

РОСЖЕЛДОР
Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Ростовский государственный университет путей сообщения»
(ФГБОУ ВО РГУПС)
Тихорецкий техникум железнодорожного транспорта – филиал РГУПС
(ТТЖТ – филиал РГУПС)

А.В. Украинский

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ
по выполнению практических занятий по дисциплине
«Вычислительная техника»
для студентов специальности
11.02.06 Техническая эксплуатация транспортного радиоэлектронного
оборудования (по видам транспорта)
III курса

Тихорецк, 2024

УТВЕРЖДАЮ

Заместитель директора по УР
Н.Ю. Шитикова



Методические указания по выполнению практических занятий по дисциплине «Вычислительная техника» разработаны в соответствии с рабочей учебной программой для специальности 11.02.06 Техническая эксплуатация транспортного радиоэлектронного оборудования (по видам транспорта).

Организация-разработчик: Тихорецкий техникум железнодорожного транспорта – филиал Федерального государственного бюджетного образовательного учреждения высшего образования «Ростовский государственный университет путей сообщения» (ТТЖТ – филиал РГУПС).

Составитель:

Украинский А.В., преподаватель ТТЖТ – филиала РГУПС

Рекомендованы цикловой комиссией № 4 специальностей 09.02.01, 11.02.06, 38.02.01

Протокол заседания № 1 от «02» сентября 2024 г.

Содержание

Практическое занятие № 1	5
Практическое занятие № 2	8
Практическое занятие № 3	10
Практическое занятие № 4	13
Практическое занятие № 5	14
Практическое занятие № 6	17
Практическое занятие № 7	19
Практическое занятие № 8	33
Практическое занятие № 9	37
Практическое занятие № 10	46
Практическое занятие № 11	48
Практическое занятие № 12	50
Практическое занятие № 13	52
Практическое занятие № 14	54
Практическое занятие № 15	56
Практическое занятие № 16	58
Практическое занятие № 17	60
Практическое занятие № 18	62
Практическое занятие № 19	64
Практическое занятие № 20	66
Список используемой литературы	69

Пояснительная записка

Настоящие методические рекомендации устанавливают требования к выполнению практических занятий.

Методические рекомендации к практическим занятиям составлены в соответствии с рабочей учебной программой по дисциплине Вычислительная техника и предназначены для студентов 3 курса специальности 11.02.06 Техническая эксплуатация транспортного радиоэлектронного оборудования (по видам транспорта).

Курс содержит практические занятия, соответствующие лекционному материалу. Основной целью курса является изучение формирование компетенции специальности 11.02.06 Техническая эксплуатация транспортного радиоэлектронного оборудования (по видам транспорта) для современного рынка труда.

Каждое практическое занятие по курсу содержит название, цели работы, теоретический материал, задания и контрольные вопросы. В методических рекомендациях подробно описан ход выполнения работы.

Курс практических занятий рассчитан на 40 часа.

Практические занятия выполняются студентами индивидуально в отдельных тетрадях.

В результате выполнения работ студенты закрепляют знания, полученные на теоретических занятиях и отрабатывают навыки эффективного применения современные технологии программирования для разработки прикладного программного обеспечения

Рекомендации предназначены для повышения качества и облегчения процесса выполнения заданий.

Практическое занятие № 1

Тема: Информация, единицы измерения

Цель работы:

Изучить понятие информации, единицы измерения.

Перечень используемого оборудования:

Монитор, системный блок, устройства ввода-вывода.

Краткие сведения из теории

Информация — это данные, которые имеют смысл и значение для получателя. Информация может быть представлена в различных формах, таких как текст, изображения, звук и видео.

Единицы измерения информации:

1. **Бит (bit):** Основная единица измерения информации, представляющая собой минимальную единицу данных, которая может принимать одно из двух значений (0 или 1).
2. **Байт (byte):** Состоит из 8 бит и используется для измерения объема данных.
3. **Килобайт (KB):** 1024 байта.
4. **Мегабайт (MB):** 1024 килобайта.
5. **Гигабайт (GB):** 1024 мегабайта.
6. **Терабайт (TB):** 1024 гигабайта.

Виды сигналов:

1. **Аналоговые сигналы:** Непрерывные сигналы, которые могут принимать любые значения в определенном диапазоне. Примеры: звуковые волны, электрические сигналы.
2. **Цифровые сигналы:** Дискретные сигналы, которые могут принимать ограниченное число значений. Примеры: бинарные данные, цифровые изображения.

Параметры сигналов:

1. **Амплитуда:** Максимальное значение сигнала.
2. **Частота:** Количество повторений сигнала за единицу времени (измеряется в герцах, Гц).
3. **Фаза:** Угол сдвига сигнала относительно начального момента времени.
4. **Период:** Время, за которое сигнал повторяется один раз.

Задания для выполнения студентами

Задание 1: Определение единиц измерения информации

- Преобразуйте следующие значения в байты:

1. 512 бит
2. 2 килобайта
3. 1 мегабайт
4. 0.5 гигабайта

Пример решения:

1. $512 \text{ бит} = 512 / 8 = 64 \text{ байта}$

2. $2 \text{ килобайта} = 2 * 1024 = 2048 \text{ байт}$

3. $1 \text{ мегабайт} = 1 * 1024 * 1024 = 1,048,576 \text{ байт}$

4. $0.5 \text{ гигабайта} = 0.5 * 1024 * 1024 * 1024 = 536,870,912 \text{ байт}$

Задание 2: Работа с аналоговыми сигналами

- Определите параметры аналогового сигнала с амплитудой 5 В, частотой 100 Гц и периодом 0.01 секунды.

Пример решения:

Амплитуда: 5 В

Частота: 100 Гц

Период: 0.01 секунды

Задание 3: Работа с цифровыми сигналами

- Определите параметры цифрового сигнала, который передает данные со скоростью 1 Мбит/с (мегабит в секунду).

Пример решения:

Скорость передачи данных: 1 Мбит/с

Амплитуда: 5 В (если используется стандартное напряжение)

Частота: 1 МГц (если используется бинарная модуляция)

Период: 1 микросекунда ($1 / 1 \text{ МГц}$)

Задание 4: Преобразование сигналов

- Преобразуйте аналоговый сигнал с амплитудой 3 В, частотой 50 Гц и периодом 0.02 секунды в цифровой сигнал с частотой дискретизации 1 кГц.

Пример решения:

Аналоговый сигнал:

Амплитуда: 3 В

Частота: 50 Гц

Период: 0.02 секунды

Цифровой сигнал:

Частота дискретизации: 1 кГц

Период дискретизации: 0.001 секунды

Содержание отчета

1. Наименование и цель работы;
2. Теоретические сведения;
3. Решение заданий
4. Ответы на контрольные вопросы;
5. Вывод о проделанной работе.

Контрольные вопросы по теме

1. Что такое информация и какие единицы измерения используются для её количественного описания?
2. Какие основные единицы измерения информации вы знаете?
3. Что такое аналоговые сигналы и какие их основные параметры?
4. Что такое цифровые сигналы и какие их основные параметры?
5. Как преобразовать биты в байты и наоборот?
6. Как определить параметры аналогового сигнала, зная его амплитуду, частоту и период?
7. Как определить параметры цифрового сигнала, зная его скорость передачи данных?
8. Как преобразовать аналоговый сигнал в цифровой сигнал с заданной частотой дискретизации?

Пример ответа на контрольный вопрос

Вопрос 1: Что такое информация и какие единицы измерения используются для её количественного описания?

Ответ: Информация — это данные, которые имеют смысл и значение для получателя. Информация может быть представлена в различных формах, таких как текст, изображения, звук и видео. Для количественного описания информации используются следующие единицы измерения: бит (bit), байт (byte), килобайт (KB), мегабайт (MB), гигабайт (GB) и терабайт (TB). Бит является основной единицей измерения информации и представляет собой минимальную единицу данных, которая может принимать одно из двух значений (0 или 1). Байт состоит из 8 бит и используется для измерения объема данных. Килобайт равен 1024 байтам, мегабайт — 1024 килобайтам, гигабайт — 1024 мегабайтам, а терабайт — 1024 гигабайтам.

Практическое занятие № 2

Тема: Носители информации

Цель работы:

Перечень используемого оборудования:

Монитор, системный блок, устройства ввода-вывода, инструменты для ремонта и технического обслуживания вычислительной техники.

Краткие сведения из теории

Виды носителей и способы записи информации:

1. Магнитная запись:

- **Принцип:** Информация записывается на магнитный носитель (например, жесткий диск, магнитная лента) путем изменения намагниченности материала.
- **Примеры:** Жесткие диски (HDD), магнитные ленты.

2. Оптическая запись:

- **Принцип:** Информация записывается на оптический носитель (например, CD, DVD) путем изменения отражательной способности поверхности лазером.
- **Примеры:** CD-ROM, DVD, Blu-ray.

3. Flash-память:

- **Принцип:** Информация записывается на полупроводниковый носитель путем изменения состояния транзисторов.
- **Примеры:** USB-накопители, SSD (твердотельные накопители), карты памяти.

Типы памяти:

1. Оперативное запоминающее устройство (ОЗУ):

- **Назначение:** Используется для временного хранения данных и программ, с которыми работает процессор.
- **Характеристики:** Быстрая, но требует постоянного питания для сохранения данных.
- **Примеры:** DRAM, SRAM.

2. Постоянное запоминающее устройство (ПЗУ):

- **Назначение:** Используется для хранения данных, которые не должны изменяться (например, BIOS).
- **Характеристики:** Медленнее ОЗУ, но сохраняет данные без питания.
- **Примеры:** ROM, EPROM, EEPROM.

Жесткий диск (HDD):

- **Устройство:** Состоит из одного или нескольких магнитных пластин, которые вращаются на шпинделе. Данные записываются и считываются с помощью магнитных головок.
- **Принцип доступа к информации:** Данные записываются и считываются с помощью магнитных головок, которые перемещаются над поверхностью пластин. Доступ к данным осуществляется по секторам и дорожкам.

Задания для выполнения студентами

1. Задание 1: Определение видов носителей и способов записи информации

- Опишите принципы магнитной, оптической и Flash-записи информации. Приведите примеры носителей для каждого способа записи.
- 2. **Задание 2: Определение типов памяти**
 - Опишите назначение и основные характеристики оперативного запоминающего устройства (ОЗУ) и постоянного запоминающего устройства (ПЗУ). Приведите примеры каждого типа памяти.
- 3. **Задание 3: Устройство и принцип работы жесткого диска**
 - Опишите устройство жесткого диска (HDD) и принцип доступа к информации. Какие компоненты входят в состав жесткого диска и как они взаимодействуют?
- 4. **Задание 4: Сравнение различных типов носителей**
 - Сравните магнитные диски, оптические диски и Flash-память по скорости доступа, емкости, надежности и стоимости. Приведите примеры использования каждого типа носителя.

Контрольные вопросы по теме

1. Какие виды носителей информации вы знаете и какие способы записи они используют?
2. Что такое магнитная запись и какие носители используют этот способ записи?
3. Что такое оптическая запись и какие носители используют этот способ записи?
4. Что такое Flash-память и какие носители используют этот способ записи?
5. Какие типы памяти вы знаете и какое их назначение?
6. Что такое оперативное запоминающее устройство (ОЗУ) и какие его основные характеристики?
7. Что такое постоянное запоминающее устройство (ПЗУ) и какие его основные характеристики?
8. Какое устройство жесткого диска (HDD) и как он работает?
9. Как осуществляется доступ к информации на жестком диске?
10. Сравните магнитные диски, оптические диски и Flash-память по скорости доступа, емкости, надежности и стоимости.

Пример ответа на контрольный вопрос

Вопрос 1: Какие виды носителей информации вы знаете и какие способы записи они используют?

Ответ: Виды носителей информации включают магнитные носители, оптические носители и Flash-память. Магнитные носители используют магнитную запись, при которой информация записывается на магнитный носитель путем изменения намагниченности материала. Примеры магнитных носителей включают жесткие диски (HDD) и магнитные ленты. Оптические носители используют оптическую запись, при которой информация записывается на оптический носитель путем изменения отражательной способности поверхности лазером. Примеры оптических носителей включают CD-ROM, DVD и Blu-ray. Flash-память использует полупроводниковый носитель, на котором информация записывается путем изменения состояния транзисторов. Примеры Flash-памяти включают USB-накопители, SSD (твердотельные накопители) и карты памяти.

Практическое занятие № 3

Тема: Логические элементы

Цель занятия:

Освоить основные принципы работы логических элементов и научиться проектировать и анализировать логические схемы.

Краткие сведения из теории

Логические элементы — это базовые строительные блоки цифровых схем, которые выполняют логические операции над входными бинарными сигналами. Они используются для создания различных комбинационных и последовательных логических схем, а также в компьютерах и других цифровых устройствах. Каждый логический элемент принимает два или более входных сигнала и формирует один выходной сигнал согласно заранее определенным логическим правилам.

Логические элементы являются основными компонентами цифровых устройств и вычислительных систем. Они выполняют базовые логические операции, такие как И (AND), ИЛИ (OR), НЕ (NOT), ИСКЛЮЧАЮЩЕЕ ИЛИ (XOR), И-НЕ (NAND), ИЛИ-НЕ (NOR) и ИСКЛЮЧАЮЩЕЕ ИЛИ-НЕ (XNOR). Эти элементы используются для построения более сложных логических схем, таких как сумматоры, мультиплексоры, демультиплексоры и триггеры.

Логические элементы могут быть реализованы с помощью различных технологий, включая транзисторы, диоды и интегральные схемы. Понимание работы логических элементов и их взаимодействия является ключевым для проектирования и анализа цифровых систем.

Основные типы логических элементов:

1. AND:
 - Выход 1 только при наличии на всех входах 1.
 - Логическая функция: $F=A \cdot B$ $F=A \cdot B$.
2. OR:
 - Выход 1, если хотя бы один вход равен 1.
 - Логическая функция: $F=A+B$ $F=A+B$.
3. NOT (инвертор):
 - Инвертирует входной сигнал ($0 \rightarrow 1$ и $1 \rightarrow 0$).
 - Логическая функция: $F=\bar{A}$.
4. NAND:
 - Выход 0 только тогда, когда все входы равны 1.
 - Логическая функция: $F=\bar{A} \cdot \bar{B}$.
5. NOR:
 - Выход 1 только тогда, когда все входы равны 0.
 - Логическая функция: $F=\overline{A+B}$.
6. XOR (исключающее или):
 - Выход 1 только при неравенстве входов.
 - Логическая функция: $F=A \oplus B$.
7. XNOR (исключающее и):
 - Выход 1 только при равенстве входов.

- Логическая функция: $F = \overline{A} \oplus \overline{B}$.

Задания для выполнения студентами

1. Реализация логических функций

- Используя логические элементы И (AND), ИЛИ (OR) и НЕ (NOT), реализуйте следующие логические функции:
 - ИСКЛЮЧАЮЩЕЕ ИЛИ (XOR).
 - И-НЕ (NAND).
 - ИЛИ-НЕ (NOR).
- Постройте схемы для каждой функции и объясните их работу.

2. Таблицы истинности

- Составьте таблицы истинности для следующих логических выражений:
 - A AND B.
 - A OR B.
 - NOT A.
 - A XOR B.
 - A NAND B.
 - A NOR B.
- Объясните, как каждая таблица истинности соответствует логическому выражению.

3. Анализ логических схем

- Рассмотрите следующую логическую схему:

A	---	>	AND	---	>	OR	---	>	OUT
B	---					C	---		

- Составьте таблицу истинности для этой схемы.
- Объясните, как работает схема и какие логические операции она выполняет.

4. Проектирование сумматора

- Используя логические элементы, спроектируйте полный сумматор (Full Adder) для двух однобитных чисел A и B с входом переноса C_{in}.
- Составьте таблицу истинности для сумматора.
- Объясните, как работает сумматор и какие логические операции он выполняет.

5. Проектирование мультиплексора

- Используя логические элементы, спроектируйте мультиплексор 2-на-1 (2-to-1 multiplexer) с двумя входами данных D0 и D1 и одним входом выбора S.
- Составьте таблицу истинности для мультиплексора.
- Объясните, как работает мультиплексор и какие логические операции он выполняет.

Контрольные вопросы

1. Какие основные логические элементы существуют и какие операции они выполняют?
2. Что такое таблица истинности и как она используется для анализа логических функций?
3. Как реализовать логическую функцию ИСКЛЮЧАЮЩЕЕ ИЛИ (XOR) с помощью базовых логических элементов?

4. Что такое логический элемент И-НЕ (NAND) и как он используется в цифровых схемах?
5. Как работает полный сумматор (Full Adder) и какие логические операции он выполняет?
6. Что такое мультиплексор и как он используется в цифровых системах?
7. Как реализовать логическую функцию ИЛИ-НЕ (NOR) с помощью базовых логических элементов?
8. Что такое логический элемент ИЛИ (OR) и как он используется в цифровых схемах?
9. Как работает логический элемент НЕ (NOT) и какие операции он выполняет?
10. Как использовать таблицы истинности для анализа и проектирования логических схем?
11. Какова роль логических элементов в цифровых системах?
12. Какие логические операции реализуются всеми базовыми логическими элементами?
13. Как можно применять карты Карно для упрощения логических выражений?
14. В чем отличие между AND и NAND?
15. Каковы преимущества использования логических элементов при проектировании схем?

Практическое занятие № 4

Тема: Создание и упрощение схем на логических элементах

Цель занятия:

Научиться создавать, анализировать и упрощать логические схемы.

Краткие сведения из теории

Логические схемы — это схемы, которые состоят из логических элементов (например, AND, OR, NOT, NAND, NOR, XOR и XNOR). Эти элементы выполняют логические операции над входными бинарными сигналами и выдают один выходной сигнал.

Основные логические элементы:

1. AND: Выдаёт 1 только тогда, когда все входные сигналы равны 1.
2. OR: Выдаёт 1, если хотя бы один из входных сигналов равен 1.
3. NOT: Инвертирует входной сигнал ($0 \rightarrow 1$ и $1 \rightarrow 0$).
4. NAND: Выдаёт 0 только тогда, когда все входные сигналы равны 1 (инверсия AND).
5. NOR: Выдаёт 1 только тогда, когда все входные сигналы равны 0 (инверсия OR).
6. XOR: Выдаёт 1 тогда и только тогда, когда количество единиц на входах нечётное.
7. XNOR: Выдаёт 1 тогда, когда количество единиц на входах чётное (инверсия XOR).

Упрощение логических схем — это процесс преобразования сложной логической схемы в более простую, сохраняя её функциональность. Это можно сделать с помощью методов, таких как:

- Алгебра логики: Использование свойств логических операций для преобразования формул.
- Картинки Карно (Картинки Карно): Визуальный метод упрощения логических выражений с помощью сетки.

Задания для выполнения студентами

1. Создание логической схемы:
- Разработайте логическую схему для функции, описанной таблицей истинности:

A	B	C	F(A, B, C)
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Контрольные вопросы

1. Что такое логические элементы и каковы их основные функции?
2. Объясните процесс упрощения логических схем с использованием алгебры логики.
3. Какова роль картинки Карно в упрощении логических выражений?
4. Какие преимущества предоставляет упрощение логических схем в процессе проектирования?
5. Каковы отличия между различными логическими элементами (например, AND и NAND)?

