

**БЕЛГОРОДСКИЙ ЮРИДИЧЕСКИЙ ИНСТИТУТ МВД РОССИИ
ИМЕНИ И.Д. ПУТИЛИНА**

**ОСНОВЫ ИНФОРМАТИКИ
И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ
В ОВД**

УЧЕБНОЕ ПОСОБИЕ

**Белгород
Белгородский юридический институт МВД России
имени И.Д. Путилина
2016**

УДК 004: 351.749
ББК 32.97
О 75

Печатается по решению
редакционно-издательского совета
Бел ЮИ МВД России
имени И.Д. Путилина

Авторы: **Прокопенко А.Н.**, начальник кафедры информационно-компьютерных технологий в деятельности органов внутренних дел Бел ЮИ МВД России имени И.Д. Путилина, кандидат технических наук, доцент; **Жукова П.Н.**, профессор кафедры информационно-компьютерных технологий в деятельности органов внутренних дел Бел ЮИ МВД России имени И.Д. Путилина, доктор физико-математических наук, доцент; **Дрога А.А.**, заместитель начальника кафедры информационно-компьютерных технологий в деятельности органов внутренних дел Бел ЮИ МВД России имени И.Д. Путилина; **Страхов А.А.**, заместитель начальника кафедры информатики и математики Московского университета МВД России имени В.Я. Кикотя.

О 75 **Основы информатики и информационных технологий в ОВД:** учебное пособие / А.Н. Прокопенко и [др.]. – Белгород: Бел ЮИ МВД России имени И.Д. Путилина, 2016. - 144 с.

ISBN 978-5-91776-130-5

Учебное пособие содержит теоретический и практический материал основных положений информатики и информационных технологий, используемых в деятельности правоохранительных органов. Рассматриваются информация и её свойства, роль информации в жизни общества, понятие информации и информационной системы, качества, количество и свойства информации применительно к профессиональной деятельности.

Предназначено для курсантов и слушателей образовательных организаций системы МВД России.

УДК 004: 351.749
ББК 32.97

ISBN 978-5-91776-130-5

© РИО Белгородского юридического института МВД России
имени И.Д. Путилина, 2016

СОДЕРЖАНИЕ

Введение	5
1. Информация и информационные процессы	6
1.1. Понятие информации	6
1.2. Понятие информационной системы, сигнала, помехи	8
1.3. Характеристики и свойства информации	14
1.4. Информатика и информационные технологии	16
<i>Вопросы для самоконтроля</i>	17
2. Кодирование и представление информации в ЭВМ	18
2.1. Измерение количества информации	18
2.2. Электронная вычислительная машина, машинный код	22
2.3. Кодирование графических изображений	28
2.4. Кодирование звука и видео	33
<i>Вопросы для самоконтроля</i>	36
3. Системы счисления	37
<i>Вопросы для самоконтроля</i>	44
4. Основы алгебры логики	45
4.1. Понятие алгебры логики	45
4.2. Типы операций в алгебре логики	48
4.3. Диаграммы Эйлера для теоретико-множественных операций	53
<i>Вопросы для самоконтроля</i>	56
5. Эволюция электронных вычислительных машин	57
5.1. История машинных вычислений, компьютерная архитектура	57
5.2. Первое поколение ЭВМ – на электронных лампах (1945-1955)	59
5.3. Второе поколение ЭВМ – на транзисторах (1955-1965)	62
5.4. Третье поколение ЭВМ – на интегральных схемах (1965-1980)	64
5.5. Четвертое поколение ЭВМ – на сверхбольших интегральных схемах (с 1980)	65
<i>Вопросы для самоконтроля</i>	67
6. Конфигурация и характеристики основных устройств современного персонального компьютера	68
6.1. Типы компьютеров и их сравнительные характеристики	68

6.2. Конфигурация современного персонального компьютера	72
6.3. Периферийные устройства персонального компьютера	85
Вопросы для самоконтроля	88
7. Создание программ на компьютере	89
7.1. Порядок и правила решения задач с использованием компьютера ...	89
7.2. Алгоритмы и программы	93
7.3. Программирование на компьютере	99
Вопросы для самоконтроля	105
8. Программное обеспечение современных компьютеров	106
8.1. Основные понятия, назначение и классификация программного обеспечения	106
8.2. Системное программное обеспечение и операционные системы	110
8.3. Файловые системы	114
Вопросы для самоконтроля	119
9. Компьютерные сети	120
9.1. Основные понятия компьютерных сетей	120
9.2. Топологии компьютерных сетей	124
9.3. Функционирование компьютерных сетей – сетевая архитектура и сетевые протоколы	129
9.4. Интернет: история создания и особенности функционирования	132
Вопросы для самоконтроля	140
Список источников	141

ВВЕДЕНИЕ

В настоящее время информатика и ее практические результаты становятся важнейшим двигателем научно-технического прогресса и развития человеческого общества. Скорость развития средства обработки и передачи информации поразительна, в истории человечества этому бурно развивающемуся процессу нет аналога. Сведения, касающиеся прикладной области, быстро устаревают. На смену одним технологиям приходят другие, более совершенные и более сложные. Специалисты в области информационных технологий должны непрерывно обучаться и повышать свою квалификацию. Однако иметь теоретические и практические знания в области информатики в наше время стало необходимо для всех, потому что общество, в котором мы живем, является информационным.

Оперативно-служебная деятельность правоохранительных органов неразрывно связана с современными информационными системами, и, следовательно, от уровня информационной поддержки профессиональной деятельности напрямую зависит ее эффективность и в конечном итоге качество результатов. Огромный объем аккумулируемой органами внутренних дел информации (открытой, конфиденциальной, служебной, персональных данных, государственной тайны) ставит эти органы в положение одной из наиболее информационно емких государственных структур.

Используемые в ОВД информационные ресурсы побуждают предъявлять повышенные требования к сотрудникам, эксплуатирующим имеющиеся системы в повседневной деятельности. Таким образом, важнейшую роль в подготовке квалифицированных специалистов играют образовательные организации МВД России, производящие обучение, переподготовку и повышение квалификации сотрудников всех категорий.

Учебное пособие написано в стиле lecture notes и соответствует расширенному курсу лекций, который авторы читают курсантам и слушателям всех специальностей при изучении дисциплины «Информатика и информационные технологии в профессиональной деятельности органов внутренних дел».

1. ИНФОРМАЦИЯ И ИНФОРМАЦИОННЫЕ ПРОЦЕССЫ

*Информация есть информация,
а не материя и не энергия.*

Норберт Винер

1.1. Понятие информации

В соответствии с «Толковым словарем...» информация – это сведения об окружающем мире и протекающих в нем процессах или сообщения, осведомляющие о положении дел, которые воспринимаются человеком или специальными устройствами¹.

Понятие «информация» в современном виде возникло в XX веке. Изначально термин «информация» происходит от латинского слова «informatio», что означает разъяснение, осведомление, изложение. Впоследствии понятие «информация» было распространено на все нематериальные виды объектов, не являющиеся энергией.

В течение веков в христианстве и в других религиях применялось понятие «слово». Библия ясно говорит, что основой всего являлась чистая информация, называя ее «словом»: «В начале сотворил Бог небо и землю. Земля же была безвидна и пуста, и тьма над бездною; и Дух Божий носился над водою. И сказал Бог: да будет свет. И стал свет. И увидел Бог свет, что он хорош; и отделил Бог свет от тьмы. И назвал Бог свет днем, а тьму ночью. И был вечер, и было утро: день один» (Быт. 1:1-5).

Понятие «слово» и имело тот смысл, который придается сегодня понятию «информация». Впоследствии вместо «слова» стали использовать понятия «знания», «сведения». С появлением технических устройств возникли понятия «данные», «сообщения» и др. Затем объединяющим понятием для всех видов нематериальных объектов стало понятие «информация».

Многие исследователи и философы неоднократно пытались сформулировать понятие информации, однако не достигли серьезных успехов. Одним из примеров может являться определение информации, данное основоположником кибернетики как науки Норбертом Винером², приведенное в эпиграфе к главе. Причем определение, сформулированное им, можно считать скорее философским, чем техническим или правовым, поэтому использовать в практической деятельности невозможно.

По мере развития технических средств обработки информации, связи и телевидения информацию начали измерять количественно. Способ определения количества передаваемой информации был предложен в 1949 году К. Шенноном и У. Уивером в статье «Математическая теория связи». Одновременно

¹ Ожегов С.И. Толковый словарь русского языка / С.И. Ожегов, Н.Ю. Шведова. – М.: Азъ, 1993. С. 205.

² Винер Н. Кибернетика, или управление и связь в животном и машине. – 2-е изд. – М.: Наука; Главная редакция изданий для зарубежных стран, 1983. – 344 с.

Н. Винер предложил понимать под информацией не просто сведения, а только сведения новые и полезные для принятия решения, обеспечивающего достижение цели управления. Остальные сведения учеными-кибернетиками информацией не признавались. Они получили название «данные» – признаки или записанные наблюдения, которые по каким-то причинам не используются, а только хранятся. Считалось, что при возникновении потребности в данных для уменьшения неопределенности в чем-либо они превращаются в информацию.

Однако современный подход к понятию «информация» не делает различия между видами информации по ее новизне, виду, объему, способу передачи и хранения и т.д. Все виды «слова» получили общее название «информация», которое объединило в своем составе и сообщения, и данные, и сведения, и знания.

Мы, определяя понятие «информация» в рамках данного учебного пособия, будем опираться на законодательные и нормативные правовые акты.

Впервые в нашей стране понятие «информация» было закреплено в 1995 году Федеральным законом от 20.02.1995 № 24-ФЗ «Об информации, информатизации и защите информации»¹. В статье 2 указанного Федерального закона было установлено следующее определение:

Информация – это сведения о лицах, предметах, фактах, событиях, явлениях и процессах независимо от формы их представления.

Принятый в 2006 году Федеральный закон от 27.07.2006 № 149-ФЗ «Об информации, информационных технологиях и о защите информации»² изменил определение информации. Новое определение было закреплено в статье 2 указанного Федерального закона:

информация – это сведения (сообщения, данные) независимо от формы их представления.

Таким образом, новое определение информации нивелировало различия между сведениями, сообщениями и данными, причем без расшифровки их содержания. Это позволяет говорить об унификации подхода к определению понятия информация.

Также поменялся подход к понятию «данные», которые сейчас трактуются как вид информации:

данные – информация, представленная в виде, пригодном для обработки автоматическими средствами при возможном участии человека³.

Причем в отличие от понятия информации, понятие данных с 1990 года не претерпело изменений.

¹ Об информации, информатизации и защите информации: Федеральный закон РФ от 20.02.1995 № 24-ФЗ (утратил силу) // Собрание законодательства РФ. 1995. № 8. Ст. 609.

² Об информации, информационных технологиях и о защите информации: Федеральный закон от 27.07.2006 № 149-ФЗ // Собрание законодательства РФ. 2006. № 31 (ч. 1). Ст. 3448.

³ ГОСТ 15971–90. Системы обработки информации. Термины и определения: утв. Постановлением Госстандарта СССР от 26.10.1990 № 2698 // Режим доступа: СПС «Консультант-Плюс».

1.2. Понятие информационной системы, сигнала, помехи

Принципиальным отличием информации от других философских категорий – материи и энергии – является то, что ее можно передавать без потери в месте передачи. Материя не может быть подвержена передаче – ее можно только переместить. Энергия может быть передана, однако при этом она исчезнет в месте передачи и появится в месте приема.

Информация же может быть передана неограниченное число раз без потери в месте передачи. Фактически осуществляется не передача информации, а передача ее копии. Именно в этом заключается принципиальное отличие информации от энергии. На этом отличии информации построено все наше современное общество, в основе которого лежит постоянный обмен информацией. Вообще, любое общество становится возможным только тогда, когда начинается обмен информацией – появляется «слово».

Для передачи информации могут применяться различные способы – радиопередатчик, спутник, телевизионная башня, вышка мобильной связи и другие устройства. Часть устройств не только передает, но и одновременно принимает информацию, например, компьютер или телефон. Кроме того, передача информации осуществляется не только техническими устройствами, но и в живой природе. Растения и животные имеют и меняют цвет, издают звуки, распространяют запахи. Все это – примеры распространения информации. Кроме того, каждый живой организм имеет нервную систему, по которой к разным частям этого организма идут команды. Если кровеносную систему можно сравнить с передачей энергии, то нервная система – уже чистая передача информации. Любое живое существо – это очень сложная информационная система. Вселенная буквально заполнена информацией. Без информации невозможна жизнь ни в нашем мире, ни в каком-либо ином.

Однако для полноценного обмена информацией и ее использования одного передатчика недостаточно. Необходим приемник, который сможет не только принять информацию, но и расшифровать ее, и позволить осуществить ее дальнейшее использование. При одновременном использовании передатчика и приемника возникает информационная система.

Рассмотрим упрощенную схему информационной системы. В качестве примера можно использовать общение между двумя людьми. Например, Вы получили на семинаре хорошую оценку и решили позвонить маме. Сначала Ваш речевой аппарат вызывает колебания мембраны телефонной трубки, которые преобразуются в электрические колебания. При этом вы являетесь *отправителем информации*. В телефоне осуществляется *обработка информации*, кроме того, он является еще и *передатчиком информации*, преобразуя электрические сигналы в радиоволны. Далее информация проходит через *реальный мир* и подвергается воздействию *помех*, которые могли помешать прохождению сигнала мобильной связи. Такой помехой может быть другой радиосигнал, удаленность вышки, помехи от промышленности и т.д. В телефоне мамы радиоволны преобразуются обратно в электрические сигналы, а затем в звуковые колебания. Таким образом, телефон одновременно является *приемником и обра-*

ботчиком информации. При этом мама является *получателем информации.* После этого происходит обработка информации мозгом и ее окончательное восприятие. Вместо телефона в этом примере может быть компьютер или другие устройства.

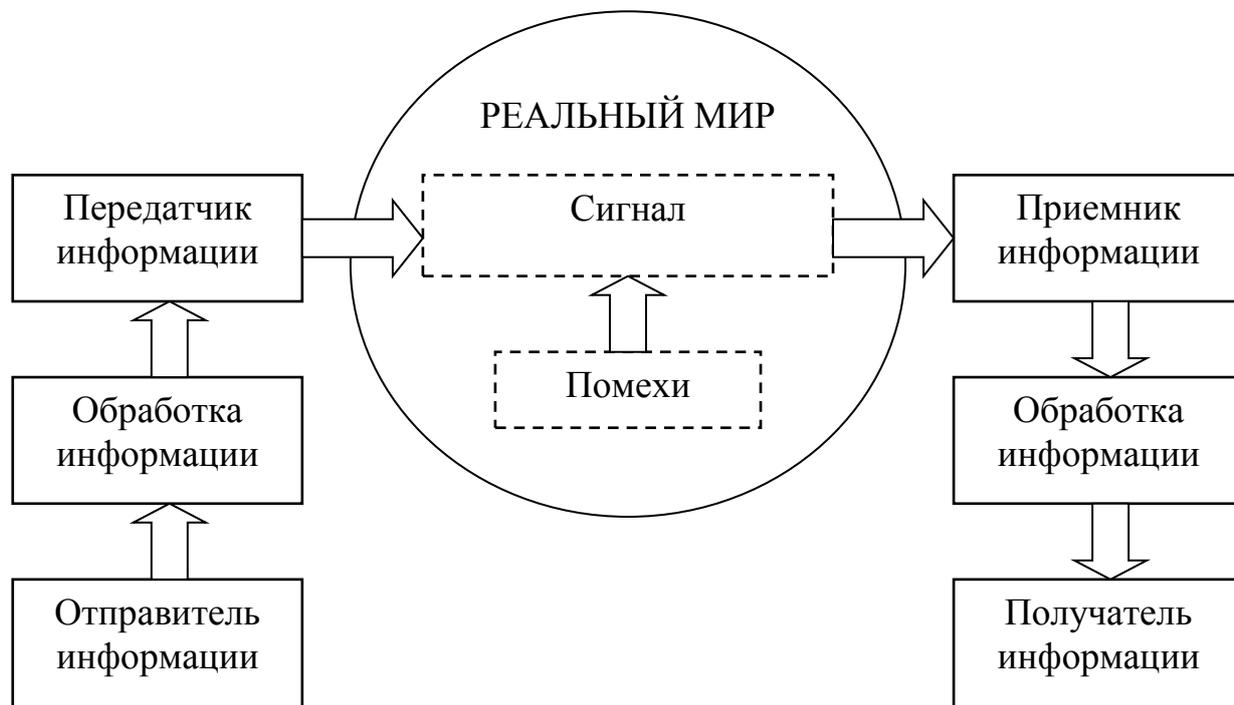


Рис. 1. Общая структурная схема информационной системы

По способу восприятия органами чувств человека информация может быть визуальная, акустическая, тактильная, обонятельная, вкусовая.

Естественное желание человека не ограничиваться возможностями органов чувств приводит к появлению новых методов и средств обнаружения, преобразования, обработки, передачи и хранения информации. Информацию можно получить из электрических сигналов, магнитных полей, ультра- или инфразвуковых колебаний, электромагнитных волн и т.д.

Любая передача информации возможна только при решении вопросов совместимости на двух уровнях.

Первый уровень совместимости – технический.

Сигнал – это совокупность информации и носителя. Носители информации могут иметь самую различную природу. Это могут быть электрические сигналы, радиоволны, звуки, запахи, оптические изображения и т.д. Соответственно, устройства должны «разговаривать» на одной волне. Все наши органы чувств воспринимают информацию на различных носителях. Вкусно-невкусно – это тоже информация.

Сигнал характеризуется мощностью и объёмом информации, которую он несёт. Мощность определяется передатчиком информации. Когда вы разговариваете с рядом стоящим человеком, вам не нужно повышать голос. Мощность одна. Если вы выступаете на митинге без усилителя с микрофоном, придётся кричать, чтобы вас услышали, то есть увеличивать мощность сигнала.

Таким образом, **сигнал** – это физическое явление, наличие, отсутствие или изменение которого представляет информацию.

Приёмник информации характеризуется чувствительностью, то есть той минимальной мощностью сигнала, при которой он может принимать информацию, не теряя её. Чем меньшую мощность сигнала может принимать приёмник, тем выше его чувствительность. Один герой сказки, приложив ухо к земле, слушал, как растёт трава. Очень высокая чувствительность. А иному приходится кричать в ухо. Чувствительность низкая. Но мощность сигнала – это не только электрическая или механическая мощность. Мощность в теории информации понятие более широкое. Например, в случае с запахом мощность – это концентрация молекул вещества в воздухе. В этом случае одним из самых чувствительных природных приёмников является нос собаки.

Сигнал передатчика должен быть достаточно мощным для того, чтобы его принял приемник. Например, настраивая радиоприемник в автомобиле, Вы подстраиваете приемник под сигнал передатчика. Аналогичная ситуация возникает при настройке телевизора и т.д.

В зависимости от характера передачи сигнал может быть дискретным или непрерывным.

В случае когда передатчик информации вырабатывает непрерывное сообщение (соответственно, параметр сигнала – непрерывная функция от времени), соответствующая информация называется непрерывной.

Примером непрерывного сообщения может служить музыка, передаваемая модулированной звуковой волной; параметром сигнала в этом случае является давление, создаваемое этой волной в точке нахождения приемника - человеческого уха. Непрерывный сигнал в технике называют аналоговым сигналом.

Аналоговый сигнал – сигнал данных, у которого каждый из представляющих параметров описывается функцией времени и непрерывным множеством возможных значений.

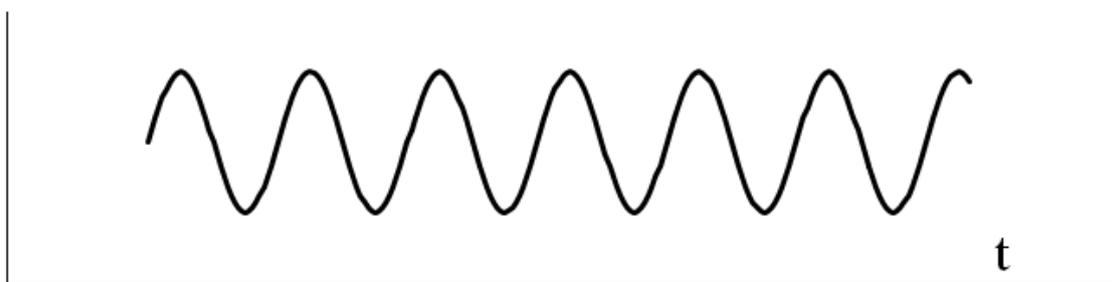


Рис. 2. Пример аналогового сигнала

Если же параметр сигнала принимает последовательное во времени конечное число значений (при этом все они могут быть пронумерованы), сигнал называется дискретным, а сообщение, передаваемое с помощью таких сигналов, – дискретным сообщением. Информация, передаваемая источником, в этом случае также называется дискретной. Пример дискретного сообщения – процесс чтения книги, информация в которой представлена текстом, т.е. дискрет-

ной последовательностью отдельных значков (букв). Дискретный сигнал в технике называют цифровым сигналом.

Цифровой сигнал – это сигнал, у которого каждый из представляющих параметров описывается функцией дискретного времени и конечным множеством возможных значений.

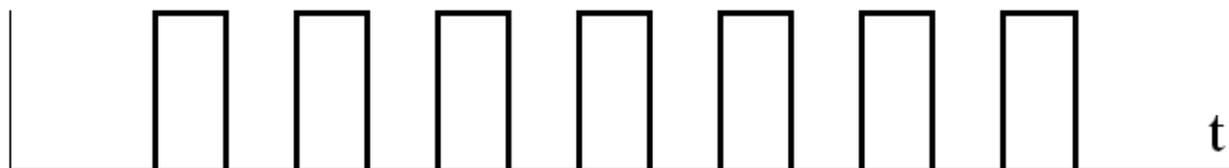


Рис. 3. Пример цифрового сигнала

Любое сообщение может быть представлено как дискретное, иначе говоря, последовательностью знаков некоторого алфавита. Возможность дискретизации непрерывного сигнала с любой желаемой точностью (для возрастания точности достаточно уменьшить шаг) принципиально важна с точки зрения информатики. Компьютер – цифровая машина, то есть внутреннее представление информации в нем дискретно. Дискретизация входной информации (если она непрерывна) позволяет сделать ее пригодной для компьютерной обработки.

Во время приема-передачи или автоматизированной обработки информации возможны преобразования информационных сигналов с изменением амплитуды, формы или временных характеристик.

Аналого-цифровое преобразование сигнала – преобразование информационного сигнала, при котором при заданном шаге дискретизации функция непрерывного множества возможных значений сигнала данных заменяется функцией конечного множества соответствующих значений этого сигнала.

Цифро-аналоговое преобразование сигнала – преобразование информационного сигнала, при котором функция конечного множества возможных значений сигнала данных заменяется функцией непрерывного множества соответствующих значений этого сигнала.

При аналого-цифровом преобразовании речь в микрофон мобильного телефона преобразуется в цифровой сигнал, состоящий из нулей и единиц. А при цифро-аналоговом преобразовании сигнал из мобильной сети преобразуется в звуки человеческой речи, транслируемые динамиком мобильного телефона. Аналогичные преобразования осуществляются при записи видео и его воспроизводстве на аналоговых устройствах.

Отдельно необходимо обратить внимание на проблемы помех. Например, если вы выступаете на митинге, то помехами будут шумы в городе: звуки автомобилей, музыка, ветер и др. Сигнал, который принимается радиоприемником, может быть искажен другим радиосигналом. Солнечный свет мешает передаче сигнала с проектора в лекционной аудитории на экран. Таким образом, можно определить, что *помеха – это сигнал, не несущий полезной информации, но мешающий передаче информации.*

Необходимо отметить, что в другом случае ветер или музыка будут сигналом, а речь на митинге будет помехой. В теории информации для оценки соотношения мощности сигнала к мощности помехи используется понятие помехоустойчивость. *Помехоустойчивость определяет, возможна ли вообще передача сигнала*, поскольку если мощность помехи равна или больше, чем мощность сигнала, то передача информации становится невозможной. Даже если сигнал сильнее помехи, его мощности может быть недостаточно для того, чтобы приемник смог его распознать и воспроизвести.

Для устранения влияния помех используют различные виды усиления сигнала, его кодирование или минимизируют мощность самих помех. Например, большие телескопы устанавливаются в горах или выносятся в космос, как, например, Хаббл, чтобы максимально уменьшить помехи, создаваемые атмосферой, при наблюдениях или, иными словами, при получении информации от небесных объектов.

Помехи бывают естественные и искусственные. Естественных помех великое множество, их примеры приведены выше. Искусственные помехи делаются для того, чтобы подавить каналы передачи информации, забив их помехами. Искусственные помехи активно применяются вооруженными силами и спецслужбами. Например, специальные системы радиоэлектронной борьбы забивают помехами информационные каналы и сети каналов противника, ослепляют радиолокационные станции противовоздушной обороны, забивая их экраны помехами, разрушают сети управления войсками. Подобные системы могут вывести из строя сложную электронику, управляющую кораблем или самолетом.

Второй уровень совместимости – лингвистический.

Информация отправителя должна быть понятна получателю. Проблема понятности информации может возникать как при передаче информации между людьми, так и при передаче информации между механизмами, устройствами, компьютерами и т.д.

Человек имеет определенный объем представлений и слов, которыми он может выразить свои мысли и впечатления. Эту совокупность понятий, представлений и слов, их выражающих, называют *тезаурусом*. В теории информации также используется понятие языка.

Язык – множество литер (символов), условных знаков и правил, которые используются для передачи информации.

Язык естественный – язык, правила которого основаны на текущем использовании без их точного предварительного описания. Например, русский язык, английский язык и т.д.

Язык искусственный – язык, правила применения которого точно установлены до начала его использования. Например, язык глухонемых.

Представить информационные сигналы на языке пользователя тоже не всегда достаточно для полноценного восприятия информации. Наверняка у каждого была ситуация, когда, слушая разговор специалистов в малоизвестной сфере деятельности, общий смысл понимаешь с трудом или вообще понимаешь не так, хотя каждое слово знакомо. Если общаются два физика или программиста, то вряд ли неспециалисты поймут их разговор. Поэтому иногда вместо определения «тезаурус» используют словосочетание «*понятийный аппарат*».

Оценить качество принятого информационного сообщения, можно только сопоставив его с имеющимися *знаниями* – субъективными образами реального мира, зафиксированными в человеческом сознании в форме понятий и представлений.

Очевидно, что в зависимости от целевого назначения и понятийного аппарата субъекта одна и та же информация может интерпретироваться по-разному. Иными словами информация – это знание, относящееся к фактам, событиям, предметам, процессам и идеям, которое в рамках определенного контекста имеет конкретное значение.

Обязательным условием приёма и использования информации является *сопряжение приёмника с сигналом*. Степень сопряжённости количественно определяется значениями от 0 до 1. Например, перед вами книга на китайском языке, в котором вы ничего не понимаете. Информация есть, но приёмник её воспринять не может. Приёмник не сопряжён, или, говоря другими словами, сопряжённость приёмника с сигналом равна 0. Но может быть сопряжённость и 0,6 и 0,3 и любая, принимающая значение от 0 до 1. Например, перед вами английская газета, а вы учили английский в школе. Копаясь в словаре, вы кое-что поймёте, а кое-что нет. Информация прошла, но не вся. Это неполная сопряжённость. Ключ и замок – это тоже информационно-сопряжённая система, но её сопряжённость может принимать только два значения: подходит – 1, не подходит – 0.

Для сопряжения приёмника с сигналом используются преобразователи информации. Например, глаз человека не видит рентгеновских лучей. Врач-рентгенолог стоит перед экраном из вещества, преобразующего рентгеновское излучение в видимый свет. Есть преобразователи, трансформирующие телевизионный сигнал системы Секам в сигнал системы ПАЛ, и наоборот, поскольку телевизор, работающий в системе ПАЛ, не даст изображения, если радиосигнал организован в системе Секам. Всё это называется устройствами сопряжения. Переводчик с одного языка на другой – это тоже сопрягающее устройство.

Фундаментальным положением теории информации является тот факт, что *энергия сигнала не будет приниматься приёмником, если приёмник не сопряжён с сигналом*.

В процессе обращения информации в информационной системе осуществляется ее обработка. Сам процесс обращения информации в информационной системе называется информационным процессом.

Таким образом, *обработка информации* – это совокупность операций сбора, накопления, ввода, вывода, приема, передачи, записи, хранения, регистрации, уничтожения, преобразования и отображения информации.

Информационный процесс – это процесс создания, сбора, обработки, накопления, хранения, поиска, распространения и использования информации.

1.3. Характеристики и свойства информации

Информация является особым видом ресурса. В отличие от ресурсов, связанных с материальными предметами, информационные ресурсы являются неисчерпаемыми и предполагают существенно иные методы воспроизведения и обновления, чем материальные ресурсы¹.

Информация имеет характеристики и свойства.

Характеристики информации:

1. *Объем* – отражает количественную оценку информации (более подробно – в следующих главах учебного пособия).

2. *Достоверность (точность)*. Информация достоверна, то есть правдива, если она точно воспроизводит реально существующие объекты. Если информация является недостоверной, то принятие на ее основе решений может привести к негативным последствиям. Достоверная информация может стать недостоверной, так как обладает свойством устаревать, то есть реально существующий объект меняется, а информация о нем остается неизменной. В качестве примера можно привести карты в Интернете и данные навигатора, которые необходимо регулярно обновлять для обеспечения достоверности.

3. *Полнота (достаточность)*. Информация является полной, если она содержит минимальный, но достаточный для правильного решения набор сведений. Если сведений больше, чем необходимо, то такая информация называется избыточной. Неполная или избыточная информация снижает эффективность принимаемых решений. Например, информация о том, что занятие будет проводиться с таким-то взводом, в такой-то аудитории, по такому-то предмету и в определенное время является полной. Если изъять любую из составляющих, то информация станет неполной.

4. *Ценность*. Определяется степенью актуальности информации для решения поставленной задачи, а также тем, какое она найдет применение в дальнейшей деятельности человека. Например, для курсанта вопросы к семинару, который будет проходить завтра, являются ценной информацией, а сведения об особенностях жизни африканских племен – неценной информацией (если это не является предметом обучения).

5. *Доступность*. Определяет соответствие информации тезаурусу пользователя, а также его возможности воспринимать информацию с целью дальнейшего преобразования. Величину доступности можно определять с помощью коэффициента сопряжения.

6. *Своевременность*. Определяет поступление информации не позднее заранее назначенного момента времени, согласованного со временем решения задачи. Если информация поступила позже, то ее ценность может быть снижена до нуля. Например, информация о том, что поезд отправляется в 15.30, если она поступила в 16.00, является абсолютно бесполезной и несвоевременной.

¹ Могилев А.В. Информатика: учебник / А.В. Могилев, Н.И. Пак, Е.К. Хённер. – М., 2001. С. 22.

7. *Надежность (помехоустойчивость)*. С помощью этой характеристики можно определить наличие ошибочно переданных приемнику информации сообщений. В том случае если количество ошибок превышает определенный предел, то информация признается ненадежной, и ее дальнейшее использование не допускается.

Все характеристики информации имеют технический характер и могут быть измерены.

Свойства информации:

1. *Запоминаемость*. Для начала работы с информацией она должна быть сохранена в каком-либо виде – запомнена. С юридической точки зрения, запоминание информации – это ее фиксация.

2. *Передаваемость (воспроизводимость, тиражируемость)*. Сохраненная информация может быть передана в виде сигнала неограниченное количество раз. Информация является уникальным неиссякаемым и неистощимым ресурсом. Можно изготовить неограниченное количество копий одной и той же информации.

3. *Преобразуемость (изменяемость)*. Информация может менять способ и форму своего существования. При преобразовании количество информации меняется. Преобразование информации может являться редактированием существующей информации или созданием новой информации. Возможны различные сочетания двух указанных видов преобразования информации.

4. *Стираемость (уничтожение)*. Информация может быть уничтожена. Уничтожение может быть безвозвратным, если до того информация не была скопирована. Если уничтожение осуществляется после копирования, то такое действие с информацией называется перемещением.

Свойства информации имеют скорее философский, чем технический характер. Именно на свойствах информации базируются основы правового регулирования оборота информации.

Необходимо отметить, что информационные отношения построены в основном на обороте информации, зафиксированной определенным образом, т.е. имеющей организационную форму. Эта форма может быть выражена в виде документа, книги, файла, сигнала и т.д. В результате создается информационный объект, которым уже и управляет право. Однако взаимоотношения информации и права складываются сложно, поскольку во многих случаях правовые нормы не учитывают тиражируемость информации. Если информация представлена в виде информационного объекта в единичном числе, то проблем при правовом регулировании не возникает. При множественности информационных объектов возникают сложности, а в случае множественно распространяемой информации, не являющейся информационным объектом, правовое регулирование ее оборота весьма проблематично.

1.4. Информатика и информационные технологии

Термин «информатика» возник в середине 60-х годов прошлого столетия во Франции и трактовался как «информационная автоматика»; в Англии и США соответствовал «науке о компьютерной технике».

Ему предшествовал другой термин – «документалистика», употребляющийся для определения деятельности по сбору, хранению и распространению документов. С годами поле деятельности документалистики резко возросло – она превратилась в научную дисциплину по сбору, распространению, переработке и использованию информации.

В середине 70-х гг. появилась новая тенденция в определении информатики, связанная с процессами обработки и передачи информации.

В нашей стране в 1983 г. термин «информатика» трактовался академией наук СССР как «комплексная научная инженерная дисциплина, изучающая все аспекты разработки, проектирования, создания, оценки, функционирования основанных на ЭВМ систем переработки информации, их применения и воздействия на различные области социальной практики».

На сегодняшний день существует много различных определений информатики, которые характеризуют многогранность ее функции. Одно из наиболее общих определений информатики выглядит следующим образом:

***Информатика** – это область человеческой деятельности, связанная с процессами преобразования информации с помощью компьютеров и их взаимодействием со средой обитания.*

Информатика в широком смысле представляет собой единство разнообразных отраслей науки, техники и производства, связанных с переработкой информации с помощью компьютеров и телекоммуникационных средств связи во всех сферах человеческой деятельности.

В настоящий момент чаще используется термин информационные технологии, несмотря на то, что информатика – это более широкое понятие. В соответствии со статьей 2 Федерального закона от 27.07.2006 № 149-ФЗ «Об информации, информационных технологиях и о защите информации»:

***информационные технологии** – процессы, методы поиска, сбора, хранения, обработки, предоставления, распространения информации и способы осуществления таких процессов и методов.*

В настоящий момент информатика и информационные технологии может восприниматься с трех позиций:

1. *Информатика и информационные технологии являются самостоятельной отраслью мировой экономики.* В информационную деятельность включают не только деятельность, имеющую отношение к производству компьютерной техники и программного обеспечения. К информационной деятельности относятся также телекоммуникационные системы и системы обеспечения использования информационных ресурсов. Начиная от Интернета, социальных сетей, компьютерных игр и т.п., заканчивая мобильной связью, спутниковым позиционированием, другими телекоммуникационными системами.

2. *Информатика и информационные технологии являются отраслью науки*, которая изучает структуру и свойства информации, а также закономерности ее обработки. Предметом изучения науки информатики являются процессы сбора, получения, хранения, поиска, переработки, хранения и использования информации. Также к сфере деятельности информационных технологий как науки относится получение фундаментальных знаний об информационных системах и разработка методологии процессов управления любыми объектами на базе компьютерных информационных систем.

3. *Информатика и информационные технологии являются прикладной научной дисциплиной в других отраслях науки*. Она занимается изучением закономерностей в информационном процессе, созданием информационных моделей, систем и технологий в конкретных предметных областях.

Центральное место в прикладной информатике занимает компьютер как основное техническое средство информатики. Кроме компьютера к техническим средствам информатики можно отнести средства тиражирования (полиграфическое оборудование, копировальные аппараты), средства связи (телефон, факс, радио, телекоммуникационные системы), технику управления и конторского труда (счетные и множительные приборы и аппараты).

Комплекс индустрии информатики и информационных технологий является ведущим в информационном обществе. Произошла всеобщая информатизация общества. При этом информатика является связующей нитью между разными отраслями науки и техники.

Кроме всего вышеперечисленного, информатика и информационные технологии являются учебной дисциплиной, изучаемой в школах и вузах вне зависимости от специальности получаемого образования. На настоящий момент информатика является такой же обязательной учебной дисциплиной, как физкультура, иностранный язык, философия и т.д.

Вопросы для самоконтроля:

1. Раскройте смысл понятия «информация».
2. Дайте определение понятиям информационной системы, передатчика и приемника информации.
3. Поясните принцип работы информационной системы.
4. Дайте определение понятиям сигнала и помехи.
5. Какие бывают виды сигналов?
6. Какие применяются преобразования сигналов?
7. Дайте определение понятиям тезауруса и сопряжения приемника с сигналом.
8. Что такое обработка информации и информационный процесс?
9. Перечислите характеристики информации.
10. Перечислите свойства информации.
11. Раскройте понятия «информатика» и «информационные технологии».

2. КОДИРОВАНИЕ И ПРЕДСТАВЛЕНИЕ ИНФОРМАЦИИ В ЭВМ

2.1. Измерение количества информации

Как измерить информацию? Воспользуемся основными положениями теории информации.

Упорядоченная последовательность символов, предназначенная для передачи единичной порции информации, в теории информации называется **сообщением**.

Упорядоченный конечный набор символов (character set), предназначенный для кодирования информационного сообщения в процессе его интерпретации и обработки, называют **алфавитом**. Количество символов в наборе – **мощностью алфавита**. Мощность алфавита из арабских цифр – 10. Мощность алфавита русского языка – 33, английского языка – 26.

С помощью алфавита русского языка {А, Б, ...Я} можно точно отобразить любое слово русского или, с определённой «похожестью», иностранного языка. С помощью арабских цифр можно отобразить множество натуральных чисел. Если в этот алфавит добавить знак «-», то можно будет отобразить любое целое число. Если добавить знак разделителя «,», то алфавит позволит отобразить любое рациональное число.

Представим информационное сообщение, которое может принимать одно из десяти значений. Если их закодировать десятичными цифрами {0, 1, 2, ... 9}, то для передачи такого сообщения понадобится одна десятичная цифра. Последовательность из двух элементарных сообщений будет представлена двумя цифрами, из трех – тремя и т.д. В данном случае последовательность из N цифр представляет N элементарных сообщений.

Для примера рассмотрим пронумерованный список пилотов Формулы-1:

0	Дженсон Баттон
1	Кими Райконен
2	Льюис Хэмилтон
3	Марк Веббер
4	Даниил Квят
5	Нико Росберг
6	Роман Грожан
7	Себастьян Феттель
8	Фелипе Масса
9	Фернандо Алонсо

После завершения очередного этапа соревнований распределение мест, начиная с первого, представлено следующим образом:

1 9 4 2 8 0 5 3 6 7

Из данной строки ясно видно, кто из пилотов каким пришел на финиш.

Если с помощью того же символьного набора $\{0, 1, 2, \dots, 9\}$ нужно закодировать сообщение, принимающее одно из ста различных значений, то потребуются две десятичные цифры (00, 01, 02, ... 99). Каждый дополнительный десятичный разряд позволяет увеличить в 10 раз число распознаваемых значений. Количество разрядов, необходимых для кодирования N состояний сообщения, растет как десятичный логарифм числа N ($\log_{10}N$). Основание логарифма равно мощности кодового алфавита.

При алфавитном подходе общий вид формулы для расчета количества разрядов, необходимых для кодирования N состояний (исходов) сообщения с помощью алфавита мощностью K :

$$I = \log_k N$$

Любое сообщение можно закодировать алфавитом, состоящим из двух символов. Одним символом такого множества $\{0, 1\}$ можно закодировать два возможных значения сообщения, двумя символами – 4 (00, 01, 10, 11) и т.д. Следовательно, количество разрядов, необходимых для кодирования N значений сообщения с помощью алфавита $\{0, 1\}$, рассчитывается как

$$I = \log_2 N$$

Например, нужно на дежурство по столовой случайным образом выбрать одного из ста курсантов. Независимо от значений их персональных данных, каждого курсанта можно закодировать двумя десятичными разрядами:

$$I = \lg 100 = 2 \text{ десятичных разряда.}$$

Чтобы представить в двоичной кодировке 256 различных цветовых оттенков, потребуется:

$$I = \log_2 256 = 8 \text{ двоичных разрядов.}$$

На первых персональных компьютерах для представления печатных знаков английского алфавита и служебных управляющих кодов использовалась американская таблица ASCII¹ на 128 кодов.

Но компьютерный алфавит состоит из двух знаков: $\{0, 1\}$.

Вычисляем для двоичного представления $I = \log_2 128 = 7$ разр.

Для кодирования букв национальных алфавитов и псевдографики код ASCII увеличили до 8 бит, что позволило использовать в тексте 256 символов и управляющих кодов.

Для расчета максимального количества различных значений, которое может принимать элементарное сообщение, закодированное с помощью алфавита мощностью K , применяется формула:

$$N = K^I, \text{ где } I \text{ – количество разрядов в сообщении,}$$

¹ American Standard Code for Information Interchange.

или

$N = 2^l$ для кодового алфавита $\{0, 1\}$.

В середине XX в. один из создателей математической теории информации, американский инженер и учёный Клод Элвуд Шеннон, изучая системы передачи информационных сигналов, пришёл к выводу, что трудность передачи сообщения не зависит от его содержания; передавать бессмысленные сообщения не менее трудно, чем осмысленные. Если смысловая часть информационного сообщения зависит от контекста и семантики, то трудность его передачи зависит от количества возможных кодированных состояний, которые должны быть распознаны приемником.

Шеннон определил, что любое принятое сообщение – это снятая неопределенность, а единицу измерения минимальной порции информации, которую может передать источник, содержащий два равновероятных сообщения, предложил называть битом (binary digit – двоичная цифра).

Бит – это количество информации, которую может передать источник, содержащий два равновероятных информационных сообщения или, другими словами, сообщение с двумя равновероятными исходами. При алфавитном подходе **бит** – это минимальное количество информации, соответствующее одному разряду в двоичной кодировке (0 или 1).

Поясним вероятностный подход к измерению количества информации.

При подбрасывании монетки могут выпасть орел или решка, т.е. любое подбрасывание монетки сопровождается возникновением 1 бита информации.

В игре «Угадай число» один участник загадывает число из заданного интервала. Например, в интервале от 1 до 16 загадано число 3. Второй игрок должен угадать число с помощью минимального количества вопросов, подразумевающих ответы только «Да» или «Нет». Ниже представлен пошаговый алгоритм оптимального решения:

Шаг	Вопрос	Ответ	Получено информации	Неопределенность знания (количество возможных исходов)
	Старт		(xxxx)	16
1.	$A > 8$	Нет (0)	1 бит (0xxx)	8
2.	$A > 4$	Нет (0)	2 бит (00xx)	4
3.	$A > 2$	Да (1)	3 бит(001x)	2
4.	$A = 3$	Да (1)	4 бит(0011)	Финиш

Могут быть другие пути решения. Если отгадывающий игрок знает, что ведущий почти всегда загадывает «3», т.е. вероятность данного исхода значительно выше, он может на первом шаге (1 бит) угадать: « $A = 3$ ».

