



Журнал
для
пользователей
компьютеров

11 (55)
ноябрь 2002

Издательство "Техно-ПРЕСС", С.-Петербург

**О компьютерах
идеальных и оптимальных**

Третье измерение грядет!

**В компании умных
чайников**

**Как Microsoft
изобрела колесо**

To be XOR not to be?

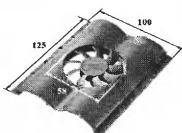
**Быстро, удобно,
"безбумажно"**

Крейг Барретт –

Нам предстоит цифровое будущее

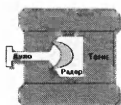
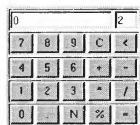
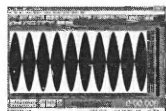


ALU
ALU
FPU
Load/
Store
1 2 3



1921567900

167407030



КОМПЬЮТЕРЫ

Intel смотрит в будущее.....	2
Два логических процессора на одном физическом.....	4
Третье измерение грядет!.....	5
О компьютерах идеальных и оптимальных.....	8
Что и как устаревает.....	11
Энциклопедия современного компьютера. Материнские платы.....	14
В компании умных чайников.....	19
Ох уж этот Quantum!.....	22
Как Microsoft изобрела колесо.....	24

ПЕРИФЕРИЯ

Пишем компакт-диск.....	27
"Зеркалка" для наследников Карла Буллы.....	31

КОМПЬЮТЕРНАЯ ГРАФИКА

Писать или набирать?.....	34
---------------------------	----

МУЗЫКАЛЬНЫЙ ПК

Дрожь в голосе — это так прекрасно!.....	36
--	----

ИНТЕРНЕТ

Идеальная партнерская программа.....	40
Самозагрузчик на PHP.....	44
Игры с графическими форматами: природу не обманешь.....	47
Калькулятор и Системный Анализ.....	48

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

Игры с таблицами.....	51
Сорок веков таблице умножения.....	52
XP: экстремальное программирование.....	54
To be XOR not to be?.....	57
Быстро, удобно, "безбумажно".....	58
Четыре танкиста и компьютер.....	61

МУЛЬТИМЕДИЯ

Сила звука. Цифровое радиовещание в диапазонах ДВ, СВ и КВ.....	64
---	----





INTEL СМОТРИТ В БУДУЩЕЕ

Владимир Буслаев

Заметным событием октября 2002 года стало проведение в Москве, в здании Российской Академии наук, крупного международного конгресса разработчиков аппаратного и программного обеспечения Intel Developer Forum (IDF). Россия стала первой страной Восточной Европы, на территории которой прошла выездная сессия Форума индустрии информационных технологий. Президент российского представительства корпорации Intel Стив Чейз в своем вступительном слове отметил: "Для проведения выездных сессий IDF выбираются страны, занимающие лидирующее положение в своих регионах, и на принятие решения о проведении форума в Москве повлияли потрясающие показатели роста компьютерной и телекоммуникационной индустрии".

"Мы находимся на пороге удивительной эры. Нам предстоит цифровое будущее, когда люди будут иметь доступ к Интернету в любое время и в любом месте с помощью устройств, сочетающих вычислительные и коммуникационные возможности."

Крейг Барретт

Основным событием форума стали доклад и пресс-конференция главного исполнительного директора корпорации Intel Крейга Барретта. Выступая перед аудиторией из примерно 800 человек, К. Барретт обрисовал будущее цифровых технологий, особое внимание уделив взаимопроникновению компьютерных и коммуникационных технологий, изменению общества под влиянием информатизации, а также перспективам развития информационной индустрии.

Ключевые моменты развития информационных технологий в настоящее время, отмеченные и в докладе, и в ответах на вопросы на пресс-конференции, можно свести к следующим тезисам:

- Основное направление развития — конвергенция компьютерных и коммуникационных технологий.

- Мощные персональные компьютеры и серверы в сочетании с высокопроизводительными сетями на основе технологий Gigabit Ethernet стандартов 802.11a, 802.11b и GPRS, оптоволоконных систем и модульной коммуникационной инфраструктуры объединят весь мир в единое информационное поле.

- Действие закона Мура будет справедливо еще в течение 10—15 лет. Ключевым фактором для продолжения его реализации становится нанотехнология, или молекулярное производство: в августе этого года Intel представила самый передовой в отрасли 90-нанометровый технологический процесс производства логических интегральных схем, который будет внедрен в массовое производство в 2003 году. В самом ближайшем будущем этот техпроцесс позволит создать микропро-

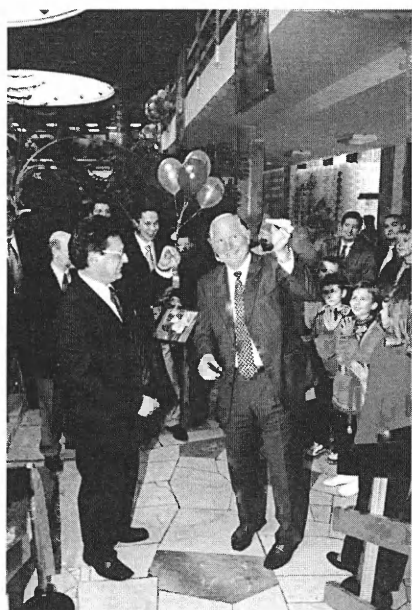
цессоры, способные работать с тактовыми частотами более 3 ГГц при одновременном уменьшении размера кристалла на 50%.

- Сегодня Intel применяет в массовом производстве 0,13-микронную литографическую технологию (кристалл, изготовленный по этой технологии, вмещает 55 млн транзисторов). Montecito, один из процессоров семейства Itanium 2, выпуск которого намечен на 2004 год, будет выполнен по 0,09-микронной технологии.

- Современные процессоры Pentium 4 имеют тактовые частоты 2,20 ГГц и более. Процессор Pentium 4 с тактовой частотой 3 ГГц корпорация планирует выпустить до конца этого года. Процессор с миллиардом транзисторов, появление которого ожидается к концу нынешнего десятилетия, будет работать на частоте, приближающейся к 20 ГГц.

В области коммуникаций Intel развивает несколько новых направлений:

- кремниевая волоконная оптика;
- кремниевое радио: создание эффективных, доступных и широко распространенных средств связи;
- специализированные сенсорные сети.



К. Барретт на открытии Центра семейного компьютерного творчества в Московском городском дворце детского (юношеского) творчества

К. Барретт призвал Россию поддержать развитие компьютерной и телекоммуникационной отраслей экономики с акцентом на технологии следующего поколения: "Россия с ее быстро развивающимся рынком технологий и обилием технических талантов имеет все предпосылки для того, чтобы стать лидером мировой отрасли высоких технологий. Рост информационных технологий на мировом рынке открывает перед Россией значительные возможности для создания новых рабочих мест и новой продукции, для расширения экспорта. Россия все явственнее превращается в ключевой центр разработки программного обеспечения со значительным потенциалом для роста за счет экспорта оборудования, знаний и опыта в области информационных и коммуникационных технологий. Инвестируя средства в разработку телекоммуникационных и вычислительных технологий, Россия может дать новый импульс своей экономике и создать предпосылки для дальнейшего роста. Стремясь содействовать разработке Россией технологий следующего поколения, корпорация Intel инвестирует средства в продукцию, архитектуру, обучение и программы, помогающие России строить надежную инфраструктуру и развивать сообщество разработчиков".

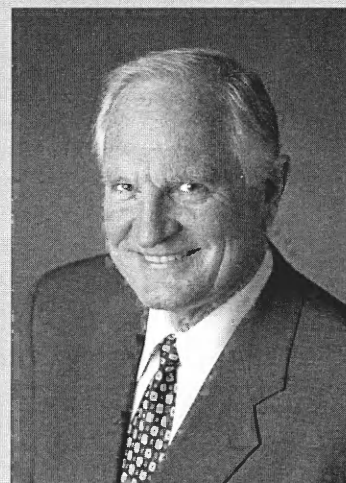
Кроме того К. Барретт призвал к модернизации телекоммуникационной инфраструктуры России и повсеместному внедрению беспроводного широкополосного доступа.

В тот же день К. Барретт принял участие в церемонии открытия первого в России Центра семейного компьютерного творчества в Московском городском дворце детского (юношеского) творчества. Центр предназначен для обучения детей и взрослых использованию цифровых технологий в таких областях, как фотография, обработка видеоизображений, компьютерная графика и т. п.

Вслед за Москвой должны состояться сессии Форума в Тайпее (Тайвань), Токио, Мумбае (Индия), Шенцзяне (Китай). Следующий IDF в России Intel планирует организовать в 2003 году.

Четвертый президент

Крейг Барретт родился в 1939 году в Сан-Франциско. В 1964 году закончил Стэнфордский университет, где получил степени бакалавра, магистра, а затем и доктора наук. В корпорации Intel работает с 1974 года. Занимал должности менеджера по развитию технологий, вице-президента (1984), старшего вице-президента (1987), исполнительного вице-президента (1990).



В мае 1997 года стал четвертым с момента основания корпорации президентом Intel, с мая 1998 года является также главным исполнительным директором компании. Автор более 40 технических статей о влиянии микроструктуры на свойства различных материалов и учебника по основам материаловедения. Является членом Национальной академии инженерных наук, государственного комитета США по обучению математике и естественным наукам в XXI веке, а также комитета по внешнеэкономической политике США в условиях глобализации XXI века, членом совета директоров Ассоциации полупроводниковой промышленности США. За успехи в науке и бизнесе награжден несколькими национальными медалями и премиями США. Среди них золотая медаль им. Харди (1969), почетная премия им. Гарри Гатоса (1996, MIT), медаль Green Cross for Safety (2001, Национальный совет по безопасности), медаль Генри Форда II "За лидерство в бизнесе" (2001).

В своем пленарном докладе на Московском форуме IDF д-р Крейг Барретт особо остановился на технологиях Intel, направленных на повышение производительности вычислений. Одной из таких перспективных технологий, по его мнению, является технология Hyper-Threading (HT), или технология одно-временной многопоточности (Simultaneous Multi-Threading, SMT).

Первоначально она создавалась как один из вариантов повышения производительности процессора. Идею технологии выдвинул в 1993 году почетный сотрудник корпорации Intel Глен Хинтон, заметив, что ресурсы процессоров часто используются не полностью. В 1996 году инженеры Intel приступили к интеграции технологии HT в Willamette, ядро процессоров Pentium 4 и Xeon. С февраля этого года технологией оснащаются серверы на базе процессоров Xeon. Руководство Intel обещает, что HT появится в настольных ПК на базе процессора Pentium 4 с тактовой частотой 3,06 ГГц в конце 2002 года. Ожидается, что это увеличит производительность до 25% при работе с приложениями, оптимизированными для технологии HT и многозадачных сред.

Как это работает

Существуют разные варианты повышения производительности процессора. Увеличение тактовой частоты, как известно, чревато побочными эффектами, прежде всего — разогревом процессора. Другое направление — увеличение размеров кэш-памяти, введение дополнительных исполнительных устройств. Однако это приводит к усложнению микроархитектуры процессора и его удорожанию.

Технология HT основана на принципе многопоточной обработки. Один процессор обречен на выполнение всегда только одного однородного потока в каждый момент времени. Теоретическое число инструкций, которые процессор может выполнить за определенный такт, намного выше действительного при

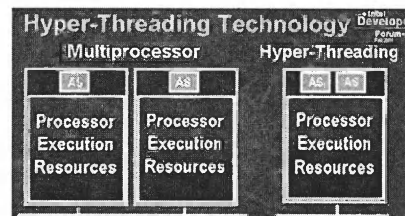
Два логических процессора на одном физическом

Владимир Буслаев

выполнении реальных задач. Связано это с тем, что исполнительные модули процессора (например, у Pentium 4 их в общей сложности семь) не всегда максимально загружены, и часть вычислительной мощности теряется. В среднем при выполнении кода, типичного для набора команд IA-32, реально используется только 35% ресурсов процессора. Остальные 65% просто-напросто простаивают. Следовательно, необходимо реализовать работу процессора таким образом, чтобы в каждом тактовом цикле возможности процессора использовались максимально, то есть чтобы он мог выполнять на аппаратном уровне несколько потоков одновременно.

Технология HT является чем-то промежуточным между многопоточной обработкой, реализованной в многопроцессорных системах, и параллелизмом на уровне инструкций в однопроцессорных системах.

Фактически новая технология позволяет организовать два логических процессора на одном физическом. С точки зрения операционной системы и запущенного приложения в системе существует два процессора, что дает ей возможность распределять загрузку задач между ними точно так же, как в случае многопроцессорной конфигурации.



Каждый из двух логических процессоров имеет свои регистры и контроллер прерываний (Architecture State, AS). Соответственно две параллельно выполняе-

мые задачи работают с независимыми регистрами и прерываниями, но при этом обе используют одни и те же ресурсы процессора (блоки целочисленного сложения, сложения с плавающей точкой, блоки циклического сдвига и т. п.). В архитектуру нового процессора помимо элементов собственно технологии HT добавляется кристалл кэша 3-го уровня. При этом поверхность самого кристалла возрастает всего на 5%.

Таким образом, от реальной двухпроцессорной конфигурации HT отличается только тем, что оба логических процессора используют одни и те же ресурсы, одну кэш-память, разделяемую между двумя потоками, и одну системную шину. Команды от обоих исполняемых параллельно потоков одновременно посылаются для обработки ядру процессора.

Используя принцип out-of-order (исполнение командных инструкций не в порядке их поступления), ядро процессора параллельно обрабатывает оба потока за счет использования нескольких исполнительных модулей. При этом никакого регулятора потоков, разделяющего ресурсы процессора между задачами, не существует.

Главное препятствие на пути к реализации этой простой на вид идеи заключается в отсутствии программ, которые смогли бы заполнить инструкциями все исполнительные устройства процессора. Обычно программы обходятся либо несложными целочисленными вычислениями да несколькими операциями загрузки и хранения данных, а операции с плавающей точкой остаются в стороне либо загружают работой в основном устройства для операций с плавающей точкой. Вот почему рост производительности Intel гарантирует только при работе с приложениями, оптимизированными для технологии HT и многозадачных сред.

Это можно проиллюстрировать на простом примере. Представьте себе процессор с тремя исполнительными устройствами: целочисленным арифметико-логическим (ALU), устройством для работы с плавающей точкой (FPU) и устройством загрузки/хранения (записи и чтения) данных из памяти. Предположим, что этот процессор может выполнять любую операцию за один такт и распределять операции по всем трем устройствам одновременно. Пусть к этому процессору на

выполнение отправляется поток из следующих трех инструкций:

1. Целочисленное сложение
2. Целочисленное сложение
3. Сохранение результата

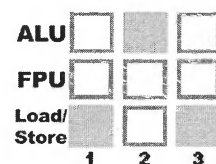
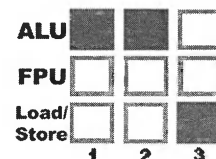
На рисунке показан уровень загрузки исполнительных устройств (белым цветом обозначены

незадействованные устройства, темно-серым — работающие). Как это видно, в каждый такт используется только 33% всех исполнительных устройств, а блок вычисления с плавающей точкой FPU вообще оказался незадействованным.

Отправим на выполнение другую последовательность инструкций (светло-серым цветом обозначены работающие устройства):

1. Загрузка данных
2. Целочисленное сложение
3. Сохранение данных

И снова загруженность исполнительных устройств составила 33%.



Третье измерение грядет!

Благодаря технологии Light Field Mapping реалистичные интерактивные 3D-объекты, создание которых ранее было доступно лишь разработчикам кинематографических спецэффектов на оборудовании класса high-end, становится возможным с помощью обычных ПК.

Особый интерес на Московском форуме IDF вызвала демонстрация специалистами нижегородского Центра Intel по разработке программного обеспечения технологии Light Field Mapping (LFM), созданной в сотрудничестве с исследовательской лабораторией Intel в Санта-Кларе (США). Технология предназначена для моделирования освещения на поверхности объектов по изображениям, полученным в результате фотографирования или сканирования. Посредством LFM любые трехмерные объекты или сцены, реальные и виртуальные, могут быть представлены

в виде цифровых моделей и затем показаны в режиме реального времени под любым углом зрения с абсолютно реалистичным освещением, что актуально, например, для кино и компьютерных игр, для реализации "интернет-витрин", создания 3D-контента в электронных энциклопедиях.

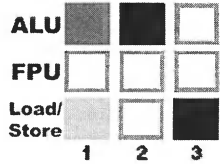
LFM представляет собой программную технологию кодирования, сжатия и визуализации трехмерных объектов, основанную на обработке реальных изображений. По словам Сергея Молинова, представителя нижегородского Центра и одного из разработчиков новой технологии,

LFM дает точное моделирование световых бликов на поверхностях трехмерных объектов независимо от их сложности и существенно сокращает время на разработку интерактивных пространственных сцен, поскольку берет на себя большую часть работы по воссозданию световых эффектов.

Суть технологии состоит в том, что поверхность трехмерного объекта представляется состоящей из участков, каждому из которых соответствует определенное значение функции освещенности (light fields). На основе компактного представления этой функции и происходит ее быстрое отображение современными графическими картами. Созданные LFM-модели обладают возможностью пошагового улучшения качества визуализации, в силу чего пригодны для передачи по сети.

- Основные этапы технологии LFM:
1. Объект фотографируется с

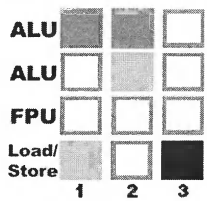
Предположим, что процессор оснащен HT, и попытаемся предложить ему выполнить одновременно два этих потока. Темно-серые прямоугольники указывают на выполнение инструкций первого потока, светло-серые — на выполнение инструкций второго потока, а черные — на конфликт, когда на одно устройство пришли сразу две инструкции из разных потоков. Ясно, что параллельного выполнения потоков не получилось. Процессор выполняет их медленнее, чем если бы выполнял без HT. В чем же дело? А в том, что на выполнение процессору были отправлены два однотипных потока: оба состоят из операций загрузки/сохранения и сложения. Если бы параллельно запускалось "целочисленное" приложение и приложение, работающее с плавающей точкой, то ситуация была бы иной. Из этого примера ясно, что эффективность HT сильно зависит от вида нагрузки на ПК, то есть от конкретного его использования.



— на выполнение инструкций второго потока, а черные — на конфликт, когда на одно устройство пришли сразу две инструкции из разных потоков.

Добавим в процессор второе целочисленное устройство и посмотрим, как потоки будут выполняться обоими устройствами.

В этом случае единственный конфликт случился только на последней операции. Кста-



ти, данный гипотетический процессор в чем-то похож на Pentium 4 (он содержит три целочисленных устройства:

два ALU и одно медленное целочисленное устройство для циклических сдвигов).

Естественно, мало смысла просто добавлять в процессор дублирующие исполнительные устройства для увеличения производительности от использования HT. Это крайне дорого с точки зрения занимаемого на кремнии пространства. Поэтому наряду с введением новых элементов в архитектуру кристалла Intel предлагает разработчикам оптимизировать свои программы под HT.

Так, используя новые инструкции, можно приостановить работу одного из логических процессоров и тем

самым увеличить производительность приложений, которые не выигрывают от HT. Приложение не станет работать медленнее. Система будет работать на одном логическом процессоре, и производительность останется такой же, что и на однопроцессорных компьютерах. А когда приложение сочтет, что от HT будет выигрыш в производительности, второй логический процессор просто возобновит работу.

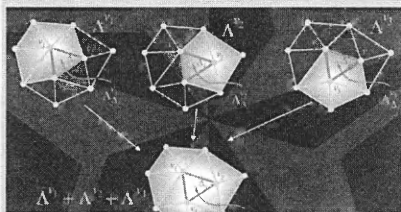
Где это работает

Сегодня наибольший прирост в производительности от использования HT наблюдается в серверных приложениях, что объясняется разнообразием выполняемых процессором операций. Сервер баз данных при включенной опции Hyper-Threading может работать на 20—30% быстрее. Чуть меньший прирост производительности наблюдается на веб-серверах и в других сферах.

Из пользовательских приложений, оптимизированных под HT (разделенных на потоки), значительный выигрыш в производительности могут дать Microsoft Windows XP, Adobe Photoshop и др.

разных планов в условиях не меняющейся освещенности (несколько сотен снимков), этим устанавливается точное соответствие между изображением и геометрией объекта.

2. Каждое изображение (освещенная поверхность объекта) разбивается на элементарные участки — формируется множество изображений каждого участка.



3. Для каждого фотоснимка путем нормализации количества элементарных изображений создается матрица, строками которой являются элементарные изображения, а столбцами — изображения.

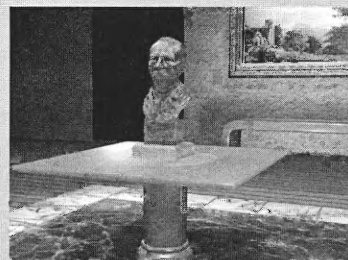
4. Путем параметризации функции освещенности формируется

матрица освещенности (наблюдаемости) элементарных изображений.

5. Полученная четырехмерная функция для каждого элементарного изображения аппроксимируется двумя двумерными функциями (произведением или суммой), представляющими собой изображение объекта. Такая аппроксимация при достаточной точности дает высокую степень сжатия — 100:1.

6. Двумерные функции можно дополнительно сжать как обычные изображения в 80—100 раз.

Визуализация (восстановление функции освещенности) сводится к операциям перемножения и наложения изображений, описываемых двумерными функциями, и достаточно просто реализуется современными графическими картами.



Трехмерное изображение, созданное с помощью технологии LFM

К основным направлениям дальнейшего развития LFM, по мнению разработчиков, следует отнести создание моделей, учитывающих переменную освещенность объектов, что позволит осуществлять их анимацию, и добавление в LFM поддержки прозрачности, что позволит воссоздавать 3D-объекты со сложной геометрией (деревья и т. п.);

Intel предлагает разработчикам ПО пакет Open Light Fields (OLF) с открытым исходным кодом, реализующий технологию LFM. Руководство компании надеется, что первые продукты, созданные на основе LFM, появятся через год-полтора, а в будущем она станет стандартом.

Подробнее о LFM см. на <http://www.intel.com/research/mrl/research/LFM>.

Владимир Буслаев



Сертификат
качества разработки,
проектирования
и производства
ISO-9001
Вся продукция
K-Системс
сертифицирована

Только в том случае,
если у Вас есть компьютер

K-SYSTEMS Irbis Favorite

на базе процессора Intel® Pentium® 4 -

У Вас всегда найдется время для творчества

Использование K-SYSTEMS Irbis Favorite

на базе процессора Intel® Pentium® 4

повышает производительность

при работе с программами для обработки

аудио, видео и сложными графическими

приложениями на 50% по сравнению

с системами предыдущего поколения!

ЭКОНОМИТ ВРЕМЯ

НАСТОЯЩИХ

ПРОФЕССИОНАЛОВ

Санкт-Петербург

ул. Маяковского,

дом 24,

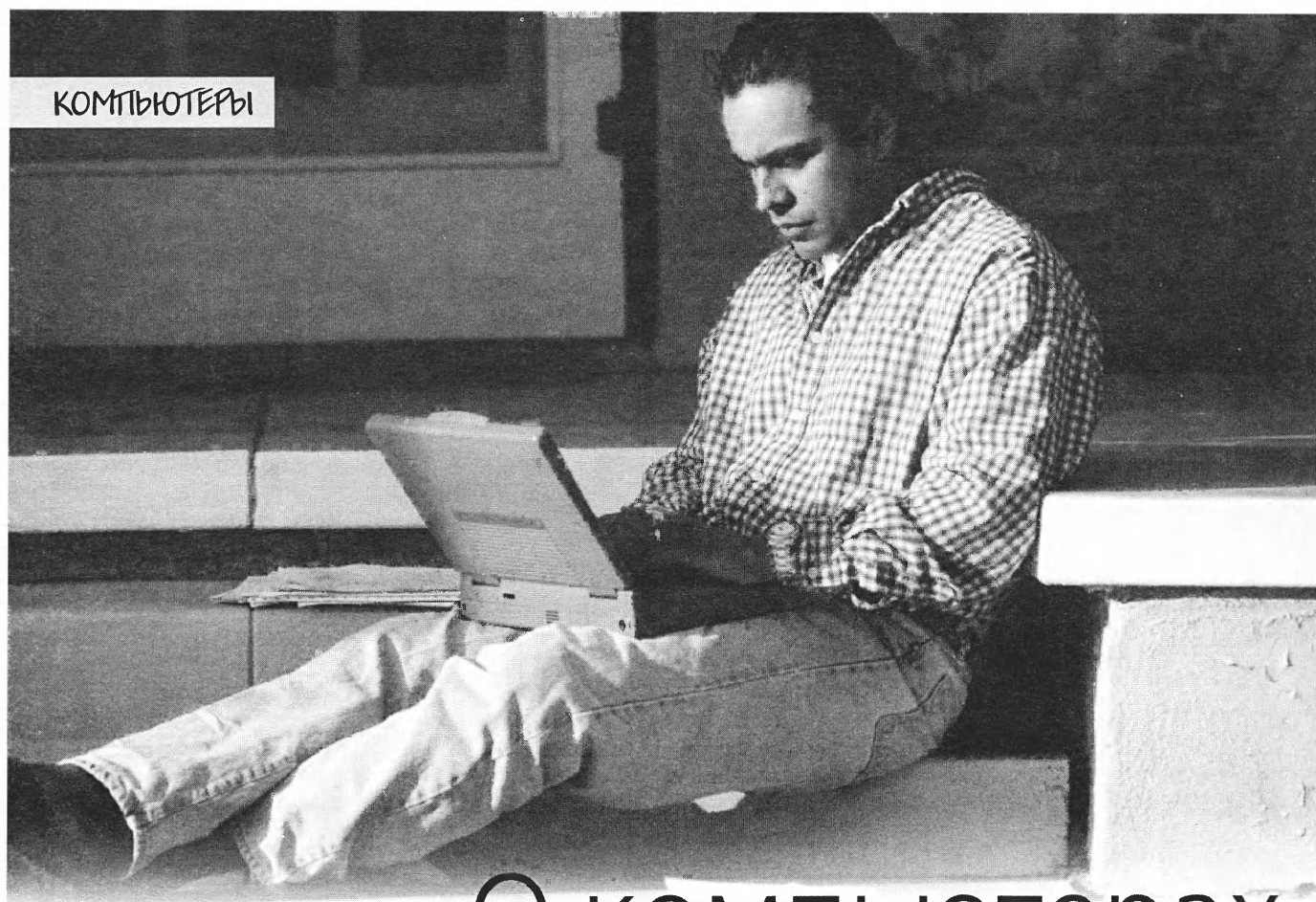
(812) 327-6556,

279-1909

sales@k-systems.spb.ru

www.k-systems.ru

Intel, логотип Intel Inside и Pentium - зарегистрированные товарные знаки Intel Corporation и его филиалов в США и других странах



Михаил Кузин

О компьютерах ИДЕАЛЬНЫХ и ОПТИМАЛЬНЫХ

Темпы прогресса в области компьютерных технологий никогда не снижаются, и уследить за появлением всех новинок очень сложно. Даже профессионал иногда не в состоянии подобрать необходимую конфигурацию компьютера, что уж говорить о тех, кто про компьютер знает только из фильмов да от своих друзей. Конечно, всегда можно положиться на совет продавца в магазине, но при этом надо учитывать два очень серьезных момента. Во-первых, продавцы в магазине не всегда отличаются высокой квалификацией. Чаще всего это просто люди, неплохо разбирающиеся в компьютерах (именно неплохо). Во-вторых, задача продавца — продать как можно больше компьютеров по самой высокой цене, так как от этого напрямую зависит его зарплата, и, таким образом, положившись на совет продавца, вы можете потратить лишних \$150—200.

Хотите иметь компьютер идеальной конфигурации? Нет ничего проще. Выбирайте самые новые и самые дорогие комплектующие, и получите компьютер своей мечты. Но учтите: то, что сегодня было самым новым, завтра становится уже старым, — такова компьютерная “диалектика природы”. Так что тем, кто не стремится к роли калифа на час, лучше задуматься не об идеальной конфигурации, а об оптимальной. Одно из главных правил выбора компьютера: “Пренебрегать ничем нельзя, он должен быть сбалансирован, иначе даже второстепенная вещь может затормозить всю систему”.

Разумеется, оптимальная конфигурация компьютера зависит от тех задач, которые вы на него возлагаете: в зависимости от назначения к компьютеру предъявляются разные требования (учитывая аудиторию журнала, я остановлюсь только на тех компьютерах, которые предназначены для персонального пользования или небольших сетей). По этому основополагающему критерию все вариации сводятся к трем типам компьютеров:

- очень надежно, но умеренно по производительности (офисный компьютер);

- быстро и не обязательно очень надежно (игровой компьютер для дома);

- быстро и надежно (компьютер для дизайнера, состоящий из brand-компонентов).

На всякий случай внесу ясность: офисный, домашний и дизайнерский — это стандартное разделение компьютеров. Если классифицировать их по двум качествам, надежности и производительности, всего получается четыре варианта сочетаний, но говорить о ненадежном и медленном ПК неразумно, поэтому остается только три. Ну, а слова “офисный”,

“домашний” и “дизайнерский” не стоит понимать буквально. Смысл не в расположении, а в назначении ПК.

Разберемся с оптимальной конфигурацией для каждого варианта.

Офисный компьютер

На офисный компьютер обычно возлагаются такие задачи, как ведение бухгалтерии, составление различных документов, организация быстрой связи между удаленными подразделениями. Для большинства фирм оптимальной будет та конфигурация, которая дает максимальное соотношение цена/качество. Причины очевидны: если маленькая фирма покупает два-три компьютера, то директор вряд ли захочет переплачивать \$200 за то, что у его секретаря файл будет сохраняться на две секунды быстрее. Кроме того, руководство маленьких компаний зачастую довольно скептически относится к новым технологиям и, если они обходятся слишком дорого, предпочтение отдается печатной машинке и ксероксу. Да и крупные компании, покупающие компьютеры большими партиями, все равно переплачивать \$200 за те же секунды не стремятся (помножьте эти \$200 на пятьдесят, а иногда и того больше).

А теперь разберемся, что такое “цена/качество” для офисного компьютера. Что такое цена — понятно всем, а вот о качестве надо поговорить поподробнее.

Первый и самый важный критерий оценки качества офисного компьютера — это надежность. Таким компьютерам приходится работать по десять часов без перерыва, и они не должны давать абсолютно никаких сбоев, так как любой такой сюрприз может уничтожить результаты многодневного труда. В связи с этим лучше выбирать комплектующие от надежных, проверенных фирм, даже если они дороже “безмянных”. Зато вы будете уверены, что даже на трети сутки непрерывной работы компьютер не перезагрузится сам собой, из-за ошибки четности памяти, вызванной некачественным исполнением проводников в плате.

Второй важный критерий — производительность компьютера. Конечно, лишняя секунда, затраченная на проверку орфографии в документе, на работе в офисе не скажется, но если, например, файл с трехмерным макетом будущего загородного дома открывается в присутствии заказчика три минуты, то это может лишить фирму важного контракта. Здесь впору вспомнить известное изречение “Скупой платит дважды”. Так что в офис, наверное, не следует покупать высокопроизводительную систему на базе Intel Pentium 4 2,4 ГГц, но и экономить, покупая морально устаревшие компьютеры на базе Celeron 333, тоже не стоит.

Немаловажный критерий для офисного компьютера — его внешний дизайн. Для придания солидности фирме всю технику в офисе обычно стремятся унифицировать, так как строгость и единообразие дизайна создает впечатление порядка и дисциплины, что положительно сказывается на отношении клиентов к фирме.

Ну, а теперь рискну порекомендовать оптимальную конфигурацию, которая даст приемлемую производительность на стандартных офисных приложениях (Microsoft Office, AutoCad, Adobe PhotoShop и т. п.).

- Монитор: 17" ЭЛТ или 15" LCD;
- процессор: Intel или AMD с частотой 700—1300 МГц;
- материнская плата: на чипсете Intel или VIA в зависимости от выбранного процессора;
- оперативная память: 256 SDRAM PC-133
- HDD: 40 Gb ATA/100, 5400 rpm
- CD-ROM: 52x
- FDD, клавиатура, мышь — стандартные
- сетевое оборудование и периферийные устройства (принтеры, сканеры и т. д.) выбираются в зависимости от профиля фирмы.

При дефиците средств можно купить компьютер и попроще (хотя много денег на этом не сэкономить), а при больших финансовых возможностях верхнего предела нет, но главное внимание надо уделять опе-

ративной памяти, жесткому диску и процессору.

Домашний компьютер

Домашний компьютер для 50% пользователей — это устройство для игр и просмотра фильмов, для 40% — средство связи с внешним миром и только для 10% — рабочий инструмент. Но те, кто использует компьютер преимущественно для выхода в Интернет, обычно тоже не прочь посмотреть фильмы и послушать музыку на компьютере. Кроме того, в последнее время, с появлением дешевых сканеров и сравнительно недорогих цифровых фотоаппаратов, компьютер многие начинают использовать для занятий фотографией. Но снимки мало сохранить в фотоальбоме, иногда их хочется отредактировать...

По конфигурации домашний компьютер отличается от офисного так же, как белое от черного. Если у офисного компьютера на весах надежность — производительность бесспорно перетягивает первое, то для домашнего компьютера важнее производительности ничего нет. Не страшно, если придется ставить заново систему каждые три месяца, это нормальное явление, но вот если вместо фильма на экране вы видите слайд-шоу, возникает непреодолимое желание выкинуть этот компьютер. Дизайн домашнего компьютера — дело личного вкуса. Главное, чтобы все “красиво” работало.

Все перечисленное диктует достаточно высокие требования к производительности компьютера в целом. Оптимальная модель домашнего компьютера будет, конечно, существенно дороже офисного. На данный момент она может выглядеть примерно так:

- Монитор: 17" ЭЛТ
- Процессор: Intel или AMD с тактовой частотой 1000—1600 МГц
- Материнская плата: на чипсете Intel или VIA в зависимости от выбранного процессора
- Оперативная память: 256—512 DDR RAM или RDRAM
- HDD: 40—60 Gb ATA/100, 5400 rpm

- CD-RW: 32x/12x/48x или DVD ROM: 16x

- Видеокарта: на основе GeForce 2 или GeForce 4 MX440 с 64 Mb видеопамати

- Звуковая карта: SB Live! 5.1 + акустическая система 4.1 или 5.1

- Клавиатура с дополнительными мультимедиа-кнопками, мышь с колесиком

- Модем: US Robotics 56k PCI или ZyXEL OMNI 56k ext

Данную модель, конечно же, очень дешевой не назовешь, но зато она обеспечит отличную производительность для любых задач в течение еще трех-четырёх лет, и дополнительные расходы потребуются разве что на увеличение оперативной памяти.

Компьютер для дизайнера

Профессионалы называют такой компьютер графической станцией. Этот компьютер — вещь серьезная. Самое важное для него, конечно, производительность. Причем, если в предыдущих двух случаях она зависела в основном от частоты процессора и размера оперативной памяти, то здесь основной тормоз системы — дисковая подсистема. Дело в том, что все серьезные графические программы создают свои временные файлы на жестком диске и постоянно работают с ними. Соответственно, если на одном и том же жестком диске расположены файл подкачки Windows, временные файлы графической программы и прослушиваемые музыкальные файлы, то работа такой системы будет вызывать только отрицательные эмоции или действовать как снотворное, в зависимости от характера пользователя. Идеальным решением было бы приобретение материнской платы с интегрированным RAID-контроллером.

Отдельно надо упомянуть об устройстве CD-RW. Сейчас разница по цене между приличным CD-ROM и CD-RW не очень большая (\$20—30), так что если он сегодня и не нужен, стоит подумать о завтрашнем дне, — мы же не на один месяц компьютер покупаем. Самое простое и очевидное решение — купить привод CD-

RW 24x/12x/48x от известного производителя. Но сейчас прогресс в этой области идет быстрее, чем хотя бы полгода назад. Появились модели с рабочими скоростями 48x/24x/48x. Хотя дорогая вещь, но стоит раскошелиться.

Ну, а теперь самое важное — конфигурация:

- * Монитор: 19" или 21" ЭЛТ

- * Процессор: Intel или AMD с частотой 1,6—2,2 ГГц

- * Материнская плата: на чипсете Intel или VIA в зависимости от выбранного процессора

- Оперативная память: 1024 DDR RAM или RDRAM

- HDD: 40—60 Gb ATA/133, 7200 rpm, связка из двух HDD с RAID-контроллером

- CD-RW: 48x/24x/48x

- Видеокарта: на основе GeForce 4 Ti 4600/4400 или ATI Radeon 8500 с 128 Mb видеопамати

- Звуковая карта: в зависимости от музыкальных пристрастий дизайнера (от встроенной AC'97 до Creative Audigy Platinum)

- Клавиатура с дополнительными мультимедиа-кнопками, мышь с колесиком, графический планшет

- Модем: US Robotics 56k PCI или ZyXEL OMNI 56k ext

Конкретнее о комплектующих

А теперь разберемся в некоторых характеристиках компьютера, перечисленных выше.

Оперативная память

SDRAM, DDR RAM, RDRAM — это типы оперативной памяти, различающиеся многими параметрами. SDRAM — самая медленная, но и самая дешевая, а RDRAM — самая быстрая, но и самая дорогая. Советую выбирать модули оперативной памяти от известных производителей (Samsung, Kingston, Transcend, Kingmax и т. д.). Модули NoName дешевле, но через год ваш компьютер начнет "виснуть" так часто, что вы пожалеете, что его купили.

Жесткие диски (HDD)

В основном они различаются скоростью вращения (rpm, чем она больше, тем меньше время поиска) и

скоростью передачи данных (ATA/100, ATA/133). Сразу отмечу, что даже купив HDD 7200 rpm ATA/133 вместо HDD 5400 rpm ATA/100, большого прироста производительности вы не получите, так что все зависит от ваших финансовых возможностей. Учтите также, что чем больше rpm у жесткого диска, тем больше он греется, а это может привести к сбоям.

Модемы

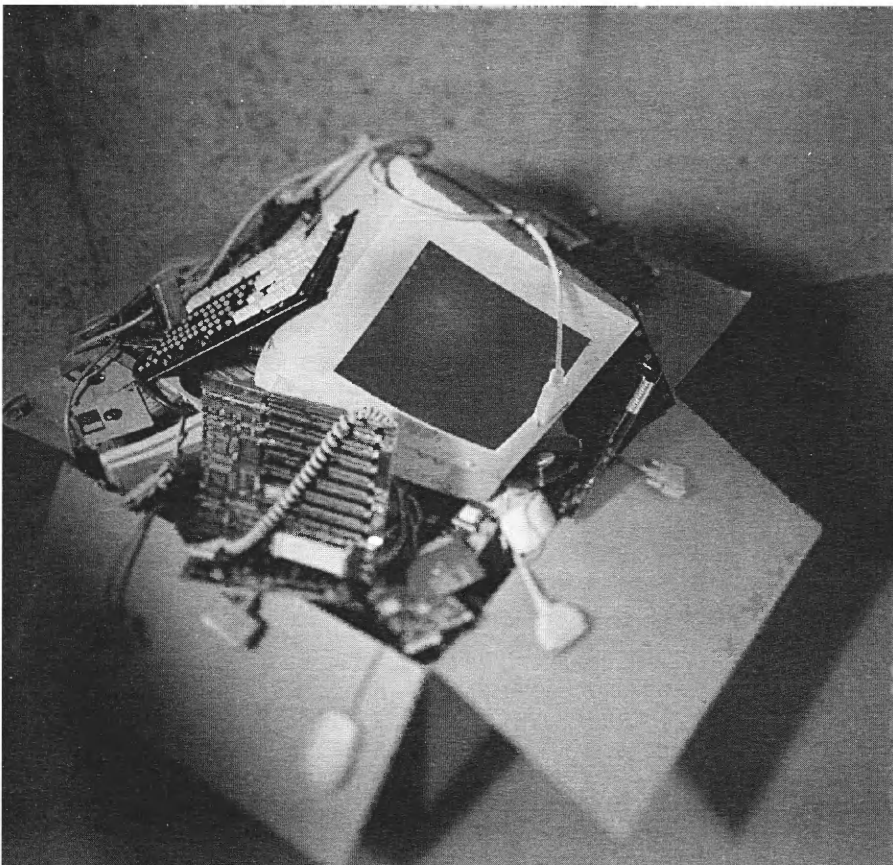
В основном модемы различаются по двум параметрам: внешний — внутренний и soft — hard. Внешний модем, в отличие от внутреннего, не занимает дополнительный PCI- или ISA-разъем на материнской плате. К тому же на нем есть светодиоды, которые показывают, работает модем или нет. С другой стороны, для внешнего модема нужен дополнительный блок питания, он занимает место на столе и стоит дороже.

Soft-модемы передают центральному процессору все функции по приему и передаче данных, их компрессии, проверке правильности приема и передачи и т. д., снижая тем самым производительность всей системы. Hard-модемы выполняют данные действия самостоятельно, поэтому работают быстрее, однако стоят дороже.

На прощание

Поскольку выбор компьютера зависит от конкретной ситуации, я сознательно старался давать лишь рекомендации общего характера, не углубляясь в детали. Например, если дома компьютер нужен в основном для набора текстов, а самой ресурсоемкой игрой будет тетрис или пазьянс-косынка, то 256 Mb RDRAM и GeForce MX 440 будут явно лишними.

Ну, и, конечно, решающим критерием выбора компьютера всегда был и остается объем финансовых ресурсов. Если денег много, то и в офис можно поставить компьютер с конфигурацией графической станции. И наоборот, если мало, не стоит отчаиваться: даже самый серьезный графический редактор, пусть медленно, но все же будет работать на компьютере с процессором Intel Celeron 366 МГц.



