

**МИНИСТЕРСТВО ОБЩЕГО И ПРОФЕССИОНАЛЬНОГО  
ОБРАЗОВАНИЯ РОСТОВСКОЙ ОБЛАСТИ  
ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ РОСТОВСКОЙ ОБЛАСТИ  
«РОСТОВСКИЙ-НА-ДОНУ КОЛЛЕДЖ РАДИОЭЛЕКТРОНИКИ,  
ИНФОРМАЦИОННЫХ И ПРОМЫШЛЕННЫХ ТЕХНОЛОГИЙ»  
(ГБПОУ РО «РКРИПТ»)**

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ  
ПРАКТИЧЕСКИХ (ЛАБОРАТОРНЫХ) РАБОТ  
ПО ДИСЦИПЛИНЕ**

**ОП.07 ЦИФРОВАЯ СХЕМОТЕХНИКА**

**Специальность:**

11.02.16 Монтаж, техническое обслуживание и ремонт  
электронных приборов и устройств

**Квалификация выпускника:**

Специалист по электронным приборам и устройствам

**Форма обучения:** очная

Ростов-на-Дону  
2023

СОГЛАСОВАНО

Начальник методического отдела

  
Н.В. Вострякова  
«27» марта 2023 г.

УТВЕРЖДАЮ

Заместитель директора

по учебно-методической работе

  
С.А. Будасова  
«28» марта 2023 г.

ОДОБРЕНО

Цикловыми комиссиями радиоэлектроники

и технического обслуживания

радиоэлектронной техники

Пр. № 8 от «1» февраля 2023 г.

Председатель ЦК

  
В.Ю. Махно

Методические указания по выполнению практических (лабораторных) работ разработаны в соответствии с рабочей программой учебной дисциплины ОП.07 Цифровая схмотехника для специальности 11.02.16 Монтаж, техническое обслуживание и ремонт электронных приборов и устройств.

**Разработчик(и):**

**Самойлова Т.А.** – преподаватель высшей квалификационной категории ГБПОУ РО «РКРИПТ»

**Рецензенты:**

**Калиенко И.В.** – к.т.н., преподаватель ГБПОУ РО «РКРИПТ»

**Маскаев Е.Н.** – главный конструктор АО «Алмаз»

## СОДЕРЖАНИЕ

<b>Введение</b>	4
1. Практическая работа 1	7
2. Практическая работа 2	15
3. Практическая работа 3	22
4. Практическая работа 4	25
5. Практическая работа 5	42
6. Практическая работа 6	53
7. Практическая работа 7	60
8. Практическая работа 8	65
9. Лабораторная работа 1	70
10.Лабораторная работа 2	74
11.Лабораторная работа 3	79
12. Лабораторная работа 4	83
13.Лабораторная работа 5	87
14.Лабораторная работа 6	94
15. Лабораторная работа 7	99

## Введение

Лабораторные и практические занятия по учебной дисциплине ОП.07 «Цифровая схемотехника» составляют важную часть теоретической и профессиональной практической подготовки и направлены на подтверждение теоретических положений и формирование практических умений и практического опыта. Обучающимися осваиваются следующие умения

- производить выбор элементной базы для проектирования цифровых схем;
- производить синтез и анализ цифровых схем;
- проводить исследование типовых схем цифровой электроники;
- выполнять упрощение логических схем
- использовать универсальные базисы для построения схем на логических элементах
- выполнять сравнительную характеристику ЦИМС различных типов логик, пользуясь справочной литературой
- выполнять построение счетчиков с произвольным основанием
- использовать комбинационные устройства для реализации логических функции

Лабораторные и практические занятия относятся к основным видам учебных занятий.

Выполнение студентами лабораторных и практических работ направлено:

- на обобщение, систематизацию, углубление, закрепление полученных теоретических знаний по конкретным темам дисциплин;
- формирование умений применять полученные знания на практике;
- реализацию единства интеллектуальной и практической деятельности;
- развитие интеллектуальных умений (аналитических, проектировочных, конструкторских и др.) у будущих специалистов;
- выработку при решении поставленных задач таких профессионально значимых качеств, как самостоятельность, ответственность, точность, творческая инициатива.

Ведущей дидактической целью лабораторных занятий является экспериментальное подтверждение и проверка существенных теоретических положений.

Ведущей дидактической целью практических занятий является формирование практических умений – профессиональных или учебных, необходимых в последующей учебной деятельности.

Содержанием лабораторных работ по дисциплине наблюдение развития явлений, и процессов. В ходе выполнения заданий у студентов формируются практические умения и навыки обращения с приборами,

установками, лабораторным оборудованием, аппаратурой, которые могут составлять часть профессиональной практической подготовки, а также исследовательские умения (наблюдать, сравнивать, анализировать, устанавливать зависимости, делать выводы и обобщения, самостоятельно вести исследование, оформлять результаты).

Содержанием практических занятий по дисциплине являются решение разного задач, в том числе профессиональных выполнение вычислений, расчетов.

Содержание практических, лабораторных занятий охватывают весь круг профессиональных умений, на подготовку к которым ориентирована данная дисциплина, которые в дальнейшем закрепляются и совершенствуются в процессе курсового проектирования, практикой по профилю специальности и преддипломной практикой.

Лабораторные занятия проводятся в специально оборудованных учебных лабораториях. Практическое занятие должно проводиться в учебных кабинетах или специально оборудованных помещениях (площадках). Продолжительность занятия – не менее 2-х академических часов. Необходимыми структурными элементами занятия, помимо самостоятельной деятельности студентов, являются инструктаж, проводимый преподавателем, а также организация обсуждения итогов выполнения работы.

Все студенты, связанные с работой в лаборатории, обязаны пройти инструктаж по безопасному выполнению работ, о чем расписываются в журнале инструктажа по технике безопасности.

Выполнению лабораторных и практических работ предшествует проверка знаний студентов, их теоретической готовности к выполнению задания.

Лабораторные и практические работы студенты выполняют под руководством преподавателя. При проведении лабораторных и практических занятий учебная группа может делиться на подгруппы численностью не менее 8 человек. Объем заданий для лабораторных и практических занятий спланирован с расчетом, чтобы за отведенное время они могли быть выполнены качественно большинством студентов.

Формы организации работы обучающихся на лабораторных работах и практических занятиях: фронтальная, групповая и индивидуальная.

При фронтальной форме организации занятий все студенты выполняют одновременно одну и ту же работу. При групповой форме организации занятий одна и та же работа выполняется бригадами по 2 - 5 человек. При индивидуальной форме организации занятий каждый студент выполняет индивидуальное задание.

Отчет по практической и лабораторной работе представляется в печатном виде в формате, предусмотренном шаблоном отчета по практической, лабораторной работе. Защита отчета проходит в форме доклада обучающегося по выполненной работе и ответов на вопросы преподавателя.

Оценки за выполнение лабораторных работ и практических занятий могут выставляться по пятибалльной системе или в форме зачета и учитываться как показатели текущей успеваемости студентов.

**Критерии оценки лабораторных, практических работ.**

**Оценка «5»** ставится, если учащийся выполняет работу в полном объеме с соблюдением необходимой последовательности проведения опытов и измерений; самостоятельно и рационально монтирует необходимое оборудование; все опыты проводит в условиях и режимах, обеспечивающих получение правильных результатов и выводов; соблюдает требования правил безопасности труда; в отчете правильно и аккуратно выполняет все записи, таблицы, рисунки, чертежи, графики, вычисления; правильно выполняет анализ полученных результатов.

**Оценка «4»** ставится, если выполнены требования к оценке «5», но было допущено два - три недочета, не более одной негрубой ошибки и одного недочёта.

**Оценка «3»** ставится, если работа выполнена в ходе проведения опыта и измерений были допущены грубые ошибки.

**Оценка «2»** ставится, если работа выполнена не полностью и объем выполненной части работы не позволяет сделать правильных выводов: если опыты, измерения, вычисления, наблюдения производились неправильно.

## ПРАКТИЧЕСКАЯ РАБОТА №1

ПЕРЕВОД ЧИСЕЛ ИЗ ОДНОЙ СИСТЕМЫ СЧИСЛЕНИЯ В ДРУГУЮ.

**1. Цель работы:** Получить практическое представление о способах представления чисел в цифровых системах о биполярных кодах положительных и отрицательных чисел

**2. Время выполнения работы-2час**

**3.Краткие теоретические сведения**

### ПОНЯТИЕ О СИСТЕМАХ СЧИСЛЕНИЯ

Системы счисления - это способ представления чисел и соответствующие ему правила действия над числами. Существует множество систем счисления (СС), но применительно к цифровым системам можно выделить следующие: двоичная, восьмеричная, десятичная и шестнадцатеричная, которые относятся к позиционным СС.

Позиционной системой счисления называется такая система, в которой количественное значение каждой цифры зависит от ее позиции (места) в числе. Примером можно привести обычную десятичную систему счисления. Например, число 909 содержит цифру 9 означающую девять сотен и цифру 9 в правой позиции означающую девять единиц.

Основанием системы счисления ( $p$ ) называется количество знаков или символов, используемых для изображения числа в данной системе счисления.

Количественное значение символа определяется **номером разряда ( $k$ )**, т.е. местом расположения этого символа в числовом ряду (нумерация справа налево, начиная с нуля).

В зависимости от номера разряда они имеют свои **весовые коэффициенты**.

В общем случае *весовой коэффициент*  $p^{n-1}$ ,

*где  $n$  – порядковый номер разряда*

*$p$  – основание системы счисления*

Для десятичных чисел весовые **коэффициенты** это  $10^0, 10^1, 10^2$  и т. д, **ДЕСЯТИЧНАЯ СИСТЕМА СЧИСЛЕНИЯ** имеет основание 10 и для записи числа в этой системе требуется 10 цифр от 0 до 9. В качестве примера возьмем десятичное число 529,162.

$$\begin{array}{cccccc} 5 & 2 & 9, & 1 & 6 & 2 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 10^2 & 10^1 & \cdot 10^0 & 10^{-1} & 10^{-2} & 10^{-3} \\ 529,162_{10} = 5 \cdot 10^2 + 2 \cdot 10^1 + 9 \cdot 10^0 + 1 \cdot 10^{-1} + 6 \cdot 10^{-2} + 2 \cdot 10^{-3} \\ = 500 + 20 + 9 + 0,1 + 0,06 + 0,002 \end{array}$$

Цифра 10 в подстрочном индексе числа обозначает систему счисления.

Так как в основу функционирования цифровых систем положен двоичный способ представления информации, основной системой счисления в них является двоичная система счисления.

**ДВОИЧНАЯ СИСТЕМА СЧИСЛЕНИЯ** имеет основание  $P=2$  и для записи числа в этой системе требуется 2 цифры 0 и 1. Возьмем двоичное число 101011

$$\begin{array}{cccccc} 1 & 0 & 1 & 0 & 1 & 1 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 2^5 & 2^4 & 2^3 & 2^2 & 2^1 & 2^0 \end{array}$$

$$101011_2 = 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 32 + 8 + 2 + 1 = 43_{10}$$

**ВОСЬМЕРИЧНАЯ СИСТЕМА СЧИСЛЕНИЯ** имеет основание  $P=8$  и для записи числа в этой системе требуется 8 цифр от 0 до 7

$$152_8 = 1 \cdot 8^2 + 5 \cdot 8^1 + 2 \cdot 8^0 = 64 + 40 + 2 = 106_{10}$$

Данная система является вспомогательной для ЭВМ и используется для более краткого представления двоичных чисел

**ШЕСТНАДЦАТЕРИЧНАЯ СИСТЕМА СЧИСЛЕНИЯ** имеет основание  $P=16$  и использует знаки 10 цифр 0,1,2,3,4,5,6,7,8,9 и шесть латинских букв А, В, С, D, E, F.

Дес.	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Шестн.	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>E</b>	<b>F</b>

$$1EA_{16} = 1 \cdot 16^2 + 14 \cdot 16^1 + 10 \cdot 16^0 = 1 \cdot 256 + 14 \cdot 16 + 10 \cdot 1 = 490_{10}$$

Данная система счисления, как и восьмеричная, является вспомогательной. Запись двоичного числа в шестнадцатеричной системе счисления сокращает количество разрядов в 4 раза.

#### **ДВОИЧНО-ДЕСЯТИЧНАЯ СИСТЕМА**

При хранении десятичных чисел каждая цифра десятичного числа представляется в двоичной форме путем замены каждого разряда десятичного числа четырехразрядным двоичным кодом. Такая форма представления десятичных чисел называется двоично-десятичной. Например, число  $328,7_{10}$  в двоично-десятичной форме запишется следующим образом:

$$328,7_{10} = 0011\ 0010\ 1000,0111_{2-10}$$

Так называемый код 8-4-2-1 относится к взвешенным кодам. Цифры в названии кода означают вес единиц в соответствующих двоичных разрядах. Он соответствует первым десяти комбинациям натурального двоичного кода.

Число в десятичном коде	Двоично-десятичный код 8-4-2-1			
	$2^3$ <b>8</b>	$2^2$ <b>4</b>	$2^1$ <b>2</b>	$2^0$ <b>1</b>
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0

5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

Двоично-десятичная система используется в ЭВМ не только как вспомогательная система счисления при вводе и выводе данных, но и как основная при решении задач с большим количеством десятичных чисел.

Непозиционной системой счисления называется такая система, в которой количественное значение каждой цифры не зависит от занимаемой ею позиции в изображении числа, а определяется лишь самим символом (цифрой). Например, в римской системе счисления число XX (двадцать) содержит символ X, который означает 10 единиц не зависимо от позиции.

### БИПОЛЯРНЫЕ КОДЫ

В двоичной системе счисления для представления знака числа используется дополнительный знаковый разряд (один или несколько), который располагается перед старшим числовым разрядом. Для положительных чисел значение знакового разряда – 0, для отрицательных – 1. Двоичный код числа со знаком называют также **прямым кодом**. При этом знаковым разрядом является старший разряд числа

Примеры в восьмиразрядной сетке:

Изображаемое число	Прямой код							
+1101 (+13)	0	0	0	0	1	1	0	1
+101110 (+93)	0	1	0	1	1	1	0	1
- 1101 (-13)	1	0	0	0	1	1	0	1

Прямой код используется при хранении чисел в памяти ЭВМ, а также при выполнении операций умножения и деления.

**Обратный код** отрицательного числа получается путем замены всех “0” на “1” и всех “1” на “0” прямого кода. Причем, знаковый разряд при этом остается неизменным. Замена “0” на “1” называется инвертированием (также и замена “1” на “0”). Обратный код для положительного числа совпадает с прямым.

Примеры в восьмиразрядной сетке:

Изображаемое число	Обратный код							
+1101 (+13)	0	0	0	0	1	1	0	1
+1011101 (+93)	0	1	0	1	1	1	0	1
- 1101 (-13)	1	1	1	1	0	0	1	0

Дополнительный код положительного числа совпадает с прямым, а код отрицательного числа образуется как **результат увеличения на 1 его обратного кода**. Процесс получения дополнительного кода отрицательного числа можно разбить на два этапа – получить вначале обратный код, а затем из него построить дополнительный.

Примеры в восьмиразрядной сетке:

Число	->	-101101
Прямой код	->	1101101
Обратный код	->	1010010
		+1
Дополнительный	->	1010011

Для восстановления прямого кода числа из дополнительного нужно полностью повторить (и именно в том же порядке!) действия, которые использовались при переводе из прямого в дополнительный код: сначала все цифры, кроме цифры, изображающей знак, заменить на противоположные, а затем прибавить 1.

Для обнаружения переполнения разрядной сетки в ЭВМ применяются модифицированные прямой, обратный и дополнительный коды. В этих кодах знак кодируется двумя разрядами, причем знаку "плюс" соответствует комбинация 00, а знаку "минус" - комбинация 11.

**Задание:** Произвести перевод чисел, заданных преподавателем, из одной формы в другую выполнить запись заданных чисел в прямом, обратном и дополнительном двоичных кодах для восьмиразрядной ячейки и для 16-разрядной ячейки

#### 4. Порядок выполнения работы

4.1. Внести числа, заданные преподавателем в таблицу

№ вар	Номер пункта задания					
	3.3.1	3.3.2	3.3.3	3.3.4	3.3.5	3.3.6

4.2. Изучить краткие теоретические сведения.

4.3. Выполнить следующие действия

4.3.1. Перевести заданное десятичное число в двоичную, восьмеричную и шестнадцатеричную системы счисления.

4.3.2. Перевести в десятичную систему счисления заданное двоичное, восьмеричное и шестнадцатеричное число.

4.3.3. Перевести в заданное двоичное число в восьмеричную и шестнадцатеричную системы счисления.

4.3.4. Перевести в двоичную систему счисления заданное восьмеричное и шестнадцатеричное число.

4.3.5. Перевести в двоично-десятичную систему счисления заданное десятичное число

4.3.6. Перевести заданное десятичное число в двоичную систему счисления и записать их в прямом, обратном и дополнительном двоичных кодах для восьмиразрядной ячейки и для 16-разрядной ячейки. Полученный в 16-разрядном коде результат преобразовать в шестнадцатеричную форму.

4.3.7. Ответить на контрольные вопросы

#### 4.4. Оформить отчет и сдать его преподавателю

### 5. Указания к выполнению работы

#### 5.1. Перевод числа в десятичной системе счисления в двоичную, восьмеричную и шестнадцатеричную

При переводе целого десятичного числа в систему с основанием  $P$  его необходимо последовательно *делить* на  $P$  до тех пор, пока не останется остаток, меньший основания. Число в системе с основанием  $P$  записывается как последовательность остатков от деления, записанных в обратном порядке, начиная с последнего частного.

При переводе дробной части она умножается на основание, после чего целая часть запоминается и отбрасывается. Вновь полученная дробная часть умножается на основание и т.д. Процедура продолжается до тех пор, пока дробная часть не станет равной нулю. Целые части выписываются после двоичной запятой в порядке их получения. Результатом может быть либо конечная, либо периодическая двоичная дробь. Поэтому, когда дробь является периодической, следует оборвать умножение на каком-либо шаге.

#### Перевод числа из десятичной СС в двоичную

Пример: Перевести число  $234,35_{10}$  в систему счисления с основанием 2.

##### Перевод целой части:

Будем делить число 234 последовательно на 2 и записывать остатки, не забывая нулевые:

$$234 : 2 = 117 \text{ остаток } \mathbf{0}$$

$$117 : 2 = 58 \text{ остаток } \mathbf{1}$$

$$58 : 2 = 29 \text{ остаток } \mathbf{0}$$

$$29 : 2 = 14 \text{ остаток } \mathbf{1}$$

$$14 : 2 = 7 \text{ остаток } \mathbf{0}$$

$$7 : 2 = 3 \text{ остаток } \mathbf{1}$$

$$3 : 2 = \mathbf{1} \text{ остаток } \mathbf{1}$$

Результат последнего деления на 2 уже не делится, и эта цифра будет старшей цифрой нашего числа. Выписав все остатки, начиная с последнего, получим двоичное представление целой части числа:

$$234_{10} = 11101010_2$$

##### Перевод дробной части:

	0,35
	$\times 2$
0←	$\underline{\mathbf{0},7}$
	0,7
	$\times 2$
1←	$\underline{\mathbf{1},4}$
	0,4
	$\times 2$
0←	$\underline{\mathbf{0},8}$

$$1 \leftarrow \begin{array}{r} 0,8 \\ \times 2 \\ \hline 1,6 \end{array}$$

Дробь периодическая, поэтому остановимся на **четырёх** значащих цифрах после запятой. Результат записывается, начиная с целой части первого действия умножения.

$$0,35_{10} = 0,0101_2$$

Ответ:  **$234,35_{10} = 11101010,0101_2$**

### Перевод числа из десятичной СС в восьмеричную

Пример: Перевести число  **$100,35_{10}$**  в систему счисления с основанием 8

Перевод целой части:

$$100:8=12 \text{ - остаток } 4.$$

$$12:8=1, \text{ остаток } 4.$$

$$100_{10} = 144_8.$$

Перевод дробной части:

$$\begin{array}{r} 2 \leftarrow \begin{array}{r} 0,35 \\ \times 8 \\ \hline 2,80 \end{array} \\ \hline 6 \leftarrow \begin{array}{r} 0,7 \\ \times 8 \\ \hline 6,40 \end{array} \\ \hline 3 \leftarrow \begin{array}{r} 0,4 \\ \times 8 \\ \hline 3,20 \end{array} \end{array}$$

Очевидно, что дробь является периодической. Поэтому остановимся на **трех** знаках после запятой.

$$0,35_{10} = 0,263_8$$

Ответ:  **$100,35_{10} = 144,263_8$**

### Перевод числа из десятичной СС в шестнадцатеричную

Пример: Перевести число  **$426,35_{10}$**  в систему счисления с основанием 16

Решение:

Перевод целой части

$426:16=26$  - остаток 10, что соответствует А в шестнадцатеричной СС.

$26:16=1$  - остаток 10, что соответствует А в шестнадцатеричной СС.

$$426_{10} = 1AA_{16}$$

Перевод дробной части:

$$\begin{array}{r|l} & 0,35 \\ & \times 16 \\ 5 \leftarrow & \underline{5,60} \\ \hline & 0,60 \\ & \times 16 \\ 9 \leftarrow & \underline{9,60} \end{array}$$

Очевидно, что дробь является периодической. Поэтому остановимся на **двух** знаках после запятой

$$0,35_{10} = 0,59_{16}$$

Ответ:  $426,35_{10} = 1AA,59_{16}$

## 5.2. Перевод числа из двоичной, восьмеричной и шестнадцатеричной систем счисления в десятичную

При переводе чисел из системы счисления с основанием  $P$  в десятичную систему счисления необходимо пронумеровать разряды целой части справа налево, начиная с нулевого, и дробной части, начиная с разряда сразу после запятой, слева направо (начальный номер 1). Затем вычислить сумму произведений соответствующих значений разрядов на основание системы счисления в степени, равной номеру разряда. Это и есть представление исходного числа в десятичной системе счисления.

Пример: Перевести число  $101100,01_2$  в систему счисления с основанием  $10$

Решение:

$$101100,01_2 = 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 + 0 \cdot 2^{-1} + 1 \cdot 2^{-2} = 32 + 8 + 4 + 0,25 = 44,25_{10}$$

Пример: Перевести число  $52,43_8$  в систему счисления с основанием  $10$

Решение:

$$52,43_8 = 5 \cdot 8^1 + 2 \cdot 8^0 + 4 \cdot 8^{-1} + 3 \cdot 8^{-2} = 40 + 2 + 0,5 + 0,046875 = 42,546875_{10}$$

Пример: Перевести число  $1E,A_{16}$  в систему счисления с основанием  $10$

Решение:

$$1E,A_{16} = 1 \cdot 16^1 + 14 \cdot 16^0 + 10 \cdot 16^{-1} = 16 + 14 + 0,625 = 30,625_{10}$$

## 5.3. Перевод числа из двоичной системы счисления в восьмеричную и шестнадцатеричную

При преобразовании двоичного кода в восьмеричный или в шестнадцатеричный, двоичный код делится соответственно на триады (тетрады) справа налево. Затем триады (тетрады) заменяются восьмеричными (шестнадцатеричными) цифрами

Пример: Перевести в восьмеричную систему счисления двоичное число  $1001100101,011$

Решение:

$$1001100101,011_2 = 001\ 001\ 100\ 101,011 = 1145,3_8$$

Пример: Перевести в шестнадцатеричную систему счисления двоичное число 100110010,1011

**Решение:**

$$100110010,1011_2 = 0001\ 0011\ 0010, 1011 = 132,В_{16}$$

#### **5.4. Перевод числа из восьмеричной и шестнадцатеричной систем счисления в двоичную**

При преобразовании восьмеричного кода числа в двоичный, необходимо его каждую восьмеричную цифру заменить соответствующим трёхзначным двоичным кодом (триадой).

Пример: Перевести восьмеричное число 531,6 в двоичную систему счисления

**Решение:**

$$531,6_8 = 101\ 011\ 001,110_2$$

При преобразовании шестнадцатеричного кода числа в двоичный, необходимо каждую шестнадцатеричную цифру заменить четырёхзначным двоичным кодом (тетрадой).

Пример: Перевести шестнадцатеричное число 83В,Ев двоичную систему счисления

**Решение:**

$$83В,Е_{16} = 1000\ 0011\ 1011,1110_2$$

#### **5.5. Перевод числа из десятичной системы счисления в двоично-десятичную**

В двоично - десятичной (2-10) системе каждая десятичная цифра представляется четырехразрядным двоичным эквивалентом, представленном в коде 8-4-2-1.

Пример: Перевести из десятичной системы счисления в двоично-десятичную число 836,5

**Решение:**

$$836,5_{10} = 1000\ 0011\ 0110,0101_2$$

#### **5.6. Запись двоичных чисел в прямом, обратном и дополнительном кодах**

Положительное число в обратном и дополнительном коде остается без изменения только в знаковой части ставится 0 (или 00 в модифицированном коде).

Для образования обратного кода отрицательного числа необходимо в знаковом разряде поставить 1 (или 11 в модифицированном коде) цифровые разряды инвертировать (заменить 1 на 0, а 0 на 1).

Для образования дополнительного кода нужно к обратному коду прибавить 1.

Пример: Получить дополнительный код числа -117 для 8- и 16-разрядной ячейки.



## АРИФМЕТИЧЕСКИЕ ДЕЙСТВИЯ С ДВОИЧНЫМИ ЧИСЛАМИ

**1. Цель работы:** Получить практическое представление о методике выполнения арифметических операций над двоичными числами

**2. Время выполнения работы-4час**

**3. Краткие теоретические сведения**

ФОРМА ПРЕДСТАВЛЕНИЯ ЧИСЕЛ И РАЗМЕЩЕНИЕ ИХ В РАЗРЯДНОЙ СЕТКЕ

В цифровых системах применяются две формы представления двоичных чисел:

-естественная форма, или форма с фиксированной запятой (точкой) в виде последовательности цифр с постоянным для всех чисел положением запятой (точкой), отделяющей целую часть от дробной

- нормализованная форма с плавающей запятой (точкой)

При представлении чисел с *фиксированной запятой* каждому разряду ячейки памяти соответствует всегда один и тот же разряд числа, а "запятая" "находится" справа после младшего разряда, то есть вне разрядной сетки. Для кодирования знака двоичного числа используется старший разряд (если число положительное он равен нулю, если отрицательное - единице). Целая часть числа располагается в одной разрядной линии, причем заполнение производится с младшего разряда, а дробная часть – в другой разрядной линии и заполнение происходит с разряда, следующего за знаковым разрядом. Целая часть числа дополняется, если это необходимо, слева нулями до соответствующего числа разрядов, а дробная часть дополняется нулями справа.

$$+45,25_{10} = 101101,010_2$$

Целая часть числа:

Знак числа	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	1

Дробная часть числа:

Знак числа	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

$$\lfloor 45,25_{10} = 101101,010_2$$

Целая часть числа

Знак числа	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	0	0	0	0	0	0	0	0	1	0	1	1	0	1

Дробная часть числа:

Знак числа	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

Вещественные числа хранятся и обрабатываются в компьютере в формате с плавающей запятой

Формат чисел с плавающей запятой базируется на экспоненциальной форме записи:

$$A = m \times q^n,$$

где  $m$  – мантисса числа,

$q$  – основание системы счисления,

$n$  – порядок числа.

Для однозначности представления чисел с плавающей запятой используется нормализованная форма, при которой мантисса отвечает условию:

$$q^{-1} \leq |m| < 1, \text{ т. е. число называется}$$

нормализованным, если мантисса меньше единицы (**правильная дробь**) и после запятой следует первая значащая цифра «1». Если же после запятой следует ноль, то число называется **ненормализованным**.

Представление чисел в нормализованном виде позволяет иметь в разрядной сетке большее число значащих цифр и, следовательно, повышает точность вычислений. Если в процессе вычислений получаются ненормализованные числа, то они автоматически нормализуются самой машиной. Нормализация чисел осуществляется путем сдвига мантиссы с соответствующим изменением порядка. Если  $p$  старших разрядов мантиссы равны нулю, то нормализация состоит в сдвиге мантиссы на  $p$  разрядов влево и уменьшении порядка на  $p$  единицу. При этом в младшие  $p$  разрядов мантиссы записываются нули, и величина числа при этом не меняется.

**Например,** двоичное число 101101,010 в формате с плавающей запятой

$$101101,010_2 = 0,101101010 \times 10^{110}$$

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	1	1	0	1	0	1	1	0	1	0	1
Знак числа	Знак порядка	порядок						мантисса							

Младший разряд в этом примере теряется при нормализации вправо.

## АРИФМЕТИЧЕСКИЕ ОПЕРАЦИИ НАД ДВОИЧНЫМИ ЧИСЛАМИ

### Правило сложения двоичных чисел

$$0+0=0$$

$$0+1=1$$

$$1+0=1$$

$$1+1=10$$

↑  
Единица переноса.

Сложение двух многоразрядных чисел проводится поразрядно с учетом единиц переноса от предшествующих разрядов. Явление переноса состоит в том, что при сложении цифр двух чисел, имеющих равные порядки, результат сложения в последующем старшем разряде должен быть увеличен на единицу.

