

Глава 2

КОМПЬЮТЕР

2.1. Логические основы компьютера



2.1.1. Логические элементы и переключательные схемы

В параграфе 1.6.3 описан способ представления логических формул в виде схем, в которых используются графические изображения логических операций И, ИЛИ, НЕ.

Логическая схема отображает процесс вычисления логической формулы, заключающийся в определенной последовательности выполнения логических операций. Каждый отдельный блок логической схемы, обозначающий логическую операцию, называется **логическим элементом**. Этим элементам присвоены следующие названия:

- логический элемент **И** — **конъюнктор**: осуществляет конъюнкцию — логическое умножение;
- логический элемент **ИЛИ** — **дизъюнктор**: осуществляет дизъюнкцию — логическое сложение;
- логический элемент **НЕ** — **инвертор**: осуществляет инверсию — логическое отрицание.

Развитие электротехники в начале XX века привело к созданию автоматических переключательных устройств. Основу их конструкции составляют перечисленные выше логические элементы, функции которых выполняют простые электрические цепи. Наличие электрического тока в некотором участке цепи можно рассматривать как сигнал, обозначающий логическую единицу (истину). Отсутствие тока — логический ноль (ложь).

Функцию конъюнктора могут выполнять два последовательно соединенных переключателя в электрической цепи. Переключатель может принимать два состояния: «включено», «выключено». На рисунке 2.1 показана такая цепь с источником тока и электрической лампочкой в качестве индикатора тока. Ток по цепи



будет проходить только тогда, когда оба переключателя находятся в положении «включено», что соответствует двум логическим единицам на входе. Во всех других случаях тока не будет.

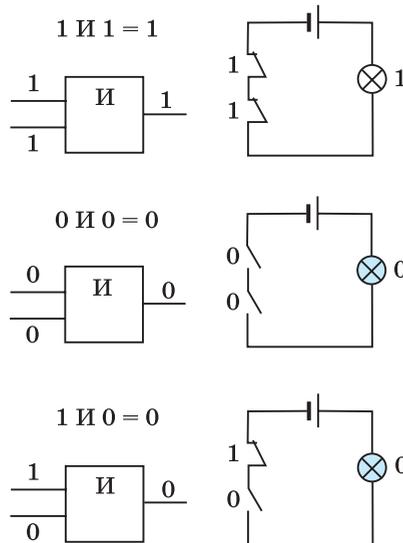


Рис. 2.1. Переключательная модель конъюнктора

Функцию дизъюнктора в электрической цепи выполняют два параллельных переключателя (рис. 2.2). Ток в цепи не будет только в том случае, если оба они выключены. Во всех других вариантах ток по цепи идет.

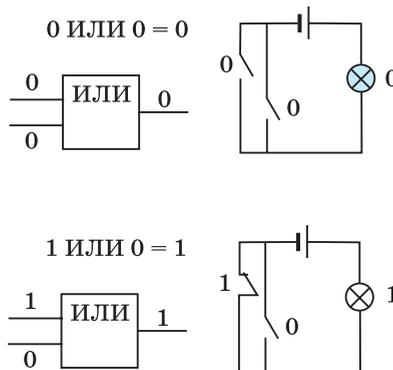


Рис. 2.2. Переключательная модель дизъюнктора

А как построить электрическую цепь, работающую по принципу инвертора? Идея простая: нужно переключатель поставить параллельно к участку цепи, на котором установлена лампочка (рис. 2.3). Тогда через лампочку пойдет ток, если переключатель разомкнут, и ток не пойдет, если переключатель замкнут (его сопротивление много меньше, чем сопротивление лампочки).

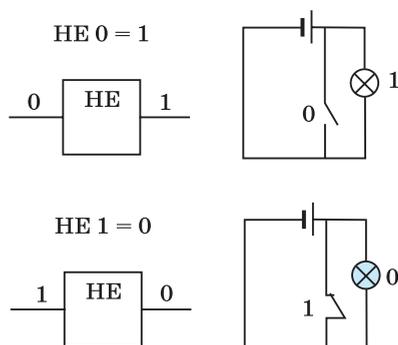
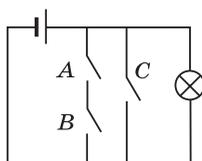


Рис. 2.3. Переключательная модель инвертора

Любая сложная логическая функция, представленная в нормальной форме (содержащая только операции И, ИЛИ, НЕ), может быть смоделирована с помощью электрической цепи, составленной из множества переключателей, соединенных по определенной схеме.

Пример. Нарисуем переключательную схему, которая будет моделировать следующую логическую формулу: $F = \neg(A \& B \vee C)$. Решением задачи является следующая схема:



Эволюция физической реализации переключателей

Рассмотренные переключательные (контактные) реализации логических элементов используются в автоматических устройствах. Одним из первых применений для таких устройств стали коммутаторы телефонных станций, где они использовались для соединения между собой различных абонентов телефонной сети. Ясно, что скорость работы такой системы существенно зависит от времени срабатывания переключателя.

Первые переключательные схемы создавались на основе электромагнитного (электромеханического) реле (рис. 2.4). К электромагниту при протекании тока через его обмотку притягивается контактный рычажок-переключатель, который замыкает электрическую цепь и переходит в состояние «включено». Когда ток не проходит через электромагнит, переключатель размыкает цепь.

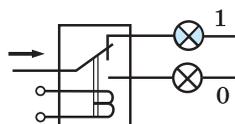


Рис. 2.4. Переключатель на основе электромеханического реле

Срабатывание электромагнитного реле происходит примерно за 0,1 секунды. Гораздо быстрее работает электронное реле, построенное на основе электронного прибора — триода. Первые триоды были ламповыми (рис. 2.5). Если на сетку триода подается отрицательный потенциал, то ток через триод не проходит. Если потенциал положительный, то триод пропускает электрический ток. Электронные переключательные системы работают в тысячи раз быстрее электромеханических.

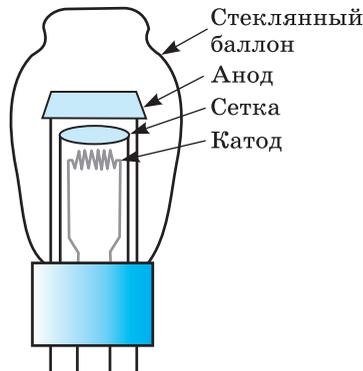


Рис. 2.5. Схема лампового триода

С изобретением полупроводниковых приборов на смену электронным лампам пришли транзисторы. Транзисторные схемы компактнее и надежнее в эксплуатации, чем ламповые. В 1970-х годах начинается выпуск микросхем. Современные технологии позволяют создавать микросхемы, помещающие в себе миллионы транзисторов.

Система основных понятий

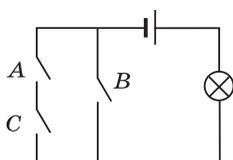


Логические элементы и переключательные схемы		
Конъюнктор	Дизъюнктор	Инвертор
Реализует логическую операцию И	Реализует логическую операцию ИЛИ	Реализует логическую операцию НЕ
Последовательно соединенные переключатели	Параллельно соединенные переключатели	Переключатель, параллельный участку цепи
<i>Эволюция физической реализации переключательных схем</i>		
На основе электромагнитных реле	На основе лампового триода	На основе полупроводниковых приборов: транзисторов, микросхем

Вопросы и задания



1. Перечислите названия логических элементов, используемых в логических схемах. Дайте объяснение этим названиям.
2. Зачем в схемах на рис. 2.1, 2.2, 2.3 нужны лампочки? Как вы думаете, являются ли лампочки необходимым элементом в релейных автоматах?
3. Напишите логическую формулу, которую реализует следующая переключательная схема:



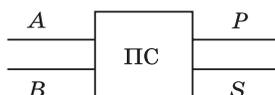
4. Нарисуйте переключательные схемы, реализующие следующие логические формулы: 1) $A \ \& \ B \ \& \ C \ \& \ D$; 2) $A \ \vee \ B \ \vee \ C$; 3) $A \ \& \ C \ \vee \ B$; 4) $\neg(A \ \& \ B)$; 5) $\neg(A \ \vee \ B)$.



2.1.2. Логические схемы элементов компьютера

Математический аппарат логики работает с двумя величинами: «истина» — логическая единица и «ложь» — логический ноль. В компьютере используется двоичное представление информации, а математические вычисления производятся в двоичной системе счисления. Эта аналогия приводит к идее поиска связи между двоичной арифметикой и алгеброй логики. Если вычисления в двоичной системе счисления можно свести к выполнению логических операций, а логические операции можно технически

Это схема **полусумматора** двоичных чисел. Полусумматор — это устройство с двумя логическими величинами на входе: A и B и двумя величинами на выходе: P и S . Без деталей внутреннего устройства будем обозначать полусумматор так:



Сумматор

При суммировании многозначных двоичных чисел на значение каждого разряда в сумме влияют не только значения этого же разряда в слагаемых, но и переносимая единица, которая может возникнуть при сложении цифр в предыдущем разряде. Такой механизм суммирования проиллюстрирован на следующем примере:

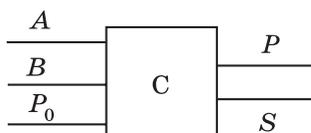
$$\begin{array}{r}
 \\
 \\
 + \\
 \hline
 1
 \end{array}$$

$1 \leftarrow$ $1 \leftarrow$ $1 \leftarrow$ $1 \leftarrow$ $1 \leftarrow$

Здесь в верхней строке единицы со стрелками обозначают переносимые единицы из результата сложения в предыдущем разряде.

Полусумматор производит сложение одноразрядных двоичных чисел, поэтому устройство полусумматора не учитывает переносимой единицы.

Одноразрядный сумматор устроен так, что он учитывает переносимую единицу. Поэтому на входе у него три величины, а на выходе — две. Схематически одноразрядный сумматор изобразим так:



Здесь A и B обозначают цифры в соответствующем разряде слагаемых, а P_0 — переносимая цифра, которая может быть нулем или единицей.

В таблице 2.2 представлены всевозможные варианты сложения одноразрядных двоичных чисел A , B и P_0 .

Таблица 2.2

Сложение трех одноразрядных двоичных чисел

<i>A</i>	<i>B</i>	<i>P₀</i>	<i>P</i>	<i>S</i>
0	0	0	0	0
0	1	0	0	1
1	0	0	0	1
1	1	0	1	0
0	0	1	0	1
0	1	1	1	0
1	0	1	1	0
1	1	1	1	1

Теперь задача состоит в том, чтобы найти две логические функции от трех логических аргументов: $P(A, B, P_0)$ и $S(A, B, P_0)$, значения которых совпадали бы со значениями в таблице. Вот как выглядят эти функции:

$$P = (A \& B) \vee (A \& P_0) \vee (B \& P_0); \quad (3)$$

$$S = (A \vee B \vee P_0) \& \neg P \vee (A \& B \& P_0). \quad (4)$$

Логическую схему, реализующую на выходе эти две функции, можно изобразить двумя способами. Первый способ — составить ее из логических элементов в соответствии с данными формулами. На входе такой схемы три величины A , B и P_0 , на выходе — P и S . Вторым способом — составить схему сумматора как комбинацию двух полусумматоров и одного дизъюнктора. Приведем второй вариант логической схемы одноразрядного сумматора (рис. 2.7).

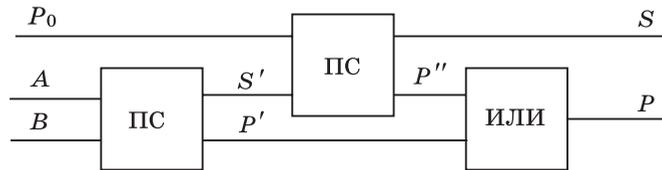


Рис. 2.7. Логическая схема одноразрядного сумматора

Процессор компьютера производит вычисления с многоразрядными двоичными числами. Следовательно, для выполнения сложения в процессоре должен использоваться **многоразрядный сумматор**. Его конструкция представляет собой цепочку одноразрядных сумматоров. Между ними должна быть организована связь по переносу единицы из одного разряда в другой. Например, в 32-разрядном процессоре такая цепочка содержит 32 сумматора. Схематически цепочка из 32 сумматоров изображена на рис. 2.8.

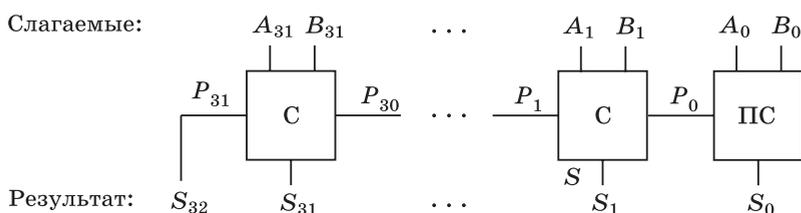


Рис. 2.8. Схема работы 32-разрядного сумматора

Слагаемые числа заносятся в специальные ячейки процессора, которые называются **регистрами**. Двоичные разряды регистра нумеруются справа налево, начиная с нуля. Поэтому схему на рис. 2.8 надо просматривать справа налево. A_0 обозначает нулевой двоичный разряд первого слагаемого, B_0 — нулевой двоичный разряд второго слагаемого, S_0 — нулевой разряд суммы. Соответственно A_1, B_1, S_1 — первые разряды слагаемых и суммы и т. д. Для сложения нулевых разрядов используется полусумматор, так как для него нет переносимой единицы. Далее следует цепочка одноразрядных сумматоров. Переносимый разряд P с предыдущего сумматора передается на вход следующего сумматора.

Последний двоичный разряд в регистре имеет номер 31; S_{31} — старшая цифра в слагаемом. Разряд переноса P_{31} оказывается в то же время 32-м разрядом суммы (результата) — S_{32} . Но поскольку такого разряда в регистре не существует, его значение не сохраняется. Если $S_{32} = 0$, то это роли не играет. Если же $S_{32} = 1$, то единица исчезает. Такая ситуация называется *целочисленным переполнением*. При этом процессор работы не прерывает. К этому вопросу мы вернемся немного позже, когда будем рассматривать представление целых чисел в памяти компьютера и особенности целочисленной арифметики в компьютере.

Многоразрядный сумматор является базовым элементом арифметико-логического устройства — вычислительного блока процессора. С его помощью выполняется не только сложение, но и вычитание чисел, которое сводится к сложению (подробнее об этом будет рассказано в параграфе 2.4.1). А поскольку умножение сводится к многократному сложению, а деление — к многократному вычитанию, эти операции тоже реализуются с помощью сумматора.

Триггер

Центральными устройствами компьютера являются процессор и внутренняя память. Сумматор — это устройство в составе процессора. С его помощью компьютер выполняет вычисления. Во *внутренней памяти* хранятся исполняемая программа и оперативные данные. В составе процессора есть специальные ячейки памяти, которые называются **регистрами**.

Основным элементом памяти (внутренней и регистровой) является один двоичный разряд — бит памяти. Бит памяти может принимать два устойчивых состояния и, следовательно, несет 1 бит информации. Техническая реализация бита памяти должна обеспечивать его устойчивую работу и высокую скорость переключения из одного состояния в другое, т. е. переход от нуля к единице и обратно.

Элемент электронной схемы компьютера, реализующий бит регистровой памяти и статической внутренней памяти, называется триггером¹⁾. Триггер на основе лампового триода был изобретен в 1918 г. российским ученым-электротехником М. А. Бонч-Бруевичем. Впоследствии триггеры стали создавать на базе полупроводниковых транзисторов и микросхем.

На рисунке 2.9 приведена схема входов и выходов одного из самых распространенных типов триггера — *RS*-триггера. Она содержит два входа и один выход. На входы поочередно подаются *импульсные электрические сигналы*. С выхода снимается сигнал: 1 или 0. Триггер должен работать так: если на выходе 0, то при подаче на вход *S* сигнала (единицы), на выходе *Q* должна установиться 1. После прекращения подачи сигнала на вход *S* на выходе единица должна сохраняться.



Рис. 2.9. Входы и выходы *RS*-триггера

Снятие единицы на выходе происходит только в результате подачи сигнала на вход *R*. После этого на выходе *Q* устанавливается 0 и сохраняется до тех пор, пока на *S* не будет подан сигнал, т. е. 1. Подача сигналов на входы *S* и *R* может производиться только по очереди. Одновременно сигналы на *S* и на *R* поданы быть не могут.

Логическая схема *RS*-триггера показана на рис. 2.10.

RS-триггер состоит из двух дизъюнкторов и двух инверторов. Кроме входов для управляющих сигналов *R* и *S* в устройстве триггера реализована еще *обратная связь*. Обратной связью называют подачу выходного сигнала на вход устройства. Из схемы видно, что обратная связь имеет *перекрестный* характер. Основным выходом является *Q*. Выход *Q* является дополнительным, используемым только для обратной связи. Логическое значение на выходе \bar{Q} всегда противоположно значению на выходе *Q*.

¹⁾ О типах внутренней памяти компьютера подробно рассказано в параграфе 2.5.4.

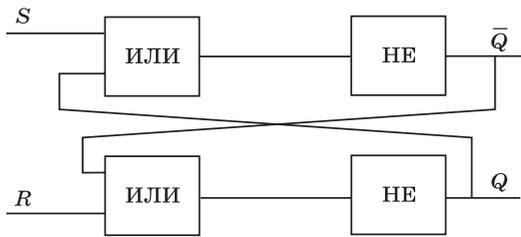


Рис. 2.10. Логическая схема RS-триггера

Устойчивым начальным состоянием триггера является: $S = 0$, $R = 0$, $Q = 0$, $\bar{Q} = 1$. После подачи сигнала на вход S установится устойчивое состояние: $S = 1$, $R = 0$, $Q = 1$, $\bar{Q} = 0$. После снятия сигнала с S (сигнал подается кратковременно) состояние триггера будет следующим: $S = 0$, $R = 0$, $Q = 1$, $\bar{Q} = 0$, т. е. на основном выходе останется 1.

Чтобы сбросить единицу на выходе, подается сигнал на вход R . Установится следующее состояние: $S = 0$, $R = 1$, $Q = 0$, $\bar{Q} = 1$. После прекращения действия этого сигнала ноль на выходе Q сохранится: $S = 0$, $R = 0$, $Q = 0$, $\bar{Q} = 1$. Вернулись в первоначальное состояние! В процессе работы компьютера такие переключения происходят многократно.

Система основных понятий



Логические схемы элементов компьютера		
Полусумматор	Сумматор	Триггер
Суммирует два одноразрядных двоичных числа без учета единицы переноса разряда	Суммирует два одноразрядных двоичных числа с учетом единицы переноса разряда	Реализует бит регистровой и статической внутренней памяти
<p>A и B — слагаемые двоичные цифры; S — младшая цифра суммы; P — старшая цифра суммы</p>	<p>A и B — слагаемые: двоичные цифры; P_0 — входной разряд переноса; S — младшая цифра суммы; P — разряд переноса</p>	<p>S — вход для включения; R — вход для выключения; Q — выходной бит</p>
N-разрядный сумматор		
Цепочка из одного полусумматора и $(N-1)$ сумматоров с передачей разряда переноса. Устройство в составе процессора		



Вопросы и задания

1. Логическими элементами какого устройства компьютера являются сумматор и полусумматор?
2. В чем разница между сумматором и полусумматором?
3. В среде электронной таблицы воспроизведите таблицу 2.1 для полусумматора, используя логические формулы (1), (2).
4. В среде электронной таблицы воспроизведите таблицу 2.2 для одноразрядного сумматора, используя логические формулы (3), (4).
5. Смоделируйте в электронной таблице работу восьмиразрядного сумматора.
6. В чем состоит назначение триггера? В каких устройствах компьютера он используется?
7. Объясните, почему на входы S и R нельзя подавать сигналы одновременно.

Практикум. Раздел 6 «Устройство компьютера»



2.2. Эволюция устройства вычислительной машины

Аналитическая машина Бэббиджа



Чарльз Бэббидж
(1791–1871)

Мечта создать машину, которая сама могла бы выполнять математические вычисления без участия в этом процессе человека, давно волновала умы ученых. Впервые попытался реализовать эту мечту английский ученый-изобретатель Чарльз Бэббидж. Свое изобретение Бэббидж назвал Аналитической машиной. Демонстрация Аналитической машины состоялась в Лондоне в 1862 году.

Аналитическая машина Бэббиджа — это механическое устройство, умеющее выполнять все арифметические операции. В отличие от ранее изобретенного арифмометра (Б. Паскаль, Г. Лейбниц, В. Однер), управлением работой которого человек занимался вручную, Аналитическая машина работает автоматически под управлением программы, заранее составленной человеком. Именно с проекта Бэббиджа берет свое начало профессия программиста. А первым в истории программистом стала коллега Бэббиджа — Ада Лавлейс — дочь английского поэта Джорджа Байрона.



Середина XIX века — это эпоха больших достижений в области механики. Ни электротехники, ни тем более электроники в ту пору еще не было. Поэтому неудивительно, что вычислительная машина Бэббиджа была механической. Из-за большой сложности, а потому низкой надежности механизма проект Аналитической машины не был доведен до конца. Тем не менее бесспорна заслуга Бэббиджа в истории вычислительной техники, в продвижении науки к созданию автоматических вычислительных машин.

Бэббидж выдвинул ряд идей в конструкции вычислительной машины, которые нашли продолжение в будущих изобретениях. Он предложил **состав основных устройств**: *устройство памяти*; *арифметическое устройство*, осуществляющее вычисления; *устройство управления* работой машины; *устройства для ввода* исходных данных и программы; *устройство для вывода* результатов. Правда, называл их Бэббидж иначе. Он ассоциировал работу своей машины с работой предприятия, имеющего на входе сырье, а на выходе — готовые изделия. Поэтому память он называл *складом*, арифметическое устройство — *фабрикой*, устройство управления — *конторой*.

Для хранения программ использовались *перфокарты* — металлические пластины с отверстиями, расположение которых заключало в себе информацию о выполняемых машиной вычислительных операциях. Последовательность карт составляла *программу* вычислений. Еще в 1804 году перфокарты были изобретены французом Ж. М. Жаккардом и применялись в ткацких станках для управления их работой. В связи с этим будет справедливо отметить, что ткацкий станок был первым в истории техники программно управляемым устройством.

В составе Аналитической машины имелся блок, посредством которого машина последовательно «просматривала» программу на перфокартах и выполняла вычислительные операции в заданном порядке. Машина производила вычисления с 50-разрядными десятичными числами. Сложение и вычитание выполнялись за 1 секунду, умножение и деление — за 1 минуту. В системе команд Аналитической машины была команда условного перехода, позволяющая реализовать нелинейные алгоритмы.

Таким образом, наиболее значимые для будущего идеи Бэббиджа — *определение состава устройств вычислительной машины и реализация принципа программного управления*.

Релейные вычислительные машины

Следующим шагом в истории развития вычислительной техники стало создание машин на основе *электромеханических реле*. Вначале это были машины специализированного назначения,

т. е. каждая такая машина выполняла только один определенный вид обработки данных. Например, при обработке больших массивов числовых данных требовалось отсортировать эти данные в определенном порядке (задача сортировки), распечатать числовые таблицы на бумаге (задача табулирования); вычислить некоторые итоговые характеристики чисел: сумму, среднее значение и пр. (статистическая обработка).

Согласно идее Жаккара–Бэббиджа, исходные данные представлялись на перфокартах, на которые они наносились с помощью специального устройства — перфоратора. Сами же вычислительные машины назывались *счетно-перфорационными*. Первой в истории фирмой, занявшейся производством счетно-перфорационных машин, была американская компания IBM. Ее основал в 1897 году американский изобретатель Герман Холлерит.

На базе счетно-перфорационных машин уже в конце XIX века во многих странах появились *машиносчетные станции* для механизированной обработки информации, послужившие прообразом вычислительных центров (ВЦ), возникших в середине XX столетия. На таких станциях выполнялся широкий круг экономических и научно-технических расчетов. Большое распространение машиносчетные станции получают в 20–30-х годах прошлого века.