Практическое занятие № 5

Тема: Понятия систем счисления. Способы и методы перевода чисел

Цель занятия:

Ознакомиться с основами систем счисления и методов перевода чисел/

Краткие сведения из теории

Система счисления — это способ представления чисел с использованием определенного набора символов (цифр) и правил их комбинирования. Основные системы счисления включают:

1. **Десятичная система (основание 10):** Используется в повседневной жизни. Цифры: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.
2. **Двоичная система (основание 2):** Основа вычислительной техники. Цифры: 0, 1.
3. **Восьмеричная система (основание 8):** Используется в некоторых компьютерных системах. Цифры: 0, 1, 2, 3, 4, 5, 6, 7.
4. **Шестнадцатеричная система (основание 16):** Широко используется в программировании. Цифры: 0-9, A-F (где A=10, B=11, ..., F=15).

Способы и методы перевода чисел

Перевод чисел между различными системами счисления может быть выполнен несколькими методами:

1. **Метод деления с остатком:** Используется для перевода чисел из десятичной системы в любую другую систему счисления.
2. **Метод замены:** Используется для перевода чисел из одной системы счисления в другую через промежуточную десятичную систему.
3. **Метод таблиц:** Используется для перевода чисел между системами счисления с использованием таблиц соответствия.

Примеры которые демонстрируют основные методы перевода чисел между различными системами счисления.

Перевод чисел из десятичной системы в двоичную

1. **13 (десятичная) в двоичную:**
 - $13 / 2 = 6$ остаток 1
 - $6 / 2 = 3$ остаток 0
 - $3 / 2 = 1$ остаток 1
 - $1 / 2 = 0$ остаток 1
 - Результат: 1101
2. **25 (десятичная) в двоичную:**
 - $25 / 2 = 12$ остаток 1
 - $12 / 2 = 6$ остаток 0
 - $6 / 2 = 3$ остаток 0
 - $3 / 2 = 1$ остаток 1
 - $1 / 2 = 0$ остаток 1
 - Результат: 11001

Перевод чисел из двоичной системы в десятичную

1. 1011 (двоичная) в десятичную:

- $1 * 2^3 + 0 * 2^2 + 1 * 2^1 + 1 * 2^0$
- $8 + 0 + 2 + 1 = 11$

2. 11001 (двоичная) в десятичную:

- $1 * 2^4 + 1 * 2^3 + 0 * 2^2 + 0 * 2^1 + 1 * 2^0$
- $16 + 8 + 0 + 0 + 1 = 25$

Перевод чисел из десятичной системы в восьмеричную

1. 27 (десятичная) в восьмеричную:

- $27 / 8 = 3$ остаток 3
- $3 / 8 = 0$ остаток 3
- Результат: 33

2. 63 (десятичная) в восьмеричную:

- $63 / 8 = 7$ остаток 7
- $7 / 8 = 0$ остаток 7
- Результат: 77

Перевод чисел из восьмеричной системы в десятичную

1. 35 (восьмеричная) в десятичную:

- $3 * 8^1 + 5 * 8^0$
- $24 + 5 = 29$

2. 127 (восьмеричная) в десятичную:

- $1 * 8^2 + 2 * 8^1 + 7 * 8^0$
- $64 + 16 + 7 = 87$

Перевод чисел из десятичной системы в шестнадцатеричную

1. 15 (десятичная) в шестнадцатеричную:

- $15 / 16 = 0$ остаток 15 (F)
- Результат: F

2. 31 (десятичная) в шестнадцатеричную:

- $31 / 16 = 1$ остаток 15 (F)
- $1 / 16 = 0$ остаток 1
- Результат: 1F

Перевод чисел из шестнадцатеричной системы в десятичную

1. A (шестнадцатеричная) в десятичную:

- $A = 10$

2. 1F (шестнадцатеричная) в десятичную:

- $1 * 16^1 + F * 16^0$
- $16 + 15 = 31$

Задания для выполнения студентами

Задание 1: Перевод чисел из десятичной системы в двоичную

Переведите следующие числа из десятичной системы в двоичную:

1. 13
2. 25
3. 42
4. 100

Задание 2: Перевод чисел из двоичной системы в десятичную

Переведите следующие числа из двоичной системы в десятичную:

1. 1011
2. 11001
3. 101010
4. 1100100

Задание 3: Перевод чисел из десятичной системы в восьмеричную

Переведите следующие числа из десятичной системы в восьмеричную:

1. 27
2. 63
3. 125
4. 255

Задание 4: Перевод чисел из восьмеричной системы в десятичную

Переведите следующие числа из восьмеричной системы в десятичную:

1. 35
2. 127
3. 245
4. 377

Задание 5: Перевод чисел из десятичной системы в шестнадцатеричную

Переведите следующие числа из десятичной системы в шестнадцатеричную:

1. 15
2. 31
3. 63
4. 255

Задание 6: Перевод чисел из шестнадцатеричной системы в десятичную

Переведите следующие числа из шестнадцатеричной системы в десятичную:

1. A
2. 1F
3. 3F
4. FF

Контрольные вопросы по теме

1. Что такое система счисления?
2. Какие основные системы счисления вы знаете?
3. Как перевести число из десятичной системы в двоичную?
4. Как перевести число из двоичной системы в десятичную?
5. Как перевести число из десятичной системы в восьмеричную?
6. Как перевести число из восьмеричной системы в десятичную?
7. Как перевести число из десятичной системы в шестнадцатеричную?
8. Как перевести число из шестнадцатеричной системы в десятичную?
9. Какие методы перевода чисел между системами счисления вы знаете?
10. Приведите примеры использования различных систем счисления в вычислительной технике.

Практическое занятие № 6

Тема: Арифметические действия с числами в различных видах счисления.

Цель занятия:

Изучить арифметических действий с числами в различных системах счисления является.

Краткие сведения из теории

Арифметические действия с числами могут выполняться в различных системах счисления, включая десятичную, двоичную, восьмеричную и шестнадцатеричную. Понимание этих систем и правил выполнения арифметических операций критически важно для работы с цифровыми устройствами и программированием.

Основные системы счисления:

1. Десятичная система (основание 10):
 - Использует цифры от 0 до 9.
 - Основные операции: сложение, вычитание, умножение, деление.
2. Двоичная система (основание 2):
 - Использует цифры 0 и 1.
 - Основные арифметические действия аналогичны десятичным, но обозначения и правила отличаются.
3. Восьмеричная система (основание 8):
 - Использует цифры от 0 до 7.
 - Может воспринимать двоичные числа и упрощает представление для больших наборов данных.
4. Шестнадцатеричная система (основание 16):
 - Использует цифры от 0 до 9 и буквы A (10) до F (15).
 - Широко используется в программировании для представления данных и управления памятью.

Примеры арифметических действий

1. Сложение:
 - Десятичная: $27+15=42$
 - Двоичная: $11011_2+1011_2=11110_2$ ($27 + 11$)
 - Шестнадцатеричная: $1A_{16}+2F_{16}=49_{16}$ ($26 + 47$)
2. Вычитание:
 - Десятичная: $50-14=36$
 - Двоичная: $110010_2-1100_2=110110_2$ ($50 - 12$)
 - Шестнадцатеричная: $3C_{16}-1B_{16}=1B_{16}$ ($60 - 27$)
3. Умножение:
 - Десятичная: $6\times7=42$

- Двоичная: $101_2 \times 11_2 = 1111_2$ (5×3)
- Шестнадцатеричная: $2F_{16} \times 3_{16} = 5D_{16}$

4. Деление:

- Десятичная: $81 \div 9 = 9$
- Двоичная: $101100_2 \div 11_2 = 10010_2$
- Шестнадцатеричная: $B0_{16} \div 4_{16} = 1D_{16}$ $B0_{16} \div 4_{16} = 1D_{16}$

Задания для выполнения студентами

1. Сложение и вычитание:

- Выполните сложение и вычитание следующих чисел в двоичной и шестнадцатеричной системах:
 - $1011_2 + 1101_2$
 - $1A_{16} - 8_{16}$

2. Умножение и деление:

- Умножьте и поделите следующие числа в двоичной и восьмеричной системах:
 - $110_2 \times 101_2$
 - $57_8 \div 7_8$

3. Преобразования:

- Преобразуйте следующие числа из десятичной системы в двоичную и шестнадцатеричную:
 - 45 и 255.

4. Практическое задание:

- Напишите программу на любом языке программирования, которая принимает на вход два числа и операцию (сложение, вычитание) и выводит результат в разных системах счисления.

Контрольные вопросы

1. В чем заключается отличие между разными системами счисления?
2. Какова процедура сложения и вычитания в двоичной системе?
3. Какие преимущества дает использование шестнадцатеричной системы в программировании?
4. Как конвертировать числа из одной системы счисления в другую?
5. Какие трудности могут возникнуть при работе с арифметическими операциями в разных системах счисления?
6. Что такое система счисления?
7. Какие основные системы счисления существуют?
8. Как выполняется сложение чисел в двоичной системе?

Практическое занятие № 7

Тема: Вирусы и антивирусы

Цель работы:

Получить практические навыки определения видов вирусного заражения, инсталляции и настройки антивирусной программы.

Перечень используемого оборудования:

Монитор, системный блок, устройства ввода-вывода, антивирусные программы

Краткие сведения из теории

Компьютерный вирус – это специально написанная небольшая по размерам программа, имеющая специфический алгоритм, направленный на тиражирование копии программы, или её модификацию и выполнению действий развлекательного, пугающего или разрушительного характера.

Тем или иным способом вирусная программа попадает в компьютер и заражает их. Программа, внутри которой находится вирус, называется **зараженной**. Когда такая программа начинает работу, то сначала управление получает вирус. Вирус находит и заражает другие программы, а также выполняет какие-либо вредоносные действия. Например, портит файлы или таблицу размещения файлов на диске, занимает оперативную память и т.д. После того, как вирус выполнит свои действия, он передает управление той программе, в которой он находится, и она работает как обычно. Тем самым внешне работа зараженной программы выглядит так же, как и незараженной. Поэтому далеко не сразу пользователь узнаёт о присутствии вируса в машине.

Многие разновидности вирусов устроены так, что при запуске зараженной программы вирус остается в памяти компьютера и время от времени заражает программы и выполняет нежелательные действия на компьютере. Пока на компьютере заражено относительно мало программ, наличие вируса может быть практически незаметным.

К числу наиболее характерных **признаков заражения компьютера вирусами** относятся следующие:

- некоторые ранее исполнявшиеся программы перестают запускаться или внезапно останавливаются в процессе работы;
- увеличивается длина исполняемых файлов;
- быстро сокращается объём свободной дисковой памяти;
- на носителях появляются дополнительные сбойные кластеры, в которых вирусы прячут свои фрагменты или части повреждённых файлов;
- замедляется работа некоторых программ;
- в текстовых файлах появляются бессмысленные фрагменты;
- наблюдаются попытки записи на защищённую дискету;
- на экране появляются странные сообщения;
- появляются файлы со странными датами и временем создания (несуществующие дни несуществующих месяцев, годы из следующего столетия, часы, минуты и секунды, не укладывающиеся в общепринятые интервалы и т. д.);
- операционная система перестаёт загружаться с винчестера;
- появляются сообщения об отсутствии винчестера;
- данные на носителях портятся.

Любая дискета, не защищённая от записи, находясь в дисковом диске заражённого компьютера, может быть заражена. Дискеты, побывавшие в заражённом компьютере, являются разносчиками вирусов. Существует ещё один канал распространения вирусов, связанный с компьютерными сетями, особенно всемирной сетью Internet. Часто источниками заражения являются программные продукты, приобретённые нелегальным путем.

Существует несколько **классификаций компьютерных вирусов**:

1. **По среде обитания** различают вирусы сетевые, файловые, загрузочные и файлово-загрузочные.

2. **По способу заражения** выделяют резидентные и нерезидентные вирусы.

3. **По степени воздействия** вирусы бывают неопасные, опасные и очень опасные;

4. **По особенностям алгоритмов** вирусы делят на паразитические, репликаторы, невидимки, мутанты, троянские, макро-вирусы.

Загрузочные вирусы заражают загрузочный сектор винчестера или дискеты и загружаются каждый раз при начальной загрузке операционной системы.

Резидентные вирусы загружаются в память компьютера и постоянно там находятся до выключения компьютера.

Самомодифицирующиеся вирусы (мутанты) изменяют свое тело таким образом, чтобы антивирусная программа не смогла его идентифицировать.

Стелс-вирусы (невидимки) перехватывают обращения к зараженным файлам и областям и выдают их в незараженном виде.

Троянские вирусы маскируют свои действия под вид выполнения обычных приложений.

Вирусом могут быть заражены следующие объекты:

1. **Исполняемые файлы**, т.е. файлы с расширениями имен .com и .exe, а также оверлейные файлы, загружаемые при выполнении других программ. Вирусы, заражающие файлы, называются **файловыми**. Вирус в зараженных исполняемых файлах начинает свою работу при запуске той программы, в которой он находится. Наиболее опасны те вирусы, которые после своего запуска остаются в памяти резидентно - они могут заражать файлы и выполнять вредоносные действия до следующей перезагрузки компьютера. А если они заразят любую программу из автозапуска компьютера, то и при перезагрузке с жесткого диска вирус снова начнет свою работу.

2. **Загрузчик операционной системы и главная загрузочная запись жесткого диска**. Вирусы, поражающие эти области, называются **загрузочными**. Такой вирус начинает свою работу при начальной загрузке компьютера и становится резидентным, т.е. постоянно находится в памяти компьютера. Механизм распространения загрузочных вирусов - заражение загрузочных записей вставляемых в компьютер дискет. Часто такие вирусы состоят из двух частей, поскольку загрузочная запись имеет небольшие размеры и в них трудно разместить целиком программу вируса. Часть вируса располагается в другом участке диска, например, в конце корневого каталога диска или в кластере в области данных диска. Обычно такой кластер объявляется дефектным, чтобы исключить затирание вируса при записи данных на диск.

3. **Файлы документов, информационные файлы баз данных, таблицы табличных процессоров** и другие аналогичные файлы могут быть заражены **макро-вирусами**. Макро-вирусы используют возможность вставки в формат многих документов макрокоманд.

Если не принимать мер по защите от вирусов, то последствия заражения могут быть очень серьезными. Например, в начале 1989 г. вирусом, написанным американским студентом Моррисом, были заражены и выведены из строя тысячи компьютеров, в том числе принадлежащих министерству обороны США. Автор вируса был приговорен судом к трем месяцам тюрьмы и штрафу в 270 тыс. дол. Наказание могло быть и более строгим, но суд учел, что вирус не портил данные, а только размножался.

Для защиты от вирусов можно использовать:

- о общие средства защиты информации, которые полезны также как страховка от физической порчи дисков, неправильно работающих программ или ошибочных действий пользователей;
- о профилактические меры, позволяющие уменьшить вероятность заражения вирусом;
- о специализированные программы для защиты от вирусов.

Общие средства защиты информации полезны не только для защиты от вирусов. Имеются две основные разновидности этих методов защиты:

- резервное копирование информации, т. е. создание копий файлов и системных областей дисков на дополнительном носителе;
- разграничение доступа, предотвращающее несанкционированное использование информации, в частности, защиту от изменений программ и данных вирусами, неправильно работающими программами и ошибочными действиями пользователей.

Несмотря на то, что общие средства защиты информации очень важны для защиты от вирусов, все же их одних недостаточно. Необходимо применять специализированные программы для защиты от вирусов. Эти программы можно разделить на несколько видов:

1. Программы-**детекторы** позволяют обнаруживать файлы, зараженные одним из нескольких известных вирусов.

2. Программы-**доктора**, или **фаги**, восстанавливают зараженные программы убирая из них тело вируса, т.е. программа возвращается в то состояние, в котором она находилась до заражения вирусом.

3. Программы-**ревизоры** сначала запоминают сведения о состоянии программ и системных областей дисков, а затем сравнивают их состояние с исходным. При выявлении несоответствий об этом сообщается пользователю.

4. **Доктора-ревизоры** - это гибриды ревизоров и докторов, т.е. программы, которые не только обнаруживают изменения в файлах и системных областях дисков, но и могут автоматически вернуть их в исходное состояние.

5. **Программы-фильтры** располагаются резидентно в оперативной памяти компьютера, перехватывают те обращения к операционной системе, которые используются вирусами для размножения и нанесения вреда, и сообщают о них пользователю. Пользователь может разрешить или запретить выполнение соответствующей операции.

Ни один тип антивирусных программ по отдельности не дает полной защиты от вирусов. Поэтому наилучшей стратегией защиты от вирусов является **многоуровневая защита**.

Средствами разведки в защите от вирусов являются программы-детекторы, позволяющие проверять вновь полученное программное обеспечение на наличие вирусов.

На первом уровне защиты находятся резидентные программы для защиты от вируса. Эти программы могут первыми сообщить о вирусной атаке и предотвратить заражение программ и диска.

Второй уровень защиты составляют программы-ревизоры, программы-доктора и доктора-ревизоры. Ревизоры обнаруживают нападение тогда, когда вирус сумел пройти сквозь первый уровень. Программы-доктора применяются для восстановления зараженных программ, если ее копий нет в архиве, но они не всегда лечат правильно.

Доктора-ревизоры обнаруживают нападение вируса и лечат зараженные файлы, причем контролируют правильность лечения.

Третий уровень защиты - это средства разграничения доступа. Они не позволяют вирусам и неверно работающим программам, даже если они проникли в компьютер, испортить важные данные.

В **резерве** находятся архивные копии информации и эталонные диски с программными продуктами. Они позволяют восстановить информацию при ее повреждении на жестком диске.

Среди наиболее распространенных российских антивирусных пакетов следует отметить **Kaspersky Antivirus, DrWeb, Adinf**. Перечисленные средства могут оказать серьезную помощь в обнаружении вирусов и восстановлении повреждённых файлов, однако не менее важно и соблюдение сравнительно простых **правил антивирусной безопасности**.

1. Следует избегать пользоваться нелегальными источниками получения программ. Наименее же опасен законный способ покупки фирменных продуктов.
2. Осторожно следует относиться к программам, полученным из сети Internet, так как нередки случаи заражения вирусами программ, распространяемых по электронным каналам связи.
3. Всякий раз, когда дискета побывала в чужом компьютере, необходимо проверить дискету с помощью одного или двух антивирусных средств.
4. Необходимо прислушиваться к информации о вирусных заболеваниях на компьютерах в своем районе проживания или работы и о наиболее радикальных средствах борьбы с ними. Атакам нового вируса в первую очередь подвергаются компьютеры образовательных учреждений.
5. При передаче программ или данных на своей дискете её следует обязательно защитить от записи.

Антивирус Касперского предназначен для антивирусной защиты персональных компьютеров, работающих под управлением операционной системы Windows.