Пример сложения многоразрядных чисел

$$\begin{array}{r}
 \phantom{+} 18 = 010010 \\
 + 23 = 010111 \\
 \hline
 41 = 101001
 \end{array}$$

1 1 1 – перенос 1 в старшие разряды

### Правило вычитания двух двоичных чисел

$$0-0=0$$

$$1-0=1$$

$$1-1=0$$

$$10-1=1|$$

← Единица заёма из старшего разряда

### Умножение двоичных чисел

$$\begin{array}{r}
 10101 \\
 \times 1001 \\
 \hline
 10101 \\
 + 10101 \\
 \hline
 10111101 \\
 21 \times 9 = 189
 \end{array}$$

$$\begin{array}{r}
 1101 \\
 \times 11 \\
 \hline
 1101 \\
 + 1101 \\
 \hline
 100111 \\
 13 \times 3 = 39.
 \end{array}$$

Если при поразрядном вычитании приходится вычитать из нуля в уменьшаемом единицу в вычитаемом, то делается заем из старшего разряда, в котором есть значащая цифра — единица.

При выполнении арифметических операций в цифровых системах применяют **прямой, обратный** и **дополнительный** коды чисел, которые уже применялись при выполнении Практической работы №1 для представления **целых** положительных и отрицательных чисел.

Как вы уже знаете, **прямой код** числа это двоичный код этого числа со знаком.

Операция вычитания в цифровых системах реализуется с помощью операции сложения. Вычитаемое при этом представляется в дополнительном коде, а если расчет не требует высокой точности - в обратном коде.

**Обратный код** отрицательного числа получается путем замены всех “0” на “1” и всех “1” на “0” прямого кода. Причем, знаковый разряд при этом остается неизменным. Обратный код для положительного числа совпадает с прямым.

Для получения **дополнительного кода** числа необходимо к обратному коду прибавить единицу. Дополнительный код для положительного числа совпадает с прямым. Для перевода числа из дополнительного кода обратно в прямой код необходимо сначала инвертировать его, а затем прибавить единицу.

При выполнении арифметических операций над двоичными числами с **плавающей запятой** вначале необходимо **выровнять порядки чисел до большего**.

### **Задание**

- выполнить операции сложения над двоичными числами А и В с фиксированной и плавающей запятой.

-выполнить операцию вычитания над числами А и В в обратном и дополнительном кодах

### **4. Порядок выполнения**

4.1. Внести числа, заданные преподавателем в отчет

4.2. Изучить краткие теоретические сведения.

4.3. Перевести заданное десятичное число в двоичную систему счисления и записать в формате с фиксированной и с плавающей запятой.

4.4. Выполнить над полученными двоичными числами операции сложения в формате с фиксированной и с плавающей запятой. Проверить полученный результат (перевести в десятичную систему).

4.5. Выполнить над полученными двоичными числами операции вычитания в в формате с плавающей запятой в обратном и дополнительном кодах. Проверить полученный результат (перевести в десятичную систему).

4.6. Ответить на контрольные вопросы

4.7. Оформить отчет и сдать его преподавателю

## 5. Указания к выполнению работы

**5.1. Представить заданные числа, согласно варианту, в двоичной форме.** Исходные данные для примера представляют собой два числа в десятичной системе счисления, над которыми необходимо произвести арифметических операции.

Допустим, дано два числа  $A_{10}=60,35$ ;  $B_{10}=30,11$ .

Выполним перевод чисел в двоичную форму с точностью до четырех знаков после запятой.

Получим  $A = 111100,0101_2$   $B=11110,0001_2$

**5.2. Выполнить операции сложения над двоичными числами A и B с фиксированной и плавающей запятой.**

**5.2.1. Сложение двух многоразрядных чисел с фиксированной запятой**

Сложение двух многоразрядных чисел с фиксированной запятой проводится поразрядно с учетом единиц переполнения от предшествующих разрядов.

$$\begin{array}{r} \text{Для чисел A и B} \\ \phantom{+} 1111 \\ +111100,0101 \\ \hline \phantom{+} 11110,0001 \\ \hline 1011010,0110 \end{array}$$

**Выполнить проверку.** В сумме  $A+B=90,46_{10}$ . Полученная двоичная форма совпадает с десятичным эквивалентом (достаточно проверки целой части числа).

**5.2.2 Сложение двух многоразрядных чисел с плавающей запятой**

При выполнении арифметических операций над двоичными числами с плавающей запятой вначале необходимо нормализовать эти числа, затем **выровнять порядки чисел до большего.**

**5.2.2.1. Нормализовать заданные числа и выровнять порядки чисел до большего.**

$$\begin{aligned} A &= 111100,0101 = 0,1111000101 \cdot 10^{110} \\ B &= 11110,0001 = 0,111100001 \cdot 10^{101} = 0,0111100001 \cdot 10^{110} \end{aligned}$$

**5.2.2.2. Сложить поразрядно с учетом единиц переполнения от предшествующих разрядов. Результат нормализовать.**

$$\begin{array}{r} +0,1111000101 \\ \phantom{+} 0,0111100001 \\ \hline 1,0110100110 \cdot 10^{110} = 0,10110100101 \cdot 10^{111} \end{array}$$

В примере результат представляет собой неправильную дробь. Для нормализации следует **увеличить порядок** на единицу.

Выполните проверку. В сумме  $A+B=90,46_{10}$ . Полученная двоичная форма совпадает с десятичным эквивалентом (достаточно проверки целой части числа).  $90_{10}=1011010_2$ .

### 5.3. Выполнить операцию вычитания над числами А и В в обратном и дополнительном кодах.

Для операции вычитания необходимо выполнить следующее преобразование:

$A - B = A + (-B)$ , используя обратный и дополнительный коды числа В.

#### 5.3.1. Нахождение обратного и дополнительного кодов числа (-В)

Для образования обратного кода отрицательного числа необходимо в знаковом разряде поставить 1, а цифровые разряды инвертировать (заменить 1 на 0, а 0 – на 1).

Возьмем нормализованное значение числа В ( $V_{пр}$ ) и получим обратный код числа В (уменьшаемого)  $-V_{обр}$

$$V_{пр} = 0,0111100001 \cdot 10^{110}$$

$$-V_{обр} = 1,1000011110 \cdot 10^{110}$$

Для образования дополнительного кода отрицательного числа необходимо прибавить 1 к младшему разряду обратного кода числа

$$-V_{доп} = +1,1000011110$$

$$\begin{array}{r} 1 \\ \hline 1,1000011111 \cdot 10^{110} \end{array}$$

#### 5.3.2 Вычитание (сложение) чисел А и В в обратном коде

При сложении чисел в обратном коде разряды значащих чисел складываются поразрядно справа налево, а знаковые разряды складываются как разряды целых чисел. Единица переполнения, которая получается в знаковом разряде, **прибавляется к младшему разряду суммы**. Последнее действие носит название **циклического переноса**.

$$A_{пр} = 0,1111000101 \cdot 10^{110}$$

$$-V_{обр} = 1,1000011110 \cdot 10^{110}$$

$$\begin{array}{r} + 0,1111000101 \\ \underline{1,1000011110} \\ 10,0111100011 \\ \downarrow \rightarrow \underline{\hspace{1.5cm} 1} \\ 0,0111100100 \cdot 10^{110} = 0,111100100 \cdot 10^{101} \end{array}$$

В примере результат представляет собой правильную дробь, но после запятой следует ноль, а не единица. Для нормализации следует **уменьшить порядок** на единицу.

Выполните проверку. В примере разность  $A - B = 30,24_{10}$ . Полученная двоичная форма совпадает с десятичным эквивалентом (достаточно проверки целой части числа).  $30_{10} = 11110_2$ .

#### 5.3.3 Вычитание (сложение) чисел А и В в дополнительном коде

При сложении чисел в дополнительном коде значащие разряды складываются поразрядно, знаковые разряды – как разряды целых чисел, а единица переполнения, которая получается в знаковом разряде, не учитывается (отбрасывается). Именно поэтому операция вычитания с

использованием только обратного кода приводит к ошибке, определяемой единицей в младшем разряде, и при точных расчетах не применяется.

$$A_{\text{пр}} = 0,1111000101 \cdot 10^{110}$$

$$-B_{\text{доп}} = 1,1000011111 \cdot 10^{110}$$

$$\begin{array}{r} +0,1111000101 \\ \underline{1,1000011111} \\ 10,0111100100 \cdot 10^{110} = 0,0111100100 \cdot 10^{110} = \\ 0,111100100 \cdot 10^{101} \end{array}$$

В примере результат представляет собой правильную дробь, но после запятой следует ноль, а не единица. Для нормализации следует **уменьшить порядок** на единицу.

Выполните проверку. В примере разность  $A-B=30,24_{10}$ . Полученная двоичная форма совпадает с десятичным эквивалентом (достаточно проверки целой части числа).  $30_{10}=11110_2$ .

Если знаковый разряд результата равен нулю, то полученное число представлено в прямом коде. Если в знаковом разряде единица, то результат **отрицательный** и представлен в обратном коде или в дополнительном коде и его необходимо преобразовать в прямой код путем замены 1 на 0 и 0 на 1.

## 6. Указания к выполнению отчета

Отчет должен содержать тему и цель работы, задание и порядок выполнения, результаты выполнения задания под номерами, соответствующими заданию.

## 7. Контрольные вопросы

1. Какие формы представления чисел применяются в цифровых системах?
2. Что представляет собой нормализованная форма числа с плавающей запятой?
3. Как получить обратный код отрицательного числа?
4. Как получить дополнительный код отрицательного числа?

## 8. Список литературы

1. Келим Ю. М. Вычислительная техника/ Учеб. пособие для студ. СПО.– М.: Издательский центр «Академия», 2023г.
2. Вычислительная техника : учеб. пособие / Т.Л. Партыка, И.И. Попов. — 3-е изд., перераб. и доп. -М.: ФОРУМ:ИНФРА-М, 2023- электронный портал znanium.

## ПРАКТИЧЕСКАЯ РАБОТА №3

### ПОСТРОЕНИЕ ТАБЛИЦ ИСТИННОСТИ ДЛЯ ЗАДАННЫХ ЛОГИЧЕСКИХ ФУНКЦИЙ

**1.Цель работы:** Получить практическое представление о способах представления, преобразования логических функций и их реализации.

**2.Время выполнения работы-2 час.**

**3. Краткие теоретические сведения**

Для описания функционирования цифровых аппаратных средств используют алгебру логики. Создателем алгебры логики является английский математик Джордж Буль, в честь которого она названа булевой алгеброй.

Логической (или двоичной) переменной в булевой алгебре называется переменная, которая может принимать только два значения – «0» или «1». Соответственно, логической (или двоичной) функцией называется логическая переменная, значение которой зависит от других логических переменных. Элементарная логическая функция содержит одну логическую операцию, а основе элементарных логических функций (или логических операций) строятся логические выражения

Логические функции могут быть представлены (задаться) различными способами

-аналитическая форма представления логических функций, в виде аналитических выражений (формул)

-табличная форма (при помощи таблицы истинности) ТИ. ТИ- таблица, которая содержит значения логической функции при всех возможных сочетаниях значений двоичных переменных. Количество таких сочетаний  $N = 2^n$ , где  $n$  – количество двоичных переменных, от которых зависит значение логической функции. А сами эти комбинации представляют собой двоичные коды номеров строк, начиная с нуля. Например, если логическая функция  $y = f(x_1, x_2, x_3)$  зависит от трёх логических переменных ( $n = 3$ ) –  $x_1, x_2, x_3$  – то количество различных сочетаний значений  $x_1, x_2, x_3$  (строк таблицы истинности)  $N = 2^3 = 8$ .

-в виде схем на логических элементах

#### **Порядок выполнения логических операций в логических выражениях**

1. Операция инверсии и действия в скобках
2. Логическое умножение
3. Логическое сложение

Например:

Если  $a = 1, b = 0$ , а функция имеет вид  $y = a \cdot b + (a + b)$ , то значение функции определим как  $y = 1 \cdot 0 + (1 + 0) = 0 + 1 = 1$ .

Если знак инверсии стоит над какой то частью логического выражения, то это равнозначно тому, что эта часть заключена в скобки

**Задание:** Перейти от аналитической формы задания логической функции к табличной

#### 4. Порядок выполнения

**4.1.** Найдите значение заданной в варианте задания (табл. 1) логической функции при всех комбинациях входных переменных и заполните таблицу истинности заданной функции.

#### 5. Указания к выполнению

5.1. При нахождении значений функции при указанных значениях входных переменных, следует использовать основные законы алгебры логики.

Порядок выполнения логических операций в логических выражениях

1. Операция инверсии
2. Логическое умножение
3. Логическое сложение

Для изменения порядка следует применять скобки.

Если знак инверсии (черта) стоит над какой-то частью логического выражения, то это равнозначно тому, что эта часть заключена в скобки.

Например, пусть задана логическая функция:

$$y = \overline{x_1 + x_2} \cdot x_3 + x_2 \cdot \overline{x_3}$$

Необходимо определить значение  $y$  при  $x_1 = 1$ ;  $x_2 = 1$ ;  $x_3 = 0$ .

Перепишем выражение в следующем виде:

$$y = \overline{(x_1 + x_2)} \cdot x_3 + x_2 \cdot \overline{(x_3)}$$

Подставим вместо  $x_1$ ,  $x_2$ ,  $x_3$  их значения:

$$y = \overline{(1 + 1)} \cdot 0 + 1 \cdot \overline{(0)}$$

Вначале выполняются действия в скобках:

$$y = \overline{(1)} \cdot 0 + 1 \cdot 1$$

$$y = 0 \cdot 0 + 1 \cdot 1$$

Затем выполняются операции логического умножения:

$$y = 0 + 1$$

И, наконец, операция логического сложения:

$$y = 1.$$

Найдите значения функции

$$f(0, 0, 0, 0)$$

$$f(0, 0, 0, 1)$$

$$f(0, 0, 1, 0)$$

$$f(0, 0, 1, 1)$$

$$f(0, 1, 0, 0)$$

.....

$$f(1, 1, 1, 1)$$

и заполните таблицу истинности

a	b	c	d	f
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

### 6. Указания к выполнению отчета

Отчет должен содержать:

1. Тему и цель работы, задание и порядок выполнения,
2. Исходную функцию задания п.3.1
4. Вычисления значений функции для всех комбинаций входных переменных
3. Заполненную таблицу истинности функции

### 7. Контрольные вопросы

1. Приведите три формы представления логических функций.
2. От чего зависит количество комбинаций (строк) в таблице истинности логической функции? Приведите формулу для расчета
3. Каков порядок выполнения логических операций в логических выражениях?

### 8. Список литературы

1. Келим Ю. М. Вычислительная техника/ Учеб. пособие для студ. СПО.– М.: Издательский центр «Академия», 2019г.

2. Вычислительная техника: Учебное пособие / Душкин А.В., Ланкин О.В., Чекризов Р.В. - Воронеж: Воронежский институт ФСИН России, 2015. - электронный портал znanium.

3. Ю.В. Новиков, Скоробогатов Основы микропроцессорной техники. – 3-е изд., испр. – М.: Интернет-Университет информационных технологий; БИНОМ. Лаборатория знаний, 2006г.

## ПРАКТИЧЕСКАЯ РАБОТА №4 СРАВНЕНИЕ ЦИМС РАЗЛИЧНЫХ СЕРИЙ ПО ПАРАМЕТРАМ

**1.Цель работы:** Научиться определять параметры логических ИМС по справочной литературе, сравнивать БЛЭ по быстродействию, помехоустойчивости, нагрузочной способности, потребляемой мощности и другим параметрам, приобретение навыков анализа работы принципиальных схем БЛЭ интегральных микросхем.

### **2. Время выполнения работы-4час**

### **3.Краткие теоретические сведения**

Основные типы логик цифровых интегральных микросхем

-ТЛНС – транзисторная логика с непосредственной связью между логическими элементами;

-РТЛ – резистивно-транзисторная логика;

-РЕТЛ – резистивно-ёмкостная транзисторная логика;

-ДТЛ – диодно-транзисторная логика;

-ТТЛ – транзисторно-транзисторная логика;

-ТТЛШ – транзисторно-транзисторная логика с диодами Шоттки;

-ЭСЛ – эмиттерно-связанная логика;

-КМОП – с комплементарными МОП-транзисторами;

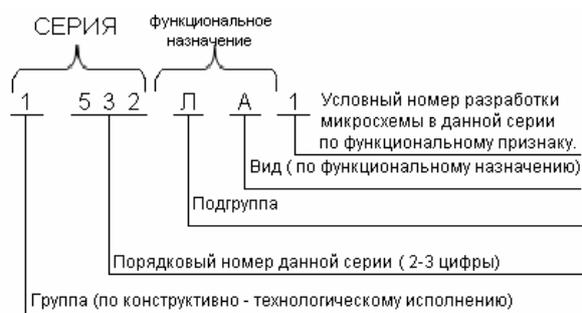
-ИСЛ – истоково-связанная логика;

-И<sup>2</sup>Л – интегральная инжекционная логика.

Для удобства разработчиков аппаратуры по технологическим, схемотехническим признакам ЦИМС выпускаются сериями.

Серия – это совокупность ИС различного функционального назначения, имеющих общие электрические и эксплуатационные характеристики, выполненных по единой технологии и объединенных одним конструктивным решением (вид корпуса). ИМС одной серии согласованы по входным и выходным уровням сигнала, имеют, как правило, одинаковое напряжение питания. Все ИС серии можно совместно применять. Функционально полная серия обычно содержит в своем составе несколько десятков типов ИС, выполняющих различные логические и арифметические операции и представляющих собой как логические элементы (И, ИЛИ, ИЛИ-НЕ и т.д.), так и целые узлы и блоки аппаратуры (регистры, счетчики, дешифраторы и т.д.)

По принятой системе обозначение микросхемы должно состоять из четырёх элементов



Первый элемент – цифра, соответствующая конструктивно-технологической группе:

- 1,5,7 – полупроводниковые ИС;
- 7 – бескорпусные;
- 2,4,8 –гибридные ИС;
- 3 – ИС, выполненные по совмещенной технологии (пленочные);
- 6 – полупроводниковые ИС на арсениде галлия.

Второй элемент – 2 или 3 – цифры, присвоенные данной серии.

Таким образом, первые два элемента определяют полный номер микросхемы.

Третий и четвертый элементы соответствуют подгруппе и виду ИС.

Пятый элемент – порядковый номер разработки микросхемы в данной серии, в которой может быть несколько одинаковых по функциональному признаку микросхем. Он может состоять как из одной цифры, так и из нескольких.

Для микросхем широкого применения в начале обозначения ставится буква К: К133ЛА1. Материал и тип корпуса обозначают буквой перед цифровым обозначением се-рии: Р – пластмассовый, М – металлокерамический, Е – металлополимерный и т.д. (может отсутствовать).

Условные обозначения подгрупп и видов логических элементов:

ЛИ – схемы И; ЛЛ – ИЛИ; схемы НЕ – ЛН; схемы И-НЕ/ИЛИ-НЕ – ЛБ; схемы И-ИЛИ – ЛС; схема И-НЕ – ЛА; ИЛИ-НЕ – ЛЕ; прочие – ЛП.

Параметры ЦИМС подразделяются на статические и динамические.

Статические параметры характеризуют работу ИМС при статических 0 или 1 на входе и выходе.

К статическим параметрам относятся:

1. Напряжение источника питания  $U_{ип}$ .
2. Входные и выходные напряжения логического нуля и логической единицы:  $U_{вх0}$ ,  $U_{вх1}$ ,  $U_{вых0}$ ,  $U_{вых1}$ .

3. Входные и выходные токи логического нуля и логической единицы:  $I_{вх0}$ ,  $I_{вх1}$ ,  $I_{вых0}$ ,  $I_{вых1}$ .

4. Коэффициент разветвления показывает количество входов микросхем нагрузок, которые можно подключить к данной микросхеме без потери её работоспособности (характеризует нагрузочную способность ИМС):  $K_p$ .

5. Коэффициент объединения по входу  $K_{об}$  показывает, количество

входов микросхемы, по которым реализуется выполняемая ею функция.

6. Напряжение статической помехи – это максимально допустимое статическое напряжение на входе, при котором микросхема не теряет своей работоспособности. Характеризует помехоустойчивость ИМС. Обозначение: Уст.п.

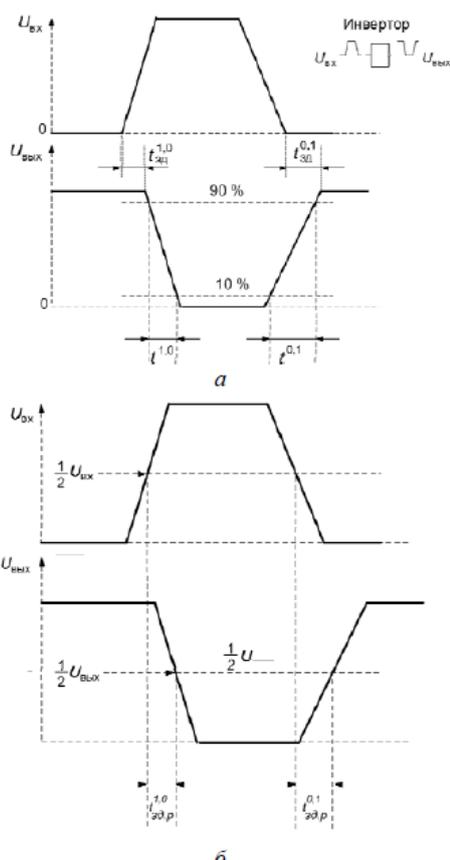
7. Средняя потребляемая мощность от источника питания  $P_{\text{пот.ср}}$ .

$$P_{\text{пот.ср}} = \frac{P^0_{\text{пот.}} + P^1_{\text{пот.}}}{2}$$

Типовые параметры цифровых логических ИС

Техно- логия	Серия	Логические элементы			Триггеры
		Время задержки распространения, нс	Мощность потребления, мВт	Нагрузка $R_{\text{ц}}$ , кОм	Частота переключения, (среднее значение), МГц
ТТЛ	54/74 (К155)	≈20	10	0,4	≈10
ТТЛШ	54S/74S (530, 531)	5÷7	20	0,28	≈50
	54LS/74LS (533, 555)	10	2	2	≈25
	54AS/74AS (1530)	1,5	20÷40	-	≈160
	54ALS/74ALS (1533)	4	1,2	2	≈70
	74F (1531)	3	4	0,28	≈125

Динамические параметры характеризуют работу ИМС в момент переключения из нуля в единицу или из единицы в ноль.



$t^{01}$  – длительность положительного перепада (переднего фронта);  $t^{10}$  – длительность отрицательного перепада (заднего фронта);  $t_{зд}^{01}$  – время задержки распространения при выключении логического элемента.  $t_{зд}^{10}$  – время задержки распространения при включении логического элемента;

$$t_{зд.р.ср} = \frac{t_{зд.р.}^{01} + t_{зд.р.}^{10}}{2}$$

$t_{зд.р.ср}$  – *среднее время задержки распространения сигнала* определяет длительность выполнения логических операций и служит **усредненным параметром быстродействия**

**Задание:** Сравнить ЦИМС по быстродействию, помехоустойчивости, нагрузочной способности, потребляемой мощности, проанализировать работу электрических принципиальных схем

#### 4. Порядок выполнения

4.1. Получите вариант задания у преподавателя

4.2. Составьте таблицу справочных данных для заданных ИМС

Таблица 1

Параметры	Тип, логика, серия		
	ТТЛШ	ЭСЛ	КМОП
	ИМС:	ИМС:	ИМС:
Рпотр, мВт			
U вых. по 1,			

В			
U вых. по 0, В			
I вх. по 0, мА			
I вх. по 1, мА			
U пит., В			
К раз.			
T ср. зад. р., нс			
U пом. ст., В			
F max, МГц			

4.3. Начертите УГО логических элементов и таблицы истинности заданных микросхем.

4.4. Начертите электрические схемы БЛЭ заданных ИМС.

4.5. Опишите принцип работы электрических схем БЛЭ заданных ИМС в виде таблицы состояний элементов схем.

4.6. Начертите корпуса микросхем и расшифруйте условные обозначения (маркировку) этих корпусов.

4.7. Сделайте выводы по быстродействию, помехоустойчивости, нагрузочной способности, потребляемой мощности изученных микросхем и ответьте на контрольные вопросы.

5. Указания к оформлению отчета

Отчет должен содержать:

тему и цель работы, задание и порядок выполнения.

таблицу справочных данных для заданных ИМС и выводы по быстродействию, помехоустойчивости, нагрузочной способности, потребляемой мощности.

УГО логических элементов и таблицы истинности заданных микросхем.

электрические схемы БЛЭ заданных ИМС и таблицы, отражающие принцип их работы.

корпуса микросхем с расшифровкой их маркировки.

## 6. Контрольные вопросы

1 Какие функции реализуют логические элементы И, ИЛИ, ИЛИ-НЕ, исключающее ИЛИ, И-НЕ. Напишите уравнения этих функций.

2. Напишите таблицу истинности для трехходового элемента И, ИЛИ, ИЛИ-НЕ, И-НЕ
3. С помощью какого параметра определяется быстродействие ИМС?
4. Сделайте расшифровку обозначениям ИМС заданного варианта.

### **7. Список литературы**

1. Основы электроники: учебник для СПО/ О.В., Миловзоров, И.Г.Панков, — 6-е изд.перераб. и доп. —Москва: Издательство Юрайт, 2021
2. Электронная техника: учебник / М.В. Гальперин. — 2-е изд., испр. и доп. — М. : ИД «ФОРУМ» : ИНФРА-М, 2023г. электронный портал znanium.
3. Берикашвили В.Ш. Черепанов А.К. «Электронная техника.– М.: «Академия», 2018г.
4. Колпакова Т.И., Калиенко И.В., Методическое пособие «Логические элементы», РКРИПТ, 2017.

## Приложения

### Приложение А

ЛОГИЧЕСКИЙ ЭЛЕМЕНТ	ОБОЗНАЧЕНИЕ
И	ЛИ
ИЛИ	ЛЛ
И-НЕ	ЛА
ИЛИ-НЕ	ЛЕ
ИЛИ-НЕ, ИЛИ	ЛП
НЕ	ЛН
И-ИЛИ-НЕ	ЛР
ИЛИ-НЕ/ИЛИ	ЛМ
И-ИЛИ	ЛС
И-ИЛИ-НЕ/И-ИЛИ	ЛК
РАСШИРИТЕЛИ	ЛД

## Приложение Б

### Сравнительные характеристики серий логических микросхем

Сравнительные характеристики основных параметров ТТЛ и КМОП серий представлены в табл. Б1 и на рис. Б1.

Высокое быстродействие в сочетании с низкой потребляемой мощностью, большой нагрузочной способностью, широкий набор логических и интерфейсных микросхем серии КР1533 позволяет создавать вычислительные устройства цифровой автоматики с качественно новыми

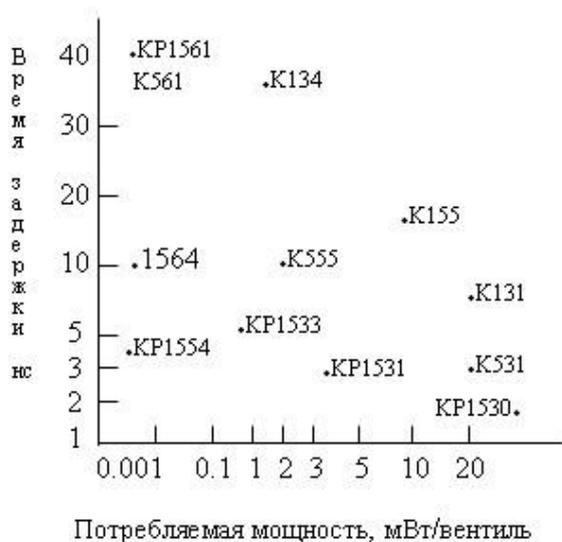


Рис. Б1

характеристиками и высокими технико-экономическими показателями.

Для наглядности типовые значения времени задержки распространения и потребляемой мощности в пересчете на логический вентиль для различных отечественных серий микросхем приведены ниже (на графике приведены значения статической потребляемой мощности с учетом ее слабой зависимости от частоты для

данных серий, за исключением КМОП серии КР1554).

Существенной особенностью серии КР1533 является наличие интерфейсных и буферных микросхем, обладающих повышенной нагрузочной способностью по выходу в состоянии высокого и низкого уровня ( $I_{OH} = 15 \text{ mA}$ ,  $I_{OL} = 24 \text{ mA}$ ) и меньшей в сравнении с серией КР1531 мощностью потребления при практически сравнимом быстродействии. Микросхемы серии КР1533, имеющие функциональные аналоги в других сериях, совпадают с ними в части разводки выводов в корпусе. Это позволяет проводить полную замену микросхем серий К555, К533, К155, КР1531 и добиваться уменьшения размеров блоков питания, уменьшения рассеиваемой мощности и повышения надежности.

