



Профессиональное образовательное частное учреждение среднего профессионального образования

**«Высший юридический колледж:
экономика, финансы, служба безопасности»**

Пушкинская ул., д. 268, 426008, г. Ижевск. Тел.: (3412) 32-02-32. Тел./факс: 43-62-22. E-mail: mveu@mveu.ru, mveu.ru

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ

к выполнению практических работ

при изучении учебной дисциплины

ОП.02. Архитектура компьютерных систем

для специальности

09.02.03 Программирование в компьютерных системах

Ижевск, 2020 г.

Практическая работа – небольшой научный отчет, обобщающий проведенную учащимся работу, которую представляют для защиты преподавателю.

В процессе практического занятия учащиеся выполняют одну или несколько практических работ (заданий) под руководством преподавателя в соответствии с изучаемым содержанием учебного материала.

Ведущей дидактической целью практических занятий является формирование практических умений - профессиональных (умений выполнять определенные действия, операции, необходимые в последующем в профессиональной деятельности) или учебных (умений решать задачи по математике, физике, химии, информатике и др.), необходимых в последующей учебной деятельности по общепрофессиональным и специальным дисциплинам; практические занятия занимают преимущественное место при изучении общепрофессиональных и специальных дисциплин. Состав и содержание практических занятий направлены на реализацию Государственных требований.

На практических занятиях учащиеся овладевают первоначальными профессиональными умениями и навыками, которые в дальнейшем закрепляются и совершенствуются в процессе курсового проектирования и производственной (преддипломной) практики.

Наряду с формированием умений и навыков в процессе практических занятий обобщаются, систематизируются, углубляются и конкретизируются теоретические знания, вырабатывается способность и готовность использовать теоретические знания на практике, развиваются интеллектуальные умения.

К практическим работам предъявляется ряд требований, основным из которых является полное, исчерпывающее описание всей проделанной работы, позволяющее судить о полученных результатах, степени выполнения заданий и профессиональной подготовке учащихся.

Практические работы:

Тема практической работы №№1,2. Перевод чисел из одной системы счисления в другую. Арифметические действия с двоичными числами.

Объем часов 4

Цель практической работы. Изучить системы счисления. Получить умения по арифметическим действиям с разными системами счисления

Система счисления, или просто счисление, или нумерация,— набор конкретных знаков—цифр вместе с системой приемов записи, которая представляет числа этими цифрами.

Цель работы – приобретение навыков выполнения операций в различных системах счисления.

1. Основные понятия систем счисления

Система счисления — это совокупность правил и приемов записи чисел с помощью набора цифровых знаков. Количество цифр, необходимых для записи числа в системе, называют основанием системы счисления.

Основание системы записывается в справа числа в нижнем индексе: 5_{10} ; 1110110_2 ; $AF178_{16}$.

Различают два типа систем счисления:

- позиционные, когда значение каждой цифры числа определяется ее позицией в записи числа;
- непозиционные, когда значение цифры в числе не зависит от ее места в записи числа.

Примером непозиционной системы счисления является римская: числа IX, IV, XV и т.д. Примером позиционной системы счисления является десятичная система, используемая повседневно.

Любое целое число в позиционной системе можно записать в форме многочлена:

$$X_S = \{A_n A_{n-1} \dots A_2 A_1\} = A_n \cdot S^{n-1} + A_{n-1} \cdot S^{n-2} + \dots + A_2 \cdot S^1 + A_1 \cdot S^0,$$

где S — основание системы счисления;

A_n — цифры числа, записанного в данной системе счисления;

n — количество разрядов числа.

Пример. Число 6293_{10} запишется в форме многочлена следующим образом:

$$6293_{10} = 6 \cdot 10^3 + 2 \cdot 10^2 + 9 \cdot 10^1 + 3 \cdot 10^0$$

Десятичная система счисления – в настоящее время наиболее известная и используемая. неправильное название удерживается и поныне.

Десятичная система использует десять цифр — 0, 1, 2, 3, 4, 5, 6, 7, 8 и 9, а также символы “+” и “–” для обозначения знака числа и запятую или точку для разделения целой и дробной частей числа.

В вычислительных машинах используется двоичная система счисления, её основание — число 2. Для записи чисел в этой системе используют только две цифры — 0 и 1.

Таблица 1. Соответствие чисел, записанных в различных системах счисления

Десятичная	Двоичная	Восьмеричная	Шестнадцатеричная
1	001	1	1
2	010	2	2
3	011	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10

2. Правила перевода чисел из одной системы счисления в другую

Перевод чисел из одной системы счисления в другую составляет важную часть машинной арифметики. Рассмотрим основные правила перевода.

1. Для перевода двоичного числа в десятичное необходимо его записать в виде многочлена, состоящего из произведений цифр числа и соответствующей степени числа 2, и вычислить по правилам десятичной арифметики:

$$X_2 = A_n \cdot 2^{n-1} + A_{n-1} \cdot 2^{n-2} + A_{n-2} \cdot 2^{n-3} + \dots + A_2 \cdot 2^1 + A_1 \cdot 2^0$$

При переводе удобно пользоваться таблицей степеней двойки:

Таблица 2. Степени числа 2

n	0	1	2	3	4	5	6	7	8	9	10
2^n	1	2	4	8	16	32	64	128	256	512	1024

Пример. Число 11101000_2 перевести в десятичную систему счисления.

$$11101000_2 = 1 \cdot 2^7 + 1 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 = 232_{10}$$

2. Для перевода восьмеричного числа в десятичное необходимо его записать в виде многочлена, состоящего из произведений цифр числа и соответствующей степени числа 8, и вычислить по правилам десятичной арифметики:

$$X_8 = A_n \cdot 8^{n-1} + A_{n-1} \cdot 8^{n-2} + A_{n-2} \cdot 8^{n-3} + \dots + A_2 \cdot 8^1 + A_1 \cdot 8^0$$

При переводе удобно пользоваться таблицей степеней восьмерки:

Таблица 3.4. Степени числа 8

n	0	1	2	3	4	5	6
8^n	1	8	64	512	4096	32768	262144

Пример. Число 75013_8 перевести в десятичную систему счисления.

$$75013_8 = 7 \cdot 8^4 + 5 \cdot 8^3 + 0 \cdot 8^2 + 1 \cdot 8^1 + 3 \cdot 8^0 = 31243_{10}$$

3. Для перевода шестнадцатеричного числа в десятичное необходимо его записать в виде многочлена, состоящего из произведений цифр числа и соответствующей степени числа 16, и вычислить по правилам десятичной арифметики:

$$X_{16} = A_n \cdot 16^{n-1} + A_{n-1} \cdot 16^{n-2} + A_{n-2} \cdot 16^{n-3} + \dots + A_2 \cdot 16^1 + A_1 \cdot 16^0$$

При переводе удобно пользоваться таблицей степеней числа 16:

Таблица 3. Степени числа 16

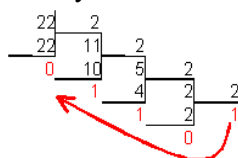
n	0	1	2	3	4	5	6
16^n	1	16	256	4096	65536	1048576	16777216

Пример. Число $FDA1_{16}$ перевести в десятичную систему счисления.

$$FDA1_{16} = 15 \cdot 16^3 + 13 \cdot 16^2 + 10 \cdot 16^1 + 1 \cdot 16^0 = 64929_{10}$$

4. Для перевода десятичного числа в двоичную систему его необходимо последовательно делить на 2 до тех пор, пока не останется остаток, меньший или равный 1. Число в двоичной системе записывается как последовательность последнего результата деления и остатков от деления в обратном порядке.