В 1940-х годах создаются *релейные вычислительные машины с программным управлением, обладающие алгоритмической универсальностью*. Это уже были прообразы будущих ЭВМ, на которых можно было выполнять сложные научно-технические расчеты. Скорость расчетов на таких машинах превышала на порядок скорость вычислений с помощью арифмометров с электроприводом.

Наиболее крупными проектами данного периода были машины Z1, Z2 и Z3, созданные немецким изобретателем Конрадом Цузе. В США созданием релейных машин занимается Говард Эйкен (машины MARK-1, 1944 г.; MARK-2, 1947 г.). Машины Цузе и Эйкена производили вычисления в двоичной системе счисления. Программы управления работой машин задавались с помощью *перфорированной ленты*. На машине MARK-1 операция сложения выполнялась за 0,3 секунды, умножение — за 5 секунд и деление — за 15 секунд.

Последней крупной разработкой релейной вычислительной техники стала машина РВМ-1, построенная в 1957 году в Советском Союзе. Ее конструктор — Николай Иванович Бессонов. Машина проработала до конца 1964 года и в основном использовалась для решения экономических задач.

Первая электронная вычислительная машина

Хотя электронный триггер был изобретен еще в 1918 году М. А. Бонч-Бруевичем, однако его применение в вычислительных машинах долгое время оставалось нереальным по причине слишком высокой стоимости и громоздкости таких машин. Поэтому с 20-х по 40-е годы XX века основой элементной базы оставались электромеханические реле.

В 1940-х годах стало понятно, что двигаться по пути развития релейных машин дальше было некуда. Инерционность механического реле сильно ограничивала скорость вычислений. Прогресс был возможен только на пути применения *электронного реле — триггера*, в устройстве которого используются электронные лампы. Развитие радиотехнической промышленности сделало более доступным использование *электронных ламп*.

Поиски конструкции электронной вычислительной машины были начаты в 1937 году американским ученым Джоном Атанасовым. Работы Атанасова носили в большей степени исследовательский характер. Их результаты оказали сильное влияние на создателей *первой электронной вычислительной машины (ЭВМ)* — Джона Уильяма Мочли и Джона Преспера Эккерта. В 1945 году они построили ЭВМ, получившую название *ENIAC* (Electronic Numerical Integrator and Computer) — рис. 2.11. В основе ее конструкции использовались ламповые триггеры. Публичная демонстрация работы машины ENIAC состоялась 15 февраля 1946 года. Эта дата считается началом эры ЭВМ.

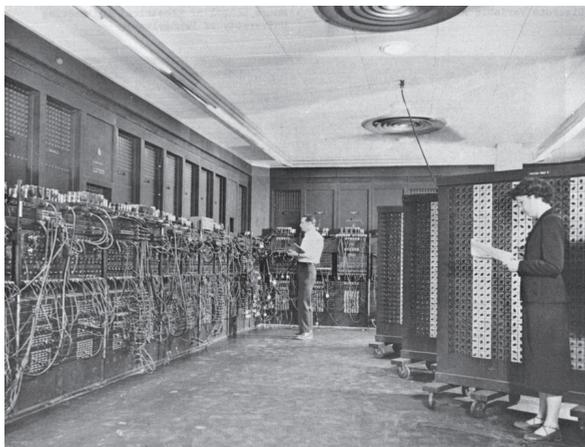


Рис. 2.11. ЭВМ ENIAC

Машина содержала в себе 18 тысяч электронных ламп. Ее масса составляла 30 тонн. Быстродействие было примерно в 1000 раз выше, чем у релейных машин. Операция сложения выполнялась за 0,2 миллисекунды (мс), умножения — за 2,8 мс.

В памяти машины хранились только числа. Программа задавалась путем коммутации (т. е. контактного соединения) триггеров на 40 наборных полях. На перепрограммирование машины требовались недели. Надежность машины была очень низкой. Поиск неисправностей мог занимать многие часы.

По своей структуре ЭВМ ENIAC напоминала механические вычислительные машины. Регистры памяти состояли из триггерных колец: на каждом кольце по 10 триггеров, как зубцов на шестерне механической машины. Нужная десятичная цифра запоминалась путем включения одного из триггеров кольца.

Базовые принципы устройства ЭВМ

Хотя первая ЭВМ ENIAC открыла эру электронных вычислительных машин, однако назвать ее прообразом будущих компьютеров нельзя.



Джон фон Нейман
(1903–1957)

В 1946 году публикуется статья американских авторов А. Беркса, Г. Гольдстейна, Джона фон Неймана «Предварительное рассмотрение логической конструкции электронного вычислительного устройства». В этой работе обобщался опыт всех предыдущих поисков конструкции автоматической вычислительной машины и предлагались новые идеи для дальнейшего развития в этой области. Изложенные в работе принципы устройства ЭВМ принято называть **принципами фон Неймана**. Опишем основные принципы.

Состав основных устройств ЭВМ. Универсальная вычислительная машина должна включать в себя устройство памяти, арифметическое устройство для выполнения вычислений, устройство управления и устройства для связи с оператором ЭВМ (устройства ввода/вывода).

Принцип двоичного кодирования. Данные и программы представляются в памяти машины в форме двоичного кода. Наименьшей частицей памяти является один двоичный разряд — бит памяти. Память дискретна.

Принцип хранимой программы. Во время выполнения расчетов числа и программа вычислений находятся в общей памяти

машины. Между ними нет разницы в том смысле, что и то, и другое — двоичные коды.

Принцип адресуемости памяти. Отдельные двоичные разряды памяти — биты объединяются в ячейки, каждая из которых имеет свой адрес. Адрес ячейки — это ее порядковый номер в памяти. Запись в память данных или чтение из памяти происходит по адресам. В первых компьютерах ячейка памяти хранила либо число, либо команду программы. Поэтому ячейки были большими — содержали десятки битов. Начиная с третьего поколения ЭВМ, в оперативной памяти стали адресоваться 8-разрядные ячейки — байты памяти. Для хранения данных или команд большей длины используются ячейки, состоящие из нескольких последовательных байтов (двух, четырех и более), которые называются машинными словами. Адрес машинного слова равен адресу младшего байта (имеющего наименьший номер).

Принцип программного управления. Работа ЭВМ происходит автоматически под управлением программы, помещенной в оперативную память. Программа — это последовательность команд. Команда содержит следующую информацию: какая операция должна выполняться, откуда (из каких ячеек памяти) извлекаются операнды, куда помещается результат операции. В некоторых командах указывается адрес следующей исполняемой команды. Команды выполняются по одной, друг за другом. Исполнение программы завершается по команде «останов». Управление последовательностью операций осуществляет устройство управления (УУ). Арифметические и логические операции выполняет арифметико-логическое устройство (АЛУ). УУ и АЛУ входят в состав процессора. На время выполнения команды она сама, а также операнды записываются в специальные ячейки памяти процессора — регистры. Адрес очередной исполняемой команды находится в счетчике команд — специальном регистре процессора.

Семейства ЭВМ и архитектура

Принципы фон Неймана оказались настолько фундаментальными, что уже более 60 лет они остаются основой для устройства электронных вычислительных машин. Первой в истории вычислительной машиной, созданной в соответствии с принципами фон Неймана, была английская ЭВМ EDSAC, построенная в 1949 году. Серийное производство ЭВМ началось в 1950-х годах.

Первой отечественной ЭВМ была МЭСМ (Малая Электронная Счетная Машина) конструкции Сергея Алексеевича Лебедева. Впоследствии под руководством Лебедева были созданы и



Сергей Алексеевич
Лебедев
(1902–1974)

серийно выпускались ламповая ЭВМ М-20, ее полупроводниковый аналог М-220, машины БЭСМ-3М, БЭСМ-4. Первой отечественной ЭВМ с быстродействием 1 миллион операций в секунду была БЭСМ-6, производство которой началось в 1967 году. Позже Лебедев руководил разработкой отечественного многопроцессорного вычислительного комплекса «Эльбрус».

С развитием серийного производства электронных вычислительных машин формируется понятие об **архитектуре ЭВМ** как о *совокупности базовых принципов устройства и функционирования, объединяющих семейство машин.*

Семейство ЭВМ — это множество различных моделей программно совместимых машин, т. е. машин, для которых возможна переносимость программ с одной модели на другую. Обычно здесь делают оговорку о совместимости «снизу вверх», т. е. программы, работающие на более простых моделях, обязательно должны работать на более сложных моделях. Обратный перенос не всегда возможен.

Для программной совместимости главными принципами является общность системы команд (одинаковые языки процессоров) и одинаковые форматы представления данных. Первыми отечественными семействами ЭВМ были: ламповые и полупроводниковые машины конструкции С. А. Лебедева (М-20, М-220, М-222, БЭСМ-3, БЭСМ-4); серия машин «Урал» (Урал-2, Урал-4, Урал-16); серия машин «Минск» (Минск-12, Минск-14, Минск-32). Первым семейством ЭВМ на интегральных схемах стала серия машин IBM-360, выпуск которой начался в США в 1964 году.

Примерно в то же время начинается выпуск серии ЭВМ PDP американской фирмы DEC. Это малые ЭВМ, способные работать в *режиме реального времени* и, благодаря этому, управлять работой различных технических устройств. Поэтому их еще называют управляющими машинами.

Архитектуру машин серии PDP называют архитектурой «общей шины». Позже такая архитектура стала использоваться в микроЭВМ и персональных компьютерах (ПК). Некоторые сведения об устройстве ПК вам известны из курса информатики 7–9 классов. Более подробные знания вы получите в следующих параграфах.

Система основных понятий



Эволюция устройства вычислительной машины	
Аналитическая машина Бэббиджа	Основа технического устройства: механическая передача
	Основные идеи: состав устройств (память, процессор, устройства ввода/вывода); принцип программного управления
Релейные вычислительные машины	Элементная база: электромеханические реле
	Специализированные счетно-перфорационные машины (Г. Холлерит)
	Универсальные вычислительные машины с программным управлением: Z1, Z2, Z3 (К. Цузе, Германия), MARK-1, MARK-2 (Г. Эйкен, США), PBM-1 (Н. И. Бессонов, СССР)
Первая ЭВМ ENIAC	Машина на электронных триггерных схемах. Д. У. Мочли, Д. П. Эккерт
	В памяти — числа; коммутируемая программа
Принципы Джона фон Неймана	<i>Базовые принципы устройства ЭВМ:</i> состав основных устройств ЭВМ; принцип двоичного кодирования; принцип хранимой программы; принцип адресности памяти; принцип программного управления
Семейства ЭВМ и архитектура	Архитектура: базовые принципы устройства и функционирования, объединяющие семейство ЭВМ
	Семейство ЭВМ: множество программно совместимых моделей компьютеров

Вопросы и задания



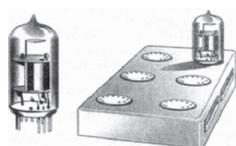
1. В чем состоит вклад Ч. Бэббиджа в историю вычислительной техники?
2. Для чего использовались счетно-перфорационные машины?
3. В чем главное отличие релейных вычислительных машин Z, MARK, PBM от счетно-перфорационных?
4. Как расшифровывается название первой ЭВМ?
5. Сформулируйте основные принципы устройства ЭВМ Джона фон Неймана. Какие из этих принципов не выполнялись в ЭВМ ENIAC?
6. Что такое семейство ЭВМ, архитектура ЭВМ?
7. Какие современные семейства ЭВМ вы знаете? Подготовьте презентацию.



2.3. Смена поколений ЭВМ

Серийное производство ЭВМ начинается в разных странах в 1950-х годах. Историю развития ЭВМ принято делить на поколения. Переход от одного поколения к другому связан со сменой элементной базы, на которой создавались машины, с изменением архитектуры ЭВМ, с развитием основных технических характеристик (скорости вычислений, объема памяти и др.), с изменением областей применения и способов эксплуатации машин.

В основе архитектуры ЭВМ разных поколений лежат принципы фон Неймана. Однако в процессе развития происходят некоторые отклонения от неймановской архитектуры.



К ЭВМ первого поколения относят **ламповые машины**, в которых основными элементами устройства были электронно-вакуумные лампы. Выпускались эти машины в 1950-х годах.

Архитектура машин первого поколения наиболее точно соответствовала основным принципам фон Неймана. В этих машинах один процессор управлял работой всех устройств: внутренней и внешней памяти, устройств ввода и вывода, как показано на рис. 2.12.

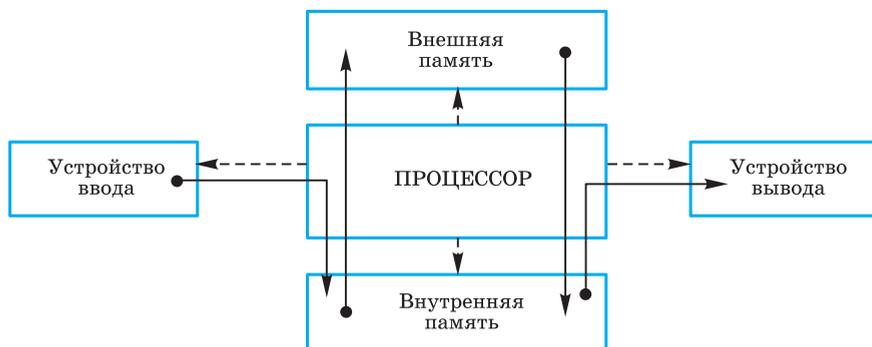


Рис. 2.12. Структура однопроцессорной ЭВМ. Сплошные стрелки — передача данных, пунктирные стрелки — управляющее воздействие

Примерами наиболее совершенных отечественных машин первого поколения являются: М-20, Урал-4, Минск-14. Скорость счета (быстродействие) этих машин находилась в пределах от 2000 оп./с (операций в секунду) (Минск-14) до 20 000 оп./с (М-20). Объем оперативной памяти (ОЗУ) составлял несколько тысяч ячеек. Например, в ЭВМ М-20 ОЗУ содержало 4096 45-разрядных ячеек. Приблизительно это равно 22 килобайтам.

В качестве устройств внешней памяти использовались накопители на *магнитной ленте (НМЛ)* и накопители на *магнитных барабанах (НМБ)*. Объем информации, помещавшейся на НМЛ, был примерно равен 1,5 Мб, а на НМБ — 60 Кб.

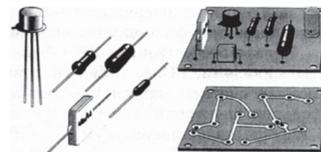
Ввод данных и программ осуществлялся с помощью перфорационных носителей: перфокарт или перфолист. Вывод производился с помощью печатающего устройства на бумагу.

На ЭВМ первого поколения программисты работали, главным образом, на *языках машинных команд*. Хотя уже в этот период появляются и развиваются языки программирования высокого уровня, создаются компиляторы: программы перевода с языков высокого уровня на язык машинных команд. Термин «компилятор» впервые ввела Грейс Хоппер в 1951 году. В то время она была капитаном ВМФ США (в дальнейшем — единственная женщина-адмирал ВМФ). Хоппер разработала компилятор с языка, упрощающего программирование математических вычислений. Эти работы были продолжены Джоном Бэкусом, который в 1957 году разработал *первый язык программирования высокого уровня ФОРТРАН*. Название расшифровывается как «транслятор формул». Язык был реализован впервые на американской ЭВМ IBM 704. Большое распространение в дальнейшем получила версия языка Фортран-60. Язык Фортран с того времени и до наших дней остается одним из основных языков для научных расчетов.

Ламповые ЭВМ были громоздкими, потребляли большое количество электроэнергии (сотни киловатт), часто требовали ремонта из-за выхода из строя деталей. Стоимость таких машин была очень высокой, обслуживающий персонал составлял десятки, а иногда и сотни человек.

Использование ЭВМ происходило в режиме прямого взаимодействия программиста или оператора ЭВМ с машиной. Через устройство ввода с перфоносителя (перфолист или перфокарт) вводилась программа и исходные данные. Затем шло исполнение программы, при этом программист (оператор) мог вмешиваться в ход ее выполнения с пульта управления машиной. Работа заканчивалась выводом результатов на печать. После этого начиналась работа со следующей программой. При таком режиме работы КПД использования ЭВМ был крайне низок.

ЭВМ второго поколения — это машины на базе **полупроводниковых элементов — транзисторов**. Транзистор — это полупроводниковый триод. В отличие от электронной лампы, которая требует некоторого времени для разогрева и по-



этому не сразу входит в рабочий режим, транзистор включается в работу мгновенно. Кроме того, транзисторы много реже выходят из строя, компактнее и дешевле электронных ламп. Все эти преимущества обусловили переход в 1960-х годах к производству машин второго поколения.

Некоторые модели машин второго поколения повторяли по своей архитектуре ЭВМ первого поколения и продолжали вместе с ними семейство программно совместимых машин. Так, «транзисторным продолжением» отечественной машины М-20 стали М-220, М-222, БЭСМ-4. Их быстродействие было примерно таким же, как у М-20: 20–27 тысяч оп./с. Зато оперативная память в 4–8 раз больше.

В этот период появляются ЭВМ с новыми элементами в архитектуре. Например, в архитектуре американской машины CDC-6600, помимо центрального процессора, присутствовали периферийные процессоры, которые назывались каналами ввода/вывода (рис. 2.13). Их задача состояла в автономном управлении устройствами ввода/вывода и внешней памяти, что освобождало от этой работы центральный процессор. В результате КПД использования центрального процессора существенно возрос. Быстродействие CDC-6600 составляло 3 млн оп./с.

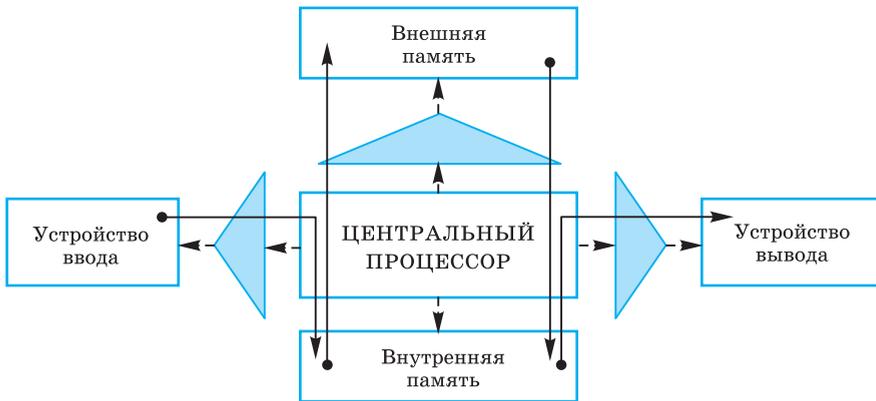


Рис. 2.13. Структура ЭВМ с одним центральным процессором и периферийными процессорами управления внешними устройствами (треугольники)

Значительным достижением отечественного электронного машиностроения стала ЭВМ БЭСМ-6 конструкции С. А. Лебедева и В. А. Мельникова (рис. 2.14). Эта машина имела быстродействие 1 млн оп./с. Оперативная память составляла от 192 до 768 Кб. В ее конструкции использовались 60 тысяч транзисторов и 200 тысяч полупроводниковых диодов. В архитектуре БЭСМ-6



Рис. 2.14. ЭВМ БЭСМ-6

были реализованы новые идеи, в том числе периферийные процессоры для управления внешними устройствами. Надежность работы машины была очень высокой. Выпуск БЭСМ-6 начался в 1968 году. Эксплуатация этих машин продолжалась около 20 лет.

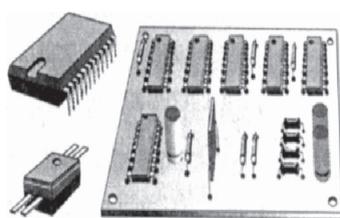
На машинах второго поколения появляются *первые операционные системы*. Изменился режим использования ЭВМ. Вместо режима прямого доступа внедряется *режим пакетной обработки заданий*. Суть его в следующем. Несколько программ (пакет) записываются на магнитную ленту. К программам прилагаются управляющие команды. Задание — это программа плюс управляющие команды, определяющие последовательность действий, выполняемых над программой: ввести программу, вызвать и запустить транслятор, ввести данные, запустить программу на исполнение и т. п.

Задания вводились в компьютер с магнитной ленты. После окончания выполнения одного задания сразу же начинало выполняться следующее задание. ЭВМ практически не простаивала. Результаты записывались на магнитную ленту. Работа заканчивалась выводом результатов на печать. Управление потоком заданий осуществляла системная программа, которая называлась *монитором (диспетчером)* пакетной обработки заданий.

В период второго поколения возникает и расширяется производство малых ЭВМ. Это более простые и дешевые машины по сравнению с большими ЭВМ. Среди отечественных малых ЭВМ второго поколения были машины: Проминь, Мир, Наири. У этих машин были свои оригинальные конструктивные решения. Например, на машине «Мир» использовался дисплей со световым пером, а также была реализована система аналитических вычислений под названием «язык Аналитик».

Большие результаты в этот период были достигнуты в области *автоматизации программирования*. В основном работа была направлена на *создание новых языков программирования высокого уровня и разработку трансляторов к ним, а также систем отладки программ и библиотек стандартных программ*. В европейских странах развивается и распространяется язык АЛГОЛ, в США — ФОРТРАН и КОБОЛ (специализированный язык для решения экономических задач, созданный Г. Хоппер). В 1964 году в США разрабатывается язык Бейсик, предназначенный для начинающих программистов, который позже нашел широкое применение на микроЭВМ и персональных компьютерах. Появляются новые непроцедурные языки, специализированные языки для определенных классов задач: ЛИСП — язык функционального программирования, СНОБОЛ — язык для обработки строк, СИМУЛА-1 — язык моделирования, АПЛ и др.

«Настоящая история электронной вычислительной техники и ее революционизирующего влияния на общество начинается с конца 1960-х годов, когда появились ЭВМ так называемого третьего поколения». (Академик Н. Н. Моисеев.)



Элементной базой ЭВМ **третьего поколения** стали **интегральные схемы (ИС)** — миниатюрные электронные приборы, объединявшие в себе сначала сотни, а затем тысячи элементов электронной схемы. Промышленное производство интегральных схем началось в 1962 году. В Америке их стали

называть *чипами*. Пионером в производстве чипов была фирма Intel (INTEgrated ELectronics — «интегральная электроника»).

В 1964 году компания IBM объявила о создании шести моделей ЭВМ нового поколения на интегральных схемах. Серия этих машин получила название IBM/360. Это были программно совместимые машины с общей архитектурой, но разной производительности, комплектации и стоимости. Быстродействие разных моделей машин находилось в диапазоне от нескольких десятков тысяч до нескольких миллионов операций в секунду.

В 1970 году начинается выпуск серии машин IBM/370. Они обладали программной совместимостью с IBM/360, но были более совершенными. Элементной базой для них стали большие интегральные схемы — БИС.

ЭВМ серии IBM/360/370 содержали в своей конструкции один центральный процессор и несколько периферийных процессоров для управления внешними устройствами, которые назывались каналами ввода/вывода. Это позволило реализовать *мультипро-*

граммный режим работы: пока одна программа занята вводом/выводом данных, другая программа занимает центральный процессор, выполняя вычисления.

Операционные системы ЭВМ третьего поколения могли реализовать функцию разделения ресурсов ЭВМ между несколькими выполняемыми программами. К разделяемым ресурсам, прежде всего, относятся время работы процессора и оперативная память. Задача ОС состоит в том, чтобы разные программы, выполняемые одновременно на ЭВМ, «не мешали» друг другу и чтобы КПД работы центрального процессора был максимальным, иначе говоря, чтобы центральный процессор не простаивал. ОС берет на себя также заботу о порядке использования несколькими программами общих внешних устройств: внешней памяти, устройств ввода/вывода.

На машинах третьего поколения стали широко использоваться *накопители на магнитных дисках (НМД)* — устройствах внешней памяти *прямого доступа* (в отличие от НМЛ — устройствах *последовательного доступа*). Задача операционной системы заключалась также в *управлении работой с файлами на магнитных дисках*.

Третья задача ОС — *обеспечение взаимодействия пользователя* (оператора ЭВМ, программиста) *с компьютером*. Это взаимодействие осуществляется на специальном командном языке операционной системы. Техническим средством такого взаимодействия (интерфейса) сначала были консоли — электрические пишущие машинки, а позже — мониторы с электронно-лучевым дисплеем.

