

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Локтионова Оксана Геннадьевна  
Должность: проректор по учебной работе  
Дата подписания: 16.06.2025 12:27:40  
Уникальный программный ключ:  
0b817ca911e6668abb13a5d426d39e5f1c11eabhf73e945df4a4851fda56d089

**МИНОБРНАУКИ РОССИИ**

**Федеральное государственное бюджетное образовательное  
учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)**

**Кафедра вычислительной техники**

**УТВЕРЖДАЮ**  
Проректор по учебной работе  
**О.Г. Локтионова**  
« 19 » \_\_\_\_\_ 2021 г.



**АРХИТЕКТУРА ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ И  
КОМПЬЮТЕРНЫХ СЕТЕЙ**

Методические указания по выполнению лабораторных работ для студентов направления подготовки «Математическое обеспечение и администрирование информационных систем»



УДК 004

Составители: И.В. Зотов, Д.О. Бобынцев

Рецензент

Кандидат технических наук Т.Н. Конаныхина

**Архитектура вычислительных систем и компьютерных сетей: методические указания по выполнению лабораторных работ / Юго-Зап. гос. ун-т; сост. И.В. Зотов, Д.О. Бобынцев. Курск, 2021. 34 с. Библиогр.: с. 33.**

Содержат сведения по основам вычислительных систем, построения сетей и средствам телекоммуникаций. Указывается порядок выполнения работ, содержание отчета и правила оформления работ, контрольные вопросы.

Предназначено для студентов направления подготовки «Математическое обеспечение и администрирование информационных систем».

Текст печатается в авторской редакции.

Подписано в печать *19.04.21*. Формат 60x84 1/16.  
Усл.печ. л. 1,98. Уч.-изд. л. 1,79. Тираж 100 экз. Заказ. *446* Бесплатно.

Юго-Западный государственный университет.

305040, г. Курск, ул. 50 лет Октября, 94.

## Лабораторная работа №1 Системы счисления

### *Теоретический материал*

Под **системой счисления** понимается способ представления любого числа с помощью некоторого алфавита символов, называемых цифрами.

Все системы счисления делятся на **позиционные** и **непозиционные**.

**Непозиционными** системами являются такие системы счисления, в которых каждый символ сохраняет свое значение независимо от места его положения в числе.

Примером непозиционной системы счисления является римская система. К недостаткам таких систем относятся наличие большого количества знаков и сложность выполнения арифметических операций.

Система счисления называется **позиционной**, если одна и та же цифра имеет различное значение, определяющееся позицией цифры в последовательности цифр, изображающей число. Это значение меняется в однозначной зависимости от позиции, занимаемой цифрой, по некоторому закону.

Примером позиционной системы счисления является десятичная система, используемая в повседневной жизни.

Количество  $p$  различных цифр, употребляемых в позиционной системе определяет название системы счисления и называется **основанием** системы счисления - " $p$ ".

В десятичной системе используются десять цифр: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9; эта система имеет основанием число десять.

Любое число  $N$  в позиционной системе счисления с основанием  $p$  может быть представлено в виде полинома от основания  $p$ :

$$N = a_n p^n + a_{n-1} p^{n-1} + \dots + a_1 p + a_0 + a_{-1} p^{-1} + a_{-2} p^{-2} + \dots$$

здесь  $N$  - число,  $a_j$  - коэффициенты (цифры числа),  $p$  - основание системы счисления ( $p > 1$ ).

Принято представлять числа в виде последовательности цифр:

$$N = a_n a_{n-1} \dots a_1 a_0 . a_{-1} a_{-2} \dots$$

В этой последовательности точка отделяет целую часть числа

от дробной (коэффициенты при положительных степенях, включая нуль, от коэффициентов при отрицательных степенях). Точка опускается, если нет отрицательных степеней (число целое).

Таблица 1 – Наиболее важные системы счисления

Двоичная (Основание 2)	Восьмеричная (Основание 8)		Десятичная (Основание 10)	Шестнадцатеричная (Основание 16)	
		триады			тетрады
0	0	000	0	0	0000
1	1	001	1	1	0001
	2	010	2	2	0010
	3	011	3	3	0011
	4	100	4	4	0100
	5	101	5	5	0101
	6	110	6	6	0110
	7	111	7	7	0111
			8	8	1000
			9	9	1001
				A	1010
				B	1011
				C	1100
				D	1101
				E	1110
				F	1111

В ЭВМ применяют позиционные системы счисления с недесятичным основанием: двоичную, восьмеричную, шестнадцатеричную.

В аппаратной основе ЭВМ лежат двухпозиционные элементы, которые могут находиться только в двух состояниях; одно из них обозначается 0, а другое - 1. Поэтому основной системой счисления применяемой в ЭВМ является двоичная система.

**Двоичная система счисления.** Используется две цифры: 0 и 1. В двоичной системе любое число может быть представлено в виде:

$$N = b_n b_{n-1} \dots b_1 b_0 . b_{-1} b_{-2} \dots$$

где  $b_j$  либо 0, либо 1.

**Восьмеричная система счисления.** Используется восемь цифр: 0, 1, 2, 3, 4, 5, 6, 7. Употребляется в ЭВМ как вспомогательная для записи информации в сокращенном виде. Для представления одной цифры восьмеричной системы используется три двоичных разряда (триада) (Таблица 1).

**Шестнадцатеричная система счисления.** Для изображения чисел употребляются 16 цифр. Первые десять цифр этой системы обозначаются цифрами от 0 до 9, а старшие шесть цифр - латинскими буквами: 10-A, 11-B, 12-C, 13-D, 14-E, 15-F. Шестнадцатеричная система используется для записи информации в сокращенном виде. Для представления одной цифры шестнадцатеричной системы счисления используется четыре двоичных разряда (тетрада) (Таблица 1).

**Рассмотрим перевод чисел из одной системы счисления в другую.**

**Перевод чисел в десятичную систему** осуществляется путем составления степенного ряда с основанием той системы, из которой число переводится. Затем подсчитывается значение суммы.

Пример.

а) Перевести  $10101101.101_2 \rightarrow "10"$  с.с.

$$10101101.101_2 = 1 \cdot 2^7 + 0 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 + 1 \cdot 2^{-1} + 0 \cdot 2^{-2} + 1 \cdot 2^{-3}$$

б) Перевести  $703.04_8 \rightarrow "10"$  с.с.

$$703.04_8 = 7 \cdot 8^2 + 0 \cdot 8^1 + 3 \cdot 8^0 + 0 \cdot 8^{-1} + 4 \cdot 8^{-2} =$$

$451.0625_{10}$  в) Перевести  $B2E.4_{16} \rightarrow "10"$  с.с.

$$B2E.4_{16} = 11 \cdot 16^2 + 2 \cdot 16^1 + 14 \cdot 16^0 + 4 \cdot 16^{-1} = 2862.25_{10}$$

**Перевод целых десятичных чисел в недесятичную систему счисления** осуществляется последовательным делением десятичного числа на основание той системы, в которую оно

переводится, до тех пор, пока не получится частное меньше этого основания. Число в новой системе записывается в виде остатков деления, начиная с последнего.

Пример.

