

радиомир

9 • 2005

Ваш компьютер

- JAZZ-овый охват
- Темная лошадка — BlackICE
- Автоматизация открытия документов в браузере Opera
- Mr.Gluk Reset Service 6.0R





КОМПЬЮТЕРЫ **СИТИПРИНТ** ДЛЯ ДОМА И ОФИСА

**Ваши сотрудники способны на большее?
Позвольте им доказать это**

Компьютер от компании "СитиПринт" на базе процессора **Intel® Pentium® 4** с технологией **HT** поможет Вашим сотрудникам выполнять больше работы за меньшее время.



 **СитиПринт**
КОМПЬЮТЕРЫ И ОРГТЕХНИКА

220006, г.Минск, ул.Полевая 28-7А.
Тел./факс (017)2850134, 2218728.
www.citiprint.by

Учредитель и издатель:
ООО "Радиомир Пресс"

Свидетельство о регистрации
ПИ №77-9841 от 17.09.2001

Главный редактор
Ольга СТРЕЖАНКОВА

Зам. гл. редактора
Иван БЕЛЬСКИЙ

Редколлегия:
Сергей ДРОЗДОВСКИЙ,
Владимир КУЦЕНКО,
Елена ЛЕВИТМАН,
Геннадий ПЕЧЕНЬ,
Геннадий ШУЛЬГИН

Контактные телефоны:
в Москве (095) 105-99-89,
в Минске (017) 249-41-47

Компьютерная верстка —
Янина БЕЛЬСКАЯ

Техническая графика —
Наталья БЕЛЬСКАЯ,
Мария КАЛАБУХОВА

Оформление обложки —
Наталья БЕЛЬСКАЯ,
Надежда БОГОМОЛОВА

Адреса для писем:
119454, РФ, г.Москва, а/я 37,
факс (095) 432-62-04;
220095, РБ, г.Минск-95, а/я 199

E-mail: rm@radio-mir.com
WWW: <http://radio-mir.com>

Наши платежные реквизиты:
получатель: ООО "Радиомир Пресс",
ИНН 7729404741, КПП 772901001,
р/с 40702810900010000084
в ООО КБ "Агропромкредит"
ф-л Центральный, г.Москва,
корр. счет 30101810500000000109
БИК 044525109
Адрес банка: 125315, г.Москва,
Ленинградский пр., дом 76, корп. 4

Материалы для публикации принимаются
в рукописном, печатном и электронном
вариантах

Требования к графическим материалам рек-
ламного характера в электронном виде:
CorelDRAW до 10.0, все шрифты в кривых;
bitmaps 300 dpi; TIFF 300 dpi; CMYK.
Приложить печатную копию.

За достоверность рекламной и другой публи-
куемой информации несут ответственность
рекламодатели и авторы. Мнение редакции
не всегда совпадает с мнениями авторов

Адрес редакции: 119454, РФ, г.Москва,
ул.Колотоянца, 6 — 233, тел. (095) 105-99-89

Подписано к печати 22.07.2005 г.
Формат 60 x 84 1/8.
Печать офсетная, 6 печ. л.
Цена свободная.

© ООО "Радиомир Пресс". Воспроизведение матери-
алов журнала в любом виде без письменного разре-
шения редакции запрещено. При цитировании ссылка
на "Радиомир. Ваш компьютер" обязательна.

Отпечатано в типографии
ООО "Красногорская типография".
Заказ № 2054.

радиомир Ваш компьютер

ЕЖЕМЕСЯЧНЫЙ МАССОВЫЙ ЖУРНАЛ

№9/сентябрь/2005

ЧИТАЙТЕ В НОМЕРЕ

ВОКРУГ ПК

МУЛЬТИМЕДИА

В.КУЦ. JAZZ-овый охват 2

НЕ ТОЛЬКО НОВИЧКУ

А.ГРИНЧУК. Windows XP для пользователя
Установка и удаление программ 4
Е.ЗАЙЦЕВА. Эффективная работа с Microsoft Word XP
Создание и управление диаграммами в документе 8
В.КУЦ. ОРФО 2004 проверяет правописание 11
Б.КИСЕЛЕВ. MATLAB. Пакет Simulink 14
С.РЮМИК. Интернет-калейдоскоп, freeware #22 18

КОММУНИКАЦИИ

Е.ЛЕБЕДЕВ. Темная лошадка — BlackICE 20
И.РОЩИН. Автоматизация открытия документов
в браузере Opera с помощью Perl-скрипта 22

ДАЙДЖЕСТ 24

ПРОГРАММЫ И АЛГОРИТМЫ

УРОКИ ПРОГРАММИРОВАНИЯ

О.ВАЛЬПА. Borland C++ Builder 6 для начинающих 27
А.КОРБИТ, Т.КРИВОНОСОВА. Динамические структуры двнных:
линейные списки и рекурсия 28
А.КОРБИТ, Т.КРИВОНОСОВА. Поиск и сортировки:
сортировка вставками (включениями) 32

ДИАЛОГ ПРОГРАММИСТОВ

П.СОКОЛЕНКО. Арифметика длинных чисел 33
И.РОЩИН. Программа для поиска и замены 36

РЕЦЕПТЫ

РКАРПАЧ. В DOS с нуля — что-то, где-то, когда-то слышал 38

МИР 8 БИТ

А.ПЕХИМЕНКО. Обзор электронной прессы о Spectrum 46
Д.БЫСТРОВ (Alone Coder). Mr.Gluk Reset Service 6.0R
Прошивка ПЗУ для ZX Spectrum 42

ИГРОТЕКА

А.ВЕНДИЛОВСКИЙ. Star Wars. Republic Commando 45

ДОСКА ОБЪЯВЛЕНИЙ

Куплю, продам, обменяю 47

Полные листинги некоторых программ расположены по адресу
<http://radio-mir.com>



JAZZ-ОВЫЙ ОХВАТ

Трансконтинентальная корпорация Jazz Hipster Group (со штаб-квартирой на Тайване) — совсем не новичок на рынке акустических систем. С 1981 г. ее продукция радует любителей хорошим качеством звучания и оригинальным дизайном. Диапазон интересов компании довольно широк и простирается от недорогих классических двухполосных колонок и вплоть до полностью укомплектованных цифровых домашних кинотеатров. Но все-таки наибольшее внимание фирма уделяет компьютерным акустическим системам (АС) среднего и верхнего ценового диапазона. И среди них, в связи с резким ростом популярности DVD, наибольший интерес вызывают многоканальные акустические системы формата 5.1, которые являются "необходимыми и вполне достаточными" для обеспечения полноценного объемного звука.

Аудиоформат 5.1 (первая цифра — количество колонок, вторая — количество сабвуферов) разрабатывался, в первую очередь, для просмотра фильмов на DVD. Кроме того, многоканальная акустика дает очень неплохие результаты в компьютерных играх, да и при прослушивании музыки, особенно современной. В состав этих комплектов входят две фронтальные колонки, две тыловые, одна центральная и сабвуфер.

В любой акустической системе главную роль играют фронтальные колонки, поскольку именно они отвечают за тональный баланс и воспроизведение всех музыкальных нюансов. Чаще всего фронтальные колонки строятся по двухполосной схеме, включающей два динамика — широкополосный и высокочастотный. Центральный излучатель не менее важен, поскольку на него ложится ответственность за "озвучивание" речевых диалогов героев фильмов и т.п. В хорошей акустике в центральном спикере обычно объединены два широкополосных динамика, иногда дополненные еще и высокочастотным. Требования к тыловым колонкам менее строгие, чем к фронтальным, ведь они играют вспомогательную роль — на них ложится "подзвучка" при воспроизведении музыкальных композиций и создание трехмерных спецэффектов в фильмах и играх.

За качественное воспроизведение самых низких частот звукового диапазона "отвечает" сабвуфер. Так как низ-

кочастотные звуки (ниже 50 Гц) практически не локализируются слухом, т.е. человек не в состоянии определить, откуда идет звук, использование для воспроизведения басов всех каналов одной-единственной колонки вполне оправдано. Тем более, что у низкой локализации басов есть еще и дополнительный плюс — при размещении колонок в комнате, сабвуфер можно установить практически произвольно, в самом удобном месте.

Но и это еще не все. Во многих качественных системах 5.1 можно встретить звуковые процессоры, которые декодируют многоканальный звук в соответствии с определенными форматами — для акустики 5.1 это Dolby Digital, DTS и Dolby Prologic. Именно наличие этого компонента и позволяет использовать компьютерную акустику формата 5.1 для создания полноценного домашнего кинотеатра.

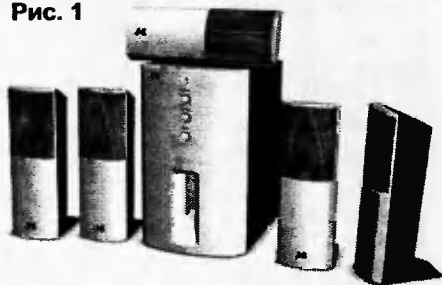
Несмотря на то что в настоящее время широко рекламируются "продвинутые" 6.1- и 7.1-канальные конфигурации — с шестью (классический набор 5.1 плюс центральный тыловой громкоговоритель) и семью (дополнительная пара для боковых каналов) сателлитами, качество эффекта окружающего звучания зависит отнюдь не от наличия или отсутствия дополнительных каналов, а от общего уровня системы, а также от корректности ее размещения в помещении прослушивания и тщательности настройки каждого из каналов. Поэтому, лучше "хорошие, пусть даже 5.1", чем "плохие, но 7.1". Именно в соответствии с этим лозунгом и строит свою производственную программу компания Jazz Hipster, наибольшее внимание уделяя разработке и производству акустических систем среднего ценового диапазона, простирающегося от 100 до 300 USD. С некоторыми представителями современной пинейки 5.1-канальной акустики Jazz Hipster мы сегодня и познакомимся.

Jazz Hipster J-9959

К системам начального уровня (начальный уровень Jazz Hipster, разумеется, это совсем не то, что начальный уровень, к примеру, Genius) можно отнести набор J-9959 (рис.1), который пришел на смену заслуженному ветерану J-9906. Он, так же как и предшественник, выполнен в строгом класси-

ческом стиле. Сочетание черного и серебристого цветов в оформлении как сателлитов, так и сабвуфера достаточно элегантно, чтобы вписаться в любой интерьер. Форма магнитоэкранированных сателлитов, оснащенных двухдюймовыми "широкополосниками", также не выходит за пределы классических канонов — прямоугольный параллелепипед, причем относительно крупных размеров. Корпус сабвуфера (кстати, достаточно компактный, для того чтобы не было проблем

Рис. 1



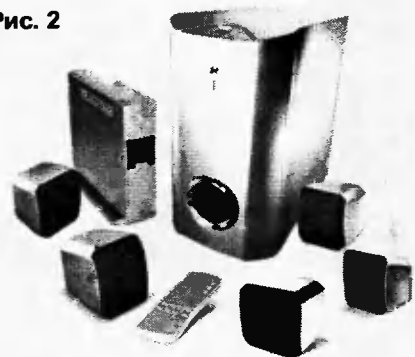
с его размещением) сделан из ДСП, он имеет отверстие фазоинвертора, обеспечивающего дополнительный поток низких частот.

Заявленная выходная мощность системы составляет всего 55 Вт (естественно, измеренных по методике RMS, что означает — "честных" ватт), которых хватит лишь для озвучивания небольших помещений (не более 20 м²). Встроенный в сабвуфер усилитель, имеющий минимальный набор регулировок (регулятор громкости, кнопки включения и подъема басов), для получения полноценного многоканального звука требует использования внешнего декодера, что, наряду с относительно небольшой мощностью системы, является одним из самых существенных недостатков АС J-9959.

Jazz Hipster J-9938

Указанного недостатка лишен комплект Jazz Hipster J-9938 (рис.2), состоящий из шести колонок и цифрового декодера, который поддерживает различные эффекты и режимы объемного

Рис. 2



звучания, включая AAC, Dolby Digital, Dolby Prologic II и DTS, а также имеет широкие возможности настройки некоторых специфических параметров.

Комплект состоит из пяти маленьких однополосных сателлитов, сабвуфера, отдельного блока декодера-усилителя и пульта дистанционного управления, который позволяет пользователю менять не только уровень громкости и выбирать источник звука, но и управлять эффектами объемного звука.

Декодер выполнен в виде маленькой башни с расположенной в верхней части лицевой стороны панелью управления. Основные органы управления системой сосредоточены в верхней части блока, для отображения текущих установок и другой полезной информации используется специальный светодиодный дисплей. Декодер позволяет подключить до трех внешних источников звука. Таким образом, пользователь может одновременно подключить к Jazz Hipster J-9938 компьютер и DVD-плеер.

Сабвуфер системы выполнен в деревянном корпусе с пластиковой накладкой на лицевой стороне. А вот сателлиты, как фронтальные, так и тыловые — целиком пластиковые и оснащены специальным магнитным экраном, устраняющим негативное влияние электромагнитного излучения на стоящие рядом приборы.

Точно так же как и у J-9959, выходная мощность системы не превышает скромных по нынешним временам 45 Вт (20 Вт приходится на сабвуфер и по 5 Вт — на сателлиты), что, конечно же, совсем немного, однако, учитывая низкую цену вкупе с широкими возможностями цифрового декодера, J-9938 представляется довольно удачным решением начального уровня.

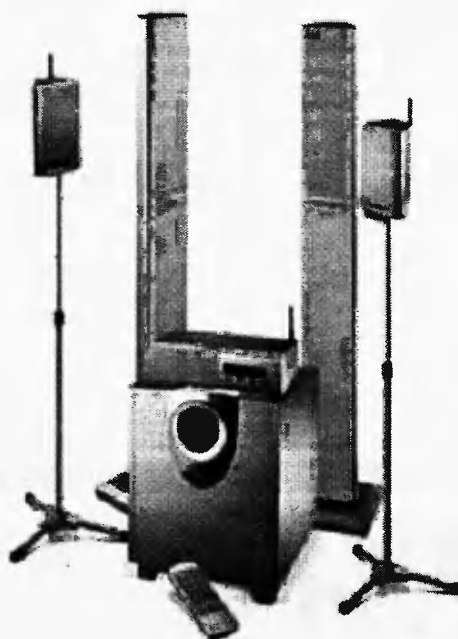
Jazz Hipster J-9940

На другом полюсе все того же среднего ценового диапазона располагается одна из наиболее качественных и дорогих моделей — J-9940. Эта АС выпускается уже не первый год, однако благодаря прекрасному соотношению цены и качества она и до сих пор успешно конкурирует с большинством более современных моделей аналогичного класса. J-9940 отличают довольно внушительные размеры активного 100-ваттного сабвуфера (330 x 410 x 330 мм) и фронтальных колонок (напольной конструкции, 130 x 1100 x 87 мм), выполненных целиком из дерева. Хотя центральный и задние спикеры гораздо скромнее фронтальных, тем не менее, мощность все они “вы-

Рис. 3



Рис. 4



дают” одинаковую — по 20 Вт каждый. Суммарная выходная мощность всей системы — 200 Вт (RMS). Естественно, что все динамики имеют магнитное экранирование.

Кроме привычной активной акустической системы JS-9940, в модельной линейке фирмы существует еще и 2 ее продвинутых модификации: JS-9940B (рис.3) и JS-9940W (рис.4).

Первая из них имеет 5.1-канальный декодер Dolby Digital/DTS и Dolby Pro Logic, что делает ее просто идеальной основой для создания полноценного домашнего кинотеатра начального уровня. Тыловые сателлиты второй системы — беспроводные, получающие сигнал от усилите-

ля мощности по радиоканалу. Благодаря такому решению не приходится путаться с длинными проводами, протягивая их через всю комнату к месту расположения динамиков, что, безусловно, в высшей степени удобно. В системах J-9940W и JS-9940B блок управляющей электроники совмещен с центральной колонкой (330 x 95 x 200 мм), на передней панели которой располагаются органы управления и информационный дисплей.

Безусловно, любая АС из линейки J-9940 способна обеспечить самое высокое качество звука не только при просмотре DVD и в компьютерных играх, но и для прослушивания музыки, в том числе и классической — наиболее сложной для любой акустики.

Кроме этих комплектов, к данной линейке можно отнести еще и акустическую систему J-9941 (рис.5) с фронтальной акустикой уменьшенных габаритов и ее родную “сестру” J-9940B, оснащенную цифровым 5.1-канальным декодером.

Акустическая система J-9940 представляет собой классический шести-канальный набор, состоящий из центральной, двух фронтальных колонок-башен и двух компактных тыловых, 100-ваттного сабвуфера и пульта ДУ. Все колонки выполнены по двухполосной схеме и имеют магнитное экранирование.

Честно говоря, покупая какую-либо из акустических систем, произведенных компанией Jazz Hipster, вы вряд ли сэкономите средства, потому что они стоят существенно выше дешевых китайских подделок. Но и звучание имеют несравнимо более качественное — так что в данном случае более высокая цена вполне оправдана. Что же касается хорошо известных брендов, то их продукция, при сравнимом качестве, находится на одном с Jazz-ом ценовом уровне, а иногда и превышает его.

Рис. 5



А.ГРИНЧУК,
E-mail: gav@nihe.niks.by

Windows XP для пользователя

Установка и удаление программ

Не спешите отчаиваться. Даже если вас съели дикари, у вас все равно есть два выхода.

Джеймс Кук

В прошлой статье мы были вынуждены обратиться к программе, не имеющей непосредственного отношения к Windows XP. Повторим еще раз общий вывод — программы, встроенные в Windows, всегда под рукой и проще в использовании, зато альтернативные варианты предлагают больше возможностей. У нас остается два выхода — либо мучиться со встроенными программами, либо заняться установкой того, что нам нужно. К счастью, в Windows XP есть одна хорошая особенность — в случае неудачной установки можно вернуться к предыдущему, нормально работающему состоянию системы. Шансы на успех возрастают, если перед экспериментами с установкой вы не поленитесь и вручную создадите точку восстановления. Строго говоря, эта функция появилась еще в Windows Millennium, но тогда она работала не лучшим образом — спустя несколько недель резервными файлами замусоривался весь винчестер. Сейчас все работает намного четче, стабильнее и аккуратнее.

В результате перед нами возникают следующие задачи:

- подготовить точку восстановления перед установкой программы;
- установить программу;
- при необходимости произвести настройку системы для работы с программой;
- удалить программу, если она ока-

залась ненужной.

В первую очередь настроим параметры восстановления системы: **Пуск/Панель управления/Система**. Переключаемся на вкладку **Восстановление системы** (рис.1) и там, выбирая диск и нажимая кнопку **Параметры**, указываем размер дискового пространства. Желательно не скупиться: больше места — надежнее пройдет восстановление. Затем переходим к созданию точки восстановления: **Пуск/Справка и поддержка/Отмена изменений с помощью Восстановления системы** (рис.2). Вот здесь вам предлагается либо восстановить систему, либо перейти к созданию новой точки восстановления (рис.3). В последнем случае следует переключиться в режим создания, нажать кнопку **Далее** и дать название точке восстановления (рис.4). Система автоматически запомнит время и дату. Если же нужно восстановить прежнее состояние — переключиться в режим восстановления (рис.3) и после кнопки **Далее** выбрать один из возможных вариантов (рис.5).

Что касается непосредственно установки программ, то здесь можно говорить только о статистических за-

Рис. 1

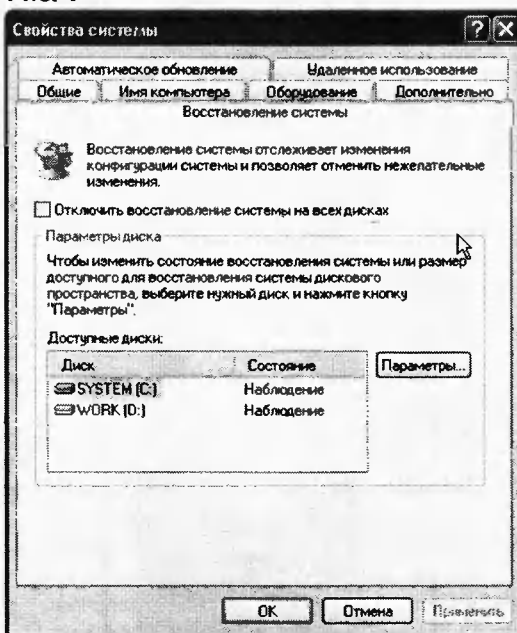
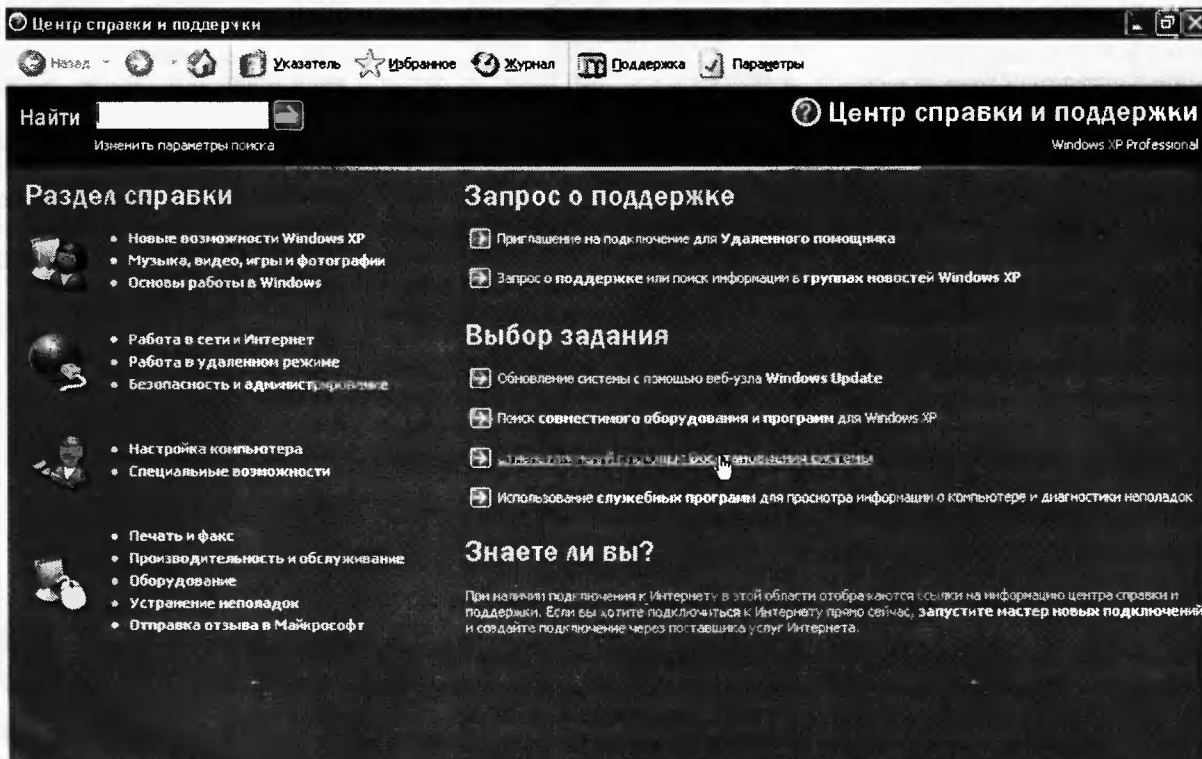


Рис. 2



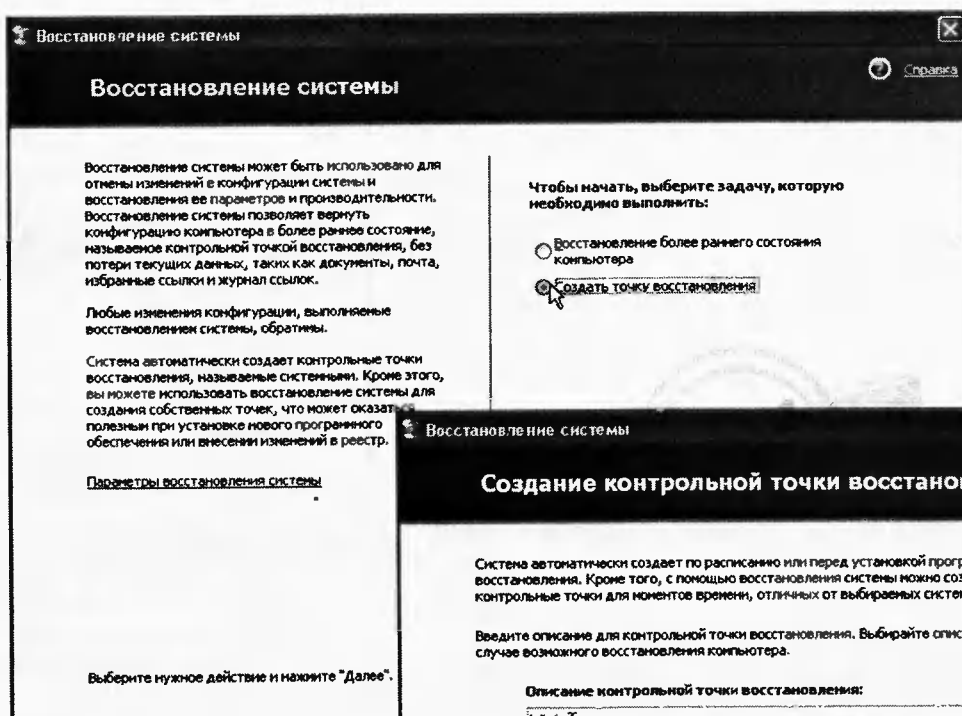


Рис. 3

здаются "с запасом", и не каждому пользователю нужны все возможности. Тогда для экономии дискового пространства можно либо потребовать типичную установку, либо самостоятельно выбрать необходимые компоненты (рис.9). Несколько нажатий на кнопку **Далее** — и, после некоторого ожидания у индикатора хода инсталляции (рис.10),

Рис. 4

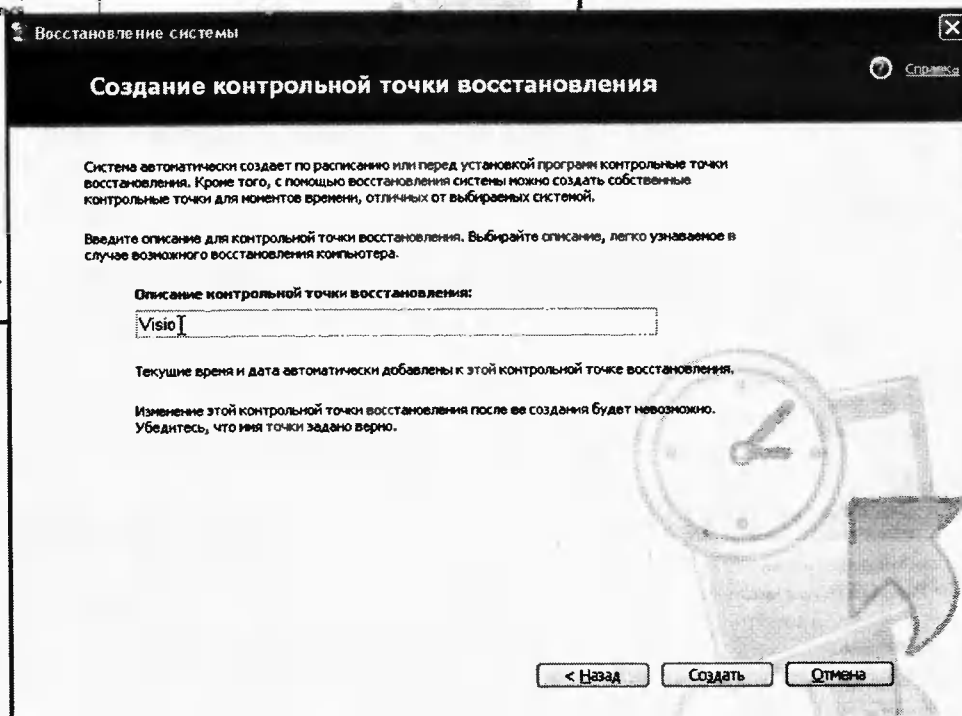
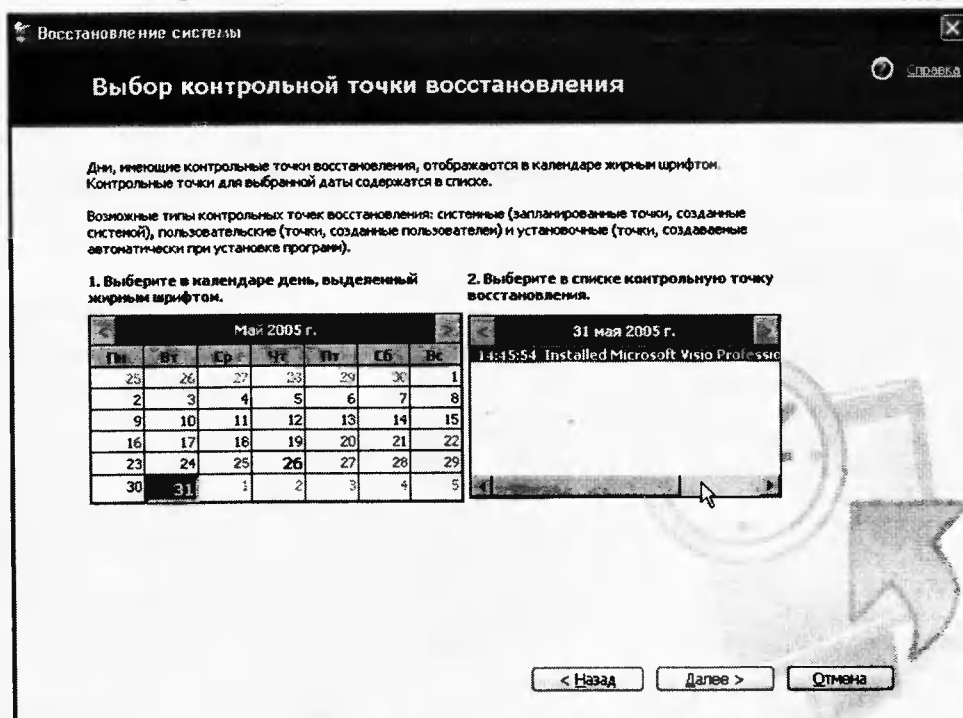


Рис. 5



кономерностях — слишком уж много написано программ, у каждой свои особенности и требования. Стандарты установки выработаны только самые общие, их мы рассмотрим на примере программы Visio. Часто программы поставляются на компакт-дисках, здесь достаточно просто нажать на кнопку начала установки. Иногда приходится заходить в папку с программой и находить установочный файл. Его типичные названия — Install.exe или Setup.exe (рис.6). Подробности сообщаются в файлах с призывным названием Readme (от read me — прочитай меня). После двойного щелчка по установочному файлу и начинается собственно установка (инсталляция) программы.

Прежде всего вам предложат ознакомиться с лицензионным соглашением и принять его (рис.7), без согласия инсталляция будет прекращена. Часто этот этап сопровождается требованием ввести регистрационный номер (ключ) программы и окошком для его ввода. На следующем шаге обычно указывается папка для установки программы (рис.8). По умолчанию она создается в папке C:\Program Files, ее легко изменить на любую другую, выбрав **Browse (Обзор)**. Но все-таки программные файлы удобнее хранить именно в C:\Program Files. Большие программы обычно со-

Рис. 6

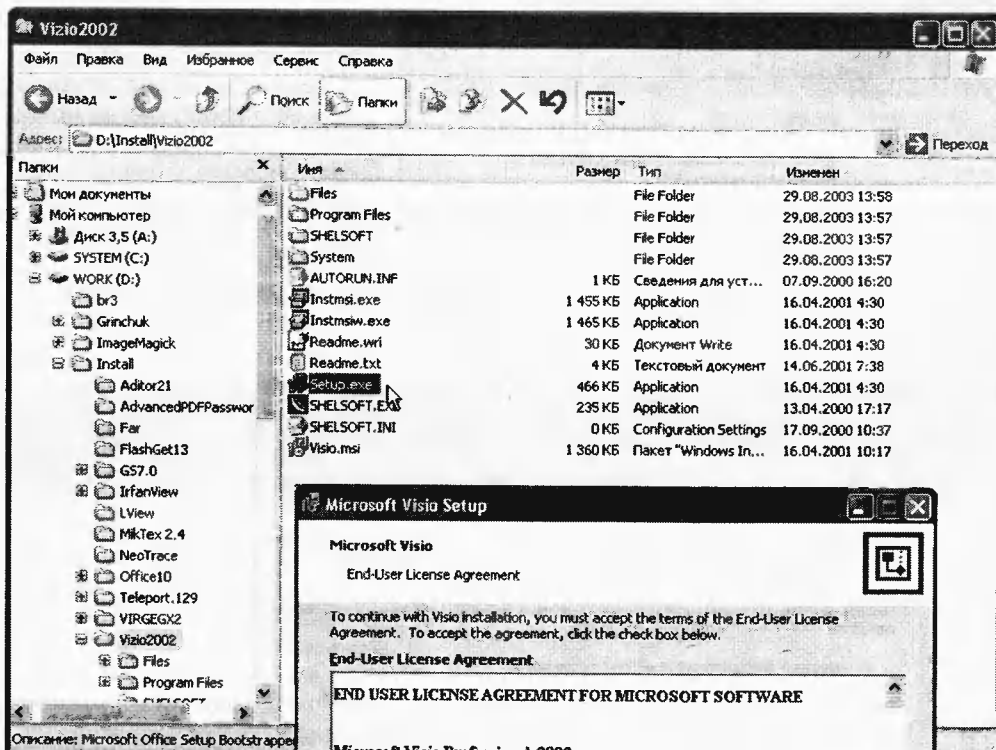
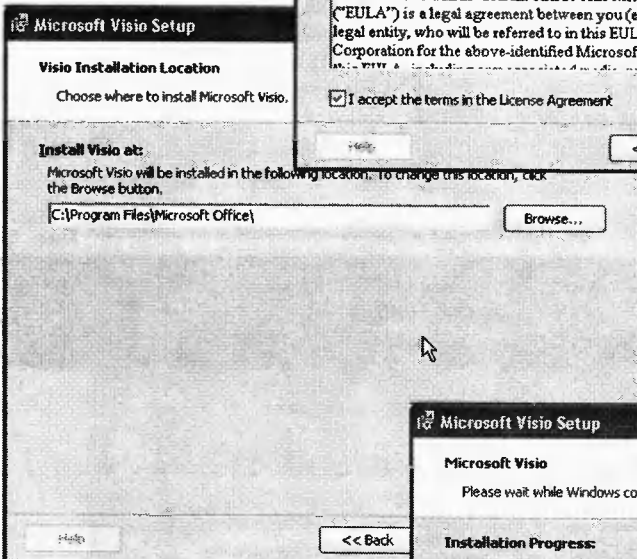


Рис. 7

Рис. 8



цель достигнута (рис.11)!

После этого программа становится доступной из меню пуск (Пуск/Программы), узнает свои файлы и автоматически запускается при попытке их открыть. И вот тут-то иногда начинаются проблемы. Например, есть десятки программ для работы с графическими файлами, наверняка несколько из них у вас уже установлены. Обычно последняя установленная программа перехватывает

работу с файлами определенного типа. Не всегда это удобно. Если с помощью программы приходится открывать данный тип файлов только время от времени, лучше в Проводнике щелкнуть по файлу правой клавишей мыши, выбрать Открыть с помощью и указать программу (рис.12). Нет программы в списке — не проблема, в вашем распоряжении пункт меню Выбрать программу. Остается просмотреть список программ (рис.13), из которого выбирается нужная. Другое дело, если вы хотите прочно связать программу с типом файлов. Тогда на последнем шаге (рис.13) активизируйте опцию Использовать ее для всех файлов такого типа.