ЧТО И КАК УСТАРЕВАЕТ

Николай Богданов-Катьков

Сейчас риск купить технику, которая устареет в ближайшем будущем, велик как никогда.

Некотрые компьютерные фирмы проводят семинары для покупателей. На них собирается самая разная публика — от старшекласников, которым родители собираются купить компьютер, до менеджеров конкурирующих фирм (последних даже больше). Общй итог рассуждений на таких семинарах сводится к совету: "Покупайте самую современную технику, которая устареет не скоро". Вопрос не только в том, сколько она будет стоить: именно самое современное "железо" дешевеет быстрее всего. Но дешевеет оно, пока лежит в магазине, а уже купленный компьютер ОБЕСЦЕНИВАЕТСЯ!

Что и как устареваеа

Пару лет назад в уважаемом журнале "PC Magazine RE" я прочитал буквально следующее: "Пользователь, довольствующийся 14-дюймовым монитором, достоин только сожаления...". К тому моменту мой 14-дюймовый отпраздновал пятилетний юбилей трудовой деятельности. Выкидывать? Жалко, отремонтировать его не приходилось ни разу, да и яркость не снизилась.

Пока я раздумывал, в другом, не менее уважаемом журнале "Мир ПК" появилась статья, автор которой (Стив Басс) делился восторженными впечатлениями о новом мониторе — 19-дюймовом Mitsubishi. Действительно, монитор отменный, но по-

путно автор упомянул, что за последние два года "похоронил" два монитора, 15- и 17-дюймовый, оба Sony. Кто же на кого должен смотреть с сожалением?

Нет, мониторы не устаревают в полном смысле слова. Монитор, не вызывающий усталость глаз, не распространяющий вокруг себя вредных излучений, может работать сколько угодно. Другое дело, если я займусь, например, дизайном. Тогда мне понадобится монитор покрупнее и с более высокими параметрами изображения.

В одной из недавних статей я рассматривал различные критерии старения компьютерной техники. Компьютер устареваеа, когда:

- 1) появляются новые, заведомо более мощные компоненты,
- 2) появляются программы, требующие больших системных ресурсов,
- 3) появляется новое оборудование, несовместимое со старым.

Очевидно, что если руководствоваться первым критерием, новый компьютер придется покупать чуть ли не каждый месяц.

С программами дело сложнее. Если программа "тормозит", тут никуда не денешься: либо новый компьютер, либо апгрейд. Но много ли за последние два года появилось программ, для которых недостаточно добротного компьютера двухлетней давности? Конечно, игры могут быть сколь угодно ресурсоемкими, но "развлекательный" компьютер — статья особая.

Хуже всего третий случай, когда оказывается, что новое "железо" несовместимо со старым. Вспомните, когда трехдюймовые дисководы вытеснили старые пятидюймовые, внезапно большую ценность обрели компьютеры, оснащенные обоими дисковыми: они позволяли переписывать с пятидюймовых дискет на трехдюймовые все, что копилось годами и представляло немалую ценность.

Вот характерный пример из жизни. Недавно в редакцию "Магии ПК" обратился читатель, купивший новый компьютер. У него был сканер с интерфейсом SCSI, который успешно

работал со старым компьютером, но к новому его подключить не удалось: интерфейсная карта, на которой находится SCSI-контроллер, вставлялась в слот ISA. Шина ISA медленная, еще стандарт PC99 рекомендовал отказаться от нее. Системные платы с шиной ISA исчезли из продажи больше года назад. Что делать?

Ситуация достаточно распространенная. Под слот ISA в свое время было выпущено много комплектующих — модемы, звуковые карты. Даже если они вполне работоспособны, в новый компьютер их не установишь — некуда.

Купить обычный контроллер SCSI для слота PCI? Тоже не выход. Стандартов SCSI существует больше десятка, более ранние и более поздние. Все они друг с другом несовместимы. Сейчас в продаже контроллеры Ultra 160 SCSI, встречаются и более ранние. Сканеру же нужен скорее всего контроллер SCSI-II или же Wide Ultra SCSI, очень старые, каких сейчас и не купишь.

Единственный совет, который можно было дать, — поискать интерфейсную карту для слота PCI в фирмах, занимающихся апгрейдом и скупкой старого "железа".

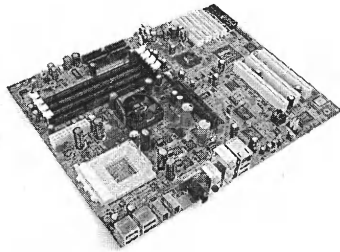
И такое случается нередко. Старое, но вполне работоспособное оборудование приходится выбрасывать только потому, что оно несовместимо с новым "железом". Отчасти это закономерно. Старые, более медленные интерфейсы должны уступить место современным скоростным.

Революции и эволюция

В прессе много говорят о новой системной плате ABIT AT7. Плата под Socket A, поддерживается быстрая память DDR333 до 4 Гбайт, есть USB 1.1 и 2.0, всего десять (!) портов. Контроллер IEEE 1394 (FireWire), шестиканальный звук, AGP 4x, RAID-массив до 12 винчестеров ATA/100. По оценке фирмы, продвигающей эту плату, — "очень неплохой вариант для серьезных пользователей, конфигурирующих серьезную и универсальную машину".

Все это, конечно, здорово, но...

На плате нет ни разъемов PS/2, ни COM-портов, ни даже принтерного порта LPT. На компьютер, собранный на этой плате, можно установить Windows 2000, XP или ME, но старый привычный Win'98 уже не подойдет: USB 2.0 он не поддерживает.



Революционное новшество? В каком-то смысле, да. Можно ожидать, что плата с поддержкой скоростных комплектующих и кучей современных интерфейсов устареет не скоро. К тому же в соответствии со стандартом PC2001 на плате и не должно быть устаревших COM, LPT и PS/2 портов. Всю периферию надо подключать через USB или IEEE 1394. Но насколько удобно будет пользоваться компьютером без принтерного порта?

Большинство современных принтеров выпускается с портом USB или с обоими сразу: LPT и USB. Но еще год назад большинство лазерных принтеров выпускалось только с портом LPT. К принтерному порту подключают не только принтеры, но и сканеры. С ними та же картина. Сканеры с интерфейсом USB два года назад почти не предлагались. Обычно принтер, струйный или лазерный, служит не год, не два, а значительно дольше. Сканер — тем более.

Что же получается? Купил новый компьютер — покупай к нему новые

Принтер даром?

Этой осенью цена младшего струйного принтера в модельной линейке Lexmark, модели Z-13, снизилась примерно до уровня цены трехцветного картриджа к нему. Поскольку принтер продается в комплекте с картриджем, можно считать, что сам он не стоит ничего. Даже меньше чем ничего: известная компьютерная фирма предлагает купившим принтер Z-13 в подарок пачку бумаги. Та же фирма предлагает принтер в подарок тому, кто купит компьютер...

принтер и сканер! Вот уж действительно, "пора менять железо...", причем все сразу. "Что ж, — скажет читатель, — пусть плата несовместима со старьем, зато сама она устареет не так скоро".

А что в ней устареет раньше всего? Обратим внимание на дисковый интерфейс. Ну, RAID-массив — вещь полезная, хотя дюжину винчестеров не упрятать даже в BigTower, это ясно. Но ATA/100 — какова будет его судьба через год-другой? 100 Мбайт/с — это общая пропускная способность, которая делится на все устройства IDE. Реальная скорость передачи данных для современных винчестеров — 25—35 Мбайт/с, некоторые модели дотягивают до 45. Получается, что если RAID-массив будет состоять более чем из трех винчестеров, ни один из них не сможет работать на своей максимальной скорости.

Сейчас появились винчестеры и системные платы ATA/133. У этого интерфейса не только выше пропускная способность. Стандарты ATA/66 и 100 поддерживают 28-разрядную адресацию данных, и объем винчестера может достигать примерно 137 Гбайт. В ATA/133 адресация 48-разрядная, и максимальный объем винчестера составляет 150 млн Гбайт. Плотность записи на одну пластину достигла 40 Гбайт, и все фирмы начинают выпускать винчестеры объемом 160 Гбайт. На очереди дальнейший рост.

ATA/133 будет последним параллельным дисковым интерфейсом: разработан и внедряется последовательный интерфейс SerialATA. Его пропускная способность составляет ныне 150 Мбайт/с, а в ближайшем будущем возрастет до 300 и 600.

Именно на SerialATA делают сейчас ставку ведущие производители комплектующих. Для обеспечения совместимости с параллельными интерфейсами предполагается выпускать переходники (точнее, контроллеры). Однако итоговая пропускная способность этой "связки" не может быть выше, чем у наиболее медленного устройства, то есть 100 Мбайт/с. К тому же от поддержки RAID придется отказаться.

Получается, что года через два эта системная плата будет обречена на сосуществование с винчестерами, которые к тому времени будут безнадежно устаревшими. Это очень сильно ограничит возможность апгрейда.

Наконец, взамен графического порта AGP 4x уже разработан AGP 8x. Появились и первые видеокарты с ним, и соответствующие чипсеты для системных плат. Нет, для большинства применений порта 4x вполне достаточно, но все же в ближайшем будущем видеокарты верхнего ценового диапазона будут рассчитаны именно на AGP 8x.

Все эти недостатки могут показаться несущественными сейчас, но они встанут в полный рост через год-другой, когда возникнет вопрос апгрейда.

В лабиринтах апгрейда

Многие покупают компьютеры в расчете на то, что за месяц до конца гарантийного срока можно будет заменить процессор и видеокарту на более скоростные, которые к тому времени значительно подешевеют. В этом есть резон. Действительно, современнейшая видеокарта за год может подешеветь раза в два, а процессор — даже в три-четыре.

Однако и тут есть свои "подводные камни". Первые процессоры Pentium 4 появились в ноябре 2000 года. К лету следующего года первые

модели с тактовыми частотами 1,3—1,5 ГГц значительно подешевели, и компьютеры на их основе стали покупать все чаще. Прошел еще год, и весной 2002 года их владельцы, рассчитывавшие на апгрейд, поняли, что Intel подложила им свинью...

Дело в том, что первые Pentium 4 выпускались с процессорными разъемами Socket-423. Выпустили еще некоторое количество таких же процессоров на 1,7 и 1,8 ГГц, но затем большая их часть выпускалась уже с разъемами Socket-478, как и все последующие процессоры. Соответствующие разъемы были и на системных платах.

При апгрейде можно заменить процессор Pentium 4 частотой 1,4 ГГц на 1,8 ГГц (если посчастливилось его достать!), но заметной прибавки в скорости это не даст. Реальную прибавку обеспечивают процессоры, начиная с 2 ГГц и с более высокой частотой системной шины (533 МГц вместо 400), а для них нужна уже другая системная плата.

Заменить одновременно процессор и системную плату? Можно. Но тут другой подвох: к весне этого года все процессоры и платы с Socket-423 числились устаревшими. Многие фирмы под разными предлогами вообще отказывались принимать старые комплектующие или предлагали за них гроши.

Снова "революционная" ситуация, но принципиально иная. В данном случае нельзя сказать, что бо-

лее совершенный процессорный разъем вытеснил старый. В самом деле, где это видано, чтобы новый, суперсовременный, широко разрекламированный процессор устарел через год? Просто чтобы заставить пользователей чаще покупать новые компьютеры, надо искусственно ограничить возможности апгрейда...

Вернемся к примеру с системной платой. Если не учитывать восемнадцати (!) разновидностей интерфейса SCSI (он редко применяется в настольных ПК), сейчас существуют:

- две спецификации USB,
- две спецификации шины PCI,
- три дисковых интерфейса,
- две спецификации AGP,
- три стандарта памяти DDR SDRAM.

Системная плата — тот узел компьютера, который обеспечивает взаимодействие всех остальных устройств, и для нее поддержка различных стандартов наиболее важна. При таком разном единственном грамотным решением при конструировании системных плат будет максимальная универсальность. Ясно, что плата, в которой отсутствуют старые порты и имеется устаревающий дисковый интерфейс, этому критерию не удовлетворяет.

Беда в том, что это общая проблема: сейчас риск купить технику, которая устареет в скором будущем, велик как никогда.

У богатых свои причуды

Хочу в Великобританию. Нет, не жить, а по помойкам пройтись. Вернусь обратно в Россию, стану миллионером.

Представьте себе, приходите вы на свалку и собираете принтеры, сканеры, копиры, факсы, MP3-проигрыватели, которым нет и двух-трех лет "отроду". Самый настоящий Клондайк. Вы спросите, в чем тут дело? Все очень просто. В Великобритании считают, что экономически выгоднее купить новый принтер, чем покупать картридж к старому. Действительно, за деньги, потраченные на картридж для старого цветного

принтера, порой можно купить новый принтер.

Еще пример — устройства, несовместимые с Windows XP (а эта ОС — самая распространенная в Англии). Так вот, если к Windows XP что-то не подходит, то это ЧТО-ТО просто выбрасывается. Нет, деньги там тоже умеют считать. Но в Великобритании невозможно ни продать, ни, к примеру, подарить школам технику даже трех-четырёхлетней давности. Это старье там просто никому не нужно.

К тому же при малейшей неисправности житель туманного Альби-

она к мастеру не идет. Это у нас дешевле мастера вызвать, чем новую модель купить. Там работа стоит куда дороже. Действительно, если сломался видеомагнитофон, зачем его чинить, если проще купить новый, навороченный аппарат, который и простое видео крутит, и DVD, и еще много чего. Вот и выбрасывают исправные вещи. Да что говорить, если по данным компании Eurosource, которая занимается переработкой выброшенных мобильных телефонов, только никому не нужных сотовых телефонов в Великобритании — порядка 80 миллионов! Это при населении 60 млн человек...

Нам бы их проблемы...

Сергей Артюхов

Энциклопедия современного компьютера

МАТЕРИНСКИЕ ПЛАТЫ

Александр Дудкин

Материнская плата, называемая также системной, является основой системного блока, определяя вместе с процессором архитектуру и производительность компьютера. Она связывает воедино компоненты ПК, управляет синхронизированной работой всех его подсистем. Несмотря на большое разнообразие в исполнении, на любой материнской плате обязательно устанавливаются такие компоненты, как процессор и сопроцессор; память ROM, RAM и SRAM; схемы ввода/вывода; схемы интерфейсов и шин, кварцевый генератор, схемы управления напряжением. Кроме того, возможно наличие большого количества интерфейсных контроллеров, контроллеров (адаптеров) портов. Главным набором микросхем в современных материнских платах является чипсет, который управляет работой всех остальных контроллеров и компонентов, согласуя их работу во времени. Чтобы понять, как все это работает и взаимодействует, необходимо разобраться с физической структурой и электрической начинкой материнских плат.

Из чего она состоит?

Печатная плата

Материнская плата (МП) представляет собой печатную плату (PCB, Printed Circuit Board), на которой смонтированы все элементы. Эта плата обычно состоит из четырех слоев (плоских камедевых пластин), между которыми находятся соединительные элементы цепи — пропаянные дорожки. Наружные слои, верхний и нижний, являются сигнальными, а внутренние используются для заземления и разводки питания (этим достигается коррекция и защита сигнала).

МП для двухпроцессорных систем, а также для процессоров с более чем 425 контактами — шестислойные. Pentium 4 (478 контактов) и Athlon XP (462 контакта) потребляют большое количество энергии, поэтому качественная разводка МП для них особенно важна. Шестислойная разводка предотвращает перекрестные помехи между сигнальными слоями. Такие платы могут иметь три или четыре сигнальных слоя, один для заземления и один или два для питания. Шестислойные МП (особенно под Pentium 4) крайне дороги.

Форм-факторы

Основные форм-факторы (типоразмеры) МП, как и блока питания, — AT и ATX. Компьютеры IBM PC AT с 1983 по 1997 год имели форм-фактор AT (30x35 см), который затем был уменьшен и стал называться Baby-AT (BAT, 22x33 см). Последний использовался для установки всех процессоров от 8086 до Pentium II, но в продаже такие корпуса можно увидеть и сейчас. В платах AT обычно использовался полноразмерный 5-контактный разъем DIN для клавиатуры (также назывался AT), совместимый с IBM XT. Сегодня самый распространенный форм-фактор — ATX (30,5x24 см), имеющий ряд новшеств в сравнении с AT:

- Все внешние разъемы располагаются в два «этажа» и правого края. Для них в корпусе предусмотрено большое прямоугольное окно, закрываемое алюминиевыми заглушками.
- Процессор находится под блоком питания, что создает дополнительное охлаждение, хотя и малоэффективное.
- Разъемы контроллеров Floppy и IDE располагаются близко к ячейке для крепления винчестера и диске

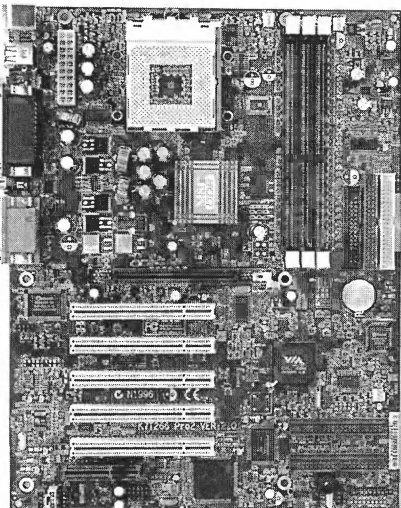
ководов, а модули памяти легкодоступны.

Его разновидность — Micro ATX (mATX, 28,4x21 см). Оба варианта МП можно поставить в один и тот же корпус, а mATX еще и в "маленькие" корпуса Midi и Mini Tower. Основное отличие mATX, кроме размеров, — в уменьшенном количестве слотов для памяти и для дополнительных карт и отсутствии ISA, что позволяет существенно сократить стоимость этой МП.

Существуют также стандарты LPX и NLX, но они используются только в фирменных моделях компьютеров.

Поддержка процессора

Безусловно, процессор физически и электрически должен быть совместим с МП. Сейчас в абсолютном большинстве CPU используется разъем Socket (Slot-1 использовались под Pentium II, Slot-A — под Athlon Classic, K75 и K76). Правда, Intel ведет политику постоянной смены типов разъемов: Socket 370 —> Socket 423 —> Socket 478, причем Socket 370 имеет несколько электрически несовместимых разновидностей — их принято различать по типам корпусов процессоров: PPGA, FCPGA и FCPGA2. Первый из них не поддерживает ядро Coppermine и Tualatin, второй — только Tualatin. Компания AMD придерживается политики большей заботы о покупателе и использует только один разъем Socket A для всех процессоров, начиная с Thunderbird и Spitfire.



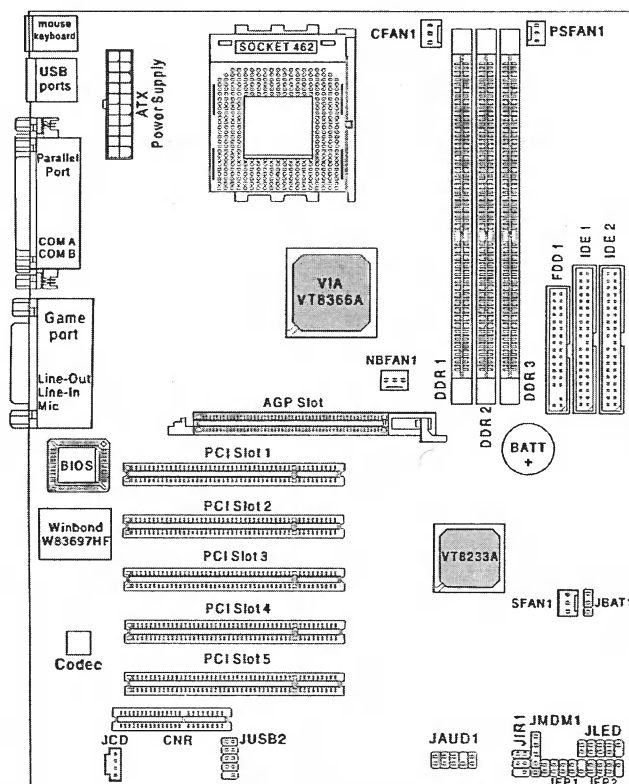
Чем она питается?

На материнскую плату подается напряжение от блока питания (БП): в случае AT форм-фактора это 5 В, в случае ATX — 3,3 В, поэтому в МП форм-фактора AT используется несколько микросхем VRM (Voltage Regulator Module) для понижения напряжения, а в ATX такой модуль один. Различные компоненты, установленные на МП, питаются от разного напряжения: +5 В (например, чип BIOSa, часы реального времени, контроллер клавиатуры, чипы DRAM, большая часть контроллеров, коннекторы) и +3,3 В (кэш L2, чипсет, чипы SDRAM, AGP). Двигатели накопителей и кулеры питаются от +12 В. Но БП вырабатывает также и отрицательные напряжения. В современных компьютерах они не используются, а раньше напряжение —5 В подводилось к шине ISA и использовалось для питания старых контроллеров НГМД; напряжения +12 и —12 В подводились к шине ISA для питания различных адаптеров и контроллеров последовательных портов.

Блок питания помимо прямого назначения контролирует включение и выключение компьютера, точнее, не позволяет компьютеру включиться при нестандартном напряжении. Перед запуском системы выполняется проверка выходного напряжения с помощью сигнала Power_Good +5 В, который вырабатывается БП через 0,1—0,5 с после запуска компьютера, если проверки прошли успешно. Этот сигнал подается на МП, где микросхема тактового

генератора формирует сигнал начальной установки CPU. При отсутствии сигнала Power_Good эта микросхема будет подавать сигнал начальной установки, не позволяя компьютеру включиться до установки нормального питания. В некоторых дешевых БП этого сигнала нет, а цепь его просто подключена к +5 В. Учтите, использовать такие БП — значит подвергать смертельной опасности и МП, и процессор.

Блоки питания для корпусов AT и ATX имеют разные принципы подключения к системной плате: AT подключается через два 6-контактных разъема (PS8 и PS9), а ATX — одним 20-контактным разъемом с ключом, исключающим возможность неправильного подсоединения к МП. В AT-корпусах питание 220 В от БП подается на 4-жильный кабель, соединенный с кнопкой питания Power. При нажатии этой кнопки контакты замыкаются, и переменное напряжение возвращается в БП, а оттуда на МП. В корпусах ATX система включается подачей сигнала PS_ON (Power_On), поступающего с 14-го контакта 20-кон-



Материнская плата ATX и ее принципиальная схема

тактного разъема при нажатии кнопки питания, которая не имеет двух положений, как ранее, а замыкается на коннектор PW_BTN на МП, генерируя низкий активный сигнал. При высоком уровне этого сигнала БП выключается. Поскольку этот сигнал подается с МП, он может быть создан ОС или переключателем Power, благодаря чему мы имеем возможность выключать компьютер с помощью Windows.

Еще одним новшеством в ATX стало использование сигнала "ждущего режима" +5 В SB (Stand_by, Soft power), который всегда подает на плату питание малой мощности, даже когда компьютер выключен. Эти два сигнала позволяют, используя APM (Advanced Power Management), в ОС Windows ME/2000/XP кнопкой Power на клавиатуре выключать питание, а кнопкой Sleep — переходить в "ждущий режим" (обратите внимание, в Windows 98 обе кнопки переводят ПК в "ждущий режим").

При подключении устройств к МП необходимо соблюдение полярности. Первый контакт колодки на плате необходимо совместить с первым контактом на шнуре (шлейфе) — он часто помечается красным цветом.

Регуляторы напряжения

Некачественное питание часто приводит к неустойчивой работе и даже выгоранию некоторых элементов (особенно в слотах). Источник питания подает на МП 5 В прямого напряжения, следовательно, для некоторых компонентов требуется регулировка мощности (например, для ядра CPU). Для этого используется дополнительный модуль VRM (модуль стабилизатора напряжения) или плата регулятора напряжения.

Напряжение питания цепей I/O для всех CPU сейчас установлено 3,3 В, а вот с напряжением питания ядра V_{core} (V_{cc}), которое регулируется с помощью VRM, проблем больше. У CPU Athlon Thoroughbred (как и у P4 Northwood) это 1,5 (1,6) В, ядро Coppermine может работать при напряжениях 1,5—1,65 В,

Tualatin — 1,475—1,55 В, Spitfire — 1,4—1,6 В, Thunderbird — 1,6—1,75 В, Palomino — 1,7—1,85 В. Остальные CPU не имеют разброса по напряжениям. Напряжение можно выставить больше указанного диапазона, что используется при разгоне для уменьшения тепловыделения. На всех процессорах, начиная с Pentium MMX, имеется два регулятора напряжения — один для цепи I/O (3,3 В), а другой для питания на ядре процессора, но в платах форм-фактора ATX используется VRM только для ядра процессора.

Для того чтобы схема держала определенный диапазон напряжения, на МП обычно устанавливается набор резисторов. Сейчас на большинстве плат стоит так называемый автодетект (автоопределение), то есть схема сама определяет и распределяет напряжение, ориентируясь по контактам VID [0:4] на процессоре, что исключает потребность в джамперах. На процессорах AMD напряжение лучше выставлять вручную, теперь это легко сделать через BIOS.

Конденсаторы

Регуляторы напряжения не могут реагировать мгновенно на резкие изменения (потребление энергии процессором может меняться мгновенно от низкого к высокому и наоборот), поэтому питающее напряжение "сглаживается" конденсаторами. На МП используются два типа конденсаторов: алюминиевые (они же оксидно-электрические) и танталовые. Последние предпочтительнее, так как оксидно-электрические конденсаторы чувствительны к высокой температуре, имеют особенность высыхать и терять свою емкость.

Важный параметр у конденсаторов (помимо емкости) — это значение ESR (эквивалент последовательности сопротивления). Рост сопротивления ведет к снижению напряжения и, вследствие этого, вырабатывается больше тепла. Следовательно, значение ESR у конденсаторов должно быть как можно меньше.

Генератор тактовых импульсов

Каждый компонент в компьютере работает по синхронизирующим импульсным тактам (clock, CLK), которые выдает тактовый генератор (Clock Generator Chip), но не каждый компонент работает на одних и тех же частотах. Поэтому в компьютерах класса 486 и выше применяется деление опорной частоты генератора для синхронизации шин и внутреннее умножение частоты в процессорах.

За опорную частоту берется частота системной шины (FSB, Front Side Bus, host bus clk), а частота локальных шин образуется с помощью встроенных делителей. Для каждой шины чипсет может поддерживать как один, так и несколько коэффициентов деления. Этот коэффициент обычно меняется через каждые 33 МГц по опорной частоте. Для локальной шины с базовой частотой 33 МГц (PCI) должны поддерживаться коэффициенты соотношения с системной шиной 1/2 (при FSB 66—99 МГц); 1/3 (100—133 МГц) и 1/4 (133—166 МГц), то есть используются делители 2, 3 и 4. Для шины 66 МГц (AGP) должен поддерживаться коэффициент 1, 2/3 и 1/2, а делители соответственно — 1, 1,5 и 2. Наличием нескольких делителей и, соответственно, более широкими возможностями по выбору CPU и его разгону славятся чипсеты VIA и SIS.

То же самое относится и к шинам USB, ISA. В промежутках, где коэффициент неизменен, с ростом FSB растет и частота локальных шин, что характерно для разгона по шине (см. таблицу). Использование нестандартных частот системной шины может привести к некорректной работе или даже к выгоранию устройств, находящихся на локальных шинах.

Частота системной шины	Изменение частоты шины 33 МГц	Изменение частоты шины 66 МГц
66 МГц	33	66
75 МГц	37,5	75
83 МГц	41,5	83
95 МГц	47,5	95
100 МГц	33,3	66,6
110 МГц	36,6	73,3
120 МГц	40	180
133 МГц	44,3	88,6

Для поддержки CPU с различной частотой используются множители частоты: $\text{Частота ядра (Core clock)} = \text{FSB} \times \text{множитель}$.

BIOS и RTC

После установки сигнала начальной загрузки процессора начинает считываться программа по адресу FFFF:0000 (Flash ROM BIOS). В ней содержится информация об устройстве МП, чипсете и прочих параметрах. BIOS может быть записан в ROM, но с 1996 года используются Flash ROM/EEPROM, которые можно перепрограммировать. Этот чип имеет примерно от 512 Кбайт до 4 Мбайт памяти, программируется на заводе и может быть "перепрошит" только специальной утилитой от производителя BIOS. Перезаписываемый чип можно отличить по маркировке: он должен иметь надпись формата 28xxxx или 29xxxx. После включения ПК запускается процесс, называемый "Самопроверка при включении" (POST, Power-On Self-Test): определяется процессор, количество установленной памяти, наличие и работоспособность всех зарегистрированных компонентов, после чего считывается CMOS. По завершении этой операции в порядке, указанном в CMOS, на каждом загружаемом устройстве идет загрузка низкоуровневых драйверов для работы с "железом", и ROM BIOS принимает управление на себя.

Интегральная схема CMOS содержит особые параметры пользователя, которые считываются сразу после того, как определен процессор. Эта схема обычно встраивается в чип часов реального времени (RTC, Real Time Clock), в котором содержится информация о дате и времени. До меню параметров в CMOS можно добраться через специальное меню во время операции POST нажатием клавиши DEL или F1, далее изменения вводятся вручную. RTC и CMOS хранят информацию только до тех пор, пока поступает напряжение из небольшой батарейки 2032 на МП. Если эту батарею отсоединить, информация в CMOS теряется и должна быть введена заново при следующей загрузке.

Контроллеры

Это микросхемы для согласования и обмена данными разных стандартов, таких как IDE, Floppy, SCSI. Типовой состав — двухканальный контроллер IDE с поддержкой режима UltraDMA/100, контроллер FDD, контроллер портов, в том числе USB 1.1 с пропускной способностью 12 Мбит/с и инфракрасный (IR) порт. Некоторые контроллеры не включаются в чипсет, поскольку предназначены для приборов, не являющихся общепринятыми. Сюда входят контроллеры SCSI (80—160 Мбит/с) и IEEE1394 (FireWire, 400 Мбит/с). Если производитель МП хочет обеспечить поддержку устройства, которое не поддерживается чипсетом, надо добавлять специальный контроллерный чип.

Наиболее важными являются контроллер прерываний 8259 и контроллер ввода/вывода (Super I/O). Микросхема Super I/O включает в себя контроллер гибких дисков, контроллер последовательных и параллельного портов, контроллер клавиатуры и мыши. В хороших Super I/O реализована функция буферизации потоков в портах с помощью микросхемы UART NS 16550A (позволяет значительно увеличить пропускную способность), однако в последнее время многие функции Super I/O реализованы в южном мосту чипсета, и необходимость в этой микросхеме отпала.

Шины, слоты...

Основных шин, используемых в МП, четыре: шина CPU (данных), шина адреса, шина памяти и шины ввода/вывода. Главная характеристика любой шины — разрядность, определяющая пропускную способность. К тому же чем шире шина (по разрядности), тем большая частота может быть использована:

$\text{Полоса пропускания (bandwidth)} = \text{Частота} \times \text{Разрядность}$

Разрядность шины определяет также разрядность банков памяти. Шина данных в CPU теперь 64-разрядная. Поскольку память SDRAM и DDR также 64-разрядная, банк памяти можно организовать одним модулем памяти. Именно поэтому в

системах Pentium 32-разрядные SIMM-модули работали парно. Старые платы использовали FPM или EDO SIMM, но сейчас основным стандартом ОЗУ — SDRAM DIMM и DDR SDRAM. Модули Rambus RDRAM (RIMM) так и не прижились на рынке. Сегодня вряд ли можно насчитать два-три чипсета, которые имеют поддержку для FPM или EDO. Выбирая из DIMM и DDR, следует учитывать, что контроллеры разных чипсетов поддерживают разное количество банков памяти, а значит, и разный максимальный объем ОЗУ.

В число слотов шин ввода/вывода на МП могут входить ISA (Industry Standard Architecture), PCI (Peripheral Component Interface), AGP (Advanced Graphics Port) и AMR (Audio Modem Riser). Слоты ISA позволяют использовать 8- и 16-битные карты с рабочей частотой шины 8 МГц. Хотя существует определенный круг людей, которые используют эти "старинные" карточки, производители пытаются как можно скорее уйти от этого стандарта. PCI-слоты работают в 32-битном режиме и поддерживают частоту шины до 33 МГц (в ряде случаев и 66 МГц). AGP — это специальный 32-битный слот для видеокарт, он работает на частоте 66 МГц, обеспечивая значительную пропускную способность для графических приложений (до 1 Гбайт/с в режиме 4X, 266 МГц). Стандартом стало использование двух контроллеров USB 1.0 с пропускной способностью 12 Мбит/с, но с 2001 года постепенно вводится новый стандарт, у которого bandwidth до 480 Мбайт/с.

Современные видеокарты обладают большой вычислительной мощностью, и, соответственно, у них высокое энергопотребление и тепловыделение. В итоге в ряде качественных МП стали применять более дорогой разъем AGP Pro. В нем кроме стандартных контактов AGP имеются 48 дополнительных контактов электропитания. Разъем удлинился за счет присоединения через ограничитель в слоте 20 контактов с одной стороны и 28 с другой. AGP Pro позволяет использовать видеокарты мощностью до 110 Вт! Стандартные

AGP-видеокарты совместимы с разъемом AGP Pro, но обратной совместимости нет.

Сегодня большинство МП, если имеют слоты ISA, то один, редко два. Один слот ISA, как правило, бывает только на платах форм-фактора full-ATX из-за нехватки места (для установки еще одной карты ISA мешает PCI-слот). Это значит, что плата с двумя слотами ISA, пятью PCI и одним AGP будет устойчиво работать с картами на два ISA и четыре PCI или же на один ISA и пять PCI.

Интеграция, однако

Последние несколько лет одна из "горячих" тем — интеграция МП: нужно ли встраивать в нее видео, звук и другие возможности. Большинство "продвинутых" пользователей и геймеров решительно выступают против интеграции, так как это якобы ограничивает возможности их выбора. Они считают, что интеграция должна осуществляться на платах класса low-end, поставляемых на "массовый рынок". Однако производители находят интеграцию МП довольно привлекательной, так как это позволяет им снизить цену на товар.

Так или иначе, стало нормой интегрировать на МП видео и звук, а теперь еще и сетевые и модемные контроллеры. В качестве интегрированного звука используются контроллеры, удовлетворяющие спецификации AC'97. Она разделяет контроллеры на две части, соединенные интерфейсом AC'97 Link. Первая часть (цифровая) обеспечивает связь с CPU и контролирует соот-

ветствующие цифровые потоки, вторая (аналоговая) участвует в оцифровке сигнала и его декодировании (ЦАП и АЦП). Цифровая часть обычно встраивается в северный мост чипсета, а аналоговая выполнена на плате AMR. Но чаще аналоговая часть звука, представленная кодеком, выполняется на самой МП.

В качестве кодека применяются микросхемы разных производителей, соответственно и в качестве звука может быть значительная разница. Наиболее качественные кодеки выполнены на микросхемах CMI 8738, ALC и некоторых других, обеспечивающих шестиканальный звук. Коммуникационный интерфейс CNR (Communications Network Riser), также удовлетворяющий стандарту AC'97, организован по тому же принципу, но предназначен для реализации модема (протокол V.90), многоканального звука и сетевых функций (10/100 Ethernet, LAN). Все эти встроенные средства дают до 10% дополнительной нагрузки на процессор, что не так уж и много.

Встроенная графика до последнего времени была реализована плохо, впрочем, для офисных систем она была идеальна. Но расширение возможностей CPU привело к возможности создания мощных видеосредств, примером чему является чипсет nVidia nForce x20D, который име-

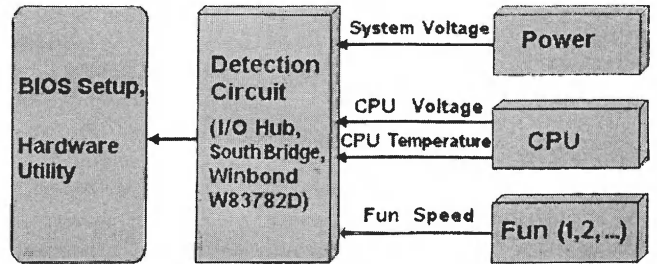
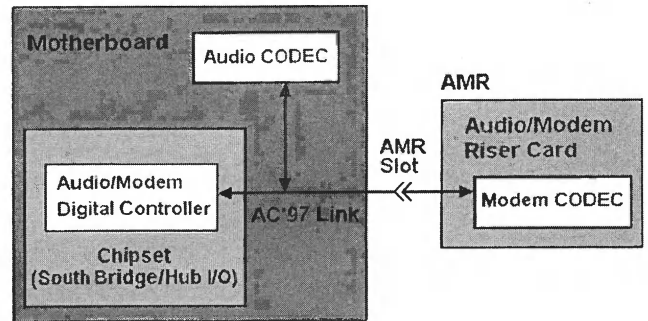


Схема мониторинга аппаратных средств

ет GeForce MX и обеспечивает изображение, близкое к реальному GF MX AGP. В этих видеочипах используется двухканальная память DDR, и нагрузка на ОЗУ минимальна. Неплохое качество обещано и в i845G. Чипсеты со встроенным видео i752 распространения не получили, так как подразумевали использование части ОЗУ в качестве видеопамати.



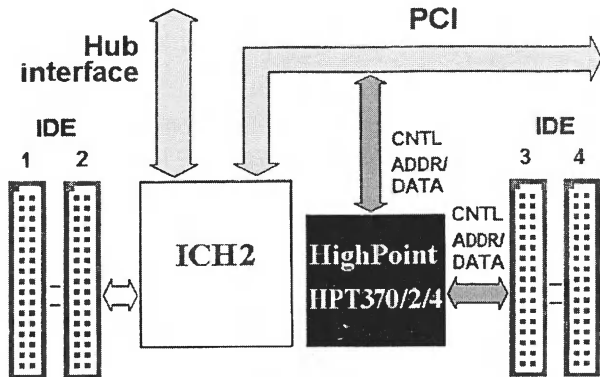
Подключение платы AMR

В последнее время, с удешевлением RAID-микросхем, многие фирмы стали встраивать RAID в свои МП (лидером в этом считается Abit). Чаще всего используются микросхемы HPT 368 370 и 374 как самые надежные, позволяющие организовать RAID 0+1. Впрочем, RAID можно реализовать и с помощью PCI-карт Promise.

Последним важным интегрированным средством стал аппаратный мониторинг (на программный полагаться нельзя!). Сначала он был реализован на отдельных микросхемах (типа Winbond W83782D), а начиная с чипсетов VIA 694x и KT133A — в структуре чипсета.

Чипсет заслуживает отдельного разговора, и речь о нем пойдет в следующем номере.

*Остались вопросы?
mailto: mpc@tp@spb.ru*



Микросхемы, обеспечивающие создание RAID 0/1/0+1



В КОМПАНИИ УМНЫХ ЧАЙНИКОВ

Михаил Виргилиев

Вынужден начать эту статью с извинений перед читателями. С первой статьей об автоматизации дома произошло практически то же самое, что происходит сейчас с самой автоматизацией домов в нашей стране. Лучше честно признать, что первый блин получился комом: статья готовилась так быстро, что иллюстрации просто не успели в печать.

Национальные особенности X10

Для начала попытаемся понять, почему у нас системы домашней автоматизации пока не стали так же популярны как, например, в США. Как известно, американцы среднего достатка предпочитают жить в собственных домах с приусадебными

участками. В большинстве районов США климат очень теплый или жаркий, и жить в доме без кондиционера так же некомфортно, как у нас — без отопления. Далее, в силу социального уклада многие работающие американцы вынуждены совершать довольно длительные поездки. И, наконец, в США довольно много районов, находящихся в зоне регулярного влияния стихийных природных факторов. Важная особенность — наличие в домах нескольких телефонных линий и нескольких компьютеров, значительно большая насыщенность бытовой техникой, чем в нашей стране.

В силу перечисленных особенностей жизни частный американский дом можно охарактеризовать следующими образом:

1. Регулирование температуры требуется практически круглый год.

2. Дом часто остается без хозяина в относительно малонаселенной местности, вдали от полицейских участков, следовательно, роль охранных и противопожарных систем в безопасности жилища далеко не последняя.

3. В любой сезон требуются искусственный полив растений на приусадебном участке, а также слежение за метеорологической обстановкой.

4. Вследствие большой величины среднего американского дома по сравнению с российским, гораздо большее значение имеют контроль за потреблением электроэнергии и управление освещением в доме.

5. Поскольку в большинстве частных домов нет централизованного горячего водоснабжения и отопления, домашние системы отопления и водоснабжения требуют пристального контроля.

Совершенно очевидно, что в таких условиях средства домашней автоматизации являются не роскошью, а необходимостью.

Несмотря на устоявшееся в нашей стране мнение о том, что Америка — очень богатая страна, на самом деле это не совсем так. Реальная статистика говорит о том, что средний доход на душу населения в США не превышает 30 тысяч долларов в год (сравните с явно завышенными цифрами 70—120 тысяч долларов в год, которые почему-то с удовольствием приводятся в нашей прессе). Именно этим, в первую очередь, объясняется любовь американцев к недорогим японским автомобилям. Именно поэтому большинство граждан США не могут позволить себе тратить слишком много на средства повышения комфорта и именно в этом причина популярности систем домашней автоматизации, использующих устаревший, но “дешевый” стандарт X10.

Технические особенности X10

Возвращаясь к технической стороне домашней автоматизации, предлагаю внимательно посмотреть на картинку, где в схематичной фор-

ме показана система домашней автоматизации, реализованная с помощью устройств стандарта X10. Прежде всего рисунок отражает тот факт, что системной следует считать набор разнородных компонентов, объединенных неким общим способом или, скорее, принципом управления. В реальной жизни компонентов может быть даже больше, а используемых принципов управления — не один, а несколько.