а) Перевести  $181_{10} \rightarrow "8"$  с.с.

$$\begin{array}{r|l} 181 & 8 \\ -176 & 22 \\ \hline 5 & 16 \\ & 2 \\ \hline & 6 \end{array}$$

Результат:  $181_{10} = 265_8$

б) Перевести  $622_{10} \rightarrow "16"$  с.с.

$$\begin{array}{r|l} 622 & 16 \\ -48 & 38 \\ \hline 142 & 32 \\ -128 & 6 \\ \hline 14 & \end{array}$$

Результат:  $622_{10} = 26E_{16}$

**Перевод правильных дробей из десятичной системы счисления в недесятичную.**

Для перевода правильной десятичной дроби в другую систему эту дробь надо последовательно умножать на основание той системы, в которую она переводится. При этом умножаются только дробные части. Дробь в новой системе записывается в виде целых частей произведений, начиная с первого.

Пример.

Перевести  $0.3125_{10} \rightarrow "8"$  с.с.

$$\begin{array}{l} \downarrow \\ \begin{array}{r|l} 0 & 3125 \times 8 \\ \hline 2 & 5000 \times 8 \\ 4 & 0000 \end{array} \end{array}$$

Результат:  $0.3125_{10} = 0.24_8$

**Замечание.** Конечной десятичной дроби в другой системе счисления может соответствовать бесконечная (иногда периодическая) дробь. В этом случае количество знаков в представлении дроби в новой системе берется в зависимости от требуемой точности.

Пример.

Перевести  $0.65_{10} \rightarrow "2"$  с.с. Точность 6 знаков.

0	65 × 2
1	3 × 2
0	6 × 2
1	2 × 2
0	4 × 2
0	8 × 2
1	6 × 2
...	

$\downarrow$

Результат:  $0.65_{10} \approx 0.10(1001)_2$

**Для перевода неправильной десятичной дроби в систему счисления с недесятичным основанием необходимо отдельно перевести целую часть и отдельно дробную.**

Пример:

Перевести  $23.125_{10} \rightarrow$  "2" с.с.

1) Переведем  
целую часть:

23	2
22	11 2
1	10 5 2
	1 4 2 2
	1 2 1
	0

$\swarrow$

2) Переведем  
дробную часть:

0	125 × 2
0	25 × 2
0	5 × 2
1	0

$\downarrow$

Таким образом:  $23_{10} = 10111_2$ ;  $0.125_{10} = 0.001_2$ .

Результат:  $23.125_{10} = 10111.001_2$ .

Необходимо отметить, что целые числа остаются целыми, а правильные дроби - дробями в любой системе счисления.

**Для перевода восьмеричного или шестнадцатеричного числа в двоичную форму достаточно заменить каждую цифру этого числа соответствующим трехразрядным двоичным числом (триадой) (таблица 1) или четырехразрядным двоичным числом (тетрадой) (таблица 1), при этом отбрасывают ненужные нули в старших и младших разрядах.**

Пример.

а) Перевести  $305.4_8 \rightarrow$  "2" с.с.

$$\begin{array}{cccc} \underline{3} & \underline{0} & \underline{5} & \underline{.4} \\ \underline{011} & \underline{000} & \underline{101} & \underline{100} \end{array} \cdot 8 = 11000101.1_2$$

б) Перевести  $7B2.E_{16} \rightarrow "2" \text{ с.с.}$

$$\begin{array}{cccc} \underline{7} & \underline{B} & \underline{2} & \underline{.E} \\ \underline{0111} & \underline{1011} & \underline{0010} & \underline{1110} \end{array} \cdot 16 = 11110110010.111_2$$

**Для перехода от двоичной к восьмеричной (шестнадцатеричной) системе** поступают следующим образом: двигаясь от точки влево и вправо, разбивают двоичное число на группы по три (четыре) разряда, дополняя при необходимости нулями крайние левую и правую группы. Затем триаду (тетраду) заменяют соответствующей восьмеричной (шестнадцатеричной) цифрой.

Пример.

а) Перевести  $1101111001.1101_2 \rightarrow "8" \text{ с.с.}$

$$\begin{array}{cccccc} \underline{001} & \underline{101} & \underline{111} & \underline{001} & \underline{.110} & \underline{100} \\ \underline{1} & \underline{5} & \underline{7} & \underline{1} & \underline{6} & \underline{4} \end{array} = 1571.64_8$$

б) Перевести  $1111111011.100111_2 \rightarrow "16" \text{ с.с.}$

$$\begin{array}{cccccc} \underline{0111} & \underline{1111} & \underline{1011} & \underline{.1001} & \underline{1100} \\ \underline{7} & \underline{F} & \underline{B} & \underline{9} & \underline{C} \end{array} = 7FB.9C_{16}$$

**Перевод из восьмеричной в шестнадцатеричную систему и обратно** осуществляется через двоичную систему с помощью триад и тетрад.

Пример. Перевести  $175.24_8 \rightarrow "16" \text{ с.с.}$

$$\begin{array}{cccccc} \underline{1} & \underline{7} & \underline{5} & \underline{.2} & \underline{4} \\ \underline{001} & \underline{111} & \underline{101} & \underline{010} & \underline{100} \end{array} \cdot 8 = 1111101.0101_2 = \begin{array}{ccc} \underline{0111} & \underline{1101} & \underline{.0101} \\ \underline{7} & \underline{D} & \underline{5} \end{array} = 7D.5_{16}$$

Результат:  $175.24_8 = 7D.5_{16}$ .

## 1. Двоичная арифметика.

Правила выполнения арифметических действий над двоичными числами задаются таблицами двоичных сложения, вычитания и умножения.

При сложении двоичных чисел в каждом разряде производится сложение цифр слагаемых и переноса из соседнего младшего разряда, если он имеется. При этом необходимо учитывать, что  $1+1$  дают нуль в данном разряде и единицу переноса в следующий.



Таблица 2 – Правила выполнения арифметических действий над двоичными числами

Таблица двоичного сложения	Таблица двоичного вычитания	Таблица двоичного Умножения
0+0=0	0-0=0	0× 0=0
0+1=1	1-0=1	0× 1=0
1+0=1	1-1=0	1× 0=0
1+1=10	10-1=1	1× 1=1

Пример. Выполнить сложение двоичных чисел:  
а) X=1101, Y=101;

$$\begin{array}{r}
 \phantom{X=} \phantom{Y=} \phantom{X+Y=} \\
 \phantom{X=} \phantom{Y=} \phantom{X+Y=} 1 \phantom{0} \phantom{1} \phantom{0} \phantom{1} \leftarrow \text{единицы переноса} \\
 X= \phantom{+} 1101 \\
 Y= + \phantom{1} 101 \\
 \hline
 X+Y= 10010
 \end{array}$$

Результат 1101+101=10010.

б) X=1101, Y=101, Z=111;

$$\begin{array}{r}
 \phantom{X=} \phantom{Y=} \phantom{Z=} \phantom{X+Y+Z=} \\
 \phantom{X=} \phantom{Y=} \phantom{Z=} \phantom{X+Y+Z=} 1 \phantom{1} \phantom{1} \phantom{0} \phantom{0} \phantom{1} \leftarrow \text{единицы переноса} \\
 X= \phantom{+} 1101 \\
 Y= + \phantom{1} 101 \\
 Z= + \phantom{1} 111 \\
 \hline
 X+Y+Z= 11001
 \end{array}$$

Результат 1101+101+111=11001.

При вычитании двоичных чисел в данном разряде при необходимости занимается 1 из старшего разряда. Эта занимаемая 1 равна двум 1 данного разряда.