Защита от вирусов и вредоносных программ - обнаружение и уничтожение вредоносных программ, проникающих через съемные и постоянные файловые носители, электронную почту и протоколы интернета. Можно выделить следующие варианты работы программы (они могут использоваться как отдельно, так и в совокупности):

Постоянная защита компьютера - проверка всех запускаемых, открываемых и сохраняемых на компьютере объектов на присутствие вирусов.

Проверка компьютера по требованию - проверка и лечение как всего компьютера в целом, так и отдельных дисков, файлов или каталогов. Такую проверку вы можете запускать самостоятельно или настроить ее регулярный автоматический запуск.

Восстановление работоспособности после вирусной атаки. Полная проверка и лечение позволяет вам удалить все вирусы, поразившие ваши данные при вирусной атаке.

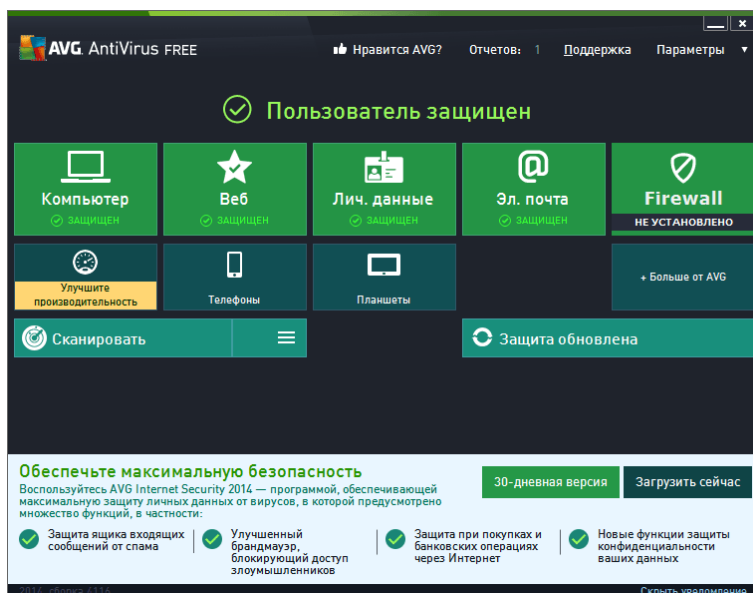
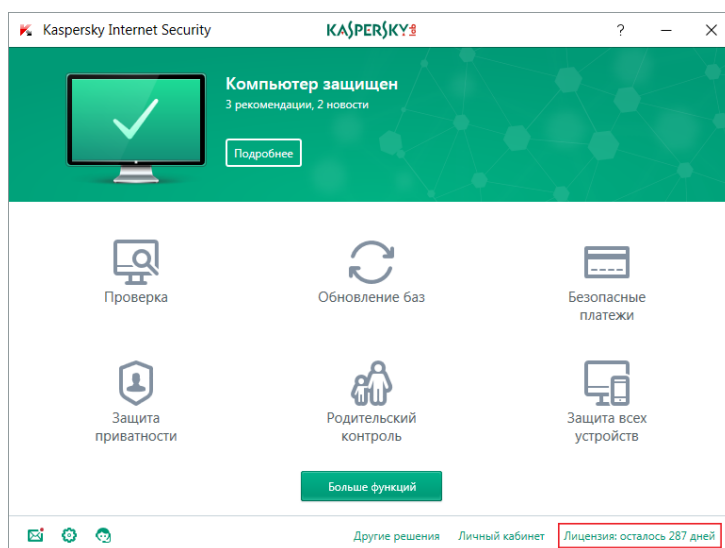
Проверка и лечение входящей/исходящей почты - анализ на присутствие вирусов и лечение входящей почты до ее поступления в почтовый ящик и исходящей почты в режиме реального времени. Кроме того, программа позволяет проверять и лечить почтовые базы различных почтовых клиентов по требованию.

Обновление антивирусных баз и программных модулей - пополнение антивирусных баз информацией о новых вирусах и способах лечения зараженных ими объектов, а также обновление собственных модулей программы. Обновление выполняется с серверов обновлений Лаборатории Касперского или из локального каталога.

Карантин - помещение объектов, возможно зараженных вирусами или их модификациями, в специальное безопасное хранилище, где вы можете их лечить, удалять, восстанавливать в исходный каталог, а также отправлять экспертам Лаборатории Касперского на исследование. Файлы на карантине хранятся в специальном формате и не представляют опасности.

Формирование отчета - фиксирование всех результатов работы Антивируса Касперского в отчете. Подробный отчет о результатах проверки включает общую статистику по проверенным объектам, хранит настройки, с которыми была выполнена та или иная задача, а также последовательность проверки и обработки каждого объекта в отдельности.

AVG Antivirus, AVG Anti-Virus — антивирусная система производства чешской компании AVG Technologies, имеющая сканер файлов, сканер электронной почты и поддерживающая возможность автоматического наблюдения. Система безопасности AVG сертифицирована всеми главными независимыми сертификационными компаниями, такими как ICSA, AV-TEST, Virus Bulletin, Checkmark (лаборатория West Coast Labs).



AVG AntiVirus Free - один из самых популярных бесплатных антивирусных продуктов, обладающий всеми необходимыми инструментами для своевременной и качественной защиты компьютера и системы от всех известных и новейших угроз. Это приложение включает в себя такие компоненты как:

Антивирус и антишпион (AntiVirus, Anti-Spyware) - обеспечивает мощную защиту от вирусов, троянов, червей, потенциально нежелательных программ, шпионских модулей, рекламного и другого вредоносного ПО.

Анти-руткит (Anti-Rootkit) - позволяет находить руткиты, которые скрываются в системе или маскируются под другие приложения.

Resident Shield - модуль сканирования в фоновом режиме всех файлов, с которыми в данный момент времени работает система.

Identity Protection - компонент для защиты личных данных, который предотвратит кражу персональной и приватной информации (логины, пароли, номера кредитных карт и т.д.).

Сканер электронной почты (E-mail Scanner) - полная проверка всей входящей и исходящей корреспонденции при использовании любого почтового клиента.

LinkScanner - защита от интернет атак при ежедневном серфинге, когда вы ищите нужную информацию или посещаете ресурсы всемирной паутины.

PC Analyzer - всесторонний анализ компьютера, который выявляет проблемы в работе реестра, ошибки фрагментации, находит мусорные файлы и т.д., что поможет вам оптимизировать быстродействие системы и освободить дополнительное место на дисках.

Стоит отметить, что вирусные базы этого комплексного антивирусного решения обновляются несколько раз в день, так что ни один самый новый вредонос не останется незамеченным для AVG AntiVirus Free. Кроме того, этот бесплатный антивирус быстр в работе, очень умерен к потребляемым ресурсам ПК, позволяя полноценно функционировать без ущерба другим приложениям даже на слабых машинах.

Norton Security – комплексный антивирус, построенный на основе надежных решений безопасности от компании Symantec.

Новый продукт включает в себя набор надежных технологий антивирусных программ мирового класса – Norton Antivirus, Norton Internet Security и Norton 360, и заменяет их в едином решении защиты.

Norton Security использует 5 уровней безопасности компьютера, важных данных и онлайн-активности, включая технологии сетевой защиты, такие как брандмауэр и система предотвращения вторжений (IPS), технологии файловой защиты (традиционные антивирусные возможности), технологии защиты на основе репутации (Insight) и технологии на основе поведенческого анализа (SONAR).

Основные возможности Norton Security.

- Один сервис для защиты всех ваших устройств под управлением операционных систем Windows, Mac OS X и мобильных платформ Android, iOS.

- Обеспечивает защиту от вирусов, шпионских программ, вредоносных программ и других интернет-атак.

- Обеспечивает защиту приватности, независимо от того, какое устройство вы используете.

- Блокирует небезопасные веб-сайты и предотвращает подозрительные загрузки.

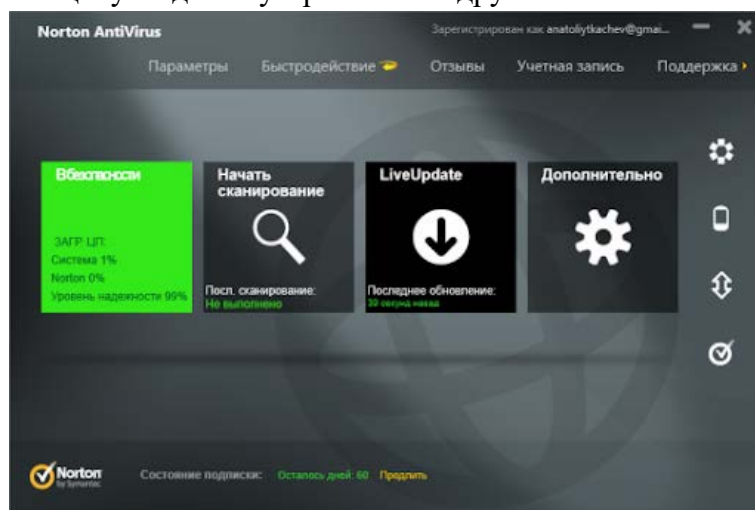
- Позволяет переносить защиту с одного устройства на другое.

- Позволяет добавлять больше защиты, если у вас появилось больше устройств.

- Легко находит потерянные или украденные смартфоны и планшеты.

- Автоматическое резервное копирование ваших ценных фотографий, фильмов и файлов с вашего ПК на защищенное облачное хранилище объемом 25 Гб.

- Обеспечивает



достаточную гибкость для защиты цифровой жизни всей вашей семьи.

Avira Free Antivirus – бесплатный антивирус, антишпион и антируткит с облачной технологией защиты Avira Protection Cloud, в режиме реального времени защищает компьютер от вредоносных программ и позволяет управлять Брандмауэром Windows.

Антивирус предлагает базовую защиту от вредоносных программ, но может быть усилен приложениями и сервисами от Avira, включая добавление интернет-защиты, родительского контроля и защиту Android, что позволяет защищать компьютер, важную информацию, мобильные устройства и ваших детей от различного рода онлайн-угроз. Решение Авира в режиме реального времени защищает компьютер от вирусов, программ-шпионов, рекламного ПО, руткитов, вредоносных веб-сайтов и загрузок.

Основные возможности Avira Free Antivirus:

Антивирус и антишпион – Эффективная защита в режиме реального времени и по запросу от различного рода вредоносных программ: вирусов, троянов, интернет-червей, программ-шпионов и рекламного ПО. Постоянные автоматические обновления и эвристическая технология AHeAD надежно защищают от известных и новых угроз.

Облачная защита – Облачная технология защиты Avira Protection Cloud - классификация угроз в реальном времени и быстрое сканирование системы.

Защита от руткитов – Анти-руткит Avira защищает от сложных в обнаружении угроз - руткитов.

Управлением Брандмауэром Windows – Avira Free Antivirus позволяет редактировать сетевые правила для приложений, изменить профили сети (Частная, Общая) и управлять расширенными параметрами Брандмауэра Windows в режиме повышенной безопасности.

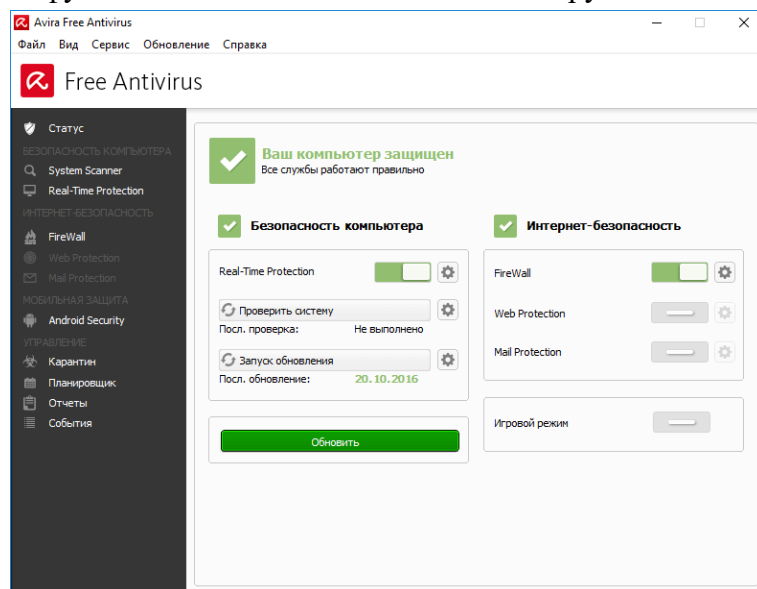
Интернет-защита – Безопасный поиск, блокировка фишинговых и вредоносных веб-сайтов, защита от слежения. Эта функция является частью панели инструментов Avira Browser Safety для браузеров Chrome, Firefox и Opera (устанавливается отдельно от Avira Free Antivirus).

Родительский контроль – С функцией Социальные сети на основе технологии Avira Free SocialShield вы можете контролировать деятельность своих детей в Интернете: проверять их учетные записи в социальных сетях на наличие комментариев, фотографий и т.д., которые могут повлиять на вашего ребенка с негативной стороны.

Защита Android-устройств – Антивирусное решение Avira Free Antivirus, помимо защиты компьютера от различных угроз, предлагает установить приложение Avira Antivirus Security для Android, которое защитит ваш смартфон или планшет от потери и кражи, а также позволит блокировать нежелательные звонки и SMS-сообщения.

McAfee LiveSafe - универсальное средство защиты ваших устройств, важных данных и веб-серфинга, позволяет устанавливать антивирусное ПО на каждый компьютер на Windows, Mac, на смартфоны и планшеты Android, iPhone и iPad.

Основные функции McAfee LiveSafe:



- Антивирус и антишпион: Защита от вирусов и интернет-угроз
- McAfee WebAdvisor: Защита от рискованных веб-сайтов и предотвращение загрузки опасных файлов
- True Key: Простое управление именами пользователей и паролями
- McAfee Mobile Security: Всесторонняя защита мобильных устройств
- Безопасное облачное хранилище

Основные возможности McAfee LiveSafe

Устройства под управлением Windows

- Защита с возможностями комплексного антивируса McAfee Internet Security.
- Защитите себя от вирусов, вредоносных программ, программ-вымогателей, шпионских программ, нежелательных программ и прочих угроз
- Защита подключения по Wi-Fi, средства веб-безопасности McAfee WebAdvisor и инструменты оптимизации компьютера
- Защита от нежелательной почты и родительский контроль

Устройства под управлением Android

- Всесторонняя защита мобильных устройств с помощью McAfee Mobile Security для Android
- Резервное копирование данных, защита от краж, защита конфиденциальности приложений и защита подключения по Wi-Fi для планшетов и смартфонов

Устройства Mac и iOS

- Защита от вредоносных программ, брандмауэр и средства веб-безопасности для компьютеров Mac
- Защита файлов и защита от кражи для планшетов и смартфонов под управлением iOS

Диспетчер личных данных True Key

- Быстрый и удобный вход на сайты, в приложения и т. д. с помощью технологии распознавания лица и доверенных устройств (многофакторная проверка подлинности)

5 учетных записей пользователей

Безопасное облачное хранилище

- Обеспечение конфиденциальности личных данных

Безопасное хранение и просмотр фотографий и важных документов в облаке

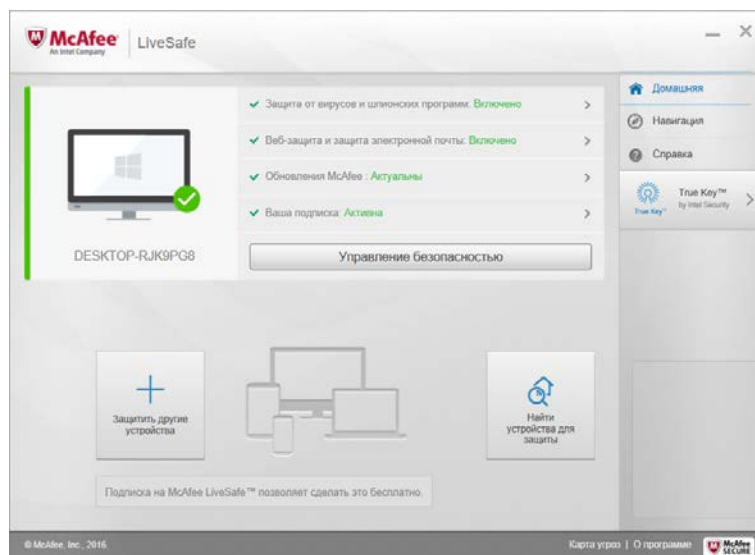
Консоль управления безопасностью

- Проверка состояния безопасности всех защищенных устройств

- Удаленное обнаружение и блокировка потерянного или украденного смартфона или планшета, а также уничтожение хранящихся на нем данных

- Поддерживаемые ОС: Windows, Android, Mac и iOS

Бесплатная поддержка



– На время действия подписки: техническая поддержка, услуги консультантов по безопасности и служба по работе с клиентами.

Антивирус Dr.Web - базовая защита от вирусов, шпионского ПО, шифровальщиков и хакеров. Включает комплекс программ: антивирус, антишпион, фаервол, монитор активности и защита от вымогателей

Защитные компоненты Антивируса Dr.Web

Антивирус. Блокирует вирусы и вредоносные программы, предотвращая им доступ в защищаемую систему, и вылечивает инфицированные программы и файлы.

Фаервол. Создает барьеры для хакерских атак, нацеленных на проникновение в систему.

Защита от эксплойтов. Обеспечивает защиту от вредоносных объектов, которые используют уязвимости в популярном программном обеспечении для проведения атаки.

Монитор активности. Поведенческий анализ защищает от проникновения новейших, наиболее опасных вредоносных программ.

Защита от шифровальщиков. Блокирует программы-вымогатели на основе поведенческих признаков.

Dr.Web Cloud. Проверка ссылок на серверах Dr.Web в режиме реального времени.



Антивирус NOD32 - новое антивирусное решение от ESET, предлагающее улучшенный эвристический анализ неизвестных угроз, «облачные» технологии ESET Live Grid для определения репутации файлов и обновленный интерфейс, включает возможности автоматического сканирования компьютера во время его простоя, проверки файлов непосредственно во время загрузки и возможность отменять установленные обновления.

Антивирус ESET NOD32: Основные компоненты

Компьютер

- Защита файловой системы в режиме реального времени
- Защита документов
- Контроль устройств
- Проактивная защита HIPS
- Игровой режим
- Защита Anti-Stealth (обнаружение руткитов)

Интернет и электронная почта

- Защита доступа в Интернет
- Защита почтового клиента
- Защита от фишинга

Преимущества ESET NOD32 Antivirus

– Устранение всех типов угроз, включая вирусы, руткиты и шпионское ПО. Возможности «облачного» сканирования для быстрой проверки компьютера, высокий уровень защиты и проверка USB-флешек, CD и DVD-дисков при подключении.

– Антивирус NOD32 оставляет больше ресурсов для программ, которые вы используете ежедневно, решение позволяет играть, работать и посещать веб-сайты в Интернете без замедлений. Оставляет больше времени для онлайн-активности в режиме экономии батареи, и предлагает игровой режим, чтобы защита не отвлекала всплывающими окнами.

– Обеспечивает высокий уровень защиты с настройками по умолчанию. Антивирус NOD32 позволяет легко настроить и запустить сканирование и быстро найти необходимые параметры. Кроме того, вы можете точно настроить вашу безопасность с использованием более сотни детальных настроек.

