

Глава 5. Устройство компьютера

Компьютер – это программируемый автомат для обработки данных.

Из этого определения можно сделать вывод, что компьютер состоит из двух важнейших составляющих: *аппаратной части* и *программного обеспечения* (ПО). В технической литературе их часто называют английскими терминами *hardware* и *software*¹.

Поскольку одно и то же оборудование может быть перенастроено на выполнение новых задач простой заменой ПО, такие универсальные компьютеры можно выпускать большими партиями, и это делает их производство проще и дешевле. За счет этого во многих областях они заменили специализированные устройства.

Исторически существовало два принципиально разных типа вычислительных машин – *аналоговые* и *цифровые*. Они различались по способу представления обрабатываемых данных: в аналоговой или цифровой форме. Цифровая техника быстро доказала свои преимущества:

- высокую точность вычислений;
- универсальность и быстроту перехода от одной задачи к другой;
- способность хранить большие объемы данных.

В результате почти все современные компьютеры работают только с дискретной (цифровой) информацией. Поэтому в этой главе рассматривается только цифровая вычислительная техника.

5.1. История развития вычислительной техники

История появления и развития вычислительной техники – это обширная тема, которой посвящено множество книг. С точки зрения курса информатики исторические сведения интересны, прежде всего, тем, что позволяют отследить основные направления развития компьютерной техники и попытаться предвидеть ее ближайшее будущее.

В отечественной технической литературе приблизительно до 80-х годов прошлого века везде использовался термин *электронно-вычислительная машина* (ЭВМ). Позднее это словосочетание стало постепенно вытесняться новым более коротким названием *компьютер*. Все разновидности современной вычислительной техники сейчас называются только компьютерами, но, тем не менее, старые модели по традиции именуются ЭВМ.

5.1.1. Вехи истории



Блез Паскаль (1623-1662)
(www.thocp.net)

Первая механическая машина, с помощью которой можно было производить вычисления, была изготовлена известным французским ученым *Блезом Паскалем* (1623-1662) в 1645 году. Чтобы отдать дань уважения этому изобретению, один из языков программирования впоследствии был назван именем *Паскаль*.

Идея о реализации вычислений в автоматическом (без участия человека) режиме впервые была предложена и детально развита английским ученым *Чарльзом Бэббиджем* (1791-1871). Он спроектировал и описал аналитическую машину, состав и принципы действия которой фактически повторились в будущих ЭВМ.

¹ Дословный перевод этих терминов затруднителен: *ware* – это изделия, *hardware* изначально – металлические изделия, скобяные товары, а применительно к компьютеру – его детали (платы, монитор и прочее «железо»); термин *software* исключительно компьютерный, возник как противопоставление слов *soft* (мягкий, гибкий, податливый) и *hard* (твердый, жесткий, негнущийся), т.е. *software* гибко «подстраивает» *hardware* к решению разнообразных задач

Бэббидж посвятил всю свою жизнь работе над машиной, но построить ее из механических деталей не удалось: уровень техники XIX века не позволял изготовить столь сложный и точный механизм.



Чарльз Бэббидж (1791-1871)
(ru.wikipedia.org)



Ада Лавлейс (1815-1852)
(www.science20.com)

Программированием для аналитической машины Бэббиджа занималась *Ада Лавлейс* (дочь английского поэта Д.Г. Байрона, 1815-1852). Ее идеи оказали большое влияние на развитие программирования. Например, ей принадлежат термины «цикл» и «рабочая ячейка». В честь первого в мире программиста один из языков программирования получил имя *Ада*.

Первой ЭВМ², продемонстрировавшей на практике возможность автоматических расчетов по программе, считается ЭНИАК (сокращение от английского словосочетания *Electronic Numeric Integrator and Computer*). Он был построен в 1944 году в США под руководством *Джона Моучли*; главным инженером проекта был *Преспер Эккерт*. ЭНИАК содержал 18000 электронных ламп и, занимая зал 9×15 м², потреблял около 150 кВт электроэнергии; он выполнял более 350 умножений и 5000 сложений за секунду. Данные вводились в машину с помощью перфокарт, а программа обработки набиралась с помощью штекеров на специальных панелях.



Компьютер «ЭНИАК»
(www.fi.edu)



Джон фон Нейман (1903-1957)
(startupgallery.org)

Опыт построения первой ЭВМ был проанализирован *А. Берксом*, *Г. Голдстейном* и *Д. фон Нейманом*. В 1946 году они опубликовали работу «Предварительное рассмотрение логической конструкции электронного вычислительного устройства», ставшую классической. Сформулированные в ней принципы построения вычислительных машин используются и сейчас, несмотря на то, что со времени публикации прошло более полувека. В компьютерной литературе эти принципы часто называют *фон-неймановской архитектурой* ЭВМ, хотя Джон фон Нейман не был ее единоличным автором.

Дальнейший прогресс вычислительной техники во многом определялся развитием ее элементной базы. Важной вехой на этом пути стало создание в 1947 году *транзистора* (*У. Шокли*, *Д. Бардин* и *У. Браттейн*). Транзистор – это полупроводниковый прибор для управления электрическими сигналами. На основе транзисторов могут собираться цифровые электронные схемы, такие как логические элементы и триггеры.

В 1958 году *Дж. Килби* разработал первую *интегральную*



Транзисторы
(ru.wikipedia.org)

² Вопросы приоритета в конструировании самой первой вычислительной машины до сих пор являются предметом непрекращающихся споров, в том числе и судебных.

микросхему – кристалл, в котором размещается не один транзистор, а целая схема на нескольких транзисторах – например, один или несколько триггеров. Все ее вспомогательные радиодетали (резисторы, конденсаторы и другие) также изготавливаются с помощью полупроводниковых технологий.

Первый *микропроцессор* Intel 4004 был разработан под руководством инженера М. Хоффа и выпущен в 1971 году. Он был четырехбитным, поскольку предназначался для микрокалькуляторов и должен был производить вычисления над десятичными числами



С. Джобс и С. Возняк
с компьютером Apple-I
(cedmagic.com)

(см. разд. 2.10 о двоично-десятичном кодировании). Интересно, что цель его создания была чисто технологическая: заменить 12 специализированных микросхем одной универсальной, но программируемой. Достоинства микропроцессора как главного узла для компьютеров были оценены позднее.

В ходе совершенствования элементной базы вычислительные машины становились все более мощными и компактными. В результате появились персональные компьютеры (ПК), которыми мы сейчас пользуемся. В 1976 году два молодых приятеля С. Джобс и С. Возняк в гараже родителей Джобса собрали ПК Apple, положивший начало известному ныне семейству компьютеров. А в 1981 году был продемонстрирован первый компьютер другого семейства – IBM PC (*IBM Personal Computer*), потомки которого в нашей стране особенно широко распространены.

5.1.2. Поколения ЭВМ (совершенствование элементной базы)

Неудачная попытка Ч. Бэббиджа построить механическую аналитическую машину показала, насколько важную роль играет элементная база. Именно поэтому дальнейшую историю вычислительной техники принято делить на периоды в соответствии с теми элементами, из которых изготавливались машины.

Первое поколение ЭВМ относят к периоду примерно 1945-1955 годов. Эти машины были построены на базе электронных ламп³. Открыл его уже описанный ранее ЭНИАК. В нашей стране машинами первого поколения были МЭСМ (малая электронная счетная машина, 1951 год), БЭСМ (большая электронная счетная машина, 1952 год), Стрела (1953 год), Урал (1954 год), М-20 (1959 год). Все эти машины были огромными, неудобными и дорогими.



ЭВМ первого поколения МЭСМ
(фото из Единой коллекции цифровых ресурсов)

Второе поколение (примерно 1955-1965 годы) появилось, когда на смену лампам в электронных схемах пришли транзисторы. Первый экспериментальный компьютер на транзисторах ТХ-0 был создан в 1955 году в Массачусетском технологическом институте

³ В середине XX века было разработано несколько счетных машин на электромагнитных реле, но их из-за малого количества не принято включать в классификацию поколений.

(США). ЭВМ на транзисторах были значительно меньше и имели существенно более высокое быстродействие; они потребляли гораздо меньше энергии, были надежнее и не требовали таких громоздких систем отвода тепла, как ламповые машины. Многие машины второго поколения уже помещались в обычной комнате среднего размера, например, ЭВМ серии *Наири* (1964 год) или *МИР* (машина инженерных расчетов, 1965 год). Наиболее производительными ЭВМ этого поколения стали *Стретч* (США, 1960 год), *Атлас* (Великобритания, 1961 год), *CDC 6600* (США, 1964 год) и *БЭСМ-6* (СССР, 1967 год).



ЭВМ второго поколения БЭСМ-6
(фото из Единой коллекции цифровых ресурсов)

Третье поколение ЭВМ (примерно 1965-1975 годы) связано с появлением интегральных микросхем. Казалось бы, размеры этих ЭВМ снова должны существенно уменьшиться, но этого не произошло. Дело в том, ЭВМ третьего поколения были предназначены для коллективной (многопользовательской) работы. Это было время крупных вычислительных центров, предоставлявших услуги огромному числу пользователей из многих организаций. Поэтому главное внимание уделялось не уменьшению размеров и стоимости машин, а повышению их вычислительной мощности и эффективности обработки больших объемов данных.

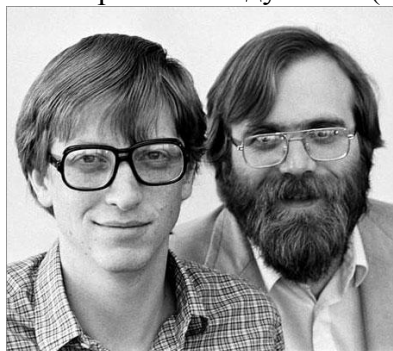
Отличительная черта третьего поколения – выпуск *семейств* вычислительных машин, которые совместимы между собой как аппаратно (все устройства сконструированы по одинаковым стандартам), так и программно (имеют одинаковую систему команд). Впервые идею общей *архитектуры*, обеспечивающей выполнение написанных ранее программ на любой новой модели, предложила фирма IBM, которая разработала семейства больших ЭВМ *IBM/360* и *IBM/370*. В этот период в СССР было принято решение перейти к копированию зарубежной техники ради обеспечения совместимости. В результате в странах Восточной Европы были выпущены «аналоги» упомянутых выше семейств под общим названием *ЕС ЭВМ* (Единая система ЭВМ). Одновременно появились мини-ЭВМ семейства *СМ ЭВМ* (Семейство малых ЭВМ), аналогичное известному зарубежному семейству *PDP* фирмы *DEC*.

Четвертое поколение берет свое начало примерно с 1975 года. Прогресс в электронике дал возможность существенно увеличить плотности «упаковки» элементов на кристалле, и в одной микросхеме теперь удавалось собрать целый узел, например, *микропроцессор*. Микросхемы такого уровня стали называть БИС (большие интегральные схемы, от 1000 до 10000 элементов на кристалле), а позднее – СБИС (сверхбольшие ИС, более 10000 элементов). Именно они стали основой четвертого поколения ЭВМ, которое существует вплоть до настоящего времени.



Уменьшение размеров ячеек ЭВМ первого – третьего поколений (в каждой из них – по два триггера)

Увеличение плотности схемы позволило, в первую очередь, повысить быстродействие компьютеров⁴. Кроме того, возросла и надежность, поскольку значительная часть электрических соединений выполнена внутри кристалла. Однако при высокой плотности монтажа ухудшается теплоотдача от миниатюрных деталей, поэтому требуются специальные меры по отводу тепла (например, установка вентиляторов охлаждения).



Б. Гейтс и П. Аллен (chernykh.net)

Первый восьмиразрядный процессор *Intel 8080*, предназначенный специально для компьютеров, был выпущен в 1974 году. На его базе был разработан микрокомпьютер *Альтаир*, имевший большой коммерческий успех. Он вошел в историю еще и потому, что в 1975 году молодой студент Билл Гейтс со своим другом Полом Алленом реализовали на нем язык программирования BASIC. Чуть позднее они создали известную компанию *Microsoft*.

Кроме персональных компьютеров, к четвертому поколению относятся *серверы* – мощные вычислительные машины, которые используются для управления компьютерными сетями. Они предоставляют свои ресурсы (например, принтеры, файлы или программы) в коллективное пользование. Серверы могут эффективно обслуживать большое количество пользователей одновременно. Например, два сервера *Hewlett-Packard T600* (по 12 процессоров в каждом), установленные в системе резервирования билетов *Amadeus*, способны практически без задержек обслуживать примерно 60 миллионов запросов в сутки (система имеет около 180 тысяч терминалов в более чем ста странах мира)⁵.

Важное направление в компьютерах четвертого поколения – *параллельная* (одновременная) обработка данных. Если решаемую задачу удастся разбить на независимые друг от друга действия, то их не обязательно делать друг за другом, а можно для экономии времени выполнять одновременно. Правда, для этого требуется несколько процессоров, но современный уровень техники это позволяет. Более того, в последнее время были сконструированы многоядерные процессоры, т.е. фактически несколько процессоров в одном кристалле.

Мощные многопроцессорные компьютеры, в которых выполняется параллельная обработка данных, называют *суперкомпьютерами*. Это уникальные устройства, поэтому они изготавливаются штучно.

В литературе часто упоминаются суперкомпьютеры серии *CRAY*, разработанные под руководством *Сеймура Крэя*. Первая модель этой серии, *CRAY-1*, была построена в США в 1976 году и имела огромный коммерческий успех.

Все развитые страны ведут жесткую конкуренцию в области суперкомпьютеров, поскольку обладание такой техникой позволяет решать стратегически важные вычислительные задачи:

- исследование геофизики Земли, прогнозирование изменений климата на планете;
- создание математических моделей молекул (полимеров, кристаллов и т.п.), синтез новых материалов и лекарств;
- расчет процессов горения и взрыва, а также моделирование других физических задач (обтекание летательных аппаратов, прочность кузовов автомобилей);
- расчеты процессов нефте- и газодобычи, а также сейсморазведки недр;
- проектирование новых электронных устройств.

Приведем несколько примеров применения суперкомпьютеров. Исследователи фирмы *IBM* на протяжении десятилетий изучают деятельность мозга и пытаются моделиро-

⁴ При быстродействии 10^9 элементарных операций в секунду (типичное по порядку величины значение для современного компьютера) за время каждой из них электрический сигнал со скоростью $3 \cdot 10^8$ м/с успевает пройти путь всего 30 см.

⁵ <http://parallel.ru/vvv/lec1.html>

вать ее. В 2009 году появилось сообщение⁶ о том, что полученная в рамках проекта модель мозга по своим возможностям превзошла уровень кошки: моделируется 1 миллиард нейронов и 10 триллионов связей между ними! Модель работает на базе суперкомпьютера *Blue Gene/P*, имеющего 147 456 процессоров и 144 Тбайт оперативной памяти.

По данным компании *Ford Motor*, благодаря детальному моделированию на суперкомпьютере, количество разрушаемых в крэш-тестах⁷ автомобилей удастся сократить на треть.⁸

Применение суперкомпьютеров для расчета состава лекарств позволяет уменьшить срок их разработки с нескольких лет до полугода.

Мощные компьютеры используются при создании компьютерных спецэффектов в кино. Например, в фильме «Властелин колец» фирме *WETA Digital* потребовалось столько суперкомпьютеров, что Новая Зеландия вышла на первое место в мире по их количеству на душу населения.⁹

В 2009 году в МГУ введен в строй самый мощный российский суперкомпьютер Ломоносов производительностью около 400 Тфлопс¹⁰. В его состав входят 8892 многоядерных процессора (общее число ядер – 35776). На момент запуска, Ломоносов занимал в мировом рейтинге суперкомпьютеров Top500 двенадцатое место.



Суперкомпьютер Ломоносов

Много шума наделал японский проект *пятого поколения* (1982-1992 годы). Было заявлено, что в основе компьютеров пятого поколения будут уже не вычисления, а логические заключения, т.е. произойдет переход от обработки данных к обработке знаний. Машину обещали научить воспринимать речевые команды человека, читать рукописный текст, анализировать графические изображения и делать многие другие нетривиальные для компьютера вещи. Планы проекта были грандиозны. Но, несмотря на щедрое финансирование и передовые позиции японских технологий, успехи оказались весьма скромными. На основе программного моделирования на компьютерах четвертого поколения удалось реализовать лишь отдельные детали проекта, причем реальная машина, работающая на базе логических выводов, так и не вышла за стены лабораторий.

Таким образом, создание принципиально новых компьютеров пятого поколения закончилось неудачей. Все компьютеры, используемые в настоящее время, по-прежнему построены на базе идей четвертого поколения. Классификация поколений «замерла» в ожидании новых революционных идей. Такие идеи особенно необходимы еще и потому, что электронная техника уже подошла к пределу быстродействия, который определяется законами физики: для увеличения скорости передачи данных требуется уменьшать размеры электронных деталей, но плотность упаковки транзисторов в полупроводниковом кристалле и так уже практически достигла максимально возможной. Поэтому идет поиск неэлектронных средств хранения и обработки данных.