В случаях когда наступление одного исхода элементарного сообщения более вероятно, для его кодирования можно использовать укороченное кодовое слово, чем для менее вероятных.

Проиллюстрируем на примере.

На экзамене по информатике курсанты могут получить оценки:

5, 4, 3, 2

Предположим, что курсанты 112 взвода выучили не все билеты одинаково хорошо и могут с одинаковой вероятностью получить любую оценку:

$$P_5 = P_4 = P_3 = P_2 = \frac{1}{4}.$$

Имеем 4 равновероятных исхода, каждый из которых можно закодировать 2-хразрядным двоичным словом:

- 5 – 00
- 4 – 01
- 3 – 10
- 2 – 11

Тогда кодовая строка с результатами экзамена 20 курсантов по номерованному списку взвода может выглядеть следующим образом:

10	00	01	11	01	00	10	11	01	00	10	01	00	11	00	11	01	10	11	10
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Информационный объем сведений об оценках 20 курсантов составит 40 бит для любого взвода с такой подготовкой.

Теперь представим, что вероятности получения оценок распределяются иначе:

$$P_5 = \frac{1}{2}; P_4 = \frac{1}{4}; P_3 = \frac{1}{8}; P_2 = \frac{1}{8},$$

и часто получаемую оценку 5 закодируем коротким однобитовым словом, а менее вероятные 3 и 2 – трехбитовыми словами:

- 5 – 0
- 4 – 10
- 3 – 110
- 2 – 111

Распознать при приеме сообщения, закодированные словами разной длины, не сложнее, чем одинаковой, но общий информационный объем кодовой строки может быть короче:

0	0	10	10	0	110	0	10	111	0	111	10	0	0	110	0	110	0	10	0
---	---	----	----	---	-----	---	----	-----	---	-----	----	---	---	-----	---	-----	---	----	---

$$I = 35 \text{ бит.}$$

Подобный вероятностный подход используется на практике, например, в коде Морзе, где чаще встречающиеся буквы обозначаются более короткими комбинациями точек и тире:

А • –	И • •	Р • – •	Ш – – – –	1 • – – – –	б – • • • •
Б – • • •	Й • – – –	С • • •	Щ – – • –	2 • • – – –	7 – – • • •
В • – –	К – • –	Т –	Ъ, Ъ – • • –	3 • • • – –	8 – – – • •
Г – – •	Л • – • •	У • • –	Ы – • – –	4 • • • –	9 – – – – •
Д – • •	М – –	Ф • • – •	Э • • • • •	5 • • • • •	0 – – – – –
Е •	Н – •	Х • • • •	Ю • • – –		
Ж • • • –	О – – –	Ц – • – •	Я • – • –		
з – – • •	П • – – •	Ч – – – •			

Количество информации объемом 1 бит – очень незначительная величина.

Минимально адресуемой порцией данных при их записи/чтении в компьютерной памяти является байт.

$1 \text{ байт} = 1 \text{ Б} = 8 \text{ бит}$
--

Единицы бит и байт применяют с приставками СИ в соответствии с международным стандартом МЭК 60027-2, но это не совсем корректно. При определении реального количества информации нужно понимать, что:

1 Кбайт = 2^{10} байт=1024 байт, а не 1000 байт!

1 Мбайт = 2^{20} байт=1024 Кбайт=1 048 576 байт, а не 1 000 000 байт!

1 Гбайт = 2^{30} байт=1024 Мбайт=1 073 741 824 байт, а не 10^9 байт!

1 Тбайт= 2^{40} байт=1024 Гбайт=1 099 511 627 776 байт, а не 10^{с 12 нулями}!

В Тбайтах измеряют объём современных жёстких дисков. Разница между величинами с коэффициентами в двоичном и десятичном представлении составляет почти 10 %.

2.2. Электронная вычислительная машина, машинный код

Вычислительная машина (VM, computer) – совокупность технических средств, создающая возможность проведения обработки информации и получение результата в необходимой форме.

Электронная вычислительная машина (ЭВМ, electronic computer) – вычислительная машина, основные функциональные устройства которой выполнены на электронных компонентах¹.

Еще в 1940 г. Норберт Винер сформулировал следующие требования к вычислительным машинам²:

1. Центральные суммирующие и множительные устройства должны быть цифровыми.

2. Эти устройства, являющиеся, по существу, переключателями, должны состоять из электронных ламп, а не из зубчатых передач или электромеханических реле. Это необходимо, чтобы обеспечить достаточное быстродействие.

3. В соответствии с принципами, принятыми для ряда существующих машин белловских телефонных лабораторий, должна использоваться более экономичная двоичная, а не десятичная система счисления.

4. Последовательность действия должна планироваться самой машиной так, чтобы человек не вмешивался в процесс решения задачи с момента введения исходных данных до снятия окончательных результатов. Все логические операции, необходимые для этого, должна выполнять сама машина.

5. Машина должна содержать устройство для запасания данных. Это устройство должно быстро их записывать, надежно хранить до стирания, быст-

¹ ГОСТ 15971–90. Системы обработки информации. Термины и определения: утв. Постановлением Госстандарта СССР от 26.10.1990 № 2698 // Режим доступа: СПС «КонсультантПлюс».

² Винер Н. Кибернетика, или управление и связь в животном и машине. – 2-е изд. – М.: Наука; Главная редакция изданий для зарубежных стран, 1983. – 344 с.

ро считывать, быстро стирать их и немедленно подготавливаться к запасанию нового материала.

Почему же основной алфавит языка вычислительной техники состоит из двух знаков: 0 и 1?

Информация в электронном вычислительном устройстве циркулирует в виде электрических или электромагнитных сигналов. Если мы будем кодировать информацию десятичными цифрами, то для отображения одного разряда придется использовать устройства, которые должны переключаться в десять состояний. Распознавание этих состояний в каждый момент времени – процедура технически сложная, зависящая от уровня внешних помех.

Проще изготовить устройство, информационный сигнал которого на выходе в зависимости от входного воздействия может принимать два значения (например, напряжение есть/нет), и этот сигнал будет выдавать 1 бит информации. В каждый момент времени, называемый тактом, вычислительная система будет обрабатывать параллельно установленное количество бит информации, переключаясь из одного состояния в другое по заданному алгоритму.

В 1959 г. в вычислительном центре Московского государственного университета коллективом конструкторов под руководством Николая Петровича Брусенцова была разработана малая ЭВМ «Сетунь» на основе троичной логики $\{0,1,2\}$, до сих пор не имеющая аналогов в истории вычислительной техники.

В ней минимальная непосредственно адресуемая единица главной памяти называлась Трайт, который был равен 6 тритам и мог отображать $N = 3^6 = 729$ кодированных символов. Трайт был лишь немногим больше 8-битного байта, но уже достаточно велик, чтобы закодировать, например, алфавит, включающий русские и латинские заглавные и строчные буквы, цифры, математические и служебные знаки.

Компьютерная архитектура (computer architecture) – совокупность средств и правил, определяющих принципиальную организацию компьютерной системы, обеспечивающих взаимодействие устройств и (или) программ.

Машинный код – система команд (инструкций), которая интерпретируется непосредственно процессором или встроенными микропрограммами конкретного компьютера.

Каждая **машинная инструкция** описывает элементарную операцию, выполняемую процессором, и состоит из оператора и операндов.

Программа для выполнения алгоритмов в машинных кодах называется **микропрограммой** и, как правило, реализуется на аппаратном уровне (в постоянном запоминающем устройстве – ПЗУ).

Очевидно, что только процессор легко «понимает» программу на своем машинном коде. В ходе эволюции компьютеров программы стали описываться на языках высокого уровня, более понятных человеку, и переводиться в машинные коды с помощью специализированного инструментального программного обеспечения.

Семейство процессоров, объединенное общей архитектурой, имеет свой собственный набор инструкций. Процессор А совместим с процессором В, если процессор А «понимает» машинный код процессора В.

Машинное слово – порция информации, зависящая от архитектуры компьютера и обрабатываемая в каждый момент времени (такт) как единое целое.

Машинное слово измеряется в байтах или битах и определяет следующие характеристики компьютера:

- структуру и содержание машинной команды;
- объем данных, обрабатываемых за один такт центральным вычислительным устройством (разрядность процессора);
- объема адресуемой порции данных при их чтении/записи процессором (разрядность шины данных);
- границы числового диапазона, напрямую поддерживаемого процессором и др.

В течение нескольких десятилетий величина машинного слова составляла 16 бит (2 байта), что в языках программирования соответствует типу данных «целое число»; 32 бита (4 байта) – двойное целое число.

В современной компьютерной литературе можно встретить базовый размер машинного слова 32 разряда (4 байта), тогда 16 разрядов – полуслово, 64 – двойное слово.

Например, для процессоров архитектуры IBM-PC префикс x86 означает, что машинное слово равно 32 битам. В зависимости от программы обработки данных одно и то же машинное слово может интерпретироваться процессором как:

- машинная инструкция;
- число;
- текстовый символ;
- элемент графического изображения;
- элемент звуковой последовательности;
- элемент видео и др.

Машинная инструкция содержит двоичный код элементарной операции, коды операндов и указание на адрес следующей команды. Код операции указывает процессору, что и как нужно сделать, коды операндов указывают, с какими данными произвести операцию.

Первые компьютеры назывались электронными вычислительными машинами (ЭВМ), так как были предназначены для математических расчетов – обработки числовой информации.

Целые числа являются простейшими числовыми данными, с которыми оперируют компьютерные программы, и представляются в беззнаковом формате (для неотрицательных целых чисел) и со знаком.

Для **беззнакового** представления целых чисел все разряды слова данных отводятся под кодирование самого числа. Максимальное значение числа рассчитывается по формуле:

$$I_{\max} = 2^n, \text{ где } n - \text{ количество двоичных разрядов.}$$

Например, одним байтом (8 бит) можно закодировать целые числа без знака в диапазоне $[0 \div 255]$.

7	6	5	4	3	2	1	0	разряд числа двоичный код = 137 ₁₀
1	0	0	0	1	0	0	1	
2 ⁷		+		2 ³		+	2 ⁰	

Рис. 4. Представление числа без знака одним байтом

Для представления **числа со знаком** самый старший (левый) бит отводится под знак числа, остальные разряды – под само число. Если число положительное, то знаковый разряд = 0, отрицательное = – 1.

В современных программах используют **прямой, обратный и дополнительный коды** чисел со знаком.

Представление, когда старший разряд отводится под знак, а остальные – под двоичную запись числа, называется **прямым кодом**.

знак	6	5	4	3	2	1	0	разряд числа двоичный код = – 9 ₁₀
1	0	0	0	1	0	0	1	
–				2 ³		+	2 ⁰	

Рис. 5. Представление числа одним байтом в прямом коде

При выполнении арифметических операций числа со знаком обычно представляются в **обратном** или **дополнительном** коде.

Для положительных чисел (знаковый разряд слова данных = 0) прямой, обратный и дополнительный коды совпадают.

Обратный код отрицательного числа получается инвертированием всех двоичных разрядов (кроме знака), т.е. все 0 меняются на 1 и наоборот.

Дополнительный код отрицательного числа получается инвертированием всех двоичных разрядов (кроме знака) и прибавлением числа 1 к получившемуся обратному коду.

Алгоритм преобразования чисел из прямого кода в обратный и дополнительный код достаточно прост:

$$\text{если прямой код } A > 0 \Rightarrow A_{\text{обр}} = A_{\text{доп}} = A, \text{ иначе } A_{\text{обр}} = \bar{A}; A_{\text{доп}} = \bar{A} + 1^1$$

	знак	6	5	4	3	2	1	0
Прямой код (I)	1	0	0	0	1	0	0	1
Обратный код (I _{обр})	1	1	1	1	0	1	1	0
Дополнительный код (I _{доп})	1	1	1	1	0	1	1	1

Рис. 6. Представление числа в прямом, обратном и дополнительном коде

Справедливо и обратное преобразование чисел из обратного и дополнительного кода в прямой код:

$$\begin{aligned} \text{если } A_{\text{обр}} > 0 &\Rightarrow A = A_{\text{обр}}, \text{ иначе } A = \bar{A}_{\text{обр}}; \\ \text{если } A_{\text{доп}} > 0 &\Rightarrow A = A_{\text{доп}}, \text{ иначе } A = \bar{A}_{\text{доп}} + 1. \end{aligned}$$

¹ Черта над переменной обозначает ее обратное значение или инверсию.

Представление чисел в дополнительном и обратном коде упрощает систему машинных команд для выполнения арифметических операций, позволяя исключить операцию вычитания, заменив ее на операцию сложения:

$$A - B = A + (-B).$$

Прямой, обратный и дополнительный коды позволяют максимально точно представлять целые числа в пределах диапазона, ограниченного разрядностью слова данных. При кодировке целых чисел со знаком однобайтовым словом можно закодировать значения в диапазоне $[-128 \div +127]$, двумя байтами – $[-32\,768 \div +32\,767]$. Для отображения больших значений в программировании применяют 2-х, 4-хбайтовые и выше целочисленные типы данных.

Для кодирования **рациональных чисел** используется представление с фиксированной точкой¹, когда слово данных делится на два фиксированных сегмента, кодирующих целую и дробную части.

знак	целая часть	дробная часть
1	1101	101

= -13,625₁₀

Рис. 7. Формат числа с фиксированной точкой

Недостатками кодирования с фиксированной точкой являются относительно небольшой диапазон и точность представления чисел из-за ограниченного количества двоичных разрядов целой и дробной части.

При обработке данных на компьютере экспоненциальная форма представления действительного числа называется кодированием с **плавающей точкой** (иногда – с плавающей запятой).

Стандарт IEEE 754 определяет три формата слов данных с плавающей точкой:

- с одинарной точностью (32 бита),
- с удвоенной точностью (64 бита),
- с повышенной точностью (80 бит).

В соответствии с указанным стандартом слово данных с плавающей точкой представляется в виде набора битов, одна часть которых кодирует мантиссу числа в нормализованном экспоненциальном виде, другая часть – экспоненту (порядок числа), и один бит используется для знака числа:

<i>одинарная точность</i>			
знак	экспонента		мантисса
1 бит	1 бит	7бит	23 бит
<i>удвоенная точность</i>			
знак	экспонента		мантисса
1 бит	1 бит	10бит	52 бит

Рис. 8. Форматы стандарта IEEE с плавающей точкой

¹ Иногда это представление называют «с фиксированной запятой».

Диапазон действительных чисел, которые можно закодировать данным способом, значительно увеличивается, но иногда за счет потери точности. Ниже в таблице показаны минимальные и максимальные значения чисел с плавающей точкой в зависимости от заданной точности.

IEEE 754	Точность	Слово (бит)	Диапазон	Экспонента (бит)
Half precision	половинная	16	$6,10 \cdot 10^{-5} \div 65504$	5
Single precision	одинарная	32	$3,4 \cdot 10^{-38} \div 3,4 \cdot 10^{38}$	8
Double precision	удвоенная	64	$1,7 \cdot 10^{-308} \div 1,7 \cdot 10^{308}$	11
Extended precision	повышенная	80	$3,4 \cdot 10^{-4932} \div 3,4 \cdot 10^{4932}$	15

Сложение и вычитание двух чисел, представленных в экспоненциальной форме с заданной степенью точности (например, 2 знака после запятой), осуществляется после приведения их к одному порядку:

$$A = 3,22E+01 = 3,22 * 10^1 = 32,2;$$

$$B = 2,34E-01 = 2,34 * 10^{-1} = 0,234;$$

$$A + B = 32,2 + 0,234 = 32,434 = 3,2434 * 10^1 \approx 3,24E+01;$$

$$A - B = 32,2 - 0,234 = 31,926 = 3,1926 * 10^1 \approx 3,19E+01.$$

В процессе эволюции одновременно с числовой информацией на электронных вычислительных машинах стали обрабатывать тексты, графику, видео и т.п.

Наиболее популярным 7-битовым стандартом кодировки текстовой (символьной) информации в ранних моделях персональных компьютеров являлся ASCII, который за счет дополнительного 8-го бита содержал расширенную часть (десятичные коды: $128 \div 255$).

На ранних моделях персональных компьютеров национальные кодовые страницы устанавливались дополнительно после установки операционной системы. Некоторые модели современных компьютеров до сих пор нуждаются в русификации, т.к. при отсутствии необходимых кодовых страниц или неправильном выборе кодовой страницы в текстовом редакторе вместо понятных русских слов на экране или принтере отображается абракадабра.

Альтернативой ранним системам компьютерного кодирования текста стал многобайтный Юникод (Unicode), предложенный еще в 1991 г. в качестве международного стандарта. Первая версия Юникода представляла собой двухбайтовую кодировку, часто отображаемую четырьмя шестнадцатеричными цифрами (например, U+0410), то есть максимальное число различных кодов было 2^{16} (65 536).

Юникод разделен на несколько диапазонов. Область от U+0000 до U+007F содержит символы основного набора ASCII. Далее расположены знаки национальных алфавитов, знаки пунктуации и графические символы. Под символы кириллицы выделены диапазоны [U+0400 ÷ U+052F], [U+2DE0 ÷ U+2DFF], [U+A640 ÷ U+A69F].

Юникод имеет форматы UTF (англ. Unicode transformation format): UTF-8, UTF-16 и UTF-32. В слове данных старшие байты могут следовать перед младшими (UTF-16 big-endian, UTF-32 BE) либо после младших (UTF-16 little-endian, UTF-32 LE).

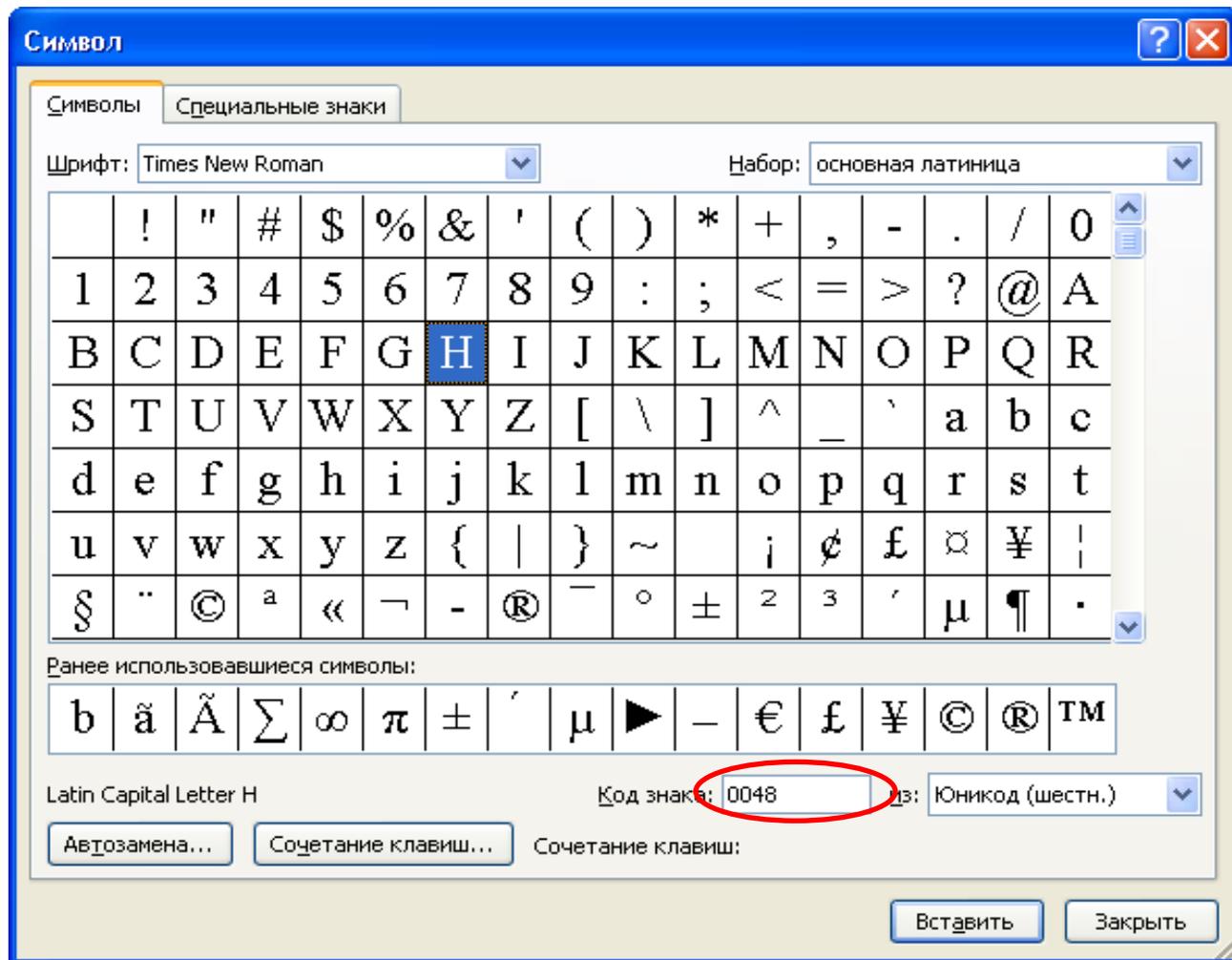


Рис. 9. Юникод UTF-16LE

В программах Microsoft в основном используется формат UTF-16LE. В операционных системах семейства UNIX и Mac OS X принята форма UTF-8 или UTF-32.

Внедрение Юникода привело к изменению подхода к байтовым кодировкам. Если раньше кодировка задавалась набором шрифтовых знаков, то теперь она задаётся таблицей соответствия с символами Юникода.

2.3. Кодирование графических изображений

Современная графика содержит элементы, каждый из которых кодируется как массив данных. С точки зрения основных принципов формирования и обработки изображения, отображаемого на экран монитора или выводимого на печать, различают следующие **способы представления изображения**:

- растровый;

- векторный;
- фрактальный;
- трёхмерный (3D).

Минимальным элементом **растрового изображения** является **точка** или **пиксел**. Каждое растровое изображение имеет форму прямоугольника и состоит, как мозаика, из разноцветных пикселей. Чем больше пикселей содержится в изображении, тем выше качество прорисовки деталей.

Величина, определяющая отношение количества элементов растрового изображения на единицу длины, называется **разрешением** или **разрешающей способностью**.

В разном контексте термин разрешение может иметь разный смысл. В одном случае разрешение определяет размер изображения при заданном качестве в другом, наоборот, качество изображения при заданном размере.

Разрешение изображения – это конкретная величина, которая задается при создании изображения в графическом редакторе или при его оцифровке из некомпьютерного формата (сканирование, фотографирование) и хранится в файле изображения. Разрешение изображения измеряется в пикселях на дюйм – ppi (англ. pixels per inch – пиксели на дюйм).



Рис. 10. Соотношение параметров изображения в пикселях, сантиметрах и пикс./дюйм

Изображение может быть монохромное (черно-белое), серое или цветное. Каждый пиксел растрового изображения характеризуется глубиной цвета.

Глубина цвета – количество бит, кодирующее каждый цвет.

Если для «раскраски» каждого пикселя отводится, например, 1 бит, то изображение получится монохромное (черно-белое). Один байт позволяет закодировать 256 цветов или оттенков серого цвета в диапазоне от белого до черного.

Большинство цветовых оттенков образуется смешением нескольких основных цветов спектра. Спецификация, описывающая правила разделения цветового оттенка на составляющие компоненты, называется **цветовой моделью**. В компьютерной графике применяются цветовые модели: RGB (красный, зеленый, синий), CMYK (голубой, пурпурный, желтый, черный) и др.

Для цветовых моделей глубина цвета позволяет пропорционально закодировать градации яркости базовых цветов.

	Коды основных цветов (R,G,B)
черный	[0,0,0]
белый	[255, 255, 255]
красный	[255,0, 0]
зеленый	[0,255, 0]
синий	[0,0, 255]

Рис. 11. Трехбайтовое цветовое кодирование модели RGB

Перемножив горизонтальный и вертикальный размер растрового изображения в пикселях и умножив результат на глубину каждого цвета соответствующей цветовой модели, можно приблизительно оценить размер файла. Например, для цветовой модели RGB формула будет выглядеть следующим образом:

$$I = P_{гор} * P_{верт} * C_R * C_G * C_B = P_{гор} * P_{верт} * C^3.$$

На рисунке 12 представлена одна и та же фотография с экраным размером изображения 472x709, печатного оттиска 4x6 (см) и разрешением при печати – 300 (пикс./дюйм). Отличие – в разных цветовых моделях представления.

Реальные размеры файлов получились меньше расчетных из-за применения специальных алгоритмов сжатия растровых изображений.

В ряде случаев современные графические дизайнеры используют цветовую модель HSB, в которой цвет пикселя разлагается на три составляющие:

- hue (цветовой тон) – номер цвета в спектральной палитре;
- saturation (насыщенность) – чистота цвета или степень отсутствия «серых примесей»;
- brightness (яркость) – соотношение цветового тона и чёрного цвета. Нулевое значение этого параметра окрашивает пиксел в черный цвет.

Изображения растровой графики редко рисуют с помощью компьютерных программ «с нуля». Чаще для этой цели используют цифровые фотографии или отсканированные изображения. Растровая графика используется в тех случаях, когда надо передать максимум деталей и цветовых особенностей реальных объектов.

В **векторной графике** основным элементом изображения является **линия** (прямая, ломаная, дуга, замкнутая), кодируемая математическими формулами. Объем памяти, занимаемый векторным изображением, не зависит от его физического размера. В ячейках памяти хранятся только параметры математических формул, и при масштабировании линий изображения меняются только они.

Линии имеют характеристики: **форма, тип** (сплошная, пунктирная и т.п.), **толщина, цвет** и др. Простейшая линия, если она не замкнута, имеет две ограничивающие вершины, которые называются узлами. У сложных линий есть промежуточные узлы. Внутренняя область замкнутого контура имеет свойства заполнения: цвет, штриховка, текстура и др.

Векторная графика предназначена в первую очередь для создания графических объектов, не требующих максимально точного соответствия оригиналу или высокого качества прорисовки: эмблемы, логотипы, географические карты, схемы, шрифты и т.д.



RGB-8бит/канал
980 Кбайт



Grayscale-8бит
327 Кбайт



Битовый формат
41 Кбайт

Рис. 12. Зависимость размера изображения от его цветовой модели



Рис. 13. Масштабированное векторное изображение

Фрактальное изображение (лат. *fractus* – дроблённый, сломанный, разбитый) – сложная геометрическая фигура, обладающая свойством самоподобия, то есть составленная из нескольких частей, каждая из которых подобна всей фигуре целиком. Изображение является фактически графиком функции, описанной математическими уравнениями. В файле фрактального изображения сохраняются только алгоритмы и формулы. Фракталы позволяют описывать целые классы изображений, для детального описания которых требуется относительно мало памяти. С другой стороны, фракталы слабо применимы к изображениям вне этих классов.



Рис. 14. Фрактальное изображение

Создание художественной фрактальной композиции состоит не в рисовании или оформлении, а в математическом описании и программировании. Фрактальную графику редко применяют для создания печатных или электронных документов, но ее часто используют в развлекательных программах.

Трёхмерная графика (3D-графика) как самостоятельное направление векторной графики содержит приёмы и методы создания объёмных моделей, которые максимально соответствуют реальным объектам. Объёмные изображения можно вращать и рассматривать со всех сторон. Для создания объёмных изображений используют геометрические фигуры и гладкие поверхности. Обычно сначала создаётся каркас объекта, «раскрашивается» его поверхность, а потом имитируются освещение, прозрачность и другие параметры окружающего пространства. Для движущихся объектов задается скорость и траектория движения.

Трёхмерная графика широко используется в компьютерном моделировании физических объектов и процессов, в анимации, кинематографии и компьютерных играх.

Независимо от метода хранения и обработки, на цифровой экран или принтер все изображения выводятся в виде множества растровых точек. Следовательно, непосредственно перед выводом на экран или печать графического объекта все изображения перекодируются в растровые с установленным разрешением.

Если размер изображения в пикселях совпадает с экранным разрешением дисплея, то изображение будет занимать весь экран без дополнительных преобразований. Это соответствие можно использовать при подборе фоновых картинок рабочего стола MS Windows.

При оцифровке изображения на сканере используют также оптическое разрешение в пикселях по горизонтали и вертикали, например, 9600x4800 пикс.

Чтобы оценить качество растрового изображения, полученного на сканере, оптическое разрешение в пикселях делят на соответствующие габаритные размеры в дюймах или сантиметрах. Например, у сканера формата А4 область сканирования 30x22 см при оптическом разрешении 9600x4800 пикс. Получим:

разрешающая способность по вертикали – $9600/30 = 320$ пикс./см;

разрешающая способность по горизонтали – $4800/22 \approx 220$ пикс./см.

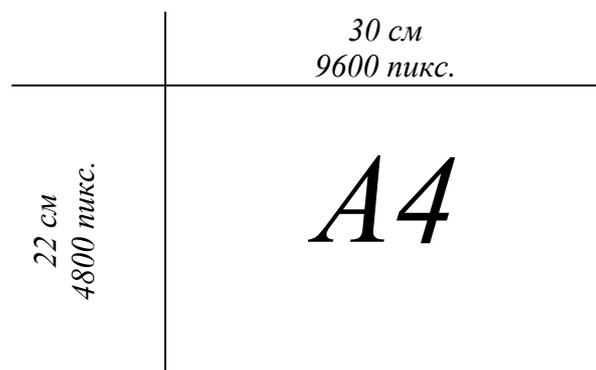


Рис. 15. Габаритные размеры области сканирования

Если сканируемый рисунок имеет линейные размеры 3х3 см, то при разрешении 320х220 его размер в пикселях: $320 * 3 \times 220 * 3 \Rightarrow 960 \times 660$ пикс.

Разрешение матрицы цифровой фотокамеры, так же как экрана монитора, характеризуется размером получаемых изображений в пикселях, но, в отличие от экранов, популярным стало использование не двух чисел, а округлённого значения суммарного количества пикселей, выражаемого в мегапикселях.

Например, есть фотоаппарат с матрицей 8 Мпикс и пропорциями 4:3, тогда:

$$4X_{\text{пикс.}} * 3X_{\text{пикс.}} = 8 \text{ Мпикс.} \Rightarrow X = \sqrt{8\,000\,000/12} \approx 816,5,$$

$A = 816,5 * 4 \approx 3\,265$; $B = 816,5 * 3 \approx 2\,450$, т.е. этим фотоаппаратом можно делать снимки с максимальным размером 3265х2450.

Разрешение принтера – это параметр, отражающий количество точек в одном дюйме печатного оттиска изображения и измеряемый в единицах dpi (англ. dots per inch – точки на дюйм). Чем больше будет концентрация точек в дюйме, тем более четкой будет картинка.

И дисплеи и принтеры, чтобы передавать истинный цвет каждого пикселя, комбинируют сочетание оттенков основных цветов модели. Цвето- и световосприятие глаза в проходящем (дисплей) и отраженном (принтер) свете различны. По этой причине экранное разрешение изображения и разрешение печатного оттиска – это совершенно разные вещи. Чтобы человеческий глаз не замечал отдельные пиксели в изображении, приняты следующие стандартные настройки:

72 ppi (англ. pixels per inch – пиксели на дюйм) – минимальное разрешение для качественного воспроизведения изображения на компьютерных мониторах или плакатах, разглядываемых издали;

300 ppi – минимальное разрешение изображения, обеспечивающее фотографическое качество печати.

2.4. Кодирование звука и видео

Звуковая информация в природе представляется чаще всего в виде непрерывных (аналоговых) сигналов, и для обработки на компьютере необходимо их преобразование в двоичные коды. Кодирование аналоговой аудиоинформации в цифровую форму (или **дискретизация**) осуществляется с помощью специальных устройств – аналого-цифровых преобразователей (АЦП), обратное преобразование осуществляется цифро-аналоговыми преобразователями (ЦАП).

АЦП в определённые моменты времени измеряет уровень входного звукового сигнала и переводит его значение в соответствующий двоичный код. Такой процесс называется импульсно-кодовой модуляцией.

Качество оцифровки звукового сигнала зависит в первую очередь от двух параметров:

частота считывания уровня сигнала;

количество разрядов, выделяемых на кодировку уровней сигнала.

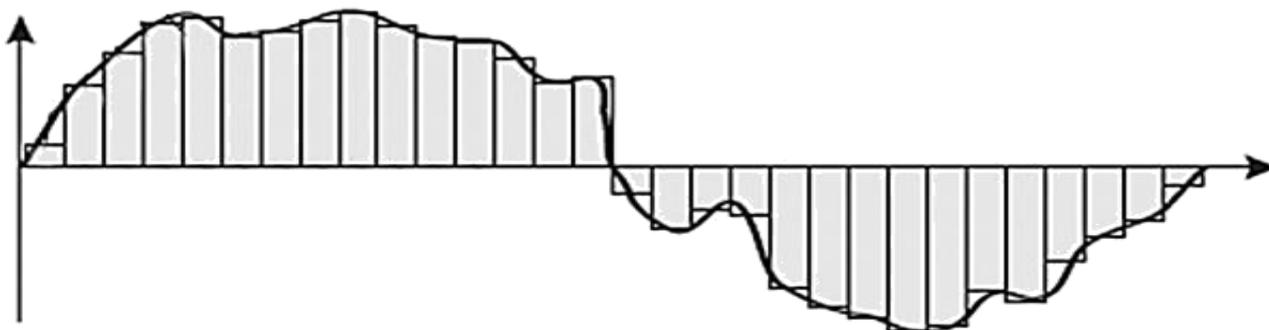


Рис. 16. Преобразование аналогового сигнала в цифровой сигнал

Частота временного шага выборки мгновенных значений звуковых сигналов называется **частотой дискретизации**. Измеряется в Герцах (Гц).

Количество разрядов, которыми кодируется амплитуда звукового сигнала в момент выборки, называется **глубиной звука**.

По аналогии с растровой графикой, когда более детальная прорисовка изображения получается за счет увеличения числа пикселей, а цветопередача – за счет глубины цвета, качество оцифрованного звука улучшается при увеличении частоты дискретизации (количества значений) и глубины звука. Но при этом увеличивается размер звукового файла.

Чистый звуковой тон, представляемый синусоидой, в природе встречается редко. Обычно мы слышим «симфонию» звуков различной частоты. Чтобы обеспечить приемлемое качество оцифровки в соответствии с теоремой Найквиста-Котельникова, частота дискретизации должна превышать максимальную составляющую частоту звука как минимум в 2 раза.

Человеческое ухо воспринимает звуковые колебания в диапазоне частот от 20 Гц до 20 кГц. Поэтому для записи качественного звука частота дискретизации должна превышать 40 кГц. Голосовое сообщение находится в меньшем диапазоне частот и проще в восприятии (например, голос в телефонной трубке), поэтому для микрофонной записи речи вполне достаточно будет установить частоту дискретизации в 8 КГц.

Глубина звука определяет, насколько схожими в деталях будут исходное и оцифрованное звучание. Например, если глубина звука 8 бит, то можно закодировать 256 различаемых значений амплитуд. Обычно глубину звука выбирают в 16 бит и выше.

Глубина звука, частота дискретизации и алгоритм сжатия определяют формат звукового файла при компьютерной обработке. Выделяют три основные группы аудиоформатов:

- без сжатия (wav, aiff);
- со сжатием без потерь (ape, flac);
- с потерями при сжатии (mp3, ogg).

Информационное **видеообщение** состоит из двух компонентов: видео и аудио. Кодирование аудиокompонента ничем не отличается от кодирования звука, описанного ранее.

Видеоизображение формируется как последовательность кадров, которые меняются с определенной частотой. Каждый кадр кодируется как обычное растровое изображение, то есть разбивается на множество пикселей. Эффект движения создается за счет быстрой смены кадров на экране. Чем больше частота кадров в секунду, тем более плавным и естественным будет казаться движение. В традиционном плёночном кинематографе используется частота 24 кадра в секунду. Системы телевидения PAL и SECAM используют 25 кадров в секунду, а система NTSC использует около 30 кадров в секунду (точнее 29,97 fps – frames per second). Компьютерные оцифрованные видеоматериалы хорошего качества, как правило, используют частоту 30 кадров в секунду и выше. Некоторые современные профессиональные камеры могут снимать с частотой до 120 кадров в секунду. А специальные камеры для сверхбыстрой съёмки снимают с частотой до 1000 кадров в секунду и выше, что необходимо, например, для детального изучения траектории полёта пули или структуры взрыва.

Количество кадров в секунду при выводе видео на экран иначе называют **частотой кадровой развертки** и измеряют в Герцах.

Построчная (прогрессивная) развёртка – метод отображения, передачи или хранения движущихся изображений, в котором все строки каждого кадра отображаются последовательно.

Чересстрочная развёртка – метод отображения, передачи или хранения изображений, при котором каждый кадр разбивается на два полукадра из четных и нечетных строк. Чересстрочная развёртка применяется для ускорения вывода изображений на экран при ограниченной полосе пропускания (в аналоговой) или ширине канала (в цифровой технике). По сравнению с прогрессивной (покадровой) разверткой, при чересстрочной развёртке частота вывода полукадров повышается в 2 раза.

По аналогии с разрешением компьютерных изображений, цифровой видеосигнал также имеет похожий параметр, но измеряемый в пикселях на линию, т.е. сначала указывается количество пикселей в горизонтальной линии (горизонтальное разрешение), а затем количество самих линий (вертикальное разрешение). Разрешение картинки у основных систем аналогового телевидения:

720x576 пикс./лин. для стандартов PAL и SECAM;

720x480 пикс./лин. для NTSC;

1920x1080 пикс./лин. для HDTV.

Количество цветов и глубина цвета описываются стандартными цветовыми моделями, в основном RGB.

Для стереовидео (3D) нужно два видеоканала, часто называемых слоями. Чтобы каждый канал правильно воспринимался «своим» глазом, используют оптические фильтры, установленные в очки.

Для сравнения: фильм длительностью 1,5 часа реального времени в цифровом видеформате DV занимает примерно 20 Гб, в MPEG-2 – 4,7 Гб (DVD-диск), а в формате MPEG-4 – 700 Мб (CD-диск).

Вопросы для самоконтроля:

1. Какие существуют единицы измерения информации?
2. Дайте определение понятию вычислительной машины и электронной вычислительной машины.
3. Дайте определение понятию компьютерной архитектуры.
4. Дайте определение понятию машинного кода.
5. Что используется для кодирования рациональных чисел?
6. Как при выполнении арифметических операций представляются числа со знаком?
7. Что представляют из себя целые числа?
8. Каковы требования к вычислительным машинам?
9. Почему основной алфавит языка вычислительной техники состоит из двух знаков: 0 и 1?
10. Дайте определение понятию машинного слова.
11. В чем измеряется машинное слово?
12. Что такое Юникод?
13. В чем отличие способов представления изображения: растрового, векторного, фрактального, трёхмерного.
14. Как кодируется звук?
15. Как кодируется видео?

3. СИСТЕМЫ СЧИСЛЕНИЯ

Системой счисления называют систему приемов и правил, позволяющих устанавливать взаимно-однозначное соответствие между любым числом и его представлением в виде совокупности конечного числа символов.

Множество символов, используемых для такого представления, называют цифрами. Проще говоря, система счисления это – способ представления любого числа с помощью символов алфавита.

В зависимости от способа изображения чисел с помощью алфавита системы счисления делятся на:

1. **Непозиционные.** В них любое число определяется как некоторая функция от численных значений совокупности цифр, представляющих это число. Цифры в непозиционных системах счисления соответствуют некоторым фиксированным числам. Например, непозиционной системе – римская система счисления. В вычислительной технике непозиционные системы не применяются.

2. **Позиционные.** В этих системах одна и та же цифра может принимать различные численные значения в зависимости от номера разряда этой цифры в совокупности цифр, представляющих заданное число. Пример такой системы – арабская десятичная система счисления.

В позиционной системе счисления любое число записывается в виде последовательности цифр:

$$A = a_{m-1} a_{m-2} \dots a_k \dots a_0, a_{-1} \dots a_{-n}.$$

Позиции, пронумерованные индексами k ($-n < k < m-1$), называются разрядами числа.

Сумма $m+1$ соответствует количеству разрядов числа (m – число разрядов целой части числа, n – дробной части).

Каждая цифра a_k в записываемой последовательности может принимать одно из N возможных значений.

Количество различных цифр (N), используемых для изображения чисел в позиционной системе счисления, называется основанием системы счисления.

Основание N указывает, во сколько раз единица $k+1$ -го разряда больше единицы k -го разряда, а цифра a_k соответствует количеству единиц k -го разряда, содержащихся в числе.

Основание позиционной системы счисления определяет ее название. В вычислительной технике применяются двоичная, восьмеричная, десятичная и шестнадцатеричная системы.

В дальнейшем чтобы явно указать используемую систему счисления, будем в индексе указывать основание системы счисления.

Таким образом, при переводе из одной системы счисления в другую число представляется в виде суммы:

$$A = a_{m-1} \cdot N^{m-1} + a_{m-2} \cdot N^{m-2} + \dots + a_k \cdot N^k + \dots + a_0 \cdot N^0 + a_{-1} \cdot N^{-1} + \dots + a_{-n} \cdot N^{-n}.$$

В двоичной системе счисления используются только две цифры: 0 и 1.

В восьмеричной системе счисления для записи чисел используется восемь цифр (0,1,2,3,4,5,6,7), а в шестнадцатеричной – шестнадцать (0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F).

Для хранения и обработки данных в ЭВМ используется двоичная система, так как она требует наименьшего количества аппаратуры по сравнению с другими системами. Все остальные системы счисления применяются для удобства пользователей.

В связи с тем, что в основу вычислительной техники положена двоичная система счисления, ограничимся ее рассмотрением.

Перевод числа из двоичной системы в другую выполняется по универсальному алгоритму: сумма произведений последовательности 0 и 1, составляющих двоичное число, на основании двоичной системы, причем степень основания соответствует разряду числа минус один:

$$A_2 = B_{10} = a_{m-1} \cdot 2^{m-1} + a_{m-2} \cdot 2^{m-2} + \dots + a_k \cdot 2^k + \dots + a_0 \cdot 2^0 + a_{-1} \cdot 2^{-1} + \dots + a_{-n} \cdot 2^{-n} .$$

Например, при переводе двоичного числа 10101,101 в десятичную систему счисления, используя указанную формулу, получаем:

$$10101,101_2 = 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 + 1 \cdot 2^{-1} + 0 \cdot 2^{-2} + 1 \cdot 2^{-3} = 21,625_{10} .$$

Обратное преобразование из десятичной системы счисления в двоичную происходит следующим образом: десятичное число делим на 2, полученное целое частное, в свою очередь, делим на 2 и т.д. После проведенных действий, двоичное число формируется из набора 0 и 1, начинающихся с последнего частного и остатков от деления:

$$26/2=13(\text{ост. } 0) \quad 13/2=6(\text{ост. } 1) \quad 6/2=3(\text{ост. } 0) \quad 3/2=1(\text{ост. } 1);$$

$$26_{10}=11010_2.$$

В двоичной системе очень просто выполняются арифметические и логические операции над числами.

