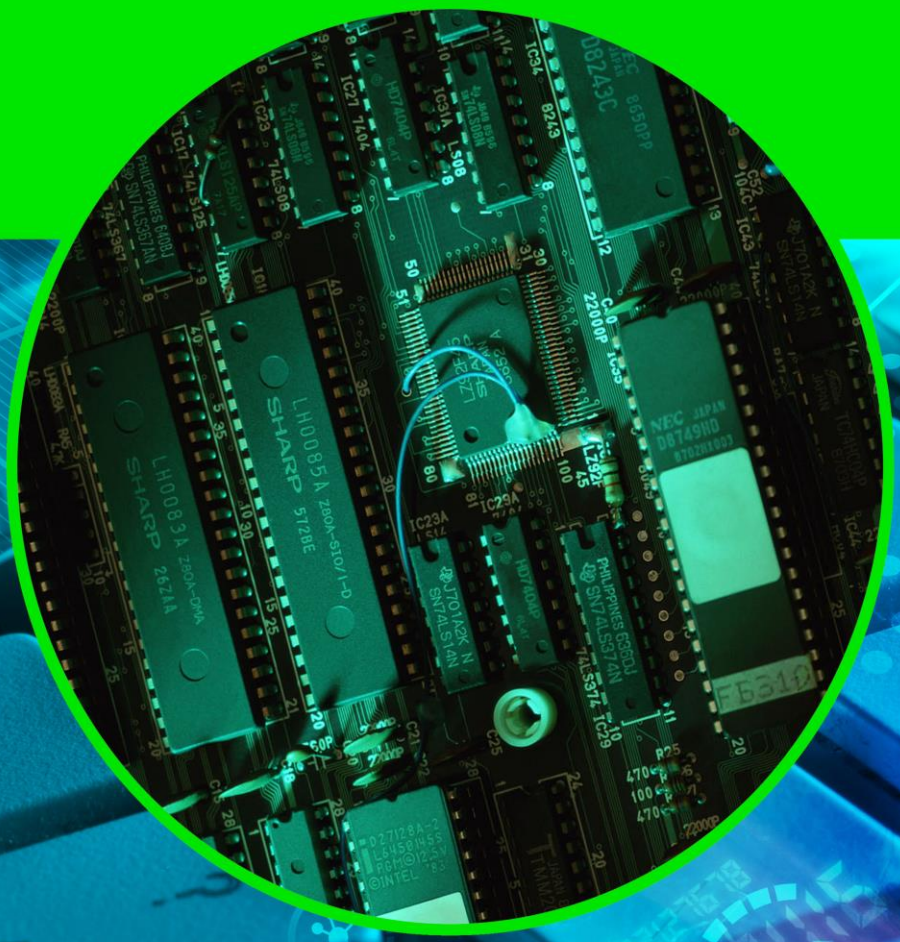
16+

Сеченовский Университет

С.В. Грейбо
Т.Е. Новосёлова
Н.Н. Пронькин
И.Ф. Семёнычева

# АРХИТЕКТУРА ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

## УЧЕБНОЕ ПОСОБИЕ



ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
**ПЕРВЫЙ МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ МЕДИЦИНСКИЙ УНИВЕРСИТЕТ ИМЕНИ И.М. СЕЧЕНОВА**
МИНИСТЕРСТВА ЗДРАВООХРАНЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
**(СЕЧЕНОВСКИЙ УНИВЕРСИТЕТ)**

ГРЕЙБО Сергей Владимирович
НОВОСЁЛОВА Тамара Евгеньевна
ПРОНЬКИН Николай Николаевич
СЕМЁНЫЧЕВА Ирина Флюровна

# АРХИТЕКТУРА ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

**Учебное пособие**

**Москва**
**2019**

УДК 004.2
ББК 22.18
А87

## Рецензент:

Герасимов А.Н., д.ф.-м..н., проф. Первый Московский государственный медицинский университет имени И.М. Сеченова (Сеченовский Университет)

## Авторы:

Грейбо Сергей Владимирович
Новосёлова Тамара Евгеньевна
Пронькин Николай Николаевич
Семёнычева Ирина Флюровна

Архитектура вычислительных систем [Электронный ресурс]: учебное пособие – Эл. изд. - Электрон. текстовые дан. (1 файл pdf: 77 с.). - Грейбо С.В., Новосёлова Т.Е., Пронькин Н.Н., Семёнычева И.Ф. 2019. – Режим доступа: http://scipro.ru/conf/computerarchitecture.pdf. Сист. требования: Adobe Reader; экран 10'.

ISBN 978-5-907072-89-3

Учебное пособие по изучению дисциплины «Архитектура вычислительных систем» разработано в соответствии с требованиями государственного образовательного стандарта.

В пособии представлены методические материалы по курсу «Архитектура вычислительных систем». Изложены базовые сведения и освещен круг вопросов, связанных с построением и функционированием вычислительных устройств и систем.

Материал излагается простым, доступным языком.

Учебное пособие рассмотрено и одобрено на заседании кафедры «Медицинская информатика и статистика» 21 января 2019 г., протокол № 5.

При верстке электронной книги использованы материалы с ресурсов: Designed by Freepik

ISBN 978-5-907072-89-3
9 785907 072893

© Грейбо С.В., Новосёлова Т.Е., Пронькин Н.Н., Семёнычева И.Ф. 2019
© Первый Московский государственный медицинский университет имени И.М. Сеченова (Сеченовский Университет), 2019
© Оформление: издательство НОО Профессиональная наука, 2019

# Содержание

ВВЕДЕНИЕ ..... 5
ГЛАВА 1. АРХИТЕКТУРА ЭВМ ..... 6
*1.1. Фон-неймановская архитектура ЭВМ ..... 7*
*1.2. Этапы развития средств вычислительной техники. Качественные отличия поколений ЭВМ ..... 9*
*1.3. Основные этапы решения задачи на ЭВМ ..... 10*
*1.4. Порядок обработки информации в ЭВМ ..... 12*
ГЛАВА 2. РОЛЬ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ В ОРГАНИЗАЦИИ ВЫЧИСЛИТЕЛЬНОГО ПРОЦЕССА ..... 14
*2.1. Состав и назначение компонентов программного обеспечения . 14*
*2.2. Особенности структуры вычислительной системы ..... 15*
*2.3. Классификация элементов ЭВМ ..... 17*
*2.4. Понятие интерфейса ..... 18*
*2.5. Роль контроллеров в схемах периферийных устройств ..... 19*
*2.6. Понятие и функции кэш-памяти ..... 20*
*2.7. Состав ядра компьютера ..... 20*
*2.8. Назначение системных вызовов и драйверов в современных операционных системах ..... 22*
ГЛАВА 3. КЛАССИФИКАЦИЯ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ ..... 23
*3.1. Альтернативные пути развития вычислительной техники ..... 29*
ГЛАВА 4. ХАРАКТЕРИСТИКИ УСТРОЙСТВ ВВОДА-ВЫВОДА. ВЗАИМОДЕЙСТВИЕ УСТРОЙСТВ ПК ..... 31
*4.1. Состав устройств ввода-вывода в ПК. Порядок подключения внешних устройств к ПК ..... 31*
*4.2. Устройство персонального компьютера ..... 49*
*4.3. Причины появления и развития режима реального времени ..... 50*
*4.4. Основные отличия однопрограммных и многопрограммных режимов работы компьютера ..... 51*
*4.5. Виды прерываний, реализуемые в ПК ..... 55*
*4.6. Реализация в арифметико-логическом устройстве алгоритмов основных операций ..... 57*
*4.7. Реализация программного принципа управления в управляющем устройстве ..... 60*
САМОСТОЯТЕЛЬНАЯ РАБОТА СТУДЕНТА ПО ИЗУЧЕНИЮ УЧЕБНОЙ ДИСЦИПЛИНЫ ..... 61
КОНТРОЛЬНЫЕ ВОПРОСЫ ..... 62
СЛОВАРЬ ТЕРМИНОВ И ОПРЕДЕЛЕНИЙ ..... 64
БИБЛИОГРАФИЧЕСКИЙ СПИСОК ..... 76

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

# ВВЕДЕНИЕ

Дисциплина «Архитектура вычислительных систем» входит в федеральный компонент блока общепрофессиональных дисциплин для подготовки бакалавров по направлениям 09.03.02 «Информационные системы и технологии» и 01.03.03 «Механика и математическое моделирование».

Целью дисциплины является ознакомление студентов с основными принципами организации аппаратного обеспечения ЭВМ и вычислительных систем, принципами работы периферийных устройств и их взаимодействия в составе системы.

Задачи дисциплины – приобретение практических навыков по работе на персональном компьютере, выбору рациональной конфигурации оборудования в соответствии с решаемой задачей; обеспечению совместимости аппаратных и программных средств вычислительной техники.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

# ГЛАВА 1. АРХИТЕКТУРА ЭВМ

С середины 60-х годов XX века очень сильно изменился подход к созданию вычислительных машин. Вместо разработки аппаратуры и средств математического обеспечения стала проектироваться система, состоящая из синтеза аппаратных (hardware) и программных (software) средств. При этом на главный план выдвинулась концепция взаимодействия. Так возникло новое понятие – архитектура ЭВМ.

**Под архитектурой ЭВМ принято понимать совокупность общих принципов организации аппаратно-программных средств и основных их характеристик, определяющая функциональные возможности ЭВМ при решении соответствующих типов задач.**

Архитектура ЭВМ охватывает обширный круг проблем, связанных с созданием комплекса аппаратных и программных средств и учитывающих большое количество определяющих факторов. Среди этих факторов самыми главными являются: стоимость, сфера применения, функциональные возможности, удобство в эксплуатации, а одним из основных компонентов архитектуры считаются аппаратные средства.

Архитектуру вычислительного средства (ВС) необходимо отличать от структуры ВС. **Структура вычислительного средства определяет его текущий состав на определенном уровне детализации и описывает связи внутри средства.** *Архитектура же определяет основные правила взаимодействия составных элементов вычислительного средства, описание которых выполняется в той мере, в какой необходимо для формирования правил взаимодействия.* Она устанавливает не все связи, а только наиболее необходимые, которые должны быть известны для более грамотного использования применяемого средства. Так, пользователю ЭВМ не обязательно знать, на каких элементах выполнены электронные схемы, схемно или программно исполняются команды и так далее. ***Архитектура ЭВМ действительно отражает круг проблем, которые относятся к общему проектированию и построению вычислительных машин и их программного обеспечения.***

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

## 1.1. Фон-неймановская архитектура ЭВМ

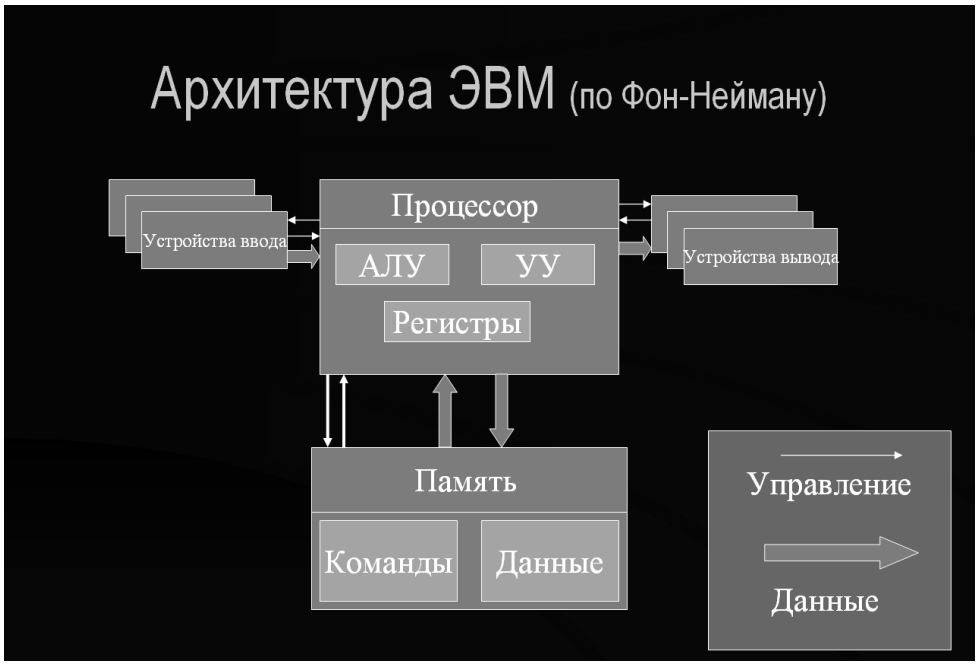



*Рисунок 1. Классическая архитектура ЭВМ (архитектура фон Неймана)*

Основные характеристики вычислительной техники

1. Быстродействие ЭВМ
2. Производительность ЭВМ
3. Емкость, или объем памяти
4. Точность вычислений
5. Система команд
6. Стоимость ЭВМ
7. Надежность ЭВМ

Принципы ДЖОНА ФОН НЕЙМАНА

- **Принцип двоичного кодирования**. Согласно этому принципу, вся информация, поступающая в ЭВМ, кодируется с помощью двоичных сигналов.
- **Принцип программного управления**. Из него следует, что программа состоит из набора команд, которые выполняются процессором автоматически друг за другом в определенной последовательности.
- **Принцип однородности памяти**. Программы и данные хранятся в одной и той же памяти. Поэтому ЭВМ не различает, что хранится в данной ячейке

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

памяти – число, текст или команда. Над командами можно выполнять такие же действия, как и над данными.

- **Принцип адресности**. Структурно основная память состоит из пронумерованных ячеек; процессору в произвольный момент времени доступна любая ячейка.

Классическая архитектура (архитектура фон Неймана) – одно арифметико-логическое устройство (АЛУ), через которое проходит поток данных, и одно устройство управления (УУ), через которое проходит поток команд – программа (рис. 1). Это **однопроцессорный компьютер**. К этому типу архитектуры относится и архитектура персонального компьютера с общей шиной. Все функциональные блоки здесь связаны между собой общей шиной, называемой также **системной магистралью.**

**Физически магистраль представляет собой многопроводную линию с гнездами для подключения электронных схем. Совокупность проводов магистрали разделяется на отдельные группы: шину адреса, шину данных и шину управления.**

Периферийные устройства (принтер и др.) подключаются к аппаратуре компьютера через специальные контроллеры **– устройства управления периферийными устройствами.**

***Контроллер*** **– устройство, которое связывает периферийное оборудование или каналы связи с центральным процессором, освобождая процессор от непосредственного управления функционированием данного оборудования.**

Многопроцессорная архитектура. Наличие в компьютере нескольких процессоров означает, что **параллельно может быть организовано много потоков данных и много потоков команд**. Таким образом, параллельно могут выполняться несколько фрагментов одной задачи. Структура такой машины, имеющей общую оперативную память и несколько процессоров, представлена на рис. 2.

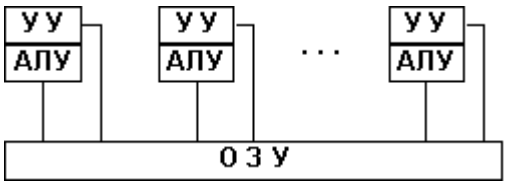



*Рисунок 2. Архитектура многопроцессорного компьютера*

Многомашинная вычислительная система. Здесь **несколько процессоров, входящих в вычислительную систему, не имеют общей оперативной памяти, а имеют каждый свою (локальную).** Каждый компьютер в многомашинной системе имеет классическую архитектуру, и такая система применяется достаточно широко. Однако эффект от применения такой вычислительной системы может быть получен

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

только при решении задач, имеющих очень специальную структуру: **она должна разбиваться на столько слабо связанных подзадач, сколько компьютеров в системе.**

Преимущество в быстродействии многопроцессорных и многомашинных вычислительных систем перед однопроцессорными очевидно.

Архитектура с параллельными процессорами. Здесь **несколько АЛУ работают под управлением одного УУ.** Это означает, что множество данных может обрабатываться по одной программе – то есть по одному потоку команд. Высокое быстродействие такой архитектуры можно получить только на задачах, в которых одинаковые вычислительные операции выполняются одновременно на различных однотипных наборах данных. Структура таких компьютеров представлена на рис. 3.

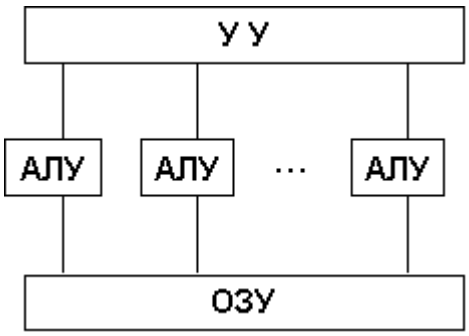



*Рисунок 3. Архитектура с параллельным процессором*

В современных машинах часто присутствуют элементы различных типов архитектурных решений. Существуют и такие архитектурные решения, которые радикально отличаются от рассмотренных выше.

## *1.2. Этапы развития средств вычислительной техники. Качественные отличия поколений ЭВМ*

**1-е поколение (середина 40 – середина 50-х)**

Основа – электронно-вакуумные лампы.

Быстродействие – десятки тысяч операций в секунду.

Оперативная память – несколько килобайт.

Программирование – преимущественно в машинных кодах.

*Пример – БЭСМ, УРАЛ.*

**2-е поколение (середина 50 – середина 60-х)**

Основа – полупроводниковая элементная база.

Быстродействие – сотни тысяч операций в секунду.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

Оперативная память – десятки килобайт.

Программирование – наличие ОС и языков программирования высокого уровня.

*Пример – Минск-22, Весна, БЭСМ-6.*

**3-е поколение (середина 60 – середина 70-х)**

Основа – интегральные схемы.

Быстродействие – миллионы операций в секунду.

Память – до сотен мегабайт.

Программирование – наличие ОС и появление алгоритмических языков программирования.

*Пример – ЕС ЭВМ (ЕС-1022 … ЕС-1066), СМ.*

**4-е поколение (середина 70 – начало 90-х)**

Основа – большие интегральные схемы.

Быстродействие – сотни миллионов операций в секунду.

Память – до нескольких гигабайт.

Программирование – наличие ОС и широкое использование алгоритмических языков программирования.

*Пример – AT/286, AT/386 … Pentium II.*

**5-е поколение (начало 90-х – настоящее время)**

Основа – сверхбольшие интегральные схемы.

Быстродействие – миллиарды операций в секунду.

Память – до нескольких терабайт.

Программирование – широкое использование объектно-ориентированных систем визуального программирования.

*Пример – Pentium III и далее.*

**6-е и последующие поколения**

оптоэлектронные ЭВМ с массовым параллелизмом и нейронной структурой – с распределенной сетью большого числа (десятки тысяч) несложных микропроцессоров, моделирующих архитектуру нейронных биологических систем.

## *1.3. Основные этапы решения задачи на ЭВМ*

В настоящее время на ЭВМ решают самые разнообразные задачи, от расчета баллистических траекторий до завоевания инопланетных территорий (пока только в компьютерных играх). В каждом случае ЭВМ выполняет какую-то программу, обычно довольно сложную.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

- **Постановка задачи**

Под постановкой задачи понимают математическую или иную строгую формулировку решаемой задачи. Этот этап включает определение целей создаваемой программы и определение ограничений, налагаемых на программу. При постановке задачи должны быть определены требования:

- ✓ ко времени решения поставленной задачи;
- ✓ объему необходимых ресурсов, например, оперативной памяти;
- ✓ точности достигаемого результата.

