

# !komputer

ISSN 0236-1949

3





**компьютер**© Финансы и статистика и **компьютер**

Москва, 1990

УДК 681.3 (082)

Сборник статей

Издается с 1990 г.

Выходит ежеквартально

Выпуск 3

Ответственные за выпуск:

К.Коробов, М.Цар

Редакторы: О.Ермилина, А.Поплавский

Художественные редакторы: С.Щипка,  
С.Витте

Корректор: Т.Васильева

Адрес редакции:

101000 Москва, ул. Чернышевского 7.

Издательство "Финансы и статистика",  
редакция сборника "Компьютер"

телефоны: 209-99-94; 925-35-02

Набор: **NEXT** sp. z o.o., 00-095 Варшава,  
ул. Сенаторска 40/46,

тел. 27 12 83, факс 19 34 00

Настольная издательская система:

Atari Mega ST2, лазерный принтер, сканер  
CANON IX 12F,

программа: DMC № 080247

*ca/amus*

Подписано в печать 08.1990 г.

Формат 60\*90 1/8. Бумага офсетная № 1.

Гарнитура "Сети 70г". Печать офсетная.

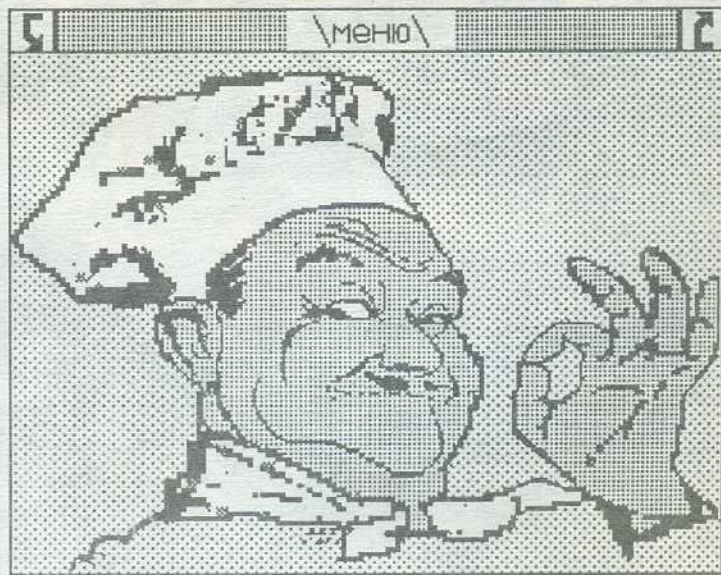
Усл.-печ. л. 8,0. Усл.-кр. отт. 28,0. Уч.-изд. 12,35

Тираж 100.000 экз. Заказ № 0-441.

Цена 1 руб. 50 к.

Полиграфический комбинат "Молодь"

252119 г. Киев, ул. Пархоменко, д. 38-44

© Составление  
К.Коробов, М.Цар,  
1990© Оформление,  
проекты шрифтов  
С.Щипка, 1990**В мире "Компьютера"****Покорение Востока**

Марек Цар

6

**История одной**

программы

8

**Питер Нортон**

Данута Маевская

9

**Автоматизированное  
проектирование**

Ежи Врубель

10

**169 рисунков за 8 недель**

Павел Лонг

11

**Нечеткие компьютеры**

Юрий Чернышов

13

**EUROMATH**

Халина Мадейчик

16



Технология баз данных  
Михаил Когаловский 17

### Компьютер на работе

AutoCAD в СССР  
Петр Кадович 23

AutoCAD даром?!  
Збигнев Блевоньский 25

Плоттеры  
Збигнев Блевоньский 26

Оценка качества  
перьевых плоттеров  
Томаш Зелиньский 30

Люминография  
Александр Григорьев 31

Заполнение  
видеобуфера EGA  
Михаил Сальников 34

Компьютерный склад  
информации 35  
М.Дец, М.Матушак, М.Млынарски

С ПК по телефону 38

Дискотека  
Роман Хмелевский 39

Словари, словари...  
Марек Цар 42

Norton Utilities: NI и BE  
Виктор Писаревский 45

SHEZ  
Ян Стожек 49

Русифицируем MS Word  
Виктор Фигурнов 51

Пишем AUTOEXEC.BAT  
Виктор Фигурнов 52

### Компьютер дома

Ханойская башня  
и кролики  
Анджей Кадлоф 55

Компрессия памяти  
ZX Spectrum  
Владислав Балясов 57

HeToBa  
Петр Завицкий 59

Ремонт Atari XL/XE  
ВладиславSOLEцкий 60

Секреты Atari XL/XE  
Петр Грабчиньский 63





# Уважаемый читатель!

Первые письма, которые стали приходить в редакцию нашего сборника после выхода первого номера "Компьютера", свидетельствуют о том, что с важнейшей задачей - удовлетворить потребности наших читателей - нам не удалось справиться. Другого толкования появившегося чуть ли не в каждом письме вопроса о том, где и когда можно подписаться на "Компьютер", я не вижу.

"Готов платить вдвое больше, но не бегать по киоскам в безуспешных поисках". - Ливинец Д.И. (Минская обл.)

"Когда будет подписка ?!!!" - Тимофеев В.М. (г. Рига)

К сожалению, "Компьютер" - не подписное издание и в 1991 г. таким же останется. Чтобы хоть как-то помочь нашим читателям приобрести следующие выпуски мы публикуем на 61 странице адреса и телефоны книжных магазинов, в которых можно купить наш сборник. Звоните, пишите, заходите, спрашивайте. "Компьютер" выпускается ежеквартально.

Впечатления наших читателей от первого выпуска - самые разнообразные.

"Не надо нас дразнить IBM: больше пишите про более доступные ЭВМ". - Казовский М.Г. (г. Минск)

"Сборник в основном рассчитан на новичков и, к сожалению, пока мало интересен профессионалам". - Харитонов С.В. (г. Москва)

"Слишком большой уклон на компьютеры, малораспространенные среди непрофессионалов в СССР". - Новиков Д.В. (г. Троицк)

"Изложение слишком популярно!!!" - Нарыжный Е.В. (г. Долгопрудный)

"Чтение журнала считаю праздником. Понравилась теплая и уважительная интонация в обращении к читателям". - Денис Баталов (выпускник 91 московской школы)

"Пестро, но интересно. Импонирует универсализм". - Бойко И.В. (г. Минск)

"В журнале достигнут удачный компромисс между безоглядной перепечаткой зарубежной прессы (...) и жалобами на отечественные болячки. Ваш (по)читатель - Кирилл Злобин, студент ВМиК МГУ".

"Пусть будет такой, какой есть. Журнал такого уровня тоже нужен". - Бартугов Б.С. (г. Элиста)

На вопрос о том, как читатели оценивают издательский уровень, ответы были тоже самыми разнообразными: от справедливых замечаний о путанице с цветами в первом выпуске на с. 7, 58, 61-62 (подвела типография) до кратенького "Это ведь не бурда" (Николаенко Е.М. - г. Одесса). Преобладали "хорошо" и "отлично".

Но, как и любая другая редакция, больше всего мы ценим пожелания читателей относительно содержания следующих выпусков. Отдельные, как видно из предшествующего и настоящего "Компьютера", нам уже удалось удовлетворить:

"Напишите про русификацию ZX Spectrum". - попросили несколько читателей.

"Неплохо было бы описывать ключи к популярным программам типа XTree, PC Tools, Norton Disc Doctor и другим утилитам, не описанным пока в советской прессе". - Излаков Ф.В. (г. Москва)

Были и другие пожелания:

"Хотелось бы прочесть о возможности сопряжения разных ПЭВМ и технических решениях этих вопросов". - Белинков С.В. (г. Мурманск)

"Надо больше прикладных статей для простых компьютеров. Ведь IBM не всем по карману". - Апшазов Р.Э. (г. Харьков)

"Мой совет: кто будет Вашим массовым потребителем, тому и уделяйте больше внимания". - Каунас Г.Т. (г. Харьков)

Да, Вы правы. И хотя далеко не все читатели будут довольны, но мы решили посвятить следующий выпуск сборника популярным домашним компьютерам - ZX Spectrum, Atari XL/XE, БК.

К сожалению, никто из читателей не предложил нам пока для публикации свои материалы. Ждем с нетерпением. Давайте редактировать вместе не только эту, но и другие страницы "Компьютера".

В заключение несколько слов о старых и новых мероприятиях, которые организованы нашей редакцией. Многие читатели обращают внимание на дороговизну на рынке программного обеспечения. Появилось даже предложение, чтобы редакция стала бесплатным распространителем программ.

Программы (пока только для IBM) типа shareware и peeware распространяются нами через московский и варшавский узлы сети FIDO (см. информацию в 1 и 2 выпусках сборника). Здесь же все вооруженные компьютерами и модемами желающие могут переписать текстовые файлы предшествующих выпусков "Компьютера" и набор советских, американских, австрийских и польских антивирусных программ.

*Marcel Car.*

Марек Цар

Варшава, Москва, август 1990



Что, когда, где

**НОЯБРЬ**

### COMDEX Fall '90

Международная компьютерная ярмарка

12-16 ноября

Лас-Вегас, штат Невада, США

Организатор: The Interface Group, 300 First Avenue, Needham, MA 02194, USA

телефон 617/4496600

телекс 951176

факс 617/4496953

### ELECTRONICA

14 специализированная ярмарка по конструктивным элементам и конструктивным группам в области электроники

6-10 ноября

Мюнхен, Германия

Организатор: Muenchener Messe- und Ausstellungsgesellschaft mbH, Postfach 121009, D-80000 Muenchen 12, BRD

телефон (089) 5107-0

телекс 5212086 AMEG D

факс (089) 5107-306

### NOBBY+ELEKTRONIK

Международная выставка компьютерной техники и электроники

8-11 ноября

Штутгарт, Германия

### AutoCAD Expo '90

Международная выставка программных средств и чертежной техники, связанной с САПР AutoCAD

29 октября - 2 ноября

Москва

Организатор: СП "Параллель", 123448 Москва, Карамышевская наб., 1

телефон 946 28 37

### ЖИЗНЬ И КОМПЬЮТЕР

Международная выставка компьютерной, множительной техники и средств связи с сопровождающим Всесоюзным научно-практическим семинаром по автоматизации инженерного труда

19-25 ноября

Харьков, Украина

Организатор: НПО "Турбоатом", 310323 Харьков, Московский пр-т, 199

телефоны 90 20 20, 92 30 11

телеграф Жико

телекс 115125 tina su

### СЕМИНАР

"Математическое моделирование и создание САПР для расчета, анализа и синтеза антенных систем и их элементов"

ноябрь, 3 дня

Черноголовка

Организатор: ЦП ВНТО радиотехники, электроники и связи, 103897 Москва, ГСП-3, ул. Кузнецкий мост, 20

телефоны 921 71 08, 924 70 72

### СЕМИНАР

"Локальные вычислительные сети для САПР ИЭТ и систем управления ГПС на базе ЭВМ семейства "Электроника"

ноябрь, 3 дня

Москва, ВДНХ СССР

Организатор: ЦНИИ "Электроника", 117415 Москва, пр-т Вернадского, 39

телефон 432 91 20

### КОНФЕРЕНЦИЯ

"Автоматизация проектирования и производства вычислительных машин и систем"

ноябрь, 3 дня

Москва

Организатор: Московский Государственный технический университет, 107005 Москва, ул. 2-я Бауманская, 5

телефон 267 05 53

### СЕМИНАР

"Модели микроЭВМ семейства CM 1800 (технические средства, программное обеспечение, вопросы применения)"

ноябрь, 3 дня

Москва, ВДНХ СССР

Организатор: НПО

"Электронмаш", 117812

Москва, ул. Вавилова, 24

телефон 433 57 81

### КОНФЕРЕНЦИЯ

"Создание и применение гибридных экспертных систем"

ноябрь, 3 дня

Рига, Латвия

Организатор: Рижский политехнический институт, 226355 Рига, ул. Ленина, 1

телефон 22 58 85

**ДЕКАБРЬ**

### SINGAPORE INFORMATICS '90

Международная ярмарка вычислительной и информационной техники

12-16 декабря

Сингапур, World Trade Center

Организатор: Singapore Federation of Computer Industries; c/o Techfairs corp. Pte, 111 North Bridge Road; #16-01 Peninsular Plaza; Singapore 0617

телефон 65 337-6265

телекс 55540

факс 65 336-0718

### КОНФЕРЕНЦИЯ

"Состояние и тенденции развития отечественных средств вычислительной техники"

декабрь, 3 дня

Москва, ВДНХ СССР

Организатор: ЦНИИ "Электроника", 117415 Москва, пр-т Вернадского, 39

телефон 432 91 20

### IV КОНФЕРЕНЦИЯ

"Проблемы оптической памяти"

декабрь, 3 дня

Телави, Грузия

Организатор: Институт Кибернетики АН Грузинской ССР, 380086 Тбилиси 86, ул. С. Эули, 5

телефон 31 27 06

### СОВЕЩАНИЕ

"Экспертные системы"

декабрь, 3 дня

Суздаль

Организаторы: Национальный комитет СССР по автоматическому управлению, Институт проблем управления (автоматики и телемеханики) Мин-электротехприбора СССР и АН СССР, 117342 Москва, ул. Профсоюзная, 65

телефон 334 91 30

### КОНФЕРЕНЦИЯ

"Информатизация и моделирование территориальных социально-экономических объектов"

декабрь, 3 дня

Новосибирск

Организатор: Вычислительный центр СО АН СССР, 630090 Новосибирск, пр-т Академика Лаврентьева, 6

телефон 35 05 40

**ЯНВАРЬ 1991**

### COMPUTER '91

VI Международная компьютерная выставка

22-25 января

Варшава, Польша

Организатор: AGPOL, Sienkiewicza 12, 00-010 Warszawa, Polska

телефон 26 49 83

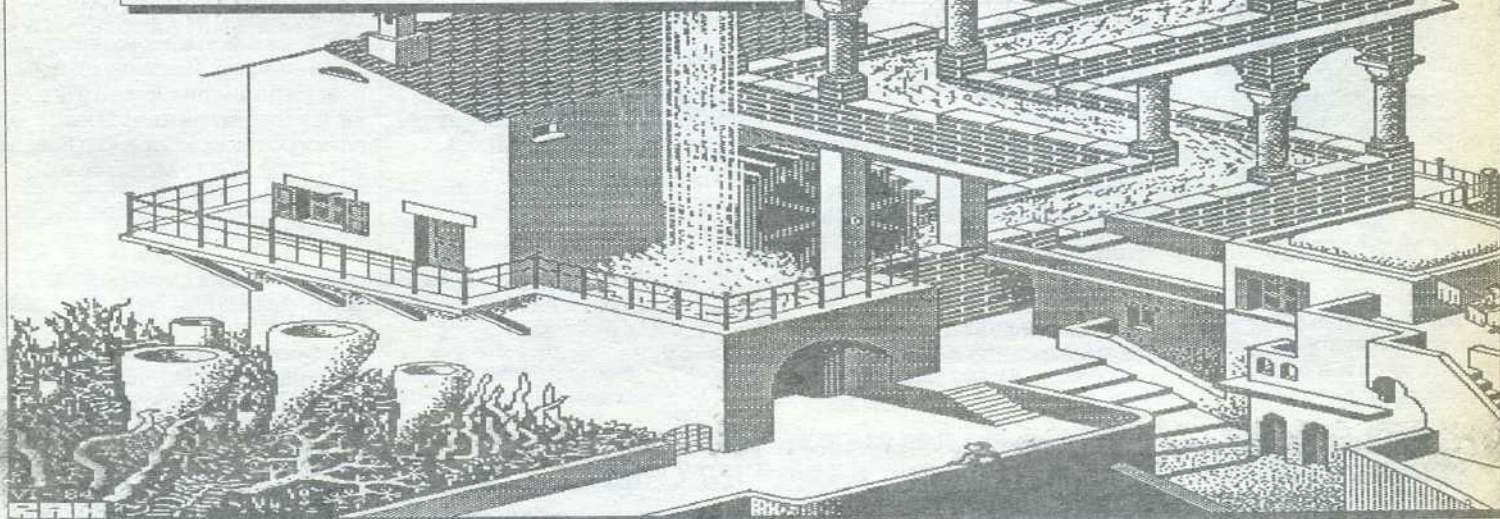
телекс 813567 pl

факс 26 47 42



## В мире "Компьютера"

- ☐ Покорение Востока
- ☐ История одной программы
- ☐ Питер Нортон
- ☐ Автоматизированное проектирование
- ☐ 169 чертежей за 8 недель
- ☐ Нечеткие компьютеры
- ☐ EUROMATH
- ☐ Технология баз данных



## В мире "Компьютера"

© Марек Цар

# Покорение Востока

С Романом Альбрехтом, представителем фирмы AUTODESK Ltd., беседует Марек Цар.

Марек Цар: AUTODESK — одна из немногочисленных западных фирм, последовательно реализующих стратегию борьбы за рынки Восточной Европы. В этой борьбе вы добились определенных успехов, под которыми я подразумеваю количество программ AutoCAD, проданных легальным пользователям. Насколько прочна ваша позиция на этом рынке?

Роман Альбрехт: AUTODESK завоевала 60 % мирового рынка систем автоматизированного проектирования (САПР) для всех типов ЭВМ. На рынке персональных компьютеров наше преимущество еще более ощутимо: из десяти PC, на которых в мире реализуются САПР, восемь оснащено программой AutoCAD. До конца первого квартала 1990 г. во всем мире мы продали 360 тысяч наших программных продуктов.

Первые попытки выйти на рынки Восточной Европы были предприняты 5 лет назад. В 1990 г. общая стоимость проданных нами здесь программ составит примерно 5 млн. дол.

С самого начала было очень трудно. Ведь и по сегодняшний день здесь считается вполне нормальным брать, а не покупать программное обеспечение. К счастью, положение меняется и все больше фирм покупает, а не просто "копирует" наши продукты. AutoCAD завоевал от 60 до 90 % рынка систем автоматизированного проектирования. Самое большое количество наших продуктов продано в Чехословакии (1 200) и Советском Союзе (1 500). В настоящее время в СССР ведутся переговоры о закупке 3 000 наших систем Министерством энергетики.

Но я хотел бы подчеркнуть, что на этом рынке количество проданных копий — не самое главное для AUTODESK. Мы знаем уровень квалификаций инженеров и архитекторов из стран Восточной Европы. Знаем также, что сегодня при разработке хороших проектов все труднее приходится без ЭВМ. Чем мощнее будут средства проектирования, тем быстрее повысится уровень инженерных разработок, тем раньше Восточная Европа начнет торговать с Западной на равных правах. Пока еще вы продаете на Запад преимуще-



ственно сырье, а покупаете там современную технологию. Но в не столь отдаленном будущем такое положение должно измениться.

Задача AUTODESK - помочь в этом. Именно этой цели мы подчинили в Восточной Европе сеть наших дилеров, сеть центров по обучению программе AutoCAD, сеть разработчиков прикладных программ для AutoCAD.

**М.Ц.** Но ведь не будете же вы "благотельствовать"?

**Р.А.** Наш опыт показывает, что в Восточной Европе можно найти очень хорошее программное обеспечение. Так, разработанный советскими программистами компилятор AutoLISP, а также программы ABase и GLISP мы считаем самыми лучшими продуктами.

**М.Ц.** Сколько фирм в Восточной Европе занимается распространением ваших программных продуктов?

**Р.А.** При постройке сети дилеров мы обращаем внимание на то, чтобы были учтены все специальности, в которых применяются наши программы: архитектура, механика, электроника и пр. Таким образом должны специализироваться и дилеры.

В настоящее время в СССР распространением программных продуктов AUTODESK занимается 5 фирм, в Чехословакии и Венгрии - по 4, в Польше, ГДР и Югославии - по 2, в Болгарии и Румынии - по 1.

**М.Ц.** Какие языковые версии программы AutoCAD они продают?

**Р.А.** Русскую, польскую и чешскую. Продолжается разработка венгерской, причем именно венгры первыми получают национальную разработку AutoCAD версии 11. В Болгарии используется русско-язычная версия. Разрабатывать ли румынскую языковую версию - мы пока еще не знаем. Сложнее обстоит дело в Югославии. Там 5 различных языков и пришлось бы разработать столько же языковых версий, а это нам пока не под силу.



Фото Мирослава Стемпняка

В общей сложности AUTODESK подготовила 15 языковых версий AutoCAD, в том числе испанскую, французскую, немецкую, шведскую, японскую и др.

**М.Ц.** Насколько я знаю, национальные версии продаются гораздо дешевле исходной, американской.

**Р.А.** Да, но только пользователям в Восточной Европе. Эта как раз и есть та форма помощи, о которой я говорил раньше. Если оригинальный AutoCAD стоит 3 000 фунтов стерлингов, то "восточноевропейский" - 1 200 фунтов стерлингов, причем каждому, кто покупает несколько копий этой программы, мы дополнительно предлагаем как минимум 20%-ую скидку.

**М.Ц.** AUTODESK была первой фирмой в мире, которая не только продавала странам Восточной Европы свои программы, но и организовала там центры по обучению AutoCAD.

**Р.А.** Мы придерживаемся этого принципа не только в Восточной Европе, но и во всем мире. До нас САПР продавались, как правило, вместе с мини-ЭВМ. Мы первыми решили удовлетворить потребности пользователей, оставляя решение вопроса выбора компьютера за ними и предлагая обучение обслуживанию наших программных продуктов в специальных центрах.

**М.Ц.** Вашему примеру решили последовать и другие фирмы, в том числе американская фирма NOVELL.

**Р.А.** Но между нашими центрами и центрами фирмы NOVELL - существенное различие. Их система обучения рассчитана, главным образом, на разработчиков программного обеспечения, работающего в среде их сетей. Наши центры - для обычных инженеров.

Для того чтобы работать с AutoCAD, не надо быть программистом. Я бы даже сказал больше: эта программа построена так, что инженер может с ней работать вообще без специального обучения. Проводились эксперименты, которые показали, что научиться AutoCAD без учебника можно за 2 месяца, с учебником - за 3 недели, а наш центр готовит специалистов всего за неделю. Кроме того, не все возможности этой программы, которые будут использоваться архитектором, интересны инженеру-механику. В наших центрах учитывается специфика работы инженеров.

**М.Ц.** AUTODESK достиг огромного успеха - ваша программа стала мировым стандартом. Тестируя другие системы, компьютерные журналы сравнивают их с вашей. Ограничивается ли вы только выпуском новых версий AutoCAD, учитывая все более совершенные технологические решения компьютерной техники?

**Р.А.** В последнее время мы разработали два совершенно новых продукта. Hypertext - это воплощение в жизнь столь популярной сегодня в мире идеи Multimedia. Она объединяет все виды информации: текст, картины, filmy, свет, музыку и пр. Hypertext AUTODESK представляет собой своеобразную базу данных, благодаря которой пользователь может проиллюстрировать подготавливаемый текст фрагментом фильма и дополнить его музыкой.

Второй продукт - CyberSpace - САПР нового поколения. В этой системе конструктор общается с ЭВМ с помощью голоса, используя при этом специальные перчатки и шлем с дисплеями, расположенными прямо перед глазами.

Благодаря дисплеям он видит в перспективе все, что на самом деле находится в памяти компьютера, причем связь - двусторонняя. Достаточно сделать шаг влево, чтобы изображение в памяти компьютера поменялось соответственно новому "углу зрения". Благодаря перчаткам конструктор может "взять в руки" изображенный в памяти ЭВМ объект (например, окно или стенку в проекте нового здания) и перенести его в другое место. Новая система обеспечивает ощущение трехмерности. CyberSpace - это не только программное обеспечение, но и специальное оборудование. Стоимость нашего АРМ, оснащенного САПР нового поколения, составляет всего 28 000 дол.

**М.Ц.** Благодарю Вас за беседу.





5 \ в мире "Компьютера" \ 12

# История одной программы

## 1982 год

**Январь.** 13 программистов компьютерных систем под руководством Джона Уокера (John Walker) и Дэна Дрейка (Dan Drake) принимают решение об образовании фирмы, выпускающей программное обеспечение.

**Апрель.** В Калифорнии суд регистрирует фирму AUTODESK Inc., уставный фонд которой составляет 59 030 долларов. Основателями являются 11 американцев и 4 европейца.

**Декабрь.** Завершена работа по выпуску программы AutoCAD 1.0 для компьютеров S-100 с процессором Z-80.

## 1983 год

**Январь.** AutoCAD версия 1.1 для компьютеров IBM PC и Victor 9000.

**Апрель.** Первая инаязычная версия программы (на немецком языке).

**Май.** AutoCAD версия 1.2.

**Август.** AutoCAD версия 1.3.

**Сентябрь.** Французская версия программы.

**Октябрь.** Версия 1.4 с ADE-1 (Advanced Drafting Extension).

## 1984 год

**Февраль.** В Швейцарии основан первый зарубежный филиал фирмы AUTODESK.

**Август.** Возникают филиалы фирмы в Швеции и Великобритании.

**Октябрь.** На рынок выпущен AutoCAD версия 2.0 с ADE-2. Основан первый авторизованный центр по обучению пользователей программы.

**Ноябрь.** Первая трехмерная версия программы. Число легальных пользователей AutoCAD превышает 10 000.

## 1985 год

**Март.** Создание филиала в Японии.

**Май.** Появление версии 2.1 с ADE-3.

**Август.** Версия программы на японском языке.

**Сентябрь.** AutoCAD по-итальянски.

**Ноябрь.** Испаноязычная версия программы. Число пользователей превышает 35 000.

**Декабрь.** Версия 2.18 с AutoLISP.

## 1986 год

**Июнь.** Версия 2.5 для IBM PC и 32-разрядных компьютеров Sun-2 и Sun-3. На рынок выпущен пакет AutoCAD AEC Architectural для автоматизации работы архитектора.

**Октябрь.** Читатели американского журнала "PC World" присуждают программе AutoCAD премию World-Class CAD; на основе программы AutoCAD выпускается AutoSketch - очень простой пакет САПР.

**Декабрь.** Крупнейший немецкий компьютерный журнал "Chip" присуждает программе AutoCAD звание "Лучшая программа 1986 года".

## 1987 год

**Март.** AutoCAD версия 2.6 для IBM PC, DEC VAX и IBM PS/2.

**Май.** В подготовленном журналом "Business Week" списке самых быстроразвивающихся фирм AUTODESK занимает ведущее место.

**Июнь.** AUTODESK организует в Вашингтоне выставку "AutoCAD Expo", в которой принимает участие более 90 фирм, выпускающих программы, которые работают в среде AutoCAD. Число легальных пользователей превышает 100 000.

**Август.** На рынке появляется AutoCAD Mechanical - программа для конструкторов механического оборудования.

**Сентябрь.** Появляется AutoCAD версия 9 и программа AutoShade (возможность шатировки).

**Октябрь.** Читатели "PC World" второй раз присуждают программе премию World-Class CAD. "PC Magazine" награждает программу AutoShade за "техническое совершенство". "Digital Review" объявляет AutoCAD программой года.

## 1988 год

**Январь - март.** Авторизованные центры по обучению программе AutoCAD возникают в СССР, Польше, Чехословакии и Болгарии. Число легальных пользователей достигает 150 000.

**Октябрь.** Выставка "AutoCAD Expo" впервые проводится в Москве. Число пользователей превышает 170 000.

## 1989 год

**Январь.** AutoCAD версия 10.0 - трехмерная САПР с полной перспективой.

**Сентябрь.** Польско-язычная версия программы. Число пользователей во всем мире превышает 200 000.

**Ноябрь.** Выставка "AutoCAD Expo" второй раз проводится в Москве. Число пользователей превышает четверть миллиона.

## 1990 год

**Январь.** Фирма AUTODESK выпускает компилятор языка AutoLISP (разработанный



советскими программистами); в международной анкете, проводимой журналом "Chip", AutoCAD версия 10.0 признана лучшей программой 1989 года.

**Март.** "AutoCAD Expo" в Великобритании.

Впервые демонстрируются элементы AutoCAD версия 11.0: интегрированный AutoSOLID и интерфейс, обеспечивающий связь программы с языком Си.

**Апрель.** Число владельцев легальных копий программы AutoCAD достигает 300 000.



\ в мире "Компьютера" \

© Данута Маевская

# Питер Нортон

Вся профессиональная жизнь 45-летнего Питера Нортон-а связана с программированием.

В 1982 г. он случайно стер исходный файл с жесткого диска своего ПК. Восстановление файла оказалось сложным и кропотливым делом. Но Нортон вспоминает этот момент своей жизни с некоторым благоговением. Ведь именно таким образом судьба заставила его написать программу, являющуюся прообразом сегодняшних утилит.

Затем появились и другие утилиты, способствующие облегчению труда программиста. Нортон попытался их продать и, вдохновленный первым успехом, решил именно так зарабатывать себе на жизнь. Он слегка изменил написанную для собственного употребления программу восстановления стертых с диска данных и выпустил ее на рынок программного обеспечения под названием UnErase. Она сразу же приобрела большую популярность.

Главным мотивом основания собственной фирмы, как утверждает, кокетничая, сам Нортон, была боязнь перед другим трудом. Но успех пришел не сразу.

Свою деятельность Питер Нортон начал с рассылки бесчисленного количества бесплатных, полных обещаний реклам. Такая расточительность чуть не привела его в 1982 г. к банкротству, хотя, конечно, способствовала его популярности.

Сегодня фирма PETER NORTON COMPUTING, Inc. (Санта Моника, Калифорния) широко известна всем пользователям компьютеров, совместимых с IBM PC. Она знаменита такими великолепными программами, как Norton Commander, Norton Integrator, Norton Utilities, Norton Disc Doctor. В 1989 г. объем продажи фирмы составлял 5 млн. дол., а по результатам коммерческой деятельности она располагается среди третьего десятка изготовителей программного обеспечения. При этом надо учесть, что в фирме работают всего лишь 13 человек.

Одна из последних разработок Питера Нортон-а (не считая новых версий программ Norton Utilities 4.5 и Norton Commander 3.0) называется "The Norton On-Line Programmer's Guides". Это — серия из четырех одинаково действующих автоматизированных справочников по различным языкам программирования: Ассемблеру, Си, Бейсику и Паскалю.

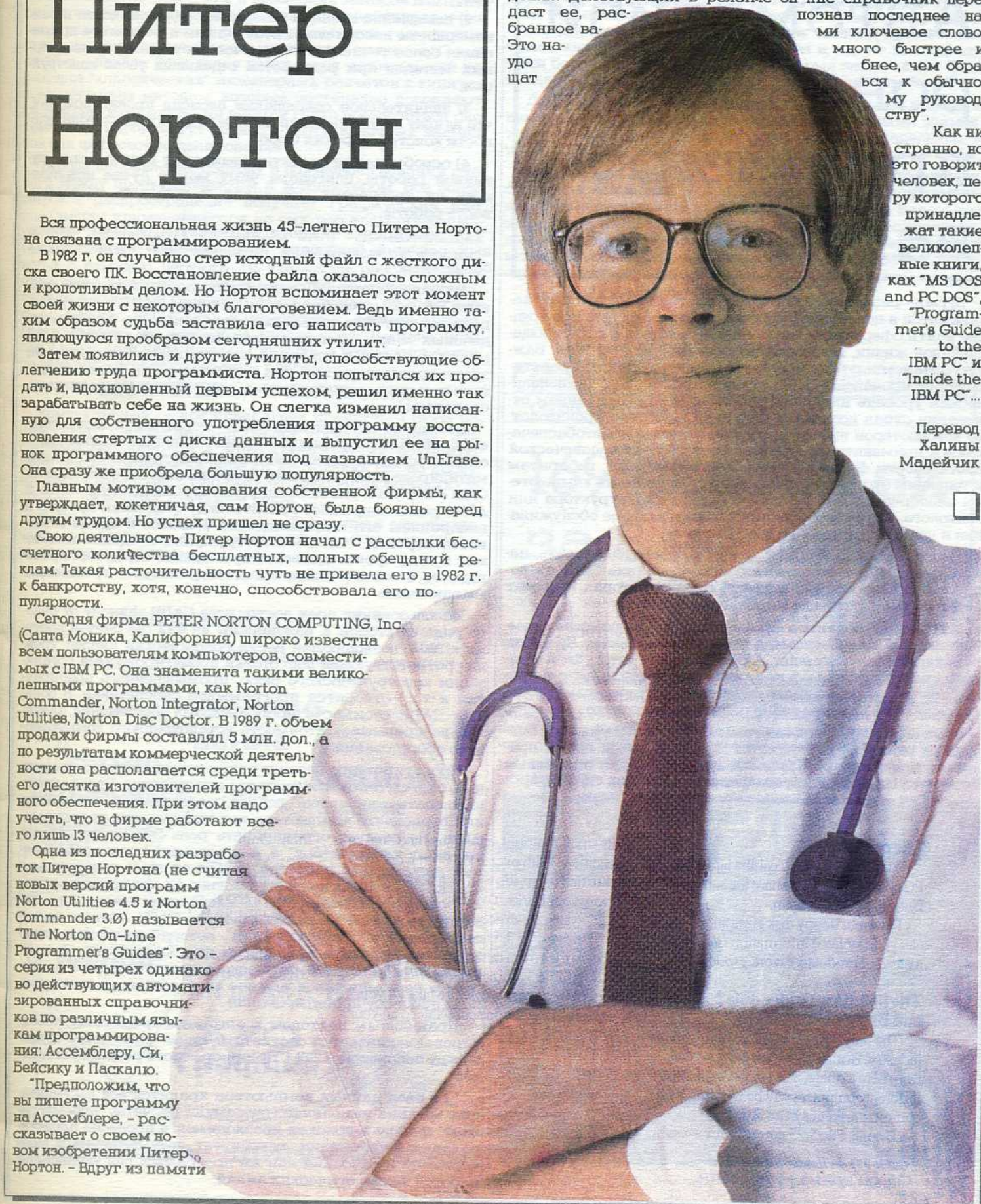
"Предположим, что вы пишете программу на Ассемблере, — рассказывает о своем новом изобретении Питер Нортон. — Вдруг из памяти

вылетает необходимая информация об определенном регистре. Благодаря моей программе вы сможете, нажав клавишу, "обратиться" к справочнику по Ассемблеру и тут же вывести на экран необходимые в данную минуту сведения. Действующий в режиме on-line справочник передаст ее, рас-  
бранное ва-  
Это на-  
удо  
чат

познав последнее на-  
ми ключевое слово.  
много быстрее и  
бнее, чем обра-  
сь к обычно-  
му руковод-  
ству".

Как ни  
странно, но  
это говорит  
человек, пе-  
ру которого  
принадле-  
жат такие  
великолеп-  
ные книги,  
как "MS DOS  
and PC DOS",  
"Program-  
mer's Guide  
to the  
IBM PC" и  
"Inside the  
IBM PC"...

Перевод  
Халины  
Мадейчик





# Автоматизированное проектирование

Бурное развитие компьютерной техники, которое наблюдается в последние годы, привело к тому, что все труднее найти сферу экономики, науки, техники, а также общественной жизни, в которой не применялись бы ЭВМ. Важную роль компьютеры играют сегодня в процессе проектирования, производства и эксплуатации машин. Технологический уровень наиболее развитых в промышленном отношении стран позволяет наладить массовое производство компьютеров вместе с соответствующим матобеспечением и применять их почти во всех сферах человеческой деятельности. Благодаря приемлемым ценам, габаритам и условиям эксплуатации можно устанавливать компьютеры непосредственно на рабочем месте конструктора или технолога, в производственных цехах, в сфере обслуживания и даже в самих машинах.

Перед такой ситуацией стоит и промышленность наших стран, ибо эффективное использование ЭВМ становится необходимостью в любой стране, которая не хочет отстать далеко позади мирового прогресса.

Автоматизированное проектирование с помощью компьютера (САПР - Система Автоматизированного Проектирования; CAD - Computer Aided Design) - это такой процесс проектирования, в котором компьютер вместе с программным обеспечением является инструментом проектировщика на каждом этапе разработки проекта.

С начала 80-х годов системы САПР стали важным элементом компьютерного рынка и в ближайшее время превратятся в стандартные элементы оснащения проектных бюро. Рис. 1 демонстрирует сферу применения САПР [1, 3].



Рис. 1. Сфера применения САПР

К достоинствам САПР можно отнести:

- 1) упрощение выработки оптимального конструктивного решения, а тем самым снижение стоимости производства, эксплуатации и достижения высшего качества машин;
- 2) повышение степени безопасности, надежности и равномерности износа элементов машин в результате применения более точных математических моделей и инженерных методов при разработке отдельных узлов конструкции;
- 3) значительное сокращение периода проектирования, что влияет на уменьшение издержек и рост производительности конструкторских бюро;
- 4) освобождение проектировщиков от рутинной работы, благодаря чему можно лучше использовать их творческий потенциал и повышать эффективность работы;
- 5) расширение области применения готовых проектных решений благодаря использованию компьютерных баз данных;
- 6) проведение углубленных исследований на этапе проекта. Это возможно благодаря методам цифрового моделирования, которые позволяют анализировать влияние отдельных конструктивных параметров на качество всей машины (также в условиях экстремальной нагрузки) на этапе проектирования, без необходимости создания прототипа и проведения стендовых или эксплуатационных исследований.

Выгода от применения САПР значительно возрастает, если автоматизированное проектирование сопряжено с автоматизацией управления производством (АСУП - Автоматизированная Система Управления Производством; CAM - Computer Aided Manufacturing). Объединение САПР и АСУП резко сокращает время между созданием проекта и введением его в производство. Дешевле и проще становится переход к новым конструкциям, введение изменений в ходе производства, выпуск малых серий или производство в соответствии с индивидуальными требованиями заказчика.

Использование всех достоинств САПР зависит от целого ряда факторов. К важнейшим из них (не считая доступа к ЭВМ, позволяющей вести графический диалог, уровня соответствующей подготовки проектировщиков, организации конструкторского бюро, промышленного предприятия и т.д.) относится программное обеспечение. В больших САПР стоимость программного обеспечения значительно превышает стоимость установленного компьютерного оборудования.

Проектировщик должен знать возможности современных систем автоматизации проектирования. Часто можно наблюдать ошибочный подход, состоящий в недооценке или переоценке компьютера как инструмента проектировщика. Не следует ограничивать роль ЭВМ к "производительному калькулятору" т.е. к автоматизации традиционных инженерных расчетов. Польза от этого относительно невелика, поэтому не стоит удивляться, что проектировщики, именно так понимающие назначение компьютера, не заинтересованы в использовании САПР. Нельзя также стремиться к "автоматизации", которая почти полностью заменяет человека в процессе проектирования. При попытке введения "автоматического проектирования" достигается нечто противоположное творческому по своей природе процессу проектирования.

Правильным подходом к автоматизированному проектированию является соответствующее "разделение труда" между человеком и ЭВМ. Принципы этого разделения таковы:

□ в базе данных компьютера хранится вся документация, машина выполняет трудоемкие расчеты, "просматривает" файлы в поисках необходимой в процессе проектирования информации, генерирует документацию в виде чертежей и описаний или же программы для управления работой технологических линий;



□ проектировщик формулирует задачи, разрабатывает концепции, используя возможности САПР, принимает критерии оценки и решения.

Современные САПР совершенствуются не только в отношении моделирования, но и в отношении методов коммуникации проектировщика с машиной. Первое направление сводится прежде всего к совершенствованию геометрического моделирования и математических расчетов. С конца 70-х - начала 80-х годов начинается решительный переход от "классических" инженерных расчетов к типично компьютерным методам, которые постепенно дополняются новыми, позволяющими легко и гибко пользоваться системой. Примером могут служить метод конечных элементов или цифровое моделирование.

Второе направление состоит в совершенствовании диалога между проектировщиком и компьютером. Все более широкое распространение получает графический диалог (вместо применявшегося до сих пор алфавитно-цифрового). Используя современных САПР, проектировщик приближается к классической "инженерной дискуссии", когда с помощью чертежа он творчески разрабатывает идею своего проекта. Чертеж является не только геометрической картиной проектируемой машины, но и логической схемой. Пользуясь компьютерной терминологией, можно сказать, что чертеж на экране графического дисплея - это своего рода внешняя память, в которую проектировщик записывает результаты работы.

### СОСТАВНЫЕ ЭЛЕМЕНТЫ САПР

Компьютерный рынок предлагает проектным бюро целый спектр систем автоматизации проектирования. Эти системы различаются в зависимости от их возможностей, что отражается на цене (от нескольких сот до десятков тысяч долларов). Стоимость оборудования больших систем не превышает 20% общей цены.

САПР состоит из базы данных, библиотеки матобеспечения и подсистемы для коммуникации между отдельными элементами системы.



Важным элементом САПР являются модули для геометрического моделирования. Они позволяют строить геометрическую модель конструкции, т. е. вводить в память ЭВМ все геометрические характеристики. Моделирование ведется обычно в режиме диалога и проектировщик управляет всеми стадиями создания модели на экране графического дисплея. В его распоряжении - набор примитивов, таких, как прямая, окружность, дуга и т.д. Из них образуются более сложные элементы, которые затем могут быть введены в библиотеку, например, в качестве символов.

САПР дает возможность вводить изменения в созданную геометрическую модель, дополняя ее новыми элементами, видоизменяя существующие, устраняя отдельные и т.д.

В практике проектирования весьма часто применяется параметрический модуль, который служит для изменения размеров указанного фрагмента чертежа. Благодаря этому модулю проектировщик может легко проанализировать несколько вариантов конструкции. Параметрический модуль может быть использован также при исследовании оптимизации конструкции.

Перевод Анджее Поплавского

□ \ в мире "Компьютера" \ □

Обработка Павла Лонга

## 169 чертежей за 8 недель

Представьте себя на месте руководителя многоотраслевого проектного бюро, которому предложили заказ на разработку проекта фабрики взрывчатых веществ, а также на подготовку технической документации (почти 200 листов чертежей). Срок работы - восемь недель. Вдобавок ко всему документация, разумеется, должна соответствовать используемой заказчиком системе Intergraph. Такое предложение получила нью-йоркская фирма Amman & Whitney.