Основные возможности ESET NOD32 Antivirus

– Интеллектуальный режим «Smart mode» для HIPS: располагается между автоматическим и интерактивным режимами. Способен выявлять подозрительную активность и вредоносные процессы в системе.

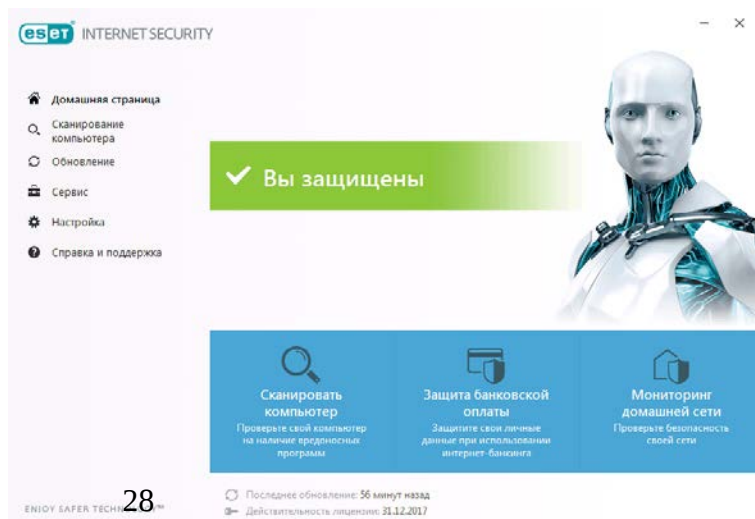
– Блокировщик эксплойтов (Exploit Blocker): дополнительный уровень защиты для уязвимых приложений (например, браузеров, почтовых клиентов, программ просмотра PDF-файлов). Поддерживает Java и помогает улучшить обнаружение и защиту от такого рода уязвимостей.

– Дополнительный модуль сканирования памяти. Модуль HIPS включает в себя дополнительный модуль сканирования памяти, который сканирует выполняющиеся приложения при изменении их состояния с целью обнаружения возможных подозрительных или вредоносных действий. Работает в сочетании с Блокировщиком эксплойтов, чтобы укрепить защиту от вредоносных программ, разработанные для обхода обнаружения антивирусных продуктов за счет использования обфускации и/или шифрования.

– ESET SysRescue Live: Новое поколение утилиты ESET SysRescue, которая позволяет создавать загрузочный диск в виде USB флэш-носителя или CD/DVD-диска с возможностью сканировать и очищать систему от угроз, даже если Windows не может загрузиться.

– Защита от уязвимостей: дополнительные расширенные возможности фильтрации для обнаружения различных типов атак и уязвимостей, которые могут использоваться против компьютера.

– Контроль устройств. Этот модуль пришел на смену модулю защиты съемных носителей, который был доступен в версиях 5 и 6. Он дает возможность сканировать, блокировать или настраивать расширенные фильтры и разрешения, а также выбирать



способ доступа к устройству и его использования для пользователя.

- Улучшенная защиты от фишинга. Добавлено блокирование мошеннических сайтов. Пользователи могут сообщать о подозрительных сайтах и сайтах, ошибочно квалифицированных как мошеннические.

Panda Free Antivirus – бесплатный «облачный» антивирус Panda Dome с проактивными технологиями. Защита компьютера с максимальной производительностью. Эффективная веб-защита, антифишинг и блокировка вредоносных загрузок

Основные особенности Panda Free Antivirus

- Самый быстрый антивирус в обнаружении новых вирусов. Система «коллективного разума» автоматически выявляет новые угрозы за минимальный промежуток времени.

- Удобный для пользователей интерфейс. Отсутствие непрекращающихся уведомлений и трудных вопросов. Бесплатный антивирус Panda работает не раздражая пользователей.

- Обнаруживает больше угроз, чем другие продукты. Серверы Panda Security использует информацию, полученную от миллионов пользователей антивирусных продуктов Panda по всему миру, для автоматического обнаружения и классификации новых видов вредоносных программ, появляющихся каждый день.

Основные возможности Panda Free Antivirus

- Антивирусная и антишпионская защита "из облака" (бесплатный «облачный» антивирус и антишпион).

- Антируткит против скрытых угроз.

- Онлайновая и оффлайновая постоянная защита.

- Поведенческая блокировка новых и неизвестных угроз.

- Анализ поведения запускаемых приложений.

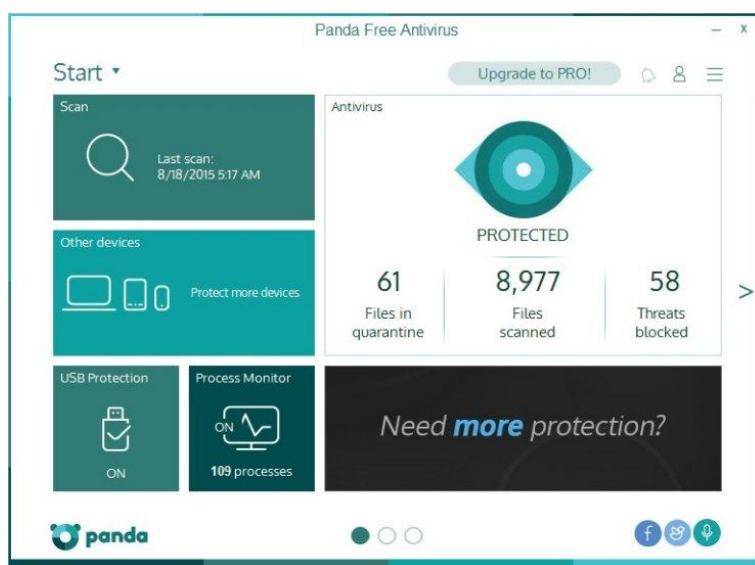
- Улучшенная оффлайновая защита благодаря отключению автозапуска на компьютере.

- URL и веб-фильтрации вредоносных и фишинговых сайтов, блокировка загрузок вредоносного ПО методом "drive-by".

- Менеджер процессов: отчет о запущенных приложениях с URL-мониторингом (информация об интернет-соединении для каждого процесса).

- Расширенные настройки и исключения для экспертов.

- Автоматическая вакцинация USB для защиты от угроз автозапуска - Autorun-вирусов.



Comodo Internet Security – бесплатное комплексное решение безопасности. Включает антивирус и антишпион, фаервол с контент-фильтром, проактивную защиту HIPS, песочницу и Безопасный шоппинг

Основные компоненты Comodo Internet Security Premium:

- Антивирус и антишпион
- Персональный фаервол
- Поведенческий анализ
- Система HIPS
- Система Viruscore
- Контент-фильтр
- Автоматическая песочница
- Виртуальный рабочий стол
- Comodo Secure DNS
- Диск аварийной очистки
- Comodo Cleaning Essentials
- Менеджер процессов
- Веб-браузер Comodo Dragon
- Безопасный шоппинг
- Internet Security Essentials

Основные возможности Comodo Internet Security Premium:

• Антивирус Comodo - проактивный антивирусный движок, который автоматически обнаруживает и устраняет все виды вирусов, интернет-червей и других вредоносных программ. Наряду с мощным сканированием по требованию, по доступу и запланированной проверкой, пользователи CIS теперь могут просто перетащить нужный объект в область на главном экране для запуска мгновенного поиска вирусов.

• Фаервол - полностью настраиваемый брандмауэр с фильтрацией пакетов, который постоянно защищает вашу систему от входящих и исходящих интернет-атак.

• Контент-фильтр – данная функция настраивает Фаервол на фильтрацию доступа к веб-сайтам в соответствии с указанными правилами и профилями.

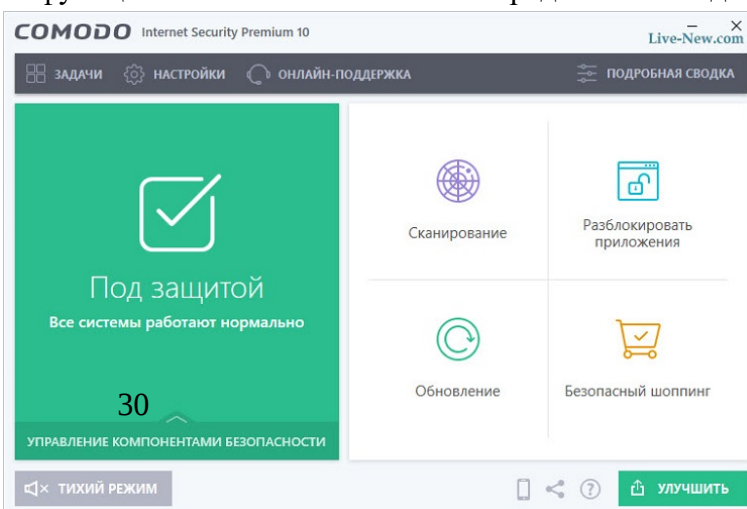
• Comodo Защита+ -набор проактивных технологий защиты, направленных на сохранение целостности, безопасности и конфиденциальности вашей операционной системы и пользовательских данных.

- Поведенческий блокиратор – проверяет подлинность каждого исполняемого файла и процесса запущенного на компьютере и предотвращает их действия, которые могут нанести вред вашему компьютеру. Неизвестные процессы и приложения автоматически запускаются в песочнице, где работают под множеством ограничений, благодаря чему они не могут нанести вред вашему компьютеру. Это дает ненадежным (но безвредным) приложениям свободу действий, в то время для ненадежных (и потенциально опасных) приложений Comodo Internet Security предотвращает повреждение компьютера или данных.

- Viruscore - это система, позволяющая проводить динамический анализ поведения запущенных процессов и вести запись их активности. Viruscore контролирует деятельность процессов, запущенных на вашем компьютере и предупреждает вас, если они пытаются выполнить подозрительные действия.

- HIPS (Host Intrusion Protection) – система предотвращения вторжений, основанная на правилах, которая контролирует активность всех приложений и процессов на вашем компьютере. HIPS Comodo блокирует деятельность вредоносных программ путем прекращения любых действий, которые могут привести к повреждению операционной системы, системной памяти, реестра или личных данных.

• Comodo Secure Shopping – функция Безопасный шоппинг предназначена для защищенных онлайн-покупок и интернет-банкинга. Предлагает всестороннюю защиту от кейлоггеров, перехвата SSL-трафика, отслеживания памяти и удаленного контроля. Инструмент защищает банковские транзакции, номера



кредитных карт и другую персональную информацию.

- Internet Security Essentials - инструмент проверки сертификатов SSL защищает от фальшивых (фишинговых) сайтов, которые пытаются украсть конфиденциальную информацию.

- Comodo Secure DNS - DNS-сервис, который использует список блокировки в режиме реального времени (RBL) вредных веб-сайтов (например, фишинговых сайтов, вредоносных сайтов, шпионских сайтов, сайтов с чрезмерным количеством рекламы и т.д.) и предупреждает вас, когда вы пытаетесь получить доступ к сайтам, содержащим потенциально опасный контент.

- Comodo Rescue Disk - встроенный в Comodo Internet Security мастер позволяет записать загрузочный диск, который позволяет выполнить антивирусное сканирование зараженной системы Windows без загрузки локальной системы.

- Дополнительные утилиты – область Расширенные задачи включает ссылки, которые позволяют установить другие бесплатные продукты безопасности Comodo, включая антивирусную утилиту Comodo Cleaning Essentials для очистки зараженной системы, Comodo Killswitch для мониторинга процессов и активности системы.

- Comodo Dragon - быстрый и универсальный интернет-браузер на основе Chromium, использующий непревзойденный уровень безопасности Comodo.

Comodo Internet Security - отлично сбалансированный программный комплекс для обеспечения безопасности в Сети. В комплекс включены все необходимые утилиты для спокойного серфинга в Интернете, для борьбы с вирусами, троянами, adware-модулями и т.д.

Контрольные вопросы

1. Компьютерный вирус это?
2. Что значит зараженная программа?
3. Какие вредоносные действия выполняет вирус?
4. Какие характерные признаки заражения компьютера вирусами?
5. Какие существуют классификаций компьютерных вирусов?
6. Что делают загрузочные, резидентные, самомодифицирующиеся, троянские вирусы?
7. Какие объекты могут быть заражены вирусом?
8. Что применяется для защиты от вирусов?
9. Две разновидности методов защиты?
10. Какие виды программ для защиты от вирусов существуют и что они позволяют делать?
11. Программы находящиеся с первого по третий уровень защиты?
12. Самые популярные антивирусные пакеты?
13. Какие правила антивирусной безопасности следует соблюдать?
14. Каковы назначение и основные функции Антивируса Касперского?
15. Проверка компьютера по требованию Антивируса Касперского это?
16. Как осуществляется проверка почты Антивируса Касперского?
17. Карантин это?
18. Что входит в состав AVG?

19. Anti-Spyware какую защиту обеспечивает?
20. Каково назначение компонентов Анти-руткит, Resident Shield, Identity Protection, E-mail Scanner, LinkScanner?
21. Как часто обновляются вирусные базы антивирусника?
22. Перечислите главные независимые сертификационные компании.
23. Какие технологии антивирусных программ входят в Norton Security?
24. Какие технологии сетевой защиты включает Norton Security?
25. Перечислите основные возможности Norton Security.
26. Что представляет собой Avira Free Antivirus?
27. Что представляют собой такие возможности Avira Free Antivirus как: Антивирус и антишпион, Облачная защита, Защита от руткитов, Управлением Брандмауэром Windows, Интернет-защита, Родительский контроль, Защита Android-устройств?
28. Что представляет собой McAfee LiveSafe?
29. Каковы основные функции McAfee LiveSafe?
30. Запишите основные возможности McAfee LiveSafe с пояснениями.
31. Антивирус Dr.Web это?
32. Какими защитными компонентами обладает Антивирус Dr.Web и их назначение?
33. Антивирус NOD32 это?
34. Какие основные компоненты Антивируса ESET NOD32?
35. Преимущества ESET NOD32 Antivirus?
36. Основные возможности ESET NOD32 Antivirus?
37. Panda Free Antivirus это?
38. Основные особенности Panda Free Antivirus?
39. Основные возможности Panda Free Antivirus?
40. Comodo Internet Security это?
41. Основные компоненты Comodo Internet Security Premium?
42. Основные возможности Comodo Internet Security Premium?

Практическое занятие № 8

Тема: Форм-фактор системных плат

Цель занятия:

Изучить форм-факторы системных плат.

Перечень используемого оборудования:

Монитор, системный блок, устройства ввода-вывода.

Краткие сведения из теории

Главным узлом, определяющим возможности компьютера, является системная, или материнская (от англ. motherboard) плата. На ней обычно размещаются:

- базовый микропроцессор;
- оперативная память;
- сверхоперативное ЗУ, называемое также кэш-памятью;
- ПЗУ с системной BIOS (базовой системой ввода/вывода);
- набор управляющих микросхем, или чипсетов (chipset), вспомогательных микросхем и контроллеров ввода/вывода;
- КМОП-память с данными об аппаратных настройках и аккумулятором для ее питания;
- разъемы расширения, или слоты (slot);
- разъемы для подключения интерфейсных кабелей жестких дисков, дисководов, последовательного и параллельного портов, инфракрасного порта, а также универсальной последовательной шины USB;
- разъемы питания;
- преобразователь напряжения с 5В на 3,3В для питания процессора (некоторым процессорам требуется также и меньшее напряжение, например 2,2В для AMD K6-3D);
- разъем для подключения клавиатуры и ряд других компонентов.

На платах формата ATX и NLX также находятся разъемы мыши и клавиатуры в стандарте PS/2, разъемы параллельного и последовательного портов (12). На материнской плате также могут находиться микросхемы видеоадаптера и звуковой платы (т. н. платы All-In-One), а также контроллера SCSI. На платах для 286, 386 и 486-х процессоров также могут находиться специальные разъемы для установки микросхем математического сопроцессора и/или процессора OverDrive.

Для подключения индикаторов, кнопок и динамика, расположенных на корпусе системного блока, на материнской плате имеются специальные миниатюрные разъемы-вилки. Подобные же разъемы служат как контакты для перемычек (jumpers) при задании аппаратной конфигурации системы. Если на системной плате сосредоточены все элементы, необходимые для его работы, то она называется All-In-One. У большинства персональных компьютеров системные платы содержат лишь основные функциональные узлы, а остальные элементы расположены на отдельных печатных платах (платах расширения), которые устанавливаются в разъемы расширения. Например, устройство формирования изображения на экране монитора - видеоадаптер пока чаще всего располагается на отдельной плате расширения - видеокарте.

Важной характеристикой материнской платы является ее форм-фактор, определяющий ее геометрические размеры, расположение разъемов расширения и процессора, точек крепления платы, а также тип разъема питания платы и питающие напряжения. Поэтому форм-фактор платы определяет также используемый тип корпуса и

блока питания. В настольных системах с корпусами типа desktop или tower чаще всего используются два несовместимых между собой форм-фактора плат (а, соответственно, и корпусов ПК) - AT и ATX.

AT - форм-фактор плат на основе процессоров от 286 до 486, значительной части Pentium и некоторой части Pentium Pro. Для новых процессоров (Pentium II, Celeron, K6-3D) AT-платы практически не выпускаются. Наиболее распространенный формат baby AT имеет размеры 12х9,6 дюйма, или 305х244 мм. Существуют AT-платы большего (Full-size AT) и меньшего (3/4 baby AT) размера. Если в какой-либо корпус можно установить AT-плату большего размера, то в этот же корпус можно установить любую AT-плату меньшего размера. Для питания AT-платы требуются напряжения $\pm 12\text{В}$ и $\pm 5\text{В}$.

ATX, предложенный Intel форм-фактор платы, имеет такие же размеры, как и baby AT, но разъемы расширения расположены параллельно короткой стороне платы, процессор сдвинут в сторону вентилятора блока питания, разъемы для подключения дисководов и жестких дисков расположены вблизи соответствующих устройств, что позволяет применять короткие интерфейсные кабели и повышает надежность передачи данных. Продуманная конструкция позволяет устанавливать во все разъемы расширения платы полной длины, обеспечивает удобный доступ к ним и хорошее охлаждение. Подавляющее большинство плат для семейства процессоров P6 (Pentium II, Celeron, Xeon) выпускаются в формате ATX. В частности, Intel прекратила выпуск плат формата AT уже для процессоров Pentium MMX, перейдя к выпуску плат формата ATX. В этом формате используется новый 20-контактный разъем питания, через который подаются напряжения $\pm 12\text{В}$, $\pm 5\text{В}$ и $+3,3\text{В}$, а также сигналы soft power, позволяющие программно производить включение и выключение питания компьютера. Все необходимые разъемы для подключения внешних устройств расположены на задней панели платы.

В корпусах уменьшенного размера типа slim используются платы старого (LPX) или (NLX) форм-факторов, которые соотносятся между собой так же, как AT и ATX.

Подавляющее большинство современных материнских плат являются так называемыми "зелеными" (green motherboard), т. е. аппаратно и программно поддерживают несколько экономичных режимов с пониженным энергопотреблением.