В первую очередь, ученые попытались использовать в качестве носителя информации свет – так появились оптические процессоры¹¹. В них можно применять *параллельную обработку данных*, например, одновременно выполнять какую-то операцию со всеми пик-

⁶ Computerworld, 2009, N 39

⁷ Крэш-тесты – это тесты, исследующие поведение машин при сильном ударе о бетонное препятствие.

⁸ <http://parallel.ru/vvv/lec1.html>

⁹ <http://www.edu.ru/grants/?newsid=5248>

¹⁰ Флопс (FLOPS – floating point operations per second) – единица измерения количества операций с вещественными числами за 1 секунду; приставка «тера» добавляется по тем же правилам, что и при измерении информации; очевидно, что операции над вещественными числами гораздо сложнее и выполняются гораздо дольше, чем над целыми.

¹¹ <http://ysa.ifmo.ru/data/publications/BOOK008/paper1-001.doc>

селями изображения. В 2003 году был выпущен оптический процессор *Enlight256*¹², у которого оптическое ядро, а входные и выходные данные представлены в электронном виде. Быстродействие этого процессора – 8 триллионов операций в секунду. Он состоит из 256 лазеров, набора линз и фотоприемников. Оптические процессоры используются в военной технике и при обработке видеоданных в реальном времени.

Большие надежды связаны с разработкой квантовых компьютеров, в которых применяются идеи квантовой физики, описывающей законы микромира и поведение отдельных элементарных частиц. Данные для обработки в квантовом компьютере записываются в систему *кубитов* – квантовых битов. Затем с помощью специальных операций состояние этой системы изменяют по определенному алгоритму. Конечное состояние системы кубитов – это и есть ответ в задаче. Особые свойства кубитов¹³ позволяют организовать параллельную обработку данных, так же, как и в многопроцессорных системах. Поэтому многие задачи, для решения которых сейчас не хватает вычислительных ресурсов (например, раскрытие шифров), будут достаточно быстро решены, как только квантовый компьютер будет построен.

В некоторых лабораториях ведется разработка биологических компьютеров (*биокомпьютеров*), которые работают как живой организм. Ячейки памяти биокомпьютеров – это молекулы сложных органических соединений, например, молекулы ДНК, в которых хранится наследственная информация. Сам процесс вычислений – это химическая реакция, результат – состав и строение получившей молекулы.

Проводятся также исследования и в области нанотехнологий, с помощью которых можно будет построить транзистор размером с молекулу.

5.1.3. Развитие возможностей от поколения к поколению

С каждым поколением вычислительных машин развиваются их *аппаратные возможности*. ЭВМ становятся более мощными и универсальными. Расширяется количество обрабатываемых **типов данных**:

- 1 поколение – только числовые данные;
- 2 поколение – добавляется простейшая обработка символов;
- 3 поколение – числа, текстовые и графические данные;
- 4 поколение – добавляются аудио- и видеоданные.

Появившись как устройство для облегчения вычислений, компьютер сейчас все более активно обрабатывает разнообразную нечисловую информацию. Чтобы подчеркнуть широкие возможности современных компьютеров, введен специальный термин – *мультимедиа*.

Мультимедиа (от латинских слов *multum* – много и *medium* – средства) – одновременное использование различных форм представления информации (графика, текст, видео, фотографии, анимация, звук и т.д.) и их объединение в одном объекте¹⁴.

Как правило, при использовании технологий мультимедиа человек может влиять на показ материалов: перейти вперед или вернуться назад, изменить настройки, выбрать один из предложенных вариантов и т.п. Подобное взаимодействия человека и компьютера называют *интерактивностью* (взаимной активностью). Пример мультимедиа-объекта – компьютерная презентация.

Другое направление в развитии аппаратной части – это увеличение разнообразия и одновременно рост сложности **внешних устройств**, присоединяемых к ЭВМ:

- 1 поколение – штекеры и переключатели, индикаторные лампочки, устройства ввода с перфокарт;

¹² <http://dkws.narod.ru/linux/etc/optical/cpu.html>

¹³ В отличие от привычного нам бита, кубит устроен так, что способен вместить в себя гораздо больше информации.

¹⁴ Термин «мультимедиа» имеет также несколько связанных с приведенным определением значений: мультимедиа-компьютер, мультимедиа-носитель, программные и аппаратные средства мультимедиа и др.

2 поколение – перфоленты, магнитные ленты и барабаны, печатающие устройства;

3 поколение – магнитные диски, текстовые и графические мониторы, графопостроители;

4 поколение – огромное разнообразие внешних устройств, в том числе

- накопители на лазерных дисках;
- мышь, джойстик, шлемы виртуальной реальности и др.;
- возможность подключения бытовой электроники (фотоаппаратов, музыкальных плееров, сотовых телефонов и др.) с помощью кабелей и беспроводных соединений.

Каждое новое поколение компьютеров расширяет возможности **программного обеспечения**. Использовать современный компьютер без ПО практически невозможно. Его стоимость нередко намного превышает стоимость аппаратной части (в первых поколениях было наоборот!).

1 поколение. Программы разрабатывали хорошо подготовленные специалисты на машинном языке, сама программа представляла собой последовательность чисел (машинных кодов). Стандартного программного обеспечения практически не было.

2 поколение. Появились первые языки программирования. Некоторые из них были разработаны для конкретных машин, но значительно удобнее оказались машинно-независимые языки, такие как *Фортран* (1957) и *Алгол* (1960). Написать программу на таком языке было значительно проще: с этим уже вполне мог справиться рядовой научный работник, причем не обязательно с математическим образованием. В конце второго поколения появились *мониторы* – специальные программы, управляющие последовательным прохождением заданий через ЭВМ в автоматическом режиме. Их дальнейшее развитие в следующем поколении привело к появлению операционных систем.

3 поколение. Созданы первые операционные системы (ОС), которые обеспечивали работу компьютеров в многопользовательском режиме и управляли большим количеством сложных внешних устройств (в первую очередь, магнитными дисками). Для «общения» с ОС разработаны специальные языки управления заданиями. Широкое распространение получили созданные ранее языки программирования, например, *Фортран* для математических вычислений и *Кобол* для экономических расчетов. Начали появляться пакеты прикладных программ для решения задач в конкретных областях.

4 поколение. Для управления компьютером пользователь теперь использует не язык программирования, а различные меню и кнопки. Например, необходимую команду можно выбрать из меню (перечня доступных в данной ситуации возможностей) на естественном языке. Стало реально освоить компьютер после очень короткой подготовки. Программное обеспечение для ПК становится необычайно разнообразным – написано столько программ, что их трудно даже просто систематизировать. Казалось бы, такое разнообразие ПО должно сделать программирование ненужным, но нередко проще написать собственную небольшую программу решения конкретной задачи, чем тратить время на поиск и освоение возможностей готовых универсальных пакетов.

Таким образом, при переходе от поколения к поколению возрастает вычислительная мощность компьютеров. Значительная часть новых возможностей направляется на повышение удобства работы пользователя. В человеко-машинном общении отчетливо прослеживается движение от машинного языка к языкам, естественным для человека. В результате расширяется область применения и круг пользователей компьютерной техники.



Контрольные вопросы

1. Что такое компьютер?
2. Охарактеризуйте программную и аппаратную части компьютера.
3. Почему универсальный компьютер с изменяемой программой удобнее, чем специализированная техника? Ответ обоснуйте.
4. Что такое цифровая и аналоговая техника?
5. Почему цифровая техника вытеснила аналоговую?
6. Перечислите основные вехи в истории развития вычислительной техники.

7. Какова заслуга Чарльза Бэббиджа?
8. В честь кого названы языки программирования Ада и Паскаль? Какое отношение они имеют к вычислительной технике?
9. *Подберите дополнительный материал по вопросу о приоритете создания первой ЭВМ.
10. Что такое транзистор и микросхема? Из чего они изготавливаются?
11. С какой целью разрабатывались первые микропроцессоры?
12. *Почему микропроцессор *Intel 4004* был специально спроектирован для работы только с четырехбитными данными? *Указание:* вспомните, как можно хранить отдельные десятичные цифры числа. (Ответ: для вычислений в калькуляторах удобен 4-разрядный двоично-десятичный код).
13. По какому принципу ЭВМ делятся на поколения?
14. Почему время существования того или иного поколения всегда указывается приблизительно?
15. Перечислите все поколения ЭВМ и назовите элементную базу каждого из них.
16. Что дает уменьшение базовых элементов вычислительной техники?
17. *Почему электронные схемы требуют охлаждения? Все ли элементы нуждаются в дополнительном охлаждении?
18. Какие поколения вычислительных машин построены на базе полупроводниковых технологий? Чем отличается друг от друга их элементная база?
19. Объясните, почему большинство ЭВМ третьего поколения имели крупные габариты, несмотря на очередное уменьшение размеров элементной базы.
20. Когда появились первые семейства ЭВМ? Какая фирма предложила идею? В чем преимущества выпуска совместимых моделей?
21. Компьютеры какого поколения сейчас стоят на полках магазинов?
22. Какие разновидности компьютеров входят в четвертое поколение?
23. Как вы понимаете термин «персональный компьютер»?
24. Какие семейства персональных компьютеров вы знаете? Какое из них появилось раньше?
25. Перечислите бытовые приборы, в которых применяются микропроцессоры.
26. Что такое суперкомпьютеры? Зачем они используются?
27. Найдите в Интернете рейтинг суперкомпьютеров *Top500*. Какие страны занимают в нем лидирующее положение? Есть ли там российские компьютеры?
28. *Зачем в суперкомпьютерах так много процессоров? Подумайте, любая ли задача может быть решена быстрее, если ее считать параллельно на множестве процессоров? (В качестве помощи можно воспользоваться аналогией с распределением частей одного большого задания между учениками класса.)
29. Назовите примеры вычислительных машин каждого из четырех поколений. Найдите дополнительный материал об этих машинах.
30. Что вы можете сказать о судьбе пятого поколения?
31. *Почему, по-вашему, уже довольно давно не происходило смены поколений?
32. Найдите дополнительный материал о разрабатываемых в лабораториях принципиально новых компьютерах.
33. Какие типы данных обрабатывались на ЭВМ каждого из поколений?
34. Как изменялся набор внешних устройств при переходе от одного поколения к другому?
35. Опишите, как происходило развитие программного обеспечения.
36. Что вы можете сказать по поводу роли программного обеспечения: уменьшается она или увеличивается по сравнению с предыдущими поколениями?
37. Предположим, что появился процессор с каким-то принципиально новым свойством. Как быстро этим свойством смогут воспользоваться потребители? Какова роль программного обеспечения в этом?

38. Быстродействие вычислительной техники постоянно растет. Как же тогда объяснить, что пользователи жалуются на «медлительные» компьютеры и все время стараются купить новые, еще более производительные?
39. *Влияет ли развитие программных средств на развитие аппаратной части? (Ответ: ПО все усложняется, что требует все более мощную аппаратуру. Кроме того, во многих случаях функции оборудования перекладываются на программы, что позволяет упростить аппаратную часть компьютера.)
40. Что представляли собой программы для первых машин? Почему для их записи было удобно использовать не двоичную систему счисления, а восьмеричную или шестнадцатеричную?
41. Зачем были созданы языки программирования? Когда они появились?
42. Когда появились операционные системы и с чем это связано?
43. Попробуйте назвать положительные и отрицательные последствия огромного разнообразия существующих программ.
44. Почему развитие ПО расширяет количество пользователей компьютера?
45. *Насколько сейчас, по-вашему, актуально умение программировать? Попробуйте найти аргументы «за» и «против» (учитывайте разные цели работы на компьютере у людей).

5.2. Фундаментальные принципы устройства компьютеров

В предыдущем разделе вы увидели, что вычислительная техника в своем развитии прошла целый ряд характерных этапов. Несмотря на это, некоторые фундаментальные (базовые, основные) принципы устройства ЭВМ почти не изменились. Поэтому логично начать знакомство с устройством компьютера именно с них.

Классические принципы построения ЭВМ были предложены в работе *А. Беркса, Г. Голдстейна и Д. фон Неймана «Предварительное рассмотрение логической конструкции электронного вычислительного устройства»*. Обычно выделяют¹⁵ следующие наиболее важные идеи этой работы:

- состав основных компонентов вычислительной машины;
- принцип двоичного кодирования;
- принцип адресности памяти;
- принцип иерархической организации памяти;
- принцип хранимой программы;
- принцип программного управления.

Рассмотрим их подробнее.

5.2.1. Общие принципы

Основные компоненты машины. В самом первом разделе с таким названием фон Нейман с соавторами определили и обосновали состав ЭВМ:

«Так как законченное устройство будет универсальной вычислительной машиной, оно должно содержать несколько основных органов, таких как орган арифметики, памяти, управления и связи с оператором. Мы хотим, чтобы машина была полностью автоматической, т.е. после начала вычислений работа машины не зависела от оператора».

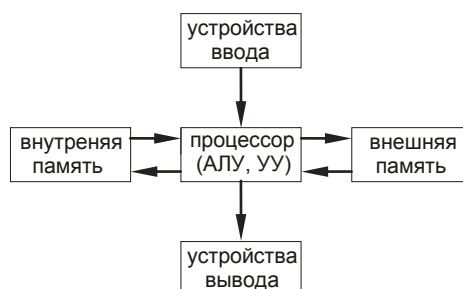
Таким образом, ЭВМ должна состоять из нескольких блоков, каждый из которых выполняет вполне определённую функцию. Эти блоки есть и в современных компьютерах:

- **арифметико-логическое устройство (АЛУ)**, в котором выполняется обработка данных;

¹⁵ Эта техническая статья не содержит отдельного пронумерованного перечня принципов, поэтому в учебной литературе встречаются не принципиальные отличия в их формулировке и описании.

- *устройство управления (УУ)*, обеспечивающее выполнение программы и организующее согласованное взаимодействие всех узлов машины; сейчас АЛУ и УУ изготавливают в виде единой интегральной схемы – *микروпроцессора*;
- *память* – устройство для хранения программ и данных; память обычно делится на *внутреннюю* (для временного хранения данных во время обработки) и *внешнюю* (для длительного хранения между сеансами обработки);
- *устройства ввода*, преобразующие входные данные в форму, доступную компьютеру;
- *устройства вывода*, преобразующие результаты работы ЭВМ в форму, удобную для восприятия человеком.

В классическом варианте все эти устройства взаимодействовали через процессор:



Принцип двоичного кодирования. Устройства для хранения двоичной информации и методы ее обработки наиболее просты и дешевы. Поскольку в ЭВМ используется двоичная система, необходимо переводить данные из десятичной формы в двоичную (при вводе) и наоборот (при выводе результатов). Однако такой перевод легко автоматизируется, и многие пользователи даже не знают об этих внутренних преобразованиях.

В первых машинах использовались только числовые данные. В дальнейшем ЭВМ стали обрабатывать и другие виды информации (текст, графика, звук, видео), но это не привело к отмене принципа двоичного кодирования. Даже цифровые сигнальные процессоры¹⁶, предназначенные для обработки цифровых сигналов в реальном времени, используют двоичное представление данных.

В истории известен пример успешной реализации троичной ЭВМ «Сетунь» (1959 год, руководитель проекта *Н. П. Брусенцов*), но он так и остался оригинальным эпизодом и не оказал влияния на эволюцию вычислительной техники. В первую очередь, это связано с серьёзными проблемами, которые возникают при изготовлении элементов троичного компьютера (триггеров – ячеек с тремя устойчивыми состояниями, сумматоров и т.д.) на основе полупроводниковых технологий. Эти проблемы так и не были решены, тогда как наладить массовое производство аналогичных устройств для двоичных компьютеров оказалось значительно проще.

5.2.2. Принципы организации памяти

Принцип адресности памяти. Оперативная память машины состоит из отдельных битов. Для записи или считывания группы соседних битов объединяется в *ячейки памяти*, каждая из которых имеет свой *адрес* (номер). Нумерацию ячеек принято начинать с нуля.

Адрес ячейки памяти – это её номер.

Ячейка – это минимально возможный считываемый из памяти объем данных: невозможно прочитать меньшее количество бит, а тем более отдельный бит.