Возможно, после нескольких установок программ вы заметите, что на диске осталось мало места. Причиной может послужить и большое число точек восстановления системы. Тогда лучше

Рис. 9

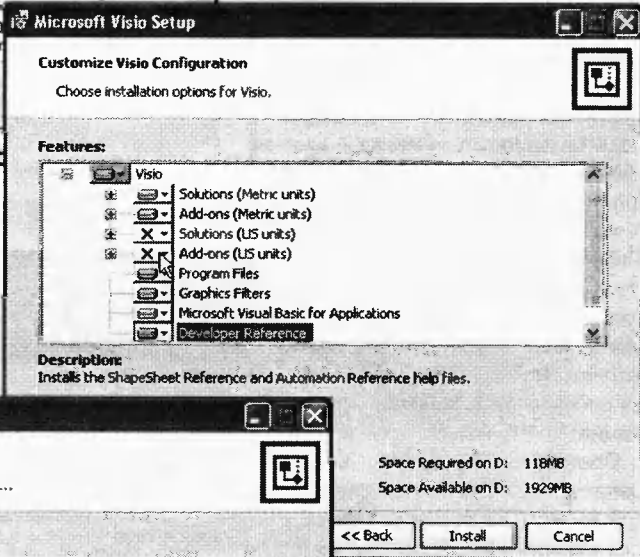


Рис. 10

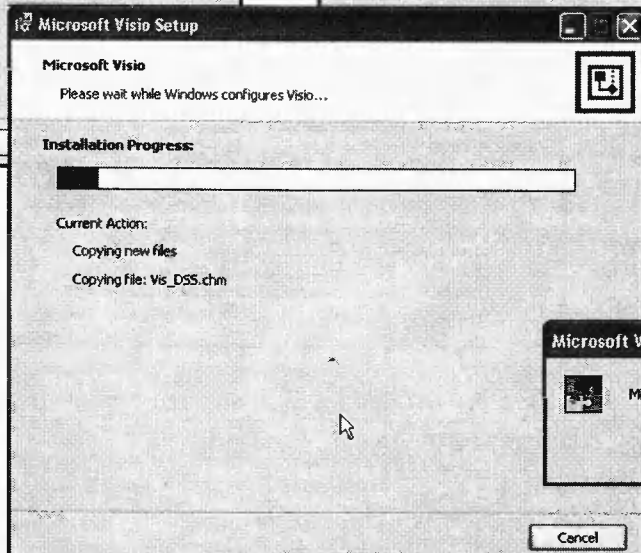


Рис. 11

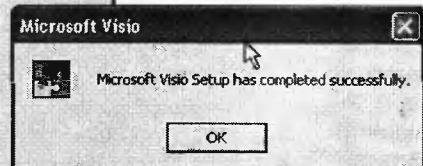


Рис. 12

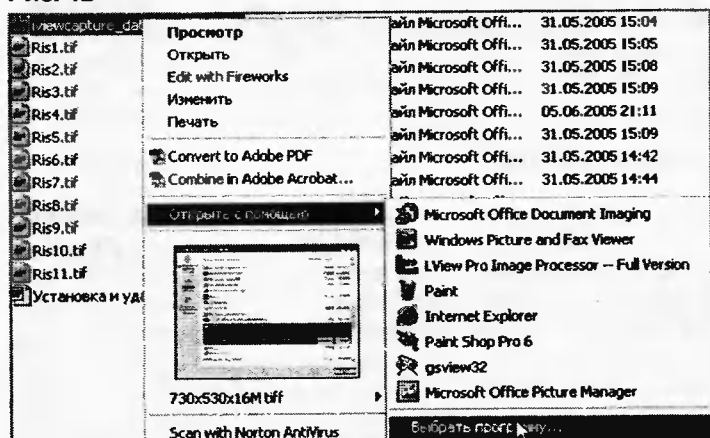


Рис. 13

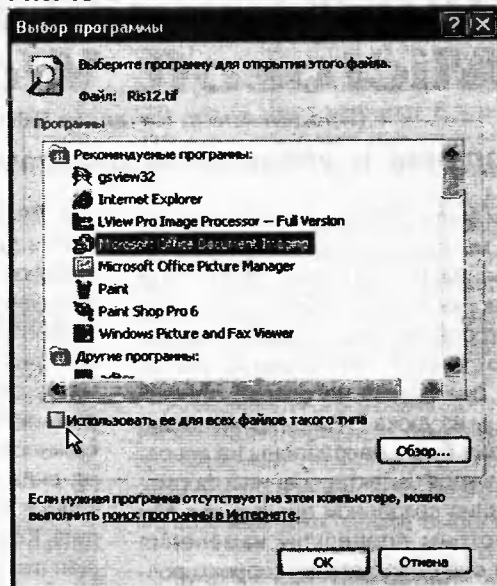
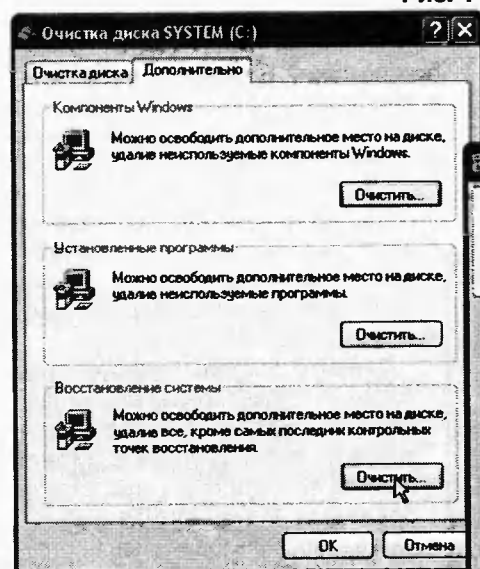
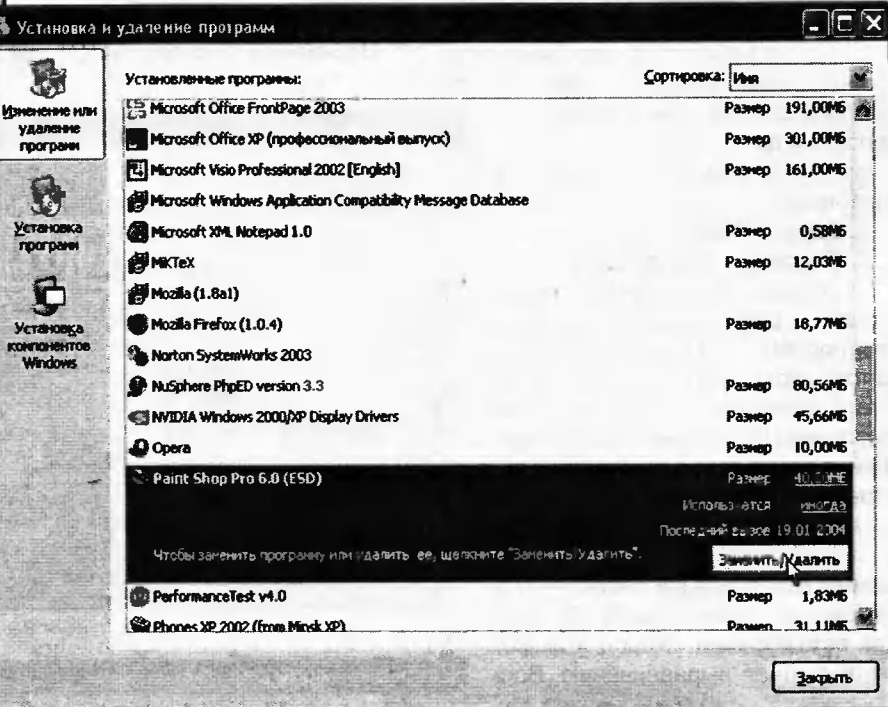


Рис. 14



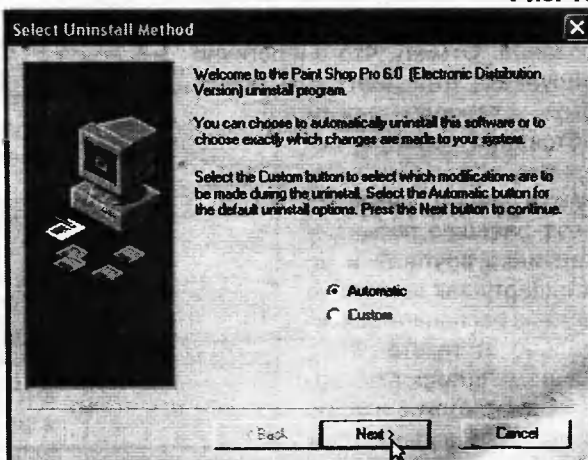
поступить так: подождите несколько дней, убедитесь, что все работает нормально (если нет — воспользуйтесь точкой восстановления). После этого можно приступить к очистке всех точек восстановления, кроме последней — в Проводнике щелкните правой клавишей мыши по значку диска C, **Свойства/Очистка диска/Дополнительно**. На появившейся вкладке без труда находится нужная кнопка (рис.14).

Со временем вы заметите, что некоторые программы вам совсем не нужны или мешают работе других программ. Тогда просто приступаем к их удалению, как раз эта операция вызывает меньше всего вопросов. Действительно, ломать — не строить. Можно подстраховаться и предварительно создать точку восстановления. А дальше — **Пуск/Панель управления/Установка и удаление программ**. При щелчке по нужной (извините, ненужной) программе появляется кнопка удаления (рис.15). Все, что требуется от пользователя, так это определиться со способом удаления (автоматическое или "вручную", предпочтительно первое — рис.16), да нажать кнопку **Далее**. Вот и все.



Мы освоили действительно важный момент работы с Windows. А чтобы наш труд не прошел даром, в следующий раз рассмотрим вспомогательные программы, которые помогают лучше настроить нашу систему и сделать работу намного более удобной и безопасной.

Рис. 16



Эффективная работа с Microsoft Word XP

(Продолжение. Начало в №№1-8/2005)

Создание и управление диаграммами в документе

Как и было обещано на предыдущем уроке, мы продолжим работу над форматированием диаграммы средствами надстройки Microsoft Graph, при помощи которой мы уже создали пока довольно стандартную объемную столбиковую диаграмму на основе данных таблицы из урока 7. Наши дальнейшие действия будут направлены на освоение приемов редактирования и форматирования созданной диаграммы. Мы рассмотрим процедуры изменения типа и вида диаграммы, корректировку исходных данных для ее построения, манипуляции набором и цветовым оформлением элементов диаграммы. Мы научимся вносить в диаграмму дополнительные элементы, такие как подписи данных, линия тренда, заголовки и др.

Ну что ж, приступим к работе и выполним

1. Запуск Word, и

2. Открытие файла **Urok_8.doc** через команду **Открыть** меню **Файл**. После загрузки документа мы увидим под таблицей созданную на прошлом уроке диаграмму Microsoft Graph. Наведем указатель мыши на диаграмму и щелкнем правой клавишей, в результате по периметру диаграммы появятся маркеры в виде небольших черных квадратиков. Это указывает на то, что картинка диаграммы ведет себя как символ — она раздвигает окружающий ее текст как вставленный символ и может подвергаться выравниванию. Воспользуемся этим и выровняем диаграмму по центру, применив соответствующий инструмент форматирования. Отмечу, что аналогично рисункам, для диаграмм может быть установлен режим обтекания текстом, например, вокруг рамки. Однако в данном случае это неэффективно, поскольку картинка крупная и по стандарту так или иначе должна располагаться отдельно от текста. Итак, начнем «шлифовку» нашей диаграммы, что, по сути, представляет собой

3. Редактирование и форматирование диаграмм. Напоминаю, что переход в окно Microsoft Graph выполняется двойным щелчком по картинке диаграммы. Оказавшись там, прежде всего изменим тип и вид диаграммы. Для этого потребуется команда **Диаграмма/Тип диаграммы**, что даст возможность в соответствующем диалоге на вкладке **Стандартная** (рис.1) из списка **Тип** выбрать, например, позицию **Коническая**, а в разделе **Вид** щелчком указать последний из предлагаемых вариантов, название которого вы увидите внизу под образцами, как только его выберете. В нашем случае это **Объемный вариант гистограммы со столбцами в виде конусов**.

Рис. 1

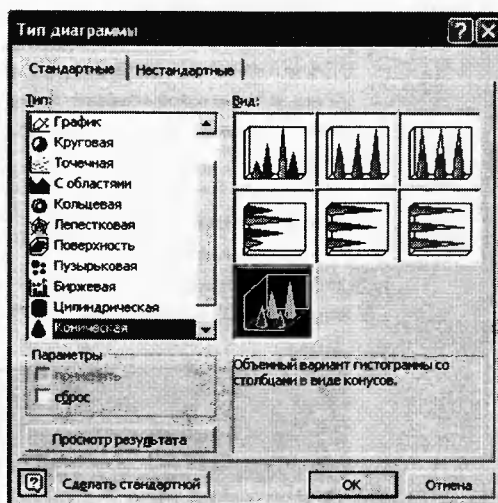


Рис. 2

D:\TrojanWord\Urok_8.doc - Таблица данных

		A	B	C	D	E
	ФИО	I квартал	II квартал	III квартал	IV квартал	
1	Афонина	156	174	173	155	
2	Даньчик А.	113	126	114	129	
3	Дольчин А.	156	174	183	154	
4	Кулаков И.	123	156	147	124	

мы со столбцами в виде конусов. Подтвердим выбор кнопкой **ОК** и увидим результат. Может оказаться так, что после изменения типа и вида диаграммы мы потеряем некоторые из подписей данных. Это случается потому, что отведенных размеров диаграммы недостаточно. Однако не торопитесь изменять размеры диаграммы — это может оказаться неэффективным, потому что с изменением размеров автоматически увеличивается размер букв подписей. В принципе, с этим можно справиться, выделяя неудачно оформленные элементы диаграммы (подписи осей, легенду и т.п.) и изменяя шрифт и его размеры при помощи соответствующих инструментов, хорошо вам знакомых по форматированию текста. Но лучше мы сначала немного подредактируем диаграмму, исключив избыточные данные.

Давайте оставим информацию только по двум первым кварталам, для чего нам потребуется отредактировать таблицу данных. Может оказаться так, что она у вас не отображается. В этом случае подключим ее, воспользовавшись инструментом **Режим таблицы**. В таблице данных наведем указатель мыши на заголовок столбца **С**, и нажав левую кнопку, выделим столбцы **С** и **D** (рис.2), а затем выполним команду **Данные/Исключить строку/столбец**, и далее **Данные/Ряды образуют столбцы**. Теперь диаграмма не перегружена и смотрится значительно лучше (рис.3).

Дополним диаграмму заголовком и иначе расположим

Рис. 3

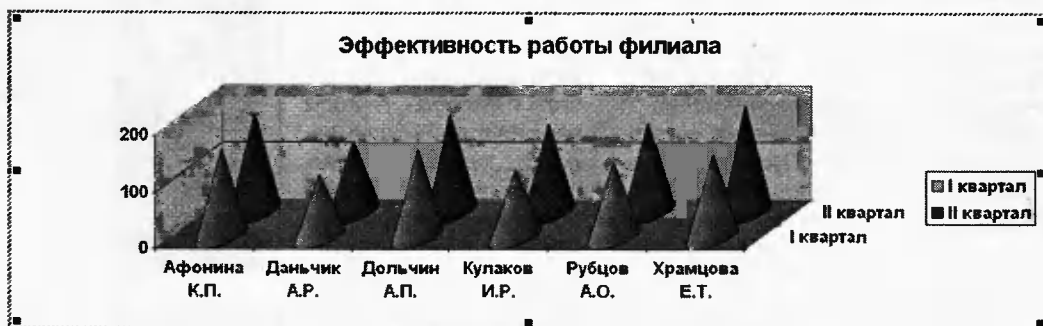
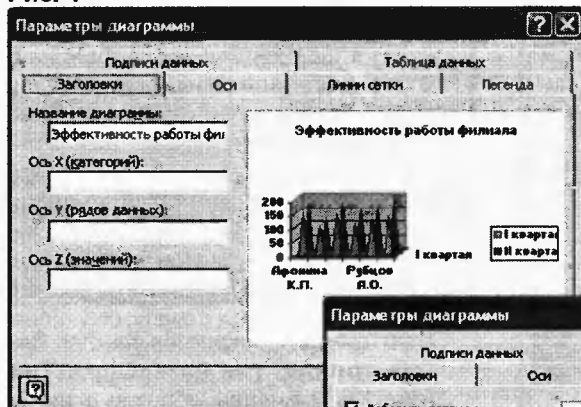


Рис. 4



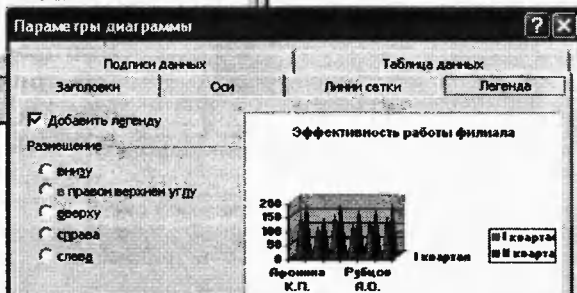
легенду. Для этого воспользуемся диалогом **Диаграмма/Параметры диаграммы**, где на вкладке **Заголовки** (рис.4) в поле **Название диаграммы** напечатаем текст "Эффективность работы филиала". Перейдем на вкладку **Легенда** (рис.5) и при установленном флажке **Добавить легенду** выберем переключатель размещения **справа**. И это еще не все! На вкладке **Подписи данных** (рис.6) в разделе **Включить в подписи** установим флажок **значения**, и только теперь завершим работу с диалогом кнопкой **ОК**. Полюбуемся на результат (рис.7). Что еще здесь хотелось бы сделать, так это, чтобы лучше читались данные рядом с каждым конусом. А также, возможно, изменить цветовое решение.

Здесь вступают в свои права приемы форматирования диаграммы. Начнем с раскраски ярлычков подписей данных. По умолчанию они прозрачные и читаются на фоне картинки диаграммы. Наведем курсор мышки на подписи данных по I кварталу (рис.7) и щелкнем правой клавишей. В результате выделятся ярлычки всего этого ряда, и тогда мы воспользуемся инструментом **Цвет заливки** и выберем, например, травяной цвет. Аналогично поступим с другим рядом. Далее снимем фон стенок и основания. Для этого лучше применить контекстное меню. Найдём всплывающую подсказку **Стенки** и щелкнем правой кнопкой, а затем выберем команду **Формат стенок**. В открывшемся диалоге (рис.8) в разделе **Заливка** выберем переключатель **прозрачная** и **ОК**. Повто-

рим эти действия для основания.

Теперь позаботимся о внешнем виде области диаграммы. Очень эффективно для нее смотреться заливки градиентного типа, которые придают диаграмме дополнитель-

Рис. 5



ую процедуру **ОК**. Можно также по-иному решить и представление конусов. Выделим один из рядов и вызовем **Способы заливки** из списка инструмента **Цвет заливки**. В этом случае у нас откроется диалог со вкладкой **Узор**, где можно указать вариант узора, например, **Широкий диагональный 1**, а также цвет штриховки и фона в соответствующих списках (рис.10). Можете поэкспериментировать в этом направлении и подобрать наиболее интересный для вас дизайн диаграммы.

Объемные гистограммы допускают также манипуляции с трехмерными параметрами. Для этого предназначена специальная команда **Диаграмма/Объемный вид**, после выполнения которой в диалоге **Формат трехмерной проекции** (рис.11) при помощи соответствующих кнопок можно изменить возвышение (то есть угол высоты точки, откуда зритель смотрит на картинку) и повернуть угол зрения на трехмерную проекцию вправо или влево. В примере (рис.12) я установила возвышение 20 и поворот влево до 30.

Рис. 6

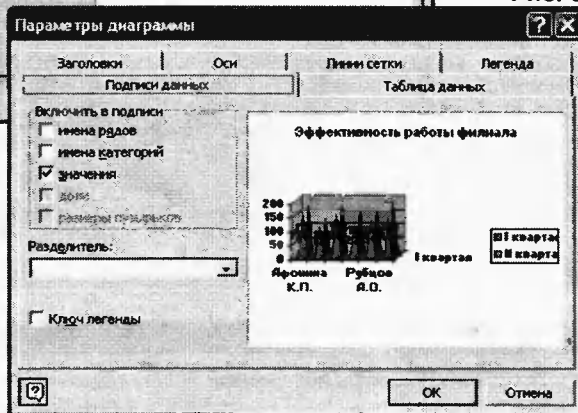


Рис. 7



Рис. 8

ный объем. Выделим на диаграмме элемент **Область диаграммы** и воспользуемся инструментом **Цвет заливки**, где под набором обычных цветов найдем и выберем позицию **Способы заливки**. В развернувшемся окне (рис.9) на вкладке **Градиентная** в разделе **Цвета** выберем переключатель **заготовка**, затем в списке **Название заготовки** укажем, например, **туман**, а далее в разделе **Тип штриховки** щелкнем переключатель **от центра** и выберем первый образец в разделе **Варианты**, завершая

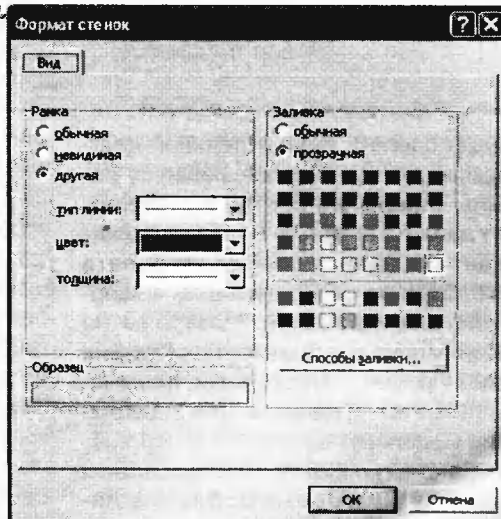
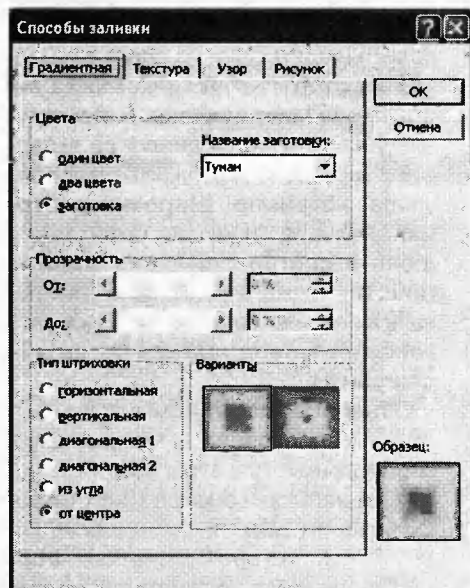
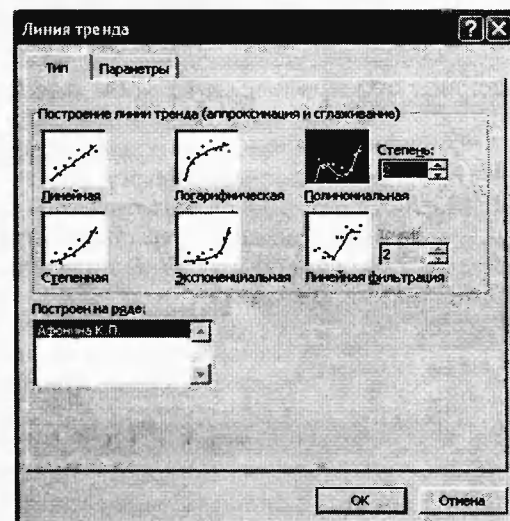


Рис. 9



В завершение я хотела бы показать интересный прием подбора моделирующей функции для графика и построения прогноза на будущее по исходным данным. Удобнее

Рис. 13



будет сделать это в отдельной диаграмме для данных, касающихся работы отдельного сотрудника. Поэтому давайте выделим в таблице фрагмент из первых двух строк и пяти столбцов, то есть данные за 4 квартала для Афонинной. Построим на этой основе диаграмму типа График вида График с маркерами, помечающими точки данных. Для добавления моделирующей кривой и прогноза на следующий квартал вызываем команду **Диаграмма/Добавить ли-**

Рис. 10

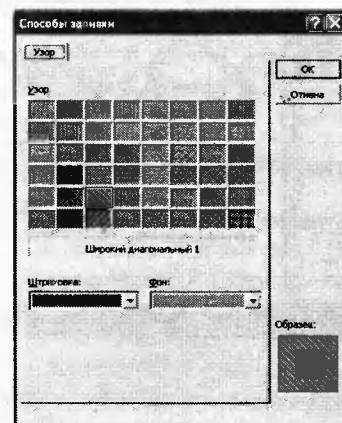
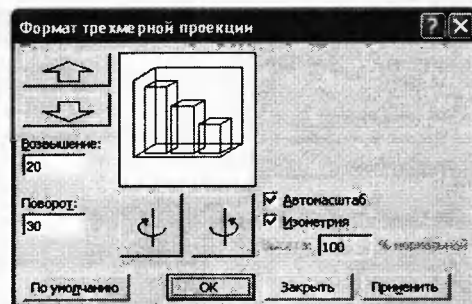


Рис. 11

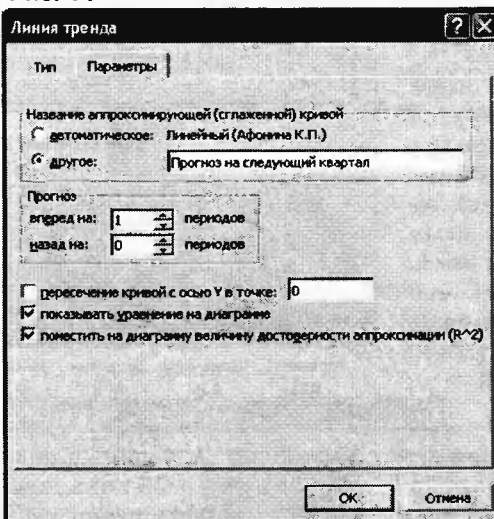


нию тренда. Откроется диалог **Линия тренда**, где на вкладке **Тип** (рис.13) выбираем образец с типом моделирующей функции, например,

Рис. 12

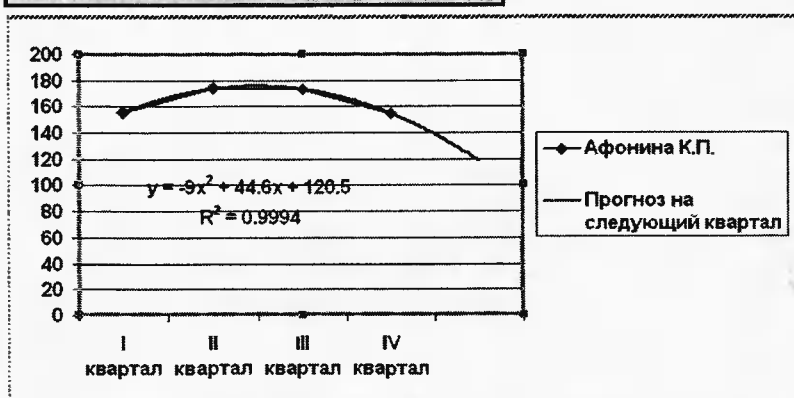


Рис. 14



Полиномиальная Степень 2. Далее на вкладке **Параметры** (рис.14) выбираем период прогноза, указав в разделе **Прогноз** в списке **вперед на**, например 1, а также для получения аналитического выражения моделирующей функции и точности аппроксимации устанавливаем флажки **показывать уравнение на диаграмме** и **поместить на диаграмму величину достоверности**

Рис. 15





верности аппроксимации. Можно также задать название для моделирующей кривой, в разделе **Название аппроксимирующей (сглаженной) кривой** выбрав переключатель **другое** и напечатав рядом, например, "Прогноз на следующий квартал". Результат наших усилий можно сравнить с **рис. 15**, где мы видим дополнительную кривую с продолжением на следующий квартал, а также квадратный трехчлен моделирующей функции. Обратите внимание на высокую точность моделирования, о которой судят по величине коэффициента R^2 — чем он выше, тем точнее прогноз. На этом наши нынешние труды мы завершаем, еще раз закрепив все действия по таблице **Краткое содержание Урока 9**, и выполняя

4. Сохранение файла под новым именем, вызвав команду **Файл/Сохранить как**, и скорректировав имя файла на **Urok_9.doc** и нажав **Enter**, а затем

5. Завершение работы Word, вызвав команду **Файл/Выход**.

(Продолжение следует)

Краткое содержание Урока 9	
Действие	Реализация
1. Запуск Word	Щелчок по значку
2. Загрузка файла Urok_8.doc	Файл/Открыть , указать маршрут и двойной щелчок по имени файла Urok_8.doc
3. Редактирование и форматирование диаграмм	
Редактирование диаграмм	
Возврат к режиму редактирования уже созданной диаграммы	Двойной щелчок внутри диаграммы
Выход из режима редактирования диаграммы и возврат в Word	Щелчок вне области диаграммы
Изменение данных	Выделить ячейку таблицы данных и ввести новые данные
Исключение или Включение столбцов или строк в таблице данных	1. В таблице данных выделить мышью заголовки столбцов или строк. 2. Данные/Исключить или Включить строку/столбец
Изменение типа диаграммы	Диаграмма/Тип диаграммы или инструмент Тип диаграммы
Изменение расположения рядов данных	Данные/Ряды образуют строки или Ряды образуют столбцы , либо инструменты По строкам По столбцам
Построение линии тренда	Выделить ряд данных и Диаграмма/Добавить линию тренда , затем указать на вкладке Тип функцию аппроксимации, на вкладке Параметры — период прогноза, отображать или нет уравнившие линии и значение коэффициента достоверности R^2
Форматирование диаграмм	
Оформление элементов диаграммы	В режиме редактирования диаграммы выделить элемент и использовать инструменты на панели Форматирование , Стандартная или соответствующие команды меню Формат
Включение/отключение линий сетки	Инструменты Линии сетки оси категорий Линии сетки оси значений
Включение/отключение легенды	Инструмент Легенда
Добавление заголовков, подписей данных, управление осями и линиями сетки	Диаграмма/Параметры диаграммы/Заголовки , Подписи данных , Оси , Линии сетки и др. и выбрать соответствующие переключатели и установить флажки
Управление параметрами трехмерных проекций диаграммы	Диаграмма/Объемный вид
4. Сохранение файла под новым именем	Файл/Сохранить как , указать маршрут, впечатать имя файла Urok_9.doc и Enter
5. Завершение работы Word	Файл/Выход

ОРФО 2004 проверяет правописание

В.КУЦ,

E-mail: victor_kootz@mail.ru

В среде офисного программного обеспечения сложилась довольно парадоксальная ситуация — одна из самых наиболее востребованных программ, постоянно используемая всеми без исключения ответственными пользователями сверхпопулярного текстового процессора Microsoft Word, сама по себе не особо известна широкой публике. А ведь без нее, являющейся одной из важнейших составных частей офисного пакета от Microsoft, он никогда бы не имел и сотой доли нынешней популярности на просторах бывшего Советского Союза. Я имею в виду систему проверки правописания ОРФО, разработанную российской компанией "Информатик" (www.informatic.ru) несколько лет тому назад, сильно уре-

занный вариант которой встроен во все версии Microsoft Office, начиная с "антикварного" MS Office 95. В этом легко убедиться, заглянув в русское издание пакета MS Office 9x/2000/XP/2003 в раздел "Справка/О программе".

Итак, каждый из пользователей, хотя бы раз запускавших Word, знает, что делают встроенные туда модули ОРФО — они проверяют орфографию, грамматику и стилистику документа, расстановку в нем переносов и подбирают (в случае необходимости) синонимы. Однако нетрудно предположить, что выпускаемая компанией "Информатик" автономная версия программы ОРФО делает то же самое, только гораздо лучше. Полезность такой версии программы для проверки правописания никому объяснять не

нужно. Грамотность большинства из нас — что уж греха таить — оставляет желать лучшего. Более того, даже у достаточно грамотных пользователей при наборе текста нередко случаются досадные опечатки. Так что проверка, пусть даже только на предмет правильного написания слов, не помешает никому, поэтому ОРФО окажется полезной каждому.

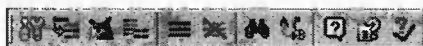
С другой стороны, вполне закономерно встает вопрос — зачем, имея локализованную версию одного из офисных пакетов от Microsoft, имеющих собственные программы проверки орфографии (возможностей которых вполне достаточно среднему пользователю), устанавливать себе что-то еще? Ответу. Все интегрированные системы проверки правописания гораз-

до менее функциональны, чем ОРФО, и, даже находясь в пределах одного винчестера, никак между собой не связаны. А ведь вы работаете с текстом далеко не только в MS Office, не правда ли? А ОРФО можно рассматривать как программу-симбионт, которая без проблем сопрягается с подавляющим большинством других программ, работающих с текстом. Поэтому при вызове программы проверки орфографии из любой из них будет вызываться один и тот же модуль ОРФО, который будет использовать один и тот же словарь. Удобно!

Итак, пакет ОРФО 2004 — это уникальный набор лингвистических программ, предназначенных для качественной проверки и исправления текстов на различных языках и в той или иной мере интегрируемый в популярные офисные пакеты Microsoft Office 97/2000/XP/2003: Access, Excel, Outlook, PowerPoint, Word; MS Outlook 4.x/5.x/6.x; MS FrontPage 97/98/2000/XP/2003; MS Publisher 98/2000/XP/2003; Lotus WordPro 96/97/2000; Corel WordPerfect 7.0/8.0/9.0; Adobe PageMaker 6.x/7.0; QuarkXPress 4.x/5.x.

Естественно, наибольшие преимущества от применения ОРФО получают пользователи Microsoft Word, к которому подключаются все входящие в состав программы основные модули проверки правописания, расстановки переносов, грамматики, стилистики и тезауруса, а для доступа к другим ее модулям можно воспользоваться удобной панелью инструментов (рис.1).

Рис. 1



Единственное, за что можно "попенять" разработчикам Панели инструментов ОРФО — так это за ее некоторую назойливость. Один раз установленная, она стремится всегда быть на виду. Ее, конечно, можно отключить — но только до очередной перезагрузки Word'a, после чего панель ОРФО возвращается на прежнее место.

В других программных продуктах ОРФО 2004 предлагает гораздо меньшую функциональность. Например, при ее сопряжении с Excel, PowerPoint, Access, Exchange, Visual Fox Pro и Corel Draw оказывается доступной только проверка правописания, в программах MS Works и FrontPage 2000/XP/2003 дополнительно к проверке правописания имеется возможность подобрать синоним для заданного слова, а в MS

Publisher и Adobe PageMaker добавляется и проверка расстановки переносов.

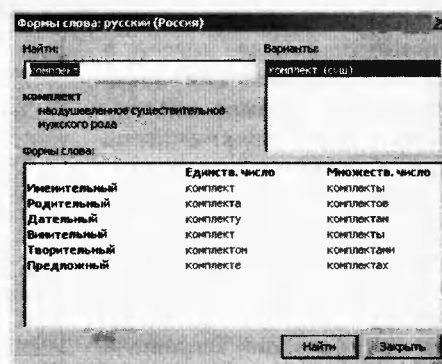
Не следует упускать из виду, что во всех вышеперечисленных приложениях появляются (либо обновляются) соответствующие модули для работы с текстом на русском языке. Например, если в какой-либо прикладной программе есть возможность проверять орфографию, то модуль ОРФО будет делать это, но если в программе не предусмотрена проверка грамматики, то она не сможет вызываться из этой программы, даже если соответствующий модуль ОРФО и будет присутствовать в компьютере. Значит, в поддерживаемых программой ОРФО 2004 приложениях оказываются доступны только те возможности ОРФО, которые вызываются стандартными для данной программы способами. Дополнительное преимущество ОРФО (по сравнению со стандартными модулями проверки орфографии) в том, что он может сопрягаться не только с локализованными, но и с оригинальными англоязычными версиями вышеперечисленных программ.

ОРФО 2004 выпускается в трех различных вариантах: наиболее простой базовый комплект для русского языка, более функциональный профессиональный комплект для русского языка и многоязычный комплект для английского, немецкого, французского, испанского, итальянского и португальского языков.

Базовый комплект для русского языка обеспечивает встроенную проверку орфографии в подавляющем большинстве современных офисных пакетов. Словарь общей лексики содержит около 220 тыс. терминов. По сравнению с модулем проверки орфографии в MS Office, в ОРФО 2004 включены дополнительно 10 тысяч новых слов (фамилии, географические названия, новые слова и т.п.). Основная его особенность — ведение так называемого морфологического пользовательского словаря, в который автоматически добавляются все словоформы (можно выбирать одну из предложенных парадигм). Таким образом, однажды добавленное в словарь новое слово будет корректно восприниматься при любом изменении последнего согласно правилам русского языка (рис.2).

Еще одна интересная черта модуля проверки орфографии — одновременная работа с русско-ан-

Рис. 2



глийскими документами. Тем более, что ОРФО, в отличие от Word'a (который аналогичную возможность обеспечивает только в случае корректной расстановки языковых признаков), различает языки по используемым символам, а потому проверка двуязычных текстов осуществляется гораздо корректнее.

Помимо проверки орфографии, базовый комплект ОРФО "умеет" правильно расставлять переносы, в том числе и так называемые "мягкие" переносы, во всех словах документа, что может пригодиться при верстке многоколоночных буклетов и сложных таблиц (рис.3). Предусмотрена возможность выбора пользователем одного из двух вариантов настройки на книжное (для обычных текстов) или газетное (при форматировании текста в узкие колонки) качество переноса. Модуль расстановки переносов, помимо MS Word, также работает еще и с MS Publisher, Adobe PageMaker, QuarkXPress 4.x/5.x и Corel WordPerfect 7.0/8.0.

