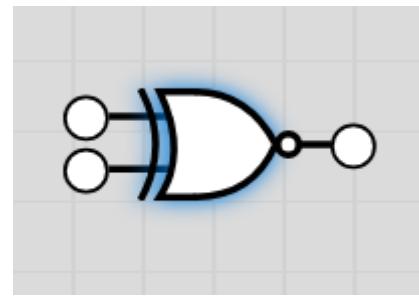


Рисунок 7 – обозначение логического элемента XNOR (Исключающее НЕ-ИЛИ)

Таблица истинности:

Вход А	Вход В	Выход (A XNOR B)
0	0	1
0	1	0
1	0	0
1	1	1

Практический блок:

Логические элементы широко используются в различных областях техники и информационных технологий. Разберем их работу на примере задач «светофор» и «сигнализация».

ЗАДАЧА: «Светофор»

Цель этой работы — построить схему, моделирующую работу простого светофора, используя логические элементы AND и NOT. Светофор должен управлять движением автомобилей и пешеходов на перекрестке. Для реализации этой задачи используются комбинации логических операций для моделирования различных условий.

Определим условия работы светофора, алгоритм его работы, а затем построим логическую схему.

Условия работы светофора

- Автомобили могут двигаться **только** если для них горит зеленый свет.
- Красный свет для автомобилей означает, что движение **запрещено**.
- Пешеходы могут переходить дорогу, **только** если для них горит зеленый свет и при этом для автомобилей горит красный свет.
- Если пешеход нажимает **кнопку** загорается красный свет для автомобилей и зеленый свет для пешеходов.
- Если на перекрестке есть трафик и пешеход **не нажимал** кнопку, светофор для автомобилей **остается** зеленым.

Алгоритм работы светофора

- **A (Автомобильное движение):** Состояние светофора для автомобилей ($0 = \text{красный}$, $1 = \text{зеленый}$).
- **P (Пешеходное движение):** Состояние светофора для пешеходов ($0 = \text{красный}$, $1 = \text{зеленый}$).
- **B (Кнопка пешехода):** Состояние кнопки пешехода ($0 = \text{не нажата}$, $1 = \text{нажата}$).
- **T (Трафик):** Датчик трафика на дороге ($0 = \text{нет трафика}$, $1 = \text{есть трафик}$).
- Автомобили могут двигаться, если ($A = 1$ и $P = 0$).

- Пешеходы могут двигаться, если ($P = 1$ и $A = 0$).
- Если кнопка В нажата ($B = 1$), то через некоторое время $A = 0$ и $P = 1$.
- Если датчик трафика Т активен ($T = 1$) и кнопка не нажата ($B = 0$), то светофор для автомобилей остается зеленым ($A = 1$).

Логические операции

- $A = T \text{ AND NOT } B$ (автомобильное движение при наличии трафика и ненажатой кнопке пешехода).
- $P = B \text{ AND NOT } T$ (пешеходное движение при нажатой кнопке или отсутствии трафика и красном свете для автомобилей).

Построение логической схемы

Основными элементами на схеме будут:

- Входы для Т, В.
- Выходы для А, Р.
- Логический элемент AND для $A = T \text{ AND NOT } B$.
- Логический элемент AND для $P = B \text{ AND NOT } T$
- Логические элементы NOT для инверсии Т и В.

Таким образом, для реализации базового светофора будет справедлива следующая таблица истинности:

- $T = 0, B = 0 \rightarrow A = 0, P = 0$.
- $T = 1, B = 0 \rightarrow A = 1, P = 0$.
- $T = 1, B = 1 \rightarrow A = 0, P = 1$.
- $T = 0, B = 1 \rightarrow A = 0, P = 1$

Реализация логических схем будет происходить на таких бесплатных онлайн-ресурсах, как:

1. <https://simulator.io/board>
2. <https://logic.ly/demo/>

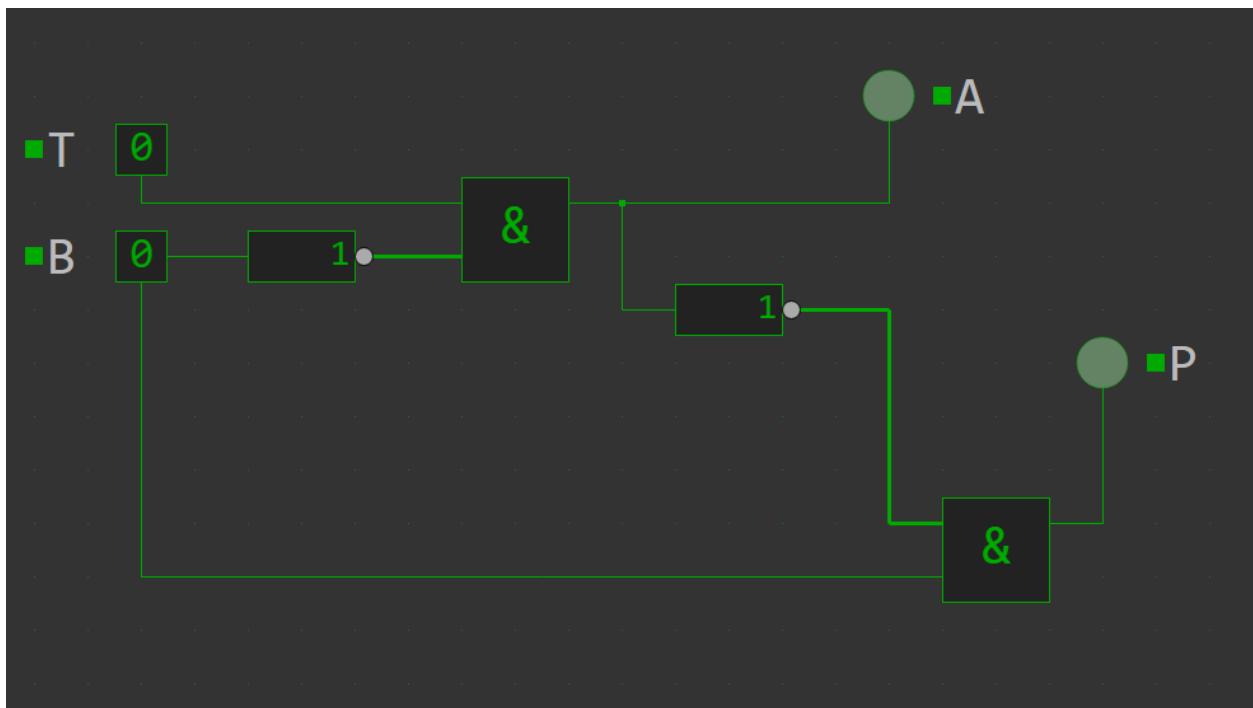


Рисунок 8 – логическая схема светофора

Добавьте сначала входные и выходные сигналы, затем логические элементы AND и NOT. Подключите сигналы Т и инвертированный сигнал В к элементу AND и выходу А. Проверьте, что по включению Т – выход становится равным 1, а при включении В – 0. Далее, подключите ко второму элементу AND вход В и инвертированный сигнал от первого элемента AND. Таким образом, мы гарантируем, что одновременно может быть активен лишь один выход – «зеленый сигнал» для автомобилей или для пешеходов.

ЗАДАЧА: «Сигнализация»

Цель этой работы — построить схему, моделирующую работу домашней сигнализации, используя логические элементы AND, OR, XOR, NOT.

Условие задачи

Сигнализация должна реагировать на три входных сигнала:

1. **X** — Датчик движения.
2. **Y** — Датчик двери.
3. **Z** — Датчик окна.

Эти сигналы образуют 8 комбинаций, 7 из которых приведут к срабатыванию сигнализации:

1. $X = 0, Y = 0, Z = 0$: $\text{NOT } X = 1 \text{ AND}(Y, Z) = 0 \text{ XOR}(Y, Z) = 0 \text{ NOR}(X, Y) = 1$
2. $X = 0, Y = 0, Z = 1$: $\text{NOT } X = 1 \text{ AND}(Y, Z) = 0 \text{ XOR}(Y, Z) = 1 \text{ NOR}(X, Y) = 1$
3. $X = 0, Y = 1, Z = 0$: $\text{NOT } X = 1 \text{ AND}(Y, Z) = 0 \text{ XOR}(Y, Z) = 1 \text{ NOR}(X, Y) = 0$
4. $X = 0, Y = 1, Z = 1$: $\text{NOT } X = 1 \text{ AND}(Y, Z) = 1 \text{ XOR}(Y, Z) = 0 \text{ NOR}(X, Y) = 0$
5. $X = 1, Y = 0, Z = 1$: $\text{NOT } X = 0 \text{ AND}(Y, Z) = 0 \text{ XOR}(Y, Z) = 1 \text{ NOR}(X, Y) = 0$
6. $X = 1, Y = 1, Z = 0$: $\text{NOT } X = 0 \text{ AND}(Y, Z) = 0 \text{ XOR}(Y, Z) = 1 \text{ NOR}(X, Y) = 0$
7. $X = 1, Y = 1, Z = 1$: $\text{NOT } X = 0 \text{ AND}(Y, Z) = 1 \text{ XOR}(Y, Z) = 0 \text{ NOR}(X, Y) = 0$