- **Проектирование программы**

Если задача вычислительная, то на этом этапе следует выбрать метод расчета, если разрабатывается компьютерная игра, должен быть определен ее сценарий. В любом случае следует выбрать или создать некую формальную модель, которая, в конечном счете, реализуется в будущей программе. На этапе проектирования определяют вид данных, с которыми будет работать программа, основные части, из которых программа будет состоять и характер связей между этими частями.

- **Разработка алгоритма**

На этом этапе следует разработать детали проекта программы. Детализацию необходимо довести до той степени, когда кодирование деталей программы (перевод их на алгоритмический язык) станет тривиальным. Возможно, детализация потребует нескольких стадий, от крупных блоков к все более мелким, и в результате должно получиться то, что называется ***алгоритмом*** решения задачи. Алгоритм – центральное понятие программирования, поэтому познакомиться с ним следует как можно раньше.

- **Кодирование**

После того как алгоритм разработан, его записывают на алгоритмическом языке, и этот процесс называют ***кодированием алгоритма***. Для выполнения данного этапа необходимо знать хотя бы один из многих существующих языков программирования, а лучше знать несколько, чтобы выбрать наиболее подходящий для решаемой задачи.

- **Отладка и тестирование программы**

Целью данного этапа является поиск и устранение ошибок в программе.

Ошибки бывают синтаксические (нарушение грамматики алгоритмического языка) и смысловые (искажение самого алгоритма решения задачи).

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

Разделение процесса разработки программ на 5 этапов носит весьма условный характер. В случае простых программ, которые предстоит писать начинающим программистам, некоторые этапы сливаются, например, проектирование с разработкой алгоритма или кодирование с отладкой. В случае сложных программ могут добавиться новые фазы разработки, например, *проектирование базы данных или создание изображений*.

Программа живет, приобретает новые функции, совершенствует старые, избавляется от последних ошибок и, наконец, умирает, уступив натиску более молодых программ, покоряющих сердца пользователей сверканием инструментальных панелей, трехмерностью изображений и стереофоничностью звуков.

## *1.4. Порядок обработки информации в ЭВМ*

**Понятие информации является основополагающим понятием информатики**. Любая деятельность человека представляет собой процесс сбора и переработки **информации**, принятия на ее основе решении и их выполнения. С появлением современных средств вычислительной техники информация стала выступать в качестве ***одного из важнейших ресурсов научно-технического прогресса***.

**Информацию можно:**

- **создавать;**
- **передавать;**
- **воспринимать;**
- **использовать;**
- **запоминать;**
- **принимать;**
- **копировать;**
- **формализовать;**
- **распространять;**
- **преобразовывать;**
- **комбинировать;**
- **обрабатывать;**
- **делить на части;**
- **упрощать;**
- **собирать;**
- **хранить;**
- **искать;**
- **измерять;**
- **разрушать;**
- **и др.**

Все эти процессы, связанные с определенными операциями над информацией, называются **информационными процессами.**

**Обработка информации – получение одних информационных объектов из других информационных объектов путем выполнения некоторых алгоритмов.**

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

Обработка является одной из основных операций, выполняемых над информацией, и главным средством увеличения объёма и разнообразия информации.

**Средства обработки информации – это всевозможные устройства и системы, созданные человечеством, и в первую очередь, компьютер** – универсальная машина для обработки информации.

**Компьютеры обрабатывают информацию путем выполнения некоторых алгоритмов.**

Каким образом рассмотренные понятия могут быть приложены к техническим проблемам передачи информации по линиям связи, к задачам хранения и обработки информации с помощью ЭВМ? Для того чтобы сообщение могло быть передано **от источника к потребителю**, оно должно быть каким-либо образом отображено, зафиксировано. Такое отображение осуществляется с помощью некоторых **символов (знаков)**. Конечный упорядоченный набор знаков, используемых для передачи сообщений, называют алфавитом. Последовательность символов алфавита – словом. Сообщение фиксируется как последовательность слов. Простой и всем понятный пример такого представления сообщения – его запись словами, составленными из букв русского алфавита. В канале связи способ представления сообщения может изменяться: один алфавит заменяется другим. *Правило отображения одного алфавита в другой называется* ***кодом***, а сама процедура – перекодировкой сообщения.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

# ГЛАВА 2. РОЛЬ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ В ОРГАНИЗАЦИИ ВЫЧИСЛИТЕЛЬНОГО ПРОЦЕССА

## *2.1. Состав и назначение компонентов программного обеспечения*

**Назначением ЭВМ является выполнение программ.**
***Программа содержит команды, определяющие порядок действий компьютера.***
***Совокупность программ для компьютера образует программное обеспечение (ПО).***

В первом приближении все программы, работающие на компьютере, можно условно разделить на **три категории** (рис. 4):

1. **прикладные программы**, непосредственно обеспечивающие выполнение необходимых пользователям работ;
2. **системные программы**, выполняющие различные вспомогательные функции, например:
   - управление ресурсами компьютера;
   - создание копий используемой информации;
   - проверка работоспособности устройств компьютера;
   - выдача справочной информации о компьютере и др.;
3. **инструментальные программные системы**, облегчающие процесс создания новых программ для компьютера.

*Рисунок 4. Категории программного обеспечения*

При построении классификации ПО нужно учитывать тот факт, что стремительное развитие вычислительной техники и расширение сферы приложения компьютеров резко ускорили процесс эволюции программного обеспечения.

На сегодняшний день можно сказать, что **более или менее определённо сложились следующие группы программного обеспечения:**

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

- операционные системы и оболочки;
- системы программирования (трансляторы, библиотеки подпрограмм, отладчики и т.д.);
- инструментальные системы;
- интегрированные пакеты программ;
- динамические электронные таблицы;
- системы машинной графики;
- системы управления базами данных (СУБД);
- прикладное программное обеспечение.

## *2.2. Особенности структуры вычислительной системы*

Вычислительная система (ВС) – результат интеграции аппаратных и программных средств, функционирующих в единой системе и предназначенных для решения задач определенного класса.

Структура вычислительной системы состоит из пяти уровней:

- Аппаратный уровень вычислительной системы

Определяется наборами аппаратных компонентов и их характеристиками, используемыми вышестоящими уровнями иерархии и определяющими воздействие на них.

*Физические ресурсы: процессор, оперативная память, внешние устройства*.

- Уровень управления физическими ресурсами вычислительной системы

Уровень управления физическими ресурсами ВС является первым уровнем системного программного обеспечения вычислительной системы, его назначение – систематизация и стандартизация правил программного использования физических ресурсов. На этом уровне обеспечивается создание программ управления физическими ресурсами. Для обеспечения управления физическими ресурсами используются программы, которые называются *драйверами физического ресурса* (устройства).

Драйвер физического устройства – программа, основанная на использовании команд управления конкретного физического устройства и предназначенная для организации работы с данным устройством.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

- Уровень управления логическими/виртуальными ресурсами вычислительной системы

Логическое/виртуальное устройство (ресурс) – устройство, некоторые эксплуатационные характеристики которого (возможно, все) реализованы программным образом.

Драйвер логического/виртуального ресурса – программа, обеспечивающая существование и использование соответствующего ресурса.

Этот уровень ориентирован на пользователя. Команды данного уровня не зависят от физических устройств, они обращены к предыдущему уровню. На базе этого уровня могут создаваться новые логические ресурсы. При организации драйвера логического устройства могут использоваться драйверы физических или логических/виртуальных устройств. Система поддерживает иерархию драйверов. Многоуровневая унификация интерфейса.

Ресурсы вычислительной системы – совокупность всех физических и виртуальных ресурсов. Одной из характеристик ресурсов является их конечность, следовательно, возникает конкуренция за обладание ресурсом между его программными потребителями.

**Операционная система – это комплекс программ, обеспечивающий управление ресурсами вычислительной системы.**

Разветвленная иерархия виртуальных и физических устройств. Драйверы можно разделить на 3 группы:

- Драйверы физических устройств
- Драйверы виртуальных устройств, обобщающих характеристики соответствующих физических устройств
- Драйверы виртуальных устройств, не имеющих аппаратной реализации (т.е. "полностью" виртуальных, например, драйвер файловой системы)

- Уровень систем программирования

Система программирования – это комплекс программ, обеспечивающий поддержание жизненного цикла программы в вычислительной системе.

***Жизненный цикл программы в вычислительной системе состоит из четырех основных этапов:***

- *Проектирование программного продукта.*
- *Кодирование (программная реализация).*
- *Тестирование и отладка.* Тестирование – проверка программы на тестовых нагрузках. Тест – заранее определенные входные данные. Тестовое покрытие – минимальный набор тестов, наиболее полно проверяющих программу.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

Отладка – процесс поиска, локализации и исправления зафиксированных при тестировании ошибок.

- *Внедрение и сопровождение.* Внедрение – установка программного комплекса на объектную вычислительную систему и его первичная настройка. Сопровождение – исправление недочетов разработки и внедрения программного комплекса (например, выпуск патчей).

- Уровень прикладных систем

---

Прикладная система – программная система, ориентированная на решение или автоматизацию решения задач из конкретной предметной области.

## *2.3. Классификация элементов ЭВМ*

При рассмотрении структуры любой ЭВМ обычно проводят ее детализацию. Как правило, в структуре ЭВМ выделяют следующие структурные единицы*: устройства, узлы, блоки и элементы*. Такая детализация соответствует вполне определенным операциям преобразования информации, заложенным в программах пользователей.

- Нижний уровень обработки реализуют *элементы*. Каждый элемент предназначается для обработки единичных электрических сигналов, соответствующих битам информации.
- *Узлы* обеспечивают одновременную обработку группы сигналов – информационных слов.
- *Блоки* реализуют некоторую последовательность в обработке информационных слов – функционально обособленную часть машинных операций (блок выборки команд, блок записи-чтения и др.).
- *Устройства* предназначаются для выполнения отдельных машинных операций и их последовательностей.

Элементы ЭВМ можно классифицировать по различным признакам. Наиболее часто такими признаками являются: тип сигналов, назначение элементов, технология их изготовления и т.д.

**В ЭВМ широко применяют два способа физического представления сигналов: импульсный и потенциальный. При импульсном способе представления сигналов единичному значению некоторой двоичной переменной ставится в**

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

**соответствие наличие импульса (тока или напряжения), нулевому значению – отсутствие импульса. Длительность импульсного сигнала не превышает один такт синхроимпульсов.**

При потенциальном или статическом представлении сигналов единично значение двоичной переменной отображается высоким уровнем напряжения, а нулевое значение – низким уровнем.

По своему назначению элементы делятся на ***формирующие, логические и запоминающие***.

К *формирующим элементам* относятся различные формирователи, усилители, усилители-формирователи и т.п. Данные элементы служат для выработки определенных электрических сигналов, восстановления их параметров (амплитуды, полярности, мощности, длительности).

В каждой ЭВМ имеются специальные блоки, формирующие сигналы тактовой частоты, серии синхронизирующих и управляющих сигналов, координирующих работу всех схем ЭВМ. **Интервал времени между импульсами основной частоты называется тактом**. *Длительность такта является важной характеристикой ЭВМ, определяющей ее потенциальную производительность. Время выполнения любой операции ЭВМ связано с определенным числом тактов.*

Простейшие *логические элементы* преобразуют входные сигналы в соответствии с элементарными логическими функциями. В свою очередь, полученные сигналы могут формировать следующий уровень сигналов и т. д. Сложные преобразования в соответствии с требуемыми логическими зависимостями могут приводить к построению многоуровневых схем. Каждая такая схема представляет собой композицию простейших логических схем.

*Запоминающим элементом* называется элемент, который способен принимать и хранить код двоичной цифры (единицы или нуля). Элементы памяти могут запоминать и сохранять исходные значения некоторых величин, промежуточные значения обработки и окончательные результаты вычислений. Только запоминающие элементы в схемах ЭВМ позволяют проводить обработку информации с учетом ее развития.

## *2.4. Понятие интерфейса*

Для того, чтобы соединить друг с другом различные устройства компьютера, они должны иметь одинаковый **интерфейс** (англ. interface от inter – между, и face – лицо).

***Интерфейс*** **– это средство сопряжения двух устройств, в котором все физические и логические параметры согласуются между собой.**

Если интерфейс является общепринятым, например, утверждённым на уровне международных соглашений, то он называется **стандартным**.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

Каждый из функциональных элементов (память, монитор или другое устройство) связан с ***шиной определённого типа – адресной, управляющей или шиной данных***.

Для согласования интерфейсов периферийные устройства подключаются к шине не напрямую, а через свои **контроллеры** (адаптеры) и **порты** примерно по такой схеме:

## *2.5. Роль контроллеров в схемах периферийных устройств*

**Контроллеры и адаптеры** представляют собой наборы электронных цепей, которыми снабжаются устройства компьютера с целью совместимости их интерфейсов. Контроллеры, кроме этого, осуществляют непосредственное управление периферийными устройствами по запросам микропроцессора.

***Порты устройств*** **представляют собой некие электронные схемы, содержащие один или несколько регистров ввода-вывода и позволяющие подключать периферийные устройства компьютера к внешним шинам микропроцессора.**

Портами также называют **устройства стандартного интерфейса**: *последовательный, параллельный и игровой порты (или интерфейсы)*.

***Последовательный порт*** **обменивается данными с процессором побайтно, а с внешними устройствами – побитно.**

***Параллельный порт*** **получает и посылает данные побайтно.**

К **последовательному** порту обычно подсоединяют медленно действующие или достаточно удалённые устройства, такие, как мышь и модем. К **параллельному** порту подсоединяют более "быстрые" устройства – принтер и сканер. Через **игровой** порт подсоединяется джойстик. Клавиатура и монитор подключаются к своим **специализированным** портам, которые представляют собой просто **разъёмы**.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

## 2.6. Понятие и функции кэш-памяти

***Кэш*** **(англ. cache), или сверхоперативная память – очень быстрое ЗУ небольшого объёма, которое используется при обмене данными между микропроцессором и оперативной памятью для компенсации разницы в скорости обработки информации процессором и несколько менее быстродействующей оперативной памятью.**

Кэш-памятью управляет специальное устройство – **контроллер,** который, анализируя выполняемую программу, пытается **предвидеть, какие данные и команды вероятнее всего понадобятся в ближайшее время процессору, и подкачивает их в кэш-память.** При этом возможны как "**попадания**", так и "**промахи**". В случае попадания, то есть, если в кэш подкачаны нужные данные, извлечение их из памяти происходит без задержки. Если же требуемая информация в кэше отсутствует, то процессор считывает её непосредственно из оперативной памяти. Соотношение числа попаданий и промахов определяет эффективность кэширования.

Кэш-память реализуется на **микросхемах статической памяти SRAM** (Static RAM), более быстродействующих, дорогих и малоёмких, чем DRAM (SDRAM). Современные микропроцессоры имеют **встроенную кэш-память**, так называемый **кэш первого уровня** размером 8, 16 или 32 Кбайт. Кроме того, на системной плате компьютера может быть установлен **кэш второго уровня** ёмкостью 256, 512 Кбайт и выше.

## 2.7. Состав ядра компьютера

***Центральный процессор*** **(CPU, от англ. Central Processing Unit) – это основной рабочий компонент компьютера, который выполняет арифметические и логические операции, заданные программой, управляет вычислительным процессом и координирует работу всех устройств компьютера.**

Центральный процессор в общем случае содержит в себе:

- **арифметико-логическое устройство;**
- **шины данных и шины адресов;**
- **регистры;**
- **счетчики команд;**
- **кэш – очень быструю память малого объема (от 8 до 512 Кбайт);**
- **математический сопроцессор чисел с плавающей точкой.**

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

Современные процессоры выполняются в виде **микропроцессоров**. **Физически микропроцессор представляет собой интегральную схему – тонкую пластинку кристаллического кремния прямоугольной формы площадью всего несколько квадратных миллиметров, на которой размещены схемы, реализующие все функции процессора.** Кристалл-пластинка обычно помещается в пластмассовый или керамический плоский корпус и соединяется золотыми проводками с металлическими штырьками, чтобы его можно было присоединить к системной плате компьютера.

В компьютере подразумевается ***процессорное ядро***, на основе которого **процессор выполняет свою главную функцию – математические вычисления на базе определенного набора инструкций**. Любое процессорное ядро имеет свое *кодовое название*. В качестве примера возьмем известного производителя Nortwood, хотя подобных производителей сегодня существует очень много. Процессорное ядро отвечает за математические вычисления, а значит, оно в большей степени влияет на общую производительность компьютера. Одновременно с этим важно понимать, что процессорное ядро работает на определенной частоте, которая зависит от техпроцесса (0.13 мкм, 0.18 мкм, и.т.д.), который применялся в ходе изготовления процессорного ядра. На сегодняшний день рынок компьютерных комплектующих предлагает не только *одноядерные процессоры*, но и более производительные *двухядерные и даже четырехядерные процессоры с поддержкой работы на самых высоких тактовых частотах*. Нужно отметить, что **количество ядер в одном процессоре зависит от модельного ряда**, которые создал производитель, к примеру, семейство Core 2 Duo сочетает в себе 2 ядра в одном процессоре, тогда, когда линейка процессоров Core 2 Quad – это уже четырехядерный процессор для выполнения множества задач. *К сожалению, многие люди ложно полагают, что объединение двух ядер в одном процессоре приводит к двукратному увеличению производительности компьютера, но на самом деле все не так. Многоядерные процессоры изначально создавались* ***для многозадачной среды****, тогда, когда использование всего потенциала двух или четырех ядер в однозадачной среде просто невозможно.*

У многих начинающих пользователей сложилось мнение, что при увеличении количества ядер, пропорционально увеличивается и производительность процессора. На самом деле это не так, вернее не совсем так. *Двухъядерные (и более) разработаны для возможности решения нескольких параллельных задач на одном компьютере, таким образом, появилась возможность распределить процессы между ядрами.*

Фактически – это означает, что одновременно на компьютер вы можете заниматься разными делами, отнимающими много ресурсов у системы, так как нагрузка будет распределяться между ядрами. Например, вы можете слушать музыку в интернете, одновременно работая с последней версией Photoshop.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

## *2.8. Назначение системных вызовов и драйверов в современных операционных системах*

Организация ОС как иерархия уровней. Уровни образуются группами функций операционной системы – файловая система, управление процессами и устройствами и т.п. Каждый уровень может взаимодействовать только со своим непосредственным соседом – выше- или нижележащим уровнем. Прикладные программы или модули самой операционной системы передают запросы вверх и вниз по этим уровням.

| Уровни | Функции | | | | |
|---|---|---|---|---|---|
| 7 | обработчик системных вызовов | | | | |
| 6 | файловая система 1 | | … | файловая система n | |
| 5 | виртуальная память | | | | |
| 4 | драйвер 1 | драйвер 2 | … | … | драйвер n |
| 3 | управление потоками | | | | |
| 2 | обработка прерываний, управление памятью | | | | |
| 1 | сокрытие низкоуровневой аппаратуры | | | | |

*Рисунок 5. Пример структуры многоуровневой системы*

***Файловая система*** **– это средство для организации хранения файлов на каком-либо носителе.**

Обслуживает файлы специальный модуль операционной системы, называемый ***драйвером файловой системы***. Каждый файл имеет имя, зарегистрированное в ***каталоге*** – оглавлении файлов.