Профессиональный комплект для русского языка, помимо функциональности базового комплекта, дополнительно обеспечивает гораздо более эффективную, чем в MS Office, грамматическую и стилистическую проверку текстов. Этот модуль осуществляет проверку правильности расстановки запятых, согласования окончаний слов, употребления слов в различных

Рис. 3

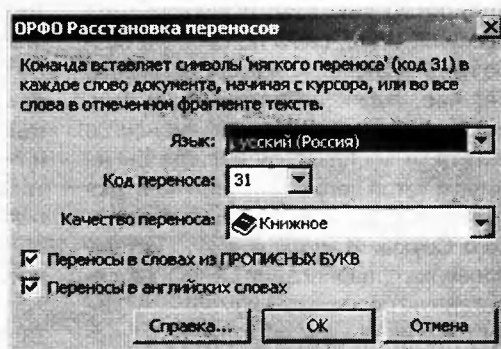


Рис. 4

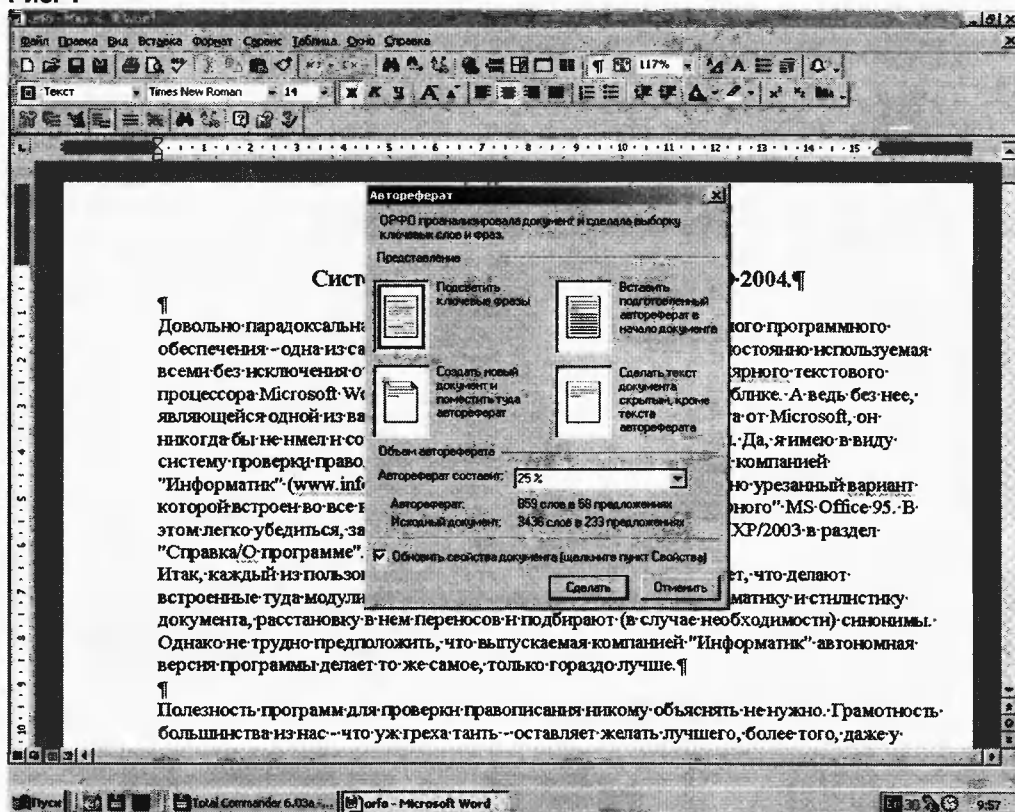
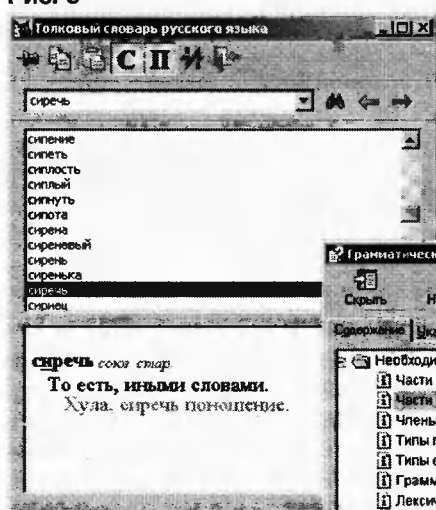


Рис. 5



контекстах и т.п., опираясь на большую базу грамматических данных, включающую в себя более 40 групп правил, и проверяет тексты в соответствии с одним из трех стилей: строгим (когда задействованы все правила), для деловой и для обычной переписки. Правда, работает он только с MS Word 9x/2000/2003, но его использование вместо штатной системы грамматической и стилистической проверки текстов обес-

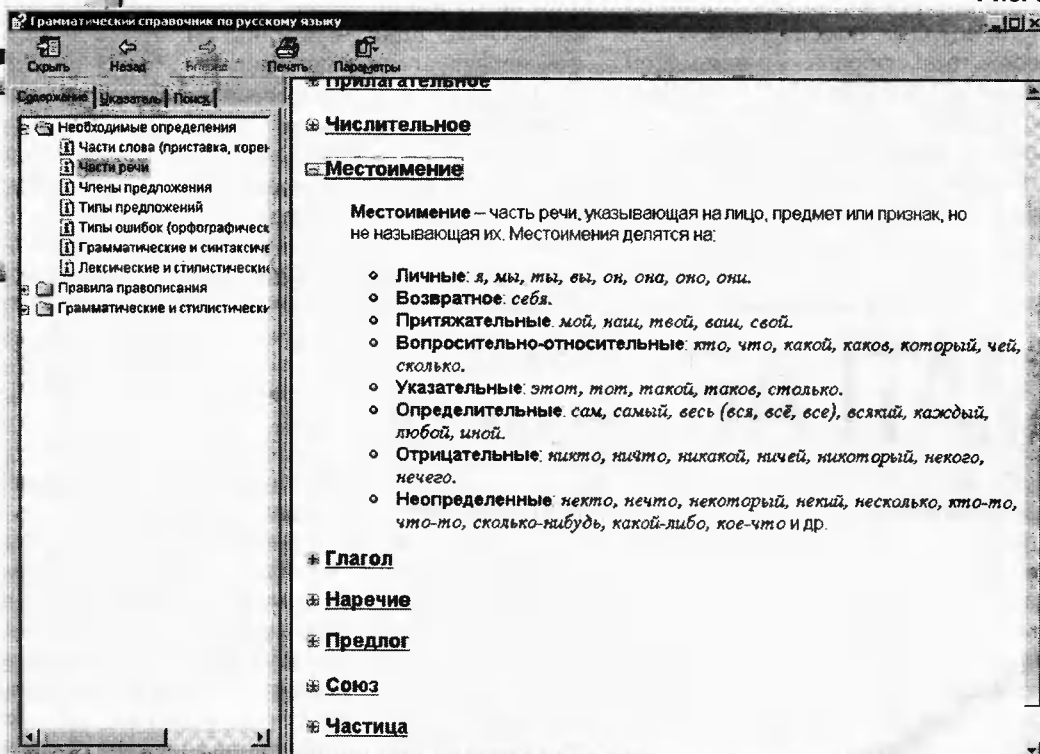
печивает заметное повышение качества и читабельности текстов.

Еще один модуль — тезаурус, или словарь синонимов русского языка, обеспечивающий распознавание синонимов и антонимов русских слов, содержит порядка 60000 слов и выраже-

жет быть вызван из MS Word.

Грамматический справочник русского языка, выполненный в виде стандартного HLP-файла, представляет собой хорошо подготовленный путеводитель по правилам русской грамматики (рис.6). Данный модуль может

Рис. 6



ний. Найденные синонимы сразу ставятся в ту же форму, что и искомое слово. Основное его отличие от стандартного модуля Word — корректная работа со сложными словоформами.

Существенно ускорить обработку больших текстовых массивов позволит система авторефератирования документов. Автореферат автоматически составит список ключевых слов текущего документа на русском языке и выберет наиболее информативные фразы с этими словами в зависимости от заданного пользователем объема (рис.4).

Толковый словарь русского языка, включающий более 110 тыс. русских слов и выражений и предлагающий их развернутые толкования вместе с примерами употребления (рис.5). Данный модуль используется как автономная программа, и мо-

быть вызван из MS Word или использоваться как автономная программа. Также он вызывается из окна пояснений при грамматической проверке.

Функциональность многоязычного комплекта для английского, немецкого, французского, испанского, итальянского и португальского языков приблизительно соответствует функциональности базового комплекта для русского языка, за исключением модуля расстановки переносов.

Профессиональный и многоязычный комплект ОРФО обеспечивают подключение дополнительных тематических словарей, содержащих специфические термины по разным отраслям человеческой деятельности.

Кроме того, в состав пакета включен ОРФО Редактор — самостоятельное приложение, обеспечивающее проверку орфографии в собственном миниредакторе (чрезвычайно похожем на WordPad). С его помощью можно проверять текст, набранный в самом этом редакторе, либо текст в буфере обмена Windows (рис.7). Он поддерживает практически все лингвистические средства ОРФО, но, к сожалению, имеет существенное ограничение — «понимает» только тексты в формате RTF.

Еще одно чрезвычайно полезное

приложение пакета — ОРФО Агент — дает возможность активизировать модуль проверки орфографии ОРФО при помощи «горячих» клавиш, благодаря чему можно осуществлять проверку орфографии в тех приложениях, в которых подобная лингвистическая функция не предусмотрена (рис.8). Единственное (да и то, чисто теоретическое) ограничение ОРФО Агента в том, что он функционирует только в стандартных редактируемых окнах известных ОРФО типов (например, Notepad, WordPad, Write, ICQ и т.п.), при этом проверяется либо выделенный фрагмент, либо текст после курсора.

После установки пакета ОРФО Агент будет запускаться автоматичес-

MATLAB. Пакет Simulink

Система MATLAB содержит набор пакетов прикладных программ (MATLAB Application Toolboxes), среди которых особое место занимает Simulink. Знакомством с этим пакетом мы заканчиваем учебный цикл, предложенный в [1]. Ниже даны начальные сведения, в следующих статьях будут рассмотрены два примера с возрастающей сложностью.

Пакет Simulink предназначен для математического моделирования динамических систем, представленных своей функциональной блок-схемой, именуемой моделью. Более полное определение назначения пакета дано

в [2]. Среди литературных источников для знакомства с Simulink подходит [3].

Познакомимся с этим пакетом на примере решения задачи обыкновенных дифференциальных уравнений (ДУ), поскольку аналоговые динамические системы часто описываются такими уравнениями. Более того, мы рассмотрим пример, описанный в [4], где также дана элементарная теория вопроса.

Для построения функциональной блок-схемы моделируемых систем и процессов, Simulink имеет библиотеку блочных компонентов и редактор блок-схем.

Рис. 7

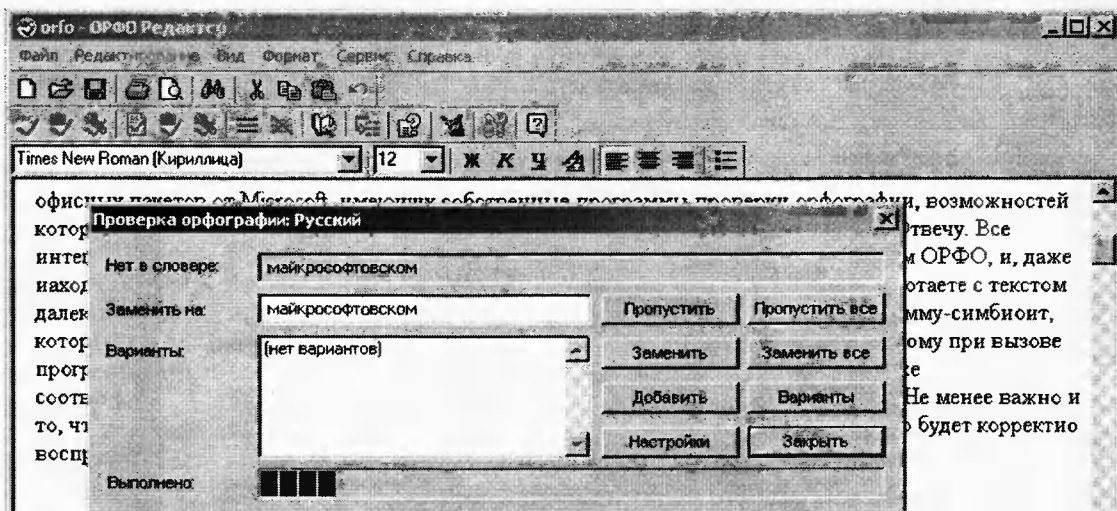
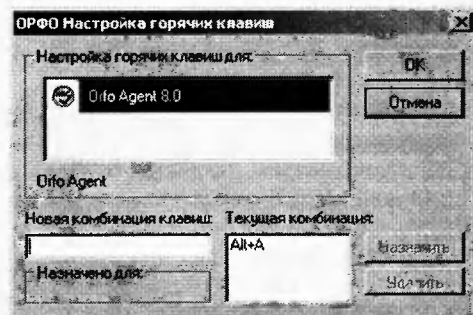


Рис. 1 Формы слова

Итак, предлагаемый Вашему вниманию пакет ОРФО 2004 — это уникальный набор лингвистических программ, предназначенных для качественной проверки и исправления текстов на различных языках в популярных пакетах Microsoft Office 97/2000/XP/2003: Access, Excel, Outlook, PowerPoint, Word; MS Outlook 4.x/5.x/6.x, MS FrontPage 97/98/2000/XP/2003, MS Publisher 98/2000/XP/2003, Lotus WordPro 96/97/2000, Corel WordPerfect 7.0/8.0/9.0, Adobe PageMaker 6.x/7.0, QuarkXPress 4.x/5.x. Естественно, что наиболее удобно работать с ОРФО все-таки в микрофотоскоп Word, используя для вызова различных модулей программы удобную панель инструментов.

Рис. 8



ки при старте ОС, а его значок — постоянно находиться в системном трее панели задач Windows.

В целом пользоваться ORFO 2004 довольно удобно и полезно, особенно тем, кто интенсивно работает с документами.

Б.КИСЕЛЕВ,
г.Минск, БГАТУ,
каф.ВТ.

Пусть необходимо исследовать процесс, описываемый дифференциальным уравнением [4]

$$a_2 \frac{d^2 y}{dt^2} + a_1 \frac{dy}{dt} + a_0 y = x(t) \quad (1)$$

при $x(t) = 1(t)$ (ступенчатое воздействие) и начальных условиях: $t = t_0$, $y = y_0$, $y' = y'_0$.

Такие уравнения описывают движение колебательных систем: груза, подвешенного на пружине, маятника, электрического RLC-контура и т.д. под воздействием возмущения $x(t)$.

Для моделирования уравнение (1) удобно привести к виду

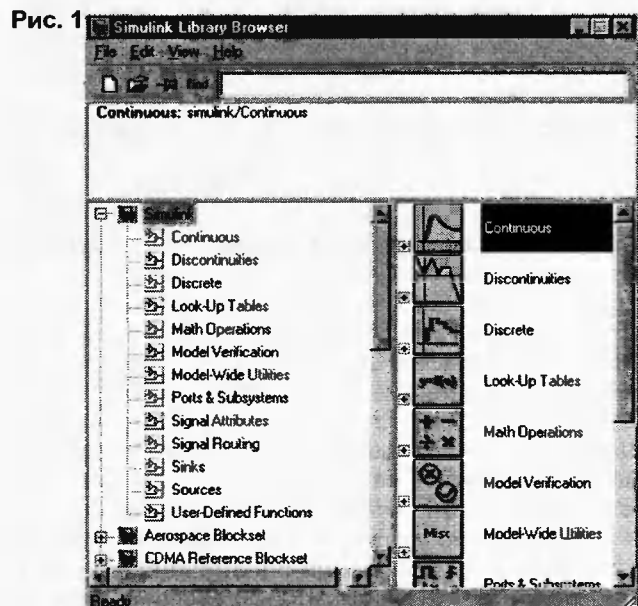
$$y'' = \frac{1}{a_2}(x(t) - a_1 y' - a_0 y) \quad (2)$$

или к виду:

$$\begin{aligned} y' &= y_1 \\ y_1' &= \frac{1}{a_2}(x(t) - a_1 y_1 - a_0 y) \end{aligned} \quad (3)$$

Построение модели

В окне **MATLAB** выберем кнопку **Simulink** (перед кнопкой ?). Появится окно **Simulink Library Browser** (рис.1).



Слева в нем расположен каталог пакетов, а справа — разделы библиотеки блоков пакета. По умолчанию обычно в левой части окна раскрыт пункт **Simulink** (перечислены его разделы), а справа — иконки разделов: **Continuous** (непрерывные компоненты), **Discontinuous** (кусочно-непрерывные компоненты), **Discrete** (дискретные компоненты), и т.д. Удачно скомпоновать разделы библиотеки непросто, поэтому их состав и наименования меняются от версии к версии, но имена блоков (компонентов) не меняются. Поэтому, если на вашем компьютере другая версия пакета, найти необходимый блок несложно.

Если пункт **Simulink** не раскрыт, щелкнем ЛКМ (левой кнопкой мыши) на знаке "+" возле пункта **Simulink**.

Чтобы увидеть набор имеющихся блоков в разделе, нужно щелкнуть ЛКМ на данном разделе слева или на "+" возле иконки раздела.

В главном меню окна **Simulink Library Browser** выберем **File – New – Model** — появится окно **untitled**. В этом окне будем создавать модель, "перетаскивая" соответствующие блоки из библиотеки Simulink в окно **untitled**.

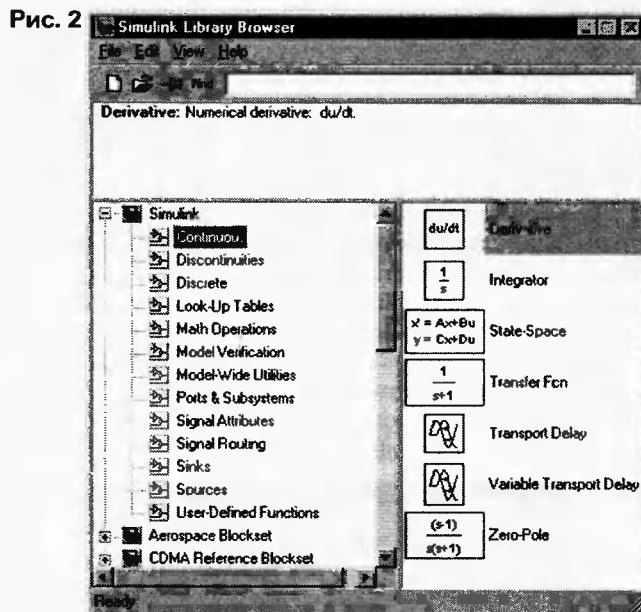
Для модели нам понадобятся два интегратора, т.к. старшая производная в (2) имеет второй порядок. Соответствующий блок находится в разделе **Continuous** (рис.2). Перета-

щим мышкой иконку этого блока в окно **untitled**. Для этого, нажав ЛКМ на иконке интегратора и удерживая ее нажатой, перетащим блок в окно **untitled** и отпустим ЛКМ — появится пиктограмма блока. Второй интегратор можно поместить в окно таким же путем или выполнить процедуру копирования.

Коэффициенты уравнения (2) a_0 , a_1 реализуем двумя блоками **Gain** (усилитель или умножитель), которые перетащим из раздела **Math Operations**. Заметим, что одинаковые блоки автоматически получают свои порядковые номера. Коэффициент $\frac{1}{a_2}$ реализуем блоком **Fcn** (универсальный вычислитель), который находится в разделе **User-Defined Functions**.

Источник входного воздействия $1(t)$ можно представить блоком **Constant**, который находится в разделе **Source** (источник). На его выходе сохраняется постоянная величина. Перетащим мышкой иконку этого блока в окно **untitled**.

Регистратор выходного сигнала (решения уравнения) выберем, например, в виде осциллографа. Соответствующий блок находится в разделе **Sinks**. Перета-

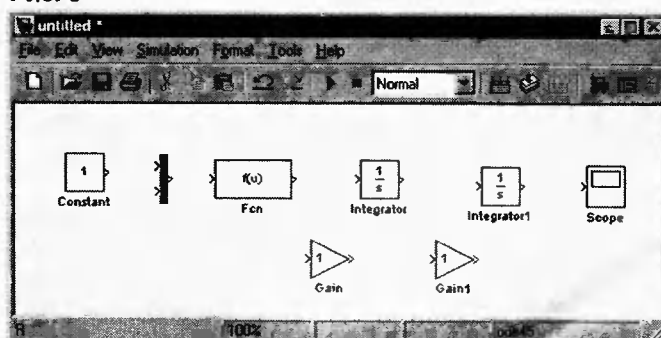


щим мышкой иконку этого блока в окно **untitled**.

Объединитель сигналов **Mux**, который находится в разделе **Signal Routing**, также переместим в окно **untitled**.

В результате это окно может принять вид, соответствующий рис.3. Значки ">" на границах блоков показывают точки входов и выходов. Обратим внимание, что источник сигнала имеет только выходную точку, а регистратор — только входную.

Рис. 3



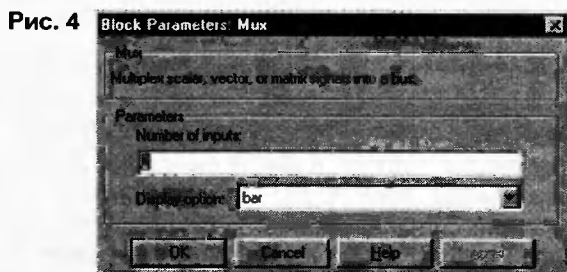
Возможно, понадобятся другие блоки, тогда будем их добавлять по мере необходимости. Сохраним выполненное (выбрав **File – Save**) под именем **Difur**. По умолчанию этому файлу дается расширение **.mdl**. В дальнейшем при открытии файла с таким расширением автоматически будет открываться и Simulink.

Теперь необходимо установить связи между блоками. При моделировании дифференциальных уравнений удобнее начинать с конца. Очевидно, что на входе осциллографа должен быть сигнал y . Этот сигнал можно снять с выхода блока **Integrator**, на вход которого подается y' , поэтому нужно соединить выход интегратора с входом блока **Scope**. Для этого указатель мыши установим на вход блока **Scope**, указатель должен измениться на крестик, при нажатой ЛКМ проведем мышкой до выхода интегратора и, когда появится двойной крестик, отпустим ЛКМ.

Сигнал y' возьмем с выхода блока **Integrator1**, на вход которого подается y'' , поэтому соединим вход блока **Integrator** с выходом блока **Integrator1**.

На входе блока **Integrator1** должен быть сигнал y'' . Его можно получить, вычислив правую часть выражения (2). В качестве коэффициента " $-a_0$ " используем блок **Gain**, а в качестве коэффициента " $-a_1$ " — **Gain1**. Соединим вход блока **Gain** с сигналом y , т.е. с линией между интеграторами. Позже установим коэффициент **Gain1** равным " $-a_1$ ". Тогда на выходах блоков **Gain** будем иметь второе и третье слагаемые в скобках (2). Первое слагаемое получим с выхода блока **Constant**. Все эти три выхода объединим блоком **Mux**.

Прежде чем сделать соответствующие соединения, необходимо увеличить число входов у блока **Mux**. Дважды щелкнем ЛКМ на блоке **Mux** — откроется его блок параметров (рис.4). В строке **Number of inputs:** (число входов) установим значение 3 и нажмем **Apply – OK**.

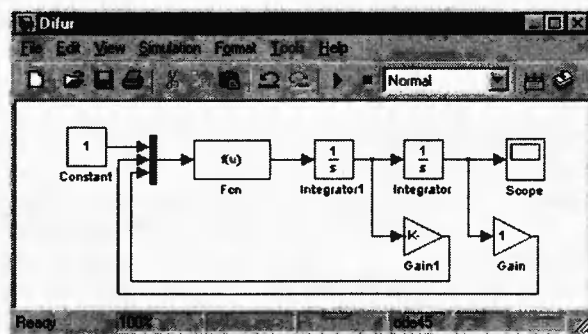


Установим связи, для этого соединим выходы блоков **Gain**, **Gain1** и **Constant** с входами блока **Mux**. Для улучшения расположения получившихся связей установим указатель мыши на требуемую линию и при нажатой ЛКМ поднимем ее вверх либо опустим вниз, чтобы избежать пересечения с другими блоками. Рекомендуем поманипулировать линиями связей, чтобы представлять возможности их изменения. "Потаскайте" линии за углы.

Осталось соединить **Mux** и **Fcn** (на выходе последнего будет правая часть выражения (2)), а также блоки **Fcn** и **Integrator1**. В результате получим схему, показанную на рис.5.

Теперь определим параметры блоков. Пусть значения коэффициентов уравнения (1) следующие: $a_0 = 1$, $a_1 = 0.04$, $a_2 = 0.01$, а начальные условия — нулевые. Установим курсор мыши на требуемом блоке, например **Gain**, и дважды щелкнем ЛКМ. Откроется окно **Block Parameters:**, в кото-

Рис. 5



ром в строке **Gain** запишем -1 , и нажмем **Apply – OK**.

Аналогично поступим с блоком **Gain1**, в блоке параметров которого наберем -0.04 . Обратите внимание на изменения в пиктограммах этих блоков.

Аналогично откроем окно блока **Fcn** и в строке **Expression** (выражение) запишем выражение:

$$100 \cdot (u(1) + u(2) + u(3)).$$

Здесь $u(1)$, $u(2)$, $u(3)$ — компоненты вектора входного (для блока **Fcn**) сигнала, который формируется из входных сигналов блока **Mux** (нумеруются они сверху вниз). Множитель 100 — это $1/a_0$.

Не забывайте периодически сохранять сделанные изменения.

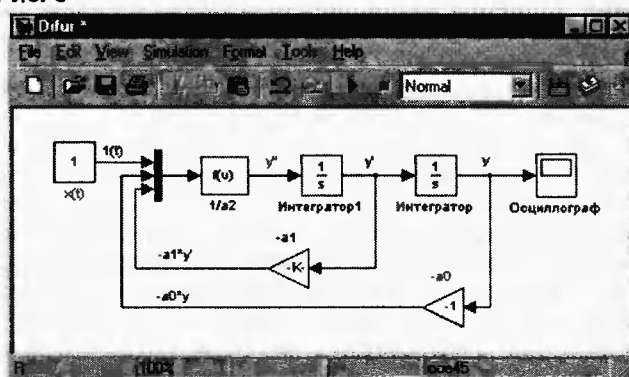
В блоках **Integrator** ничего не меняем, т.к. начальные условия нулевые.

В окне блока **Scope** можно установить масштаб по осям, время моделирования и другие параметры, используя панель инструментов окна. Однако в простых случаях это можно не делать, а подобрать их после моделирования и сохранить.

"Облагородим" рис.5 с целью его лучшего восприятия. Чтобы заменить название блока, надо щелкнуть ЛКМ на названии и в появившейся вокруг него рамке сделать исправления. Чтобы вообще убрать название, следует выделить блок, щелкнуть на нем ЛКМ и выбрать **Format – Hide Name** (спрятать имя). Для помещения надписи в произвольное место следует в предполагаемом центре надписи дважды щелкнуть ЛКМ и в открывшейся рамке поместить требуемый текст. Чтобы повернуть блок на 180° , следует дважды выполнить последовательность операций: выделить блок, щелкнуть на нем ЛКМ и выбрать **Format – Rotate block** (повернуть блок). Процедуру поворота выполним над блоками **Gain**. В итоге рис.5 может принять вид рис.6.

Теперь нужно установить параметры самой модели. Для этого в окне **Difur** в меню **Simulation** (моделирование) выберем пункт **Simulation Parameters...**. В открывшемся

Рис. 6





ся окне **Simulation Parameters** изменим только параметр **Stop Time** (время окончания моделирования) на 5 с (по умолчанию стояло 10) — решение произвольное, и опирается на интуитивное знание задачи. Затем нажмем **Enter** и выберем **Apply** — **OK**. Если это значение окажется неудачным, можно будет повторить моделирование при других значениях **Stop Time**. Среди параметров данного окна отметим раздел **Solver options** (опции решателя). В правой строке со скроллингом стоит имя решателя дифференциальных уравнений, который использует **Simulink** в данной задаче. По умолчанию установлен **ode45**, который реализует наиболее распространенный метод для решения обыкновенных дифференциальных уравнений — метод Рунге-Кутты четвертого порядка.

Рис. 7

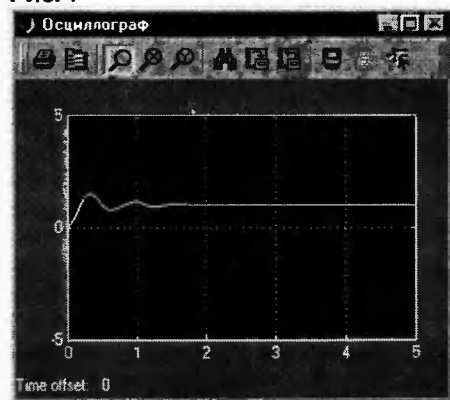


Рис. 8

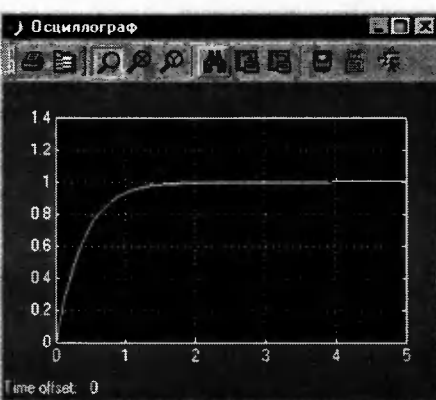
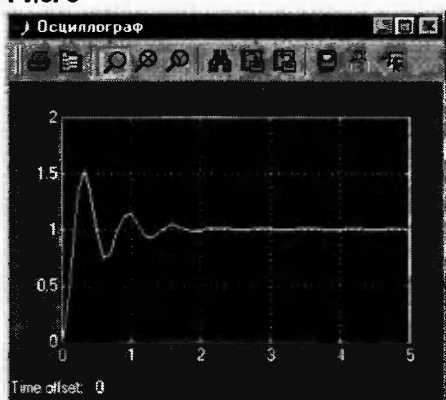


Рис. 9

Теперь можно начать моделирование. Для этого в окне **Difur** в меню **Simulation** следует выбрать команду **Start** или нажать кнопку **(Start simulation)** на панели инструментов окна. Если моделирование закончилось нормально, то посмотреть результаты можно щелкнув ЛКМ на блоке "Осциллограф". Откроется соответствующее окно (рис.7). Видно, что масштаб выбран не очень удачно. Для его изменения можно вызвать свойства окна, но проще начать с нажатия кнопки **Autoscale** (автошкалирование — "бинокль") на панели инструментов окна "Осциллограф". Получим картинку как на рис.8. Видно, что время моделирования можно уменьшить до 3 с.

Для изучения свойств уравнения (1) поменяем коэффициент a_1 с 0,04 на 0,4. Для этого щелкнем ЛКМ на блоке $-a_1$. В открывшемся окне изменим параметр и нажмем **Apply** — **OK**. Повторим моделирование, результат представлен на рис.9.

Таким образом можно исследовать некоторый процесс. Например, если данное уравнение описывает движение маятника в вязкой среде, то первое решение (рис.8) соответствует умеренной вязкости, а второе — сильной (рис.9).

Модель можно было построить иначе — это дело вкуса и конкретных потребностей. Например, в нашем случае можно убрать блоки **Gain**, а внутри блока **Fcn** записать выражение:

$$100 \cdot (u(1) - 0.04 \cdot u(3) - u(2)).$$

Можно все коэффициенты задать блоками **Constant**, что удобнее, если коэффициенты надо менять. Тогда схема может выглядеть как на рис.10, а внутри блока **Fcn** следует записать:

$$(u(1) - u(3) \cdot u(6) - u(2) \cdot u(5)) / u(4).$$

В заключение заметим, что **Simulink** имеет специальный редактор дифференциальных уравнений, с помощью кото-

рого можно непосредственно решать задачу Коши. Для вызова редактора следует в командном окне ввести **dee**.

Упражнения

Постройте модели решения предложенных ниже уравнений (более подробно они описаны в упражнениях к работе [4]).

1. Решите систему уравнений (попробуйте использовать разные решатели).

$$u' = 998u + 1998v$$

$$v' = -999u - 1999v$$

При $u(0) = v(0) = 1$.

2. Исследуйте модель Лотка-Вольтерра

$$\begin{cases} y_1' = Py_1 - py_1y_2 \\ y_2' = -Ry_2 + ry_1y_2 \end{cases}$$

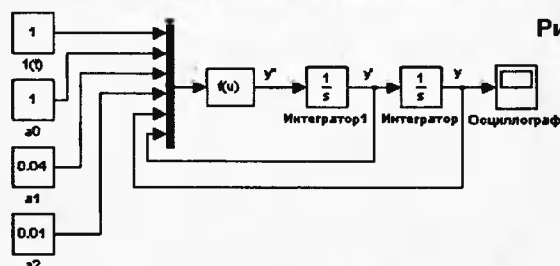


Рис. 10

Взять $P = 3$, $R = 2$, $p = r = 1$. Считать, что в начальный момент времени $y_1 = 3$, $y_2 = 4$. Решение отобразить на плоскости y_1, y_2 (для этого можно использовать регистратор **XY Graf**). Решить систему для $t = 100$. Попробуйте два решателя — **ode45** и **ode 23s**. Сравните результаты.

Вопросы для самопроверки

1. Для чего предназначен пакет **Simulink**?
2. В чем заключается технология построения модели в **Simulink**?
3. В чем заключается технология выбора параметров блоков в **Simulink**?
4. Как установить параметры моделирования?
5. Поясните результаты моделирования.

Литература

1. Киселев Б.М. Компьютерная математическая система **MATLAB**. — Радиомир. Ваш компьютер, 2003, №5, С.2.
2. Черных И.В. **Simulink**: Инструмент моделирования динамических систем. — <http://matlab.ru>, раздел "Simulink".
3. Гулятьев А.К. Визуальное моделирование в среде **Matlab**: Учебный курс. — Питер, 2000.
4. Киселев Б.М. **MATLAB**. Решение обыкновенных дифференциальных уравнений. — Радиомир. Ваш компьютер, 2005, №5, С.16.

С.РЮМИК,
г.Чернигов.

Интернет-калейдоскоп, freeware #22

Что толку в книжке,
если в ней нет ни картинок, ни разговоров.
Л.Кэррол. "Алиса в Стране Чудес"

Самыми распространенными в Интернете формами хранения книг являются текстовые *.txt- и текстово-графические *.djvu-файлы. Последний случай характерен для книг по электронике, когда сотня страниц с электрическими схемами вмещается в объем всего лишь 1 — 2 Мб. Плата за комфорт — повышенное время на распаковку и просмотр информации.

Неожиданность может подстергать при распечатке больших по объему djvu-книг. Иногда связка "компьютер — принтер" не справляется с обработкой сложных потоков данных. В этом случае помогает разбиение книги на отдельные странички

Комикс-битвы

Книги бывают разные — для прочтения, для просмотра, для того и другого одновременно. Например, комиксы (амер. comic book, comic strip, comics — "веселые картинки") — это красивая книжка с яркими рисунками и с выделяющимися вставками текста. Рождение комиксов относят к 1896 г., когда в газете New York World была опубликована серия необычных по стилю рисунков карикатуриста Ричарда Аутколта.

Психологи до сих пор не могут прийти к единому мнению, чего больше — вреда или пользы — приносят комиксы (<http://www.comics.com.ua/>

articles/article0006_1.shtml). Комиксы создаются не только для детей, но и для взрослых (детективы, фантазии <http://nazer-comics.narod.ru/html/translate.html>).

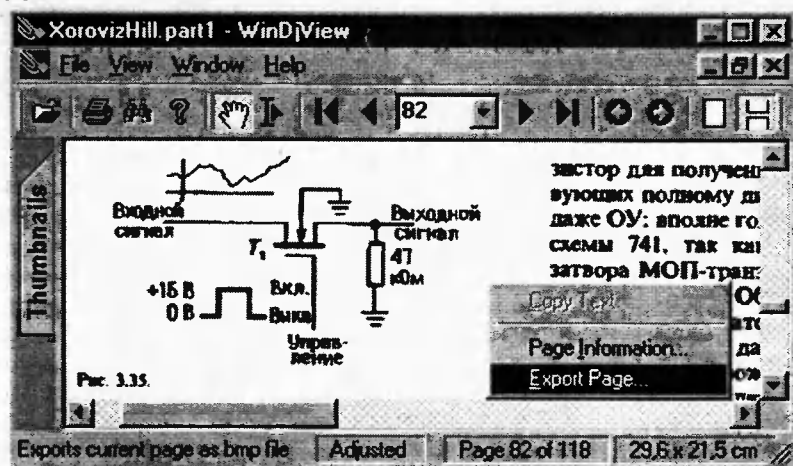
Весьма серьезно к комиксам относятся в Японии. Рисунки в стиле Манга здесь возведены в ранг культуры. На фирме Sakakibara Kikai даже сделали робота "а-ля из комикса" под названием "Land Walker", вес — 1 тонна, рост — 3 м, скорость — 1,5 км/ч (видеоролик http://www.sakakibara-kikai.co.jp/products/other/images/robo_s.wmv, 2,7 Мб).

Отечественные комиксы отличаются от американских и европейских собратьев. Чтобы убедиться в этом, достаточно посетить сайты, перечисленные на странице ссылок <http://nazer-comics.narod.ru/html/links.html>. В Укрнете, в частности, можно увидеть комикс-сериалы с национально-кулинарным колоритом (<http://www.comics.com.ua/>, рис.2).

