

Министерство образования и науки Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Магнитогорский государственный технический университет им. Г.И. Носова»  
Многопрофильный колледж



**МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ  
ПРАКТИЧЕСКИХ РАБОТ**

по учебной дисциплине  
**Архитектура компьютерных систем**  
для студентов специальности  
**09.02.03 «Программирование в компьютерных системах»**  
базовой подготовки

Магнитогорск, 2017

## **ОДОБРЕНО**

Предметно-цикловой комиссией «Информатики и вычислительной техники»

Председатель И.Г.Зорина

Протокол № 7 от 14 марта 2017 г.

Методической комиссией

Протокол №4 от 23 марта 2017г

## **Составители**

преподаватель ФГБОУ ВО МГТУ МпК М.А. Путилина

преподаватель ФГБОУ ВО МГТУ МпК А.А. Андре

Методические указания по выполнению практических работ разработаны на основе рабочей программы учебной дисциплины «Архитектура компьютерных систем»

Содержание практических работ ориентировано на подготовку студентов к освоению профессионального) модуля основной профессиональной образовательной программы по специальности 09.02.03 Программирование в компьютерных системах и овладению профессиональными компетенциями.

## СОДЕРЖАНИЕ

1 Введение	4
2 Методические указания	6
Практическая работа 1	6
Практическая работа 2	8
Практическая работа 3	11
Практическая работа 4	13
Практическая работа 5	16
Практическая работа 6	21
Практическая работа 7	23
Практическая работа 8	25
Практическая работа 9	27
Практическая работа 10	28
Практическая работа 11	31
Практическая работа 12	32

## 1 ВВЕДЕНИЕ

Важную часть теоретической и профессиональной практической подготовки студентов составляют практические работы.

Состав и содержание практических работ направлены на реализацию действующего федерального государственного образовательного стандарта среднего профессионального образования.

Ведущей дидактической целью практических работ является экспериментальное подтверждение и проверка существенных теоретических положений (законов, зависимостей).

В соответствии с рабочей программой учебной дисциплины «Архитектура компьютерных систем» предусмотрено проведение лабораторных работ.

В результате их выполнения, обучающийся должен:

уметь:

- получать информацию о параметрах компьютерной системы;
- подключать дополнительное оборудование и настраивать связь между элементами компьютерной системы;
- производить инсталляцию и настройку программного обеспечения компьютерных систем;

Содержание лабораторных работ ориентировано на подготовку студентов к освоению профессионального модуля основной профессиональной образовательной программы по специальности и овладению профессиональными компетенциями:

ПК 1.1. Выполнять разработку спецификаций отдельных компонент

ПК 1.2. Осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля

ПК 1.5. Осуществлять оптимизацию программного кода модуля

ПК 2.2. Реализовывать базу данных в конкретной системе управления базами данных (СУБД).

ПК 2.4. Реализовывать методы и технологии защиты информации в базах данных

ПК 3.1. Анализировать проектную и техническую документацию на уровне взаимодействия компонент программного обеспечения

ПК 3.2. Выполнять интеграцию модулей в программную среду

ПК 3.4. Осуществлять разработку тестовых наборов и тестовых сценариев

А также формированию общих компетенций

ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес

ОК 2. Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество

ОК 3. Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность

ОК 4. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития

ОК 5. Использовать информационно-коммуникационные технологии в профессиональной деятельности

ОК 6. Работать в коллективе и в команде, эффективно общаться с коллегами, руководством, потребителями

ОК 7. Брать на себя ответственность за работу членов команды (подчиненных), результат выполнения заданий

ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации

ОК 9. Ориентироваться в условиях частой смены технологий в профессиональной деятельности

Выполнение студентами практических работ по учебной дисциплине «Архитектура компьютерных систем» направлено на

- обобщение, систематизацию, углубление, закрепление, развитие и детализацию полученных теоретических знаний по конкретным темам учебной дисциплины;

- формирование умений применять полученные знания на практике, реализацию единства интеллектуальной и практической деятельности;

- формирование и развитие умений: наблюдать, сравнивать, сопоставлять, анализировать, делать выводы и обобщения, самостоятельно вести исследования, пользоваться различными приемами измерений, оформлять результаты в виде таблиц, схем, графиков;

- приобретение навыков работы с различными приборами, аппаратурой, установками и другими техническими средствами для проведения опытов;

- выработку при решении поставленных задач профессионально значимых качеств, таких как самостоятельность, ответственность, точность, творческая инициатива.

Продолжительность выполнения практической работы составляет не менее двух академических часов и проводится после соответствующей темы, которая обеспечивает наличие знаний, необходимых для ее выполнения.

## 2 МЕТОДИЧЕСКИЕ УКАЗАНИЯ

### Тема 1.2 Арифметическая и логическая организация ЭВМ

#### Практическая работа № 1

#### Перевод чисел из одной системы счисления в другую

**Цель работы:** сформировать навыки и умения переводить числа из одной системы счисления в другую.

**Выполнив работу, Вы будете:**

уметь:

переводить числа из одной системы счисления в другую

**Материальное обеспечение:** персональный компьютер, калькулятор

**Порядок выполнения работы:**

1. Ознакомьтесь с теоретическими сведениями
2. Выполните 5 заданий
3. Решение и результаты оформите в тетради
4. Ответьте на вопросы преподавателя

**Краткие теоретические сведения:**

Для записи любой цифры восьмеричного необходимы три двоичные цифры (триады). Поэтому преобразуемое двоичное число разделяют справа налево на группы по три двоичных цифры, при этом самая левая группа может содержать меньше трёх двоичных цифр. Например, двоичная цифра 011 есть цифра три в восьмеричной системе счисления. Затем каждую группу двоичных цифр выражают в виде восьмеричной цифры, представленной в таблице:

Двоичная система	Восьмеричная система
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

Например, двоичное число 1101111011, разбитое на группы по три двоичные цифры, можно записать как 1 101 111 011 и затем после записи каждой группы одной восьмеричной цифрой получить восьмеричное число 1573.

Аналогично преобразуется двоичное число в шестнадцатеричную систему счисления. Преобразуемое двоичное число делят на группы по четыре двоичных цифры в каждой (тетрады), поскольку для записи любой цифры шестнадцатеричного числа необходимы четыре двоичных цифры.

Десятичная система	Двоичная система	Шестнадцатеричная система
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Поэтому двоичное число 1101111011, использованное в предыдущем примере, после разбиения на группы по четыре двоичных цифры, можно записать 11 0111 1011 и после выра-

жения каждой группы одной шестнадцатеричной цифрой получить шестнадцатеричное число 37В.

Например:

$$1\ 111\ 101\ 001_2 = 011\ 111\ 101\ 001_2 = 3754_8$$

$$11\ 1110\ 1001_2 = 0011\ 1110\ 1001_2 = 3E9_{16}$$

Выполним несколько заданий. Делим доску пополам, выходит два ученика, первая половина доски перевод в 8-ую, вторая половина доски – в 16-ую системы счисления.

### Задание 1

Переведите двоичные числа в восьмеричную систему счисления и шестнадцатеричную.

а) 11110110011 б) 1101101001001 в) 1001101011001

г) 11011111011 д) 1010111011101 е) 1110111101011

*Перевод чисел из восьмеричной и шестнадцатеричной систем счисления в двоичную.*

Преобразование восьмеричного или шестнадцатеричного числа в двоичное осуществляется простым переводом каждой цифры исходного числа в группу из трёх (триад – для восьмеричного) или из четырёх (тетрад – для шестнадцатеричного числа) двоичных цифр.

$$\text{Например, } 123_8 = 001\ 010\ 011_2$$

$$A17_{16} = 1010\ 0001\ 0111_2$$

Если после перевода целая часть двоичного числа начинается с нулей, то их отбрасывают. То же самое делают с нулями в конце дробной части.

### Задание 2.

1. Перевести восьмеричные числа в двоичную систему счисления с помощью таблицы:

а)  $324_8$  б)  $1576_8$  в)  $37,25_8$  г)  $206,125_8$

2. Перевести шестнадцатеричные числа в двоичную систему счисления с помощью таблицы:

а)  $A59_{16}$  б)  $87_{16}$  в)  $2CE_{16}$  г)  $1F5A_{16}$

### Задание 3.

Решите по вариантам следующие задачи

Вариант №1

Переведите двоичные числа:

а) 101011011; 1111110011; 100000001110 в восьмеричную систему счисления

б) 11110111011; 101010101; 111111 в шестнадцатеричную систему счисления

Вариант №2

Переведите двоичные числа:

а) 111011011; 000110101; 0101010111 в восьмеричную систему счисления

б) 00110011; 11100011101; 011011011 в шестнадцатеричную систему счисления

### Задание 4.

Постройте в координатной плоскости заданную фигуру по плану, предварительно осуществите перевод координат точек из двоичной системы счисления в десятичную.

1) Постройте окружность с центром в точке (1010, 1010), с радиусом 101;

2) Постройте точки и соедините их с отрезками, закрасьте соответствующим цветом.

Синий (11; 111), (100; 1000), (101; 111), (100; 110)

Синий (111; 111), (1000, 1000), (1001; 111), (1000; 110)

Красный (100; 100), (110; 11), (1000; 100)

### Задание 5.

Заполните магический квадрат цифрами в десятичной системе счисления

1)  $1000_2$  2)  $0001_2$  3)  $0110_2$  4)  $0011_2$  5)  $0101_2$

6)  $0111_2$  7)  $0100_2$  8)  $1001_2$  9)  $0010_2$

1	2	3
4	5	6
7	8	9

### Форма представления результата:

Выполненная работа

### Критерии оценки

Оценка «отлично» выставляется, если выполнены все задания практического занятия, допущены 1-2 недочеты, исправленные по требованию преподавателя.

Оценка «хорошо» выставляется, если работа выполнена в полном объеме, допущены одна ошибка или более двух недочетов при выполнении задания, исправленные по замечанию преподавателя.

Оценка «удовлетворительно» выставляется, если задания выполнены не в полном объеме, допущены 1-2 ошибки при выполнении заданий, но продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

Оценка «неудовлетворительно» выставляется, если выполнено менее половины заданий, не продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

## Тема 1.2 Арифметическая и логическая организация ЭВМ

### Практическая работа № 2

#### Представление чисел в машине

#### Цели работы:

Убедиться, что внутреннее представление чисел в современном компьютере по-прежнему двоичное

Проверить утверждение о том, что числа в памяти компьютера хранятся в виде двоичного кода

Убедиться, что данные в файлах представлены в двоичном виде. Проверить утверждение о принципиальной неразличимости двоичных кодов для разных видов информации.

#### Выполнив работу, Вы будете:

уметь:

проверять внутреннее представление чисел в компьютере

распознавать, что в памяти компьютера числа хранятся в двоичном виде

распознавать, что данные в файлах тоже представлены в двоичном виде

**Материальное обеспечение:** компьютер с лицензионным программным обеспечением (язык программирования Турбо Паскаль)

#### Порядок выполнения работы:

1. Наберите предложенные программы
2. Откомпилируйте программы и запустите на выполнение
3. Ознакомьтесь с теоретическими сведениями
4. Ответьте на вопросы преподавателя

В компьютерной литературе утверждается, что компьютер хранит и обрабатывает данные в двоичной системе. А действительно ли это так? Мы вводим десятичные числа и получаем на экране десятичный результат. Можно ли быть уверенным в том, что компьютер производил действия именно в двоичной системе? Как это проверить?

