

ПОПУЛЯРНЫЕ ЛЕКЦИИ ОБ УСТРОЙСТВЕ КОМПЬЮТЕРА



↓

Функциональные узлы ЭВМ
и их взаимодействие

Организация памяти
и методы доступа к ней

Обмен информацией
с внешними устройствами

Роль
программного обеспечения

↓

+ CD-ROM



→

О С Н О В Ы

И Н Ф О Р М А Т И К И

Е. А. Еремин

ПОПУЛЯРНЫЕ ЛЕКЦИИ ОБ УСТРОЙСТВЕ КОМПЬЮТЕРА

Санкт-Петербург

«БХВ-Петербург»

2003

УДК 681.3.06(075.3)
ББК 32.973я721
Е70

Еремин Е. А.

Е70 Популярные лекции об устройстве компьютера. — СПб.:
БХВ-Петербург, 2003. — 272 с.: ил.

ISBN 5-94157-313-8

Книга представляет собой тщательно систематизированное, но в то же время популярное описание фундаментальных принципов устройства ЭВМ. Понятное и подробное изложение теории сопровождается большим количеством примеров, многие из которых оригинальны и достаточно необычны. После знакомства с очередной лекцией читателям предлагается провести на своем компьютере серию любопытных экспериментов. Они не просто иллюстрируют и дополняют лекционный материал, но и позволяют убедиться в том, насколько интересным и логически совершенным является наш ставший таким привычным электронный помощник.

Для широкого круга пользователей

УДК 681.3.06(075.3)
ББК 32.973я721

Группа подготовки издания:

Главный редактор	<i>Екатерина Кондукова</i>
Зам. главного редактора	<i>Людмила Еремеевская</i>
Зав. редакцией	<i>Григорий Добин</i>
Редактор	<i>Ирина Радченко</i>
Компьютерная верстка	<i>Ольги Сергиенко</i>
Корректор	<i>Зинаида Дмитриева</i>
Дизайн обложки	<i>Игоря Цырульниковца</i>
Зав. производством	<i>Николай Тверских</i>

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 11.07.03.

Формат 70×100^{1/16}. Печать офсетная. Усл. печ. л. 21,9.

Тираж 3000 экз. Заказ №

"БХВ-Петербург", 198005, Санкт-Петербург, Измайловский пр., 29.

Гигиеническое заключение на продукцию, товар № 77.99.02.953 Д.001537.03.02
от 13.03.2002 г. выдано Департаментом ГСЭН Минздрава России.

Отпечатано с готовых диапозитивов
в Академической типографии "Наука" РАН
199034, Санкт-Петербург, 9 линия, 12.

ISBN 5-94157-313-8

© Еремин Е. А., 2003

© Оформление, издательство "БХВ-Петербург", 2003

Содержание

Предисловие	7
Введение	9
Лекция 1. Что мы будем изучать и почему	13
1.1. О возможных подходах к изучению компьютера	14
1.2. О содержании лекций.....	18
1.3. Стремительное развитие ВТ и стабильность ее фундаментальных принципов.....	20
1.3.1. Темпы развития ЭВМ. Поколения ЭВМ.....	20
1.3.2. Система команд типичной ЭВМ второго поколения	23
1.3.3. Система команд типичной ЭВМ четвертого поколения.....	25
1.3.4. Сравнение систем команд ЭВМ второго и четвертого поколений.....	27
1.4. Некоторые выводы.....	29
1.5. Вопросы для осмысления.....	30
Лекция 2. Функциональные блоки ЭВМ	32
2.1. Основные блоки ЭВМ	34
2.2. Взаимодействие блоков ЭВМ	35
2.2.1. Классическая структура ЭВМ.....	35
2.2.2. Как это на самом деле работало.....	37
2.2.3. Структура современных ЭВМ.....	39
2.2.4. Режим прямого доступа к памяти.....	43
2.3. Выводы	44
2.4. Вопросы для осмысления.....	44
2.5. Любопытные эксперименты.....	45
2.5.1. Представление чисел в машине.....	46
2.5.2. Двоичное представление чисел в памяти.....	47
2.5.3. Двоичное представление информации на диске	48
2.5.4.* Может ли машина сама формировать себе программу?	50
2.5.5. Можно ли обойтись без умножения?.....	55

Лекция 3. Процессор ЭВМ	57
3.1. Назначение процессора и его устройство.....	57
3.1.1. Процессор и микропроцессор	58
3.1.2. Арифметико-логическое устройство.....	60
3.1.3. Устройство управления.....	61
3.1.4. Программно-доступные регистры.....	62
3.1.5. Разрядность процессора	62
3.2. Как работает процессор.....	63
3.2.1. Основной алгоритм работы процессора.....	63
3.2.2. Проблема начала работы ЭВМ.....	65
3.2.3. Организация ветвлений.....	66
3.2.4. Оптимизация выполнения команд.....	67
3.2.5. Тактовая частота	68
3.3. Система команд процессора	69
3.3.1. Основные группы команд.....	69
3.3.2. Процессоры RISC- и CISC-архитектуры.....	70
3.3.3. Структура команд.....	70
3.3.4. Пример программы в командах процессора.....	71
3.4. Выводы	73
3.5. Вопросы для осмысления.....	73
3.6. Любопытные эксперименты.....	74
3.6.1. Какой процессор находится внутри компьютера?.....	74
3.6.2.* Измерение быстродействия процессора.....	77
3.6.3. Экспериментальное сравнение эффективности программ.....	85
3.6.4. Как используются логические инструкции процессора	86
 Лекция 4. Память ЭВМ: ОЗУ.....	 92
4.1. Назначение и виды памяти.....	92
4.1.1. Внутренняя и внешняя память.....	92
4.1.2. ОЗУ, ПЗУ, ППЗУ и некоторые другие виды памяти.....	93
4.1.3. Статическое и динамическое ОЗУ.....	97
4.1.4. Кэш-память.....	98
4.2. Организация внутренней памяти.....	99
4.2.1. Ячейка, слово, байт.....	99
4.2.2. О хранении многобайтовых данных	101
4.3. Адресация памяти	103
4.3.1. Адресное пространство памяти	103
4.3.2. Методы адресации данных.....	103
4.3.3. Стековая организация памяти	105
4.4. Выводы	107
4.5. Вопросы для осмысления.....	108
4.6. Любопытные эксперименты.....	108
4.6.1. Сколько в компьютере ОЗУ?.....	108
4.6.2. Как используется ОЗУ?	110
4.6.3. Порядок хранения данных в IBM PC	112
4.6.4. Оценка эффективности применения кэш-памяти	113
4.6.5.* Экспериментальное изучение триггера	114

Лекция 5. Память ЭВМ: устройства внешней памяти 119

5.1. Назначение и виды внешней памяти.....	119
5.1.1. Внешняя память на бумажных носителях	120
5.1.2. Внешняя память на магнитных носителях	121
5.1.3. Внешняя память на оптических носителях	126
5.2. Организация данных во внешней памяти	128
5.2.1. Размещение информации на носителях.....	128
5.2.2. Доступ к информации на внешних носителях.....	130
5.2.3. Файловая система.....	131
5.2.4. Роль контроллеров.....	131
5.3. Взаимодействие различных видов памяти.....	132
5.3.1. Взаимодействие внутренней и внешней памяти.....	132
5.3.2. Виртуальная память.....	133
5.3.3. Иерархия памяти	134
5.4. Выводы	135
5.5. Вопросы для осмысления.....	136
5.6. Любопытные эксперименты.....	137
5.6.1. Логический доступ к сектору дискеты	137
5.6.2.* Физический доступ к сектору дискеты	141
5.6.3. Чтение секторов CD.....	144
5.6.4. Форматирование дискеты.....	147
5.6.5. Считывание S.M.A.R.T.-параметров жесткого диска.....	148