Микросхемы серии КР1554, обладая всеми преимуществами КМОП микросхем, превзошли новейшие серии ТТЛ ИС по быстродействию и нагрузочной способности по выходу, что позволит разработчикам аппаратуры существенно улучшить технические и технико-экономические характеристики разрабатываемых изделий.

Сравнительные характеристики серий логических микросхем

Таблица Б1

Наименование параметра	Обознач.	К555	КР153 3	КР156 1	1564	КР 1554	Ед.изм
				К561			
Технология		ТТЛШ	ТТЛШ	КМОП	КМОП	КМОП	
Диапазон напряжения питания	U <sub>cc</sub>	5+5%	5+10%	3,0...1 5	2,0...6, 0	2,0...6, 0	В
Входное напряжение	U <sup>1</sup> <sub>вх.</sub>	2,0	2,0	3,15	3,15	3,15	В
	U <sup>0</sup> <sub>вх.</sub>	0,8	0,8	0,9	0,9	1,35	
Выходное напряжение	U <sup>1</sup> <sub>вых.</sub>	2,7	2,7	U <sub>cc</sub> -0.1	U <sub>cc</sub> -0.1	U <sub>cc</sub> -0.1	В
	I <sup>0</sup> <sub>вых.</sub>	0,5	0,5	0,1	0,1	0,1	
Входной ток	I <sup>1</sup> <sub>вх.</sub>	20	20	+0,3	+0,1	+0,1	мкА
	I <sup>0</sup> <sub>вх.</sub>	-400	-200	-0,3	-0,1	-0,1	
Выходной ток	I <sup>1</sup> <sub>вых.</sub>	-0,4	-0,4	-0,44	-4,0	-24	мкА
	I <sup>0</sup> <sub>вых.</sub>	8,0	8,0	0,44	4,0	24	
Запас помехоустойчивости	U <sub>пом</sub> ст	0,7	0,7	1,25	1,25	1,25	В
Мощность потр. на вентиль (статическая)	P <sub>потр.</sub>	2,0	1,2	0,0025	0,0025	0,0025	мВт
Частота переключения	F <sub>max</sub>	33,0	50,0	8,0	50,0	150,0	МГц
Время задержки распространения	Т.з.р.	10,0	5,0	40,0	10,0	4,0	нс
				160,0			

**Примечания:**

- 1 Статические параметры представлены для диапазона температур.
- 2 Значения параметров I<sub>в</sub>, P<sub>в</sub>, Э, F<sub>МАКС</sub>, рассчитаны по типовым значениям.
- 3 Значения параметров t<sub>р</sub> представлены для К555 при U<sub>cc</sub>=5.0 В, C<sub>L</sub>=15нФ, для КР1533, 1564, КР1554 - при U<sub>cc</sub>= 5.0 В ± 10%, C<sub>L</sub>=50нФ, типовое значение - при 25°С, максимальное значение для КР1533 - в диапазоне от -10 до +70°С, максимальное значение для КР1554 - в диапазоне от -45 до +85°С.

Таблица Б.2

## Основные параметры серий ИМС

Технологии	Серии ИМС	Рпотр, мВт	Тср. з. р., нс	F max, МГц	U пит., В	U лог1, В	U лог0, В	I вх.1, mA	I вх. 0, mA	К раз.	U пом. ст., В
ТТЛШ	K555	2	10	15	5±0.25	2,0	0,5	-2,4	0,4	30	0,5
	533	2	9,5	45		2,4	0,4	-0,1	0,02	20,40	0,8
	530	19	5	125		2,0	0,8	-0,6	0,02	30	0,8
	KP1531	4	3	130		8,2	0,3	-0,001	0,001	15	0,1
	1530	19	2,5	50		7,2	2,9			15	
	KP1533	1	4	70...10		3,15	0,9			15	0,8
	1531	4	2	130	2,4	0,4			5...10	1,25	
КМОП	K176	0,03	200	2	9(9...12)	3,15	0,9			15	0,8
	564	0,06	150	4	9(3...15)	-0,7...-	-1,6...-			15	1,25
	1564	0,0024	10	50	5(2..6)	1,0	1,9	-0,1	0,1	30-50	1,25
	1554	0,04	4	150	5(3,3...5.5)					30-50	1,25
	561	0,0024	50	5	5(3...15)	4,8	0,9			15	0,8
ЭСЛ	K500	25	2	200	-5,2±0.25	-0,98	-1,63	0,0005	0,265	15	0,1
	K1500	40	0.75	200	-4,5±0.25	-1,03	-1,61	0,0005	0,35		0,1
	K100	25	2	200	-5,2±0.25	-0,98	-1,63				0,1
	193	100		200...2000	+5,2±0.25						

Корпуса серий ТТЛШ  
 К555, КМ555, КР1531: 201.14-1; 201.14-3;  
 201.401-8  
 1533, КР1533: 201.14-1

Корпуса серий КМОП  
 К561, 564: 201.14-1; 201.14-6  
 КР1561: 201.14-3; 201.14-8  
 КР1554: 201.14-1

Корпуса серий ЭСЛ: 402.16-6; 402.16-32  
 буква М: 238.16-2; 201.16-2  
 буква Т: 210.16-8

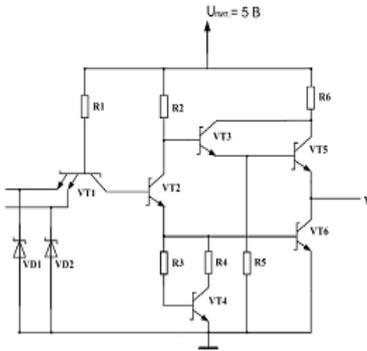
| KP1564: 201.14-2

|

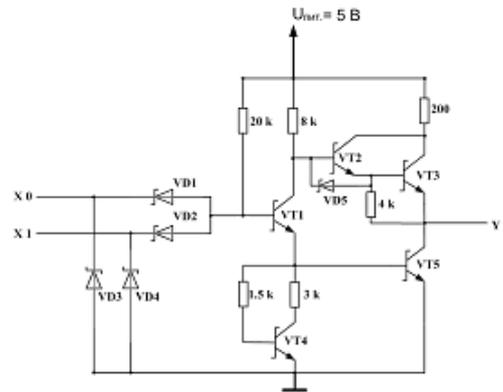
Приложение В

Электрические схемы базовых логических элементов

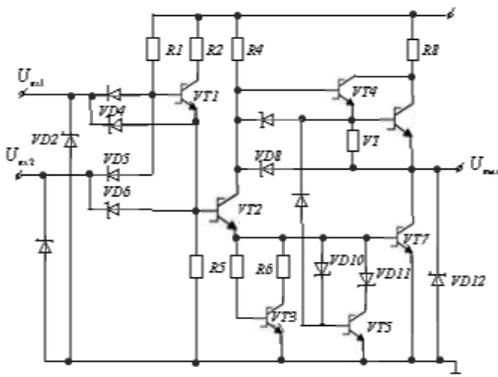
Электрическая схема элемента И-НЕ ТТЛШ (ИС 530, 531, КМ530, КМ531, КР531)



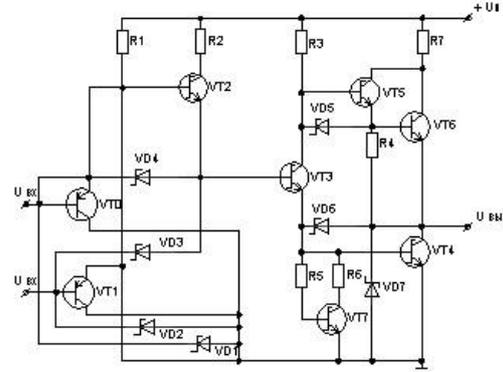
Электрическая схема элемента И-НЕ ТТЛШ (ИС 555, 533, К533, КМ555, К555)



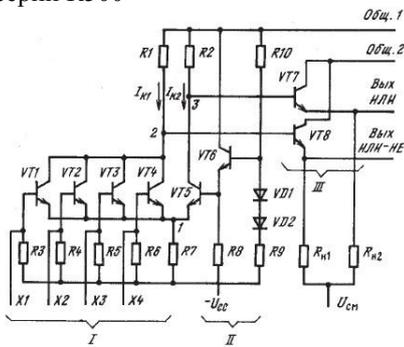
Электрическая схема базового элемента серии КР1531



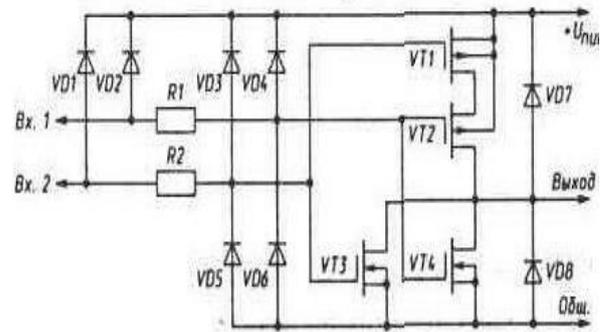
Электрическая схема базового элемента серии КР1533



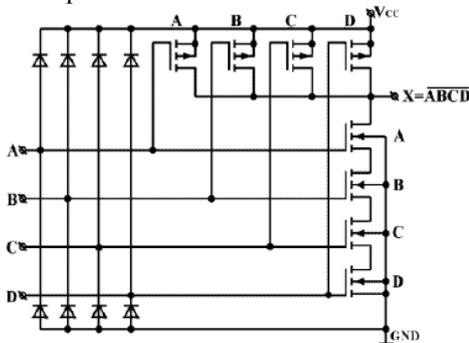
Электрическая схема базового элемента ИЛИ/ИЛИ-НЕ серии К500



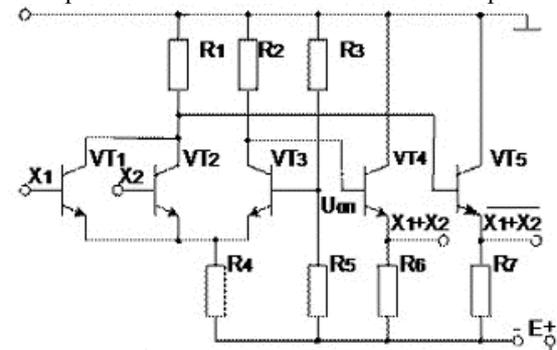
Электрическая схема базового элемента ИЛИ-НЕ К564



Электрическая схема базового элемента И-НЕ серии 1561



Электрическая схема базового элемента серии К1500



Приложение Г

Таблица Г

1-ИМС серий ТТЛШ К555, КМ555, 1531, КР1531, 1533, КР1533

Функциональное назначение	Условное обозначение	Номер рисунка
Два ЛЭ И-НЕ	ЛА1	Г1
Четыре ЛЭ 2И-НЕ	ЛА3	Г2
Три ЛЭ 3И-НЕ	ЛА4	Г3
Два ЛЭ 4И-НЕ с ОК	ЛА7	Г4
Четыре ЛЭ 2И-НЕ с ОК	ЛА8	Г5
Четыре ЛЭ 2И-НЕ с ОК	ЛА9	Г6
Три ЛЭ 3И-НЕ с ОК	ЛАЮ	Г7
Четыре буферных элемента 2И-НЕ с ОК	ЛА13	Г8
Два ЛЭ 4И-НЕ(Z)	ЛА 17	Г9
Четыре ЛЭ 2ИЛИ-НЕ	ЛЕ1	ГШ
Четыре ЛЭ 2И	ЛИ1	Г11
Три ЛЭ 3И	ЛИЗ	Г12
Четыре ЛЭ 2ИЛИ	ЛЛ1	Г13

Условные графические обозначения серий ТТЛШ

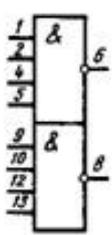


Рис. Г1

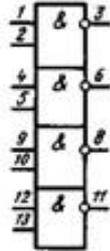


Рис. Г2

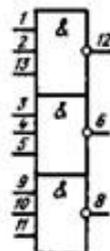


Рис. Г3

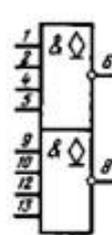


Рис. Г4

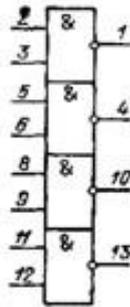


Рис. Г5

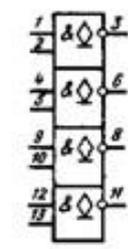


Рис. Г6

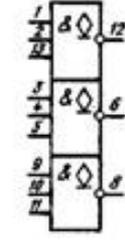


Рис. Г7

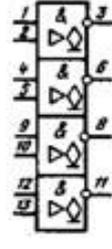


Рис. Г8

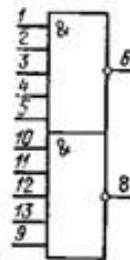


Рис. Г9

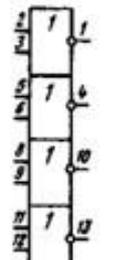


Рис. Г10



Рис. Г11

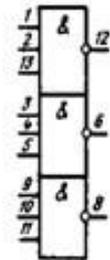


Рис. Г12

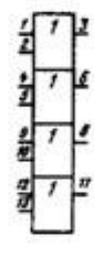


Рис. Г13



Приложение Д

Таблица Д1

ИМС серий ЭСЛ 100, К100, 500, К500, 1500, К1500

Функциональное назначение	Условное обозначение	Номер рисунка
Два ЛЭ2ИЛИ-НЕ и ЛЭ4ИЛИ-НЕ	ЛЕ106	Д1
Три ЛЭ ИЛИ-НЕ с мощным выходом	ЛЕ123	Д2
Три ЛЭ ИЛИ-НЕ с мощным выходом (быстродействующие магистральные усилители)	ЛЕ211	Д3
Два ЛЭ 2-3 ИЛИ-2И/ЗИЛИ-2И-НЕ	ЛК117	Д4
ЛЭ 3-3-3-4 ИЛИ-4И-НЕ/ 3-3-3-3 ИЛИ-4И	ЛК121	Д6
Два ЛЭ ИЛИ с мощным выходом	ЛЛ210	Д6
Четыре ЛЭ 2ИЛИ-НЕ/2ИЛИ	ЛМ101	Д7
Четыре ЛЭ 2ИЛИ-НЕ и ЛЭ 2ИЛИ-НЕ/2ИЛИ	ЛМ102	Д8
Два ЛЭ 2ИЛИ-НЕ/2ИЛИ и ЛЭ 3ИЛИ-НЕ/3ИЛИ	ЛМ105	Д9
Два ЛЭ 5ИЛИ-НЕ/5ИЛИ и 4ИЛИ-НЕ/4ИЛИ	ЛМ109	Д10
Три ЛЭ ИСКЛЮЧАЮЩЕЕ ИЛИ-НЕ/ИЛИ	ЛП107	Д11
Четыре приемника с линии	ЛП115	Д12
Три приемника с линии	ЛПИ 16	Д13
Три приемника с линии	ЛП216	Д14
Два ЛЭ3-ЗИЛИ-2И	ЛС118	Д15
ЛЭ3-3-3-4ИЛИ-4И	ЛС119	Д16

Условные графические изображения логических элементов ЭСЛ

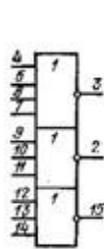


Рис. Д1



Рис. Д2

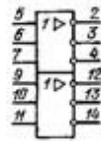


Рис. Д3

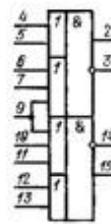


Рис. Д4

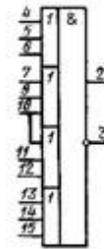


Рис. Д5



Рис. Д6

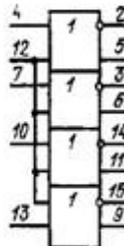


Рис. Д7

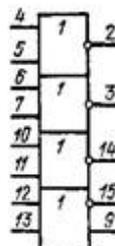


Рис. Д8



Рис. Д9

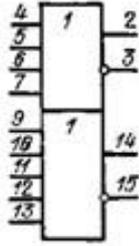


Рис. Д10

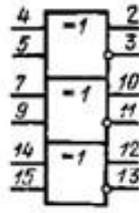


Рис. Д11

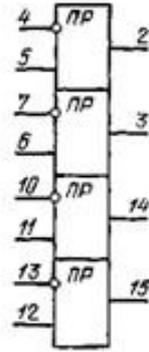


Рис. Д12

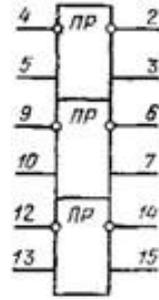


Рис. Д13

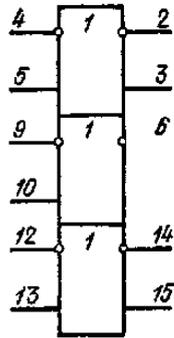


Рис. Д14

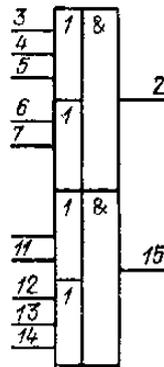


Рис. Д15

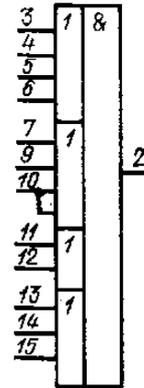


Рис. Д16

Приложение Е

Таблица Е.1

ИМС серий КМОП 561, К561, 564, К564, К1554, КР1554, К1561, 1564

Функциональное назначение	Условное обозначение	Номер рисунка
Два ЛЭ 4И-НЕ	ЛА1	Е1
Четыре ЛЭ 2И-НЕ	ЛА3	Е2
Три ЛЭ 3И-НЕ	ЛА4	Е3
Четыре ЛЭ 2И-НЕ	ЛА7	Е4
Два ЛЭ 4И-НЕ	ЛА8	Е5
Три ЛЭ 3И-НЕ	ЛА9	Е6
Четыре ЛЭ 2ИЛИ-НЕ	ЛЕ1	Е7
Четыре ЛЭ 2ИЛИ-НЕ	ЛЕ5	Е8
Два ЛЭ 4 ИЛИ-НЕ	ЛЕ6	Е9
Три ЛЭ 3 ИЛИ-НЕ	ЛЕ10	Е10
Четыре ЛЭ 2И	ЛИ1	Е11
Три ЛЭ 3И	ЛИЗ	Е12
Шесть ЛЭ НЕ	ЛН2	Е13
Четыре ЛЭ 2 ИЛИ	ЛЛ1	Е14
Четыре ЛЭ ИСКЛЮЧАЮЩЕЕ ИЛИ	ЛП2	Е15
Два ЛЭ 3ИЛИ-НЕ и ЛЭ НЕ	ЛП4	Е16
Три ЛЭ 3И-ИЛИ	ЛС1	Е17

Условные графические изображения логических элементов КМОП



Рис. Е1



Рис. Е2



Рис. Е3



Рис. Е4

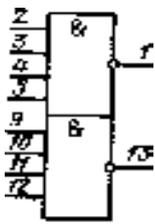


Рис. E5

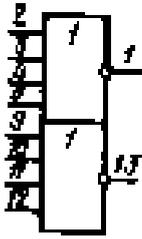


Рис. E6

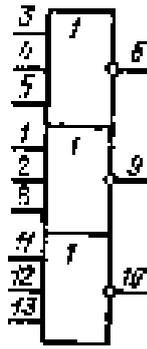


Рис. E7

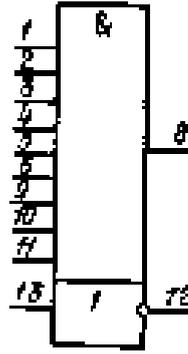


Рис. E8

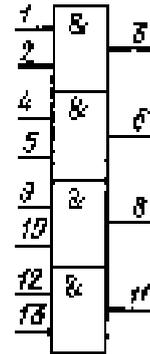


Рис. E9

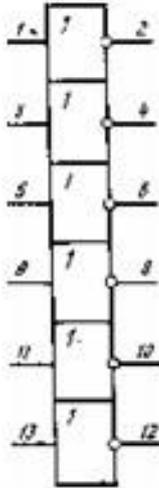


Рис. E10

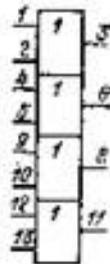


Рис. E11

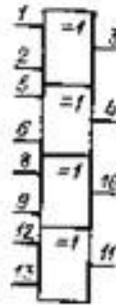


Рис.

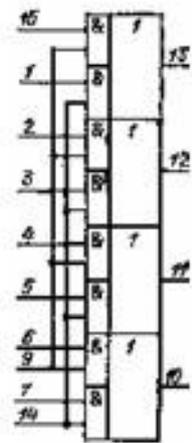


Рис. E13

Рис. E14

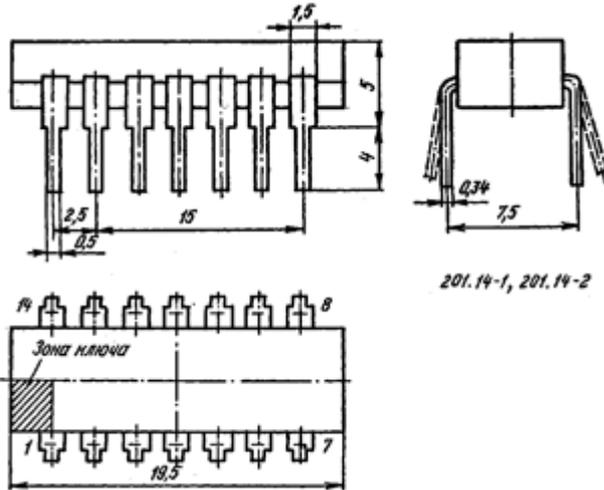
Рис. E15

Рис. E16

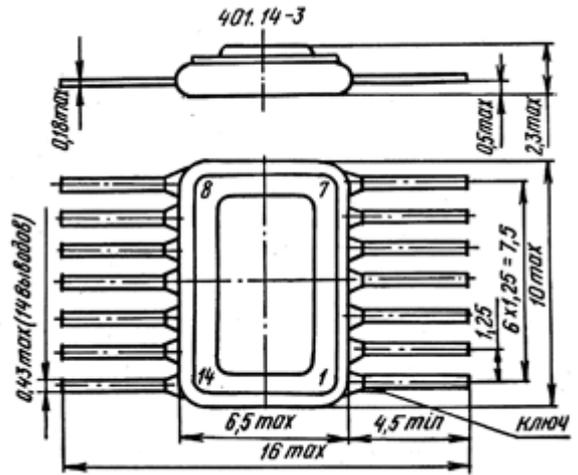
Рис. E17

Приложение Ж  
КОРПУСА ИМС

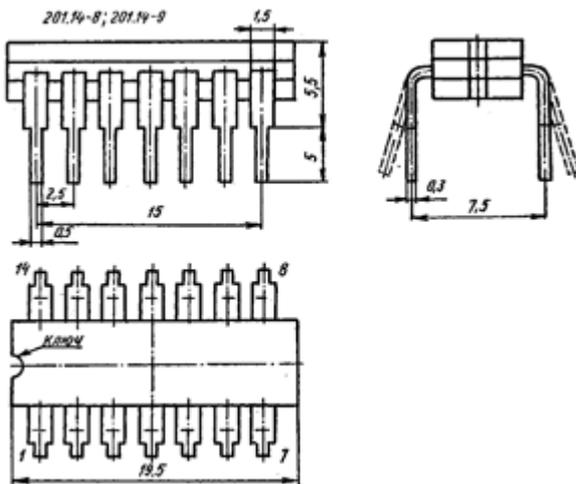
201.14-1



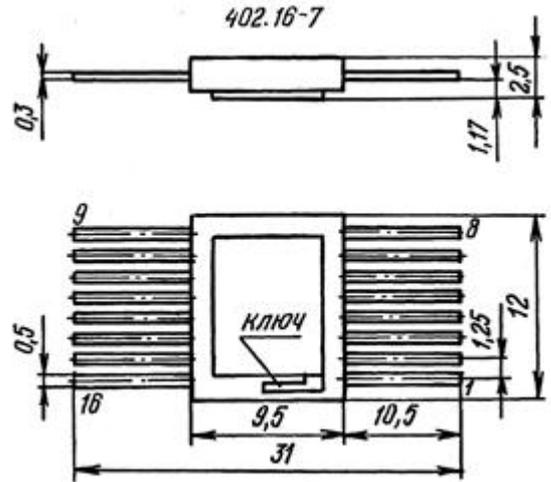
201.14-3



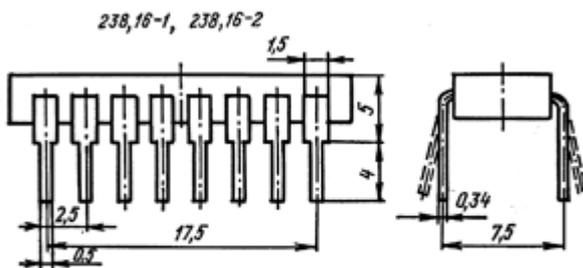
201.14-8, 201.14-9



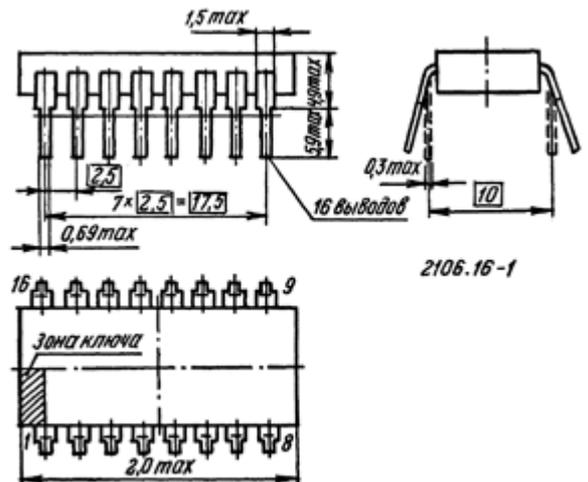
402.16-6, 402.16-7



238.16-1, 238.16-2

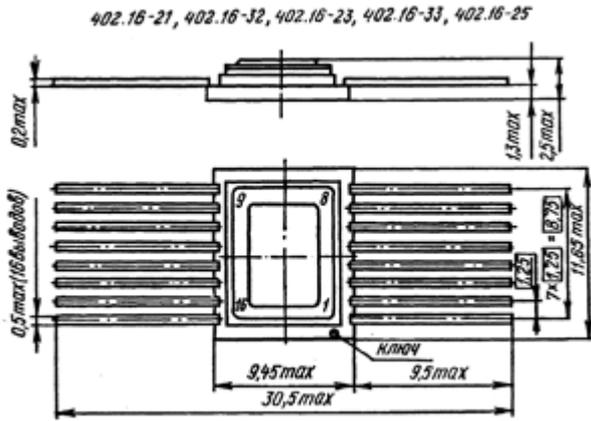


210.16-8

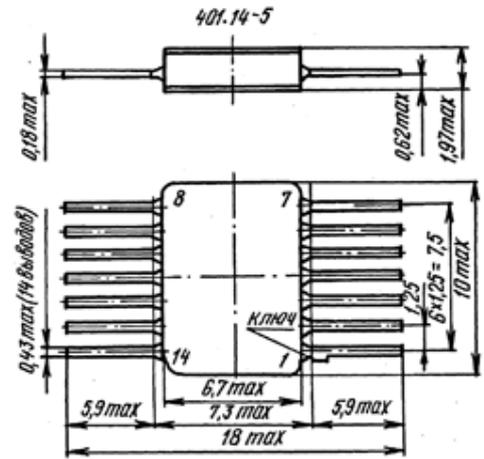


КОРПУСА ИМС

402.16-21, 402.16-23, 402.16-32, 402.16-33, 402.16-25

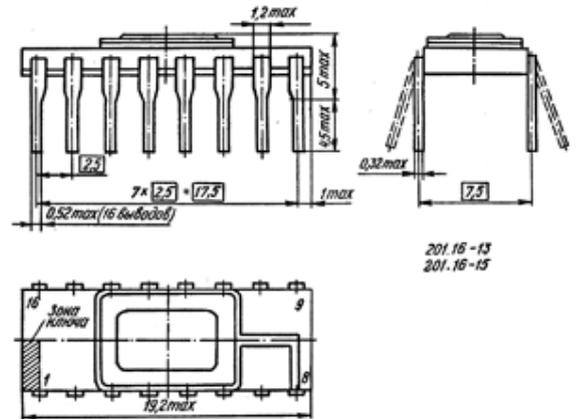
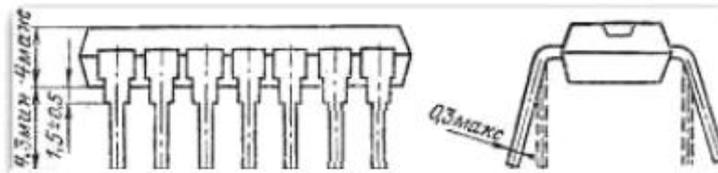


401.14-5



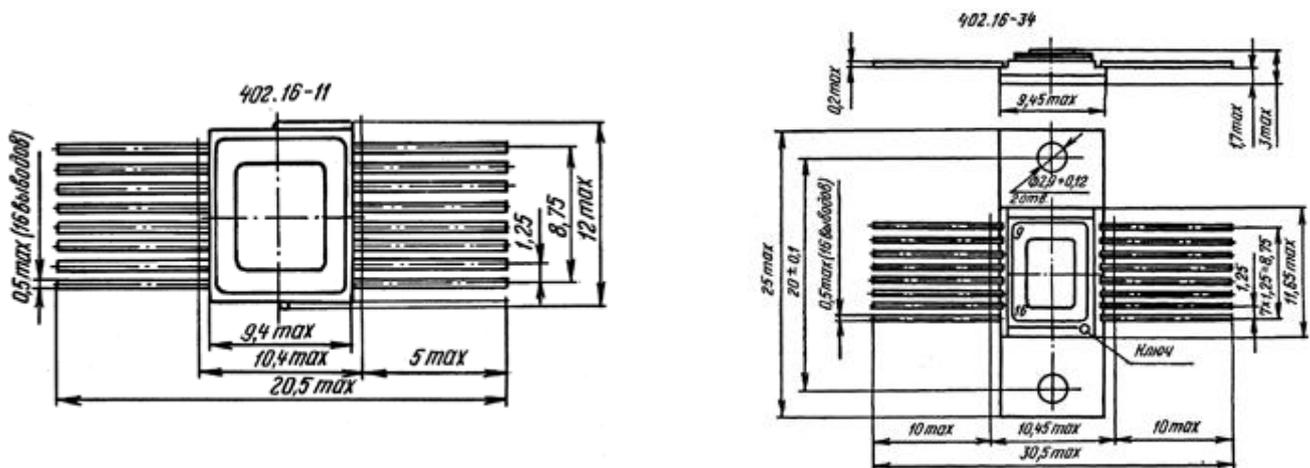
201.14-6

201.16-13, 201.16-15



402.16-11

402.16-34



## ПРАКТИЧЕСКАЯ РАБОТА №5

### ПОСТРОЕНИЕ ЛОГИЧЕСКИХ СХЕМ В ОСНОВНОМ И УНИВЕРСАЛЬНЫХ БАЗИСАХ

**1. Цель работы:** Получить практическое представление о способах представления, преобразования логических функций и их реализации.