Пример. Число 22_{10} перевести в двоичную систему счисления.



$$22_{10} = 10110_2$$

5. Для перевода десятичного числа в восьмеричную систему его необходимо последовательно делить на 8 до тех пор, пока не останется остаток, меньший или равный 7. Число в восьмеричной системе записывается как последовательность цифр последнего результата деления и остатков от деления в обратном порядке.

Пример. Число 571_{10} перевести в восьмеричную систему счисления.

$$\begin{array}{r}
 571 \overline{) 8} \\
 \underline{56} \\
 11 \overline{) 8} \\
 \underline{8} \\
 3 \overline{) 8} \\
 \underline{3} \\
 0
 \end{array}$$

$571_{10} = 1073_8$

6. Для перевода десятичного числа в шестнадцатеричную систему его необходимо последовательно делить на 16 до тех пор, пока не останется остаток, меньший или равный 15. Число в шестнадцатеричной системе записывается как последовательность цифр последнего результата деления и остатков от деления в обратном порядке.

Пример. Число 7467_{10} перевести в шестнадцатеричную систему счисления.

$$\begin{array}{r}
 7467 \overline{) 16} \\
 \underline{7456} \\
 11 \overline{) 16} \\
 \underline{16} \\
 0
 \end{array}$$

$7467_{10} = 1D2B_{16}$

7. Чтобы перевести число из двоичной системы в восьмеричную, его нужно разбить на триады (тройки цифр), начиная с младшего разряда, в случае необходимости дополнив старшую триаду нулями, и каждую триаду заменить соответствующей восьмеричной цифрой (табл. 3).

Пример. Число 1001011_2 перевести в восьмеричную систему счисления.

$$001 \ 001 \ 011_2 = 113_8$$

8. Чтобы перевести число из двоичной системы в шестнадцатеричную, его нужно разбить на тетрады (четверки цифр), начиная с младшего разряда, в случае необходимости дополнив старшую тетраду нулями, и каждую тетраду заменить соответствующей шестнадцатеричной цифрой (табл. 3).

Пример. Число 1011100011_2 перевести в шестнадцатеричную систему счисления.

$$0010 \ 1110 \ 0011_2 = 2E3_{16}$$

9. Для перевода восьмеричного числа в двоичное необходимо каждую цифру заменить эквивалентной ей двоичной триадой.

Пример. Число 531_8 перевести в двоичную систему счисления.

$$531_8 = 101011001_2$$

10. Для перевода шестнадцатеричного числа в двоичное необходимо каждую цифру заменить эквивалентной ей двоичной тетрадой.

Пример. Число EE_{16} перевести в двоичную систему счисления.

$$EE_{16} = 111011101000_2$$

11. При переходе из восьмеричной системы счисления в шестнадцатеричную и обратно, необходим промежуточный перевод чисел в двоичную систему.

Пример 1. Число FEA_{16} перевести в восьмеричную систему счисления.

$$FEA_{16} = 111111101010_2$$

$$111 \ 111 \ 101 \ 010_2 = 7752_8$$

Пример 2. Число 6653_8 перевести в шестнадцатеричную систему счисления.

$$6653_8 = 110110101011_2$$

$$1101 \ 1010 \ 1011_2 = DAB_{16}$$

3. Арифметические действия над целыми числами в 2-ой системе счисления :

1. Операция сложения выполняется с + 0 1
использованием таблицы двоичного сложения в 0 0 1
одном разряде:

$$1 \ 1 \ 10_2$$

Пример.

$$\begin{array}{rcl}
 \text{а) } +1001_2 & \text{б) } +1101_2 & \text{в) } +11111_2 \\
 1010_2 & 1011_2 & 1_2 \\
 10011_2 & 11000_2 & 100000_2
 \end{array}$$

2. Операция вычитания выполняется с
использованием таблицы вычитания, в которой 1
обозначается заем в старшем разряде.

Пример.

$$\begin{array}{r}
 \text{а) } -101110011_2 \\
 \quad 100011011_2 \\
 \quad 001011000_2
 \end{array}
 \qquad
 \begin{array}{r}
 \text{б) } -110101101_2 \\
 \quad 101011111_2 \\
 \quad 001001110_2
 \end{array}$$

3.Операция умножения выполняется по обычной схеме, применяемой в десятичной с/с с последовательным умножением множимого на очередную цифру множителя.

Пример.

$$\begin{array}{r}
 \text{а) } \times 11001_2 \\
 \quad 1101_2 \\
 \quad 11001 \\
 \quad 11001 \\
 \quad 11001 \\
 \quad 11001 \\
 \hline
 101000101_2
 \end{array}
 \qquad
 \begin{array}{r}
 \text{б) } \times 101_2 \\
 \quad 11_2 \\
 \quad 101 \\
 \quad 101 \\
 \hline
 1111_2
 \end{array}$$

$$\begin{array}{r}
 \times 0 \ 1 \\
 0 \ 0 \ 0 \\
 1 \ 0 \ 1
 \end{array}$$

4.Операция деления выполняется по алгоритму, подобному алгоритму выполнения операции деления в 10-ой с/с.

Пример.

$$\begin{array}{r}
 101000101_2 \quad \overline{) 1101_2} \quad 100011000_2 \quad \overline{) 1111_2} \\
 \underline{1101} \qquad \qquad \underline{11001_2} \quad 1111 \\
 1110 \qquad \qquad \qquad 10100 \\
 \underline{1101} \qquad \qquad \qquad 1111 \\
 001101 \qquad \qquad \qquad 1010_2 \text{ -остаток} \\
 \underline{1101} \\
 0
 \end{array}$$

3.1.Сложение и вычитание в восьмеричной системе счисления.

При выполнении сложения и вычитания в 8-ой с/с необходимо соблюдать следующие правила:

1) в записи результатов сложения и вычитания могут быть использованы только цифры восьмеричного алфавита;

2) десяток восьмеричной системы счисления равен 8, т.е. переполнение разряда наступает, когда результат сложения больше или равен 8.

В этом случае для записи результата надо вычесть 8, записать остаток, а к старшему разряду прибавить единицу переполнения;

3)если при вычитании приходится занимать единицу в старшем разряде, эта единица переносится в младший разряд в виде восьми единиц.

Пример

$$\begin{array}{r}
 + 770_8 \\
 236_8 \\
 \hline
 1226_8
 \end{array}
 \qquad
 \begin{array}{r}
 + 750_8 \\
 236_8 \\
 \hline
 512_8
 \end{array}$$

3.2.Сложение и вычитание в шестнадцатеричной системе счисления.