Наиболее развитыми возможностями обладала операционная система OS/360. На ее основе был реализован многопользовательский режим работы ЭВМ. Несколько пользователей одновременно на одной машине могли исполнять свои программы, вводя их через *терминал — клавишное устройство с дисплеем*. Вернулся режим прямого взаимодействия пользователя (программиста) с компьютером, как во времена первого поколения ЭВМ. Но теперь появилась *возможность параллельной работы многих пользователей*. Помещения с терминалами — *терминальные залы* — располагались отдельно от машинного зала.

Основным средством работы на ЭВМ остаются языки программирования высокого уровня. Большое распространение получает язык PL-1 (в переводе — «язык программирования номер один»). Это первый универсальный язык, предназначенный для работы с любыми типами данных. Разрабатываются многочисленные *пакеты прикладных программ (ППП)* для различных областей производства, экономики, науки. ППП предназначены для их использования непрограммирующими пользователями через терминальную систему связи с ЭВМ.

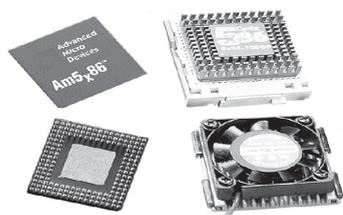
Архитектура ЭВМ серии IBM/360/370 фактически стала международным стандартом для больших машин третьего поколения. В ряде других стран в те же годы начинается выпуск серий машин, совместимых с IBM/360/370. Такие машины выпускались на предприятиях Великобритании, ФРГ, Японии, Голландии. В Советском Союзе совместно с европейскими социалистическими странами в 1970–1980-х годах выпускались машины Единой Серии ЭВМ — ЕС ЭВМ. Эти машины в своих конструктивных и программных решениях повторяли IBM/360/370.

В конце 1960-х годов начинается выпуск серии ЭВМ PDP американской фирмы DEC. Это малые ЭВМ, способные работать в *режиме реального времени* и благодаря этому управлять работой различных технических устройств. Поэтому их еще называют управляющими машинами.

В этой линии наиболее известной серией были мини-ЭВМ семейства PDP-11. Архитектуру машин этой серии называют архитектурой «общей шины». *Общая шина* — это *информационная магистраль*, к которой подключаются процессор, память и контроллеры внешних устройств. Позже такая архитектура стала использоваться в микроЭВМ и персональных компьютерах (ПК).

В основе программного обеспечения мини-ЭВМ лежали *операционные системы реального времени*. Они управляли не только распределением ресурсов машины в мультипрограммном режиме работы, но и временем реакции ЭВМ на внешнее воздействие при управлении техническими устройствами: промышленными установками, лабораторным оборудованием, транспортными средствами и пр. Мини-ЭВМ стали работать даже на бортах космических кораблей. На базе мини-ЭВМ на предприятиях создаются АСУ ТП — автоматизированные системы управления технологическими процессами.

Со временем стала стираться разница между эксплуатационными качествами малых и больших машин. В 1970–1980 годах в США выпускается серия малых ЭВМ VAX-11 с 32-разрядным процессором, миллионным быстродействием и гигабайтной памятью. По тем временам такие машины можно было отнести к категории суперЭВМ.



Переход к **четвертому поколению ЭВМ** происходил *по двум направлениям*. Первое было связано с разработкой **суперкомпьютеров**, второе — с изобретением и использованием в ЭВМ **микропроцессоров**. И тот, и другой процессы берут свое начало в 1970-х годах.

Первый в истории микропроцессор появился в 1971 году. Тогда фирма Intel представила свой четырехразрядный микропроцессор Intel-4004, выполнявший 60 000 операций в секунду. Первая микроЭВМ на основе восьмиразрядного процессора Intel-8008 была создана в 1973 году.

Об истории микроЭВМ и персональных компьютеров будет рассказано подробно в следующем параграфе. А сейчас мы обсудим другую линию ЭВМ четвертого поколения: линию *суперкомпьютеров*.

Изобретатели ЭВМ понимали, что конструкция машины с одним центральным, «вычисляющим» процессором имеет естественный физический предел быстродействия. Он связан с двумя причинами: небесконечная величина скорости передачи сигнала между узлами компьютера и нарастающее тепловыделение при работе процессора на высоких значениях тактовой частоты. Причем второй фактор оказался наиболее существенным, поскольку процессор может попросту сгореть. Процессоры, работающие со скоростями, измеряемыми десятками миллионов операций в секунду, приближаются к такому пределу.

Поиски решения этой проблемы привели к отказу от фон-неймановской архитектуры ЭВМ с одним центральным процессором. Появились две фундаментальные идеи, лежащие в основе архитектуры суперкомпьютеров: принцип *конвейерной обработки* и принцип *векторной обработки*.

Сначала о **конвейерном методе**. Наверное, каждый из вас представляет себе производственный конвейер, например по сборке автомобилей. Сборка состоит из последовательных операций, каждую из которых выполняет один рабочий (или робот). Изделие в процессе перемещения по конвейеру постепенно приобретает свой окончательный вид. А теперь представьте себе, что все операции по сборке машины выполнял бы один универсальный рабочий-сборщик. При какой технологии производства завод будет выпускать больше машин за рабочий день? Конечно, при использовании конвейерной технологии!

В компьютере выполнение каждой операции (команды) разбивается на отдельные этапы. Процессорный конвейер — это цепочка процессоров, каждый из которых выполняет только один этап операции и передает результат следующему процессору в конвейере. После этого сразу же приступает к выполнению своего этапа следующей операции.

Пусть, для примера, выполнение каждой команды разбивается на 4 этапа и каждый процессор в конвейере выполняет свою работу за 1 единицу времени. Через 4 единицы времени с конвейера «сойдет» 1-я команда, через 5 единиц — 2-я команда и т. д.



Для выполнения N команд потребуется $N + 3$ единицы времени. Если бы команды последовательно выполнял один процессор, затрачивая 4 единицы времени на каждую, то N команд было бы выполнено за $4N$ единиц времени.

Конвейерный принцип работы используется не только в суперкомпьютерах, но и в обычных персональных компьютерах, начиная с процессора i80486. Подробнее об этом расскажем позже.

Векторная обработка предполагает использование множества процессорных элементов, которые могут работать параллельно, одновременно выполняя одну и ту же операцию над разными данными. Подобные вычисления часто выполняются при обработке массивов чисел — векторов. Например, при сложении двух одномерных массивов из 100 элементов $\{a_1, a_2, \dots, a_{100}\}$ и $\{b_1, b_2, \dots, b_{100}\}$ получается массив такого же размера, элементы которого вычисляются по формуле: $c_i = a_i + b_i, i = 1, \dots, 100$. Если эту задачу решает однопроцессорный компьютер, то сложения будут производиться последовательно и затраченное время на вычисления составит $100 \cdot \tau$ секунд, где τ — время выполнения операции сложения. Если же в компьютере имеются не менее 100 процессоров, то, работая параллельно, они затратят на эту работу всего τ с.

На применении конвейерного принципа работы при выполнении вычислений с векторами основаны **векторно-конвейерные ЭВМ**. **Матричные ЭВМ** работают по векторному принципу. При этом все процессорные элементы выполняют синхронно одну и ту же последовательность операций.

В **многопроцессорных ЭВМ** каждый процессорный элемент независимо от других работает по своей программе. Взаимодействие между ними происходит путем передачи данных. Эта передача осуществляется через память компьютера. Обеспечение быстрого обмена данными между параллельно работающими процессорами — узловая проблема многопроцессорных ЭВМ.

Первые суперкомпьютеры создаются в 1970-х годах. Большую известность имели машины, разработанные американским конструктором Сеймуром Крейем, основателем компании Cray Research по производству суперкомпьютеров. В 1976 году была выпущена ЭВМ CRAY-1 с производительностью 130 млн операций вещественной арифметики в секунду. В CRAY-1 использовалась *векторно-конвейерная технология* вычислений.

Одной из первых *матричных* суперЭВМ была машина ILLIAC-IV, разработанная в Иллинойском университете (1972 г.). Ее быстродействие составляло 200 млн оп./с.

В 1980-х годах Крей занимается разработкой многопроцессорных суперЭВМ. Модели CRAY-2 (рис. 2.15), CRAY-3 и др. достигают

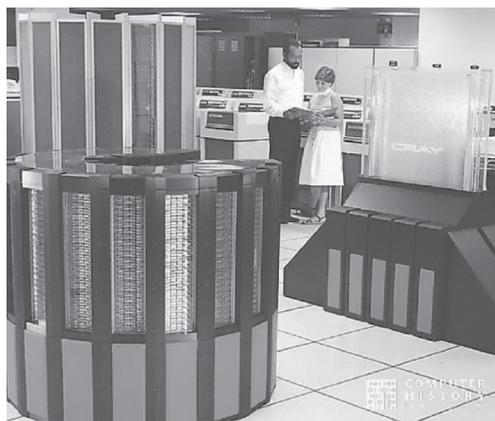


Рис. 2.15. Суперкомпьютер CRAY-2

быстродействия в 10^9 – 10^{10} оп./с. Высокими темпами в этот период развивается производство суперЭВМ в Японии. Японский суперкомпьютер SX-3 имел максимальное быстродействие 20 млрд оп./с.

Многопроцессорные ЭВМ 1980–1990-х годов содержали от нескольких единиц до нескольких десятков процессоров. Современные многопроцессорные суперЭВМ на основе микропроцессоров содержат тысячи параллельно работающих процессоров. Их быстродействие исчисляется величинами порядка 10^{12} оп./с.

Многопроцессорные суперкомпьютеры имеют очень высокую стоимость. Более дешевым способом обеспечения параллельных вычислений является технология распределенных вычислений.

Распределенные вычисления — способ реализации параллельных вычислений путем использования множества компьютеров, объединенных в сеть. Такие вычислительные системы еще называют мультикомпьютерными системами.

Распределенные вычисления часто реализуются с помощью **компьютерных кластеров** — нескольких компьютеров, связанных в локальную сеть и объединенных специальным программным обеспечением, реализующим параллельный вычислительный процесс.

Кластерные системы на сегодняшний день являются самым дешевым способом организации параллельных вычислений, поскольку для них можно использовать уже имеющиеся у пользователя компьютеры. Однако возможности организации взаимодействия отдельных компьютеров по сравнению с организацией узлов в многопроцессорной системе невелики, равно как и скорости передачи данных. Это обстоятельство накладывает ограничения на классы задач, для решения которых целесообразно использование кластерных систем.

Зачем нужны сверхбыстрые компьютеры. Несмотря на стремительно нарастающую производительность компьютеров, которая каждые 4–5 лет по важнейшим показателям практически удваивается, всегда есть классы задач, для которых существующей производительности не хватает. Укажем некоторые из них.

1. Расчеты, лежащие в основе реализации математических моделей многих процессов в природе и технике. Гигантские вычислительные ресурсы необходимы для более надежного и долгосрочного прогноза погоды, для решения аэрокосмических задач, в том числе и оборонных, для решения многих инженерных задач и т. д.
2. Поиск информации в гигантских базах данных, достигающих по объему хранимой информации нескольких терабайтов (1 терабайт = 1024 Гбайт = 10^{12} байтов).
3. Моделирование интеллекта. При всех фантастических показателях объем оперативной памяти современных компьютеров составляет лишь малую долю объема памяти человека.



Система основных понятий

Смена поколений ЭВМ				
	1 1950-е годы	2 1960-е годы	3 1970-е годы	4 (суперЭВМ) Начиная с 1970-х годов
<i>Элементная база:</i>	Электронные лампы	Транзисторы	Интегральные схемы (ИС) и большие интегральные схемы (БИС)	БИС, СВИС, микропроцессоры
<i>Максимальное быстродействие (оп./с):</i>	10–20 тыс.	100 тыс. – 3 млн	10 млн	10^9 – 10^{12}
<i>Максимальная емкость ОЗУ (Кбайт):</i>	100	1000	10 000	10^7
<i>Архитектура:</i>	фон-неймановская однопроцессорная	фон-неймановская однопроцессорная. Появление периферийных процессоров	Центральный процессор + каналы ввода/вывода. Шинная архитектура	Конвейерно-векторные, матричные, многопроцессорные. Мультикомпьютерные системы
<i>Примеры моделей и серий ЭВМ:</i>	М-20, Урал-4, Минск-14	М-220, БЭСМ-4, Минск-32, БЭСМ-6	IBM/360/370, PDP-11, ЕС ЭВМ	CRAY, ILLIAC IV, SX-3



Вопросы и задания

1. Расскажите о смене элементной базы компьютеров, происходившей при переходе от одного поколения к другому. Как при этом менялись основные характеристики ЭВМ? Подготовьте презентацию.
2. В чем состоял отход от архитектуры фон Неймана на ЭВМ второго и третьего поколений?
3. Что позволило реализовать мультипрограммный режим работы на ЭВМ третьего поколения?
4. Какие функции выполняли первые операционные системы?
5. В чем особенности мини-ЭВМ, отличавшие их от «больших» машин?
6. Назовите две линии в четвертом поколении ЭВМ.
7. Что такое суперкомпьютеры? Что такое конвейерная и векторная технологии?
8. Что представляют собой мультикомпьютерные системы?

2.4. Обработка чисел в компьютере

2.4.1. Представление и обработка целых чисел

Понятие числа в математике

Число — важнейшее математическое понятие. Первоначально математика оперировала лишь с целыми положительными (натуральными) числами. Понятие отрицательного числа вводится в работах Рене Декарта в XVII веке. В математике ряд целых чисел — это бесконечное дискретное множество (рис. 2.16).

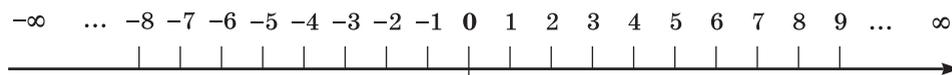


Рис. 2.16. Математическое множество целых чисел на числовой оси

Понятие вещественного (действительного) числа в математику ввел Исаак Ньютон в XVIII веке. С математической точки зрения, множество вещественных чисел бесконечно и непрерывно. Оно включает в себя бесконечное множество целых чисел и еще бесконечное множество нецелых чисел. Между двумя любыми точками на числовой оси лежит бесконечное множество вещественных чисел.

Целые числа в компьютере

Согласно принципу фон Неймана, числа в компьютере представляются и обрабатываются в двоичной системе счисления. С правилами представления целых чисел вы знакомились в курсе информатики 7–9 классов. Кратко повторим их.

Для хранения целых чисел могут использоваться ячейки разной длины: 1 байт, 2 байта, 4 байта. Далее для примера будем рассматривать двухбайтовое представление целых чисел.

Наименьшее по абсолютной величине число, хранимое в ячейке памяти, равно нулю (во всех разрядах — ноль). Самое большое положительное число имеет такое представление в памяти компьютера в двухбайтовой ячейке в двоичной форме:

01111111 11111111

Это же число в шестнадцатеричной форме:

7F FF

В десятичной системе это число равно $2^{15} - 1 = 32\ 767$.



Если размер ячейки памяти равен N битов, то максимальное целое положительное число, хранимое в ячейке, равно $2^{N-1} - 1$.

Первый слева двоичный разряд хранит код знака числа. У положительных чисел в этом разряде ноль, у отрицательных — единица.

Как получить представление целого положительного числа в памяти, например числа 5628? Покажем это на примере. Сначала переведем число в шестнадцатеричную систему.

$$5628_{10} = 15FC_{16}$$

Используя двоично-шестнадцатеричную таблицу, можно расписать двоичное представление этого числа в двухбайтовой ячейке памяти:

00010101 11111100

Теперь рассмотрим представление целого отрицательного числа в памяти компьютера. Действует следующее правило.



Отрицательные целые числа представляются в дополнительном двоичном коде.

Получим представление в памяти числа -5628 . Для этого нужно выполнить следующую последовательность действий.

1. Получить двоичное представление для абсолютной величины (положительного значения) этого числа; оно представлено выше и выглядит так:

00010101 11111100

2. Инвертировать этот двоичный код путем замены в нем всех единиц на нули, а нулей на единицы:

11101010	00000011
----------	----------

3. К этому коду добавить единицу; получим дополнительный код числа:

11101010	00000100
----------	----------

Это и есть искомое представление числа -5628 . В шестнадцатеричной форме оно выглядит так: EA 04.

Представление в памяти максимального по абсолютной величине отрицательного числа, равного -2^{15} , в шестнадцатеричной форме имеет вид: 80 00. Попробуйте доказать это самостоятельно.

Может возникнуть вопрос: в чем смысл дополнительного кода? Зачем производить все перечисленные действия? Чтобы на него ответить, сложим двоичные коды чисел $+5628$ и -5628 :

	00010101	11111100
+	11101010	00000100
1	00000000	00000000

Единица в старшем разряде вышла за пределы ячейки. Помните: в параграфе 2.1.2 мы говорили об исчезновении разряда переноса. Сумма, полученная в ячейке, равна нулю. Но так и должно быть!

Теперь можно объяснить смысл понятия «дополнительный код». Если X — двоичное положительное целое число, а \hat{X} — его дополнительный код, то $X + \hat{X} = 2^{16}$. В общем случае для N -разрядной ячейки выполняется: $X + \hat{X} = 2^N$. Следовательно, \hat{X} дополняет X до значения 2^N .

Максимальное по абсолютной величине отрицательное целое число, представляемое в N -разрядной ячейке памяти, равно -2^{N-1} .



Арифметическое устройство компьютера умеет выполнять операцию сложения, но не умеет выполнять вычитание. Вычитание в компьютере сводится к сложению первого слагаемого с дополнительным кодом второго слагаемого. Это одна из конструкторских «хитростей», позволяющая оптимизировать устройство процессора компьютера.

Рассмотренное представление целых чисел называется **представлением в формате с фиксированной запятой**.

Мы рассмотрели **формат представления целых чисел со знаком**, т. е. положительных и отрицательных. Бывает, что нужно работать только с положительными целыми числами. В таком

случае используется **формат представления целых чисел без знака**. В этом формате самое маленькое число — ноль (все биты — нули), а самое большое число для 16-разрядной ячейки:

1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

В десятичной системе это $2^{16} - 1 = 65\,535$. По модулю это в два раза больше, чем в представлении со знаком.

Из всего сказанного делаем вывод:



Целые числа в памяти компьютера — это дискретное ограниченное множество.

Границы этого множества зависят от размера выделяемой ячейки памяти под целое число, а также от формата: «со знаком» или «без знака». Шаг в компьютерной последовательности целых чисел, как и в математической, остается равным единице.

Особенности целочисленной машинной арифметики

Числовые расчеты могут производиться на множестве целых чисел или на множестве вещественных чисел. С математической точки зрения, целые числа являются подмножеством множества вещественных чисел. Поэтому, казалось бы, можно было и не разделять в компьютере числа на целые и вещественные и иметь дело только с вещественным числовым типом данных.

Однако целочисленная арифметика в компьютере имеет три очень существенных преимущества по сравнению с вещественной арифметикой:

- 1) целые числа всегда представимы своими точными значениями;
- 2) операции целочисленной арифметики дают точные результаты;
- 3) операции целочисленной арифметики выполняются быстрее, чем операции вещественной арифметики.

Недостатком целого типа данных является сравнительно узкий диапазон допустимых значений (для типа `integer` от $-32\,768$ до $32\,767$). При исполнении программы автоматически не контролируется выход значения целой величины за эти границы. Выход числового значения за допустимый в компьютере диапазон называется *эффектом переполнения*. *Переполнение при работе с целыми числами не вызывает прерывания работы процессора*. В таком случае получается ошибочный результат. Если такая опасность есть, то программист должен сам предусматривать в своей программе предупреждение целочисленного переполнения. Чаще всего целый тип используется для представления счетчиков, номеров, индексов и других целочисленных величин.



Пример. Определим, каким окажется результат вычисления произведения $20000 \cdot 2$, если числа представляются в формате с фиксированной запятой в двухбайтовой ячейке памяти.

Переведем число 20000 в двоичную систему счисления:

$$20000 = 4E20_{16} = 100\ 1110\ 0010\ 0000_2.$$

Умножим это число на $2 = 10_2$:

$$(0100\ 1110\ 0010\ 0000 \cdot 10)_2 = 1001\ 1100\ 0100\ 0000_2.$$

Запишем результат в двухбайтовую ячейку:

1	0	0	1	1	1	0	0	0	1	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

В формате с фиксированной запятой это отрицательное число, поскольку в 15-м разряде стоит единица. Определим, какому десятичному числу соответствует такой код. Для этого выполним последовательность преобразований из этого дополнительного кода в абсолютную величину числа. Это будут преобразования, обратные рассмотренным выше при получении кода отрицательного целого числа.

$$\begin{aligned} 1001\ 1100\ 0100\ 0000 &\rightarrow 1001\ 1100\ 0011\ 1111 \rightarrow \\ &\rightarrow 0110\ 0011\ 1100\ 0000 \rightarrow 63C0_{16} = 25536_{10}. \end{aligned}$$

Следовательно, результат умножения оказался таким: $20000 \cdot 2 = -25536$, что неверно. Правильный результат: $20000 \cdot 2 = 40000 > 32767$, он выходит за диапазон допустимых значений и вызывает переполнение. Но прерывания работы процессора при этом не произойдет.

Система основных понятий



Представление и обработка целых чисел	
Формат представления целых чисел с фиксированной запятой	
<i>Положительные числа</i>	<i>Отрицательные числа</i>
В N -разрядном двоичном коде	В дополнительном N -разрядном коде $\hat{X} = 2^N - X$
Максимальное значение: $2^{N-1} - 1$	Максимальное по модулю значение: -2^{N-1}
Особенности целочисленной машинной арифметики: <ol style="list-style-type: none"> 1) целые числа всегда представимы своими точными значениями; 2) операции целочисленной арифметики дают точные результаты; 3) операции целочисленной арифметики выполняются быстрее, чем операции вещественной арифметики; 4) целочисленное переполнение не вызывает прерывания работы процессора 	



Вопросы и задания

1. Почему множество целых чисел, представимых в памяти компьютера, дискретно и ограничено?
2. Определите диапазон целых чисел, хранящихся в 1 байте памяти, в двух вариантах: со знаком и без знака.
3. Получите представление в памяти компьютера числа 157 в 8-разрядной ячейке памяти в формате «со знаком».
4. Получите представление в памяти компьютера числа -157 в 8-разрядной ячейке памяти в формате «со знаком».
5. Получите представление в 16-разрядной ячейке памяти следующих целых десятичных чисел: а) 35, б) -35 , в) 967, г) -967 .
6. Определите, каким целым десятичным числам соответствуют следующие 16-ричные представления в памяти компьютера с двухбайтовой ячейкой: а) 7FFF, б) 016B, в) BFF5, г) FFFA.
7. Определите результат вычисления суммы целых десятичных чисел: $25000 + 30000$ в формате с фиксированной запятой для 16-разрядной ячейки.



2.4.2. Представление и обработка вещественных чисел

Первые микропроцессоры, которые применялись в персональных компьютерах, были приспособлены только для выполнения команд целочисленной арифметики (процессоры i8086, i80286, i80386). С дробными числами они работать не умели. В то же время решение многих задач научного и экономического содержания требует выполнения расчетов с нецелыми числами. В математике все множество целых и дробных чисел называется действительными числами, в программировании обычно их называют вещественными числами.

Существуют два пути решения этой проблемы. Первый путь — программная реализация операций над вещественными числами в рамках системы команд целочисленной арифметики. Например, операцию сложения вещественных чисел можно реализовать подпрограммой, состоящей из множества команд обработки целых чисел. По этой причине сложение вещественных чисел на компьютере будет происходить гораздо медленнее, чем сложение целых чисел.

Второй путь — использование специального процессора, в котором арифметические операции над вещественными числами реализованы аппаратно в его арифметическом устройстве. В этом случае разница между скоростью выполнения операций над целыми и над вещественными числами становится значительно меньше.