Уже более 25 лет в фирме, в которой занято 300 человек, для проектных и расчетных работ используются компьютеры. Когда она получила описанное выше предложение (июль 1987 г.), компания располагала пятью ПЭВМ (Compaq 386 и IBM PC AT), программой AutoCAD версия 2.52 и графопостроителями CalComp 1043. Успешно применяя AutoCAD и программное обеспечение для сопряжения отдельных отраслей проектных работ, фирма Amman & Whitney разработала за восемь недель строительный (архитектура, конструкции, монтаж), транспортный и механический проекты. Расскажем о том, как проходила работа.

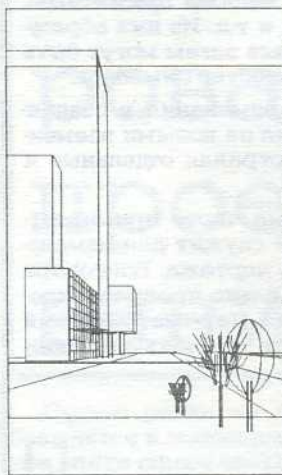
### График работ

К выполнению заказа Amman & Whitney приступила в середине августа. Работу следовало завершить 9 октября. Необходимо было спроектировать 12 зданий фабрики взрывчатых веществ. Дополнительную трудность представляло требование, что во избежание накопления "взрывчатой" пыли каждое здание должно ежедневно промываться водой. Для этого облицовка стен и потолков должна быть





гладкой, а канализационная система – безупречной. По предварительным расчетам следовало выполнить 199 архитектурных и конструкторских чертежей, в том числе проекты сантехнической, электрической проводки и противопожарной сети.

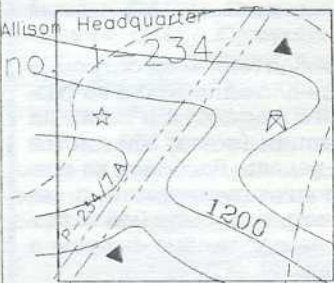


Прежде всего следовало обдумать организацию труда и подготовить график работ. Было принято решение о применении программы AutoCAD, но, учитывая требование заказчика о совместимости с системой Intergraph, количество слоев (layers) было ограничено 64. Затем были систематизированы названия слоев по отдельным видам работ. Точно так же авторы проекта поступили с цветами, которые были присвоены отдельным слоям (требование совместимости с Intergraph). Были унифицированы размеры и масштаб чертежей, шрифты нанесенных на чертежи текстов и комментариев и пр.

Затем была проведена оценка времени работы компьютеров (первоначально их было только 5). Оказалось, что выполнить заказ на них – невозможно. Фирма приобрела две дополнительные ПЭВМ и наняла на работу еще двух операторов AutoCAD. К работе приступило 10 человек, располагающих 7 машинами.

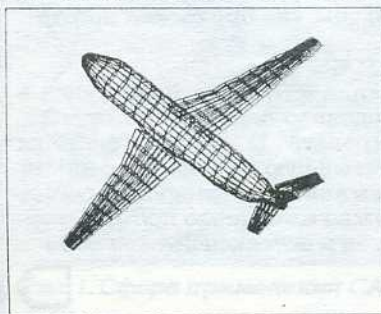
### Преодоление барьеров

Очередной трудностью, которую пришлось преодолеть, оказалось построение чертежей на графопостроителях. Всего следовало подготовить 199 листов чертежей. Для вычерчивания каждого из них в среднем требовалось 45 минут. В общей сложности эта работа должна была длиться приблизительно две недели. На случай заторов разработчики договорились с сервисной фирмой FLM, которая могла за одну ночь изготовить 60 чертежей и утром доставить их в двух экземплярах на пленке и в шести озаглавленных копиях.



Для лучшей организации в фирме Amman & Whitney была предусмотрена чертежная. В ней работал один человек, и все операторы передавали ему дискеты, на которые были выведены чертежи. Дополнительно ему сообщалась информация о формате чертежа (обычно A1 или A0) и количестве

пишущих элементов в графопостроителе. На каждой дискете писалась фамилия и номер телефона оператора, которому передавалась информация о завершении процесса черчения. Когда чертежная была перегружена, т.е. если с момента передачи дискеты до выдачи листа проходило более 24 часов, разработчики пользовались услугами фирмы FLM. Практика подтвердила предусмотренную авторами необходимость стандартизации имен файлов на дискетах. Название каждого чертежа включало номер проектируемого здания, букву, соответствующую специальности, к которой относился лист (архитектурные, конструкторские и т.п. чертежи), и порядковый номер чертежа. Итак, файл с именем 503-E-3 содержал третий чертеж проекта электрической проводки в здании номер 503. Заведующий про-

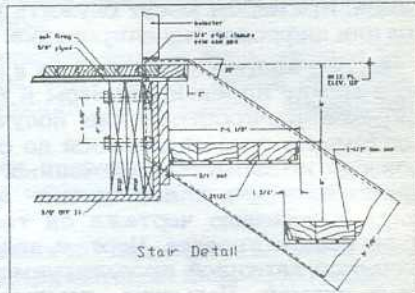


ектным отделом отслеживал маршрут, который проходил каждый лист, и в любую минуту мог ответить на вопрос, на каком этапе и у кого находится тот или иной чертеж.

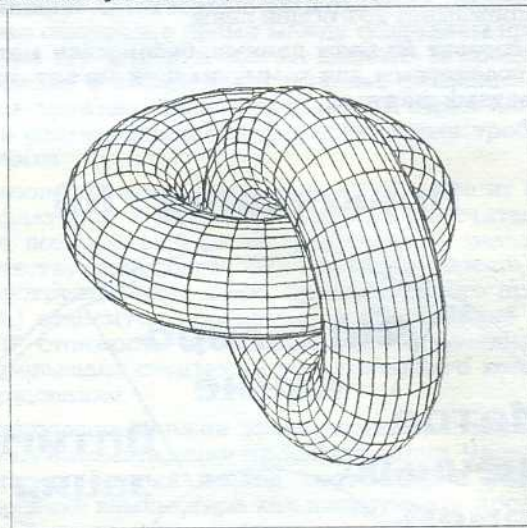
Столь тщательно продуманная организация труда полностью оправдалась. Но для того чтобы соблюсти сроки, работать приходилось больше восьми часов в день.

### Экономия

Очень много времени удалось сэкономить благодаря тому, что архитектурно-строительные проекты передавались всем специалистам сразу же после их завершения. Не встречалось, наверное, ни одной разработки, которая обошлась бы без переделки проекта здания. И на этот раз в проекты отдельных строений пришлось добавить новые стены. При этом, конечно, пришлось внести исправления в проекты, ранее переданные архитектурно-строительной группой другим специалистам. Но современная техника способствовала тому, что никто не работал с устаревшими чертежами дольше нескольких часов.



Кроме того, выяснилось, что проекты канализационной и противопожарной сети требуют по 30 чертежей каждый. Сразу было принято решение об объединении обеих сетей на одних и тех же чертежах. Для этого дополнительно потребовалось всего лишь 8 часов работы, зато было сэкономлено время черчения 30 листов. Благодаря этому весь проект составил уже не 199, а 169 листов.



Было обнаружено, что конструкторы тратят слишком много времени, многократно повторяя одни и те же операции при описании балок и стоек. В меню программы была введена новая опция, позволяющая выполнить всю последовательность этих операций одним нажатием клавиши. Подобная оптимизация конструкторского труда проводилась и в других случаях. Она привела к дополнительной экономии минут, часов, дней кропотливого труда.

### Финал

Согласно плану 9 октября весь проект, включающий 169 чертежей и их полное описание, был готов. С августа в фирме Amman & Whitney работа, что называется, кипела. За несколько дней до истечения срока можно было даже позволить себе на небольшую передышку и замедление ударного темпа работы. Лишний раз подтвердилась тривиальная мысль о том, что располагая хорошими орудиями труда и усовершенствовав организацию труда, любое задание можно выполнить в немыслимый, казалось бы, срок.

Перевод Халины Мадейчик



\ в мире "Компьютера" \

© Юрий Чернышов

# Нечеткие компьютеры

## 90-е годы - эра компьютеров пятого поколения

Последнее десятилетие XX века будет отмечено знаменательным событием в области вычислительной техники - появлением компьютеров пятого поколения. С технологической точки зрения эти компьютеры продолжают линию развития элементной базы: электронные лампы - транзисторы - интегральные схемы - большие интегральные схемы - сверхбольшие интегральные схемы, и в этом смысле смена поколений давно ожидалась. Однако с точки зрения архитектуры - это качественно новые компьютеры, способные выполнять "разумные" функции, что для многих специалистов в начале 80-х годов, когда стало известно о японском проекте, явилось неожиданностью.

Главной целью разработки новой архитектуры компьютеров пятого поколения было стремление преодолеть кризис программного обеспечения [1]. Этот кризис затрагивал как разработчиков программного обеспечения, что приводило к задержке внедрения новых компьютеров, так и пользователей, которые вынуждены (к сожалению до сих пор) долго учиться общению с компьютером, что ограничивало широкое распространение вычислительной техники особенно в тех областях, где проблемы с трудом поддаются математической формализации (медицина, маркетинг, социология и др.).

Причина такого кризиса более чем очевидна - это существование глубокого разрыва (или семантического пробела) между языками, которыми пользуются компьютеры, и естественными языками. Исследуя истоки этого разрыва, японские специалисты пришли к следующему выводу: отойти от традиционной архитектуры компьютеров, спроектировав их на совершенно новых принципах, а именно на принципах обработки знаний. Вспомним, что универсальность существующих компьютеров достигается за счет определенного набора команд и независимого механизма их обработки. Проблема лишь в преобразовании решаемой задачи в последовательность таких команд (но эта проблема, как указано выше, не так проста). Точно так же в компьютерах пятого поколения универсальность должна быть достигнута за счет представления знаний в некотором унифицированном формате и создания механизма их обработки (машины логических выводов). При этом любая поставленная человеком задача может быть решена компьютером, разумеется, при условии, что он (компьютер) имеет необходимые для ее решения знания.

Достаточно подробный анализ архитектуры компьютеров пятого поколения приведен в [1] и книге Дж. Симонса [2]. Кроме того, в Японии вышла книга авторов проекта, в которой рассмотрены основные концепции, программная среда и архитектура первых компьютеров пятого поколения - машины последовательных выводов PSI и машины базы знаний DELTA [3].

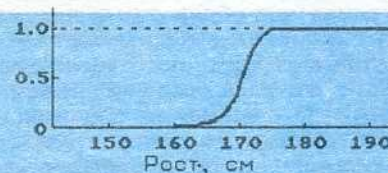
Еще раз отметим, что пятое и последующие поколения компьютеров определяются главным образом не технологической базой, а архитектурой, ориентированной на знания и их использование. С этой точки зрения новые технические идеи - оптические компьютеры, биочипы, трехмер-

ные интегральные схемы и т.д. значительно повысят быстродействие и другие характеристики компьютеров. Однако, по-видимому, не они будут определять свойства будущих поколений. Поэтому перейдем к новым идеям в области архитектуры компьютеров.

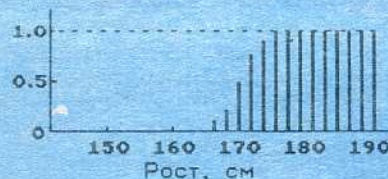
## Нечеткие знания и выводы

Одним из распространенных способов представления знаний является продукционные правила типа "если...то" [4]. Они удобны для поиска цепочки правил, ведущих от фактов к цели или от цели к фактам, выбранным из базы знаний. Подобный способ применяется во многих экспертных системах (в частности хорошо известной системе MYCIN), а также лежит в основе языка Пролог. Но существует огромное множество знаний, имеющих нечеткую, вероятностную природу. Например, в области медицины это знания: "Если высокая температура держится несколько дней, то возможно воспаление легких" или "Если после еды медленно снижается количество сахара в крови, то у больного инсулиновая недостаточность". Для подобных знаний, скажем, в системе MYCIN введен коэффициент уверенности (Cf) со значением от -1 до 1 (от ненадежных знаний до достоверных знаний). Этот способ достаточно прост, но в какой-то степени субъективен. К тому же Cf определяет все правило, а что же делать с такими понятиями, как высокая температура, несколько дней, медленно и т.д.?

В 1965 г. Л. Заде ввел понятие нечеткого множества, которое можно описать в виде так называемой функции принадлежности. Эта функция ставит в соответствие числовым значениям некоторого понятия (элементам нечеткого множества) числа из отрезка [0, 1]. Например, понятие "высокий человек" графически можно представить, как на рис. 1,а.



а)



б)

Для удобства можно сделать дискретную выборку (рис. 1,б) и сформировать вектор:

$$W_{\text{tall}} = (0, 0, \dots, 0.1, 0.2, \dots, 0.9, 1, 1, \dots).$$

Нечеткий дедуктивный вывод с использованием продукционного правила можно записать следующим образом:

$$\begin{array}{l} \text{(знание) если } x \text{ есть } A, \text{ то } y \text{ есть } B \\ \text{(факт) } x \text{ есть } A' \\ \hline \text{(вывод) } y \text{ есть } B' \end{array} \quad (1)$$

где  $A, A', B, B'$  - нечеткие понятия (множества). Если их представить в виде векторов

$$A = (a_1, a_2, \dots, a_M), \quad 0 \leq a_i \leq 1, \quad (2)$$

$B = (b_1, b_2, \dots, b_N), \quad 0 \leq b_i \leq 1$ , то можно определить матрицу нечетких отношений между  $A$  и  $B$ :

$$r_{11} \dots r_{1N}$$

$$R = \dots$$

$$r_{M1} \dots r_{MN}$$

Результат вывода по формуле (1) определяется как

$$r_{11} \dots r_{1N}$$

$$B' = (b'_1, \dots, b'_N) = (a'_1, \dots, a'_M) * \dots$$

$$r_{M1} \dots r_{MN}$$

или

$$b'_j = \bigvee_i a_i * r_{ij},$$

$$i$$

(3)







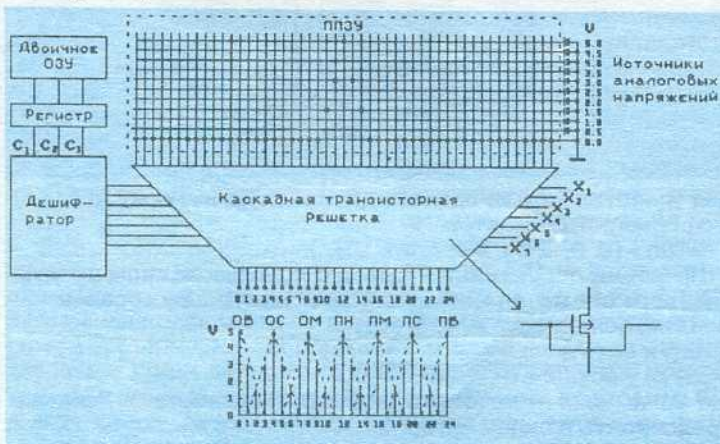
где \* - операция получения минимума. При выводе с помощью цепочки правил (1) конечный результат определяется как операция получения максимума. Примеры выводов и другие способы определения нечетких знаний приведены в [4].

Разумеется, трудно изложить кратко всю теорию. Существуют нечеткие множества, определенные на структурах, лингвистических переменных, булевой алгебре, введен ряд операций над множествами (так, в качестве операции \* используют алгебраическую сумму) и т.д. Приложения теории нечетких множеств дают отличные результаты, особенно в области управления сложными технологическими процессами (например, доменным процессом или прокаткой трубы), автомобилем или поездом, в области маркетинга и т.д. [5].

### Нечеткие компьютеры

Как уже отмечалось, компьютеры пятого поколения будут делать "разумные" выводы на основании заложенных в них знаний. Причем для реализации традиционных дедуктивных выводов необходима операция отождествления, или поиска по образцу. Другими словами, для вывода с помощью правила "если  $x$  есть  $A$ , то  $y$  есть  $B$ " необходимо прежде всего проверить, существует ли в базе знаний факт " $x$  есть  $A$ ". Для нечетких выводов необходим поиск близких по значениям нечетких фактов " $x$  есть  $A$ ", т.е. так называемое нестрогое отождествление (soft matching) одной нечеткой информации с другой. Нечеткие выводы в принципе можно реализовать программно на существующих цифровых компьютерах, но на такую обработку потребуется много времени и непроизводительных накладных расходов. Поэтому разрабатывают специализированные аппаратные средства для обработки нечетких множеств, которые называют нечеткими компьютерами.

Рассмотрим архитектуру одного из первых нечетких компьютеров UFM-1, который создал Т. Ямакава в университете Кумамото [6]. В отличие от традиционных двоичных слов в таком компьютере данные представляются в виде нечетких слов - векторов со значениями элементов от 0 до 1 (2), которым соответствуют уровни напряжения от 0 до 5 В. Для хранения нечетких слов, т.е. функций принадлежности, сконструирована специальная нечеткая память (рис. 2).

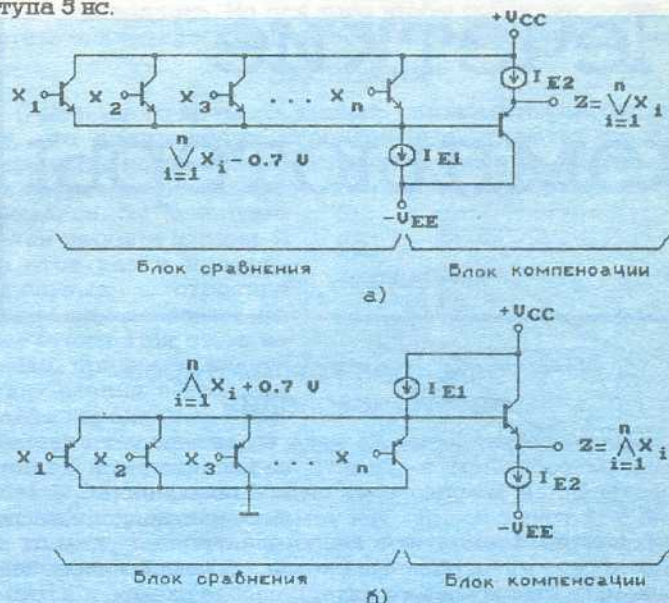


В такой памяти можно хранить данные в виде следующих семи "уровней":

- ПБ - положительное большое,
- ПС - положительное среднее,
- ПМ - положительное малое,
- ПН - почти нулевое,
- ОМ - отрицательное малое,
- ОС - отрицательное среднее,
- ОБ - отрицательное большое.

Память состоит из ППЗУ, в которую пользователь "прошивает" функцию принадлежности "почти нулевых" данных, и внешнего ОЗУ для хранения эквивалентов указанных выше нечетких данных (от 111 до 001 соответственно). Блок дешифровки преобразует их в семибитное слово, а каскад

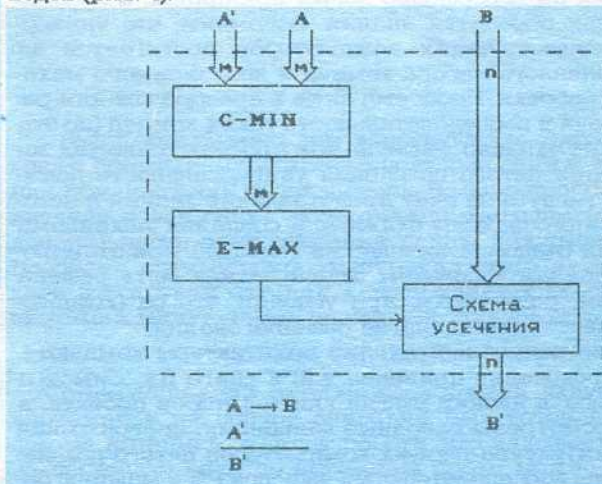
ная транзисторная решетка (pass transistor array) выдает на 25 выходных контактов нечеткое слово в виде уровней напряжения от 0 до 5 В. Чип нечеткой памяти ULF8206, разработанный на базе p-MOS-технологии, имеет время доступа 5 нс.



Нечеткие логические функции (известно несколько десятков таких функций) в отличие от традиционной двоичной логики можно определять различными способами. Особенно часто используют функции нечеткой логической суммы (MAX) и нечеткого логического произведения (MIN). На рис. 3 показаны схемы функций MAX и MIN, работающие в режиме напряжения. Значения истинности 0...1 соответствуют напряжениям 0...5 В.

Схемы состоят из блоков сравнения и компенсации. Первые сравнивают входные напряжения, и на объединенных эмиттерах матриц pnp- или rnp-транзисторов формируются минимальное (-0.7 В) либо максимальное (+0.7 В) значения напряжений. Поэтому схемы называют нечеткими логическими вентилями с объединенными эмиттерами. Напряжение 0.7 В на эмиттерах компенсируется на следующем каскаде, и на выходе Z выдается минимальное или максимальное значение. Данные вентили не критичны к напряжениям источников питания, которые могут быть от 6 до 51 В (Vcc) и от -1 до -46 В (Vee), что, разумеется, положительно сказывается на их надежности.

На базе таких вентилях строится машина нечетких выводов (рис. 4).

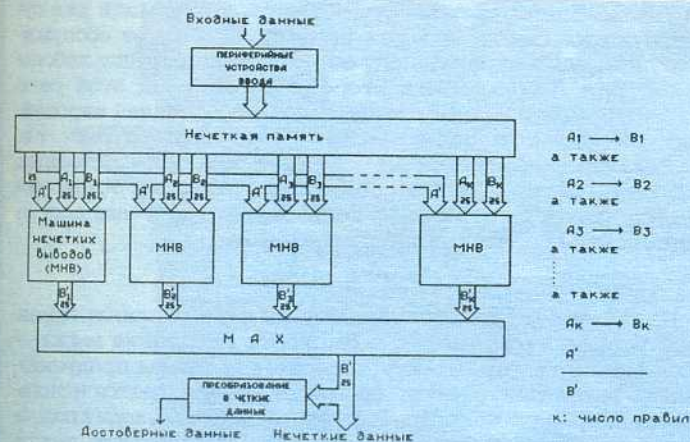


Здесь C-MIN состоит из  $m$  параллельных двухвходовых схем MIN, E-MAX - из  $m$  параллельных одновходовых схем MAX, схема усечения - из  $p$  параллельных двухвходовых схем MIN. Если из базы знаний выбрано продукционное правило "если  $x$  есть  $A$ , то  $y$  есть  $B$ " и нечеткое знание " $x$  есть  $A$ ", то на  $p$  сигнальных выходах этой машины появят-



ся уровни напряжения, соответствующие нечеткому выводу В. Таким образом, машина будет выполнять обработку знаний по формуле (3). Быстродействие такой машины составляет 100 нс, т.е. она может делать до 10 миллионов FIPS (нечетких выводов в секунду).

Очевидно, что максимальная эффективность нечеткого компьютера достигается за счет параллельной работы нескольких машин нечетких выводов (рис. 5).



Каждое нечеткое слово передается в виде уровней напряжения по сигнальным линиям (на рисунке 25 линий) сразу на все машины, что позволяет применить одновременно к продукционным правилам. Блок МАХ реализует логическую сумму к нечетким выводам. Полученный вывод необходимо преобразовать в достоверные данные, что выполняется с помощью специального блока преобразования. Для простоты шина и блок управления на рисунке не показаны.

По структуре данных, архитектуре аппаратных средств и алгоритмам выводов нечеткий компьютер резко отличается от существующих машин логических выводов, поэтому его смело можно назвать компьютером шестого поколения.

### Нечеткое программное обеспечение

Нечеткие случайные процессы издавна были объектом программирования на цифровых компьютерах. Создано немало программ для нечеткого управления, нечеткого математического планирования, нечеткого поиска библиографии и др. Все такие программы написаны на традиционных языках программирования. Но без сомнения, непосредственное манипулирование нечеткими множествами как средство языка откроет новые интересные приложения компьютерных систем.

Нечеткие множества, которые можно представить в виде дискретной функции принадлежности, несложно описать на любом языке, особенно на языках обработки списков. Однако в целом операции над нечеткими множествами будут достаточно громоздкими. Значения функции принадлежности, вообще говоря, определяются субъективно, и вычисления не нужно выполнять с высокой точностью. В этом смысле представление нечетких множеств на существующих цифровых компьютерах избыточно. Кроме того, на такие вычисления уходит довольно много времени, поскольку нечеткие множества обрабатываются последовательно элемент за элементом. Решить проблему быстродействия можно за счет проектирования специальных аппаратных средств, в которых значения функций принадлежности представляются в аналоговом виде или в упакованном цифровом виде, а обрабатываются параллельно, как в нечетком компьютере. Однако до настоящего времени каких-либо серьезных исследований связи нечетких программных средств с нечеткими компьютерами не ведется.

Одна из программных систем обработки нечетких множеств создана в Японии для персонального компьютера серии PC-9800 с использованием языка Лисп [7]. Перечислим особенности этой системы:

- \* позволяет описать почти все нечеткие множества и отношения;
- \* предусматривает встроенные функции обработки нечетких множеств;
- \* определяет собственные функции такой обработки;
- \* обеспечивает совместимость с традиционными программами, написанными на Лиспе.

Нечеткие функции представляются как

$[0.1/a, 0.2/b, 0.3/c, \dots]$ ,

а нечеткие отношения —

$[0.3/\langle a, 1 \rangle, 0.9/\langle b, 2 \rangle, 0.3/\langle c, 3 \rangle, \dots]$ .

Введены операции присвоения  $\text{setd}$ , объединения (логической суммы)  $\text{union}$ , пересечения (логического произведения)  $\text{intersection}$  нечетких множеств. Функция  $\text{imply-arithm}$  образует нечеткое отношение множеств  $V$  и  $U$  из продукционных правил типа (1), если  $A$  и  $A'$  принадлежат  $V$ , а  $B$  и  $B' \in U$ . Нечеткие выводы выполняются с помощью функции  $\text{image}$ .

Вместе с тем данная система ориентирована на детерминированные алгоритмы обработки нечетких множеств. Для эвристических алгоритмов, включающих поиск нечетких знаний в базе и автоматическое применение продукционных правил, необходимы более мощные системы типа языка Пролог. Рассмотрим принципы обработки нечетких множеств на примере нечеткого Пролога [8].

Факты в базе знаний нечеткого Пролога представляются в виде функций принадлежности. Например, факт "высокий человек" (рис. 1) можно описать следующим образом:  $\text{tall (Height) } [0/(0), 0/(160), 0.1/(166), 0.2/(168), 0.5/(170), 0.8/(174), 0.95/(178), 1/(180), 1/(225)] \text{ repeat } \#9$ .

Здесь в скобках указан рост человека, а перед знаком "/" — значение функции принадлежности. Аналогично можно представить такие факты, как "молодой"  $\text{young (Age)}$ , "обеспеченный"  $\text{rich (Money)}$  и другие. В частности, приведенные факты позволяют легко построить "службу знакомств". Сведения о желающих найти друга составляются в виде фактов, например:

$\text{/* молодая девушка, невысокая, зарплата небольшая */}$   
 $\text{girl ('Марина', 21, 155, 120).}$

$\text{/* мужчина, немолодой, высокий, обеспеченный */}$   
 $\text{boy ('Андрей', 38, 176, 380).}$

Пожелание Марины встретить немолодого, высокого, обеспеченного мужчину можно представить в виде правила

$\text{boy\_friend ('Марина', Boy) :- boy (Boy, Age, Height, Money),}$   
 $\text{tall (Height), not (young (Age)), rich (Money).}$

Сделав запрос нечеткому Прологу

$? \text{- boy (Boy, \_, Money), rich (Money),}$

можно организовать поиск только обеспеченных мужчин вне зависимости от роста и т.д. Разумеется, среди найденных будет и Андрей:

$(\text{Boy} = \text{'Андрей'}, \text{Money} = 380) [\text{MT} = 1.0, \text{T} = 1.0 \text{ with CONF} = 1.0]$ ,

где  $T$  — значение истинности ответа,  $\text{CONF}$  — степень достоверности вывода, а  $\text{MT}$  — обобщенное значение истинности, рассчитанное по формуле

$$\text{MT} = (2T - 1) * \text{CONF} * 0.3 + 0.3.$$

Таким образом, благодаря нечеткому Прологу можно обрабатывать утверждения даже со значениями истинности менее 0.5, что в традиционном Прологе невозможно.



1. Труды Международной конференции по компьютерным системам пятого поколения. — В 7-ми т. — Токио: 19-22 окт. 1981 г., JPRDC. — Т. 5.
2. Симонс Дж. ЭВМ пятого поколения: компьютеры 90-х годов. — М.: Финансы и статистика, 1985.
3. Введение в ЭВМ пятого поколения / Под ред. К. Фути. Токио: Омск, 1987.
4. Представление и использование знаний / Под ред. А. Уэно, М. Исидрука. — М.: Мир, 1989.
5. Терао Т., Асаи К., Сугано С. Прикладные нечеткие системы. — Токио: Омск, 1989.
6. Yamakawa T. A simple fuzzy computer hardware system employing MIN & MAX operations. — Proc. 2nd IFSA Congress, Tokyo, July 1987.
7. Умано М. Система обработки нечетких множеств на Лиспе: сб. тр. 3-го симпозиума по нечетким системам. — Токио, 1987. — С. 167-172.
8. Мукайдов М. Что такое нечеткий Пролог? // Computer today, 1988. — N 25. — P. 4-46.



\ в мире "Компьютера" \

Обработка Халины Мадейчик

# EUROMATH

Математики объединяющейся Европы также решили объединить свои силы. Помочь им в этом должна EUROMATH - европейская сеть математиков. Никто еще не принимался за реализацию столь грандиозного замысла, исходя из принципа "сделай сам". Никто не будет проектировать и строить сеть для математиков. Они должны это сделать собственноручно. Назначение сети - оптимизация организации международного сотрудничества и обмена научной информацией. К работе привлечено более 10 000 математиков из стран ЕЭС.

Для реализации проекта, который сводится в принципе к объединению при помощи компьютерной сети всех важнейших научных центров, в 1987 г. была организована неправительственная международная организация "Европейский математический трест" (ЕМТ - European Mathematical Trust). Ее председателем является Ф. Топсе из Института математики Копенгагенского университета.

В каждой из 19 европейских стран, участвующих в реализации проекта, создан национальный Координационный комитет. ЕМТ предоставляет юридическую базу для координации деятельности разработчиков из различных стран и для централизованного финансирования различных групп и бригад разработчиков. Со временем наряду с чисто координационной деятельностью, связанной собственно с реализацией проекта, ЕМТ будет также выполнять функции посредника по связям с различными заинтересованными фирмами и организациями. Трест намерен распространять по сети разнообразные программные продукты, представляющие интерес для пользователей-математиков. При этом накладывается одно условие: эти продукты должны соответствовать стандартам сети. Стандарты, равно как и описания пользовательского интерфейса, будут опубликованы.

Проекты, подобные Euromath, требуют обычно больших расходов. По предварительным данным только технические средства связи сети обойдутся в несколько миллионов экю. Финансирование работ на начальном этапе обеспечивается за счет Комиссии ЕС (85%) и правительства Дании (15%).

## Работа кипит

Идея возникла в 1983 г. Окончательный план работ был утвержден в начале 1988 г. Техническая часть проекта должна быть завершена в декабре 1991 г. Экспериментальная эксплуатация сети будет продолжаться год. На начальном этапе была исследована предметная область, определен функциональный состав системы и реализованы программные средства, обеспечивающие работу с данными и управление сетями связи.

Следующий, основной этап предполагает создание важнейших элементов системы Euromath: системы подготовки документации, информационно-поисковой системы, протоколов интеграции и т.п.

Предварительный этап реализации проекта уже завершен. В настоящее время проводится опытная эксплуатация сети. Предполагается, что эксплуатация в промышленных масштабах будет начата в конце будущего года. Основная цель настоящего этапа состоит в том, чтобы продемонстрировать в реальных условиях возможности современной инструментальной среды, спроектированной и реализованной с учетом потребностей четко определенного круга пользователей-ученых.

## Архитектура EUROMATH

Система строится по принципу открытой архитектуры, что облегчает ее дальнейшее развитие и расширение, а также предполагает высокую степень децентрализации. Такой подход позволяет существенно снизить затраты на связь и обеспечивает гибкость при обработке больших потоков данных. Ядро системы образуют несколько мощных ЭВМ, расположенных в своих регионах. Связь между этими узлами планируется осуществлять при помощи уже существующих сетей. Каждая из этих ЭВМ будет обслуживать либо одну страну, либо ее часть, либо группу небольших стран. Техническая сторона реализации этой сети осуществляется в сотрудничестве с Ассоциацией европейских научно-исследовательских сетей. В узлах сети, т.е. на ЭВМ, формирующих ядро, будут храниться основные массивы данных и программные средства. Установив связь с любой из этих ЭВМ, пользователь получает доступ к удаленным базам данных.

## Центр сети

Наличие единого, постоянного центра является гарантией успешного развития и быстрого внедрения технических новшеств - архитектура системы обязана предусматривать такой центр, если систему предполагается использовать всерьез и надолго. В функции центра входит сопровождение системного программного обеспечения, техническое и коммерческое обслуживание системы, осуществление мероприятий по ее дальнейшему развитию и совершенствованию. В соответствии с предложением правительства Дании ЕМТ принял решение разместить центр сети Euromath на территории этой страны.

## Залог успеха - интеграция функций

Функции сети можно подразделить на 3 следующих широких класса:

- поддержка личного общения между пользователями;
- информационный поиск;
- подготовка и тиражирование документов.

Реализация каждой из перечисленных функций по отдельности на современном этапе не представляет никаких технических трудностей. Особенность системы заключается в том, что в ней все эти функции объединены для оптимального обеспечения потребностей пользователей-математиков. Важно и то, что сеть обеспечивает междисциплинарные связи. Как известно, математика - основа всех наук. Ее язык и инструменты служат ученым и специалистам, работающим во многих областях. Сеть может оказаться полезной справочной и рабочей средой для пользователей этой категории.

## Планы на будущее

Можно ожидать, что уже через сравнительно короткое время все европейские организации, имеющие интересы в области математики, смогут предоставить своим сотрудникам доступ к современным рабочим станциям. В этом случае абонент системы Euromath сможет воспользоваться всеми ее функциональными возможностями: от разного рода вычислений до издания высококачественной документации. Однако общая идеология проекта требует также учитывать интересы тех пользователей, в распоряжении которых нет столь совершенного терминального оборудования. Эта проблема пока еще не решена.

В дальнейшем планируется расширить рамки системы за пределы Европы и превратить ее во всемирную сеть. Особый интерес представляет включение в нее развивающихся стран. Это обеспечит, по мере развития средств коммуникации, приобщение малых и слаборазвитых стран к передовым достижениям математики без больших капитальных вложений и создания собственных крупных центров по накоплению и переработке знаний. Такой подход позволит ускорить научный прогресс в тех регионах, где его уровень еще недостаточен. Одно из первых новых национальных звеньев сети Euromath возникает в СССР.



[illegible]

# Технология баз данных

© Михаил Коголовский



Во втором номере сборника мы познакомили вас с общим состоянием рынка программного обеспечения систем баз данных для ПЭВМ. Было отмечено функциональное многообразие используемых на ПЭВМ средств управления данными, рассмотрены методы обеспечения их дружественного характера. Проведен анализ моделей данных, поддерживаемых "персональными" СУБД, особенностей их архитектуры и возможностей представления результатов обработки пользовательских запросов.

Сейчас предлагаем вашему вниманию заключительную часть обзора.

## Инструментарий администратора баз данных

Администратор базы данных играет довольно ответственную роль в системе баз данных. К его функциям, как известно, относятся начальная загрузка данных в базу данных, управление доступом к данным, защита физической целостности базы данных и восстановление ее при разрушениях, реструктуризация и реорганизация базы данных, обеспечение приемлемого уровня производительности системы. Выполнение этих функций в социальной пользовательской среде — довольно сложная проблема.

Однако весьма часто речь идет не об использовании базы данных в обстановке локальной сети ПЭВМ в мультипользовательской среде, а о работе в локальном режиме. При этом функции пользователя и администратора базы данных обычно совмещаются в одном лице, а если и возникает мультипользовательская среда, то взаимодействие различных пользователей с системой базы данных осуществляется не параллельно, а в последовательно распределенных во времени сеансах. В такой ситуации функции администратора базы данных существенно упрощаются.

В этой связи СУБД для ПЭВМ, предназначенные для работы в локальном режиме, и локальные версии сетевых СУБД чаще всего не располагают традиционными инструментальными средствами администрирования данными, имеющимися в составе каждой СУБД на больших вычислительных системах.

Нужно отметить, что в реляционных СУБД выполнение многих функций администрирования данными к тому же существенно упрощается. Так, начальную загрузку базы данных можно легко выполнить, последовательно загружая данные в таблицы базы данных с помощью обычных средств импорта данных из ASCII-файлов, которые имеются в большинстве систем. Такие файлы могут быть подготовлены с помощью какого-либо из многочисленных текстовых редакторов либо получены путем экспорта из какой-либо другой системы. Загрузка данных в таблицы может осуществляться в любом порядке, если этому не препятствуют какие-либо специфицированные мультитабличные ограничения целостности. Это обстоятельство нужно учитывать в системе R:base for DOS.

Сложнее обстоит дело с СУБД, оперирующими с базами данных сетевой структуры. В этой связи в системах db\_Vista и MDBS-III (mdbis Inc.) для администратора базы данных предусмотрены утилиты инициализации и начальной загрузки базы данных [1].

Имеются две разновидности средств управления доступом пользователей к базе данных. Первая из них — управление доступом с помощью паролей. Обычно пароли устанавливаются в реляционных СУБД на уровне базы данных и на уровне отдельных таблиц дифференцированно по классам операций манипулирования данными.

Особыми полномочиями обладает владелец данных. Только предъявляя пароль владельца, можно модифицировать структуру или произвести выгрузку базы данных, назначить или изменить пользовательские пароли. Такой подход к управлению доступом реализован, например, в системе R:base for DOS.

Аналогичный подход, но с более высокой разрешающей способностью, реализован в сетевой версии системы Paradox 2.0. Для этой цели имеется специальный модуль

системы — Paradox Protection Generator. Помимо паролей на уровне базы данных и таблиц здесь предусматривается также защита данных на уровне полей. Может быть также ограничен доступ к объектам, ассоциированным с данной таблицей — формам ввода/вывода, отчетам и т.д.

В системе dBaseIII PLUS (сетевая версия) управление доступом может быть установлено на уровне доступа к самой СУБД при входе в систему, на уровне файлов базы данных или полей записей файлов, а также с помощью специального кодирования данных в заданных группах файлов. В последнем случае пользователь должен будет предъявить для доступа к данным ключ для их декодирования.

Другая разновидность подхода к управлению доступом реализована в системах с языком SQL. Здесь используются предусмотренные в языке механизмы передачи полномочий доступа с помощью команд GRANT и REVOKE. Подробности по этому вопросу можно найти в [2].

В некоторых СУБД, эксплуатируемых и в мультипользовательской обстановке, вообще не предусмотрены средства управления доступом, например в системе db\_Vista.

Важнейшая функция администратора базы данных — поддержание ее физической целостности. Разработчики СУБД обычно рекомендуют периодически создавать контрольную копию базы данных и использовать ее для восстановления при разрушениях. Для этой цели в языке системы R:base for DOS, например, предусмотрены команды BACKUP и UNLOAD для выгрузки всей базы данных или ее части вместе с описанием. Эти данные могут быть загружены в базу данных при восстановлении. В других системах можно использовать для этих целей обычные средства экспорта — импорта данных.

Складывается, однако, впечатление, что в СУБД для ПЭВМ не предусматриваются какие-либо более уточненные средства, позволяющие администратору значительно эффективнее выполнять эти функции. Вместе с тем эта проблема уже была решена в СУБД для "больших" вычислительных систем. С этой целью в процессе функционирования системы базы данных системными средствами поддерживается журнал изменений данных, и он используется вместе с контрольной копией при разрушении базы данных для ее восстановления. К сожалению, никакой явной информации об использовании техники журнализации в какой-либо из "персональных" систем в доступных публикациях не приводится. Можно, вероятно, предположить, что в этом отношении исключением является система MDBS-III, которая согласно [1] располагает средствами автоматического восстановления базы данных. Это косвенным образом свидетельствует о наличии журнализации на уровне оператора или транзакции.

Иная ситуация со средствами реструктуризации базы данных. Такие средства, более или менее развитые, имеются в большинстве коммерческих систем.

Практически во всех реляционных СУБД имеется возможность добавлять в существующую базу данных новые отношения (таблицы, файлы) или уничтожать имеющиеся отношения.

В системе dBaseIII PLUS имеются возможности удаления, добавления, переименования и изменения характеристик полей записей файлов базы данных. После такой реструктуризации сохраняются значения тех полей, характеристики которых не подверглись изменениям.

Аналогичными возможностями располагают системы Paradox и R: base for DOS.

Более сложный характер имеет реструктуризация баз данных сетевой структуры. Однако в [3] сообщается, что для системы db\_Vista создан модуль db\_Revise, который позволяет легко модифицировать структуру существующей базы данных и, кроме того, конвертировать ее данные в новую среду хранения, в соответствии с пересмотренной структурой. Функция реструктуризации базы данных предусмотрена также в системе MDBS-III.

Реорганизация базы данных осуществляется с помощью ряда процедур, воздействующих на среду хранения



ния базы данных с тем, чтобы изменить размещение данных в пространстве памяти и/или методы доступа. Цели реорганизации - повысить производительность системы, вернуть для повторного использования неиспользуемые в данный момент части пространства памяти базы данных.

В предлагаемых в настоящее время пользователю реляционных системах модификация методов доступа сводится к созданию новых или отказу от некоторых существующих индексов, поддерживающих инверсию таблиц базы данных по каким-либо ключевым столбцам (полям). Такими возможностями обладают многие системы - dBaseIII PLUS, Paradox, R:base for DOS и др.