#### Программа для проверки внутреннего представления чисел в компьютере

```
PROGRAM BinaryRepresentation1;
VAR m,m0,i: INTEGER; x,h: REAL;
BEGIN m0:=128;
      FOR m:=m0-1 TO m0+1 DO
        BEGIN h:=1/m; x:=0;
              FOR i:=1 TO m DO x:=x+h;
```



```
WRITELN(m, ' ', x-1)
```

```
END
```

```
END.
```

### Краткие теоретические сведения:

Внешний цикл FOR обеспечивает трехкратное исполнение программы для значений  $m$  от  $m_0-1$  до  $m_0+1$  (в нашем случае это 127, 128 и 129). Нетрудно заметить, что речь идет о числе, являющемся точной степенью двойки (т.е. 128) и двух «соседних». Далее для каждого из трех чисел  $m$  раз прodelывается суммирование величины  $1/m$  и печатается разность, показывающая, насколько полученное число отличается от 1. Запустив программу, мы увидим на экране дисплея следующий результат:

```
127 -7.2759576142E-12
128      0.0000000000E+00
129      2.1827872843E-11
```

Иными словами, только для числа  $1/128$ , которое представляется в двоичной системе точно, результат получился хоть и очень маленький ( $10^{-11}-10^{-12}$ ), но ненулевой.

Может быть это случайность? Проверьте для других  $m_0$ , например 512 или 2048 – каков будет результат? В то же время, при  $m_0=126$ , все три результата будут отличны от нуля, т.е. степени двойки действительно являются «особенными».

**Вывод.** Полученные результаты можно объяснить единственным способом: компьютер действительно хранит числа и производит вычисления в двоичной системе. Именно поэтому только для степеней двойки ему удастся получить точный результат.

### Двоичное представление чисел в памяти

Можно ли заглянуть в память компьютера, чтобы увидеть, каким образом там храниться информация? Опишем еще один достаточно убедительный эксперимент, подтверждающий, что до сих пор информация в компьютере хранится именно в двоичной системе.

Попробуем извлечь целое число непосредственно из ОЗУ и поверить, действительно ли там находится тот самый двоичный код, о котором столько пишут в книгах по информатике.

Для проверки потребуется очень небольшая, но нетривиальная программа на языке Паскаль.

### Программа для проверки хранения чисел в виде двоичного кода

```
PROGRAM BinaryRepresentation2;
VAR i,m, b: BYTE;
BEGIN b:=65;
      m:=$80; {bin = 1000 0000}
      FOR i:=1 TO 8 DO
        BEGIN IF (MEM[SEG(b):OFS(b)] AND m) = 0
              THEN WRITE('0')
              ELSE WRITE('1');
              m:= m SHR 1
        END; WRITELN
      END.
```

Кратко разберем, как она работает. В «изучаемую» переменную  $b$  поместим некоторое значение, например 65. затем, предварительно подготовив в переменной  $m$  так называемую «маску» для выделения самого старшего бита числа, выведем на экран по очереди содержимое всех 8 битов переменной  $b$ . Для этого в цикле будем извлекать из памяти байт, в котором лежит значение переменной  $b$  (в программе это записано как довольно специфическое выражение  $\text{MEM}[\text{SEG}(b):\text{OFS}(b)]$ , которое действительно «напрямую» извлекает из ОЗУ значение «того самого» байта), выделять из него один бит с помощью операции  $\text{AND}$  и выводить результат анализа на экран. Подчеркнем, что последняя строка цикла сдвигает «маску» на один разряд вправо, тем самым в следующий раз будет выделяться соседний бит.

Запустив описанную программу, увидим на экране строку

```
01000001
```

которая и является двоичным представлением исходного числа 65. разумеется, можно повторить опыт и с другими значениями.

**Вывод.** В памяти компьютера числа хранятся в двоичном виде.

### Двоичное представление информации на диске

Что происходит при записи данных во внешнюю память, например, на магнитный диск? Для этого проведем новый эксперимент.

Рассмотрим еще один эксперимент, представляющий собой некоторое усовершенствование предыдущего опыта. Он показывает, что двоичное представление информации хранится не только в ОЗУ, но и в файлах на дисках.

Начнем с того, что в простом текстовом редакторе, таком как Блокнот, наберем единственный символ – латинскую заглавную букву А. Ничего больше не нажимая на клавиатуре (даже клавишу <Enter>), сохраним этот простейший текстовый файл из одной буквы под именем proba.txt.

Проверим, что получилось, используя широко распространенную программу Far (аналог NortonCommander для среды Windows). Используя для просмотра полученного файла традиционную клавишу <F3>, дополнительно переведем программу в режим Нех-кодов, нажав клавишу <F4>, и увидим следующую картину:

Proba.txt	DOS	1	Col 0 100%
00000000: 41		A	

Отчетливо видно, что текст состоит из единственного символа А (справа), имеющего шестнадцатеричный код 41 (слева). Попутно обратим внимание на выделенный размер файла – 1 байт.

Итак, первый вывод, который можно сделать уже сейчас, - символы в файлах на внешних устройствах хранятся в виде их числовых кодов. Но мы пойдем дальше. Наберем еще одну программу на Паскале, которая во многом совпадает с предыдущей.

### Программа для проверки двоичного представления данных на диске

```
PROGRAM BinaryRepresentation3;
VAR  b,i,m: BYTE; f: FILE OF byte;
BEGIN ASSIGN(f,'proba.txt'); RESET(f);
      READ(f,b);
      m:=$80; {bin = 1000 0000}
      FOR i:=1 TO 8 DO
        BEGIN IF (b AND m) = 0
              THEN WRITE('0')
              ELSE WRITE('1');
              m:= m SHR 1
        END;
      WRITELN; WRITELN(b);
      CLOSE(f);
END.
```

Приведенная выше программа читает из файла в переменную b значение его единственного байта, а затем выводит на экран в двоичном виде (алгоритм вывода полностью заимствован из предыдущего примера), а также в десятичном.

В результате исполнения программы на экране появятся две строки:

01000001

65

т.е. значение числа в двоичной и десятичной системах соответственно.

**Вывод.** Непосредственный вывод данных из файла продемонстрировал, что не только в ОЗУ, но и на диске данные хранятся в двоичном виде. Особо подчеркнем, что данный эксперимент дополнительно показал еще одно интересное свойство хранения информации в современных компьютерах – по виду двоичного кода принципиально невозможно определить, явля-

ется ли он числом, текстом или чем-то другим. Действительно, мы сохранили текст из единственной буквы, а прочитали из файла целое число.

#### **Форма представления результата:**

Выполненная работа

#### **Критерии оценки**

Оценка «отлично» выставляется, если выполнены все задания практического занятия, допущены 1-2 недочеты, исправленные по требованию преподавателя.

Оценка «хорошо» выставляется, если работа выполнена в полном объеме, допущены одна ошибка или более двух недочетов при выполнении задания, исправленные по замечанию преподавателя.

Оценка «удовлетворительно» выставляется, если задания выполнены не в полном объеме, допущены 1-2 ошибки при выполнении заданий, но продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

Оценка «неудовлетворительно» выставляется, если выполнено менее половины заданий, не продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

## **Тема 1.2. Логические основы ЭВМ, элементы и узлы**

### **Практическая работа № 3**

#### **Решение арифметических и логических задач**

##### **Цели работы:**

Научится строить функциональные схемы по формулам

##### **Выполнив работу, Вы будете:**

уметь:

Используя логические формулы строить функциональные схемы

**Материальное обеспечение:** тетрадь, ручка, линейка, карандаш

#### **Краткие теоретические сведения**

Сейчас выполняются интегральные схемы с **высокой** степенью интеграции – 80 млн. транзисторов. В интегральном исполнении выпускаются отдельные функциональные узлы ЭВМ: счетчики, сумматоры, дешифраторы.

Из основных логических элементов с помощью правил преобразования функций алгебры логики можно построить функциональную схему, реализующую любую сложную логическую операцию.

#### **Алгоритм построения функциональных схем**

1. Для каждой строки таблицы истинности с единичным значением построить минтерм. Минтермом называется терм-произведение (конъюнкция), в котором каждая переменная встречается только один раз – либо с отрицанием, либо без него. Переменные, имеющие нулевые значения в столбце, входят в минтерм с отрицанием, а переменные со значением единица – без отрицания.

2. Объединить все минтермы операцией дизъюнкции.

3. Упростить логическое выражение.

**ПРИМЕР 1.** Составить функциональную схему, выполняющую логическую операцию сравнения одноразрядных двоичных чисел А и В, с записью 1 при их неравенстве.

Условия работы запишем в виде таблицы истинности.

<b>A</b>	0	0	1	1
<b>B</b>	0	1	0	1
<b>P</b>	0	1	1	0

1. Выбираем столбцы, в которых  $P=1$ , и стоим для них минтермы.

1 строка –  $\bar{A} \cdot B$

2 строка –  $A \cdot \bar{B}$

2. Объединяем минтермы:  $P = \bar{A}B + A\bar{B}$ .

3. Данное логическое выражение является простым и сокращению не подлежит.

4. Преобразуем формулу с целью упрощения:  $P \equiv \bar{P} \equiv \overline{\bar{A}B + A\bar{B}} \equiv \overline{\bar{A}B} \cdot \overline{A\bar{B}}$

Данная формула имеет следующие элементы:

«НЕ» =  $\bar{A}, \bar{B}$

«И-НЕ» =  $\overline{\bar{A}B}, \overline{A\bar{B}}, P$

1. Выбираем столбцы, в которых  $P=1$ , и стоим для них минтермы.

1 строка –  $\bar{A} \cdot B$

2 строка –  $A \cdot \bar{B}$

2. Объединяем минтермы:  $P = \bar{A}B + A\bar{B}$ .

3. Данное логическое выражение является простым и сокращению не подлежит.

4. Преобразуем формулу с целью упрощения:  $P \equiv \bar{P} \equiv \overline{\bar{A}B + A\bar{B}} \equiv \overline{\bar{A}B} \cdot \overline{A\bar{B}}$

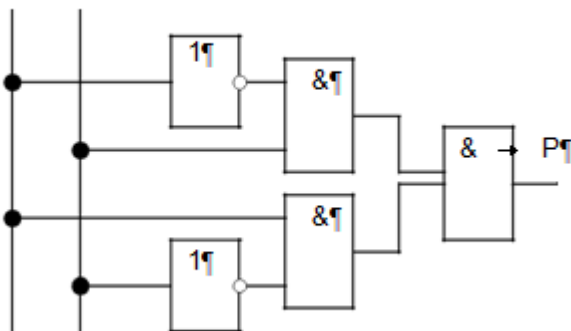
Данная формула имеет следующие элементы:

«НЕ» =  $\bar{A}, \bar{B}$

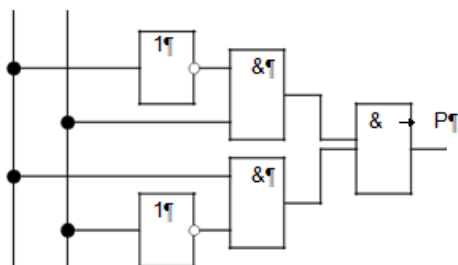
«И-НЕ» =  $\overline{\bar{A}B}, \overline{A\bar{B}}, P$

В результате получим функциональную схему сравнения двоичных чисел в соответствии с данной формулой:

$A \rightarrow B$

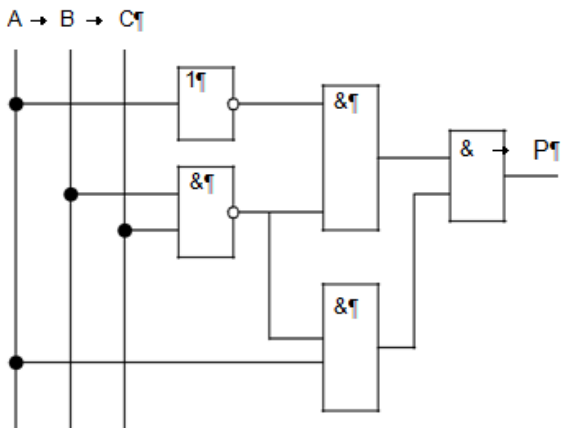


$\bar{A} \rightarrow \bar{B}$



$$P = \overline{\bar{A}B} \cdot \overline{A\bar{B}}$$

**ПРИМЕР 2.** Построить функциональную схему для формулы



**ЗАДАНИЕ 1.** Построить функциональные схемы на одном базисе, реализующих следующие выражения:

1.  $P = \overline{AB} + AC$ ;
2.  $P = AB + \overline{C}$ ;
3.  $P = A + B + C$ ;
4.  $P = AB + BC + AC$ ;
5.  $P = (A + B + C)(\overline{AB + BC}) + AB$ ;
6.  $P = \overline{(A + B)}C$ ;
7.  $P = \overline{AB + BC} + \overline{ABC}$ ;
8.  $P = \overline{ABC} + \overline{ADE}$
9.  $P = \overline{AB + CD + EF}$
10.  $P = \overline{A + BC} + \overline{AD}$

**Форма представления результата:**

Выполненная работа

**Критерии оценки**

Оценка «отлично» выставляется, если выполнены все задания практического занятия, допущены 1-2 недочеты, исправленные по требованию преподавателя.