Пример. Заданы двоичные числа X=10010 и Y=101. Вычислить X-Y.

$$\begin{array}{r}
 10010 \\
 -101 \\
 \hline
 01101
 \end{array}$$

Результат 10010 - 101=1101.

Умножение двоичных чисел производится по тем же

правилам, что и для десятичных с помощью таблиц двоичного умножения и сложения.

Пример.  $1001 \times 101 = ?$

$$\begin{array}{r} 1001 \\ \times 101 \\ \hline 1001 \\ 1001 \\ \hline 101101 \end{array}$$

Результат  $1001 \times 101 = 101101$ .

Деление двоичных чисел производится по тем же правилам, что и для десятичных. При этом используются таблицы двоичного умножения и вычитания.

Пример.  $1100.011 : 10.01 = ?$

$$\begin{array}{r} 110001.1 \quad | \quad 1001 \\ \underline{1001} \quad | \quad 101.1 \\ 1101 \\ \underline{1001} \\ 1001 \\ \underline{1001} \\ 0 \end{array}$$

Результат  $1100.011 : 10.01 = 101.1$ .

### Варианты задания

1. Перевести следующие числа в десятичную систему счисления:

а)  $110111_2$ ; б)  $10110111.1011_2$ ; в)  $563.44_8$ ; г)  $721.35_8$ ; д)  $1C4.A_{16}$ ; е)  $9A2F.B5_2$ .

2. Перевести следующие числа из "10" с.с в "2", "8", "16" с.с.:

а) 463; б) 1209; в) 362; г) 3925; д) 11355.

3. Перевести следующие числа из "10" с.с в "2", "8", "16" с.с. (точность вычислений - 5 знаков после точки):

а) 0.0625; б) 0.345; в) 0.225; г) 0.725; д) 217.375; е) 31.2375; ж) 725.03125; з) 8846.04.

4. Перевести следующие числа в двоичную систему счисления:

а)  $1725.326_8$ ; б)  $341.34_8$ ; в)  $7BF.52A_{16}$ ; г)  $3D2.C_{16}$ .

5. Перевести следующие числа из одной системы счисления в

другую:

- а)  $11011001.01011_2 \rightarrow "8" \text{ с.с.};$
- б)  $1011110.1101_2 \rightarrow "8" \text{ с.с.};$
- в)  $1101111101.0101101_2 \rightarrow "16" \text{ с.с.};$
- г)  $110101000.100101_2 \rightarrow "16" \text{ с.с.}$

6. Перевести следующие числа из одной системы счисления в другую:

- а)  $312.7_8 \rightarrow "16" \text{ с.с.};$     б)  $51.43_8 \rightarrow "16" \text{ с.с.};$
- в)  $5B.F_{16} \rightarrow "8" \text{ с.с.};$     г)  $D4.19_{16} \rightarrow "8" \text{ с.с.}$

7. Заданы двоичные числа  $X$  и  $Y$ . Вычислить  $X+Y$  и  $X-Y$ , если:

- а)  $X=1101001; Y=101111;$
- б)  $X=101110110; Y=10111001;$
- в)  $X=100011001; Y=101011.$

8. Заданы двоичные числа  $X$  и  $Y$ . Вычислить  $X*Y$  и  $X/Y$ , если:

- а)  $X=1000010011; Y=1011;$
- б)  $X=110010101; Y=1001;$
- в)  $X=100101.011; Y=110.1;$
- г)  $X=100000.1101; Y=101.01.$

### Контрольные вопросы

1. Какая система счисления называется позиционной?
2. Какая система счисления называется непозиционной?
3. Что такое основание системы счисления?
4. Какое количество десятичных чисел можно закодировать  $N$ -ным количеством двоичных разрядов?
5. Какие системы счисления используются в ЭВМ?
6. Определите правила перевода чисел из двоичной системы в десятичную.
7. Определите правила перевода чисел из десятичной системы в двоичную.
8. Определите правила перевода чисел из десятичной системы счисления в 16-ричную.

## **Лабораторная работа №2**

### **Извлечение команд**

#### *Теоретический материал*

Микропроцессор (МП) - основная полупроводниковая схема, выполняющая в компьютере обработку информации и содержащая арифметико-логическое устройство, регистровую память, КЭШ, устройство управления, шину сопряжения компонентов, порты (адресуемые регистры ввода-вывода) и т. д. КЭШ - быстродействующая память относительно небольшого объема, предназначенная для временного хранения часто используемых данных.

Микропроцессор характеризуется набором команд, системой адресации команд и данных, типом (форматом) обрабатываемых данных, конвейерной обработкой и другими показателями.

Операционное устройство - программно-управляемое устройство обработки информации, являющееся частью оборудования МП, и содержащее арифметико-логическое устройство и регистровую память, выполняющее список микроопераций (синхросигналов), необходимых для реализации различных команд (арифметических, условных переходов, пересылки данных, сдвига, логических и других). В слове состояния операционного устройства устанавливаются в конце выполнения команды соответствующие флаги (биты-признаки), например, старший бит знака, переноса, паритета, нуля или переполнения. Память - оперативное запоминающее устройство, хранящее загружаемую операционную систему команды и данные программ, обрабатываемые микропроцессором в текущее время, в которую операционная система помещает прикладные программы, считываемые с диска для их выполнения. Память характеризуется объемом и быстродействием. Различают память статического и динамического способа действия. Память динамического типа требует периодической регенерации (восстановления) информации, содержащейся в последовательно расположенных ячейках памяти.

Устройство управления - устройство, координирующее работу компьютера и обеспечивающее управление операциями в нем

путем формирования микроопераций (синхросигналов), генерируемых в соответствии с кодом операции выполняемой команды. Наиболее широкое применение получили устройства с "хранимой логикой", которые содержат микропрограммную память (ПИЗУ), предназначенную для хранения последовательностей микрокоманд (элементарных операций), обеспечивающих выполнение машинных команд.

Счётчик команд - схема, предназначенная для временного хранения адреса команды, размещенной в памяти, содержимое которой, увеличивается для реализации последовательной (естественной) адресации ячеек памяти, и для хранения адреса перехода в случае необходимости нарушения последовательной адресации.

Регистр команд - регистр, служащий для временного хранения исполняемой команды.

Дешифратор команд (кода операций) - схема, определяющая последовательность формирования синхросигналов (микроопераций) устройство управления для реализации текущей команды программы.

### *Выполнение работы*

Интерфейс разработанного приложения представлен на рис 1.