Такой процессор называется **математическим сопроцессором**. Он работает с вещественными числами, представленными в формате с плавающей запятой. В настоящее время в современных моделях компьютеров функции сопроцессора интегрированы в центральный процессор.

Формат с плавающей запятой

Формат с плавающей запятой использует представление вещественного числа R в виде произведения **мантиссы** m на основание системы счисления n в некоторой целой степени p , которую называют **порядком**: $R = m \cdot n^p$.

Представление числа в форме с плавающей запятой неоднозначно. Например, для десятичных чисел справедливы следующие равенства:

$$2,5324 = 0,25324 \cdot 10^1 = 0,0025324 \cdot 10^3 = 2532,4 \cdot 10^{-3} \text{ и т. п.}$$

Чтобы не было неоднозначности, в компьютере используют **нормализованное представление числа в формате с плавающей запятой**. Мантисса в нормализованном представлении должна удовлетворять условию: $0,1_n \leq m < 1_n$. Иначе говоря, мантисса должна быть меньше единицы и первая значащая цифра не должна быть нулем.

В памяти компьютера мантисса представляется как целое число, содержащее только значащие цифры (0 целых и запятая не хранятся). Представление вещественного числа в памяти сводится к представлению пары целых чисел: мантиссы и порядка.

В разных типах компьютеров применяются различные варианты представления чисел в формате с плавающей запятой. Для примера рассмотрим один из вариантов представления в памяти вещественного числа в четырехбайтовой ячейке памяти.

В ячейке должна содержаться следующая информация о числе: знак числа, порядок и значащие цифры мантиссы.

±	машинный порядок	М	А Н Т И С С	А
1-й байт	2-й байт	3-й байт	4-й байт	

В старшем бите 1-го байта хранится знак числа: 0 обозначает плюс, 1 — минус. Оставшиеся 7 битов первого байта содержат **машинный порядок**. В следующих трех байтах хранятся значащие цифры мантиссы (24 разряда).

В семи двоичных разрядах помещаются двоичные числа в диапазоне от 0000000 до 1111111. Значит, машинный порядок изме-



няется в диапазоне от 0 до 127 (в десятичной системе счисления). Всего 128 значений. Но порядок может быть как положительным, так и отрицательным. Разумно эти 128 значений разделить поровну между положительными и отрицательными значениями порядка: от -64 до 63 .



Машинный порядок смещен относительно математического и имеет только положительные значения. Смещение выбирается так, чтобы минимальному математическому значению порядка соответствовал ноль.

Связь между машинным порядком (Mr) и математическим (p) в рассматриваемом случае выражается формулой: $Mr = p + 64$.

Полученная формула записана в десятичной системе счисления. В двоичной системе формула имеет вид: $Mr_2 = p_2 + 100\ 0000_2$.

Для записи представления в памяти вещественного числа необходимо:

- 1) перевести модуль данного числа в двоичную систему счисления с 24 значащими цифрами;
- 2) нормализовать двоичное число;
- 3) найти машинный порядок в двоичной системе счисления;
- 4) учитывая знак числа, выписать представление числа в четырехбайтовом машинном слове.

Пример 1. Записать внутреннее представление числа 250,1875 в формате с плавающей запятой.

Решение

1. Переведем число в двоичную систему счисления с 24 значащими цифрами: $250,1875_{10} = 11111010,0011000000000000_2$.
2. Запишем его в форме нормализованного двоичного числа с плавающей запятой: $0,111110100011000000000000 \cdot 10^{1000}$. Здесь мантисса, основание системы счисления ($2_{10} = 10_2$) и порядок ($8_{10} = 1000_2$) записаны в двоичной системе.
3. Вычислим машинный порядок в двоичной системе счисления: $Mr_2 = 1000 + 100\ 0000 = 100\ 1000$.
4. Запишем представление числа в четырехбайтовой ячейке памяти с учетом знака числа. Цифры сверху указывают номера разрядов в ячейке:

31	30	24	23	16	15	8	7	0
0	1001000	11111010	00110000	00000000				

В шестнадцатеричной форме этот код имеет вид: 48FA3000.

Пример 2. По шестнадцатеричной форме представления числа в памяти компьютера в формате с плавающей запятой C9811000 восстановить десятичное число.

Решение

1. Перейдем к двоичному представлению числа в четырехбайтовой ячейке, заменив каждую шестнадцатеричную цифру четырьмя двоичными цифрами:

C9	81	10	00
1100 1001	1000 0001	0001 0000	0000 0000

2. Заметим, что получен код отрицательного числа, поскольку в старшем разряде с номером 31 записана 1. Получим порядок числа: $p = 1001001_2 - 1000000_2 = 1001_2 = 9_{10}$.
3. Запишем число в формате нормализованного двоичного числа с плавающей запятой с учетом знака числа: $-0,100000010001000000000000 \cdot 10^{1001}$.
4. Число в двоичной системе счисления имеет вид: $-100000010,001_2$.
5. Переведем число в десятичную систему счисления: $-100000010,001_2 = -(1 \cdot 2^8 + 1 \cdot 2^1 + 1 \cdot 2^{-3}) = -258,125_{10}$.

Диапазон вещественных чисел значительно шире диапазона целых чисел. Положительные и отрицательные числа расположены симметрично относительно нуля. Следовательно, максимальное и минимальное числа равны между собой по модулю.

Наименьшее по абсолютной величине число равно нулю. Наибольшее по абсолютной величине число в формате с плавающей запятой — это число с самой большой мантиссой и самым большим порядком.

Для четырехбайтового машинного слова таким числом будет:

$$0,111111111111111111111111 \cdot 10_2^{11111111}$$

После перевода в десятичную систему счисления получим: $(1 - 2^{-24}) \cdot 2^{63} \approx 10^{19}$.

Множество вещественных чисел, представимых в памяти компьютера в формате с плавающей запятой, является ограниченным и дискретным.

Количество математических действительных чисел, точно представимых в памяти компьютера, вычисляется по формуле:

$$N = 2^t \cdot (U - L + 1) + 1.$$

Здесь t — количество двоичных разрядов мантииссы; U — максимальное значение математического порядка; L — минимальное значение математического порядка. Для рассмотренного нами варианта ($t = 24$, $U = 63$, $L = -64$) получается: $N = 2\ 147\ 483\ 649$.

Особенности вещественной машинной арифметики

В языках программирования (в том числе в Паскале) целый тип данных относится к *порядковым типам*. Вспомним, что это значит:

- величины этого типа принимают конечное множество значений, которые могут быть пронумерованы;
- на множестве значений данного типа работают понятия: «предыдущий элемент», «последующий элемент».

Почему же вещественный тип данных не является порядковым?

Вещественные числа в памяти компьютера представляются в формате с плавающей запятой, т. е. в виде совокупности пары чисел: целого порядка и нормализованной мантииссы. Поскольку размер ячейки памяти ограничен, в большинстве случаев мантиисса оказывается «обрезанной», т. е. приближенной. Точное представление в памяти компьютера имеет лишь дискретное конечное множество вещественных значений. Поэтому множество вещественных чисел в машинном представлении есть дискретное, конечное множество, хотя оно и является отражением континуума действительных чисел.

На рисунке 2.17 изображена положительная часть действительной числовой оси, на которой штрихами отмечены значения, точно представимые в вещественном типе данных. Эта картина симметрично отражается на отрицательную полуось.

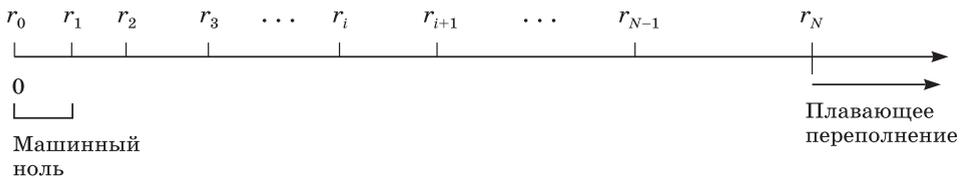


Рис. 2.17. Ряд точно представимых вещественных чисел

С ростом абсолютного значения числа интервал между соседними точками растет. Он равен (при двоичной нормализованной форме с плавающей запятой) $2^{-t} \cdot 2^p = 2^{p-t}$, где p — порядок числа, а t — количество двоичных разрядов в мантииссе. Ясно, что с ростом абсолютной величины числа его порядок (p) растет и, следовательно, растет шаг между двумя соседними значениями.

Минимальный шаг $\Delta r_{\min} = |r_1 - r_0| = 2^{p_{\min} - t}$; максимальный шаг: $\Delta r_{\max} = |r_N - r_{N-1}| = 2^{p_{\max} - t}$. Например, если $p_{\min} = -64$; $p_{\max} = 63$; $t = 24$, то имеем $\Delta r_{\min} = 2^{-88}$; $\Delta r_{\max} = 2^{39}$.

Казалось бы, значения множества точно представимых вещественных чисел можно пронумеровать и, таким образом, определить на нем понятия «следующий», «предыдущий». Однако расстояние между двумя последовательными значениями на этом множестве оказывается величиной «субъективной», в частности зависящей от размера ячейки памяти, в которой хранится число. Например, если под мантиссу выделяется 3 байта, то «следующее» значение получается путем прибавления к мантиссе величины 2^{-24} ; если 5 байтов, то 2^{-40} .

Если число X из бесконечного множества действительных чисел попадает между двумя точно представимыми значениями r_i и r_{i+1} , то оно заменяется на значение меньшего по модулю числа из этой пары. Следовательно, в общем случае действительные числа хранятся в памяти приближенно, т. е. несут в себе погрешность, которая называется *погрешностью машинного округления*.

Из сказанного следует, что если два действительных числа X и Y удовлетворяют условиям $r_i < X < r_{i+1}$; $r_i < Y < r_{i+1}$, но $X \neq Y$, то в машинном представлении они неразличимы.

Если в процессе вычисления порядок результата превышает максимально допустимый, то такая ситуация называется *переполнением порядка*. В этом случае процессор прерывает работу.

Диапазон значений $0 \leq r < r_1$, где r_1 — минимальное, не равное нулю значение, представимое в памяти машины, называется *машинным нулем*. Это значит, что любое число, лежащее в этом диапазоне, в компьютере представляется нулем.

Разность между вещественной единицей и ближайшим к ней числом, представимым в памяти машины, называется *машинным эпсилон* — ε . Иначе говоря, если $r_i = 1$, то $r_{i+1} = 1 + \varepsilon$. Легко понять, что величина машинного ε связана только с разрядностью мантиссы в представлении вещественных чисел на данном компьютере.

Для определения величины машинного ε можно использовать следующую программу:

```
Program Epsilon;
Var Eps: Real;
begin Eps:=1/2;
  while 1.0 + Eps > 1.0 Do
    begin
      Eps:=Eps/2;
      WriteLn(Eps)
    end
end.
```



Программа будет выводить на экран последовательность отрицательных степеней двойки: 2^{-2} , 2^{-3} , 2^{-4} , Последнее выведенное число будет равно машинному эпсилон. Вычисления, выполненные на компьютере автора, дали следующий результат:

$$\varepsilon = 2^{-41} \approx 4,55 \cdot 10^{-13}.$$



Система основных понятий

Представление и обработка вещественных чисел																						
Представление вещественных чисел																						
<i>Формат с плавающей запятой:</i>	$R = m \cdot n^p$ <i>m — мантисса, n — основание системы счисления; p — порядок. $0,1 \leq m \leq 1$ — условие нормализации</i>																					
<i>Размещение в 32-разрядной ячейке памяти:</i>	<table style="width: 100%; text-align: center; border-collapse: collapse;"> <tr> <td style="width: 5%;"></td> <td style="width: 10%;">31</td> <td style="width: 10%;">30</td> <td style="width: 10%;">24</td> <td style="width: 10%;">16</td> <td style="width: 10%;">8</td> <td style="width: 10%;">0</td> </tr> <tr> <td style="border: 1px solid black;">±</td> <td style="border: 1px solid black;">МП</td> <td style="border: 1px solid black;">М</td> <td style="border: 1px solid black;">А</td> <td style="border: 1px solid black;">Н</td> <td style="border: 1px solid black;">Т</td> <td style="border: 1px solid black;">И</td> </tr> <tr> <td style="border: 1px solid black;">С</td> <td style="border: 1px solid black;">С</td> <td style="border: 1px solid black;">А</td> <td colspan="4"></td> </tr> </table> <p><i>МП — машинный порядок: математический порядок + смещение; $МП \geq 0$</i></p>		31	30	24	16	8	0	±	МП	М	А	Н	Т	И	С	С	А				
	31	30	24	16	8	0																
±	МП	М	А	Н	Т	И																
С	С	А																				
Особенности вещественной машинной арифметики: 1) ограниченное дискретное множество значений; диапазон шире, чем у целых чисел; 2) приближенные вычисления; 3) вычисления происходят медленнее, чем для целых чисел; 4) переполнение порядка приводит к прерыванию работы процессора																						



Вопросы и задания

1. Почему множество действительных (вещественных) чисел, представимых в памяти компьютера, дискретно и ограничено?
2. Из каких двух частей состоит представление числа в формате с плавающей запятой?
3. Во сколько раз увеличится диапазон представления вещественных чисел в компьютере, если вместо одного байта под порядок числа будет выделено 2 байта?
4. Получите два варианта представления положительного десятичного числа 5: в формате с фиксированной запятой и в формате с плавающей запятой в четырехбайтовой ячейке.
5. Получите два варианта представления отрицательного десятичного числа -5: в формате с фиксированной запятой и в формате с плавающей запятой в четырехбайтовой ячейке.

6. В некотором компьютере для представления вещественного числа отводится 2 байта. Порядок занимает 7 битов. Сколько различных действительных чисел точно представимы в памяти такого компьютера?
7. Получите шестнадцатеричную форму внутреннего представления отрицательного числа $-123,125$ в формате с плавающей запятой в четырехбайтовой ячейке.

Практикум. Раздел 6 «Устройство компьютера»



2.5. Персональный компьютер и его устройство

2.5.1. История и архитектура персональных компьютеров

Первые микроЭВМ

Первыми электронными вычислительными машинами (ЭВМ) могли пользоваться только сотрудники крупных лабораторий и научно-исследовательских центров. Это было связано не только с их высокой стоимостью, но и с относительно большими размерами самих ЭВМ. И то, что в современном мире компьютеры можно встретить в офисах, школах, детских садах, квартирах, стало возможным благодаря разработке **микропроцессора**, который позволил значительно уменьшить размеры компьютера, сделав его действительно персональным устройством.

В 1969 году фирма Intel получила от японской компании Busicom заказ на разработку набора из 12 микросхем для семейства программируемых калькуляторов. В то время все микросхемы, выполняющие вычисления, разрабатывались специально под продукт заказчика. Однако сотрудники Intel Тед Хофф, Федерико Феджин и Стен Мэйзор предложили спроектировать единую универсальную микросхему — центральный процессор ЭВМ общего назначения, который взял бы на себя все функции отдельных микросхем. Главная особенность этого процессора заключалась в том, что в нем несколько интегральных микросхем размещались на одном кристалле кремния. Хофф разработал архитектуру микропроцессора, Мэйзор — систему команд, а Феджин спроектировал саму микросхему.

15 ноября 1971 года фирма Intel представила миру первый четырехразрядный микропроцессор, названный **i4004** (рис. 2.18). Характеристики первого микропроцессора: 4-разрядный, коли-



Рис. 2.18.

Первый микропроцессор

чество транзисторов 2300, тактовая частота 108 кГц. Появление i4004 стало первым шагом на пути развития персональных компьютеров, доступных не только специалистам в области вычислительной техники, но и обычным людям.

МикроЭВМ — это компьютер, построенный на базе микропроцессора. В апреле 1972 года фирма Intel выпустила 8-разрядный процессор **i8008** с тактовой частотой 0,5 МГц. На базе этого процессора в 1973 году создается первая микроЭВМ Intellec-8 (фирма Intel).

Первые микроЭВМ особого успеха у потенциальных пользователей не имели. Для их использования требовалось знание программирования. Стоимость первых микроЭВМ была достаточно велика (до \$2500). Чтобы снизить цену своей продукции, производители решили опробовать идею «компьютера-конструктора». Пользователь покупал набор деталей и инструкцию по сборке, после чего должен был самостоятельно паять и тестировать собранные узлы. В этих компьютерах не было ни клавиатуры, ни дисплея, ни долговременной памяти. Программы вводились переключением тумблеров на передней панели, а результаты считывались со светодиодных индикаторов. К компьютерам-конструкторам относятся **Micral**, **SCELB1-8H**, **MARK 8**. Стоимость таких компьютеров была около \$500–600. Но даже подобный маркетинговый ход не привлек большого количества покупателей.



Рис. 2.19. Компьютер «Altair-8800»

Первым микрокомпьютером, который привлек внимание потребителей, стал **Altair 8800** (рис. 2.19) фирмы MITS (1975 г.). В компьютере использовался процессор i8080 фирмы Intel. Разработчик Altair 8800 Эд Робертс решил использовать в компьютере системную плату с гнездами, куда подключались процессор, память и другие необходимые устройства. Подобная открытая архитектурная концепция была названа S-100 Bus.

Персональные компьютеры

Появление понятия «персональный компьютер» (ПК) связано с именами американских специалистов Стива Джобса и Стива Возняка и основанной ими компании Apple. В 1976 году создается ПК Apple-1, а в 1977 году — Apple-2 (рис. 2.20).

Главное отличие персональных компьютеров от микрокомпьютеров заключалось в том, что работать с ними могли не только специалисты по вычислительной технике и программисты, но и люди других профессий, а также школьники и студенты. В состав устройств ПК включаются графический дисплей, удобная клавиатура, принтеры. В 1978–1979 годах появляется прикладное программное обеспечение: текстовый редактор WordStar и табличный процессор VisiCalc.

Популярность персональных компьютеров привела к некоторому снижению спроса на большие компьютеры и мини-компьютеры (мини-ЭВМ). Это вызвало серьезное беспокойство у ведущей компании по производству компьютеров — фирмы IBM. В 1979 году IBM решила также выйти на рынок персональных компьютеров. Чтобы сэкономить деньги, руководство фирмы разрешило подразделению, ответственному за разработку персонального компьютера, при конструировании использовать блоки, изготовленные другими фирмами, что и было реализовано. Таким образом, в компьютерном мире окончательно сложился принцип *открытой архитектуры* персонального компьютера, заложенный еще разработчиками Altair 8800.



Рис. 2.20.
Персональный компьютер Apple-2

Открытая архитектура персонального компьютера — это архитектура, предусматривающая модульное построение компьютера с возможностью добавления и замены отдельных устройств благодаря наличию опубликованной документации на эти устройства.



Первый персональный компьютер фирмы IBM был выпущен в 1981 году и назван **IBM PC model 5150**. Он использовал процессор **i8088** фирмы Intel с тактовой частотой 4,77 МГц. Объем оперативной памяти составлял 64 Кбайт. Устройства долговременной памяти отсутствовали. Для этого процессора фирмой Microsoft была разработана новая операционная система **MS-DOS**.

Принцип открытой архитектуры привел к тому, что очень скоро компания IBM оказалась лишь одной из множества фирм, разрабатывающих и продающих персональные компьютеры. В 1983 году IBM выпустила модернизированный PC model 5160, который предусматривал возможность установки жесткого диска объемом 10 или 20 Мб. В 1984 году была выпущена следующая модель, названная PC AT (Advanced Technology — усовершенствованная

технология). В этой модели использовался новый 16-разрядный процессор Intel 80286 (**i80286**) с тактовой частотой до 20 МГц. На этом лидерство компании закончилось. Первый «IBM-совместимый» компьютер на базе процессора Intel 80386 был изготовлен компанией Compaq Computers. Это был первый 32-разрядный процессор, который положил начало семейству процессоров IA-32 (32-bit Intel Architecture).



Рис. 2.21.
Первый компьютер
Macintosh

В 1984 году фирма Apple выпустила первый персональный компьютер, в котором был графический интерфейс пользователя вместо стандартного на тот момент интерфейса командной строки. Для работы с этим интерфейсом впервые стало массово применяться новое устройство — мышь. Компьютер был назван **Macintosh** (рис. 2.21). Первоначально компьютеры Macintosh создавались на базе процессоров Motorola, которые позже были заменены более мощным процессором IBM PowerPC. В 2006 году Apple перешла на процессоры Intel. В отличие от своих конкурентов фирма Apple полностью самостоятельно разрабатывает операционную систему для своих компьютеров (**Mac OS**).

Эволюция микропроцессоров

Фирма **Intel** со времени выпуска самого первого процессора и по сей день является одним из ведущих разработчиков микропроцессоров. Чуть позже конкуренцию Intel составила фирма **AMD**. В настоящее время Intel и AMD являются ведущими производителями процессоров для персональных компьютеров. Некоторое время конкуренцию им составляли также фирмы **Apple Computer**, **IBM** и **Motorola**, разработавшие процессор **Power PC**, речь о котором уже шла выше.

Важным этапом в развитии микропроцессорной техники стал процессор **i80486**. Он был выпущен в 1989 году. Первое новшество заключалось в том, что впервые в центральный процессор стал интегрироваться *математический сопроцессор*, предназначенный для выполнения арифметических вычислений с плавающей запятой. Кроме того, в этом процессоре была реализован *конвейер*, применяемый ранее в суперЭВМ.

В марте 1993 года был выпущен процессор **Pentium**, в котором впервые появляется *предсказание переходов*. Суть предсказания переходов заключается в том, что при выполнении команды условного перехода (например, при выполнении команды ветвле-

ния) специальный блок микропроцессора определяет наиболее вероятное направление перехода, не дожидаясь окончания анализа условия. Процессор начинает выбирать из памяти и выполнять команды по предсказанной ветви программы. Так как направление перехода может быть предсказано неверно, получаемые результаты не записываются в память или регистры, а накапливаются в специальном буфере результатов. Если после анализа условия оказывается, что направление перехода было выбрано верно, все полученные результаты переписываются из буфера по месту назначения, а выполнение программы продолжается в обычном порядке. Если направление перехода предсказано неверно, то буфер результатов очищается и аннулируются результаты всех уже выполненных этапов этих команд конвейера. Конвейер начинает загружаться с первой команды другой ветви программы.

Для обеспечения оптимальной производительности переходы должны предсказываться максимально точно, иначе будет постоянно складываться ситуация, когда после выполнения перехода будет выясняться, что считалось совсем не то, что нужно. Максимальная эффективность предсказаний для Pentium составляла примерно 80%. Для современных процессоров точность предсказаний переходов составляет более 95%.

В конце 1993 года фирмы Apple Computer, IBM и Motorola совместно разработали микропроцессор **Power PC**. В 1994 году он стал использоваться в компьютерах **Macintosh**. В этом процессоре была реализована суперскалярная обработка, позволяющая выполнять в каждом такте 3 команды. Это стало возможным благодаря использованию архитектуры команд **RISC** (сокращенный набор команд постоянной длины). Все команды архитектуры **RISC** имеют одинаковую длину (что облегчает их выборку из памяти) и выполняются процессором за один такт. Все предыдущие процессоры либо использовали сложный набор команд **CISC** (расширенный набор команд переменной длины), либо относились к разряду **CISC**-процессоров с **RISC**-ядром. Процессорам, использующим архитектуру **CISC**, приходилось тратить дополнительное время на декодирование команд, так как их длина могла меняться от 8 до 108 битов.

Важным преимуществом процессоров **PowerPC** (начиная с моделей 603 и 604) была пониженная потребляемая мощность. В целях энергосбережения любой незагруженный исполнительный блок отключался, а при необходимости автоматически включался.

Однако в начале 2000-х годов развитие платформы **PowerPC** зашло в тупик. Создание новой архитектуры потребовало бы огромного количества времени и средств, поэтому в 2006 году фирма Apple решила перевести компьютеры **Macintosh** на процессоры **Intel**.

В марте 2000 года фирма AMD выпустила первый процессор с тактовой частотой, превышающей 1 ГГц, который назывался **Athlon K7**. Это позволило значительно укрепить позиции фирмы на рынке микропроцессорной техники.

В сентябре 2003 года AMD представила первые 64-разрядные процессоры для персональных компьютеров (**Athlon 64**).