Не сработает ($S = 0$) сигнализация только в одном случае: $X = 1, Y = 0, Z = 0$: $\text{NOT } X = 0 \text{ AND}(Y, Z) = 0 \text{ XOR}(Y, Z) = 0 \text{ NOR}(X, Y) = 0$

Таблица истинности:

X	Y	Z	NOT X	AND (Y, Z)	XOR (Y, Z)	NOR (X, Y)	S
0	0	0	1	0	0	1	1
0	0	1	1	0	1	1	1
0	1	0	1	0	1	0	1
0	1	1	1	1	0	0	1
1	0	0	0	0	0	0	0
1	0	1	0	0	1	0	1
1	1	0	0	0	1	0	1
1	1	1	0	1	0	0	1

Построение логической схемы

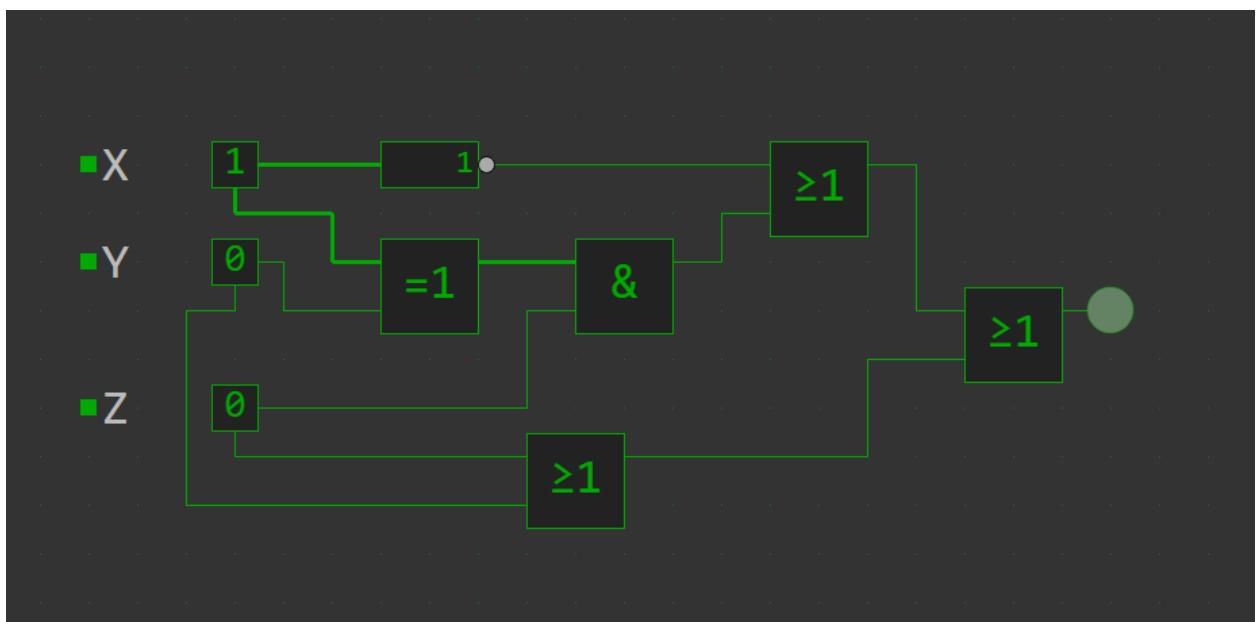
Элементы схемы:

- X (Датчик движения)
- Y (Датчик двери)
- Z (Датчик окна)
- NOT X — Инверсия сигнала X

- AND — Используется для обнаружения ситуации, когда сработали датчики Y и Z.
- OR — Используется для объединения условий срабатывания сигнализации.
- XOR — Для обнаружения ситуации, когда только один из двух датчиков (например, Y или Z) сработал.

Логика работы схемы:

- NOT X — Если датчик движения не сработал.
- AND (Y, Z) — Срабатывает, если оба датчика двери и окна активированы.
- XOR (Y, Z) — Срабатывает, если активирован только один из датчиков двери или окна, но не оба одновременно.
- OR (AND, XOR, NOR) — Суммирует условия срабатывания сигнализации.



Задание для отчета по лабораторной работе

1. Выполнить самостоятельно задачи «светофор» и «сигнализация», результаты отразить в отчете.
2. Выполнить одну из задач по вариантам (номер варианта == номеру студента по списку). Ход решения и результат отразить в отчете.

Варианты для самостоятельной работы:

1. Построение схемы управления освещением

Условие: создать схему, которая управляет освещением в комнате. Есть два входных сигнала: датчик движения и выключатель. Свет включается, если сработал датчик движения или выключатель включен.

2. Сигнализация на двери

Условие: построить схему для сигнализации на двери. Если дверь открыта и сигнализация включена, должна сработать сирена.