***Драйвер*** *файловой системы* обеспечивает доступ к информации, записанной на магнитный диск, по имени файла и распределяет пространство на магнитном диске между файлами.

***Системный вызов*** *(англ. system call)* в программировании и вычислительной технике – обращение прикладной программы к ядру операционной системы для выполнения какой-либо операции.

С точки зрения программиста системный вызов обычно выглядит как вызов подпрограммы или функции из системной библиотеки.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

# ГЛАВА 3. КЛАССИФИКАЦИЯ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

Термин ***вычислительная система*** появился в начале – середине 60-х гг. при появлении ЭВМ III поколения. Это время знаменовалось переходом на новую элементную базу – интегральные схемы. *Следствием этого явилось появление новых технических решений: разделение процессов обработки информации и ее ввода-вывода, множественный доступ и коллективное использование вычислительных ресурсов в пространстве и во времени. Появились сложные режимы работы ЭВМ –* ***многопользовательская и многопрограммная обработка.***

Отражая эти новшества, и появился термин “вычислительная система”.

***Здесь является подчеркивание возможности построения параллельных ветвей в вычислениях, что не предусматривалось классической структурой ЭВМ.***

**Под *вычислительной системой* (ВС) будем понимать совокупность взаимосвязанных и взаимодействующих процессоров или ЭВМ, периферийного оборудования и программного обеспечения, предназначенную для подготовки и решения задач пользователей.**

Отличительной особенностью ВС по отношению к ЭВМ является наличие в них нескольких вычислителей, реализующих параллельную обработку.

***Создание ВС преследует следующие основные цели:***

- повышение производительности системы за счет ускорения процессов обработки данных,
- повышение надежности и достоверности вычислений,
- предоставление пользователям дополнительных сервисных услуг и т.д.

*Параллелизм в вычислениях в значительной степени усложняет управление вычислительным процессом, использование технических и программных ресурсов. Эти функции выполняет операционная система ВС.*

Существует большое количество ***признаков, по которым классифицируют вычислительные системы:***

- **по целевому назначению и выполняемым функциям,**
- **по типам и числу ЭВМ или процессоров,**
- **по архитектуре системы,**
- **режимам работы,**
- **методам управления элементами системы,**
- **степени разобщенности элементов вычислительной системы и др.**

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

Однако основными из них являются **признаки структурной и функциональной организации вычислительной системы.**

*По **назначению** вычислительные системы* делят на универсальные и специализированные. *Универсальные ВС* предназначаются для решения самых различных задач. *Специализированные системы* ориентированы на решение узкого класса задач.

*По **типу** вычислительные системы* различаются на многомашинные и многопроцессорные ВС. *Многомашинные* вычислительные *системы* (ММС) появились исторически первыми. Уже при использовании ЭВМ первых поколений возникали задачи повышения производительности, надежности и достоверности вычислений.

*Многопроцессорные вычислительные системы* (МПС) строятся при комплексировании нескольких процессоров. В качестве общего ресурса они имеют общую оперативную память (ООП). Параллельная работа процессоров и использование ООП обеспечиваются под управлением единой общей операционной системы. По сравнению с ММС здесь достигается наивысшая оперативность взаимодействия вычислителей-процессоров.

*По **типу ЭВМ или процессоров,*** используемых для построения ВС, различают однородные и неоднородные системы. Однородные системы предполагают комплексирование однотипных ЭВМ (процессоров), неоднородные – разнотипных. В *однородных системах* значительно упрощаются разработка и обслуживание технических и программных (в основном ОС) средств. В них обеспечивается возможность стандартизации и унификации соединений и процедур взаимодействия элементов системы. Упрощается обслуживание систем, облегчаются модернизация и их развитие. Вместе с тем существуют и *неоднородные ВС,* в которых комплексируемые элементы очень сильно отличаются по своим техническим и функциональным характеристикам.

*По **степени территориальной разобщенности*** вычислительных модулей ВС делятся на системы *совмещенного (сосредоточенного)* и *распределенного(разобщенного)* типов. Обычно такое деление касается только ММС. Многопроцессорные системы относятся к системам совмещенного типа. Более того, учитывая успехи микроэлектроники, это совмещение может быть очень глубоким. При появлении новых СБИС (сверхбольших интегральных схем) появляется возможность иметь в одном кристалле несколько параллельно работающих процессоров.

*По **методам управления элементами** ВС* различают *централизованные, децентрализованные и со смешанным управлением*. Помимо параллельных вычислений, производимых элементами системы, необходимо выделять ресурсы на обеспечение управления этими вычислениями. *В централизованных ВС* за это

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

отвечает главная, или диспетчерская, ЭВМ (процессор). Ее задачей являются распределение нагрузки между элементами, выделение ресурсов, контроль состояния ресурсов, координация взаимодействия. Централизованный орган управления в системе может быть жестко фиксирован или эти функции могут передаваться другой ЭВМ (процессору), что способствует повышению надежности системы. Централизованные системы имеют более простые ОС. *В децентрализованных системах* функции управления распределены между ее элементами. Каждая ЭВМ (процессор) системы сохраняет известную автономию, а необходимое взаимодействие между элементами устанавливается по специальным наборам сигналов. С развитием ВС и, в частности, сетей ЭВМ интерес к децентрализованным системам постоянно растет.

В системах *со смешанным управлением* совмещаются процедуры централизованного и децентрализованного управления. Перераспределение функций осуществляется в ходе вычислительного процесса исходя из сложившейся ситуации.

*По* ***принципу закрепления вычислительных функций*** за отдельными ЭВМ (процессорами) различают системы с *жестким и плавающим закреплением функций.* В зависимости от типа ВС следует решать задачи статического или динамического размещения программных модулей и массивов данных, обеспечивая необходимую гибкость системы и надежность ее функционирования.

*По* ***режиму работы*** *ВС* различают системы, работающие *в оперативном* и *неоперативном временных режимах.* Первые, как правило, используют режим реального масштаба времени. Этот режим характеризуется жесткими ограничениями на время решения задач в системе и предполагает высокую степень автоматизации процедур ввода-вывода и обработки данных.

Чтобы дать более полное представление о многопроцессорных вычислительных системах, помимо высокой производительности необходимо назвать и другие отличительные особенности.

Прежде всего, это ***необычные архитектурные решения, направленные на повышение производительности:***

- ***работа с векторными операциями,***
- ***организация быстрого обмена сообщениями между процессорами,***
- ***организация глобальной памяти в многопроцессорных системах и др.).***

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

*Понятие архитектуры высокопроизводительной системы является достаточно широким, поскольку под архитектурой можно понимать и способ параллельной обработки данных, используемый в системе, и организацию памяти, и топологию связи между процессорами, и способ исполнения системой арифметических операций*. Попытки систематизировать все множество архитектур впервые были предприняты в конце 60-х годов и продолжаются по сей день.

В 1966 г. М. Флинном (Flynn) был предложен чрезвычайно удобный подход к классификации архитектур вычислительных систем. В его основу было положено понятие потока, под которым понимается последовательность элементов, команд или данных, обрабатываемая процессором. Соответствующая система классификации основана на рассмотрении числа потоков инструкций и потоков данных и описывает четыре архитектурных класса:

- **ОКОД** - Вычислительная система с одиночным потоком команд и одиночным потоком данных (**SISD**, Single Instruction stream over a Single Data stream).
- **ОКМД** - Вычислительная система с одиночным потоком команд и множественным потоком данных (**SIMD**, Single Instruction, Multiple Data).
- **МКОД** - Вычислительная система со множественным потоком команд и одиночным потоком данных (**MISD**, Multiple Instruction Single Data).
- **МКМД** - Вычислительная система со множественным потоком команд и множественным потоком данных (**MIMD**, Multiple Instruction Multiple Data).

SISD (single instruction stream / single data stream) – одиночный поток команд и одиночный поток данных. К этому классу относятся последовательные компьютерные системы, которые имеют один центральный процессор, способный обрабатывать только один поток последовательно исполняемых инструкций. В настоящее время практически все высокопроизводительные системы имеют более одного центрального процессора, однако каждый из них выполняет несвязанные потоки инструкций, что делает такие системы комплексами SISD-систем, действующих на разных пространствах данных. Для увеличения скорости обработки команд и скорости выполнения арифметических операций может применяться конвейерная обработка. В случае векторных систем векторный поток данных следует рассматривать как поток из одиночных неделимых векторов.

MISD (multiple instruction stream / single data stream) – множественный поток команд и одиночный поток данных. Теоретически в этом типе машин множество инструкций должно выполняться над единственным потоком данных. До сих пор ни одной реальной машины, попадающей в данный класс, создано не было.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

SIMD (single instruction stream / multiple data stream) – одиночный поток команд и множественный поток данных. Эти системы обычно имеют большое количество процессоров, от 1024 до 16384, которые могут выполнять одну и ту же инструкцию относительно разных данных в жесткой конфигурации. Единственная инструкция параллельно выполняется над многими элементами данных.

MIMD (multiple instruction stream / multiple data stream) – множественный поток команд и множественный поток данных. Эти машины параллельно выполняют несколько потоков инструкций над различными потоками данных. В отличие от упомянутых выше многопроцессорных SISD-машин, команды и данные связаны, потому что они представляют различные части одной и той же задачи. Например, MIMD-системы могут параллельно выполнять множество подзадач с целью сокращения времени выполнения основной задачи. Большое разнообразие попадающих в данный класс систем делает классификацию Флинна не полностью адекватной.

***Классификация архитектур вычислительных систем нужна для того, чтобы понять особенности работы той или иной архитектуры, но она не является достаточно детальной, чтобы на нее можно было опираться при создании МВС, поэтому следует вводить более детальную классификацию, которая связана с различными архитектурами ЭВМ и с используемым оборудованием.***

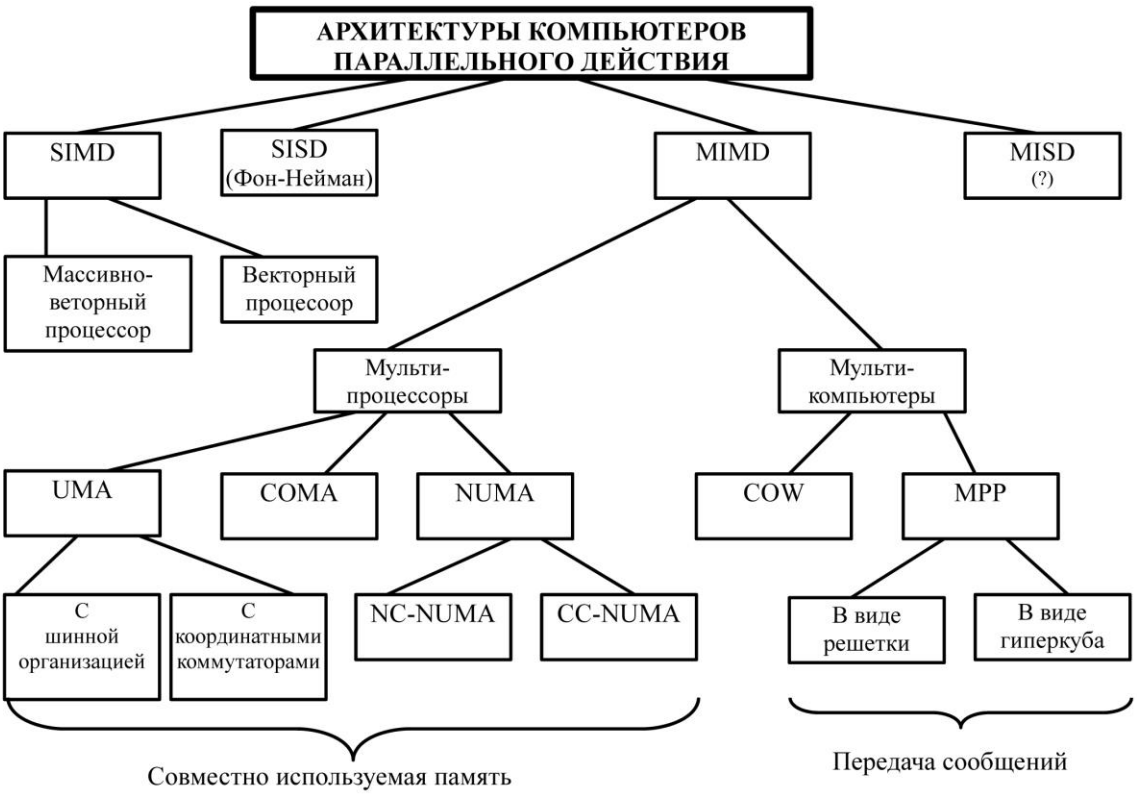



*Рисунок 6. Классификация архитектур вычислительных систем*

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

**Машины SIMD** распались на две подгруппы. В первую подгруппу попадают многочисленные суперкомпьютеры и другие машины, которые оперируют векторами, выполняя одну и ту же операцию над каждым элементом вектора. Во вторую подгруппу попадают машины типа ILLIAC IV, в которых главный блок управления посылает команды нескольким независимым АЛУ.

В нашей классификации категория MIMD распалась на мультипроцессоры (машины с памятью совместного использования) и мультикомпьютеры (машины с передачей сообщений). Существует три типа мультипроцессоров. Они отличаются друг от друга по способу реализации памяти совместного использования. Они называются **UMA (Uniform Memory Access – архитектура с однородным доступом к памяти), NUMA (NonUniform Memory Access – архитектура с неоднородным доступом к памяти)** и **COMA (Cache Only Memory Access – архитектура с доступом только к кэш-памяти).**

В машинах **UMA** каждый процессор имеет одно и то же время доступа к любому модулю памяти. Иными словами, каждое слово памяти можно считать с той же скоростью, что и любое другое слово памяти. Если это технически невозможно, самые быстрые обращения замедляются, чтобы соответствовать самым медленным, поэтому программисты не увидят никакой разницы. Это и значит «однородный». Такая однородность делает производительность предсказуемой, а этот фактор очень важен для написания эффективной программы.

Мультипроцессор **NUMA**, напротив, не обладает этим свойством. Обычно есть такой модуль памяти, который расположен близко к каждому процессору, и доступ к этому модулю памяти происходит гораздо быстрее, чем к другим. С точки зрения производительности очень важно, куда помещаются программа и данные. Машины **COMA** тоже с неоднородным доступом, но по другой причине.

Во вторую подкатегорию машин MIMD попадают мультикомпьютеры, которые в отличие от мультипроцессоров не имеют памяти совместного использования на архитектурном уровне. Другими словами, операционная система в процессоре мультикомпьютера не может получить доступ к памяти, относящейся к другому процессору, просто путем выполнения команды LOAD. Ему приходится отправлять сообщение и ждать ответа. Именно способность операционной системы считывать слово из отдаленного модуля памяти с помощью команды LOAD отличает мультипроцессоры от мультикомпьютеров. Как мы уже говорили, даже в мультикомпьютере пользовательские программы могут обращаться к другим модулям памяти с помощью команд LOAD и STORE, но эту иллюзию создает операционная система, а не аппаратное обеспечение. Разница незначительна, но очень важна. Так как мультикомпьютеры не имеют прямого доступа к отдаленным модулям памяти, они иногда называются машинами **NORMA (NO Remote Memory Access – без доступа к отдаленным модулям памяти).**

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

**Мультикомпьютеры** можно разделить на две категории. Первая категория содержит процессоры **MPP (Massively Parallel Processors – процессоры с массовым параллелизмом) –** дорогостоящие суперкомпьютеры, которые состоят из большого количества процессоров, связанных высокоскоростной коммуникационной сетью.

Вторая категория мультикомпьютеров включает рабочие станции, которые связываются с помощью уже имеющейся технологии соединения. Эти примитивные машины называются **NOW (Network of Workstations – сеть рабочих станций)** и **COW (Cluster of Workstattions – кластер рабочих станций).**

## *3.1. Альтернативные пути развития вычислительной техники*

И сегодня специалисты в разных областях науки и техники ищут альтернативные пути дальнейшего развития микроэлектроники.

На сегодняшний день вряд ли можно сказать, как именно он будет устроен самый “последний” компьютер.

Во многом благодаря компьютерной технике стало возможным появление и развитие таких современных наукоемких отраслей **как молекулярная биология, генная инженерия, квантовая физика и др., стала возможным обширная интеграция накопленных научных знаний.**

Бесспорно, все это не предел. Вопрос лишь в том, какие еще функции сможет взять на себя ЭВМ и как скоро это произойдет?

***Дальнейший прогресс будет двигаться в направлении дальнейшей миниатюризации ЭВМ с одновременным увеличением ее быстродействия.***

- Устройство будущих компьютеров будет основано на применении главным образом передовых отраслей широкого спектра научных дисциплин (**молекулярная электроника, молекулярная биология, робототехника), а также квантовой механики, органической химии и др.** А для их производства компьютеров будут необходимы значительные экономические затраты, в несколько десятки раз превышающие затраты на производство современных “классических” полупроводниковых компьютеров.
- Современные полупроводниковые компьютеры скоро исчерпают свой потенциал, и даже при условии перехода к трехмерной архитектуре микросхем их быстродействие будет ограничено значением **$10^{15}$ операций в секунду**.
- Скорость компьютерных вычислений достигнет значения **$10^{51}$ операций в секунду**.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

- Главным выводом можно считать тот факт, что разнообразие существующих на сегодняшний момент научных разработок в области микроэлектроники, а также обширности накопленных знаний в области других научных дисциплин позволяет надеяться на создание “суперкомпьютера” в сроки 100-300 лет. Т.е. можно сказать, что “в ближайшем будущем” мы сможем пользоваться идеальным ЭВМ.

Данные выводы могут вызвать некий скептизм, но не стоит отчаиваться, ведь область применения ЭВМ станет чрезвычайно обширной:

- по мере поступления рыночной информации автоматически управлять процессами производства продукции;
- накапливать человеческие знания и обеспечивать получение необходимой информации в течение нескольких минут;
- регулировать движение всех видов транспорта;
- ставить диагнозы в медицине;
- обрабатывать налоговые декларации;
- создавать новые виды продукции;
- вести домашнее хозяйство;
- *и главное – ЭВМ сможет вести диалог с человеком*.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

# ГЛАВА 4. ХАРАКТЕРИСТИКИ УСТРОЙСТВ ВВОДА-ВЫВОДА. ВЗАИМОДЕЙСТВИЕ УСТРОЙСТВ ПК

## *4.1. Состав устройств ввода-вывода в ПК. Порядок подключения внешних устройств к ПК*

**Внешняя память (ВЗУ) предназначена для длительного хранения программ и данных, и целостность её содержимого не зависит от того, включен или выключен компьютер.** В отличие от оперативной памяти, **внешняя память не имеет прямой связи с процессором.** Информация от ВЗУ к процессору и наоборот циркулирует примерно по следующей цепочке:

В состав внешней памяти компьютера входят:

- накопители на **жёстких магнитных дисках;**
- накопители на **гибких магнитных дисках;**
- накопители на **компакт-дисках;**
- накопители на **магнито-оптических компакт-дисках;**
- накопители на **магнитной ленте** (стримеры) и др.