Профессиональные художники проводят в Интернете спортивные соревнования по комиксам, называемые "комикс-битвами", с настоящей судейской коллегией, правилами, призами и зрительской аудиторией. Вот, например, как выглядел финальный поединок на чемпионате 2004—2005 гг.

Рис. 2

Рис. 1



ные файлы. В программе WinDjView (автор — А.Жежерун, <http://windjview.sourceforge.net/ru/index.html>, WinDjView-0.3.4.exe, 483 Кб) правой кнопкой мыши можно через опцию Export Page (рис.1) перевести текущую страницу из *.djvu в *.bmp-формат, а затем без проблем распечатать ее в любом текстовом редакторе.

Общая методика оформления рукописей и книг в виде файлов *.djvu разработана А.Лушниковым, http://www.bogoda3.nm.ru/how_make.rar, 9 Кб. Рекомендации, как сделать оцифровку книги при помощи фотоаппарата, дает К.Шубин на странице <http://chemister.da.ru/Other/Text/digital-photo.htm>.



(рис.3). Кстати, любой желающий может оставить свой критический (или наоборот) отзыв к каждому комиксу, в этом тоже есть определенный азарт.

Еще одно направление творчества — русификация текстов комиксов. На странице <http://spidermedia.ru/docs/articles/deirdre/translate.html> рассказано о двух технологиях этого процесса: прямая замена через графический редактор и "программистский подход" с коррекцией html-кода и формированием всплывающих переводов. Интересная идея для оформления своих программ.

Рис. 3

Чемпионат по комикс-битвам 2004-2005 > Финал. Максим Богдановский vs Владимир Сахнов			
Финал. Максим Богдановский vs Владимир Сахнов			
			
Ход №1. Максим Богдановский-553 просмотров 23 комментария	Ход №2. Владимир Сахнов-455 просмотров 12 комментариев	Ход №3. Максим Богдановский-466 просмотров 16 комментариев	Ход №4. Владимир Сахнов-450 просмотров 15 комментариев

Клуб авантюристов

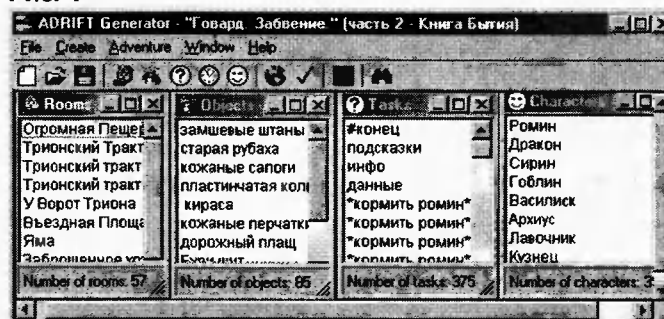
Под таким названием действует в Сети центр разработчиков текстовых адвентюр <http://advanturist.h1.ru/>. Основная философия клуба: "И одиночка может создать классную текстовую игру". Речь идет о "геймплее" — способности игры надолго удерживать внимание пользователя.

Разумеется, текстовые адвентюры (по-другому, Interactive Fiction, IF) не могут похвастаться необычными видеоэффектами. Но в них есть изюминка, заставляющая игрока думать, а не механически нажимать кнопки джойстика. "Лучший творец графики — это ваше воображение, а оно всегда с вами". Такую рекламу своим играм дала в 1975 г. фирма Infocom — основатель жанра.

Различают IF на русском и английском языках. Мнение о том, что, создавая англоязычную игру, можно научиться говорить по-английски, довольно спорно. Практика показывает, что без глубокого "погружения" в языковую среду здесь не обойтись. Иное дело — русскоязычные тексты, где легко уловить смысловые оттенки и поэкспериментировать в применении тех или иных слов.

Для программирования игр имеются инструменты: ADRIFT (<http://files.gamenavigator.ru/pub/games/adrift39.zip>, 2,8 Мб, рис.4) и RTADS (<http://advanturist.h1.ru/download.html>, 4 Мб). В качестве примера можно взять игру "Приключение" (А.Мирошников и др., <http://www.taplap.ru/articles/65/Adv.zip>, 2,3 Мб). По оформлению она напоминает "Звездное наследие" (ZX-SPECTRUM). Единственное, что следовало бы улучшить, это читаемость текста, который иногда сливается с фоном. Образец для подражания? Комиксы!

Рис. 4



SMS Quest

Владельцы мобильных телефонов могут не беспокоиться, речь пойдет о текстовой игре "СМС Квест" (автор — Е.Бычков, http://smsquest.by.ru/download/smsq_084.rar, 94 Кб). Чтобы поиграть в нее, надо скачать интерпретатор URQ (Universal Ripsoft Quest), ссылка на который и еще на десяток игр находится по адресу <http://smsquest.by.ru/quests.shtml>. Это одно из интересных направлений, где требуется приложить руки программистам.

На сайтах <http://indat.team-x.ru/>, <http://www.textquest.boom.ru/> размещены примеры текстовых квестов и РПГ на русском языке, а также их интерпретаторы — специальные программы для создания и просмотра игр.

Солидный проект текстово-графической РПГ под названием Time of Old Ages функционирует уже несколько лет (<http://www.timeold.ru/clicks.php?timeold.zip>, 592 Кб). Это русскоязычная РПГ на 5 персонажей, 40 локаций и 20 квестов, язык программирования — Visual Basic. Команда Ice Independent Developers Group приглашает к участию в проекте всех любителей жанра — от бета-тестеров до профессиональных кодеров и сценаристов.

Однако, как и в случае с самодельными текстовыми адвентюрами, цветное выделение надписей на рисунках оставляет желать лучшего (рис.5). Здесь также хорошо бы позаимствовать ти-

пичные приемы разрисовки комиксов.

Ключевые слова для поиска: djvu, комиксы, текстовые адвентюры, Interactive Fiction, TimeOld.

Рис. 5



Темная лошадка — BlackICE

Е. ЛЕБЕДЕВ,
г. Череповец.

В качестве вступления

В современном киберпространстве или просто Глобальной сети Интернет с каждым месяцем путешествие (серфинг) становится все опаснее и опаснее. Поэтому умные люди придумали разные средства защиты пользователей от всевозможных опасностей — начиная от вирусных атак и до вторжений хакеров.

Что представляет собой защита компьютера в сети? Что является "ахиллесовой пятой" компьютеров? И какими последствиями грозят Интернет-опасности?

Первой и главной опасностью можно смело считать вирусы. Незащищенный компьютер не продержится даже часа при непрерывном подключении к Сети.

Второй угрозой являются spyware-программы. Чаще всего — это зловерные рекламные скрипты, которые "защиты" в определенные сайты. С какой целью их создают? Только для того, чтобы напичкать рекламой чей-либо компьютер.

К третьей угрозе можно отнести атаки хакеров. Хотя этот вид угрозы, в отличие от вирусов, не так часто встречается на "территории" Интернета.

Если столько опасности в Сети, то как можно защититься от всей этой "нечисти"? Имеется, пусть и не 100%-я, но защита, временное "лекарство". К таким средствам можно отнести:

- сетевые экраны (фильтры) или firewall'ы (огненные стены);
- антирекламные модули (anti-spyware).

Темная лошадка

Сегодня мы рассмотрим одного из представителей сетевых экранов, т.к. они играют очень важную роль в построении сетевой защиты.

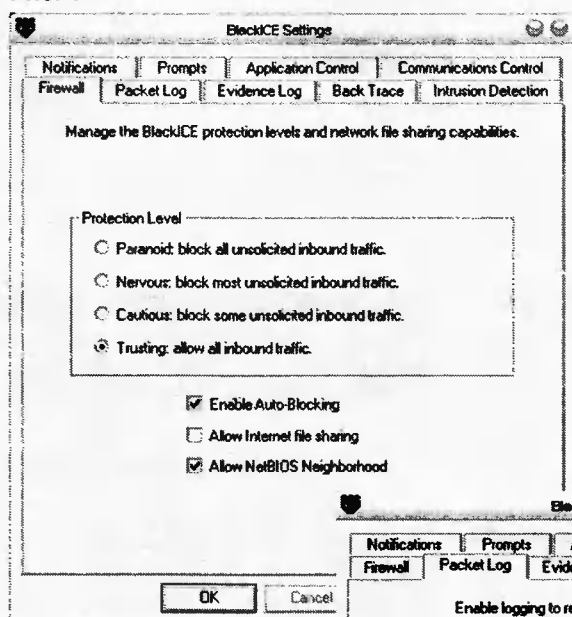
Множество бумажных и электронных изданий перебрали по косточкам известные сетевые экраны (в дальнейшем будем называть просто "фаерволы"). Среди особо замеченных были: Agnitum Outpost Firewall, Zone Alarm Pro, Kerio Personal Firewall, Norton Personal Firewall и наш отечественный продукт Kaspersky Anti-Hacker.

Да, эти продукты очень популярны, особенно Outpost Firewall. Но не стоит забывать и про малоизвестные продукты, которые ничуть не уступают раскрученным прессой маркам.

Сегодня речь пойдет о таком малоизвестном продукте от конторы

Internet Security Systems. Этот фаервол называется BlackICE PC Protection Version BlackICE. Ко мне в руки попала версия 3.6.cny.

Рис. 1



Первое впечатление

Дистрибутив версии 3.6 занимает 6 Мб и представляет собой стандартный "установочник". Я не стал спешить с запуском программы и просмотрел комплектацию фаервола. Среди всякого "конфигурационного добра" нашелся файл помощи — readme.txt. Он оказался бы полезным для опытных пользователей, но совершенно бесполезен для тех, кто впервые решил познакомиться с программой. В readme нет никакой информации о возможностях программы, и вообще — для чего она служит. Правда, разработчики указали системные требования к аппаратной части компьютера. Чуть ниже нас посылают "в баню" — дословно: "Документацию по компьютерной защите BlackICE можно найти по следующему адресу: <http://www.iss.net/support/documentation>".

Большим плюсом BlackICE является активное участие разработчиков в обновлении своего "детиса". В readme в каждой новой версии приводится список детекции новых типов атак и список обновлений в плане безопасности. Т.е.

наша "темная лошадка" имеет очень хороший потенциал для исполнения роли щита безопасности от интернет-атак.

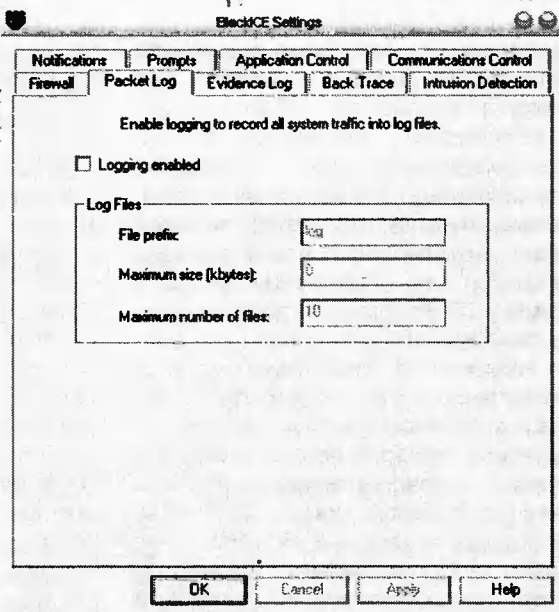
Перейдем к знакомству с интерфейсом программы. После запуска BlackICE, в "области часов" появляется иконка с глазом, которая символизирует то, что защита активирована. Если глаз перечеркнут красным, то защита находится в неактивном состоянии.

В BlackICE имеется два вида настроек — обычные и расширенные. Обычные настройки переносят нас на форму с кучей закладок.

Первая закладка — "Firewall" (рис.1). Здесь пользователю сразу же предлагают выбрать режим работы фаервола и некоторые моменты блокирования соединений.

В BlackICE создано 4 режима работы:

Рис. 2



- "Параноидальный" — блокирует весь входящий трафик, не отмеченный в правилах;
- "Нервный" — блокируется большая часть трафика;
- "Подозрительный" — блокируется некоторый подозрительный трафик;
- "Доверительный" — пропускается весь входящий трафик.

Доступны 3 опции блокировки:

- "Enable Auto-Blocking" — разрешить автоблокировку;
- "Allow Internet file sharing" — разрешить доступ к ресурсам из внешней сети;
- "Allow NetBIOS Neighborhood" — разрешить "Сетевое окружение".

Переходим на следующую закладку "Packet Logging" (рис.2):

Опция "Logging enabled" включает режим записи в журнал всех входящих и исходящих пакетов, т.е. всего сетевого трафика. Для логов можно задать префикс файла, максимальный размер и максимальное количество файлов.

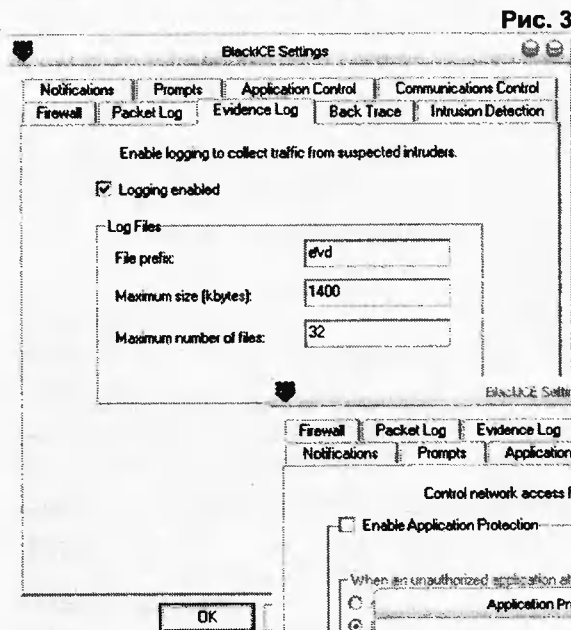


Рис. 3

Переходим на следующую закладку "Evidence Log" (дословно — Журнал доказательств) (рис.3).

BlackICE включает в себя прекраснейшую функцию — он умеет записывать всю возможную информацию по удаленному атакующему, что напрочь отсутствует в популярных фаерволах.

Наш "подопытный" также включает в себя контроль за приложениями и за соединениями.

Закладка "Communications Control" отвечает за контроль соединений (рис.4).

Если включить данную опцию, то сразу же включится режим сканирования системы на наличие приложений. Но перед этим будет выдано сообщение, как показано на рисунке: "BlackICE создаст базовый список приложений на Вашем компьютере. Ваша система будет занята некоторое время, пока будет осуществляться процесс сканирования". В целом сканирование длится 15 минут, хотя длительность процесса зависит от аппаратных возможно-

стей вашего компьютера и количества установленных приложений.

Фаервол предлагает четыре возможных варианта действий в случае, когда неавторизованное приложение пытается получить доступ в сеть:

- всегда закрывать такие приложения;

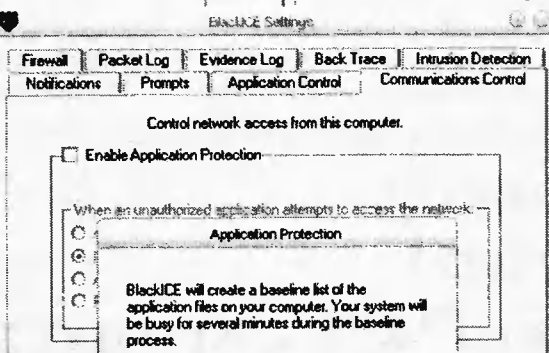
- спросить пользователя перед закрытием приложения;

- всегда блокировать сетевой доступ для приложения;

- спросить пользователя перед ограничением доступа приложению в сеть.

Теперь рассмотрим что такое "Контроль приложений" (рис.5).

Рис. 4



ратным, т.к. можно заблокировать доступ в сеть совершенно безобидным приложениям, которыми вы пользуетесь.

Функция контроля приложений включает в себя реакции на два основных события:

- "When an unknown application launches" — когда запущено неизвестное приложение;

- "When a modified application launches" — когда одно из авторизованных приложений модифицировано и запускается.

Каждому из этих событий можно назначить соответствующую реакцию фаервола:

- "Ask me what to do" — спросит меня, что делать;

- "Always terminate the application" — всегда закрывать приложение.

Стоит отметить один момент — опции контроля приложений и соединений взаимосвязаны. Если отключить опцию контроля приложений, то автоматически отключится контроль соединений для приложений!

Далее следует настройка правил для фаервола (рис.6), что является стандартом для подобного класса программ.

Рис. 5

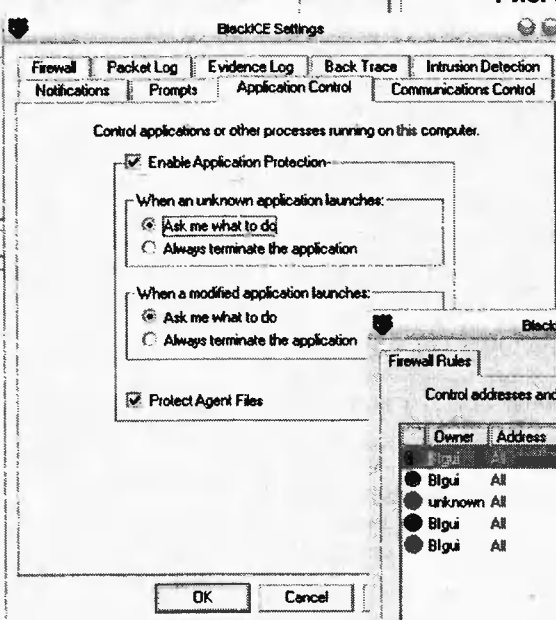
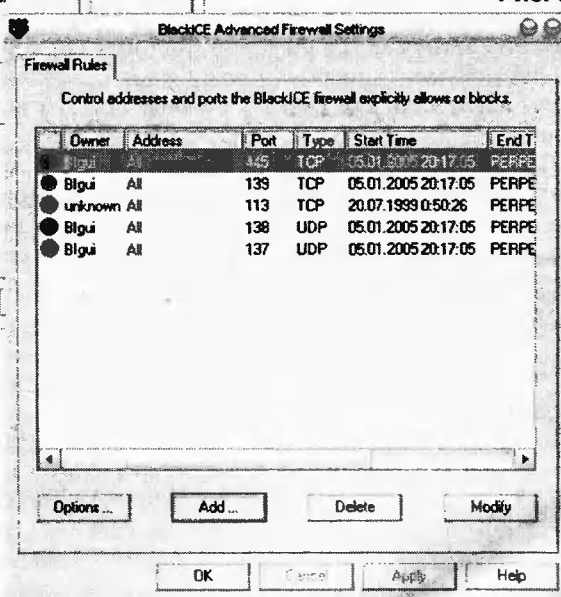


Таблица правил:

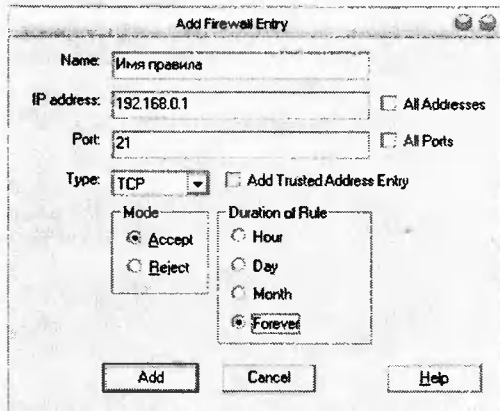
- Owner — приложение;
- Address — адреса;
- Port — порт;
- Type — тип протокола;
- Start Time — время запуска;
- End Time — время окончания;
- Name — название приложения.

Рис. 6



Контроль приложений позволяет отслеживать запуск подозрительных приложений в системе. Опция очень полезна для обнаружения троянов и подобных вредоносных программ. С настройкой данной функции следует быть очень акку-

Рис. 7



Кнопкой "Add" добавляется новое правило (рис.7). Здесь вводим имя для правила, IP-адрес, к которому это правило относится, конкретный порт (если это необходимо), тип протокола, режим (Разрешить или Запретить) и "время жизни правила" — это функция, которая есть только у BlackICE. Популярные фаерволы такой функции не имеют!

Скачки, скачки

Чтобы убедиться в описанных выше возможностях, проверим фаервол в деле. Для тестирования использовался гостевой доступ к провайдеру. Мы хотим получить доступ к IRC-серверу. Запускаем популярный клиент для IRC-сетей — mIRC. Ставлю "параноидальный" режим — запускаю mirc. Фаервол молчит — спокойно пропускает пакеты. Естественно, т.к. mirc зарегистрирован у него в базе. Копируем mirc (всю папку) в другое место и запускаем. BlackICE сразу же "моргает глазом" и сообщает, что обнаружено незарегистрированное приложение, которое пытаются запустить. Разрешаем запуск. Далее в mirc'e пытаемся соединиться с любым IRC-сервером. Получаем сообщение, что mirc пытается соединиться с удаленным сервером. Разреша-

ем или запрещаем соединение программе.

Пробуем блокировку IRC-порта 6667. Заходим из контекстного меню через пункт "Advanced firewall settings" в раздел настроек блокировки-разрешения пакетов. Добавляем правило:

- <Все IP-адреса>;
- Порт: 6667;
- Reject (Блокировать);
- Forever — Вечное правило;

Пробуем соединиться с удаленным сервером. Системное окно mirc-клиента выдает событие: "could not connect (permission denied)". Т.е. соединение было запрещено.

Послесловие

Очень грамотный и профессиональный фаервол. Начинающих пользователей может слегка испугать англоязычный (что успешно "лечится" русификатором) и строгий (это дело привычки) интерфейс.

Автоматизация открытия документов в браузере Opera с помощью Perl-скрипта

И.РОЩИН,

Москва.

ZXNet: 500:95/462.53

E-mail: bestview@mtu-net.ru

WWW: http://www.ivr.da.ru

Как-то мне надо было решить такую задачу: обеспечить, чтобы в указанное время автоматически запускался браузер (Opera 5.12), и в нем открывались окна с указанными документами — так, чтобы после открытия каждого окна была пауза определенной продолжительности. Последнее обусловлено тем, что при одновременном открытии нескольких документов они загружаются параллельно, так что скорость загрузки каждого из них невелика (у меня модем на 14400 бит/с). Как я не раз замечал, при этом некоторые документы так и не загружаются до конца (видимо, сервер, отдающий документ, досрочно разрывает соединение). А если загружать документы по очереди, то все нормально.

Для решения этой задачи я написал простой скрипт на Perl. Возможно, он пригодится и вам для решения такой же или похожей задачи. Конечно, нужно будет внести в скрипт соответствующие изменения, если у вас ситуация не точь-в-точь такая же, как у меня.

Текст скрипта приведен ниже.

```
#!/usr/bin/perl

$run_time_h='12'; # Нужное время запуска (часы).
$run_time_m='00'; # Нужное время запуска (минуты).

@pages=( # Список адресов открываемых страниц.
'http://example1/',
'http://example2/',
'http://example3/'
);
```

Проверка: в адресах не должно быть символа "'", так как этим # символом адреса ограничиваются, когда передаются 'Опере' # в командной строке.

```
foreach $page (@pages)
{
    if ($page =~ "'") { die ("Error: found '\"' in \"$page\"!\n"); }
}
```

Ждем указанного времени запуска.

```
print ("Waiting run time ($run_time_h:$run_time_m)...\n");
while (@time=localtime,
    $time[2]*60+$time[1] != $run_time_h*60+$run_time_m)
{
    sleep (20); # Интервал (в секундах) между проверками времени.
}
```

Запускаем "Оперу" и ждем 60 секунд, чтобы она успела загрузиться.

```
print (scalar(localtime). " Starting \"Opera\"...\n");
system ("start \"C:\Program Files\Opera\Opera.exe\"");
print ("Pause...\n");
sleep (60);
```

По очереди открываем перечисленные в списке @pages страницы. # После передачи "Опере" адреса страницы ждем 45 секунд.

```
foreach $page (@pages)
{
    print (scalar(localtime). " Opening page \"$page\"...\n");

    $page =~ s/'/%'/g; # Удваиваем "'" в адресе, чтобы командный
# интерпретатор Windows считал этот символ
# обычным символом.
```



```
system ('start "C:\Program Files\Opera\Opera.exe" "" .%page.%');
print ("Pause...\n");
sleep (45);
}
```

Время запуска "Оперы" задается в переменных \$run_time_h и \$run_time_m (соответственно, часы и минуты). Адреса открываемых документов задаются в списке @pages. Само собой, адресов может быть не три, как для примера приведено в скрипте, а столько, сколько нужно. Адреса указываются в одиночных кавычках и перечисляются через запятую.

Если адрес длинный, его можно разделить на части удобной длины и записать каждую часть в отдельной строке, заключив в одиночные кавычки; между частями должна стоять точка (в Perl это оператор склейки строк). Вот пример:

```
'адрес 1',
'адрес 2 (часть 1)',
'адрес 2 (часть 2)',
'адрес 2 (часть 3)',
'адрес 3'
```

Адреса не должны содержать одиночных кавычек (так как сам адрес ограничивается ими) и двойных кавычек (они используются как ограничители при передаче адреса "Опере"). Вообще-то кавычки и так не входят в список символов, допустимых в web-адресах, но мало ли, вдруг вы, скажем, скопируете из какого-либо текста адрес, ограниченный кавычками, и случайно вместе с адресом скопируете кавычки.

Если вдруг в адресе попадет одиночная кавычка, то, скорее всего, при запуске скрипта Perl выдаст сообщение об ошибке. Отсутствие двойных кавычек в адресах проверяется непосредственно в скрипте; если они будут обнаружены, выполнение скрипта прекратится с выдачей соответствующего сообщения.

После запуска скрипт выводит строку с указанием времени, когда он должен начать работу, и ждет наступления этого момента. Когда он наступит, происходит запуск "Оперы" (обратите внимание: в скрипте указан полный путь к исполняемому файлу "Оперы"; если у вас он не такой, то надо указать в скрипте свой путь). Затем следует 60-секундная пауза, чтобы "Опера" успела загрузиться (продолжительность этой паузы при необходимости можно изменить).

Далее в цикле по очереди открываются окна с документами. Чтобы открыть окно с некоторым документом в браузере Opera 5.12, достаточно просто запустить этот браузер, указав в командной строке адрес этого документа. Именно это действие и выполняется в скрипте. При этом не будет запущен новый экземпляр браузера, а в уже запущенном откроется новое окно с указанным документом. Не знаю, так же или нет ведут себя другие браузеры. После открытия окна следует 45-секундная пауза (ее продолжительность также можно изменить).

В процессе работы скрипт выводит информацию о том, когда и что он делает, как в нижеприведенном примере.

```
Waiting run time (12:00)...
Mon Apr 25 12:00:11 2005 Starting "Opera"...
Pause...
Mon Apr 25 12:01:15 2005 Opening page "http://example1/"...
Pause...
Mon Apr 25 12:02:02 2005 Opening page "http://example2/"...
Pause...
```

```
Mon Apr 25 12:02:50 2005 Opening page "http://example3/"...
Pause...
```

Если вы решите использовать описанный скрипт (возможно, внося в него какие-либо изменения), то рекомендую сначала протестировать его (установив в нем время запуска, скажем, через минуту после текущего времени). Плохо, если скрипт будет запущен для решения серьезной задачи, а потом вдруг окажется, что он не сработал так, как нужно.

Осталось рассказать о некоторых проблемах, обнаруженных в процессе регулярного использования данного скрипта (под Windows 95 OSR 2.1 RUS; версия Perl — ActivePerl 5.8.6.811).

Во-первых, если какой-либо из адресов, указанных в списке, слишком длинный, то "Опера" не открывает окно с соответствующим документом (открывается пустое окно). Видимо, это связано с ограничением на длину командной строки в Windows. Экспериментально было установлено, что не открываются документы с адресами длиннее примерно 240 символов.

Во-вторых, если при запуске скрипта дописать в командной строке ">имя_файла" (или поместить в начале скрипта команду "open (STDOUT, ">имя_файла");"), чтобы перенаправить вывод скрипта в файл с указанным именем, то при открытии некоторых документов, в тот момент, когда скрипт запускает "Оперу", передавая ей в командной строке адрес документа, командный интерпретатор Windows почему-то выдает сообщение:

Каталог поиска для COMMAND задан неверно
Слишком много параметров

После этого командный интерпретатор не возвращает управление скрипту, так что все следующие документы также не будут открыты. Эта проблема возникает, по моим наблюдениям, когда адрес документа содержит символы "%" (хотя, быть может, это лишь совпадение). Если отказаться от перенаправления вывода, то этот же документ открывается нормально.

В-третьих, если запустить скрипт с дискеты, то, когда наступает указанное время и скрипт запускает "Оперу", почему-то происходит обращение к дискете, и если дискета отсутствует (например, вынута после запуска скрипта), то "Опера" не запустится.

В-четвертых, если предыдущий сеанс работы с "Оперой" был завершен некорректно (например, нажатием Reset), то при запуске "Опера" выводит окно с сообщением: "Предыдущий сеанс работы Opera был прерван" — и ждет, что пользователь выберет один из трех вариантов продолжения работы ("Продолжить с места разъединения", "Загрузить сохраненные окна при запуске", "Не открывать окна при запуске"). А если пользователя нет (а ведь скрипт как раз и предназначен для открытия документов в автоматическом режиме, без участия пользователя), то, естественно, "Опера" так и будет ждать, и открытие документов не произойдет.

В-пятых, при открытии большого числа документов, особенно содержащих изображения, может не хватить памяти. Тогда вполне вероятно появление окна с сообщением: "Программа выполнила недопустимую операцию и будет закрыта". Причем, возможно, вы даже увидите в окне "Оперы", что большая часть документов загружена — но, увы, недоступна... Хотя, может, что-то и удастся извлечь из кэша.



ЛИСТАЯ СТРАНИЦЫ

Хакерству с помощью поисковой системы Google посвящена статья "Шпионские страсти" (СНП, 5/2005, С.100...103).

Оказывается, один из самых опасных сайтов в Сети — это Google.com. Не всем известен тот факт, что Google индексирует все без исключения файлы, которые лежат в незащищенном доступе, начиная от таких незамысловатых данных как пароли и заканчивая строго секретными документами. Винить в этом нужно в первую очередь самих владельцев сайтов, которые размещают незащищенную информацию, совершенно не задумываясь о возможных негативных последствиях. Взлом с помощью Google — это новый способ, который используют хакеры для проникновения в бережно хранимые корпоративные секреты. Им даже не нужно прибегать к использованию специальных утилит — анализаторов протоколов (password sniffers) или сканеров портов.

К какой же информации можно получить доступ, используя Google? Это могут быть секретные презентации в PowerPoint, резервные копии электронной почты, отправленные письма, а также огромное количество паролей и номеров пластиковых карт. При этом ни разу не придется проходить аутентификацию для просмотра таких данных. Достаточно всего пары простых, но эффективных приемов, чтобы Google преподнес желаемый результат на блюде с голубой каемочкой.

В чем секрет? Дело в том, что поисковая система Google использует много тысяч ботов, которые непрерывно и, надо заметить, на абсолютно законных основаниях бороздят просторы Интернета, просматривая всю находящуюся там информацию. Вдобавок к этому, Google по-

лучает списки ссылок от своих партнеров, например Орега. Бесплатная версия браузера, окупаемая исключительно за счет баннерной рекламы, отправляет Google перечень всех сайтов, которые посещает пользователь. Поисковая система классифицирует их согласно категориям, и, в итоге, пользователь видит в браузере Орега специально отобранные рекламные баннеры, которые, как предполагается, отвечают его интересам. Между тем, по адресу, который пользователь набирает в строке, направляется еще один бот, чья задача заключается в том, чтобы внести эту ссылку в базу данных. Однако некоторым сайтам, которые числятся в базе данных Google, лучше было бы остаться незамеченными.

Если правильно сформулировать запрос поиска, то Google в считанные секунды может снабдить всеми необходимыми файлами с конфиденциальным содержанием. Для этого понадобится знать только волшебное заклинание, состоящее из ключевых слов для поиска и нужного оператора.

В последнее время новые стратегические планы многих ведущих компаний преподносятся как презентации, созданные в программе PowerPoint, как известно, такие файлы имеют расширение PPT. Для того чтобы было понятно, что информация предназначена для узкого круга адресатов, каждый слайд обычно снабжается пометкой "Конфиденциально" или "Для служебного пользования". Если в обычную строку поиска Google ввести следующий запрос:

ext:ppt confidential "for internal use only" то меньше чем через одну секунду Google сформирует список из нескольких тысяч различных файлов, которые в точности соответствуют заданным критериям.

С помощью операторов Google можно найти все, что нужно:

- allintitle: — ограничивает результаты поиска только теми страницами, в строке заголовка которых встречаются все параметры поискового запроса;

- intitle: — в отличие от предыдущего оператора, в строке заголовка должно быть найдено только первое слово, указанное в запросе. Остальные слова поиска ищутся в тексте страницы;

- allinurl: — используется, если все ключевые слова поиска должны встретиться в адресе;

- inurl: — аналог оператора "intitle:", в строке адреса ищет только первое слово запроса;

- .. (numrange) — при поиске чисел можно сформулировать область значений. Например, если определить запрос как "100..150", то будут найдены все страницы, содержащие числа от 100 до 150;

- daterange: — с помощью этого оператора можно ограничить количество результатов определенным периодом времени. Следует иметь в виду, что параметры нужно задавать согласно юлианскому календарю;

- ext: — целенаправленно ищет ссылки на какой-либо файловый формат. В качестве альтернативы можно попробовать и "filetype:";

- cache: — загружает найденный веб-сайт из кеша Google. Это удобно в том случае, если нужный сервер больше не существует;

- site: — запускает поиск на каком-либо определенном веб-сайте;

- related: — при использовании этого оператора Google покажет в результатах поиска похожие сайты;

- info: — получение краткого описания сайта;

- link: — набрав перед запросом этот оператор, можно найти все сайты, ссылающиеся на указанную страницу.

Результаты тестирования 14 моделей жидкокристаллических мониторов представил Д.Ерохин в статье "17 дюймов жидких кристаллов" (Подводная лодка, 5/2005, С.72...81).

Несмотря на достаточно долгий (по компьютерным меркам) срок развития, технология производства ЖК-дисплеев так и не смогла пока достичь того уровня, когда можно без скидок говорить об их паритете с ЭЛТ по качеству вывода. Для офисных приложений (текст, графики, диаграммы и т.п.) ЖК-модели с их идеальной четкой и яркой картинкой и отсутствием мерцания подходят как нельзя лучше. Но как только дело касается сколько-нибудь серьезной работы с графикой, тут и обнаруживается, что ЭЛТ-аппараты еще рано сдавать в утиль. Мало того, что по своей природе ЖК-дисплеи неспособны передавать глубокий черный цвет, так и с цветопередачей дела у них обстоят, мягко говоря, не идеально. Особенно это справедливо для недорогих моделей с быстрыми матрицами

типа TN (twisted nematic). Заявленное в документации значение 16 млн. отображаемых цветов на деле зачастую оказывается звывшенным более чем в... 60 (!) раз и составляет 262 144 оттенка (на каждый из трех цветовых каналов приходится не по 8, а по 6 битов). Недостающие цвета компенсируются путем эмуляции (поочередным выводом соседних цветов с частотой кадровой развертки и сглаживанием). Справедливости ради нужно заметить, что у лучших моделей ЖК-мониторов меха-

Модель	Acer AL1715ms	AG Neovo E-17	BenQ FP71E+
Штатное разрешение	1028x1024	1028x1024	1028x1024
Размер пиксела, мм	0,264	0,264	н/д
Число отображаемых цветов, млн.	16,2	16,7	16,2
Яркость, кд/м²	270	300	300
Контрастность	450:1	400:1	500:1
Заявленное полное время отклика, мс	12	16	8
Углы обзора по горизонтали, градусов	75/75	140	140
Углы обзора по вертикали, градусов	75/60	130	130
Цифровой ввод DVI	-	-	+
Диапазон строчной развертки, кГц	30-85	24-80	31-83
Диапазон кадровой развертки, Гц	56-75	49-75	56-76
Полоса пропускания, МГц	135	135	135
Мощность встроенных динамиков, Вт	2x1	н/д	2x1
Встроенный блок питания	+	-	+
Потребляемая мощность, работа/ожидание, Вт	40/2	48/3	40/1



Табл. 1

Модель	Уровень серого							
	32	64	96	128	160	192	224	255
	Время отклика пикселей, мс							
Acer AL1715ms	17/2/19	19/2/21	22/2/24	24/2/26	25/2/27	24/2/26	15/3/18	12/4/16
AG Neovo E-17	20/1/21	25/2/27	29/2/31	29/2/31	24/2/26	23/3/26	14/3/17	14/3/17
BenQ FP71E+	16/2/18	18/2/20	25/2/27	26/2/28	26/2/28	25/2/27	21/3/24	12/3/15
CTX F773	18/2/20	20/2/22	21/2/23	22/2/24	29/2/31	29/2/31	27/2/29	21/3/24
GreenWood LC724FT	19/2/21	23/2/25	27/2/29	25/2/27	24/2/26	20/3/23	9/3/12	9/3/12
LG Flatron L1740P	18/2/20	20/2/22	32/2/34	30/2/32	28/2/30	25/2/27	22/3/25	17/3/20
NEC AccuSync LCD72VM	16/3/19	17/3/20	17/3/20	17/3/20	20/3/23	11/4/15	10/5/15	10/5/15
NEC AccuSync LCD72XM	17/2/19	20/2/22	24/2/26	25/2/27	24/2/26	19/3/22	9/3/12	9/3/12
OFT 1703A	17/2/19	19/2/21	24/2/26	25/2/27	25/2/27	14/3/17	10/3/13	10/3/13
Philips Brilliance 170X5	17/3/20	18/2/20	22/2/24	23/3/26	22/3/25	10/4/14	10/4/14	10/4/14
RoverScan Futura 172	17/2/19	18/2/20	18/2/20	21/2/23	20/2/22	9/3/12	9/3/12	9/3/12
Samsung SyncMaster 720B	16/2/18	18/2/20	21/2/23	22/2/24	23/2/25	25/2/27	20/3/23	15/4/19
Sony SDM-HS75P	18/4/22	20/3/23	23/3/26	25/3/28	21/4/25	18/7/25	7/10/17	6/11/17
ViewSonic VX750	20/2/22	21/2/23	22/2/24	22/2/24	21/3/24	21/3/24	12/4/16	12/4/16

низм эмуляции реализован весьма хорошо, однако их не так уж много. Остальные же аппараты нетрудно "вывести на чистую воду", заставив отобразить градиент, на котором в большей или меньшей степени (в зависимости от настроек яркости и контрастности) будет заметна "полосатость".