При выполнении этих действий в 16-ой с/с необходимо соблюдать следующие правила:

1)при записи результатов сложения и вычитания надо использовать цифры шестнадцатеричного алфавита: цифры, обозначающие числа от 10 до 15 записываются латинскими буквами, поэтому, если результат является числом из этого промежутка, его надо записывать соответствующей латинской буквой;

2)десяток шестнадцатеричной системы счисления равен 16, т.е. переполнение разряда поступает, если результат сложения больше или равен 16, и в этом случае для записи результата надо вычесть 16, записать остаток, а к старшему разряду прибавить единицу переполнения;

3)если приходится занимать единицу в старшем разряде, эта единица переносится в младший разряд в виде шестнадцати единиц.

Примеры.

$$\begin{array}{r}
 + B09_{16} \\
 TFA_{16} \\
 \hline
 1A03_{16}
 \end{array}
 \qquad
 \begin{array}{r}
 + B09_{16} \\
 7FA_{16} \\
 \hline
 30F_{16}
 \end{array}$$

Задание

1. Выполнить перевод чисел

а) из 10-ой с/с в 2-ую систему счисления: 165; 541; 600; 720; 43,15; 234,99.

б) из 2-ой в 10-ую систему счисления: 110101₂; 11011101₂; 110001011₂; 1001001,111₂

в) из 2-ой с/с в 8-ую, 16-ую с/с:

100101110₂; 100000111₂; 111001011₂; 1011001011₂; 110011001011₂; 10101,10101₂; 111,011₂

г) из 10-ой с/с в 8-ую, 16-ую с/с: 69; 73; 113; 203; 351; 641; 478,99; 555,555

д) из 8-ой с/с в 10-ую с/с: 35₈; 65₈; 215₈; 327₈; 532₈; 751₈; 45,454₈

е) из 16-ой с/с в 10-ую с/с: D8₁₆; 1AE₁₆; E57₁₆; 8E5₁₆; FAD₁₆; AFF,6A7₁₆

2. Выпишите целые десятичные числа, принадлежащие следующим числовым промежуткам:

[10101₂; 110000₂]; [14₈; 20₈]; [18₁₆; 30₁₆]

3. Выполнить операции:

а) сложение в двоичной системе счисления

+ 1001001 ₂	+ 1011101 ₂	+ 10110011 ₂	+ 10111001,1 ₂
1011011 ₂	11101101 ₂	1010101 ₂	10001101,1 ₂

б) вычитание в 2-ой системе счисления

– 100001000 ₂	– 110101110 ₂	– 11101110 ₂	– 10111001,1 ₂
10110011 ₂	10111111 ₂	1011011 ₂	10001101,1 ₂

в) умножение в 2-ой системе счисления

× 100001 ₂	× 100101 ₂	× 111101 ₂	× 11001,01 ₂
111111 ₂	111011 ₂	111101 ₂	11,01 ₂

г) деление в 2-ой системе счисления

- 1) 111010001001₂ / 111101₂
- 2) 100011011100₂ / 110110₂
- 3) 10000001111₂ / 111111₂

д) сложение 8-ых чисел

+ 715 ₈	+ 524 ₈	+ 712 ₈	+ 321 ₈	+ 5731 ₈	+ 6351 ₈
73 ₈	57 ₈	763 ₈	765 ₈	1376 ₈	737 ₈

е) вычитание 8-ых чисел

– 137 ₈	– 436 ₈	– 705 ₈	– 538 ₈	– 7213 ₈
72 ₈	137 ₈	76 ₈	57 ₈	537 ₈

ж) сложение 16-ых чисел

+ A13 ₁₆	+ F0B ₁₆	+ 2EA ₁₆	+ ABC ₁₆	+ A2B ₁₆
16F ₁₆	1DA ₁₆	FCE ₁₆	C7C ₁₆	7F2 ₁₆

з) вычитание 16-ых чисел

– A17 ₁₆	– DFA ₁₆	– FO5 ₁₆	– DE5 ₁₆	– D3C1 ₁₆
1FC ₁₆	1AE ₁₆	AD ₁₆	AF ₁₆	D1F ₁₆

4. Вычислите выражение:

(1111101₂ + AF₁₆) / 36₈; 125₈ + 11101₂ × A2₁₆ / 1417₈

Тема практической работы №3. Построение логических схем.

Объем часов 2/1

Цель практической работы. Научиться строить логические схемы

Задание 1.

Операция, выражаемая словом "не", называется *логическим отрицанием (инверсией)* делает истинное выражение ложным и, наоборот, ложное – истинным. Обозначается « \neg ».

Обозначение: НЕ, $\neg A$, NOT A

Таблица истинности для логического выражения A имеет вид

A	$\neg A$
0	1
1	0

Конъюнкция (логическое умножение)

Операция, выражаемая связкой "и", называется *логическим умножением* (конъюнкцией) и обозначается " \wedge " (может также обозначаться знаками « \cdot » (точка) или $\&$). Высказывание $A \wedge B$ истинно тогда и только тогда, когда оба высказывания A и B истинны.

Обозначение: A и B , $A \wedge B$, $A \cdot B$, $A \text{ AND } B$

Таблица истинности для логических переменных A и B

A	B	$A \wedge B$
1	1	1
1	0	0
0	1	0
0	0	0

Дизъюнкция (логическое сложение)

Операция, выражаемая связкой "или" (в неисключающем смысле этого слова), называется логическим сложением (дизъюнкцией) и обозначается знаком \vee (или $+$). Высказывание $A \vee B$ ложно тогда и только тогда, когда оба высказывания A и B ложны.

Обозначение: A ИЛИ B , $A \vee B$, $A+B$, $A \text{ OR } B$

Таблица истинности для логических переменных A и B

A	B	$A \vee B$
1	1	1
1	0	1
0	1	1
0	0	0

В алгебре логики любую логическую функцию можно выразить через основные логические операции, записать ее в виде логического выражения и упростить ее, применяя законы логики и свойства логических операций. По формуле логической функции легко рассчитать ее таблицу истинности. Необходимо только учитывать порядок выполнения логических операций (приоритет) и скобки. Операции в логическом выражении выполняются слева направо с учетом скобок.

Приоритет выполнения логических операций:

- инверсия,
- конъюнкция,
- дизъюнкция.

Задание 1.

Построить таблицу истинности для логической функции $F = A \wedge (B \vee \neg B \wedge \neg C)$

1. Определить *количество строк* в таблице истинности, которое равно количеству возможных комбинаций значений логических переменных, входящих в логическое выражение: количество строк $= 2^n$, где n – количество переменных. Количество логических переменных – 3 (A, B, C) поэтому количество строк $= 2^n = 8$.

A	B	C	$\neg B$	$\neg C$	$\neg B \wedge \neg C$	$B \vee \neg B \wedge \neg C$	$A \wedge (B \vee \neg B \wedge \neg C)$
0	0	0	1	1	1	0	0
0	0	1	1	0	0	0	0
0	1	0	0	1	0	1	0
0	1	1	0	0	0	1	0
1	0	0	1	1	1	0	0
1	0	1	1	0	0	0	0
1	1	0	0	1	0	1	1
1	1	1	0	0	0	1	1

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

2. Определить количество столбцов:

количество столбцов = количество переменных + количество операций.