Как хорошо видно на рисунке, большая часть устройств соединена ничем иным, как обычной электропроводкой, по которой, помимо электроэнергии, передается и управляющий сигнал. Передача происходит с помощью радиопульсов с частотой заполнения около 130 кГц, привязанных к моментам перехода через ноль сетевого напряжения. Такой способ передачи несколько снижает влияние помех, которые всегда имеются в сети переменного тока 220 В 50 Гц. Как показала практика, даже в момент перехода сетевого напряжения через ноль в сети все равно присутствует напряжение или ток помех (чаще и то, и другое), в зависимости от качества электроэнергии, количества подключенных потребителей и характера создаваемой ими нагрузки.

К сожалению, большинство домашних электроприборов создает некий уровень дополнительных помех, и зачастую этот уровень выше того, который может быть "переварен" приемниками "умных" электроприборов. Разумеется, на уровне протокола X10 используются некоторые методы защиты от ошибок в передаваемых командах, однако далеко не всегда эта защита оказывается достаточно эффективной. Собственно говоря, это и есть одна из главных причин того, что "умные" приборы стандарта X10 так часто вызывают нарекания пользователей.

С другой стороны, далеко не в каждом доме электропроводка в идеальном состоянии, и это также не способствует высокой надежности метода передачи сигнала от "умных" датчиков к "умным" исполнительным устройствам, который заложен в стандарте X10.

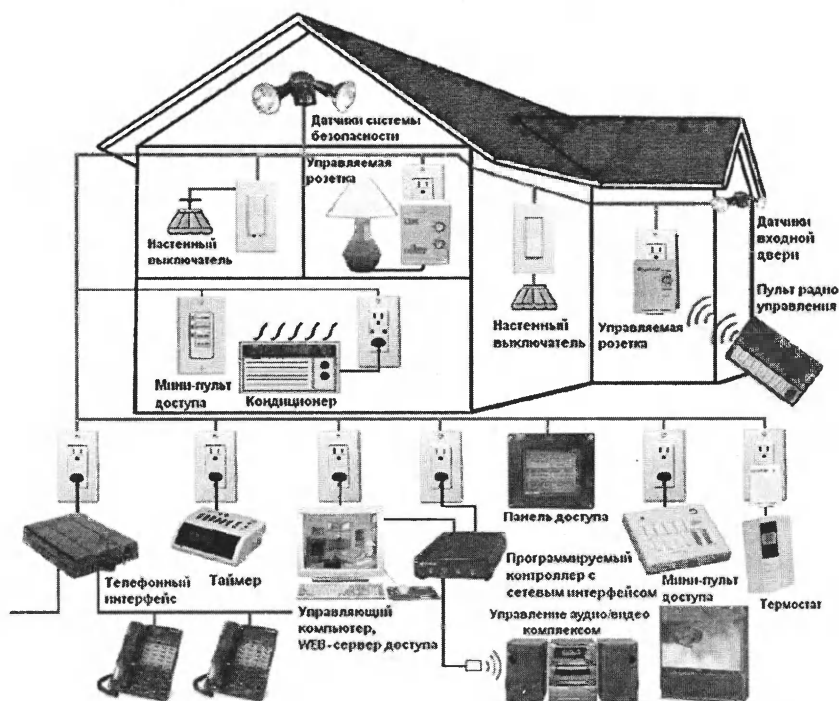
Не влезая в дебри теории передачи и обнаружения сигналов, скажу лишь главное: надежность правильного приема информации в условиях помех определяется соотношением энергии сигнала и помехи (при условии, что их характеристики удовлетворяют определенным статистическим соотношениям). Влия-

ние помех можно до некоторой степени снизить (и, соответственно, увеличить достоверность приема управляющих сигналов), если принимать управляющие сигналы не все время, а лишь в моменты перехода через ноль сетевого напряжения, то есть несколько увеличить чувствительность приемников в эти моменты времени.

Следует отметить, что далеко не все "умные" домашние электроприборы оснащаются такими усовершенствованными приемными устройствами. Стоят они существенно дороже, чем обычные "умные" приборы, и найти их в продаже значительно сложнее.

Впрочем, есть и другие способы повышения достоверности передаваемых сигналов управления, позволяющие добиться почти тех же результатов, но с меньшими затратами. Эти методы сводятся в основном к локализации и подавлению помех непосредственно в месте их возникновения. Самый простой и эффективный метод — контроль исправности электропроводки, в частности, надежности контактов в обычных электрических розетках, выключателях, патронах электрических ламп и т. п. К сожалению, визуальный осмотр и профилактическое затягивание винтов и гаек не всегда дают надежные результаты. Без специальных измерительных приборов здесь не обойтись. Называются такие приборы индикаторами сигналов X10. О недостаточном уровне сигнала или повышенном уровне помех эти индикаторы сигнализируют соответствующими светодиодами.

Вообще говоря, случаи возникновения помех критического уровня в сети "умных" приборов стандарта X10 — не редкость. Например, источниками помех, серьезно нарушающих работу "умных" приборов, могут стать различные электронные и даже пассивные балластные устройства люминесцентных и галогеновых ламп. В таких случаях лучше всего заменить "шумящую" лампу на прибор другого производителя или другого типа, либо установить специальный помехоподавляющий фильтр.



“Умные” розетки и выключатели

Рассказ об “умных” приборах стандарта X10 я начну с самого простого и популярного такого устройства. Это “умная” розетка, рассчитанная на управление всего лишь двумя командами — “ВКЛЮЧИТЬ” и “ВЫКЛЮЧИТЬ”.

Замечу попутно, что устройства бывают с однонаправленной и двунаправленной передачей команд. Однонаправленные устройства являются самыми простыми, они умеют только “слушать” и выполнять команды. Двунаправленные устройства способны откликаться на команды подтверждением их выполнения, что иногда необходимо по соображениям надежности или для того, чтобы задать правильную последовательность выполнения команд в подсистеме из нескольких “умных” устройств.

Итак, снимем крышечку с задней части “умной” розетки. Внутри, как правило, находятся электронные платы (одна или две) с довольно большим количеством электронных компонентов. По сути это обыкновенный аналоговый радиоприемник, выход которого соединен с микропроцессором, а тот в свою очередь — с мощным симметричным тиристором, электронным аналогом реле. Все это оснащено соответствующей вспомогательной электроникой — фильтрами и источником питания. Имеются и обычные кнопки, чтобы “умной” розеткой или выключателем можно было управлять с того места,

где они расположены. В схеме могут присутствовать также плавкий предохранитель и светодиоды для индикации состояния Включено/Выключено. Такой простенький “умный” приборчик в США можно приобрести за смешные \$8—10. В Европе его цена возрастет долларов (евро) этак до 20—25, а в России, скорее всего, еще раз удвоится. Тем не менее, это не останавливает некоторых от приобретения данной игрушки, уже не очень дешевой.

Надо сказать, что дороговизна — понятие всегда относительное. Установив одно или несколько таких устройств, в некоторых случаях можно сэкономить довольно приличную сумму, например, за счет удаленного выключения освещения. Стоимость сэкономленной электроэнергии через некоторое время может существенно превысить затраты на “умный” выключатель.

Некоторые приборчики специально разработаны для управления яркостью осветительных ламп или мощностью нагревательных приборов. Это уже несколько более “умные” выключатели, они способны запоминать заданную мощность лампы или нагревательного прибора. Всего можно установить до 256 уровней яркости или степеней нагрева, что позволяет осуществлять практически плавную регулировку освещенности или мощности. Эффект от использования таких устройств еще больше, чем от выключателей на два положения.

По внешнему виду подобные устройства зачастую трудно отличить от обычных евророзеток и евровыключателей — разве что по чуть большему размеру панелей и дополнительному индикаторам.

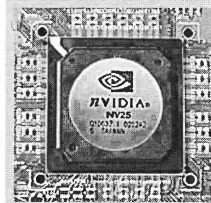
В следующий раз я расскажу об “умных” приборах других типов. Вы ждали появления на сцене главного героя, персонального компьютера? Запаситесь терпением. А на чем, по-вашему, я писал эту статью? Не на “умном” электрочайнике, честное слово.

До встречи.



О долгожданном

Чип nVidia NV30 будет представлен 18 ноября на осенней выставке Comdex. Он должен выйти минимум с двумя частотами. NV30 представляет собой графический процессор с восемью конвейерами рендеринга, по два текстурных блока на каждом. Он будет обладать расширенной поддержкой функций DirectX 9.0, более производительной памятью DDR-II. Шина памяти будет иметь ширину всего 128 бит, а частота составит не 400



и 800 МГц, как считалось ранее, а 500 МГц и 1 ГГц (что примерно на 400 МГц быстрее, чем у видеоадаптера ATI Radeon 9700). Таким образом, NV30 обещает стать самым мощным графическим акселератором.

1 октября в Европе и 22 октября в России AMD анонсировала первые процессоры на 0,13-микронном ядре Thoroughbred, поддерживающие частоту системной шины 333 МГц. Ими стали две модели: Athlon XP 2700+ и Athlon XP 2800+, имеющие реальные тактовые частоты ядра 2,17 и 2,25 ГГц. Поставки первого процессора AMD планирует начать в ноябре (к нам 2700+ попадет в декабре, а 2800+ — в будущем году).

После того как Maxtor и Western Digital выпустили своих HDD-монстров на 250 и 200 Гбайт, IBM анонсировала линейку накопителей Deskstar 180GXP (7200 об/мин, кэш 2 или 8 Мбайт). Эти приводы оснащены двигателями на гидродинамических подшипниках, поддерживают UltraATA 6 и S.M.A.R.T. Емкость — 60, 80, 120 или 180 Гбайт (60 Гбайт на пластину). Изюминка в них — фирменная технология Tagged Command Queuing (TCQ), повышающая производительность HDD примерно на 25%. Суть в том, что команды, поступающие на HDD, сводятся в кратковременный архив, а затем выполняются наиболее эффективным способом.

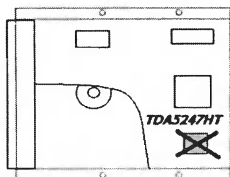
Александр Дудкин

Сразу вспомнились частые случаи выхода из строя винчестеров печально известных серий CX и LCT этой фирмы два-три года назад. Можно представить, до каких температур на них нагревались микросхемы. Предварительный диагноз ясен: локальный перегрев электронной платы и ее отдельных микросхем плюс малая распространенность (и высокая цена) специальных кулеров для охлаждения именно электронных плат винчестеров. Если бы в те времена обратили особое внимание на эту проблему, то многие из погибших винчестеров "сверхгорячих" серий QUANTUM можно было бы сохранить или, по крайней мере, значительно продлить им срок жизни. Ситуацию мог бы спасти специальный винчестерный кулер, например, типа SHDC-A.

Как показало вскрытие

Я лично был свидетелем того, как погиб один из винчестеров серии QUANTUM CX на 6 Гбайт после ремонта в квартире знакомого с циклевкой пола машинкой без пылесборника. Приличный слой очень тонкой древесной пыли буквально на всех компонентах ПК уменьшил не только производительность всех кулеров охлаждения, но и теплоотдачу корпуса винчестера. Через пару недель винчестер скончался. Напоследок судорожно погремел приводами головок, загружая ОС только с пятой-шестой попытки, и тихо умер навсегда. Диагноз патологоанатома из ремонтной службы подтвердил мое предположение о выходе из строя не самого диска, а именно одной из микросхем (контроллер привода головок). Это послужило еще одним напоминанием о необходимости проведения регламентных работ (чистка, смазка) хотя бы раз в полгода или чаще.

С QUANTUM LCT история закончилась тем же. Ремонт с заменой выгоревшего контроллера привода головок и полезным напутствием специалиста из сервисной службы,



Положение "горячей" микросхемы

Геннадий Васильев

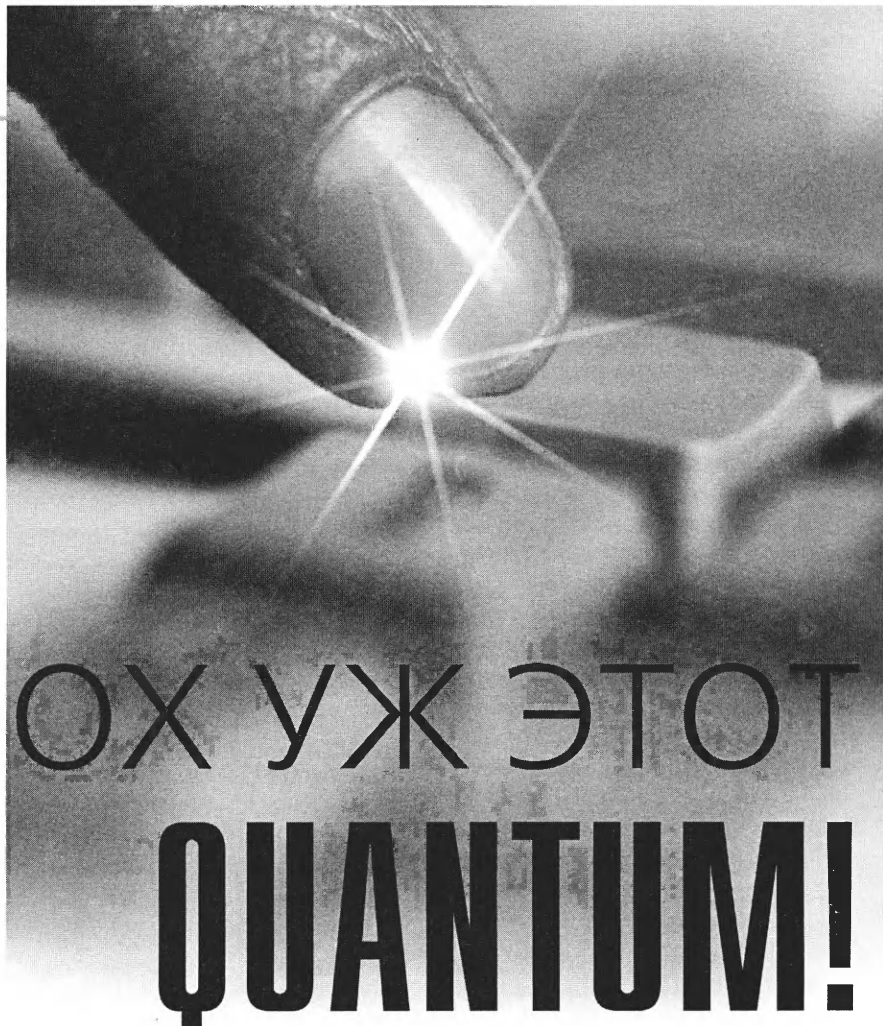
Не поверите — обжег палец! Подвела давняя радиолюбительская привычка проверять на ощупь нагрев радиодеталей. Никак не ожидал, что одна из микросхем на электронной плате QUANTUM CR на 6,2 Гбайт окажется нагретой до обжигающей пальцы температуры. На первый взгляд (на покрасневший палец) показалось, что температура явно больше 60°С. Увы, измерения это подтвердили.

который без обиняков предложил увеличить количество оперативной памяти на ПК, где будет использоваться винчестер.

Обжегши пальчик, я решил измерить температуру всех микросхем на плате QUANTUM CR в трех вариантах нагрузки именно на привод головок: холостой ход (min), дефрагментация всего диска (midi) и перенос

файла (211 Мбайт) с первого логического диска С на последний Q. Результат (таблица 1) подтвердил опасения насчет перегрева контроллера привода головок (микросхема TDA5147BH).

Измерения окончательно развеяли все сомнения относительно причин выхода из строя винчестеров. Долгая работа при температуре под +80—90°С не только выводила из

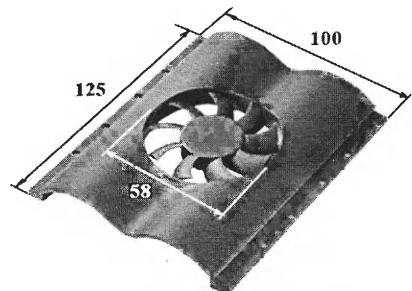


Название микросхемы	Температура в градусах		
	Холостой ход	Дефрагментация	Перенос файла
Корпус	32	33	33.5
TDA5147BH	62.5	65	73.5
F711754BPBL	46	46.5	47
Lc384161-Ct	39	39.5	40
MS241C	41	42	42

стройка микросхему, "поджаривался" сам диск.

Профилактика и гигиена

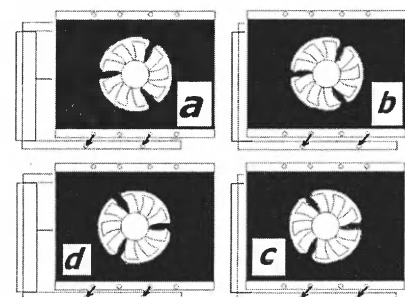
Стало ясно, что особое внимание надо обратить на эту самую горячую микросхему, то есть основной поток охлаждающего воздуха направить именно на нее. Для этого достаточно выбрать правильную ориен-



Винчестерный кулер SHDC-A

тацию и место крепления винчестерного кулера SHDC-A (это полный аналог кулера TTC-HD11 известной фирмы Titan, только ценой почти в два раза ниже).

А вариантов крепления кулера относительно винчестера может быть от 4 до 8 — таковы особенности расположения крепежных отверстий на обоих устройствах. На корпусе винчестера их по два с каждой стороны, а на корпусе кулера — по четыре. К тому же трехлучевой вариант расположения перегородок крепления электродвигателя на корпусе кулера создает в отдельных местах своеобразную аэродинамическую тень. Возможное смещение корпусов в обе стороны и их поворот на 180° позволяет найти такое расположение крыльчатки кулера, при ко-



Варианты расположения кулера на винчестере

тором эффект охлаждения горячей микросхемы максимальный.

А эффект-то немалый (см. таблицу 2).

Уменьшение температуры микросхемы даже на несколько градусов — это уже неплохо, особенно летом, в сильную жару. На деле же температура микросхемы снизилась почти на 30°C. Ну, а разница в 5—6° в зависимости от взаимного расположения корпусов говорит сама за себя.

Результат охлаждения по выбранному варианту (d) просто порадовал: ни одного сбоя в работе винчестера и ни одного бэд-блока за все жаркое дачное лето 2002 года. Даже при наружной температуре +33—35°C температура корпуса винчестера и горячей микросхемы не поднимались выше +35/+51°C (максимальная рабочая температура QUANTUM CR +50°C, а ее микросхем +70°C).

Смешная цена на винчестерный кулер типа SHDC-A (порядка \$4,5 с подшипником скольжения sleeve bearing, вариант с ball bearing на \$1—1,5 дороже) подтолкнула к идее оборудовать ими все мои винчестеры емкостью более 2 Гбайт. На основном ПК с винчестером MAXTOR (40 Гбайт, ATA133, 7200, 2 Мбайт) пришлось повторить ту же операцию по поиску оптимального расположения кулера, и температура самой горячей микросхемы снизилась на пару-тройку градусов. Вроде и пустяк, но я доволен. Тем более, что после слияния фирм MAXTOR и QUANTUM и исчезновения с рынка новых винчестеров под именем QUANTUM никто не застрахован от получения под именем MAXTOR славного потомка горячих "Квантов огненного шара" — QUANTUM Fireball.

Как жить дальше?

Боюсь "накаркать", но мне кажется, что именно в борьбе за получение таких вот пустячков в виде сни-

Вариант расположения	Температура в градусах					
	Холостой ход		Дефрагментация		Перенос файла	
	TDA	Корпус	TDA	Корпус	TDA	Корпус
Без SHDC-A	62.5	32	65	33	73.5	33.5
a	45.5	29	47	29.5	49	30
b	45	29.5	46	29.5	49.5	30
c	43	30	45.5	30	48	31
d	40.5	29.5	42	29.5	43	29.5

жения на 2—3 градуса температуры процессора, винчестера, ОЗУ и других компонентов ПК скоро и будет состоять вся наша "компьютерная" жизнь. Очень хотелось бы ошибиться, но глобальное потепление (температура мирового океана повышается на 0,1 градуса каждый год) и закон Мура (увы, и в области тепловой мощности ПК он, судя по всему, тоже скоро станет справедливым) не дают нам никаких шансов расслабиться.

Хотим мы того или нет, винчестерный кулер становится такой же обязательной частью современного ПК, как кулер для процессора или блока питания. А для современных высокооборотных винчестеров и одного кулера мало. Охлаждать нужно не только плату с электроникой, но и сам корпус. Я лично для этих целей пользуюсь шестикулерным управляемым вариантом (коммутация мощности обдува с помощью тумблеров). Остается благодарить судьбу за то, что подобного рода кулеров в компьютерных магазинах становится все больше, а сами они все дешевле.

Есть смысл установить и в BIOS, и в программах мониторинга работы кулеров контроль за оборотами самых важных, охлаждающих процессор, блок питания и винчестер. Остановка любого из них чревата потерей либо самого устройства, либо важной информации. Использование системных плат без датчиков с возможностью контроля за работой всех этих трех кулеров, к тому же без функций alarm и shutdown, в наше температурно-напряженное время — верх непредусмотрительности.

*Остались вопросы?
mailto: mpc@tp.spb.ru*

Я расскажу о том, как добиться скроллинга даже тогда, когда на мышь и колесика нет, а также о нескольких малоизвестных способах работы со скроллером.

Колесо обозрения

Даже такое нехитрое устройство, как мышь, не осталось без усовершенствований под влиянием стремительного технического прогресса. Совсем недавно мышь была дополнена новым устройством, сделавшим работу с ней еще удобнее и приятнее. Речь идет о скроллере, или колесе прокрутки, предназначенном для удобного перемещения по объектам, которые не помещаются на экране целиком (например, длинным текстам или веб-страницам).

Колесико это может быть использовано во всех версиях Microsoft Windows, начиная с Windows 98. Windows 95 использовать его для прокрутки не позволяет, однако с помощью специальных программ и драйверов использовать скроллер все-таки можно.

Само колесико — это изобретение Microsoft: недаром именно с мышами этой фирмы оно изначально ассоциировалось среди пользователей. Поэтому применять скроллер в Linux и иных отличных от Windows операционных системах затруднительно: настройкой конфигурационных файлов можно добиться разве что работы колесика по назначению, когда курсор мыши находится непосредственно над полосой прокрутки.

Колесико есть — прокрутки нет

Хорошо, когда колесико есть и оно работает. А что делать, если не работает? Причина — либо техническая неисправность, либо проблема с драйверами, либо отсутствие поддержки данной технологии операционной системой. Помимо “родных” драйверов, которые поставляются вместе с устройством, поддержку “колесной” мыши на вашем компьютере можно ожидать и



Как Microsoft ИЗОБРЕЛА КОЛЕСО

Антон Орлов

Тем, у кого и мышь двухкнопочная, и компьютер слабый, не “тянущий” ничего выше Windows 95, скроллинга тоже хочется!

от драйверов Logitech, поставляемых с мышами этой фирмы и имеющихся на ее сайте (<http://logitech.com/cf/support/mousefiles.cf>, размер — несколько мегабайт). Эти драйверы, кстати, имеют немало дополнительных функций. Например, с их помощью можно заставить операционную систему воспринимать щелчок средней кнопкой мыши как нажатие одной из клавиш клавиатуры, скажем, F5 или F1 (не путайте их со встроенными в Windows и доступными через окно установки оборудования! “Виндовские” драйверы дополнительных функций не имеют).



Обычно мышь с колесиком вполне нормально распознается операционной системой (начиная с Windows 98) и работает сразу после установки. Дополнительные драйверы приходится устанавливать лишь в том случае, когда они могут дать пользователю больше возможностей, чем стандартные (например, драйверы для мышей Logitech). В последних версиях Windows очень многие драйверы уже входят в комплект операционной системы, и при необходимости можно заменить драйвер, не прибегая к дополнительным источникам ПО.

Но значительно более интересен другой вопрос — можно ли использовать скроллинг при работе с теми мышами, на которых такого колесика вообще нет?

Колесика нет — прокрутка есть

Если на вашей мыши колесико отсутствует, а хоть как-то попробовать возможности скроллинга хочется, то можно использовать следующие приемы.

- В Windows 98 и более старших версиях нажатие средней кнопки мыши почти во всех драйверах дает возможность прокручивать отображаемый на экране текст путем простого смещения мыши и удержания ее в этом положении. Такой режим обозначается изменением формы курсора: он состоит уже из двух частей — фиксированной (овал со стрелками внутри) и перемещаемой (простая черная стрелка). При смещении мыши без нажатия кнопок начнется прокрутка текста, тем более быстрая, чем дальше перемещаемая часть курсора будет находиться от фиксированной.



- В Microsoft Word (начиная с 97-й версии) есть команда Autoscroll. При ее вызове у пользователя появляется возможность прокручивать текст. В Word 97 курсор окажется прикрепленным к вертикальной полосе прокрутки, и при движениях мыши от середины этой полосы вверх или вниз документ будет автоматически прокручиваться в соответствующем направлении. В отличие от предыдущего, этот прием будет работать даже в том случае, когда Word установлен под Windows 95. В Word XP при вызове Autoscroll происходит то же, что и при нажатии на среднюю кнопку мыши — появляется составной курсор, и документ прокручивается при смещении мыши вверх или вниз. Если вы решите использовать данный прием, назначьте команде Autoscroll сочетание клавиш или кнопку на панели инструментов.

- В драйверах для мышей

Logitech есть вкладка Buttons. Она позволяет задать сочетание клавиш или кнопку мыши для перехода в режим скроллинга. Таких режимов несколько. В режиме Auto Scroll на экране отображается такой же составной курсор, как и при нажатии средней кнопки мыши (при использовании этого драйвера иметь среднюю кнопку вовсе не обязательно!). В режиме Universal Scroll на экране появляется кисть руки, с помощью которой можно "тащить" документ вверх или вниз с довольно высокой скоростью. На той же вкладке можно указать системе, чтобы при нажатии соответствующей комбинации клавиш курсор сливался с горизонтальной или вертикальной линиями прокрутки, и скроллинг происходил вдоль выбранной линии при простом смещении мыши (как при работе команды Auto Scroll в MS Word). Так что драйверы для мышей Logitech пригодятся и владельцам простых двухкнопочных мышей.

- Программа MouseImp (<http://avsmir.narod.ru/Programs/mouse.zip>) позволяет делать три вещи. Во-первых, — осуществлять прокрутку простыми движениями мыши (неважно, "сколькочнопочной"). На вкладке Direct Scroll окна настроек программы можно выбрать одну из кнопок мыши для включения режима скроллинга, и тогда при ее нажатии и смещении в нажатом виде курсор превратится в изображение кисти руки, и смещение его в любую сторону вызовет прокрутку текста (похоже на режим Universal Scroll в драйверах Logitech).

Можно указать способ совмещения режима скроллинга и стандартных функций той или иной кнопки мыши. Один способ предусматривает переход в режим скроллинга в

снизу себе по душе. Среди мышей, и издания техники! фактически оказалась нутыми пользователями!

мышью для включения режима скроллинга, и тогда при ее нажатии и смещении в на-

момент движения мышью при нажатой включающей скроллинг кнопке — тогда для появления, скажем, выпадающего меню правой кнопки эту кнопку следует просто нажать и отпустить. Второй способ дает возможность переключаться между режимами скроллинга и стандартной функции, просто щелкая мышью.

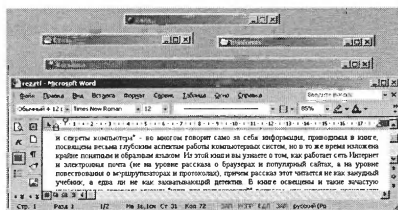
Еще одно достоинство MouseImp — функция AutoShrink. При включении данного режима программа в зависимости от настроек либо автоматически активирует окно, над которым находится курсор, либо сворачивает все неактивные окна в строки заголовков. Правда, при работе функции AutoShrink с окнами Office XP возникают различные "глюки", так что с этим пакетом AutoShrink лучше не использовать. MouseImp работает и в Windows 95. Являясь по сути shareware, она не докучает пользователя всплывающими окнами или отказами работать.

- Наверняка многим читателям известна программа ToggleMouse. У нее много полезных и интересных функций. Среди прочего есть и возможность скроллинга с помощью любой мыши. Комбинация клавиш для перехода в режим скроллинга назначается на специальной вкладке настроек программы, а сам скроллинг производится так же, как и в последних версиях Windows при нажатии средней кнопки мыши, — перемещением курсора вверх или вниз от того положения, в котором он был при включении режима скроллинга.

ToggleMouse доступна с адреса <http://www.toggle.com/>, а русификатор к ней можно загрузить со страницы <http://alexagf.narod.ru/toggle.htm>. Отличие shareware-версии от зарегистрированной сводится к тому, что через каждый час программа автоматически выгружается из памяти и ее приходится запускать заново. В Windows 95 программа тоже прекрасно работает.

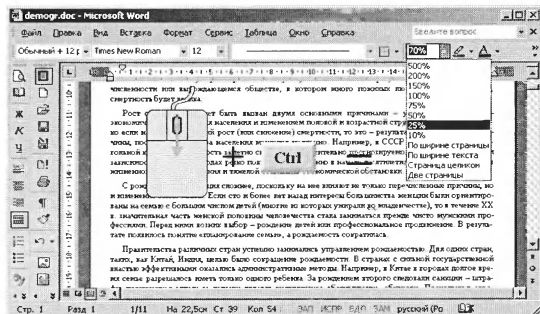
Немного хитростей

Казалось бы: ну колесико и колесико, крути его и крути — для чего



MouseImp, функция AutoShrink





Попробуйте, вращая колесико, удерживать клавишу Ctrl

его еще можно использовать? А вот попробуйте, работая в Word или Microsoft Internet Explorer, удерживать клавишу Ctrl! Будет меняться масштаб отображения документа — точно-точно как если бы вы его меняли с помощью соответствующего пункта меню. Прокрутка вверх ("к мышиному проводу") увеличивает масштаб отображения текста, вниз — уменьшает. Согласитесь, что действие колесиком значительно удобнее, чем многократное щелканье кнопками.

Данный эффект доступен во всех приложениях Microsoft Office, в браузерах Internet Explorer и Opera, а также в некоторых других программах фирмы БГ. Однако в Internet Explorer изменяется не собственно масштаб отображения, а размер шрифта текста, — картин-

ки остаются нетронутыми (в отличие, например, от Opera). Если же размер шрифта веб-страницы задан жестко с помощью тегов style, масштабирования не произойдет. Но если в пункте меню "Сервис" — "Свойства обозревателя" на вкладке "Общие" в открывающемся при щелчке на кнопку "Оформление" диалоговом окне отметить пункт "Не учитывать размеры шрифтов, указанные на веб-страницах", то абсолютно весь текст страниц, в том числе и тот, параметры которого заданы

в таблицах стилей, будет отображаться шрифтом одного размера, а уже его можно увеличить или уменьшить прокруткой колесика мыши с нажатой клавишей Ctrl. Придется, конечно, мириться с тем, что одинаковый размер шрифта на всех веб-страницах испортит дизайн просматриваемых сайтов.

Если при работе в Internet Explorer или Opera крутить колесико при нажатой клавише Shift, то прокрутка вниз станет соответствовать нажатию в браузере кнопки "Назад", а вверх — нажатию кнопки "Вперед". Иногда это удобно. Opera, начиная с версии 5.10, воспринимает вращение колесика при нажатой правой кнопке мыши как команду на переключение между открытыми в браузере окнами (аналог комбинации клавиш Ctrl + Tab).

Как видите, даже такая простая вещь, как колесо прокрутки, может стать довольно интересной темой.

Клавиатурные страсти

Берем клавиатуру, складываем ее вчетверо...

Спрашивается, а зачем ее складывать? И все-таки это возможно. На ежегодной выставке MACWORLD expo в Нью-Йорке фирма CryWolf представила на суд общественности свою новую разработку. Это плоская (!) водонепроницаемая (!) гибкая (!) клавиатура CoolMac.



Стоит этот шедевр инженерной мысли порядка \$60.

Хитрая клавиатура для "карманников"

Так называемые карманные компьютеры довольно активно входят в повседневную жизнь. Все в них хорошо, но одна мелочь никак не способствует удобству пользования таким компьютером — больно уж трудно вводить текстовую информацию

большого объема вручную. Фирма Biz Technology представила раскладную клавиатуру для компьютеров Palm, которая к тому же имеет функции подставки и защитного чехла. Теперь и случайное повреждение экрана компьютеру не грозит, и работать с ним стало удобнее.



Помоги руками перед монитором

Не прекращаются попытки усовершенствовать манипуляторы типа "мышь". Интересное предложение сделал школьник из Великобритании Тобиас Петерсон. На кисти рук надеваются перчатки с датчиками, которые отслеживают свое перемещение в пространстве и выдают сигналы в компьютер. Отслеживаются и движения пальцев, соответствующие "клику" мышки. Предпола-

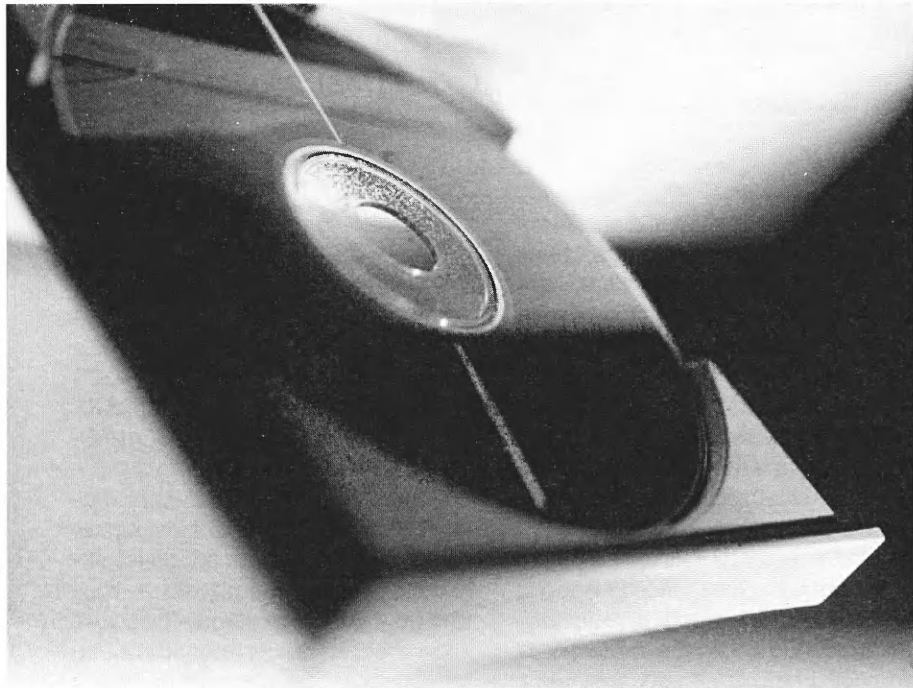


гается, что данное устройство после доработки будет востребовано производителями игр. А пока идет работа над тем, чтобы "отвязать" манипулятор от проводов. Скорее всего, их заменит инфракрасный канал связи.

Световая клавиатура

Проблема всех миниатюрных электронных устройств, а именно карманных компьютеров, мобильных телефонов и т.п. заключается в механизме ввода информации — клавиатуре. Довольно сложно разместить полноценную клавиатуру, а имеющихся 10—12 кнопок, к сожалению, недостаточно. Фирма Canesta (США) разработала виртуальную клавиатуру с тремя инфракрасными датчиками. Один проецирует изображение на поверхность стола, а два других регистрируют перемещения пальцев в воздухе и передают информацию в компьютер. Стоимость такого устройства порядка \$35—40. Профессиональная машинистка может набирать на такой клавиатуре до 80 слов в минуту.

Сергей Артюхов



ПИШЕМ компакт-диск

Георгий Баранов

Жесткий диск современного мультимедийного компьютера перегружен информацией. И с каждым днем ее становится все больше. Фотографии, музыка, видеофильмы, программы — все это надо всегда иметь под рукой и, к тому же, не потерять вместе со "слетевшим винтом". Разрешить эту дилемму очень просто: купите пишущий или перезаписывающий привод, CD-R или CD-RW.

С его помощью вы сможете создавать собственные диски, даже не прерывая текущей работы на ПК. Современные технологии позволяют полностью записать диск всего за две—три минуты, включая не только запись, но и подготовку к ней.

Немного терминологии

Типы носителей

CD-R (CD-Recordable) — диск с возможностью лишь однократной записи (включая запись в несколько приемов).

CD-RW (CD-ReWritable) — перезаписываемый диск. В противопо-

ложность CD-R, на него можно записывать данные около 1000 раз. Поскольку эти диски имеют пониженный коэффициент отражения, большинство аудиопроигрывателей и некоторые старые компьютерные приводы CD-ROM не могут их прочитать.

Стандарты

Red Book — Красная книга от Philips и Sony, основополагающий стандарт CD. Содержит только физические характеристики диска и определение нормального аудиодиска.

White Book — Белая книга, стандарт для Video CD, разработан Philips, Sony, Matsushita и JVC (версии 1.1 и 2.0).

Yellow Book — Желтая книга (ECMA-130), стандарт, в котором Philips и Sony определили расширение аудиодисков до дисков данных (CD-ROM). Подразумевает два формата записи данных.

В **Mode 1** информация записывается блоками (секторами): 2048 байт собственно данных плюс 276 байт корректирующих кодов, заголовок сектора, 12 байт синхронизирующих данных. Применяется для за-

писи CD с данными и немультимедийных дисков.

Mode 2 — формат, используемый для большинства мультимедийных дисков. Размер блока данных в формате Mode 2/Form 1 2048 байт (276 байт корректирующей информации), а в Mode 2/Form 2 — 2342 байт (не записываются корректирующие коды, применяется только для видео, где мелкие ошибки не страшны).

Дорожки и сессии

Записи на диске делятся на дорожки и сессии. Каждый диск содержит минимум 1 и максимум 99 физических дорожек (треков).

Сессия — это область данных на диске. В каждой сессии присутствует одна или более дорожек. На одном диске могут быть записаны различные сессии. Каждая сессия содержит максимум одну дорожку с данными и несколько аудио- или (и) видеодорожек.

Приводы CD-ROM и бытовые CD-проигрыватели читают только закрытую сессию. При записи на мультисессионный CD каждый раз записывается новая сессия. Если же вы записываете в несколько приемов аудио-CD, каждый раз добавляются только дорожки с данными.

Служебные зоны на диске

Каждая сессия состоит из зоны Lead-In, дорожки с данными и зоны Lead-Out. Обе зоны требуют около 15 Мбайт на каждую сессию и не содержат никаких полезных данных.

Lead-In — зона, обозначающая начало сессии, содержит ее оглавление (TOC — Table of Contents, информация о дорожках данной сессии).

Lead-Out — обозначает окончание сессии, не содержит никаких данных.

Post-gap — послепауза, короткая секция (2 с) в конце CD или после дорожки, показывающая, что вид данных меняется.

Pre-gap — короткая секция (2 с) перед началом каждой дорожки. Имеет формат новой дорожки, позволяет CD-приводу выяснить новый формат. Служит также паузой между звуковыми дорожками на аудио-CD.

Типы дисков

CD-DA. Один из самых распространенных стандартов для аудио-CD, описан в Красной книге. Эти диски можно слушать как через привод CD-ROM, так и через CD-проигрыватель. Аудиоданные записываются фреймами длительностью по 1/75 с. Каждую секунду сохраняется 44 100 разверток аудиосигнала. Развертка занимает 2 байта (16 бит) и содержит запись двух каналов (левой и правой). Объем данных в одном фрейме получается $44100 \times 2 \times 2/75 = 2352$ байт. Это и есть размер физических блоков на CD.

CD-Extra. В этом формате сочетаются аудио- и компьютерные данные. На диске CD-Extra записываются две сессии. Первая содержит аудиоданные, вторая — обычные данные. Проигрыватель музыкальных CD воспроизводит только первую сессию. Диски CD-Extra содержат дополнительные данные, например, текст и рисунки, которые можно воспроизвести на мультимедийных ПК и на специальных проигрывателях для CD-Extra.

CD-G. Представляет собой расширение CD-DA — аудиодиск с дополнительной графической информацией, которая записывается в субкоды R—W. Прослушивание музыки или голоса сопровождается просмотром фотографий или рисунков. Просмотр возможен на бытовых CD-G-проигрывателях и ПК.

CD-Text. Расширение аудио-CD. Содержит обычные аудиоданные, проигрываемые как CD-DA, и текстовую информацию, записываемую в каналы субкода, — название диска и каждой музыкальной композиции, имя исполнителя. Возможно создание нескольких вложенных текстовых меню. Дополнительную информацию могут читать проигрыватели дисков, приспособленные для CD-Text, — устройства типа CD-ROM.

CD-I. Специальный формат для мультимедиа-проигрывателей Philips. Видео-CD (bridge-disc) содержит CD-I-приложение, которое можно воспроизвести на CD-I-проигрывателе.

CD-ROM. Стандарт для записи данных на CD (Желтая книга). Блок данных занимает 2048 байт, 304 байта в каждом блоке используются для дополнительной информации и для коррекции ошибок. Полный размер блока — 2352 байта. Информация на CD-ROM может храниться в разных форматах, причем некоторые системы поддерживают свои форматы (например, Macintosh и большинство UNIX-систем).

Photo CD. Специальный формат для хранения фотографий. Разработан фирмой Kodak. Для записи настоящих Photo CD Kodak выпускает специальное оборудование и сертифицированные диски Photo CD. Спецификация Photo CD предусматривает использование CD-ROM XA и мультисессионную запись.

Режимы записи

Disc-at-Once (диск за один проход) — режим, при котором запись всего диска осуществляется за один проход лазера: включение лазера — запись — отключение лазера. В этом случае на CD-R записывается все, включая промежутки между дорожками, без прерываний со стороны программы. Данный метод используется для создания мастер-дисков.

Session-At-Once (сессия за один проход) — режим, при котором лазер за один проход записывает одну сессию (каждая может содержать несколько дорожек), между которыми лазер выключается. Используется в основном в CD-Extra.