Материнская плата, помимо разъемов для установки процессора, оперативной памяти и плат расширения, несет на себе множество дополнительных электронных компонентов, которые обеспечивают нормальное функционирование системы и во многом определяют потребительские качества материнской платы и компьютера в целом.

К таким компонентам прежде всего относятся:

- набор микросхем логики платы (чипсет, или chipset), обеспечивающий поддержку процессора, памяти и большинства интерфейсов ввода/вывода;
- кэш-память (первого, второго или третьего уровня);
- контроллер клавиатуры;
- контроллер ввода-вывода, обслуживающий дисководы гибких дисков и порты ввода/вывода;

дополнительные интегрированные контроллеры (видео, сетевой, SCSI, звук и т. п.).

Все чаще встречаются высокоинтегрированные решения, когда часть перечисленных контроллеров реализована в рамках чипсета (например, Cyrix MediaGX).

Вспомогательные микросхемы и устройства.

Микропроцессор, чипсет, память, контроллеры, порты ввода/вывода и разъемы различных шин еще не исчерпывают конструкцию материнской платы. Для создания полной системы необходимы также вспомогательные микросхемы, такие как преобразователь напряжения, тактовый генератор, таймер, различные контроллеры, буферы адреса и данных и т. п. Функции многих из них интегрированы в чипсете, однако некоторые компоненты в любом случае остаются снаружи.

К дополнительным компонентам материнской платы можно отнести температурные датчики LM78, собирающие информацию о температуре процессора, материнской платы, скорости вращения вентилятора и др.

Системную шину можно упрощенно представить как совокупность сигнальных линий, объединенных по их назначению (данные, адреса, управление), которые также имеют вполне определенные электрические характеристики и протоколы передачи информации. Основной обязанностью системной шины является передача информации между процессором (или процессорами) и остальными электронными компонентами компьютера. По этой шине осуществляется не только передача информации, но и адресация устройств, а также происходит обмен специальными служебными сигналами. Используемые в настоящее время шины отличаются по разрядности, способу передачи сигнала (последовательные или параллельные), пропускной способности, количеству и типу поддерживаемых устройств, а также протоколу работы. Как правило, шины ПК можно представить в виде некой иерархической структуры - шинной архитектуры. Особенностью современных ПК является наличие шины ISA, унаследованной от самых первых моделей IBM PC. Кроме нее, в ПК применяются шины EISA, MCA, VLB, PCI, PCMCIA (CardBus) и AGP.

Стандарт AT. В данном стандарте на тыльную часть панели системного блока выводится один разъем DIN-5S для подключения клавиатуры. Разъемы портов устанавливаются либо на картах расширения, либо на переходных планках и соединяются с системной платой шлейфами. Разъем подключения блока питания выполнен в виде установленных в ряд 12-ти ножевых штырей, к которым подключаются два разъема: P8 и P9 (по 6 гнезд в каждом).

Стандарт ATX. Системные платы стандарта ATX имеют одноключевой внутренний разъем источника питания, на тыльной стороне встроенную панель разъемов ввода вывода размеров 15,9 x 4,4 см. Данные платы отличаются более удобным расположением элементов, а также наличием разъемов USB.

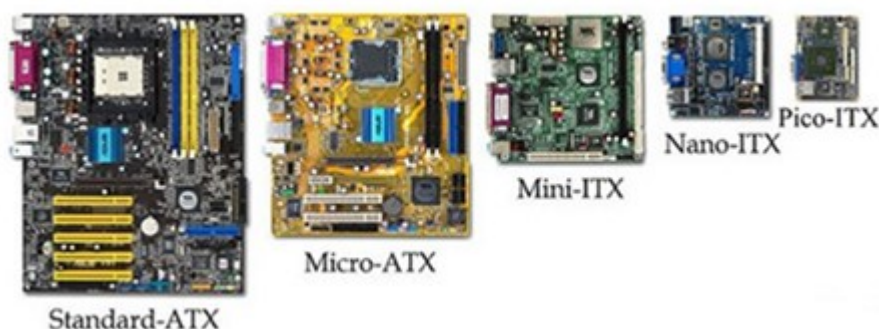
Стандарт NLX. Плата данного стандарта с помощью специального разъема (NLX Riser Connection) устанавливается в соединительную ризер-карту NLX. Через ризер-карту к системной плате подается питание, устанавливаются платы расширения и подключается FDD и IDE устройства. На плате размещены разъемы клавиатуры, мыши, LPT, COM, USB. Она может быть легко заменена другой платой с другим процессором или несколькими процессорами.

Каждой материнской плате выдается свой номер организацией Federal Communication Commission (FCC). На плате имеется маркировка, расположенная на задней панели.

Например: FCC ID i670MNI56k,

Первые 3 символа - производитель, остальные - устройство. Узнать производителя можно на сайте FCC введя маркировку в поиск.

Форм фактор материнской платы – это определенный стандарт, благодаря которому определяется размер системной платы, ее способ крепления в корпусе ПК, наличие и расположение портов и интерфейсов шин, а также разъемов ЦП и оперативной памяти.



В настоящее время существует несколько форм-факторов системных плат:

- ATX – самая большая по величине материнская плата, которая имеет много слотов и удобное расположение портов ввода — вывода.
- EATX – это системная плата ничем не отличающаяся от ATX, кроме размеров, которые составляют 30,5 см. х 33 см.
- MicroATX – это уменьшенный вариант материнской платы ATX. Она была разработана для ПК, которым не требуется изменение конфигурации, поэтому она имеет 4 слота расширения. Как правило, такие платы оснащены 24 пиновым разъемом питания и имеют размеры, 24,5 см. х 24,5 см.
- BTX – материнская плата, разработанная для создания небольших системных блоков. Тем не менее, она имеет 7 слотов расширения при размерах 26.7 х 32.5 см.
- Micro BTX является уменьшенной копией «материнки» BTX, которая имеет 4 слота расширения и размеры 26.7 х 26.4 см.
- MiniITX – самая маленькая, из современных системных плат для компьютера. Ее размеры составляют 17 см на 17 см.
- SSIEEB и SSICEB. Материнские платы этих форм-факторов применяются для создания серверов. Размеры составляют: 30.5 х 33.0 см. и соответственно 30.5 х 25.9 см.

Содержание отчета

1. Наименование и цель работы;
2. Теоретические сведения;
3. Выполнение заданий;
4. Ответы на контрольные вопросы;
5. Вывод о проделанной работе.

Контрольные вопросы

1. Типы электронных плат управления работой компьютера?
2. Основные характеристики материнской платы?
3. Устройства, расположенные на материнской плате, их характеристики?
4. Характеристики шин – тип подключаемых устройств, скорость передачи данных?
5. Контроллеры и адаптеры, их назначение и основные характеристики?
6. Какие стандарты плат существуют?
7. Форм фактор материнской платы?
8. Какая плата считается стандартом?
9. Размеры EATX?
10. В чем отличие MicroATX и Micro BTX?
11. Где применяется MiniITX?
12. Какие платы применяются в серверах?
13. Стандарт NLX?
14. Что означает аббревиатура FCC?
15. Приведите пример материнской платы?

Практическое занятие № 9

Тема: UEFI BIOS материнской платы

Цель работы:

Изучить BIOS материнской платы (AMI, Award). Получить практические навыки работы с UEFI BIOS материнской платы.

Перечень используемого оборудования:

Монитор, системный блок, устройства ввода-вывода.

Краткие сведения из теории

BIOS (basic input/output system) - базовая система ввода-вывода - это встроенное в компьютер программное обеспечение, которое ему доступно без обращения к диску. На PC BIOS содержит код, необходимый для управления клавиатурой, видеокартой, дисками, портами и другими устройствами.

Обычно BIOS размещается в микросхеме ПЗУ (ROM), размещенной на материнской плате компьютера (поэтому этот чип часто называют ROM BIOS). Эта технология позволяет BIOS всегда быть доступным, несмотря на повреждения, например, дисковой системы. Это также позволяет компьютеру самостоятельно загружаться. Поскольку доступ к RAM (оперативной памяти) осуществляется значительно быстрее, чем к ROM, многие производители компьютеров создают системы таким образом, чтобы при включении компьютера выполнялось копирование BIOS из ROM в оперативную память. Задействованная при этом область памяти называется Shadow Memory (тенивая память).

В настоящее время, почти все материнские платы комплектуются Flash BIOS, BIOSom, который в любой момент может быть перезаписан в микросхеме ROM при помощи специальной программы.

BIOS PC стандартизирован, поэтому, в принципе менять его, также как, например, операционные системы нет необходимости. Дополнительные возможности компьютера можно использовать только использованием нового программного обеспечения.

BIOS, который поддерживает технологию Plug-and-Play, называется PnP BIOS. При использовании этой технологии BIOS должен быть обязательно прошит во Flash ROM.

Определение типа микросхемы ПЗУ. Определить тип микросхемы ПЗУ, установленной на материнской плате, несложно. Для этого необходимо посмотреть на маркировку чипа ROM (28 или 32-контактная микросхема с наклейкой производителя BIOS), отодрав наклейку. Маркировка означает следующее (xxx означает три произвольных цифры):

28Fxxx - 12V Flash память

29Cxxx - 5V Flash память

29LVxxx - 3V Flash memory (паритет)

28Cxxx - EEPROM, почти то же, что и Flash память

27Cxxx - с окошком. EPROM: только для чтения, требует программатор для записи и ультрафиолетовую лампу для стирания

PH29EE010: SST ROM Чип - перепрошиваемый

29EE011: Winbond чип - 5V Flash память

29C010: Atmel Chip - 5V Flash память

Любые другие микросхемы, не имеющие окошка с маркировкой, не начинающейся с цифр 28 или 29, являются, скорее всего не Flash-памятью. Если же на микросхеме есть окошко - это верный признак того, что это не Flash.

Перепрошивка новых версий BIOS. Существует несколько причин, по которым это приходится делать. Основная из них - Windows не всегда хорошо конфигурируется, если используются старые версии BIOS.

В настоящее время используются жесткие диски объемом более 528Мбайт. Для работы такого диска в системе необходимо поддержка LBA со стороны BIOS. Если BIOS не поддерживает LBA, то для работы с большими жесткими дисками приходится применять специальные утилиты. Применение таких утилит вызовет работу Windows в compatible mode, что отрицательно сказывается на быстродействии.

Полная поддержка Plug-and-Play со стороны Windows возможна только в случае применения PnP BIOS. Это очень веская причина для перепрошивки BIOS.

Кроме вышеуказанного, в новых версиях BIOS исправляются мелкие ошибки и недоработки. Новые версии могут содержать новые возможности, как то загрузка с CD ROM, SCSI перед IDE и т.п.

Описание основных настроек Setup BIOS приведено в приложении 1.

Примечание.Изменение настроек в CMOS может привести к выходу компьютера из строя. В связи с этим примите следующие меры:

не вносите никаких изменений в настройки без указания преподавателя;
записью на отдельном листе бумаги четко фиксируйте все параметры до их изменения и после;

по окончании работы закройте программу Setup BIOS без сохранения внесенных изменений (Exit without saving). Перед закрытием программы обратитесь к преподавателю для контроля.

Порядок выполнения:

Включите компьютер выключателями находящимися на системном блоке.

Включите монитор (если монитор компьютера имеет питание, отдельное от системного блока).

При появлении информации на экране нажмите клавишу DELETE - произойдет запуск программы SETUP и откроется основное меню.

С помощью клавиш управления курсором выберите пункт меню STANDARD CMOS SETUP (Стандартные настройки микросхемы CMOS).

В открывшемся окне проверьте установку системных часов, системного календаря, количество и объем жестких дисков.

Вернитесь в предыдущее меню с помощью клавиши ESC.

Выберите пункт BIOS FEATURES SETUP (Настройки параметров BIOS).

В открывшемся окне проверьте:

- с какого диска начинается запуск компьютера. Последовательность запуска задается в пункте BOOT SEQUENCE. С помощью клавиш PAGE UP и PAGE DOWN просмотрите и отметьте все возможные для данной версии CMOS варианты запуска. Особое внимание обратите на вариант запуска, начинающегося с жесткого диска C: (он используется при штатной работе), и на вариант запуска, начинающегося с гибкого диска A:; - он используется при восстановлении работоспособности компьютера, если загрузка с жесткого диска по каким-то причинам невозможна.

- состояние защиты компьютера от несанкционированного доступа: задается пункт Security Option. Setup - отключено (для режима настройки), System - включено (для штатной работы системы).

Вернитесь в предыдущее меню нажатием клавиши ESC.

Выберите пункт IDE HDD AUTO DETECT (Автоматическое определение жестких дисков). Обратите внимание на порядок тестирования дисков.

Завершите работу с программой SETUP без сохранения результатов изменения. Для этого нажмите клавишу ESC и при получении запроса подтвердите выход без сохранения изменений нажатием клавиши Y (Yes - Да).

Ошибки при тестировании и загрузке системы могут оказаться настолько серьезными, что компьютер окажется неспособным даже включить видеосистему и вывести на экран монитора сообщение о типе ошибки. Именно для таких случаев практически во всех BIOS предусмотрена система звукового оповещения о характере ошибки, найденной при процедуре тестирования. Таким образом, услышав, что компьютер «пищит», и определив последовательность выдаваемых им звуковых сообщений, пользователь сможет распознать разновидность ошибки на слух, даже не смотря на экран монитора. Невозможно знать все виды звуковых сигналов и их коды наизусть, так как у разных производителей они могут отличаться.

Поэтому каждому пользователю необходимо знать коды сигналов, соответствующих BIOS, установленной на материнской плате именно его компьютера [6]. Для каждого варианта BIOS существует таблица, в которой приводится расшифровка сигналов ошибок BIOS. Для того чтобы расшифровка сигналов была успешной, обычно требуется тщательный подсчет того, сколько раз компьютер «пищит», и если вы пропустите один-единственный писк, то это может привести к тому, что расшифровка окажется неудачной и ее придется начинать сначала.

Коды ошибок AMI BIOS

По сравнению со звуковыми сигналами других производителей BIOS звуковые сигналы BIOS AMI отличаются большим разнообразием. В большинстве случаев эти звуковые сигналы позволяют определить неисправность еще на стадии загрузки компьютера и процедуры проверки оборудования POST. Как правило, для определения того компонента, который вызывает проблему, необходимо лишь сосчитать количество издаваемых системным динамиком сигналов. В таблице приведены звуковые сигналы и описания ошибок, издаваемые BIOS AMI. Во всех случаях указано количество сигналов и их тип (длинные/короткие).

Таблица 1 - Описание значений звуковых сигналов BIOS AMI

Сигнал	Описание ситуации
Нет сигналов	Данная ситуация, возможно, самая неприятная из тех, с которыми может столкнуться пользователь. Как правило, она означает, что питание на материнскую плату не подается или BIOS вообще неисправна. Отсутствие питания на материнской плате обычно означает либо неисправность/обрыв кабеля питания, либо неисправность блока питания компьютера.
Один короткий	Один-единственный короткий сигнал - это тот самый сигнал, который привыкли слышать каждый раз при запуске ПК все пользователи. Он означает, что никаких ошибок и проблем в ходе проверки оборудования не было обнаружено, и компьютер может загружаться дальше.
Длинный непрерывный	Такой сигнал означает, что неисправен блок питания ПК. Тем не менее, в отличие от ситуации с полным отсутствием сигналов, в данном случае питание на материнскую плату поступает, однако его параметры не соответствуют номиналу.
Два коротких	Этот сигнал говорит об ошибке в оперативной памяти. Данная ошибка может свидетельствовать как о неисправности самих модулей памяти, так и всего лишь о том, что какой-то из модулей плохо вставлен в слот.
Три коротких	Данная разновидность сигнала тоже свидетельствует об ошибке в ОЗУ. Но это ошибка довольно специфическая и редко встречающаяся - это ошибка в первых 64 КБ оперативной памяти.

Четыре коротких	Данный сигнал говорит о неисправности системного таймера. К счастью, подобный вид неисправности встречается нечасто, однако нередко единственным способом его устранения является замена всей материнской платы.
Пять коротких	Подобным образом BIOS сообщает пользователю о неисправности сердца персонального компьютера - центрального процессора. Однако далеко не всегда данная неисправность может быть связана с дефектом самого чипа процессора. Часто для устранения проблемы достаточно бывает проверить надежность установки процессора в сокет.
Шесть коротких	Это сообщение указывает на неисправность контроллера клавиатуры, а также на отсутствие самой клавиатуры. Очень часто для исправления подобной ситуации достаточно бывает проверить контакт в разъеме клавиатуры на системном блоке.
Семь коротких	Подобный набор звуков указывает на довольно серьезную неисправность - а именно, на отказ системной платы. Впрочем, иногда ошибка может исчезнуть после проверки контактов кабеля питания на системной плате.
Восемь коротких	Сообщение BIOS, свидетельствующее об отказе видеопамяти. Но здесь, как и при многих других ошибках, иногда проблема может заключаться всего лишь в плохом контакте - в данном случае, между слотом материнской платы и видеокартой.
Девять коротких	В данном случае BIOS сигнализирует об ошибке контрольной суммы памяти BIOS. Причины данного явления могут быть разными - ошибка в новой версии BIOS или случайный сбой в памяти CMOS. Часто проблему можно исправить перепрошивкой BIOS.
Десять коротких	При таком наборе сигналов BIOS сообщает об ошибке записи в память CMOS. Как правило, эта ошибка серьезнее предыдущей и часто требует вмешательства специалиста из сервисной мастерской.
Одиннадцать коротких	Если вы сумели досчитать до одиннадцати и не сбиться, то знайте, что BIOS сигнализирует вам об ошибке расположенной на материнской плате кэш-памяти. В этом случае обычно требуется замена микросхемы кэш-памяти.

Коды ошибок AWARD BIOS

По сравнению с сигналами AMI звуковые сигналы Award BIOS не столь разнообразны, однако в подавляющем большинстве случаев их набора вполне хватает для кодирования всех возможных ошибок материнской платы. Отличительными особенностями Award BIOS является широкое использование длинного сигнала, а также использование таких видов сигналов, как непрерывные и постоянно повторяющиеся звуковые сигналы. В таблице приведены звуковые сигналы Award и соответствующие им проблемные ситуации, а также возможные способы их решения.

Таблица 2 - Описание значений звуковых сигналов Award BIOS

Сигнал	Описание ситуации
Отсутствие сигналов	Это может свидетельствовать о неисправности как блока питания, так и самой материнской платы. В некоторых случаях неисправным может оказаться системный динамик материнской платы. Для исправления ситуации можно попробовать проверить контакт силового кабеля, идущего от блока питания к материнской плате.
Один короткий	Подобный сигнал пользователь слышит чаще всего. Он означает, что процедура проверки оборудования BIOS POST прошла успешно, и загрузка компьютера может быть продолжена.