Лекция 6. Устройства ввода/вывода 151

6.1. Назначение и виды устройств ввода/вывода.....	151
6.1.1. Устройства ввода	152
6.1.2. Устройства вывода.....	155
6.2. Организация ввода/вывода.....	158
6.2.1. Порты	158
6.2.2. Обмен по программе.....	159
6.2.3. Обмен по прерываниям.....	160
6.3. Необходимость программной настройки устройств.....	162
6.4. Об объединении компьютеров в сеть.....	163
6.5. Выводы	164
6.6. Вопросы для осмысления.....	165
6.7. Любопытные эксперименты.....	166
6.7.1. Как отображаются вводимые символы?.....	166
6.7.2.* Реализация печати символов на низком уровне	168
6.7.3. Прямая запись в видеопамять	172
6.7.4. Изучение событий, связанных с мышью.....	174
6.7.5. Поиск пикселей на мониторе.....	177

Лекция 7. Роль программного обеспечения 179

7.1. Компьютер — единство аппаратной и программной частей.....	179
7.2. Типы программного обеспечения	184
7.2.1. Системное ПО	184
7.2.2. Прикладное ПО	184
7.2.3. Системы программирования.....	185

7.3. "Слой" программного обеспечения	185
7.4. Главная программа — операционная система.....	187
7.5. Порядок загрузки ПО	192
7.5.1. Тестирование оборудования.....	192
7.5.2. Чтение загрузочного сектора	193
7.5.3. Чтение начального загрузчика ОС.....	194
7.5.4. Загрузка операционной системы.....	194
7.5.5. Запуск остального ПО	195
7.6. Какое ПО устанавливать на компьютер	195
7.7. Еще раз о роли программирования.....	197
7.8. Выводы	198
7.9. Вопросы для осмысления.....	199
7.10. Любопытные эксперименты.....	200
7.10.1. Какие программы установлены на компьютере?.....	200
7.10.2. Определение размера кластера	203
7.10.3. Несколько экспериментов с именами файлов.....	204
7.10.4.* Изучение расположения файлов на дискете.....	206
7.10.5. Слои программного обеспечения	214
7.10.6. Проверка антивирусного ПО.....	216

Лекция 8. Учебные модели компьютера 220

8.1. Реальный компьютер или модель?	220
8.2. Обзор существующих учебных моделей.....	222
8.2.1. Модели ЭВМ в учебниках информатики	222
8.2.2. Учебный компьютер Е97	225
8.2.3. Зарубежные модели.....	228
8.2.4. MMIX Д. Кнута — RISC-процессор тысячелетия.....	230
8.3. Сравнение учебных моделей	234
8.4. Выводы	236
8.5. Вопросы для осмысления.....	236
8.6. Любопытные эксперименты.....	237
8.6.1. Вычисления на трехадресной ЭВМ	237
8.6.2. Вычисления на двухадресной ЭВМ	245
8.6.3. Вычисления на одноадресной ЭВМ	249
8.6.4. Как ЭВМ принимает решения	253
8.6.5. Как работает язык высокого уровня.....	259

Приложение. Содержание CD-диска..... 264

Список литературы 266

Предметный указатель..... 271

Предисловие

Данный курс лекций был первоначально задуман и разработан для педагогической газеты "Информатика", где имел название "Основы вычислительной техники" [29]. Главной идеей публикации было поддержать вытесняемую из школьного предмета "Основы информатики и вычислительной техники" вторую его составляющую, т. е. основы вычислительной техники. Для этой цели был тщательно подобран, проанализирован и систематически изложен материал по наиболее важным закономерностям устройства ЭВМ. Известно, что сведения по указанной теме обязательно есть практически в каждом учебнике информатики, но они невелики по объему и, чаще всего, поверхностны. Более того, нередки случаи, когда в погоне за упрощением или повышением наглядности изложения некоторые авторы используют описания, не вполне соответствующие фактическому устройству вычислительной техники.

Судя по отзывам, попытка отбора и обобщения материала оказалась удачной. Поэтому возникла идея сделать содержание лекций доступным более широкой аудитории читателей путем издания настоящей книги.

По сравнению с газетной версией, книга не просто поменяла название. В текст был внесен целый ряд изменений и дополнений, поскольку появилась возможность существенно увеличить объем излагаемого материала и рассмотреть большее число примеров. Наиболее существенным расширением стали появившиеся описания многочисленных экспериментов, которые иллюстрируют и дополняют теоретический материал. Проверая изложенные в лекциях принципы, читатели смогут на собственном опыте убедиться, например, в справедливости упомянутого в каждом учебнике принципа двойного представления и хранения информации, или узнать, сколько памяти содержит компьютер, на котором они работают, и достаточно ли ее для установленного программного обеспечения.

Отметим, что материалы лекций трактуются не только как форма проведения учебных занятий, а в более широком смысле. Известно, что в России

давно существовало такое явление, как публичные лекции, когда ученые или преподаватели высших учебных заведений рассказывали в популярной форме о достаточно сложных теориях и явлениях окружающего мира. Именно так автор и стремился построить свое изложение, всеми силами стараясь сделать материал интересным и понятным без каких-либо специальных предварительных знаний. Поэтому даже если читатель не является школьником или студентом, его не должно настораживать слово лекция в названии книги.

Пользуясь случаем, хочется выразить благодарность редакции газеты "Информатика" и лично ее редактору С. Л. Островскому за всестороннюю поддержку работы над первоначальным вариантом лекций.

Введение

Знать, как он работает, не менее важно, чем уметь работать с ПК. Вы можете вполне успешно пользоваться услугами компьютера, не понимая того, что в нем происходит. Однако чем глубже вы представляете процессы, происходящие в ПК, тем лучше будете использовать его возможности... Если что-нибудь случится в процессе работы с компьютером, вероятность того, что вы примите правильное решение, а не наделаете глупостей и не испортите все окончательно, будет выше.

П. Нортон

Важно... понимать, что эта вот привычка к пониманию логики вещей — сильная сторона российского образования. На эту привычку, умение, желание понять, разобраться в любом предмете в Америке сегодня есть большой спрос.

Я полагаю, что такой спрос и в Америке, и в России, и во всем мире с годами будет только увеличиваться.

А. Г. Кушниренко

Уважаемый читатель!

Вы держите в руках еще одну книгу о компьютерах. И, наверное, первый вопрос, который у вас возникает — чем она отличается от множества других?

Прежде всего, обратите внимание на тематику книги. Она посвящена внутреннему устройству компьютера, что немедленно дистанцирует ее от множества самоучителей, хороших и разных, часто обещающих вам немедленное освоение всех возможностей компьютера по приведенным в них рецептам и рекомендациям. Напротив, автор попытается помочь вам в постепенном понимании того, как работает компьютер, и это в свою очередь обеспечит плавный, но существенный рост ваших возможностей в грамотном его использовании.