**2. Время выполнения работы-4 час.**

**3. Краткие теоретические сведения**

Схемы, реализующие логические функции, называются логическими элементами (ЛЭ). Основные логические элементы имеют, как правило, один выход ( $Y$ ) и несколько входов, число которых равно числу аргументов ( $X_1; X_2; X_3 \dots X_N$ ). На электрических схемах логические элементы обозначаются в виде прямоугольников с выводами для входных (слева) и выходных (справа) переменных. Внутри прямоугольника изображается символ, указывающий функциональное назначение элемента. Работу логических элементов описывают с помощью **таблиц истинности**. В таблице истинности указываются значения логической функции при всех возможных сочетаниях значений двоичных переменных. Количество таких сочетаний  $N = 2^n$ , где  $n$  – количество двоичных переменных, от которых зависит значение логической функции. А сами эти комбинации представляют собой двоичные коды номеров строк, начиная с нуля. Например, если логическая функция  $y = f(x_1, x_2, x_3)$  зависит от трёх логических переменных ( $n = 3$ ) –  $x_1, x_2, x_3$  – то количество различных сочетаний значений  $x_1, x_2, x_3$  (строк таблицы истинности)  $N = 2^3 = 8$ .

С помощью схем ЛЭ можно реализовать любую логическую функцию.

Задачи, решаемые при разработке цифровых логических устройств, можно разделить на две категории:

1. Синтеза.
2. Анализа.

Синтез - это процесс построения схемы цифрового устройства по заданию.

Анализ - процесс обратный синтезу.

Таким образом, логическую функцию можно представить

- в аналитической форме (в виде логических выражений)

- в табличной форме (в виде таблиц истинности)

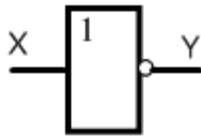
- в виде электрической функциональной схемы на логических элементах

Основными логическими функциями являются:

**Логическое отрицание (или инверсия).** Записывается эта функция так:

$$Y = \bar{X}$$

Данная функция реализуется логическим элементом, который называется инвертором или же элементом **НЕ**



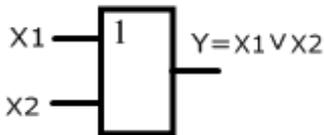
x	y
0	1
1	0

**Логическое сложение или дизъюнкция.** Записывается:

$$Y = X1 \vee X2$$

или  $Y = X1 + X2$

Реализуется логическим элементом, который называется элементом **ИЛИ**



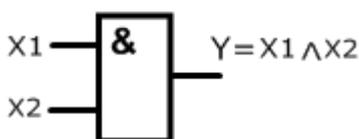
x1	x2	y
0	0	0
0	1	1
1	0	1
1	1	1

**Логическое умножение или конъюнкция.** Записывается:

$$Y = X1 \wedge X2$$

или  $Y = X1 \cdot X2$

Реализуется логическим элементом, который называется элементом **И**



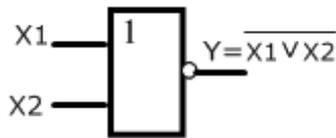
x1	x2	y
0	0	0
0	1	0
1	0	0
1	1	1

**Отрицание логического сложения.** Записывается:

$$Y = \overline{X1 \vee X2}$$

$$\text{или } Y = \overline{X1 + X2}$$

Реализуется логическим элементом, который называется элементом **ИЛИ-НЕ** или элементом Пирса



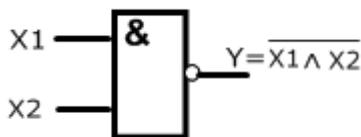
x1	x2	y
0	0	1
0	1	0
1	0	0
1	1	0

**Отрицание логического умножения.** Записывается:

$$y = \overline{X1 \wedge X2}$$

$$\text{или } Y = \overline{X1 \cdot X2}$$

Реализуется логическим элементом, который называется элементом **И-НЕ** или элементом Шеффера



x1	x2	y
0	0	1
0	1	1
1	0	1
1	1	0

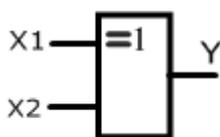
**Функция неравнозначности (сложение по модулю 2).**

Записывается:

$$Y = X1 \wedge \overline{X2} \vee \overline{X1} \wedge X2$$

$$\text{или } Y = X1 \cdot \overline{X2} + \overline{X1} \cdot X2$$

Реализуется логическим элементом, который называется элементом **исключающее ИЛИ**



x1	x2	y
0	0	0
0	1	1
1	0	1
1	1	0

При выполнении практической работы будем использовать следующие тождества:

- 1)  $\overline{1} = 0$ ;      2)  $\overline{0} = 1$ ;    3)  $\overline{\overline{x}} = x$ ;  
 4)  $x \cdot x = x$ ;                      5)  $x + x = x$ ;  
 6)  $x \cdot 1 = x$ ;                      7)  $x + 1 = 1$ ;  
 8)  $x \cdot 0 = 0$ ;                      9)  $x + 0 = x$ ;  
 10)  $\overline{x} + x = 1$                       11)  $\overline{x} \cdot x = 0$

### Порядок выполнения логических операций в логических выражениях

1. Операция инверсии и действия в скобках
2. Логическое умножение
3. Логическое сложение

Например:

Если  $a = 1$ ,  $b = 0$ , а функция имеет вид  $y = a \cdot b + (a + b)$ , то значение функции определим как  $y = 1 \cdot 0 + (1 + 0) = 0 + 1 = 1$ .

Если знак инверсии стоит над какой то частью логического выражения, то это равнозначно тому, что эта часть заключена в скобки

### Основные законы алгебры логики

п/п	Закон	Логическое сложение	Логическое умножение
1	Переместительный	$x_1 \vee x_2 = x_2 \vee x_1$	$x_1 x_2 = x_2 x_1$
2	Сочетательный	$(x_1 \vee x_2) \vee x_3 = x_1 \vee (x_2 \vee x_3)$	$(x_1 x_2) x_3 = x_1 (x_2 x_3)$
3	Распределительный	$(x_1 \vee x_2) x_3 = x_1 x_3 \vee x_2 x_3$	$x_1 x_2 \vee x_3 = (x_1 \vee x_3)(x_2 \vee x_3)$
4	Инверсии	$\overline{x_1 \vee x_2} = \overline{x_1} \overline{x_2}$	$\overline{x_1 x_2} = \overline{x_1} \vee \overline{x_2}$

При выполнении практических работ мы будем неоднократно использовать закон (тождества) де Моргана:

1) инверсия конъюнкции входных переменных есть дизъюнкция инверсий входных переменных

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

2) инверсия дизъюнкции входных переменных есть конъюнкция инверсий входных переменных.

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

$$\overline{a \cdot \overline{b} \cdot c} = \overline{a} + b + \overline{c}$$

$$\overline{\overline{a} + \overline{b} + c} = a \cdot b \cdot \overline{c}$$

**Понятие о дизъюнктивной нормальной форме (ДНФ) и конъюнктивной нормальной форме (КНФ).**

Дизъюнктивная нормальная форма (ДНФ) - это такая форма представления функции, при которой логические выражения функции строятся в виде дизъюнктивного ряда членов, каждый из которых является простой конъюнкцией аргументов. На простом языке, ДНФ - это форма записи функции в виде логической суммы слагаемых, каждое из которых является логическим произведением переменных. Т.е. ДНФ представляет собой дизъюнкцию элементарных конъюнкций. Если в каждом члене ДНФ представлены все аргументы или их инверсии, причем, каждая конъюнкция включает в себя каждую переменную только один раз в прямом или инверсном виде, раз такая форма записи называется совершенной (СДНФ).

Существует также конъюнктивная нормальная форма (КНФ) - форма представления функции в виде конъюнкции (логического умножения) ряда членов, каждый из которых является простой дизъюнкцией (логическим сложением) аргументов. Т.е. КНФ представляет собой конъюнкцию элементарных дизъюнкций. Если в каждом члене КНФ представлены все аргументы или их инверсии, причем, каждая дизъюнкция включает в себя каждую переменную только один раз в прямом или инверсном виде, то такая форма называется совершенной (СКНФ).

Для нахождения СДНФ и СКНФ любой БФ существуют следующие алгоритмы.

Пусть БФ трех переменных  $F$  задана таблицей истинности (таблица 7).

Таблица 7

Составим СДНФ для  $F$ :

- выделяем наборы переменных, на которых функция равна 1;
- записываем для этих наборов конъюнкции, при этом если переменная равна 1, то эта переменная записывается без отрицания, если же переменная равна 0, то такая переменная записывается с отрицанием;
- объединяем элементарные конъюнкции знаками дизъюнкций;
- полученное выражение будет являться совершенной ДНФ.

$x$	$y$	$z$	$F$
0	0	0	1
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

$$F_{\text{сднф}} = \bar{x} \cdot \bar{y} \cdot \bar{z} + \bar{x} \cdot y \cdot z + x \cdot \bar{y} \cdot z + x \cdot y \cdot \bar{z} + x \cdot y \cdot z$$

Алгоритм нахождения СКНФ:

- выделяем те наборы переменных, на которых функция равна 0;

- из этих наборов переменных составляем дизъюнкции, учитывая то, что если переменная равна 0, то она записывается без отрицания, а если 1 – с отрицанием;
- объединяем элементарные дизъюнкции знаками конъюнкций;
- полученное выражение является совершенной КНФ.

$$F_{\text{СКНФ}} = (x + y + \bar{z}) \& (\bar{x} + y + z) \& (x + \bar{y} + z)$$

### Понятие о базисе

Минимальный набор логических операций, с помощью которых можно реализовать любую, сколь угодно сложную функцию. Функционально полными являются 3 базиса:

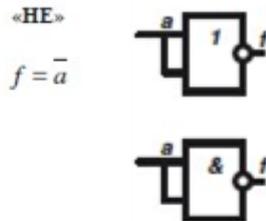
- 1) "И-ИЛИ-НЕ" (базис конъюнкции, дизъюнкции, инверсии)
- 2) "И-НЕ" (базис Шеффера)
- 3) "ИЛИ-НЕ" (базис Пирса).

Перевод в базис И-НЕ или ИЛИ-НЕ позволяет использовать одноименные микросхемы, а также уменьшить набор корпусов. Перевод в эти базисы осуществляется с использованием тождества де Моргана.

$$\overline{x1 + x2} = \bar{x1} \bar{x2}$$

$$\overline{x1 x2} = \bar{x1} + \bar{x2}$$

А реализация операции инверсии одной переменной выполняется следующим образом



**Задание ч.1:** Перейти от аналитической формы задания к схемной в основном логическом базисе И,ИЛИ,НЕ

#### **4. Порядок выполнения**

**4.1.** Построить схему заданной в таблице вариантов логической функции в логическом базисе И,ИЛИ,НЕ,

#### **5. Указания к выполнению**

5.1. При построении схемы вы можете основываться на примерах как с предварительным разбиением на элементарные логические функции, так и без предварительного разбиения

5.2. Построение схемы заданной функции в базисе И,ИЛИ,НЕ

#### **Пример 1:**

$$f = (a + b) \cdot \overline{c \cdot d}$$

разобьем эту функцию на ряд элементарных логических функций

$$f_1 = a + b$$

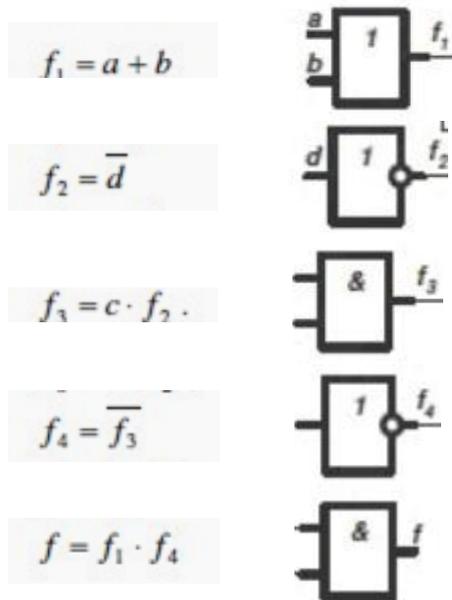
$$f_2 = \overline{d}$$

$$f_3 = c \cdot f_2$$

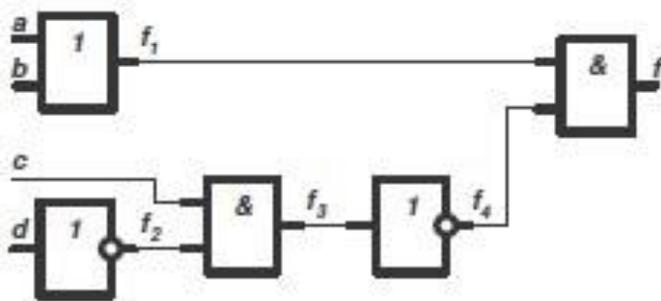
$$f_4 = \overline{f_3}$$

$$f = f_1 \cdot f_4$$

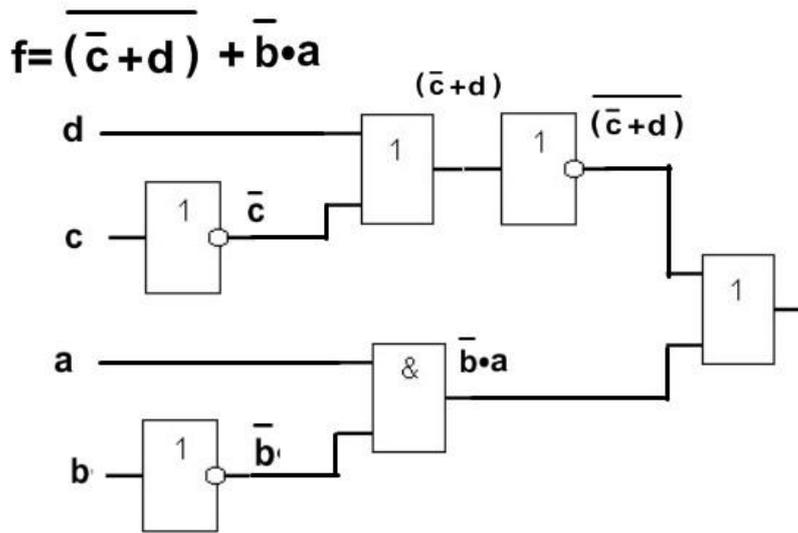
каждую элементарную логическую операцию реализует соответствующий логический элемент,



далее строим схему в соответствии с порядком выполнения логических операций



**Пример 2:** Построить схему заданной логической функции в базисе И,ИЛИ,НЕ без предварительного разбиения



### 6. Указания к выполнению отчета

Отчет должен содержать:

1. Тему и цель работы, задание и порядок выполнения,
2. Исходную функцию задания
3. Схему на логических элементах в базисе И, ИЛИ, НЕ, реализующую заданную логическую функцию.

**Задание ч.2:** Выполнить реализацию логических функций в заданных универсальных базисах

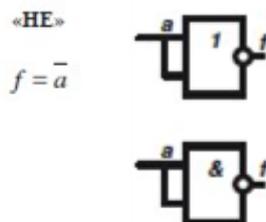
### 4. Порядок выполнения

**4.1.** Преобразуйте логическую функцию с использованием тождества де Моргана для перевода в базисы И-НЕ или ИЛИ-НЕ (в зависимости от варианта задания).

**4.2.** Постройте схему, используя ЛЭ И-НЕ или ИЛИ-НЕ (в зависимости от варианта задания).

### 5. Указания к выполнению

При построении схемы реализуйте операцию инверсии путем замены логического элемента НЕ на элементы И-НЕ или ИЛИ-НЕ с объединением входов.



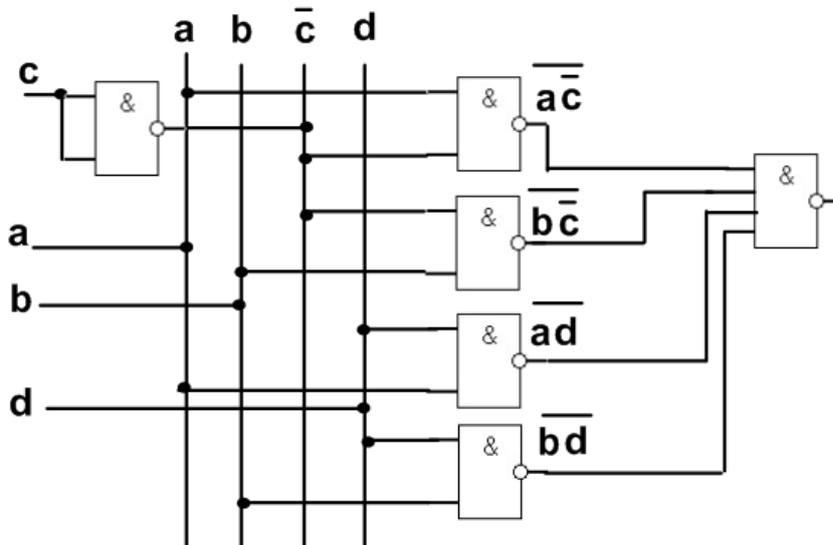
**Пример.** Построить схему заданной логической функции в базисе И-НЕ

$$f = a \cdot \bar{c} + b \cdot \bar{c} + a \cdot d + b \cdot d$$

выполним преобразования с применением тождества де Моргана,

$$f = \overline{a \cdot \bar{c} + b \cdot \bar{c} + a \cdot d + b \cdot d} = \overline{a \cdot \bar{c} \cdot b \cdot \bar{c} \cdot a \cdot d \cdot b \cdot d}$$

затем строим схему на логических элементах И-НЕ



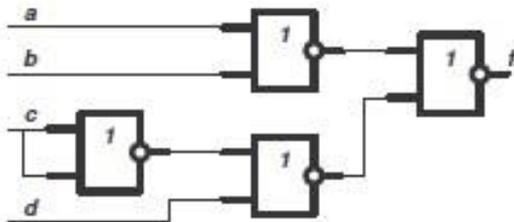
**Пример.** Построить схему заданной логической функции в базисе ИЛИ-НЕ

$$f = (a + b) \cdot (\bar{c} + d)$$

выполним преобразования с применением тождества де Моргана,

$$f = \overline{\overline{(a + b) \cdot (\bar{c} + d)}} = \overline{\overline{(a + b)} + \overline{(\bar{c} + d)}}$$

затем строим схему на логических элементах ИЛИ-НЕ



## 6. Указания к выполнению отчета

Отчет должен содержать:

1. Тему и цель работы, задание и порядок выполнения,
2. Преобразования логической функции с использованием тождества де Моргана
3. Схему в заданном базисе, реализующую заданную логическую функцию.

**Задание ч.3:** Выполнить реализацию логических функций в совершенной нормальной дизъюнктивной и конъюнктивной формах

## 4. Порядок выполнения

- 4.1. По заданной (см. табл.1) таблице истинности запишите СДНФ и СКНФ
- 4.2. Постройте полученные СДНФ и СКНФ в базисе И, ИЛИ, НЕ.
- 4.3. Ответьте на контрольные вопросы

## 5. Указания к выполнению

### 5.1. Запись функции в СДНФ

Правило перехода от табличной формы задания функции к СДНФ

- выбрать те наборы аргументов (строки таблицы истинности), на которых  $f(X_1, X_2, \dots, X_n)=1$ .
- если при этом  $X_i$  имеет значение '1', то остается без изменений, если '0', то с отрицанием.
- все конъюнктивные члены соединить знаком дизъюнкции.

**Пример:** Пусть логическая функция задана в виде таблицы

x3	x2	x1	f(x <sub>1</sub> , x <sub>2</sub> , x <sub>3</sub> )
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

Из таблицы истинности видно, что функция принимает значение логической единицы только на трех наборах переменных, т.е. на 1,2, и 5-м наборах (счет строки от нуля). Тогда аналитическое выражение функции будет выглядеть так

$$\text{СДНФ} = \bar{x}_3 \wedge x_2 \wedge \bar{x}_1 \vee \bar{x}_3 \wedge \bar{x}_2 \wedge x_1 \vee x_3 \wedge \bar{x}_2 \wedge \bar{x}_1 \vee x_3 \wedge x_2 \wedge x_1$$

### 5.2 Запись функции в СКНФ

Правило перехода от табличной формы задания функции к СКНФ или правило записи функции по нулям.

- выбрать те наборы аргументов (строки таблицы истинности), на которых  $f(X_1, X_2, \dots, X_n)=0$ .
- если при этом  $X_i$  имеет значение '0', то остается без изменений, если '1', то с отрицанием.
- все дизъюнктивные члены соединить знаком конъюнкции  $\wedge$ .

Например: Пусть логическая функция задана в виде таблицы

x <sub>1</sub>	x <sub>2</sub>	x <sub>3</sub>	f(x <sub>1</sub> , x <sub>2</sub> , x <sub>3</sub> )
0	0	0	0
0	0	1	1
0	1	0	0

0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

Из таблицы истинности видно, что функция принимает значение логического нуля в строках с номерами 0, 2, 5. Тогда аналитическое выражение функции будет выглядеть так

$$\text{СКНФ} = F(X_1, X_2, X_3) = (X_1 \vee X_2 \vee X_3) \wedge (X_1 \vee \bar{X}_2 \vee X_3) \wedge (\bar{X}_1 \vee X_2 \vee \bar{X}_3)$$

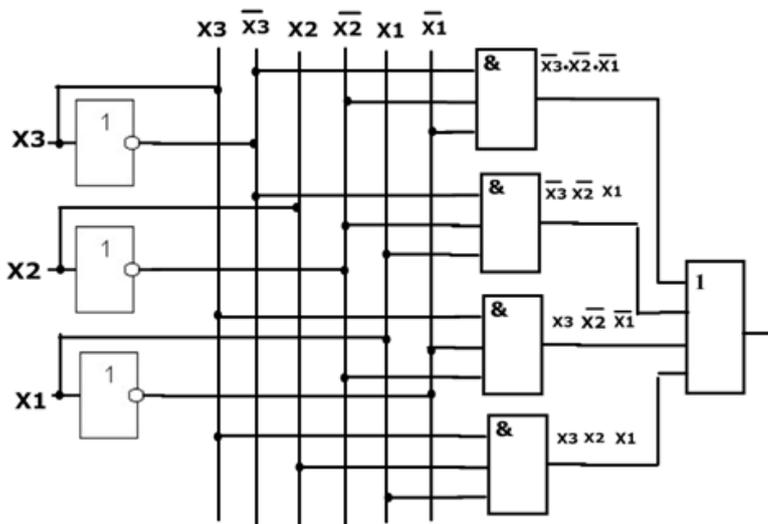
**5.3.** Построение схем производится аналогично первой части задания

Пример построения схем СДНФ и СКНФ функции

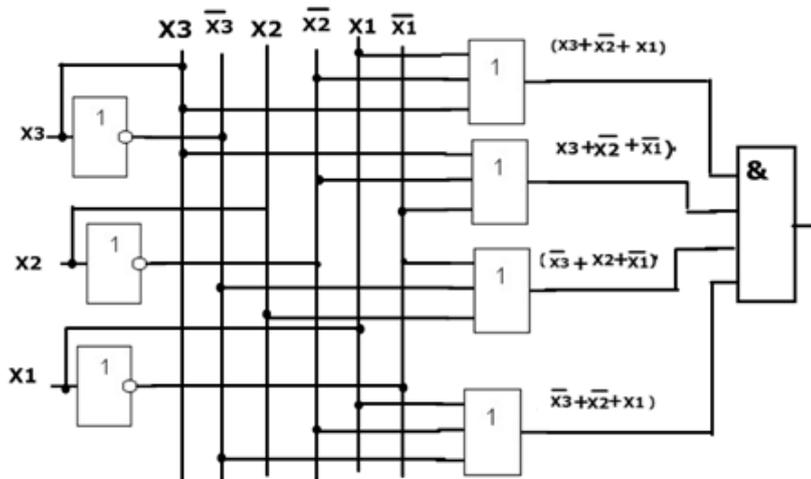
По заданной таблице истинности запишите СДНФ и СКНФ

X3	X2	X1	Y
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

$$Y_{\text{СДНФ}} = \bar{x}_3 \cdot \bar{x}_2 \cdot \bar{x}_1 + \bar{x}_3 \cdot \bar{x}_2 \cdot x_1 + x_3 \cdot \bar{x}_2 \cdot \bar{x}_1 + x_3 \cdot x_2 \cdot x_1$$



$$Y_{\text{СКНФ}} = (x_3 + \bar{x}_2 + x_1) \cdot (x_3 + \bar{x}_2 + \bar{x}_1) \cdot (\bar{x}_3 + x_2 + \bar{x}_1) \cdot (\bar{x}_3 + \bar{x}_2 + x_1)$$



## 6. Указания к выполнению отчета

Отчет должен содержать:

1. Тему и цель работы, задание и порядок выполнения,
2. Исходную таблицу истинности задания.
3. СДНФ и СКНФ, полученные по таблице.
4. Схемы СДНФ и СКНФ, на логических элементах

## 7. Контрольные вопросы

1. Что называется базисом логической функции? Какие логические базисы вам известны?
2. Сформулируйте оба тождества де Моргана.
3. Как получить СДНФ функции?
4. Как получить СКНФ функции?

## 8. Список литературы

1. Основы электроники: учебник для СПО/ О.В., Миловзоров, И.Г.Панков, — 6-е изд. перераб. и доп. — Москва: Издательство Юрайт, 2021
2. Вычислительная техника : учеб. пособие / Т.Л. Партыка, И.И. Попов. — 3-е изд., перераб. и доп. -М.: ФОРУМ:ИНФРА-М, 2023- электронный портал znanium.
3. Колпакова Т.И., Калиенко И.В., Методическое пособие «Логические элементы», РКРИПТ, 2017.
4. Ю.В. Новиков, Скоробогатов Основы микропроцессорной техники. – 3-е изд., испр. – М.: Интернет-Университет информационных технологий; БИНОМ. Лаборатория знаний, 2020г.

## СИНТЕЗ ЦИФРОВЫХ УЗЛОВ С ПРИМЕНЕНИЕМ МЕТОДА МИНИМИЗАЦИИ

**1. Цель работы** Приобретение практических навыков минимизации логических функций, заданных в виде таблицы истинности, методом карт Карно и реализации их на современной элементной базе

**2. Время выполнения работы-4час**

**3. Краткие теоретические сведения**

Минимизация логических функций необходима для упрощения сложных выражений этих самых функций. Минимизировать логические функции можно с помощью, так называемых карт Карно.

Карты Карно – это графическое представление таблиц истинности логической функций. Они были изобретены в 1952 Эдвардом В. Вейчем и усовершенствованы в 1953 Морисом Карно, физиком из «Bell Labs». Карта представляет собой прямоугольник (или квадрат) разбитый на клеточки. Количество клеточек (N) равно количеству возможных неповторяющихся комбинаций значений двоичных переменных, от которых зависит заданная функция, т.е.  $N = 2^n$ , где n – количество переменных. Если  $n = 4$ : (x1, x2, x3, x4), то  $N = 2^4 = 16$ , следовательно, карта Карно будет содержать 16 клеточек:


Карта размечается системой координат, соответствующих значениям **входных переменных** заданной в виде таблицы истинности функции. Каждой клетке карты будет соответствовать комбинация  $x_4x_3x_2x_1$ , получающаяся на пересечении строки  $x_4x_3$  и столбца  $x_2x_1$ .