Для более рационального размещения данных в пространстве памяти после реструктуризации базы данных система R:base for DOS использует команду PACK. Команда с таким же ключевым словом в системе dBaseIII PLUS физически удаляет ранее логически удаленные записи файла базы данных, уплотняя размещение оставшихся записей.

В версии системы Oracle для IBM PC с 286 и 386 микропроцессорами инструментарий администратора базы данных сосредоточен в модуле SQL\*PLUS [4].

### Средства разработки приложений

Диапазон пользователей рассматриваемого класса СУБД распространяется от случайных пользователей или пользователей, способных удовлетворять свои информационные потребности непосредственно с помощью СУБД, до разработчиков приложений, построенных таким образом, что инструментальные средства становятся скрытыми от пользователя. Каждая из СУБД в той или иной мере тяготеет к одному из этих полюсов, а некоторые системы сочетают обе возможности.

Системы, ориентированные на разработчиков, обладают развитыми средствами для создания приложений.

В этом классе средств следует прежде всего отметить появление систем автоматизации проектирования баз данных. Так, сообщается о создании системы Master [5], обеспечивающей решение этой задачи и использующей в качестве инструмента концептуального моделирования модель сущностей-связей Чена [6].

Другой важный элемент инструментария разработки приложений - мощные языки программирования, которыми обладают системы dBaseIII PLUS, R:base for DOS, Paradox, Clipper и т.д. Сюда же следует отнести и SQL, реализованный во многих системах и часто квалифицируемый благодаря многим его достоинствам как язык четвертого поколения. Ряд из этих систем предоставляет разработчику удобную среду для написания и редактирования программ. В системах R:base for DOS и Clipper имеются компиляторы, редакторы связей и символические отладчики.

Необходимыми элементами инструментария разработки системы баз данных являются средства реализации меню, экранных форм ввода/вывода данных и генерации отчетов. Такими средствами обладают многие СУБД - dBaseIII PLUS, dBaseIV, R:base for DOS, Paradox, DataEase и др. Для системы dBaseIII PLUS разработан целый ряд сопутствующих программных продуктов, развивающих возможности системы в указанных направлениях - редакторы экранных форм и меню и генераторы кода FlashCode (Software Bottling Comp.) и QuickCode (Fox & Geller), Genifer (Bytel Corp.), UI-Programmer (WallSoft Systems Inc.) [7], генераторы отчетов Relational Report Writer (Concentric Data Systems Inc.) [8] и Quick Report (Fox & Geller).

Наконец, следует указать средства высокого уровня для разработки приложений, которые позволяют создавать сложные прикладные системы практически без программирования.

В этой связи следует упомянуть систему Paradox с его компонентом Paradox Personal Programmer [9], систему R:base for DOS с ее EXPRESS-модулями [10], dBaseIV с ее

средствами генерации приложений, названные выше генераторы программного кода для системы dBaseIII PLUS и ряд других средств.

Нужно отметить, что в технологии персональных баз данных начинает приобретать право гражданства такой инструмент, как словари-справочники данных. Зачатки его, ориентированные не на систему, а на пользователя, можно найти в системе dBaseIII PLUS в виде ее каталога. Дальнейший шаг связан с реализацией в ряде систем - DB2, Oracle, dBaseIV и др. - языка SQL, предоставляющего пользователю доступ как к данным из базы данных, так и к метаданным. По существу, это упрощенный интегрированный словарь-справочник данных. Как сообщается в [11], в полной мере интегрированный словарь-справочник данных реализован в системе Advanced Revelation (Cosmos Inc.).

### Базы данных в сетевой обстановке

В то время как СУБД для отдельных персональных компьютеров предназначены главным образом для одиночных пользователей, и это позволяет существенно упростить их функции, сетевые версии таких систем практически почти уже ничем не отличаются в функциональном аспекте от традиционных систем. Здесь уже приходится иметь дело с мультипользовательской средой и решать связанные с нею специфические проблемы параллельных процессов, управления доступом и защиты целостности данных.

Системы баз данных, построенные с использованием сетевых версий таких массовых СУБД, как R:base for DOS, dBaseIII PLUS, Paradox, Advanced Revelation и других, часто неправомерно называют системами распределенных баз данных. Фактически же мы имеем здесь дело не с распределенными базами данных, а с распределенным (сетевым) доступом к централизованной базе данных.

Такие системы создаются на основе оборудования и программного обеспечения различных локальных вычислительных сетей. Большинство СУБД для ПЭВМ IBM PC может работать в сетях IBM PC Network (IBM Corp.), Novell Network (Novell Inc.) и ряде других.

Архитектура систем баз данных с сетевым доступом [12] предполагает выделение одной из ПЭВМ сети в качес-







ве центральной. Она будет выполнять функции машины для управления файлами (File Server). Имеется в виду, что эта машина обладает жестким диском, на котором хранится совместно используемая централизованная база данных. Все другие машины сети выполняют функции рабочих станций (Work Station), предоставляемых пользователям базы данных. Рабочие станции могут не иметь жесткого диска.

Производительность сетевой системы существенно зависит от конфигурации используемого в ней оборудования. Если рабочие станции, например, не обладают жестким диском, то на диске центральной машины желательно иметь программные средства СУБД, а также приватные пользовательские директории. Поскольку СУБД с сетевым доступом построены из расчета, что вся обработка данных осуществляется на рабочей станции, рассматриваемый вариант конфигурации сети характеризуется большими потоками данных (сетевым трафиком), что отрицательно сказывается как на производительности, так и на надежности системы. Для того чтобы их сократить, желательно иметь жесткие диски на рабочих станциях и хранить на них используемые программные средства. На этих же дисках могут быть выделены приватные пользовательские директории.

Сетевые версии СУБД для ПЭВМ отличаются от локальных их версий именно тем, что они рассчитаны на мультипользовательскую обстановку. Для этого они должны обладать некоторыми специальными механизмами, позволяющими многим пользователям совместно обращаться к общим ресурсам данных из централизованной базы данных. К числу этих механизмов относятся механизмы блокирования ресурсов, благодаря которым возможно производить обновление данных при параллельной работе различных пользователей, а также механизмы управления доступом, обеспечивающие доступ к базе данных в рамках тех полномочий, которые предоставлены конкретным пользователям.

В сетевых СУБД используются разнообразные протоколы блокирования ресурсов. Некоторые системы обеспечивают в определенных ситуациях автоматическое блокирование ресурсов, освобождая разработчика приложения от необходимости заботиться об этом. В других системах (Clipper [13], db\_Vista) имеются только механизмы явного блокирования. Протокол, принятый в системе R:base for DOS, предполагает блокирование до уровня поля. Это позволяет двум пользователям одновременно обновлять одну и ту же строку таблицы, но разные ее поля. Тем самым минимизируется время ожидания доступа к блокированному ресурсу.

Система Advanced Revelation не использует дисциплины автоматического блокирования полного файла, но осуществляет автоматическое блокирование на уровне записи при обновлении.

Представляет интерес техника блокирования ресурсов в системе dBaseIII PLUS. Здесь предусмотрено три уровня блокирования - автоматическое блокирование файла при операциях над полным файлом, явное блокирование файла и явное блокирование записи.

Весьма тонкие механизмы блокирования ресурсов предусмотрены в системе Paradox. Имеется, в частности, такой вид блокирования, когда один пользователь блокирует строку таблицы для обновления, а другой пользователь может тем не менее ее в это время читать. Работа в таком режиме поддерживается возможностями "освежения" (refreshing) экрана рабочей станции с заданной частотой. Paradox позволяет также информировать пользователя о том, кто в данный момент блокирует запрашиваемые им данные.

Выбор дисциплины блокирования ресурсов оказывает заметное влияние на производительность всей системы. Разработчик приложения должен выбирать всегда наименьший возможный в каждой ситуации ресурс для блокирования с тем, чтобы в минимальной степени сдерживать пользователей параллельно работающих приложений.

Сложной проблемой, связанной с мультипользовательским режимом работы с базами данных, является проблема тупиковых ситуаций (Deadlock) [14]. В технологии баз данных на больших и мини-ЭВМ предусматривается специальная техника профилактики возникновения тупиковых ситуаций и отката транзакций при их возникновении. Инструментарий баз данных для ПЭВМ не предусматривает каких-либо специально предназначенных для этого системных механизмов. Однако обеспечиваются возможности для приложений направлять специальные сообщения о намерениях, связанных с блокированием ресурсов, поддерживаемых для этих целей системой "почтовый ящик", а также получать информацию из такого "почтового ящика" об установленных в данный момент блокировках. Подобная информация позволяет каждому приложению соблюдать "джентельменскую линию поведения" для того, чтобы исключить возникновение тупиковых ситуаций.

Еще одна проблема, имеющая отношение к рассматриваемому вопросу - управление доступом к базе данных в мультипользовательской среде локальной сети ПЭВМ. Мы не будем здесь обсуждать эту проблему, поскольку она уже была рассмотрена в п.8.

В заключение этого раздела нужно заметить, что в последнее время происходит некоторая трансформация подходов к использованию баз данных в сетевой среде, направленная на повышение роли машины для управления файлами (File Server) в локальной сети ПЭВМ [15]. Если существующие СУБД отводят этой машине, главным образом, роль хранилища базы данных, то в соответствии с новыми тенденциями на ней предусматривается выполнение основного объема обработки данных, и рабочие станции будут уже получать не "сырье" для последующей обработки, а готовые результаты. При этом значительно сокращается сетевой трафик и разгружаются рабочие станции. Кроме того, в случае использования такого мобильного языка, как SQL, появляется возможность комплексировать в единой сети большие вычислительные системы, мини-ЭВМ и ПЭВМ, и пользователь вообще не должен будет знать, где находятся обрабатываемые его транзакцией данные.

Для использования указанного подхода требуются новые инструментальные средства. Целый ряд фирм - разработчиков программного обеспечения систем баз данных - объявил о создании такого рода инструментария. Значительная его часть ориентируется на язык SQL. Так, фирмы Microsoft, Sybase Inc. и Ashton-Tate Inc. объявили о совместном продукте SQL Server, предназначенном для этой цели и ориентированном на среду OS/2. Этот продукт будет поддерживаться системой dBaseIV. В конце 1989г. ожидаются аналогичные продукты IBM Corp., Oracle Corp., Micromin Inc. С продуктом Oracle будут сопрягаться Paradox и QuickSilver, с разработками IBM и Microsoft - система DataEase. Фирма Micromin Inc. планирует создание собственного продукта ATLAS-II, работающего под OS/2 и включающего поддержку SQL для распределенной обработки [16] на больших и мини-ЭВМ, а также на персональных рабочих станциях.

## Обмен данными с другими системами

Разработчики СУБД для ПЭВМ особое внимание уделяют проблемам обмена данными с другими программными системами - с системами баз данных, с системами "электронных таблиц", с различными прикладными системами.

Нужно отметить, что средства обмена данными с внешними системами на ПЭВМ стали неотъемлемым компонентом большинства программных продуктов, связанных с обработкой данных. Более того, некоторые форматы файлов стали де-факто стандартами обменного формата данных - это форматы SDF (текстовые файлы с фиксированным форматом записи без разделителей между полями), ASCII-формат с разделителями между полями, DIF (Data Interchange Format - формат файлов пакета VisiCalc) и др. Средства экспорта-импорта данных в развитых СУБД настолько развиты, что практически не возникает каких-либо проблем обмена данными между компонентами инстру-





ментальных комплексов, предназначенных для реализации сложных информационных технологий.

Для иллюстрации приведем сведения об обменных форматах файлов нескольких популярных СУБД.

Система dBaseIII PLUS:

SDF, ASCII, PFS:Professional File.

Система dBaseIV:

SDF, ASCII, PFS:Professional File, dBaseII, FrameworkII, Rapid File, DIF, SYLK (MultiPlan), WKS (Lotus 1-2-3)

Система Paradox 2.0

ASCII, dBaseII, dBaseIII, dBaseIII PLUS, Lotus 1-2-3, Symphony, PFS:Professional File, DIF.

Система DataEase:

ASCII, SDF, DIF, MultiMate, Lotus 1-2-3, dBaseIII (импорт)

Система R:base for DOS:

ASCII, SDF, Lotus 1-2-3, Symphone, dBaseII, dBaseIII, MultiPlan, DIF.

Нетрудно видеть, что перечисленные реляционные СУБД могут довольно легко обмениваться данными. С большинством упоминаемых здесь форматов файлов можно познакомиться в руководстве [17].

### Мобильность программного обеспечения систем баз данных

Целый ряд программных продуктов, используемых в технологии баз данных на персональных ЭВМ, имеет различные версии, разработанные для различных моделей ЭВМ и различных операционных систем. Приведем в подтверждение несколько примеров.

Система Oracle: реализована для ЭВМ IBM 360/370, VAX и IBM PC различных моделей, в том числе для IBM PC с процессором 386. Работает под операционными системами MS-DOS, VM, MVS, VAX/MVS, UNIX.

Система Paradox: IBM PC/XT, IBM PC/AT и IBM PC/386, в среде MS-DOS и OS/2.

Система db\_Vista: на IBM PC/XT, IBM PC/AT, IBM PC/386, VAX в среде MS-DOS, UNIX, XENIX, VMS, OS/2.

Система dBaseIV: IBM PC и PS/2 в среде MS-DOS и OS/2.

Эти примеры можно было бы продолжить, но они уже достаточно красноречиво говорят о благоприятной обстановке для решения проблемы мобильности программного обеспечения систем баз данных, основанных на перечисленных СУБД. Что касается конверсии в новую операционную обстановку самих данных хранимых в базе данных, то, как показано в предыдущем разделе, эта задача решается достаточно легко.

Особо благоприятные условия имеются для мобильности приложений, реализованных на языке SQL, поскольку здесь обеспечивается уже в значительной мере независимость приложения от конкретной СУБД.

Немаловажное обстоятельство заключается в том, что некоторые СУБД имеют реализации на различных классах ЭВМ, что позволяет в значительной мере добиться независимости от класса ЭВМ. Система Oracle - на больших ЭВМ IBM 360/370, мини-ЭВМ - VAX и персональных компьютерах. Система DB2 - на IBM 360/370 и OS/2, система INGRES - на PDP/11 и IBM PC.

### Борьба за производительность СУБД

Быстрая эволюция микро-ЭВМ, повышение их производительности и увеличение емкости жестких магнитных дисков позволяют создавать системы баз данных даже в локальном варианте с объемами данных, достигающими сотен мегабайт. Вместе с тем приобретают первостепенную важность проблемы повышения производительности самих СУБД, используемых в таких условиях. Для решения этих проблем предпринимаются различные шаги.

Прежде всего разрабатываются компиляторы для СУБД, работающие в режиме интерпретации. Так, для систем dBaseIV и R:base for DOS их разработчиками - фирмами Ashton-Tate и Microrim Inc. - объявлен выпуск на рынок программного обеспечения компиляторов языков этих систем. Ранее несколькими фирмами был создан ряд компиляторов и псевдокомпиляторов языка системы dBaseIII PLUS, обладающих существенно более высокой производительностью по сравнению с работой dBaseIII PLUS в режиме интерпретации - Clipper [13,20], Foxbase+ [21], Quick Silver и др.

Совершенствование используемых в СУБД методов доступа к данным в базе данных - другой важный путь повышения производительности СУБД. Хорошо известно, что в системах Clipper и Foxbase+ значительно более эффективно, чем в dBaseIII PLUS, построены механизмы индексирования файлов базы данных, и это является причиной достижения существенно более высокого уровня производительности методов доступа к данным в этих системах по сравнению с системой-прототипом.

Но этим не исчерпываются возможные подходы. В [22] сообщается о трансляторе программ на языке dBase в язык Си, что позволяет далее произвести компиляцию и редактирование полученного программного текста и иметь в результате быстро исполняемые загрузочные программные модули.

Повышению производительности систем баз данных способствует использование расширения основной памяти, которым располагают ПЭВМ IBM PC AT. Такие возможности предусматриваются, в частности, в системах Paradox и Clipper.

Разработчикам системы R:base for DOS удалось значительно повысить ее производительность по сравнению с предыдущей версией - R:base System V - за счет того, что эта новая версия написана на языке Си, в то время как R:base System V - на Фортране.

Заметим, наконец, что существенный вклад в повышение производительности систем баз данных в сетевой обстановке призваны обеспечить новые подходы, основанные на изменении распределения функций между рабочими станциями и машиной управления файлами. Об этом более подробно шла речь в разделе 10.

### Замечания о терминологии

Формирование и употребление терминологии в области баз данных для ПЭВМ требуют самостоятельного изучения. И здесь мы лишь кратко коснемся этого вопроса.

Прежде всего обращает на себя внимание тот факт, что в этой области практически вышли из употребления некоторые пласты традиционной терминологии. Для примера можно сослаться на терминологию среды хранения баз данных, на терминологию языков описания данных.

По нашему мнению, это явление имеет естественные причины. Механизмы управления средой хранения, по существу, скрыты от пользователя новых СУБД, не подверга-





ются его явным воздействиям и тем самым не входят в сферу его деятельности. В связи с этим и отпадает потребность в соответствующем понятийном аппарате и выражающей его терминологии. Что касается языков описания данных, то в большинстве систем они уже перестают существовать как явные языковые средства. Их функции реализуются, главным образом, с помощью различных интерактивных дружественных интерфейсов - меню, экранных форм описания данных и т.д.

Вместе с тем новая технология породила и ряд новых терминов. Их появление можно объяснить специфическими технологическими особенностями функционирования рассматриваемой группы СУБД. Главным звеном здесь являются пользовательские интерфейсы, объекты, с которыми они оперируют, и их свойства - экранные формы ввода/вывода, меню, средства помощи пользователю в оперативном режиме.

Значительно активнее стала использоваться терминология параллельных процессов в базах данных, что связано с широким распространением локальных сетей ПЭВМ и систем баз данных, функционирующих в сетевом режиме. Другие отклонения терминологии "персональных" баз данных от традиционной возникают, по нашему мнению, вследствие небрежности, являются неоправданными либо вовсе ошибочными. Приведем несколько примеров.

Весьма часто, особенно в периодических изданиях, вместо "СУБД" используется термин "база данных". Система Paradox в документации [23] квалифицируется не как СУБД, а как "программа управления базами данных" (Data Base Management Program). Там же общепотребительный термин "представление" (View) используется не для обозначения виртуального объекта данных, а для совершенно иных целей - здесь это образ на экране таблицы из базы данных.

В документации по системе R:base [10] для обозначения понятия "ограничение целостности" используется термин "правило" (Rule). Название теоретико-множественной операции UNION служит, в отличие от его традиционного использования в реляционных базах данных, для обозначения естественного соединения отношений по общим атрибутам (явно не указываемым).

В документации системы db\_Vista [24] ключ базы данных получил название "адрес базы данных" (Database Address), язык определения данных - "язык определения базы данных" (Database Definition Language), а индекс - "файл ключей" (Key File). Весьма своеобразно трактуется здесь и понятие транзакции.

Нам представляется, что терминологию баз данных для ПЭВМ следует изучать, систематизировать, она должна найти свое отражение в терминологических, переводных и толковых словарях по технологии баз данных.

### Заключение

Мы можем в заключение констатировать, что за десятилетие своего существования технология баз данных на персональных ЭВМ сформировалась как весьма развитая ветвь общей технологии баз данных. Она обладает в настоящее время развитым спектром инструментальных средств для широкого диапазона областей применения.

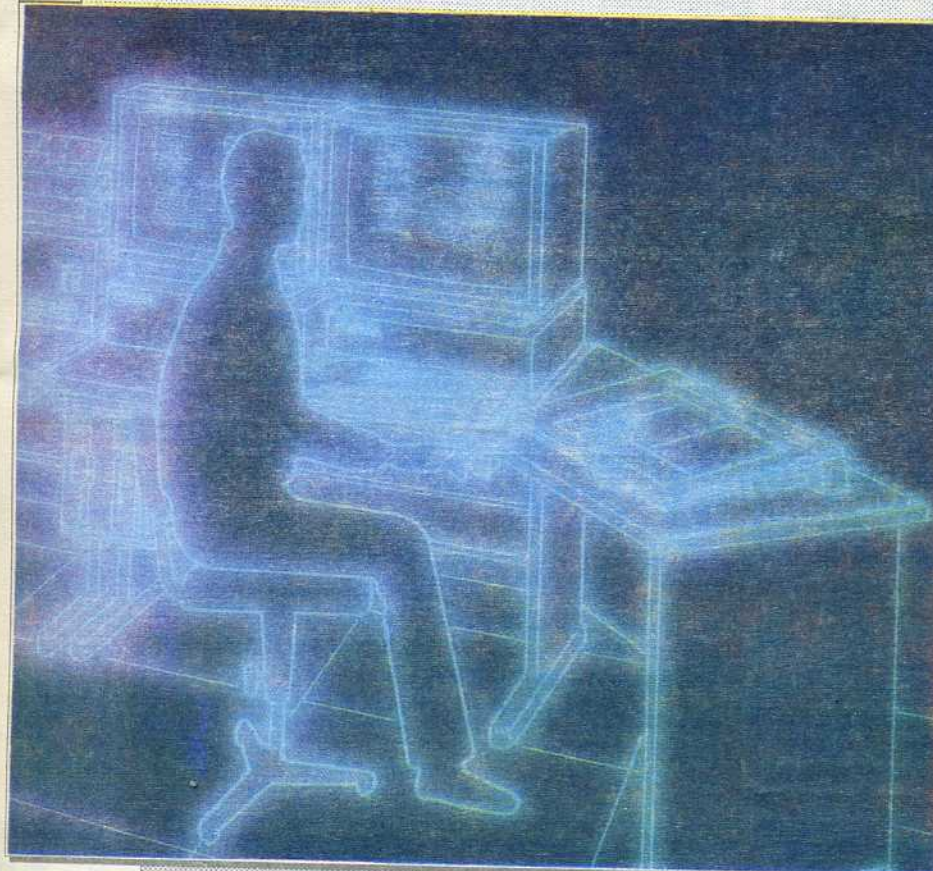
Наиболее мощные системы рассматриваемого класса в функциональном аспекте приблизились по своим возможностям к крупным СУБД для больших вычислительных систем. Они обеспечивают в высокой степени дружественную операционную среду для пользователей и хорошо приспособлены для организации распределенной обработки данных.

"Персональные" системы управления базами данных с успехом вытесняют "большие" СУБД во многих практических приложениях.



1. PC Tech Journal, May 1988, p.115, 137.
2. Дейт К. Руководство по реляционной СУБД DB2. - М.: Финансы и статистика, 1988.
3. PC Tech Journal, May 1988, p.39.
4. PC World, June 1987, p.16.
5. Byte, March 1988, p.88.
6. Chen P.P. The entity-relationship model: Toward a unified view of data. ACM Trans. Database Syst., 1, 1976, p.9-36.
7. Byte, August 1987, p.245.
8. PC World, July 1989, p.8.
9. Paradox Personal Programmer. Relational Database Release 2.0. Ansa Software, 1987.
10. S.Condliffe: R:base for DOS. A Business User's Guide. John Wiley & Sons, Inc., 1988.
11. PC Magazine, October 11, 1988, p.154.
12. Chapel H. and Nesary M.: Share the Wealth. PC World, June 1988, p.134-143.
13. Clipper. Nantucket Inc., Summer 1987.
14. Ульман Дж. Основы систем баз данных. - М.: Финансы и статистика, 1983.
15. PC World, July 1989, p.63.
16. PC World, February 1989, p.11.
17. Walden J. File Formats for Popular PC Software. A Programmer's Reference. John Wiley & Sons Inc., 1986.
18. Берещанский Д.Г. Практическое программирование на dBase. - М.: Финансы и статистика, 1989. - 192 с.
19. Крамм Р. Системы управления базами данных dBase II и dBase III для персональных компьютеров. - М.: Финансы и статистика, 1988. - 283 с.
20. Beam G. Advanced Clipper: dBase Compiler Applications. TAB Books Inc., Blue Ridge Summit, 1988, 291p.
21. FoxBase+. Relational Database Management System. User Manual. Fox Software Inc., January 1987.
22. PC Magazine, July 1988, p.30.
23. Paradox. User's Guide. Release 2.0. ANSA Software, 1987.
24. db\_Vista User's Guide. Raima Corp., 1986.





## Компьютер на работе

- ☐ AutoCAD в СССР
- ☐ AutoCAD даром?! ...
- ☐ Плоттеры
- ☐ Оценка качества перьевых плоттеров
- ☐ Люминография
- ☐ Заполнение видеобуфера EGA
- ☐ Компьютерный склад информации
- ☐ С ПК по телефону
- ☐ Дискотека
- ☐ Словари, словари ...
- ☐ Norton Utilities: NI и BE
- ☐ Shez
- ☐ Русифицируем Word
- ☐ Пишем AUTOEXEC.BAT

## Компьютер на работе

© Петр Кадович

# AutoCAD в СССР

С генеральным директором советской фирмы PARALLEL Спартак Чеботаревым беседовал наш корреспондент Петр Кадович.

**Петр Кадович:** Расскажите, пожалуйста, коротко о своей фирме.

**Спартак Чеботарев:** PARALLEL - небольшая программно-техническая фирма, основанная в январе 1989 г. как совместное предприятие AUTODESK Ltd. из Лондона и научно-производственного кооператива INTROGRAF из Москвы. Наша задача - производство и маркетинг инженерного программного обеспечения высокого класса для микро-ЭВМ как в Советском Союзе, так и на Западе. Оттуда и по-

шло название PARALLEL - параллельный обмен продуктами на одинаковом технологическом уровне между СССР и Западом.

**П.К.** Вы занимаетесь только разработкой программных продуктов?

**С.Ч.** Исключительно. Мы - новички на рынке, а торговля компьютерным оборудованием - дело рискованное. Кроме того, PARALLEL - небольшая фирма (всего 18 человек) и хочет таковой оставаться. Намеченного нами верхнего предела занятости - 24 человека - мы достигнем к концу текущего года.

**П.К.** Работает ли ваше программное обеспечение только в среде AutoCAD?

**С.Ч.** Пока да, но у нас есть свои соображения по поводу совершенно оригинальной системы автоматизации проектирования, основанной на искусственном интеллекте.

**П.К.** В то же время вы распространяете русифицированную версию программы AutoCAD в Советском Союзе.

**С.Ч.** Да, но этим же занимается также советско-американское совместное предприятие ДИАЛОГ и несколько представителей западных фирм в Советском Союзе, например QUEST, FEXIMA и др.

**П.К.** И все они поставляют русско-язычную версию этой программы?

**С.Ч.** По выбору: русскую или оригинальную, англо-язычную, но зато более дорогостоящую.

**П.К.** Сколько законных копий AutoCAD применяется в СССР?







С.Ч. По данным на начало текущего года - около 1000.

П.К. Что конкретно из программного обеспечения предлагает PARALLEL?

С.Ч. Компилятор языка AutoLISP, генератор программ в системе AutoLISP - GLISP, а также параметрическую базу данных - ABASE.

П.К. И эти продукты распространяются под "флагом" фирмы AutoDesk?

С.Ч. Компилятор AutoLISP поставляется как интегральная часть пакета AutoCAD, а значит, как продукт фирмы AutoDesk. Кроме того, архитектурное приложение AutoCAD AEC сейчас поставляется только в компилированном виде. GLISP и ABASE будут распространяться совместно с AUTODESK, но как продукты фирмы PARALLEL.

П.К. Помогает ли Вам репутация фирмы AUTODESK?

С.Ч. Конечно. Но в то же время сотрудничество с одним из мировых лидеров в области программного обеспечения дается нелегко. AUTODESK предъявляет огромные, не встречающиеся у нас требования. Путь от хорошей программы до готового коммерческого продукта намного длиннее чем многим может показаться.

П.К. А чего Вы требуете от своих программистов?

С.Ч. Чтобы работать соответственно стандартам AUTODESK - нужно нанимать на работу только лучших программистов. От них мы требуем полной "преданности" фирме. Не может быть речи о "подработке" в других фирмах. Но в то же время работа наших программистов очень хорошо оплачивается.

П.К. Найти таких людей, наверное, не так просто?

С.Ч. Да, это дело не из простых. Программистов, занятых в нашей фирме, можно разделить на две возрастные группы: 32 - 36-летних и более молодых, 18 - 24-летних. Отдельные работы выполняются небольшими коллективами, в которых более опытные специалисты могут обучать своих младших коллег.

П.К. Кто работает с вашими программами?

С.Ч. В частности, автозавод КАМАЗ, фирма, связанная с реализацией проекта челночного космического корабля "Буран", несколько авиазаводов и большое число проектно-конструкторских бюро, в которых AutoCAD стал уже стандартом. На нашем программном обеспечении - я имею в виду AutoCAD и базу данных ABASE - проводится работа по восстановлению потерпевших от землетрясения районов Армении.

П.К. Конечно, не все работают с оригинальными программами?

С.Ч. К сожалению. Но все чаще осознается потребность в, так сказать, легализации своего труда.

П.К. Какие у программы AutoCAD конкуренты в СССР?

С.Ч. Кое-где применяется Drafix Ultra, VersaCAD, CADkey и другие системы, но в подавляющем большинстве случаев это пиратские копии. Только AutoCAD и Drafix Ultra распространяются в СССР официальным путем. Но AutoCAD - единственная полностью русифицированная САПР, которую можно легально приобрести в СССР.

П.К. AutoCAD - безусловно, самая распространенная САПР, причем не только в СССР, но и во всем мире. А как обстоят дела с оригинальными системами, разработанными советскими программистами?

С.Ч. В некоторых областях проектно-конструкторских работ микрокомпьютеры не в состоянии обеспечить требуемую скорость и эффективность решения задач. В таких случаях применяются большие ЭВМ и специализированное, как правило, отечественное программное обеспечение.

П.К. Следовало бы также упомянуть описанную на страницах "Мира ПК" САПР "Черту" ("Вишенка"). Наша

редакция хотела протестировать ее, равно как и упомянутую Вами программу Drafix Ultra. К сожалению, это было невозможно, несмотря на продолжавшиеся несколько месяцев переговоры, а в случае программы Drafix Ultra даже заключенное соглашение с официальным поставщиком этой программы в СССР. Ни с Черту, ни с Drafix Ultra нашим специалистам так и не удалось поработать.

Разрешите поблагодарить Вас за интересную беседу. В заключение мне хотелось бы выразить надежду, что с покупкой САПР AutoCAD у ваших потенциальных клиентов таких проблем не будет.

Перевод Халины Мадейчик

## ЦЕНТРЫ ОБУЧЕНИЯ САПР AutoCAD в СССР

### МОСКВА

Центр обучения по вычислительной технике

Ломоносовский пр-т, 16

телефон 133 24 35

факс 133 24 35

просить Татьяну Кравчук

Учебный центр Мосгорремстроя

ул. Яблочкова, 5

телефон 210 33 04

просить Валентина Горкавенко

ЦНИИ Промзданий при Госстрое СССР

Дмитриевское шоссе, 46

телефон 482 94 39

просить Ивана Кузьминского

### ЛЕНИНГРАД

"Механобр"

ул. 21-я линия, 8а

телекс 121419 meobr su

телефон 213 92 74

просить Евгения Буйчева

### РИГА

Рижский политехнический институт

1 Kalku Str

телекс 161172 tema su

телефон 21 38 89

просить Олафа Бринкманиса

Проектный Институт "Pilssetprojekts"

38 Kr Valdemara Str

телефон 27 30 52

просить Леонида Айзенберга



\компьютер на работе\

© Збигнев Блевоньский

# AutoCAD даром ?!

## ...но не для всех

Можно ли стать зарегистрированным владельцем программы AutoCAD бесплатно? Можно, но надо выполнить минимум требований, которые ставит фирма AUTODESK Ltd., т. е. стать "разработчиком программного обеспечения" (получить Application Developer Status).

### Что такое прикладное программное обеспечение?

Напомним, что AutoCAD является универсальным инструментом, а скорее средой для создания чертежей и проектов, которая фактически стала международным стандартом. Прикладное программное обеспечение - это решение определенных проблем, дополняющее и расширяющее возможности пакета AutoCAD. Обычно это библиотеки символов, процедур, системы меню, файлы "сценариев" (script), иногда целые программы на языке AutoLISP (или на другом языке). Прикладное ПО позволяет выполнить конкретную, специализированную работу (расчеты, чертежи) или максимально облегчает ее.

### Кто такой "разработчик ПО" в понимании AUTODESK?

Под этим термином кроется фирма, занимающаяся разработкой программного обеспечения, работающего в среде AutoCAD, гарантирующая постоянное развитие ПО и всестороннюю помощь пользователям, начиная с элементарного курса использования данной программы и кончая поддержкой в области применения в нетипичных условиях. К разработчикам ПО также можно отнести авторов адаптаций уже существующих программ, например американских, для нужд европейского рынка или переводов англоязычных программ на русский язык. Разработчик ПО пользуется помощью и поддержкой AUTODESK. Он получает, во-первых, новейшую информацию о развитии системы AutoCAD, во-вторых, бесплатное программное обеспечение, которое может помочь ему в работе (например, Kalvinator - программа, шифрующая программы на языке AutoLISP, компилятор AutoLISP и т.д.), в-третьих, бесплатный пакет AutoCAD. Кроме того, разработчик имеет право (также бесплатно) поместить описание своих продуктов в каталоге прикладного программного обеспечения фирмы AUTODESK.

### Не является ли бесплатный AutoCAD искушением для...?

Безусловно, да. Много раз возникала такая ситуация, что некоторая фирма хотела попасть в список разработчиков, но на вопрос об умении пользоваться пакетом AutoCAD ответ звучал примерно так: как только мы получим программу (разумеется, бесплатно), то научимся с ней работать и быстро начнем писать ПО. Такие фирмы не рассматриваются как серьезные партнеры. Прежде чем фирма станет партнером AUTODESK, проверяется ее оснащение, возможности, опыт и методика работы, качество документации, методы тестирования программных продуктов в ходе разработки, созданные с их помощью проекты и т.д. Большое внимание уделяется обслуживанию поль-

зователя - программа не может попасть в каталог AUTODESK без адреса конкретного человека, который будет помогать пользователю. В Великобритании в связи с большим количеством кандидатов проверку проводят представители фирмы. В других странах предварительный отбор проводится на уровне главного торгового представителя или зарегистрированного центра обучения. Окончательное решение принимается в штаб-квартире AUTODESK.

### Как проверяются предлагаемые программы?

Если даже не проводится полный тест, то контролируется работа программы, изучается документация, высказывается мнение экспертов. Программа оценивается с точки зрения ее соответствия нормам, а также принимается во внимание простота и удобство работы с ней. Если все требования выполнены, то автор (фирма) принимает условно в качестве разработчика ПО. Реализация "условия" проверяется в дальнейшем фирмой AUTODESK и самой жизнью, т. е. клиентами. Проверка со стороны AUTODESK проводится перед каждым изданием каталога (обычно два раза в год). Если все идет правильно, предоставляется так называемый полный статус разработчика. Иногда помещенная в каталог программа не пользовалась успехом у клиентов и сами авторы отказывались от внесения ее в очередное издание. Однако не было еще случая, чтобы программа в результате жалоб пользователей была изъята из каталога.

### Сколько версий каталога издается?

Каталог, который издается в Лондоне (над ним работают только два человека), включает ПО фирм из Англии, Шотландии, Ирландии, Бельгии, Голландии и стран Восточной Европы. Он содержит около 250 программ. "Условных" партнеров в настоящее время примерно 170, "полных" - примерно 125. Каталоги издаются также в США, Швеции, Австралии, Японии. Количество прикладных программ во всем мире превысило уже 500. Издание каталога - одно из проявлений философии фирмы: чем больше на рынке хороших программ для среды AutoCAD, тем больше удовлетворенных клиентов имеет фирма AUTODESK - автор идеи открытого пакета САПР.

Статья подготовлена на основе беседы с Э. Ньюзом, руководителем отделения прикладных программ фирмы AUTODESK Ltd.

Перевод Анджея Поплавского

Если вы профессионально занимаетесь вопросами САПР, рекомендуем вам для чтения:

США

"Micro CAD News" - ежемесячник, посвященный различным системам автоматизации проектирования.

"CADANCE" - ежемесячник, предназначенный для пользователей системы AutoCAD.

Канада

"Cadalyst" - ежемесячник, рассчитанный на опытных пользователей системы AutoCAD.

Великобритания

"CADUser" - издающийся каждые два месяца журнал, предназначенный для пользователей AutoCAD.

ФРГ

"CADCAE" - ежемесячник, освещающий новинки в области САПР.

"CADCAE Report" - популярное ежемесячное издание для пользователей различных САПР.

Польша

"CAD Forum" - журнал, выходящий раз в два месяца, предназначен для пользователей САПР.





© Збигнев Блевоньский

Собственно говоря, не плоттеры, а графопостроители, но среди специалистов распространено все-таки это название. Плоттер – это внешнее (периферийное) устройство для вывода чертежей “из компьютера”. Они были изобретены и начали использоваться раньше, чем компьютеры. “Пращуром” плоттера можно назвать графическое регистрирующее устройство для записи изменений электрических параметров, типа самописцев, применяемых при ЭКГ. Это устройство гениально по своей простоте. Оно состоит из механизма равномерного передвижения бумаги (ось X) и механизма преобразования измеряемых электрических величин на движение пера. В результате адаптации этой концепции для целей технического черчения в 1959 г. были созданы барабанные графопостроители. Затем появились плоттеры планшетного типа – большие, точные, но очень дорогие. Их, как и системы САПР, могли покупать только крупные фирмы. Это была эпоха ICL (ODP 1304) и IBM 360/370 (РЯД-32). Развитие мини- и микрокомпьютеров, а также рост интереса к системам САПР, которые стали доступны для небольших фирм, придали значительное ускорение в конструировании и производстве плоттеров. Долгие годы плоттеры ассоциировались с фирмой CALCOMP. Первая модель барабанного плоттера этой фирмы, Calcomp 565, появилась в 1959 г. В 1965 г. фирма HOUSTON INSTRUMENT продемонстрировала небольшой планшетный графопостроитель с шаговым двигателем. HEWLETT-PACKARD в 1980 г. показал рулонную модель HP7580. Потом начали появляться все новые и новые модели этих и многих других фирм.

### Различия в конструкциях

Типичным свойством, отличающим графопостроители высокого качества от низкокачественных, является привод – шаговый или серво. В шаговом двигателе движение осуществляется как бы прыжками, между определенными позициями. Такой двигатель дешев, но производит много шума, не дает высокой точности и работает медленно. Сервопривод же является сложной системой “двигатель – электронное управление”, обеспечивающей плавное движение. Сервопривод – дорогое удовольствие, однако работает тихо, быстро и точно. В настоящее время наблюдается не только снижение цен сервоприводов, но и рост качества (точности) шаговых двигателей.

Вторым свойством, по которому различаются конструкции графопостроителей, является метод передвижения по отношению друг к другу бумаги и перьев. Традиционным решением является планшетный плоттер. В нем бумага прикрепляется к рабочей плоскости липкой лентой, магнитами, иногда электростатически или пневматически. Над бумагой передвигается (с помощью одного двигателя) штанга с направляющими, по которым в свою очередь перемещается (с помощью второго двигателя) пишущий узел. Достоинство планшетных плоттеров – простота крепления бумаги и возможность использования всей ее поверхности. Недостаток лишь один – они занимают много места.

Другим решением являются рулонные графопостроители, иногда называемые “выжималками”. Бумага крепится в них фрикционным методом, между валиком с шерохо-

ватой поверхностью и резиновыми прижимными роликами. При черчении перо передвигается поперек бумаги (один двигатель, ось Y), а бумага перемещается поперек плоттера (второй двигатель, ось X). Устройства такого типа занимают относительно мало места и в настоящее время преобладают в группе крупноформатных графопостроителей. “Предком” таких плоттеров были барабанные плоттеры, в которых специальная бумага (чаще всего перфорированная по краям, как в принтерах) в процессе черчения поступала из рулона. Затем появились так называемые конвейерные плоттеры, в которых лист бумаги крепился к передвигающейся во время работы специальной широкой бесконечной пластиковой ленте, как в конвейере.

### Формат чертежа

Первым параметром, определяющим пригодность плоттера для определенных целей, является максимальный формат чертежа, который можно на нем вычертить. Формат соответствует нормам ANSI (американским) или ISO/DIN (европейским). Величины листов бумаги в этих нормах отличаются незначительно (см. табл.). Большинство фирм производит графопостроители форматов A3/A4, относительно простые в производстве и дешевые. Плоттеры больших форматов A0/A1 встречаются гораздо реже, так как для них нужна более точная “механика” и они значительно дороже. Малые плоттеры используются чаще всего для так называемой деловой графики (business graphics), например в программе Lotus 1-2-3, большие – для выполнения чертежей с помощью систем САПР. Формат чертежа определяется на уровне программы (например, AutoCAD), хотя некоторые плоттеры имеют специальные датчики для автоматического определения формата чертежа в зависимости от величины листа бумаги. Графопостроители больших форматов позволяют выполнять чертежи на всех форматах меньше максимального. В рулонных графопостроителях бумага может подаваться вручную, в отдельных листах или автоматически из рулона.

### Скорость движения пера

Максимальная скорость движения пера говорит о том, насколько быстро может передвигаться пишущий узел: в некоторых плоттерах возможно независимое управление скоростью черчения и скоростью передвижения узла с поднятым пером. Для каждого плоттера должна быть известна осевая скорость (скорость передвижения пера по оси), а также суммарная скорость (скорость передвижения по отношению к обеим осям). Скорость измеряется в дюймах или миллиметрах в секунду и обычно составляет от 30 до 800 мм/с. Следует помнить, что максимальная скорость, которую можно узнать из рекламного материала фирмы, относится к физическим возможностям плоттера и имеет мало общего с фактической скоростью черчения, которая зависит от типа пера (фломастеры, шариковые ручки) и бумаги (калька, обычная бумага, прозрачная пленка) (см. табл.).

### Ускорение

Ускорение – это величина, определяющая, насколько быстро пишущий узел достигает максимальной скорости как при поднятом пере, так и при черчении. Ускорение представляется как кратность земного ускорения (g), которое равняется 9,81 см/с, и составляет обычно 0,5 – 4 g. Ускорения плоттеров с шаговыми двигателями редко превышают 1 g. Эффективное ускорение при черчении (с опущенным пером) ограничено возможностями пера.