Самым значимым событием 2005 года в области микропроцессоров стало появление в продаже CPU (центрального процессора) с двумя ядрами. Ядро представляет собой часть микропроцессора, содержащую его основные функциональные блоки и осуществляющую выполнение одного потока команд. О причинах перехода к многоядерным процессорам будет сказано в следующем параграфе. Первыми двухъядерными процессорами стали процессоры **Pentium D** фирмы Intel и **Athlon64 X2** фирмы AMD. Одними из наиболее революционных многоядерных процессоров стали процессоры линейки **Core 2 Duo** фирмы Intel.

В настоящее время основу рынка микропроцессоров составляют многоядерные процессоры, использующие в своем составе от 2 до 8 ядер, к каковым относятся процессоры **Intel Core 2 Quad**, **Phenom X3** и **X4** (фирмы AMD) и другие.



Система основных понятий

История и архитектура персональных компьютеров
1971 — создание первого микропроцессора i4004 (фирма Intel)
1975 — создание микроЭВМ Altair 8800 (фирма MITS)
1976 — концепция персонального компьютера, фирма Apple, ПК Apple-1
1981 — появление первого персонального компьютера фирмы IBM PC с окончательно сложившимся принципом открытой архитектуры
Открытая архитектура персонального компьютера — это архитектура, предусматривающая модульное построение компьютера с возможностью добавления и замены отдельных устройств благодаря наличию опубликованной документации на эти устройства
1984 — появление первого персонального компьютера Macintosh фирмы Apple с графическим интерфейсом пользователя
Наиболее важные этапы в развитии микропроцессоров: i80486 (1989) — встроенный сопроцессор, появление конвейерной обработки данных; Pentium (1993) — предсказание переходов; Power PC (1993) — использование архитектуры команд RISC; Athlon K7 (2000) — тактовая частота процессора впервые превысила 1 ГГц; Athlon 64 (2003) — первый 64-разрядный процессор; Pentium D (2005) — первый двухъядерный процессор



Вопросы и задания

1. Кто и когда разработал первый микропроцессор?
2. Во сколько раз тактовая частота первого микропроцессора была ниже максимальной тактовой частоты процессора Pentium IV (3800 МГц)?
3. В чем причина популярности персонального компьютера Altair 8800?
4. Что означает принцип открытой архитектуры? В каком компьютере были заложены основы этого принципа?
5. В чем особенности компьютера Macintosh?
6. В чем суть предсказания переходов?
7. Когда впервые в компьютерах стали применяться жесткие диски?
8. В какой модели процессора тактовая частота впервые превысила 1 ГГц?
9. В какой модели персонального компьютера стало массово использоваться устройство «мышь»?
10. В чем различие архитектуры команд RISC и CISC?
11. Какая фирма впервые выпустила 64-разрядные процессоры?

2.5.2. Микропроцессор: основные элементы и характеристики

Одним из важнейших устройств компьютера является **центральный процессор (CPU** — англ. *central processing unit*, что переводится как «центральное вычислительное устройство»). Именно от типа процессора и его характеристик в первую очередь зависит производительность компьютерной системы в целом.

Центральный процессор — это устройство компьютера, предназначенное для выполнения арифметических и логических операций над данными, а также координации работы всех устройств компьютера.

Состав микропроцессора

Современные центральные процессоры для персональных компьютеров выполняются в виде отдельных микросхем и называются **микропроцессорами**. В дальнейшем будем считать понятия «микропроцессор» и «процессор» равнозначными.

Схема состава микропроцессора показана на рис. 2.22.

Основным элементом микропроцессора является **ядро**, от которого зависит большинство характеристик самого процессора. Ядро представляет собой часть микропроцессора, содержащую его основные функциональные блоки и осуществляющую выполнение одного потока команд.



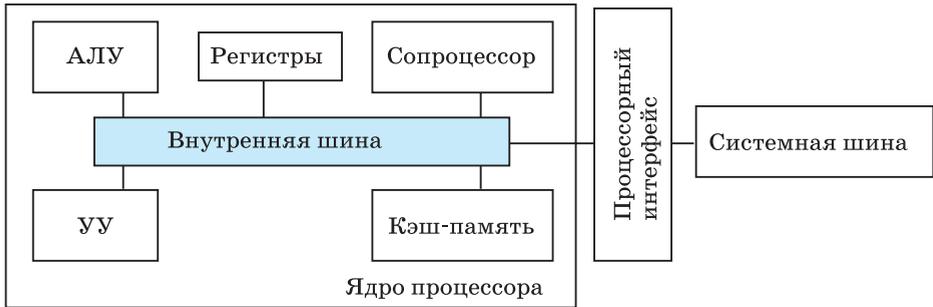


Рис. 2.22. Схема состава микропроцессора

Современные процессоры могут иметь более одного ядра, т. е. могут быть **многоядерными** (рис. 2.23). Многоядерные процессоры способны выполнять одновременно несколько потоков команд. Основная причина перехода к многоядерным процессорам была вызвана тем, что повышение производительности микропроцессоров путем дальнейшего наращивания тактовой частоты достигло физического предела в связи с очень высоким уровнем тепловыделения и энергопотребления. Производительность многоядерного процессора увеличивается за счет распараллеливания обработки данных между несколькими ядрами.



Рис. 2.23. Трехъядерный процессор AMD Phenom X3

Ядро процессора помещается в **корпус** (пластмассовый или керамический) и соединяется проводками с металлическими ножками (выводами), с помощью которых процессор присоединяется к системной плате компьютера. Количество выводов и их расположение определяют тип **процессорного интерфейса (разъема)**. Каждая системная плата ориентирована на один определенный тип разъема.



1. **Арифметико-логическое устройство (АЛУ)** выполняет все математические и логические операции.

2. **Управляющее устройство (УУ)** обеспечивает выполнение процессором последовательности команд программы.

3. **Набор регистров** — ячейки памяти внутри процессора, используемые для размещения команд программы и обрабатываемых данных.

4. **Кэш-память (кэш)** — сверхбыстрая память, хранящая содержимое наиболее часто используемых ячеек оперативной памяти, а также части программы, к которым процессор обратится с наибольшей долей вероятности. Процессор в первую очередь пытается найти нужные данные именно в кэш-памяти, а если их там не оказывается, обращается к более медленной оперативной памяти. Кэш-память делится на два или три уровня, которые обозначаются L1, L2 и L3 (чаще всего уровнем два).

5. **Сопроцессор** — элемент процессора, выполняющий действия над числами с плавающей запятой.

Характеристики микропроцессора

Тактовая частота. Для каждой выполняемой процессором команды требуется строго определенное количество единиц времени (тактов). Тактовые импульсы формируются генератором тактовой частоты, установленным на системной плате. Чем чаще они генерируются, тем больше команд процессор выполняет за единицу времени, т. е. тем выше его быстродействие. Тактовая частота обычно выражается в мегагерцах. 1 МГц равен 1 миллиону тактов в секунду. Первые модели процессоров Intel (i8008x) работали с тактовыми частотами, меньшими 5 МГц. Сегодня тактовая частота последних процессоров превышает 3 ГГц (1 ГГц = 1000 МГц). Внутренняя архитектура процессора, как и тактовая частота, также влияет на работу процессора, поэтому два CPU с одинаковой тактовой частотой не обязательно будут тратить одинаковое время на выполнение одной команды. Если, например, микропроцессору Intel 80286 требовалось 20 тактов, чтобы выполнить команду умножения двух чисел, то Intel 80486 или старше мог выполнить это же действие за один такт. Некоторые процессоры способны выполнять более одной команды за 1 такт. Их называют *суперскалярными*.

Различают внутреннюю и внешнюю тактовую частоту.

Внешняя тактовая частота — это частота, с которой процессор обменивается данными с оперативной памятью компьютера. Как уже было сказано выше, она формируется генератором тактовых импульсов (кварцевым резонатором).

Внутренняя тактовая частота — это частота, с которой происходит работа внутри процессора. Именно это значение указывается в прайс-листах фирм, продающих процессоры.

Первые процессоры имели одинаковую внутреннюю и внешнюю частоту, но, начиная с процессора i80486, для определения внутренней частоты стал применяться *коэффициент умножения*. Этот коэффициент определяется подачей напряжения на определенные контакты центрального процессора. Таким образом, для современных процессоров справедлива формула:

! Внутренняя тактовая частота =
= внешняя тактовая частота × коэффициент.

Например, если внешняя тактовая частота 133 МГц, а коэффициент умножения равен 10, то внутренняя тактовая частота равна $133 \cdot 10 = 1330$ МГц.

Разрядность процессора определяет количество битов данных, которые он может принять и обработать одновременно. Первые процессоры были 8-разрядные и 16-разрядные. Современные процессоры имеют разрядность 32 или 64 бита.

Объем кэш-памяти. Как уже было сказано, при поиске нужной информации процессор в первую очередь обращается к кэш-памяти. Поэтому чем выше ее объем, тем больше вероятность, что необходимые данные будут найдены именно там.

Технологические нормы. Технологические нормы определяют расстояние между соседними транзисторами. Чем меньше расстояние, тем короче каналы транзисторов и тем больше их быстродействие. Кроме того, уменьшение расстояния понижает уровень мощности тепловыделения. В настоящее время все процессоры производятся с технологическими нормами 0,09 микрона, 0,065 микрона и 0,045 микрона (1 микрон = 10^{-6} метров). Иногда технологические нормы указывают в нанометрах (1 нм = 10^{-9} м).

Количество ядер. Большинство современных процессоров выпускаются с несколькими ядрами (обычно их два или четыре). Благодаря наличию нескольких ядер процессор может одновременно обрабатывать несколько потоков программных команд, т. е. решать параллельно несколько задач в режиме реального времени.

Для определения основных характеристик процессора можно воспользоваться специальными сервисными программами. Примером такой программы является CPU-Z (рис. 2.24). Ее можно бесплатно скачать из Интернета.

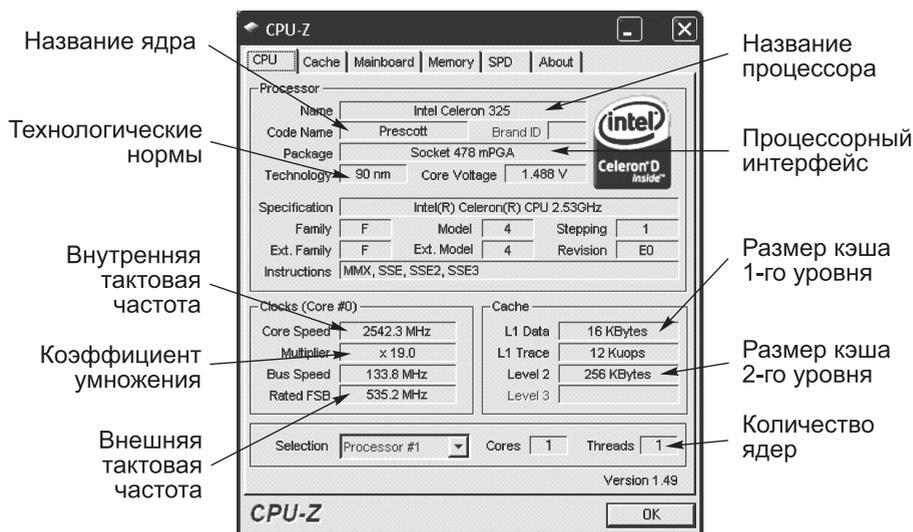


Рис. 2.24. Интерфейс программы CPU-Z

Система основных понятий



Микропроцессор: основные элементы и характеристики

Центральный процессор — это элемент компьютера, предназначенный для выполнения арифметических и логических операций над данными, а также координации работы всех устройств компьютера

Основные элементы ядра процессора

Арифметико-логическое устройство	Управляющее устройство	Набор регистров	Кэш-память	Сопроцессор
----------------------------------	------------------------	-----------------	------------	-------------

Основные характеристики процессора

Тактовая частота	Разрядность	Объем кэш-памяти	Технологические нормы	Количество ядер
------------------	-------------	------------------	-----------------------	-----------------

$$\text{Внутренняя тактовая частота} = \text{внешняя тактовая частота} \times \text{коэффициент}$$

Вопросы и задания



1. Какие функции выполняет центральный процессор?
2. Чем был вызван переход к многоядерным процессорам?
3. В чем причина использования сопроцессора?



4. Внутренняя тактовая частота процессора равна 2394 МГц. Коэффициент умножения равен 18. Определите внешнюю тактовую частоту (в МГц).
5. Как увеличение объема кэш-памяти влияет на производительность процессора?



Практикум. Раздел 6 «Устройство компьютера»

2.5.3. Системная (материнская) плата

Чтобы отдельные устройства компьютера могли взаимодействовать, их подключают к многослойной печатной плате, называемой **системной (материнской) платой** (рис. 2.25). Название происходит от английского *motherboard*. Иногда используется сокращение *МВ* или слово *mainboard* — «главная плата».

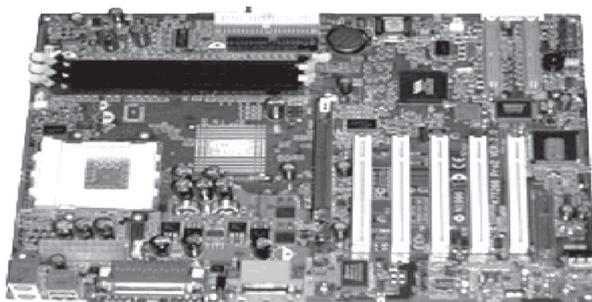


Рис. 2.25. Системная плата

Чипсет

Основой системной платы является набор ключевых микросхем, называемый **набором системной логики**, или **чипсетом** (chipset). Набор микросхем управляет соединениями процессора с различными компонентами компьютера. Именно он определяет тип и быстродействие используемого процессора, скорость, тип и объем оперативной памяти, а также потенциальные возможности компьютерной системы в целом.



Набор системной логики (чипсет) — это набор микросхем, обеспечивающий взаимодействие процессора с остальными компонентами компьютера.

Обычно чипсет состоит из двух компонентов, показанных на рис. 2.26. Называются эти компоненты *Северный мост* (North Bridge или MCH — Memory Controller Hub) и *Южный мост*

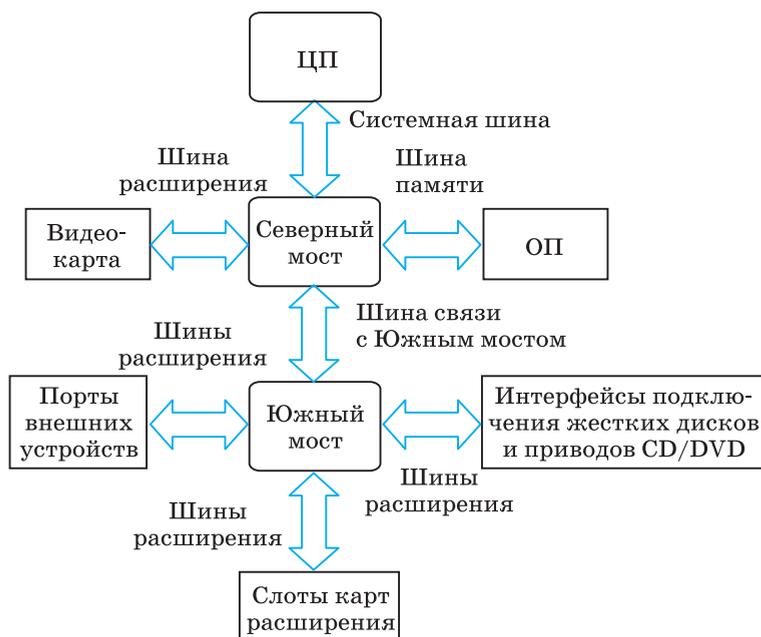


Рис. 2.26. Пример взаимосвязи основных компонентов материнской платы

(South Bridge или ICH — I/O Controller Hub). Северный мост работает с самыми скоростными устройствами, обеспечивая быструю и надежную связь процессора с оперативной памятью и видеоадаптером (видеокартой). Южный мост работает с более медленными устройствами, такими как жесткие диски, звуковые карты, приводы CD/DVD, сетевые карты, устройства USB и т. д. Южный мост также обеспечивает работу системных часов. Существуют модели чипсетов, в которых Северный и Южный мосты объединены в одну микросхему.

Шины

Для передачи информации между отдельными устройствами используются несколько видов шин.

Шины — это каналы связи, применяемые для передачи данных между отдельными устройствами компьютера и представляющие собой совокупность проводников, которые имеют определенные электрические характеристики.



По способу передачи данных шины делятся на *параллельные* и *последовательные*.

В **параллельных шинах** данные передаются одновременно по нескольким проводникам целыми группами битов. Количество проводников определяет *разрядность* шины.

В **последовательных шинах** данные передаются по одному проводнику бит за битом.

В связи с тем что при увеличении разрядности параллельных шин значительно возрастает уровень помех и усложняется процесс синхронизации всех параллельно идущих сигналов, в настоящее время происходит переход к последовательным шинам. Они работают с более высокой, чем у параллельных шин, тактовой частотой.

Любая шина состоит из трех частей.

1. **Шина данных.** По этой шине передаются данные между устройствами.
2. **Шина адреса.** По этой шине передаются адреса ячеек памяти.
3. **Шина управления,** предназначенная для передачи управляющих сигналов, т. е. команд.

Основной шиной большинства материнских плат (за исключением последних решений с отсутствующим Северным мостом) является **системная шина**. Системная шина работает в качестве канала связи между процессором и Северным мостом чипсета. Одной из важнейших характеристик системной шины является *тактовая частота*, которая в конечном счете определяет скорость передачи информации между двумя основными компонентами компьютера: процессором и чипсетом.

Шина памяти используется для передачи данных между процессором и оперативной памятью. Эта шина соединена с Северным мостом чипсета, и ее разрядность совпадает с разрядностью системной шины. Существуют модели процессоров, работающих с оперативной памятью напрямую, без участия Северного моста.

Шины расширения — это шины, используемые для подключения к материнской плате периферийных устройств.

Интерфейсы материнской платы и карты расширения

Подключение устройств к шинам осуществляется через интерфейсы.

Интерфейс — это элемент для соединения нескольких устройств.

На материнской плате многие интерфейсы представлены в виде специальных разъемов. Некоторые разъемы используются для подключения карт расширения и называются **слотами**.

Карта расширения (адаптер) — это печатная плата, которую подключают к материнской плате компьютерной системы с целью добавления дополнительных функций.



Через слоты подключаются такие карты расширения, как звуковые и сетевые карты, а также видеоадаптеры (видеокарты).

Видеоадаптер — это устройство, преобразующее код изображения, находящийся в памяти компьютера, в видеосигнал для монитора.

Звуковая карта — это устройство, позволяющее воспроизводить и записывать звук.

Сетевая плата (сетевая карта, Ethernet-адаптер) — печатная плата, позволяющая компьютерам взаимодействовать посредством локальной сети.

Для подключения к материнской плате периферийных устройств (таких как принтер, модем, клавиатура, мышь, сканер и т. п.) используются специальные интерфейсы, называемые **портами**.

Порт — это интерфейс для подключения периферийных устройств.



Рассмотрим основные порты материнской платы.

USB. Этот последовательный интерфейс служит для подключения различных внешних устройств. В настоящее время с интерфейсом USB выпускаются мыши, клавиатуры, принтеры, сканеры, флеш-накопители, жесткие диски (с помощью специального кабеля), модемы, джойстики.

LPT-порт (устаревший). Практически единственным широко распространенным внешним устройством с параллельным интерфейсом LPT является принтер.

COM-порт (устаревший) — последовательный порт, используемый ранее для подключения различных манипуляторов (мышь, трекбол), а также внешних модемов.

PS/2 — интерфейсы для подключения мыши и клавиатуры.

RJ-45 — порт для подключения сетевого кабеля.

AUDIO — разъемы для подключения звуковых устройств.



Система основных понятий

Системная (материнская) плата		
Взаимодействие отдельных устройств компьютера осуществляется с помощью многослойной печатной платы, называемой системной (материнской) платой		
Набор системной логики (чипсет) — это набор микросхем, обеспечивающий взаимодействие процессора с остальными компонентами компьютера		
<i>Северный мост</i>		<i>Южный мост</i>
Шины — это каналы связи, применяемые для передачи данных между отдельными устройствами компьютера и представляющие собой совокупность проводников, которые имеют определенные электрические характеристики		
Состав шины		
<i>Шина адреса</i>	<i>Шина данных</i>	<i>Шина управления</i>
Типы шин по способу передачи данных		
<i>Последовательные шины</i>		<i>Параллельные шины</i>
Интерфейс представлен на системной плате в виде разъема для подключения какого-либо устройства		
Порт — это интерфейс для подключения периферийных устройств		
Карта расширения (адаптер) — это печатная плата, которую подключают к материнской плате компьютерной системы с целью добавления дополнительных функций		
Слот — это интерфейс для подключения карт расширения к материнской плате		



Вопросы и задания

1. От какого элемента материнской платы зависит тип поддерживаемого процессора?
2. Какие устройства связывает системная шина?
3. В чем различие последовательных и параллельных шин? Почему дальнейшее развитие параллельных шин является неперспективным?
4. Что такое шины расширения?
5. Какие устройства подключаются через слоты расширения?
6. Какие устройства подключаются через порты?
7. Из каких трех частей состоит любая шина?
8. Через какие интерфейсы подключается (или могла подключаться ранее) мышь?



Практикум. Раздел 6 «Устройство компьютера»

2.5.4. Системная (внутренняя) память компьютера

Для реализации функции хранения информации в компьютере используются следующие **основные типы памяти**: *кэш-память*, *ПЗУ*, *оперативная память (ОЗУ)*, *долговременная (внешняя) память*. Первые три типа памяти образуют **внутреннюю (системную) память** компьютера. Основными характеристиками любого типа памяти являются объем, время доступа и плотность записи информации.

Объем памяти — это максимальное количество информации, которое может быть помещено в эту память. Объем памяти измеряется в килобайтах, мегабайтах, гигабайтах.

Время доступа к памяти — это минимальное время, достаточное для размещения в памяти единицы информации. Время доступа обычно измеряется в наносекундах (наносекунда — это одна миллиардная часть секунды).

Плотность записи информации представляет собой количество информации, записанной на единице поверхности носителя.

Кэш-память

Кэш-память является элементом микропроцессора. О ее назначении было сказано в параграфе 2.5.2. Физически кэш-память основана на микросхемах **статической памяти SRAM (Static Random Access Memory)**. Для создания ячейки статической памяти используется от 4 до 8 транзисторов, которые в совокупности образуют *триггер*. Статическая память работает гораздо быстрее динамической (речь о динамической памяти пойдет дальше), но к ее недостаткам относятся высокая стоимость и низкая плотность хранения информации. Эти два недостатка не позволяют использовать статическую память в качестве ОЗУ.

Постоянное запоминающее устройство (ПЗУ)

ПЗУ (постоянное запоминающее устройство) — энергонезависимая память, используемая только для чтения.

Данный вид памяти используется для хранения только такой информации, которая обычно не меняется в ходе эксплуатации компьютера. Типичным примером использования ПЗУ является хранение в нем базового программного обеспечения, используемого при загрузке компьютера (BIOS). Микросхемы ПЗУ (рис. 2.27) располагаются на материнской плате.

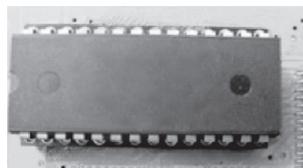


Рис. 2.27. Микросхема ПЗУ



Из всех трех типов системной памяти ПЗУ имеет самое большое время доступа, т. е. является наиболее медленной. Ее объем обычно равен 128 или 256 Кбайт.

Оперативное запоминающее устройство



ОЗУ (оперативное запоминающее устройство) — энергозависимая память, применяемая для временного хранения команд и данных, необходимых процессору для выполнения текущих операций.

Как вы уже знаете, наименьшей частицей памяти является бит, в котором хранится либо 0, либо 1. Отдельные биты объединяются в **ячейки**, каждая из которых имеет свой адрес, поэтому процессор при необходимости может обратиться к любой ячейке за одну операцию. Минимальной адресуемой ячейкой оперативной памяти является байт. Для выбора нужной ячейки используется ее адрес, передаваемый по адресной шине. Адресация байтов начинается с нуля.