Таблица сложения:				
0	+	0	=	0
0	+	1	=	1
1	+	0	=	1
1	+	1	=	10

Таблица умножения:				
0	*	0	=	0
0	*	1	=	0
1	*	0	=	0
1	*	1	=	1

В определенной степени денежная система является непозиционной системой счисления. Любая сумма может быть набрана комбинацией монет и/или купюр, которые в данном случае составляют алфавит.

Для сложения двух чисел в подобной системе нужно сложить символы одного «веса» и заменить «младшие» на «старшие» в соответствующей пропорции при «переполнении».

Вспомните, как маленькие деньги меняли на большие, чтобы кошелек не разбухал. Или, наоборот, когда в магазине нет сдачи с крупной купюры, приходится искать, где ее разменять.

Более совершенными были непозиционные системы счисления, основанные на буквенных алфавитах: греческом, славянском и др. В них числа от 1 до 9, целые количества десятков (от 10 до 90) и целые количества сотен (от 100 до 900) обозначались буквами алфавита.

Таблица 1. Греческая и славянские системы счисления

Число	Греческий алфавит	Кириллица		Глаголица	
1	Α, α	А (аз)	Ⓐ	А (аз)	Ⓐ
2	Β, β	В (веди)	Ⓑ	Б (буки)	Ⓑ
3	Γ, γ	Г (глаголь)	Ⓒ	В (веди)	Ⓒ
4	Δ, δ	Д (добро)	Ⓓ	Г (глаголь)	Ⓓ
5	Ε, ε	Е (есть)	Ⓔ	Д (добро)	Ⓔ
6	Ζ, ζ (стигма)	С (зело)	Ⓚ	Е (есть)	Ⓚ
7	Ζ, ζ	З (земля)	Ⓛ, Ⓜ	Ж (живете)	Ⓛ, Ⓜ
8	Η, η	И (иже)	Ⓝ	С (зело)	Ⓝ
9	Θ, θ	Θ (фита)	Ⓞ	З (земля)	Ⓞ
10	Ι, ι	Ι (и)	Ⓟ, Ⓠ	Ι (и)	Ⓟ
20	Κ, κ	Κ (како)	Ⓡ	И (иже)	Ⓡ
30	Λ, λ	Λ (люди)	Ⓢ	Ѓ (гервь)	Ⓢ
40	Μ, μ	Μ (мыслете)	Ⓣ	Κ (како)	Ⓣ
50	Ν, ν	Ν (наш)	Ⓤ	Λ (люди)	Ⓤ

Число	Греческий алфавит	Кириллица	Кириллица	Глаголица	Глаголица
60	Ξ, ξ	Ѓ (кси)	Ѓ	М (мыслете)	Ɑ
70	Ο, ο	О (он)	О	Н (наш)	Ɱ
80	Π, π	П (покой)	П	О (он)	Ɐ
90	Ϙ, ϙ (коппа)	Ч (червь)	Ϙ, ϙ	П (покой)	Ɒ
100	Ρ, ρ	Р (рцы)	ρ	Р (рцы)	ⱱ
200	Σ, Ϻ	С (слово)	Ϟ	С (слово)	Ⱳ
300	Τ, τ	Т (твёрдо)	Ϝ	Т (твёрдо)	ⱳ
400	Ο, ο и Υ, υ	У (ук)	Ϡ, ϡ	У (ук)	ⱴ
500	Φ, φ	Ф (ферт)	ϕ	Ф (ферт)	Ⱶ
600	Χ, χ	Х (хер)	ϗ	Х (хер)	ⱶ
700	Ψ, ψ	Ψ (пси)	ϙ	У (от)	ⱷ
800	Ω, ω	Ω (омега)	Ϡ	Щ (шта)	ⱸ
900	Ϡ, ϡ (сампи)	Ц (цы)	Ϡ, ϡ	Ц (цы)	ⱹ
1000	—	Ϡa	Ϡa	Ч (червь)	ⱺ

Чтобы отличать буквы от цифр, над буквами с числовым значением писался специальный знак «титло» (~). Этот знак мог ставиться над каждой буквой либо же он мог быть длинным и покрывать всё число. Для обозначения тысяч слева от соответствующей букво-цифры писалась маленькая диагональ влево вниз и на ней две маленькие черточки – † (U+0482).

†аψs – 1706 г.;

†зрї – 7118 год по летосчислению «от сотворения мира» (1610 год от Рождества Христова).

Еще одним примером системы счисления, сохранившейся до наших дней, может служить римская система, алфавит которой состоит из знаков: М, D, С, L, X, V, I.

Римская система счисления – непозиционная, но последовательность расположения символов уже имеет значение.

$\pm C$		$\pm X$		$\pm I$		
M	D	C	L	X	V	I
1000	500	100	50	10	5	1

Рис. 18. Римская система счисления

Чтобы записать число в римской системе, его нужно разложить на сумму тысяч, полутысяч, сотен, полусотен, десятков, пятерок и единиц и поменять их на символы алфавита:

$$37 = (10 + 10 + 10) + 5 + (1 + 1) = XXXVII.$$

Любая из римских цифр записывается не более трех раз подряд. Если меньшая цифра стоит справа от большей, то она прибавляется к его значению, а если слева, то вычитается из него:

$$\begin{aligned} 888 &= 500 + 388 = 500 + (100 + 100 + 100 + 88) = 500 + (100 + 100 + 100 + (50 + 38)) = \\ &= 500 + (100 + 100 + 100 + (50 + (10 + 10 + 10 + 8))) = \\ &= 500 + (100 + 100 + 100 + (50 + (10 + 10 + 10 + (5 + 1 + 1 + 1)))) = DCCCLXXXVIII; \end{aligned}$$

$$2012 = 1000 + 1000 + 12 = 1000 + 1000 + (10 + 2) = 1000 + 1000 + (10 + (1 + 1)) = MMXII;$$

$$\begin{aligned} 1989 &= 1000 + 989 = 1000 + (500 + 489) = 1000 + (500 + (100 + 100 + 100 + 100 + 89)) = \\ &= 1000 + ((500 + 100 + 100 + 100 + 100) + 89) = 1000 + (900 + 89) = \\ &= 1000 + ((-100 + 1000) + (50 + 39)) = 1000 + ((-100 + 1000) + (50 + (10 + 10 + 10 + 9))) = \\ &= 1000 + ((-100 + 1000) + (50 + (10 + 10 + 10 + (5 + 1 + 1 + 1 + 1)))) = \\ &= 1000 + ((-100 + 1000) + (50 + (10 + 10 + 10 + (-1 + 10)))) = MCMLXXXIX. \end{aligned}$$

Обратный перевод осуществляется, соответственно, в обратном порядке. Символьная последовательность преобразуется в сумму тысяч, полутысяч, сотен, полусотен, десятков, пятерок и единиц с учетом правила максимального трехкратного повторения:

$$\begin{aligned} IV &= -1 + 5 = 4; VI = 5 + 1 = 6; IX = -1 + 10 = 9; LXI = 50 + 10 + 1 = 61; \\ MMXV &= 1000 + 1000 + 10 + 5 = 2015; \\ CMXCIX &= (-100 + 1000) + (-10 + 100) + (-1 + 10) = 999. \end{aligned}$$

Недостатки непозиционных систем счисления:

представление больших чисел требует расширения алфавита;
невозможно представлять дробные и отрицательные числа;
сложно выполнять арифметические вычисления.

В позиционных системах счисления эти недостатки устраняются.

В позиционной системе счисления с основанием q (q -ичная система) для записи чисел требуется цифровой алфавит мощностью q $\{0, 1, \dots, q - 1\}$. Количественный эквивалент (значение) цифры зависит от ее позиции в записи числа.

Единицами разрядов служат последовательные степени числа q .

q единиц младшего разряда образуют 1 единицу старшего разряда.

В q -ичной системе счисления любое действительное число в **развернутой форме** может быть записано в следующем виде:

$$A_q = \pm (a_n * q^n + a_{n-1} * q^{n-1} + \dots + a_1 * q^1 + a_0 * q^0 + a_{-1} * q^{-1} + a_{-2} * q^{-2} + \dots + a_{-m} * q^{-m}) = \pm \sum a_i * q^i$$

q – основание q -ичной позиционной системы счисления (мощность алфавита);

a_i – разряд числа (цифра) в q -ичной системе счисления;

q^i – весовой (позиционный) коэффициент;

$n + 1$ – число целых разрядов числа, включая ноль;

m – число дробных разрядов числа.

В представлении чисел обычно используется свернутая (цифровая) форма записи числа:

$$A_q = \pm a_{n-1} a_{n-2} \dots a_1 a_0, a_{-1} \dots a_{-m}$$

Точность представления действительного числа зависит от количества разрядов. Примеры записи числа в различных системах счисления с использованием всех цифр алфавита:

$10,0101_2$ – двоичная система счисления;

210_3 – троичная система счисления;

$76543,210_8$ – восьмеричная система счисления;

$FEDCB, A9876543210_{16}$ – шестнадцатеричная система счисления.

Десятичное значение q -ичного числа легко определяется путем представления числа в виде развернутой записи и вычислением суммы его разрядов, умноженных на весовой коэффициент:

система счисления	свернутая запись	развернутая запись	значение
двоичная	1011_2	$1*2^3 + 0*2^2 + 1*2^1 + 1*2^0$	11_{10}
троичная	1011_3	$1*3^3 + 0*3^2 + 1*3^1 + 1*3^0$	31_{10}
пятеричная	1011_5	$1*5^3 + 0*5^2 + 1*5^1 + 1*5^0$	131_{10}
восьмеричная	1011_8	$1*10^3 + 0*10^2 + 1*10^1 + 1*10^0$	521_{10}
десятичная	1011_{10}	$1*10^3 + 0*10^2 + 1*10^1 + 1*10^0$	1011_{10}
шестнадцатеричная	1011_{16}	$1*10^3 + 0*10^2 + 1*10^1 + 1*10^0$	4113_{10}

Из таблицы видно разницу значений числа, представленного одинаково, но в различных системах счисления.

Чтобы перевести целое q -ичное число A в десятичную систему счисления, необходимо:

1. Выполнить последовательное деление числа A и получаемых целых частных на 10 до тех пор, пока не получим частное, меньшее делителя.

2. Сформировать новое число из остатков от деления, начиная с последней цифры.

$$\begin{array}{r} 237_{10} \Big| 8_{10} \\ -16 \quad 29 \quad \Big| 8_{10} \\ \hline 77 \quad -24 \quad 3 \end{array}$$

$$\begin{array}{r} 237_{10} \Big| 16_{10} \\ -16 \quad 14 \quad \Big| \\ \hline 77 \end{array}$$

$$\begin{array}{r} 237_{10} \Big| 2_{10} \\ -2 \quad 118 \quad \Big| 2_{10} \\ \hline 03 \quad -10 \quad 59 \quad \Big| 2_{10} \end{array}$$

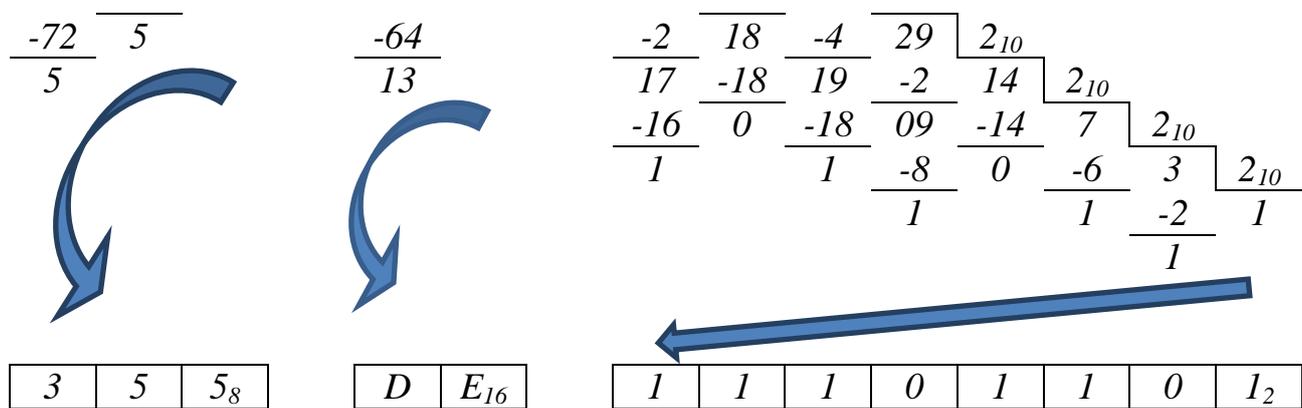


Рис. 19. Перевод числа 23710 из десятичной в восьмеричную, шестнадцатеричную и двоичную системы счисления

Для представления данных при их обработке на компьютере используются двоичное и шестнадцатеричное кодирование. Двоичные коды – это естественный машинный язык. Шестнадцатеричные коды применяются для сокращения записи значений двоичных машинных слов. Один байт можно представить, как 8 бит, а можно – как 2 шестнадцатеричных разряда по 4 бит.

Вопросы для самоконтроля:

1. Дайте определение понятию системы счисления.
2. Охарактеризуйте позиционную систему счисления.
3. Охарактеризуйте непозиционную систему счисления.
4. Что определяет основание системы счисления?
5. Какая самая простая система счисления?
6. Какая система счисления используется для хранения и обработки данных в ЭВМ?
7. По какому алгоритму выполняется перевод числа из одной системы счисления в другую?
8. По какому алгоритму выполняется обратное преобразование?
9. Каковы правила сложения двоичных чисел?
10. Каковы правила умножения двоичных чисел?

4. ОСНОВЫ АЛГЕБРЫ ЛОГИКИ

4.1. Понятие алгебры логики

В основе цифровой схемотехники лежит **алгебра двоичной логики**¹ – раздел математической логики, в котором изучаются операции с логическими высказываниями.

Логическое высказывание (суждение) – это повествовательное предложение, в котором что-то утверждается или отрицается.

*завтра пойдет дождь;
Москва – столица Российской Федерации;
все курсанты любят информатику;
некоторые курсанты имеют государственные награды;
этот курсант имеет звание генерал-полковник полиции.*

В алгебре логики конкретное содержание высказывания не имеет значения, важно, **истинное** оно или **ложное**.

Высказывание не должно быть противоречивым как, например, известный парадокс «я лжец»².

Любое высказывание состоит из связанных между собой понятий. В отличие от высказываний понятия не могут быть истинными или ложными.

Понятие – это мысленные образы объектов и их свойств, выраженные словами или словосочетаниями. Большинство понятий объединяет **множество объектов** с общими отличительными признаками. По аналогии с множеством в математике, любое понятие имеет количественную меру – **объем понятия**. Например, в понятие «президент США» входит свыше сорока объектов, а объем понятия «президент СССР» равен единице. В некоторых случаях объем понятия может быть неопределенным – «хороший следователь» или бесконечным – «натуральные числа».

Логические высказывания могут быть простыми и сложными. Сложные высказывания выражают логические отношения между простыми высказываниями. В алгебре логики простым высказываниям соответствуют логические переменные, обозначаемые латинскими буквами А, В, С и т.п. Сложным высказываниям соответствуют логические функции F(A, B, C) и т.п. Логические переменные и логические функции принимают значение И(стина) или 1, если высказывание истинно, и Л(ожь) или 0, если ложно.

В структуре простого высказывания выделяют четыре основные части:

1. **Субъект (S)** – основное понятие высказывания.

Все курсанты Бел ЮИ МВД России имени И.Д. Путилина (S) являются сотрудниками полиции.

¹ Далее – алгебра логики.

² Если данное высказывание истинно, то говорящий не лжец, а он утверждает обратное, и наоборот.

2. **Предикат (P)** – понятие, связанное с субъектом каким-либо отношением.

Все учебники (S) являются книгами (P).

3. **Смысловая связка** – слово, соединяющее субъект и предикат.

Связка может быть положительная и отрицательная. В роли связки могут быть слова есть, нет, является, не является, может, не может, будет, не будет и т.п.

4. **Квантор** – указатель на объем понятия-субъекта.

В логическом высказывании квантор определяет границы истинности понятия:

\forall – «любой...», «каждый...», «все...» или «для любого...», «для каждого...», «для всех...» и т.п.;

\exists – «существует...», «найдётся...» или «для некоторых...» и т.п.

В контексте некоторых высказываний субъект, предикат, связка или квантор могут явно отсутствовать, но они подразумеваются.

(все) Кошки (S) – (есть) опасные хищники (P).

Сегодня (текущий день – S) (есть) солнечно (солнечный день – P).

Между субъектом S и предикатом P возможны следующие отношения:

S и P не имеют общих объектов \Rightarrow понятия **несовместимы**.

(все) Параллельные прямые (S) (есть) не пересекаются (не пересекающиеся геометрические объекты – P).

S и P состоят из одинаковых объектов \Rightarrow понятия **равнозначны**.

Информация (S) – это сведения о чем-либо (P).

Юрий Долгорукий – основатель Москвы.

Объекты S частично совпадают с объектами P \Rightarrow понятия **пересекаются**.

Грибы (S) бывают несъедобные (продукты – P).

В лесной фауне Подмосковья (S) встречаются ежи (P).

Все объекты S являются частью P \Rightarrow понятие S в **подчинении** P.

Любая окружность (S) – это множество точек (P).

Все объекты P являются частью S \Rightarrow понятие P в **подчинении** S.

Среди людей (S) встречаются гении (P).

Если у совместимых понятий объем одного понятия в высказывании меньше объема другого, то первое понятие называется **видовым**, а второе – **родовым**.

Отношения между понятиями в простом высказывании наглядно демонстрируются с помощью круговых диаграмм Эйлера¹, где каждое понятие (множество однородных объектов) изображается кругом. Взаимное расположение кругов отражает отношение между понятиями.

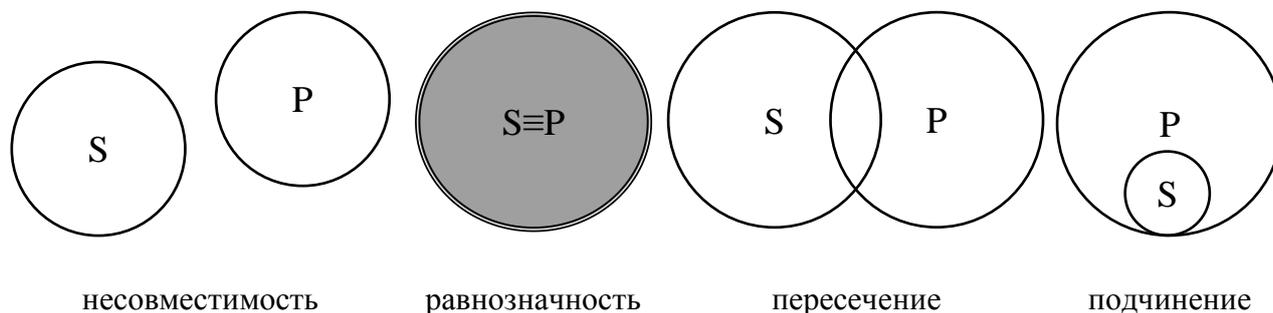


Рис. 20. Варианты отношений между субъектом и предикатом

В простом высказывании объем субъекта может быть общим (все) и частным (часть), а связка может быть утвердительной или отрицательной, что порождает четыре комбинации.

Тип А. Общеутвердительные высказывания имеют общий объем субъекта и утвердительную связку. Субъект и предикат в таком высказывании могут находиться только в отношении равнозначности или в отношении подчинения.

Все S есть P.

Или, если S и P рассматривать как множества с элементами s и p:

$\forall s \in P$ или $\forall p \in S$.

Все звезды являются небесным телами.

Информация – это сведения.

Москва основана в 1147 г.

Тип Е. Общеотрицательные высказывания имеют общий объем субъекта и отрицательную связку. Субъект и предикат в таком высказывании могут находиться только в отношении несовместимости.

Все S не есть P (ни одно S не является P).

$\forall s \notin P$ или $\nexists s \in P$.

(Ни один) Российский полицейский не может подрабатывать в коммерческой фирме.

Высказывания, в которых объем субъекта равен единице, также считаются общими, так как речь идет о понятии в полном объеме.

Дмитрий Донской не является основателем Москвы.

¹ Иногда их называют «диаграммы Эйлера-Венна».

Тип I. Частноутвердительные высказывания имеют частный объем субъекта и утвердительную связку. Субъект и предикат в таком высказывании могут находиться только в отношении пересечения или в отношении подчинения.

Некоторые S есть P.

$\exists s \in P.$

Некоторые слушатели Московского университета МВД России – иностранные граждане.

Среди деревьев в лесу встречаются сосны. (Некоторые деревья в лесу – это сосны).

Тип O. Частноотрицательные высказывания имеют частный объем субъекта и отрицательную связку. Субъект и предикат в таком высказывании могут находиться только в отношении пересечения или в отношении подчинения.

Некоторые S не есть P.

$\exists s \notin P.$

Некоторые курсанты не проживают в общежитии.

Некоторые пешеходы никогда не нарушают ПДД.

Важно помнить, что в алгебре логики нас интересует не содержание высказывания, а истинно оно или ложно. Любое сложное высказывание является истинным или ложным в зависимости от истинности или ложности входящих в него простых суждений.

4.2. Типы операций в алгебре логики

При исследовании сложных высказываний обычно используют следующие **типы операций с простыми высказываниями**:

Отрицание простого высказывания: \bar{A} , $\neg A$.

Если $A = И(стина)$, то $\bar{A} = Л(ожь)$, и наоборот.

$A = \langle \text{«Ежи – это птицы»} \rangle;$

$\bar{A} = \langle \text{«Неверно, что ежи – это птицы»} \rangle = \langle \text{«Ежи – это не птицы»} \rangle.$

Конъюнкция (логическое И): $A \wedge B$, $A \& B$, $A \cdot B$

$A \wedge B = И$, если $A = B = И$ одновременно.

$A = \langle \text{«У Васи рыжие волосы»} \rangle;$

$B = \langle \text{«Осенью часто идут дожди»} \rangle;$

$A \wedge B = \langle \text{«У Васи рыжие волосы»} \wedge \langle \text{«Осенью часто идут дожди»} \rangle.$

Нестрогая дизъюнкция (логическое ИЛИ): $A \vee B$; $A + B$.

$A \vee B = И$, если $A = И$ или $B = И$ или $A = B = И$ одновременно.

$A = \langle \text{«Курсант может быть отчислен за академическую неуспеваемость»} \rangle;$

$B = \langle \text{«Курсант может быть отчислен за нарушение дисциплины»} \rangle;$

$A \vee B =$ «Курсант может быть отчислен за академическую неуспеваемость или за нарушение дисциплины».

Строгая дизъюнкция (исключающее ИЛИ): $A \oplus B, A \underline{\vee} B.$

$A \oplus B = И,$ если или $A = И$ или $B = И,$ но не одновременно.

$A =$ «Победителем турнира может быть Петя»;

$B =$ «Победителем турнира может быть Вова»;

$A \oplus B =$ «Победителем турнира может быть или Петя, или Вова».

Импликация (следование): $A \Rightarrow B, A \rightarrow B.$

$A \Rightarrow B = И,$ если $A = B = И$ или $A = Л$ при любом $B.$

$A =$ «Железо является металлом»;

$B =$ «Все металлы являются проводниками электрического тока»;

$A \Rightarrow B =$ «Если железо является металлом, то оно обязательно является проводником электрического тока».

Эквиваленция (равносильность): $A \Leftrightarrow B, A \leftrightarrow B, A \equiv B, \overline{A \oplus B}.$

$A \Leftrightarrow B = И$ тогда и только тогда, когда $A = B$ или, наоборот, $B = A.$

$A =$ «Переменная S – четная»;

$B =$ «Переменная S делится без остатка на 2»;

$A \Leftrightarrow B =$ «Если переменная S четная, то она делится без остатка на 2».

И при импликации, и при эквиваленции сложное высказывание часто строится с помощью составного союза «если ..., то...», и из утверждения A всегда следует утверждение $B.$ Разница в том, что при эквиваленции из B также всегда следует $A,$ а при импликации – нет.

Рассмотренные варианты поведения функции, соответствующей сложному логическому высказыванию, можно записать в таблице истинности или изобразить в виде диаграмм Эйлера:

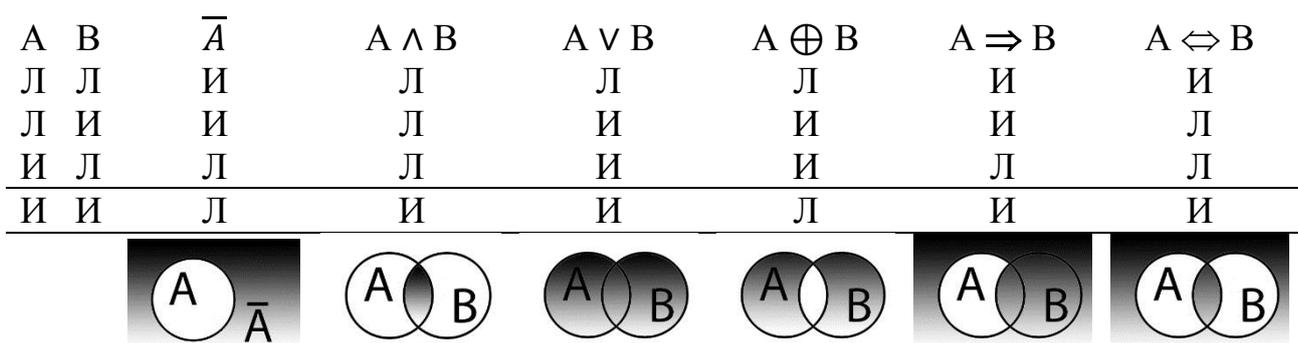


Рис. 21. Таблицы истинности и диаграммы Эйлера

Принцип построения диаграммы Эйлера для сложных высказываний рассмотрим на функции эквиваленции. Из таблицы истинности видно, что функция $A \Leftrightarrow B = И$ в двух случаях, когда $(A = Л$ и $B = Л)$ или $(A = И$ и $B = И).$

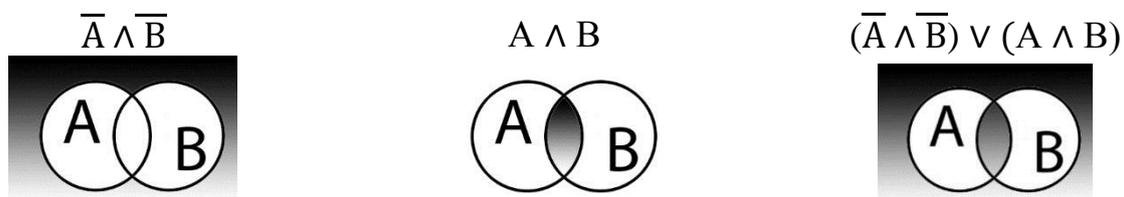


Рис. 22. Построение диаграммы Эйлера для функции эквиваленции

Из рисунка видно, что эквиваленцию можно выразить через дизъюнкцию, конъюнкцию и отрицание: $A \leftrightarrow B = (\bar{A} \wedge \bar{B}) \vee (A \wedge B)$.

Законы логики высказываний, принимающих два возможных состояния – истину и ложь, легли в основу цифровой схемотехники, которая также оперирует бинарными сигналами и состояниями устройств $\{0; 1\}$.

Ниже в таблице перечислены основные функции математической логики, которую иногда называют булева¹ алгебра. В зависимости от контекста и области применения у функций могут встречаться другие названия и обозначения.

Обозначение	Название	A	0	0	1	1
		B	0	1	0	1
$F(A) = F(B) = 0$	константа 0		0	0	0	0
$F(A, B) = A \cdot B$	конъюнкция		0	0	0	1
$F(A, B) = \overline{A \rightarrow B}$	инверсия импликации $A \rightarrow B$		0	0	1	0
$F(A, B) = A$	переменная A		0	0	1	1
$F(A, B) = \overline{B \rightarrow A}$	инверсия импликации $B \rightarrow A$		0	1	0	0
$F(A, B) = B$	переменная B		0	1	0	1
$F(A, B) = A \oplus B$	исключающее ИЛИ		0	1	1	0
$F(A, B) = A \vee B$	дизъюнкция		0	1	1	1
$F(A, B) = A \downarrow B$	стрелка Пирса		1	0	0	0
$F(A, B) = A \equiv B$	равносильность		1	0	0	1
$F(A, B) = \bar{B}$	инверсия переменной B		1	0	1	0
$F(A, B) = B \rightarrow A$	импликация $B \rightarrow A$		1	0	1	1
$F(A, B) = \bar{A}$	инверсия переменной A		1	1	0	0
$F(A, B) = A \rightarrow B$	импликация $A \rightarrow B$		1	1	0	1
$F(A, B) = A B$	штрих Шеффера		1	1	1	0
$F(A) = F(B) = 1$	константа 1, тавтология		1	1	1	1

Две булевы функции тождественны друг другу, если на любых одинаковых наборах аргументов они принимают равные значения. Тождества алгебры логики называются равносильностями, правилами, законами, теоремами, но суть от этого не меняется:

Коммутативность	$A \vee B = B \vee A;$
-----------------	------------------------

¹ Джордж Буль – английский математик, в XIX в. внес большой вклад в создание математической логики.

	$A \cdot B = B \cdot A;$ $A \oplus B = B \oplus A;$ $A \equiv B = B \equiv A.$
Ассоциативность	$A \vee B \vee C = (A \vee B) \vee C = A \vee (B \vee C);$ $A \cdot B \cdot C = (A \cdot B) \cdot C = A \cdot (B \cdot C);$ $A \oplus B \oplus C = (A \oplus B) \oplus C = A \oplus (B \oplus C).$
Дистрибутивность	$(A \vee B) \cdot C = A \cdot C \vee B \cdot C;$ $A \cdot B \vee C = (A \vee C) \cdot (B \vee C);$ $(A \oplus B) \cdot C = A \cdot C \oplus B \cdot C.$
Идемпотентность	$A \vee A = A \quad A \cdot A = A.$
Правила де Моргана	$\overline{A \cdot B} = \overline{A} \vee \overline{B};$ $\overline{A \vee B} = \overline{A} \cdot \overline{B}.$
Операции с константами {0; 1}	$A \cdot 0 = 0;$ $0 \vee A = 1 \quad A = A;$ $1 \vee A = 1.$
Инверсия	$A \vee \overline{A} = 1;$ $A \cdot \overline{A} = 0;$ $\overline{\overline{A}} = A.$
Поглощение	$A \vee A \cdot B = A \cdot 1 \vee A \cdot B = A (1 \vee B) = A;$ $A (A \vee B) = A \cdot A \vee A \cdot B = A \vee A \cdot B = A.$
Склеивание	$A \cdot B \vee A \cdot \overline{B} = A (B \vee \overline{B}) = A;$ $(A \vee B) (A \vee \overline{B}) = A (B \vee \overline{B}) = A;$ $A (\overline{A} \vee B) = A \cdot \overline{A} \vee A \cdot B = A \cdot B;$ $A \vee \overline{A} \cdot B = (A \vee \overline{A}) (A \vee B) = A \vee B.$
Приведение к базису $\{\neg; \vee; \wedge\}$ ¹	$A \mid B = \overline{A} \cdot B;$ $A \downarrow B = \overline{A \vee B};$ $A \rightarrow B = \overline{A} \cdot B \vee A \cdot B = \overline{A} \vee B;$ $A \oplus B = \overline{A} \cdot B \vee A \cdot \overline{B};$ $A \equiv B = \overline{A \oplus B} = (A \rightarrow B) (B \rightarrow A)$

Вычисление значений сложных логических выражений выполняется в порядке согласно их приоритету:

1. Выражение в скобках.
2. Инверсия.
3. Конъюнкция.
4. Дизъюнкция.
5. Импликация, эквивалентность и др.

Существуют разные методы записи условия и решения логических задач.

Одним из способов задания логической функции любой сложности является таблица истинности.

¹ Тожества выводятся по таблице истинности.

Разобьём формулу из предыдущего примера на элементарные операции и для результата каждой операции создадим свой столбец в таблице истинности в порядке очередности исполнения.

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

Из таблицы истинности видно, что результат тот же – функция истинна в двух случаях: когда все исходные переменные или истины либо ложны одновременно.

На практике не менее актуальна задача составления логической формулы по таблице истинности, например, какие сигналы должны быть на выходе логической схемы при заданных входных.

A	B	C	F	минтермы	макстермы
0	0	0	1	$\bar{A}\bar{B}\bar{C}$	
0	0	1	1	$\bar{A}\bar{B}C$	
0	1	0	0		$A \vee \bar{B} \vee C$
0	1	1	0		$A \vee \bar{B} \vee \bar{C}$
1	0	0	0		$\bar{A} \vee B \vee C$
1	0	1	1	$A\bar{B}C$	
1	1	0	0		$\bar{A} \vee \bar{B} \vee C$
1	1	1	1	$A B C$	

Из таблицы истинности выбираем строки, где значение функции принимает значение 1. Из комбинации входных переменных или их отрицаний сформируем элементарные конъюнкции (минтермы) и сложим их логически (дизъюнкция):

$$F = \bar{A}\bar{B}\bar{C} \vee \bar{A}\bar{B}C \vee A\bar{B}C \vee ABC.$$

Выражение логической функции в виде дизъюнкции элементарных конъюнкций входных переменных называется **дизъюнктивной нормальной формой** или **ДНФ**.

4.3. Диаграммы Эйлера для теоретико-множественных операций

Если рассматривать некоторые понятия, используемые в логических высказываниях как дискретные множества объектов с общими характерными признаками, то с ними можно осуществлять операции дискретной математики.

Множества могут быть конечными и бесконечными.

Объекты, объединенные общими признаками, называют элементами множества и обычно обозначают строчными латинскими буквами, иногда с индексами: a_i, b_i, \dots, z_i . Тогда соответствующие им множества обозначают прописными латинскими буквами: A, B, \dots, Z .

Объединяющее множество всех элементов, которые могут встретиться в рассматриваемой области, называют универсальным и обозначают как U . Пустое множество обозначают как V .

Запись $a \in A$ означает, что элемент «а» принадлежит множеству A , $b \notin A$ означает, что элемент «b» не принадлежит множеству A .

Множество можно определить перечислением его элементов:

$$C = \{c_1, c_2, c_3, c_4, c_5\}$$

или, если множество большое, с помощью подходящих предикатов:

$$C = \{c:F(x)\},$$

например, $C = \{c : c \in \mathbb{N} \text{ и } c = 2n, \text{ где } n = 1, 2, 3, \dots\}$ – множество четных натуральных чисел.

Некоторые множества имеют стандартные обозначения и область применения:

$\mathbb{N} = \{1, 2, 3, \dots\}$ – множество натуральных чисел;

$\mathbb{Z} = \{\frac{p}{q} : p, q \in \mathbb{Z}, q \neq 0\}$ – множество целых чисел;

$\mathbb{Q} = \{1, 2, 3, \dots\}$ – множество рациональных чисел;

$\mathbb{R} = \{\text{все десятичные дроби}\}$ – множество вещественных чисел и т.п.

Множество натуральных чисел исторически используется для дискретного счета целого количества единиц измерения, множество целых чисел – для представления чисел и операций с ними в позиционных системах счисления, рациональных – для операций с частями целого. Вещественные числа предназначены для измерения непрерывных величин.

Множество A является **подмножеством** B ($A \subset B$), если $\forall a_i \in B$.

Если множества состоят из одинаковых элементов, то они называются **равными** ($A = B$).

Объединением множеств A и B называется множество C , элементы которого принадлежат A или B :

$$C = A \cup B = \{c_i : c_i \in A \text{ или } c_i \in B\}.$$

Пересечением множеств A и B называется множество C , элементы которого принадлежат и A и B :

$$C = A \cap B = \{c_i : c_i \in A \text{ и } c_i \in B\}.$$

Дополнением множества A до U называется множество C (\bar{A}), элементы которого не принадлежат A , но принадлежат универсальному множеству U :

$$C = \bar{A} = \{c_i : c_i \notin A \text{ и } c_i \in U\}.$$

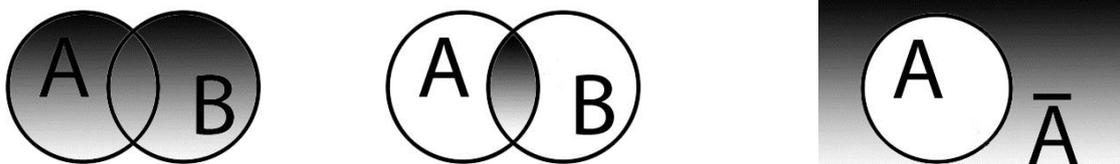


Рис. 23. Диаграммы Эйлера для объединения, пересечения и дополнения множеств

По аналогии с основными тождествами алгебры логики операции над множествами обладают такими же свойствами.

Коммутативность	$A \cup B = B \cup A$ $A \cap B = B \cap A$
Ассоциативность	$(A \cup B) \cup C = A \cup (B \cup C)$ $(A \cap B) \cap C = A \cap (B \cap C)$
Дистрибутивность	$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$ $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$
Идемпотентность	$A \cup A = A$ $A \cap A = A$
Правила де Моргана	$\overline{A \cap B} = \bar{A} \cup \bar{B};$ $\overline{A \cup B} = \bar{A} \cap \bar{B}$

В дополнение к основным операциям над множествами в алгоритмах компьютерной обработки данных используются также разность, симметрическая разность и декартово произведение множеств.

Разностью множеств A и B называется множество C , элементы которого принадлежат A и не принадлежат B :

$$C = A \setminus B = \{c_i : c_i \in A \text{ и } c_i \notin B\},$$

$$C = A \setminus B = A \cap \bar{B}.$$

Симметрической разностью множеств A и B называется множество C , элементы которого принадлежат A и не принадлежат B или принадлежат B и не принадлежат A :

$$C = A \Delta B = (A \setminus B) \cup (B \setminus A) = \{c_i : c_i \in A \text{ и } c_i \notin B \text{ или } c_i \in B \text{ и } c_i \notin A\}.$$

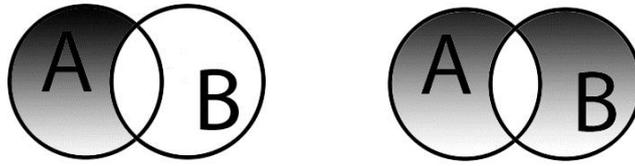


Рис. 24. Диаграммы Эйлера для разности и симметрической разности множеств

Докажем тождество:

$$A \Delta B = (A \cup B) \cap \overline{A \cap B}.$$

$$\begin{aligned} A \Delta B &= (A \cup B) \cap (\overline{A \cap B}) = ((A \cup B) \cap \overline{A}) \cup ((A \cup B) \cap \overline{B}) = \\ &= ((A \cap \overline{A}) \cup (B \cap \overline{A})) \cup ((A \cap \overline{B}) \cup (B \cap \overline{B})) = (A \cap \overline{B}) \cup (B \cap \overline{A}) = \\ &= (A \setminus B) \cup (B \setminus A). \end{aligned}$$

Мощностью конечного множества $|C|$ называется величина равная количеству его элементов.

$ A \cup B = A + B - A \cap B $

Решим задачу.

На поисковые запросы Google выдал следующие результаты:

«Бел ЮИ»	1 500 тыс. страниц
«МВД России»	8 000 тыс. страниц
«Бел ЮИ»&«МВД России»	500 тыс. страниц

Сколько страниц содержат хотя бы одну из этих фраз?

$$8\,000 + 1\,500 - 500 = 9\,000$$

Группа состоит из 25 курсантов. В октябре 16 человек должны заступить в суточный наряд на КПП и по курсу и 12 курсантов – в наряд по столовой. Сколько курсантов будут зачислены в оба наряда, если известно, что 3 человека в октябре освобождены от нарядов?

$$\begin{aligned} 25 - 3 &= 16 + 12 - x \\ x &= 28 - 22 = 6 \end{aligned}$$

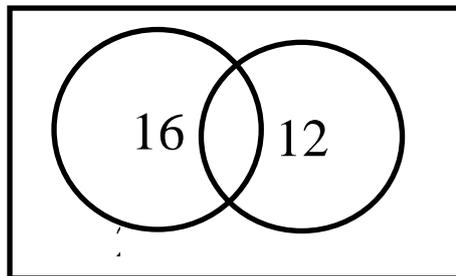


Рис. 25. Диаграмма Эйлера для анализа множества курсантов, заступающих в наряды

Декартовым или прямым произведением множества A на множество B называют множество C всех упорядоченных пар (a_i, b_j) :

$$C = A \times B = \{(a_i, b_j) : a_i \in A \text{ и } b_j \in B\}.$$

Например, $A = \{0, 1, 2\}$, $B = \{3, 4\}$.

$$A \times B = \{(0, 3), (0, 4), (1, 3), (1, 4), (2, 3), (2, 4)\}.$$

Декартово произведение трех множеств X, Y, Z будет состоять из упорядоченных троек элементов:

$$X \times Y \times Z = \{(x, y, z) : x \in X, y \in Y, z \in Z\}.$$

Декартово произведение n множеств будет состоять из упорядоченных n -ок (произносится «энок») элементов, которые называются **кортежами**.

$$A_1 \times A_2 \times \dots \times A_n = \{(a_1, a_2, \dots, a_n) : a_i \in A_i, i = 1, 2, \dots, n\}.$$

Декартово произведение множества самого на себя типа

$A \times A, A \times A \times A, A \times A \times A \times A$ и т.д.

принято записывать в виде степени: A^2, A^3, A^4 и т.д. соответственно.

Примером использования упорядоченных пар на множестве вещественных чисел \mathbb{R} является декартова система координат, где пара из абсциссы и ординаты однозначно определяет положение точки на плоскости.

С помощью декартова произведения определяется понятие отношения, используемого при проектировании и эксплуатации реляционных баз данных.

Вопросы для самоконтроля:

1. Дайте определение понятия логического высказывания (суждения).
2. Охарактеризуйте операцию конъюнкция.
3. Охарактеризуйте операцию нестрогая дизъюнкция.
4. Охарактеризуйте операцию строгая дизъюнкция.
5. Охарактеризуйте операцию отрицание.
6. Охарактеризуйте операцию импликация.
7. Охарактеризуйте операцию эквиваленция.
8. Укажите приоритеты логических операций.
9. Укажите основные способы задания логической функции.
10. Охарактеризуйте взаимодействия множеств.
11. Как используют диаграммы Эйлера для теоретико-множественных операций?

5. ЭВОЛЮЦИЯ ЭЛЕКТРОННЫХ ВЫЧИСЛИТЕЛЬНЫХ МАШИН

5.1. История машинных вычислений, компьютерная архитектура

Вообще, архитектура (др.-греч. *αρχι* – главный, *τέκτων* – строитель) – это базовая организация системы, воплощенная в ее компонентах, их отношениях между собой и с окружением, а также принципы, определяющие проектирование и развитие системы (IEEE 1471).

Компьютерная архитектура (англ. **Computer architecture**) – совокупность средств и правил, определяющих принципиальную организацию компьютерной системы, обеспечивающих взаимодействие устройств и (или) программ.

С точки зрения пользователя, **компьютерная архитектура** – это совокупность основных технических решений, использованных при разработке компьютера: принципы взаимодействия устройств, интерфейсы, система команд, организация памяти, алгоритмы обработки данных т.п.