Два коротких	Это сообщение в Award BIOS зарезервировано для тех случаев, когда ошибка не является серьезной и позволяет компьютеру работать в обычном режиме. Подробное текстовое сообщение о сути ошибки при этом выводится на экран монитора. Дальнейшие действия пользователя по устранению ошибки обычно зависят от разновидности ситуации. Например, причиной появления ошибки может являться разрядившаяся батарейка памяти CMOS. В подобном случае ее необходимо заменить.
Короткий повторяющийся	Сигнал указывает на неисправность блока питания или повреждение цепей питания. Для исправления ситуации можно попробовать проверить контакты проводов, идущих от блока питания к материнской плате.
Один длинный и один короткий	Сообщение, говорящее о неисправности оперативной памяти или вообще об ее отсутствии. Если имеет место последний вариант, то модули оперативной памяти необходимо установить, а если память уже присутствует в слотах, то можно попробовать переустановить микросхемы ОЗУ. Если же это не помогает, то, скорее всего, проблема заключается в неисправности модуля памяти.
Длинный повторяющийся	Ошибка, похожая на предыдущую и свидетельствующая о проблемах с ОЗУ. Данная ошибка чаще всего возникает при неправильной установке модулей памяти. Для исправления ситуации следует проверить, правильно ли установлены модули, и если нет, то переустановить их.
Один длинный и два коротких	Подобные звуковые сигналы сообщают об ошибке видеокарты. Часто в подобной ситуации проблема заключается всего лишь в плохо установленной в слоте расширения плате графического ускорителя, хотя причиной может быть и неисправность микросхемы видеокарты.
Три длинных	Это сообщение означает, что BIOS выявила ошибку контроллера клавиатуры. Как и в других случаях, тут можно попробовать проверить соединение клавиатуры с системным блоком. В некоторых случаях сигнал пропадает после повторной перезагрузки компьютера. Если же клавиатура работает, но сигналы все равно остаются, то их обычно можно отключить при помощи специальной опции BIOS.
Один длинный и три коротких	Звуковое сообщение, как и предыдущее, свидетельствующее об ошибке клавиатуры. Но, в отличие от предыдущей ошибки, в данном случае сигналы говорят о том, что ошибка проявляется в несколько другой ситуации - BIOS обнаружила клавиатуру, но не смогла к ней обратиться.
Один длинный и девять коротких	Сигналы говорят об ошибке постоянной памяти CMOS. Данный сбой может быть как случайным, исчезающим после перезагрузки ПК, так и быть следствием неисправности микросхемы памяти CMOS, а также неправильной перепрошивки BIOS.

Коды ошибок Phoenix BIOS

Издаваемые Phoenix BIOS сигналы кодируются по несколько иному принципу, нежели звуковые сигналы AMI BIOS или Award BIOS. Как правило, Phoenix BIOS выдает несколько серий коротких сигналов, которые отделены друг от друга продолжительными паузами. По комбинации серий сигналов и пауз можно понять, к какому типу неисправности относится данное звуковое сообщение. Большинство сообщений состоит из трех серий сигналов, но есть сообщения, кодируемые четырьмя или двумя сериями. Набор ошибок, которые кодируют звуковые сигналы, у Phoenix BIOS очень велик и охватывает почти все возможные ситуации, связанные с неисправностями аппаратной части компьютера. В таблице описаны различные сообщения об ошибках Phoenix BIOS. Цифры обозначают количество звуков в серии, а дефис - паузу между сериями.

Таблица 3 - Описание значений звуковых сигналов Phoenix BIOS

Сигнал	Описание ситуации
--------	-------------------

Отсутствие сигнала в	Как и в аналогичных случаях с другими BIOS, если сигналы полностью отсутствуют, то это означает, что питание на материнскую плату не подается, или сама BIOS является неисправной. Чтобы удостовериться, что питание на плату не подается лишь по случайной причине, а не вызвано какой-либо неисправностью оборудования, следует убедиться в том, что блок питания включен.
1-1-3	Подобные сигналы свидетельствуют об ошибке чтения/записи памяти CMOS.
1-1-4	Эта ситуация схожа с предыдущей, но она означает ошибку контрольной суммы памяти CMOS. В обоих случаях для решения проблемы можно попытаться перезагрузить компьютер, а если это не помогает, то переустановить батарейку CMOS.
1-2	Сигналы свидетельствуют о неисправности в одном из дополнительных контроллеров, оснащенных собственной BIOS.
1-2-2-3	Сигналы, характерные для ошибки контрольной суммы BIOS.
1-2-1	Данные сигналы говорят об общей ошибке материнской платы.
1-2-2	Общая ошибка контроллера DMA.
1-2-3	Также связанная с контроллером DMA ошибка, возникающая при неудаче проверки чтения/записи в контроллере DMA.
1-3-1	Неисправность микросхемы регенерации модуля ОЗУ.
1-3-3 или 1-3-	Сообщения об ошибке, связанной с проверкой первых 64 килобайт оперативной памяти.
1-4-2, или 4-3- 1, или 1-	Ошибка тестирования ОЗУ. Для исправления данной проблемы можно попробовать переустановить модули ОЗУ.
1-4-3	Невозможно инициализировать системный таймер.
2-1-2-3	Довольно специфическая ошибка, связанная с проверкой информации об авторском праве в ROM BIOS.
2-2-3-1	Данное сообщение генерируется в случае ошибки при тестировании непредвиденных прерываний.
3-1-1 и 3-1-2	Ошибка в контроллере DMA, возникающая при попытке обратиться, соответственно, к 1 и 2 каналам DMA).
3-1-2	Неисправность контроллера прерываний.
3-2-4, или 4-2 3, или 1-	Невозможно инициализировать контроллер клавиатуры. Для исправления ситуации можно попробовать заново подключить клавиатуру и перезагрузить компьютер.
3-3-4	Неисправность графической карты.
3-4-1	Неисправность или отсутствие монитора. При появлении данной ошибки следует проверить, подключен ли монитор к компьютеру.
3-4-2	Ошибка BIOS графической карты.
4-4-2	Тестирование успешно завершено. Следует обратить внимание на тот факт, что у Phoenix BIOS подобное сообщение имеет совершенно другой вид, нежели у других BIOS.
4-2-4	Ошибка, связанная с проверкой работы ЦП в защищенном режиме.
4-3-4	Неисправность часов реального времени. Для ликвидации данной проблемы можно попробовать перезагрузить компьютер.
4-4-1 и 4-4-2	Ошибки при проверке, соответственно, последовательного и параллельного порта. Проверьте, не подключены ли к данным портам какие-либо устройства, поскольку они также могут вызывать данное сообщение.

4-4-3	Неисправность математического сопроцессора. В настоящее время сопроцессоры практически не используются, поэтому шанс встретиться с подобным сообщением крайне невелик.
-------	--

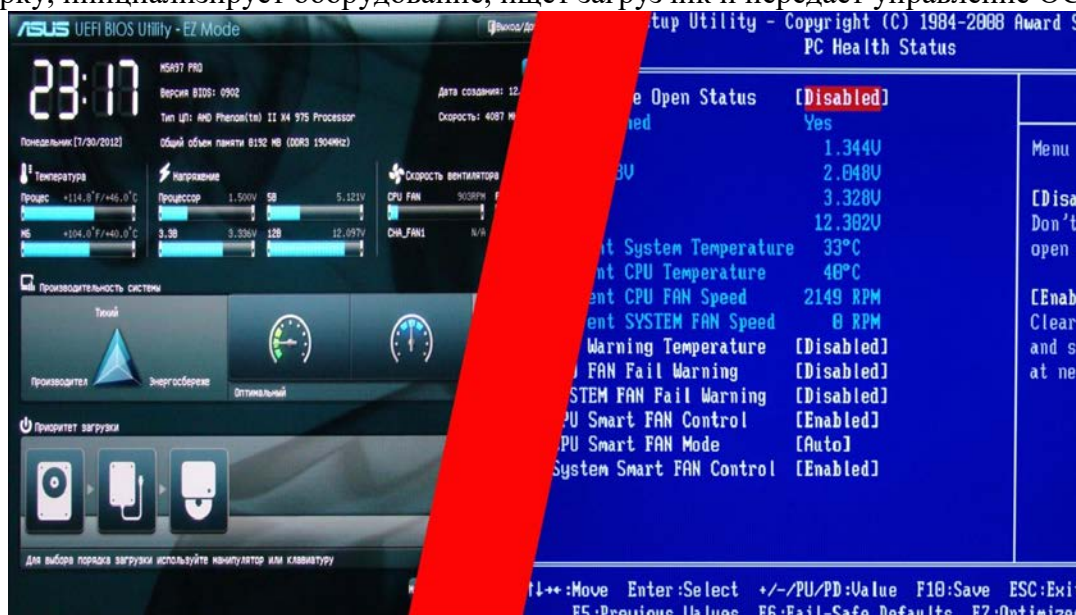
Кроме того, сообщения, начинающиеся с двух коротких сигналов и имеющие три серии сигналов, также связаны с ошибкой записи (или чтения) в определенные разряды (с 0-го по 15-й) первых 64 КБ ОЗУ. Подобные ситуации возникают довольно редко.

Также в некоторых случаях Phoenix BIOS использует и сигналы других типов. Например, длинные непрерывающиеся сигналы свидетельствуют о неисправности системной платы. Если системный динамик воет подобно сирене, причем звук периодически меняет свою частоту с высокой на низкую и наоборот, то речь идет о серьезной неисправности видеокарты. Постоянный непрерывающийся звук означает отсутствие кулера центрального процессора или его выход из строя.

Унифицированный расширяемый микропрограммный интерфейс (UEFI)

UEFI - система ввода-вывода вычислительных машин является недооцененной угрозой в иерархии ЭВМ. Атаки на систему ввода-вывода сложны и используются редко, но имеют чрезвычайно тяжелые последствия, в связи с тем, что специалисты не знают, в каком месте файла искать утечку, и для анализа им требуется всесторонние «ручные» тесты, анализ исполняющего файла и анализ данных объемных спецификаций, представленные официальными производителями. Все эти действия занимают достаточно долго времени, и в период от получения технического задания до финального отчета о проверке файла на уязвимости программный продукт, комплектующее ЭВМ и технологии устаревают. Вследствие чего заказчик получает устаревшие ЭВМ, что влечет за собой потерю времени, технологий и возможностей эксплуатации ЭВМ и личные потери заказчика в целом.

Unified Extensible Firmware Interface - стандартизированный расширяемый интерфейс встроенного программного обеспечения. Интерфейс-подложка между компонентами компьютера и операционной системой. По сути тот же BIOS, только несколько улучшенный. Делает все то же самое, что и стандартный BIOS, т.е. проводит проверку, инициализирует оборудование, ищет загрузчик и передает управление ОС.



Интерфейс UEFI и BIOS

Основные преимущества UEFI перед BIOS

Времени на загрузку уходит гораздо меньше, что достигается при помощи параллельной инициализации, в отличие от BIOS, который инициализирует все компоненты последовательно. Примечательно и то, что операционная система сможет

использовать драйвера UEFI, а не загружать свои собственные. Например, если вы нетребовательны к графической подсистеме, то можете не скачивать, а следовательно и не устанавливать новые драйверы, а использовать драйверы, предлагаемые UEFI.

Преимущество заключается в том, что можно производить загрузку с дисков объемом более 2 Тб. Дело в том, что BIOS для загрузки использовал MBR (Main Boot Record) - основная загрузочная запись, которая может адресовать только 2 Тб пространства, UEFI же использует GPT (Guid Partition Table) - это другой стандарт формата размещения разделов на физическом жестком диске, который позволяет адресовать целых 9 ЗБ (Зеттабайт).

UEFI также обзавелся графическим интерфейсом с поддержкой мыши, для кого это плюс, а для кого и минус. Появились некоторые встроенные программы, например, браузер. Таким образом, UEFI стал представлять не только интерфейс между операционной системой и аппаратными компонентами, но и некое подобие Live CD. Поддержка криптографии и прочих методов защиты.

Содержание отчета

1. Наименование и цель работы;
2. Теоретические сведения;
3. Выполнение заданий;
4. Ответы на контрольные вопросы;
5. Вывод о проделанной работе.

Контрольные вопросы

1. Что такое BIOS?
2. Как определить, что установленный на материнской плате BIOS, прошит во Flash ROM?
3. Зачем необходима перепрошивка новых версий BIOS?
4. Где можно скачать новые версии BIOS?
5. Что делать, если производителя и название материнской платы определить не удастся?
6. Как перепрошить Flash BIOS?
7. Почему прошивальщик Award BIOS выдает сообщение «Insufficient memory»?
8. Что будет, если запортировать BIOS или прошить неправильную версию?
9. Как восстановить поврежденный BIOS?
10. Что такое PROM, EPROM и EEPROM и чем они отличаются?
11. Как сбросить установки BIOS (включая пароль) в значения по умолчанию из DOS?
12. Как подобрать (снять) пароль на Setup (загрузку)?
13. Как аппаратно сбросить CMOS (вместе с паролями)?
14. Что означают аварийные сигналы, выдаваемые AMI BIOS при загрузке?

15. Что означают аварийные звуковые сигналы, выдаваемые Award BIOS при загрузке?
16. Что означают аварийные звуковые сигналы, выдаваемые Phoenix BIOS при загрузке?
17. Что такое UEFI?
18. Какие основные преимущества UEFI перед BIOS?
19. Может ли операционная система использовать драйвера UEFI?
20. Какое ограничение на объем жесткого диска в BIOS?
21. Почему UEFI может использовать диски большего объема?
22. Что такое MBR?
23. В чем отличие GPT от MBR?
24. Какие программы и средства защиты присутствуют в UEFI?

Практическое занятие № 10

Тема: Изучение работы микропроцессора.

Цель занятия:

Закрепить теоретические знания и развить навыки работы с микропроцессорами, а также понять их структуру и принципы функционирования.

Краткие сведения из теории

Основные понятия и структура микропроцессора

Микропроцессор — это центральный элемент вычислительной системы, который выполняет основные вычислительные операции и управляет работой всей системы.

Основные компоненты микропроцессора включают:

1. **Арифметико-логическое устройство (АЛУ):** Выполняет арифметические и логические операции.
2. **Управляющее устройство (УУ):** Управляет выполнением команд и синхронизацией работы всех компонентов микропроцессора.
3. **Регистры:** Временные хранилища данных и адресов, используемые для выполнения операций.
4. **Шина данных:** Линии, по которым передаются данные между компонентами микропроцессора и другими устройствами.
5. **Шина адреса:** Линии, по которым передаются адреса памяти.
6. **Шина управления:** Линии, по которым передаются управляющие сигналы.

Основные этапы работы микропроцессора

1. **Считывание команды:** Микропроцессор считывает команду из памяти.
2. **Декодирование команды:** Управляющее устройство определяет, какую операцию нужно выполнить.
3. **Выполнение команды:** АЛУ выполняет необходимую операцию.
4. **Запись результата:** Результат операции записывается в память или регистр.

Типы микропроцессоров

1. **CISC (Complex Instruction Set Computing):** Микропроцессоры с набором сложных команд. Пример: Intel x86.
2. **RISC (Reduced Instruction Set Computing):** Микропроцессоры с набором простых команд. Пример: ARM.

Задания для выполнения студентами

Задание 1: Изучение структуры микропроцессора

1. Нарисуйте блок-схему микропроцессора, указав основные компоненты.
2. Опишите функции каждого компонента микропроцессора.

Задание 2: Анализ работы микропроцессора

1. Опишите этапы выполнения команды микропроцессором.
2. Приведите пример команды и опишите, как она выполняется микропроцессором.

Задание 3: Сравнение CISC и RISC микропроцессоров

1. Опишите основные различия между CISC и RISC микропроцессорами.
2. Приведите примеры микропроцессоров каждого типа и их применения.

Задание 4: Практическое задание на работу с микропроцессором

1. Используя симулятор микропроцессора (например, Logisim или другие), напишите и выполните простую программу, которая выполняет арифметическую операцию (например, сложение двух чисел).
2. Опишите шаги выполнения программы и результаты.

Задание 5: Анализ производительности микропроцессора

1. Изучите характеристики современного микропроцессора (например, Intel Core i7 или AMD Ryzen).
2. Опишите основные параметры, влияющие на производительность микропроцессора (тактовая частота, количество ядер, кэш-память и т.д.).

Контрольные вопросы по теме

1. Что такое микропроцессор и какие функции он выполняет?
2. Какие основные компоненты входят в структуру микропроцессора?
3. Опишите этапы выполнения команды микропроцессором.
4. Какую роль играет арифметико-логическое устройство (АЛУ) в микропроцессоре?
5. Какую роль играет управляющее устройство (УУ) в микропроцессоре?
6. Что такое регистры и какую роль они играют в работе микропроцессора?
7. Какие типы микропроцессоров вы знаете?
8. В чем основные различия между CISC и RISC микропроцессорами?
9. Приведите примеры микропроцессоров CISC и RISC.
10. Какие параметры влияют на производительность микропроцессора?

Практическое занятие № 11

Тема: Понятие сетей и их организация.

Цель занятия:

Изучить компьютерные сети и их организацию.

Краткие сведения из теории

Компьютерная сеть — это система взаимосвязанных компьютеров и других устройств, которые могут обмениваться данными и ресурсами. Компьютерные сети могут быть локальными (LAN), глобальными (WAN), корпоративными (MAN) и другими. Основные компоненты компьютерной сети включают:

1. **Узлы (ноды):** Компьютеры, серверы, маршрутизаторы и другие устройства.
2. **Средства передачи данных:** Кабели (медные, оптоволоконные), беспроводные технологии (Wi-Fi, Bluetooth).
3. **Протоколы:** Наборы правил и стандартов, которые определяют, как данные передаются и обрабатываются в сети (например, TCP/IP, HTTP, FTP).
4. **Сетевые устройства:** Маршрутизаторы, коммутаторы, точки доступа, модемы.

Организация компьютерных сетей

Организация компьютерных сетей включает несколько уровней и аспектов:

1. **Топология сети:** Физическое и логическое расположение узлов и средств передачи данных. Основные топологии включают:
 - Звезда
 - Кольцо
 - Шина
 - Дерево
 - Сетка
 - Гибридные топологии
2. **Модели сетевого взаимодействия:**
 - Модель OSI (Open Systems Interconnection): 7 уровней (физический, канальный, сетевой, транспортный, сеансовый, представления, прикладной).
 - Модель TCP/IP: 4 уровня (сетевой доступ, интернет, транспортный, прикладной).
3. **Сетевые протоколы:** Наборы правил и стандартов, которые определяют, как данные передаются и обрабатываются в сети. Примеры:
 - TCP/IP (Transmission Control Protocol/Internet Protocol)
 - HTTP (HyperText Transfer Protocol)
 - FTP (File Transfer Protocol)
 - SMTP (Simple Mail Transfer Protocol)
4. **Сетевые услуги:** Сервисы, предоставляемые сетью, такие как электронная почта, веб-серфинг, файловый обмен, удаленный доступ и другие.

Задания для выполнения студентами

Задание 1: Изучение топологий сетей

1. Опишите основные топологии сетей (звезда, кольцо, шина, дерево, сетка, гибридные).

2. Приведите примеры использования каждой топологии.
3. Нарисуйте схемы для каждой топологии.

Задание 2: Анализ модели OSI

1. Опишите каждый уровень модели OSI.
2. Приведите примеры протоколов, соответствующих каждому уровню.
3. Объясните, как данные передаются между уровнями модели OSI.