Несмотря на то что минимальной адресуемой ячейкой оперативной памяти является байт, физически по шине передаются не отдельные байты, а **машинные слова**. Размер машинного слова зависит от разрядности процессора. То есть размер машинного слова определяется количеством битов, к которым процессор имеет одновременный доступ. Например, для 16-разрядного процессора размер машинного слова будет равен 2 байтам. Адрес машинного слова равен адресу младшего байта, входящего в состав это слова. Например, для 32-разрядного процессора адресация машинных слов будет выглядеть следующим образом:

Машинное слово	Байты			
	0	1	2	3
0	0	1	2	3
4	4	5	6	7
8	8	9	10	11
...				

Физически ОЗУ строится на микросхемах **динамической памяти DRAM (Dynamic Random Access Memory)**. В динамической памяти ячейки построены на основе областей с накоплением зарядов (конденсаторов), занимающих гораздо меньшую площадь, чем триггеры, и практически не потребляющих энергии при хранении. При записи бита в такую ячейку в ней формируется электрический заряд, сохраняющийся в течение 2–4 миллисекунд. Но для сохранения заряда ячейки необходимо постоянно регенерировать (перезаписывать) ее содержимое. В связи с этим скорость доступа к ячейкам ОЗУ ниже, чем к статической памяти. Для

создания ячейки динамической памяти достаточно всего одного транзистора и одного конденсатора, поэтому она дешевле статической памяти и имеет бóльшую плотность упаковки.

Оперативная память изготавливается в виде небольших печатных плат с рядами контактов, на которых размещаются интегральные схемы памяти (модули памяти, рис. 2.28). Модули памяти различаются по размеру и количеству контактов (в зависимости от типа используемой памяти), а также по быстродействию и объему. Объемы оперативной памяти современных компьютеров могут измеряться несколькими гигабайтами (в среднем от 1 до 4 Гбайт).

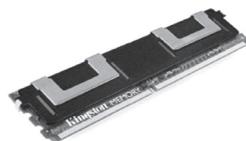


Рис. 2.28. Модуль оперативной памяти

Система основных понятий

Системная (внутренняя) память компьютера		
Системная память компьютера		
Кэш-память	Постоянное запоминающее устройство (ПЗУ)	Оперативное запоминающее устройство (ОЗУ)
ПЗУ — энергонезависимая память, используемая только для чтения. ОЗУ (оперативное запоминающее устройство) — энергозависимая память, применяемая для временного хранения команд и данных, необходимых процессору для выполнения текущих операций		
Байт — минимальная адресуемая ячейка оперативной памяти. Адресация байтов начинается с нуля		
Машинное слово — это последовательность байтов, рассматриваемая при передаче данных по шине как единое целое. Размер машинного слова зависит от разрядности процессора. Адрес машинного слова совпадает с адресом младшего байта		

Вопросы и задания

1. Запишите названия всех трех типов системной памяти в порядке возрастания объема, а затем — в порядке убывания времени доступа.
2. Почему статическая память не используется в качестве оперативной памяти?
3. Почему для динамической памяти требуется постоянная регенерация?
4. Байты с какими номерами (адресами) входят в 32-разрядное машинное слово с адресом 18 (адрес слова дан в шестнадцатеричной системе счисления)?
5. Что хранится в ПЗУ?

Практикум. Раздел 6 «Устройство компьютера»



2.5.5. Долговременная (внешняя) память компьютера

Долговременная (внешняя) память — это энергонезависимая память, предназначенная для длительного хранения информации.

Процессор не имеет прямого доступа к содержимому внешней памяти. Чтобы процессор мог обработать данные из долговременной памяти, они должны быть сначала загружены в оперативную память. В настоящее время к основным устройствам долговременной памяти относятся жесткие магнитные диски, накопители на оптических дисках, устройства флеш-памяти. Ранее для длительного хранения информации использовались также магнитные ленты, дискеты, магнито-оптические диски.

Жесткий магнитный диск

Основным устройством внешней памяти является **жесткий магнитный диск**.

Внутри жесткого диска находятся одна или несколько пластин, насаженных на общий шпиндель. Данные обычно записываются на обеих сторонах каждой пластины, хотя в некоторых жестких дисках производители наряду с двухсторонними пластинами могут использовать и односторонние. Запись и чтение информации осуществляются с помощью головок чтения/записи. Под пластинами располагается двигатель, который вращает их с достаточно большой скоростью. Скорость вращения пластин измеряется в оборотах в минуту (rpm). Первые жесткие диски имели скорость вращения 3600 rpm. В современных жестких дисках скорость вращения возросла до 7200, 10 000 и 15 000 оборотов в минуту.

В процессе записи цифровая информация, хранящаяся в оперативной памяти, преобразуется в переменный электрический ток, который поступает на магнитную головку, а затем передается на магнитный диск, но уже в виде магнитного поля. После прекращения действия внешнего поля на поверхности диска образуются зоны остаточной намагниченности.

Перед использованием жесткого диска необходимо выполнить операцию его форматирования. Форматирование включает в себя три этапа.

1. **Низкоуровневое форматирование диска.** При этом процессе на жестком диске создаются *физические структуры: дорожки, секторы, управляющая информация*. Этот процесс выполняется заводом-изготовителем на пластинах, которые не содержат еще никакой информации.

2. **Разбиение на разделы.** Этот процесс разбивает жесткий диск на *логические диски* (C:, D: и т. д.). Эту функцию выполняет операционная система.
3. **Высокоуровневое форматирование.** Этот процесс также выполняется операционной системой и зависит от ее типа. При высокоуровневом форматировании создаются логические структуры, ответственные за правильное хранение файлов, а также, в некоторых случаях, системные загрузочные файлы в начале диска.

Рассмотрим подробнее, что происходит с диском при низкоуровневом форматировании. Как уже было сказано выше, пластины жесткого диска делятся на **дорожки** и **секторы**. Каждая из дорожек представляет собой кольцо. На пересечении дорожек и секторов образуются **блоки** (рис. 2.29). Обычно объем одного блока составляет 512 байтов. Одинаково расположенные дорожки на всех сторонах пластин образуют **цилиндр** (рис. 2.30).

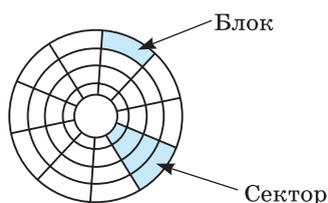


Рис. 2.29. Секторы и блоки жесткого диска

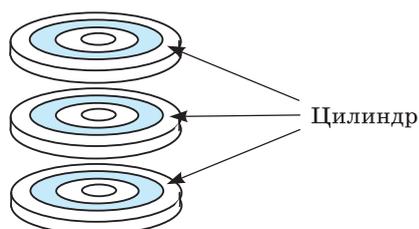


Рис. 2.30. Цилиндры жесткого диска

Ранее для разметки жестких дисков использовалось стандартное форматирование, при котором количество блоков на всех дорожках пластины было одинаковым. В современных жестких дисках используется **зонная запись**. При зонной записи по мере продвижения к внешнему краю диска дорожки разбиваются на всё большее число секторов. Дорожки с одинаковым количеством секторов образуют **зону** (рис. 2.31). Метод зонной записи позволил значительно увеличить емкость жестких дисков.

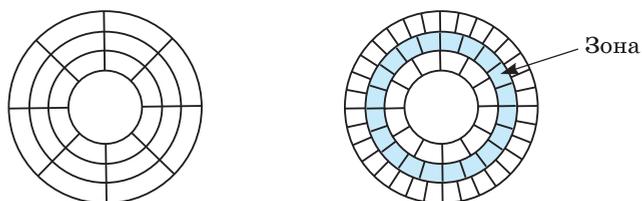


Рис. 2.31. Стандартное форматирование (слева) и зонная запись (справа)

Емкость современных жестких дисков измеряется сотнями гигабайтов. Существуют жесткие диски объемом 1 терабайт (1 терабайт = 1024 гигабайта) и более.

Для повышения надежности хранения данных, а также для повышения скорости чтения/записи информации при работе с большими объемами данных используются массивы из нескольких дисков, управляемых контроллером и воспринимаемых компьютерной системой как единое целое — **RAID-массивы**. Существуют несколько уровней RAID-массивов, отличающихся друг от друга способом записи информации. Например, в массивах RAID 1 используется идея зеркалирования, заключающаяся в том, что все данные одновременно записываются на два жестких диска. Это значительно повышает надежность хранения информации. В массивах RAID 0 используется идея чередования, когда чтение и запись параллельно ведутся на несколько жестких дисков. Записываемый файл разбивается на части определенного размера и посылается одновременно на все имеющиеся накопители. Это увеличивает скорость работы с диском.

Оптические диски

Жесткие диски изначально создавались в качестве внутренних устройств и не были предназначены для резервного копирования и переноса информации с одного компьютера на другой. Около 10–15 лет назад самым распространенным устройством, предназначенным для этих целей, были **дискеты (гибкие магнитные диски)**. Однако их емкость по современным меркам была очень мала (1,44 Мбайт), поэтому на смену им пришли **оптические диски CD (компакт-диски)**, позволяющие хранить достаточно большие объемы информации (650–800 Мбайт) и намного превосходящие дискеты по степени надежности. Для работы с компакт-дисками на компьютере необходимо наличие специального привода (оптического накопителя).

Различают диски «только для чтения» (**CD-ROM**), изготавливаемые промышленным способом, для однократной записи (**CD-R**) и для **многократной записи (CD-RW)**. Диски последних двух типов предназначены для записи на специальных пишущих оптических накопителях. Все типы дисков имеют одинаковую структуру хранения информации. Данные с помощью луча красного лазера записываются на спиральную дорожку, идущую от центра диска к его периферии. Вдоль дорожки располагаются углубления, называемые *питами* (pit — «углубление»). На записываемых дисках питы имитируются темными пятнами специального регист-

пирующего слоя, получившимися в результате нагрева нужного участка лазером. Чередованием углублений и промежутков между ними и кодируется любая информация.

Диски DVD имеют более высокую плотность записи данных, чем CD-диски. Существуют диски, на которых запись информации производится в два слоя. В зависимости от указанных выше параметров DVD-диски могут иметь объем 4,7 Гб или 8,5 Гб.

Все компакт-диски (и CD, и DVD) имеют одинаковую структуру хранения информации. Скорость чтения/записи оптических приводов измеряется в единицах, кратных базовой скорости (обозначается 16x, 24x, 48x и т. д.). Для приводов CD базовая скорость равна 150 Кб/с, для DVD — 1,385 Мб/с.

Blu-ray (Blu-ray Disc) является названием формата оптического диска следующего поколения. В Blu-Ray для записи и чтения данных вместо красного лазера, который используется в DVD и CD-ROM, применен синий лазер. У синего лазера длина волны значительно меньше длины волны красного лазера. Это позволяет сделать толщину дорожки данных тоньше, что приводит к значительному увеличению емкости носителя. Формат был разработан для обеспечения возможности записи, перезаписи и воспроизведения видео высокого разрешения (HD-video), а также для хранения больших объемов данных. Емкость нового формата — от 25 до 50 Гб.

Флеш-память

По устройству флеш-память (flash-память) напоминает микросхему динамической энергозависимой памяти, в которой вместо конденсаторов в ячейках памяти установлены транзисторы. При подаче напряжения транзистор принимает одно из фиксированных положений — закрытое или открытое. Он остается в этом положении до тех пор, пока на него не будет подан новый электрический заряд, изменяющий его состояние. Таким образом, последовательность логических нулей и единиц формируется в этом типе памяти подобно статической памяти: закрытые для прохождения электрического тока ячейки распознаются как логические единицы, открытые — как логические нули. Можно сказать, что флеш-память объединяет в себе качества статической и динамической памяти: она позволяет сравнительно быстро записывать и считывать данные и хранит их даже после выключения питания. Эта особенность позволяет использовать микросхемы флеш-памяти в качестве альтернативы компакт-дискам и жестким дискам.



Рис. 2.32.

USB flash drive темпами.

USB flash drive (флеш-накопитель, рис. 2.32) — устройство на основе флеш-памяти для хранения и переноса данных с одного компьютера на другой. Флеш-память заключена в корпус, напоминающий по внешнему виду брелок. Интерфейс подключения к компьютеру — USB. Емкость современных флеш-накопителей достигает 16–32 Гб и продолжает расти быстрыми



Система основных понятий

Долговременная (внешняя) память компьютера						
Внешняя (долговременная) память — это энергонезависимая память, предназначенная для долговременного хранения информации						
Основные устройства внешней памяти						
<i>Жесткие магнитные диски</i>	<i>Накопители на оптических дисках CD и DVD</i>			<i>Флеш-накопители</i>		
В процессе низкоуровневого форматирования происходит разметка пластин жесткого диска на дорожки и секторы. При высокоуровневом форматировании создаются логические структуры, ответственные за правильное хранение файлов						
Цилиндр — это одинаково расположенные дорожки на всех сторонах пластин жесткого диска						
Основные типы оптических дисков						
CD-ROM	CD-R	CD-RW	DVD-ROM	DVD-R	DVD-RW	Blu-ray
USB flash drive — устройство для хранения и переноса данных с одного компьютера на другой, основанное на использовании флеш-памяти.						



Вопросы и задания

1. Какие устройства долговременной памяти используются на вашем рабочем компьютере?
2. Назовите основные элементы жесткого диска.
3. Что такое цилиндр жесткого диска?
4. Подумайте, почему использование зонной записи позволило увеличить объем жесткого диска.
5. За счет чего производители могут увеличивать объемы жестких дисков?
6. В чем различие дисков CD-ROM и CD-R?
7. Какой объем могут иметь диски DVD? От чего он зависит?



8. Диск CD-R имеет скорость чтения 48x, диск DVD-R — 16x. Какой из дисков будет быстрее считывать информацию и почему?
9. Имеется комбинированный привод CD/RW—DVD/ROM. Можно ли с помощью него: а) записывать диски DVD; б) читать диски blu-ray; в) записывать диски CD-R?
10. Где используется флеш-память?
11. Каков объем жесткого диска со следующими параметрами: количество головок — 16, количество цилиндров — 1024, количество секторов — 63, объем блока — 512 байтов (считать, что используется стандартное форматирование)?

Практикум. Раздел 6 «Устройство компьютера»



2.5.6. Устройства ввода и вывода информации

Устройства ввода предназначены для ввода информации в компьютер.



К устройствам ввода относятся клавиатура, мышь, сканер, микрофон, джойстик, световое перо, веб-камера и ряд других устройств.

Клавиатура

Основным устройством ввода символьной информации является **клавиатура**. Нажатие клавиш замыкает определенные электрические контакты клавиатуры, и, в зависимости от нажатой клавиши или их комбинации, в память компьютера передается специальный скан-код или их последовательность. Преобразование скан-кода в код ASCII выполняют соответствующие модули базовой системы ввода/вывода (BIOS).

Для пользователей, много времени проводящих за клавиатурой, важную роль может играть ее эргономичность. **Эргономичность** означает оптимальную приспособленность определенного устройства к физиологии человека. Эргономичные клавиатуры (рис. 2.33) обладают некоторыми дополнительными свойствами. Например, имеют развернутые под небольшим углом вертикальные ряды клавиш, относящиеся к зонам действия рук, или подставку для кистей рук.

Существуют беспроводные клавиатуры, которые для связи с компьютером используют радиоканал.



Рис. 2.33. Эргономичная клавиатура Microsoft

Мышь

Компьютерная мышь относится к устройствам ввода, обеспечивающим интерфейс пользователя с компьютером. С помощью мыши пользователь указывает на те или иные объекты на экране монитора, а также выбирает действие, которое необходимо выполнить с этими объектами. Кроме того, компьютерные мыши используются для рисования объектов в графических редакторах.

Современные мыши (**оптические**) определяют свое месторасположение путем высокоскоростного сканирования поверхности, на которой они находятся. Снизу мыши расположен светодиод, освещающий поверхность, по которой перемещается мышь. Рядом размещена маленькая камера, которая порядка тысячи раз в секунду сканирует поверхность под мышью. Данные передаются в специальный цифровой сигнальный процессор (DSP), вычисляющий расстояние, на которое должен переместиться указатель мыши на экране, а также направление перемещения.

Существуют **беспроводные мыши** с радио- и инфракрасными передатчиками.

Сканер

Сканером (рис. 2.34) называется устройство для ввода в компьютер изображений, нанесенных на плоскую поверхность. Сканер



Рис. 2.34. Сканер

позволяет вводить в компьютер изображения текстов, рисунков, слайдов, фотогафий, чертежей и другой графической информации. В основе принципа работы сканера лежит отражение от объекта или прохождение через объект света. Свет от яркой лампы, располагающейся внутри устройства, отражается от сканируемого объекта. Сам объект предварительно размещается на стекле сканера изображением вниз. Приемник света фиксирует яркость и цвет отражения от каждой точки, а затем преобразует световые импульсы в электрический сигнал.

Прочие устройства ввода информации

Графический планшет — устройство для ввода рисунков от руки непосредственно в компьютер. Состоит из пера и плоского планшета, чувствительного к нажатию пера.

Микрофон — устройство для преобразования звука в электрический сигнал.

Веб-камера — цифровое устройство, производящее видеосъемку, оцифровку, сжатие и передачу цифрового видео по компью-

терным сетям. В последние годы веб-камеры находят очень широкое применение для проведения видеоконференций, а также для организации системы видеонаблюдения.

Джойстик — устройство управления в компьютерных играх. Представляет собой рычаг на подставке, который можно отклонять в двух плоскостях. На рычаге могут быть различные гашетки, кнопки и переключатели.

Устройства вывода предназначены для передачи информации от компьютера к пользователю.

Основными устройствами вывода являются монитор, принтер, акустические колонки, наушники, плоттер.

Мониторы

Монитор — это устройство для визуального отображения (вывода) текстовой и графической информации.

Самым распространенным в настоящее время типом мониторов являются жидкокристаллические мониторы LCD. Однако еще достаточно большое число пользователей применяют устаревшие мониторы с электронно-лучевой трубкой (CRT-мониторы). Существуют также газоплазменные мониторы, которые пока являются достаточно большой редкостью ввиду их высокой цены.

CRT-мониторы (*Cathode Ray Tube* — «катодно-лучевая трубка», рис. 2.35). В основе этих мониторов лежит катодно-лучевая или электронно-лучевая трубка (ЭЛТ). Внутри этой трубки вакуум. С передней стороны монитора внутренняя часть трубки покрыта слоем люминофора, который светится при бомбардировке его заряженными частицами, испускаемыми тремя электронными пушками. Люминофор наносится в виде наборов точек трех основных цветов — красного (Red), зеленого (Green) и синего (Blue). Сочетаниями этих цветов с различной степенью интенсивности можно представить все то многообразие цветов, которое поддерживается современными видеокартами (более 16 млн цветов).

Наборы точек люминофора располагаются по треугольным триадам. Такую триаду точек называют *пикселем*. Из пикселей формируется изображение на экране монитора. Расстояние между центрами пикселей по диагонали называется **размером зерна (пикселя)**. Чем меньше размер зерна, тем выше четкость изображения.



Рис. 2.35. CRT-монитор



Важной характеристикой мониторов является **разрешающая способность**, которая определяется количеством пикселей, размещающихся по горизонтали и вертикали монитора. Современные мониторы должны обеспечивать разрешающую способность не менее 1024×768 пикселей.

Жидкокристаллические (ЖК) мониторы (LCD — Liquid Crystal Display, рис. 2.36) используют так называемые жидкие кристаллы, которые могут изменять свою структуру и прозрачность под действием электрического напряжения. Когда нет электрического заряда, жидкие кристаллы находятся в аморфном состоянии и пропускают свет. Количеством света, проходящего через жидкие кристаллы, можно управлять с помощью электрических зарядов. Лампы подсветки LCD-монитора освещают экран, а жидкие кристаллы для каждой ячейки пропускают только ту часть света, которая необходима для придания ячейке нужной яркости. Большинство жидкокристаллических мониторов используют тонкую пленку из жидких кристаллов, помещенную между двумя стеклянными пластинами. Каждому пикселю изображения соответствуют три ячейки для отображения красной, зеленой и синей точек.



Рис. 2.36.
LCD-монитор

Преимуществами жидкокристаллических мониторов являются их компактность, низкое энергопотребление, отсутствие электромагнитного излучения, высокие уровни яркости. Однако, в отличие от CRT-мониторов, у LCD-мониторов есть такая характеристика, как стандартное разрешение. Только в стандартном разрешении ЖК-монитор воспроизводит изображение наиболее качественно. Разрешение больше стандартного нельзя установить в принципе. При уменьшении разрешения изображение становится нечетким, с «зазубринами». Также, по сравнению с мониторами ЭЛТ, у жидкокристаллических мониторов хуже точность цветопередачи. Существенным недостатком является зависимость контрастности от угла обзора.

Преимуществами жидкокристаллических мониторов являются их компактность, низкое энергопотребление, отсутствие электромагнитного излучения, высокие уровни яркости. Однако, в отличие от CRT-мониторов, у LCD-мониторов есть такая характеристика, как стандартное разрешение. Только в стандартном разрешении ЖК-монитор воспроизводит изображение наиболее качественно. Разрешение больше стандартного нельзя установить в принципе. При уменьшении разрешения изображение становится нечетким, с «зазубринами». Также, по сравнению с мониторами ЭЛТ, у жидкокристаллических мониторов хуже точность цветопередачи. Существенным недостатком является зависимость контрастности от угла обзора.

Принтеры

Принтер — это устройство для вывода цифровой информации на бумагу.

Существуют три основных типа принтеров: матричные, струйные и лазерные.

Матричные принтеры формируют изображение построчно с помощью печатающей головки, которая ударяет по бумаге через

красящую ленту. Головка содержит ряд иголок (pin), от количества которых зависит качество изображения. В настоящее время этот тип принтеров устарел из-за большого количества недостатков, таких как низкая скорость печати, высокий уровень шума при работе, низкое качество изображения. К плюсам можно отнести невысокую стоимость расходных материалов и возможность печати на бумаге практически любого качества.

Струйные принтеры (рис. 2.37) печатают путем набрызгивания чернил на бумагу через мелкие сопла в печатающей головке. В черно-белых принтерах используется один цвет краски (черный), в цветных принтерах — голубой, пурпурный и желтый. В более дорогих моделях цветных принтеров к трем базовым цветам добавляется черный, так как чисто черный цвет с помощью трех базовых цветов получить невозможно. Стоимость самих струйных принтеров обычно невысока, но зато очень высока стоимость расходных материалов (картриджей, наполненных чернилами). Кроме того, для струйного принтера нужна бумага высокого качества, иначе краска будет расплываться. Скорость печати струйных принтеров (особенно цветных) очень невысока. Но, несмотря на все недостатки, струйные принтеры широко распространены, так как они обеспечивают достаточно высокое качество печати.



Рис. 2.37. Струйный принтер

В **лазерных принтерах** изображение создается путем переноса на бумагу специального порошка (тонера). Источник света (лазер) освещает предварительно заряженную поверхность фотобарабана. На тех местах, куда попал свет, меняется заряд, и к ним притягивается тонер. Затем тонер за счет электростатики переносится на бумагу, после чего попадает в печку, где и закрепляется под действием высокой температуры. Качество такого изображения очень высокое. Так как лазерные принтеры формируют изображение постранично, а не построчно (как матричные и струйные принтеры), то и скорость их работы достаточно высока. Скорость работы лазерных принтеров измеряется в страницах в минуту. Современные принтеры обеспечивают скорость печати 20–40 страниц в минуту.

Лазерные принтеры могут быть монохромными или цветными, однако цветные принтеры очень дороги. Основным недостатком лазерных принтеров является высокая стоимость. К достоинствам (помимо высокого качества печати и скорости) можно отнес-

ти низкий уровень шума, долговечность полученных отпечатков, невысокую стоимость расходных материалов. Картриджа для лазерного принтера хватает на достаточно большое количество отпечатков.