### Вертикальное движение пера

Обычно перо опускается и прижимается к бумаге с помощью электромагнита, а поднимается возвратной пружиной. В дорогих моделях сила опускания пера контролируется с помощью пневматических или электромагнитных амортизаторов, что продлевает работу пера и уменьшает шум.

### Эффективная скорость черчения

Практическая скорость черчения зависит от всех вышеуказанных технических параметров, формата чертежа и от скорости смены перьев (если в данной модели есть та-



кая возможность), а также от производительности ЭВМ, обрабатывающей чертеж.

### Разрешение

Разрешение – это минимальное расстояние, на которое может быть перемещено перо. Чем разрешение выше, т. е. чем меньше такое расстояние, тем ровнее вычерчиваются косые и кривые линии. Следует отличать аппаратное разрешение (возможности самого графопостроителя) от программного разрешения (способности программы управлять плоттером). В рекламах речь идет об аппаратном разрешении, составляющем обычно 0,001 дюйма (около 0,025 мм), что соответствует разрешению принтера 1000 точек/дюйм (при разрешении лазерных принтеров 300 точек/дюйм).

Программное разрешение обычно гораздо выше и не вызывает сложностей. Его влияние может быть заметно при особо больших (длинных) чертежах, например свыше 2 м. Так, драйвер ADI для программы AutoCAD допускает максимум 65535 шагов по каждой оси.

### Повторяемость

Повторяемость – это максимальная погрешность, которая может появиться при перемещении пера из одной точки в другую и обратно. Она является результатом внутренних люфтов в механизмах плоттера. Повторяемость представляется в линейных единицах, ее величина не должна превышать 0,004 дюйма (0,1 мм) – естественно, чем меньше, тем лучше.

На повторяемость влияет также качество и точность исполнения пера (находится ли пишущий штифт точно по оси пера). При плоттерах со сменными перьями повторяемость снижается при смене пера, а дополнительно каждое перо может иметь иначе расположенный штифт. В этом отношении хуже всего представляются плоттеры с шариковыми ручками.

### Точность

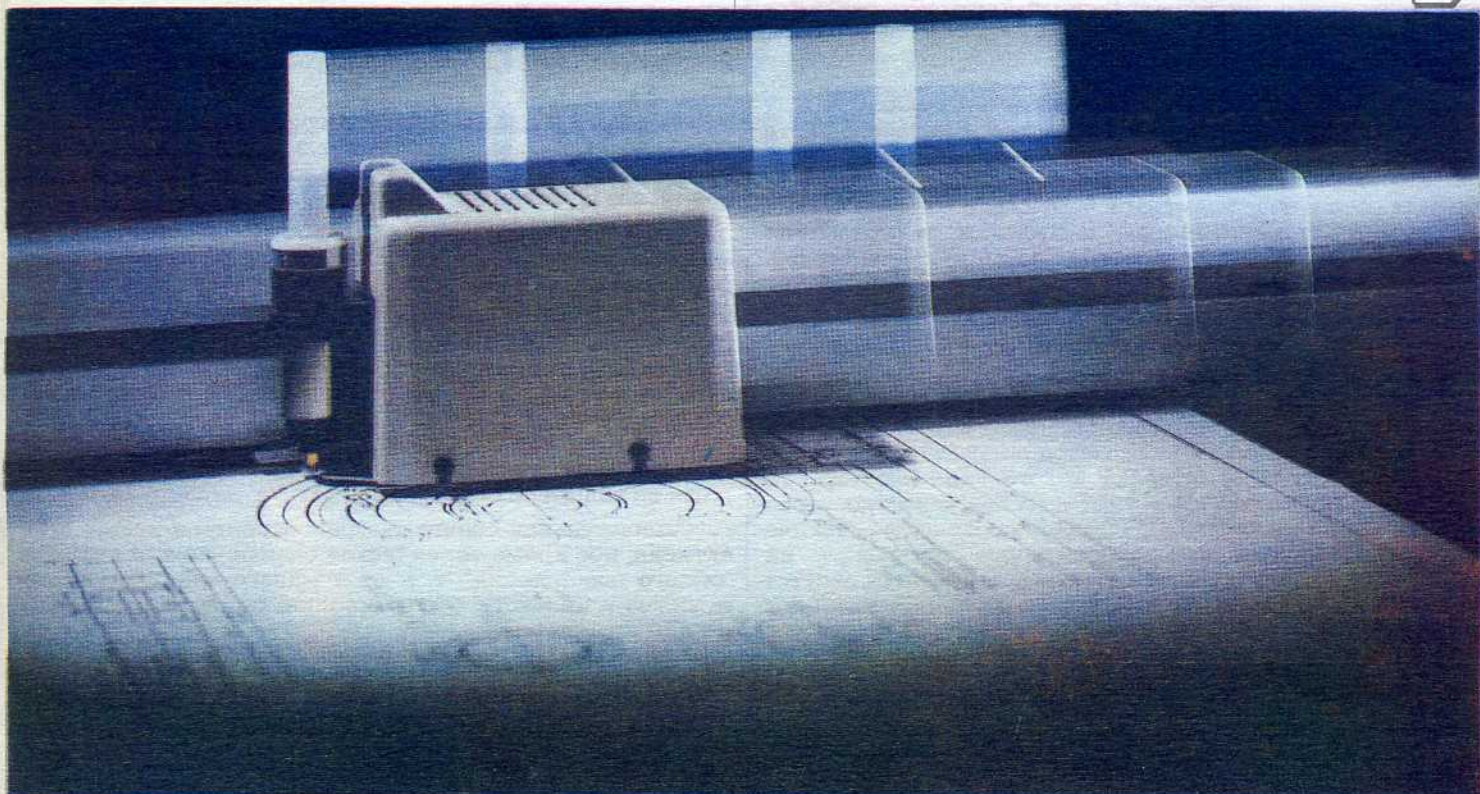
Это важнейший параметр качества плоттера. Точность – максимальная погрешность при вычерчивании конечных точек отрезка. Она представляется в виде абсолютной погрешности (в линейных единицах) и относительной погрешности (как процент длины вычерчиваемого отрезка). Показатели точности ниже 0,01 дюйма (0,254 мм) и 0,1% можно считать весьма хорошими.

Стоит отметить, что плоттер с нормальной повторяемостью может быть неточным систематически повторяя одну и ту же ошибку.

### Язык – управление плоттером

В каждый плоттер встроен графический язык, позволяющий компьютерной программе легко контролировать работу устройства. Этот язык переводит поступающие команды в электрические импульсы, непосредственно управляющие работой механизмов графопостроителя. Стандартом являются два языка лидирующих фирм: HPGL (Hewlett-Packard Graphic Language, продукт фирмы HEWLETT-PACKARD) и DM/PL (Digital Microprocessor/Plotting Language, продукт фирмы HOUSTON INSTRUMENT). Не стал стандартом язык (протокол) CalComp 906, разработанный не менее известным производителем плоттеров. Дешевые модели плоттеров эмулируют DM/PL как более простой в применении, однако в пакетах типа САПР предпочтение отдается языку HPGL, как дающему более широкие возможности. В связи с этим графопостроители фирмы HOUSTON из новейшей серии DMP-60 эмулируют также стандарт соперника HPGL.

Большинство плоттеров располагает встроенными шрифтами, штриховкой, различными видами точек и линий, процедурами выравнивания дуг и окружностей, а также оптимизации всего процесса черчения. Однако лишь немногочисленные программы применяют эти возможности. Так, с целью обслуживания как можно большего числа моделей плоттеров AutoCAD почти полностью "берет на себя" контроль над графопостроителем, почти не используя его встроенные возможности. В ходе подготовки чертежа AutoCAD переводит все его элементы на последовательности отрезков прямых разной длины. Программа также проводит оптимизацию черчения: например, все элементы данного цвета чертятся сразу, благодаря чему перо с этим цветом используется только один раз. Связь с плоттером идет в обход DOS, что, во-первых, ускоряет работу, а во-вторых освобождает пользователя от установления параметров передачи данных командой MODE. Разработанный относительно недавно драйвер ADI (Autodesk Device Interface) для плоттеров вначале игнорировался производителями оборудования в связи с мощной позицией HPGL и DM/PL. Со временем оказалось, что ADI имеет ряд досто-







## Дополнительные возможности

Компьютер высылает данные на плоттер быстрее, чем они могут быть перенесены на бумагу. Стандартно графопостроители снабжаются памятью ОЗУ до 20 Кбайт, которая используется как буфер данных, поступающих из ЭВМ. Но это лишь незначительно сокращает время блокирования машины. Для ускорения этого процесса в некоторых плоттерах предусмотрена возможность установки дополнительной памяти, обычно в пределах 1 - 4 Мбайта, что позволяет ввести в ОЗУ данные даже о большом чертеже и освободить компьютер.

Совершенно необходимо, чтобы не используемое в данный момент перо автоматически закрывалось во избежание высыхания: это также облегчает работу, устраняя "расписывание" перьев.

Графопостроители могут быть использованы также как устройства ввода графической информации типа диджитайзера. Однако это не очень удобно и никогда не заменит настоящего графического планшета.

## Что купить?

Разумеется, то, что вы можете себе позволить. Цена плоттера в основном зависит от его формата, точности (как исполнения устройства, так и черчения) и безотказности. Если в месяц приходится выполнять более 30 чертежей большого формата, то следует приобрести наиболее "прочный" (так называемый heavy-duty) и, к сожалению, дорогой графопостроитель. При 20 чертежах в месяц можно купить модель подешевле, при еще меньшем объеме работы - одно из дешевых устройств. Следует, однако, помнить, что нередко дешевизна может обойтись дороже. Важно также обратить внимание на простоту обслуживания, доступность эксплуатационных материалов и техническое обслуживание.

\* \* \*

Говоря "плоттер", мы до сих пор имели в виду устройства, вычерчивающие линии пером определенного типа. Это наиболее распространенный тип плоттеров. Перьевые плоттеры отличаются низкими ценами (300 - 1200 дол.), высоким качеством, широким выбором пишущих устройств и материала, на котором выполняется чертеж. Одним словом, сплошные достоинства! Но все же это не единственный тип графопостроителей. Итак, что же еще стоит знать потенциальному покупателю?

## Электростатические плоттеры

Этот тип плоттеров дает самое высокое качество чертежа. Миниатюрные острия головки электризуют точки бумаги, которые затем притягивают краситель, который оставляет след в виде элементов чертежа. Возможна работа в цвете. Самые большие недостатки: высокая цена (12000 - 120000 дол.) и необходимость применения специальной бумаги. Крупнейшее достоинство - поистине огромная скорость черчения.

## Электрофотография<sup>1</sup>

Это технология сегодняшнего и завтрашнего дня. Чертеж создается также с помощью красителя, однако носитель активируется с помощью сильного источника света, обычно лазера (аналогичный метод используется в современных лазерных принтерах). Такого типа плоттеры гарантируют большую скорость и высокое качество черчения (до 1200 точек/дюйм). По сравнению с электростатическими плоттерами они относительно дешевы - ниже 20000 дол. Их основным недостатком является отсутствие цвета, однако в последнее время появились уже цветные модели.

<sup>1</sup> Обычно употребляется термин "электрография"...  
Примеч. ред

## Термические плоттеры

Постепенно появляющиеся на рынке графопостроители такого типа имеют разрешение 160 - 300 точек/дюйм. Чертеж образуется в результате нанесения на бумагу расплавленного воскового красителя. Цены колеблются от 300 до 9000 дол. Крупнейший недостаток - высокие цены эксплуатационных материалов.

## Струйные плоттеры

Ненадежность первых моделей плоттеров такого типа практически перечеркнула надежды, которые возлагались на эту технологию. В настоящее время наблюдается появление на рынке моделей, которые продаются как гибриды "принтер/плоттер" с возможностью работы на форматах А3 или даже А2. Новейшие модели довольно надежны и характеризуются разрешением 300 - 400 точек/дюйм, т. е. могут конкурировать с лазерными принтерами. Выпускаются два основных типа: с непрерывной или пульсирующей струей красителя. Цены - от 700 до 7000 дол., хотя попадаются модели и по 75000 дол. Достоинство - дешевые эксплуатационные материалы, крупнейший недостаток - отсутствие цвета.

## Матричные (мозаичные) плоттеры

По принципу действия они аналогичны матричным принтерам, но по размерам достигают формата А2. Обычно снабжаются 24-игольными головками, имеют обширный буфер, могут работать в цвете. Их относительно высокая цена (400 - 7000 дол.) и в то же время низкое качество чертежей объясняют, почему они практически исчезают с рынка.

## Как оценивать плоттеры?

Вернемся снова к наиболее распространенным, классическим перьевым плоттерам. При оценке наибольшее значение имеют два параметра, которые трудно объединить - быстродействие и высокое качество чертежа. Оба находятся в прямой зависимости от технических свойств графопостроителя (о чем мы писали выше) и от комбинации "перо - бумага". Производители плоттеров и перьев дают следующие рекомендации в отношении выбора скорости черчения, перьев и бумаги:

- черновики можно выполнять на бумаге или кальке с максимальной доступной скоростью и с применением шариковых ручек. Качество чертежа обычно оставляет желать лучшего (особенно цвет), но работа идет очень быстро;
- обычные чертежи можно выполнять на бумаге или кальке с немного уменьшенной скоростью черчения (300-400 мм/с), в качестве перьев используя рейсфедеры;
- чертежи высшего качества следует выполнять со значительно уменьшенной скоростью (150-300 мм/с), лучше всего на пластиковой пленке рейсфедерами и специальной тушью для пленки;
- при чертежах высокой точности следует помнить о влиянии температуры и влажности на изменение размеров листа, на котором выполняется чертеж, - это относится к бумаге, кальке и ацетатной пленке, но не касается полиэфирной пленки.

Контроль скорости и точности плоттера, а также качества чертежа проводится обычно на двух многоцветных сложных чертежах. Первый из них, так называемый рабочий, выполняется шариковыми ручками с наибольшей доступной скоростью, а второй, окончательный, - рейсфедерами на кальке со скоростью 200 мм/с.

Окончательная оценка графопостроителя всегда субъективна, ибо кроме технической оценки зависит также от "общего впечатления", т. е. от внешнего вида устройства, удобства и простоты обслуживания, качества исполнения и т.д.

Перевод Анджея Поплавского



## ФОРМАТЫ ЧЕРТЕЖНЫХ ЛИСТОВ

## Американский стандарт (ANSI)

| Символ | Размер   |                             |
|--------|----------|-----------------------------|
|        | дюймы    | миллиметры (приблизительно) |
| A      | 8.5 x 11 | 216 x 279                   |
| B      | 11 x 17  | 279 x 432                   |
| C      | 17 x 22  | 432 x 558                   |
| D      | 22 x 34  | 558 x 864                   |
| E      | 34 x 44  | 864 x 1116                  |

## Европейский стандарт (ISO/DIN)

| Символ | Размер                 |            |
|--------|------------------------|------------|
|        | дюймы (приблизительно) | миллиметры |
| A4     | 8.3 x 11.7             | 210 x 297  |
| A3     | 11.7 x 16.5            | 297 x 420  |
| A2     | 16.5 x 23.4            | 420 x 594  |
| A1     | 23.4 x 33.1            | 594 x 840  |
| A0     | 33.1 x 48.8            | 840 x 1188 |

| Тип пера                    | Рекомендуемая скорость черчения |      | Замечания             |
|-----------------------------|---------------------------------|------|-----------------------|
|                             | дюйм/с                          | мм/с |                       |
| Фломастер:<br>водяной       | 18                              | 450  | закрашивание контуров |
| масляный                    | 4                               | 100  | черчение на пленке    |
| шариковая ручка:<br>водяная | 18                              | 450  | черновики             |
| масляная                    | 18                              | 450  |                       |

| Тип пера                  | Рекомендуемая скорость черчения |      | Замечания                 |
|---------------------------|---------------------------------|------|---------------------------|
|                           | дюйм/с                          | мм/с |                           |
| керамическое перо         | 4                               | 100  | тонкие линии              |
| рейсфедер:<br>одноразовый | 8                               | 200  | постоянная толщина линии  |
| наполняемый               | 8                               | 200  | необходимо хорошо чистить |

## СЛОВАРЬ

CAD - Computer Aided Design, система автоматизированного проектирования, САПР

ADI - Autodesk Device Interface, аппаратный интерфейс Autodesk

HPGL - Hewlett-Packard Graphic Language, графический язык HEWLETT-PACKARD

DM/PL - Digital Microprocesor/Plotting Language, графический язык HOUSTON INSTRUMENT

ANSI - American National Standards Institut, Американский национальный институт стандартов

ISO - International Standards Organisation, Международная организация по стандартизации, ИСО

DIN - Deutsche Industry Normen, Германские промышленные стандарты

acceleration - ускорение

accuracy - точность

axial speed - скорость по оси

check-plot - контрольный (черновой) чертеж

diagonal speed - скорость по суммарному вектору

dpi - dot per inch, точек/дюйм

elektrophotography - электрофотография

ips - inch per second, дюйм/с

resolution - разрешение

repeatability - повторяемость

## ПЛОТТЕРЫ

belt plotter - конвейерный графопостроитель

dot-matrix plotter - матричный (мозаичный) плоттер

drum plotter - барабанный графопостроитель

electrostatic plotter - электростатический плоттер

flatbed plotter - планшетный графопостроитель

liquid ink-jet plotter - струйный плоттер

rollerbed plotter - рулонный графопостроитель

termal transfer plotter - термический плоттер

## БУМАГА, ПЛЕНКА

paper - бумага

transluscent paper - полупрозрачная бумага, калька

vellum - пергамент, калька

film - пленка

matte film - матовая пленка, обычно полиэфирная

clear film - прозрачная пленка, обычно ацетатная

## ПЕРЬЯ

fibre tip pen - фломастер

hard nib pen - фломастер с упрочненным штифтом

roller ball pen - шариковая ручка

drafting pen - рейсфедер

stainless steel drafting pen - рейсфедер со штифтом из нержавеющей стали

tungsten tip drafting pen - рейсфедер со штифтом из вольфрамовых сплавов

## ТУШЬ

water base ink - водяная тушь

oil base ink - маслянистая тушь

film ink - тушь для пленки



Компьютер на работе

Обработка Томаша Зелиньского

# Оценка качества перьевых плоттеров



В международном масштабе стандартный тестовый чертеж по норме ISO является признанным вспомогательным средством для оценки качества работы графопостроителей.

## ЦЕЛЬ ТЕСТА ISO

До недавнего времени сравнение качества работы плоттеров вызывало определенные трудности. Вопросы эстетики связаны с понятием вкуса, а фирмы-изготовители демонстрировали качество своих продуктов с помощью разных демонстрационных чертежей. Это положение вещей изменилось благодаря Германскому комитету нормализации, который ввел стандарт DIN 32866, содержащий тестовый чертеж для оценки качества работы векторных графопостроителей, который вскоре стал международным стандартом.

## ПРИМЕНЕНИЕ ТЕСТА ISO

С помощью стандартного чертежа пользователь может объективно оценить важнейшие свойства перьевого плоттера: точность установки пера в нужной позиции, повторяемость, качество черчения прямых, окружностей и шрифта. Каждая из 12 частей тестового чертежа служит самостоятельным критерием оценки. Дополнительно для устранения погрешностей, обусловленных видом материала, принято выполнять чертеж тушью на пленке.

Некоторые современные графопостроители снабжены программой, оптимизирующей черчение по цвету, т. е. сначала вычерчиваются все элементы одного цвета, затем другого и т. д. В связи с тем, что это нередко положительно влияет на качество работы плоттера, чертеж-тест должен быть в таком случае выполнен дважды: один раз - с включенной оптимизацией, второй раз - с выключенной.

## ОЦЕНКА КАЧЕСТВА ЧЕРТЕЖА

Оценка тестового чертежа должна проводиться по частям (зонам) в соответствии с критериями стандарта. Следует обратить внимание на то, выполняются ли элементы чертежа согласно рекомендациям стандарта и в указанной очередности. Важно также субъективное впечатление о качестве чертежа.

### ЗОНА 1: "Тонкий крест"



Состоящий из двух пересекающихся отрезков крест должен быть вычерчен дважды: в начале теста и в его конце. Фигуры должны точно совпадать, что позволяет оценить повторяемость плоттера. В идеальном случае два креста выглядят как один.

По нашему мнению, если бы второй крест был повернут по отношению к первому на 45°, то лучше было бы видно даже минимальное отклонение.

### ЗОНА 2: "Веер"

В этой части вычерчиваются расположенные веером отрезки с общей точкой в левом верхнем углу. Угловое расстояние между отрезками постоянное. Тест демонстриру-

ет способность плоттера чертить линии в любом направлении. Отдельные отрезки должны быть ровными, без зигзагов, а углы между ними - равными.

### ЗОНА 3: "Рыбьи кости"

Здесь чертится нечто напоминающее рыбьи кости - несколько групп, состоящих из трех отрезков. Они вычерчиваются поочередно слева направо и справа налево, а также сверху вниз и снизу вверх. Равномерность расстояний между отрезками свидетельствует о точности ведения пера и параллельности, а также позволяет оценить качество вычерчивания прямых. Волнистость векторов свидетельствует о динамических недостатках плоттера. Начала и концы отрезков должны точно совпадать. Частыми погрешностями бывают "перетянутые" и "недотянутые" отрезки (overshoot и undershoot).

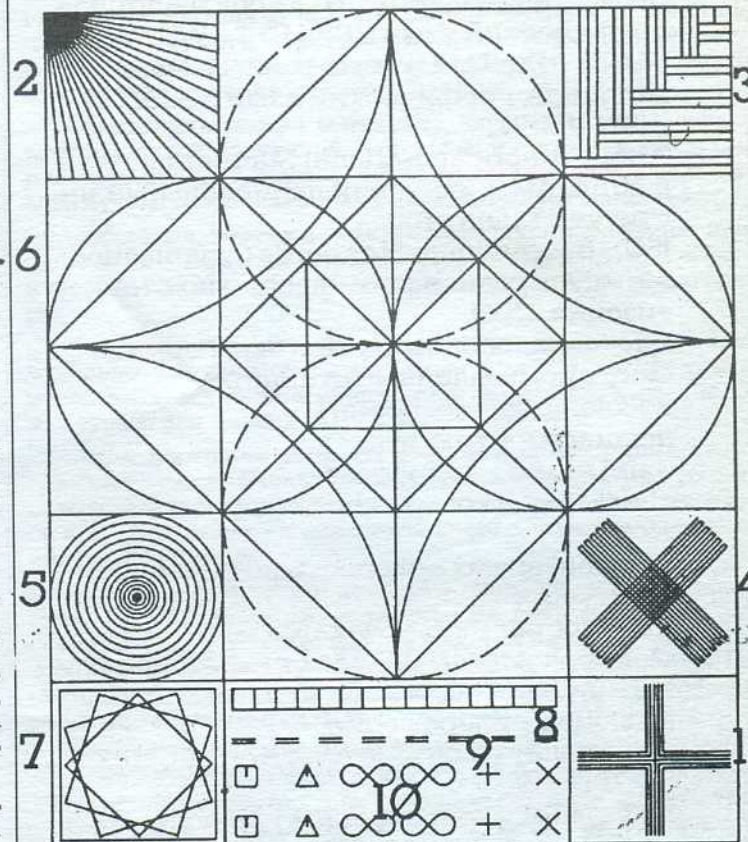


Кроме того, начало и конец отрезка должны быть четкими, т. е. без клякс (blobs) и пунктиров, которые возникают из-за "подпрыгивания" пера при его опускании (skipping).



### ЗОНА 4: "Штриховка"

В данной части теста по всей зоне вычерчиваются прямые под углом 45°, попеременно соединенные короткими отрезками, направленными к этим прямым под углом 90°. Вычерчиваются две группы линий и отрезков,



ABCDEFGHIJKLMNOPQRSTUVWXYZ  
abcdefghijklmnopqrstuvwxyz  
0123456789

## ЗОНА 1



расположенные по отношению друг к другу под углом  $90^\circ$ .

Здесь четко просматриваются достоинства и недостатки плоттера: динамические погрешности выявляются при вычерчивании длинных линий, а точность наведения пера — коротких отрезков.

#### ЗОНА 5: "Концентрические окружности"

Окружности вычерчиваются поочередно вправо и влево, причем начальная точка каждый раз сдвигается на  $30^\circ$ . Окружности не могут быть деформированными и должны быть расположены концентрически на равном расстоянии друг от друга. Начала и концы окружностей при вычерчивании не должны быть заметны. Попеременное изменение направления черчения является хорошей проверкой точности повторения. Оно не позволяет использовать в этом фрагменте теста содержащийся в ПЗУ плоттера генератор окружностей. Процедура Curve Management, встроенная в некоторые модели графопостроителей, существенно улучшает результат.

#### ЗОНА 6: "Окружности, дуги и квадраты"

В центральной части чертежа-теста выполняются окружности, дуги и квадраты. Некоторые из них чертятся пунктиром. Окружности должны быть без деформаций, а пунктирные линии иметь равные пробелы. Касательные должны точно совпадать с окружностями, а углы квадратов — с концами дуг.

#### ЗОНА 7: "Квадраты"

Три квадрата, расположенные друг по отношению к другу под углом  $30^\circ$ , выявляют способность плоттера чертить прямые в разных направлениях (не только в осевых). Линии должны быть прямыми, а углы квадратов — равными точно  $90^\circ$ .

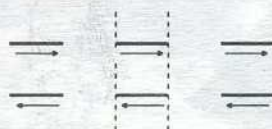
#### ЗОНА 8: "Меандры"

Здесь вычерчиваются два графика, состоящие из меандров и в результате дающие последовательность небольших квадратов. Эта часть теста особенно хорошо демонстрирует повторяемость плоттера, так как перо подводится к одним и тем же точкам с разных направлений. Дважды вычерчиваемые вертикальные отрезки должны выглядеть как один, а горизонтальные — превратиться также в один длинный отрезок.



#### ЗОНА 9: "Пунктиры"

Две горизонтальные пунктирные линии вычерчиваются параллельно друг другу слева направо и справа налево. В точных плоттерах начальные и конечные точки отрезков должны находиться точно одна над другой. Элементы линий должны быть равномерными.



#### ЗОНА 10: "Символы"

В этой зоне нужно особо присмотреться к горизонтальным "восьмеркам" с постоянно изменяющейся кривизной. Кроме того, следует обратить внимание на симметрию и места перехода прямых в кривые.

#### ЗОНА 11: "Текст"

Здесь вычерчиваются буквы алфавита, а также цифры. Необходимо обратить внимание на вид символов, их четкость. Не должно быть клякс и погрешностей типа незамкнутых окружностей или линий разной толщины.

#### ЗОНА 12: "Крест"

Эта фигура чертится с использованием четырех перьев и позволяет оценить повторяемость с учетом смены цвета. Линии должны быть параллельными, не "перетянутыми" или "недотянутыми", особенно в местах изменения цвета.

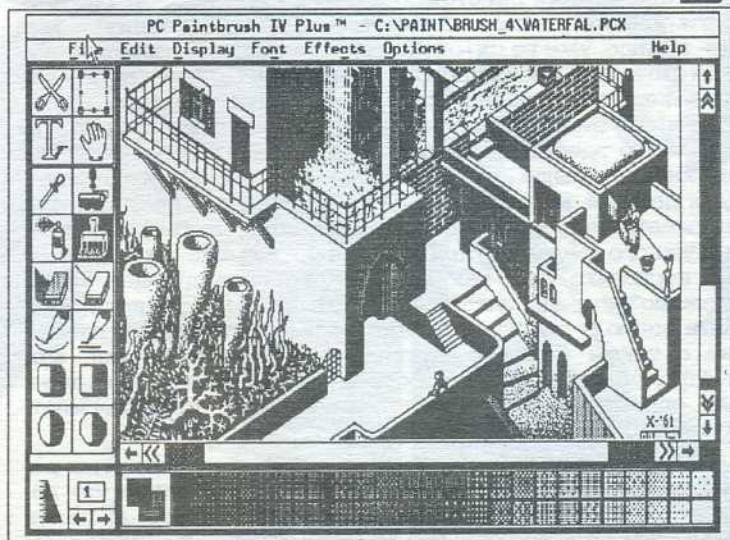
Перевод Анджее Поплавского

\компьютер на работе\

© Александр Григорьев

# Люмино- графия

Термин, позаимствованный из фототехники, но, на мой взгляд, полнее всего отражающий истинную природу рисования электронным лучом на люминофоре, вынесен в заголовок, как бы в противовес мнению одного почтенного художника, сравнившего компьютерную графику с вышиванием на пальцах крестом. Разумеется, его ироническое замечание относилось не к самому процессу создания изображений на экране дисплея с помощью графического пакета, а лишь о внешнем виде компьютерного имиджа, первооснову которого действительно составляет графическая точка — пиксел, имеющая форму прямоугольника. Что поделаешь, быть может только это видимое невооруженным глазом ограничение и не позволяет сегодня переместить компьютерную графику из прикладной области в область искусства, т.е. туда, где самооценочность творчества — менее похожими на растровые имитации фотографий. Компьютерная графика переживает период счастливого младенчества и, следовательно, по мере совершенствования техники и создания более гибких и мощных программных средств она будет привлекать все больше внимания. Пока же, скорее всего из-за слишком малого количества персональных ЭВМ в нашей стране, очень немногие дизайнеры или художники-графики имеют возможность работать непосредственно с графическими пакетами. Обидно сознавать, что большинство профессионалов все еще относится к компьютерному рисованию с долей скептицизма, а если и испытывают нужду в переносе на экран своих рисунков, то добиваются этого путем сканирования, отводя машине роль копировщика, а не инструмента, способного заменить и карандаш, и пантограф, и кисть, и палитру с







красками. Но, думаю, всем нам скоро придется изменить привычным стереотипам. Расширение компьютерного парка и в особенности появление настольных издательских систем, позволяющих предельно быстро производить верстку и создавать оригинал-макеты книг, наверняка подтолкнет советских художников-оформителей, а возможно даже и иллюстраторов, к люминографии, т.е. к созданию имиджей с помощью графического пакета.

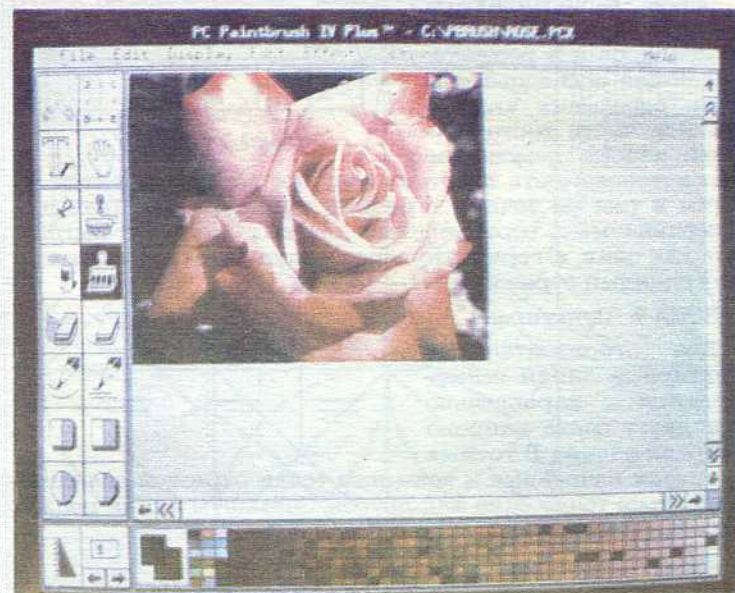
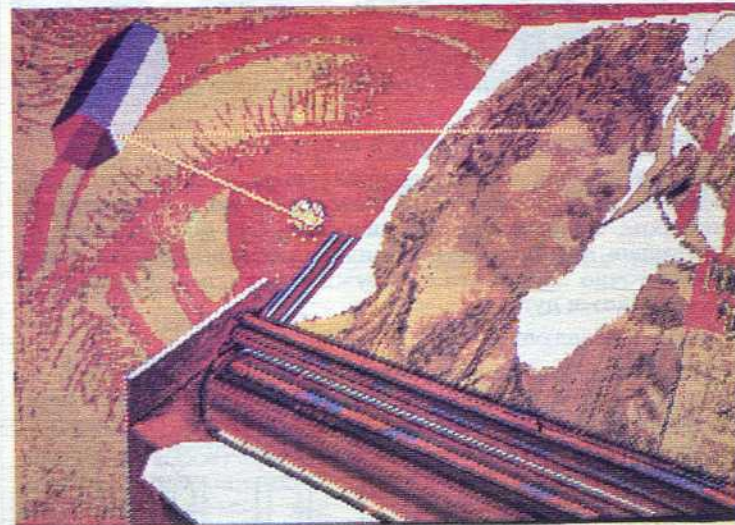
В настоящее время среди наших пользователей получили широкое распространение такие графические редакторы, как Show Partner, EGA Paint, Show F/X, Dr.Hallo, Microsoft Paintbrush, Story Board, Fantavision. Они позволяют выполнять различные графические работы широкого диапазона, начиная от простых геометрических построений и кончая мультимедийными объектами произвольной формы, с помощью персональных ЭВМ стандартной конфигурации, т.е. не ориентированных непосредственно на графику. Правда, рисование на экране дисплея с помощью "мыши" — процесс трудоемкий, но, осмелюсь утверждать, не более, чем любая деятельность художника, особенно если учесть, что в этом случае он лишен малопривлекательного соприкосновения с жидкими и сухими красителями, пульверизаторами, трафаретами, не нуждается в большом количестве дополнительных приспособлений, не требует специального помещения и т.д. Люминография бесплотна, но зрима; она притягивает, захватывает и покоряет. Нужно лишь привыкнуть к тому, что движение курсора (точки, рисующей на экране линию) в точности повторяет перемещение "мыши" по любой плоской горизонтальной поверхности. Но такой "свободный" режим рисования неудобен, так как манипулятор излишне чувствителен к любому, даже незначительному, часто произвольному движению кисти. Поэтому выполнить некий вразумительный рисунок таким способом крайне трудно — нужна определенная сноровка и твердая рука. Вот тогда-то и следует обратиться к графическому пакету, где описанный режим является одним из многих (и, как правило, реже всего применяемых) инструментальных средств.

По сути своей все графические пакеты — и не только вышеперечисленные — очень близки друг к другу и преследуют двоякую цель: максимально облегчить процесс рисования, а также упростить общение пользователя с компьютером во время создания имиджей. Из этих общих свойств вытекают различия, которые позволяют потребителю, во-первых, отличить один пакет от другого и, во-вторых, выбрать для своей работы наиболее приемлемый и удобный. Разумеется, удобство — понятие относительное, целиком зависящее от индивидуальных черт художника. Следовательно, мое предпочтение пакетов, имеющих постоянно присутствующее на экране символьное меню (Story Board, Fantavision, Paintbrush, Dr.Hallo), которое окружает область рисования наподобие рамки, никоим образом не умаляет достоинств таких пакетов, как EGA Paint или Show Partner, где экран всегда полностью открыт для рисования, а меню — правда, уже не символьные, а словесные — высвечиваются лишь при необходимости смены инструмента, цвета, применения одного из вспомогательных режимов и т.п. EGA Paint в этом отношении — редактор весьма оригинальный: меню здесь появляется при нажатии соответствующей кнопки "мыши" непосредственно в том месте экрана, где в данный момент находится курсор. По площади оно занимает очень мало места, но для того, чтобы добраться до нужного режима, приходится нажимать на клавишу еще несколько раз, так как каждое последующее меню "вложено" в предыдущее. Show Partner, с его транспарантными меню в этом отношении более доступен, не говоря уже о таком шедевре пиктографической выдумки, как Dr.Hallo, рамка-меню которого почти не содержит словесных определений и чем-то напоминает древнеегипетскую стеллу с иероглифами.

Набор инструментальных средств вышеупомянутых редакторов примерно равнозначен, исключая разве что пакет Dr.Hallo, где этих средств слишком много, и пакет Fantavision, где их слишком мало. "Специалист подобен

флюсу", — говорил Козьма Прутков и его справедливые слова можно без тени иронии адресовать и пакету Dr.Hallo, в котором отдано предпочтение инструментарию (среди большого количества инструментальных режимов можно заблудиться, словно в дремучем лесу, хотя такой нужный режим, как zooming, представлен одним малым увеличением пиксела), и пакету Fantavision, ориентированному целиком и полностью на создание анимационных эффектов в ущерб изобразительным средствам.

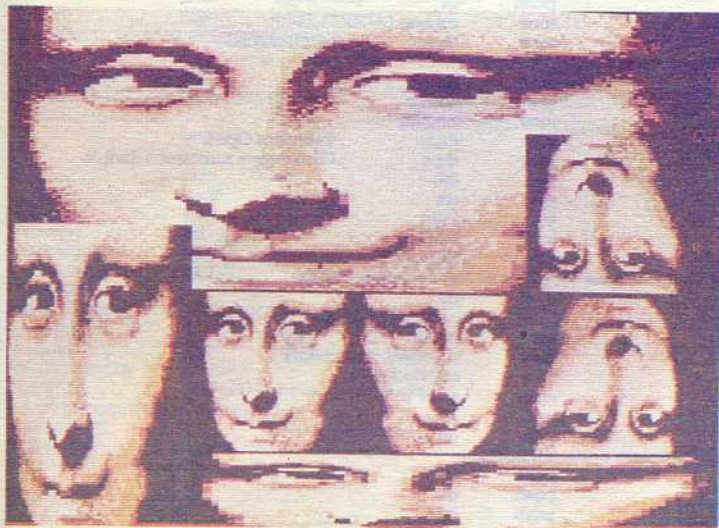
В поисках "золотой середины" можно остановиться на двух пакетах: Show F/X и Story Board. Они имеют приемлемый арсенал принадлежностей для рисования, а также достаточно действенный механизм экспонирования готовых изображений, включая анимацию отдельных экран-



ных объектов. Но все же, с моей точки зрения, редактор Microsoft Paintbrush наиболее удобен именно для рисования, именно для создания высококачественных имиджей, и работать на этом редакторе — одно удовольствие. Конечно же, и он не лишен недостатков (например, отсутствует режим пропорционального уменьшения/увеличения, невозможно перемещать отдельные части изображения при полностью открытом экране, да и открывать экран сравнительно долго, в отличие от пакетов Dr.Hallo и Story Board, где это делается одним нажатием пробельной клавиши; нет такого удобного средства, как редактор экранных шрифтов, которым так интересен Story Board; отсутствует механизм экспонирования и т.п.), а потому возникает проблема совместимости разных графических пакетов, иногда решаемая, чаще — нет, но всегда вызывающая у нашего потребителя массу затруднений.



И наконец, самое общее отрицательное свойство упомянутых пакетов состоит в том, что все они – "иностранцы", а русификация графического редактора – задача далеко не тривиальная, если конечно под русификацией понимать не включение в шрифтовую библиотеку нескольких самодельных рабочих шрифтов, а перевод на русский язык всех существующих в пакете меню и вспомогательных справочных материалов. Можно задаться вопросом: почему почти никто из импортных лаврах наши отечественные программисты? Почему они не создают свои собственные оригинальные графические программные продукты? Ответить на него нетрудно: из-за очевидной трудоемкости самого процесса и невозможности добиться финансирования будущей разработки. И тем не менее такие разработки существуют, хотя и не идут, как правило, ни в какое сравнение с их роскошными западными прообразами. Они весьма малочисленны, сделаны, что называется, "на коленке". Единственное исключение – графический редактор "Мультикорд", который поддерживает все старые режимы адаптеров EGA и VGA, удобен в работе и вполне конкурентоспособен с западными аналогами по своим функциональным возможностям. Опираясь на собственный (не

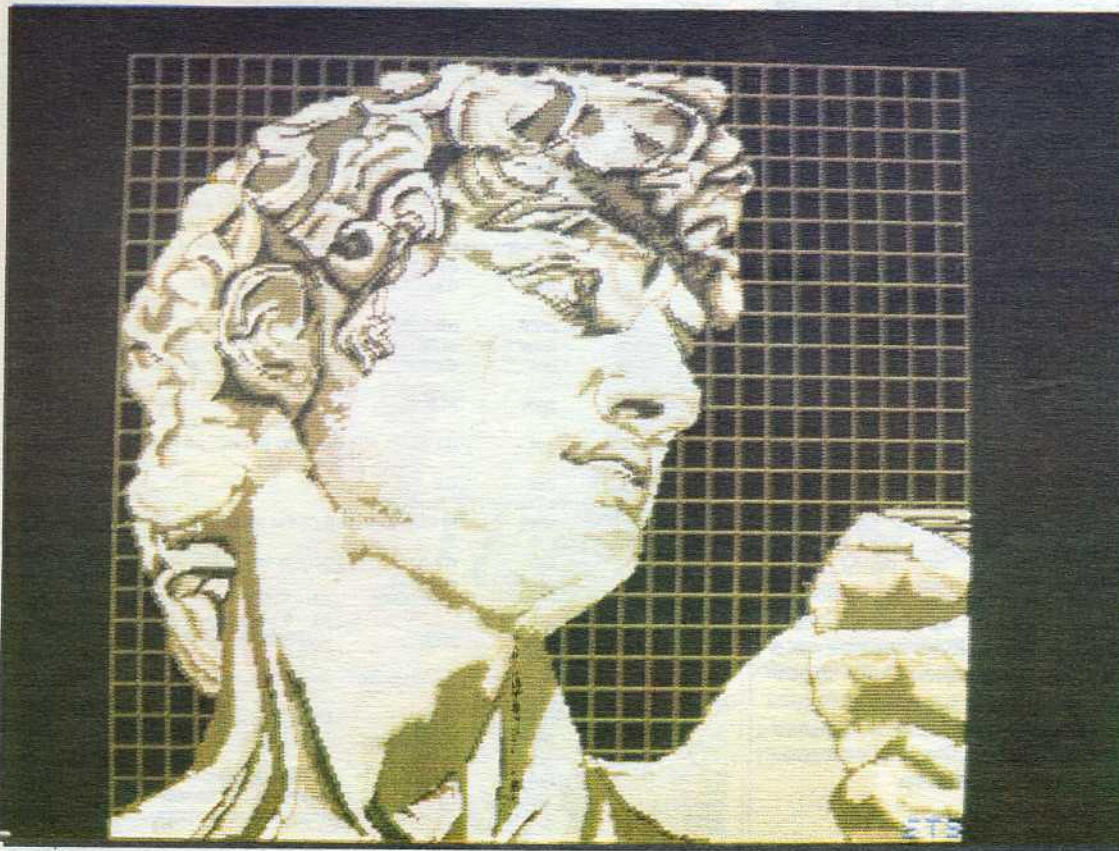


всегда положительный) опыт, хотелось бы посоветовать дизайнеру или художнику, решившему сменить рапидограф или кисть на "мышь" – приобрести англо-русский словарь и обратиться к любому имеющемуся в его распоряжении западному пакету. Этим он избавит себя от нервных стрессов, сможет войти в новую для него область наиболее безболезненным путем и встретит во всеоружии долгожданное появление на рынке полноценных отечественных пакетов, а оно, надеюсь, не за горами!