**Накопители на гибких магнитных дисках**

*Гибкий диск* (*англ. floppy disk*), или лискета, – носитель небольшого объема информации, представляющий собой гибкий пластиковый диск в защитн

ой оболочке. Используется для переноса данных с одного компьютера на другой и для распространения программного обеспечения.

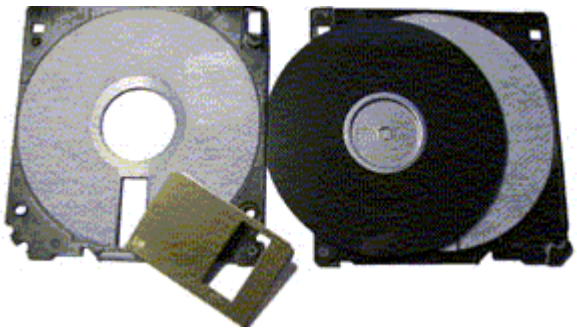


*Рисунок 7. Устройство дискеты*

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

Дискета состоит из круглой полимерной подложки, покрытой с обеих сторон магнитным окислом и помещенной в пластиковую упаковку, на внутреннюю поверхность которой нанесено очищающее покрытие. В упаковке сделаны с двух сторон радиальные прорези, через которые головки считывания/записи накопителя получают доступ к диску.

Способ записи двоичной информации на магнитной среде называется **магнитным кодированием.** Он заключается в том, что магнитные домены в среде выстраиваются вдоль дорожек в направлении приложенного магнитного поля своими северными и южными полюсами. Обычно устанавливается однозначное соответствие между двоичной информацией и ориентацией магнитных доменов.

Информация записывается по концентрическим ***дорожкам*** (*трекам*), которые делятся на ***секторы***. Количество дорожек и секторов зависит от типа и формата дискеты. Сектор хранит минимальную порцию информации, которая может быть записана на диск или считана. Ёмкость сектора постоянна и составляет 512 байтов.

*Рисунок 8. Поверхность магнитного диска*

В настоящее время наибольшее распространение получили **дискеты со следующими характеристиками:** диаметр 3,5 дюйма (89 мм), ёмкость 1,44 Мбайт, число дорожек 80, количество секторов на дорожках 18.

Дискета устанавливается в **накопитель на гибких магнитных дисках** (англ. **floppy-disk drive**), **автоматически в нем фиксируется**, после чего механизм накопителя раскручивается до частоты вращения 360 мин$^{-1}$. В накопителе вращается сама дискета, магнитные головки остаются неподвижными. Дискета вращается только при обращении к ней. Накопитель связан с процессором через **контроллер гибких дисков.**

Трехдюймовые дискеты, которые могут хранить **до 3 Гбайт** информации. Они изготовливаются по новой технологии **Nano2** и требуют специального оборудования для чтения и записи.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

## Накопители на жестких магнитных дисках

Если гибкие диски – это средство переноса данных между компьютерами, то **жесткий диск – информационный склад компьютера**.

***Накопитель на жёстких магнитных дисках*** **(англ. HDD – Hard Disk Drive) или винчестерский накопитель – это наиболее массовое запоминающее устройство большой ёмкости, в котором носителями информации являются круглые алюминиевые пластины – платтеры, обе поверхности которых покрыты слоем магнитного материала. Используется для постоянного хранения информации – программ и данных.**

*Рисунок 9. Винчестерский накопитель со снятой крышкой корпуса*

Как и у дискеты, рабочие поверхности платтеров разделены на кольцевые концентрические дорожки, а дорожки – на секторы. Головки считывания-записи вместе с их несущей конструкцией и дисками заключены в герметически закрытый корпус, называемый **модулем данных.** При установке модуля данных на дисковод он автоматически соединяется с системой, подкачивающей очищенный охлажденный воздух. **Поверхность** платтера имеет **магнитное покрытие** толщиной всего лишь в 1,1 мкм, а также **слой смазки** для предохранения головки от повреждения при опускании и подъёме на ходу. При вращении платтера над ним образуется **воздушный слой,** который обеспечивает воздушную подушку для зависания головки на высоте 0,5 мкм над поверхностью диска.

Винчестерские накопители имеют очень большую ёмкость: от 10 до 100 Гбайт. У современных моделей скорость вращения шпинделя (вращающего вала) обычно составляет 7200 об/мин, среднее время поиска данных 9 мс, средняя скорость передачи данных до 60 Мбайт/с. В отличие от дискеты, жесткий диск **вращается непрерывно**. Все современные накопители снабжаются **встроенным кэшем** (обычно 2 Мбайта), который существенно повышает их производительность. Винчестерский накопитель связан с процессором через **контроллер жесткого диска.**

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

## Накопители на компакт-дисках

Здесь носителем информации является CD-ROM (Compact Disk Read-Only Memory – компакт диск, из которого можно только читать).

CD-ROM представляет собой прозрачный полимерный диск диаметром 12 см и толщиной 1,2 мм, на одну сторону которого напылен светоотражающий слой алюминия, защищенный от повреждений слоем прозрачного лака. Толщина напыления составляет несколько десятитысячных долей миллиметра.

Информация на диске представляется в виде последовательности **впадин** (углублений в диске) и **выступов** (их уровень соответствует поверхности диска), расположеных на спиральной дорожке, выходящей из области вблизи оси диска. На каждом дюйме (2,54 см) по радиусу диска размещается 16 тысяч витков спиральной дорожки. Для сравнения – на поверхности жесткого диска на дюйме по радиусу помещается лишь несколько сотен дорожек. Емкость CD достигает **780 Мбайт**. Информация наносится на диск при его изготовлении и не может быть изменена.

CD-ROM обладают высокой удельной информационной емкостью, что позволяет создавать на их основе справочные системы и учебные комплексы с большой иллюстративной базой. Один CD по информационной емкости равен почти 500 дискетам. Считывание информации с CD-ROM происходит с достаточно высокой скоростью, хотя и заметно меньшей, чем скорость работы накопителей на жестком диске. CD-ROM просты и удобны в работе, имеют низкую удельную стоимость хранения данных, практически не изнашиваются, не могут быть поражены вирусами, с них невозможно случайно стереть информацию.

В отличие от магнитных дисков, компакт-диски имеют не множество кольцевых дорожек, а **одну – спиральную,** как у грампластинок. В связи с этим, угловая скорость вращения диска не постоянна. Она линейно уменьшается в процессе продвижения читающей лазерной головки к краю диска.

Для работы с CD-ROM нужно подключить к компьютеру **накопитель CD-ROM**, преобразующий последовательность углублений и выступов на поверхности CD-ROM в последовательность двоичных сигналов. Для этого используется **считывающая головка с микролазером и светодиодом.** Глубина впадин на поверхности диска равна четверти длины волны лазерного света. Если в двух последовательных тактах считывания информации луч света лазерной головки переходит с выступа на дно впадины или обратно, разность длин путей света в этих тактах меняется на полуволну, что вызывает усиление или ослабление совместно попадающих на светодиод прямого и отраженного от диска света.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

Если в последовательных тактах считывания длина пути света не меняется, то и состояние светодиода не меняется. В результате ток через светодиод образует последовательность двоичных электрических сигналов, соответствующих сочетанию впадин и выступов на дорожке.

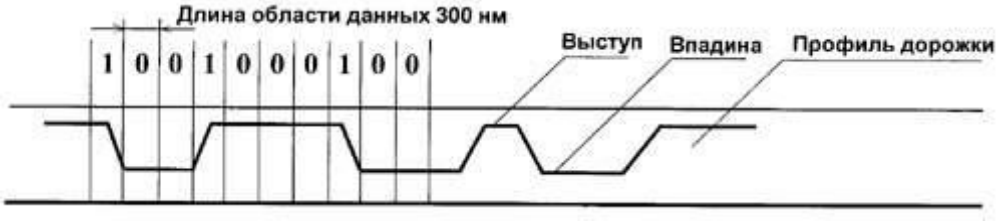



Рисунок 10. Профиль дорожки CD-ROM

Различная длина оптического пути луча света в двух последовательных тактах считывания информации соответствует двоичным единицам. Одинаковая длина соответствует двоичным нулям.

Сегодня почти все персональные компьютеры имеют накопитель CD-ROM. Но многие мультимедийные интерактивные программы слишком велики, чтобы поместиться на одном CD. **На смену технологии CD-ROM стремительно идет технология цифровых видеодисков DVD**. Эти диски имеют тот же размер, что и обычные CD, но вмещают **до 17 Гбайт данных**, т.е. по объему заменяют 20 стандартных дисков CD-ROM. На таких дисках выпускаются **мультимедийные игры и интерактивные видеофильмы** отличного качества, позволяющие зрителю просматривать эпизоды под разными углами камеры, выбирать различные варианты окончания картины, знакомиться с биографиями снявшихся актеров, наслаждаться великолепным качеством звука.

### Записывающие оптические и магнитооптические накопители

**Записывающий накопитель CD-R** (Compact Disk Recordable) способен, наряду с прочтением обычных компакт-дисков, записывать информацию на специальные оптические диски емкостью 650 Мбайт. В дисках CD-R отражающий слой выполнен из золотой пленки. Между этим слоем и поликарбонатной основой расположен регистрирующий слой из органического материала, темнеющего при нагревании. В процессе записи лазерный луч нагревает выбранные точки слоя, которые темнеют и перестают пропускать свет к отражающему слою, образуя участки, аналогичные впадинам. Накопители CD-R, благодаря сильному удешевлению, приобретают все большее распространение.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

*Рисунок 11. Накопитель CD-MO*

**Накопитель на магнито-оптических компакт-дисках CD-MO** (Compact Disk – Magneto Optical) (рис. 7). Диски CD-MO можно многократно использовать для записи. Ёмкость от 128 Мбайт до 2,6 Гбайт.

**Записывающий накопитель CD-R** (Compact Disk Recordable) способен, наряду с прочтением обычных компакт-дисков, записывать информацию на специальные оптические диски. Ёмкость 650 Мбайт.

**Накопитель WARM** (Write And Read Many times), позволяет производить многократную запись и считывание.

**Накопители на магнитной ленте (стримеры) и накопители на сменных дисках**

***Стример*** **(англ. tape streamer) – устройство для резервного копирования больших объёмов информации. В качестве носителя здесь применяются кассеты с магнитной лентой ёмкостью 1 – 2 Гбайта и больше.**

Стримеры позволяют записать на небольшую кассету с магнитной лентой огромное количество информации. Встроенные в стример средства аппаратного сжатия позволяют автоматически уплотнять информацию перед её записью и восстанавливать после считывания, что увеличивает объём сохраняемой информации.

Недостатком стримеров является их сравнительно низкая скорость записи, поиска и считывания информации.

В последнее время всё шире используются **накопители на сменных дисках,** которые позволяют не только увеличивать объём хранимой информации, но и переносить информацию между компьютерами. Объём сменных дисков – от сотен Мбайт до нескольких Гигабайт.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

*Рисунок 12. Накопитель на сменных дисках*

Что такое аудиоадаптер?

***Аудиоадаптер*** **(Sound Blaster или звуковая плата) это специальная электронная плата, которая позволяет записывать звук, воспроизводить его и создавать программными средствами с помощью микрофона, наушников, динамиков, встроенного синтезатора и другого оборудования.**

Аудиоадаптер содержит в себе два преобразователя информации:

- **аналого-цифровой,** который преобразует непрерывные (то есть, аналоговые) звуковые сигналы (речь, музыку, шум) в цифровой двоичный код и записывает его на магнитный носитель;
- **цифро-аналоговый,** выполняющий обратное преобразование сохранённого в цифровом виде звука в аналоговый сигнал, который затем воспроизводится с помощью акустической системы, синтезатора звука или наушников.

Профессиональные звуковые платы позволяют выполнять сложную обработку звука, обеспечивают стереозвучание, имеют собственное ПЗУ с хранящимися в нём сотнями тембров звучаний различных музыкальных инструментов. Звуковые файлы обычно имеют очень большие размеры. Так, трёхминутный звуковой файл со стереозвучанием занимает примерно 30 Мбайт памяти. Поэтому **платы Sound Blaster, помимо своих основных функций, обеспечивают автоматическое сжатие файлов.**

**Область применения звуковых плат** – компьютерные игры, обучающие программные системы, рекламные презентации, "голосовая почта" (voice mail) между компьютерами, озвучивание различных процессов, происходящих в компьютерном оборудовании, таких, например, как отсутствие бумаги в принтере и т.п.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

Что такое видеоадаптер и графический акселератор?

*Видеоадаптер* – это электронная плата, которая обрабатывает видеоданные (текст и графику) и управляет работой дисплея. Содержит видеопамять, регистры ввода вывода и модуль BIOS. Посылает в дисплей сигналы управления яркостью лучей и сигналы развертки изображения.

Наиболее распространенный видеоадаптер на сегодняшний день – адаптер SVGA (Super Video Graphics Array – супервидеографический массив), который может отображать на экране дисплея 1280x1024 пикселей при 256 цветах и 1024x768 пикселей при 16 миллионах цветов.

С увеличением числа приложений, использующих сложную графику и видео, наряду с традиционными видеоадаптерами широко используются разнообразные устройства компьютерной обработки видеосигналов:

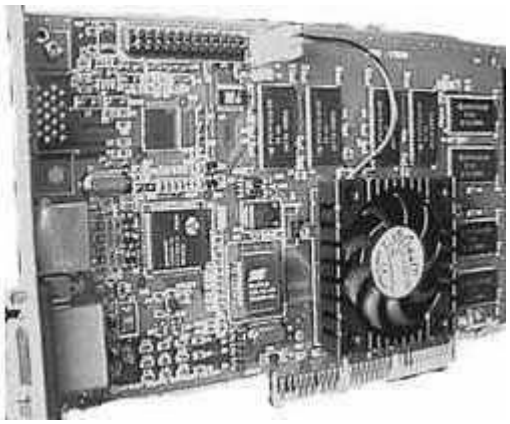


*Рисунок 13. Графический акселератор*

Графические акселераторы (ускорители) – специализированные графические сопроцессоры, увеличивающие эффективность видеосистемы. Их применение освобождает центральный процессор от большого объёма операций с видеоданными, так как акселераторы самостоятельно вычисляют, какие пиксели отображать на экране и каковы их цвета.

Фрейм-грабберы, которые позволяют отображать на экране компьютера видеосигнал от видеомагнитофона, камеры, лазерного проигрывателя и т. п., с тем, чтобы захватить нужный кадр в память и впоследствии сохранить его в виде файла.

TV-тюнеры – видеоплаты, превращающие компьютер в телевизор. TV-тюнер позволяет выбрать любую нужную телевизионную программу и отображать ее на экране в масштабируемом окне. Таким образом можно следить за ходом передачи, не прекращая работу.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

Что такое клавиатура?

*Клавиатура компьютера* – устройство для ввода информации в компьютер и подачи управляющих сигналов. Содержит стандартный набор клавиш печатной машинки и некоторые дополнительные клавиши – управляющие и функциональные клавиши, клавиши управления курсором и малую цифровую клавиатуру.

Все символы, набираемые на клавиатуре, немедленно отображаются на мониторе в позиции курсора (курсор – светящийся символ на экране монитора, указывающий позицию, на которой будет отображаться следующий вводимый с клавиатуры знак).

Наиболее распространена сегодня клавиатура с раскладкой клавиш QWERTY (читается "кверти"), названная так по клавишам, расположенным в верхнем левом ряду алфавитно-цифровой части клавиатуры:

*Рисунок 14. Клавиатура компьютера*

Такая клавиатура имеет 12 функциональных клавиш, расположенных вдоль верхнего края. Нажатие функциональной клавиши приводит к посылке в компьютер не одного символа, а целой совокупности символов. Функциональные клавиши могут программироваться пользователем. Например, во многих программах для получения помощи (подсказки) задействована клавиша F1, а для выхода из программы – клавиша F10.

Управляющие клавиши имеют следующее назначение:

- Enter – клавиша ввода;
- Esc (Escape – выход) клавиша для отмены каких-либо действий, выхода из программы, из меню и т.п.;
- Ctrl и Alt – эти клавиши самостоятельного значения не имеют, но при нажатии совместно с другими управляющими клавишами изменяют их действие;
- Shift (регистр) – обеспечивает смену регистра клавиш (верхнего на нижний и наоборот);

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

- **Insert** (вставлять) – **переключает режимы вставки** (новые символы вводятся посреди уже набранных, раздвигая их) **и замены** (старые символы замещаются новыми);
- **Delete** (удалять) – **удаляет символ** с позиции курсора;
- **Back Space** или ← — удаляет символ перед курсором;
- **Home** и **End** – обеспечивают **перемещение курсора в первую и последнюю позицию строки**, соответственно;
- **Page Up** и **Page Down** – обеспечивают **перемещение по тексту на одну страницу** (один экран) назад и вперед, соответственно;
- **Tab** – **клавиша табуляции**, обеспечивает перемещение курсора вправо сразу на несколько позиций до очередной позиции табуляции;
- **Caps Lock** – фиксирует верхний регистр, обеспечивает **ввод прописных букв вместо строчных**;
- **Print Screen** – обеспечивает **печать информации**, видимой в текущий момент на экране.
- **Длинная нижняя клавиша** без названия – предназначена **для ввода пробелов**.
- Клавиши ↑, ↓, ← и →служат для перемещения курсора **вверх, вниз, влево и вправо** на одну позицию или строку.

**Малая цифровая клавиатура** используется в двух режимах – **ввода чисел и управления курсором**. Переключение этих режимов осуществляется клавишей **Num Lock**.

Клавиатура содержит встроенный **микроконтроллер** (местное устройство управления), который выполняет следующие функции:

- последовательно опрашивает клавиши, считывая введенный сигнал и вырабатывая двоичный **скан-код** клавиши;
- управляет световыми индикаторами клавиатуры;
- проводит внутреннюю диагностику неисправностей;
- осуществляет взаимодействие с центральным процессором через порт ввода-вывода клавиатуры.

Клавиатура имеет **встроенный буфер** – промежуточную память малого размера, куда помещаются введённые символы. В случае переполнения буфера нажатие клавиши будет сопровождаться звуковым сигналом – это означает, что символ не введён (отвергнут). Работу клавиатуры поддерживают специальные программы, "зашитые" в BIOS, а также драйвер клавиатуры, который обеспечивает возможность ввода русских букв, управление скоростью работы клавиатуры и др.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

### Что такое видеосистема компьютера?

Видеосистема компьютера состоит из трех компонент:

- **монитор** (называемый также дисплеем);
- **видеоадаптер**;
- **программное обеспечение** (драйверы видеосистемы).

**Видеоадаптер** посылает в монитор сигналы управления яркостью лучей и синхросигналы строчной и кадровой развёрток. **Монитор** преобразует эти сигналы в зрительные образы. А **программные средства** обрабатывают видеоизображения – выполняют кодирование и декодирование сигналов, координатные преобразования, сжатие изображений и др.