Любая компьютерная архитектура объединяет:

- hardware – аппаратное обеспечение;
- software – программное обеспечение;
- brain ware – алгоритмическое обеспечение (алгоритмы и методы решения задач).

Компьютеры, имеющие одинаковую архитектуру, совместимы друг с другом, как, например, персональные компьютеры, имеющие архитектуру IBM PC.

Принято считать, что первую **счетную машину** сконструировал в 1642 году французский ученый Блез Паскаль. Это была механическая конструкция с шестеренками и ручным приводом.



Рис. 26. Паскалина

Паскалина позволяла складывать и вычитать десятичные числа, автоматически учитывая единицу переноса. Первые варианты «Паскалины» имели пять разрядов, позднее их число увеличилось до восьми, что позволяло находить сумму до 99999999.

Тридцать лет спустя немецкий математик Готфрид Вильгельм Лейбниц построил механический вычислитель, который, помимо сложения и вычитания, выполнял умножение и деление.



Рис. 27. Счетная машина Лейбница

Профессор математики Кембриджского университета Чарльз Бэббидж в 1822 году построил вычислительное устройство, способное производить арифметические операции с точностью до шестого знака после запятой. Первая разностная машина Бэббиджа рассчитывала значения логарифмов для таблиц морской навигации штамповала результаты на металлическую пластину.

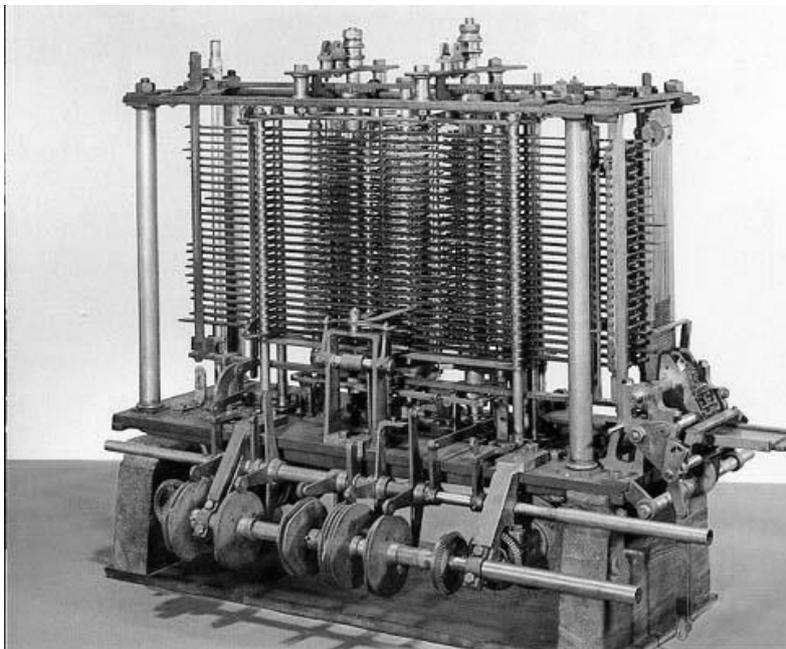


Рис. 28. Разностная машина Бэббиджа

Совершенствуя свое изобретение, Бэббидж разработал проект универсальной аналитической вычислительной машины, по структуре похожей на цифровой компьютер. Аналитическая машина Бэббиджа состояла из арифметического устройства («мельница»), 1000 десятичных 50-разрядных слов-регистров памяти («склад»), устройства ввода данных с перфокарт и устройства

вывода (перфоратор и печатающее устройство). Перфокарты операций переключали машину между режимами сложения, вычитания, деления и умножения. Перфокарты переменных управляли передачей данных из памяти в арифметическое устройство и обратно. Числовые перфокарты могли быть использованы как для ввода данных в машину, так и для сохранения результатов вычислений, если памяти было недостаточно. Определённый набор перфокарт с данными и операциями фактически являлся программой вычислений.

Разработку программного обеспечения для будущей аналитической машины Бэббидж возложил на свою знакомую Аду Августу Лавлейс, дочь британского поэта лорда Байрона.

В конце 30-х годов прошлого века немец Конрад Зус сконструировал несколько автоматических счетных машин с использованием электромагнитных реле, но все они были уничтожены во время бомбежек Берлина.

Одновременно в Америке Джон Атанасов спроектировал вычислительную машину с двоичной арифметикой.

Говард Айкен, познакомившись в библиотеке Гарварда с трудами Бэббиджа, в 1944 году при финансовой поддержке компании IBM создает компьютер на базе электромеханических реле. «Mark I» имел 72 слова по 23 десятичных разряда каждое и выполнял любую операцию за 6 секунд. Ввод-вывод данных осуществлялся с перфоленты.

5.2. Первое поколение ЭВМ – на электронных лампах (1945-1955)

Стимулом, ускорившим создание электронных компьютеров, стала Вторая мировая война. В начале войны английской разведке удалось достать немецкую шифровальную машину ENIGMA. Чтобы расшифровать перехваченную радиограмму, требовалось в кратчайшие сроки произвести огромное количество вычислений. В 1943 году в секретной лаборатории при участии Алана Тьюринга был создан первый в мире электронный цифровой компьютер COLOSSUS. Но, поскольку британское правительство полностью засекретило этот проект на 30 лет, архитектура COLOSSUS не повлияла на дальнейшее развитие вычислительной техники.

В США Джон Моушли, знакомый с работами Атанасова, и его студент Дж. Преспер Экерт в 1946 г. сконструировали электронный компьютер, который они назвали ENIAC (Electronic Numerical Integrator and Computer¹). ENIAC состоял из 18 000 электровакуумных ламп и 1500 реле, весил 30 тонн и потреблял 140 киловатт электроэнергии. У машины было 20 регистров (ячеек памяти), каждый из которых мог содержать 10-разрядное десятичное число.

В 1946 году один из участников проекта ENIAC Джон фон Нейман вместе с Артуром Бёрксом и Германом Голдстейном опубликовали статью «Предварительное рассмотрение логического конструирования электронного вычисли-

¹ Электронный цифровой интегратор и вычислитель.

тельного устройства», где сформулировали основные принципы конструирования цифровых ЭВМ, актуальные до настоящего времени.



Рис. 29. Вакуумные электронные лампы

Принципы фон Неймана:

1. Основными блоками ЭВМ являются:
 - арифметико-логическое устройство (АЛУ);
 - устройство управления;
 - запоминающее устройство;
 - устройства ввода-вывода.

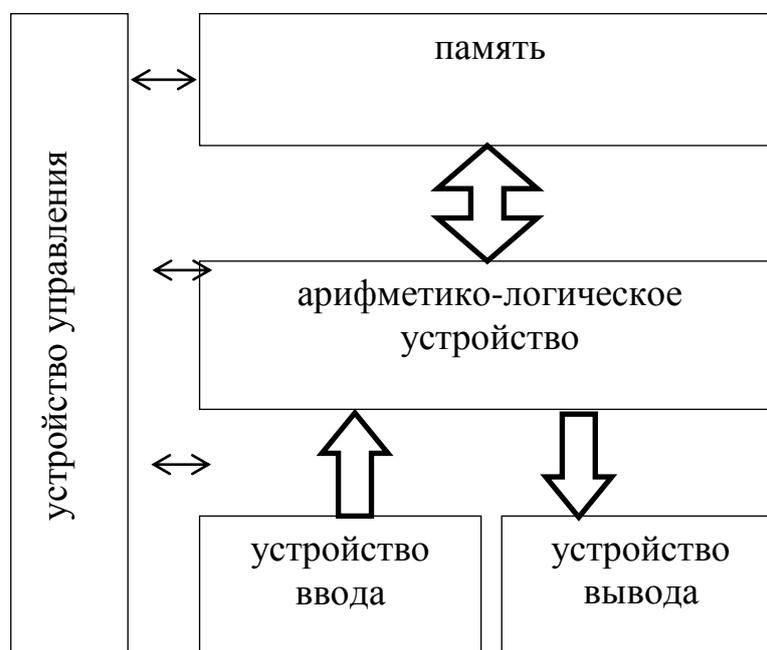


Рис. 30. Структурная схема ЭВМ Джона фон Неймана

2. Вся информация, поступающая в ЭВМ, кодируется с помощью двоичных сигналов (битов) и группируется в более крупные адресуемые единицы, называемые словами.

3. Память разбита на ячейки с пронумерованными адресами, по которым процессором осуществляются операции чтения/записи слов данных.

4. Программа состоит из последовательного набора команд, которые считываются из памяти и выполняются процессором автоматически друг за другом.

5. Программы и данные хранятся в одной и той же памяти. Что записывать и как интерпретировать содержимое ячейки памяти, зависит от конкретной программы.

Когда Экерт и Моушли приступили к разработке очередной ЭВМ EDVAC (Electronic Discrete Variable Computer¹), Джон фон Нейман в Принстонском Институте специальных исследований начал разрабатывать собственную версию EDVAC под названием IAS (Immediate Address Storage²). Объем памяти машины фон Неймана составлял 4096 40-битных слов. Каждое слово данных представляло или 2 команды по 20 бит, или целое число со знаком.

Примерно в то же время, когда Фон Нейман работал над машиной IAS, исследователи МТИ разрабатывали свой компьютер Whirl wind I. В отличие от IAS, ENIAC и других машин того же типа со словами большой длины, машина Whirl wind I имела слова по 16 бит и предназначалась для работы в реальном времени. Этот проект привел к изобретению Джейм Форрестером памяти на магнитном сердечнике, а затем и первого серийного мини-компьютера.

Известная фирма IBM только в 1952 году сконструировала свой первый компьютер IBM 701, в который входили отдельными устройствами:

IBM 701 – центральный процессор;

IBM 706 – электростатическая память на лампах (2048 слов по 36 бит);

IBM 711 – устройство ввода с перфокарт (150 карт/мин.);

IBM 716 – принтер (150 строк/мин.);

IBM 721 – устройство вывода на перфокарты (100 карт/мин.);

IBM 726 – устройство чтения/записи на магнитную ленту (100 бит/дюйм)

IBM 731 – устройство чтения/записи на магнитный барабан и др.

701 стал первым коммерческим компьютером, лидировавшим на рынке в течение десяти лет.

Через три года появился компьютер 704, у которого было 4 Кбайт памяти на магнитных сердечниках, команды по 36 бит и процессор с плавающей точкой. В 1958 году компания IBM начала работу над последним компьютером на электронных лампах, 709, который, по сути, представлял собой усложненную версию 704.

¹ Электронная дискретная параметрическая машина.

² Память с прямой адресацией.

5.3. Второе поколение ЭВМ – на транзисторах (1955-1965)

Полупроводниковый транзистор был изобретен сотрудниками лаборатории Bell Laboratories Джоном Бардином, Уолтером Браттейном и Уильямом Шокли, за что в 1956 году они получили Нобелевскую премию в области физики. В течение нескольких лет транзисторы¹ совершили революцию в компьютерной схемотехнике.

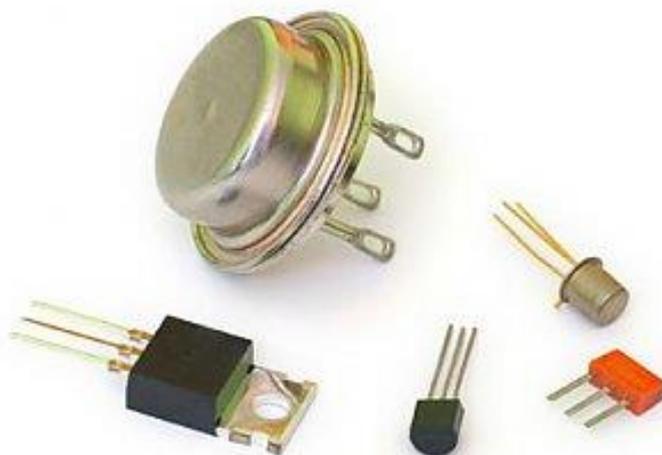


Рис. 30. Транзисторы

Первые 16-битные компьютеры на транзисторных переключателях TX-0, TX-2 (Transistorized experimental computer) были разработаны в лаборатории Массачусетского технического института (МТИ). Чтобы организовать серийное производство транзисторных ЭВМ, один из разработчиков (Кеннет Ольсен) в 1957 году основал компанию DEC (Digital Equipment Corporation). В 1961 году появился компьютер PDP-1. Он имел память на 4096 18-битных слов и быстродействие – 200 000 команд в секунду. Важным нововведением в PDP-1 был дисплей размером 512x512 пикселей, на котором можно было рисовать точки. Вскоре студенты МТИ разработали для PDP-1 первую в мире компьютерную игру «Война миров».

Через несколько лет компания DEC разработала мини-ЭВМ PDP-8 – 12-разрядный компьютер, который стоил гораздо дешевле, чем PDP-1 (16 000 \$ vs. 120 000 \$). В PDP-8 впервые осуществлялся обмен данными через **общую шину**, или магистраль – набор сигнальных проводов, для параллельного подключения устройств по единой спецификации². Использование общей шины позволило конструировать ЭВМ по модульному принципу – из унифицированных взаимозаменяемых устройств, подключаемых к одинаковым разъемам.

¹ Электронные переключатели напряжения, соответствующего логическому 0, в напряжение логической 1, и наоборот.

² Описание требований обмена данными и параметров сигналов.

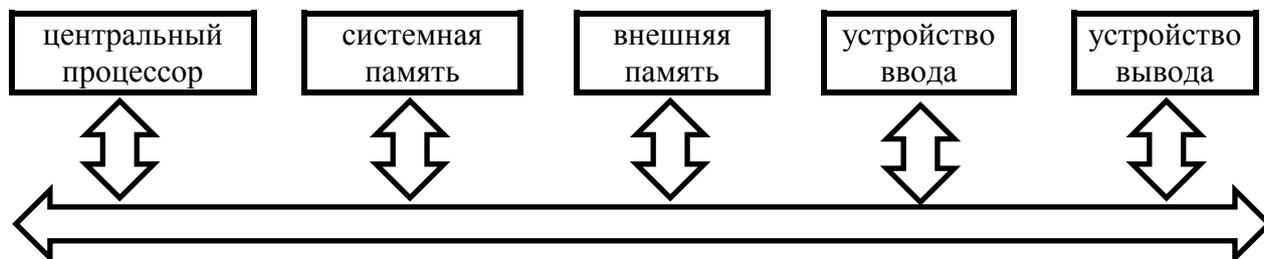


Рис. 32. Схема подключения устройств к интерфейсу «Общая шина»

Как и процессор, шина включала адресные, информационные и управляющие сигналы. Ячейки памяти и все устройства, подключенные к шине, имели уникальный адрес, и их количество ограничивалось разрядностью шины адреса. В каждый момент времени два устройства могли обмениваться информацией, количество которой определялось разрядностью шины данных. И такой топологический подход с тех пор стал использоваться практически во всех компьютерах.

Архитектура современных компьютеров также использует шинные интерфейсы на разных системных уровнях: системная шина, USB (Universal Serial Bus¹), PCI (Peripheral Component Interconnect bus²) и др.

В 1964 году компания CDC (Control Data Corporation) для проведения сложных расчетов выпустила машину 6600 с повышенным быстродействием. В центральном процессоре впервые использовали параллельные вычисления. Несколько встроенных функциональных устройств для сложения, умножения, деления, операций ввода-вывода могли работать одновременно. В некоторых программах машина выполняла 10 команд одновременно.

Основной идеолог архитектуры компьютера CDC 6600 Сеймур Крей посвятил жизнь созданию суперкомпьютеров, среди которых выделяется серия Cray.

Во втором поколении следует упомянуть еще один компьютер – Burroughs B5000. Разработчики компаний DEC, IBM, CDC совершенствовали свои модели за счет аппаратного обеспечения, стараясь снизить его стоимость (DEC) или заставить работать быстрее (IBM и CDC). Производители B5000 пошли другим путем. Они разработали машину с намерением программировать ее на языке высокого уровня Algol 60 (предшественнике языков C и Java), сконструировав аппаратное обеспечение так, чтобы упростить задачу компилятора.

¹ Универсальная последовательная шина.

² Шина подключения периферийных устройств.

5.4. Третье поколение ЭВМ – на интегральных схемах (1965-1980)

Изобретение в 1958 году Робертом Нойсом кремниевой интегральной схемы позволило размещать в одном небольшом устройстве десятки транзисторов, что уменьшило размеры и стоимость компьютеров, повысило скорость их работы и дало толчок новым архитектурным особенностям.

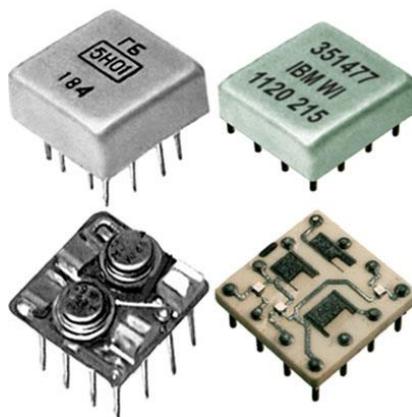


Рис. 33. Простейшие интегральные схемы

Компания IBM 7 апреля 1964 года анонсировала линейку компьютеров System/360, которые были предназначены как для научных, так и для коммерческих расчетов. Это было семейство компьютеров¹ с единым машинным языком и своим языком программирования низкого уровня – ассемблером.

Каждая новая модель семейства превосходила по ряду параметров предыдущую, но программы, написанные для одной из них, могли использоваться в другой.

IBM/360 была первой серийной 32-разрядной ЭВМ.

IBM/360 могла работать в многозадачном режиме, выполняя несколько программ одновременно. Например, когда одна программа ожидала завершения ввода-вывода, другая выполнялась процессором.

В IBM/360 впервые были использованы встроенные микропрограммы для реализации отдельных команд процессора.

Шестнадцатеричная система счисления, широко применявшаяся в документации IBM/360, вытеснила восьмеричную.

Байт как минимально адресуемая ячейка памяти и 8-битные кодовые таблицы символов IBM/360 приняты разработчиками всех стран.

Старшие модели семейства IBM/360 и последовавшее за ними семейство IBM/370 были одними из первых компьютеров с виртуальной памятью и первыми серийными компьютерами, поддерживающими реализацию виртуальных машин.

Архитектура IBM/360 была настолько удачной, что стала на многие годы промышленным стандартом для компьютеров класса **мэйнфрэйм**².

¹ Ряд компьютеров с одинаковой архитектурой.

² Высокопроизводительный компьютер для многозадачной и многопользовательской работы.

Дальнейшим развитием IBM/360 стали системы 370, 390 и System z. Многие другие фирмы стали выпускать совместимые с IBM/360 компьютеры, например, семейство 470 фирмы Amdahl (англ.), мейнфреймы Hitachi, UNIVAC 9200/9300/9400 и др.

В СССР аналогом IBM/360 были машины серии ЕС ЭВМ.

Мини-компьютеры в третьем поколении также сделали большой шаг вперед. Для массового производства на заводах с низкоквалифицированным персоналом фирмой DEC были разработаны 16-битные компьютеры PDP-11. Компьютеры собирались из типовых функциональных модулей, которые при сборке подключались к одинаковым шинным разъёмам общей печатной платы. В отличие от других компьютеров своего времени, первые модели PDP-11 имели общую универсальную шину (Unibus). Ячейки системной памяти и регистры устройств ввода-вывода находились в едином адресном пространстве. При необходимости обмена данными любое устройство, подключенное к шине, могло обратиться к процессору с сигналом прерывания и «захватить шину» на определённое время.

Многие архитектурные решения мини-компьютеров PDP-11 легли в основу персональных микро-ЭВМ.

5.5. Четвертое поколение ЭВМ – на сверхбольших интегральных схемах (с 1980)

Появление сверхбольших интегральных схем (СБИС), включавших миллионы транзисторов, позволило создавать компьютер на одной печатной плате, а впоследствии – и на одной микросхеме. До появления PDP-11 компьютеры были настолько велики и дороги, что приобретали их только крупные организации с вычислительными центрами. В 80-х годах стали разрабатываться компьютеры и для персонального использования.

Первые массовые персональные компьютеры собирались на микропроцессорах Intel 8080 и продавались как комплект из печатной платы, интегральных схем, кабелей, источника питания и устройств ввода-вывода. Программное обеспечение к компьютеру не прилагалось. Покупателю приходилось собирать компьютер и писать к нему программное обеспечение самостоятельно. Позднее для архитектуры Intel 8080 появилась операционная система CP/M, написанная Гари Килдаллом. Операционная система помещалась на дискету и включала в себя файловую систему, а также интерпретатор командной строки для выполнения пользовательских команд с клавиатуры.

Первый персональный компьютер, продававшийся в собранном виде, был разработан в конце 70-х Стивом Джобсом и Стивом Возняком под маркой Apple (а позднее и Apple II), быстро набрав популярность в учебных заведениях и среди домашних пользователей.

IBM, долгое время лидировавшая на рынке больших компьютеров, к разработке персональных компьютеров приступила достаточно поздно. Первый компьютер с архитектурой IBM PC (IBM Personal Computer) на центральном процессоре Intel 8088 появился в 1981 году, но сразу стал самым покупаемым

компьютером в истории. Маленькая фирма Microsoft разработала для IBM PC операционную систему MS-DOS.



Рис. 34. Микропроцессор Intel 8080

Что касается пятого поколения, то оно больше ассоциируется не с конкретной архитектурой, а со сменой парадигмы¹. В настоящее время в мире господствует идея «всепроникающей» компьютеризации. Компьютеры управляют автомобилем, делают «умным» жилой дом, встраиваются в документы и т.д.

Компьютерная индустрия – наиболее динамичная в настоящее время. Способность производителей размещать все больше и больше транзисторов на кристалле микросхемы приводит к уменьшению размера компьютеров, увеличению вычислительной мощности процессора, объема памяти и других характеристик. Первоначально закон Мура² гласил, что количество транзисторов на одной микросхеме удваивается каждые 18 месяцев.

Закон Мура тесно связан с эффективным циклом в экономике. Развитие компьютерных технологий приводит к повышению качества продукции и снижению цены. Снижение цены ведет к массовому спросу. Массовый спрос – к удовлетворению потребностей широкого круга потребителей, что, в свою очередь, стимулирует развитие компьютерной индустрии, порождает спрос на лучшие технологии. Новые прикладные задачи приводят к возникновению новых компьютерных рынков и новых IT-компаний, конкурирующих между собой.

Следуя закону Мура, компьютерная индустрия, создавая широкий спектр разнообразных компьютеров, идет двумя путями:

создание компьютеров с нарастающей вычислительной мощностью в пределах заданной цены;

устойчивое снижение цены на устаревшие модели (в два раза за 1,5-2 года).

Другой двигатель компьютерного прогресса – закон Мирвольда³, утверждающий, что программное обеспечение подобно газу, который стремится полностью заполнить резервуар.

Например, первые текстовые редакторы умещались на дискетах емкостью 360 Кбайт, современные электронные редакторы занимают сотни мегабайт. В будущем, несомненно, они будут занимать гигабайты. Программное обеспечение продолжает развиваться и порождает постоянный спрос на высокопроизводительные процессоры, на память большего объема с минимальным временем

¹ Совокупность идей и технических решений, определяющих назначение и область применения компьютера.

² Гордон Мур – один из основателей фирмы Intel.

³ Натан Мирвольд, главный администратор компании Microsoft.

доступа, на устройства ввода-вывода более высокой емкости и производительности. Когда компьютер стоил миллионы долларов и имел сравнительно невысокие вычислительные ресурсы, никому в голову не приходило разрабатывать компьютерные игры.

У персонального компьютера IBM PC/XT в 1982 году объем жесткого диска составлял всего 10 Мбайт. Тридцать лет спустя у наследников IBM PC/XT устанавливаются жесткие диски емкостью несколько Тбайт.

Меньше чем за два десятилетия мы перешли от модемов, соединяющих два компьютера со скоростью 300 бит/с, к проводным и опτικο-волоконным сетям, где скорость передачи – уже десятки Гбит/с. Оптико-волоконные трансатлантические телефонные кабели (например, TAT-12/13) стоят около 700 млн долларов, действуют в течение 10 лет и могут передавать 300 000 звонков одновременно, поэтому себестоимость сеанса 10-минутной межконтинентальной связи составляет менее 1 цента. Лабораторные исследования подтвердили, что возможны системы связи, работающие со скоростью 1 Тбит/с на расстоянии более 100 км без усилителей.

Вопросы для самоконтроля:

1. Дайте определение понятию вычислительной машины.
2. Дайте определение понятию компьютерной архитектуры.
3. Кто сконструировал первую счетную машину?
4. Что включает в себя архитектура фон Неймана?
5. Как осуществляется обмен данными через общую шину (магистраль)?
6. Охарактеризуйте первое поколение ЭВМ.
7. Охарактеризуйте второе поколение ЭВМ.
8. Охарактеризуйте третье поколение ЭВМ.
9. Охарактеризуйте четвертое поколение ЭВМ.
10. Что включает в себя любая компьютерная архитектура?

6. КОНФИГУРАЦИЯ И ХАРАКТЕРИСТИКИ ОСНОВНЫХ УСТРОЙСТВ СОВРЕМЕННОГО ПЕРСОНАЛЬНОГО КОМПЬЮТЕРА

6.1. Типы компьютеров и их сравнительные характеристики

Профессор Амстердамского свободного университета, доктор физики Эндрю Стюарт Таненбаум в своей книге «Архитектура компьютера» предложил следующую классификацию современных компьютеров, которая считается общепринятой в мире:

Тип	Цена, \$
«Одноразовые» компьютеры	0,5
Встроенные компьютеры (микроконтроллеры)	5
Игровые компьютеры	50
Персональные компьютеры	500
Сетевые серверы	5000
Комплексы рабочих станций	50 000-500 000
Мэйнфреймы (суперкомпьютеры)	5 000 000

«Одноразовые компьютеры» – внедряемые компьютерные чипы, которые окружают нас повсеместно. Например, это RFID-микрочипы¹, которые применяются в самых разнообразных сферах человеческой деятельности:

- промышленность и торговля;
- транспортная и складская логистика, системы управления багажом;
- системы контроля и управления доступом;
- медицина - мониторинг состояния пациентов, наблюдение за перемещением по зданию больницы;
- библиотеки - станции автоматической книговыдачи, быстрая инвентаризация;
- удостоверения личности и паспорта;
- транспортные платежные карты;
- имплантаты для животных;
- системы обеспечения безопасности и т.д.

В современных микрочипах RFID кроме идентификаторов может храниться и другая информация об объекте, его уникальных свойствах, качествах и т.п. Например, персональные, в том числе и биометрические, данные человека.

Микроконтроллер – однокристалльный компьютер, способный в автоматическом режиме выполнять определенные задачи и предназначенный для

¹ RFID (англ. Radio Frequency IDentification, радиочастотная идентификация) – метод автоматической идентификации объектов, в котором посредством радиосигналов считываются или записываются данные, хранящиеся в так называемых транспондерах, или RFID-метках.

управления электронными устройствами. Диапазон устройств, работающих под управлением микроконтроллеров, крайне широк:

- бытовые приборы (будильники, стиральные машины, сушильные аппараты, микроволновые печи, охранные сигнализации);
- узлы и агрегаты транспортных средств;
- комплексы вооружений (крылатые ракеты, межконтинентальные баллистические ракеты, торпеды);
- средства телекоммуникаций;
- периферийные компьютерные устройства (принтеры, сканеры, модемы, приводы CD-ROM);
- развлекательные устройства (видеомагнитофоны, DVD-плееры, музыкальные центры, MP3-плееры, телеприставки);
- средства обработки видеоизображений (телевизоры, цифровые фотокамеры, видеокамеры, объективы, фотокопировальные устройства);
- медицинское оборудование (рентгеноскопические аппараты, томографы, кардиомониторы, цифровые термометры);
- торговое оборудование (торговые автоматы, кассовые аппараты) и т.д.

По объемам ежегодных продаж микроконтроллеры опережают на несколько порядков компьютеры всех остальных типов (за исключением «одноразовых»).

В отличие от микросхем RFID, выполняющих минимальный набор функций, микроконтроллеры хоть и невелики по размерам, но представляют собой полноценные вычислительные устройства. Каждый микроконтроллер состоит из процессора, памяти и средств ввода-вывода. Ввод-вывод, как правило, осуществляется посредством кнопок и переключателей с контролем состояния световых индикаторов, дисплея и звуковых компонентов устройства. Программное обеспечение микроконтроллеров в большинстве случаев «прошивается» производителем в постоянной памяти. Все микроконтроллеры можно разделить на два типа: универсальные и специальные. Первые фактически являют собой обычные программируемые компьютеры, уменьшенные в размере. Специальные же микроконтроллеры отличаются индивидуальной архитектурой и набором команд, приспособленным для решения определенного круга задач, например, связанных с воспроизведением мультимедийных данных.

Цены на микроконтроллеры невысоки и определяются разрядностью (4-, 8-, 16- и 32-разрядные), типом, емкостью памяти и рядом других факторов.

Дополнительными требованиями к архитектуре могут стать габариты, энергопотребление, ударостойкость и т.д. С целью обеспечить работу в реальном времени (с минимальной задержкой) в 80-е годы была предложена RISC¹-архитектура процессора, в котором быстродействие увеличивается за счёт упрощения кодировки и сокращения количества машинных инструкций.

Игровые компьютеры (консоли, приставки) – это, по существу, обычные компьютеры, в которых расширенные возможности графических и звуковых контроллеров сочетаются с ограничениями по объему программного обеспечения и пониженной расширяемостью. Первоначально в эту категорию вхо-

¹ Restricted (reduced) instruction set computer – компьютер с сокращённым набором команд.

дили компьютеры с процессорами низших моделей для простых игр типа пинг-понга, которые предусматривали вывод изображения на экран телевизора. С годами игровые компьютеры превратились в достаточно мощные системы, которые по некоторым параметрам производительности ничем не хуже, а иногда даже лучше персональных компьютеров.

В следующую категорию входят **персональные компьютеры**. Именно они ассоциируются у большинства людей со словом «компьютер». Персональные компьютеры бывают двух видов: настольные и портативные (ноутбуки, наладонники, планшеты). На них устанавливаются сложные операционные системы, они расширяемы, при работе с ними используется широкий спектр внешних устройств и программного обеспечения.

Центральным компонентом любого персонального компьютера является материнская плата с системными устройствами и стандартными интерфейсными разъемами для подключения периферийных модулей.

Персональные компьютеры, имеющие сетевое программное обеспечение и подключенные к локальной вычислительной сети, часто называются **рабочими станциями**. Мощные персональные компьютеры с соответствующим программным обеспечением могут использоваться в качестве **сетевых серверов** как в локальных сетях, так и в Интернете.

В связи с тем, что по соотношению цена/производительность позиции рабочих станций и персональных компьютеров постоянно улучшаются, в последние годы появилась практика их объединения в рамках кластеров рабочих станций (Clusters Of Workstations, COW), которые иногда называют просто кластерами. Они состоят из нескольких персональных компьютеров или рабочих станций, подключенных друг к другу по высокоскоростной сети и снабженных специальным программным обеспечением, которое позволяет направлять их ресурсы на решение единых задач (как правило, научных и инженерных). В большинстве случаев компоненты кластера – это совершенно обычные коммерческие машины, которые можно приобрести по отдельности в любом компьютерном магазине. Высокоскоростные сетевые соединения, как правило, тоже можно организовать при помощи стандартных сетевых плат. Кластеры отличаются удобством масштабирования – любой кластер можно расширить с десятка до нескольких тысяч машин. Количество компонентов кластера обычно ограничивается лишь толщиной кошелька пользователя. Нередко в виде кластеров организуются веб-серверы. Если частота обращений к страницам веб-сайта исчисляется тысячами в секунду, дешевле организовать кластер из нескольких сотен (или даже тысяч) серверов и распределить между ними нагрузку по обработке запросов. Кластеры, реализующие такую схему, часто называют серверными фермами (server farms).

Термин **мэйнфрейм** (от *англ.* mainframe) имеет три смысловых значения:

- компьютер с архитектурой IBM System/360, 370, 390, z-Series;
- универсальная супер-ЭВМ с возможностью прямого подключения нескольких пользовательских терминалов¹ в режиме on-line, предназначенный для

¹ Устройств ввода-вывода пользовательских данных.

выполнения ресурсоемких вычислений и организации централизованных хранилищ данных большой ёмкости;

– мощный сервер поддержки сетевых сервисов, включая облачные технологии обработки данных.

Одной из важнейших характеристик компьютера является **производительность** – количество операций, осуществляемых ЭВМ в единицу времени. Производительность определяется путём запуска на испытуемом компьютере тестовой программы, которая решает задачу с известным количеством операций и подсчитывает время, за которое она была решена.

Существует два показателя производительности процессоров: на операциях с целочисленными данными **MIPS**¹ и на операциях с данными вещественного типа **MFLOPS**². Динамика увеличения этой характеристики в процессе эволюции ЭВМ показана в таблице:

год	производительность	ед. измерения
1941	10^0	флопс
1949	10^3	кфлопс
1964	10^6	Мфлопс
1987	10^9	Гфлопс
1997	10^{12}	Тфлопс
2008	10^{15}	Пфлопс
2015	10^{18}	Эфлопс

Сейчас существует достаточно большое количество программ и утилит для самостоятельной оценки производительности компьютера (3D Mark, Si Software Sandra, Everest, HWiNFO32). У каждой есть свои преимущества и свои недостатки. Часто тесты компьютера разрабатываются в интересах конкретных фирм-производителей. Поэтому, чтобы достоверно сравнить два компьютера, необходимо использовать несколько тест-программ и, главное, тестировать на задачах, интересующих пользователя.

Национальным стандартом РФ в 2009 г. определён ряд характеристик, отражающих надёжность технических средств, в том числе и ЭВМ³:

1. **Надёжность** (Reliability, dependability) – свойство объекта сохранять во времени в установленных пределах значения всех параметров, характеризующих способность выполнять требуемые функции в заданных режимах и условиях применения, технического обслуживания, хранения и транспортирования.

¹ Внесистемная единица измерения производительности компьютеров, равная миллиону операций с фиксированной точкой в секунду (англ. Million Instructions Per Second).

² Внесистемная единица измерения производительности компьютеров, равная миллиону операций с плавающей запятой в секунду (от англ. Mega Floating point OPERations per Second, произносится как мегафлопс).

³ ГОСТ Р 27.002-2009. Национальный стандарт Российской Федерации. Надёжность в технике. Термины и определения: утв. Приказом Ростехрегулирования 09.12.2009 № 649-ст. – М.: Стандартинформ, 2010.

Примечание. Надежность является комплексным свойством, которое в зависимости от назначения объекта и условий его применения может включать безотказность, долговечность, ремонтпригодность и сохраняемость или определенные сочетания этих свойств:

– **безотказность** (Reliability, failure-free operation) – свойство объекта непрерывно сохранять работоспособное состояние в течение некоторого времени или наработки, где наработка может быть как непрерывной величиной (продолжительность работы в часах, километраж пробега и т.п.), так и целочисленной величиной (число рабочих циклов, запусков и т.п.);

– **долговечность** (Durability, longevity) – свойство объекта сохранять работоспособное состояние до наступления предельного состояния при установленной системе технического обслуживания и ремонта;

– **ремонтпригодность** (Maintainability) – свойство объекта, заключающееся в приспособленности к поддержанию и восстановлению работоспособного состояния путем технического обслуживания и ремонта;

– **время восстановления** (Restoration time) – продолжительность восстановления работоспособного состояния объекта;

– **сохраняемость** (Storability) – свойство объекта сохранять в заданных пределах значения параметров, характеризующих способности объекта выполнять требуемые функции, в течение и после хранения и (или) транспортирования.

2. **Срок службы** (Useful lifetime, lifetime) – календарная продолжительность эксплуатации от начала эксплуатации объекта или ее возобновления после ремонта до перехода в предельное состояние, т.е. в состояние объекта, при котором его дальнейшая эксплуатация недопустима или нецелесообразна, либо восстановление его работоспособного состояния невозможно или нецелесообразно.

3. **Точность результатов вычислений** – характеристика, присущая программе вычислений и представлением исходных данных (байт, слово, двойное слово). В современных языках программирования этот диапазон может быть увеличен в несколько раз.

6.2. Конфигурация современного персонального компьютера

Конфигурация компьютера – это совокупность его функциональных частей и связей между ними, обусловленная основными техническими характеристиками этих функциональных частей, а также требованиями решаемых задач.

Интерфейс (interface) – это совокупность средств и правил, обеспечивающих взаимодействие устройств вычислительной машины или системы обработки информации и (или) программ¹.

¹ ГОСТ 15971–90. Системы обработки информации. Термины и определения: утв. Постановлением Госстандарта СССР от 26.10.1990 № 2698 // Режим доступа: СПС «КонсультантПлюс».

В соответствии со схемой фон Неймана любой компьютер, в том числе и персональный, состоит из системных и внешних (периферийных) устройств:

Системная (материнская) плата	
Системные устройства (подключаются к системной шине)	Периферийные устройства (подключаются через внешние шины)
– микропроцессор	– устройства ввода
– оперативная память	– устройства вывода
– постоянная память	– устройства ввода-вывода
– контроллер-концентратор ввода-вывода	– устройства внешней памяти

Характеристики устройств рассмотрим на примере сборки конкретного настольного персонального компьютера (ПК).

Основные устройства ПК:

- системный блок;
- монитор;
- клавиатура;
- манипулятор «мышь».



Рис. 35. Персональный компьютер

Персональный компьютер – это прежде всего системный блок, который иногда ошибочно называют процессором. Основой системного блока является **корпус, или шасси**. Именно в корпусе фиксируются все устройства компьютера и блок их электропитания.

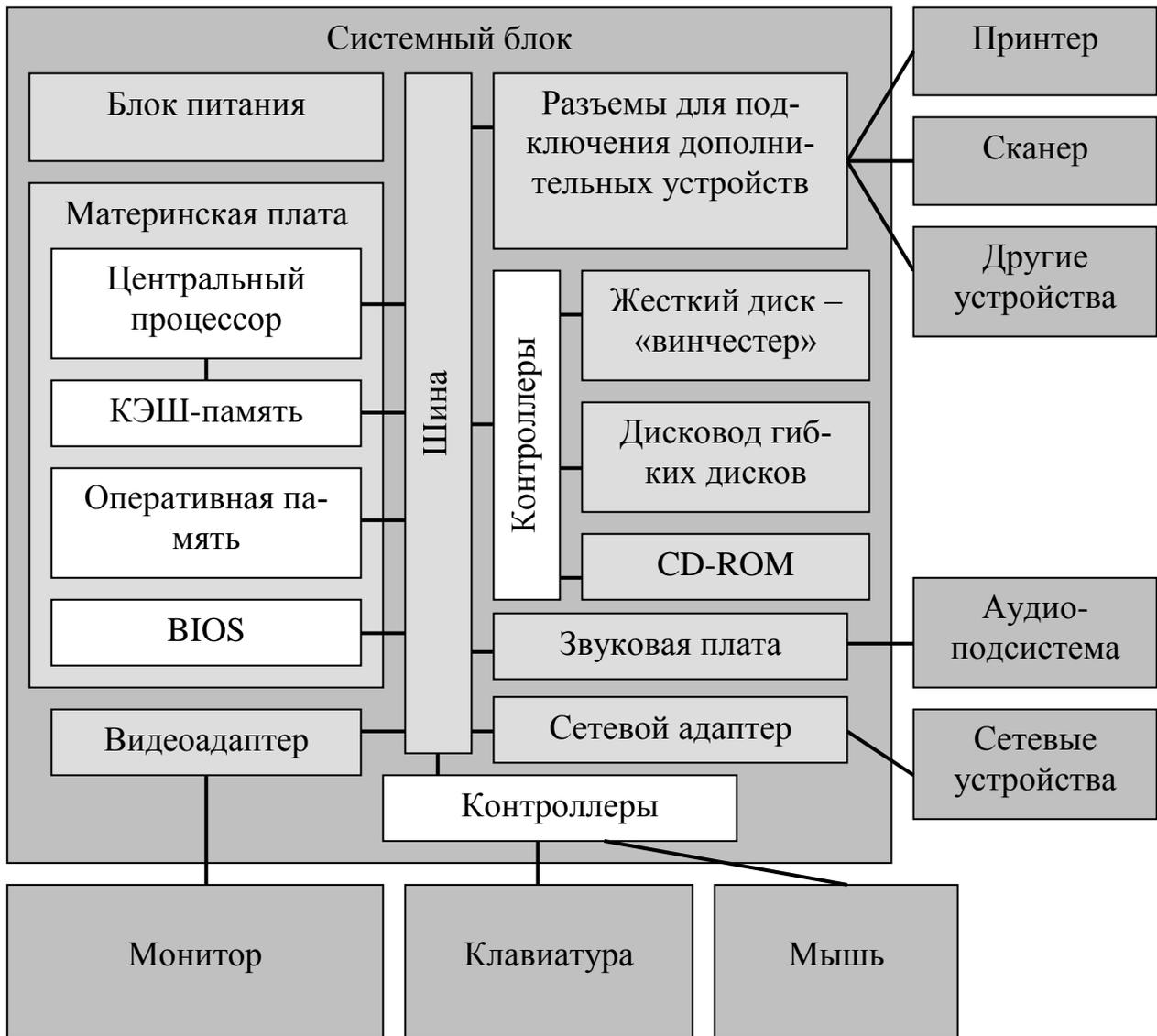


Рис. 36. Структурная схема устройства персонального компьютера

У настольных ПК наиболее широкое распространение получили корпуса двух типов:

1. Desktop (десктоп), располагающийся на столе горизонтально.
2. Tower (тауэр), располагающийся вертикально.



Рис. 37. Корпус персонального компьютера desktop (десктоп)



Рис. 38. Корпуса ПК mini-tower, midi-tower, big-tower

Корпуса последнего типа подразделяются, в свою очередь, на подтипы, различающиеся по числу 5,25” отсеков для установки приводов внешней памяти.

Для установки в специальные телекоммуникационные 19-серверные стойки и шкафы стойки используются Rack-корпуса.

При выборе корпуса ПК важным параметром является его форм-фактор.

Форм-фактор (англ. form factor) – стандарт, задающий габаритные размеры технического изделия, а также описывающий дополнительные совокупности его технических параметров, например, геометрическую форму, типы дополнительных элементов, размещаемых в/на устройстве, их положение и ориентацию.

Форм-фактор корпусов ПК определяет следующие характеристики:

- геометрические размеры материнских плат;
- общие требования по положению разъёмов и отверстий на корпусе;
- форму и положение ряда разъёмов (преимущественно питания);
- геометрические размеры блока питания;
- положение блока питания в корпусе;
- электрические характеристики блока питания.

ATX (от англ. Advanced Technology Extended) – наиболее распространенный форм-фактор настольных ПК. Отличается легким доступом к внутренним узлам компьютера, улучшенной вентиляцией внутри корпуса, возможностью установки большего числа полноразмерных плат расширения, расширенными возможностями по управлению энергопотреблением.

Micro ATX – малогабаритный вариант, хорошо подходящий для компактных ПК с минимумом плат расширения.