В среде заядлых игроков, для которых важна быстрая прорисовка динамичных сцен и хорошая передача темных тонов, ЖК-мониторы не котируются. Кстати сказать, отчасти именно в угоду игроманам рынок массовых ЖК-мониторов практически на 100% оказался заполненным моделями на основе матрицы типа TN с наихудшей цветопередачей, но зато сравнительно быстрой реакцией пикселей. И это понятно, ведь гораздо проще продать аппарат с большим (или, в случае с ЖК-мониторами, меньшим) числом "попугаев", чем объяснять покупателю вещи, которые затруднительно выразить числами. В случае с ЖК-мониторами в

качестве "попугаев" выступают миллисекунды, в которых измеряется время отклика пикселей или, иными словами, быстроедействие матрицы. Почему "попугаи"? Да потому, что указываемое в документации время отклика лишь косвенно соотносится с реальной жизнью (не говоря уже о том, что оно порой просто не соответствует действительности). Дело в том, что паспортное время отклика — это полное время, необходимое для изменения яркости пикселя с черного на белый (от 10 до 90%) и обратно. На практике же чаще встречаются не пограничные (черно-бело-черные) переходы, а промежуточные (например, с черного на серый). Таким образом, скорость переключения пикселя от черного до белого не может характеризовать реальную ситуацию при работе пикселя в заданном динамическом диапазоне "яркостей", так как при переходе от темного до серого (не максимально яркого) состояния имеют место два противодействующих фактора

времени гашения, третье — полное время отклика).

Очевидно, что при выборе ЖК-монитора предпочтение следует отдавать той модели, которая вам подходит по совокупности тех или иных параметров, ведь совершенство недостижимо. Технические характеристики протестированных мониторов сведены в табл. 2.

Выбор ЖК-монитора — задача не из простых, потому что на данном этапе развития эти устройства еще не стали настолько совершенными, дабы идеальнейшим образом удовлетворять запросам всех пользователей. Если нужно, чтобы монитор достойно справлялся с выводом динамичных сцен, то, как правило, приходится жертвовать качеством цветопередачи, ограничиваясь только моделями, выполненными на базе матриц типа TN+Film. Ведь именно эти матрицы имеют самое маленькое время отклика пикселей (недавно объявлены изделия с минимальным временем откли-

Табл. 2

CTX F773	GreenWood LC724FT	LG Flatron L1740P	NEC AccuSync LCD72VM/XM	OFT 1703A	Philips Brilliance 170X5	RoverScan Futura 172	Samsung SyncMaster 720B	Sony SDM-HS75P	ViewSonic VX750
1028x1024	1028x1024	1028x1024	1028x1024	1028x1024	1028x1024	1028x1024	1028x1024	1028x1024	1028x1024
0,264	0,264	0,264	0,264	0,264	0,264	0,264	0,264	н/д	0,264
16	16,7	16,2	16,2	16,7	16	16,2	16,2	н/д	н/д
400	400	300	250	300	250	260	300	420	270
450:1	500:1	700:1	450:1	500:1	450:1	450:1	600:1	600:1	450:1
16	14	12	16	16	16	16	12	8	16
н/д	160	160	70/70	80/60	80/80	150	160	160	н/д
н/д	140	160	65/60	70/70	70/70	140	160	н/д	н/д
+	-	+	-/+	-	-	+	-	+	+
30-80	22-82	30-83	31-81	30-81	30-82	24-80	30-81	28-80	30-82
59-75	50-76	56-75	56-75	55-75	56-76	49-75	56-75	48-75	50-75
140	140	н/д	140	133	135	140	135	н/д	140
2x1	2,5x1	-	2x1	н/д	2x2	2x1	2x1,5	-	н/д
+	-	+	+	-	+	-	-	+	-
39/2	н/д	43/1	34/н/д	40/3	45/1	48/3	34/2	45/1	н/д/1



ка 4 мс, что уже совсем близко к воздействию мониторов на ЭЛТ). Напротив, если монитор предполагается использовать для более или менее серьезной работы с графикой, то зачастую приходится приносить в жертву скорость

реакции пикселей, выбирая модель с матрицей, отличной от TN+Film (PVA, MVA или IPS). Впрочем, надо заметить, что прогресс не стоит на месте, и сегодня стали доступны сравнительно быстрые матрицы не только TN-типа. К со-

жалению, поскольку TN-матрицы не только самые быстрые, но и самые дешевые в производстве, на настоящий момент практически нереально найти 17-дюймовый монитор с матрицей какого-либо иного типа.



Что такое морфинг, и какие программы используются для получения этого видеоэффекта, рассказывает А. Прохоров в статье "Программы по морфингу на любой вкус" (КомпьютерПресс, 5/2005, С.172...175).

Морфинг — это видеоэффект, заключающийся в плавном перетекании одного изображения в другое. Иногда морфингом называют эффект наплыва, когда одна сцена затемняется или затуманивается, а вторая в это время проявляется все резче и сменяет первую. Однако лучше разделять эти понятия. В отличие от наплыва, морфинг обеспечивает соответствие характерных (наиболее привлекающих внимание) точек и контуров начального и конечного изображений. Например, преобразование одного лица в другое, при котором задается соответствие положения характерных точек глаз, носа и т.д. Существует целый ряд программ для морфинга на ПК, позволяющих быстро создавать интересные эффекты. За несколько минут можно сделать видеоролик, превращающий ребенка в пожилого человека, одно животное в другое, изменить марку автомобиля и т.д. Морфинг-роликом можно украсить скрин-сейверы, web-страницы, рекламные и музыкальные клипы, видеофильмы, образовательные ролики, презентации, поздравительные открытки и т.п.

Основные термины и определения

Single morph (одиночный морфинг) — морфинг исходного изображения (source image) в итоговое (target image). Обычная программа для морфинга позволяет сгенерировать либо видеоролик (плавное перетекание одного изображения в другое), либо серию промежуточных изображений.

Photo to photo-morphing (фото в фото-морфинг) — морфинг исходного изображения в результирующее, при котором результатом является серия статичных изображений, иллюстрирующих перетекание, или одно изображение, имеющее по 50% от двух исходных. Например, при морфинге изображений двух братьев можно получить изображение их виртуального третьего брата, который будет иметь по 50% черт каждого из братьев, или при морфинге изображения мужчины в изображение женщины — изображение их виртуального ребенка. Для того чтобы создать реалистичный морфинг-проект, необходимо подобрать изображения и подготовить их с помощью фоторедактора. Некоторые морфинг-программы имеют встроенные функции фоторедактирования, другие предполагают, что подготовительная стадия выполняется во внешнем приложении, например в программе Photoshop. Обычно сначала необходимо отредактировать фон,

размеры изображений, освещение, резкость и т.д.

Photo to video-morphing (фото в видео) — морфинг исходного изображения в результирующее, при котором результатом является видеоролик.

Video to video morphing (видео в видео) — морфинг на базе двух видеороликов.

Multiple morph — множественный морфинг, то есть морфинг серии превращений. Еще более сложным, по сравнению с множественным морфингом лиц людей разных рас, является множественный морфинг мало похожих объектов. Например, плавное преобразование человека в различных животных. Для создания столь сложного множественного морфинга важно иметь общую визуальную тему — задний фон, какой-либо неизменный элемент, присутствующий на всех кадрах. Чем больше похожи фотографии по композиции (одинаковый центр композиции), тем более убедительным будет результат. Мозг человека устроен так, что в большей степени отслеживает плавность перехода в точке, которая концентрирует его внимание, а плавность морфинга на удалении от этой точки менее критична.

Anchor points, control points, key dots (якорные точки, контрольные точки, реперные точки, ключевые точки, точки соответствия) — пары точек, определяющие соответствие между указанными пикселями исходного и конечного изображений. В ряде программ, кроме точек, используются линии. Пользователи должны расставить точки и провести линии (контуры) на исходном изображении и задать их перемещение на конечном изображении. Необходимо расставить точки на исходной картинке и затем указать для них новое положение на итоговой. Чем больше точек и линий будет задано, тем более плавным и естественным будет видеоряд морфинга.

Real time playing (проигрывание в реальном времени) — функция позволяет просмотреть морфинг-эффект во время редактирования до экспорта в результирующий файл. Данная функция дает возможность отработать параметры проекта и только после этого экспортировать его в файл. Она особенно важна для профессионалов, которые часто прибегают к задачам морфинга и которым требуется высокая производительность работы.

Real time preview (мгновенный предпросмотр) — функция позволяет увидеть, как редактирование проекта влияет на результат.

Morph 2 + Images — функция дает возможность объединить несколько проектов по морфингу в непрерывный морфинг-проект. Для каждого нового морфинга предыдущий итоговый кадр (target image) явля-

ется начальным (source image).

Warp morphs (морфинг-искажение) — большинство программ для морфинга предлагают эффекты искажения фотографий, похожие на те, с которыми сталкиваются посетители зала кривых зеркал. Здесь нет единой терминологии — производители могут называть одни и те же эффекты по-разному. Однако чаще всего эффект Warp morphs подразумевает, что можно вручную деформировать часть изображения, создавая иллюзию движения. С помощью якорных точек нужно обозначить область, которую должен занять деформированный объект (например, можно задать анимацию, при которой у человека растет нос или меняется форма головы, вытягиваясь в виде конуса, и т.п.).

Deform/distortion morphs (морфинг-деформирование) — обычно такие формы морфинга задаются автоматически: пользователь выбирает из меню эффект и применяет его или ко всему изображению, или локально с помощью мыши (щелчок или перетаскивание).

Mask morphing (морфинг с маской) — маска изолирует часть изображения, делая ее неизменяемой. Mask morphing-инструменты полезны, если необходимо анимировать только часть изображения.

Layered morphing (послойный морфинг) — каждый слой изменяется отдельно, а затем слои совмещаются, и зритель видит суммируемый эффект.

Auto reverse morphing (авторевверс-морфинг) — морфинг осуществляется от исходного изображения до конечного и обратно.

Auto loop morphing (циклический морфинг) — изменение происходит от исходного изображения до конечного, и так по кругу.

Программы

Профессиональным пользователям можно рекомендовать программу MorphMan (<http://www.stoik.com>), имеющую уникальные функции послойного морфинга и морфинга видео в видео. Тем, кто хочет иметь максимально полный набор функций по подготовке и редактированию изображений в одной программе, подойдет программа FantaMorph (<http://www.fantamorph.com>). Пользователям Adobe Premiere стоит обратить внимание на плагин WinMorph (<http://www.debugmode.com/winmorph/>). Домашним пользователям, для которых не столь важна производительность и критична цена, можно рекомендовать Morpher (<http://www.asahinet.or.jp/~FX6M-FJMY/morph00e.html>) или WinMorph. Для тех, кто хочет ознакомиться с морфингом в общих чертах и добиться быстро эффекта, оптимальный выбор — программа Morpher.



Borland C++ Builder 6

для начинающих

О.ВАЛЬПА,
E-mail: sandh@narod.ru

(Продолжение. Начало в №№7-12/2004, 1-5,8/2005)

В этой статье рассказывается о назначении и применении на практике некоторых методов визуальных компонентов.

Методы компонентов

Наряду с обработчиками событий, Borland C++ Builder 6 позволяет использовать при написании программы множество методов компонентов. Некоторые из них мы уже применяли в рассмотренных программах, например LoadFromFile, SaveToFile и др. В отличие от обработчиков событий, методы не вызываются в результате каких-либо действий пользователя, а выполняются при достижении программой строки, в которой они записаны. Методы вставляются в программу самим разработчиком по мере необходимости, например, для отображения рисунка на экране монитора, работы с файлами и т.д. По сути, методы являются функциями или подпрограммами. Так же как свойства и события, одни и те же методы могут быть применимы для разных типов компонентов. В табл.1 приведено краткое описание большинства методов, расположенных в алфавитном порядке.

Табл. 1

Метод	Описание
Add	Добавляет новый элемент в список
Assign	Копирует изображение одного графического объекта в другой
BeginDrag	Используется при перетаскивании компонента
BringToFront	Позволяет изменить последовательность перекрывания компонентов
BrushCopy	Копирует часть изображения битовой матрицы на канву
CanFocus	Определяет, может ли компонент получать сообщения пользователя
ChangeScale	Используется для изменения масштаба компонента
Chord	Рисует заполненную замкнутую фигуру, ограниченную дугой окружности, эллипса и хордой
ClassName	Возвращает имя типа объекта
Clear	Очищает список или текст
ClientToScreen	Преобразует координаты клиентской области в координаты экрана
ContainsControl	Определяет, является ли указанный компонент прямым или косвенным наследником данного оконного компонента
ControlAtPos	Определяет, какой дочерний компонент имеется в указанной позиции
CopyRect	Копирует часть изображения
Delete	Удаляет элемент с указанным индексом из списка
DisableAlign	Временно запрещает выравнивание дочерних компонентов
Dormant	Создает изображение битовой матрицы в памяти
Draw	Рисует графическое изображение в указанную позицию канвы
DrawFocusRect	Рисует изображение прямоугольника в виде, используемом для отображения рамки фокуса, с помощью операции XOR
DisableAlign	Временно запрещает выравнивание дочерних компонентов
Ellipse	Рисует заполненный эллипс или окружность
Expand	Увеличивает емкость списка типа TList
FillRect	Заполняет указанный прямоугольник канвы, используя текущее значение Brush
FindNextControl	Возвращает следующий в последовательности табюляции оконный дочерний компонент
FloodFill	Закрашивает текущей кистью замкнутую область канвы, ограниченную определенным цветом

Focused	Определяет, находится ли оконный элемент в фокусе
FrameRect	Рисует на канве текущей кистью прямоугольную рамку
Free	Освобождает память, занимаемую объектом
GetTabOrderList	Строит список дочерних оконных компонентов в последовательности табюляции
HandleAllocated	Проверяет наличие дескриптора окна компонента
HandleNeeded	Создает дескриптор окна, если до этого он не существовал
Hide	Делает компонент невидимым
IndexOf	Определение первого вхождения в список заданного элемента
Insert	Вставляет элемент в заданную позицию списка
Invalidate	Сообщает Windows о необходимости полностью перерисовать компонент
LineTo	Рисует на канве прямую линию
LoadFromClipboardFormat	Загружает изображение из буфера обмена в формате Clipboard
LoadFromFile	Загружает изображение, хранящееся в файле
LoadFromResourceID	Загружает битовую карту из файла ресурсов по указанному идентификатору
LoadFromResourceName	Загружает битовую карту из файла ресурсов по указанному имени
LoadFromStream	Загружает графическое изображение из указанного потока
Lock	Блокирует канву, запрещая рисовать на ней
Move	Меняет текущую позицию элемента в списке на позицию, заданную в параметрах метода
MoveTo	Устанавливает позицию пера на позицию, заданную в параметрах метода
Pie	Рисует заполненную замкнутую фигуру в виде сегмента окружности или эллипса
Polygon	Рисует на канве заполненную замкнутую фигуру в виде многоугольника
Polyline	Рисует на канве ломаную линию
DisableAlign	Временно запрещает выравнивание дочерних компонентов
Rectangle	Рисует на канве закрашенный прямоугольник
Refresh	Перерисовывает изображение компонента на экране
Remove	Удаляет элемент с заданным значением из списка TList
Repaint	Перерисовывает изображение компонента на экране
ReplaceDockedControl	Встраивает компонент на место другого компонента
RoundRect	Рисует на канве прямоугольную рамку со скругленными углами
SaveToClipboardFormat	Создает копию изображения в формате Clipboard
SaveToFile	Сохраняет графическое изображение в файле
SaveToStream	Сохраняет графическое изображение в потоке
ScaleBy	Масштабирует оконный элемент и все содержащиеся в нем компоненты
ScaleControls	Масштабирует дочерние компоненты оконного элемента, не изменяя масштаб самого элемента
ScreenToClient	Преобразует координаты экрана в координаты клиентской области компонента
ScrollBy	Сдвигает содержимое оконного элемента
SelectFirst	Передаёт фокус дочернему компоненту, первому в последовательности табюляции
SelectNext	Передаёт фокус дочернему компоненту, следующему в последовательности табюляции за указанным компонентом
SendCancelMode	Прерывает модальное состояние элемента управления
SendToBack	Переносит компонент ниже других компонентов



SetBounds	Устанавливает одновременно свойства Left, Top, Width и Height
SetChildOrder	Изменяет позицию компонента в списке дочерних компонентов оконного элемента
SetFocus	Передает фокус элементу
SetZOrder	Перемещает компонент на вершину или в низ Z-последовательности
Show	Делает видимым невидимый компонент
StretchDraw	Рисует графическое изображение в указанную прямоугольную область канвы, подгоняя размер изображения под заданную область
TextExtent	Возвращает длину и высоту текста (в пикселах), который предполагается написать на канве текущим шрифтом
TextHeight	Возвращает высоту текста (в пикселах), который предполагается написать на канве текущим шрифтом
TextOut	Пишет указанную строку текста на канве, начиная с указанной позиции
TextRect	Пишет указанную строку текста на канве, начиная с указанной позиции и усекая текст, выходящий за пределы указанной прямоугольной области
TextWidth	Возвращает длину текста (в пикселах), который предполагается написать на канве текущим шрифтом
TryLock	Блокирует канву, если она не была заблокирована, не разрешая другим нитям многопоточного приложения рисовать на ней
Unlock	Уменьшает на единицу значение свойства LockCount, способствуя тем самым разблокировке канвы, когда LockCount станет равным нулю
Update	Перерисовывает компонент

Рассмотрим более подробно назначение и описание некоторых из этих методов, наиболее часто используемых, с примерами их применения.

Начнем с создания небольшой программы, которая наглядно покажет, как применяются методы Add (добавить), Hide (скрыть), Show (показать) и Delete (удалить).

Создайте новое приложение и поместите на его форму компонент ListBox и четыре кнопки типа Button. Измените размеры

формы и разместите на ней компоненты так, как показано на рис. 1.

Задайте свойство названия — Caption — для формы и кнопок в соответствии с этим же рисунком. Теперь дважды щелкните левой кнопкой мыши по компоненту первой кнопки Button1 и впишите для нее код обработки события OnClick в виде программной строки:

```
ListBox1->Items->Add(String(i++));
```

Эта строка содержит метод Add, который позволит добавить в компонент ListBox1 элемент списка, представляющий собой строку, формирующуюся из переменной i с помощью функции String. Каждый раз при вызове данной программной строки с методом Add, переменная i будет увеличивать свое значение, благодаря тому что после нее вписан оператор автоинкремента "++" (увеличения значения на единицу). Описание переменной с присвоением ей начального значения в виде строки int i=0; необходимо вписать в программу в инспекторе объектов, сразу после строк с описанием включений #include. В этом случае переменная будет доступна любому обработчику программы. Если описать эту переменную в самом обработчике события, то она станет локальной (недоступной другим обработчикам), и будет инициализироваться каждый раз при формировании этого события.

Аналогично сформируйте заготовку кода обработки события OnClick для компонента кнопки Button2 и впишите в него программную строку:

```
ListBox1->Hide();
```

Эта строка содержит метод Hide, который скрывает изображение компонента на форме от пользователя программы.

Подобным образом впишите в код обработки события OnClick для компонента кнопки Button3 программную строку:

```
ListBox1->Show();
```

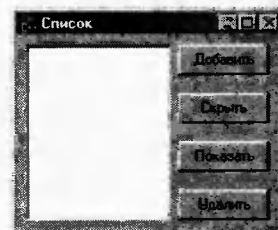


Рис. 1

Динамические структуры данных: линейные списки и рекурсия

А.КОРБИТ, Т.КРИВОНОСОВА,

г. Минск, БГУИР, ФТИУ, каф.ВМИП.

E-mail: a_korbit@mail.ru

Как уже было показано [1], простой способ связать множество элементов — сделать так, чтобы каждый элемент содержал ссылку на следующий. Такой список называют *однонаправленным (односвязным)*. Если добавить в каждый элемент ссылку на предыдущий элемент, получится *двунаправленный список (двухсвязный)*, если последний элемент связать указателем с первым, получится *кольцевой список*.

Каждый элемент списка содержит *ключ*, идентифицирующий этот элемент. Ключ обычно бывает либо целым числом, либо строкой, и является частью поля данных. В качестве ключа в процессе работы со списком могут выступать разные части поля данных.

Например, если создается линейный список из записей, содержащих фамилию, год рождения, стаж работы и пол, любая часть записи может выступать в качестве ключа. При упорядочивании такого списка по алфавиту ключом будет фамилия, а при поиске, например, ветеранов труда ключом будет стаж работы. Как правило, ключи должны быть уникальными, но в ряде случаев ключи разных элементов списка могут совпадать. И в последнем случае лучше всего использовать схемы организации структур данных по принципу "хеширования".

Над списками можно выполнять следующие операции:

- начальное формирование списка (создание первого

элемента);

- добавление элемента в конец списка;
- обработку (чтение, удаление и т.п.) элемента с заданным ключом;
- вставку элемента в заданное место списка (до или после элемента с заданным ключом);
- упорядочивание списка по ключу.

Если программа состоит из функций, решающих вышеперечисленные задачи, то необходимо соблюдать следующие требования:

- все параметры, не изменяемые внутри функций, должны передаваться с модификатором const;
- указатели, которые могут изменяться (например, при удалении из списка последнего элемента, указатель на конец списка требуется скорректировать), передаются по адресу.

Удаление элемента из списка

Рассмотрим подробнее *функцию удаления элемента из списка*. Ее параметрами должны быть указатели на начало списка, на конец списка и ключ элемента, подлежащего удалению.

Удаление из списка происходит по-разному, в зависимости от того, находится элемент в начале, в середине или в конце списка. Сначала нужно проверить, находится ли уда-



Эта строка содержит метод Show, который показывает изображение компонента на форме для пользователя программы. Наконец, впишите код обработки события OnClick для компонента кнопки Button4 в виде программной строки:

```
if (i>0) ListBox1->Items->Delete(--i);
```

Эта строка содержит метод Delete, который удаляет указанный элемент списка из компонента ListBox1. Обратите внимание, что в качестве номера удаляемого элемента списка мы использовали ту же переменную i, которая применялась при добавлении элементов списка в компонент. При этом перед ее использованием она автоматически декрементируется (уменьшает свое значение на единицу) благодаря применению оператора "--". Метод удаления имеет смысл применять только в том случае, если в списке есть элементы, т.е. имеется что удалять. Для проверки данного условия в программную строку включен оператор сравнения переменной индекса элементов списка с нулем — `if(i>0)`, который позволяет выполнять удаление элемента из списка только при натуральном значении переменной i, т.е. когда элементы в списке есть. Сохраните созданный проект, запустите приложение на выполнение и проверьте, как оно работает.

Теперь можно рассмотреть некоторые детали примененных и некоторых других методов, которые часто используются в программах.

Метод Add применяется к компонентам, имеющим списки или строки. Данный метод возвращает при выполнении индекс добавленного элемента. Например, программная строка `i=ListBox1->Items->Add("Новый элемент списка");` присвоит переменной i значение возвращаемого параметра. Индекс первого элемента списка всегда равен нулю. Данный метод увеличивает значение свойства Count (номер элемента) на единицу. Если значение Count равно значению свойства Capacity (размер списка), то увеличивается значение Capacity, и перераспределяется память под новые элементы. Напомню, что доступ к свойствам

Count и Capacity осуществляется с помощью строк `ListBox1->Item->Count` и `ListBox1->Item->Capacity` соответственно. Используя данные свойства, можно исключить в предыдущей программе переменную i, сохранив при этом все ее функции. Предлагаю читателям проделать это самостоятельно для получения собственного опыта.

Метод Hide делает компонент невидимым, задавая свойству Visible компонента значение False. Если компонент является контейнером для других компонентов, то эти дочерние компоненты также делаются невидимыми. Несмотря на то что компонент становится невидимым, его свойства и методы остаются доступными в программе.

Метод Show делает видимым ранее невидимый компонент. Он задает значение True свойству Visible компонента и проверяет, является ли видимым родительский компонент.

Метод Delete удаляет из списка элемент с указанным индексом.

Метод CanFocus определяет, может ли компонент получать сообщения пользователя, т.е. может ли он получать фокус. Функция возвращает True, если у компонента и всех его родителей свойства Visible и Enabled установлены в True. В противном случае возвращается значение False.

Метод ChangeScale(M,D) используется для изменения масштаба компонентов. При его применении масштабируются такие свойства компонента как Top и Left, определяющие его местоположение, а также Width и Height, определяющие его размер. Параметры метода M и D определяют соотношение новых размеров компонентов и их прежних размеров. Например, чтобы уменьшить размеры компонента до 50% начального значения, нужно задать M равным 50, а D равным 100. То же самое можно сделать, задав M=1 и D=2. Если вы хотите увеличить размер, то нужно задать значение M больше чем значение D. Так, например, программная строка `"ChangeScale(1,2);"` уменьшит значение компонентов в 2 раза.

(Продолжение следует)

ляемый элемент в начале списка — в этом случае следует скорректировать указатель *begin* на начало списка, так чтобы он указывал на следующий элемент в списке, адрес которого находится в адресном поле первого элемента. Новый начальный элемент списка должен иметь в поле указателя на предыдущий элемент значение NULL.

Если удаляемый элемент находится в конце списка, требуется сместить указатель *end* конца списка на предыдущий элемент. Кроме того, нужно обнулить для нового последнего элемента указатель на следующий элемент. Если удаление происходит из середины списка, то единственное, что надо сделать — обеспечить связь предыдущего и последующего элементов. После корректировки указателей память из-под удаленного элемента освобождается.

Сортировка элементов списка

Сортировка связанного списка заключается в изменении связей между элементами. В данном случае наиболее прост линейный выбор. Алгоритм состоит в том, что исходный список просматривается, и каждый элемент вставляется в новый список на место, определяемое значением его ключа.

При работе со списками удобно пользоваться рекурсивными алгоритмами, и прежде чем рассмотреть рекурсивные приемы работы со списками, вспомним, что такое *рекурсия*.

Рекурсивные функции

Рекурсивной (самовызываемой или самовызывающей) называют функцию, которая прямо или косвенно вызывает сама себя.

Вторая основная особенность — при каждом обращении к рекурсивной функции создается новый набор объектов автоматической памяти, локализованных в коде функции.

Возможность прямого или косвенного вызова позволяет различать прямую или косвенную рекурсию. Функция называется косвенно рекурсивной в том случае, если она содержит обращение к другой функции, содержащей прямой или косвенный вызов первой функции. В этом случае по тексту определения функции ее рекурсивность (косвенная) может быть не видна. Если в функции используется вызов этой же функции, то имеет место прямая рекурсия, т.е. функция, по определению, рекурсивная.

Рекурсивные алгоритмы эффективны в задачах, где рекурсия использована в самом определении обрабатываемых данных. Поэтому изучение рекурсивных методов нужно проводить, вводя динамические структуры данных с рекурсивной структурой, что далее и будет сделано. Рассмотрим вначале только принципиальные возможности, которые предоставляет язык Си для организации рекурсивных алгоритмов.

В рекурсивных функциях необходимо выполнять следующие правила:

- при каждом ее вызове в функцию передавать модифицированные данные;

- на некотором шаге должен быть прекращен дальнейший вызов этой функции, т.е. рекурсивный процесс должен шаг за шагом так упрощать задачу, чтобы для нее появилось нерекурсивное решение (чтобы не допустить ошибку, заключающуюся в том, что функция будет вызывать саму себя бесконечно);



- после завершения каждого экземпляра рекурсивной функции, в вызывающую функцию должен возвращаться некоторый результат для дальнейшего его использования.

Пример 1. Заданы два числа — a и b . Больше из них надо разделить на меньшее, используя рекурсию.

Текст программы может быть следующим:

```
double proc(double, double);
void main (void) {
    double a, b;
    puts(" Введи значения a, b : ");
    scanf("%lf%lf", &a, &b);
    printf("\n Результат деления : %lf", proc(a,b));
    getch();
}
//-----Функция-----
double proc(double a, double b) {
    if ( a < b ) return proc ( b, a );
    else return a/b;
}
```

Если $a > b$, т.е. условие, поставленное в функции ($a < b$), не выполняется, то функция *proc()* возвращает нерекурсивный результат.

Пусть теперь условие выполнилось, тогда *proc()* обращается сама к себе, аргументы в вызове меняются местами, и последующее обращение приводит к тому, что условие не выполнится, и функция возвращает нерекурсивный результат.

Пример 2 (классический). Функция для вычисления факториала неотрицательного числа.

```
double fact (int k) {
    if ( k < 0 ) return 0;
    if ( k == 0 ) return 1;
    return ( k * fact ( k - 1 ) );
}
```

Для отрицательного аргумента результат (по определению факториала) не существует. В этом случае функция возвращает нулевое значение. Для нулевого параметра функция возвращает значение 1 (по определению, $0! = 1$). В противном случае вызывается та же функция с уменьшенным на 1 значением параметра, и результат умножается на текущее значение параметра. Тем самым для положительного значения параметра k организуется вычисление произведения $k * (k-1) * (k-2) * \dots * 3 * 2 * 1 * 1$.

Последнее значение "1" — результат выполнения условия $k=0$, т.е. последовательность рекурсивных обращений к функции *fact* прерывается при вызове *fact(0)*. Именно этот вызов приводит к последнему значению "1" в произведении, так как последнее выражение, из которого вызывается функция, имеет вид: $1 * fact(1-1)$.

Пример 3. В языке Си отсутствует операция возведения в степень, а используется библиотечная функция *double pow(double, double)*. Поэтому следующая рекурсивная функция вычисления целой степени вещественного ненулевого числа может оказаться полезной:

```
double expo(double a, int n) {
    if (n == 0) return 1;
    if (a == 0.0) return 0;
    if (n > 0) return a * expo(a, n-1);
    if (n < 0) return expo (a, n+1) / a;
}
```

При обращении вида *expo(2.0, 3)* рекурсивно выполняются вызовы функции *expo()* с изменяющимся вторым аргументом: *expo(2.0, 3)*, *expo(2.0, 2)*, *expo(2.0, 1)*, *expo(2.0, 0)*. При этих вызовах последовательно вычисляется произведение $2.0 * 2.0 * 2.0 * 1$

и формируется нужный результат.

Вызов функции для отрицательного значения степени,

например *expo(5.0, -2)*, эквивалентен вычислению выражения $expo(5.0, 0) / 5.0 / 5.0$.

Отметим математическую неточность. В функции *expo()* для любого показателя при нулевом основании результат равен нулю, хотя возведение в нулевую степень нулевого основания должно приводить к ошибочной ситуации.