**Монитор – устройство визуального отображения информации (в виде текста, таблиц, рисунков, чертежей и др.).**

Подавляющее большинство мониторов сконструированы на базе **электронно-лучевой трубки (ЭЛТ)**, и принцип их работы аналогичен принципу работы телевизора. Мониторы бывают алфавитно-цифровые и графические, монохромные и цветного изображения. Современные компьютеры комплектуются, как правило, цветными графическими мониторами.

### Монитор на базе электронно-лучевой трубки

Основной элемент дисплея – **электронно-лучевая трубка**. Её передняя, обращенная к зрителю часть с внутренней стороны покрыта **люминофором – специальным веществом, способным излучать свет при попадании на него быстрых электронов**.

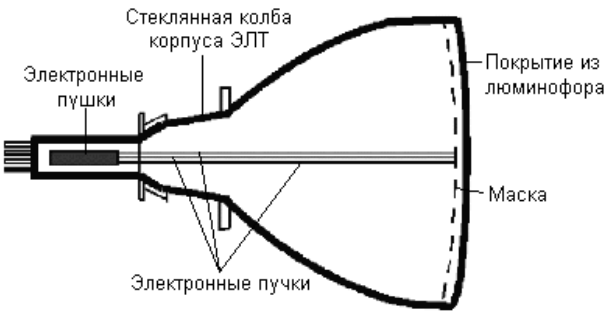



*Рисунок 15. Схема электронно-лучевой трубки*

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

Люминофор наносится в виде наборов точек трёх основных цветов – красного, зелёного и синего. Эти цвета называют основными, потому что их сочетаниями (в различных пропорциях) можно представить любой цвет спектра.

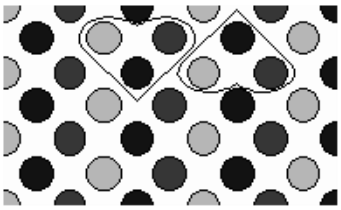


*Рисунок 16. Пиксельные триады*

Наборы точек люминофора располагаются по треугольным триадам. Триада образует пиксел – **точку, из которых формируется изображение** (англ. pixel – picture element, элемент картинки).

Расстояние между центрами пикселов называется **точечным шагом монитора**. Это расстояние существенно влияет на чёткость изображения. Чем меньше шаг, тем выше чёткость. Обычно в цветных мониторах шаг составляет 0,24 мм. При таком шаге глаз человека воспринимает точки триады как одну точку "сложного" цвета.

На противоположной стороне трубки расположены три (по количеству основных цветов) **электронные пушки.** Все три пушки "нацелены" на один и тот же пиксел, но каждая из них излучает поток электронов в сторону "своей" точки люминофора. Чтобы электроны беспрепятственно достигали экрана, из трубки откачивается воздух, а между пушками и экраном создаётся высокое электрическое напряжение, ускоряющее электроны. Перед экраном на пути электронов ставится **маска** – тонкая металлическая пластина с большим количеством отверстий, расположенных напротив точек люминофора. Маска обеспечивает попадание электронных лучей только в точки люминофора соответствующего цвета.

Величиной электронного тока пушек и, следовательно, яркостью свечения пикселов, управляет сигнал, поступающий с видеоадаптера.

На ту часть колбы, где расположены электронные пушки, надевается **отклоняющая система** монитора, которая заставляет электронный пучок пробегать поочерёдно все пикселы строчку за строчкой от верхней до нижней, затем возвращаться в начало верхней строки и т.д. Количество отображённых строк в секунду называется **строчной частотой развертки.** А частота, с которой меняются кадры изображения, называется **кадровой частотой развёртки.** Последняя не должна быть ниже 85 Гц, иначе изображение будет **мерцать**.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

## Жидкокристаллические мониторы

Все шире используются наряду с традиционными ЭЛТ-мониторами. **Жидкие кристаллы** – это особое состояние некоторых органических веществ, в котором они обладают текучестью и свойством образовывать пространственные структуры, подобные кристаллическим. Жидкие кристаллы могут изменять свою структуру и светооптические свойства под действием электрического напряжения. Меняя с помощью электрического поля ориентацию групп кристаллов и используя введённые в жидкокристаллический раствор вещества, способные излучать свет под воздействием электрического поля, можно создать высококачественные изображения, передающие более 15 миллионов цветовых оттенков.

Большинство ЖК-мониторов использует тонкую плёнку из жидких кристаллов, помещённую между двумя стеклянными пластинами. Заряды передаются через так называемую **пассивную матрицу** – сетку невидимых нитей, горизонтальных и вертикальных, создавая в месте пересечения нитей точку изображения (несколько размытого из-за того, что заряды проникают в соседние области жидкости).

**Активные матрицы** вместо нитей используют прозрачный экран из транзисторов и обеспечивают яркое, практически не имеющее искажений изображение. Экран при этом разделен на независимые ячейки, каждая из которых состоит из четырех частей (для трёх основных цветов и одна резервная). Количество таких ячеек по широте и высоте экрана называют **разрешением экрана.** Современные ЖК-мониторы имеют разрешение 642x480, 1280x1024 или 1024x768. Таким образом, экран имеет от 1 до 5 млн точек, каждая из которых управляется собственным транзистором. По компактности такие мониторы не знают себе равных. Они занимают в 2 – 3 раза меньше места, чем мониторы с ЭЛТ и во столько же раз легче; потребляют гораздо меньше электроэнергии и не излучают электромагнитных волн, воздействующих на здоровье людей.

## Сенсорный экран

*Рисунок 17. Сенсорный экран*

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

Общение с компьютером осуществляется путём прикосновения пальцем к определённому месту чувствительного экрана. Этим выбирается необходимый режим из меню, показанного на экране монитора. (**Меню** – это выведенный на экран монитора список различных вариантов работы компьютера, по которому можно сделать конкретный выбор.) Сенсорными экранами оборудуют рабочие места операторов и диспетчеров, их используют **в информационно-справочных системах** и т.д.

**Что такое принтер, плоттер, сканер?**

***Принтер*** **– печатающее устройство. Осуществляет вывод из компьютера закодированной информации в виде печатных копий текста или графики.**

Существуют тысячи наименований принтеров. Но основных видов принтеров три: матричные, лазерные и струйные.

*Рисунок 18. Матричный символ*

**Матричные принтеры** используют комбинации маленьких штырьков, которые бьют по красящей ленте, благодаря чему на бумаге остаётся отпечаток символа. Каждый символ, печатаемый на принтере, формируется из набора 9, 18 или 24 игл, сформированных в виде вертикальной колонки. Недостатками этих недорогих принтеров являются их шумная работа и невысокое качество печати.

**Лазерные принтеры** работают примерно так же, как ксероксы. Компьютер формирует в своей памяти "образ" страницы текста и передает его принтеру. Информация о странице проецируется с помощью лазерного луча на вращающийся барабан со светочувствительным покрытием, меняющим электрические свойства в зависимости от освещённости.

После засветки на барабан, находящийся под электрическим напряжением, наносится красящий порошок – **тонер,** частицы которого налипают на засвеченные участки поверхности барабана. Принтер с помощью специального горячего валика протягивает бумагу под барабаном; тонер переносится на бумагу и "вплавляется" в

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

неё, оставляя стойкое высококачественное изображение. **Цветные** лазерные принтеры пока очень дороги.

**Струйные принтеры** генерируют символы в виде последовательности **чернильных точек**. Печатающая головка принтера имеет крошечные **сопла,** через которые на страницу выбрызгиваются быстросохнущие чернила. Эти принтеры требовательны к качеству бумаги. **Цветные** струйные принтеры создают цвета, комбинируя чернила **четырех** основных цветов – **ярко-голубого, пурпурного, желтого и черного.**

Принтер связан с компьютером посредством **кабеля** принтера, один конец которого вставляется своим разъёмом в **гнездо** принтера, а другой – в **порт** принтера компьютера. **Порт – это разъём, через который можно соединить процессор компьютера с внешним устройством**.

Каждый принтер обязательно имеет свой **драйвер** – программу, которая способна переводить (транслировать) стандартные команды печати компьютера в специальные команды, требующиеся для каждого принтера.

***Плоттер*** **(графопостроитель) – устройство, которое чертит графики, рисунки или диаграммы под управлением компьютера.**

*Рисунок 19. Роликовый плоттер*

Плоттеры используются для получения сложных конструкторских чертежей, архитектурных планов, географических и метеорологических карт, деловых схем. **Плоттеры рисуют изображения с помощью пера.**

**Роликовые плоттеры** прокручивают бумагу под пером, а **планшетные плоттеры** перемещают перо через всю поверхность горизонтально лежащей бумаги.

Плоттеру, так же, как и принтеру, обязательно нужна специальная программа – **драйвер**, позволяющая прикладным программам передавать ему инструкции: поднять и опустить перо, провести линию заданной толщины и т.п.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

**Сканер – устройство для ввода в компьютер графических изображений. Создает оцифрованное изображение документа и помещает его в память компьютера.**

*Рисунок 20. Планшетный сканер*

Если принтеры выводят информацию из компьютера, то сканеры, наоборот, **переносят информацию с бумажных документов в память компьютера.** Существуют **ручные сканеры**, которые прокатывают по поверхности документа рукой, и **планшетные сканеры**, по внешнему виду напоминающие копировальные машины.

**Если при помощи сканера вводится текст, компьютер воспринимает его как картинку, а не как последовательность символов.** Для преобразования такого графического текста в обычный символьный формат используют **программы оптического распознавания образов.**

**Что такое модем и факс-модем?**

***Модем* – устройство для передачи компьютерных данных на большие расстояния по телефонным линиям связи.**

Цифровые сигналы, вырабатываемые компьютером, нельзя напрямую передавать по телефонной сети, потому что она предназначена для передачи человеческой речи – непрерывных сигналов звуковой частоты.

**Модем обеспечивает преобразование цифровых сигналов компьютера в переменный ток частоты звукового диапазона** – этот процесс называется **модуляцией**, **а также обратное преобразование,** которое называется **демодуляцией**. Отсюда название устройства: **модем** – **мо**дулятор/**дем**одулятор.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

*Рисунок 21. Схема реализации модемной связи*

Для осуществления связи один модем вызывает другой по номеру телефона, а тот отвечает на вызов. Затем модемы посылают друг другу сигналы, согласуя подходящий им обоим **режим связи**. После этого передающий модем начинает **посылать модулированные данные** с согласованными скоростью (количеством бит в секунду) и форматом. Модем на другом конце **преобразует полученную информацию в цифровой вид** и передает её своему компьютеру. Закончив сеанс связи, модем отключается от линии.

*Рисунок 22. Внешний модем*

Управление модемом осуществляется с помощью специального **коммутационного программного обеспечения**.

Модемы бывают **внешние**, выполненные в виде отдельного устройства, и **внутренние,** представляющие собой электронную плату, устанавливаемую внутри компьютера. Почти все модемы поддерживают и функции факсов.

**Факс – это устройство факсимильной передачи изображения по телефонной сети**. Название "факс" произошло от слова "факсимиле" (лат. fac simile – сделай подобное), означающее точное воспроизведение графического оригинала (подписи, документа и т.д.) средствами печати. Модем, который может передавать и получать данные как факс, называется **факс-модемом.**

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

Что такое манипуляторы?

Манипуляторы (мышь, джойстик и др.) – это специальные устройства, которые используются для **управления курсором.**

**Мышь** имеет вид небольшой коробки, полностью умещающейся на ладони. Мышь связана с компьютером кабелем через специальный блок – адаптер, и её движения преобразуются в соответствующие перемещения курсора по экрану дисплея. В верхней части устройства расположены управляющие кнопки (обычно их три), позволяющие задавать начало и конец движения, осуществлять выбор меню и т.п.

**Джойстик** – обычно это стержень-ручка, отклонение которой от вертикального положения приводит к передвижению курсора в соответствующем направлении по экрану монитора. Часто применяется в компьютерных играх. В некоторых моделях в джойстик монтируется датчик давления. В этом случае, чем сильнее пользователь нажимает на ручку, тем быстрее движется курсор по экрану дисплея.

**Трекбол** – небольшая коробка с шариком, встроенным в верхнюю часть корпуса. Пользователь рукой вращает шарик и перемещает, соответственно, курсор. В отличие от мыши, трекбол не требует свободного пространства около компьютера, его можно встроить в корпус машины.

**Дигитайзер** – устройство для преобразования готовых изображений (чертежей, карт) в цифровую форму. Представляет собой плоскую панель – **планшет,** располагаемую на столе, и специальный инструмент – **перо,** с помощью которого указывается позиция на планшете. При перемещении пера по планшету фиксируются его координаты в близко расположенных точках, которые затем преобразуются в компьютере в требуемые единицы измерения.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

## 4.2. Устройство персонального компьютера

Рассмотрим устройство компьютера на примере самой распространенной компьютерной системы – персонального компьютера. **Персональным компьютером** (ПК) называют сравнительно недорогой универсальный микрокомпьютер, рассчитанный на одного пользователя. Персональные компьютеры обычно проектируются на основе принципа открытой архитектуры.

**Принцип открытой архитектуры заключается в следующем:**

- **Регламентируются и стандартизируются только описание принципа действия компьютера и его конфигурация (определенная совокупность аппаратных средств и соединений между ними). Таким образом, компьютер можно собирать из отдельных узлов и деталей, разработанных и изготовленных независимыми фирмами-изготовителями.**
- **Компьютер легко расширяется и модернизируется за счёт наличия внутренних расширительных гнёзд, в которые пользователь может вставлять разнообразные устройства, удовлетворяющие заданному стандарту, и тем самым устанавливать конфигурацию своей машины в соответствии со своими личными предпочтениями.**

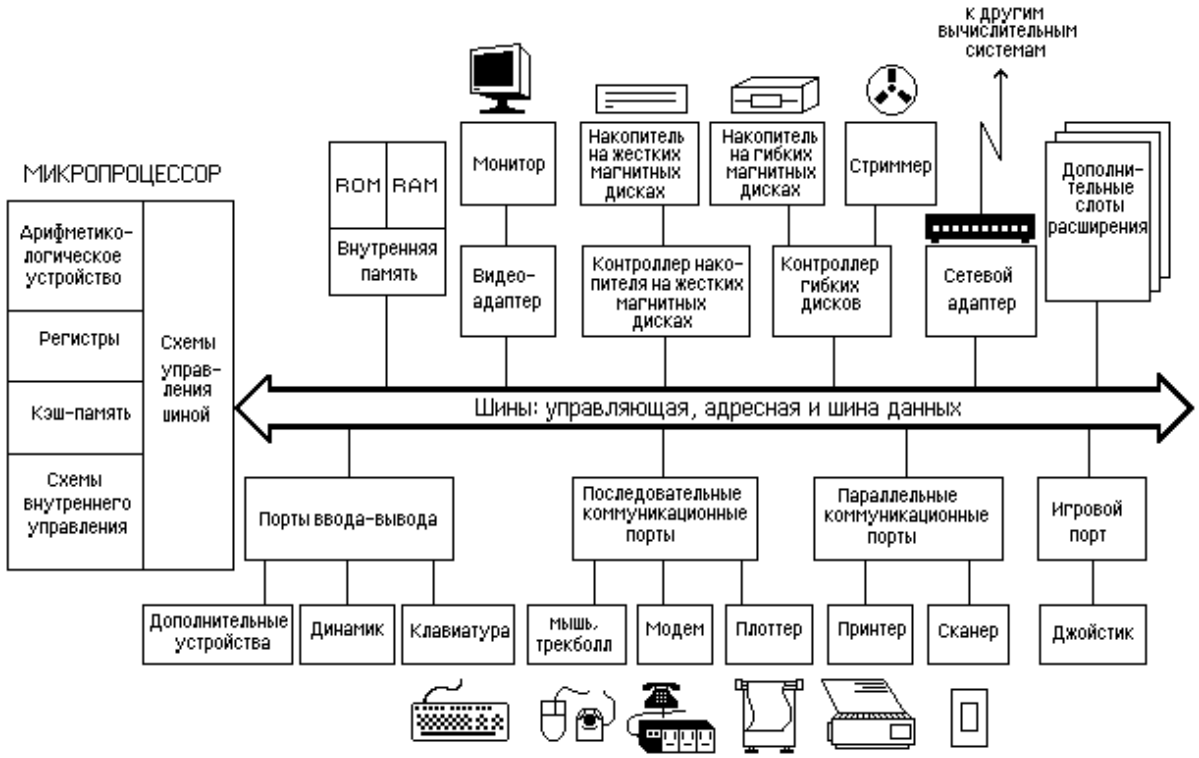



*Рисунок 23. Упрощённая блок-схема, отражающая основные функциональные компоненты компьютерной системы в их взаимосвязи*

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

## *4.3. Причины появления и развития режима реального времени*

**Реальное время** – режим работы автоматизированной системы обработки информации и управления, при котором учитываются ограничения на временны́е характеристики функционирования.

Примеры временных характеристик и связанных с ними ограничений:

- deadline – дедлайн – предельный срок завершения какой-либо работы;
- latency – латентность – время отклика (задержка реакции) системы на внешние события;
- jitter – джиттер – разброс значений времени отклика.
-

Также (преимущественно, в материалах рекламного и коммерческого характера) встречаются термины:

- *жёсткое реальное время* – режим работы системы, при котором нарушение временных ограничений равнозначно отказу системы;
- *мягкое реальное время* – режим работы системы, при котором нарушения временных ограничений приводят к снижению качества работы системы.

**Система реального времени** (СРВ) – это любая система, работающая в режиме реального времени.

Назначение систем, работающих в режиме реального времени, – взаимодействие с объектами внешнего (по отношению к системе) мира в темпе процессов, протекающих в этих объектах.

Как правило, система реального времени должна:

- либо по собственной инициативе воздействовать на внешние процессы в заданные моменты времени;
- либо реагировать на внешние события, происходящие на объекте, в течение заданных интервалов времени.

За своевременность воздействий на объект отвечает характеристика deadline. Задержка реакции на внешнее событие характеризуется значениями latency и jitter.

Примеры систем, работающих в режиме реального времени:

- АСУ ТП химического реактора;
- АСНИ в области ядерной физики;

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

- система обработки аудио- и видеопотоков при трансляции в прямом эфире;
- интерактивная компьютерная игра.

При создании систем реального времени приходится решать проблемы привязки внутрисистемных событий к моментам времени, своевременного захвата и освобождения системных ресурсов, синхронизации вычислительных процессов, буферизации потоков данных и т. п. Системы реального времени обычно используют специализированное оборудование (например, таймеры) и программное обеспечение (например, Операционные системы реального времени).

## *4.4. Основные отличия однопрограммных и многопрограммных режимов работы компьютера*

### *Режимы работы ЭВМ*

Одним из путей повышения производительности вычислительных машин и систем, их эффективности и надежности является использование различных форм реализации параллельного способа функционирования основных устройств ЭВМ.