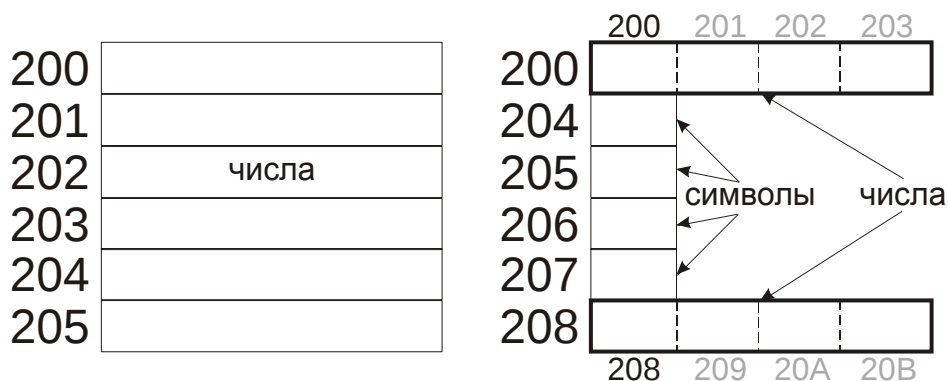
Использование чисел для нахождения в памяти требуемых ячеек выглядит абсолютно естественно: в компьютерах любая информация кодируется числами, так что адреса ячеек не исключение из этого фундаментального правила. Если номера соседних ячеек отличаются на единицу, удобно организовывать их последовательную обработку.

¹⁶ В англоязычной литературе их называют *DSP = Digital Signal Processor*.

Разрядность ячеек памяти (количество бит в ячейке) в разных поколениях была различной. Первоначально ЭВМ были построены исключительно для математических расчетов. Числа желательно было представлять как можно точнее, поэтому ячейки ОЗУ в первых машинах были длинными. Кроме чисел, машина должна была хранить в памяти еще и команды программы; как правило, в то время размер числовой ячейки совпадал с размером команды, что существенно упрощало устройство памяти.

Примерно на стыке второго и третьего поколений ЭВМ стали использовать для обработки символьной информации, что привело к серьезному неудобству: в существующую числовую ячейку памяти помещалось 4-5 символов. Инженеры выбрали наиболее простое решение проблемы – уменьшить размер ячейки так, чтобы можно было обращаться к каждому символу отдельно. *Байтовая* память, основой которой стала восьмибитная ячейка, прекрасно зарекомендовала себя и используется в компьютерной технике до настоящего времени.

В результате перехода к «коротким» ячейкам памяти числа стали занимать несколько ячеек (байт), каждая из которых имеет собственный адрес. На рисунке слева показана организация ячеек памяти первых ЭВМ, а на рисунке справа – современная (байтовая) структура памяти.



На левом рисунке числа занимают по одной ячейке, причем номера этих ячеек отличаются на единицу. Справа показано два 32-битных числа, которые хранятся в байтах 200-203 и 208-20B (адреса указаны в шестнадцатеричной системе). По принятому правилу за адрес числа принимается наименьший из адресов, так что в данном случае адреса чисел – 200 и 208. Кроме того, на правом рисунке между числами (в байтах с 204 по 207) размещены четыре символа. Заметим, что современные компьютеры могут извлекать из памяти до восьми соседних байтовых ячеек за одно обращение к памяти.

Очень важно, что информация может считываться из ячеек и записываться в них в произвольном порядке, поэтому организованную таким образом память принято называть *памятью с произвольным доступом* (англ. *RAM = random access memory*). Чтобы лучше понять смысл этого термина, сравните такую память с магнитной лентой, данные с которой можно получить только путем последовательного чтения.

Часто термин RAM отождествляют с русским термином ОЗУ – *оперативное запоминающее устройство*. Это не совсем точно. Дело в том, что кроме ОЗУ существует еще одна разновидность памяти с произвольным доступом – *постоянное запоминающее устройство* (ПЗУ, англ. *ROM = Read Only Memory* – память только для чтения). Главное отличие ПЗУ от ОЗУ заключается в том, что при решении задач пользователя содержимое ПЗУ не может быть изменено. ПЗУ гораздо меньше ОЗУ по объему, но это очень важная часть компьютера, поскольку в нем хранится доступное в любой момент программное обеспечение. Благодаря этому ПО компьютер сохраняет работоспособность даже тогда, когда в ОЗУ нет никакой программы.

Таким образом, ОЗУ и ПЗУ – это два вида памяти с произвольным доступом, обращение к данным в которых построено на основе принципа адресности.

Принцип иерархической организации памяти. К памяти компьютера предъявляются два противоречивых требования: ее объем должен быть как можно больше, а ско-

рость работы – как можно выше. Ни одно реальное устройство не может удовлетворить им одновременно. Любое существенное увеличение объема памяти неизбежно приводит к уменьшению скорости ее работы. Действительно, если память большая, то обязательно усложняется механизм нахождения в ней требуемых данных¹⁷, а это сразу замедляет чтение из памяти. Кроме того, чем быстрее работает память, тем она дороже, и, следовательно, меньше памяти можно установить за приемлемую для потребителей стоимость.

Чтобы преодолеть противоречие между объемом памяти и ее быстродействием, используют несколько различных видов памяти, связанных друг с другом. Когда в 1946 году впервые формулировался этот принцип, в состав ЭВМ предполагалось включить всего два вида памяти: оперативную память и память на магнитной проволоке (предшественник устройств хранения данных на магнитной ленте). Дальнейшее развитие вычислительной техники подтвердило необходимость построения иерархической памяти: в современном компьютере уровней иерархии гораздо больше.

5.2.3. Выполнение программы



Фрагмент коммутационной панели устройства IBM-557; требуемая операция получается соединением (коммутацией) соответствующих отверстий

Принцип хранимой программы.

Первые ЭВМ программировались путем установки перемычек на специальных панелях, так что процесс подготовки к решению задачи мог растянуться на несколько дней. Такое положение дел никого не устраивало, и в фон-неймановской архитектуре было предложено представлять команды в виде двоичного кода. Код программы, записанный заранее¹⁸ на перфокарты или магнитную ленту, можно было ввести в машину достаточно быстро.

Поскольку команды программы и данные по форме представления стали одинаковыми, их можно хранить в *единой памяти*¹⁹ вместе с данными. Не существует принципиальной разницы между двоичными кодами машинной команды, числа, символа и т.д. Это утверждение иногда называют **принципом однородности памяти**. Из него следует, что команды одной программы могут быть получены как результат работы другой. Именно так текст программы на языке высокого уровня переводится (транслируется) в машинные коды конкретной машины.

Код программы может сохраняться во внешней памяти (например, на дисках) и затем загружаться в оперативную память для повторных вычислений. Благодаря простоте замены программ, ЭВМ стали универсальными устройствами, способными решать самые разнообразные задачи в произвольном порядке и даже одновременно.

Принцип программного управления. Любая обработка данных в вычислительной машине происходит по программе. Принцип программного управления определяет наиболее общий механизм автоматического выполнения программы.

¹⁷ Например, память большого объема требует многоадресного адреса, что, в свою очередь, приводит к очень большому количеству линий связи. В итоге приходится как-то изменять способ адресации, например, передавать адрес по частям.

¹⁸ До появления персональных компьютеров для этого использовались специальные *устройства подготовки данных*. Такая схема ускоряла процесс ввода и исключала простои ЭВМ, связанные с длительным набором программ.

¹⁹ Известна также так называемая *гарвардская архитектура*, в которой программы и данные хранятся в разных областях памяти. Несмотря на повышение надежности, она не получила широкого распространения.

Важным элементом устройства управления в машине фон-неймановской архитектуры является специальный регистр – *счетчик адреса команд*²⁰. В нем в любой момент хранится адрес команды программы, которая будет выполнена следующей.

Используя значение из счетчика, процессор считывает из памяти очередную команду программы, расшифровывает ее и выполняет. Затем процесс повторяется для следующей команды и т.д. Процессор выполняет команды по следующему алгоритму (его часто называют *основным алгоритмом работы процессора*):

- 1) из ячейки памяти, адрес которой записан в счетчике адреса команд, выбирается очередная команда программы; на время выполнения она сохраняется в специальном *регистре команд*;
- 2) значение счетчика адреса команд увеличивается так, чтобы он указывал на следующую команду;
- 3) выбранная команда выполняется (например, при сложении двух чисел оба слагаемых считываются в АЛУ, складываются и результат операции сохраняется в регистре или ячейке памяти);
- 4) далее весь цикл повторяется сначала.

Таким образом, автоматически выполняя одну команду программы за другой, компьютер может исполнить любой линейный алгоритм. Для того, чтобы в программе можно было использовать ветвления и циклы, необходимо нарушить естественную последовательность выполнения команд. Для этого существуют специальные *команды перехода*, которые на этапе 3 заносят в счетчик адреса новое значение – адрес перехода. Чаще всего в программах используется *условный переход*, то есть переход происходит только при выполнении определенного условия.

Легко понять, что запуска основного алгоритма работы процессора в счетчик адреса команд должно быть предварительно занесено начальное значение – адрес первой выполняемой команды. В первых ЭВМ оператор вводил этот адрес вручную. В современных компьютерах при включении питания в счетчик аппаратно заносится некоторое значение, которое указывает на начало программы, хранящейся в ПЗУ. Эта программа тестирует устройства компьютера и приводит их в рабочее состояние, а затем загружает в ОЗУ *начальный загрузчик* операционной системы (как правило, с диска). Ему и передается дальнейшее управление, а стартовая программа из ПЗУ завершает свою работу. Начиная с этого момента, поведение компьютера уже определяется установленным на нем программным обеспечением (см. гл. 6).

Чтобы ускорить выполнение программы, основной алгоритм работы процессора был значительно усовершенствован. Идея была заимствована из конвейерного производства, где несколько рабочих одновременно выполняют различные операции (каждый над своим экземпляром изделия). Аналогично в современных микропроцессорах для каждого этапа выполнения команды создан отдельный аппаратный блок. Выполнив свою операцию, он передает результаты следующему блоку, а сам начинает выполнять очередную команду.

Проще всего понять этот механизм на примере первого этапа – выборки команды из ОЗУ. Специализированный блок выборки извлекает из памяти последовательно расположенные команды, не дожидаясь окончания их обработки. Прочитанные команды размещаются в специальной рабочей памяти внутри микропроцессора. В итоге, когда первая из выбранных команд будет завершена, за следующей не придется обращаться к ОЗУ, т.к. она уже находится во внутренней памяти микропроцессора. Учитывая, что обращение к ОЗУ занимает значительно больше времени, чем пересылка данных внутри процессора, такая *опережающая выборка* значительно ускоряет выполнение программы.

На практике применение конвейерного метода не так просто. Например, следующую команду часто не удастся выполнить, поскольку она использует результат предыдущей, или сразу нескольким командам потребуется одновременно обратиться к ОЗУ. Тем

²⁰ В различных процессорах этот регистр может называться по-разному: например, в семействе *Intel* он обозначается *IP = Instruction Pointer* (указатель на инструкцию).

не менее, этот метод широко применяется в микропроцессорах. В некоторых моделях используются параллельные конвейеры, так что в некоторых случаях к моменту завершения выполнения одной команды уже готов результат следующей.

5.2.4. Что называют архитектурой

Описанные фон Нейманом и его соавторами классические принципы построения вычислительных устройств применялись во всех поколениях ЭВМ. В дополнение к ним в каждом конкретном семействе (*PDP*, *EC ЭВМ*, *Apple*, *IBM PC* и др.) формулируются свои собственные принципы устройства, благодаря которым обеспечивается аппаратная и программная совместимость моделей. Для пользователей это означает, что все существующие программы будут работать и на новых моделях того же семейства компьютеров. В литературе общие принципы построения конкретного семейства компьютеров называют *архитектурой*. К архитектуре обычно относят:

- принципы построения системы команд и их кодирование;
- форматы данных и особенности их машинного представления;
- алгоритм выполнения команд программы;
- способы доступа к памяти и внешним устройствам;
- возможности изменения конфигурации оборудования.

Стоит обратить внимание на то, что архитектура описывает именно общее устройство вычислительной машины, а не особенности изготовления конкретного компьютера (набор микросхем, тип жесткого диска, емкость памяти, тактовая частота). Например, наличие видеокарты как устройства для организации вывода информации на дисплей входит в круг вопросов архитектуры. А вот является ли видеокарта частью основной платы компьютера или устанавливается на нее в виде отдельной платы, с точки зрения архитектуры значения не имеет. Иначе могло бы получиться, что для интегрированной в плату видеокарты потребовалась бы отдельная версия графического редактора!



Контрольные вопросы

1. Найдите материалы, подтверждающие, что Джон фон Нейман не был единоличным автором «фон-неймановской» архитектуры ЭВМ.
2. *Чем еще известен Джон фон Нейман?
3. Перечислите принципы фон-неймановской архитектуры и кратко объясните каждый из них.
4. Назовите основные компоненты вычислительного устройства. Каково их назначение? Согласны ли вы с тем, что полученный набор узлов логичен и обоснован?
5. В чем состоит принцип двоичного кодирования?
6. *Найдите материалы о троичной ЭВМ «Сетунь». Сравните двоичные и троичные ЭВМ.
7. Вспомните, как кодируются в компьютере числа, тексты, графика. Соблюдается ли принцип двоичного кодирования?
8. По какому алгоритму вводимые в компьютер десятичные числа можно перевести во внутреннее двоичное представление? Как перевести обратно результаты расчёта?
9. Что такое ячейка памяти? Что такое адрес ячейки?
10. Что вы знаете о разрядности ячеек ОЗУ разных поколений?
11. Почему появилась байтовая память?
12. Можно ли заменить в ячейке памяти содержимое одного бита, не затрагивая значений соседних? Почему? (Ответ: нет, только путем считывания, обработки в процессоре и обратной записи)
13. Приведите примеры различных типов данных и назовите их разрядность. Сколько байт памяти потребуется для хранения каждого из этих типов данных?
14. Что такое иерархическая организация памяти?

15. Почему большая по объему память обычно работает медленнее, чем маленькая?
16. В чем состоит принцип хранимой программы?
17. Где может храниться программа?
18. Можно ли к нечисловым данным (символам, графическим и звуковым данным) применять арифметические операции? (Ответ: При определенных условиях можно, потому что все виды информации закодированы в виде двоичных чисел. Очень важно делать это осмысленно, учитывая закономерности кодирования данных. Вот несколько разумных примеров. Путем формального прибавления к коду символа определенной константы можно переходить от заглавных букв к строчным. Вычисляя средние значения интенсивности цветов соседних точек можно получить примерный цвет «промежуточной» точки. Умножая все значения интенсивности звукового сигнала на некоторый множитель, можно сделать общее звучание громче.)
19. Как вы понимаете фразу «любая обработка данных в вычислительной машине происходит по программе»? Чем компьютер в этом отношении отличается от простого калькулятора?
20. Сформулируйте основной алгоритм выполнения команды в компьютере.
21. Что такое счетчик адреса команд и какова его роль в основном алгоритме?
22. Опишите, что происходит в момент включения компьютера с точки зрения принципа программного управления.
23. Можно ли нарушить последовательность выполнения команд в программе? Для чего это может потребоваться?
24. Всегда ли в новом компьютере есть какая-либо программа? (Ответ: Минимальный набор программ, включая программу начальной загрузки, хранится в ПЗУ.)
25. Что такое конвейер и как он работает при выполнении программы?
26. *Почему команды перехода нарушают работу конвейера?
27. Какие из предложенных в «Предварительном рассмотрении...» принципов продолжают применяться в современных компьютерах без всяких изменений, а какие сохранились, но в несколько измененном виде? Объясните, почему потребовались эти изменения.
28. Что такое архитектура? Какие детали устройства компьютера к ней не относятся?
29. В чем преимущества единой архитектуры семейств ЭВМ для пользователей и для производителей?
30. Какие семейства вычислительных машин вы знаете?



Задачи

1. * Используя условные команды «считать байт из памяти», «записать байт в память», а также стандартные логические операции «И», «ИЛИ» и «НЕ», составьте алгоритм который в байте, хранящемся в памяти, а) установит младший бит данных в единицу, не затрагивая содержимого остальных двоичных разрядов; б) сбросит его в ноль.
2. Описанная в «Предварительном рассмотрении...» конструкция ЭВМ имела ячейки памяти, состоящие из 40 двоичных разрядов. Сколько современных байтовых ячеек потребуется для хранения числа такой же разрядности?
3. В языке Паскаль есть тип чисел **word**, значением которого являются целые положительные 16-разрядные числа. Сколько байт занимает такое число в памяти? Какое максимальное значение может иметь этот тип чисел? (65535)
4. В микропроцессорах семейства *Intel* для увеличения на единицу одного из регистров процессора используется команда, имеющая код 41_{16} . Пользуясь таблицей символов, определите, какая буква соответствует этому же самому коду. Если рассматривать этот код как целое число, чему оно будет равно в десятичной системе счисления?
5. *Изучите содержимое текстового файла, используя программу, которая способна отображать данные в виде шестнадцатеричных кодов. Попробуйте найти коды из-

вестных вам символов. Сделайте то же самое с графическим файлом: удастся ли вам найти там те же самые коды («символы»)?

6. *Найдите таблицу кодов команд процессора *Intel*. Сравните эти коды с кодами, которые вы нашли в предыдущем задании. Удалось ли найти совпадения?

5.3. Магистрально-модульная организация компьютера

5.3.1. Что значит «устройство компьютера»?