С другой стороны, об устройстве компьютеров уже издано множество книг — справочников, пособий по выбору, ремонту и модернизации, технической литературы. Данная книга заметно отличается и от них. Во-первых, она написана в предположении, что читатель не имеет особой предварительной теоретической подготовки. Во-вторых, предметом обсуждения яв-

ляются не технические детали (успевающие порой устареть быстрее, чем появятся хорошие подробные книги о них), а фундаментальные, "проверенные временем" сведения, которые могут пригодиться при использовании компьютеров не только сегодня, но и, по крайней мере, в ближайшем будущем. В-третьих, данная книга не является сухим академическим изложением теории. Напротив, каждая лекция в дополнение к изложенному доступным языком материалу содержит большое количество практически полезных познавательных упражнений. Например: как определить объем памяти вашего компьютера и увидеть, достаточно ли ее; каким образом узнать расположение на диске только что сохраненного файла и влияет ли его фрагментация на скорость считывания данных, а также множество других подобных примеров. Перелистайте книгу — увидите сами.

Таким образом, остается лишь небольшая группа книг об устройстве компьютера, где доступным языком излагаются наиболее фундаментальные принципы его работы. Если дополнительно учесть уже упоминавшуюся демонстрацию практического применения этих знаний, то читатель, вероятно, согласится, что подобной литературы не так уж много. Определенную надежду на то, что эта книга получилась оригинальной, интересной и заслуживающей внимания широкой аудитории, дает автору его многолетний (начиная с 1974 года) опыт работы с вычислительной техникой, чуть менее короткий (с 1979 года) опыт преподавания многочисленных курсов по данной тематике и, наконец, достаточно большое количество написанных учебных пособий и публикаций образовательного характера (с 1985 по 2002 год их вышло в свет более сорока).

Следующая не менее важная характеристика книги — кому она может быть полезна? Любому человеку, интересующемуся компьютерами. Важно только, чтобы он действительно испытывал положительные эмоции, сидя за дисплеем, и у него хотя бы иногда возникали при этом вопросы "Почему?", "Зачем?" и "Как?" Особый интерес книга представляет для школьников и студентов, а также для преподавателей учебных заведений любого уровня. Иными словами, она вполне может служить обычной книгой для познавательного чтения, и в то же время, ее материал можно положить в основу учебного курса по соответствующему разделу информатики.

Еще раз подчеркнем, что предлагаемые вашему вниманию лекции призваны не столько разъяснять детальное устройство компьютера какой-либо определенной существующей архитектуры, сколько изложить наиболее общие базовые принципы и основную логику работы современных вычислительных машин. Иначе говоря, книга не претендует на освещение всевозможных технических деталей компьютера семейства IBM PC, процессора Pentium или программного обеспечения фирмы Microsoft для этого класса машин, хотя примеры в основном используются из этой области (в силу "национальных особенностей" компьютерного развития нашей страны триада IBM PC — Pentium — Windows является на данный момент существенно преобладающей).

Надеюсь, читатели уже составили некоторое представление о том, что они смогут узнать, познакомившись с этой книгой, равно как и о том, что в ней заведомо не стоит искать. А теперь несколько слов, как построено изложение материала в лекциях.

Первая лекция имеет вводный характер. В ней подробно рассматривается, насколько полезны знания устройства и логики работы компьютера для тех, кто его использует. Это единственное место в книге, где автор явным образом пытается убедить читателей в необходимости понимания основ функционирования вычислительной техники. В этой же лекции дается объяснение тому, по каким критериям отбирался материал для курса. Хотя все вышеизложенные вопросы в известном смысле являются вспомогательными, автор все же советует не пропускать вводной лекции и ознакомиться с ней, тем более что она содержит несколько оригинальных интересных примеров и иллюстраций.

Вторая лекция является ключевой. Именно здесь анализируется структура ЭВМ, ее основные блоки и взаимодействие между ними. Фактически этот материал является фундаментом для всех последующих лекций.

Лекции 3—6 улучшат ваши познания по вопросам функционирования каждого отдельного узла компьютера. Их материал тесно связан между собой, а также с *лекцией 2*, поэтому в тексте присутствует множество перекрестных ссылок. Раскрывается назначение каждого устройства, принципы и особенности организации его работы. Большое внимание при этом уделяется влиянию взаимодействия отдельных устройств на производительность всей машины в целом.

Седьмая лекция является переходом от изучения аппаратной части к программной. Здесь рассматривается значение программного обеспечения в современном компьютере, его состав, подробно проанализирована роль операционной системы в организации работы оборудования ЭВМ. Отдельно разбирается процесс загрузки программного обеспечения в компьютер.

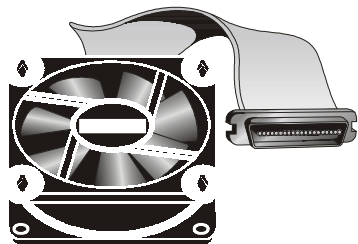
Наконец, *последняя лекция* посвящена учебным моделям компьютера. Ее основная идея состоит в том, что далеко не всегда удобно изучать принципы работы вычислительной техники на основе реальной ЭВМ. В лекции рассматривается целый ряд учебных моделей, включая самую современную "модель тысячелетия", предложенную классиком фундаментальной литературы по алгоритмам Д. Кнудом. Проводится сравнительный анализ достоинств и недостатков моделей.

Каждая из перечисленных выше лекций начинается с изложения теоретического материала, который составляет большую часть лекции. Затем формулируются основные выводы, и предлагается перечень вопросов, направленных на осмысление изложенного материала. Завершает большинство лекций описание любопытных экспериментов по теме лекции, которые читатели могут воспроизвести, следуя подробным рекомендациям книги. Хотя разбор

и повторение этих экспериментов не является обязательным, они, тем не менее, служат существенным дополнением для более глубокого усвоения содержания лекций.

Для проведения экспериментов всегда требуются соответствующие приборы и специализированные инструменты. Для данного курса их роль выполняет программное обеспечение. Во многих случаях удается воспользоваться стандартными средствами операционной системы, однако для организации более тонких и интересных экспериментов требуются более гибкие средства. Поэтому для некоторой части экспериментов приходится запускать специально разработанные для этой цели автором короткие программы на языке Паскаль (данный язык программирования обоснованно считается одним из самых наглядных и простых для понимания). В случае, если читатель не знаком с простейшими конструкциями Паскаля и не планирует в ближайшем будущем их изучать, для повторения описанных экспериментов можно запустить уже откомпилированный исполняемый файл с прилагаемого к книге CD-диска.

ЛЕКЦИЯ 1



Что мы будем изучать и почему

Наша первая лекция, как это обычно принято, имеет вводный характер. Прежде чем начинать изучение конкретных вопросов, полезно обсудить некоторые наиболее важные положения, относящиеся ко всему курсу в целом.

В первую очередь следует разобраться в том, какую роль играет данный курс в освоении компьютера. В каком-то смысле в лекции будет сделана еще одна попытка убедить сомневающихся читателей, а заодно и обосновать полезность знания фундаментальных основ работы вычислительной техники (ВТ). Возможно, некоторые читатели уже имеют четко оформившееся положительное мнение по этому вопросу — для них начальная часть лекции может показаться не совсем актуальной, и в этом случае ее можно прочитать не очень внимательно.

Следующий важный вопрос — это принципы отбора материала для данного курса. По мнению автора, ознакомившись с нижеизложенным, читатель сразу поймет, что он найдет в этой книге, а какой материал в нее сознательно не включен, и поэтому его лучше поискать в других изданиях.