Режимы работы вычислительных систем подразделяются на **однопрограммные и многопрограммные.**

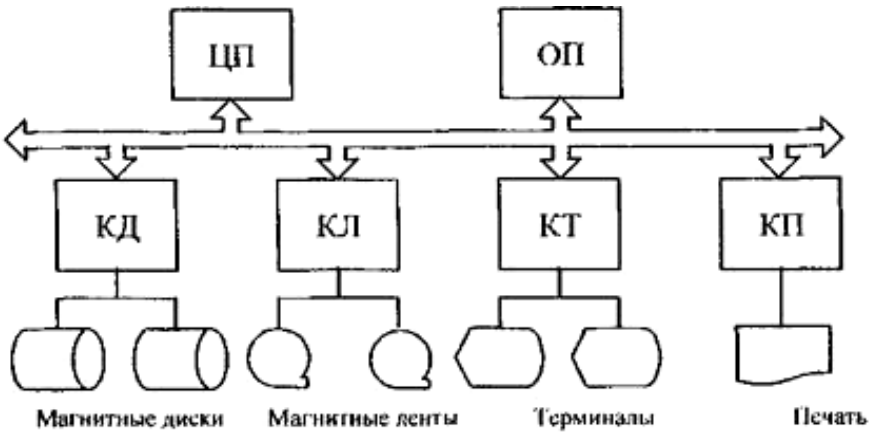



*Рисунок 24. ЭВМ с общей шиной.*

В *однопрограммном режиме* в оперативной памяти присутствует только одна программа пользователя, которая выполняется от начала до конца. Обычно в ОП обязательно присутствуют еще программы операционной системы. Возможно совмещение работы центрального процессора с операциями ввода-

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

вывода. Однако программа не может продолжить свою работу до завершения операции передачи данных с ВУ, поэтому появляются простои в работе процессора, которые значительно снижают эффективность вычислительной системы.

В *многопрограммном режиме* в памяти расположено несколько пользовательских программ. Процессор выполняет программы путем последовательного перехода во времени от одной программы к другой. Моменты переключения должны выбираться так, чтобы повысить общую производительность вычислительной системы. Чаще всего переключение происходит, когда одна программа ожидает окончания операций ввода-вывода. Чтобы процессор не простаивал, ему можно поручить выполнение другой программы.

По стратегии обслуживания пользователя различают ***режимы индивидуального пользования, пакетной обработки и коллективного пользования.***

- *Режим индивидуального использования* был характерен для ЭВМ первого поколения. Пользователем был программист, он сам работал за пультом управления. Он мог вводить данные в память, выбирать точки старта и остановки программы, наблюдал по индикаторам на пульте за ходом ее выполнения. Этот режим удобен программисту, но дорогостоящее по тем временам оборудование ЭВМ использовалось очень неэффективно: когда программист обдумывает дальнейшие действия – ЭВМ стоит.

  В дальнейшем режим индивидуального пользования возродился в первых персональных компьютерах, которые тоже работали в однопрограммном режиме, но управление ими было на более высоком уровне: не адресами и ячейками, а командами, вводимыми с терминала.
- Стремление повысить эффективность использования оборудования ЭВМ привело к созданию *пакетного режима обслуживания.* Здесь работой ЭВМ управляет оператор, пользователь не имеет доступа к пульту. Пользователь оформляет задание для ЭВМ на внешних носителях (раньше использовались перфокарты или перфоленты), используя язык управления заданиями, и отдает его оператору ЭВМ. Для каждого задания указываются класс (характер программы) и приоритет (срочность). Все задания вводятся оператором в ЭВМ и накапливаются в ее внешней памяти. Операционная система последовательно в порядке приоритета либо по заранее составленному расписанию выполняет накопленный пакет заданий без участия оператора.

  Режим пакетной обработки может быть реализован в однопрограммном и многопрограммном режимах работы ЭВМ. Многопрограммная пакетная обработка обеспечивает высокую степень загрузки оборудования, но при

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

этом из-за отсутствия непосредственной связи между пользователем и системой производительность и эффективность труда самих пользователей снижается по сравнению с индивидуальным обслуживанием.

- Появление индивидуальных устройств общения пользователей с ЭВМ – терминалов – позволило найти выход из положения – *режим коллективного пользования.* В этом режиме возможен одновременный доступ нескольких независимых пользователей к вычислительной системе. Каждому пользователю предоставляется терминал, с помощью которого он устанавливает связь с вычислительной системой коллективного пользования (ВСКП) и выполняет программы. Терминалы могут быть установлены на значительном удалении от ЭВМ.

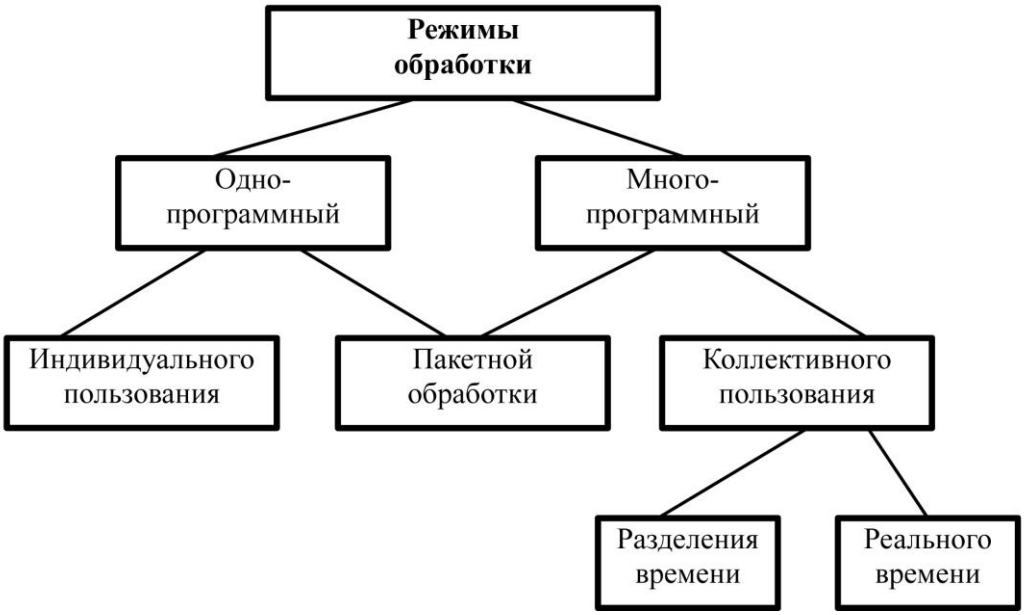



*Рисунок 25. Режимы работы ЭВМ.*

Пользователь общается с ЭВМ посылкой команд (запросов). Команда вызывает выполнение программы. По ходу ее выполнения ЭВМ передает на терминал результаты и сообщения о ходе выполнения программы. Просмотрев результаты выполнения команды, пользователь посылает новую команду. Между ЭВМ и пользователем ведется диалог, поэтому такой режим часто называют *диалоговым или интерактивным режимом.*

В простых ВСКП обработка запросов производится последовательно. При выполнении запроса одного пользователя другие запросы ожидают своей очереди. Длинные программы значительно увеличивают время ожидания других даже коротких запросов.

*Режим разделения времени* позволяет сократить время ожидания коротких запросов. В этом режиме каждому запросу выделяется для выполнения временной интервал *(квант).* Большинство программ коротких запросов успевают выполниться

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

за время кванта. Если программа не заканчивается за квант, процессор передается следующей программе. Обработка запросов напоминает сеанс одновременной игры в шахматы, когда гроссмейстер, сделав очередной ход, переходит к следующему игроку. Режим разделения времени обеспечивает уменьшение времени ожидания коллектива пользователей. Поскольку размер кванта невелик (по человеческим меркам, но ЭВМ за время кванта может выполнить несколько тысяч команд), создается впечатление индивидуальной работы каждого пользователя с ЭВМ.

*Режим реального времени* используется для программ управления технологическими процессами. Для них существует предельно допустимое время, в течение которого должна быть выполнена та или иная программа управления объектом. В противном случае может наступить аварийная ситуация, которая будет иметь катастрофические последствия. При поступлении запроса на выполнение программы реального времени выполнение всех других программ приостанавливается, и все ресурсы вычислительной системы отдаются программе реального времени. Программы реального времени обычно выполняются периодически (раз в секунду или более редко) и не занимают всего времени работы ЭВМ. В промежутке между выполнением программ реального времени могут выполняться диалоговые программы и программы пакетной обработки.

**Важнейшей характеристикой, определяющей быстродействие процессора, является тактовая частота, то есть количество тактов в секунду. Такт – это промежуток времени между началами подачи двух последовательных импульсов специальной микросхемой – генератором тактовой частоты, синхронизирующим работу узлов компьютера**

***Различные формы многопрограммных (мультипрограммных) режимов работы различаются*** в основном *значимостью различного рода ресурсов* и *правилами перехода от обслуживания одной программы пользователя к другой.* Эти правила отличаются условиями прерывания текущей программы и условиями выбора новой программы из очереди, которой передается управление.

**Различают следующие виды многопрограммной работы:**

- **классическое мультипрограммирование,**
- **режим разделения времени,**
- **режим реального времени**
- **и целый ряд производных от них.**

*Режим классического мультипрограммирования, или пакетной обработки,* применительно к однопроцессорным ЭВМ является основой для построения всех других видов многопрограммной работы. Режим имеет целью обеспечить минимальное время обработки пакета заданий и максимально загрузить процессор.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

Пакет заданий упорядочивается в соответствии с приоритетами заданий, и обслуживание программ ведется в порядке очередности. Обычно процессор обслуживает наиболее приоритетную программу. Как только ее решение завершается, процессор переключается на следующую по приоритетности программу. В этом режим во многом похож на режим косвенного доступа. В режиме мультипрограммирования имеется существенное отличие. Если при обслуживании наиболее приоритетной программы создается ситуация, что вычисления не могут быть продолжены (например, требуется ввести дополнительные данные), то прерывание обслуживания сопровождается передачей управления следующей по приоритетности программе. Но как только условия, препятствующие продолжению наиболее приоритетной задачи, отпадут, процессор вновь возвращается к продолжению решения ранее прерванной программы.

## *4.5. Виды прерываний, реализуемые в ПК*

***Современная ЭВМ представляет собой комплекс автономных устройств, каждое из которых выполняет свои функции под управлением местного устройства управления независимо от других устройств машины.*** Включает устройство в работу центральный процессор. Он передает устройству команду и все необходимые для ее исполнения параметры. После начала работы устройства центральный процессор отключается от него и переходит к обслуживанию других устройств или к выполнению других функций.

Можно считать, что центральный процессор *переключает свое внимание* с устройства на устройство и с функции на функцию. На что именно обращено внимание ЦП в каждый данный момент, определяется выполняемой им программой.

Во время работы в ЦП поступает (и вырабатывается в нем самом) большое количество различных сигналов. Сигналы, которые выполняемая в ЦП программа способна воспринять, обработать и учесть, составляют *поле зрения* ЦП или другими словами – входят в зону его внимания.

Для того чтобы ЦП, выполняя свою работу, имел возможность реагировать на события, происходящие вне его зоны внимания, наступления которых он “не ожидает”, существует ***система прерываний ЭВМ***. При отсутствии системы прерываний все заслуживающие внимания события должны находиться в поле зрения процессора, что сильно усложняет программы и требует большой их избыточности. Кроме того, поскольку момент наступления события заранее не известен, процессор в ожидании какого-либо события может находиться длительное время, и чтобы не пропустить его появления, ЦП не может “отвлекаться” на выполнение какой-либо другой работы. Такой режим работы

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

(режим сканирования ожидаемого события) связан с большими потерями времени ЦП на ожидание.

Таким образом, система прерываний позволяет микропроцессору выполнять основную работу, не отвлекаясь на проверку состояния сложных систем при отсутствии такой необходимости, или прервать выполняемую работу и переключиться на анализ возникшей ситуации сразу после ее появления.

В зависимости от места нахождения источника прерываний они могут быть разделены на:

- ***внутренние (программные и аппаратурные)*** и
- ***внешние*** прерывания (поступающие в ЭВМ от внешних источников, например, от клавиатуры или модема).

***Принцип действия системы прерываний*** **заключается в следующем**: при выполнении программы после каждого рабочего такта микропроцессора изменяются содержимое регистров, счетчиков, состояние отдельных управляющих триггеров, т.е. изменяется состояние процессора. Информация о состоянии процессора лежит в основе многих процедур управления вычислительным процессом. Не вся информация одинаково актуальна, есть существенные элементы, без которых невозможно продолжение работы. Эта информация должна сохраняться при каждом “переключении внимания процессора”.

Совокупность значений наиболее существенных информационных элементов называется *вектором состояния* или *словом состояния процессора* (в некоторых случаях она называется *словом состояния программы).*

При возникновении события, требующего немедленной реакции со стороны машины, ЦП прекращает обработку текущей программы и переходит к выполнению другой программы, специально предназначенной для данного события, по завершении которой возвращается к выполнению отложенной программы. **Такой режим работы называется** ***прерыванием.***

Каждое событие, требующее прерывания, сопровождается специальным сигналом, который называется ***запросом прерывания.*** Программа, затребованная запросом прерывания, называется ***обработчиком прерывания.***

Запросы на прерывание могут возникать из-за сбоев в аппаратуре (зафиксированных схемами контроля), переполнения разрядной сетки, деления на нуль, выхода за установленные для данной программы области памяти, затребования периферийным устройством операции ввода-вывода, завершения этой операции ввода-вывода или возникновения при этой операции особых условий и т.д.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

Некоторые из этих запросов порождаются самой программой, но время их возникновения невозможно предсказать заранее.

При наличии нескольких источников запросов прерывания часть из них может поступать одновременно. Поэтому в ЭВМ устанавливается определенный порядок (дисциплина) обслуживания поступающих запросов. Кроме того, в ЭВМ **предусматривается *возможность разрешать* или *запрещать* прерывания определенных видов.**

ПЭВМ может выполнять 256 различных прерываний, каждое из которых имеет свой номер (двухразрядное шестнадцатеричное число).

**Все прерывания делятся на две группы:**

- **прерывания с номера 00h по номер IFh называются *прерываниями базовой системы ввода-вывода*** (BIOS -Basic Input-Output System);
- **прерывания с номера 20h по номер FFh называются *прерываниями DOS.***

Прерывания DOS имеют более высокий уровень организации, чем прерывания BIOS, они строятся на использовании модулей BIOS в качестве элементов.

**Прерывания делятся на три типа: *аппаратурные, логические и программные. Аппаратурные*** прерывания вырабатываются устройствами, требующими внимания микропроцессора: прерывание **№ 2** – отказ питания; **№ 8** – от таймера; № 9 – от клавиатуры; **№ 12** – от адаптера связи; **№ 14** – от НГМД; **№ 15**- от устройства печати и др.

Запросы на ***логические*** прерывания вырабатываются внутри микропроцессора при появлении “нештатных” ситуаций: прерывание **№ 0** – при попытке деления на 0; **№ 4** – при переполнении разрядной сетки арифметико-логического устройства; **№ 1** – при переводе микропроцессора в пошаговый режим работы; **№ 3** – при достижении программой одной из контрольных точек. Последние два прерывания используются отладчиками программ для организации пошагового режима выполнения программ (трассировки) и для остановки программы в заранее намеченных контрольных точках.

## *4.6. Реализация в арифметико-логическом устройстве алгоритмов основных операций*

**Арифметико-логическое устройство** (АЛУ) (англ. *arithmetic and logic unit, ALU*) – блок процессора, который под управлением *устройства управления* (УУ) служит для выполнения арифметических и логических преобразований (начиная от элементарных) над данными, представляемыми в виде машинных слов, называемыми в этом случае операндами.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

Функции регистров, входящих в арифметико-логическое устройство:

- Рг1 – аккумулятор (или аккумуляторы) – главный регистр АЛУ, в котором образуется результат вычислений;
- Рг2, Рг3 – регистры операндов (слагаемого / сомножителя / делителя / делимого и др.) в зависимости от выполняемой операции;
- Рг4 – регистр адреса (или адресные регистры), предназначенные для запоминания (бывает, что формирования) адреса операндов результата;
- Рг6 – индексных регистров, содержимое которых используется для формирования адресов;
- Рг7 – вспомогательных регистров, которые по желанию программиста могут быть аккумуляторами, индексными регистрами или использоваться для запоминания промежуточных результатов.

Часть операционных регистров могут быть адресованы в команде для выполнения операций с их содержимым и их называют программно-доступными. К таким регистрам относятся: сумматор, индексные регистры и некоторые вспомогательные регистры. Остальные регистры нельзя адресовать в программе, то есть они являются программно-недоступными.

Такая сложная логическая структура АЛУ может характеризоваться количеством отличающихся друг от друга микроопераций, которые необходимы для выполнения всего комплекса задач, поставленных перед арифметико-логическим устройством. На входе каждого регистра собраны соответствующие логические схемы, обеспечивающие такие связи между регистрами, что позволяет реализовать заданные микрооперации. Выполнение операций над словами сводится к выполнению определённых микроопераций, которые сводятся в свою очередь … управляют передачей слов в АЛУ и действиями по преобразованию слов. Порядок выполнения микрокоманд определяется алгоритмом выполнения операций. То есть, связи между регистрами АЛУ и их функциями зависят в основном от принятой методики выполнения логических операций, в том числе арифметических или специальной арифметики.

## Операции в АЛУ

Все выполняемые в АЛУ операции являются логическими операциями (функциями), которые можно разделить на следующие группы:

- операции двоичной арифметики для чисел с фиксированной точкой;

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

- операции двоичной (или шестнадцатеричной) арифметики для чисел с плавающей точкой;
- операции десятичной арифметики;
- операции индексной арифметики (при модификации адресов команд);
- операции специальной арифметики;
- операции над логическими кодами (логические операции);
- операции над алфавитно-цифровыми полями.

Современные ЭВМ общего назначения обычно реализуют операции всех приведённых выше групп, а малые и микроЭВМ, микропроцессоры и специализированные ЭВМ часто не имеют аппаратуры арифметики чисел с плавающей точкой, десятичной арифметики и операций над алфавитно-цифровыми полями. В этом случае эти операции выполняются специальными подпрограммами.

К арифметическим операциям относятся сложение, вычитание, вычитание модулей («короткие операции») и умножение и деление («длинные операции»). Группу логических операций составляют операции дизъюнкция (логическое ИЛИ) и конъюнкция (логическое И) над многоразрядными двоичными словами, сравнение кодов на равенство. Специальные арифметические операции включают в себя нормализацию, арифметический сдвиг (сдвигаются только цифровые разряды, знаковый разряд остаётся на месте), логический сдвиг(знаковый разряд сдвигается вместе с цифровыми разрядами). Обширна группа операций редактирования алфавитно-цифровой информации. Каждая операция в АЛУ является логической функцией или последовательностью логических функций описываемыхдвоичной логикой для двоичных ЭВМ, троичной логикой для троичных ЭВМ, четверичной логикой для четверичных ЭВМ, …, десятичной логикой для десятичных ЭВМ и т. д..

## Классификация АЛУ

*По способу действия над операндами АЛУ делятся на последовательные и параллельные*. В последовательных АЛУ операнды представляются в последовательном коде, а операции производятся последовательно во времени над их отдельными разрядами. В параллельных АЛУ операнды представляются параллельным кодом и операции совершаются параллельно во времени над всеми разрядами операндов.