Компьютер – это пример очень сложной техники. При изучении таких систем возможно несколько разных подходов. Например, можно изучать:

- устройство конкретного экземпляра компьютера: набор микросхем, тип основной платы, конструкцию и разновидности модулей памяти и т.п.;
- семейство компьютеров, например, IBM-совместимые персональные компьютеры;
- различные конструкции компьютеров (настольные компьютеры, портативные компьютеры, карманные компьютеры);
- *функциональное устройство* компьютера, т.е. его основные узлы и способы взаимодействия между ними.

Каждый из этих подходов полезен при решении определенных задач. Так для настройки конкретного компьютера необходимо точно знать марки и параметры его устройств. Определить эти данные можно с помощью специального программного обеспечения. К сожалению, любые знания в этой области очень быстро устаревают, поскольку аппаратура постоянно меняется.

Если изучать особенности одного *семейства компьютеров*, мы получим «однобокое» представление об устройстве компьютерной техники, так как каждое семейство имеет свои особенности.

Современные компьютеры очень разнообразны и поэтому имеют самую различную *конструкцию* и внешний вид. Настольный ПК состоит из системного блока и подключенных к нему внешних устройств. Такая конструкция удобна для пользователя, поскольку все устройства можно разместить на столе так, как ему хочется.

В переносных компьютерах весь минимально необходимый набор устройств собран в одном корпусе. Сейчас такие компьютеры называют *ноутбуками* (англ. *notebook* – тетрадь, блокнот). По своим вычислительным возможностям они практически не уступают настольным ПК.

Растет популярность так называемых *нетбуков* (от слов «Интернет» и «ноутбук») – так называют очень маленькие и легкие переносные компьютеры. Кроме меньшего размера и веса, нетбуки отличаются от ноутбуков большим временем автономной работы и меньшей стоимостью. Нетбуки предназначены для пользователей, которые применяют компьютер главным образом для работы в Интернете и подготовки простых документов. Их используют люди, совершающие большое число поездок.

Карманные персональные компьютеры (КПК)²¹ умещаются на ладони. У большинства из них даже нет клавиатуры, а для ввода информации нажимают пластиковой палочкой (она называется *стилус*) на сенсорный (реагирующий на прикосновение) экран.

С другой стороны, мощные серверы и суперкомпьютеры по-прежнему собираются в виде крупных «шкафов», напоминающих ЭВМ предыдущих поколений. Наконец, нельзя не упомянуть и о бытовой электронике, которая все больше и больше приближается к традиционным компьютерам.

Разнообразие типов современных компьютеров говорит о том, что конструкция – это не самое главное. В то же время, как показано в п. 5.2, их функциональное устройство

²¹ Их называют также наладонниками (англ. *palmtop*) и PDA (англ. *Personal Digital Assistant* – персональный цифровой помощник).

практически не изменяется. Поэтому далее мы подробно рассмотрим основные узлы компьютера (процессор, память и устройства ввода и вывода) и взаимодействие между ними.

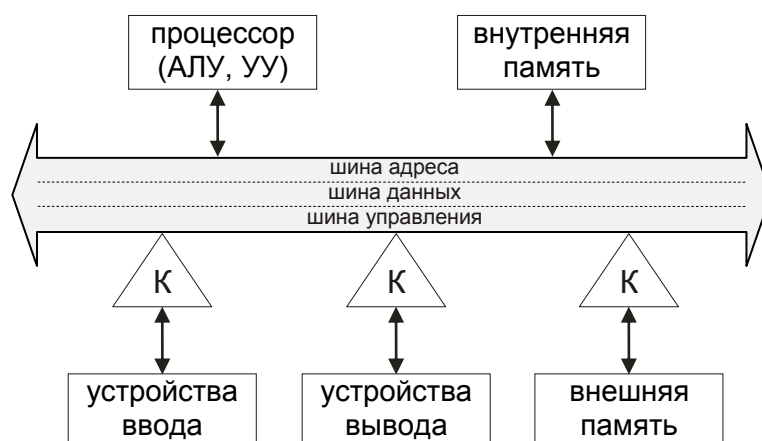
5.3.2. Взаимодействие устройств

Процессор должен обмениваться данными с внутренней памятью и устройствами ввода и вывода. Выделить отдельные каналы для связи процессора с каждым из многочисленных устройств нереально. Вместо этого сделана общая линия связи, доступ к которой имеют все устройства, использующие ее по очереди. Такой информационный канал называется *шиной*.

Шина (или *магистраль*) – это группа линий связи для обмена данными между несколькими устройствами компьютера.

Традиционно шина делится на три части:

- шина данных, по которой передаются данные;
- шина адреса, определяющая, куда именно передается информация;
- шина управления, которая организует процесс обмена (несет сигналы чтение/запись, обращение к внутренней/внешней памяти, данные готовы/не готовы и т.п.).



Рассмотрим процесс записи данных из процессора в память. Процессор выставляет на шину данных информацию для записи, на шину адреса – нужный адрес памяти, а на шину управления – сигналы для записи информации в память. Далее он вынужден ожидать, пока данные будут «взяты» с шины. В это время все остальные устройства постоянно «слушают» шину (проверяют ее состояние). В нашем примере по сигналам на шине память обнаруживает, что для нее имеются данные. Она сохраняет их по заданному адресу и должна по шине управления сообщить процессору, что операция завершена. На практике, учитывая высокую надежность работы памяти, сигнал подтверждения часто не используется: процессор просто выжидает определенное время и продолжает выполнение программы. Из этого примера понятно, что для успешного обмена данными по шине должны быть введены четкие правила (их принято называть *протоколом шины*), которые должны соблюдать все устройства.

По сравнению с первыми ЭВМ, взаимодействие процессора с внешними устройствами организуется теперь по-другому. В классической архитектуре процессор контролировал все процессы ввода-вывода. Получалось так, что быстродействующий процессор тратил много времени на ожидание при работе с значительно более медленными внешними устройствами. Поэтому появились специальные электронные схемы, которые руководят обменом информацией между процессором и внешними устройствами. В третьем по-

колении такие устройства назывались *каналами ввода-вывода*, а в четвертом – *контроллерами*²² (на схеме они обозначены буквой К).

Контроллер – это электронная схема для управления внешним устройством и для простейшей предварительной обработки данных.

Современный контроллер – это микропроцессор, предназначенный специально для обслуживания одного (или даже нескольких однотипных) устройств ввода-вывода или внешней памяти. Нагрузка на центральный процессор при этом существенно снижается, и это увеличивает эффективность работы всей системы в целом. Контроллер, собранный в виде отдельной микросхемы называют *микроконтроллером*.

В качестве примера рассмотрим контроллер современного жесткого диска. Его основная задача – по принятым от процессора координатам найти на диске требуемые данные, прочитать их и передать в ОЗУ. Но контроллер способен выполнять и другие, порой весьма нетривиальные функции. Так он сохраняет в служебной области диска информацию обо всех имеющихся на магнитной поверхности некачественно изготовленных секторах (а их при современной высокой плотности записи избежать не удастся!) и способен «на ходу» подменять их резервными, что создает видимость диска, который полностью свободен от дефектов²³.

Как видно из приведенной выше схемы, теперь данные могут передаваться между внешними устройствами и ОЗУ напрямую, минуя процессор. Кроме того, наличие шины существенно упрощает подсоединение к ней новых устройств. Архитектуру, которую можно легко расширять за счет подключения к шине новых устройств, часто называют *магистрально-модульной*.

Если спецификация на шину (детальное описание всех ее логических и физических параметров) является открытой (опубликована), то производители могут разрабатывать к такой шине любые дополнительные устройства. Такой подход называют *принципом открытой архитектуры*. При этом в компьютере предусмотрены стандартные разъемы для подключения новых устройств, удовлетворяющих стандарту. Поэтому пользователь может собрать такой компьютер, который ему нужен. Необходимо только помнить, что при подключении любого нового устройства нужно установить специальную программу – *драйвер*, которая обеспечивает обмен данными между этим устройством и процессором.

В современных компьютерах для повышения эффективности работы используется несколько шин, например, одна – между процессором и памятью, другая – от процессора к видеосистеме и т.д.

5.3.3. Обмен данными с внешними устройствами

Существуют три режима обмена данными между центральным процессором (ЦП) и внешними устройствами:

- программно-управляемый ввод/вывод;
- обмен с устройствами по прерываниям;
- прямой доступ к памяти (ПДП).

При **программно-управляемом** обмене все действия по вводу или выводу предусмотрены в теле программы. Процессор полностью руководит ходом обмена, включая ожидание готовности периферийного устройства и прочие временные задержки, связанные с процессами ввода/вывода. Достоинства этого метода – простота и отсутствие дополнительного оборудования, недостаток – большие потери времени из-за ожидания быстро работающим процессором более медленных устройств ввода/вывода.

При обмене **по прерываниям** устройства ввода-вывода в случае необходимости сами требуют внимания процессора. Например, клавиатура оповещает процессор каждый

²² Это название происходит от английского слова *control* – управление; не следует путать с русским словом «контролёр».

²³ http://spider.nrcde.ru/music/articles/hardware/hdd_outsins.html

раз, когда была нажата или отпущена клавиша; все остальное время процессор выполняет программу, вообще «не отвлекаясь» на клавиатуру. Когда прерывание произошло, ЦП «откладывает» на некоторое время выполнение основной программы и переходит на служебную программу обработки прерывания. Завершив его обработку, ЦП снова возвращается к тому месту программы, где она оказалась прервана. При этом основная программа даже «не заметит» возникшей задержки. Этот режим обмена более сложен, но зато значительно эффективнее – процессор не тратит время на ожидание.

Представим себе, что в кабинете начальника идет совещание, и в этот момент по телефону поступает важная информация, требующая немедленного принятия решения. Секретарша, не дожидаясь конца совещания, сообщает начальнику о звонке. Тот, прервав свое выступление, снимает трубку, выясняет суть дела и сообщает свое решение. Затем он продолжает совещание, как ни в чем не бывало. Здесь роль ЦП играет начальник, а телефонный звонок – это запрос (требование) на прерывание. «Секретарша» в компьютере тоже предусмотрена – это контроллер прерываний, анализирующий и сортирующий все поступающие прерывания с учетом их важности (*приоритета*).

Механизм прерываний используется не только в аппаратной части, но и в программах, которые основаны на обработке *событий* (нажатий на клавиши, команд управления от мыши и т.п.). Такая технология лежит в основе современных операционных систем и применяется в системах разработки программ *MS Visual Studio*, *Visual Basic*, *Delphi*, *Lazarus* и им подобных.

В обоих описанных выше вариантах управление обменом выполнял центральный процессор. Именно он извлекал из памяти выводимые данные (или записывал туда вводимые), подсчитывал их количество и полностью контролировал работу шины. Если передаваемые данные не требуют сложной обработки, ЦП напрасно расходует время на проведение обмена. Чтобы освободить процессор от этой работы и увеличить скорость передачи *крупных блоков* данных от устройства ввода в память и обратно, применяется **прямой доступ к памяти** (ПДП, англ. *DMA = Direct Memory Access*).

Принципиальное отличие ПДП состоит в том, что в этом режиме процессор не *производит* обмен, а только *подготавливает* его, программируя контроллер ПДП: устанавливает режим обмена, а также передает начальный адрес ОЗУ и количество циклов обмена. Далее контроллер в ходе ПДП самостоятельно наращивает первое значение и уменьшает второе, что позволяет освободить центральный процессор.

Изложенный материал о режимах ввода/вывода может быть сведен в таблицу (здесь УВВ обозначает устройство ввода-вывода):

вид обмена	начинает обмен	руководит обменом	текущая программа	программа обмена
программный	ЦП	ЦП	программа обмена – часть текущей программы	
прерывания	УВВ	ЦП	прерывается	специальная подпрограмма
ПДП	УВВ, ЦП	контроллер ПДП	выполняется параллельно	отсутствует (обмен идет аппаратно)



Контрольные вопросы

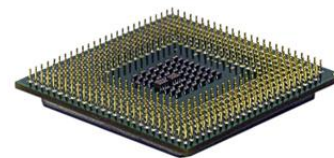
1. Как можно определить, какие именно платы и устройства установлены в вашем компьютере? Для чего это может потребоваться?
2. Как вы думаете, что более полезно для глубокого понимания работы компьютера: изучение функционального устройства компьютера или изучение его конструкции?
3. Как устройства компьютера обмениваются данными?
4. Что такое шина? Почему обмен данными между устройствами компьютера с помощью шины оказался наилучшим решением?
5. Из каких частей состоит шина? Охарактеризуйте каждую из них.
6. Что такое магистрально-модульная архитектура и в чем ее главное достоинство?

7. В чем заключается принцип открытой архитектуры?
8. Используя приведенное в тексте объяснение процесса записи данных в память, попробуйте объяснить, как происходит считывание данных из ячейки памяти с заданным адресом.
9. Что такое контроллер и для чего он нужен?
10. Объясните, как использование контроллеров позволяет повысить быстродействие компьютера в целом.
11. Сравните магистрально-модульную архитектуру компьютера с классической. Выделите наиболее перегруженный блок на каждой из схем.
12. Почему в современном компьютере несколько шин?
13. Что требуется для успешного присоединения к компьютеру нового устройства?
14. Расскажите о разных режимах обмена данными с внешними устройствами.
15. Как расшифровывается сокращение ПДП и что это такое?
16. Как выполняется обмен данными в режиме ПДП?
17. Предложите наиболее подходящий режим обмена данными с клавиатурой. (Ответ: лучше всего подходит обмен по прерываниям; при программном обмене ЦП будет тратить слишком много времени на напрасный опрос клавиатуры, а для ПДП у «медленной» клавиатуры просто не хватит данных.)
18. Какой режим лучше всего подходит для обмена данными с жестким диском? (Ответ: режим ПДП.)
19. Где в программировании применяются принципы обработки прерываний?

5.4. Процессор

Центральным устройством, во многом определяющим возможности компьютера, является процессор.

Процессор – это блок, предназначенный для автоматического считывания команд программы, их расшифровки и выполнения.



Вид микропроцессора со стороны выводов

Название «процессор» происходит от английского глагола «to process» – обрабатывать. Иными словами, процессор – это блок компьютера, который автоматически обрабатывает информацию по заданной программе.²⁴

Процессор, изготовленный в виде большой или сверхбольшой интегральной схемы (БИС, СБИС), называется *микропроцессором*.

Любой процессор обязательно включает в себя две важные части, каждая из которых решает свои задачи:

- *арифметико-логическое устройство* (АЛУ), выполняющее обработку данных, и
- *устройство управления* (УУ), которое управляет выполнением программы и обеспечивает согласованную работу всех узлов компьютера.

5.4.1. Арифметико-логическое устройство

В простейшем случае АЛУ состоит из двух регистров, сумматора и схем управления операциями (об устройстве сумматора и регистров можно прочитать в главе 3). При выполнении операций в регистры помещаются исходные данные, а в сумматоре они складываются.

Как показано в главе 4, все остальные арифметические операции могут быть тем или иным способом сведены к сложению. Тем не менее, нередко для ускорения умножения и деления инженеры идут на усложнение АЛУ. Например, в процессорах широко использу-

²⁴ Не следует путать процессор как аппаратный блок с мощными программами обработки, которые также часто называют процессорами (например, текстовый или табличный процессор).

ется метод умножения чисел с использованием таблиц, в которых записаны готовые произведения небольших чисел.

АЛУ не только выполняет вычисления, но и анализирует полученный результат. Обычно проверяется два свойства: равенство нулю (совпадение всех разрядов сумматора с нулем) и отрицательность результата (определяется проверкой знакового разряда – см. главу 4). Результаты этого анализа заносятся в определенные биты *регистра состояния* процессора. Используя эти значения, можно сделать вывод об истинности или ложности условий $R = 0$, $R \neq 0$, $R > 0$, $R < 0$, $R \geq 0$, $R \leq 0$, где R обозначает результат операции. Это позволяет организовать ветвления в программе, например, для неотрицательного числа вычислять квадратный корень, а иначе – выдать сообщение об ошибке.

Как правило, АЛУ работает только с целыми числами. Операции с вещественными числами выполняются в *математическом сопроцессоре*, который встроен внутрь современных микропроцессоров.

5.4.2. Устройство управления

Главная задача устройства управления – обеспечить автоматическое выполнение последовательности команд программы в соответствии с основным алгоритмом работы процессора (см. 5.2.3). УУ выполняет следующие действия:

- извлечение из памяти очередной команды;
- расшифровка команды, определение необходимых действий;
- определение адресов ячеек памяти, где находятся исходные данные;
- занесение в АЛУ исходных данных;
- управление выполнением операции;
- сохранение результата.

Таким образом, выполнение каждой машинной команды состоит из элементарных действий, которые называются *микрокомандами*.

В зависимости от сложности, машинная команда может быть выполнена за различное число микрокоманд. Например, пересылка числа из одного внутреннего регистра микропроцессора требует значительно меньшего числа действий, чем умножение. Команды, работающие с оперативной памятью, выполняются дольше, чем команды, работающие только с регистрами процессора.