Пример 4. Ввести целые числа, не равные нулю (без использования массива), и вывести их в обратном порядке.

```
#include<stdio.h>
//----- рекурсивная функция-----
void Rec() {
    int i;
    scanf("%d", &i);
    if ( i != 0 ) Rec();
    else printf ("\n Вывод чисел : ");
    printf(" %d ", i );
}
//-----
void main() {
    printf("\n Вводите числа. Признак окончания ввода - 0 : ");
    Rec();
}
```

Результат выполнения программы:

Вводите числа. Признак окончания ввода — 0 : 1 2 3 4 5 6 7 8 0
Вывод чисел : 0 8 7 6 5 4 3 2 1

Пример 5. Рассмотрим функцию определения корня уравнения $f(x)=0$ на отрезке $[a,b]$ с заданной точностью *eps*. Предположим для простоты, что исходные данные задаются без ошибок, т.е. $eps > 0$, $f(a)*f(b) < 0$, $b > a$, и вопрос о возможности существования нескольких корней на отрезке $[a,b]$ нас не интересует. Не очень эффективная рекурсивная функция для решения поставленной задачи приведена в следующей программе:

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>

int counter = 0; // Счетчик обращений к тестовой функции

//--Нахождение корня уравнения методом деления отрезка пополам--
double Root(double f(double), double a, double b, double eps)
{
    double fa = f(a), fb = f(b), c, fc;
    if ( fa * fb > 0 )
    {
        printf("«\n На интервале a,b НЕТ корня!");
        exit(1);
    }
    c = (a + b) / 2.0;
    fc = f(c);
    if (fc == 0.0 || (b - a) < eps) return c;
    return (fa * fc < 0.0) ? Root(f, a, c, eps) : Root(f, c, b, eps);
}

//-----
void main()
{
    double x, A=0.1, B=3.5, EPS=5e-5;
    double fun(double); // Прототип тестовой функции
    x = Root (fun, A, B, EPS);
    printf ("\n Число обращений к тестовой функции = %d ", counter);
    printf ("\n Корень = %lf ", x);
    getch();
}

//----- Определение тестовой функции - fun -----
double fun (double x)
{
    counter++; // Счетчик обращений - глобальная переменная
    return (2.0/x * cos(x/2.0));
}
```

Значения A , B и EPS заданы постоянными только для тестового анализа полученных результатов, а в дальнейшем лучше вводить данные с клавиатуры.

В результате выполнения программы с определенными в ней конкретными данными получим:



Число обращений к тестовой функции = 54

Корень = 3.141601

(можно добавить контроль — значение функции в найденном корне должно быть очень маленьким).

В рассматриваемой программе использовали библиотечную функцию `exit`, прототип которой размещен в заголовочном файле `stdlib.h`. Функция `exit` позволяет досрочно завершить выполнение программы (возвращает операционной системе значение своего параметра).

Неэффективность предложенной программы связана, например, с излишним количеством обращений к программной реализации функции, для которой определяется корень. При каждом рекурсивном вызове `Root` повторно вычисляются значения $f(a)$ и $f(b)$, хотя они уже известны после предыдущего вызова. Предложите свой вариант исключения лишних обращений к функции f при сохранении рекурсивности.

В литературе по программированию рекурсиям уделено достаточно внимания, как в теоретическом плане, так и в плане рассмотрения механизмов реализации рекурсивных алгоритмов. Сравнивая рекурсию с итерационными методами, отмечают, что рекурсивные алгоритмы наиболее пригодны в случаях, когда поставленная задача или используемые данные определены рекурсивно. В тех же случаях, когда вычисляемые значения определяются с помощью простых рекуррентных соотношений, гораздо эффективнее применять итерационные методы.

Таким образом, рассмотренные выше примеры только иллюстрируют схемы организации рекурсивных функций, но не являются примерами эффективного применения рекурсивного подхода к вычислениям.

При обработке динамических информационных структур, которые включают рекурсивность в само определение обрабатываемых данных, применение рекурсивных алгоритмов не имеет конкуренции со стороны итерационных методов.

Рекурсия при обработке списков

Структура, такая как динамический односвязный список, представляет собой конструкцию, которую удобно обрабатывать при помощи рекурсивных процедур. Рассмотрим формирование списка и вывод списка на экран дисплея в виде рекурсивных функций. Сделаем это с целью показать на простом примере особенности рекурсивной обработки динамических информационных структур.

Например, пусть, как и раньше, данная структура имеет вид:

```
struct Spis {
    int info;
    Spis *Next;
};
```

В каждом звене списка имеется информационная часть (данные — поле `info`) и адресная часть (адрес следующего звена списка — `Next`). Если адресная часть нулевая (`NULL`), то список пройден до конца. Для начала просмотра списка нужно знать только адрес его первого элемента.

Рассмотрим, какие действия должны выполнять функция рекурсивного формирования списка и функция его рекурсивного просмотра с выводом данных о звеньях списка на экран дисплея.

Функция рекурсивного заполнения списка `input()` должна возвращать указатель на сформированный и заполненный данными с клавиатуры динамический список. Предполагается, что при каждом вызове этой функции формируется новый элемент списка.

Функция заканчивает работу, если для очередного звена списка для информационной части (`info`) введено нулевое значение переменной.

В противном случае заполненная с клавиатуры структу-

ра подключается к списку, а ее элементу (`Spis *Next`) присваивается значение, которое вернет вызванная рекурсивно функция `input()`.

Прежде чем рассматривать текст функции `input`, помещенный в приведенной ниже программе, еще раз сформируем использованные в ней конструктивные решения.

1. При каждом обращении к этой функции в памяти формируется новый элемент списка, указатель на начало которого возвращает эта функция.

2. Выделение памяти для элементов списка и заполнение их значениями полностью определяет пользователь.

3. Если для очередного звена списка вводится нулевое значение элемента `info`, то звено не подключается к списку, процесс формирования списка заканчивается. Такой набор данных, введенных для элемента очередного звена, считается терминальным.

4. Если нулевое значение элемента `info` введено для первого звена списка, то функция `input` возвращает значение `NULL`.

5. Список определяется рекурсивно как первое (головное) звено, за которым следует присоединяемый к нему список.

6. Псевдокод рекурсивного алгоритма формирования списка:

Если введены терминальные данные для звена:

- освободить память, выделенную для звена;
- вернуть нулевой указатель.

Иначе:

- элементу связи звена присвоить результат формирования продолжения списка (использовать тот же алгоритм).

Все — если

вернуть указатель на звено.

Для структуры типа `Spis` выделяется память. После того как пользователь введет данные, выполняется их анализ. В случае терминальных значений операция `delete` освобождает память от ненужного звена списка, и выполняется оператор `return NULL`.

В противном случае элементу связи (указатель `Spis *Next`) присваивается результат рекурсивного вызова функции `input`. Далее все повторяется для нового экземпляра этой функции. Когда при очередном вызове `input` пользователь введет терминальные данные, функция `input` вернет значение `NULL`, которое будет присвоено указателю связи предыдущего звена списка, и выполнение функции будет завершено.

Функция рекурсивного просмотра и печати списка. Аргументом функции является адрес начала списка (или адрес любого его звена), передаваемый как значение указателя — параметра `Spis *begin`. Если параметр имеет значение `NULL` — список исчерпан, и исполнение функции прекращается. В противном случае выводятся на экран значения элементов той структуры, которую адресует параметр, и функция `print` рекурсивно вызывается с параметром `begin → Next`. Тем самым выполняется продвижение к следующему элементу списка. Конец известен — функция печатает "Список исчерпан!" и больше не вызывает себя, а завершает работу.

Текст программы с рекурсивными функциями:

```
#include <stdlib.h>
#include <stdio.h>

struct Spis {
    int info;
    Spis *Next;
};

//----- Функция ввода и формирования списка -----
Spis* input(void) {
    int i;
    Spis *t = new Spis;
    printf("\n info ? : ");
    scanf("%d", &i);
    if (i == 0) {
```

```

        delete t;
        return NULL;
    }
    t -> info = i;
    t -> Next = input ();    // Рекурсивный вызов
    return t;
}
//----- Функция вывода списка на экран -----
void print (Spis *t) {
    if (t == NULL) {
        printf ("\n Spisok - END! ");
        return ;
    }
    printf ("\n Info = %d ", t->info);
    print (t -> Next );    // Рекурсивный вызов
}
//-----
void main () {
    Spis *begin = NULL;    // Начало списка
    printf ("\n Vvedi Info : ");
    begin = input ();    // Ввод списка
    printf ("\n Spisok : ");    // Напечатать список
    print ( begin );
    getch();
}

```

Рассмотрим подробно работу программы формирования списка с использованием функции *input*.

1. Объявлен указатель на начало списка и изначально установлен в *NULL*. Запустив на выполнение функцию *main*, через *begin = input()*; будет вызвана функция *input* первый раз (*input* возвращает указатель на структуру).

2. Теперь работает функция *input()*; (первый шаг):

1) захват памяти: *t = new Spis*;



2) формирование информационной части *info*;

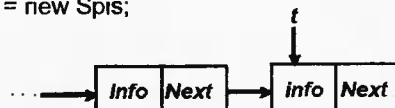
3) если введен ноль: (*i = 0*), то освобождаем захваченную память и возвращаем *NULL*, т.е. *begin = NULL*; — список закончился, так и не начавшись.

4) иначе, т.е. *i* № 0, используя указатель структуры *t* на тип *Spis* вызывается функция *input* на следующий шаг *t -> Next = input()*; и возвращает значение указателя *t* на первый созданный элемент: *return t*; таким образом, указатель на начало *begin* установлен на этот элемент.

При последующих вызовах рекурсивной функции работает функция *input*, вызванная через указатель последнего созданного элемента:

1) текущий захват памяти под новый элемент (напомним, элементы добавляются в список с конца):

t = new Spis;



2) формирование информационной части *info*;

3) если *i = 0*, то освобождаем захваченную память для созданного элемента (*delete t*;) и возвращаем *NULL*, т.е. формирование очереди закончилось.

4) иначе, т.е. *i* № 0, формируем информационную часть и, используя адресную часть только что созданного элемента, опять обращаемся к функции *input*:

t -> Next = input();

и возвращаем указатель на только что созданный (последний) элемент, который будет размещен в адресной части теперь предпоследнего элемента.

Литература

1. А.Корбит, Т.Кривоносова Организация памяти и структуры данных. — Радиомир. Ваш компьютер, 2005, №6, С.28.
2. А.В.Крячков и др. Программирование на С и С++. — М.: Горячая линия — Телеком, 2000.
3. Т.А.Павловская С/С++. Программирование на языке высокого уровня. — СПб.: ПИТЕР, 2002.

А.КОРБИТ, Т.КРИВОНОСОВА,

г.Минск, БГУИР, ФИТИУ, каф.ВмиП.

E-mail: a_korbit@mail.ru

Поиск и сортировки: сортировка вставками (включениями)

В[1] была сформулирована общая постановка задачи сортировки и рассмотрены методы сортировки выбором. Теперь рассмотрим алгоритмы сортировки при помощи вставки.

Метод 1. Линейная (простая) вставка

Данный метод основан на последовательной вставке элементов в упорядоченный рабочий массив. Линейная вставка чаще всего используется, когда динамически вносятся изменения в массив, все элементы которого известны, и который все время должен быть упорядоченным.

Алгоритм линейной вставки следующий. Первый элемент исходного массива помещается в первую позицию рабочего массива. Следующий элемент исходного массива сравнивается с первым. Если этот элемент больше, он помещается во вторую позицию рабочего массива. Если этот элемент меньше, то первый элемент сдвигается на вторую позицию, а новый элемент помещается на первую. Далее все выбираемые из исходного массива элементы последовательно сравниваются с элементами рабочего массива, начиная с первого, до тех пор пока не встретится элемент больше добавляемого. Тогда этот элемент и все последующие за ним элементы рабочего массива смещаются на одну позицию, освобождая место для нового элемента.

Таким образом, каждый новый элемент *a[i]* будем вставлять на подходящее место в уже упорядоченную совокуп-

ность *a[1], a[2], ..., a[i-1]*. Это место определяется последовательными сравнениями элемента *a[i]* с упорядоченными элементами *a[1], ..., a[i-1]*.

Такой метод сортировки и назвали сортировкой простыми вставками. Он, как и сортировка выбором, требует порядка $(N^2 - N)/2$ операций сравнения.

Метод 2. Центрированная и двоичная вставки

Центральный элемент этого массива часто называют медианой, она разбивает массив на две ветви — нисходящую (левую) и восходящую (правую).

Алгоритм центрированной вставки можно сформулировать так.

В позицию, расположенную в середине рабочего массива, помещается первый элемент (он и будет медианой). Нисходящая и восходящая ветви имеют указатели, которые показывают на ближайшие к началу и концу занятые позиции. После загрузки первого элемента в центральную позицию, оба указателя совпадают и показывают на него. Следующий элемент исходного массива сравнивается с медианой. Если новый элемент меньше, то он размещается на нисходящей ветви, в противном случае — на восходящей ветви. Кроме того, соответствующий концевой указатель продвигается на единицу вниз (нисходящая ветвь) или на единицу вверх (восходящая ветвь).



Каждый последующий элемент исходного массива сравнивается вначале с медианой, а затем с элементами на соответствующей ветви, до тех пор пока не займет нужную позицию.

Метод 3. Двоичная (бинарная) вставка

Здесь для поиска места для вставки элемента $a[i]$ используется алгоритм двоичного (бинарного) поиска. Элемент $a[i]$ сравнивается вначале с элементом $[i/2]$. Затем, если он меньше, то сравнивается с элементом $[i/4]$, а если больше, то с $[i/2] + [i/4]$, и т.д., до тех пор пока для него не найдется место. Все элементы рабочего массива, начиная с позиции вставки и ниже, сдвигаются на одну позицию, освобождая место для i -го элемента.

Количество операций сравнения будет порядка $N \cdot \log_2(N)$.

Вариант алгоритма двоичной вставки

Начало цикла (выполнять для i от 2 до N)

$R=i$;

$L=1$

Начало цикла: выполнять пока $L < R$;

$k=(L+R)/2$;

если $a[k] > a[i]$

тогда

$L=k+1$;

иначе

$R=k$;

все

конец цикла

$k=R$;

$x=a[i]$;

Начало цикла (выполнять для m от i до $k+1$ шаг -1)

$a[m] = a[m-1]$;

конец цикла

$a[k]=x$;

все

конец цикла

Цикл "пока $L < R$ " — цикл поиска места вставки. Он основан на методе двоичного поиска.

Цикл "для m от i до $k+1$ шаг -1" сдвигает элементы для освобождения места вставки.

Пример

Пусть есть последовательность чисел $\{0, 1, 9, 2, 4, 3, 6, 5\}$. Сортировать будем по убыванию

Изначально рабочий массив пуст. Размещаем в нем 1. Для простоты будем записывать всю совокупность, а элемент, который вставляем, будет подчеркнут.

Существующий массив отделен вертикальной линией.

Для вставки элемента в нужное место все элементы, стоящие за ним, сдвигаем до позиции, которую занимал вставляемый на данном шаге элемент.

1) 1, | 9, 2, 4, 3, 6, 5, 0

2) 9, 1, | 2, 4, 3, 6, 5, 0

3) 9, 2, 1, | 4, 3, 6, 5, 0

4) 9, 4, 2, 1, | 3, 6, 5, 0

5) 9, 4, 3, 2, 1, | 6, 5, 0

6) 9, 6, 4, 3, 2, 1, | 5, 0

7) 9, 6, 5, 4, 3, 2, 1, | 0

Имеем:

8) 9, 6, 5, 4, 3, 2, 1, 0 |

Литература

1. А.Корбит, Т.Кривоносова. Поиск и сортировки: сортировка выбором. — Радиомир. Ваш компьютер, 2005, №8, С.36.

2. В.Котов и др. Методы алгоритмизации. — Мн.: Народная асвета, 1996.

3. А.Крячков и др. Программирование на С и С++. Практикум. — М.: Горячая линия-Телеком, 2000.

П.СОКОЛЕНКО,

г.Ростов-на-Дону.

E-mail: pts@aaanet.ru

http://macro.aaanet.ru

Арифметика длинных чисел

(Окончание. Начало в №№7-8/2005)

8. Деление длинных чисел

Деление отличается от других арифметических операций тем, что значения разрядов частного не вычисляются, а подбираются. Поэтому деление длинных чисел выполняется достаточно долго, а соответствующая процедура имеет большой размер. На практике необходимость деления длинных чисел возникает не так уж часто, и в таких случаях надо использовать специальные приемы для его ускорения. В данном разделе описана простая процедура, выполняющая деление длинного числа на обычное 32-разрядное число.

Деление целых чисел выполняют инструкции `div` и `idiv`. Первая из них интерпретирует содержимое старших битов делимого и делителя как часть кода числа, а вторая — как знак числа. По этой причине инструкция `idiv` неприменима при делении длинных чисел. Отрицательные длинные числа перед делением надо инвертировать, а после деления определить знак результата. Если делимое и делитель имели разные знаки, то частное и остаток инвертируются.

Деление длинного числа на двойное слово выполняет процедура 6. Ее входными параметрами являются адрес массива, содержащего делимое, и значение делителя. Выходным параметром является адрес массива для записи частного, по размеру он должен совпадать с массивом, содержащим делимое. Если делимое вам больше не понадобится, то его адрес можно указать в качестве адреса частного.

Процедура 6. Деление длинного числа на двойное слово

```

PUBLIC _bigdiv
ASSUME cs:subr
subr SEGMENT word public 'subr'
.486
;
; Размещение параметров в стеке:
; [esp+12] - адрес длинного числа (делимого)
; [esp+8] - значение делителя
; [esp+4] - адрес частного (результата)
;
;
_bigdivproc far
mov edi, [esp+12]; в edi адрес делимого
mov ecx, [edi] ; в ecx размер делимого
mov ebx, [esp+8] ; в ebx значение делителя
mov esi, [esp+4] ; в esi адрес для записи частного
mov [esi],ecx ; в esi(0) предварительный размер частного
xor edx, edx ; edx = 0, очистка старшей части
; первого делимого
;
; Цикл деления длинного числа на 32-разрядное
divlp: mov eax,[edi+ecx*4] ; eax = edi(i), очередная часть
; делимого
div ebx ; eax:edx = edx:eax / ebx, деление
mov [esi+ecx*4], eax ; edi(i) = eax, запись частного
; в память
loop divlp ; управление повторами цикла (i = i - 1)
;
; Проверка содержимого старшего разряда частного
mov eax, [esi] ; eax = esi(0), размер частного
cmp [esi+eax*4], ecx ; старший разряд частного пустой?
jne @F ; -> нет
dec dword ptr [esi] ; Z(0) = Z(0) - 1, исключение

```



```

@@:  ret                ; пустого разряда
     _bigdivENDP        ; возврат на вызывающий модуль
subr  ENDS
     END

```

В процедуре 6 тело цикла, выполняющего деление (divr), состоит всего из трех команд, без учета управления повторами цикла. Простота цикла объясняется тем, что деление длинного числа производится начиная со старших разрядов. Это позволяет использовать учетверенное значение счетчика цикла в качестве смещения в указателях адресов и исключает необходимость формирования старшей части делимого для следующей итерации цикла (содержимого регистра edx), его формирует инструкция div.

Обозначим буквами X, Y и Z делимое, делитель и частное, а буквой n — количество разрядов делимого. X и Z — это массивы, содержащие длинные числа, а Y — 32-разрядное число.

Цикл divr повторяется от i=n, до 1, вычисления выполняются по формуле:

$$Z(i) = X(i) / Y. \quad (7)$$

Здесь X(i) и Z(i) — разряды делимого и частного.

В результате деления старший разряд частного может оказаться пустым, в таком случае для его исключения количество разрядов частного уменьшается на 1.

При выходе из цикла divr в регистре edx находится остаток от деления. Если он нужен в основной программе, то выберите по своему усмотрению способ передачи содержимого регистра edx в основную программу. Например, можно добавить параметр, содержащий адрес для записи остатка от деления.

9. Преобразование длинного числа в последовательность цифр

Для визуальной оценки результатов работы с длинными числами надо иметь возможность их вывода на экран монитора или на печать. Операционные системы не поддерживают вывод длинных чисел, от них это и не требуется, но любая ОС позволяет вывести заранее сформированную последовательность десятичных или шестнадцатеричных цифр. Процедура, преобразующая длинное целое положительное число в последовательность десятичных цифр в коде ASCII, описана в данном разделе. Для вывода отрицательного числа его надо предварительно инвертировать и предусмотреть вывод знака "минус".

Входным параметром процедуры 7 является адрес преобразуемого длинного числа, а выходным — адрес буфера для записи строки символов в коде ASCII. Требуемый размер буфера зависит от количества цифр в числе, которое точно не известно, но его можно оценить. Если выводимое число содержит k разрядов, то выделив для буфера (k+1)*10 байтов, вы наверняка не ошибетесь. Процедура заполняет буфер в таком порядке: количество цифр в буфере, последняя (младшая) цифра, предпоследняя цифра, ..., первая (старшая) цифра. Поэтому выводить результат на экран или на печать надо начиная с конца буфера. О выводе знака мы поговорим после описания процедуры.

Процедура 7. Преобразование длинного числа в строку цифр в коде ASCII

```

PUBLIC  _lngdiv
ASSUME  cs:subr
subr    SEGMENT word public 'subr'
        .486
;       Размещение параметров в стеке:
;       [esp+8] - адрес преобразуемого длинного числа (X)

```

```

;       [esp+4] - адрес буфера для записи строки ASCII символов (Y)
;
_lngdiv proc far
mov edi, [esp+8] ; в edi адрес преобразуемого числа X
mov ebx, [edi]   ; ebx = X(0), размер преобразуемого числа
mov esi, [esp+4] ; в esi адрес буфера для записи символов Y
jmp @F           ; на начало внешнего цикла
; Внутренний цикл деления на 10
innrpl: mov eax, [edi+ecx*4] ; eax = X(i), чтение очередного
                           ; разряда числа
        div _ddten          ; edx:eax / 10, деление
mov [edi+ecx*4], eax ; X(i) = eax, запись частного
                           ; в память
loop innrpl ; управление повторами цикла (i = i - 1)
; Запись ASCII-кода цифры в выходной массив
extrpl: inc esi           ; j = j + 1, адрес для записи цифры
add dl, '0'              ; преобразование цифры в код ASCII
mov [esi], dl            ; Y(j) = dl, запись очередной цифры
                           ; в буфер
; Подготовка к следующему делению и исключение пустых разрядов
@@:     xor edx, edx       ; очистка старшей части делимого
mov ecx, ebx             ; новое количество повторов
or [edi+ecx*4], edx ; старшее слово содержит ноль?
jne innrpl              ; нет, на начало внутреннего цикла
dec ebx                 ; уменьшение размера делимого
jne @B                  ; возврат на проверку следующего разряда
; Вычисление количества цифр в буфере
mov eax, [esp+4] ; в eax адрес начала буфера
sub esi, eax ; esi = esi - eax, количество цифр
jne @F ; в буфере есть цифры
mov [eax], word ptr 1 ; длинное число оказалось нулем
ret ; возврат на вызывающий модуль
@@:     mov ebx, esi ; ebx = esi, количество цифр
mov byte ptr [eax], bl ; запись количества цифр в буфере
ret ; возврат на вызывающий модуль
_lngdiv ENDP
_ddten dd 10
subr    ENDS
        END

```

В процедуре 7 реализован следующий алгоритм преобразования длинного числа в последовательность цифр. Исходное число делится на 10, остаток от деления преобразуется в код ASCII и записывается в буфер. Частное от первого деления снова делится на 10, остаток от деления записывается в буфер. И так повторяется до тех пор, пока число не будет преобразовано полностью. Обратите внимание, что при таком способе формирования старшая цифра числа находится в конце, а младшая — в начале буфера. Цикл деления ничем не отличается от цикла, описанного в процедуре 6, только в данном случае он имеет метку innrpl, и его выполнение повторяется многократно.

После выхода из цикла деления, в младшем байте регистра edx находится очередная цифра числа. Для преобразования в код ASCII к нему прибавляется ASCII-код цифры 0 (30h), и результат записывается в очередной байт буфера. Если вам нужен двоично-десятичный код цифр, то просто удалите команду add dl, "0".

При каждом делении содержимое старшего разряда преобразуемого числа уменьшается. Как только оно окажется равным нулю, надо уменьшить на 1 количество повторов цикла деления. Преобразование закончится, как только младший разряд преобразуемого числа окажется пустым. Перед выходом из процедуры в младшее слово буфера записывается количество цифр в числе. Если исходное число было равно нулю, то в буфере окажется только одна цифра — ноль.

Процедура не формирует знак числа, поскольку только программист может решить, было число положительным или отрицательным. Знак числа лучше проверять в вызывающем модуле, код знака можно поместить в конце буфера, после первой цифры числа, увеличив на 1 количество символов, находящихся в буфере. В таком случае знак будет выведен на экран или на печать перед первой цифрой числа.

10. Преобразование последовательности цифр в длинное число

В данном разделе мы рассмотрим преобразование, обратное описанному в предыдущем разделе. Любая операционная система позволяет разместить в указанном буфере последовательность кодов символов, вводимых с клавиатуры. Остается только преобразовать введенную последовательность кодов цифр в длинное целое число.

Такое преобразование выполняет процедура 8. Ее входным параметром является адрес буфера, содержащего строку цифр, представленных в коде ASCII. Код цифры занимает один байт, в младшем (нулевом) байте буфера должно быть указано количество цифр в строке. Процедура не проверяет, являются ли символы цифрами, такую проверку надо делать в процессе формирования строки. Если число имеет знак, то его код не должен находиться в буфере.

Выходным параметром является адрес массива, выделенного для записи формируемого длинного числа. Содержимое двойных слов этого массива не имеет значения, а их количество должно быть на 1 больше предполагаемого количества цифр в числе. Вы не ошибетесь, если будете считать, что в одном двойном слове помещается 9 цифр (это оценка, а не точное значение).

Процедура 8 формирует положительное число. Если вводится отрицательное число, то знак надо исключить из буфера, а сформированное число инвертировать с помощью процедуры 4.

Процедура 8. Преобразование строки цифр в код ASCII в длинное число

```

PUBLIC      _bigcnv
ASSUME      cs:subr
subr SEGMENT word public 'subr'
        .486

; Размещение параметров в стеке:
; [esp+8] - адрес строки цифр в коде ASCII (Y)
; [esp+4] - адрес формируемого длинного числа (X)
;
_bigcnv proc far
mov esi, [esp+8] ; в esi адрес строки цифр в коде ASCII Y
mov edi, [esp+4] ; в edi адрес формируемого
                ; длинного числа X
xor ecx, ecx    ; ecx = 0, предварительная очистка
mov [edi], ecx  ; X(0) = 0, нулевой размер числа
inc dword ptr [edi] ; X(0) = 1, исходный размер числа
mov [edi+4], ecx ; X(1) = 0, очистка первого двойного слова
mov cl, [esi]   ; cl = Y(0), количество цифр в строке
; Начало внешнего цикла формирования длинного числа
nxtdig: xor ebx, ebx ; ebx = 0, смещение для указателей адресов
push ecx ; сохранение счетчика цифр в строке
mov ecx, [edi] ; ecx = X(0), текущий размер числа
mov edx, ebx ; edx = 0, очистка старшей части множимого
mov eax, ebx ; eax = 0
mov al, [esi+1] ; al = Y(j), код очередной цифры
sub al, '0' ; al = al - 30h, вычитание ASCII-кода нуля
; Цикл умножения формируемого числа на 10
lpmul: xchg eax, [edi+ebx*4+4] ; перестановка: eax = X(i),
                             ; X(i) = eax
mul _ddten ; edx:eax = eax*10
add [edi+ebx*4+4], eax ; X(i+1) = X(i) + eax,
                     ; запись результата
adc edx, 0 ; edx = edx + C, учет возможного
           ; переполнения
inc ebx ; ebx = ebx + 1, смещение
           ; для следующего слова
mov eax, edx ; eax = edx, пересылка старшей части
           ; произведения
loop lpmul ; управление повторами цикла
or edx, edx ; есть новый разряд у формируемого числа?
je @F ; -> нет
mov [edi+ebx*4+4], edx ; X(i+1) = edx, запись
                     ; в новый разряд
inc dword ptr [edi] ; X(0) = X(0) + 1, увеличиваем

```

```

; размер числа
@@: pop ecx ; ecx = текущее количество цифр в строке
inc esi ; esi = esi + 1, равносильно j = j + 1
loop nxtdig ; управление повторами внешнего цикла
ret ; возврат на вызывающий модуль

_bigcnv ENDP
_ddten dd 10 ; число 10, описанное как двойное слово
subr ENDS
END

```

Обычно для преобразования строки цифр в десятичное число создается цикл, в котором уже сформированная часть числа умножается на 10, и к произведению прибавляется очередная цифра. Процедура 8 формирует длинное число, поэтому умножение на 10 выполняет не одна команда, а внутренний цикл, имеющий метку `lpmul`.

Количество разрядов формируемого числа заранее не известно, изначально оно равно 1, а затем увеличивается на 1 после обработки каждой девятой или десятой цифры. В процедуре 8 к формируемому числу добавляется новый разряд, если при выходе из цикла умножения содержимое регистра `edx` отлично от нуля.

Выполнение процедуры продолжается до тех пор, пока не будут обработаны все цифры, находящиеся в буфере.

11. Заключение

Описанные в данной статье процедуры затрачивают если не минимальное, то близкое к нему время на получение результата, но, тем не менее, их выполнение можно ускорить почти в два раза. Для этого в них надо использовать 64-разрядные инструкции общего назначения, исполняемые современными моделями микропроцессоров (МП) Intel и AMD.

Первым 32-разрядным МП был Intel 386, выпущенный в 1985 г. С тех пор в семействе МП x86 произошли очень большие изменения, но они не коснулись инструкций общего назначения. Появились новые инструкции, имеющие 64- и 128-разрядные операнды, а операнды инструкций общего назначения остались 32-разрядными.

В конце сентября 2003 г. AMD представила первые 64-разрядные МП Athlon 64 и Athlon 64 FX. Кроме традиционных (наследственных) режимов работы МП x86, они поддерживают 64-разрядный режим, при котором инструкции общего назначения могут работать с 64-разрядными адресами и данными. При работе в новом режиме изменяются имена, количество (теперь 16) и разрядность (64) регистров общего назначения, а перед традиционными именами инструкций появляются префиксы, указывающие разрядность операндов. Поэтому описанные в данной статье процедуры несложно переделать для работы с 64-разрядными целыми числами.

Корпорация Intel не торопилась с ответом, ее первый 64-разрядный МП Xeon появился в конце июня 2004 г., а презентация 64-разрядного Pentium 4 состоялась только 25 февраля 2005 г. Intel назвала новую технологию EM64T, а AMD — x86-64. Руководство Intel заявило, что обе технологии совместимы на программном уровне и различаются в способе реализации. Таким образом, реально существует техническая база для работы в 64-разрядном режиме. Но, к сожалению, необходимое для этого системное программное обеспечение до сих пор либо отсутствует, либо доступно ограниченному числу программистов. Только по этой причине описанные процедуры работают с 32-разрядными целыми числами.

И последнее. Если вас интересует работа с длинными числами, то рекомендую прочитать в Internet статью И.А.Кантора "Длинные числа и операции с ними" (<http://algolist.manual.ru/maths/longnum.php>), описанные в ней вычислительные процедуры составлены на Си.

(Продолжение. Начало в №8/2005)

Соответственно, на текущем диске должно быть достаточно свободного места для хранения временного файла. В результате произведенных замен размер обработанных



Код выхода	Выводимое сообщение	Комментарий
1	Syntax: mreplace.exe [options] ini-file work-file (or @filelist)	Программа запущена без аргументов или с неправильным их количеством
2	Error in "имя файла": filename too long (line x)!	В указанном файле-списке, в строке x, длина имени файла (возможно, включающего полный путь к файлу) больше максимально допустимой (128 символов — определяется константой MAX_FN_LEN в тексте программы)
3	Error: can't delete temp file!	Ошибка при удалении временного файла
4	Error: sequence for replace not found!	Для последней искомой последовательности, указанной в ini-файле, не задана заменяющая ее последовательность
5	Error: empty ini-file!	В ini-файле не определено ни одной пары "искомая последовательность, заменяющая последовательность"
6	Error: line x too long!	Длина строки x ini-файла больше максимально допустимой (1024 символа — определяется константой MAX_STR_LEN в тексте программы)
7	Syntax error in line x, position y!	В ini-файла обнаружена синтаксическая ошибка при рассмотрении строки x, позиции y
8	Error: filename too long (line x)!	В ini-файле, в строке x, указано имя двойного файла (возможно, включающее полный путь к файлу), длина которого (имени, а не файла) больше максимально допустимой (128 символов — определяется константой MAX_FN_LEN в тексте программы)
9	Error: too many "?" in sequence for replace (line x)!	В ini-файле задана заменяющая последовательность, количество подстановочных символов "?" в которой больше, чем в соответствующей искомой последовательности. Это обнаружено при рассмотрении строки x ini-файла
10	Error: can't get current directory!	Ошибка при определении текущего каталога
11	Error: can't set directory "x"!	Ошибка при переходе в каталог x
12	Can't open file "имя файла" for read! (или "for write!")	Ошибка при открытии указанного файла для чтения (или записи). Возможно, такого файла нет, или указано длинное имя файла вместо короткого. Если ошибка при открытии файла для записи — возможно, диск с этим файлом защищен от записи, или для этого файла установлен атрибут "Read only" ("только чтение")
13	Close error!	Ошибка при закрытии файла
14	Read error!	Ошибка при чтении из файла
15	Write error!	Ошибка при записи в файл. Может возникнуть, например, при недостатке свободного места на диске
16	Out of memory!	Не хватает места в памяти (программа использует только "обычную" память — до 640 Кб). Возможно, поможет освобождение памяти от резидентных программ и драйверов, без которых можно обойтись. Или, если в ini-файле задано несколько пар "искомая последовательность, заменяющая последовательность", можно разбить этот файл на несколько файлов, в каждом из которых будет меньшее количество пар последовательностей, а потом по очереди запускать программу с каждым из них. Замечу, что иногда таков разбивание невозможно. Вот простейший пример: если в исходном ini-файле заданы две пары последовательностей, одна из которых заменяет 1 на 2, а другая — 2 на 1, то если разбить этот файл на два и применить эти файлы поочередно, результат замены будет совсем другим

Программа возвращает операционной системе код выхода: 0 — успешное завершение, а ненулевое значение свидетельствует об ошибке (сообщение об ошибке выводится на экран). Подробная информация о кодах выхода и ошибках приведена в таблице.

Если вы хотите запустить в многозадачной среде сразу несколько экземпляров программы, убедитесь вначале, что они не будут мешать друг другу, пытаясь обрабатывать одни и те же файлы. Также учтите, что программа в процессе работы создает временный файл с фиксированным именем в том каталоге, который был текущим при ее запуске, поэтому нельзя, чтобы при запуске разных экземпляров программы текущие каталоги совпадали.

Сказанное в предыдущем абзаце не относится к случаю, когда несколько экземпляров программы запущены в режиме "read only", в котором запись на диск не производится.

Если вам часто приходится последовательно применять несколько ini-файлов к некоторому обрабатываемому файлу (файлам), то удобнее не набирать вручную каждый раз соответствующие команды, а создать командный файл (bat-файл). Например, пусть в файлах, список которых содержится в текстовом файле "list.txt", надо последовательно выполнить замены, описанные в файлах "1.ini" — "6.ini". Тогда bat-файл будет таким:

```
mreplace.exe 1.ini @list.txt
@if errorlevel 1 goto end
mreplace.exe 2.ini @list.txt
@if errorlevel 1 goto end
mreplace.exe 3.ini @list.txt
@if errorlevel 1 goto end
mreplace.exe 4.ini @list.txt
@if errorlevel 1 goto end
mreplace.exe 5.ini @list.txt
@if errorlevel 1 goto end
mreplace.exe 6.ini @list.txt
@if errorlevel 1 goto end
:end
```

Если файл "mreplace.exe" расположен не в текущем каталоге и не в одном из каталогов, перечисленных в переменной окружения "path" (устанавливаемой в файле "autoexec.bat"), то перед именем файла "mreplace.exe" надо указывать полный путь к нему.

Обратите внимание: в bat-файле после каждой команды вызова программы "mreplace.exe" стоит команда "@if errorlevel 1 goto end" (символ "@" в ее начале нужен лишь для того, чтобы она не выводилась на экран при выполнении bat-файла). Эта команда проверяет, какой код выхода вернула программа "mreplace.exe", и если этот код больше или равен 1 (то есть при выполнении программы была какая-то ошибка), то происходит переход на метку "end". Так как эта метка находится в конце bat-файла, при переходе на нее выполнение bat-файла завершается.

Если бы bat-файл не содержал команд перехода при ошибке, то могла бы возникнуть следующая ситуация: при одном из вызовов программы "mreplace.exe" происходит ошибка (скажем, обнаруживается синтаксическая ошибка в соответствующем ini-файле), сообщение об ошибке выводится на экран, но при следующих вызовах программы "mreplace.exe" на экран тоже что-то выводится, поэтому сообщение об ошибке может оказаться сдвинутым за край экрана. Тогда после окончания выполнения bat-файла пользователь не увидит этого сообщения и будет думать, что никаких ошибок не было.

(Продолжение следует)



Р.КАРПАЧ,

E-mail: Commander-Norton@tut.by
www.fdd5-25.net

В DOS с нуля — что-то, где-то, когда-то слышал

Подробный рассказ об установке и работе в DOS для всех начинающих пользователей x86 компьютеров и просто неравнодушных downgrader'ов

Вместо преамбулы

“Ну, вот! Опять! Кому это сейчас нужно?!” — грозно скажет иной читатель, увидев заголовок этой статьи. Постараюсь сразу же успокоить — нужно!!! Если быть откровенным, мне не очень-то и хотелось возвращаться к теме DOS, вроде бы, все уже написано. Но, как говорят в народе: “Не тут-то было”. Очень часто, если не сказать, ежедневно, на электронную почту или в форум www.fdd5-25.net приходят вопросы о том, как установить DOS, что для этого нужно, что лучше — Windows 3.1 или 3.11, и так далее, и тому подобное. Не во многих изданиях могут компетентно ответить на такие неоднозначные вопросы. Все дело в том, что чуть ли не каждый автор компьютерных СМИ считает своим долгом продвигать “якобы” новые, более совершенные технологии, которые зачастую оказываются банальной рекламной ширмой для еще одного обольщения. При этом миллионы пользователей “морально устаревшей техники” отодвигаются на второй план, исходя из того, что новые Windows XP, Explorer, Офис, игры на машины 80-х — 90-х годов не установить. Ну и не нужно!!! Пора признать честно, что функциональность операционных систем и программ меняется не так быстро, как яркие “красивости” в играх и “софте”. Вместо того чтобы совершенствовать еще толком не использованные возможности одной технологии, производители перескакивают на другую. Яркий тому пример — AGP и PCI Express. При этом все СМИ дружно включаются в хор, хвалебно воспевающий: “ЭТО ВСЕМ НУЖНО НЕМЕДЛЕННО ПРИОБРЕСТИ, ВЕДЬ ЭТО КРУТО!” Но почему-то глаз повсеместно замечает DOS, WINDOWS 9x, NT 4.0 и 2000, вместо новых XP или LINUX. Вместо новейших версий программ, пользователи по-прежнему используют Word 6.0-2000 и подобные вещи. Их компьютеры вполне справляются с необходимыми задачами, возникающими при работе. А если нет разницы, зачем платить больше?

Но нет, нужно покупать 3-гигагерцовые процессоры и 100-гигабайтные винчестеры, чтобы играть в игры со старым сюжетом и еще более улучшенной картинкой, или забивать все банальными фильмами или порнографией? Это и есть итог компьютерного прогресса — большой развлекательный агрегат? Банальная коробка для развлечений? Для этого нужно было пройти путь от машины Бэббиджа до Athlon? Посмотрите на свой Pentium 4. Вы каждый день работаете в “ЮНИГРАФИКС”, который создает и просчитывает сложные конструкции, вы рассчитываете сложнейшие физические задачи, вы моделируете черные дыры и столкновения планет? Нет! Вы просто играете или слушаете музыку. А вот с банальной офисной работой вполне справится и Pentium 2 — 3. Вопрос ставится не “DOS vs Windows”, а “Стоит ли тратить деньги на новые программы и комплектующие, если и старые справляются со своей работой нормально?”. При осознании этого мы можем подойти к той черте, когда можно объяснить, для кого написана эта статья. Она, в первую очередь, для пользователей, которые пытаются всунуть Windows ME на 66-й или 166-й Intel и при этом, сидя в Internet, слушать MP3. Скажу откровенно, недостаток информации лишает эту категорию пользователей возможности полноценно работать с такими компьютерами. И дело не в невоз-

можности их применения в современных условиях, а просто в отсутствии информации. Поверьте, и 66-й Intel может многое. Одним словом — downgrade.

Загрузочная дискета DOS

Стандартная ситуация — есть некий компьютер класса 286...Pentium 2, из которого из-за отсутствия денег, нежелания покупать новый или просто из интереса нужно выдавить все возможное и невозможное. При этом, это ПК с абсолютно пустым жестким диском размером до 8 Гб. Что же делать? С чего начать?

В первую очередь следует установить MS-DOS 6.22. Почему именно его? Работа в DOS открывает доступ к огромным ресурсам программ и игр, написанных с начала 80-х до сегодняшнего дня, и именно эта ОС обеспечит вам возможность корректной работы в Windows 3.x — 95.

Для того чтобы установить DOS 6.22 на локальный компьютер, понадобится загрузочная дискета. (<http://fdd5-25.net/oses.php>). Ее можно создать, запустив инсталлятор DOS, а можно скопировать с уже готовой дискеты, после чего чуть доработать. Как? Очень просто. После создания инсталлятором DOS 6.22 системной дискеты, на ней появится куча разных файлов. К сожалению, большинство из них для нас будет бесполезно, так как зачастую DOS сейчас ставят на машины “с нуля”, и мало того, с компакт-диска, на который предварительно записана инсталляция. На системной дискете должны находиться лишь файлы:

- himem.sys — драйвер расширенной памяти;
- format.com — утилита форматирования дисков;
- fdisk.exe — утилита создания разделов на жестком диске (не работает с MFM-дисками);
- scandisk.exe — утилита сканирования диска на наличие ошибок;
- mscdex.exe — утилита опознавания CD-ROM;
- драйверы клавиатуры и мыши (<http://fdd5-25.net/drivers.php>);
- autoexec.bat и config.sys — системные файлы, которые вы создадите сами;
- оболочка Norton Commander 1.0 — 3.0 (<http://fdd5-25.net/shells.php>);
- драйвер CD-ROM, тот что необходим — IDE, MITSUMI, PANASONIC (<http://fdd5-25.net/drivers.php>);
- io.sys, msdos.sys, command.com — эти не подлежат удалению ни при каких обстоятельствах, именно они хранят святая святых — DOS!

После того как перечисленные файлы окажутся на вашей загрузочной дискете, можно приступать ко второй части работы — конфигурированию загрузочной дискеты. Здесь важную роль играют всего два файла — autoexec.bat и config.sys. В них вы должны будете сами, своими руками, указать расположение нужных программ для запуска и применения. Все должно выглядеть именно так:

Autoexec.bat

@ECHO OFF

A:\MSCDEX.EXE /S /d:mscd001

A:\keyrus.com — драйвер кириллицы в DOS, может быть и другим;

A:\mouse.com — драйвер мыши;

A:\NC\nc.exe — запуск Norton Commander (должен находиться на дискете в папке NC).

Config.sys

DEVICE=A:\himem.sys — драйвер расширенной памяти;
DEVICE=A:\mscdrom.sys /d:mscd001 — сам драйвер CD-ROM, который скачан с fdd5-25.net.

LASTDRIVE=Z
NUMLOCK=OFF

DOS=UMB

DOS=HIGH

Все вышеописанные файлы должны легко умещаться даже на дискете 5"25 (720 Кб), что будет обязательно необходимым условием для владельцев машин 8088...80386.



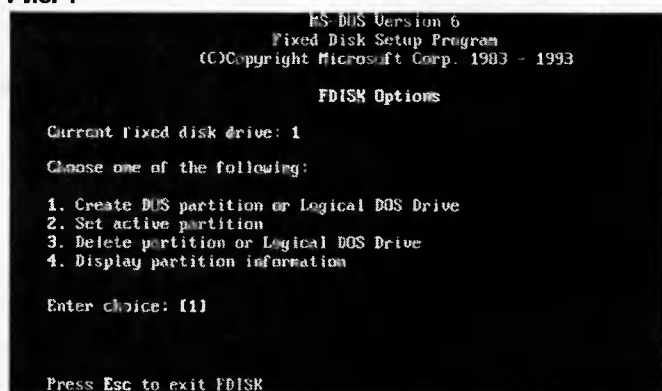
Если все вышеуказанные рекомендации выполнены верно — значит, загрузка должна пройти без проблем.

Конфигурирование HDD для работы с DOS

Для создания разделов на вашей загрузочной дискете, как я уже писал, есть утилита fdisk.com. Как же ею воспользоваться, ведь это же не Partition Magic (шутка ли)?! Отсутствие графического интерфейса еще не означает полную бесполезность программы, это вы заметите при дальнейшей работе в DOS (адресовано тем читателям, которые по той или иной причине впервые слышат об FDISK).

Чтобы получить полную и исчерпывающую информацию о вашем HDD, необходимо запустить утилиту FDISK. После этого она войдет в так называемый диалоговый режим, где отобразится черное страшное главное меню программы (рис.1).

Рис. 1



Здесь следует выбрать пункт 4 "DISPLAY PARTITION INFORMATION". В нем можно ознакомиться с характеристиками жесткого диска или дисков:

- номером раздела;
- статусом активности;
- типом раздела;
- меткой тома;
- объемом раздела в мегабайтах;
- типом файловой системы.

Активным будет являться тот раздел, с логического диска которого будет происходить загрузка. Такой раздел один и отмечается буквой "А". Чтобы загрузка была возможна, активным должен являться первичный раздел DOS.

Что же делать нам? В большинстве случаев, о которых мне писали пользователи, на компьютерах не было нормально установленной системы, так как машины были собраны из даром отданных комплектующих, либо лежали, законсервированные в небытие. Есть, правда, и третий вариант — желание downgrade'ра снести, непонятно зачем, установленную на Pentium 100 Windows 98 или что-то подобное. В любом случае — вся нужная информация с жесткого диска, если таковая имела, давно находилась на компактках или дискетах.

Для того чтобы создать разделы DOS, нужно уничтожить neDOS разделы, если таковые имеются. Чтобы это сделать, производим следующие манипуляции:

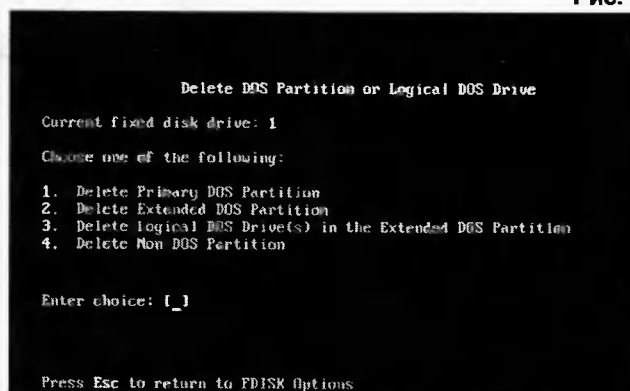
- выбираем цифру "3" и переходим в раздел "DELETE PARTITION OR LOGICAL DOS DRIVE" (рис.2);
- выбираем опцию "DELETE NON DOS PARTITION". К ним относятся все разделы DOS выше версии 6.22, Linux, OS/2, QNX, Miraculix и так далее. Следуем приведенным там инструкциям;
- если на диске более одного раздела (D:, E:, F:), ниже и включая DOS 6.22, тогда вначале заходим в раздел №2 "DELETE EXTENDED DOS PARTITION". Далее следуем инструкциям;
- после удаления разделов заходим в раздел 3 "DELETE LOGICAL DOS DRIVERS", где благополучно удаляем все разделы, кроме C:;
- удаляем основной раздел в опции 1 и перезагружаем компьютер для вступления параметров в силу.

И вот перед нами чистый жесткий диск, не имеющий никаких разделов и данных. Что нужно делать дальше? А вот что:

- запускаем все тот же FDISK;
- выбираем пункт 1 "CREATE DOS PARTITION OR LOGICAL DRIVE";
- в появившемся окне выбираем пункт 1 "CREATE PRIMARY DOS PARTITION".

После проведения вышеописанных манипуляций должен возникнуть диалог, где предлагается указать количество мегабайтов, которые вы хотите отдать под будущий диск C:. Обычной практикой является создание 2 и более логических разделов. Для этого нужно указать конкретный размер каждого раздела. Например, общий объем диска 2048 Мб, значит, диску C: можно отдать 1024, а остаток FDISK просчитает автоматически.

Рис. 2



Сразу же после ввода размера раздел создается. Его характеристики появляются в таблице наряду с первичными характеристиками раздела DOS. Для формирования логических дисков в расширенном разделе требуется:

- выбрать пункт "CREATE LOGICAL DOS DRIVERS IN THE EXTENDED DOS PARTITION" из меню "CREATE DOS PARTITION OR LOGICAL DRIVE";
 - после отображения сообщения ввести размер логического диска в мегабайтах;
 - повторить действия столько раз, сколько нужно логических дисков;
 - вернуться в главное меню путем нажатия клавиши "ESC".
- Далее нужно назначить активным разделом раздел 1:
- в главном меню FDISK выбрать №2 "SET ACTIVE PARTITION";
 - по умолчанию должен быть выбран — 1.

После этого выходим из FDISK и перезагружаемся с загрузочной дискеты.

Следующим шагом должно стать форматирование разделов. Это несложная процедура.

Для формата диска C: введите команду `FORMAT C: /U /C`. Если же в системе присутствуют другие логические диски, то их нужно отформатировать командой `FORMAT D: /U /C`. Ключ /S записывает системные файлы на диск, /U — предписывает безусловное форматирование, /C — проверяет качество записи секторов.

Установка MS-DOS 6.22

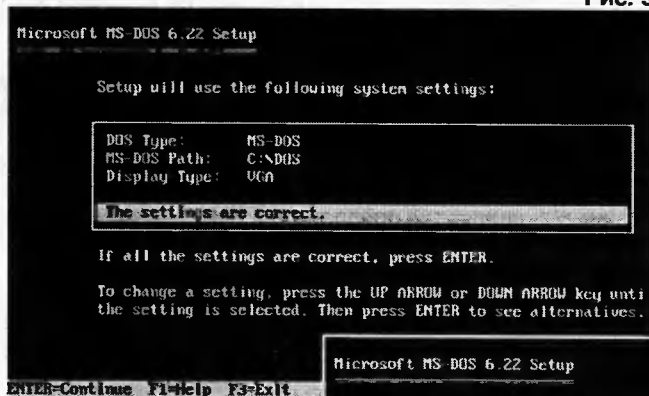
Чтобы произвести установку операционной системы, нужно иметь ее дистрибутив. Обычно MS-DOS 6.22 поставляется на компакт-диске или дискетах. В любом случае установка будет проходить одинаково, конечно же, с поправкой на то, что версия на дискетах будет создавать много хлопот (автор не любит часто доставать и вставлять дискеты).

Производимые действия:

- вставить CD-ROM с MS-DOS 6.22 в ваш привод, либо дискету под номером один в дисковод;



Рис. 3



- запустить файл setup.exe, после появления диалога нажать ENTER.

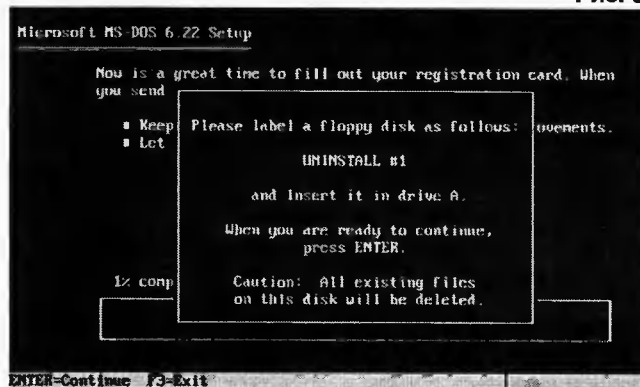
После двух нажатий клавиши ENTER у вас появится меню для указания характеристик установки (рис.3).

Обычно все оставляют по умолчанию, как есть, за исключением тех случаев, когда тип монитора является не VGA. Хотя я не заметил никакой разницы на EGA-мониторе при установленном VGA.

Следующий диалог (рис.4) будет настоятельно требовать от вас указания тех приложений, которые нужны на вашем компьютере. Думаю, здесь все более или менее понятно.

Если вы установите Windows 3.X или выше до установки DOS, то операционная система потребует указать точное расположение оболочки на диске.

Рис. 5



Одним из важных недостатков инсталлятора DOS 6.22 является его мания создания собственных копий — загрузочных uninstall-дискет (рис.5). К сожалению, перед тем как устанавливать систему, всегда нужно иметь запасную пустую дискету 5,25" или 3,5".

По окончании копирования файлов, MS-DOS 6.22 оповестит вас о том, что система уже установлена, системные файлы изменены, а вам следует вынуть загрузочную дискету из дисковода и перезагрузить компьютер. Что я пока делать не советую! Все дело в том, что при загрузке с жесткого диска, вы окажетесь в чистом DOS, вернее, в command.com. А ведь, согласитесь, это неудобно, тем более, многие не знают консольных команд. Поэтому для облегчения работы я предлагаю использовать программу-оболочку, скажем, все тот же Norton Commander с вашей дискеты. Что же нужно сделать? Ваши действия:

- скопируйте оболочку в каталог NC на диске C;
- скопируйте драйвера CD-ROM, mouse.com, keyrus.com в корневой каталог;

- отредактируйте файлы CONFIG.SYS и AUTOEXEC.BAT на диске C:

autoexec.bat

@ECHO OFF — команда отключения эха отображения команд;
C:\DOS\SMARTDRV.EXE /L /X — запуск дополнительного кэша диска;

C:\DOS\MSCDEX.EXE /D:MTMIDE01 /M:10 — инициализация CD-ROM;

C:\keyrus.com — загрузка драйвера кириллицы;

C:\DOS\mouse.com — загрузка драйвера мыши;

PROMPT \$p\$g — изменяет вид командной подсказки;

PATH C:\DOS;C:\NC — указывает,

Рис. 4

в каких каталогах DOS должна искать исполняемые файлы;
rem keyrus — позволяет включать комментарии в командный файл;
NC — запуск NORTON COMMANDER.

config.sys

DEVICE=C:\DOS\SETVER.EXE — позволяет запускать приложения ранних DOS-версий, выдающих сообщение "INCORRECT DOS VERSION";

DEVICE=C:\DOS\SMARTDRV.EXE /DOUBLE_BUFFER

DEVICE=C:\DOS\HIMEM.SYS

— драйвер расширенной памяти;

DEVICE=C:\DOS\EMM386.EXE RAM — обеспечивает доступ к старшим адресам памяти;

REM BUFFERS=30,0

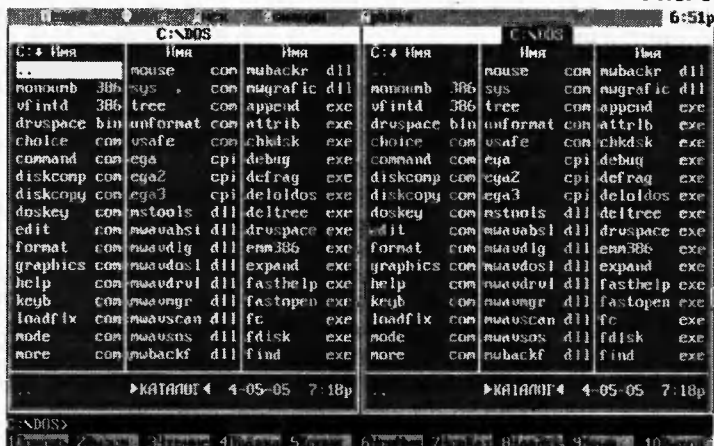
FILES=40

DOS=UMB

LASTDRIVE=Z

FCBS=4,0

Рис. 6



DOS=HIGH

[COMMON]

DEVICE=C:\MTMMTCDAI.SYS /D:MTMIDE01

Если все предписания будут выполнены верно, то на экране монитора радостно и весело замигает синие панели Norton Commander (рис.6), а вы, как пользователь, откроете для себя новые широкие горизонты программного обеспечения под DOS. Напомню лишь еще одну вещь — при желании, из MS-DOS 6.22 можно выйти в Internet, создать свой web server, смотреть видео DivX, слушать музыку MP3, работать с офисными приложениями, создать локальную сеть, и так далее, и тому подобное. Знайте, что не следует втискивать на 100-й Pentium Windows 98 только из-за того, что, якобы, DOS ничего не может. Не заблуждайтесь. О широких возможностях морально устаревших ПК и downgrade я неоднократно писал в журнале "Радиомир. Ваш компьютер" и, конечно же, на www.fdd5-25.net. Дерзайте, все в ваших руках!



Обзор электронной прессы о Spectrum

(Окончание. Начало в №№6-8/2005)

А.ПЕХИМЕНКО,
E-mail: paalex@inbox.ru

Наименование	Тема
Online 29	Биполярный ЦАП, схема Soup Drive на м/с BB55 и ПА1
Online 38	Усилитель к АЦП на м/с УД6
Online 39	Схема светового пистолета
Online 79	Режим 512*192 точки для компьютера Pentagon
Online 80	Описание дополнительного графического экрана
Online 27	Схема АЦП через AY
Online 28	Схема Ковокс
Online 29	Биполярная ЦАП, схема Soup Drive-1
Online 32	Подключение процессора от Dendy к Spectrum
Online 33-37	О процессоре Dendy
Online 38	Схема усилителя для АЦП
Online 39	Схема светового пистолета
Online 56	Распайка разъемов CGA-, EGA-, VGA-мониторов, подключение к Spectrum CGA-, EGA-, VGA-мониторов
Online 58	О разработках ZX Scorpions
Online 59	Проблема цвета в EGA- и CGA-мониторах
Online 78	Схема последовательного порта, описание компьютера ATM
Online 79-80	Схема режима 512*192 точки для Pentagon
Online 81-83	Описание компьютера TURBO 2+
Online 88	Схема контроллера последовательного порта на м/с BB51
Online 78	Описание портов ATM-TURBO v4 1-4 50
Online 79	Схема режима 512*192 точки для компьютера Pentagon
Online 83	Описание порта TURBO 2+ v7.0-7 1
Polesse 16	Схема расширения памяти "Таганрога" до 1 Мб, схема гомельского C-DOS-модема
Polesse 11	Разгон компьютера "Байт"
Polesse 14	Схема аппаратной паузы, схема квази-турбо
Polesse 16	Расширение компьютера Таганрог до 1 Мб
Promiceland2	Как правильно паять, изготовление печатных плат, схема подмены ПЗУ на CASH, распиновка м/с CASH, схема Kempston mouse, переделка мыши
Proton 1	Подключение монитора с TTL-входом к компьютеру GRM
Proton 06	Дешифратор АОН, схема подсветки выключателя, схема дежурного освещения
Proton 08	Выключатель модема, импульсный номеронабиратель
Proton 11	Схема переделки контроллера дисководов для чтения на IBM дискет Spectrum
Proton 13	Подключение принтера, разводка разъема
Proton 22	Доработка C-DOS-модема — улучшение качества приема
Proton 26	Доработки Beta disk-интерфейса, аппаратная остановка неактивных дисководов, доработка Beta disk интерфейса — дисководы C и D, выбор FM и MFM, разводка м/с AY8910
Proton 29	Описание м/с 1533TM2, 1533TM7, 1533TM8, 1533TM9 и их зарубежные аналоги
Proton 30	Доработка C-DOS-модема
Proton 31	О доработке C-DOS-модема
Proton 35	Доработка ПК Scorpions — сигналы ЮRQGE, RDR
Proton 39	Доработка компьютера GRM
Proton 48	Аналоги зарубежных микросхем
Rip 2	Схема преобразователя +5 В — +12 В для м/с ВГ93
Realtime 1	О ПК "ZX-пехт"
Realtime 2	Совок, виды винчестера, доработки компьютера Scorpion, схема LPT COVOX, доработка порта #FD в ZS SCORPION-256

Rush 01	Схема 128 цветов для компьютера Pentagon, схема блокировки порта клавиатуры при подключении C-DOS-модема, м/с — потомки Z80
Rush 02	Схема подключения CGA-монитора
S club 4	Защита м/с ВГ 93
Sinclair town	м/с 1818ВГ93, блок-схема контроллера дисководов
Shock 1	Подключение Hayes-модема к компьютеру ZS-Scorpion-256
Shocker	Схема подключения IBM-карт к Spectrum
Specinfo2-4	Описание платы GMX
Specrofoxy-5	Головка сканера
Sp-prog 1	О ПК "Спринтер", схема модема Майорова
Specsy 3	Схема выключателя уличного освещения
Specsy 6, Specsy 7	Сканер
Stown 2	БИС KP1818ВГ93
Sp-fon-06	Схема подключения AY
Sp-fon-10	Турбирование м/с 1818ВГ93
Sp-fon-12	Исправленная и доработанная схема турбирования м/с 1818ВГ93
Sp-fon-16	Универсальная схема дешифрации порта расширения памяти для машин с дополнительной памятью Profi и Scorpion
Sp-fon-17	Порт #DFFD — #7FFD, схема расширения ОЗУ до 512 Кб, схема псевдо-ПЗУ
Sp-fon-18	Несколько полезных разработок, схема АЦП, об улучшении связи по модему
Sp-fon-20	Схема Kempston mouse
Sp-fon-21	Описание контроллера SMUC
Spectrum expert-1-ZX, Spectrum expert-2-ZX	Bus-советы
Seet 1	Кодовый замок с емкостной памятью, доработка АОН
Target 10	О ТВ-стандарте Spectrum
Think 29 5	Описание и программирование General Sound, описание и схема подключения м/с CMOS-часов
Think 42	Схема Vicomm 2-модема
Think 43	Схема Kempston mouse, подключение буфера Hayes-модема
Think 44	М/с 61256, схема проецирования расширенной памяти в любое окно
Think 45	М/с 4M2612 из Sega
Ultimatum 1	Описание Vicomm-модема, плата с расположением деталей, о Soup Drive
World 14	Подключение м/с SIMM к Spectrum
Voyager 1	Автостарт TR-DOS a Pentagon, о подключении General Sound к компьютеру Profi, системный разъем компьютера Scorpion, Profi
Voyager 2	Схема Vicomm-модема с подъемом трубки
X-RAY 03	Схема и описание аппаратного мультиколора
Voyager 4	"Спринтер 2", схема дешифрации портов компьютера Profi, о подключении Hayes-модема
Vw1	Перекрытие клавиш, устранение глюка
ZX FE 01, Hardware	Multicolor — байт пикселей на байт атрибута для Pentagon, Profi по порту #EFF7, ZXPAD
ZX FE 02	Как попасть в 4-ю страницу ПЗУ на Pentagon, работа с HD-дисками, схема блокировки критичных портов при короткой адресации, схема усилителя на м/с KA2212 (712, 7312) на 3 Вт, проект миди-интерфейса, CHROME — итальянский клон Spectrum, проект ZXPAD
ZX Format 1	Тормоз для ПК KAY, отключатель порта #1FFD



ZX Format 2	Переключатель Turbo/Normal в компьютер Scorpion
ZX Format 3	General sound, сравнение скорости компьютеров Scorpion и KAY, архитектура шины, обзор типов модемов, глюки TR-DOS
ZX Format 5	General sound, XTR vodem, схема Kempston mouse, переделка мыши, схема LPT — Covox
ZX-Format 6	Multicolor
ZX-Club 04	Схема Профи, Covox
ZX-Club 07	Схема расширения компьютера "Синтез-128"
ZX-Club 09	Расширение памяти компьютера Pentagon до 512 Кб, схема Kempston mouse
ZX Guide 2	Режим 384*304 точки на Pentagon, 3,5 — 1,75 МГц на AY8910, схема ABC-стерео на AY8910, схема подключения AY8910
ZX Guide 4,	PAL-coder, Secam-coder, NTSC-coder, АТ-клавиатура
ZX Guide 4, 5	Дешифрация портов, PAL/NTSC-coder
ZX hard 00	Советы начинающим — как правильно паять, схема пробника для проверки плат, работа с сигналом IORQGE/
ZX hard 01	Схема последовательного порта для Spectrum, логические схемы, схема разъема принтера, проект OSP — звукоаппаратура
ZX hard 02	Как травить плату, ЦАП, схема простого усилителя на м/с KA2209 (P= 3 Вт), логические элементы
ZX light 10,	Схема стереоусилителя
ZX light#0B	Доработанный Vicom 3+-модем и как его подключить к компьютерам Pentagon и Profi
ZX light#0E	Расширение памяти компьютера "ZS Scorpion 256" до 1 Мб
ZX magazine1	Схема LPT — Covox
ZX-news 02	Распайка штекеров подключения Vicom-модема к Spectrum
ZX-news 13	Схема Vicom-модема с подъемом трубки
ZX-news 20	Тормоз для ПК
ZX-news 21, 24	Ремонт компьютера Scorpion
ZX-news 23	Подключение 3,5"-дискета
ZX-news 32	О компьютере "Спринтер"
ZX-news 35	Схема и описание CMOS-часов
ZX-news 38	Схема подключения внутреннего Hayes-модема

ZX-news 41	Схема подключения внутреннего Hayes-модема с исправленной ошибкой
ZX-pilot 14	Схема турбирования м/с BF93 для компьютера Pentagon
ZX-pilot 27	Схема гашения яркости при воспроизведении черного цвета
ZX-pilot 30	Улучшение звука AY а компьютере Pentagon, схема подключения CGA-монитора
ZX-pilot 31	Схема, плата устройства видеопорта, описание работы, настройки, текст обслуживающей программы, о плате GMX — подсоединение и доработки
ZX-pilot 35	Схема подключения AY
ZX-pilot 37	Схема формирования INT
ZX-pilot 39	Прошивки вместо 128-го Бейсика (EWS, GRS, RC, NeOS, Madrom, QC), схема включения 4-й страницы ПЗУ, схема турбирования контроллера дискета, схема 128 цветов в компьютер Scorpion, соответствие м/с "желтой" и "зеленой" плат компьютера Scorpion
ZX-pilot 40	Схема расширения памяти компьютера Solon Pentagon до 256 Мб
ZX-power 1	Схема AY-mouse
ZX-power 2	Схема контроллера последовательного порта, усиление сигнала на оптопарах мышей, микропроцессоры семейства Z80, схема режима 512*192 точки с описанием
ZX-power 3	Схема подключения Hayes-модема
ZX-power 4	Подключение модемов
ZX-Guide 2	Схема экрана 380*304 для компьютера Pentagon, схема DMA Sound blaster
ZX Rules 1	Схема T-REX'S Device 1.2 для компьютера Профи-псевдо ПЗУ
ZX-TIME 3,	Автономное питание от аккумулятора
ZX-TIME 4,	Аппаратный мультикопир
ZX-TIME 8,	Схема привязки видеосигнала к уровню черного
За рулем 00	Схема правильных и неправильных подключений AY, о видеопорте
ZX-ZONE 2	Схема доработки кнопки Magic в компьютер Pentagon, схема Gigascreep для компьютера Pentagon
#Z-80 01	Схема Kempston mouse на м/с КП11
#Z-80 05	Схема подключения модулей памяти Simm к Spectrum

Mr.Gluk Reset Service 6.0R

Прошивка ПЗУ для ZX Spectrum

Д.БЫСТРОВ (Alone Coder),
г. Рязань.

1. ПЗУ

В некоторых моделях ZX (например, Pentagon 128) ПЗУ имеет 16 Кб свободных ячеек, которые не задействованы по схеме (0-я четверть, 27512). Исправить этот недочет можно простой доработкой, не добавляя на плату новые элементы [1]. Тогда при сбросе будет организовываться передача управления в эту область ПЗУ, а выполнение команды в ОЗУ — закрывать доступ к ней.

Помимо различных тестов и прототипов операционных систем, существует несколько сервисных прошивок для этой странички. Если не производить доработку, можно использовать страницу ПЗУ Basic 128 (2-ю четверть, 27512), поскольку Basic 128 совершенно бесполезен.

Сервисные прошивки предназначены для расширения функций машины без загрузки со съемного носителя. Извест-

ны прошивки Mr.Gluk Reset Service (далее — Gluk), MadRom [2] и RainRom [3]. Первая из них разработана Р.Мамедовым и Р.Гавриловым (в Fido зарегистрирован как Andrey Isaev. Информация о реальном имени получена от В.Березенко) из г.Иваново, существует, вероятно, с 1996 г. (в июне 1997 г. уже была выпущена версия 5.0). В 2000 г. она "перешла" в мои руки и многократно обновлялась. Две другие прошивки имели всего по 2 известных версии и "сошли с дистанции" соответственно в 1999 и 2002 гг.

Немного статистики. Опрос, проведенный в начале 2005 г. на сайте zx.rk.ru (15 человек), показал, что:

- только трое из пятнадцати имеют сервисную прошивку — но не Gluk, хотя у всех, кроме троих, есть четыре и более страниц ПЗУ;
- у восьми человек 48k-прошивка

имеет поддержку Basic 128, у четырех — "чистая" или дополненная чем-то другим версия, остальные трое не интересовались, что у них прошито;

- только у 7 человек в Basic 48 исправлен обработчик NMI (ячейка #006d должна содержать код #28), 4 человека не знают, исправлено или нет;

- только у троих (из 13 имеющих Basic 128) выход в TR-DOS организован верхним пунктом;

- у 8 (!) человек TR-DOS имеет версию 5.03, только у троих — 5.04 и всего у одного — 5.11-13! Версию 6.05e не потрудился прошить никто из опрошенных;

- у 5 человек в дополнительной страничке ПЗУ прошит теневой монитор, у двоих — тест, у одного — CP/M, у одного — своя прошивка, командер же не прошит ни у кого;



- только 5 человек настраивали свою прошивку самостоятельно, из них 4 писали для этого свой код;

- и, к крайнему сожалению, только 3 человека могут перешить ПЗУ у себя дома (у 2 из них Flash ROM), у 2 (один из них в числе предыдущих) компьютер может полноценно заменять ПЗУ с помощью ОЗУ, 7 человек могут перешить за деньги, 3 — не могут и 1 — не хочет.

Описываемая версия [4] выпущена в марте 2005 г. вместе с полным исходником. К прошивке прилагается адаптированная TR-DOS v6.09e вместе с исходником изменений относительно TR-DOS v5.04T. На Пентагоне порядок страниц ПЗУ такой: 0 — Gluk, 1 — TR-DOS, 2 — Basic 128 (либо Quick Commander [5]), 3 — Basic 48 (рекомендуется исправить содержимое ячейки #006d на #28 для правильной работы NMI).

2. Устройства

Gluk работает и на стандартных 128k-машинах, но дополнительно поддерживает:

- CMOS-часы по схеме P.Мамедова (порты #eff7, #dff7, #bff7) [6] (в этой статье устаревшая раскладка портов. Исправленное описание и схема прилагаются в комплекте прошивки) — используются как для показа времени, так и для сохранения настроек, см.п.5. Редактор ОЗУ часов встроен в прошивку, см.п.4 (S);

- Kempston Mouse (порты #fadb, #fbdf, #ffdf) с левой кнопкой на D0 (стандарт Create Soft [7], принятый в эхо-конференциях REAL.SPECCY и ZX.SPECTRUM [8]). Ролик и средняя кнопка не задействованы. Поддержку можно отключить клавишей 8;

- музыкальный сопроцессор (порты #fffd, #bffd) — можно проверить звук всех 3 каналов, см.п.4 (D);

- Covox (ЦАП) по ZX Lprint III (порт #fb) — можно проверить в цветовой таблице, см.п.4 (C);

- Cache 32k по ZX Lprint III (порты #7b, #fb) [9] — есть выходы туда. Можно запускать оттуда сам Gluk;

- дополнительные 64 Kб ПЗУ (подключение по сбросу, отключение по команде в ОЗУ, предполагается, что Gluk в предпоследней странице ПЗУ) — переход по сбросу с горячими кнопками;

- функции очистки RAM-диска настроены на расширение памяти — 1 Mб по порту #7ffd. Если этой памяти нет, то предполагается, что XAS (или дру-

гой теневой ассемблер, у меня — ALASM 5.x [11]) и STS (см.п.5) лежат в #11-й и #17-й страницах соответственно;

- защелка памяти свыше 128 Kб (2 бита порта #eff7);

- HDD по Nemo (порты #10, #11, #30, #50, #70) [10] — для загрузки с HDD одной кнопкой. Это позволяет почти неограниченно расширять функции;

- при входе в Gluk микросхема BB55, если она есть в машине (порт #7f), инициализируется на вывод по всем портам.

Параметры устройств можно изменить в исходниках, инструкции по сборке прилагаются в комплекте.

3. Память

Сброс не очищает память, чтобы не уничтожить лежащие в памяти программы или их резидентные. Это позволяет при написании программ не перезагружать среды разработки и исходники. На практике, отлаживая программы и подготавливая файлы для них, кнопку сброса приходится нажимать много десятков раз в день.

Это относится к страничной памяти. Нижняя память (системные переменные и буфера, необходимые интерпретатору Бейсика) инициализируется таким образом, чтобы запускаемые программы не требовали Basic 128. При запуске программ из встроенных boot'a и командера дополнительно устанавливается RAMTOP=65367, UDG=65368, а 0-я страница ОЗУ подвергается очистке. При выходе в XAS и STS никакой очистки не происходит. Более того, восстанавливается содержимое основного экрана. Оба экрана постоянно хранятся в памяти и могут быть записаны через опцию "saVe scr." (выделенная буква соответствует горячей кнопке. Практически все функции Gluk имеют горячую кнопку).

Распределение памяти:

- #5fdx...#9bd3 — Gluk и его стек;
- #9bd4...#9bff — состояние стека и регистров при сбросе;
- #9c00...#a4ff — буфер каталога и "уменьшалки" экрана;
- #a500...#bff — захваченный 5-й экран.

4. Сброс

При нажатии кнопки сброса происходит небольшая задержка (4000 тактов — для подавлениядребезга кнопки Reset), в нижней памяти ищется ALASM (если найден — перебрасывается в #51-ю страницу), а далее — опрос клавиш:

- Space — выход в TR-DOS. TR-DOS v6.09e проверяет наличие в памяти резидентов Honey Commander'a и других программ;

- H — переход в 0-ю (4-й бит порта #7ffd сброшен) страницу кэша по адресу #60. Там должны быть команды пор:por:por bc,af:inc sp,sp. Таким образом сохраняются все регистры, и в кэше можно разместить отладчик;

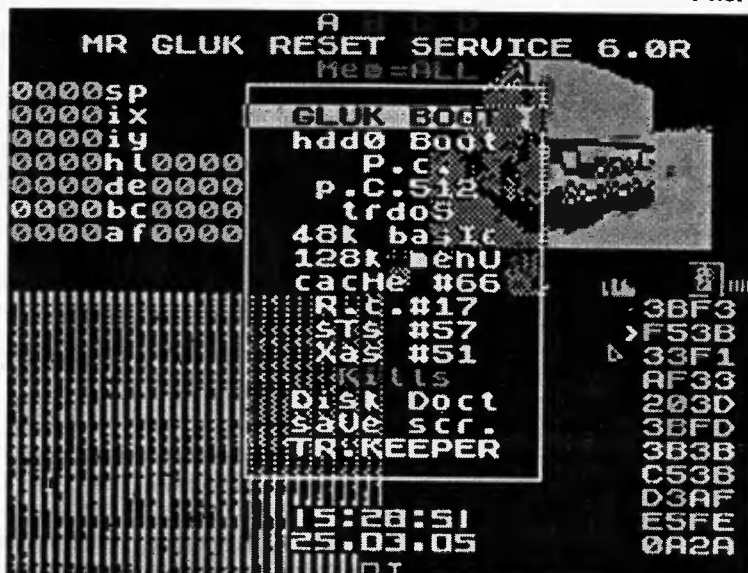
- C — цветная таблица. Сверху — все возможные цвета с flash, в середине — все чистые цвета, внизу — наложение цветов в сетке. На границе средней и нижней трети можно проверять равенство яркости чистых темных и чересеточных ярких красок. На Covox выдается звук. Любая крайняя клавиша (например, Enter или Space) — изменение цвета бордюра;