Функциональность современных материнских плат во многом определяется набором микросхем системной логики – **чипсетом**, которые обеспечивают взаимодействие процессора с другими устройствами компьютера, поддерживают стандартный набор интерфейсов ввода-вывода с соответствующими разъемами (портами) на материнской плате.

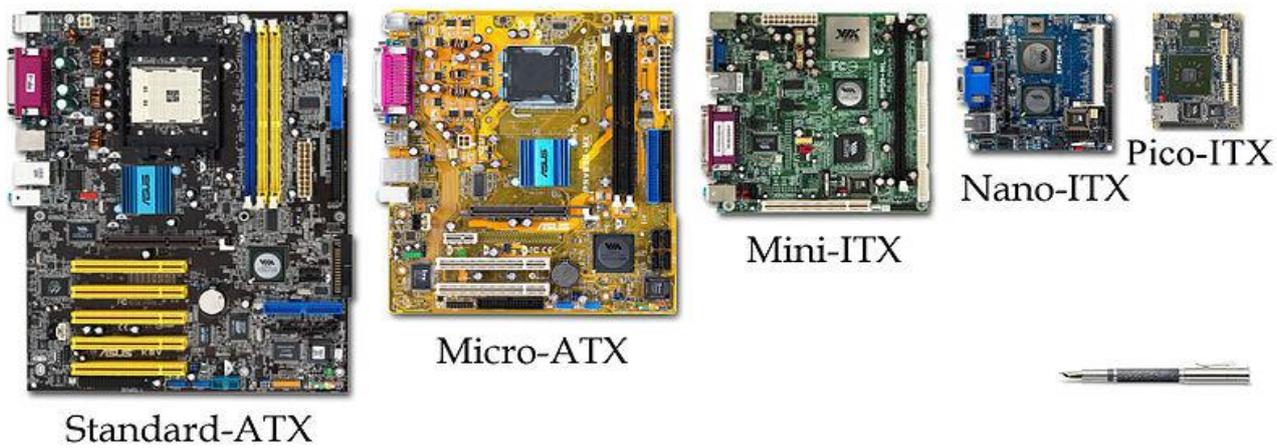


Рис. 39. Сравнительные размеры материнских плат ПК в зависимости от форм-фактора

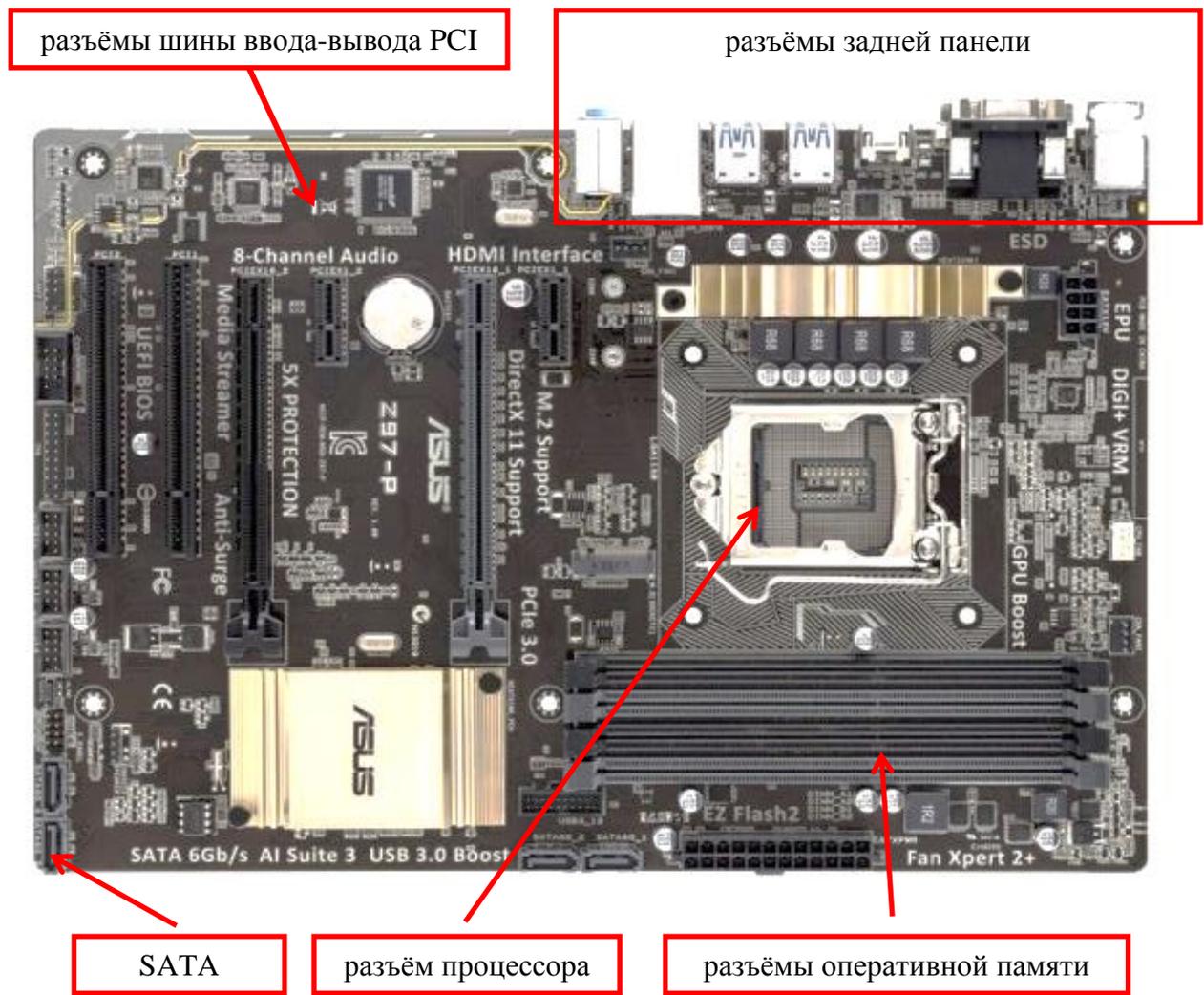


Рис. 40. Основные разъёмы материнской платы

Чтобы минимизировать время ожидания при считывании процессором данных из относительно медленной оперативной памяти, в современных ПК обычно предусмотрены три уровня кэш-памяти, интегрированной в чип центрального процессора:

- кэш-память первого уровня (L1);
- кэш-память второго уровня (L2);

– кэш-память третьего уровня (L3).

В связи с относительно высокой потребляемой мощностью на единицу площади на корпус микропроцессора устанавливается система охлаждения (кулер), состоящая, как правило, из металлического радиатора и вентилятора на нем. Форм-фактор кулера определяется по типу разъема процессора (гнезда на материнской плате).



Рис. 41. Процессор Intel Core i7-4770K



Рис. 42. Оперативная память Kingston «Value RAM» DDR3

Внешнюю оперативную память нужно выбирать с учетом частоты системной шины и двухканальной поддержки процессором.

Оперативная память типа DDR3 (double data rate¹) позволяет процессору считывать/записывать 2 бита данных за 1 такт по системной шине, т.е. эффективная частота передачи данных будет в два раза выше частоты системной шины. Для оптимального взаимодействия памяти и процессора поддерживаемая этими устройствами частота системной шины должна совпадать.

Пропускная способность или пиковая скорость передачи данных – характеристика производительности памяти, которая равна произведению частоты системной шины на объем данных, передаваемых за такт.

Латентность – временная задержка сигнала (в тактах системной шины), от которой зависит пропускная способность.

Следует четко понимать, что оперативная память – это энергозависимая память для размещения программ и данных во время их обработки процессором. После выключения компьютера данные, хранимые в оперативной памяти, будут потеряны. Для долговременного хранения компьютерной информации используются другие виды памяти, называемой внешней.

Внешние устройства подключаются к материнской плате через специализированные разъёмы ввода-вывода, содержащие набор сигналов соответствующего

¹ Двойная скорость передачи данных.

щей шины. В отличие от схемы подключения точка-точка, к шине можно параллельно подключить несколько устройств.

Шина ввода-вывода – подсистема, которая обеспечивает обмен данными между процессором и функциональными блоками ввода-вывода по определённым правилам. Стандарты шин устанавливают требования к физическим и логическим параметрам сигналов.

Основными характеристиками шин материнской платы ПК являются назначение (какие устройства подключаются), тип разъёма, тактовая частота, напряжение питания, разрядность данных и пропускная способность.

Разъёмы шин персонального компьютера обычно конструируются так, что несовместимое устройство в него не вставится физически.

Спецификация PCI была разработана лидерами компьютерной индустрии фирмами Intel, IBM, Compaq, DEC, NCR и в начале 90-х прошлого века принята как открытый индустриальный стандарт. Стандарт PCI поддерживает технологию plug-and-play¹, конфигурирование и настройка подключаемых устройств осуществляются программным способом.

В базовой реализации 1992 года 32-разрядная шина поддерживала одновременно до десяти внешних устройств и передачу данных с тактовой частотой 33 МГц. Впоследствии тактовая частота была удвоена (66 МГц), а в каждом такте передавалась 64-разрядная порция данных.

К шине PCI можно подключать сетевые адаптеры, звуковые и видеокарты, TV-тюнеры и другие внешние устройства ввода-вывода с максимальной пропускной способностью:

$$P = 528 \text{ МБ/с} (66 \text{ МГц} * 8 \text{ байт}).$$

Суть технологии PCI Express (PCIE) заключается в замене параллельной шины на несколько высокоскоростных последовательных соединений «точка-точка». По аналогии с компьютерными сетями PCI Express выполняет роль универсального коммутатора, соединяющего по дуплексным² последовательным каналам внешние устройства и процессор. Технология передачи данных, примененная в PCI Express, обеспечивает возможность масштабирования порции данных (PCI Express x1, x2, x4, x8, x16 и x32). Для снижения количества сбоев и искажений при передаче информации на частоте $F = 2,5 \text{ ГГц}$ в PCI E применяется избыточное кодирование (1 байт кодируется 10-ю битами).

Пропускная способность шины PCI E x32 достигает 16 Гб/сек. Как правило, в скоростные PCI E x16, PCI E x32 разъёмы настольных ПК вставляются высокопроизводительные видеокарты.

¹ Дословно – «втыкай и работай» (англ.).

² Передача может осуществляться одновременно в двух направлениях.

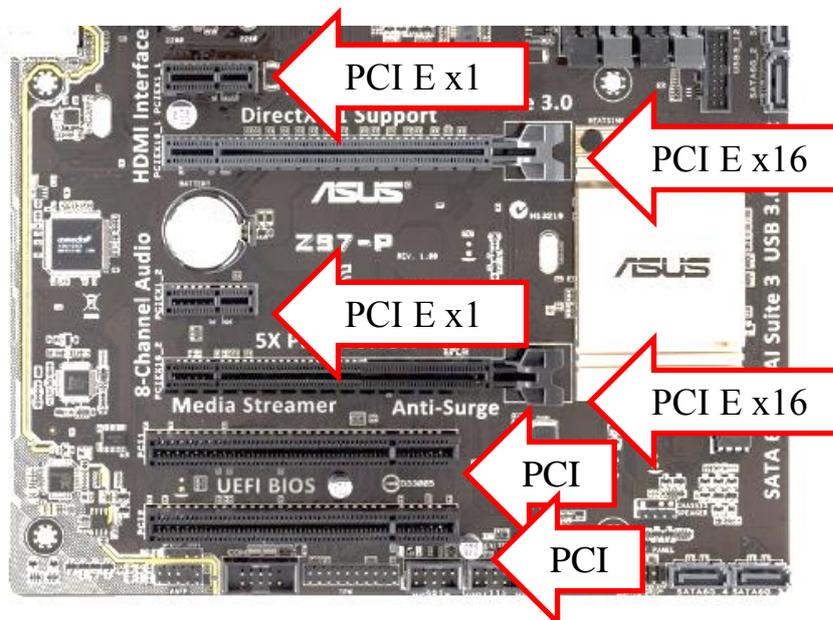


Рис. 43. Расположение разъемов PCI на материнской плате ASUS Z97-P

В подсистему отображения видеоинформации входят два основных компонента:

- видеоадаптер;
- монитор.

На многих материнских платах уже имеются встроенные видеоадаптеры, наличие которых можно определить по установленным на задней панели разъёмам.

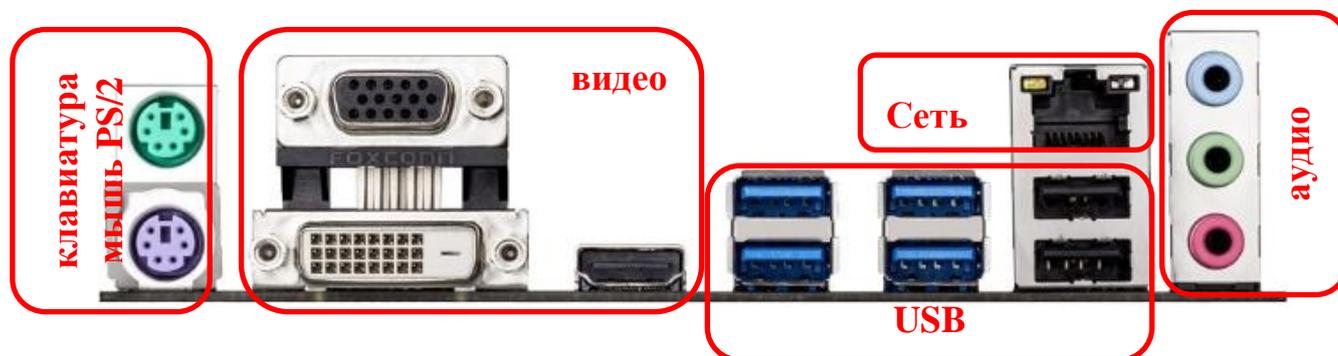


Рис. 44. Расположение разъемов на задней панели ASUS Z97-P

Однако у подобных графических процессоров или интегрированных видеоадаптеров относительно невысокая вычислительная мощность, и при необходимости работы с высококачественной графикой следует устанавливать внешние видеокарты.

Унифицированные конвейерные процессоры – это отдельные процессоры для параллельной обработки фрагментов видео и изображений 3D-графики. Сложные процессы построения и перемещения объемных объектов, изменения цветовых оттенков с помощью специальных алгоритмов разбиваются на более простые процедуры, выполняемые одновременно. Сложные с виду геометрические объекты могут быть быстро визуализированы при помощи множества простых геометрических форм.



Рис. 45. Видеокарта ASUS GTX 750 TI-PH-2GD5

Подключение монитора осуществляется по нескольким интерфейсным разъёмам: D-Sub (для аналогового видеосигнала), 2 DVI-D, HDMI. Максимальное разрешение, поддерживаемое видеокартой – 2560x1600 пикселей.

Основным узлом компьютерного монитора может быть:

- жидкокристаллический дисплей LCD (Liquid Crystal Display);
- электронно-лучевая трубка CRT (Cathodes Ray Tube).

За счет множества достоинств жидкокристаллические мониторы вытеснили с рынка ПК мониторы с электронно-лучевыми трубками. Например, у ЖК-мониторов значительно меньше вес и толщина. Для отображения информации используется вся поверхность экрана. В ЖК-мониторах применяется непосредственная адресация каждого пикселя. ЖК-экраны плоские и лишены геометрических искажений. Жидкокристаллические мониторы имеют более низкое энергопотребление. Нет проблемы вредного электромагнитного излучения и т.д.

На экране цифрового монитора отображается прямоугольное растровое изображение, состоящее из множества трехцветных точек, подсвеченных с разной интенсивностью (RGB-тетрад)¹. Исторически сложилось, что пропорции сторон дисплея соответствовали телевизионным форматам:

- 4x3 – формат кадра со времен кинематографа;
- 16x9 – широкоэкранный формат цифрового телевидения стандартной и высокой чёткости (HDTV);
- 16x10 – наиболее близкий к пропорции «золотого сечения» (1,61) формат применяется в ранних моделях широкоформатных компьютерных мониторов и ноутбуках.

¹ Некоторые мониторы работают в монохромном (одноцветном) режиме.



Рис. 46. Мультимедийный цифровой ЖК-монитор Asus VS278Q

Вместе с пропорциональным отношением сторон одной из основных характеристик монитора является линейное разрешение в пикселях. Физический размер экрана определяется длиной диагонали в дюймах или сантиметрах.

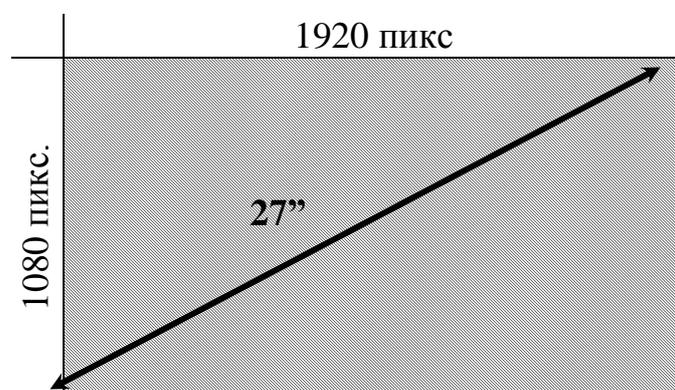


Рис. 47. Линейные характеристики дисплея

Для подключения к видеоадаптеру (видеокарте) мониторы оснащаются аналоговым интерфейсом VGA, цифровым DVI или мультимедийным интерфейсом высокой четкости HDMI.

Версия HDMI 2.0 рассчитана на трехканальную передачу видеосигнала с пропускной способностью до 6 Гбит/с на канал. Интерфейс поддерживает:

- передачу сигнала формата Full HD 3D со скоростью 120 кадров/с (FPS);
- передачу 3D форматов со скоростью 25 FPS;
- разрешение UHD 4K при частоте кадров 50/60 Гц (50/60 FPS);
- до 32 каналов аудио с максимальной звуковой частотой 1536 кГц;
- одновременную передачу двойного видеопотока на одном экране;
- одновременную передачу до 4 аудиопотоков нескольким пользователям;
- дисплеи с соотношением сторон 21:9 и т.д.

Клавиатура – одно из важнейших устройств управления компьютером, используется для ввода в систему команд и данных.

Существуют промышленные стандарты клавиатур:

- 84-клавишная клавиатура AT;
- 101-клавишная расширенная клавиатура;
- 104-клавишная расширенная клавиатура Windows.

Клавиатуры подключаются к компьютеру с помощью 6-контактного разъема PS/2 (mini-DIN) или 4-контактного USB. Некоторые клавиатуры поставляются в комплекте с адаптером-переходником USB-PS/2.

Каждая клавиша имеет уникальный скан-код, который при нажатии передается в компьютер. Операционная система преобразует его в соответствующие алфавитно-цифровые символы или управляющие коды. В Windows можно установить несколько раскладок клавиатур для поддержки различных языков. Независимо от нарисованного на клавише символа, скан-кода, посылаемый клавишей, можно интерпретировать как угодно.

Многие считают, что появление и распространение **мыши** и графического интерфейса пользователя – это заслуга Apple, но очевидно, что сама идея и технология были заимствованы у SRI и Xerox. С появлением Windows и OS/2 манипулятор «мышь» стал обязательной принадлежностью всех персональных компьютеров. Эти устройства выпускаются различными производителями, имеют разнообразные конструкции и размеры. Некоторые компании, взяв за основу стандартную мышь и перевернув ее, создали устройство трекбол, где пальцем двигается шарик. Интерфейсы подключения мыши – такие же как у клавиатуры, поэтому производители часто изготавливают интегрированные комплекты, объединенные общим дизайном.

Фото-сенсор оптической мыши с определенной частотой «фотографирует» поверхность. Сравнивая полученные изображения, процессор мыши высчитывает координаты перемещения мыши. Лучшие модели работают с частотой 7000 кадров в секунду и используют лазерную подсветку. Разрешение – это количество пикселей на дюйм, которое фиксирует оптический сенсор.



Рис. 48. Беспроводная клавиатура и мышь Genius KB-8005 с USB-приемником

В настоящее время существует три основных типа устройств **внешней памяти ПК**:

- магнитные;
- оптические;
- электронные.

К материнской плате ПК накопители на жестких магнитных дисках и приводы (дисководы) накопителей на оптических дисках подключаются в основном по стандарту АТА (англ. Advanced Technology Attachment), реализованному через интерфейсы двух типов:

- PATA (IDE) – параллельный интерфейс подключения накопителей (жёстких дисков и оптических дисководов) к компьютеру;
- SATA – последовательный интерфейс.

В настоящее время накопители на жестких магнитных дисках начинают вытесняться твердотельными накопителями (англ. SSD, solid-state drive) – энергонезависимыми перезаписываемыми полупроводниковыми запоминающими устройствами. Несмотря на то, что внешне они имеют форм-факторы HDD, SSD – это не диски с движущимися механическими частями, а печатная плата с электронными микросхемами памяти.



Рис. 49. Магнитный жесткий диск Western Digital WD20EZRX



Рис. 50. Твердотельный накопитель SSD Kingston SV300S37A/60G

Преимущества SSD-накопителей по сравнению с HDD:

- высокая производительность: скорости чтения и записи ограничены лишь пропускной способностью интерфейса и применяемых контроллеров;
- стабильность времени считывания файлов вне зависимости от их расположения или фрагментации;
- SSD позволяют увеличить скорость работы компьютера (включение и выключение, установку и запуск операционной системы, приложений, требующих больших объемов памяти, открытие и закрытие файлов больших объемов) в несколько раз, минимально – в 3-4 раза;
- низкая потребляемая мощность;

- полное отсутствие шума от движущихся частей и охлаждающих вентиляторов;
- высокая механическая стойкость за счет отсутствия подвижных частей;
- широкий диапазон рабочих температур;
- малый вес.

Недостатки полупроводниковых накопителей:

- ограниченное количество циклов перезаписи (обычная флэш-память позволяет записывать данные примерно 10 000 раз, более дорогостоящие виды памяти – более 100 000 раз);
- высокая цена;
- рост стоимости в зависимости от ёмкости у SSD-накопителей выше, чем у HDD.

В устройствах с оптическим принципом хранения данных запись и считывание осуществляются на вращающийся диск с помощью лазерного луча, а не магнитного поля. Стандарты компьютерных оптических технологий можно разделить на три основные группы:

- CD;
- DVD;
- DVD с повышенной плотностью и Blue-Ray (BD).

Как правило, термином «CD/DVD/BD» называют как сам носитель, так и привод (дисковод), в который он вставляется.

Типовой оптический носитель информации CD-ROM диаметром 120 мм и толщиной 1,2 мм предназначен только для хранения около 700 Мбайт данных, что соответствует 80 минутам высококачественного звучания в несжатом формате.

Активно используются CD/DVD/BD-диски в индустрии компьютерных игр, видео-, книгоиздании. Большинство дистрибутивов компьютерных программ распространяется на оптических компакт-дисках. В основном это связано с невысокой стоимостью чистой «болванки».

Оптические диски могут иметь возможность:

- заводской «штамповки» (ROM);
- однократной записи и «дозаписи» на свободное место ($\pm R$);
- перезаписи (RW).

Стандарт Blue-ray позволяет записывать до 25 Гбайт данных или более 11 часов непрерывного видео на одностороннем однослойном диске диаметром 120 мм с помощью синего лазера. На один такой носитель можно записать 4,5-часа фильма в формате HDTV.

В течение долгих лет единственным универсальным средством переноса данных с ПК на ПК была 3,5-дюймовая дискета объемом 1,44 Мбайт. Сейчас приводы FDD на ПК не устанавливаются.

Высокая производительность, большая емкость и небольшой размер новейших устройств флэш-памяти делают ее прекрасным дополнением при использовании в портативных компьютерах и других цифровых устройствах.

На сегодняшний день среди съёмных флэш-карт достаточно большой выбор: Compact Flash, Micro drive, Secure Digital, mini SD, micro SD, xD-Picture, Smart Media, Memory Stick, Multi Media Card, RS-MMC, DV-RS-MMC, micro MMC, micro Memory Stick и др.



Рис. 51. Флэш-карты

Все компьютеры, начиная с 1996 года, стали оснащаться портами USB, а в XXI в. загрузка с USB-устройств стала стандартной функцией BIOS – базовой системы ввода-вывода, размещаемой на материнской плате ПК.

USB (англ. Universal Serial Bus¹) – последовательный интерфейс передачи данных для подключения периферийных устройств. Сейчас к USB подключается практически любая периферия: переносные приводы магнитных и оптических дисков, клавиатуры, мыши, аудио- и видеоадаптеры, TV-тюнеры, сканеры, сетевые адаптеры и т.п.

6.3. Периферийные устройства персонального компьютера

Принтером называется устройство, предназначенное для распечатки текстов и графических изображений на бумаге.

Тип принтера прежде всего определяется желаемым качеством бумажной копии, а также такими характеристиками, как скорость печати, т.е. сколько листов в минуту, размеры принтера и др. Под качеством печати понимают количество точек на единицу длины, которое может напечатать принтер, а также возможность цветной печати.

По принципу переноса изображения на носитель принтеры делятся на **матричные, лазерные, струйные и сублимационные.**

По количеству цветов печати – на **чёрно-белые (монохромные) и цветные.**

Подключать принтеры к компьютерам можно по проводным каналам через SCSI кабель, последовательный порт (COM), параллельный порт (LPT, IEEE 1284), локальную сеть (LAN), по шине USB, а также посредством беспро-

¹ Универсальная последовательная шина.

водного соединения через ИК-порт (IRDA), по Bluetooth, по Wi-Fi (в том числе с помощью Air Print).

ИК-соединение возможно только с устройством, находящимся в прямой видимости, в то время как использующие радиоволны интерфейсы Bluetooth и Wi-Fi функционируют на расстоянии до 10-100 метров.

Некоторые принтеры (в основном струйные фотопринтеры) располагают возможностью автономной (то есть без компьютера) печати, обладая устройством чтения flash-карт или портом сопряжения с цифровым фотоаппаратом, что позволяет осуществлять печать фотографий напрямую с карты памяти или фотоаппаратов. Принтеры, поддерживающие технологию Air Print, дают возможность распечатывать документы и фотографии с непосредственно мобильных устройств на базе iOS без использования кабеля (соединение осуществляется по Wi-Fi). Air Print доступна для iPad, а также для iPhone и iPod Touch не ниже третьего поколения.



Рис. 52. Принтер Samsung CLP-365

Сетевой принтер – принтер позволяющий принимать задания на печать от нескольких компьютеров, подключенных к локальной сети. Программное обеспечение сетевых принтеров поддерживает один или несколько специальных протоколов передачи данных, таких как IPP. Такое решение является наиболее универсальным, так как обеспечивает возможным вывод на печать из различных операционных систем, чего нельзя сказать о Bluetooth- и USB-принтерах.

Качественное изображение получается при использовании **струйных принтеров**. Изображение (в том числе и цветное) формируется микрокаплями специальных чернил, выдуваемых на бумагу с помощью сопел. Недостаток этих устройств – в ограниченности ресурса головки, что приводит к более высокой стоимости листа распечатки. Струйный принтер требует лучшей бумаги, чем матричный. Для цветной печати струйные принтеры часто бывают самыми оптимальными, т.к. лазерный принтер хотя и дает несколько лучшее изображение, но стоит значительно дороже.

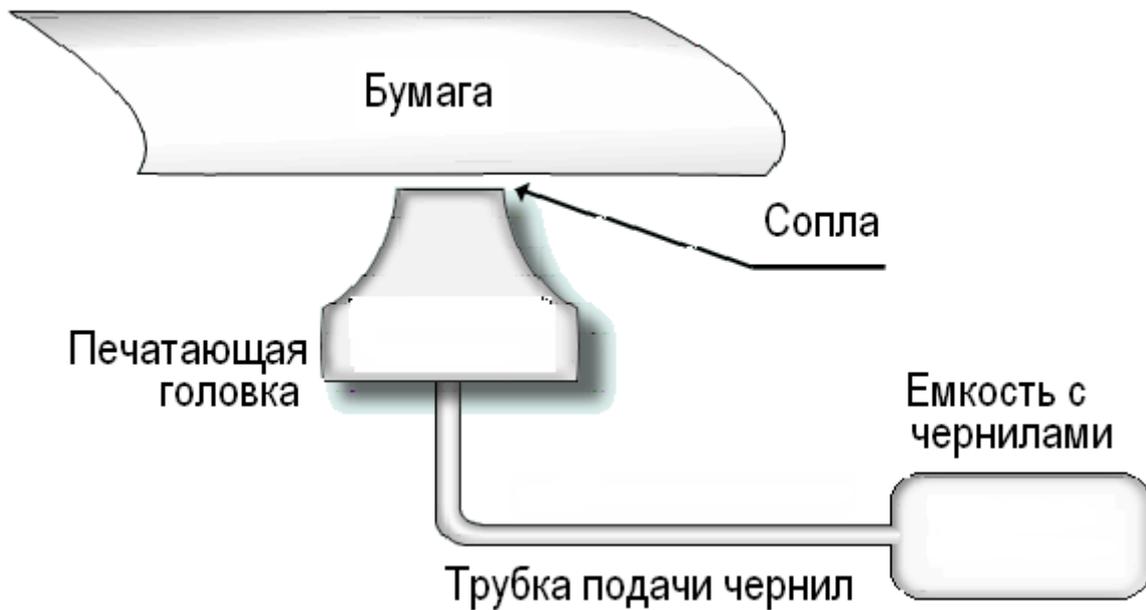


Рис. 53. Принцип работы струйного принтера

Лазерные принтеры обеспечивают наивысшее качество печати. Они работают по следующему принципу: луч лазера, пробегая по селеновому барабану, электризует его, наэлектризованный барабан притягивает частицы краски, после чего изображение переносится с барабана на бумагу. Далее лист проходит через тепловой барабан, где под действием тепла изображение фиксируется на бумаге. Изображение получается за 3-15 секунд, а его качество близко к типографскому.

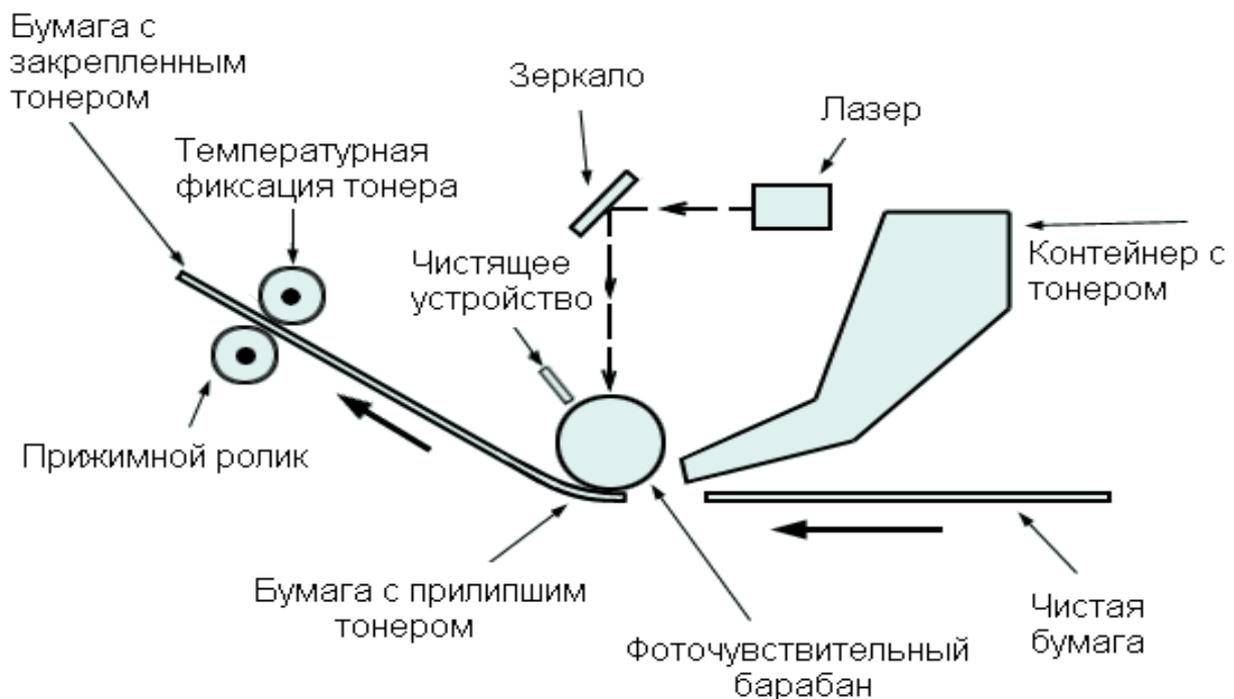


Рис. 54. Принцип работы лазерного принтера

Led-принтеры вместо полупроводникового лазера используют мельчайшие светодиоды.

Часто принтеры совмещают со сканерами. Подобные устройства называются многофункциональными устройствами (МФУ).

Сканер предназначен для ввода в компьютер информации с бумаги или с других оригиналов. Они передают ее в компьютер в виде оцифрованного изображения. Если на бумаге был печатный текст, то специальные программы (Fine Reader) могут достаточно успешно распознать его, если, конечно, текст не рукописный.

Сканеры различаются по способу использования. Ручные сканеры – дешевые компактные устройства, которые можно использовать для сканирования вручную небольших картинок, фотографий.

Страничные сканеры протягивают бумагу через себя.

Планшетные сканеры наиболее дорогие и громоздкие, но универсальные. Все современные модели планшетных сканеров являются цветными.

Барабанные сканеры – решение высшего уровня. Они имеют большую скорость сканирования. Недостаток – высокая цена.

Встречается другая классификация сканеров: ручные, планшетные, роликовые, проекционные.

Графический планшет (дигитайзер). Изображение вводится в ПК со специального планшета, оборудованного чувствительной поверхностью. Эта поверхность реагирует на испускаемые пером сигналы и передает точные координаты «точки соприкосновения» в ПК. Тесный контакт пера с рабочей поверхностью планшета не обязателен, поэтому между ними может находиться лист бумаги. На бумаге может быть изображен рисунок. Так можно ввести в компьютер рисунок, для чего его нужно положить под специальную пленку и контуры обвести пером.

Вопросы для самоконтроля:

1. Что такое конфигурация компьютера?
2. Дайте определение понятию интерфейса.
3. Перечислите и охарактеризуйте основные устройства ПК.
4. Что такое форм-фактор?
5. Что такое пропускная способность?
6. Что входит в подсистему отображения видеoinформации?
7. Что такое унифицированные конвейерные процессоры?
8. Как определяется физический размер экрана?
9. Охарактеризуйте основные типы устройств внешней памяти ПК.
10. Охарактеризуйте стандарты компьютерных оптических технологий.
11. Охарактеризуйте типы принтеров.

7. СОЗДАНИЕ ПРОГРАММ НА КОМПЬЮТЕРЕ

7.1. Порядок и правила решения задач с использованием компьютера

Информатика, или компьютерная наука (Computer Science) неотделима от **кибернетики** – науки «о связи, управлении и контроле в машинах и живых организмах»¹. Возникновение и активное развитие кибернетики связано с созданием в середине прошлого века электронных вычислительных машин, которые стали основными техническими средствами для решения кибернетических задач.

В кибернетике управляемая система рассматривается как множество взаимосвязанных объектов, способных воспринимать, запоминать и перерабатывать информацию, а также обмениваться ею. Подобными системами могут быть технические автоматы, компьютеры, человеческий мозг, биологические популяции, человеческое общество и пр.

В основе теории управления лежит **кибернетический цикл**.

В каждый момент времени система находится в каком-то состоянии, которое можно описать с заданной точностью через определенные значения ее параметров. Если нужно перевести систему в другое состояние, следует применить к ней управленческое воздействие, а потом оценить степень соответствия полученного результата (новых значений параметров системы) тому, что требуется, и, при необходимости, скорректировать управленческое воздействие. Контроль полученного результата осуществляется за счет **обратной связи**.

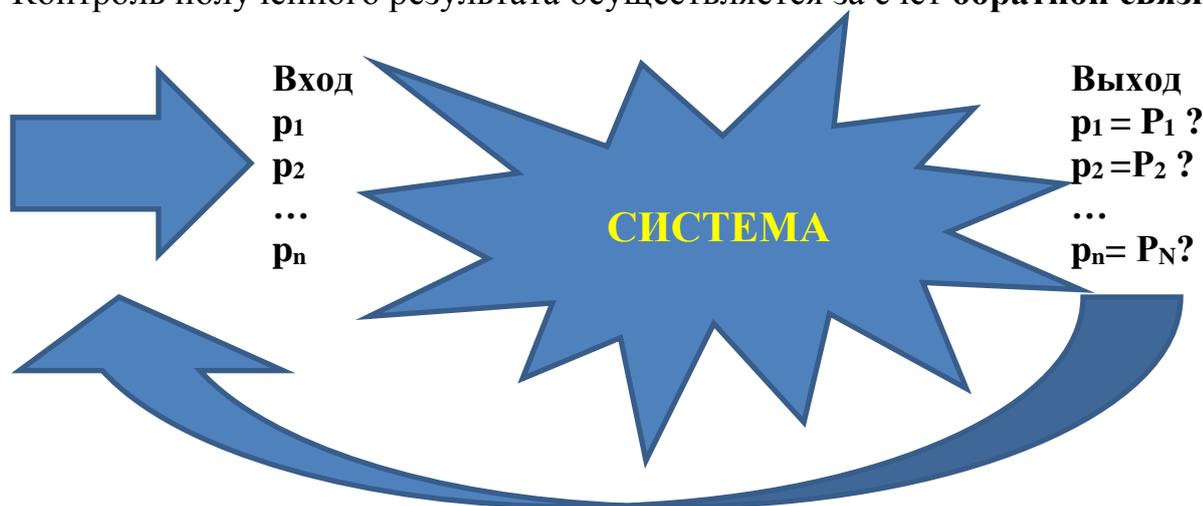


Рис. 55. Кибернетический цикл

Решение задачи с применением компьютера можно рассматривать как последовательный процесс, реализуемый по законам кибернетики.

В зависимости от характера задачи и конкретных целей некоторые этапы могут быть пропущены.

¹ Винер Н. Кибернетика или Управление и связь в животном и машине / пер. с англ. И.В. Соловьева и Г.Н. Поварова; под ред. Г.Н. Поварова. – 2-е изд. – М.: Наука; Главная редакция изданий для зарубежных стран, 1983. – 344 с.

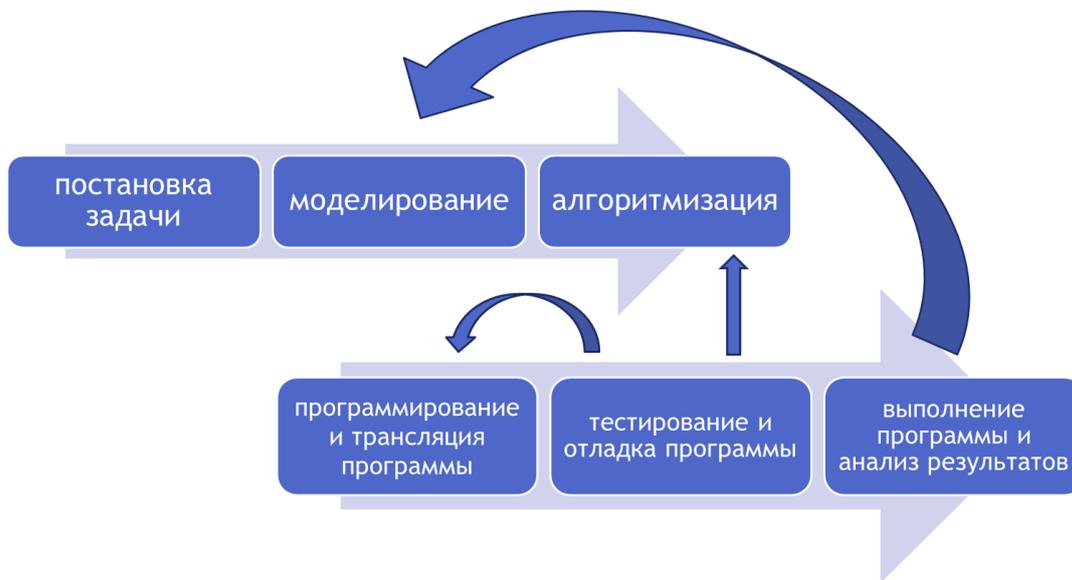


Рис. 56. Типовые этапы решения задачи с помощью компьютера

Первый этап – постановка задачи – включает:

- сбор информации о задаче;
- определение целей решения и вида выходных результатов;
- определение состава и описание типов данных, подлежащих обработке;
- описание условий выполнения задачи.

На этом этапе задача формализуется, т.е. записывается формальными методами. Сюда входят определение сущности процессов обработки информации, формулирование ограничений (время выполнения, точность и т.п.) и разработка методов обработки ошибок.

Данный этап является основополагающим, поэтому ему должно быть уделено особое внимание. Для этого прежде всего необходимо, чтобы задача была полно и четко описана. К сожалению, это требование часто нарушается. Как иронически отмечают программисты, основные усилия уходят на то, чтобы понять, чего хочет заказчик, в отличие от того, что он говорит. Поскольку в роли заказчика чаще всего выступают лица, мало знакомые с компьютерной техникой, корректная постановка задачи для них может оказаться достаточно сложной проблемой. Поэтому обычно для разработки задания привлекаются как специалисты-предметники, так и профессиональные программисты. Этот этап может потребовать существенных временных затрат. Однако подобные издержки не могут идти ни в какое сравнение с потерями, возникающими при необходимости частичной или полной переделки программы, составленной по некачественно сформулированным требованиям.

Второй этап – моделирование – включает:

- определение отношений, логических и математических выражений, устанавливающих зависимости входных и выходных данных с учетом условий выполнения задачи;
- разработка словесного или математического описания поведения системы, объекта или процесса.

При построении математических моделей далеко не всегда удается найти точные формулы, и тогда используются математические методы приближительной оценки с заданной степенью точности.

Третий этап – разработка алгоритма – включает:

- описание пошагового решения задачи одним из общепринятых способов (словесное, блок-схема, псевдокод);
- разработка метода и средств оценки правильности работы алгоритма.

Алгоритмизация – этап, предшествующий кодированию на язык исполнителя (компьютера). Следует подчеркнуть, что сам по себе компьютер не умеет решать никаких задач: даже для сложения двух чисел человек должен написать соответствующую команду, обеспечить ее ввод в память машины и заставить ее выполнять эту команду. Для решения же более сложной задачи человек должен составить целую программу, состоящую из многих сотен или даже тысяч команд. Перед составлением программы создается алгоритм, на основе которого впоследствии и пишется программа.

Четвёртый этап – программирование – включает:

- перевод алгоритма на язык программирования;
- трансляцию программы в исполняемые машинные коды.

Программа – это представление алгоритма решения задачи на языке исполнителя, а языком компьютера являются двоичные сигналы.

В современных компьютерах прикладное программное обеспечение в ряде случаев позволяет решить задачу (реализовать алгоритм) без составления программы на языке программирования. Например, непосредственное программирование не обязательно при использовании специализированных прикладных программ с удобными графическими интерфейсами: табличных и текстовых процессоров, систем управления базами данных, систем автоматизированного проектирования, графических, видео- и аудиоредакторов.

Пятый этап – отладка и тестирование.