Каждая из микрокоманд машинной инструкции запускается с помощью управляющего импульса. Опорную последовательность импульсов для этих целей УУ получает от *генератора тактовых импульсов*. Интервал между двумя соседними импульсами называется *тактом*.

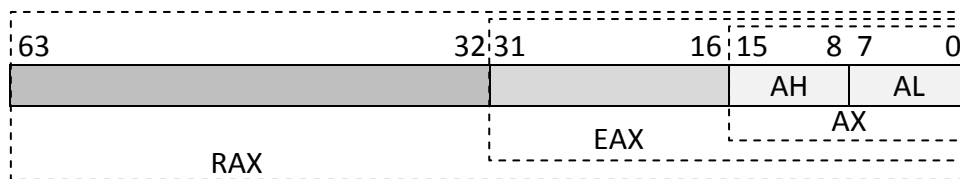
Если две микрокоманды полностью независимы друг от друга, то их можно выполнить одновременно (за один такт), даже если они принадлежат к разным командам программы. Такая оптимизация широко применяется в современных процессорах для организации конвейерной обработки ради увеличения быстродействия.

5.4.3. Регистры процессора

Кроме регистров АЛУ и УУ, в микропроцессоре есть много других регистров. Большинство из них – внутренние, они недоступны программисту. Однако есть несколько регистров, специально предназначенных для использования программным обеспечением. Их часто называют *регистрами общего назначения* (РОН), подчеркивая тем самым универсальность их функций. В РОН могут храниться не только сами данные (числа, коды символов и т.д.), но и адреса ячеек памяти, где эти данные находятся. Например, если требуется обработать последовательные ячейки памяти, то к содержимому такого регистра нужно каждый раз прибавлять размер ячейки.

Количество регистров и их устройство в разных процессорах отличается друг от друга. Например, в процессорах семейства *Intel* имеется небольшой набор 64-разрядных

РОН. Ради обеспечения программной совместимости со старыми (32- и 16-разрядными) процессорами, эти РОН имеют вложенную структуру, напоминающую матрешку:



На рисунке показана структура 64-разрядного регистра RAX. Его младшие 32 бита (с нулевого по 31-й) образуют регистр EAX для 32-разрядных вычислений. К младшим 16 битам EAX (0-15), в свою очередь, можно также обращаться как к самостоятельному регистру AX. Наконец, биты 0-7 и 8-15 образуют два 8-разрядных регистра AH и AL. Отчетливо видно, что наращивание разрядности процессоров семейства Intel происходило постепенно. Такая структура регистров обеспечивает совместимость с предыдущими моделями и позволяет процессору легко обрабатывать 8-, 16-, 32- и 64-разрядные данные.

Кроме рассмотренного выше регистра RAX, в процессорах Intel имеются также аналогичным образом устроенные регистры RBX, RCX и RDX, а также некоторые другие. Это регистры неравноценны, по справочникам можно определить, как и с каким регистром работает та или иная команда.

5.4.4. Основные характеристики процессора

Как вы уже знаете, для организации выполнения команд в компьютере есть генератор импульсов, каждый из которых «запускает» очередной такт машинной команды. Очевидно, что чем чаще следуют импульсы от генератора, тем быстрее будет выполняться операция. Следовательно, *тактовая частота*, измеряемая количеством тактовых импульсов в секунду, может быть характеристикой быстродействия процессора.

Тактовая частота – количество тактовых импульсов за одну секунду.

В настоящее время тактовая частота измеряется в гигагерцах, т.е. в миллиардах (10^9) импульсов за секунду. Эту частоту нельзя установить сколь угодно высокой, поскольку процессор может просто не успеть выполнить действие очередного такта до прихода следующего импульса.

Нужно понимать, что использовать тактовую частоту для сравнения быстродействия процессоров можно только в том случае, если оба процессора устроены одинаково. Например, если какая-то команда в одном из процессоров выполняется за два такта, а в другом – за три, то при равенстве частот первый будет работать в полтора раза быстрее.

Приближенно можно считать, что процессор выполняет за один такт одну простую команду (типа пересылки регистр-регистр). Тогда при тактовой частоте 4 ГГц за 1 сек выполняется около 4 миллиардов таких операций. Это примерная оценка, потому что при конвейерном методе скорость выполнения команд сильно зависит от множества факторов, например, от порядка следования команд в программе.

Другая характеристика, позволяющая судить о производительности процессора, – это его *разрядность*.

Разрядность – это максимальное количество двоичных разрядов, которые процессор способен обрабатывать за одну команду.

Чаще всего разрядность определяют как размер регистров процессора в битах.

Однако, важны также разрядности шины данных и шины адреса, которые поддерживает процессор. *Разрядность шины данных* – это максимальное количество бит, которое может быть считано за одно обращение к памяти. *Разрядность шины адреса* – это количество адресных линий; она определяет максимальный объем памяти, который способен поддерживать процессор. Этот объем памяти часто называют величиной *адресного пространства*, он вычисляется по формуле 2^R , где R – количество разрядов шины адреса.

Все три разрядности могут не совпадать. Так, у процессора Pentium II были 32-разрядные регистры, разрядность шины данных – 64 бита, а шины адреса – 36 бит.

5.4.5. Система команд процессора

Каждая модель процессора имеет собственную систему команд. Поэтому, как правило, процессоры могут выполнять только программы, написанные специально для них. Тем не менее, обычно новые процессоры одной и той же серии (например, процессоры *Intel*) поддерживают все команды предыдущих моделей.

В системах команд разных процессоров есть много общего. Они обязательно включают следующие группы машинных команд:

- команды передачи (копирования) данных;
- арифметические операции;
- логические операции, например, «НЕ», «И», «ИЛИ», «исключающее ИЛИ»;
- команды ввода и вывода;
- команды переходов.

Существует два основных подхода к построению системы команд процессора:

- процессоры с полным набором команд (англ. *CISC* = *Complex Instruction Set Computer*);
- процессоры с сокращенным набором команд (англ. *RISC* = *Reduced Instruction Set Computer*).

CISC-процессоры содержат широкий набор разнообразных команд. При этом на скорость их выполнения обращают меньшее внимание, главное – удобство программирования. При разработке RISC-процессоров набор команд, наоборот, весьма ограничен, но это позволяет значительно ускорить их выполнение. Многие современные процессоры (например, процессоры *Intel*) – гибридные, у них полный набор команд, которые выполняются RISC-ядром. Это позволяет совместить достоинства обоих подходов.

Почти все инструкции, входящие в систему команд компьютера, состоят из двух частей – *операционной* и *адресной*. Операционная часть – *код операции* – указывает, какое действие необходимо выполнить. Адресная часть описывает, где хранятся исходные данные и куда поместить результат. Часто исходные данные для команды (содержимое регистра или ячейки памяти, константа) называют *операндами*.

Рассмотрим для примера одну из наиболее простых команд процессора *Intel*, которая состоит из четырех байт и имеет шестнадцатеричный код **81 C2 01 01**. Она может быть разбита на три неодинаковые по длине части:

- код операции **81C** обозначает сложению регистра с константой;
- первый операнд **2** – это условное обозначение регистра DX;
- константа **0101**, которая добавляется к регистру.

Отметим, что система команд процессоров *Intel* очень сложна и плохо подходит для изучения в школьном курсе информатики.



Контрольные вопросы

1. Для чего нужен процессор? Почему он так называется?
2. Какие узлы входят в состав процессора? Зачем нужны АЛУ и УУ?
3. Как выполняется АЛУ в простейшем случае? Как в АЛУ используется сумматор?
4. Почему удобно, что АЛУ автоматически сравнивает результат действия с нулем?
5. Подумайте, как с помощью логических операций с битами сумматора установить факт его равенства или неравенства нулю.
6. Для чего служит математический сопроцессор?
7. Какую роль играет УУ в автоматическом выполнении программ?
8. Как называется элементарное действие в машинной команде?
9. Зачем нужен генератор тактовых импульсов?

10. Что такое РОН? Для каких целей он может использоваться?
11. *Найдите информацию о регистрах процессора *Intel*. Постарайтесь разобраться в назначении наиболее важных из них.
12. Что такое тактовая частота и как она влияет на быстродействие компьютера?
13. Тактовые частоты двух процессоров, изготовленных фирмами *Intel* и *AMD* равны. Означает ли это, что их быстродействие одинаково?
14. *Объясните, как применение конвейера влияет на количество команд, выполняемых за один такт.
15. На что влияет разрядность процессора? Какие разновидности разрядности вы знаете? Что характеризует каждая из них?
16. Какие группы операций входят в систему команд любого процессора?
17. Что такое RISC- и CISC-процессоры? Чем они отличаются?
18. Какие части можно выделить в команде процессора?



Задачи

1. Обозначим символом Z бит, определяющий факт равенства результата R нулю, а символом N – бит, фиксирующий отрицательность R :
 - $Z=1$ при $R=0$, и $Z=0$ в противном случае;
 - $N=1$ при $R<0$, и $N=0$ в противном случае.Напишите логическое выражение, включающее значения N и Z , которое дает:
 - а) 1 при $R \leq 0$, и 0 в противном случае;
 - б) 1 при $R > 0$, и 0 в противном случае.
2. Оцените, сколько миллиардов простых операций типа пересылки регистр-регистр может выполнить за 1 мин. процессор с тактовой частотой 1 ГГц.
3. Сопоставьте тактовую частоту процессора с максимальной частотой звуковых колебаний, которые слышит человек. Что можно сказать о возможностях современного компьютера в обработке звуковой информации?
4. Какое максимальное десятичное целое число без знака можно поместить в 32-разрядный регистр? (Ответ: $2^{32}-1 = 4\,294\,967\,295$)
5. Сколько символов, закодированных в двухбайтной кодировке UNICODE, можно загрузить одновременно в 64-разрядный регистр?
6. Процессор Pentium II имеет 36-разрядную шину адреса. Какой объем памяти он может адресовать? (Ответ: 64 Гб)

5.5. Память

Как мы уже знаем, процессор способен выполнять программу, но ее команды хранятся в памяти. Таким образом, память – это другое устройство, без которого вычислительный автомат не может быть построен. Кроме того, память одновременно используется (что не менее важно) и для хранения обрабатываемых данных.

Память – это устройство компьютера, которое используется для записи, хранения и выдачи по запросу команд программы и данных.

Существует большое количество видов памяти, которые различаются по устройству, организации, функциям и т.д. Обычно выделяют *внутреннюю* и *внешнюю* память. Термины эти имеют историческое происхождение, связанное с конструкцией первых ЭВМ: одна часть памяти находилась внутри главного шкафа (в котором размещался процессор), а другая – вне его.

Современные компьютеры, конечно, выглядят совсем по-другому, из-за чего названия утратили свою прежнюю наглядность. Тем не менее, деление памяти на два типа по-прежнему сохраняется. Различие между ними кроется, прежде всего, в назначении. Внутренняя память предназначена для хранения программ и данных, которые используются для задач, решаемых в данный момент. А внешняя память служит для того, чтобы сохра-

нить данные на длительный срок, пока они не потребуются, именно поэтому ее еще часто называют *долговременной*.

5.5.1. Внутренняя память

Внутренняя память – часть памяти компьютера, которая используется для хранения программ и данных во время решения задачи.

Часто ее называют *основной памятью*. В состав внутренней памяти входят ОЗУ и ПЗУ.

Внутренняя память строится в соответствии с базовыми принципами, описанными ранее в п.5.2. Основное отличие внутренней памяти от внешней – произвольный доступ к отдельным ячейкам памяти по их адресам (обращение к внешней памяти происходит иначе, см. далее).

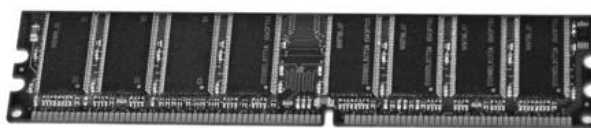
Информация, хранящаяся в ОЗУ, считается временной (оперативной), поэтому пользователь должен сам сохранять необходимые данные во внешней памяти.

Часто говорят, что при выключении питания информация в ОЗУ пропадает. Строго говоря, это не совсем правильно, поскольку существуют элементы памяти, способные сохранять свое состояние даже после отключения питания. Однако при повторном включении (или перезагрузке) компьютера программное обеспечение неспособно восстановить, где и какая информация находилась «в прошлый раз». Именно поэтому если при наборе текста перезагрузить компьютер, работу придется повторять заново.



Магнитное ОЗУ: слева – биты памяти, справа – устройство выборки нужного адреса

Внутренней память может быть построена на основе самых разных технологий. Самые первые ЭВМ имели ОЗУ на электронно-лучевых трубках, причем их количество соответствовало разрядности памяти (каждый бит числа считывался из отдельной трубки). Затем появилась память на магнитных сердечниках. Намагниченное состояние сердечника соответствовало единичному состоянию бита, ненамагниченное – нулевому. Заметим, что данные в магнитных ячейках памяти полностью сохранялись и после выключения питания. Наконец, развитие микроэлектроники позволило изготовить компактную полупроводниковую память, которая сейчас и применяется в ПК.



Модуль полупроводниковой памяти

Существуют два типа оперативной памяти, отличающиеся по технологии изготовления – *статическая* и *динамическая*. Первая строится на триггерах (об устройстве триггера говорилось в главе 3), а вторая – на полупроводниковых конденсаторах. Конденсатор намного проще и меньше триггера, так что на одном и том же кристалле можно сделать гораздо больше запоминающих элементов динамического типа, чем статического. Поэтому динамическая память имеет *большую* емкость и меньшую стоимость, чем статическая. К сожалению, у нее есть очень существенный недостаток: она работает намного медленнее статической. Сейчас в персональных компьютерах используется динамическая оперативная память.

Что касается ПЗУ, то технологии их изготовления также постепенно совершенствовались. Первоначально информация в ПЗУ заносилась только на заводе. Затем появились *программируемые* ПЗУ, которые потребитель мог заполнить сам, поместив «чистую» («пустую») микросхему в специальное устройство – *программатор*. В некоторых микросхемах этого типа в качестве запоминающих элементов использовали тонкие токопроводящие дорожки.

дающие перемычки. Наличие перемычки означало единицу. Программатор мощными импульсами тока пережигал нужные перемычки, тем самым устанавливая биты в нулевое состояние²⁵. Очевидно, что процесс записи информации таким способом был необратимым.

Позднее появились *перепрограммируемые ПЗУ*, в которых очистка информации сначала производилось ультрафиолетовыми лучами, а затем – с помощью электрических импульсов. Современные перепрограммируемые ПЗУ используют *флэш-память*. Каждый элемент такой памяти изготовлен на основе особой разновидности транзисторов, так что это тоже полупроводниковая память. Изменить содержимое такого ПЗУ можно даже без программатора, запустив специальную программу.



Флэш-BIOS на плате компьютера

Внутри компьютеров семейства *IBM PC* есть еще один особый вид памяти – *память конфигурации* (CMOS-память). В ней хранятся разнообразные настройки аппаратного обеспечения, а также часы и календарь, благодаря которым компьютер всегда знает текущую дату и время. Данные сохраняются благодаря питанию от небольшой батарейки. CMOS-память – это особая память, которая не входит в адресное пространство внутренней памяти. Поэтому к ней невозможно обратиться просто по адресу, и в этом смысле она скорее похожа на внешнюю память. Для работы с памятью конфигурации в ПЗУ современного ПК предусмотрена специальная программа (она называется *BIOS Setup*), причем работать с ней можно только до загрузки операционной системы (при включении компьютера).

5.5.2. Внешняя память

Внешняя память – часть памяти компьютера, которая используется для долговременного хранения программ и данных.

Этот вид памяти позволяет повторно использовать программы и данные. Благодаря этому текст достаточно набрать один раз, а цифровые фотографии можно рассматривать в течение многих лет.

К внешней памяти относятся разнообразные устройства хранения данных, начиная от накопителей на магнитных дисках и кончая современными внешними запоминающими устройствами на основе полупроводниковой флэш-памяти.

Любой тип внешней памяти состоит из некоторого носителя информации (например, диска или полупроводникового кристалла) и электронной схемы управления (*контроллера*).