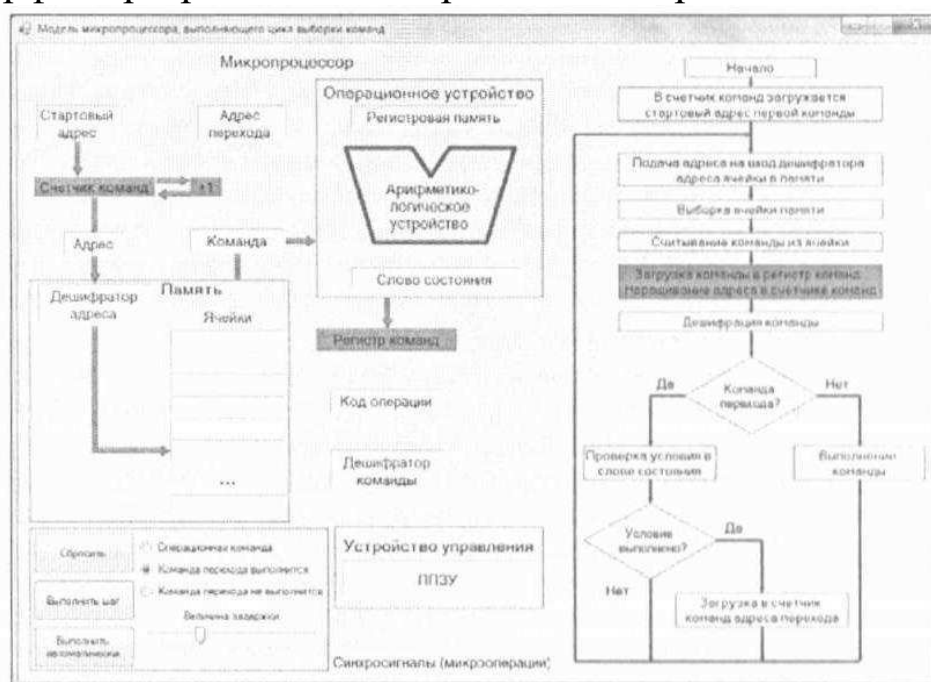


Рис. 1 – Окно программы



Предлагаемая демонстрационная модель микропроцессора, подробно показывает процесс циклической выборки команд. Выполнение каждой итерации процесса выборки команд сопровождается подсветкой соответствующих вычислительных блоков микропроцессора и вершин на блок-схеме алгоритма работы.

Модель реализует цикл выборки команд и позволяет продемонстрировать работу микропроцессора и следующих случаях:

1. Выбрана операционная команда.
2. Выбрана команда перехода, но переход не будет осуществлен (не выполнены условия перехода).
3. Выбрана команда перехода, и переход будет выполнен.

В левой нижней части приложения расположена панель, на которой находятся все управляющие элементы, это:

1. Кнопки запуска моделирования и сброса в начальное состояние.
2. Группа кнопок `RadioButton`, выбора типа команды.
3. Ползунок для регулирования задержки в автоматическом режиме.

Запустить программу можно в двух режимах (соответственно по названию кнопок):

а) пошаговый: пользователь управляет переключением модели на  $j$  следующий шаг алгоритма;

б) автоматический: программа демонстрирует процесс моделирования. Задержка между шагами регулируется пользователем с помощью ползунка.

На каждом шаге моделирования приложение подсвечивает выполняемый блок в блок-схеме моделируемого алгоритма (расположен в правой части приложения) и блоки устройств в модели, которые задействованы в текущем шаге, или блоки данных, пересылаемых другому блоку модели.

Выбрать тип команды, обработку которой демонстрирует модель на текущем проходе цикла выборки, пользователь может с помощью группы кнопок `RadioButton`, модель подстроится подтип команды в процессе моделирования.

Для получения дополнительной информации о блоке процессора предусмотрена функция всплывающей подсказки, пользователь может кликнуть по одному из блоков процессора и получить сообщение с подробным описанием его состава и функций, которые он выполняет.

### **Контрольные вопросы**

1. Что такое микропроцессор?
2. Что такое кэш-память?
3. Чем характеризуется микропроцессор?
4. Что представляет собой устройство управления?
5. Что такое счётчик команд?
6. Что такое регистр команд?
7. Что такое дешифратор команд?
8. Что такое арифметико-логическое устройство?

## Лабораторная работа №3 Трёхшинная конфигурация

### *Теоретический материал*

Для управления работой компонентов в системе микропроцессор использует шину данных, шину адреса и шину управления (рис. 2).

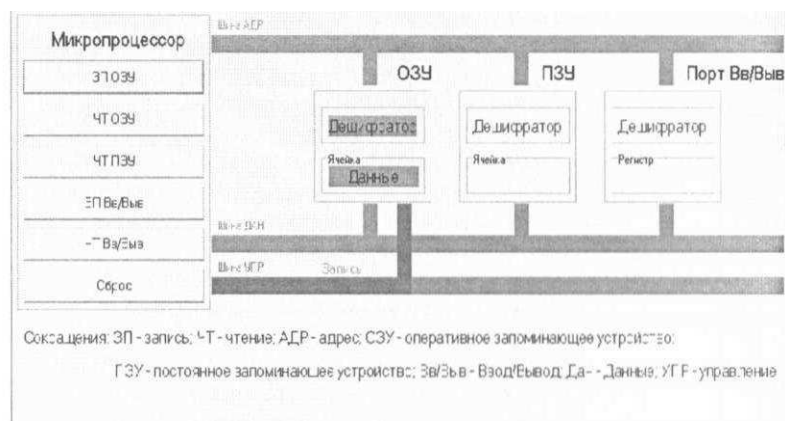


Рис. 2 – Трёхшинная конфигурация

Шина данных - это двунаправленная шина с тремя состояниями, к которой в общем случае подключены многочисленные источники и приемники информации, но в текущий момент времени к шине для совместной работы подключены только два устройства: активное (микропроцессор) и пассивное, например, оперативная память (ОЗУ). Третье состояние шины обеспечивается специальными шинными формирователями, характеризующимися высоким сопротивлением (состояние отключено), если их работа не разрешена соответствующим сигналом.

ОЗУ является источником и приемником данных, поэтому микропроцессор (МП) выполняет операцию записи или чтения памяти. В первом случае МП устанавливает адрес на шину адреса для выбора ячейки памяти и данные на шину данных, а затем направляет сигнал записи по соответствующей линии шины управления. В режиме чтения МП также устанавливает адрес на шину адреса и сигнал чтения памяти на одноименной линии шины

управления, который вызывает появление данных на шине данных. После этого МП принимает эти данные в свой внутренний регистр.

Аналогичным образом осуществляется доступ к порту ввода и вывода (Вв/Выв) с учетом операций записи (Зп Вв/Выв) и чтения (Чт Вв/Выв), выполняемых МП. Для считывания данных из ПЗУ МП устанавливает адрес и сигнал чтения ПЗУ, вызывающий появление данных на соответствующей шине.

Для опознавания устройств, применяются дешифраторы адресов, предназначенные для выбора конкретной ячейки памяти или порта ввода/вывода.

При нажатии кнопки, например, запись Зп ОЗУ на экране визуально демонстрируется: передача адреса по соответствующей шине адреса (АДР), его дешифрация, вывод данных по шине данных (ДАН) и установка сигнала записи в ячейку по шине УПР.

Сигнал записи является магистральным, то есть является доступным для всех устройств, но только в одном устройстве, опознавшим собственный адрес, произойдет запись данных в ячейку или регистр.

При чтении данных из ПЗУ также показывается передача адреса по шине адреса (АДР) и установка магистрального сигнала чтения, вызывающего появление данных на шине ДАН. Эти действия происходят по нажатию кнопки Чт ПЗУ, причем в ячейке ПЗУ демонстрируется подсветкой наличие информации.

#### *Выполнение работы.*

Выполнить последовательно операции чтения ОЗУ, запись в ОЗУ, чтение ПЗУ, чтение из порта ввода/вывода, запись в порт ввода/вывода. Описать порядок прохождения каждой операции (приложить соответствующий скриншот выполнения операции на модели).