Отладка программы – это процесс устранения выявленных в ней алгоритмических, синтаксических, семантических и других ошибок.

Тест – это такой набор входных воздействий на систему, для которого известен конечный результат. Соответственно, тестирование алгоритмов и программ – это их «прогон» на специально подобранных наборах входных данных, при которых известен выходной результат.

В последние годы для создания новых программ или решения типовых задач на ПК используются специализированные **среды программирования**. Данные среды позволяют использовать ранее созданные модули, т.е. создавать программу не из отдельных команд, а как бы складывать ее из «кубиков». Подобные модули, библиотеки и т.д. пишут программисты по всему миру. Поскольку выработаны стандарты совместимости, то можно без особых трудозатрат использовать их для создания новых программ. Среды программирования объединяют в себе три этапа: собственно, само программирование, отладку и тестирование.

Шестой этап – выполнение программы и анализ результатов.

Тестирование, отладка и анализ результатов – важнейшие процедуры контроля правильности решения задачи.

Алгоритмические ошибки, выявленные на данных этапах, могут потребовать от разработчика возврата к этапу моделирования или корректировки алгоритма, что и является обратной связью.

В Интернете в разных вариантах можно найти шуточный пример процесса решения задачи изготовления качелей, но без обратной связи (рис. 57).

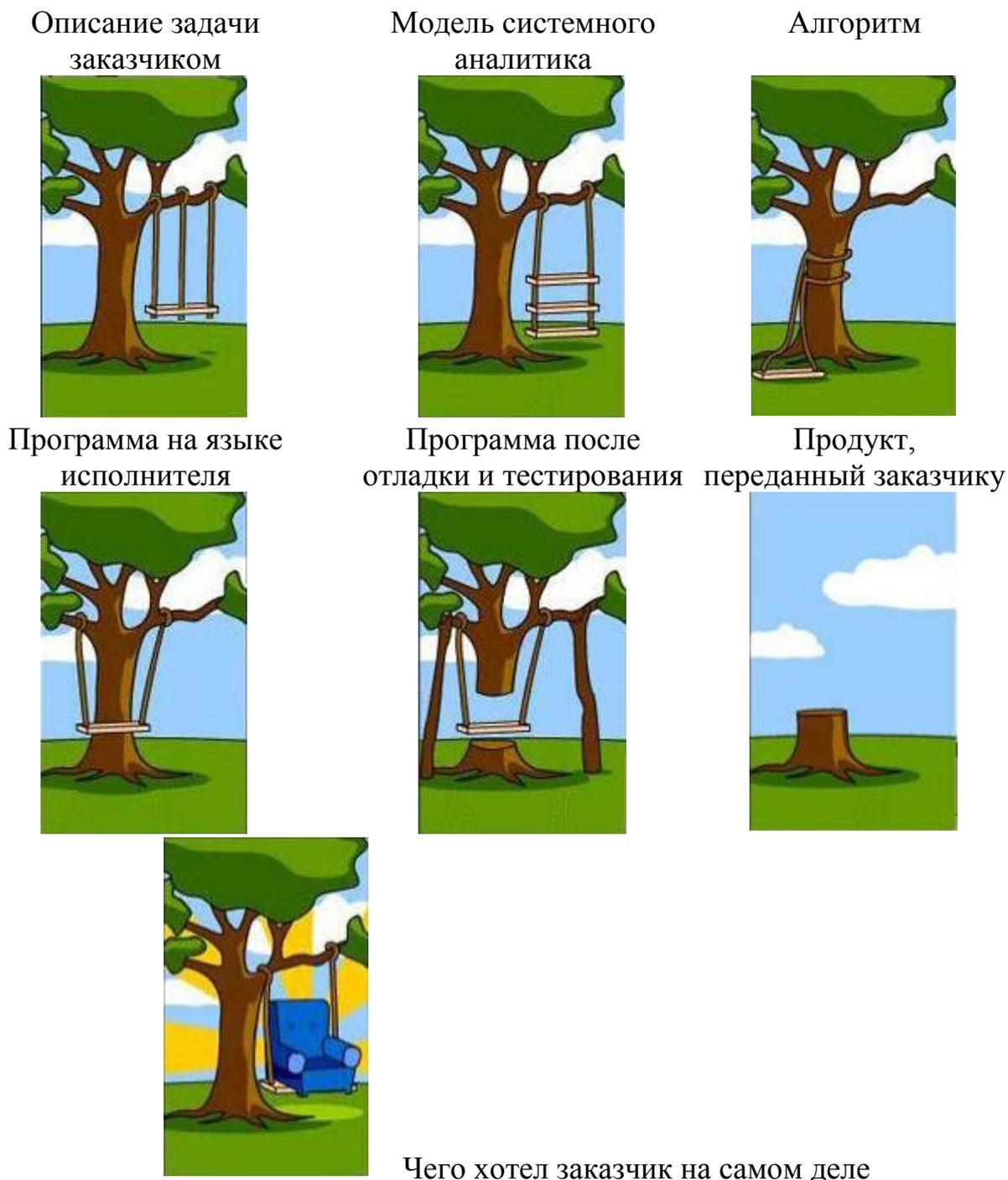


Рис. 57. Проект «Качели»

Существуют дополнительные этапы решения задач с применением компьютера. Они возникают в том случае, когда создание программ осуществляется компьютерными фирмами на заказ. К этим этапам относятся документирование, сопровождение, а также расширенное и повторное проектирование.

Документирование. На этом этапе разрабатывается программная документация, которая дает возможность тем, кто должен использовать программу, разобраться в ее работе с целью расширения возможностей программы для других вариантов использования. Для документирования широко используются такие средства, как блок-схемы, программные комментарии и карты памяти.

Сопровождение. На этом этапе осуществляется корректировка программы при изменении условий или места ее использования. Надлежащие средства тестирования и хорошая документация должны существенно уменьшить частоту и трудоемкость операций сопровождения.

Расширенное и повторное проектирование. На этом этапе происходит переработка программы с тем, чтобы она могла решать задачи, не укладывающиеся в рамки исходной постановки. Разумеется, при этом проектировщики всегда стремятся использовать программы, созданные для решения ранее поставленных задач. Проектировщики ПО не должны рассматривать любую задачу совершенно изолированно от задач, которые могут возникнуть в будущем.

Каждый этап разработки и решения задачи влияет на другие этапы. Постановка задачи должна включать в себя некоторые соображения по поводу плана тестирования, стандарта документирования, методам сопровождения и возможному расширению на другие задачи. Проект программы должен содержать положения, относящиеся к отладке, тестированию и документированию. В каждый момент времени программист выполняет работы, соответствующие сразу нескольким этапам. Таким образом, программирование, отладка, тестирование, документирование нередко оказываются тесно переплетенными.

7.2. Алгоритмы и программы

Понятие алгоритма – одно из основных в программировании и информатике. Это последовательность команд, предназначенная исполнителю, в результате выполнения которой он должен решить поставленную задачу. Алгоритм должен описываться на формальном языке, исключающем неоднозначность толкования. Исполнитель может быть человеком или машиной. Исполнитель должен уметь выполнять все команды, составляющие алгоритм. Множество возможных команд конечно и изначально строго задано. Действия, выполняемые по этим командам, называются элементарными.

Алгоритм – это строгая конечная последовательность действий, инструкций для машины (либо для человека), выполнение которых обязательно приводит к решению поставленной задачи.

В соответствии с международным стандартом ISO/IEC 2382-1 *алгоритм – конечный упорядоченный набор четко определенных правил для решения проблемы.*

Запись алгоритма на формальном языке называется программой. Иногда само понятие алгоритма отождествляется с его записью, так что слова алгоритм и программа – почти синонимы. Небольшое различие заключается в том, что под алгоритмом, как правило, понимают основную идею его построения. Программа же всегда связана с записью алгоритма на конкретном формальном языке.

Алгоритмы можно описывать человеческим языком – словами. Так и в математике – все теоремы и утверждения можно записывать без специальных обозначений. Но специальный, формальный язык записи утверждений, сильно облегчает жизнь математикам: исчезает неоднозначность, появляются краткость и ясность изложения. Всё это позволяет математикам говорить и писать на одном языке и лучше понимать друг друга.

Разницу между программой и алгоритмом можно пояснить следующим образом. Алгоритм – это метод, схема решения какой-то задачи. А программа – это конкретная реализация алгоритма, которая может быть скомпилирована и выполнена на компьютере. Алгоритм, в свою очередь, является реализацией идеи решения и предшествует программе.

Любой алгоритм должен обладать следующими **свойствами**:

1. **Понятность** – язык описания и способы реализации алгоритма должны быть понятны его исполнителю.

Исполнитель алгоритма – это некоторая абстрактная или реальная (техническая, биологическая или биотехническая) система, способная выполнить действия, предписываемые алгоритмом.

У каждого исполнителя – свой интеллектуальный уровень и своя система команд. После получения команды исполнитель должен совершить адекватное действие.

2. **Дискретность** – процесс решения задачи должен быть разбит на достаточно простые операции (этапы), которые исполнитель способен последовательно реализовать.

Одним из проявлений таланта американского автопромышленника Генри Форда стало внедрение конвейера на сборке автомобиля. Технологические процессы, выполняемые высококвалифицированными специалистами с невысокой скоростью, Форд разбил на последовательные элементарные операции, которые могли выполнять «дешевые» рабочие, а позднее – автоматы. Это позволило «Ford Motor Company» значительно увеличить производительность труда, снизить себестоимость продукции и долгое время под лозунгом «автомобиль для всех» лидировать на авторынке США.

3. **Определенность** (детерминированность) – каждое правило алгоритма должно быть четким, исключая неоднозначное толкование.

Выполнение инструкций алгоритма должно носить «механический» характер и не требовать дополнительных данных в виде указаний, уточнений, разъяснений или иных сведений о решаемой задаче.

Например, инструкцию «*Ввести максимальное натуральное число*» выполнить невозможно.

4. **Конечность** (результативность) – алгоритм должен приводить к решению задачи за конечное число шагов либо в нем должно быть предусмотрено завершение после выполнения заданного условия (числа шагов, прошедшего времени, количества циклов и т.д.), с выдачей промежуточных результатов.

Например, попробуйте выполнить инструкцию: «*Вперед шагом марш*» без дополнительных ограничений.

5. **Массовость** – алгоритм решения задачи должен выполняться для любого набора допустимых исходных данных.

Например, построение конкретного взвода в шеренгу по росту, где конкретный курсант Петров встает слева от конкретного курсанта Иванова не является алгоритмом. Однако это не означает, что данная задача не подлежит алгоритмизации для абстрактных курсантов.

Алгоритмический язык – искусственный язык, предназначенный для выражения алгоритмов.

Основные формы представления алгоритмов

В настоящее время существует множество способов задания алгоритмов, определяемых в зависимости от того, для какого исполнителя предназначен алгоритм. Таким образом, можно выделить четыре основные формы представления алгоритма:

- словесная – описание последовательности действий на естественном языке;
- формульно-словесная – задание инструкций о выполнении действий с использованием математических символов в сочетании со словесными пояснениями;
- графическая – блок-схема;
- алгоритмический язык – программа на алгоритмическом языке, например, на Бейсике.

При словесном способе записи алгоритм, как правило, задается в произвольном изложении на естественном языке, что вызывает неудобство при его последующем кодировании для выполнения на компьютере. К недостаткам словесных описаний требуемых действий можно отнести то, что они:

- трудноформализуемы;
- многословны (содержат избыточную информацию);
- допускают неоднозначность толкования.

Графическое представление в виде блок-схем в нашей стране регламентируется ГОСТ 19.701-90 (ИСО 5807-85) «Единая система программной документации. Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила выполнения». При графическом представлении алгоритм изображается в виде последовательности связанных между собой функциональных блоков, каждый из которых соответствует выполнению одного или нескольких действий.

Команды алгоритма, предписывающие выполнить заданные действия, записываются внутри символов. В схеме алгоритма символы соединяются линиями, показывающими очередность выполнения команд алгоритма.

Для изображения символов и построения схем алгоритмов установлены следующие стандарты:

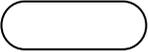
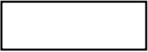
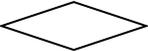
	Блок начала/конца алгоритма
	Блок ввода/вывода информации
	Блок действия
	Блок условия
	Блок переноса

Рис. 58. Основные функциональные блоки алгоритмов

Логическая структура любого алгоритма может быть представлена комбинацией трех базовых структур:

- следование,
- ветвление,
- цикл.

Характерной особенностью базовых структур является наличие в них одного входа и одного выхода.

Базовая алгоритмическая структура «следование» образуется рядом действий, следующих одно за другим:

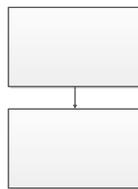


Рис. 59. Базовая алгоритмическая структура «следование»

Базовая алгоритмическая структура «ветвление» обеспечивает в зависимости от результата проверки условия (да или нет) выбор одного из альтернативных путей работы алгоритма. Каждый из путей ведет к общему выходу, так что работа алгоритма будет продолжаться независимо от того, какой путь будет выбран.

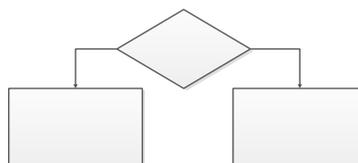


Рис. 60. Базовая алгоритмическая структура «ветвление»

Базовая алгоритмическая структура «цикл» обеспечивает многократное выполнение некоторой совокупности действий, которая называется телом цикла.

В соответствии со свойством конечности любой цикл должен завершаться либо при выполнении какого-либо условия, либо при обнулении счетчика заданных повторений цикла.

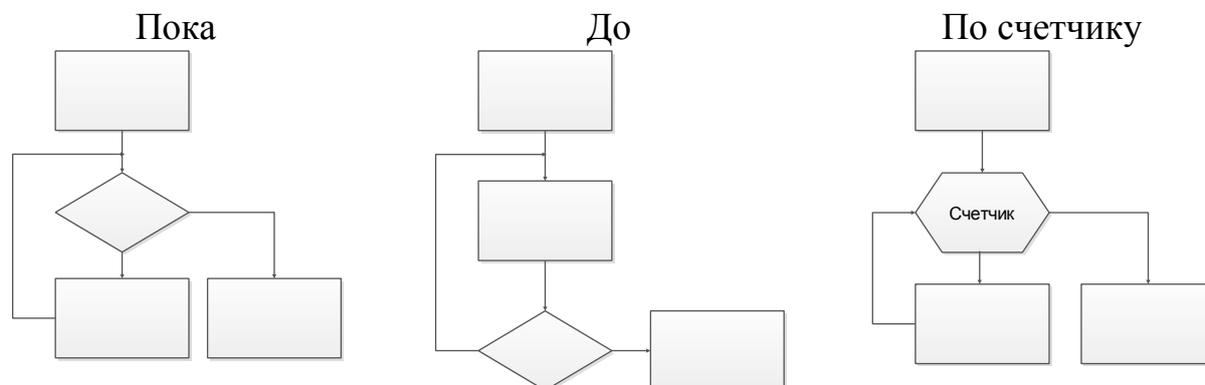


Рис. 61. Основные циклические структуры

Существуют разные способы представления одного и того же алгоритма.

Например, по результатам четырех выстрелов положение средней точки попадания можно определить следующим образом:

- соединить двумя отрезками пары близколежащих пробоин;
- найти середины получившихся отрезков и соединить их.

Средняя точка попадания будет в середине нового отрезка.

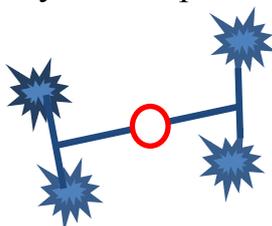


Рис. 62. Определение средней точки попадания по четырем выстрелам

Чтобы компьютер в автоматическом режиме решил поставленную задачу, алгоритм необходимо представить в двоичном коде, разместить в ячейках системной памяти и загрузить в процессор первую инструкцию.

Алгоритм, представленный на языке компьютера, называется **компьютерной программой**.

В соответствии с ГОСТом 28297-89 «Языки программирования. Термины и определения» **язык программирования (ЯП)** – язык, предназначенный для представления программ.

Вплоть до третьего поколения программы для ЭВМ писались непосредственно в двоичных кодах, не требующих дополнительного кодирования.

Каждая элементарная операция алгоритма (или иначе – машинная инструкция) кодировалась **машинным словом** – последовательностью битов, одна часть которых составляла **код операции**, а другая – **операнды**, т.е. указание на данные, над которыми выполняется операция.

1	1	1	0	1	1	1	1	1	1	1	0	0	0	0	0
код операции				операнд 1						операнд 2					

Рис. 63. Пример двухбайтовой машинной инструкции

Язык программирования, предназначенный для представления программ в форме, позволяющей выполнять ее непосредственно техническими средствами обработки информации, – **машинный язык**¹ (МЯ).

Программа на машинном языке выполняется «как есть», без дополнительного преобразования, но разрабатывать программу на машинном языке достаточно сложно.

При создании нового компьютера разработчики решают, какие инструкции включить в машинный язык, и стараются сделать их как можно проще, чтобы упростить реализующие их аппаратные схемы.

Машинные слова-инструкции обычно выполняют операции типа:

- сложить 2 числа;
- сдвинуть число вправо на 1 разряд;
- инвертировать число;
- проверить, не является ли число нулем;
- увеличить число на 1 и т.д.

Языки высокого уровня (ЯВУ) – языки программирования, понятия и структура которых удобны для восприятия человеком. ЯВУ состоят из множества лексических конструкций, условных знаков, синтаксических и семантических правил для точного описания программ.

Между МЯ и ЯВУ существуют промежуточные языки программирования, называемые ассемблерами – машинно-ориентированные языки, описывающие машинные команды в словесных мнемосокодах.

Ассемблеры являются машинно-ориентированными, т.е. они настроены на структуру машинных команд конкретного компьютера. Процессоры с разной архитектурой имеют разные ассемблеры.

Процесс перевода программы с ЯВУ на МЯ называется **трансляцией**, а программное или техническое средство, выполняющее трансляцию программы, – **транслятор**.

В настоящее время существуют два типа трансляторов: **компиляторы и интерпретаторы**.

При компиляции программа однократно перекодирована с ЯВУ на МЯ, и затем, как правило, исполняемый файл сохраняется в двоичном (машинном) формате (например, с расширениями .EXE или .COM) на внешнем носителе. Чтобы выполнить программу, достаточно «запустить» этот файл.

¹ ГОСТ 15971–90. Системы обработки информации. Термины и определения: утв. Постановлением Госстандарта СССР от 26.10.1990 № 2698 // Режим доступа: СПС «Консультант-Плюс».

В отличие от компиляции при интерпретации не создается новая программа, а происходит последовательная перекодировка отдельных инструкций с ЯВУ на МЯ и их немедленное выполнение, как синхронный перевод фильма. Примером ручного ввода и выполнения команд в режиме интерпретатора может служить командная строка MS Windows, а автоматического – командные файлы MS Windows (с расширением .BAT) или WEB-страницы (.HTML).

В соответствии с ГОСТом 19781-90 **система программирования** – это система, образуемая языком программирования, компиляторами или интерпретаторами программ, представленных на этом языке, соответствующей документацией, а также вспомогательными средствами для подготовки программ к форме, пригодной для выполнения.

В идеале компьютер можно сконструировать так, чтобы он сразу выполнял команды или целые программы на языках высокого уровня в режиме интерпретатора – например, компьютер с машинным языком типа BASIC. Любая операция, выполняемая программным обеспечением, может быть реализована аппаратно, но тогда значительно увеличится стоимость аппаратной части, и компьютер будет «привязан» к одному языку. В качестве примера можно привести программируемые калькуляторы, применявшиеся в 70-80-х годах прошлого века.

Разделение функций аппаратного и программного обеспечения определяется такими факторами, как стоимость, производительность, надежность компьютерной системы и т.д. Например, в автоматизированных системах управления, требующих немедленного реагирования на внешние воздействия (авиация, АЭС), выполнение программных инструкций целесообразно реализовать аппаратно без дополнительной трансляции, а игру «Sonic» с Sega Mega Drive можно запускать в MS Windows в режиме программной эмуляции.

BIOS¹ в IBMPC архитектуре – это часть аппаратного или программного обеспечения персонального компьютера? Граница между аппаратной и программной реализацией инструкций алгоритма постоянно смещается. Сегодняшнее программное обеспечение может быть завтрашним аппаратным обеспечением и наоборот. Пользователю не важно, как на самом деле выполняется программа.

Карен Панетта Ленц, доктор в области инженерии компьютерных систем, образно говорила: «Аппаратное обеспечение – это всего лишь окаменевшее программное обеспечение».

7.3. Программирование на компьютере

Первыми популярными машинно-независимыми **языками высокого уровня**, появившимися в 1950-х гг., были Фортран, Кобол (в США) и Алгол (в Европе). Языки Фортран и Алгол были предметно ориентированы на научно-технические расчеты математического характера. Кобол – язык для решения

¹ Базовая система ввода-вывода (англ. Basic input/output system).

экономических задач. В Коболе хорошо развиты средства обработки текстов, включая различные форматы печатных документов.

На больших ЭВМ третьего поколения использовался язык PL-1 (Programming Language One), разработанный фирмой IBM. Это был первый язык, претендовавший на универсальность, т.е. на возможность решать любые задачи: вычислительные, обработки текстов, накопления и поиска информации. Однако он оказался слишком сложным, и транслятор PL-1 для машин семейства IBM 360/370 содержал ряд ошибок. На ЭВМ класса мини- и микро- он вообще не получил распространения. Однако тенденция к универсализации языков оказалась перспективной. Старые языки были модернизированы в универсальные варианты – Алгол-68 и Фортран-77.

Значительным событием в программировании стало создание швейцарским профессором Николаусом Виртом языка Паскаль с концепцией структурного программирования.

Наибольший успех в распространении этого языка обеспечили персональные компьютеры. Для ПК фирма Borland International Inc (США) разработала первую систему программирования – Турбо Паскаль, включающую не только язык и транслятор, но еще и операционную оболочку. Турбо Паскаль вышел за рамки учебного предназначения и стал языком профессионального программирования с универсальными возможностями – с наиболее оптимальным транслятором. Впоследствии Паскаль стал основой многих современных языков программирования: Ада, Модула-2, Delphi и др.

Язык программирования Си (английское название – С) создавался как инструментальный язык для разработки операционных систем, трансляторов, баз данных и других системных и прикладных программ. Так же как и Паскаль, Си – это язык структурного программирования, но, в отличие от Паскаля, в нем заложены возможности непосредственного обращения к некоторым машинным командам, к определенным участкам памяти компьютера. Дальнейшее развитие Си привело к созданию объектно-ориентированных языков С++ и С#.

Первые языки программирования высокого уровня были **процедурными**, и программы состояли из команд, последовательно кодирующих выражения и инструкции алгоритмов обработки данных. Основное внимание было уделено обработке – алгоритму, который необходим для выполнения требуемых вычислений.

Позднее в некоторых процедурных языках появилась возможность описывать и вызывать в программе типовые функции обработки данных с заданными параметрами. Метод разработки программы как цепочки последовательно вычисляемых функций называется **функциональным программированием**.

В соответствии с ГОСТ 19781-90¹ **структурное программирование**² – это метод построения программ, использующий только иерархически вложенные конструкции, каждая из которых имеет единственную точку входа и един-

¹ ГОСТ 19781-90. Государственный стандарт Союза ССР. Обеспечение систем обработки информации программное. Термины и определения: утв. и введен в действие Постановлением Госстандарта СССР от 27.08.1990 № 2467. – М.: ИПК Издательство стандартов, 1998.

² Методология структурного программирования предложена Э. Дейкстрой в 70-х годах прошлого века и позднее усовершенствована Н. Виртом.

ственную точку выхода. В программе используются всего три вида алгоритмических структур:

- последовательная,
- условного перехода,
- циклическая.

Основными принципами структурного программирования являются:

- пошаговая детализация,
- модульная организация программы.

Следование принципам структурного программирования позволило сделать тексты больших программ удобочитаемыми. Появилась возможность коллективной разработки сложных программ, разбивая их на отдельные, относительно автономные модули.

Язык программирования определяет набор лексических, синтаксических и семантических описаний и правил (спецификаций), определяющих структуру и вид программы, а также реакцию компьютера на ее управляющие коды.

Лексические правила определяют словарный состав языка.

Семантика отвечает за смысловое значение единиц языка.

Основное назначение **синтаксических правил** (правописания) – придать однозначный смысл языковым конструкциям.

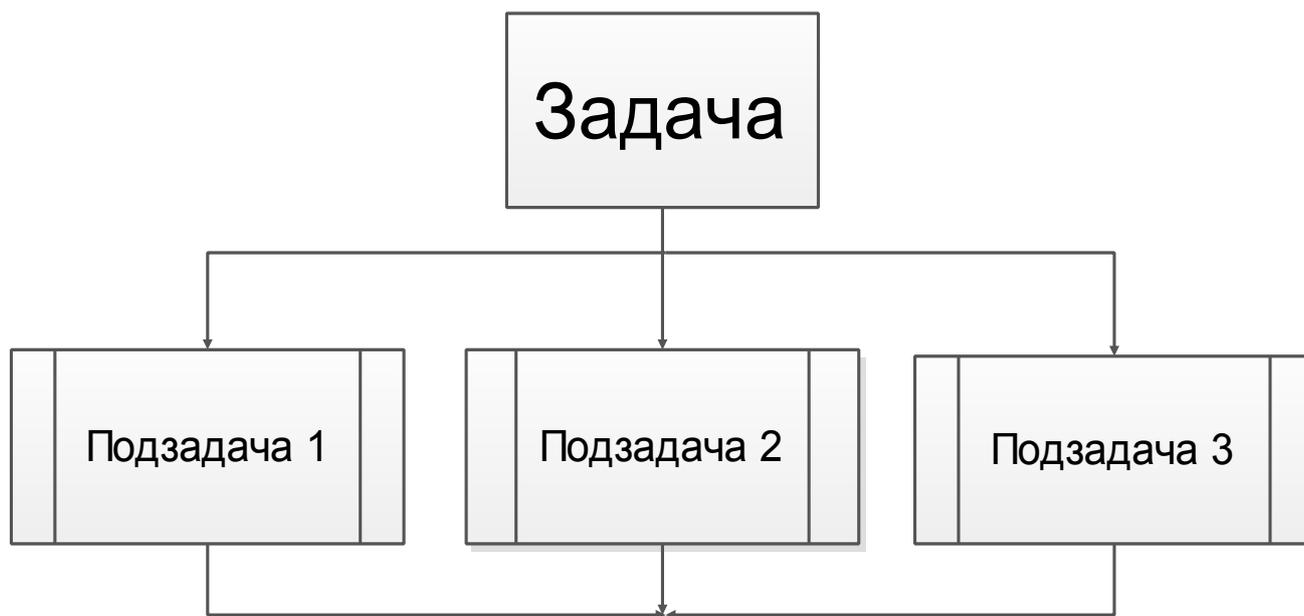


Рис. 64. Функциональная декомпозиция программы

В общем виде спецификации языка программирования должны соответствовать следующей структурной схеме:

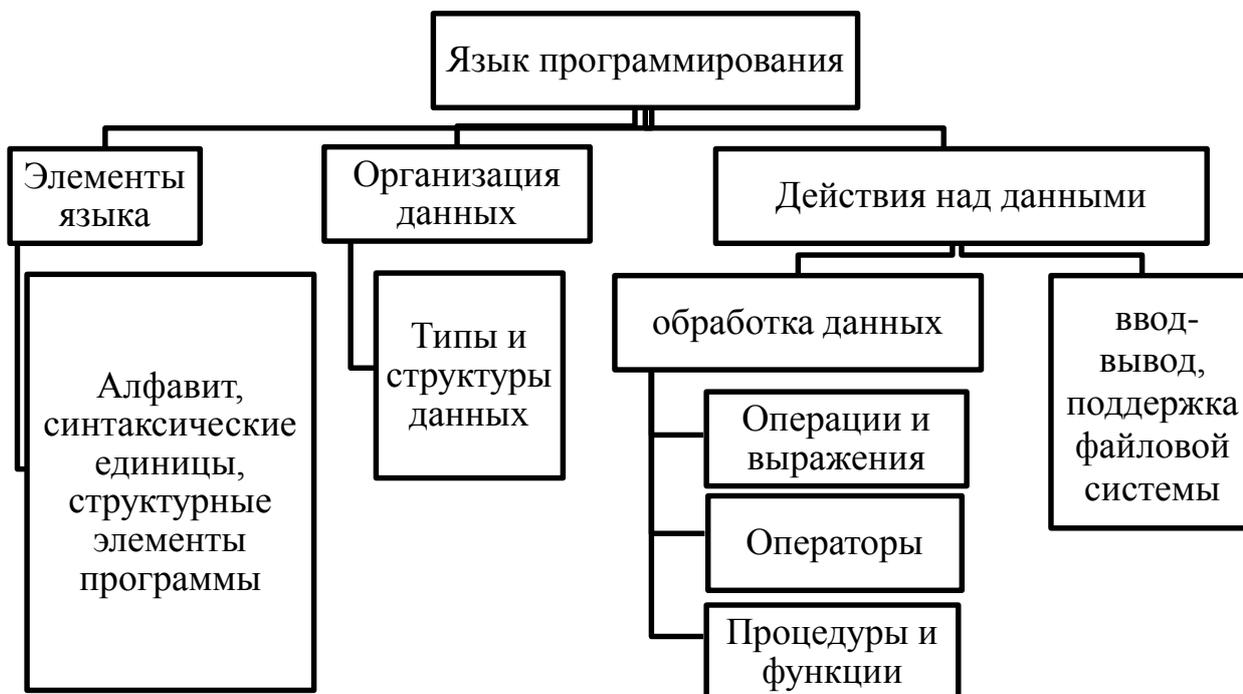


Рис. 65. Структурная схема языка программирования

Любая программа обрабатывает данные – информацию, представленную в различных форматах. Перед непосредственной обработкой любые данные размещаются в оперативной памяти. Например, в байтовой ячейке памяти по адресу FFF0 хранится код:

FFF0>	1	1	1	0	0	0	0	0
-------	---	---	---	---	---	---	---	---

Это может быть:

- десятичное число без знака 224;
- десятичное число со знаком -96;
- буква «а» в кодировке Windows CP 1251;
- глубина цвета пикселя и др.

Для того чтобы программа правильно интерпретировала кодовое слово, в ее начале описывают типы данных, которые определяют:

- объем и структурную организацию блока памяти, выделяемого для хранения значений;
- диапазон возможных значений;
- множество допустимых операций, применимых к этим данным.

Элементарное действие, выполняемое над данными, в программировании называется **операцией**.

В основных языках программирования выделяют четыре типа операций:

- арифметические (+, -, *, \, /, ^, mod);
- логические (not, and, or, xor, eqv, imp);
- строковые (+, &);
- операции отношения (=, <>, <, >, <=, >=, Is, Like).

Операции выполняются над **операндами**.

Переменная – это поименованный элемент данных, значение которого может присваиваться и изменяться в программе в пределах объявленного типа.

Совокупность однотипных индексированных переменных называется **массивом**. Массив имеет размерность. Двумерный массив, например, в Excel, – это таблица. Трехмерному можно сопоставить рабочую книгу.

И тогда, запись в ячейку *B1* формулы *=Лист1!A1* будет означать присваивание *B1* значение ячейки, расположенной на пересечении столбца *A* и строки *1* листа *Лист1*.

Константа – это элемент данных, не меняющий своего значения.

Например, если в ячейку *A1* таблицы Excel ввести *Мос*, а в ячейку *B1* ввести формулу *=A1&"У"*, то на языке программирования это будет означать: *B1=A1&"У"* и *B1* будет присвоено текстовое значение *МосУ*. В данном примере *A1* – это переменная, а *"У"* – константа. **&** – строковый оператор.

В свою очередь, *B1=A1&"У"* является **выражением**, или **формулой**.

Объектно-ориентированное программирование (ООП) – метод построения программ как совокупностей объектов и классов объектов, которые могут вызывать друг друга для выбора и выполнения операций.

Основное отличие процедурного программирования от объектно-ориентированного в том, что первое ориентировано на моделирование и выполнение алгоритмических шагов, а последнее – на описание и воспроизведение предметной области решаемой задачи. Методы структурного программирования позволяют фиксировать поведение объектов реального мира таким способом, при котором детали реализации скрыты.

С появлением ООП появилась новая структура данных – **класс**.

Класс представляет собой тип данных, имеющий в составе:

- свойства или значимые атрибуты объекта;
- методы или возможные действия с объектом.

Основными принципами объектно-ориентированного программирования являются:

- инкапсуляция (любой класс рассматривается как «чёрный ящик»);
- абстрагирование (объекты являются моделями реальных сущностей из предметной области);
- наследование (классовая иерархия подразумевает общие свойства и методы у класса-предка и класса-потомка);
- полиморфизм (классы-потомки могут изменять реализацию унаследованных методов класса-предка, сохраняя неизменным его интерфейс).

Например, сравните объекты-фигуры (Вставка \Rightarrow Фигуры) и методы, которые с ними связаны.

Объектно-ориентированное программирование используется в языках: Simula 67, Smalltalk, Lisp, Clu, Actor, Eiffel, Objective C, C++, Java.

Эволюция языка высокого уровня хорошо прослеживается на примере языка BASIC, который опытные программисты не воспринимают всерьез. Наибольшее распространение Бейсик получил на микро-ЭВМ. Само название

BASIC¹ предполагает, что это непрофессиональный язык и, действительно, этот язык был разработан в начале 1960-х годов в Дартмутском университете и задумывался как наглядное средство преподавания методов программирования студентам колледжей. Во многих ранних версиях он был интерпретируемым языком. Каждая строка перед выполнением интерпретировалась, чем и была обусловлена низкая скорость обработки кода. В большинстве современных вариантов языка BASIC программы компилируются и выполняются значительно быстрее, их переносимость на другие аппаратные платформы улучшилась.

В 1991 году популярность BASIC возросла при поддержке компании Microsoft, которая выпустила свою версию Visual Basic для Windows. Этот продукт значительно упростил разработку пользовательских приложений для персональных компьютеров. Visual Basic представляет собой основу VBA (Visual Basic for Applications) – языка написания сценариев для приложений Microsoft и в настоящее время входит в состав всех приложений MS Office, а также ряда других производителей.

Секрет популярности использования VBA заключается в грамотной интерпретации объектной модели и управлении объектами в каждом отдельном приложении.

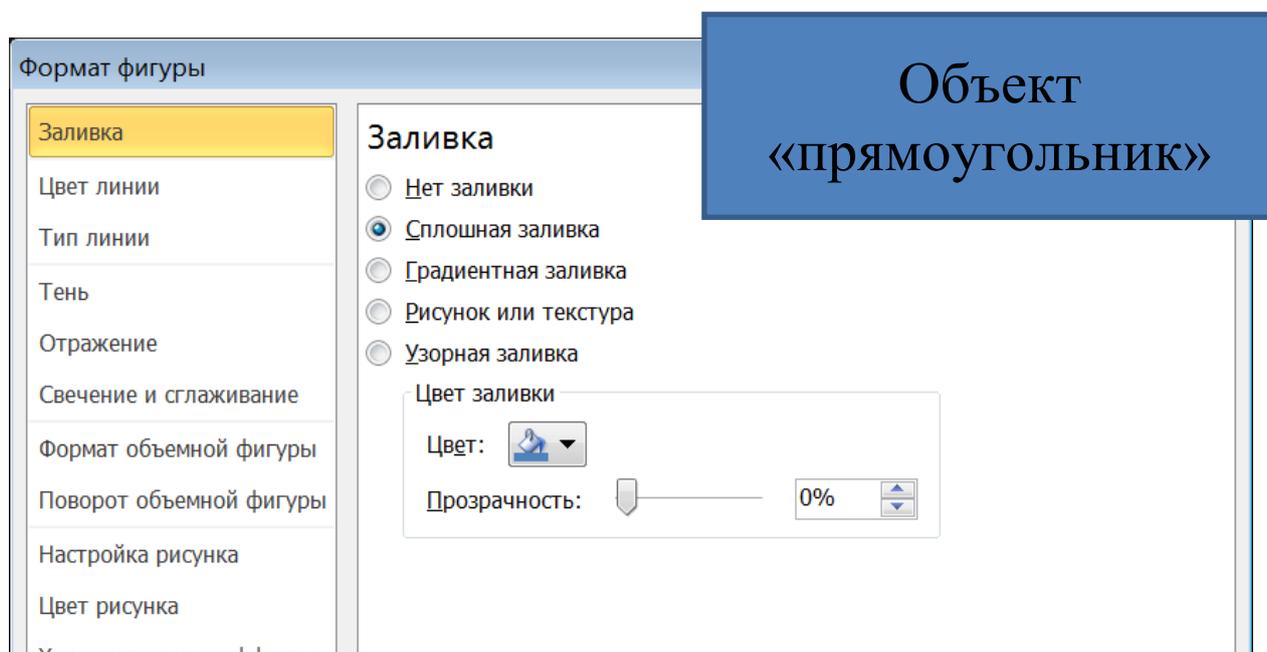


Рис. 65. Свойства класса «Прямоугольник» в MS Word

Так, в объектной модели Word объектами форматирования текста являются страница, абзац, шрифт и т.д., а в Excel представлены мощные объекты для анализа данных: рабочие листы, диаграммы, сводные таблицы, сценарии, а также многочисленные математические, финансовые, инженерные и общие функции.

¹ Универсальный символический язык инструкций для начинающих (англ. Beginner's All-purpose Symbolic Instruction Code).

Логическое программирование – метод построения программ как совокупности логических правил с предварительно определенными алгоритмами для обработки входных данных программы в соответствии с ее правилами.

Первым языком логического программирования был язык Planner, в котором была заложена возможность автоматического вывода результата из данных и заданных правил перебора вариантов (совокупность которых называлась планом). Затем был разработан язык Prolog, который не требовал плана перебора вариантов и был в этом смысле упрощением языка Planner.

Языки ЛИСП и Пролог – прототипы языков «искусственного интеллекта».

ЛИСП появился в 1965 г. Язык ЛИСП основан на понятии рекурсивно определенных функций. А поскольку доказано, что любой алгоритм может быть описан с помощью некоторого набора рекурсивных функций, то ЛИСП, по сути, является универсальным языком. С его помощью на ЭВМ можно моделировать достаточно сложные процессы, в частности, интеллектуальную деятельность людей.

Язык Пролог разработан во Франции в 1972 г. также для решения проблемы «искусственного интеллекта». Пролог позволяет в формальном виде описывать различные утверждения, логику рассуждений и получать от компьютера ответы на поставленные вопросы.

Вопросы для самоконтроля:

1. Перечислите и охарактеризуйте типовые этапы решения задачи с помощью компьютера.
2. Дайте определение алгоритма.
3. Перечислите и охарактеризуйте свойства алгоритма.
4. Дайте определение понятию алгоритмического языка.
5. Перечислите и охарактеризуйте основные формы представления алгоритмов.
6. Как может быть представлена логическая структура любого алгоритма?
7. Дайте определение понятию компьютерной программы.
8. Какие операции обычно выполняют машинные слова-инструкции?
9. Что такое транслятор?
10. Дайте определение понятию системы программирования.
11. Что такое структурное программирование?
12. Что такое объектно-ориентированное программирование?

8. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ СОВРЕМЕННЫХ КОМПЬЮТЕРОВ

8.1. Основные понятия, назначение и классификация программного обеспечения

В основу функционирования компьютера положен **программный принцип управления**, т.е. компьютер выполняет только операции, находящиеся в оперативной памяти, которые записаны в виде команд в программах.

Программы служат интерфейсом между аппаратными ресурсами ЭВМ и пользователем - человеком или периферийным устройством.

Понятию **интерфейс** можно дать общее определение – это средства взаимодействия, средства связи, сопряжения, согласования.

Всего различают три вида интерфейса:

1. Физический интерфейс – взаимосвязь на уровне электронных компонентов.
2. Интерфейс программиста – комплекс правил и соглашений о стыковке программ.
3. Интерфейс пользователя – набор средств диалога, взаимодействия программы (компьютера) с человеком. Такой диалог обеспечивается через основные элементы интерфейса: меню и окно.

Схема современного интерфейса пользователя с аппаратно программной средой компьютера можно представить следующей схемой:

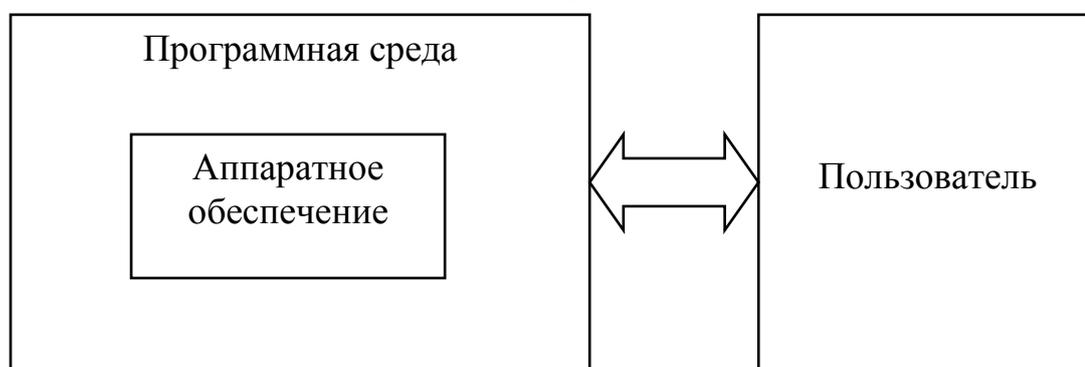


Рис. 66. Интерфейс пользователя с аппаратно-программной средой компьютера

Программное обеспечение ЭВМ является оболочкой ее аппаратной среды и обеспечивает интерфейс с пользователем.

Программное обеспечение (ПО) – составляющая информационных технологий, включающая компьютерные программы и данные, предназначенные для решения определённого круга задач и хранящиеся на машинных носителях.

Программное обеспечение представляет собой либо данные для использования в других программах либо алгоритм, реализованный в виде последовательности инструкций для процессора.

В компьютерном жаргоне часто используется слово «софт» (англ. software). Данное слово впервые в этом смысле применил математик из Принстонского университета Джон Тьюки в статье *American Mathematical Monthly* в 1958 г. В области вычислительной техники и программирования программное обеспечение – это совокупность всей информации, данных и программ, которые обрабатываются компьютерными системами.

Все программное обеспечение для компьютеров может быть представлено тремя видами программ: системным программным обеспечением, средствами для создания приложений, а также прикладным программным обеспечением.



Рис. 67. Классификация программного обеспечения

Прикладное программное обеспечение – программы, предназначенные для выполнения определенных пользовательских задач и рассчитанные на непосредственное взаимодействие с пользователем.

Инструментальное программное обеспечение – программное обеспечение, предназначенное для использования в ходе проектирования, разработки и сопровождения программ. В ходе изучения данного курса вы не будете изучать инструментальное программное обеспечение, которым пользуются только специалисты в области информационных технологий.

Системное программное обеспечение – это набор программ, которые управляют компонентами вычислительной системы, такими как процессор, коммуникационные и периферийные устройства, а также которые предназначены для обеспечения функционирования и работоспособности всей системы. В отличие от прикладного, системное программное обеспечение (операционная система) используется для обеспечения работы компьютера самого по себе и выполнения прикладных программ.