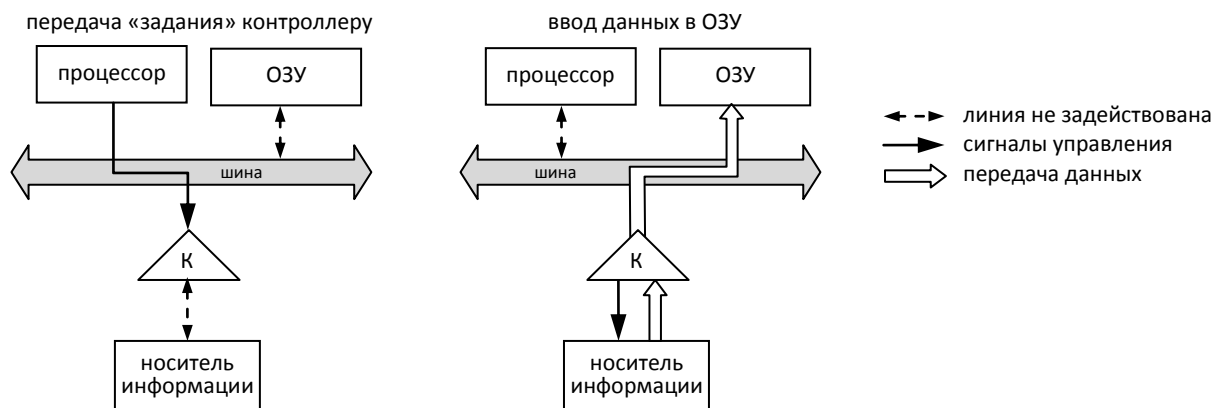
Машинный носитель информации – это средство длительного хранения данных в компьютерном формате. Носитель может быть съемным (как в накопителях на оптических дисках), а может быть помещён внутрь неразборного устройства (жесткий магнитный диск – винчестер).

В переносных устройствах внешней памяти, например, во внешних винчестерах и флэш-накопителях, носитель и схема управления объединены в единый блок. Такие устройства подключаются к компьютеру снаружи через разъем.

Центральный процессор не может непосредственно обращаться к данным на носителе, он работает с ними через контроллер внешней памяти. На рисунке схематично показано, как читаются данные с внешнего носителя информации в ОЗУ²⁶.

²⁵ Так сгорают плавкие предохранители в бытовой аппаратуре.

²⁶ В действительности процесс обмена более сложен, в нём участвует еще и контроллер ПДП.

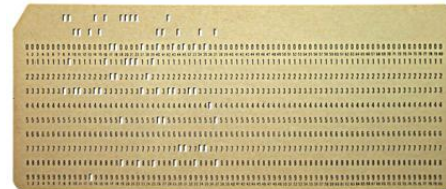


Для связи с контроллером процессор использует *порты* – регистры контроллера, к которым процессор может обратиться по номеру. Процессор передает контроллеру «задание» на передачу данных, и контроллер берет руководство процессом на себя. В это время центральный процессор может параллельно выполнять программу дальше или решать другую задачу. Таким образом, выполнить чтение (и запись) данных из внешней памяти гораздо сложнее, чем из внутренней памяти.

Для внешней памяти характерны следующие черты:

- обменом данными управляют контроллеры;
- прежде чем процессор сможет непосредственно использовать программу или данные, хранящиеся во внешней памяти, их нужно предварительно загрузить в ОЗУ;
- данные располагается блоками (на дисках их принято называть *секторами*); блок данных читается и пишется как единое целое, что существенно ускоряет процедуру обмена; работать с частью блока невозможно.

В качестве внешней памяти используются самые разные носители. Первоначально программы и данные сохранялись на бумажных *перфокартах* и *перфолентах*. Подписанные обычной ручкой или карандашом, они сортировались программистами вручную. Затем произошел переход к магнитным носителям: магнитным лентам, барабанам и дискам.



Перфокарта

На *магнитных дисках* биты данных хранятся в виде небольших намагниченных (или ненамагниченных) областей. Сектора размещаются на концентрических окружностях (имеющих общий центр), которые называются *дорожками*. Поскольку длина дорожки зависит от положения на диске, количество секторов на дорожках может быть неодинаковым. Доступ к секторам диска – произвольный, максимальная скорость достигается тогда, когда читаемые или записываемые сектора располагаются подряд.

Управление такой сложной системой очень трудоемко – поэтому, как нам уже известно из истории вычислительной техники, появление магнитных дисков вызвало создание специального ПО для работы с ними – операционных систем (ОС). ОС берет на себя все технические детали, предоставляя пользователю работу с некоторыми наборами данных – *файлами*. Таким образом, начиная с дисковых накопителей, наличие файловой системы – это характерная черта внешней памяти, которая существенно отличает ее от внутренней.

Следующей технологией хранения информации стали *оптические компакт-диски* (англ. *CD = Compact Disk*). При записи данных (одним из способов) луч лазера «выжигает» на поверхности диска дорожку, в которой чередуются впадины и возвышения. При считывании также применяется луч лазера, только меньшей интенсивности, чтобы не разрушить данные. Для распознавания нулей и единиц используется различное отражение от перепадов глубины и ровной поверхности диска. В отличие от магнитных дисков, где ин-

формация хранится в виде на отдельных замкнутых дорожках, данные на оптическом диске записываются вдоль непрерывной спирали, как на старых грампластинках²⁷.

Сейчас широко используются оптические диски следующих поколений: DVD (англ. *Digital Versatile Disk* – цифровой многоцелевой диск, емкость до 17 Гбайт) и *Blu-ray*-диски (емкостью до 66 Гбайт). Они имеют тот же диаметр, что и *CD*-диски, но для повышения плотности записи используют лазер с меньшей длиной волны.

Были разработаны также комбинированные магнитооптические диски. Носителем информации в них служит магнитное вещество, которое при нагреве лазером ориентируется в магнитном поле и меняет оптические свойства поверхности диска. После восстановления нормальной температуры такие диски необычайно устойчивы к внешним воздействиям. Тем не менее, они не получили распространения из-за высокой стоимости и малой скорости записи.

Отметим, что на всех видах дисков есть разметка на сектора, благодаря которой контроллер может быстро находить нужную информацию. Сами данные помещаются между «заголовком» сектора и его завершающей записью.

Наконец, последнее достижение в области устройств внешней памяти – запоминающие устройства на базе флэш-памяти. В ней нет движущихся частей, а носителем информации служит полупроводниковый кристалл. Данные во флэш-памяти обновляются только блоками, но для устройств внешней памяти это вполне естественно. Максимальное количество перезаписей данных для каждого блока хотя и велико, но все же ограничено. Поэтому встроенный контроллер при записи использует специальный алгоритм для выбора свободных блоков, стараясь загружать сектора диска как можно более равномерно.

Кроме широко распространенных флэш-дисков («флэшек»), этот вид памяти используется в картах памяти для фотоаппаратов, плееров и сотовых телефонов, а также в твердотельных винчестерах (англ. *SSD = Solid State Disk*). Напомним, что ПЗУ также может изготавливаться на базе флэш-памяти.



Флэш-карта

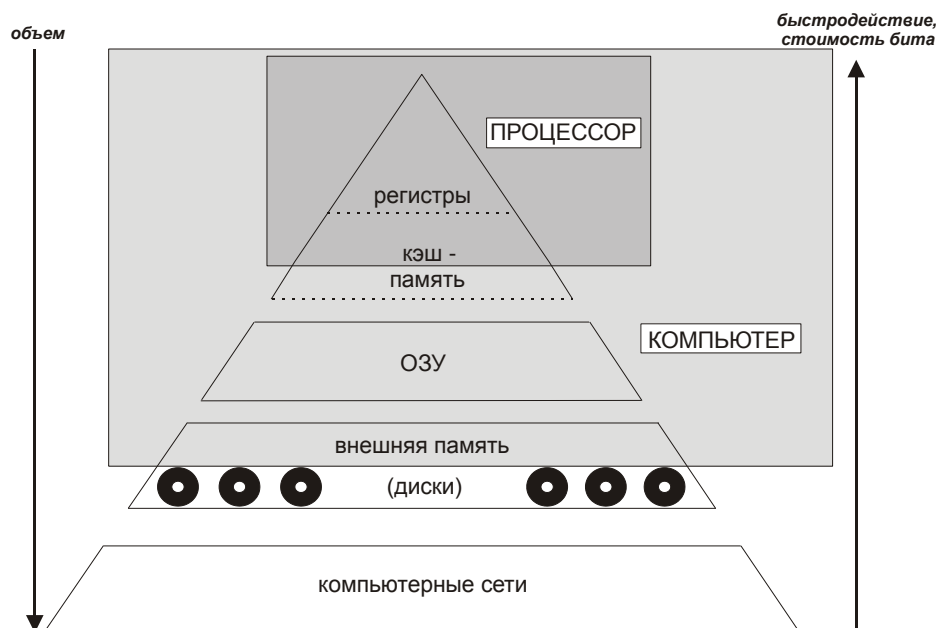
5.5.3. Взаимодействие разных видов памяти

Итак, мы познакомились с разными видами внутренней и внешней памяти. Осталось разобраться, как они взаимодействуют между собой.

Иерархия памяти. Кэширование. Как следует из обсуждения в п.5.2.2, невозможно создать память, которая имела бы как большой объем, так и высокое быстродействие. Поэтому используют многоуровневую (иерархическую) систему из нескольких типов памяти. Как правило, чем больший объем имеет память, тем медленнее она работает.

Самая быстрая (и очень небольшая) память – это регистры процессора. Гораздо больше по объему, но заметно медленнее, внутренняя память (ОЗУ и ПЗУ). Далее следует огромная, но еще более медленная внешняя память. Наконец, последний уровень – это данные, которые можно получить из компьютерных сетей.

²⁷ В отличие от грампластинок, спираль раскручивается от центра к краям.



Для редактирования файла с диска (внешняя память) программа обработки загружает его в ОЗУ (внутренняя память), а конкретные символы, с которыми в данные доли секунды работает процессор, «поднимаются» по иерархии выше – в регистры процессора.

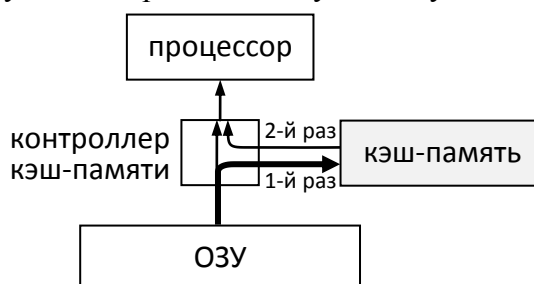
Производительность компьютера в первую очередь зависит от «верхних» уровней памяти – процессорной памяти и ОЗУ. Быстродействие процессоров значительно выше, чем скорость работы ОЗУ, поэтому процессору приходится ждать, пока до него дойдут данные из оперативной памяти. Чтобы улучшить ситуацию, между процессором и ОЗУ добавляют еще один слой памяти, который называют кэш-памятью (от англ. *cache* – тайник, прятать).

Кэш-память – это память, ускоряющая работу другого (более медленного) типа памяти, за счёт сохранения прочитанных данных на случай повторного обращения к ним.

Кэш-память – это статическая память, которая работает значительно быстрее динамического ОЗУ. В ней нет собственных адресов, она работает не по фон-неймановскому принципу адресности.

При чтении из ОЗУ процессор обращается к контроллеру кэш-памяти, который хранит список всех ячеек ОЗУ, копии которых находятся в кэше. Если требуемый адрес уже есть в этом списке, то запрашивать ОЗУ не нужно, и контроллер передает процессору значение, связанное (*ассоциированное*) с этим адресом²⁸. Такой принцип организации памяти называется *ассоциативным*.

Если нужных данных нет в кэш-памяти, они читаются из ОЗУ, но одновременно попадают и в кэш – при следующем обращении их уже не нужно читать из ОЗУ.



Обычно в кэш-память заносится содержимое не только запрошенной ячейки, но и ближайших к ней (эта стрелка на рисунке показана более толстой линией). Таким образом, в кэше хранятся копии часто используемых ячеек ОЗУ, и передача этих данных в процессор происходит быстрее.

²⁸ Это напоминает поиск в Интернете содержимого документа по его названию.

В работе кэш-памяти есть две основные трудности. Во-первых, объем кэш намного меньше объема ОЗУ и он быстро заполняется – приходится заменять наиболее «ненужные» (например, редко используемые) данные. Во-вторых, если считанные из кэш-памяти данные обрабатываются процессором и сохраняются в ОЗУ, нужно обновлять и содержимое кэша. Обе эти задачи решает контроллер кэш-памяти. Несмотря на трудности, кэширование во многих случаях повышает скорость выполнения программы в несколько раз.

Сама кэш-память также строится по многоуровневой схеме: в современных процессорах есть, по крайней мере, 2-3 уровня. Некоторые из них входят в состав процессора, а остальные выполнены в виде отдельных микросхем (поэтому на схеме многоуровневой памяти кэш только частично расположен внутри процессора). Кэш для программы и для данных изготавливается отдельно. Это удобно потому, что считываемую программу, в отличие от данных, не принято изменять, поэтому кэш команд можно делать проще.

Подчеркнем, что термин «кэширование» в вычислительной технике имеет довольно широкий смысл: речь идет о сохранении информации в более быстродействующей памяти с целью повторного использования. Например, браузер кэширует файлы, полученные из Интернета, сохраняя их на винчестере в специальной папке. В накопителе на жестком диске также используется кэширование. Таким образом, кэш может быть организован как с помощью аппаратных средств (кэш процессора), так и программно (кэш браузера).

Виртуальная память. Пользователям хочется, чтобы программное обеспечение было интеллектуальным и дружелюбным, и чтобы в нем были предусмотрены все самые мелкие детали, которые им (пользователям) могут потребоваться. Программистам хочется написать программу с наименьшими затратами сил и времени, поэтому они широко используют среды быстрой разработки программ (англ. *RAD = Rapid Application Development*). В результате программы все больше увеличиваются в размере. Кроме того, объем обрабатываемых данных постоянно растет. Поэтому компьютерам требуется все больше и больше памяти, особенно в многозадачном режиме, когда одновременно запускается сразу несколько программ.

Как же согласовать эти требования с ограниченным объемом ОЗУ? Современные операционные системы используют для этого идею *виртуальной памяти*. Предполагается, что компьютер обладает максимально допустимым объемом памяти, а реально установленное ОЗУ – лишь некоторая *часть* этого пространства. Оставшаяся часть размещается в специальном системном файле или отдельном разделе жесткого диска. Если емкости ОЗУ не хватает для очередной задачи, система копирует «наименее нужную» (дольше всего не использовавшуюся) часть ОЗУ на диск, освобождая необходимый объем памяти. Когда, наоборот, потребуются данные с диска, они будут возвращены в освобожденное таким же образом место ОЗУ (и это совсем не обязательно будет то самое первоначальное место!).

При использовании виртуальной памяти выполнение программ замедляется, но зато они могут выполняться на компьютере с недостаточным объемом ОЗУ. В этом случае установка дополнительного ОЗУ может повысить быстродействие во много раз.

Использование виртуальной памяти еще раз подтверждает, что деление памяти на внутреннюю и внешнюю память – это искусственная мера. Она вызванная тем, что невозможно создать идеальную память, удовлетворяющую всем требованиям сразу.

5.5.4. Основные характеристики памяти

Для пользователя важны, прежде всего, объем памяти, ее быстродействие и стоимость.

Информационная емкость – это максимально возможный объем данных, который может сохранить данное устройство памяти.

Емкость памяти измеряется в тех же самых единицах, что и объем информации, т.е. в битах, байтах и производных единицах (чаще всего – в мегабайтах или гигабайтах).

Для дисков часто говорят о *форматированной* и *неформатированной* емкости. Первая величина – это объем «полезной» памяти, а вторая включает еще и ту область диска, которую занимает служебная разметка.

Для оценки *быстродействия* памяти используют несколько величин. Любая операция обмена данными включает не только саму передачу данных, но и подготовительную часть. Это может быть, например, поиск нужного сектора диска или установка адреса внутри микросхемы ОЗУ. Время подготовки соизмеримо со временем передачи, так что пренебрегать им нельзя. Общее время обмена данными от начала подготовки до окончания передачи называют *временем доступа*.

Время доступа – интервал времени от момента посылки запроса информации до момента получения результата на шине данных.

При измерении этой величины обычно рассматривают самый сложный случай, когда данные считываются или записываются в случайных местах памяти. На практике байты или сектора часто читаются по порядку, поэтому время ввода или вывода уменьшается.

Для ОЗУ время доступа измеряется в наносекундах ($1 \text{ нс} = 10^{-9} \text{ с}$), а для винчестеров – в миллисекундах ($1 \text{ мс} = 10^{-3} \text{ с}$). Такая разница связана с тем, что дисковод должен сначала переместить считывающую головку в нужное положение.

Поскольку устройства внешней памяти работают с целыми блоками данных, для их характеристики требуется какой-то дополнительный показатель.

Средняя скорость передачи данных – это количество передаваемых за единицу времени данных после непосредственного начала операции чтения (т.е. без учета подготовительной стадии).

Эта характеристика обычно измеряется в мегабайтах в секунду (Мбайт/с).

Для оценки стоимости памяти используют отношение стоимости модуля памяти к его информационной емкости. Часто говорят о стоимости одного бита или *стоимости одного гигабайта*.

Для дисковых накопителей часто указывают *частоту вращения* (в оборотах в минуту). Чем быстрее вращается диск, тем выше может быть скорость считывания и записи.