### **Контрольные вопросы**

1. Что такое шина?
2. Какие шины составляют системную шину ЭВМ?
3. Что такое шина данных?
4. Какие адреса передаются по шине адреса?
5. Для чего используется шина управления?

## Лабораторная работа №4 Организация памяти

### *Теоретический материал*

Память с произвольной выборкой (основная память) - это устройство хранения информации (команд, данных), которая может быть извлечена в любой момент времени для дальнейшей ее обработки. Основная память есть запоминающее устройство (энергозависимое, для считывания и записи текущей информации), поэтому его называют еще оперативным (ОЗУ). Известны несколько типов ОЗУ: статические и динамические. В статических ОЗУ основным элементом памяти является триггер, а в динамических - конденсатор, который требует периодической подзарядки (регенерации) для сохранения единичного бита информации (наличие заряда). В режиме регенерации, каждый раз, через определенное время, выполняется считывание значений из ячеек памяти и автоматическая их перезапись. Там, где хранится логическая 1, присутствует на конденсаторе заряд, а где ее нет - заряд отсутствует, что соответствует логическому 0. Для реализации большой емкости памяти наиболее употребительны ОЗУ динамического типа, так как для построения элемента памяти, хранящего бит информации, требуется меньшее число транзисторов, чем в статической памяти. Кроме того, динамическая память потребляет меньше энергии (мощности). Недостатком ОЗУ динамического типа является наличие дополнительных схем регенерации памяти. Поэтому для создания памяти небольшой емкости используют ОЗУ статического типа. Микросхемы ОЗУ имеют битовую или байтовую (словарную) организацию, например,  $64\text{К} \times 1$  или  $2\text{К} \times 8$ , но, в любом случае, применяется матричное расположение битов информации в микросхеме. Число выводов корпуса микросхемы ограничивает реализацию байтового варианта микросхемы. Для построения памяти емкостью  $64\text{К} \times 8$  требуется 8 микросхем  $64\text{К} \times 1$ , образующих одну линейку.

На рис. 3 представлена схема памяти  $1\text{К} \times 8$ , построенная на микросхемах  $1\text{К} \times 1$ .



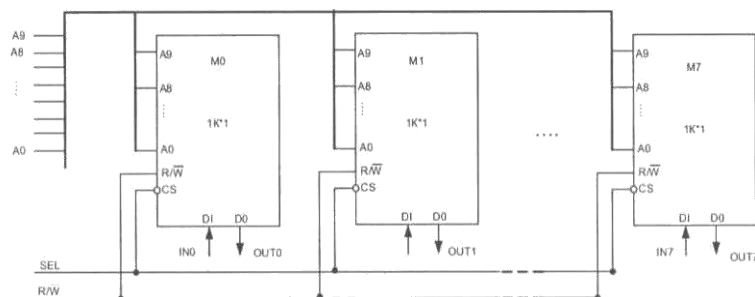


Рис. 3 – Схема ЗУ 1К\*8

На схеме биты адреса, разделенные на две группы A4-A0 и A9-A5 для выбора соответственно строк и столбцов, на пересечении которых находится необходимый элемент памяти, причем входы адреса столбцов управляют формирователями и усилителями считывания.

В схеме ЗУ все адресные линии объединены в одну шину и подключены параллельно с одноименными номерами. Каждая микросхема имеет один ввод бита данных (IN) и один вывод (OUT), то есть содержит отдельные контакты. Если выбор ( $SEL=0$ ) микросхем разрешен - то выполняется запись или чтение 8 битов данных, с помощью сигнала на линии R/W. При отсутствии сигнала  $SEL=1$  микросхемы переводятся в состояние "отключено", что позволяет наращивать микросхемы, образуя матричное их соединение (несколько строк и несколько столбцов).

На рис. 4 изображена структурная схема ЗУ 1 К\* 1.

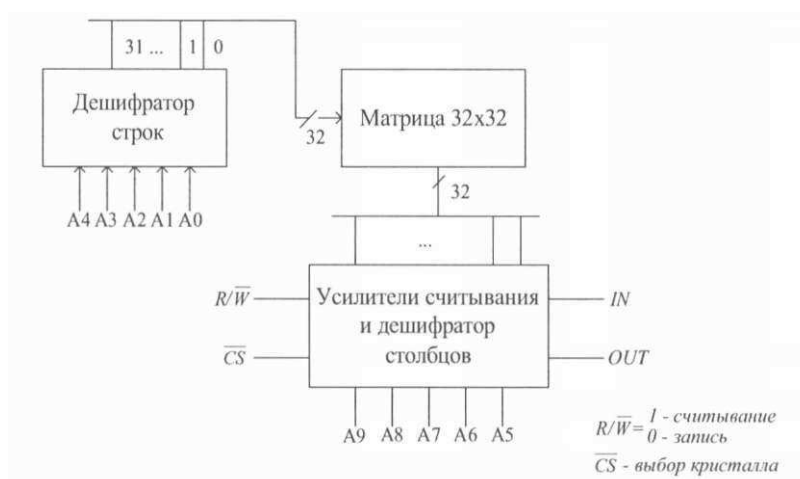


Рис. 4 – Структурная схема ЗУ 1 К\* 1

На схеме биты адреса, разделенные на две группы А4-А0 и А9-А5 для выбора, соответственно, строк и столбцов, на пересечении которых находится необходимый элемент памяти, причем входы адреса столбцов управляют формирователями и усилителями считывания.

Различают в общем случае три адресных пространства: ввода-вывода (адресуемых регистров), памяти данных команд (ОЗУ) и памяти конфигурации. Причем, в периферийных микропроцессорных устройствах эти адресные пространства могут объединяться в одну общую память. В этом случае возникает задача выделения с помощью дешифраторов одной части адресов под порты ввода-вывода, другой части под ОЗУ (RAM) и еще одной части под постоянную память (ROM). В системных процессорах эти пространства разделены и для реализации этой цели используются дополнительные сигналы, указывающие к какому пространству относится тот или иной адрес команды (рис. 16).

Внешний сигнал своим низким уровнем, указывает, что обращение при выполнении ассемблерной команды MOV осуществляется к ОЗУ. Если выполняется ассемблерная команда IN или OUT, то микропроцессором формируется сигнал высокого уровня (1) и доступ выполняется к порту ввода-вывода.

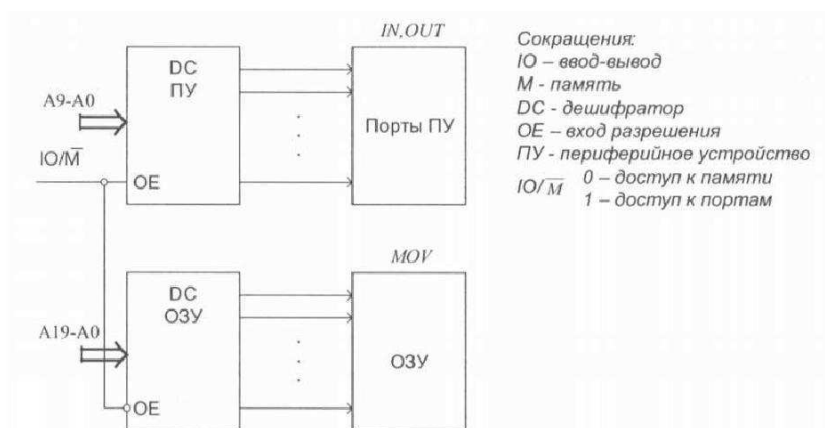


Рис. 5 – Разделенные адресные пространства ПУ и ОЗУ

Рассмотрим задачу дешифрации адреса, когда в микропроцессорной системе содержится три запоминающих устройства, каждый из которых имеет емкость 16К (рис. 6).