Track-at-Once (дорожка за один проход) — метод, при котором каждая дорожка пишется отдельно. Между записями дорожек лазер отключается. В дисках типа CD-ROM нет разницы между записями Disk-at-Once и Track-at-Once. Для аудио-CD предпочтителен метод Track-at-Once как более гибкий.

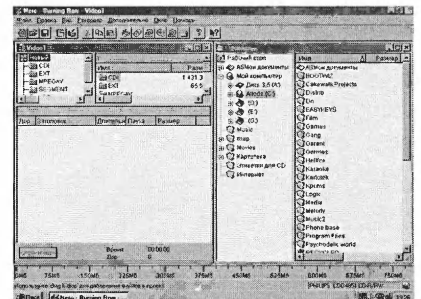
Программы для записи и перезаписи дисков**Ahead Nero Burning Rom**

Назначение: Универсальный пакет программ для записи и копирования дисков.

Возможности: Подготовка данных (музыка, видео) к записи, собственно запись, изготовление этикеток и обложек для дисков.

Форматы: CD-ROM, CD-XA, CD-DA, CD-Text, VideoCD, SuperVCD, CD MixedMode, CD-Extra, BootableCD, Shared HybridCD, HFS, ISO/HFS-Hybrid, UDF и UDF-Bridge

Один из наиболее совершенных и распространенных пакетов. К тому же один из немногих, способных записывать SuperVideoCD (этот формат придумали китайцы, чтобы не покупать DVD; может содержать до 35 минут фильма качества "почти DVD"). Для записи VCD и SVCD нужны готовые файлы форматов соответственно MPEG1 и MPEG2. К сожалению, в Nero нет встроенного декодера MPEG, и для этого придется воспользоваться другими программами. Приятная особенность программы — возможность создания загрузочного диска из файлов с жесткого диска. Программа поддерживает запись мультисессионных дисков, но по своей технологии, поэтому при продолжении мультисессионного диска, записанного другой программой, предыдущая сессия может не импортироваться.



В помощь меломанам встроена удобная утилита-декодер основных типов аудиофайлов MP3, VQF, WAV и WMA. Правда, Nero Burning Rom (Нерон сжигает Рим) конфликтует с Adaptec DirectCD, поэтому одновременно их использовать нельзя. Для создания нового проекта можно пользоваться помощником. Из списка следует выбрать один из типов будущего CD — data, audio, video, mixed. Есть также функции "Создание диска из образа" и "Копирование дисков". Если отключить помощника, при выборе нового проекта

появится более информативное окно свойств проектов CD. Через меню можно также создать копию жесткого диска.

После выбора проекта можно установить все параметры будущего диска: вид файловой системы (ISO Level 1, Level 2, joliet, кодировка и многое другое). Запись возможна с предварительным тестированием и без него. Номер и описание диска, время и дату файлов необходимо задать вручную. Скорость записи дорожки "на лету" до 12x. Программа автоматически определяет максимальную скорость записи. Есть возможность кэширования маленьких и сетевых файлов. Емкость диска задается вручную, но можно применить пережиг*. К программе есть и русский интерфейс. В целом, это удобная и качественная программа для пользователей всех уровней.

AudioCD MP3 Studio 2000

Назначение: Запись и копирование музыкальных дисков

Возможности: Записывает диски CD-DA с других CD-DA и WAV-файлов, конвертирует из MP3.

Формат: CD-DA

Количество настроек минимально, все делается в автоматическом режиме. При низком качестве диска можно установить скорость считывания 1x. При необходимости перед записью можно прослушать дорожки. Программа рассчитана на мощный компьютер и, учитывая отсутствие настроек, скорее всего не вызовет особого интереса.

WinOnCD PowerEdition

Назначение: Универсальная программа для записи и копирования дисков

Форматы: CD-ROM, CD MixedMode, CD-XA, CD-DA, CD-Text, VideoCD, Super VideoCD, CD-Extra, BootableCD, SharedHybridCD, DVD-R.

Данная программа стоит на одном уровне с Nero Burning Rom. Позволяет записывать все виды информации на все виды дисков, включая даже диски DVD-R. Легче всего с помощью программы создаются VideoCD. К тому же в программу встроена утилита расширенного редактирования VideoCD. Есть возможность создания фотоальбомов на базе VideoCD. В версию WinOnCD 3.8 включена возможность создания SuperVideoCD. Как и в Nero Burning Rom, здесь нет встроенного конвертера видеофайлов, то есть необходимо заранее подготовить файлы в другой программе. Особенность программы — возможность создания так называемых "Музыкальных альбомов". Диски этого типа создаются на основе VideoCD и предназначены для проигрывания на VideoCD-плеерах музыкальных композиций в формате MPEG1 Layer 2. На одном диске можно создать до 99 альбомов общей продолжительностью до 7 часов (каждый альбом оформляется отдельным фоновым рисунком с названием). Для создания такого диска можно использовать практически любой формат музыкальных файлов, конвертация в нужный формат происходит автоматически. В целом простая в управлении программа, поддерживающая все виды компакт-дисков. Единственный серьезный недостаток — занимает много места на диске.

WinOnCD 3.7

Назначение: Универсальная программа для записи и копирования дисков.

Форматы: CD-ROM, CD MixedMode, CD-XA, CD-DA, CD-Text, VideoCD, CD-Extra, BootableCD.

Это облегченная версия WinOnCD. Поставляется со многими приводами CD-RW. Занимает намного меньше места, чем WinOnCD PowerEdition, но и умеет при этом

намного меньше. Основные отличия видны даже по списку форматов записываемых дисков.

WinOnCD ToGo

Назначение: Программа записи дисков для неопытных пользователей.

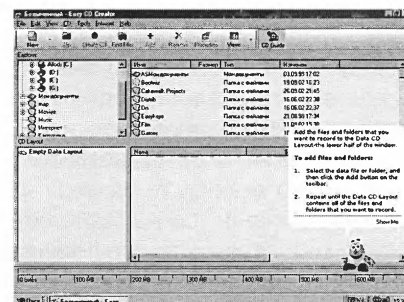
Форматы: CD-ROM, CD-XA, CD-DA.

Еще более облегченная версия WinOnCD. Ранее поставлялась со многими приводами CD-R. Занимает еще меньше места, но и умеет при этом совсем немного. Не может записывать видео и загрузочные диски.

Adaptec (Roxio) Easy CD Creator Deluxe

Назначение: Универсальный пакет для записи дисков.

Форматы: CD-ROM, CD MixedMode, CD-XA, CD-DA, CD-Text, VideoCD, CD-Extra, BootableCD, SharedHybridCD.



Один из самых "навороченных" пакетов для записи компакт-дисков под Windows. В нем можно создать диск с данными любого вида. В пакет входит большой ряд программ:

- Adaptec Easy CD Creator Deluxe — собственно программа записи
- CD Copier — программа копирования дисков
- VideoCD Creator — программа записи видео-дисков
- Jewel Case Creator — программа создания этикеток для дисков
- Sound Editor — звуковой редактор
- Session Selector — программа выбора сессии
- DirectCD — драйвер, позволяющий работать с компакт-диском как с дискетой большого размера

*) Пережиг (overburning) — запись данных сверх номинальной емкости болванки (74, 80 или 90 минут). Количество дополнительного места зависит от вида болванки и может достигать нескольких минут. Не все приводы поддерживают возможность пережигания, да и сам он поддерживается не всеми программами. Чтобы узнать точную емкость диска, можно использовать программы Feurio или CDRDiag.

(запись, стирание файлов в ЛЮБОЙ программе).

Для создания загрузочного компакт-диска используется только загрузочная дискета. Встроена проверка загрузочной области будущего диска, чего нет ни в одной аналогичной программе. Проверка очень важна, так как иногда с нормально работающей загрузочной дискеты создается неработающий загрузочный диск. При записи аудио-CD возможно использование даже внешних источников звука (данные можно редактировать). При желании можно создать альбом MP3. Нет поддержки SVCD.

Feurio

Назначение: Программа для создания аудио-CD

Формат: CD-DA.

Профессиональная программа для записи аудиодисков. Позволяет делать и копировать любые аудио-компакты, в том числе и с информацией CD-Text. Есть встроенный декодер MP3-файлов. Одновременно можно вести несколько проектов записи или записывать на несколько CD-рекордеров. С сайта производителя можно скачать демо-версию без ограничения срока действия (отключается только одновременная запись на несколько приводов и еще некоторые малозначимые функции).

NTI CD-Maker

Назначение: Универсальная программа для записи дисков.

Форматы: CD-ROM, CD-ROM XA, AudioCD, VideoCD, Mixed Mode и CD-Extra.

В программе можно создать диски всех основных форматов (правда, при создании VideoCD нельзя добавить картинки — только MPEG-файлы, поэтому создать PhotoCD не получится). Встроен конвертер аудиофайлов MP3/WAV. Есть интересная возможность записи живого звука напрямую с аудиокарты. В программе можно легко создать копии с любых дисков, как на лету, так и через образ на диске. Очень простой интерфейс. Все операции по созданию диска разбиты на два шага. Первый — выбор типа

проекта: CD Copy, Data CD, Audio CD, Video CD (конвертеров также нет), MixedMode CD, CD-Extra и Custom CD (создание собственного проекта из готовых файлов). После выбора типа проекта надо выбрать на диске подходящие файлы. Второй шаг — собственно запись. В целом простая программа, не требующая особых знаний и вполне подходящая для начинающих.

Padus DiskJigger

Назначение: Программа для копирования CD

Форматы: AudioCD, Data CD (CD-ROM, Yellow Book Mode 1), Data CD (XA format), Mixed mode CD (Data + Audio), CD-i Bridge format CD (PhotoCD, Video-CD), Multisession CD-R и pressed multisession CDs (Enhanced CD, Blue Book), CD+G.

Хотя программа предназначена для копирования и тиражирования дисков практически любых видов, в ней удобно выполнена только возможность копирования. Запись реализована настолько неудобно, что не хочется ею пользоваться. Программа состоит из одного окна. При выборе проекта предлагаются на выбор варианты: премастеринг проекта CD, создание AudioCD, копирование диска на одном приводе CD-RW, копирование на лету с CD-плеера на CD-recorder, копирование с плеера в образ, запись образа на диск CD-R, очистка диска CD-RW. Все дальнейшие операции сводятся к простому нажатию кнопок.

В целом для создания своих CD программа сложна, но для копирования подходит прекрасно.

CDRWin

Назначение: Универсальная программа для записи дисков различных форматов

Форматы: CD-ROM (mode 1), CD-ROM (XA), AudioCD, CD-Text, CD+G, CD-I, Mixed-Mode, Multisession CD, BootableCD.

Одна из многих программ, работающих не только под Windows, но и под DOS. С одной стороны, подходит для записи дисков на

ПК с маленьким объемом винчестера, а с другой, — это мощный пакет для мастеринга почти любых типов дисков. Можно записывать и смешанные типы CD. Интерфейс простой, последовательно нажимаются кнопки в зависимости от цели, но важно знать, где что находится. Новичку разобраться трудно. Есть подробная система помощи, в которой можно найти ответ практически на любой вопрос. Программа занимает всего 1,5 Мбайт на диске.

Программы для точного копирования (клонирования) дисков

BlindRead и BlindWrite

Назначение: Программы для создания точных копий с CD (копирование — BlindRead и запись — BlindWrite)

Форматы: все.

Идеально подходят для копирования защищенных дисков. При записи "ридер" считывает информацию о диске и содержимое TOC. Интерфейс аскетичен — без меню. Только несколько страниц с параметрами и элементами управления, среди которых выбор языка, извлечение информации с диска и многое другое. Доступен русский интерфейс (help без перевода). Вообще-то довольно неудобно, когда разделены чтение и запись, но, возможно, скоро эти программы будут объединены в одном пакете.

CloneCD

Назначение: Программа для создания слепков с CD

Форматы: все.

Программа предназначена только для копирования и, по утверждению производителей, создает полностью идентичные диски, копируя все и обходя защиту. Процесс записи полностью контролируется. Программа позволяет работать с дисками CD-R (CD-RW) как с дискетой — копировать через Проводник, не используя программы записи. Идеальная программа для фирм, занимающихся тиражированием дисков.



"Зеркалка" для наследников Карла Буллы

Марин Милчев

В предыдущем номере журнала рассматривались цифровые камеры, содержащие те или иные технические "изюминки". Аппараты данного обзора не претендуют на особую оригинальность конструкции, но вот их сенсоры отличаются от обычных ПЗС-матриц.

"Слоеные" матрицы

Как известно, пиксели ПЗС-матрицы не умеют различать цвета — они регистрируют только уровень освещенности. Поэтому над каждым светочувствительным элементом сенсора располагается светофильтр одного из основных (красный, синий, зеленый) или дополнительных (бирюзовый, пурпурный, желтый) цветов. Равномерная мозаика, в чем-то напоминающая шахматную доску и содержащая, как правило, 50% "зеленых" и по 25% "синих" и "красных" элементов, называется Байеровской системой светофильтров. Недостающие цвета вычисляются на основе данных о соседних пикселях, поэтому второе распространенное обозначение этого решения — схема с интерполированным цветом.

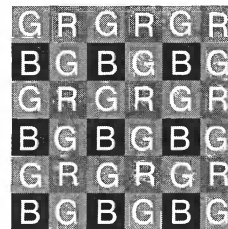
В принципе, в большинстве случаев "вычисленное" изображение аналогично оригиналу. Но, как показывает история развития техники, если какая-то схема имеет определенный конструктивный недостаток, всегда находятся "прогрессивные разработчики", изделия которых, согласно рекламным заверениям, лишены всех ошибок прежних систем. Правда, при этом умалчивается о целом "букете" новых проблем...

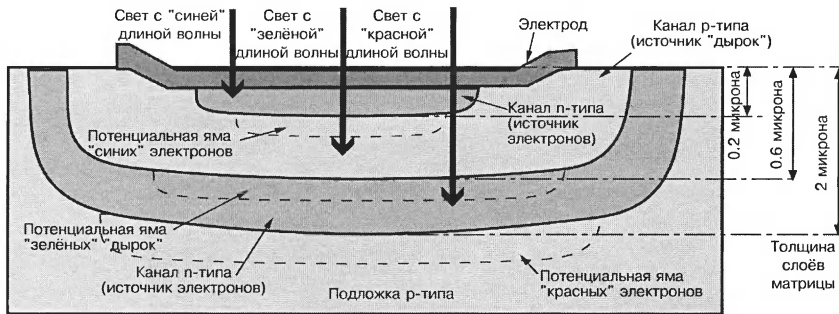
Например, Foveon с давних пор является непримиримым противником "интерполированного цвета". Сначала инженеры этой фирмы разработали студийную (стационарную) камеру с расщеплением света с помощью призмы на три составляющих (красную, синюю и зеленую), регистрируемых тремя отдельными ПЗС-матрицами. Несмотря на то, что данное решение было реализовано и в полевой (портативной) камере Minolta RD-175, сложность и громоздкость конструкции сказались как на цене, так и на массогабаритных характеристиках аппарата. Кроме того, разработчики Minolta отнюдь

не отказались от интерполяции цвета — две матрицы, расположенные со смещением на полпикселя по горизонтали и вертикали, были полностью "зелеными", а третья состояла из "синих" и "красных" элементов поровну. Просто разрешение сенсоров было невысоким, и было решено компенсировать этот изъян таким способом.

В камере Foveon требовалось обеспечить одинаковый уровень сигнала для каждого из цветовых каналов, поэтому перед каждой съемкой необходимо было провести калибровку системы, что для полевой камеры недопустимо. Кроме того, разрешение сенсоров и, соответственно, их цена, были значительно выше.

Поэтому функцию разделения света решили возложить на сам сенсор. Дело в том, что глубина проникновения света в материал матрицы (кремний) зависит от длины волны, то есть от цвета. "Синие" фотоны могут проникнуть на глубину 0,2 микрона, "красные" гораздо глубже — на 2 микрона. Как и следовало ожидать, средним по "пробивной" способности оказался зеленый свет.





В результате появился своеобразный "кремниевый наполеон".

Как видно из рисунка, слои каждого пиксела входят друг в друга, как матрешки. Чередуюсь по типу основных носителей (п-типа и р-типа), каждый следующий слой образует новую потенциальную яму — в зависимости от слоя, для электронов или для "дырок". Толщина и материал подбирались так, чтобы проникающие фотоны разделялись именно по тем диапазонам спектра, которые содержат основные цвета. Затем потенциалы ям считывались, и каждый пиксел получал "истинное неинтерполированное" значение цвета.

Разумеется, нашлись скептики, которые сходу указали на ряд "конструктивных особенностей", сильно расходящихся с рекламными заявлениями Foveon.

Например, при переполнении потенциальной ямы пиксела избыточный заряд стремится "растечься" по соседним элементам, в итоге кадр начинает пестреть белыми кляксами. С этой неприятностью, именуемой блюмингом, постоянно борются разработчики сенсоров — то создают рядом с пикселем специальные "канавки" для "сброса" избыточных электронов, то прилагают определенный потенциал к тыльной стороне матрицы. Называются эти меры электронным дренажом, соответственно, боковым и вертикальным.

В матрице Foveon избыточный заряд может двигаться не только по горизонтали, но и по вертикали, взаимодействуя с соседними слоями. Как видно из схемы сенсора, вертикальный дренаж при трех расположенных по вертикали хранилищах зарядов неприменим, поэтому остается "копать канавки". В результате уменьшается и без того небольшая

полезная (светочувствительная) площадь пиксела.

"Слоеная" структура сенсора является причиной еще ряда взаимосвязанных недостатков. В частности, теряя энергию при переходе из слоя в слой, не все фотоны доходят до "места назначения", что ведет к ослаблению светочувствительности матрицы (не более ISO 100 по паспортным данным). Пользователь вынужден шире открывать диафрагму, в результате чего больший процент лучей падает на поверхность сенсора под большим углом. В обычных матрицах эти лучи просто не регистрируются, а вот в изделии Foveon за счет преломления на стыке слоев они могут попасть в "чужой" слой.

Предположим, что разработчикам каким-то образом удалось решить все перечисленные проблемы. Означает ли это, что многослойные матрицы при этом раз и навсегда заменят сенсоры "классической" конструкции? Вряд ли.

Дело в том, что при прочих равных характеристиках Байеровские схемы будут всегда лидировать по такому параметру, как динамический диапазон (способность отличать самые темные оттенки от самых светлых). Этот параметр определяется "емкостью" потенциальной ямы, которая "глубже" у матриц с интерполяцией цвета. Ну, а профессионального фотографа мало будет интересоваться, насколько точно передан цвет "слоеным" сенсором, если одна половина кадра будет состоять из белого пятна, а вторая — из черного.

Разумеется, скептицизм требует некоторого обоснования. Созданная на базе сенсора Foveon F7-35X3-A25B профессиональная цифровая "зеркалка" Sigma SD9 была анонсирована еще в феврале этого

года, одновременно с тремя другими аппаратами — Canon D60, Nikon D-100 и Fuji FinePix S2. Обладая рядом сходных характеристик (коэффициент фокусного расстояния SD9 — 1.7, остальных камер — 1.5-1.6), аппарат Sigma сильно уступал как по разрешению (3 мегапиксела против 6), так и по чувствительности (ISO 100 против ISO 1000—1600).

Прошло полгода, а особого всплеска по части продаж Sigma SD9 не наблюдается. Конечно, можно объяснить это интуитивным стремлением людей покупать "мегагерцы и мегапиксели", но пользователи профессиональных цифровых "зеркалок" — народ серьезный, на рекламные посулы не поддается. Так что до повсеместного распространения "слоеных" матриц еще далеко. К тому же последние разработки Kodak и Canon делают будущее этой технологии еще более неопределенным.

КМОП-сенсоры

Подавляющее большинство микрорезисторных компонентов производится по технологии КМОП, что означает "комплементарные метал-окисел-полупроводник" (CMOS — complementary metal oxide semiconductor).

В КМОП-матрицах фотоны преобразуются в заряд таким же образом, как и в ПЗС-матрицах. Отличие заключается в том, что преобразование заряда в напряжение и его усиление происходит прямо внутри элемента матрицы.

Для синхронизации работы, а также передачи полученных значений используются адресные шины столбцов и строк матрицы. При этом возможно считывание всей матрицы, столбца либо строки и даже отдельного элемента. Более того, от-



падает необходимость в регистрах сдвига и управляющих микросхемах. Значительно сокращается и энергопотребление.

Технология производства КМОП-матриц широко распространена, тогда как процесс изготовления ЭОП на основе ПЗС достаточно сложен и требует специализированного оборудования. Тем, что практически любой радиоэлектронный завод может в кратчайшие сроки наладить выпуск ЭОП типа КМОП-сенсоров, определяется их невысокая стоимость.

С момента появления КМОП-матрицы декларируются "наиболее перспективными ЭОП", однако ряд недостатков не позволяет им полностью вытеснить ПЗС-сенсоры.

Во-первых, в каждом элементе матрицы присутствуют преобразователь заряд—напряжение и усилитель сигнала. Как и любые электронные компоненты, эти устройства при обработке сигнала добавляют к нему помехи — электронный шум. Причем для каждого пиксела матрицы уровень электронного шума разный. Во-вторых, из-за размещения вокруг пиксела дополнительных компонентов, "обвязки", площадь светочувствительного элемента мала даже по сравнению с ПЗС-матрицами с буферизацией строк, что снижает чувствительность ЭОП данного типа.

Для борьбы с электронным шумом применяется технология "активных пикселей". К преобразователю и усилителю каждого элемента матрицы добавляется специальный шумоподавляющий фильтр, измеряющий уровень помех и "вычитающий" их из сигнала. Общее усложнение сенсора ведет к росту себестоимости и увеличению доли брака в производстве. Кроме того, сокращается и без того малая площадь светочувствительного элемента.

В матрицах с небольшой диагональю (полдюйма и меньше) для увеличения чувствительности, как и в ПЗС-сенсорах с буферизацией строк, применяются микролинзы. Кроме того, непрерывно ведутся работы по уменьшению размеров дополнительных компонентов.

Совокупность показателей

КМОП-сенсоров такова, что они чаще встречаются в любительских камерах начального уровня с невысоким разрешением, простой оптикой и, разумеется, привлекательной для покупателя ценой. Однако все чаще встречаются исключения из этого правила.

Для своих профессиональных камер EOS D30 (2000 г.) и D60 (2002 г.) фирма Canon использовала крупногабаритные (15,1 x 22,7 мм) КМОП-матрицы с разрешением 2160x1440 и 3072x2048. Технология "активных пикселей" позволила снизить уровень электронного шума в этих сенсорах. Стоимость ЭОП из таких элементов сравнима с ценой ПЗС-сенсора, но для камеры профессионального класса это обстоятельство не является серьезным препятствием. Большие размеры матрицы позволили увеличить относительную площадь светочувствительных элементов (размеры "обвязки" даже с учетом шумоподавляющих компонентов сохраняются прежними), так что светочувствительность этого сенсора находилась на приемлемом уровне (у D30 — до ISO 1600, у D60 — до ISO 1000), а коэффициент фокусного расстояния составлял 1.6.



Однако в топ-модели цифровых "зеркалок" — EOS-1D — Canon все же использовала ПЗС-матрицу, что дало повод скептикам усомниться в пригодности КМОП-сенсоров для "серьезных" задач. Появившаяся на днях обновленная версия этого фотоаппарата (EOS-1Ds) полностью опровергла эти предположения, поскольку снабжена 11-мегапиксельным КМОП-сенсором, который по

своим размерам (35,8 x 23,8 мм) идентичен кадру 35-миллиметровой пленки, то есть коэффициент фокусного расстояния просто отсутствует. Под стать почти вдвое большему по сравнению с предыдущими моделями разрешению (4064x2704) и прочие характеристики — чувствительность (ISO 100—1250) и скорость съемки (до 10 кадров со скоростью 3 кадра в секунду). Камера "понимает" объективы серии Canon EF и вспышки моделей Canon Speedlite 380EX, 420EX и 550EX.

И даже 11 мегапикселей не стали пределом по разрешению. Kodak анонсировала свою модель "зеркалки", DCS PRO 14n, разрешение которой составляет 4536x3024, то есть 14 мегапикселей. Размер нового сенсора, выполненного тоже по КМОП-технологии, также полностью совпадает с кадром пленки 35 мм. Чувствительность составляет ISO 80-640, скорость съемки 1,7 кадр/с при серии из 8 кадров, также обеспечивается совместимость со всей оптикой под байонет Nikon F и со вспышками SB-80DX, 28DX и 50DX.

Следует отметить, что DCS PRO 14n, в отличие от предыдущих моделей серии DCS, представляет собой не просто "крышку с матрицей" для стандартного Nikon F5. Инженеры Kodak разработали совершенно новый корпус на базе Nikon F80 с элементами конструкции F100 и F5. Благодаря этому новая камера выглядит не таким уродливым кирпичом, как аппараты серии 720x/760, и по габаритам (131 x 158 x 89 мм) практически идентична Nikon D-100. И хотя по цене DCS PRO 14n почти в два раза (\$4000) дороже существующих на рынке 6-мегапиксельных "зеркалок" Nikon D-100, Canon D60 и Fuji FinePix S2, совместимость с отличной оптикой Nikon и огромное разрешение обеспечивает новинке Kodak подавляющее преимущество.

Ну, а будущее технологии КМОП-матриц выглядит сейчас совсем в другом свете. Теперь ясно, что наибольший успех их ждет не в любительских "мыльницах" с полдюймовыми по диагонали сенсорами, а в профессиональных "зеркалках" на базе 35-миллиметровых камер.

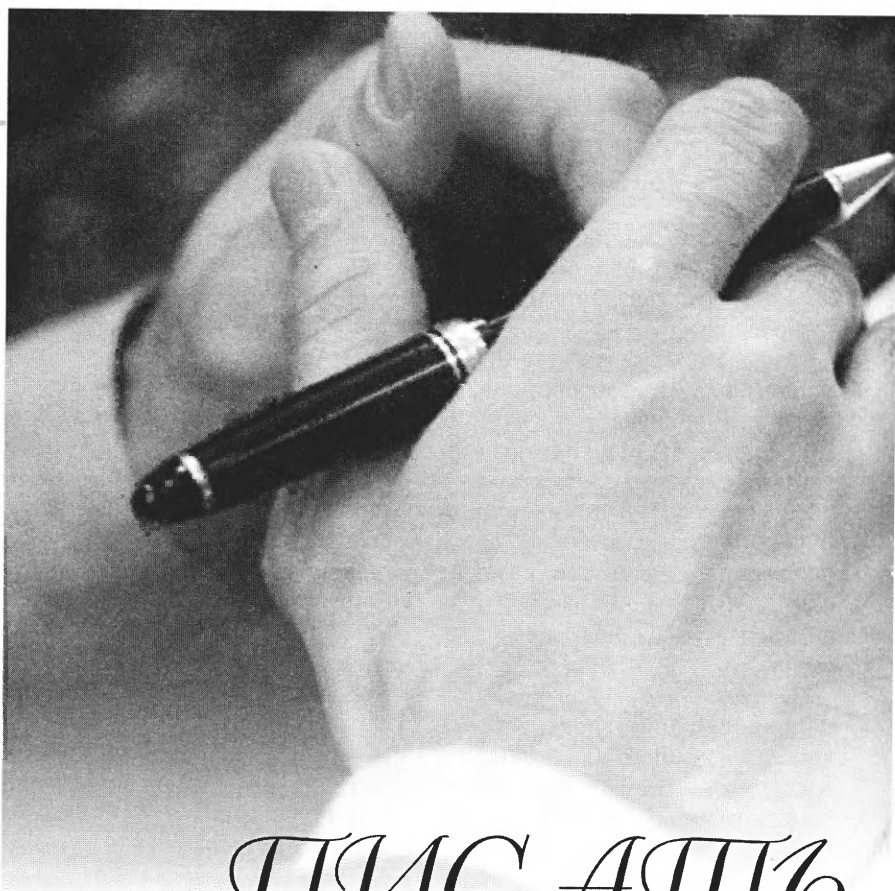
Уменьшение размеров карманных компьютеров неминуемо ведет к ликвидации клавиатуры как отдельного устройства. С помощью электронного пера знаки можно вводить либо рукописно, либо путем набора (указания) на виртуальной клавиатуре.

Интересно сравнить между собой эти два метода ввода информации с точки зрения скорости и безошибочности ввода. Что же лучше — писать знаки или набирать? Ответ на этот вопрос зависит от множества факторов — от длины алфавита (чем он длиннее, тем больше размер клавиатуры и тем больше времени уходит как на поиск знака, так и на движение руки до клавиши), от сложности знаков, размера и разрешающей способности экрана, надежности распознавания рукописных знаков, удобства рабочей позы, освещения и т. д. Ограничим задачу случаем ввода числовой информации.

В большинстве случаев с карманным компьютером работают без стола, стоя. Проведем эксперимент с измерением скорости рукописного ввода цифр и набора их на виртуальной клавиатуре по методике, описанной в предыдущем номере журнала.

Хронометраж (таблица 1) показывает, что в среднем написание одной цифры занимает 345 мс. Для удобства сравнения сложности цифр в правой графе таблицы приведены относительные значения (ко времени написания самой быстрой цифры). Самой быстрой оказалась цифра 1, для ее написания достаточно всего 170 мс. Самые медленные — цифры 4 и 5. Это связано с необходимостью возвратных движений и отрыва пера от поверхности. Разумеется, к этим измерениям нужно относиться критически. Ведь на время написания цифры влияют и размеры вычерчиваемых цифр, и, особенно, допускаемая степень упрощения формы цифр, что, в конечном итоге, может сказаться на надежности распознавания.

Если от записи цифры перейти к записи числа, к суммарному времени записи добавляется время пере-



ПИСАТЬ

ИЛИ НАБИРАТЬ?

Виталий Шнейдеров

ключения нашего сознания с одной цифры на другую. При этом нужно исходить из объема кратковременной памяти человека, которая определяется пределами 5—9 цифр. Не случайно при записи длинных чисел мы разбиваем их на короткие группы, например, шестизначное число 164179 на два трехзначных, 164 и 179.

Хронометраж показал, что среднее время написания одной цифры в числе равно 337 мс.

Хронометраж написания цифр

Цифра	Время, мс	Относительное время
1	170	1
2	310	1.8
3	350	2
4	450	2.6
5	480	2.8
6	290	1.7
7	350	2
8	380	2.2
9	370	2.2
0	240	1.4

Теперь посмотрим, сколько времени займет набор цифр на виртуальной цифровой клавиатуре (цифра выбирается пером на экране). Получилось, что в среднем на набор одной цифры требуется 280 мс. Это время складывается из времени зрительного поиска цифры на экране и движения пера к экрану (20 мс) и времени контакта пера с экраном (80 мс).

Если при рукописном вводе числа среднее время контакта пера с поверхностью составляло 80%, а 20% времени перо было поднято, при клавиатурном вводе это соотношение изменилось с точностью до наоборот: время контакта пера с поверхностью (время указания) составляет 28%, а время поиска и движения к цифре — 72% от общего среднего времени набора одной цифры. Таким образом, оказывается, что клавиатурный набор происходит почти в полтора раза быстрее, чем рукописный.

Очевидно, чтобы увеличить ско-

рость написания цифр, необходимо упростить их начертание. Попробуем прикрыть листом бумаги нижнюю половину цифр. Оказывается, что верхней части достаточно, чтобы правильно опознать большую часть цифр. Правда, по нижней части распознавать цифры значительно труднее: практически не различаются такие цифры, как 3 и 5, 6 и 8, 1 и 7. Тем не менее, современное начертание цифр отличается некоторой избыточностью, и первым шагом в их упрощении может стать выписывание только верхней или нижней части. Так, для цифры 5 это может быть верхний горизонтальный штрих, а для цифры 2 — нижний горизонтальный.

1 2 3 4 5 6 7 8 9 0

1 2 3 4 5 6 7 8 9 0

Другой резерв ускорения письма основан на том факте, что при быстром письме происходит сглаживание углов. Значит, в упрощенных начертаниях цифр нужно избегать прямых и острых углов, возвратных движений. Это касается цифр 4, 5 и 7. Но, сглаживая цифры, нужно сохранить их узнаваемость.

Дополнительные возможности по упрощению начертания цифр дает их пространственная привязка относительно двух горизонтальных линий строки. Так, цифры 3, 5 и 6 можно "привязать" к верхней линии, а цифры 2, 9 и 0 — к нижней. Такое кодирование оставляет возможность свободной, естественной для каждого человека манеры письма. Поскольку здесь нет привязки к строгим форматам подобно почтовому индексу на конвертах, допускаются большие вариации как в размерах, так и в форме знаков. Программа распознавания для такого упрощенного написания цифр оказывается предельно простой, поскольку в этом случае достаточно определять только вертикальное положение пера относительно строки в трех градациях — выше строки, в строке и ниже строки.

1 2 3 4 5 6 7 8 9 0

Подобным образом, шаг за шагом можно значительно упростить графику цифр, сохранив при этом их ассоциативность с классическим начертанием и выработанную веками манеру написания.

Остается проверить, насколько сократилось время написания упрощенных цифр. Результаты измерения приведены в таблице 2: в среднем столбце данные для стандартного начертания цифр и в правом — для упрощенного.

Хронометраж упрощенного написания цифр

Цифра	Время, мс	Время, мс
1	170	140
2	310	150
3	350	140
4	450	220
5	480	150
6	290	160
7	350	250
8	380	180
9	370	170
0	240	200
среднее значение	345	174

Из таблицы видно, что среднее время написания цифр сократилось до 174 мс, то есть стало почти в два раза меньше!

Отметим, что возможности распознавания знаков в режиме on-line оказываются выше, чем в режиме off-line. Так, две окружности могут казаться идентичными, но одна из них была нарисована при движении пера по часовой стрелке, а другая —

против часовой. Возможности, возникающие при вводе в реальном времени, могут дополнительно использоваться для кодирования рукописных знаков.

В журнале "Вопросы стенографии" за 20-е годы прошлого века приведен вариант стенографической записи цифр. Интересно сопоставить их с тем, что получилось у нас.

1 2 3 4 5 6 7 8 9 0

Цифры 1, 2 и 7 почти совпали. Предложенные тогда начертания цифр 3, 4, 5, 6, 8 и 9 отличаются предельной простотой, но утрачивают ассоциативность с рукописными знаками. Восстановить по ним исходные цифры достаточно трудно. Здесь требуется дополнительная работа, переучивание. Но в те бурные годы никакие трудности не останавливали энтузиастов. Более того, для каждого числа от 0 до 100 был создан свой стенографический знак! Как говорится, учиться, учиться и еще раз учиться.

В целом можно сказать, что переход на упрощенное написание цифр ускорит ввод и не усложнит их распознавание. На экране монитора они будут отображаться в классическом виде.

Естественно, задача компьютерной стенографии должна решаться системно. Специально для скептиков замечу, что в этом нет ничего противоестественного. Как видно из последнего рисунка, цифры (как и буквы) эволюционировали на протяжении веков, и воспринимать их как нечто неизменное нельзя. Эволюция продолжается.

- 1 2 3 4 5 6 7 8 9 0 • IX век, Индия
- 1 2 3 4 5 6 7 8 9 0 • X век, западные арабы
- 1 2 3 4 5 6 7 8 9 • X век, Испания
- 1 2 3 4 5 6 7 8 9 0 • XII век, Франция
- 1 2 3 4 5 6 7 8 9 0 • XIII век, Франция
- 1 2 3 4 5 6 7 8 9 0 • XIV-XV века, готические цифры
- 1 2 3 4 5 6 7 8 9 0 • XV-XVI века, цифры эпохи Возрождения
- 1 2 3 4 5 6 7 8 9 0 • Современные восточноарабские цифры

Действительно, наши голоса основаны не на идеальных звуках — синусоидальных колебаниях с разными частотами и неизменными во времени амплитудами. Звуки вокруг нас очень сильно отличаются от математически идеальных колебаний. Громкость каждого звука непрерывно меняется в зависимости от случайных факторов. Отдельные колебания складываются друг с другом, вычитаются и даже перемножаются. Звуковые волны подвержены рефракции и интерференции. Энергия звукового колебания в ограниченном помещении заставляет его резонировать, вызывает многочисленные переотражения и реверберацию. Словом, звук в нашем мире мечется, живет, дышит, переливается и рождает прекрасное — музыку.

Музыка существует вместе с нами в нашем живом пространстве, а все живое мимолетно, исчезает вместе с мгновениями, уносящимися в прошлое. А так хочется сохранить хотя бы след, воспоминания о них! Вот мы и пытаемся записать музыку. Но обычно из-за невозможности отыскать подходящий для исполнения музыки уголок звукового мира, из-за несовершенства и механистичности средств записи, преобразования и хранения звуковой информации музыка из живой превращается в мертвую. Во всяком случае, музыкальные звуки, как и звуки вообще, оказываются оторванными от естественного звукового пространства. Вспомните, как неправдоподобно звучат голоса красноармейцев в сценах кинокартин времен Великой Отечественной войны, снятых в павильонах эвакуированной студии "Мосфильм".

Имитация акустической среды

Можно ли запечатлеть в записи музыки живое звуковое пространство, в котором она исполнялась? Можно, но ценой огромных затрат, которые под силу только мощным звукозаписывающим фирмам, и только в специально созданных по-



ДРОЖЬ В ГОЛОСЕ- ЭТО ТАК ПРЕКРАСНО

Юрий Петелин

Уроки музыки на компьютере

Вообразите себе мир, в котором звуку не от чего отражаться. Ваши слова, щебет птиц, шелест листвы, шум дождя, рождаясь, улетают в безграничное пространство или, наоборот, без остатка поглощаются некоей сферой, абсолютно черным телом. Представьте, что этот мир населяют существа, голоса которых, основанные на идеальных гармонических колебаниях, звучат абсолютно ровно, монотонно, беспристрастно. Такой страшный мир имеет право на существование, но он не может быть миром людей, и в нем вряд ли возникло бы явление, которое мы называем музыкой. К счастью, мир, в котором мы живем, не таков, и музыка в нем есть.

мещениях — студиях, концертных залах. В них может быть хорошая акустика, но опять-таки это будет акустика зала, а не, скажем, акустика горного ущелья или лесной поляны. Словом, реально у отдельно взятого звукооператора, звукорежиссера есть только два варианта: либо записывать музыку в мертвом звуковом пространстве, либо попытаться оживить его, искусственно смоделировав свойства, присущие тому или иному варианту организации звукового мира. Конечно, второй вариант предпочтительней. Он, правда, создает у слушателя лишь иллюзию присутствия в некоей акустической атмосфере. Это обман, но слушатель либо не знает о нем, либо легко мирится с этим фактом.

До 80-х годов прошлого века средства имитации акустических свойств среды были уникальными, громоздкими, несовершенными, дорогими и малодоступными. С развитием цифровых устройств и соответствующих методов обработки сигналов появились приборы, реализующие различные звуковые эффекты. Но и они были дороги, остаются такими и по сей день.

Революция в технологии сотворения звуковых миров свершилась в тот день, когда началось массовое производство звуковых карт, устанавливаемых в компьютер в качестве платы расширения. Правда, не о музыкантах и звукорежиссерах думали разработчики первых "саунд-бластеров". У их продукции был бо-

лее массовый потребитель — армия любителей компьютерных игр. Но, к счастью для музыкантов, виртуальные миры, которые компьютер рисовал на экране монитора, становились все более сложными, а совершенствование визуальной составляющей игр вызывало необходимость адекватного улучшения звукового сопровождения. Нужно было, чтобы игрок ощущал смену поля боя не только через изменение текстуры изображения, но и за счет изменения характера фоновых звуков. В общем, постепенно звуковые карты обрели элементы, необходимые для синтеза музыки, записи звука и создания звуковых эффектов.

Звуковые карты оказались существенно дешевле, чем музыкальные синтезаторы и приборы для создания эффектов. Но это объясняется не только тем, что их начинка менее совершенна, чем у соответствующей специализированной аппаратуры. Главное — большой спрос и, соответственно, массовый характер производства. Вот так из вод первобытного океана компьютерных игр вышли на берега творчества новые существа — компьютерные музыканты.

Звуковые эффекты

Использование звуковых карт, плат оцифровки звука и музыкальных редакторов предоставляет компьютерному музыканту довольно широкие возможности по применению различных звуковых эффектов и приемов обработки звука. Звуковые эффекты могут быть реализованы аппаратным путем, и тогда их можно использовать в реальном времени, как, например, это сделано в высококачественных звуковых картах. Для этого в их состав включены цифровые сигнальные процессоры (DSP — Digital Signal Processor). Такой процессор позволяет обрабатывать звуковые сигналы в реальном времени на основе аналого-цифрового преобразования сигнала с последующей обработкой (алгоритмы цифровой фильтрации и цифровой задержки). Правда, полноценный DSP чрезвычайно дорог и применяется

только в приборах профессионального назначения. Сигнальные процессоры звуковых карт значительно упрощены в сравнении со своими старшими братьями. Обычно они не позволяют использовать одновременно большое число эффектов, к тому же почти все эффекты реализуются лишь одновременно для всех каналов.

Для выбора эффектов и управления их параметрами по MIDI служат контроллеры. В составе большинства музыкальных редакторов имеется соответствующий интерфейс. Чаще всего управление контроллерами эффектов осуществляется путем "рисования" графика изменения параметра эффекта или с помощью одного из регуляторов виртуального микшера, входящего в состав того или иного музыкального редактора.