Контрольные вопросы

1. Зачем в компьютере нужна память?
2. С какой целью память делится на внутреннюю и внешнюю?
3. Верно ли, что внешняя память располагается вне корпуса компьютера? Приведите примеры. (Ответ: нет, например, жесткий диск находится внутри системного блока).
4. Назовите все виды компьютерной памяти, которые вы знаете. Зачем они используются? Какими свойствами обладают?
5. К каким видам памяти применим принцип адресности фон Неймана?
6. Что означает термин «произвольный доступ к памяти»?
7. Зачем нужно ПЗУ в компьютере? Можно ли при необходимости изменить его содержимое на домашнем компьютере?
8. Что такое носитель информации? Какие носители вы можете назвать?
9. Какими носителями внешней памяти вы пользовались? Каков их объем и какую примерно его часть вы использовали?
10. Перечислите все известные вам виды дисков.
11. Что такое сектор диска?
12. Можно ли считать с диска отдельно взятый байт? Как его все-таки получить? (Ответ: только программным путем: считать весь сектор в буфер в ОЗУ и скопировать оттуда требуемый байт).
13. Какую роль играет контроллер при считывании данных с диска?

14. Почему любую программу перед выполнением требуется загрузить в оперативную память?
15. Что такое флэш-память, и в каких запоминающих устройствах она используется?
16. Какие разновидности полупроводникового ОЗУ существуют? Что служит в них запоминающим элементом? Каковы свойства названных вами типов ОЗУ?
17. Для чего создана кэш-память, как она работает и как повышает производительность компьютера?
18. Может ли программа обращаться к ячейкам кэш-памяти? Подумайте, относится ли кэш-память к архитектуре компьютера? Почему? (Ответ: не может; детали, недоступные программисту, обычно не относят к архитектуре.)
19. *Почему кэш называют ассоциативной памятью? Сравните с человеческой памятью, которую тоже часто называют ассоциативной.
20. Кэширование – это аппаратный или программный прием? Приведите примеры.
21. Перечислите все известные вам уровни иерархии компьютерной памяти и кратко охарактеризуйте их. Как меняются объем и быстродействие памяти при переходе на другой уровень иерархии (вверх или вниз)?
22. Почему компьютерам требуются все большие объемы памяти?
23. Как работает механизм виртуальной памяти?
24. *Чем ограничен объем виртуальной памяти?
25. Какие основные характеристики используются для памяти? В каких единицах они измеряются?
26. Какая характеристика используется только для внешней памяти?



Задачи

1. Определите информационный объем каждого вида памяти в вашем домашнем компьютере (ОЗУ, кэш-память, жесткий диск, примерный суммарный объем CD-дисков с данными и т.п.). Оцените отношение объемов этих уровней памяти.
2. Сравните приведенные в тексте данные о времени доступа для ОЗУ и дисковых накопителей: на сколько порядков они различаются?
3. Пусть емкость жесткого диска составляет 137 438 953 472 байт. Пользуясь калькулятором, переведите этот объем в гигабайты по правилам, принятым в информатике. Сравните ответ с тем числом, которое получится, если коэффициенты при переводе принять равными 1000 (так делают, например, в физике: в 1 кг ровно 1000 г!). Какие конфликты в связи с этим могут возникнуть (и реально возникали!) у потребителей и фирм-изготовителей жестких дисков? (Ответ: объем составляет 128 Гб при делении на 1024 и 137 Гб при «десятичной» оценке, т.е. отличие превышает 7%)

5.6. Устройства ввода

5.6.1. Что относится к устройствам ввода?

Информация в компьютер может вводиться с помощью самых разнообразных устройств, но не каждое из них называют устройством ввода. Например, *не являются* устройствами ввода устройства внешней памяти, рассмотренные в предыдущем разделе. Прием данных по сети также не является вводом, поскольку здесь (как и в случае внешней памяти) данные уже были введены ранее и сохранены в компьютерном формате, а теперь просто копируются.

Но, даже исключив из рассмотрения все перечисленные устройства, мы по-прежнему будем иметь довольно пёструю картину. Сравните между собой подключенный к компьютеру датчик температуры, веб-камеру, осуществляющую съемку для круглосудочной трансляции городского пейзажа в Интернет, мышку, которой пользователь запус-

кает программы или выбирает из меню требуемое действие, и, наконец, клавиатуру, с помощью которой можно не только набрать текст, но и отдавать команды компьютеру.

Устройством ввода называется устройство, которое:

- позволяет человеку отдавать компьютеру команды и/или
- выполняет первичное преобразование данных в форму, пригодную для хранения и обработки в компьютере.

К устройствам ввода относятся

- клавиатура;
- манипуляторы: мышь, трекбол, сенсорная панель, джойстик, трекпойнт;
- сканер;
- микрофон, видеокамера и другие источники мультимедийных данных;
- световое перо и графический планшет – специализированные устройства ввода графической информации;
- датчики.

Заметим, что некоторые устройства ввода, например, датчики и веб-камеры, работают без непосредственного участия человека.

5.6.2. Клавиатура

Одним из первых устройств ввода была клавиатура. С ее помощью человек вводит в компьютер текст. Текст может быть записью числа: тогда компьютер по программе преобразует текстовую строку в соответствующее двоичное число, с которым может работать процессор.

Кроме символьных клавиш, на клавиатуре есть *дополнительные* (управляющие) клавиши. Значения некоторых из них жёстко задано (например, клавиши управления курсором, *Page Up*, *Home*, *Delete*, *Print Screen* и др.), функции других (в первую очередь, функциональных клавиш *F1-F12*) программист может назначить сам. Клавиши *Shift*, *Caps Lock*, *Ctrl* и *Alt* изменяют результат нажатия остальных клавиш. С их помощью можно, например, вводить заглавные буквы.

В простейших типах клавиатур при нажатии клавиши соединяются два контакта и замыкается электрическая цепь. Роль контактов в наиболее распространенных моделях играет специальное токопроводящее напыление, наносимое на гибкую изолирующую полимерную пленку. Более качественные клавиатуры могут использовать, например, *герконы* (герметичные контакты), срабатывающие от приближающегося к ним магнита. Еще один вариант – это емкостные клавиатуры, где при нажатии клавиши сближаются две небольшие пластины, образующие конденсатор. Емкостные клавиатуры более долговечны, так как в них нет механического контакта деталей.

Работой современной клавиатуры руководит встроенный в нее микроконтроллер, который:

- опрашивает все клавиши и фиксирует изменение их состояния: нажатие или отпускание;
- временно (до момента передачи в центральный процессор) хранит коды нескольких последних нажатых или отпущенных клавиш (*скан-коды*)²⁹;
- при наличии данных посылает требование прерывания центральному процессору и затем (по его запросу) передает имеющиеся данные;
- управляет световыми индикаторами клавиатуры;
- выполняет диагностику неисправностей клавиатуры.

Контроллер клавиатуры выполняет лишь минимальную обработку информации: в компьютер уходят исключительно данные о нажатии или отпускании клавиши с заданным

²⁹ Скан-коды представляют собой номера клавиш и не имеют ничего общего с кодовыми таблицами символов, изученными в главе 2

номером. Распознавание кода набранного символа с учетом состояния клавиш сдвига выполняет программа, принимающая данные. Такое решение в очередной раз показывает, что аппаратная часть компьютера всегда делается максимально универсально, а все особенности работы компьютера определяются программным обеспечением.

Клавиатура имеет определенные технические характеристики, такие как усилие нажатия клавиш (в ньютонах) и ход клавиш (в миллиметрах).

5.6.3. Манипуляторы

Для ввода команд и данных в компьютер широко используются *манипуляторы* – разнообразные по конструкции устройства, воздействуя на которые (путем их перемещения, давления на их чувствительную поверхность и т.п.), пользователь может управлять компьютером, не набирая текста.

Общая идея работы манипуляторов заключается в следующем. На экране монитора отображается указатель (*курсор*), с помощью которого человек может «показать» компьютеру интересующий его объект. Манипулятор перемещает курсор вслед за определенными перемещениями руки человека. Когда курсор установлен в нужное место экрана, человек сообщает об этом компьютеру, обычно нажимая кнопку на манипуляторе. В принципе манипулятор позволяет даже набирать тексты, используя нарисованную на экране клавиатуру.

Самый распространенный манипулятор – *компьютерная мышь*. Это название принято связывать с кабелем («хвостом»), соединяющим устройство с компьютером. Многим современным мышам «хвост» уже не нужен: они передают данные о своем движении с помощью электромагнитных волн (к компьютеру при этом подсоединяется специальное устройство для приема и декодирования радиоволн). Такие мыши более удобны, хотя стоят дороже и используют дополнительные элементы питания (батарейки или аккумуляторы).

Первоначально датчики движения мыши были механическими: при перемещении мыши вращался находящийся внутри нее шарик. Шарик, в свою очередь, вращал два взаимно перпендикулярных колесика, и их поворот фиксировался электронным устройством. Полученная информация об изменении координат передавалась в компьютер. Такая механическая конструкция была неудобна, так как шарик и колесики приходилось часто очищать от пыли и грязи.

Оптические мыши, которые используются сейчас, не содержат механических частей, поэтому они долговечны и обладают высокой точностью. Расположенная «под брюхом» миниатюрная видеокамера снимает изображение поверхности стола через небольшие промежутки времени (для подсветки используется светодиод или портативный лазер). Сравнивая полученные картинки, специальный микропроцессор вычисляет перемещение мыши по двум осям координат. Этот метод даёт плохие результаты, когда поверхность очень гладкая и однородная (например, стекло). В таких случаях значительно лучше работают лазерные мыши, потому что подсветка лазером дает более контрастное изображение.

Наиболее интересная характеристика оптической мыши – это *разрешение оптического сенсора* (видеокамеры). Оно определяется как количество точек, которые способны различить устройство на отрезке заданной длины. Чем выше разрешение, тем точнее мышь способна отслеживать перемещение (это важно, например, при точной обработке изображений в графическом редакторе). Разрешение обычно измеряется в точках на дюйм (англ. **dpi** = **dots per inch**). Обычное разрешение мыши – около 1000 dpi, а у некоторых особо «точных» экземпляров – в несколько раз больше.

Кроме разрешения, на качество работы мыши влияет количество кадров, которые делает видеокамера за одну секунду (до десяти тысяч). Размеры каждого кадра определя-

ются датчиком, обычно они находятся в пределах от 16×16 до 30×30 пикселей³⁰. Зная эти данные, можно найти скорость обработки изображения в мегапикселях в секунду (Мп/с). Для игровых мышей важна также *максимальная скорость движения* – она может достигать нескольких метров в секунду.



Так выглядит трекбол
(www.mousearena.com)

Шаровой манипулятор – *трекбол* – это перевернутая мышь. Его чувствительный элемент – закрепленный шар, который вращается вокруг своего центра. Название «трекбол» происходит от английских слов *track* – направляющее устройство и *ball* – шар. Для портативных компьютеров он удобнее мыши, потому что не требует дополнительного ровного пространства. Кроме того, трекболы могут работать там, где есть вибрация. Сейчас трекболы практически не используются.

В ноутбуках в качестве встроенного «заменителя» мыши устанавливают еще один тип манипулятора – *сенсорную панель* (англ. *touchpad*), воспринимающая движение по ней пальца. Панель состоит из небольшой чувствительной к давлению поверхности и двух кнопок. Короткое касание чувствительной панели заменяет щелчок мышью (можно использовать также кнопки рядом с панелью). Современные панели способны воспринимать не только перемещение, но и другие команды. Например, для прокрутки документа можно проводить пальцем вдоль правой или нижней границы панели (там, где в окне принято располагать полосы прокрутки). Некоторые панели даже способны анализировать сочетания движений по ним двух пальцев.

«Менее серьезный» манипулятор – *джойстик* (англ. *joy stick* – «веселая» рукоятка) – используется, в основном, в компьютерных играх и может быть оформлен самым причудливым образом. Джойстик имеет ручку, при повороте которой внутри корпуса замыкаются контакты, соответствующие направлению наклона ручки. В некоторых моделях дополнительно установлен датчик давления, и чем сильнее пользователь наклоняет ручку, тем быстрее движется указатель по экрану.

В некоторых ноутбуках в середине клавиатуры устанавливается *трекпойнт* (это слово можно перевести с английского как указатель курса или маршрута). Трекпойнт – это кнопка, которая определяет направление давления пальца и преобразует эту команду в перемещение курсора на экране.



Трекпойнт
(www.globalnerdy.com)

5.6.4. Сканер

Сканер – это устройство для ввода в компьютер графической информации.

С его помощью можно преобразовать в компьютерные данные рисунки, фотографии, снимки на фотопленке (негативы и слайды), а также получить снимки объектов не слишком большой толщины.

Часто при помощи сканера в компьютер вводят офисные документы. При этом сканер передает в компьютер изображение документа в виде картинку. Чтобы отсканированный текст можно было редактировать, нужно превратить эту картинку в коды символов. Для этого используют программы оптического распознавания символов (англ. *OCR = Optical Character Recognition*). OCR-программы пытаются «угадать» в пикселях рисунка очертания букв, и определить, какие это буквы, сверяя контуры с имеющимися у них об-

³⁰ <http://computer.howstuffworks.com/mouse5.htm>,
<http://home.comcast.net/~richardlowens/OpticalMouse/>

разцами. В принципе, можно распознавать и рукописный текст, однако программы справляются с ним значительно хуже, чем с печатным (подумайте почему?).

Принцип работы сканера достаточно прост. Луч света от яркого источника пробегает вдоль сканируемой поверхности, а светочувствительные датчики при этом воспринимают отраженные лучи и определяют их интенсивность и цвет. Можно сказать, что сканер – это очень сильно упрощенный цифровой фотоаппарат.

Сканеры могут иметь разную конструкцию:

- *ручные*, где считывающую головку перемещает пользователь (вспомните, как считываются штрих-коды);
- *планшетные*, в которых неподвижный объект кладется на стекло, а светочувствительная головка перемещается внутри сканера;
- *рулонные*, протягивающие бумагу с изображением мимо неподвижной головки;
- *барабанные*, где сканируемый объект наклеивается на вращающийся барабан, который медленно вращается мимо неподвижной головки; такие сканеры обеспечивают наилучшее качество сканирования и применяются в издательской деятельности.

Сканеры часто объединяют в одном корпусе с лазерным принтером, копировальным аппаратом и факсом – получается *многофункциональное устройство* (МФУ).

Самая важная характеристика сканера – разрешающая способность.

Разрешающая способность – это максимальное количество точек на единицу длины, которые способен различить сканер.

Разрешающая способность сканера измеряется в пикселях на дюйм (англ. *ppi = pixels per inch*). При сканировании совсем не обязательно устанавливать максимально возможное разрешение. Конечно, чем оно выше, тем лучше качество, но зато и файл займет больше места на диске! Рекомендуемое разрешение зависит от того, зачем сканируется материал (см. таблицу).

Применение	Разрешение, ppi
Сканирование в отраженном свете:	
иллюстрации для Web-страниц	75-150
сканирование текста без распознавания	150-200
сканирование текста для распознавания	300-400
цветное фото для печати на струйном принтере	200
цветное фото для типографской печати	не менее 300
Сканирование в проходящем свете:	
35-мм пленка, для Web-страниц	200-600
35-мм пленка, для печати на струйном принтере	600-2000

Другая важная характеристика режима сканирования – *глубина цвета* (разрядность), то есть количество двоичных разрядов, которое используется для кодирования цвета одного пикселя (см. главу 2). Если для кодирования цвета использовано N двоичных разрядов, то общее количество возможных цветов равно 2^N . Высококачественные устройства позволяют сканировать изображения с глубиной цвета 48 бит.

5.6.5. Цифровые датчики

Датчик – устройство, регистрирующее какую-либо физическую величину и преобразующее ее в сигналы (обычно электрические).

Компьютер способен не просто хранить большое количество данных, полученных от многочисленных датчиков, но и проводить их математическую обработку. Таким образом, на основе компьютера может быть построена мощная цифровая лаборатория.

Многие датчики вырабатывают аналоговые данные. Поэтому для их подсоединения к компьютеру необходимо устройство, преобразующее аналоговые сигналы в цифровые – *аналого-цифровой преобразователь* (АЦП).