По способу представления чисел различают АЛУ:

1. для чисел с фиксированной точкой;
2. для чисел с плавающей точкой;
3. для десятичных чисел.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

*По характеру использования элементов и узлов АЛУ делятся на блочные и многофункциональные*. В блочном АЛУ операции над числами с фиксированной и плавающей точкой, десятичными числами и алфавитно-цифровыми полями выполняются в отдельных блоках, при этом повышается скорость работы, так как блоки могут параллельно выполнять соответствующие операции, но значительно возрастают затраты оборудования. В многофункциональных АЛУ операции для всех форм представления чисел выполняются одними и теми же схемами, которые коммутируются нужным образом в зависимости от требуемого режима работы.

По своим функциям АЛУ является операционным блоком, выполняющим микрооперации, обеспечивающие приём из других устройств (например, памяти) операндов, их преобразование и выдачу результатов преобразования в другие устройства. Арифметико-логическое устройство управляется управляющим блоком, генерирующим управляющие сигналы, инициирующие выполнение в АЛУ определённых микроопераций. Генерируемая управляющим блоком последовательность сигналов определяется кодом операции команды и оповещающими сигналами.

## *4.7. Реализация программного принципа управления в управляющем устройстве*

**Принцип программного управления. Из него следует, что программа состоит из набора команд, которые выполняются процессором автоматически друг за другом в определенной последовательности.**

Выборка программы из памяти осуществляется с помощью **счетчика команд**. Этот регистр процессора **последовательно увеличивает хранимый в нем адрес очередной команды на длину команды**.

А так как команды программы расположены в памяти друг за другом, то тем самым организуется выборка цепочки команд из последовательно расположенных ячеек памяти.

Если же нужно после выполнения команды перейти не к следующей, а к какой-то другой, используются команды **условного** или **безусловного переходов**, которые **заносят в счетчик команд номер ячейки памяти, содержащей следующую команду**. Выборка команд из памяти прекращается после достижения и выполнения команды ***"стоп"***.

Таким образом, **процессор исполняет программу автоматически, без вмешательства человека**.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

# САМОСТОЯТЕЛЬНАЯ РАБОТА СТУДЕНТА ПО ИЗУЧЕНИЮ УЧЕБНОЙ ДИСЦИПЛИНЫ

## Задача 1

**С помощью данной формулы опишите *конфигурацию* системного блока:**

а) ПЭВМ X5000MB: Core i5-2400/ 4 Гб/ 1 Тб/ 1 Гб GeForce GTX560Ti/ DVDRW/ Win7 Premium

б) P4–1000/128 SDRAM (PC133)/13Gb/32Mb AGP/50x/SBLivel28

## Задача 2

**На основании данной конфигурации системного блока составьте формулу описания ПЭВМ для прайс-листа.**

| | Конфигурация |
|---|---|
| Case | Minitower INWIN Z589 <Silver-Black> MicroATX 400W (24+4пин) |
| CPU | CPU Intel Core i3-2120 3.3 ГГц/SVGA/0.5+ 3Мб/5 ГТ/с LGA1155 |
| Cooler | Arctic Cooling Alpine 11 Pro rev.2 Cooler (775/1155, 500-2000об/мин, 23.5дБ, Al) |
| Cooler | Arctic Cooling Arctic F9 Pro (3пин, 92x92x34mm, 23.5дБ, 2000об/мин) |
| RAM | 2 шт. Original SAMSUNG DDR-III DIMM 2Gb <PC3-10600> |
| HDD | HDD 1 Tb SATA 6Gb/s Seagate Barracuda <ST1000DM003> 7200 rpm 64Mb |
| Video | 1Gb <PCI-E> DDR-5 Sapphire <RADEON HD7770 GHZ Ed.> (OEM) DVI+HDMI+DualminiDP+Crossfire |
| CD ROM | DVD RAM & DVD±R/RW & CDRW Optiarc AD-7280S <Black> SATA (OEM) |
| FDD | Sema <SFD-321F/TS41UB Black>3.5" Internal USB2.0 CF/MD/xD/MMC/SD/MS(/Pro/Duo)Card Reader/Writer+1portUSB2.0 |
| M/B | GigaByte GA-H67MA-USB3-B3 rev1.0 (RTL) LGA1155 <H67> 2xPCI-E+Dsub+DVI+HDMI GbLAN SATA RAID MicroATX 4DDR-III |
| ПО | Microsoft Windows 7 Домашняя расширенная SP1 32&64-bit Рус |

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

# КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Каковы основные этапы развития средств вычислительной техники?
2. Какими качественными отличиями характеризуются поколения ЭВМ?
3. Каков порядок подготовки задачи к решению ее на ЭВМ?
4. Каков порядок обработки информации в ЭВМ?
5. Каковы особенности структуры вычислительной системы?
6. Как взаимодействуют между собой устройства ПК?
7. Каковы особенности фон-Неймановской структуры ЭВМ?
8. В чем смысл принципа программного управления?
9. Какова роль программного обеспечения в организации вычислительного процесса?
10. Каковы основные функции операционных систем?
11. Какие системы счисления используются в компьютере и в чем состоят особенности их применения?
12. Какова связь логических выражений алгебры логики со схемами ЭВМ?
13. Как классифицируются элементы ЭВМ?
14. Какова логика работы сложных логических схем: регистров, счетчиков, сумматоров и др.?
15. Как компьютер управляет последовательностью выполнения команд?
16. Какие режимы работы используются в ПК и в ВС?
17. Какие архитектурные решения необходимы для организации многопрограммного режима работы ЭВМ?
18. Какие способы адресации применяются в ПК и почему?
19. Как реализуется программный принцип управления в ЦУУ?
20. Как реализуются в АЛУ алгоритмы основных операций?
21. Что такое конвейер команд?
22. Какие устройства образуют ядро компьютера?
23. Чем отличаются RISK- и CISK-процессоры?
24. Основные регистры процессора компьютеров IBM PC.
25. Зачем нужна иерархическая структура памяти в компьютере?
26. Чем объясняется использование матричной организации ОП?
27. Как определяется физический адрес ОП в реальном и защищенных режимах?
28. Зачем нужна КЭШ-память?
29. Для чего необходима стековая память и как она организована?
30. Какие устройства ввода-вывода применяются в ПК?
31. Какие характеристики устройств ввода-вывода важно знать пользователю и почему?
32. С какой целью в ЭВМ реализован режим прерываний?
33. Какие виды прерываний реализуются в ПК?
34. Какие действия выполняет процессор при появлении запроса на прерывание?
35. В чем особенности динамического распределения памяти?

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

36. Как организуется страничное и сегментное распределение памяти?
37. Что такое защита памяти и в каких случаях она необходима?
38. Зачем нужен интерфейс?
39. Какие способы управления обменом информацией нашли применение в интерфейсах?
40. Назначение микросхем системной логики.
41. Каков порядок подключения внешних устройств к ПК?
42. Назначение системных вызовов и драйверов в современных операционных системах.
43. Какова роль контроллеров в схемах периферийных устройств?
44. Состав и назначение компонентов программного обеспечения.
45. Основные отличия однопрограммных и многопрограммных режимов работы компьютера.
46. Недостатки классической пакетной обработки.
47. Суть режима кругового циклического обслуживания.
48. Причины появления и развития режима реального времени.
49. По каким признакам классифицируются вычислительные системы?
50. Какие принципы заложены в основу классификации архитектур ВС?
51. Какие виды параллелизма применяются в архитектуре ВС?
52. В чем заключаются отличия многопроцессорных от многомашинных вычислительных комплексов?
53. Преимущества, обеспечиваемые применением компьютерных сетей.
54. Модель взаимодействия открытых систем.
55. Понятие протокола и интерфейса в компьютерных сетях.
56. Назначение и состав протоколов в компьютерных сетях.
57. Виды коммутаций в компьютерных сетях.
58. Методы защиты информации в компьютерных сетях.
59. Отличительные особенности построения локальных компьютерных сетей.
60. Виды сервисов в сети Интернет.
61. Адресация компьютеров в сети Интернет.
62. Принцип построения и работы электронной почты в сети Интернет.
63. Сервис WWW. Поиск информации в сети Интернет.
64. Принцип построения корпоративных компьютерных сетей.
65. Многоядерные структуры микропроцессоров.
66. Принцип построения параллельных вычислений при использовании многоядерных микропроцессоров.
67. Какие факторы влияют на дальнейшее развитие средств высокопроизводительной вычислительной техники и почему?
68. Альтернативные пути развития вычислительной техники.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

# СЛОВАРЬ ТЕРМИНОВ И ОПРЕДЕЛЕНИЙ

**Адаптер** – устройство связи компьютера с периферийными устройствами.

**Адрес** – номер конкретного байта оперативной памяти компьютера.

**Алгоритм** – заранее заданное понятное и точное предписание возможному исполнителю совершить определенную последовательность действий для получения решения задачи за конечное число шагов.

**Алфавит** – фиксированный для данного языка набор основных символов, т. е. букв алфавита, из которых должен состоять любой текст на данном языке.

**Арифметико-логическое устройство (АЛУ)** – часть процессора, которая производит выполнение операций, предусмотренных данным компьютером.

**Архитектура компьютера** – описание компьютера на общем уровне, определяющее принципы действия, информационные связи и взаимное соединение основных логических узлов компьютера.

**Архитектура фон Неймана** – архитектура компьютера, имеющего одно арифметико-логическое устройство, через которое проходит поток данных, и одно устройство управления, через которое проходит поток команд.

**Аудиоадаптер (звуковая плата)** – специальная электронная плата, которая позволяет записывать звук, воспроизводить и создавать его программными средствами с помощью микрофона, наушников, динамиков, встроенного синтезатора и другого оборудования.

**База данных** – один или несколько файлов данных, предназначенных для хранения, изменения и обработки больших объемов взаимосвязанной информации.

**Байт** – группа из 8 бит, рассматриваемая при хранении данных как одно целое.

**Библиотека стандартных программ** – совокупность подпрограмм, составленных на одном из языков программирования и удовлетворяющих единым требования к структуре, организации их входов и выходов, описаниям подпрограмм.

**Бит** – наименьшая единица информации в цифровом компьютере, принимающая значение 0 или 1.

**Ввод** – считывание информации с внешнего устройства в память компьютера.

**Видеоадаптер** – электронная плата, которая обрабатывает видеоданные (текст и графику) и управляет работой дисплея.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

**Вирус компьютерный –** специально написанная небольшая программа, которая может «приписывать» себя к другим программам для выполнения каких-либо вредных действий: портит файлы, «засоряет» оперативную память.

**Внешняя память** – совокупность запоминающих устройств для длительного хранения данных. Информация, размещенная на внешних носителях, не зависит от того, включен или выключен компьютер.

**Второе поколение компьютеров –** с конца 1950-х до середины 1960-х гг. Элементная база – полупроводниковые элементы. По сравнению с ЭВМ предыдущего поколения улучшены все технические характеристики. Для программирования используются алгоритмические языки.

**Вывод** – результаты работы программы, выдаваемые компьютером пользователю, другому компьютеру или во внешнюю память.

**Выражение** – в языке программирования – запись правила для вычисления некоторого значения. Строится из констант, переменных и указателей функций, объединенных знаками операций.

**Гипертекст –** текст, содержащий в себе связи с другими текстами, графической, видео- или звуковой информацией.

**Графический редактор** – программа, позволяющая создавать и редактировать изображения на экране компьютера.

**Диск** – круглая металлическая или пластмассовая пластина, покрытая магнитным материалом, на которую заносится информация в виде концентрических дорожек, разделенных на секторы.

**Дисковод** – устройство, управляющее вращением магнитного диска, чтением и записью данных на нем.

**Дисплей** – устройство визуального отображения информации (в виде текста, таблицы, рисунка, чертежа и т. д.) на экране электронно-лучевого прибора.

**Драйверы** – программы, расширяющие возможности операционной системы по управлению устройствами ввода-вывода, оперативной памятью и т. д. С помощью драйверов возможно подключение к компьютеру новых устройств или нестандартное использование имеющихся устройств.

**Идентификатор** – символическое имя переменной, которое идентифицирует ее в программе.

**Интегральная схема** – реализация электронной схемы, выполняющей некоторую функцию, в виде единого полупроводникового кристалла, в котором изготовлены все компоненты, необходимые для осуществления этой функции.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

**Интегрированные пакеты программ** – пакеты программ, выполняющие ряд функций, для которых ранее создавались специализированные программы, в частности текстовые редакторы, электронные таблицы, системы управления базами данных.

**Интернет** – всемирная компьютерная сеть, объединяющая десятки тысяч сетей всего мира. Ее назначение – обеспечить любому желающему постоянный доступ к любой информации.

**Интерпретатор** – разновидность транслятора. Переводит и выполняет программу с языка высокого уровня в машинный код строка за строкой.

**Интерфейс** – электронная схема сопряжения двух устройств, обменивающихся информацией.

**Информатика –** дисциплина, изучающая структуру и общие свойства информации, а также закономерности и методы ее создания, хранения, поиска, передачи и преобразования с использованием компьютерных технологий.

**Информационная система –** взаимосвязанная совокупность средств, методов и персонала, участвующих в обработке данных.

**Информационная технология –** совокупность методов, производственных и программно-технических средств, объединенных в технологическую цепочку, обеспечивающую сбор, хранение, обработку, вывод и распространение информации для снижения трудоемкости процессов использования информационных ресурсов, повышения надежности и оперативности.

**Информационно-поисковая система (ИПС) –** система, выполняющая функции хранения большого объема информации, быстрого поиска требуемой информации, добавления, удаления и изменения хранимой информации, вывода ее в удобном для человека виде.

**Информационный процесс –** процесс, в результате которого осуществляется прием, передача (обмен), преобразование и использование информации.

**Информация –** сведения об объектах и явлениях окружающей среды, их параметрах, свойствах и состоянии, которые уменьшают имеющуюся о них степень неопределенности, неполноты знаний. Применительно к обработке данных на компьютерах – произвольная последовательность символов, несущих смысловую нагрузку.

**Исполнитель алгоритма –** человек или автомат (в частности, процессор компьютера), умеющий выполнять определенный набор действий. Исполнителя характеризуют среда, элементарные действия, система команд, отказ.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

**Итерационный цикл –** вид цикла, для которого число повторений операторов тела цикла заранее неизвестно. На каждом шаге вычислений происходит последовательное приближение и проверка условий достижения искомого результата. Выход из цикла происходит в случае выполнения заданного условия.

**Клиент** (рабочая станция) **–** любой компьютер, имеющий доступ к услугам сервера. Клиентом также называют прикладную программу, которая от имени пользователя получает услуги сервера.

**Клиент-сервер –** модель взаимодействия пользователя с рабочими станциями, при которой сервер сначала обрабатывает запросы, а затем посылает клиенту то, в чем он нуждается.

**Ключевое слово –** слово языка программирования, имеющее определенный смысл для транслятора. Его нельзя использовать для других целей, например, в качестве имени переменной.

**Команда –** описание элементарной операции, которую должен выполнить компьютер. Обычно содержит код выполняемой операции, указания по определению операндов (или их адресов), указания по размещению получаемого результата. Последовательность команд образует программу.

**Компилятор** – транслятор, выполняющий преобразование программы, составленной на исходном языке, в объектный модуль.

**Контроллер** – устройство, которое связывает периферийное оборудование или каналы связи с центральным процессором, освобождая процессор от непосредственного управления функционированием данного оборудования.

**Курсор –** светящийся символ на экране дисплея, указывающий позицию, на которой будет отображаться следующий вводимый с клавиатуры знак.

**Кэш** (сверхоперативная память) **–** очень быстрое запоминающее устройство малого объема. Используется для компенсации разницы в скорости обработки информации процессором и несколько менее быстродействующей оперативной памятью.

**Логический тип –** тип данных, представляемых значениями «истина» или «ложь» («да» или «нет»).

**Логический элемент** (вентиль) **–** часть электронной схемы, выполняющая элементарную логическую функцию.

**Логическое высказывание –** любое предложение, в отношении которого можно однозначно сказать, истинно оно или нет.

**Локальная сеть –** сеть, которая связывает ряд компьютеров в зоне, ограниченной пределами одной комнаты, здания или предприятия.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

**Манипуляторы** (джойстик, мышь, трекбол и др.) – специальные устройства для управления курсором.

**Маршрутизатор** – программа, определяющая путь передачи сообщения между узлами и абонентами сети.

**Массив** – последовательность однотипных элементов, число которых фиксировано и которым присвоено одно имя. Компьютерный эквивалент таблицы. Положение элемента в массиве однозначно определяется его индексами.

**Математическая модель** – система математических соотношений – формул, уравнений, неравенств и т. д., отражающих существенные свойства объекта.

**Машинный язык** – совокупность машинных команд компьютера, отличающаяся числом адресов в команде, назначением информации, задаваемой в адресах, набором операций, которые может выполнить машина и др.

**Меню** – выведенный на экран дисплея список различных вариантов работы компьютера, по которому можно сделать конкретный выбор.

**Микропроцессор** – процессор, выполненный в виде интегральной схемы. Состоит из цепей управления, регистров, сумматоров, счетчиков команд, очень быстрой памяти малого объема.

**Модем** – устройство, обеспечивающее преобразование цифровых сигналов компьютера в переменный ток частоты звукового диапазона (модуляцию), а также обратное преобразование (демодуляцию). Используется для соединения компьютера с другими компьютерными системами через телефонную сеть.

**Монитор** – см. Дисплей.

**Мост** – устройство для соединения двух сетей, присутствие которого для абонентов невидимо (в отличие от шлюзов).

**Оболочки** – программы, создаваемые для упрощения работы со сложными программными системами, такими как операционная система DOS. Они преобразуют неудобный командный интерфейс в дружественный графический интерфейс или интерфейс типа меню.

**Обработка информации** – в информатике – любое преобразование информации из одного вида в другой, производимое по строгим формальным правилам.

**Общая шина** – топология сети, при которой ее узлы подключены к общему линейному информационному каналу.

**Объект** – совокупность свойств (структур данных, характерных для этого объекта), методов их обработки (подпрограмм изменения свойств) и событий, на которые данный объект может реагировать и которые приводят, как правило, к изменению свойств объекта.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

**Оперативная память –** быстрое запоминающее устройство не очень большого объема, непосредственно связанное с процессором и предназначенное для записи, считывания и хранения выполняемых программ и данных, обрабатываемых этими программами.

**Оператор –** фраза алгоритмического языка, определяющая некоторый законченный этап обработки данных. В состав операторов входят ключевые слова, данные, выражения и др.

**Операционная система –** комплекс взаимосвязанных программ, предназначенных для автоматизации планирования и организации процесса обработки программ, ввода-вывода и управления данными, распределения ресурсов, подготовки и отладки программ, других вспомогательных операций обслуживания.

**Описание –** раздел программы, идентифицирующий структуры данных, которыми должна манипулировать программа, и описывающий их типы.

**Основание системы счисления –** количество различных цифр, используемых для изображения чисел в данной системе счисления.

**Отладка –** этап компьютерного решения задачи, при котором происходит устранение явных ошибок в программе. Часто производится с использованием специальных программных средств – отладчиков.