В компьютерных студиях звуковые эффекты часто создаются программным способом. Средствами звуковых редакторов обработке подвергается заранее записанный в цифровой форме звуковой сигнал. Недостаток такого подхода в том, что некоторые звуковые эффекты невозможно использовать в реальном времени, в процессе записи, а достоинство — в том, что отказ от обработки в реальном времени позволяет применять самые сложные алгоритмы. Поэтому число достижимых звуковых эффектов и вариантов каждого эффекта в данном случае значительно выше, чем при аппаратной реализации. Кроме того, имеется возможность практически неограниченного "вложения" эффектов один в другой. Предел диктуется не техническими (точнее, не математическими) возможностями, а здравым смыслом и эстетическими критериями.

О том, как воспользоваться звуковыми эффектами, имеющимися в популярных звуковых редакторах, я и собираюсь рассказать в очередной серии статей. Звуковые карты и звуковые редакторы предоставляют в наше распоряжение массу звуковых эффектов. Среди них есть и не похожие ни на что, даже противоречащие законам физики (на-

пример, не затухающее, а нарастающее эхо). Сегодня речь пойдет о самых простых эффектах — разновидностях вибрато.

Вибрато

Суть эффекта вибрато заключается в периодическом изменении одного из параметров звукового колебания: амплитуды, частоты или фазы. Изменение (колебание) параметра происходит с очень малой частотой — единицы герц. Различают амплитудное, частотное и фазовое вибрато. В любом случае этот эффект обогащает спектр исходного колебания. Читатели, знакомые с основами радиотехники, понимают, что на самом деле происходит модуляция звукового колебания низкочастотным сигналом. Законы физики неумолимы — спектр сигнала при этом действительно расширяется.

Первоначально словом "вибрато" именовалась модуляция любого параметра звукового колебания. Но со временем некоторые из разновидностей этого эффекта получили свое название. Во многих публикациях по электронной музыке теперь под вибрато подразумевают только частотное вибрато. Это не совсем верно. Следует различать амплитудное вибрато, частотное вибрато и тембровое вибрато, а у фазового вибрато даже имеется специальное название — фейзер (от английского Phaser — фазовариатор).

Владение приемом вибрато отличает очень хорошего певца от просто хорошего. Кстати, в итальянской номенклатуре качества певческих голосов один из вариантов звука без вибрато обозначается термином "воче дура". Не пугайтесь, в переводе на русский это означает всего лишь "голос прямой".

Скрипка в руках талантливого музыканта потому и звучит так божеественно, что, совершая едва заметные перемещения вдоль грифа прижимающими струны пальцами, он осуществляет частотное вибрато. Тремоло (частный случай амплитудного вибрато) является основным приемом игры на мандолине, домре и балалайке.

Амплитудное вибрато и тремоло

Сущность амплитудного вибрато состоит в периодическом изменении амплитуды звукового сигнала с очень небольшой частотой, от долей герца до 10—12 Гц. Если частота вибрато вне этих пределов, нужный эстетический эффект не достигается.

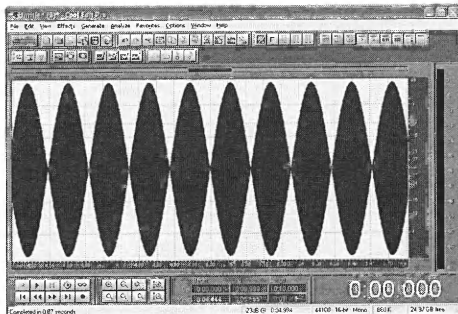
Спектр, а значит, и тембр сигнала с амплитудным вибрато богаче по отношению к тембру исходного сигнала. С таким спектром можно продвигать различные манипуляции, например, изменять уровни спектральных составляющих с помощью фильтров.

Степень проявления эффекта характеризуется глубиной вибрато: отношением максимального изменения амплитуды вибрато к амплитуде исходного сигнала. Диапазон допустимых значений глубины вибрато — от 0 до 1. Оптимальная с точки зрения художественного результата частота амплитудного вибрато составляет 6—8 Гц.

Особой разновидностью амплитудного вибрато является тремоло. Его отличительные признаки — относительно высокая частота вибрации (10—12 Гц), максимальная глубина, равная 1, и импульсная форма результирующего сигнала.

В аналоговых устройствах амплитудное вибрато реализуется с помощью перемножителей сигналов. Основной недостаток этих устройств — неполное подавление управляющего сигнала. При большой глубине вибрато это проявляется в виде ясно прослушиваемого «стука» с частотой модуляции. В компьютерных программах этот недостаток может быть полностью устранен.

Компьютерные музыканты имеют дело с двумя вариантами реализации амплитудного вибрато: аппаратным и программным. Аппаратный способ предполагает наличие в звуковой карте усилителей с управляемым коэффициентом усиления. Программный способ заключается в перемножении значений цифровых отсчетов звуковых колебаний со значениями отсчетов функции, описывающей управляющий сигнал.



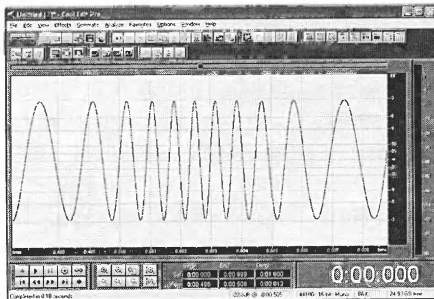
Сигнал с глубоким амплитудным вибрато, сгенерированный в редакторе Cool Edit Pro 2

При обработке вокальных партий амплитудным вибрато нужно пользоваться очень осторожно. Глубина его не должна быть большой, тремоло совсем недопустимо. Пик моды на амплитудное вибрато остался в десятках лет позади. Сейчас его используют редко. И напрасно. Это очень красивый эффект.

Частотное вибрато

Суть частотного вибрато заключается в периодическом изменении частоты звукового колебания. Частотное вибрато получило широкое распространение лишь после создания электронных музыкальных инструментов. Реализовать этот эффект на акустических инструментах довольно сложно. Правда, в период расцвета ВИА появились соло-гитары особой конструкции: натяжение всех струн в них можно было одновременно менять с помощью подвижной подставки для крепления струн и рычага, то есть частотное вибрато здесь исполнялось вручную.

Реализация частотного вибрато в электромузыкальных инструментах и синтезаторах проста и естественна.



Сигнал с частотным вибрато, сгенерированный в редакторе Cool Edit Pro 2 (иллюстрация для наглядности утрирована: слишком в больших пределах меняется частота)

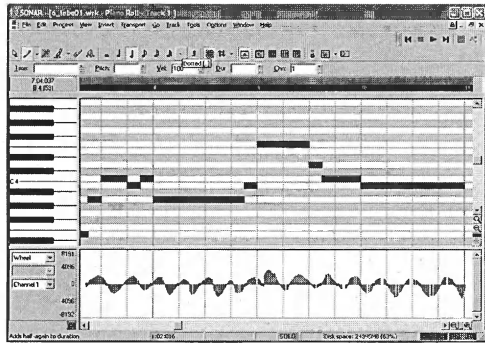
Работу всех узлов как аппаратных, так и программных синтезаторов синхронизирует опорный генератор. Если менять его частоту, то изменяться будут и частоты всех синтезируемых колебаний. Если изменение частоты происходит по периодическому закону, получается частотное вибрато. По существу, при частотном вибрато расширяется спектр исходного сигнала, а тембр периодически изменяется во времени.

Красивое звучание получается только в том случае, когда глубина частотного вибрато (относительное изменение частоты звука) невелика. Как известно, единица музыкальных интервалов, в 1200 раз меньшая, чем октава, называется центом. Интервал между соседними полутонами в темперированной гамме равен в точности 100 центам. Колебание высоты тона при частотном вибрато не должно превышать нескольких десятков центов. В противном случае создается впечатление нарушения строя инструмента.

Частотное вибрато используется и само по себе, и входит составной частью в более сложные звуковые эффекты.

С точки зрения технической реализации очень близким к частотному вибрато является эффект (точнее, исполнительский прием) глиссандо. При игре, например, на фортепиано этот прием означает скольжение пальцев по клавишам. В электронной музыке под глиссандо понимают перестройку высоты звуков взятых нот. Диапазон перестройки может достигать интервала, превышающего октаву. Ручным регулятором изменяют или напряжение, или цифровой код, которые управляют частотой опорного генератора. В электронных синтезаторах и MIDI-клавиатурах для исполнения глиссандо имеется специальный элемент управления: колесо или рукоятка, а стандартом MIDI предусмотрено специальное сообщение, передаваемое при изменении состояния манипуляторов высоты тональной перестройки.

В отличие от первых электромузыкальных инструментов эти средства позволяют выполнять не только глиссандо, но и ручное (иногда гово-



Графическое управление перестройкой высоты тона в программе Sonar 2 XL: в верхней секции окна вы рисуете "отпечатки клавиш" для своей мелодии, а в нижней — график перестройки тона

рять "пальцевое") частотное вибрато. Закон колебания высоты звука подчиняется воле исполнителя, и эффект перестает быть механическим и монотонным.

Тембровое вибрато

Эффект тембрового вибрато также направлен на изменение спектра звуковых колебаний. Физическая его сущность состоит в том, что исходное колебание с богатым тембром пропускается через полосовой частотный фильтр, у которого периодически изменяется либо частота

настройки, либо полоса пропускания, либо по различным законам изменяются оба параметра. При этом фильтр выделяет из всего спектра исходного колебания те частотные составляющие, которые попадают в "мгновенную" полосу его пропускания. Так как полоса пропускания "дышит" по ширине и "гуляет" по частоте, тембр сигнала периодически меняется.

Кроме автоматического тембрового вибрато используют еще и ручное (чаще даже "ножное" — с управлением от педали). Такой вариант эффекта известен под названиями "Вау-вау", или "квакушка".

Красиво звучат протяжные аккорды, взятые на электрогитаре, сигнал которой пропущен через блок тембрового вибрато, если цикл перестройки фильтра синхронизирован с моментом возникновения колебания струны. Звук каждого очередного взятого аккорда перетекает от одного края своей тембральной области до другого.

Если звуковая карта содержит перестраиваемые резонансные фильтры или хотя бы фильтры нижних

частот с перестраиваемой частотой среза, то этот эффект может быть получен аппаратным способом в реальном времени.

Тембровое вибрато есть в арсенале средств звуковых редакторов. Например, в программе Cool Edit Pro 2 записанный аудиотрек можно обработать фильтром Dynamic EQ (см. "Магия ПК" №10/2002), предварительно "нарисовав" в окне этого эффекта графики изменения частоты настройки и добротности полосового фильтра.

Можно еще долго рассказывать о работе с виртуальными панелями управления эффектов, создающих "прекрасную дрожь в голосе". Все детали на эту тему вы можете найти в книгах серии "Секреты мастерства": "Музыкальный компьютер", "Sonar" и "Cool Edit Pro 2".

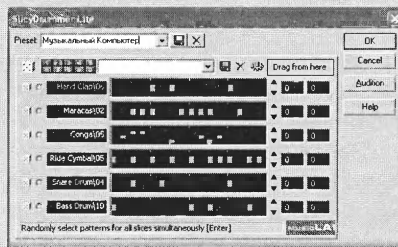
В следующем номере журнала я планирую рассказать об эффектах, в основе которых лежит задержка звука. Речь пойдет о диле, реверберации, хорусе, флэнжере и фейзере. А на сайте <http://musicalpc.com> есть форумы, специально предназначенные для обмена мнениями о работе с различными MIDI- и аудиоэффектами.

Гитары и барабаны зазвучали по-настоящему

Специалисты MusicLab, Inc. (Россия, Ростов-на-Дону, www.musiclab.com) порадовали компьютерных музыкантов несколькими новинками. Вышел в свет 21-й том библиотеки стилей гитарного аккомпанемента для MIDI-плагина Rhythm'n'Chords 2 Pro. Новая коллекция носит название "California", она объединила в себе стили, навеянные одним из хитов всех времен и народов — песней "Отель Калифорния".

Не менее интересны и разработанные ранее тома библиотеки (Jazz Pro, Latin, Latin Pop, Pop Slow, Rock Medium, Rhythm & Blues, Funk, World, Reggae, Country, Dance Techno, Blues Slow и др.). Сегодня библиотека насчитывает 1370 стилей, достоверно моделирующих игру на настоящих акустической, электро- и бас-гитарах.

В поставку популярного музыкального редактора Sonar 2.0 XL включен также MIDI-плагин MusicLab SlicyDrummer Lite. Он генерирует партии ударных инструментов, используя псевдослучайные числа и алгоритм исключения недопустимых с эстетической точки зрения комбинаций ритмических последовательностей отдельных инструментов.



MusicLab SlicyDrummer Lite пополнил ряд таких плагинов, как Rhythm'n'Chords (моделирование

гитарного аккомпанемента), Looper (управление библиотекой стилей гитарного аккомпанемента), Fixed Length (преобразование длительности нот), VeloMaster Lite (динамическая обработка значений параметра Velocity MIDI-сообщений — "уровня громкости" MIDI-нот).

Разнообразить партии ударных инструментов можно с помощью плагина MusicLab Fill-in Drummer, специально предназначенного для создания барабанных вступлений, заполнений и окончаний.

Пожалуй, проблема создания в MIDI-композициях партий гитарного аккомпанемента и сложных ритмических рисунков пользователями, не имеющими навыков игры на реальных гитаре и ударных инструментах, полностью решена. Гитары и барабаны зазвучали по-настоящему.

Юрий Петелин



ИДЕАЛЬНАЯ ПАРТНЕРСКАЯ ПРОГРАММА

Игорь Ананченко

В сети Интернет море различной информации о заработке и, в частности, о партнерских программах. Это и не удивительно — тема из ряда вечных. Практически каждый пользователь Сети рано или поздно присоединяется к хотя бы одной партнерской программе по продвижению той или иной продукции в Сети. Не стану утверждать, что каждый участник программы заработает хоть какие-то деньги, так как одно дело — желание поучаствовать в надежде без труда получить легкие деньги, и совсем другое — упорный труд и большой заработок, который, в принципе, могут дать партнерские программы.

Большинство встречающихся в Сети рекомендаций по партнерским программам адресовано новичкам, то есть это информация для участников. С информацией для организаторов дело обстоит значительно хуже. Между тем, организация партнерской программы — дело далеко

Отечественные организаторы партнерских программ обманывают так же, как и иностранные, но делают это с оглядкой, помня о том, что личная разборка как последний аргумент со стороны обманутого — вариант хоть и редкий, но не из области научной фантастики.

не простое. Организатор должен найти разумный компромисс между собственными расходами, связанными с раскруткой программы, и привлекательностью ее для участников (очевидно, что чем больше размер выплачиваемых комиссионных, тем привлекательнее партнерская программа). Другими словами, создание успешной партнерской программы — это ноу-хау, которым организаторы успешных программ не спешат делиться со своими конкурентами.

Предлагаю свою точку зрения на то, какой должна быть идеальная партнерская программа, с учетом нынешних реалий на отечественном рынке. Если вы не собираетесь создавать свою партнерскую программу, изложенная ниже информация может пригодиться при принятии ре-

шения об участии или неучастии в той или иной программе.

Национальная специфика

"Какой должна быть идеальная партнерская программа? Программа, от участия в которой вебмастер из России просто не смог бы отказаться?" — так был поставлен вопрос в электронном письме, с которым обратился ко мне владелец зарегистрированной в США фирмы, занимающейся торговлей информационной продукцией в сети Интернет.

Для организации партнерской программы необходимо решить два основных вопроса. Первый и самый важный — вопрос о том, как будет реализована программа и на кого она рассчитана. Второй вопрос

связан с применением технических средств для реализации программы. На нем я останавливаться не буду, так как существует множество готовых решений, реализованных в виде набора всевозможных скриптов, модулей и программных утилит. Системный администратор средней квалификации вполне способен настроить предлагаемое ПО с учетом требований операционной системы и конкретной конфигурации оборудования. Техническая часть реализации партнерской программы может быть довольно трудоемкой, но не требует таких умственных усилий, которые нужны для создания оптимальной партнерской программы с учетом специфики фирмы, рынка и других аспектов.

Из-за низкой покупательной способности населения России и стран СНГ иностранные организаторы партнерских программ сегодня все-речь рассматривают россиян в качестве возможных партнеров только если они являются вебмастерами. Причина тривиальна: для них это дешевая рабочая сила.

Партнерские программы, организованные отечественными фирмами, менее доходны в сравнении с зарубежными. Мой личный опыт участия в программе продажи книг интернет-магазина "Колибри" показывает, что, разместив описания книг и ссылку на форму заказа на сайте со средней посещаемостью, можно заработать не более \$50 в месяц. В плане возможности заработать партнерские программы иностранных книжных магазинов более привлекательны, но тут есть одно неприятное НО: трудности материализации заработанного в виде наличных денежных знаков.

Для интернациональной партнерской программы вопрос о том, как заплатить деньги иностранному партнеру, один из важнейших. В США этот вопрос решают с помощью именных чеков или, значительно реже, прямым перечислением денег на банковский счет партнера. Получив чек от фирмы, расположенной в США, американец несет его в ближайший банк, где ему тут же выплачивают заработанные долла-

ры. Фирма может выписать чек и гражданину другой страны, например, России. Доставка чека письмом из США в этом случае занимает примерно месяц. Затем чек сдается на инкассо (обналичивание) в один из коммерческих банков. Долларов 15 за эту операцию возьмет российский банк и еще столько же спишет иностранный. Практика показывает, что при обналичивании чека на сумму \$100 вы получите не более \$70 наличными. Причем сама процедура связана с заполнением бумаг валютного контроля, в которых заработок приходится называть материальной помощью. Вам придется расписаться в том, что деньги не получены в результате коммерческой деятельности и что вы предупреждены об ответственности. Увы, если не идти на подобное "лукавство", то расходы на операцию могут даже превысить сумму, указанную в чеке.

Перечисление денег в банк на свой валютный из-за границы связано с теми же проблемами валютного контроля, причем не менее \$30 будет снято за операцию перевода.

Вы, наверное, не раз видели по телевизору рекламу быстрых денежных переводов от Western Union. Стоимость перевода с помощью этой службы соизмерима с затратами при обналичивании чека, но оплачивает эти расходы отправитель денег. Естественно, партнерская программа, в которой есть возможность получить деньги с помощью Western Union, а не по чеку, для россиян более привлекательна. Но даже при том, что с помощью этой службы "перевести деньги так же просто, как и передать их лично из рук в руки", бумага для валютного контроля (в двух экземплярах), в которой вы расписываетесь в том, что деньги получены не в результате предпринимательской деятельности, все равно любезно прилагается.

Вывод простой и неприятный. Или не стоит связываться с зарубеж-

ными партнерскими программами и участвовать только в отечественных, малодоходных, или нужно быть готовым к тому, что за честно заработанные в иностранной партнерской программе баксы придется ответить в соответствии с законом.

Есть ли альтернатива?

Альтернативный вариант выплат в иностранной валюте существует. Можно воспользоваться одной или сразу двумя системами анонимных электронных платежей, которые доступны для отечественных пользователей: WebMoney Transfer (<http://www.webmoney.ru>) и Яндекс.деньги (<http://money.yandex.ru>), ранее известной как PayCash.

Система WebMoney Transfer в настоящее время более удобна, так как дает возможность вывода не только рублей, но и иностранной валюты (евро и доллары США). Электронные деньги в WebMoney Transfer именуется титульными знаками (1 WMZ эквивалентен 1 доллару), а, значит, получение этих титульных знаков не является получением денег. Фиксировать в налоговой инспекции получение титульных знаков не требуется (это не деньги!). Накапливать знаки можно в неограниченном количестве, как и обменивать их на товары или услуги с теми, кто согласен на такой обмен. Титульные знаки можно обменять на деньги, но после этого к полученной сумме уже применимы правила, относящиеся к денежным операциям, в том числе и налогового законодательства.

На честном слове и на одном крыле

Привлекательность той или иной партнерской программы для ее участников напрямую зависит от доверия к организаторам программы. Получить честно заработанные деньги в рамках стандартной партнерской программы от американского или иного зарубежного рабо-

Партнерская программа — программа заработка, предлагаемая владельцем интернет-ресурса (сайта, портала, сервера) пользователям Сети (партнерам). Партнеры рекламируют ресурс, привлекая на него новых клиентов, за что в качестве вознаграждения получают определенный процент от стоимости товара, проданного клиентам.

тодателя практически невозможно, если организатор сам не пожелает эти деньги выплатить. Пользователь, заполняя на сайте фирмы соглашение об участии в партнерской программе, фактически соглашается работать на честном слове, надеясь, что организатор программы честно оплатит выполненную работу. Увы, доморощенные организаторы партнерских программ иногда обманывают, как и иностранные, хотя делают это с оглядкой, помня о том, что личная разборка как последний аргумент со стороны обманутого — вариант хоть и редкий, но не из области научной фантастики.

Подводным камнем, о который разобьется иностранная партнерская программа, ориентированная на пользователей Интернета из России и стран СНГ, может стать минимальный размер выплат участникам. Поскольку соглашения в этом случае заключаются фактически на честном слове, участники программы заинтересованы в скорейшем получении денег: заработал свои \$5—10 и тут же их получил. Если организатор установил планку на уровне \$100, то для получения такой суммы придется потрудиться намного больше. Если же организатор обманет и не выплатит минимальную сумму (а в некоторых программах она достигает \$500), то для участника программы это будет весьма ощутимой финансовой потерей. С другой стороны, организатору программы просто невыгодно перечислять \$5 чеком или банковским переводом при накладных расходах в \$30 и более за саму операцию.

Учитывая все эти жизненные реалии, привожу основные характеристики идеальной партнерской программы:

1. *Весьма желательно, чтобы программа позволяла зарабатывать иностранную валюту — доллары или евро — и была ориентирована на покупателей-иностранцев из развитых стран (у них более толстый кошелек).*

2. *Лучше всего, если деньги можно будет получать с помощью*

систем анонимных электронных платежей, поскольку чеки — это слишком долго и неприятно.

3. *Необходима четкая и ясная система статистики по партнерской программе.*

4. *Желательно наличие русскоязычной службы поддержки.*

5. *Минимальная выплата заработанного — от \$5—10 и выше.*

6. *Желательно, чтобы заработанное можно было получать хотя бы раз в две недели.*

7. *Информация об организаторах должна внушать доверие.*

Светить, но не светиться...

Как организатору программы, так и партнеру перед началом работы хочется как можно больше узнать о противоположной стороне. Предоставлять же подробную информацию о себе самом, как правило, особого желания нет. На это есть несколько причин. Иногда продавец не хочет, чтобы его имя было напрямую связано с продаваемым товаром. Например, если интернет-магазин продает оргтехнику, то при высоком качестве продукции вполне уместна (и реально публикуется) реклама типа "Лучшие компьютеры в Сети у

На станциях петербургского метро много разной рекламы, как товаров и услуг, так и собственно рекламы. Один из рекламных стихов о рекламе звучит примерно так "Находит без труда телок родную маму — ну и тебя найдут, ты только дай рекламу". Наверное, рекламист, поместивший надпись на фоне коровьего вымени с прильнувшим к нему теленком, надеялся, что этот плакат вызовет у потенциального клиента чувство идилии. У кого как, а у меня эта картина вызывает ассоциации только с бедным предпринимателем, которого доит и рэкет, и налоговая, и прочие государственные службы. И текст, на мой взгляд, был бы уместней такой: "Рекламу дашь — к тебе придут, побьют и деньги отберут".

Васи Пупкина!". Однако если Вася — законный владелец секс-шопа, то у него не возникнет желания публиковать рекламу, в которой вместо лучших компьютеров рядом с его фамилией будут упоминаться суперэф-

фективные вагинальные вибраторы. Он вообще не захочет, чтобы ответственность знала о его причастности к такого рода бизнесу. О секс-индустрии в Сети я упоминал не случайно, так как предлагаемые в ее рамках партнерские программы наиболее доходны.

Вторая причина, также весьма важная для многих организаторов: не много найдется богатых людей, готовых охотно поделиться информацией о себе и своих доходах.

Применительно к системе анонимных платежей перед продавцом встает вопрос, как показать клиентам, что он является реально существующей, а не виртуальной личностью, но при этом не засвечивать без надобности свои паспортные данные, место прописки и другую информацию, которую может использовать злоумышленник. Конечно, можно вести дела анонимно, но захотите ли вы заплатить за товар хотя бы \$20 без всяких гарантий, что продавец — тот самый человек, за которого себя выдает?!

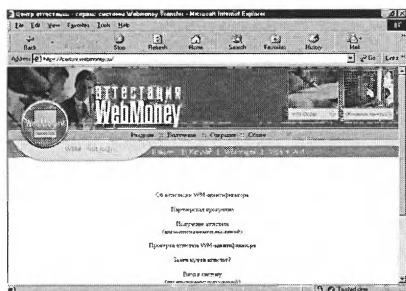
WM-аттестация

Для решения этой проблемы системой WebMoney Transfer предлагается специальная программа — аттестация платежного инструмента (WM-аттестация). Программа предусматривает выдачу цифровых аттестатов, удостоверяющих личность владельца WM-идентификатора в соответствии с его паспортными данными или данными о компании (из устава, учредительного договора и других документов компании). Изъявивший желание получить WM-аттестат добровольно отказывается от своей анонимности в системе, тем самым декларируя намерение не нарушать в ходе финансовой деятельности интересы других

участников. WM-аттестация не является обязательным условием. Каждый участник WebMoney Transfer сам решает, получать ли ему WM-аттестат на один из своих WM-идентификаторов и стоит ли вступать в парт-

нерские отношения с неаттестованными владельцами WM-идентификаторов.

Естественно, клиенты online-сервисов при прочих равных условиях предпочтут иметь дело с аттестованным поставщиком услуг. Кроме того, владелец WM-аттестата вправе рассчитывать на большее доверие своих партнеров при WM-расчетах, связанных с повышенным риском (кредитные операции, оплата в рассрочку и т. д.). При этом и физические, и юридические лица, сообщая о себе определенную информацию, имеют возможность устанавливать, какие данные будут общедоступными, а какие — доступными только для самого центра аттестации.



Программа WM-аттестации имеет три уровня, соответственно участник может получить WM-аттестат одного из трех типов: аттестат регистратора (аттестатора), персональный аттестат и аттестат доверия.

Все перечисленные типы аттес-

татов различаются правами и ответственностью их владельцев. Аттестат регистратора выдает только сам WM-центр (стоимость этой процедуры 500 WMZ), персональный аттестат может выдать один из регистраторов (от 4 до 10 WMZ), а аттестат доверия — как регистратор, так и владелец персонального аттестата.

Информацию о том, кем и когда аттестован WM-ресурс, можно узнать, сформировав запрос вида <https://certsrv.webmoney.ru/certview.asp?wmid=342489351066>, где 342489351066 — номер WM-ресурса (в данном случае моего).

Не стоит думать, что с приобретением аттестата теряется конфиденциальность. Никто кроме вла-

дельца аттестата и конкретных партнеров и покупателей не будет ничего знать о сделках. Более того, лишь владелец аттестата может выставить свою аттестационную информацию для всеобщего обозрения.

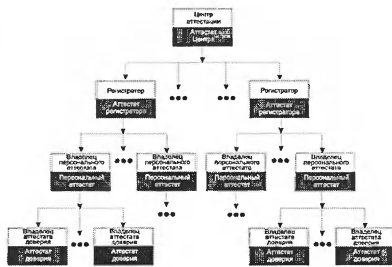
Поскольку аттестация позволяет

добиться максимального доверия участников системы, добавлю еще один пункт к перечню необходимого для идеальной партнерской программы:

8. Для используемого в партнерской программе WM-ресурса крайне желательно провести процедуру аттестации.

Спад или подъем?

Сейчас достаточно часто можно встретить суждения о кризисе электронной коммерции и трудностях, с которыми сталкивается предприниматель, ведущий бизнес в Сети. Многие пользователи жалуются на отсутствие программ для реального заработка в Интернете. Действительно, сокращается число программ, в которых организаторы выплачивают деньги участникам за простое кликанье по ссылкам и посещение ресурсов. Партнерство, основанное лишь на тупом кликанье мышкой, необратимо уходит в прошлое. Однако свертывание таких программ нельзя рассматривать как наступление кризиса электронной коммерции. Просто сегодня организаторы программ хотят, чтобы их партнерами были квалифицированные специалисты в области рекламы, электронной коммерции и разработки веб-сайтов. А это означает не кризис, а переход на более высокий качественный уровень.



Вопросы ваши, ответы наши

В.: Можно ли отправить SMS-сообщение знакомому через Интернет, не имея под рукой сотового телефона?

О.: Возможность отправки SMS-сообщений предоставляют все фирмы - операторы сетей мобильной связи второго поколения (в России - только стандарта GSM900/1800). Чтобы отправить SMS-сообщение через Интернет на мобильный телефон, подключенный к сети данного оператора надо зайти на его (оператора) сайт, например www.nwgsms.ru или www.spb.mts.ru, найти соответствующую страницу, указать номер абонента и написать текст. Некоторые фирмы (например

"МегаФон") оказывают эту услугу бесплатно всем желающим. Однако с сайта данной фирмы - оператора можно отправить SMS только на телефон, подключенный к его сети. Но как быть, если вам известен только номер телефона, но неизвестно, к какой сети он подключен? Универсальный, бесплатный и общедоступный сервис предоставляет сайт ICQ (www.icqsms.com). Узнать все подробности можно на этом сайте, а процедура отправки сообщения предельно проста. Текст сообщения отправляется по электронной почте. Адрес выглядит следующим образом: +XXXXXXXXXXXX@icqsms.com Здесь

XXXXXXXXXXXX — номер телефона в международной одиннадцатизначной нумерации. Например при отправке SMS-сообщения на мобильный телефон 999-99-99 в Петербурге адрес будет: +7812999999@icqsms.com (+7 — код страны, а 812 — код Петербурга). Если номер мобильного телефона федеральный, а не городской, например, 8-921-777-77-77, адрес будет: +7921777777@icqsms.com. Таким способом можно отправить SMS-сообщение в любую страну, любой город, в сеть любой фирмы - оператора мобильной связи. Единственное требование — надо знать коды страны и города.



САМОЗАГРУЗЧИК на PHP

Антон Орлов

Предоставить посетителям возможность загружать файлы на ваш сайт довольно просто, как и создать простейшую, но эффективную систему ограничения этой возможности с помощью пароля.

Просмотр веб-сайта обычно является "односторонним" процессом. Посетитель может просматривать сайт, получая с его страниц заложенную владельцем информацию, а если пожелает что-либо изменить на этом сайте, должен связаться с владельцем и, скажем, прислать ему свои статьи для размещения на сайте. Ну или, в крайнем случае, воспользоваться гостевой книгой или форумом сайта, чтобы поместить там свои рассуждения или мнения.

Наверняка у многих вебмастеров возникало желание дать пользователям возможность помещать свои файлы на сайте. Такая возможность весьма желательна, например, при организации файлового архива: в самом деле, почему бы каждому посетителю не поделиться с остальными любимой программой? Конечно,

в этом случае наилучшим вариантом стал бы вход на аккаунт сайта по FTP, но практически все службы хостинга выделяют лишь один логин и один пароль для такого входа. Ясное дело, предоставлять их всем подряд нежелательно.

Или вы издаете Интернет-журнал и назначили одного из своих коллег ведущим раздела. Ему ведь тогда придется присылать вам свои материалы для раздела, а труд по размещению их на сайте целиком и полностью ляжет на вас. А если обновления требуются каждый день?

Между тем, выход из положения есть — использовать возможности

загрузки файлов на сайт и работы с ними средствами PHP.

С чего начать?

Для начала следует создать папку, куда будут помещаться закладываемые файлы (скажем, files в том примере, что будет рассматриваться ниже), и присвоить ей атрибут 772, что означает предоставление посетителям сайта возможности записывать в нее файлы. Для присвоения атрибута достаточно, зайдя с помощью какого-нибудь FTP-клиента (скажем, CuteFTP) на аккаунт, выбрать из меню (в CuteFTP — из меню правой кнопки мыши) пункт CHMOD

Если у вас до сих пор нет возможности работать с PHP на своем компьютере, посетите сайт Дмитрия Бородина "PHP по-русски" (<http://php.spb.ru>) и загрузите с него два файла — описание PHP на русском языке и веб-сервер Apache с PHP-модулем, сконфигурированным для немедленного использования. На сайте есть и подробная инструкция по установке этого веб-сервера.

и указать, что виду пользователей Public разрешается в эту папку что-либо записывать.

Немного теории

Загрузка файла с веб-страницы возможна только через специальную форму. Эта форма в заголовке должна иметь параметр ENCTYPE со значением "multipart/form-data", а кроме того, специальное поле типа FILE. Оно выглядит как поле ввода имени файла с кнопкой "Обзор", нажав на которую, можно отобразить окно выбора файла. В заголовке формы также следует указать параметр ACTION, значением которого должно быть имя страницы с обрабатывающим загруженный файл сценарием. Кроме того, в форме обязательно должна присутствовать кнопка типа SUBMIT. При нажатии на эту кнопку браузер начнет передавать файл, указанный в поле типа FILE, на сервер.

Саму по себе загрузку файла на сервер умеют осуществлять практически все браузеры (за исключением самых старых моделей Microsoft Internet Explorer и Netscape Navigator), а воспринять ее могут все веб-серверы (кроме CERN и некоторых самых простых), в том числе и самый распространенный — Apache.

Файл, загруженный на веб-сервер, помещается в его временную папку и находится там до тех пор, пока сервер не закончит обрабатывать и отдавать браузеру пользователя ту страницу, имя которой было указано в параметре ACTION формы загрузки файла. После полной выдачи страницы (то есть после "ухода" к пользователю самого последнего ее символа) загруженный файл автоматически удаляется. Отсюда следует, что если с файлом планируется дальнейшая работа, на странице-обработчике должны присутствовать команды, перемещающие этот файл в какую-либо папку.

Для копирования файла используется команда *copy*, имеющая формат *copy ("имя копируемого файла (и путь к нему)", "путь к папке, в которую нужно файл скопировать и его*

новое имя там"). При использовании данной команды для работы с файлом, загруженным во временную папку, путь к этому файлу можно не указывать, так как он будет подставляться по умолчанию, а путь к папке, куда файл должен быть помещен, следует указать относительный, от того каталога, в котором находится страница с обрабатывающим загруженный файл сценарием. Заботиться об удалении загруженного файла из временной папки не стоит — это произойдет автоматически.

Той странице, которая указана в параметре ACTION заголовка формы и на которую происходит переход после завершения загрузки файла, передается несколько переменных, содержащих информацию о загруженном файле. Именно на их основе сценарий на ней сможет работать с загруженным файлом.

Вот эти переменные:

1. Переменная, имеющая то же имя, что и поле с типом file в исходной форме. Скажем, если оно выглядело как `<INPUT NAME="zakachfile" TYPE="file">`, то переменная будет иметь имя \$zakachfile. В эту переменную записывается то имя, которое загруженный файл имеет в папке временных файлов (оно создается автоматически и не совпадает с исходным именем файла). Именно с этим именем будут работать команды копирования загруженного файла в ту или иную папку.

Если в исходной форме присутствовало несколько полей типа FILE с разными именами, то для каждого из них создается своя переменная со значением, относящимся к соответствующему файлу, и все эти переменные передаются сценарию, обрабатывающему закаченные файлы. Таким образом можно загружать сразу несколько файлов за один раз.

2. Переменная, имеющая имя "Переменная 1_name" (к имени первой переменной просто приписывается с конца горизонтальная черточка и слово name: например, для указанного выше примера ее имя будет выглядеть как \$zakachfile_name). Значением этой переменной является исходное имя файла в системе отправителя — значит, можно будет

назвать загруженный файл точно так же, как он назывался на компьютере отправителя, указав полученное через эту переменную исходное имя в параметрах команды копирования.

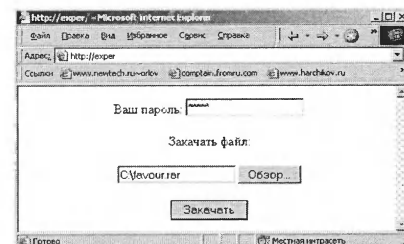
3. Переменная, имеющая имя "Переменная 1_size" (к имени первой переменной просто приписывается точно так же с конца горизонтальная черточка и слово size, например, для указанного примера ее имя будет \$zakachfile_size). Ее значение — размер загруженного файла в байтах. С помощью данной переменной можно, скажем, сортировать загруженные файлы или отказываться в загрузке файлам, превышающим определенный размер.

4. Переменная, имеющая имя "Переменная 1_type". Ее значение — тип загруженного файла согласно спецификации MIME (например, "image/gif").

Все эти переменные можно использовать в PHP-сценарии, расположенном на указанной в параметре ACTION заголовка передающей файл формы странице.

Практика

Рассмотрим устройство реально работающего кода. Для начала в него следует добавить еще один блок — авторизацию доступа, то есть сделать так, чтобы загрузка производилась только в случае правильно введенного пароля. Иначе сделать сайт неработоспособным будет довольно просто — любой недоброжелатель может поместить на него пару десятков дистрибутивов Windows XP, обрекая вас не только на необходимость заниматься массовым удалением ненужных файлов с сайта, но и на переговоры с провайдером хостинга о дополнительной оплате за истраченное дисковое



Форма ввода имени файла для загрузки и пароля

пространство. На той веб-странице, с которой должны загружаться файлы, следует поставить форму:

Код формы должен быть следующим. Сначала — заголовок формы со всеми необходимыми параметрами и их значениями:

```
<FORM ENCTYPE="multipart/
form-data" ACTION="up.php"
METHOD=POST>
```

Помните, что метод GET для загрузки файлов не подходит, поэтому не стоит его указывать в форме. Используйте метод POST, как в данном примере.

В форме должно присутствовать поле ввода пароля (естественно, если вы желаете проводить авторизацию закачивающих файлы пользователей). Его значение при передаче формы запишется в переменную с тем же именем, что и у этого поля (в данном случае — в переменную \$parol), как и при работе с любой другой формой. Эту переменную вы сможете использовать в сценарии на странице, указанной в параметре ACTION заголовка формы. Если вы укажете в параметре TYPE этого поля значение "text", то вводимые пользователем символы будут отображаться на экране и могут быть, в частности, скопированы в буфер обмена или подставлены с помощью функции автозаполнения форм (если таковая будет присутствовать в браузере пользователя), если "password" — то заместятся звездочками точно так же, как и в других полях ввода пароля, наверняка вам знакомых.

```
Ваш пароль: <INPUT
NAME="parol" TYPE="password">
```

И, разумеется, поле ввода имени файла для загрузки — типа file:

```
Закачать файл: <INPUT
NAME="upfl" TYPE="file">
```

Выглядит это поле всегда как строка для ввода текста с кнопкой "Обзор" рядом (надпись на кнопке выводится на языке локализации браузера).

Не забудьте поместить в форму кнопку начала загрузки:

```
<INPUT TYPE="submit"
VALUE="Закачать">
```

В принципе, перечисленных компонентов вполне хватает для реали-

зации системы загрузки файлов, однако при желании вы можете дополнять форму другими компонентами (скажем, полем для ввода желаемого имени файла на сайте, куда этот файл будет загружен), соответственно изменив следующий ниже сценарий.

```
</FORM>
```

Теперь не менее важная часть — сценарий обработки загруженного файла. На странице, имя которой указано в параметре ACTION заголовка формы загрузки файла, в любом ее месте следует поместить такой код:

```
<?php
```

Если в поле ввода имени файла ничего не было, то выполнение сценария прекращаем (с выводом соответствующего сообщения пользователю о данной ситуации), иначе выполняется следующая за данным условием команда elseif:

```
if ($upfl=="none")
{ echo ("Вы за-
были указать
файл...");}
```

В том случае, когда введенный пользователем пароль не соответствует тому, что указан в данном сценарии (обратите внимание, что правильный пароль для загрузки

файлов указывается именно здесь!), выполнение сценария прекращаем с выводом сообщения об этом пользователю. Если же пароль правильный, то выполняется следующая за данным условием команда elseif.

Если вы хотите назначить пользователям разные пароли (чтобы, скажем, иметь возможность впоследствии запретить загружать файлы на сайт лишь одному из них, не затронув интересы остальных), то просто укажите здесь соответствующее условие (может выглядеть так: "elseif ((\$parol!="qwert1")&&(\$parol!="qwert2"))", если пароли — qwert1 и qwert2):

```
elseif ($parol!="qwert")
{ echo ("Ваш пароль невер-
ен!");}
```

Ну и, наконец, само копирование. По его окончании пользователю выдается сообщение о том, увенчалось или нет данное мероприятие успехом. В условии elseif в нижеследующей строке проверяется, выдал ли команда copy значение True, как это должно быть при успешном копировании (команда copy, как и многие другие команды PHP, возвращает значение о результате операции, которое может быть прочитано путем присвоения его какой-либо переменной или просто при указании в инструкции if или иной).

```
elseif (copy($upfl, "files/
$upfl_name"))
```

```
{ echo("Файл $upfl_name раз-
мером $upfl_size байт успешно за-
гружен на сайт.");}
```

В том случае, если почему-либо копирование не удастся, об этом выдается сообщение. В принципе это и необязательно — неудача ко-

пирования является ошибкой PHP, и информация об этом все равно выводится с подробным описанием причин данной ошибки, однако все же стоит ясно указать пользователю на проблему. К тому же интерпретатор PHP выдает сообщения на английском языке, и не все смогут их легко понять.

Else

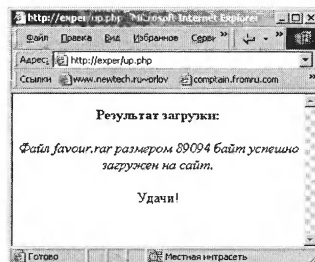
```
{ echo("Не удалось скопиро-
вать $upfl");}
```

Конец скрипта:

```
?>
```

Вот, собственно, и все. Пользователь, зайдя на первую страницу (см. рисунок), должен ввести свой пароль и найти с помощью открывающегося при нажатии на кнопку "Обзор" диалогового окна файл для загрузки. После нажатия кнопки "Загрузить" он увидит одно из трех сообщений — об успешной загрузке, о неудаче загрузки файла или о неверном пароле. Ну и, если он так и не указал файл для загрузки, об этом ему также будет сказано.