Оценка «хорошо» выставляется, если работа выполнена в полном объеме, допущены одна ошибка или более двух недочетов при выполнении задания, исправленные по замечанию преподавателя.

Оценка «удовлетворительно» выставляется, если задания выполнены не в полном объеме, допущены 1-2 ошибки при выполнении заданий, но продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

Оценка «неудовлетворительно» выставляется, если выполнено менее половины заданий, не продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

### Тема 1.3 Электронные блоки ЭВМ

#### Практическая работа № 4

#### Моделирование логических элементов в "EWB"

**Цель работы:** произвести вычисление по формуле  $z=ax+by$  на одноадресной модели Малютка.

**Выполнив работу, Вы будете:**

уметь:

производить вычисления на одноадресной модели компьютера Малютка

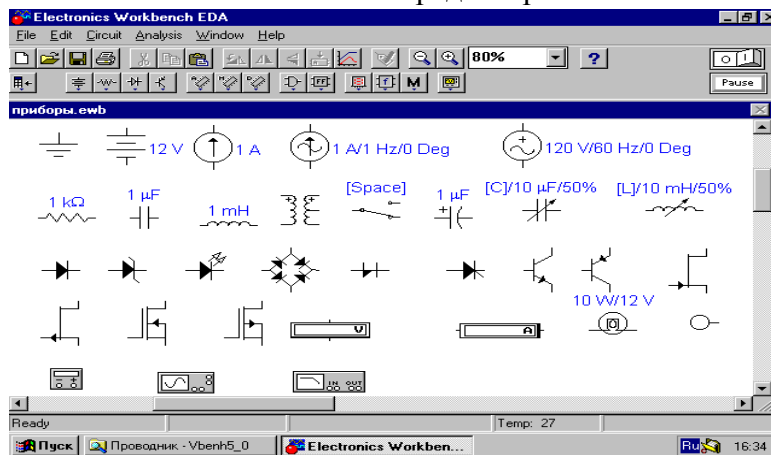
**Материальное обеспечение:** персональный компьютер с лицензионным программным обеспечением (EWB версия 5.12)

**Краткие теоретические сведения**

Знакомство с системой Electronics Workbench 5.12

Запустив интегрированный пакет Electronics Workbench 5.12, вы увидите диалоговое окно и окно редактирования (рис.1). Окно редактирования заполнено некоторыми компонентами. Диалоговое окно Electronics Workbench содержит поле меню, библиотеку компонентов и линейку контрольно-измерительных приборов, расположенных в одном поле. Поле меню аналогичное многим Windows-приложениям. Опции главного меню легко изучить самостоятельно.

Рис.1 Окно редактирования



**Задание 1.** Создать электронную модель однородного участка электрической цепи в электронной лаборатории Electronics Workbench 5.12 (рис.2). Проверить справедливость закона Ома для этой цепи.

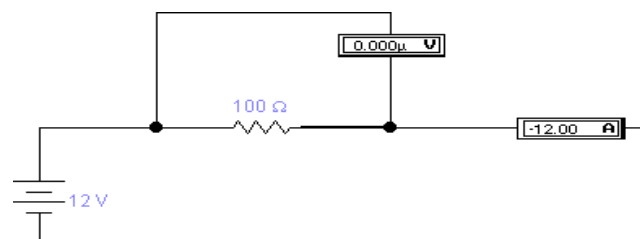


Рис. 2 Электронная модель однородного участка

**Задание 2.** Изучить прибор осциллограф. В режиме В/А осциллографа получить фазовую диаграмму

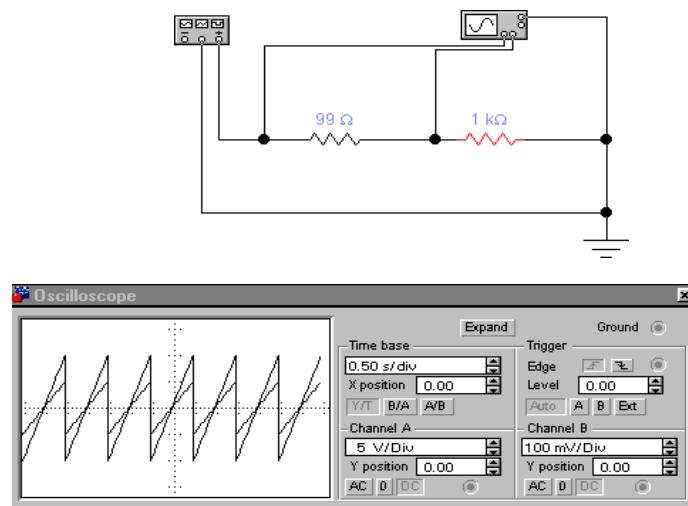


Рис. 3 Прибор осциллограф

**Решение.** Для этого собрать схему, показанную на рис.3.

Воспользуемся функциональным генератором для получения вольтамперной характеристики резисторов, подавая на резистор R1 пилообразное напряжение. Длительность первого полупериода установим равной 99% периода. Напряжение генератора будем подавать на канал А осциллографа с резистора R1. Напряжение, прямо пропорциональное силе тока, будем подавать на канал В осциллографа с дополнительного резистора R2 по схеме характеристики графа.

**Задание 3.** Получить диаграмму в графическом формате.

**Решение.** Сохранение диаграммы в графическом формате и ее редактирование осуществляется с помощью программы Аналитические графики (Analysis Graphs) (рис.4).

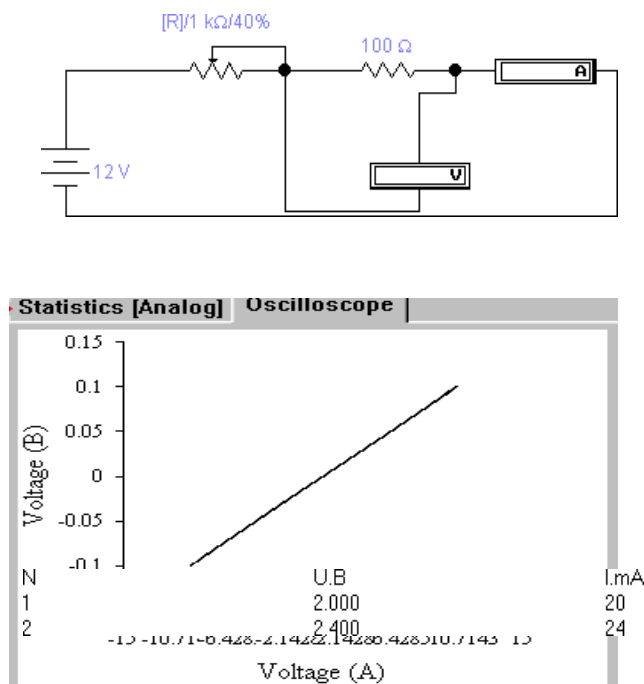


Рис. 4 Диаграмма в графическом формате

Для установки параметров осей, графиков и линий сетки щелчком правой кнопки мыши вызывается контекстное меню, в котором надо выбрать команду Свойства.

**Задание 4.** Копировать любую область окна Electronics Workbench

**Решение.** В Electronics Workbench можно имитировать и ручной режим проведения опыта: увеличивать напряжение на резисторе 100 Ом, уменьшая сопротивление переменного резистора нажатием клавиши R (или любую, назначенную пользователем), согласно определенному декременту. Показания приборов в этом случае просто вводятся с клавиатуры в окно Description. На рис. 5 изображен скопированный в буфер обмена результат такого эксперимента.

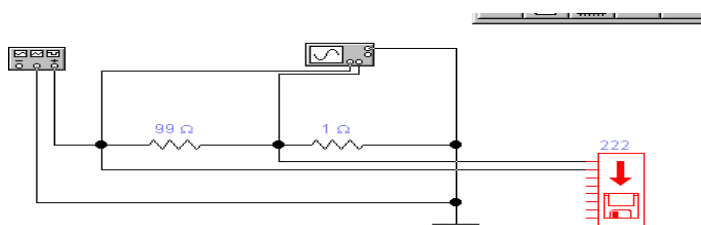


Рис.5. Образец схемы для моделирования

Копирование любой области окна Electronics Workbench 5.12 производится с помощью команды Копировать как точечный рисунок (Copy as Bitmap), просмотр содержимого буфера обмена – с помощью команды Показать буфер обмена (Show Clipboard) в меню Правка (Edit). Скопированный таким образом рисунок также можно вставить в текст Word.

#### **Форма представления результата:**

Выполненная работа

#### **Критерии оценки**

Оценка «отлично» выставляется, если выполнены все задания практического занятия, допущены 1-2 недочеты, исправленные по требованию преподавателя.

Оценка «хорошо» выставляется, если работа выполнена в полном объеме, допущены одна ошибка или более двух недочетов при выполнении задания, исправленные по замечанию преподавателя.

Оценка «удовлетворительно» выставляется, если задания выполнены не в полном объеме, допущены 1-2 ошибки при выполнении заданий, но продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

Оценка «неудовлетворительно» выставляется, если выполнено менее половины заданий, не продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

### Тема 1.3 Электронные блоки ЭВМ Практическая работа №5 Моделирование триггеров

**Цель работы:** научиться моделировать триггеры

**Выполнив работу, Вы будете:**

уметь:

строить схемы синхронных и асинхронных триггеров

исследовать в различных режимах логику работы RS-триггера

**Материальное обеспечение:** персональный компьютер с лицензионным программным обеспечением (программа EWB версия 5.12)

Порядок выполнения работы

**Задание 1.** Построить на элементах 2И-НЕ и 2ИЛИ-НЕ схемы асинхронных RS- триггеров и исследовать логику их работы в статическом режиме. Для этого собрать схемы с использованием пробников и переключателей.

Образцы схем для моделирования приведены на рисунке 1. Исследуемые схемы и таблицы занести в отчет.

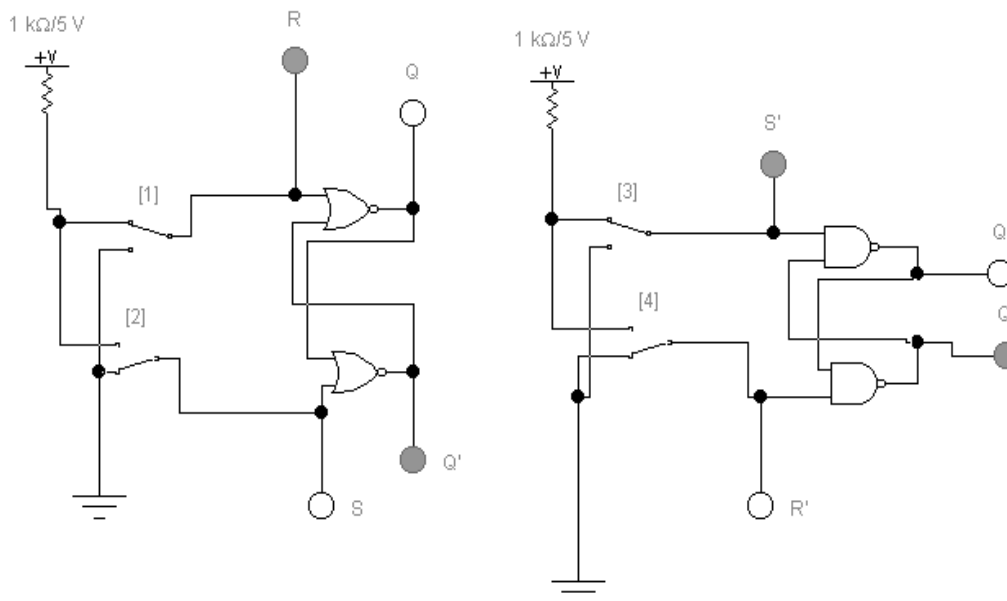


Рис.1 Образцы схем для моделирование

**Задание 2.** Построить на элементах 2И-НЕ и 2-2И-2ИЛИ-НЕ схемы синхронных RS- триггеров и исследовать логику их работы в статическом режиме. Образцы схем для моделирования приведены на рисунке 2 и 3. В качестве элементов 2-2И-2ИЛИ-НЕ использована микросхема 7455, в которой располагается элемент 4-4И-2ИЛИ-НЕ. Исследуемые схемы и таблицы занести в отчет.