Количество логических операций - 5 (умножение - 2, сложение - 1, отрицание - 2), поэтому количество столбцов $3+5=8$

3. Построить таблицу истинности с указанным количеством строк и столбцов, обозначить столбцы и внести возможные наборы значений исходных логических переменных.

A	B	C	\bar{B}	\bar{C}	$\bar{B} \wedge \bar{C}$	$B \vee \bar{B} \wedge \bar{C}$	$A \wedge (B \vee \bar{B} \wedge \bar{C})$
0	0	0	1	1	1	1	0
0	0	1	1	0	0	0	0
0	1	0	0	1	0	1	0
0	1	1	0	0	0	1	0
1	0	0	1	1	1	1	1
1	0	1	1	0	0	0	0
1	1	0	0	1	0	1	1
1	1	1	0	0	0	1	1

4. Заполнить таблицу истинности по столбцам, выполняя базовые логические операции в необходимой последовательности и в соответствии с их таблицами истинности

Задание 2.

Построить таблицы истинности для логических функций

- 1) $F = A \vee \bar{B} \wedge (A \vee B)$.
- 2) $F = \bar{A} \wedge B \vee (A \wedge B)$.
- 3) $F = A \wedge B \wedge (C \vee \overline{A \wedge B})$.

Часть 2. Построение логических выражений для переключаемых схем.

Переключаемая схема — это схематическое изображение некоторого устройства, состоящего из переключателей и соединяющих их проводников, а также из входов и выходов, на которые подаётся и с которых снимается электрический сигнал.

В компьютерах и других автоматических устройствах широко применяются электрические схемы, содержащие сотни и тысячи переключаемых элементов: реле, выключателей и т.п. При разработке схем используется аппарат алгебры логики.


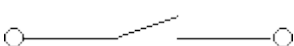
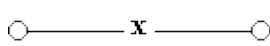
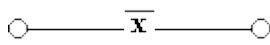
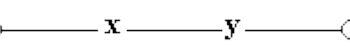
Каждый переключатель имеет только два состояния: замкнутое и разомкнутое. Переключателю X поставим в соответствие логическую переменную x , которая принимает значение 1 в том и только в том случае, когда переключатель X замкнут и схема проводит ток; если же переключатель разомкнут, то x равен нулю.

Будем считать, что два переключателя X и \bar{X} связаны таким образом, что когда X замкнут, то \bar{X} разомкнут, и наоборот. Следовательно, если переключателю X поставлена в соответствие логическая переменная x , то

переключателю \bar{X} должна соответствовать переменная \bar{x} .

Всей переключаемой схеме также можно поставить в соответствие логическую переменную, равную единице, если схема проводит ток, и равную нулю — если не проводит. Эта переменная является функцией от переменных, соответствующих всем переключателям схемы, и называется **функцией проводимости**.

Функции проводимости F некоторых переключаемых схем:

- 1)  Схема не содержит переключателей и проводит ток всегда, следовательно $F=1$;
- 2)  Схема содержит один постоянно разомкнутый контакт, следовательно $F=0$;
- 3)  Схема проводит ток, когда переключатель x замкнут, и не проводит, когда x разомкнут, следовательно, $F(x) = x$;
- 4)  Схема проводит ток, когда переключатель x разомкнут, и не проводит, когда x замкнут, следовательно, $F(x) = \bar{x}$;
- 5)  Схема проводит ток, когда оба переключателя замкнуты, следовательно, $F(x,y) = x \wedge y$;

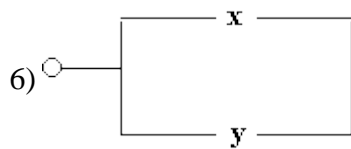


Схема проводит ток, когда хотя бы один из переключателей замкнут, следовательно, $F(x,y)=x \cup y$;

Любая сложная схема может быть преобразована в отдельные группы и представлена в виде логических функций нескольких переменных.

Задание 3.

Определить и проанализировать функции проводимости переключательных схем.

1)

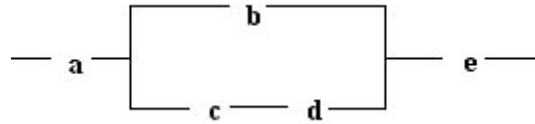


рис. 2

2)

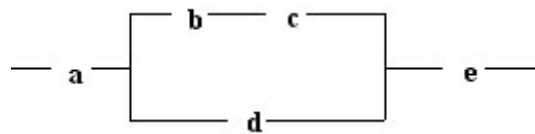


рис. 3

3)

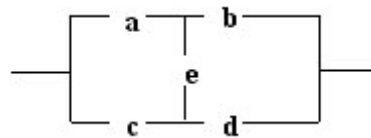


рис. 4

Тема практической работы №4,5. Современная вычислительная система как многоуровневая иерархическая система. Получение информации о параметрах компьютерной системы: шины расширений ввода-вывода.

Объем часов 4

Цель практической работы. Изучить принципы работы многоуровневой иерархической системы. Изучить шины расширений ввода-вывода

Задание 1.

Вывод одиночных символов. Для вывода символа на экран можно использовать непосредственное обращение к функции ввода-вывода операционной системы или BIOS, например:

```
MOV AH, 6  
MOV DL, 'C'  
INT 21H
```

Здесь в регистр AH загружается код функции, указывающий операционной системе, что должен быть выведен символ, содержащийся в регистре DL. Выводимый символ заносится в регистр DL, по команде INT 21H происходит прерывание и управление передается DOS. Одновременно с появлением символа на экране курсор перемещается на одну позицию вправо. Затем управление возвращается прикладной программе и начинается выполнение команды, следующей за командой INT 21H. В регистр DL можно помещать также ASCII код символа. Например, заглавной букве А соответствует код 41H.

В рассмотренном выше примере на экран выводится один символ, однако существует способ, при котором за одно обращение на экран выводится сразу несколько символов: эта возможность реализуется путем вызова функции вывода символьной строки. Соответствующая последовательность команд имеет вид

```
MOV AH, 9H  
MOV DX, OFFSET MSG  
INT 21H
```

Здесь в регистр AH помещается код вызываемой функции 9H, а в регистр DX загружается адрес последовательности байтов, которая должна быть выведена на экран. При обработке прерывания управление передается операционной системе, и строка символов выдается на экран. Выводимая строка может состоять из любых символов, но должна заканчиваться знаком конца строки – \$

Например, необходимо вывести на экран строку: THIS IS A MESSAGE Для описания строки, выводимой на экран, используется инструкция:

```
MSG DB 'THIS IS A MESSAGES','$'
```

Здесь MSG – имя переменной, содержащей строку, DB – тип переменной (последовательность байтов), '\$' – конец строки.

Управление дисплеем. При других обращениях к дисплею, используется прерывание INT 10H, по которой управление передается соответствующей подпрограмме BIOS.