	Номер ОЗУ			Номер ячейки												16-разрядное значение	
	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1		A0
ОЗУ0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000H
	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	1FFF
ОЗУ1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	2000H
	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	3FFF
ОЗУ2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4000H
	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	5FFF

Рис. 6 - Распределение адресов между ОЗУ0, ОЗУ 1 и ОЗУ2

На рис. 6 изображена схема дешифрации адресов для трех ОЗУ, построенная в соответствии с рис. 7.

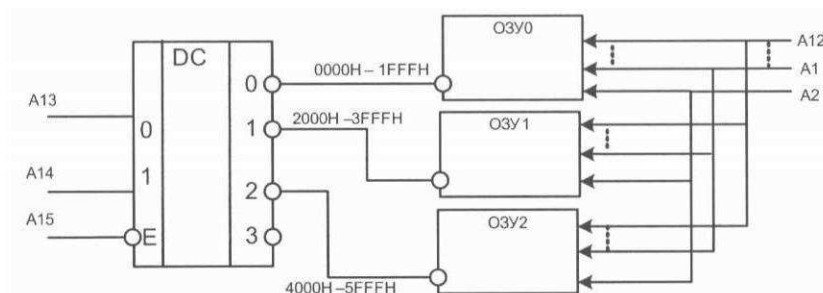


Рис. 7 - Схема дешифрации адресов

Меняя значения в битах адреса A14 и A13 (00, 01 и 10), можно осуществить выбор соответствующего ОЗУ. Биты A12...A1 A0 поступают параллельно и непосредственно на адресные входы ОЗУ2, ОЗУ1, ОЗУ0 для доступа к соответствующим ячейкам памяти с помощью встроенных дешифраторов.

### *Выполнение работы.*

#### Статическая память.

На рис. 8 представлена модель статической памяти, в которой реализуется описанный выше способ дешифрации адресов, когда старшая часть битов адреса используется для выбора блока памяти (БП), а младшие биты применяются для доступа к ячейкам памяти этого блока.

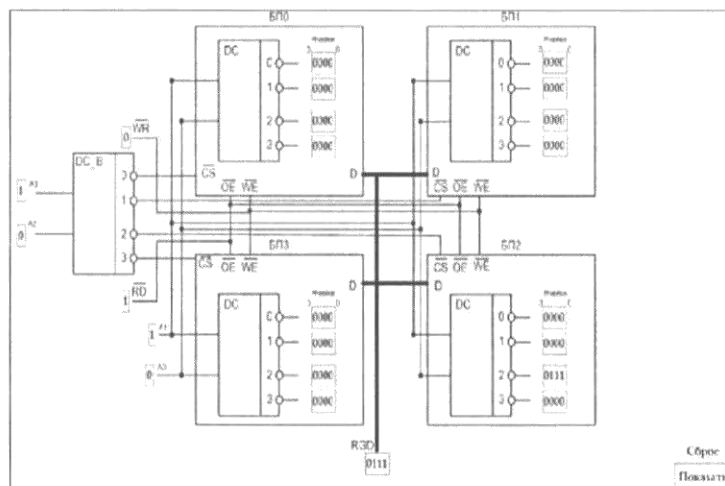


Рис. 8 – Модель статической памяти с каскадной дешифрацией адресов ячеек

Модель демонстрирует принцип работы памяти с учетом диаграмм записи и чтения данных.

В режиме записи необходимо в адресных окнах ввода задать значение номера блока и номера ячейки памяти. Набрать в окне регистра данных RGD двоичный код. Затем установить активное значение сигнала записи WR равное 0. Причем значение сигнала чтения RD должно быть пассивным и равным 1. С помощью клавиши мыши нажать кнопку «Показать». На экране монитора наблюдается выбор блока памяти и ячейки с записанным в нее двоичным кодом.

Аналогичным образом осуществляется режим чтения записанной ранее двоичной информации. Устанавливается адрес соответствующего блока памяти (ячейки) и активный сигнал чтения RD=0 (WR= 1). После этого нажимается кнопка «Показать» и в окне регистра RGD наблюдается код, считанный из выбранной ячейки памяти. Предварительно окно ввода RGD должно быть очищено с помощью клавиатуры.

Динамическая память.

На рис. 9 показана упрощенная модель динамической памяти с матрицей элементов памяти 4\*4, схемой элемента памяти, содержащего конденсатор С и транзисторы, и схемами дешифрации и регенерации заряда конденсаторов.

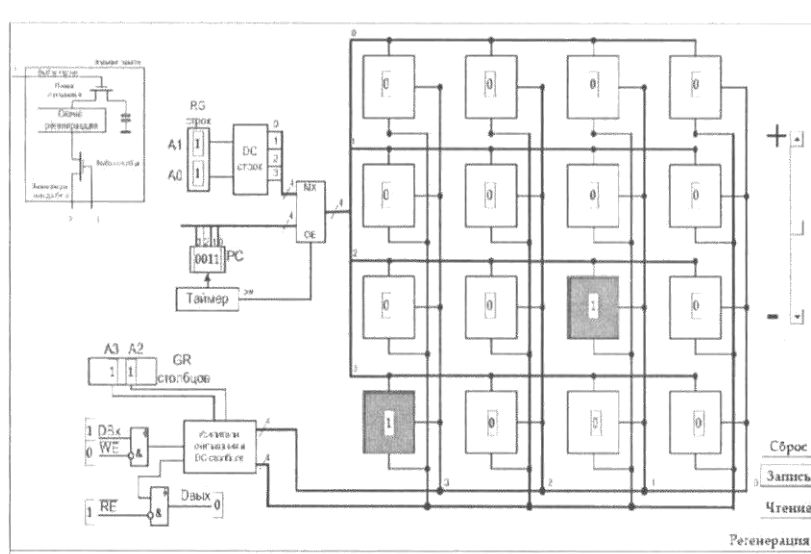


Рис. 9 - Модель динамической памяти

Единичное состояние элемента памяти отмечается красным цветом (конденсатор заряжен). Модель демонстрирует режимы: запись, чтение и регенерацию памяти. В режиме записи бита устанавливаются адреса строки (A1 A0), столбца (A3 A2), значение бита на DVx (0 или 1) и активный сигнал WE=0. После этого нажимается кнопка «Запись». На экране монитора наблюдается заряд конденсатора С для единичного значения красным цветом и выбор элемента памяти и подсвечивается схема регенерации памяти в элементе памяти, демонстрируя подзарядку конденсатора С (повторная запись). Разряд конденсаторов со временем происходит по всей строке и для сохранения информации необходимо через определенные интервалы времени, определяемые таймером, с помощью счетчика СТ2 осуществляется периодическое считывание и восстановление заряда конденсатора, то есть выполняется регенерация всех строк.