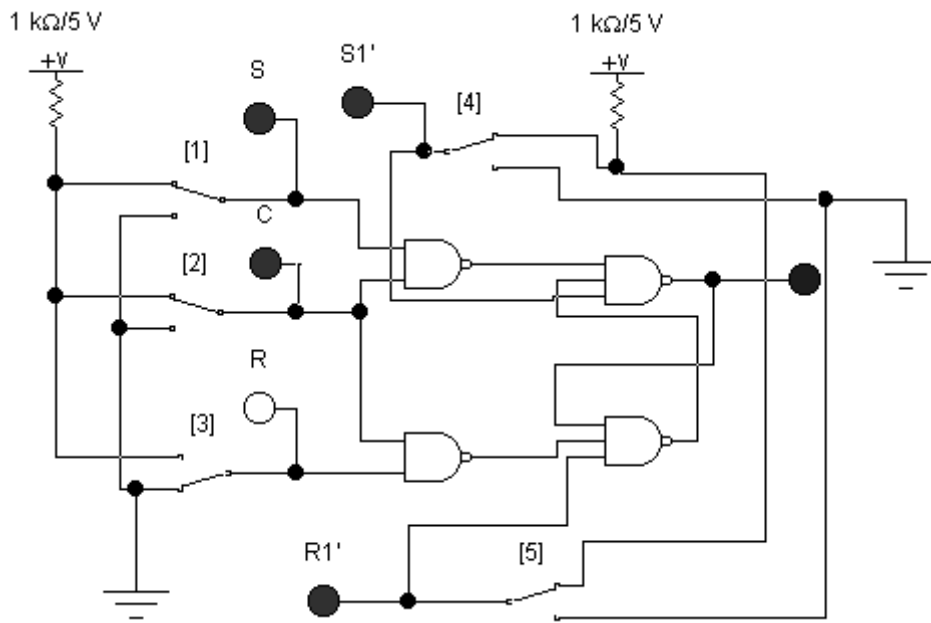


Рис. 2

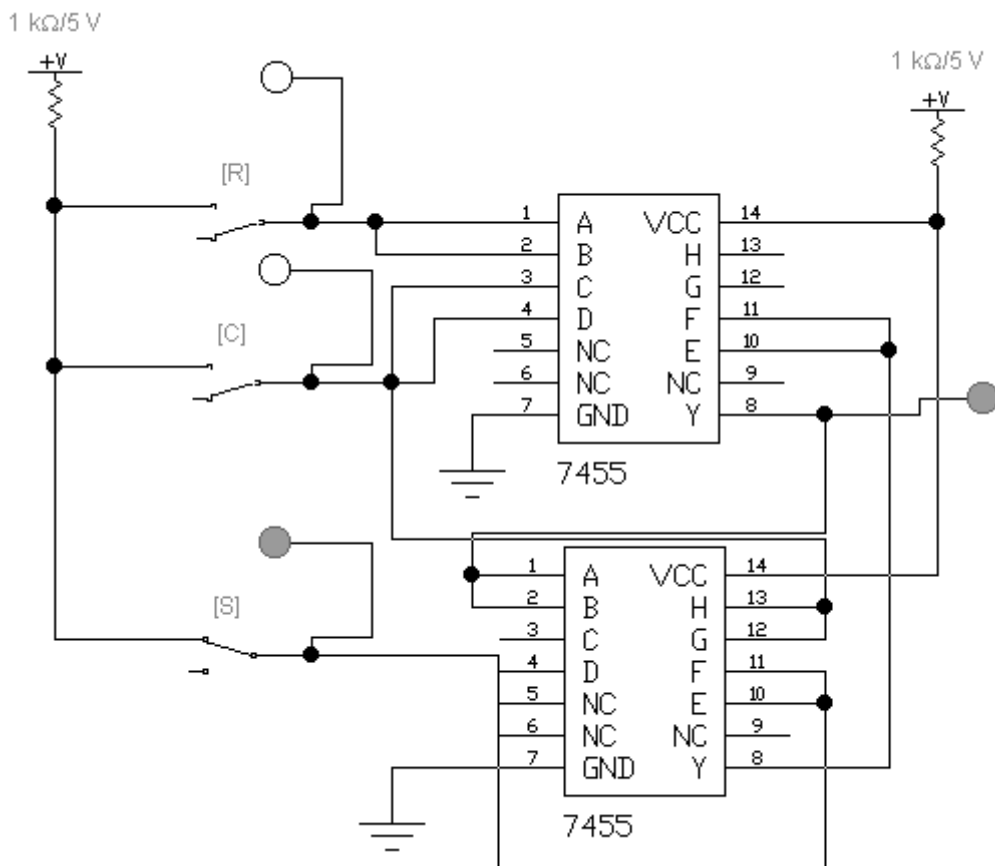


Рис. 3

**Задание 3.** Исследовать в статическом режиме логику работы RS-триггера, который имеется в библиотеке программы. Для этого собрать схему, показанную на рисунке 4.

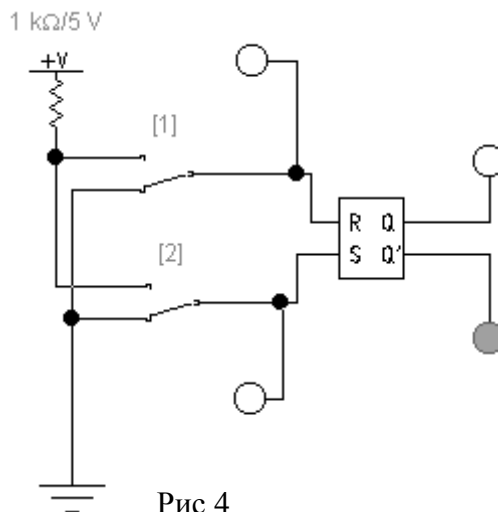


Рис.4

**Задание 4.** Исследовать в статическом режиме логику работы двухтактного RS-триггера. Для этого собрать схему, показанную на рисунке 5.

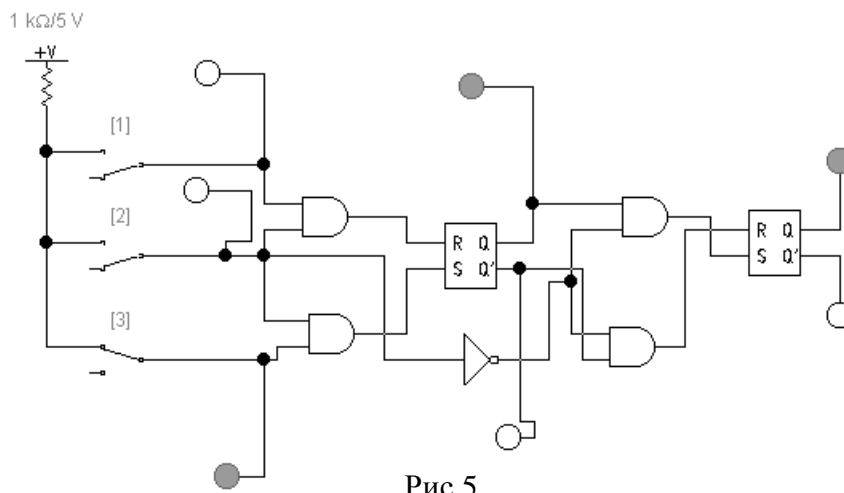


Рис.5

**Задание 5.** Исследовать в статическом режиме логику работы асинхронного D-триггера. Для этого собрать схему, показанную на рисунке 6.

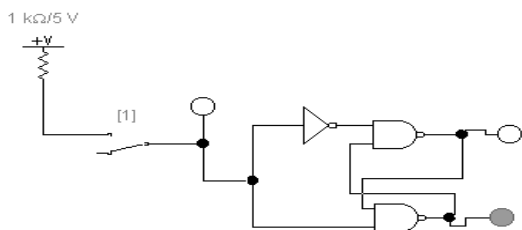


Рис.6

**Задание 6.** Исследовать в динамическом режиме логику работы асинхронного D-триггера. Для этого собрать схему, показанную на рисунке 7. Для визуального наблюдения работы схемы установить частоту генератора 1 Гц. Зарисовать полученную осциллограмму. Исследуемую схему и таблицу занести в отчет.

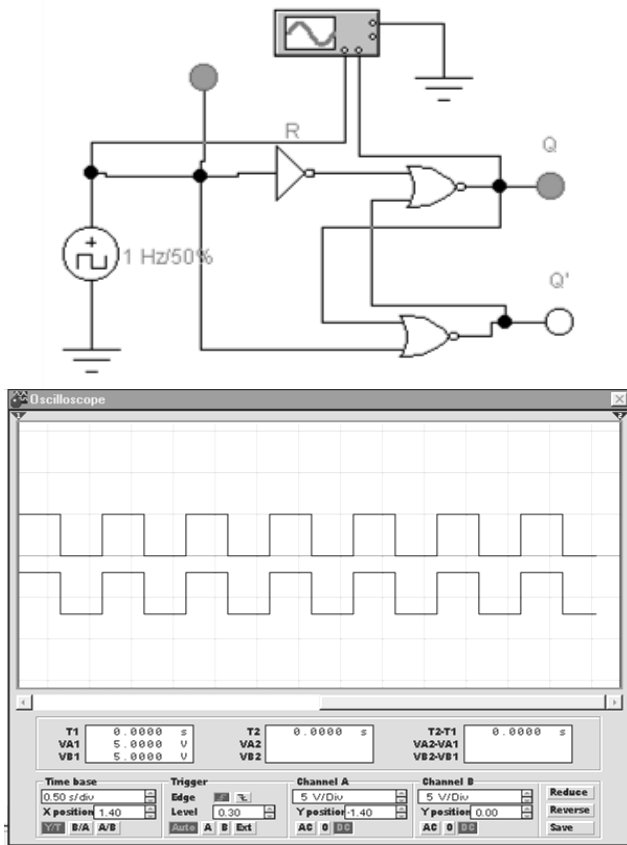


Рис. 7

**Задание 7.** Собрать и исследовать в статическом режиме схему синхронного D- триггера на элементе 2И-2И-2ИЛИ-НЕ, в качестве которого использовать микросхему 7451 с 2-мя элементами 2И-2И-2ИЛИ-НЕ. Схема для исследования показана на рисунке 8. Результаты исследования занести в отчет.