Текст данного сценария вы можете загрузить с адреса <http://www.harchikov.ru/up.rar>.



Сообщение пользователю в случае успешной загрузки

Перспектива

Весьма интересным представляется сочетание данного сценария с приемом, описанным в статье "Просто добавь статью" в октябрьском номере нашего журнала. Можно выделить коллеге-ведущему раздела отдельную папку для его статей и позволить ему загружать их туда через веб-интерфейс, как описано выше. На сайт же загруженные статьи будут выкладываться автоматически с помощью "сканера директорий", так что вашему коллеге для обновления своего раздела не придется вас даже тревожить. При реализации данного способа, конечно, желательно сделать еще нечто вроде простого "файл-менеджера", который бы позволил удалять ненужные файлы из каталога. О том, как реализовать такую возможность — в одной из следующих статей цикла.

Наверняка вы заметили, что все описанные выше сценарии имеют один существенный недостаток, который, в частности, крайне затрудняет загрузку больших файлов: переход на страницу со сценарием, обрабатывающим загруженный файл и помещающим его в нужную папку, а заодно и проверяющим правильность пароля пользователя, возможен только после того, как он целиком и полностью будет скопирован с компьютера пользователя во временную папку веб-сервера. А это значит, что пользователю, который ввел неверный пароль, чтобы получить сообщение об этом, придется ждать окончания бесполезной загрузки его файла на сервер, тратя на это свое время в Сети



и трафик. Если загружаемые файлы маленькие, это еще как-то можно потерпеть, а если они имеют мегабайтные размеры? Этак вполне

можно получить пачку неприятных писем от разозленных пользователей, потративших по часу времени в Интернете на то, чтобы в конце концов получить отказ в загрузке файла. Поэтому для создания полноценного сервиса разме-

щения пользовательских файлов необходима еще и служба авторизации пользователей, позволяющая им, единожды введя логин и пароль, в дальнейшем при работе на вашем сайте их уже не проверять. Созданию такой службы также будет посвящена одна из следующих статей о программировании на PHP.

Игры с графическими форматами: природу не обманешь

Большинство изображений, которые мы получаем из Интернета в составе веб-страниц, к сожалению, не пригодно для использования в качестве иллюстраций в текстовом редакторе: у них низкая разрешающая способность. С этой проблемой столкнулся и я. Выяснилось, что статью, написанную на основе Интернет-публикаций, проиллюстрировать невозможно, так как почти все изображения имеют непригодное для полиграфической печати разрешение 72 dpi. Первая мысль — напечатать картинки и отсканировать их с более высоким разрешением. Но реально это не придаст им необходимых 300 dpi.

Увеличение растровых картинок почти сразу ведет к пропорциональной потере четкости — ведь количество точек растра при этом не меняется. А если сделать из растрового изображения векторное? Берем тот же Visio, вставляем в него картинку BMP, сохраняем как WMF. И вот, на тебе — тяни, сколько хочешь. Казалось бы, из маленького фото люби-

мого песика можно сделать хоть плакат формата А1.

Увы, такие трюки не проходят. Вы получите файл в формате WMF, но при растягивании он не будет сохранять все детали изображения. В основе векторных форматов — изображение, полученное путем наложения большого, но конечного количества математических кривых, построенных по определенным формулам. Этих кривых недостаточно для построения изображения реального объекта средствами графического интерфейса Windows.



На самом деле для решения задачи требуется адекватный инструмент, соответствующий природе самой задачи. Поскольку изображения

реальных объектов — это набор точек растра, яркость которых меняется по случайным законам, обработку таких изображений нужно вести с помощью вероятностных или близких к ним методов.

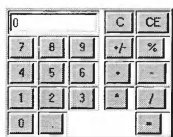
Вот тут-то меня и выручил набор встроенных фильтров программы просмотра IrfanView. Я использовал фильтр передискретизации Resize/Resample из меню Image. В окошке Width (Ширина) я установил число пикселей, равное исходному с коэффициентом 300/72, то есть с некоторым запасом умножил исходную ширину на 5 (высота устанавливается автоматически), в окошке DPI вместо 72 установил 300, в окошке Size method — Resample. Результаты работы фильтра демонстрирую, увеличив изображение кошачей мордочки: слева фрагмент изображения с исходными 72 dpi, справа — скорректированный до 300 dpi. Разумеется, эти 300 dpi — не совсем настоящие, однако некоторого повышения разрешения добиться удалось. На правом рисунке блочная структура изображения почти незаметна.

Михаил Виргулиев

Поводом для размышлений на эту тему послужили поиски скрипта обычного раскрывающегося (pop-up) калькулятора, коих в Интернете можно найти великое множество. Я не считаю это зазорным — зачем изобретать велосипед? После недолгих скитаний по сайтам для вебмастеров я остановился на компактной и непритязательной программке, написанной на javascript (<http://www.dummyproof.com/>).

Перед тем как приступить к адаптации найденного скрипта для собственных нужд я проанализировал базовые функции калькулятора и основные недостатки реализации.

Первое, что бросалось в глаза, это управление калькулятором только при помощи мыши, в то время как



большинство использует для этих целей дополнительную клавиатуру. Не понравилась также возможность

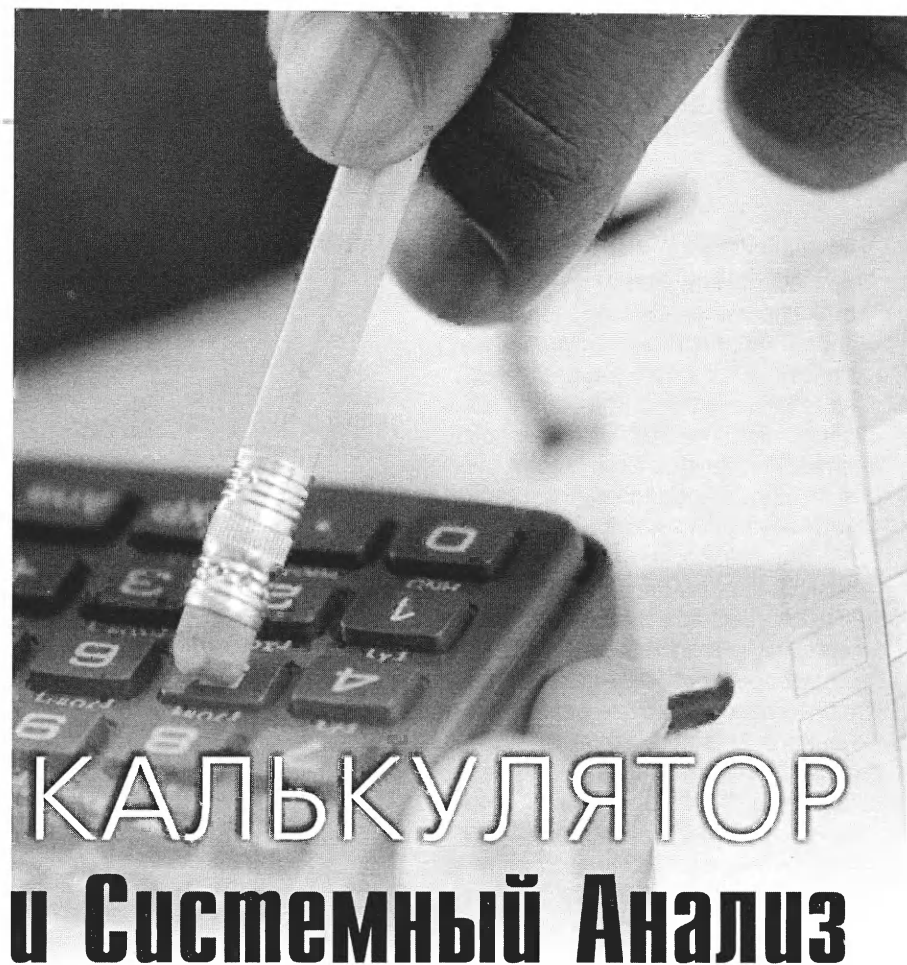
вводить в поле для чисел все, что заблагорассудится, и отсутствие кнопки заборя последнего введенного символа — вместо этого предлагалось полностью удалять последнее введенное число при помощи кнопки "CE". Ну и, наконец, тестируя операцию сложения с десятичной точкой, а именно $12.44 + 12.43$, я получил совершенно неожиданный результат 24.869999999999997 .

Пришлось констатировать неутешительный вывод — потребительские свойства скрипта не соответствуют моим представлениям о том, каким должен быть калькулятор, предлагаемый на суд общественности. Отметив про себя стройность фразы "потребительские свойства скрипта", которую можно будет удачно вернуть при разговоре с прихотливым заказчиком, я задумался — а что же это такое и от чего оно зависит? Выводы получились следующие:

1. Внешний вид, обусловленный общим дизайном.

2. Интуитивно понятный интерфейс и удобство его использования.

3. Сведение к минимуму возможных ошибок.



Олег Аверин

Любая, даже самая примитивная задача требует взвешенного и полноценного подхода. К ее конкретной реализации предъявляются те же требования, что и к реализации самого сложного проекта, включая разработку пользовательского интерфейса, анализ предметной области, реализацию алгоритма и обработку ошибок.

4. Локализация системных ошибок.

5. Компактность и оптимальность кода.

Если вдуматься, это общие требования, предъявляемые к любой сложной информационной системе, каковой может быть как отдельная программа типа игры "Косынка", так и сайт целиком. Причем в последнем случае возникают также дополнительные требования — нужно учесть разнообразие браузеров у пользователей, а в случае с серверными скриптами еще и конкретную конфигурацию серверного окружения.

В начале 90-х годов прошлого века мне довелось участвовать в разработке законченных программных комплексов, рассчитанных на абсолютно неподготовленного пользователя. Тогда менеджеры проектов уделяли особое внимание

вопросам минимизации "не вынужденных" ошибок и интуитивности интерфейса, а бета-тестеры первым делом пытались ввести в поля ввода данных заведомо ошибочную информацию.

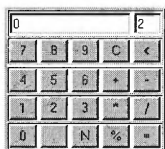
Теперь другие времена — системы стали куда более сложными, и за многочисленными возможностями различных коммерческих систем уже не видно таких очевидных вещей, как превентивная обработка ошибок пользовательского ввода. А между тем, дело это взято на вооружение хакерами, которые вскрывают серверы, вводя в строке запроса заведомо неверные данные, и проникают в систему в результате ошибок командной строки. Просто задача "защиты от дурака", превалировавшая в 90-е годы, плавно перетекла в задачу защиты от хакера.

Повышаем потребительские свойства скрипта

Вернусь к калькулятору и сформулированным выше требованиям.

Внешний вид. Поскольку это рор-ур калькулятор, вопросы дизайна сводятся к выбору цвета фона и кнопок. Для простоты оставим все как есть.

Интуитивный и удобный. Тут придется слегка поработать. Сделаем все кнопки одного размера с помощью встроенной таблицы стилей и форму курсора для кнопок — в виде руки. Добавим кнопку заборя последнего введенного символа и уберем кнопку "CE". Логически объединим цифровые и операционные кнопки, а также кнопки для редактирования строки ввода. Добавим разрядность вычислений (об этом ниже) и возможность управления с клавиатуры, используя обработчик событий `onkeypress()`.



Минимум ошибок. Автор явно игнорировал это требование. Ничего удивительного — Интернет носит ярко выраженный потребительский характер, то есть в подавляющем большинстве случаев пользователю что-то нужно найти и взять из Интернета, а не наоборот, и он подчинится всем требованиям, иначе ничего не получит. Например, при регистрации почтового ящика нужно заполнить некую форму. Правила заполнения обычно ограничиваются фразой типа "Поля, отмеченные красным цветом, обязательны", а дальше — хоть топ-топ. Зачастую такой подход переносится и на коммерческие проекты, где неверное указание почтового адреса при подписке может привести к потере потенциального клиента.

Похожим принципом руководствовался автор программы. Раз это калькулятор, пользователь будет вводить только цифры, иначе не получит результата. Слишком самонадеянно, поэтому ограничим ввод в поле значений с помощью директивы `disabled=true`. Остается поле разрядности — `name=Prc`. Для него применим контроль по событию `onchange()`, запретив вводить сюда что-либо кроме цифр.

Система. Это самая неизведанная область. Если в случае с пользовательским вводом еще можно что-то предположить, руководствуясь здравым смыслом, то ошибки системы выявляются экспериментальным путем и часто накладывают отпечаток не только на алгоритм, но и на интерфейс. В данном случае ошибки в реализации javascript-операций с плавающей запятой со стороны браузера фирмы Microsoft, полагающей, что $12.44 + 12.43 = 24.869999999999997$, заставили ввести разрядность вычислений. Впрочем, нет худа без добра — повысилась функциональность программы.

Теперь о кнопках. Логично предположить, что в качестве заборя нужно использовать `BackSpace`. Ан нет! Оказывается, функция `onkeypress()` не генерирует кода для этой клавиши — проще говоря, никак не распознается системой. Придется заменить на "<" — по виду похоже, да и для алгоритма подходяще.

Код. Для меня это единственный интересный пункт. Компактность кода и нестандартные решения являются украшением любой программы и служат источником удовлетворения от проделанной работы. Решение этой задачи начинается с анализа области данных. В случае с калькулятором мы имеем дело с тремя типами кнопок — числа и десятичная точка как источник данных, кнопки математических операций и, наконец, кнопки редактирования. Соответственно, появились три глобальные переменные — `num`, `ops` и `bedt`, отвечающие каждая за свой тип данных с учетом регистра клавиатуры.

В оригинальной версии использовались отдельные функции `Num()` для чисел и `Operation()` для операций, причем и те, и другие использовали явное значение числа или операции в качестве параметра. Изящнее воспользоваться значением `value` для каждой кнопки, оно одновременно является и именем кнопки. Именно поэтому значение "+/—" было заменено на "N", а в качестве заборя использована клавиша "<". Так конкретная реализация алгоритма наложила отпечаток на интерфейс.

В итоге получилась одна функция

вместо восьми, предложенных автором. Вкупе с повышенной функциональностью и обработкой ошибок ввода результат получился вполне приемлемым.

Отдельно стоит упомянуть о функции `prcs()`, отвечающей за округление. Этот монстр появился на свет лишь потому, что функция `round()` класса `Math` умеет округлять только до ближайшего целого числа! Просто в javascript отсутствует возможность округления с точностью до позиции десятичной точки (для IE 5.xx).

Системный Анализ

В этой статье я не ставил себе цель показать, как пишется обычный калькулятор, хотя и он может на что-то согдиться. Я хотел на простейшем примере показать, что любая, даже самая примитивная задача требует взвешенного и полноценного подхода. Требования к ее реализации те же, что и к реализации самого сложного проекта, включая разработку пользовательского интерфейса, анализ предметной области, реализацию алгоритма и обработку ошибок.

Для интернет-проектов это тем более актуально. Структура сайта создается в самом начале и во многом определяет его возможности. На нее влияют все факторы, присущие любой сложной системе. Найти баланс между разными критериями решения задачи, вписать ее в дизайн и не подменить контент алгоритмом — сложная и важная задача.

Увы, сегодня комплексному подходу при создании сайтов уделяется мало внимания. Это видно по структуре страниц, скорости работы сайтов, анализу кода скриптов, которые часто просто копируются с сайта на сайт с незначительной доработкой (в основном меняются только исходные параметры). Понятие сайта как единой сложной системы подменяется набором страниц, связанных гиперссылками.

Когда неспособность сайта решать новые задачи, неорганизованность контента и промахи в скриптах станут очевидными, остается только один выход — редизайн.

Да пребудет с вами Сила, имя которой — Системный Анализ.

Код калькулятора после доработки:

```

<HTML>
<HEAD>
<style> .b{width: 28px; cursor: hand;}
.t{width: 28px;}
</style>
<script language="javascript"> akk = 0;
fnew = 1;
fops = '';
bnum = '0123456789';
bops = '+-*/%';
bedt = 'cc<,бнτ';
function prec(n,p) {
  if(isNaN(n)) return 0; n += '';
  var an = n.split('.'); i; if((p=parseInt(p))==0) return
Math.round(n);
  if(isNaN(parseInt(an[1]))) an[1] = '';
  n = an[0]; an = an[1]; n += an.substring(0,p);
  if(an.length<=p){for(i=1;i<=(p-an.length);++i)n+='0';}
  else {n
  =
Math.round(n)+Math.round('0.'+an.substring(p,p+1));}
  an+='';n+='';for(i=1;i<=p;++i){an = n.substring(n.length-
i,n.length-i+1)+an;}
  return (an.length==n.length?'0':n.substring(0,n.length-
an.length))+ '.' +an;
}
function calc(ev) {
  var i; ev = ev.toLowerCase();
  if(bnum.indexOf(ev)>=0) {
    Clc.Out.value += ev;
    if(fnew){ Clc.Out.value = ev; fnew = 0; }
  } else if(bops.indexOf(ev)>=0) {
    i = Clc.Out.value;
    if( !(fnew && (fops != "=")) ){
      if(ev=='%') akk = (parseFloat(i)/100)*parseFloat(akk);
      else if(fops=='+') akk += parseFloat(i);
      else if(fops=='-') akk -= parseFloat(i);
      else if(fops=='*') akk *= parseFloat(i);
      else if(fops=='/') akk /= parseFloat(i);
      else akk = parseFloat(i);
      fops = ev; fnew = 1;
      Clc.Out.value = prec(akk,Clc.Prc.value);
    }
  } else if(bedt.indexOf(ev)>=0) {
    i = bedt.indexOf(ev);
    if(i<2) {
      Clc.Out.value = 0; fnew = 1; akk = 0;
    } else if(i<5) {
      i = Clc.Out.value.length;
      if(i>1) Clc.Out.value = Clc.Out.value.substring(0,i-1);
      else{ Clc.Out.value = '0'; fnew = 1; }
    } else {
      Clc.Out.value *= -1;
    }
  } else if(ev=='.') {
    if(Clc.Out.value.indexOf('.')<0){ Clc.Out.value += ev; fnew
= 0; }
  }
}
</script>
<script EVENT="onkeypress()" FOR="document"

```

```

  if(event.srcElement.name=='')
  calc(event.keyCode==13?'':String.fromCharCode(event.keyCode));
</script>
<script EVENT="onclick()" FOR="document"
LANGUAGE="javascript">
  if(event.srcElement.name=='')
  calc(event.srcElement.value);
</script>
</HEAD>
<BODY BGCOLOR=#FFFFFF TEXT=#000000>
<BASEFONT FACE="Arial" SIZE="3">
<FORM name="Clc" action="">
<table border=2 cellspacing=1 cellpadding=1>
<tr>
<td>
<input name="Out" disabled=true type="Text" style="width: 125px"
value="0">
<input class="t" name="Prc" type="Text" value="2" maxlength="1"
onchange="if(bnum.indexOf(this.value)<0){this.value='0';}">
</td>
</tr>
<tr>
<td>
<input class="b" type="Button" value="7">
<input class="b" type="Button" value="8">
<input class="b" type="Button" value="9">
<input class="b" type="Button" value="C" title="Сброс">
<input class="b" type="Button" value="<" title="Забой">
</td>
</tr>
<tr>
<td>
<input class="b" type="Button" value="4">
<input class="b" type="Button" value="5">
<input class="b" type="Button" value="6">
<input class="b" type="Button" value="+">
<input class="b" type="Button" value="-">
</td>
</tr>
<tr>
<td>
<input class="b" type="Button" value="1">
<input class="b" type="Button" value="2">
<input class="b" type="Button" value="3">
<input class="b" type="Button" value="*">
<input class="b" type="Button" value="/">
</td>
</tr>
<tr>
<td>
<input class="b" type="Button" value="0">
<input class="b" type="Button" value=".">
<input class="b" type="Button" value="N" title="+\-">
<input class="b" type="Button" value="%">
<input class="b" type="Button" value="=">
</td>
</tr>
</table>
</FORM>
</BODY>
</HTML>

```

ИГРЫ

С ТАБЛИЦАМИ

Александр Хайт

В последние годы речь о базах данных идет постоянно. Нужны они почти всем: БД "Кадры", БД "Библиотека", БД "Склад"... Большинство пользователей, кое-как набирающих текст в Word, полагают, что достаточно свести интересующую их информацию в таблицу (это текстового-то редактора) и набрать побольше данных (то есть строк этой таблицы) — и база данных готова. "А что еще нужно?" — недоумевают они.

Народ поопытнее, освоившийся с Excel, уже знает, что помещенная в электронные таблицы информация может служить не только основой для расчета по введенным формулам. С помощью автофильтра из нее просто выбирать только те данные, которые необходимы именно сейчас. А кое-кто умеет пользоваться и расширенным фильтром, составляя при этом запросы с весьма непростыми условиями. Неужели всего этого мало и нужны какие-то специальные базы данных, которые заказыва-

ют в крупных фирмах, за которые платят огромные деньги?

Раз платят, значит нужны. Поэтому и языков, ориентированных на разработку БД, достаточно много. И цена базы определяется в первую очередь не объемом информации, а удобством поиска нужных сведений, быстротой, точностью и оперативностью их получения.

Спрос на разработчиков баз данных сейчас довольно высокий, в первую очередь на тех, кто имеет опыт работы в Oracle, SQL-server, Delphi. Можно ли научиться работе в этих инструментальных средах самостоятельно? Можно, хотя и не просто. Без серьезного курса теории здесь не обойтись.

Коротко о главном

При знакомстве с базами данных нужно учитывать следующее:

1. Профессиональные БД создают программисты и даже целые коллективы разработчиков для того, чтобы ими удобно было пользова-

ся представителям иных профессий: искать, модифицировать данные, автоматизировать расчеты и т. д. Разработчик занимается заполнением БД обычно только в процессе отладки.

2. Для рядового пользователя БД — это формы и таблицы, кнопки и другие инструменты, а работа с ней ведется по принципу "что нажать, чтобы получить нужный результат?".

3. Созданная на заказ БД должна удовлетворять техническому заданию. Грамотно написанное ТЗ требует обеспечить быстродействие базы, минимизацию объема, пользовательский сервис, меры по безопасности хранения информации и регламентации доступа к ней отдельным пользователям или категориям пользователей).

4. В идеале БД должна быть сначала полностью спроектирована на бумаге, а затем реализована средствами визуального и языкового кодирования.

Почему именно Access?

Если вы хотите понять, что такое базы данных, может быть, создать несколько таких БД для домашних целей или для небольшой фирмы, то начинать лучше с Access. Чем Access симпатичен?

1. Доступность. Он входит в состав пакета Microsoft Office, который есть у каждого, хотя мало у кого установлен в полном объеме.

2. Тесная связь с другими приложениями офисного пакета. Например, простые однотабличные базы Excel легко конвертируются в Access, а отчеты, подготовленные в БД, можно редактировать в Word.

3. Визуальность. Access дает возможность просто и быстро реализовывать не очень "навороченные" проекты, на ходу постигая и возможности инструмента, и логику построения БД.

4. Универсальность. В Access разработчики компании Microsoft, насколько возможно, совместили несовместимое: средства профессиональной разработки и интерактивную среду пользователя.

5. Гибкость. Практика показыва-

ет, что даже неплохо спроектированная БД часто нуждается в доработке: то заказчик чего-то не учел, то разработчик. А бывает и так, что нужно адаптировать уже имеющуюся БД к новым условиям, и при этом не потерять данные, измеряемые десятками тысяч записей. Интерактивная среда Access вполне подходит для такой "композитной" технологии.

6. Полноценность. В Access можно применять языки SQL (специализированный язык баз данных) и Visual Basic, создавать весьма сложные и полезные БД.

И последнее. Access, возможно, наилучший вариант средства обучения, да и в реальных разработках его возможности используются крайне мало. Однако по своим возможностям он далеко выходит за рамки других офисных приложений, потому и осваивается труднее. Так что примите совет: не переходите в Access, пока не освоите прилично Excel.

Можно ли в Access все понять самостоятельно, как в Word? Вероятно, квалифицированный пользователь, уже привыкший до многого доходить своим умом, быстро разберется в том, как создать таблицу с помощью вкладки "Таблицы", мастера или конструктора. А для готовой таблицы, если воспользоваться мастерами, не составит труда разработать запрос на выборку, форму, отчет. А вот дальше желательно хоть немного почитать теорию...

Немного терминологии

База данных — это определенным образом структурированная информация. Предметы или явления, информация о которых сведена в БД, называются сущностями. В случае с записной книжкой в качестве сущности выступает человек, и для его описания мы обычно ограничиваемся наиболее важными сведениями — фамилия, имя, телефон, адрес.

Таблицы. Информацию в БД можно организовать по-разному. Самый простой и понятный способ — табличная организация. В таблицы мы обычно сводим данные об однородных сущностях. Это столбцы, которые в терминологии баз данных называют полями, или кортежами, и строки, называемые записями. Каждый столбец таблицы описывает какую-то одну сторону сущности, например, фамилию, дату рождения, сумму на банковском счете... Данные в каждом столбце всегда одного типа: текстового, числового, денежного, логического или какого-то еще (например, дата или формула, если сущность моделируется средствами электронных таблиц). Ну, а строка описывает все стороны одного экземпляра сущности. Чем больше записей-строк, тем богаче ваша БД.

Обычно разработчик БД определяет, какие поля содержит таблица, и описывает их, а оператор заполняет строки данными или удаляет их.

Опытным пользователям все это знакомо по работе с Excel. Ведь электронные таблицы предназначены не только для расчетов, но и для ведения однотабличных баз данных.

А что, есть еще и многотабличные? Да, есть. Структура, согласно которой информация представлена в БД, называется информационной моделью. Модели бывают разные, в том числе и реляционная. Если говорить кратко, такие БД состоят из нескольких связанных между собой таблиц, что позволяет минимизировать общий объем данных.

Допустим, у человека может быть несколько счетов в банке, и вовсе ни к чему в каждой записи повторять его фамилию, адрес и прочее. Нужно просто создать отдельную таблицу "Счета" и связать ее с таблицей данных о человеке одним полем, называемым ключевым. Самое простое — дать каждому клиенту банка персональный номер (универсальный идентификатор). Ясно, что в таблице "Счета" одно из полей должно содержать этот номер. Вот и связь образовалась, причем типа "один ко многим", коль скоро у одного клиента может быть много счетов.

В Access таблицы создаются в основном во вкладке "Таблицы", а связи устанавливаются вызовом схемы данных, для чего есть кнопка на панели управления. На схеме данных показаны поля таблиц, которые сконструировал разработчик. Соединяя

Сорок веков таблице умножения

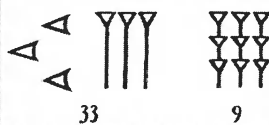
Математическими таблицами начали пользоваться в глубокой древности. Таблицы были неотъемлемой частью вычислительной практики Древнего Вавилона. Клинописные тексты записывали с помощью ромбовидной в сечении палочки на глиняных табличках — плоских прямоугольных кусочках сырой глины. Палочка оставляла клиновидные следы: острые, если глина проминалась острым углом ромба, и широкие, если тупым. Благодаря прочности обожженных табличек до наших дней дошло значительно больше математических

текстов в сравнении с Древним Египтом, где для записи использовали хрупкий папирус. Из примерно 500000 глиняных табличек, хранящихся в разных собраниях древностей, около 150 содержат математические тексты и около 200 — математические таблицы.

В большинстве случаев трудно определить, к какому периоду относятся эти тексты, поскольку многие из них были обнаружены не археологами, а местными жителями, кото-

рые занимались самостоятельными раскопками, чтобы продать находки иностранцам. Наиболее древнюю часть табличек относят к эпохе Хаммурапи — 18 век до н.э.

Вавилоняне пользовались разнообразным арсеналом таблиц: таблицами умножения, обратных значений (аликвотных дробей); процентных вычислений, квадратов и кубов натуральных чисел, пифагоровых чисел и др. Отлаженная система таблиц позволяла им довольно успешно опери-



Запись чисел 33 и 9 в Вавилонской системе счисления

мышью поля из одной таблицы с однотипными полями другой, вы можете установить связь, а затем уточнить ее параметры с помощью контекстного меню. Обладая достаточным опытом, можно спроектировать структуру таблиц и связей для достаточно сложных реляционных БД.

Когда структура БД сформирована, перед разработчиком встает задача обеспечить оператору базы комфорт. Это не только удобства при заполнении таблиц, но и возможность выборки нужных записей из одной таблицы или нескольких связанных. Например, может потребоваться узнать остатки на счетах, открытых до 1998 года, по всем клиентам, указав заодно данные об этих клиентах. Механизм запросов позволяет легко решить такую задачу.

Формы. Оператор базы данных обычно не программист. Это банковский операционист, библиотекарь или менеджер по кадрам. Он привык работать с бумажными формами. В идеальной БД пользователь ничего не знает ни о таблицах, ни о запросах. Он работает только с формами. Если изобразить привычную форму на экране, оператору будет удобно, он допустит меньше ошибок. Формы служат также для разграничения доступа к БД: сами формы или элементы управления на них открываются по паролю.

Вкладка "Формы" обеспечит разработчику БД довольно широкие

возможности. Мастер форм позволяет создавать сложные панели для просмотра и редактирования сразу нескольких связанных таблиц или запросов. А при размещении кнопок в форме появляется мастер, упрощающий описание реакции на нажатие. Надо понимать, что с каждой формой связан свой программный модуль, а то, что выводится на экран, — лишь средство визуализации.

Отчеты, макросы. Хотя электронные документы прочно вошли в нашу жизнь, обычную бумагу никто не отменил. Важнейшие решения подписываются и утверждаются пока еще в форме бумажных документов. В БД такие документы формируются во вкладке "Отчеты". И снова мастер поможет быстро и без хлопот создать документ, используя выборки и группировки данных в связанных таблицах. Далее отчет можно отредактировать в режиме конструктора и вывести на печать или конвертировать в документ Word для дальнейшей правки.

Если требуется определить реакцию на нажатие кнопки или иное событие, а разработчик плохо знаком с программированием, он может создать макрос. В Access макрос (в отличие от Word или Excel) — это программа, в которой указывается и условие, и действия, выполняемые по этому условию.

Хотя визуальные инструменты позволяют быстро реализовать мно-

гие функции хорошо спроектированной БД, в ряде случаев их может не хватить, особенно если создается оригинальный пользовательский интерфейс, решается нестандартная задача, сочетающая механизм БД и традиционное программирование, или требуется изменить имеющуюся БД, сохранив в ней максимум записей. В среде Visual Basic с применением механизма SQL реализация даже самых сложных идей оказывается весьма лаконичной. Только для этого нужно перерасти уровень пользователя.

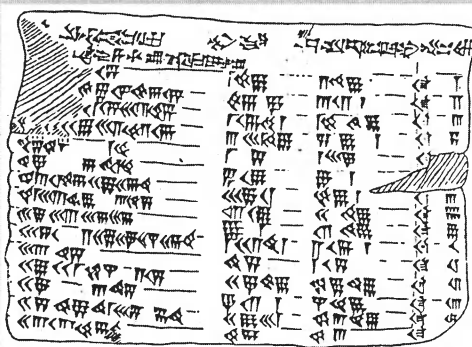
Освоить Access самостоятельно, без книг и курсов, вы вряд ли сможете. Визуальную среду — пожалуй, если поймете принцип формирования таблиц и схемы связей. А для того чтобы научиться красивым и оригинальным приемам, нужны и теоретическая база, и опыт.

Однако знания лучше всего приобретать в процессе практики. Вот к ней и приступим. Предлагаю всем интересующимся почитать какой-нибудь толстый учебник по базам данных, а в следующем номере расскажу о приемах решения довольно интересных и не совсем типичных для среды Access задач. Разобрав несколько этюдов, вы сможете поставить задачу себе и, решая ее, шаг за шагом наберетесь практического опыта, а заодно получите созданный своими руками забавный и полезный программный продукт.

ровать многозначными числами шестидесятеричной системы счисления. Широкое использование таблиц в вычислениях объясняется особенностями системы счисления Древнего Вавилона. Эта система позиционна, но не абсолютно; шестидесятерична, но не вполне; аддитивно-мультипликативна, но в записи чисел иногда использовался и субтрактивный принцип.

Естественно, возникает вопрос, почему основание этой системы счисления столь велико? Одни видят причину в том, что число 60 имеет наибольшее число делителей среди сравнительно небольших чисел, другие усматривают в этом связь с нуждами астрономи-

ческих наблюдений, третьи — с шестидесятеричной денежно-весовой системой, возникшей после объединения системы мер шумеров и аккадцев (1 талант стали считать равным 60 минам, 1 мина — 60 шекелям).



Фрагмент математической таблицы

Сложение и вычитание в шестидесятеричной системе счисления особых сложностей не вызывало. Трудности начинались при умножении и делении. В первом случае пользовались таблицей умножения (вдумайтесь, каково было бедным вавилонянам вызубрить таблицу размерами 58 x 60, если нашим школярам с трудом дается таблица всего 8 x 10). Операция деления чисел $a : b$ сводилась к умножению a на $1/b$, которое находили по таблице обратных значений (поразрядное деление еще не было изобретено). Благодаря формализму вавилонской математики это правило применяли при делении на 60 и даже на 1.

Лариса Брылевская

Пишем, а затем правим

Процесс проектирования и разработки ПО сравним с производством сложных вычислительных механизмов. Традиционные методы проектирования ПО зачастую малоэффективны, поскольку их девиз — "Пишем (код), а затем правим". В результате из-за постоянного изменения и исправления ошибок код становится запутанным и малопонятным, а по мере усложнения системы конечный продукт получается настолько сложным, что даже сами программисты-разработчики боятся обслуживать и поддерживать его.

Да и рынок диктует разработчикам свои условия в виде все новых технологий, так что продукты, разрабатываемые в рамках долгосрочных проектов, иногда устаревают, так и не увидев свет.

Все эти условия и ограничения заставляют полностью пересмотреть тот подход, которым еще недавно руководствовались менеджеры в различных IT-компаниях.

**Планирование,
Проектирование
и Конструирование**

Когда некая система создается в соответствии с принятыми прототипами и на основе жестких спецификаций, основным критерием качества ПО становится всестороннее планирование. Центральную роль в таком проекте играет системный архитектор — он один владеет всей информацией о реализуемой задаче и является единственным ответственным лицом за изменение всей спецификации проекта. При таком подходе время, отведенное на дизайн, планирование и проектирование всей системы в несколько раз превышает собственно конструирование конечного продукта. Он оправдан в таких сферах, как строительство или машиностроение, но программное обеспечение — субстанция тонкая и виртуальная, зависящая от внешних изменений, поэтому очень трудно на начальном этапе предусмотреть все характеристики и состояния будущей системы.



**ЖР:
ЭКСТРЕМАЛЬНОЕ
ПРОГРАММИРОВАНИЕ**

Андрей Озеров

*Развитие — ...процесс подсознательный, который сразу же прекращается, когда о нем начинают думать.
Б. Шоу*

**Анализ, Взаимодействие
и Обучение**

Современные менеджеры и руководители IT-компаний вынуждены искать более эффективные методы организации процесса при создании сложных систем. Например, все большую популярность обретают так называемые адаптивные методики (ASD — Adaptive Software Development), где определяющие факторы — взаимодействие, выраженное через общение участников проекта, качество и естественная простота конечного продукта. Такой процесс в первую очередь ориентирован на быстрое создание программного продукта при достаточно низких затратах на производство и дальнейшее обслуживание системы. Жизненный цикл проектов на основе ASD можно описать такими терминами: анализ, взаимодействие и обучение. Интеграция этих понятий подразумевает непрерывные изме-

нения внутреннего состояния системы при тесном взаимодействии каждого члена проекта. Процесс создания ПО при этом разбивается на мелкие этапы-итерации (milestones), что позволяет концентрироваться на упрощении системы (refactoring). Данный подход хорошо проиллюстрировал Эдвард Туфт в виде упражнения для начинающих разработчиков графов: "Нарисуйте граф так, как хотите, затем стирайте до тех пор, пока вы не начнете удалять из графа действительно полезную информацию. Таким образом вы получите наиболее удачный дизайн для графа". Конечным итогом в этом случае является простое по своему интерфейсу и дизайну ПО.

К числу адаптивных методик можно отнести Lean Development, SCRUM, семейство методологий Crystal, а также экстремальное программирование. Все они направлены на достижение одной цели —

сделать процесс разработки ПО ориентированным на человеческий фактор. Иными словами, человеческие ресурсы рассматриваются в них в качестве основного двигателя проекта по созданию качественного программного продукта.

Однажды и Только Однажды

Методика экстремального программирования (XP) в последнее время все больше ассоциируется с лозунгом "Однажды и только однажды" (OAOO — Once and Only Once). Термин "однажды" в рамках XP подразумевает предельную простоту и доступность системы. Вот как определяет простоту системы Кент Бек, один из основоположников теории экстремального программирования:

1. Система (программа) должна выражать собой все, что вы хотите сообщить о ней всем остальным участникам проекта;

2. Система не должна содержать дублирующегося кода;

3. Система должна состоять из наименьшего количества объектов и классов;

4. Система должна содержать в себе наименьшее количество методов, а значит, и наименьшее количество взаимосвязей.

По сути экстремальное программирование — это набор правил и принципов, призванных помочь группе разработчиков систематизировать и качественно улучшить процесс проектирования ПО.

В соответствии с практикой экстремального программирования весь набор принципов и характеристик процесса создания программных продуктов сводится к следующим понятиям:

1. Игра в планирование (planning game). Планирование в XP абстрагирует процесс разработки ПО для двух участников — Бизнес (Business) и Разработчики (Development), при объединении которых вырабатывается наиболее ценное бизнес-решение за довольно короткий период времени. Термин "игра в планирование" может иметь различные толкования, но в своей основе они опираются на такие категории:

- Карточки историй (story cards), каждая из которых представляет собой упрощенную схему реализации того или иного бизнес-процесса в рамках всей системы. В основном такие истории пишутся на карточках размером 4x6 с указанием заголовка и описания проблемы;

- Оценка затрат на реализацию всего набора историй и каждой в отдельности, определяя таким образом количество итераций, через которые проходит весь процесс создания системы. Каждая итерация формирует определенный этап развития ПО. Частота одной такой итерации варьируется в пределах 1—3 недели.

- Анализ ситуации со стороны заказчика, который решает, какие истории следует реализовывать в первую очередь, а какие можно оставить до следующего релиза продукта.

Термин "экстремальное программирование" получил признание благодаря трудам Кента Бека и Варда Каннингхема в процессе разработки объектно-ориентированных систем с использованием языка программирования Smalltalk в середине 90-х годов.



Кент Бек со своим коллегой и соавтором экстремального программирования Вардом Каннингхемом

2. Небольшие по функциональности релизы ПО (Smallest Release), что подразумевает реализацию строго ограниченных историй с фиксированной функциональностью в рамках каждого релиза продукта.

3. Системная метафора (System Metaphor). Каждый проект должен иметь конкретную идею-метафору, которая является точкой опоры всей системы.

4. Простой дизайн (Simple Design). Код каждого модуля программы, как и всей системы, должен быть простым и доступным, чтобы вносимые в него изменения решали настоящие проблемы, а не будущие.

5. Непрерывное тестирование (Continuous Testing). Перед тем как начать программировать функциональные возможности системы, программисты должны написать специальные тесты и только затем приступить к ее реализации. Когда весь набор тестов проходит без ошибок, работу системы можно считать адекватной и правильной.

Тесты в экстремальном программировании подразделяются на несколько типов:

- Модульные Тесты (Unit Tests) — такие тесты, которые пишут сами программисты в зависимости от требований к системе. Каждый такой тест ориентирован на проверку одного состояния или характеристики конкретного класса. В основном модульные тесты составляются с использованием специальных утилит, таких как JUnit.

- Функциональные Тесты (Functional Tests) проектируют заказчики для проверки реакции системы на различные внешние изменения.

- Параллельные Тесты (Parallel Tests) предназначены для того, чтобы доказать, что новая система работает в точности как старая.

- Стресс-тесты (Stress Tests) разрабатываются для того, чтобы симитировать наибольшую нагрузку на систему. Такие тесты используются, когда надо оценить производительность громоздких продуктов.

- Тесты Обезьян (Monkey Tests) предназначены для того, чтобы продемонстрировать реакцию системы на ввод пользователем бессмысленного набора данных.

6. Упрощение (Refactoring). Каждый член группы должен заботиться об упрощении интерфейса объектов, а также о том, чтобы код программы не содержал дублирующих структур. Упрощение кода системы возможно только благодаря постоянному тестированию и интеграции производимых изменений в конечную версию продукта на каждом этапе.

7. Парное программирование (Pair Programming). Весь разрабатываемый код системы пишется парами программистов, сидящих за одним компьютером. Такой подход обеспечивает одновременный анализ и проверку репродуцируемого кода.



Парное программирование (слева — Кент Бек)

8. Коллективная ответственность (Collective Code Ownership). Каждый член проекта должен быть в ответе за код каждого модуля. Таким образом, любой программист может внести изменения в любое место системы без значительного ущерба и затрат времени.

9. Постоянная интеграция системы (Continuous Integration). Все изменения должны быть внесены в код

действующей системы не реже чем раз в день. Такая частота интеграции обеспечивается 100-процентным выполнением всех составленных тестов.

10. 40-часовая рабочая неделя (40-Hour Work Week). Сверхурочные и работа по выходным дням значительно снижают умственную активность программистов, делая их все больше восприимчивыми к стрессовым ситуациям, что, в свою очередь, отражается на качестве всего продукта и на обстановку в команде.

11. Представитель заказчика (On-site Customer). Команда разработчиков должна иметь непрерывный и ежедневный контакт с представителем заказчика, который на правах члена группы принимает участие в разработке системы и отвечает на возникшие вопросы.