Наконец, наиболее яркой и интересной обещает стать вторая половина лекции. В ней поставлена цель продемонстрировать, что, несмотря на огромную скорость развития вычислительной техники, ее фундаментальные основы во многом остаются неизменными. Помимо теоретического обоснования нашего курса, излагаемый материал содержит еще целый ряд оригинальных и интересных примеров, которые определенно привлекут внимание читателей. Обычно популярные книги стараются всячески поразить нас стремительными изменениями в мире компьютеров. В данной же лекции, напротив, автор призывает обратить внимание на другую сторону процесса и удивляться тому, что многие принципы до сих пор не изменились со времен изобретения самых первых ЭВМ.

Примечание

В тексте книги не делается особого различия между терминами "компьютер" и "ЭВМ", поскольку разница эта достаточно условна и, по крайней мере, при изложении наиболее универсальных закономерностей практически незаметна.

1.1. О возможных подходах к изучению компьютера

Давайте начнем разговор с того, что определим для себя, стоит ли вообще изучать устройство компьютера, и если да, то как это лучше делать. Обсуждение этого вопроса требуется непременно, поскольку даже многие из тех, кто использует компьютер более или менее регулярно, не считают знание основ его работы необходимостью. Вот как образно характеризует ситуацию один из известных авторов компьютерной литературы Чарльз Петцольд (Charles Petzold) [67]:

"Приходилось мне слышать и такое: "Народу нет дела до того, как работают компьютеры". И это, по-моему, тоже верно. Мне, например, доставляет удовольствие узнавать, как работает тот или иной прибор. Но при этом я предпочитаю выбирать, в чем я хочу разбираться, а в чем — нет. Мне, скажем, трудно без насилия над собой объяснить принцип действия холодильника.

И все же люди часто задают вопросы, которые подтверждают их интерес к внутреннему устройству компьютера. Один из них: "В чем разница между оперативной и постоянной памятью?"

Это, без сомнения, очень важный вопрос. На подобных понятиях основан весь рынок персональных компьютеров. Предполагается, что даже новичок понимает, сколько *мега* в одной и *гига* другой потребуется его программам. Считается также, что в самом начале знакомства с компьютером пользователь постигает концепцию файла и представляет себе, как файлы загружаются из постоянной в оперативную память, и затем из оперативной записываются обратно в постоянную".

Процесс освоения компьютера у каждого человека происходит по-своему, следовательно, единого универсального ответа на вопрос "нужно ли представлять себе устройство компьютера", вероятно, не существует. Мнения на этот счет сильно, порой диаметрально, расходятся.

Существует большое количество людей, которым принципиально все равно, как работает компьютер. В основу своего взаимодействия с ним они кладут один-два любимых самоучителя или справочника, советы грамотных друзей и, конечно же, хорошую память. Если сегодня мы зайдем в какой-либо из книжных магазинов, то увидим множество книг (и даже серий книг), в заголовках которых стоят слова "без проблем", "за 14 дней", "для занятых",

"одним взглядом" и множество аналогичных. Ничуть не пытаюсь ругать или отвергать эти книги (в конце концов, раз их выпускают в таком количестве, значит кому-то они полезны!), давайте все же задумаемся, в чем их основная суть. Мне кажется, вы со мной согласитесь, что главный прием, с помощью которого такого рода книги учат нас, наиболее кратко и емко характеризуется словом "рецепт". В самом деле, каждый фрагмент, начинающийся со слов "для того, чтобы выполнить действие X, необходимо..." и содержащий далее детальный перечень конкретных шагов, есть не что иное, как рецепт (или, если хотите, более красиво, алгоритм) достижения поставленной цели. В случае, когда такой рецепт удается найти (в памяти пользователя, в книге, в конспекте лекций всевозможных курсов или в виде совета знающего друга), любой, даже самый малоопытный человек довольно успешно справляется с работой, если он просто будет внимательно следовать "руководящим указаниям".

В каких случаях рецептурный подход дает удовлетворительные результаты? Когда человек пользуется компьютером время от времени и разнообразие его действий невелико: несколько режимов работы в одной-двух программах. Когда программы, с которыми он работает, практически не меняются или меняются крайне редко. Наконец, когда у него постоянно есть рядом более опытный коллега, способный в любую минуту ответить на вопрос или помочь разрешить возникшую трудность. Довольно много профессий, начиная от бухгалтера и кончая писателем, могут удовлетворять перечисленным условиям.

Тем не менее, я глубоко убежден, что подобный подход сильно ограничен, поэтому то и дело такие пользователи что-то делают неправильно, причем порой необратимо неправильно, так что результаты работы фактически теряются. Особенно часто это случается в нестандартных или неожиданных ситуациях: изменение настроек используемой программы другими пользователями, нехватка места на диске, отсутствие файла, сбой компьютера и т. п. Кроме того, не всегда удается найти в книге "рецепт" именно для того действия, которое вы собираетесь сделать. Часто имеющееся описание отличается от ситуации на вашем компьютере какой-то незначительной, на первый взгляд, деталью, но именно она мешает успешно завершить операцию. Иногда не совсем четко сформулировано, как именно нужно выполнить тот или иной шаг. И это далеко не самый полный перечень.

Вот простой и вполне жизненный пример. Человека научили загружать картинки в графический редактор Paint, наносить на них некоторые поясняющие тексты и сохранять результаты изменений обратно на диск. Первая партия картинок уже обработана, и все идет замечательно. А вот со второй неожиданно возникают проблемы: пользователь делает все точно так же, но машина упорно "не желает" сохранять исправленный рисунок, ссылаясь на отсутствие доступа к файлу (между прочим, речь идет не о какой-нибудь древней версии операционной системы, а о Windows 2000!) Так в чем тут дело? Оказывается, вторая партия картинок была скопирована с компакт-

диска, и поэтому все они автоматически получили статус "только чтение", что, естественно, препятствует внесению изменений. В результате прекрасно работавший до этого рецепт оказывается бессильным. А что касается плохой диагностики ошибок в современном программном обеспечении, на которое хочется списать возникшую трудность, так это, к сожалению, скорее правило, чем исключение.

Кстати, именно на поверхностное понимание работы компьютера зачастую рассчитаны вирусы. Вспомните, например, один из нашедших почтовых вирусов, который содержал фотографию известной российской теннисистки Анны Курниковой [И-6]. На деле такая "фотография", прилагаемая к письму, была исполняемым файлом "с двойным расширением" AnnaKournikova.jpg.vbs. Расчет был во многом сделан на то, что многие почтовые программы не отображают расширения приложенных файлов: в результате пользователи видели интригующую подпись AnnaKournikova.jpg, и многие из них в прямом смысле слова собственноручно запускали вирусную программу. А между тем увидеть истинный тип файла при желании совсем нетрудно — достаточно стандартным образом, используя правую кнопку мыши, посмотреть его свойства.

И еще один пример. Допустим, в качестве упражнения по освоению электронной таблицы Excel вы решили проверить тот факт, что $n! = (n - 1)! * n$. Примерное решение задачи может выглядеть так, как показано в табл. 1.1.