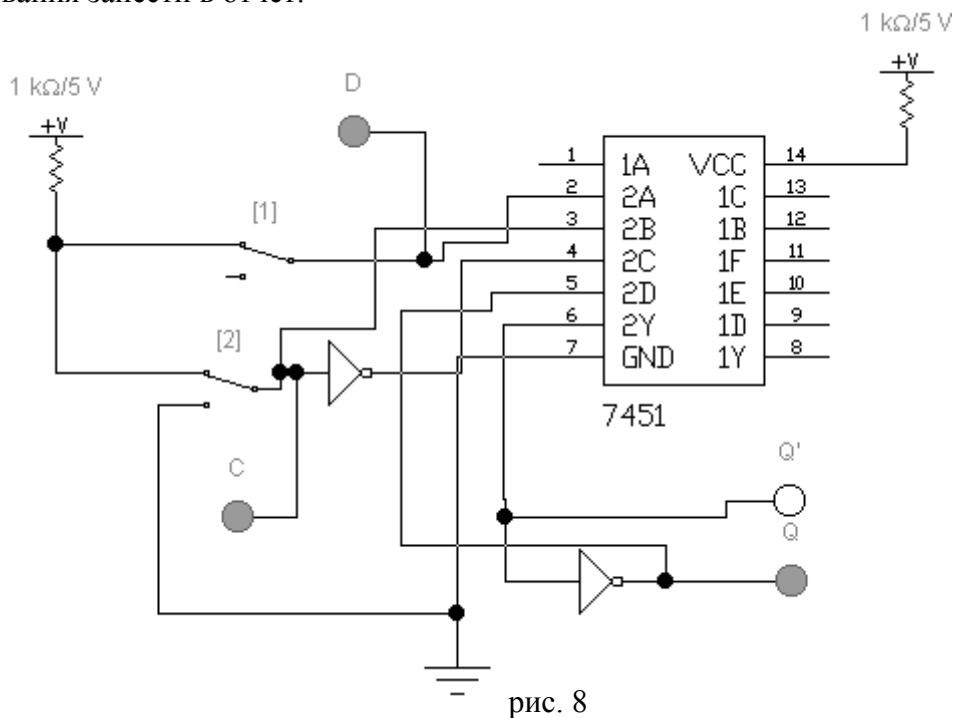


рис. 8

**Задание 8.** Собрать и исследовать микросхему 7474, состоящую из 2-х синхронных D-триггеров. Схема показана на рисунке 9. Результаты исследования занести в отчет.

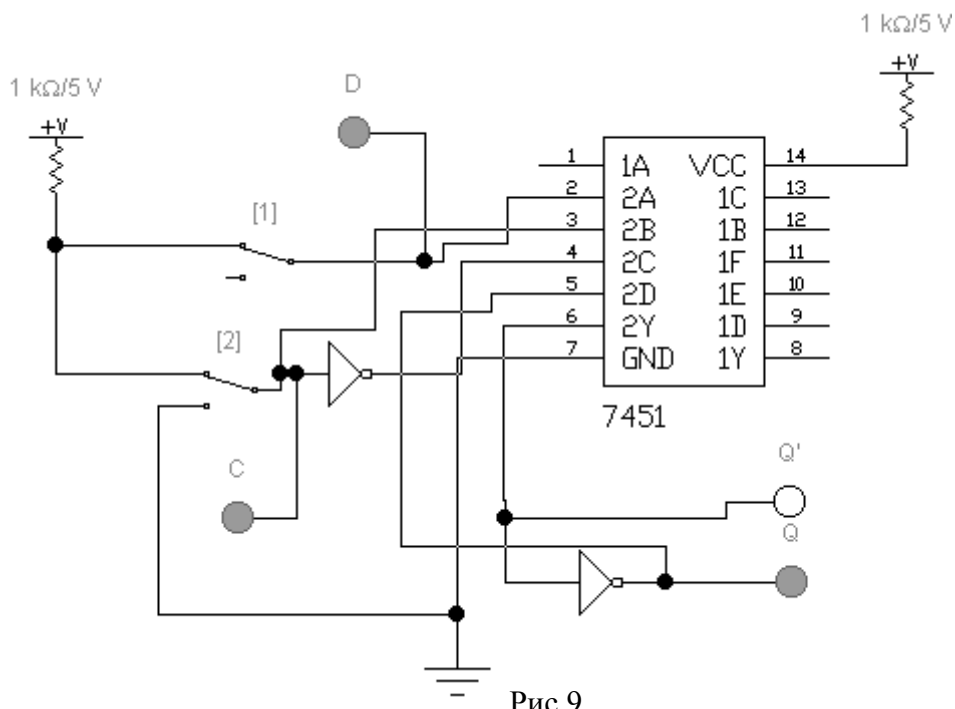


Рис.9

**Задание 9.** Собрать схему и исследовать работу асинхронного T-триггера, построенного на базе синхронного D-триггера в статическом режиме.

Соответствующая схема показана на рисунке 10. В качестве синхронного D-триггера использовать микросхему 7474 с дополнительными асинхронными входами установки и сброса (инверсные входы R и S). Результаты исследования занести в отчет.

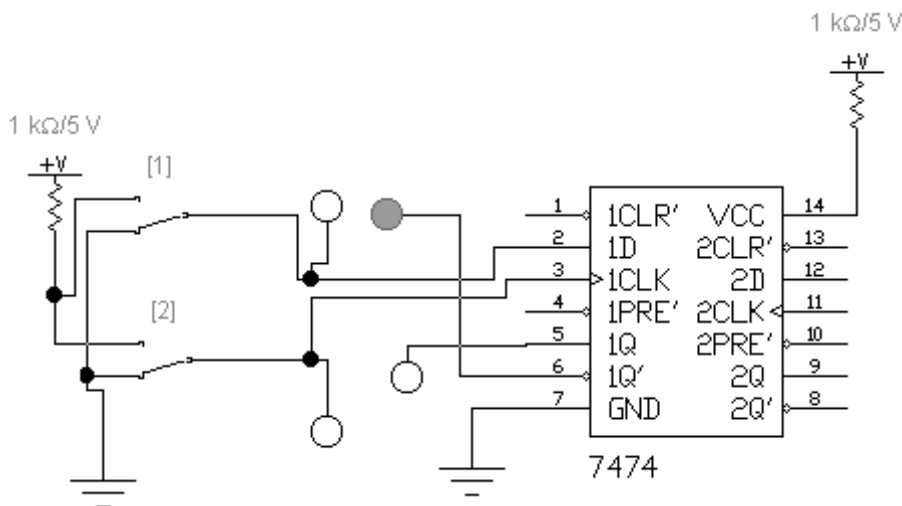


Рис. 10

**Форма представления результата:**

Выполненная работа

**Критерии оценки**

Оценка «отлично» выставляется, если выполнены все задания практического занятия, допущены 1-2 недочеты, исправленные по требованию преподавателя.

Оценка «хорошо» выставляется, если работа выполнена в полном объеме, допущены одна ошибка или более двух недочетов при выполнении задания, исправленные по замечанию преподавателя.

Оценка «удовлетворительно» выставляется, если задания выполнены не в полном объеме, допущены 1-2 ошибки при выполнении заданий, но продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

Оценка «неудовлетворительно» выставляется, если выполнено менее половины заданий, не продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

### Тема 1.3 Электронные блоки ЭВМ

#### Практическая работа № 6. Моделирование регистров в "EWB"

**Цель:** ознакомление с работой регистров с помощью инструментальных средств цифровой части пакета EWB, закрепления теоретического материала, приобретение навыков создания и моделирования цифровых устройств.

**Выполнив работу, Вы будете:**

уметь:

собирать схемы регистров

исследовать регистры

**Материальное обеспечение:** персональный компьютер с лицензионным программным обеспечением (программа EWB 5.12)

**Задание 1.** Параллельная запись кода регистра по входам предыдущей установки.

Собрать схему параллельного регистра на JK-триггерах в пакете EWB 5.12. Заполнить двоичным кодом двоичный генератор слов. Для этого в нулевую ячейку записать все нули, в первую ячейку – код своего номера по списку в группе, во вторую ячейку – № + 3, в четвертую – код “обнуления” всех триггеров регистра. Пример заполнения генератора слов та схема исследования регистра на JK-триггерах приведенные на рис.1. “Обнуление” всех триггеров регистра выполняется на 4-ом такте работы генератора слов (001016).

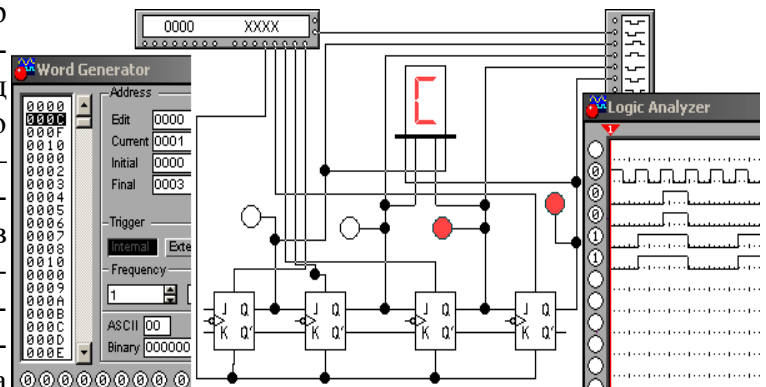


Рис.1. Схема исследования регистра на JK-триггерах

**Задание 2.** Последовательная запись кода в регистр сдвига по информационным входам.

Собрать схему последовательного регистра, которая приведена на рис. 2.

С помощью генератора слов в последовательный регистр записать свой номер по порядку в учебной группе. Например, необходимо записать код номера 1210 = С16. Для этого в нулевую ячейку генератора слов записываются все нули, в первую – вес старшего разряда, то есть 0001, во вторую – 0001, а в третью и четвертую – все нули (рис2). Таким образом в генераторе слов последовательно записан код С16.

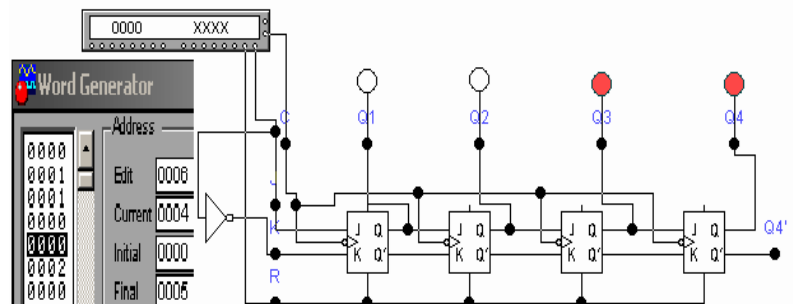


Рис. 2

**Задание 3.** Создать макрос из схемы опыта 2, исследовать и построить 8-разрядный регистр сдвига вправо.

Для создания макроса из схемы опыта 2 необходимо поставить точку на тех выводах макроса, которые необходимы. Выделить точку, нажать правую клавишу мышки, войти к меню: Open – Label та проставить имя. Потом выделить необходимые элементы, нажать на панели инструментов пиктограмму Circuit, потом – Subcircuit т а указать имя, например: RG\_JK. Схема соединения элементов к макросу RG\_JK приведена на рис. 3.

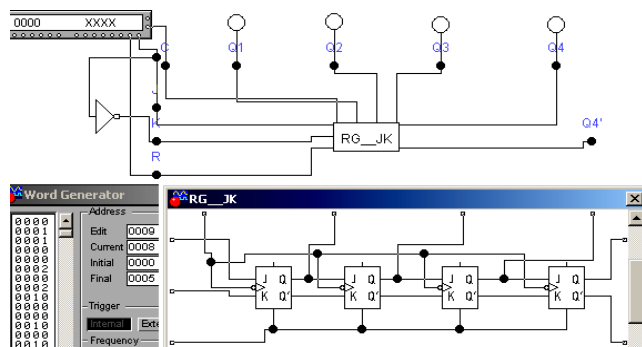


рис. 3

Выберите второй макрос из пиктограммы с именем Favorites но соедините их так, как приведено на рис. 4. Запишите в генератор слов такую последовательность, которая бы позволяла отобразить на светодиодах два одинаковых числа, которые отвечают номеру по порядку в группе.