Для демонстрации этого режима нажимается кнопка «Регенерация». В элементах памяти матрицы розовым цветом отмечается разряженное состояние конденсаторов, соответствующее единичным состояниям. После этого таймером запускается в работу счетчик СТ2, который через мультиплексор МХ осуществляет последовательное сканирование каждой строки. После выбора строки там, где был розовый цвет (разряженное состояние), происходит его замена на красный цвет (восстановленное состояние конденсатора). Этот процесс



повторяется до тех пор, пока не наступит сигнал «Сброс» (нажатие соответствующей кнопки). В этом режиме дешифратор (DC) столбцов не участвует, входа DVx и выхода OВых отключены ( $WE=RE=1$ ).

В режиме Чтения с помощью содержимого регистра RG строк и регистра KC столбцов выбирается элемент памяти, устанавливается активный сигнал разрешения чтения  $RE=0$  ( $WE=1$ ) и на выходе ПВых наблюдается значение выбранного элемента памяти (1 или 0). Предварительно в элементах памяти устанавливаются единичные биты (красный цвет). Предусматривается занесение первых битов путём щелчка кнопкой мыши по элементу памяти в режиме регенерации.

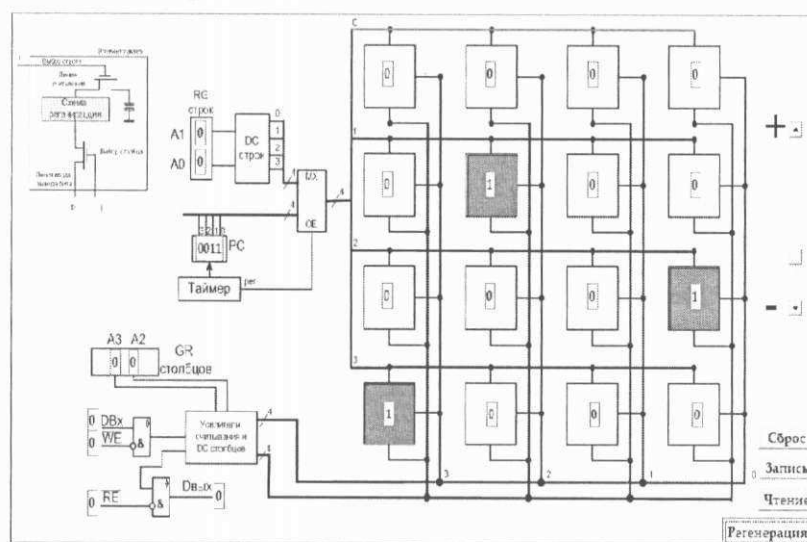


Рис 10 - Пример работы с моделью динамической памяти  
Имитационная модель приведена в папках «Статическая и динамическая памяти».

### Контрольные вопросы

1. Дайте определение статической памяти.
2. Дайте определение динамической памяти.
3. В чём преимущества и недостатки статической памяти?
4. В чём преимущества и недостатки динамической памяти?
5. Что является ячейкой статической памяти?
6. Что является ячейкой динамической памяти?
7. Какой вид памяти обладает большим быстродействием?

## **Лабораторная работа №5**

### **Диалоговые протоколы**

Практически все протоколы верхних уровней строятся на основе диалога: первая сторона отправляет запрос, вторая сторона отвечает на него. В ряде протоколов диалог может состоять из одного запроса и одного ответа (например, в протоколе HTTP), в других диалог может включать десятки запросов и ответов (например, FTP, POP3, SMTP).

#### **Протокол SMTP**

Протокол SMTP (Simple Mail Transfer Protocol, простой протокол передачи почты) отвечает за отправку почты другому адресату.

Сеанс работы протокола SMTP состоит из нескольких этапов: аутентификация, передача параметров письма (адреса, тема), отправка текста сообщения и завершение соединения.

Поскольку работа с почтой производится от имени конкретного пользователя, то первым этапом диалога идёт авторизация – клиент передаёт серверу имя пользователя и пароль. Оба протокола поддерживают безопасные методы аутентификации с применением протокола SSL, однако для данной практической работы для простоты используется нешифрованная аутентификация. Протокол SMTP кодирует пароль кодировкой base64, что обеспечивает минимальную защиту, которая, впрочем, легко вскрывается.

К письму могут прилагаться двоичные файлы, но из-за того, что протокол предусматривает передачу только ASCII-информации, файлы должны быть закодированы в текстовый формат по стандарту MIME.

Таблица 2 – Основные команды протокола SMTP

Команда	Параметры	Описание
HELO	домен	Начало сессии
AUTH LOGIN		Аутентификация: сервер запросит логин и пароль, закодированные с помощью base64
MAIL FROM:	<e-mail отправителя>	Адрес отправителя письма
RCPT TO:	<e-mail получателя>	Адрес получателя письма
DATA		После этой команды сервер все данные воспринимает как тело письма. Признаком окончания тела является строка с единственной точкой.
HELP	[команда]	Вывод справочной информации по указанной команде (если она указана)
QUIT		Завершение сеанса

### Протокол POP3

Протокол POP3 (Post Office Protocol) предназначен для получения почты с почтового сервера на локальную машину.

Таблица 3 – Основные команды протокола POP3:

Команда	Параметры	Описание
USER	имя пользователя	Аутентификация пользователя
PASS	пароль	Пароль открытым текстом
STAT		Получение количества сообщений в ящике и их размер
LIST	[сообщение]	Получение информации обо всех сообщениях в ящике (или о конкретном, если оно указано)
DELE	сообщение	Удаление указанного сообщения
RETR	сообщение	Загрузка указанного сообщения
QUIT		Завершение сеанса

## Используемые утилиты

Поскольку протоколы POP3 и SMTP основаны на текстовом диалоге между клиентом и сервером, допустимо использовать вместо специальной почтовой программы обычный клиент `telnet` или одну из совместимых с этим протоколом альтернативных программ. Для подключения к почтовому серверу нужно знать имя сервера и порт.

Имя почтового сервера может не совпадать с именем почтового адреса, например, для адреса `user@mail.ru` почтовый сервер для получения почты будет `pop.mail.ru`, а для отправки почты - `smtp.mail.ru`. Имена почтовых серверов для каждого домена обычно хранятся в записи типа MX на DNS-сервере (а при её отсутствии – используется адрес сервера из записи типа A), кроме того, их обычно можно получить в документации почтовой службы. Для протокола POP3 стандартный номер порта 110, для SMTP – 25, однако они также могут отличаться в разных службах.

Для кодирования и декодирования пароля для протокола SMTP можно использовать одну из подходящих утилит, или же один из онлайн-перекодировщиков.

### Задание

При выполнении практической работы потребуется подключение к почтовому серверу. Допустимо использовать любой сервер, на котором имеется учетная запись.

Используя SMTP-сервер, отправить письмо на свой адрес. Используя POP3-сервер получить количество писем в ящике, список писем с информацией о них, загрузить письмо, отправленное через SMTP-сервер.

В отчёт следует вставить полные логи сеансов связи с каждым из серверов. Для сохранения пароля к учетной записи в секрете пароль из отчета нужно удалить.

### Контрольные вопросы

1. Из каких этапов состоит сеанс работы протокола SMTP?
2. Какие вы знаете команды протокола SMTP?
3. Для чего предназначен протокол POP3?
4. Какие вы знаете команды протокола POP3?
5. Что делает команда RETR?
6. Что делает команда DATA?



## Лабораторная работа №6

### Команды диагностики сети

Команды тестирования сети входят в состав сетевых операционных систем и позволяют просмотреть параметры отдельных узлов сети, проверить их работоспособность и в случае обнаружения проблемы – определить проблемный участок.