Таблица 1.1. Пример работы с электронной таблицей Excel

A1 = 5		C1 = ФАКТР(A1)	
A2 = A1 - 1	B2 = ФАКТР(A2)	C2 = B2 * A1	
		C3 = C1 - C2	D3 = ЕСЛИ(C1 - C2 = 0; "да"; "нет")

Примечание

На всякий случай напомним читателю, что запись $n!$, в математике называемая "красивым" термином *факториал*, есть просто произведение последовательных целых чисел от 1 до n . Например, $5! = 120$.

Аккуратно внесем формулы в электронную таблицу (можно использовать файл overflow.xls, подготовленный на прилагаемом к книге CD-диске в каталоге под номером 1), и Excel послушно подтвердит, что все правильно, написав "да" в ячейке D3. А если взять другое число, допустим, 25? К нашему удивлению, появится результат, изображенный на рис. 1.1.

Неожиданным здесь является то, что ячейка C3, вычисленная как разность C1 - C2, равняется нулю, но проверка в клетке D3 утверждает, что это не так! Эффект имеет простое объяснение, но только при условии понимания принципов хранения и обработки чисел в ЭВМ. Все дело в том, что $25!$ до-

вольно большое число и в ходе вычислений становится недостаточно имеющейся разрядной сетки для полного сохранения результатов; приходится последние (наименее важные) разряды округлять.

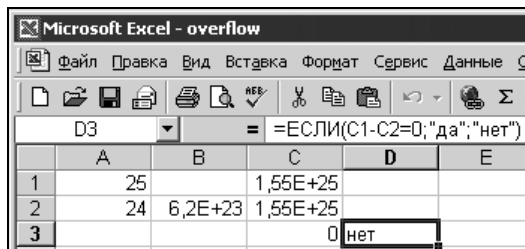


Рис. 1.1. "Странный" результат вычислений при $n = 25$

Итак, пусть мы поняли, что "выросли" из работы по готовым рецептам и хотим научиться более эффективной работе с компьютером. Какой же путь выбрать взамен? В качестве альтернативы можно предложить изучение наиболее общих закономерностей работы компьютера и его логики. Понимание тех принципов, на которых построено компьютерное оборудование и управляющее им программное обеспечение, позволит вам более грамотно анализировать возникающие в ходе работы трудности, а значит, и преодолевать их. Перечитайте еще раз цитату, вынесенную в качестве эпиграфа к книге — она принадлежит перу человека-легенды Питера Нортона, создателя известной каждому пользователю IBM PC программы Norton Commander, а также автору множества не менее известных во всем мире популярных книг о работе компьютера. Согласитесь, к его мнению стоит прислушаться.

В идеале понимание принципов работы компьютера должно приводить к полной ненужности "рецептов": используя свои знания и опыт, человек сам приобретает способность формулировать для себя такие "рецепты". Причем, особенно важно, что они не будут единими на все случаи жизни, как в справочнике, а полностью соответствующими конкретной сложившейся ситуации. Вы полагаете, это невозможно? А как же тогда ваши знакомые советчики, эти своеобразные "ходячие энциклопедии"? Неужели вы думаете, что они все свои рекомендации помнят наизусть?

Конечно, и на этом пути есть свои трудности и неудобства. Дело в том, что современный компьютер имеет достаточно сложное устройство, которое, к тому же, постоянно совершенствуется. Иными словами, материала для изучения много, и он все время меняется. Ситуация еще более усложняется тем, что обычно большая часть литературы по устройству компьютеров написана для людей, собирающих новые компьютеры или осуществляющих модернизацию старых. Отсюда в них рассматривается огромное количество технических деталей, которые рядовому пользователю, разумеется, не требуются. Как же тогда быть? Неужели мы выбрали тупиковый для нас путь?

Попробуем все же найти выход! Просто надо очень тщательно проанализировать имеющийся материал об устройстве компьютера, отобрать наиболее важные и существенные сведения, а второстепенные подробности и тонкости не принимать во внимание. К тому же, как будет показано в *разд. 1.3*, такие фундаментальные принципы как раз меняются гораздо реже, чем технические детали, что с наших позиций тоже выгодно — не придется слишком часто переучиваться.

Таким образом, путь освоения компьютера, кажется, стал понятен. Остается решить, как отобрать материал в соответствии со сформулированными выше принципами.

1.2. О содержании лекций

Итак, мы выяснили, что сведения для изучения устройства вычислительной техники нуждаются в тщательном отборе. В данном разделе будут описаны главные принципы, по которым осуществлялся подбор материала для наших лекций.

Начнем с того, что курс лекций направлен на знакомство с наиболее важными, наиболее фундаментальными принципами работы вычислительных устройств, т. к. именно они позволяют создать у рядового пользователя компьютера некоторую стройную взаимосвязанную картину. Данную особенность очень хорошо выразил один из авторов нескольких широко известных учебников информатики А. Г. Кушниренко. Он писал:

"Я спрашиваю себя: могу ли я сформировать целостное представление о компьютерах и компьютерных технологиях, не объяснив, что такое Гигабайт или чем текстовый файл отличается от документа Word? Выясняется, что не могу. Без этого картина получается какая-то однобокая. Значит, я буду пытаться объяснить в своем учебнике, что такое Гигабайт, что такое ASCII и какие бывают способы представления текстовых файлов.

Могу ли я построить целостное представление о компьютерах, не объяснив квантовой механики или двух режимов работы процессора Intel, — да, я могу без этого всего обойтись, и потому в моем учебнике этого скорее всего не будет. Но я могу себе представить, что разные режимы работы процессора Intel могут оказаться существенным элементом другой картины мира, представленной в другом учебнике".

В приведенной цитате упомянут и еще один существенный принцип отбора материала — технические детали не представляют интереса для общеобразовательного курса за исключением, может быть, немногочисленных иллюстрирующих изложение примеров. Действительно, существует большое количество особенностей устройства различных моделей компьютеров, но они интересны в основном специалистам. Само по себе знание многочисленных конкретных фактов и числовых характеристик в области компьютерного

оборудования ничего плохого, разумеется, не представляет. Важно только не потерять "за деревьями леса", т. е. видеть в этих разрозненных сведениях некоторую систему. Об опасности подмены систематических знаний набором несвязанных фактов замечательно сказал американский фантаст Рэй Бредбери в одной из наиболее любимых мною книг "451 градус по Фаренгейту":

"Устраивайте разные конкурсы, например: кто лучше помнит слова популярных песенок, кто может назвать все главные города штатов или кто знает, сколько собрали зерна в штате Айова в прошлом году. Набивайте людям головы цифрами, начинайте их безобидными фактами, пока их не затошнит, — ничего, зато им будет казаться, что они очень образованные. У них даже будет впечатление, что они мыслят, что они движутся вперед, хоть на самом деле они стоят на месте".

В качестве дополнения к обоим сформулированным выше принципам приведем еще один важный тезис. Технические детали устройства ЭВМ устаревают необычайно быстро, а фундаментальные принципы, напротив, используются в течение длительного времени. Достаточно для примера сказать, что базовые идеи построения вычислительных устройств, выдвинутые известным математиком Джоном фон Нейманом с группой соавторов в 1946 году, по-прежнему сохраняют свою актуальность, несмотря на то, что сменилось уже три поколения ЭВМ!

Вопросам кажущегося противоречия между бурным развитием вычислительной техники и необыкновенной стабильности основных принципов ее устройства посвящается следующий раздел.