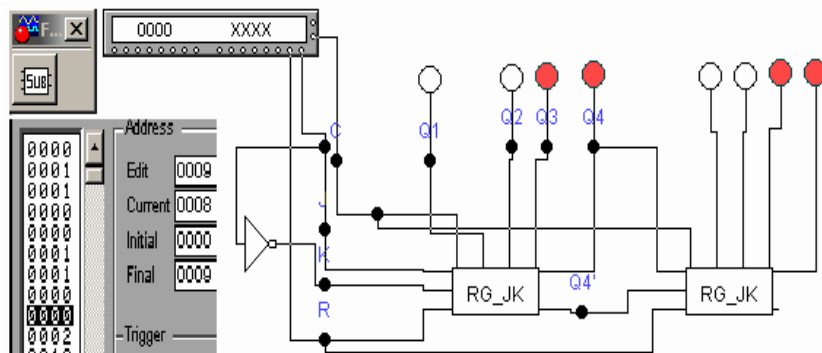


Рис. 4. Схема 8-разрядного регистра

**Задание 4.** Исследовать регистр, согласно варианта, который приведен в табл. .1.

Навести функциональную схему и часовые диаграммы его функционирования

Таблица .1

№ з/п	Серия SN74	Отечественные МС	Функциональное назначение
1	7491	134ИР2	8- разрядный регистр сдвига
2	74164	155ИР8	8- разрядный регистр сдвига с параллельными выходами
3	74165	555ИР9	8- разрядный регистр сдвига с параллельным вводом информации
4	74166	555ИР10	8- разрядный регистр сдвига с синхронным параллельным вводом
5	74173	155ИР15	4- разрядный регистр с тремя состояниями
6	74194	155ИР11	4- разрядный универсальный регистр
7	74195	155ИР12	4- разрядный регистр сдвига с параллельным вводом
8	74198	155ИР13	8- разрядный универсальный регистр сдвига
9	74273	155ИР35	8-розр. регистр с установкой 0
10	74373	155ИР22	8-розр. буферный регистр с 3 состояниями и потенциальным управлением

11	74374	155ИР23	8- разрядный буферный регистр с 3 состояниями и импульсным управлением
12	74377	155ИР27	8- разрядный регистр с разрешением записи
13	74395	533ИР25	4- разрядный параллельный регистр сдвига
14	7491	134ИР2	8- разрядный регистр сдвига

**Форма представления результата:**

Выполненная работа

**Критерии оценки**

Оценка «отлично» выставляется, если выполнены все задания практического занятия, допущены 1-2 недочеты, исправленные по требованию преподавателя.

Оценка «хорошо» выставляется, если работа выполнена в полном объеме, допущены одна ошибка или более двух недочетов при выполнении задания, исправленные по замечанию преподавателя.

Оценка «удовлетворительно» выставляется, если задания выполнены не в полном объеме, допущены 1-2 ошибки при выполнении заданий, но продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

Оценка «неудовлетворительно» выставляется, если выполнено менее половины заданий, не продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

**Тема 1.3 Электронные блоки ЭВМ**

**Практическая работа № 7. Моделирование счетчиков в "EWB"**

**Цель:** изучение структуры и исследование работы суммирующих и вычитающих двоичных счетчиков, а также счетчиков с  $n$  коэффициентом пересчета, отличным от 2 .

**Выполнив работу, Вы будете:**

уметь:

Моделировать различные счетчики.

**Порядок выполнения работы:**

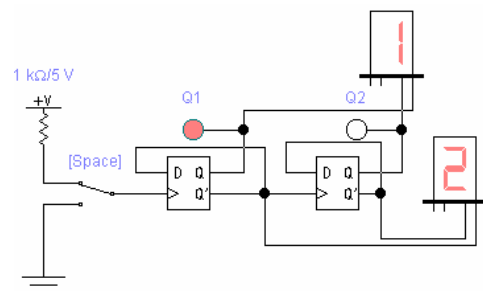
1. Начертить схему электрическую функциональную трехразрядного двоичного суммирующего счетчика (для четных вариантов) на базе D-триггеров или вычитающего счетчика (для нечетных вариантов) на базе JK-триггеров и начертить временные диаграммы, поясняющие его работу.

2. Разработать на базе D-триггеров схему электрическую функциональную счетчика с коэффициентом пересчета, равным девяти (для нечетных вариантов) или десяти (для четных вариантов), и начертить временные диаграммы, поясняющие его работу.

3. Разработать схему трехразрядного двоичного синхронного последовательного (для нечетных вариантов) или синхронного параллельного (для четных вариантов) счетчика.

4. Смоделировать схему электрическую функциональную трехразрядного двоичного счетчика, разработанного в п.п. 1.1 в среде Electronics Workbench. Пример моделирования двухразрядного суммирующего счетчика на базе D-триггеров приведен на рисунке 1.

Рис. 1



5. Включить схему и, подавая на вход схемы тактовые импульсы при помощи ключа, наблюдать за изменением показаний Decoded Seven- Segment Display (семисегментных индикатора с двоично-десятичным дешифратором на входе), подключенных к прямым и инверсным выходам счетчика. Убедиться в правильности работы счетчика. По состояниям логических пробников Q1-Q2, подключенных к прямым выходам триггеров, получить временные диаграммы входных и выходных сигналов, поясняющих работу счетчика.

6. Смоделировать схему электрическую функциональную счетчика, разработанного в п.п. 1.2 в среде Electronics Workbench. Пример моделирования трехразрядного суммирующего счетчика с коэффициентом пересчета, равным пяти на базе D-триггеров приведен на рисунке 2.

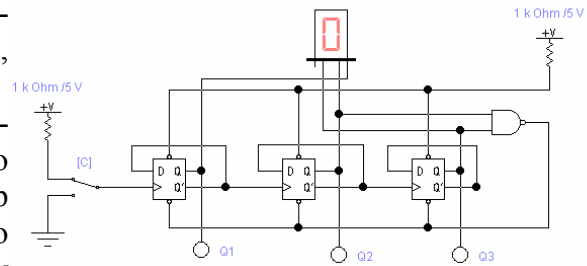


Рис. 2

Включить схему и, подавая на вход схемы тактовые импульсы при помощи ключа, наблюдать за изменением показаний Decoded Seven- Segment Display, подключенного к прямым выходам счетчика. Убедиться в правильности работы счетчика. По состояниям логических пробников Q1- Q3, подключенных к прямым выходам триггеров, получить временные диаграммы входных и выходных сигналов, поясняющих работу счетчика.

Смоделировать схему электрическую функциональную счетчика, разработанного в п.п. 1.3 в среде Electronics Workbench.

Включить схему и, подавая на вход схемы тактовые импульсы с выхода генератора слов Word Generation в режиме Sycle.

Получить временные диаграммы входных и выходных сигналов для смоделированного счетчика на экране логического анализатора Logic

Analyzer и по состояниям логических пробников, подключенных к его прямым выходам.

Обработка экспериментальных данных

Составить отчет о выполнении лабораторной работы. Включить в отчет схемы, полученные при выполнении п.п. 1.1-1.3, а также результаты их моделирования и диаграммы входных и выходных сигналов для каждой из выполненных схем.

Сравнить диаграммы для разработанных теоретически и смоделированных в среде Electronics Workbench схем и сделать выводы.

### Форма представления результата:

Выполненная работа

### Критерии оценки

Оценка «отлично» выставляется, если выполнены все задания практического занятия, допущены 1-2 недочеты, исправленные по требованию преподавателя.

Оценка «хорошо» выставляется, если работа выполнена в полном объеме, допущены одна ошибка или более двух недочетов при выполнении задания, исправленные по замечанию преподавателя.

Оценка «удовлетворительно» выставляется, если задания выполнены не в полном объеме, допущены 1-2 ошибки при выполнении заданий, но продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

Оценка «неудовлетворительно» выставляется, если выполнено менее половины заданий, не продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.



## Тема 2.1 Внутренняя структура вычислительной машины Практическая работа № 8. Определение конфигурации ЭВМ

### Цели:

повторить назначение основных устройств компьютера, их необходимость в данной конфигурации;

повторить назначение программного обеспечения компьютера;

развить умения оформлять результаты работы в виде отчетов, выполненных в различных приложениях офисных технологий.

Выполнив работу, Вы будете:

уметь:

выбирать конфигурацию различных ПК

выбирать оборудование в зависимости от конфигурации

**Материальное обеспечение:** компьютер, виртуальные компоненты и программное обеспечение, выход в интернет

### Теоретическое обоснование.

При выборе компьютера, в первую очередь, нужно определить для каких целей совершается покупка: для работы в офисных программах или же для развлечений. Первая, наиболее часто встречаемая, проблема при покупке компьютера это выбор несбалансированной системы. Вторая немаловажная проблема - влияние рекомендаций консультантов в магазинах компьютерной техники, которые зачастую заинтересованы продать залежавшийся товар.

Таким образом, целью исследований является повышение эффективности выбора комплектующих персонального компьютера с учетом ценового фактора и фактора совместимости элементов.

Для достижения поставленной цели решаются следующие задачи:

- 1) анализ разнообразия представленных комплектующих на рынке и алгоритмов выбора оптимальных элементов;
- 2) выбор и адаптация алгоритма подбора к условиям поставленной проблемы – оптимизации выбора комплектующих персонального компьютера;
- 3) реализация алгоритма выбора на примере интернет-магазина компьютерной техники;
- 4) анализ адекватности алгоритма выбора комплектующих персонального компьютера посредством интернет-магазина компьютерной техники.

Из каких компонентов состоит ПК: [http://assembly-pc.narod.ru/a\\_pc.html](http://assembly-pc.narod.ru/a_pc.html)

Процесс сборки ПК: [http://assembly-pc.narod.ru/a\\_pc.html](http://assembly-pc.narod.ru/a_pc.html)

### Порядок выполнения работы:

Данная работа направлена на закрепление умений правильно выбирать конфигурацию компьютера для выполнения различного вида задач (как учебного, так и личного плана). Работа проводится с использованием данных интернет магазинов РЕТ, Салон 2116, DNS-Курск. Результаты лабораторной работы оформить: а) в текстовом редакторе MS WORD (имя файла "Фамилия.doc"), б) в виде презентации (имя файла "Фамилия.pps), где представить информацию по каждому из этапов:

1. Открыть страницу <http://assembly-pc.narod.ru/index.html>

2. Изучить теоретические сведения по разделам «Компоненты ПК», «Процесс сборки ПК»

3. Выполнить задание 1

4. Выполнить задание 2

5. Выполнить задание 3

6. Пройти тест. После правильного выполнения заданий и прохождения теста приступить к подбору компонентов для комплектования системного блока предварительно получив у преподавателя вид заданного ПК

8 произвести выбор компонентов согласно этапам.

I Этап - выбор конфигурации;

II Этап - подробный состав оборудования, включая периферийные устройства;

III Этап - детальные рекомендации по использованию данного ПК

**I Этап. Конфигурация.** По указанию преподавателя нужно выбрать конфигурацию компьютера, которую затем необходимо будет "собрать" (подобрать подходящее оборудование с использованием интернет магазина компьютерной техники)

Различные конфигурации или как планируется использовать компьютер?

1. **Офисный** (Набор текстов, выполнение математических (простых) расчетов, оформление отчетов и докладов, составление презентаций, работа в Интернете).
2. **Фото- и видеообработка** (Получение информации с внешних устройств (сканер, веб-камера, микрофон), обработка информации (работа с графической, звуковой и видео-информацией), вывод информации на внешние устройства (принтер, цифровая камера), размещение информации в Интернете)
3. **Игровой компьютер** (поддержка сложной трехмерной графики, возможность хранить игры на жестком диске в виртуальных образах).
4. **Домашний** (Многозадачность, возможность решения на компьютере различных учебных и личных задач, быстрый ввод и вывод различной информации с помощью внешних устройств, работа в Интернете)
5. **Школьный** (Использование компьютера учениками на уроках информатики и других предметах).
6. **Рабочее место учителя** (Использование компьютера учителем для подготовки и проведения уроков по различным предметам)
7. **Сервер** (Компьютер, предоставляющий свои ресурсы пользователям сети)

Выбрав конфигурацию для сборки, переходим к оборудованию. Выбор оборудования должен соответствовать конфигурации (т.е. для решения простых задач можно использовать более "слабый" компьютер, чем для решения сложных).