Координаты строк и столбцов следуют не в естественном порядке возрастания двоичных кодов, а в порядке 00, 01, 11, 10. Это, так называемый, код Грея. Изменение порядка следования наборов сделано для того, чтобы соседние наборы (отличающиеся между собой лишь цифрой одного разряда) были соседними в геометрическом смысле. Процесс минимизации использует закон склеивания и заключается в формировании прямоугольных областей. Чем больше ячеек в прямоугольнике, тем меньше переменных содержится в соответствующем ему логическом выражении.

<b>x4x3</b>	⇨	⇩				
			<b>x2x1</b>			
			00	01	11	10
00	0000	0001	0011	0010		
01	0100	0101	0111	0110		
11	1100	1101	1111	1110		
10	1000	1001	1011	1010		

		<u>X1</u>	<u>X2</u>	
	0	1	0	1
X3	1	0	1	0
	0	1	0	0
X4	0	1	1	0

В каждой клетке карты Карно затем проставляется соответствующее значение функции на заданном наборе, т.е. ячейки таблицы, соответствующие наборам переменных, обращающих функцию в единицу, заполняются единицами, остальные – нулями. Далее выбирается наилучшее покрытие таблицы прямоугольными областями (наилучшим считается такое покрытие, которое образовано минимальным числом областей, а если таких вариантов несколько, то из них выбирается тот, который дает максимальную суммарную площадь областей). Области состояются либо только из нулей, либо только из единиц. Количество элементов в области может быть 1, 2, 4, 8, 16 и т.д., т.е.  $2^n$ . Один и тот же элемент может быть включен в разные области сколько угодно раз.

Используя этот метод, можно записать минимальную форму функции МДНФ и МКНФ. Если области состояются из единиц, то логическое выражение представляется в виде МДНФ, причем каждый член МДНФ представляет собой логическое выражение (произведение аргументов функции) для соответствующей области на карте. Если значение аргумента в области одинаково для всех ее элементов (в данном случае единиц), то этот аргумент заносится в выражение, причем если его значение равно нулю, то он записывается с инверсией.

Если области состояются из нулей, то логическое выражение представляется в виде МКНФ и если значение аргумента равно единице, то он записывается с инверсией.

**Задание:** По заданной таблице истинности составьте карту Карно, запишите минимальные формы ДНФ и КНФ и постройте схемы на логических элементах соответственно И-НЕ и ИЛИ-НЕ

#### 4. Порядок выполнения

**4.1.** Заполните исходную таблицу истинности, используя таблицу вариантов

$X_i$	Значения логических переменных
X4	0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1
X3	0 0 0 0 1 1 1 1 0 0 0 0 1 1 1 1
X2	0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1
X1	0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1
Вариант	Значения логической функции

4.2. Заполните карту Карно, перенося из таблицы истинности значения функции в клетки с координатами, соответствующей комбинации входных переменных.

**x2x1**

**x4x3**  $\Rightarrow$   $\Downarrow$

	00	01	11	10
00	0000	0001	0011	0010
01	0100	0101	0111	0110
11	1100	1101	1111	1110
10	1000	1001	1011	1010

4.3. Выделите контуры в карте Карно для записи минимальных формы ДНФ (МДНФ) и КНФ (МКНФ)

4.4. Запишите МДНФ и МКНФ

4.5. Постройте схемы, реализующую полученную функцию в базисах: МДНФ в базисе И-НЕ, МКНФ в базисе ИЛИ-НЕ

4.6. Подберите ЛЭ в интегральном исполнении для схем, используя справочную литературу.

4.7. Ответьте на контрольные вопросы

## 5. Указания к выполнению

5.1. При заполнении карты Карно следует помнить, что каждой клетке карты будет соответствовать комбинация  $x_4x_3x_2x_1$  входных переменных таблицы истинности, получающаяся на пересечении строки  $x_3x_4$  и столбца  $x_1x_2$ .

5.2. Объединение «1» (или «0») в группы производится по следующим правилам:

Группа должна быть прямоугольной (или квадратной) формы.

Группа может содержать только  $2^n$  клеточек, где  $n$  – целое число. Соответственно, в нашем примере группа может содержать 1, 2, 4, 8 или 16 клеточек.

При объединении клеточек в группы карту можно сворачивать как по горизонтали, так и по вертикали, т.е. нужно представить себе, что она нанесена на шар.

Одна клеточка может входить в несколько групп одновременно.

Нужно стремиться, чтобы число групп было как можно меньше, а сами группы – как можно больше.

Каждая 1 или 0 могут входить одновременно в несколько объединений, но каждое новое объединение должно отличаться от уже имеющихся хотя бы одним новым компонентом.

5.3. Функцию в МДНФ преобразуйте в базис И-НЕ. Для этого необходимо дважды инвертировать левую и правую части выражения. Одну инверсию используем для замены знаков дизъюнкции на знаки конъюнкции, пользуясь тождеством де Моргана, другая инверсия остаётся для возвращения в исходное состояние. Таким образом, в выражении МДНФ остаются только знаки конъюнкции и инверсии – И-НЕ.

5.4. Функцию в МКНФ преобразуйте в базис ИЛИ-НЕ пользуясь тождеством де Моргана.

5.5. Определите количество ИМС, необходимое для реализации функции по формуле:

$$m = k/n,$$

где  $k$  – число элементов используемых в схеме;

$n$  – число элементов в ИМС.

5.6. Из справочника выберите тип интегральных микросхем, которые можно использовать для реализации полученных функций.

Например:

Пусть задана логическая функция от 4-х переменных  $y = f(x_1, x_2, x_3, x_4)$  в виде следующей таблицы истинности:

<b>x4</b>	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
<b>x3</b>	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
<b>x2</b>	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
<b>x1</b>	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
<b>y</b>	0	0	1	1	0	0	0	0	1	1	1	1	0	0	0	0

Контуры «1» и «0» в заполненной карте выглядят следующим образом

<b>x2x1</b>	00	01	11	10
<b>x4x3</b>				
00	0000 <b>0</b>	0001 <b>0</b>	0011 <b>1</b>	0010 <b>1</b>
01	0100 <b>0</b>	0101 <b>0</b>	0111 <b>0</b>	0110 <b>0</b>
11	1100 <b>0</b>	1101 <b>0</b>	1111 <b>0</b>	1110 <b>0</b>
10	1000 <b>1</b>	1001 <b>1</b>	1011 <b>1</b>	1010 <b>1</b>

<b>x2x1</b>	00	01	11	10
<b>x4x3</b>				
00	0000 <b>0</b>	0001 <b>0</b>	0011 <b>1</b>	0010 <b>1</b>
01	0100 <b>0</b>	0101 <b>0</b>	0111 <b>0</b>	0110 <b>0</b>
11	1100 <b>0</b>	1101 <b>0</b>	1111 <b>0</b>	1110 <b>0</b>
10	1000 <b>1</b>	1001 <b>1</b>	1011 <b>1</b>	1010 <b>1</b>

Затем для каждой группы выписываются те переменные, которые в пределах данной группы не изменяются. Если группа содержит «1» то переменные объединяются знаком « $\square$ », а над теми переменными, которые равны «0» ставится знак инверсии. Группы переменных объединяются знаком «+». Например, для группы, содержащей «1» и выделенной сплошной линией  $x_3$  равно «0» в каждой клеточке  $x_4$  – равно «1» в каждой клеточке, а  $x_1$  и  $x_2$  в одних клеточках равны «0», в других – «1». У группы выделенной пунктиром,  $x_2$  в каждой в каждой клеточке принимает значение «1»,  $x_3$  – «0».

$$Y_{\text{мднф}} = x_4 \cdot \overline{x_3} + \overline{x_3} \cdot x_2$$

Для групп, содержащих «0», всё делается наоборот. Инверсия ставится над теми переменными, которые равны «1», внутри группы переменные объединяются знаком «+», а между группами – « $\cdot$ ». Тогда получим:

$$Y_{\text{МКНФ}} = \overline{x_3} (x_2 + x_4)$$

$Y_{\text{мднф}}$  в базисе И-НЕ

$$y = \overline{\overline{x_4 \cdot \overline{x_3} + \overline{x_3} \cdot x_2}} = \overline{\overline{x_4 \cdot \overline{x_3}} \cdot \overline{\overline{x_3} \cdot x_2}}$$

Выбираем подходящую микросхему по числу входов: 2И-НЕ, 3И-НЕ или 4И-НЕ. Для данного примера выбираем микросхему 2И-НЕ (К561ЛА7 из табл.1 Приложений), т.е. элемент микросхемы выполняет операцию конъюнкции с последующей инверсией результата, имеет два входа, таких элементов в данной микросхеме четыре.  $m = k/n = 4/4 = 1$ . Получаем схему (рис.1).

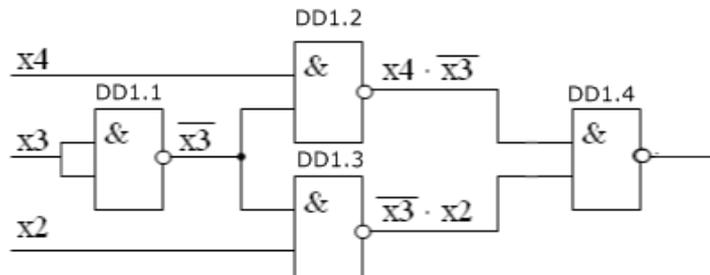


Рис.1. Схема в базисе И-НЕ

Таблица - спецификация к схеме

Обозначение в схеме	Тип ИМС	Количество	Коэффициент использования
DD1.1 – DD1.4	К561ЛА7	1	4/4

$Y_{\text{МКНФ}}$  в базисе ИЛИ-НЕ

$$y = \overline{\overline{\overline{x_3} (x_2 + x_4)}} = \overline{\overline{x_3} + (x_2 + x_4)}$$

Для реализации данного выражения выбираем логический элемент 2ИЛИ-НЕ (К561ЛЕ5 из табл.1 Приложений), элемент с двумя входами,

выполняющий операцию дизъюнкции с последующей инверсией результата, таких элементов в микросхеме четыре.  $k=2, n=4$ . Строим схему (рис.4).

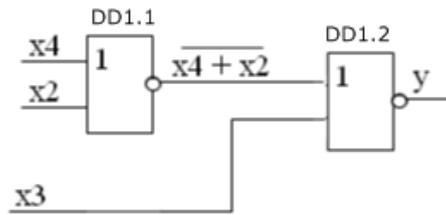


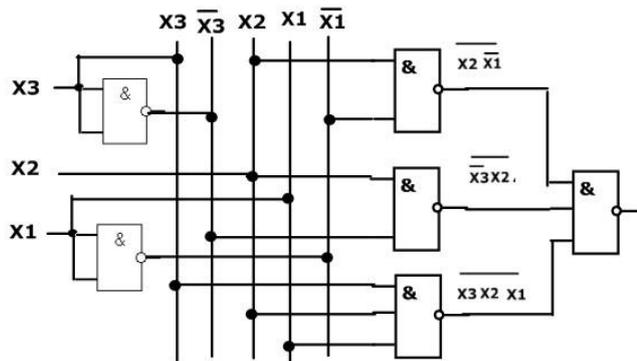
Рис.2. Схема в базисе ИЛИ-НЕ

Таблица - спецификация к схеме

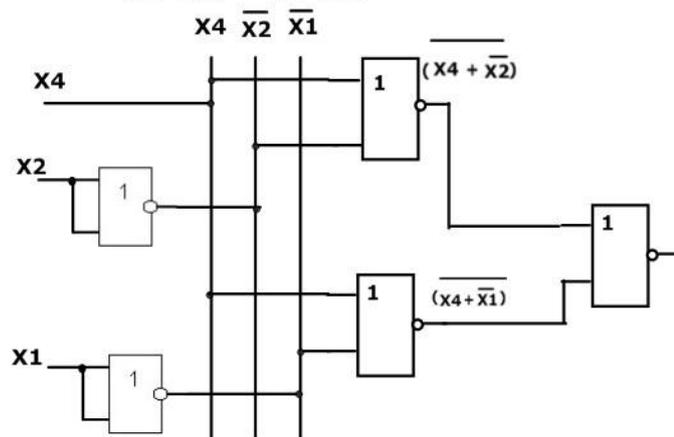
Обозначение в схеме	Тип ИМС	Количество	Коэффициент использования
DD1.1 – DD1.2	K561ЛЕ5	1	2/4

**Примеры построения схем**

$$Y_{\text{МДНФ или-не}} = \overline{x_2 \overline{x_1} x_3 x_2} \wedge \overline{x_3 x_2 x_1}$$



$$Y_{\text{МКНФ или-не}} = (x_4 + \overline{x_2}) + (x_4 + \overline{x_1})$$



**6. Указания к выполнению отчета**

Отчет должен содержать

1. Тему и цель работы, задание и порядок выполнения.
2. Исходную таблицу истинности.
3. Карту Карно заданной функции с обозначением контуров.

4. УМДНФ и УМКНФ, полученные в результате минимизации исходной функции

5. Преобразования логической функции с использованием тождества де Моргана

6. Схемы электрические принципиальные заданной функции в базисах И-НЕ, ИЛИ-НЕ.

7. УГО выбранных микросхем и основные параметры

### **7. Контрольные вопросы**

1. В чем заключается цель минимизации логической функции?

2. Сколько клеток содержит карта Карно?

3. Что такое код Грея?

### **8. Список литературы**

1. Вычислительная техника: Учебное пособие / Душкин А.В., Ланкин О.В., Чекризов Р.В. - Воронеж: Воронежский институт ФСИИ России, 2023. - электронный портал znanium.

2. Ю.В. Новиков, Скоробогатов Основы микропроцессорной техники. – 3-е изд., испр. – М.: Интернет-Университет информационных технологий; БИНОМ. Лаборатория знаний, 2020г.

## Приложения

Табл.1

*Функциональный состав логических ИМС, используемых в работе*

Тип ИМС	Выполняемая функция	№ рисунка
ЛЕ5	2ИЛИ-НЕ	1
ЛЕ 6	4ИЛИ-НЕ	2
ЛЕ10	3ИЛИ-НЕ	3
ЛА7	2И-НЕ	4
ЛА8	4И-НЕ	5
ЛА9	3И-НЕ	6

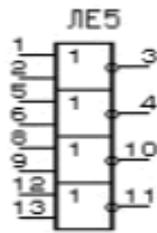


Рис.1

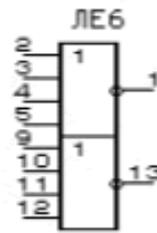


Рис.2

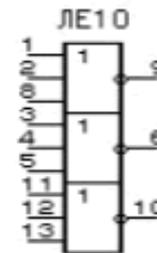


Рис.3

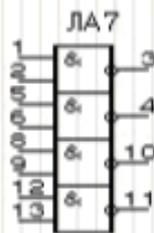


Рис.4

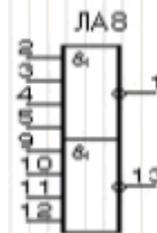


Рис.5

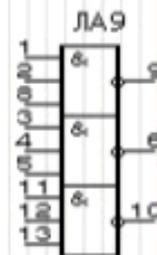


Рис.6

## ПРАКТИЧЕСКАЯ РАБОТА №7

### ПРОЕКТИРОВАНИЕ УСТРОЙСТВА НА БАЗЕ КОМБИНАЦИОННЫХ ЦИФРОВЫХ УЗЛОВ ПО ЗАДАННОЙ ТАБЛИЦЕ ИСТИННОСТИ

**1. Цель:** Приобретение практических навыков построения схем цифровых устройств на базе специальных комбинационных узлов

**2.Время выполнения работы-4час**

**3. Краткие теоретические сведения**

Дешифратором (Decoder — DC)  $m \times n$  называют комбинационное устройство с  $m$  входами и  $n$  выходами, преобразующее  $m$ -разрядный двоичный код в  $n$ -разрядный унитарный код, имеет  $n$  информационных входов и  $2^n$  выходов. Каждой комбинации логических уровней на входах будет соответствовать активный уровень на одном из  $2^n$  выходов. Помимо информационных входов дешифраторы обычно имеют дополнительные входы управления. Сигналы на этих входах разрешают функционирование дешифратора или переводят его в пассивное состояние. В дешифраторах высокого уровня унитарный код содержит единственную 1, в дешифраторах низкого уровня - единственный 0.

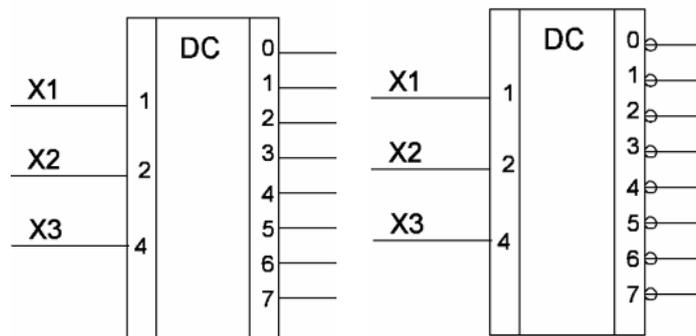


Рис.1. Условное графическое обозначение дешифратора

Мультиплексором (Multiplexer - MUX)  $M \times 1$  называют комбинационное устройство с  $M$  информационными ( $D_0, D_1, \dots, D_{M-1}$ ),  $K$  адресными ( $A_0, A_1, \dots, A_{K-1}$ ) входами и одним выходом ( $Y$ ), которое осуществляет передачу сигнала с заданного адресным кодом информационного входа на его выход. Это комбинационное цифровое устройство предназначено для поочередной передачи на один выход одного из нескольких входных сигналов, то есть для их мультиплексирования. Количество мультиплексируемых входов называется количеством каналов мультиплексора.

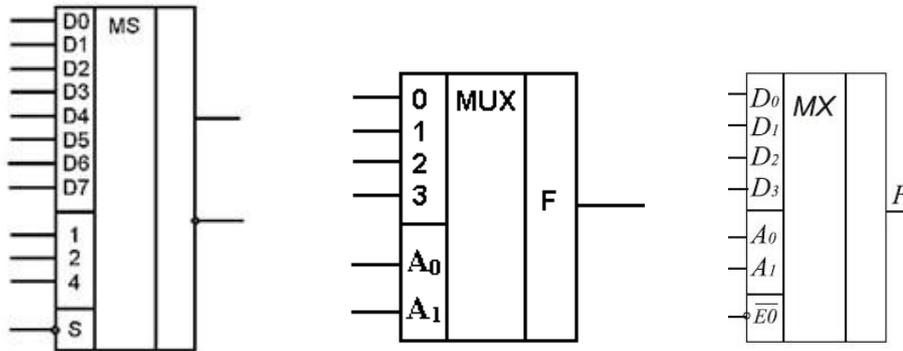


Рис.2. Условное графическое обозначение мультиплексора

Управление работой мультиплексора (выбор номера канала) осуществляется с помощью входного кода адреса. Например, для 4-канального мультиплексора необходим 2-разрядный управляющий (адресный) код, а для 16-канального — 4-разрядный код. Разряды кода обозначаются 1, 2, 4, 8 или  $A_0, A_1, A_2, A_3$  (ABC).

**Задание ч.1:** Разработать схему логического устройства, реализующего логическую функцию, заданную в варианте задания, на базе дешифратора.

#### 4. Порядок выполнения

4.1. Составьте таблицу истинности заданной функции, определяя значение логической функции при всех возможных комбинациях входных переменных.

4.2. Выпишите номера строк, в которых функций приобретает единичные и нулевые значения.

4.3. Постройте две схемы, реализующие заданную функцию: одну на дешифраторе с прямыми выходами и логическом элементе ИЛИ, а другую на дешифраторе с инверсными выходами и логическом элементе И.

#### 5. Указания к выполнению

5.1. Чтобы реализовать логическую функцию на дешифраторе нужно подать логические переменные на адресные входы дешифратора.

5.2. Выбрать наборы входных переменных, при которых функция принимает значения «1». Так как функция должна равняться лог. «1» при реализации любого из этих наборов соответствующие выходы следует подать на входы схемы ИЛИ при использовании дешифраторов с активным сигналом лог. «1» на выходе или схемы И-НЕ, если используется дешифратор с активным сигналом лог. «0» на выходе.

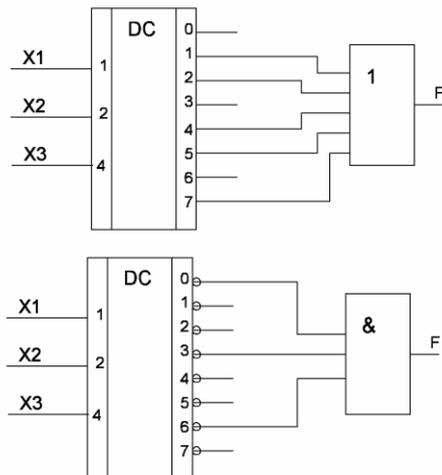
5.3. При использовании дешифраторов с активным сигналом лог. «0» удобнее выбрать наборы входных переменных, при которых функция принимает значения «0» и подать соответствующие выходы на входы схемы И.

Например: Необходимо реализовать функцию, описанную таблицей истинности, на базе дешифратора.

Таблица истинности имеет вид:

m	X <sub>3</sub>	X <sub>2</sub>	X <sub>1</sub>	y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

Функция принимает значения «1» в 1,2,4,5,7 строках, и значения «0» в 0,3 и 6. Тогда схемы реализации будут выглядеть следующим образом



## 6. Указания к оформлению отчета

Отчет должен содержать:

1. Тему и цель работы, задание и порядок выполнения.
2. Заданную функцию согласно варианту.
3. Таблицу истинности с указанием номеров строк.
4. Схемы п.п. 3.3.

**Задание ч.2:** Разработать схему логического устройства, реализующего логическую функцию, заданную в виде карты Карно в варианте задания, на базе мультиплексора.

## 4. Порядок выполнения

4.1. Составьте МКНФ заданной логической функции по карте Карно путем объединения нулей.

4.2. Составьте таблицу истинности для вновь полученной функции, используя только три младших аргумента X<sub>3</sub>, X<sub>2</sub>, X<sub>1</sub>.

4.3. Для реализации в виде схемы подайте на адресные входы мультиплексора три младших аргумента X<sub>3</sub>, X<sub>2</sub>, X<sub>1</sub>, а на соответствующие информационные входы значения функции, полученные в соответствующих строках таблицы

## 5. Указания к выполнению

5.1. Для получения  $Y_{\text{МКНФ}}$  следует произвести объединение клеток карты Карно, содержащих «0». Инверсия ставится над теми переменными, которые равны «1», внутри группы переменные объединяются знаком «+», а между группами – «·».

5.2. Чтобы заполнить таблицу истинности функции найдите ее значение при всех комбинациях входных переменных.

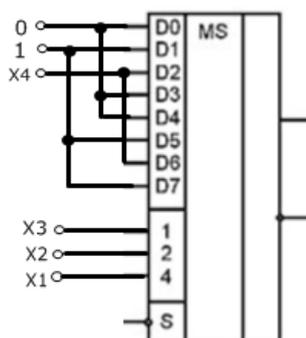
Например: пусть в результате минимизации функции, заданной в виде карты Карно  $Y_{\text{МКНФ}}$  получилось равным

$$y = (X1 + X2) \cdot (X1 + X4) \cdot (\overline{X2} + X3)$$

Таблица истинности данной функции для трех младших аргументов

X3	X2	X1	Y
0	0	0	0
0	0	1	1
0	1	0	X4
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	X4
1	1	1	1

По таблице определим, какие сигналы следует подать на информационные входы мультиплексора, чтобы реализовать функцию Y.



## 6. Указания к оформлению отчета

Отчет должен содержать:

1. Тему и цель работы, задание и порядок выполнения.
2. Заданную функцию в виде карты Карно согласно варианту с указанием выбранных контуров.
3. МКНФ функции.
4. Таблицу истинности новой функции с указанием номеров строк.
5. Схемы п.п. 3.3.

## 7. Контрольные вопросы

1. Каково назначение дешифратора?
2. Каково количество выходных шин полного дешифратора при дешифрации трехразрядного числа?
3. Приведите условное графическое обозначение дешифратора высокого уровня 3X8 и укажите, при какой входной комбинации высокий уровень (лог«1») будет на выходе под номером «3»?
4. Приведите условное графическое обозначение дешифратора низкого уровня 3X8 и укажите, при какой входной комбинации низкий уровень (лог. «0») будет на выходе под номером «6»?
5. Каково назначение мультиплексора?
6. Какой информационный вход ( $D0... D4$ ) мультиплексора подключается к выходу при подаче на адресные входы  $A0$  и  $A1$  кодов  $01$  и  $10$  соответственно?
7. Какой четырехразрядный адрес обеспечивает передачу на выход информации со следующих входов:  $D2, D10, D5, D14$ ?

### **8.Список литературы**

1. Келим Ю. М. Вычислительная техника/ Учеб. пособие для студ. СПО.– М.: Издательский центр «Академия», 2019г.
2. Вычислительная техника : учеб. пособие / Т.Л. Партыка, И.И. Попов. — 3-е изд., перераб. и доп. -М.: ФОРУМ:ИНФРА-М, 2023- электронный портал znanium.
3. Основы электроники: учебник для СПО/ О.В., Миловзоров, И.Г.Панков, — 6-е изд.перераб. и доп. —Москва: Издательство Юрайт, 2021

## ПРАКТИЧЕСКАЯ РАБОТА №8 ПОСТРОЕНИЕ СЧЕТЧИКОВ ПО ПРОИЗВОЛЬНОМУ ОСНОВАНИЮ

**1. Цель работы:** Приобретение практических навыков построения схем цифровых устройств на базе специальных комбинационных узлов

**2.Время выполнения работы-4час**

**3. Краткие теоретические сведения**

Счетчиками называют устройства для подсчета числа входных импульсов и фиксации этого числа в каком-либо коде. В процессе работы счетчик последовательно изменяет свое состояние. Количество возможных состояний называется модулем счета  $K_{сч}$  или емкостью счетчика (предельное число импульсов, которое может быть подсчитано счетчиком). Основными элементами счетчика являются триггеры, количество которых определяет число разрядов счетчика  $n$  и его модуль счета  $K_{сч}$ . Нулевое состояние всех триггеров принимается за нулевое состояние счетчика. При подаче счетных импульсов счетчик последовательно изменяет свое состояние от нулевого до максимального, равного  $K_{сч}$ . Например, при  $n=3$   $K_{сч}=8$ , т.е. счетчик имеет 8 устойчивых состояний и каждый восьмой импульс, поступающий на его вход, будет возвращать счетчик в исходное состояние (обнуление счетчика). Это свойство позволяет использовать счетчики для деления частоты. При построении счетчика триггеры соединяют последовательно. Выход каждого триггера непосредственно действует на тактовый вход следующего. Для того чтобы реализовать суммирующий счетчик, необходимо счетный вход очередного триггера подключать к инверсному выходу предыдущего. Для того чтобы изменить направление счета (реализовать вычитающий счетчик), можно осуществить

-считывание выходных сигналов счетчика не с прямых, а с инверсных выходов триггеров.

-изменение структуры связей в счетчике путем подачи на счетный вход триггера сигнала не с инверсного, а с прямого выхода предыдущего каскада.

В зависимости от способа кодирования внутренних состояний различают:

- двоичные счетчики;
- двоично-десятичные (декадные) счетчики;
- кольцевые счетчики – состояние счетчика определяется местоположением одной единственной 1 или 0;
- счетчики Джонсона – состояние счетчика определяется количеством 1 или 0.

В зависимости от способа подсчета различают следующие виды счетчиков:

- суммирующие;
- вычитающие;
- реверсивные;
- кольцевые.

По способу тактирования работы различают:

- синхронные счетчики, для работы которых требуется синхросигнал;
- асинхронные счетчики, работающие без синхросигналов.

По структурной организации, зависящей от способа подачи счетных импульсов на разряды счетчиков, различают:

- счетчики с последовательным (каскадным) переносом – счетные импульсы поступают только на вход первого разряда, а с его выхода – на вход второго и т.д. (счетчики с последовательным переносом отличаются простотой, но при этом имеют низкое быстродействие);

- счетчики с параллельным переносом – счетные импульсы поступают одновременно на все разряды (такие счетчики имеют более сложную организацию, но обеспечивают высокое быстродействие);

- параллельно-последовательные счетчики, представляющие собой комбинацию первых двух способов подачи счетных импульсов (такие счетчики используются для получения больших значений модуля счета).

Счётчик с произвольным основанием можно построить, используя микросхему счётчика, у которого имеется асинхронный вход сброса в нулевое состояние. Чтобы счётчик считал до  $N$ , достаточно в момент достижения в процессе счёта состояния  $N$  сформировать с помощью выходного состояния счётчика в этот момент и комбинационной схемы активный сигнал на входе сброса. Как только счётчик в процессе счёта переберёт все состояния от нуля до  $N-1$ , то с приходом очередного импульса на счётный вход счётчик достигает состояния  $N$  и происходит сброс его в нулевое состояние. То есть после состояния  $N-1$  счётчик к приходу следующего импульса окажется в нулевом состоянии и счёт начнётся заново.