Наконец, еще один принцип, часто используемый автором в данной книге, состоит в описании тех или иных особенностей устройства ЭВМ в контексте их исторического развития. Подобный подход вполне оправдан: как замечательно писал уже цитировавшийся в начале лекции Ч. Петцольд, он очень полезен для понимания современного состояния дел:

"Компьютеры наших дней сложнее тех, что появились 25 или 50 лет назад, но в основе своей они остались теми же. Вот почему изучать историю технологии так удобно: чем дальше вы уходите в прошлое, тем проще становится технология. Рано или поздно вы достигаете этапа, разобравшись в котором уже не представляет особого труда".

Таковы наиболее важные принципы отбора материала для всех лекций нашего курса. Подчеркивая важность этих принципов, хочется в заключение привести еще одну известную и очень подходящую к случаю цитату из художественной литературы. В своем рассказе "Этюд в багровых тонах" классик логического детектива Артур Конан Дойл писал:

"Мне представляется, что человеческий мозг похож на маленький пустой чердак, который вы можете обставить, как хотите. Дурак наташит туда всякой рухляди, какая попадется под руку, и полезные, нужные вещи уже

некуда будет всунуть, или в лучшем случае до них среди всякой завали и не докопаешься. А человек толковый тщательно отбирает то, что он поместит в свой мозговой чердак. Он возьмет лишь инструменты, которые понадобятся ему для работы, но зато их будет множество, и все он разложит в образцовом порядке. Напрасно люди думают, что у этой маленькой комнатки эластичные стены и их можно растягивать сколько угодно. Уверю вас, придет время, когда, приобретая новое, вы будете забывать что-то из прежнего. Поэтому страшно важно, чтобы ненужные сведения не вытесняли собой нужных".

1.3. Стремительное развитие ВТ и стабильность ее фундаментальных принципов

1.3.1. Темпы развития ЭВМ. Поколения ЭВМ

Развитие вычислительной техники в литературе принято характеризовать понятием *поколение ЭВМ*. Каждое новое поколение представляет собой существенный шаг вперед и открывает перед пользователями новые горизонты и перспективы. Однако никогда не образуется избытка вычислительной мощности: возникают новые задачи и опять требуется улучшение характеристик машин.

В основе существующего деления на поколения лежит элементная база, на которой строятся ЭВМ (табл. 1.2).

Таблица 1.2. Поколения ЭВМ

Поколение	Элементная база	Способ общения
1	Электронные лампы	Двоичные коды
2	Транзисторы	+ языки программирования
3	Микросхемы	+ язык управления заданиями
4	БИС	+ пользовательский интерфейс

Тема поколений ЭВМ подробно обсуждается в каждом учебнике информатики, поэтому разрешите здесь не повторять в очередной раз одни и те же описания. За подробностями можно обратиться, например, к книге [78]. Отметим только, что довольно четкая смена поколений, происходившая раньше примерно раз в десять лет, сейчас замедлилась. Не очень ясна и картина с пятым поколением ЭВМ, созданием которого занимались японские ученые: несмотря на огромную проделанную ими работу, мы по-

прежнему продолжаем пользоваться микропроцессорными машинами четвертого поколения. По-видимому, эти факты лишний раз подтверждают некоторую искусственность всяческих классификаций, хотя, разумеется, сомнений в том, что славная летопись поколений будет со временем продолжена и дополнена, не возникает.

Обязательно обратите внимание на последний столбец табл. 1.2. Развитие вычислительной техники сопровождается отчетливым совершенствованием способов общения человека с машиной и созданием новых все более удобных устройств ввода и вывода данных. Происходит постепенная замена языка машины (двоичные коды) на все более близкие человеку формы обмена информацией.

Чтобы показать огромную скорость развития в производстве вычислительной техники, обычно приводится масса разнообразных сравнений и фактов. Например, в 1993 году мощность первых процессоров Pentium превышала вычислительные возможности NASA на момент высадки космонавтов на Луну. Автору, чтобы не повторяться, хочется привести какой-нибудь оригинальный пример. Перед вами на рис. 1.2 фотография ячеек ЭВМ трех разных поколений (см. слева направо): на лампах, на транзисторах и на микросхемах; надеюсь, качество фотографии позволяет вам хорошо рассмотреть последнюю маленькую ячейку в виде "многоногого жучка". Подчеркнем, что с точки зрения выполняемых ими функций все ячейки абсолютно одинаковы — каждая из них представляет собой два триггера (*триггер* — это электронная схема для хранения одной минимальной единицы информации, т. е. 1 бита).

Фотография фактически является наглядной диаграммой уменьшения размеров элементов вычислительной техники для первых трех поколений (авторучка специально находится в кадре для того, чтобы дать читателям возможность иметь некоторый характерный масштаб изображения; для этой же цели в качестве фона подложена миллиметровая бумага). Понятно, что вычленить два триггера из современного микропроцессора четвертого поколения не представляется возможным: для этого надо из кристаллика длиной порядка 1 см выделить ничтожную его часть. Даже если бы это каким-то чудесным образом удалось, камера, с помощью которой сделана данная фотография, все равно не сумела бы запечатлеть столь крошечный объект.

Как уже было сказано, показанные на фотографии ячейки содержат триггеры. Поскольку триггер является важным цифровым устройством, способным хранить один бит информации, мы еще неоднократно встретимся с ним в нашем курсе. А пока в ознакомительных целях приведем на рис. 1.3 схему некогда распространенной микросхемы K1TK552 (два D-триггера) [74], фотография которой, кстати, и была приведена на рис. 1.2. Левая часть схемы реализует определенную входную логику, а правая, состоящая из двух симметричных узлов, образует собственно триггер. На выходе Q отражается хранящаяся в триггере информация: 0 или 1.