Задание 3: Анализ модели TCP/IP

1. Опишите каждый уровень модели TCP/IP.
2. Приведите примеры протоколов, соответствующих каждому уровню.
3. Объясните, как данные передаются между уровнями модели TCP/IP.

Задание 4: Практическое задание на работу с сетевыми устройствами

1. Используя симулятор сетей (например, Cisco Packet Tracer или другие), создайте простую сеть с несколькими узлами и сетевыми устройствами (компьютеры, маршрутизаторы, коммутаторы).
2. Настройте сеть для обеспечения связи между узлами.
3. Опишите шаги настройки и результаты.

Задание 5: Анализ сетевых протоколов

1. Изучите основные сетевые протоколы (TCP/IP, HTTP, FTP, SMTP).
2. Опишите, как работает каждый протокол.
3. Приведите примеры использования каждого протокола.

Контрольные вопросы по теме

1. Что такое компьютерная сеть и какие основные компоненты она включает?
2. Какие основные топологии сетей вы знаете?
3. Опишите модель OSI и ее уровни.
4. Опишите модель TCP/IP и ее уровни.
5. Какие основные сетевые протоколы вы знаете?
6. Как работает протокол TCP/IP?
7. Как работает протокол HTTP?
8. Как работает протокол FTP?
9. Как работает протокол SMTP?
10. Какие сетевые услуги предоставляются компьютерными сетями?

Практическое занятие № 12

Тема: Передача информации по локальной сети.

Цель занятия:

Развить навыки работы с локальными сетями, а также понять их структуру и принципы функционирования.

Краткие сведения из теории

Понятие локальной сети (LAN)

Локальная сеть (LAN) — это сеть, охватывающая ограниченную географическую область, такую как офис, здание или кампус. Основные компоненты локальной сети включают:

1. **Узлы (ноды):** Компьютеры, серверы, принтеры и другие устройства.
2. **Средства передачи данных:** Кабели (медные, оптоволоконные), беспроводные технологии (Wi-Fi, Bluetooth).
3. **Сетевые устройства:** Маршрутизаторы, коммутаторы, точки доступа, концентраторы.
4. **Протоколы:** Наборы правил и стандартов, которые определяют, как данные передаются и обрабатываются в сети (например, Ethernet, Wi-Fi).

Основные методы передачи данных в LAN

1. **Ethernet:** Наиболее распространенный метод передачи данных в проводных сетях. Использует кабели (UTP, STP, оптоволокно) и стандарты (10BASE-T, 100BASE-TX, 1000BASE-T).
2. **Wi-Fi:** Беспроводной метод передачи данных, использующий радиоволны. Стандарты включают 802.11a, 802.11b, 802.11g, 802.11n, 802.11ac.
3. **Token Ring:** Метод передачи данных, использующий токен для управления доступом к сети.

Протоколы и стандарты передачи данных

1. **Ethernet:** Определяет физический и канальный уровни модели OSI. Использует CSMA/CD (Carrier Sense Multiple Access with Collision Detection) для управления доступом к сети.
2. **Wi-Fi:** Определяет физический и канальный уровни модели OSI. Использует CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) для управления доступом к сети.
3. **TCP/IP:** Набор протоколов, используемых для передачи данных в сети. Включает протоколы TCP (Transmission Control Protocol) и IP (Internet Protocol).

Задания для выполнения студентами

Задание 1: Изучение топологий локальных сетей

1. Опишите основные топологии локальных сетей (звезда, кольцо, шина, дерево, сетка, гибридные).
2. Приведите примеры использования каждой топологии.
3. Нарисуйте схемы для каждой топологии.

Задание 2: Анализ методов передачи данных в LAN

1. Опишите метод Ethernet и его основные стандарты.
2. Опишите метод Wi-Fi и его основные стандарты.
3. Сравните методы Ethernet и Wi-Fi, указав их преимущества и недостатки.

Задание 3: Практическое задание на работу с сетевыми устройствами

1. Используя симулятор сетей (например, Cisco Packet Tracer или другие), создайте простую локальную сеть с несколькими узлами и сетевыми устройствами (компьютеры, маршрутизаторы, коммутаторы).
2. Настройте сеть для обеспечения связи между узлами.

3. Опишите шаги настройки и результаты.

Задание 4: Анализ протоколов передачи данных

1. Изучите протокол Ethernet и его основные характеристики.
2. Изучите протокол Wi-Fi и его основные характеристики.
3. Опишите, как работает протокол TCP/IP в локальной сети.

Задание 5: Практическое задание на передачу данных

1. Используя симулятор сетей, создайте сценарий передачи данных между двумя узлами в локальной сети.
2. Опишите шаги передачи данных и результаты.
3. Проанализируйте возможные проблемы и способы их решения.

Контрольные вопросы по теме

1. Что такое локальная сеть (LAN) и какие основные компоненты она включает?
2. Какие основные топологии локальных сетей вы знаете?
3. Опишите метод Ethernet и его основные стандарты.
4. Опишите метод Wi-Fi и его основные стандарты.
5. Какие основные протоколы используются для передачи данных в локальной сети?
6. Как работает протокол Ethernet?
7. Как работает протокол Wi-Fi?
8. Как работает протокол TCP/IP в локальной сети?
9. Какие сетевые устройства используются в локальной сети?
10. Какие методы управления доступом к сети используются в Ethernet и Wi-Fi?

Практическое занятие № 13

Тема: Современные операционные среды.

Цель занятия:

Получить навыки работы с различными операционными системами.

Краткие сведения из теории

Операционная система (ОС) — это программное обеспечение, которое управляет аппаратными ресурсами компьютера и предоставляет платформу для выполнения приложений. Основные функции операционной системы включают:

1. **Управление процессами:** Планирование и выполнение задач.
2. **Управление памятью:** Распределение и управление оперативной памятью.
3. **Управление файлами:** Организация и управление файлами и каталогами.
4. **Управление устройствами:** Взаимодействие с периферийными устройствами (принтеры, диски и т.д.).
5. **Сетевое взаимодействие:** Поддержка сетевых протоколов и услуг.

Операционные системы семейства MS Windows

Microsoft Windows — это семейство операционных систем, разработанных компанией Microsoft. Основные версии включают:

1. **Windows 10:** Современная версия ОС, предназначенная для настольных компьютеров, ноутбуков и планшетов.
2. **Windows Server:** Версия ОС, предназначенная для серверов и корпоративных сетей.
3. **Windows 11:** Последняя версия ОС, предлагающая улучшенный интерфейс и новые функции.

Операционные системы семейства Linux

Linux — это семейство операционных систем, основанных на ядре Linux. Основные дистрибутивы включают:

1. **Ubuntu:** Популярный дистрибутив, предназначенный для настольных компьютеров и серверов.
2. **Debian:** Стабильный и надежный дистрибутив, используемый в качестве основы для многих других дистрибутивов.
3. **CentOS:** Дистрибутив, основанный на Red Hat Enterprise Linux (RHEL), предназначенный для серверов.
4. **Fedora:** Дистрибутив, спонсируемый Red Hat, предназначенный для настольных компьютеров и серверов.

Основные различия между Windows и Linux

1. **Лицензирование:** Windows — коммерческая ОС, Linux — открытая и бесплатная.
2. **Интерфейс:** Windows имеет графический интерфейс, Linux может иметь различные графические интерфейсы (GNOME, KDE и т.д.).
3. **Управление пакетами:** Windows использует установщики (.exe, .msi), Linux использует пакетные менеджеры (apt, yum, dnf).
4. **Безопасность:** Linux считается более безопасной из-за своей архитектуры и открытости кода.

Задания для выполнения студентами

Задание 1: Изучение операционных систем семейства MS Windows

1. Опишите основные версии операционных систем семейства MS Windows.
2. Перечислите основные функции и особенности Windows 10.
3. Опишите основные различия между Windows 10 и Windows 11.

Задание 2: Изучение операционных систем семейства Linux

1. Опишите основные дистрибутивы операционных систем семейства Linux.
2. Перечислите основные функции и особенности Ubuntu.
3. Опишите основные различия между Ubuntu и Debian.

Задание 3: Сравнение Windows и Linux

1. Сравните лицензирование и стоимость Windows и Linux.
2. Сравните интерфейсы пользователя Windows и Linux.
3. Сравните методы управления пакетами в Windows и Linux.
4. Сравните уровень безопасности Windows и Linux.

Задание 4: Практическое задание на установку и настройку Windows

1. Установите Windows 10 на виртуальную машину (например, с помощью VirtualBox или VMware).
2. Настройте основные параметры системы (пользователи, сетевые настройки, обновления).
3. Опишите шаги установки и настройки.

Задание 5: Практическое задание на установку и настройку Linux

1. Установите Ubuntu на виртуальную машину (например, с помощью VirtualBox или VMware).
2. Настройте основные параметры системы (пользователи, сетевые настройки, обновления).
3. Опишите шаги установки и настройки.

Контрольные вопросы по теме

1. Что такое операционная система и какие основные функции она выполняет?
2. Какие основные версии операционных систем семейства MS Windows вы знаете?
3. Какие основные дистрибутивы операционных систем семейства Linux вы знаете?
4. В чем основные различия между Windows и Linux?
5. Какие основные функции и особенности имеет Windows 10?
6. Какие основные функции и особенности имеет Ubuntu?
7. Как управляются пакеты в Windows и Linux?
8. Какие методы обеспечения безопасности используются в Windows и Linux?
9. Какие шаги необходимо выполнить для установки Windows 10 на виртуальную машину?
10. Какие шаги необходимо выполнить для установки Ubuntu на виртуальную машину?

Практическое занятие № 14

Тема: Текстовые редакторы.

Цель занятия:

Закрепить теоретические знания и развить навыки работы с различными текстовыми редакторами, а также понять их структуру и принципы функционирования.

Краткие сведения из теории

Текстовый редактор — это программное обеспечение, предназначенное для создания, редактирования и сохранения текстовых файлов. Текстовые редакторы могут быть простыми или сложными, в зависимости от их функциональности и назначения. Основные категории текстовых редакторов включают:

1. **Простые текстовые редакторы:** Предназначены для создания и редактирования текстовых файлов без форматирования. Примеры: Notepad (Windows), TextEdit (MacOS).
2. **Расширенные текстовые редакторы:** Предлагают дополнительные функции, такие как подсветка синтаксиса, автодополнение, поддержка плагинов и интеграция с системами контроля версий. Примеры: Notepad++, Sublime Text, Visual Studio Code.
3. **Текстовые процессоры:** Предназначены для создания и редактирования форматированных документов. Примеры: Microsoft Word, LibreOffice Writer.

Основные функции текстовых редакторов

1. **Создание и редактирование текста:** Основная функция всех текстовых редакторов.
2. **Подсветка синтаксиса:** Выделение различных элементов кода разными цветами для улучшения читаемости.
3. **Автодополнение:** Автоматическое завершение слов и кода на основе введенных символов.
4. **Поддержка плагинов:** Возможность расширения функциональности за счет установки дополнительных модулей.
5. **Интеграция с системами контроля версий:** Поддержка работы с системами контроля версий, такими как Git.
6. **Поддержка различных форматов файлов:** Возможность работы с различными типами файлов (.txt, .html, .css, .js и т.д.).

Задания для выполнения студентами

Задание 1: Изучение простых текстовых редакторов

1. Опишите основные функции и особенности Notepad (Windows) и TextEdit (MacOS).
2. Создайте текстовый файл с помощью Notepad или TextEdit и сохраните его.
3. Опишите шаги создания и сохранения текстового файла.

Задание 2: Изучение расширенных текстовых редакторов

1. Опишите основные функции и особенности Notepad++, Sublime Text и Visual Studio Code.
2. Установите один из расширенных текстовых редакторов (например, Notepad++).
3. Создайте и отредактируйте HTML-файл с использованием подсветки синтаксиса и автодополнения.
4. Опишите шаги создания и редактирования HTML-файла.

Задание 3: Изучение текстовых процессоров

1. Опишите основные функции и особенности Microsoft Word и LibreOffice Writer.
2. Создайте форматированный документ с использованием Microsoft Word или LibreOffice Writer.

3. Опишите шаги создания и форматирования документа.

Задание 4: Практическое задание на работу с текстовыми редакторами

1. Установите и настройте расширенный текстовый редактор (например, Visual Studio Code).
2. Установите несколько плагинов для расширения функциональности редактора (например, плагины для поддержки различных языков программирования).
3. Создайте и отредактируйте файл с исходным кодом на языке программирования (например, Python или JavaScript).
4. Опишите шаги установки и настройки редактора, а также создания и редактирования файла с исходным кодом.

Задание 5: Сравнение текстовых редакторов

1. Сравните функциональность простых текстовых редакторов (Notepad, TextEdit) и расширенных текстовых редакторов (Notepad++, Sublime Text, Visual Studio Code).
2. Сравните функциональность текстовых редакторов и текстовых процессоров (Microsoft Word, LibreOffice Writer).
3. Опишите, в каких случаях лучше использовать простые текстовые редакторы, расширенные текстовые редакторы и текстовые процессоры.

Контрольные вопросы по теме

1. Что такое текстовый редактор и какие основные функции он выполняет?
2. Какие основные категории текстовых редакторов вы знаете?
3. Какие основные функции и особенности имеет Notepad (Windows)?
4. Какие основные функции и особенности имеет TextEdit (MacOS)?
5. Какие основные функции и особенности имеет Notepad++?
6. Какие основные функции и особенности имеет Sublime Text?
7. Какие основные функции и особенности имеет Visual Studio Code?
8. Какие основные функции и особенности имеет Microsoft Word?
9. Какие основные функции и особенности имеет LibreOffice Writer?
10. В каких случаях лучше использовать простые текстовые редакторы, расширенные текстовые редакторы и текстовые процессоры?

Практическое занятие № 15

Тема: Процессоры электронных таблиц.

Цель занятия:

Изучить работу с различными процессорами электронных таблиц, а также понять их структуру и принципы функционирования.

Краткие сведения из теории

Процессоры электронных таблиц — это программное обеспечение, предназначенное для создания, редактирования и анализа табличных данных. Электронные таблицы позволяют пользователям организовывать данные в строки и столбцы, выполнять вычисления, создавать графики и диаграммы, а также автоматизировать повторяющиеся задачи.

Основные функции процессоров электронных таблиц

1. **Создание и редактирование таблиц:** Ввод и редактирование данных в ячейках таблицы.
2. **Форматирование данных:** Изменение внешнего вида ячеек, строк и столбцов (шрифты, цвета, границы и т.д.).
3. **Формулы и функции:** Выполнение арифметических и логических операций, использование встроенных функций для вычислений.
4. **Графики и диаграммы:** Визуализация данных с помощью различных типов графиков и диаграмм.
5. **Фильтрация и сортировка:** Организация данных по определенным критериям.
6. **Макросы и автоматизация:** Создание и выполнение скриптов для автоматизации повторяющихся задач.
7. **Импорт и экспорт данных:** Обмен данными с другими приложениями и форматами файлов.

Популярные процессоры электронных таблиц

1. **Microsoft Excel:** Один из самых популярных процессоров электронных таблиц, входящий в состав пакета Microsoft Office.
2. **Google Sheets:** Веб-приложение для работы с электронными таблицами, входящее в состав Google Workspace.
3. **LibreOffice Calc:** Бесплатный и открытый процессор электронных таблиц, входящий в состав пакета LibreOffice.
4. **Apple Numbers:** Процессор электронных таблиц, разработанный компанией Apple для macOS и iOS.

Задания для выполнения студентами

Задание 1: Изучение основных функций Microsoft Excel

1. Опишите основные функции и особенности Microsoft Excel.
2. Создайте новую электронную таблицу в Excel и введите данные в несколько ячеек.
3. Примените форматирование к ячейкам (измените шрифт, цвет, границы и т.д.).
4. Опишите шаги создания и форматирования таблицы.

Задание 2: Работа с формулами и функциями в Excel

1. Опишите, как использовать формулы и функции в Excel.

2. Создайте таблицу с числовыми данными и выполните несколько арифметических операций (сложение, вычитание, умножение, деление).
3. Используйте встроенные функции Excel (например, SUM, AVERAGE, MAX, MIN) для вычислений.
4. Опишите шаги выполнения операций и использования функций.

Задание 3: Создание графиков и диаграмм в Excel

1. Опишите, как создавать графики и диаграммы в Excel.
2. Создайте таблицу с данными и постройте график или диаграмму на основе этих данных.
3. Опишите шаги создания графика или диаграммы.

Задание 4: Фильтрация и сортировка данных в Excel

1. Опишите, как фильтровать и сортировать данные в Excel.
2. Создайте таблицу с данными и примените фильтрацию и сортировку по определенным критериям.
3. Опишите шаги фильтрации и сортировки данных.

Задание 5: Изучение Google Sheets

1. Опишите основные функции и особенности Google Sheets.
2. Создайте новую электронную таблицу в Google Sheets и введите данные в несколько ячеек.
3. Примените форматирование к ячейкам (измените шрифт, цвет, границы и т.д.).
4. Опишите шаги создания и форматирования таблицы.

Задание 6: Сравнение Microsoft Excel и Google Sheets

1. Сравните основные функции и особенности Microsoft Excel и Google Sheets.
2. Опишите преимущества и недостатки каждого процессора электронных таблиц.
3. Приведите примеры задач, для которых лучше подходит Microsoft Excel и Google Sheets.

Контрольные вопросы по теме

1. Что такое процессор электронных таблиц и какие основные функции он выполняет?
2. Какие основные функции и особенности имеет Microsoft Excel?
3. Как использовать формулы и функции в Microsoft Excel?
4. Как создавать графики и диаграммы в Microsoft Excel?
5. Как фильтровать и сортировать данные в Microsoft Excel?
6. Какие основные функции и особенности имеет Google Sheets?
7. В чем основные различия между Microsoft Excel и Google Sheets?
8. Какие преимущества и недостатки имеет Microsoft Excel по сравнению с Google Sheets?
9. Какие преимущества и недостатки имеет Google Sheets по сравнению с Microsoft Excel?
10. Приведите примеры задач, для которых лучше подходит Microsoft Excel и Google Sheets.

Практическое занятие № 16

Тема: Языки программирования.

Цель занятия:

Понять основы языков программирования и применить их на практике.

Краткие сведения из теории

Языки программирования — это формальные языки, предназначенные для записи компьютерных программ. Они позволяют программистам давать инструкции компьютеру, которые тот может выполнять. Языки программирования делятся на несколько категорий:

1. **Машинные языки:** Прямое представление инструкций процессора в виде двоичных кодов.
2. **Ассемблеры:** Низкоуровневые языки, где каждая инструкция соответствует одной машинной инструкции, но записывается в более удобной для человека форме.
3. **Высокоуровневые языки:** Языки, которые абстрагируют детали аппаратного обеспечения и предоставляют более удобные для программиста конструкции. Примеры: C, C++, Java, Python.
4. **Скриптовые языки:** Языки, предназначенные для автоматизации задач и написания небольших программ. Примеры: JavaScript, PHP, Perl.

Каждый язык программирования имеет свои особенности, синтаксис и области применения. Выбор языка зависит от задачи, которую нужно решить, и от предпочтений программиста.