12. Стандарты программирования (Coding Standards). Постоянные изменения и ответственность за весь проект в целом каждого программиста подразумевают единые правила (стиль) программирования, делая код доступным и понятным любому специалисту в рамках проекта.

Авторы и последователи экстремального программирования считают, что только следование всем перечисленным принципам может гарантировать высокую эффективность и производительность будущей системы.

Что почитать об XP

Вопросам экстремального программирования и другим адаптивным методикам посвящено множество книг и пособий. Заинтересовавшимся читателям порекомендую книгу Кента Бека "Экстремальное программирование" (изд-во Питер, Санкт-Петербург, 2002). Различные высказывания и комментарии можно встретить на страницах многих интернет-обзоров.

Полезные ссылки:

<http://www.xprogramming.ru> — русский ресурс, посвященный проблемам экстремального программирования

[XProgramming.com](http://www.xprogramming.com) — обзор книг по XP, документация и статьи

<http://www.extremeprogramming.org/> — обзор книг по XP, документация и статьи

Вопросы ваши, ответы наши

В: После установки на компьютер нового программного обеспечения и перезагрузки операционная система Windows 2000 не загружается. Как в этом случае восстановить работоспособность системы?

О: Программное обеспечение или драйвер устройства, не сертифицированные для работы под ОС Windows 2000, могут вызвать критические ошибки в работе системы. В данном случае требуется отключить автоматический запуск соответствующего сервиса или драйвера. Для этого необходимо:

1. Загрузить компьютер с набора загрузочных? дискет Windows 2000 или CD-ROM диска Windows 2000.
2. В диалоговом окне "Welcome to Setup" нажать "R" (Recovery), затем "C" (Start Recovery Console).
3. Зарегистрироваться в системе как администратор.
4. Перейти в каталог

%SystemRoot%\System32, выполнить команду LISTSVC.

5. Просмотреть список сервисов и драйверов и найти тот, который является причиной сбоя.

6. Выполнить команду DISABLE ServiceName, где ServiceName — наименование сервиса или драйвера, который необходимо отключить.

7. Для выхода из Recovery Console ввести EXIT. Компьютер автоматически выполнит перезагрузку операционной системы.

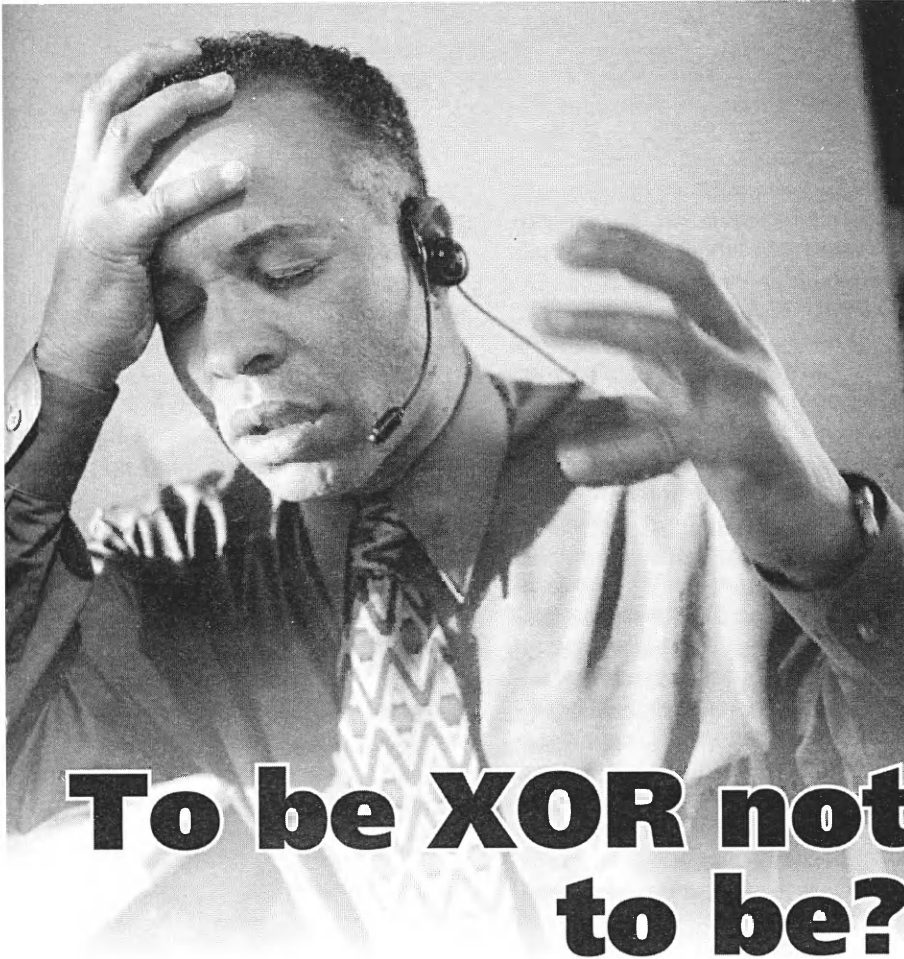
После этих действий при старте ОС отключенный сервис или драйвер не будет загружаться.

В: При регистрации пользователей в домене Windows NT используются скрипт-файлы (logon-script) для подключения сетевых дисков. Необходимо не подключать сетевые диски, если пользователь подключается к сети через сервер RAS. Как это сделать?

О: При формировании скрипта вы можете воспользоваться утилитой CHECKRAS.EXE из пакета BackOffice Resource Kit 4.0. Она позволяет узнать, подключается пользователь по локальной сети или через сервер удаленного доступа.

Пример скрипт-файла с использованием утилиты CHECKRAS:

```
if exist %windir%\checkras.exe goto yes
copy \\server\netlogon\ checkras.exe %windir%
:yes
%windir%\checkras
if errorlevel 1 goto ras
; далее идут команды для соединения по локальной сети
...
goto end
:ras
; список команд в случае соединения через RAS
...
:end
```



Антон Орлов

Логических функций довольно много. И, Или, Не — лишь некоторые из них. Но одна такая функция, с первого взгляда малозаметная, весьма примечательна. Это функция Хор — основа основ любого шифрования информации.

Для тех, кто интересуется вопросами шифрования информации, расскажу о том, какую роль в этом играет функция Хор. Команда, выполняющая эту функцию, есть практически во всех языках программирования. Я приведу пример простой программы шифрования на языке Visual Basic for Applications, встроенном в Microsoft Office.

А всего-то Хор...

Хор — некое подобие функции Или (Or). Как известно, функция Или дает на выходе 1, если хотя бы на один из входов подано значение 1. Так же действует и Хор, но, в отличие от Или, дает 0, если на входы подано сразу две единицы. Можно сказать, что Хор — это функция, которая дает на выходе 1 тогда и только тогда, когда значения, поданные на ее входы, различны.

Функция XOR		
1 вход	2 вход	Результат
0	0	0
1	0	1
0	1	1
1	1	0

Обратите внимание на интересную вещь: если на вход функции Хор подать значение 1-го входа и результат, то получится значение 2-го входа! То же самое — со значением 2-го входа и результатом: получается значение 1-го входа. Таким образом, зная итог функции Хор и значение на любом из входов, можно получить значение на другом входе. Следовательно, функция Хор обратима.

Как раз на этом и основаны принципы шифрования. Допустим, у нас есть последовательность битов, подлежащая шифрованию. Для определенности будем считать, что это текст. Понятно, что каждый символ текста может быть записан в соответствии со стандартом ASCII числом от 0 до 256 (или, в стандар-

те Unicode, двумя такими числами). После перевода этих чисел в двоичную систему счисления мы получим последовательность нулей и единиц (допустим, она будет такой: 1101010101011010100110101011...). Если представить так же пароль (пусть это будет 10010101010011), то для шифрования надо сопоставить каждый бит исходного текста соответствующему биту пароля, повторив пароль несколько раз так, чтобы у каждого бита шифруемого текста был соответствующий ему бит пароля.

Поместим одну последовательность под другой, чтобы было понятнее:

```
1101010101011010100110101011...
1001010101001110010101010011...
```

А теперь применим к каждому двум битам, стоящим в обеих последовательностях на одинаковых местах, функцию Хор. Вот что получится:

```
0100000000010100110011111000...
```

Это и есть зашифрованная паролем информация.

Для расшифровки необходимо применить ту же функцию Хор к итоговой последовательности вместе с паролем. Раз функция Хор обратима, значит, в результате получится исходная последовательность. Останется лишь превратить ее в набор символов.

Небольшой практикум

Для того чтобы вы лучше представили себе работу системы шифрования, разберем пример простой программы на VBA, использующей функцию Хор. Перед каждой строчкой кода я привожу краткое пояснение.

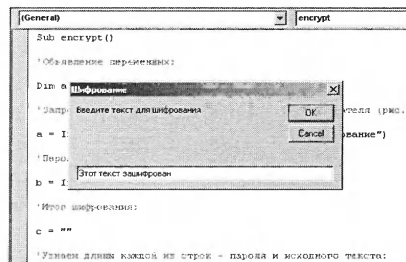
Название программы:
Sub encrypt()

Объявление переменных — присвоим им строковый тип, так как работать будем именно со строками:

Dim a, b, c, d As String

Запрос исходного текста для шифрования от пользователя через функцию InputBox. Она выводит простое окно с вопросом и строчкой для ввода текста:

```
a = InputBox("Введите текст для шифрования", "Шифрование")\
```



Запрос текста для шифрования

Запрос пароля от пользователя с помощью той же функции:

```
b = InputBox("Введите пароль", "Шифрование")
```

Обнуление переменной для записи результата шифровки. Эта переменная в результате работы программы и будет содержать зашифрованный текст.

```
c = ""
```

Для того чтобы иметь возможность применить функцию Xor последовательно к каждому биту исходной строки и пароля (при необходимости повторив пароль несколько раз), с помощью специальной функции VBA определим длины этих строк:

```
dlength = Len(a)
dlinpass = Len(b)
```

Теперь начнем шифрование, работая последовательно с каждым символом исходной строки.

```
For cn = 1 To dlength
```

"Вытащим" с помощью функции

Mid нужный символ из строки исходного текста и запишем его во временную переменную.

```
temp = Mid(a, cn, 1)
```

Затем определим ASCII-код символа командой Asc. Функция VBA Xor может работать только с числами, правда, не требуя их перевода в двоичную систему счисления (это делается автоматически). В принципе было бы нетрудно заставить работать эту функцию и с символами, предусмотрев их автоматический перевод в ASCII-коды, однако разработчики VBA этого почему-то не сделали, так что придется выполнять такую конвертацию самостоятельно. Надеюсь, вы понимаете, что логическая функция Xor и функция VBA Xor — это все-таки не одно и то же: вторая хоть и основана на первой, но имеет еще и некоторые другие алгоритмы, например, перевода чисел в двоичную систему счисления.

```
ltext = Asc(temp)
```

Теперь точно так же, с помощью функции Mid, "вытащим" соответствующий символ пароля. Поскольку пароль короче, чем шифруемый текст, символы из него берутся с начала до конца несколько раз — пароль как бы повторяется до тех пор, пока суммарная длина последовательности не станет равной длине

текста. Номер символа пароля, который должен быть взят для работы функции Xor, вычисляется на основе остатка от деления длины шифруемого текста на длину пароля.

```
temp = Mid(b, ((cn - 1) Mod dlinpass) + 1, 1)
```

Той же функцией Asc снова определим ASCII-код, теперь уже взятого символа пароля:

```
ltpass = Asc(temp)
```

И, наконец, используем функцию Xor — вот оно, собственно шифрование:

```
rez = ltext Xor ltpass
```

Для того чтобы иметь возможность записать полученное число в качестве результата, переведем его в строку функцией Str. Воспользуемся также функцией Trim, чтобы убрать вставляемые функцией Str пробелы в конце строки:

```
d = Trim(Str(rez))
```

Если хотите сократить размер программы, то разобранные выше шесть строк можно уместить в одну:

```
d = Trim(Str(Asc(Mid(a, cn, 1)) Xor Asc(Mid(b, ((cn - 1) Mod dlinpass) + 1, 1))))
```

А теперь сделаем так, чтобы каждый символ занимал ровно три позиции, независимо от величины его ASCII-кода. Для этого просто добавим нужное количество нулей в начало числа, полученного в результате

Быстро, удобно, "безбумажно"

Н у не располагает экран монитора к комфортному чтению. Иногда чтение даже самого обычного ReadMe.txt превращается в настоящую пытку словом — приходится постоянно спотыкаться и останавливаться, что особенно утомляет, когда размер этого самого ReadMe.txt "тянет" не на один десяток килобайт. Для того чтобы текстовые файлы читались с удобством и лучше воспринимались, приходится прибегать к разного рода ухищрениям: вставлять в них дополнительные элементы форматирования, менять тип или размер шрифта, подстраивая текст под индивидуальные потребности читающего.

Рассмотренная ниже программа под названием BookShelf превращает любой текстовый файл в книгу. И выглядит этот электронный заменитель привычной бумажной книги очень натурально. А главное, перелистывание страниц развернутой виртуальной книги превращает чтение в дело не только познавательное, но и комфортное.

Как это выглядит

С помощью программы BookShelf можно открывать текстовые документы разных типов. Прежде всего, конечно же, это "чистые" ASCII-файлы (файлы с расширением .txt и им подобные), файлы формата

MS Word (с расширениями .doc и .rtf) и HTML-файлы. Файлы первого типа BookShelf может открывать самостоятельно. Для открытия файлов второго типа требуются наличие в системе редактора MS Word, желательно версии не ниже 2000, хотя можно использовать и более ранние версии. Ну и, наконец, файлы третьего типа также требуют наличия HTML-редактора, правда только в том случае, если HTML-файл слишком сложен и помимо текста содержит ряд других элементов, таких как баннеры или комплексные мультимедийные вставки. Ко всему этому BookShelf умеет открывать ZIP-архивы для предварительного просмотра их содержимого.

применения функции Xor. Это необходимо для того, чтобы потом, при расшифровке, можно было разбить полученную строку на части, соответствующие кодам отдельных символов.

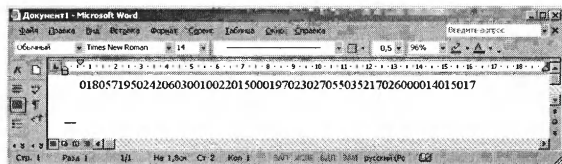
```
Select Case Val(d)
Case 0 To 9
d = "00" + d
Case 10 To 99
d = "0" + d
End Select
c = c + d
```

Вот и окончание цикла, применяемого к каждому символу исходной строки:

```
Next cn
```

Теперь в переменной с — зашифрованная строка, каждой исходной букве соответствуют 3 цифры. Эту строку цифр можно записать, например, в документ:

```
Selection.TypeText Text:=c
End Sub
```



Результат работы программы — зашифрованная строка

Данную строку чисел можно хранить или пересылать в качестве зашифрованного текста.

А теперь разберем по строкам и программу расшифровки данных. Вначале, естественно, название программы:

```
Sub decrypt()
Объявление переменных:
Dim a, b, c, d As String
```

Получение от пользователя строки для расшифровки. В это окно следует вводить ту числовую последовательность, которая получена в результате работы разобранный выше программы шифрования.

```
c = InputBox("Введите текст для расшифрования", "Шифрование")
```

```
Ввод пароля:
b = InputBox("Введите пароль", "Шифрование")
```

Переменная для записи итога расшифровывания:

```
a = ""
```

Узнаем длину каждой из строк — расшифровываемой и пароля:

```
dlinp = Len(c)
dlinpass = Len(b)
```

Собственно расшифровка (по тому же принципу, что разобранный выше, но вместо функции получения ASCII-кода символа используется обратная — Chr, а символы-цифры берутся из строки для расшифровки по три):

```
For cn = 1 To dlinp Step 3 a = a +
```

```
Chr(Val(Mid(c, cn, 3)) Xor Asc(Mid(b, (Int(cn / 3) Mod dlinpass) + 1, 1)))
Next cn
```

В итоге в переменной a — расшифрованная строка, ее можно записать, к примеру, в активный документ:

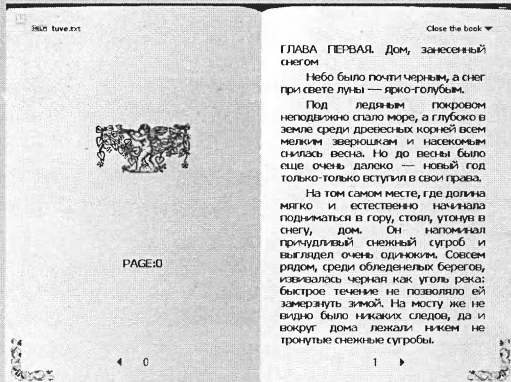
```
Selection.TypeText Text:=a
End Sub
```

Весь этот код, без сомнения, крайне прост. Особых гарантий надежности шифрования эта программа не даст. Однако для нужд сокрытия информации от незнакомых с криптографией она вполне сгодится. Если вы хотите ее использовать, то введите код в Редакторе Visual Basic for Applications и добавьте мало-мальски удобный интерфейс (как это сделать, смотрите в разделе "Учебник по VBA" на сайте <http://comptain.fromru.com>).

При желании программу можно улучшить. Например, стоит добавить в нее функцию перевода итоговой строки цифр в строку символов (попросту посчитав каждые стоящие три подряд цифры трехзначным ASCII-кодом и воспользовавшись VBA-функцией Chr для перевода его в символ), чтобы текст, получающийся в итоге шифрования, выглядел как бессмысленный набор букв. Однако имейте в виду, что при таком переводе в строке-результате шифрования

Какие же "полезности и приятности" дает BookShelf? Во-первых, можно сделать одинаковыми для всех открываемых файлов (txt, doc, rtf, html и т. д.) массу параметров самого текста: размер шрифта, режим выравнивания, цвет и текстуру фона, пространственное расположение страниц. Во-вторых, в программе работает система закладок — заложили нужное место, затем быстро открыли. В-третьих, качественно работает поиск. В-четвертых — такая приятная мелочь: если, дочитав до определенного места, вы закрыли книгу, то откроете ее именно на том месте, на той странице, на которой остановились. Ну и, конечно, книга, полученная в результате работы BookShelf, в обязательном порядке снабжается нумерацией страниц.

Требования не суровы, скорее мягко желательны. Во-первых, открываемый текст не должен содержать различных служебных символов (типа Esc или Ctrl+A), которые могут



попасть в текст при редактировании его в старом текстовом редакторе.

Во-вторых, текст должен иметь разграничители параграфов (абзацев) в виде пустых строк, дополнительных пробелов или отступов перед началом каждого блока текста, подлежащего отделению. Но на самом деле практически каждый txt-файл уже содержит разного рода форматирование "табы" и "спэйсы", так что здесь беспокоиться не о чем. Любой текстовый файл уже готов к употреблению в BookShelf. Исключения, конечно, бывают, но редко.

Такие вот книжки

В зависимости от способа подачи информации задействуются разные электронные форматы. И чаще всего при определении целесообразности использова-

вполне могут оказаться символы с ASCII-кодом до 32, то есть нечитаемые. Это приведет к тому, что ввод строки-результата в какое-либо диалоговое окно окажется невозможным, и впоследствии расшифровать текст будет нельзя. Поэтому вам придется либо записывать результат не в документ, а в файл, и точно так же, из файла, его считывать, либо разрабатывать какие-нибудь подпрограммы, отслеживающие и корректирующие подобные ситуации.

Ключ от квартиры, где деньги лежат

Наверное, вы уже поняли, что, зная зашифрованный текст и часть незашифрованного, можно с помощью той же функции XOR определить пароль, а затем с его помощью расшифровать остальную часть текста. Поэтому в современных системах защиты текста паролем используются более сложные способы, например, повторение пароля не подряд, а с определенными промежутками, с обратным расположением символов в пароле, с двойным шифрованием (результат первого шифрования снова шифруется другим паролем или в сочетании с каким-то другими алгоритмами, преобразующими текст или символы).

Та технология, что описана выше, именуется шифрованием с закры-

тым ключом — в ней для кодирования и расшифровки используется одна и та же последовательность символов в качестве пароля-ключа.

При шифровании с открытым ключом применяют две последовательности символов, причем зашифрованное с помощью первой последовательности можно расшифровать только с помощью второй, и наоборот. Обе последовательности связаны между собой определенным законом, причем из второй последовательности можно получить первую, но из первой вторую не узнаешь: обратное преобразование невозможно. Алгоритм шифрования и расшифровки в этом случае довольно сложен, только базовый программный код занимает несколько десятков страниц.

Сгенерировав специальной программой пару таких последовательностей, называемых ключами, можно первую из них сообщить тому, от кого ждешь секретных посланий (говорят, "сообщить ему публичный ключ"), а вторую хранить у себя ("оставить себе секретный ключ"). То, что будет зашифровано отправителем публичным ключом, даже сам отправитель расшифровать не сможет. Это сможет сделать только обладатель секретного ключа. Таким образом, те письма, что приходят к вам, сможете прочитать только вы.

Соответственно, если кто-то другой сообщит вам свой публичный ключ, то зашифрованные с его помощью письма прочтет только он сам, так как лишь у него есть секретный ключ, которым их можно расшифровать.

Шифрование с открытым ключом можно использовать и для подтверждения неизменности электронного текста при пересылке или хранении — это уже технология электронной подписи. При ее использовании рассчитывается контрольная сумма сообщения (скажем, сумма всех кодов букв), а затем она шифруется секретным ключом. Любой, у кого есть второй (публичный) ключ из этой пары, может расшифровать ее и, рассчитав самостоятельно контрольную сумму пришедшего сообщения, сравнить полученную величину с расшифрованной. При несопадении этих двух величин возникает подозрение о несанкционированном изменении текста сообщения в процессе пересылки (учтите, что под цифровой подписью в многих программах имеется в виду несколько другой механизм, но это уже и совсем другая тема).

Разумеется, в действительности все эти манипуляции проделывают специальные программы, а не сами пользователи. Именно на таком принципе работает механизм шифрования PGP.

Удачи!

ния того или иного формата берется в расчет либо состав самой информации (случай, когда содержание определяет форму), либо специфические условия ее прикладного окружения (яркий пример — WAP-справочники). Но текстовый формат еще долго будет тревожить умы и сердца компьютерных пользователей по причине своей нетребовательности и простоты.

Как таковой формат уже не изменить, но вот некий интерфейс к нему "прикрутить" можно. Программа BookShelf как раз и реализует такую возможность. Что же до совместимости и прочих "технических" недостатков простого текстового формата, то с ними придется смириться. Тем более что существуют форматы, специально предназначенные для

сохранения информации в той или иной специализированной форме, с гарантированной совместимостью и переносимостью.



Программа BookShelf будет успешно работать под всеми ОС семейства Windows, от 95 до XP. Распространяется она свободно. Тут,

правда, есть оговорка. Через пять дней пользования при старте программы начинает возникать маленькое диалоговое окошко, ненавязчиво предлагающее зарегистрировать данную копию программы. И все. Более никаких ограничений. Размер дистрибутива 360 Кбайт. Имеется русифицированная версия. Загрузить программу можно с сайта авторов: <http://www.text-reader.com/>. Для работоспособности некоторых элементов, в частности, симпатичных "ходячих" часов, потребуется установить Macromedia Flash Player (<http://www.macromedia.com/>) версии не ниже 5.0. Но и без плеера программа прекрасно выполняет свои обязанности, разве что выглядит чуть менее ярко.

Роман Матвеевко

ЧЕТЫРЕ ТАНКИСТА И КОМПЬЮТЕР

Андрей Озеров

“Первый — товсь! Пли!”. Такими примерно возгласами заканчивают рабочий день программисты нашей компании, когда все нормальные люди расходятся после напряженного трудового дня по своим домам. Они не выясняют отношения при помощи Age of Empires или квако-подобных “мочилок”, а оттачивают свое мастерство в тактике и стратегии ведения танкового боя. Новое интеллектуальное увлечение под незамысловатым названием Robocode уже больше полугода будоражит умы программистов всего мира, поскольку открывает новые возможности для саморазвития и, попутно, обучения Java-технологиям.

Немного истории

Опытные программисты наверняка помнят о нескончаемых игровых баталиях CoreWars, происходивших еще на заре развития компьютерных технологий. Тогда на подобном ассемблеру языке Redcode они писали маленькие программы, кото-

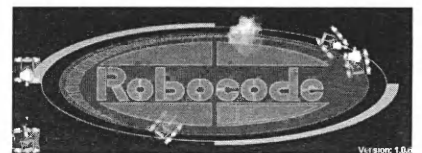
Война закончена лишь тогда, когда похоронен последний солдат.

Александр Суворов

рые боролись друг с другом в области компьютерной памяти (core) некоего абстрактного компьютера. На первый взгляд игра сводилась к написанию элементарного вируса, целью которого было завоевание всего игрового пространства. Несмотря на незатейливость сюжета, именно эта идеология легла в основу целого игрового направления, известного и ныне как “Бои роботов”. История “Боев” насчитывает множество реализаций и видов используемого ПО: от простых программ командной строки до полнофункциональных программных комплексов, графически представляющих арену битвы. Один из последних и ярких примеров такой программы — проект “Robot Battle”, позволяющий программно создавать свои танки-боты с использованием языка высокого уровня, подобного Visual Basic. Во время баталий такие танки должны были уметь стрелять по противнику, объезжать мины и собирать энергию. Мастерство программиста заключалось в написании собственно-

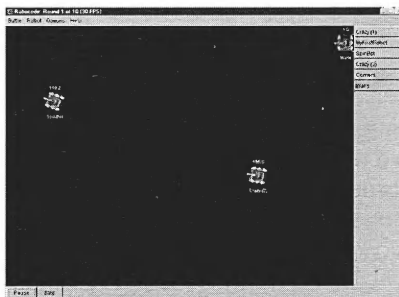
го интеллектуального “движка” для своего бота, обеспечивающего тактическое и стратегическое превосходство над танками противника.

Так или иначе, все проекты данного направления были ориентированы на игровой процесс и скорее представляли собой некое подобие логических игр. Ситуация изменилась с выходом в 2001 году программы-игры Robocode, написанной с использованием технологий Java ведущими программистами компании IBM. Основная цель этой программы — не участие в игре, где также необходимо управлять своим танком-ботом, а самообучение, изучение самого языка программирования Java.



Отцом-создателем игры Robocode стал сотрудник IBM Мэтью

Нельсон. Являясь ярким поклонником таких игр, как Warcraft, Sim City и Diablo, в середине 2000 года он не смог устоять перед соблазном создать свой игровой "шедевр". Проект Robocode по замыслу Нельсона должен был стать сиквелом игры Robot Battle, адаптированным под Java. По мере развития проекта автору и пришла в голову мысль совместить игровой процесс с обучением. Дело в том, что язык программирования Java является универсальной объектно-ориентированной технологией, предоставляющей программистам неограниченные возможности по созданию программ любой сложности — от небольших графических утилит до крупных проектов. Использование этого языка для моделирования поведения собственных объектов-роботов, по мнению Нельсона, значительно повысит интерес к Java-технологиям.



Игровое поле Robocode

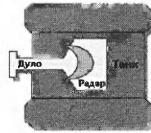
Доступность и открытость модели Robocode позволяет любому начинающему программисту за несколько минут "написать" простой танк-робот и приступить к изнурительной осаде неприятельских позиций.

Игры разума для программистов

Что же представляет собой игра Robocode? Во-первых, это графическая оболочка, состоящая из игровой панели и специального редактора (Robot Editor), где участники сражений могут составить код управления для своей боевой машины и откомпилировать его. Во-вторых, это виртуальная машина Java, которая выступает в качестве основного

движка игры, то есть отвечает за весь игровой процесс. И в третьих, это Java-классы, на основе которых программируются танки и сама игровая среда.

Архитектура танка-бота довольно проста. Танк представляет собой боевую единицу, состоящую из ходовой части, пушки и радара, который должен определять координаты противника.



Для того чтобы принять участие в сражении, вам необходимо выбрать подходящее имя своему боевому монстру, в то время как специальный мастер-помощник построит вам каркас кода, состоящий из Java-класса, унаследованного от абстрактного класса Robot. Вот как будет выглядеть ваша программа для управления танком:

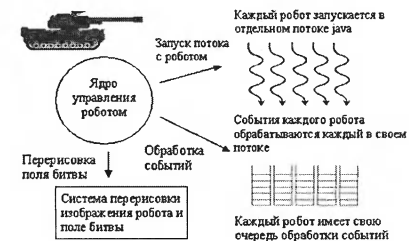
```
package test;
import robocode.*;
/**
 * My first robot
 */
public class MyFirstRobot extends Robot
{
    public void run() {
        while(true) {
            ahead();
        }
    }
    public void
onScannedRobot(ScannedRobotEvent
e) {
        fire(1);
    }
}
```

Присмотритесь к коду управления. Класс MyFirstRobot унаследован от базового класса Robot, который предлагает простейший интер-

фейс команд поведения боевой машины. В частности, ваш танк будет способен маневрировать при помощи таких команд-методов:

- turnRight(double degree) и turnLeft(double degree) — повернуть танк на заданный угол вправо или влево;
- ahead(double distance) и back(double distance) — задать направление движения вперед или назад с указанием расстояния в пикселах (если ваш танк столкнется с другим танком или стеной, движение прекратится);
- turnGunRight(double degree) и turnGunLeft(double degree) — повернуть на заданный угол пушку танка в зависимости от направления движения машины;
- turnRadarRight(double degree) и turnRadarLeft(double degree) — повернуть на заданный угол радар танка;

Увы, такой танк не сможет одновременно двигаться и стрелять. Для этого вам придется унаследовать ваш класс от другого абстрактного класса AdvancedRobot, интерфейс которого определяет более сложное поведение машины.

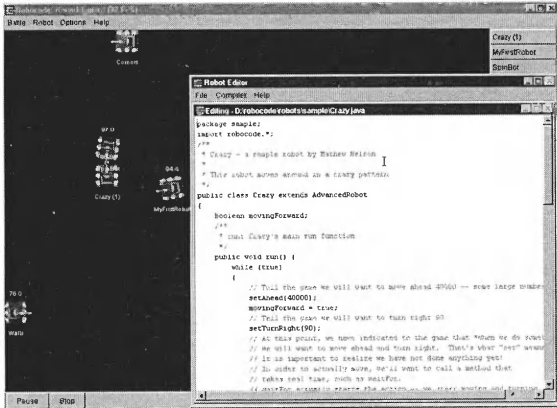


Архитектура игрового процесса Robocode

Во время сражения каждому танку-программе виртуальная Java-машина выделяет отдельный поток, областью которого и ограничено пове-

Методы и соответствующие им события

Метод класса	Событие	Описание
onScannedRobot()	ScannedRobotEvent	Во время пеленга радаром местоположения противника. В теле этого метода обычно реализуется алгоритм обстрела области, в которой был обнаружен противник.
onHitByBullet()	HitByBulletEvent	Во время попадания снарядом в корпус вашего танка. Здесь программисты должны предусмотреть алгоритм маневрирования танком.
onHitRobot()	HitRobotEvent	Во время столкновения с танком противника.
onHitWall()	HitWallEvent	Во время столкновения со стеной.



Встроенный редактор кода управления танков-ботов в игре Robocode

дение машин-ботов. В пределах данного потока ваш танк может реагировать на различные внешние события, реакцию на которые программисты могут реализовать в соответствующих методах (см. таблицу).

Например, в приведенном выше листинге в методе onScannedRobot (ScannedRobotEvent e) предлагается по умолчанию открыть огонь по противнику при помощи команды-метода fire(). Мощность удара контролируется параметром метода. В данном случае выбрана максимальная сила удара — 3.

Набор команд-методов, позволяющих управлять поведением боевой машины, достаточно обширен. Вам остается лишь правильно выбрать алгоритмы для реализации тактики сражения, и тогда победа будет за вами. Победа присваивается по результатам сражений в 10 раундах. В каждом раунде вы можете выставить в качестве противника любого из 20 танков-ботов, входящих в стандартную поставку программы, или попробовать свои силы в битве против танка вашего коллеги. Из стандартных боевых машин отмечу несколько. Например, тактика танка Corner заключается в том, чтобы как можно быстрее занять позицию в углу поля битвы и вести прицельный огонь по машинам-соперникам. Танк под названием Wall передвигается по периметру поля и довольно эффективно ведет об-

стрел всего пространства. Уйти от огневой атаки такого монстра довольно трудно. Вы всегда можете посмотреть на код реализации всех танков, которые входят в поставку программы, если сомневаетесь при выборе стратегий и поведения своей боевой машины. Код и описания для каждого танка доступны для свободного просмотра и редактирования.

Лига Robocode — лучшие из лучших

После выхода игры-программы за несколько дней с сайта IBM было скачано больше 1000000 копий. За несколько месяцев игра обрела такую популярность среди программистов всего мира, что разработчики решили устроить групповой турнир RoboLeague (<http://apps.alphaworks.ibm.com/rumble/>), в котором каждый программист со своим танком-ботом может попробовать свои силы и даже выиграть приз — ноутбук IBM Thinkpad. По результатам последних сражений победа досталась танку-боту под названием SandboxLump, созданному Эвансом Паулом. Его танк просто растерзал всех своих соперников, не оставив им ни малейшего шанса на выживание.

Русские начинают и выигрывают

В Интернете достаточно много сайтов, посвященных реализации танков-роботов. Так, на сайте самой IBM представлен целый раздел с описанием различных тактик и методик ведения танковой войны. Там можно найти полезные советы, помогающие реализовать систему движения или сканирова-

ния поля битвы на наличие других танков. Однако наибольший интерес среди участников проекта вызвала российская реализация танка с использованием SWITCH-технологий*, предложенная сотрудниками Санкт-Петербургского государственного института точной механики и оптики Н. И. Туккелем и А. А. Шальто (<http://www.softcraft.ru/auto/switch/tanks.shtml>). Наши ученые на основе теории конечных автоматов предложили модель танка, который запросто обыгрывает любого чемпиона лиги роботов. Точная математическая модель и использование современных методик программирования позволили российскому танку занять достойное место среди призеров лиги Robocode.

Если вы заинтересовались этой увлекательной игрой и хотите попробовать свои силы в программировании танка-робота, а может, просто начинаете осваивать язык Java, то для начала посетите сайт разработчиков Robocode <http://robocode.alphaworks.ibm.com/home/home.html>, где вы бесплатно получите копию игры, а также полезные советы и рекомендации.

Удачи вам, господа танкисты!

Полезные ссылки

<http://www.thetech.org/exhibits/online/robotics/robotart/> — сайт для ценителей роботов

<http://www.robotbattle.com/home.html> — сайт разработчиков игры Robot Battle

<http://www.koth.org> — Сервер игры "Бои в памяти" (Царь горы), в его стенах ведутся смертельные бои между "вирусами"



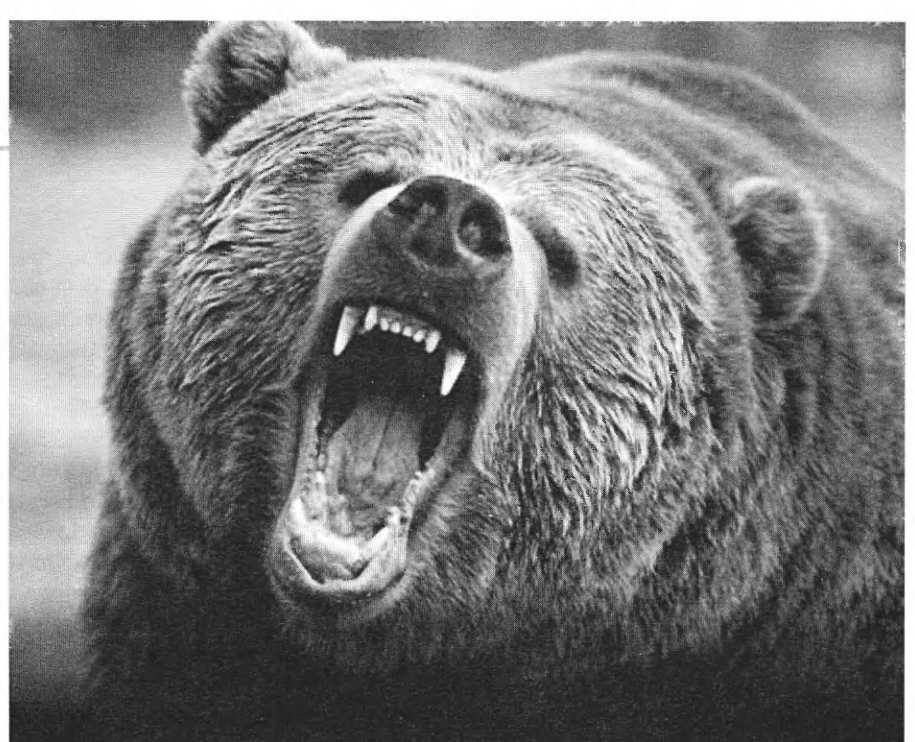
Остались вопросы?
mailto: mpc@tp.spb.ru

*) SWITCH-технологии — систематизированная парадигма программирования на основе аппаратной реализации алгоритмов использования конечных автоматов.

Самая старая, но достаточно надежная и предельно простая технология аналогового радиовещания с амплитудной модуляцией (АМ) и сегодня остается наиболее распространенной радиотехнической системой в мире. Сейчас уже около двух с половиной миллиардов радиоприемников на всех континентах принимают передачи тысяч национальных и международных станций на длинных (ДВ, 1—10 км), средних (СВ, 100—1000 м) и коротких (КВ, 10—100 м) волнах. Все эти диапазоны, расположенные на частотах до 30 МГц, предоставляют беспрецедентные возможности для дальних радиопередач по эфиру. Но именно этим диапазонам присущ и наибольший уровень шумов и помех, обусловленный самой физикой распространения радиоволн, влиянием ионосферы, магнитными бурями и т. п. Другими словами, внедрение цифрового вещания на длинных, средних и коротких волнах является одним из наиболее перспективных, но и сложных направлений. Поэтому в последние годы в мире активизировались поиски оптимального цифрового формата для радиовещания в данных диапазонах. Наибольших успехов в исследовании и реализации различных цифровых систем модуляции в полосах АМ-вещания достигли Англия, Германия, Франция и США. Россия принимает участие в испытаниях таких систем.

Стандарт DRM

Для объединения усилий специалистов и организаций из разных стран по созданию всемирного стандарта цифрового радио для диапазонов нынешнего АМ-вещания в 1996 году был создан международный некоммерческий консорциум Digital Radio Mondiale (DRM), включающий сегодня более 60 участников из 20 государств Европы, Азии и Америки. В DRM входят представители вещательных компаний, операторов сетей связи, производителей электронных компонентов и приемо-передающего оборудования, регулирующие ведомства, веща-



СИЛА ЗВУКА

Цифровое радиовещание в диапазонах ДВ, СВ и КВ

Игорь Сколотнев

тельные союзы, университеты и исследовательские центры, а также Международный Союз Электросвязи (ITU). Членами DRM в настоящее время являются такие организации и компании, как BBC, "Голос Америки", "Немецкая волна", "Радио Нидерландов", Deutsche Telecom AG, Merlin, Robert Bosch, Harris, RFI, Thomcast, Continental и многие другие. Россию в консорциуме DRM представляют Государственная радиовещательная компания "Голос России" и главный центр радиовещания и связи (ГЦУРС).

В результате нескольких лет деятельности DRM в январе 2000 года представил в ITU проект стандарта с описанием технологии, предполагающей единое для всего мира решение по организации цифрового радиовещания в диапазонах АМ. Эта технология обещает реализацию совместимой вещательной системы, позволяющей использовать недоро-

гие цифровые радиоприемники, а также существующий парк аналоговых радиоприемников и передатчиков (создана в основном на базе системы Skywave-2000 французской фирмы Thomcast). Осенью 2001 года ITU рекомендовал всем своим членам использовать систему DRM.

Внедрение ЦРВ по стандарту DRM в традиционных диапазонах АМ-вещания дает целый ряд неоспоримых преимуществ, уже подтвержденных и натурными испытаниями:

- значительное улучшение качества приема и звучания;
- отсутствие в принимаемом сигнале помех, вызванных промышленными шумами и сигналами соседних частот, а также искажений, обусловленных замираниями;
- возможность передачи аудиосигналов совместно с данными (чертежи, картинки, текстовые информационные пакеты свежих новостей, служебная информация и т. п.);

- сохранение существующих размеров зон обслуживания при значительном уменьшении (в 6 раз) мощности передатчиков, что не только дает экономию электроэнергии, но и снижает уровень "электро-смога", а, следовательно, и ущерб окружающей среде;

- возможность создания одно-частотных сетей для охвата вещанием территорий любых размеров;

- возможность использования всех уже существующих аналоговых радиоприемников и лишь немного модернизированных АМ-передатчиков;

- гибкость параметров передачи, позволяющая использовать DRM не только во всех диапазонах ниже 30 МГц (ДВ, СВ, КВ), но и на УКВ;

- возможность выбора режимов для оптимизации пропускной способности/качества и надежности/устойчивости передачи;

- высокая эффективность использования спектра: от 3 до 4 бит/Гц/с;

- открытость системы для последующего улучшения путем внедрения новых методов компрессии и процессов кодирования.

Одно из серьезных преимуществ DRM заключается в том, что ширина полосы формируемого цифрового вещательного сигнала полностью эквивалентна ширине полосы существующего аналогового сигнала и, таким образом, не требуется выделения каких-либо дополнительных диапазонов частот для организации цифрового радиовещания. Но при этом слушатели, например, коротковолнового диапазона получают качество звука, практически, не уступающее тому, которое сегодня имеется в диапазоне УКВ. Сейчас верхняя граница диапазона частот аудиосигнала в диапазоне КВ не превышает 4 кГц, а при цифровом вещании она составит от 10 до 15 кГц, причем все сегодняшние проблемы аналогового приема (замирания, помехи, интерференционные искажения и т. п.) уйдут в прошлое. Кроме этого, высокая помехозащищенность цифровых сигналов обеспечивает качественный прием на носимые и возимые

приемники, а также позволяет формировать меньше альтернативных каналов вещания на разных частотах, нежели нынешняя аналоговая технология.