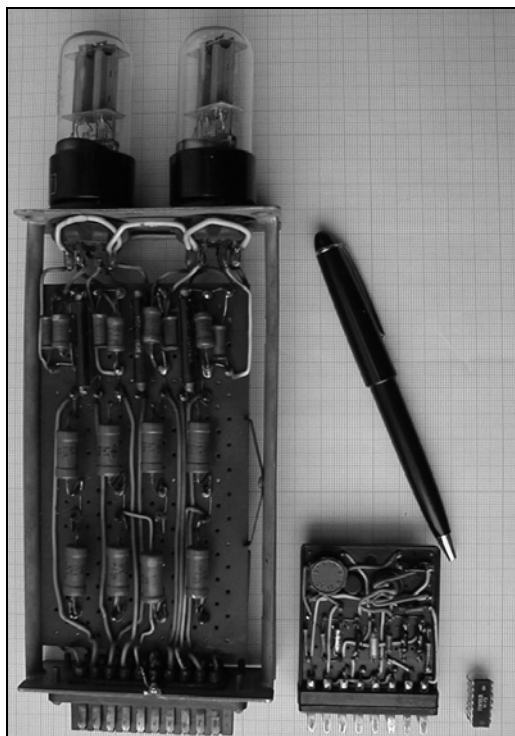


Рис. 1.2. Уменьшение размеров ячеек ЭВМ первых трех поколений

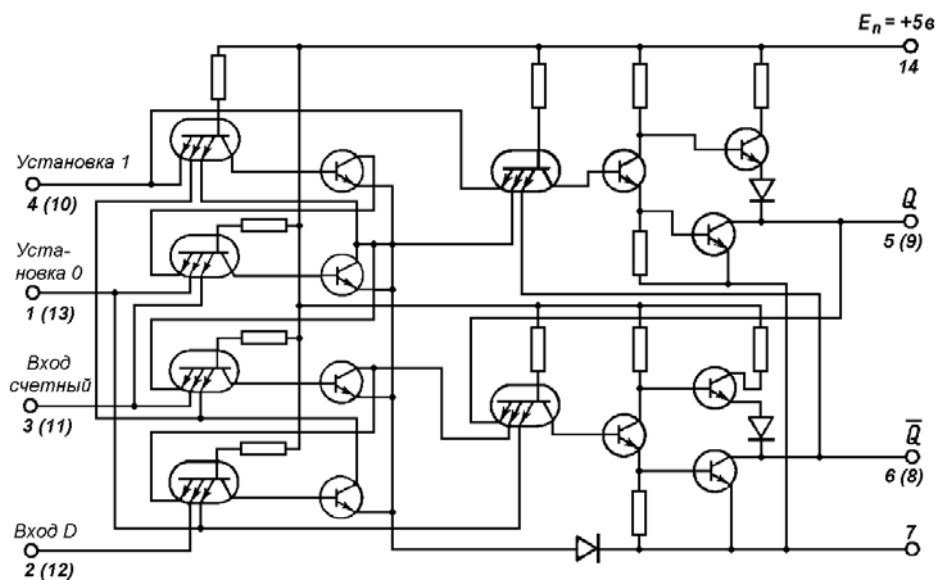


Рис. 1.3. Принципиальная схема интегрального триггера

Итак, темпы развития технологии производства компонентов вычислительной техники поистине впечатляющи. А насколько изменяются фундаментальные принципы построения ЭВМ? Сравним между собой две достаточно типичные вычислительные машины, с которыми автору приходилось лично иметь дело. Одна из них, ЭВМ модели Наири-2, принадлежала ко второму поколению и была в свое время достаточно распространена. Первый сеанс общения с этой машиной у автора состоялся в 1974 году; уже более десяти лет прошло с тех пор, как ее разобрали на составные части (кстати, из главного блока после небольшой переделки вышел замечательный стол огромных размеров). В качестве второго компьютера, с которым мы будем сравнивать, возьмем обычный компьютер семейства IBM PC с процессором Intel 80486. Такие машины еще вполне можно увидеть, хотя, разумеется, это далеко не последняя модель. Вы спросите почему не Pentium? Да потому, что система команд процессоров все больше расширяется, но в основном за счет достаточно специфических операций. Все это только смазывает картину, хотя и принципиально не меняет ее (см. прим. 1 к табл. 1.4).

1.3.2. Система команд типичной ЭВМ второго поколения

ЭВМ Наири-2 [55] была разработана в 1968 году и серийно производилась в Ереване. Ее внешний вид показан на рис. 1.4: машина состояла из двух крупных по размеру блоков, причем слева на стойке с блоками питания стоят периферийные устройства (электрифицированная пишущая машинка, фотосчитыватель с бумажной ленты и перфоратор для ее вывода), а собственно машина находится на фотографии справа. Обязательно обратите внимание на длинные ряды лампочек в центральной верхней части машины; для удобства считывания они были разбиты на группы с помощью колпачков белого и черного цвета и отображали двоичное состояние внутренних регистров процессора ЭВМ.

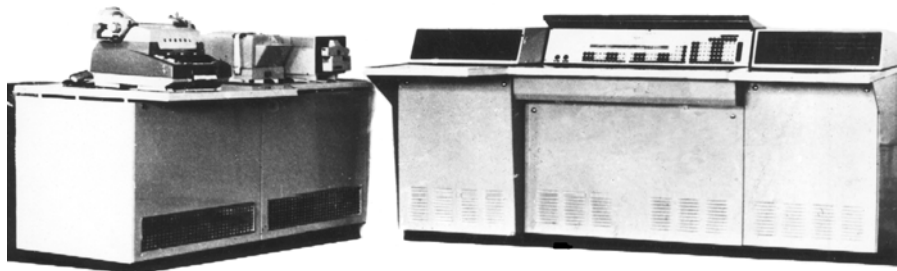


Рис. 1.4. Вид ЭВМ второго поколения Наири-2

Наири-2 могла работать с целыми и вещественными числами (стандартные вещественные числа хранились в одной ячейке, а длинные — в двух после-

довательных ячейках). В систему ее команд входило 102 инструкции, из них 47 — собственно машинные операции (табл. 1.3), а остальные — так называемые псевдооперации, реализуемые при помощи специальных подпрограмм в долговременном запоминающем устройстве — ДЗУ. (Интересно вспомнить, что ДЗУ было выполнено на ферритовых кольцах, причем информация в нем в прямом смысле слова "прошивалась" путем протравливания тонкого медного провода через сердечник при единице и мимо него при нуле.) Важное положение среди псевдоопераций занимали разнообразные действия над вещественными числами, которые мы для простоты изложения рассматривать не будем.

Таблица 1.3. Полная система машинных операций ЭВМ Наири-2 (1968 г.)

п, п1	Передача числа
с, с1, с2, с3	Сложение
в, в1, в2, в3	Вычитание
у, у1, у2, у3	Умножение
д, д1, д2, д3	Деление
в4	Вычитание модулей
л, л1	Логическое сложение (AND)
л, л1	Логическое умножение (OR)
м, м1	Сложение по модулю 2 (XOR)
а, а1	Сдвиг вправо (арифметический)
δ, δ1	Сдвиг влево (логический)
г, г1	Длинный сдвиг
н, н1	Нормализация
и, и1, и2, и3	Абсолютный переход с возвратом
е, е1, е2, е3	Относительный переход с возвратом
к, к1	Останов
ч	Чтение
о, о1	Обращение (вывод)
х	Холостая команда

Примечание

Большинство инструкций имеют несколько модификаций, которые обозначены цифровым индексом. Например, операция п копирует содержимое одной ячейки памяти в другую, а п1 сохраняет результат предыдущей операции в память.

Приведем также примеры некоторых (далеко не всех!) псевдоопераций ЭВМ Наири-2, которые находились в ее ДЗУ, состоявшем из 16 384 ячеек (команд). Это ln — вычисление логарифма, sn — вычисление синуса, pp — печать чисел с плавающей запятой, ps — печать содержимого (в двоичном виде), pk — печать команд, cd , vd , ud , dd — арифметические операции с длинными числами, sk , vk , uk , dk — действия над комплексными числами, ga — вычисление гамма функции, mn и mx — нахождение минимума и максимума в массиве и многие другие.

1.3.3. Система команд типичной ЭВМ четвертого поколения

А теперь для сравнения приведем систему команд хорошо знакомого всем процессора Intel 80486, который и поныне все еще продолжает служить людям в некоторых не слишком новых компьютерах. На рис. 1.5 можно увидеть фотографию главной платы компьютера на базе процессора Intel 80486. Табл. 1.4 построена по данным монографии [7].