\* \* \*

Упоминаемые в статье графические редакторы позволяют создавать изображение растрового типа. В настоящее время все большее развитие и распространение получают пакеты, предназначенные для векторной графики (в том числе для векторного проектирования высококачественных шрифтов, используемых в настольных издательских системах). Этим программам мы посвятим статью в одном из ближайших выпусков "Компьютера".

Редакция





 \компьютер на работе\

© Михаил Сальников

# Заполнение видеобuffers EGA

В предыдущей статье ("Компьютер", N 2) мы дали обзор графических адаптеров и коротко остановились на особенностях EGA. Теперь мы рассмотрим пример программы, позволяющей вывести на экран заранее подготовленную картинку из файла на диске.

В программе показана работа с внутренними регистрами адаптера, точнее их установка для выброса картинки. Для корректировки данных видеобuffers в программе применяется режим записи EGA 0. В этом режиме корректировку можно выполнять как побайтно, так и по пикселям. Режим корректировки задается значением регистра Enable Set/Reset (0Fh - изменяется пиксел, 00h - байт). В зависимости от значения регистра Data/Rotate Function графический контроллер может выполнять с данными CPU и видеобuffers некоторые логические операции (AND, OR, XOR, Замена). В приведенной программе используется операция замены, т.е. данные из внутреннего буфера программы замещают данные в блоках памяти видеобuffers. Внутренний блок видеобuffers выбирается с помощью порта ввода-вывода Sequencer. После соответствующей установки регистров графического контроллера любая команда записи процессора по адресу видеобuffers (A000h) будет изменять хранящиеся в нем данные, а значит, и изображение на экране дисплея. Приведенный фрагмент программы взят из работающего оригинального графического пакета. Дополнительные пояснения к программе даны в комментариях к коду (не забывайте и техническое руководство по адаптеру). Попробуйте поэкспериментировать с приведенным фрагментом. Программа может применяться в качестве функции для программ на языке Си (модель Large). Попробуйте оптимизировать ее (при выделении фрагмента мы убрали 95% кода, включая внутренние подпрограммы, и могли оставить "мусор"), придумайте процедуру обработки ошибок открытия файла. В работающей программе выход осуществляется сложнее, поскольку нужно восстановить регистры EGA в том виде, в котором они были до входа в программу. Кроме того, следует обратить внимание еще на один вопрос. Хранить точный образ видеобuffers в файле крайне не экономично, поэтому должны существовать некоторые процедуры упаковки и распаковки картинки. Если вы знакомы с каким-либо из развитых графических пакетов, например PC PaintBrush, Story Board, Мультикорд и т.п., вы знаете, что каждый из них использует свой специальный формат хранения образов экрана в файлах, обеспечивающий значительное сжатие данных. Постарайтесь придать приведенному фрагменту законченный вид. Возможно, если в дальнейшем вы будете работать с другой моделью памяти Си, вам придется изменить команды приема параметров. Разработанная функция пригодится нам в дальнейшей работе.

```

Программа      OutScreen
Функции:
    Копирует содержимое внешнего файла в видеобuffer
    адаптера EGA побайтно строка за строкой.
    Внутренние блоки видеобuffers заполняются в
    следующем порядке: 3,2,1,0.

Вызов:
    _outscreen(char far *FileName)

Входные параметры:
    FileName - имя внешнего файла, содержащего
    образ видеобuffers EGA (картинку)

Ограничения: EGA 16 цветов, разрешение 640 x 200

a      EQU Word ptr [bp+6]
b      EQU Word ptr [bp+8]
.MODEL LARGE
.CODE OUTSCR_TEXT
    ASSUME CS:OUTSCR_TEXT,DS:OUTSCR_DATA
;SS:OUTSCR_STACK

;OUTSCREEN
PUBLIC  _OUTSCREEN
PROC   _OUTSCREEN
    push    bp
    mov     bp,sp
    push    ds
    push    si
    push    di
    mov     ax,seg OUTSCR_DATA
    mov     dx,ax    ;Загрузка DS

;открытие внешнего файла
    push    ds
    xor     ax,ax
    mov     ah,3Dh
    mov     dx,b
    mov     dx,a
    int     21h
    pop     ds
    jnc     L001
    jmp     FILEerr
L001:    mov     mov     Handle,ax

;чтение внешнего файла во внутренний буфер
    push    ds
    mov     ax,seg BUFFERS
    mov     dx,ax
    mov     dx,offset ScreenBuff
    mov     cx,80*200*4
    mov     ah,3Fh
    int     21h
    pop     ds

;сохранение ряда регистров
    push    si
    push    di
    push    cx
    push    ax
    push    bx
    push    ds

;установка адресов
    push    ds
    mov     ax,seg BUFFERS
    mov     dx,ax
    mov     si,offset ScreenBuff ;SI-буфер в RAM
    pop     ds
    mov     ax,0A000h
    mov     es,ax
    ;ES:DI - Адрес видеобuffers

    xor     di,di

;установка графического контроллера для режима записи 0
    xor     ax,ax
    mov     dx,3CEh ;DX - адрес порта
    ;графического контроллера
    mov     ah,00h ;функция Data/Rotate (Replace)
    mov     al,3h  ;номер регистра Data/Rotate
    out     dx,ax  ;выбор регистра 03H
    ;графического контроллера
    mov     al,5   ;AL = 5 - номер регистра графического режима
    out     dx,al  ;выбор регистра графического режима

    inc     dx
;3CFh - номер порта регистра данных
    mov     al,00h ;AL=0 режим записи EGA 0

;режим чтения EGA 0
    out     dx,al  ;установка режима записи EGA 0
    mov     dx,3CEh
    mov     ax,0000h
    ;AH = 0 (Режим изменения байта)
    ;AL = 8 - Номер Bit Mask регистра
    ;Изменение байта в видеобuffers
    out     dx,al

;Цикл по строкам
    mov     LoopVar,200 ;загрузка числа строк
Loop0:    pop     bp
    mov     di,0C4h ;DX=3C4h (Sequencer I/O port)
    mov     ax,0002h ;AL=1000b (3-й блок памяти)
Loop1:    pop     bp
    out     dx,ax ;выбор следующего блока
    ;внутренней памяти EGA
    ;для записи информации

```



```

push    di
push    ax
mov     cx,80      ; число байт в строке блока
push    ds
mov     ax,seg BUFFERS
mov     ds,ax
rep     movsb
pop     ds          ; восстановление регистров
pop     ax
pop     di
shr     ah,1        ; AH= новое значение Map Mask
                        ; (номер внутреннего блока)

jnz     Loop1
add     di,80
dec     LoopVar
jnz     Loop0

; восстановление регистров
pop     ds
pop     bx
pop     ax
pop     cx
pop     di
pop     si

L93:
pop     ax,Handle
mov     ah,3Eh      ; INT 21h функция 3Eh
                        ; закрытие файла

int     21h
jmp     Endoutscr
nop

FILEerr:
; Обработка ошибок открытия файла

Endoutscr:
nop
; возврат в вызывающую программу

pop     di
pop     si
pop     ds
mov     sp,bp
pop     bp
ret

_OUTSCREEN
OUTSCR_TEXT
FARDATA
OUTSCR_DATA SEGMENT byte public 'DATA'
LoopVar     DB      (0)
Handle      DW      (0)
OUTSCR_DATA ENDS
BUFFERS     SEGMENT byte public 'DATA'
; внутренний буфер программы
ScreenBuff  DB      80*200*4 dup (?)
BUFFERS     ENDS
END

```

\компьютер на работе\

© Мариуш Дец, Марек Матушак, Марек Млынарский

# Компьютерный склад информации

Продолжаем описание гибких дисков, начатое во втором выпуске "Компьютера".

В связи с тем, что в настоящее время наибольшее распространение получил формат D-9, структуру ТРФ и каталога проанализируем именно для этого формата. Однако сначала попытаемся подытожить сказанное выше и рассмотрим логическую организацию дискеты, записанной в формате D-9.

Таблица 3

Логическая организация дискеты MS-DOS (формат D-9).

| Номер стороны | Номер дорожки | Номер сектора | Логическая область дискеты |
|---------------|---------------|---------------|----------------------------|
| 0             | 0             | 1             | Загрузочный сектор         |
| 0             | 0             | 2-3           | ТРФ (первая копия)         |
| 0             | 0             | 4-5           | ТРФ (вторая копия)         |
| 0             | 0             | 6-9           | Каталог                    |
| 1             | 0             | 1-3           |                            |
| 1             | 0             | 4-9           | Область                    |
| 0,1           | 1-39          | 1-9           | данных                     |

## Как найти файлы?

Минимальная часть диска, резервируемая для одного файла, называется единицей распределения памяти (кластером). Это означает, что даже если файл состоит всего из 1 байта, а единицей распределения памяти является, например, 1 Кбайт, то под этот файл на диске будет выделен 1 Кбайт. Следует подчеркнуть, что кластер объемом 1 Кбайт может состоять из одного сектора длиной 1 Кбайт, двух секторов длиной по 512 байт и т.д.

В системе MS-DOS единицей распределения памяти является кластер, состоящий из одного или двух секторов длиной 512 байт (в зависимости от формата дискеты). В рассматриваемом формате D-9 кластер состоит из двух соседних секторов.

ТРФ состоит из 12-битовых элементов. Первый элемент содержит идентификатор формата дискеты (см. табл. 4), второй не используется (содержит код FFF - двенадцать бит), а остальные выполняют две функции:

- \* определяют статус кластера (свободен, занят, поврежден);

- \* определяют логическую очередность кластеров, принадлежащих файлу.

Как видно из табл. 4, каждый элемент ТРФ (за исключением двух первых) однозначно связан с определенной парой секторов области данных.

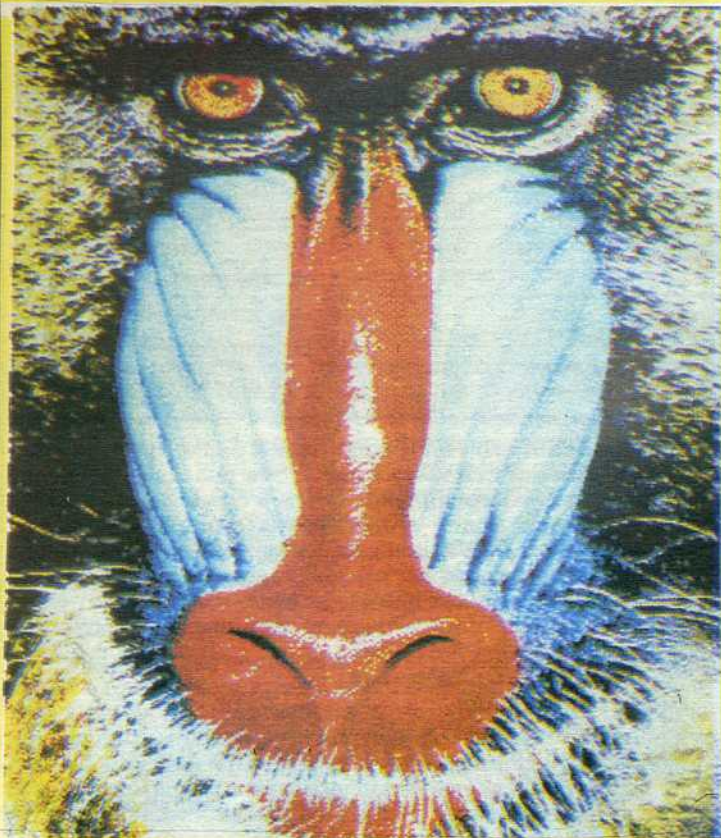




Таблица 4

## Структура таблицы размещения файлов.

| Элемент ТРФ | Функция                                                 |
|-------------|---------------------------------------------------------|
| 0           | идентификатор формата                                   |
|             | FFE - формат S-8                                        |
|             | FFF - формат D-8                                        |
|             | FFC - формат S-9                                        |
|             | FFD - формат D-9                                        |
|             | FF9 - формат QD-9 или QD-15                             |
| 1           | не используется (код FFF)                               |
| 2           | статус кластера 2 (секторы 4 и 5, дорожка 0, сторона 1) |
| 3           | статус кластера 3 (секторы 6 и 7, дорожка 0, сторона 1) |
| 4           | статус кластера 4 (секторы 8 и 9, дорожка 0, сторона 1) |
| 5           | статус кластера 5 (секторы 1 и 2, дорожка 1, сторона 0) |
| *           | *                                                       |
| *           | *                                                       |
| *           | *                                                       |

Кластеры обозначаются последующими числами, начиная с 2. Номер последнего кластера зависит от формата дискеты; для формата D-9 он составляет 355. В связи с тем, что количество кластеров превышает 256, каждый элемент ТРФ, как уже говорилось, кодируется 12 битами (для жестких дисков большой емкости 16 битами). В связи с этим элементы ТРФ следует считать парами, всякий раз анализируя три очередных байта. Чтобы разъяснить, каким образом на этой основе создаются 12-битовые значения, возьмем следующий пример: если упомянутые три байта содержат AB CD EF (шестнадцатерично), то закодированные в них элементы ТРФ имеют значения DAB и EFC.

Элементы таблицы размещения файлов, определяющие статус кластера, могут принимать следующие значения:

000 - кластер свободен,

002-FF0 - кластер занят (за исключением последнего кластера файла),

FF1-FF7 - кластер поврежден (чаще всего употребляется код FF7),

FF8-FFF - кластер занят, последний кластер файла (чаще всего используется код FFF).

Перейдем теперь к ответу на вопрос, как на основе ТРФ определить размещение файла на диске. Вы уже знаете, как выглядит последний кластер файла, и помните, что номер кластера (номер элемента ТРФ) однозначно присвоен паре секторов. Нужна еще следующая информация: как выяснить, где находится начало файла и как найти очередной кластер, зная номер n-го кластера файла.

Номер первого элемента ТРФ, связанного с данным файлом, представлен в каталоге. Допустим, что для некоторого файла он имеет значение 3. Если в третьем элементе ТРФ записано значение FFF (конец файла), то это означает, что кластер 3 (секторы 4 и 5 нулевой дорожки первой стороны дискеты) содержит весь файл (его длина не превышает 1024 байт). Если бы файл был больше, то в 3-м элементе ТРФ (прочитанном в каталоге) находился бы номер кластера, содержащего дальнейшую часть файла (номер, указывающий очередной элемент ТРФ).

Рассмотрим еще один пример. На этот раз предположим, что в соответствующем месте каталога указан второй элемент ТРФ. Кроме того, пусть содержимое ТРФ

аналогично представленному в табл. 5. На основании значения, прочитанного в каталоге, можно выяснить, где на дискете находится начало файла. Затем проверим содержимое второго элемента ТРФ. Оно составляет 005, т.е. вторая часть файла размещена в кластере с номером 5. В свою очередь пятый элемент ТРФ указывает, что третья часть файла расположена в третьем кластере. Столь же легко найти и очередные три части файла (кластеры 4, 7 и 6). Поскольку шестой элемент ТРФ содержит код FFF, то шестой кластер и является концом цепочки.

Необходимо, однако, помнить, что одной таблицы размещения файлов недостаточно для того, чтобы найти первый кластер файла. Нужно еще разобраться в структуре каталога.

Таблица 5

## Пример размещения файла на диске

| Номер элемента ТРФ | Содержимое | Размещение файла на диске |         |        | Очередность частей файла |
|--------------------|------------|---------------------------|---------|--------|--------------------------|
|                    |            | сторона                   | дорожка | сектор |                          |
| 2                  | 005        | 1                         | 0       | 4,5    | 1                        |
| 3                  | 004        | 1                         | 0       | 6,7    | 3                        |
| 4                  | 007        | 1                         | 0       | 8,9    | 4                        |
| 5                  | 003        | 0                         | 1       | 1,2    | 2                        |
| 6                  | FFF        | 0                         | 1       | 3,4    | 6                        |
| 7                  | 006        | 0                         | 1       | 5,6    | 5                        |

## Что вы найдете в каталоге MS-DOS?

В каталоге системы MS-DOS каждый файл описан с помощью 32 байт, образующих элемент (строку) каталога. Таким образом, каждый сектор каталога содержит 16 строк. В одной из них (обычно в первой) может быть записано имя диска (метка тома). Тип и размер полей, из которых состоит каждая строка каталога, представлены в табл. 6.

Таблица 6

## Структура единицы каталога системы MS-DOS.

| Относительный адрес | Размер поля | Описание                        |
|---------------------|-------------|---------------------------------|
| 0                   | 8           | имя файла                       |
| 8                   | 3           | тип файла                       |
| 11                  | 1           | атрибут                         |
| 12                  | 10          | резервное поле (нули)           |
| 22                  | 2           | время создания файла            |
| 24                  | 2           | дата создания файла             |
| 26                  | 2           | номер начального кластера файла |
| 28                  | 4           | длина файла                     |

Имя файла и его тип (расширение) записываются в кодах ASCII. При записи имени диска эти два поля объединяются, т.е. для него предназначаются 11 символов. Неиспользованные байты полей заполняются символом пробела. Первый байт поля имени файла используется также для обозначения стертых файлов (шестнадцатеричный код E5) и свободных строк в каталоге (код 00).

Байт атрибута файла может принимать следующие шестнадцатеричные значения:

01 - файл только для чтения,

02 - скрытый файл,

04 - системный файл,

08 - имя диска,

10 - подкаталог,

20 - архив (так классифицируются, например, файлы пользователя).





Скрытые и системные файлы недоступны для типичных команд ОС, таких как DIR. Подкаталоги функционируют на правах обычных файлов.

Слова, описывающие время и дату создания файла, рассчитываются по следующим формулам:

$$\text{время} = \text{час} * 2048 + \text{минуты} * 32 + \text{секунды} / 22$$

$$\text{дата} = (\text{год} - 1980) * 512 + \text{месяц} * 64 + \text{день}$$

Длина файла представлена в байтах. Добавим еще, что последние четыре поля (время, дата, номер первого кластера длины файла) записываются, начиная с младшего байта. Например, в поле длины файла, размер которого составляет 513 байт, будут записаны следующие величины: 01, 02, 00, 00.

### Что вы найдете в каталоге CP/M?

Нет общего правила, определяющего размещение каталога на диске системы CP/M. Причина заключается в том, что в старых версиях системы CP/M несколько начальных дорожек отводилось для модулей операционной системы. В ряде случаев в системе CP/M работают односторонние дискеты. В связи с этим ограничимся описанием структуры каталога именно таких дискет.

Каталог односторонней дискеты CP/M занимает обычно 1 Кбайт и состоит из 64 элементов размером в 32 байта. Иначе говоря, на дискете может быть записано не более 64 файлов.

Первый байт строки каталога содержит информацию о его занятости. Значение E5 (шестнадцатерично) свидетельствует о том, что строка свободна. Каждое другое значение говорит, что строка занята находящимся на дискете файлом или именем дискеты. Младший полубайт содержит информацию о номере пользователя, которому принадлежит файл. Старший полубайт позволяет определить,

что содержит данное место каталога: имя файла, обычный файл или же файл, защищенный "паролем".

Следующие 11 байт содержат имя файла (8 байт) и его тип (3 байта). Старший бит первого байта типа определяет статус файла. Значение бита, равное 1, разрешает только чтение, значение, равное 0, проведение любых операций. Старший бит второго байта типа определяет принадлежность файла к "подкаталогу" системных файлов - значение, равное 1, или к файлам пользователя - значение, равное 0.

Если описание файла занимает больше одной строки каталога, то байт, находящийся за последним символом типа, содержит: для первого места - 0, для второго - 1 и т.д.

Четырнадцатый и пятнадцатый байты строки каталога резервируются для системы, а в случае файлов с последовательным доступом содержат нули.

Шестнадцатый байт строки каталога содержит количество стандартных блоков CP/M размером 128 байт, входящих в состав файла или его фрагмента, описываемого на данном месте каталога.

Следующие байты (17-32) описывают номера единиц распределения памяти (кластеров), содержащих данный файл. На основании номеров кластеров система рассчитывает соответствующие им номера дорожек и секторов, в которых записан файл. В случае файлов, защищенных "паролем" на этих позициях записаны данные, необходимые для его идентификации.

О дискетах можно написать по крайней мере в четыре раза больше. Жаль, что на этот раз пора уже кончать наш рассказ, но мы еще вернемся к этой теме в очередных выпусках "Компьютера".

Перевод Анджея Попплавского



# С ПК по телефону

Сегодня электронная почта все настойчивее входит в повседневную деловую жизнь предприятий, организаций и многочисленных индивидуальных пользователей персональных компьютеров. Для обеспечения оперативного и надежного потока информации по телефонной сети организациям, имеющим удаленные филиалы, фирмам, распространяющим программные продукты, и промышленным предприятиям, налаживающим собственные производственные и коммерческие сети, рекламным агентствам, редакциям и издательствам целесообразно использовать модемы.

Центральная научно-исследовательская лаборатория МЕНАТЕП предоставила редакции сборника "Компьютер" разработанные и выпускаемые ей модемы "Lexand TS-2400", предназначенные для компьютеров, совместимых с IBM PC.

Модем работает в трех режимах передачи данных: 600, 1200 и 2400 бит/с. При работе в режимах 600 и 1200 бит/с. он полностью удовлетворяет международному стандарту CCITT v.23, а в режиме 2400 используется оригинальный протокол разработчика с циклическим помехозащищенным кодом, адаптированным к особенностям советских телефонных сетей. Модем устанавливается в свободный разъем на системной шине компьютера. Он гальванически развязан с телефонной линией при помощи оптоэлектронного устройства, что надежно защищает персональный компьютер.

Разработанный для модема "Lexand TS-2400" пакет прикладных программ легко может быть освоен начинающим пользователем. Основная программа пакета (LINE.EXE) может работать как с русским, так и с английским текстом меню. Полная система подсказок работает на любом уровне меню. Программное обеспечение предусматривает автоматический набор номера вызываемого абонента по любым телефонным линиям - внутренним, городским, междугородным. Если номер абонента занят, автомат будет осуществляться до тех пор, пока телефон не освободится. Аналогично работает и автоматический вызов по междугородней телефонной сети. В электронной записной книжке может содержаться до 300 номеров абонентов. Во время приема и передачи файлов для корреспондентов предусмотрена визуализация текущего состояния системы и статистики приема/передачи.

Пакет прикладных программ организован таким образом, что модемы при связи между собой предварительно тестируют телефонную линию и "договариваются" на какой скорости они будут вести передачу. Если в процессе передачи/приема качество связи ухудшается, они автоматически переходят на более низкую скорость. Как правило, в пределах одного города (Москвы) работа идет на максимальной скорости. При междугородней связи Москва-Ленинград, Минск, Киев, Рига, Вильнюс, Таллинн скорость - 2400 бит/с. А вот на больших расстояниях, например Москва-Новосибирск, работа идет, как правило, на скорости 1200 бит/с.

Программное обеспечение модема предполагает возможность работы в автоматическом "безоператорском" ре-

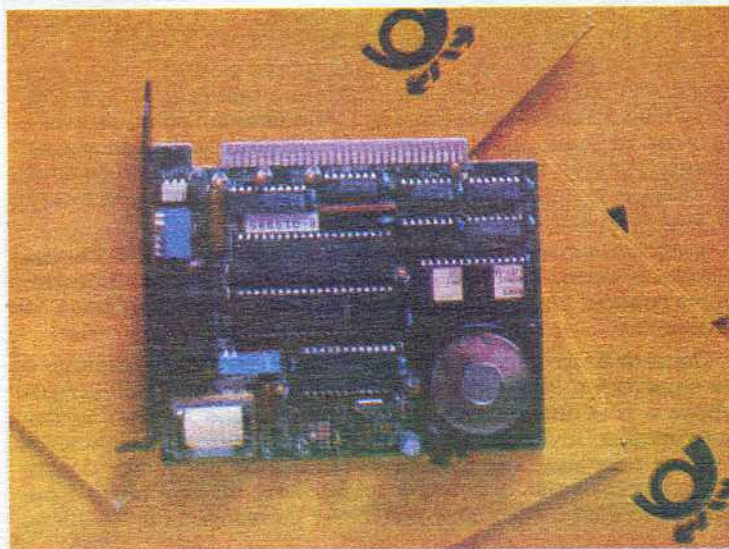
жиме на приемном конце. При этом в оперативную память загружают программу-драйвер (MODEM.EXE) в качестве резидента. Тогда модем сам поднимет трубку, ответит корреспонденту и, если корреспондент знает ваш пароль, примет файл или группу файлов и разместит их в заранее заданной пользователем области на жестком диске.

Настройка модема осуществляется программой-драйвером при помощи команды MODEM/I. При этом пользователь должен ответить на несколько вопросов. Адрес модема меняется только в том случае, если оригинальная разработка пользователя имеет системный адрес модема. Ваше имя при желании может автоматически передаваться корреспонденту для идентификации принятой информации в безоператорском режиме. Указывая путь к каталогу сообщений, вы организуете на жестком диске "почтовый ящик", куда в ваше отсутствие будет поступать информация по модему. В этом "почтовом ящике" автоматически создается и дополняется файл MAIL.LST, где накапливаются сообщения, какие файлы, кто и когда вам передал. Установка количества звонков для активизации модема позволяет пользователю определить, после которого телефонного звонка его модем САМ поднимет трубку. Остальные установки служат для настройки модема на конкретную, "не очень хорошую" телефонную линию.

Пакет прикладных программ модема постоянно совершенствуется. Например, еще совсем недавно нельзя было работать в "безоператорском" режиме, теперь такая возможность появилась. Следующая разработка программного обеспечения модема будет иметь терминальный режим, который позволит просматривать разрешенные области жесткого диска в отсутствии корреспондента и передавать их на свой компьютер. Предусмотрена возможность связи с зарубежными модемами (V-modem и Z-modem). Кроме того, будет описана работа программы-драйвера. Это позволит пользователю самому писать программное окружение, что может быть очень полезным при создании специализированных АРМ, расширении возможностей локальных сетей, разработке различных информационных систем.

Центральная научно-исследовательская лаборатория МЕНАТЕП была создана в 1989 году. Она объединила талантливых специалистов в различных областях физики и электроники. Лаборатория начала свою деятельность с разработок наукоемкой электроники для экспериментаторов широкого профиля, медицинской электроники и средств телекоммуникации. В настоящее время ведутся разработки по следующим основным направлениям:

- \* создание программно-аппаратных сред;
- \* медицинская электроника;
- \* магнитооптика.





\ в мире "Компьютера" \

© Роман Хмелевский

# Дискотека

Московская фирма ТЕРМИНАЛ предоставила нам для ознакомления несколько своих программных продуктов. Все они отличаются высоким профессиональным исполнением.

## ДЖИНН 1.0

Эта программа представляет собой резидентную справочную систему по терминологии ПЭВМ и командам MS-DOS. После загрузки в память компьютера программа обеспечивает пользователю мгновенный доступ к справочной информации. Работая с любой программой или находясь в командной строке MS-DOS, достаточно указать курсором на английское слово или название команды и нажать управляющую клавишу. На экране тут же появится окно с переводом и толкованием этого слова или словосочетания, в которых оно употребляется. Пользователь может читать указанную информацию или перемещаться по перекрестным ссылкам. Получив нужные сведения, можно спокойно покинуть программу и продолжить работу - экран восстанавливает свое прежнее состояние. Есть также возможность скопировать описа-

ние из Джинна непосредственно в редактируемый текст (если используемый редактор или текстовый процессор работает в текстовом режиме).

Авторы программы предусмотрели также режим, в котором можно производить поиск по русским и английским словам. В отличие от других систем подобного типа Джинн обеспечивает не только перемещение по перекрестным ссылкам, но и быстрый поиск и группировку информации, связанной с любым из нескольких тысяч или десятков тысяч слов.

Словарь программы Джинн включает следующие три раздела:

- терминологию по ПЭВМ - более пяти тысяч понятий и терминов с переводами и краткими пояснениями, специально подготовленными автором "Англо-русского словаря по программированию и информатике" А.В. Борковским;
- шесть тысяч широко используемых английских слов и выражений с переводом на русский язык;
- справочник по командам MS-DOS.

Словарь и справочники занимают на жестком диске около 1 Мбайта.

В предоставленный нашей редакции пакет

Джинн входят три дискеты - одна установочная и две с файлами в упакованном виде. На установочной дискете заходится программа установки и инструкция к ней.

Все программы, представленные нам для описания ТЕРМИНАЛОМ, в процессе установки на жесткий диск привязываются к компьютеру. Их нельзя перенести на другой компьютер. Количество машин, на которых можно установить Джинн, фиксировано на установочной дискете, а сама дискета защищена от копирования.

Джинн, согласно описанию, работает на любом совместимом с IBM PC компьютере с винчестером, операционной системой PC-DOS или MS-DOS версии 3.00 либо старше и резидентной частью 40 Кбайт ОЗУ. К выпуску готовится версия 2.0.

## Программа: ДЖИНН 1.0

Автор: Аркадий Борковский  
Язык: Си

Форма распространения: 1 дискета 5,25" DS/DD

Документация: файл с описанием на дискете

Цена: 450 руб. за 1 установку

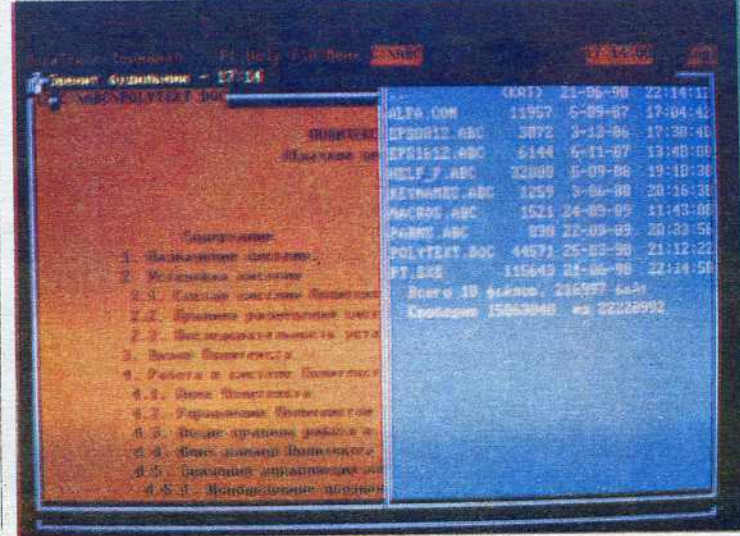
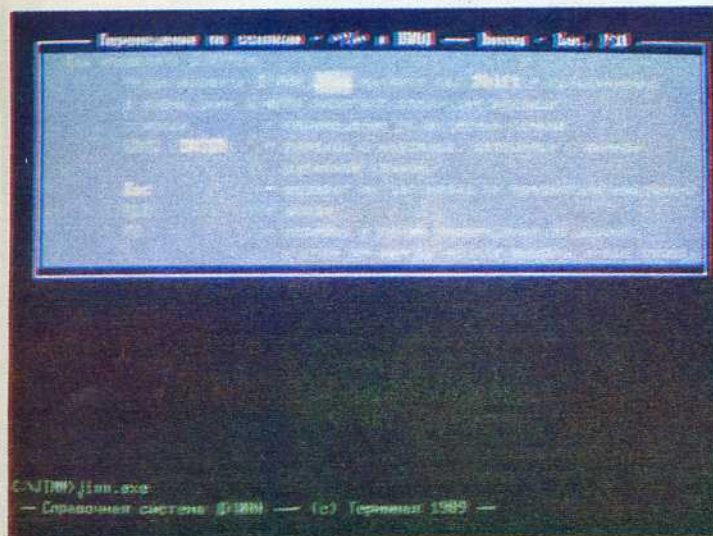
## ПОЛИТЕКСТ

Это новая версия программы ABC и очередной оригинальный текстовый процессор для IBM PC, разработанный советскими программистами. До него были Лексикон и Микро-Мир.

Программа, которую предлагает московская фирма ТЕРМИНАЛ, эффективно сочетает функции си-

стемы подготовки текстов и файловой системы (этим она несколько напоминает МикроМир). Предоставляя в распоряжение пользователя мощный и гибкий аппарат работы с файлами, система одновременно дает широкие возможности обработки текстовой информации.

Многооконный экранный редактор Политекста позволяет создавать как документы, так и программы на любом языке программирования, но при условии, что в вашем компьютере русские буквы расположены в соответствии с альтернативной кодировкой (см. "Компьютер", N 1). Работая с ним, можно использовать кириллицу и латинский алфавит в произвольных сочетаниях. Отдельные символы, слова, строки или целые фрагменты легко передвигать по всему тексту. При редактировании нескольких текстов, размещенных в различных окнах, можно переносить целые фрагменты из одного окна в другое. Документы можно форматировать автоматически, устанавливая абзацный отступ, поля и разбивая их на страницы. Подготовленный материал легко вывести на наиболее распространенные принтеры, причем печать выполняется как в графическом, так и в текстовом режимах (пакет включает файлы со шрифтами для печатающих устройств, работающих в режиме EPSON STANDARD). Интересной особенностью Политекста является возможность совмещения обработки одного текста с печатью другого (в фоновом режиме).







Система управления файлами Политекста заменяет все файловые операции ОС ДОС. Для просмотра файла или каталога стоит лишь "указать" его. Команды CHDIR, DIR, TYPE во время работы с Политекстом не нужны. Таким же образом, не прибегая к программе распаковки и не выходя в ДОС, можно просмотреть и архивный файл с расширением имени .ARC. После указания такого файла на экране появится его оглавление.

Столь же просто в среде Политекста выполняется копирование, перенос, удаление и переименование отдельных файлов и их групп. С помощью этой программы можно просмотреть даже "невидимые" в ДОС файлы. Программа также частично заменяет Norton Commander или PC Tools, позволяя изменять любой атрибут файла: сделать его невидимым, защитить от записи или удаления, объявить системным или архивным. Наконец, она позволяет запустить .COM- и .EXE-файлы.

Итак, вызвав Политекст, можно его не покидать в течение всего рабочего дня.

Программа снабжена хорошо продуманным интерфейсом пользователя. Информация занимает не весь экран, а лишь окно, форму и размеры которого пользователь может установить сам. Кроме окон, на экране всегда присутствует список доступных команд. Описание возможных действий (HELP) в любой момент можно получить, нажав одну клавишу.

Политекст позволяет программировать клавиатуру так, что цепочка часто употребляемых действий будет автоматически выполняться после нажатия всего одной клавиши (так называемые макрокоманды).

Предоставленный нашей редакции для ознакомления пакет Политекст записан на одной дискете. На ней помещаются программа установки Политекста, описание самой программы, а также файлы с параметрами окон и описанием начального состояния системы, со значениями макроопределений, с макроопределениями и со шрифтами.

У Политекста есть один недостаток: плохой драйвер клавиатуры, который, кстати говоря, устанавливается отдельно. Для набора этого текста мне пришлось обратиться к польской программе, которая позволила свободно переключать режимы работы клавиатуры, не прибегая к помощи команд Политекста.

Программа: ПОЛИТЕКСТ 1.0  
Автор: Аркадий Борковский  
Язык: Си  
Форма распространения: 1 дискета 5,25" DS/DD  
Документация: файл с описанием на дискете  
Цена: 700 руб. за 1 установку

## ЩИТ

"В СССР и некоторых других странах отсутствует законодательство, защищающее авторские права создателей программных продуктов. Отсутствуют также механизмы, регулирующие распространение программных продуктов, в том числе так называемых "фирменных", т.е. завезенных из-за границы и неуправляемо тиражируемых без каких-либо лицензий на это. Последствием этого является: отсутствие стимулов разработчикам к распространению своих продуктов, отсутствие рынка программных продуктов, размытое понятие надежности программного обеспечения, невероятная по масштабности завирусованность ПЭВМ в СССР. До сих пор некоторые администраторы считают диким покупать программные продукты, когда их так легко украсть. Возникла некая экологическая ниша в данной проблеме, которую мы пытаемся заполнить", - пишут в руководстве к своему очередному программному продукту представители фирмы ТЕРМИНАЛ.

Программная система Щит, которую предлагает ТЕРМИНАЛ, предназначена для обеспечения защиты пользовательских программ от несанкционированного копирования, а также для установки защищенных программ на жесткий диск. Защита от копирования заключается в "привязке" защищаемых файлов с расширением .EXE и .COM

(их может быть до 29) к дискете типа DS/DD (360 Кбайт). Привязанный или защищенный файл с помощью специальной программы может быть установлен заранее заданное (при формировании дискеты) число раз (до 99) на жесткий диск ПЭВМ и работоспособен только на данной ПЭВМ.

Программная система Щит работает в среде операционных систем, совместимых с MS-DOS 3.0 и старше.

Программа: ЩИТ 1.0  
Автор: Аркадий Большаков  
Язык: Си, Ассемблер  
Форма распространения: 1 дискета 5,25" DS/DD  
Документация: файл с описанием на дискете  
Цена: 800 руб. за 1 установку

## ШПРИЦ

Очередной продукт ТЕРМИНАЛА представляет собой своеобразную антивирусную программу-оболочку. Она довольно эластична: количество работающих с ней антивирусных программ можно увеличивать при появлении их новых версий.

В исходном варианте Шприц позволяет проверить диск при помощи следующих антивирусных программ:

ANT317.COM - проверка увеличения файлов на 317 байт;

XSOMP.EXE - комплексная антивирусная проверка. Проверка проводится в текущем каталоге и его подкаталогах;

XCONTROLEX - поиск зараженных файлов в текущем каталоге. Проверка

проводится в текущем каталоге и его подкаталогах; XFIND.EXE - программа просматривает файлы, ищет в них вирус и сообщает о результатах работы. Проверка проводится в текущем каталоге и его подкаталогах;

XPREP.EXE - программа просматривает все файлы и "лечит" те из них, которые оказались заражены, с помощью встроенной в нее обезвреженной модели вируса. Проверка проводится в текущем каталоге и его подкаталогах;

KILL648.EXE - реставратор от вируса 648. Находит все файлы, зараженные вирусом, в текущем оглавлении. Обеззараживание таких файлов - автоматическое;

KILL1704.EXE - данная программа проверяет файлы на вирус длиной около 1700 байт, заражающий .COM-файлы и имеющий также резидентную часть;

TYPEOFF.EXE - тест на вирус "осыпание экрана";

LOOKCOM.EXE - Проверка COM-файлов на наличие вируса. Попытка обеззараживания производится автоматически;

TBOMB.EXE - поиск вируса срабатывающего "по таймеру";

SV.EXE - программа просматривает в текущем каталоге файлы и ищет в них вирус;

Программа: ШПРИЦ 1.0  
Автор: Аркадий Борковский  
Язык: Си  
Форма распространения: 1 дискета 5,25" DS/DD  
Документация: файл с описанием на дискете  
Цена: 600 руб. за 1 установку





## MKS\_VIR 2.85

Совершенно другого типа антивирусную программу предоставила нам польская фирма UNIKS. MKS.VIR проверяет указанный пользователем диск или несколько дисков на присутствие вирусов в записанных на нем файлах, независимо от расширения их имени. В дополнение к этому программа тестирует оперативную память ПЭВМ и boot-сектор данного диска (там также могут быть вирусы). Во избежание заражения MKS.VIR начинает работу с самопроверки.

Проверка может проводиться в двух режимах: на присутствие всех известных автору вирусов – в версии 2.85 их 147 (это долго и очень утомительно), и на присутствующие вирусы, обнаруженных в Польше (их 40).

**MKS.VIR 2.85** не только обнаруживает, но и обезвреживает 48 разновидностей вирусов. Для того чтобы пользователь мог получить самую необходимую информацию, автор программы — Марек Селль — “зашил” данные о нескольких, чаще всего встречающихся разновидностях вирусов, а также их проявлениях.

Программа предельно проста в употреблении. После загрузки пользователь

должен только выбрать режим проверки (все или только пойманные уже вирусы) и указать, какой диск следует проверить. Затем программа сканирует все содержимое оперативной памяти и указанного диска. В случае обнаружения вируса спрашивает, уничтожить ли его или нет.

Автор программы придумал оригинальный способ пополнения своей коллекции вирусов. Тот пользователь, который обнаружит ранее неизвестный автору вирус, может передать его по указанному в программе адресу и получить в награду новую версию программы, включающую средство обезвреживания новой мутации. Разработка новой версии *MKS.VIR*, по словам автора, занимает примерно 2 недели.

Программа распространяется в двух версиях: демонстрационной (обезвреживает только 14 старых мутаций вирусов) и лицензионной. Демонстрационная версия имеется в московском узле редакционной сети FIDO (см. "Компьютер", N 1) и ее можно получить с помощью модема.