**Пакеты прикладных программ** (ППП) **–** специальным образом организованные программные комплексы, рассчитанные на общее применение в определенной проблемной области и дополненные соответствующей технической документацией.

**Первое поколение компьютеров –** с середины 1940-х до конца 1950-х гг. Элементная база – электронные лампы. ЭВМ отличаются большими габаритными размерами, потреблением большого количества энергии, малой скоростью действия, низкой надежностью. Программирование ведется в машинных кодах.

**Переменная –** величина, значение которой может меняться в процессе выполнения программы.

**Пиктограмма –** условное изображение информационного объекта или операции.

**Подпрограмма –** самостоятельная часть программы, которая создается независимо от других частей и затем вызывается по имени.

**Поколения компьютеров –** условная, нестрогая классификация вычислительных систем по степени развития аппаратных и программных средств, а также способов общения с ними.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

**Порты устройств –** электронные схемы, содержащие один или несколько регистров ввода-вывода и позволяющие подключать периферийные устройства компьютера к внешним шинам микропроцессора.

**Постоянная память –** энергонезависимое постоянное запоминающее устройство (ПЗУ), изготовленное в виде микросхемы. Используется для хранения данных, не требующих изменения. Содержание памяти специальным образом «зашивается» в ПЗУ при изготовлении. В ПЗУ находится программа управления работой самого процессора, программы управления дисплеем, клавиатурой, принтером, внешней памятью, программы запуска и остановки компьютера, тестирования устройств.

**Прикладная программа –** любая конкретная программа, способствующая решению какой-либо задачи в пределах данной проблемной области.

**Приложение –** совокупность программ, реализующих обработку данных в определенной области применения.

**Принципы открытой архитектуры:**

1. Регламентируются и стандартизируются только описание принципа действия компьютера и его конфигурация (определенная совокупность аппаратных средств и соединений между ними). Таким образом, компьютер можно собирать из отдельных узлов и деталей, разработанных и изготовленных независимыми фирмами-изготовителями.
2. Компьютер легко расширяется и модернизируется за счет наличия внутренних расширительных гнезд, в которые пользователь может вставлять разнообразные устройства, удовлетворяющие заданному стандарту, и тем самым устанавливать конфигурацию своей машины в соответствии со своими личными предпочтениями.

**Принципы фон Неймана:**

1. **Принцип программного управления**. Программа состоит из набора команд, которые выполняются процессором автоматически друг за другом в определенной последовательности.
2. **Принцип адресности.** Основная память состоит из перенумерованных ячеек; процессору в произвольный момент времени доступна любая ячейка.
3. **Принцип однородности памяти.** Программы и данные хранятся в одной и той же памяти. Поэтому компьютер не различает, что хранится в данной ячейке памяти — число, текст или команда. Над командами можно выполнять такие же действия, как и над данными.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

**Приоритет** – число, приписанное задаче, процессу или операции и определяющее очередность их выполнения или обслуживания.

**Программирование –** написание текста на алгоритмическом языке. Целью программирования является описание процессов обработки данных.

**Программное обеспечение** – совокупность программ, выполняемых компьютером, а также вся область деятельности по проектированию и разработке программ.

**Протокол коммуникации** – согласованный набор конкретных правил обмена информацией между различными устройствами передачи данных.

**Псевдокод** – система обозначений и правил, предназначенная для единообразной записи алгоритмов. Занимает промежуточное место между естественным и формальными языками.

**Разделение времени –** режим работы, при котором время процессора предоставляется различным приложениям последовательно квантами. По истечении кванта времени приложение возвращается в очередь ожидания обслуживания.

**Растр –** множество точечных элементов, с помощью которых знак фиксируется на бумажном носителе или отображается на экране дисплея.

**Реальное время –** режим обработки данных, при котором обеспечивается взаимодействие вычислительной системы с внешними по отношению к ней процессами в темпе, соизмеримом со временем протекания этих процессов.

**Регистр –** специальная запоминающая ячейка, выполняющая функции кратковременного хранения числа или команды и выполнения над ними некоторых операций. Отличается от ячейки памяти тем, что может не только хранить двоичный код, но и преобразовывать его.

**Регистр команд –** регистр устройства управления для хранения кода команды на период времени, необходимый для ее выполнения.

**Свойство** – характеристика объекта, его параметр. Все объекты наделены определенными свойствами, которые в совокупности выделяют объект из множества других объектов.

**Сервер –** высокопроизводительный компьютер с большим объемом внешней памяти, который обеспечивает обслуживание других компьютеров путем управления распределением дорогостоящих ресурсов совместного пользования (программ, данных и периферийного оборудования).

**Сетевая операционная система –** система, осуществляющая реализацию протоколов с реализацией управления серверами.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

**Сеть компьютерная –** совокупность компьютеров, соединенных с помощью каналов связи и средств коммуникации в единую систему для обмена сообщениями и доступа пользователей к программным, техническим, информационным и организационным ресурсам сети.

**Синтаксис –** набор правил построения фраз языка, позволяющий определить, какие комбинации символов являются осмысленными предложениями в этом языке.

**Система** – любой объект, который одновременно рассматривается и как единое целое, и как совокупность разнородных объектов, объединенных для достижения определенного результата.

**Система команд** – совокупность операций, выполняемых компьютером.

**Система программирования** – система для разработки новых программ на конкретном языке программирования. Предоставляет пользователю мощные и удобные средства разработки программ: транслятор, редактор текстов программ, библиотеки стандартных программ, отладчик и др.

**Система счисления –** математическая модель или знаковая система, в которой числа записываются по определенным правилам с помощью цифр.

**Система управления базами данных** (СУБД) – система программного обеспечения, позволяющая обрабатывать обращения к базе данных, поступающие от прикладных программ конечных пользователей.

**Системные программы** – программы общего пользования, выполняемые вместе с прикладными и служащие для управления ресурсами компьютера: центральным процессором, памятью, вводом-выводом.

**Событие** – изменение состояния объекта.

**Сортировка** – распределение упорядоченных данных по возрастанию или убыванию значений признака сортировки: упорядочение записей файла по одному или нескольким ключам.

**Структурное программирование** – метод разработки программ, требующий разбиения программы на небольшие независимые части (модули). Обеспечивает возможность проведения доказательства правильности программ, повышает уверенность в правильности конечной программы.

**Сумматор** – электронная логическая схема, выполняющая суммирование двоичных чисел.

**Схема алгоритма (блок-схема)** – графическое представление алгоритма в виде последовательности блоков, соединенных стрелками.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

**Счетчик команд** – регистр устройства управления, содержимое которого соответствует адресу очередной выполняемой команды. Служит для автоматической выборки команд программы из последовательных ячеек памяти.

**Таблица истинности** – табличное представление логической схемы (операции), в котором перечислены все возможные сочетания значений истинности входных сигналов (операндов) вместе со значением истинности выходного сигнала (результата операции) для каждого из этих сочетаний.

**Текстовый редактор** – программа для ввода и изменения текстовых данных (документов, книг, программ и т.д.). Обеспечивает редактирование строк текста, контекстный поиск и замену частей текста, автоматическую нумерацию страниц, обработку и нумерацию сносок, проверку правописания, построение оглавлений, распечатку текста на принтере и др.

**Тест** – некоторая совокупность данных для программы, а также точное описание всех результатов, которые должны получиться в результате работы программы.

**Тестирование** – этап решения задачи на компьютере, в процессе которого проверяется работоспособность программы, не содержащей явных ошибок.

**Тип данных** – понятие языка программирования, определяющее структуру констант, переменных и других элементов данных, а также разрешенные их значения и операции, которые можно над ними выполнять.

**Топология компьютерной сети** – логический и физический способ соединения компьютеров, кабелей и других компонентов, в целом составляющих сеть. Топология характеризует свойства сетей, не зависящие от их размеров. Наиболее распространенные виды топологий: линейная, кольцевая, древовидная, звездообразная, полносвязная.

**Транслятор** – программа-переводчик. Преобразует программу, написанную на одном из языков высокого уровня, в программу, состоящую из машинных команд.

**Трафик** – поток сообщений в сети передачи данных.

**Третье поколение компьютеров –** с середины 1960-х до конца 1970-х гг. Элементная база – интегральные схемы. Значительное уменьшение габаритных размеров ЭВМ, повышение их надежности, увеличение производительности. Доступ с удаленных терминалов.

**Триггер** – электронная схема, широко применяемая в регистрах компьютера для надежного запоминания одного бита информации. Имеет два устойчивых состояния, которые соответствуют двоичной единице и двоичному нулю.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

**Упаковщики (архиваторы)** – программы, позволяющие записывать информацию на дисках более плотно, а также объединять копии нескольких файлов в один архивный файл.

**Устройство управления (УУ)** – часть процессора, выполняющая функцию управления устройствами компьютера.

**Файл** – именованная совокупность любых данных, размещенная на внешнем запоминающем устройстве и хранимая, пересылаемая и обрабатываемая как единое целое. Файл может содержать программу, числовые данные, текст, закодированное изображение и т.д. Имя файла регистрируется в каталоге.

**Файл-сервер** – содержит базу данных и программы управления данными для обеспечения многопользовательских запросов.

**Цикл** – прием в программировании, позволяющий многократно повторять одну и ту же последовательность команд (операторов).

**Четвертое поколение компьютеров –** поколение машин, разработанных после 1970 г. Элементная база – микропроцессоры, большие интегральные схемы. Емкость ОЗУ – десятки мегабайт. Машины этого поколения представляют собой персональные компьютеры либо многопроцессорные и (или) многомашинные комплексы. Быстродействие – до нескольких сотен миллионов операций в секунду.

**Шлюз** – устройство соединения сети с другими сетями, например локальной сети с глобальной сетью.

**Электронная почта** – система пересылки сообщений между пользователями вычислительных систем, в которой на компьютер возлагаются все функции по хранению и пересылке сообщений.

**Электронная таблица** – работающая в диалоговом режиме программа обработки числовых данных, хранящая и обрабатывающая данные в прямоугольных таблицах.

**Язык ассемблера** – машинно-ориентированный язык программирования, в котором команды компьютера записывается с использованием символических адресов и мнемокодов. На ассемблере можно писать программы более эффективные, чем программы, полученные в результате трансляции с языка высокого уровня.

**Язык высокого уровня** – язык программирования, более близкий к естественному языку, чем машинный код или язык ассемблера. Каждый оператор в нем соответствует нескольким командам машинного языка или ассемблера.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

**ASCII** – американский стандартный код обмена информацией. Широко используется для кодирования в виде байта букв, цифр, знаков операций и других компьютерных символов.

**Host-ЭВМ** – ЭВМ, установленная в узлах сети и решающая вопросы коммутации сети, доступа к сетевым ресурсам: модемам, факс-модемам, большим ЭВМ и др., главная, ведущая, центральная ЭВМ.

**HTML** (HyperText Markup Language) – язык разметки гипертекста. Используется для создания документов, предназначенных для размещения в системе WWW.

**HTTP** (HyperText Transfer Protocol) – протокол передачи гипертекста, служащий для установления связи с документами HTML.

**Hub** – концентратор, хаб. Устройство, обеспечивающее преобразование передаваемых сигналов таким образом, что сеть может быть дополнена новыми рабочими станциями.

**OLE** (Object Linked and Embedding) – технология, позволяющая вставлять ранее созданные объекты (рисунки, графики, таблицы) во вновь создаваемые.

**SMPT** (Simple Management Transfer Protocol) – упрощенный протокол передачи почтовых сообщений системами электронной почты.

**SQL** (Structured Query Language) – язык программирования, который используется при работе с реляционными базами данных в современных СУБД.

**TCP/IP** (Transmission Control Protocol / Internet Protocol) – протокол коммуникации в Интернет. В современном понимании TCP/IP – это не один, а два протокола, лежащих на разных уровнях. Протокол TCP – протокол транспортного уровня. Он управляет тем, как происходит передача информации. Протокол IP – адресный. Он принадлежит сетевому уровню и определяет, куда происходит передача.

**URL** (Uniform Resource Locator) – универсальный локатор ресурсов, характеризующий полный путь к конкретному документу или разделу на компьютере, подключенном к сети Интернет, а также метод доступа к нему, т.е. протокол работы с программами-серверами, функционирующими на удаленном компьютере.

**Web page** – Web-страница в системе WWW.

**WWW (Word Wide Web)** – глобальная сетевая система, объединяющая серверы Интернет, Всемирная паутина.

НОО «Профессиональная наука» использует Creative Commons Attribution (CC BY 4.0): лицензию на опубликованные материалы - https://creativecommons.org/licenses/by/4.0/deed.ru|

# БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Бройдо В.Л. Архитектура ЭВМ и систем: учебник. Гриф МО РФ / В.Л. Бройдо, О.П. Ильина. – 2-е изд. – СПб.: Питер, 2009.

2. Гапоненко В.Ф., Пронькин Н.Н. и др. Проектирование информационных систем в управлении. Учебно-методический комплекс. – Московский городской университет управления Правительства Москвы. Москва, 2011.

3. Глущенко В.М., Гапоненко В.Ф., Елизаров В.С., Пронькин Н.Н. Экоинформатика: основы и перспективы развития. Учебно-методический комплекс для направления 020800.62 "Экология и природопользование". – Московский городской университет управления Правительства Москвы. Москва, 2010.

4. Глущенко В.М., Елизаров В.С., Каманин И.О., Пронькин Н.Н. Информатика. Учебно-методический комплекс для направления 230200.62 "Информационные системы". – Московский городской университет управления Правительства Москвы. Москва, 2009.

5. Глущенко В.М., Пронькин Н.Н. и др. Информационные системы и технологии. Учебник. – Московский городской университет управления Правительства Москвы. Москва, 2012.

6. Елизаров В.С., Ковалева Е.Д., Пронькин Н.Н. Математика и информатика. Учебно-методический комплекс. – Московский городской университет управления Правительства Москвы. Москва, 2009.

7. Кузин А.В. Архитектура ЭВМ и вычислительных систем: учебник для студ. учреждений среднего проф. образования. Гриф МО РФ / А.В. Кузин, С.А. Пескова. – Москва: ФОРУМ: ИНФРА-М, 2006.

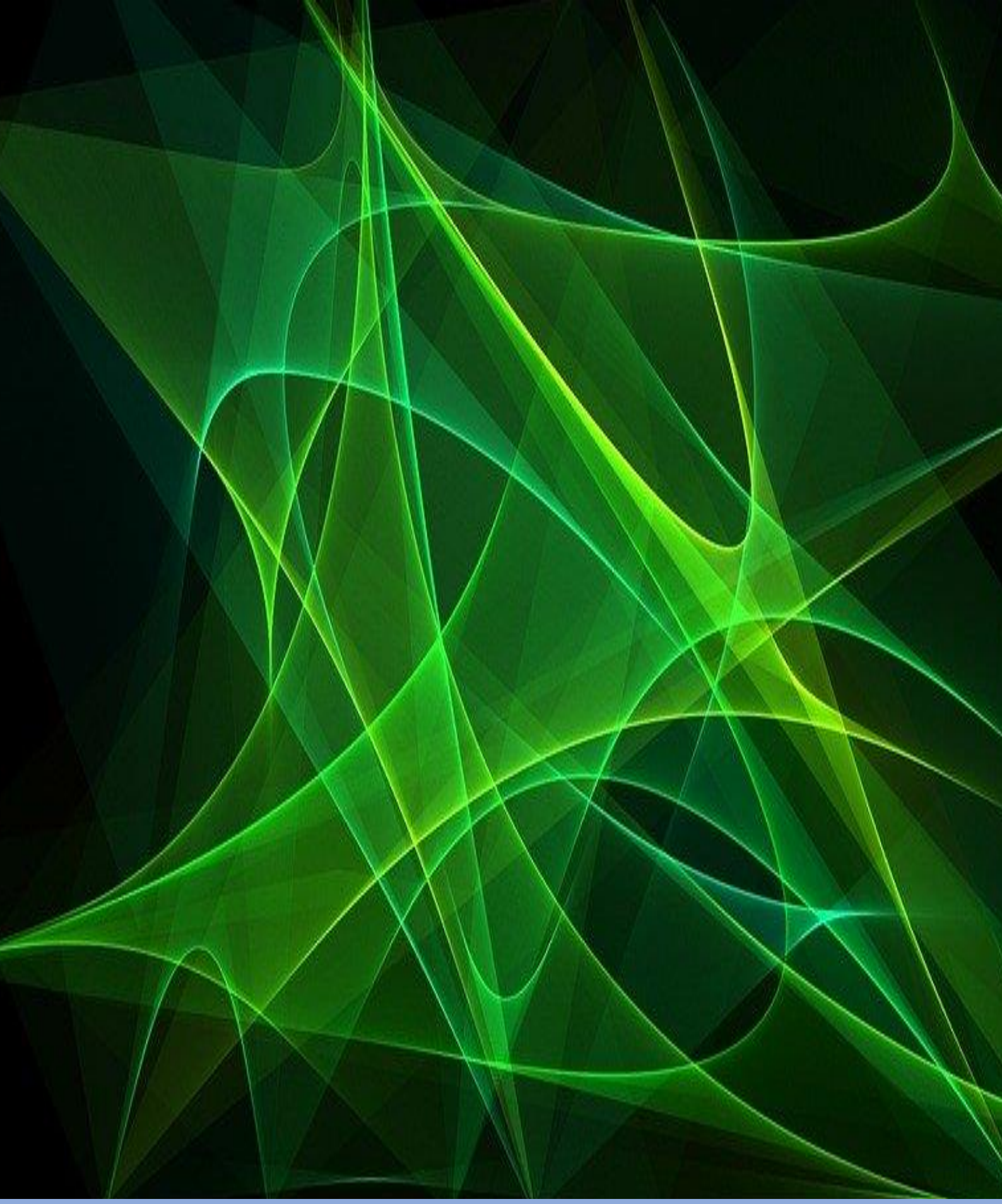
8. Максимов Н.В. Архитектура ЭВМ и вычислительных систем: учебник. Гриф МО РФ / Н.В. Максимов, Т.Л. Партыка, И.И. Попов. – 2-е изд., перераб. и доп. – Москва: ФОРУМ, 2008.

9. Пронькин Н.Н. Архитектура ЭВМ и систем. Учебно-методический комплекс для направления 230200.62 "Информационные системы". – Московский городской университет управления Правительства Москвы. Москва, 2011.

10. Пронькин Н.Н. Информатика. Учебно-методический комплекс для направления 020800.62 "Экология и природопользование" – Московский городской университет управления Правительства Москвы. Москва, 2011.

11. Пронькин Н.Н. Информатика. Учебно-методический комплекс для специальности 071401.65 "Социально-культурная деятельность". – Московский городской университет управления Правительства Москвы. Москва, 2012.

12. Таненбаум Э. Архитектура компьютера. – 5-е изд. – Санкт-Петербург: Питер, 2007.



ISBN 978-5-907072-89-3

Усл. печ. л. 3,3.
Объем издания 1,8 MB
Оформление электронного издания:
НОО Профессиональная наука, mail@scipro.ru
Дата размещения: 15.11.2019 г.
URL: http://scipro.ru/conf/computerarchitecture.pdf