- D — демонстрация "Grass", автор — С.Просвилов (PSB^Halloween). Можно проверить звучание всех каналов AY, они включаются один за другим;

- S — редактор ОЗУ CMOS-часов. Действуют стрелки и цифробуквы. Изменения сразу заносятся в CMOS;

- CS — переход в STS, лежащий в странице #57;

Рис. 1



- 0, U, SS, 5, 7 — выход в страницы ПЗУ #0, 2, 4, 5, 7 соответственно. Для #0 и 4 надо иметь в #3d2f этих ПЗУ байты пор:ret. Для #5 нужно не только наличие этих байтов в #4, но и команда out (c), а в той же странице #4 по адресу #60. Первые два выхода — по адресу #0, остальные — по адресу #62 (все регистры сохраняются, если там пор bc,af:inc sp,sp. Для #4 сохраняется и номер включенной страницы ОЗУ);

- любая клавиша из оставшихся — встроенная подсказка;



- если клавиши не нажаты — переход в меню.

5. Меню

Экран меню показан на рис.1.

Рассмотрим элементы меню в порядке следования сверху вниз:

- A B C D — выделен текущий диск-код. Выбор производится мышью или кнопками 1...4;

- сообщение — может быть заменено надписью из ОЗУ часов. Для этого #11-я ячейка ОЗУ часов должна содержать #AA, а #12-я — #47. Сообщение начинается с адреса #13 и заканчивается байтом #ff, длина — не более 27 байтов;

- Mem=ALL — режим памяти (48k/128k/ALL), учитываемый при запуске из boot, выходе в 48k Basic, P.C., R.C. и TR-DOS. Выбор режима производится мышью или кнопкой M. Текущая настройка сохраняется в CMOS-часах, если они есть и задействованы (переключается клавишей 9);

- GLUK BOOT — boot без выхода из оболочки, автоматически определяет смену диска;

- hdd0 Boot — загрузка по адресу #6000 (и запуск оттуда же) #6000 байт с 3-го сектора 0-й дорожки 0-й стороны Master HDD, подключенного по схеме Nemo;

- P.c. — Perfect Commander v2.0 со 128 Кб памяти. Имеет встроенный текстовый редактор;

- p.c.512 — он же без ограничения памяти;

- trdos — выход в TR-DOS (версии 5.1xf или 6.0xe, где x — произвольная цифра) с запуском либо резидента, либо boot;

- 48k basic — выход в Бейсик через RST 0;

- 128k menU — выход во 2-ю страницу ПЗУ по адресу 0;

- cache #66 — выход в 1-ю (4-й бит порта #7ffd установлен) страницу кэша по адресу #66;

- R.c.#17 — выход в Real Commander 1.x или Quick Commander по сигнатуре #31,#ff в начале #17-й страницы ОЗУ;

- sTs #57 — выход в STS по адресу #db00 в #57-й странице (для 128k-машин — в #17-й);

- Xas #51 — выход в ассемблер по

адресу #c000 в #51-й странице (для 128k-машин — в #11-й);

- Kills — см. §6;

- Disk Doct — дисковый доктор. При входе нужно ввести 3 шестнадцатеричных цифры дорожки-сектора. Измененные байты помечаются инверсией, их можно восстановить клавишей Delete (cs/0). Сдвиг информации ниже курсора в разные стороны — ss/Q, ss/W. Сохранение сектора — cs/Enter. Выбор другого сектора — L. Выход — Break;

- saVe scr. — можно посмотреть экраны и сохранить их на диск клавишей Enter;



- TR.KEEPER — меню сохранения, просмотра и восстановления каталога на 161 дорожке. Дорожка перед записью форматируется.

- правый верхний угол — уменьшенная копия основного экрана, левый нижний — дополнительный;

- левый верхний угол — регистры процессора в момент сброса, правый нижний — содержимое стека в этот момент. Знак ">" соответствует SP.

Предусмотрена возможность заменить GLUK BOOT опцией, обработчик которой лежит в 1-й странице кэша (вызывается по адресу 0). Наличие этой опции определяется так: #18 по адресу 0 и строка "GLUK" по адресу 2. Имя опции лежит по адресу 6 и занимает 9 байтов (последний символ — с установленным 7-м битом). Также организована замена надписи "MENU128" на надпись из ПЗУ 128-го меню, если по адресу 0 опять-таки присутствует #18, а по адресу 2 — опять-таки "GLUK". Имя опции тоже лежит по адресу 6 и занимает 9 байтов — и последний символ опять с установленным 7-м битом.

6. Подменю Kills

Экран данного подменю показан на рис.2.

- Low mem — очищает память в диапазоне адресов #5b00...#5fdx. Исходно она находится в том состоянии, в котором ее застал сброс (не считая системных переменных). Резидент ALASM стирается этой же опцией, изначально он не поврежден;

- Pages — очищает страницы #10, #11, #13, #14, #16, #17;

- ALL PAGES — очищает всю память: сначала 512 Кб по маске #c7 в порту #7ffd, потом, если получится, остальные 512 Кб;

- cache 0-#65 — очищает начало 0-й

страницы кэша и заносит коды пор:пор:пор bc,af:inc sp,sp по адресу #60. См. §4 (H);

- R/D Code 512k — подготавливает RAM-диск в верхних (невидимых) 512 Кб. RAM-диск поддержан в TR-DOS 6.09e, и его видят все программы, обращающиеся к диску D: по адресу #3d13;

- R/D Game 768k — подготавливает RAM-диск размером 768 Кб, оставляя нижние 256 Кб свободными;

- R/D Fido 896k — RAM-диск максимального размера;

- copY disk — копирует 160 дорожек с текущего дисководов на тот,

который укажете. Буфером служит #10-я страница ОЗУ.

Постарайтесь прошить себе Gluk, работать с машиной станет значительно удобнее!

Источники информации

1. Машин С. (U#74). ROMкадром. — Infemo#4 (zx.da.ru/press/infemo/infemo4.zip).
2. zx.da.ru/system/madrom35.zip.
3. zx.da.ru/system/rromf.zip.
4. ob.raww.net/files/gluk6_0r.zip.
5. ob.raww.net/files/qs_3_04.zip.
6. Мамедов Р. (Mr.Gluk). Ода часам. — Deja Vu #8 (zx.da.ru/press/dejavu/dejavu8.zip).
7. Романов М. (Mad Max). Kempston mouse. — ZX-Format #5 (zx.da.ru/press/zx-form/zx-form5.zip).
8. REAL.SPECCY 22 Nov 01 from Oleg Maxsimuk; ZX.SPECTRUM 20 Apr 03 from Kirill Frolov.
9. Гетманец В. (V.M.G.). Псевдо-ПЗУ. — Spectrofon #17 (zx.da.ru/press/spectrofon/sp17.zip).
10. ZX.SPECTRUM 16 Feb 00 и 06 Jul 03 from Kirill Frolov.
11. ob.raww.net/files/alam5_0.zip.



Star Wars. Republic Commando

А.ВЕНДИЛОВСКИЙ,
г.Минск.

... Я стою в тени, и хотя сейчас ночь, огромный костер, разведенный на поляне, дает очень яркий свет. Но я не хочу сейчас быть замеченным. Юноша, со взглядом преждевременно повзрослевшего человека, смотрит сквозь пламя. Я не буду ему мешать. Пусть молодой Люк Скайокер простится со своим отцом. Как давно я называл его Знакином, и как долго после — Дартом Вейдером. Черные доспехи пожирал огонь, я не знаю, что видит сейчас Люк, почему вместе со слезами горя у него улыбка радости — я не джедай и никогда им не стану. Там, внизу, в долине, идет праздник — Империя пала, и новая Республика начинает свою жизнь. Но здесь нас окутывает тишина, я помню старую Республику, я помню, как все начиналось, теперь я могу позволить себе вспомнить это и... проститься.

Это было почти тридцать лет назад. Тогда Республику только-только начали сотрясать бури, которые в дальнейшем превратятся в ураган, который сметет все, что было нам известно. Светлые джедаи хранили мир, но то там, то сям вспыхивали войны. Им еще не стало известно, что Лорд Ситхов, повелитель темной стороны Силы, решил нанести свой удар возмездия за прошлые поражения, а самое главное, никто даже не мог догадаться, кто прячется под капюшоном темного господина. Но прежние руководители республики стали понимать, что, чтобы противостоять наступающей армии дроидов, им потребуется своя, не менее профессиональная армия. Среди тысяч

рас существовала в далекой галактической дали и такая, что посвятила себя проблемам генетики и клонирования, естественно, за огромные деньги. К ним-то и обратились (как

не поворачивается называть этого ублюдка своим отцом), и вскоре в одном из чанов открыл глаза я — один из тысяч армии клонов. Удивлены? Вы привыкли видеть нас в бе-

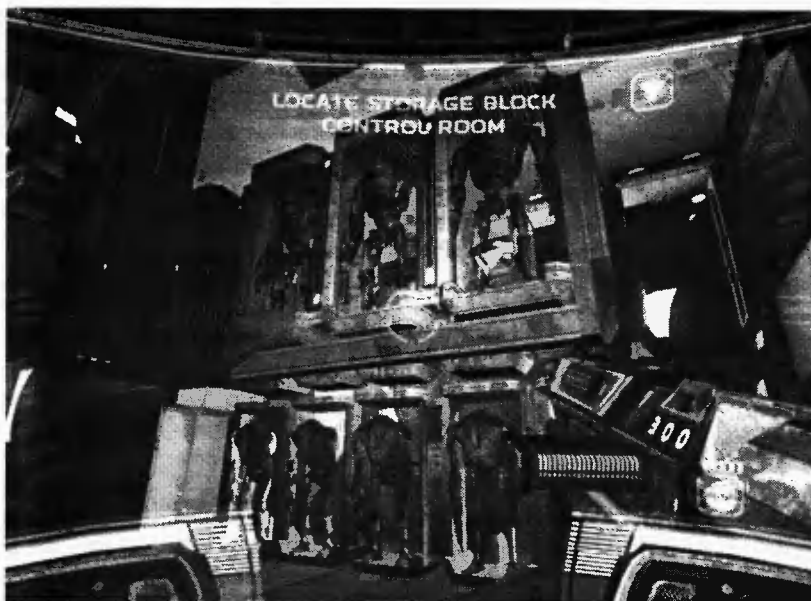
лых доспехах и называть имперскими штурмовиками? А ведь мы были командос старой республики... Мы должны были стать силой, которая защитит республику от потрясений, но излишнее подчинение, впечатанное в нас с рождения, заставит нас в дальнейшем так же выполнять указания Императора. Всего за семь лет мы из детей превратились во взрослых мужчин,

чей мир ограничивался лишь тренировками и полигонами. Мы были генетически видоизменены, для лучшей выживаемости и ускоренной реакции. Не считайте нас бездушными машинами — мы тоже умеем чувствовать боль потерь и радость победы. Поэтому и не снимаем свои шлемы — без шлема мы обычно видим своих погибших товарищей. Странно видеть свое лицо мертвым, будто это ты уже мертв, словно часть тебя умерла вместе с ним — мы ведь все же клоны... Джедаям доставалась слава, а мы делали свою работу, не претендуя на величие. Я

помню высадку на ту планету, где пришлось перебить целую армию дроидов, нам тогда сильно досталось... Кажется, вкус песка во рту ос-



позже выяснилось, опять же с подачи будущего Императора — его путь к власти был очень хитроумен...) для создания своей армии, огромной, опытной и всецело подчиняющейся



руководству. Сделка была заключена. За основу был взят генетический материал одного очень известного профессионального наемника (язык



тался у меня на всю жизнь. Тот последний бой, что изменил мою жизнь и жизнь тех, кто был рядом, навсегда останется в моей памяти. Наверное, мне повезло. Тогда мы стояли бок о бок с мастером Йодой, и как считалось, должны были окончательно покончить с угрозами Торговой Гильдии. И когда победа была в наших руках, пришел другой приказ — уничтожить джедаев! Мы были обречены рабски выполнять приказы, и наше оружие повернулось против союзников... До этого еще никто не применял на мне Силу, странное ощущение безумного потока и

стихии, пронзающей душу. Отчаянье Йоды дало и нам шанс — Сила освободила меня от бездумной покорности, а мои люди поверили мне. Мы прокладывали путь по трупам тех, кто когда-то прикрывал нас. Словно каждый раз убиваешь сам себя, словно частичка твоей души умирает тоже. Потом были двадцать лет партизанской войны, и сейчас я стою здесь и вспоминаю, глядя на пламя погребального костра — что это, конец прошлого или начало нового будущего? Или и то, и другое сразу, и будет ли там место для нас? Но об этом я подумаю завтра — сейчас время воспоминаний...

Если чем и радовал своих поклонников "товарищ" Лукас, так это качественными экшенами от первого лица. И если все остальное больше напоминало рекламный ширпотреб из серии "а сейчас заработаем на фанатах еще пару баксов", то эти экшены благосклонно принимались публикой и не позволяли линейке игр "Звездные войны" скатиться ниже уровня плintуса. Слава Богу, и эта игра не стала исключением, а по многим параметрам она подняла планку подобных игр на новую высоту, еще раз доказав, что при желании и наличии рук, растущих из правильного места, можно добиться многого.

Сюжет игры пляшет между вторым и третьим эпизодами фильма, где нам придется играть за командира-клона,

а не за всемогущего джедая, обремененного философскими проблемами. Работа тоже будет "грязная" — зачи-



стить планету, разгромить фабрику, устроить диверсию и при этом еще и самому выжить, и ребят из отряда не положить. Движок игры — очень переработанный UT2004 — добротнo "вылизан", бесполезно о нем что-то рассказывать, нет такого геймера, которому не попадалась бы игра на этом движке, так что все уже прекрасно знают как "плюсы", так и "минусы" подобного оснащения.

Перейдем к составляющим геймплея, по традиции — к оружию. Его у вас может быть только три вида... Да подождите вы тянуться за валидолом, на самом деле не все так плохо, как кажется, основное оружие десантника невероятно изменчиво — легким движением руки автомат превращается в снайперскую винтовку или штурмовик, и при этом считается одним и тем же оружием! Вдобавок можно прихватить гранатомет (только не забудьте — с гранатами всегда будет большая напряженка!) или подхватить выпавшее из рук противника оружие. Патронов к нему вы позже все равно не найдете, но некоторые экземпляры — весьма забавны и могут серьезно помочь в трудных ситуациях.

Так как бегать вам придется не в одиночку, а с командой, то тут я должен очень серьезно похвалить программистов — как приятно видеть, что твои партнеры не изображают ходячие мишени, а очень качественно работают группой, оказывая игроку серьезную поддержку. Не мечутся,

используют укрытия, прикрывают огнем — во многих случаях пройти уровень одному просто физически невоз-

можно. И тут появляется тактическая составляющая игры — мы можем отдавать различные приказы своим бойцам: вести огонь из определенного укрытия, взорвать дверь, вылечить раненого товарища (в том числе и себя!), и от правильности команд будет зависеть, сможете ли вы пройти уровень. Создатели помогли игроку, расставив призрачные силуэты бойцов в контрольных точках, так что нам остает-

ся только ткнуть указующим перстом и послать туда выбранного бойца. Противники тоже дремать не дадут — и летающие насекомые с огнеметами, и тяжелые дроиды всех мастей, и вообще непонятные твари, так и норовящие цапнуть за пятку... И под веселые дружные комментарии вместе с крепкими выражениями ваша команда практически бегом будет двигаться с уровня на уровень.

Вывод: Очень качественная игра, за которой вы проведете не один вечер!

System Requirements

Computer

Required: 100% DirectX 9.0c compatible computer
CPU

Required: Intel Pentium 3 1.0 MHz or AMD Athlon 1000 MHz

Recommended: Intel Pentium 4 2.0 GHz or AMD Athlon XP 2400+

Memory

Required: 256 MB RAM

Recommended: 512 MB RAM

Graphics Card

Required: 64 MB 3D Graphics card with Hardware Vertex and Pixel

Shader (VS/PS) Capability

Recommended: 128 MB 3D Graphics card with Hardware Vertex and Pixel Shader (VS/PS) Capability

Диск предоставлен магазином "MediaCraft" www.mediacraft.shop.by



КУПЛЮ, ПРОДАМ, ОБМЕНЯЮ

Для публикации бесплатных объявлений некоммерческого характера о покупке и продаже компьютерных комплектующих и программ, их текст можно присылать в письме по адресам:



119454, г. Москва, а/я 37 или
220095, г. Минск-95, а/я 199,
передавать по тел. в Минске (017) 249-41-47,
либо через E-mail: rm@radio-mir.com
WWW: <http://radio-mir.com>

Ищу адреса организаций, высылающих радиодетали в небольших количествах почтой.
393012, Тамбовская обл., Никифоровский р-н,
с.Никольское, Е.А.Карпенко.

Продам устройство электропривода антенны (типа МЭО, с возможностью установки сельсинов).
Ищу схемы: включения сельсинов БС-155А, тестера Ц4353, цифрового вольтметра В7-16 (с описанием).
222310, Минская обл., г.Молодечно,
ул.Дроздовича, 3 — 65. С.Н.Марчук.
Тел. 8-01773-5-37-22, 8-0295-00-41-29 (моб.).

Куплю не очень дорого диод Ганна, конвертер СВЧ, тюнер-плату для компьютера.
А.Федоров.
E-mail: alexuv@bk.ru

Продам автомобильные транковые радиостанции (заблокированные) Motorola-MC micro (2 шт.), предназначенные для работы в диапазоне 450 МГц в системе MRT-1327.
Тел. в Минске: 8-0165-32-00-25 (дом.),
8-029-673-60-87 (моб.). Александр.

Продаю новые лампы ГУ-19, ГУ-50, ГИ-7БТ, В4-защитки В1В, реле ТКЕ52ПД1, КПЕ и другие детали для усилителей мощности, а также коллекцию CD по радиолюбительской тематике.
а/я 103, г.Чебоксары, РФ, 428023.
Тел. 313442, 89030640378, Валерий.
E-mail: ra4yuv@cbx.ru
Ищу справочные данные на микросхему STV2246D.
E-mail: ua9udn@kuzbass.net

Куплю (недорого) TFT-дисплей SONY MVC-FD81.
Константин.
E-mail: kos@intertade.ru

Продам новый спутниковый ресивер Strymaster XL15, автомагнитола Kenwood KRC15.
Владимир Урбеков.
E-mail: uw-7275@mail.ru

Куплю корпус Р-160, Р-680 (желательно со жгутами).
E-mail: rx6bs@mail.ru

Куплю приемник УС-9; лампы ГУ-32, ГУ-50, ГУ-80, ГУ-82 и др.; справочную литературу по усилительным лампам.
Тел. (моб.) +7-095-509-53-43, Михаил.
E-mail: mogutov@dol.ru

Ищу два разъема (мамы) питания и автоматики от блока УМ Р-140.
397600, Воронежская обл., г.Калач, а/я 10,
И.В.Цапусову.
Тел. 8 (07363) 65-1-80.
E-mail: rw3qw@box.vsi.ru

Изготавливаю на заказ одно- и двусторонние печатные платы (без металлизации) по готовым чертежам, а также корпуса и "слесарку" из алюминия и стали (в ограниченных количествах).
E-mail: rdscoder@yandex.ru
Куплю документацию на В7-16А.
184506, Мурманская обл., г.Мончегорск-6, а/я 19.
Анатолий.
E-mail: ua1zhv@monch.mels.ru

Куплю СТВ-тюнер, СВЧ-головку, СТВ-декодеры в рабочем состоянии.
E-mail: rbtsm@yandex.ru

Куплю миниатюрные кварцы на 10,5, 17,5 и 24,5 МГц; малогабаритный приемник на радиолюбительские

диапазоны.
Тел. в Минске 280-12-97. Василий.

Куплю (срочно) журналы: "Радиолобитель" NN1 и 7 за 1998 г., "Радио" N11/1998.
456080, РФ, Челябинская обл., г.Трехгорный,
ул.Прохача, 4 — 2. С.В.Янаевой.

Куплю журналы "Радио" NN1-6/1955, N4/1957; N4/1959; N2/1963; выпуски "В помощь радиолюбителю" N1/1956; N3/1957; N6/1958; N22/1965; 5-секционный конденсатор от средневолнового радиоконуса (5x10...500 пФ); фирменные панели для ГУ-33Б; лимб настройки от радиоприемника "Атмосфера 2" или этот радиоприемник (можно в нерабочем состоянии).

Помогу приобрести ФЭМ215-03.000-3,0; ФЭМ215.000-00.300-3,0; ФЭМ215.000-03.000-1,4; ФЭМ215.000-06.000-3,0; кварцевые фильтры ФП2П1-457-34,785М-45Б; кварцы 216,65, 213,35, 216,65, 213,15 кГц.
393190, РФ, Тамбовская обл., г.Котовск,
главпочтамт, а/я 73. В.Тихонову, UA3RBX.
Тел. 8-(241)-2-06-37.

Продаю приборы В3-13, С1-19Б, лабораторный блок питания Б5-47 (30 В/3 А), ГСС-6А; новые усилители на 2хГИ-7Б, 3хГУ-50 (бестрансформаторный); блоки питания для импортных трансиверов, оконечный УЗЧ 2x200 Вт, 2x100 Вт (Н-Н); реле РЭС60 и РЭС49 (по 100 шт. б/у).

Куплю или обменяю реле РЭС15, РЭС14, В1В или аналогичные, ВЧ-разъемы СР-50-164П.
429500, РФ, Чувашия, п.Кугеси,
ул.Советская, 36. Н.В.Свинцову, RA4YM.
Тел. 8-(3540)-2-16-98.

Продаю конденсаторы К73-45-3 (9 кВ) для умножителей и люстр Чижевского.
Тел. в Бресте 23-56-44,
мобильный: (029)-527-35-80. Петр.

Куплю журнал "Радиомир. КВ и УКВ" N3/2003.
443004, РФ, г.Самара,
ул.Калининградская, 50 — 26. В.А.Смирнов.

Ученик 9-го класса примет в дар любой отечественный компьютер (можно Spectrum), желательно в рабочем состоянии.
Тел. (0942)-41-12-36. Сергей.

Продаю комплектующие для трансивера RA3AO (корпус, печатные платы с металлизацией, 20 кварцев А520 и т.д.); набор основных деталей для сборки трансивера (в комплекте: 2 ЭМФДП, кварц, 4 ИМС и КПЕ); новый верньер от Р-311; ЭМФ500-В и ЭМФ500Н, ЭМФДП 500-20; цифровую шкалу для "Largo"; кварцы 100 и 500 кГц, 1, 8, 10 МГц и т.д. в миниатюрных корпусах; транзисторы 2Т920В, КТ922А, В, Г; КТ931А, КТ971А и т.д.; б/у приборы (нерабочие) Е7-15, В7-35, ЧЗ-36; 12-вольтовые ВЧ-реле (на керамике), реле-"хлопушки", новые РЭС14 и РЭС15; б/у РЭС47, РЭС49 и т.д.; ЦШ-частотомер (по "Конурт") до 200 МГц; делитель ЦШ-02; вариометр от РСБ-70, КПЕ для усилителя мощности; конденсаторы КСО-8 (2500 В), КТ-3 и многое другое.

Куплю журналы "Радиомир. КВ и УКВ" NN7 — 9 за 2004 г.
461201, РФ, Оренбургская обл.,
пос. Новосергиевка, ул.Садовая, 20.
В.В.Печенков.

Куплю Р-399 или Р-160.
Тел. (35362)-4-37-45,
Александр.
E-mail: igor@yandex.ru

Продаю анализатор спектра С4-27 (0,01...39,6 ГГц) и генератор сигналов Г4-107 (12,5...400 МГц). Документы, комплектность, ЗИП.
Тел. 8-0722-36-48-31 (раб.).
Николай Александрович.

Продаю УВ-309 и УВ-261.
E-mail: zakupka@sama.ru

Продаю конденсаторы керамические и электролитические, транзисторы, резисторы постоянные и переменные, диоды, реле, двигатели постоянного тока ДПМ. Дмитрий.
E-mail: dimlab@land.ru

Куплю ТКЕ24П1Г. Юрий.
E-mail: oev_sks@mail.ru

Куплю монтажный провод МГШВ, МГТФ, МС. Владимир Александрович.
E-mail: radiobox@list.ru

Куплю детали Philips — BB212-AM, м/с SA86456.
Владимир.
E-mail: second2@rol.ru

Куплю схему переносной радиостанции "Волна" или поменяю на комплект схем стационарной радиостанции "Волна-К". Николай Федорович.
61170, Украина, Харьков,
ул.Тимуровцев, 17 — 42.
Тел. 67-30-08.

Куплю FT-857D (желательно в Южном регионе).
E-mail: z-centre@mail.ru

Продаю литературу для ZX SPECTRUM.
Владимир.
E-mail: ua1ogy@atnet.ru

Меняю анализатор спектра СК4-59 (в комплекте, с документацией) на трансивер.
E-mail: ew3mw@rambler.ru
Тел. 8-016-47-28-583; 8-029-329-27-11.

Изготавливаю на заказ одно- и двусторонние печатные платы (без металлизации) по готовым чертежам, а также корпуса и "слесарку" из алюминия и стали (в ограниченных количествах).
E-mail: rdscoder@yandex.ru

Продам IC-706MK2G+DSP.
624449, Свердловская область, г.Красноуринск.
Тел. 34314-3-17-24. Владимир
E-mail: ra9cck@mail.ru

Куплю переднюю панель к автомагнитоле LG TCC6230.
E-mail: UW-7275@mail.ru

Меняю комплект аппаратуры для "Охоты на лис" (4 автоматических передатчика на 3,5 и 144 МГц с аккумуляторами, 8 приемников на 3,5 МГц и 8 приемников на 144 МГц) на современный импортный КВ-трансвер.

305000, РФ, г.Курск, а/я 88, В.В.Повалеву,
UA3WW.
E-mail: rifsut@kursknet.ru

Продам за 150 руб. 16 различных б/у радиоламп.
396560, Воронежская обл., пгт.Подгоренский,
ул.Вокзальная, 35 — 54. Д.Квиткин.

Продам или обменяю старые радиоприемники "Фестиваль" и "Мир" в отличном состоянии.

Куплю блоки Б1-4, Б1-2 и Б1-8 к радиоприемнику Р-160П, валкодеры от станков с ЧПУ Z-100, реле В1В, РПВ2/7, замыкатели П1Д, оттяжки с изоляторами от "телескопа", керамический переключатель от согласующего устройства на 5 положений от Р-130.
352040, Краснодарский край, ст.Пааловская,
ул.Кирова, 37. Ю.Демченко.

Продам подшивки журнала "Радио" (за 1953...2002 гг.), сборники "В помощь радиолюбителю", отдельные номера журналов "КВ журнал" и "Радиолобитель", сборники "Лучшие конструкции радиовыставок", "Радиомеханики", 93 различных справочника, около 500 радиолюбительских книг, обширный ассортимент отечественных радиодеталей.
Для ответа на почтовый запрос необходимо приложить марки на сумму 10 руб.
301362, Тульская обл., г.Алексин, а/я 73.
Т.Стойнова.

Изготовим высококачественные кварцевые фильтры (SSB, CW, "подчисточные" — 8-, 8- и 4-кристалльные соответственно) на частоту 8,667 кГц.
309502, г.Старый Оскол,
м-рн Солнечный, 3 — 50. И.Щедрин.
Тел. (0725) 42-48-86.

Продам недорогой б/у мониторы "Электроника 32 ВТЦ202", "Электроника МС6105.01", "Электроника МС6105.05", "Электроника БМ31М/01"; б/у компьютеры МС0511-2, МС0511-01-7.
413500, Саратовская обл., г.Ершов,
Победы 1 Б, кв.16.
А.Давыдов, UA4CTQ.

Продам измеритель неоднородностей линий Р5-15 или обменяю его на трансивер FT-817. Кроме того, предлагаю лампы для телевизоров в большом ассортименте.
623271, Свердловская обл., г.Детярск,
ул.Декабристов, 17. Ю.Светогоров, RA9CCU.
Тел. (8-34397) 6-00-25, (8-34397) 6-00-30.



КУПЛЮ, ПРОДАМ, ОБМЕНЯЮ

Куплю (срочно) журналы: "Радиолобитель" NN1 и 7 за 1998 г., "Радио" N11/1998, 456080, РФ, Челябинская обл., г.Трехгорный, ул.Прохача, 4 — 2. С.В.Янаевой.

Куплю журналы "Радио" NN1-6/1955, N4/1957; N4/1959; N2/1963; выпуски "В помощь радиолюбителю" N1/1956; N3/1957; N6/1958; N22/1965; 5-секционный конденсатор от средневолнового радиокомпы (5x10...500 пФ); фирменные панели для ГУ-33Б; лимб настройки от радиоприемника "Атмосфера 2" или этот радиоприемник (можно в нерабочем состоянии).

Помогу приобрести ФЭМ215-03,000-3,0; ФЭМ215,000-00,300-3,0; ФЭМ215,000-03,000-1,4; ФЭМ215,000-06,000-3,0; кварцевые фильтры ФП2П1-457-34,785М-45В; кварцы 216,65, 213,35, 216,85, 213,15 кГц.

393190, РФ, Тамбовская обл., г.Котовск, главпочтамт, а/я 73.
В.Тихонову, UA3RBX.
Тел. 8-(241)-2-06-37.



Радиомир. Лучшие конструкции. Выпуск 1.

ООО "Радиомир Пресс", 2005 — 112 с.

Книга представляет собой сборник статей, опубликованных в разные годы в журнале "Радиомир" и заново отредактированных для данного издания.

В выпуске приведены схемы и описания устройств, используемые радиолюбителями в различных областях.

По поводу приобретения книг можно обращаться в редакцию по E-mail: rm@radio-mir.com, а также во все организации, имеющие журнал "Радиомир" в розничной продаже. Информация о них приведена рядом.

При приобретении книги через редакцию ее стоимость составляет:

- для жителей России — 50 рос. рублей;
- для жителей Беларуси — 3900 бел. рублей;

Правила приобретения — аналогично адресной подписке на журналы через редакцию.

Выберите себе вариант подписки!

Подписка через почтовые отделения

Радиомир

- для жителей России и стран СНГ (кроме Беларуси): 48996 — подписка по каталогу Агентства "Роспечать" (72370 — годовая), 25785 — подписка по Объединенному каталогу "Пресса России" с адресной рассылкой (электронный адрес подписки в INTERNET — www.presssafe.ru);
- для жителей Беларуси: 48996, (489962 — для организаций) — подписка по каталогу РО "Белпочта" "Газеты и журналы Республики Беларусь" (раздел "Издания РФ, распространяемые по прямым договорам") и через киоски Мингорсоюзпечати.

Радиомир. КВ и УКВ

- для жителей России и стран СНГ (кроме Беларуси): 48924 — подписка по каталогу Агентства "Роспечать", (71545 — годовая);
- для жителей Беларуси: 48924, (489242 — для организаций) — подписка по каталогу РО "Белпочта" "Газеты и журналы Республики Беларусь" (раздел "Издания РФ, распространяемые по прямым договорам") и через киоски Мингорсоюзпечати.

Радиомир. Ваш компьютер

- для жителей России и стран СНГ (кроме Беларуси): 48925 — подписка по каталогу Агентства "Роспечать" (45995 — годовая);
- для жителей Беларуси: 48925, (489252 — для организаций) — подписка по каталогу РО "Белпочта" "Газеты и журналы Республики Беларусь" (раздел "Издания РФ, распространяемые по прямым договорам") и через киоски Мингорсоюзпечати.

Кроме того, предприятия регионов России, а также ближнего и дальнего зарубежья могут оформить подписку на журналы "Радиомир", "Радиомир. КВ и УКВ", "Радиомир. Ваш компьютер" через ООО "Корпоративная Почта" по телефонам: (095) 953-92-62, 953-92-02, 953-93-20.

Внимание! Адресная подписка через редакцию. Подписаться на имеющиеся в наличии отдельные номера журналов, а также на любой период, начиная со следующего после оплаты месяца, можно через редакцию. Для этого нужно оплатить необходимую сумму через Сбербанк России или оформить почтовый перевод на наш расчетный счет. Текущие цены приведены в таблице. В цену включена доставка журналов в отдельном конверте по адресу подписчика, указанному в графе "Назначение платежа" при оплате через Сбербанк России или в графе "Для письма" при оплате почтовым переводом. Наложенным платежом журналы не высылаются.

Наши платежные реквизиты:

- для жителей России и стран СНГ (кроме Беларуси) —
получатель: ООО "Радиомир Пресс", ИНН 7729404741, КПП 772901001,
р/с 40702810900010000084 в ООО КБ "Агропромкредит", ф-л Центральный, г.Москва,
корр. счет 3010181050000000109, БИК 044525109
Адрес банка: 125315, г.Москва, Ленинградский пр-т, дом 76, корп. 4;
- для жителей Беларуси —
получатель: УП "РЛД", УНН 190218686
р/с 3012000004862 в ф-ле №524 АСБ "Беларусбанк" в г.Минске код 121.
Адрес банка: 220028, г.Минск, ул.Физкультурная, 31.

При оплате через Сбербанк в графе "Назначение платежа" необходимо написать свой почтовый индекс, полный адрес, фамилию, имя и отчество полностью и точно перечислить, какие конкретно номера какого из журналов Вы заказываете. При оплате почтовым переводом все сведения о заказе необходимо указать в графе "Для письма".

Для ускорения процесса получения журналов заказ можно продублировать по E-mail: rm-sales@radio-mir.com. Вся информация — там же или по тел. в г.Минске (017) 221-01-10, (017) 505-13-65.

Год	Можно заказать следующие номера журналов			Расценки на 1 экз. (с учетом пересылки)		
	Радиомир	Радиомир. Ваш компьютер	Радиомир. КВ и УКВ	По России и СНГ (рос. руб.)	По Беларуси (бел. руб.)	По Украине (гривны)
2001	9, 10, 12	7, 9 — 12	7, 8, 10, 11	25	1000	7
2002	2 — 12	1 — 12	1 — 3, 5 — 12	30	1500	7
2003	1 — 12	1 — 12	1, 5 — 8, 10 — 12	35	1800	8
2004	1 — 12	1 — 6, 8 — 12	1 — 12	40	2100	9
2005	1 — 12	1 — 12	1 — 12	45	3000	11

Приобретение отдельных номеров журналов

В РОССИИ:

В ЗАО "Интерпресс — Экспресс": тел. в Москве (095) 208-85-50, 208-67-55, e-mail: inter@aiif.ru, <http://www.interpress-express.com>.

В магазинах радиодеталей "ЧИП и ДИП" (единая справочная — тел. (095) 780-95-09):

- г.Москва, ул.Гиларовского, д.39 (ст. метро "Проспект Мира" — радиальная);
- г.Москва, ул.Ивана Франко, д.40, к.1, стр. 2 (платф.Рабочий поселок, 15 мин. от Белорусского вокзала);
- г.Москва, ул.Беговая, д.2;
- г.Москва, ул.Земляной вал, д.34.

На радиорынках в Москве: Митинском (места R4, S8, K52, E50, G56), Царицынском (место 121). В ООО "Альфино-Книга КТК": 115533, г.Москва, ул.Нагатинская, д.6, эт.1, оф.ВII (ст. метро "Нагатинская"), тел. (095) 258-91-94, 258-91-95, 258-91-96, 111-05-96, 111-63-28. E-mail: abook@mail.ru, abook@inbox.ru. На радиорынках: "Царицыно" (место 13/А), "Митино" (ряд 1, контейнер 17 и место Т-8).

На УКРАИНЕ:

В Киеве в фирме "Торм", тел. (044) 227-72-73.

В КАЗАХСТАНЕ:

В фирме ТОО СП "Аргументы и факты. Казахстан": тел. в Алма-Ате (3272) 50-22-60, 50-21-64.

В БЕЛАРУСИ:

В Минске в магазинах "Книга XXI век", пр.Ф.Скородины, д.92, тел. (017) 264-27-97 (ст.метро "Московская") и "Глобус", ул.Воподарского, д.16, тел. (017) 227-30-67 (ст.метро "Площадь Независимости").

и не подпадающие в США и других странах

КОМПЬЮТЕРЫ **СИТИПРИНТ** ДЛЯ ДОМА И ОФИСА



Безграничные возможности

Откройте для своей семьи новые способы обучения, общения, развлечений – приобретите компьютер от компании “СитиПринт” на базе процессора **Intel® Pentium® 4** с технологией HT.



СитиПринт
КОМПЬЮТЕРЫ И ОРГТЕХНИКА

220006, г. Минск, ул. Полевая 28-7А.
Тел./факс (017) 2850134, 2218728.
www.citiprint.by

Подписка-2005!

радиомир

E-mail: rm@radio-mir.com

WWW: <http://radio-mir.com>

— для жителей России и стран СНГ (кроме Беларуси): 48996 — подписка по каталогу Агентства "Роспечать" (72370 — годовая), 25785 — подписка по Объединенному каталогу "Пресса России" с адресной рассылкой (электронный адрес подписки в INTERNET — www.presscafe.ru);

— для жителей Беларуси: 48996, 489962 (для организаций) — подписка по каталогу РО "Белпочта" "Газеты и журналы Республики Беларусь" (раздел "Издания РФ, распространяемые по прямым договорам") и через киоски Мингорсоюзпечати.

- В мире оживших звуков
- Рядом с телефоном
- Танцуем от питания
- Автоматика всегда поможет
- Сам себе лекарь
- Вокруг автомобиля
- Первым делом технология
- Видеотехника
- Измерения
- Не только новичку
- Личная радиосвязь
- Справочный материал
- Радиолюбительская ярмарка

радиомир

КВ и УКВ

— для жителей России и стран СНГ (кроме Беларуси): 48924 — подписка по каталогу Агентства "Роспечать", (71545 — годовая);

— для жителей Беларуси: 48924, 489242 (для организаций) — подписка по каталогу РО "Белпочта" "Газеты и журналы Республики Беларусь" (раздел "Издания РФ, распространяемые по прямым договорам") и через киоски Мингорсоюзпечати.

- | | |
|----------------------|-----------------------------|
| ➤ Новости | ➤ Компьютер на радиостанции |
| ➤ QUA | ➤ УКВ |
| ➤ Дипломы | ➤ Усилители |
| ➤ DX-info | ➤ Трансиверы |
| ➤ Соревнования | ➤ Антенны |
| ➤ Эфирная робинзоада | ➤ Доска объявлений |

радиомир

Ваш компьютер

— для жителей России и стран СНГ (кроме Беларуси): 48925 — подписка по каталогу Агентства "Роспечать" (45995 — годовая);

— для жителей Беларуси: 48925, 489252 (для организаций) — подписка по каталогу РО "Белпочта" "Газеты и журналы Республики Беларусь" (раздел "Издания РФ, распространяемые по прямым договорам") и через киоски Мингорсоюзпечати.

- | | |
|--------------------------|------------------------|
| ➤ Мультимедиа | ➤ Диалог программистов |
| ➤ Не только новичку | ➤ Рецепты |
| ➤ Коммуникации | ➤ Мир 8 бит |
| ➤ Уроки программирования | ➤ Игротека |
| | ➤ Доска объявлений |

Кроме того, предприятия регионов России, а также ближнего и дальнего зарубежья могут оформить подписку на журналы "Радиомир", "Радиомир. КВ и УКВ", "Радиомир. Ваш компьютер" через ООО "Корпоративная Почта" по телефонам: (095) 953-92-62, 953-92-02, 953-93-20.

В Украине журналы "Радиомир", "Радиомир. КВ и УКВ", "Радиомир. Ваш компьютер" можно приобрести в фирме "Торм", тел. (044) 227-72-73.