Приведем список вирусов, обезвреживаемых программой **MKS\_VIR** (некоторые вирусы имеют по несколько названий, а большинство из них – несколько

разновидностей и мутаций):

1701, Cascade, Falling  
Tears, Herbs Leaves, Autumn  
Leaves, 1704, Blackjack, 507,  
534, 648, Vienna-A, DOS-62,  
BRAIN, Ashar, CHOINKA,  
DARK AVENGER, DEN-ZUK,  
EXE-VIR, Italian-A, Italian-B,  
PingPong, Bouncing Ball,  
JERUSALEM-B, Friday 13,  
Israeli, PLO, Surv 3.00,  
MUZYK, Oropax, STONED,  
Marijuana, New Zealand,  
VACSINA, YALE, Alameda,  
Meritt, Yankee Doodle, W13,  
AIDS.

**MKS\_VIR** обнаруживает (но пока не обезвреживает) следующие разновидности и мутации вирусов, известных автору по описаниям в мировой компьютерной печати:

100 Years, 17Y4, 1709, 17XX,  
2730, 2881, 2930, 3066,  
Traceback 3551, Macho, 3555,  
405, 4K, 640K, 867, AIDS  
Trojan, Agriplan Agiplan,  
Aids, Alabama, Amstrad,  
April 1st, Cascade FORMAT,  
Chaos DATACRIME 1168,  
DATACRIME 1280, DATACRIME  
II, DBASE December 24th,  
Devil Dance, Disk Killer, Do  
Nothing, Fu Manchu Ghost,  
Ghostballs virus, Golden Gate,  
HAHAHA, Haloechen,  
Icelandic Saratoga, Israeli  
Boot, PLO, sUMsDos, Joker,  
Lehigh, Lisbon 648 Machosoft,  
Mistake, Mix 1, Mixi, Mixi-2,  
New Zealand, Nichols Ohio,  
Palet, Pentagon, Perfume,  
PingPong/286, South African.

Stupid SYSLOCK, Saratoga,  
Icelandic, Spanish, Sunday,  
Surv, Swap, Sylvia Syslock,  
Taiwan, The OGRE,  
Traceback, Typo, Vienna "B"  
Vienna 62 C, Virus-90, Zero-  
Bug, Palette, Vcomm, virus-B.

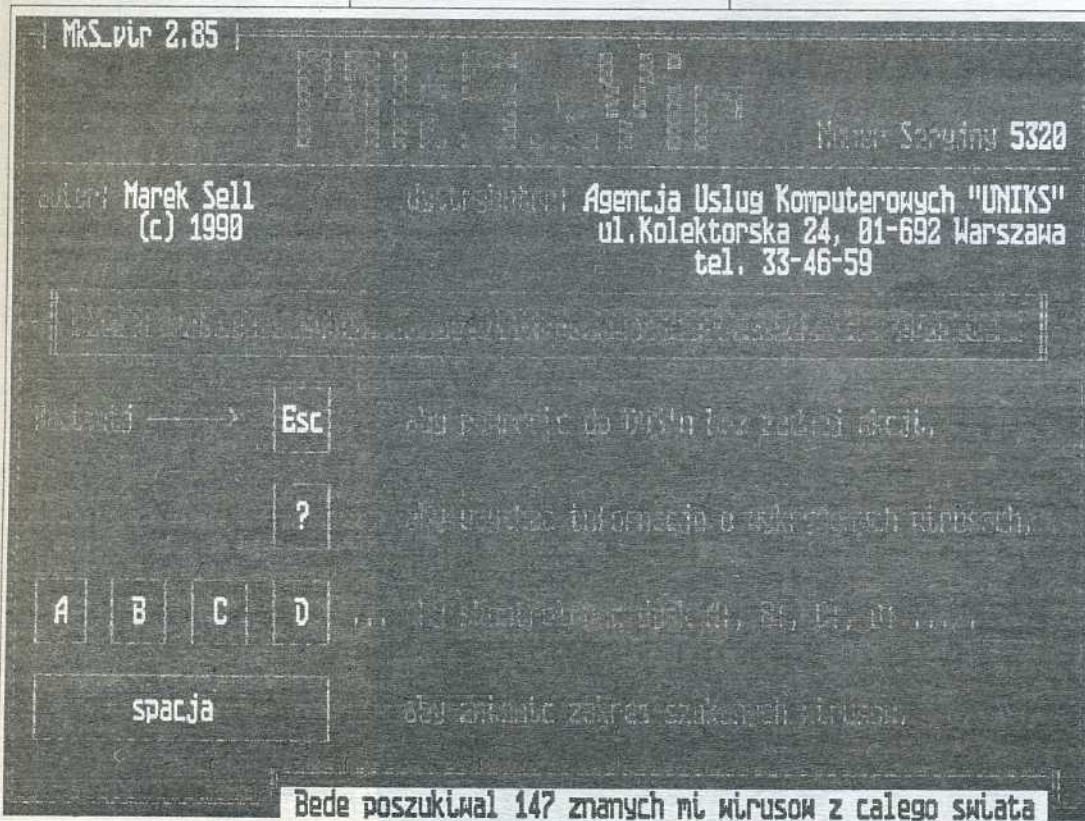
Программа: MKS\_VIR 2.85  
Автор: Marek Селль (Marek Sell)  
Язык: Ассемблер  
Форма распространения: 1 дискета 5,25" DS/DD  
Документация: брошюра, дополнительно файл с описанием на дискете  
Цена: 490.000 злотых  
© Игорь Егуминев

## АТІR – інструмент для моделювання

При проектировании и совершенствовании технологических цепочек, используемых на химических, пищевых, фармацевтических и обрабатывающих предприятиях, при проектировании и наладке технических систем, состоящих из взаимодействующих агрегатов, при согласовании экономических интересов отдельных хозяйственных подразделений, работающих в едином комплексе, возникает необходимость "проиграть" разнообразные варианты поведения объектов, зависящие от величин параметров и выбора характеристик. Здесь вам на помощь придет автоматизированная система проектирования и управления - АТР.

Первоначально АТТР предназначалась для расчета системы охлаждения на фармацевтическом производстве, но заложенные в нее принципы позволяют применять ее в тех областях, где составляющие объект отдельные агрегаты в процессе работы обмениваются между собой веществом, энергией и информацией.

Являясь в отличие от большинства систем автоматизированного проектирования и управления (CAD/CAM) открытой системой, инструментальной оболочкой, система *ATIR* может быть приспособлена к различным объектам. Если вы затрудняетесь наполнить предметную область системы, содержащую блоки для формирования моделей исследуемых объектов, разра-





ботчики системы придут вам на помощь. Пользуясь настроенной системой, вы сможете самостоятельно, не будучи профессионалом в области программирования и вычислительной математики, формировать модели и проводить на них расчеты. Благожелательный интерфейс упрощает освоение и использование системы, общение с которой происходит на русском или английском языке в форме приказов.

Система ATIR поставлена на ЭВМ серии СМ (PDP-11) в операционной среде RSX. Она допускает многотерминальную работу, обеспечивая хранение и защиту информации каждого пользователя. Для ее работы достаточно 64 Кбайт ОЗУ. Модули системы и информация пользователей сохраняются на диске, что обеспечивает достаточную быстроту обмена. Версия системы для персонального компьютера находится в процессе разработки.

Если вы заинтересовались описанными здесь программами, то по всем вопросам, связанным с получением дополнительной информации и их приобретением, обращайтесь в редакцию "Компьютера".

# Словари, словари...

Мне довелось оттестировать две программы, которые существенным образом облегчают подготовку текстов в повседневной редакционной работе, так сказать, в боевых условиях. Автором первой из них - КОРРЕКТОРА - является Аркадий Борисович Борковский. Ее распространением занимается московская фирма ТЕРМИНАЛ. Вторую - ОРФО - разработала группа по естественно-языковым системам московской фирмы ИНФОРМАТИК под руководством Олега Георгиевича Григорьева. Тест ОРФО проводился на версии 1.0. Готова уже версия 2.0, поэтому я приведу также некоторые сведения и о ней. Редакция "Компьютера" благодарна обеим фирмам за предоставленную возможность

ознакомиться с их программными продуктами и провести их испытания.

В работе каждой редакции самым трудоемким "технологическим" процес-

сом является многократная перепечатка одних и тех же текстов. Достаточно нескольких опечаток на странице, чтобы типография оказалась набирать текст.

Когда в нашей редакции появились микроЭВМ, трудоемкость этого процесса существенно снизилась. Машинописке не приходится больше перепечатывать исправленный редакторами или корректорами текст, теперь ей достаточно исправить ранее набранный текстовый файл, записанный на диске. Благодаря этому уменьшается число новых опечаток, которые непременно появляются, когда приходится переписывать весь текст. Но каждый напечатанный текст все равно

Комплект состоит из трех дискет, следующего содержания:  
1-ая (Установочная) 2-ая дискетта ...

ДИСКЕТТА

Имя файла дискетта

Обработано текста 5 %  
Исправлено 0

Просмотрено 39 слов  
Добавлено в словарь 0

Найдено 1  
Поправлено 0

Russian spelling checker

Russian spelling checker

Russian spelling checker

Russian spelling checker

Russian spelling checker

Russian spelling checker

Russian spelling checker

Контроль правописания

Контроль правописания

Контроль правописания

Контроль правописания

Контроль правописания

Контроль правописания

Контроль правописания

Научно-технический кооператив  
Т Е Р М И Н А Л представляет  
Русский Контроль Правописания  
КОРРЕКТОР 1.0

Контроль правописания

Контроль правописания

Контроль правописания

Контроль правописания

Контроль правописания

Контроль правописания

Контроль правописания

Контроль правописания

Контроль правописания

Russian spelling checker

Russian spelling checker

Russian spelling checker

Russian spelling checker

Russian spelling checker

Russian spelling checker

приходится передавать на вычитку корректору.

Перечисленные в начале этой статьи программы, в сущности, заменяют корректора. Их задача - "прочитать" текст, указать ошибки, предоставить возможность их исправить, запомнить новые слова и значения. Неотъемлемой частью обеих систем являются словари, состав которых пополняется по мере обработки очередных текстов. Новые слова программы сохраняют в отдельном файле, причем КОРРЕКТОР записывает в нем слова целиком, а ОРФО - только их корни.

## Корректор

Наиболее существенное различие между этими системами заключается в том, что КОРРЕКТОР проверяет весь текст, предоставляя



**ОРФО** \* Проверка текста прервана по Вашему требованию.  
Словари, словари ...

## Марек Цар

В повседневной редакционной работе, в - так сказать - боевых условиях мне пришлось протестировать две программы, которые существенным образом облегчили процесс обработки текстов. Автором первой из них - **КОРРЕКТОРА** - является Аркадий Борисович Борковский. Ее распространением занимается московская фирма **ТЕРМИНАЛ**. Вторую - **ОРФО** - разработала группа по естественно-языковым системам московской фирмы **ИНФОРМАТИК** под руководством Олега Георгиевича Григорьева. Тест **ОРФО** проводился на версии 1.0. К выпуску на рынок готова уже версия 2.0 и по ходу изложения будут даваться также некоторые сведения о ней. Редакция "Компьютера" благодарна обеим фирмам за предоставленную нам возможность ознакомиться и провести тест их программных продуктов.

В работе каждой редакции самым трудоемким "технологическим" процессом является многократная распечатка машинописками одних и тех же текстов. Достаточно нескольких опечаток на странице, чтобы типография отказалась принять текст к набору.

Когда в нашей редакции появились микроЭВМ - трудоемкость этого процесса существенно снизилась. Машинописки не

возможность исправлять его, а **ОРФО** может проверить только ту часть текста, которая выведена на экран дисплея. Для того чтобы исправить текст, вам придется воспользоваться текстовым процессором. Это недостаток **ОРФО**, особенно для пользователей, которые, как я, работают с уже готовыми файлами в настольных издательских системах. В версии 2.0 возможна будет также проверка файла.

Поскольку о программе **ОРФО** рассказать вкратце невозможно, познакомим читателей с **КОРРЕКТОРОМ**. Простота этого программного продукта столь заманчива, что, должен признаться, в своей повседневной работе я пользуюсь именно им.

Для того чтобы **КОРРЕКТОР** проверил русско-язычный текст (коды кириллицы должны соответствовать альтернативному варианту), достаточно написать имя программы, а после пробела - имя текстового файла, который будет проверяться. Остальную работу **КОРРЕКТОР** проводит сам. Если во время проверки текста программа обнаруживает незнакомое ей слово, то оно выводится на экран вместе с контекстом. Теперь пользователь дол-

жен решить, как поступить с данным словом: записать его в словарь, пропустить или исправить. В последнем случае слово появляется в специальном окне; его можно исправить, перемещая курсор к месту опечатки. После исправления

ошибки программа продолжает проверку текста. Выбор опции "Дополнить" приводит к автоматической записи незнакомого слова в специальный файл словаря пользователя. После команды "Пропустить" **КОРРЕКТОР** автома-

тически продолжает проверку текста. О том, насколько продвинулась работа, свидетельствуют данные, появляющиеся в углу экрана (процент обработанного текста, число проверенных слов). По завершении проверки пользователю предоставляется возможность сохранить на диске словарь с новыми словами и исправленный текст, которому автоматически присваивается расширение .DOC. Затем программа заканчивает свою работу. Для проверки очередного текста описанный процесс повторяется. Просто, быстро и надежно, не правда ли?

У **КОРРЕКТОРА** есть один, существенный с точки зрения пользователя, недостаток. Его внимание должно быть все время приковано к экрану монитора, а это весьма утомительно. Хотелось бы, чтобы при обнаружении в тексте незнакомого слова программа подавала звуковой сигнал. Тогда по ходу проверки можно бу-

**ОРФО**

НОВОЕ СЛОВО: **РЕКУРСИИ**

Образуйте нужную форму слова.

Примеры:

Вижу (одного, одну, одно, один)

кого, что? **РЕКУРСИИ**

сына, дом

Вижу (многих, многие)

кого, что? **РЕКУРСИИ**

синовей, дома

Нет (многих)

кого, чего? **РЕКУРСИИ**

синовей, домов

Рад (одному, одному)

кому, чему? **РЕКУРСИИ**

сыну, дому

Нет такой формы: ↓

Ввести: ←

Вернуться: Esc

Давайте на двух классических примерах посмотрим, что же представляет собой эта техника (в обоих случаях будем пользоваться языком Паскаль).

В свое время французский математик Ликас рассказывал следующую легенду:

дет отвлекаться от работы. Это важно. **КОРРЕКТОР** не в состоянии складывать и спрягать слова, не подскажет, где пропущена запятая, не отметит стилистических ошибок. Он имеет другое на-





значение: сравнить каждое слово со словарем, и если такого слова в нем нет - обратить на это внимание пользователя. И все. С ОРФО дело обстоит сложнее.

## ОРФО

ОРФО - резидентная, т.е. постоянно остающаяся в оперативной памяти машины программа. Она может работать с любым текстовым процессором, который использует текстовый (не графический) режим. Для того чтобы ею воспользоваться, следует сперва загрузить ее в память компьютера, а затем вызвать текстовый процессор и проверяемый текстовый файл. Поскольку в различных текстовых процессорах текст располагается в разных местах экрана, в самом начале работы приходится указывать программе рамки, в которых будет размещен текст. После этого, нажав соответствующую комбинацию клавиш, запускаем ОРФО. Вы можете наблюдать, как работает эта программа: курсор "считывает" текст на экране, отмечая по ходу дела незнакомые программе слова, несогласованность слов в предложениях (высвечивается все предложение) и неверные сочетания знаков препинания. Если у вас хорошая зрительная память, то вы можете исправлять ошибки с помощью своего текстового процессора. Если нет, то с помощью другой комбинации клавиш можно выделять незнакомые программе слова и перемещать к

ним курсор.

Поскольку ОРФО 1.0 работает только с текстом, который выведен на экран, его приходится все время перемещать (scrolling), что довольно утомительно, особенно при текстах с небольшим количеством ошибок.

Мой любимый текстовый процессор - PC-Write. Для того чтобы проверить текст с помощью ОРФО, я должен загрузить программу-корректор, затем процессор текстов и, наконец, текстовый файл. Потом нажатием комбинации клавиш отмечаю границы поля, в котором будет проверяться текст. Очередная комбинация клавиш - и ОРФО начинает проверку текста. Видны ошибки. Комбинация клавиш - и курсор находит незнакомое (ошибочное?) слово. Если в этом слове просто допущена опечатка, исправляем его, воспользовавшись текстовым редактором. Если все правильно, но словарь ОРФО не содержит этого слова, - вновь комбинация клавиш и можно приступить к подготовке нового слова для записи в словарь. Это самая необычная сторона работы. Программа начинает обсуждение указанного курсором слова. Прежде всего вам придется ответить, к какой части речи относится слово. Одна из немногочисленных возможностей выбирается по меню с помощью клавиш перемещения курсора и указания соответствующей данному слову части речи. Затем задаются еще 3-4 вопроса. Отвечая на них, пользователь указывает системе 3-4 формы нового слова. Какими именно будут вопро-

сы - зависит от проверяемого слова и ваших ответов. В худшем случае, если вы недостаточно хорошо знаете русский язык, отвечать на эти вопросы придется несколько раз, поскольку система отказывается принимать неправильные ответы. В самом сложном случае мне пришлось ответить на семь вопросов. К счастью, в течение всего обсуждения одной клавишей можно вызывать пояснения (HELP) и изучать необходимые сведения.

Прежде чем вы сможете ввести новые слова в словарь, ОРФО создаст на основании ваших ответов все формы незнакомого ей слова и выведет их для проверки. Если пользователь допустил ошибку, то придется повторить процедуру ответов на вопросы. Если все правильно, достаточно нажать ВВОД.

Если вам показалось, что все это сложно, то в этом моя вина. Я не сумел коротко и просто объяснить принципы работы программы. Ее большое преимущество перед КОРРЕКТОРОМ заключается в том, что после сеанса вопросов и ответов в словарь можно записать не одно, а сразу несколько, от 12 (для существительных) до 20 (для глаголов) слов.

Насколько просто и быстро можно работать с этими программами - оценят читатели. В заключение хотелось бы привести лишь некоторые временные показатели. На проверку этого текста КОРРЕКТОРОМ я потратил 7,5 минуты (от начала загрузки программы до выхода в DOS). За это время я исправил 14 ошибок, словарь был пополнен 128 словами, а исправленный текст можно было подвергать дальнейшей обработке. При этом я нажимал различные клавиши 270 раз. Длина текстового файла, подвергающегося проверке, - 10 342 байта. Количество проверенных программой слов - 1333.

На выполнение такой же работы в среде ОРФО мне понадобилось 10,5 минуты (от загрузки программы до записи исправленного текстового файла на диск). Было исправлено 14 ошибок, за 5 сеансов вопросов и ответов словарь был пополнен 58 словами (словарь ОРФО значительно больше сло-

варя КОРРЕКТОРА), устранены 2 стилистические ошибки. Нажимать различные клавиши и их комбинации мне пришлось 350 раз.

В ОРФО версия 2.0 - по утверждению авторов - скорость обработки текстов будет увеличена примерно в 1,5 раза. Кроме того, программа получит новые возможности:

- автоматизированная замена ошибочного слова на правильное;

- проверка на согласование слов с дефисом (например, "женщины-врач" будет подсвечено как ошибка), а также проверка правильности словообразования с некоторыми приставками (например, "поляблока" будет исправлено на "пол-яблока");

- режим автопроверки, в котором система подает звуковой сигнал, если было набрано ошибочное слово.

Программы подобного типа на Западе называют spell checkers. Они довольно распространены, но обычно входят в состав текстового процессора. В Польше существуют две такие разработки (TAG и QRTEKST), основанные на том же принципе единства с редактором текста. ОРФО выходит далеко за пределы обыкновенного spell checker типа КОРРЕКТОРА и может быть отнесена к интеллектуальным программам обработки текстов.

Идея, которая лежит в основе разработок советских программистов, очень привлекательна, особенно для тех, кто профессионально занимается настольными издательскими системами. Идеальный продукт должен сочетать простоту КОРРЕКТОРА с лингвистическим совершенством ОРФО и, помимо этого, предусматривать возможность расстановки по всему тексту знаков переноса с кодом, который выберет пользователь. Разработчик такой программы может рассчитывать на ее сбыт везде, где с компьютерами общаются на русском языке.

Программа:  
КОРРЕКТОР 1.0  
Автор:  
АРКАДИЙ  
БОРКОВСКИЙ  
Язык: Си  
Форма распространения:  
3 дискеты  
5,25" DS/DD  
Документация:  
файл с описанием  
на дискете  
Цена:  
1400 руб. за 1  
установку

Программа:  
ОРФО 1.0  
Автор:  
группа под руководством  
ОЛЕГА ГРИГОРЬЕВА  
Язык: Си, Ассемблер  
Форма распространения:  
3 дискеты  
5,25" DS/DD  
Документация:  
брошюра  
Цена:  
1100 руб. за 1  
установку



\компьютер на работе\

© Виктор Писаревский

# Norton Utilities: NI и BE

Несмотря на то что операционная система DOS содержит очень большое количество команд (MS-DOS - около пятидесяти, DR DOS - на шесть больше), в ней нет средств для выполнения многих полезных функций по обслуживанию дисков, файлов, каталогов и т.д. Поэтому различные фирмы выпускают многочисленные комплексы программ для выполнения операций, которые затруднительно или невозможно выполнить с помощью команд DOS. Мы расскажем об одном из наиболее широко используемых комплексов вспомогательных программ - **Norton Utilities** (Advanced Edition 4.80, (C) Copr 1987-88, Peter Norton).

## Состав комплекса

Norton Utilities включает более двадцати программ, выполняющих разнообразные функции по работе с файлами, каталогами, дисками, а также программу NI (Norton Integrator), выдающую справки о формате вызова программ из Norton Utilities и позволяющую выполнять эти программы.

В Norton Utilities входят следующие программы.

**Программы для работы с файлами:** FA - просмотр и установка атрибутов; FD - установка даты и времени файлов; FF - поиск файлов на диске; FI - установка и вывод комментариев; FS - вывод информации о размере отдельных файлов и их групп; LP - печать с форматированием; TS - поиск заданного текста на диске и в файлах; Wipefile - удаление отдельных файлов и групп файлов с возможностью уничтожения их информации (для обеспечения секретности).

**Программы для работы с каталогами:** LD - вывод списка каталогов; NCD - наглядный переход из каталога в каталог.

**Восстановление удаленных файлов и каталогов:** QU - восстановление удаленных файлов; UD - восстановление удаленных каталогов; NU (режим Unerase) - автоматическое и ручное восстановление файлов и каталогов.

**Программы для работы с дисками:** DI - получение информации о диске; DT - проверка диска на наличие физических дефектов и перемещение информации из сбойных участков в безопасное место; FR - восстановление диска после случайного форматирования и сохранение информации для восстановления диска после случайного форматирования; NDD - проверка и исправление дисков при наличии физических и логических дефектов; NU (режим Explore Disk) - редактирование информации на дисках; SD - оптимизация размещения файлов на диске и ускорение работы с диском; SF - форматирование диска с сохранением информации; TS - поиск заданного текста на диске и в файлах; VL - установка метки диска; Wipedisk - уничтожение информации на диске.

**Информация о компьютере и установка параметров устройств:** SI - получение информации о компьютере; NCC - установка параметров устройств (формы курсора, режим монитора и т.д.); TM - вывод текущего и использованного времени.

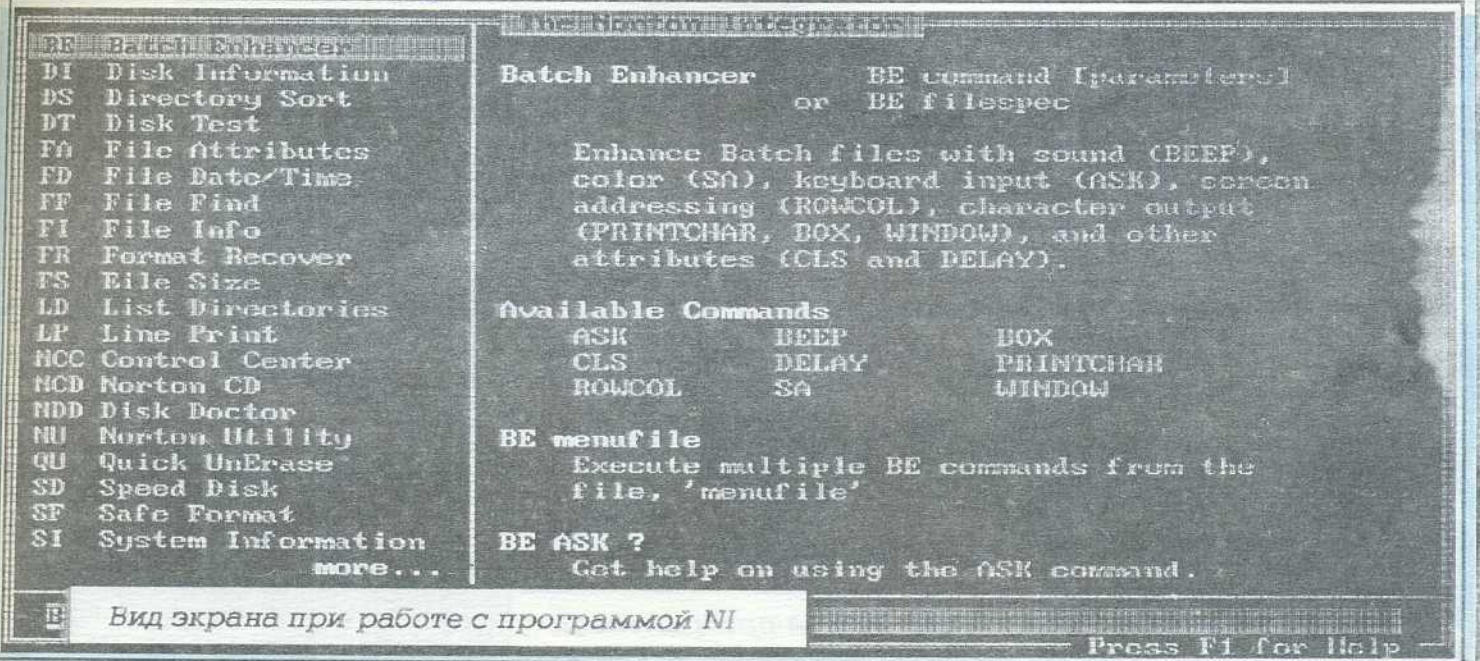
**Улучшение командных файлов:** программа BE.

## Norton Integrator

Norton Integrator (NI) - это программа-оболочка, позволяющая выводить на экран справки о формате вызова программ из комплекса Norton Utilities и выполнять эти программы. Если вы забыли, как называется нужная программа из этого комплекса или какие параметры нужно задать, чтобы выполнить какую-либо ее функцию, можно ввести команду NI и посмотреть необходимые сведения. После этого можно тут же, не выходя из программы NI, выполнить любую программу из Norton Utilities.

После ввода команды NI на экран (в левой его части) выводится список программ комплекса Norton Utilities. С помощью клавиш [↑] или [↓] можно выделить курсором (подсветкой) нужную программу, в правой части экрана выводится справка о выделенной программе.

В нижней части экрана находится командная строка, в которой можно набрать нужную команду, нажимая на соответствующие буквенно-цифровые клавиши. Для выполнения набранной команды следует нажать клавишу [Enter].



Вид экрана при работе с программой NI





С помощью клавиши [Tab] можно очистить командную строку и перейти в режим быстрого поиска. При этом можно выделить имя нужной программы из Norton Utilities, набрав первые буквы ее имени. Для выхода из режима быстрого поиска следует нажать клавишу "пробел".

Выйти из программы NI можно, нажав [Esc] или [F10].

## Общие сведения о программах комплекса Norton Utilities

Поскольку программы из Norton Utilities разрабатывались одной группой авторов, то они имеют некоторые общие свойства.

**Получение справки.** Для получения справки о назначении и режимах работы любой программы из Norton Utilities следует ввести команду: имя-программы ?, например lp ?. Для программы BE при этом выводится список режимов, а для получения справки по режиму работы этой программы надо ввести команду BE режим ?, например BE ASK ?.

**Работа с меню.** Многие программы из комплекса Norton Utilities управляются с помощью меню. Во всех этих программах работа с меню осуществляется приблизительно одинаково:

для перемещения по пунктам меню используются клавиши перемещения курсора [↑], [↓], [←] и [→];

для выбора нужного пункта меню надо выделить этот пункт и нажать клавишу [Enter];

можно выбрать нужный пункт меню, нажав на клавишу, соответствующую букве, выделенной в пункте меню желтым цветом (на монохромных мониторах - повышенной яркостью);

для выхода из любого уровня меню на предыдущий без изменений установленных режимов надо нажать клавишу [Esc].

В этом номере сборника мы расскажем о программе BE из комплекса Norton Utilities. В следующих выпусках будут описаны и другие программы.

\* \* \*

Стандартные команды DOS для организации командных файлов не дают возможности выполнять многие полезные действия, например запросы действий пользователя, меню и т.д. Поэтому различными фирмами были разработаны программы, расширяющие возможности командных файлов DOS. Предлагаем вашему вниманию одну из них - программу **BE** (Batch Enhancer) из комплекса Norton Utilities.

## Возможности программы BE

Программа BE позволяет выполнять следующие действия (в скобках указаны соответствующие режимы):

предпринимать в командном файле различные действия в зависимости от ответа пользователя (ask);

перемещать курсор в нужное место экрана (rowcol);

выводить символы на экран (printchar);

издавать звуковые сигналы (beep);

рисовать на экране прямоугольники и окна (box, window);

делать паузы при выполнении командного файла (delay);

устанавливать цвета различных частей экрана (va);

очищать экран (cls).

**Формат команды:** BE режим [параметры] или BE имя-файла. Возможные значения режимов описаны выше. Если программа BE вызвана с указанием имени файла, то этот файл должен быть текстовым и каждая его строка должна иметь вид: режим [параметры].

Для получения справки о имеющихся режимах программы BE надо ввести команду BE ?, об одном из режимов - BE режим ?.

## Задание цвета символов

Для многих функций программы BE имеется возможность задавать цвета символов и фона выводимых на экран сообщений. Задание цвета имеет следующий вид: [BOLD] цвет-символа ON цвет-фона. Здесь BOLD указывает, что цвет символа должен быть ярким. Например, blue - синий цвет, bold blue - ярко-синий. Цвет символа и фон могут задаваться как: white (белый), black (черный), blue (синий), magenta (малиновый), cyan (сиреневый), red (красный), yellow (коричневый), green (зеленый). Желтый цвет указывается как bold yellow. Например: bold yellow on blue - желтые символы на синем фоне.

Задание цвета в команде BE не обязательно. Если цвет сообщений не указан, то выводимые символы будут иметь те же цвета символов и тот же фон, что и символы, которые были ранее в том же месте экрана.

## Выполнение действий в зависимости от ответа пользователя

Иногда в командном файле нужно выполнить различные действия по выбору пользователя. Это можно сделать с помощью функции ASK программы BE. Формат вызова: BE ASK "сообщение" список-символов [параметры].





Программа BE выводит сообщение и ждет, пока пользователь не введет один из указанных в списке символов. Значение переменной ERRORLEVEL, которое можно проверить в следующих строках командного файла с помощью команды IF, устанавливается равным номеру введенного символа в списке. Следует заметить, что проверки значений переменной ERRORLEVEL надо располагать в порядке убывания значений.

Пример:

```
BE ASK "Запустить программу ALFA [Y/N]?" YN
```

При ответе N значение переменной ERRORLEVEL устанавливается равным 2, при ответе Y - равным 1. Чтобы программа ALFA запускалась только при ответе "Y", продолжение командного файла должно быть следующим:

```
if ERRORLEVEL 2 goto continue
```

```
ALFA rga
```

```
:continue
```

При вызове функции ASK программы BE можно указывать и такие параметры:

DEFAULT=символ - пользователю разрешается отвечать на выводимый запрос нажатием клавиши [Enter], что эквивалентно нажатию на клавишу с указанным в параметре DEFAULT символом. Этот символ должен содержаться в списке символов, указывающем допустимые ответы на выведенный программой BE запрос;

TIMEOUT=число - программа BE ожидает ответа на запрос в не больше, чем заданное число секунд. Если пользователь за это время не ответил на запрос, считается, что он нажал на клавишу, указанную в параметре DEFAULT. Если же параметр DEFAULT указан не был, то переменной ERRORLEVEL присваивается наибольшее возможное число. Проверку на то, что пользователь не ответил на запрос, можно выполнить с помощью команды if errorlevel N действие, где N равно числу символов в списке допустимых символов плюс один (или плюс значение параметра ADJUST, если он указан);

ADJUST=число - значение переменной ERRORLEVEL устанавливается равным не номеру введенного символа в списке, а этому номеру плюс значение параметра ADJUST минус единица. Иначе говоря, можно считать, что значение переменной ERRORLEVEL равно номеру введенного символа в списке, а нумерация символов начинается не с единицы, а с числа, указанного в параметре ADJUST;

Цвет - цвет символов и фон в выводимом сообщении.

Например, при вводе команды: BE ask "Продолжить работу? (Y/N) ", YN, DEFAULT=Y, TIMEOUT=180 после вывода со-

общения программа BE ожидает не более 180 секунд. Если пользователь в течение этого времени не ответит, то подразумевается ответ "Y". При нажатии клавиши [Enter] так же подразумевается ответ "Y".

### Выдача звуковых сигналов

Функция BEEP программы BE служит для выдачи звуковых сигналов.

Формат команды: BE BEEP [параметры] или BE BEEP имя-файла. Здесь слово параметры означает: /Dn - издать звук длительностью n/18 с; /Fn - издать звук с частотой n Гц; /Rn - повторить звук n раз; /Wn - пауза между звуками длительностью n/18 с. Например, be beep /d9 /f400 /r2 /w3 - издать два раза звук частотой 400 Гц (звук соль основной октавы) длительностью по 0,5 с, с паузой между звуками, равной 1/6 с.

При вводе команды BE BEEP имя-файла с помощью программы BE можно получить более сложные мелодии. Имя-файла должно указывать на обычный текстовый файл, в котором содержатся команды. Возможные команды: Dn, Fn, Rn и Wn; они аналогичны соответствующим параметрам, задаваемым в командной строке. Каждая команда занимает одну строку файла.

### Рисование рамки на экране

С помощью функции BOX программы BE можно нарисовать на экране двойную рамку заданного цвета. Формат команды:

BE BOX верхний-ряд левая-колонка нижний-ряд правая-колонка [цвет]

Здесь при указании границ рамки ряды и колонки экрана следует нумеровать, начиная с нуля (верхний левый угол экрана имеет координаты 0, 0). Такая же нумерация используется и в остальных функциях программы BE. Параметр цвет задает цвет символов и фона рамки.

### Вывод окна

С помощью функции WINDOW программы BE можно нарисовать на экране окно - ограниченный двойной рамкой прямоугольник заданного цвета. Эта функция похожа на функцию BOX, она имеет лишь два отличия:

при задании функции WINDOW внутренность выведенного прямоугольника закрашивается указанным в команде цветом;

в команде можно указывать параметры EXPLODE - вывести окно на экран не сразу, а "расширять" его из центра; SHADOW - справа и снизу от окна выводить его "тень".



Вид экрана при загрузке программы BE





## Установка курсора или вывод текста в нужном месте экрана

Функция ROWCOL программы BE позволяет устанавливать курсор и выводить сообщение в нужном месте экрана. Формат команды:

BE ROWCOL строка столбец ["текст"] [цвет].

Параметры: строка - указывает номер строки, в которую переводится курсор; столбец - указывает номер столбца, в который переводится курсор; текст - указывает, какой текст должен быть выведен в том месте, куда будет установлен курсор (текст должен быть заключен в кавычки, если текст не указан, то программа BE только устанавливает курсор); цвет - задает цвет символов и фон выводимого текста. Например, BE rowcol 2 3 "НУ, ПОГОДИ" red on blue - вывести надпись "НУ, ПОГОДИ" красным цветом на синем фоне, начиная с третьей позиции второй строки.

## Вывод последовательности символов на экран

Функция Printchar программы BE позволяет вывести один символ или последовательность одинаковых символов на экран. Символы выводятся, начиная с текущей позиции курсора.

Формат команды: BE PRINTCHAR символ число [цвет].  
Параметры: символ - указывает, какой символ выводится на экран; число - указывает число выводимых символов; цвет - задает цвет символов и фон при выводе символов. Например, BE printchar \* 10 bold yellow on blue - вывести 10 символов "\*" желтым цветом на синем фоне.

## Установка атрибутов символов на экране

С помощью функции SA программы BE можно установить атрибуты символов на экране для монохромных и цветных мониторов.

Формат команды для цветных мониторов: BE SA цвет /N /C.

Параметры: цвет - задает цвет символов и фон, формат задания цвета указан выше; /N - указывает, что не надо устанавливать цвет рамки экрана; /C - очистить экран.

Формат команды для монохромных мониторов: BE SA тип-символов /N. Параметры: тип-символов - указывает, должны ли быть символы обыкновенными, инверсными или подчеркнутыми (Normal - обыкновенные символы, Reverse - инверсные, т.е. темные символы на светлом фоне, Underline - подчеркнутые); /N - указывает, что не надо устанавливать цвет рамки экрана.

Например, BE SA black on cyan /C - очистить экран и установить черные символы на сиреневом фоне.

## Очистка экрана

Формат команды: BE CLS.

## Пауза при выполнении командного файла

Функция DELAY программы BE позволяет сделать паузу при выполнении командного файла. Это может быть полезно, например, для того, чтобы пользователь мог успеть прочесть сообщения, выведенные на экран.

Формат команды: BE DELAY число. Здесь число определяет длительность паузы. Единица измерения - 1/18 с. Например, BE DELAY 90 - сделать паузу продолжительностью 5 секунд.

## Пример использования программы BE

Рассмотрим пример диалогового командного файла, позволяющего выбрать одну из трех игр: DIGGER, TETRIS и CAT.

```
@echo off
```

```
echo Выберите желаемую игру:
```

```
echo D - DIGGER
```

```
echo T - TETRIS
```

```
echo C - CAT
```

```
be ask "Введите D, T или C : ", DTC
```

```
if errorlevel 3 goto cat
```

```
if errorlevel 2 goto tetris
```

```
digger
```

```
goto exit
```

```
:tetris
```

```
tetris
```

```
goto exit
```

```
:cat
```

```
cat
```

```
:exit
```

Выполнение этого командного файла выглядит не очень красиво. С помощью программы BE его можно значительно улучшить:

```
@echo off
```

```
be window 0 0 24 79 bold white on blue
```

```
be rowcol 5 30 "Выберите желаемую игру:" bold yellow on blue
```

```
be rowcol 6 30
```

```
be printchar = 23 bold yellow on blue
```

```
be rowcol 8 30 "D - Digger" bold yellow on blue
```

```
be rowcol 10 30 "T - Tetris" bold yellow on blue
```

```
be rowcol 12 30 "C - Cat" bold yellow on blue
```

```
be rowcol 14 30
```

```
be ask "Введите D, T или C : ", DTC, bold white on blue
```

```
if errorlevel 3 goto cat
```

```
if errorlevel 2 goto tetris
```

```
digger
```

```
goto exit
```

```
:tetris
```

```
tetris
```

```
goto exit
```

```
:cat
```

```
cat
```

```
:exit
```

Попробуйте самостоятельно разобраться в том, что делает данный командный файл.

Описанный командный файл имеет один недостаток - он выполняется достаточно медленно. Это связано с тем, что в нем девять раз вызывается программа BE, а каждый вызов требует времени. Издержки, связанные с многократным вызовом программы BE, можно устранить, заменив строки 2-10 командного файла на строку be game.dat, где файл game.dat содержит следующие строки:

```
window 0 0 24 79 bold white on blue
```

```
rowcol 5 30 "Выберите желаемую игру:" bold yellow on blue
```

```
rowcol 6 30
```

```
printchar = 23 bold yellow on blue
```

```
rowcol 8 30 "D - Digger" bold yellow on blue
```

```
rowcol 10 30 "T - Tetris" bold yellow on blue
```

```
rowcol 12 30 "C - Cat" bold yellow on blue
```

```
rowcol 14 30
```

```
ask "Введите D, T или C : ", DTC, bold white on blue
```

Попробуйте использовать программу BE, и вы сделаете ваши командные файлы гораздо красивее!



\компьютер на работе\

© Ян Стожек

# SHEZ

Недавно благодаря узлу сети FIDO, работающему в варшавской редакции журнала "Komputer", я познакомился с интересной программой SHEZ, разработанной Джимом Дерром (Jim Derr). Она представляет собой чрезвычайно богатую оболочку (shell) для всех используемых у нас программ упаковки и распаковки данных.

Работу надо начинать с конфигурирования программы. Для этого существует файл SHEZCFG, позволяющий выбрать используемые программы упаковки/распаковки, редактор, программу для просмотра двоичных или текстовых файлов (я предпочитаю короткую, но богатую программу LIST.COM Вернона Д. Берга), рабочий диск, тип дисплея (монокромный, цветной, черно-белый) и т.д. Можно также определить собственный шаблон \$\$, указывая типы файлов. Я работаю с файлами типов .ARC, .ZIP и .LZH и каждый раз, когда вызываю шаблон \$\$, вызываются файлы с указанными расширениями.

Итак, программа сконфигурирована. Теперь надо ввести SHEZ, нажать ENTER и можно работать. Прежде всего в глаза бросаются три окна. Самое большое, расположенное слева, занимает почти всю высоту экрана; оно названо "FILE LIST" и содержит список файлов и подкаталогов (как Norton Commander). В него выведены только те файлы, которые отвечают шаблону, а сам шаблон вместе с путем доступа виднеется у верхней кромки окна. Внизу находится сообщение "ESC-EXIT". И действительно, нажатие «ESC» вызывает текущий каталог, а если нажать эту клавишу несколько раз, то можно добраться до корневого каталога. Очередное нажатие «ESC» - и программа просит подтвердить желание закончить работу. Любая клавиша, кроме «Y» (yes), позволяет продолжить просмотр программы. Во вто-

ром окне находится информация об авторе, а в третьем, ниже, - сообщения о среде работы: текущий диск, путь доступа, каталог, в котором будут размещены распакованные файлы, объем свободной памяти рабочего диска, максимальное количество файлов из архива и максимальное количество файлов, выводимое в окне списка файлов (установленное при конфигурировании). В самом низу вы увидите знакомый элемент - список значений функциональных клавиш. Клавиша F1 традиционно вызывает окно с подсказкой (HELP). Кроме функциональных клавиш в строке значений указана клавиша «/», описанная как "MENU". Интересным свойством этой строки является то, что она изменяется при нажатии «SHIFT», «CTRL» и «ALT», указывая всякий раз новые значения функциональных и буквенных клавиш. И еще обратите внимание на верхнюю строку экрана: в ней расположено название и номер версии программы, три слова объяснения ("The Compression Companion") и... дата с днем недели и часами.