**II Этап. Оборудование** (количество неограниченно, т.е. можно использовать более одного компонента). В отчет необходимо записать тип и код выбранного компонента (с указанием интернет-магазина)

**Важно!** Оборудование нужно выбирать в зависимости от конфигурации. Не нужно выбирать все предложенное.

В расчет включается стоимость комплектующих. В подведении итогов учитывается соотношение цена/качество.

### План выполнения 2 этапа практической работы

1. Разбивка на команды по 3 человека (командир, технический эксперт, экономический эксперт)
2. Выдача задания преподавателем с указанием предназначения ПК.
3. Вход на страницу интернет-магазина.
4. Выбор комплектующих с учетом цены.
5. Проверка на совместимость подобранного оборудования.
6. Подготовка результатов проделанной работы.

**III Этап. Отчет** Детальные рекомендации по использованию данного ПК

Составить отчет


В отчете должны отразить:

этапы выбора конфигурации компьютера;

этапы выбора состава оборудования, включая периферийные устройства;

составить таблицу следующего вида

<i>№ п/п</i>	<i>Изображение компонента</i>	<i>Наименование компонента</i>	<i>Цена в руб.</i>
------------------	-----------------------------------	--------------------------------	--------------------

1.		CPU AMD A10-6800K (AD680KWOA44HL) 4.1 ГГц/4core/SVGA RADEON HD 8670D/ 4 Мб/100 Вт/5 ГТ/с Socket FM2	<b>5 550</b>
	Итого		

детальные рекомендации по использованию данного ПК

(для какой категории пользователей целесообразно использовать «виртуально» собранный компьютер).

Оформленный отчет (Шрифт Verdana, 12 пт, одинарный интервал, выравнивание по ширине, все поля по 2 см) для MS WORD или в произвольной форме в виде презентации разместить в указанной преподавателем папке.

**Форма представления результата:**

Выполненная работа

**Критерии оценки**

Оценка «отлично» выставляется, если выполнены все задания практического занятия, допущены 1-2 недочеты, исправленные по требованию преподавателя.

Оценка «хорошо» выставляется, если работа выполнена в полном объеме, допущены одна ошибка или более двух недочетов при выполнении задания, исправленные по замечанию преподавателя.

Оценка «удовлетворительно» выставляется, если задания выполнены не в полном объеме, допущены 1-2 ошибки при выполнении заданий, но продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

Оценка «неудовлетворительно» выставляется, если выполнено менее половины заданий, не продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

**Тема 2.2 Арифметико-логическое устройство (АЛУ)**

**Практическая работа № 9 Внутренние интерфейсы системной платы**

**Цель:**

Познакомиться с основными элементами системной платы

**Выполнив работу, Вы будете:**

*уметь:*

определять внутренние интерфейсы системной платы

**Материальное обеспечение:**

компьютер с лицензионным программным обеспечением (пакет прикладных программ «Microsoft Office»: табличный процессор Microsoft Excel, текстовый процессор Microsoft Word, браузер Microsoft Internet Explorer), мультимедиа проектор

**Задание:**

1. Опишите подробно интерфейсы слотов расширения (рис1)

2.. При описании укажите характеристику каждого элемента пользуясь средствами сети Интернет

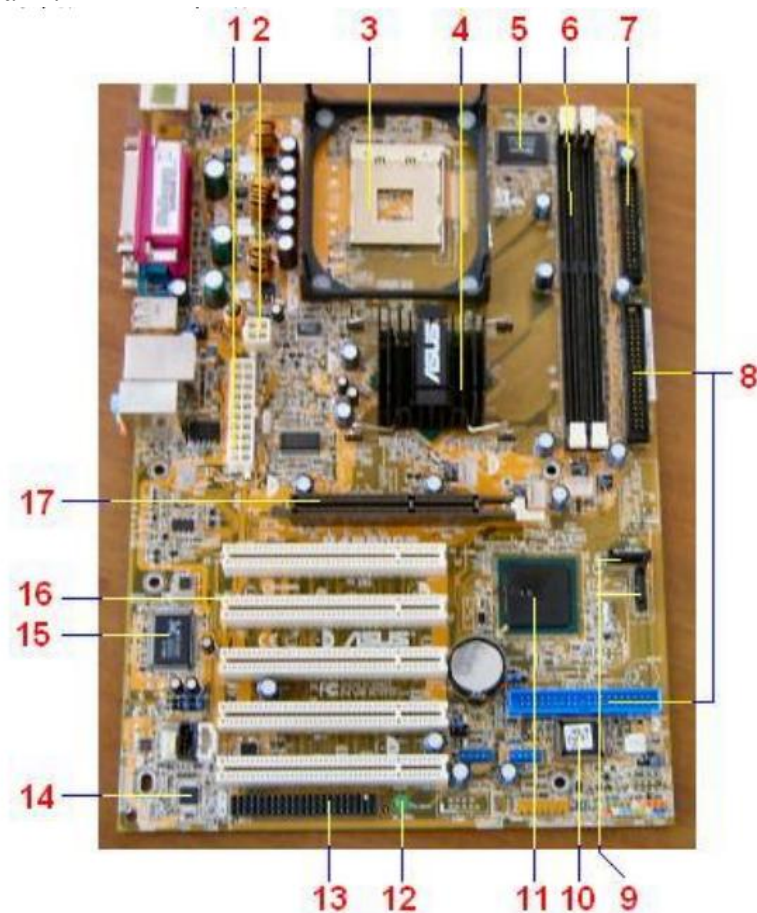
Внимательно рассмотрите представленную материнскую плату на рисунке ниже. На ее примере выделите основные компоненты, а также их назначение.

**Порядок выполнения работы:**

Пример.

Нумерация компонентов:

1. ATX power connector. 20- контактный коннектор для подключения питания материнской платы.



### Форма предоставления результата

Выполненная работа

### Критерии оценки

Оценка «отлично» выставляется, если выполнены все задания практического занятия, допущены 1-2 недочеты, исправленные по требованию преподавателя.

Оценка «хорошо» выставляется, если работа выполнена в полном объеме, допущены одна ошибка или более двух недочетов при выполнении задания, исправленные по замечанию преподавателя.

Оценка «удовлетворительно» выставляется, если задания выполнены не в полном объеме, допущены 1-2 ошибки при выполнении заданий, но продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

Оценка «неудовлетворительно» выставляется, если выполнено менее половины заданий, не продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

## Тема 2.3 Устройство управления

### Практическая работа № 10 Интерфейсы периферийных устройств IDE и SATA. Параллельные и последовательные порты и их особенности работы.

**Цель:** изучить интерфейсы подключения периферийных устройств к ПК

**Выполнив работу, Вы будете:**

уметь:

подключать жесткий диск и CD ROM к системной плате

**Материальное обеспечение:**

Компьютер с лицензионным программным обеспечением и выходом в интернет. системная плата, жесткий диск и CD ROM

### Краткие теоретические сведения:

Периферийные шины используются в основном для внешних запоминающих устройств.

Интерфейс IDE (Integrated Drive Electronics) – интерфейс устройств со встроенным контроллером. Поддерживает несколько способов обмена.

Первый способ производит обмен данными через регистры процессора под его непосредственным управлением. Следствием этого является высокая загрузка процессора при операциях ввода/вывода.

Вторым способом является использование режима прямого доступа к памяти, при котором контроллер интерфейса IDE и контроллер прямого доступа к памяти материнской платы пересылают данные между диском и оперативной памятью, не загружая центральный процессор.

В целях развития возможностей интерфейса IDE была предложена его расширенная спецификация EIDE (синонимы ATA, ATA-2). Она поддерживает накопители емкостью свыше 504 Мбайт, поддерживает несколько накопителей IDE и позволяет подключать к одному контроллеру до четырех устройств, а также поддерживает периферийные устройства, отличные от жестких дисков. Расширение спецификации IDE для поддержки иных типов накопителей с интерфейсом IDE называют также ATAPI.



Интерфейс SCSI (Small Computer System Interface) - является стандартным интерфейсом для подключения приводов компакт-дисков, звуковых плат и внешних устройств массовой памяти. Спецификацией SCSI предусматривается параллельная передача данных по 8, 16 или 32 линиям данных. Структура SCSI, по существу, является магистральной, хотя устройства включаются в нее по принципу последовательной цепочки. Каждое SCSI-устройство имеет два разъема – один входной, а другой выходной. Все устройства объединяются в последовательную цепочку, один конец которой подключается к контроллеру интерфейса. Все устройства работают независимо и могут обмениваться данными как с компьютером, так и друг с другом. К шине SCSI можно подключить до 8 устройств, включая основной контроллер SCSI (хост-



адаптер). Контроллер SCSI является, по сути, самостоятельным процессором и имеет свою собственную BIOS. К шине Wide SCSI подключается до 15 устройств.



ACPI (Advanced Configuration Power Interface – расширенный интерфейс конфигурирования и питания) – интерфейс, представляющий собой единую систему управления питанием для всех компонентов компьютера.

#### Задания:

1. Найти информацию по средующим шинам и интерфейсам: SPI, ISA, EISA, AGP, PCI-E, PCMCIA, SATA, RS-32, RS-485, LPT, IEEE 1394, I<sup>2</sup>C, IrDA, RJ45. Найти их области применения и характеристики.
2. Полученные данные занести в таблицу в рабочую тетрадь:

№ п/п	Название	Макс. Скорость	Макс. Длина линии связи	Типы сигналов, уровни напряжений	Метод передачи	Топология	разъем	Применение

3. Подключить жесткий диск к системной плате.
2. Подключить CD-ROM к системной плате.
3. Дать сравнительную характеристику периферийных устройств целевого компьютера. Определить их достоинства и недостатки.

#### Форма предоставления результата

Выполненная работа

#### Критерии оценки

Оценка «отлично» выставляется, если выполнены все задания практического занятия, допущены 1-2 недочеты, исправленные по требованию преподавателя.

Оценка «хорошо» выставляется, если работа выполнена в полном объеме, допущены одна ошибка или более двух недочетов при выполнении задания, исправленные по замечанию преподавателя.

Оценка «удовлетворительно» выставляется, если задания выполнены не в полном объеме, допущены 1-2 ошибки при выполнении заданий, но продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

Оценка «неудовлетворительно» выставляется, если выполнено менее половины заданий, не продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

## **Тема 2.4 Системная память**

### **Практическая работа № 11. Изучение работы системной памяти**

**Цель:** Знакомство с системной памятью компьютера

**Выполнив работу, Вы будете:**

уметь:

определять текущие значения всех статистических параметров памяти.

определять объемы памяти, используемые процессами

**Материальное обеспечение:** персональный компьютер с лицензионным программным обеспечением

**Порядок выполнения работы:**

Выполните задания 1-7

#### **Задание 1.**

С помощью *Диспетчера задач* определите текущие значения всех статистических параметров памяти. Запустите до 10-ти приложений и определите узкое место в системе (ОЗУ или ЦП) путем анализа графиков Хронология использования памяти и Хронология загрузки ЦП. Запишите новые значения статистических параметров памяти. Закройте открытые приложения и запишите новые значения статистических параметров памяти, сделайте выводы. Какого значения параметра Пик? Сравните с прежним его значением и сделайте выводы.

#### **Задание 2.**

Запустите приложения Блокнот, MS Word, MS Excel. С помощью *Диспетчера задач* определите объемы памяти, используемые процессами: физическую память, пиковое использование памяти, виртуальную память, выгружаемый и невыгружаемый пулы. Определите, как изменяются эти параметры при изменении активности приложений.

#### **Задание 3.**

Изучите справочную информацию о параметрах запуска утилиты TaskList. Получите с помощью утилиты информацию об используемой оперативной памяти каждым процессом системы. Запустите приложения MS Word и MS Excel. Получите с помощью утилиты TaskList информацию о PID их образов и список всех модулей, загруженных в оперативную память и используемых этими процессами. Определите работающие службы.

#### **Задание 4.**

С помощью приложения *Сведения о системе* определите: полный объем физической памяти в компьютере, общий объем виртуальной памяти, доступной (свободной) в данный момент времени виртуальной памяти. Просмотрите сведения об использовании физической памяти аппаратными компонентами компьютера; определите диапазон адресов памяти, используемый каждым из них. Запустите несколько приложений и с помощью приложения *Сведения о системе* определите используемый ими объем ОП. То же самое проделайте для выгружаемых модулей и служб.