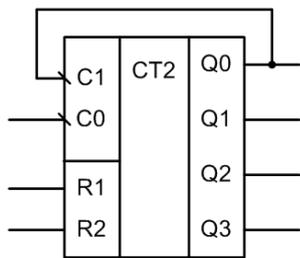


Рис.3. Условное графическое обозначение двоичного счетчика K555IE5

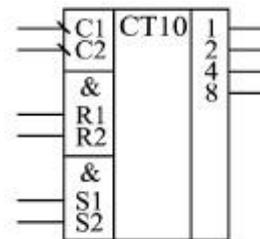


Рис.4. Условное графическое обозначение двоично-десятичного счетчика K555IE2

**Задание:** Разработать схемы двоичного и двоично-десятичного счетчика с заданным коэффициентом пересчета

#### **4. Порядок выполнения**

4.1. Представьте заданный преподавателем коэффициент пересчета в двоичном коде.

4.2. Для получения двоичного суммирующего счетчика выходы счетчика, веса которых соответствуют весам единиц в полученном двоичном числе, подсоедините ко входам логического вентиля И, выход которого подключите ко входам сброса счетчиков

4.3. Для получения двоично-десятичного счетчика заданный коэффициент пересчета представьте в двоично-десятичном коде и выходы счетчика, веса которых соответствуют весам единиц в полученном двоичном числе, подсоедините ко входам логического вентиля И, выход которого подключите ко входам сброса счетчиков

4.4. Для получения вычитающего счетчика определите число, которое будет предварительно записано в счетчик. Оно должно быть на единицу больше исходного модуля.

4.5. Представьте полученный коэффициент пересчета в двоичном коде

4.6. В соответствии с полученным числом входы счетчиков  $D_0...D_7$  подключаются либо к нулевому потенциалу, либо через резистор к положительному выводу источника питания.

#### **5. Указания к выполнению**

5.1. Для построения двоичного счетчика с произвольным модулем счета используем метод, который заключается в немедленном сбросе в “0” счетчика, установившегося в комбинацию, соответствующую числу  $K_{сч}$ .

Пример построения двоичного счетчика с коэффициентом пересчета 100.

Так как заданный модуль счета больше 16, то для реализации этого счетчика необходимо использовать два четырехразрядных счетчика. Пусть это будут микросхемы ИЕ5. ИС типа ИЕ5 представляют собой последовательные двоичные четырехразрядные счетчики, у которых первый триггер имеет индивидуальный тактовый вход  $C_1$ , а его выход  $Q_1$  не соединен с остальными разрядами счетчика. Внешний сигнал синхронизации подается на вход  $C_1$  счетчика  $D_2$ . Его выход  $Q_1$  соединен со входом  $C_2$ . Вход  $C_1$  счетчика  $D_3$  подключен к выходу  $Q_4$  счетчика  $D_2$ . В процессе подсчета импульсов счетчик последовательно принимает состояния от 0 до 100. После чего он должен сброситься в ноль, т.е. все оставшиеся комбинации должны быть исключены. Для этого, как только на его выходах появится число  $100_{10}=1100100_2$ , должен

выработаться сигнал сброса. Его можно получить, если объединить логическим И выходы Q2 первой микросхемы, и Q1 и Q2 второй микросхемы счетчика. Эту функцию выполняет вентиль 3И (элемент D1 на рис. 5), выход которого подключен ко входам сброса счетчиков.

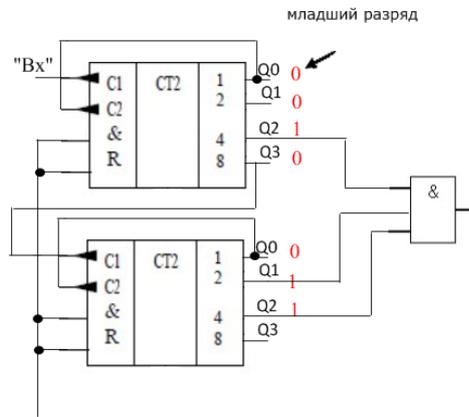


Рис.5 Схема суммирующего счетчика с коэффициентом пересчета 100 построенного на микросхемах ИЕ5

5.2. Для построения двоично-десятичного счетчика с произвольным модулем счета заданный коэффициент пересчета необходимо представить в двоично-десятичном коде, а затем выходы счетчика, веса которых соответствуют весам единиц в полученном двоичном числе, подсоединить ко входам логического вентиля И, выход которого подключить ко входам сброса счетчиков

Пример построения двоично-десятичного счетчика с коэффициентом пересчета 95

$$95_{10} = 1001\ 0101_{2-10}$$

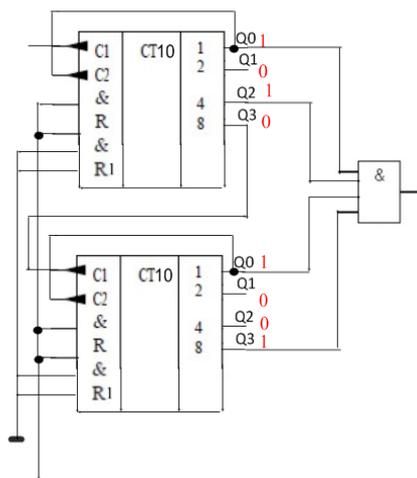


Рис.6 Схема суммирующего двоично-десятичного счетчика с коэффициентом пересчета 95, построенного на микросхемах ИЕ2

5.3. Для построения двоичного вычитающего счетчика с произвольным модулем счета используем метод предварительной загрузки числа, в основе которого лежит способ исключения начальных состояний для суммирующего



3. Ю.В. Новиков, Скоробогатов Основы микропроцессорной техники. – 3-е изд., испр. – М.: Интернет-Университет информационных технологий; БИНОМ. Лаборатория знаний, 2006г.

ЛАБОРАТОРНАЯ РАБОТА №1  
ИССЛЕДОВАНИЕ РАБОТЫ ЛОГИЧЕСКИХ ЭЛЕМЕНТОВ

**1. Цель работы:** Исследование работы цифровых логических элементов

**2. Время -2 час**

**3. Краткие теоретические сведения**

Схемы, реализующие простейшие логические функции, называются логическими элементами (ЛЭ).

На электрических схемах логические элементы обозначаются в виде прямоугольников (шириной 10...12, высота > 15) с выводами для входных (слева) и выходных (справа) переменных. Внутри прямоугольника (в левом верхнем углу) изображается символ, указывающий функциональное назначение элемента.

С помощью схем ЛЭ можно реализовать любую логическую функцию.

В соответствии с перечнем логических операций (конъюнкция, дизъюнкция и отрицание) различают три основных логических элемента (ЛЭ): **И**, **ИЛИ**, **НЕ** (рис. 1). Элементы **И**, **ИЛИ** могут иметь несколько равноправных входов (от 2 до 12) и один выход, сигнал на котором определяется комбинацией входных сигналов. Элемент **НЕ** имеет всегда только один вход.

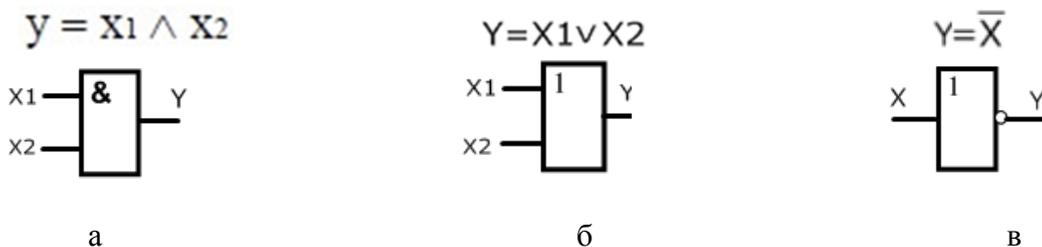


Рис. 1. Условное графическое обозначение элементов И (а), ИЛИ (б), НЕ (в)

На практике широкое применение нашли ЛЭ, совмещающие функции элементов указанных выше функционально полных систем. Это элементы **И-НЕ** и **ИЛИ-НЕ**, которые носят названия соответственно элемент Шеффера и элемент Пирса.

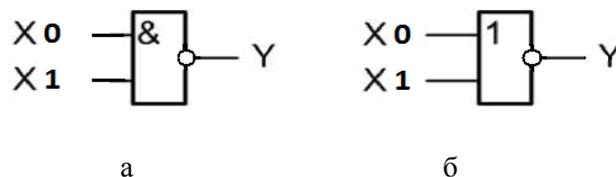
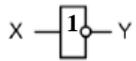
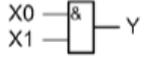
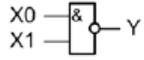
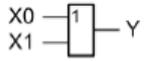
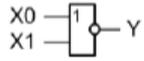
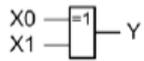


Рис. 2. Условное графическое обозначение элементов **И-НЕ** (а), **ИЛИ-НЕ** (б)

В табл. 2 представлены основные логические элементы, их обозначение, схемы и выполняемые функции.

табл. 2

Элемент	Обозначение	Схема	Функция
НЕ	ЛН		$Y = \bar{X}$
И	ЛП		$Y = X1 \wedge X0$
И-НЕ	ЛА		$Y = \overline{X1 \wedge X0}$
ИЛИ	ЛЛ		$Y = X1 \vee X0$
ИЛИ-НЕ	ЛЕ		$Y = \overline{X1 \vee X0}$
Исключающее ИЛИ	ЛШ		$Y = X1 \oplus X2$

**Примечание:** для обозначения логической операции «Исключающее ИЛИ» в логических выражениях используется символ  $\oplus$  (сложение по модулю два)

От числа входов  $n$  логического элемента зависит также количество строк таблицы истинности  $2^n$ . Например, для элемента с 3 входами таблица истинности будет содержать  $2^3=8$  строк, с 4 входами -  $2^4=16$  и т.п.

#### 4. Перечень используемого оборудования:

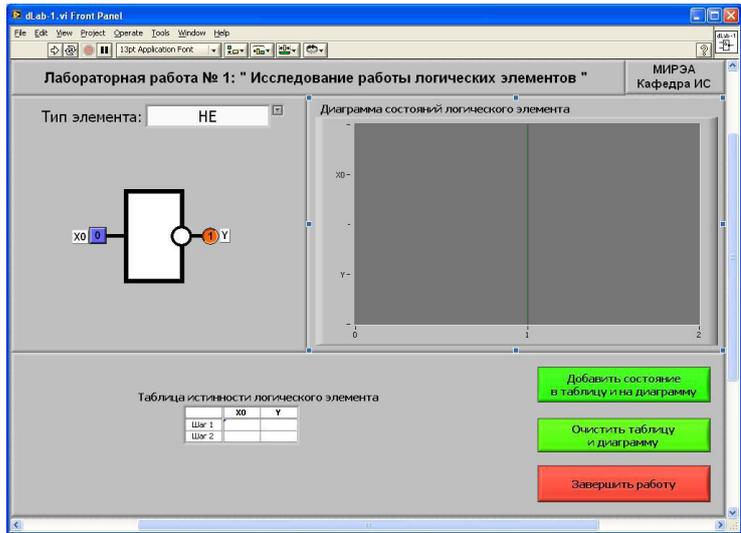
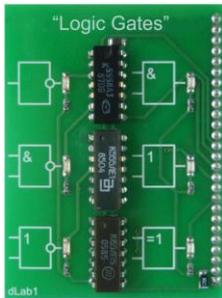
- платформа NI ELVIS II;
- лабораторный модуль dLab1 для исследования работы логических элементов.

**Задание:** Проанализируйте работу логических элементов и сделайте вывод о соответствии таблиц истинности теоретическим положениям

#### 5. Порядок выполнения работы

5.1. Установите лабораторный модуль **dLab1** на макетную плату лабораторной станции NI ELVIS.

5.2. Загрузите файл **dLab-1.vi** и запустите программу, щелкнув левой кнопкой мыши на экранной кнопке RUN. На экране появится изображение лицевой панели.



*ИССЛЕДОВАНИЕ ЛОГИЧЕСКОГО ЭЛЕМЕНТА НЕ*

5.3 . Нажмите на кнопку «**Очистить таблицу и диаграмму**».

5.4. Выберите логический элемент «НЕ». Для этого щелкните мышью на кнопке раскрытия списка элемента управления «Тип элемента». В раскрывшемся списке выберите строку с надписью «НЕ». В левой части рабочего окна появится условное графическое изображение логического элемента НЕ.

5.5. Установите на входе «X0» исследуемого логического элемента логический сигнал «0». Логический уровень изменяется при однократном нажатии с помощью манипулятора мышью на кнопку квадратной формы, расположенную около входа логического элемента. При этом на кнопке отображается состояние входа («0» - синий цвет или «1» - оранжевый цвет). На индикаторе круглой формы, расположенном около выхода логического элемента, будет отображено состояние выходного сигнала в соответствии с логической функцией исследуемого элемента.

5.6. Занесите логические состояния входа и выхода логического элемента «НЕ» в таблицу истинности и на диаграмму состояний. Для этого нажмите на кнопку «**Добавить состояние в таблицу и на диаграмму**».

5.7. Установите на входе «X0» исследуемого логического элемента логический сигнал «1», и с помощью кнопки «**Добавить состояние в таблицу и на диаграмму**» занесите логические состояния входа и выхода логического элемента «НЕ» в таблицу истинности и на диаграмму состояний.

*Таблица 3*

	<b>X0</b>	<b>Y</b>
Шаг1		
Шаг2		

5.8. Полученные таблицу истинности и диаграмму состояний скопируйте в отчет.

*ИССЛЕДОВАНИЕ ЛОГИЧЕСКИХ ЭЛЕМЕНТОВ И, И-НЕ, ИЛИ, ИЛИ-НЕ, Искл.ИЛИ*

5.9. Нажмите на кнопку «**Очистить таблицу и диаграмму**».

5.10. Выберите логический элемент И. Для этого щелкните мышью на кнопке раскрытия списка элемента управления «Тип элемента». В раскрывшемся списке выберите строку с надписью «И». В левой части рабочего окна появится условное графическое изображение логического элемента И.

5.11. Установите на входах «X0» и «X1» логического элемента значения сигналов, приведенные в первой строке табл.3.

**Таблица 3**

<i>Вход X1</i>	<i>Вход X0</i>
0	0
0	1
1	0
1	1

5.12. Занесите логические состояния входов и выходов в таблицу истинности и на диаграмму состояний. Для этого на лицевой панели ВП нажмите на кнопку «Добавить состояние в таблицу и на диаграмму». Аналогично занесите логические состояния входов и выходов для остальных строк таблицы 2.

5.13. Скопируйте полученные таблицу истинности и диаграмму состояний в отчет.

5.14. Повторите пп. 4.10. – 4.13. для логических элементов «И-НЕ», «ИЛИ», «ИЛИ-НЕ», «Искл.ИЛИ».

5.15. Запишите выводы по проведенным исследованиям и ответьте на контрольные вопросы.

## **6. Указания к выполнению отчета**

Отчет должен содержать:

- наименование и цель работы.
- перечень используемого оборудования
- схемы измерений
- таблицы истинности ЛЭ
- диаграммы состояний
- выводы по проведенным исследованиям
- ответы на контрольные вопросы.

## **7. Контрольные вопросы**

1. Назовите основные логические операции булевой алгебры.
2. От чего зависит количество комбинаций в таблице истинности логической функции?
3. Приведите условное графическое обозначение логических элементов И, ИЛИ, НЕ, И-НЕ и поясните их работу по активному уровню.

## **8. Список литературы**

1. Келим Ю. М. Вычислительная техника/ Учеб. пособие для студ. СПО.– М.: Издательский центр «Академия», 2019г.
2. Основы электроники: учебник для СПО/ О.В., Миловзоров, И.Г.Панков, — 6-е изд. перераб. и доп. — Москва: Издательство Юрайт, 2021
3. Колпакова Т.И., Калиенко И.В., Методическое пособие «Логические элементы», РКРИПТ, 2017
4. Ю.В. Новиков, Скоробогатов Основы микропроцессорной техники. – 3-е изд., испр. – М.: Интернет-Университет информационных технологий; БИНОМ. Лаборатория знаний, 2020г.

## ЛАБОРАТОРНАЯ РАБОТА №2

## ИССЛЕДОВАНИЕ РАБОТЫ ШИФРАТОРА

**1. Цель работы:** Изучение принципов работы и режимов функционирования шифратора

**2. Время выполнения работы-4час**

**3. Краткие теоретические сведения**

Шифратор это комбинационное цифровое устройство, которое осуществляет преобразование одиночного сигнала, выражающего код десятичного числа в n-разрядный двоичный код. При n выходах простейший полный шифратор должен иметь  $2^N$  входов. Активный сигнал поступает только на один из входов в данный момент времени. Каждому входу с активным сигналом соответствует определённая комбинация выходных сигналов.

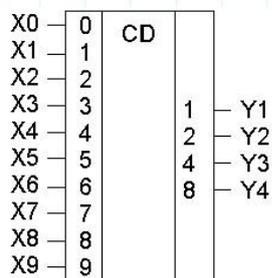


Рис.1 Условное графическое обозначение шифратора

Десятичное число	Двоичный код			
	Y4	Y3	Y2	Y1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

Рис.2 Таблица истинности шифратора десятичного кода в четырехразрядный двоичный код

Шифраторы классифицируют по ряду признаков.

*По числу входов различают:*

- полные шифраторы, число входов которых  $M = 2^N$ ;
  - неполные шифраторы, имеющих число входов  $M < 2^N$ .
- где N-число выходов

*По уровням входных и выходных сигналов выделяют:*

- шифраторы высокого уровня, активные сигналы на входах и выходах которых имеют уровень логической единицы;
- шифраторы низкого уровня, активные входные и выходные сигналы которых соответствуют уровню логического нуля.

*По функциональной значимости входов* шифраторы разделяют на две группы:

- шифраторы с равнозначными функциями входов, в которых все входы равноценны и при подаче на любой из них активного уровня сигнала на выходе формируется двоичный код. В таких шифраторах нельзя подавать несколько входных сигналов одновременно от разных источников, т. е. должна соблюдаться очередность подачи сигналов от разных источников. Если на один из входов шифратора подан сигнал, остальные входы шифратора должны быть заблокированы;

▪ приоритетные шифраторы, в которых возможна одновременная подача на входы сигналов от разных источников, однако только один из них, имеющий больший приоритет, выполнит функцию формирования выходного кода. Как правило, наивысший приоритет назначается входу с самым высоким порядковым номером.

Микросхема К555ИВ1 представляет собой приоритетный шифратор низкого уровня, преобразующий унитарный код “1 из 8” в двоичный трехразрядный код.

Данная интегральная микросхема имеет следующий набор входных и выходных сигналов:

восемь информационных входов  $X_0, X_1, \dots, X_7$ ;

три информационных выхода  $Y_0, Y_1, Y_2$ ;

вход  $E$  разрешения работы данного шифратора;

выход  $EO$  разрешения работы других шифраторов при каскадировании;

выходной сигнал группового переноса  $G$ .

Работа дешифратора разрешена при подаче нуля на вход разрешения  $E$  (enable). При этом на выходах кода  $Y_0, Y_1, Y_2$  формируется инверсный двоичный код номера активной входной линии. При одновременном поступлении нескольких входных сигналов формируется выходной код, соответствующий входу с наибольшим номером. То есть старшие входы имеют приоритет перед младшими. Поэтому такой шифратор называется приоритетным. При отсутствии входных сигналов формируется выходной код 111. Единичный сигнал на входе  $E$  запрещает работу шифратора (все выходные сигналы устанавливаются в единицу).

На выходе  $G$  вырабатывается ноль при приходе любого активного входного сигнала. Это позволяет отличить ситуацию поступления сигнала на вход  $X_0$  от ситуации отсутствия сигналов на всех входах.

Выход  $EO$  становится нулевым при отсутствии входных сигналов, если при этом разрешена работа шифратора нулевым сигналом на входе  $E$ .

Работа устройства иллюстрируется таблицей состояний табл.1

$E$	<i>Входы</i>								<i>Выходы</i>				
	$X_7$	$X_6$	$X_5$	$X_4$	$X_3$	$X_2$	$X_1$	$X_0$	$Y_2$	$Y_1$	$Y_0$	$G$	$EO$
1	x	x	x	x	x	x	x	x	1	1	1	1	1
0	1	1	1	1	1	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	0	1	1	1	0	1
0	1	1	1	1	1	0	x	x	1	0	1	0	1
0	1	1	1	1	0	x	x	x	1	0	0	0	1
0	1	1	1	0	x	x	x	x	0	1	1	0	1
0	1	1	0	x	x	x	x	x	0	1	0	0	1
0	1	0	x	x	x	x	x	x	0	0	1	0	1
0	0	x	x	x	x	x	x	x	0	0	0	0	1

**Примечание:** символ  $x$  указывает на то, что состояние соответствующего сигнала не имеет значения, т.е. не влияет на состояние выходного кода.

Сигналы EI и EO используются для наращивания разрядности шифратора. На рисунке 3 приведена схема построения шифратора 16X4 на основе двух шифраторов 8X3.

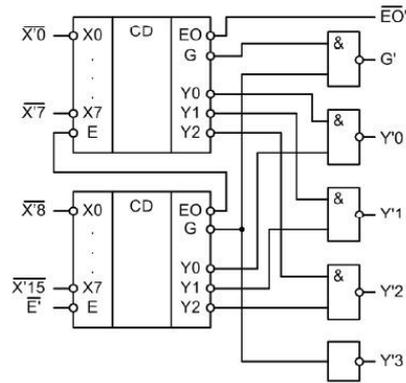


Рис.3 Схема построения шифратора 16X4

Шифраторы также могут быть использованы при организации клавиатуры для формирования кода нажатой клавиши. При этом каждому входу шифратора соответствует отдельная клавиша. Если ни одна из них не нажата, об этом свидетельствует единичное значение сигнала G. При нажатии на какую-либо клавишу выход G переходит в единичное состояние, а на информационных выходах формируется код нажатой клавиши. При использовании приоритетного шифратора в случае одновременном нажатии нескольких клавиш формируется код клавиши с наибольшим приоритетом.

#### 4.Перечень используемого оборудования:

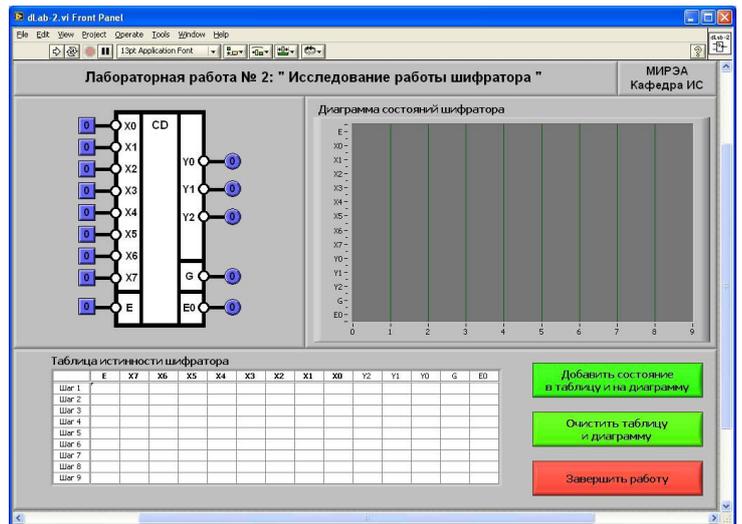
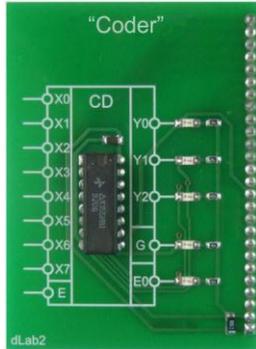
- лабораторная станция NI ELVIS II
- лабораторный модуль dLab2 для исследования работы шифратора

**Задание:** По таблице истинности и диаграмме состояний проанализируйте работу шифратора и определите, при каких условиях активный низкий уровень появляется на выходах GS и E0, какой логический сигнал на входе управления «E» дешифратора является активным, является ли шифратор приоритетным

#### 5.Порядок выполнения.

5.1. Установите лабораторный модуль **dLab2** на макетную плату лабораторной станции NI ELVIS.

5.2. Загрузите файл **dLab-2.vi** и запустите программу, щелкнув левой кнопкой мыши на экранной кнопке RUN.



5.3. Нажмите на кнопку «Очистить таблицу и диаграмму».

5.4. Установите на входе «Е» шифратора логический сигнал «0», а на входах «X0», «X1», «X2», «X3», «X4», «X5», «X6», и «X7» – значения сигналов, приведенные в первой строке табл.1.

табл.1

Ex=0	X7	X6	X5	X4	X3	X2	X1	X0	Y2	Y1	Y0	G	E0
Шаг1													
Шаг2													
Шаг3													
Шаг4													
Шаг5													
Шаг6													
Шаг7													
Шаг8													
Шаг9													

5.5. Занесите логические состояния входов и выходов шифратора первой строки в таблицу истинности и на диаграмму состояний. Для этого на лицевой панели ВП нажмите на кнопку «Добавить состояние в таблицу и на диаграмму». Аналогично занесите информацию остальных строк.

5.6. Скопируйте полученную таблицу истинности и диаграмму состояний в отчет

5.7. Установите на входе «Е» шифратора логический сигнал «1» и повторите исследование работы шифратора.

Ex=1	X7	X6	X5	X4	X3	X2	X1	X0	Y2	Y1	Y0	G	E0
Шаг1													
Шаг2													
Шаг3													
Шаг4													

Шаг5													
Шаг6													
Шаг7													
Шаг8													
Шаг9													

5.8. По таблице истинности и временной диаграмме определите, какой логический сигнал на входе управления «Е» дешифратора является активным.

5.9. По таблице истинности и временной диаграмме определите, при каких условиях активный низкий уровень появляется на выходах GS (групповой сигнал) и E0 (разрешение от выхода).

5.10. Проверьте, что исследуемый шифратор является приоритетным. Для этого сначала установите вход управления «Е» в состояние «0», а все информационные входы в состояние «1». Затем переключите любые два информационных входа, например, «Х6» и «Х3» в состояние «0». Определите, сопоставив выходной сигнал шифратора с полученной ранее таблицей истинности, какой вход из двух задействованных имеет больший приоритет.

5.11. Запишите выводы по проведенным исследованиям.

## 6. Указания к выполнению отчета

Отчет должен содержать:

1. Тему и цель работы, перечень используемого оборудования, задание и порядок выполнения
2. Таблицу истинности и временную диаграмму при логическом «0» на входе «Е» шифратора.
3. Таблицу истинности и временную диаграмму при логической «1» на входе «Е» шифратора.
4. Выводы по проведенным исследованиям

## 7. Контрольные вопросы

1. Каково назначение шифратора? Приведите условное графическое обозначение шифратора
2. Каково назначение входа управления в шифраторе? Как влияет сигнал управления на выходные функции шифратора?
3. Какой шифратор называется приоритетным?
4. Какой шифратор называется полным?

## 8. Список литературы

1. Келим Ю. М. Вычислительная техника/ Учеб. пособие для студ. СПО. – М.: Издательский центр «Академия», 2019г.

2. Основы электроники: учебник для СПО/ О.В., Миловзоров, И.Г.Панков, — 6-е изд.перераб. и доп. —Москва: Издательство Юрайт, 2021

3. Руководство пользователя NI ELVIS II. Комплект виртуальных измерительных приборов для учебных лабораторий. Издательство - NATIONAL INSTRUMENTS, 2008

5. Ю.В. Новиков, Скоробогатов Основы микропроцессорной техники. – 3-е изд., испр. – М.: Интернет-Университет информационных технологий; БИНОМ. Лаборатория знаний, 2020г.

ЛАБОРАТОРНАЯ РАБОТА №3

ИССЛЕДОВАНИЕ РАБОТЫ СУММАТОРА

**1.Цель работы:** Изучение принципов работы и режимов функционирования сумматора

**2.Время выполнения-4 час**

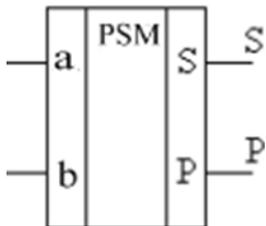
**3.Краткие теоретические сведения**

Сумматор - цифровое устройство для арифметического сложения двоичных чисел

По количеству одновременно обрабатываемых разрядов различают:

- одноразрядные сумматоры
- многоразрядные сумматоры

Полусумматором называется устройство, предназначенное для сложения двух одноразрядных переменных, имеющее два входа и два выхода и формирующее из сигналов входных слагаемых сигналы суммы и переноса в старший разряд.