Разглядывая экран, можно начать экспериментировать. Строка в верхней части окна со списком файлов выделена. Обычно там находятся две точки - указание текущего каталога. Нажимаем «ENTER» - и, действительно, мы попадаем в текущий каталог. С помощью клавиш курсора передвигаемся вверх и вниз по каталогу, а воспользовавшись клавишами «HOME» и «END» можно высветить первую и последнюю строку окна (не списка!). «PAGE UP» и «PAGE DOWN» также действуют в соответствии с логикой. Итак, можно указать нужный файл и нажать «ENTER». Изображение меняется: на экране появляется большое окно, а в нем содержимое упакованного файла с указанием величины до и после упаковки, даты создания и метода сжатия каждого файла. Первое название выделено. Еще раз нажимаем «ENTER», вызывая программу распаковки, которая некоторое время работает... и затем мы можем просмотреть данный файл с помощью указанной при конфигурировании программы. Выходим из режима просмотра (я для этого нажимаю «ESC») и возвращаемся в SHEZ. После нажатия клавиши пробела с левой стороны имени файла появляется характерная "птичка" - файл обозначен. Большинство операций можно производить на обозначенных файлах или на выделенном файле. Благодаря тому, что эти операции выполняются с помощью различных комбинаций клавиш, в программе SHEZ (иначе чем в Norton Commander) не возникает затруднений, когда обозначаешь файлы для копирования и вдруг хочешь что-либо стереть... И здесь мы дошли до главного назначения программы: управление программами-упаковщиками.

SHEZ 5.1—The Compression Companion—Saturday, July 21, 1990

00:44:28 AM

[FILE LIST]

&gt;C:\S.S\$

<DIR>  
<FIC>  
<DOS>  
<EDYTORY>  
<GRY>  
<NIC>  
<NORTON>  
<PAINT>  
<PRINTM3>  
<PZT>  
<STANT>  
<TERSTY>  
<TYPESET>  
<UTILE>  
<UNTURA>  
<WINDOWS>  
<WORD>

[ESC-EXIT]

(C) Copyright 1989, Jim Derr

[CURRENT SHEZ ENVIRONMENT]

Current Disk and Directory is:

C:\ARC\SHEZ

Extracted files will be put in:

C:\ARC\SHEZ

AVAILABLE SPACE ON C: = 1,660,928

Compressed Files Limit = 200

Wildcard Files Limit = 200

[CSERVE ID 76266,26341]

Вид экрана при загрузке программы SHEZ





После нажатия клавиши «/» вверху экрана появляется строка меню. Нажатие «ENTER» открывает окно с соответствующими опциями. Рядом с каждой из них отмечена комбинация клавиш, которая ее выполняет. Все опции доступны на уровне меню благодаря комбинациям «SHIFT», «CTRL» и «ALT» с разными клавишами. Можно распаковать указанный файл, обозначенные файлы, названные файлы или все файлы в указанный каталог. Можно добавить названные или указанные в списке файлы. Можно устранить указанный файл или обозначенные файлы из архива, стереть в указанном каталоге файлы, имеющие копии в данном архиве. Можно проверить внутренний порядок архива, изменить время создания архива и даже отредактировать файл из архива! Да, да, SHEZ позаботится о том, чтобы нужный файл был распакован, вызовет редактор, а после введения поправок снова упакует файл. Можно также запустить упакованную программу и даже проверить, нет ли в ней вирусов (для этого используется программа VIRUSCAN, продукт MCAFFEE ASSOCIATES, в последней версии содержащая ловушки для более чем 50 вирусов). Есть возможность распечатать содержимое упакованного файла на принтере или переслать его через последователь-

ный порт. Можно, наконец, преобразовать один формат упаковки в другой. Автор не забыл и о том, что оболочка — это еще не все, и что иногда надо на минутку вернуться к DOS. И снова интересная особенность: если вся программа занимает почти 100 Кбайт, то после вызова COMMAND.COM в памяти остается всего 10 Кбайт. Ко всему сказанному можно еще добавить целое множество вспомогательных операций, таких, как изменение пути доступа и очередности выведения файлов на экран, превращение самораспаковывающихся файлов в обычные и даже возможности копирования и устранения обычных файлов.

Другими словами, это очень хорошая программа — несмотря на некоторые мелкие недостатки (например, просмотр файла на разных уровнях вызывает разные команды, причем без какого-либо обоснования). Программа распространяется по типу "shareware" (скопируйте, испробуйте, если понравится — регистрационная оплата составляет только 20 долларов). Она доступна в редакционном узле сети FIDO. Узел работает ежедневно с 24.00 до 7.00 утра.

Перевод Анджея Пошлавского

```

[-----]
[SHEZ CONFIG MENU]
[-----]
Color setup
  Default color set
  Custom color set
  Black & White color set
  Monochrome set
Programs setup
Additional setup information
Save config file
Integrity check of shez program
[Press Option letter or ESC=EXIT]
  
```

Вид экрана во время конфигурирования программы SHEZ

| Extract                  |      |       |          |       |            |       |        |      |      |
|--------------------------|------|-------|----------|-------|------------|-------|--------|------|------|
| C:\PRINTMAS\PRINTMAS.LZH |      |       |          |       |            |       |        |      |      |
| NAME                     | SIZE | DATE  | TIME     | OWNER | PERMISSION | FILE  | SIZE   | DATE | TIME |
| BORDER                   | MTX  | 100   | 01/07/90 | 20:05 |            | 256   | SHRUNK |      |      |
| BORDERS                  | SHP  | 7432  | 01/07/90 | 20:05 |            | 20608 | SHRUNK |      |      |
| COLOR                    | COM  | 846   | 01/07/90 | 20:05 |            | 1000  | SHRUNK |      |      |
| COMMAND                  | COM  | 12018 | 01/07/90 | 20:05 |            | 17664 | SHRUNK |      |      |
| COMPUTER                 | FNT  | 1716  | 01/07/90 | 20:05 |            | 7424  | SHRUNK |      |      |
| DEVILLE                  | FNT  | 4785  | 01/07/90 | 20:05 |            | 15360 | SHRUNK |      |      |
| EDITOR                   | FNT  | 780   | 01/07/90 | 20:05 |            | 1408  | SHRUNK |      |      |
| FBPNUM                   |      | 13    | 01/07/90 | 20:05 |            | 128   | SHRUNK |      |      |
| FNTNAMES                 | MTX  | 87    | 01/07/90 | 20:05 |            | 256   | SHRUNK |      |      |
| FONTICON                 | SHP  | 1765  | 01/07/90 | 20:05 |            | 4096  | SHRUNK |      |      |
| FSIZES                   |      | 640   | 01/07/90 | 20:05 |            | 896   | SHRUNK |      |      |
| GEDIT                    | EXE  | 13279 | 01/07/90 | 20:05 |            | 25728 | SHRUNK |      |      |
| GRSIZE                   | MTX  | 31    | 01/07/90 | 20:05 |            | 128   | SHRUNK |      |      |
| HAMPTON                  | FNT  | 3060  | 01/07/90 | 20:05 |            | 6912  | SHRUNK |      |      |
| HGCFULL                  | EXE  | 56    | 01/07/90 | 20:05 |            | 640   | SHRUNK |      |      |
| KSW                      | CAR  | 543   | 01/07/90 | 20:05 |            | 3032  | SHRUNK |      |      |
| MAIN                     | MTX  | 102   | 01/07/90 | 20:05 |            | 256   | SHRUNK |      |      |
| M                        |      |       |          |       |            | 264   | SHRUNK |      |      |
| O                        |      |       |          |       |            | 168   | SHRUNK |      |      |

Вид экрана при работе с программой SHEZ



# Компьютер на работе

© Виктор Фигурнов

## Русифицируем MS WORD

Как известно, Microsoft Word - один из наиболее популярных редакторов текстов в СССР. Поскольку он создавался в расчете на пользователей в США и Западной Европе, то некоторые его функции при работе с текстами на русском языке работают неправильно. Для функций Word, связанных с проверкой правописания и словарем синонимов, изменить положение довольно сложно. Однако другие функции Word могут быть относительно легко исправлены. Например, изображение русских букв в графическом режиме экрана устанавливается с помощью драйвера экрана, переносы в русских текстах делаются с помощью специальных программ, считывающих документ Word и выдающих тот же файл с переносами и т.д.

Очень много неприятностей доставляют дефекты Word, связанные с неправильной заменой прописных русских букв на строчные и обратно, а также с неверной сортировкой русско-язычных текстов. Из-за них Word неправильно работает (для русско-язычных текстов):

- \* при поиске и замене строк с русскими буквами в режиме Case-No (когда поиск и замена не зависят от регистра букв);
- \* при выполнении команды Library Index (построение предметных указателей);
- \* в режиме Small Caps (капиталь) команды оформления символов Format Character.

Следует сказать, что эти дефекты во многих случаях сильно увеличивают объем ручной работы. Так, поиск и замену строк с русскими буквами приходится выполнять по два и более раза - для каждой возможной комбинации прописных и строчных букв отдельно. А выполнив 3-4 раза ручную сортировку по алфавиту предметного указателя, начинаешь видеть этот процесс ночью в кошмарных снах.

К счастью, с указанными дефектами легко справиться, и в настоящей статье предлагается способ их исправления. При его описании мы будем считать, что русские буквы на компьютере кодируются в соответствии с альтернативной кодировкой ГОСТа и что используемая версия Microsoft Word - 5.0. Поскольку большинство пользователей Word не являются программистами, предлагаемый вариант коррекции не требует написания программ, а основан на непосредственном внесении изменений в исполнимый файл WORD.EXE. Программисты, желающие оставить файл WORD.EXE в неприкосновенности, могут легко разработать программу, которая вносит указанные исправления в образ файла WORD.EXE в оперативной памяти при выполнении Microsoft Word.

Перед тем как вносить исправления в файл WORD.EXE, сделайте его копию, так как при этой работе имеется определенный риск ошибки. Для выполнения исправлений можно воспользоваться любой программой для корректировки двоичных файлов, например программой NU - Norton Utility. Формат вызова: NU WORD.EXE (предварительно следует перейти в тот каталог, в котором находится файл WORD.EXE). Следует заметить, что предлагаемый способ годится как для файла WORD.EXE с дистрибутивной дискеты, так и для файла WORD.EXE, который получается при установке Microsoft Word на жесткий диск (эти два файла имеют разные размеры).

Для устранения указанных выше дефектов необходимо внести исправления в четыре таблицы Microsoft Word: таблицу классов символов с кодами 128-255, таблицы замены прописных букв с кодами 128-255 на строчные и наоборот, и таблицу весов символов для сортировки.

Таблица классов символов с кодами 128-255 расположена в файле WORD.EXE в позициях 32AE6-32B65 (позиции исчисляются с нуля и приводятся в шестнадцатеричной системе счисления). Каждому символу в этой таблице соответствует код: 00 - не буква, 01 - строчная буква, 02 - прописная буква. Поэтому данную таблицу надо исправить следующим образом:

- с 32AE6 по 32B05 - байты с кодом 02;
- с 32B06 по 32B15 - байты с кодом 01;
- с 32B16 по 32B45 - байты с кодом 00;
- с 32B46 по 32B55 - байты с кодом 01;
- 32B56 - байт с кодом 02;
- 32B57 - байт с кодом 01;
- с 32B58 по 32B65 - байты с кодом 00.

Таблица замены прописных букв на строчные располагается в позициях 32B66 - 32BE3. Эту таблицу надо исправить так:

- с 32B66 по 32B85 - буквы а-я (кроме ё) в алфавитном порядке;
- с 32B86 по 32B95 - буквы а-п (кроме ё) в алфавитном порядке;
- с 32B96 по 32BC5 - без изменений;
- с 32BC6 по 32BD5 - буквы р-я в алфавитном порядке;
- с 32BD6 по 32BD7 - две буквы ё;
- с 32BD8 по 32BE3 - без изменений.

Таблица замены строчных букв на прописные располагается в позициях 32BE6 - 32C65. Эту таблицу надо исправить так:

- с 32BE6 по 32C05 - буквы А-Я (кроме Ё) в алфавитном порядке;
- с 32C06 по 32C15 - буквы А-П (кроме Ё) в алфавитном порядке;
- с 32C16 по 32C45 - без изменений;
- с 32C46 по 32C55 - буквы Р-Я в алфавитном порядке;
- с 32C56 по 32C57 - две буквы Ё;
- с 32C58 по 32C65 - без изменений.

Таблица весов символов для сортировки располагается в позициях 32C66 - 32E65. Если вы хотите, чтобы при сортировке и составлении предметных указателей сначала шли английские буквы, а затем русские, то эту таблицу надо изменить таким образом (после двоеточия идут шестнадцатеричные коды, справа в скобках - соответствующие символы):

```
с 32C96 по 32C9D: 42 43 44 45 46 47 48 49 ( В С D E F G H I )
с 32C9E по 32CA5: 4A 4B 30 31 32 33 34 35 ( J K 0 1 2 3 4 5 )
с 32CA6 по 32CAD: 36 4C 4E 50 52 54 56 58 ( 6 L N P R T V X )
с 32CAE по 32CB5: 5A 5C 5E 60 62 64 66 68 ( Z \ ^ b d f h )
с 32CB6 по 32CBD: 6A 6C 6E 70 72 74 76 78 ( j l n p r t v x )
с 32CBE по 32CC5: 7A 7C 7E 37 38 39 3A 3B ( z | ~ 7 8 9 ; , )
с 32CC6 по 32CCD: 3C 4D 4F 51 53 55 57 59 ( < H O P S U M Y )
с 32CCE по 32CD5: 5B 5D 5F 61 63 65 67 69 ( [ ] - a c e g i )
с 32CD6 по 32CDD: 6B 6D 6F 71 73 75 77 79 ( k m o q s u m y )
с 32CDE по 32CE5: 7B 7D 7F 3D 3E 3F 40 41 ( { } = > ? @ A )
с 32CEE по 32CED: 80 82 84 86 88 8A 8E 90 ( A B D J H K O P )
с 32CEE по 32CF5: 92 94 96 98 9A 9C 9E 98 ( T O U W X Y Z a )
с 32CF6 по 32CFD: A2 A4 A6 A8 AA AC AE B0 ( в д ж н к о п )
с 32CFE по 32D05: B2 B4 B6 B8 BA BC BE C0 ( с - t - u - v - w - x - y - z )
с 32D06 по 32D0D: 81 83 85 87 89 8B 8F 91 ( Б Г Е З И Л П С )
с 32D0E по 32D15: 93 95 97 99 9B 9D 9F A1 ( У Х Ч Ш Э Я Ё )
с 32D16 по 32D1D: C2 C3 C4 C5 C6 C7 C8 C9 ( Т - - - - - )
с 32D1E по 32D25: CA CB CC CD CE CF D0 D1 ( - - - - - )
с 32D26 по 32D2D: D2 D3 D4 D5 D6 D7 D8 D9 ( - - - - - )
с 32D2E по 32D35: DA DB DC DD DE DF E0 E1 ( Г - - - - - )
с 32D36 по 32D3D: E2 E3 E4 E5 E6 E7 E8 E9 ( т у о х к ч ш э я ё )
с 32D3E по 32D45: EA EB EC ED EE EF F0 F1 ( ь ъ ъ ъ ъ ъ )
с 32D46 по 32D4D: A3 A5 A7 A9 AB AD AF B1 ( Г в з й л п )
с 32D4E по 32D55: B3 B5 B7 B9 BB BD BF C1 ( | - - - - )
с 32D56 по 32D57: 8C 8D ( Н )
```





Если же вы хотите, чтобы при сортировке и составлении предметных указателей сначала шли русские буквы, а затем английские, то таблицу надо изменить так:

|                   |                         |                           |
|-------------------|-------------------------|---------------------------|
| с 32С96 по 32С9D: | 42 43 44 45 46 47 48 49 | ( в с д е ф г н и )       |
| с 32С9Е по 32С95: | 4A 4B 38 31 32 33 34 35 | ( j k 0 1 2 3 4 5 )       |
| с 32С96 по 32С9D: | 36 8E 98 92 94 96 98 9A | ( 6 0 p r t o u m b )     |
| с 32С9E по 32С95: | 9C 9E 88 A2 A4 A6 A8 AA | ( b d a a d m k )         |
| с 32С96 по 32С9D: | AC AE 88 B2 B4 B6 B8 BA | ( m o u s e i t n )       |
| с 32С9E по 32С95: | BC BE C0 37 38 39 3A 3B | ( 7 8 9 ; , )             |
| с 32С96 по 32С9D: | 3C 8F 91 93 95 97 99 9B | ( < p c y k ч ш )         |
| с 32С9E по 32С95: | 9D 9F A1 A3 A5 A7 A9 AB | ( э я б г е з й л )       |
| с 32С96 по 32С9D: | AD AF B1 B3 B5 B7 B9 BB | ( н п л - > т е я )       |
| с 32С9E по 32С95: | BD BF C1 3D 3E 3F 40 41 | ( а л - > т е я )         |
| с 32С96 по 32С9D: | 4C 4E 50 52 54 56 5A 5C | ( l n p r t v z \ )       |
| с 32С9E по 32С95: | 5E 60 62 64 66 68 6A 6C | ( ^ b d f h j l )         |
| с 32С96 по 32С9D: | 6E 70 72 74 76 78 7A 7C | ( п р g t v x z   )       |
| с 32С9E по 32С95: | 7E 80 82 84 86 88 8A 8C | ( ~ я в д ж н к м )       |
| с 32С96 по 32С9D: | 4D 4F 51 53 55 57 5B 5D | ( м о d s u m i )         |
| с 32С9E по 32С95: | 5F 61 63 65 67 69 6B 6D | ( - a c e g r i k n )     |
| с 32С96 по 32С9D: | C2 C3 C4 C5 C6 C7 C8 C9 | ( t - e - f - j - l - n ) |
| с 32С9E по 32С95: | CA CB CC CD CE CF DB DI | ( т - e - f - j - l - n ) |
| с 32С96 по 32С9D: | D2 D3 D4 D5 D6 D7 D8 D9 | ( т - e - f - j - l - n ) |
| с 32С9E по 32С95: | DA DB DC DD DE DF E0 E1 | ( г у ш х и ч ш )         |
| с 32С96 по 32С9D: | E2 E3 E4 E5 E6 E7 E8 E9 | ( т у ф х и ч ш )         |
| с 32С9E по 32С95: | EA EB EC ED EE EF F0 F1 | ( ь ы в з ю я ш )         |
| с 32С96 по 32С9D: | 6F 71 73 75 77 79 7B 7D | ( o q s u m u { } )       |
| с 32С9E по 32С95: | 7F 81 83 85 87 89 8B 8D | ( б г е з й л н )         |
| с 32С96 по 32С9D: | 58 59 ( х у )           |                           |

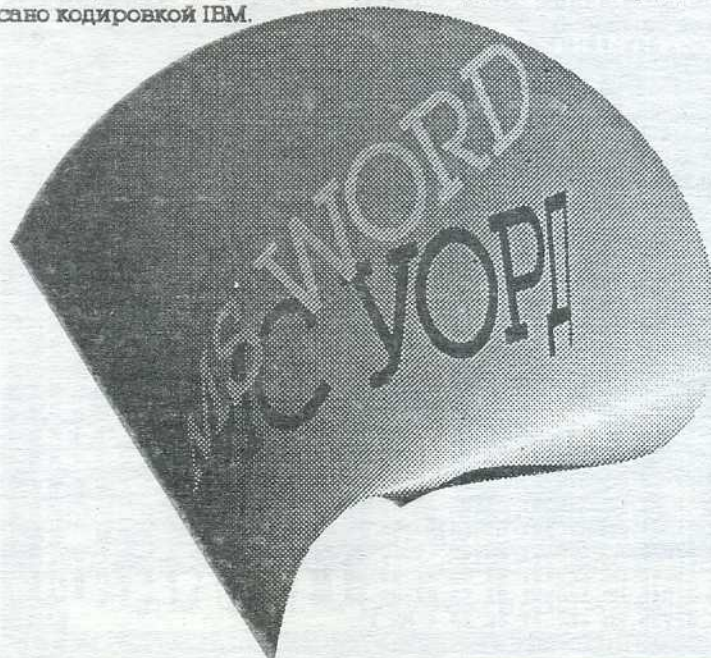
Внесите эти исправления, и вы увидите, что работа с Microsoft Word станет гораздо приятнее!



Считаем своим долгом напомнить читателям, что внесение изменений в программы, являющиеся предметом торгового оборота, связано с нарушением авторских прав. Мы убеждены, что отсутствие правового урегулирования вопросов защиты прав авторов программного обеспечения в СССР - явление временное. Не советуем вносить изменения в код программы в целях иных, чем личное пользование.

Редакция

<sup>1</sup> Разумеется, не следует понимать слова о дефектах Microsoft Word буквально, т.е. как выпад в сторону фирмы Microsoft. Наоборот, она сделала все правильно, но только при этом считала, что в позициях 128-175 и 224-241 кодовой таблицы находятся не русские буквы, а символы европейских алфавитов, греческие буквы и т.д., как это и предписано кодировкой IBM.



© Виктор Фигурнов

# Пишем файл AUTOEXEC.BAT

В настоящей статье мы продолжим начатый во втором номере сборника "Компьютер" разговор, каким образом можно создать на компьютере IBM PC (или совместимом с ним) наиболее удобную для работы обстановку. Теперь мы обсудим написание файла AUTOEXEC.BAT.

Файл AUTOEXEC.BAT должен находиться в корневом каталоге того диска, с которого загружается операционная система MS-DOS. В этом случае он автоматически выполняется при каждой начальной загрузке MS-DOS. В командный файл AUTOEXEC.BAT удобно записывать команды, которые должны выполняться каждый раз при начальной загрузке операционной системы. Эти команды могут осуществить необходимую настройку операционной системы и создать удобную для работы обстановку. Кроме того, при наличии файла AUTOEXEC.BAT операционная система не задает в процессе начальной загрузки вопросов о текущей дате и времени.

## Установка списка каталогов, в которых производится поиск выполняемых программ

С помощью команды MS-DOS Path в файле AUTOEXEC.BAT можно определить, в каких каталогах будет производиться поиск выполняемых программ. Поиск в этих каталогах будет выполняться тогда, когда пользователь введет имя команды без указания, в каком каталоге находится соответствующая программа. Формат команды: Path список-имен-каталогов.

Пример: Path = C:\EXE;C:\EXE\DOS;C:\EXE\NU;C:\TC;...;A.

Имена каталогов в списке должны разделяться точкой с запятой. Поиск выполняемых программ будет производиться сначала в текущем каталоге, а затем в каталогах, указанных в команде Path.

В списке каталогов, указанном в команде Path, следует перечислить те каталоги, в которых находятся исполняемые программы общего назначения. Целесообразно указать также каталоги .. и .., что позволит выполнять любую программу из подкаталогов первого и второго уровней того каталога, в котором она находится.

Много каталогов в команде Path указывать нежелательно, так как это приведет к длительному поиску нужной команды на диске. В большинстве случаев вместо включения имени каталога с тем или иным пакетом прикладных программ в список каталогов команды Path предпочтительнее создать в одном из этих каталогов командный файл для вызова данного пакета прикладных программ.

## Установка формата приглашения MS-DOS

С помощью команды MS-DOS Prompt можно изменять формат приглашения, показывающего, что MS-DOS готова к приему команд пользователя.

Формат команды: Prompt текст-приглашения. В тексте, указываемом в команде Prompt, можно применять специальные сочетания символов, которые имеют следующие значения: \$p - текущий дисковод и каталог; \$n - текущий дисковод; \$d - текущая дата; \$t - текущее время; \$h - удаление предыдущего символа; \$e - символ с кодом 27 (ESC); \$g - символ ">". Имеются и другие специальные сочетания



символов, но они используются редко. Остальные символы, указанные в тексте, являющиеся параметром команды Prompt, переносятся в приглашение MS-DOS без изменения.

Чаще всего употребляются следующие форматы приглашения MS-DOS:

`prompt $p$g` - устанавливает приглашение MS-DOS, содержащее информацию о текущем каталоге и символ ">" (например, `C:\WORK\DOC>`);

`prompt $t $p$g` - устанавливает приглашение MS-DOS, содержащее информацию о времени суток, о текущем каталоге и символ ">" (например, `13:05:43.56 C:\WORK\DOC>`).

Многим пользователям такое приглашение кажется слишком длинным. Они убирают из него информацию о секундах и сотых долях секунды. Для удаления символа в приглашении используется сочетание символов `$h`: `prompt $t$h$h$h$h$h$h$h $p$g`. Вид приглашения MS-DOS будет таким: `13:05 C:\WORK\DOC>`.

Если при запуске операционной системы был установлен драйвер ANSI.SYS (для этого в файле конфигурации `CONFIG.SYS` должна быть строка вида `device = имя-каталог\ansi.sys`), то в приглашении MS-DOS можно воспользоваться различными цветами символов и фона. Для этого в команде Prompt следует задать управляющие последовательности драйвера ANSI.SYS:

`$e[атрибут];...;атрибут>m`

Значения атрибутов задаются десятичными числами. Если в команде указано несколько атрибутов, то они разделяются точками с запятой. Значения атрибутов могут быть таковы: 0 - обычные символы (белые на черном фоне); 1 - символы повышенной яркости; 5 - мигающие символы; 7 - инверсное изображение (черные символы на белом фоне); 8 - невидимые символы (цвет символа совпадает с цветом фона); 30-37 - черный, красный, зеленый, коричневый, синий, голубой и белый цвета символов; 40-47 - черный, красный, зеленый, коричневый, синий, голубой и белый цвета фона.

Например, чтобы получить желтый цвет символов, надо использовать команду `$e[33]m`. С помощью таких команд можно выделять отдельные части приглашения различными цветами. Так, команда `prompt $e[7m]$p$e[0m]$g` выводит имена текущего диска и текущего каталога в инверсионном изображении (на монохромном дисплее), а остальной текст - в обычном изображении. Команда `prompt $e[36m]$t$h$h$h$h$h$h$h $e[35m]$p$e[32m]$g$e[0m` выводит в приглашении текущее время голубым цветом, текущий каталог - сиреневым, а символ ">" - зеленым цветом.

С помощью команд драйвера ANSI.SYS можно вывести часть приглашения в другом месте экрана. Например, информация о текущем времени может быть дана в правом верхнем углу экрана. Для определения приглашения MS-DOS такого вида можно воспользоваться следующими командами: `$e[s` - запоминание текущей позиции курсора; `$e[u` - установка курсора в позицию, которая была запомнена с помощью выдачи управляющей последовательности `$e[s`; `$e[строка;столбец]H` - установка курсора на данную строку и столбец.

Так, команда `prompt $e[s$e[1;52H$e[44;33]m $t $d $e[0;35m$e[u]$p$e[32m]$g$e[0m` выводит в правом верхнем углу экрана желтыми буквами на синем фоне текущее время и дату, затем возвращается в текущую строку и выводит там сиреневым цветом имя текущего каталога и зеленым цветом символ ">".

### Установка переменных окружения

Операционная система отводит специальную область оперативной памяти, называемую "окружением" (Environment), для хранения значений некоторых переменных, которые применяются операционной системой и другими программами. Окружение состоит из строк вида "переменная = значение". Здесь переменная - любая строка, не содержащая символа "=". При этом в записи перемен-

ной большие и малые латинские буквы считаются одинаковыми. Значение - любая строка символов.

Операционная система MS-DOS использует три переменные окружения: Path, Prompt и Comspec, которые устанавливаются соответственно командами Path, Prompt и Command с параметром /P. Пользователь может задавать переменные окружения с помощью команды Set. Формат команды: Set переменная = значение (например, Set CHFILES = C:\CHI).

Если указанной переменной уже было присвоено какое-то значение, то оно заменяется новым.

Прикладные программы могут анализировать область памяти, предназначенную для хранения переменных окружения, и выяснять, установлено ли значение той или иной переменной и каково это значение. Наиболее часто переменные окружения используются для того, чтобы указать, где прикладные программы должны искать вспомогательные файлы.

### Запуск необходимых программ

В файл AUTOEXEC.BAT целесообразно включить команды запуска программ, которые создают привычную рабочую обстановку на компьютере. В настоящем пункте обсуждаются программы, которые часто включаются в файл AUTOEXEC.BAT.

1. Драйверы клавиатуры. Для обеспечения ввода русских букв с клавиатуры компьютеров зарубежного производства приходится использовать специальную программу - драйвер клавиатуры. Эта программа, как правило, запускается в файле AUTOEXEC.BAT, и постоянно находится в оперативной памяти компьютера. Драйвер воспринимает нажатия клавиш на клавиатуре и передает соответствующие символы операционной системе MS-DOS. После нажатия определенной комбинации клавиш он начинает передавать в компьютер символы другого алфавита. Очень удобно, когда драйвер клавиатуры выполняет также и функции по обеспечению вывода кириллицы на экран.

2. Программы для обеспечения изображения символов кириллицы на экране. Если драйвер клавиатуры не обеспечивает изображение символов кириллицы на экране, то для этого необходимы специальные программы, например GRAFTBL или GRAFCYR.

3. Драйверы принтера. В том случае, когда принтер может печатать русские буквы в текстовом режиме, или же когда коды русских букв в принтере не совпадают с их кодами в компьютере, то для печати русских текстов необходимо применять специальные программы. Наиболее удобный такой вариант: в файле AUTOEXEC.BAT запускается драйвер принтера, который постоянно находится в оперативной памяти компьютера и перекодирует все русские буквы, посылаемые на принтер, в соответствующие им коды принтера.

4. Программы для печати копии экрана в графическом режиме. С помощью этих программ вы можете напечатать в любой момент копию экрана, нажав определенную комбинацию клавиш (например, [Shift- PrtScr]). Программа Graphics, входящая в состав MS-DOS, позволяет печатать на принтере, совместимом с IBM Color Printer или IBM Graphics Printer, копию экрана, если экран находится в режиме, совместимом с одним из графических режимов адаптера CGA. Имеются программы, печатающие копии экрана и для других мониторов и принтеров.

5. Программы для установки часов в компьютере. В первых моделях компьютера IBM PC показание часов компьютера необходимо было устанавливать заново после каждого включения компьютера. Это неудобство было устранено различными производителями компьютеров с помощью подключения специальной электронной схемы, содержащей часы и аккумуляторную батарею для обеспечения их работы даже тогда, когда компьютер выключен. С помощью программы, включаемой в файл AUTOEXEC.BAT обеспечивается считывание информации из этих часов в





часы компьютера. Такая программа может называться Timer, Realtime и т.п., она устанавливается фирмами-производителями компьютера на диск с операционной системой MS-DOS.

На многих компьютерах, например компьютерах IBM PC AT и PS/2, считывание информации из энергонезависимых часов обеспечивается во время включения электропитания автоматически.

Если ваш компьютер не имеет встроенных энергонезависимых часов, то придется устанавливать показания часов в компьютере при каждой начальной загрузке MS-DOS. В этом случае в файле AUTOEXEC.BAT должны быть команды Date и Time.

6. Программа DOSEDIT. Эта программа хранит несколько последних введенных пользователем команд и может вывести их (при нажатиях клавиш [t] и [i]) в командную строку MS-DOS. Это позволяет легко повторить одну из последних введенных команд, сделав при необходимости в ней нужные изменения. Команды с неправильными именами (т.е. команды, в ответ на которые MS-DOS выдала сообщение "Bad command or file name") программой DOSEDIT не запоминаются. Иногда в файл AUTOEXEC.BAT включается не команда DOSEDIT, а какая-нибудь аналогичная программа, например CED.

7. Программы для ранней диагностики наличия компьютерного вируса. Эти программы проверяют, не изменились ли на компьютере некоторые файлы, которые не должны изменяться (например, файлы операционной системы MS-DOS). При обнаружении изменения на экран выдается сообщение, какой файл изменен, и предупреждение о возможности наличия в компьютере вируса.

8. Программы для защиты против компьютерного вируса. Эти программы постоянно находятся в памяти, "перехватывают" все запросы к операционной системе MS-DOS на выполнение различных "подозрительных" действий (т.е. операций, которые используют компьютерные вирусы для своего "размножения" и для порчи информации в компьютере) и выводят на экран сообщения, какое действие затребовано и какая программа желает его выполнить. Пользователь может либо разрешить выполнение затребованного действия, либо запретить его. Это позволяет обнаружить компьютерный вирус на самой ранней стадии, когда он еще не успел размножиться и что-либо испортить, и тем самым свести убытки от компьютерного вируса к минимуму.

Команда запуска программы для защиты против компьютерного вируса должна находиться в файле AUTOEXEC.BAT после всех команд для запуска других резидентных программ.

9. Часто в файл AUTOEXEC.BAT включаются программы, выдающие различные напоминающие сообщения (например, сообщения о запланированных на день мероприятиях).

10. Если вы привыкли работать с какими-то программами-оболочками типа программы Norton Commander, то целесообразно вставить команду для запуска соответствующей программы в файл AUTOEXEC.BAT.

### Просмотр почты

При работе на одном компьютере нескольких пользователей часто бывает необходимо передавать друг другу какие-то сообщения. Ниже приводится фрагмент файла AUTOEXEC.BAT, позволяющий выводить на экран при начальной загрузке компьютера файл MAIL в каталоге C:\DOC, если этот файл там существует.

```
if not exist C:\DOC\MAIL goto nomail
cls
type C:\DOC\MAIL
pause :nomail
```

## Определение значений клавиш на клавиатуре

Многие пользователи используют возможности драйвера ANSI.SYS переопределять значения клавиш на клавиатуре для ускорения ввода команд. Они присваивают различным клавишам значения таким образом, чтобы при нажатии на эти клавиши выполнялись наиболее часто используемые команды.

Для того чтобы переопределить значения клавиш, надо послать на экран специальную управляющую последовательность драйвера ANSI.SYS. Затем при нажатии на эту клавишу будет выдаваться указанная в команде переопределения строка символов. Если эта строка заканчивается символом "конец строки" (символ с кодом 13), то она тут же будет выполнена как команда MS-DOS.

Чтобы при нажатии на клавишу в командную строку выводилась команда, надо послать на экран следующую управляющую последовательность драйвера ANSI.SYS: +{код-клавиши};"команда"р. Здесь стрелкой влево (+) изображается символ с кодом 27 (Escape). После нажатия на соответствующую клавишу в командную строку будет выведена указанная команда. Ее можно отредактировать и, нажав клавишу [Enter], выполнить. Если вы хотите, чтобы при нажатии на клавишу выполнялась некоторая команда, надо послать на экран управляющую последовательность драйвера ANSI.SYS: +{код-клавиши};"команда";13р.

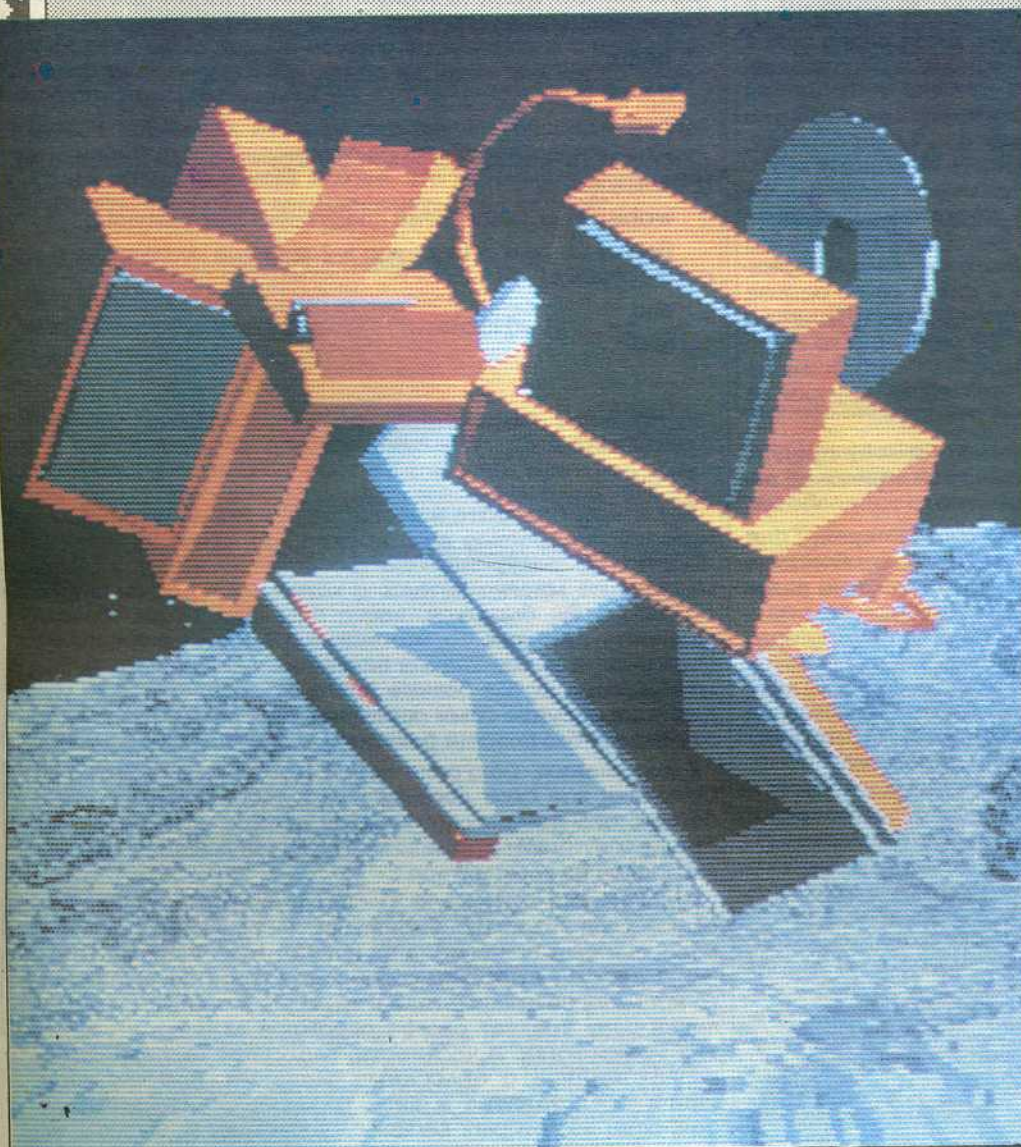
Как правило, переопределяются значения следующих клавиш и комбинаций клавиш: [F1] - [F10] (коды 0;59 - 0;68), [Shift-F1] - [Shift-F10] (коды 0;84 - 0;93), [Ctrl-F1] - [Ctrl-F10] (коды 0;94 - 0;103) и [Alt-F1] - [Alt-F10] (коды 0;104 - 0;113). При определении значений клавиш бывает затруднительно ввести символ с кодом 27, так как он является управляющим и его не всегда можно ввести с клавиатуры. Один из выходов таков: надо с помощью программы на каком-либо языке программирования создать файл с нужными управляющими последовательностями, а затем выводить этот файл командой Type.

### Пример файла AUTOEXEC.BAT

Для наглядности файл снабжен подробными комментариями. Если версия MS-DOS на компьютере до 3.3, то из первой строки следует удалить символ "@".

```
@echo off
rem Установка пути поиска программ
path C:\EXE\DOS;C:\EXE;C:\EXE\NU;C:\TC;...
rem Установка вида приглашения MS-DOS
prompt $e[36m$th$hh$hh$hh$hh$e[35m$p$e[32m$g$e[0m
rem Установка переменных окружения
set CHFILES=C:\CHI
set LIB=C:\FORTRAN;C:\MSC\LIB
set INCLUDE=C:\MSC\INCLUDE
set TMP=E:\
rem Запуск драйвера клавиатуры
C:\EXE\FILOAD\RK
rem Запуск программы DOSEDIT
DOSEDIT
rem Диагностика наличия вируса
rem
C:\EXE\ANTIAIDS\CRCTEST
C:\EXE\ANTIAIDS\CRCTEST.DAT
rem
rem Запуск программы для защиты от вируса
rem
C:\EXE\ANTIAIDS\ANTIVIRU
rem
rem Запуск программы Norton Commander
rem
NC
```





- ☐ Ханойская башня и кролики
- ☐ Компрессия памяти ZX Spectrum
- ☐ HeToBa
- ☐ Ремонт ATARI XL/XE
- ☐ Секреты ATARI XL/XE

Компьютер дома

© Анджей Кадлоф

## Ханойская башня и кролики

В большинстве языков программирования высокого уровня при составлении программ можно использовать механизм рекурсии. Однако эта мощная техника не получила широкого распространения среди пользователей домашних компьютеров. Главной помехой в этом был, пожалуй, весьма популярный Бейсик: он не дает возможности пользоваться рекурсией естественным образом. Рекурсивное программирование при всей своей эффективности достаточно трудно и его применение требует определенного опыта.

Теоретические основы рекурсии исходят из принципов математической индукции. Несмотря на то, что этот метод широко используется в современной математике, немногим известна его сущность. Многие специалисты по информатике относятся к рекурсии со смешанными чувствами — она имеет как противников, так и поклонников. А истина, как известно, всегда находится посередине.

Давайте на двух классических примерах посмотрим, что же представляет собой эта техника (в обоих случаях будем пользоваться языком Паскаль).