Контрольные вопросы

1. Зачем нужны устройства ввода?
2. Можно ли сетевую карту, через которую компьютер получает данные, назвать устройством ввода? Почему?
3. Перечислите все известные вам устройства ввода. С какими из них вы работали?
4. Что можно ввести в компьютер с помощью клавиатуры?
5. Что такое управляющие клавиши? Зачем они используются?
6. Что такое функциональные клавиши?
7. Нажатие одной и той же клавиши вызывает разную реакцию компьютера в зависимости от состояния клавиш сдвига – *Shift*, *Alt* и *Ctrl*. Сколько различных команд можно ввести с помощью одной клавиши?
8. Работая в электронной таблице, пользователь нажал клавиши «3», «2» и «1». Какие операции должен выполнить компьютер, чтобы соответствующее число было записано в память? Указание: вспомните пример 1 из п. 4.3.4. (Ответ: анализ правильности записи числа, преобразование кодов символов в значения цифр, перевод десятичной записи числа в двоичное число)
9. Зачем в клавиатуре установлен микроконтроллер?
10. Почему клавиатура не передает в компьютер готовые коды символов?
11. Как выполняется локализация (использование национальных символов) на клавиатуре? Найдите сведения по этому вопросу в Интернете.
12. Что такое манипулятор?
13. Какие разновидности манипуляторов вы знаете? С какими из них непосредственно работали? Сравните приемы работы с ними. Обоснуйте свой ответ.
14. Как устроена компьютерная мышь?
15. *В старых моделях мышей вращение колесиков фиксировалось с помощью источника света и фотодатчика: при прохождении прорези на диске датчик фиксировал появление света. Докажите, что с одним таким датчиком можно зафиксировать движение и даже измерить его скорость, но нельзя определить направление вращения. Предложите, что нужно сделать, чтобы его узнать? (Поставить рядом два датчика).
16. Каким образом движение мыши управляет перемещением курсора на экране?
17. Вспомните все известные вам приемы работы с мышью.
18. Что такое беспроводная мышь?
19. Что такое трекбол и как он работает?
20. Что такое сенсорная панель?
21. Как устроен джойстик?
22. Что такое трекпойнт?
23. Что можно делать с помощью сканера?
24. Можно ли с помощью сканера получить фотографию реального объекта?
25. Как происходит распознавание отсканированного текста?
26. Какие бывают сканеры по конструкции?
27. Назовите наиболее важные характеристики сканеров.
28. Какое устанавливать разрешение при сканировании? На что оно повлияет?
29. Что такое датчики? Каковы возможности компьютера в автоматизации эксперимента?



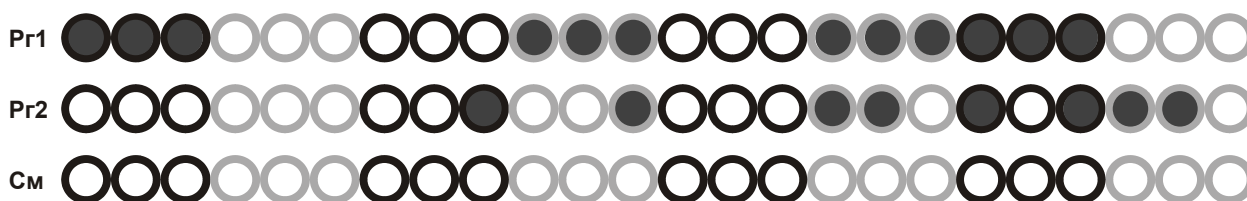
Задачи

1. Видеокамера оптической мыши имеет 16×16 пикселей, за одну секунду обрабатывается 9000 кадров. Рассчитайте скорость обработки данных в мегапикселях? (Ответ: около 2,3 Мп/с)
2. Вычислите, сколько точек получится при сканировании изображения 10×10 см с разрешением 300 ppi? Оцените объем полученного файла при сканировании в режиме 256 оттенков серого. Прodelайте аналогичную оценку для режима 24-битного цвета.

5.7. Устройства вывода

Устройства вывода – это устройства, которые представляют компьютерные данные в форме, доступной для восприятия человеком.

Первыми устройствами вывода были панели индикаторных лампочек. Каждая из них показывала состояние отдельного бита: горящая лампочка обозначала единицу, а выключенная – ноль. Для чтения результата нужно было хорошо знать двоичную систему.



Этот схематический рисунок изображает индикаторную панель на пульте ЭВМ первых поколений. Разноцветные колпачки патронов с лампочками помогали правильно считывать результат: каждая группа из трёх битов – это одна восьмеричная цифра. Если считать, что горящие лампочки на рисунке обозначены тёмным цветом, то в регистре P_2I читается восьмеричное число 70070770_8 , а сумматор $См$ очищен (заполнен нулями).

Такие панели использовались для обслуживающего персонала вплоть до третьего поколения ЭВМ, однако для большинства пользователей такой вывод данных был непонятен. Первые «настоящие» устройства вывода печатали числа в десятичном виде на бумагу. Затем печатающие устройства научились печатать не только цифры, но и буквы. Они работали по принципу печатающей машинки: рельефный шаблон символа ударял по красящей ленте, прижатой к бумаге, и оставлял отпечаток.

Кроме устройств, печатающих символы, появились графопостроители (плоттеры), которые рисовали перьями на бумаге графики функций и простейшие картинки. Так как современные принтеры могут выводить текст и графику (в том числе и цветную), специальные устройства для вывода графики практически не используются.

Революционным событием стало создание мониторов. Это позволило избавиться от ненужного расхода бумаги – теперь можно было выводить на печать только самое необходимое. Кроме того, управление и обслуживание ЭВМ стало более удобным.

Компьютеры четвертого поколения начали обрабатывать мультимедийную информацию – звуковые и видеоданные. Поэтому к компьютерам стали подключать устройства для вывода такой информации: звуковые колонки, наушники, телевизор и т.п. Некоторые из этих устройств (например, наушники) – аналоговые, поэтому для вывода необходимо преобразовать дискретные компьютерные данные в аналоговую форму. Для этого используется специальная электронная схема, которую называют *цифро-аналоговый преобразователь* (ЦАП). ЦАП для вывода звуковой информации входит в состав звуковой карты.

Эволюция устройств вывода не остановилась – все время разрабатываются устройства новых типов, порой весьма экзотические. Например, появились сообщения о создании «3D-принтеров», которые способны под управлением компьютера создавать объемные тела из различных материалов (прежде всего, из пластика).

5.7.1. Монитор

Компьютерный монитор состоит из дисплея (панели, на которую смотрит человек) и электронных схем, позволяющих выводить на этот дисплей текстовую и графическую информацию.

Мониторы во многом используют телевизионные технологии. В конце XX века для компьютеров применялись мониторы на основе электронно-лучевых трубок (ЭЛТ), но они были вытеснены жидкокристаллическими (ЖК) мониторами, которые обладают рядом *преимуществ*:

- малый вес и размеры;
- в 2-4 раза меньшее потребление электроэнергии;
- нет искажений изображения, характерных для электронно-лучевых трубок;
- значительно ниже уровень электромагнитного излучения.

Тем не менее, некоторые профессионалы по-прежнему работают на электронно-лучевых мониторах. Дело в том, что ЖК-мониторы имеют *недостатки*, которые трудно устранить в производстве:

- цветопередача хуже, чем у ЭЛТ-мониторов; например, очень трудно получить чисто черный цвет;
- контраст и цвета изображения меняются в зависимости от угла, под которым мы смотрим на монитор;
- при быстром изменении изображения заметно «запаздывание» (жидкие кристаллы не могут поворачиваться слишком быстро);
- при существующих технологиях изготовления у многих мониторов есть дефектные точки, которые не работают (так называемые «битые пиксели»);
- могут отображать четкую картинку только в одном разрешении, совпадающем с размерами матрицы.

Независимо от конструкции, экран любого монитора строится из отдельных точек, причем каждая из них образована близко расположенными областями трех основных цветов – красного, зеленого и синего. Для ЖК-монитора эти области имеют форму прямоугольников, слегка вытянутых по вертикали. Расстояние между их центрами – доли миллиметра, поэтому глаз человека воспринимает все три составляющие как одну точку «суммарного» цвета.

Элементы экрана часто называют пикселями, что не совсем точно. Строго говоря, пиксель – это точка рисунка, а не экрана. Например, на одном и том же мониторе можно легко устанавливать разные размеры рабочего стола в пикселях. Компьютер «проектирует» пиксели картинки на экран с учетом установленного разрешения. При этом одному пикселю может соответствовать одна или несколько точек экрана.

Элемент ЖК-экрана – это жидкий кристалл, способный под воздействием электрического напряжения менять свои оптические свойства. Каждая точка управляется своим полупроводниковым транзистором. Подчеркнем, что сам жидкий кристалл не способен светиться, он лишь регулирует пропускание света от расположенного за ним источника света.

Наиболее важные характеристики мониторов – это *размер диагонали* (в дюймах) и *максимальное разрешение* (количество точек экрана по ширине и высоте). Для ЖК-мониторов максимальное разрешение – это количество элементов матрицы. Если установить другое (более низкое) разрешение, то качество изображения будет хуже, т.к. видеосистеме придется «растягивать» картинку на реально существующие точки.

Процессор передает данные для вывода *видеокарте* (*видеоконтроллеру*), которая управляет выводом изображения на монитор. Современные видеокарты содержат микропроцессор для обработки графической информации (*графический ускоритель*) и собственную *видеопамять*. Можно считать, что видеокарта – это специализированный компь-

ютер, который существенно ускоряет построение и вывод на монитор графических изображений, особенно трехмерных.

Инженеры активно работают над созданием все более совершенных устройств вывода. Большое внимание уделяется разработке так называемых 3D-дисплеев, которые смогут отображать информацию в трех измерениях.

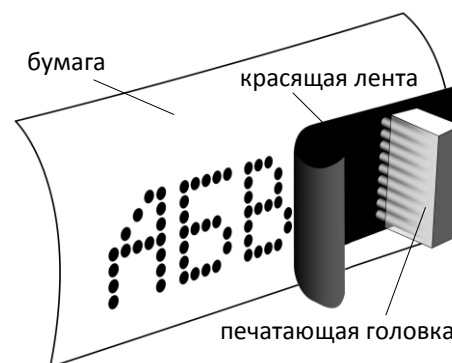
5.7.2. Печатающие устройства

Печатающие устройства (*принтеры*) служат для вывода текстовой и графической информации на бумагу или плёнку. Современные принтеры обрабатывают символы как графику, т.е. рисуют их. На принтерах можно печатать очень сложные изображения, в том числе цветные фотографии.

В настоящее время существует четыре основных типа принтеров: матричные, струйные, лазерные и сублимационные.

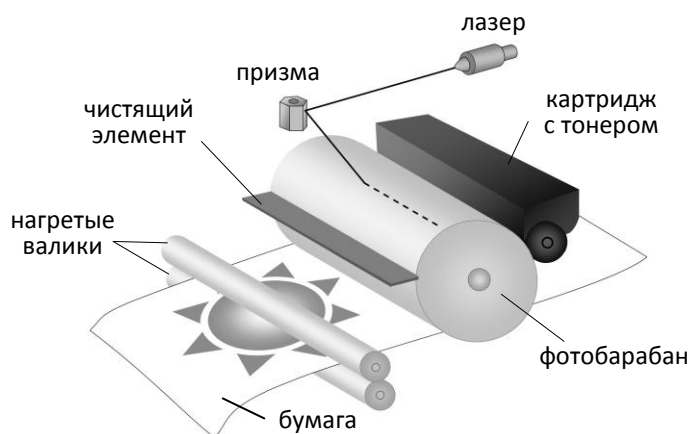
Матричные принтеры – это последнее поколение принтеров с ударным принципом работы. Печатающая головка содержит вертикальный ряд иглол, которые под воздействием управляющих сигналов ударяют по красящей ленте, оставляя на бумаге отпечатки в виде маленьких точек.

Головка движется в горизонтальном направлении, что позволяет сформировать строку из символов произвольного вида. На таком принтере можно получать не только тексты, но черно-белые рисунки, однако вывод графики происходит очень медленно. Достоинства матричных принтеров – дешевизна самих принтеров и расходных материалов (красящих лент), а также способность печатать практически на любой бумаге. Однако они не могут обеспечить высокое качество печати, работают медленно и сильно шумят.



Печатающая головка *струйных* принтеров содержит крошечные отверстия, через которые под большим давлением на бумагу выбрасываются чернила. Диаметр получаемых при этом точек гораздо меньше, чем у матричных принтеров, что позволяет получить значительно лучшее качество печати. В цветных принтерах чаще всего устанавливается два картриджа: один с черной краской, а второй – с голубой, фиолетовой и желтой (вспомните цветовую модель CMYK). Изображение строится из точек этих цветов. В некоторых моделях для повышения качества используют шесть базовых цветов. Для печати на струйных принтерах необходима качественная бумага, кроме того, напечатанное изображение расплывается при попадании воды.

Лазерные принтеры обеспечивают очень высокое качество печати. Компьютер строит в памяти полный образ страницы и передает его принтеру. Тот с помощью лазерного луча построочно переносит изображение на вращающийся барабан – строит электростатическую копию картин. Затем к барабану притягиваются мелкие частицы красящего порошка – *тонера*, причем, чем сильнее наэлектризован участок барабана, тем больше краски он получает. На следующем этапе бумага прижимается к барабану, в результате на ней строится отпечаток картин. Чтобы краска не осыпалась, на выходе нагретый валик вплавляет частицы тонера в бумагу. Поскольку лазерные принтеры используют достаточно сложные технологии, они стоят дороже матричных и струйных, и потребляют больше электроэнергии.



Светодиодные принтеры (их тоже часто называют лазерными) работают по такому же принципу, но изображение переносится на барабан не лазером, а светодиодной матрицей.

Сублимационный принтер печатает изображение совсем иначе: головка принтера нагревает поверхность, размягчая ее, а затем «впрыскивает» крохотные частицы красителя. Сверху наносится защитный слой, который предохраняет краску от разрушения солнечными лучами, и в итоге образуется очень стойкое изображение. Сублимационные принтеры прекрасно подходят для печати на пластиковых картах и компакт-дисках, часто используются для печати фотографий. Их недостатки – низкая скорость печати (более 1 минуты на одну фотографию) и высокая стоимость.

Важнейшей характеристикой принтера является его разрешающая способность.

Разрешающая способность принтера – это максимальное количество точек, которые он способен напечатать на единицу длины.

По традиции разрешающая способность измеряется в точках на дюйм (англ. *dpi* = *dots per inch*). Все современные струйные и лазерные принтеры имеют разрешающую способность не ниже 300 dpi, что обеспечивает высококачественную печать. Некоторые принтеры позволяют пользователю менять разрешающую способность, регулируя тем самым качество печати.

Обратим внимание на разницу обозначений *ppi* (пиксели на дюйм) и *dpi* (точки на дюйм). В *ppi* измеряется разрешение цифрового изображения (например, отсканированного), а в *dpi* – качество печати принтера. Каждый пиксель картинки может изображаться принтером в виде нескольких точек. Вспомните, что примерно то же самое происходит при выводе пикселей изображения на монитор (см. 5.7.1).

Принтеры также часто сравнивают по *скорости* печати (в количестве страниц в минуту). Наименьшая скорость печати у сублимационных и матричных принтеров, а наибольшая – у лазерных. Цветная печать, как правило, выполняется дольше, чем более простая чёрно-белая.

5.8. Устройства ввода/вывода

Некоторые компьютерные устройства нельзя однозначно отнести ни к устройствам ввода, ни к устройствам вывода. Пример такого «гибрида» – сенсорный экран. С одной стороны, на него выводится информация, а с другой – пользователь вводит команды, нажимая на нужный участок изображения. Сенсорные экраны применяют в портативных компьютерах, платёжных и информационных терминалах, а также для представления презентаций.

В некоторых мобильных телефонах, карманных и планшетных персональных компьютерах сенсорный экран заменил клавиатуру и занимает всю переднюю панель. Многие из этих устройств (например, смартфон *iPhone* и планшетный компьютер *iPad* фирмы *Apple*) используют технологию *мультитач* (англ. *multitouch*). Это значит, что сенсорный экран отслеживает нажатия и движения пальцев в нескольких точках одновременно. На-

пример, сближая пальцы рук, пользователь уменьшает масштаб изображения на дисплее, а раздвигая — увеличивает.



Контрольные вопросы

1. Зачем нужны устройства вывода?
2. В чем сходство и различие устройств ввода-вывода и внешней памяти?
3. Перечислите все известные вам устройства вывода. С какими из них вы работали?
4. Что такое МФУ?
5. Что такое АЦП и ЦАП?
6. Что такое монитор и каковы его возможности?
7. Что является элементом изображения в мониторе?
8. Отличается ли пиксель от точки экрана?
9. Компьютер выводит на экран монитора число, хранящееся в ячейке памяти. Какие действия он должен выполнить для этого? (Ответ: перевод двоичного числа в десятичное — получение его десятичных цифр и преобразование их в коды соответствующих символов).
10. Назовите наиболее важные характеристики мониторов?
11. Зачем нужна видеокарта? Каким образом она позволяет разгрузить центральный процессор?
12. Что такое видеопамять?
13. Какие типы принтеров вам известны? Опишите каждый из них.
14. Как работает лазерный принтер?
15. Что такое разрешающая способность принтера?
16. В чем отличие единиц *dpi* и *ppi*?
17. Как пересчитать сантиметры в дюймы?



Задачи

1. Рассмотрите схему индикаторной панели в содержимого регистров, приведенный в начале раздела 5.7. Прочитайте коды чисел, хранящиеся в регистрах *Pz1* и *Pz2*. Считая числа беззнаковыми целыми, сложите их, пользуясь правилами восьмеричной арифметики. Как будет выглядеть сумматор, когда в нем появится код суммы? (Ответ: 70201646₈)