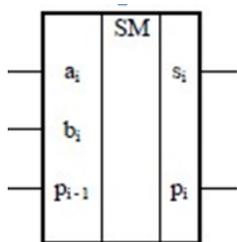
a	b	S	P
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

a и b- слагаемые (операнды)

S-сумма

P-выход переноса в старший разряд

Одноразрядным полным сумматором называется устройство, предназначенное для сложения двух одноразрядных кодов, имеющее три входа и два выхода, и формирующее из сигналов входных слагаемых и сигнала переноса из младших разрядов сигналы суммы и переноса в старший разряд.



ai	bi	pi-1	si	pi
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1
$a_i + b_i + p_{i-1}$				

ai и bi - слагаемые (операнды)

P i-1 – вход переноса из младшего разряда

si-сумма

pi-выход переноса в старший разряд

Многоразрядным сумматором называется устройство, предназначенное для сложения двух многоразрядных кодов, формирующее на выходе код суммы и сигнал переноса в случае, если результат сложения не может быть представлен кодом, разрядность которого совпадает с разрядностью кодов слагаемых.

В последовательных сумматорах операция сложения выполняется последовательно разряд за разрядом, начиная с младшего. В параллельных сумматорах все разряды входных кодов суммируются одновременно.

**Классификация сумматоров**

Комбинационные сумматоры – устройства, не имеющие собственной памяти.

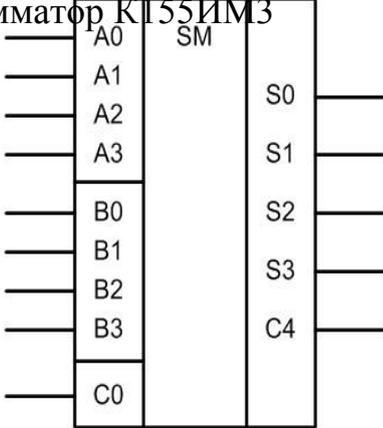
Накапливающие сумматоры, снабженные собственной внутренней памятью, в которой аккумулируются результаты выполненной операции. При этом каждое очередное слагаемое прибавляется к уже имевшемуся в устройстве значению.

В синхронных сумматорах время выполнения операции арифметического суммирования двух кодов не зависит от вида самих кодов и всегда остается постоянным. В асинхронных сумматорах время выполнения операции зависит от вида слагаемых. Поэтому по завершении выполнения суммирования необходимо вырабатывать специальный сигнал завершения операции.

В зависимости от используемой системы счисления различают двоичные, двоично-десятичные и другие типы сумматоров.

Для увеличения быстродействия многоразрядного сумматора применяется схема ускоренного переноса, которая в соответствии с состоянием сигналов на информационных входах и входного сигнала переноса формирует выходной сигнал переноса. Такой сумматор называется сумматором с параллельным переносом. На этом принципе построен

четырёхразрядный сумматор К155ИМ3



Путем соединения выводов переноса C0, C4 четырехразрядных сумматоров в последовательную цепь можно построить сумматоры с разрядностью 8, 12, 16 и т.д. Такой многоразрядный сумматор называют

сумматором с последовательным групповым переносом.

**4. Перечень используемого оборудования**

- лабораторная станция NI ELVIS II
- лабораторный модуль dLab5 для исследования работы сумматора

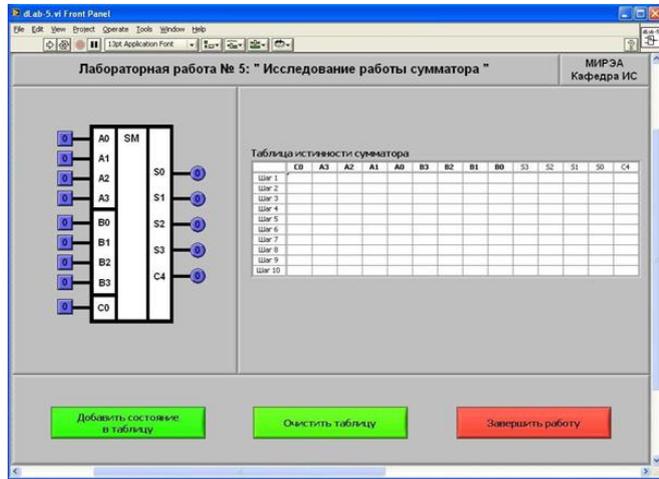
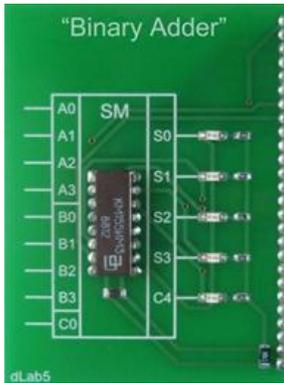
**Задание:**

По таблице истинности и диаграмме состояний проанализируйте работу микросхемы K155ИМ3 сравните полученные результаты с расчетами, выполненными вручную,

**5.Порядок выполнения.**

5.1. Установите лабораторный модуль **dLab5** на макетную плату лабораторной станции NI ELVIS.

5. 2. Загрузите файл dLab-5.vi и запустите программу, щелкнув левой кнопкой мыши на экранной кнопке RUN



5.3. Нажмите на кнопку «Очистить таблицу».

5.4. Установите на входах «C0», «A0», «A1», «A2», «A3», «B0», «B1», «B2» и «B3» – значения сигналов, приведенные в первой строке табл.1. Логический уровень изменяется при однократном нажатии с помощью манипулятора мышь на кнопку квадратной формы, расположенную около соответствующего входа. При этом на кнопке отображается состояние входа («0» - синий цвет или «1» - оранжевый цвет). На индикаторах круглой формы, расположенных около выходов «S0»,«S1»,«S2»,«S3»и «C4» сумматора, будет отображено состояние его выходных сигналов.

Таблица 1

Вход C0	Вход A3	Вход A2	Вход A1	Вход A0	Вход B3	Вход B2	Вход B1	Вход B0	S3	S2	S1	S0	C4
0	0	0	1	0	0	1	0	0					
0	1	0	0	1	1	1	0	1					
0	0	1	0	1	0	1	1	0					
0	1	0	1	1	0	1	1	1					
0	1	1	1	1	1	1	1	1					
1	0	0	1	1	0	1	0	1					
1	0	0	1	0	1	0	0	0					
1	1	0	0	1	0	0	1	1					

1	1	1	1	0	1	1	1	0					
1	1	1	1	1	1	1	1	1					

5.5. Занесите логические состояния входов и выходов сумматора в таблицу истинности. Для этого на лицевой панели ВП нажмите на кнопку «Добавить состояние в таблицу».

5.6. Повторите пп.4.4 – 4.5 для остальных строк табл.5.2.

5.7. Скопируйте полученную таблицу истинности в отчет

5.8. Проверьте полученные результаты сложения двоичных чисел с помощью уравнения

$$C_0 + 2^0(A_0 + B_0) + 2^1(A_1 + B_1) + 2^2(A_2 + B_2) + 2^3(A_3 + B_3) = \\ = 2^0 S_0 + 2^1 S_1 + 2^2 S_2 + 2^3 S_3 + 2^4 C_4,$$

5.9. Результаты проверки занесите в отчет в таблицу 2.

C0	A3	A2	A1	A0	B3	B2	B1	B0	S3	S2	S1	S0	C4
0	0	0	1	0	0	1	0	0					
0	1	0	0	1	1	1	0	1					
0	0	1	0	1	0	1	1	0					
0	1	0	1	1	0	1	1	1					
0	1	1	1	1	1	1	1	1					
1	0	0	1	1	0	1	0	1					
1	0	0	1	0	1	0	0	0					
1	1	0	0	1	0	0	1	1					
1	1	1	1	0	1	1	1	0					
1	1	1	1	1	1	1	1	1					

5.10. Запишите выводы по проведенным исследованиям

5.11. Ответьте на контрольные вопросы

## 6. Указания к выполнению отчета

Отчет должен содержать:

1. Тему и цель работы, перечень используемого оборудования, задание и порядок выполнения

2. Таблицу истинности и результаты вычислений

3. Выводы по проведенным исследованиям

4. Ответы на контрольные вопросы

## 7. Контрольные вопросы

1. Что такое сумматор ?

2. В чем отличие накапливающих сумматоров от комбинационных?

3. Как классифицируются сумматоры в зависимости от системы счисления?

4. Какой сумматор называется полусумматором? Поясните его работу с помощью таблицы истинности и схемы.

5. Какой сумматор называется одноразрядным полным сумматором? Поясните его работу с помощью таблицы истинности и схемы.

## **8. Список литературы**

1. Келим Ю. М. Вычислительная техника/ Учеб. пособие для студ. СПО.– М.: Издательский центр «Академия», 2019г.
2. Основы электроники: учебник для СПО/ О.В., Миловзоров, И.Г.Панков, — 6-е изд. перераб. и доп. — Москва: Издательство Юрайт, 2021
3. Руководство пользователя NI ELVIS II. Комплект виртуальных измерительных приборов для учебных лабораторий. Издательство - NATIONAL INSTRUMENTS, 2018
4. Ю.В. Новиков, Скоробогатов Основы микропроцессорной техники. – 3-е изд., испр. – М.: Интернет-Университет информационных технологий; БИНОМ. Лаборатория знаний, 2020г.

ЛАБОРАТОРНАЯ РАБОТА №4  
ИССЛЕДОВАНИЕ РАБОТЫ АСИНХРОННОГО RS-ТРИГГЕРА

**1.Цель работы:** Изучение работы асинхронного RS-триггера

**2.Время выполнения – 2 час**

**3.Краткие теоретические сведения**

Триггер – устройство, которое может находиться в одном из двух устойчивых состояний. Переход из одного устойчивого состояния в другое происходит скачкообразно под воздействием управляющих сигналов. Одно из этих устойчивых состояний обозначается как состояние лог. 0, другое - состояние лог. 1, т.е. триггер используется для хранения значения одной логической переменной

Триггер имеет два выхода: прямой Q и инверсный. Уровнями напряжения на этих выходах определяется состояние, в котором находится триггер: если напряжение на выходе Q соответствует уровню лог. 0 ( $Q = 0$ ), то принимается, что триггер находится в состоянии лог. 0, при  $Q = 1$  триггер находится в состоянии лог. 1. Логический уровень на инверсном выходе представляет собой инверсию состояния триггера (в состоянии  $Q = 1$  и наоборот).

Поведение триггера описывается таблицей переходов. Различают полную и сокращенную таблицу переходов. В полной таблице переходов определяется последующее состояние триггера  $Q(t+1)$  в зависимости от состояний входных информационных сигналов и предыдущего состояния триггера:

$$Q(t+1) = f(Q(t), x_i(t)).$$

Сокращенная таблица переходов не принимает в расчёт предыдущее состояние триггера, поскольку поведение триггера полностью определяется состоянием входных сигналов.

По характеру реакции на входные сигналы триггеры делятся на два типа: асинхронные и синхронные. Асинхронный триггер характеризуется тем, что входные сигналы действуют на состояние триггера непосредственно с момента их подачи на входы, в синхронных триггерах - только при подаче синхронизирующего сигнала на управляющий вход С. Если триггер имеет хотя бы один вход синхронизации, то он считается синхронным. Асинхронный триггер не имеет входов синхронизации.

Асинхронный RS-триггер является базовым при создании более сложных триггеров. В простейшем случае асинхронный RS-триггер имеет два входа: S (Set) – вход установки триггера в единичное состояние, R (Reset) - вход установки триггера в нулевое состояние. Активный сигнал по входу S в момент появления заставляет триггер перейти в единичное состояние. Активный сигнал по входу R в момент появления заставляет триггер перейти в нулевое состояние.

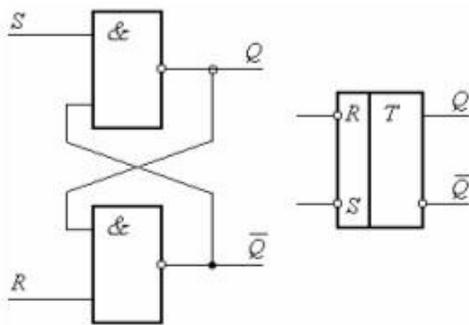


Рис 1. УГО RS-триггера с инверсными входами и схема на логических элементах И-НЕ

Поведение триггера можно определить сокращённой таблицей истинности (табл. 1), в которой сигналы на входах R и S определены для момента времени n, а состояние триггера определяется для следующего момента времени, который определяют как n+1.

Таблица 1

$S_n$	$R_n$	$Q_{n+1}$	<b>Примечание</b>
0	0	-	запрещено
0	1	1	запись единицы
1	0	0	запись нуля
1	1	$Q_n$	хранение

Поведение триггера также можно описать таблицей переходов (табл. 2). Эта таблица определяет значения сигналов на входах, при которых происходит переход триггера из исходного состояния  $Q_n$  в состояние  $Q_{n+1}$ . Исходное и конечное состояние триггера записаны, соответственно в столбцах  $Q_n$  и  $Q_{n+1}$ , а значения сигналов в момент времени «n» на его входах - в столбцах  $S_n$  и  $R_n$

Таблица 2

$Q_n$	$S_n$	$R_n$	$Q_{n+1}$
0	1	×	0
0	0	1	1
1	1	0	0
1	×	1	1

#### 4. Перечень используемого оборудования:

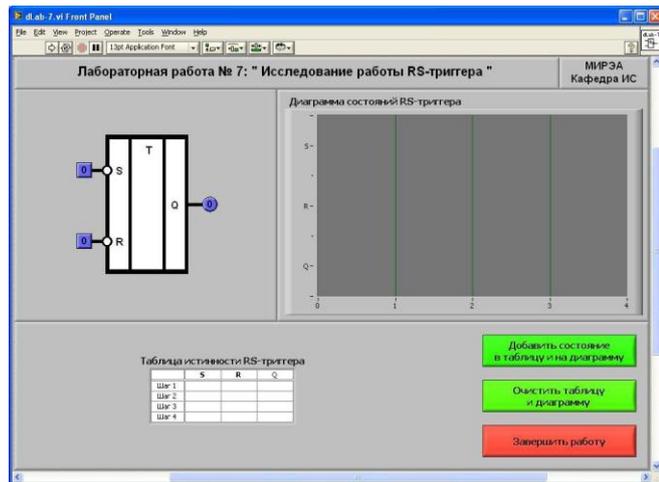
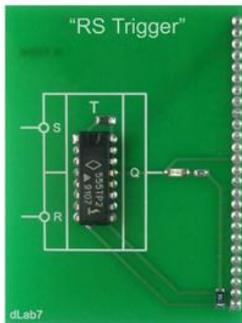
- лабораторная станция NI ELVIS II
- лабораторный модуль dLab7 для исследования работы триггера

**Задание:** По таблице истинности и диаграмме состояний проанализируйте работу триггера, заполните таблицу переходов триггера

#### 5. Порядок выполнения.

5.1. Установите лабораторный модуль **dLab7** на макетную плату лабораторной станции NI ELVIS.

5.2. Загрузите файл **dLab-7.vi** и запустите программу, щелкнув левой кнопкой мыши на экранной кнопке RUN.



5.3. Нажмите на кнопку «**Очистить таблицу и диаграмму**».

5.4. Установите на входах «S» и «R» триггера значения сигналов, приведенные в первой строке табл. 3. Логический уровень изменяется при однократном нажатии с помощью манипулятора мышь на кнопку квадратной формы, расположенную около соответствующего входа. При этом на кнопке отображается состояние входа («0» - синий цвет или «1» - оранжевый цвет). На индикаторах круглой формы, расположенных около выходов триггера, будет отображено состояние выходных сигналов.

таблица. 3

Вход R	Вход S
0	0
0	1
1	0
1	1

таблица 4

	S	R	Q	/Q
Шаг1				
Шаг2				
Шаг3				
Шаг4				

5.5. Занесите логические состояния входов и выходов триггера в таблицу истинности и на диаграмму состояний. Для этого на лицевой панели ВП нажмите на кнопку «**Добавить состояние в таблицу и на диаграмму**». Аналогично занесите информацию остальных строк. Заполните таблицу 4 отчета.

5.6. Изменяя с помощью элементов управления «S» и «R» состояние входных сигналов триггера, заполните в отчете таблицу переходов RS-триггера (табл.5.)

таблица. 5

Выход $Q_n$	Вход R	Вход S	Выход $Q_{n+1}$
0	x	0	
0	0	1	
1	1	0	
1	0	x	

$Q_n$  – состояние триггера до подачи управляющих сигналов;

$Q_{n+1}$  – состояние триггера после подачи управляющих сигналов;

x – любое состояние входа.

5.7. Запишите выводы по проведенным исследованиям и ответьте на контрольные вопросы

### **6. Указания к выполнению отчета**

Отчет должен содержать:

- наименование и цель работы.
- перечень используемого оборудования
- таблицы истинности, переходов и временные диаграммы работы триггера в исследуемых режимах
- выводы по проведенным исследованиям
- ответы на контрольные вопросы.

### **7. Контрольные вопросы**

1. Что такое триггер? Почему триггеры называются устройствами последовательного типа?

2. Опишите принцип работы RS-триггера.

3. Какая комбинация входных сигналов RS-триггера считается запрещенной?

4. Какое минимальное число входов у асинхронного RS-триггера? Опишите их назначение.

5. На каких логических элементах могут быть построены асинхронные RS-триггеры? Определите для каждого случая запрещенную комбинацию сигналов

### **8. Список литературы**

1. Келим Ю. М. Вычислительная техника/ Учеб. пособие для студ. СПО. – М.: Издательский центр «Академия», 2019г.

2. Основы электроники: учебник для СПО/ О.В., Миловзоров, И.Г. Панков, — 6-е изд. перераб. и доп. — Москва: Издательство Юрайт, 2021

3. Руководство пользователя NI ELVIS II. Комплект виртуальных измерительных приборов для учебных лабораторий. Издательство - NATIONAL INSTRUMENTS, 2008

4. Ю.В. Новиков, Скоробогатов Основы микропроцессорной техники. – 3-е изд., испр. – М.: Интернет-Университет информационных технологий; БИНОМ. Лаборатория знаний, 2006г.

## ИССЛЕДОВАНИЕ РАБОТЫ JK-ТРИГГЕРА

**1.Цель работы:** Изучение работы универсального JK-триггера

**2.Время выполнения-4час**

**3. Краткие теоретические сведения**

Триггером называется устройство, способное формировать два устойчивых значения выходного сигнала и скачкообразно изменять эти значения под действием внешнего управляющего сигнала. Способность формировать на выходе два устойчивых значения сигнала, которые могут поддерживаться без изменения сколь угодно длительный промежуток времени, и позволяет применять триггер в качестве элемента памяти.

Состояние триггера определяют по выходному сигналу. В нем может храниться либо 0, либо 1. Обычно триггер наряду с основным прямым выходом  $Q$  имеет еще инверсный выход, потенциал которого имеет значение, обратное значению прямого выхода, (если  $Q = 1$ , то  $Q' = 0$ ). Состоянию триггера 1 соответствует на выходе  $Q$  высокий уровень сигнала (1), а на выходе  $Q'$  низкий (0). Состоянию триггера 0 соответствует на выходе  $Q$  низкий уровень сигнала (0), а на выходе  $Q'$  высокий (1).

Триггеры имеют различные типы входов.

R (от английского RESET) - отдельный вход установки в состояние 0;

S (от английского SET) - отдельный вход установки в состояние 1;

K - вход установки универсального триггера в состояние 0;

J - вход установки универсального триггера в состояние 1;

T - счетный вход;

D (от английского DELAY) - информационный вход установки триггера в состояние, соответствующее логическому уровню на этом входе;

C - управляющий (синхронизирующий) вход.

По характеру реакции на входные сигналы триггеры делятся на два типа: асинхронные и синхронные. Асинхронный триггер характеризуется тем, что входные сигналы действуют на состояние триггера непосредственно с момента их подачи на входы, в синхронных триггерах - только при подаче синхронизирующего сигнала на управляющий вход C.

Синхронные триггеры подразделяются на триггеры со статическим и динамическим управлением по входу C. Триггеры со статическим (потенциальным) управлением воспринимают информационные сигналы при подаче на C-вход уровня 1 (прямой C-вход) или 0 (инверсный C-вход).

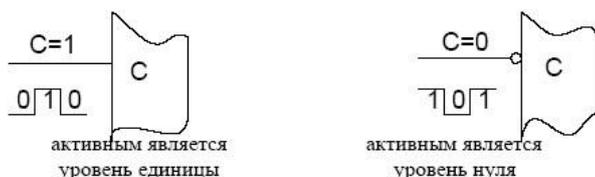


Рис.1.Статическое (потенциальное) управление

Триггеры с динамическим управлением воспринимают информационные

сигналы при изменении сигнала на С-входе от 0 к 1 (прямой динамический С-вход) или от 1 к 0 (инверсный динамический С-вход).

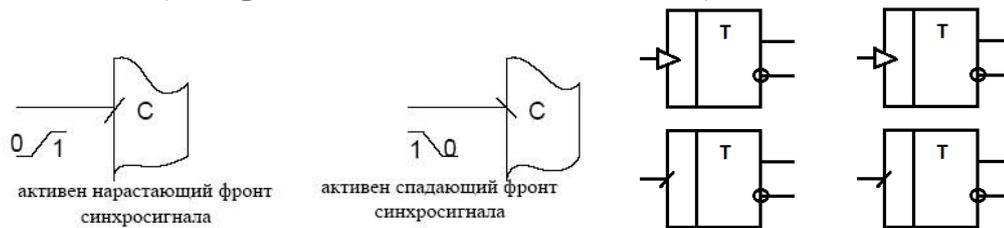


Рис.2. Динамическое управление

По принципу построения триггеры со статическим управлением можно разделить на одноступенчатые и двухступенчатые. Одноступенчатые триггеры характеризуются наличием одной ступени запоминания информации. В двухступенчатых триггерах имеется две ступени запоминания информации. Вначале информация записывается в первую ступень, а затем переписывается во вторую и появляется на выходе.

По функциональным возможностям различаются:

- а) триггер с отдельной установкой состояний 0 и 1 (RS-триггер);
- б) триггер с приемом информации по одному входу D (D-триггер или триггер задержки);
- в) триггер со счетным входом Т (Т-триггер);
- г) универсальный триггер с информационными входами J и К (JK-триггер).

JK-триггер имеет два информационных входа J и К, предназначенные для установки его выхода в логическое состояние 1 или 0. В интегральной схемотехнике JK-триггеры обычно выполняются синхронными, поэтому сигналы на информационных входах влияют на состояние JK-триггера только при поступлении тактового сигнала на его вход синхронизации С. Условное графическое изображение простейшего JK-триггера приведено на рис.3.

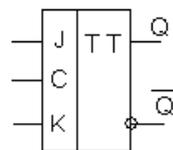


Рис.3. Условное графическое обозначение JK-триггера

Работа JK-триггеров описывается таблицей истинности (табл. 1) и таблицей переходов (табл. 2)

табл.1

<b>J</b>	<b>K</b>	<b>Q<sub>t+1</sub></b>	<b>Режим работы</b>
0	0	Q <sub>t</sub>	хранение
0	1	0	запись нуля
1	0	1	запись единицы
1	1	Q <sub>t</sub> '	переключение (счетный режим)

В JK-триггере комбинация  $J=K=1$  приводит к переходу выхода Q триггера в противоположное состояние.

табл.2

J	K	Q	Q <sub>t+1</sub>
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Чтобы расширить функциональные возможности JK-триггера, его снабжают асинхронными входами R и S, которые имеют приоритет по отношению к другим входам. На рис. 4 представлен JK-триггер K555ТВ9. При подаче логического нуля на вход S триггер асинхронно устанавливается в единичное состояние, а при подаче логического нуля на вход R – в нулевое состояние.

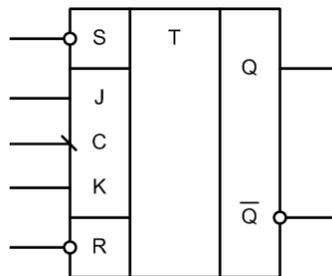


Рис.4. Условное графическое обозначение JK-триггера с асинхронными входами

#### 4. Перечень используемого оборудования

- лабораторная станция NI ELVIS II
- лабораторный модуль dLab8 для исследования работы JK- триггера

**Задание:** Проанализируйте работу триггера в статическом и динамическом режимах, для чего получите таблицу переходов JK-триггера, подавая на входы соответствующие сигналы, определите, какие комбинации входных сигналов «J» и «K» соответствуют режимам работы JK-триггера; исследуйте влияние установочных входов R и S на работу триггера, определите активный уровень сигналов асинхронного управления триггером, изучите работу JK-триггера в рабочих режимах, по временной диаграмме определите, по какому перепаду тактового импульса на входе «C» происходят переключения JK-триггера.

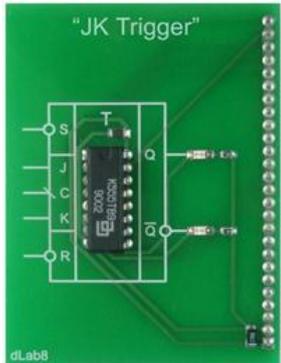
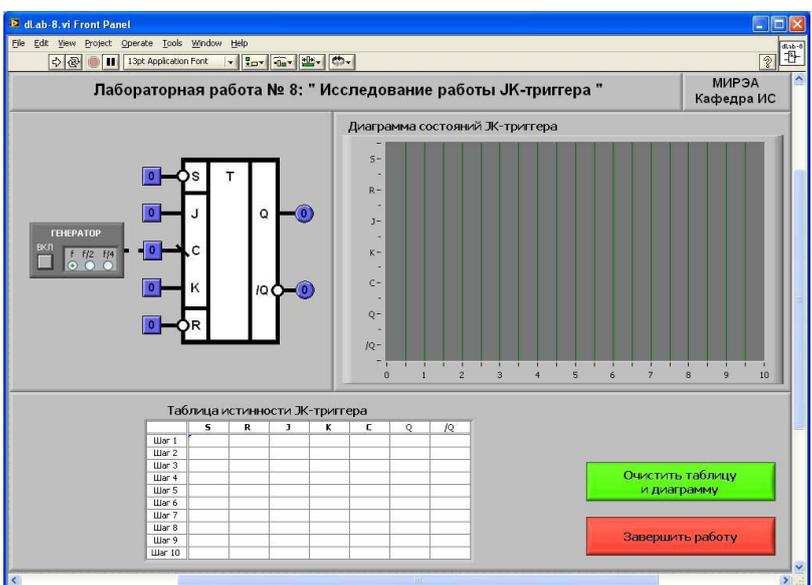
#### 5.Порядок выполнения.

### 5.1. Исследование JK-триггера в статическом режиме

Статический режим исследования JK-триггера реализуется при подаче на его тактовый вход «С» одиночных импульсов в ручном режиме. Для этого генератор импульсов («ГЕНЕРАТОР»), расположенный на лицевой панели ВП, должен быть выключен (кнопка «ВКЛ» отжата). Подача одиночного импульса прямоугольной формы на вход «С» триггера производится однократным нажатием на кнопку квадратной формы, расположенную около этого входа.

5.1.1. Установите лабораторный модуль **dLab8** на макетную плату лабораторной станции NI ELVIS.

5.1.2. Загрузите файл **dLab-8.vi** и запустите программу, щелкнув левой кнопкой мыши на экранной кнопке RUN.

	S	R	J	K	С	Q	Q̄
Шаг 1							
Шаг 2							
Шаг 3							
Шаг 4							
Шаг 5							
Шаг 6							
Шаг 7							
Шаг 8							
Шаг 9							
Шаг 10							

5.1.3. Выключите генератор импульсов, если он был включен.

5.1.4. Нажмите на кнопку **«Очистить таблицу и диаграмму»**.

5.1.5. Входы асинхронной установки «S» и асинхронного сброса «R» установите в состояние «1». Логический уровень изменяется при однократном нажатии с помощью манипулятора мышь на кнопку квадратной формы, расположенную около соответствующего входа. При этом на кнопке отображается состояние входа («0» - синий цвет или «1» - оранжевый цвет).

5.1.6. Установите на входах «J» и «K» триггера значения сигналов, приведенные в первой строке табл.3.

табл.3.

<i>Вход J</i>	<i>Вход K</i>
0	0
0	1
1	0
0	0
1	0
0	1
0	0
1	1
1	1
1	1

Таблица истинности

	S	R	J	K	C	Q	/Q
Шаг1							
Шаг2							
Шаг3							
Шаг4							
Шаг5							
Шаг6							
Шаг7							
Шаг8							
Шаг9							
Шаг10							

5.1.7. Нажмите и отпустите кнопку, расположенную около входа «С». На индикаторах круглой формы, расположенных около выходов триггера, будет отображено состояние его выходных сигналов. Логические состояния входов и выходов триггера будут автоматически занесены в таблицу истинности и на диаграмму состояний. В графу «С» таблицы истинности заносится символ  $\sqcap$ , означающий подачу импульса на вход «С». Аналогично занесите информацию остальных строк.

5.1.8. По результатам исследований заполните таблицу переходов JK-триггера (табл.4)

табл. 4

<b>J</b>	<b>K</b>	<b>Q<sub>t</sub></b>	<b>Q<sub>t+1</sub></b>
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

$Q_t$  – состояние триггера до подачи управляющих сигналов;

$Q_{t+1}$  – состояние триггера после подачи управляющих сигналов;

5.1.9. По таблице истинности и таблице переходов определите, какие комбинации входных сигналов «J» и «K» соответствуют режимам работы JK-триггера, и заполните табл.5.