3. Контроль уровня воды

Условие: разработать схему, которая активирует насос для откачки воды, если уровень воды в резервуаре превышает определенный порог, но только если насос включен.

4. Система контроля температуры

Условие: создать схему управления вентилятором, который включается, если температура выше заданного уровня или если вручную активирован режим охлаждения.

5. Детектор четности

Условие: разработать схему, которая определяет, является ли данное двоичное число четным или нечетным (проверка на четность числа с помощью операции XOR).

6. Управление светофором на перекрестке

Условие: построить схему управления светофором, которая меняет сигналы в зависимости от состояний дорожных датчиков и наличия пешеходов.

7. Система аварийного оповещения

Условие: создать схему для аварийного оповещения, которая активируется, если сработали одновременно два из трех датчиков (дым, огонь, угарный газ).

8. Электронный кодовый замок

Условие: разработать схему кодового замка, который открывает дверь только при вводе правильного кода (с использованием комбинации нескольких логических операций).

9. Система резервного питания

Условие: создать схему, которая переключает питание на резервный источник, если основной источник питания отключился.

10. Автоматическое освещение с ночным режимом

Условие: построить схему автоматического освещения, которая включает свет ночью, если есть движение, но при этом не активируется днем.

11. Сигнализация автомобиля

Условие: разработать схему для автомобильной сигнализации, которая активируется, если одновременно сработают датчик удара и датчик открытия двери.

12. Каскадное освещение

Условие: создать схему для управления каскадным освещением в подъезде. Лампы включаются последовательно при активации соответствующих датчиков движения на каждом этаже.

13. Термодатчик

Условие: построить схему управления отоплением, которая включает обогреватель, если температура ниже определенного значения и активирован режим обогрева.

14. Защита от перегрева

Условие: создать схему, которая выключает систему питания устройства при перегреве или превышении допустимого уровня тока.

15. Система управления воротами

Условие: построить схему автоматического управления воротами, которая открывает ворота при наличии сигнала от датчика автомобиля или при нажатии кнопки управления.

16. Контроль доступа

Условие: разработать схему для системы контроля доступа, которая разрешает вход в помещение только при наличии правильного кода и соответствующего уровня доступа.

17. Блокировка двигателя

Условие: создать схему, которая блокирует двигатель, если одновременно активированы датчики аварийного состояния (перегрев, низкий уровень масла).

18. Контроль напряжения

Условие: построить схему, которая включает аварийное питание при понижении напряжения ниже заданного уровня.

19. Система энергосбережения

Условие: создать схему управления освещением, которая автоматически выключает свет, если в комнате никого нет, и включается снова при возвращении.

20. Сигнализация на окне

Условие: разработать схему, которая активирует сигнализацию, если открыто окно и сигнализация включена.

21. Умный дом: управление светом

Условие: создать схему управления освещением в доме, где свет включается только при наличии движения и если уровень освещенности в комнате ниже определенного порога.

22. Система обнаружения протечек

Условие: построить схему для системы обнаружения протечек, которая активирует сигнал тревоги при одновременном срабатывании датчиков воды и давления.

23. Охранная система

Условие: создать схему охранной системы, которая включает сигнализацию, если срабатывает хотя бы один из трех датчиков: движения, разбития стекла или открытия двери.

24. Термостат с двойной защитой

Условие: построить схему терmostата, который отключает систему обогрева при достижении заданной температуры, а также при срабатывании аварийного датчика перегрева.

25. Система аварийного освещения

Условие: разработать схему аварийного освещения, которая включается при отключении основного источника питания и наличии сигнала от датчика присутствия.

26. Автоматическое управление шлагбаумом

Условие: создать схему управления шлагбаумом, который поднимается, если обнаружено транспортное средство, и если шлагбаум не заблокирован.

27. Управление вентилятором

Условие: разработать схему управления вентилятором, который включается только при высокой температуре и при нажатии на кнопку активации.

28. Распознавание команд по кнопкам

Условие: создать схему, которая распознает правильную последовательность нажатий кнопок для открытия доступа к устройству.

Список литературы для самостоятельного изучения

1. Хенnessи, Дж. Л., Паттерсон, Д. А. Компьютерная архитектура: количественный подход. — 5-е изд. — М.: Вильямс, 2016. — 944 с.
2. Таненбаум, Э. Архитектура компьютера. Структурный подход. — 5-е изд. — СПб.: Питер, 2013. — 832 с.
3. Архитектура вычислительных систем [Электронный ресурс]: учебное пособие – Эл. изд. - Электрон. текстовые дан. (1 файл pdf: 77 с.). - Грейбо С.В., Новосёлова Т.Е., Пронькин Н.Н., Семёнычева И.Ф. 2019