Выбор режима. Дисплей IBM PC допускает вывод изображения в различных форматах, как текстовых, так и графических. Задание режима производится с помощью следующей программы:

```
MOV AH,0  
MOV AL,13H  
INT 10H
```

В регистр AH загружается 0, что указывает на изменение режима вывода информации на экран. В регистр AL также заносится число, определяющее режим, в котором на экране будет выводиться информация.

Вывод точки на экран. Для вывода на экран точки в графическом режиме можно использовать непосредственное обращение к функции BIOS или запись в видеопамять, например:

```
MOV AL,4  
MOV AH,0CH  
INT 10H
```

В регистр AL загружается номер цвета выводимой точки (в различных видеорежимах число цветов может быть разным – 2, 16 или 256). В регистр AH загружается номер функции BIOS (0C – вывод точки). По команде INT 10H управление передается BIOS и производится вывод точки на экран.

Считывание с клавиатуры. Ввод данных с клавиатуры можно производить двумя способами: считывание отдельных символов и ввод целой строки символов. Для ввода одного символа можно использовать функции BIOS, например:

```
MOV AH,00H  
INT 16H
```

В регистр АН загружается 0 – код функции BIOS для ввода 1 символа. По команде INT 16H происходит прерывание, и управление передается BIOS. Код введенного символа помещается в регистр AL.

Считывание строки производится иначе. Пока нажаты не все клавиши, управление программе не возвращается, а операционная система организует считывание данных с клавиатуры и запоминание кодов, соответствующих нажатым клавишам. Когда нажимается клавиша ENTER, управление передается основной программе вместе с последовательностью кодов, соответствующих нажатым клавишам. Такой процесс называется буферизованным вводом с клавиатуры. Буферизованный ввод сопровождается эхо-отображением, т. е. при считывании кода нажатой клавиши символ, соответствующий этому коду, выводится на экран дисплея. Допустим, что была нажата клавиша А; в этом случае буква А появится на экране, а в буфер прикладной программы будет передан один байт данных — код, соответствующий букве А.

При использовании функции буферизованного ввода необходимо указывать максимальное число символов, подлежащих считыванию, а при возврате из функции программе должно сообщаться фактическое число считанных с клавиатуры символов. Для обеспечения возможности обработки цепочки введенных символов пользователь, обращаясь к функции считывания строки, в качестве параметра передает адрес переменной, в которую будет помещена считанная строка. Строка может быть описана следующим образом:

```
BUF DB 10,0,0,0,0,0,0,0,0,0,0,0
```

В первом байте (число 10) указывается максимальное число клавиш, коды которых предполагается считывать в программе. Сюда же включается и код клавиши ENTER, которая должна нажиматься последней. Если пользователь попытается ввести больше символов, чем указано в буфере, раздастся сигнал звукового предупреждения, а очередной «лишний» код будет проигнорирован. Когда после завершения операции считывания управление будет вновь возвращено прикладной программе, второй байт строки будет содержать число считанных кодов. Таким образом, первые два байта не содержат самих кодов символов, соответствующих нажимавшимся клавишам. Эти представления находятся в следующих десяти байтах строки. Последним занесенным в буфер кодом будет код, соответствующий клавише ENTER.

Последовательность команд обращения к функции буферизованного ввода выглядит

следующим образом:

```
MOV AH, .0AH  
MOV DX.OFFSET BUF  
INT 21H
```

В регистр AH загружается код (0AH) вызова функции ввода строки. Метка BUF указывает на описанную выше строку. По команде INT 21H управление передается операционной системе. По мере того как будут нажиматься клавиши на клавиатуре, символы, соответствующие этим клавишам, будут появляться на экране, а соответствующие им восьмибайтовые коды будут заноситься в строку BUF. После введения необходимого числа символов необходимо нажать клавишу ENTER. Код, соответствующий этой клавише, помещается в строку BUF, и управление передается прикладной программе.

Ход работы

1. Наберите в любом текстовом редакторе тексты программ и сохраните их с расширением .ASM.
2. Откомпилируйте программы.
3. Запустите программы.
4. Загрузите откомпилированные программы в отладчик. Просмотрите структуру программы, выполните ее в пошаговом режиме. Для программы из таблицы 5.3 запишите код введенного с клавиатуры символа. Для программы из таблицы 5.4 найдите в памяти введенную строку символов.
5. Ответьте на вопросы к практической работе.
6. Сделайте вывод по проделанной работе.

Пример программы 5.1. Вывод текста на экран, используя функции операционной системы.

<code>.model small</code>	Выбираем модель памяти
<code>.stack 100h</code>	Задаем размер стека (256 байт)
<code>.data</code>	
<code>st1 db 'Ok!',13,10,'\$'</code>	На экран будет выведена строка - "Ok!"
<code>.code</code>	
<code>start:</code>	
<code>mov ax,@data</code>	Настроим сегментный регистр DS на
<code>mov ds,ax</code>	сегмент данных
<code>mov ah,09h</code>	Функция DOS для вывода на экран
	строки, адрес которой помещен в
	регистр DX
<code>mov dx,offset st1</code>	Загрузим в DX адрес строки, которую
	надо вывести на экран
<code>int 21h</code>	Вызов DOS
<code>mov ah,4ch</code>	Функция DOS завершения программы
<code>int 21h</code>	Вызов DOS
<code>end start</code>	

Пример программы 5.2. Рисование горизонтальной линии, используя функции BIOS.

<code>.model small</code>	
<code>.stack 100h</code>	
<code>.code</code>	
<code>mov ah,00h</code>	
<code>mov al,13h</code>	Задаем номер графического экрана (13h -
<code>int 10h</code>	EGA 320x240 256 цветов)
<code>mov cx,319</code>	Задаем число повторений цикла
<code>mov dx,100</code>	Задаем координаты точки
<code>lin:</code>	
<code>mov al,4</code>	Задаем цвет точки (4 - красный)
<code>mov ah,0ch</code>	Функция BIOS для вывода точки на экран
<code>int 10h</code>	
<code>loop lin</code>	Переход на метку lin
<code>mov ah,4ch</code>	Функция DOS завершения программы
<code>int 21h</code>	
<code>end</code>	

Пример программы 5.3. Ввод символа с клавиатуры, используя функции BIOS.

<code>.model small</code>	
<code>.stack 100h</code>	
<code>.code</code>	
<code>mov ah,00h</code>	Функция BIOS ввода с клавиатуры (ASCII код
	введенного символа в регистре AL)
<code>int 16h</code>	Вызов BIOS
<code>mov ah,4ch</code>	Функция DOS завершения программы
<code>int 21h</code>	
<code>end</code>	

Пример программы 5.4. Ввод строки символов с клавиатуры, используя функцию DOS.

```

.model small
.stack 100h
.data
string DB 50 dup("$")
.code
    mov ax,@data
    mov ds,ax
    mov dx,offset string
    mov ah,0ah
    int 21h

    mov ah,4ch
    int 21h
end

```

Подготовим место в памяти для введенной строки

Настроим регистр DS на начало сегмента данных

Настроим регистр DX на начало отведенной под строку области памяти

Функция DOS ввода строки символов,

ENTER – конец ввода

Функция DOS завершения программы

Тема практической работы №6. Поддержка функционирования информационных систем: подключение системной платы ПК.