Наличие в составе сигнала DRM служебной информации позволяет придать радиовещанию новые функции. Например, слушателю будет совсем не обязательно знать частоту, на которой вещает нужная ему радиостанция: достаточно будет всего лишь набрать на кнопочном пульте ее сокращенное название. Приемник сам найдет все нужные частоты и настроится на ту из них, условия приема на которой в данный момент окажутся наилучшими. В дальнейшем приемник будет автоматически отслеживать условия приема на разных частотах и выбирать ту частоту, где обеспечивается наиболее высокое качество.

Задача использования существующих аналоговых приемников в технологии DRM решена путем организации совместной передачи цифрового и аналогового вещательного сигналов. Владельцы обычных радиоприемников смогут принимать аналоговую программу, а цифровая будет прослушиваться у них лишь как слабый высокочастотный шум.

Технические аспекты

Основные технические решения для этой системы ЦРВ (принципы сжатия звуковых сигналов, способы модуляции и т. п.) были разработаны в сообществе DRM.

В соответствии с исходными идеями, система должна вписываться в существующее распределение радиочастот, а для вещания в диапазонах ниже 30 МГц сейчас используются каналы с шириной полосы 9 и 10 кГц. Из-за столь малой ширины полосы частот скорость передачи информации в одном канале системы составила всего 24 Кбит/с.

Для того чтобы добиться высококачества передачи звука на столь низких скоростях в DRM (в отличие от стандарта DAB, использующего аудиокомпрессию по технологии MUSICAM — MPEG ISO уровень II) был выбран наиболее совершенный

на сегодняшний день алгоритм кодирования натуральных звуков MPEG4 AAC (Advanced Audio Coding) в сочетании со специальным методом повышения качества передачи верхних частот Spectral Band Replicatoin (SBR), который до этого в телерадиовещании вообще не применялся. Такой комплекс технологий позволил передавать аудиосигналы с частотами до 15 кГц.

Помимо аудиосигналов в DRM могут передаваться и данные. Всего в сигнале DRM может быть до четырех (мультиплексированных воедино) цифровых потоков, каждый из которых несет звук или данные. Информация такого цифрового канала разбивается на логические кадры длительностью по 400 мс каждый.

Дополнительно к основному каналу передаются еще два сервисных ("скоростного доступа" с информацией о параметрах кодирования, модуляции и т. п., а также канал описания услуг, содержащий программу передач, ссылки на альтернативные частоты и др.), которые мультиплексируются в транспортные суперкадры длительностью 1200 мс.

Основой организации цифрового радиоканала по проекту DRM является применение высокоэффективной цифровой модуляции COFDM, уже хорошо зарекомендовавшей себя при использовании в международных вещательных стандартах DAB и DVB. Однако при выборе параметров модуляции пришлось учитывать сразу несколько взаимосвязанных и противоречивых требований:

- защитный интервал должен быть длиннее самых больших задержек, возникающих при многолучевом распространении сигналов на КВ, однако с позиций эффективности передачи желательно, чтобы он не превышал 20% от общей длительности символа;

- длительность символов может быть увеличена при увеличении числа несущих, однако количество несущих, которые могут быть размещены в полосе частот канала, ограничивается влиянием эффекта Доплера, возникающего при мобильном приеме (для снижения влияния сме-

щения частот необходимо, чтобы разность между ними раз в 20 превышала Доплеровское смещение частоты).

С учетом всех этих факторов было решено в полосе 9/10 кГц использовать всего около 200 несущих, а в качестве основных типов модуляции принять квадратурные амплитудные 64QAM и 16QAM (для более высокой помехоустойчивости может использоваться квадратурная фазовая манипуляция QPSK).

Для помехоустойчивого кодирования применяется перемежение данных и сверточное кодирование, что позволяет восстанавливать сигналы при высоких уровнях помех и селективных замираний в радиоканале. Стандартом предусмотрено четыре режима работы систем DRM, позволяющих на первом этапе внедрения передавать спектр, состоящий из цифровой части COFDM и обычной аналоговой АМ с двумя боковыми полосами, а на втором этапе — одну боковую полосу аналоговой АМ (именно за счет этого достигается совместимость с обычными

АМ-радиоприемниками). Третий и четвертый, чисто цифровые режимы радиовещания, предусматривают прием только на специальные цифровые радиоприемники.

Испытания DRM

Первые опытные цифровые передачи в диапазоне КВ начались в 1999 году, а в нашей стране — годом позже. Так, в ноябре 2000 года в одном из радиоцентров Подмосковья была организована ретрансляция программы "Голос России" по технологии DRM. Эксперимент проводился компанией Thomcast совместно с ГЦУРС. Трансляция, которая велась в КВ-диапазоне, с хорошим качеством принималась в Германии, Англии и Франции. Через месяц аналогичные испытания были проведены сразу с двумя КВ-передатчиками (в Иркутске и на острове Сайпан), сигналы которых уверенно принимались в Японии. Важным фактом стало то, что все опытные трансляции велись с помощью обычных АМ-передатчиков, лишь

дополненных цифровыми модуляторами.

Альтернативные технологии ЦРВ для АМ-диапазонов

Несмотря на согласование еще в конце 1999 года проекта стандарта DRM с рядом ведущих фирм-разработчиков систем ЦРВ Европы и Америки, для АМ-диапазонов разрабатывались и другие системы, например, система DMW, предложенная Deutsche Telecom. В качестве системы компрессии в DMW принят MPEG-2 Layer III, а в качестве системы модуляции — 32 APSK. Скорость передачи данных — 21—36 Кбит/с.

В 1999 году эта система тестировалась в Москве. Вещание велось в диапазоне СВ передатчиком мощностью 5 кВт, что обеспечило устойчивый прием в радиусе 40 км. Система проявила существенно большую помехоустойчивость по сравнению с аналоговой, однако принятый сигнал часто напоминал "машинное" звучание.

Новые технологии для сотовой телефонии

На Форуме IDF в Тайбэе (Тайвань) 15 октября Intel представила новые процессоры семейства PXA и флэш-память StrataFlash Wireless Memory, которые позволяют значительно повысить производительность, время автономной работы, а также емкость запоминающих устройств сотовых телефонов.

StrataFlash — первая в отрасли микросхема энергонезависимой памяти с многоуровневыми ячейками и напряжением питания 1,8 В, выполненная по технологии 0,13 микрон. Использование многоуровневых ячеек флэш-памяти обеспечивает высокоскоростной доступ к данным и двухкратное увеличение объема данных, хранимых в одной ячейке.

Такая ячейка потребляет почти на 40% меньше энергии, чем самые

экономичные из выпускавшихся ранее. Новые компоненты памяти будут выпускаться с емкостью 64, 128 и 256 Мбайт.

Кроме того, Intel планирует использовать новую технологию сборки компонентов памяти, получившую название "лестничной пакетной сборки кристаллов". Она позволит объединить в одном корпусе до четырех кристаллов памяти StrataFlash Wireless Memory общей емкостью до 1 Гбайт в одном корпусе. Увеличение емкости памяти на единицу объема повлечет снижение габаритов и стоимости устройств.

В представленных Intel процессорах PXA261 (200 МГц) и PXA262 (200 и 300 МГц) воплощена технология "система на одной микросхеме". Объединение в одном корпусе процессора на базе архитектуры Intel

XScale и расположенных под ним кристаллов флэш-памяти StrataFlash Wireless Memory (одного у PXA261 и двух у PXA262), то есть функций обработки и хранения данных, уменьшает количество компонентов сотового телефона, делает его еще более компактным.

Новые процессоры открывают пользователям сотовых телефонов доступ к таким функциям, требующим высокой производительности, как декодирование видео формата MPEG4, распознавание речи и рукописного текста, поддержка Java-приложений. Все это направлено на создание беспроводных устройств связи, сочетающих голосовые функции и возможности полноценного доступа к Интернету.

Пробные поставки процессоров PXA261 и PXA262 уже начались. Появление первых сотовых телефонов на их основе можно ожидать в начале следующего года. Массовое производство флэш-памяти StrataFlash Wireless Memory (128 Мбайт) начнется во втором квартале 2003 года.



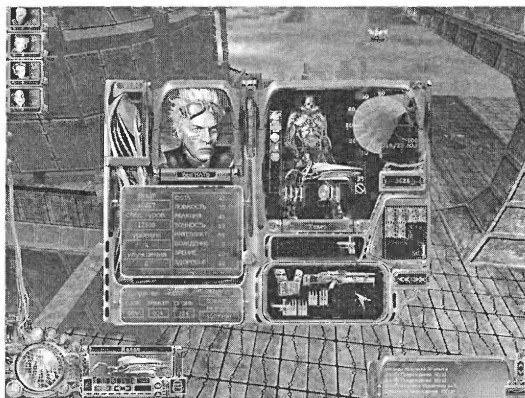
Нод доступа: РАЙ

Киберпанк? Ну, вот представьте себе не слишком далекое будущее. Скажем, летающие машины уже есть, но стоят они чертовски дорого. Встречается энергетическое оружие, хотя даже армия предпочитает проверенное временем кинетическое, стреляющее пулями. Полеты в космос еще не стали общедоступными. Физические недостатки человека вполне можно устранить путем вживления имплантов. Дорого, но уже возможно.

Земля не сгорела в ядерном пламени, не захлебнулась при повышении уровня Мирового океана и не столкнулась с астероидом. Объединенное правительство Земли переехало на космическую станцию Альфа. И хотя колонизация других планет уже технически возможна, процесс этот оказался слишком дорогим, длительным и лишенным какой-либо романтики. Поэтому несколько успешно колонизованных планет используются в основном как источники натуральных ресурсов, а дальнейшая экспансия в космические дали пока не встречает особого энтузиазма. На

марсианских шахтерских колониях, как водится, уже отгремела революция, окончившаяся войной с правительством, хотя эта война все равно ничего не изменила.

Коль скоро создать рай на земле по техническим причинам пока не представляется возможным, создается рай виртуальный — Сеть. Она же Реальность. Любому будущему добропорядочному гражданину еще до появления на свет присваивается учетный номер и вживляется разъем для подключения к Реальности.



В виртуальности созданы все условия для компенсации несовершенства реального мира. Управляет всей Сетью центральный компьютер

Системы, расположенный на той же правительственной Альфе.

На Земле, с изрядно подпорченным климатом, но все еще пригодной для жизни без специальных средств, построены огромные многоуровневые города. Привилегия видеть звезды доступна в них лишь самым обеспеченным. Нижний Город — трущобы и промзона, первый уровень одного из мегаполисов. Старые дома, темные подворотни, магазины, где вместе с буханкой хлеба и пакетом синтетического молока вам продадут практически любое оружие, не задавая глупых вопросов о наличии разрешения. В Нижнем Городе можно спрятаться от кого и от чего угодно, но можно и вообще пропасть бесследно.

Государственные законы действуют только в присутствии полиции, все остальное время господствуют местные неписанные законы. Полиция предпочитает не заниматься таким рискованным делом, как патрулирование улиц Нижнего Города, перекаладывая эту обязанность на плечи набранных из местного населения Контролеров, а то и просто специализированных ботов. Время от времени, "для галочки", организуются показательные рейды с привлечением спецподразделений и боевой техники — тут главное не попасть под горячую руку, любое неповиновение может быть истолковано как агрессия в отношении стражей порядка. Доступные цены на недвижимость несколько омрачаются воз-

можностью получить серьезную передозировку свинца в перестрелке между уличными бандами. Тем не менее, жить в Нижнем Городе вполне можно.

Главный герой этой игры, хакер (лучше даже так — Хакер), не то что бы доволен сложившимся порядком вещей, но до последнего времени не имел веских причин что-то менять в своей жизни. Работа системным администратором в местной городской управе приносила небольшой, но стабильный доход, позволяя жить, не особенно задумываясь о завтрашнем дне, а главное — обеспечивала возможность подключать-

ся к Сети. Именно в Сети он мог полностью реализовать свои способности, направленные во вполне мирное, хотя и незаконное русло — обход или взлом систем защиты. Любых систем любых организаций. Не ради наживы или стремления к сомнительной славе, а просто из спортивного интереса. Но вот однажды, находясь в Сети, в закрытых для публичного доступа областях, Хакер перехватывает некое сообщение. Даже не успев толком разобраться, что же попало ему в руки, он обнаруживает, что к его дому стягивается поднятая по тревоге полиция, а в дверь квартиры ломится спецназ Службы Безопасности Системы. Это могло означать только одно — на сей раз он крепко влип.

Уже имея за плечами два года отключения от Сети в качестве наказания за грехи молодости (это называется "условный срок"), а в перспективе — возможное стирание личности, Хакер не стал испытывать судьбу на прочность и добровольно сдаваться властям, а просто подался в бега. Правда, очень скоро он выясняет, что задерживать его живым никто не намерен, и не намеревался изначально. Полиции отдан приказ стрелять сразу на поражение. Оказавшись на улице, с пистолетом за пазухой и целой кучей вопросов, требующих срочных ответов, Хакер оказывается перед выбором — пустить себе пулю в лоб или попытаться разобраться в сложившейся ситуации.

Вот в таких условиях и начинается новая игра MIST land — Код Достапа: Рай (далее КДР).

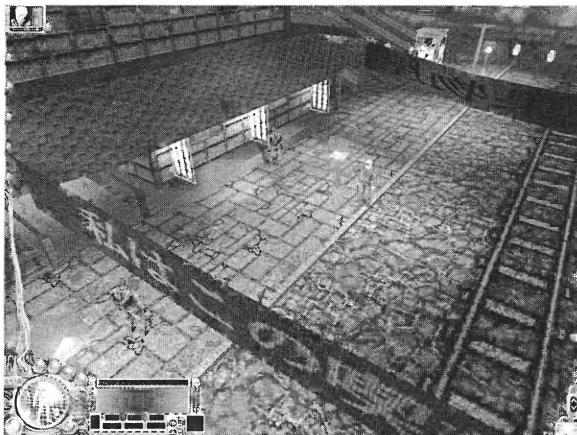
Что внутри?

Теперь попробуем разобраться с этой игрой поподробнее и копнуть, так сказать, поглубже.

Стратегическая часть

Сразу стоит сказать, что игра эта походовая, причем все время — классический вариант, без поблажек. Даже если в данном районе

кроме команды игрока нет никого вообще, очки хода (они же очки действия, action points и проч.) все равно никуда не денутся, а возможность игры в реальном времени, наоборот, не появится. Прямо как в X-Com и Incubation, хотя кое в чем КДР отличается от них довольно сильно.



Кроме того, игра достаточно сложная, в ней много боев (процентов эдак 85—90% времени). Сражения начинаются практически с самого начала, да и в уже зачищенных секторах вновь появляются враги. Естественно, подразумевается наличие в команде у игрока как минимум двух-трех соратников. Герой-одиночка в КДР долго не выживет как из-за приличного количества врагов, так и из-за достаточно высокой "стоимости" (в очках хода) выстрела из сколько-нибудь серьезного оружия. Со стрельбой связано еще несколько забавных нюансов. Хотя попадания в разные части тела и отслеживаются, но возможность вести огонь прицельно (в голову или в ноги) в игре отсутствует. Оцениваемая вероятность попадания на самом деле показывает шансы попасть не в указанную цель, а туда, куда целился данный персонаж, и даже при наличии высокоточного оружия можно промахнуться из-за элементарного неумения правильно целиться и стрелять.

Зато на каждой отдельной карте достаточно мест и для хитрых обходных маневров (благо, поле зрения всех персонажей рассчитывается честно), и для ожесточенных перестрелок. У разных противников своя

тактика ведения боя, часто даже в составе групп, что вынуждает разрабатывать адекватные ответные ходы, а не применять одну и ту же беспроигрышную комбинацию.

Одновременно в команде игрока может находиться до шести человек (и не совсем человек тоже), включая главного героя Хакера, в случае гибели которого игра заканчивается. Перемещаться по карте можно и на городском транспорте, и даже в экспропрированном на благо общего дела транспортном средстве.

Можно делать практически все, что угодно. Убить мэра? Можно. Попытаться с боем прорваться в Верхний Город, а из него в Небесный? Можно, хотя и с сомнительными шансами на успех. Работать на некоторую организацию, а потом перебежать к конкурентам? Пожалуйста. В любом районе города обязательно найдется желающий снабдить команду каким-нибудь заданием. Впрочем, его можно и не выполнять. А можно вообще просто всех убить, если патронов хватит. В общем, полная свобода перемещений и действий. Что, собственно, от хорошей стратегической игры и требуется.

Ролевая система

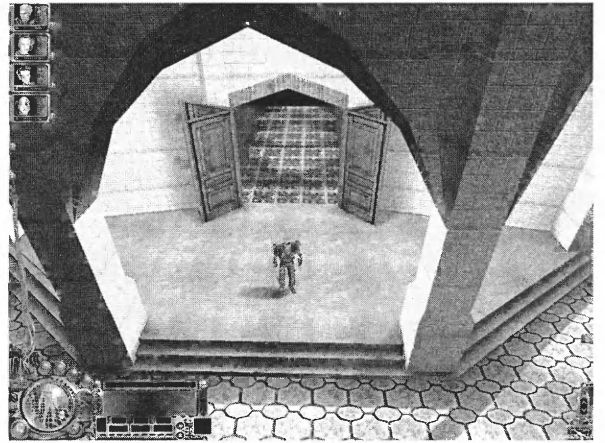
Каждый персонаж в игре помимо биографии описывается целым набором различных характеристик. На первый взгляд, с характеристиками выходит даже некоторый перебор — их восемь. При более детальном рассмотрении выясняется, что все подобрано верно. Сила отвечает непосредственно за количество очков хода у персонажа и за максимальный вес инвентаря, а ловкость определяет "стоимость" каждого действия (например, шаг вперед). Реакция позволяет персонажу автоматически стрелять по врагам во время их хода. Интеллект необходим для выполнения сложных действий (например, взлом электронного замка) и для использования высокотехнологичных предметов экипировки и оружия.

Точность, зрение и здоровье говорят сами за себя. Есть еще сопротивляемость воздействию болевого шока, огню и электричеству, а также мораль, но эти характеристики определяются наличием брони, имплантов, а в случае морали — текущей обстановкой.

Стоит также отметить, что снижение здоровья влечет за собой снижение ловкости и точности: раненый персонаж становится менее эффективным. Специальных навыков и умений нет (если не считать, что Хакер умеет взламывать электронные устройства, начиная с банкоматов и кончая полицейскими роботами). Любой персонаж может использовать любой предмет/оружие или управлять любой техникой, если он отвечает «минимальным системным требованиям». Для оружия, как правило, определяющие навыки — сила и интеллект. С накоплением опыта персонажи повышают свой уровень и получают некоторое количество бонусных очков для распределения между основными характеристиками. Число очков для каждого персонажа строго фиксировано и не зависит ни от каких внешних факторов.

Почти все характеристики можно изменять с помощью установки соответствующих имплантов, но при этом надо учитывать, что установить их можно максимум 8, а возможность удалить неугодные отсутствует. Вся ролевая система подведена под разделение персонажей по специализации, и, при беглом взгляде на набор характеристик, сразу можно опреде-

лить, кем может быть в команде потенциальный новобранец. Возможность перекрыть интеллектуала в бойца-терминатора, а снайпера в водителя, практически отсутствует. В результате требуется аккуратный подбор оптимальной команды с последующим целенаправленным развитием каждого персонажа.



Высокие технологии

Движок КДР и выдаваемая им картинка способны удивить, пожалуй, только человека, совершенно не знакомого с современной трехмерной графикой. Достаточно скромные модельки, пусть и неплохо анимированные, отсутствие буйств архитектуры, многоэтажных зданий, тотального разрушения и голливудских спецэффектов хотя и скрашивается удачной общей стилистикой и кое-какими мелочами (типа отображения на персонажах надетой брони и используемого оружия), но все же отнюдь не являются главными достоинствами игры. Картинка не отвратительна, но скучна.

Гораздо интереснее дело обстоит с музыкой и озвучкой — для КДР написан отличный саундтрек, практически в каждом районе города уникальная музыка, которая еще и меняется во время боя в зависимости от количества врагов — поддерживает, так сказать, боевой дух. Музыка на пятерку. Каждый персонаж в команде у игрока очень разговорчив, на любое действие записано несколько разных фраз, включая комментарии по поводу открывания дверей. Даже умирая, вместо предсмертных

хрипов, лежащий в луже собственной крови герой может разродиться какой-нибудь тирадой. Враги тоже не гнушаются переговариваться между собой, звать подмогу и осыпать игрока проклятиями. Только что кулаками не грозят. В результате крупная перестрелка озвучивается не только выстрелами, но и пререканиями участвующих сторон. В общем, со звуком все тоже очень даже в порядке.

Про интерфейс управления ничего особенного сказать нельзя. В меру удобный, с парой ляпов (авторы обещали в первом патче исправить), глаза не мозолит и играть не мешает.

Подводя итоги

Код Доступа: Рай — отличная проходная стратегия для ценителей жанра и желающих к нему приобщиться. В игре продуманная и отлаженная механика, скрытая под достаточно скромной графикой. Жесткая ролевая составляющая несколько ограничивает свободу развития персонажей, но зато повышает общий интерес к игре. Полная свобода действий дает возможность поступать именно так, как игрок считает нужным. Отличный набор для победы. Но...

Людям, мало интересующимся подобными вещами, КДР покажется серой, скучной, совершенно не привлекательной и чертовски сложной игрой.

Игра не для всех. И выбор предстоит сделать вам.

Showstopper

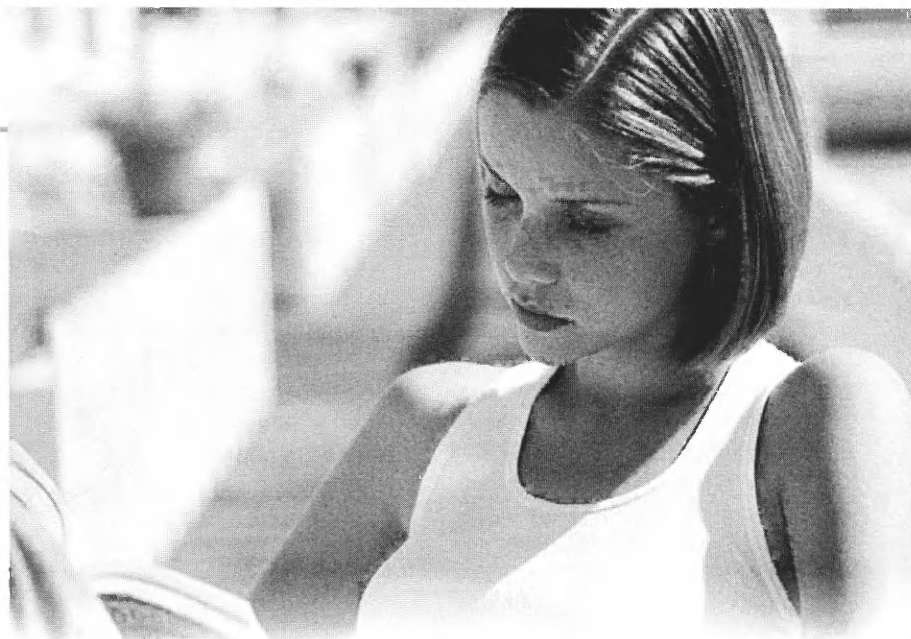


MegaSoft eWriter v3.0 rus

Свершилось! Наконец-то в наших магазинах появилась локализованная версия известной программы, при помощи которой были созданы такие бестселлеры, как "Вечера на ранчо близ Сан-Диего" (Evenings in the ranch near San Diego) и "Три Зеленых берета и морпех" (The three Green Berets and Marine). Напомним читателям, что версия 1.0 умела создавать только учебники для средней школы, в версиях 2.0 и 2.5 уже можно было делать газетные очерки и критические статьи. Версия 3.0 дает возможность создать полноценное литературное произведение. Такие писатели, как Стефан Квин и Мэри Ланкастер в своем творчестве постоянно используют MegaSoft eWriter. В базу данных русской версии включены все авторы, опубликовавшие хотя бы одно произведение на русском языке. Список огромен: от Пушкина и Льва Толстого до малоизвестных писателей (беру наугад), таких как Григорий Власов или Василий Пупкин.

Приступить к написанию романа вы можете сразу же после инсталляции. "Файл" — "Создать", и в появившемся диалоговом окне вам предстоит выбрать форму и жанр будущего произведения. Вы можете выбрать роман, повесть, рассказ, новеллу или поэму. Из жанров в стандартный комплект поставки входят детектив, женский роман, путевые заметки и фэнтези. Можно вписать и свои жанры и формы, но работа эта требует филологического образования, а потому проще дополнительные модули заказать у производителя. Сейчас на сайте компании MegaSoft (www.MegaSoft.com) доступны модули для создания исторических романов: Древний Рим и Древняя Греция, Раннее Средневековье, Ренессанс. Для русской версии вы можете заказать следующие модули: Киевская Русь, Великая смута, 17 век, Первая и вторая мировые войны.

Следующим шагом вам предстоит определить возраст будущих читателей, их социальный статус и образовательный ценз.



Дайджест журнала КОМПЬЮТЕР*** за 2020 год

Внимание. Если вы создаете произведение, рассчитанное на детей и подростков, не забудьте отключить опции "Использование ненормативной лексики", "Насилие" и "Секс".

Далее вы должны определить место действия. В русской версии вы можете выбрать любой из 3000 российских городов или указать любую из 84 стран ближнего и дальнего зарубежья. Если вы не хотите конкретизировать место действия, можете воспользоваться опцией "Обобщенный русский город".

В следующем диалоговом окне вам предстоит создать героев своего произведения. Их количество не может превышать 256 персонажей.

Совет. Не давайте своим персонажам имена по умолчанию.

Работа по созданию персонажей весьма интересная и творческая. Во-первых, вы должны определить статус: главный герой, герой второго плана, эпизодический, положительный или отрицательный. Вы можете определить не только пол, возраст и внешность персонажа, но и

его психотип, выбрав один из 65536 стандартных, либо создав новый. Однако для этого неплохо иметь диплом психолога. Ваш персонаж может быть животным, а в случае фэнтези — любым из более чем 1000 типов существ.

В следующем диалоговом окне вы должны определить сюжет. Как известно, в мировой литературе существует всего 7 оригинальных сюжетов. Авторы программы пошли по тому же пути, предлагая в качестве основного сюжета выбрать один из семи и определить статистический вес остальных сюжетов. Вам также предстоит определить главную и побочные сюжетные линии, указать события, участников и последствия. Внутреннюю часть работы вы можете оставить на усмотрение компьютера, однако это не значит, что она вас устроит.

Далее наступает черед проверки на непротиворечивость. Так, если вы дали своим персонажам китайские имена, а действие происходит в Моршанске, программа затребует от вас обоснования этого факта.

Отслеживается также всякое упоминание погибших и умерших персонажей после факта их гибели, хронология и родственные связи героев. Персонаж не может появиться в одно и то же время в разных местах, а внук не может жениться на своей бабушке.

Допустим, если вы указали, что у вашего героя "Ягуар" последней модели, но компьютер — африканской сборки, пейджер монгольский, мобильник фиджийский, а костюм хоть и от Кардена, но пошит на Малой Арнаутской в Одессе, программа упомянет, что автомобиль либо угнан из Европы, либо собран на Таганрогском велосипедном заводе.

После этого, в принципе, вы можете дать команду на генерацию текста произведения. Однако не стоит торопиться. Рассмотрим еще несколько интересных опций.

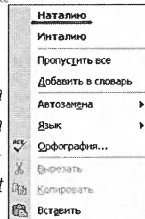
Как уже отмечалось, в базе данных содержатся сведения о всех русских писателях. Вы можете дать команду программе писать произведение в стиле Достоевского или Жванецкого, Чехова или братьев Стругацких. Еще более интересна опция соединения стилей разных писателей (не более 5 писателей сразу, од-

нако разработчики обещают расширить возможности этой опции).

Внимание. При выборе писателей упорядочьте их по стилистической характеристике и выбирайте писателей, укладывающихся в разброс не более 200 пунктов. Слишком большой разброс в стилях может привести к непредсказуемым последствиям.

Я хочу в Анталию

Образец творчества далекого предка MegaSoft eWriter — программы Microsoft Word



Напоследок укажите, от первого или от третьего (по умолчанию) лица будут излагаться события. Еще раз сохраните проект и запускайте генерацию текста. Процедура эта очень долгая даже для SuperMind-XX. Вы смело можете ложиться спать: работа займет от нескольких часов до нескольких дней.

Разработчики держат в секрете алгоритм генерации текста. Автор этих строк 10 раз запуск генера-

цию текста с одними и теми же установками и получил 10 полностью идентичных текстов, совпадающих вплоть до запятой. Текст хранится в специальном формате, разработанном специалистами MegaSoft. Рассказ объемом 5 машинописных листов с двумя персонажами и тремя событиями весит больше 3 Гбайт. Если вы задумали большой роман — запаситесь памятью. Однако этот формат оправдывает себя. Если вы вносите изменения в проект — добавляете или убираете персонаж, изменяете ход событий или психотип героя — все ваши изменения сразу динамически обновляют текст. Причем процедура эта происходит практически мгновенно, что экономит вам массу времени.

Вы можете и сами вносить изменения в текст или давать команду компьютеру переписать тот или иной эпизод или абзац, добавив экспрессии, описательности и т. д., но обычно этого не требуется: вы сразу можете отсылать произведение в редакцию.

Как нам стало известно, из российских писателей eWriter приобрели Марина Александрова и Абдулла Чингизов, а издательство "Букмэн"

Mosaic Killer...



Убийца браузера Mosaic — именно так расшифровывается название проекта Mozilla, ранее известного как Netscape Navigator (Communicator). Тем, кто забыл, напоминаю, что Mosaic — это та основа, на которой было построено громоздкое и уродливое здание Microsoft Internet Explorer. И если в строке адреса в любой версии браузеров Netscape-Mozilla набрать `about: mozilla` (без кавычек), то вашим очам предстанут строки из фирменного нетскейповского "Евангелия", содержащие пожелание "долгой жизни" некоторым программам (не трудно догадаться, каким и чьим).

Долгое время этот проект считался исключительно тестовым и экспериментальным, но вот, с выходом версии 1.0, он получил как бы аттестат зрелости. Не за горами выход версии 1.1, которая еще больше упрочит место проекта на рынке браузеров.

Итак, Mozilla ver 1.0.

Пакет состоит из многочисленных компонентов, причем каждый из них имеет свою группу авторов. В основе всего лежит кроссплатформенный движок-ядро под кодовым названием Gecko, который можно использовать при разработке других программ (чем и воспользо-

вались разработчики браузера Galeon, который, кстати "весит" намного меньше чем Mozilla).

В состав пакета входят почти все программы, необходимые для работы в Интернете: собственно браузер, почтовый и новостной клиенты, WYSIWYG-HTML-редактор (в русской версии названный "компоновщик"), адресная книга и несколько мелких, но полезных утилит, среди которых есть отладочные инструменты для веб-дизайнеров.

Теперь о каждой из этих программ чуть подробнее.



разорвало все авторские контракты и полностью перешло на автоматическое создание книг.

ButtonPress Programmer

Является ли работа программиста творческой? Этот вопрос не стоял разве что перед первыми пользователями Эниака. Программисты всячески облегчали свой сложный, рутинный и утомительный труд, придумав ассемблер, языки высокого уровня, подпрограммы, модули и объекты, визуальное программирование. Programmer 1.0 компании ButtonPress — апофеоз идеи облегчения труда программиста: это программа, которая пишет другие программы!

Началось все с невинного транслятора Poliglot, который переводил исходники с Паскаля на Си и наоборот. Постепенно транслятор стал поддерживать до 50 языков и диалектов. Именно Poliglot и послужил толчком к написанию Programmer'a. Ведь если аналог оператора PRINT

есть в любом языке, то некоторые команды и процедуры на другие языки переводятся не совсем тривиальным способом.

Задумавшись над тем, как часто программисту приходится изобретать велосипед. Программисты прекрасно понимают, о чем я веду речь. Задача, которую решает программист N, уже наверняка решена программистом M. Только отсутствие связи между ними, языковые и юридические проблемы приводят к тому, что каждый идет своим путем.

Кто из программистов не писал программу по зарплате? Или не создавал базу данных по клиентам и документам? Вавилонское столпотворение сред разработки, особенностей работы предприятий и приводит к тому, что программа, написанная одним программистом, устраивает только одного заказчика.

Programmer позволяет преодолеть это препятствие легко и элегантно. Вы выбираете стандартную задачу (список впечатляет!), отве-

чаете на появляющиеся вопросы (кое-что можно оставить на усмотрение компьютера) и запускаете генерацию кода. Через 5 минут программа готова.

На вполне естественном языке (английском) вы описываете действие своей функции, выбираете один из 15 языков программирования и получаете код.

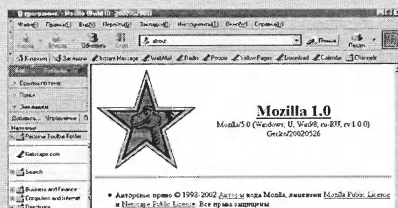
Все дело в том, что Programmer не выполняет работу программиста, а только облегчает ее. Задача должна быть четко поставлена и не допускать других решений. Если современный программист теперь может не знать ни одного языка программирования (а их более 1000, и у знающего 10 шансов получить работу не больше, чем у знающего только один), то думать за него компьютер не в состоянии. Программист должен разбить задачу на подзадачи, выбрать алгоритм.

Итак, программа компании ButtonPress отнюдь не угрожает профессии программиста, а, наоборот,

Браузер

Браузер имеет приятный дизайн и на нормальном компьютере достаточно быстро загружается. На расположенной сбоку панели Sidebar можно разместить много полезной информации: прогнозы погоды, курсы валют, поисковые формы и многое другое. За счет так называемой конвейерной обработки данных (данные грузятся единым потоком, а не по одному потоку на каждый файл, как в других браузерах) ощутимо повышается скорость загрузки страниц. При помощи специальной опции, вызываемой через меню "Настройки — Дополнительно — Сценарии и окна", вы сможете запретить показ рекламы во всплывающих окнах и в строке статуса. Технология PSM (Personal Security Manager) позволяет управлять паролями и сертификатами прямо из окна настроек. Кроме того Mozilla поддерживает стандарт HTML 4.01, XHTML (Extensible Hypertext Markup Language, расширяемый язык разметки гипертекста — нечто среднее

между HTML и XML), а также таблицы каскадных стилей CSS всех трех стандартов, язык MathML (математический HTML) и многие другие современные стандарты и технологии. Если вас чем-то не устраивает дизайн браузера, его можно легко изменить, используя специальные темы.



Возможно использование различных фильтров для сообщений, а для большей безопасности вы можете установить дополнительную программку Enigmail, которая шифрует сообщения с добавлением электронной подписи.

Адресная книга

Хотя эта программа предназначена для работы совместно с предыдущей, ее можно запустить отдельно и использовать в качестве органайзера.

Композитор

Эту программу вряд ли можно назвать предназначенной для профессиональных веб-дизайнеров, но для начинающих пользователей она будет в самый раз. К тому же при отправке с помощью почтового клиента из этого пакета писем и сообщений в NEWS-группы вы будете пользоваться именно "компоновщиком". В нем вы сможете создать страницу, содержащую текст в различном шрифтовом и цветовом

Почта и новости

Для нормальной работы вам потребуется достаточно мощный компьютер, но с ним вы получите уникальную программу, стоящую на одном уровне с The Bat! и Eudora, а по некоторым характеристикам даже выше их. Программа поддерживает протоколы SMTP, POP3 и IMAP, работает через SSL (защита портов).



облегчает ее, делает более творческой, расширяет доступ к ней большому количеству людей. Подобно тому, как изобретение фотографии не уничтожило живопись, так и Programmer не в состоянии заменить человека. Хотя, возможно, это дело не такого далекого будущего.

DuraLex и Parliament

Эти две юридические программы разработаны российской компанией Blintus Technology Inc благодаря развитию технологии распознавания естественного языка.

Название первой говорит само за себя: "Закон суров". Как известно, содержание органов юстиции, адвокатур, органов дознания и исполнительных учреждений очень дорого обходиться государству. Успехи медицины позволяют надеяться, что исправительные учреждения в скором времени превратятся в лечебные, а органы дознания — в службу психологической помощи, но пока ситуация остается очень сложной.

оформлении, вставить в текст картинки, создать текстовые или графические ссылки, а при необходимости разместить все это в таблице. Разумеется, здесь также можно изменить цвет фона или поместить фоновую картинку. При работе доступны три основных режима: обычный, HTML-код и режим предварительного просмотра. В первом режиме доступна функция "Показать все теги", которая, помимо прочего, поможет быстрее понять суть HTML и научиться писать страницы вручную, без применения специальных программ.

Если при работе вы столкнулись с каким-то глюком или ошибкой, посетите официальный сайт разработчиков (англоязычный <http://www.mozilla.org> или русскоязычный <http://www.mozilla.ru>). Скорее всего, там уже лежит исправленная версия программы. Все найденные ошибки исправляются в кратчайшие сроки, и специально для этого предусмотрен сервис под названием BugZilla.

Георгий Баранов

Содержание судей и контроль за их профессионализмом — очень болезненная тема. Людям свойственно заблуждаться, они могут намеренно защищать интересы одной из сторон, их можно купить, можно запугать или запутать в сложных и противоречивых показаниях. Программа DuraLex — решение этой проблемы.

Компьютер помнит все обстоятельства дела, знает все действующие законы и подзаконные акты, он абсолютно объективен и беспристрастен. Уходят в прошлое процессы, тянувшиеся годами. Любое дело решается в считанные минуты. Приговоры справедливы, и, как правило, не обжалуются. Большая часть судов Москвы, Санкт-Петербурга, Ростова-на-Дону и Новосибирска уже ликвидированы, а юридические институты перепрофилированы: в этих городах внедрена программа DuraLex.

Программа Parliament, моделирующая процесс прохождения законопроектов через Думу, заказана Администрацией президента. В про-

грамму занесены данные о всех депутатах: не только их политические пристрастия, но и личные данные, вплоть до содержания сахара в моче. В течение полугода перед вынесением какого-либо законопроекта на обсуждение в Думу этот процесс моделируется на компьютере. Как утверждает наш осведомленный источник в Кремле, расхождение расчетных данных с реальными составляет не более 5%. По понятным причинам подробности работы с этой программой (как и с DuraLex) засекречены.

Сейчас в Кремле усиленно обсуждается проект замены Думы компьютерной программой Parliament. Однако расчет с ее помощью показывает, что данный законопроект не встретит поддержки у депутатов.

P.S. Как нам стало известно в процессе подготовки журнала к печати, Белый Дом обратился к руководству Blintus Technology Inc с просьбой адаптировать программу к слушаниям в Конгрессе США.

Григорий Власов

Маленький японский кроссворд

		1	1	1		1	1	1	1	1	1	1	1	4	3	3			
		2	1	1	1	1	7	8	7	7	8	7	7	1	1	1	2		
		2	3	3	4	7	1	1	1	1	1	1	1	1	1	1	2		
		2	2																
		1	1	4															
		1	1	4															
			1	5															
			1	5															
			1	6	1														
			1	7	1														
			1	9	1														
			1	9	1														
			1	7	1														
			1	6	1														
				5	1														
				5	1														
			4	1	1														
			4	1	1														
			2	2															

Логотип какой программы здесь зашифрован?

Кроссворд прислал Александр Полуни

12-я международная выставка

INWECOM

**информационные технологии
и вычислительная техника**



INWECOM 2002

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ:
ГОНКА ПРОДОЛЖАЕТСЯ

Санкт-Петербург
26 – 30 ноября 2002

Санкт-Петербург

Выставочный комплекс в Гавани (В.О., Большой пр., 103)

Общая часть

Тематические разделы:

- Компьютеры
- Мониторы
- Комплектующие и компьютерные аксессуары
- Периферийные устройства
- Операционные системы и инструментальные средства
- Телекоммуникационные технологии и средства
- Интернет-технологии, провайдеры, услуги
- Мультимедиа технологии
- Цифровые технологии
- Техника и оборудование для офиса
- Учебно-методическая литература
- Сервис и услуги

Специализированная часть

Тематические разделы:

- Автоматизированные и информационные системы предприятий
- Банковские системы и оборудование
- Сетевое оборудование
- Программные средства и системы для бизнеса
- САПР и геоинформационные системы
- Электронная коммерция
- Полиграфическое оборудование
- Презентационное оборудование и проекционная техника

Отдельные выставочные блоки:

- Региональная информатика
- Промышленная автоматизация
- Современные средства обучения (выставка "EDUCOM")
- Информационная безопасность



**Одновременно с выставкой проводится 3-й бизнес-форум
"Информационные технологии в развитии Северо-Западного региона России"**

Организатор:



РЕСЭК
ВЫСТАВОЧНОЕ
ОБЪЕДИНЕНИЕ

Дополнительная информация:

Тел./факс: (812) 235-7391, 320-9688

E-mail: inwecom@restec.ru

www.restec.ru/it-week

При поддержке:

Ассоциации экономического взаимодействия субъектов Северо-Запада РФ
Администрации Санкт-Петербурга

Генеральный информационный спонсор:



**ОТКРЫТЫЕ
СИСТЕМЫ**
Open Systems Publications

Информационная поддержка:

PCWEEK
COMPUTER WEEK

**КОМПЬЮТЕР
ИНФО**
СВЕДЕНИЯ О ТЕХНИКЕ И ПОДСОЗНАНИЕ