#### **Задание 5.**

Определите объем оперативной памяти компьютера и рекомендуемый объем файла подкачки. Проведите дефрагментацию жесткого диска, на который предполагается поместить файл подкачки, установите его желаемое значение (Как определяется это значение?) и перезагрузите компьютер.

#### **Задание 6.**

Создайте два журнала счетчиков (бинарного и текстового форматов) и внесите в них счетчики, позволяющие оптимизировать виртуальную память (память \ доступно байт, память \ обмен страниц в сек, файл подкачки \ % использования) и проведите наблюдение за ситуациями, порождающими недостаток памяти. Запустите журналы счетчиков и некоторое время наблюдайте за системой. Результаты выведите в таблицу (в Excel) и на диаграммы *Системного*

монитора. Выберите другие счетчики, упомянутые в третьем разделе. Выполните анализ полученных результатов и дайте рекомендации по улучшению конфигурации ПК.

#### **Задание 7.**

Найдите на диске (дисках) файл подкачки и установите его размер. С помощью счетчиков файл подкачки \ % использования, файл подкачки \ % использования (пик) определите оптимальное значение размера файла подкачки и установите его.

#### **Форма предоставления результата**

Выполненная работа

#### **Критерии оценки**

Оценка «отлично» выставляется, если выполнены все задания практического занятия, допущены 1-2 недочеты, исправленные по требованию преподавателя.

Оценка «хорошо» выставляется, если работа выполнена в полном объеме, допущены одна ошибка или более двух недочетов при выполнении задания, исправленные по замечанию преподавателя.

Оценка «удовлетворительно» выставляется, если задания выполнены не в полном объеме, допущены 1-2 ошибки при выполнении заданий, но продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

Оценка «неудовлетворительно» выставляется, если выполнено менее половины заданий, не продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

### **Тема 2.5 Основные команды процессора, использование прерываний, программы-отладчики**

#### **Практическая работа № 12 Компиляция и запуск программ в машинных кодах**

**Цель:** Знакомство с макрокомандами отладчика учебной модели, процедурами набора, выполнения и трассировки фрагментов ассемблерного кода.

#### **Выполнив работу, Вы будете:**

уметь:

производить вычисления в машинных кодах на учебной модели компьютера Малютка

**Материальное обеспечение:** персональный компьютер с лицензионным программным обеспечением (учебная модель компьютера малютка, язык программирования Ассемблер)

#### **Порядок выполнения работы:**

1. Наберите предложенные программы
2. Откомпилируйте программы и запустите на выполнение
3. Ознакомьтесь с теоретическими сведениями
4. Ответьте на вопросы преподавателя

#### **Краткие теоретические сведения:**

Как и любая реальная машина, каждая учебная модель ЭВМ тоже имеет свои особенности. Малютка, разумеется, не является исключением из правила. Будучи одноадресной ЭВМ, она во всех своих операциях активно использует особый регистр – сумматор. Другой особенностью модели является то, что она все обрабатываемые данные хранит только в ОЗУ, и адреса ячеек непосредственно входят в команду. Последнее возможно только потому, что память Малютки невелика – всего 256 ячеек и, следовательно, двух шестнадцатеричных цифр вполне достаточно для представления полного адреса.

Любая команда ЭВМ Малютка состоит из 12-ти двоичных разрядов. Таким образом, для записи машинной инструкции требуется всего 3 шестнадцатеричных цифры. Первая обозначает код операции, а две последующие – адрес используемой ячейки ОЗУ. Примеры некоторых инструкций показаны в табл.1

**Таблица 1. Примеры некоторых инструкций ЭВМ Малютка**

<b>од</b>	<b>Мнемоника</b>	<b>Описание</b>	<b>Пример</b>
NN	LDA (NN)	Содержимое ячейки памяти с номером NN ==>СМ	0 09 – содержимое ячейки 09 считывается в СМ



NN	STA (NN)	CM ==> ячейку памяти с номером NN	1 0D – сохранить содержимое сумматора в ячейку 0D
NN	ADD (NN)	CM+ячейка NN ==>CM	A 0D – прибавить к CM целое число из ячейки 0D
NN	MULT (NN)	CM+ячейка NN ==>CM	B 0A – CM умножить на целое число из ячейки 0A
00	HLT	Останов	

В этой таблице отчетливо видно, что для осуществления вычислений один операнд необходимо предварительно извлечь в сумматор, и только потом между ними и вторым операндом из ОЗУ можно выполнить требуемое действие; наконец, результат операции, полученный в сумматоре, при необходимости может быть сохранен в памяти отдельной инструкцией.

Рассмотрим программную реализацию Малютки. Малютка представляет собой реализацию языка Ассемблер. Ассемблер допускает использование буквенных обозначений – так называемых идентификаторов. Все элементы команды в таком языке являются идентификаторами. В частности, вместо числового кода операции используется его мнемонический эквивалент: для инструкции считывания из ОЗУ вместо цифры 1 пишется **lda**, а вместо кода операции сложения **A** – **add**. Но код операции далеко не главное. Гораздо более существенным является замена конкретных адресов ОЗУ идентификаторами. Дело в том, что при таком подходе из программы полностью исчезает «привязка» к конкретным адресам памяти и ее становится гораздо легче изменять. Но как же тогда процессор сможет исполнить такую текстовую программу, если он понимает единственный язык – язык цифровых кодов? Текст на Ассемблере обрабатывается специальной системной программой-транслятором, которая преобразует этот текст в привычную процессору числовую форму. Подчеркнем, что сделать подобный перевод достаточно просто, поскольку каждой строке на Ассемблере соответствует одна машинная команда; фактически достаточно произвести замену определенных сочетаний символов соответствующими двоичными кодами.

**Таблица 2.** Программа вычислений по заданной формуле на одноадресной ЭВМ

Метка	Мнемоника	Адрес	Код	Действие	Комментарий
	lda (a)	1	009	(9) ==> CM	Считать a из ОЗУ
	mult (x)	2	B0A	CM * (A) ==> CM	a*x
	sta (z)	3	10D	CM ==> (D)	Сохранить в z
	lda (b)	4	00B	(B) ==> CM	Считать b из ОЗУ
	mult (y)	5	B0C	CM * (C) ==> CM	b*y
	add (z)	6	A0D	CM * (D) ==> CM	+ ax
	sta (z)	7	10D	CM ==> (D)	Сохранить в z
	hlt	8	F00	Стоп	
a:	dw 2	9	002	2	a
x:	dw 3	A	003	3	x
b:	dw 4	B	004	4	b
y:	dw 5	C	005	5	y
z:	dw 0	D			z=ax+by

Дополнительно поясним, что шестнадцатеричной цифрой (старшая цифра всех используемых в задаче адресов ОЗУ равна нулю, поэтому она не указана) в скобках в столбце «Действие» обозначено содержимое соответствующих ячеек Малютки. При этом конкретные числовые значения ячеек и коды команд являются результатом трансляции текстовой программы. Для их получения уже требуется приступить к работе с программным обеспечением Малютки.

Запустите файл TMA1 из папки malutka, которая находится на диске C:. После запуска программы вы попадете в обыкновенный текстовый редактор. Наберите текст программы:

```

lda (a)
mult (x)
sta (z)
lda (b)
mult (y)
add (z)
sta (z)
hlt

```

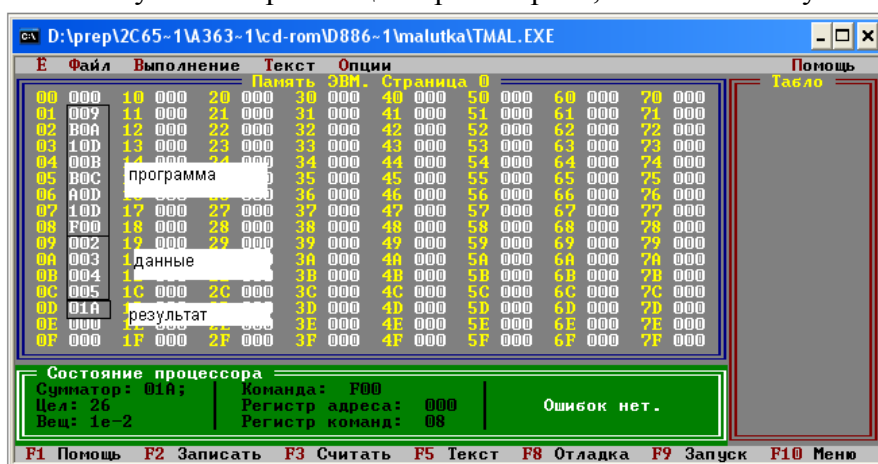
```

a: dw 2
x: dw 3
b: dw 4
y: dw 5
z: dw 0

```

Запустите программу на исполнение (перед этим она автоматически будет оттранслирована), нажав клавишу <F9>.

Результаты трансляции просмотрите, нажав клавишу <F5>.



Итак, если в формулу  $z=ax+by$  подставить значения  $a=2$ ,  $b=3$ ,  $x=4$ ,  $y=5$ , то получится ответ  $1A_{16}$  или  $26_{10}$

### Произведите вычисления по формулам на одноадресной модели Малютка:

#### Вариант 1

1.  $a+b=c$
2.  $a+b+c=y$
3.  $ax+b=y$
4.  $a(x+b)(x+d)=z$
5.  $a(x+b)+c=y$

#### Вариант 2

1.  $a+b=c$
2.  $a+b+c=y$
3.  $ax+b=y$
4.  $ax+bx=z$
5.  $ab+bx+cy=z$

#### Вариант 3

1.  $a+b=c$
2.  $a+b+c=y$
3.  $ax+b=y$
4.  $ax+b(x+d)=z$

5.  $ax^2+bx^2+c=y$
- Вариант 4**
1.  $a+b=c$
  2.  $a+b+c=y$
  3.  $ax+b=y$
  4.  $a+x+b(y+d)=z$
  5.  $ax^3+bx+c=y$
- Вариант 5**
1.  $a+b=c$
  2.  $a+b+c=y$
  3.  $ax+b=y$
  4.  $ax+bx^3=z$
  5.  $ax+b(x+c)=y$
- Вариант 6**
1.  $a+b=c$
  2.  $a+b+c=y$
  3.  $ax+b=y$
  4.  $a(x+b)=z$
  5.  $a^2+b(x+c)=y$
- Вариант 7**
1.  $a+b=c$
  2.  $a+b+c=y$
  3.  $ax+b=y$
  4.  $a(x+d)+c=z$
  5.  $ax+bx^3+c=y$
- Вариант 8**
1.  $a+b=c$
  2.  $a+b+c=y$
  3.  $ax+b=y$
  4.  $a(x+b)(x+d)=z$
  5.  $ax+by+c^2=d$
- Вариант 9**
1.  $a+b=c$
  2.  $a+b+c=y$
  3.  $ax+b=y$
  4.  $ax+bx=z$
  5.  $ax+b(x+c)=y$
- Вариант 10**
1.  $a+b=c$
  2.  $a+b+c=y$
  3.  $ax+b=y$
  4.  $ax+bx^2=z$
  5.  $x+b+c(a+d)=y$

**Форма представления результата:**

Выполненная работа

**Критерии оценки**

Оценка «отлично» выставляется, если выполнены все задания практического занятия, допущены 1-2 недочеты, исправленные по требованию преподавателя.

Оценка «хорошо» выставляется, если работа выполнена в полном объеме, допущены одна ошибка или более двух недочетов при выполнении задания, исправленные по замечанию преподавателя.

Оценка «удовлетворительно» выставляется, если задания выполнены не в полном объеме, допущены 1-2 ошибки при выполнении заданий, но продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.

Оценка «неудовлетворительно» выставляется, если выполнено менее половины заданий, не продемонстрированы умения, достаточные для дальнейшего усвоения программного материала.