В соответствии с ГОСТом 19781-90 «Единая система программной документации. Обеспечение систем обработки информации программное» **программное обеспечение** – совокупность программ системы обработки информации и программных документов, необходимых для эксплуатации этих программ.

В соответствии с ГОСТ Р ИСО/МЭК 9126-93 «Оценка программной продукции. Характеристики качества и руководства по их применению» **программное обеспечение (software)** – программы, процедуры, правила и любая соответствующая документация, относящиеся к работе вычислительной системы.

Программная продукция (software product) – программный объект, предназначенный для поставки пользователю.

Для работы на современных персональных компьютерах с системными или прикладными программами не требуется специальных навыков программиста и знаний языков программирования. Программное обеспечение (ПО) персональных компьютеров (ПК) можно разделить на:

- **системное** – обеспечивает автоматическую загрузку и функционирование ПК, а также общее управление компьютером и профилактическое обслуживание;

- **прикладное** – обеспечивает решение пользовательских задач.

Для задач программирования – т.е. разработки системных и прикладных компьютерных программ существует отдельный вид ПО – **инструментальное ПО**.

В связи с повсеместным использованием технологий графических интерфейсов (visual-технологии) и объектного программирования граница между прикладным и инструментальным программным обеспечением практически стерта. Например, к какому виду ПО можно отнести пакет MSOffice 2010, в который, помимо общеизвестных прикладных возможностей, включены средства поддержки языка программирования VBA (Visual Basic for Applications) или создания WEB-страниц на языке HTML или пакет 3-мерного графического моделирования Autodesk 3d Studio Max.

Программное обеспечение является объектом авторского права и охраняется законодательством Российской Федерации как литературные произведения (Гражданский кодекс РФ от 18.12.2006 № 230-ФЗ, часть 4).

Ниже приведены некоторые **типы лицензий** на программное обеспечение:

Freeware – бесплатная программа без функциональных ограничений. Сфера использования регулируется авторским правом. Владелец программы может запретить ее свободное распространение или коммерческое использование. Например, программа может быть бесплатна только для дома или для учебного процесса при условии некоммерческого использования.

Shareware – условно-бесплатная программа. Коммерческая программа с бесплатным периодом использования и (или) ограниченным функционированием. Предлагается сначала опробовать программу в действии, а затем оплатить ее полную стоимость. «Условность» может лежать в очень широких границах, от периодического напоминания о необходимости заплатить за программу при каждом запуске до блокирования основных функций, ограничивающих использование программы по прямому назначению.

Trial, trial ware – пробная (оценочная) версия программы. Ограничена временем использования или блокированием отдельных функций.

Demo, demo ware – ограниченная версия программы, демонстрирующая ее основные возможности.

Adware – бесплатная рекламно-ориентированная программа без функциональных ограничений, но со встроенными окнами для показа рекламы, которая может подгружаться через Интернет. Модуль фоновой загрузки рекламы без уведомления пользователя фактически является «трояном».

Donate ware – программа, не требующая обязательной регистрации и оплаты для нормального функционирования, но ненавязчиво предлагающая оказать какую-либо услугу автору, в том числе и денежное пожертвование.

Open source – бесплатное программное обеспечение с «открытым» исходным кодом, т.е. возможностью его модифицировать без согласования с автором. Используется «как есть» (as is) без каких-либо авторских гарантий.

Beta – некоммерческая версия программы для тестирования ее возможностей, как правило, ограниченным кругом пользователей. В процессе создания программа проходит несколько этапов, характеризующихся ее готовностью к окончательному выпуску: Alpha, Beta, RC (Release Candidate).

На стыке программного и аппаратного обеспечения есть еще **firmware (программируемое оборудование)** – управляющая программа для однократной записи в энергонезависимую память оборудования («прошивки»). Пользователь получает систему с установленным на заводе Firmware и не задумывается о нем (хотя многие устройства допускают обновление «прошивок» новыми версиями). Например, BIOS на материнской плате ПК или операционная система на сотовых телефонах.

В соответствии с ГОСТом 19781-90 **прикладная программа** – программа, предназначенная для решения задачи или класса задач в определенной области применения системы обработки информации. Совокупность прикладных программ, предназначенная для решения задач определенного класса, называется **пакетом прикладных программ**.

Прикладные программы можно классифицировать, например, **по типу:**

- **общего назначения** (текстовые, графические, аудио-, видеоредакторы, аудио-, видеоплееры, системы управления базами данных, электронные таблицы, веб-браузеры и т.п.);
- **профессионально ориентированные** (экспертные системы, системы автоматизированного проектирования, обеспечение АРМ, АСУ, АСУ ТП, геоинформационные системы);
- **культурно-развивающие** (мультимедиа энциклопедии, справочники, игры и т.п.);
- **контрольно-обучающие** (электронные курсы, тесты, симуляторы и т.п.).

В органах внутренних дел прикладное программное обеспечение входит в состав автоматизированных рабочих мест или автоматизированных информационных систем по направлениям профессиональной деятельности.

8.2. Системное программное обеспечение и операционные системы

После загрузки операционной системы пользователь и другие программы получают возможность управления компьютером, т.е. возможность непосредственно решать прикладные задачи: производить вычисления, готовить документы и выводить на печать, моделировать различные процессы, играть в игры и т.д. – т.е. использовать компьютер по его прямому назначению – как средство автоматизации человеческой деятельности.

Системная программа – программа, предназначенная для поддержания работоспособности системы обработки информации или повышения эффективности ее использования в процессе выполнения прикладных программ.

Управляющая программа – системная программа, реализующая набор функций управления, в который включают управление ресурсами и взаимодействием с внешней средой системы обработки информации, восстановление работы системы после проявления неисправностей в технических средствах.

Супервизор (ядро) – часть управляющей программы, координирующая распределение ресурсов системы обработки информации.

Программа обслуживания – программа, предназначенная для оказания услуг общего характера пользователям и обслуживающему персоналу системы обработки информации.

В настоящее время терминология естественно изменилась, но типовой состав системного ПО компьютеров с архитектурой IBMPC остался практически тем же.

BIOS – базовая система ввода-вывода размещается в специальной микросхеме на системной плате и устанавливается на заводе-изготовителе (firmware), но может быть «перепрошита» с учетом модификации исходных кодов.

На место BIOS приходит UEFI¹ – программируемый интерфейс между операционной системой и низкоуровневыми микропрограммами (прошивками) аппаратных устройств ПК. UEFI инициализирует оборудование при включении ПК и передает управление загрузчику операционной системы. В отличие BIOS, который всегда жестко прошит в соответствующем чипе на системной плате, коды UEFI находятся в специальной директории /EFI/, которая физически может располагаться и в микросхеме памяти на системной плате, и в разделе на жестком диске компьютера, и во внешнем сетевом хранилище.

В результате столь гибкого подхода UEFI является упрощенной, но вполне самостоятельной операционной системой с доступом ко всему аппаратному обеспечению компьютера, с тестовыми и рабочими сервисами оборудования, поддержкой протоколов коммуникаций, драйверами устройств, собственной графической оболочкой, из-под которой можно запускать собственные EFI-приложения. А уже поверх всего этого хозяйства расположен собственно загрузчик, отвечающий за запуск на компьютере основной операционной системы (или нескольких систем). На уровне UEFI можно, к примеру, выходить в Интернет или организовывать резервное копирование жестких дисков.

В принципе, в каждой из основных на сегодня операционных систем (Windows, OS X, Linux) уже имеется поддержка загрузки через UEFI. Но следует также отметить, что пока UEFI все еще является очень молодой системой, и реально очень немногие ОС пользуются всеми ее преимуществами, перечисленными выше.

Операционная система – совокупность системных программ, предназначенная для обеспечения определенного уровня эффективности системы обработки информации за счет автоматизированного управления ее работой и предоставляемого пользователю определенного набора услуг (ГОСТ 15971-90 «Системы обработки информации. Термины и определения»).

Основные **функции** современных ОС ПК:

- управление интерфейсами передачи данных между системными и периферийными устройствами;
- управление оперативной памятью (загрузка программ, распределение между процессами, очистка, поддержка файла подкачки – виртуальной памяти);
- обработка запросов обслуживания внешних устройств и программ (ввод и вывод данных, запуск и остановка библиотечных процедур и функций);
- поддержка файловых систем хранения данных на внешних носителях;
- обеспечение интерфейса пользователь-компьютер;
- профилактика, диагностика и восстановление работоспособности компьютера после сбоев;
- защита процессов и данных от несанкционированного доступа или воздействия;
- поддержка многозадачного и многопользовательского режима работы ПК с разграничением доступа к его ресурсам.

¹ Единый расширяемый интерфейс ПО Firmware (англ. Unified Extensible Firmware Interface).

Интерфейс – совокупность средств и правил, обеспечивающих взаимодействие устройств вычислительной машины или системы обработки информации и (или) программ.

Основные компоненты операционной системы:

1. **Загрузчик** – небольшая программа для передачи управления от BIOS к ОС, размещаемая на внешнем устройстве по определённому адресу и обеспечивающая загрузку ядра операционной системы.

2. **Ядро ОС (супервизор)** – основная часть операционной системы, управляющая аппаратными ресурсами вычислительной системы (процессор, память и устройства ввода-вывода) и предоставляющая программным процессам координированный доступ к этим ресурсам. На уровне ядра могут быть реализованы поддержка файловой системы и управление обменом по локальной вычислительной сети.

3. **Драйвер устройства** – программа, предназначенная для управления периферийным устройством, в том числе файловой системой для устройств внешней памяти. С одной стороны, драйвер поддерживает интерфейс конкретного устройства, с другой – конкретной операционной системы. Другими словами, операционная система управляет некоторым «виртуальным» устройством, которое понимает ее стандартный набор команд. Драйвер переводит эти команды в команды, которые понимает физическое устройство. Наличие драйверов позволяет разработчикам нового компьютерного оборудования «подружить» его с любой операционной системой.

4. **Средства поддержки файловых систем (ФС)** – правила и методы организации, хранения и именования данных, размещаемых на внешних носителях. ОС и ФС взаимодействуют через драйверы файловой системы.

5. **Командный процессор (интерпретатор системных команд)** – компонент операционной системы, отвечающий за выполнение отдельных системных инструкций, вводимых пользователем в специальной командной строке в текстовом формате. При необходимости выполнить заданную последовательность инструкций они могут быть сгруппированы в командный файл.

6. **Операционная оболочка** – программное средство автоматизированного управления компьютером с интуитивно понятным пользовательским интерфейсом (как правило, графическим с поддержкой манипулятора «мышь»). Служит альтернативой командному процессору для ввода и выполнения системных инструкций, в том числе файловых операций. Может входить в состав дистрибутива ОС (проводник MS Windows, рабочий стол, панель управления), а может существовать как отдельная программа независимого разработчика (Total Commander).

С 1990-х годов наиболее распространёнными операционными системами персональных компьютеров являются ОС семейства Windows и семейства UNIX (Linux и Mac OS X).

Служебные и сервисные программы – совокупность программ, автоматизирующих процессы установки и настройки различных параметров системы, пользовательского интерфейса, диагностики, профилактики, оптимизации и поддержки системы в актуальном работоспособном состоянии, восстановления после сбоев и т.д.

В большинстве случаев все необходимые функции по обслуживанию компьютера можно реализовать встроенными средствами операционной системы. В MS Windows к таким средствам можно отнести Центр обновления, Центр поддержки, Брандмауэр, Диспетчер учетных данных, Архивацию и восстановление системы и др.

Особое место занимают **средства обеспечения информационной безопасности** – системные программы обеспечения конфиденциальности, целостности и доступности обрабатываемых на компьютере данных с учетом правил разграничения доступа.

В качестве примера можно привести программу архивации данных True Image, которая позволяет создавать резервные копии операционной системы, приложений, пользовательских настроек и всех имеющихся данных, а также надежно уничтожать всю конфиденциальную информацию, ставшую ненужной. С ее помощью можно выполнять резервное копирование файлов и папок, настроек и писем почтовых клиентов Майкрософт и даже целых дисков и их разделов. Acronis Online Backup позволяет хранить самые важные файлы в удаленном хранилище. Данные будут защищены даже в случае потери, кражи или уничтожения компьютера. Непрерывная защита Acronis периодически (каждые пять минут) сохраняет изменения, произошедшие в системе и файлах, что, при необходимости, дает возможность легко вернуться к состоянию на любой момент времени. Все это позволяет легко и быстро восстановить данные из резервных копий в случае отказа жесткого диска, вирусной атаки или атаки вредоносного программного обеспечения.

Широкое применение у пользователей имеют программы поддержки оптических дисков (создание образов, запись, диагностика), например, от производителей Nero или Ashampoo.

Ни один компьютер в современной организации не функционирует без защиты от вредоносных программ. В дополнение к функциям классической антивирусной защиты (сигнатурный поиск в системе известного вредоносного ПО) антивирусные программные продукты применяют проактивные технологии (поиск уязвимостей и предотвращение заражения системы), а также средства защиты от сетевых атак. Лидерами антивирусной индустрии в нашей стране являются «Лаборатория Касперского» и компания «Доктор Веб».

Не помешают на компьютере специализированные программы для тонкой настройки параметров ОС – твикеры (tweaker), которые используют стандартные возможности системы (например, настройки реестра) или скрытые от обычного пользователя.

Динамические библиотеки – файлы, содержащие исполняемые коды и данные, которые могут использовать несколько программ одновременно.

В операционных системах семейства MS Windows большая часть функциональных возможностей операционной системы обеспечивается динамическими библиотеками (.DLL, .OCX., .CPL).

Когда несколько программ используют одну и ту же библиотеку функций (например, вывод на печать), размер программ на диске и в оперативной памяти может значительно уменьшиться, скорость загрузки, и, соответственно, производительность увеличится. Наиболее востребованные библиотеки могут раз-

мещаться в оперативной памяти сразу после запуска компьютера и выполняться в фоновом режиме (во время «простоя» центрального процессора).

Итак, системное программное обеспечение, в отличие от прикладного, не решает конкретные пользовательские задачи, а обеспечивает работу компьютера и других программ, управляет информационными ресурсами вычислительной системы, защищает информацию и т.д.

8.3. Файловые системы

В соответствии с принципами фон Неймана компьютеры могут выполнять только инструкции, находящиеся в оперативной памяти, – достаточно дорогостоящем устройстве относительно небольшой емкости. Вместе с тем основная часть программного обеспечения размещается на устройствах внешней памяти (магнитных или оптических дисках, флэш-памяти и т.д.) в файлах и папках (каталогах).

Файл – поименованная область на устройстве внешней памяти, организованная для долговременного хранения однородных данных.

Пользователь или программа с файлами могут выполнять **файловые операции**: создание, копирование, удаление, загрузка в память и выполнение, извлечение данных и т.д., в зависимости от типа файла.

Папка (каталог) – поименованная область на устройстве внешней памяти, организованная для группирования файлов по каким-либо признакам.

Каталоги могут образовывать иерархические структуры из нескольких отдельных деревьев (как в DOS/Windows) или же объединяться в одно дерево, общее для всех дисков (как в NIX-системах). Каталог верхнего уровня называется **корневой**.

Файловая система – это совокупность правил и способов организации данных на физическом пространстве носителя, а также их идентификационные признаки для обработки на компьютере.

Идентификатором файла являются его имя и адрес на внешнем носителе. Синтаксис идентификационной записи зависит от конкретной операционной системы.

Например, в MS Windows запись:

c:\win\system32\diskcopy.com

идентифицирует файл с именем «*diskcopy*», расширением «*com*» на диске «*c*» в папке «*system32*», вложенной в папку «*win*».

Символ «*:*» – указывает на имя тома, «**» – разделяет имена папок и файлов, «*.*» – отделяет имя файла от его расширения.

Том – область памяти на жестком диске. Том форматируется для определенной файловой системы, такой как FAT или NTFS, и обозначается буквой. Содержимое тома можно просмотреть, щелкнув его значок в проводнике Windows или в окне «Мой компьютер». Один жесткий диск может содержать несколько томов; тома также могут занимать несколько дисков.

Раздел диска – часть физического диска, которая ведет себя как отдельное устройство. Для хранения данных на созданном разделе необходимо сначала отформатировать его и назначить букву диска.

Разделы на базовых дисках называют базовыми томами; к ним относятся основные разделы и логические диски. Разделы на динамических дисках называют динамическими; к ним относятся простые, чередующиеся, составные, зеркальные тома и тома RAID-5.

Если файл размещен в компьютерной сети, то его адрес может выглядеть так:

http://www.opennet.ru/docs/RUS/tcp_conf/tcp01.html,

где «<http://>» – протокол обмена, а «www.opennet.ru» – доменное имя ресурса в сети Интернет.

В принципе, файловая система не должна зависеть от ОС, но любая ОС через драйвера и интерфейс программирования приложений (API) поддерживает определенные файловые системы.

Расширение имени файла указывает на его тип.

В некоторых операционных системах или файловых менеджерах можно устанавливать соответствия между расширениями файлов и приложениями. Когда пользователь открывает файл с зарегистрированным расширением, автоматически запускается соответствующая этому расширению программа. Некоторые расширения показывают, что файл сам является программой.

Иногда расширение указывает формат лишь в общем (например, расширение .doc использовалось для множества различных форматов текста – как простого, так и форматированного; а расширение «txt» не даёт никакой информации о том, в какой кодировке текст в файле), из-за чего необходимо использовать и другие способы определения типа.

Иногда расширение указывает только один из используемых в файле форматов (например, расширение «.ogg» первоначально использовалось для всех файлов в формате Ogg, независимо от кодеков, которыми закодированы содержащиеся в контейнере Ogg данные). Также расширение обычно не указывает версию формата (например, файлы в разных версиях XHTML могут использовать одни и те же расширения).

Кроме расширения в файловых системах предусмотрены дополнительные атрибуты файла.

Например, в MS Windows у каждого файла и папки имеется страница свойств, где показаны такие сведения, как размер, местоположение и дата создания файла или папки. При просмотре свойств файла или папки также можно получить следующие сведения:

- атрибуты файла или папки (только для чтения, скрытия, архивации, индексирования, сжатия и шифрования);
- тип файла;
- имя программы, назначенной для открытия данного файла;
- количество файлов или подпапок, содержащихся в данной папке;
- дату последнего изменения или обращения к файлу.

По типу носителя файловые системы можно классифицировать следующим образом:

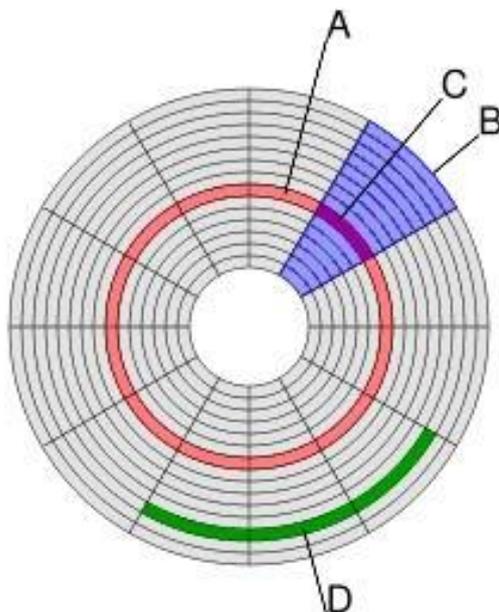
- с произвольным доступом (например, жёсткий диск): FAT32, NTFS, ext2 и др.;
- с последовательным доступом (например, магнитные ленты): QIC и др.;
- для оптических носителей: ISO9660, HFS, UDF и др.;
- для флэш-памяти: YAFFS, Extreme FFS, exFAT;
- сетевые файловые системы: NFS, CIFS, SSHFS, GmailFS и др.

Таким образом, основные **задачи файловой системы**:

- отображение логической модели файловой системы на физическую организацию хранилища данных;
- идентификация имен файлов, размещенных на внешних носителях;
- поддержка интерфейса файловых операций;
- обеспечение надежности хранения файлов, устойчивости к сбоям питания, ошибкам аппаратных и программных средств;
- обеспечение механизма разграничения доступа пользователей и программ при совместной работе с файлами.

Сравнительный обзор файловых систем FAT, FAT32 и NTFS

FAT является наиболее простой из поддерживаемых Windows файловых систем. Основой файловой системы FAT является таблица размещения файлов, которая помещена в самом начале тома. На случай повреждения на диске хранятся две копии этой таблицы. Кроме того, таблица размещения файлов и корневой каталог должны храниться в определенном месте на диске (для правильного определения места расположения файлов загрузки).



- (A) дорожка
- (B) геометрический сектор
- (C) сектор дорожки
- (D) кластер

Рис. 68. Разметка поверхности жесткого диска

Диск, отформатированный в файловой системе FAT, делится на кластеры, размер которых зависит от размера тома. Одновременно с созданием файла в каталоге создается запись и устанавливается номер первого кластера, содержащего данные. Такая запись в таблице размещения файлов сигнализирует о том, что это последний кластер файла, или указывает на следующий кластер.

Каталог FAT не имеет определенной структуры, и файлы записываются в первом обнаруженном свободном месте на диске. Кроме того, файловая система FAT поддерживает только четыре файловых атрибута: «Системный», «Скрытый», «Только чтение» и «Архивный».

В файловой системе FAT использован традиционный формат имен 8.3, имена файлов должны состоять из символов ASCII. Имя файла или каталога должно состоять не более чем из 8 символов, затем следует разделитель «.» (точка) и расширение длиной до 3 символов. Первым символом имени должна быть буква или цифра. При определении имени можно использовать все символы, за исключением перечисленных ниже:

«пробел», «.», «"», «/», «\», «[», «]», «:», «;», «|», «=», «,»

Использование этих символов может привести к получению неожиданных результатов. Имя не должно содержать пробелов.

Имена, зарезервированные для системы: CON, AUX, COM1, COM2, COM3, COM4, LPT1, LPT2, LPT3, PRN, NUL.

Разделы FAT имеют ограничение по размеру: 4 Гб под Windows и 2 Гб под MS-DOS. Максимальный размер файла равен 2 Гб.

FAT32 в операционных системах семейства Windows поддерживает тома объемом от 512 Мб до 2 Тб. Windows XP позволяет отформатировать в FAT32 тома объемом до 32 Гб. Максимальный размер файла равен 4 Гб.

В NTFS значительно увеличен допустимый размер файлов и томов – до 2^{64} байт (16 экзбайт или 18 446 744 073 709 551 616 байт). Не может использоваться для гибких дисков, так как объем служебной информации, необходимой для функционирования NTFS, не помещается на дискете. Рекомендуемый минимальный размер тома равен примерно 10 Мб.

Для задания имен файлов используется кодировка Юникод, и наряду с длинными именами обеспечена поддержка формата 8.3. Максимальный размер файла ограничен только размером тома.

NTFS в целях повышения надежности отслеживает все транзакции в отношении файловой системы и для восстановления целостности достаточно выполнить «откат» на предыдущее состояние.

Имена файлов в NTFS могут состоять не более чем из 255 символов, включая любое расширение. В отображении имен сохраняется регистр введенных символов, но сами имена не зависят от регистра. NTFS не различает имена в зависимости от регистра. В именах могут быть использованы любые символы, за исключением указанных ниже:

«?», «"», «/», «\», «<», «>», «*», «|», «:»

Для совместимости носителей под разными операционными системами, такими как Unix, Mac OS, Windows, Международная организация по стандартизации выпустила стандарт ISO 9660, описывающий файловую систему для оптических дисков.

Максимальный размер файла в ISO 9660 ограничен 2ГБ. Максимальная глубина вложенности папок – 8. В имени файла можно использовать только латинские буквы в верхнем регистре, цифры и символ подчеркивания. Имена файлов не должны включать пробелов. Имена файлов не должны начинаться или заканчиваться точкой. Имена файлов не должны иметь более одной точки. Имена каталогов не должны содержать точки.

На замену ISO 9660 разработан UDF (Universal Disk Format) – спецификация формата файловой системы, независимой от операционной системы для хранения файлов на оптических носителях.

UDF позволяет дозаписывать файлы на CD-R или CD-RW дисках, один файл одновременно, без существенных потерь дискового пространства, без использования метода пакетной записи, который изобиловал ошибками в различных прошивках различных приводов. Также UDF учитывает возможность выборочного стирания некоторых файлов на перезаписываемых носителях CD-RW, освобождая место на диске. UDF лучше подходит для DVD, так как имеет лучшую поддержку для дисков большого объёма – нет ограничения в 2 и 4 ГБ на размер файла. Допустимы фрагментированные файлы.

Несмотря на то, что UDF-формат изначально создавался для применения на оптических носителях, существует возможность создания разделов с файловой системой UDF на жестких дисках или флэш-накопителях в ОС GNU/Linux, Windows Vista, Windows 7, MacOS X. В Windows XP существует частичная поддержка UDF разделов, такие устройства будут доступны только для чтения.

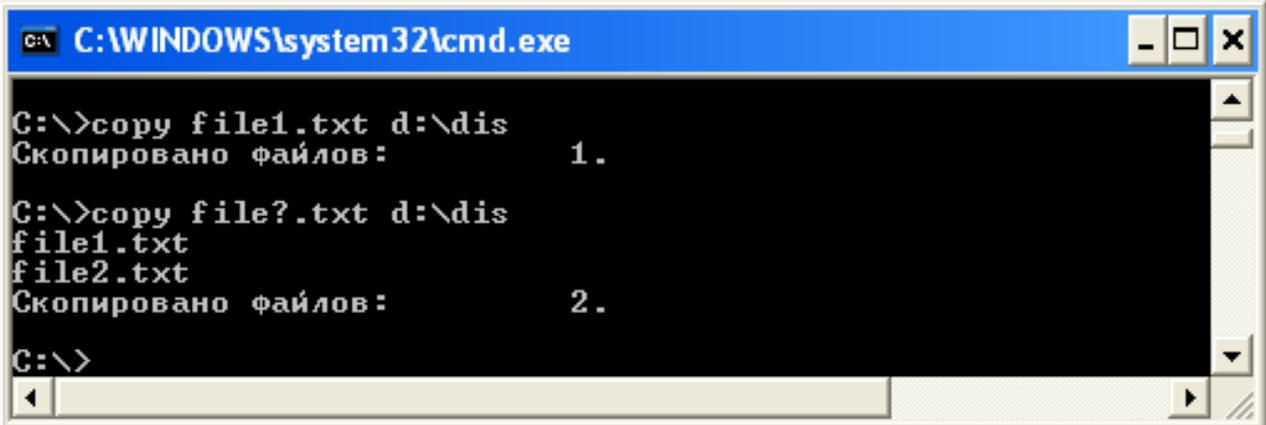
В функции современных операционных систем входит обеспечение выполнения файловых операций как для программ, так и в автоматизированном режиме под управлением пользователя.

Пользователь может манипулировать файлами с помощью командного процессора (интерпретатора командной строки) или графической оболочки (проводник Windows).

В случае если необходимо выполнить групповую операцию с несколькими файлами, их можно описать с помощью символов маски: «*» и «?».

Символ «*» маскирует в своей позиции любое число любых символов.

Символ «?» маскирует в своей позиции один символ.



```
C:\WINDOWS\system32\cmd.exe
C:\>copy file1.txt d:\dis
Скопировано файлов:      1.

C:\>copy file?.txt d:\dis
file1.txt
file2.txt
Скопировано файлов:      2.

C:\>
```

Рис. 69. Пример применения маскирующего символа «?»

Вопросы для самоконтроля:

1. Дайте определение понятию программного обеспечения.
2. Дайте определение понятию программной продукции.
3. Перечислите и охарактеризуйте программное обеспечение персональных компьютеров.
4. Дайте определение понятию прикладной программы.
5. Как можно классифицировать прикладные программы?
6. Дайте определение понятию системной программы.
7. Дайте определение понятию управляющей программы.
8. Дайте определение понятию операционной системы.
9. Что такое BIOS?
10. Перечислите основные функции современных операционных систем.
11. Перечислите основные компоненты современных операционных систем.
12. Дайте определение понятию файловой системы.
13. Поясните разницу между файловыми системами FAT, FAT32 и NTFS.

9. КОМПЬЮТЕРНЫЕ СЕТИ

9.1. Основные понятия компьютерных сетей

До появления компьютерных сетей многопользовательская или многозадачная работа стала осуществляться на больших ЭВМ (мэйнфреймах) в двух режимах:

- пакетный;
- разделения времени.

В первом варианте пакеты пользовательских программ на перфокартах или перфолентах ставились в очередь друг за другом на внешнем устройстве, последовательно загружались в оперативную память, обрабатывались центральным процессором, и результаты выводились на печать или экран. Каждая программа «захватывала» процессор на все время своей работы. До начала выполнения пользовательских программ в память загружалась операционная система и, при необходимости, транслятор.

В процессе повышения сложности решаемых задач все чаще возникала необходимость диалога с программой отображения промежуточных результатов и ввода дополнительных данных для продолжения работы. Пакетная обработка с монопольным доступом программы к процессору стала приводить к простоям ресурсов ЭВМ.

Операционные системы с режимом разделения процессорного времени между несколькими программами позволили организовать многопользовательский доступ к мэйнфрейму с нескольких параллельных терминалов (клавиатура + монитор). Пока одни задачи выполнялись в медленном диалоговом режиме (например, ввод или редактирование данных), другие занимали основную часть процессорного времени (вычисления).

В настоящее время достаточно широко используются терминалы – тонкие клиенты или специализированные рабочие места, подключаемые к суперкомпьютеру и в многопользовательском доступе использующие его информационно-вычислительные ресурсы. Как правило, тонкий клиент состоит из монитора, клавиатуры, мыши и бесшумного системного блока с пассивным охлаждением, USB-разъемами, без жесткого диска и подвижных деталей (рис. 70).

С расширением сферы использования компьютеров и увеличением их числа возникла необходимость обмена цифровой информацией между территориально распределёнными пользователями в минимально кратчайшие сроки.

Сначала компьютеры стали соединять друг с другом по схеме **точка-точка**. При таком соединении удаленный компьютер подключался по стандартным параллельным или последовательным интерфейсам и воспринимался операционной системой как одно из внешних устройств. Позднее компьютеры стали связывать между собой по телефонным коммутируемым и некоммутируемым каналам через модемы¹.

¹ Модулятор-демодулятор – устройство для обмена данными по телефонной линии.

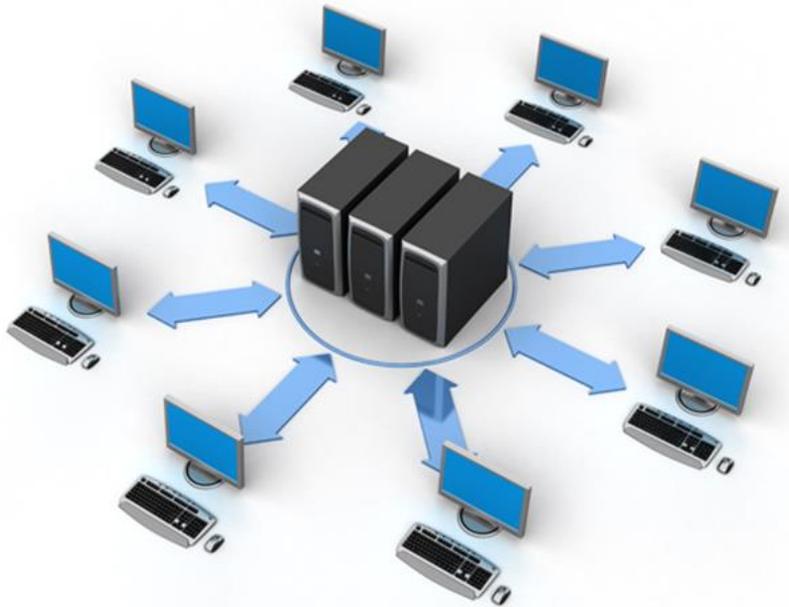


Рис. 70. Архитектура типа сервер – тонкий клиент

Телеобработка данных – совокупность методов, обеспечивающих пользователям дистанционный доступ к ресурсам систем обработки данных и ресурсам средств связи.

Вычислительная сеть – взаимосвязанная совокупность территориально рассредоточенных систем обработки данных, средств и (или) систем связи и передачи данных, обеспечивающая пользователям дистанционный доступ к ее ресурсам и коллективное использование этих ресурсов.

Архитектура вычислительной сети – совокупность принципов логической и физической организации технических и программных средств, протоколов и интерфейсов вычислительной сети.

Ресурсы вычислительной сети – программные, технические, информационные и организационные средства вычислительной сети, предназначенные для решения задач пользователей.

Современные компьютерные сети объединяют от нескольких единиц до сотен миллионов компьютеров и охватывают территории от отдельной комнаты до всей поверхности планеты Земля с околоземным пространством.

Принята следующая классификация компьютерных сетей **по масштабу**:

1. *Глобальная вычислительная сеть* (англ. Wide area network, WAN) – неоднородная вычислительная сеть из компьютеров и других сетей, практически не имеющая ограничений по количеству узлов и абонентов, а также длине каналов связи между ними.

2. *Локальная вычислительная сеть* (англ. Local area network, LAN) – вычислительная сеть определенной архитектуры, охватывающая небольшую территорию (здание, предприятие) и имеющая ограниченное адресное пространство для подключения компьютеров и активного сетевого оборудования.

3. *Домашняя сеть* (англ. Personal Area Network PAN) – локальная вычислительная сеть, развернутая в пределах жилого помещения для решения бытовых задач владельца и имеющая точки доступа к внешним телекоммуникационным сетям (телефонные каналы, Интернет).

Абонент (хост) – это активное сетевое устройство, подключенное к сети и реализующее сервисы типа «клиент-сервер». Чаще всего абонентом сети является компьютер, но может быть сетевой принтер, дисковый массив или другое периферийное устройство с непосредственным сетевым подключением.

Сетевой узел – устройство, соединяющее несколько звеньев данных вычислительной сети и осуществляющее коммутацию и (или) маршрутизацию данных по сети. Сетевые узлы могут быть конечные и промежуточные¹.

Классификация компьютерных сетей **по типу телекоммуникационной среды**:

1. Кабельные (телефонный провод, коаксиальный кабель, витая пара, волоконно-оптический кабель и пр.).

2. Беспроводные (3G, LTE, Wi-Fi, Bluetooth и пр.)².

Классификация компьютерных сетей **по типу функционального взаимодействия компьютеров**:

1. Одноранговая (децентрализованная, пиринговая³) – вычислительная сеть, управление ресурсами которой может взять на себя любой сетевой компьютер с соответствующей операционной системой.

2. С выделенным сервером (англ. server based) – вычислительная сеть, в которой управление ресурсами осуществляется компьютерами со специализированным программным обеспечением (файл-серверы, принт-серверы, серверы приложений, почтовые серверы и т.д.).

Сервер – это абонент сети, который обслуживает сетевые запросы других абонентов и предоставляет им свои ресурсы.

Выделенный (dedicated) сервер – это сервер, занимающийся только сетевыми задачами. Невыделенный сервер может помимо обслуживания сети выполнять и другие задачи.

Клиент – это абонент сети, который использует сетевые ресурсы.

Под сервером и клиентом следует понимать не сами компьютеры, а функции установленных на них программных приложений. В зависимости от программного обеспечения любой компьютер сети может быть как клиентом, так и сервером. Например, Интернет-браузеры являются клиентским программным обеспечением для просмотра страниц WEB-серверов.

Классификация компьютерных сетей **по пропускной способности**:

- низкоскоростные (до 10 Мбит/с),
- среднескоростные (до 100 Мбит/с),
- высокоскоростные (100 Мбит/с и выше).

Пропускная способность современных локальных вычислительных сетей составляет не менее 100 Мбит/с.

С помощью компьютерных сетей решаются следующие **задачи**:

- информационный обмен (on-line текстовые, аудио-, видеоконференции; электронная почта; социальные сети);

¹ ГОСТ 24402–88. Телеобработка данных и вычислительные сети. Термины и определения: сборник ГОСТов. - М.: Стандартинформ, 2005.

² англ. Wireless personal area network, WPAN.

³ англ. peer-to-peer, P2P – равный к равному.

- удаленный доступ к информационным ресурсам и ПО (автоматизированные информационные системы, базы данных, дистрибутивы ПО и обновления);
- совместное использование программного обеспечения (облачные технологии, игры);
- совместное использование дорогостоящих или уникальных периферийных устройств (принтеры, графопостроители);
- организация хранилищ архивных данных больших объемов с возможностью быстрого поиска в режиме удаленного доступа (NAS, ЦОД – центры обработки данных);
- объединение вычислительных мощностей сетевых компьютеров для совместного решения сложных задач (распределенные вычисления, DDoS-атаки);
- мониторинг и удаленное управление объектами в режиме реального времени (охранные системы, видеонаблюдение, управление городскими коммунальными объектами, «умный» дом) и др.

При очевидных достоинствах сетевые технологии обработки данных имеют и недостатки, которые необходимо учитывать при проектировании и эксплуатации компьютерных систем:

- сетевое оборудование и монтаж структурированных кабельных систем (СКС) требуют дополнительных затрат;
- обслуживание сети осуществляется выделенными специалистами – системными администраторами сети;
- проводная сеть «привязывает» компьютеры к сетевым розеткам;
- сетевая обработка данных повышает требования и, соответственно, затраты на обеспечение информационной безопасности.

Частично эти проблемы решает применение облачных технологий.

Облачная (рассеянная) обработка данных (англ. Cloud computing) – технология обработки данных, в которой информационные ресурсы предоставляются пользователю как удаленный Интернет-сервис. Пользователь, подключившись к облачному серверу, может просматривать и изменять свои данные с помощью облачных приложений, но его не волнуют проблемы приобретения, установки и настройки операционной системы и программного обеспечения на стороне этого сервера.

В качестве примера можно привести Microsoft Office 365. В зависимости от выбранного тарифного плана пользователь может приобрести лицензию на право использования Word, Excel, PowerPoint, OneNote, Outlook, Publisher, Access, InfoPath, Share Point Workspace и Lync последних версий. Дистрибутив для установки скачивается с сервисного портала portal.microsoftonline.com.

Доступ к portalу Office 365 осуществляется как с персональных компьютеров с ОС Windows, так и с любой модели смартфона с поддержкой протокола Exchange Active Sync. С Office 365 могут работать смартфоны с операционными системами Android и Symbian (с клиентом Nokia Mail for Exchange), а также семейства iPhone и Black Berry.

9.2. Топологии компьютерных сетей

По аналогии с любой сетевой структурой компьютерные сети также состоят из узлов и связей между ними, и, следовательно, рассматривать особенности их конструирования следует прежде всего с этой точки зрения.

Топология компьютерной сети – схема соединения узлов и абонентов сети.

В глобальных гетерогенных (неоднородных) сетях, где сеансы связи между двумя абонентами могут осуществляться по разным маршрутам, топология не имеет существенного значения и обычно скрыта от пользователей. Конкретные топологические схемы имеет смысл рассматривать при проектировании локальных вычислительных сетей.

Все многообразие сетевых топологий сводится к комбинациям трех базовых типов:

- шина;
- звезда;
- кольцо.

В топологии типа **шина** все компьютеры параллельно подключаются к одной линии связи. Информация от каждого компьютера одновременно передается всем остальным компьютерам.

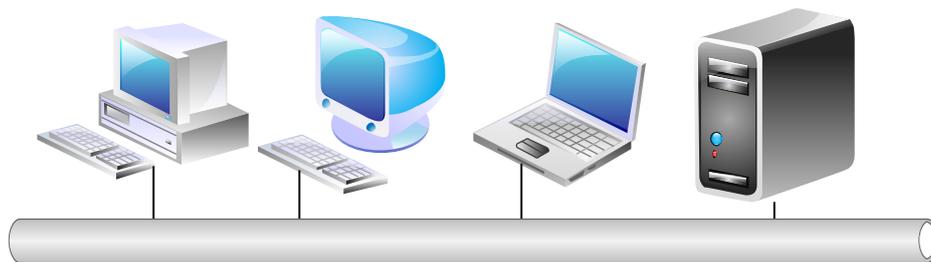


Рис. 71. Сетевая топология типа шина

Шина предполагает идентичность сетевого оборудования и равноправие всех абонентов в сети. Одновременно по шине могут обмениваться данными только два абонента в полудуплексном (half duplex) режиме (либо в прямом, либо в обратном направлении). При попытке нескольких устройств начать обмен разрешение возможных коллизий ложится на сетевое оборудование каждого отдельного абонента. Подключение новых абонентов к шине или отказ оборудования подключенных абонентов не приводит к нарушению работоспособности сети.

Шинная топология критична к обрывам сетевого кабеля. В связи с особенностями распространения высокочастотных электрических сигналов по длинным линиям связи на краях шины устанавливают пассивные согласующие устройства – терминаторы. Без терминаторов в сетевом кабеле возникают помехи, искажающие полезный сигнал и нарушающие работоспособность всей сети.

Для увеличения длины кабельной сети с топологией шина на краях вместо терминаторов подключают специальные усилители-повторители или репитеры, а к ним – дополнительные сегменты сети.

Короткое замыкание в любой точке сетевого кабеля или отказ конечных узлов сетевого оборудования любого абонента также выводят из строя всю сеть.

В топологии типа **звезда** все абоненты подключаются отдельными кабелями к центральному узлу на базе компьютера или специализированного сетевого устройства.

Звезда – это топология сети с выделенным центром, к надежности и производительности которого предъявляются повышенные требования. Выход из строя периферийного абонента или его сетевого оборудования никак не отражается на работоспособности оставшейся части сети. Обрыв кабеля или короткое замыкание в нем также отключают только одного абонента.



Рис. 72. Сетевая топология типа звезда

Для подключения абонента к центральному узлу обычно используются линии связи типа точка-точка, что упрощает требования к сетевому оборудованию. Но недостатком звезды является большой расход кабеля. Другим недостатком топологии звезда является ограничение количества подключаемых периферийных абонентов. Однако вместо периферийного абонента допустимо подключение еще одного центрального узла со своими абонентами, что порождает топологию **дерево**.

Кольцо – все абоненты последовательно подключаются по замкнутой цепочке. В классическом кольце каждый абонент передает информацию соседнему компьютеру только в одном направлении. Однако иногда в целях повышения скорости доставки данных и надежности сети предусматривают подключение двух (или более) параллельных линий связи, которые работают в противоположных направлениях.

На каждой линии связи, как и в случае звезды, работают только один передатчик и один приемник (точка-точка). Каждый абонент выступает в роли репитера-ретранслятора входного сигнала. На практике размеры кольцевых сетей достигают десятков километров. Центрального узла в кольцевой топологии нет, все абоненты равноправные. Право на захват сети переходит последовательно по кругу. Однако часто в кольце выделяется специальный абонент, ко-

торый управляет обменом. Максимальное количество абонентов в кольце может быть довольно велико (до тысячи и больше), но повреждение кабеля в любом месте кольца или отказ абонента приводит к нарушению работоспособности всей сети.