табл. 5

<i>Режим работы</i>	<i>Вход J</i>	<i>Вход K</i>
Хранение информации		
Установка «1»		
Установка «0»		
Переключение		

## 5.2. Исследование JK-триггера в динамическом режиме

Динамический режим исследования JK-триггера реализуется при подаче на его тактовый вход «С» последовательности импульсов. Для этого генератор

импульсов («ГЕНЕРАТОР»), расположенный на лицевой панели ВП, должен быть включен (кнопка «ВКЛ» нажата). На выходе генератора формируется последовательность прямоугольных импульсов и подается на вход «С» триггера. Для удобства исследования включайте на время действия небольшого количества импульсов.

5.2.1. Нажмите на кнопку «Очистить таблицу и диаграмму».

5.2.2. Включите генератор импульсов на время действия одного-двух импульсов. На графический индикатор выводится временная диаграмма входных и выходных сигналов JK-триггера. В этом режиме таблица истинности не заполняется, а кнопка «Очистить таблицу и диаграмму» заблокирована от нажатия и имеет затененное изображение.

5.2.3. Изменяя логические состояния входов «S» и «R», изучите работу JK-триггера в режиме «Переключение» при подаче сигналов асинхронной установки «S» и асинхронного сброса «R», используя генератор импульсов. Определите активный уровень сигналов асинхронного управления триггером. Используйте таблицу 6. Полученные временные диаграммы скопируйте в отчет.

табл.6

S	R	J	K	Q	Q\
0	1	1	1		
1	0	1	1		
1	1	1	1		

5.2.4. Определите, какие из входов «J», «K» и «С» влияют на работу триггера (и влияют ли), если на вход «S» или «R» подан активный уровень сигнала асинхронного управления, используя генератор импульсов. Заполните таблицу 7.

табл.7

S	R	J	K	Q	Q\
0	1	0	0		
0	1	0	1		
0	1	1	0		
0	0	1	1		
1	0	0	0		
1	0	0	1		
1	0	1	0		
1	0	1	1		

Полученные временные диаграммы скопируйте в отчет. Сделайте вывод о работе JK-триггера в режиме асинхронного управления.

5.2.5. Установите входы «S» и «R» в состояние «1». Изменяя логические состояния входов «J» и «K», изучите работу JK-триггера в рабочих режимах, используя генератор импульсов. Сопоставьте наблюдаемые временные диаграммы с таблицей истинности и таблицей переходов JK-триггера.

S	R	J	K	Q	Q\
1	1	1	0		
1	1	0	1		
1	1	0	0		
1	1	1	1		

5.2.6. По временной диаграмме определите, по какому перепаду тактового импульса на входе «С» происходят переключения JK-триггера.

5.2.7. Запишите выводы по проведенным исследованиям.

5.2.8. Ответьте на контрольные вопросы

## 6. Указания к выполнению отчета

Отчет должен содержать:

- наименование и цель работы.
- перечень используемого оборудования
- таблицы истинности, переходов и временные диаграммы работы триггера в исследуемых режимах
- выводы по проведенным исследованиям
- ответы на контрольные вопросы.

## 7. Контрольные вопросы

1. Назовите условия задания режимов работы JK-триггера.
2. В чём разница между синхронным и асинхронным триггером?
3. Может ли JK-триггер оказаться в запрещённом состоянии, какому режиму соответствует комбинация  $J=K=1$  ?
4. Почему JK-триггер называется универсальным?

## 9. Список литературы

1. Келим Ю. М. Вычислительная техника/ Учеб. пособие для студ. СПО. – М.: Издательский центр «Академия», 2019г.
2. Основы электроники: учебник для СПО/ О.В., Миловзоров, И.Г. Панков, — 6-е изд. перераб. и доп. — Москва: Издательство Юрайт, 2021
3. Руководство пользователя NI ELVIS II. Комплект виртуальных измерительных приборов для учебных лабораторий. Издательство - NATIONAL INSTRUMENTS, 2008
4. Ю.В. Новиков, Скоробогатов Основы микропроцессорной техники. – 3-е изд., испр. – М.: Интернет-Университет информационных технологий; БИНОМ. Лаборатория знаний, 2006г.

## ИССЛЕДОВАНИЕ РАБОТЫ РЕГИСТРА СДВИГА

**1.Цель работы:** Изучение принципов работы и режимов функционирования регистра сдвига на примере микросхемы К555ИР11

**2.Время выполнения работы – 4час**

**3.Краткие теоретические сведения**

Регистр – функциональный узел последовательного типа, осуществляющий прием, хранение и выдачу чисел с двоичным представлением цифр разрядов. По способу записи информации регистры подразделяются на параллельные (регистры памяти), последовательные (регистры сдвига) и параллельно-последовательные.

В параллельном регистре запись и выдача слова осуществляется в параллельной форме одновременно во всех разрядах регистра. Его называют регистром хранения или регистром памяти. Эти регистры запоминают входные сигналы только в момент времени, определяемый сигналом синхронизации. В параллельных регистрах для хранения каждого разряда информации предназначен отдельный триггер.

Последовательный регистр характеризуется последовательной записью и выдачей слова, начиная со старшего или младшего разряда. В сдвиговых регистрах при записи многоразрядного числа используется цепочка триггеров, информация о каждом разряде последовательно переходит от одного триггера к другому. В зависимости от направления сдвига различают регистры:

- со сдвигом вправо (в сторону младших разрядов),
- со сдвигом влево (в сторону старших разрядов),
- реверсивные (сдвигающие и вправо и влево).

Параллельно-последовательные регистры входы и выходы как для параллельной, так и для последовательной формы приема и передачи слова. Запись может осуществляться в параллельном коде, а считывание в последовательном или наоборот. Такие регистры называют универсальными. Примером универсального регистра служит интегральная микросхема К555ИР11, условное графическое обозначение которой показано на рис. 1

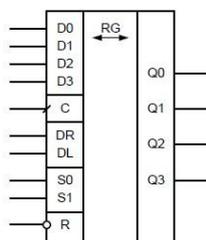


Рис.1 Условное графическое обозначение микросхемы универсального регистра К555ИР11

Регистр К555ИР11 может работать в следующих режимах (табл. 1): сброс, хранение данных, сдвиг влево, сдвиг вправо, и параллельная загрузка. Микросхема имеет входы: тактовый (С), параллельной загрузки (D0 – D3), выбора режима работы (S0 и S1), асинхронного сброса (R). Данные также могут поступать в регистр в последовательном коде на входы DL (при сдвиге влево) и

DR (при сдвиге вправо). Все операции кроме сброса выполняются в регистре синхронно по фронту тактовых импульсов. Внутренний код регистра может быть прочитан на выходах Q0 – Q3.

Таблица 1

Режим работы	Вход							Выход			
	R	C	S1	S0	DR	DL	Dn	Q0	Q1	Q2	Q3
Сброс	0	×	×	×	×	×	×	0	0	0	0
Хранение	1	×	0	0	×	×	×	q <sub>0</sub>	q <sub>1</sub>	q <sub>2</sub>	q <sub>3</sub>
Сдвиг влево	1	↑	1	0	×	0	×	q <sub>1</sub>	q <sub>2</sub>	q <sub>3</sub>	0
	1	↑	1	0	×	1	×	q <sub>1</sub>	q <sub>2</sub>	q <sub>3</sub>	1
Сдвиг вправо	1	↑	0	1	0	×	×	0	q <sub>0</sub>	q <sub>1</sub>	q <sub>2</sub>
	1	↑	0	1	1	×	×	1	q <sub>0</sub>	q <sub>1</sub>	q <sub>2</sub>
Параллельная загрузка	1	↑	1	1	×	×	d <sub>n</sub>	d <sub>0</sub>	d <sub>1</sub>	d <sub>2</sub>	d <sub>3</sub>

**Примечания:** - символ x обозначает безразличное состояние входа;  
- символ ↑ обозначает фронт тактового сигнала.

Синхронная параллельная загрузка осуществляется путем установки напряжения высокого уровня на управляющие входы S0, S1. На объединенные входы-выходы, находящиеся в состоянии высокого импеданса подается четырехразрядное слово, которое по положительному фронту на C записывается в триггеры.

Сдвиг вправо осуществляется синхронно с приходом положительного фронта тактового импульса на C при установке на S0 высокого, а на S1 низкого уровней. В этом режиме данные последовательно считываются с входа DR.

Сдвиг влево осуществляется синхронно с приходом положительного фронта тактового импульса на C при установке на S1 высокого, а на S0 низкого уровней. В этом режиме данные последовательно считываются с входа DL.

Режим хранения реализуется при подаче на оба управляющих входа S0, S1 напряжения низкого уровня.

Сброс выходов в состояние низкого уровня происходит асинхронно при подаче нуля на R.

Области применения сдвиговых регистров весьма разнообразны. В двоичной арифметике сдвиг числа на один разряд влево соответствует умножению его на 2, а сдвиг на один разряд вправо – делению пополам. В аппаратуре передачи данных универсальные регистры преобразуют параллельный код в последовательный и обратно. Передача данных последовательным кодом по сравнению с параллельной передачей существенно экономит число линий связи, однако при этом увеличивается время обмена.

#### 4. Перечень используемого оборудования

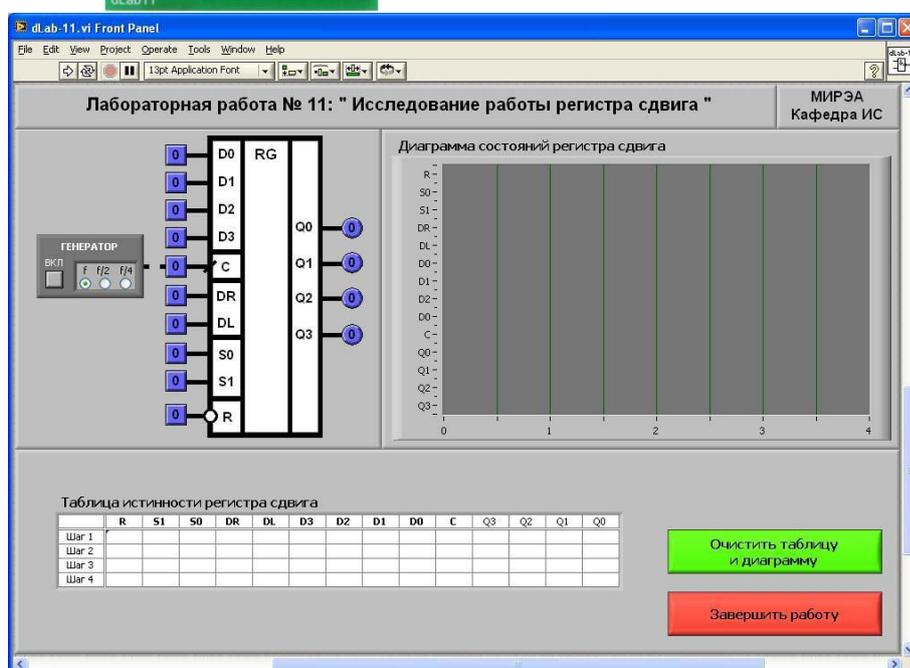
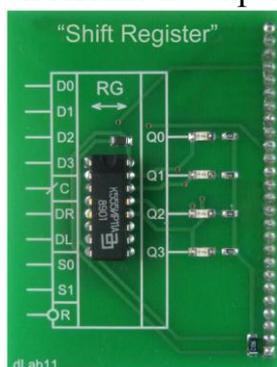
- лабораторная станция NI ELVIS II
- лабораторный модуль dLab11 для исследования работы регистра сдвига

**Задание:** По таблице истинности и диаграмме состояний проанализируйте работу регистра сдвига, определите условия задания режимов

### 5. Порядок выполнения.

5.1. Установите лабораторный модуль **dLab11** на макетную плату лабораторной станции NI ELVIS.

5.2. Загрузите файл **dLab-11.vi** и запустите программу, щелкнув левой кнопкой мыши на экранной кнопке RUN.



#### *Режим сдвига вправо*

5.3. Выключите генератор импульсов, если он был включен.

5.4. Нажмите на кнопку «Очистить таблицу и диаграмму».

5.5. Установите на входах выбора режима сигналы:  $S0=1$ ,  $S1=0$ ,  $R=1$ . Логический уровень изменяется при однократном нажатии с помощью манипулятора мышь на кнопку квадратной формы, расположенную около соответствующего входа.

5.6. Выполните сброс регистра. Для этого на лицевой панели ВП с помощью манипулятора мышь сначала переведите в отжатое, а затем в нажатое состояние кнопку квадратной формы, расположенную около входа «R». На

индикаторах выходных сигналов регистра «Q0», «Q1», «Q2» и «Q3» должны установиться нулевые значения.

5.7. Установите на входе последовательных данных «DR» логический сигнал «1».

5.8. Нажмите и отпустите кнопку, расположенную около входа «С». На индикаторах круглой формы, расположенных около выходов «Q0», «Q1», «Q2» и «Q3» регистра, будет отображено состояние его выходных сигналов. Логические состояния входов и выходов триггера будут автоматически занесены в таблицу истинности и на диаграмму состояний. В графу «С» таблицы истинности заносится символ  $\lceil$ , означающий подачу импульса на вход «С».

5.9. Установите на входе последовательных данных «DR» логический сигнал «0».

5.10. Трижды нажмите и отпустите кнопку, расположенную около входа «С».

5.11. Скопируйте полученные таблицу истинности и диаграмму состояний в отчет.

5.12. По таблице истинности и диаграмме состояний определите, в каком направлении (от Q0 к Q3 или от Q3 к Q0) смещается логическая единица, записанная в регистр на первом такте. Вывод запишите в отчет.

*Режим сдвига влево*

5.13. Нажмите на кнопку «Очистить таблицу и диаграмму».

5.14. Установите на входах выбора режима сигналы: S0=0, S1=1, R=1.

5.15. Выполните сброс регистра.

5.16. Установите на входе последовательных данных «DL» логический сигнал «1».

5.17. Нажмите и отпустите кнопку, расположенную около входа «С». На индикаторах круглой формы, расположенных около выходов «Q0», «Q1», «Q2» и «Q3» регистра, будет отображено состояние его выходных сигналов. Логические состояния входов и выходов триггера будут автоматически занесены в таблицу истинности и на диаграмму состояний. В графу «С» таблицы истинности заносится символ  $\lceil$ , означающий подачу импульса на вход «С».

5.18. Установите на входе последовательных данных «DL» логический сигнал «0».

5.19. Трижды нажмите и отпустите кнопку, расположенную около входа «С».

5.20. Скопируйте полученные таблицу истинности и диаграмму состояний в отчет.

5.21. По таблице истинности и диаграмме состояний определите, в каком направлении (от Q0 к Q3 или от Q3 к Q0) смещается логическая единица, записанная в регистр на первом такте. Вывод запишите в отчет.

*Режим параллельной загрузки*

5.22. Нажмите на кнопку «Очистить таблицу и диаграмму».

5.23. Установите на входах выбора режима сигналы:  $S0=1$ ,  $S1=1$ ,  $R=1$ .

5.24. Выполните сброс регистра.

5.25. Установите на входах параллельной загрузки «D0», «D1», «D2» и «D3» значения сигналов, приведенные в первой строке табл. 2

табл. 2

Вход D3	Вход D2	Вход D1	Вход D0
0	1	0	1
1	1	1	1
0	0	1	0
1	1	0	0

4.26. Нажмите и отпустите кнопку, расположенную около входа «С». На индикаторах круглой формы, расположенных около выходов «Q0», «Q1», «Q2» и «Q3» регистра, будет отображено состояние его выходных сигналов. Логические состояния входов и выходов триггера будут автоматически занесены в таблицу истинности и на диаграмму состояний. В графу «С» таблицы истинности заносится символ  $\square$ , означающий подачу импульса на вход «С».

5.27. Аналогично занесите информацию остальных строк

5.28. Скопируйте таблицу истинности в отчет.

5.29. По таблице истинности и диаграмме состояний проверьте соответствие выходных сигналов регистра Q0, Q1, Q2 и Q3 сигналам на входах параллельной загрузки D0, D1, D2 и D3. Вывод запишите в отчет.

5.30. Запишите выводы по проведенным исследованиям

5.31. Ответьте на контрольные вопросы

## 6. Указания к выполнению отчета

Отчет должен содержать:

- наименование и цель работы.
- перечень используемого оборудования
- таблицы истинности и диаграммы состояний работы регистров в режимах сдвига вправо и влево
  - таблицу истинности и диаграмму состояний регистра в режиме параллельной загрузки
  - выводы по проведенным исследованиям
  - ответы на контрольные вопросы.

## 7. Контрольные вопросы

1. Какие функции выполняют регистры?
2. Перечислите режимы работы микросхемы универсального регистра K555IP11 и соответствующие им комбинации управляющих входов S1 S0.
3. В чем состоит отличие процедуры занесения информации в режиме параллельной загрузки и режимах сдвига?

4. Какой регистр называется реверсивным?

### **8.Список литературы**

1. Келим Ю. М. Вычислительная техника/ Учеб. пособие для студ.СПО.– М.: Издательский центр «Академия», 2019г.

2. Основы электроники: учебник для СПО/ О.В., Миловзоров, И.Г.Панков, — 6-е изд.перераб. и доп. —Москва: Издательство Юрайт, 2021

3. Руководство пользователя NI ELVIS II. Комплект виртуальных измерительных приборов для учебных лабораторий. Издательство - NATIONAL INSTRUMENTS, 2018

4. Ю.В. Новиков, Скоробогатов Основы микропроцессорной техники. – 3-е изд., испр. – М.: Интернет-Университет информационных технологий; БИНОМ. Лаборатория знаний, 2020г.

## ИССЛЕДОВАНИЕ РАБОТЫ ИССЛЕДОВАНИЕ ОПЕРАТИВНОГО ЗУ

**1.Цель работы:** Получить практическое представление о функционировании оперативного запоминающего устройства

**2.Время выполнения-4час**

**3.Краткие теоретические сведения**

Оперативным запоминающим устройством (ОЗУ) называется схема, служащая для записи, хранения и считывания массива многоразрядных двоичных слов (кодов), размещаемых каждое в своей запоминающей ячейке в соответствии с уникальным адресом для каждого запоминаемого слова. По мере поступления входных запоминаемых слов, каждому из них в соответствие ставится свой адрес. При считывании в ответ на установку адреса каждое слово появляется на выходе запоминающего устройства. ОЗУ относятся к энергозависимым устройствам, т.е. при отключении питания информация разрушается.

К основным характеристикам ОЗУ относятся:

- емкость памяти - наибольший объем информации, который одновременно может храниться в ОЗУ. Базовой единицей измерения емкости памяти служит бит, представляющий собой один разряд двоичного числа;

- организация запоминающего устройства, для оценки которого служит произведение числа хранимых слов на их разрядность. Это произведение равно емкости памяти ОЗУ. Например, два ОЗУ с организацией 32x32 и 64x16 имеют одинаковую емкость памяти 1024 бит;

- быстродействие (производительность) ОЗУ может характеризоваться временем цикла записи, временем цикла считывания информации, а также временем обращения памяти, включающим в себя оба цикла;

К основным управляющим и информационным сигналам относятся:

- $A_0...A_n$  - адрес, или адресный код. Адресный код является номером элемента или ячейки памяти, в котором хранится бит, байт или слово информации. Число разрядов адресного кода «n» определяет емкость памяти ОЗУ. Например, 12-разрядный код  $A_{11}A_{10}A_9...A_0$  позволяет обратиться к любой из  $2^{12} = 4096$  ячеек памяти ОЗУ;

- (CS) (Chip Select) или (CE) (Chip Enable) – сигналы выбора кристалла, или (CE) (Chip Enable) – сигналы выбора кристалла, или микросхемы, активизирующие работу микросхемы ОЗУ при нулевом логическом уровне;
- R/W (Read/Write) – сигнал, управляющий режимом работы памяти: при  $R/W = 0$  производится запись, при  $R/W = 1$  - чтение;

- DI, DO (Data Input, Data Output) – входные и выходные  $m$ -разрядные данные ОЗУ, передаваемые по совмещенной или отдельным шинам. Число разрядов  $m$  определяется организацией ЗУ.

По способу хранения информации ОЗУ делятся на две группы:

- статические ОЗУ, в которых элементами памяти являются статические триггеры;
- динамические ОЗУ, в которых состояние элементов памяти (обычно полупроводниковых емкостей) требует периодического проведения процесса регенерации (восстановления) исходных уровней сигналов.

Условное графическое обозначение микросхемы статического ОЗУ K531PY8 с организацией 16x4 представлено на рис. 1.

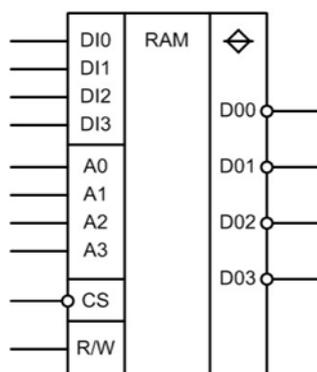


Рис.1 УГО микросхемы K531PY8

Микросхема K531PY8 имеет адресные входы A0 – A3, информационные входы DI0 – DI3, вход выбора режима R/W (чтение/запись), вход выбора микросхемы CS и информационные выходы DO0 – DO3.

Для записи в микросхему четырехразрядного двоичного кода, поданного на информационные входы DI0 – DI3, необходимо установить адрес A0 – A3 ячейки памяти, в которую производится запись, а затем выбрать режим записи R/W=0 и подать сигнал выбора микросхемы CS=0. В данном режиме записываемая информация на выходы DO0 – DO3 микро. В данном режиме записываемая информация на выходы DO0 – DO3 микросхемы не передается.

Для чтения данных из ОЗУ необходимо установить адрес A0 – A3 требуемой ячейки памяти, выбрать режим чтения R/W=1 и подать сигнал выбора микросхемы CS=0. Информация из ячейки памяти передается на информационные выходы DO0 – DO3 через инвертирующие усилители с тремя состояниями выхода. Управление выходом совмещено с сигналом CS, то есть при CS=1 выходные каскады имеют высокоимпедансное состояние (Z-состояние), а при CS=0 в режиме чтения R/W=1 – данные могут быть прочитаны на выходах DO0 – DO3.

Таким образом, при CS=1 независимо от состояния входа R/W микросхема переходит в режим хранения информации с переключением выхода в высокоимпедансное состояние. Так как ОЗУ является асинхронным, то при выполнении записи или чтения данных в течение всего времени действия сигнала CS=0 адресный код на входах A0 – A3 должен быть неизменным.

#### 4.Перечень используемого оборудования:

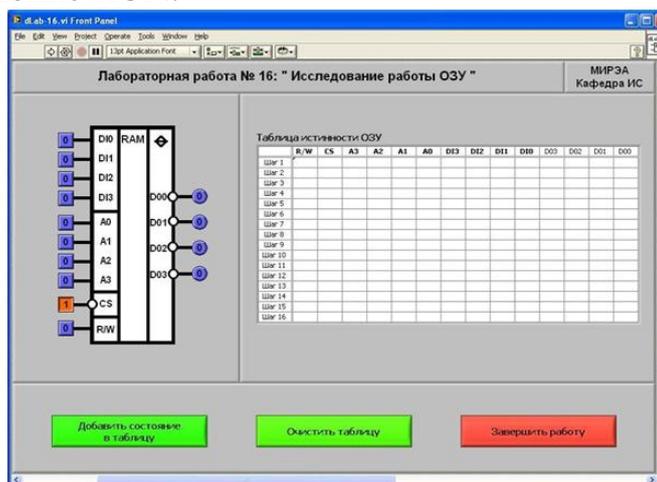
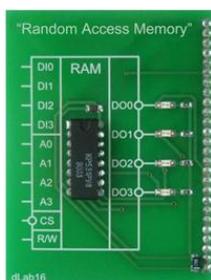
- лабораторная станция NI ELVIS II
- лабораторный модуль dLab16 для исследования работы ОЗУ

**Задание:** Проанализируйте работу микросхемы ОЗУ K531PY8 в режимах записи и считывания и сравните полученные результаты.

### 5. Порядок выполнения.

5.1. Установите лабораторный модуль **dLab16** на макетную плату лабораторной станции NI ELVIS.

5.2. Загрузите файл **dLab-16.vi** и запустите программу, щелкнув левой кнопкой мыши на экранной кнопке RUN.



### **РЕЖИМ ЗАПИСИ ДАННЫХ**

5.3. Нажмите на кнопку «**Очистить таблицу**».

5.4. Установите на входах адреса «A0» - «A3» и данных «D10» - «D13» значения сигналов, приведенные в первой строке табл.16.1. Логический уровень изменяется при однократном нажатии с помощью манипулятора мышь на кнопку квадратной формы, расположенную около соответствующего входа. При этом на кнопке отображается состояние входа («0» - синий цвет или «1» - оранжевый цвет).

5.5. Задайте режим записи в ОЗУ, установив на входе «R/W» логический сигнал «0».

5.6. Установите на входе «CS» сигнал «0».

5.7. Занесите логические состояния входов ОЗУ в таблицу истинности. Для этого на лицевой панели ВП нажмите на кнопку «Добавить состояние в таблицу». В режиме записи данных информация о состоянии выходов «D00» - «D03» в таблицу истинности не заносится (в соответствующие ячейки помещается символ «—»).

5.8. Установите на входе «CS» сигнал «1».

5.9. Повторите пп.4.1.2 – 4.1.6 для остальных строк табл.16.1.

5.10. Скопируйте полученную таблицу истинности в отчет.

<i>Вход A3</i>	<i>Вход A2</i>	<i>Вход A1</i>	<i>Вход A0</i>	<i>Вход DI3</i>	<i>Вход DI2</i>	<i>Вход DI1</i>	<i>Вход DI0</i>
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	0
0	0	1	1	0	0	1	1
0	1	0	0	0	1	0	0
0	1	0	1	0	1	0	1
0	1	1	0	0	1	1	0
0	1	1	1	0	1	1	1
1	0	0	0	1	0	0	0
1	0	0	1	1	0	0	1
1	0	1	0	1	0	1	0
1	0	1	1	1	0	1	1
1	1	0	0	1	1	0	0
1	1	0	1	1	1	0	1
1	1	1	0	1	1	1	0
1	1	1	1	1	1	1	1

### ***РЕЖИМ ЧТЕНИЯ ДАННЫХ***

5.11. Нажмите на кнопку «Очистить таблицу».

5.12. Установите на входах адреса «A0» - «A3» значения сигналов, приведенные в первой строке табл.16.2.

5.13. Задайте режим чтения из ОЗУ, установив на входе «R/W» логический сигнал «1».

5.14. Установите на входе «CS» сигнал «0». Информация из выбранной ячейки памяти поступит на выход микросхемы и будет отображаться на индикаторах круглой формы, расположенных около выходов «DO0», «DO1», «DO2» и «DO3».

5.15. Занесите логические состояния входов ОЗУ в таблицу истинности. Для этого на лицевой панели ВП нажмите на кнопку «Добавить состояние в таблицу». В режиме чтения данных информация о состоянии входов «DI0» - «DI3» в таблицу истинности не заносится (в соответствующие ячейки помещается символ «—»).

<i>Вход A3</i>	<i>Вход A2</i>	<i>Вход A1</i>	<i>Вход A0</i>
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1

5.16. Повторите пп.4.1.10 – 4.1.13 для остальных строк табл.16.2.

5.17. Скопируйте полученную таблицу истинности в отчет.

5.18. По таблицам истинности сравните, записанные в ОЗУ и прочитанные из него данные. Результаты сравнения занесите в отчет. При этом следует помнить, что данные выводятся из ячеек памяти на выход микросхемы в инвертированном виде.

## **6. Указания к выполнению отчета**

Отчет должен содержать:

1. Тему и цель работы, перечень используемого оборудования, задание и порядок выполнения
2. Таблицы истинности в режимах записи и считывания
3. Выводы по проведенным исследованиям
4. Ответы на контрольные вопросы

## **7. Контрольные вопросы**

1. Какое запоминающее устройство называется оперативным?
2. Перечислите основные параметры ОЗУ. Дайте им характеристику.
3. В чем различие статического и динамического ОЗУ?
4. Опишите принцип работы ОЗУ типа K531PY8.

## **8. Список литературы**

1. Келим Ю. М. Вычислительная техника/ Учеб. пособие для студ. СПО.– М.: Издательский центр «Академия», 2019г.
2. Основы электроники: учебник для СПО/ О.В., Миловзоров, И.Г.Панков, — 6-е изд. перераб. и доп. — Москва: Издательство Юрайт, 2021

3. Руководство пользователя NI ELVIS II. Комплект виртуальных измерительных приборов для учебных лабораторий. Издательство - NATIONAL INSTRUMENTS, 2018

4. Ю.В. Новиков, Скоробогатов Основы микропроцессорной техники. – 3-е изд., испр. – М.: Интернет-Университет информационных технологий; БИНОМ. Лаборатория знаний, 2020г.

5. Угрюмов Е.П. Цифровая схемотехника. - СПб.: БХВ - Петербург, 2020

-