Графопостроитель

Графопостроитель (плоттер) — устройство для вывода данных в графической форме на бумагу, пластик, фоточувствительный материал или иной носитель путем черчения.

Устройства ввода и вывода звуковой информации

Микрофон — устройство для *ввода* звуковой информации в компьютер. Микрофон преобразовывает звуковые колебания в колебания электрического тока.

Компьютерные колонки и наушники — устройства для *вывода* оцифрованного звука.

Компьютерные колонки (динамики) бывают разного качества: от недорогих пластиковых до дорогих стереосистем с высококачественным звуком. Усилитель в компьютерных колонках встроен прямо в них и не нуждается в отдельном подключении. Часто применяется система из нескольких (двух, четырех или пяти) колонок с сабвуфером, который усиливает звучание низких частот, плохо воспринимаемых человеческим ухом.



Система основных понятий

Устройства ввода и вывода информации					
Устройства ввода предназначены для передачи информации от пользователя к компьютеру					
Основные устройства ввода информации					
Клавиатура	Мышь	Сканер	Джойстик	Микрофон	Веб-камера
Устройства вывода предназначены для передачи информации от компьютера к пользователю					
Основные устройства вывода информации					
Монитор	Принтер	Плоттер	Колонки	Наушники	



Вопросы и задания

1. Какие устройства можно использовать для ввода в память компьютера графической информации?
2. Что означает эргономичность клавиатуры?

3. Возможен ли ввод текстовой информации с помощью мыши? Обоснуйте свой ответ.
4. Где вы встречались с использованием веб-камер?
5. Каких три базовых цвета используются при выводе изображения на экран монитора? Как формируются все остальные цвета спектра?
6. Назовите основные характеристики любого монитора.
7. В чем преимущества мониторов с электронно-лучевой трубкой по сравнению с LCD-мониторами?
8. Можно ли изменить стандартное разрешение LCD-монитора? Если можно, то как это повлияет на качество изображения?
9. В чем заключаются недостатки струйных принтеров?
10. Какие базовые цвета используются для печати цветного изображения?
11. Какой тип принтеров удобнее использовать при необходимости печати большого количества документов? Почему?

2.6. Программное обеспечение ПК

2.6.1. Виды программного обеспечения

Современный компьютер представляет собой единую систему, в которую входят аппаратное обеспечение (*hardware*) и программное обеспечение (*software*).

Программное обеспечение (ПО) — это совокупность программ, используемых при работе на компьютере и обеспечивающих функционирование его аппаратных средств, выполнение различных задач пользователя, а также разработку и отладку новых программ.



Классификация программного обеспечения

Существует несколько вариантов классификации программного обеспечения по типам. Чаще всего можно встретить следующий вариант.

1. **Системное ПО** — комплекс программ, обеспечивающих работу компьютера и компьютерных сетей в целом. Основу системного ПО составляют операционные системы.
2. **Прикладное ПО** — программы, предназначенные для решения определенного круга задач в различных областях человеческой деятельности (офисные пакеты, бухгалтерские программы, графические пакеты, системы управления базами данных и т. д.).

3. **Инструментальное ПО** — средства для разработки и отладки программ. Все системное и прикладное программное обеспечение создается программистами с помощью инструментального ПО.

Системное программное обеспечение

Системное программное обеспечение, в свою очередь, можно разделить на две части:

- 1) базовое ПО;
- 2) сервисные программы (утилиты).

В состав **базового ПО** входят операционные системы, сетевые программные средства, а также драйверы устройств¹⁾. Без базового программного обеспечения работа компьютера и компьютерных сетей в принципе невозможна.

Сервисные программы (утилиты) — это вспомогательные компьютерные программы, расширяющие возможности операционной системы. К ним относятся файловые менеджеры, архиваторы, антивирусные программы, средства диагностики системы и некоторые другие.

Файловые менеджеры — это программы, обеспечивающие удобный и наглядный интерфейс для работы с файлами. Любая операционная система содержит в своем составе некоторый файловый менеджер (например, Проводник для Windows). Однако некоторые пользователи предпочитают использовать для работы с файлами программы от сторонних производителей. К числу наиболее популярных файловых менеджеров в ОС Windows относятся Windows Commander (рис. 2.38), Far Manager, Total Commander. Для ОС Linux используются такие файловые менеджеры, как BeeSoft Commander, Midnight Commander, Gentoo, Nautilus и др.

Архиваторы — это программы, позволяющие с помощью специальных алгоритмов сжатия данных уменьшить размер файлов и выполнить упаковку одного или нескольких файлов в один архив. Существуют различные алгоритмы сжатия, в соответствии с которыми создаются архивные файлы различных форматов (ZIP, RAR, ARJ и т. д.). Обычно одна программа архивации позволяет использовать несколько различных форматов. Для операционной системы Windows самыми популярными являются архиваторы WinRAR, WinZip, 7-Zip. В ОС Linux наиболее часто применяется совокупность двух программ: tar (которая объединяет несколько файлов в один) и gzip (которая выполняет непосредственно сжатие).

¹⁾ Более подробно речь об операционных системах будет идти в параграфе 2.6.2, а о сетевых программных средствах — в главе 4.

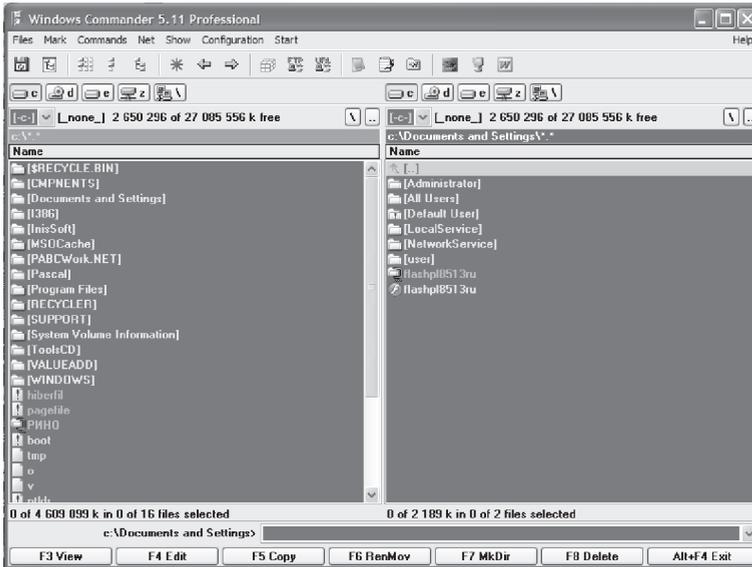


Рис. 2.38. Файловый менеджер Windows Commander

Антивирусные программы — это программы для обнаружения вредоносных программ и лечения зараженных файлов. Вредоносное программное обеспечение — это класс программ, предназначенных для выполнения различных несанкционированных пользователем действий, чаще всего направленных на причинение ущерба (уничтожение или повреждение данных, замедление работы компьютерной системы, перехват секретной информации и т. д.). К вредоносным программам относятся компьютерные вирусы, троянские программы, сетевые черви, шпионские программы (spyware), программы для навязывания рекламы (adware), перехватчики клавиатуры (keylogger) и другие. Многие антивирусные программы способны не только вылечить зараженные файлы, но и предотвратить заражение компьютера. Для этого в памяти компьютера должна постоянно находиться программа, которая контролирует любые действия с файлами. Она называется *монитором*. Современные антивирусы способны также проверять на наличие вредоносного ПО не только файлы, но и электронную почту, а также страницы, загружаемые из Интернета. К наиболее популярным антивирусным программам относятся Антивирус Касперского, NOD32, Dr.Web, Panda Antivirus.

К **средствам диагностики системы** относятся программы скапирования и дефрагментации жестких дисков, программы сбора сведений о системе и др.

Прикладное программное обеспечение

Прикладные программы (приложения) дают пользователю возможность решать разнообразные информационные задачи, не прибегая к программированию. Невозможно представить себе пользователя современного компьютера, который бы ни разу не применял для подготовки документов текстовые редакторы, не играл бы в компьютерные игры или не пользовался различными медиаплеерами для просмотра видеофильмов или прослушивания музыки. Все перечисленные выше программы относятся к прикладному ПО.

Многие прикладные программы входят в состав **пакетов прикладных программ (ППП)**, которые позволяют решать целый комплекс взаимосвязанных задач для конкретной предметной области.

Прикладное программное обеспечение можно условно разделить на две группы:

- 1) прикладные программы общего назначения;
- 2) целевое прикладное программное обеспечение.

Прикладные программы общего назначения предназначены для широкого круга пользователей и применяются одновременно во многих областях человеческой деятельности. К данному типу прикладного ПО относятся:

- текстовые процессоры;
- графические пакеты;
- системы управления базами данных;
- средства для создания презентаций;
- браузеры;
- программы для работы с электронной почтой;
- органайзеры;
- компьютерные игры и т. д.

Очень часто несколько программных продуктов, функционально дополняющих друг друга, объединяются в **интегрированный пакет прикладных программ**. В качестве примеров можно привести пакеты прикладных программ Microsoft Office и OpenOffice (рис. 2.39), включающие в своем составе текстовый процессор, табличный процессор, систему управления базами данных, средство для создания презентаций и некоторые другие программы. Пользователи интегрированных пакетов получают однотипный интерфейс для работы с различными частями пакета, общие сервисы (словарь и средства проверки правописания, построитель диаграмм, конвертер данных и т. д.), а также возможность создания документов, интегрирующих в себе возможности различных программ, входящих в состав интегрированного пакета.

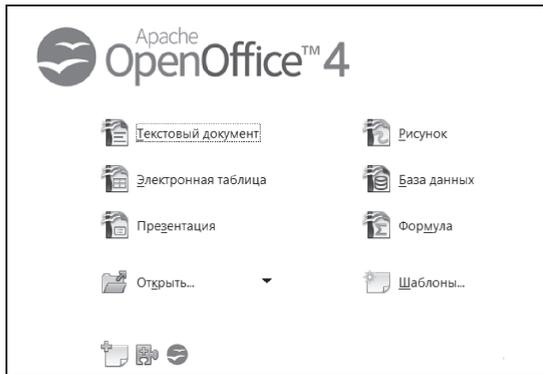


Рис. 2.39. Интегрированный пакет прикладных программ OpenOffice

К целевому прикладному программному обеспечению относятся пакеты, учитывающие специфику поддерживаемых задач пользователя и имеющие более узкое применение. К ним относятся:

- пакеты программ для проведения научных и статистических расчетов;
- программы для автоматизации бухгалтерского учета;
- настольные издательские системы;
- системы автоматизированного проектирования;
- системы управления персоналом и др.

Инструментальное программное обеспечение

Инструментальное программное обеспечение включает средства для разработки программ и предназначено для программистов. Все типы программного обеспечения (включая и инструментальное) создается с помощью данного типа ПО.

Программы для первых ЭВМ писались на языке машинных кодов, что было очень трудоемким и сложным для человека процессом. Поэтому еще в 50-х годах прошлого века появились первые языки программирования высокого уровня (Фортран, Кобол), позволяющие упростить труд программиста. Позже появилось огромное количество других языков программирования (более 8 тысяч). К наиболее известным языкам программирования относятся Basic, Pascal, C++, C#, Fortran, JAVA, PHP, Perl. Для того чтобы компьютер мог понимать программы, написанные на этих языках программирования высокого уровня, для каждого из них используются программы — трансляторы с этих языков.

Транслятор — это программа, которая преобразует программу, написанную на одном из языков высокого уровня, в программу, состоящую из машинных команд.



Трансляторы делят на компиляторы и интерпретаторы. **Компиляторы** транслируют всю программу, но без ее выполнения. **Интерпретаторы**, в отличие от компиляторов, выполняют пооператорную обработку и выполнение программы.

Для удобства работы программистов, помимо трансляторов, разрабатывается целый комплекс программных средств, объединяемых в **системы программирования**. Система программирования может включать:

- транслятор;
- интегрированную среду разработки;
- текстовый редактор;
- отладчик;
- набор библиотек (возможно, с исходными текстами программ);
- справочные системы и т. п.

Очень часто для одного и того же языка программирования разрабатывается несколько различных систем программирования, отличающихся друг от друга версиями языка, трансляторами, интерфейсом работы, а также отдельными компонентами, входящими в состав системы. Например, для языка Паскаль существуют такие системы программирования, как Borland Pascal, Turbo Pascal, Free Pascal, PascalABC (рис. 2.40) и некоторые другие.

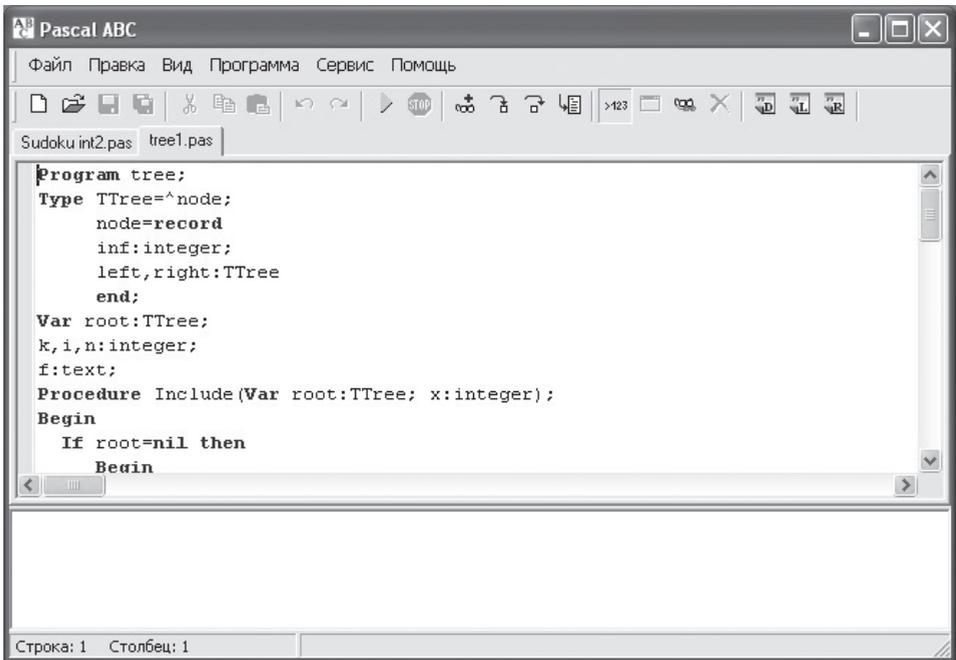


Рис. 2.40. Система программирования PascalABC



О профессиях: системный администратор

В наше время практически невозможно представить себе офис, фирму или любое другое учреждение, в котором рабочие места сотрудников не были бы оснащены персональными компьютерами. Это позволяет, с одной стороны, значительно ускорить выполнение сотрудниками своих должностных обязанностей, но, с другой стороны, часто ставит ряд достаточно серьезных проблем. Появляется большая зависимость от компьютерной техники, которая не застрахована от поломок и различных сбоев как в аппаратной части, так и в программном обеспечении. Для решения всех проблем, связанных с неполадками и нестандартными ситуациями, требуется помощь профессионала. Этим профессионалом является системный администратор.

Системный администратор — сотрудник, в обязанности которого входит создание оптимальной работоспособности компьютеров и программного обеспечения для пользователей.

В круг типовых задач системного администратора обычно входят:

- установка и конфигурирование нового аппаратного и программного обеспечения;
- устранение неполадок в системе;
- создание пользовательских учетных записей;
- установка необходимых обновлений для операционной системы и используемых программ;
- обеспечение информационной безопасности в компании;
- подготовка и сохранение резервных копий данных;
- поддержка работоспособности локальной сети;
- консультирование пользователей и т. п.

В различных учреждениях круг обязанностей системного администратора может очень сильно различаться. В небольших организациях один системный администратор отвечает за все: закупку оборудования, установку программного обеспечения, мелкий ремонт оборудования, настройку и поддержку локальной сети, консультирование пользователей, информационную безопасность и т. д. В крупных организациях обычно работает сразу несколько системных администраторов, у каждого из которых свой участок работы.

Системных администраторов можно разделить на несколько категорий.

- **Системный администратор малой компании** (от 5 до 50 рабочих мест) — занимается поддержанием работоспособности небольшой сети.
- **Администратор сети** — занимается разработкой и обслуживанием локальных сетей.
- **Системный инженер** — занимается установкой аппаратной инфраструктуры (серверов и систем хранения данных), операционных систем, а также поддержкой их функционирования.

- **Администратор безопасности сети** — занимается проблемами информационной безопасности.

Приведенный выше список можно расширить, включив в него администраторов почтовых серверов, баз данных, веб-серверов, консультантов пользователей и т. д.

Работа системного администратора требует:

- технического склада ума;
- логического мышления;
- высокой степени ответственности;
- умения самостоятельно решать сложные технические вопросы;
- постоянного повышения своей квалификации;
- стрессоустойчивости.



Система основных понятий

Классификация ПО					
Программное обеспечение (ПО) — это совокупность программ, используемых при работе на компьютере и обеспечивающих функционирование его аппаратных средств, выполнение различных задач пользователя, а также разработку и отладку новых программ					
Типы программного обеспечения					
Системное ПО		Прикладное ПО		Инструментальное ПО	
Типы системного ПО					
Базовое ПО		Утилиты			
Операционные системы (ОС)	Драйверы устройств	Файловые менеджеры	Архиваторы	Антивирусные программы	Средства диагностики
Сервисные программы (утилиты) — это вспомогательные компьютерные программы, расширяющие возможности операционной системы					
Типы прикладного ПО					
Программное обеспечение общего назначения			Целевое программное обеспечение		
Пакеты прикладных программ (ППП) — это комплекс взаимосвязанных программ для решения задач конкретной предметной области					
Интегрированные пакеты прикладных программ — это набор нескольких программных продуктов, функционально дополняющих друг друга					
Инструментальное ПО					
Инструментальное программное обеспечение включает средства для разработки программ и предназначено для программистов					
Транслятор — это программа, которая преобразует программу, написанную на одном из языков высокого уровня, в программу, состоящую из машинных команд					

Вопросы и задания

1. Что такое программное обеспечение?
2. Что входит в состав базового программного обеспечения?
3. Для чего используются файловые менеджеры?
4. Какую функцию выполняют архиваторы?
5. Перечислите список прикладного ПО, установленного на вашем компьютере.
6. В чем особенности интегрированных пакетов программ? Приведите примеры таких пакетов и перечислите их компоненты.
7. Какие программы относятся к целевому программному ПО? Приведите примеры.
8. К какому типу ПО относятся приведенные ниже программы:
 - программа обмена сообщениями;
 - операционная система Linux;
 - архиватор;
 - обучающая программа;
 - антивирусная программа;
 - транслятор с языка Паскаль;
 - программа-переводчик?
9. Почему необходимо использование трансляторов?
10. Что может входить в состав системы программирования?

Практикум. Раздел 7 «Программное обеспечение»

2.6.2. Функции операционной системы

Операционная система — это комплекс программ, обеспечивающих управление заданиями и распределение ресурсов компьютера, предоставляющих интерфейс для работы с пользователем, а также организующих работу с файлами.

Рассмотрим подробнее основные функции операционной системы.

Управление заданиями и распределение ресурсов

ЭВМ первого поколения не имели операционной системы как таковой. Поэтому общение между пользователем (точнее говоря, программистом) и компьютером велось на языке машинных кодов. Труд программиста облегчали лишь различные библиотеки наиболее часто используемых подпрограмм, а также некоторые служебные программы, которые упрощали работу оператора с оборудованием.

В 1950-х годах появляются первые алгоритмические языки и трансляторы к ним. Для выполнения программы требовалось исполнение целого ряда управляющих действий со стороны оператора ЭВМ: запуск транслятора, ввод программы в оперативную память, загрузка библиотечных подпрограмм, запуск оттранслированной программы на исполнение, включение устройства вывода на печать и пр. Выполнение всех этих действий оператором вручную существенно снижало производительность ЭВМ. Поэтому для более эффективного использования достаточно дорогого машинного времени были разработаны первые системы *пакетной обработки*, которые автоматизировали работу оператора по *управлению отдельными заданиями*. Они, по сути дела, стали предшественниками современных операционных систем. Первой такой системой считается General Motors Operating System (монитор обработки данных в пакетном режиме), разработанная в 1955 году.

В конце 1960-х годов происходит переход к поколению компьютеров, работающих на интегральных микросхемах, что позволило создавать машины с гораздо более сложной архитектурой. Одним из наиболее значительных достижений того времени стала реализация возможности одновременной загрузки в память компьютера нескольких программ и их попеременного выполнения одним процессором (мультипрограммирование). В результате появились первые **мультипрограммные системы** пакетной обработки, которые распределяли процессорное время и память между несколькими заданиями. Недостатком таких систем стало отсутствие для программиста возможности взаимодействия со своей программой во время ее исполнения, поскольку все действия по запуску пакета заданий осуществляли операторы ЭВМ.

Эта проблема была решена благодаря появлению таких устройств ввода/вывода, как клавиатуры и терминалы с электронно-лучевыми дисплеями, что привело к разработке мультипрограммных **систем разделения времени**. В таких системах один центральный процессор и блок оперативной памяти взаимодействуют с множеством терминалов, за каждым из которых работает отдельный пользователь. Каждому пользователю выделялся определенный интервал процессорного времени и некоторый объем памяти на выполнение его задания. Если выделенного времени не хватает для завершения задания, то происходит прерывание выполнения задания, а затем, когда это задание снова получит свою порцию процессорного времени, его исполнение продолжается с того состояния, в котором оно было прервано. В числе первых мультипрограммных многопользовательских операционных систем разделения времени, разработанных в середине 1960-х годов, были

TSS/360 (IBM), CTSS (Массачусетский технологический институт) и MULTICS (компании American Telegraph&Telephone Bell Labs).

В начале 1970-х годов появляются сетевые операционные системы, которые предназначены для организации распределенного хранения и обработки данных между несколькими компьютерами, соединенными каналами связи.

Основными ресурсами современных компьютеров являются процессорное время, оперативная память, дисковая память, устройства ввода/вывода данных. Распределение ресурсов компьютера происходит между *процессами*.

Процесс — это объект, который возникает в операционной системе после того, как пользователь или сама система запустили программу на выполнение. При этом одна и та же программа может породить несколько процессов (например, если на одном компьютере какая-либо программа будет запущена от имени нескольких разных пользователей).

Управление ресурсами включает решение следующих задач:

- определение, какому процессу, когда и в каком объеме следует выделить данный ресурс;
- отслеживание состояния и учет использования ресурса;
- разрешение конфликтов между процессами.

От эффективности алгоритмов управления ресурсами компьютера во многом зависит производительность всей компьютерной системы в целом. В зависимости от особенностей управления ресурсами операционные системы делят на следующие типы:

- *многозадачные* (допускается одновременное выполнение нескольких пользовательских программ) и *однозадачные* (допускается выполнение только одной пользовательской программы);
- *многопользовательские* (допускается одновременный запуск программ от имени нескольких пользователей) и *однопользовательские* (допускается запуск программ только от имени одного пользователя);
- *многопроцессорные* (допускается использование на компьютерах с несколькими процессорами) и *однопроцессорные* (допускается использование на компьютерах только с одним процессором).

Предоставление интерфейса работы для пользователя

Чтобы успешно решать свои задачи, пользователь не обязан знать подробности аппаратного устройства компьютера. Операционная система скрывает от него большую часть особенностей архитектуры и предоставляет возможность простой и удобной работы с файлами и аппаратными средствами (принтерами, сканерами,

дисками и т. д.). Если бы пользователь работал с компьютером без участия ОС, то, например, для организации чтения блока данных с диска ему пришлось бы использовать более десятка команд с указанием множества параметров: номера блока на диске, номера сектора на дорожке и т. д.