Объем часов 2

Цель практической работы. Научиться поддерживать функционирование информационных систем

Задание 1.

1. Убедитесь в том, что компьютерная система обесточена (при необходимости, отключите систему от сети).
2. Разверните системный блок задней стенкой к себе.
3. По наличию или отсутствию разъемов USB установите форм-фактор материнской платы (при наличии разъемов USB - форм-фактор ATX, при их отсутствии - AT).
4. Установите местоположение и снимите характеристики следующих разъемов:
 - питания системного блока;
 - питания монитора;
 - сигнального кабеля монитора;
 - клавиатуры;
 - последовательных портов (два разъема);
 - параллельного порта;
 - других разъемов.
5. Убедитесь в том, что все разъемы, выведенные на заднюю стенку системного блока, не взаимозаменяемы, то есть каждое базовое устройство подключается одним единственным способом.
6. Изучите способ подключения мыши.

Мышь может подключаться к разъему последовательного порта или к специальному порту PS/2, имеющему разъем круглой формы. Последний способ является более современным и удобным. В этом случае мышь имеет собственный выделенный порт, что исключает возможность ее конфликта с другими устройствами, подключаемыми к последовательным портам. Последние модели могут подключаться к клавиатуре через разъем интерфейса USB.

7. Заполните таблицу:

Разъем	Тип разъема	Количество контактов	Примечания

8. Определить наличие основных устройств персонального компьютера.
9. Установите местоположение блока питания, выясните мощность блока питания (указана на ярлыке).
10. Установите местоположение материнской платы.
11. Установите характер подключения материнской платы к блоку питания.

Для материнских плат в форм-факторе AT подключение питания выполняется двумя разъемами. Обратите внимание на расположение проводников черного цвета - оно важно для правильной стыковки разъемов.

12. Установите местоположение жесткого диска.

Установите местоположение его разъема питания. Проследите направление шлейфа проводников, связывающего жесткий диск с материнской платой. Обратите внимание на местоположение проводника, окрашенного в красный цвет (на жестком диске он должен быть расположен рядом с разъемом питания).

13. Установите местоположения дисководов гибких дисков и дисковода CD-ROM.

Проследите направление их шлейфов проводников и обратите внимание на положение проводника, окрашенного в красный цвет, относительно разъема питания.

14. Установите местоположение платы видеоадаптера.

Определите тип интерфейса платы видеоадаптера.

15. При наличии прочих дополнительных устройств выявите их назначение, опишите характерные особенности данных устройств (типы разъемов, тип интерфейса и др.).

16. Заполните таблицу:

Устройство	Характерные особенности	Куда и при помощи чего подключается

Подготовьте отчет по практической работе.

Отчет должен содержать:

- наименование работы;
- цель работы;
- задание;
- последовательность выполнения работы;
- письменные ответы на контрольные вопросы;
- вывод о проделанной работе.

Тема практической работы №7. Установка и настройка программного обеспечения компьютерных систем: повышение производительности процессоров.

Объем часов 2/1

Цель практической работы. Научится контролировать производительность блоков ЭВМ

Задание 1.

Для сравнения различных типов компьютеров между собой применяют стандартные методики измерения производительности. Они позволяют разработчикам и пользователям осуществлять выбор между альтернативами на основе количественных показателей.

Производительность ПК во многом определяется производительностью ЦП. Производительность ЦП зависит от двух параметров: частоты синхронизации (тактовой частоты) и среднего количества тактов на команду. Невозможно изменить ни один из указанных параметров изолированно от другого, поскольку базовые технологии, используемые для изменения каждого из этих параметров, взаимосвязаны: частота синхронизации определяется технологией изготовления аппаратных средств и функциональной организацией процессора; среднее количество тактов на команду зависит от функциональной организации и архитектуры системы команд. Когда сравниваются две машины, необходимо рассматривать все компоненты, чтобы понять относительную производительность.

Кроме этого на производительность ПК влияет разрядность ЦП. Разрядность ЦП определяет, сколько разрядов отводится для формирования адреса при обращении к памяти. От этого зависит количество данных, обрабатываемых одной командой, и объем памяти, которую может использовать ЦП.

MIPS. Одной из альтернативных единиц измерения производительности процессора (по отношению ко времени выполнения) является MIPS - (миллион команд в секунду). В общем случае MIPS есть скорость операций в единицу времени, т.е. для любой данной программы MIPS есть просто отношение количества команд в программе к времени ее выполнения. Таким образом, производительность может быть определена как обратная ко времени выполнения величина, причем более быстрые машины при этом будут иметь более высокий рейтинг MIPS.

Однако использование MIPS в качестве метрики для сравнения наталкивается на две проблемы. Во-первых, MIPS зависит от набора команд процессора, что затрудняет

сравнение по MIPS компьютеров, имеющих разные системы команд. Во-вторых, MIPS даже на одном и том же компьютере меняется от программы к программе.

MFLOPS. Обычно для научно-технических задач производительность процессора оценивается в MFLOPS (миллионах чисел-результатов вычислений с плавающей точкой в секунду, или миллионах элементарных арифметических операций над числами с плавающей точкой, выполненных в секунду). Как единица измерения, MFLOPS, предназначена для оценки производительности только операций с плавающей точкой, и поэтому не применима вне этой ограниченной области. Ясно, что рейтинг MFLOPS зависит от машины, от программы. Этот термин менее безобидный, чем MIPS. Он базируется на количестве выполняемых операций, а не на количестве выполняемых команд. Именно поэтому рейтинг MFLOPS предназначался для справедливого сравнения различных машин между собой.

LINPACK – это пакет программ на Фортране для решения систем линейных алгебраических уравнений. Целью создания LINPACK отнюдь не было измерение производительности, но они могут быть использованы для этой цели. В основе действующего варианта LINPACK лежит решение системы 100 уравнений, содержащих 100 неизвестных (в последнем варианте размером 1000x1000). Все операции выполняются над числами с плавающей точкой, представленными с двойной точностью. Результат измеряется в MFLOPS.

Для многопроцессорных систем также имеются параллельные версии LINPACK, и такие системы часто показывают линейное увеличение производительности с ростом числа процессоров.

Однако, как и любая другая единица измерения, рейтинг MFLOPS для отдельной программы не может быть обобщен на все случаи жизни, чтобы представлять единственную единицу измерения производительности компьютера, хотя очень соблазнительно характеризовать машину единственным рейтингом MIPS или MFLOPS без указания программы.