#### Используемые утилиты

Утилита `ipconfig` предназначена для отображения параметров текущей конфигурации сети TCP/IP, таких как IP-адрес, маска сети, шлюз по умолчанию, а также параметры DHCP и DNS. По умолчанию команда выводит сокращенный набор параметров, однако с помощью дополнительных ключей можно вывести полный список.

Утилита `ping` позволяет проверить наличие IP-соединения с другим узлом сети, а также время оборота пакета. Дополнительные ключи позволяют изменить количество попыток соединения, время ожидания ответа и т. д.

Утилита `tracert` позволяет определить путь до узла назначения в виде списка маршрутизаторов между исходным и конечным узлами, а также время оборота пакета. Если очередной узел не ответил в заданное время, то вместо времени оборота программа ставит звездочку. Адрес узла может быть определен только в том случае, если от него пришёл хотя бы один ответ.

Утилита `nslookup` служит для получения информации о домене. Утилита подключается к указанному DNS-серверу (если сервер не задан, то используется сервер текущего подключения) и извлекает поля требуемого типа. Если DNS-сервер ответственен за запрашиваемую зону, то ответ считается достоверным.

Утилита `netstat` предназначена для отображения активных TCP-соединений, прослушиваемых узлом портов, таблицы маршрутизации узла, а также статистики Ethernet и протоколов сетевого и транспортного уровней.

#### Задание

Использовать команды сетевой диагностики `nslookup`, `ping`, `tracert` для получения информации о двух доменах, проверки их работоспособности, отслеживания пути. Домены можно выбрать

произвольные, но не рекомендуется использовать общеизвестные домены (такие, как `google.com` или `yandex.ru`), а также домены томской сети.

Использовать команду `ipconfig` для выяснения IP-адреса компьютера, маски сети и шлюза по умолчанию, а также дополнительных параметров. Отобразить кэш DNS.

Использовать команду `netstat` для отображения статистики протоколов и открытых соединений/портов. Привести список процессов, прослушивающих порты.

Для каждой из команд следует постараться максимально использовать доступные опции.

### **Контрольные вопросы**

1. Определите назначение утилы `ipconfig`.
2. Определите назначение утилиты `ping`.
3. Определите назначение утилиты `tracert`.
4. Определите назначение утилиты `nslookup`.
5. Определите назначение утилиты `netstat`.
6. Что такое DNS?
7. Что такое маска сети?
8. Что такое шлюз?
9. Что такое DHCP?
10. Из каких частей состоит IP-адрес?

## **Лабораторная работа №7**

### **Анализ пакетов локальной сети**

В обычном режиме работы сетевая карта узла обрабатывает только те пакеты, которые адресованы данному узлу, или же всем узлам сети, если пакет широковещательный. Однако существует и другой режим работы сетевой карты – так называемый *неразборчивый режим* (Promiscuous mode), в котором сетевая карта получает все проходящие по сети пакеты.

Поскольку данный режим позволяет при определенных условиях перехватывать и анализировать трафик, предназначенный для чужих узлов, использование данного режима является незаконным в ряде стран, но даже если закон не запрещает его использование явно, перехват чужих данных может привести к возникновению административной или уголовной ответственности. Тем не менее, данный режим полезен в ряде случаев администраторам сети, поскольку позволяет отследить проблемный трафик (например, широковещательный шторм) и выяснить его причину.

Анализатор трафика (sniffer, сниффер) – это программа, предназначенная для перехвата и анализа сетевого трафика.

Чтобы перевести сетевую карту в неразборчивый режим, требуются привилегии администратора локальной машины, поэтому данная Лабораторная работа в учебной аудитории выполняется в виртуальной машине.

#### **Задание**

Выполнить захват трафика, произвести его анализ.

#### **Последовательность выполнения работы**

- 1) Запустите программу VirtualBox.
- 2) Выберите и запустите виртуальную машину Ubuntu.
- 3) Откройте терминал двойным щелчком по ярлыку на рабочем столе.
- 4) Введите команду `sudo wireshark`.
- 5) Когда терминал запросит пароль, введите `ips`.
- 6) В секции *Capture* в окне программы Wireshark выберите интерфейс `eth0`.



7) Запустите браузер FireFox и откройте какой-либо сайт, закройте браузер.

8) С помощью фильтра на панели инструментов отфильтруйте захваченный трафик, оставив только пакеты протокола TCP.

9) Скопируйте в отчёт строки анализатора трафика, относящиеся к следующим событиям (если за время захвата трафика было установлено несколько TCP-соединений, удалите из отчёта лишние строки):

- a) установка TCP-соединения;
- b) передача данных;
- c) завершение TCP-соединения.

10) Закройте Wireshark (на вопрос программы о сохранении результатов ответьте отрицательно) и другие открытые в виртуальной машине окна.

11) Завершите работу виртуальной машины: выберите в меню *Машина* пункт *Заккрыть...*, отметьте пункт *Сохранить состояние машины* и нажмите кнопку ОК.

12) Дождитесь сохранения состояния виртуальной машины.

13) Закройте программу VirtualBox.

### **Контрольные вопросы**

1. Что такое сетевая карта?
2. Какая сеть называется локальной?
3. Что такое пакет?
4. В чём состоит неразборчивый режим работы сетевой карты?

## Список литературы

1. Гребешков, А.Ю. Вычислительная техника, сети и телекоммуникации [Текст] : учебное пособие : [для студентов вузов, обуч. по направлению подготовки 11.03.02 - "Инфокоммуникационные технологии и системы связи" квалификации (степени) "бакалавр" ] / А. Ю. Гребешков. - Москва : Горячая Линия–Телеком, 2016. - 190 с.

2. Гриценко, Ю.Б. Вычислительные системы, сети и телекоммуникации [Электронный ресурс] : учебное пособие / Ю.Б. Гриценко ; Министерство образования и науки Российской Федерации ; Томский Государственный Университет Систем Управления и Радиоэлектроники (ТУСУР) ; Факультет дистанционного обучения. - Томск : ТУСУР, 2015. - 134 с. Режим доступа: <http://biblioclub.ru/>

3. Пятибратов, А.П. Вычислительные системы, сети и телекоммуникации [Электронный ресурс] : учебник / А.П. Пятибратов, Л.П. Гудыно, А.А. Кириченко ; под ред. А. П. Пятибратов. - 4-е изд., перераб. и доп. - Москва : Финансы и статистика, 2014. - 735 с. Режим доступа: <http://biblioclub.ru/>

4. Бройдо, В.Л. Вычислительные системы, сети и телекоммуникации [Текст] : учебное пособие / В. Л. Бройдо. - 2-е изд. - СПб. : Питер, 2005. - 703 с.

## Содержание отчета

Перед выполнением лабораторной работы студент обязан ознакомиться с теоретическим материалом по данной теме.

**Отчет по лабораторной работе должен содержать:**

- 1) тему работы;
- 2) условие задания;
- 3) ход выполнения работы;
- 4) результаты выполнения работы;
- 5) выводы.

Лабораторная работа считается выполненной, если все задания выполнены, получены соответствующие результаты, составлен полный отчёт по указанной форме. После выполнения студент допускается к защите работы.

На защиту выносятся:

- 1) теоретические сведения по теме данной работы;
- 2) результаты выполнения работы.

Лабораторная работа считается выполненной, если она выполнена и защищена.