В *пакетных операционных системах* (речь о которых шла выше) общение с ОС осуществлялось оператором, а сам программист был лишен возможности ведения диалога. В системах разделения времени появился **командный интерфейс** работы пользователя, и общение между пользователем и компьютером велось посредством набора специальных команд. Каждая операционная система имела свой подобный набор команд. Примерами ОС с командным интерфейсом работы являются CP/M, UNIX, MS-DOS (рис. 2.41).

```

13.08.2009 23:29 <DIR> tmp
27.12.2008 20:19 <DIR> Unreal Commander
29.07.2009 00:18 <DIR> WebServers
15.11.2008 01:10 <DIR> wincmd
09.10.2009 22:28 <DIR> WINDOWS
13.04.2009 10:43 <DIR> work
          3 файлов          32 байт
          11 папок 70 930 042 880 байт свободно

C:\>cd tmp

C:\tmp>dir
Том в устройстве C не имеет метки.
Серийный номер тома: 180C-B498

Содержимое папки C:\tmp
13.08.2009 23:29 <DIR> -
13.08.2009 23:29 <DIR> -
          0 файлов          0 байт
          2 папок 70 930 042 880 байт свободно

C:\tmp>cd ..

C:\>_

```

Рис. 2.41. Командный интерфейс ОС MS-DOS

Для удобства работы пользователей в некоторых ОС были разработаны специальные **программы-оболочки**, позволяющие работать с файлами с помощью меню. Это избавляло от необходимости вводить команды с клавиатуры. Примером такой оболочки является программа Norton Commander, разработанная для ОС MS-DOS.

В начале 1980-х годов появляются первые цветные графические мониторы, что послужило толчком для создания **операционных систем с графическим интерфейсом** работы для пользователя. В 1984 году для компьютеров Macintosh фирмы Apple была разработана первая операционная система с графическим интерфейсом MacOS (рис. 2.42). Пользователи получили возможность управлять своим компьютером с помощью нового устройства, названного *мышью*. Операционная система MacOS использовала



Рис. 2.42. Графический интерфейс операционной системы MacOS

оконный интерфейс для представления и организации информации.

Графический интерфейс работы имеют многие современные операционные системы (Windows, Linux, MacOS), используемые на персональных компьютерах.

Организация работы с файлами

Переход от носителей данных с последовательным доступом (перфолент, перфокарт и магнитных лент) к накопителям с произвольным доступом — магнитным дискам привел к необходимости разработки системных программных средств для организации хранения и поиска данных на таких носителях. В состав ОС были включены *файловые системы*.

Файловая система определяет способ организации данных на внешнем носителе; выполняет преобразование символьных имен файлов, с которыми работает пользователь, в физические адреса данных на диске; организует совместный доступ к файлам; защищает их от несанкционированного доступа и т. д.

Как известно, информация на диске хранится в секторах и само устройство может выполнять лишь команды «считать/записать информацию в определенный сектор на диске». Файловая система предоставляет пользователю работать с более удобным для него понятием «**файл**». Для идентификации файлов используются имена. Файловые системы обычно имеют иерархическую структуру, в которой файлы организуются в каталоги нескольких уровней (рис. 2.43).

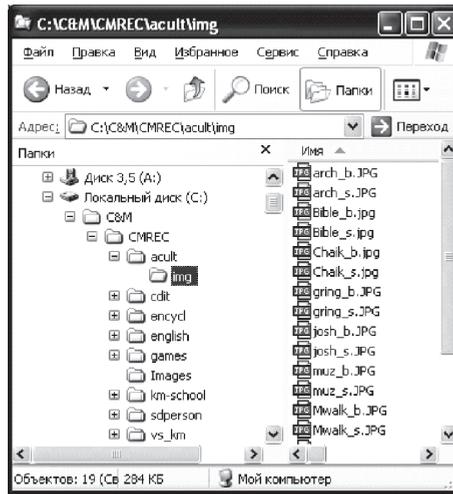


Рис. 2.43. Иерархическая файловая структура

Различие между файловыми системами заключается в основном в способах распределения между файлами пространства на диске и организации на диске служебных областей.

Файловая система состоит из следующих компонентов:

- совокупности всех файлов на диске;
- наборов служебных структур данных, используемых для управления файлами (например, каталогов файлов);
- комплекса системных программных средств, реализующих управление файлами.

Современные операционные системы стремятся обычно при установке выбирать одну из нескольких файловых систем. В ОС Windows используются файловые системы FAT32 и NTFS, в ОС Linux применяется достаточно большое количество файловых систем, из которых наиболее распространены ext2/ext3.

Так как файловая система активно использует устройства ввода/вывода, она должна поддерживать работу с *драйверами* устройств.



Драйвер — это программа управления работой внешнего устройства компьютера.

Драйверы бывают стандартными и загружаемыми. *Стандартные драйверы* управляют работой стандартных устройств (монитор, клавиатура, диски, принтеры), записываются в ПЗУ и образуют в совокупности «базовую систему ввода/вывода» — BIOS. *Загружаемые драйверы* используются для управления

дополнительными внешними устройствами ПК. Часть загружаемых драйверов входит в состав операционной системы, часть драйверов необходимо устанавливать дополнительно с дисков, которые обычно прилагаются в поставке с каждым устройством. При необходимости драйверы можно скачать из Интернета с сайтов разработчиков устройств.

Система основных понятий



Операционные системы		
<p>Операционная система — это комплекс программ, обеспечивающих управление заданиями и распределение ресурсов компьютера, предоставляющих интерфейс для работы с пользователем, организующих работу с файлами</p>		
Функции операционной системы		
<p><i>Управление заданиями и распределение ресурсов</i> Задачи:</p> <ul style="list-style-type: none"> определение, какому процессу, когда и в каком объеме следует выделить данный ресурс; отслеживание состояния и учет использования ресурса; разрешение конфликтов между процессами 	<p><i>Предоставление интерфейса работы пользователю</i> Типы интерфейса:</p> <ul style="list-style-type: none"> командный; графический 	<p><i>Организация работы с файлами (файловая система)</i> Компоненты:</p> <ul style="list-style-type: none"> совокупность всех файлов на диске; наборы служебных структур данных, используемых для управления файлами; комплекс системных программных средств, реализующих управление файлами
<p>Процесс — это объект, который возникает в операционной системе после того, как пользователь или сама система запустили программу на выполнение</p>		
<p>Файловая система определяет способ организации данных на внешнем носителе; выполняет преобразование символьных имен файлов, с которыми работает пользователь, в физические адреса данных на диске; организует совместный доступ к файлам; защищает их от несанкционированного доступа и т. д.</p>		
<p>Драйвер — это программа управления работой внешних устройств компьютера</p>		
<p><i>Стандартные драйверы</i> управляют работой стандартных устройств, записываются в ПЗУ и образуют в совокупности «базовую систему ввода/вывода», называемую BIOS</p>	<p><i>Загружаемые драйверы</i> используются для управления дополнительными внешними устройствами ПК. Часть загружаемых драйверов входит в состав операционной системы</p>	



Вопросы и задания

1. Что такое операционная система? В чем необходимость ее использования?
2. Какие действия позволяли автоматизировать системы пакетной обработки? С чем связано появление этих систем?
3. Какие типы операционных систем появились в конце 1960-х годов?
4. Что такое многозадачная операционная система?
5. Почему возникла потребность в разработке файловых систем?
6. В чем суть работы операционных систем разделения времени?
7. Что такое процесс? В чем его отличие от программы?
8. Что понимается под понятием «ресурс компьютера»?
9. Перечислите основные задачи ОС по управлению ресурсами.
10. Чем был вызван переход к ОС с графическим интерфейсом работы?
11. За что отвечает файловая система?
12. Перечислите компоненты файловой системы.
13. Как называется программа, управляющая работой отдельного внешнего устройства компьютера?
14. В чем различие стандартных и загружаемых драйверов?



Практикум. Раздел 7 «Программное обеспечение ПК»

2.6.3. Операционные системы для ПК

Понятие о разрядности операционной системы

Помимо рассмотренных в предыдущем параграфе типов ОС (многозадачные и однозадачные, многопользовательские и однопользовательские и т. п.), существует также деление операционных систем на 8-, 16-, 32- и 64-разрядные.

Разрядность ОС — это количество битов в адресе оперативной памяти, к которому она может напрямую обратиться. От разрядности ОС зависит, например, максимальная длина имени файла, количество вложенных папок и ряд других характеристик. Разрядность операционной системы говорит также о том, с какими прикладными программами она сможет работать (например, 16-разрядная ОС не сможет работать с 32-разрядными приложениями). Разрядность операционной системы не может превышать разрядности процессора.

История развития ОС для персональных компьютеров

В середине 1970-х годов появились первые персональные компьютеры, которые нуждались в собственной операционной системе. Первой ОС для персональных ЭВМ стала CP/M, разработанная в 1974 году. Она была установлена на многие 8-разрядные компьютеры (в том числе на Apple-1). Размер ее программного кода составлял всего 4 килобайта.

В начале 1980-х годов в мире начинается массовое использование персональных компьютеров, что потребовало разработки для них новых операционных систем. Первой популярной ОС стала 16-разрядная операционная система MS-DOS, разработанная компанией Microsoft. Это была однозадачная однопользовательская ОС с командным интерфейсом работы. Запуск MS-DOS мог осуществляться как с жесткого диска, так и с дискеты. Разработчики первых ОС для персональных компьютеров считали, что при индивидуальном использовании компьютера нет смысла в поддержке мультипрограммирования и многопользовательского режима работы. Для удобства работы с файлами для MS-DOS была разработана специальная оболочка Norton Commander (речь о ней шла в предыдущем параграфе).

В 1984 году для компьютеров Macintosh фирмы Apple была разработана первая операционная система с графическим интерфейсом MacOS (см. предыдущий параграф).

В 1985 году фирма Microsoft разработала первую версию многозадачной операционной системы с графическим интерфейсом Windows 1.0 (рис. 2.44). Но настоящий переворот в сфере операционных систем был совершен в 1990 году, когда вышла версия Windows 3.0. Для обмена данными между приложениями использовался буфер обмена, был реализован полноэкранный

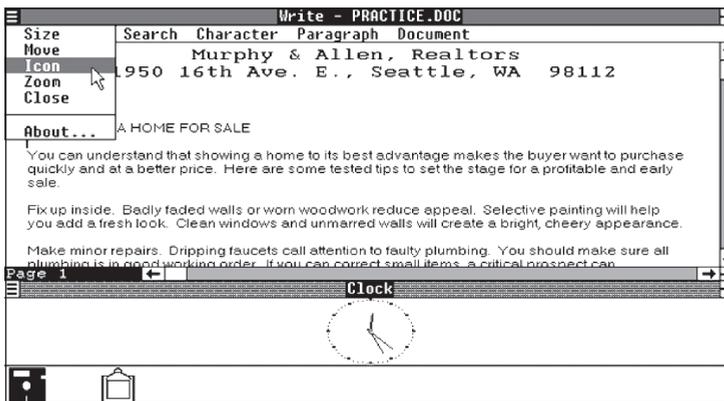


Рис. 2.44. Интерфейс ОС Windows 1.0

режим работы для DOS-приложений, появилось понятие виртуальной памяти, что позволило запускать приложения, размер которых превышал размер оперативной памяти. В Windows 3.1 была поддержка шрифтов True Type, а также технология Drag&Drop (перенос мышью файлов и директорий). Однако все эти версии Windows нельзя было назвать полноценными операционными системами, так как их графический интерфейс был интегрирован «поверх» операционной системы MS-DOS. Но стоит заметить, что, несмотря на это, исполняемые файлы Windows имели другой формат (т. е. приложения, написанные для Windows, нельзя было запускать из MS-DOS). Помимо этого, Windows имела собственную систему распределения памяти, а также в ее состав был включен ряд новых драйверов.

В апреле 1987 года компании IBM и Microsoft объявили о совместных планах по созданию новой операционной системы OS/2 (читается как «ОС пополам»). Однако спустя несколько лет фирма Microsoft полностью переключилась на Windows. OS/2, в отличие от первых версий Windows, была полноценной многозадачной операционной системой с графическим интерфейсом (рис. 2.45). В 1993 году появилась первая 32-разрядная операционная система OS/2 2.1. Она могла выполнять приложения, созданные для ОС Windows, и содержала большое количество драйверов для периферийных устройств.

В 1995 году выходит *полноценная 32-разрядная операционная система Windows'95*, интерфейс которой существенно отличался

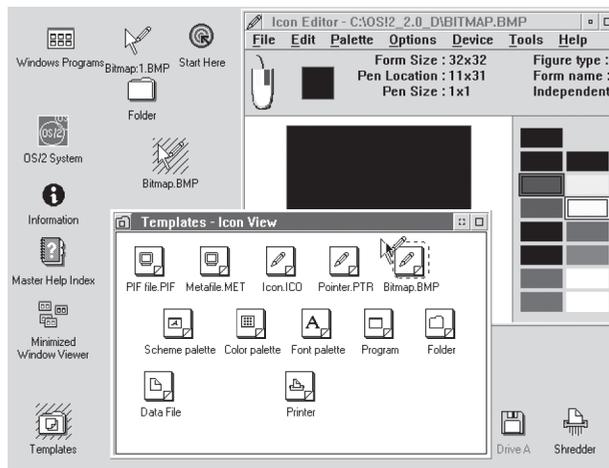


Рис. 2.45. Интерфейс OS/2 2.0

от предыдущих версий Windows. Появились такие объекты, как Панель задач, Рабочий стол и Главное меню. Далее все последующие операционные системы семейства Windows (Windows'98, Windows 2000, Windows XP, Windows Vista и т. д.) стали использовать подобный вариант интерфейса. ОС Windows Vista стала первой 64-разрядной ОС в семействе Windows.

Важным событием в мире операционных систем стала разработка *операционной системы UNIX*. В 1969 году сотрудник Bell Labs Кен Томпсон разрабатывает на языке ассемблер первую версию операционной системы UNIX для мини-ЭВМ PDP-7, создав для нее собственную файловую систему, простой командный интерпретатор, подсистему управления памятью, а также ряд дополнительных утилит. В 1973 году Деннис Ритчи специально для разработки UNIX создает новый язык программирования C (Си). Очередная версия UNIX была написана на языке C, благодаря чему стало возможным переносить UNIX на компьютеры других типов, в том числе и на ПК. В 1991 году выходит свободно распространяемая версия UNIX, разработанная Линусом Торвальдсом из Хельсинки (Финляндия). Называлась она *Linux*. В обсуждении разработки этой операционной системы принимали участие программисты со всего мира, используя группы новостей USENET. Каждый мог внести свои предложения и усовершенствования. Благодаря этому в настоящее время существует огромное количество дистрибутивов¹⁾ Linux, среди которых Mandriva Linux (рис. 2.46), Fedora Core, Knoppix, Debain и ряд других.

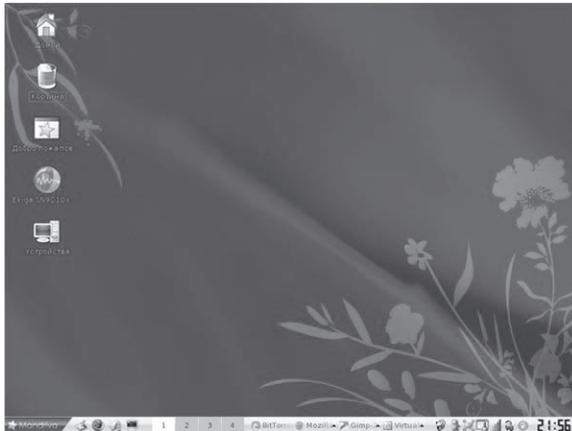


Рис. 2.46. Интерфейс Linux Mandriva

¹⁾ *Дистрибутив* — это набор файлов, приспособленный для установки программного обеспечения.

Сравнение ОС Windows и Linux

В настоящее время наиболее распространенными ОС для персональных компьютеров на базе Intel-совместимых процессоров являются операционные системы семейств Windows и Linux. Проведем краткое сравнение этих двух семейств ОС.

И Windows, и Linux являются многопользовательскими многозадачными операционными системами. В обеих ОС применяется система идентификации пользователей с возможностью разграничения прав доступа к файлам и каталогам, а также к другим ресурсам компьютера (дискам, принтерам, сканерам и т. д.). И Windows, и Linux имеют встроенные средства для работы пользователей в сети. Подавляющее большинство современных версий Linux поддерживает графический интерфейс работы пользователя, который близок к интерфейсу Windows.

Теперь о различиях этих двух операционных систем. Первое различие, которое лежит на поверхности, заключается в том, что Linux относится к числу свободно распространяемых ОС, т. е. дистрибутивы Linux можно свободно скачать из Интернета или приобрести в магазине по доступным ценам, в отличие от Windows, которая является полностью коммерческой программой.

Второе различие заключается в доступности исходных кодов дистрибутивов Linux, что дает возможность любому заинтересованному программисту внести в них свои изменения и создать на их основе свой собственный вариант дистрибутива.

Следующее различие — модульность установки Linux. В отличие от ОС Windows, которая в своем составе содержит довольно скромный набор приложений (Блокнот, Paint, Калькулятор и некоторые другие), любой дистрибутив Linux включает достаточно широкий спектр прикладного и инструментального ПО. Поэтому уже при установке данной ОС пользователь может сразу же выбрать нужный ему набор отдельных приложений или их модулей (офисные программы, мультимедиа, игры, графические оболочки и т. д.). Достаточно большое разнообразие дистрибутивов Linux (около сотни) позволяет каждому пользователю найти именно тот, который устроит его с точки зрения набора необходимых компонентов.

В отличие от Windows, которая позволяет работать только с двумя типами файловых систем (FAT и NTFS), Linux поддерживает большое количество файловых систем (в том числе и тех, которые применяются в Windows). Это дает возможность пользователю непосредственно из Linux обращаться к тем разделам

жесткого диска, которые используются другими операционными системами. Основными файловыми системами самой Linux являются ext2 и ext3. Разница между ними заключается в том, что ext3 является полностью журналируемой, т. е. в ней предусмотрена автоматическая запись данных, позволяющих восстановить файловую систему при сбоях в работе компьютера. В отличие от файловых систем, используемых в Windows, в Linux не существует понятия логического диска, а сама файловая структура имеет четкую иерархию, содержащую точно определенный набор каталогов. В частности, каталог */boot* содержит файлы, необходимые при загрузке системы, */home* — домашние директории пользователей и т. д. Причем для работы с отдельными устройствами используются так называемые «файлы-дырки», с помощью которых происходит обращение к этим устройствам. Сами устройства регистрируются в каталоге */dev*.

В Linux хорошо развиты консольный интерфейс работы и командный язык. Многие действия по администрированию операционной системы возможно выполнить только через командную строку.

В Linux существует только один администратор системы — пользователь с именем *root*. Все действия, связанные с администрированием и настройкой системы, должны выполняться только от его имени.

Для ОС Windows существует гораздо большее количество драйверов различных устройств, поэтому при работе с Linux возможна ситуация, когда устройство будет неработоспособно из-за отсутствия для него нужного драйвера даже на сайте производителя.

Во многих версиях Linux существует достаточно сложная процедура установки новых пакетов (приложений). Очень часто можно встретить ситуацию, когда один пакет не может быть установлен по причине отсутствия какого-либо другого пакета, содержащего необходимую библиотеку. В отличие от этого дистрибутивы приложений Windows содержат весь набор необходимых библиотек.

Понятие о BIOS.

Порядок загрузки компьютера

Чтобы понять порядок загрузки компьютера и операционной системы, рассмотрим подробнее, что такое BIOS. В компьютере существует особый тип системной памяти: **ПЗУ** — **постоянное запоминающее устройство**, в которой хранится программа **BIOS** —

базовая система ввода/вывода. Назначение этого небольшого (256 Кб) программного кода — поиск, тестирование и установка некоторых параметров устройств компьютера. После включения компьютера счетчик команд процессора устанавливается на начальный адрес ПЗУ и стартует выполнение программы BIOS. Таким образом, на первоначальном этапе ищется и тестируется установленное оборудование. Современные компьютеры в основном используют внешние устройства «*plug and play*» («включи и работай»), которые способны сообщить процессору свои основные характеристики и условия работы, что и происходит во время работы BIOS.

Если все оборудование функционирует нормально, происходит переход к следующему этапу — поиску главной загрузочной записи (MBR — Master Boot Record). Она может находиться на жестком диске, на компакт-диске, на устройстве флеш-памяти или быть получена с помощью сетевой платы. Компьютер опрашивает перечисленные устройства по очереди в том порядке, в котором это указано в параметрах BIOS. Эти параметры можно менять с помощью программы SETUP, вызов которой осуществляется в первые несколько секунд загрузки компьютера.

Главная загрузочная запись содержит *таблицу разделов* (Partition Table), в которой хранится информация обо всех разделах, определенных на диске. Кроме таблицы разделов диска MBR содержит небольшой объем исполняемого кода. Исполняемый код читает таблицу разделов, определяет системный раздел (раздел, где установлена ОС), находит его начало и загружает в память *загрузочный сектор раздела* (Partition Boot Sector), после чего и начинается непосредственная загрузка самой операционной системы. Главная загрузочная запись не зависит от операционной системы. Что касается загрузочного сектора раздела, то он зависит как от операционной системы, так и от применяемой на данном томе файловой системы.

Возможна ситуация, когда на компьютере установлено несколько операционных систем, т. е. существует несколько системных разделов. В этом случае пользователю будет предложено выбрать нужную ОС для загрузки из списка. В противном случае будет загружаться операционная система, заданная по умолчанию.



Система основных понятий

Операционные системы для ПК					
Разрядность ОС — это количество битов в адресе оперативной памяти, к которому она может напрямую обратиться					
Основные ОС для ПК					
CP/M 8-разрядная	MS-DOS 16-разрядная	OS/2 32-разрядная	MacOS 16-, 32-, 64-разрядная	Windows 16-, 32-, 64-разрядная	Linux 32-, 64-разрядная
Сравнительные характеристики Windows и Linux					
Характеристика	Windows		Linux		
Условия распространения	Коммерческая		Свободно распространяемая		
Доступность исходного кода	Нет		Да		
Поддерживаемые файловые системы	FAT, NTFS		Ext2, Ext3, Fat, NTFS и ряд других		
Интерфейс работы	Графический		Командный, графический		
Состав дистрибутива	ОС + небольшой набор прикладных программ		ОС + большое количество прикладных пакетов и средств программирования		
Система идентификации пользователей	Есть		Есть		
Поддержка работы в сети	Есть		Есть		
Администратор системы	Произвольное количество		Единственный с именем root		
Структура корневого каталога	Несколько стандартных каталогов + произвольное количество пользовательских папок		Жестко закрепленная		
Сложность установки дополнительных приложений	Простая		Сложная		
BIOS — это программный код, хранящийся в ПЗУ, предназначенный для поиска, тестирования и установки некоторых параметров устройств компьютера					

**Вопросы и задания**

1. Что такое разрядность ОС? На какие характеристики компьютерной системы в целом она влияет?
2. Назовите известные вам 32-разрядные операционные системы.
3. Почему первые версии Windows нельзя назвать полноценными операционными системами?
4. Перечислите недостатки ОС Linux.
5. Перечислите недостатки ОС Windows.
6. Может ли пользователь root создавать собственные каталоги в корневом каталоге Linux?
7. На компьютере установлено две ОС: Windows XP (в разделе C: жесткого диска) и Linux Mandriva (в разделе D: жесткого диска). Можно ли, загрузившись в ОС Windows, прочитать файлы с диска D? Почему?
8. Какую функцию выполняет BIOS?
9. Где может находиться MBR? Как изменить порядок ее поиска?
10. Может ли на одном компьютере быть несколько загрузочных секторов? Если да, то при каких условиях?



Практикум. Раздел 7 «Программное обеспечение ПК»



ЭОР к главе 2 на сайте ФЦИОР (<http://fcior.edu.ru>)

2.1

- Логические элементы
- Логические ИС
- Сумматор двоичных чисел

2.2–2.3

- Архитектура компьютера
- Архитектура машин пятого поколения

2.4

- Число и его компьютерный код
- Числа с фиксированной и плавающей запятой

2.5

- Процессор
- Внутренняя память компьютера. Внешняя память компьютера. Типы накопителей информации
- Устройства ввода информации
- Устройства вывода информации

2.6

- Классификация ПО
- Основные функции и состав операционной системы
- Основные элементы интерфейса и управления