Ход работы

1. Запустите Qbasic и наберите программу:

```
Cls: n=1000: dim a(n)
For I=1 to n
A(i)=int(rnd(1)*100)
Print a(i):print
```

```

Next I
For I=1 to n-1
For j=I+1 to n
If a(i)>a(j) then swap a(i), a(j)
Next j
Next I
For I=1 to n
Print a(i): print
Next I
Print "Конец работы"

```

2. Запустите программу и измерьте время ее выполнения. Для повышения достоверности результатов произведите измерения не менее 3 раз.
3. С помощью программы SPEEDSYS выясните характеристики ПК: тактовую частоту процессора, его разрядность, индекс производительности.
4. Повторите пункты 1-4 на другом ПК.
5. Результаты всех измерений занесите в таблицу 3.1.
6. Ответьте на вопросы к практической работе.
7. Напишите вывод по проделанной работе.

Таблица 3.1

Сравнение характеристик двух ПК

	ПК № 1	ПК № 2
Время выполнения программы, с		
Время выполнения программы, с		
Время выполнения программы, с		
Среднее время выполнения программы, с		
Частота процессора, МГц		
Разрядность процессора, бит		
Индекс производительности		

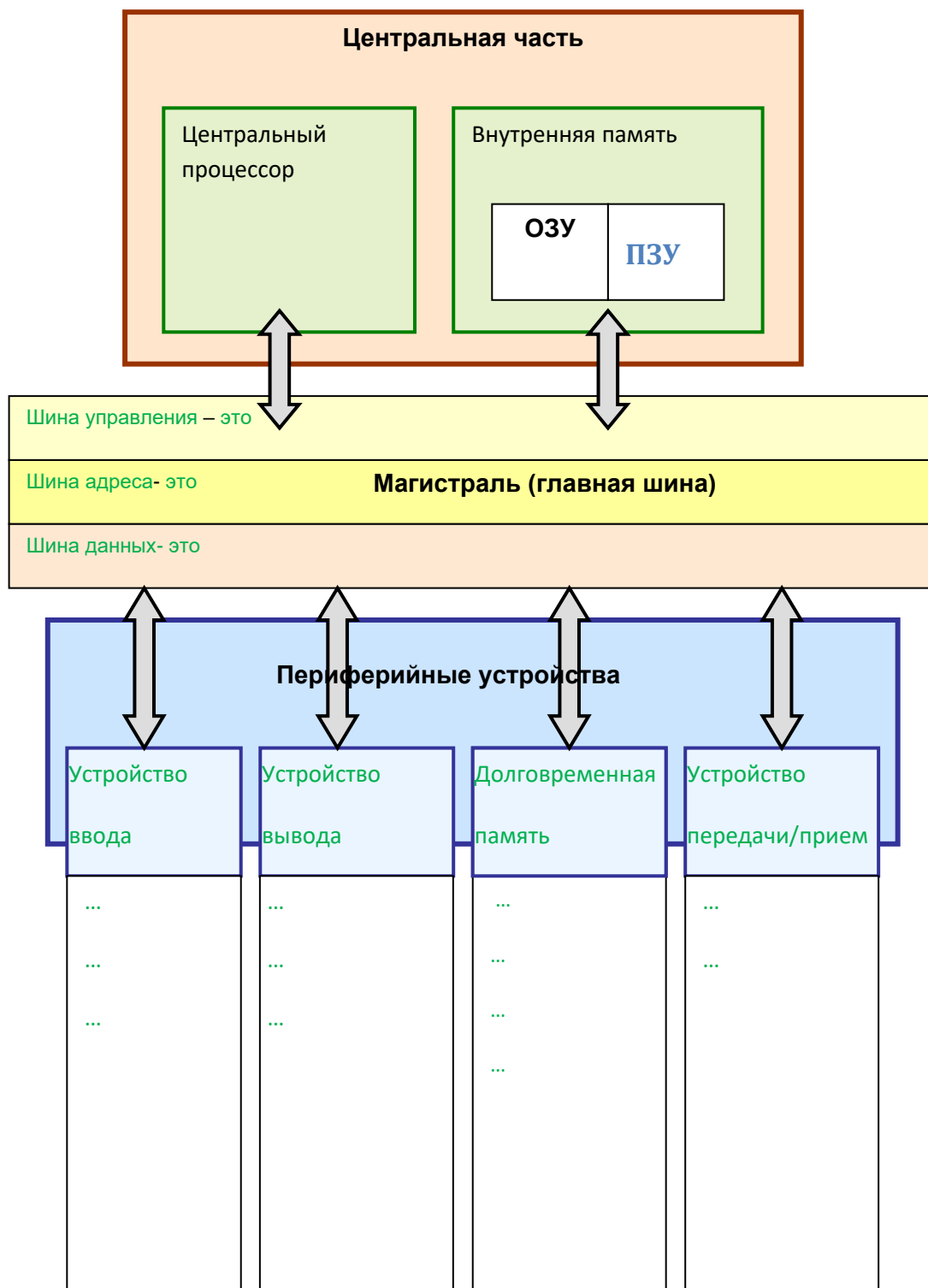
Тема практической работы №8. Магистрально-модульный принцип построения ЭВМ.

Объем часов 2/1

Цель практической работы. Понимать магистрально-модульный принцип построения

Задание 1.

Заполни схему недостающими определениями «Магистрально-модульный принцип построения компьютера».



Задание 2.

Восстанови описание последовательности обработки данных на компьютере.

Обработка данных на компьютере:

1. Пользователь запускает программу, хранящуюся в _____ памяти, она загружается в _____ память и начинает выполняться.
 2. Выполнение: процессор считывает команды и выполняет их. Необходимые данные загружаются из _____ памяти в _____ память или вводятся с помощью устройств _____ информации.
 3. Выходные (полученные) данные записываются процессором в _____ или _____ память, а также предоставляются пользователю с помощью устройств _____ информации.
- Для обеспечения информационного обмена между различными устройствами должна быть предусмотрена какая-то магистраль для перемещения потоков информации.

Задание 3.

Допиши утверждения.

Модульный принцип построения компьютера заключается в том, что ...

Магистрально-модульный принцип имеет ряд достоинств:

Принцип открытой архитектуры -

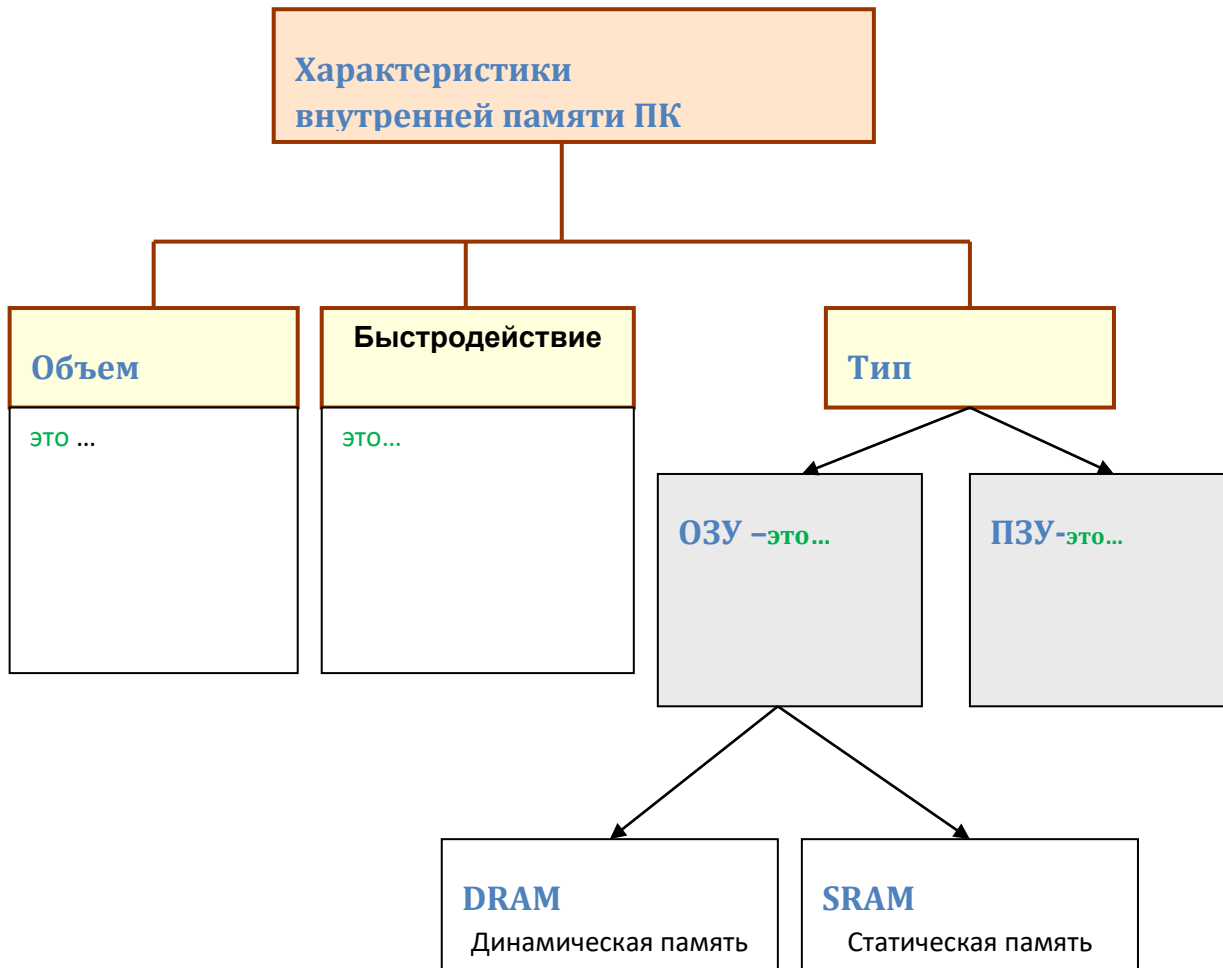
Тема практической работы №9. Модули динамической памяти.

Объем часов 2

Цель практической работы. Изучить модули динамической памяти

Задание 1.

Заполни схему недостающими характеристиками внутренней памяти компьютера.



Тема практической работы №10. Подключение дополнительного оборудования и настройка связи между элементами компьютерной системы: настройка отображения в разных режимах работы видеосистемы.

Объем часов 2/2

Цель практической работы. Научится настраивать видеосистему под конкретного пользователя

Задание 1.

Видеоподсистема персонального компьютера объединяет монитор и графический адаптер (видеокарту). Возможности видеоподсистемы в целом зависят как от монитора, так и от графического адаптера.

К настраиваемым параметрам видеоподсистемы относятся:

- разрешение экрана;
- частота обновления экрана (для мониторов с ЭЛТ);
- количество отображаемых цветов;
- заставка (в энергосберегающем режиме);
- фоновый рисунок;
- параметры энергосбережения.

Для настройки параметров видеоподсистемы компьютера с операционной системой Windows необходимо выбрать **Пуск|Панель управления**, а в открывшемся окне **Панель управления** выбрать ярлык **Экран**(рис. 1).

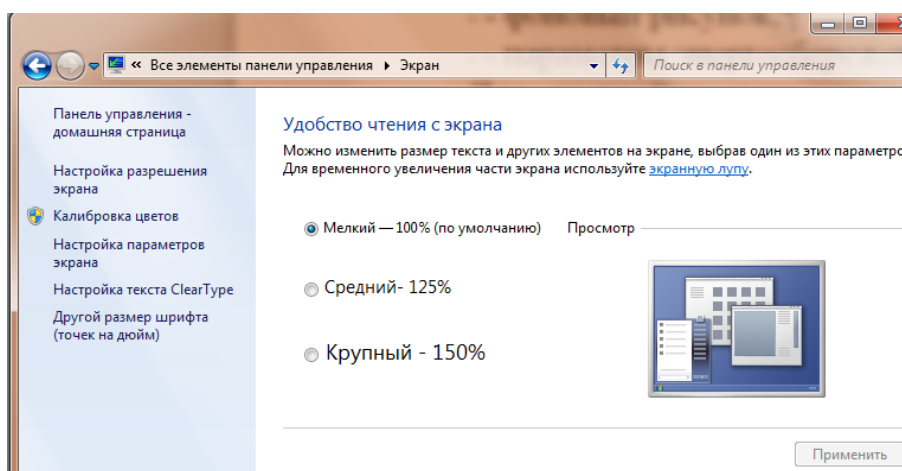


Рисунок 1

Посмотрите содержание и уясните функционал других закладок инструмента настройки **Экран**.

Задание

1. При одном и том же фоновом рисунке рабочего стола задайте разрешение 800 на 600 и 1024 на 768. Что изменилось, почему? Для каждого из этих разрешений сделайте копии экранов (используйте клавишу клавиатуры Print Screen), вставьте содержимое буфера в документ MS Word для отправки преподавателю.
2. Установите последовательно частоту обновления экрана 60, 75 Гц. Что Вы заметили?
3. Выберите один из стандартных фоновых рисунков рабочего стола, размер которого меньше размеров экрана и установите его. Скопируйте содержимое экрана в буфер и вставьте его в ранее созданный документ MS Word для отправки преподавателю.

Общие рекомендации

По всем вопросам, связанным с изучением дисциплины (включая самостоятельную работу), консультироваться с преподавателем.

Контроль и оценка результатов

Оценка за выполнение практической работы выставляется в форме **по пятибалльной системе** и учитывается как показатель текущей успеваемости студента.

По пятибалльной системе:

Качественная оценка индивидуальных образовательных достижений		Критерии оценки результата
балл (оценка)	вербальный аналог	
5	отлично	Представленные работы высокого качества, уровень выполнения отвечает всем требованиям, теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, выполнены все предусмотренные практической работой задания.
4	хорошо	Уровень выполнения работы отвечает всем требованиям, теоретическое содержание курса освоено полностью без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные практической работой задания выполнены, некоторые из выполненных заданий, возможно, содержат ошибки.
3	удовлетворительно	Уровень выполнения работы отвечает большинству основных требований, теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных практической работой заданий выполнено, некоторые виды заданий выполнены с ошибками.
2	не удовлетворительно	Теоретическое содержание курса освоено частично, необходимые практические навыки работы не сформированы, большинство предусмотренных практической работой заданий не выполнено.