Старая легенда гласит:

“В Индии, в городе Бенаресе, в главном храме, где находится центр Вселенной, поставил Брахма на березовой доске три алмазных стержня высотой в один локоть (576 мм) и толщиной с туловище пчелы. При сотворении мира на один из этих стержней были нанизаны 64 кружка различного диаметра с отверстиями в центре. Сделаны они были из чистого золота, а нанизаны таким образом, что образовывали усеченный конус. Долгие годы, днем и ночью, сменяя друг друга, жрецы переносят этот конус с первой палочки на третью, используя вторую только как вспомогательную. Они должны свято соблюдать два условия: во-первых, за один раз не переносить больше одного кружка, а во-вторых, никогда не располагать большой кружок на меньшем. Когда жрецы смогут закончить свою работу — тогда и наступит конец света...”





Нет ничего удивительного в том, что, имея перед собой такую перспективу, этой проблемой, кроме жрецов, занялись и математики. Они подсчитали, что для выполнения задачи Брахмы нужно перенести кружки 18 446 744 073 709 551 615 раз. Если жрецы будут добросовестно относиться к своей работе и переносить по одному кружку в секунду, то они справятся с заданием всего лишь через пятьсот миллиардов лет. После таких обнадеживающих и успокаивающих результатов вычисления легенда была предана забвению. Однако забыли ее не все. Французский математик Люкас вспомнил о ней и на ее основе придумал игру. Он назвал ее "Ханойская башня". Принцип игры остался тем же, что и в задаче Брахмы, однако количество кружков сократилось до восьми. В таком варианте можно, соблюдая все ограничения, перенести кружки за 255 ходов.

Прежде чем вы начнете читать дальше, я хочу предложить вам придумать алгоритм решения этой задачи на вашем компьютере. Давайте подумаем вместе.

Как взялся бы за эту работу математик? Сначала, для собственного удобства, он ввел бы некоторые обозначения, а именно обозначил бы те стержни, на которые навешиваются кружки, буквами А, В и С. Принимаясь за решение новой задачи, математики обычно сначала анализируют самый простой случай, т. е. допускают, что существует только один кружок. "Ведь это совсем просто! - воскликнет читатель. - Достаточно перенести этот кружок со стержня А на С - и все!" Между прочим, именно в этом заключается могущество принципа творческого упрощения сложных решений! Если я могу справиться с одним кружком, - думаю математик, - то почему бы не решить эту задачу с  $n$  кружками?" (Всем известно, что в большинстве своем математики - оптимисты!) Затем в соответствии со своими предположениями наш герой попытается определить, продвигается ли дело, если он минимально усложнит задачу, т. е. добавит еще один кружок. Оказывается, продвигается. Сначала он перенесет  $n$  кружков со стержня А на В (это он уже умеет делать), а потом оставшийся на А самый большой кружок он перенесет на С (это в том банальном случае, когда есть только один кружок), после чего  $n$  кружков с В перенесет на С. Он вполне сможет это сделать, так как самый большой кружок уже находится на С и не препятствует свободному переносу остальных, поскольку их только  $n$  штук. Вот, казалось бы, и все! Наш математик решит, что он справился с задачей и дальше заниматься ею неинтересно. От случая к случаю он будет хвастаться тем, что решил задачу не только для восьми кружков, но и для произвольного их количества.

"Подождите-подождите! Не так быстро!" - воскликнете вы и будете совершенно правы. Такой вывод можно было бы принять, если бы математик действительно сумел решить задачу с  $n$  кружками. Но этот картонный домик моментально рассыплется, если выяснится, что наш герой просто похвастался. Однако тут я должен встать на защиту математика. В своих действиях он воспользовался принципом математической индукции. Сущность его логических рассуждений находится не в том фантастическом предположении, что он сможет сделать почти все, а в доказательстве, что, зная решение задачи с  $n$  кружками, он сможет ее решить и с  $n+1$  кружками. Именно в этом доказательстве находится рецепт того, как же после тривиального случая с одним кружком решить эту же проблему с двумя, тремя, четырьмя и т.д. кружками. Другое дело, что явный перечень найденных таким образом решений возможен только для ограниченного количества кружков. Но это зависит не от метода решения, а от необходимого числа шагов. Чем их больше - тем лучше проявит себя компьютер.

А вот и программа, решающая нашу задачу.

Program Ханой;

type кружки : 0..8; [число кружков]

стержни : 'A'..'C'; [обозначение стержней]

procedure Башня (n : кружки; a,b,c : стержни);

begin

if n > 0 then

begin

Башня (n-1,a,c,b); [перенос n-1 кружков со стержня a на b, пользуясь c]

writeln (a,c); [напечатай следующий шаг решения]

Башня (n-1,b,a,c) [перенос n-1 кружков со стержня b на c, пользуясь a]

end

end;

begin

Башня (8,'A','B','C')

end.

Вот и все! Простота и элегантность таких решений производят прекрасное впечатление. Это - одно из главных достоинств рекурсии. Она позволяет скрывать в своем механизме многие детали, с которыми в другом случае пришлось бы бороться, часто затемняя таким образом сущность решения. То, что вначале выглядело трудным и сложным, в конце концов оказалось очень простым. Это математики любят больше всего. Если я вас еще не убедил - попробуйте сами составить программу, решающую эту задачу без помощи рекурсии, и сравните результаты!

К сожалению, рекурсия порой может быть обманчива. Сохраняя все свои достоинства в теоретических разработках, она иногда оказывается крайне неэффективной при некритичном применении в случае решения конкретных задач на компьютере.

Итальянский математик Леонардо из Пизы, более известный под именем Фибоначчи (сокращение от filius Bonacci - сын Боначчи), в начале XII века написал книгу "Liber abacci", в которой содержатся почти все знания того времени об арифметике и алгебре. В последующие столетия этот трактат оказал огромное влияние на развитие математики. В частности, благодаря ему европейцы познакомились с арабскими цифрами. Большая часть книги представлена в виде решения определенных задач, одна из которых касалась популяции кроликов и выглядела примерно так:

"Сколько пар кроликов родится в течение года от одной пары? Некто поместил одну пару кроликов в огороженное со всех сторон пространство, чтобы узнать, сколько же потомства появится за год, а кроличья натура такова, что взрослая пара производит на свет пару молодых каждый месяц, а молодые подрастают и начинают размножаться через два месяца..." Далее идут вычисления количества пар кроликов в каждом месяце и затем выводятся окончательный результат: к концу года будет 377 пар кроликов.

Давайте попытаемся найти алгоритм, позволяющий подсчитывать количество кроликов через  $n$  месяцев. Разрешим опять нашему математику заняться этой проблемой. Он опять сначала обозначит символами  $u_n$  число кроликов в начале  $n$ -го месяца, а затем займется самым простым случаем, т. е. определит  $u_1=1$ . Далее он предположит, что уже знает, сколько кроликов будет через  $n$  месяцев. В каждом следующем месяце будут все те кролики, которые были в предыдущем и, кроме того, каждая пара, которая прожила уже два месяца, даст свое потомство. Другими словами, математик составил рекуррентную формулу следующего вида:

$$(1) \quad u_0=0, u_1=1$$

$$(2) \quad u_{n+1} = u_n + u_{n-1}$$

С помощью этой формулы можно легко определять очередные значения последовательности  $u_n$ . Проверьте -  $u_{12}$  действительно равно 377. У вас должны получиться числа 0, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233 и 377. Последователь-



ность чисел, отвечающих условиям (1) и (2), весьма важна для математики. Она может неожиданно появляться во многих решениях, ничего общего не имеющих с исходными кроликами. В честь Леонардо из Пизы эти числа называются числами Фибоначчи.

Напишем функцию, которая определяет очередные элементы числовой последовательности. Элегантность их рекуррентного определения такова, что невозможно устоять перед искушением и не воспользоваться функцией:

```
function Fibonacci (n:integer) : integer;
if n in [0,1] then Fibonacci := n
else Fibonacci := Fibonacci (n-1) + Fibonacci (n-2)
end;
```

(Не пытайтесь, даже случайно, использовать такие "штучки" в Бейсике! Он не предусматривает того, чтобы функция вызывала сама себя. Если же все-таки захотите поэкспериментировать с рекурсией, то воспользуйтесь, так же, как здесь, Паскалем или Лого.)

Без рекурсии такая функция должна была бы выглядеть примерно так:

```
function Fib (n:integer) : integer;
var i,u,v : integer;
begin
v:=0; u:=1; i:=1;
while i<n do
begin
i:=i+1;
u:=u+v
v:=u-v
end;
Fib:=u
end;
```

Теперь вы сами видите, что вариант с использованием рекурсии выглядит изящнее. К сожалению, только на бумаге! Хотя обе функции действуют правильно, но уже для  $n > 6$  начинает замедляться работа рекуррентной функции. Почему так происходит?

Для вычисления  $n$ -го числа первая функция двукратно вызывает сама себя для вычисления двух предыдущих значений последовательности. Для каждого из них она вновь вызывается по два раза. Итак, для вычисления  $n$ -го числа эта функция должна вызвать сама себя примерно  $2^n$  раз. Нереккуррентный вариант для вычисления  $n$ -го числа Фибоначчи потребует только  $3n$  операций.

Из приведенного примера можно сделать важный вывод: применять рекуррентные алгоритмы следует весьма осторожно. Если есть возможность не пользоваться рекурсией, то, может быть, безопаснее так и поступать? Но это, конечно, не означает, что ее нужно избегать любой ценой. Существует большое количество алгоритмов рекуррентного типа, для решения которых этот способ очень удобен. "Ханойская башня" служит ярким доказательством.

Возвращаясь к последнему примеру, добавлю еще, что не все математики удовлетворены рекуррентным решением. Для вычисления последовательности Фибоначчи вывели даже следующую формулу:

$$u_n = \frac{(1+5^{1/2})^n - (1-5^{1/2})^n}{5^{1/2}}$$

Несмотря на свою простоту она, однако, не позволяет вычислять  $n$ -ное число Фибоначчи быстрее, чем приведенная выше нереккуррентная функция Фибоначчи. По крайней мере не с помощью компьютера.

Перевод Тадеуша Радюша

Компьютер дома

© Владислав Балясов

# Компрессия памяти ZX Spectrum

Предлагаем вниманию читателей программу компрессии/декомпрессии памяти для компьютера ZX Spectrum. Она обладает следующими особенностями.

1. Программа полностью перемещаемая, т.е. может размещаться по любому удобному адресу памяти (это относится как к компрессору, так и к декомпрессору).

2. Программа предназначена для компрессии и декомпрессии областей памяти произвольной длины, но с успехом может применяться и для компрессии экрана.

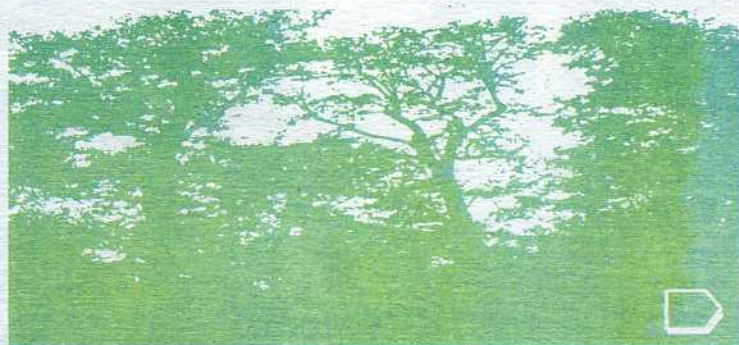
3. Декомпрессия выполняется чрезвычайно быстро, соизмеримо со скоростью выполнения команды LDIR микропроцессора.

Алгоритм компрессии ощутимо превосходит по скорости подобную программу, опубликованную в 1987 г. в польском журнале "Komputer". Значительно выше и его эффективность. Как пример можно привести программу BOULDER DASH IV - при первоначальной длине "кодированного" файла в 23 Кбайта "сжатый" файл занимает всего 15.5 Кбайта, т.е. "сжат", с помощью предлагаемой вниманию читателей программы на 30%.

Ниже приведены распечатки, полученные с помощью ассемблера GENS4 (или GENS3), операторы DATA, содержащие коды компрессора и декомпрессора (для ввода программ без ассемблера), и пример использования компрессора - программа на Бейсике, предназначенная для управления компрессором (листинги 1 - 5 соответственно).

Для использования приведенной программы на Бейсике необходимо перенести коды из операторов DATA в строку комментариев (первая строка программы) любым удобным способом, например применяя READ и POKE (строка находится в памяти по адресу 23760, если не используется INTERFACE 1 и/или дисковая система). Первые 116 байт строки содержат компрессор, затем 4 зарезервированных байта и декомпрессор - 31 байт. Декомпрессор вызывается обычным способом - RANDOMIZE USR X. Обратите внимание, что первые две команды декомпрессора - это установка адреса "сжатого" файла в памяти (HL) и адреса, по которому будет размещаться восстановленный файл.

Компрессор вызывается с помощью оператора DEF FN и функции FN. В программу на Бейсике декомпрессор возвращает адрес последнего использованного для сжатия байта памяти.





### Листинг 1

```
(C) BY BALIASOV V.A. 9 MAY 1989
FILE COMPRESSOR ROUTINE
DEF FN C(FROM,TO,LENGTH)=USR
65000
LET A=FN C(16384,40000,6912)
ORG 65000
MAIN LD IX,(23563)
LD DE,8
LD B,3
PARML LD L,(IX+4)
LD H,(IX+5)
PUSH HL
ADD IX,DE
DJNZ PARML
POP HL
POP IX
POP DE
ADD HL,DE
EX DE,HL
COMP PUSH IX
LD (IX+0),0
COMPI INC IX
LD C,0
LD A,(HL)
INC HL
CP (HL)
JR NZ,GETCA
INC HL
CP (HL)
JR NZ,GETCA1
DEC HL
DEC HL
GETCSM CP (HL)
JR NZ,GETCE
INC HL
AND A
SBC HL,DE
ADD HL,DE
INC C
JR NC,COMP2
JR NZ,GETCSM
GETCE SCF
JR COMP2
GETCA1 DEC HL
GETCA AND A
SBC HL,DE
ADD HL,DE
INC C
COMP2 LD (IX+0),A
EX AF,AF
EX (SP),IX
INC (IX+0)
DEC C
LD A,255
CP (IX+0)
EX (SP),IX
JR Z,SKIPC
INC A
CP C
JR Z,NEXT
SKIPC INC IX
LD (IX+0),C
INC IX
EX AF,AF
SKIPCI POP BC
JR C,COMP
COMPT LD (IX+0),0
PUSH IX
POP BC
RET
NEXT EX AF,AF
JR C,COMPI
JR SKIPCI
```

### Листинг 2

```
(C) BY BALIASOV V.A. 9 MAY 1989
FILE DECOMPRESSION ROUTINE
INPUT PARAMETERS:
DE - "TO" POINTER
HL - "FROM" POINTER
ORG 65300
LD DE,16384 ; "TO"
LD HL,40000 ; "FROM"
LD B,0
REST LD A,(HL)
OR A
RET Z
LD C,A
INC HL
LDIR
LD A,(HL)
INC HL
AND A
JR Z,REST
PUSH HL
DEC DE
LD H,D
LD L,E
INC DE
LD C,A
LDIR
POP HL
JR REST
```

### Листинг 3

```
1000 DATA 221,42,11,92,17,8,0,6,3,221
1001 DATA 110,4,221,102,5,229,221,25,16,245
1002 DATA 225,221,225,209,25,235,221,229,221,54
1003 DATA 0,0,221,35,14,0,136,35,190,32
1004 DATA 23,35,190,32,18,43,43,190,32,10
1005 DATA 35,167,237,82,25,12,48,11,32,243
1006 DATA 55,24,6,43,167,237,82,25,12,221
1007 DATA 119,0,8,221,227,221,52,0,13,62
1008 DATA 255,221,190,0,221,227,40,4,60,185
1009 DATA 40,19,221,35,221,113,0,221,35,8
1010 DATA 193,56,179,221,54,0,0,221,229,193
1011 DATA 201,8,56,174,24,240
```

### Листинг 4

```
1000 DATA 17,TOL,TOH,33,FROML,FROMH
1001 DATA 6,0,126,183,200
1002 DATA 79,35,237,176,126,
1003 DATA 35,167,40,244,229
1004 DATA 27,98,107,19,79
1005 DATA 237,176,225,24,233
```

### Листинг 5

```
1 REM .....
10 CLEAR VAL "25499"
12 DEF FN C(F,T,L)=USR E
15 DEF FN H(I)=INT(I/VAL "256")
20 DEF FN L(I)=I-VAL "256"*FN H(I)
30 LET E=PEEK VAL "23625"+VAL "256"*PEEK VAL
"23636"+VAL "5"
50 PRINT "By Baliasov V.A. Moscow 1989"
100 INPUT "Load file at ";F
105 INPUT "Length of file ";L:LET X=L
110 INPUT "Address of compressed file ";T
112 GOSUB VAL "850"
115 LOAD R$ CODE F
120 LET A=FN C(F,T,L)+SGN PI
140 CLS:LET L=A-T
150 GOSUB VAL "800"
160 INPUT "Include decompressor (Y/N)";LINE R$
170 IF R$="Y" AND R$="y" THEN GOTO 500
180 INPUT "Address of compressed file ";F
190 INPUT "Decompress to ";T1
195 LET E=E+VAL "120"
200 FOR I=NOT PI TO VAL "30":POKE A+I,PEEK
A+I:NEXT I
210 POKE A+VAL "1",FN L(T):POKE A+VAL "2",FN
H(T1)
220 POKE A+VAL "4",FN L(F):POKE A+VAL "5",FN
H(F)
250 PRINT "Address of decompressor ";+L
255 LET L=L+VAL "31"
260 GOSUB VAL "800"
500 INPUT "Save compressed file (Y/N)";LINE R$
510 IF R$="Y" AND R$="y" THEN STOP
520 GOSUB VAL "850"
530 SAVE R$ CODE T,L
800 PRINT "Memory used from ";T; TO ";T+L-
SGN PI
804 LET R$=STRING$(VAL "100"*(SGN PI-L/X))
805 PRINT "Compress factor ";R$(TO
4);"%":RETURN
850 INPUT "Enter file name ";LINE R$:RETURN
9999 SAVE "COMPRESSOR" LINE VAL "10"
```

Компрессия



Компьютер дома

© Петр Завицкий

# HeToBa

HeToBa - вспомогательная программа-утилита для ZX Spectrum. Ее задача состоит в том, чтобы облегчить жизнь тем, кто хочет прислать распечатку своей программы, например, в редакцию "Компьютера", но не имеет желания внимательно и кропотливо вводить в строки DATA несколько сотен байт данных. Эта работа действительно требует напряженного внимания - при значительном количестве данных очень легко ошибиться. HeToBa способна заменить человека и выполнить свою задачу быстро и безошибочно.

Запущенная программа генерирует строки DATA, содержащие 32 байта памяти, которые представлены в виде шестнадцатеричных чисел, а также контрольный байт.

## Как пользоваться программой?

1. Вводим строки 9990 - 9999 и записываем на кассету под названием "LOADER" (программа-загрузчик, вписывающая машинный код из строк DATA в память). Этой программой мы будем пользоваться для загрузки как программы HeToBa, так и генерируемых ею программ. "LOADER" написал не я - это фрагмент программы "Alima" Роланда Вацлавека, опубликованной в польском журнале "Miody Technik", N 3, 1986 г.

2. Допишем строки 1 - 1010 и записываем на кассету, что позволит многократно использовать программу HeToBa. Затем вводим команду RUN.

3. Вводим в память компьютера программу, которую нужно закодировать в строках DATA (под любой адрес, не конфликтующий с программой HeToBa и находящийся ниже, чем RAMTOP).

4. Вводим команду NEW и добавляем строку: 1 NEW.

5. С помощью операторов POKE вписываем адрес начала нашей программы (под адреса 65316 и 65317) и его величину (под адреса 65319 и 65320). Первым, конечно, вводится младший байт. Затем следует выполнить PRINT USR 65314. Спустя несколько секунд HeToBa завершит свою задачу. В памяти Бейсика теперь записаны строки DATA с рядами байтов машинной программы. Они пронумерованы в порядке очередности, начиная с номера 2. Достаточно найти последнюю строку, чтобы выяснить их число. Затем дописываем строку: 1 DATA [адрес начала программы в памяти], [число строк].

6. И последнее: выполняем MERGE "LOADER". Готово - машинный код перенесен в строки DATA.

Внимание! Перед запуском машинного кода программы HeToBa необходимо выполнить NEW и дописать строку 1 NEW (PRINT, REM или RUN), как указано в пункте 4. Отсутствие этой строки может вызвать сбой системы. В заключение добавим, что строки 2 - 8 листинга являются результатом работы самой программы HeToBa.

Перевод Анджея Поплавского

1 DATA 65314,7

2

DATA "DD21000001000078B1C879E61F3200583E05  
CB38CB193D20F9C52100003A0058F0"

3

DATA "A7280103114A00190B78B120FA010600A7ED4  
2444D2A4B5C2BCD5516C178B1132"

4

DATA "02002A535C2828C5CDEFF06200E00DD7E  
00CDC8FF79DD86004FDD2310F179CD40"

5

DATA "C8FF362223360D2313C10B78B120D83A0058A  
7C847CDEFF0E0079DD86004FDDC1"

6

DATA "7E00DD23CDC8FF10F179CDC8FF3A0058473  
E20904736222336202336202310F8A3"

7

DATA "424B0B360DC9C5CDCEFFC1C94FCB3FCB3  
FCB3FCB3FCDE5FF7779E60FCDE5FF236E"

8

DATA "7723C9FE0A3803C637C9C630C972237323364  
62336002336E423362223C9CF

1000 CLEAR 65313: PRINT "МАГНИТОФОН СТОП  
!": PAUSE 100

1001 CLS: PRINT "HeToBa 2.0""Петр ЗАВИЦКИЙ":  
PAUSE 100

1010 PRINT " INVERSE 1;"Длина - 65319""Старт -  
65316""Запуск - USR 65314"

9990 REM \*\*\*\*\*

9991 REM \*\* HeToBa 2.0 - загрузчик \*\*

9992 REM \*\*\*\*\*

9993 READ A,S: FOR F=1 TO S

9994 READ L\$: LET L=LEN L\$: LET S=0: LET K=2

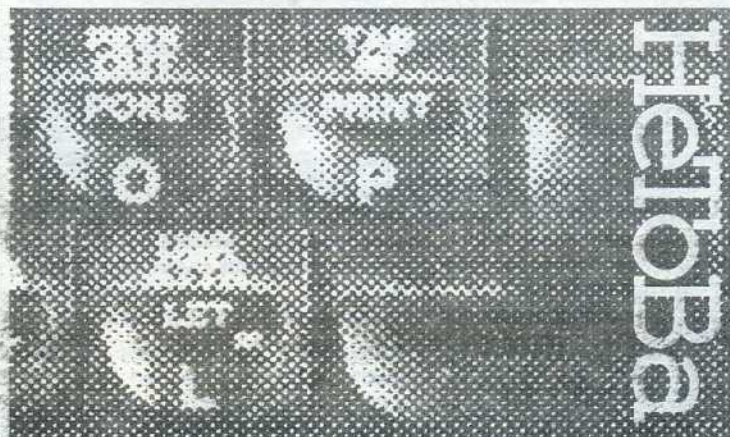
9995 LET A=L\$(K-1): LET B=L\$(K)

9996 LET C=(CODE A\$-48-(7\*(A\$,"@")))\*16+CODE  
B\$-48-(7\*(B\$,"@"))

9997 IF K<L THEN POKE A,C: LET S=S+C: LET  
K=K+2: LET A=A+1: GO TO 9993

9998 IF S-256\*int (S/256)>C THEN PRINT  
"ОШИБКА В СТРОКЕ ";F+1: STOP

9999 NEXT F





\компьютер дома\

© Владислав Солецкий

# Ремонт ATARI XL/XE

Случается, что блоки питания компьютеров выходят из строя - ничто в мире не вечно... Такая же грустная история произошла и с моим компьютером - я был вынужден отремонтировать его блок питания собственными силами. Теперь мне хотелось бы поделиться опытом с читателями журнала. Я очень надеюсь, что он окажется полезным и поможет избежать целого ряда ошибок, которых, к сожалению, я не миновал.

К компьютерам ATARI выпускались блоки питания нескольких типов. Если мы имеем дело с "легким" блоком, корпус которого напоминает кружево (это сделано отнюдь не для красоты, а только для улучшения теплообмена), то ремонт не так сложен, как может показаться сначала. После того как снят ажурный корпус, мы получаем свободный доступ к элементам блока и можем понять схему и приступить к ремонту. Хуже, если выходит из строя "тяжелый" блок питания, причем "тяжелый" в прямом смысле этого слова, так как электрическая схема таких блоков после монтажа заливается эпоксидной смолой. Теоретически такой блок отремонтировать невозможно. Однако, когда случается авария, а замена блока связана с непреодолимыми трудностями, волей-неволей приходится делать теоретически невозможную работу.

Меня не было в тот момент, когда блок питания компьютера вышел из строя, но как мне рассказали, при этом в домашней сети "вылетели" автоматические пробки и из-за противного специфического запаха трудно было дышать.

На заводском щитке испорченного блока питания было написано: DV-315 UP и T40E 22 VA: 5V=1.5A 7.5VA. Измерение сопротивления показало, что перегорела первичная обмотка силового трансформатора. Но как до нее добраться? После усиленного опроса друзей выход нашелся.

Как я уже говорил, вся внутренняя поверхность блока питания залита эпоксидной смолой. В частности, в ней утоплены выступы пластмассового дна корпуса, однако она заполняет не весь объем. Между слоем смолы и дном находятся небольшие участки свободного пространства высотой 7-10 мм, в чем можно убедиться, простучав дно корпуса. Такая конструкция блока питания обеспечивает теплообмен через верхнюю и боковую его поверхности. Для повышения теплоотдачи следовало бы просверлить ряд отверстий диаметром 3 мм в боковых стенках.

Чтобы снять корпус, необходимо острым ножом срезать все залитые смолой выступы, причем именно срезать, а не пытаться применить силу, так как это обязательно приведет к повреждению корпуса.

После того как корпус снят, можно приступить к дальнейшим электрическим измерениям. Печатная плата с элементами слегка прикрыта эпоксидной смолой и поэтому изменения можно произвести, только воспользовавшись чем-либо тонким и острым, например булавками.

Если трансформатор исправен и мы сумеем определить, какие же элементы платы повреждены, то попробуем заменить их новыми, срезав неисправные. Но вполне возможно, что потребуются дальнейший демонтаж блока

питания. Тут уж, волей-неволей, придется прибегнуть к более варварским методам.

Положите что-то мягкое, например кусок ткани, на твердое, жесткое основание (лучше всего - бетонное), и начните сильно колотить нижней частью блока до тех пор, пока из него не вывалятся все внутренности, т.е. трансформатор с платой. Три вывода придется отпаять или отрезать. Затем вооружитесь небольшим зубилом и молотком и аккуратно отбейте эпоксидную смолу, расположенную вдоль пластины алюминиевого радиатора. Желательно, чтобы кто-нибудь при этом помогал вам придерживать то, что осталось от блока питания. Затем, очень осторожно и деликатно, вам придется с помощью того же зубила "выкрошить" остатки эпоксидной смолы.

На первичной обмотке трансформатора находится предохранитель 2A/250V/110°C. Зачастую именно он и выходит из строя. Я бы посоветовал вам применить предохранитель 0,6A/250V, который можно установить непосредственно в сетевую розетку. Второй предохранитель имел номинал 5A/250V. Я заменил его более чувствительным, 1,6A/250V.

Если же вышла из строя обмотка трансформатора, то необходимо его разобрать. Сечение сердечника трансформатора - 5 см, диаметр проволоки, используемой для первичной обмотки, - 0,17 мм. Согласно моим расчетам, для ремонта вы можете воспользоваться проволокой диаметром 0,2 мм.

Электронную часть блока питания я описывать не буду, я - не электронщик. Самому бы мне не справиться с этой частью ремонта без помощи друга.

При перемотке первичной обмотки я перестарался, и при нагрузке напряжение падало, поэтому пришлось отмотать провод обратно до тех пор, пока напряжение не составило 5 В. В конце концов на корпусе осталось 1495 витков проволоки диаметром 0,2 мм.

Иногда в блоке питания выходит из строя диод выпрямителя, а так как отечественных аналогов (1,5A) в Польше не существует, то вместо него приходится использовать транзистор BD 282 (или BD 283). Транзисторы нужно подбирать таким образом, чтобы они выделяли на корпус одинаковое количество теплоты.

Заключительная стадия ремонта - размещение и укрепление в корпусе платы и трансформатора - зависит исключительно от вашей изобретательности.

Перевод Тадеуша Радюша





Zakłady Elektroniczne ELWRO  
 Biuro Handlu Zagranicznego  
 ul. Ostrowskiego 9, 53-238 Wrocław  
 tel. 616833 tlx. 0715518 факс 446521

Гарантийный и послегарантийный сервис, его сеть в стране и за рубежом будут в Твоем распоряжении, если Ты купишь в ЭЛВРО:

- подсистемы телеобработки данных TELE JS-M с эмуляторами терминалов на ПК ELWRO 801 AT,
- ПК ELWRO 801 AT (совместимый с IBM PC/AT) в 23 конфигурациях /по желанию - обеспечиваем расширение конфигурации/,
- школьные микрокомпьютеры:
- ELWRO 800 Junior в сетевом и автономном вариантах,
- ELWRO 804 Junior PC, дешевый домашний микрокомпьютер (новинка)
- печатающие устройства 15" Elwro-Seikosha MP 5350 AI (300 стр. в сек.),
- калькуляторы,

# ELWRO

Представительство:  
 ВВТ "ЭЛВРО", 19390 Москва,  
 ул. Мосфильмовская 54  
 тел. 147-64-83, телекс 414405

- корпуса ПК с блоками питания и главной платой,
  - услуги по модернизации вычислительных центров:
  - замена IC современной и надежной ЕС2134,
  - модернизация ЕС8371.01,
  - увеличение объема дисковой памяти до 2 Гигабайт и более,
  - модернизация парка терминалов путем введения или замены старых моделей новыми, в том числе программируемых, основанных на ПК PC/AT,
  - замена старых модемов надежными современными, программируемыми ЕС8113.
- Сроки поставок - от 2 до 8 недель.  
 Высокое качество поставок.  
 Цены на 20 проц. ниже, чем у конкурирующих фирм.  
 Обучение по программе, согласованной с получателем.

Предварительные заказы на журнал "КОМПЬЮТЕР" принимаются в следующих книжных магазинах и организациях:

Москва:

117334, Ленинский пр-т 40, "Дом научно-технической книги"

101000, ул. Кирова 6, "Дом Книги", тел. 928-87-44

103031, ул. Петровка 15,

пр-т Калинина 26,

ул. Большая Полянка 28

191186 ЛЕНИНГРАД Невский пр-т 28, тел. 219-90-05

370000 БАКУ ул. Ворошилова 1, тел. 93-60-51

375010 ЕРЕВАН ул. Абовяна 1/1

ЗЕЛЕНОГРАД Сосновая Аллея, корп. 607

330063 ЗАПОРЖЬЕ ул. Горького 55, Облкниготорг

664003 ИРКУТСК ул. К.Маркса 20

252054 КИЕВ ул. Менжинского 2, тел. 216-48-11

277012 КИШИНЕВ ул. Пушкина 15, тел. 22-51-33, 22-53-26

350001 КРАСНОДАР ул. К.Либкнехта 87, тел. 33-91-29

660049 КРАСНОЯРСК пр-т Мира 86, Облкниготорг

290006 ЛЬВОВ ул. Народной Гвардии им. Франко 1, Облкниготорг

220050 МИНСК ул. Кирова 3, тел. 20-73-27

220030 МИНСК ул. Красноармейская 24, Облкниготорг

212006 МОГИЛЕВ пр-т Мира 20, Облкниготорг

173016 НОВГОРОД ул. Менделеева 1, Облкниготорг

630049 НОВОСИБИРСК Красный пр-т 150

270001 ОДЕССА, ул. Жуковского 22, Облкниготорг

644099 ОМСК ул. Ленина 17

614600 ПЕРМЬ ул. К.Маркса 19, Облкниготорг

226001 РИГА ул. Петра Стучки 46, тел. 29-75-45

226253 РИГА ул. Падомью 24

620077 СВЕРДЛОВСК ул. Антона Валика 12

620151 СВЕРДЛОВСК ул. К.Либкнехта 16, тел. 51-42-74

333000 СЕВАСТОПОЛЬ ул. Ленина 13

355102 СТАВРОПОЛЬ ул. Коминтерна 12, книжная база

200001 ТАЛЛИН ул. Пярну, инт. 10

700000 ТАШКЕНТ ул. Пушкина, КВЦ-1, корп. 6,

380009 ТБИЛИСИ ул. Ленина 22, тел. 93-44-78

720010 ФРУНЗЕ Ленинский пр-т 182, тел. 42-83-00





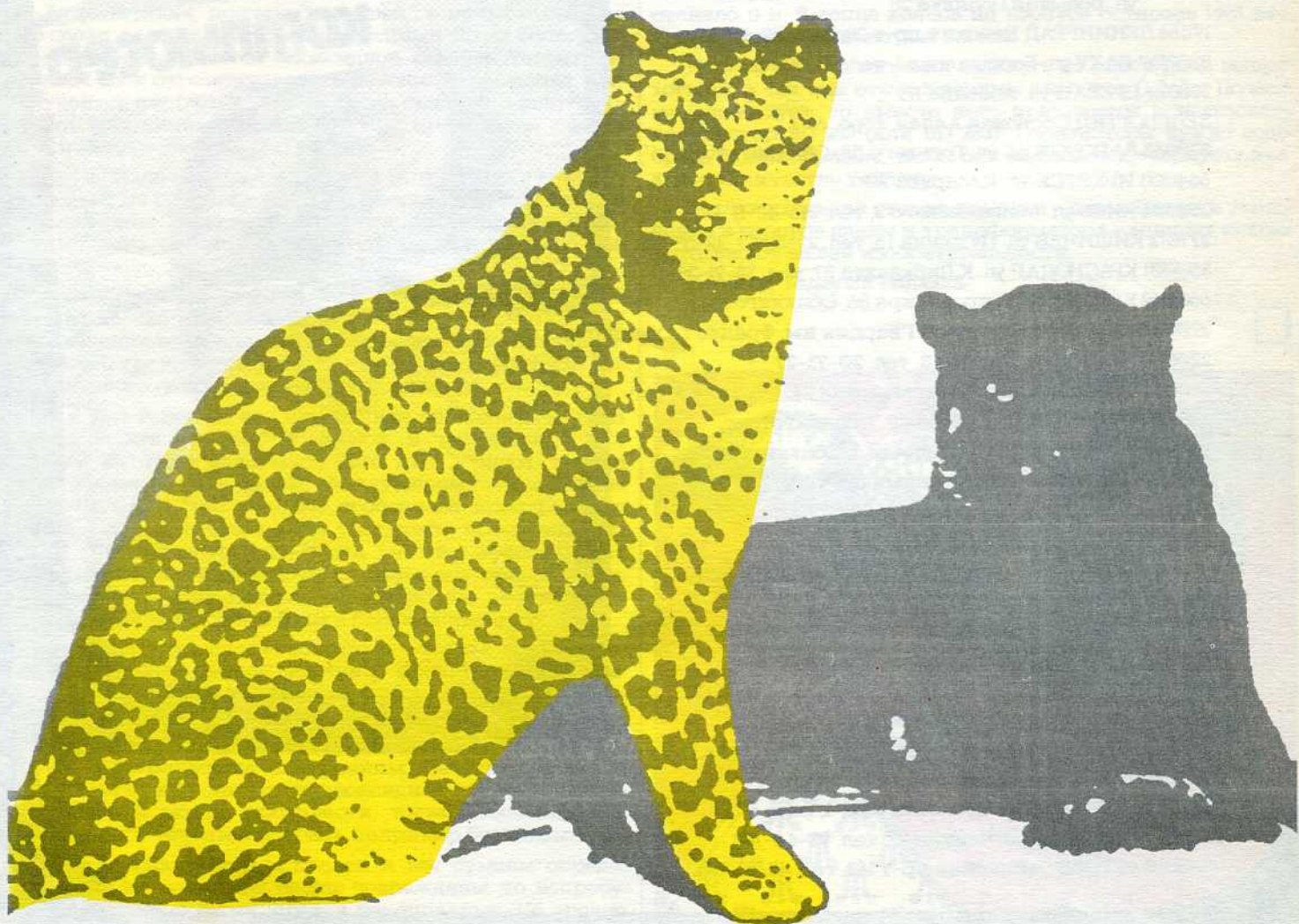
ИЗДАТЕЛЬСТВО "ФИНАНСЫ И СТАТИСТИКА" СОВМЕСТНО  
С РЕДАКЦИЕЙ ПОЛЬСКОГО ЖУРНАЛА **KOMPUTER**  
осуществляют выпуск сборника **kompiuter** на русском языке.  
Издательство "Финансы и статистика" принимает заказы на сборник  
и производит отправку наложенным платежом.  
ЗАКАЗ высылается по адресу: 101000 г. Москва, ул. Чернышевского, 7.  
Тел. 925-35-02, 923-18-68.

## ЗАКАЗ

От кого.....

Адрес..... Телефон.....  
(почтовый индекс указывать обязательно)

№ выпусков..... Количество экземпляров журнала "Компьютер".....





\компьютер дома\

© Петр Грабчинский

# Секреты ATARI XL/XE

Продолжаем описание адресов и функций простейших ячеек памяти домашнего компьютера ATARI XL/XE. Напоминаем:

\* POKE (англ. запихивать, заталкивать) - позволяет поместить в определенную ячейку памяти любую величину от 0 до 255;

\* PEEK (англ. заглядывать, подсматривать) - позволяет проверить, какая величина находится в данный момент в определенной ячейке памяти компьютера.

559 - POKE 559,0 выключает ANTIC, в результате чего гаснет экран, а вычисления ускоряются на 30%, например:

```
10 GOSUB 110
20 GOSUB 100
30 GOSUB 120
35 REM то же самое, но ANTIC выключен
40 GOSUB 110
50 Z=PEEK(559):POKE 559,0:GOSUB 100
60 POKE 559,Z:GOSUB 120
70 END
100 FOR T=1 TO 100:W=T*SIN(T):NEXT T:RETURN
110 POKE 18,0:POKE 19,0:POKE 20,0:RETURN
120 X=256:V=X*X*PEEK(18)+X*PEEK(19)+PEEK(20)
121      ??:?      "=>Время      вычислений
";V/50;"секунд":RETURN
```

560, 561 - содержат адрес начала Display List (DL - это инструкции, благодаря которым ANTIC "знает", как выглядит экран и как выводить информацию).

621 - POKE 621,255 выключает клавиатуру, 0 - нормальное состояние.

622 - POKE 622,255 замедляет передвижение текста программы при выводе на дисплей.

694 - с 0 - нормальные символы, с 128 - инверсные, относится только к символам, введенным после изменения содержимого этой ячейки.

702 - с 0 - строчные буквы, с 64 - прописные буквы, с 128 - символы в режиме "CONTROL".

710 - содержимое определяет цвет фона экрана в графических режимах 0 и 8 (нормальное значение - 148); например, POKE 710,192 дает черные символы на зеленом фоне.

729 - определяет время от нажатия клавиши до автоматического повтор; нормальное значение - 40 (система PAL), 0 - без повтора 1 - немедленный повтор.

730 - определяет частоту повторов клавиатуры, нормальное значение - 5; 0 - одно повторение, 1 - многократный повтор с большой частотой.

731 - величины, отличные от 0, выключают звук, сопровождающий нажатие клавиши; нормальное значение - 0.

732 - информирует о нажатии клавиши HELP; 17 - нажатие HELP, 81 - HELP + SHIFT, 145 - HELP + CONTROL, 0 - блокирует HELP.

741, 742 - содержат последний адрес X свободной памяти:

$X = \text{PEEK}(741) + 256 * \text{PEEK}(742) - 1$

743, 744 - содержат первый адрес свободной памяти.

752 - POKE 752,1 выключает курсор (нормально 0).

755 - POKE 755,n влияет на вид курсора и символов; в зависимости от величины n (от 0 до 7) курсор может быть видимым или невидимым, прозрачным или нет, а символы будут появляться в нормальном виде или "вверх ногами".

756 - POKE 756,204 включает так называемые международные символы; нормальное значение - 224.

763 - содержит код ASCII последнего символа. Пример использования:

```
10 OPEN "2,4,0,"K:"
20 GET #2,A
30 PRINT PEEK(763); " ";CHR$(A)
40 GOTO 20
```

764 - содержит внутренний код (Internal Code) последней нажатой клавиши (255 означает, что ни одна из клавиш не была нажата).

```
10 PRINT PEEK(764)
20 GOTO 10
```

765 - содержимое этой ячейки определяет цвет, который по инструкции XIO 18 будет использован для заполнения фона.

767 - POKE 767, 255 останавливает вывод текста программы на экран (равнозначно нажатие CONTROL-1), нормальное значение - 0.

838, 839 - POKE 838,202:POKE 839,254 посылает вводимые с клавиатуры символы на принтер.

43234 - содержимое ячейки определяет версию встроенного Бейсика: 234 - версия C, 96 - версия B, 162 - версия A (только модели 400/800).

53279 - содержит код нажатия функциональных клавиш: 0 - OPTION+SELECT+START, 1 - OPTION+SELECT, 2 - OPTION+START, 3 - OPTION, 4 - SELECT+START, 5 - SELECT, 6 - START, 7 - ни одна клавиша не нажата. Пример использования - петля ожидания нажатия клавиши START:

```
100 IF PEEK(53279) < 6 THEN GOTO 100
```

54018 - управление магнитофоном; POKE 54018,52 - мотор старт, POKE 54018,60 - мотор стоп.

Перевод Анджее Поплавского





В следующем выпуске: Большие секреты ZX Spectrum и маленькие секреты TR DOS / ZX Spectrum и принтер / Подключаем Spectrum к IBM / О ячейках памяти маленького Atari и не только / Какое Бейсик выбрать? / Полезные советы начинающим взломщикам. Много других материалов о домашних компьютерах.