Рис. 1.5. Вид главной платы компьютера на базе процессора Intel 80486

Таблица 1.4. Основные команды микропроцессора Intel 80486

MOV	Пересылка данных
PUSH, PUSHA, POP, POPA	Чтение/запись в стек
LEA, LDS, LES, LFS, LGS, LSS	Загрузка адреса и селекторов
MOVSX, MOVZX	Пересылка с расширением знака
XCHG	Обмен содержимым
XLAT	Преобразование кодов
BSWAP	Перестановка байтов
ADD, ADC, XADD, INC, AAA, DAA	Сложение
SUB, SBB, DEC, CMP, CMPXCHG, NEG, AAS, DAS	Вычитание
MUL, IMUL, AAM	Умножение
DIV, IDIV, AAD	Деление
CBW, CWDE, CWD, CWQ	Преобразование разрядности данных
NOT	Логическое НЕ
AND	И
OR	ИЛИ
XOR	Исключающее ИЛИ
TEST	Логическое сравнение операндов
SHL, SAL	Логический и арифметический сдвиг влево
SHR	Сдвиг вправо
SHLD, SHRD	Двухоперандные сдвиги
ROL, ROR, RCL, RCR	Циклические сдвиги
BT, BTS, BTR, BTC, BSF, BSR	Операции с битами
LODS, STOS, INS, OUTS, MOVS, CMPS, SCAS	Операции со строками
JMP	Безусловный переход
JA/JNBE, JAE/JNB/JNC, JB/JNAE/JC, JBE/JNA, JE/JZ и др.	Условные относительные переходы
CALL, RET, INT, IRET	Переход к подпрограммам и возврат
SETA/SETNBE, SETAE/SETNB и др.	Условная установка байта
LOOP, LOOPE/LOOPZ, LOOPNE/LOOPNZ, JCXZ	Организация циклов

Таблица 1.4 (окончание)

MOV	Пересылка данных
IN	Ввод из порта в аккумулятор
OUT	Вывод из аккумулятора в порт
HLT	Останов
NOP	Отсутствие операции

Примечания

- Для краткости в таблицу не включены некоторые группы команд, связанные с управлением процессором, организацией защиты памяти, поддержкой языков высокого уровня и некоторые другие. Кстати, именно эти инструкции являются максимально машинно зависимыми, т. е. существенно зависят от модели процессора. Иными словами, с точки зрения общеобразовательных целей они представляют гораздо меньший интерес.
- Особо следует сказать об операциях с вещественными числами. В процессорах фирмы Intel для выполнения этих действий используется специальный блок — так называемый математический сопроцессор. Начиная с модели 486, он изготавливается в том же самом кристалле, что и основной процессор. Команды сопроцессора в таблице также не отражены.

Перейдем к анализу табл. 1.3 и 1.4.

1.3.4. Сравнение систем команд ЭВМ второго и четвертого поколений

А теперь самая главная и интересная часть — сравним между собой табл. 1.3 и 1.4. Напомним читателю, что они разделены огромным по меркам развития вычислительной техники временем — более 20 лет. Серым цветом в таблицах выделены клетки, содержащие аналогичные операции. И как ни удивительно, они охватывают большую часть анализируемых таблиц! Кроме того, не следует забывать, что процессоры фирмы Intel относятся к категории CISC-процессоров (Complex Instruction Set Computer), т. е. процессоров с расширенной системой операций. А это значит, что для многих других процессоров результаты аналогичного сравнения будут еще более близкими.

Примечание

Читателям, которые усомнятся в вышеизложенном по причине того, что 486-й процессор уже устарел, советую обратиться к любой книге по процессорам Pentium [14] и убедиться, что появившиеся в последних процессорах новые команды принципиально не изменяют ситуации.

Итак, получаем важный и интересный вывод: ядро системы команд, образуемое такими группами операций, как перепись информации, арифметические и логические команды, сдвиги и переходы, за несколько поколений (!) ЭВМ практически не изменилось. Неужели вас, уважаемые читатели, это не удивляет?!

Еще одно наблюдение состоит в том, что дополнительные команды процессоров Intel (см. табл. 1.4) в основном расширяют возможности программиста, а не предоставляют принципиально новых возможностей. Скажем, наличие группы команд для организации циклов во многих случаях облегчает написание программы, но, с другой стороны, любой цикл можно реализовать и без них, с помощью инструкций условных переходов. Или операции над отдельными битами, которые всегда можно выполнить на базе стандартных логических операций.

Пожалуй, наиболее важным прогрессом в развитии ЭВМ является переход к обработке значительно большего количества различных данных, который как раз не очень заметен из сравнения обсуждаемых таблиц. Хотя ЭВМ первых поколений и могли работать с несколькими видами информации (в основном числами) и даже имели "зачаточные" возможности по выводу отдельных символов, развитые средства обработки строковых данных там отсутствовали. Работа со звуком и видео в те времена, естественно, даже не обсуждалась. Основу памяти таких машин составляла ячейка под стандартное число или команду (в Наири ее длина равнялась 36 двоичным разрядам), поэтому данные другой длины не могли при такой организации рационально храниться.

Обязательно обратите внимание на появление в табл. 1.4 новых по сравнению с табл. 1.3 групп команд преобразования данных: пересылка с расширением знака, перестановка байтов, преобразование разрядности данных, операции со строками. Причина этих нововведений состоит в том, что память современных компьютеров имеет значительно более гибкую байтовую структуру, о которой мы обязательно подробно поговорим в следующих лекциях.

И последнее замечание, которое касается обеих анализируемых таблиц. Обратите внимание на простоту команд, которые непосредственно могут исполняться процессором даже современной ЭВМ. Данный факт интересен тем, что сложное программное обеспечение, содержащее тысячи таких простых команд, как сложение, сравнение и др., может вести себя настолько разумно, что как-то забывается, что в его основе лежат все те же несложные базовые инструкции процессора.

1.4. Некоторые выводы

Итак, подведем итоги того, что мы узнали из первой лекции.

- ❑ Понимание основных принципов работы такого достаточно сложного в обращении и настройке устройства, как компьютер, полезно любому человеку, систематически его использующему.
- ❑ Подход, условно названный в лекции рецептурным, когда все манипуляции с компьютером выполняются в соответствии с заранее подготовленными кем-то описаниями, хорошо работает лишь в простейших случаях. Для выполнения квалифицированной работы на компьютере его ограниченность отчетливо видна.
- ❑ Необходимость в практическом применении знаний о фундаментальных основах ВТ возникает не всегда. Чем более необычна для пользователя ситуация, тем острее он нуждается в подобных знаниях, и особенно в умении их анализировать.
- ❑ В основе функционирования современного компьютера лежит большое количество идей, теорий, принципов и технических решений. Для того чтобы лучше ориентироваться в них, необходимо сформировать для себя некоторое систематическое представление о работе компьютера. Следовательно, очень важно суметь выделить те наиболее существенные принципы и факты, которые необходимо хорошо знать и достаточно глубоко понимать.
- ❑ Изучение тех или иных принципов устройства ЭВМ в их историческом развитии часто оказывается очень полезным для понимания современного состояния дел. К тому же, чем дальше вы уходите в прошлое, тем проще становится разобраться в рассматриваемых технологиях и понять суть дела.
- ❑ Несмотря на огромный технический прогресс, который увеличил на несколько порядков быстродействие ЭВМ и соответственно уменьшил их размеры, фундаментальные принципы построения ВТ в основном сохранились.
- ❑ Как это не парадоксально звучит, в основе всех самых "умных" и изощренных программ лежит достаточно ограниченная система простых инструкций, которые непосредственно понимает процессор компьютера. Секрет состоит в универсальности этих команд и в возможности их комбинировать.
- ❑ На самом низком (аппаратном) уровне инженеры стараются создать компьютер максимально просто. Для адаптации машины к задачам, которые требуется решать на практике, используется сложное программное обеспечение. Очевидно, что установить новое программное обеспечение значительно проще, чем изменять конструкцию компьютера.