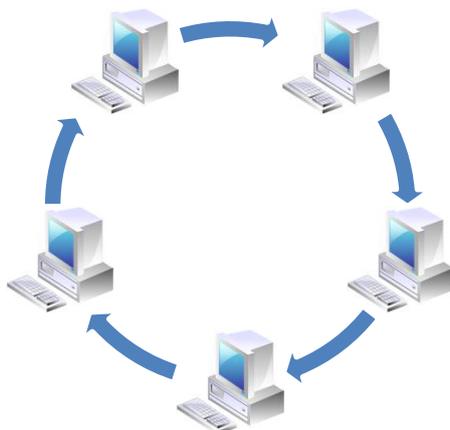


Рис. 73. Сетевая топология типа кольцо

Часто в локальных сетях применяются комбинированные топологии типа звезда-шина и звезда-кольцо.

В высоконадёжных специализированных сетях с высокой пропускной способностью используется топология сетевых ячеек типа **решетка (mesh)**, в которой один абонент непосредственно соединяется со многими.

При рассмотрении сетевых структур важно понимать, что топология сети может быть:

- физическая (сеть рассматривается как множество узлов и связей между ними);
- логическая (рассматриваются пути движения пакетов данных).

Например, в большинстве реализаций шинной топологии стандарта Ethernet абоненты подключаются к выделенным узлам (сетевые коммутаторы, концентраторы, точки доступа, маршрутизаторы). Т.е. физически – это пассивная звезда, а логическая топология Ethernet – это шина, в которой абоненты подключены параллельно.

С точки зрения среды передачи данных, между узлами сети каналы связи могут быть кабельными и беспроводными.

Кабельная система – это совокупность телекоммуникационных кабелей, шнуров и коммутационных устройств, предназначенных для подключения к информационно-вычислительной системе различных сетевых устройств.

Структурированная кабельная система – законченная совокупность кабелей связи и коммутационного оборудования, отвечающая требованиям соответствующих нормативных документов¹.

Сетевые кабели можно разделить на три группы:

- металлические коаксиальные кабели (англ. coaxial cable);

¹ ГОСТ Р 53245-2008. Информационные технологии. Системы кабельные структурированные. Монтаж основных узлов системы. Методы испытания: утв. Приказом Ростехрегулирования от 25.12.2008 № 786-ст. - М.: Стандартинформ, 2009.

- металлические кабели типа «витая пара», экранированные и неэкранированные;
- оптоволоконные кабели (англ. fiber optic).

Каждый тип кабеля имеет свои преимущества и недостатки, зависящие от характеристик:

- частотная полоса пропускания кабеля и затухание сигнала;
- скорость распространения сигнала по кабелю или задержка сигнала на метр длины кабеля;
- помехозащищенность кабеля и устойчивость к внешним воздействиям;
- волновое сопротивление.



Рис. 74. Оптоволоконный кабель, коаксиальный кабель и «витая пара»

В современных компьютерных сетях наиболее популярными являются кабели с витыми парами (двумя или четырьмя). Витая пара обычно используется для передачи информации на расстояние до 100 м на скоростях до 1000 Мбит/с.

Согласно стандарту EIA/TIA 568 существуют пять основных и две дополнительные категории кабелей на основе неэкранированной витой пары:

- категория 1 – обычный телефонный кабель (пары проводов не витые) с большим разбросом волнового сопротивления, полосы пропускания, перекрестных наводок, по которому можно передавать только речь;
- категория 2 – кабель из перекрученных пар для передачи данных в полосе частот до 1 МГц также с большим разбросом параметров;
- категория 3 – кабель из витых пар с девятью витками проводов на метр длины с волновым сопротивлением 100 Ом для передачи данных в полосе частот до 16 МГц;
- категория 4 – кабель, передающий данные в полосе частот до 20 МГц в остальном похож на кабель категории 3;
- категория 5 – наиболее популярный кабель в сетях типа Fast Ethernet, рассчитанный на передачу данных в полосе частот до 100 МГц. Состоит из витых пар, имеющих не менее 27 витков на метр длины;
- категории 6 и 7 – кабели для передачи данных в полосе частот до 600 МГц.

Недостатком неэкранированных витых пар является слабая защищенность от внешних электромагнитных помех и от съема информации. Перехват информационного сигнала возможен как контактным способом (воткнутые в

кабель иголки), так и с помощью бесконтактного «чтения» электромагнитного излучения. Для минимизации ущерба от этих недостатков применяется экранирование кабелей.

В кабеле STP каждая из витых пар помещается в металлическую оплетку-экран для уменьшения излучений кабеля, защиты от внешних электромагнитных помех и снижения взаимного влияния пар проводов друг на друга.

В настоящее время в локальных компьютерных сетях применяется пакетная технология передачи данных Ethernet 802.3 стандарта IEEE в различных модификациях:

- Ethernet (до 10 Мбит/с);
- быстрый Ethernet (Fast Ethernet, 100 Мбит/с);
- гигабитный Ethernet (Gigabit Ethernet, 1 Гбит/с).

В зависимости от пропускной способности спецификации Ethernet, использующие в качестве среды передачи данных витую пару, обозначаются:

- 10 BASE-T;
- 100 BASE-T;
- 1000 BASE-T.

Коаксиальный кабель состоит из центрального медного провода, заключенного в диэлектрический слой, и металлической оплетки (экрана) в общей изоляционной оболочке.

Экранированный коаксиальный кабель имеет высокую помехозащищенность, волновое сопротивление 50 Ом, пропускную способность 10 Мбит/с и позволяет передавать данные на расстояние до 500 м.

Спецификации Ethernet, использующие в качестве среды передачи коаксиальный кабель, обозначаются:

- 10 BASE5 («толстый» Ethernet диаметром 10 мм);
- 10 BASE2 («тонкий Ethernet» диаметром 5 мм).

Оптоволоконный (или волоконно-оптический кабель) содержит несколько стекловолоконных световодов, по которым информационные сигналы проходят расстояния до нескольких десятков километров со скоростью света. Кабель обладает высокой помехозащищенностью, обеспечивает идеальную гальваническую (электрическую) развязку компьютеров сети. Электромагнитные помехи не влияют на качество сигнала, а металлический экран обычно защищает кабель от механических повреждений.

Спецификации Ethernet, использующие в качестве среды передачи оптоволоконный кабель, обозначаются:

- 10 BASE-FX (10 Мбит/с до 2 километров);
- 100 BASE-SX, 100BASE-FX (100 Мбит/с до 10 километров);
- 1000 BASE-SX, 1000BASE-LX (1 Гбит/с до 50 километров).

Одновременно с кабельными каналами связи в компьютерных сетях широко используются беспроводные технологии:

- инфракрасные каналы;
- радиоканалы.

Пропускная способность инфракрасных каналов обычно не превышает $5 \div 10$ Мбит/с, но при использовании инфракрасных лазеров может достигать 100 Мбит/с.

Инфракрасные каналы используются для подключения периферийных устройств по схеме точка-точка (интерфейс IrDA) и делятся на две группы:

- каналы прямой видимости (до нескольких километров при отсутствии препятствий между абонентами);
- каналы на рассеянном излучении (в пределах одного помещения сигналы могут отражаться от стен, потолка, пола и других препятствий).

Радиоканал использует обмен данными по радиоволнам в радиусе действия передатчика. Беспроводные сетевые технологии обычно позволяют устанавливать связь на расстоянии до 100 м на частотах 2,4 ГГц и 5 ГГц. Скорость передачи – до 100 Гбит/с.

В компьютерных сетях для беспроводного подключения по радиоканалам используются стандарты Bluetooth, Wi-Fi¹, LTE и др., включая технологии передачи цифровых данных сотовой связи. В большинстве переносных компьютеров устройства подключения к сети по радиоканалу являются встроенными.

Основными недостатками радиоканалов, ограничивающими их применение, являются слабая помехоустойчивость и высокая вероятность перехвата информационного сигнала.

В зависимости от выбранного интерфейса существуют три типа связи между абонентами А и В:

- симплексная – однонаправленная ($A \Rightarrow B$, $A \Leftarrow B$);
- полудуплексная – двунаправленная поочередная ($A \Leftrightarrow B$), т.е. в каждый момент времени обмен осуществляется только в одну сторону;
- дуплексная – двунаправленная одновременная ($A \leftrightarrow B$).

9.3. Функционирование компьютерных сетей – сетевая архитектура и сетевые протоколы

Сетевая архитектура – это комбинация топологий, методов доступа к среде передачи данных и протоколов, необходимых для создания работоспособной сети.

Сетевой протокол – набор правил и действий (очередности действий), позволяющий осуществлять соединение и обмен данными между двумя и более включёнными в сеть устройствами.

Международная организация по стандартизации ИСО² в конце 70-х годов прошлого века предложила универсальную многоуровневую сетевую модель межкомпьютерного обмена данными для разработки сетевых протоколов – **OSI**³ (базовая эталонная модель взаимодействия открытых систем, ЭМВОС).

¹ англ. Wireless Fidelity.

² англ. International Organization for Standardization, ISO.

³ англ. Open systems interconnection basic reference model.

Эталонная модель OSI состоит из семи уровней¹, на каждом из которых описываются процессы сетевого взаимодействия абонентов и их аппаратно-программное обеспечение:

1. Физический уровень.
2. Уровень звена данных.
3. Сетевой уровень.
4. Транспортный уровень.
5. Сеансовый уровень.
6. Уровень представления данных.
7. Прикладной уровень.

Прикладной уровень обеспечивает взаимодействие пользовательских приложений с сетевыми службами и сервисами, контролирует целостность принятых данных.

Уровень представления обеспечивает предварительное преобразование данных из различных форматов приложений в универсальные сообщения (кодирование, сжатие) для последующего формирования пакетов транспортного уровня.

Сеансовый уровень обеспечивает организацию временных сеансов обмена данными между абонентами.

Транспортный уровень обеспечивает деление большого информационного сообщения на достаточно малые фрагменты (пакеты) и контролирует последовательность их передачи в сеть.

Сетевой уровень обеспечивает выбор маршрута для отправки пакетов с учетом различной адресации и загруженности сегментов сети.

Уровень звена данных обслуживает сетевой уровень и обеспечивает формирование помехозащищенных кадров данных непосредственно для передачи в сеть.

Физический уровень обеспечивает подключение абонента к физической среде передачи данных. На данном уровне описываются методы кодирования двоичных данных (0 и 1) электрическими, оптическими или радиосигналами, а также типы кабелей и соединительных разъемов.

Таким образом, на каждом уровне происходит преобразование исходного информационного сообщения в унифицированные форматы данных в соответствии с протоколами уровня. В реальных сетях все семь уровней явно не выделяются. Например, широко распространенная модель TCP/IP отличается от модели OSI и имеет 4 уровня со своими протоколами, называемые стек:

уровни OSI	уровни TCP/IP	протоколы
7	IV – прикладной (application layer)	HTTP, FTP, HTTPS, SMTP
6		
5	III – транспортный (transport layer)	TCP, UDP

¹ ГОСТ Р ИСО/МЭК 7498-1-99. Государственный стандарт Российской Федерации. Информационная технология. Взаимосвязь открытых систем. Базовая эталонная модель. Часть 1. Базовая модель: принят и введен в действие Постановлением Госстандарта России от 18.03.1999 № 78. - М.: ИПК Издательство стандартов, 1999.

4		
3	II – сетевой (internet layer)	IP, ICMP, IGMP
2	I – канальный (link layer)	Ethernet, SLIP, Token Ring
1		

В сетях TCP/IP любое сетевое сообщение содержит адреса передающего абонента и абонента приемника:

MAC-адрес¹ – уникальный 6-байтовый идентификатор сетевого устройства, устанавливаемый на заводе-изготовителе и не зависящий от конфигурации сети;

IP-адрес² – уникальный сетевой адрес абонента, зависящий от конфигурации сети.

Уникальность MAC-адресов достигается тем, что каждый производитель сетевого оборудования получает в координирующем комитете IEEE Registration Authority до шестнадцати миллионов адресов (2^{24}), которые распределяет между изготавливаемыми устройствами.

Для поиска адресата в каждой сети хранятся специальные таблицы соответствия IP и MAC адресов. Если сеть состоит из нескольких сегментов, то в соединительных узлах осуществляется переадресация пакета из одной логической сети (подсети) в другую, или **маршрутизация**.

В версии протокола IPv4 IP-адрес имеет размер 4 байта, представляемых обычно в десятичном формате и разделяемых точками, например: 194.1.133.20.

bin	1100 0010	0000 0001	1000 0101	0001 0100
dec	194	1	133	20

В пределах одного физического сегмента сети каждому абоненту (хосту) обязательно присваивается свой уникальный IP-адрес. В сетевых операционных системах IP-адрес настраивается как свойство сетевого адаптера.

Сетевые адаптеры (контроллеры, сетевые карты) – внешние устройства компьютера для организации обмена по сети в соответствии с принятой архитектурой и канальными протоколами модели OSI (физический уровень и уровень представления данных).

Информационное сообщение, преобразованное в совокупность пакетов данных на верхних уровнях OSI, пересылается в буферную память сетевого адаптера с командой начать передачу. Адаптер кодирует данные в соответствии протоколами своего уровня, анализирует текущее состояние сети и при первой же возможности направляет пакеты в виде битовых последовательностей в сеть.

В режиме приема адаптер «слушает» сеть, т.е. сравнивает MAC-адрес получателя во входящих пакетах со своим. Если адрес совпадает, то сетевой адаптер сохраняет пакет в свою буферную память, проверяет его на целостность и сообщает о приеме данных на верхние уровни. Многие адаптеры имеют функ-

¹ англ. Media Access Control.

² англ. Internet Protocol Address.

ции операционной системы и позволяют реализовать удаленную загрузку компьютера по сети.

Репитеры и трансиверы – сетевые устройства канального уровня, предназначенные для обеспечения заданных физических параметров информационных сигналов или их преобразования с учетом физической среды канала связи.

Концентраторы (hub) – сетевые устройства канального уровня, предназначенные для объединения в единую сеть нескольких физических сегментов.

Коммутаторы (switch) – сетевые устройства канального уровня, предназначенные для объединения в единую сеть нескольких физических сегментов с фильтрацией пакетов. Коммутаторы анализируют MAC-адреса пакетов и пропускают только пакеты «своей» сети.

Мосты (bridge), маршрутизаторы (router) и шлюзы (gateway) – устройства сетевого уровня, предназначенные для объединения разнородных сетей с разными протоколами обмена, форматами пакетов, методами кодирования, скоростью передачи и т.д. Фактически это полноценные компьютеры со специализированным программным обеспечением и возможностью удаленного доступа для управления по сети.

С развитием глобальных сетей все информационное пространство разбили на иерархические области – **домены** и абонентам одновременно с IP-адресами стали присваивать доменные имена.

Домен – это множество сетевых абонентов, объединенных общим адресным суффиксом.

DNS (доменная система имен)¹ – это распределенная в сети база данных с IP-адресами сетевых хостов и соответствующими им символьными именами. В каждом домене сети создается свой DNS-сервер с подобной базой данных в виде таблиц.

9.4. Интернет: история создания и особенности функционирования

Интернет – всемирная система добровольно объединённых компьютерных сетей, построенная на использовании протокола IP и маршрутизации пакетов данных. Интернет образует глобальное информационное пространство, служит физической основой для Всемирной паутины и множества систем (протоколов) передачи данных. Часто упоминается как Всемирная сеть и Глобальная сеть. Когда сейчас слово Интернет употребляется в обиходе, то чаще всего имеется в виду Всемирная паутина и доступная в ней информация, а не сама физическая сеть.

После запуска Советским Союзом искусственного спутника Земли в 1957 году Министерство обороны США посчитало, что на случай войны Америке нужна надёжная система передачи информации. Агентство передовых исследовательских проектов США (ARPA) предложило разработать для этого компьютерную сеть. Разработка такой сети была поручена Калифорнийскому универ-

¹ англ. Domain Name System.

ситету в Лос-Анджелесе, Стэнфордскому исследовательскому центру, Университету штата Юта и Университету штата Калифорния в Санта-Барбаре. Компьютерная сеть была названа ARPANET (англ. Advanced Research Projects Agency Network), и в 1969 году в рамках проекта сеть объединила четыре указанных научных учреждения, все работы финансировались за счёт Министерства обороны США. Затем сеть ARPANET начала активно расти и развиваться, её начали использовать учёные из разных областей науки.

Первый сервер ARPANET был установлен 1 сентября 1969 года в Калифорнийском университете в Лос-Анджелесе. Компьютер «Honeywell 516» имел 12 КБ оперативной памяти.

К 1971 году была разработана первая программа для отправки электронной почты по сети, программа сразу стала очень популярна.

В 1973 году к сети были подключены через трансатлантический телефонный кабель первые иностранные организации из Великобритании и Норвегии, сеть стала международной.

В 1970-х годах сеть в основном использовалась для пересылки электронной почты, тогда же появились первые списки почтовой рассылки, новостные группы и доски объявлений. Однако в то время сеть ещё не могла легко взаимодействовать с другими сетями, построенными на других технических стандартах. К концу 1970-х годов начали бурно развиваться протоколы передачи данных, которые были стандартизированы в 1982-83 годах. 1 января 1983 года сеть ARPANET перешла с протокола NCP на TCP/IP, который успешно применяется до сих пор для объединения (или, как ещё говорят, «наслоения») сетей. Именно в 1983 году термин «Интернет» закрепился за сетью ARPANET.

В 1984 году была разработана система доменных имён (англ. Domain Name System, DNS).

В 1984 году у сети ARPANET появился серьёзный соперник. Национальный научный фонд США (NSF) основал обширную межуниверситетскую сеть NSFNet (сокр. от англ. National Science Foundation Network). Она была составлена из более мелких сетей (включая известные тогда сети Usenet и Bitnet) и имела большую пропускную способность, чем ARPANET. К этой сети за год подключились около 10 тысяч компьютеров, звание «Интернет» начало плавно переходить к NSFNet.

В 1988 году был изобретён протокол Internet Relay Chat (IRC), благодаря чему в Интернете стало возможно общение в реальном времени (чат).

В 1989 году в Европе в стенах Европейского совета по ядерным исследованиям (CERN) родилась концепция Всемирной паутины. Её предложил британский учёный Тим Бернерс-Ли, он же в течение двух лет разработал протокол HTTP, язык HTML и идентификаторы URI.

В 1990 году сеть ARPANET прекратила своё существование, полностью проиграв конкуренцию NSFNet. В том же году было зафиксировано первое подключение к Интернету по телефонной линии (т.н. «дозвон», англ. Dialup access).

В 1991 году Всемирная паутина стала общедоступна в Интернете, а в 1993 году появился веб-браузер NCSA Mosaic. Всемирная паутина набирала популярность.

В 1995 году NSFNet вернулась к роли исследовательской сети, маршрутизацией всего трафика Интернета теперь занимались сетевые провайдеры, а не суперкомпьютеры Национального научного фонда.

В том же 1995 году Всемирная паутина стала основным поставщиком информации в Интернете, обогнав протокол пересылки файлов FTP, был образован Консорциум всемирной паутины (W3C). Можно сказать, что Всемирная паутина преобразила Интернет и создала его современный облик. С 1996 года Всемирная паутина почти полностью подменяет собой понятие «Интернет».

В 1990-е годы Интернет объединил в себе большинство существовавших тогда сетей (хотя некоторые, как Фидонет, остались обособленными). Объединение выглядело привлекательным благодаря отсутствию единого руководства и открытости технических стандартов Интернета, что делало сети независимыми от бизнеса и конкретных компаний. К 1997 году в Интернете насчитывалось уже около 10 млн компьютеров, было зарегистрировано более 1 млн доменных имён. Интернет стал очень популярным средством для обмена информацией.

В 1998 году папа римский Иоанн Павел II учредил всемирный День Интернета (30 сентября).

В настоящее время Интернет доступен не только через компьютерные сети, но и через спутники связи, радиосигнал, кабельное телевидение, телефон, сотовую связь, специальные опτικο-волоконные линии, электропровода и даже через трубы водопровода. Всемирная сеть стала неотъемлемой частью жизни в развитых и развивающихся странах.

Интернет состоит из многих тысяч корпоративных, научных, правительственных и домашних сетей. Объединение сетей разной архитектуры и топологии стало возможно благодаря протоколу IP (сокр. от англ. Internet Protocol) и принципу маршрутизации пакетов данных. Протокол IP был специально создан независимым в отношении физических каналов связи. То есть любая система (сеть) передачи цифровых данных, проводная или беспроводная, может передавать и трафик Интернета. На стыках сетей специальные маршрутизаторы (программные или аппаратные) занимаются сортировкой и перенаправлением пакетов данных, исходя из IP-адресов получателей этих пакетов. Протокол IP образует единое адресное пространство в масштабах всего мира, но в каждой отдельной сети может существовать и собственное адресное подпространство, которое выбирается исходя из класса сети. Такая организация IP-адресов позволяет маршрутизаторам однозначно определять дальнейшее направление для каждого мельчайшего пакета данных. В результате между отдельными сетями Интернета не возникает конфликтов, и данные беспрепятственно и точно передаются из сети в сеть по всей планете и ближнему космосу.

Сам протокол IP был рождён в дискуссиях внутри организации IETF (сокр. от англ. Internet Engineering Task Force, Task force – группа специалистов для решения конкретной задачи), чьё название можно вольно перевести как «Группа по решению задач проектирования Интернета». IETF и её рабочие группы по сей день занимаются развитием протоколов Всемирной сети. Комитеты организации публикуют т.н. документы RFC. В этих документах даются технические спецификации и точные объяснения по многим вопросам. Некоторые документы RFC возводятся организацией IAB (сокр. от англ. Internet Archi-

ecture Board – Совет по архитектуре Интернета) в статус Стандартов Интернет (англ. Internet Standard). С 1992 года IETF, IAB и ряд других Интернет-организаций входят в Общество Интернета (англ. Internet Society, ISOC). Общество Интернета предоставляет организационную основу для разных исследовательских и консультативных групп, занимающихся развитием Интернета.

Но если протокол IP отвечает за адресацию компьютеров в сети, то для обеспечения надежной передачи данных используется еще один протокол – TCP (Transmission Control Protocol). Он обеспечивает разбиение сообщения на небольшие части, называемые пакетами, и контролирует факт доставки каждого пакета получателю, повтор передачи пакета в случае ошибки и «сборку» пакетов на компьютере получателя в правильной последовательности. Так как эти протоколы в современных сетях всегда используются совместно, то говорят, что они образуют связку протоколов TCP/IP. Связка TCP/IP – основа Интернета.

Для того чтобы компьютерная вычислительная сеть могла нормально функционировать, необходимо отличать компьютеры друг от друга. Это достигается присвоением каждому компьютеру уникального адреса. Существует два основных способа адресации компьютеров в Интернете – цифровая (так называемая IP-адресация) и доменная. Цифровой и доменный адрес компьютера являются просто разными способами представления «имени» компьютера.

IP-адрес компьютера представляет собой группу чисел, разделенных между собой точками. Согласно распространенной сейчас версии протокола IP-адрес состоит из четырех чисел, каждое из которых должно находиться в диапазоне от 0 до 255 (4 числа по 8 бит каждое, всего 32 бита). Однако в связи с лавинообразным ростом количества компьютеров, подключенных к Интернету, таких адресов стало не хватать. Поэтому новая версия IP-протокола предусматривает уже 16 чисел (128 бит). Обычно первое и второе число определяют адрес сети, третье число – адрес подсети, а четвертое число является номером компьютера в этой подсети.

Адресация компьютеров с помощью чисел, несомненно, удобна, но только для самих компьютеров. Пользователю сети сложно запомнить большое количество цифровой информации. Гораздо легче люди запоминают так называемые «мнемонические», т.е. основанные на привычных словах имена компьютеров. Именно по такому принципу организована доменная адресация. Доменное имя компьютера – это тот же IP-адрес, но преобразованный в более удобную для человека форму. Доменный адрес компьютера представляет собой несколько слов, разделенных точками. Каждое слово – это имя компьютера или домена, в который он входит. В зависимости от уровня вложенности различают домены первого, второго, третьего и т.д. уровней.

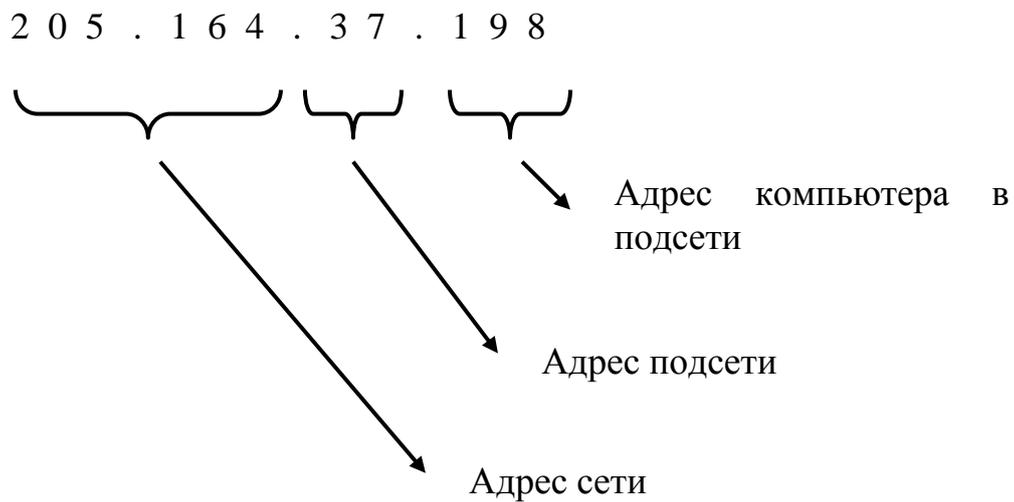


Рис. 75. Пример IP-адреса

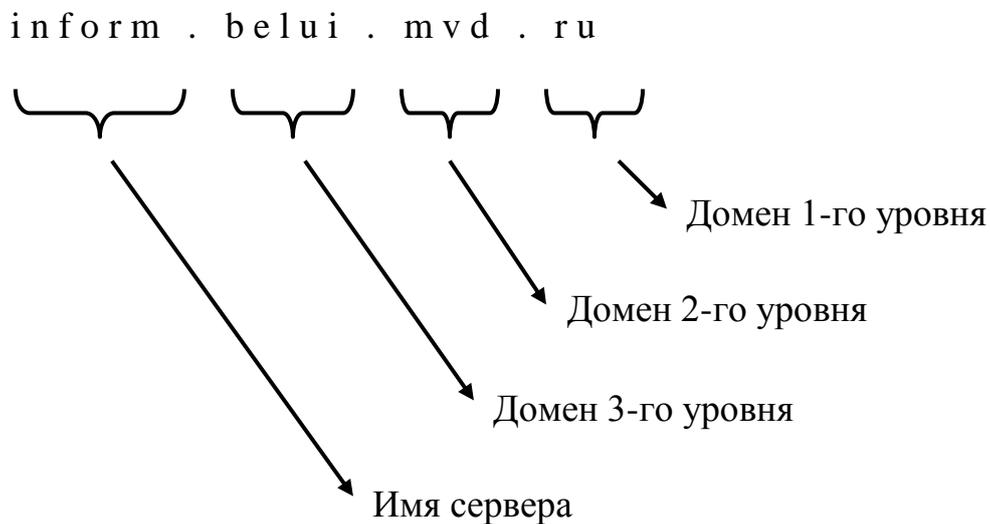


Рис. 76. Пример доменной адресации

Существуют зарезервированные имена доменов первого уровня. Так, домен .ru или .рф означает, что входящие в него домены (и компьютеры) расположены на территории России, .gov – правительственные домены, .com – домены коммерческих организаций и т.д.

Электронная почта – это система, позволяющая пользователям отправлять сообщения через модем или сетевое соединение с одного компьютера на другой. Если вы подключены к Интернету, можете посылать свои сообщения любому, у кого также есть доступ к Сети. Единственное, что вам в этом случае нужно знать, – адрес электронной почты получателя.

Адрес электронной почты похож на доменное имя. Он состоит из имени почтового ящика пользователя и имени почтового сервера, разделенных знаком «@». Официальное название этого символа – коммерческое, в настоящее время в России его чаще всего называют «собака».



Рис. 77. Пример адреса электронной почты

Сообщения, посылаемые через Интернет, имеют вид текстовых файлов. Если у вас есть соответствующая программа обработки электронной почты, то вы можете вложить в свои сообщения другие файлы. Вложенные файлы могут быть какими угодно – графическими изображениями, звуковыми файлами, другими документами и даже программами.

Основная особенность электронной почты заключается в том, что информация отправляется получателю не напрямую, а через промежуточное звено – электронный почтовый ящик, который представляет собой место на сервере, где сообщение хранится, пока его не запросит получатель. В большинстве случаев для доступа к почтовому ящику требуется наличие пароля. Доступ к почтовому серверу может предоставляться как через почтовые программы, так и через веб-интерфейс. Наиболее популярная программа для работы с электронной почтой – Microsoft Outlook.

Наиболее популярные сервисы Интернета - это:

- Всемирная паутина;
- социальные сети;
- веб-форумы;
- блоги;
- Интернет-магазины и Интернет-аукционы;
- группы (сайты) новостей;
- электронные платёжные системы;
- Интернет-радио, Интернет-телевидение;
- IP-телефония, Skype;
- мгновенный обмен сообщениями;
- IRC (реализовано так же, как веб-чаты);
- поисковые системы.

Поисковая система – это веб-сайт, предоставляющий возможность поиска информации в Интернете.

Комплекс программ, обеспечивающий функциональность поисковой системы, называют поисковым движком, или поисковой машиной. Основными критериями качества работы поисковой машины являются релевантность, полнота базы, учёт морфологии языка. Индексация информации осуществляется специальными поисковыми роботами.

Средства поиска и структурирования, иногда называемые поисковыми механизмами, используются для того, чтобы помочь людям найти информацию, в которой они нуждаются. Средства поиска типа агентов, пауков, кроуле-

ров и роботов используются для сбора информации о документах, находящихся в сети Интернет. Это специальные программы, которые занимаются поиском страниц, извлекают гипертекстовые ссылки на этих страницах и автоматически индексируют информацию, которую они находят для построения базы данных. Каждый поисковый механизм имеет собственный набор правил, определяющих, как собирать документы. Некоторые следуют за каждой ссылкой на каждой найденной странице и затем, в свою очередь, исследуют каждую ссылку на каждой из новых страниц и так далее. Некоторые игнорируют ссылки, которые ведут к графическим и звуковым файлам, файлам мультимедиа; другие игнорируют ссылки к ресурсам типа баз данных; другие проинструктированы, что нужно просматривать прежде всего наиболее популярные страницы.

Агенты – самые «интеллектуальные» из поисковых средств. Они могут искать сайты специфической тематики и возвращать списки сайтов, отсортированных по их посещаемости. Агенты могут обрабатывать содержание документов, находить и индексировать другие виды ресурсов, не только страницы. Они могут также быть запрограммированы для извлечения информации из уже существующих баз данных. Независимо от информации, которую агенты индексируют, они передают ее обратно базе данных поискового механизма.

Агенты извлекают и индексируют различные виды информации. Некоторые, например, индексируют каждое отдельное слово во встречающемся документе, в то время как другие индексируют только наиболее важные 100 слов в каждом, размер документа и число слов в нем, название, заголовки и подзаголовки и так далее. Вид построенного индекса определяет, какой поиск может быть сделан поисковым механизмом и как полученная информация будет интерпретирована.

Агенты могут также перемещаться по Интернету и находить информацию, после чего помещать ее в базу данных поискового механизма. Администраторы поисковых систем могут определить, какие сайты или типы сайтов агенты должны посетить и проиндексировать. Проиндексированная информация отсылается базе данных поискового механизма так же, как было описано выше.

Общий поиск информации в Сети осуществляют программы, известные как **пауки**. Пауки сообщают о содержании найденного документа, индексируют его и извлекают итоговую информацию. Также они просматривают заголовки, некоторые ссылки и посылают проиндексированную информацию базе данных поискового механизма.

Кроулеры просматривают заголовки и возвращают только первую ссылку.

Роботы могут быть запрограммированы так, чтобы переходить по различным ссылкам различной глубины вложенности, выполнять индексацию и даже проверять ссылки в документе. Из-за их природы они могут застревать в циклах, поэтому, проходя по ссылкам, они используют значительные ресурсы. Однако имеются методы, предназначенные для того, чтобы запретить роботам поиск по сайтам, владельцы которых не желают, чтобы они были проиндексированы.

Когда кто-либо хочет найти информацию, доступную в Интернете, он посещает страницу поисковой системы и заполняет форму, детализирующую информацию, которая ему необходима. Здесь могут использоваться ключевые слова, даты и другие критерии.

База данных отыскивает предмет запроса, основанный на информации, указанной в заполненной форме, и выводит соответствующие документы. Чтобы определить порядок, в котором список документов будет показан, база данных применяет алгоритм ранжирования. В идеальном случае документы, наиболее соответствующие пользовательскому запросу, будут помещены первыми в списке. Различные поисковые системы используют различные алгоритмы ранжирования, однако основные принципы определения соответствия (релевантности) следующие:

- количество слов запроса в текстовом содержимом документа;
- местоположение искомых слов в документе;
- удельный вес слов, относительно которых определяется релевантность, в общем количестве слов документа.

База данных выводит ранжированный подобным образом список документов и возвращает его человеку, сделавшему запрос. Различные поисковые механизмы также выбирают различные способы показа полученного списка – некоторые показывают только ссылки; другие выводят ссылки с первыми несколькими предложениями, содержащимися в документе или заголовок документа вместе со ссылкой.

При щелчке на ссылке к одному из найденных документов этот документ запрашивается у того сервера, на котором он находится.

Первой поисковой системой для Всемирной паутины был «Wandex», уже не существующий индекс, который создавал «World Wide Web Wanderer» – бот, разработанный Мэтью Грэйем (англ. Matthew Gray) из Массачусетского технологического института в 1993. Также в 1993 году появилась поисковая система «Aliweb», работающая до сих пор. Первой полнотекстовой (т.н. «crawler-based» – то есть индексирующей ресурсы при помощи робота) поисковой системой стала «WebCrawler», запущенная в 1994. В отличие от своих предшественников, она позволяла пользователям искать по любым ключевым словам на любой веб-странице, с тех пор это стало стандартом во всех основных поисковых системах. Кроме того, это был первый поисковик, о котором было известно в широких кругах. В 1994 был запущен «Lycos», разработанный в университете Карнеги Мелона.

Вскоре появилось множество других конкурирующих поисковых машин, таких как «Excite», «Info seek», «Inktomi», «Northern Light» и «AltaVista». В некотором смысле они конкурировали с популярными Интернет-каталогами, такими, как «Yahoo!». Позже каталоги соединились или добавили к себе поисковые машины, чтобы увеличить функциональность. В 1996 году русскоязычным пользователям интернета стало доступно морфологическое расширение к поисковой машине AltaVista и оригинальные российские поисковые машины Rambler и Aport. 23 сентября 1997 года была открыта поисковая машина Яндекс.

Популярными русскоязычными поисковыми системами являются www.yandex.ru и www.rambler.ru. Также у нас в стране активно используется поисковик Google.

Вопросы для самоконтроля:

1. Дайте определение понятию вычислительной сети.
2. Дайте определение понятию архитектуры вычислительной сети.
3. Что включают в себя ресурсы вычислительной сети?
4. Проведите классификацию компьютерных сетей по масштабу.
5. Проведите классификацию компьютерных сетей по типу телекоммуникационной среды.
6. Проведите классификацию компьютерных сетей по пропускной способности.
7. Дайте определение топологии компьютерной сети.
8. Перечислите и охарактеризуйте основные топологии компьютерной сети.
9. Дайте определение сетевой архитектуры.
10. Что такое сетевой протокол?
11. Перечислите иерархические уровни эталонной модели OSI.
12. Перечислите и охарактеризуйте коммутационное оборудование компьютерной сети.
13. Как возник Интернет?
14. На каких принципах построена сеть Интернет?
15. Как работает протокол TCP/IP?
16. Что такое доменное имя?
17. Перечислите сервисы Интернета.

СПИСОК ИСТОЧНИКОВ

1. Об информации, информационных технологиях и о защите информации: Федеральный закон от 27.07.2006 № 149-ФЗ // Собрание законодательства РФ. 2006. № 31 (ч. 1). Ст. 3448.
2. Об информации, информатизации и защите информации: Федеральный закон РФ от 20.02.1995 № 24-ФЗ (утратил силу) // Собрание законодательства РФ. 1995. № 8. Ст. 609.
3. ГОСТ Р 27.002-2009. Национальный стандарт Российской Федерации. Надежность в технике. Термины и определения: утв. Приказом Ростехрегулирования 09.12.2009 № 649-ст. – М.: Стандартинформ, 2010.
4. ГОСТ Р 53245-2008. Информационные технологии. Системы кабельные структурированные. Монтаж основных узлов системы. Методы испытания: утв. Приказом Ростехрегулирования от 25.12.2008 № 786-ст. – М.: Стандартинформ, 2009.
5. ГОСТ Р ИСО/МЭК 7498-1-99. Государственный стандарт Российской Федерации. Информационная технология. Взаимосвязь открытых систем. Базовая эталонная модель. Часть 1. Базовая модель: принят и введен в действие Постановлением Госстандарта России от 18.03.1999 № 78. – М.: ИПК Издательство стандартов, 1999.
6. ГОСТ 15971-90. Системы обработки информации. Термины и определения: утв. Постановлением Госстандарта СССР от 26.10.1990 № 2698 // Режим доступа: СПС «КонсультантПлюс».
7. ГОСТ 19781-90. Государственный стандарт Союза ССР. Обеспечение систем обработки информации программное. Термины и определения: утв. и введен в действие Постановлением Госстандарта СССР от 27.08.1990 № 2467. – М.: ИПК Издательство стандартов, 1998.
8. ГОСТ 24402–88. Телеобработка данных и вычислительные сети. Термины и определения: сборник ГОСТов. – М.: Стандартинформ, 2005.
9. Бройдо В.Л., Ильина О.П. Вычислительные системы, сети и телекоммуникации: учебник для вузов. – СПб.: Питер, 2011. - 688 с.
10. Грошев А.С. Информатика: учебник для вузов. – Архангельск: Арханг. гос. техн. ун-т, 2010. - 468 с.
11. Грошев А.С., Закляков П.В. Информатика. – М.: ДНК-Пресс, 2014. - 592 с.
12. Винер Н. Кибернетика, или управление и связь в животном и машине. – 2-е изд. – М.: Наука; Главная редакция изданий для зарубежных стран, 1983. - 344 с.
13. Информатика и математика: учебное пособие / А.А. Дрога, П.Н. Жукова, Д.Н. Копонев, Д.Б. Лукьянов, А.Н. Прокопенко. – Белгород: БелЮИ МВД России, 2008. - 288 с.

14. Лапин В.В. Информатика и информационные технологии в профессиональной деятельности: примерная программа для образовательных учреждений высшего профессионального образования МВД России. – М.: ДГСМК МВД России, 2011.
15. Макарова Н.В., Волков Б.В. Информатика. – СПб.: Питер, 2011. - 576 с.
16. Могилев А.В. Информатика: учебник / А.В. Могилев, Н.И. Пак, Е.К. Хённер. – М., 2001.
17. Ожегов С.И., Шведова Н.Ю. Толковый словарь русского языка. – М.: Азъ, 1993.
18. Правовая защита информации (Информационное право): учебное пособие / Прокопенко А.Н., Александров А.Н., Дрога А.А. – Белгород: Бел ЮИ МВД России, 2012. - 227 с.
19. Сибирский В.К. Правовая информатика. – М.: МИЭМП, 2010. - 124 с.
20. Симонович С.В. Информатика. Базовый курс. – СПб.: Питер, 2011. - 540 с.
21. Степанов А.Н. Информатика. Базовый курс для студентов гуманитарных специальностей высших учебных заведений. – СПб.: Питер, 2011. - 719 с.
22. Смелянский Р.Л. Компьютерные сети: в 2 т. Т. 1. Системы передачи данных: учебник. – М.: Академия, 2011. - 304 с.
23. Смелянский Р.Л. Компьютерные сети: в 2 т. Т. 2. Сети ЭВМ: учебник. – М.: Академия, 2011. - 240 с.
24. Таненбаум Э.С. Современные операционные системы: учебное пособие. – СПб.: Питер, 2010. - 1120 с.
25. Хлебников А.А. Информационные технологии: учебник. – М.: КНОРУС, 2014. - 472 с.
26. Большая юридическая библиотека. [Электронный ресурс]. Режим доступа: <http://www.ur-library.info>.
27. Виртуальный компьютерный музей. [Электронный ресурс]. Режим доступа: <http://www.compute-museum.ru>.
28. Газета «Информатика» издательского дома «Первое сентября». [Электронный ресурс]. Режим доступа: <http://inf.1september.ru>.
29. Единая коллекция цифровых образовательных ресурсов. [Электронный ресурс]. Режим доступа: www.school-collection.ru.
30. Интернет-экзамен в сфере профессионального образования (ФЭПО). [Электронный ресурс]. Режим доступа: <http://www.fepo.ru/index.php>.
31. Информатика и информационные технологии: сайт лаборатории информатики МИОО. [Электронный ресурс]. Режим доступа: <http://ii.metodist.ru>.

32. Информационная система «Единое окно доступа к образовательным ресурсам». [Электронный ресурс]. Режим доступа: <http://window.edu.ru/window>.
33. Клякс@.net: Информатика в школе. Компьютер на уроках. [Электронный ресурс]. Режим доступа: <http://www.klyaksa.net>.
34. Методические материалы и программное обеспечение для школьников и учителей: сайт К.Ю. Полякова. [Электронный ресурс]. Режим доступа: <http://kpolyakov.newmail.ru>.
35. Олимпиадная информатика. [Электронный ресурс]. Режим доступа: <http://www.olympiads.ru>.
36. Российский общеобразовательный портал. [Электронный ресурс]. Режим доступа: <http://www.school.edu.ru>.
37. Сетевой класс Белогорья. [Электронный ресурс]. Режим доступа: <http://www.belclass.net>.
38. Система электронного обучения «Пегас» БелГУ. [Электронный ресурс]. Режим доступа: <http://pegas.bsu.edu.ru>.
39. Федеральный портал «Российское образование». [Электронный ресурс]. Режим доступа: <http://www.edu.ru>.
40. Федеральный центр информационных образовательных ресурсов. [Электронный ресурс]. Режим доступа: <http://fcior.edu.ru>
41. IT-образование в России. [Электронный ресурс]. Режим доступа: http://www.edu-it.ru/inf_vuz.

УЧЕБНОЕ ИЗДАНИЕ

Прокопенко Алексей Николаевич,
кандидат технических наук, доцент;
Жукова Полина Николаевна,
доктор физико-математических наук, доцент;
Дрога Андрей Анатольевич;
Страхов Андрей Александрович

Основы информатики и информационных технологий в ОВД

Учебное пособие

Редактор
Комп. верстка

Ю.Н. Удалова
И.Ю. Чернышева

Подписано в печать 2016 г., 9 усл.-печ. л., бумага офсетная, печать трафаретная.
Тираж 50 экз. Заказ № 30

Отпечатано в отделении полиграфической и оперативной печати
Белгородского юридического института МВД России имени И.Д. Путилина
г. Белгород, ул. Горького, 71

ISBN 978-5-91776-130-5