Задания для выполнения студентами

1. **Задание 1: Изучение синтаксиса языка Python**
 - Напишите программу на Python, которая выводит на экран текст "Hello, World!".
 - Напишите программу, которая запрашивает у пользователя его имя и выводит приветствие в формате "Hello, [Имя]!".
2. **Задание 2: Работа с переменными и типами данных**
 - Напишите программу на Python, которая запрашивает у пользователя два числа и выводит их сумму, разность, произведение и частное.
 - Напишите программу, которая запрашивает у пользователя его возраст и выводит сообщение о том, является ли он совершеннолетним (18 лет и старше).
3. **Задание 3: Условные операторы и циклы**
 - Напишите программу на Python, которая запрашивает у пользователя число и выводит сообщение о том, является ли оно положительным, отрицательным или равным нулю.
 - Напишите программу, которая выводит на экран таблицу умножения для чисел от 1 до 10.
4. **Задание 4: Функции и модули**
 - Напишите функцию на Python, которая принимает два числа и возвращает их сумму.

- Напишите модуль на Python, который содержит функции для вычисления факториала числа и проверки, является ли число простым.

Контрольные вопросы по теме

1. Что такое язык программирования?
2. Какие существуют категории языков программирования?
3. Что такое машинный язык и ассемблер?
4. В чем разница между высокоуровневыми и скриптовыми языками программирования?
5. Какие основные элементы синтаксиса есть в языке Python?
6. Как в Python объявляются переменные и какие типы данных существуют?
7. Какие условные операторы и циклы существуют в Python?
8. Что такое функция в Python и как она объявляется?
9. Как в Python создаются и используются модули?
10. Приведите примеры задач, для решения которых можно использовать Python.

Практическое занятие № 17

Тема: Структура программы.

Цель занятия:

Ознакомиться с основами структуры программ.

Краткие сведения из теории

Структура программы — это организация и расположение различных элементов программы, таких как переменные, функции, модули и блоки кода, которые обеспечивают её корректную работу и удобство в поддержке и развитии. Хорошо структурированная программа легко читается, модифицируется и отлаживается.

Основные элементы структуры программы:

1. **Переменные и константы:** Хранят данные, используемые в программе.
2. **Функции и процедуры:** Блоки кода, выполняющие определённые задачи.
3. **Модули:** Группы функций и переменных, объединённые по смыслу.
4. **Блоки кода:** Части программы, выполняющие определённые действия (например, циклы, условия).
5. **Комментарии:** Текстовые пояснения, помогающие понять логику работы программы.

Пример структуры программы на языке Python:

```
# Импорт модулей
import math

# Константы
PI = 3.14

# Функции
def calculate_area(radius):
    return PI * radius * radius

def main():
    radius = 5
    area = calculate_area(radius)
    print(f"Площадь круга с радиусом {radius} равна {area}")

# Основной блок кода
if __name__ == "__main__":
    main()
```

Задания для выполнения студентами

1. **Задание 1: Создание простой программы**
 - Напишите программу на языке Python, которая вычисляет площадь прямоугольника. Программа должна содержать функцию для вычисления площади и основной блок кода для ввода данных и вывода результата.
2. **Задание 2: Работа с модулями**

- Создайте модуль, содержащий функции для вычисления площади круга и треугольника. В основной программе импортируйте этот модуль и используйте его функции для вычисления площадей.

3. Задание 3: Комментарии и документация

- Добавьте комментарии к вашей программе, объясняющие её работу. Также добавьте документацию к функциям, используя строки документации (docstrings).

4. Задание 4: Рефакторинг кода

- Перепишите программу, написанную в Задании 1, используя функции и модули для улучшения её структуры и читаемости.

Контрольные вопросы по теме

1. Что такое структура программы и почему она важна?
2. Какие основные элементы входят в структуру программы?
3. Что такое функция и зачем она нужна в программе?
4. Как использовать модули для организации кода?
5. Зачем нужны комментарии в программе и как их правильно писать?
6. Что такое документация функций и как её добавить в код?
7. Как рефакторинг кода может улучшить его структуру и читаемость?

Пример ответа на контрольный вопрос

Вопрос 1: Что такое структура программы и почему она важна?

Ответ: Структура программы — это организация и расположение различных элементов программы, таких как переменные, функции, модули и блоки кода. Она важна, потому что хорошо структурированная программа легко читается, модифицируется и отлаживается. Это упрощает работу с кодом, делает его более понятным для других разработчиков и уменьшает вероятность ошибок.

Практическое занятие № 18

Тема: Визуальная среда программирования

Цель занятия:

Ознакомиться с визуальной средой программирования.

Краткие сведения из теории

Визуальная среда программирования — это инструмент, который предоставляет графический интерфейс для разработки программного обеспечения. В отличие от текстовых редакторов, визуальные среды программирования позволяют разработчикам создавать и редактировать код с помощью графических элементов, таких как окна, кнопки, меню и другие визуальные компоненты.

Основные характеристики визуальной среды программирования:

1. **Графический интерфейс:** Позволяет разработчикам визуально создавать и редактировать элементы интерфейса.
2. **Драг-энд-дроп:** Возможность перетаскивания элементов интерфейса на рабочую область.
3. **Интеграция с редактором кода:** Поддержка редактирования кода вручную и автоматическая генерация кода для визуальных элементов.
4. **Отладка и тестирование:** Встроенные инструменты для отладки и тестирования программ.
5. **Библиотеки компонентов:** Набор готовых визуальных компонентов, которые можно использовать в программе.

Примеры визуальных сред программирования:

- **Visual Studio:** Популярная среда разработки от Microsoft, поддерживающая множество языков программирования.
- **Delphi:** Среда разработки для языка Pascal, известная своей мощной визуальной системой.
- **Qt Designer:** Инструмент для создания графических интерфейсов на основе библиотеки Qt.

Задания для выполнения студентами

1. **Задание 1: Установка и настройка визуальной среды программирования**
 - Установите одну из визуальных сред программирования (например, Visual Studio или Qt Designer).
 - Настройте среду для работы с выбранным языком программирования.
2. **Задание 2: Создание простого графического интерфейса**
 - Создайте простое приложение с графическим интерфейсом, содержащее кнопку и текстовое поле.
 - Напишите код, который при нажатии на кнопку изменяет текст в текстовом поле.
3. **Задание 3: Работа с визуальными компонентами**
 - Добавьте в ваше приложение несколько визуальных компонентов (например, чекбоксы, радиокнопки, списки).
 - Напишите код, который обрабатывает события этих компонентов и изменяет состояние приложения.
4. **Задание 4: Отладка и тестирование**
 - Используйте встроенные инструменты отладки для проверки работы вашего приложения.
 - Найдите и исправьте возможные ошибки в коде.

Контрольные вопросы по теме

1. Что такое визуальная среда программирования и какие её основные характеристики?
2. Какие преимущества предоставляет визуальная среда программирования по сравнению с текстовыми редакторами?
3. Какие инструменты и функции обычно включаются в визуальные среды программирования?
4. Приведите примеры популярных визуальных сред программирования.
5. Как визуальные среды программирования помогают в создании графических интерфейсов?
6. Что такое драг-энд-дроп и как он используется в визуальных средах программирования?
7. Как интегрировать визуальные компоненты с кодом в визуальной среде программирования?
8. Какие инструменты отладки и тестирования предоставляют визуальные среды программирования?

Пример ответа на контрольный вопрос

Вопрос 1: Что такое визуальная среда программирования и какие её основные характеристики?

Ответ: Визуальная среда программирования — это инструмент, который предоставляет графический интерфейс для разработки программного обеспечения. Основные характеристики визуальной среды программирования включают графический интерфейс, возможность перетаскивания элементов (драг-энд-дроп), интеграцию с редактором кода, встроенные инструменты для отладки и тестирования, а также библиотеки готовых визуальных компонентов. Эти характеристики позволяют разработчикам создавать и редактировать программы с помощью графических элементов, что упрощает процесс разработки и улучшает визуальное представление программы.

Практическое занятие № 19

Тема: Программирование на языке высокого уровня.

Цель занятия:

Ознакомиться с основами разработки на языке высокого уровня.

Краткие сведения из теории

Программирование на языке высокого уровня — это процесс создания программного обеспечения с использованием языков программирования, которые абстрагируют детали аппаратного обеспечения и предоставляют удобные и понятные конструкции для разработчиков. Языки высокого уровня, такие как Python, Java, C++ и другие, позволяют писать код, который легко читается и поддерживается, а также обеспечивают высокую производительность и гибкость.

Основные характеристики языков высокого уровня:

1. **Абстракция:** Скрытие деталей низкоуровневых операций и аппаратного обеспечения.
2. **Удобство использования:** Простые и понятные синтаксические конструкции.
3. **Портативность:** Возможность запуска программ на различных платформах без изменений.
4. **Библиотеки и фреймворки:** Набор готовых функций и инструментов для ускорения разработки.
5. **Автоматическое управление памятью:** Обеспечение безопасности и эффективности работы с памятью.

Примеры языков высокого уровня:

- **Python:** Интерпретируемый язык, известный своей простотой и читаемостью.
- **Java:** Объектно-ориентированный язык, широко используемый в веб-разработке и корпоративных приложениях.
- **C++:** Язык, сочетающий возможности низкоуровневого программирования с высокоуровневыми конструкциями.

Задания для выполнения студентами

1. Задание 1: Написание простой программы на Python

- Напишите программу на Python, которая запрашивает у пользователя два числа и выводит их сумму.

```
# Пример кода
num1 = float(input("Введите первое число: "))
num2 = float(input("Введите второе число: "))
sum = num1 + num2
print(f"Сумма чисел {num1} и {num2} равна {sum}")
```

2. Задание 2: Работа с условиями и циклами

- Напишите программу на Python, которая запрашивает у пользователя число и выводит все чётные числа от 1 до этого числа.

```
# Пример кода
num = int(input("Введите число: "))
for i in range(1, num + 1):
    if i % 2 == 0:
        print(i)
```

3. Задание 3: Работа с функциями

- Напишите программу на Python, которая содержит функцию для вычисления факториала числа. Программа должна запрашивать у пользователя число и выводить его факториал.

```
# Пример кода
def factorial(n):
    if n == 0:
        return 1
    else:
        return n * factorial(n - 1)

num = int(input("Введите число: "))
print(f"Факториал числа {num} равен {factorial(num)}")
```

4. Задание 4: Работа с массивами и списками

- Напишите программу на Python, которая создаёт список из 10 случайных чисел и выводит его, а также находит и выводит максимальное и минимальное значения в списке.

```
# Пример кода
import random

numbers = [random.randint(1, 100) for _ in range(10)]
print("Список чисел:", numbers)
print("Максимальное значение:", max(numbers))
print("Минимальное значение:", min(numbers))
```

Контрольные вопросы по теме

1. Что такое язык высокого уровня и какие его основные характеристики?
2. Какие преимущества предоставляют языки высокого уровня по сравнению с языками низкого уровня?
3. Приведите примеры популярных языков высокого уровня.
4. Как работают условные операторы и циклы в языках высокого уровня?
5. Что такое функция и как её определить в языке высокого уровня?
6. Как работать с массивами и списками в языках высокого уровня?
7. Какие библиотеки и фреймворки можно использовать для ускорения разработки на языках высокого уровня?
8. Как обеспечивается автоматическое управление памятью в языках высокого уровня?

Пример ответа на контрольный вопрос

Вопрос 1: Что такое язык высокого уровня и какие его основные характеристики?

Ответ: Язык высокого уровня — это язык программирования, который абстрагирует детали аппаратного обеспечения и предоставляет удобные и понятные конструкции для разработчиков. Основные характеристики языков высокого уровня включают абстракцию, удобство использования, портативность, наличие библиотек и фреймворков, а также автоматическое управление памятью. Эти характеристики позволяют писать код, который легко читается и поддерживается, а также обеспечивают высокую производительность и гибкость.

Практическое занятие № 20

Тема: Создание алгоритма для микропроцессора.

Цель занятия:

Научиться созданию алгоритма для микропроцессора.

Краткие сведения из теории

Создание алгоритма для микропроцессора — это процесс разработки последовательности инструкций, которые микропроцессор выполняет для решения определённой задачи. Алгоритмы для микропроцессоров обычно пишутся на языках низкого уровня, таких как ассемблер, или на языках высокого уровня, которые затем компилируются в машинный код.

Основные этапы создания алгоритма для микропроцессора:

1. **Определение задачи:** Понимание требований и целей алгоритма.
2. **Разработка алгоритма:** Создание последовательности шагов для решения задачи.
3. **Написание кода:** Реализация алгоритма на языке программирования, подходящем для микропроцессора.
4. **Отладка и тестирование:** Проверка работы алгоритма и исправление ошибок.
5. **Оптимизация:** Улучшение производительности и эффективности алгоритма.

Примеры языков программирования для микропроцессоров:

- **Ассемблер:** Язык низкого уровня, который предоставляет прямой доступ к аппаратным ресурсам.
- **C:** Язык высокого уровня, который часто используется для программирования встраиваемых систем.
- **MicroPython:** Версия языка Python, предназначенная для микроконтроллеров.

Задания для выполнения студентами

Задание 1: Написание простого алгоритма на ассемблере

- Напишите программу на ассемблере для микропроцессора, которая выполняет сложение двух чисел и сохраняет результат в памяти.

```
; Пример кода для микропроцессора x86
section .data
num1 dd 5
num2 dd 10
result dd 0

section .text
global _start

_start:
    mov eax, [num1]    ; Загрузка первого числа в регистр eax
    add eax, [num2]    ; Сложение второго числа с eax
    mov [result], eax ; Сохранение результата в память

    ; Завершение программы
    mov eax, 1          ; Системный вызов для выхода
    xor ebx, ebx        ; Код возврата 0
    int 0x80           ; Вызов ядра
```

Задание 2: Написание алгоритма на языке C для микроконтроллера

- Напишите программу на языке C для микроконтроллера, которая мигает светодиодом с заданной частотой.

```
// Пример кода для микроконтроллера Arduino
#include <Arduino.h>

const int ledPin = 13; // Пин светодиода

void setup() {
    pinMode(ledPin, OUTPUT); // Установка пина как выхода
}

void loop() {
    digitalWrite(ledPin, HIGH); // Включение светодиода
    delay(1000);                // Задержка на 1 секунду
    digitalWrite(ledPin, LOW);  // Выключение светодиода
    delay(1000);                // Задержка на 1 секунду
}
```

Задание 3: Написание алгоритма на MicroPython для микроконтроллера

- Напишите программу на MicroPython для микроконтроллера, которая считывает данные с датчика температуры и выводит их на дисплей.

```
# Пример кода для микроконтроллера ESP32 с MicroPython
import machine
import time

# Инициализация датчика температуры
sensor = machine.ADC(machine.Pin(34))

while True:
    # Чтение данных с датчика
    temperature = sensor.read()
    # Преобразование данных в градусы Цельсия
    temperature = (temperature * 3.3 / 4095) * 100
    # Вывод данных на дисплей
    print("Температура: {:.2f} °C".format(temperature))
    time.sleep(1)
```

Задание 4: Отладка и оптимизация алгоритма

- Выберите один из предыдущих алгоритмов и выполните его отладку и оптимизацию. Найдите и исправьте возможные ошибки, улучшите производительность и эффективность алгоритма.

Контрольные вопросы по теме

1. Что такое алгоритм для микропроцессора и какие этапы включает его создание?
2. Какие языки программирования используются для создания алгоритмов для микропроцессоров?
3. Какие преимущества и недостатки имеет использование ассемблера для программирования микропроцессоров?
4. Как написать простую программу на ассемблере для выполнения арифметических операций?
5. Как написать программу на языке C для микроконтроллера, которая управляет периферийными устройствами?
6. Как использовать MicroPython для программирования микроконтроллеров?
7. Какие инструменты отладки и тестирования можно использовать для проверки работы алгоритмов для микропроцессоров?

8. Как оптимизировать алгоритм для микропроцессора для улучшения его производительности и эффективности?

Пример ответа на контрольный вопрос

Вопрос 1: Что такое алгоритм для микропроцессора и какие этапы включает его создание?

Ответ: Алгоритм для микропроцессора — это последовательность инструкций, которые микропроцессор выполняет для решения определённой задачи. Создание алгоритма для микропроцессора включает несколько этапов: определение задачи, разработка алгоритма, написание кода, отладка и тестирование, а также оптимизация. На этапе определения задачи необходимо понять требования и цели алгоритма. Затем разрабатывается последовательность шагов для решения задачи. На этапе написания кода алгоритм реализуется на языке программирования, подходящем для микропроцессора. После этого выполняется отладка и тестирование алгоритма для проверки его работы и исправления ошибок. Наконец, алгоритм оптимизируется для улучшения его производительности и эффективности.....

Список используемой литературы

Основная

1. Миленина, С. А. Электротехника, электроника и схемотехника : учебник и практикум для среднего профессионального образования / С. А. Миленина, Н. К. Миленин ; под редакцией Н. К. Миленина. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 406 с. — (Профессиональное образование). — ISBN 978-5-534-04676-2. — Текст : электронный // Образовательная платформа Юрайт [сайт],. — URL: <https://urait.ru/bcode/511738> (дата обращения: 16.01.2024).
2. Толстобров, А. П. Архитектура ЭВМ : учебное пособие для среднего профессионального образования / А. П. Толстобров. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 162 с. — (Профессиональное образование). — ISBN 978-5-534-16832-7. — Текст : электронный // Образовательная платформа Юрайт [сайт],. — URL: <https://urait.ru/bcode/531856> (дата обращения: 16.01.2024).
3. Новожилов, О. П. Электроника и схемотехника в 2 ч. Часть 1 : учебник для среднего профессионального образования / О. П. Новожилов. — Москва : Издательство Юрайт, 2023. — 382 с. — (Профессиональное образование). — ISBN 978-5-534-10366-3. — Текст : электронный // Образовательная платформа Юрайт [сайт],. — URL: <https://urait.ru/bcode/517772> (дата обращения: 16.01.2024).
4. Математика и информатика : учебник и практикум для среднего профессионального образования / Т. М. Беляева [и др.], ; под редакцией В. Д. Элькина. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 402 с. — (Профессиональное образование). — ISBN 978-5-534-10683-1. — Текст : электронный // Образовательная платформа Юрайт [сайт],. — URL: <https://urait.ru/bcode/512073> (дата обращения: 16.01.2024).

Интернет-ресурсы

- www.ttgt.org (Сайт Тихорецкого Техникума Железнодорожного Транспорта)
- www.studentlibrary.ru (Электронная библиотека)
- www.urait.ru (Электронная библиотека)
- www.fcior.edu.ru (Федеральный центр информационно-образовательных ресурсов — ФЦИОР).
- www.school-collection.edu.ru (Единая коллекция цифровых образовательных ресурсов).
- www.megabook.ru (Мегаэнциклопедия Кирилла и Мефодия, разделы «Наука / Математика. Кибернетика» и «Техника / Компьютеры и Интернет»).
- www.ict.edu.ru (портал «Информационно-коммуникационные технологии в образовании»).